School of Engineering and Design

Brunel University

Flexible Cross Layer Design for improved Quality of Service in MANETs

A thesis submitted for the degree of Doctor of Philosophy

By

Ilias Kiourktsidis

2011
LONDON, UK

Table of Contents

List of Figures

List of Tables

Abbreviations

Abbreviation	**Description**
ABR	Associativity Based Routing
ACK	Acknowledgement
AODV	Ad hoc On-Demand Distance Vector
AP	Access Point
BER	Bit Error Rate
BN	Bayesian Network
BNR	Bayesian Network Routing
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access
CoS	Class of Service
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
CTS	Clear to Send
DCF	Distributed Coordination Function
DS	Deterministic Sampling
DSARP	Delay-Sensitive Adaptive Routing Protocol
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
ELFN	explicit link failure notification
ESMP	Evolvable Self Modifying protocol
ETX	Estimated number of Retransmissions
FFD	Full Function Devices
FHSS	Frequency Hopping Spread Spectrum
FTE	Frame Transmission Efficiency
GA	Genetic Algorithm
GF	Greedy Forwarding
GPS	Global Positioning System
GTS	Guaranteed Time Slot
HARP	Hybrid Ad hoc Routing protocol
HCF	Hybrid Coordination Function
HWMP	Hybrid Wireless Mesh Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ISM	Industrial, Scientific and Medical

KR	Knowledge Representation
LAN	Local Area Network
LFI	Link Fragmentation and Interleaving
LLE	Link Life Expectation
LQI	Link Quality Indication
MAC	Medium Access Control
MANET	Mobile Ad Hoc Networks
MIMO	Multiple Input Multiple Output
ML	Machine Learning
OFDM	orthogonal frequency division multiplexing
OSI	Open Systems Interconnection
PAN	Personal Area Network
PCF	Point Coordination Function
PDF	Probability Density Function
QoS	Quality of Service
RCP	rate control protocol
RFD	Reduced Function Devices
RREP	Route Reply
RREQ	Route Request
RRER	Route Error
RTCP	real-time transport protocol
RTO	Retransmission Timeout
RTP	real-time protocol
RTS	Request to Send
RTT	Round Trip Time
SA	Simulation Annealing
SMP	Self Modifying protocol
SNR	Signal to Noise Ratio
SRS	Stochastic Remainder Sampling
SSA	Stability-Based Adaptive Routing
TCP	Transport Control Protocol
TDMA	Time Division Multiple Access
UDP	User Datagram Protocol
UWB	Ultra Wideband
VBR	Variable Bit Rate
WPAN	Wireless Personal Area Network
WUSB	Wireless Universal Serial Bus

Acknowledgements

I owe a special debt of gratitude to those who contributed to the completion of this work. First of all, I would like to thank my supervisors Dr Marios Hadjinikolaou and Dr Jonathan Loo, for their guidance through my PhD. Their continuous support for this scientific work has proved determinant.

I would like to thank my friends Dr. Grigorios Koulouras and MSc. Dimitrios Stavroulakis for their unselfish encouragement and support.

I would like to thank the National Foundation of Scholarships (I.K.Y.) for the financial support during my studies.

Last but not least my mother Kiriaki and my sister Triantafillia for their unending support and love that enabled me to fulfil this work.

Abstract

Mobile Ad hoc Networks (MANETs) are becoming increasingly important because of their unique characteristics of connectivity. Several delay sensitive applications are starting to appear in these kinds of networks. Therefore, an issue in concern is to guarantee Quality of Service (QoS) in such constantly changing communication environment. The classical QoS aware solutions that have been used till now in the wired and infrastructure wireless networks are unable to achieve the necessary performance in the MANETs. The specialized protocols designed for multi-hop ad hoc networks offer basic connectivity with limited delay awareness and the mobility factor in the MANETs makes them even more unsuitable for use. Several protocols and solutions have been emerging in almost every layer in the protocol stack.

The majority of the research efforts agree on the fact that in such dynamic environment in order to optimize the performance of the protocols, there is the need for additional information about the status of the network to be available. Hence, many cross layer design approaches appeared in the scene. Cross layer design has major advantages and the necessity to utilize such a design is definite. However, cross layer design conceals risks like architecture instability and design inflexibility. The aggressive use of cross layer design results in excessive increase of the cost of deployment and complicates both maintenance and upgrade of the network. The use of autonomous protocols like bio-inspired mechanisms and algorithms that are resilient on cross layer information unavailability, are able to reduce the dependence on cross layer design. In addition, properties like the prediction of the dynamic conditions and the adaptation to them are quite important characteristics.

The design of a routing decision algorithm based on Bayesian Inference for the prediction of the path quality is proposed here. The accurate prediction capabilities and the efficient use of the plethora of cross layer information are presented. Furthermore, an adaptive mechanism based on the Genetic Algorithm (GA) is used to control the flow of the data in the transport layer. The aforementioned flow control mechanism inherits GA's optimization capabilities without the need of knowing any details about the network conditions, thus, reducing the cross layer information dependence. Finally, is illustrated how Bayesian Inference can be used to suggest configuration parameter values to the other protocols in different layers in order to improve their performance.

1 Introduction

1.1 Overview

Nowadays, the communication between devices is mainly taking place through infrastructure computer networks. In the near future, the necessity of communication between devices, without the existence of a network infrastructure, will arise. The devices that need infrastructure-less networking are usually mobile devices, which can exist in areas without network coverage. Mobile Ad hoc Networks (MANETs) are those kinds of networks that provide communication between mobile devices, without the need of a pre-existing network.

As computers are still becoming smaller and more energy efficient [Keyes, R.W. (2008)], the number of mobile devices is growing day by day and along with the recent technological advances in wireless communications, wireless and mobile ad hoc networks are expected to play an important role in applications in occasions that wireless access to wired backbone is either ineffective or impossible. Mobile ad hoc networks are composed of a set of mobile nodes communicating through wireless channels, without the help from any fixed preinstalled infrastructure. MANETs can be useful whenever the intense utilization of a network is needed like emergency situations, rescue operations, security, military operations etc. The basic difference of MANETs to other ad hoc network types is the frequent topology changes caused by node mobility. In addition, the wireless communication medium has variable and unpredictable characteristics. The signal strength and propagation delay may vary. Furthermore, mobile stations limited capabilities and battery consumption raise additional problems. All these characteristics introduce new problems that are not encountered in the conventional wired and wireless networks. The operation of

MANETs in an optimized way and use of the communication and computation resources in a balanced way is a challenging task. The protocols operating in a MANET environment must be designed to be self-configured, adapt to the network condition changes and the target requirements must be posed from the applications. Fig. 1 depicts a snapshot of a MANET topology.

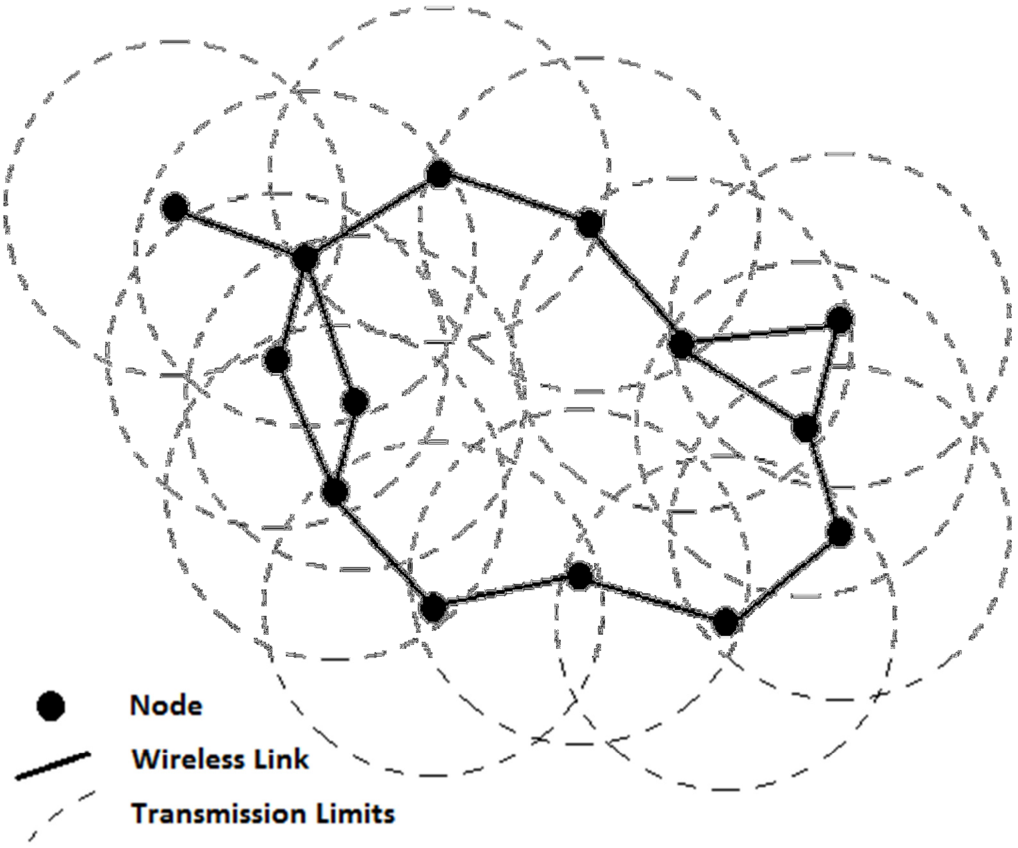


Fig. 1. Typical MANET topology.

1.2 Motivation

Hence, the design of protocols and algorithms for MANETs poses new and interesting research challenges, some of them particularly to mobile ad hoc networks. Most of the research in this emerging technology targets on physical and networking areas of the MANETs. However, in most of these studies, it has been neglected to target real end to end support on Quality of Service (QoS). Network layer solutions have been researched extensively, but most of the proposed work manages to solve only parts of the problems. Therefore, a combination of them can increase the advantages and decrease the disadvantages of each solution and result in a more flexible mechanism.

In addition, the belief that Transport layer can be utilized more in order to provide better QoS, has motivated the author to look for solutions, which either standalone or combined with other lower layer mechanisms can improve the overall performance. Furthermore, a strong motivation for the author is the promising role of the MANETs in interesting future scientific and commercial applications.

Finally, the majority of the research efforts focused on cross layer design approaches are proposing solutions which are heavily dependent on the cross layer information. Cross layer design has both benefits and drawbacks [Kawadia, V. (2005)]. The aggressive use of cross layer design must be avoided. The author motivated by the absence of mechanisms that are able to reduce the dependence on cross layer design, hence, utilize the benefits and eliminate the drawbacks.

1.3 Scope of the Thesis

In light of this, the scope of this thesis is to focus on several aspects of mobile ad hoc networking and, in particular, on methods for improving routing selection and flow control in network and transport layers respectively. The final purpose of the improvements in network and transport layers is the enhancement of QoS that the application experiences.

The target is to develop mechanisms that are able to work standalone with minimum dependence on the other layers. The reason for that is the variety of technologies in MAC and physical layers. A protocol that depends on specific low layer characteristics may not be able to co-operate with many of these technologies. The author believes that the modularity in the protocol stack must be preserved in order to maintain its flexibility, especially in a wireless environment, where the physical and MAC layers are always changing. In case that interaction with other layers is possible, then the additional information is used to improve the performance of protocols.

In transport layer a Genetic Algorithm (GA) is proposed to regulate the data rate. The scheme that is proposed can operate without the need of any information from lower layers. GA has the natural ability to auto-adapt in changing environments by finding the most appropriate configurations. This ability is inherited to the transport layer protocol and it adapts to network conditions.

Unfortunately, network layer and specifically the routing mechanism in MANETs require additional information and cross layer design cannot be avoided. In this thesis a flexible mechanism is presented that is able to operate and produce results even though there is partial lack of cross layer information, due to insufficient data or protocol incompatibilities. A Bayesian Network (BN) is used to model the network path and to provide inference about the applicability for communication of each discovered path. Because of the BN properties this protocol has the ability to adapt to the cross layer information availability.

Finally, a cross layer design between the two proposed algorithms is introduced. Again, the coexistence of the two protocols is not obligatory for their operation, but when they do coexist, the performance can be increased.

1.4 Contributions of the thesis

This thesis has focused on the improvement of QoS in MANETs by proposing two novel mechanisms for two different protocol layers. During the attempt to design the proposed protocols, several contributions to knowledge emerged.

The design principles of both proposed protocols introduced an architecture philosophy that targets to minimize the cross layer information dependency.

Furthermore, the results derived from the evaluation of one of the proposed protocols shown that different variable size of packets and packet rate can experience different QoS in terms of end to end delay, jitter of delay and throughput in MANETs.

Another significant finding from the experimental results is that GA is able to cope effectively with the speed and the amount of changes in a MANET environment. Thus, it is suitable for use in these kinds of environments.

One more contribution is the design of a statistical model that describes a path in MANETs. This model can be used to confirm several existing assumptions noted in literature and to discover new characteristics for MANETs.

This model seems to be promising, since it can be used for routing decisions, as well as, to suggest configuration solutions to the other layers.

1.5 Thesis Outline

The present thesis is organized as follows.

In Chapter 2, an extensive literature review on the work that has been done in MANETs is presented. The review is organised on three main parts, work that has been done in MAC layer, in network layer and in transport layer. Various protocols are presented and further organised in sub categories. Also discussions about the cross layer architecture and the QoS are included in order to give to the reader a more complete representation of the complexity of MANET environment and the need of new approaches to the problem.

In Chapter 3 the Evolvable Self Modifying Protocol (ESMP) that uses a GA-based flow control algorithm is presented. The protocol implementation is described and the performance results are presented as well. Furthermore, the complexity of the MANETs environment is described and the need of an efficient heuristic algorithm is highlighted.

Chapter 4 describes the routing protocol mechanism called Bayesian Network Routing (BNR). Most of the work in the network layer is concentrated on the statistically based routing decision algorithm. The protocol details and the statistical data collection procedure are described. The Bayesian model that is created is used to show how the several network metrics are related and verifies the behaviour that is described in the literature. Finally, experimental results are presented.

In Chapter 5 a cross layer design between BNR and ESMP is tested and presented in order to improve the initial population selection in ESMP. The results of this design are presented. The integration of BNR and ESMP is also presented along with the overall performance results.

Finally, chapter 6 draws together results and conclusions from the previous parts of the thesis and discusses possible directions for future research.

2 Literature Review and Problem Discussion

2.1 Introduction

There has been a lot of research on protocols that aim to deliver the QoS that applications need in a multi-hop ad hoc network. Those protocols are operating mostly in the MAC and network layers. In this chapter many solutions are introduced, organized in categories according to the protocol layer that they are targeting. They are further subcategorized whenever it is necessary. The solutions that are presented here are designed for mesh networks in general and the majority of them are designed especially for MANETs. The advantages and weaknesses of the proposed solution and the suitability for MANET's environment are examined. In this chapter we are trying to present the solutions and also to help the reader to realise the complexity of the environment in contrast to the traditional networks. That complexity forces the researchers to look for new non traditional and more advance algorithms and mechanisms.

Firoiu et al. [Firoiu, V (2002)] provide a comprehensive survey of available QoS mechanisms from the technical and the business perspective. Zhu et al. [Zhu, H (2004)] classified the QoS solutions to the following categories: link adaptation in physical layer, channel access coordination and admission control strategies in the MAC, and higher layers. In addition, no complete QoS solution can be performed within a single protocol layer [Firoiu, V (2002)],[Zhu, H (2004)].

The research areas in Ad hoc networks, like in most of network types, are related mainly with the following areas.

- QoS
- Path metrics

- Routing
- Security
- Self configuration
- Multicast
- Cooperation and incentive mechanisms
- Load balancing
- Energy conservation

Some of them are further analysed in the following chapters.

2.2 MANETs

MANETs is a special type of multi-hop ad hoc networks. Ad hoc networks are the type of networks where two or more mobile devices can form a network on the fly and exchange information. Classical example of ad hoc networks is when users activate the Bluetooth interface to connect with other devices in the same area and exchange data without the need of a preinstalled network infrastructure in the area. Multi-hop ad hoc networks are going a step further by allowing nodes that are not reachable by their radios to communicate by using intermediate nodes. The intermediate nodes are relaying the data packets on behalf of the sender node.

Other special types of ad hoc networks are the mesh networks wireless, sensor networks and vehicular networks. Usually mesh networks and sensor networks are not mobile or their mobility is very low thus their characteristics and requirements are very different than the MANETs who are designed for high mobility nodes. On the other hand, vehicular networks are a special type of MANET where the mobility pattern of the mobile devices is very specific and brings special characteristics that need special solutions.

The quality or even availability of a mobile connection relies on the position of the mobile transceiver relative to the fixed transceiver’s antenna. In order to cover a bigger area in an efficient way, a network looks like a honeycomb, where every antenna covers one of the cells. Such a network is likely to have dark areas because in the real world it has to coexist with geographical barriers such as buildings, trees, roads and vehicles. In some cases, it is not even practical to try to build such a network, if not for any other reason, because it is inefficient or impossible to supply enough energy to both uplink and downlink transmitters. Satellite communications can be useful but cannot be considered as the ideal solution to this problem, due to limited capacity, the high energy consumption in mobile device and possible situations where the mobile device cannot have a line sight to the satellite.

In a cellular network the device is subservient to the network; even if two devices are used right next to each other they cannot communicate even if just one of them cannot make connection to a fixed access point. The inspiration from the internet architecture where two communication peers have equalled roles can give to wireless networks the same flexibility that internet enjoys. Another aspect of flexibility on the internet is the dynamic way that information flows on the internet. The nodes on the internet that have the role of routers continually learn what connections other nodes have available. In mobile networks perhaps, this information exchange should happen at a higher frequency because the changes in the network topology and the quality of the connections are varying faster than in internet topology, but the method is suitable. In that way, the topology and connections will constantly change and adapt in the conditions and will not be a fixed and highly structured network topology as in a cellular network.

If MANETs follow the same principles that internet has and are described above, then it is easy to get a good idea of how a MANET will eventually be of use. Mobile devices in a MANET form a peer-to-peer communication network by using any channel available to reach the destination. Other nodes are used to relay the traffic when direct communication is not possible. When a connection to another network is required then the access point of the cellular network can be used as a getaway and can be reached from the mobile devices through other relaying nodes if it is necessary. This could generate interesting new business models, because the end users could under certain conditions bypass commercial networks like the case the devices are close to each other.

Ad hoc and MANET technologies are still immature, and developments are promising. Their final impact relies quite a lot on the way in which the telecom world and the hardware suppliers will act on its appearance on the scene and may feel threatened on losing some of the control in the area of data exchange between regular users in an urban environment.

In any case mobile ad hoc networking will find its way where mobile fixed network infrastructure is not developed or destroyed and in cases where the existing networks are not suitable or access to them is restricted. Anywhere a small or big temporary network is required, ad hoc networks can give the solution. In areas that need disaster relief from earthquakes, hurricanes etc for search and rescue missions when the infrastructure is wiped out or there is not access, a quickly deployed ad hoc network can be used. Ad hoc networks also can be useful in military applications when the units need to communicate in a hostile environment with no available infrastructure.

2.3 Quality of Service in MANETs

QoS refers to satisfaction of certain requirements in a communication session, in terms of a set of link or path performance constraints. Some of the performance constraints can be the end-to-end delay, delay jitter, bandwidth, packet lost percentage, etc. Another type of QoS metrics can be the power consumption, security, etc although delay and bandwidth are considered the most common QoS metrics, especially for user experience in multimedia applications. QoS metrics in multi-hop networks can be concave, additive or multiplicative. For example, bandwidth is a concave metric because the link across the path that has the less bandwidth will be the bottleneck of the entire route. Delays in each link will add up to the total end-to-end delay, thus delay is an additive metric. Finally, the probability that a packet will be lost is a multiplicative metric; since the reliabilities of each link must be multiplied together in order to find the total probability that a packet will be lost on the way to the destination node. It can be understood that these metrics are not necessarily independent since end-to-end delay can be affected by the transmission rate or the link reliability. QoS routing is an important part in the design of a QoS system.

One type of routing protocols that is targeting to ensure QoS in a network path is the resource reservation protocols. Often there is a misconception between QoS-Based routing and resource reservation. Resource reservation protocols in fact provide a mechanism to request and reserve resources along a network path but they do not discover an adequate path that can accommodate the QoS requirements. On the other hand, QoS-Based routing objectives are destined to discover a communication path that can provide the required QoS guarantees and balance the network resources. Due to the dynamic nature of MANETs, it is hard to discover a path that can be adequate for the same amount of time that the communication session lasts, at the same time it is hard to reserve any resources that may not be available the moment the communication will take place.

QoS performance can be either deterministic or statistically guaranteed. In most types of networks like wired networks and wireless networks with centralized control where everything is "under control", it is easy to guarantee QoS in a deterministic manner because, most of the times, there is a certainty about the availability of the network resources. In contrast, in MANETs, as aforementioned, there are no guarantees that the network conditions along with the network resources will remain the same for the time needed. So, statistical QoS guarantees are a more realistic scenario in this type of environments.

By referring to network resources, we mean anything that is required in order to carry out a communication and which is consumed during that. Some of the network resources that exist in a MANET and that affect the performance of the QoS metrics are the following:

- Link capacity. The capacity of a link depends on the channel capacity and the fraction of the channel capacity that the link is allowed to or can use. In a congested area of the network, where many nodes are located that need to transmit packets, the fraction of the channel capacity that the link can use to transfer data is going to be limited. This will have a direct impact on the throughput and the delay.
- Node processing time. Usually mobile nodes have limited processing power without dedicated hardware for the various operations that have to complete. When a node has a lot of data to process, it takes time to process the packets that it receives and complete the required operations before retransmitting them.
- Node buffers. Any communication device has buffers to store packets waiting for processing or transmission. When the buffers are full, then the packets are dropped. Even when the buffers are half full, then the packets will experience

a delay waiting in these buffers. The more the usage of the buffers of a node, the more the delay for the packets that use this node as an intermediate hop.

- Node battery power. Mobile nodes are battery powered and it is well known that batteries do not last long. In addition, the more processing and retransmission operations a node has to do, the more battery power will consume. A node with low energy reserves may reduce the transmitting power and the radio coverage or it may fail to operate at all. In order to keep the network on balance and keep node failures to minimum, sometimes it is wiser to use routes that do not involve nodes with low battery power. Sometimes, for example in vehicular networks, battery life is not an issue.

The availability of most of the resources from the list above, in a specific moment depends on how much traffic a node and or the neighbor nodes are serving at that moment. A general rule that applies is that the best route is the one that intermediate nodes and links have most of their resources available.

But there are other factors that affect the QoS performance. Some statistical evidence about the node, channel status statistics in the near past and information regarding the whole route can be useful for the route selection process.

- Node mobility. The mobility of a node is a parameter that depends on the current speed of the node, the direction of the movement and the fact on how often the node is usually moving. So statistical data on the movement of each node can be useful to determine how mobile a node is for the last minutes and predict how mobile is going to be in the near future.
- Channel characteristics. Statistical data on the channel quality can be gathered by the nodes from previous packet exchanges with the neighbor nodes. Bit error rate, collisions, mean time waiting for transmission, signal to noise ratio etc. These and many other parameters can give a good estimation on how reliable is the channel in a specific area and how reliable is a link with a specific neighbor node.

To sum up, the challenges that have to be faced by a QoS aware protocol in a MANET, constitutes of the varying physical link properties where link degradation can occur at any time, the medium access issues in the shared environment, network topology changes due to mobility of the nodes and power consumption in the mobile nodes. Each of these challenges has to be confronted separately or as a whole in one or more protocol layers.

- Application Layer: Because the application layer is very high in the stack, it is very far from the problems which degrade the performance and subsequently very far from the solutions. Application Layer has to request a particular Class of Service (CoS) from the lower layers along with the request for establishing communication with another peer. After the communication is established, application layer adaptively changes its CoS in case the network cannot satisfy the requirements.
- Transport Layer: According to the CoS from the upper layer differentiates the services and maps the different classes of service to different routing protocols. The flow control can be also configured in a way that helps to increase QoS.
- Network Layer: Again it differentiates the packets by their CoS and it uses different routing mechanisms for each of them. It is also responsible to find routes that promise high enough QoS and is responsible to change the route when the requirements are not satisfied.
- MAC Layer: Uses differentiation again for different CoS by using different protocols and configurations so that faster, better, and assured service can be delivered to the QoS sensitive applications.
- Physical Layer: Physical layer can provide the ground for better QoS that the upper layers will try to utilize. Physical layer can use better error correction mechanisms and may allocate more time for transmission to higher CoS. In addition, it may adaptively change the modulation scheme according to the channel state and in that way to provide service even when the channel Bit Error Rate (BER) is high.

2.4 Cross-Layer design

The last several decades in the history of networking the network protocols are distributed into several independent layers. All the functionality is divided in layers and each layer is designed separately. The interaction between the protocols in the different layers is performed through a well-defined interface. This type of architecture follows the “divide and conquer” principle and has the advantage of complexity reduction and increase of the architectural flexibility. By defining exactly the functionality of each layer and the service access point interfaces in each layer, one implementation of a layer can be seamlessly replaced by another one. When a protocol is replaced by another one, there is no need to modify the rest of the network stack as long as the new protocol uses the same interface with the upper and lower layers.

However, the layered protocol design has been challenged for years for several reasons. One reason is that the need for quality of service is ever-increasing as multimedia and other time sensitive applications are becoming popular. Connectivity alone is not enough for a user anymore and features such as high speed transmission rate and quality of service (QoS) for time-critical applications are also desired. The conventional layered protocol design appears as a non-viable solution for these requirements. One hint is that most of the end to end QoS solutions that are already implemented for the Internet are using more than one protocol layers. Furthermore, the Internet consists of multiple heterogenic networks that are easier to be integrated all together with a layered architecture while on the other hand, the performance is not optimized. One example that demonstrates the reason for that phenomenon is the routing. In order to decide which route is more applicable for communication’s needs, interaction with the MAC layer is needed to collect information about the links along the path. In addition, the fact that most of the wireless networks today have no stable performance links between nodes increases the need for cross layer design.

The cross-layer design helps the protocols to optimize their performance by enabling layers to exchange state information. The cross layer information facilitates the different layers to have a better picture of the constraints and characteristics in the network and as a result allows them to make decisions that would jointly optimize the network and have better coordination. In order to achieve the desired coordination, the protocols must be developed in an integrated and hierarchical framework so as to take advantage of the inter-reliance between the protocols. The cross layer information is related to the system constraints, several metrics, the requirements of the application etc.

Even though the cross-layer design may assist drastically the performance optimization, it increases the cost of deployment because of the added complexity, jeopardizes stability, and would hold back the proliferation and development of wireless networks, as industry standards are hard to be created. So, the aggressive use of cross-layer design must be avoided. A better approach would be to minimize its dependency on the information from other layers by careful placement of the various functionalities in the protocol stack. However, cross-layer design may be a suitable approach for stand-alone wireless networks that are dedicated to a single application, especially if the task is highly critical, and for cases where reliability and performance are more important than cost.

2.5 Physical Layer

Physical layer technologies are using the advances in the sciences of communication theory, digital signal processing, RF technologies, and circuit design to improve their performance. The improvements of the physical layer technologies are mainly targeting to increase transmission rate, improving error resilience capability, and convert most of the radio functions to the software domain for easier configurability.

To facilitate the increase of the capacity in wireless networks, several physical techniques have been emerged. One of the most important advances is the orthogonal frequency division multiplexing (OFDM). OFDM has significantly increased the speed of IEEE 802.11 from 11 Mbps to 54 Mbps. Another technique that can achieve even higher transmission rate is the Ultra-Wideband (UWB). However, UWB has a short transmission distance thus is applicable only for short-distance applications. Multiple-input multiple-output (MIMO) mechanisms are used for high speed links in a wide area network. If a transmission speed as high as that of UWB is desired in a wider area network such as WLANs or WMANs, other physical techniques, such as the multiple-input multiple-output (MIMO) mechanism, are needed. So, multiple antenna systems have been used in order to further increase capacity and reduce the harm of fading and interference. Every new technique that is developed to increase the capacity and the transmission rate should target to improve the spectrum efficiency.

To increase error flexibility, many channel coding schemes have been proposed and used to shield the communication from bit errors. Wireless channel conditions are not stable, thus the bit error rate is not constant. Therefore, an adaptive channel coding scheme is used to change the coding methods and the amount of redundant information for forward error correction, according to the channel conditions.

The capability to configure the various physical layer techniques by software brings many advantages to wireless communications. Physical layer techniques can be optimized adaptively according to the variable conditions in the environment. The wireless spectrum can be utilised more efficiently and easily. New techniques can be more easily adopted from radios without costly hardware changes.

Most of the physical layer techniques that already exist are usually concentrated to single-hop point-to-point communications. However, when they are applied to multi-hop ad hoc networks, new problems appear that degrade their performance. Thus, the physical layer techniques have to be adjusted to fit the multi-hop environment or new more suitable techniques need to be invented.

2.6 Mac layer

Physical layer succeeds to give to a network node the capability for point-to-point communication. The next step for practical realization of communication between network nodes is the need of several services which are supplied by the next layer and they are described next.

The first of these services is a mechanism to interpret the bit streams and convert them into packets and vice versa. An operation mechanism and algorithms are also needed to manage transmission and reception of packets among many nodes with the objective to avoid collisions and the improvement of network performance. This type of function is called medium access control (MAC). Error detection and correction schemes are also used since there can still be errors occurring in bits or in packets, even though an advanced coding algorithm is applied in the channel. In wireless networks there are many factors like link quality variations, interference etc and as a result, additional error control is usually desired on top of physical layer.

The main task of every MAC protocol is to manage the common medium among multiple nodes. There are two main categories based on the way the channel access is coordinated in the network, centralized or distributed. In a centralized MAC protocol, the entire management is located in a centralized node, and all other nodes need permission from this certain node to access the medium. The networks that fit most in a centralized coordination are the networks that have a centralized access point like cellular networks, infrastructure mode wireless LANs, satellite networks, etc. However, in multi-hop ad hoc wireless networks, distributed medium access control is preferred, because the network itself is distributed in essence. If a centralized MAC scheme is chosen for multi-hop ad hoc networks, it lacks enough efficiency trying to maintain the centralized control among multiple nodes and the network loses the decentralised characteristics that make it useful on the first hand. In addition, a centralised MAC inhibits the scalability of the MAC protocol. As a result, distributed MAC is the only logical choice for wireless mesh networks and for MANETs as well. However, most of the current work is focusing on centralized MAC and that makes the design of distributed MAC a much more challenging task. A MAC protocol does not replace the basic multiple access schemes such as TDMA, CDMA, or OFDM that exist in the physical layer but it considers them as a starting point for the design of a MAC protocol.

The coordination of the medium access can be also done in a reservation or in a random manner. In a reservation based MAC protocol, the protocol assigns resources such as time slots, codes, subcarriers or channels, to nodes in a way that the overall QoS in terms of throughput and delay is maximized. Besides that, cross layer design by cooperation with the network and transport layers also needs to be considered. A time division multiple access (TDMA) function in MAC may force slow start performance of TCP owing to the significant differences of round trip time (RTT) before and after resource allocation [Wang, X. (2006)]. When a random access MAC protocol like CSMA/CA is used, the challenging task is to minimize collision and decrease the time for recovering when collision happens. Given that no reservation is taking place, the probability for collision becomes high when the number of users increases with impact to the throughput and QoS performance. As a result, it is impossible to guarantee QoS. On the other hand random access MAC protocols are simpler with no overhead because no separate signalling and reservation schemes are needed. Furthermore, a reservation-based scheme in MAC faces integration problems with a connectionless network. For instance, when TDMA or any other reservation mechanism is used, then TCP has to wait for the allocation of the time slots to be done. Such a delay makes TCP to assume that the network is congested before even resource allocation is completed. In another example, when reservation based MAC is used to send video traffic the reservation mechanism has no information about the bandwidth and QoS requirements. Without such information, reservation cannot allocate the required resources and more complicate mechanisms like adaptive resource estimation and dynamic time slot allocation are needed. However, when random access MAC is used a packet starts its transmission process as it arrives.

Two of the most well known problems in medium access management are the hidden terminal and the exposed terminal issues. The hidden terminal problem occurs when a node A is visible from another node B, but not from other nodes that communicate with node B. In that case, node A after sensing the medium finds that no one transmits at the moment and decides to send packets to node B. Then a collision occurs in node B. Hidden terminal issue is visualised in Fig. 2. The exposed node problem occurs when a node is prevented from sending packets to other nodes due to a neighbouring transmitter. Let’s consider an example in Fig. 3 of four nodes, two receivers R1, R2 and two transmitters S1, S2 where the two transmitters S1 and S2 are in range of each other and R1 and R2 are out of range of each other. When S1 transmits data to R1 then node S2 cannot transmit to R2 because it senses the common medium and see that someone else is already transmitting. Although this behaviour looks right it is not very efficient because R2 could still receive the transmission of S2 without interference for the reason that it is out of range of S1.

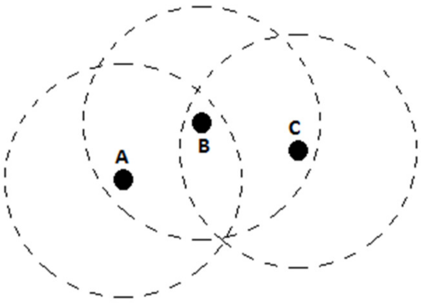


Fig. 2. Hidden terminal problem.

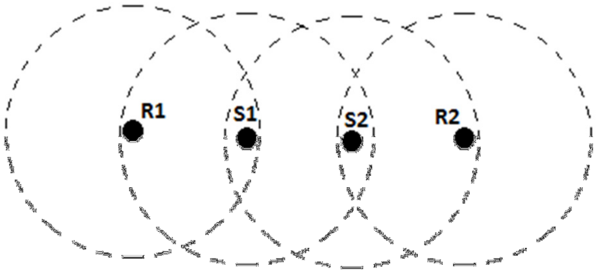


Fig. 3. Exposed node problem.

The Request to Send / Clear to Send mechanism is designed to encounter these problems. The procedure is the following: A node that has to send data initiates the process by sending a Request to Send (RTS) frame, then after the destination node receives the RTS replies with a Clear to Send (CTS) frame. All the neighbour nodes that overhear either RTS or CTS frames should avoid transmitting for a given time. This amount of time is included in both the RTS and the CTS frame. This mechanism is designed assuming that all nodes have the same transmission range. These two basic problems highlight the complexity and unpredictability in wireless multi-hop networks.

2.7 MAC Layer solutions

Due to different nature and characteristics of multi-hop ad hoc networks and especially MANETs, the protocols in each layer in the protocol stack needs to be reconfigured or redesigned in order to be optimized and operate. MAC layer is not an exception. One of the most used single channel medium access protocols is CSMA/CA. CSMA/CA is a random access protocol and no resource reservation is taking place. Several schemes have been proposed to improve its performance in wireless multi-hop networks.

Some of the proposals are targeting to improve and adjust the physical carrier sense. Physical carrier sense is vulnerable in hidden node or exposed node issues. When the sensitivity is high, many nodes become exposed. On the other hand, when the sensitivity is low many nodes become hidden from each other. Hence, in [L. Krishnamurthy (2004)] some proposals have been made about using variable carrier sense range by dynamically changing the sensitivity. However, optimizing a system that will manage the sensing range of all nodes in the network in a dynamic way still remains a research problem. One of the solutions is used in the actual implementation of an IEEE 802.11 wireless LANs, where the sensing range that is used for CSMA/CA is such that it can sense the transmissions from nodes at a distance of more than two hops away. As it is mentioned before, whenever the sensitivity is increased then the exposed node issue arises. In order to reduce exposed node problem, the physical carrier sense is modified to be directional like it is proposed in [Choudhury, R (2002)].

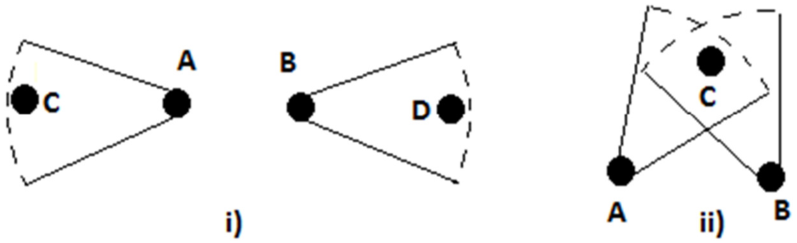


Fig. 4. Exposed and hidden terminal problems when using directional antennas.

Thus, Node A and Node B being out of the directional coverage of each other, can transmit data, in Fig. 4i, at the same time. The proposition of using directional antenna has several shortcomings. For nodes in the directed coverage of each other, exposed nodes problem still exists. On the other hand, when nodes are out of each other's coverage, the network is partitioned. Furthermore, hidden nodes will appear in situations like in Fig. 4ii, where Node C is in the coverage of Node A and Node B, but Node A and Node B cannot sense each other resulting in the hidden nodes problem and cause collisions in Node C. The most important problem for MANETs is that directional antennas are increasing the complexity to the point that the network cannot be operational. Wang et. al. [Wang, X. (2005)] proposes directional back off scheme for reducing exposed nodes problem. When a node senses a busy channel, it does not always reschedule its transmission but it checks if the destination node senses as well a busy channel. If destination does not sense a busy channel and the back off counter is zero, it initiates transmission because the destination node will be able to receive the data without collision.

In order to assist the delay sensitive applications, a new protocol has emerged in the MAC layer, IEEE 802.11e [Mangold, S. (2003)]. IEEE 802.11e is trying to guarantee small delays and better QoS in general to the applications that need it, by categorizing traffic according to four Access Categories such as voice, video, best effort and background. The 802.11e enhances both the distributed coordination function (DCF) and the point coordination function (PCF), through the new Hybrid Coordination Function (HCF). However, it does not provide guaranteed throughput.

Several companies realised that a standard is needed to solve interoperation problems in multi-hop ad hoc networks. Hence, they established an IEEE sub-working group, the IEEE 802.11s. IEEE 802.11s [Hiertz, G. (2010)] specifies both MAC layer and routing functions. The first draft of the IEEE 802.11s describing the network architecture, end-to-end protocol stack reference mode, suggests routing protocols, topology formation and optimal multichannel operation. Some of the functions that are not mentioned in the IEEE 802.11s standard so far, refer to how a mesh point supports access for legacy IEEE 802.11 clients or how there is no detailed algorithm of channel allocation in the multichannel mode. Furthermore, QoS is not under consideration in IEEE 802.11s. In addition; many other functions are still under development and discussion and will be specified in the final version of the standard.

In IEEE 802.11s the routing protocol (Hybrid Wireless Mesh Protocol or HWMP) is defined and moved to the MAC layer. Yet, it allows vendors to operate using alternate protocols. HWMP is inspired by a combination of AODV and tree-based routing. The routing and MAC layers are working transparently to each other so there is not cross-layer support. Such a design looks interesting but there is still the question if it can really improve the performance.

Many IEEE 802.11s contributors are focusing to improve the scalability problems of CSMA/CA in multi-hop ad hoc networks. The contributions are trying to configure the CSMA in a way to improve the performance in mesh networks, although it is becoming clear that multichannel operation can provide a significant improvement in scalability. However, IEEE 802.11s does not include any mechanism for interaction between CSMA/CA and multichannel operation.

Also the challenge is that MAC layer is implemented not only in software but a big part is also implemented in firmware. When software radios appear then cognitive radio will be realized in its full potential and more flexible and powerful MAC protocols can be easily developed.

2.8 Network layer approach

One place in the stack of protocols, that someone can look and try to improve the Quality of Service in a network, is the Network layer. The basic responsibility of the network layer is the routing of packets including routing through intermediate routers (in case of MANETs, other nodes). Hence, early research was focused on designing routing protocols that discover routes in the network that are able to support the communication. The routing protocols that are used now in MANETs are oriented on discovering paths in the network and use the one with the fewer hops. Some hop-count protocols are the on-demand AODV [Perkins, C. (1999)] and DSR [Johnson, D. B. (1996)] and the proactive DSDV [Perkins, C. E (1994)] and OLSR [Clausen, T (2003)].

2.8.1 Additional metrics

However, minimizing the number of hops maximizes the distance between two nodes in each hop. Usually, this approach results to low signal strength and high loss ratio. In addition, even if there are several minimum hop-count paths, there is not a way to find out which one is the best because there is not an additional metric to distinguish them. Douglas S. J et al. [De Couto, D. S. J. (2003)] used a DSDV test-bet to prove experimentally and quantify the effect that minimum hop strategy usually delivers paths with low throughput and high loss ratio. They proposed the expected transmission count metric (ETX) as alternative metric. ETX metric predicts the number of retransmissions in a path by adding the packet loss ratios in each link in the path. Hence, the criteria for choosing a path will be the ETX and not the number of hops. ETX gives a good estimation of the link quality but cannot quickly track the changes in a link quality. Also in newly established links, ETX may not be available yet. Yarvis et al. [Yarvis, M. D. (2002)] follow the same approach in sensor networks by observing the poor performance of the hop count metric and propose another loss aware metric which is based on the probability of a packet to be lost in the path. Awerbuch et al. [Awerbuch, B. (2003)] worked also on an alternative metric, the Medium Time Metric, which relies on the fact that the links in the network have different data rates. Their approach selects a route that the total medium time consumed sending a packet is minimized. This results in an increase in total network throughput. There are other proposed algorithms that take into consideration the mobility of the nodes which are based on the idea that nodes which have been stationary for a threshold period are less likely to move. Associativity Based Routing (ABR) uses stability and hop count as criteria to select a path [Toh, C.-K. (1996)]. In [Dube, R. (1997)] a routing protocol that selects the most stable routes through the dynamic network is proposed. This, on demand routing protocol, called Stability-Based Adaptive Routing (SSA) protocol, uses the signal strength and stability of each node in a path for routing decision. The simulation results show that location stability information can be a useful metric, although it must be used carefully since misinformation about stability patterns is considered quite costly and has negative impact on routing and performance. One of the reasons that a packet can be delayed, is the queue waiting inside a node's buffer. The Delay-Sensitive Adaptive Routing Protocol (DSARP) [Sheng, M. (2003)] utilizes the effect of buffer usage on delay. The route reply packet that follows the path from the destination back to the source, gathers the number of packets waiting in the buffer of each node in the path. The path with the smallest number of packet waiting in the buffers is considered the one with the less delay. This also helps the network load balance which eventually helps to distribute evenly the energy usage in all nodes of the network. Again, it is a little optimistic to assume that the path with the less buffer usage is the faster one because delay depends on other parameters also.

2.8.2 QoS Aware routing in Mobile Ad-Hoc Networks

As MANETs are becoming popular and are increasingly used for applications that need better services from the network, the research efforts are focusing on designing routing protocols that discover routes in the network which are adequate to support the QoS that the applications require. Generally, QoS in networks refers to resource reservation mechanisms that guarantee a certain quality that the network will offer. In MANETs because of nodes mobility and the unexpected condition changes in the shared wireless medium, offering guaranteed QoS is not practical. Hence, in MANETs soft QoS and QoS adaptation is used, where failure to meet QoS is allowed. The only option that is left is to create the conditions under which soft QoS will succeed by selecting routes that are more probable to have sufficient resources for enough time (resources estimation), use route maintenance mechanisms like redundant routes and multipath routing, intelligent route selection (like bio inspired, heuristic or game theory algorithms) and resource reservations for the resources that can be reserved in a MANET.

Almost all the aforementioned solutions compare their performance with the existing hop-count protocols in order to evaluate their metric by replacing the hop count metric with the new one. According to their results the new metrics are performing better, because larger throughput is achieved. However, even though the protocols in question improve the QoS by improving the performance, absolutely none of these protocols succeed to provide a better guaranteed soft-QoS for delay sensitive applications. Furthermore, each one improves the throughput and balances the network by using solely one or two metrics for routing decisions.

2.8.3 Contention Aware protocols

A non-flat hierarchy routing protocol is proposed by R. Sivakumar et al. [Sivakumar, R. (1999)] which implements a core network for performing route computations (CEDAR). Each node in the core network keeps up-to-date information about its local dynamic topology and propagates the link state information about the more stable links that connect them with core nodes far away in the network. CEDAR accomplishes to reduce overhead about route maintenance. The basic idea is that the information about stable high-bandwidth links can be made known to core nodes far away in the network, while information about dynamic links or low bandwidth links should remain local. Consequently, CEDAR approximates a minimalist local state algorithm in highly dynamic parts of the networks while it approaches the maximalist link state algorithm in highly stable parts of the networks. This technique assumes that MAC layer is capable on estimating the available link bandwidth. A heuristic interference-aware QoS routing algorithm (IQRouting) is proposed by [Gupta, R. (2005)] where the routing overhead is reduced by choosing routes based on localized information at the source nodes. In addition, it estimates a channels residual capacity by measuring the channel usage from the links that interfere with each other. The same approach of using information from nodes inside collision domain range by overhearing neighbours' transmissions has been used in [Yang, Y (2005a)]. An admission control protocol is proposed that uses this information and predicts the available bandwidth in the channel. The requests for new flows through specific areas in the network are managed according to the available bandwidth.

2.8.4 Link Stability based Routing

As it is already noted, QoS in MANETs in the sense of guaranteed resource reservation, cannot exist. One way for soft QoS success is to select routes that are more probable to be stable by having long expected lifetime. In [Rubin, I. (2003)] Rubin and Liu have created a link stability prediction mechanism. The probability distribution function (PDF) of link lifetimes under various mobility models is calculated and used to predict the residual link lifetime. By statistically predicting the lifetime of a link it is easy to avoid paths that will fail soon. The difficulty on this solution is spotted on the fact that the node mobility pattern must be known and modelled accurately for correct predictions. Apart from that, it can be a waste of useful paths in case the protocol avoids links solely because of the mobility metric based prediction results, even if the signal to noise ratio, the capacity in the channel or other metrics are in very good state. Thus, the combination of this stability metric with other metrics can produce much better results. A similar approach is followed by Trivino-Cabrera et al. [Trivino-Cabrera, A. (2010)] and Tseng et al. [Tseng, Y.-C. (2003)]. A better link stability prediction mechanism has been proposed by Shen et al. [Shen, H. (2003)] which defines the entropy of a link as a function of the relative positions and velocities of the end nodes in the link and the transmission range. The entropy of the path is the product of the link entropies along it. The more stable the path is the less entropy it has. The stability prediction has the potential to be more accurate as it involves more parameters, relative speed and position, than just a general PDF. This approach does not depend on the need of an accurate mobility model. On the other hand, it assumes that nodes have a self-configuring localization mechanism. Another way to calculate the stability of a link and how stable is the environment around it, is to check the set of neighbour nodes and how they change in a period of time. The Hybrid Ad hoc Routing protocol (HARP) [Nikaein, N. (2001)] uses the quality of connectivity metric which is a combination of buffer space and relative stability by keeping a list of neighbours and checking how many of them are changed in a period of time. The less that changed, the greater the stability was. This method is more reliable as it measures directly the mobility in respect to the neighbours. As before, these protocols can also perform better if they are combined with other metrics for the routing decisions. In addition, these protocols target solely the link lifetime without considering the overall performance lifetime. A path is not very useful even when it is up, if it cannot offer the required QoS.

2.8.5 Energy aware protocols

Mobile devices depend on their energy storage which is limited. If the power consumption is high then the mobile nodes battery lifetime will be short. Several researchers are investigating methods to reduce the energy consumption in order to avoid node failures and link breakages. In [Yu, M. (2007)], Yu et al. combine mobility prediction, link quality measurement by using ETX metric and energy consumption estimate to create a protocol that is QoS aware and at the same time makes energy efficient routing decisions. According to the results, the proposed protocol reduced link breakage and average delay in the network. But the protocol assumes that the nodes in the MANET are capable to measure their mobility by using a Global Positioning System (GPS) or other methods which unfortunately is not always true. Similarly, the residual energy in the nodes can be used as a routing metric to provide fair sharing of energy consumption in the network [Guo, Z. (2007)]. The distribution of the energy consumption in the network does not entail that the traffic in the network will be distributed at every moment. In fact, it makes it more possible to create congestion in the network due to all traffic going through the nodes with more energy than others. In [Woo, M. (1998)], the authors introduce five new power-aware metrics that can be integrated with other hop-count routing protocols to improve the overall network energy consumption. Energy aware protocols are useful in networks that the cost of maintenance in terms of recharging the batteries is high and providing constant power is hard or impossible like in sensor networks. In networks where nodes have enough power or there is no interest for saving power, most of the energy aware protocols are not useful and also they cannot operate normally providing the QoS improvements they may promise.

2.8.6 Position based protocols

Geographical position based routing algorithms have received significant attention since the progress of Global Positioning Systems (GPS) and self-configuring localization mechanisms. One of the advantages of the position based routing is that the overhead due to control messages and the size of routing tables are minimal. In geographical routing the source node forwards a packet to the geographic location of the destination instead of using the network address. Taxonomy of position based routing algorithm can be found in [Stojmenovic, I. (2002)]. One of the most popular techniques is the combined Greedy Forwarding (GF) and Face Routing. Greedy forwarding tries to bring the packet closer to the destination by forwarding it to the neighbour node that is geographically closest to the destination. GF can reach to dead ends where no other neighbour node is closer to the destination. Face routing then is used to find a path to another node where greedy forwarding can be resumed. Nevertheless, geographic routing mechanisms assume that every device has a localization mechanism and the source node somehow knows the location of the destination node. These assumptions and especially the second one are usually untrue. Furthermore geographical routing has issues with mobility and greedy forwarding fails when the network is not dense enough. In order to remove the need for localization and minimize the impact of localization errors, Virtual Coordinate systems have been proposed as an alternative to geographic location in geographical position based routing algorithms. Rao et al. [Rao, A. (2003)] are using a relaxation algorithm that associates virtual coordinates to each node so geographic routing can be used. However, virtual coordination systems produce an overhead, cancelling one of the major advantages of geographical rooting. Several studies also handle the issue that GF tends to forward the packets through lossy links because the geographically closest to the destination node is not always the best choice in terms of link quality. Seada et al. [Seada, K. (2004)] and Lee et al. [Lee, S. (2005)] have proposed solutions that attempt to balance between proximity and link quality.

2.8.7 Heuristic approaches

Multi-hop QoS routing is a multi-constraint NP-complete problem. Heuristic methods are usually the most feasible way to solve it. In [Barolli, L. (2003)], Barolli et al. proposed an application of Genetic Algorithm (GA) to find the best path among the available ones. The protocol they propose (GAMAN) encodes the paths into chromosomes by encoding the links as genes. The protocol collects the delay and reliability information for each link and uses them to calculate the fitness of each path. The GA generates an initial population and calculates the fitness according to the measured delay and reliability. Then it performs crossover and mutates the chromosomes to produce the next generation which will contain better paths. Then GA repeats again till it finds the optimum path. The limitations of this method are first that it needs to collect information for many nodes in order to have enough genetic material. Secondly, the selection of the function that calculates the fitness has to represent very accurately the real world in order to estimate how well a path can perform. This is extremely hard and can be very complicated if many metrics are involved. A guaranteed way to give a correct fitness value to a path is by trying and using it and then measuring how it performed. However this method is impractical because it produces a lot of overhead and needs too much time to test all possible paths that the GA will produce. Other heuristic optimization algorithms for the routing problem in MANETs have been used. A swarm intelligence algorithm, the ant colony optimization integrated with simulation annealing (SA) is used in [Fu, P. (2005)]. Ant colony is used to mimic the way that ants find the shortest path from the food to the nest. The local search method based on SA is used to increase the convergence rate of the ant colony algorithm and avoid the problem of non active routes.

2.9 Transport Layer approach

Like Network Layer protocols, specially designed Transport Layer protocols can also improve the performance of the communication in a MANET. The widely used TCP transport layer protocol experiences significant performance degradation as it is not designed to operate in mobile ad hoc networks. Most of the research effort has focused on creating and improving the MAC and Network layers and not enough work has been done in the transport layer.

Transport layer protocols can be divided in two main categories. These main categories are related to the type of communication needs, reliable data transport or real-time delivery.

2.9.1 Reliable data transport protocols

From the existing transport layer protocols TCP is the most widespread protocol that targets reliability. In order to adapt TCP in the different environments of different kind of networks several TCP variations have been proposed and used.

In [Mbarushimana, C. (2008)], Mbarushimana et al. discussed the negative effects of the IEEE802.11e service differentiation scheme on TCP performance especially in the presence of high priority traffic. They proposed a TCP friendly scheme that gives high priority to the ACK packets in order to avoid problems like spurious timeouts.

In a MANET, packet loss can occur due to either congestion or other errors. In wireless networks, congestion does not constitute the most frequent reason. The most common reasons of packet losses in MANETs are low link quality, route failure, route change due to mobility, etc. TCP is designed for wired networks where the most common reason for packet losses is congestion. Thus, TCP reacts to lost packets by invoking the congestion control mechanism, which slows the data rate and severely degrades the performance. To avoid the described TCP behaviour, a mechanism is needed to differentiate the reasons of packet losses. This approach is followed in [Chandran, K. (2001)] where congestion and packet loss due to route failure are handled separately. In order to inform the sender about a route failure, a feedback scheme called TCP-feedback (TCP-F) is used. The route failure is detected by the routing protocol in the network layer of the node that the failure happened. Then a route failure notification is sent to the next-hop node until it reaches the source node. When TCP in the sender receives the notification, it stops the communication and waits for a route re-establishment message. When a route re-establishment message is received TCP resumes communication without adjusting congestion window or any other parameter. A similar method to TCP-Feedback is explicit link failure notification (ELFN) proposed in [Holland, G. (1999)]. ELFN differentiates congestion from packet loss due to link failure and notifies link failure to the source node using Internet Control Message Protocol (ICMP) message. TCP freezes the communication like TCP-F does and sends periodically a probe packet to check if a new route has been established. If a probe packet succeeds the TCP in the sender goes back to the normal operation without adjusting any parameter. Several feedback-based schemes have been proposed for TCP, but no reliability mechanism is used for guarantee that the notification messages will reach their destinations. One basic characteristic that is missing from this approach is that TCP cannot adapt to any changes. After a route is re-established there is high probability that the previous TCP configuration is not appropriate for the new conditions which may lead to heavy congestion or low efficiency.

In [Dyer, T. D. (2001)], a protocol that does not rely on feedback from the lower layers and is called fixed retransmission timeout (RTO) is proposed. When a timeout occurs consecutively, the sender assumes that a route error has occurred and the lost packet is retransmitted without doubling the RTO a second time. Then, RTO remains fixed until the route is re-established and the retransmitted packet is acknowledged.

A different approach that targets to regulate the data rate in order to avoid burst transmissions is TCP adaptive pacing (TCP-AP) [ElRakabawy, S. M. (2005)]. In TCP-AP the transmission rate is adjusting by considering congestion window, contention on the end-to-end path, and spatial-reuse constraint. TCP-AP does not violate the basic architecture principles of TCP like the end-to-end semantics and does not require changes in lower layer protocols such as routing and MAC. However, TCP-AP assumes that the mechanism of rate control is effective and the estimation of contention and spatial reuse constraint is accurate.

Apart from the different TCP flavours that have been proposed, there are also researchers that have built new transport layer protocols by having the ad hoc features in mind. One of these solutions is the ad hoc transport protocol (ATP) [Sundaresan, K. (2005)]. The same work shows through detailed arguments and simulations that several of the design elements in TCP are fundamentally inappropriate for the unique characteristics of ad hoc networks. ATP is rate-based with a quick-start approach for initial rate. Delay is used for congestion detection in order to avoid the problems that TCP faces. In addition, there is no retransmission timeout, and reliability and congestion control are decoupled.

2.9.2 Real Time transport protocols

The other category of transport protocols is the delay aware protocols for real time applications. Usually UDP is used for these kinds of applications where reliability is not the basic target. Unfortunately UDP cannot guarantee real time delivery and can face large drop packet ratios while disturbing the other communications in the network. Hence, in top of UDP other protocols like real-time protocol (RTP) and real-time transport protocol (RTCP) are operating along with rate control protocol (RCP) for congestion control. Several RCP protocols have been created for wired networks that are not suitable for wireless networks for the same reason that a differentiation between the causes of dropped packets is needed. In [Cen, S. (2003)], solutions of RCP protocols for wireless networks with loss differentiation algorithms and congestion control are studied. Sadly, those solutions are not applicable for multi-hop networks since they are considering that only one wireless link exists between sender and receiver.

Fu et al [Fu, Z. (2003)] proposed an adaptive detection rate control (ADTFRC) scheme for mobile ad hoc networks. ADTFRC performs multi-metric joint identification for packet and connection behaviours based on end-to-end measurements for TCP-friendly rate control schemes. However, all the non congestion packet losses, whatever the reason of the loss, are treated in the same way which might evidently lead in poor performance.

2.10 Summary

The main open research interests in the MAC layer towards the performance improvement in MANETs are the following. Scalability in multi-hop ad hoc networks is one of the important issues and several CSMA/CA based solution solve partial scalability problems. In addition, most of the ongoing research efforts are focused on capacity, throughput or fairness. Yet, many mesh networks need to support multimedia traffic communication. As a result, one of the hottest topics in the MAC layer, as well on the other layers, is development of protocols with multiple QoS metrics such as delay, packet loss ratios, and delay jitter. An important step in the development of the protocols in MAC layer is the ability to easily reconfigure the protocols. This ability enables the freedom of new ideas to be added to the MAC protocol and the cross-layer design with the other layers such as physical and routing. MANET’s special characteristics create new opportunities for useful applications but on the same time generate special conditions. An introduction of the MANETs and their characteristics was presented. A brief introduction of several MAC and Physical layer technologies shows the variety of the solutions which make harder the standardization of the cross layer information from these layers. Routing layer depends on the cross layer information in order to select the best available path for the communication between remote nodes. However, the dependence of the routing layer from the cross layer design makes it vulnerable to availability or accuracy of the cross layer information. A more independent scheme that helps the routing protocol to adapt on the lower layer capabilities is needed. Bayesian Networks as a decision mechanism has the required characteristic of tolerance that the routing layer needs. This complicated mechanism makes harder the work of the transport layer. In order to avoid the effects of an even more complicated design, the transport layer needs a more autonomous flow control solution. Genetic Algorithm basic mechanism is introduced and shows its autonomous and self configurable nature of the algorithm.

Furthermore, in this chapter a discussion can be found about the difficulties to achieve QoS in a mobile multi-hop network and the basic differences over QoS in the classic network architectures. Soft QoS is a more suitable solution in an environment that is not practical to reserve resources, like the dynamic environment of MANETs.

In addition to correctness, a routing protocol for mobile ad hoc networks should minimize route setup and maintenance messages in order to have a low communication overhead. The protocol should also converge quickly to select a route before network changes make the route invalid. Furthermore, the routing protocol must be distributed since no centralized host is available. Distributed nature makes also the protocol scalable. Finally, it should take advantage of the technology available and utilize information about the current state of the network by using cross layer design. Most of the existing routing protocols focus solely on improving one or two aspects of the network and overlook the others. In addition, most of the protocols are relying only on specific metrics even though there are more metrics available which can give more and better information. Other solutions are assuming that some services are given, like GPS devices and MAC layer capabilities. Consequently, they are not flexible enough to be used in every single situation. A system that can take advantage of the plethora of information about the network is needed while at the same time it can be tolerant to missing or inaccurate information in order to decide which route will be used. Cross layer design between network and MAC layer is required in order to enhance the efficiency of the routing algorithm but at the same time the routing algorithm must not rely exclusively to the cross layer information.

In spite of many path metrics for multi-hop networks available, new routing metrics are still needed to further improve the performance of routing protocols. In wireless mesh networks, many performance parameters and constraints are cross-related and make routing a more difficult problem than fixed cellular or wired networks.

As discussed before, many solutions have been proposed but there are still many research issues open for both reliable transport protocols and real-time transport protocols. Most of the transport layer performance degradation problems are related to the lower layers. Thus, cross layer design is the most common design that the research is focused on. However, the higher in the protocol the greater the amount of cross layer information thus the complexity is greater. It is a challenging issue to design algorithms that can evaluate and use this information in an efficient way. Also it will be very valuable the design of protocols that can configure themselves by just sensing the complex environment that consists of the lower layers and the network.

3 QoS Aware Adaptive Flow Control in Transport Layer using Genetic Algorithm

3.1 Introduction

MANETs are becoming increasingly important and several delay sensitive applications are starting to appear in these kinds of networks. An issue in concern is to guarantee Quality of Service in such constantly changing communication environment. A term in adaptive network protocol design called Self-modifying protocol (SMP) is aiming to modify its network communication parameters at run time such that the protocol can adapt to the changing communication environment and user requirements on the fly. In this work, a self-evolvable protocol is designed using the principle of SMP with Genetic Algorithms. In essence, this work applied the Genetic Algorithm in the design of an evolvable self-modifying protocol (ESMP), which controls the flow of the data in a way that it will improve QoS that applications experience. We use Network Simulator (NS2) to evaluate the proposed Self Modifying Transport Protocol in order to demonstrate the feasibility of our new design approach. First we demonstrate how the size of the packet and the time interval between the packets can affect the performance of the protocol, and then we investigate how the genetic algorithm can find the optimum values for these parameters.

Genetic Algorithm (GA) has received increased interest in recent years, particularly with regard to the manner in which it may be applied for practical problem solving. The main reason for us to adopt GA as an optimization method is due to its robustness to a changing environment. Traditional methods of optimization are not robust to dynamic changes in the environment and often require a complete restart to provide a solution. In contrast, GA can be used to adapt a system to find out a solution to dynamic changes. The available population of evolved solutions provides a basis for further improvement and in most cases it is not necessary to reinitialize the population randomly. Furthermore, hardware implementations of GA are extremely fast and they are not very sensitive to network size [Tufte, G. (1999)].

3.2 Transport layer

The Open Systems Interconnection (OSI) model defines the several network stack layers. OSI defines as the Transport layer job the reliable arrival of messages and data flow control. The services that transport layer provide are connection oriented and connectionless. For connection-mode transmissions, a transmission may be sent or arrive in the form of packets that need to be reconstructed into a complete message at the other end. Various transport layer QoS parameters like throughput, delay, and packet error ratio must be supported by the transport layer protocol. In multi-hop ad hoc networks, both real-time and reliable traffic are supported by transport protocols. Real-time traffic is usually tolerant of packet losses but it is delay-sensitive. On the contrary, reliable or non-real-time traffic is resilient to delay, but demands accurate and lossless communication. Thus, different transport protocols are needed for reliable and real-time traffic in mesh networks.

Transport protocols like TCP, which have been designed by having in mind the usually stationary wired networks, have been realized not to perform well in MANETs [Hanbali, A. A. (2005)],[Holland, G. (1999)]. Elaarag [Elaarag, H. (2002)] describes how TCP congestion control is not suitable for the non-congestion induced packet losses in wireless environment. Then, he presents a survey of the research which has been done to improve the performance of TCP over mobile wireless networks and classify the proposed solutions into three categories: link layer, end-to-end and split. Finally, he compares the categories and the solutions in each category by presenting their strengths and weaknesses.

It has also been investigated [Kim, D. (2005)] how TCP Reno and Vegas performed over different routing protocols in MANETs and found that one TCP variant outperforms the other in proportion of which routing protocol is used. That enforces the need for a protocol that adapts to the environment, which is one of the objectives of the present thesis.

The principal problem of TCP lays in the assumption that packet losses are due to congestion. In wired networks this assumption might be true, in most cases because bit error rates are very low. Nearly all TCP versions, nowadays, assume that packet losses are due to congestion. Consequently, when a packet is detected to be lost, TCP slows down the sending rate by adjusting its congestion window. Unfortunately, in wireless ad hoc networks, there are several other reasons for a packet to be lost except congestion, making TCP not performing well.

Many proposals for the improvement of TCP in such environments are based on cross layer design. The main idea in cross layer design is that TCP collects information regarding the loss of packets from the other layers, and acts appropriately.

The transport protocols and the protocols that control the flow of the packets in an ad hoc network, must quickly adapt to these changes. Adapt means that the parameters of the transmission must change dynamically to adjust the flow of the packets to a more appropriate state for the new conditions.

Another important issue is the QoS. Most of the transport layer protocols do not care about delay or jitter and try to achieve high throughput without losing any packets. This is not always what an application needs. In multimedia applications real-time data of video or sound must be delivered to the destination with the smallest delay and jitter, in order to give the best experience to the user; however, the behavior of the transport protocol can affect the QoS metrics.

3.3 Genetic Algorithm

The genetic algorithm (GA) [Goldberg (1989)] is inspired from the natural process of evolution and natural selection and belongs to the larger class of evolutionary algorithms. GA is a search heuristic and is used in optimization and search problems. The basic operators of GA are the same with the natural evolution like selection, mutation, and crossover.

The basic element in a genetic algorithm is a population of strings which are called chromosomes and this population is the genotype of the genome. Each chromosome encodes candidate solutions for the optimization problem. Other names that are used for chromosomes are individuals or phenotypes. GA usually starts with a random population with random individuals. Then, the fitness of each individual in this initial population is evaluated. Afterwards, some of the individuals are selected by using a selection method that involves the fitness evaluation and modified by combining them and mutating some of their genes to form a new population. The new population is then used in the next iteration of the same procedure. The individuals in each new population that is created can usually score higher fitness in the evaluation. In most cases, the algorithm terminates when a satisfactory fitness level has been reached for the population or a maximum number of generations has been produced. In the latter case, a satisfactory solution may not have been found. So a typical genetic algorithm requires a genetic encoding of the solution domain and a fitness function to evaluate the solution domain.

The structure of the representation of a solution is usually a group of bits. Groups of other types like alphanumerical can be used in essentially the same way. Each gene is represented by one or more bits (or alphanumerical) and all genes have the same fixed size in each individual. The fitness function is defined over the genetic representation, is a measurement of the quality of the represented solution and is always problem dependent.

Once the genetic representation and the fitness function are defined GA is ready to run. First, it creates randomly the initial population of solutions and then improves it through recurring application of mutation, crossover, inversion and selection operators. Each operator is explained in more detail as follows.

Initialization

During initialization many random individual solutions are generated. The size of the population depends on the nature of the problem and typically contains several hundreds of individuals in order to cover all the range of the search space. For a faster convergence, the initial individuals may be seeded in search space areas where optimal solutions are likely to be found.

Selection

In order to proceed into the next population some of the individuals of the current population need to be selected. The fittest individuals are more likely to produce good quality offspring. Individual solutions are selected through a fitness-based process, where the fitter individuals are the most likely to be selected. Some more complicated selection schemes can be used but are always based on the fitness function.

Crossover

The next step is to use the individuals that are selected to produce the next generation. This procedure is called crossover. At the same time, mutation can be used to help the algorithm to avoid local maximal and "jump" into unexplored areas in the search space. Like selection there are several ways to do the crossover. The basic idea is that two parents give a part of their genome and the combination of the two parts produces the child. Thus, the child shares many of the characteristics of its parents. Crossover is repeated again and again until a new population of solutions of appropriate size is generated. Although crossover methods that are using two parents are more compatible with nature, some proposals suggest that more than two parents can produce higher quality chromosomes. These processes will result in the next generation of individuals that is different from the initial generation but with increased average fitness since the best individuals from the previous generation were used for breeding.

Termination

The repetitive generation production in the genetic algorithm has to come into an end. Usually, the termination condition that has to be reached for the GA to come into an end is the discovery of a solution that satisfies some criteria, when the successive iterations no longer produce better results, while a certain amount of time since the start is passed or a fixed number of generations is reached. There are other times where the GA is running all the time as a system is functioning in order to fine tune some system parameters and terminates along with the system.

Genetic algorithm pseudo code

```
Choose the initial population of individuals
Repeat until termination condition is true{
        Evaluate the fitness of each individual in that population
        Select the individuals for reproduction
        Crossover and mutation operations to give new offspring
}
The fittest on the last population is the solution
```

3.3.1 GA usage in computer networks designs

Most of the works that have been done in the area of computer networks using GA was about optimizing the routing in the networks. A genetic algorithm approach to the shortest path routing problem was used by Ahn and Ramakrishna [Ahn, C. W. (2002)] by working with variable-length chromosomes. Dengiz et al. [Dengiz, B. (1997)] showed how a genetic algorithm with specialized encoding, initialization and local search operators can optimize the design of communication network topologies. Another problem in the area of computer networks is the degree-constrained minimum spanning tree problem. Chou et al. [Chou, H. (2001)] experimented with several different encoding crossover and mutation methods in different network sizes and found that encoding has the greatest impact on solution quality.

These researchers tried to experiment with GA in environments that are constant; however, in the real-world situations, environments are not always constant, thus, the experimented outcomes may not be relevant and accurate.

3.4 Problem Complexity and Motivation

Most of the flow control designs on transport layer are relying on controlling the rate of the packets that are sent on the network and use a fixed size for the packets. Sometimes a flow control system might perform better if it sends the same amount of data but in smaller packets. For example, the protocol can send 100,000 bytes of data by using 100 packets of 1000 bytes each or 500 packets of 200 bytes each. More packets means more overhead but it implies less data lost if a packet is dropped. Which one will perform better, always depends on the conditions of the network and is relative to the application’s needs. In order to make it more clear we are going to give some examples.

Smaller data packets are susceptible to jitter when large packets are traversing the network through the same interfaces or occupying the media in neighbour nodes. Maybe in such occasion we can reduce the jitter by sending larger packets. In order to decide if this is the right action, the protocol must be aware of the reason that the application experiences deficient jitter. Maybe at Layer 2, a Link Fragmentation and Interleaving (LFI) mechanism is operating, which fragments big packets when there are signs of congestion in the link. Maybe in the next hop this kind of mechanism does not exist or is operating with different parameters and different thresholds. In case the protocol finally decides to send larger packets in a noisy environment then the probability to lose more data at once is becoming significant and this is going to make worse the delay and throughput and reduce the quality in general.

So the big question is how the protocol can be aware of all these parameters and how it can evaluate the significance of each one in respect of the application requirements. In a MANET where the environment is changing all the time, things are becoming more complicated.

GA, because of its nature, is an optimization algorithm that does not need to collect information about the condition of the network environment but the information on how well the individuals performed in it. First it tries randomly a population of individuals and evaluates them on how well they performed in the existing environment (calculate their fitness). Then it selects the individuals that performed better and uses them to produce the next generation of individuals that will usually perform even better. In order to correlate the usual GA problems with the data flow control problem, we refer to the network itself and its condition as environment, and every set of parameters like size of packets, time interval between packets etc., as individuals. The evaluation of each individual is done by measuring the performance in relation to the application requirements.

The rest of the chapter is organized as follows. First is introduced the protocol details and then we elaborate on the GA approach to solve this problem; specifically, we discuss our encoding strategies, various crossover and mutation operators, initialization and selection strategy, and the repair method. Next is described the research hypotheses and experimental design. Then the results are presented and discussed. Finally, the conclusions are drawn.

3.5 Proposed transport protocol that uses GA as optimization method (ESMP)

For the purpose of our study, which is to use the GA and see how it performs on finding the optimal solution as described before, we need a simple transport protocol that will function as the platform in which the GA will perform the optimization operation.

The first and basic requirement of this platform-protocol is to use the network layer service to communicate with every other node in the network and have implemented several procedures like packet acknowledgement etc. that assist the communication. Then it must be controllable on changing the size of the packets and the rate that sends the packets according to the optimization algorithm decisions. Finally, it has to measure the performance of the communication and report it back to the optimization algorithm (in our case, GA). In this section, we first describe our communication protocol and then the optimization algorithm using GA. Another concern in the design of the protocol was to avoid any stop ‘n’ wait kind of behavior because this is a QoS aware protocol.

3.5.1 Description of the transport protocol

All network protocols are using a specific designed header to accompany the data and this protocol is not an exception. The header can be found at the following image and consists of:

- IDs of the source and the destination ports like TCP and UDP,
- Sequence number: holds the number of the group of 16 packets that use the same configuration (belongs to the same individual),
- Delivery mask: is a mask that has only 1 bit set to define the number of the packet in the group and the other bits clear,
- Packet size: holds the size of the payload
- 4 flags: 2 of them are reserved and the other 2 are ACKp and ACKg flags that define if this is an acknowledgement for a packet or for a whole group. If ACKp is set then the sequence and delivery mask fields define the packet that the ACK belongs. When ACKg is set then the delivery mask is all clear, the sequence field holds the number of the group the ACK belongs. Data payload

contains the measured values for the delay, jitter, throughput and percentage of lost bytes for this group.

Bit	0-15	16-31	
0	Source (16 bits)	Destination (16 bits)	
32	Sequence (32 bits)		
64	Delivery Mask (16 bits)	Packet size (10 bits)	Flags
96	Data payload		

Fig. 5. Packet schematic layout.

The flow charts of the protocol logic are presented in Fig. 6 and Fig. 7. First, an initial population of 10 individuals is created. The first 16 packets are sent according to the genes of the first individual. The following groups of 16 packets are sent by using the configuration that is derived from the genes of the rest individuals till the first generation is completed. In the mean time, the receiver measures the delay, jitter, and throughput and counts the lost packets for every group and sends those measurements back to the sender. The sender takes those measurements and calculates the fitness for this individual (this corresponds to the measured group of 16 packets). If all 10 individuals of the current population are used and the protocol has not calculated the next population yet, then in order to avoid a stop ‘n’ wait behavior of the protocol, it continues the communication by using the configuration of the fittest individual that is available. When the fitness evaluation for all individuals is available, GA runs and produces the next generation.

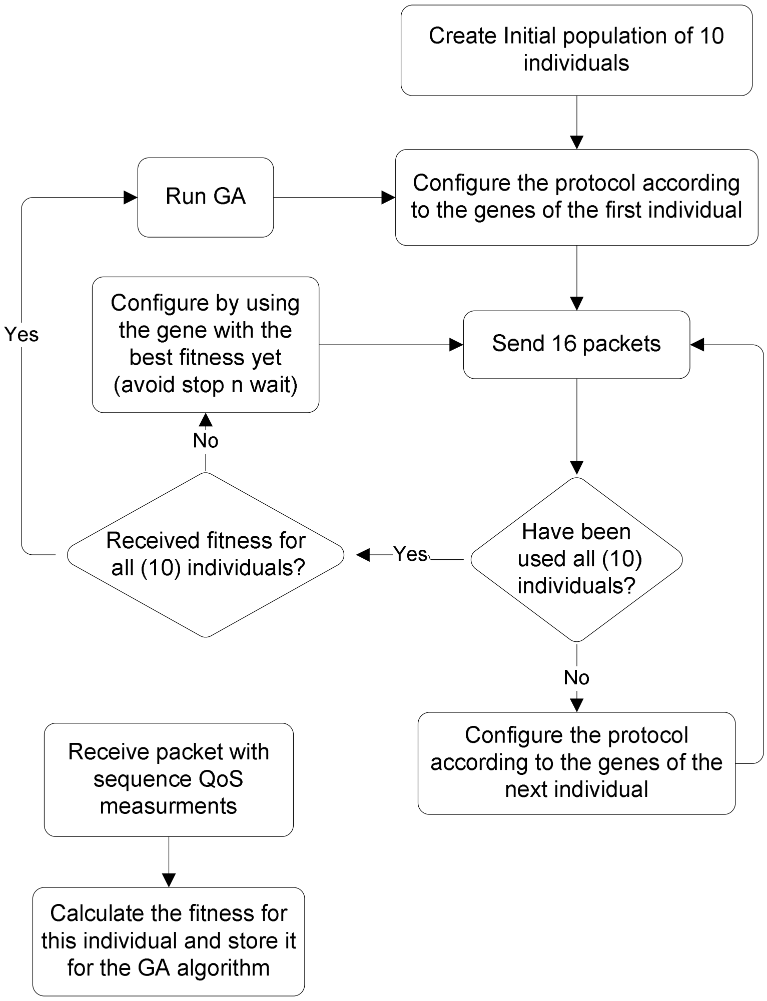

Create Initial population of 10 individuals
Run GA
Configure the protocol according to the genes of the first individual
Configure by using the gene with the best fitness yet (avoid stop n wait)
Send 16 packets
Yes
No
Received fitness for all (10) individuals?
Yes
Have been used all (10) individuals?
No
Receive packet with sequence QoS measurments
Configure the protocol according to the genes of the next individual
Calculate the fitness for this individual and store it for the GA algorithm

Fig. 6. Flow diagram of ESMP instance in the sender node.

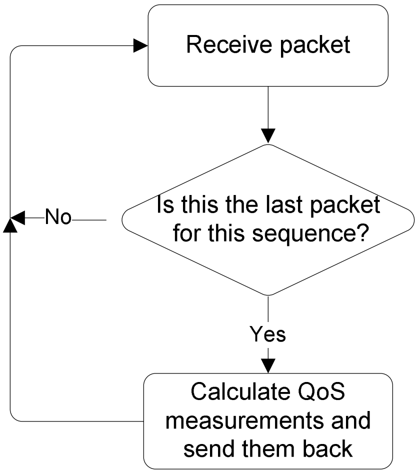

Receive packet
No
Is this the last packet for this sequence?
Yes
Calculate QoS measurements and send them back

Fig. 7. Flow diagram of ESMP instance in the receiver node.

3.5.2 Genetic information encoding

The genetic information is encoded in chromosomes that are a combination of genes. Genes are represented by using a binary system. Each chromosome represents a solution (different configuration in our problem).

Our problem solution used a 6-bit chromosome that is the concatenation of two 3-bit genes, one for the packet size and the other for the time interval, as defined in TABLE I and TABLE II, respectively. Note that the interpacket gap is analogous to the MTU packet transmission time. In our simulation, one unit corresponds to 0.009s. When the chromosome equals 111000_2, the protocol will transmit packets of 1500 bytes at every 0.009s. In this case, the throughput is calculated as 1.33 Mbps and that is the highest transmission rate for this configuration. Another example, when the chromosome equals 000111_2, the transmission rate would be 6.35 kbps.

TABLE I Packet Size

Encoding Binary	Representative value
000	100 bytes
001	300 bytes
010	500 bytes
011	700 bytes
100	900 bytes
101	1100 bytes
110	1300 bytes
111	1500 bytes

TABLE II Time interval (0.009sec for each unit)

Encoding Binary	Representative value
000	1 unit or 0.009s
001	2 units or 0.018s
010	4 units or 0.036s
011	6 units or 0.054s
100	8 units or 0.072s
101	10 units or 0.09s
110	12 units or 0.108s
111	14 units or 0.126s

3.5.3 Population

After deciding how to represent the chromosomes, we have to select the population. There are two issues which need to be considered: 1) the size of the population that is appropriate for the problem we have defined, and 2) the actual individuals in the initial population. Intensive researches have focused on how to calculate the population size [Ahn, C. W. (2002)], [Reeves, C. R. (1993)]. Large populations can spread more in the search space and produce better solutions, but on the other hand, it increases the computational cost. In our problem, as was mentioned before, bigger population translates to longer periods of inadequate communication.

In [Reeves, C. R. (1993)], the authors suggest that, every point in the search space should be reachable from the initial population by crossover only. This means that there must be at least one instance of every allele at each locus in the whole population. When the initial population is generated by a random sample with replacement and the chromosomes are binary encoded, the probability, for the above requirement to be satisfied, can be expressed as [Reeves, C. R. (1993)]:

$$P = \left(1\text{-}\left(\frac{1}{2}\right)^{N\text{-}1}\right)^{l} \qquad (1)$$

where N is the population size and l is the

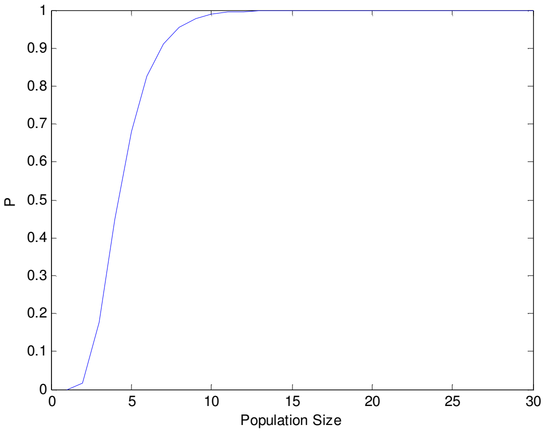


Fig. 8. Population size versus probability that every point in search space is reachable from the initial population by crossover only.

TABLE III A Sample of Probability versus Population Size

Population Size	Probability
6	0.8266
7	0.9098
8	0.9540
9	0.9768
10	0.9883
11	0.9942
12	0.9971
13	0.9985
14	0.9993

=1-12N-1l (1) can be seen in Fig. 8 and for detail purposes selected values can be found in TABLE III.

3.5.4 Fitness evaluation

The fitness function computes a fitness value for each individual in the population. As we have seen, the parameters that the protocol tries to reach a certain target value defined by the application are the throughput, delay, jitter and percentage of lost bytes. The more the measured values of these parameters are close to the target values, the better the fitness value will be.

The fitness for the delay when d is higher than D is calculated by:

$$F_d = 10 \times \frac{D}{d} \quad (2)$$

For jitter when j bigger than J

$$F_j = 10 \times \frac{J}{j} \quad (3)$$

For throughput when t smaller than T

$$F_d = 10 \times \frac{t}{T} \quad (4)$$

For lost bytes when l bigger than L

$$F_l = 10 \times \frac{L}{l} \quad (5)$$

where D, J, T, L are the target delay, jitter, throughput and percentage of lost bytes, respectively, and d, j, t, l are the measured values for the same parameters. The return fitness value would be higher when the measured values are close to the target values. In any other case that the measured value e.g. d. is better than the targeted value e.g. D, then the fitness value e.g. F_d is set to 10.

The total fitness for the individual is calculated by the summation of all partial fitness, multiplied first by a weight of significance w, as given expressed below:

$$F = w_d \times F_d + w_j \times F_j + w_t \times F_t + w_l \times F_l \quad (6)$$

where the weight can be used to regulate the priority for each metric and give a higher or lower priority over the other metrics.

3.5.5 Individuals selection for reproduction

Selection is the procedure of the GA where individuals from the population are chosen for breeding. The goal is to improve the average quality of the population by transferring the fittest chromosome to the next generation. In that way the GA focuses in areas of the search space that are more promising. The selection method that someone can adopt and use in a system can affect the performance of the GA in various ways. Hence, a selection of the fittest individuals helps to reach the equilibrium of the population faster, but on the other hand leads to suboptimal solutions because it sacrifices the genetic diversity.

In the type of problem that our research is focused, the environment changes continuously along with the search space. The optimal solutions maybe are not there for long time; hence the sub-optimal solutions sometimes are a better choice. On the other hand, because of the unsteady environment, GA needs genetic diversity to explore the new search space quickly and not stay locked to an area waiting mutation to give the required genetic diversity. The selection methods that we are using in our experiments are the following:

- Rank selector: is the simplest method as it just picks the best member of the population every time.
- Roulette wheel selector: it picks an individual based on the fitness score relative to the rest of the individuals. Individuals with higher score are more likely to be selected. The probability for an individual to be chosen equals to the fitness of the individual divided by the sum of all fitness values in the population.
- Tournament selector: it runs the roulette wheel method to a few of the individuals in the population and then picks the one with the higher score. Then runs a new tournament until it selects all individuals it needs. The

tournament selector typically chooses higher valued individuals more often than the roulette wheel selector.

- Deterministic sampling (DS) selector: it uses a two-staged selection procedure. In the first stage, each individual's expected representation is calculated. A temporary population is filled using the individuals with the highest expected numbers. Any remaining positions are filled by first sorting the original individuals according to the decimal part of their expected representation, then selecting those highest in the list. The second stage of selection is uniform random selection from the temporary population.
- Stochastic remainder sampling (SRS) selector: it consists of two stages. In the first stage, a temporary population is created by using the individuals with the highest expected representation. Any fractional expected representations are used to give the individual more likelihood of filling a position in the population. For example, an individual with expected representation of 1.2 will have 1 position then a 20% chance of a second position. The second stage of selection is uniform random selection from the temporary population.
- Uniform selector: any individual in the population has a probability, p of being chosen where p is equal to 1 divided by the population size.

3.5.6 Crossover

After the individuals are selected, crossover is used to create the next generation population by choosing a locus (or more than one locus in multipoint crossover) and then swapping the remaining alleles from one parent to the other.

In our experiments, we have two genes in each chromosome so the only choice is to use single point crossover. In other words, when crossover takes place, the child will take one gene from one parent and the other gene from the other, or in real problem terms, will take the packet size from one parent and the time interval between packets from the other one.

Another parameter for crossover is the probability for crossover to occur. Sometimes no crossover occurs and the parents are copied directly to the new population.

3.5.7 Mutation

Mutation is a simple operation in GA. Based on a low probability, a random change in the value of a gene in the chromosome will occur. The purpose of mutation in GAs is to allow the algorithm to avoid local minima by preventing the population of chromosomes from becoming too similar to each other, consequently slowing evolution. Mutation is, however, vital to ensure genetic diversity within the population. On the other hand, there is a risk that a gene will change drastically and degrade the protocol's performance.

3.5.8 Repair function

In the populations that are generated after crossover and mutation, there is a chance that some of the individuals are not capable to meet some of the application requirements. Given the required throughput, it is possible to find out if an individual is going to reach it. By multiplying the size and the number of the packets that are sent at every second, the maximum throughput that this configuration can achieve, can be found. In Fig. 9, we present the theoretical throughput for the whole search space.

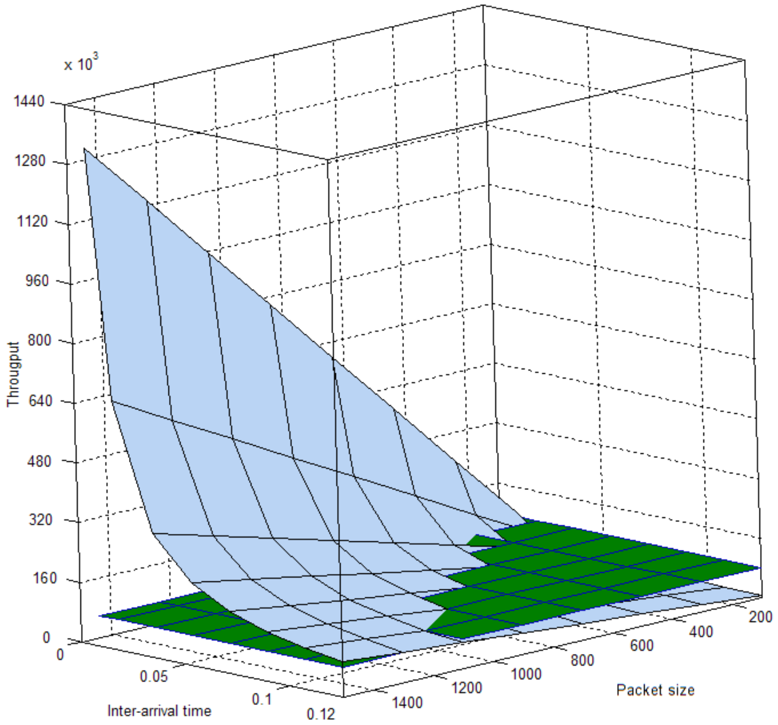


Fig. 9. Theoretical throughput in the entire search space. The horizontal surface separates the throughput surface at 80kbits/sec as an example.

Usually there are two methods to deal with infeasible chromosomes. One is to use a repair function to find and remove the infeasible chromosomes and the other is to punish them by giving a penalty. By giving a penalty the probability that the infeasible chromosome reproduces will be lower but the current generation will suffer from the consequences and there is the risk that they will be in the next population. By using a repair function, the infeasible chromosomes can be eliminated even before they will be used. By eliminating chromosomes, some of the genes maybe will disappear and will not appear in the future generations, except in the case where mutation creates them. This will help the feasible genes to dominate the population. This conceals a risk though, because there might be some genes that give better results for delay and jitter but are eliminated because they were coexistent with other genes in infeasible chromosomes. The chance for those genes to appear in the population is very low.

In order to address this problem, we use a mechanism that considers the importance of the throughput for the application. If the throughput is not so important then the repair function will not remove a chromosome that is not too distant from the target.

3.6 Experiments and Discussion

3.6.1 Evaluation of the Genetic Algorithm

In order to evaluate the GA in a network environment, we need to know the environment that it is going to operate in, and the fitness optimal values in the whole search space. By knowing the search space, we can compare the GA results. This can be achieved only in a steady environment.

Here we tried to set up a steady network environment such that we will be able to scan the whole search space and find the fitness for each genes combination. Scanning the search space is the process of trying every genes combination and measure how it performed in all the four QoS metrics aforementioned, and the final fitness value generated by the evaluation function $F = w_d \times F_d + w_j \times F_j + w_t \times F_t + w_l \times F_l$ (6).

Fig. 10 shows the network setup with 6 nodes and the flow of data. Node 0 sends constant bit rate (CBR) traffic to node 3 through nodes 2 and 5 which are the relay nodes. Node 4 also sends CBR traffic to node 3 through node 5. The nodes are not moving and the traffic is relatively constant. In this environment, node 1 sends data using ESMP to node 3 through node 2 and 5 as the relay nodes. First, every genes combination will be tested in order to scan the search space. Next the traffic will be shaped by our experimenting protocol that will try to adapt in the environment by using GA.

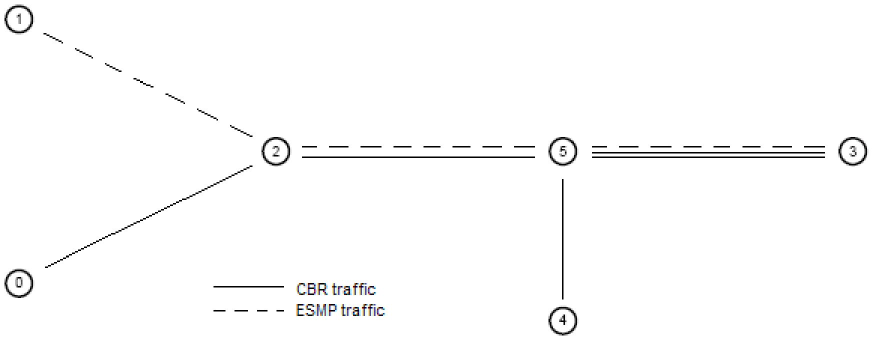


Fig. 10. Layout of network for simulations in a steady environment. Node 1 sends ESMP traffic to node 3 through nodes 2 and 5 (3 hops). Nodes 0 and 4 send CBR traffic to node 3.

3.6.1.1 Characteristics of the environment

The attempt to operate in a steady environment is not as easy as it sounds. Even if the traffic in the network is really constant and the movement of the nodes is zero there will always be a variable element. This element is the traffic of ESMP protocol which changes all the time in the process to find the best configuration. The state of the network, when a new configuration is tested, depends on the previous configuration. For example, if the previous configuration created heavy traffic then the results of this heavy traffic will affect the network's conditions for some time (for example full buffers) and consequently will affect the performance of the next configuration. In other words, we have a stateful system.

Because of the stateful nature of the problem, the way we scan the search space will affect the scan results. This can be seen in TABLE IV where the search space is scanned by trying chromosome 0 to 63 and then from 63 to 0. In the table, we present only the chromosomes that are found to have substantial difference in fitness for each scanning. Therefore, we cannot know in advance which chromosome is better for a specific network situation.

TABLE IV Chromosomes that give different fitness for two different scanning sequences

Chromosome	Fitness A	Fitness B
8	72	80
15	64	54
17	80	76
19	62	74
23	66	52
26	72	80
27	70	80
28	66	76
31	68	50
34	60	80
35	58	80
36	66	80
39	72	48
40	62	54
41	54	62
43	66	80
44	58	80
45	62	80
47	74	44
48	64	50
50	54	78
51	60	80
52	52	80
53	56	66
54	64	76
55	76	44
56	60	48
58	46	66
59	64	78
60	28	80
61	46	80
62	54	80
63	64	80

3.6.1.2 Selection method evaluation

Next is the comparison of various selection methods in terms of how quickly the algorithm converges and how much is the genome diversity in the population at that time.

The diversity of the entire population is just the average of all the individual diversities. So if every individual is completely different than all the others, the population diversity is higher than zero. If they are all the same, the diversity is zero. The probability of mutation is set to zero, to make sure that the diversity is not caused by mutation.

The ideal selection algorithm must converge very fast, achieve the maximum fitness and the diversity of the population must be high. Usually in optimization problems which use GA, the diversity is not an important parameter after GA has run, but is important for the initial population. This is because once the optimum solution is found, the algorithm terminates and the problem is solved. But in our case the environment always changes and the diversity of the population must be high enough for the GA to be efficient and continue to adapt repeatedly after any change in the environment.

TABLE V Selection methods comparison for mutation rate zero

Selection Method	No of generations	Fitness	Diversity
Rank Selector	1	80	0
Roulette Wheel	7	68	0
Tournament	6	80	0
DS Selector	5	62-63	0.14-0.2
SRS Selector	9	80	0
Uniform	2	62	0.3

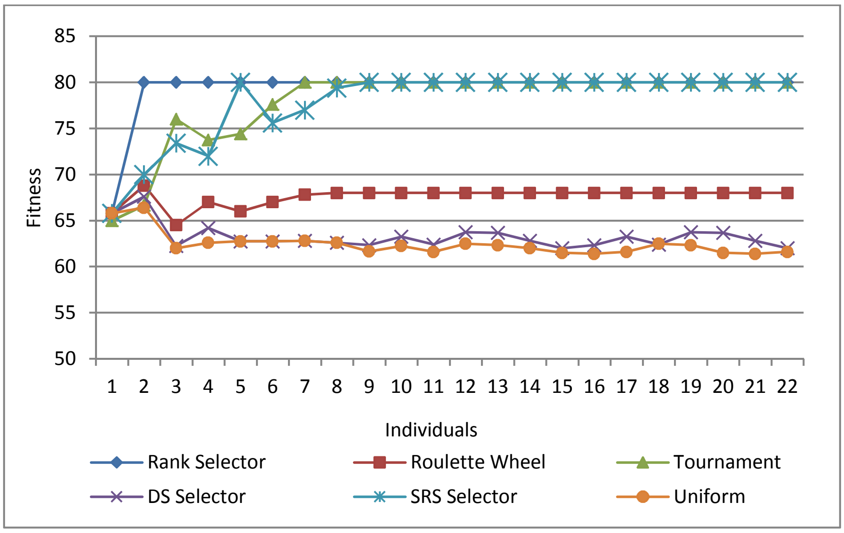


Fig. 11. Average fitness of the first 22 individuals for 6 different selection methods when mutation is zero.

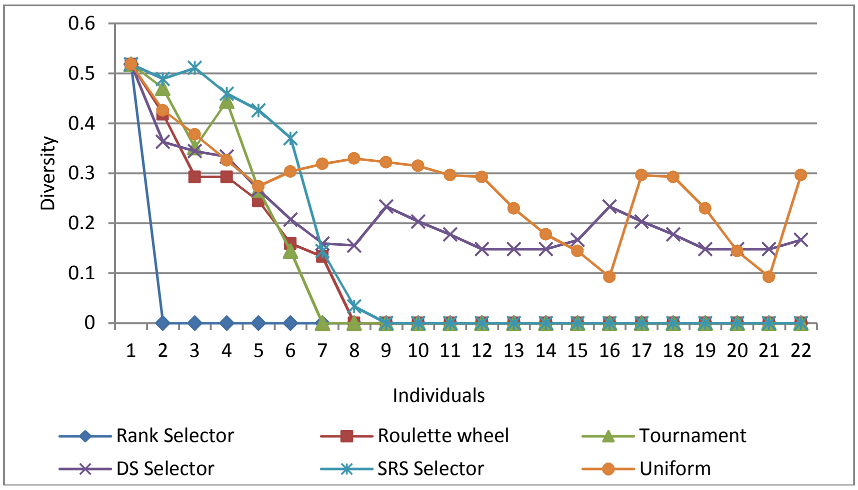


Fig. 12. Diversity of each population for the first 22 individuals for 6 different selection methods when mutation is zero.

From Fig. 11 and Fig. 12, we can see that Roulette wheel, DS and Uniform selector methods failed to reach optimum performance, specifically the Roulette wheel is converged in a low fitness while population consists of individuals with the same genome. DS and Uniform are stabilized in a low fitness but the diversity is not zero; that means the individuals are different. The other three methods, Rank, Tournament and SRS, achieved good quality chromosomes very fast, in particularly, the Rank selector is the faster. But as soon as these selectors found the perfect chromosome, then this chromosome dominates the population and the diversity became zero. Clearly there is a tradeoff between fitness and diversity without any satisfactory solution in between. Mutation can help the algorithm to produce good quality chromosomes without sacrificing the diversity. Next, the effect of mutation in the GA performance is investigated and discussed.

It is noticeable in the performance of the GA when mutation is used as it tends to force the protocol to jump out of local areas in search space. The performance of the network, as it is reflected by the fitness, is not stable. Some selection methods are stable enough but others are not stable at all. As expected, the diversity of the genomes in each population is much higher and exists almost all the time. As mutation increases, this behavior is becoming more intense. This was what we expected because this is exactly the reason that mutation is used in GA. Among the selection methods that are used, Rank selector is clearly the most stable, with the diversity becoming better as generations are passing by, without sacrificing much fitness performance.

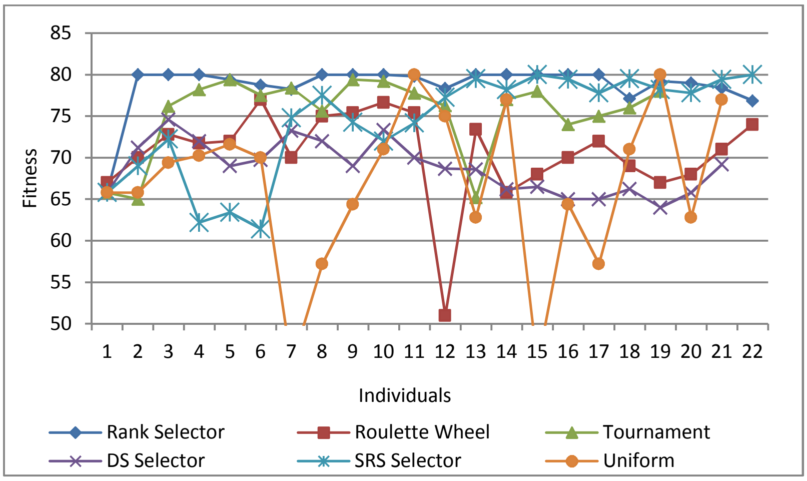


Fig. 13. Average fitness of the first 22 individuals for 6 different selection methods when mutation probability is 0.01.

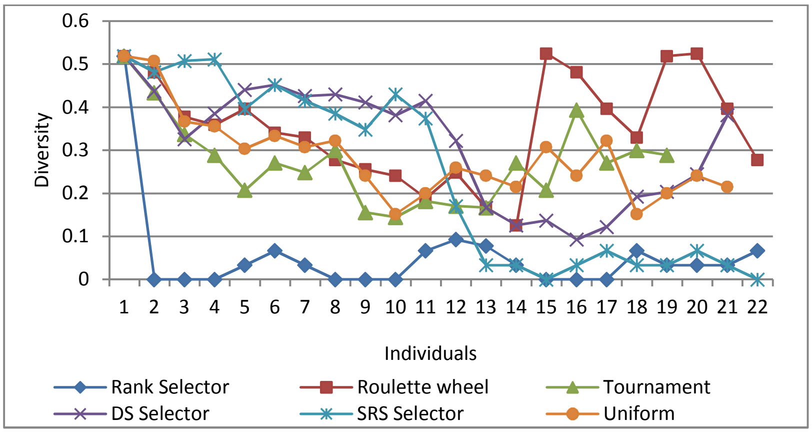


Fig. 14. Diversity of each population for the first 22 individuals for 6 different selection methods and mutation probability 0.01.

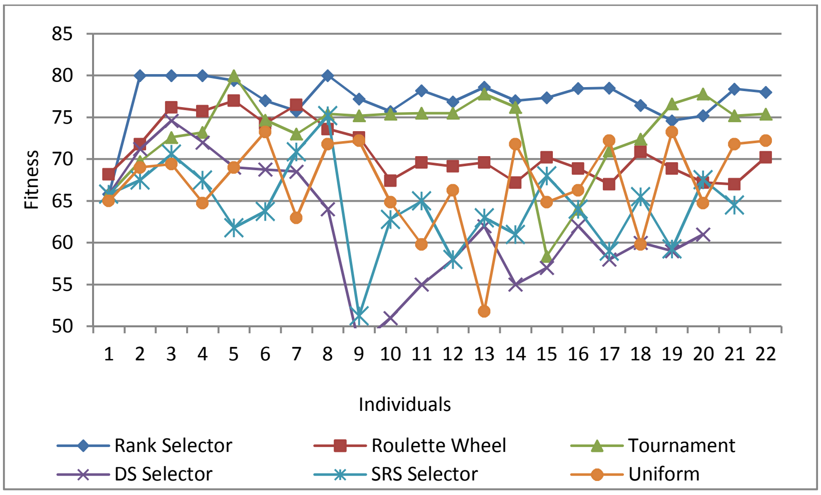


Fig. 15. Average fitness of the first 22 individuals for 6 different selection methods and mutation probability 0.05.

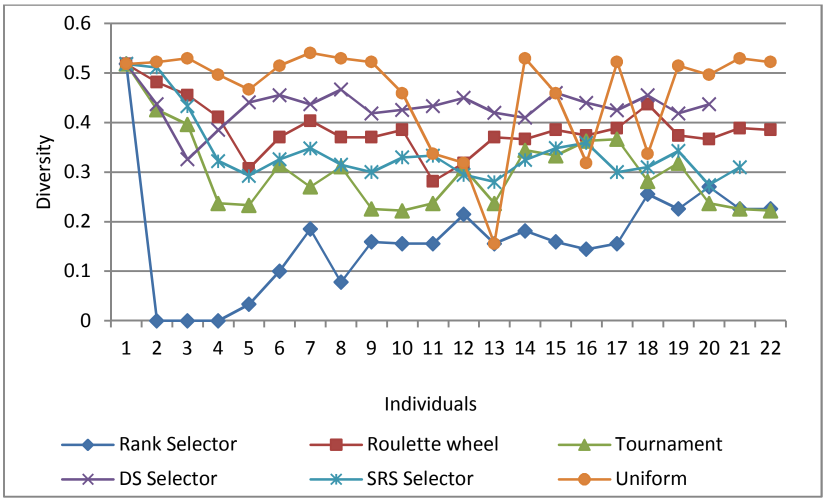


Fig. 16. Diversity of each population for the first 22 individuals for 6 different selection methods when mutation probability is 0.05.

3.6.2 Simulation in a high dynamic environment

After experimenting with GA and the protocol in a steady environment, it is time to try a real world environment that is highly dynamic. A MANET environment is the case.

In the following scenario, there are many nodes that are constantly in movement in an ad hoc network and some of them are producing traffic. Because of the movement, the links are always changing along with the multi-hop communication paths. The different traffic connections sometimes are crossing each other and sometimes they are not. Sometimes the neighbour nodes are sending traffic and occupying a proportion of the communication medium. It is not difficult to understand that the factors that affect the availability of the network, the load at the nodes and the quality of the communication in general (the network environment) are many and in constant change.

The purpose of this experiment is to investigate how the GA can follow the changes in the environment and adapt so the application requirements can be met.

We simulated a dynamic environment which consists of 50 wireless nodes with one IEEE 802.11 interface as illustrated in Fig. 17. All the nodes are moving randomly in an area of 500x500 meters and 10 of them are generating CBR or VBR traffic. Two other nodes are using the ESMP with GA protocol to exchange data between them. In the 100 seconds duration of the simulation, there were 2785 total link changes for all nodes. Through the communication of the two ESMP nodes were 178 route changes. Not all of the route changes affect the performance dramatically.

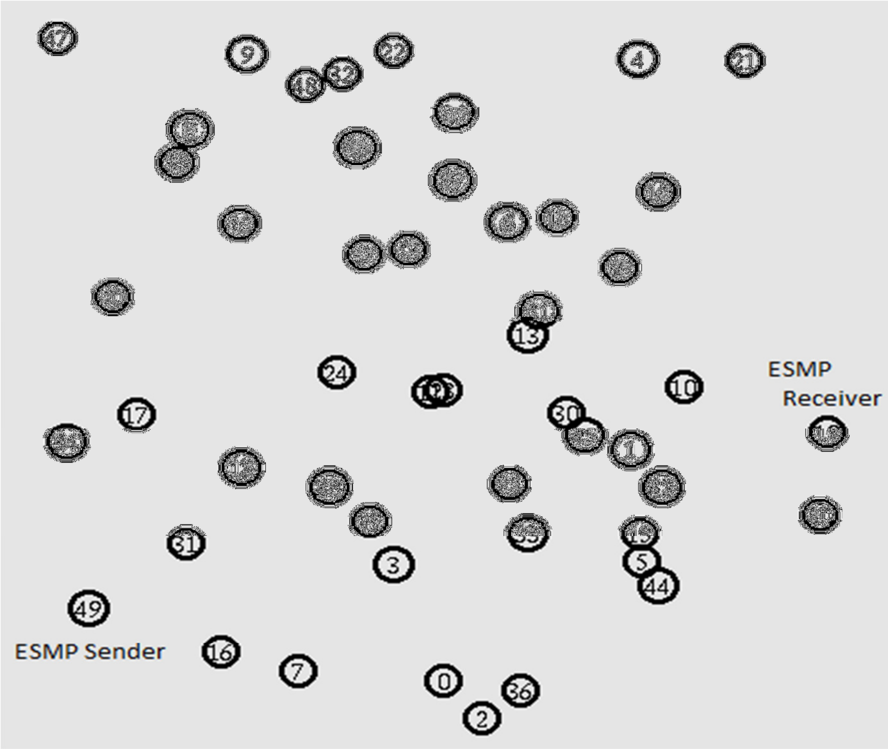


Fig. 17. Snapshot of the mesh network in the dynamic environment.

Here, we experiment with all the selection methods in the dynamic environment. As seen from the Fig. 18 - Fig. 20, the Rank selector appeared to be the most stable with slow but guaranteed adaptation.

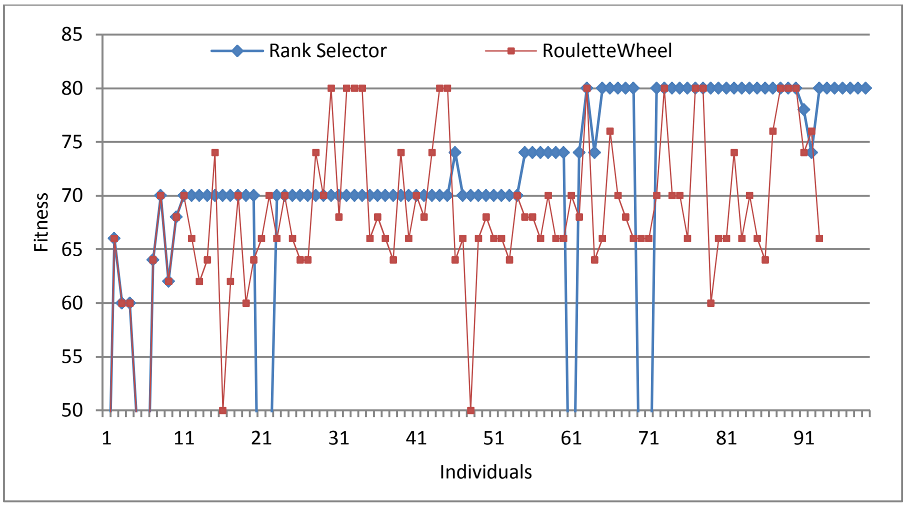


Fig. 18. Fitness performance of Rank and Roulette Wheel selectors in dynamic environment with mutation probability 0.01.

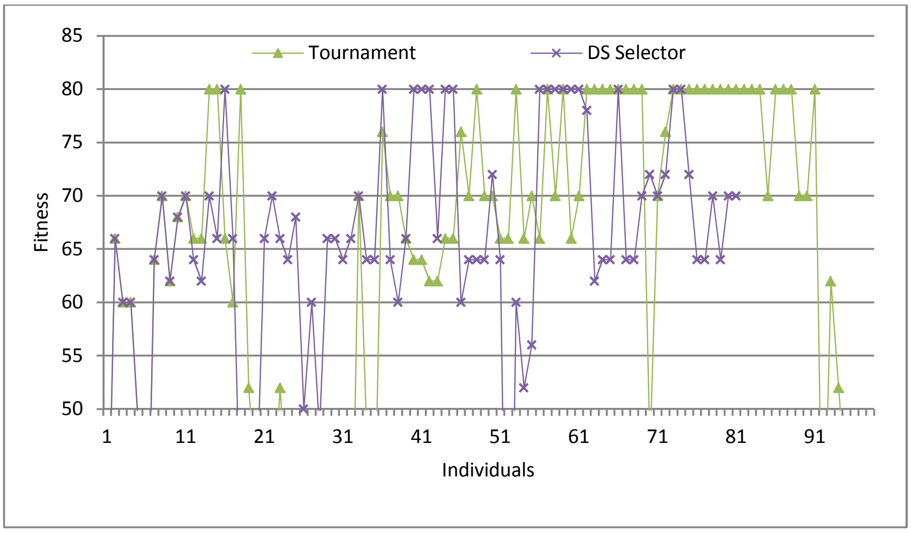


Fig. 19. Fitness performance of Tournament and DS selectors in dynamic environment with mutation probability 0.01.

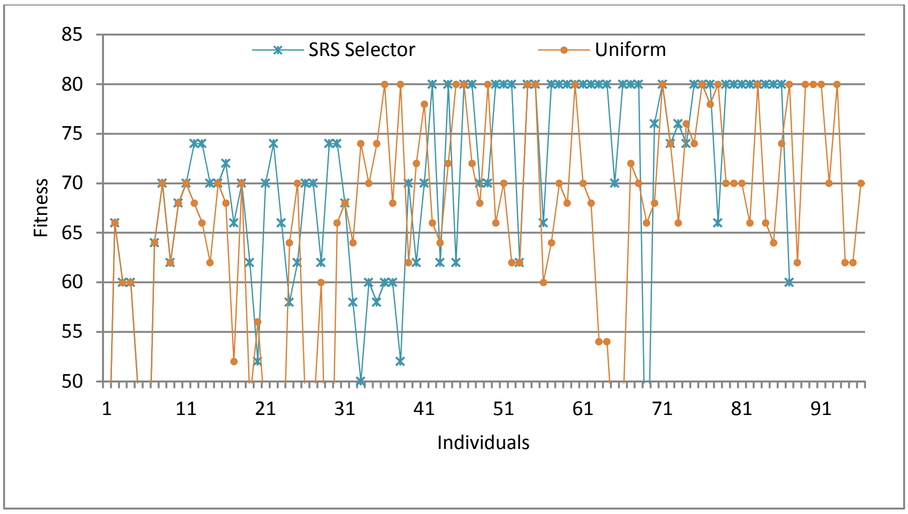


Fig. 20. Fitness performance of SRS and Uniform selectors in dynamic environment with mutation probability 0.01.

The major difference that we observed in the dynamic environment is that for some moments the fitness drops in very low levels in comparison with the steady environment. In the steady environment, the reason was the mutation but in the dynamic environment the reason is the changes that happen in the network. GA adapts every time and changes the configuration to the chromosome that performs better.

Since the Rank selector has a superior performance, it is chosen for the rest of the experiments. Next, we prove that the diversity in the genome is useful in dynamic environments by helping the GA to adapt in the changing environments.

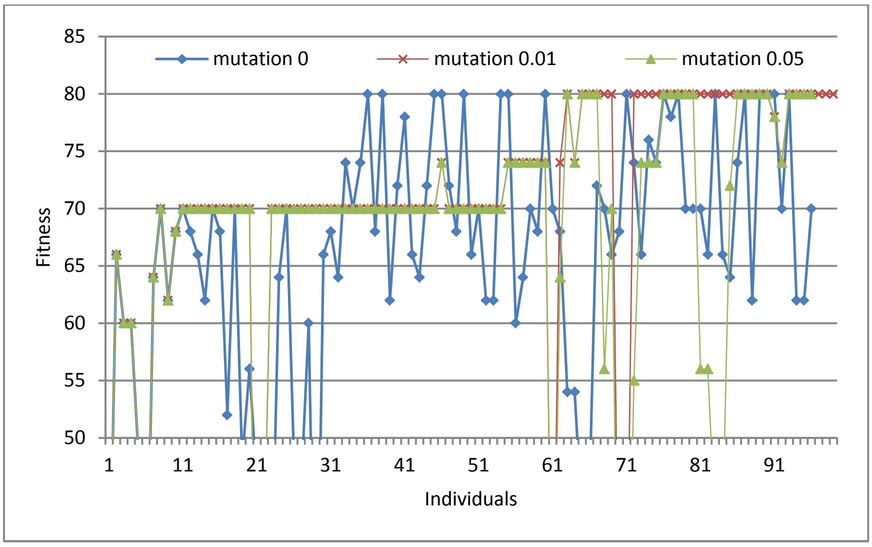


Fig. 21. Fitness performance of Rank selector in dynamic environment for various mutation probabilities.

When mutation is zero the GA cannot adapt and just follows the changes in the environment. By using mutation, the diversity produced is able to provide the GA with enough good quality genomes to adapt quickly to the changes. By further increasing the mutation ratio, an unpleasant effect of mutation is starting to emerge. With higher mutation rate, there is higher probability that genomes will jump to areas of the search space that is not suitable for the current environment.

Next, is presented a comparison of communication's performance in terms of delay and jitter when GA is used and when it is not used. In the case that GA is not used, fixed length and fixed rate of packets are used for the whole duration of the simulation and observe delay and jitter. Throughput is always fixed and all three configurations are chosen to produce the same throughput. Specifically, the three configurations are:

- configuration No.1: 900 bytes – 54 msec
- configuration No.2: 1500 bytes – 90 msec
- configuration No.3: 300 bytes – 18 msec

All three produce 133 kbps throughput. We used the same throughput so we can compare more accurately the delay and the jitter. Finally we compare them with delay and jitter when GA is enabled with a target throughput of 133 kbps.

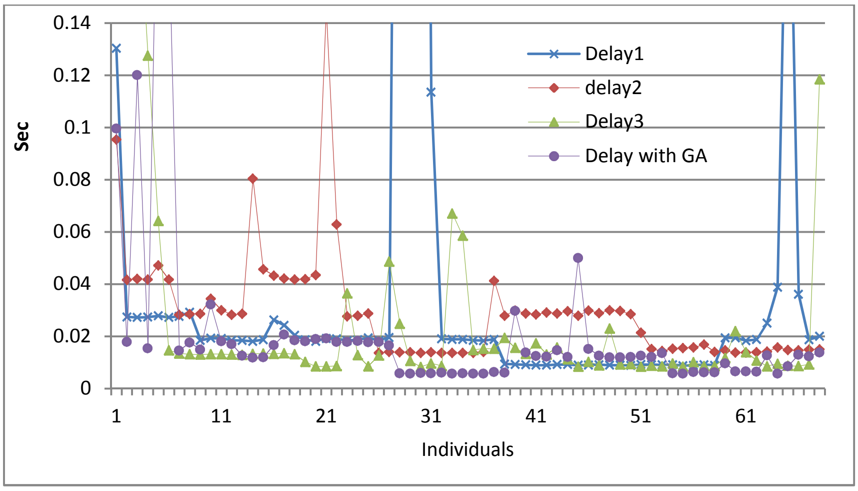


Fig. 22. Delay for the three configurations and when GA is used.

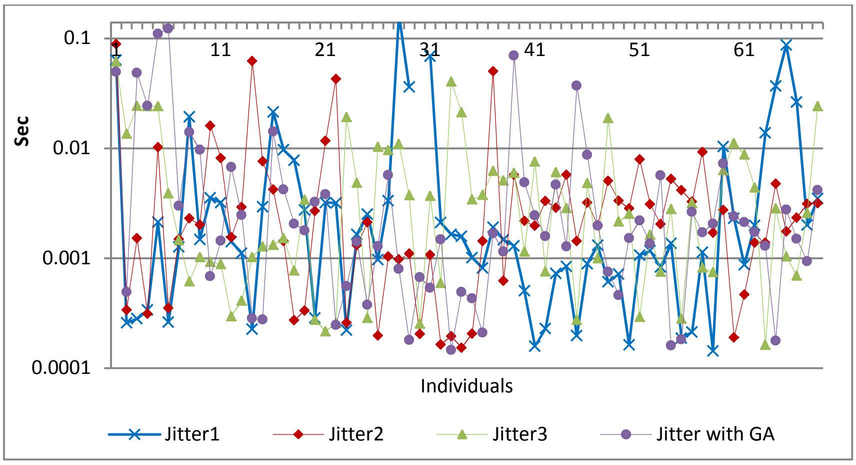


Fig. 23. Jitter for the three configurations and when GA is used.

Fig. 22 and Fig. 23 show the delay and jitter of the three fixed configurations and the one self-configured with GA. It is observed that the GA has the lowest delay and jitter in most of the time. The delay and jitter outcomes of Configuration 2 are the worst among all others. None of the three fixed configurations is the best solution at all times. Clearer performance comparisons can be seen in Table VI and VII which show the average and the standard deviation values of delay and jitter. From the tables, it is observed that the GA self-configured packet size and sending rate achieves lower delay and jitter than any of the fixed configurations. The gain from the use of GA is greater for the delay than the jitter.

In Fig. 24 and Fig. 25, the two histograms represent the density of the values of delay and jitter. When GA is used, most of the values are concentrating in lower valued areas.

TABLE VI Average and Standard Deviation Values for Delay

	Average	Standard Deviation
Configuration No.1	36 msec	108 msec
Configuration No.2	27 msec	19 msec
Configuration No.3	19 msec	20 msec
GA Enabled	13 msec	9 msec

TABLE VII Average and Standard Deviation Values for Jitter

	Average	Standard Deviation
Configuration No.1	6.6 msec	20 msec
Configuration No.2	5.4 msec	11 msec
Configuration No.3	5 msec	9 msec
GA Enabled	4.4 msec	10 msec

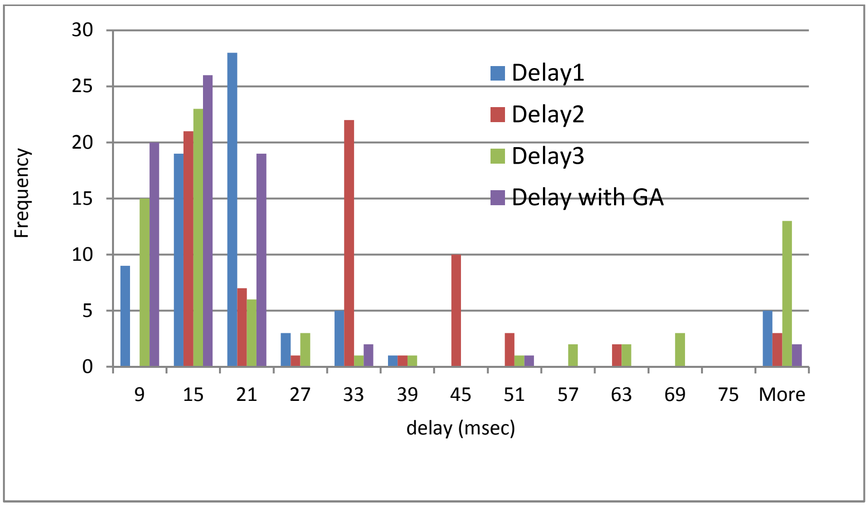


Fig. 24. Delay density for the three configurations and when GA is used.

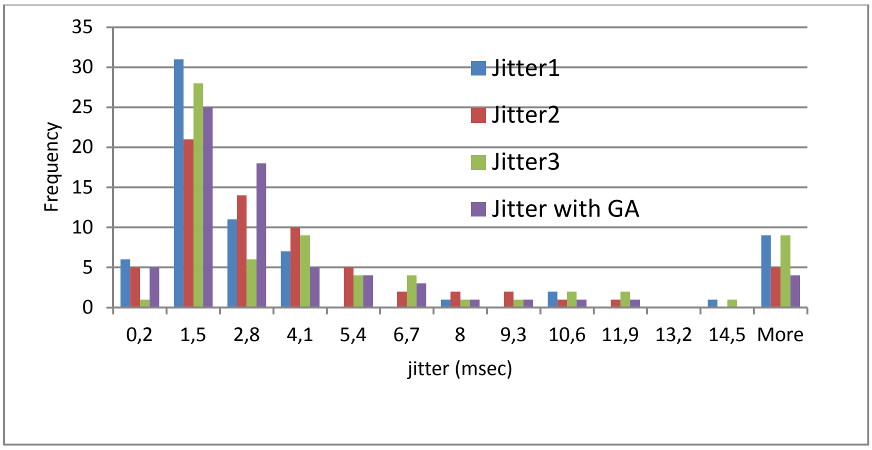


Fig. 25. Jitter density for the three configurations and when GA is used.

It is obvious that when the resources are abundant or the requirements are low, it does not matter which configuration performs better because all of them perform practically the same. Next, we set the network to its limits in order to see how it will perform then. Better performance means lower delay and jitter, higher throughput and less lost packets.

At first, we use application's requirements that are not pushing the limits of the network (non-demanding requirements), for example:

- Throughput: 0.15 Mbit
- Delay: 0.05 sec
- Jitter: 0.02 sec
- Lost packets 10%

From Fig. 26, Fig. 27 and Fig. 28, it can be seen that the throughput, delay and jitter achieve better values than the non-demanding requirements.

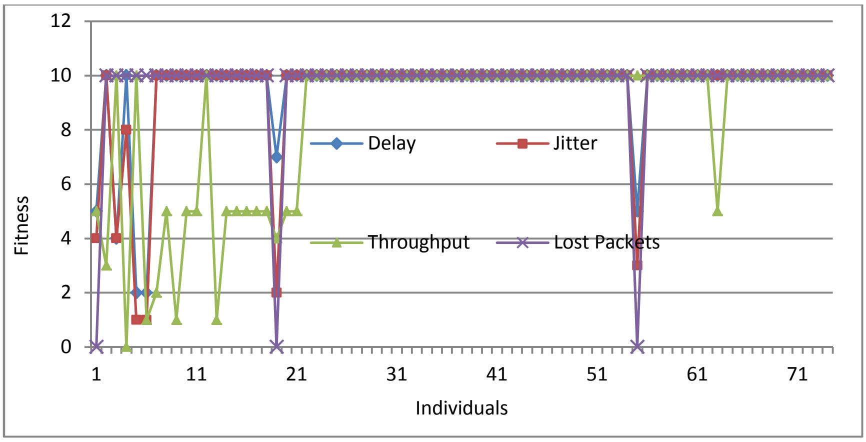


Fig. 26. Fitness achieved for all metrics (not demanding requirements).

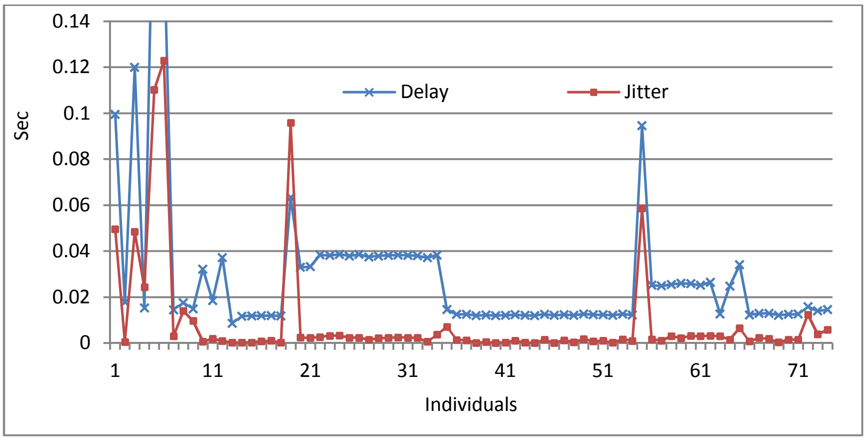


Fig. 27. Delay and jitter achieved (not demanding requirements).

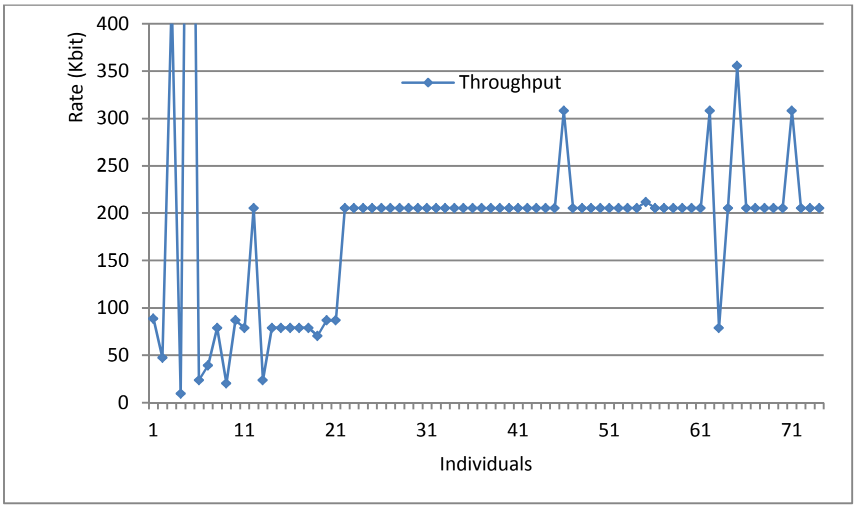


Fig. 28. Throughput achieved (not demanding requirements).

Fig. 29, Fig. 30 and Fig. 31 are the performance outcomes based on stricter requirements which push the network to its limits; for example:

- Throughput: 0.15 Mbit
- Delay: 0.02 sec
- Jitter: 0.005 sec
- Lost packets 5%

The GA-enabled protocol tries to find a solution that can satisfy the requirements and possibly perform better. However, it turns out that the throughput does not satisfy the requirement, but the delay managed to stay below the limits and achieved the required performance.

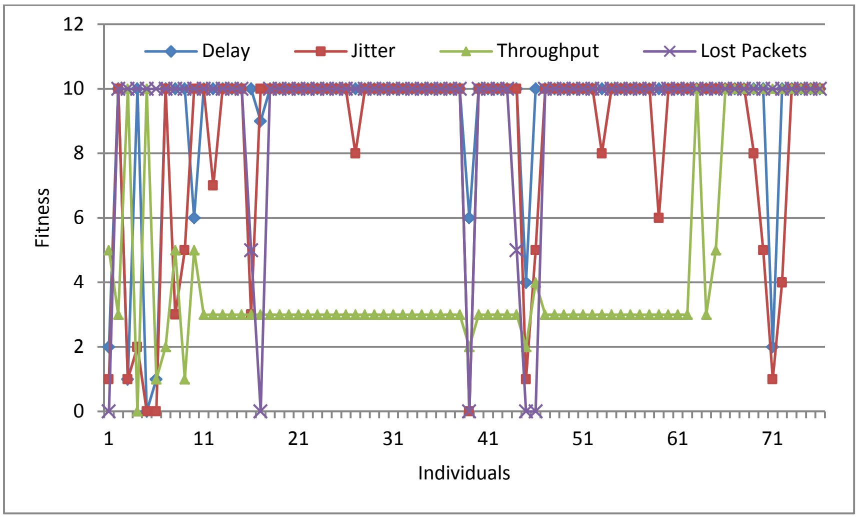


Fig. 29. Fitness achieved for all metrics (stricter requirements).

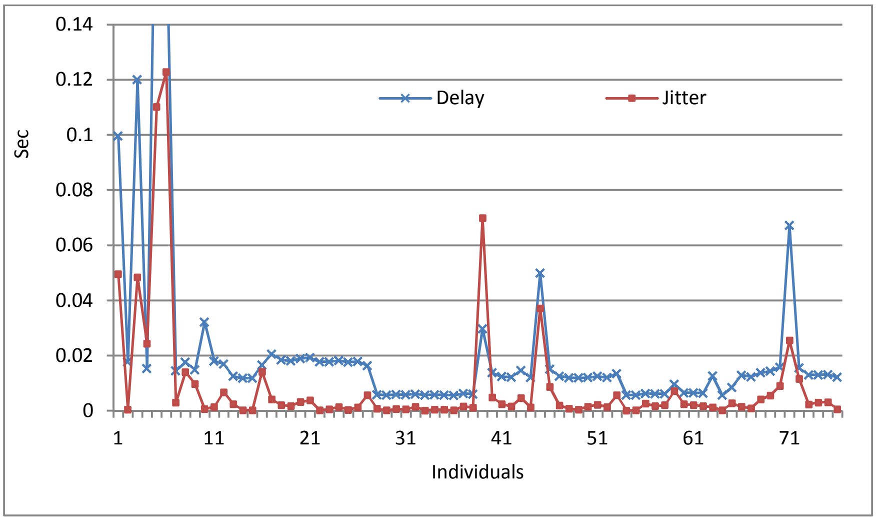


Fig. 30. Delay and jitter achieved (stricter requirements).

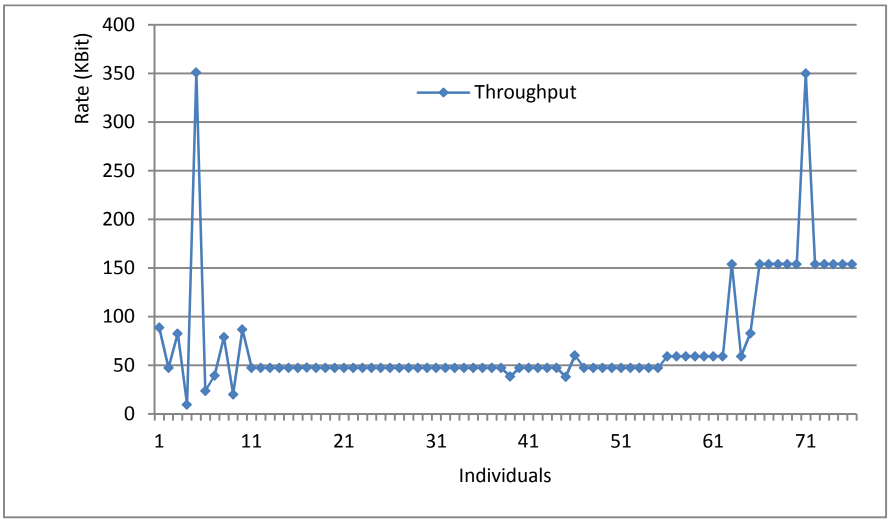


Fig. 31. Throughput achieved (stricter requirements).

In general, we can conclude from all the graphs that there are positive and negative picks with respect to the QoS metrics. These picks are the result of an individual's genome that influences positively or negatively a metric. This individual cannot survive because if it helps the throughput, at the same time it destroys the delay or vice versa, so the overall fitness is poor. Another noticeable issue is that the performance of the first 10 individuals is random and poor. That is the expected behavior of GA due to the randomly selected initial population.

Finally, we carried out experiments based on the priority mechanism of the protocol. First, we gave double priority to the throughput followed by simulations with priority to delay and jitter. Fig. 32, Fig. 33 and Fig. 34 show the performance of the system based on the priorities for the QoS metrics. When higher priority is given for a specific metric (higher than the other metrics), then the genomes that influence the performance of this metric (the one with the higher priority) have bigger probability to move in the next population. In other words, the chromosomes that will receive the higher fitness will be the ones that give a configuration to the protocol that helps the specific metric. It is observed that all of the three metrics performs according to the network conditions. In addition, their performance is related to the priority that has been assigned to them. Throughput and delay are more sensitive and perform better when they have higher priority than the jitter.

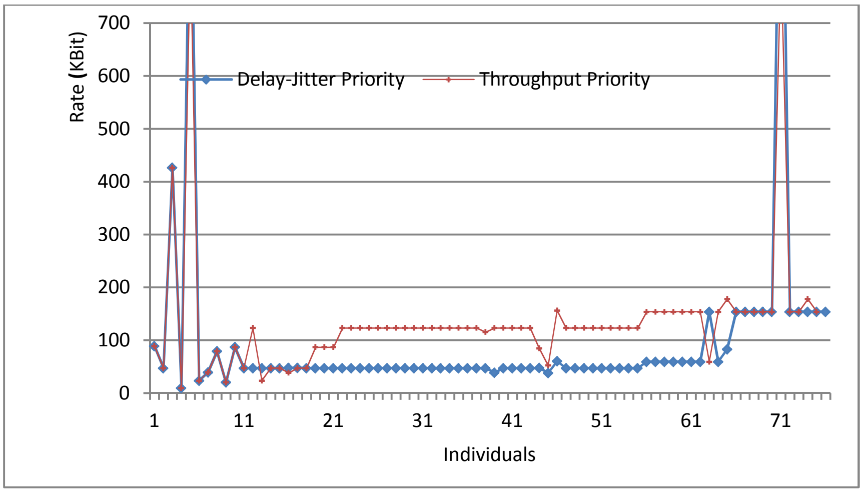


Fig. 32. Throughput achieved when priority is on throughput and when priority is on delay and jitter.

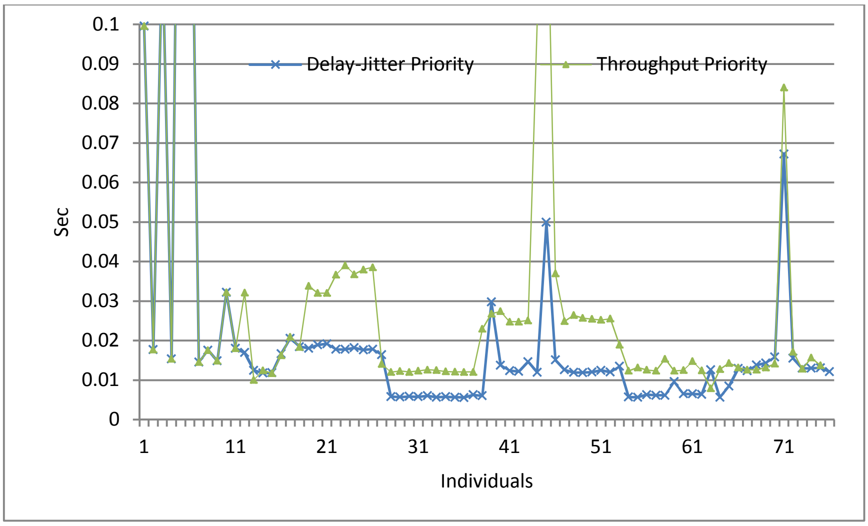


Fig. 33. Delay achieved when priority is on throughput and when priority is on delay and jitter.

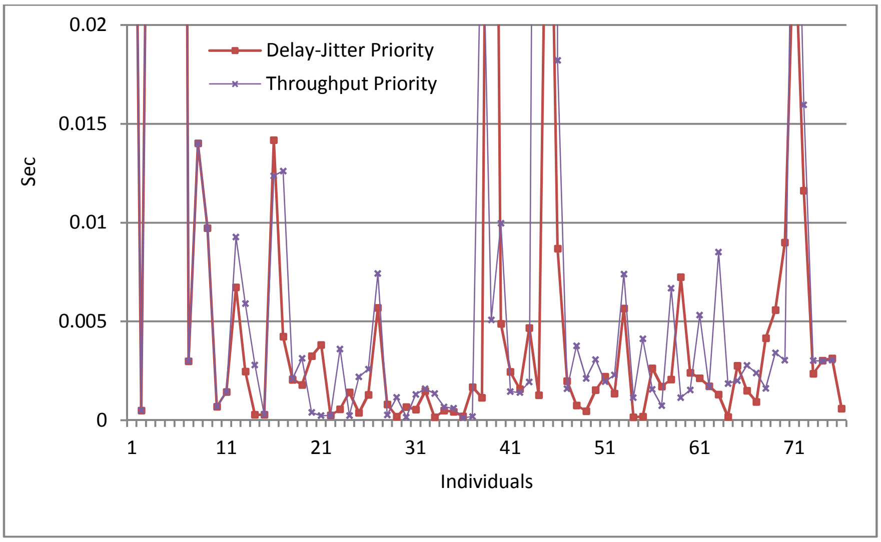


Fig. 34. Jitter achieved when priority is on throughput and when priority is on delay and jitter.

3.7 Summary

In this chapter a system is proposed not only as a congestion control but also to fine tune the flow control in order to improve the QoS required by the applications. This system does not create the conditions, in the way that the lower layers are doing it, for good quality services but takes advantage and adapts to the existing network capabilities for an optimized flow control, in order to improve service quality. We have shown that flow control can be achieved not only by changing the sending rate of the packets but also their length. The way the flow is controlled can affect not only the throughput but also the delay, jitter and lost packets.

We have investigated how well GA can regulate the parameters of the flow control in order to exploit the underlying network conditions and produce the desired results to the applications. We have analyzed the various parameters that GA has and found how to configure them in order to perform better for our problem domain. We have explored this problem domain and we have mentioned the unique characteristics it has.

The NS2 simulations verified the expected behavior of the protocol and have shown that throughput and delay performance can be improved significantly. GA can adapt quickly enough to the network conditions and re-adapt to any changes. This protocol is useful for long term connections.

The effectiveness of a GA can nearly always be enhanced by hybridization with other heuristics. Another technique that can enhance the performance of the GA is annealing the mutation rate. In this method the algorithm decreases the mutation rate by the time as the individuals reach the optimum and increases it again the moment it detects that the population is not good enough for the environment any more.

The case of additional parameters, like battery consumption, can be investigated. If the network is in good condition and the application is not very demanding, then other secondary issues, like battery consumption, can be taken into consideration by the protocol. That can be added in the protocol by measuring the energy consumption for the time period of one individual. The parameter can be added in the fitness function with a smaller weight (priority) than the other parameters so it cannot have significant effect to the selection of the individuals and practically "appear" only when the other parameters are satisfied.

Someone can say that because evolution is a process that needs time to develop, it is a waste to throw away the results every time a new communication is needed. It is interesting to investigate how the performance can be increased if every node reuses the chromosomes that had the best performance, as the starting population in a new communication session or to give this information to a new node that joins the network. The protocol must make sure that this population must have enough diversity to be flexible enough and quickly adapt.

4 Routing decisions by using Bayesian Inference

4.1 Introduction

Many routing protocols have been proposed for MANET by considering criteria for the routing selection mechanism such as hop count, end to end delay, bandwidth, mobility and signal strength but there are not any solutions that combine successfully multiple criteria. Also protocols that take in consideration the breakage lifetime of a path have been introduced. Instead a more realistic solution is proposed here, which targets the performance lifetime of a path and combines multiple criteria. This work presents a model that gives Bayesian inference about the performance lifetime of a routing path by utilizing efficiently cross layer information. The proposed Bayesian Network Routing (BNR) protocol is designed to use this model in order to assist routing selection and route maintenance mechanisms for MANETs.

In fact the aforementioned model represents a path in a MANET. The model can be used to give probability distribution of unknown parameters' states in the path, given the state of observed parameters. Thus the model can be useful for evaluating the paths that are discovered and assist the route selection process. The evaluation of a path is supported by accurate statistical data collected from MANETs during operation. This statistical model can also be used for improving our knowledge about the MANETs and the elements that affect their performance. The mechanism is a flexible cross layer design that can adjust to the amount of available cross layer information.

4.2 Routing

There are three basic network configurations for MANETs, hierarchical, flat and a hybrid that is the combination of both. In a hierarchical configuration, the nodes form clusters. In each cluster there is a node that takes a different role from the other nodes as a cluster head and has the responsibility to coordinate the cluster. The clusters communicate with each other through their cluster heads and every node's traffic must pass through the cluster head to reach the other nodes. In the flat approach on the other hand, all nodes are equal and have the same responsibilities. Nodes can communicate with each other through any node that is available on the network. The source node has the responsibility to find the available paths and select the one that believes that is the most appropriate for its communication needs. Hierarchical configuration helps to organize better the network and manage it easier. Every node is required to forward the traffic to the cluster head which has the liability to find the route to the destination's cluster. But the creation of clusters produces a lot of overhead, especially in MANETs where the mobility is high and the recreation of clusters and selection of cluster heads is needed very often. In addition, the cluster head can become a bottleneck in the communication as it is required to forward all the traffic to the other clusters. In this work we will experiment in flat configuration networks in order to take advantage of the freedom to choose any available path to the destination. Flat routing is further categorized as proactive (table based), reactive (on demand), predictive, ticket based and bandwidth based routing.

A routing protocol can be seen as an optimization mechanism that has as inputs the source and the destination addresses, and as output the route that achieves the best performance, subject to a number of requirements and constraints. There are several challenges for a routing protocol that must operate successfully in a MANET environment.

- The network topology usually is inconsistent and variable. Nodes movement, node failures, link failures due to high interference and fading are some of the reasons that the topology can be inconsistent.
- Network topology information is not enough to determine if a route is suitable to the targeted performance. If the only requirement is the connectivity, then only network topology is the information that is needed. But if other

communication metrics are considered, like delay, then the routing path selection is not related only to network topology but is also affected to many other parameters like interference in the channel, congestion in intermediate nodes etc.

- In order to achieve load balancing in a network, traffic distribution has to be taken in consideration. Routing is responsible for the path that the data will follow, hence is responsible to distribute them in a manner that will balance the network load.
- There will be moments that an optimal routing solution is not available. Considering the wildness of the ad hoc network environment and how a route can affect the performance of other routes with many potential conflict constraints, the routing optimization problem can be very complex with no optimal solution available.

Most of the routing protocols that are currently deployed in MANETs are focused on providing basic connectivity and typically support only best effort services. These routing protocols are using the shortest path available, without investigating the case that another longer path (more hops) but with better quality links exists.

4.3 Bayesian Networks

A Bayesian network or belief network is a model. This model reflects the states of a system. The model can be anything; it can be a car, a human body, an ecosystem, a stock market, generally anything in the world. Our target is to create a BN that represents the communication between two peers in a MANET. Practically, a Bayesian network is a probabilistic graphical model that represents the joint probability distribution for a set of variables via a directed acyclic graph. The variables are some characteristics of the system that the Bayesian network models. For a car, the variables can be the car engine, the tyres, the speed, the car's age etc. A Bayesian network that models a car represents the probabilistic relations between the variables of the car we just mentioned. Given the state (condition) of the car engine and tyres variables, the network can compute the probability distributions of the speed of the car and the age of the car variables or vice versa. Typically some states will tend to occur more frequently when other states are present. Thus, if you are sick, the chances of a runny nose are higher. If it is cloudy, the chances of rain are higher, and so on.

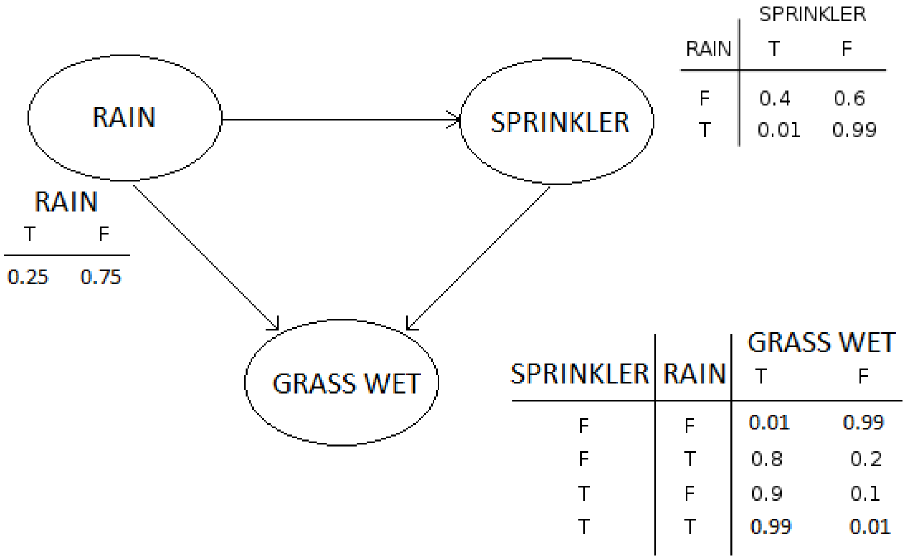
Bayesian Networks are robust for situations where there are missing input data. In this case, Bayesian networks will make the best possible prediction by using whatever information is presented. This is possible because BNs encode the correlation between the input variables. The more information you supply to them, the more accurate the results will be. This is one of the main advantages over other data analysis representations like rule bases, decision trees and artificial neural networks; or over other data analysis techniques like regression, clustering, classification and density estimation.

In order to construct a Bayesian network, for modelling any system, we first need a set of variables $X = \{X_1,\ldots,X_n\}$ which are going to be the nodes in the BN. Those nodes need to be connected to create a structure S of the network which simply encodes a set of conditional independence assertions about variables in X. Suppose a node X_i whose condition directly affects the condition of another node X_j then there must be an arc from X_i to X_j to encode this impact. X_i is called the parent of X_j. Each variable X has a probability distribution table indicating how the probability of X's values depends on all possible combinations of parental values. For example, if the parents are n Boolean variables then the probability distribution table will have 2^n entries, one entry for each of the 2^n possible combinations of its parents being true or false.

Let's consider that nodes are discrete random variables. The values that can take should be both mutually exclusive and exhaustive, which means that the variable must take on exactly one of these values at a time. Common types of discrete nodes include:

- Boolean nodes, which represent propositions, taking the binary values true (T) and false (F).
- Ordered values, low, medium, high can represent how much full a buffer is.
- Integral values, for example a node called Age might represent a car's age and have possible values from 1 to 30.

The classical example model of rain, sprinkler and wet grass is presented in Fig. 35. By examining the BN someone can find out which variables affect which others.



	SPRINKLER	
RAIN	T	F
F	0.4	0.6
T	0.01	0.99

RAIN	
T	F
0.25	0.75

		GRASS WET	
SPRINKLER	RAIN	T	F
F	F	0.01	0.99
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

Fig. 35. Bayesian network that models the relations between three variables, rain, sprinkler and grass wet. All of them are Boolean nodes. We can see that rain has a direct effect on grasses condition and on the use of a sprinkler. In figure we can see the joint distributions and the prior distributions for each variable (collected empirically or from statistical data)

4.3.1 Bayesian Inference

Bayesian inference is an approach to statistical inference on model parameters quite different from classical methods, as all forms of uncertainty are expressed in terms of probability. It treats the unknown parameters of a model, θ, as random variables and not as fixed constants. The criteria of evaluating statistical procedures of estimation are conditional only on observed data and inference is based on the distribution of the parameters given the data, rather than the data given the parameters. Therefore, a prior probability distribution, P(θ), is assigned to the unknown parameters of the model, which represents our knowledge about these parameters. Clearly, it can be realized that Bayesian inference uses a subjective interpretation of probability, as prior distributions vary from person to person, because they depend on people’s belief and experience. The data supposed to be generated by the model provide information about the parameters.

$$P(\theta|E) = \frac{P(E|\theta)\times P(\theta)}{P(E)} \qquad (7)$$

The prior distribution and the likelihood function, P(E|θ), are combined through $\theta E= P(E/\theta)\times P(\theta)P(E)$ (7), in order to produce a posterior distribution, P(θ|E), from which we are able to update our prior beliefs about the parameters of the model, or compute predictive distributions for future observations.

Because a Bayesian network is a complete model for the variables and their relationships, it can be used to answer probabilistic queries about them. Thus, it can infer about variables that cannot be observed, by relying on the variables that can be observed. The more observations available, the more reliable answer it produces.

4.4 System design

In this section we will describe how BNs can be useful for the selection of the routing path in a MANET. In a route discovery mechanism several routes can be discovered. Usually the routing algorithm selects the route with the smaller cost, where cost can be the number of hops, smallest delay, the more throughput, most stable etc. A BN can be asked to give probability of success for each route that has been discovered, like what is the probability that a specific path will provide the QoS that the application requires or give the probability that this path will be stable long enough. The route that has the highest probability to succeed will be selected for the communication.

A Bayesian network (BN) is a statistical tool that can learn from previous events. Then by observing current data it can give inference about related events. Given the metrics for a discovered path in a MANET, the BN can be used to compute the probability of this route to be stable. As we have mentioned before, BN are very tolerant to missing data so if a metric for a path is not available, the BN will make the best possible inference by using only the available metrics.

So in order to use Bayesian inference for this purpose three things are needed. A BN designed to model a path in a MANET, statistical data to create the joint distributions and a routing protocol designed to collect the observed variables for each discovered route. In the following sections we describe those three elements along with the variables that represent BN’s nodes.

4.5 Metrics representing BN’s nodes

The unique characteristics of MANETs invalidate existing solutions from both wired and wireless networks and impose unique requirements on designing routing metrics for mesh networks. More can be found in [Yang, Y. (2005b)], [Akyildiz, I.F. (2009)]. The routing metrics that we have used in our design are presented next. The values of these metrics can give hints about the performance of the route the moment that they are collected and in the near future.

Hop-Count is the most classic routing metric and the simplest one that is used in existing routing protocols like DSR, AODV and DSDV. The only information that it provides is how many links exist between source and destination, following a specific route. Sometimes hop count information is enough for finding a good routing path but most of the times there is a need for additional and more detailed information to determine whether a path can achieve good performance or not.

- Path Delay is another metric and basically it is the summation of the delays in each link that comprise the path. Path delay metric captures the packet loss ratio in the links, the queuing delays and the load in the nodes, and the contention status in all neighbouring nodes near the path area.
- Data rate for a path is the smallest data rate of all links in the path. Data rate is a concave metric because the link across the path that has the smaller data rate will be the bottleneck of the entire route.
- Loss ratio for the whole path is the product of the loss ratio in each link in the route.
- Expected Transmission Count (ETX) is the expected number of transmission attempts before a packet is successfully delivered on a link. The ETX or the whole path is the sum of the ETX on each link in the path. Packet loss ratio, collision probability and link quality affect the value of ETX.
- Frame transmission efficiency (FTE) of a node is defined as FTE = $2/(f_{ACK} + f_{RTS})$ where f_{ACK} is the Acknowledgment (ACK) failure count and f_{RTS} is the Request to send (RTS) failure count. The FTE for a path is the product of the FTEs of all nodes in the path.
- Node density is the number of nodes that can interfere with the channel in the area of a link. For the whole path, this metric is simply the summation for each link. Node density can affect the communication both in a positive and negative way. If the density is high, then it is easier to have local route repairs

when the routing protocol supports this procedure but on the other hand high density means higher interference and lower channel capacity.

- Relative mobility or the similar Link life expectation (LLE) is a metric that captures the trend of the link to stay up or not. For example if the nodes are drawing away from each other then the link tends to become weak and eventually the link will drop.
- Signal to noise ratio (SNR) for a link is a metric that can give a good estimation on the physical distance of the two nodes and the strength of the signal.
- Node related information like buffer usage and processing usage can give an indication on how much the delay will be. This metric will be additive for each node in the path.

4.5.1 Performance stability routing metric

There are also metrics that cannot be measured before the communication takes place and the path is used, like future delay, throughput, jitter and route lifetime. All these metrics are combined in one parameter that denotes the route performance stability. BN is called in to calculate the probability distribution of the routing performance stability for specific time duration and assist to the routing decision. Performance stability is a metric that denotes how stable the performance of a path is. Till now some of the routing protocols target only the link lifetime without considering the overall performance lifetime. There is no point in using a path that is still alive but cannot offer the required QoS. Let’s define as T_p the minimum time in seconds that is desirable to have stable performance in the path. Performance stability is a metric that includes the delay and throughput performance of the path in the next T_p seconds of the lifetime of the path. Path’s performance stability depends on how much the delay and the throughput will change along with the probability that the path will break or not after T_p seconds.

4.6 Bayesian Network routing protocol design

The basic mechanism for a route discovery protocol that is usually used in protocols like DSR, involves the following steps.

1. A route discovery broadcast is sent by the source node. This broadcast contains the network address of the destination node.
2. All neighbour nodes receive the broadcast and each of them checks if it is the destination node.
3. If the node that receives the broadcast is not the one that the discovery is looking for, it stores their network address in the packet and broadcasts the packet again.
4. Eventually the destination node will receive the broadcast packet. Possibly it will be received more than one time from different paths. Then it sends back a reply addressed to the source node along with the path info that was stored and found inside the broadcast packet by using the exact reverse path.
5. The source node will receive the reply with the sequence of hops that is needed to reach the destination.

By using this mechanism, the only information that the source node has about the routes is the number of hopes. The most direct route will be used although the links might have poor signal quality (and hence a greater chance that retries will be needed) or significant delays due to congestion. Our solution requires from the route discovery mechanism to collect information about the metrics in each link that comprises the path that has been discovered. Every node keeps information about several metrics and when the route reply packet passes each one of them, the node piggybacks this information. The structure of the packet and the details of this protocol is not this thesis main interest thus it is not presented here.

The measured metrics from each path, which has been discovered during route discovery, are fed to the BN model. BN then estimates the probability for each one of the discovered paths to be stable after T_p. seconds. Then the routing decision algorithm decides to use the path with the higher probability. During the communication the packets give feedback of the path’s condition by supplying the routing entity with the new metric values. BN again is used to estimate the condition of the path for the next T_p seconds. If it realizes that the probability is less than a threshold value, a routing discovery is initiated in order to discover alternative paths that have higher probability to be stable.

4.7 BN design

There are two major approaches on constructing a BN. One is the knowledge representation approach (KR) and the other is the machine learning (ML) approach. In KR approach an expert on the subject area to be modelled, constructs a BN by using his experience and knowledge in that area. In machine learning approach the structure of the BN is learned from the data. In our hybrid approach we use the experience from all the previous work that has been investigated about how each metric affects the others in order to create the structure of the BN and at the same time the creation of a correlation matrix is used to justify our design. Correlation matrix is a powerful statistical tool that can give the relation between many variables. For example, given a variable Y and a number of variables X_1 to X_p that may be related to Y, correlation matrix analysis can quantify the strength of the relationship between Y and every X_j. In that way can assess which X_j may have no relationship with Y at all, and identify which subsets of the X_j contain redundant information about Y. Thus, once one of them is known, the others are no longer informative. Next we present the procedure of collecting the required data, correlation matrix results and the BN design.

4.7.1 Statistical analysis

As it is mentioned before the proposed approach uses statistical evidence in order to design the BN and give successful inference. The data for the statistical analysis is produced in a simulated environment. A modified DSR routing protocol is used, which discovers routes in a MANET and using them in a random manner. For every route the values of each routing metric are collected by the route reply mechanism and recorded along with the performance stability of the route after 4 seconds of use. The mean life time of a path in our simulated environment, as it is observed in Fig. 36, is around 4 seconds. Thus, we choose to record the performance of the path after 4 seconds of use (T_p). This way, thousands of path cases are collected.

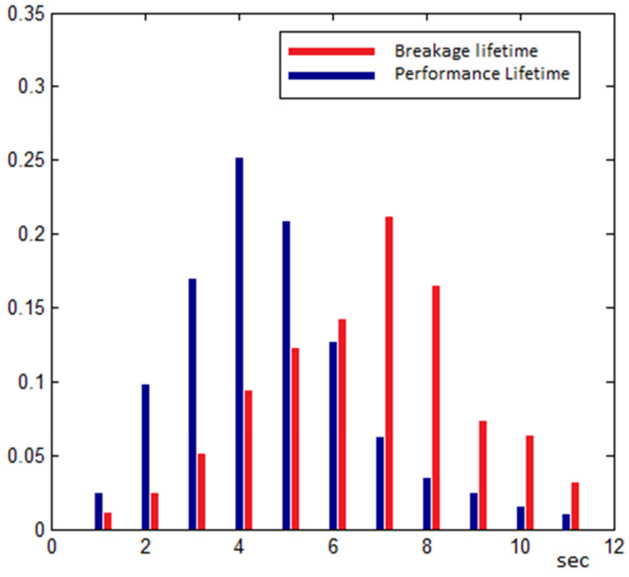


Fig. 36. Probability distribution of a path's lifetime, considering path breakage and path performance degradation.

Table VIII is a correlation matrix for all routing metric variables in the model. The numbers in the table are Pearson correlation coefficients; go from -1 to 1. Closer to 1 means strong correlation. A negative value indicates an inverse relationship (roughly, when one goes up the other goes down)

	Node density	Relative Mobility	SNR	CPU & Buffer Usage	ETX	LOSS ratio	FTE	Delay	Data rate	Hop Count	Performance Stability
Node density	1										
Relative Mobility	0.065	1									
SNR	0.0789	0.0045	1								
CPU & Buffer Usage	0.095	0.055	0.087	1							
ETX	0.657	0.153	-0.786	-0.0278	1						
Loss Ratio	0.528	0.467	-0.845	0.158	0.737	1					
FTE	0.695	-0.225	0.683	0.0845	-0.915	-0.896	1				
Delay	0.457	-0.115	-0.854	0.404	0.744	0.783	-0.853	1			
Data rate	-0.278	-0.215	0.7342	-0.383	-0.296	0.258	0.785	-0.150	1		
Hop Count	0.033	0.025	0.099	0.0845	-0.076	-0.584	-0.392	-0.756	0.637	1	
Performance Stability	0.258	0.684	0.305	0.045	0.578	0.742	0.571	0.801	-0.589	0.872	1

Table VIII. Correlation matrix for all routing metric variables in the model.

From the correlation matrix information on how the routing metrics are related, can be extracted. The arcs between the BN nodes are placed according to this knowledge and the BN is constructed.

4.7.2 Graphical Causal Model

By using general knowledge from the literature and research that has been done till now and by using the correlation matrix that is presented before, the structure of Bayesian Network is constructed and is presented here in Fig. 37. In the graphical model the only query variable that the protocol is interested in is the performance stability prediction.

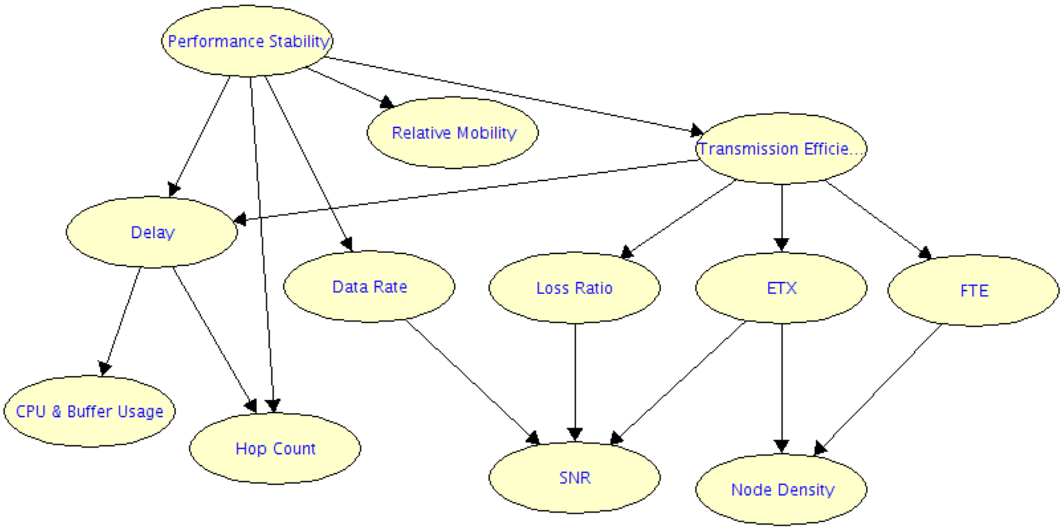


Fig. 37. Bayesian Network structure that models a path in a MANET.

4.7.3 Conditional relationships and Marginal probabilities

The last step for the BN to be ready to function is the creation of conditional probability tables in each BN node. The statistical data are used to train the network and create the required conditional probabilities.

4.8 Experimental Results

4.8.1 Bayesian Network queries

Now that BN is constructed it can be evaluated by making queries and observe the answers. The queries results are compared with the expected behavior that has been investigated in previous researches. New conclusions about path performance in MANETs can emerge by experimenting with the BN. The initial probabilities for each state of every variable can be found in Fig. 38. These probabilities correspond to the general belief of the condition of each metric in a path without observing any evidence yet.

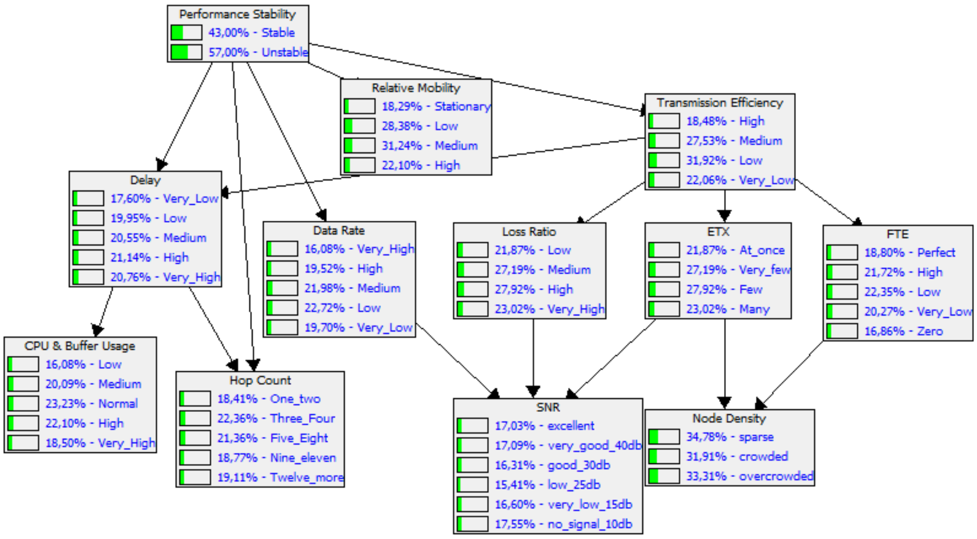


Fig. 38. Initial probabilities for each state of every variable in the BN.

It has been proved in literature that less number of hops usually results in longer distance hops, which in turn are usually not reliable links. While changing the Hop count variable in the BN we observe how the performance stability probability distribution is changing. BN results can confirm the estimation that more hops give stronger links which in turn are more reliable. Also more interesting and complicated points are coming up by experimenting with the model.

In our model delay is related to the number of hops; more hops result in bigger delay because of the additional delays in each hop. Also delay is related to performance stability in the manner of longer delay means less performance stability. Hence these two parameters conflict to each other. By increasing the number of hops the performance stability increases very little and it even starts to decrease when the number of hops is the maximum. But in case that the delay is measured and becomes fixed evidence, the hop count parameter cannot affect the delay but only the performance stability. With delay fixed, changes in the hop count make performance stability probability distribution to change as expected. In Table IX the values of performance stability when delay is unknown and when delay is fixed, can be found.

This can be observed also in Fig. 39 where the hop count evidence is nine-eleven hops, the performance stability is increased along with SNR, Data rate and transmission efficiency (compared with initial values in Fig. 38). These kinds of observations illustrate the complexity of metric's relations and how each one depends on the others. For example, big hop count can denote that the links are strong or not, because the destination can be close or far. At the same time many hops can increase the delay because of the multiple transmissions of the packet but if the frame transmission efficiency is high, then delay can be small even if there are many hops. So it is clear that hop count alone cannot give a good estimation without the decision algorithm to include delay, SNR and other metrics in the decision parameters. It is clear from the experiments that the use of BN to quantify the effect of each metric to the rest of them is beneficial.

Table IX. Hop count affect in performance stability and delay.

Hop count	Performance stability (Delay not fixed)	Delay (very low)	Performance stability (Delay fixed low)
One – Two	40.72 %	18.55 %	54.72 %
Three – Four	45 %	23.05 %	57.71 %
Five – Eight	48.31 %	23.82 %	59.38 %
Nine – Eleven	43.51 %	14.32 %	63.31 %
Twelve – more	36.41 %	6.59 %	65.11 %

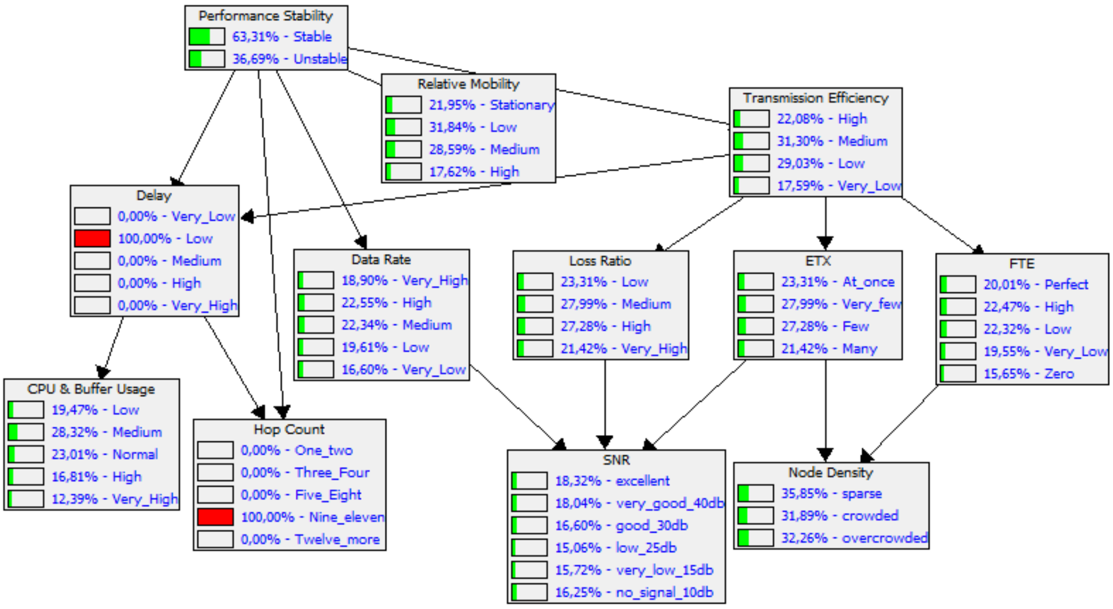


Fig. 39. Probability distributions of the states in every variable of the BN when Hop Count is nine-eleven with fixed delay.

SNR is another example in the BN that highlights the complexity and the need for use of multiple evidences about the conditions of a path, in order to have a better decision. Good SNR promises reliable links but if there is also high mobility then the path is not stable at all. Similarly SNR increases the transmission efficiency but if the area is overcrowded, the transmission efficiency can be low.

In case that the nature of a MANET is very specific, for example if the mobility is very high due to the nature of the network, then the only choice for the protocol is to deal with the mobility and select a path whose other metrics are in a state that counter balances the high mobility effect in the performance.

By observing Table X it is clear that when only one good metric value is observed, it cannot be guaranteed that the path will have good performance even when this value is evident that it is in its best quality state. At this table the probability of the performance to be stable is presented for every metric when it is observed in its best quality state. By looking on the probabilities someone cannot be certain that the path is stable. For example, even when the mobility is in stationary state the BN returns only 67.19% as probability. So there is a good chance that the path is not stable. In order to increase the certainty, more metrics observations are needed. This is evidence that protocols which use very few metrics for their decisions, are very likely to fail very often.

Table X. Stability probabilities when only one metric is observed in its best quality state.

Metric name	Best quality state	Probability that path is stable
Relative Mobility	Stationary	67.19 %
Delay	Very Low	73.29 %
CPU & Buffer usage	Low	52.89 %
Hop Count	Five-Eight	48.31 %
Data Rate	Very High	64.19 %
Loss ratio	Low	50.91 %
ETX	At once	50.91 %
FTE	Perfect	50.74 %
SNR	Excellent	46.71 %
Node Density	Sparse	52.13 %

On the other hand the BN can also be used to find the most probable status of every variable in a path that is witnessed to have stable performance. In Fig. 40 the

BN gives the probability distributions when there is the evidence that the path has stable performance. In that case, as expected, all metrics are becoming more probable to be in a higher quality state. In Fig. 40 the states with the higher probability are the most common states in a path that has a stable performance. This example has no practical value but it is presented here in order to demonstrate how BN can be useful for improving our knowledge about the MANETs.

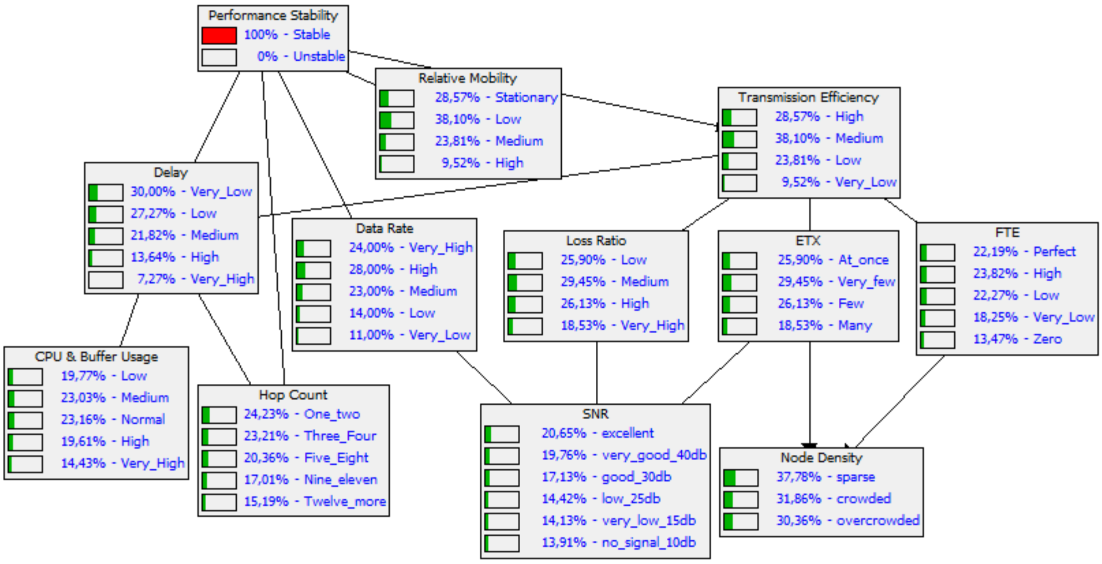


Fig. 40. Probability distributions of the states in every variable of the BN when evidence of a stable path is presented.

The number of the scenarios that can be examined can be as many as the combinations of each state in each variable node. Our intention here is just to present the complexity of the system than examining each different scenario.

4.8.2 Protocol Performance

To illustrate the effectiveness of the proposed routing algorithm, the results of BNR application on MANETs in a simulated environment are presented in this part. The performance of the BNR is presented in comparison to another established routing algorithm, the DSR.

The simulation environment consists of 100 mobile nodes with different flows of constant and variable bit rates between 20 of them. DSR is used as the routing algorithm between these nodes and throughput and delay during the 100 seconds of simulation are recorded for one of the flows. The simulation is repeated by using BNR as routing protocol between the same two nodes. The results can be found in Fig.41 and Fig.42 for throughput and delay respectively. It is clear that BNR can achieve better throughput and shorter delay than DSR during the simulation.

Next, end to end delay and throughput are presented for different mobility and different number of nodes in the network. The results are presented in Fig.43 and Fig.44. BNR can achieve higher throughput and shorter delay than DSR due to higher quality links in the paths in different mobility and node density conditions. Also from the figure it can be seen that BNR is more resilient to mobility as the rate of throughput decrease is smaller as the mobility increases when BNR is used. Results for delay are similar to the throughput results. Again BNR looks more resilient to mobility than DSR. Also there are not significant BNR performance differences between different numbers of nodes scenarios in the network.

Routing overhead is defined as the sum of all the routing control packets like RREQ, RREP, RRER etc. In Fig.45 the number of control packets corresponds to number of packets in the 100 sec simulation duration. BNR uses the same signaling as DSR, but from the results it can be observed that routing overhead is significantly reduced. These results are due to the fewer route discovery attempts when BNR is used because of the selection of paths that last longer.

Finally the packet loss ratio is examined in Fig.46. Except the quality of the selected path, packet loss ratio reflects the amount of disruption in the communication, as a result of route breakages.

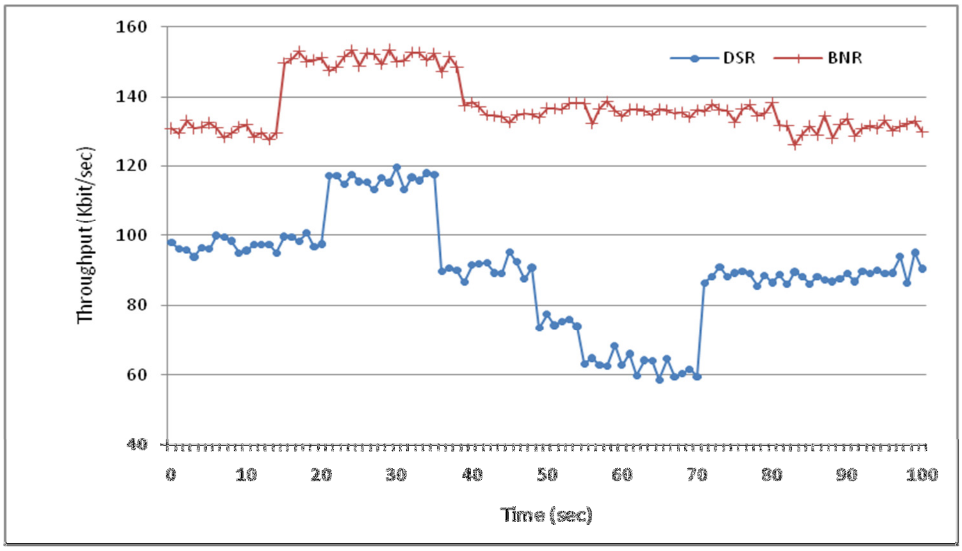


Fig.41. Throughput fluctuation between two nodes for DSR and BNR.

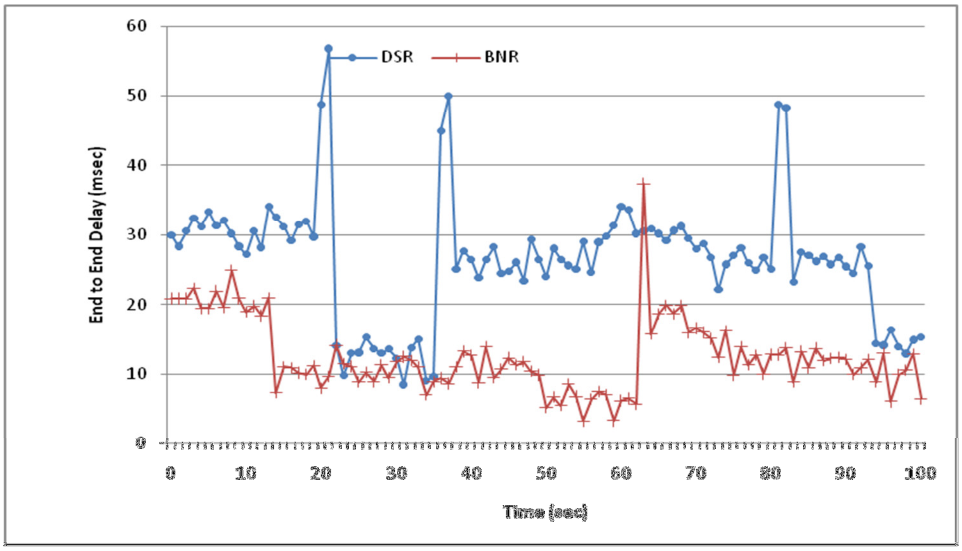


Fig.42. Delay fluctuation between two nodes for DSR and BNR.

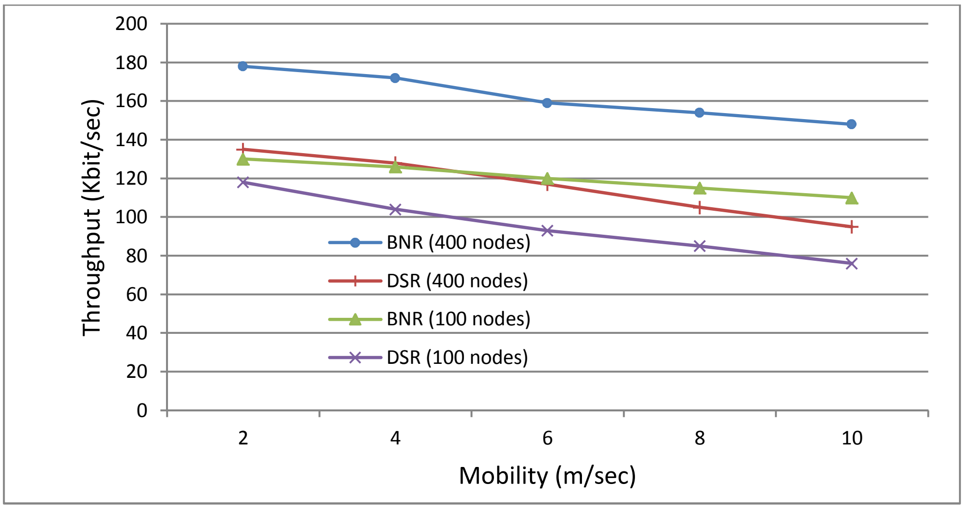


Fig.43. Throughput versus Mobility for different number of nodes for DSR and BNR.

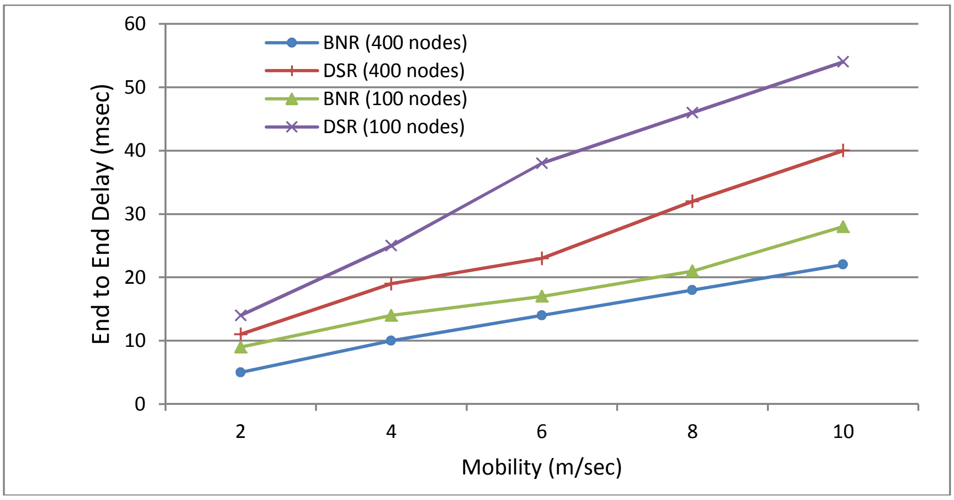


Fig.44. End to end delay versus Mobility for different number of nodes for DSR and BNR.

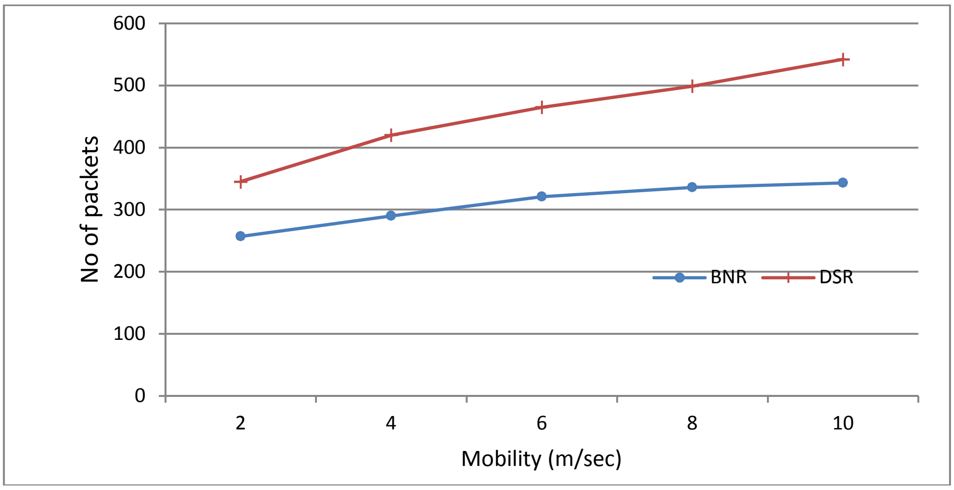


Fig.45. Routing overhead of DSR and BNR versus mobility.

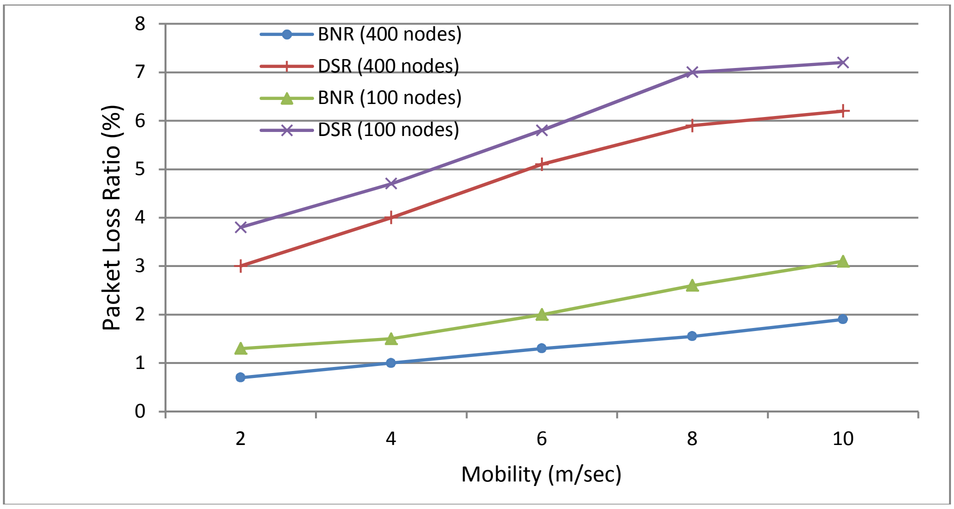


Fig.46. Packet loss ratio versus Mobility for different number of nodes for DSR and BNR.

4.8.3 Protocol Performance when metrics are missing

One of the advantages of a BN as the routing decision algorithm is the flexibility it has with missing information. In this section the performance of the routing protocol is investigated when various metrics are missing.

The first scenario that is presented assumes that the only available metric is the number of hopes. Instead of using the path with the fewer hops, BNR will choose the path that is more probable to have stable performance according to the amount of hops. In Table XI and Table XII the throughput and delay average performance is presented for DSR, BNR with hop count metric only and BNR with all metrics available. BNR can achieve better throughput and delay than DSR even when the only available metric information is the hop count.

Table XI. Average Throughput comparison between DSR, BNR with hop count metric only and BNR with all metrics available.

Average Throughput (Kbit/Sec)		
DSR	BNR (hop count)	BNR (full)
90.56	113.45	138.06

Table XII. Average Delay comparison between DSR, BNR with hop count metric only and BNR with all metrics available.

Average Delay (mSec)		
DSR	BNR (hop count)	BNR (full)
26.61	19.57	12.5

The delay metric can be measured easily but sometimes it is not possible to have a valid value quickly enough. When a delay measurement is not considered valid, it can be omitted in the BN. By observing the BN we can see that transmission efficiency, CPU & Buffer usage and hop count are related to the delay value. Hence, when delay is not available, it can be estimated by the other three values. In Table XIII the results for this case can be found.

In some cases the cross layer information can be inaccurate. For example physical layer can return SNR measurements without using enough samples. Similar mobility measurements that are based in a GPS technology can be acquired when GPS signal is low.

Thus, we investigate here the protocol performance when wrong cross layer information is supplied. As can be seen in Table XIII and

Table XIV, a wrong measurement of delay can produce worse results than a missing one.

Table XIII. Average Throughput comparison when BNR is operating with full information, without mobility, and full but inaccurate mobility.

Average Throughput (Kbit/Sec)		
BNR (full)	BNR (no mobility)	BNR (inaccurate mobility)
138.06	130.89	121.424

Table XIV. Average Delay comparison when BNR is operating with full information, without mobility, and full but inaccurate mobility.

Average Delay (msec)		
BNR (full)	BNR (no mobility)	BNR (inaccurate mobility)
12.5	14.5	19.76

4.9 Summary

Many routing metrics have been proposed by researchers. It is difficult to combine these metrics in order to provide logical conclusions about the success of the path. An efficient and adaptive method to process the information from these numerous metrics for a routing path and give inference about the eligibility of the path is introduced here. Improved routing metrics in the future promises dramatic improvement of BNs performance. The proposed algorithm targets to improve the QoS in a network by selecting the route which is most probable to be successful. It takes advantage of the plethora of the cross layer information that can be available. It is easy to introduce new metrics or remove existing ones in order to be used in different types of networks. The simulation experiments showed how can improve the performance in MANETs and the QoS that the applications will receive.

BN was designed by using statistical data from the simulation environment. More accurate conclusions can be drawn by using real world statistical data. In that case the BN must be re-designed by using the new statistical data. The same can happen in case that this method is desired to be used for different type of ad hoc networks like mesh networks.

A further improvement that is worth to be examined is the use of soft evidence in the BN. Soft evidence is not conclusive. For example we may get unreliable evidence that an event occurred, which may raise our belief in that event but not to the point where we would consider it certain. Thus, in the case of inaccurate cross layer information the performance of the protocol can be increased.

Considering the scalability of our proposed system, the protocol can suffer from the same scalability problems that DSR has. The focus of this work is to propose a decision algorithm solution and not to propose a mechanism for the rest of routing operations. However the BN based routing decision mechanism does not have any scalability issues and can be integrated with other routing protocols which improve the scalability of DSR like MDSR proposed in [Tamilarasi, M. (2007)].

The proposed routing decision algorithm can be modified for heterogeneous networks. In this case, more parameters in BN can be added. These parameters can be the technologies that are used in the links of a path assuming that different technologies provide different services and have different performance.

An interesting work for the future is how the integration in the protocol of a resource reservation mechanism can help to improve the performance. Also modifications in the protocol can take advantage of the possibility of multiple routes usage. The discovered paths can have similar probability of stable performance. Every time more than one path can be used that has high success estimation.

5 Further BN optimization and integration with GA

5.1 Introduction

In the previous chapter a Bayesian Network model for paths in MANETs was introduced and it has been shown how evidence is used to compute posterior probabilities for some hypotheses. The information for the evidence elements is cross layer information supplied mainly from lower layers. In this chapter further investigation is made regarding the use of BN to advise the other layers on how to configure some of their parameters, in order to improve the communication performance. The first part examines ways to support the transport layer and specifically the GA based, self modifying transport protocol proposed in Chapter 3. An extended version of the BN model that was introduced in Chapter 5 is used to make the cross layer suggestions in order to improve the overall performance. The target again is to create a soft relationship between the layers without any critical dependency on the cross layer information. The next part illustrates the overall performance in QoS on MANETs by using both protocols that are introduced in this thesis. In this part it is presented how each protocol adds up to the overall performance.

5.2 GA initial population optimization

Genetic Algorithm begins by creating a random initial population. As described in Chapter 3.5.3., the size of the population plays an important role on the GA's performance. There is a trade-off between computational cost and search capabilities which affect the quality of the solution. To illustrate better this phenomena, in Fig. 47

a search space example is shown along with ten random dots which represent ten random individuals in an initial population.

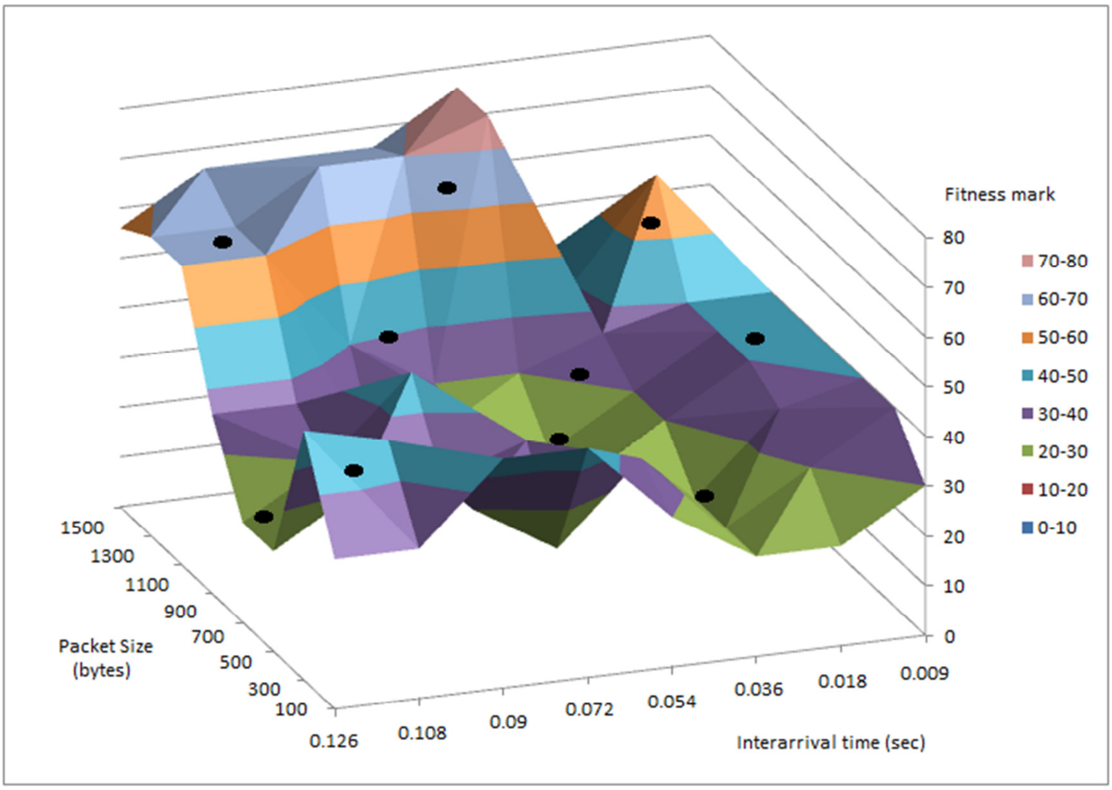


Fig. 47. Search space example for the ESMP configuration, with ten random individuals (black dots).

In Fig. 47, the initial population contains ten individuals, which is the value of population size that according to Chapter 3.5.3 is the minimum amount for the specific problem that satisfies the criteria that every point in the search space should be reachable from the initial population by crossover only. Note that all the individuals in the initial population are spread in the whole search space. The GA algorithm will select the individuals that receive the higher fitness so the next population is expected to be gathered in a smaller portion of the search space. That portion of the search space will be in a higher fitness area.

It is clear that a bigger population can take up a bigger area in the search space. Thus, more individuals result in a quicker and easier search. Usually, bigger population means computational cost but for ESMP means longer duration of inadequate communication. Notice that several of the initial individuals are placed in low fitness areas of the search space. Those individuals are translated to ESMP configurations that give bad performance. The experimental results in Chapter 3.6 show that ESMP is suffering from these bad quality individuals at the start of the communication.

If the algorithm knows approximately where the maximum fitness areas are, it can place the initial population on those areas. In that case the start of the communication can be smoother.

5.2.1 BN based initial population selection

A modified version of the BN model that was introduced in Chapter 4 is used to give inference on which area of the search space the initial population should be deployed. The new BN model contains one additional node. The search space node has 16 states. Each state represents one area in the search space. The initial population will be created in a way that the state that receives the highest probability will have the most initial individuals. In other words, the probability distribution of search space node will be used for the initial placement of the population.

For example in Fig. 48 the same search space can be seen from above divided in 16 areas. Areas like (A,1), (A,2) and (A,3) give the best fitness results, hence it is preferable most of the initial individuals to be placed on those parts of the search space. The opposite happens in areas like (C,1), (C,3) and (D,1) where individuals that may appear in those areas will result in bad performance.

Unfortunately, the search space is unknown and that is the reason in the first place, that a search optimization algorithm like GA is used to explore it. BN can be used to give inference from statistical data, on which of the 16 areas are more probable to have high fitness. The several metrics that network layer collects for routing decision can be used also for search area selection. For example, when the delay metric is observed to be high, then the selection of bigger packets and reduced rate might be beneficial. According to the statistical data the BN will propose the initial configuration and GA will optimize and change it whenever it needs to adapt in network changes. As it can be seen in Fig. 49 the “initial population” node is directly related with the same metrics which also directly affect the performance of a path.

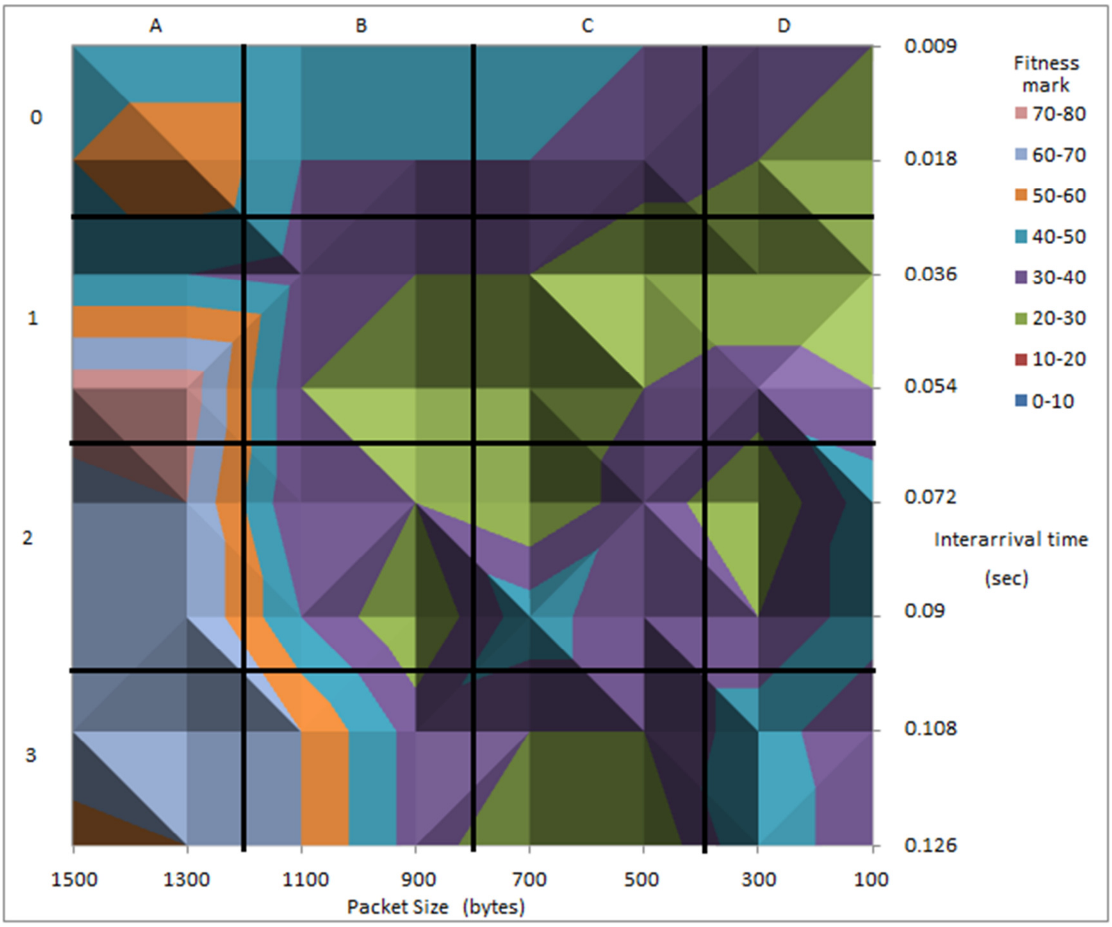


Fig. 48. Search area divided by 16 smaller areas which are represented

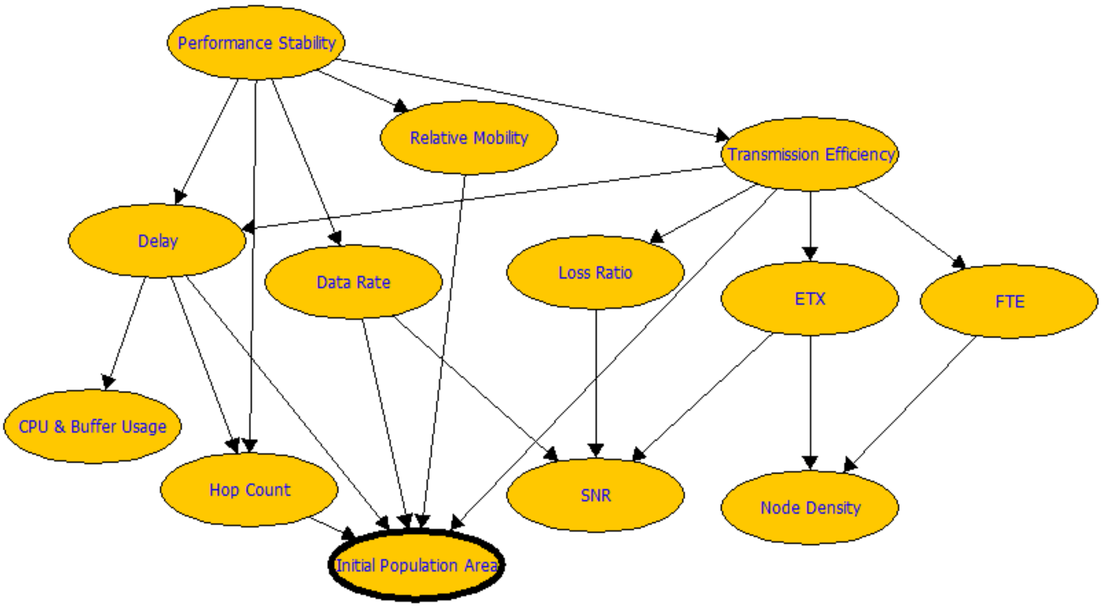


Fig. 49. Bayesian Network for initial individuals' placement selection.

5.2.2 Experimental results

The use of the BN for the creation of the initial population in ESMP results in a smoother performance at the start of the protocol. In Fig. 50 an example of the delay performance of the ESMP is shown, when random population and when BN suggested population is used. It is clear that the first 10 individuals, of the first population, have much better performance.

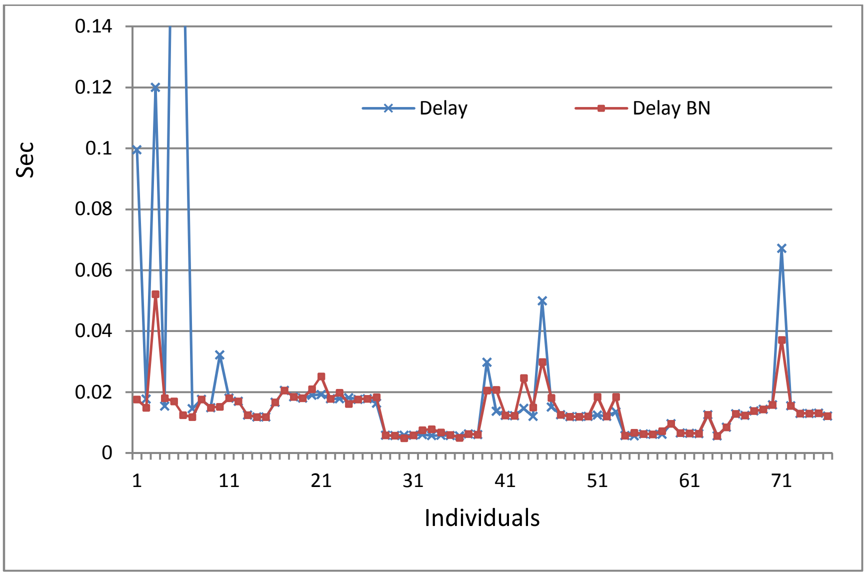


Fig. 50. Comparison in the delay performance of ESMP with and without BN's initial population suggestion.

5.3 Overall performance

As it was mentioned in the previous chapters, each of the protocol proposals targets to an explicit area of improvement in the MANETs communications. The BNR is responsible to decide which one of the available paths is more stable to use for the communication by using the best available cross layer information. In addition, ESMP utilizes the path and is trying to improve the performance further, by discovering the best flow control configuration for the path selected and adapts to any changes that happens in the path conditions during communication. At next, it becomes clear that the protocols complete each other and the performance increase is additive as the higher protocol builds on the achievements of the lower protocol.

5.3.1 Experimental Results

In this section we investigate the performance improvement of the overall performance when ESMP is operating at the same time that BNR is making the routing decisions. The experimental results that show the performance improvements of BNR are already presented in Chapter 4.8, so based on these results the integration with ESMP will be evaluated.

Table XV. Average Throughput with and without ESMP over BNR

Average Throughput (Kbit/Sec)		
DSR	BNR	with ESMP
90.56	138.06	145.86

Table XVI. Average Delay with and without ESMP over BNR

Average Delay (mSec)		
DSR	BNR	with ESMP
26.61	12.5	9.8

In Table XV and Table XVI throughput increase and delay decrease with ESMP usage are presented. Both tables are showing the average values. The performance increase is not so significant. Someone can expect greater performance increase because of the additive improvement that was already mentioned. This happens because ESMP performance is increased when the network conditions are critical. As it was proved in 3.6.2, the benefit of using ESMP is greater when the environment is stricter and the good quality individuals are spotted more easily. In other words, when the conditions in the network are not critical, any individual-configuration will perform very well. Hence, the performance improvement of ESMP is varying and cannot be illustrated well enough, by the average performance results presented in Table XV and Table XVI.

5.4 Summary

In this chapter we utilized the cross layer design architecture so the Bayesian Network can give advice to the other layers on how to achieve better performance. Network layer is chosen to host the BN for a good reason. BN already collects information for routing decision purposes, thus, it is easier to be extended a little and produce inference for additional issues.

BN is used to guess which part of the search space has the best configuration in order to give a “fast” start to the GA. BN is not meant to replace GA but only help it with where to start looking in the search space. GA will finally find the optimum solution and more importantly will adapt every time the network experiences any changes.

Finally, the overall performance benefits by using both BNR and ESMP, is examined. BNR finds the most performance stable path and ESMP finds the best configuration for the specific path and further increases the overall performance of the communication.

6 Conclusions and Future Work

In this part of the thesis, all the research efforts presented in the previous chapters are summarized and concluded along with their overall achievement to the aim and objective of the research. After that, all possible modifications, which could improve the performance of the presented research methodologies, are discussed in detail as future work.

6.1 Conclusions

This thesis has focused on the definition, design and evaluation of QoS aware solutions in the dynamic environment of MANETs. Two main novel solutions have introduced; one in the network layer as a routing decision algorithm and one in transport layer as an adaptive flow control.

6.1.1 Design Objectives

The flow control mechanism in the transport layer targets to improve the QoS experience in the network by adjusting dynamically the size and the rate of the packets send in the network. Initially, it was shown that different size of packets can experience different performance. At the same time, different size and rate of packets can change in a different way the conditions in the network. The challenge for the flow control mechanism is to find the perfect combination of size and rate to optimize the performance. The relation between the conditions in the network and the causes is extremely complicated. So, the perfect combination is hard to be found without the use of a heuristic method. The Genetic Algorithm is used generally due to its integrated ability to adapt in dynamic conditions. In ESMP, the GA constantly adapts the configuration to the new environment conditions in the MANET faster than the rate of changes that occur in the environment.

In the network layer, a novel routing decision algorithm is introduced. The main advantage of this algorithm is the flexibility to combine a variable amount of cross layer information. The main advantages of this design are two. Firstly, it utilizes the multiple routing metrics in contrary to other existing algorithms that are designed to utilize a limited amount of them, resulting on more accurate decisions. Secondly, it reduces the dependence on cross layer information.

Summarizing, the main objectives during the design of the protocols were the following:

- Cross layer minimum dependence. Cross layer design seems to be essential for the improvement of the network performance in MANETs. However, there are significant obstacles that need to be faced in cross layer design like stability, maintenance and robustness issues. A balanced use of cross layer information in system architecture can help to overcome these obstacles.
- Adaptive behaviour of the protocols. In a continuously changing environment the ability to adapt to any changes in order to maintain the communication performance, it is a compulsory characteristic.
- Autonomy oriented behaviour. The network characteristics in MANETs create a dynamic and complicated environment, where manual control is time consuming, expensive and error-prone. It is clear that autonomic characteristics like self-configuration and self-optimization needs to be adopted from protocols that operate in MANETs.
- QoS performance improvement for delay sensitive applications. The higher and most important objective was the improvement of the QoS performance on application demand. According to the application requirements, the protocols aim to increase the targeted QoS metric performance. For example, when a delay sensitive application requests a communication session, then the protocols try to improve the delay performance.

The protocols presented in this thesis achieved to either reduce the dependence from the cross layer information to minimum, in case of the route decision algorithm, or in case of the flow control mechanism, it was completely avoided to use cross layer design. Furthermore, autonomic behaviour achieved by using closed control loop in the flow control mechanism and learning capabilities from statistical data in the BN based routing decision algorithm.

6.1.2 Conclusions derived from experimental results

Generally, the experimental results confirm that the objectives have been met. The GA-based flow control algorithm succeeded to find the best configuration fast enough, most of the times during the second generation. The delay performance has been improved in comparison to the performance when a fixed configuration has been used.

Initially, different selection methods and mutation rates have been tried in order to choose the most appropriate to use for the genetic algorithm while it is operating in a MANET environment. Furthermore, the behaviour of the algorithm is investigated in different application requirements. Demanding and non demanding requirements have been used to investigate how the algorithm is able to improve a specific QoS metric, once this metric is of high importance for the application that uses the network. The algorithm succeeded on these tests and was able to adapt on both the requirements and network conditions. Throughput and especially delay were the two QoS metrics that had the best performance results from the proposed flow control algorithm. Jitter of the delay had a small improvement that was not so clear at first sight but appeared, in the statistical analysis that was presented, that most of the times were slightly better when GA was used. The major side effect of the GA use has appeared in every simulation result and was the random performance of the initial random population. Later in Chapter 5 a solution is presented with the help of the Bayesian Inference using a modified BN model.

The routing decision algorithm BNR that is presented in Chapter 4 was tested and compared with DSR algorithm. Throughput and delay results are presented for different mobility and network size scenarios. BNR performed better than DSR in all the occasions. In addition, BNR have shown to be more resilient to mobility increase than DSR. The routing overhead was also reduced when using BNR. Moreover, the ability of the Bayesian Network to offer good quality decisions when information is missing was presented. A characteristic example is the occasion where the only routing metric available was the number of hops. In that case BNR managed to perform better than DSR which always selects the less hop path.

Additionally, the BN model has been evaluated by making queries and observing the answers. The queries results were compared with the expected behaviour that has been investigated in previous researches. It was shown also, how new conclusions about path performance in MANETs can emerge by experimenting with this model.

Finally, the integration of BNR and ESMP is presented. The two protocols complement each other and offer their services by utilizing different opportunities to increase performance. At the same time, a cross layer design between the two proposed protocols is used to improve the quality of the initial population in the GA used in ESMP. Instead of a symmetric random creation of the initial population, an asymmetric random creation is used. The asymmetric random population is using a distribution that is given from a slightly modified BN model in BNR. The results are showing that the behaviour of the initial population is significantly improved.

6.2 Future Work

The purpose of this thesis was to improve the QoS in MANETs by proposing two novel mechanisms for two different protocol layers. During the attempt to design the proposed protocols, several thoughts derived about the possible future work that can be done in order to improve further the QoS in MANETs.

The presented ESMP protocol was designed to just give evidence that different configurations of packet size and rate can produce different QoS and to prove that GA is fast enough to search for the optimum configuration. Further work should be performed in implementing the complete transport protocol that satisfies the requirements of a commercial protocol. Well defined service access points should be created and tests should be performed to check the stability of the implementation in different operation conditions.

The case of additional parameters, like battery consumption, can be investigated. If the network is in good condition and the application is not very demanding, then other secondary issues, like battery consumption, can be taken into consideration from the protocol. For example, the battery consumption metric can be added in the protocol by measuring the energy consumption for the time period of one individual. A new parameter that represents the energy consumption should be added in the fitness function with a smaller weight (priority) than the other parameters so it cannot have significant effect to the selection of the individuals and practically affect the fitness evaluation only when the other parameters are satisfied.

The proposed BN statistical model describes a path in MANETs. This model seems to be promising, since it has been used successfully for routing decisions, as well as, to suggest configuration solutions to the other layers. Consequently, as future work, this sophisticated model can be used to confirm several more existing assumptions noted in literature and to discover new characteristics for MANETs. In addition, a statistical analysis in more depth can be done by including more routing metrics and other characteristics in order to improve the accuracy of the model.

BN was designed by using statistical data from the simulation environment. For more accurate results in a real environment, the BN must be re-designed by using real world statistical data. The same can be happened in case that this method is desired to be used for different type of ad hoc networks like mesh networks.

An important issue in the routing protocols is the scalability. BNR can suffer from the same scalability problems that DSR has. The focus of this work is to propose a decision algorithm solution and not to propose a mechanism for the rest of routing operations. However, the BN based routing decision mechanism does not have any scalability issues and can be integrated with other routing protocols which improve the scalability of DSR like MDSR proposed in [Tamilarasi, M. (2007)].

In chapter 5 a cross layer design between BNR and ESMP was presented in order to improve the GA performance by utilizing further the capability of BN to give inference. Following a similar approach can be used where BN inference assists the lower layers by supplying them with valuable predictions. By invoking sensitivity analysis to the BN model, can be found which variables will increase the probability for a path to be stable in case those variables increase slightly their performance. For example, let’s assume that after sensitivity analysis is performed, is found that if transmission efficiency is improved the probability the path to be stable is increased considerably. In that case the system will decide that the increase of transmission efficiency worth the effort. Physical and MAC layer can utilize the results of the sensitivity analysis by changing their transmission power, the forward error correction algorithm, modulation technique etc.

References

Ahn, C. W., & Ramakrishna, R. (2002). A genetic algorithm for shortest path routing problem and the sizing of populations. *Evolutionary Computation, IEEE Transactions on , 6* (6), 566-579.

Akyildiz, I., & Wang, X. (2009). *Wireless mesh networks.* Chichester, U.K: Wiley.

Awerbuch, B., Holmer, D., Rubens, H., & Ad, M. r. (2003). High Throughput Route Selection in Multi-Rate Ad Hoc Wireless Networks. *High Throughput Route Selection in Multi-Rate Ad Hoc Wireless Networks* .

Barolli, L., Koyama, A., & Shiratori, N. (2003). A QoS routing method for ad-hoc networks based on genetic algorithm., (pp. 175-179).

Cen, S., Cosman, P., & Voelker, G. (2003). End-to-end differentiation of congestion and wireless losses. *Networking, IEEE/ACM Transactions on , 11* (5), 703-717.

Chandran, K., Raghunathan, S., Venkatesan, S., & Prakash, R. (2001). A feedback-based scheme for improving TCP performance in ad hoc wireless networks. *Personal Communications, IEEE , 8* (1), 34-39.

Chou, H., Premkumar, G., & Chu, C.-H. (2001). Genetic algorithms for communications network design - an empirical study of the factors that influence performance. *Evolutionary Computation, IEEE Transactions on , 5* (3), 236-249.

Choudhury, R. R., Yang, X., Ramanathan, R., & Vaidya, N. H. (2002). Using directional antennas for medium access control in ad hoc networks. (pp. 59-70). ACM.

Clausen, T., & Jacquet, P. (2003). Optimized Link State Routing Protocol (OLSR). *Optimized Link State Routing Protocol (OLSR)* . RFC Editor.

Dengiz, B., Altiparmak, F., & Smith, A. (1997). Local search genetic algorithm for optimal design of reliable networks. *Evolutionary Computation, IEEE Transactions on , 1* (3), 179-188.

Dube, R., Rais, C., Wang, K.-Y., & Tripathi, S. (1997). Signal stability-based adaptive routing (SSA) for ad hoc mobile networks. *Personal Communications, IEEE , 4* (1), 36-45.

Dyer, T. D., & Boppana, R. V. (2001). A comparison of TCP performance over three routing protocols for mobile ad hoc networks. (pp. 56-66). ACM.

Elaarag, H. (2002). Improving TCP performance over mobile networks. *ACM Comput. Surv. , 34* (3), 357-374.

ElRakabawy, S. M., Klemm, A., & Lindemann, C. (2005). TCP with adaptive pacing for multihop wireless networks. (pp. 288-299). ACM.

Firoiu, V., Le, J. y., Towsley, D., & Zhang, Z.-L. (2002). Theories and Models for Internet Quality of Service. *Theories and Models for Internet Quality of Service , 90* , 1565-1591.

Fu, P., & Zhang, D. (2005). A Heuristic and Distributed QoS Route Discovery Method for Mobile Ad Hoc Networks. In W. Fan, Z. Wu, & J. Yang (Eds.), *Advances in Web-Age Information Management* (Vol. 3739, pp. 428-439). Springer Berlin / Heidelberg.

Fu, Z., Meng, X., & Lu, S. (2003). A transport protocol for supporting multimedia streaming in mobile ad hoc networks. *Selected Areas in Communications, IEEE Journal on , 21* (10), 1615-1626.

Goldberg, David E (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Kluwer Academic Publishers, Boston, MA.

Guo, Z., & Malakooti, B. (2007). Energy Aware Proactive MANET Routing with Prediction on Energy Consumption., (pp. 287-293).

Gupta, R., Jia, Z., Tung, T., & Walrand, J. (2005). Interference-aware QoS routing (IQRouting) for ad-hoc networks., *5*, pp. 6 pp. -2604.

Hanbali, A. A., & Altman, E. (2005). A Survey of TCP over Ad Hoc Networks. *IEEE Communications Surveys \& Tutorials , 7*, 22-36.

Hiertz, G., Denteneer, D., Max, S., Taori, R., Cardona, J., Berlemann, L., et al. (2010). IEEE 802.11s: The WLAN Mesh Standard. *Wireless Communications, IEEE , 17* (1), 104-111.

Holland, G., & Vaidya, N. (1999). Analysis of TCP Performance over Mobile Ad Hoc Networks Part I: Problem Discussion and Analysis of Results. *Analysis of TCP Performance over Mobile Ad Hoc Networks Part I: Problem Discussion and Analysis of Results* , 219-230. ACM.

IEEE, C. S. (2002). Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs). *Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)* .

IEEE, C. S. (2009). Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)* .

IEEE, C. S. (1997). Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications* .

J., D. S., Aguayo, D., Bicket, J., & Morris, R. (2003). A high-throughput path metric for multi-hop wireless routing. (pp. 134-146). ACM.

Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. (pp. 153-181). Kluwer Academic Publishers.

Kawadia, V., Kumar, P.R. , A cautionary perspective on cross-layer design, Wireless Communications, IEEE , vol.12, no.1, pp. 3- 11, Feb. 2005

Keyes, R.W.; , Moore's Law today, Circuits and Systems Magazine, IEEE , vol.8, no.2, pp.53-54, Second Quarter 2008

Kim, D., Bae, H., Song, J., & Cano, J.-C. (2005). Analysis of the interaction between TCP variants and routing protocols in MANETs., (pp. 380-386).

Krishnamurthy, L. (2004). Making radios more like human ears: alternative MAC techniques and innovative platforms to enable large-scale meshes. *Microsoft Mesh Networking Summit* .

Lee, S., Bhattacharjee, B., & Banerjee, S. (2005). Efficient geographic routing in multihop wireless networks. (pp. 230-241). ACM.

Mangold, S., Choi, S., Hiertz, G., Klein, O., & Walke, B. (2003). Analysis of IEEE 802.11e for QoS support in wireless LANs. *Wireless Communications, IEEE , 10* (6), 40-50.

Mbarushimana, C., Shahrabi, A., & Buggy, T. (2008). A cross-layer support for TCP enhancement in QoS-aware mobile ad hoc networks. (pp. 10-17). ACM.

Nikaein, N., Bonnet, C., & Nikaein, N. (2001). HARP - HYBRID AD HOC ROUTING PROTOCOL. *Institut Eurecom Symp. Telecommun* .

Perkins, C. E., & Bhagwat, P. (1994). Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. *SIGCOMM Comput. Commun. Rev. , 24*, 234-244.

Perkins, C., & Royer, E. (1999). Ad-hoc on-demand distance vector routing., (pp. 90-100).

Rao, A., Ratnasamy, S., Papadimitriou, C., Shenker, S., & Stoica, I. (2003). Geographic routing without location information. (pp. 96-108). ACM.

Reeves, C. R. (Ed.). (1993). *Modern heuristic techniques for combinatorial problems.* John Wiley & Sons, Inc.

Rubin, I., & Liu, Y.-C. (2003). Link stability models for QoS ad hoc routing algorithms., *5*, pp. 3084 - 3088 Vol.5.

Seada, K., Zuniga, M., Helmy, A., & Krishnamachari, B. (2004). Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks., (pp. 108-121).

Shen, H., Shi, B., zou, L., & Gong, H. (2003). A distributed entropy-based long-life QoS routing algorithm in ad hoc network., *3*, pp. 1535 - 1538 vol.3.

Sheng, M., Li, J., & Shi, Y. (2003). Routing protocol with QoS guarantees for ad-hoc network. *Electronics Letters , 39* (1), 143-145.

Sivakumar, R., Sinha, P., & Bharghavan, V. (1999). CEDAR: a core-extraction distributed ad hoc routing algorithm. *Selected Areas in Communications, IEEE Journal on , 17* (8), 1454-1465.

Stojmenovic, I. (2002). Position-based routing in ad hoc networks. *Communications Magazine, IEEE , 40* (7), 128-134.

Sundaresan, K., Anantharaman, V., Hsieh, H.-Y., & Sivakumar, A. (2005). ATP: a reliable transport protocol for ad hoc networks. *Mobile Computing, IEEE Transactions on , 4* (6), 588-603.

Tamilarasi, M., Sunder, V. S., Haputhanthri, U., Somathilaka, C., Babu, N., Chandramathi, S., et al. (2007). Scalability Improved DSR Protocol for MANETs., *4*, pp. 283-287.

Toh, C.-K. (1996). *Wireless ATM and Ad-Hoc Networks: Protocols and Architectures.* Kluwer Academic Publishers.

Trivino-Cabrera, A., de-la, J. G., & Casilari, E. (2010). Identification of Stable Links in MANETs., (pp. 276-281).

Tseng, Y.-C., Li, Y.-F., & Chang, Y.-C. (2003). On route lifetime in multihop mobile ad hoc networks. *Mobile Computing, IEEE Transactions on , 2* (4), 366-376.

Tufte, G., & Haddow, P. C. (1999). Prototyping a GA Pipeline for Complete Hardware Evolution., (pp. 76–84).

Wang, X., Rollinger, V., Patil, A., Chao, G., Wang, W., & Nova, M. (2006).

Wang, X., Wang, W., & Nova, M. P. (2005).

Wireless, U. P. (2010). *Wireless Universal Serial Bus Specification 1.1.* LSI Corporation, Hewlett-Packard Company, Intel Corporation, Microsoft Corporation, NEC Corporation, ST-Ericsson, Samsung Electronics Co., Ltd.

Woo, M., Singh, S., & Raghavendra, C. S. (1998). Power-Aware Routing in Mobile Ad Hoc Networks. *Power-Aware Routing in Mobile Ad Hoc Networks* .

Yang, Y., & Kravets, R. (2005a). Contention-aware admission control for ad hoc networks. *Mobile Computing, IEEE Transactions on , 4* (4), 363-377.

Yang, Y., Wang, J., & Kravets, R. (2005b). Designing routing metrics for mesh networks.

Yarvis, M. D., Conner, W. S., Krishnamurthy, L., Mainwaring, A., Chhabra, J., & Elliott, B. (2002). Real-World Experiences with an Interactive Ad Hoc Sensor Network.

Yu, M., Malvankar, A., Su, W., & Foo, S. Y. (2007). A link availability-based QoS-aware routing protocol for mobile ad hoc sensor networks. *Computer Communications , 30* (18), 3823-3831.

Zhu, H., Li, M., Chlamtac, I., & Prabhakaran, B. (2004). A survey of quality of service in IEEE 802.11 networks. *Wireless Communications, IEEE , 11* (4), 6-14.

APPENDIX A’ (TCL script sample for ESMP simulation)

```
if {$argc !=3} {
      set scriptname [info script]
        puts "Usage: ns $scriptname  Routing_Protocol Traffic_Pattern
Scene_Pattern "
        puts "Example:ns $scriptname DSDV cbr-50-10-8 scene-50-0-20"
        exit
}
#ns esmp-mesh.tcl DSR cbr-50-10-2 scene-50-0-20-100-500-500

set par1 [lindex $argv 0]
set par2 [lindex $argv 1]
set par3 [lindex $argv 2]

set val(chan)           Channel/WirelessChannel    ;# channel type
set val(prop)           Propagation/TwoRayGround   ;# radio-
propagationmodel
set val(netif)          Phy/WirelessPhy            ;# network
interface type
set val(mac)            Mac/802_11                 ;# MAC type

if { $par1=="DSR"} {
  set val(ifq)           CMUPriQueue
} else {
  set val(ifq)          Queue/DropTail/PriQueue    ;# interface queue
type
}

set val(ll)             LL                         ;# link layer type
set val(ant)            Antenna/OmniAntenna        ;# antenna model
set val(ifqlen)         50                         ;# max packet in
ifq
set val(rp)             $par1                      ;# routing
protocol
set val(x)              500
set val(y)              500
set val(seed)           0.0
set val(tr)             $val(rp).tr
set val(namtr)          $val(rp).nam
set val(nn)             50
set val(cp)             $par2
set val(sc)             $par3
set val(stop)           100.0

set val(rxPower)        0.00175                         ;#Potencia
recepción en W
set val(txPower)        0.00175                       ;#Potencia
transmisión en W
set val(energymodel)    EnergyModel             ;
set val(initialenergy)  1000                       ;# Initial energy
in Joules
set val(sleeppower)     0.00005                      ;#sleep power W
15 uA P=I*I*R=11.25*10E-9 W
set val(tp)             0.002                   ;#transition powerpower
consumption (Watt) in state transition from sleep to idle (active)
set val(tt)             0.005                 ;#transition time(second)
use instate transition from sleep to idle (active)
set val(ip)             1.0                        ;#idle power 8 mA
en modo activo.P=0.0032


set ns_ [new Simulator]
$ns_ color 1 Red

set tracefd [open $val(tr) w]
$ns_ trace-all $tracefd
$ns_ use-newtrace

# *** Initialize Network Animator ***
set namtrace [open $val(namtr) w]
$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

#Define a 'finish' procedure
proc finish {} {
    puts "Running Finish"
    global ns_ namtrace val
    $ns_ flush-trace
    #Close the NAM trace file
    close $namtrace
    #Execute NAM on the trace file
    #exec nam $val(namtr) &
    after 300
    #exec xgraph gaout/throughput.txt &
    #exec xgraph gaout/delay.txt &
    #exec xgraph gaout/jitter.txt &
    #exec xgraph gaout/loss.txt &
    exit 0
}

set topo [new Topography]
$topo load_flatgrid $val(x) $val(y)


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

set chan_1_ [new $val(chan)]

$ns_ node-config -adhocRouting $val(rp) \
               -llType $val(ll) \
               -macType $val(mac) \
               -ifqType $val(ifq) \
               -ifqLen $val(ifqlen) \
               -antType $val(ant) \
               -propType $val(prop) \
               -phyType $val(netif) \
               -channel $chan_1_ \
               -topoInstance $topo \
      #-energyModel $val(energymodel) \
      #-idlePower $val(ip) \
      #-rxPower $val(rxPower) \
      #-txPower $val(txPower) \
      #-sleepPower $val(sleeppower)\
      #-transitionPower $val(tp) \
      #-transitionTime $val(tt)\
      #-initialEnergy $val(initialenergy)\
-nodecolor black \
               -agentTrace ON \
               -routerTrace ON \
               -macTrace OFF


for {set i 0} {$i < $val(nn) } {incr i} {
      set node_($i) [$ns_ node]
      $node_($i) random-motion 0            ;# disable random motion
}

#$ns_ node-config -reset
#set node_(49) [$ns_ node]
#$node_(49) random-motion 0

#Setup an ESMP connection
set esmp [new Agent/ESMP]
$ns_ attach-agent $node_(49) $esmp
set esmpsink [new Agent/ESMP]
$ns_ attach-agent $node_(19) $esmpsink
$ns_ connect $esmp $esmpsink
$esmp set fid_ 1

$esmp set disable_ga_flag true
$esmp set p_mut 0.01
$esmp set p_cross 0.9
$esmp set selection_method 1  #1 to 6 for RankSelector
RouletteWheelSelector TournamentSelector DSSelector SRSSelector
UniformSelector

#Give planned values
$esmpsink set rtt_planned 0.02
$esmpsink set jitter_planned 0.005
$esmpsink set throughput_planned 0.15mb
$esmpsink set error_planned 0.1
#give weights for each QoS in the fitness calculation
$esmpsink set rtt_weight 2
$esmpsink set jitter_weight 2
$esmpsink set throughput_weight 2
$esmpsink set error_weight 2
$ns_ at 1.4 "$esmp start"
$ns_ at $val(stop) "$esmp stop"

puts "Loading connection pattern..."
```

```
source $val(cp)

puts "Loading scenario file..."
source $val(sc)

for {set i 0} {$i < $val(nn) } {incr i} {
      $ns_ initial_node_pos $node_($i) 20
}

for {set i 0} {$i < $val(nn) } {incr i} {
      $ns_ at $val(stop).000000001 "$node_($i) reset";
}

$ns_ at $val(stop).000000001 "puts \"NS EXITING...\"; $ns_ halt;
finish"

puts "Start Simulation..."

#Call the finish procedure
#$ns_ at $val(stop) "finish"

$ns_ run
```

APPENDIX B’ (R script for correlation matrix creation)

```
datafilename="routing_metrics.txt"
dataset =read.table(datafilename,header=TRUE)  #read the file with
the data
names(dataset)                          #take the metrics names?
round(cor(dataset),2)                   #calculate the correlation matrix
```