Routing Protocols for Meshed Communication, Networks Targeting Communication Quality of Service (QoS) in Rural Areas.

by
Edmundo B.F Chissungo

A dissertation submitted to the Department of
Computer Science at the University of Cape Town
in fulfillment of the
requirements for the Degree of
Master of Science

CSC 5000W

Cape Town, South Africa

2013

ABSTRACT

CHISSUNGO, EDMUNDO B.F. Routing Protocols for Meshed Communication, Networks Targeting Communication Quality of Service (QoS) in Rural Areas. (Under the direction of Professor Edwin Blake.)

Rural areas in Africa often have poor telecommunication infrastructure. Mobile phones, if available, are frequently unaffordable to most users. Wireless mesh networks (WMNs) offer an alternative possibility of low cost voice and data communications.

The focus of this research is a laboratory study of WMNs that mimic conditions found in rural areas. This work investigates routing strategies for the Mesh Potato (MP). The MP is an effective alternative communication technology that has minimal configuration requirements, low cost of deployment, low power consumption and resilience that make it an attractive choice for rural areas. The MP runs a new mesh networking algorithm called the better approach to mobile *ad hoc* networking (B.A.T.M.A.N or Batman). This allows a WMN to be established in which users can use plain old telephones to talk to each other using Voice over IP (VoIP).

Batman daemon (Batmand) is the implementation of Batman algorithm used by the MP. Batmand is a minimalistic routing protocol which performs well in laboratory experiments. The question raised is whether adding more service specific routing metrics improve the quality of service (QoS) observed in Batmand network in practice. The research investigates delay, packet loss, throughput and jitter as performance parameters (metrics) that may serve as options to improve the simplistic Batman algorithms route selection process. These metrics are essential for QoS in voice- and data-sensitive networks. Specific focus was given to delay and it is the metric added to Batmand. In addition the research examines how well the different applications such as voice and data are supported on the Batmand network under different routing scenarios.

The research approach adopted in this dissertation was experimental and an indoor testbed was created to replicate the basic scenarios encountered in the rural environment. The essential characteristics found in the Mdumbi region of the Eastern Cape, South Africa, were taken as a case study in this dissertation. The testbed was used to compare the original Batman algorithm implemented as Batmand, referred to here as O-Batmand, routing protocol and the resultant Batmand version obtained from the addition of the delay-routing metric called modified Batmand (M-Batmand).

The research produced a number of findings. As the number of hops increased the per-

formance of the network decreased for both protocols. O-Batmand is well suited for the task of routing packets inside a wireless network. It is designed and works for voice packets and supports data services. This is also true for the M-Batmand implementation.

M-Batmand was developed as an improvement to the O-Batmand implementation at the cost of increased complexity, experienced by the protocol through modifications of its route selection process. The modification involved adding network delay values to its route selection process. This addition resulted in a protocol that is delay sensitive; however, the overall performance gains were inexistent. The main conclusions drawn from this study are that O-Batmand cannot be modified to include additional metrics and be expected to improve its performance. Second conclusion is that M-Batmand did not improve the overall performance of the O-Batmand protocol. The addition of the delay metric actually hindered O-Batmand's performance to the extent that no overall performance gains were realised. Sources of performance degradations are: increased overhead, from added delay data, in the network control packets called originator messages (OGMs). M-Batmand performs calculation which O-Batmand did not increasing CPU cycle needs. Lastly upon further internal protocol investigation it is seen that the rate of route delay data updates is slower than the original metric used by the protocol. This creates route fluctuations as route selection process will change when the updated delay values are added and change again when there are not as the network obtains the updated delay data.

Both protocols support voice and data, however, the results show that the quality of the network deteriorates in the testbed with increasing hops. This affects voice more so then it does data as routes become more unstable with each increasing hop. Further Batmand is best at supporting voice and data as it outperforms M-Batmand in the laboratory experiments conducted.

This dissertation argues that while there may exist one or a combination of metrics amongst the researched list (delay, packet loss, throughput and jitter) that may actually improve the performance of the protocol, it is extremely hard to realize such gains in practice.

DEDICATION

To my parents Antonieta Bene Moiane and Filipe Chidumo, family, teachers and friends.

BIOGRAPHY

The author was born in the capital of Mozambique, Maputo. This vibrant city overlooks the Indian Ocean which also servers as tourist destinations and a source of the many seafood products Mozambique is famous for. The author has lived in Johannesburg South Africa, where he first learned English, Madrid in Spain and also New Rochelle, Westchester, New York, USA.

ACKNOWLEDGEMENTS

I would like to thank my Mother, Antonieta Bene Moiane, for serving as inspiration in my times of difficulty. Her strength and continued endeavour in her academic, social and family life gave me the strength I needed to complete this thesis.

I would like to express the deepest gratitude to my Father, Filipe Chidumo, whose hard work paved the way for me to be where I am.

In addition, I thank you to Professor Edwin Blake for proposing this project and guiding my writing of the theses.

Also a thank you to, Dr. Hanh Le, for her continued and much appreciated guidance and patience with me through the project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

- Ad Hoc - In networking it is: nodes on a network that inter connect forming a network requiring little or no planning
- AP - Access Point
- Batman/B.A.T.M.A.N - Better Approach To Mobile *ad hoc* Networking
- D - Delay
- ICT - Information Communication Technology
- ICT4D - Information Communication Technology for Development
- IP - Internet Protocol
- J - Jitter
- LAN - Local Area Networking
- M-Batmand - modified Batmand protocol
- OGM - Originator
- OLSR - Optimized link state routing protocol
- OSI - Open Systems Interconnection
- O-Batmand - original Batmand protocol
- PLR - Packet Loss Ratio
- POT - Plain Old Telephone
- QoS - Quality of Service
- RT - Routing table
- SoC - System on Chip
- Tp - Throughput
- TQ - Transmission quality
- TTL - Time to leave

- UCT - University of Cape Town
- VoIP - Voice over IP
- WiFi - Wireless Fidelity (trade mark name). Also used to refer to devices using the wireless free spectrum to send or receive data.
- WLAN - Wireless local area networks
- WMN - Wireless Mesh Network

Chapter 1

Introduction

Africa (as shown on Figure 1.1) is the world's second largest continent, boasting a wide variety and abundance of natural resources. Africa is also the world's poorest and most underdeveloped continent in the world. On the continent, the Sub-Saharan region (as shown on Figure 1.2) is the area with the worst levels of poverty and underdevelopment. This dissertation looks at a rural area in Sub-Saharan Africa, as an example of typical communications issues there in. African development has been stagnated by numerous factors, the most prominent being corrupt governments [5]. These governments often fail in central planning resulting in high levels of illiteracy [60]; serious human rights violations. It even sparked the creation of the Mo Ibrahim Foundation [54] which promotes good governance through monetary prize incentives, awarded to the non-corrupt heads of state.

The lack of stability on the continent has led to a lack of access to foreign capital, thus further crippling the rate of development and increasing poverty [66, 37, 14]. These factors further widen further increase the lack of communication infrastructure faced by this region.

Sub-Saharan Africa is the geographical term used to refer to the area of the African continent that lies south of the Sahara desert. There are 33, out of 49, countries in the region that are considered least developed by the United Nations (UN)[60]. This work understands the lack of development as poor telecommunications infrastructure. It is noted that the number of telephone lines per 1000 people is about five; one-twentieth of the average for other developing countries. The cost of local telephone calls is 100 percent higher than the average for developed countries [60].

Rural areas are described as areas with low population density: with individuals sparsely distributed over the region, and often characterised by the lack of infrastructure commonly found in urbanized areas. For instance, less than 40 percent of rural Africans live within two kilometres of an all-season road. This is the lowest level of rural accessibility in the developing world [23]. In these circumstances, market forces alone cannot drive the deployment of the necessary communications infrastructure, be those copper wires or fibre optics, which would serve as the necessary backbone for other services. The market views such endeavours as expensive, as the cost per user is much higher in rural areas than it is in urbanized areas. Reasons for the cost spike are a combination of the following: in rural areas, users are dispersed (coverage) and are often not familiar with the current technologies, resulting in lower than anticipated levels of use. Another aspect to be considered is that even if the aforementioned issues were solved, individuals in these areas are poor, and the services the infrastructure offers require fairly ex-pensive equipment. This equipment e.g., smart phones, laptops, tabular PCs, etc., can take full advantage of the services offered. In most circumstances the individuals in rural communities cannot afford these. These are just some example of many issues faced by these developing countries. Others listed here:

- Electricity: Need to power electronic devices such as any communication infrastructure.
- Landline Telephone: The necessary equipment for this is lacking. Individuals have to seek other more expensive recourse.
- Fibre Optic cables: There are no market forces to drive the expansion of this infrastructure to these areas and as a result the development lags.
- Cell Phone coverage: Coverage is low and its usage highly expensive.
- Paved roads: These rural areas often have dirt or rocky roads which are more like footpaths than actual roads, adding to the difficulties faced in these areas, especially since the roads would be needed to get the infrastructure to the planned locations.
- Devices: Those that can access communication services offered by the deployed networks infrastructure need to be affordable. This is essential, as individuals in these areas form part of the countries low income group.

A large percentage of Sub-Saharan Africa remains rural. There are numerous projects that have been initiated whose goals are to find solutions to the problems in rural areas, one such project is run by the Village Telco group [29]. Other projects are: ”Digital media to reduce maternity mortality in Sierra Leone” and ”Adapting a novel situated display system for an educational context”, are just a few of these projects. This dissertation will form part of the group of projects dealing with communication solutions aimed for rural areas.

Figure 1.1: Image of the African continent, depicting the old world view, as one large underdeveloped area of the world.

Figure 1.2: Image of the African continent, showing countries within the Sub-Saharan region.

1.0.1 ICT4D

There are many avenues to choose from as a strategy to tackle the issues found in rural Africa. The avenue chosen by this dissertation is called information communication technology (ICT) for development (ICT4D). ICT4D is a discipline that focuses on existing information ICT to create solutions for many of the known and some unknown problems found in rural Africa. ICT4D is a tool suited to task of helping rural Africa as it uses technologies appropriate for development in these poor areas, and methodologies for working with groups of users who have had no previous experience with digital technology, as these groups are found in these areas.

The project discussed here is an ICT4D approach to finding a solution for an existing telecommunication problem which plagues South Africa's rural communities. The individuals for whom the solution will serve are those that have had no previous experience with digital technology, as is often the case with most ICT projects in Africa. The region focused on is rural South Africa.

1.1 The Need to Communicate and its History in South Africa

The need to communicate can be considered a basic human need. Communication has allowed for the creation and progression of societies, and is also one of the many factors that push development forward. In today's world, communication is everything. It is what drives the creation, sharing and dissemination of monolithic amounts of data, which populate the biggest communication medium known to man, the Internet. Communication in South Africa, where our attention is drawn to, is young. Young in this context refers to the comparison between the state of communication in first-world countries and that of South Africa. This is illustrated in Figure 1.3, which shows the statistical penetration of telecommunications in Africa as compared to the world [39].

The evolution of communication in South Africa has progressed much faster than other Sub-Saharan countries. The first use of telecommunication in South Africa was a single line telegraph connecting Cape Town and Simons Town. As communication needs grew, South Africa introduced its first undersea links. Initially, these links connected Durban the commercial port of South Africa with Europe. By the 1960s, South Africa was connected to the world, and therefore, able to establish communication both domestically and internationally. Domestically, the operator was the South African Post Office. Then from 1991 Telkom took over operations. Telkom is currently a wireline and wireless telecommunications provider in South Africa. Furthermore, domestic investment occurred with landline cables deployed all over the country; however, costs of usage of the network remain higher than more developed nations in the world. This and other factors slowed the expansion of landline cables to rural areas in South Africa. This left a portion of the population without one of the key infrastructure needed for development in those areas. After some time the cell phone age came about, as shown on Figure 1.4, cell phone penetration quickly overtook, and now exceeds, that of the landline. Cell phones offered and still offer an alternative to the landline infrastructure. However, the high cost of usage still motivates the need for an economically viable alternative.

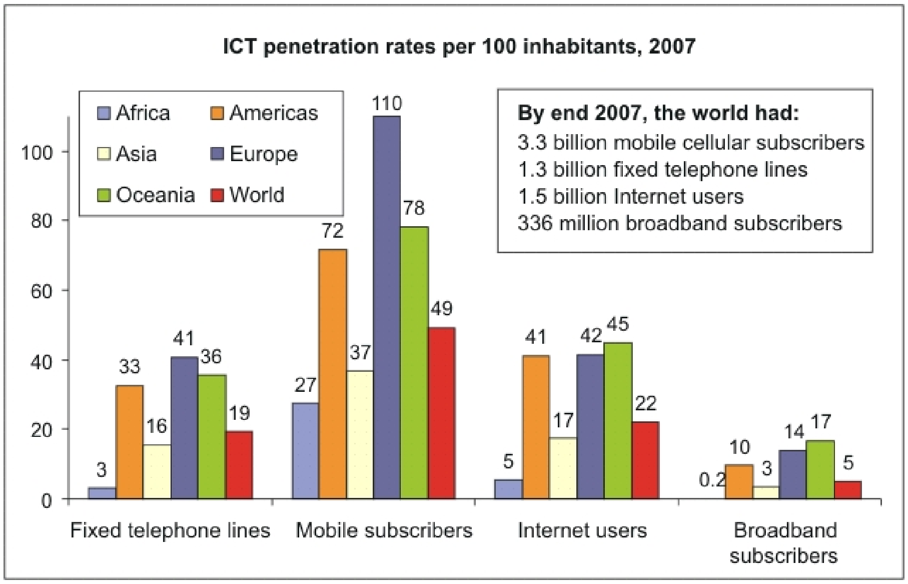


Figure 1.3: Image obtained from the International telecommunications union (ITU). Shows the state of global Information technology (ICT) in 2007. Source http://www.itu.int/ITU-D/ict/statistics/ict/index.html

1.1.1 The State of Telecommunications in Rural Africa, their Consequence and Alternatives

Figure 1.5 shows that rural areas often lack telecommunication infrastructure. These include the most basic infrastructure used for landline telephone communication. As a result of this, and the lack of alternatives to satisfy the need to communicate, rural communities have turned to using cell phones [30]. This might not seem like an issue on the surface, but it is. Cell phone operators in South Africa have high network charges on calls and other network services. This is coupled with the uneven distribution of wealth which leaves a larger portion of the population earning very little. The effects of high network charges are felt by the poor, as the charges add to a substantial portion of their income. This leaves many families with their earnings affected by their need to communicate.

There are technologies out there that have been used in rural areas where there is no infrastructure established. One such technology is the VSAT (Very Small Aperture Terminal) technology. It is a communication network composed of VSATs at each of the sites, potentially servers, hubs, switches and PC's. This is an example of a typical solution used in rural areas. This solution however, requires multiple components which cost more than individuals in rural areas can afford. The cost of purchasing and the technical expertise need to instal the devices can cost USD30,339.00 as seen quoted by [13]. Added to this is the cost of paying and individual or organization to set the network up and trouble shoot it when needed further driving costs up. Further, there are other technologies offering potential solution to communications issues in rural areas. These networks have a base station model such as cellular networks like GPRS (General Packet Radio Services) and WiMAX. These base stations with values in the tens of thousands of dollars (values varying depending on source) [7] are expensive and require a large number of client to spread the cost thus lowering it. Also WiMAX has high spectrum license costs in most countries. In rural areas the number of clients would be low so is not economically viable [67]. A solution cheaper both in components needed, ease of deployment, individuals to set up and maintain is what this dissertation will look at and the Mesh Potato (MP) [29] is just that.

In 2008, the MP was developed (Figure 1.6 shows the second generation MP). The MP (currently USD99.00) [68] is an alternative communication medium and may serve as a solution to the problem of high costs of communication by serving as a cheap alternative. The MP is in essence a wireless router connected to a plain old telephone (POT), which allows individuals to call each other wirelessly much like a regular landline. MPs are interesting as they are used for voice communication over wireless mesh networks (WMN, a type of radio based network system). These networks, as will be explained later, have features such as minimal configuration requirements, low cost of deployment and resilience which make it an attractive choice for rural areas. The MP is an attractive choice as it was designed for the purpose of being the alternative to other forms of voice communication devices. It is open hardware and uses open source software, making it easy and cheap to acquire and tweak if necessary. It is also easily deployed as it comes ready for use out the box.

The group that started the MP endeavour has also deployed the device in various regions in the world. Two well know deployments are those found in East Timor (Dilli MP Network) and Cape Town, South Africa (the Bo-Kaap community MP network). These real world testbeds have served to demonstrate the feasibility of this technology as a successful alternative to both cell phones and landline communication. The MP runs a mesh networking software called the better approach to mobile ad hoc networking (B.A.T.M.A.N or Batman). This protocol works well on the MP as it is not resource demanding (CPU usage, RAM footprint and storage space) and therefore, suited for the resource constrained MP.

The MP is new, and like most new technologies there is room for improvement. The improvements are made possible due to its open source and hardware nature. The improvements that can be thought of as the technology matures are scenario and usage specific. This means that device and its software can be improved to better serve the environment and individuals for which its use is intended. This dissertation will focus on improvements for users in rural areas.

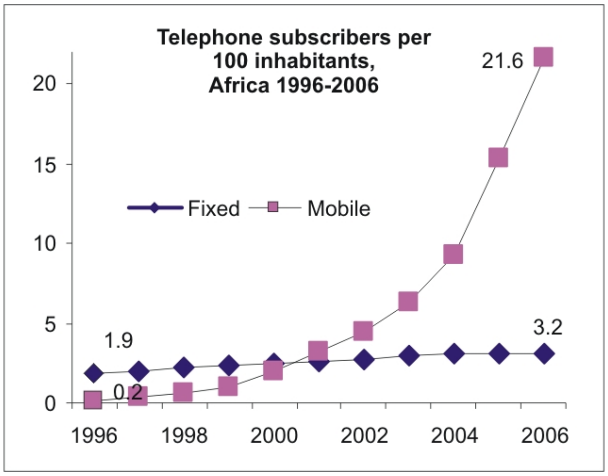


Figure 1.4: Image shows the how the cell phone serving as an alternative to the wireline telephone far exceeds in market penetration. Source http://www.itu.int/ITU-D/ict/statistics/ict/index.html

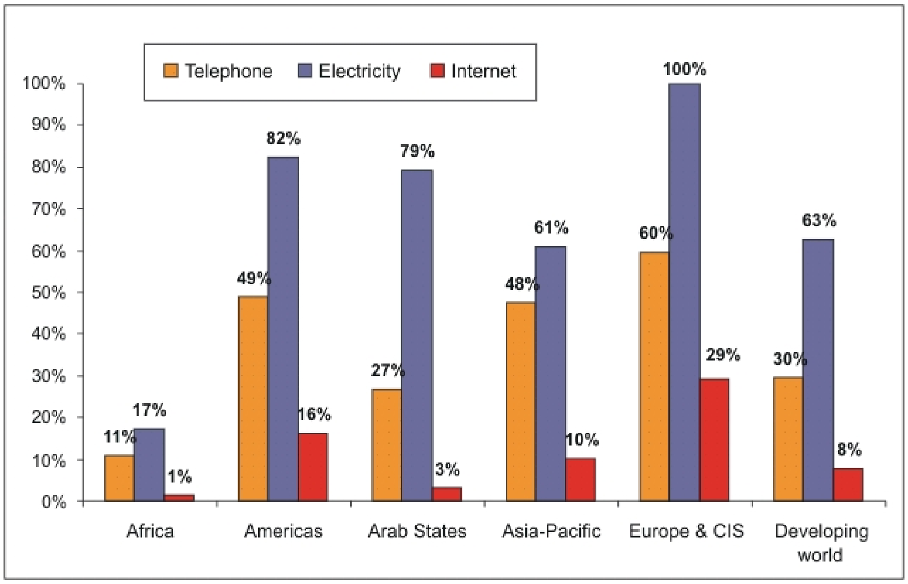


Figure 1.5: Image of the statistic from 2006, showing the basic ICT and infrastructure access in Africa as compared to the world. Source http://www.itu.int/ITU-D/ict/statistics/ict/index.html

Figure 1.6: Image of the second generation Mesh Potato (MP)

1.2 Performance impact on routing protocols in rural meshes

The improvements this dissertation will focus on can benefit from looking at the factors that impact the performance of routing protocols for rural area meshed communication. Meshed communication in rural environments benefits from the stable and predictability of its underlying topological structure. Nodes in these areas are stable or quasi-static, making them similar to wired networks. In these networks routing protocol performance result from link failures rather than unpredictable node mobility [21]. Rural meshed communication networks suffers from link outages caused by a number of reasons. The most obvious and direct one is the unreliable power which causing outages on links. Other sources of link outages are:

- Unmanageable congestion or signal loss.
- Users could disrupt and unplug nodes at will[38].
- Nodes with relatively poor connectivity were susceptible to noise from other transmitting nodes which cause link interference.
- Transmission power in an unplanned network causes performance degradation through interlink interferences. Unplanned networks in this context refer to networks where the designers did not include node transmission power in their planning of the network [61].

Lastly asymmetric links also cause performance issues for routing protocols. These asymmetric links prevented route discovery [6] causing network convergence to slow down and thus routing and communication in the network in hindered. This is all part of variable link quality issues often experience in these types of networks.

Single point failures are another factor of mesh communication that impacts the performance of the network and its routing protocol. This occurs when the number of neighbours nodes in a network have is low, this is also referred to as poor link density [6]. The result is that critical links will be introduced in the network. These critical links will be the single points of failure where if certain links fail so will sections of the network. This is especially and issue in circumstances where the mesh is used to offer Internet connectivity to nodes.

1.3 Directions to Improvement

Inspired by the poor telecommunications infrastructures depicted throughout this chapter an alternative communication device and medium has been thought of. The use of the wireless,free, spectrum in WMNs and within it the Batman routing protocol to forward packets to and from the MP devices. The MPs serve as the alternative communication device and the WMN as the medium.

Each MP in a MP network behaves as both a client in the network, receiving and sending information (data), and also as a router, forwarding data on behalf of the network. The act of forwarding data on behalf of other MPs in the network is the task of the Batman routing protocol which resides inside each MP and decides which MP is best to forward the data to. This decision is made by its metric which measure the performance of the paths available between this MP and the rest of the network. This assisted forwarding of data is needed as not all MPs that desire to communicate with each other are within direct communication range of each other to do so. Therefore, the use of the nodes in between itself and the MP it desires to communicate with is what make a WMN. Each node the data has to be forwarded from on its way to the desired communication end point (MP) is referred to as a hop.

As mentioned new technologies such as the MP often have room for improvements. There are a few manners of improving the MP. These are listed here: The first would consist of constructing a network in the rural environment, then use a network visualization software tool such as Afrimesh, details for which can be located here [2], to visualize the network. Then identify the faulty links in the network, and use supernodes to then help boost the signals of these poor links. The second would consist of the use directional antenna to boost any nodes' signal range in a particular direction. However, these options are not the best given the qualities needed in a viable solutions. Firstly, one of the key issues around choosing an alternative to wire landline telecommunications is ease of deployment, as this keeps the set up costs low. These approach would increase the amount equipment needed which would drive costs up and would also require a learned mind to assist in the set up, further driving cost up. Lastly these options are almost guaranteed to work, offering little research direction.

The research direction required to meet the requirements is an automated one which involves investigating a known routing metric, best suited to improving the routing carried out by the Batman routing protocol. An automated approach would result in the modification of the Batman algorithm and its protocol. The effect is that the no extra equipment nor learned minds are need as the original idea of the MP being easily deployable is maintained as the improvements shall be at a routing level and therefore, software based.

Lastly an automated approach to improving the Batman algorithm and its protocol not need to be tested in the field as the other two approaches require. Instead the characteristics of the rural areas can be abstracted through field studies and mimicked in a laboratory study making being in the field unnecessary.

1.4 Aims of the project

Inspired by the poor telecommunications infrastructures depicted throughout this chapter an alternative communication device and medium has been thought of. The use of the wireless-free spectrum in WMNs and within it the Batman routing protocol to forward packets to and from the MP devices. The MPs serve as the alternative communication device and the WMN as the medium.

Each MP in a MP network behaves as both a client in the network, receiving and sending information (data), and also as a router, forwarding data on behalf of the network. The act of forwarding data on behalf of other MPs in the network is the task of the Batman routing protocol which resides inside each MP and decides which MP is best to forward the data to. This decision is made by its metric which measure the performance of the paths available between this MP and the rest of the network. This assisted forwarding of data is needed as not all MPs that desire to communicate with each other are within direct communication range of each other to do so. Therefore, the use of the nodes in between itself and the MP it desires to communicate with is what make a WMN. Each node the data has to be forwarded from on its way to the desired communication end point (MP) is referred to as a hop.

The main focus of the project is to provide better quality of service (QoS) in a laboratory study of the Batman and MP network set up to mimic certain characteristics abstracted from rural areas in South Africa. This is tackled in more depth in Chapter 3.

This is achieved by modifications to the Batman algorithm and protocol which in principle, will increase stability and reliability in the MP network, making the MP more effective in rural areas. This is needed because the current, default, metric used by Batman is a statistical one. The metric is a performance parameter whose values are used to compare various network paths available and determine the most suitable to send packets on. The shortcoming of this metric is that it only takes into account the one hop link quality to every other node in the MP network. Using those statistics as a basis, the routing protocol then chooses which paths to use. A new routing scheme, that also takes into account other network information, is needed. This will improve the MP network in areas such as performance and stability.

A new routing scheme can be attained by improving the path selection metric, also known as routing metric, already used by the routing protocol on the MP. The routing protocol is the software in charge of finding the best paths to usher data through the MP network, from source to destination. The main focus is therefore, on this routing protocol. The research direction taken is to investigate a known routing metric, best suited to improving path selection on the

MP network. Once chosen, the routing metric will be embedded onto the routing protocol. The chosen metric and the existing metric will work jointly in such a way that the Batman protocol can offer better QoS on the MP network.

1.5 Research Questions

Based on the above the following research questions where derived and subsequently answered in this dissertation through a laboratory study.

1.5.1 How does B.A.T.M.A.N perform in a laboratory setup mimicking aspects rural networks?

The aim is to investigate the MPs for use in rural networks this is achieved by abstracting certain characteristics observed in rural areas in a field study and mimicking certain aspects of rural environment in a laboratory study. The exact representation of a typical rural environment is not the aim the focus is more on the topological effects of the rural network. This will enable the dissertation to observe how Batman performs in this type of setup offering insight into aspects of the MP performance in rural networks. This was carried out on an indoor testbed, where the MPs were set up to mimic a simplified version of a full scale MP network. A different approach could have been to run network simulations, using a known simulator such as NS2. However, a more practical approach to understanding the MP network and its protocol was deemed more appropriate as laboratory studies offer more insight into the inner workings of WMNs that simulators do not offer, this is explained in more detail in Chapter 2.

1.5.2 Is it necessary to develop a different route selection metric, or even a different routing protocol, to provide better QoS in the network?

This question looks at the improvements the route selection metric will add to the performance previously experienced in MP networks. If time allows, the option of implementing a different routing algorithm will be explored. This will only be necessary if Batman's performance is poor and the improvements made by the route metric are minimal.

1.5.3 How are different applications, such as, voice and data supported on the Batman Network?

This question deals with the possibility of progressively increasing the number of voice and data packets in the MP network, while observing the behaviour of the network as load in the network increases. This will show how the network responds to and thus supports both the voice and data applications as the load increases.

1.6 Thesis Organization

The rest of the document is organized as follows. The next chapter focuses on the background of all the concepts need to understand the project as a whole. Chapter three describes the test design used to test our project and determines the outcome. Chapter four focuses on the system design that attempts to improve the existing B.A.T.M.A.N routing software. Chapter five is dedicated to the project's experiment and results. Lastly, chapter six draws conclusions on the work done in this dissertation.

Chapter 2

Background

In order to accomplish the goals set out in the introduction of this dissertation, a number of artefacts need to be looked at. An in depth look at these issues will offer a detailed view of the ideas and technologies described in Chapter 1 and as well as introduce ideas and technologies that support the arguments and questions posed in that chapter. Having presented these, this dissertation can proceed to explain how they work together to meet the goals of this study and answer the research questions. In line with this, this background chapter focuses on the artefacts and surrounding ideas that make up this dissertation. This chapter provides an in depth background on the mesh potato (MP) devices and the Better Approach to Mobile *ad hoc* Networking (B.A.T.M.A.N or Batman) routing protocol, and other major components which are predominantly used by the mesh potato (MP).

A review of the surrounding literature in the field of wireless technology is also provided, as this is the primary technology used by the MP to bring information communication technologies (ICT) to individuals in rural communities. This chapter is organized as follows: the chapter starts with the background on rural networks and then proceeds to describe the MP. Therefore, attention is given to the Batman routing algorithm and, lastly, wireless technology and its supporting standards are discussed.

2.1 Rural Networks and QoS

Many rural areas, as mentioned in Chapter 1, are large and isolated areas of a country typically located at some great distances from major cities. They characterized by low population density. Furthermore, they do not have the same infrastructure as those found in urban areas. Some infrastructure, such as roads, bridges and schools, are in sub-standard conditions while others may not even exist. The lack of these make the area hard to access and communicate with turning them isolated. Absent infrastructure ranges from access to services and facilities, such as electricity to access to telecommunication and information technology, the latter of which is the focus of this study.

The lack of infrastructure in these areas is a result of limited, if any, government or private investment in the ICT area. The kinds of investment needed are often high due to the costs of purchasing, installing them and maintaining wired technologies. As a result, individuals in rural communities seek alternatives such as cell phones which, in developing countries, come with high service charges. One other alternative is the use of VSAT (Very Small Aperture Terminal) technology, however this is also and expensive alternative with costs of equipment and installation being quoted at USD30,339.00 (at a rate of 0.9878 to the Canadian dollar where quote was obtained from On 10 May 2013) [13],

Since investment in wired technologies are hard to access, helping groups living in these areas requires the use of affordable alternative. In order to be viable the alternative has to fit the following criteria: On the financial side, the alternative technology needs to have minimal upfront investments costs so that it is financially accessible. In addition, installation and maintenance costs of this technology need to be kept low, so that long term use is viable. This often means that the alternative technology needs to be capable of being quickly configured and deployed in areas where there is little to no infrastructure. If these requirements are met, then the technology could serve as a cost effective alternative to wired technology.

On the technical side; in the vast majority of rural areas, such as Mdumbi Figure 2.1 which is used in this study as a case study, the members of the community live separated from each other by long distances; therefore, the technology chosen as an alternative needs to be able to maintain communications over long distances. Figure 2.1 is a photograph of the Mdumbi region in the Transkei, Eastern Cape, area of South Africa as shown on Figure 2.2. This area is considered to be rural and has all the characteristics described above. The technology also needs to be capable of delivering relevant communication services such as Voice over IP (VoIP) [33]. The alternative that is the most compatible with both the cost and technical requirements needed from an alternative to wired technology is a WiFi enabled device.

The WiFi enabled device, MP, incorporates technologies that offer rural communities access to ICT. These devices can exploit the free wireless spectrum; creating a radio packet, store and forward, network known as WMN. The details of these technologies are described in the WMN section of this chapter. Also, MPs could offer the people in the communities a cost effective alternative to cell phone and landline usage. The MP comes with the once off cost of the purchase of the device (currently USD99.00) [68] which is decreasing as the amount of MPs manufactured increases. The MP uses the free spectrum to communicate there are no charges to communicate. It also operates ”out the box” (no set up costs) and even with the need for power supply and potential maintenance costs it is more cost effective then cell phones,VSATs and landlines. These carry with them per call billing and other service fees charged on a monthly basis and in some instances in perpetuity. The MP matches the description of a fitting alternative to wired networks, while also offering access to ICT, as it is cost effective and technologically sound. The MP is described next.

Figure 2.1: Photograph of the Mdumbi region in the Transkei, South Africa, showing some of the lacking infrastructure mentioned in this dissertation and others. Here shown are a lack of tarred roads, widely available electricity and water.(source, [author]).

2.2 Mesh Potato (MP)

The MP was created by the Village Telco group [29] who describe the MP as a wireless System on a Chip (SoC), where the processor and all wireless functionality are combined on a single chip. The MP was initially developed for VoIP, using plain old telephones (POTs), but now also supports data. MP users dial the last octet of another MPs IP address and are able to make calls to other MPs on the MP network.

Figure 2.3 shows a first-generation MP without the POT. Figure 2.3 shows the ports where the POT can connect to the MP through a foreign exchange (FX) port. Also, the image shows other ports, such as the LAN connection which is another feature of the MP. All in all, the MP combines a wireless access point (AP), much like the one shown in Figure 2.4, with telephone capabilities, to bring into existence a device that enables its users to communicate using voice over the air.

The MP uses a modified version of the *ad hoc* mode called *ad hoc* demo profile in order to circumvent bugs generated by the manner in which the media access (MAC) layer functions. This is discussed in detail later in this chapter. The *ad hoc* demo profile allows any wireless node to connect to any other node within its communication range. Once connected, the nodes form a wireless blanket or cloud and, with the assistance of a routing protocol, create a communication network. More details on the *ad hoc* demo profile is provided in the Wireless Networks section of this chapter.

Communication on the MP is established on the wireless medium (air) through the assistance of the Batmand routing protocol. Communication on the MP is established using VoIP the standards. The voice packets sent around the MP network are GSM packets. According to the village telco group the total size of a GSM packet on the MP network is 73 bytes. This indicates that the MP uses GSM 6.10 codecs made up of 40 byte IP/UDP/RTP headers and a payload of 33 bytes. This version of the GSM packet is the one used for open source VoIP applications, much like the MP. The codec's Mean Opinion Score (MOS) is 3.7. MOS tests for voice are specified by the ITU-T recommendation P.800 [70] where the scale is 1 to 5, with 5 being excellent sound quality and 1 being really bad quality. In the case of this codec, it provides good perceived voice quality. More details on the MP software architecture is provided int he appendix A.3. Listed next are the other relevant hardware resources found on a MP:

- MIPS 4k processor 180 MHz
- 8 MByte Serial Flash EEPROM
- 16 MByte RAM

In the next section, the Batmand protocol, which is based on a routing algorithm called B.A.T.M.A.N. is discussed.

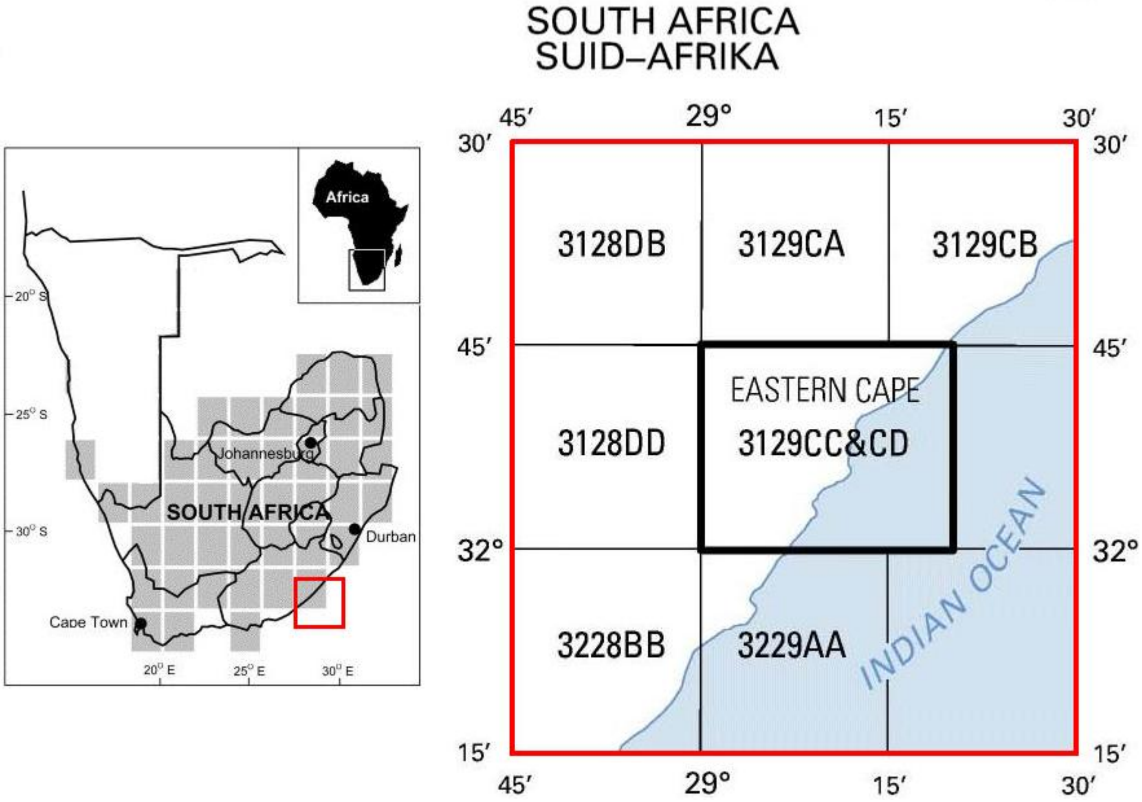


Figure 2.2: Image of the Map of South Africa (left) and the location of the specific region focused on (right), Mdumbi, Eastern Cape. The map on the left is divided into grids, shown by the green boxes, the easter cape grid is expanded onto the image on the right (used with permission, obtained from, [Chief Directorate, Center for the Geo-Spatial Information Cape Town, South Africa]).

2.3 B.A.T.M.A.N

Batman, designed in 2007and implemented the following year by Corinna “Elektra” Aichele [3] was intended for use in WMNs. Since then, it has been modified and extended by a community of open source developers [59]. At present, the Batman algorithm has been implemented as a daemon process called Batman daemon (Batmand), and as a kernel module named Batman advanced (batman–adv) for the Linux kernel. Both implementations are based on the initial algorithm developed by Corinna “Elektra” Aichele. This dissertation specifically focuses on Batmand. The Batman algorithm is explained here.

Figure 2.3: Image the first generation MP showing all the pots on all generation MPs

Figure 2.4: Image of a community wide AP

2.3.1 The Algorithm

As explained by Johnson, D., *et al.* [41], Batman does not maintain the full route to the destination. Instead, each node along the route only maintains the information about the next link through which the node can find the best route. In its routing table a MP node would have a list of all the nodes in the network (destinations) and next to each of those nodes the ID of the MP which provides the best one hop path to the destination node. node to forward the me The objective is to maximize the probability of delivering a message. Batman does not attempt to check the quality of each link, it simply establishes its existence. The protocol performs these checks by having all nodes periodically broadcast hello packets, known as originator messages (OGMs), to their neighbours. Broadcasting is when a single source sends messages to all available nodes in the broadcast domain/network. This is in contrast to unicast where a node sends messages to one specific node in the network.

Content included in the OGM packet:

- originator address
- sending node address: this is changed by receiving nodes and then the packet is re–broadcasted
- unique sequence number: The sequence number is used to check the concurrency of the message
- bidirectional link flag: used when the OGM packet received is its own and the sender is someone else
- time to leave (TTL)

Each node in a network routed by the Batman algorithm builds its routing table in the same manner. This process is described here.

When a node receives an OGM that is not its own, there are four possible scenarios, listed here:

1. The OGMs originator is not in its routing table, then a new entry is made for it. When this happens the ID of the node that forwarded the originators OGM is added to the routing table as a one hop neighbour that provides a path to the originator, and a new count is initialized and incremented. A count is the amount of OGMs from a specific originator that are received through a specific one hop neighbour.

2. The originator is in the routing table and the node that forwarded the OGM is new. In this scenario the ID of the node that forwarded the originators OGM is added to the routing table as a one hop neighbour to the originator, and count is initialized and incremented.
3. The originator is in the routing table, and the sender is not new. The count for the one hop neighbour node that forwarded the originators OGM is incremented.
4. The received OGM is its own, then it performs a test to check if the link between it and the one hop neighbour, that forwarded the OGM, is bidirectional, and the bidirectional flag is set for this path if the test is successful.

A node chooses the best one hop path by comparing the one hop links in terms of the number of OGMs that it received within the current sliding window. This value is called the transmission quality (TQ), and is the routing metric used by the Batman algorithm. Note that TQ is just a name and does not imply QoS. The sliding window is a fixed value and defines the acceptable range of the unique sequence numbers, afforded to each OGM, to be considered in the OGM count packet sent by a node. TQ is a route metric based on collected statics which are a nodes count of received OGMs, forwarded by a nodes one hop neighbour, which fall within the same sliding window. This metric only takes into account the one hop link quality to every other node. Using those statistics, Batman then gives TQ a value out of 255. The one hop neighbour with the highest TQ value is chosen as the path to use to reach an originator in the network.

Batman is, in essence, a proactive routing protocol, as it pre builds its routing table; however, the way in which it conducts route discovery and maintenance are unlike any other routing protocol and therefore, it does not fit into other pre–existing taxonomies [56]. Furthermore, as mentioned, the algorithm has three implementations, the most common ones are Batmand and Batman–adv routing protocols. The difference between the two implementations is that Batmand works on layer three of the Open Systems Interconnection (OSI) stack and works as a daemon process for the Unix operating system (OS). Batman–adv works on layer two of the OSI stack, and is a kernel module for the Unix OS.

2.4 Wireless Networks

A wireless local area network (WLAN) is a data communication system, implemented as an alternative to a wired LAN[74]. Most modern WLANs use radio waves, in the unlicensed, portion of the spread spectrum, 2.4 GHz, WLANs transmit and receive data over the air. The unlicensed portion of the spectrum is also referred to as the free portion of the spectrum meaning that it does not required licensing agreement between the users and the local communications regulator. In South Africa the regulator is the Independent Communications Authority of South Africa (ICASA) [32]. More details can be found on the Open Spectrum for Development, A South African Case Study document [64]. The standards for spectrum allocation are established by the International Telecommunications Union (ITU) [36]. Different vendors are allowed to create devices that make use of the WLAN technology and many do, however interoperability quickly became an issue.

To allow for interoperability between different vendor devices that make use of WLAN technology. The IEEE 802.11 committee was initiated in 1991 to develop standards for WLANs.

The standards set by the committee are called the IEEE 802.11x family. Today they are commonly known as wireless fidelity (WiFi), which is a brand name for the WiFi Alliance and currently is the most successful and also *de facto* wireless networking standard for WLANs [52, 12, 26].

In this family, the IEEE 802.11b has become one of the world's most popular wireless networking standard, so much so that it is the most prominent standard found on wireless network cards. Other standards that are also commonly found on wireless cards are the 802.11 a, g, n and s.

The IEEE 802.11s working group was set up in 2004 aimed at standardizing and therefore harmonizing the different WMN solutions that exist today by standardizing the architecture of WMNs. The IEEE 802.11s has been given two extra layers to the known network architectures for the physical interface (802.11a,b,g and n). These two are e which defines access to the medium and i for security. The ultimate goal of the working group is to provide a protocol that automates the entire mesh including the topology formation and paths between nodes in order to support multiple traffic types, namely broadcast/multicast and unicast all in a four-address frame format mesh [28]. The 802.1s mesh will appear to other networks as a layer two link forming a single broadcast domain. The 802.11s draft standard provides flexibility in the choice of the path selection protocol and in the link metric. The unique constraint is that only one path selection protocol and one metric can be used at a time. To ensure interoperability, the draft describes a mandatory path selection algorithm (Hybrid Wireless Mesh Protocol (HWMP)) and a mandatory link metric (Airtime Link Metric) [28] this metric is described under the path selection metric section in this chapter.

2.4.1 Card operating modes

The wireless cards are the actual hardware components that offer the device (such as the MP), on which it sits on, the capabilities to communicate with other devices that also have a wireless cards. Wireless cards and their standards operate on layer one and two of the OSI protocol stack. These cards have modes of operation where each mode enables the card to behave in a specific manner.

These modes are, *ad hoc* and infrastructure mode. Each mode sets out low–level specifications for how all wireless networks operate. The infrastructure mode makes the card dependent on an AP. These kinds of networks have specific nodes set aside to perform the communication routing on behalf of the nodes in the network. The image in Figure 2.5 shows various examples of networks created with wireless cards set to infrastructure mode. The image even shows how dedicated APs can be the backbone that route data to a gateway that can access the Internet. The *ad hoc* mode creates a point–to–point LAN bridge, more commonly known as wireless *ad hoc* or mesh networking, shown in Figure 2.6. In the context of *ad hoc* wireless networks it is interesting to note that in Latin, *ad hoc* literally means “for this”, further meaning “for this purpose only” and thus usually temporary [24]. There is a third mode that is often not known, called *ad hoc* demo (AHDEMO or Pheudo *ad hoc*) mode [73]. This is the mode in which the wireless cards on the MP and any other cards running the Batmand protocol have to be in, in order to communicate.

AHDEMO is a mode void of beaconing (no management frames, no association, no probes, just data). These beacons are those used by the cards to check the availability of the medium before use. All the nodes in the mesh need to have a channel manually assigned to them and will not hop between channels while communicating. This is key as all other modes have channel hoping during communications in order to achieve avoid over crowding on certain channels. All MPs come set to default on channel one. The advantage of this mode is that there is slightly better throughput than regular *ad hoc* mode because the card does not spend time sending beaconing and response packets before using the channel.

The Batmand routing protocol, together with the MP device on the AHDEMO mode, form what is known as WMN. WMNs are characterized by the absence of a centralized infrastructure [40]. In this type of network architecture there is no distinction between an AP and the client nodes [4] and communication is carried out directly between nodes within range, thereby making WMNs infrastructure independent.

The advent of new wireless technologies, in particular WMNs, has lead to an unprecedented opportunity to provide rural areas with access to ICT at a fraction of the cost and in a fraction of the time than would previously have been required using wired LAN [44]. WMNs can be installed incrementally, one node at a time, just as needed [42], and thus there are minimal up-front investment costs. In addition, the installation and maintenance costs of wireless networks can be kept low. Low costs offer a cost effective alternative to wired networks [31]. This is also true for networks that use both wireless and fixed-wired infrastructure such as the cellular networks.

WMNs do not use any fixed-wired infrastructure. This is in direct contrast with conventional wireless mobile communications, which are usually supported by a wired fixed infrastructure. Furthermore, a mobile device would use a single–hop wireless radio communication model to access a base–station that connects it to the wired infrastructure. On the other hand, a WMN uses multiple hops to communicate with other nodes directly without the need for a base station.

Lastly, the limitations in communication that can be experienced are due to the range of the wireless cards. Any card outside the communication rage of another wireless card, even if in the same network, will not be able to communicate. This problem arises because the cards do not provide a means for information, in the form of data packets, to travel from a node at one end of the network to another node at the other end, unless directly connected. Therefore, packets can only travel to their nearest (within radio range) neighbour. In order for all nodes to be able to communicate a packet forwarding system is needed, this is known as routing functionality and is described next.

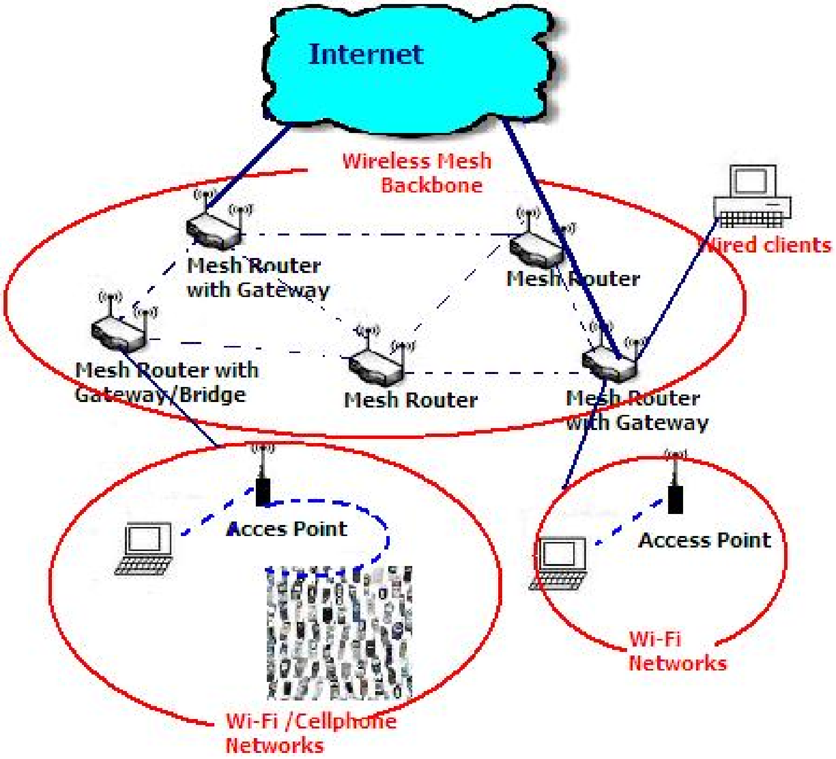


Figure 2.5: Image of an example of an infrastructure mode set network (source, [author])

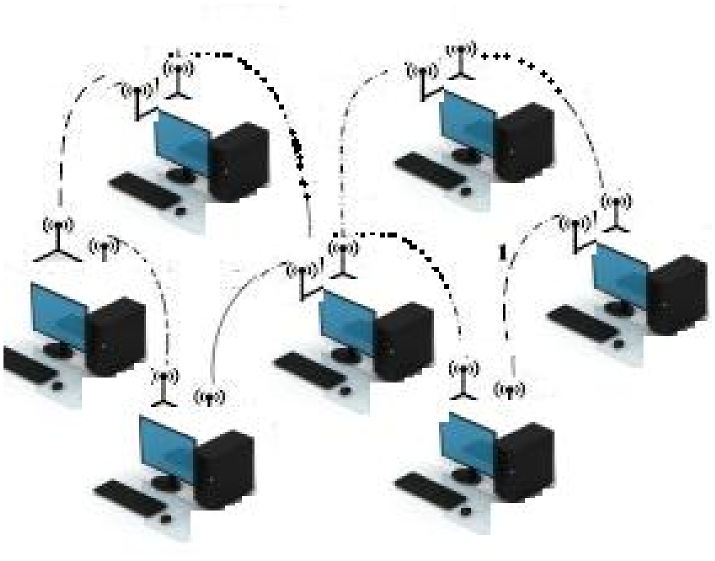


Figure 2.6: Image of an example of a wireless mesh network (WMN (source, [author])

2.4.2 Routing

Wireless devices behaving as nodes in a WMN do not have mechanisms for communicating with other nodes in the network with which they are not directly connected. Routing protocols exist as such a mechanism for doing this and are in charge of performing data forwarding (routing) between nodes and thus help form *ad hoc* networks. There exists that an abundant number of routing protocols [47] some sources suggest over 70 exist, each fitting into a pre-existing taxonomy.

The taxonomy is based on the routing strategy used by a routing protocol, which can be generalized so that protocols with similar strategies can be grouped together into categories. The two main categories are the proactive and reactive routing protocol groups. Proactive routing protocols are also known as none-adaptive (static routing) and are characterised by networks where the routes are pre–computed and the topology is either static or slow changing.

Reactive routing protocols are also known as adaptive (dynamic routing) and are characterised by a network topology where routing decision are made on demand (as roots are needed). There are also hybrid routing protocols, which are a combination of the proactive and reactive routing protocols and either lean more towards proactive or reactive, depending on their design decisions.

There are other routing strategies that can also be generalized to form groups. There are the alternative routing and multi path routing strategies, which are based on random decisions and network states (changes all the time) respectively. Then there is the distributed strategy, where network information is received from adjacent neighbours then used to determine routes. This one is contrasted with the centralised strategy, based on the master and slave idea. In this strategy a master determines routes and a slave without an associated master is unable to route its traffic.

These routing strategies produce routing protocols which are in principle different, but are similar as they use common strategies and try to achieve one or a combination of the following desirable properties [65]:

- robustness - algorithm needs to cope with networks changes (topology traffic changes)
- stability
- fairness —often a trade off with optimality
- optimality —often a trade off with fairness
- throughput
- load balancing

- congestion control

Each protocol chooses which of these properties it will focus on as some, like fairness and optimality, tend to be trade offs.

The Batmand routing protocol falls into the proactive category, but does not fit well into this grouping. This is because even though the Batmand protocol pre-builds its routing tables, much like the other protocols in the same group, the strategy it uses to maintain the route list makes it different to all others. The Batmand protocol only keeps a list of the one hop neighbours that serve as a path to the destination node. This is unlike most, if not all, other protocols in this group which maintain the full path to the destination.

The process of forwarding packets from one node to another through intermediary, nodes, otherwise known as routing, is only part of the process. To be more precise it is the final part of the entire process. In wireless networks there can be more than one path from source to a destination. Choosing the most appropriate path is called path selection.

2.4.3 Path Selection Metrics

Path selection is a process that selects one path from a list of possibilities, to be used by a packet to reach to its destination. This process involves applying a routing metric to multiple routes, which results in the selection (or prediction) of the best route to the destination. A metric is a property of a route, used by a routing algorithm to determine, from a list possible list of routes, the best choice to deliver a packet to its destination. A metric can include network information such as [15]:

- bandwidth
- network delay
- packet loss rate
- jitter
- throughput
- hop count
- path cost
- load

- reliability
- communication cost

Other metrics are built upon techniques that may include other metrics such as the ones mentioned above. The added sophistication of the metrics facilitates the selection of better paths as more network information is used to calculate the best route. All in all, these metrics serve to represent the link cost between two nodes in the network. Some of the most popular metrics used for WMNs are:

- Expected Transmission Count (ETX) [20, 47] —which is the loss rate of broadcast packets between pair of nodes.
- Round Trip Time (RTT) —This is the round trip delay between pair of nodes.
- Hop Count —This is the number of links between pair of nodes.

Continuing is presented more details discussions on the metrics mentioned above and other know metrics.

EXT is a calculated value between zero and one based on the ratio of forward and reverse transmission counts. The metrics overall goal is to choose routes with high end-to-end throughput. It is the predicted number of data transmissions required to send a packet over a specific link including retransmissions. The ETX of a route is the sum of the ETX for each link in the path [20]. ETX has a few short comings: it does not distinguish links with different bandwidths nor consider data packet sizes. This is overcome by the Expected Transmission Time (ETT) metric.

The RTT metric is based on measuring the round trip delay seen by unicast probes between neighbouring nodes [22]. The third metric is called Per-hop Packet Pair Delay (PktPair). This metric is based on measuring the delay between pair of back-to-back probes to a neighbouring node.

On a static network the ETX metric out performs the hop count and the RTT and PkPair metrics. The RTT metric gives the worst performance among the four metrics. This is due to the phenomenon of self-interference. The RTT metric uses far more paths per connection than other metrics and suffers from self-interference on all hops along the path. The PktPair metric performs better than RTT, but worse than bth hop count ETX also due to self-interference.

Interesting to note that in a mobile scenario hop count performs better as it reacts quickly to fast topology changes brought about by the dynamic environment; however, in a more static environment such as the one found in WMNs it performs poorly. A routing algorithm can select better paths by explicitly taking into account the quality of wireless links. Each of these metrics represents a different notion of what constitutes good link quality whether it is low loss rate, bandwidth, throughput, etc.

The IEEE 802.11s standardized metric is the Airtime Link Metric which is mandatory that all mesh stations must implement it to guarantee the full functioning of the mesh. It estimates the channel resource consumption as a function of the loss rate and link bandwidth.

2.4.4 The B.A.T.M.A.N Path Selection Metrics

Batman has its own built in metric it uses to select what the algorithm considers as being the best path. This metric is called transmission quality (TQ) and it is a route metric based on statics of the count of the received OGMs. In WMNs the selection of the best path involves selecting links that offer the best route to a node in the network. This best route is the one with the best metric values. The links are compared in terms of the number of originator messages that have been received (TQ) within the current sliding window. The sliding window is a fixed value that defines a range of the unique sequence numbers afforded to each OGM packet sent by an originator node. Note here is that TQ might suggest that Quality of service (QoS) is used but that in fact it is not. TQ is a simplistic metric and, as already mentioned, there are more sophisticated metrics composed of a combination of other metrics.

Furthermore, the addition of the delay metric, or any other metric, onto the Batman algorithm will result in a composite metric. This composite will be a mesh of the TQ and delay metric. Understanding how multiple metrics are combined to work together to produce one metric will add to the understanding of how to design the addition of delay onto Batman.

2.4.5 Delay

The modifications to the Batman algorithm are indented to improve the manner in which the algorithm conducts its route ranking and selection process. This improvement should result in paths that offer better communications links to be chosen. In order to improve the route ranking and thus the routes selected the TQ value has to be updated accordingly. Updating TQ affects the value used to rank the potential routes and thus affects the routes that are selected. These modifications are planned to be realised through the addition of routing metric known to affect communications in WiFi networks. One such metric is delay.

Delay is the time taken to transmit a packet from a source to a destination (one-way latency). The time taken to transmit and received the same packet, echoed back to the source, is called round trip delay. This value is measured in milliseconds (ms). The effects of delay on a caller in a VoIP network are generally perceived as echo and lag. Acceptable and unacceptable delay values, for voice applications were established by the International Telecommunication Union (ITU-T) in their G series recommendation[71]. According to the ITU-recommendation G. 114[69], delay values below 150 ms are considered acceptable. Values between 150 ms and 400 ms are acceptable, provided the callers are aware of the impairment. Lastly, values above 400 ms are deemed unacceptable.

Delay is affected by a few factors, which are referred to as delay sources, and are thus categorised. These categories are: delay at the source, delay at the receiver, and network delay[45]. Delay at the source, is the delay from the senders side, before the packet is sent. A cause of this type of delay is the VoIP codec, when processing analogue (actual voice) to digital (what is sent) conversion. Delay at the receiver, is the opposite of the sender's delay, which is the decoding and conversion process of the packet sent. Lastly we have the network delay. This is the delay experienced after the packet is sent, but before it is received. Sources of this delay are packet queuing, actual transmission time, and other components of network delay experienced in the wireless IP system. Some of the sources of delays are known while others are inconsistent. The focus is on the network delay, which is caused by network congestion, and leads to a slow delivery of packets.

Lastly, in networking, delay often implies the delay from source to a destination along a known path. This is calculated as the absolute value of the time the packet was sent subtracted by the time the same packet was received at the destination. However, this can not be so for a Batman network because, Batman nodes do not maintain full details of the route from source to destination which is the case with other algorithms that use delay as a metric. Note that Batman routing table is only aware of part of the route to the destination, and this part of the route is its best one hop neighbour through which the packet can be sent to the destination. Therefore, for Batman delay will be the time taken to send a packet to its one hop neighbours. Since the route used to send a packet may not be the same route used to receive a packet one way delay is enough. delay measurements require a clock synchronized clock shared by all the nodes but, Batman networks do not have such a clock as this is not part of the algorithm design. Since there is no network wide clock available on the Batman network, each node running any of the Batman implementations will have to use its own clock to both stamp and check the time on the OGM. Delay as understood here will be that of round trip time (RTT) which is the time taken for a OGM to leave the originator, travel along a specific route, to its one hop neighbour and come back to the originator.

This dissertation focuses on the network delay. This is the delay experienced by packets in the links between nodes, meaning the delay between the packet being released into the medium up to when it arrives at the destination node. This dissertation does not, however, focus on the delays at the sender and receivers.

The largest delays experienced along a transmission path could be as large the least stable component. This one hop delay is important and therefore is measure in 5.

2.4.6 Composite Metric

There are routing protocols that depend on a number of metrics. In such protocols each metric exerts their respective influences on the outcome of the routing decisions. The composite metric does exactly this; it combines the effects of the various metrics into a single routing value cost. This value is then used to compare possible paths to a destination. The outcome of the comparison is the decision on which option is the best path to a destination. Looking at examples of a composite metric will help clarify the idea.

The composite metric, Interior Gateway Routing Protocol (IGRP)[63], developed by Cisco, was designed and intended for use by network gateways interconnecting several networks. IGRP is a distance vector protocol meaning it exchanges a summary of the network information only with its adjacent neighbours. In the summary of the network information is the metric information used to attain the final composite metric value used in routing decision making. Specifically these metrics are:

- Topological delay - summation of packet and network delay
- Bandwidth of the slowest path segments
- Channel occupancy - load on the path
- Reliability of the path - essentially packet error rate

All of these are from the perspective of the gateway nodes; therefore, it is gateway to gateway information. One key benefit and also a feature of IGRP is that load balancing is performed by the routing protocol by splitting traffic among several paths whose metrics fall into a specified range.

Another composite metrics that aims at performing load balancing proposes a two-dimensional link-metric with hop-count or EXT as primary and the links Signal to Interference Noise Ratio (SINR) as secondary metric. Under such an assumption, the routing algorithm attempts to optimise the objective associated with the primary link-metric while using the secondary metric(s) only to exclude certain links. This results in links chosen based on the primary metric but within a threshold value established by the secondary metric. In this context the secondary metric was used to avoid using low quality and highly congested links offering the network better load balancing and higher throughput. The issue noted here is in deciding the threshold value for the secondary metric [57].

Some composite metrics seek to find a path with certain QoS characteristics so are designed to focus on such. One example of this type is the composite metrics described by [8]. They created a composite metric function consisting of bandwidth, delay and jitter as the authors realized that these are most critical to QoS support for multimedia applications. The work assumes an underlying mechanisms for gathering information about these metrics at each node. The objective of the composite metric is to find an optimal path with maximum available bandwidth and minimum delay and jitter for QoS routing in OLSR. The $compositemetric = K1 * (1/BW) + K2 * (J/D) + K3 * D$ where:

- K1,K2, and K3 are constants
- $BW = Available\ Bandwidth$
- $D = Measured\ Delay = Propagation + QueuingDelay$
- $J = jitter$

Lastly, other composite metrics are added as extensions to know routing protocols in order to enhance certain features in the route path selection process. One such example is the use of multiple metrics with the OLSR. [55] considers using EXT, Minimum Loss (ML), Minimum Delay (MD) producing what the authors called OLSR-MM (Multiple-Metric). The technique works by calculating the direct next hop links based on EXT and MD values then pruning the table to obtain the best 10 out going links with high EXT values. The following steps involve equations which normalize the metric values such that at the end each link is compared to each other link based on the resultant outcome values of the metrics used. Further, the relative importance of each metric is also compared with each other metrics and a weight adding up to one assigned to each metric. Lastly the out going link with the maximum total score is chosen as it provides the best overall quality enhancing OLSR's route selection.

2.5 Related Work

This section explores the relevant literature surrounding this work but will focus primarily on literature surrounding the Batman algorithm and its routing protocols. Two groups of work involving Batman exist: the first group carries out performance comparisons between Batman and other existing protocols; this group is the most predominant group. The second group, deals with all the work that has been conducted to try augment Batman in attempts to improve its performance in specific scenarios. The background on some of protocols more pertinent to the this sections.

2.5.1 Optimized Link State Routing (OLSR) protocol

It is an optimization version of a pure link state protocol. topological changes cause the flooding of the topological information. To reduce the possible overhead in the network protocol uses Multipoint Relays (MPR) by reducing the same broadcast in some regions in the network. There are two types of control messages, HELLO (finding hosts and link status), Topology control (TC- used to broadcast list of MPR selectors as well as own advertised neighbours). TC is broadcasted periodically by MPR hosts, only. The advantage is that it is best suited for dense networks where the source and destination keeps changing constantly. The disadvantages are that bandwidth is wasted on TC messages aggravated by increasing network size.

2.5.2 Ad hoc on Demand Distance Vector (AODV)

Routes are set up on demand, and only active routes are maintained. This reduces the routing overhead, but introduces some initial latency due to the on demand route setup. It uses a simple request-reply mechanism for route discovery. The advantages of this protocol is that it reacts quickly to the topology changes and is loop free and avoids count to infinity problem. However, it also has shortcomings such as, no routes are set up on demand, and only active routes are maintained. This reduces the routing overhead, but introduces some initial latency due to the on demand route setup. Also it is not suited for low powered devices. Furthermore packet delivery ratio drops dramatically when the number of connections increases.

2.5.3 Comparison Group

Johnson *et al.* [41] published the earliest work seen that conducts a performance comparison between Batman and the optimized link state routing (OLSR). The authors explain that Batman was created as a response to shortcomings observed in OLSR protocol, which includes routing loops and route instability due to the routing loops. A comparison is made to gauge the improvements attained by Batman over OLSR. The experiments were carried out in an indoor testbed composed of a grid of WiFi nodes and network data was collected and results analysed.

The results demonstrated that Batman scaled better, above 45 nodes, than OLSR, as less control packets flood the network. Batman achieved the best overall throughput, as well as the least delay, with the least number of hops. The average amount of time for a route change was half that of OLSR. The authors argue that this shows that Batman chooses better routes than OLSR. Batman out-performs OSLR on almost all performance metrics except packet loss; however, the percentage difference is not statistically relevant enough for a conclusion to be drawn.

Similar to [41], Ikeda *et al.* [34], 2008, considers Batman as the evolution of the Optimized Link State Routing (OLSR) protocol. This notion can be attributed to a story that in 2007 Corinna "Elektra" Aichele [3] developed the Batman algorithm out of frustration with the limitations of the OLSR protocol. Therefore, Batman can be thought of as the improvement on the shortcomings of the OLSR protocol. In their work, Ikeda *et al.* seek to evaluate Batman through experiments conducted in field tests in order to see if Batman solves the problems encountered on OLSR.

These experiments were conducted using laptops in an indoor testbed environment on their departmental floor. The results showed that Batman does not suffer from the same shortcomings found in the OLSR protocol, such as routing loops. However, Batman is still affected by throughput decrease with the number of hop traversed much like OLSR. This is explained by the authors as originating from self interference experienced in the network.

Other works exist where the OLSR and Batman performance are compared. Abolhasan *et al.* [1] not only compares Batman to OLSR but also includes the BABEL protocol in the comparisons, as it claims to provide a new and more effective approach to mesh networking, much like Batman. Their investigation focuses on the multi hop performance and the ability of each routing protocol to recover from link failures.

The experiments were conducted on an indoor multi hop mesh network testbed. The results show that Batman out performs BABEL and OLSR when it comes to route stability and packet loss. BABEL has the highest bandwidth and fastest recovery from link failures. OLSR was out-performed by both Batmand and BABEL in all tests conducted.

Work to similar Abolhasan *et al.* was later on conducted in 2010 by Murray et al. [56]. The work compares OLSR, Batman and BABEL but adds to the work by comparing OSI layer two and layer three Batman protocols, and attempting to establish the impact of routing on layer two over layer three on an indoor testbed.

The results of the comparison between OLSR, Batman and BABEL protocols are consistent with Abolhasan *et al.* so no contributions were made in that area of the study. The comparison between Batman layer two and Batmand three two did not yield conclusive results. The paper shows the need for further work into the performance difference between Batman layer two and Batman layer three. This is attempted by the works discussed next.

Barolli *et al.* [9] considered node mobility and carried out experiments that compare the two protocols considering the node mobility. In this work, the authors only compare an aspect of the protocols, namely, the Link Quality Window Size (LQWS) parameter of OLSR and Batman protocols. These experiments were carried out on a small testbed of five computers acting as nodes of a Mobile *Ad Hoc* Network (MANET). The authors compare the performance in their testbed in terms of throughput, round trip time, jitter and packet loss. The results obtained from their experiments revealed that OLSR parameter LQWS performed better if TCP data flow is used. Similar work and extensions to the is work were conducted by Ikeda on *et al.* [35, 50, 48, 49, 51].

Further comparisons are made by Garroppo *et al.* [27]. The authors compare the layer two implementation of Batman called Batman–adv and IEEE 802.11s through the reference implementation called open80211s. The evaluation is done in terms of route stability and route recovery time. Evaluation of the two layer two protocols was conducted on a testbed composed of four mesh nodes, which were laptop PCs.

The results reveal that Batman–adv has better route stability but open80211s has a faster recovery time. The authors also mention that Batman–adv has problems resuming communications after a sudden interruption; essentially it does not handle link failures well.

Closely related work was conducted by Wang *et al* [72]. The authors present a practical insight into a real-world performance of an open-source protocol based on the IEEE 802.11s draft specifications called open80211s. They do this by comparing the IEEE 802.11s OSI layer two protocol, implemented in open80211s, with Batmand and OLSR, both layer three routing protocols. The experiments are conducted on an indoor testbed, consisting of six mesh nodes and focused on the routing and forwarding capabilities of the open80211s while also looking at the stability and efficiency of open80211s.

The results obtained from the experiments show that open80211s does not perform as well as existing network layer routing protocols. This work is slightly different to other works that compare Batman to other protocols. This work uses Batman as an established routing protocol to help benchmark the performance of open80211s which is an upcoming protocol. This is similar to how OLSR was used to benchmark Batman in [41] when it was an upcoming protocol. Other comparisons are made in [25].
The experiments were conducted in an indoor testbed much like the one used in this dissertation. Difference is that this dissertation aims to use MPs and Unix boxes while the authors used portable wireless ad hoc Node (PWAN). Wang *et al.* set up two test scenarios to investigate the different performance aspects of each protocol. The iperfit network testing tool was used to collect the relevant network data such as bandwidth. Then fping (an extension of ping) was used to measure latency and round trip time. This work presents a similar approach to the one chosen to be used in this dissertations experimental study. The differences are in the data collected, the protocols tested and the general set up of the wireless testbed.

Lastly, Batman was developed by the need to over come the shortcomings of the OLSR protocol. The Comparisons conducted show that the designers accomplished this goal. It also demonstrates how to conduct comparison experiments with Batman protocols. This is useful for this work as a comparison between Batman and the version with delay added, will have to be conducted to show if the addition improves Batman's performance.

2.5.4 Augmentation Group

Kobo *et al.*[46], present a discussion on a study into optimizing Batman to handle quality aware routing on static WMNs with mobile nodes. The WMN with mobile nodes is in essence a hybrid mesh network and has greater topological changes. The author proposes to improve Batman's routing decision to be more effective in the rapidly changing mobile environment. The ultimate outcome wanted was to improve Batman's throughput in the hybrid network.

This work discusses how to extend Batman to work on a type of network that is more demanding than the type of network the original algorithm was intended to work on. The proposed extension is carried out by a first method: statistical situation monitoring by computing the standard deviation for the number of OGMs recorded. Then the variance of each link is evaluated and top ranked links chosen based on the statics. This approach extends Batman's already existing path selection metric (TQ). The second method: MAC layer situation monitoring by using received signal strength and signal to noise ratio from OSI layer 2 for each link.

The top ranked links are always considered for routing decisions. It is not clear how all the data collected will be used together to choose the links.

The author proposes to test the augmented Batman through simulations, as well as on an actual network. What is not clear is if the network will be an actual existing one or a testbed created to test the new algorithm. These works bear resemblance to our own, discussed in this dissertation.

The similarity is the extension of Batman to handle quality–aware routing on static WMNs, but without the added complexity of mobile nodes. The two works completely differ in the approach to add said extensions. The work in this dissertation focuses on adding metrics that directly influence voice quality on the MP network. Kobo *et al.*, on the other hand, focuses on metrics that influence network quality when mobility is involved.

Britton, M., *el al.* [11] seeks to improve Batman by adding their Batrytis hysteresis mechanism ,which reduces the prevalence of rapid routing changes, which for a variety of reasons can lead to application instability. The objective of their tests was to see the effect that hysteresis had on Batman.

The experiments conducted were done on both static indoor testbed and mobile outdoor testbed. Rapid changes in the routing table (route-flipping) where tested in the indoor testbed and throughput observed. The authors observe that rapid changes in the routing table (route-flipping) and the results show no obvious trend in throughput fluctuations.

Furthermore, this work focuses on augmenting the Batmand protocol and observing the subsequent performance improvements, this is something that is not evident in all the works presented in the sections above.

Lastly, the group of works presented here demonstrate that not much work has been done to improve Batman's performance through additions and modifications to its route ranking and path selection process. These works support the idea that an indoor testbed, set up to fit usage specific scenarios, works best when comparing Batman against its extended versions. This experimentation style is adopted in this work, further experimentation styles such as alternatives as well as similar styles are discussed next.

2.5.5 Experimentation Styles

Although different, the two groups of research described above have one thing in common, and that is the manner in which the experiments conducted were performed on an indoor testbed and others, although very few, through simulations. Existing works shows the benefits and drawbacks of these approaches, discussed here.

Ramanathan, R. and Hain, R.,[62] surveys the field of *ad hoc* routing and related real world testbeds. The author in this work argues that different *ad hoc* routing protocols need to be complemented with real-world experiments this view is also supported by [43]. Their reasoning is that real-world experiments need to be done in order to reveal real-world effects that may not be visible in simulation studies and also to gain practical experience. In our opinion, real-world experiments are necessary in the field of networks.

The authors explain that the accuracy of simulation studies are inherently tied to the models used to describe the physical and link layers. Furthermore, protocol implementations are usually geared to work well on the simulation environment and do not take the interaction with the ordinary protocol into account. Even though the benefits of physical testbeds are argued over simulators there are also drawbacks to physical testbeds.

Furthermore, the authors in [62] explain that the radio conditions are difficult to control, this makes it hard to reproduce the same network testbed set up, and thus also the data obtained in the results. This drawback is experienced in the majority of the studies that use physical testbeds. Ultimately this is the main drawback of testbeds, and is not easy to overcome. Other drawbacks, such as hardware availability, are considerably easier to overcome.

Lastly, in the authors findings they realised that *ad hoc* testbeds work best in small scale settings. Reasons for this are that, primarily, smaller testbeds are easier to control, and also the challenges presented by hardware availability are easier to overcome, as the equipment requirements are far less than those in larger testbeds. This is further supported by Kaba, J.T. and Raichle, D.R., who also present ways of circumventing the shortcomings of testbed usage.

Kaba, J.T. and Raichle, D.R.,[43] describes an operational *ad hoc* networking testbed that was developed to allow for the implementation, testing and study of *ad hoc* network routing protocols. The authors use techniques such as intentional attenuation of the signal level at each node in the testbed to enable some nodes to be out of range of others and thus creating multi-hop network topologies. The technique is an effective way of forcing a multi-hop network in an indoor setting where signal propagation is hard to control. This technique is also used in this dissertation, in its own indoor testbed described in chapter three. The use of signal attenuation techniques on testbeds offers a means to overcome the aforementioned shortcomings of testbeds usage [62] by enabling testbed users to create stable testing environments, where testing is deterministic and reproducible. The next work further supports the ideas discussed in here, and offers further insight into the drawbacks of physical testbeds.

Lundgren *et al.* [53] was motivated by the drawbacks of testbeds which cast a shadow on the added benefits of using testbeds to conducted experiments into wireless routing protocols. The main contribution of this paper is a formulation of an *ad hoc* experimentation and testbed methodology built upon their design of *Ad hoc* Protocol Evaluation (APE) which is, in many ways, similar to[58]. APE allowed the authors to perform evaluation of *ad hoc* protocols, where experiments are repeatable and results re-producible. APE runs on laptops and is based on the Linux operating system. It enables a post-experiment global view of the complete network connectivity at any instant, thus ensuring repeatability. The paper adds that testbeds are needed to complement modeling, simulation and emulation. In essence alone one is not enough but all three together give the best results.

Ultimately the use of of physical testbeds produces more benefits then the drawbacks it faces tarnishes its viability over simulations. The best way to seen that this statement is valid is through the observation of know real world testbeds. The first of which is the frankfurt.net meshed networks in Berlin (approximately 800 nodes), Leipzig (500 nodes) and Weimar (300 node) offer testbeds which offer free testing conditions for mesh-technologies including routing protocols. These testbeds (community meshes) have allowed for the testing, optimizing and extending of the Optimized Links State routing (OLSR) protocol. This network has grown over the years and has allowed new WMN routing protocols to be deployed on the testbed and its performance measured and benchmarked against that of the OLSR. The Batmand routing protocol was also tested on this community testbed.

Testbeds, both indoor , such as those found at university laboratories and outdoors such as those found connecting communities (Bo-Kaap, Cape Town, South Africa) serve as advantageous real world platforms in which to test, optimize and extend WMN technologies in usage specific scenarios.

2.5.6 Preliminary tests and results

The indoor experimentation styles discussed in the sections above were adopted and reported in different stages on Chissungo, *et al.*[16, 17, 18] and also in[19]. These works do not correspond

to chapters in this dissertation, that style of thesis writing was not adopted here. The indoor testbed used by these works are similar to the ones adopted in this dissertation. This dissertation adds to these works by providing the comparison between Batmand and its delay extended version.
The wireless signal of each node was reduced and obstructions placed around them in order to reduce the signal propagation forcing the need for intermediate nodes for communication between none neighbouring nodes. This resulted in a multihop store and forward network environment where the Batmand was tested and compared to other protocols.
The results obtained showed that Batmand performs well in an indoor testbed laboratory study and may serve well for communications in rural environments. This is discussed in more detail in Chapter 3 of this dissertation.

2.6 Conclusion

The background chapter has provided a deeper understanding of the artefacts and surrounding ideas which together form the background to this dissertation. WMNs along with WiFI enabled devices are shown to meet the requirements set for a cost effective alternative to wired technology. The MP and the Batmand routing protocol are addressed as viable options for rural areas. Previous work conducted on Batmand and other implementations show that the protocol performs well in indoor testbeds and also performs well when compared to existing routing protocols.

Although reading through this chapter shows a plethora of research avenues, this dissertation proposes to better the Batmand protocol through new routing metric that augments the Batmand protocol and improves its performance. This work will follow similar ideas established by previous works that attempted to augment Batmand. This is an endeavour to try to meet the QoS requirements found in rural settings. The addition of delay metric onto the Batman algorithm is explored and shows that, theoretically, this task is possible. However it also shows that there are draw backs on the frequency delay values will be update on the final composite metric which is composed of both TQ and delay.

In order for the goals set by the research questions to be met the design and testing of the routing path selection scheme method is necessary. However, first how to test the final system is examined.

Chapter 3

Test Design

The following sections discuss the design of the tests and experiments that will be carried out to determine the success of the final system. The tests described here have been used to test similar systems (see Chapter 2). Preliminary tests and results were described in Chissungo, *et al.*[16, 17, 18] and also in[19].

The idea behind the tests is to compare the modified system with the original one. These tests are experiments on real equipment and WiFi networks: our testbed. We first review the research question that led to these tests.

3.1 Research Questions Review

Research questions tackled in this dissertation are:

1. How does B.A.T.M.A.N perform in a laboratory setup mimicking aspects rural networks?
2. Is it necessary to develop a different route selection metric, or even a different routing protocol, to provide better QoS in the network?
3. How are different applications, such as, voice and data supported on the Batman Network?

A similar approach can be taken to investigate these diverse questions. The approach involves testing the final system, built as part of this project, and comparing it to the original unmodified system. The same testbed is used to answer all the questions. This testbed is motivated and described in the remainder of this chapter.

3.2 More on the Research Questions

In order to answer the research questions in this dissertation, the project requires data to provide the necessary insight. This data would, in essence, be used to compare the original Batmand routing protocol with a modified version of this original protocol. This means that two sets of data, obtained from testing both protocols, will be captured. The results obtained from the comparison would then offer the required insight needed to answer the research questions. An overview on the research questions is provided next, however, more details is are given on these question in Chapter 6.

The first question asks: How does B.A.T.M.A.N perform in a laboratory setup mimicking aspects rural networks? This question determines the viability of the protocol in its intended use. There would be no use in proceeding with researching improvements for this protocol if it cannot be used as part of the viable solution for rural networks as seen in laboratory set up which mimics characteristics of those areas. B.A.T.M.A.N algorithm being a viable alternative is one of the compelling arguments of this dissertation. To achieve this, the characteristics of rural environments, and the influence this puts on rural networks, is looked at later.

The second question asks: Is it necessary to develop a different route selection metric, or even a different routing protocol, to provide better QoS in the network? After having proven that the B.A.T.M.A.N algorithm performs adequately in the laboratory set up, the dissertation proceeds with its main objective: to research possible changes that can be made to the existing B.A.T.M.A.N algorithm in order to improve its performance, as seen from the data obtained from the tests conducted for the first research question. Reasoning behind the need for the changes is explained in Chapter 1, and that reasoning is that B.A.T.M.A.N is simple in its approach to determining the best links in the network to be used in forwarding network traffic. This dissertation argues that the use of more metrics should yield better quality of service (QoS) in the network. The changes made to the existing B.A.T.M.A.N algorithm will be performed on its Batman daemon (Batmand) implementation protocol. This will create a modified version of the same protocol. Tests conducted here, are in the same manner as those conducted for the first research question. Similar results to the ones obtained for the first research question will be obtained and compared against those of the first research question. With these comparisons in hand, an answer for this second research question can be reached.

The third research question asks: How are different applications, such as, voice and data supported on the Batman Network? This question aims to see how well services on the MP network are supported it is known that the MP supports voice over IP (VoIP) and can also support data. Answering this question will provide value by other potential uses of the MP network aside from providing VoIP service. In order to obtain the data needed to answer this question, it is ideal to weave the required tests to answer this question in with the tests that answer the first two questions. This means the experiments conducted will not only yield data for the original Batmand protocol and the modified version but also provide data to answer this question.

Lastly, this section has shown how each research question ties into the core motivations and arguments of this dissertation. Furthermore, it has revealed how the data will be used and how the experiments will be conducted. There will be one experiment which will be repeated by the modified version. Each experiment will also determine the types of services that can run on the MP network. Next, the characteristics of a rural area are explored. In doing so, the features of the network on which the experiments will be conducted, will be revealed.

3.3 Characteristics of Rural Areas and Networks

The abstracted characteristics of rural areas come from the definition of a rural area as stated on Chapter 2, but primarily from our own observations conducted during field work at Mdumbi. Mdumbi is in the Mankhosi region of the Eastern Cape, South Africa, and is used as a case study here.

The abstracted characteristics of the rural area at Mdumbi are as follows:

- Areas with low population density,
- With individuals sparsely distributed over the region, meaning a few will be close but the majority will be separated by long distances.
- Houses situated on top of hills, which in many cases does not offer direct line of sight as one hill will be shorter than its adjacent neighbours.
- Poor or lack of telecommunications infrastructure
- The lack of availability of electricity.

The following images of the Mdumbi region depict the features listed above. Figure 3.1 shows a typical rural community at Mdumbi. Distance: Each grouping, marked by overlapping the red boxes, is of those individuals within close proximity of each other (roughly 10 to 30 meters apart). Some might even belong to just one family. Focus is given to this community scenario shown, as these are the typical settings. In this figure, attention is drawn to the distances between groupings of huts, and the potential hops data in an MP network would have to traverse. In the figure we identify seven groupings of huts (each numbered). Each is separated by distances ranging from roughly 600 meters (between groupings one and two) to 40 meters (groupings four and five). The blue lines linking the groups show the potential hop data would have to traverse. If the entire community were to be connected with only the MPs, the number of hops would peak at six (connection between group one and seven). However, this kind of scenario would result in deteriorated voice quality between the edge groupings. In this scenario this would be experienced by the communications between group one and groups seven, six, five, four and potentially three. This image also shows the hilly geography of this region as the connection between one and two is over a hill and so are the connections between two and three, four and five and so forth (although harder to see). In this image it is clear that there is no line of sight between one and clusters three, fours, etc. Also Clusters five and six might not have line of sight to two all due to the hills, this makes clusters 2 and three critical for network wide communication.

In extreme circumstances some communities are made up of completely dispersed huts such as the one shown on Figure 3.2 where families are separated from other families by distances exceeding 600 metres. This set up is not as common.

3.4 Conducting experiments

There are three options that could have been chosen to conduct the necessary experiments. These are: field experiments with a deployed rural network, using a simulator to mimic the characteristics of rural environments and using an indoor physical testbed. The first approach would be the most favourable but, it is also the least tangible. The first reason being the pressing issue of powering the devices, remember that a mentioned characteristic of rural areas is that not all the inhabitants have access to electricity. Furthermore, getting the community involved in such a project requires the approval of the ”Head Man” of the area. The devices would have to be housed outside the huts where they are vulnerable to theft. Offering security to all the devices would prove a challenge. Finally, the time it would take to get around all these hurdles would be too long. This project is bound by the strict time line that binds all Masters dissertations. Therefore, field deployment in Mdumbi would pose risks to the projects successful outcome. These challenges however, can be circumvented by the remaining two options. The discussion on which of the two options, simulations or physical testbed, are covered in 2. The section concluded that physical testbeds are a better choice for this project.

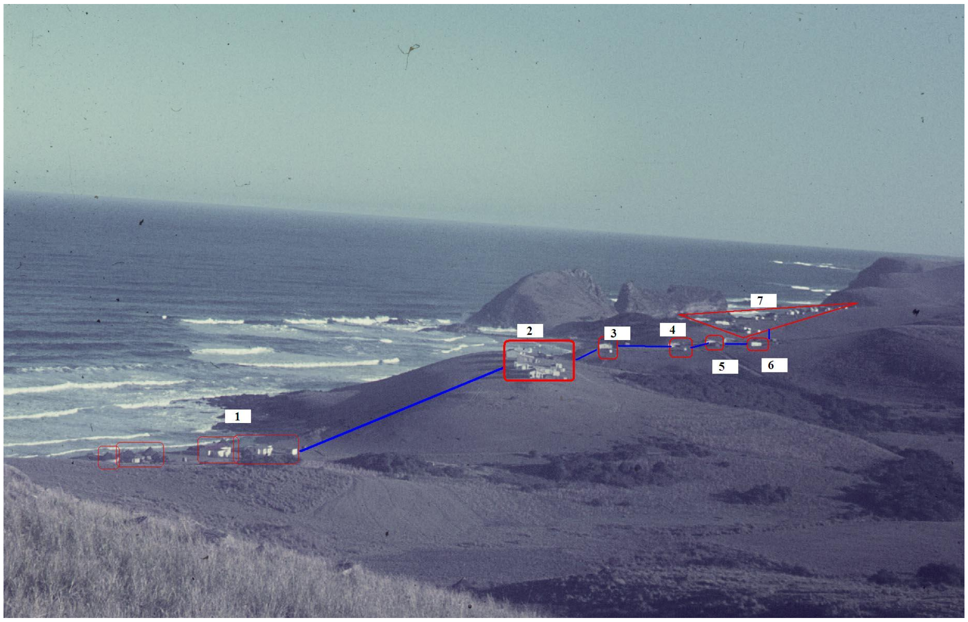


Figure 3.1: Picture shows a common Mdumbi community setting. Each number represent a grouping of huts within proximity of each other (5 to 10 meters), marked by a red box or triangle.

Figure 3.2: Picture shows huts in the Mdumbi region, in a dispersed community set up.

Experimentation will therefore be conducted on an indoor physical testbed. This testbed will be constructed to mimic certain the characteristics of Mdumbi as described in the sections above.

3.4.1 Similar Experiments

Research, detailed in the background chapter (see Chapter 2),), has shown that similar experiments were carried out using either the Batmand protocol or the Mesh Potato (MP) and were performed in the following fashion: those interested in the protocols performance, tested it on a testbed composed of Unix machines. The metrics used to gauge Batmands performance where based on the same metrics used to compare, any specific, existing routing protocols with Batmand [41, 34, 1]. Those interested in the MP, such as the village telco group, stress tested the device. They based their results on the number of phone calls and audio quality observed in their tests, essentially used those qualities as the metrics.

This project has opted to test the modified version of Batmand on Unix boxes first and if performance improvements are seen then further experimentations will take place on MP devices. Running the experiments on either of the two systems (MP Unix machines) would result in conclusive results, as it has done so in other works. The procedures for conducting the experiments on the Unix machines, are simpler (similar to installing and running any other software) than those for executing them on the MP. The procedures for conducting the experiment on the MPs were: the MPs were reflashed, with the original Batmand protocol used by this project to create the new system. The reflashing and testing of each protocol took place separately. Once reflashed, the separate systems were tested in the same manner. This is described in the next sections.

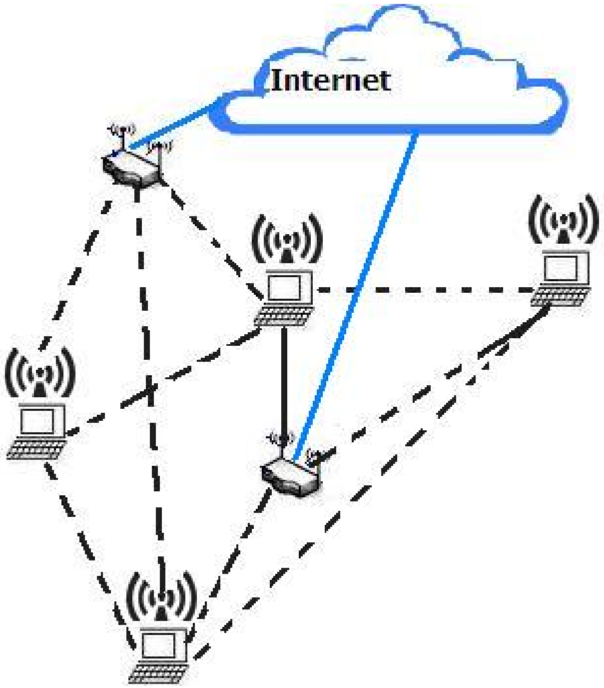


Figure 3.3: Image shows a typical urban mesh network set up with interconnected links.

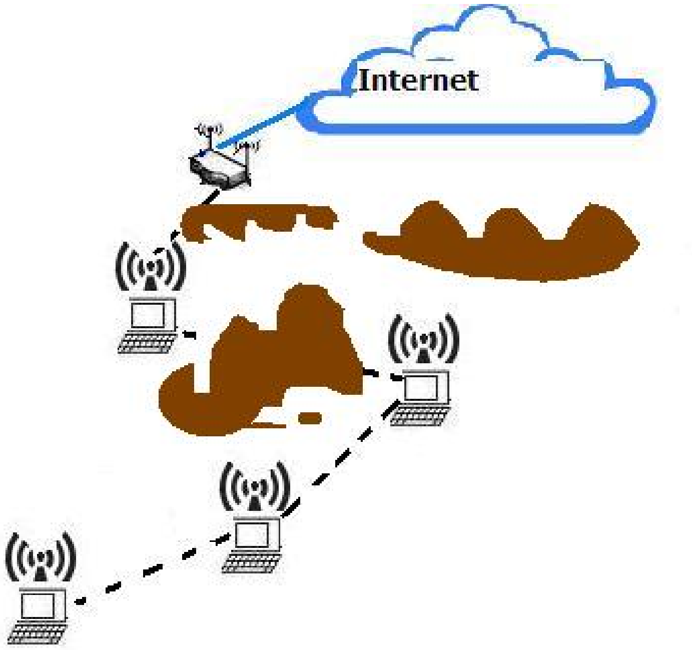


Figure 3.4: Image shows a typical rural mesh network set up with single connected links.

3.5 General description

In urban areas a mesh network is likely of have interconnected nodes offering multiple routes in the network as shown on Figure 3.3. In the urban set up distances between the nodes, and therefore the links, are often a few tens of meters as individuals using the mesh in urban areas are within close proximity. In cases where the mesh may offer Internet service, distance from the access point (AP) or gateway is also short and at times there are more then one APs/gateways. However, this is not true in rural areas sharing similar characteristics as those found at the Mdumbi region. The topological characteristics of the Mdumbi region create a multihop network environment similar to the image shown on Figure 3.4. networks communications are over multihop links as the distances between neighbouring nodes are much larger(tens to hundreds of meters) then the urban areas. No line of sight is also common between most of the neighbours as sight is obstructed by the hilly surfaces. In these types of networks there is often one AP/gateway, if available, and thus multiple points of failures as many links become critical and have no alternative paths. This is the type of topology the dissertation abstracted and shall mimic in the laboratory testbed described in chapter 4.

3.6 Summary and conclusion

This chapter has presented typical characteristics found in rural areas as observed from fieldwork conducted at Mdumbi. These topological and multihop features were then abstracted and mimicked in a laboratory set up which resulted in our testbed. The testbed however, does not aim to recreate a detailed representation of the topological and environmental characteristics present at Mdumbi. The chapter also addressed the experimentation features needed in the final tests in order to obtain the data needed to answer the research questions. Presented next is the design and implementation of the augment Batmand routing protocol.

Chapter 4

System Design

The system design chapter discusses the design, and implementation of the system that was motivated in Chapter 1 and described in Chapter 2 of this thesis. In Chapter 1 the MP was proposed as an alternative telecommunication device that may serve as a solution to the problem of high costs of telecommunication, by serving as a cheap alternative. The MP uses the Better Approach to Mobile *ad hoc* Networking (Batman) algorithm's daemon implementation called Batmand(daemon). As it often the case with new technologies, there is room for improvement as the technology grows and matures.

The main focus of this project is to provide better telecommunication in rural areas, through network delay aware quality of service (QoS) improvements on the Batmand network. This goal is pursued through a laboratory study of improvements made to the existing Batmand routing protocol. These improvements are conducted as extensions made to Batmand's route ranking and path selection process. This should, in principle, increase stability and reliability in the Batmand network, making the Batmand routing protocol more effective in rural areas.

In this dissertation the abstracted characteristics of the Mdumbi region and its individuals are taken as a case study for which the extensions to Batmand are designed for in a laboratory environment. The characteristics of these rural environments are described in Chapter 1 and Chapter 2 as:

- Areas with low population density,
- With individuals sparsely distributed over the region
- Houses are huts and are located on top of hills; some of these hills are much lower (distance from ocean level) or higher than their adjacent neighbours, which greatly affects the line of sight.

- Poor or lack of telecommunications infrastructure.
- A lack of security for any technology placed there.
- Lack of electricity, households in these regions often do not have access to electricity

The extension to Batmand will add a metric that affects wireless mesh networks (WMN) in rural environments matching the list of characteristics mentioned above. These metrics are:

- Long distances between houses, thus packets will have to hop over great distances.
- No line of sight between neighboring huts due to hilly surfaces.

Next the sections discuss what ideally the project would want to change on Batmand and then will show the things that can be changed and how the Batmand protocol was changed.

4.1 Ideal Design of the Batmand Extension

The final system, should implement the delay metric, onto the existing version of the Batmand routing protocol. This addition will extend the way Batmand ranks the routes it finds during route discovery and also, how it chooses the best path for routing voice and data traffic through the network. Research on this was detailed in Chapters 2 and 3.

Ideally, the originator message (OGM) packet is modified to include a time stamp which then is used to calculate the delay on neighbouring link. This would happen at the same time as the Batmand's, default, transmission quality (TQ) metric also gathered network link information. The combination of the network link information obtained from both protocols would yield one composite metric. The value obtained from the composite metric would then be used to rank the links and then choose a path. This process would repeat for every OGM received by all the nodes in the network, resulting in a Batmand that is aware of the delay on network links.

However, as was discussed in Chapter 2, this is not possible. The problem arises from the frequency in which TQ and delay values are updated. In the algorithm TQ is updated on when OGM packets are received which happens on four possible scenarios here listed:

1. The OGMs source is not in its routing table.
2. The originator is in the routing table and the node that forwarded the OGM is new.
3. The originator is in the routing table, and the sender is not new.
4. The received OGM is its own.

4.2 Batmand Mesh Network

The Batmand network has no synchronized clock, therefore, for delay calculations the same clock has to be used. Delay is then understood as round trip time (RTT) which is the time taken for a OGM to leave the originator (where it receives a time stamp), travel along a specific route, to its one hop neighbour and return to the originator (where the time stamp is checked). This can only happen on one of the four scenarios presented above, the fourth scenario. Link Delay can only be measured on the fourth scenario because this is the only scenario where the OGM sent is also the OGM received allowing the same clock that stamped the OGM to be used when the same OGM is received.

End to end delay breaks the design paradigm of the Batmand algorithm that design steers away from conventional routing algorithms that seek end to end network information. Batmand implementation and our Delay implementation adheres to this design principle which only seeks one hope link network information. This feature of Batman algorithm makes Batmand the protocol it is. Further, end to end OGMM Delay would return stale network information. As the OGM would traverse an entire network path which cannot be controlled by the OGM as these are broadcasted, so end to end Delay values would be for random paths. The problem with this is that this results r in TQ being updated more frequently than delay potentially affecting the route ranking value for the cases where delay can not be calculated but TQ can and is. Creating route fluctuations as the composite metric will fluctuate between TQ weighted values and TQ and delay weight values. The ideal solution would be to stop TQ updates on the other three scenarios, however, doing this would make Batmand extremely slow at obtaining network paths and also slow in responding to route changes. Lastly, if this design is taken forward it violates the Batman algorithm design as it imposes too many changes to Batmand.

The next section discusses the design of the changed protocol.

4.3 Scope and Aim of Design

The Batmand protocol implemented from the Batman algorithm used here as the original system on which the modification will be made will be referred to as the original Batmand protocol (O-Batmand). O-Batmand is the fresh install which ships with the mesh potato (MP) devices. The Batmand protocol that will result from the implementation of the delay metric will be referred to as the modified Batmand routing protocol (M-Batmand). The design of the system, to be implemented, will include details on how the chosen metric and the already existing metric, in the O-Batmand routing protocol, will interact. This interaction will output information that the M-Batmand routing protocol can then use, to route packets through the network.

Ultimately, the goal of the implementation of the design, described in this chapter, is to create a M-Batmand protocol that out performs the O-Batmand protocol. Performance is understood by the interpretation of the data collected from the experiments described in Chapter 3.

4.4 System Architecture

In order to achieve our design goals, a series of artifacts are required. The components are presented here along with further insight.

4.4.1 O-Batmand Routing Protocol

O-Batmand has many features that were added on top of the Batman algorithm. One of these features is a gateway node notification system. This means that O-Batmand nodes that have connections to other networks, such as the Internet, can announce to the MP network that they have that service. This project acknowledges the existence of such features, but will, however, not focus on them at any given point, as this is beyond the scope of this project.

4.4.2 O-Batmand General Program Flow

This section provides an overview, of the essential parts of the O-Batman protocols inner workings. Understanding the program flow enhances the understanding of the protocol, as the implementation and algorithms differ from each other. Some of these differences are due to the language used to implement the algorithm, and the constraints imposed by the resources on the device. The O-Batmand code was written in the C programming language, and then highly optimized to work efficiently within the resource constraints of the MP listed on Chapter 2.

When O-Batmand launches, it starts off by initially setting the default values for the OGM packets. These packets are the ones broadcasted by each node in the network in order to find network routes and build a routing table.

- TQ metric value is set to 255.
- Time to live (ttl) set to zero.
- OGM sequence number set to zero.

Other values are included in the implementation but not the algorithm. Two examples are, first: the gateway flag, used to mark the nodes that offer advanced services, such as an Internet Connection Bridge. The second is the protocol version number, used to mark the O-Batmand protocol version to which the OGM packet belongs.

O-Batmand drops packets that have a different version number to the current version of O-Batmand running on the device. A reason for this is to avoid clashes between different versions of O-Batmand, as code changes from one version of O-Batmand to another. With the initial values set in the packet, the program proceeds.

Having all the packet variables set the OGM packet is then added to a list that holds packets to be forwarded. This is done through the packet scheduling functions. Packets are then sent at a specific rate, the default rate is one every second as shown on Figure 4.1. The first network OGM packet is then received. Once received the OGM packet goes through a series of tests. The receiving node checks to see if:

- The last sender of this packet was itself.
- The node originated from itself.
- The version of the packet is the wrong one.
- It is a duplicate packet.

If the outcome of any of these checks is true, then the packet is dropped.

Achieving the principal goal of this project will require the integration of the O-Batmand metric, TQ, with the chosen metric. Understanding how the O-Batmand metric, TQ, was implemented is necessary and is discussed next.

4.4.3 O-Batmand Metric Integration

In the O-Batmand code, the TQ value is calculated by all nodes for each link all nodes, in this case one node, shares with its neighbouring nodes. The node in question calculates the TQ value by averaging the number of all the OGMs it received, from a neighbouring node, whose originator is some other node (the originator—source of the packet) in the network. There is one constraint and that is that all of these OGMs, from a specific originator, have to fall within the global window size. This is set to a default value of five. This is all implemented inside the O-Batmand C-file *ring_buffer.c*. Default values are defined as macros in the O-Batmand header file *batman.h*. TQ values are stored inside the neighbour node structure (*struct*), along with other details such as ttl, incoming interface details, and the originator node details such as its ID.

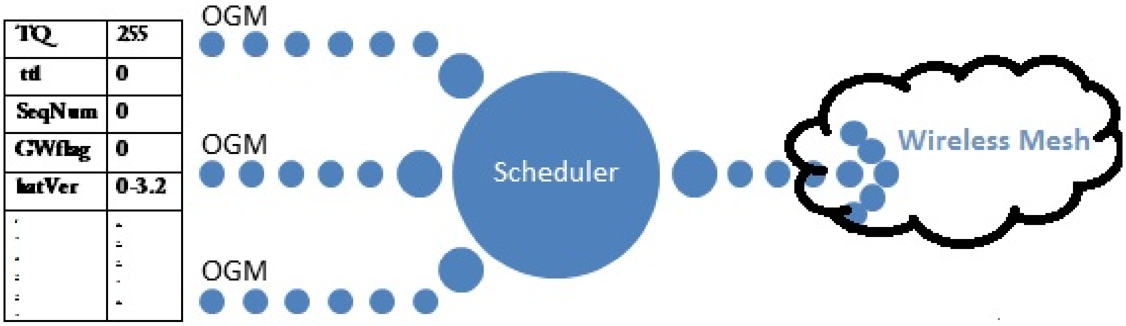


Figure 4.1: Shows the blue print of the testbed with nodes and links shown by dotted lines. This testbed is the experimental equivalent to the characteristics observed on.

4.5 M-Batmand Routing Protocol

The design shown next is the best way found, through research, as the approach to adding the delay metric onto the O-Batmand protocol to create the M-Batmand protocol.

Delay measurements are conducted during the bi-directionality tests. Once the delay calculations are completed, the additional details can be added to the ongoing TQ average for the specific link. This addition will create a new TQ value for the O-Batmand protocol, one that is delay aware. The O-Batmand that uses TQ values that are delay aware is referred to as M-Batmand.

The delay metric is merged with O-Batmand is this fashion:

1. When OGM packet is first created it is time stamped and that value recorded inside the packet as it is being sent.
2. When the M-Batmand receives is its own packet the protocol performs the bidirectional tests to see if the link between itself and the neighbour that forwarded the packet is can be used to send and receive packets thus bidirectional. At this point the link delay is calculated as the packet sent to this specific neighbour is time-stamped and when it is received, then that time is compared to the received time. This is done between one-hop neighbours as to adhere to the Batman algorithm design paradigm.

3. It then performs test to see if the OGM packets sequence number is within the global window size limits (removes stale/old packets)
4. It calculates the difference between the current time stamp and the one found in the OGM packet, this difference is the current delay.
5. It increments the total number of received packets from a neighbouring node by one, this is how TQ is updated.
6. Add current delay to the running average for this link and divide by the total number of received nodes from the specific neighbour, thus obtaining a current delay average.

How the network delay information is added, to the existing average TQ value, is detailed next.

4.5.1 Delay Assisted Transmission Quality (TQ)

Delay assisted TQ, will be the outcome of the combination of the delay and TQ metrics. This will be done in a similar manner to that of the composite metric discussed in Chapter 2. The primary advantage of using a composite metric is observed from understanding the short comings of using a single metric. When a single metric is used, all sources of network performance issues are treated as that single metric, which is used to measure and compare the performance of the links. To better explain this: assume an example where delay is the metric used for the network. If there is a link with low bandwidth, it is normally represented by a large delay. However, bandwidth limitations, as well as other metric limitations, do not accumulate in the same way delay does. Therefore, treating each metric separately means each can be handled correctly. Using more than one metric, such as the composite metric, with the appropriate weights added to each chosen metric can result in the best path being chosen.

When modifying the O-Batmand protocol, it is essential to keep in mind that excessive modifications can result in an outcome where the resultant M-Batmand, is no longer comparable to the O-Batmand. The modifications can lead to a M-Batmand that is no longer similar enough, to the original, resulting in a state where it is not clear whether M-Batman can be considered a Batman protocol. In order to avoid this, the design will affect the structure of the protocol as little as possible.

The combination of the TQ and delay values achieve similar effects to that of the IGRP protocol discussed in Chapter 2. To accomplish the combination of the two metrics this we have to transform the delay value to a value that is comparable to TQ. This means we have to transform delay values to values between zero and maximum TQ, $MaxTQ$, which is

255. This means delay values to values between zero and maximum TQ, Max_{TQ}, which is 255. This means delay values have to be normalized $Normalized_{Delay} \in \{0, 1\}$ making it a fraction between zero and one: $Normalized_{Delay} = Network_{Delay}/Max_{AllowedNetworkDelay}$, we then multiply this value by the Max_{TQ}: $N_{Delay} = Normalized_{Delay} * Max_{TQ}$. The resultant $N_{Delay} \in \{0, 255\}$. Note that the maximum allowed $Max_{AllowedNetworkDelay}$ is 400 ms[69]. Once we have this we can proceed to add its delay effects to TQ.

The initial idea was to add the effects of the delay by adding in the N_{Delay} value to that of TQ. However, this would result in a value greater than the Max_{TQ} and even up to twice the Max_{TQ}. This would lead to further modifications of O-Batmand. The modifications would enable O-Batmand to know how to handle the new, larger, TQ value. This approach would result in excessive modifications of the O-Batmand protocol, and as explained it is something unwanted. One way around this is to transform the result into a value between zero and Max_{TQ}. Another more appropriate solution for this is the following.

Delay is detrimental to QoS to the network; the effects therefore, have a negative presence on the paths. Added to this, is that the best and worst TQ values, as understood by O-Batmand, are 255 and zero respectively. From this it can be seen that delay effects should decrease the TQ value as lower values of TQ represent less favourable route. This prompts the need for a delay coefficient that inverts the value of delay taking it from a positive value to negative. (-1) does this and also maintains the absolute value calculated from the medium, therefore, not skewing or weighing the delay values, making it an appropriate coefficient for this solution. Therefore, resultant formula for Delay assisted Transmission Quality (DATQ) should be the following:

$DATQ = TQ + (-1 * N_{Delay})$

Lastly checks have to be added to ensure that DATQ is never less than zero. The checks will also ensure that any delay value from the network greater than $Max_{AllowedNetworkDelay}$ it is set to the $Max_{AllowedNetworkDelay}$. In order to assist the comparison of the two protocols the original TQ value before it becomes DATQ will be stored.

4.6 Theoretical Analysis

The implications of these modifications to the O-Batmand routing protocol have been aforementioned and are here elaborated on.

Since delay values are calculated once for every four of TQ. It is possible that the composite

DATQ metric, which is the combination of delay and TQ values, does not perform as expected. DATQ has TQ values being updated more frequently than delay values. This could affect the route ranking value, DATQ, for the cases where delay can not be calculated but TQ is. This could affect the route ranking value, DATQ, for the cases where delay can not be calculated but TQ is. This could result in route fluctuations as DATQ values will move between routes ranked solely by TQ and routes ranked by TQ and delay. This occurs on the three of the four scenarios where packets are received by the nodes but delay values are not available.

In order to make sense of the meaning of this problem take into consideration the scenario depicted on Figure 4.2. In this scenario the red blocks are nodes in a M-Batmand- routed network. The blue lines between nodes show the existing links between nodes. Next to, below or above these links are the TQ and delay (D) values for that link. In this scenario the start node (S) would like to communicate with the destination node (D). In order to establish this communication it has to hop through either nodes A of B. Ideally, in node S's routing table the one one-hop neighbour best suited to send the packets should be link S to A. Even though it has a small TQ value, the negative effects of the delay are less felt on the S to A link then on link S to B, where delay is high at 150ms. However, due to the fluctuations the routes will fluctuate between link S to A, where DATQ is in use of TQ and delay information, and S to B where DATQ is primarily influenced by the high TQ value and none of delay values.

This fluctuation causes network routing issues as route stability drops considerably. This could further stress the network and increase the processing needs of the node reducing the performance of the network. Larger networks would suffer most with this problem.

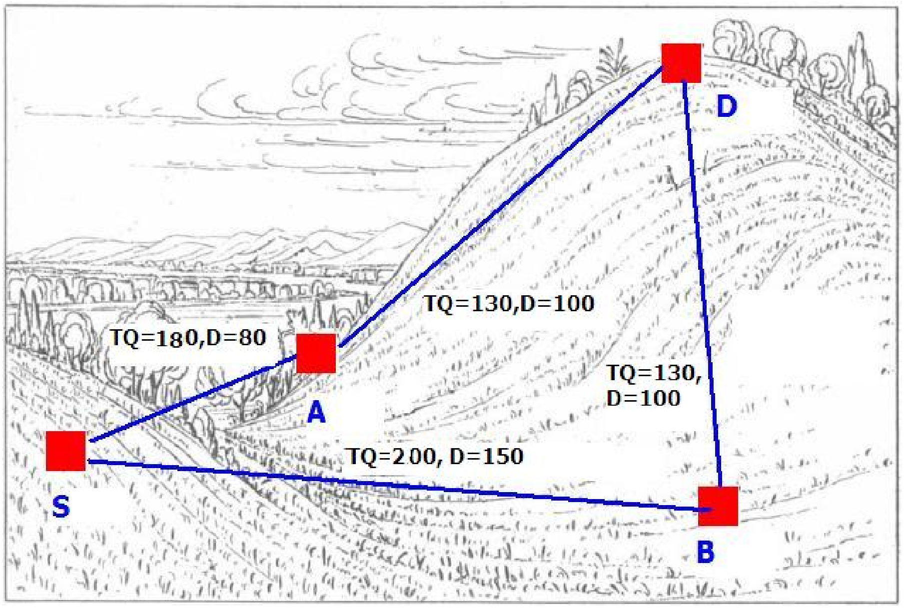


Figure 4.2: Image showing M-Batmand issues on a multiple route choice scenario. The blue lines represent links between the nodes (Red blocks). TQ and delay (D) of each link is presented next to each link.

Having the augmented Batmand designed and implemented discussed next is how and were it is tested.

4.7 Scope and Aim of Test Design

The approach used by this project is to construct a laboratory testbed, which mimics the abstracted characteristics of the topology observed at Mdumbi. The testbed fixates on aspects such as the multihop nature of the nature of the natural topology seen at Mdumbi as well as the lack of direct line of sight between houses as these are spread out. This testbed is therefore not a detailed representation of the abstracted topology. This sections and the ones following describe the manner in which this task was approached and achieved.

These characteristics where afore listed and serve as the basis for the testbed construction.The decisions made in design of the testbed serve to create laboratory scenarios, in the testbed, that closely resemble Mdumbi meshes. In doing this, the work directly answers the research question: How does B.A.T.M.A.N perform in a laboratory setup mimicking aspects rural networks?

In the rural areas, the distance between huts and the elevations (hills) on which the huts are built causes certain nodes on the rural network to be out of communication range of its neighbours, this is shown by Figure 4.3, which shows a map of varying elevations in the Mdumbi region, through topographical contour lines. On the image, the closer the contour lines are to each other, the quicker the surface changes elevation. The further apart, the fatter (slow elevation change) the area is. The Mdumbi area has both; showing areas of high elevations around those of lower, making it a hilly area. An indoor laboratory testbed would have to mimic these topological characteristic, as testing in this scenario provides data that reflects a real world scenario.

The problem with this is that indoor testbeds often suffer from node proximity issues, created by the limited space offered by the location used for the testbeds and aggravated by the nodes network cards which reach greater distances than indoor spaces often offer. To circumvent this, this work adopts techniques described by P.Gunningberg, *et al.*,[58] and B. Hagelstein, *et al.*,[72]. The authors use techniques such as, intentional attenuation of the radio signal strength at each node in the testbed. Purposely attenuating node communication signals directly forces each nodes communication range. In turn, this forces pairs of nodes out of range of each other thus creating a multi-hop network topology. The achievable numbers of hops, on the testbed, are limited by the amount by which signal from each MP can be reduced, without impacting its capability to communicate with other nodes in the network.

Intentional signal attenuation can be achieved by directly reducing the network cards output power. In doing this, the testbed mimics the effects rural characteristics have on the wireless network. In the case of the MP, accessing this functionality is difficult as the feature is not included with the firmware. Cases like this require that the devices be strategically placed in locations and surroundings that directly limit the amount of radio signal emanating from each MP to the rest of the medium. Examples of surroundings that can reduce the signal strength of WiFi devices, are thick walls and metal shielding, amongst other materials. The design decisions of the testbed are discussed next.

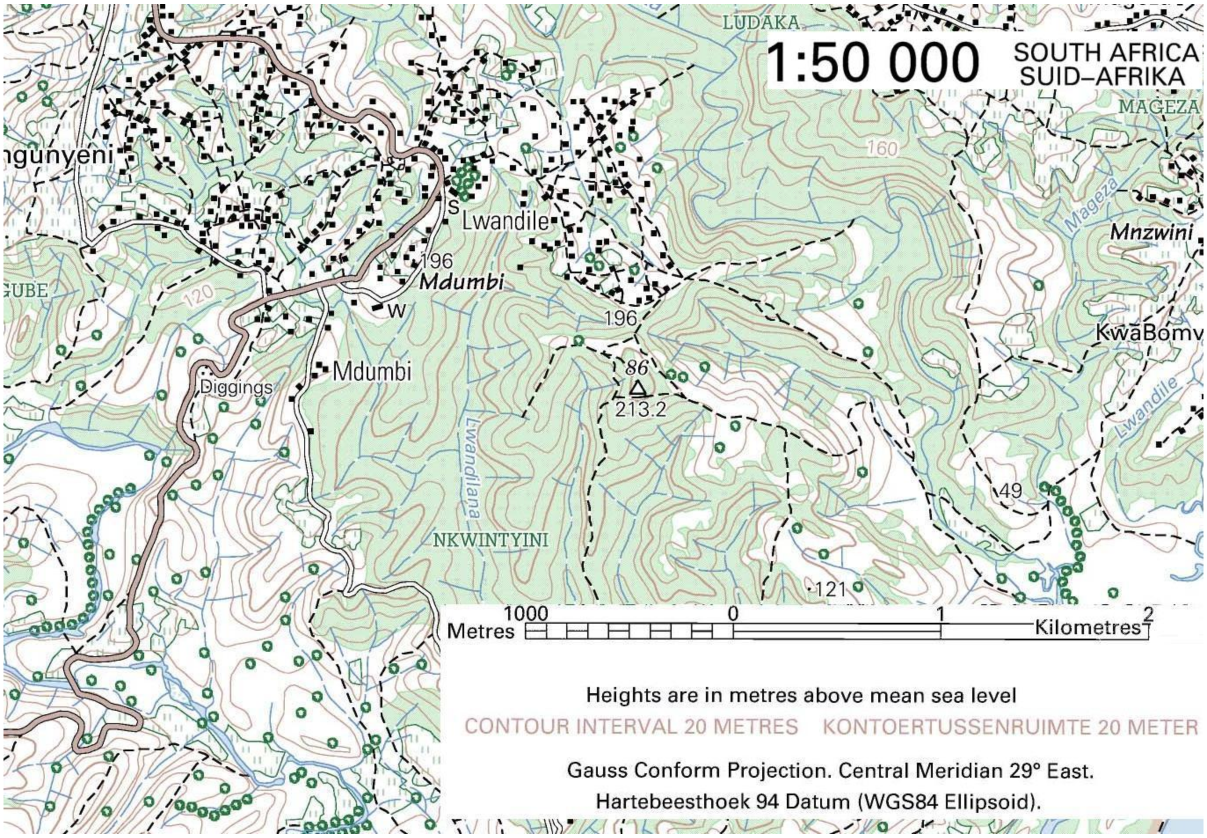


Figure 4.3: Image showing varying elevations in the region through topographical contour lines in the Mdumbi region. The closer the contour lines are to each other, the quicker the surface changes elevation, the further apart the flatter (slow elevation change) the area is. The Mdumbi area has both, showing areas of high elevations around those of lower elevations, making it a hilly area (source and permissions [Chief Directorate of the National Geo-Spatial Information, Cape Town]).

4.8 Design of the Physical Testbed

The testbed was the source of data analysed and used to answer the research questions posed by this work. The type of data needed to determine the testbed design and subsequent experiments carried out. This work seeks to compare the augmentation of the existing Batmand protocol (the system developed in this dissertation) with the original Batmand used in the augmentation. The network data needed in order to establish a comparative analysis of the performance of the two protocols are percentage packet loss, delay and throughput. These network performance values are essential in determining network performance especially in networks dealing with voice, as does the MP network. Furthermore, in the rural characteristics observed, distance be-tween nodes is a key feature affecting the performance of the wireless network, as the increased distances causes the hops experienced by the network to increase. The effects of increased hops can be viewed from the analysis of the metrics being recorded as experiment data.

Performance analysis is viewed in two ways: the first is the comparative performance of the two protocols being analysed (augmented Batmand and Batmand) and the second is the individual performance of each protocol. In order to achieve this, the testbed was created and scenarios where constructed after. The scenarios, described in this chapter, yield data that can be analysed and from that analysis the two performance views obtained. However, before the experiments are run and the data obtained, the physical testbed has to be built.

The most practical location for the testbed was the Computer Science building at the University of Cape Town (UCT). This location was most suitable because it offers enough space to distribute the nodes, full access to the building at all times, and security, as the devices cost a fair amount. The image shown on Figure 4.4 shows the blue print of the third floor of the computer science building, in which the testbed was intended to reside. Access to all the rooms is possible. The distance from one end of the corridor to the other is 55 meters.

The testbed was composed of:

- Two Unix machines.
- Unix machines with wireless cards, as many as we can practically fit in the available space.
- MPs, if performance improvements are seen on the first experiments that use Unix machines.
- Network performance measuring software.
- A WiFi signal analyzer

The two Unix machines and MPs ran one of either the original Batmand or the new system created from it. One Unix machine was placed in Room 300, on the third floor as per Figure 4.4. This machine as well as all the other nodes in the network where placed away from people walking by as to avoid unnecessary disturbances , places such as on top of bookshelves or storage cabinets or vacant sections of rooms. The Unix machine is designated Node (0) of Figure 4.4 which shows the placements of the nodes in the testbed. In the opposite direction, the second Unix machine was placed in the farthest room (bottom right corner) Node (4). In between these two Unix machines were the MP Nodes (1-3). The MPs performed all the routing on the network. No routing was performed by the Unix machines, from the routing perspective they were passive network nodes. The dotted lines in Figure 4.4, between the network nodes, represent the links between MPs that could communicate directly with each other, and the ones that could not. Each link (dotted line) represents one hop in the network. These were controlled through the use of signal attenuation, which forces such connections in the testbed. The Unix machines, behaved as the source and destination of traffic in the network. One Unix machine will generated packet traffic and the other receive this traffic. The MPs in the network were in charge of routing the traffic from one end of the network to another.

The same set up was maintained for each of the two protocols tested. The experiments started with one of the protocols, then, the MPs were reflashed with the other protocol. The same experiments repeated for the newly flashed protocol. Each of the rooms where the network nodes where placed had thick wall which served as physical barriers which assist in intentional signal attenuation. Also in these rooms additional barriers and strategic node placements further attenuated node signals as needed. Decision was made to execute the experiments after hours when the presence of individuals walking around was rare. Also minimizes any disturbances that could arise from other devices using the medium. Further more, the channel chosen for the experiments was channel one, as mentioned on 4.1. Besides being the default channel for the MPs, it was the least busy amongst the available channels in the surrounding areas as observed from the use of our wi-spy gadget which allows for spectrum activity analysis. However, this does not imply that there were not other devices active on this channel, because there were. As the universitie's wifi hotspots, students personal connections and other devices such as microwaves are active across the spectrum including the Batmand operational channel one.

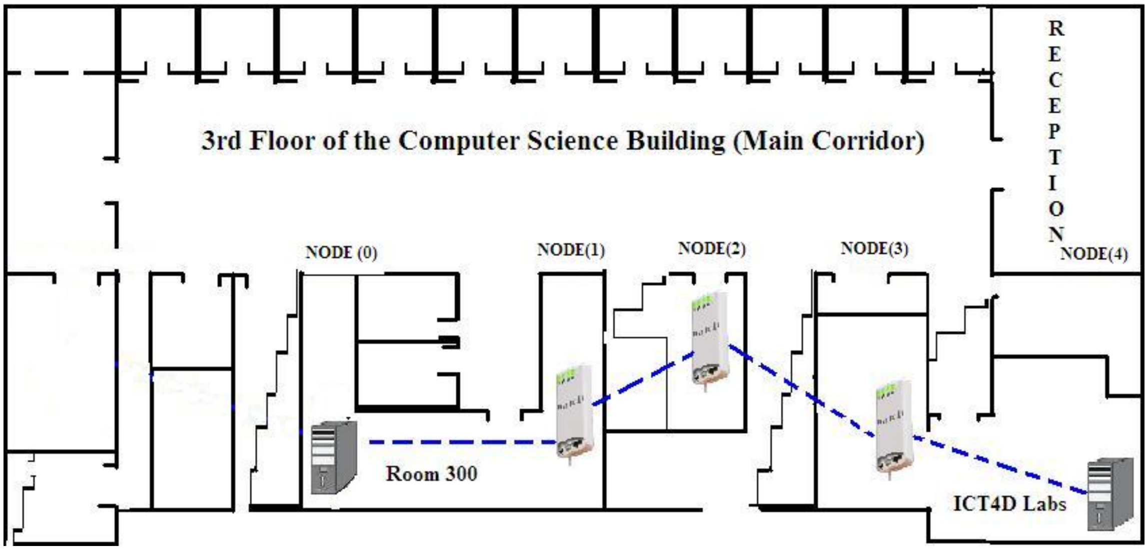


Figure 4.4: Shows the blueprint of the testbed with nodes and links shown by dotted lines.

4.8.1 Testing

Testing was conducted on the testbed matching the physical topology mentioned in Figure 4.4 and matched the scenarios described in Figure 4.5. In the testbed, the Unix machines generate traffic in the form of data packets. These packets have sizes 73 bytes and 1500 bytes, representing voice and standard Ethernet packets respectively. In doing so, the project compared the performance of the MP network when dealing with voice and data packets sizes on networks that use the original and modified Batmand routing protocols.

In each conducted experiment, the load was varied (load was the packets generated and sent by the Unix machine in charge of generating traffic). 1000 UDP packets of size 73 bytes were sent, this was to be repeated 60 times, referred here as iteration. Then, the packet sizes were increased to 1500 bytes. This was also iterated 60 times. Each of the experiments were repeated for each independent number of hops represented by the scenarios shown on Figure 4.5. In each scenario, the load and number of hops traversed were observed to see how it affected each of the metrics chosen to be scrutinized. The chosen metrics were Throughput (Tp), Jitter (J), Packet Loss Ratio (PLR) and Delay (D). These are all essential metrics that needed to be analysed so that success of the project could be seen by the measurement of the performance of the Batmand protocol. These metrics are the best suited to observe a voice networks performance, given the environmental characteristics described here. Further, these experiments were the vehicles used to obtain the answers needed for the research questions. All experimental parameters are summarized on Table 4.1. The testing was split into scenarios, each producing a part of the data that is needed to reach conclusive answers to the research questions. Breaking the test down into scenarios makes it easier to manage and analyse the copious amount of data generated.

Table 4.1: A summary of the experimental parameters.

Testing parameters	Value
Number of nodes	5
MAC	IEEE 802.11b/Channel 1
Wireless card mode	ad-hoc demo
Essid	potato
AP	01:CA:FF:EE:BA:BE
Maximum Transmitted power	16 +/- 1dBm
Minimum Transmitted power	2 +/- 1 dBm
Packet sizes	1500 and 73 bytes UDP
Data rates	1795pps and 87pps
Iterations/Duration	60
Number of Packets sent per Iteration	1000
Routing Protocols	Batmand and Modified Batmand
Metrics Scrutinized	Throughput (Tp), Jitter (J)
Metrics Scrutinized(2)	Packet Loss Ratio (PLR) and Delay (D)
Total number of achieved Hops	4
Card transmission power range	1dBm - 6dBm
Link speed	11Mbps

4.8.2 Scenarios

The testbed was deployed in phases; these phases are referred to as scenarios. Each of the scenarios matched the experiments and tests that were conducted. Therefore, each hop investigated has a scenario described here. Figure 4.5 shows the hop scenarios in detail. Each of the hop scenarios included two Unix machines and zero or more MPs, each placed in between the Unix machines as needed, to achieve the desired number of hops. The one hop scenario did not use any MPs. It is noted that the data gathered from this scenario served for comparison purposes with the scenarios that include the MPs.

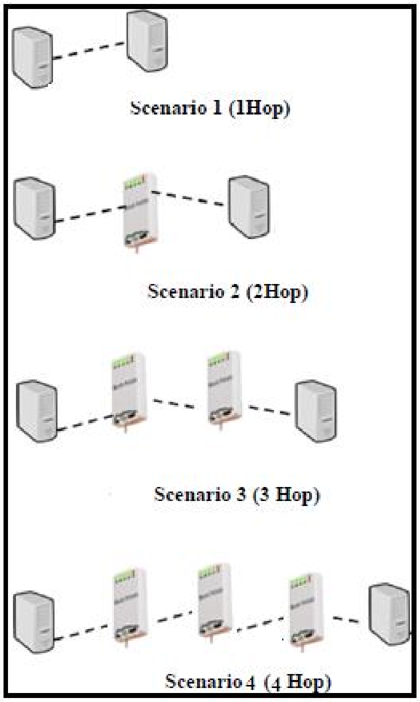


Figure 4.5: shows the scenarios that match the experiments and tests that will be conducted.

4.9 Conclusion

The design of the delay routing metric scheme has given the necessary foundation upon which to build the system. The routing metric scheme is now a tangible concept built from the initial aims. Next the system, described here, will be tested, and the results presented.

Chapter 5

Experiments and Results

This dissertation has researched and motivated the appropriate modifications to the Batman algorithms path selection process through the original Batmand (O-Batmand) routing protocol implementation. The motivation was that better quality of service (QoS) could be provided to the O-Batmand and mesh potato (MP) network by adding relevant network service metrics to the O-Batmand routing protocol. As an initial step in proving the validity of this idea, the delay metric was chosen and introduced into the O-Batmand routing protocol. This yielded a new version of the protocol named modified Batmand (M-Batmand). Experiments were then conducted in order to establish if the M-Batmand protocol with its modifications improved the O-Batmand protocols performance in a laboratory indoor testbed.

The experiment and results chapter will analyze the results obtained from conducting the experiments described in the Test Design, Chapter 3. The analysis will show whether the system motivated and implemented in this project, M-Batmand performed better than the already established system, O-Batmand. If so then the M-Batmand routing protocol will have improved upon the O-Batmand routing protocol. The scope of this chapter is to reveal the outcome of the project as a whole.

5.1 Scope

This chapter will be focusing on the effects, adding, the delay metric to the O-Batmand routing protocol has on the MP network. How delay is calculated was expalained in the System Design Chapter 4 in the Delay Assisted Transmission Quality (TQ) section. In order to observe these effects experiments were conducted and the resultant data collected and presented here.

In the first round of experiments the protocols were tested in the exact manner described on the Chapter 4. The one difference was that all the MPs were replaced with Unix boxes, these where desktop machines with Intel core two quad processors, 3 gigabytes (Gbs) of random access memory (RAM), running the Ubuntu 10.10 (Maverick Meerkat) operating system. This was done in order to fully test the M-Batmand implementation before porting it to the MPs. In this manner M-Batmand is ported if it out performs the O-Batmand protocol in these experiments making it a better option to be used on the MP devices. Next the results of these experiments will be presented through a series of tables and graphs depicting the performance of the M-Batmand in relation to its predecessor, O-Batmand.

The performances of the protocols were compared in terms of throughput, delay and percentage packet loss of UDP data packets, which were computed at the receiver. The measuring tool used for these experiments was Iperf. Each are here discussed, and the relevant conclusions drawn. These metrics are the same as those short listed to be implemented as modifications to O-Batmand. Also these are the metrics understood to best describe the QoS on the links in VoIP networks operation is a rural environment.

This data is represented graphically on box and whisker plots. This method of presenting the data was chosen as box and whisker plots, for these experiments, best show the trend in the changes which, henceforth, are referred to as oscillation, on the network. Using this type of plots the, median, upper quartile, lower quartile, maximum values, minimum and outliers can be observed much easier across all hops, showing the data distribution on each hop. In the plot, the bottom and top of the box are 25^{th} and 75^{th} percentiles, and the band near the middle of the box is the median. The ends of the whiskers represent the maximum and minimum data values. This will provides a clear picture of the hop trend as well as overall trend of the effects of increasing hops on the network. A table with the mean scores are also provided. The alternative is to use scatter plots and best fit lines which would show the strength of the correlation or better yet the effect the independent variable on the X-axis had on throughput Y-axis. Scatter plots here would work if time axis were used but they are not.

On these graphs the X-axis shows the hop number and packet size for the data being shown. The Y-axis shows the network parameter being observed, specifically throughput, delay and percentage packet loss. Note that in each hop the data displayed represents 60 iterations of 1000 packets of the designated size sent. This produced 60 data points that were then used to create the plot.

5.2 Throughput

Throughput is the average rate of successful messages/packets, delivered over a communication channel, per unit time. The throughput values show the achieved average useful bit rate in the network. In these tests throughput was measured in kilo bits per second (Kbps).

The graphs, shown here on Figures 5.1, 5.2, 5.3 and 5.4 each focus on the individual protocols and the packet sizes used in the experiments. The graphs show and compare the performances and oscillations of throughput of each protocol across all hops matching one of the four scenarios described in Chapter 4.

5.2.1 Throughput Results of all Hop Scenarios

Figure 5.1 and 5.2 show the trend observed when the dependent variable, throughput, is affected by the independent variable, load, on the four hop scenarios. The horizontal line seen on the first three hop scenarios on Figure 5.1 and one hop and three hop on Figure 5.2 represents the median values and the absence of a box means that there was very little difference between the median, upper quartile, lower quartile, maximum and minimum datum. The significance of this is that all results in these scenarios where consistent with the median values, therefore little distribution in the throughput results meaning throughput oscillations occurred. This is only fully true in the one hop scenario, for both protocols as the presence of outliers which are the points above and below median is clear evidence of oscillation of the throughput. Throughput oscillation affects the network quality of service (QoS). High oscillation result in low network QoS and indicate degrading network performance.

In the first hop scenario one would expect to see similar throughput data for the two protocols. However, this is not the case, for the 73byte sized packets O-Batmand and M-Batmand had an average throughput of 5980kbps and 2295Kpbs respectively. Int he 1500byte case the same is observed, O-Batmand and M-Batmand had an average of 2990Kbps and 1650Kbps. These values ca be seen on Tables 5.1 and 5.2. In both load cases M-Batmand had a lower throughput. The reason for this difference in throughput between the two protocols can be explained through observing the factors known to affect throughput. These factors are: packet loss, delays and jitter. In this case, the packet loss in the first hop is the primary reason for the differences in throughput. Tables 5.5 and 5.6 show the the percentage packet loss experienced in the network in the first hop scenario. In this these table it is seen that M-Batmand in the 73byte and 1500byte cases experienced an average of 1.17 percent and 2.05 percent packet loss, while O-Batmand experienced 0 percent packet loss for both cases. This coupled with the delay results caused the difference in throughput experienced between the two protocols.

O-Batmand results show higher oscillation in the two, three and four hop scenarios, however since the fourth hop has the biggest distribution in its results it has the highest oscillation thus poorest performance. M-Batman has its highest oscillation in its three hop scenario. Its largest data distribution occurred in the second hop scenario.

The general trend is that throughput decreases sharply for both protocols and is seen by the decreasing throughput values seen on the graphs and the Table 5.1 which shows the mean scores for throughput data for O-Batmand and M-Batmand for 73byte size Packets. Comparatively the highest average throughput belongs to O-Batmand which had a value of 5980.160 kbps in the one hop scenario. In the same scenario M-Batmand averaged 2295.040 kbps. In the 73 byte scenario O-Batmand performed best across all scenarios and even though both protocols experienced increased oscillation as hops increased, O-Batmand had the most oscillations while M-Batmand had the largest data distribution. Also observed was that O-Batmand and M-Batmand seem to be converging to similar throughput values at the fourth hop. This potential convergence could mean that both protocols perform equally at these large hop as the modifications made to M-Batmand seem to not affect the quality of the links at these higher hops. In addition M-Batmand's throughput appears to be slightly more stable at the fourth hop than O-Batmand is. The significance of this is discussed in the discussion section.

Similar trends are experience in the 1500byte load cases seen on Figure 5.3 and 5.4. Things to note are that trends noted in the 73byte scenarios were aggravated with the increase in load, O-Batmand had larger and more frequent oscillation and M-Batmand had wider distribution in its results and highest oscillation on its four hop scenario. The increase in packet size decreases the throughput in the network. In the 1500byte load case O-Batmand's and M-Batmand's throughput decreased to 2990.080 and 1650.256 kbps in the one hop scenarios respectively, however O-Batmand performed better out of the two protocols showing that M-Batmand did not improve on O-Batmand's throughput performance. The results show that for both protocols large packet sizes affect the stability of the network the same. This shows that instability when routing large packets is an inherent feature of the Batman algorithm. Extracted with this is that this instability was not improved on by the modification made on the M-Batmand protocol.

Large distribution in the results represents a fair amount in link quality fluctuations and represents unstable route paths. The discussion section 5.5 explores and discusses more.

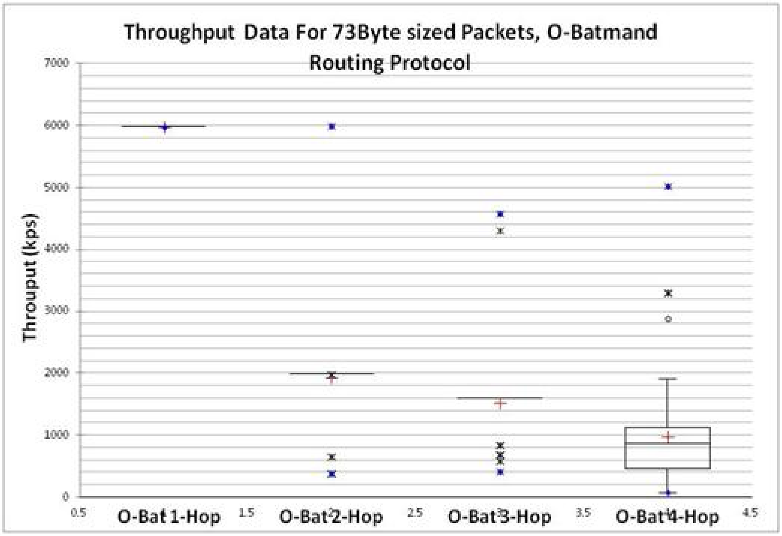


Figure 5.1: Graph for the throughput data for the all four hop scenarios. It compares O-Batmand's throughput performance when 73Byte packets are flooded though the network.

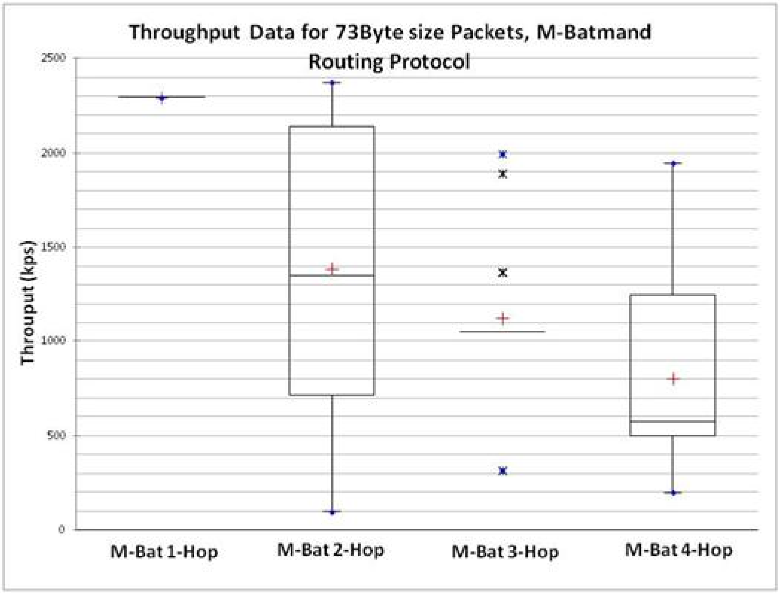


Figure 5.2: Graph for the throughput data for the all four hop scenarios. It compares M-Batmand's throughput performance when 73Byte packets are flooded though the network.

Table 5.1: Mean and Standard deviation scores for throughput data for O-Batmand and M-Batmand for 73Byte size packets in Kilo bits per second (kbps)

Hop	O-Batmand	STD	M-Batmand	STD
1	5980.160	0.000	2295.040	0.000
2	1935.433	677.066	1382.908	768.727
3	1525.913	653.620	1125.525	347.574
4	984.030	881.051	804.763	483.716

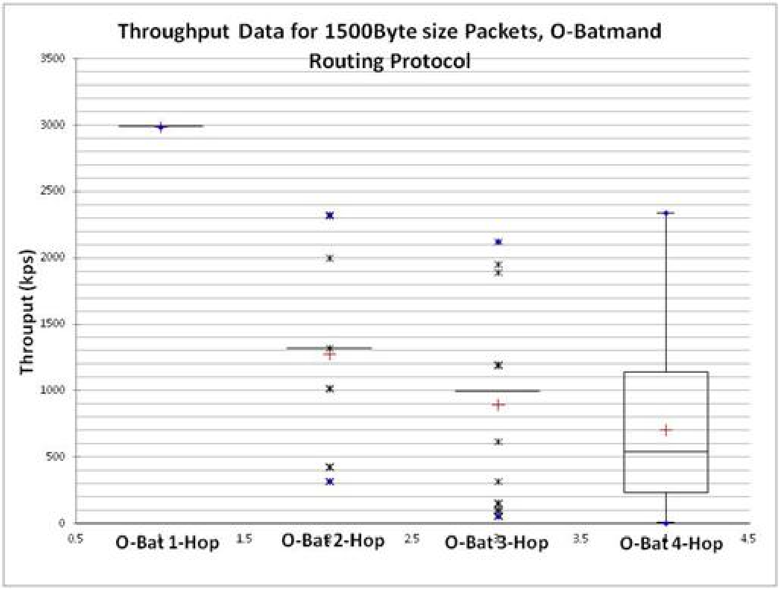


Figure 5.3: Graph for the throughput data for the all four hop scenarios. It compares O-Batmand's throughput performance when 1500Byte packets are flooded though the network.

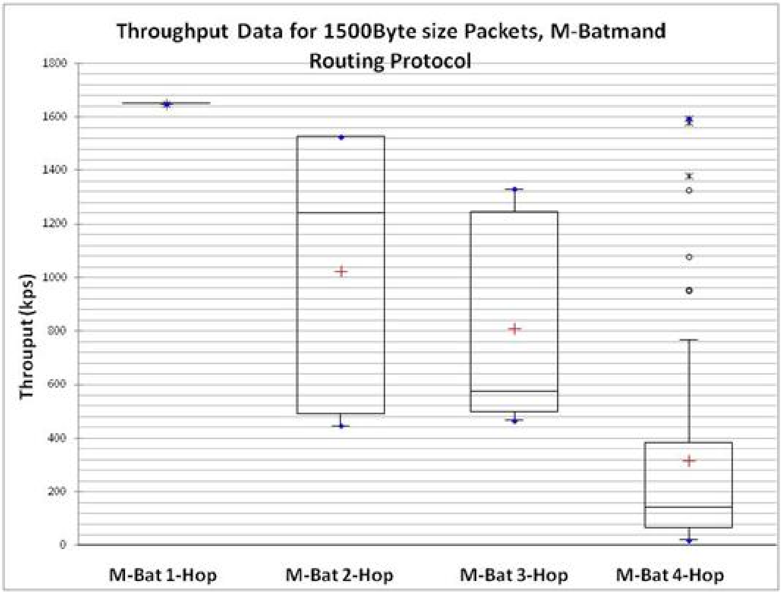


Figure 5.4: Graph for the throughput data for the all four hop scenarios. It compares M-Batmand's throughput performance when 1500Byte packets are flooded though the network.

Table 5.2: Mean and Standard deviation scores for throughput data for O-Batmand and M-Batmand for 1500Byte size packets in Kilo bits per second (kbps)

Hop	O-Batmand	STD	M-Batmand	STD
1	2990.080	0.000	1650.256	0.000
2	1278.282	388.769	1027.802	511.865
3	900.529	421.216	812.011	357.741
4	705.594	568.463	316.369	400.05

5.3 Delay

On voice calls the large delays in packet transmissions (source to destination) are perceived by the callers, as echo which is when the caller hears their own voice repeated back to them. These delay effects are also perceived as talker overlap which is when callers seem to be talking simultaneously, cutting off each others speech, but in fact the large delays in transmission and reception are in effect.

Recorded delay values obtained from experiments were shown graphically on box and whiskers plots. This was done in the same manner as the throughput results were previously presented. The resulting effects load had on delay, in each of the hop scenarios, are shown next.

5.3.1 Delay Results of all Hop Scenarios

The graphs, shown here on Figures 5.5, 5.6, 5.7 and 5.8 each focus on the individual protocols and the packet sizes used in the experiments. The graphs show and compare the performances and oscillations of throughput of each protocol across all hops matching one of the four scenarios described in Chapter 4.

The best performances are represented by the lowest values. This indicates how well the protocol handles delay in the network. The general trend observed on the 73byte load cases are represented on Figures 5.5, and 5.6,these show that delay increased with increasing hop. Delay is constant for the one hop scenario with mean scores of 0.039ms and 0.822ms for O-Batmand and M-Batmand respectively. The worst delay values are experienced on the four hop scenario where O-Batmand averaged 12.013ms and M-Batmand averaged 9.875ms. These values are well within the ITU-Recommendation G. 114 150ms delay bracket.

Comparatively M-Batmand had the lowest and thus best delay results from the second hop scenario onwards. However, the difference between the delay performances on the two protocols is small across all the hop scenarios. Delay oscillations occur most on the four hop scenario for both protocols, with M-Batmand having the largest oscillation, but again the difference is small.

The 1500byte load case can be seen on Figures 5.7 and 5.8, these presented similar findings to those found on the 73byte cases. The differences being that the delay values where significantly larger from the two hop scenario onwards, for both protocols. This indicates that larger data packets result in larger delay values in network. The worst delay values are experienced on the four hop scenario where O-Batmand averaged 40.179ms and M-Batmand averaged 20.755ms. These values are well within the ITU-Recommendation G. 114.

Comparatively on the 1500byte case, O-Batmand had lower delay values than M-Batmand on all the hop scenarios, but only by a small difference. However, on the four hop scenario the O-Batmand delay values exceed those experienced by M-Batmand by a difference of 19ms which is the largest difference in delay values between the two protocols. Delay oscillations are higher for the O-Batmand protocol from two hop through to the four hop scenario.

Overall M-Batmand protocol is more sensitive to the delay on the network than O-Batmand is at higher hop scenarios especially on the four hop scenario. This is the result of having added delay metric onto the route selection process which has made M-Batmand more sensitive to network delay than O-Batmand is. However, the delay improvements seen on M-Batmand are small, as the two protocols have small differences in their comparative delay values, and both protocols produce acceptable ITU delay values, even on four hop scenario. More is discussed in the discussion section 5.5.

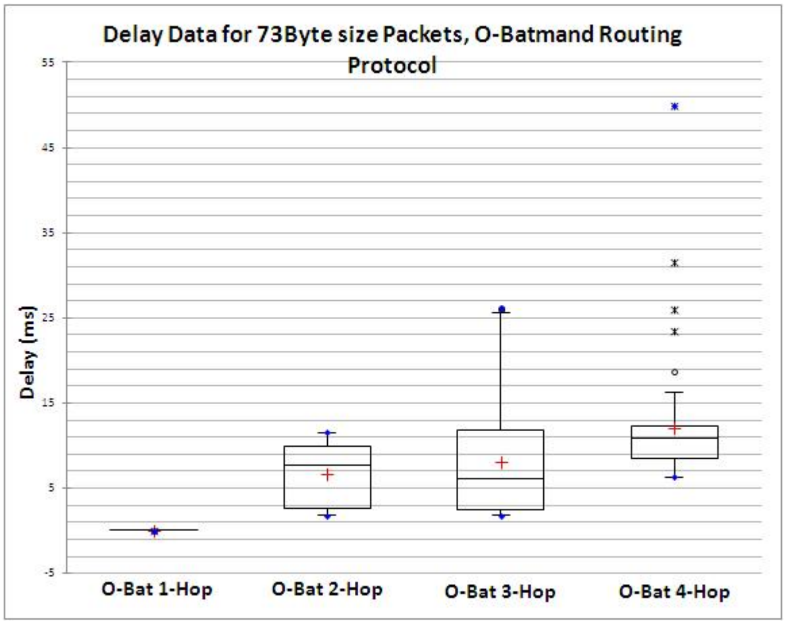


Figure 5.5: Graph for the delay data for the all four hop scenarios. It compares O-Batmand's delay performance when 73Byte packets are flooded though the network.

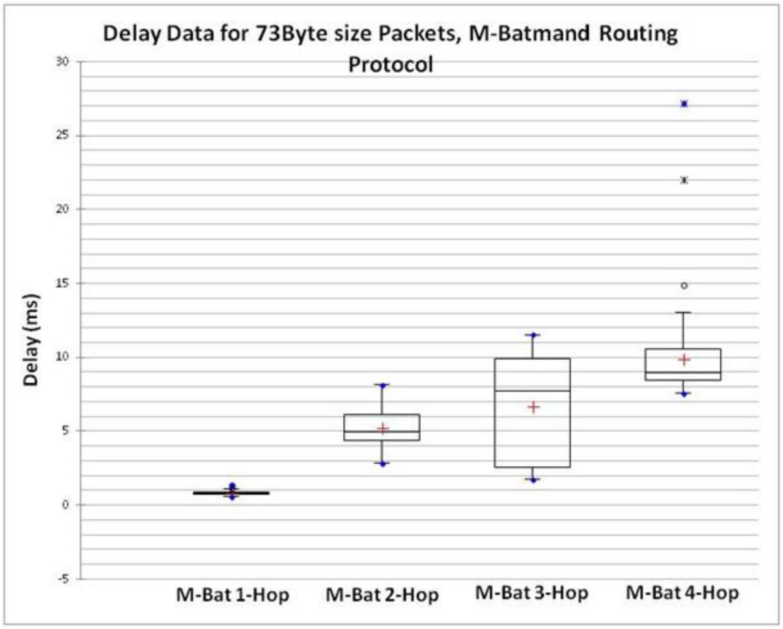


Figure 5.6: Graph for the delay data for the all four hop scenarios. It compares M-Batmand's delay performance when 73Byte packets are flooded though the network.

Table 5.3: Mean and Standard deviation scores for delay data for O-Batmand and M-Batmand for 73Byte size packets in milliseconds (ms)

Hop	O-Batmand	STD	M-Batmand	STD
1	0.039	0.002	0.822	0.149
2	6.691	3.503	5.236	1.280
3	8.094	6.705	6.691	3.503
4	12.013	6.742	9.875	3.147

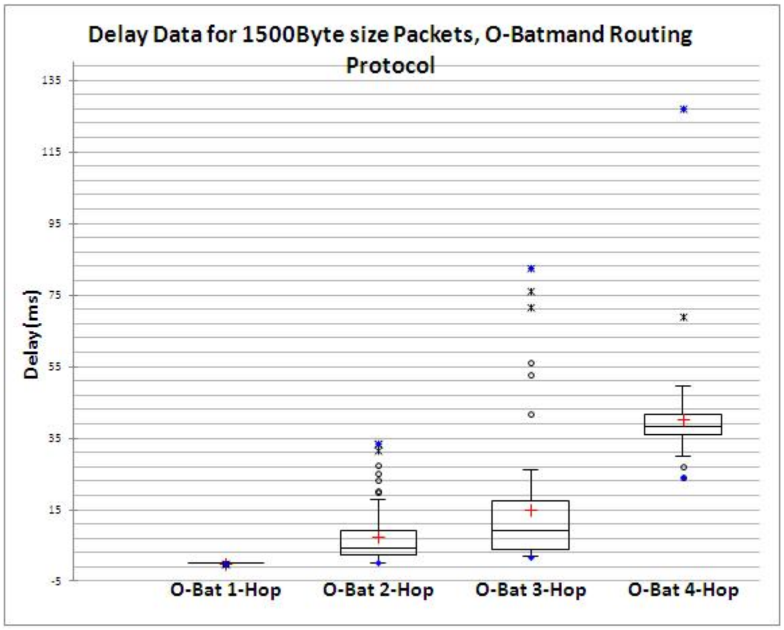


Figure 5.7: Graph for the delay data for the all four hop scenarios. It compares O-Batmand's delay performance when 1500Byte packets are flooded though the network.

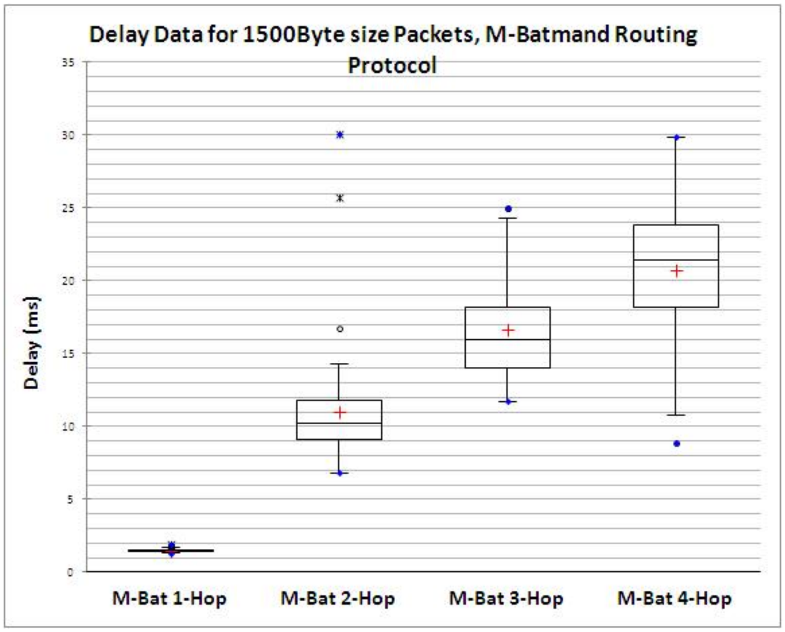


Figure 5.8: Graph for the delay data for the all four hop scenarios. It compares M-Batmand's delay performance when 1500Byte packets are flooded though the network.

Table 5.4: Mean and Standard deviation scores for delay data for O-Batmand and M-Batmand for 1500Byte size packets in milliseconds (ms)

Hop	O-Batmand	STD	M-Batmand	STD
1	0.043	0.001	1.509	0.143
2	7.627	8.483	11.015	3.772
3	15.094	18.139	16.654	3.444
4	40.179	12.998	20.755	5.055

5.4 Percentage packet loss

VoIP is not tolerant of packet loss, to the extent that high packet loss can degrade the call quality. In VoIP networks, lost packets can cause a call to break up, and too much of this can turn the conversation incomprehensible [58]. The graphs, shown here on Figures 5.9, 5.10, 5.11 and 5.12 show the distribution of the percentage packet loss experienced in the network for each hop scenario for both protocols. The mean values of the data were tabulated and presented for both 73byte and 1500byte load cases, this is presented next.

On the one hop scenario the O-Batmand protocol has lower percentage packet loss, for both load packet sizes; here O-Batmand averaged 0 percent packet loss. For the same scenarios, M-Batmand averaged 1.167 and 2.050 percent packet loss for 73byte and 1500byte load sizes respectively. The difference in performance between the two protocols in the first hop scenario is small. However, this gap increases as the hops increase and become similar at the four hop scenario. This indicates that packet loss dramatically increases with increased hops traversed and that O-Batmand manages packet loss better than M-Batmand. Furthermore, in the 1500byte case the percentage packet loss is far greater for both protocols. In the 73byte load case there is a dramatic increase in packet loss from the two hop to the three hop scenario for both protocols. This suggests that the three hop scenario link is of a weaker quality than the others. This link traverses more obstacles than the other links. Under such links O-Batmand performs better than M-Batmand as it has far less packet losses, averaging 39 percent while M-Batmand averaged 79 percent. This is also true in the 1500byte load case where O-Batmand averaged 79 percent and M-Batmand averaged 84 percent.

In general, M-Batmand has a higher packet loss than O-Batmand in all the hop scenarios and load cases. However the M-Batmand has a slightly lower packet loss at the four hop scenario in the 1500byte case but this difference is small, only three percent. There is little oscillation in packet loss values for both protocols, but it is more apparent for M-Batmand on the 1500byte case for the three and four hop cases.

In meshed networks such as our testbed there are known factors that contribute to packet loss. Factors such as interfering signals from nodes operating on the same channel. The interferences cause collisions at the receiver leading to collision induced packet loss. Note that this is usually avoided by the carrier sensing and random wait performed by wireless cards before transmitting. However, the cards in O/M-Batmand networks are operating on the *ad hoc* demo mode which if recalled does not use the MAC layer features created to avoid collisions. It stands to reason that one of the potential caused of the packet losses is this. Other sources are dropped packets due to link degradations. One last factor that affects packet loss is slow CPU processing speed. If the kernel cannot process the packets arriving at the interface fast enough due to possible contention for CPU resources then packets are dropped. In the case of the M-Batmand protocol it is possible that the added calculations (some floating point) cause contention at the kernel. Out all of these factors the most probable reason for the packet loss is the link degradation due to the signal attenuation techniques used as these weaken the transmitting signals.

The meaning of the results is discussed in the discussion section presented next.

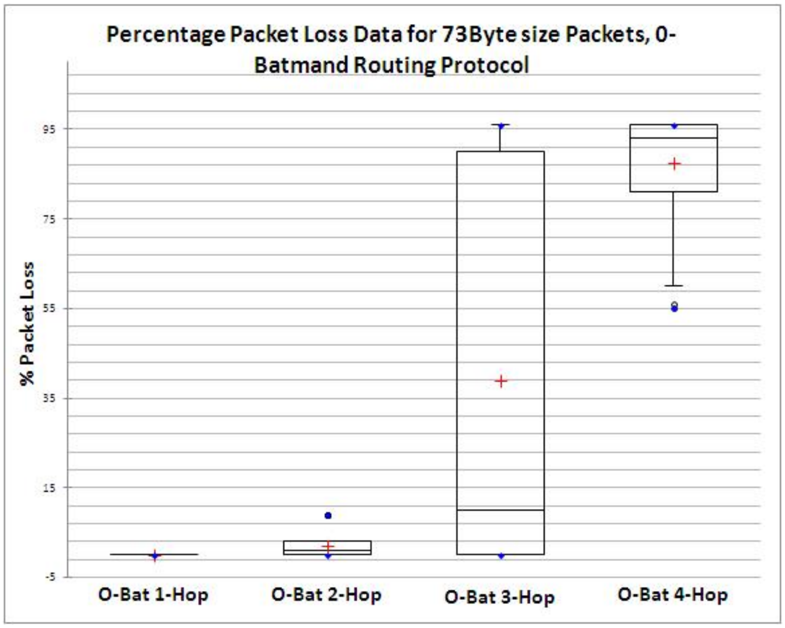


Figure 5.9: Graph for the percentage packet loss data for the all four hop scenarios. It compares O-Batmand's packet loss performance when 73Byte packets are flooded though the network.

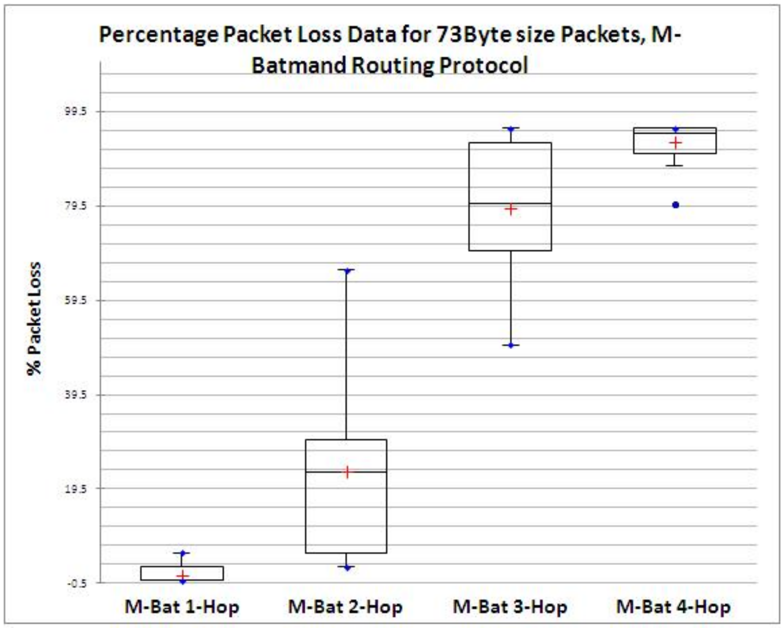


Figure 5.10: Graph for the percentage packet loss data for the all four hop scenarios. It compares M-Batmand's packet loss performance when 73Byte packets are flooded though the network.

Table 5.5: Mean and Standard deviation scores for percentage packet loss data for O-Batmand and M-Batmand for 73Byte size packets

Hop	O-Batmand	STD	M-Batmand	STD
1	0	0	1.167	1.520
2	2.067	2.577	23.133	16.507
3	39.083	41.463	79.000	13.953
4	87.367	10.811	93.117	3.971

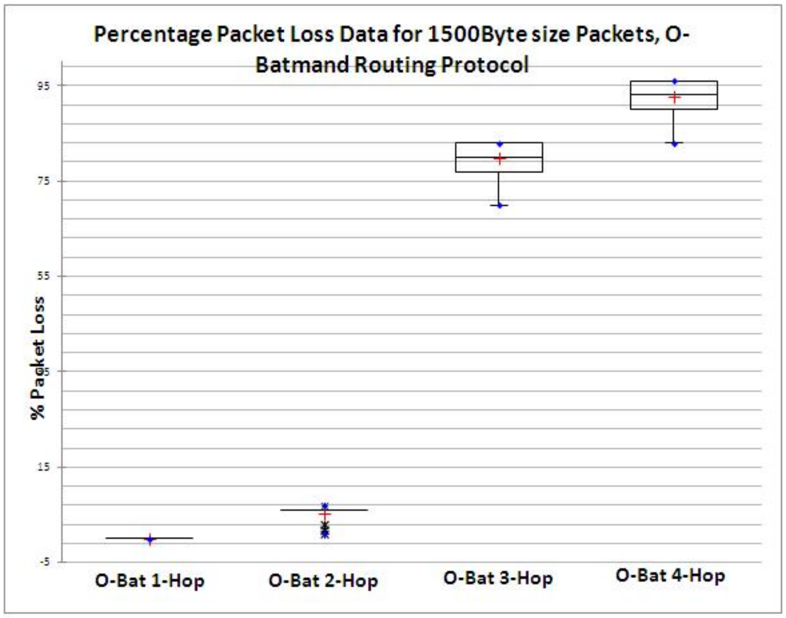


Figure 5.11: Graph for the percentage packet loss data for the all four hop scenarios. It compares O-Batmand's packet loss performance when 1500Byte packets are flooded though the network.

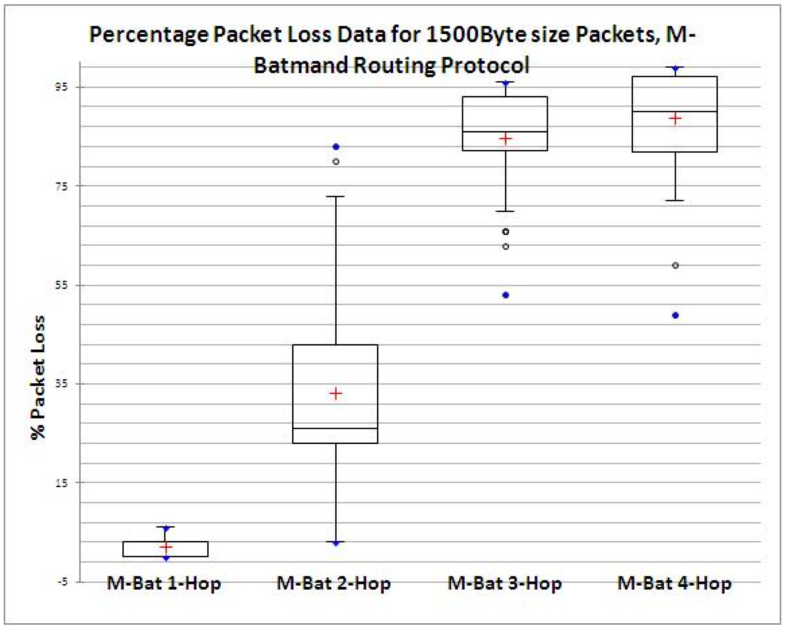


Figure 5.12: Graph for the percentage packet loss data for the all four hop scenarios. It compares M-Batmand's packet loss performance when 1500Byte packets are flooded though the network.

Table 5.6: Mean and Standard deviation scores for percentage packet loss data for O-Batmand and M-Batmand for 1500Byte size packets

Hop	O-Batmand	STD	M-Batmand	STD
1	0	0	2.050	2.150
2	5.283	1.606	33.367	19.328
3	79.717	3.867	84.650	10.226
4	91.717	3.867	88.850	10.494

5.5 Discussion

The results here shown are the values obtained from the experiments conducted on an indoor testbed, designed to mimic abstracted characteristics of the Mdumbi region taken as a case study in this dissertation. These results serve to compare the performance of the M-Batmand protocol to that of O-Batmand routing protocol which serves as acceptable performance standards for a protocol implemented from the Batman algorithm definition. The data collected in order to perform the comparisons were the network throughput, the network latency/delay and the percentage packet loss. These values are all accepted network metrics which help understand how a wireless network performs.

The general trend with both protocols and all the metrics measured was that the networks performance fell with increasing hops. There were high oscillations in throughput and delay on the two and three hop scenarios and minor oscillations for packet loss. The oscillations were often coupled with a wide distribution of the data which means that the networks performance was unstable for those hops. This is due to link quality fluctuations observed in the experiment results obtained. M-Batmand's throughput and packet loss data were large, revealing that these were the protocol's primary sources of ink quality fluctuations on both load cases on the two and three hop scenarios. However, its delay data was comparatively more stable than those of O-Batmand especially on the four hop scenario for both load cases. Network throughput is expected to decrease while packet loss and delay are expected increase with each hop traversed along a path to the destination node. In each hop the packets experiences contention, and the bandwidth is essentially halved with each hop traversed [10]. Also in each hop the packet faces potential interference from other transmitting nodes this is especially significant for the O/M-Batmand networks as its wireless cards are in *ad hoc* demo mode. The sharp decline in performance at higher hops can be explained: the distances between nodes are larger at the higher hop numbers so the already attenuated signal is further weakened as radio signals weaken with distance observed in card ranges. Also the obstacles and interferences experienced by the nodes in the testbed increases with increasing hops. This is due to the inherent nature of the testbed, where we seek to mimic certain abstracted environmental characteristics (topology, single path, multihop network) observed at the Mdumbi region. Furthermore, nodes are placed in departmental laboratories; each laboratory has its own WiFi network which operates on the same 2.4 GHz band as the testbed nodes. Also there more physical interferences between the second, third and fourth (forming the second and third hops) nodes as there are more walls and other obstacles between them. Further, the signal between these two hop scenario at some stage crosses from inside the building across a balcony (outside) space and back into the building and finally into the laboratory which gives the testbed similar conditions as large hops across elevation in the rural areas we seek to mimic.

Furthermore, for the 73byte cases at the fourth hop the M-Batmand protocol was slightly more stable than O-Batmand for throughput. The observed stability is minor and can be attribute to link quality variations standard to all mesh networks and outside the control of the experimentations.

Given the testbed conditions presented to both protocols O-Batmand performed best overall. The results obtained from the testbed show that O-Batmand is well suited for the task of routing packets inside a wireless network. It is designed and supports voice and data packets as even at the fourth hop the delay values were with the accepted rages for voice communication. All hop scenarios were under the recommended 150ms bracket set by the ITU-T in its G.114 series recommendation. However, the results show that this quality deteriorates in the testbed with increasing hops. This is also true for the M-Batmand implementation; however, the comparative performance demonstrates that M-Batmand deteriorates slower than O-Batmand on the metrics except delay where it out performed O-Batmand. This is visible from the graphs previously presented, as an example the 1500 throughput data is taken into consideration. In this scenario the O-Batmand starts off in the first hop with 2990kps then drops by 1712kps to 1278kps in the second hop. On the other had M-Batmand starts off with 1650kps in the first hop and drops by 623kps, which is less than M-Batmand's, to 1027kps. In the second hop the same trend continues where O-Batmand drops by 378kps to 900kps from hop two to hop three, while M-Batmand drops by 215kps to 812kps on the same hop rage. This trend is true for the remaining hops and also true for the other metrics measured. This demonstrates that although minor M-Batmand is more stable then O-Batmand.

The addition of the delay metric onto the Batman algorithm's route selection process has created the M-Batmand protocol which is more sensitive to network delay than O-Batmand is at higher hop scenarios especially on the four hop scenario. However, the delay improvements seen on M-Batmand are small, as the two protocols have small differences in their comparative delay values and since all the values for both protocols are within the acceptable boundaries. This improvement does not outweigh M-Batmand's lack of performance in throughput and packet loss when compared to values obtained from O-Batmand. The data, however, seems to suggest a convergence of the performance of the two protocols at an undermined hop number. This is seen form the closeness of the results on the fourth hop and apparent trend towards this seen on the previous hops. The data does not suggest that M-Batmand could eventually surpass the performance of the O-Batmand protocol when it comes to throughput and packet loss. To assume this would be to go further then any data presented here could support.

The addition of the delay metric onto the Batmand algorithm hinders the performance of the original algorithm this is seen through M-Batmand's comparatively poor performance. These results are consistent with the theoretical analysis of the M-Batmand conducted on Chapter 4. M-Batmand's poor performance is tied to the structure of the Batmand algorithm. In this algorithm there is only one section that allows for the merger of the delay calculations. As explained in Chapter 4, that section translates to the scenario when the algorithm is performing bi-directionality tests. There is only one situation available for updating delay values compared to several for the transmission quality (TQ Batmand's default metric) values. Remember that the path selection value used by the M-Batmand protocol is a composite metric called Delay Assisted Transmission Quality (DATQ). In each round of calculations DATQ is updated according to the fresh TQ and delay values available. This created a situation where the frequency with which the algorithm updates the path selection value is much higher for the TQ value than it is for the delay value. This in turn creates internal confusion as the final path selection value, DATQ, fluctuates between TQ and TQ plus delay influenced values. The resultant problem is that link quality assessments are much higher in M-Batmand, and this disturbs the proper functioning of the protocol.

Furthermore, the delay calculations add to the CPU cycles required by the protocol. The lack of calculations performed on the O-Batmand protocol is one of the reasons why the protocol performed well. The addition of the delay metric requires constant calculations. The additional calculations create contention for CPU resources. This contention slows down the kernel response to network packets and its processing reducing the protocols performance. This is aggravated by larger packets in the M-Batmand network. In general larger packet sizes slow down mesh networks because large packet sizes get fragmented by the sender and reassembled by the receiver this means that large packet sizes require more processing by the kernel. Also the fragmentation introduces more packets into the network further stressing it. This is visible through the throughput and delay experiment results which show just this. However, given that M-Batmand protocol already performs worse than O-Batmand with smaller sized packets it stands to reason that this would be worse for larger packets. The kernel demands imposed by the fragmentation and reassembling of large packet size is worsened by the demands of the additional delay calculations and the increase in the number of network packets sent. These constant calculations have slowed the performance of the M-Batmand routing protocol resulting in its poorer performance.

Lastly the results shown here for throughput, delay and packet loss show that both protocols can support voice and data on a wireless network. O-Batmand performs better at these tasks then M-Batmand does.

5.6 Conclusion

This chapter present the experiment results and discussions for the testes described in 4. The final conclusions are presented next in the Conclusion Chapter 6.

Chapter 6

Conclusion

The mesh potato (MP) is a wireless router connected to a plain old telephone (POT), which allows voice communication over wireless mesh networks (WMN).The MP fulfills most of the necessary requirements for an economically viable solution to the communication issues discussed in this dissertation. The requirements met include, low set up costs as MPs are essentially plug and play; ease of deployment, MPs use wireless mesh networking (WMN) technologies which are easily deployed; low running costs, once bought the only costs would be in powering the device, lastly, there are no maintenance costs associated with the MP.

The MP technology started in 2008 and is still in its infancy, and as is often true with most new technologies, there is still room for improvements. This dissertation sought improvement by extending the Batmand routing protocol operating on the MP device to included network delay information in its route decisions.

6.1 Evaluation of M-Batmand

6.1.1 Overview

The main focus of the project is to provide better quality of service (QoS) in a laboratory study of the Batman and MP network set up to mimic certain characteristics abstracted from rural areas in Mdumbi, Easter Cape, South Africa. This is needed because the current, default, metric used by Batmand is a simplistic and does not use any of the metrics that affect WMNs in rural environments.

6.1.2 Outcome of The M-Batmand Implementation and Findings

The resultant M-Batmand protocol was created to be an improvement on the existing, original Batmand protocol (O-Batmand) protocol. The evaluation of the resultant protocol was conducted on an indoor testbed. This testbed was designed and constructed to mimic certain abstracted characteristics of a rural environment. These were the multihop, single path nature of the links and also topological features of no direct line of sight due to hilly surface area. This testbed is described in full detail in Chapter 4.

Experiments were conducted on the M-Batmand protocol and the following conclusions were reached. The first of the conclusions: based on our results O-Batmand cannot be modified to include the delay metrics and be expected to improve its performance. Second conclusion is that M-Batmand did not improve the overall performance of the O-Batmand protocol. The addition of the delay metric actually hindered the performance of O-Batmand protocol to the extent that no overall performance gains were obtained. The B.A.T.M.A.N algorithm was designed to be simple and the additions that were made to it on M-Batmand complicate the protocol, and negatively affect the manner in which it was intended to function. The resulting complication then causes the poor performance observed on the M-Batmand results. This research serves to show which changes may or may not serve to improve the O-Batmand protocol.

The third conclusion was that addition of the delay metric improved the performance of the M-Batmand protocol from the delay point of view. The addition of the delay metric enabled M-Batmand to be more delay sensitive than O-Batmand. However, this improvement was minimal as the performance difference between M-Batmand and O-Batmand was small. Also O-Batmand's performance across all the scenarios, although lower than that of M-Batmand, was well within the expectable boundaries set by the ITUT in its G.114 series recommendation for acceptable communication quality.

In addition, the inclusion of new metrics onto the O-Batmand protocol creates an overhead that the original B.A.T.M.A.N algorithm, in its design paradigm, intended to overcome by keeping the algorithm simple. The overhead experienced is the one created by the protocol's network control packets, originator message (OGM) packets. The algorithm deals with this overhead issue by having the route decision making process involve no calculations, therefore, saving on CPU cycles. The addition of the delay metric, and potentially any other metric, violates this design decision. The delay metric implementation involves actual calculations spanning two internal functions. More details on the calculations conducted by M-Batmand can be found on Chapter 4 of this dissertation. The modifications are significantly more computationally costly and this added stress is the primary reason behind the performance degradation observed on the M-Batmand protocol.

It is clear from the above that any metrics intended to improve the O-Batmand protocol have to be minimalistic in the calculations they perform. In this case, the best metrics are those that have no actual calculations and are similar to the existing counting/statistical metric used by the O-Batmand protocol. Such metrics might exist; however, in this research none was uncovered. In that case, the development of a metric matching the specification mentioned above would be the best way to improve the performance of the O-Batmand protocol.

Finally, the intended addition of all other metrics would have also produced detrimental effects to the O-Batmand protocol's performance. This raises the question; is there one metric amongst the selected list (delay, jitter, throughput, pack loss) that may actually improve the performance of the protocol? A secondary question is; is this possible even though its very presence complicates and therefore, upsets the manner in which O-Batmand was designed, and intended to work? Answering these questions would require further research, designing, implementation and a battery of test which forms part of future work.

6.2 Research Questions

This sections deals with the research questions that this thesis has answered.

6.2.1 How does B.A.T.M.A.N perform in a laboratory setup mimicking aspects rural networks?

The aim is to investigate the MPs for use in rural networks this is achieved by abstracting certain characteristics observed at Mdumbi in a field study and mimicking certain aspects of rural environment in a laboratory study. The exact representation of a typical rural environment is not the aim the focus is more on the topological effects of the rural network.

Investigating the MPs for use in rural networks fulfilled the aim mentioned above and has lead to a solution for this research question. In doing so it has allowed the project to primarily determine the viability of the protocol in its intended use. This research question was deemed necessary as without it there would be very little use in proceeding with researching improvements for Batmand protocol, as its viability as an appropriate solution for rural networks would not be known. The viability of the MP and the Batmand protocol as an alternative to the lacking wired landline infrastructure is one of the compelling arguments behind the use of these technologies in rural areas. Experimentation on the project's testbed yielded the necessary data needed to answer this research question.

The results of the experimentations of the O-Batmand protocol revealed that, overall it is well suited for the task of routing packets inside a WMN. It works well inside a network with the environmental characteristics matching those of rural areas such as Mdumbi. The protocol is designed and works well for voice packets and even supports data traffic. These preliminary results were published in international conferences and at the time of this dissertation only these were available for citation [16, 17].

The results obtained here will also serve as values which will be used to find a solution for the second research question, discussed next.

6.2.2 Is it necessary to develop a different route selection metric, or even a different routing protocol, to provide better QoS in the network?

As discussed above, the route selection metric added to the existing O-Batmand protocol did not add positively to the performance previously experienced in said networks. Furthermore, the implementation of a different routing protocol was not necessary, as the performance of O-Batmand was more than adequate.

The O-Batmand protocol was shown to have performed adequately in the indoor testbed, which was developed to closely mimic the environmental conditions found in rural setting areas. Having this in hand, the dissertation proceeded with its main objective. This was to research possible changes that can be made to the existing O-Batmand protocol. The changes were intended to improve the protocols performance, as seen from the data obtained from the tests conducted for the first research question above.

Reasoning behind the need for the changes is that O-Batmand is simple in its approach in determining the best links in the network. Also, QoS is needed in networks offering sensitive services such as VoIP which are not guaranteed by O-Batmand's approach. Furthermore, the simplistic approach does not use any of the metrics that WMNs use in rural environments. Lastly, the changes to the O-Batmand protocol (tailored for rural areas) helps the MP device fulfill the last few requirements sought from a viable alternative for wired landline communication technology, for rural areas. This viable alternative is one of the core reasons behind the use of the MP and the O-Batmand protocol in rural areas

Modification to the O-Batmand protocol were performed at the route classification and selection level. Essentially, the manner in which O-Batmand classifies and then chooses the best routes was modified. The modifications lead from the notion that the use of more metrics to determine link quality should yield better QoS in the network. With delay added to M-Batmand testing was conducted.

Testing of the M-Batmand routing protocol was conducted in the same manner as the tests for the first research question. This was done so that the two protocols could be compared and a solution to this research question obtained. The second research question looks at the improvements the route selection metric will add to the performance previously experienced in MP networks, using the O-Batmand protocol.

The results for the M-Batmand protocol demonstrated that it is suited to the task of routing packets inside a wireless network. It does this for both voice and data packets; however, the comparative performance demonstrates that O-Batmand, as is, is better than M-Batmand.

Reasons for why O-Batmand out-performed M-Batmand are many. One possible reason for the poor performance can be the structure of the B.A.T.M.A.N. algorithm. In this algorithm there is one section that allows for the merger of the delay calculations and that is when the algorithm is performing bi-directionality tests. This is only one situation available for updating delay values compared to several for the TQ values. One must remember that the path selection value used by the M-Batmand protocol is a composite metric called DATQ. In each round of calculations DATQ is updated according to the fresh TQ and delay values available. This seems to have created a situation where the frequency with which the algorithm updates the path selection value is much higher for the TQ value than it is for the delay value. This in turn creates internal confusion as the final path selection value, DATQ, fluctuates between TQ and delay influenced values. The resultant problem is that route selection changes are much higher in M-Batmand, and this disturbs the proper functioning of the protocol. This has translated to poor performance of the protocol observed in the results. This was clearly observed from the routing table screen output which showed link quality values changing dramatically between each screen output which were displayed every second.

Lastly, this research question asked if a different routing protocol would be necessary to improve the QoS in the MP. This is not the case as O-Batmand, shown by the data that helped answer the first research question, performs well on MP networks. It supports voice packets and has the capabilities to support data. However, it is important to note that due to the limited size of the testbed created in this dissertation the scope of the answer of this research question can only cover the laboratory study type experimentation style and not the real world networks out in the field. This provides an insight into what might be found in actual field studies which were mimicked by this dissertation as its primary goal. In order to answer this question for real world networks, experiments would have to be conducted on a few areas around the world, such as East Timor and Bo-Kaap, Cape Town, South Africa and is has been a success. This is outside the scope and goals of this dissertation.

6.2.3 How are different applications, such as, voice and data supported on the Batman Network?

This question deals with the possibility of progressively increasing the number of voice and data packets in the MP network, while observing the behaviour of the network as load in the network increases. This will show how the network responds to and thus supports both the voice and data applications as the load increases.

This research question was answered through the conducted experiments. The resultant outcome can be summarized as follows: different applications such as voice can be supported by the MP. Voice on the MP network are UDP packets 73 bytes long, other applications are represented by the 1500 byte long packets. However, the rapid increase in the number of hops traversed by the data negatively impacts the performance of the network therefore, reducing the networks performance.

This question aimed to see how well services on the MP network are supported. It is known that the MP supports VoIP and can also support data. The value of this research question is that answering it showed other potential uses of the MP network aside from providing VoIP service. This is a bonus feature of these devices as the added services can further assist development in rural areas. Although this is not an argument in this dissertation, the added services can help shorten the digital divide currently experienced in Sub-Saharan Africa. Providing a network that can support data means that at some stage other service such as access to Internet may be provided. Numerous other resources would then be available to the area being extremely beneficial and also help shorten the digital divide. This will be possible with the addition of one gateway node in the network, so the costs of this would be low. An obvious down side to this is that bandwidth in Africa, as with most ICT, remains fairly high, so even with access to this service its use would be minimal. However, in time costs will decrease and access to these services increase as the rural communities would then be capable of exploiting this service.

In order to obtain the data needed to answer this question, it was ideal to weave the tests needed to answer this question in with the tests that answered the first two questions. That means that the experiments conducted yielded data from the O-Batmand and the M-Batmand protocol versions and this then was used to provide insight to answer this question. The results of the tests pointed to the conclusion that, even though O-Batmand out-performed M-Batmand, both protocols can support both voice data and services with larger data packets up to 1500byte sized packets. However, the rapid increase in the number of hops traversed by the data causes performance degradations in the network, therefore, reducing the networks performance. Essentially, there is a limit to the data services that can be supported as the number of hops increase. This would require further testing with specific services in order to determine the threshold for each service; however, this is outside the scope of this study and forms part of future work. It is essential note that the data traffic can be sent through a network using UDP and not only TCP, and the data traffic used in the testbed was UDP.

6.3 Comparisons with Other Published Results

There have been works in published literature that have measured the performance of the O-Batmand routing protocol in practice. The authors of these works also created their own testbeds and measured the protocols performance in the testbed. It is interesting to compare their results to those obtained by this dissertation.

Abolhasan *et al.* [1] built a similar testbed to the one described in this dissertation. The similarities are that O-Batmand was loaded onto a Linux kernel and that a multihop testbed was used. The difference was that their network cards were set to use IEEE 802.11a which uses the 5GHz spectrum. They measure the network delay values across multihops. In their setup O-Batmand yielded delay values in milliseconds (ms). Shown here are values from one to four hops, 0.97, 2.15, 2.21 and 2.79 using ICMP packets. Their delay after the first hop increased by small margins which is different from the results found by this dissertation which show a rapid increase for both load sizes. However, it is difficult to establish valuable differences as different spectrums were used.

Other similar works such as [72] use other network parameters such as Number of Path Outages and Round Trip Delay which this dissertation did not use making it impractical to compare with. Works like these also present their results against increasing load instead of hops as is normal in performance measurements multihop network also adding to the difficulty in making comparisons.

Future work

Future works would include amongst many other possibilities the creation of many modified O-Batmand's each containing the implementation of one of the metrics in the list previously selected in chapter four of this thesis. Running tests on all these would reveal if, in fact, there is a metric that can improve the performance of the O-Batmand protocol. If so then the improvement will have brought QoS to the O-Batmand protocol.

Furthermore, another possible direction, once having successfully completed the above, is to port the modifications to the Batman-advanced (Batmand-adv) protocol. Batmand-adv is the O-Batmand protocol but works on the layer two of the OSI protocol stack. Previous research indicate that it performs better than the O-Batmand protocol. The question there would be if the improvements that benefit the O-Batmand can also benefit Batmand-adv.

REFERENCES

[1] Abolhasan, M., Hagelstein, B., and Wang, J. Real-world performance of current proactive multi-hop mesh protocols. In *Communications, 2009. APCC 2009. 15th Asia-Pacific Conference on*, IEEE, pp. 44–47.

[2] Adeyeye, M., and Gardner-Stephen, P. The village telco project: a reliable and practical wireless mesh telephony infrastructure. *EURASIP Journal on Wireless Communications and Networking 2011*, 1 (2011), 1–11.

[3] Aichele, C., Lindner, M., and Neumann, A. B.a.t.m.a.n status report, status report 2007.

[4] Akyildiz, I., Wang, X., and Wang, W. Wireless mesh networks: a survey. *Computer networks 47*, 4 (2005), 445–487.

[5] ALGIERS, A. Governance in africas development: Progress, prospects and challenges.

[6] Allen, W., Martin, A., and Rangarajan, A. Designing and deploying a rural ad-hoc community mesh network testbed. In *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on* (2005), IEEE, pp. 4–pp.

[7] alvarion. `http://www.alvarion.com/products/product-portfolio/breezemax`, 2013. WiMAX base station prices.

[8] Aslam, N., Phillips, W., and Robertson, W. Composite metric for quality of service routing in olsr. In *Electrical and Computer Engineering, 2004. Canadian Conference on* (may 2004), vol. 2, pp. 759 – 762 Vol.2.

[9] Barolli, L., Ikeda, M., De Marco, G., Durresi, A., and Xhafa, F. Performance analysis of OLSR and BATMAN protocols considering link quality parameter. In *2009 International Conference on Advanced Information Networking and Applications* (2009), IEEE, pp. 307–314.

[10] Ben-David, Y., Fowler, S., and Hasan, S. Jaldimac–beyond simulation. Tech. rep., Department of Electrical Engineering and Computer Science University of California, Berkeley.

[11] Britton, M., and Coyle, A. Performance Analysis of the BATMAN Wireless Ad-Hoc Network Routing Protocol with Mobility and Directional Antennas1. Tech. rep., The University of Adelaide, 30 April 2011. `http://downloads.open-mesh.org/batman/papers/SANLAB`.

[12] Burness, L., Higgins, D., Sago, A., and Thorpe, P. Wireless lanspresent and future. *BT technology journal 21*, 3 (2003), 32–47.

[13] canadasatellite. `http://www.canadasatellite.ca/Thrane-Thrane-Sailor-700-VSAT-p/thrane-thrane-sailor-700-vsat.htm`, 2013. Thrane and Thrane Sailor 700 VSAT.

[14] Chen, S., and Ravallion, M. *The developing world is poorer than we thought, but no less successful in the fight against poverty.* World Bank Washington, DC, 2008.

[15] Chen, T., Tsai, J., and Gerla, M. Qos routing performance in multihop, multimedia, wireless networks. In *Universal Personal Communications Record, 1997. Conference Record., 1997 IEEE 6th International Conference on* (1997), vol. 2, IEEE, pp. 557–561.

[16] Chissungo, E., Blake, E., and Le, H. Investigation into batman-adv protocol performance in an indoor mesh potato testbed. In *Intelligent Networking and Collaborative Systems (INCoS), 2011 Third International Conference on* (2011), IEEE, pp. 8–13.

[17] Chissungo, E., Blake, E., and Le, H. Investigation into batmand-0.3. 2 protocol performance in an indoor mesh potato testbed. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists Conference on Knowledge, Innovation and Leadership in a Diverse, Multidisciplinary Environment* (2011), ACM, pp. 267–270.

[18] Chissungo, E., Blake, E., and Le, H. Performance comparison of batmand and batman-adv. 250–250.

[19] Chissungo, E., Le, H., and Blake, E. An electronic health application for disaster recovery. In *Proceedings of the 2010 Symposium on Information and Communication Technology* (2010), ACM, pp. 134–138.

[20] Couto, D., Aguayo, D., Bicket, J., and Morris, R. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks 11*, 4 (2005), 419–434.

[21] Demmer, M., and Fall, K. Dtlsr: delay tolerant routing for developing regions. In *Proceedings of the 2007 workshop on Networked systems for developing regions* (New York, NY, USA, 2007), NSDR '07, ACM, pp. 5:1–5:6.

[22] Draves, R., Padhye, J., and Zill, B. Comparison of routing metrics for static multi-hop wireless networks. *ACM SIGCOMM Computer Communication Review 34*, 4 (2004), 133–144.

[23] Ellis, F. *Rural livelihood diversity in developing countries: evidence and policy implications.* EMBRAPA-CPAF Amapa, 1996.

[24] Flauzac, O., Haggar, B., and Nolot, F. Self-stabilizing clustering algorithm for ad hoc networks. In *2009 Fifth International Conference on Wireless and Mobile Communications* (2009), IEEE, pp. 24–29.

[25] Friginal, J., de Andres, D., Ruiz, J., and Gil, P. Towards benchmarking routing protocols in wireless mesh networks. *Ad Hoc Networks* (2011).

[26] Galperin, H. Wireless networks and rural development: opportunities for latin america. *Information Technologies and International Development 2*, 3 (2005), 47–56.

[27] Garroppo, R., Giordano, S., and Tavanti, L. Experimental evaluation of two open source solutions for wireless mesh routing at layer two. In *Wireless Pervasive Computing*

(ISWPC), 2010 5th IEEE International Symposium on (Modena, Italy, May 2010), IEEE, pp. 232–237. BATMAN-adv.

[28] Garroppo, R. G., Giordano, S., and Tavanti, L. Implementation frameworks for ieee 802.11s systems. *Computer Communications 33*, 3 (2010), 336 – 349.

[29] Group, V. T. `http://www.villagetelco.org/2009/06/voice-over-b-a-t-m-a-n/`, 2010. An easy-to-use, scalable, standards-based wireless local DIY telephone company toolkit.

[30] Hodge, J. Tariff structures and access substitution of mobile cellular for fixed line in south africa. *Telecommunications Policy 29*, 7 (2005), 493–505.

[31] Huovila, T., Lassila, P., Manner, J., and Penttinen, A. State of the art analysis of wireless mesh technologies.

[32] ICASA. `http://www.icasa.org.za/corporate/Overview/tabid/56/Default.aspx/`, 2011. Independent Communications Authority of South Africa (ICASA).

[33] IETF. `http://tools.ietf.org/html/rfc791/`, 1981. request for commnet for the INTERNET PROTOCOL.

[34] Ikeda, M., De Marco, G., Barollif, L., and Takizawa, M. A bat in the lab: Experimental results of new link state routing protocol. In *Advanced Information Networking and Applications, 2008. AINA 2008. 22nd International Conference on* (2008), IEEE, pp. 295–302.

[35] Ikeda, M., Kulla, E., Hiyama, M., Barolli, L., and Takizawa, M. Experimental results of a manet testbed in indoor stairs environment. In *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on* (2011), IEEE, pp. 779–786.

[36] International Telecommunication Union, R. I.-R. S.-. Definition of spectrum use and efficiency of a radio system. *Geneva, Switzerland* (May 2006).

[37] iran daily. Africa facing food price pressures. `http://www.iran-daily.com/1390/10/19/MainPaper/4145/Page/5/MainPaper_4145_5.pdf`, January 2012.

[38] Ishmael, J., Bury, S., Pezaros, D., and Race, N. Deploying rural community wireless mesh networks. *Internet Computing, IEEE 12*, 4 (2008), 22–29.

[39] ITU. statistics on information and communication technology penetration, accessibility and use. `http://www.itu.int/ITU-D/ict/statistics/ict/index.html`, January 2012.

[40] Johnson, D., and Lysko, A. Overview of the meraka wireless grid test bed for evaluation of ad-hoc routing protocols. In *Southern African Telecommunication Networks and Applications Conference (SATNAC07)* (2007), Citeseer.

[41] Johnson, D., Ntlatlapa, N., and Aichele, C. Simple pragmatic approach to mesh routing using BATMAN. In *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries* (Pretoria, South Africa, October 2008), CSIR, p. 10.

[42] Jun, J., and Sichitiu, M. The nominal capacity of wireless mesh networks. *Wireless Communications, IEEE 10*, 5 (2003), 8–14.

[43] Kaba, J., and Raichle, D. Testbed on a desktop: strategies and techniques to support multi-hop manet routing protocol development. In *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing* (2001), ACM, pp. 164–172.

[44] Karrer, R., and Pescape, A. 2nd generation wireless mesh networks: technical, economical and social challenges. In *Future Generation Communication and Networking (FGCN 2007)* (2007), vol. 1, IEEE, pp. 262–267.

[45] Kazemitabar, H., Ahmed Ali, S., Nisar, K., Md Said, A., and Hasbullah, H. A comprehensive review on voip over wireless lan networks. *Computer Science Letters 2*, 2 (2010).

[46] Kobo, H. Quality of service-aware routing for static mesh networks with mobile nodes. In *Proceedings of the 2010 Southern Africa Telecommunication Networks and Applications Conference (SATNAC)* (August 2010), satnac. `http://www.satnac.org.za/proceedings/2010/papers/poster/Kobo %20487.pdf`.

[47] Kowalik, K., and Davis, M. Why are there so many routing protocols for wireless mesh networks. In *Irish Signal and Systems Conference* (2006), pp. 1–5.

[48] Kulla, E., Hiyama, M., Ikeda, M., and Barolli, L. Comparison of experimental results of a manet testbed in different environments considering batman protocol. In *Intelligent Networking and Collaborative Systems (INCoS), 2011 Third International Conference on* (2011), IEEE, pp. 1–7.

[49] Kulla, E., Hiyama, M., Ikeda, M., and Barolli, L. Performance comparison of olsr and batman routing protocols by a manet testbed in stairs environment. *Computers and Mathematics with Applications* (2011).

[50] Kulla, E., Ikeda, M., Oda, T., Barolli, L., Xhafa, F., and Biberaj, A. Evaluation of a manet testbed in outdoor bridge environment using batman routing protocol. In *Proceedings of the 2012 IEEE 26th International Conference on Advanced Information Networking and Applications* (2012), IEEE Computer Society, pp. 384–390.

[51] Kulla, E., Ikeda, M., Oda, T., Barolli, L., Xhafa, F., and Takizawa, M. Multimedia transmissions over a manet testbed: Problems and issues. In *2012 Sixth International Conference on Complex, Intelligent, and Software Intensive Systems* (2012), IEEE, pp. 141–147.

[52] Lee, M., Zheng, J., Ko, Y., and Shrestha, D. Emerging standards for wireless mesh technology. *Wireless Communications, IEEE 13*, 2 (2006), 56–63.

[53] Lundgren, H., Lundberg, D., Nielsen, J., Nordstrom, E., and Tschudin, C. A large-scale testbed for reproducible ad hoc protocol evaluations. In *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE* (2002), vol. 1, IEEE, pp. 412–418.

[54] Mangala, J. Skepticism over the mo ibrahim prize for achievement in african leadership. *Bulletin of the African and African American Studies Program Grand Valley State University*, 35.

[55] Moreira, W., Lopes Gomes, R., and Gomes Abelem, A. A multiple-metric approach for routing in wireless mesh networks. In *World of Wireless, Mobile and Multimedia Networks Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a* (june 2009), pp. 1 –6.

[56] Murray, D., Dixon, M., and Koziniec, T. An experimental comparison of routing protocols in multi hop ad hoc networks. In *Telecommunication Networks and Applications Conference (ATNAC), 2010 Australasian* (31 2010-nov. 3 2010), pp. 159 –164.

[57] Neishaboori, A., and Kesidis, G. Sinr-sensitive routing in wireless 802.11 mesh networks. In *Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on* (29 2008-oct. 2 2008), pp. 623 –628.

[58] Nordstrom, E., Gunningberg, P., and Lundgren, H. A testbed and methodology for experimental evaluation of wireless mobile ad hoc networks. In *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on* (2005), IEEE, pp. 100–109.

[59] OpenMesh. `http://www.open-mesh.org/`, 2011. This page serves as development platform for a collection of tools to build free and open mesh networks.

[60] Oyelaran-Oyeyinka, B., and Lal, K. Internet diffusion in sub-saharan africa: A cross-country analysis. *Telecommunications Policy 29*, 7 (2005), 507 – 527.

[61] Raman, B., and Chebrolu, K. Experiences in using wifi for rural internet in india. *Communications Magazine, IEEE 45*, 1 (2007), 104–110.

[62] Ramanathan, R., and Rosales-Hain, R. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* (2000), vol. 2, Ieee, pp. 404–413.

[63] Rutgers, C. an introduction to igrp. *online: http://www. cisco. com/application/pdf/-paws/26825/5. pdf* (1991).

[64] Song, S. Open spectrum for development south africa case study. In *Association for Progressive Communications (APC)* (2010), APC.

[65] Srikanth, V., Jeevan, A., and Avinash, B. A review of routing protocols in wireless mesh networks (wmn).

[66] Staff, I. M. F. *Regional Economic Outlook: Sub-Saharan Africa, April 2008.* Intl Monetary Fund, 2008.

[67] Subramanian, L., Surana, S., Patra, R., Nedevschi, S., Ho, M., Brewer, E., and Sheth, A. Rethinking wireless for the developing world. *IRVINE IS BURNING* (2006), 43.

[68] Telco, V. `http://store.villagetelco.com/mesh-potatoes.html`, 2013. Mesh potato price.

[69] Time, O. ITU-T Recommendation G. 114," ITU-T. Tech. rep., ITU, 2000. Accessed 30 April 2011.

[70] Union, I. T. Itu-t p.800(08/06) : Methods for subjective determination of transmission quality. Tech. rep., ITU, 1996.

[71] Union, I. T. ITU-G Transmission system and media, digital systems. Tech. rep., ITU, 2011. Accessed 30 April 2011.

[72] Wang, J.-P., Hagelstein, B., and Abolhasan, M. Experimental evaluation of ieee 802.11s path selection protocols in a mesh testbed. In *Signal Processing and Communication Systems (ICSPCS), 2010 4th International Conference on* (December 2010), pp. 1 –3.

[73] XIE, B., JIANG, X., and ZOU, F. Investigation on the ahdemo mode in multi-hop wireless mesh network.

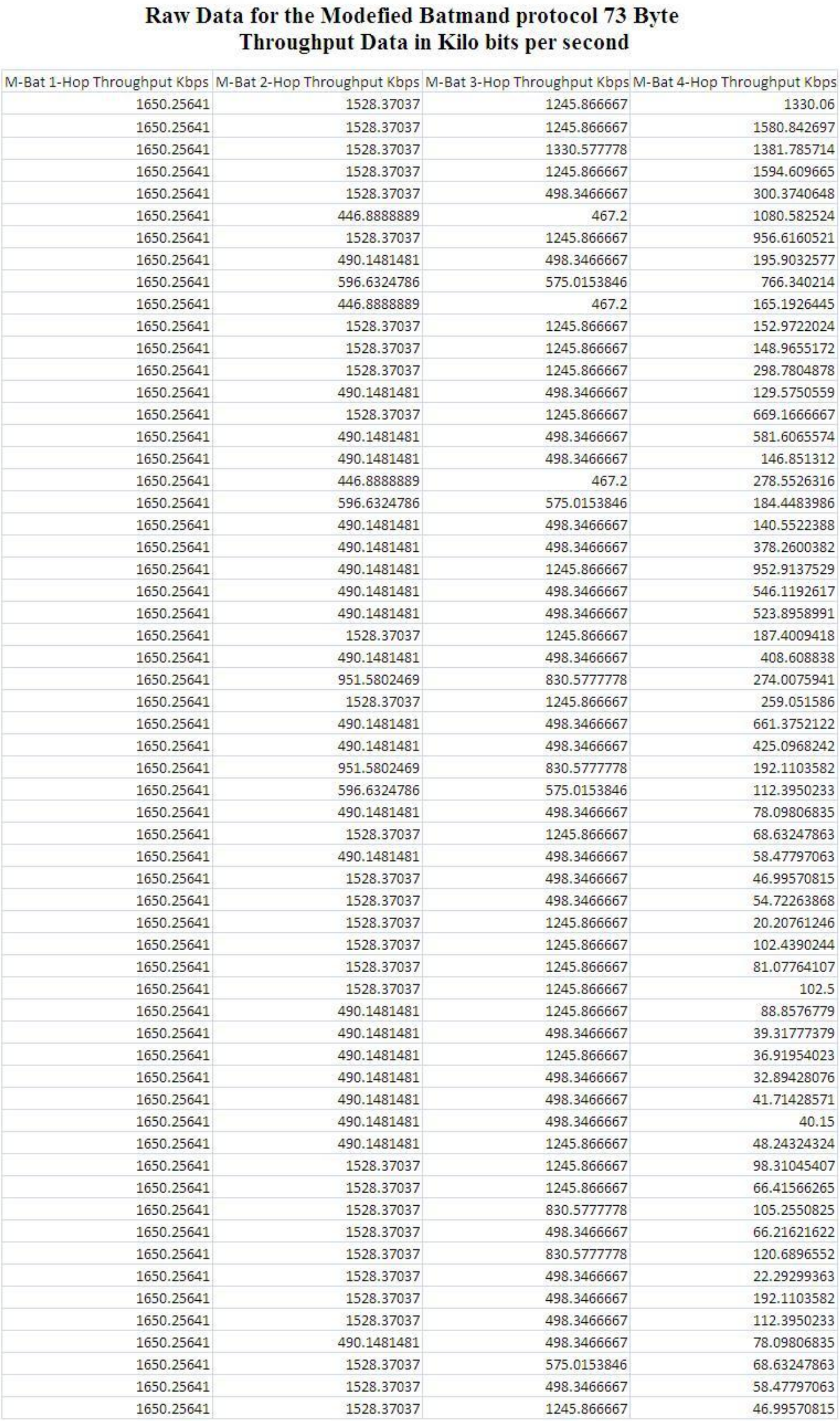
[74] Yeh, J., Chen, J., and Lee, C. Wlan standards. *Potentials, IEEE 22*, 4 (Oct/Nov 2003), 16–22.

APPENDIX

Appendix A

Appendix

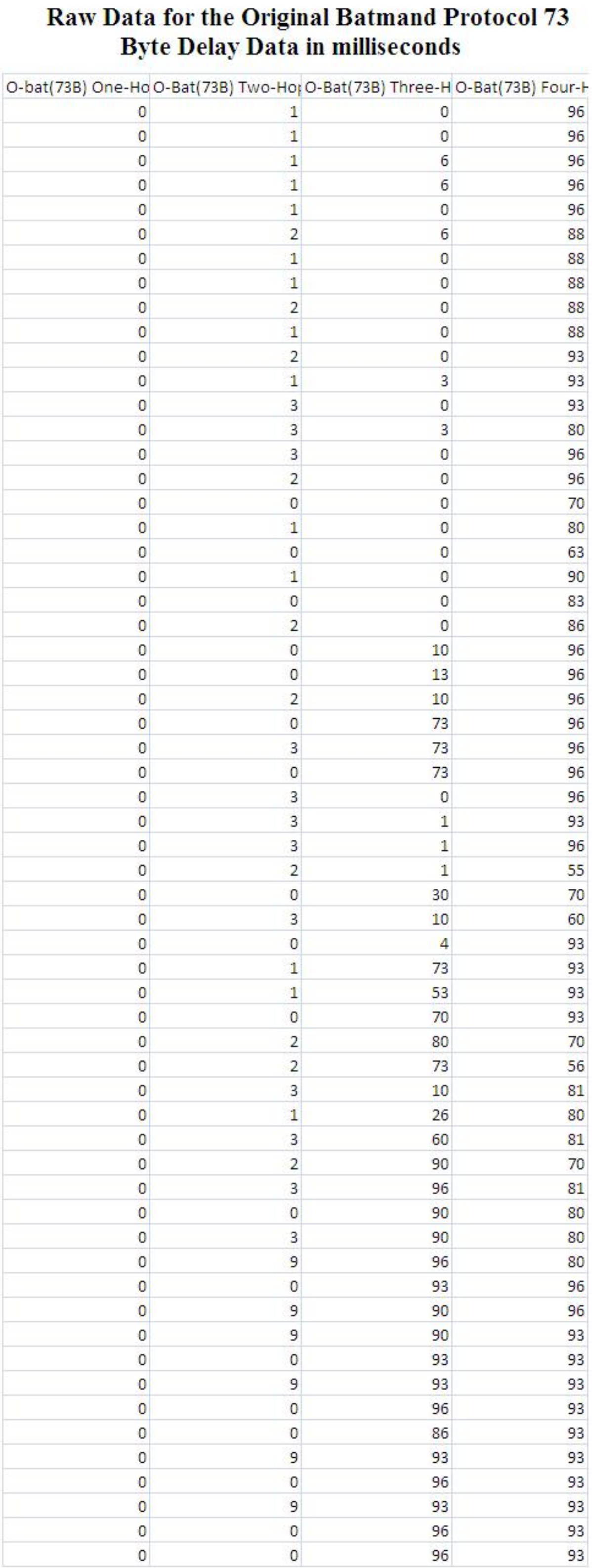
A.1 A Sample of the Raw Data Tables



Raw Data for the Modefied Batmand protocol 73 Byte
Throughput Data in Kilo bits per second

M-Bat 1-Hop Throughput Kbps	M-Bat 2-Hop Throughput Kbps	M-Bat 3-Hop Throughput Kbps	M-Bat 4-Hop Throughput Kbps
1650.25641	1528.37037	1245.866667	1330.06
1650.25641	1528.37037	1245.866667	1580.842697
1650.25641	1528.37037	1330.577778	1381.785714
1650.25641	1528.37037	1245.866667	1594.609665
1650.25641	1528.37037	498.3466667	300.3740648
1650.25641	446.8888889	467.2	1080.582524
1650.25641	1528.37037	1245.866667	956.6160521
1650.25641	490.1481481	498.3466667	195.9032577
1650.25641	596.6324786	575.0153846	766.340214
1650.25641	446.8888889	467.2	165.1926445
1650.25641	1528.37037	1245.866667	152.9722024
1650.25641	1528.37037	1245.866667	148.9655172
1650.25641	1528.37037	1245.866667	298.7804878
1650.25641	490.1481481	498.3466667	129.5750559
1650.25641	1528.37037	1245.866667	669.1666667
1650.25641	490.1481481	498.3466667	581.6065574
1650.25641	490.1481481	498.3466667	146.851312
1650.25641	446.8888889	467.2	278.5526316
1650.25641	596.6324786	575.0153846	184.4483986
1650.25641	490.1481481	498.3466667	140.5522388
1650.25641	490.1481481	498.3466667	378.2600382
1650.25641	490.1481481	1245.866667	952.9137529
1650.25641	490.1481481	498.3466667	546.1192617
1650.25641	490.1481481	498.3466667	523.8958991
1650.25641	1528.37037	1245.866667	187.4009418
1650.25641	490.1481481	498.3466667	408.608838
1650.25641	951.5802469	830.5777778	274.0075941
1650.25641	1528.37037	1245.866667	259.051586
1650.25641	490.1481481	498.3466667	661.3752122
1650.25641	490.1481481	498.3466667	425.0968242
1650.25641	951.5802469	830.5777778	192.1103582
1650.25641	596.6324786	575.0153846	112.3950233
1650.25641	490.1481481	498.3466667	78.09806835
1650.25641	1528.37037	1245.866667	68.63247863
1650.25641	490.1481481	498.3466667	58.47797063
1650.25641	1528.37037	498.3466667	46.99570815
1650.25641	1528.37037	498.3466667	54.72263868
1650.25641	1528.37037	1245.866667	20.20761246
1650.25641	1528.37037	1245.866667	102.4390244
1650.25641	1528.37037	1245.866667	81.07764107
1650.25641	1528.37037	1245.866667	102.5
1650.25641	490.1481481	1245.866667	88.8576779
1650.25641	490.1481481	498.3466667	39.31777379
1650.25641	490.1481481	1245.866667	36.91954023
1650.25641	490.1481481	498.3466667	32.89428076
1650.25641	490.1481481	498.3466667	41.71428571
1650.25641	490.1481481	498.3466667	40.15
1650.25641	490.1481481	1245.866667	48.24324324
1650.25641	1528.37037	1245.866667	98.31045407
1650.25641	1528.37037	1245.866667	66.41566265
1650.25641	1528.37037	830.5777778	105.2550825
1650.25641	1528.37037	498.3466667	66.21621622
1650.25641	1528.37037	830.5777778	120.6896552
1650.25641	1528.37037	498.3466667	22.29299363
1650.25641	1528.37037	498.3466667	192.1103582
1650.25641	1528.37037	498.3466667	112.3950233
1650.25641	490.1481481	498.3466667	78.09806835
1650.25641	1528.37037	575.0153846	68.63247863
1650.25641	1528.37037	498.3466667	58.47797063
1650.25641	1528.37037	1245.866667	46.99570815

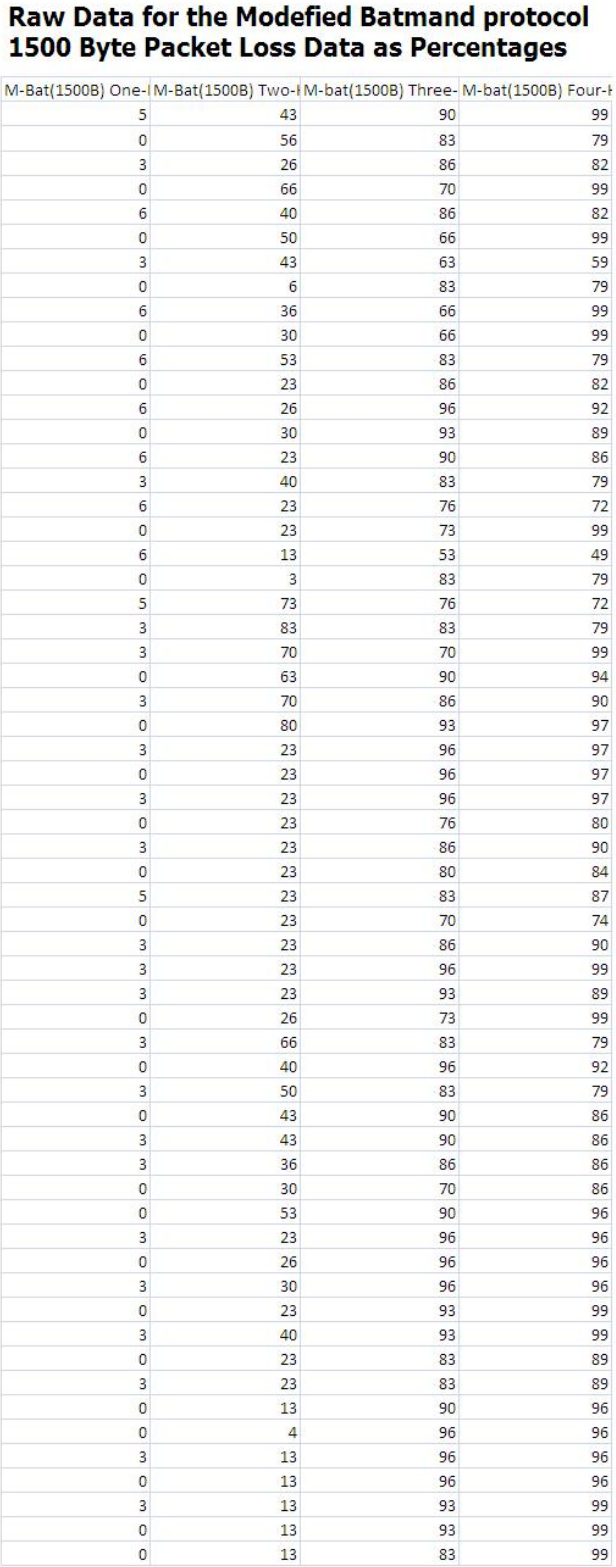
Figure A.1: Image of raw throughput data for M-Batmand in the 73byte case scenarios.



Raw Data for the Original Batmand Protocol 73 Byte Delay Data in milliseconds

O-bat(73B) One-Ho	O-Bat(73B) Two-Ho	O-Bat(73B) Three-H	O-Bat(73B) Four-H
0	1	0	96
0	1	0	96
0	1	6	96
0	1	6	96
0	1	0	96
0	2	6	88
0	1	0	88
0	1	0	88
0	2	0	88
0	1	0	88
0	2	0	93
0	1	3	93
0	3	0	93
0	3	3	80
0	3	0	96
0	2	0	96
0	0	0	70
0	1	0	80
0	0	0	63
0	1	0	90
0	0	0	83
0	2	0	86
0	0	10	96
0	0	13	96
0	2	10	96
0	0	73	96
0	3	73	96
0	0	73	96
0	3	0	96
0	3	1	93
0	3	1	96
0	2	1	55
0	0	30	70
0	3	10	60
0	0	4	93
0	1	73	93
0	1	53	93
0	0	70	93
0	2	80	70
0	2	73	56
0	3	10	81
0	1	26	80
0	3	60	81
0	2	90	70
0	3	96	81
0	0	90	80
0	3	90	80
0	9	96	80
0	0	93	96
0	9	90	96
0	9	90	93
0	0	93	93
0	9	93	93
0	0	96	93
0	0	86	93
0	9	93	93
0	0	96	93
0	9	93	93
0	0	96	93
0	0	96	93

Figure A.2: Image raw delay data for O-Batmand in the 73byte case scenarios.



Raw Data for the Modefied Batmand protocol
1500 Byte Packet Loss Data as Percentages

M-Bat(1500B) One-	M-Bat(1500B) Two-	M-bat(1500B) Three-	M-bat(1500B) Four-
5	43	90	99
0	56	83	79
3	26	86	82
0	66	70	99
6	40	86	82
0	50	66	99
3	43	63	59
0	6	83	79
6	36	66	99
0	30	66	99
6	53	83	79
0	23	86	82
6	26	96	92
0	30	93	89
6	23	90	86
3	40	83	79
6	23	76	72
0	23	73	99
6	13	53	49
0	3	83	79
5	73	76	72
3	83	83	79
3	70	70	99
0	63	90	94
3	70	86	90
0	80	93	97
3	23	96	97
0	23	96	97
3	23	96	97
0	23	76	80
3	23	86	90
0	23	80	84
5	23	83	87
0	23	70	74
3	23	86	90
3	23	96	99
3	23	93	89
0	26	73	99
3	66	83	79
0	40	96	92
3	50	83	79
0	43	90	86
3	43	90	86
3	36	86	86
0	30	70	86
0	53	90	96
3	23	96	96
0	26	96	96
3	30	96	96
0	23	93	99
3	40	93	99
0	23	83	89
3	23	83	89
0	13	90	96
0	4	96	96
3	13	96	96
0	13	96	96
3	13	93	99
0	13	93	99
0	13	83	99

Figure A.3: Image of raw percentage packet loss data for M-Batmand in the 1500byte case scenarios.

A.2 Raw Map of the Mdumbi Region

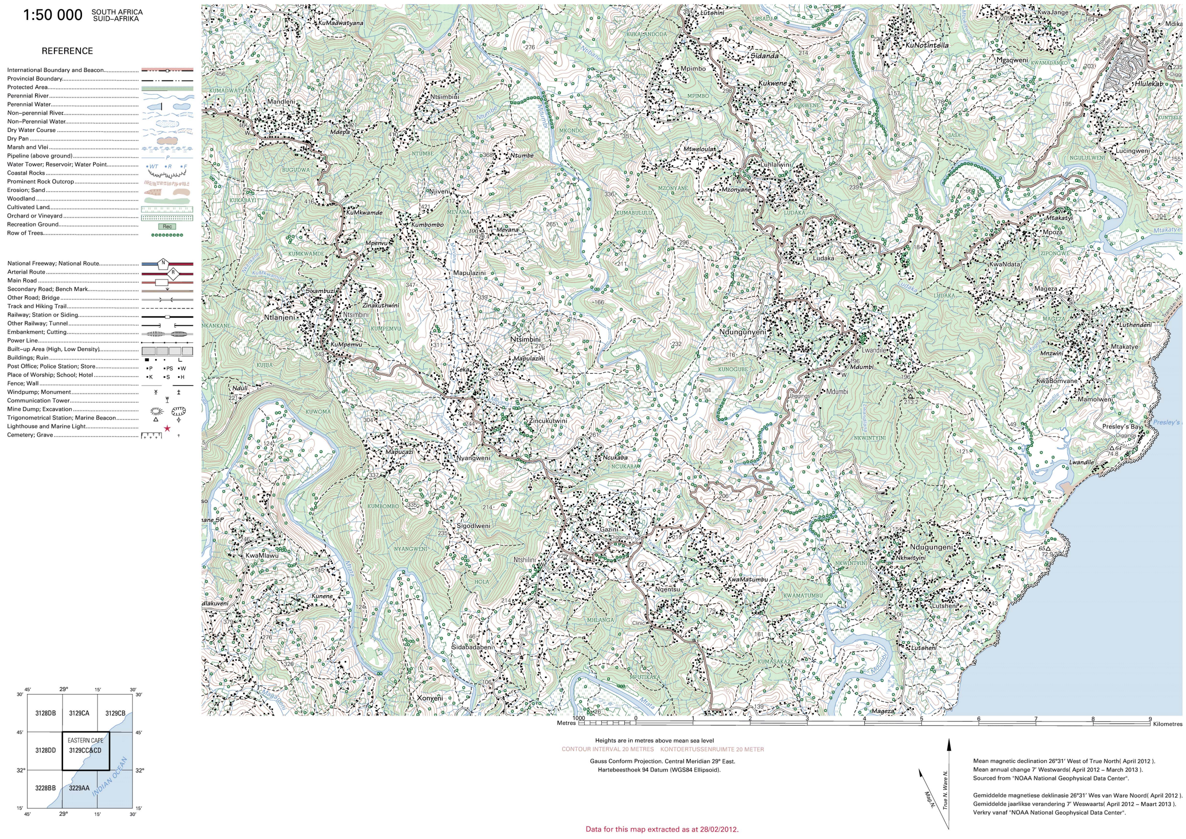


Figure A.4: Raw contour map of the Mdumbi Region, showing elevations and other details.

A.3 Mesh Potato (MP) communication and software architecture

On the MP one finds:

- Atheros AR2317 System-on-a-Chip (SoC) which combines an MIPS processor running at about 200MHz
- 802.11bg Wi-Fi.
- 16 Mb SDRAM
- LEDs and

- serial Flash

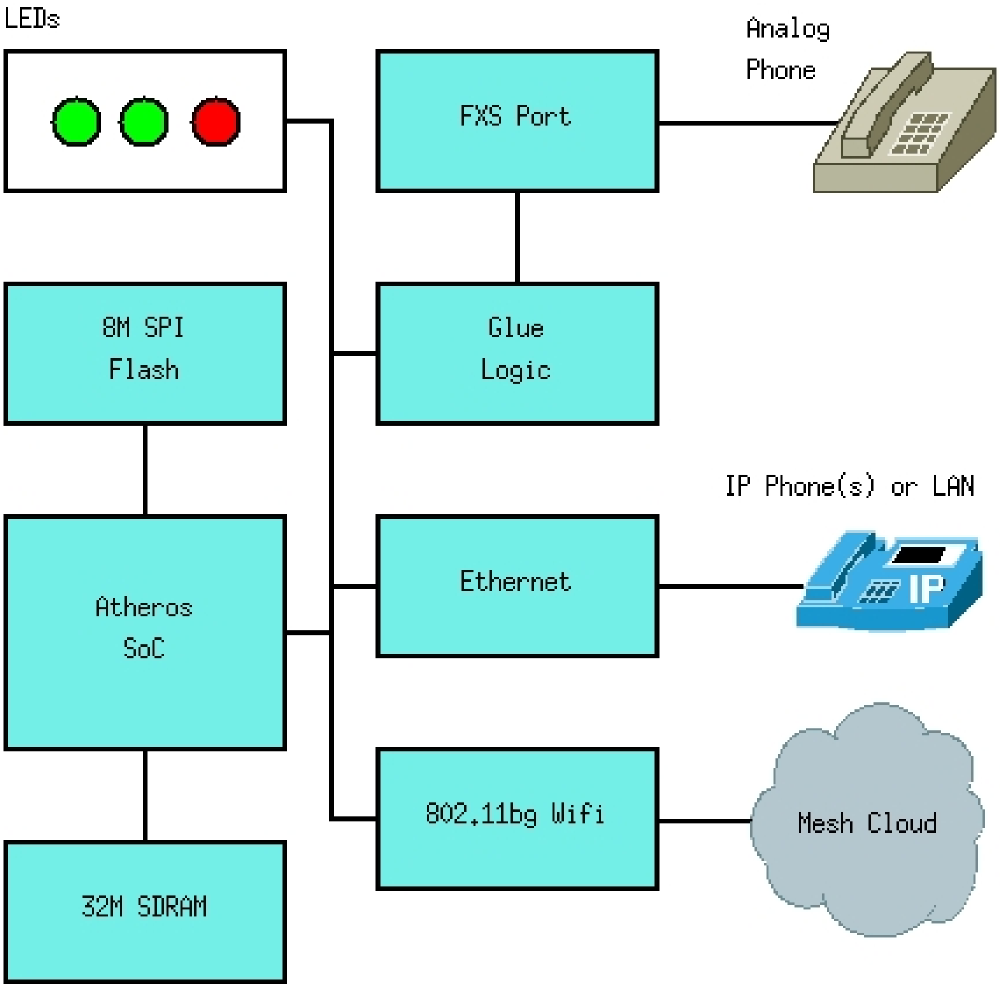


Figure A.5: Image showing the MP hardware architecture, (source [villagetelco.com])

A.4 Code Snippet

```

// rign buffer_c code , the file that joins TQ and delay to create DATQ
#include ”ring_buffer.h”
#include ”batman.h”

uint32_t rtt_min = 2000, rtt_max = 0;
/*
 * MSc: EC − Additonal code START
 *
```

```
 * The following code assumes that it is in this file that The Transmission Quality (TQ) value is
 * calculated when called specifically by one cprogram file -- originator.c L#172 and 173 then
 * ,183 and 184.
 *
 * Here we add the code that will deal with the received Delay
 * data found in the Ping_packet which was placed inside the bat_packet in paket.h
*/
uint8_t ring_buffer_delay(uint32_t curr_time, uint32_t *rtt_avg, uint32_t time_value)
{
  //printf(”MSc: EC ~ DEBUG: \n \t Inside ring_buffer.c L#38\n”);

   uint8_t tmp_delay;
   uint32_t rtt;
   uint32_t tSent; // the time when each packet was sent in millisecond

     //get rtt in millisecond
     rtt =(uint32_t)(( curr_time - time_value )  +
    + ( curr_time - time_value ) + 0.5);

     //get tSent in millisecond
     tSent =(uint32_t)(( time_value - curr_time ) +
    + ( time_value - curr_time ) + 0.5);

  // m not too sure why we need this but it forms part of the delay calculations

     if (rtt < rtt_min){
       rtt_min = rtt;
     }
     if (rtt > rtt_max){
       rtt_max = rtt;
     }


     *rtt_avg = (*rtt_avg) + rtt; // phase one of delay calcs it is conclude in the  ring_buffer_avg() function

     return tmp_delay = (uint8_t)*rtt_avg;
      //printf(”MSc: EC ~ DEBUG: \n \t ring_buffer.c L#67 return\n”);
      //return (uint8_t)0;

}
```

```c

void ring_buffer_set(uint8_t tq_recv[], uint8_t *tq_index, uint8_t value)
{
  tq_recv[*tq_index] = value;
  *tq_index = (*tq_index + 1) % global_win_size;
}

uint8_t ring_buffer_avg(uint8_t tq_recv[], uint32_t temp_delay)
{
  uint8_t *ptr, DATQ;;
  uint16_t count = 0, i = 0;
  uint32_t sum = 0;
  uint32_t Max_Delay =  400, N_Delay=0; //MSc: EC - according to the ITU-T G.114,
      400 ms is max acceptable delay

  ptr = tq_recv;

  while (i < global_win_size) {

    if (*ptr != 0) {
      count++;
      sum += *ptr;
    }

    i++;
    ptr++;

  }

  if (count == 0)
    return 0;

  //MSc: EC - we add the effects of delay to the calculated TQ value
  /*last phase of delay calculation as per the open source code
   * from which this was taken.
   */
  temp_delay =  ((count - 1) * temp_delay) / count;
  N_Delay = (temp_delay/Max_Delay) * TQ_MAX_VALUE; // MSc: EC - normalizing

  //printf(”MSc: EC ~ DEBUG: \n \t Inside ring_buffer.c L#110 return\n”);
  return DATQ = (uint8_t)( ((uint8_t)(sum / count)) - (temp_delay) ) ;// MSc: EC
    - addind the delay effects to TQ

  //return (uint8_t)(sum / count);
```

```
}
```