On Backoff Mechanisms for Wireless Mobile Ad Hoc Networks

A Thesis Submitted

By

Saher S. Manaseer

For The Degree of

Doctor of Philosophy

To

The Faculty of Information and Mathematical Sciences

University of Glasgow

Abstract

Since their emergence within the past decade, which has seen wireless networks being adapted to enable mobility, wireless networks have become increasingly popular in the world of computer research. A Mobile Ad hoc Network (MANET) is a collection of mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure. MANETs have received significant attention in recent years due to their easiness to setup and to their potential applications in many domains. Such networks can be useful in situations where there is not enough time or resource to configure a wired network. Ad hoc networks are also used in military operations where the units are randomly mobile and a central unit cannot be used for synchronization.

The shared media used by wireless networks, grant exclusive rights for a node to transmit a packet. Access to this media is controlled by the Media Access Control (MAC) protocol. The Backoff mechanism is a basic part of a MAC protocol. Since only one transmitting node uses the channel at any given time, the MAC protocol must suspend other nodes while the media is busy. In order to decide the length of node

suspension, a backoff mechanism is installed in the MAC protocol. The choice of backoff mechanism should consider generating backoff timers which allow adequate time for current transmissions to finish and, at the same time, avoid unneeded idle time that leads to redundant delay in the network. Moreover, the backoff mechanism used should decide the suitable action to be taken in case of repeated failures of a node to attain the media. Further, the mechanism decides the action needed after a successful transmission since this action affects the next time backoff is needed.

The Binary exponential Backoff (BEB) is the backoff mechanisms that MANETs have adopted from Ethernet. Similar to Ethernet, MANETs use a shared media. Therefore, the standard MAC protocol used for MANETs uses the standard BEB backoff algorithms. The first part of this work, presented as Chapter 3 of this thesis, studies the effects of changing the backoff behaviour upon a transmission failure or after a successful transmission. The investigation has revealed that using different behaviours directly affects both network throughput and average packet delay. This result indicates that BEB is not the optimal backoff mechanism for MANETs.

Up until this research started, no research activity has focused on studying the major parameters of MANETs. These parameters are the speed at which nodes travel inside the network area, the number of nodes in the network and the data size generated per second. These are referred to as mobility speed, network size and traffic load respectively. The investigation has reported that changes made to these parameters values have a major effect on network performance.

Existing research on backoff algorithms for MANETs mainly focuses on using external information, as opposed to information available from within the node, to decide the length of backoff timers. Such information includes network traffic load, transmission failures of other nodes and the total number of nodes in the network. In a mobile network, acquiring such information is not feasible at all times. To address this point, the second part of this thesis proposes new backoff algorithms to use with MANETs. These algorithms use internal information only to make their decisions. This part has revealed that it is possible to achieve higher network throughput and less average packet delay under different values of the parameters mentioned above without the use of any external information.

This work proposes two new backoff algorithms. The Optimistic Linear-Exponential Backoff, (OLEB), and the Pessimistic Linear-Exponential Backoff (PLEB). In OLEB, the exponential backoff is combined with linear increment behaviour in order to reduce redundant long backoff times, during which the media is available and the node is still on backoff status, by implementing less dramatic increments in the early backoff stages. PLEB is also a combination of exponential and linear increment behaviours. However, the order in which linear and exponential behaviours are used is the reverse of that in OLEB. The two algorithms have been compared with existing work. Results of this research report that PLEB achieves higher network throughput for large numbers of nodes (e.g. 50 nodes and over). Moreover, PLEB achieves higher network throughput with low mobility speed. As for average packet delay, PLEB significantly improves average packet delay for large network sizes especially when combined with high traffic rate and mobility speed. On the other hand, the measurements of network throughput have revealed that for small networks of 10 nodes, OLEB has higher throughput than existing work at high traffic rates. For a medium network size of 50 nodes, OLEB also achieves higher throughput. Finally, at a large network size of 100 nodes, OLEB reaches higher throughput at low mobility speed. Moreover, OLEB produces lower average packet delay than the existing algorithms at low mobility speed for a network size of 50 nodes.

Finally, this work has studied the effect of choosing the behaviour changing point between linear and exponential increments in OLEB and PLEB. Results have shown that increasing the number of times in which the linear increment is used increases network throughput. Moreover, using larger linear increments increase network throughput.

I dedicate this work to my most precious treasures in life, my family and my fiancée, Lama, for their love, support and encouragement.

Saher S. Manaseer

Acknowledgments

I would like to express my sincere appreciation to my supervisors: Dr. M. Ould-Khaoua and Dr. L. M. Mackenzie. Their continued support and guidance have always been my strongest motivation and the essential energy source behind this work.

I am highly grateful to my family for their support over the last four years, my dearest parents and brothers. I am also thankful to Eng. Lama Abu Alia for her tremendous support. My thanks to them will be everlastingly as it is her support that got me through the writing up of this thesis.

I would like to express my special appreciation to my friend and mentor, Dr. C Kyriakides, for his unlimited support and advice. I would also thank my greatest friend and the source of inspiration, Dr. M. Masada. His support has been valuable and continuous especially in the first period of my PhD course. I would love to extend my thanks to my friends and colleagues: Dr. B. Glisson, Dr. C. Unsworth, S. Oehler and Dr. Leif Azzopardi.

Last but not least, I would like to thank the University of Jordan for the financial support throughout the years of my PhD.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACK	Acknowledgement
AP	Access Point
BEB	Binary Exponential Backoff
CBR	Constant Bit Rate
CSMA/CA	Carrier sense Multiple Access with Collision Avoidance
CTS	Clear To Send
CW	contention window
DCF	Distributed Coordination Function
FIB	Fibonacci Backoff
DIFS	DCF Interframe Space
LAN	Local Area Network
LMILD	Linear Multiplicative Increase Linear Decrease
LOG	Logarithmic Backoff
MAC	Media Access Control
MANET	Mobile Ad hoc Networks
MILD	Multiplicative Increase Linear Decrease
MWP	Markovian Way Point
NAV	Network Allocation Vector
NBA	Neighbourhood Backoff Algorithm
NS	Network Simulator
OLEB	Optimistic Linear Exponential Backoff

PCF	Point Coordination Function
PDA	Personal Digital Assistant
PLEB	Pessimistic Linear Exponential Backoff
RTS	Request To Send
RWPB	Random Way Point with Borders
SBA	Sensing Backoff Algorithm
TCP	Transmission Control Protocol
WLAN	Wireless Local Area Network

Chapter 1. Introduction

1.1. Introduction

Since their emergence in the 1970s, wireless networks have become increasingly popular. This is particularly true within the past decade, which has seen wireless networks being adapted to enable mobility. There are currently two variations of mobile wireless networks [54], *infrastructure* and *ad hoc wireless* networks. Wireless networking increases availability and allows rapid deployment of wireless transceivers in a wide range of computing devices such as PDAs, laptops and desktop computers [24]. Wireless networks came as a result of the technological advances and extensions of LAN model as detailed in the IEEE 802.11 standard [37].

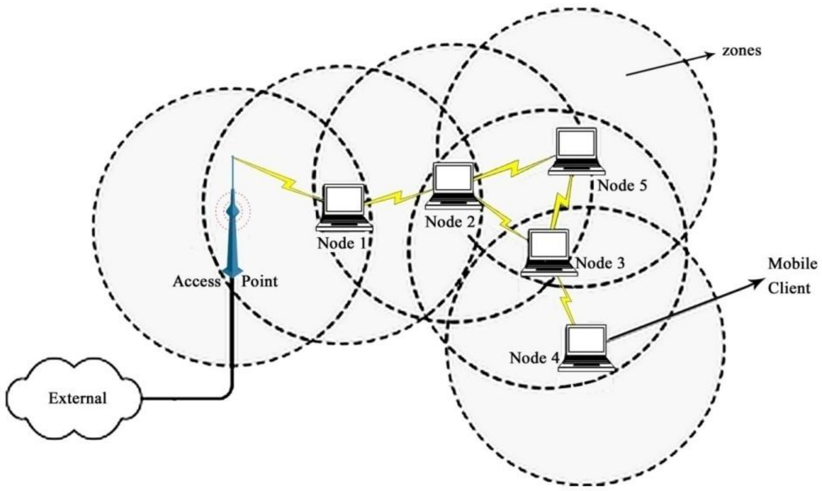


Figure 1.1: An example of an Infrastructured Wireless Network.

Figure 1.1 shows an example of the first type of wireless networks, Infrastructure Wireless Networks. Those networks with fixed and wired gateways have bridges known as base stations. This type of network is built on top of a wired network forming a reliable wireless network [37]. A mobile unit within these networks connects to and communicates with the nearest base station that is within its communication radius. Since each of the base stations has a transmission range, a node changes base stations when it moves out of the transmission range of one base station and enters the transmission range of another. The process of moving between base stations is referred to as hand-off [52]. Typical applications of this type of networks include Wireless Local Area Networks (WLANs) and Cellular Phone Networks [92].

The second type of wireless networks is Mobile Ad Hoc Networks (MANETs). In these networks, communication takes place without the need for base stations [16]. MANETs have received significant attention in recent years due to their potential applications in many domains. Such networks can be useful in disaster recovery where there is not enough time or resource to configure a wired network. Ad hoc networks are also used in military operations where the units are moving around the battlefield in a random way and a control unit cannot be used for synchronization [69]. A wireless ad hoc network is a collection of mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure [84]. In a MANET, nodes are not only senders and receivers where data and applications are located; each node in a MANET operates as a router to serve in delivering data to destinations [13].

To clarify the concept, ad hoc is defined to be a network connection method. This method is usually related to wireless devices [49]. The connection is established for the duration of one session that starts when a node joins the network and ends when the node leaves and requires no control units to organise the process. As an alternative, nodes discover other nodes within a transmission range to form a network. Connections are possible over multiple-node paths to form what is known as multihop ad hoc network [32]. It is the responsibility of routing protocols then to provide and maintain connections even if nodes are moving within the boundaries of the network area [71]. In other words, ad hoc networks are organised in an informal way, as the formal way being through designated control units [20]. Wireless ad-hoc networks are self-organizing, rapidly deployable, and require no fixed infrastructure. The wireless nodes must cooperate in order to establish communications dynamically using limited network management and administration. This is the reason why ad hoc protocols in general function in a distributed manner [28]. Nodes in an ad hoc network range from being highly mobile, to being stationary. They may also vary widely in terms of their capabilities and uses. The objective of ad hoc network architecture is to achieve increased flexibility, mobility and ease of management relative to normal or wireless networks with an *infrastructure*. [72]

It is unrealistic to expect a mobile ad hoc network to be fully connected, where a node can communicate directly with every other node in the network. Typically nodes must use a multihop path for transmission, and a packet may traverse multiple nodes before reaching its destination.

The rest of this chapter is organised as follows; Section 1.2 highlights the main features and characteristics of MANETs. Section 1.3 introduces the major challenges facing the application of MANETs. Section 1.4 lists some applications of MANETs. Section 1.5 then moves to explain the motivation behind conducting this research. Thesis statement is in section 1.6. Section 1.7 emphasises the contributions of this work. Finally, section 1.8 summarises the chapter and links it to the next chapter of this thesis.

1.2. Features and Characteristics of MANETs

MANETs have introduced new features in addition to the characteristics of Wireless networks and LANs. Due to the new type of nodes and topologies, MANETs have introduced many features.

1. **Distributed Functionality:** a MANET distributes the control and communication mechanisms amongst nodes in such a manner that each node has adequate tools to control and carry out the transmission of data [28]. For example, the medium access protocol (MAC) used by nodes in a MANET uses a Distributed Coordination Function (DCF) and a random independent backoff timer to control the medium access through the use of Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [59].
2. **Node Independency**: due to the distributed functionality in a MANET, each node functions as a standalone station. A node is capable of transmitting data to other nodes, receiving data from other nodes and routing data to destinations or next hops [88]. Because of this independence, MANETs do not use central control nodes. The ability to function without central control enables the easy and fast deployment of MANETs [85].
3. **Dynamic Network Topology**: node mobility in a wireless network results in a dynamic constantly changing topology [68]. The dynamic topology is a shared feature of both infrastructure wireless networks and MANETs. Nodes in a MANET move regardless of base stations and any fixed infrastructure.

1.3. Challenges of MANETs

Due to their wireless mobile nature, MANETs face a number of challenges. Such challenges can significantly affect performance of the network. Most of the challenges are also applicable to Infrastructure Wireless Networks. Examples of these challenges are signal fading, noise and interference [15, 82 and 93]. In addition to these challenges, the following two main challenges face the usage of MANETs.

A. **Limited Network Resources:** the two main vital resources for MANETs are channel bandwidth and energy availability [61]. Nodes have to utilise the channel in the best way to achieve the maximum successfully transmitted data size possible. Moreover, mobile nodes must use batteries as their energy sources. Therefore, a MANET must function in an approach that allows the maximum performance using the limited battery lifetime [83].

B. **Transmission Range:** The transmission capabilities of nodes in a MANET are limited by node's transmission ranges. Any two nodes can only communicate when they are within the transmission range of each other [25, 21]. If a node is to communicate with another node outside its transmission range, a third node must provide support and act as a router. For example, Figure 1.2 represents a simple MANET of 3 nodes A, B and C. If node A needs to transmit a message to node C, the only possible way to perform the transmission is through node B. Node B is referred to as a hop [87].

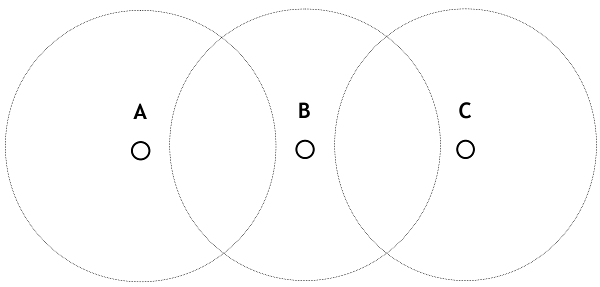


Figure 1.2: A basic MANET formed of three nodes

1.4. Applications of MANETs

Due to their significantly less demanding deployment, MANETs are suitable for application in several areas. Some of these include:

1) **Mobile Voting** [19]: a distributed ad hoc voting application, allows users to instinctively vote on issues across a mobile network.

2) **Military Operations** [80]: Mobile networks can be used in a military battlefield where different military units such as soldiers and vehicles can communicate. This is useful because it is not always possible to setup an infrastructure in such situations.

3) **Civilian outdoor applications** [57]: it is more suitable to have the ability of communicating in outdoor activities without the need of the infrastructure. In many civilian activities, MANETs are used as the main setup. Examples of such situations are taxi networks, moving cars and gatherings in any sport stadiums.

1.5. Motivation

In addition to factor like power consumption, an efficient backoff algorithm should meet at least three requirements. A backoff algorithm should maximize the total throughput of the network, minimize the delay of transmission, and finally, maintain a fair usage of the network among the transmitting nodes. Existing algorithms need improvement in order to satisfy those three characteristics.

The Binary Exponential Backoff (BEB) used in IEEE 802.11 protocol has the problem of the possibility of one node monopolising the communication channel; the last node that has successfully transmitted a packet has the best chance to use the channel again, leaving other nodes in a starvation state. This is known as the channel capture problem, and is directly related to the fairness of the channel usage. Such a characteristic requires a new backoff algorithm to ensure fairness in using the channel.

Improvements to the BEB are supposed to avoid using either too long or too short backoff periods. Long backoff times lead to longer idle time for the network. On the other hand, short backoff periods cause a heavy load on the channel because of the increasing number of channel sensing activities.

BEB uses exponentially increasing backoff window sizes, leading to long backoff periods after a small number of consecutive backoffs and hence, to long network idle time. Therefore, new modifications must use smoother increments on CW.

Simulations of backoff algorithms using wired or fully connected wireless network environments [39] cannot be trusted to indicate the behaviour of such algorithms in MANETs. In MANETs, many factors need to be considered such as mobility, channel bandwidth limitation and power consumption. A simulation for an ad hoc environment is a better way to obtain a trusted evaluation for backoff algorithms if these algorithms are to be developed for such networks.

Current work use external factors along with local ones, in order to reduce the effect of being unable to detect collisions perfectly. Moreover, existing work has not studied the effect of some factors on the performance of backoff algorithms. Examples of these factors are the number of nodes on the network, the degree of mobility and traffic load.

1.6. Thesis Statement

The backoff mechanism dramatically affects the performance of the MAC protocol, and hence the overall MANET performance. The backoff period is directly related to nodes idle times. As a result, the standard exponential back-off scheme has been shown on many occasions to result in long packet delays and low network throughput.

This thesis asserts that:

T1. Although there has been extensive research in the past on optimising the backoff period for wired LANs (e.g. Ethernet) and wireless LANs (e.g. wireless access points), there has been relatively little research activity for wireless mobile ad hoc networks (MANETs), which are characterised by multi-hop routes, various degrees of node mobility and different traffic operating conditions. This research analyses the performance of the backoff mechanism in the context of MANETs taking into consideration a number of important system parameters, including the network size, node mobility speed and traffic load.

T2. While most previous studies have suggested increasing the backoff period after each transmission failure using linear or exponential increments, this research proposes two new backoff algorithms, referred to here as Optimistic Linear-Exponential Backoff (OLEB) and Pessimistic Linear-Exponential Backoff (PLEB) that combine different types of backoff increment to fully exploit the inherent characteristics of MANETs.

T3. OLEB always attributes a transmission failure to a temporary link breakage. For example, due to the sender or receiver being outside transmission range. OLEB uses a linear increment for the backoff window for the first few transmission attempts, determined by a fixed factor. After that, OLEB uses exponential increments.

T4. PLEB always attributes a transmission failure to the presence of congestion in the network, in particular over the shared wireless medium, which may often require a long time to clear. PLEB increases the backoff window exponentially for the first few transmission attempts; determined by a fixed factor. After that, PLEB adopts a linear increment to avoid reaching long backoff periods.

1.7. Contributions

The aim of this research is to study and provide better solutions and mechanisms for the problem of optimising backoff periods in order to achieve a better level of performance. Moreover, this research aims at a better understanding of the concept of backoff with the aim of developing even better solutions in the future.

The contribution of this research starts with a group of extensive simulations of the standard backoff algorithm suggested and used by IEEE 802.11. Moreover, some modifications are applied to the standard BEB and then simulated to produce results that would help to develop new backoff techniques. The simulations performed aimed to study the effect of changing the increment and decrement behaviours of backoff algorithms on network performance.

The second contribution of this work is the first backoff algorithm, namely the Optimistic Linear-Exponential Backoff algorithm (OLEB). This first algorithm is aimed to reduce the increment factor of backoff timer in order to avoid redundant waiting time that might lead to wasting the scarce network resources.

The third contribution of this work is the second backoff algorithm, the Pessimistic Linear-Exponential Backoff (PLEB), is proposed. In spite of the extreme increments performed by this algorithm, network performance has improved compared to the existing previous solutions for the network scenarios presented in this research.

1.8. Outline of the Thesis and Chapter Summary

The rest of this thesis is organised as follows, Chapter 2 covers the preliminaries and basic concepts of backoff algorithms obtained from scanning the literature. In order to achieve a better understanding of backoff algorithms, Chapter 3 performs intensive analysis and performance evaluation of some existing backoff algorithms and some new variations suggested by this research as well. Chapter 4 introduces the Pessimistic Backoff algorithm along with the performance analysis and evaluation extracted from experiments conducted. Chapter 5 presents the Optimistic Backoff algorithm. This chapter covers results and introduces the analysis of these results. Finally, Chapter 6 lists future directions of this work and concludes the thesis.

This chapter has introduced Mobile Ad Hoc Networks explaining features and challenges of these networks. After the introduction, this chapter has provided a look of the related work in the literature followed by the main motivations behind conducting this research. Next, this chapter has continued to list the thesis statements and has then moved to emphasize the contributions of this thesis. Finally, this chapter has outlined this thesis.

The following chapter presents the main concepts and preliminaries and provides the setup of experiments conducted in this work to complete the introductory part of this thesis before Chapter 3 starts reporting experiments and analysing results.

Chapter 2. Preliminaries

2.1. Introduction

In the field of computer networks, MANETs have become an attractive subject for academics and researchers [76, 81, 40, 91, 51, 53 and 66]. This is also true for mobile wireless networks. Moreover, MANETs have created a centre of attention in commercial product development [2, 94, 78, 89, 44, 30 and 43]. A main feature of MANETs is that they do not need to use fixed gateways for packet routing. As an alternative, each mobile node is capable of functioning as a sender, a receiver and a router so it maintains routes to other nodes in the network. Supported by their flexible nature, MANETs are suitable for various purposes and applications including conference meetings, electronic classroom, and search-and-rescue operations.

The wireless medium used by MANETs has a number of problems related to it. Examples of these problems are; bandwidth sharing, signal fading, noise, interference, etc [62]. Moreover, the main sources of power in mobile nodes are batteries. Taking into account that each node acts as a sender, a receiver and a router at the same time raises the possibility of breaking the connectivity of the network whenever the battery of one node is fully consumed. Hence, designers of a mobile ad hoc network should aim for minimum power consumption. With such a shared medium, an efficient and effective MAC protocol is essential to share the scarce bandwidth resource [77].

Medium access control protocol uses a backoff algorithm to avoid collisions when more than one node is requesting access to the channel. Typically, only one of the nodes has access to the channel, while other contending nodes enter a backoff state for some period (BO) [38]. Based on the features mentioned above, the design of the MAC protocol is a significant factor affecting performance of a MANET.

The rest of this chapter is organised as follows, Section 2.2 Describes the IEEE 802 protocol as the protocol used for wired networks and expanded to be used for wireless networks. Section 2.3 introduces Backoff algorithms and, in order to provide a better understanding, this section classifies backoff networks in order to make it easier to study and improve backoff algorithms in general. Section 2.4 explores the Binary Exponential Backoff. Section 2.5 introduces related work from literature. In order to justify the research methodology, section 2.6 discusses simulation approach and it's suitability of this approach for studying mobiles ad hoc networks. Section 2.7 summarises the chapter and links it to the next chapter.

2.2. IEEE 802.11

The IEEE 802.11 MAC protocol [64] is an example of using both physical sensing and RTS/CTS handshake mechanisms. 802.11 is actually defined as the standard MAC and physical protocols for wireless LANs and is not specially designed for multi-hop ad hoc networks [46]. The MAC sub layer consists of two core functions: distributed coordination function (DCF) and point coordination function (PCF) [41, 42].

DCF controls the medium accessing through the use of Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and a random backoff time following a busy medium period [10]. Carrier sense in CSMA/CA is performed both through physical and virtual mechanisms [45]. If the medium is continuously idle for DCF Interframe Space (DIFS) duration then it is allowed to transmit a frame. If the channel is found busy during the DIFS interval, the station defers its transmission. In addition to RTS/CTS exchanges, all data packet receivers immediately send back positive acknowledgments (ACK packets) so that the sender can schedule retransmission if no ACK is received. The RTS and CTS packets used in DCF contain a Duration/ID field defining the period of time that the channel is to be reserved for the transmission of the actual data packet and the ACK packet. All other nodes overhearing either the RTS or CTS or both set their virtual sense indicator, named as Network Allocation Vector (NAV) for the channel reservation period as specified in RTS/CTS. Basically, a node can access the channel only if no signal is physically detected and its NAV value becomes zero. The RTS/CTS mechanism in 802.11 can also be used in the situations where multiple wireless networks utilizing the same channel overlap, as the medium reservation mechanism works across the network boundaries [12].

While DCF is designed for the asynchronous contention-based medium access, the 802.11 MAC protocol also defines PCF, which is based on DCF and supports allocation-based medium access in the presence of an Access Point (AP). An AP plays the role of a point coordinator and polls each participating (called CF-pollable) node in a round robin fashion to grant medium access on an allocation basis. In 802.11, DCF and PCF are used alternatively if PCF is in effect. Obviously, PCF is basically considered unsuitable for ad hoc networks because of the lack of centralized control in such networks as discussed earlier. But the major advantage of PCF is that it can guarantee maximum packet delay and thus provide quality-of-service in a sense. For this reason, some researchers indeed try to modify the PCF method to make it usable in ad hoc networks [4].

In spite of the problems mentioned above, the IEEE 802.11 standard has rapidly gained in popularity because of its simplicity and ease of implementation. It is actually now widely used in almost all test beds and simulations for the research in ad hoc networks. Hence, it is more appropriate for this research to be based on the IEEE 802.11 MAC protocol.

2.3. Backoff Algorithms

As mentioned earlier, mobile ad hoc networks have two major problems, the shared wireless channel and power saving. When designing the network, these two factors raise the need for an optimum usage of the medium via implementing a suitable Backoff algorithm as a part of the MAC protocol. The choice of the backoff technique affects the throughput and the delay over the network. For an easier understanding of the general form of Backoff functionality, this study divides Backoff algorithms into two main categories; static and dynamic backoff algorithms.

2.3.1. Static Backoff Algorithms

Some researchers [18] have proposed using an optimal fixed value as backoff period suggesting a backoff period of the form

$$BackoffTimer = I, \qquad where\ I\ is\ an\ integer \qquad (2.1)$$

In spite of the fact that the value of I can be carefully chosen depending on many factors; such as the number of nodes in the network, having a fixed value can work under a certain scenario for a specific network topology. In the case of MANETs, the major challenges would be mobility and dynamic topology, i.e. positions of nodes within the network area.

2.3.2. Dynamic Backoff Algorithms

In the second type of backoff algorithms, backoff periods are changed depending on many factors. The most common factor used is the result of last attempt of transmission by the node requesting channel access. In general, dynamic backoff algorithms deploy a customised version of the general formula 2.2. The input of the formula is the current size of Contention Window (CW) and the result of this formula is the new size of Contention Window (CW_{new}). CW_{new} is limited between a maximum value and a minimum value referred to as CW_{max} and CW_{min} respectively. CW_{new} is used then to randomly choosing the value of Backoff timer (BackoffTimer) according to formula (2.3)

$$CW_{new} = \begin{cases} Max(f(CW), CW_{max}), \text{ after successfull transmission} \\ Min(g(CW), CW_{min}), \text{ after a collision.} \\ Min(h(CW), CW_{min}), \text{ after hearing a collision} \end{cases} \quad (2.2)$$

The three functions, f(CW), g(CW) and h(CW) are the functions used by the backoff algorithm to calculate the new CW size after successful transmission, a collision and hearing a collision at another node respectively.

$$BackoffTimer=b, \text{ b is random integer, } CW_{min} < b < CW_{max} \quad (2.3)$$

2.4. Binary Exponential Backoff algorithm (BEB):

The DCF of 802.11 MAC resolves the collisions applying a slotted binary exponential backoff scheme [9].

According to BEB, when a node over the network has a packet to send, it first senses the channel using a carrier sensing technique. If the channel is found to be idle and not being used by any other node, the node is granted access to start transmitting. Otherwise, the node waits for an inter-frame space and the backoff mechanism is invoked. A random backoff time is chosen in the range [0, CW-1]. A uniform random distribution is used here, where CW is the current contention window size. The following equation is used to calculate the backoff time (BackoffTimer):

$$BackoffTimer = (b \ \ MOD \ \ CW) \times SlotTime, \ \ where\ b\ is\ a\ random\ integer \qquad (2.4)$$

The backoff procedure is performed then by imposing a waiting period of length BO on the node. Using the carrier sense mechanism, the activity of the medium is sensed at every time slot. If the medium is found to be idle then the backoff period is decremented by one time slot.

$$BackoffTimer_{new} = BackoffTimer - SlotTime \qquad (2.5)$$

So, according to IEEE 802.11, BEB uses a customized form of the general formula (2.2) described before where;

$$CW_{new} = \begin{cases} 31 & \text{, after successfull transmission} \\ Min(2 \times CW \text{ , } CW_{min}) & \text{, after a collision.} \\ CW & \text{, after hearing a collision} \end{cases} \quad (2.6)$$

If the medium is determined to be busy during backoff, then the backoff timer is suspended. This means that backoff period is counted in terms of idle time slots. Whenever the medium is idle for longer than an inter-frame space, backoff is resumed. When backoff is finished with a BO value of zero, a transfer should take place. If the node succeeds in sending a packet and receiving an acknowledgment, the CW for this node is reset to the minimum, which is equal to 31 in the case of BEB. If the transfer fails, the node starts another backoff period after the contention window size is exponentially increased.

BEB sometimes is referred to as “The truncated BEB” [48]. This means that after a certain number of increases, the exponentiation stops; i.e. the retransmission timeout reaches a ceiling, and thereafter does not increase any further. The ceiling is set at the 10^{th} exponentiation, so the maximum delay is 1023 slot times.

Since these delays cause other stations that are sending to collide as well, there is a possibility that, on a busy network, hundreds of nodes are caught in a single collision set. Because of this, after 16 attempts of transmission, the process is aborted.

BEB has a number of disadvantages. One major disadvantage is the problem of fairness [23]. BEB tends to prefer the last contention winner and new contending nodes to other nodes when allocating channel access. Backoff time is decided by choosing a random backoff value from a contention window (CW) that has a smaller size for new contending nodes and contention winners. This behaviour causes what is known as “Channel capture effect” in the network [86]. Another problem of BEB is stability. BEB has been designed to be stable for large number of nodes. Studies have shown that it is not [26].

2.5. Related Work

The Binary Exponential Backoff (BEB) has been the earliest backoff algorithm [58]. BEB has been used in Ethernet first and then was adopted as the standard backoff algorithm for wireless networks [73]. Since its’ early days, BEB has introduced challenges for wireless networks such as stability [27]. Many proposed modifications to BEB have shown that BEB does not achieve the maximum possible network throughput. This is demonstrated in Chapter 3 of this thesis. The main point of attack on BEB has appeared because of the exponential increment of the contention window size [65]. Research has proposed a modified truncated version of BEB in which the CW has a maximum value and the maximum number of increments is 16 [75]. However, research has reported the same initial shortcomings [39]. [39] has suggested using a history variable that represents the transmission failure history to decide backoff times. However, this mechanism did not change the basic operation originally used in BEB and has not achieve a major improvement in performance.

One of the directions that research on backoff has followed is the introduction of backoff optimisation based on network characteristics. [79] Has suggested that the optimal backoff is based on the total number of nodes in the network. For example, Tifour et al. in [79] have stated that, in 802.11 DCF, after each successful transmission, the CW is reset to CW_{min} regardless of network conditions such as the number of current competing nodes. They have proposed the Neighbourhood Backoff Algorithm (NBA) suggesting that, for each number of nodes (N), there is an optimal value of CW_{min} under which the number of collisions increases, leading to reducing the performance. Although this was an improved backoff mechanism in terms of network throughput, the total number of nodes in a network is not easily obtained in a dynamic environment such as a wireless network because nodes join and leave the network frequently during a network session. In a wireless network, nodes join and leave the network at no predictable basis. Another characteristic of the network that researchers have suggested to use is the traffic load on the network [90].

Z. Haas and J. Deng [18, 50 and 17] have been active in the field of backoff mechanisms. They started by suggesting the Sensing Backoff Algorithm (SBA) [18]. SBA has outperformed the Multiplicative Increased Linear Decrease backoff (MILD) suggested in [8]. MILD is based on nodes hearing collisions of other nodes over the network. After MILD, they developed an improved version of this backoff mechanism to achieve higher network performance levels. In [17] Haas and Deng proposed the Linear Multiplicative Increase and Linear Decrease (LMILD) backoff for use with the IEEE 802.11 Distributed Coordination Function. According to the LMILD scheme, colliding nodes multiplicatively increase their contention windows, while the other nodes overhearing the collisions increase their contention windows linearly. After successful transmissions, all nodes decrease their contention windows linearly. Preliminary study has shown the LMILD scheme out-performs the BEB scheme deployed in the IEEE 802.11 MAC standard and the MILD scheme over a wide range of network sizes.

The operation of the LMILD backoff algorithm for the IEEE 802.11 DCF scheme is based on an additional piece of information available to network nodes in the IEEE 802.11 WLANs. This additional information is the knowledge of the packet collisions on the channel. When a node senses that the channel is busy for RTS packet transmission time and the packet header is not detected and reported by the physical layer, it knows that an RTS packet collision has taken place. The senders of the colliding RTS packets become aware of the collision when the CTS reply is not received before timeout occurs. In addition to this information, nodes will also overhear successful packet transmissions.

In the LMILD scheme, each node experiencing an RTS collision increases its CW by multiplying it by a factor (φ). Any node overhearing a collision with the help of the above-mentioned technique increases its CW by (β) units. When a successful RTS transmission takes place, all nodes (including the sender, the receiver, and all overhearing neighbours) decrease their *CWs* by ß units.

The values of φ and ß control the speed of CWs increment in case of packet collisions. Similarly, the value of ß allows nodes to lower their CWs when a successful channel access takes place. The goal of the LMILD scheme is to dynamically maintain the CW values of all nodes close to the optimum CW value, which maximizes the throughput of the IEEE 802.11 network given a fixed number of competing nodes.

In the LMILD scheme, the failed senders increase their CWs multiplicatively, while neighbouring nodes increase their CWs linearly. Upon successful transmission of an RTS packet, which will most likely result in a successful DATA packet transmission, every node decreases its CW linearly. The ß parameter allows non-colliding nodes to react to packet collisions on the shared channel; similar to the way they react to successful transmissions on the shared channel with parameter ß. Haas and Deng have reported in their published work that the knowledge of collisions over the network is not complete. This supports the argument of this research about the difficulty of knowledge acquiring in a dynamic wireless environment.

In addition to the knowledge acquired about the total number of nodes and not being tested under ad hoc environment, LMILD has assumed that all neighbouring nodes are able to detect the existence of collisions perfectly. This might not be true in a practical IEEE 802.11 WLANs, where other devices, such as Bluetooth devices, share the frequency band. The neighbouring nodes could fail to detect the collided packets due to channel fading or they could mistake other signals as packet collisions. These misdetection and false positive problems may affect the performance of the LMILD scheme.

On the other hand, researchers have proposed new modifications on resetting the CW size after a successful transmission. Instead of resetting CW to the minimum value as suggested by BEB, [74] have proposed using an exponential decrement for CW. Although this modification has reduced the channel capture effect related to BEB, the proposed backoff mechanism has been outperformed in both network throughput and packet delay by many other modifications such as the LMILD mentioned above [74].

2.6. Research Methodology

This section explains the main points related to the methodology of conducting this research. Such points include the selected testing methods in contrast with other possible methods and the justification of selecting them. Moreover, this section describes the environments and scenarios used to test the mechanisms addressed by this work. The description includes the main elements of the environment along with the justification of choices made.

2.6.1. Assumptions

Over the course of this work, extensive simulations will be presented. The simulations conducted assume the following points unless stated otherwise.

- For the full length of simulation, nodes have sufficient power supply. At no point of the simulation lifetime a node goes offline because of lack or power.
- External network interference or noise does not exist. All the data that exist in the network is originated from within the network.
- Each node is equipped with a transmitter/receiver, or transceiver, IEEE 802.11 devices.
- The number of nodes over the network is constant for the length of simulation time. No nodes join nor leave the network for the duration of simulation.

2.6.2. Justification of the Method of Study

After deciding the domain of this study, being performance analysis and development of backoff algorithms for MANETs, the early stages of this research required making the decision of the methodology to use in order to test, measure and evaluate mechanisms and techniques subject to study over the course of this research. This section briefly discusses the different possible methods of research on networks and explains the choice of simulation as the appropriate method of study for the purpose of this work. Moreover, this section justifies the use of NS-2 as the selected simulator, and furthermore, provides information on the procedures followed in order to reduce the possibility of simulation errors.

Network research can be conducted using one of the three common methodologies. The list of possible choices consists of simulation, analytical modelling and test-beds. After careful consideration, simulation was found to be the suitable method of study in this research.

When this research work was undertaken, one option to consider was analytical modelling. In the case of multihop MANETs, analytical modelling is considerably coarse in nature which made it unsuitable to aid the study of backoff algorithms with a reasonable degree of accuracy. It is necessary to understand that, in a mobile network, many factors are involved in developing an analytical model where the relations between these factors are still not perfectly understood. Such factors include mobility speed, traffic load and network size. Moreover, the exact effects of each factor on network performance are not accurately decided making it even more justifiable to use simulation to study mobile networks. However, it should be mentioned that understanding of multi-hop wireless communications has improved during the period of this research. The incorporation of factors resulted from such an improved modelling-oriented research of multihop networks is left as a part of the future work of this research.

The last alternative to simulation considered was using a test-bed. According to the planned course of work for this research, a large number of networks were to be studied. In the case of test-beds, the possible setup is restricted by the physical structure and availability of components. It is true that using a test-bed provides realistic observation of any technique studied. However, the cost and complication of setting the test-bed up have reduced the feasibility of using them in this work. As a trade-off between the accurate realistic feedback of test-beds and the complete outcome of an analytical model, simulation has been chosen as the suitable methodology for this study.

The selection of research methodology is inadequate to start the experiments conducted by this research. One more choice that had to be made was the particular simulator to use in order to run simulations. The convenient choice was to use the popular NS-2 simulator. NS-2 is a discrete event simulator targeted at networking research. NS-2 provides extensive support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks [56, 22]. NS-2 has been extensively used in this work. It has been chosen primarily because it is a proven simulation tool utilised in several previous MANET studies as well as in other network studies. Moreover, NS-2 has been the simulator used in research carried out on backoff algorithms. [47] has performed a survey of 2200 published papers on MANETs. Over 44% of the papers in the survey have used NS-2 as the simulation tool. Figure 2.1 presents the percentages of using different simulators.

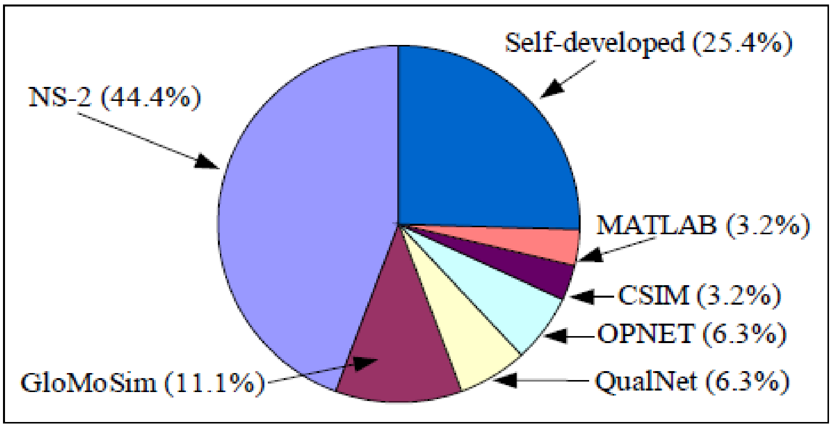


Figure 2.1: Simulator usage in 2200 published papers on MANETs [27]

It is a basic requirement to customise the simulator to meet the needs of this research and to deploy the suggested mechanisms and techniques. During the process of developing modifications to the simulator, special care was taken in order to guarantee that the algorithms implemented would function as designed and that the simulator would not exhibit unwanted side-effects; this has been accomplished through thorough use of the validation suite provided as a part of NS-2. Moreover, careful piecemeal testing of implemented features has been performed. Furthermore, real-life implementations of protocol features, such as the routing agent, were included in the simulations conducted by this research, in order to achieve an approximation that is as close as possible to real system behaviour.

2.6.3. Simulation Parameters

As for the simulation scenarios used in the performance analysis, this work uses three different values for each of the factors considered in this research: namely, the number of nodes, mobility speed, and traffic load. The number of nodes has been set to 10, 50 and 100 nodes. These values of network size combined with the traffic loads used and by controlling the number of traffic sources assure testing for different network loads. Moreover, M. Bani Yassein et al [95] have reported that that average number of neighbours for this network area and mobility model are approximately 5, 10 and 22 for networks of 10, 50 and 100 nodes respectively. These average numbers of neighbours for 100 nodes combined with the traffic load assure covering the maximum number of CW size increments which is 16. For these two reasons, the maximum network size chosen for this research is 100 nodes. The used values reflect the different network size ranging from a small meeting room with 10 nodes, to a classroom of 50 mobile nodes up to the size of a conference location with 100 nodes. [99] Have reported that two network scenarios are equivalent if the parameters in both scenarios have the same values in terms of transmission range R. According to the IEEE 802.11 1997 [100] specification, the transmission range is 20 m. The area used in this work is 4R, R is 250 m, therefore, this scenario is equivalent to an IEEE 802.11 1997 standard network working in an area of (80x80) m^2. This area fits the used example of a conference location and, in some cases, large lecture theatres. Moreover, the chosen values are used to mirror the evaluation held in the literature to measure the performance of existing backoff algorithms [48, 38, 94, 17] and are summarised below in Table 2.1.

S. Papanastasiou [96] has reported that the most frequent path length is approximately 4 hops for similar area. At a transmission range of 250 m, the minimum distance to cover this number of hops is a 1000 m, hence the

(1000x1000) m^2 network area. Random Waypoint was used in [96] and the results have been obtained regardless of node distribution. Moreover, the same point has been investigated in [98]. In [98], Random Waypoint was tested for minimum hop count for 50 nodes in a square area. Results in [98] reported that, for 10 m/s, the hop count is approximately 4 hops.

In the case of mobility speed, this research uses a speed of 1 m/s to simulate human walking speed, a speed of 4 m/s for human running speed and 10 m/s speed to simulate a moving vehicle. The same treatment has been given to the value of traffic load to deploy different levels of load on the network in order to obtain a thorough insight on the performance behaviour of our proposed algorithm.

Table 2.1: A Summary of Simulation parameters

Parameter	Value
Transmitter range	250 meters
Bandwidth	2 Mbps
Simulation time	900 seconds
Pause time	0 seconds
packet size	512 bytes
Topology size	1000×1000 m^2
Number of nodes	10, 50 and 100
Maximum speed	1,4 and 10 m/s

Constant Bit Rate (CBR) [1] allows very tight control over the bandwidth in use at any moment. Therefore, this work uses CBR traffic rates of 1 packet/s, 20 packets/s and 100 packets/s in the simulations conducted. It is worth mentioning here that, the space of possible values of the simulation parameters is theoretically unlimited. The only limitations apply to such space are time and computation power.

2.6.4. The Mobility Model

Simulating MANETs requires a thorough coverage of all aspects of the network protocol used. In order to simulate a mobile network, any conduction of research on mobile networks has to consider a mobility model for the nodes. Research on computer networks has used many suggested mobility models [11]. The random waypoint mobility model [7] is one of the most popular mobility models in MANET research and it is a focal point of relatively heavy research activity [70, 5, 60, 35, 6, 36, 34, 11].

As seen in Figure 2.2, the model starts by defining the network topology as being a collection of nodes that are placed randomly within a confined simulation space that is also known as the simulation area. After that, each node randomly selects a destination within the simulation area and travels towards it with some speed, s m/s. Once it has reached the destination, the node pauses for a predefined time, referred to as simulation mobility pause time, before it chooses another destination and repeats the process until end of simulation time.

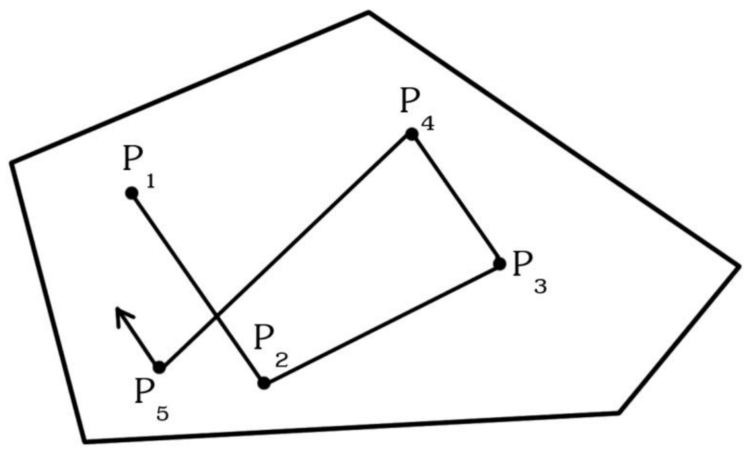


Figure 2.2: Sample movements of Random Way Point mobility model for node P

The node speed of each node is specified according to uniformly distributed values between 0 and V_{max}, where V_{max} is the maximum speed parameter. Pause time and V_{max} are both constants and are fed into the simulator as parameters. In the initial use of the random waypoint model for evaluation [70], an increase in mobility was simulated by increasing the maximum speed parameter or decreasing the pause time.

Other mobility models suggested for research on wireless networks include a variation of the Random Waypoint called Random Waypoint on the Border (RWPB) [33]. In this model, the initial distribution of nodes is near the borders of the simulation area. Another model is the Markovian Waypoint Model (MWP) [34]. MWP adds the restriction of the next destination depending on the current position of the node.

In this work, Random Waypoint has been selected as the mobility model for many reasons. First, this work aims to study the performance of MANETs under the effects of a limited set of parameters in order to allow an acceptable degree of control over the experiments while assuring the exclusion of side effects of the complexities introduced by any other aspects of the experiments such as the mobility model. Secondly, this work studies the network parameters for more general environments. This point can be missed by using a mobility model developed for specific network scenarios. Thirdly, Random Waypoint has been used by existing research considered in this thesis. Therefore, it has been selected to mirror related work for comparison purposes. Finally, up until the point where this research has started, no realistic mobility models have been suggested to reflect real life mobile networks.

2.6.5. Performance Measurements

In this work, the analysis measures the performance using two different criteria that directly relate to backoff mechanisms.

- Total network throughput: this is the total data successfully received at a time unit and measured in multiples of Bytes per Second (bps).
- Average packet delay: this is the average of total delays faced by packets between source and destination and measured in milliseconds (ms).

This thesis presents the results gathered from simulations using 95% confidence intervals. Figures throughout this work contain error bars to represent errors in measurements. However, error bars might not exist in some figures for clarity and representation purposes.

2.7. Summary and Link to Next Chapter

This chapter has described backoff algorithms and their basic operation in order to give a proper introduction to the research of this thesis. The chapter has also provided a general overview of backoff algorithms. It has then provided justification of the research methodology and the explanation of using NS-2 simulations as the method of study in this research. Moreover, this chapter has discussed the simulation parameters used in the network scenarios studied in this work. Finally, this chapter has provided a description of network mobility models and then a closer look at the random waypoint mobility model.

After introducing preliminaries and basic background in chapters 1 and 2, the next chapter introduces performance analysis of backoff algorithms aiming to build the basic understanding of factors affecting functionality of backoff algorithms in order to draw guidelines for developing backoff algorithms.

Chapter 3. Performance Analysis of Backoff Algorithms for MANETs

3.1. Introduction

Backoff algorithms have been suggested in the literature for collision avoidance and to increase the utilisation of network resources. In most backoff algorithms, the backoff timer is chosen from a contention window (CW). The size of CW is changed according to the outcome of last attempt of transmission. A failure of transmission leads to increasing the size of CW while a successful transmission leads to a reduction of the size of CW.

Existing studies [90, 18] have shown that changing the exact behaviour of increasing or decreasing CW has a great impact on the performance of the backoff algorithm. Many suggested algorithms [50, 17 and 79] have been shown to achieve better performance than the standard Binary Exponential Backoff (BEB) implemented by the IEEE 802.11 protocol. However these studies have not taken into account a number of important factors which could significantly affect the performance of a real MANET. These include traffic load, number of nodes participating in the network (referred to as network size in this thesis), and node mobility speed. So far, there has not been any study that analyses the effects of these factors on the performance of a backoff algorithm in MANETs. As an attempt to fill this gap, this chapter conducts an extensive performance analysis of backoff algorithms for MANETs under various operating traffic conditions, network sizes and mobility scenarios.

The rest of this chapter is organised as follows. Section 3.2 describes the setup of simulation experiments used in this chapter. Section 3.3 provides the simulation results along with the performance analysis. Finally, section 3.4 concludes the chapter.

In order to gain a good understanding of the performance behaviour of backoff algorithms, this research suggests studying two aspects of the backoff algorithm. Firstly, the increment behaviour needs to be examined. The method used by the backoff mechanism to increase CW size directly affects the balance between reducing the number of attempts to access the channel and reducing channel idle time. Successful collision avoidance will only be possible if adequate time is allowed between any two consecutive attempts to access the channel. On the other hand, a backoff algorithm should avoid unnecessarily long backoff periods. Imposing a long backoff period on a node is directly related to network idle time since the traffic flowing over the network is often unpredictable.

Secondly, the decrement behaviour after successful transmissions is also a major factor that needs to be explored. The backoff algorithm has to decide the reaction of a successful transmission since this decision affects the chances of nodes winning the next contention over the network. Balance should be maintained between extremely long and extremely short new values of CW. Moreover, resetting the counters to an initial value after a successful transmission has been proved undesirable [79]; a node that has successfully transmitted a message has a small window size afterwards. Therefore, this node generates smaller backoff values leading to a higher possibility of winning the next contention over the channel.

3.2. The Increment Behaviour

To provide a closer look at the effect of the increment behaviour in backoff algorithms, simulation experiments have been conducted using three different increment formulas; a logarithmic, a Fibonacci based and the standard exponential used by the standard IEEE 802.11. Both the Logarithmic and the Fibonacci algorithms are proposed by this study: their definitions and motivations are discussed below. Figure 3.1 shows the behaviour of the three increment formulas used in this chapter. In the figure, the size of CW, measured in time slots, is plotted against number of iterations. The iteration number is the number of consecutive transmission failures. As seen in the figure, the three increment behaviours are used in a manner that allows more than one aspect of the problem to be addressed. First, including the exponential increment is the way to study the standard backoff algorithm used by current networks in practice in order to assess its applicability for MANETs. Second, the Logarithmic backoff algorithm represents a backoff algorithm in which CW is increased by larger steps, compared to the standard, to examine the effect of an extreme increment on network behaviour. Using such large increment steps leads to longer waiting times. However, including this algorithm helps to address the possibility of achieving higher performance in terms of throughput and delay in spite of the fact that a waste of network time is implied. The third increment behaviour used in this research, being Fibonacci Backoff, is a more optimistic algorithm. This backoff algorithm expects the transmission failure to be resolved in a short time. Therefore, smaller increments are applied aiming on addressing the possibility of achieving even higher network performance and preserving network resources represented by network lifetime. The figure shows that the logarithmic increment is the largest and the Fibonacci increment is the smallest between the three increment behaviours used.

Figure 3.1 Three increment behaviours

3.2.1. The Logarithmic Backoff Algorithm

The first variant of increment behaviour used in this research is a logarithmic based increment backoff algorithm. According to this scheme, the new Contention Window (CW) is calculated using the following formula;

$$\mathrm{CW_{new}} = \mathrm{Log_{10}(CW)} \times \mathrm{CW} \tag{3.1}$$

By using formula 3.1, the logarithmic algorithm results in larger increment of CW, compared to increments applied by BEB (according to formula 3.2), leading to longer backoff periods.

$$\mathrm{CW_{new}} = 2 \times \mathrm{CW} \tag{3.2}$$

This change of the increment factor is achieved by deriving it from the logarithm of the current value of CW. Figure 3.2 demonstrates the basic functionality of the Logarithmic backoff algorithm (LOG). In the figure, DIFS refers to the DCF inter frame space as mentioned in the table of abbreviations.

```
Step 0: Set BackOffTimer to initial value
Step 1: While BackOffTimer ≠ 0 do
                For each TimeSlot
                    If channel is idle then BackOffTimer = BackOffTimer-1
        If channel is idle for more than DIFS then
                    Send
        If (SendFailure) then
                   CW = log (CW)*CW
                   BackOffTimer = Random x;      1 ≤ x ≤ CW-1
    Else
          CW = Initial value
          BackOffTimer = 0
       Go to step 1

Step2: Stop
```

Figure 3.2 Logarithmic Backoff Algorithm

3.2.2. Fibonacci Based Backoff Algorithm

Most backoff algorithms [31, 38] suffer from a common deficiency due to their inherent operations. Increasing the size of CW in case of failure to transmit tends to rapidly increase the size of CW to even larger sizes. Reaching such large window sizes decreases the expected wait time for a given node to access to the shared medium. Moreover, a large window size tends to contribute to increasing channel idle times, leading to a major waste in the shared channel bandwidth. Motivated by this above observation, we propose a new backoff algorithm to improve performance.

The well-known Fibonacci series is defined by the following formula [63]:

$$\mathrm{fib}(n) = \mathrm{fib}(n-1) + \mathrm{fib}(n-2),\ \ \mathrm{fib}(0) = 0,\ \ \mathrm{fib}(1) = 1,\ \ n \geq 0 \qquad (3.2)$$

This series has a number of interesting characteristics. Amongst these characteristics is a special value called the golden section property [67]; the golden section property is obtained by calculating the ratio between every two successive terms in the Fibonacci series. Figure 3.3 illustrates this property. After a certain number of terms, the ratio converges to a limit of

$$\frac{1+\sqrt{5}}{2} \approx 1.618$$

In our proposed algorithm, we have used fib(n) described in formula 3.2 as the new size of CW, leading to reducing the increment factor when more transmission failures take place and hence introducing smaller increment on large window sizes.

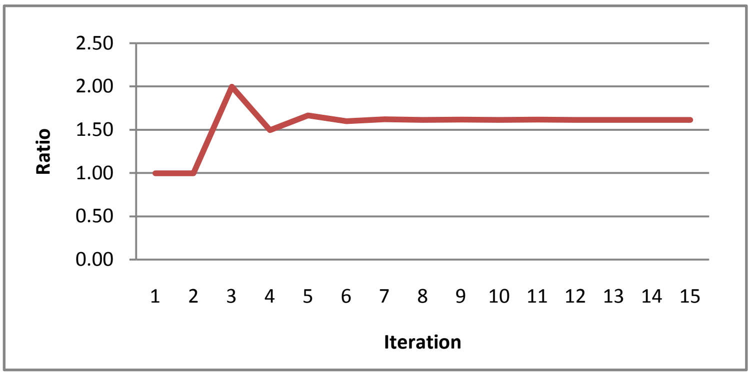


Figure 3.3 Ratio of successive Fibonacci terms.

```
Step 0: Set BackOffTimer to initial value
Step 1: While BackOffTimer ≠ 0 do
                For each TimeSlot
                    If channel is idle then BackOffTimer = BackOffTimer-1
          If channel is idle for more than DIFS then
                    Send
          If (SendFailure) then
                    CW = NextFibonacciNumber
                    BackOffTimer = Random x;     1 ≤ x ≤ CW-1
       Else
                CW = Initial value
                BackOffTimer = 0
          Go to step 1

Step2: Stop
```

Figure 3.4 Fibonacci Backoff Algorithm

It is important to mention here that, the purpose of this chapter is neither to find the optimal value of the Backoff timer nor to determine the optimal behaviour of changing the size of CW. This chapter compares three variations of backoff algorithms in order to provide indications towards choosing the optimal behaviours. In other words, this chapter is to study the effect of changing the values of these parameters on performance levels of backoff algorithms. It is a fact that MANETs introduce the challenge of dynamic network topology with parameters such as mobility. Having this in mind, it is a relatively difficult task to choose an optimal Backoff strategy applicable to all possible variants of a network topology.

3.2.3. Simulation Setup

The backoff algorithms addressed in this chapter have been evaluated using the NS-2 version 2.29 network simulator [56]. The original standard MAC protocol has been modified to implement the variations of the backoff algorithms. Modifications have mainly targeted the mathematical formulas used to calculate new CW sizes. Several topologies and mobility scenarios have been created to test the algorithm as intensively as possible. In order to provide a clearer view of the performance of each backoff mechanism, tests must use a wide range of parameter values. It is true that some values in these ranges lie outside the domain of most anticipated real-life applications of MANET technologies. However, restricting the tests to such scenarios reduces the domain and size of information that can be extracted in this work.

In order to assess the performance of different backoff mechanisms, values of mobility speed, traffic rate and network size had to be fed into the simulator. Firstly, the tests have used variable values for the total number of nodes in the network. Simulations have been carried out for networks having total number of nodes varying between 20 and 100 mobile nodes. These values have been chosen to represent as many network scenarios as possible. Moreover, these values are used to reflect the parameter values commonly used in the literature [48, 38, 94 and 17]. Mirroring existing work does not in itself justify the choices of parameter values. However, using the same values helps in comparisons with previous research. On the other hand, specific real life scenarios form the next step of research on backoff algorithms in MANETs after studying the effect of the largest value space possible and gathering enough evidence of the best backoff behaviours to be used for each different value of the parameters used in this work.

Secondly, in order to address the main challenge of MANETs, this work has used different scenarios with different values for mobility speed. The mobility model is another element needs to be set to decide the pattern of movement directions of nodes. All the nodes move according to the random way point model described in Chapter 2 [36].

Testing for speed values, ranging from 2 m/s to 20 m/s has given useful information concerning the efficiency of the proposed algorithms for both slow and highly mobile MANETs as well. It is unlikely to have such large difference of speed in the same single scenario. However, this work addresses networks with different speeds as separate standalone networks and does not deal with these networks as simultaneously coexisting in the same area.

Other simulation parameters are also set in this work. The first parameter is the area of the network field. We have chosen the area to be 1000m×1000m. Typical node transmission range is 250 m. The traffic generated by nodes is CBR traffic.

Simulation Time

Simulation runtime is one of the major factors to be decided before conducting simulations. Many issues have to be considered in setting simulation time. The following points address these issues.

- In order to reach an environment suitable for reliable data collection, simulation should allow enough time for the network to stabilize. In [97], a survey of mobility models has been conducted. Results have shown that, when calculating the average percentage of neighbours of a mobile node as an indicator of network stability, this percentage changes dramatically for simulation times up to 600 s. The situation starts to have less change after the 600s [97]. This can be seen in Figure 3.5. Based on these results, simulation times longer than 600 seconds allow more stable network.

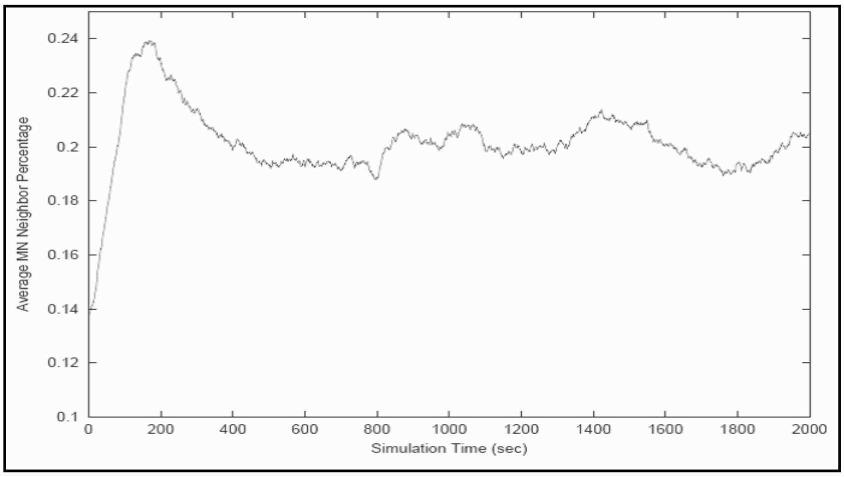


Figure 3.5 Average Mobile Nodes Neighbour Percentage vs. simulation Times for Random Waypoint mobility model [97].

- The increase of simulation time increases the accuracy of extracted results. However, after a certain point, the improvement on result accuracy becomes small enough, within a certain error margin, to stop increasing simulation time. In the preliminary work for this thesis, simulations with runtimes between 100 and 1000 seconds have been conducted. For each of these simulation times, the percentages of change on the number of both sent and received packets have been recorded. Results have shown that these percentages drop to 10% and bellow for simulation times equal to or longer than 800 seconds with no major change of this percentage beyond this time. Figure 3.6 demonstrates the results of these conducted simulations.

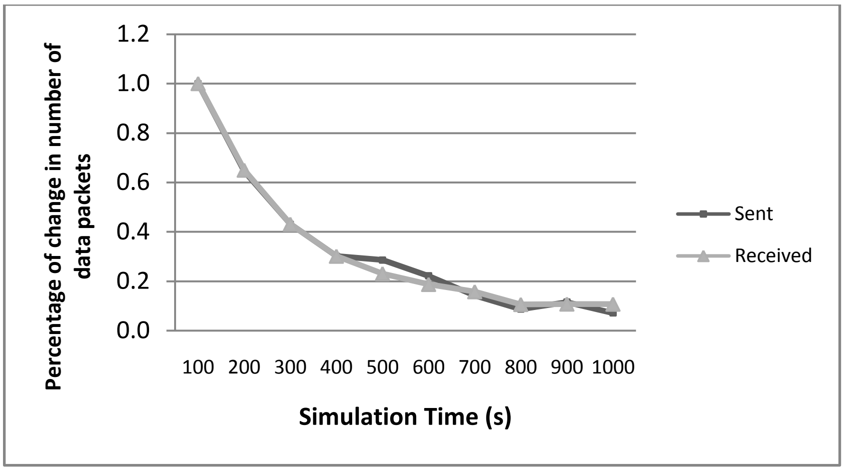


Figure 3.6 Simulation time vs. Percentage of change in number of sent and received packets.

In addition to the change in number of packets, the error of actual data compared to the final result of network throughout reaches 5% after 800 seconds as seen in Figure 3.7.

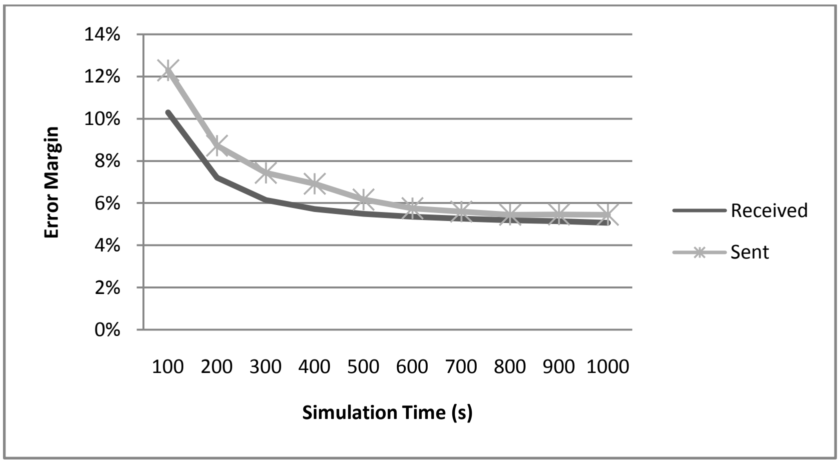


Figure 3.7 Simulation time vs. Result error margin

- Computation power is a major factor in researches similar to this work. Increasing simulation time directly leads to increasing actual runtime.

Since time and computation power are both limited, simulation time should not be redundantly long.

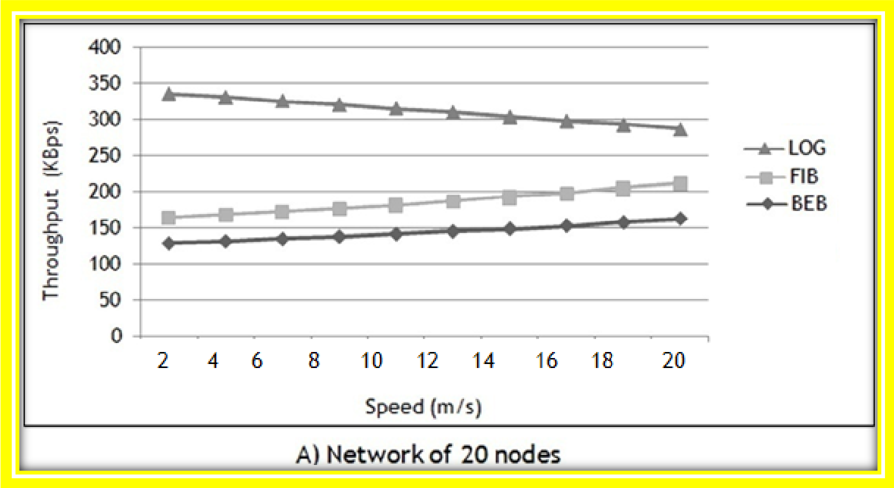
Depending on the discussion above, it has been decided that simulation time must be longer than 800 seconds in order to obtain acceptable results to be presented in this work. Moreover, simulation time should not be significantly longer than the 800 seconds in order to save computation power and time. Therefore, simulations in this work have been run for 900 seconds. The simulations have been left to run for a warm up period before counting the 900 seconds. This means that the 900 s time was used for simulation time but not for warm up time.

Table 3.1 summarizes simulation parameters for this chapter. The rest of simulation parameters have the same values as in Table 2.1 introduced earlier in Chapter 2.

Table 3.1, Summary of the parameters used in the simulation experiments.

Parameter	Value
Number of node	20,30,...,100
Maximum speed	2,..., 20 m/s
Traffic Rate	10 Packets/s

After running the experiments, the results have been analysed and presented in the following set of figures. Figure 3.8.A shows throughput of a network size of 20 nodes. Network traffic rate is 10 packets per second. Figure 3.8.B represents throughput for 30 nodes at the same other scenario parameters. The three algorithms are referred to as BEB for the Binary Exponential Backoff, LOG for the Logarithmic Backoff and FIB for the Fibonacci Backoff. The three algorithms use the same decrement behaviour as the standard BEB.



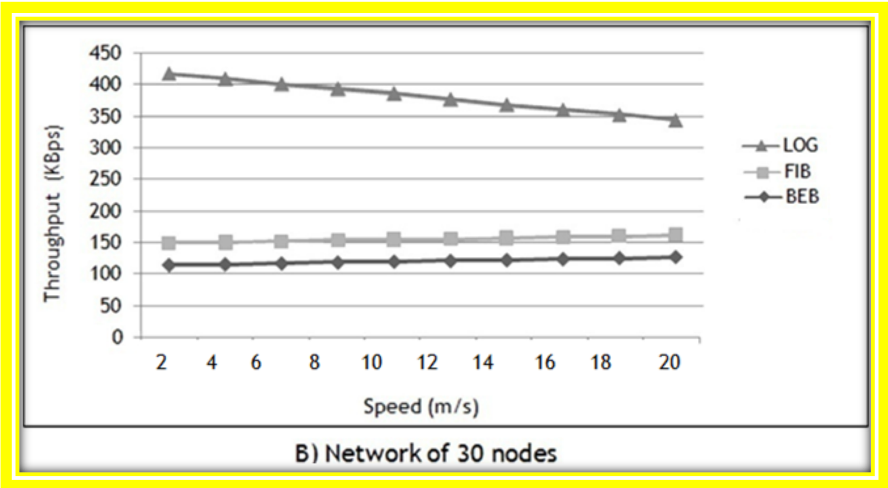


Figure 3.8 Network throughputs vs. mobility speed for LOG, BEB and FIB at traffic rate of 10 packets/s

According to the results, both LOG and FIB improve the total network throughput when compared to the standard BEB. However, considering the difference between the two increment behaviours implemented in the two algorithms reveals two observations. First, the performance improvement indicates that the increment behaviour used by BEB produces sizes of CW that are not optimal for MANETs simulated by the set of experiments. Secondly, as predicted earlier, using larger increment steps proposed by LOG achieves higher total network throughput. When the number of nodes is increased, the contention is higher to gain access to the channel. Because of the larger amount of increment on the window size, a larger size of data was successfully received by nodes over the network. Presumably, it spreads retries out and reduces chances of further collisions. The same enhancement is noticed even while increasing mobility speed. The figure suggests that the lines representing throughput for the three mechanisms would cross at some point. However, this cross will be in abnormal mobility speed at which the network throughput might drop because of transmission failures due to extremely high mobility speed.

One of the major obstacles in the way of developing a MAC protocol for MANETs is mobility. Having a long backoff value allows the node to move outside the transmission range before being allowed to retry accessing the channel. With FIB, the ceiling of backoff periods is controlled to prevent extremely long backoff periods. This can be seen in Figure 3.8.B where the throughput drops for LOG at high mobility speed but does not do so with FIB and BEB where the CW sizes are smaller in comparison with the sizes that LOG produces.

To establish a deep understanding of the improvement achieved by FIB and LOG, this work has studied the performance under multiple values of speed. Figure 3.9 depicts the same conclusions about total network throughout for different network sizes of 40 and 50 nodes under different values of speed. In general, total throughput is expected to increase by increasing network size. The three algorithms exhibit the same trends. The same conclusions can be derived for most of the scenarios simulated in this work.

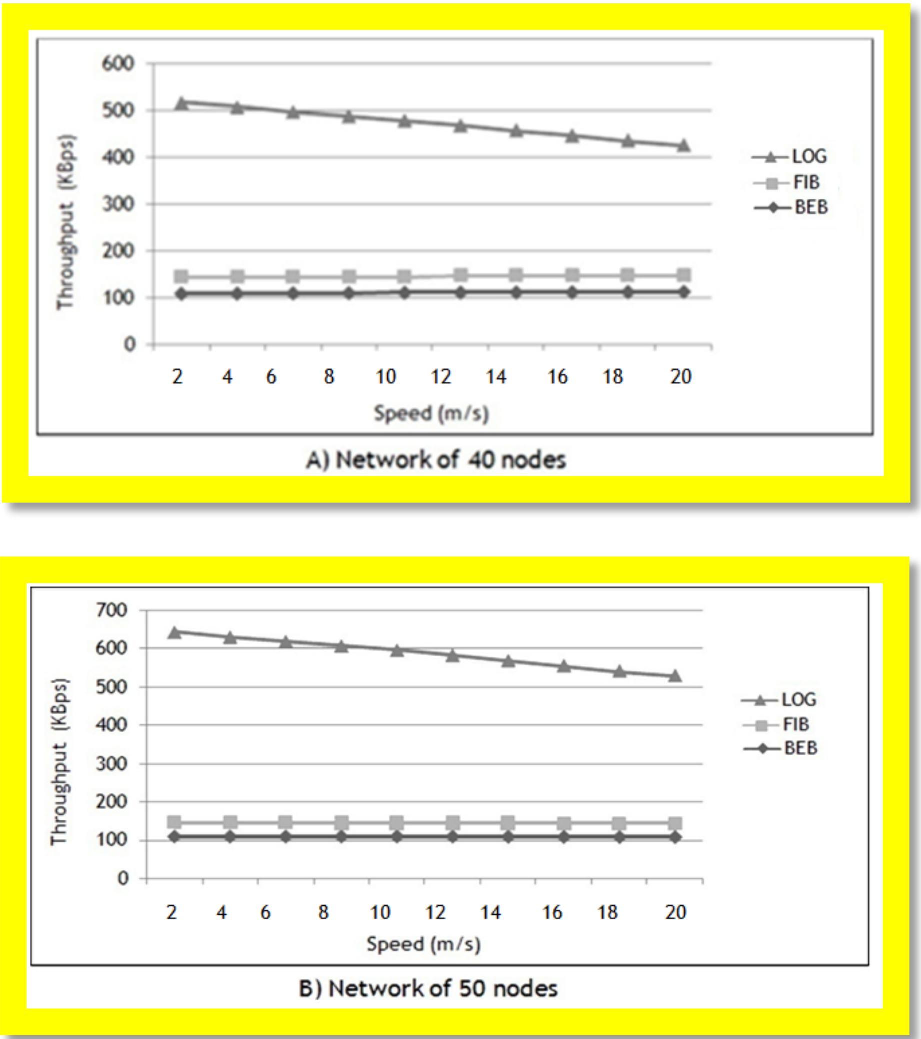


Figure 3.9 Network throughputs vs. mobility speed for LOG, BEB and FIB at traffic rate of 10 packets/s

Increasing node speed for a fixed network size affects network performance the same way increasing network size, which is the number of nodes over the network, does. Again, LOG and FIB have improved the total throughput as seen in Figure 3.10. However, by using larger increment of contention window size, LOG has made it less possible for a high speed node to access the channel before leaving the transmission range.

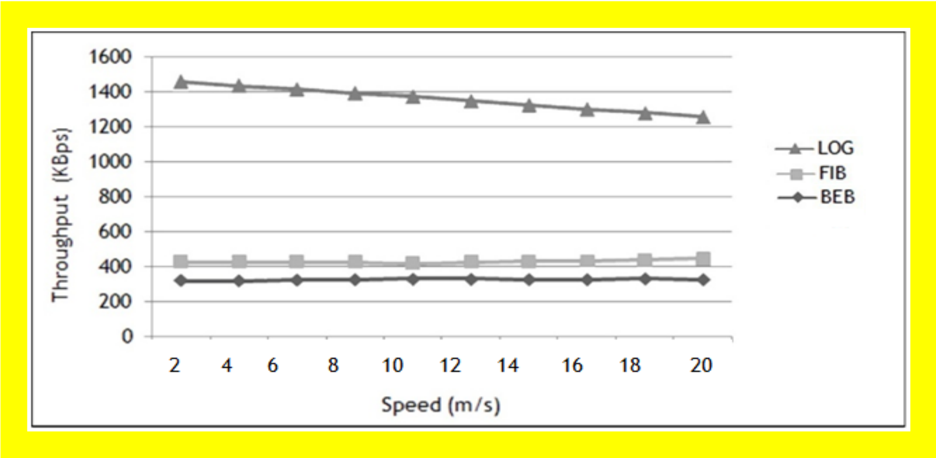


Figure 3.10 Network throughputs vs. mobility speed for LOG, BEB and FIB with 100 nodes at traffic rate of 10 packets/s

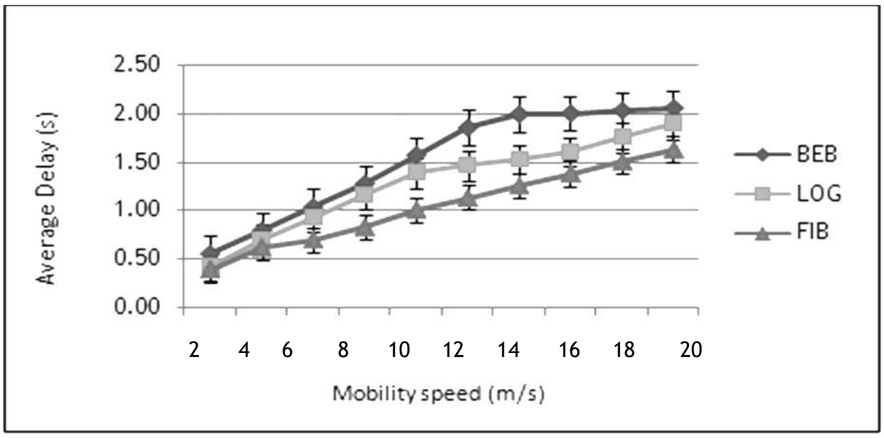


Figure 3.11 Average packet delays of LOG, BEB and FIB for 10 nodes and traffic rate of 10 packets/s

Simulations have also studied the effect of mobility speed on average packet delay. BEB exponential increment causes longer idle time increasing the average delay over the network. Moreover, using a faster increment rate suggested by LOG also increases the average network delay. As seen in Figure 3.11, FIB has reduced average delay compared to BEB and LOG. At this small network size, both values generated by LOG and FIB are smaller than the values generated by BEB. Moreover, the small network size entails smaller numbers of collisions and, hence, increments do not reach high values. For all of the three algorithms in this figure, average packet delay increases with speed. Higher mobility speeds lead to changes in network topology which means that routes and neighbours change at higher rates. This change might lead to longer waiting times and higher contention levels.

Figure 3.12 provides average delay for a network of 20 nodes. At this network size, BEB still has the higher delay than LOG and FIB. In Figure 3.13, the network size is increased to 30 nodes. At larger network sizes, LOG backoff algorithm has longer average delay than FIB and BEB. It is seen in the figure that the average network delay is more affected by speed for BEB and LOG. The smaller increments used by FIB reduce the sharpness of increment on average delay when the number of nodes is increased to 30. However, as seen in Figure 3.11, average delay increases faster at 10 m/s. This is particularly true for BEB and LOG. At higher speeds, the large CW sizes produced by LOG and BEB allow nodes to leave the transmission ranges leading to a need of more time to re-establish the links while packets are waiting for transmission. The same observation is made for 40 nodes as seen in Figure 3.14. The increased number of nodes leads to higher number of collisions. Therefore, the larger increments that LOG suggests produce longer average delay.

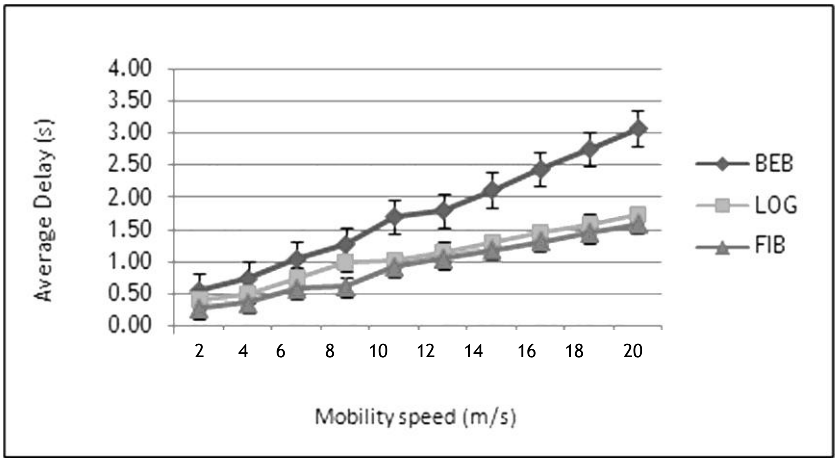


Figure 3.12 Average network delays of LOG, BEB and FIB for 20 nodes and traffic rate of 10 packets/s

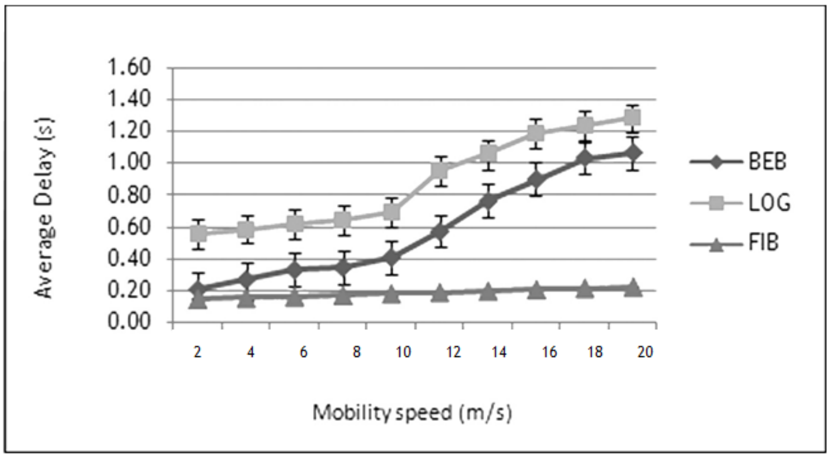


Figure 3.13 Average network delays of LOG, BEB and FIB for 30 nodes at 10 packets/s.

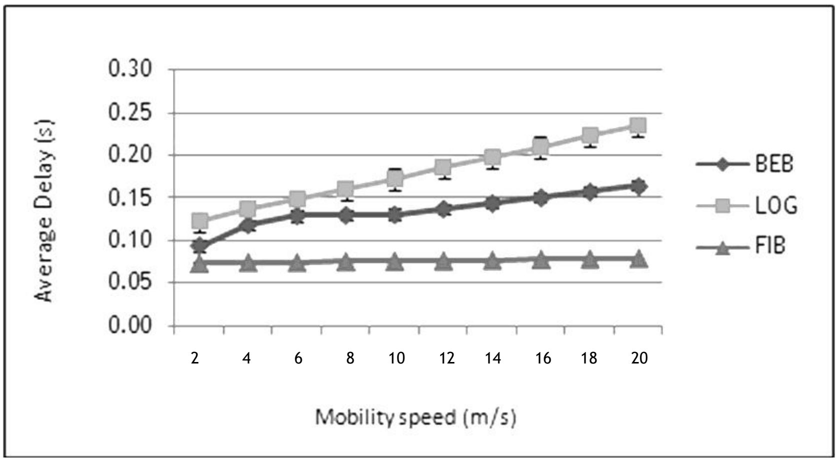


Figure 3.14 network delays of LOG, BEB and FIB for 40 nodes and traffic rate of 10 packets/s

When considering the simulation results, a change of the increment behaviour of a backoff algorithm directly affects network performance measured by the total throughput. According to the results, using larger increment steps increases the total network throughput. Having such an impact is justifiable since longer backoff times lead to less collisions, and hence to a higher possibility of a successful transmission. However, a backoff period should not just increase network throughput. It is an established fact that longer backoff timers lead to longer network delay. Therefore, a trade-off between improving network total throughput and maintaining lower average packet delay controls the development of any new backoff mechanism. On the other hand, using an increment behaviour that assures smaller increment steps, represented by the Fibonacci backoff (FIB) algorithm here, also increases network throughput. The increment of backoff times in FIB insures preserving the fundamental purpose of backoff algorithms, yet reduces network delay by cutting down node idle time while in a backoff state. To sum up, experiments performed in this section have indicated that using larger increments on CW size after a transmission failure produce significantly higher network throughput. However, the cost of larger CW sizes is longer delay. Moreover, this work has introduced the possibility of using smaller increment steps for backoff periods in case of transmission failure which slightly improves network throughput and, at the same time, reduces average packet delay.

It is worth mentioning here that it is unexpected for both larger and smaller increments to achieve higher network throughput. Although the smaller increments introduced by FIB increase network throughput, the difference in performance between FIB and BEB is significantly smaller than the difference between BEB and LOG. This indicates that the improvement on network throughput reflected in the results of this chapter does not certainly prove that smaller increments are more suitable for backoff algorithms. Moreover, the results gathered from simulations have 95% confidence interval. Therefore, the obvious and more certain result is that larger increments are better for the network scenarios addressed in this work, in terms of network throughput.

3.3. The Decrement Behaviour

In the case of a successful transmission, the contention window is reduced or reset to the initial value for the case of the standard BEB. When deciding the decrement behaviour of a backoff algorithm, a balance should exist between two sides of the formula. Firstly, a fast sudden decrement will lead to the channel capture effect as mentioned causing the performance of the network to degrade since the total throughput is decided by the traffic initiated by the channel capturing node. Moreover, leaving other nodes on long backoff times leads to longer idle times, and hence increases average packet delay. Secondly, slow decrement behaviour causes the network to have longer redundant waiting time. This is particularly true when a node can access the channel after multiple transmission failures.

In order to test the effect of decrement behaviour on the performance of a MANET this work presents another set of simulations where different versions of LOG have been evaluated. Suggested decrement formulas vary from applying decrement steps as small as 2 time slots, to the extreme of resetting the contention window to an initial value of 31 which resembles the decrement behaviour of the standard BEB. Table 3.2 summarises the different versions of LOG used where g(BO) is the formula used upon successful transmission as explained in Chapter 2. Moreover, to gain a better understanding of the decrement behaviour, simulations have been performed for a number of network scenarios.

Table 3.2 CW decrement formulas used in the five versions of LOG.

Version	Decrement formula
LOG1	g (CW) = 31
LOG2	g (CW) = CW - 2
LOG3	g (CW) = CW – 4
LOG4	g (CW) = CW - 8
LOG5	g (CW) = CW / 2

While assessing the effect of the decrement behaviour, results have shown that using larger decrement steps increases the throughput of the network. Figure 3.15 shows the total network throughput for a network of 10 nodes. By reducing the CW size after a successful transmission, the size of this decrement decides the probability of the node winning the next contention. As seen in Figure 3.15, using half the size of current CW as the new CW, represented by LOG 5, produces the best network throughput in comparison with the other decrement formulae evaluated. The decrement used in LOG 5 prevents channel monopoly by contention winners and, at the same time, reduces the possible value that will be used for the next backoff timer. This confirms and supports the argument that small decrement steps result in worse network performance because of the redundant network idle time. It is worth mentioning here that the same conclusion is valid for other network sizes. The figures have not been included here to avoid unnecessary repetition of observations.

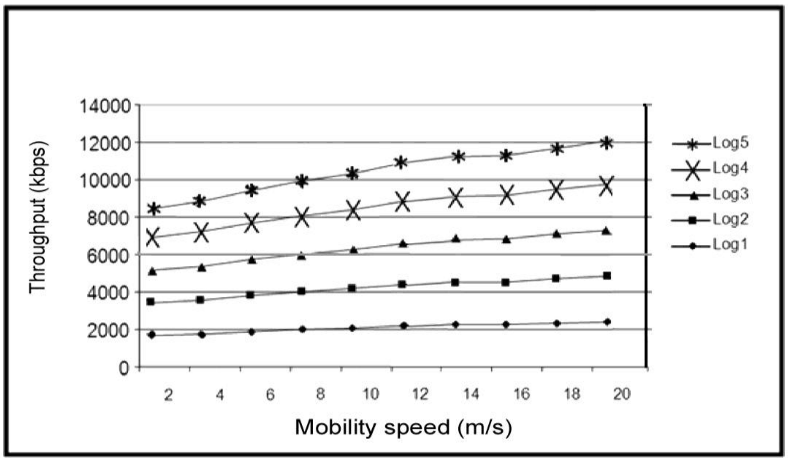
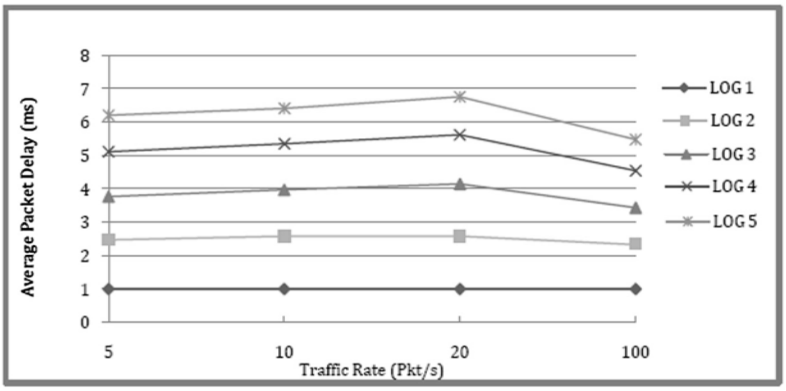


Figure 3.15 Total network throughputs for a network of 10 nodes.

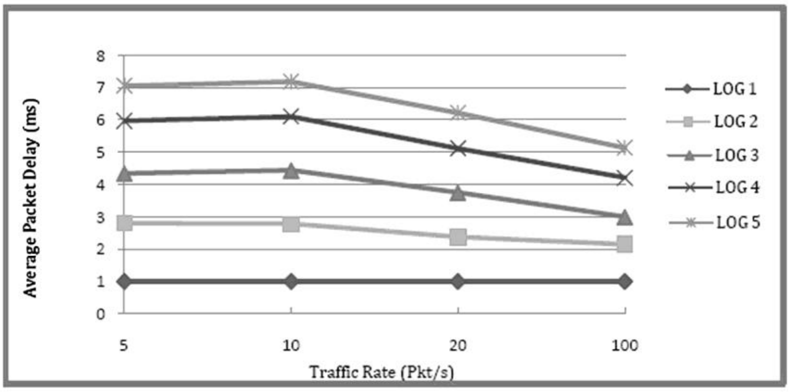
The improvement in the total throughput is inversely related to the new size of CW. Results also show the same behaviour for networks of larger number of nodes. However, the large decrement should not be as extreme as resetting to the initial CW value. Therefore, a point of balance exists to decide how large the decrement should be without causing the channel capture effect.

The task of deciding this point of balance is affected by the characteristics of a MANET such as mobility and network size. However, as the purpose of this chapter is to gather indicators on the effect of decrement and increment behaviours, the investigation of the centre of balance between large decrement steps and channel capture effect is left for the future work of this research.

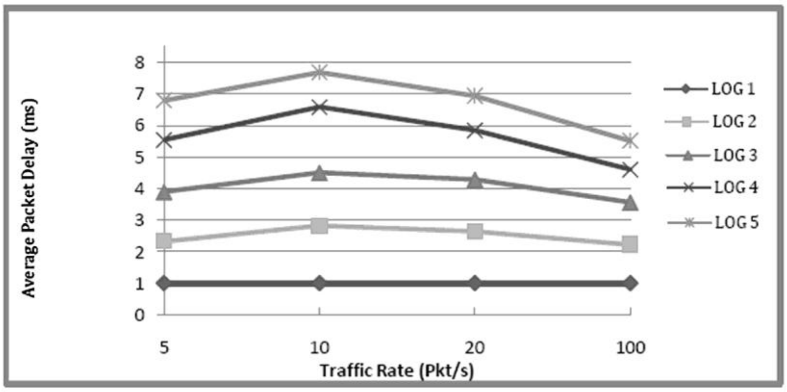
The use of different decrement steps than the reset used in the standard BEB introduces some added delay in the network. Once again, it is the trade-off between network throughput and average packet delay. This is shown in Figure 3.16. However, when using larger decrement steps, average packet delay is shorter for larger network sizes. In small network sizes, the small number of nodes reduces contention. Therefore, smaller decrement leaves CW sizes larger than necessary which leads to redundant waiting times. On the other hand, with



A) Mobility speed 1 m/s



B) Mobility speed 5 m/s



C) Mobility speed 10 m/s

Figure 3.16 Average packet delay for five versions of LOG backoff at different mobility speeds large number of nodes, the media is more likely to be in demand by many nodes. This means that the longer waiting times caused by the new decrement behaviours are not necessarily network idle times. In this figure, LOG1 uses the reset-to-initial upon successful transmission. Therefore, the channel capture effect leads to longer waiting times for the contention losers. The figure also shows that larger decrement steps lead to shorter delays. Moreover, the average packet delay in this work is calculated for successfully delivered packets only and, as seen in results throughout this thesis, the number of delivered packets is lower at high traffic rates. Therefore, the average packet delay is generally lower at high traffic loads. As seen in the figures, average delay drops at the load of 20, 10 and 10 packets/s for mobility speeds of 1, 5 and 10 m/s respectively. Moreover, average packet delay is longer for mobility speeds of 5 and 10 m/s. The higher speed causes packets to face longer delay due to the changing network topology.

To recapitulate, larger decrement steps of contention windows upon successful transmission achieve higher network throughput for the network scenarios simulated. Moreover, with larger decrement of contention window size, nodes converge quickly to the same range of backoff values leading to higher contention and higher delay.

3.4 Conclusions

In this chapter, simulations have been performed to study the effects of changing backoff algorithms on network performance. Changes applied to the algorithms modify increment behaviour upon a transmission failure and decrement behaviour after a successful transmission. Results from simulations have revealed that using different behaviours for increasing and decreasing contention window size directly affects network performance metrics such as network throughput and average packet delay. Changes applied to increment behaviours include both larger and smaller increments compared to the standard Binary Exponential Backoff. According to results, using large increments for contention windows improves total network throughput. However, the large increments have introduced extra delay. On the other hand, using smaller increment steps improves the total network throughput and decreases packet delay as well. The improvement are noticed even when the number of nodes and mobility speed are high.

Changes have also been made to the decrement behaviour. The results have revealed that larger decrement steps have produced higher performance levels. However, the balance between large decrement behaviour and channel capture effect needs still further investigation. This investigation has been left for the future work.

Chapter 4. New Proposed Backoff Algorithms for MANETs

4.1. Introduction

In most existing backoff mechanisms [90, 18], the contention window size is often increased after each transmission failure. For this purpose, the backoff mechanism uses a certain increment method in order to achieve suitable CW sizes that generate backoff timers in a way that maximizes network throughput and reduces average backoff delays. The main two increment schemes used for the Contention Window (CW) sizes are linear increment [74] and exponential increment [37]. Exponential backoff mechanisms have shown failure to achieve the best network throughput and have caused long delays over the network. The well known example of these backoff mechanisms is the standard BEB implemented in the IEEE 802.11 network protocol. On the other hand, linear increment of CW produces slower expansion of CW size. However, the linear increment does not allow adequate time before retransmission. The Linear Multiplicative Increase Linear Decrease backoff (LMILD) is an example of the linear increment behaviour.

This chapter suggests two new backoff algorithms that aim to improve the performance of a MANET in terms of network throughput and average packet delay. In the new suggested algorithms, the exponential backoff is combined with linear increment behaviour. Although the backoff period needs to be incremented after a transmission failure, the increment needs to avoid infinite extensions of the contention window size while preventing too short Backoff periods. This is because short backoff periods lead to repeated attempts to access the shared channel when it is unlikely to have finished the current transmission that caused the invocation of backoff mechanism initially. The combination of the two increment behaviours aims to merge the advantages of the two behaviours. By using the linear part, the proposed algorithms target reducing network delay. The use of the exponential increments aims to produce adequate lengths of backoff times in order to improve network throughput. The simulation results presented later in this chapter reveal that the new suggested backoff mechanisms improve both total network throughput and average packet delay.

The remainder of this chapter is structured as follows. Section 4.2 introduces the first new backoff algorithm, named The Pessimistic Linear Exponential Backoff (PLEB) Algorithm. Section 4.3 introduces the second new backoff algorithm, named The Optimistic Linear Exponential Backoff (OLEB) Algorithm. After that, Section 4.4 describes the set up of experiments. The description includes the details of network scenarios this chapter simulates and the summary of different parameters fed into the simulator. Section 4.5 reports performance results from simulation experiments and analyses network behaviour in order to assess the improvement achieved by the suggested backoff mechanisms. Finally section 4.6 concludes the chapter and outlines some future directions for this research.

4.2. The Pessimistic Linear Exponential Backoff (PLEB) Algorithm

In what follows, the new proposed backoff algorithm is referred as the Pessimistic Linear Exponential Backoff (PLEB). This algorithm assumes that a transmission failure is due to the presence of congestion in the network. This congestion could be the result of a high traffic load present in the network or a larger number of nodes located in a given network region. PLEB works on the premise that congestion is not likely to be resolved in the near future. Therefore, as a first response to a transmission failure, PLEB exponentially increases the contention window size. An exponential increment forces a longer waiting time before trying the next transmission. However, after a number of exponential increments, PLEB starts to increase the timer linearly instead in order to avoid increasing backoff more excessively. The basic functionality of PLEB aims to a less dramatic growth of the contention window size towards the maximum value allowing nodes to perform more attempts to access the channel after a reasonably affordable backoff time.

Figure 4.1 explains the increment behaviour used by PLEB while Figure 4.2 shows the basic functionality of PLEB. In Figure 4.1, the CW size is plotted against the iterations of the backoff algorithms. The iterations depict the number of repeated calls of the backoff mechanism for the current block of transmission failures. As this research adopts the same maximum value for the CW suggested and used in the standard BEB [37], the exponential increment is used until the CW is approximately halfway to the maximum value of 1023.

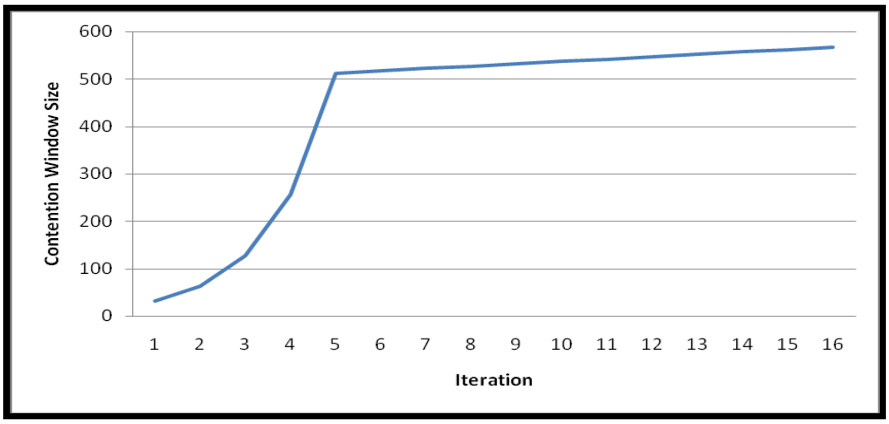


Figure 4.1 The Increment Behaviour of PLEB

```
Step 0: Set BackOffTimer to initial value
Step 1: While BackOffTimer ≠ 0 do
                 For each TimeSlot
                     If channel is idle then BackOffTimer = BackOffTimer-1
         If channel is idle for more than DIFS then
                    Send
         If (SendFailure) then
                     If (NumberOfBackoffs <= N) then
                            CW = CW*2
                     Else
                            CW=CW + T
                   BackOffTimer = Random x;      1 ≤ x ≤ CW-1
      Else
             CW = Initial value
             BackOffTimer = 0
         Go to step 1

Step2: Stop
```

Figure 4.2 Pessimistic Linear/Exponential Backoff Algorithm

4.3. The Optimistic Linear Exponential Backoff (OLEB) Algorithm

Based on the assumption that the current congestion over the network is caused by temporary short-term network conditions and are likely to disappear quickly. Typical network conditions are route breakages that are often repaired quickly. Therefore, the immediate response to a transmission failure is a linear increment of the contention window size, followed by an exponential increment, after (*N*) transmission failures.

The exponential backoff implemented by the standard IEEE 802.11 network protocol introduces reasonably long backoff timers for the first few transmission failures. However, applying such a drastic measure as an exponential increment leads to large values of backoff timers resulting in wasting the limited power of nodes. In order to overcome the problem of redundant backoff times, a new backoff algorithm that implements less dramatic increments for early backoff stages is proposed. For the first (N) transmission failures, the Optimistic Linear Exponential Backoff (OLEB) starts with a linear increment factor first before applying the exponential increment. The value of N has been chosen to allow more use of the linear behaviour. However, further investigation of choosing the value of N is introduced in Chapter 5. Such a combination of exponential and linear increments serves adequately long backoff timers by increasing the contention window size and, at the same time, avoids long redundant network idle times by using smaller increment factor than the case of exponential backoff. Figure 4.3 plots the CW sizes generated by OLEB in the case of successive transmission failures. The figure shows the size of CW against iterations. Iterations here represent the number of consecutive transmission failures of the current node. The description of the main steps of the OLEB algorithm is outlined in Figure 4.4.

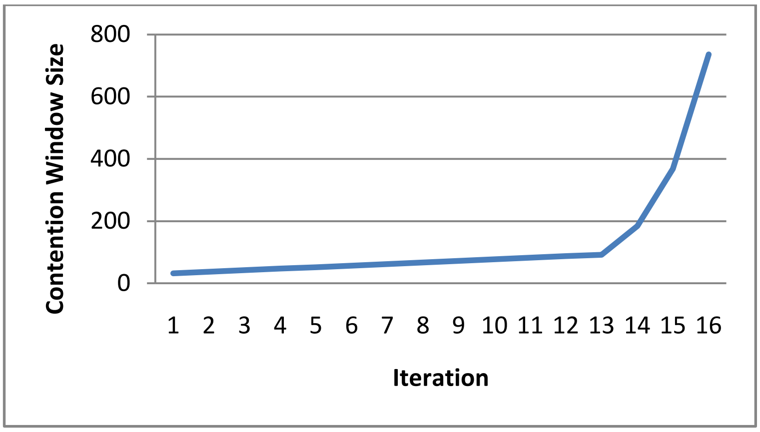


Figure 4.3 the Increment Behaviour of OLEB

```
Step 0: Set BackOffTimer to initial value
Step 1: While BackOffTimer ≠ 0 do
                For each TimeSlot
                    If channel is idle then BackOffTimer = BackOffTimer-1
            If channel is idle for more than DIFS then
                    Send
            If (SendFailure) then
                    If (NumberOfBackoffs <= N) then
                            CW=CW + T
                    Else
                            CW = CW*2
                    BackOffTimer = Random x;        1 ≤ x ≤ CW-1
        Else
                CW = Initial value
                BackOffTimer = 0
            Go to step 1

Step2: Stop
```

Figure 4.4 Optimistic Linear/Exponential Backoff Algorithm

4.4. Experiment setup

This chapter compares the performance of PLEB, against that of the Linear

Multiplicative Increase Linear Decrease (LMILD) algorithm. It has been demonstrated in the literature [74] that LMILD achieves the best performance when compared to other algorithms in the literature including the standard BEB.

As for the simulation scenarios used in the performance analysis, three different values have been used for each of the factors considered in this research: notably, the number of nodes, mobility speed, and traffic load. The number of nodes has been set to 10, 50 and 100 nodes. Such values have been chosen to reflect the different network sizes ranging from a small meeting room with 10 nodes, to a classroom of 50 mobile nodes up to the size of a conference location with 100 nodes. Moreover, the chosen values are used to mirror the evaluation held in the literature to measure the performance of existing backoff algorithms [18, 74 and 17]. These are the same parameters summarised earlier in Table 3.1.

This research uses a speed of 1 m/s to simulate human walking speed, a speed of 4 m/s for human running speed and 10 m/s speed to simulate a moving vehicle. The same treatment has been given to the value of traffic load to deploy different levels of load on the network in order to obtain a thorough insight on the performance behaviour of our proposed algorithm. Constant Bit Rate (CBR) [1] allows very tight control over the bandwidth in use at any moment. Therefore, this work uses CBR traffic rates of 1 packet/s , 20 packets/s and also 100 packets/s in the simulations conducted. It is worth mentioning here that, the space of possible values of the simulation parameters is theoretically unlimited. The only limitations apply to such space are time and computation power.

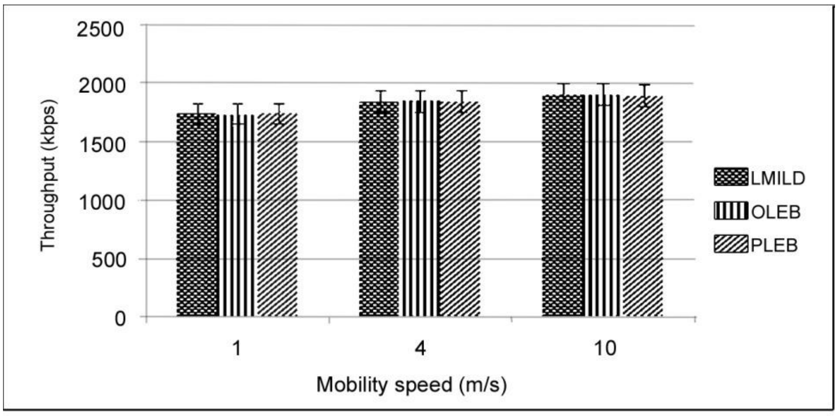
4.5. Results and analysis

In this research network performance is measured by the total network throughput and average network delay. The two measured criteria help to provide better understanding of the level of successfully transmitted data in contrast with the time cost of transmission represented by network delay. The ideal case is to have higher throughput and lower network delay.

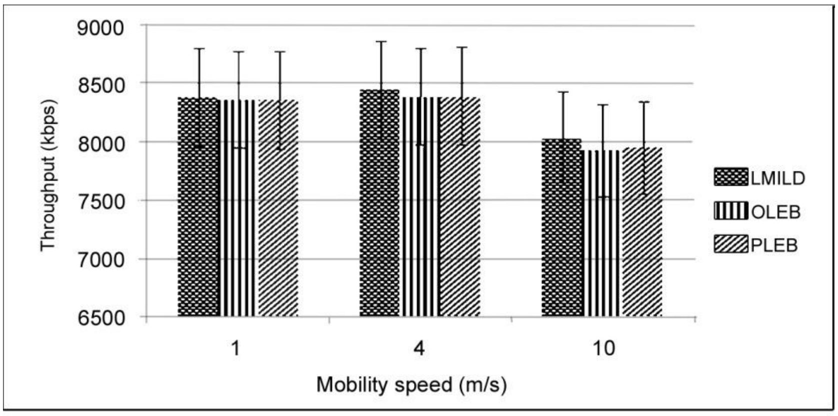
4.5.1. Network Throughput

The main purpose of networking in general is to share and transmit data among nodes. Therefore, the first criterion used to measure network performance is throughput. In this section, simulation results are presented and analyzed to assess total network throughput.

<u>Small network size (10 nodes)</u>



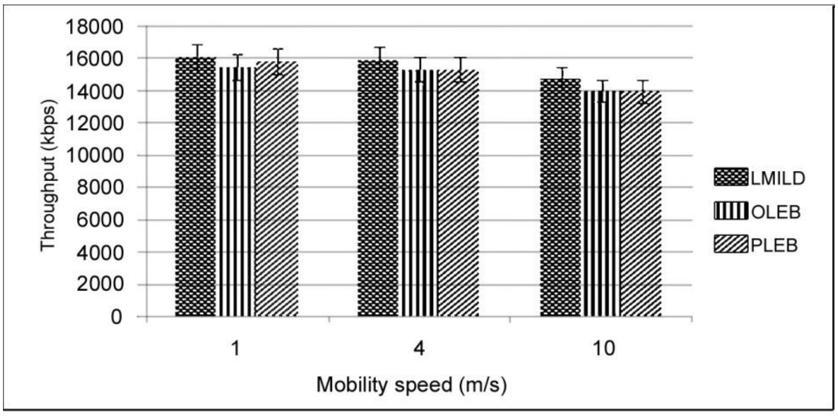
A) Traffic rate 1 packet/s



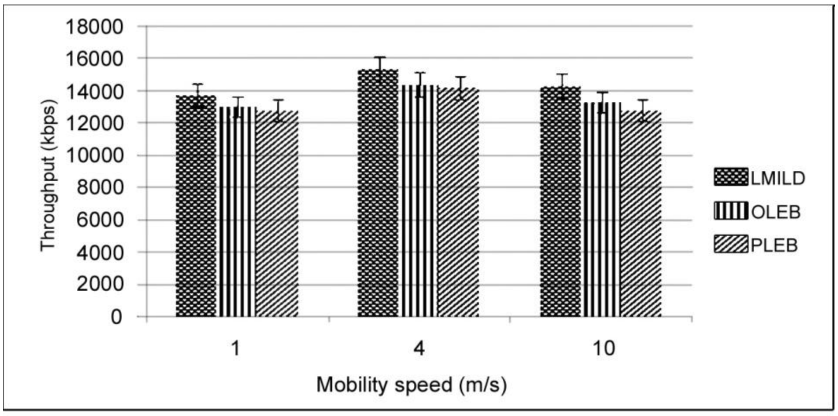
B) Traffic rate 5 packets/s

Figure 4.5 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 10 nodes and traffic rates of 1 and 5 packets/s

Figure 4.5.A presents network throughput results of 10 nodes at a traffic rate of 1 packet/s. At this small network size and low traffic rate, the contention rates are relatively low. Therefore, the three backoff mechanisms examined here achieve similar levels of throughput. This is due to the minimum need for the backoff mechanisms to be used in the first place. In Figure 4.5.B, LMILD has slightly better throughput than OLEB and PLEB at traffic rate of 5 packets/s. This is due the small network size.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.6 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 10 nodes and traffic rates of 10 and 20 packets/s

As shown in Figure 4.6.A, increasing the traffic load to 10 packets/s resulted in more data being successfully delivered to destination. This is an indication on the network functioning normally where, in an ideal world, a network is supposed to successfully deliver more data when higher traffic is being generated. As mentioned earlier, small network size is still a better environment for LMILD to function even with more traffic being injected into the network.

A closer look at the behaviour of PLEB and OLEB in Figure 4.6.A reveals that the former has a better performance than the latter when mobility speed is low. Application of an exponential increment is more appropriate when nodes are moving at lower speeds and less likely to leave transmission range. Nevertheless, a higher traffic has a negative effect on the performance of PLEB since longer waiting times are forced for larger number of data packets waiting all over the network. It is also worth mentioning that at higher traffic rates, increasing the mobility speed has a negative impact on throughput. This is different from the situation in Figure 4.5.A. The increased traffic amplifies the effect of mobility speed since the waiting time imposed by backoff algorithms is most likely to be followed by adjustment to incorporate topology changes which become more frequent with increased mobility speeds.

Figure 4.6.B presents the throughput at 20 packets/s. A network scenario with the traffic rate of 20 packets/s raises two interesting issues. First, in general,

the total throughput levels are lower than the case of 10 packets/s traffic. This result leads to a conclusion that the network is saturated and the increase in traffic is not causing more data to be transmitted. At this point, one possible scenario would be extremely long waiting time so that the new generated traffic is never being transmitted. More data to transmit leads to a higher number of transmission failures and, consequently higher backoff CW values being generated by the algorithms leading to longer idle times and less successful transmissions. Secondly, at a low mobility speed, throughput levels are low. For a higher speed, successful transmissions can be achieved as a result of topology changes. A change of the network topology could help change the route of a waiting packet because of moving outside the transmission range of the current next hop or moving into the transmission range of the destination node.

The same observations are made in Figure 4.7 which represents the throughput results of 10 nodes with traffic rate of 100 packets/s. In Figure 4.6.B and Figure 4.7, OLEB achieves higher throughput levels than PLEB for all mobility speeds used. Because of the small number of nodes in the network, the linearly increased CWs generate shorter backoff timers. The combination of the small number of nodes and the high traffic in the network produce higher network throughput because of the shorter backoff timers produced by OLEB.

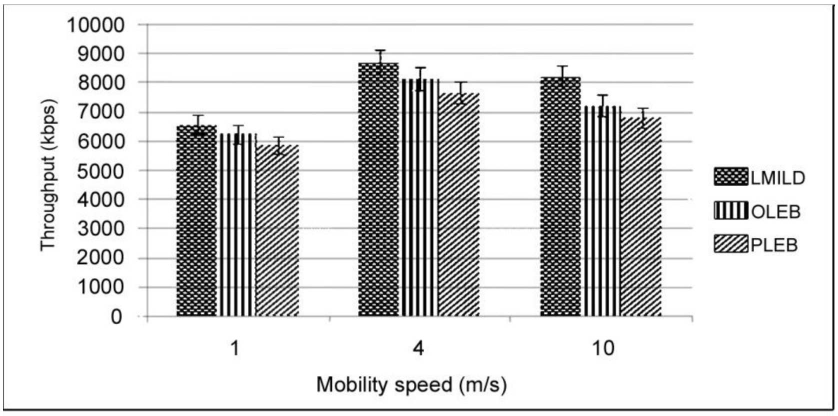
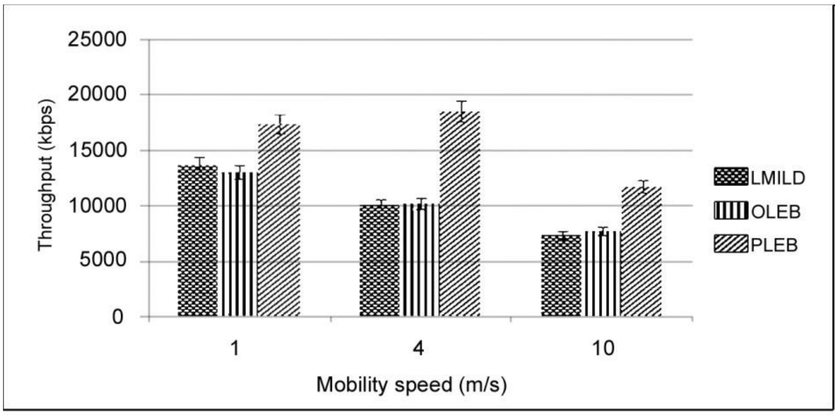


Figure 4.7 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 10 nodes and traffic rate of 100 packets/s

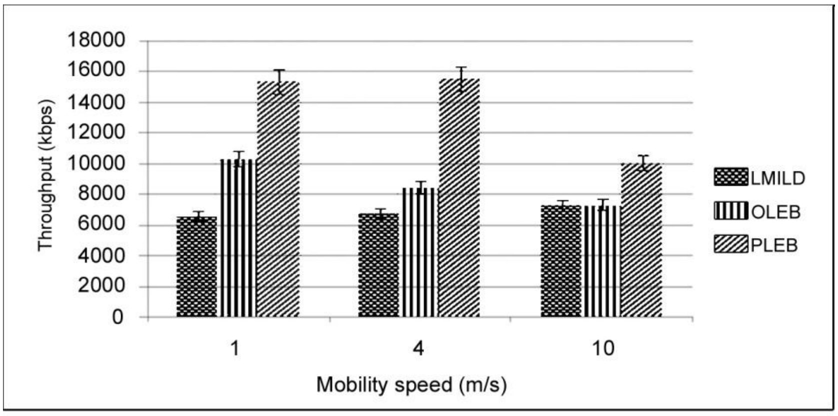
To recap, the results introduced so far in this section indicate that LMILD is the best option among the three algorithms for small network size.

Medium Size Networks (50 Nodes):

Figure 4.8.A shows network throughput with 1 packet/s traffic. When more nodes are added to the network, LMILD is not in a good environment for information gathering anymore. With the added sources of information processed by LMILD to determine the value of congestion window, LIMILD exhibits lower throughput levels than PLEB at low mobility speeds and both PLEB and OLEB at higher mobility speed. At higher speeds, PLEB still has the highest performance levels amongst the three algorithms.



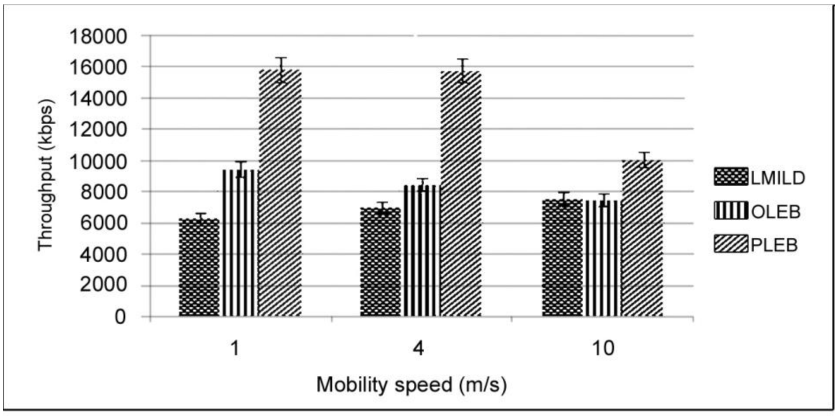
A) Traffic rate 1 packet/s



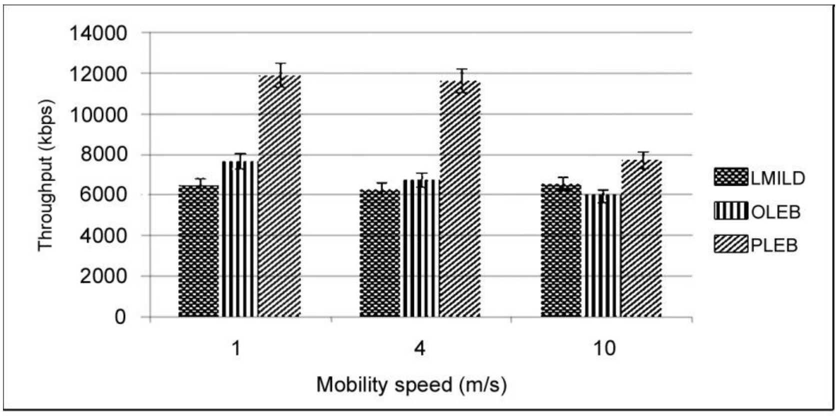
B) Traffic rate 5 packets/s

Figure 4.8 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 50 nodes and traffic rates of 1 and 5 packets/s

Because of the large size of the network and traffic rate, PLEB uses the exponential increments without causing the redundant delay; therefore, PLEB has the best network throughput among the three algorithms. However, it still suffers a drop in performance at high mobility speeds.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.9 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 50 nodes and traffic rates of 10 and 20 packets/s

Figure 4.8.B displays the network throughput after increasing traffic rate to 5 packets/s. At this traffic rate, OLEB outperforms LMILD for low and intermediate mobility speeds. However, at high speeds, performance of OLEB degrades. This is also confirmed by Figure 4.9.A. for 10 packets/s traffic rate. The linear increment without the need for information about collisions over the network gives an advantage to OLEB over LMILD as it can be seen in the results. In general, at higher traffic rates, OLEB has higher performance than LMILD at low and medium speeds but a slightly worse performance at high mobility speed. At

high speeds, the linear backoff suggested by OLEB generates shorter backoff times than needed for the increased contention. The same results hold in Figures 4.9.B for 20 packets/s traffic rate.

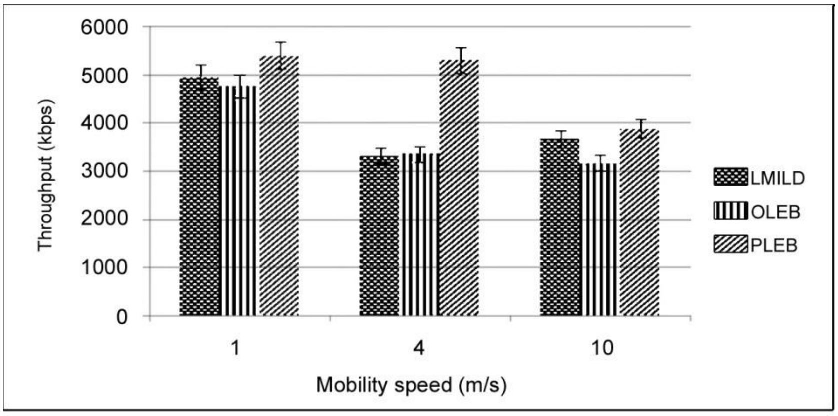


Figure 4.10 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 50 nodes and traffic rate of 100 packets/s

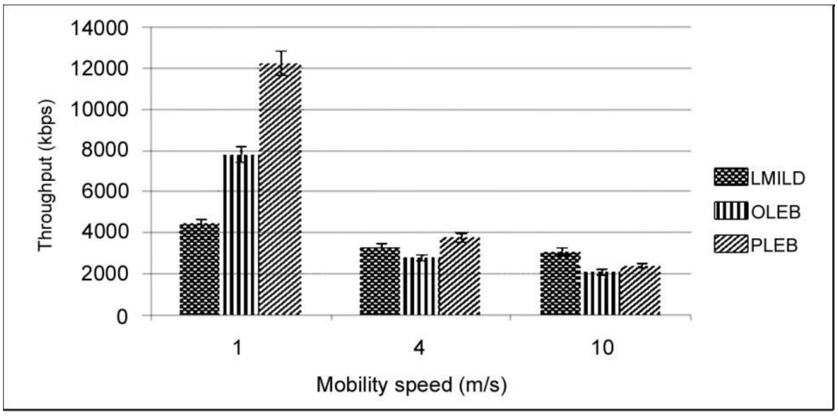
Figure 4.10 displays network throughput readings for 100 packets/s traffic. It is clear from the figure that the network transfers smaller size of data compared to lower traffic rates. This is a sign of network failure to handle such heavy traffic rate.

Large Size Networks (100 Nodes):

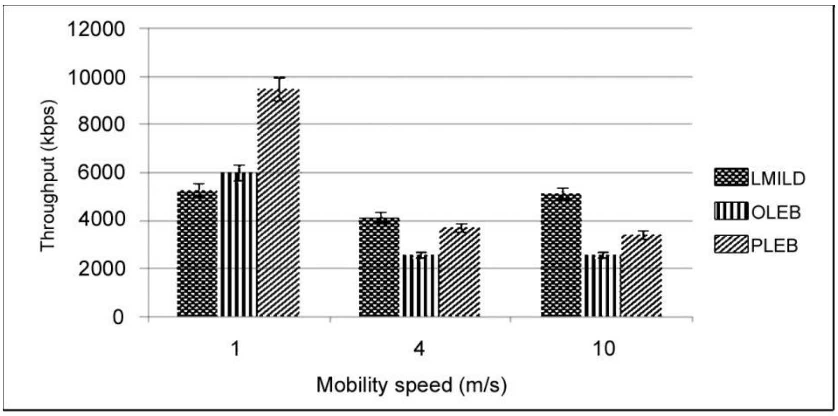
With a larger number of nodes, the network is supposed to face extra high loads and the performance of the backoff algorithms is expected to be dramatically affected. In what follows, simulation results are displayed and followed by discussion. However, the performance of the three algorithms at this network size can generally be described as follows:

- Low speed: at mobility speed of 1 m/s, PLEB has the highest throughput. As for OLEB, its performance is better than LMILD at low traffic rates.

- Medium to High speed: different performance levels can be observed at higher mobility speed. As the traffic rate increases, LMILD starts to achieve better performance. Moreover, the performance of OLEB and PLEB drop faster than the case of LMILD as speed increases.



A) Traffic rate 1 packet/s

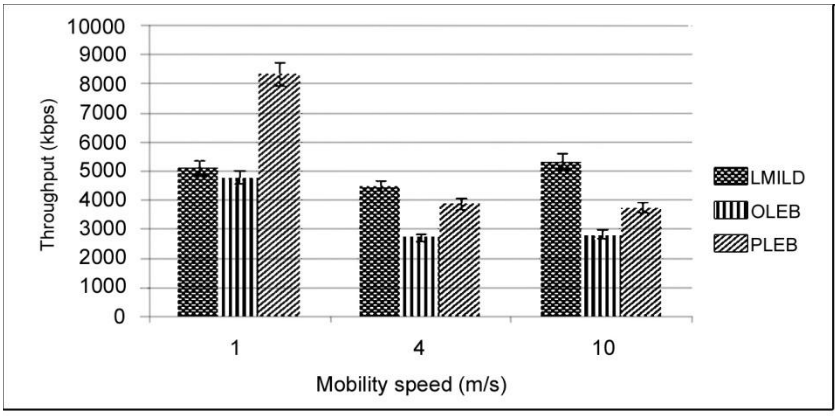


B) Traffic rate 5 packets/s

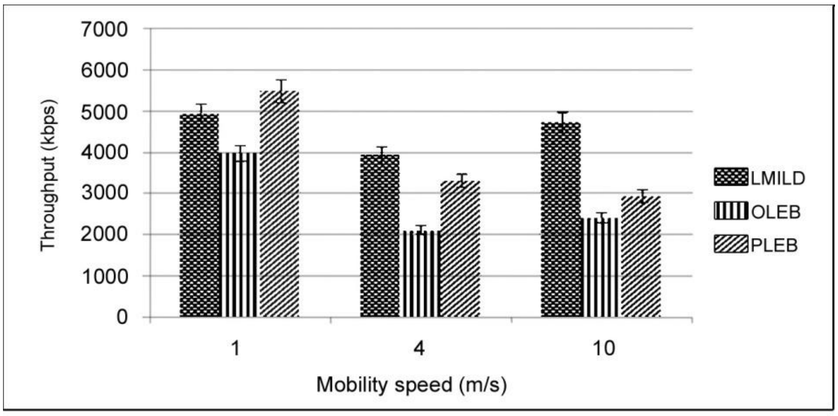
Figure 4.11 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 100 nodes and traffic rates of 1 and 5 packets/s

The first scenario considers the network with the traffic rate of 1 packet/s. As seen in Figure 4.11.A above, at low mobility speed of 1 m/s, PLEB and OLEB achieve higher throughput levels than LMILD. The network-independent functionality of OLEB and PLEB make is easier for backoff timers to be calculated and the superior performance level of PLEB is justified by the adequacy of backoff timers generated using exponentially incremented CW considering that the large number of nodes leads to higher contention over the network. On the other hand, higher mobility speeds have major impact on performance levels. When the speed is relatively high, backoff suggested by OLEB and PLEB introduce long waiting times that are not suitable for a dense highly-changing topology. It is worth mentioning that because of the dependence of LMILD on the number of nodes in the network, the effect of number of nodes is the dominant factor. Therefore, performance levels of LMILD do not change by large values with higher speeds.

As Figure 4.11.B shows, PLEB and OLEB still achieve higher throughput at low speed when traffic rate is increased to reach 5 packets/s. When considering OLEB at low speed, it can be seen in figure 4.11 that the gap between performance levels of LMILD and OLEB is smaller for traffic rate of 5 packets/s compared to traffic rate of 1 packet/s. This is an indication to the linear backoff suggested by OLEB not generating backoff timers that are long enough to achieve a relatively successful channel control.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.12 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 100 nodes and traffic rates of 10 and 20 packets/s

Once again, higher mobility speeds reduce performance levels for the three algorithms. For medium and high mobility speeds, OLEB and PLEB generate longer-than-needed backoff timers. This can be seen in Figure 4.11.B where performance does not change with increased speed. Moreover, LMILD starts to outperform PLEB because the latter generate redundantly long backoff periods.

Figure 4.12.A depicts results for a network of 100 nodes and traffic rate of 10 packets/s. Results in this figure are similar to those shown in the previous

figures. However, at this traffic rate, OLEB starts to achieve lower performance compared to LMILD at low mobility speed. Applying linear increment to all nodes produces close backoff values reducing the performance compared to multiplicatively increasing CW size of at least one node as suggested by the basic definition of LMILD. Therefore, longer backoff values are more suitable between transmissions when more failures take place over the network which is the situation in case of large number of nodes. When compared to PLEB, OLEB does not improve network throughput. This is expected to happen since the high number of contending nodes requires the longer backoff values generated by PLEB. Once again in this graph, there is similarity in performance levels between PLEB and OLEB.

Figure 4.12.B displays the same results seen in the previous figure. However, it can be noticed that the gap between OLEB and LMILD is increasing at low mobility speed when the traffic rate is increased to 20 packets/s. This is not the case for performance levels of PLEB and LMILD at the same mobility speed. At this traffic rate combined with the larger number of nodes, the linear increment suggested by OLEB generates too short backoff values and the exponential increment implemented by PLEB generates too long backoff leading in both cases to wasting the lifetime of nodes resulting in lower levels of throughput.

At the traffic rate of 100 packets/s, Figure 4.13 shows that the high traffic causes the performance of LMILD to degrade because of the high number of collisions that it has to collect information about.

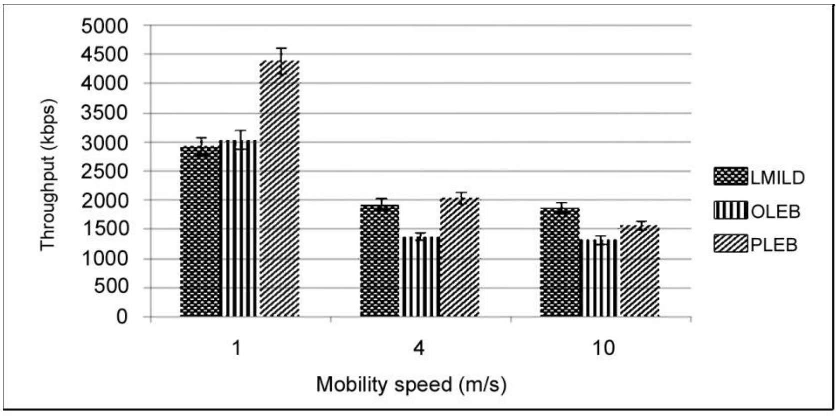


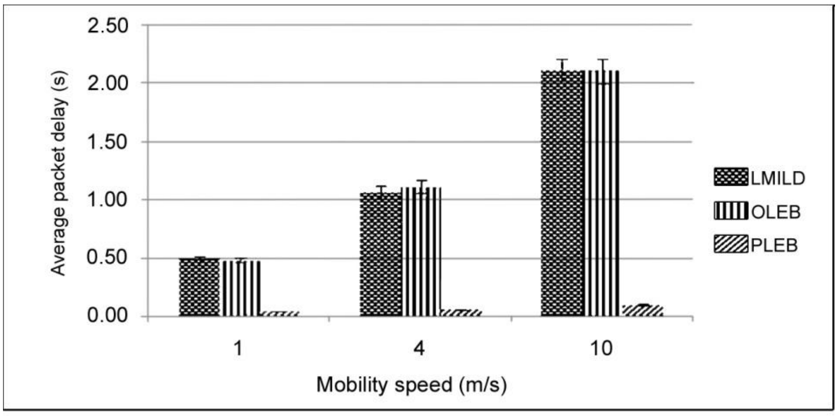
Figure 4.13 – Network speed vs. Network throughput in OLEB, PLEB and LMILD for 100 nodes and traffic rate of 100 packets/s

To recap, in the case of large network size, OLEB has poor performance that can be explained because of generating short backoff timers that are not long enough to deal with the high contention over the network. Moreover, the exponential increment of PLEB generates longer backoff timers than needed, leading to longer idle times and a decrease in the performance level for a network of large number of nodes.

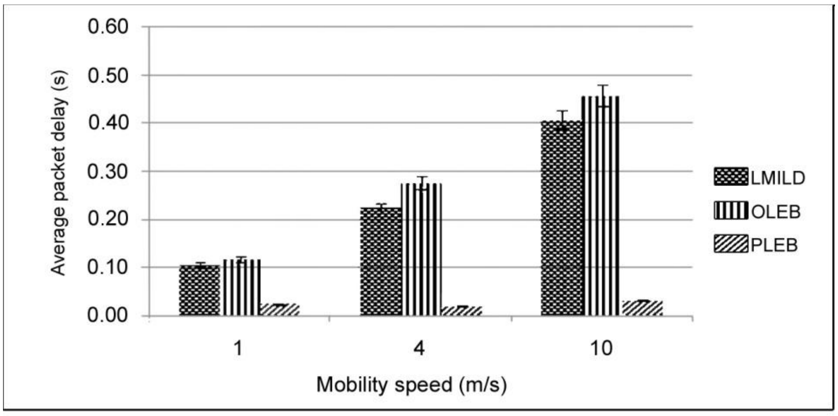
4.5.2. Average Packet Delay

In this section, Backoff algorithms subject to study are analysed by means of the average packet delay. When studying MANETs, a new aspect of importance is added to delay faced by message transmission. In this case, the need of short delays is not only raised by the efficient transmission process, it is also related to the limited life time of a battery-operated mobile node. Long delays are the main source of wasted network resources since nodes are incapable of using the channel to transmit messages.

Small Size Networks (10 Nodes):



A) Traffic rate 1 packet/s



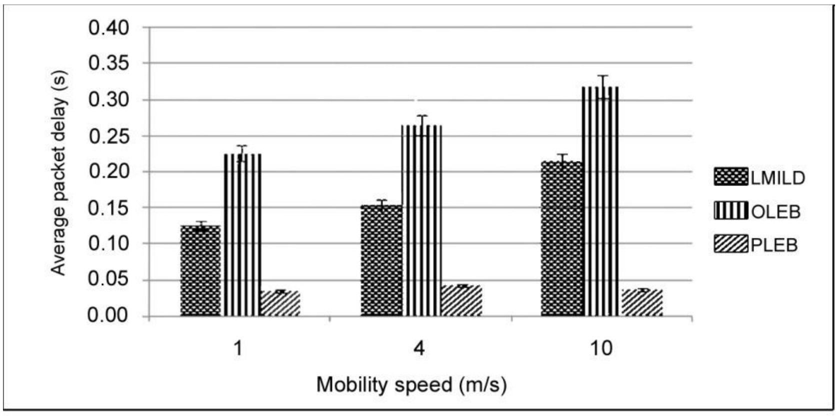
B) Traffic rate 5 packets/s

Figure 4.14 Average packet delay for LMILD, OLEB and PLEB in a network of 10 nodes and traffic rates of 1 and 5 packet/s

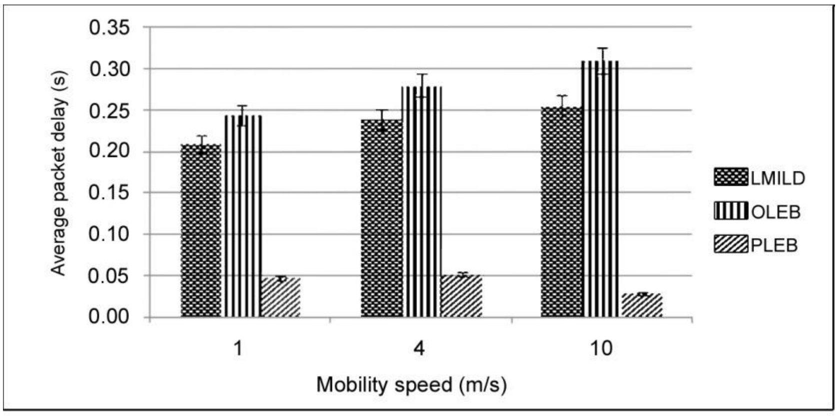
Figure 4.14.A starts this discussion by providing measurements of delay for a network of 10 nodes and a traffic rate of 1 packet/s. As seen in the figure, PLEB achieves low average packet delay compared to OLEB and LMILD. It is true that

PLEB uses exponential increments first. However, at this network size and traffic rate, contention is not expected to be high. Therefore, the exponential start appears to allow enough time for retransmission where in OLEB and LMILD, the algorithms repeat backoff mechanisms since the sizes of CWs produced are not long enough. The average packet delays of OLEB and LMILD are approximately the same. OLEB has similar behaviour to LMILD since the contention is not high and OLEB does not reach the exponential stage of backoff. It is understandable that the network, in the presence of such a small number of nodes and a low traffic rate, does not have high contention leading for a minimum number of calls of any backoff algorithm used.

When increasing the traffic rate to 5 packet/s, the performance gap between OLEB and LMILD is wider for medium and high speeds. This is demonstrated in figure 4.14.B above. At this traffic rate, LMILD has lower average packet delay than OLEB. As mentioned before, a small network size provides an easier task for LMILD since there are fewer nodes and fewer collisions to consider when deciding the next backoff period. Moreover, when proposing OLEB, the linear backoff was expected to causes less network delay compared to the exponential backoff implemented by PLEB. However, the linear increment is repeated because the algorithm does not generate adequate lengths for backoff timers. It is also seen in the figure that high mobility speeds force longer average packet delays for LMILD and OLEB, since a highly dynamic topology along with a small number of nodes provide a rich environment for more broken links and longer waiting times for a link to be established for the messages to be transmitted.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.15 Average packet delay for LMILD, OLEB and PLEB in a network of 10 nodes and traffic rates of 10 and 20 packet/s

Figure 4.15.A represents delay results for a network of 10 nodes but with 10 packets/s traffic rate. In this figure, LMILD is still showing the shorter average packet delay compared to OLEB. However, it is important to notice that using OLEB starts to cause longer delay than LMILD. As mentioned earlier in the basic definition of OLEB, a linear backoff is used first. In the case of a higher traffic rate of 10 packet/s, the linear increment does not produce the needed lengths for backoff periods. Therefore, the backoff performed for early transmission

failures is causing delay without achieving the goal of successful collision avoidance. The linear backoff is followed by the exponential backoff. Therefore, the total times caused by linear and exponential backoffs used by OLEB produces longer delay, even longer than the delay caused by PLEB. As seen in the previous figures, network delay is still higher when nodes move at higher speeds. The figure also shows that the average packet delay of PLEB is higher for this traffic rate. The higher traffic forces PLEB to use more exponential increments of CW.

Figure 4.15.B displays results for a network of 10 nodes with traffic rate of 20 packets/s. The three algorithms show the same behaviours at this traffic rate as they did at the rate of 10 packets/s for all values of speed used. Because of the added load on the network, PLEB starts to generate higher delay since higher contention is expected to exist in such network scenario. Once again, higher mobility speeds produce longer delays. However, it can be seen in the figure that increasing traffic rate does not cause the performance levels of OLEB and PLEB to become closer, this is an expected results since higher traffic rates cause more failures because of the highly contending topology. OLEB switches to exponential backoff while PLEB is using a linear backoff causing OLEB to generate longer average packet delay.

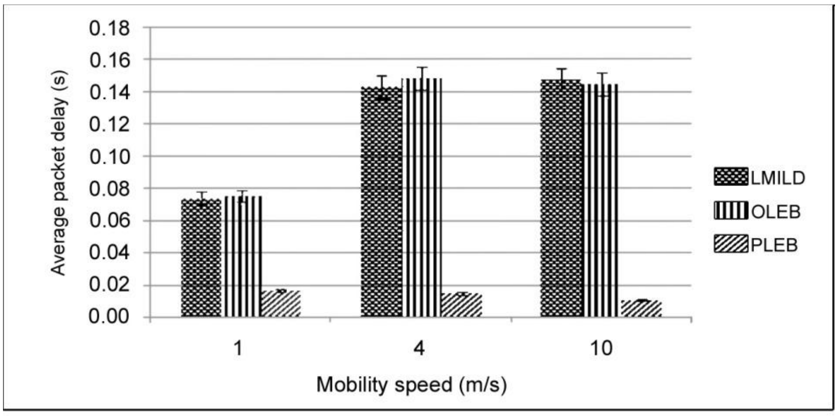


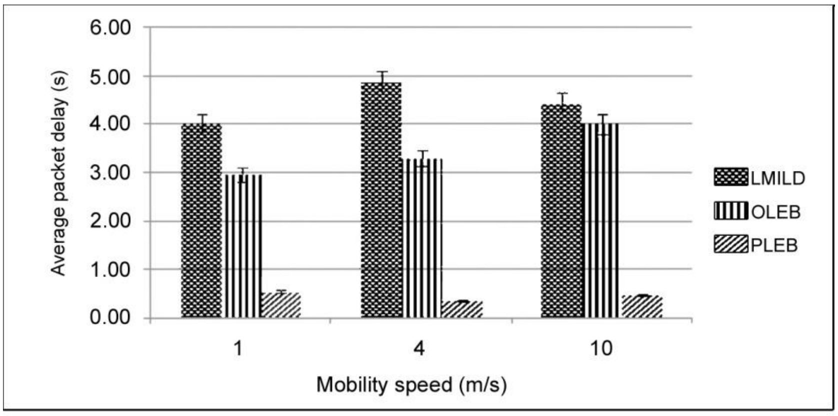
Figure 4.16 Average packet delay for LMILD, OLEB and PLEB in a network of 10 nodes and traffic rate of 100 packet/s

The last figure for this network size, Figure 4.16 above, represents a traffic rate of 100 packet/s. For this network load, LMILD is facing the problem of processing massive numbers of collisions caused by the heavy traffic load leading to longer delays. At this stage, LMILD and OLEB show the same general levels of average packet delay. However, the figures show that, for higher speeds, the network delay of LMILD and OLEB reaches a maximum limit. This means that LMILD and OLEB are producing same backoff values leading to same waiting times indicating that the network cannot transmit anymore packets reaching to saturation. What is seen in this graph is related to the problem of stability that is discussed later in this chapter.

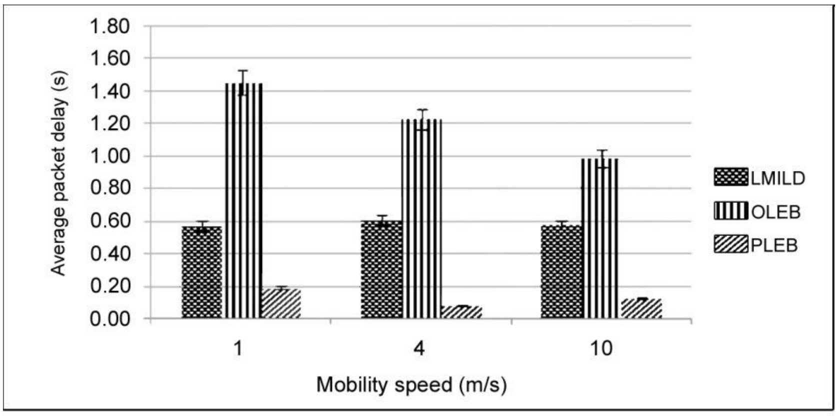
Medium Size Networks (50 Nodes):

For the next set of experiments, a network of 50 nodes is studied for delay measurements. For the first value of traffic rate, figure 4.17.A shows results for

1 packet/s. at this level, LMILD is causing longer delay since there are more nodes to consider when deciding backoff periods.



A) Traffic rate 1 packet/s

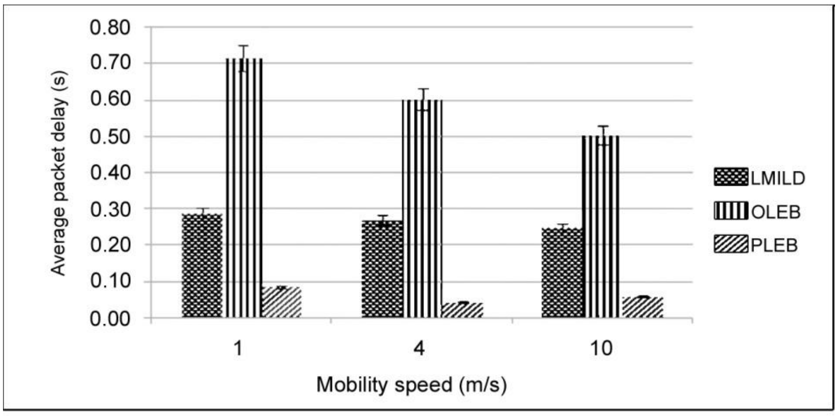


B) Traffic rate 5 packets/s

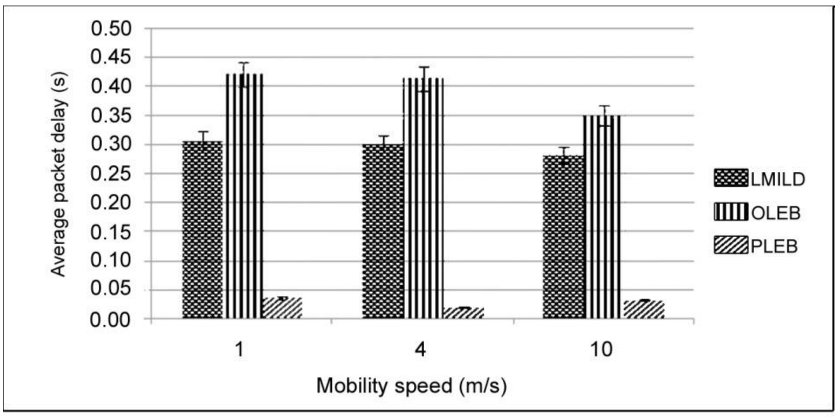
Figure 4.17 Average packet delay for LMILD, OLEB and PLEB in a network of 50 nodes and traffic rates of 1 and 5 packet/s

For all mobility speeds used in this work, the exponential response of PLEB resolves the contentions quicker leaving the performance of OLEB at lower level. However, for higher speed, OLEB starts to cause longer delays. Since OLEB uses linear backoff first, it is less sensitive to high speeds. In Figure 4.17.A, OLEB causes longer network delay when the mobility speed is increased.

A traffic rate of 5 packet/s is applied next. Figure 4.17.B shows that the effect of number of nodes on LMILD is dominated by the added traffic load. For a higher traffic load, the linear backoff used by OLEB is causing longer average packet delays compared to LMILD. When the mobility speed is increased, OLEB causes shorter average packet delays. The dynamic topology allows contentions to be resolved in shorter times since nodes are moving at high speed. With the transmission range used in this work, highly mobile nodes easily enter the transmission range of the current node which leads to more nodes available to help transmitting a packet.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.18 Average packet delay for LMILD, OLEB and PLEB in a network of 50 nodes and traffic rates of 10 and 20 packet/s

The same pattern of results are extracted for traffic loads of 10 packets/s and 20 packets/s as seen in above Figure 4.18.A and Figure 4.18.B respectively. It is also noticed in Figure 4.18.B that LMILD causes longer average packet delays. This is an expected result of the high traffic in combination with the number of nodes. This combination causes more collisions in the network and this leads to LMILD producing longer backoff timers because of the higher number of increments it applies to CWs.

Finally, when the traffic rate is increased to 100 packets/s, the linear increments caused by LMILD and OLEB cause longer average packet delays than PLEB. This can be seen in Figure 4.19.

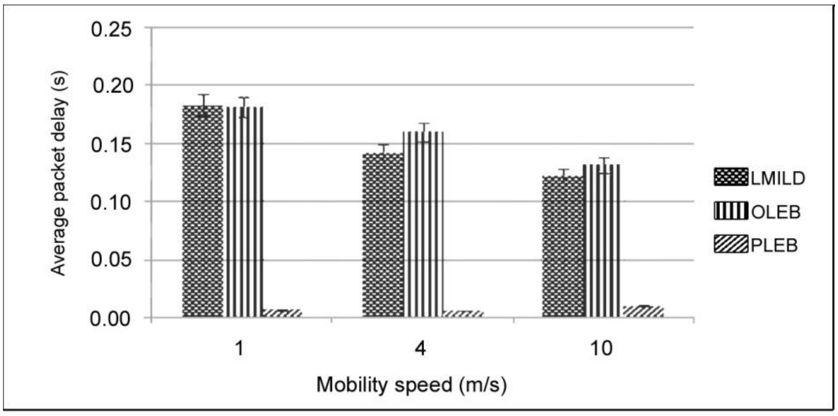
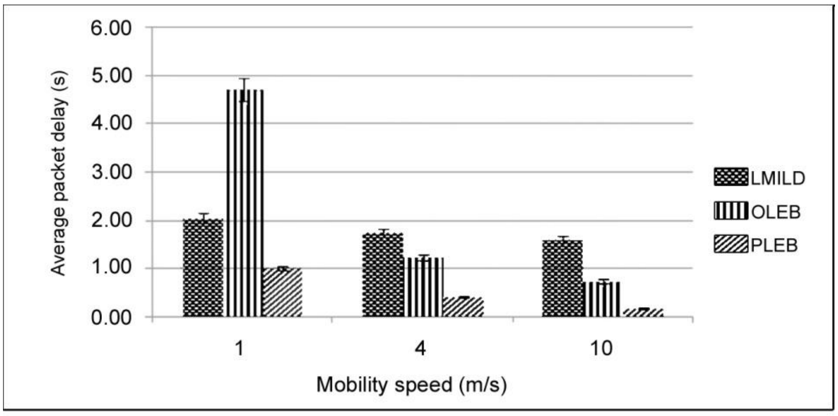
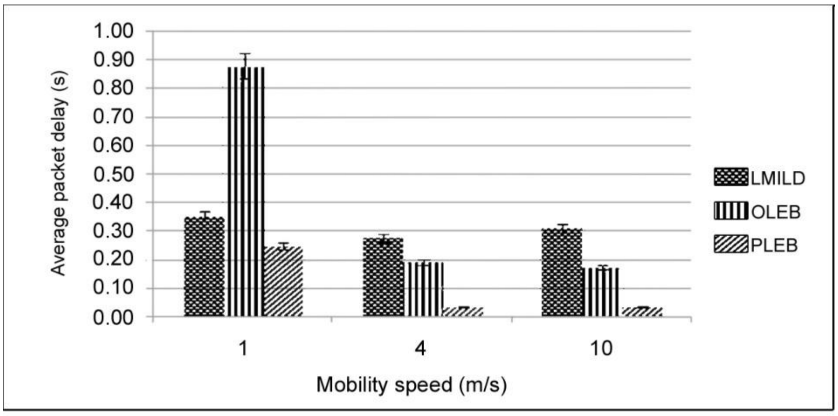


Figure 4.19 Average packet delay for LMILD, OLEB and PLEB in a network of 50 nodes and traffic rate of 100 packet/s

Large Size Networks (100 Nodes):



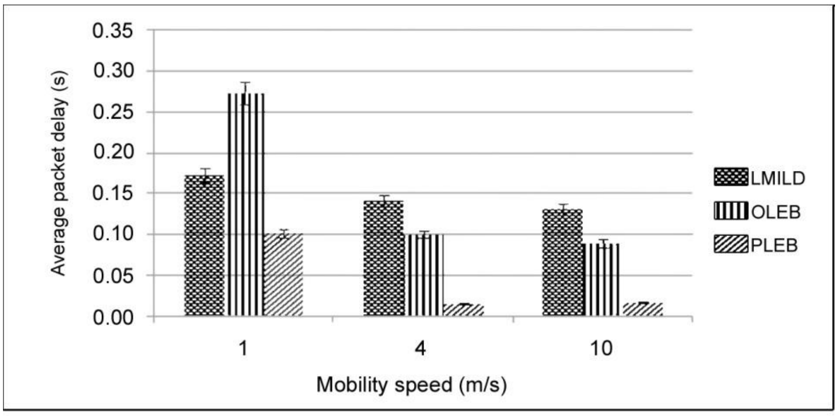
A) Traffic rate 1 packet/s



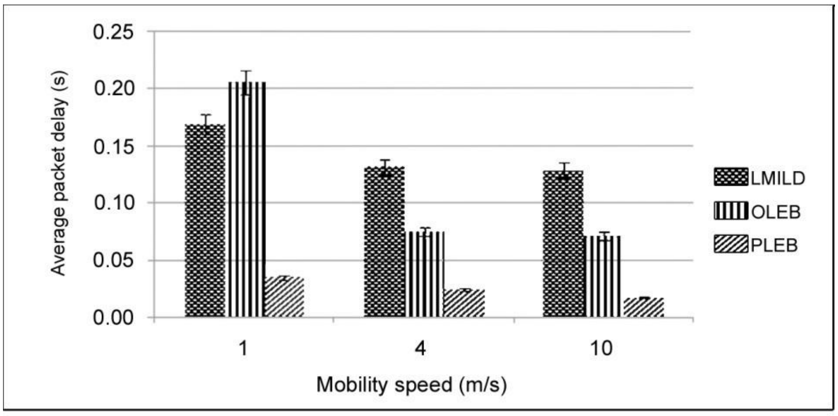
B) Traffic rate 5 packets /s

Figure 4.20 Average packet delay for LMILD, OLEB and PLEB in a network of 100 nodes and traffic rates of 1 and 5 packet/s

Figure 4.20 above shows network delay for a network of 100 nodes. The traffic rate applied here is 1 packet/s. The delay levels are lower for higher speed values. However, such a drop in delay values is not expected since more traffic is generated. This issue is related to the concept of network stability. The following section discusses the stability problem. This same observation is made about the network with traffic rate of 5 packets/s shown in Figure 4.20.B, traffic rate of 10 packets/s in Figure 4.21.A and traffic rate of 20 packets per second presented in Figure 4.21.B. on the other hand, OLEB causes longer average packet delays for low mobility speeds. This indicates that before the network performance drops, OLEB suffers under the higher number of nodes and traffic rates.



A) Traffic rate 10 packets/s



B) Traffic rate 20 packets/s

Figure 4.21 Average packet delay for LMILD, OLEB and PLEB in a network of 100 nodes and traffic rates of 10 and 20 packet/s

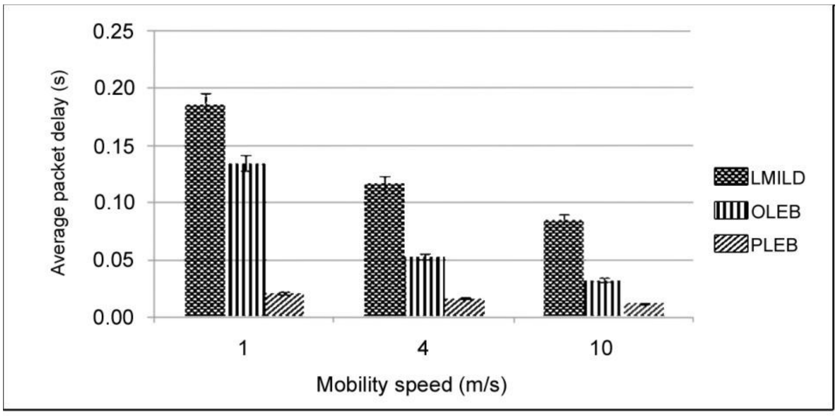


Figure 4.22 Average packet delays for LMILD, OLEB and PLEB in a network of 100 nodes and traffic rate of 100 packet/s

Figure 4.22 displays average packet delays at the traffic rate of 100 packets/s. At this traffic rate, the combined linear and exponential increments of OLEB and PLEB produce shorter average packet delays. However, after noticing the network instability, the future work of this research should study the results in Figure 4.22 in light of network stability conditions.

4.6. Conclusions

This chapter has introduced two new backoff algorithms, referred to as the Pessimistic Linear Exponential Backoff (PLEB) and the Optimistic Linear–Exponential Backoff algorithm (OLEB) to improve the performance of MANETs.

The performance of the new proposed algorithms has been analysed against that of the Linear Multiplicative Increment Linear Decrement (LMILD). The measurements of network throughput have revealed that for a small number of nodes of 10 nodes, the three algorithms addressed in this chapter achieve same network throughputs for low traffic. LMILD has slightly higher throughput at high traffic rates and all tested mobility speeds for this network size. However, OLEB achieves higher network throughput than PLEB for high traffic loads. For a medium network size of 50 nodes, PLEB has shown higher throughput than LMILD and OLEB. Moreover, OLEB has higher network throughput than LMILD. Finally, at a large network size of 100 nodes, PLEB has the highest network throughput compared to OLEB and LMILD. At medium and high mobility speeds, LMILD achieves the best network throughput and OLEB has the lowest network throughput.

OLEB causes longer average packet delay compared to PLEB and LMILD for a small network size. For a network size of 50 nodes, OLEB produces a lower average packet delays at low traffic. However, at high traffic rates, OLEB has a higher delay than LMID. In a network of 100 nodes, OLEB achieves a lower average packet delay than LMILD and OLEB for medium and high mobility speeds.

The throughput outcomes of this chapter can be summarized in the following three figures. Each figure shows total Network Throughput for a network size. In the legends, the algorithm names are followed by a postfix the represents the mobility speed. i.e., for example, LMILD1m stands for LMILD at 1 m/s. Figure 4.23 displays network throughput levels of the three algorithms for 10 nodes. As explained in the chapter, the three algorithms have close levels of throughput with a slightly higher performance for LMILD. The values are close due to the small number of nodes that leads to less use of backoff algorithms in general.

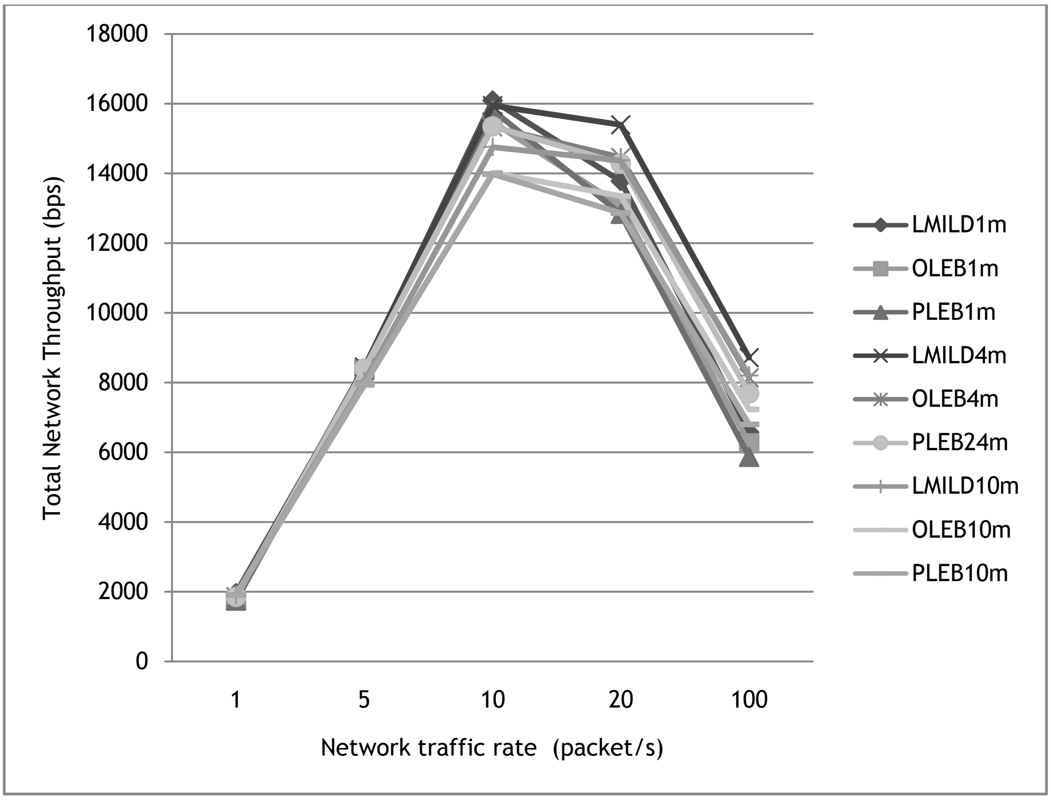


Figure 4.23: Summary of Network Throughput results for LMILD, OLEB and PLEB in a network of 10 Nodes.

Figure 4.24 represents throughput levels of the three algorithms for a network of 50 nodes. It can be seen in this figure that PLEB achieves higher throughput for speeds of 1 m/s and 4 m/s. The figure also shows steep drop of throughout levels for LMILD at 1 m/s.

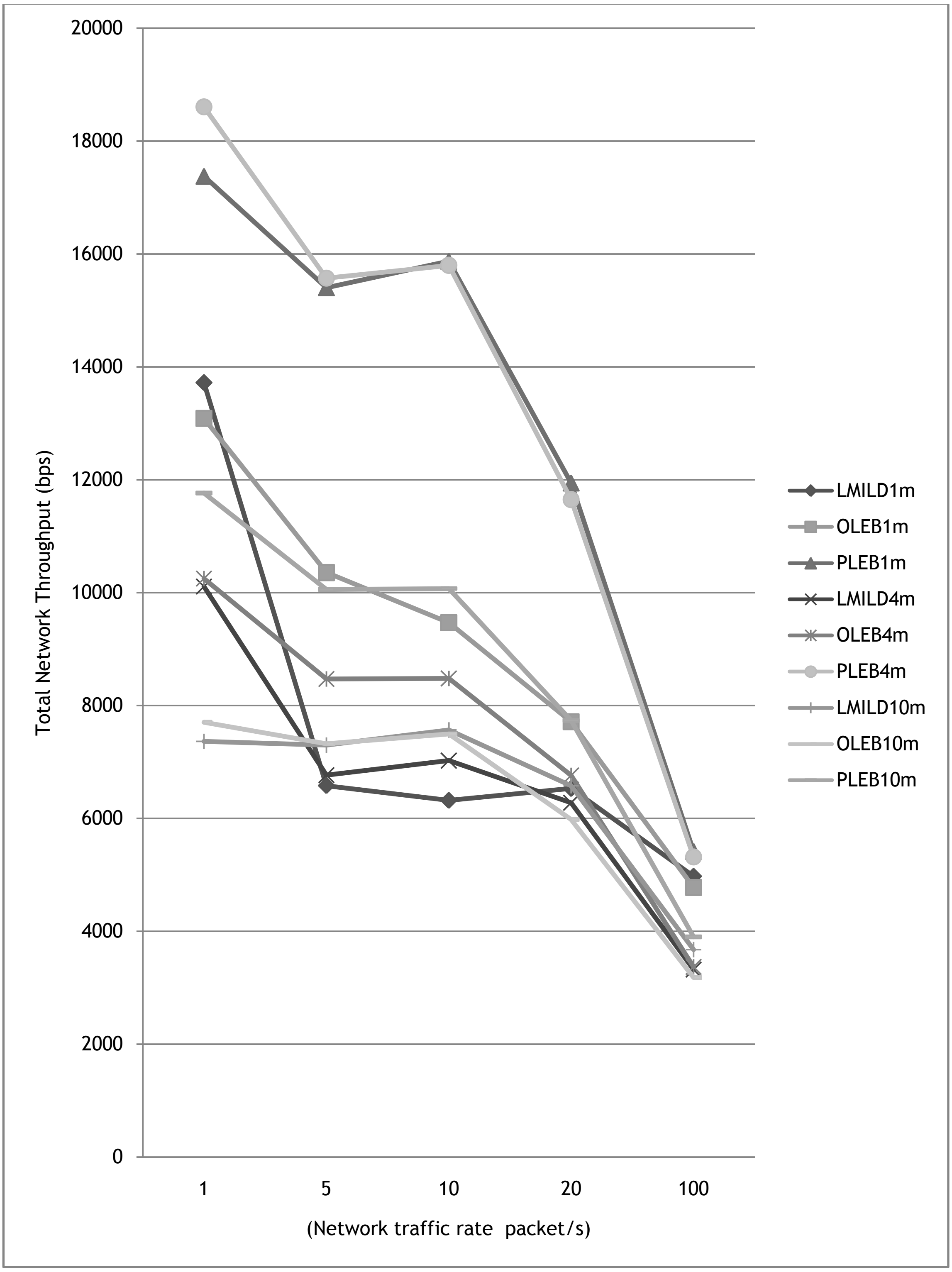


Figure 4.24: Summary of Network Throughput results for LMILD, OLEB and PLEB in a network of 50 Nodes.

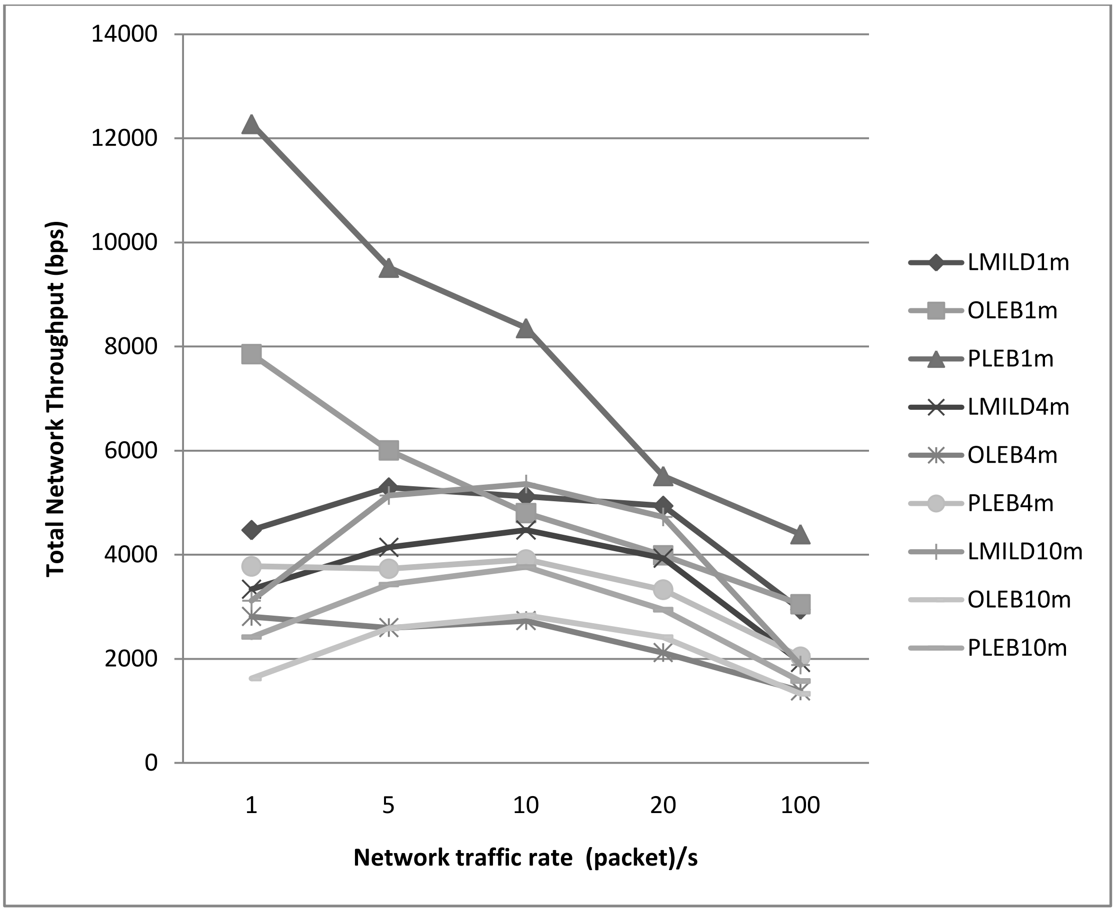


Figure 4.25: Summary of Network Throughput results for LMILD, OLEB and PLEB in a network of 100 Nodes.

Figure 4.25 presents throughout levels for 100 nodes. At this network size, lower mobility speeds allow higher throughput in general. Both of the proposed algorithms, OLEB and PLEB achieve higher throughput than LMILD at low speeds as well.

In the three graphs it can be seen that network throughput levels drop as the traffic rate increases. This is a general observation for all of the three algorithms tested.

Chapter 5. Behaviour Changing of Optimistic and Pessimistic Backoff algorithms

5.1. Introduction

Chapter 4 has introduced two proposed backoff algorithms, PLEB and OLEB. As described in the two algorithms, the exponential and linear increment behaviours are separated by a changing point. This point is the tune up factor of the two algorithms. Therefore, in this chapter, further investigation is performed to study this point in order to reach the best possible performance levels for the two algorithms. Moreover, the two algorithms use linear increments. The size of linear increments also is a tune up factor for the two algorithms. This chapter studies the linear increments to decide the best linear increment steps needed to reach highest performance levels.

The remainder of this chapter is organized as follows; Section 5.2 describes the simulation environment and the approach of studying changing points. Section 5.3 then introduces results and analysis. Finally, Section 5.4 concludes the chapter.

5.2. Simulation Environment and Approach

5.2.1. Parameters

The simulations conducted for this chapter have been based on a university campus ad hoc network. The simulation used a network area of 500 m × 500 m and network size of 500 students with identical nodes. Node mobility speeds have been set to 1 m/s, 2 m/s and 3 m/s to simulate the mobility speeds of walking students. The rest of simulation parameters have been left with the same values used in simulations of the previous chapters.

5.2.2. Approach

The point, at which the increment behaviour changes, is the factor that decides how close the algorithm is to either of the two extremes being the linear and the exponential increments. Since the size of contention window is the main subject in studying backoff algorithms, this chapter studies the changing point depending on the size of contention window rather than the number of increments. As described in the previous chapter, the maximum value of the contention window is 1024. This ceiling is used to stop the infinite increments of contention windows. In this chapter, the point of change is set at 25%, 50% and 75% of the maximum possible window size.

This chapter also studies the size of increment on CW size generated by the linear part of the two algorithms. The slope of the line that the backoff algorithm follows must be chosen in a way that insures increasing the CW and, at the same time, avoid reaching the exponential increment behaviour. The linear increment factors used in this chapter have been chosen to cover the range of increments between no increment at one end and the exponential at the other end. Therefore, the four linear increment factors used are approximately equivalent to increasing the CW size by 1.2, 1.4, 1.6 and 1.8.

5.3. Results and Analysis

In the first set of experiments, The Optimistic Backoff OLEB has been evaluated to study the effect of changing the point between the linear and the exponential increment behaviours. The three versions have been compared against the standard BEB that is used by IEEE 802.11 as shown in Figure 5.1.

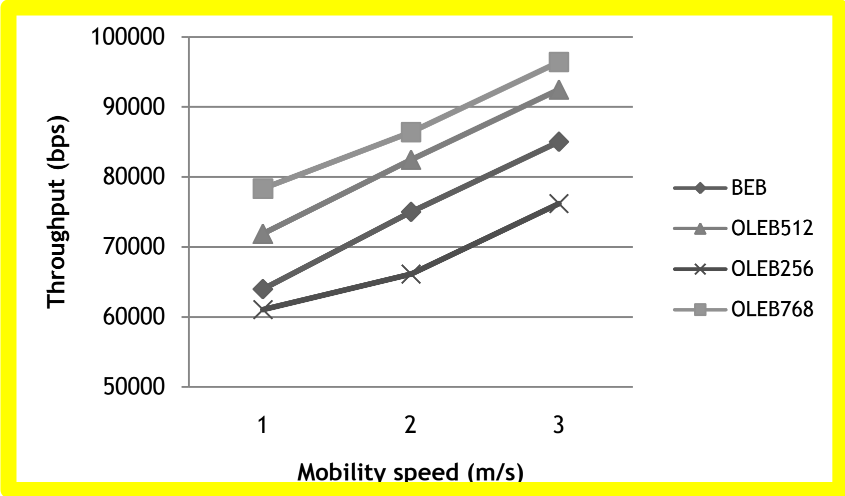


Figure 5.1: Network Throughput against Mobility Speed for different versions of OLEB

In Figure 5.1, OLEB has shown the lowest performance level in terms of Network Throughput when the changing point is set at 25% of maximum CW. In the network scenarios simulated in this chapter, the number of nodes is set to 300. Using this number of nodes, the collision rate is higher which leads to large number of nodes being put on backoff status. When the changing point is set at 25% of maximum CW, a small number of failures is followed by linear increment on CW size where, after that, the exponential increment is used. Forcing the small increments used by the linear behaviour leads to a large number of nodes adopting longer backoff timers when the increment is exponential. Therefore, the total network throughput is reduced by the extra backoff times that have resulted from changing a large number of nodes to the exponential increment behaviour on CW size.

The same set of experiments has been performed for the pessimistic backoff PLEB. The changing point has been moved to produce three different versions of

PLEB to be compared against BEB. The results shown in Figure 5.2 for network throughput against node mobility speed of the three versions of PLEB and BEB show that the lowest performance is achieved when the changing point is set to be at 75% of the maximum CW size. Working with the changing point being set to late stages increases the number of exponential increments of CW size. This leads to longer backoff times and, hence, wasting the network resources. On the other hand, the version of PLEB that uses 25% of maximum CW size as a changing point has the highest throughput levels. This supports the motivation behind integrating the linear increment into the proposed backoff algorithms.

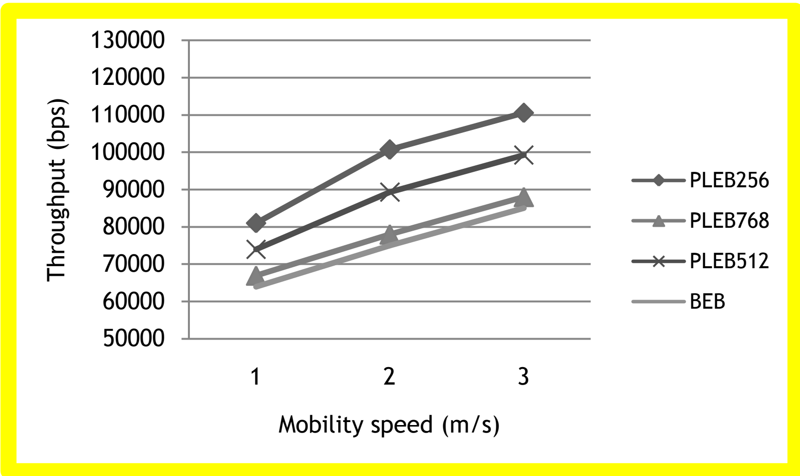


Figure 5.2: Network Throughput against Mobility Speed for different versions of PLEB

Increasing the number of times the linear increment is used forces backoff times to be chosen from relatively smaller CWs. This leads to better utilization of the limited network lifetime.

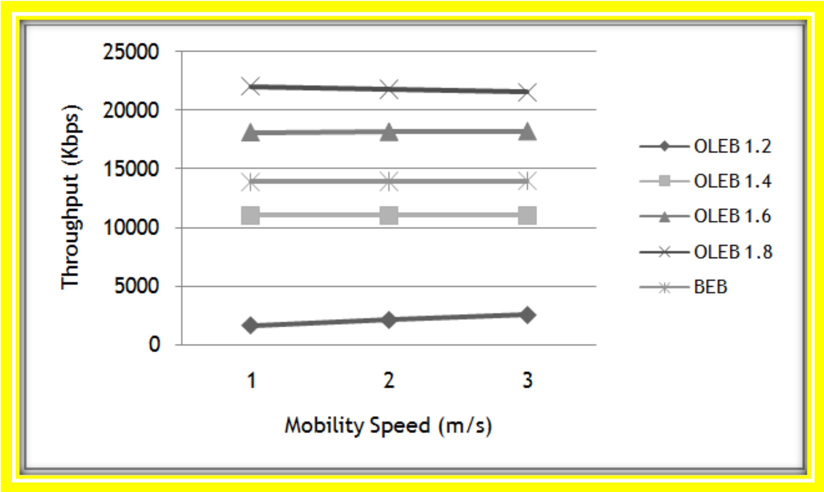


Figure 5.3: Network Throughput against Mobility Speed for different line versions of OLEB

Figure 5.3 represents Throughput results for the four linear increments implemented in OLEB. Results show that OLEB achieves the lowest throughput at the linear increment of 1.2 and the highest at the linear increment of 1.8. Since OLEB starts by using the linear increment first, using small increments combined with the large number of nodes simulated here does not allow adequate backoff time. Therefore, the longer backoff timers generated by larger increment is the suitable behaviour for this network size in terms of total network throughput.

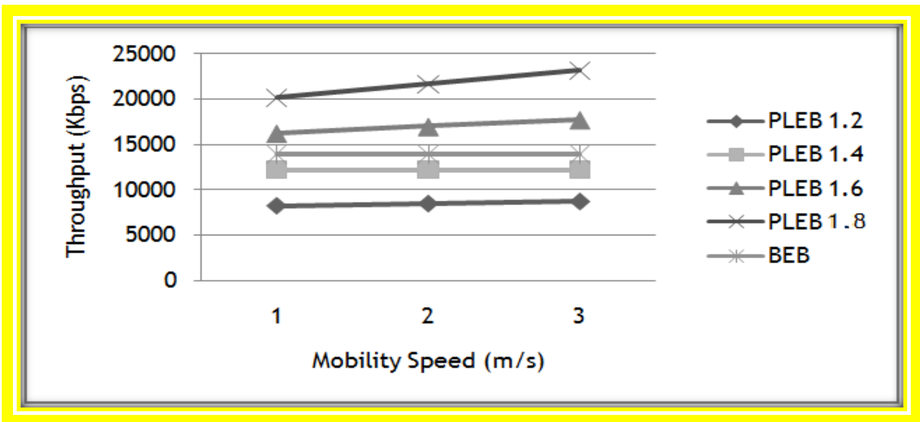


Figure 5.4: Network Throughput against Mobility Speed for different line versions of PLEB

The same linear increment factors have been also used with PLEB. The same observation that has been made on OLEB is valid for PLEB. Figure 5.4 presents network throughput results for different versions of PLEB that use different linear increment factors. Under the large network size, small linear increment factor does not allow backoff timers to be chosen from a CW that is wide enough which makes total network throughput higher for higher increment factor. The figure shows that the higher the increment factor is, the higher is the network delay.

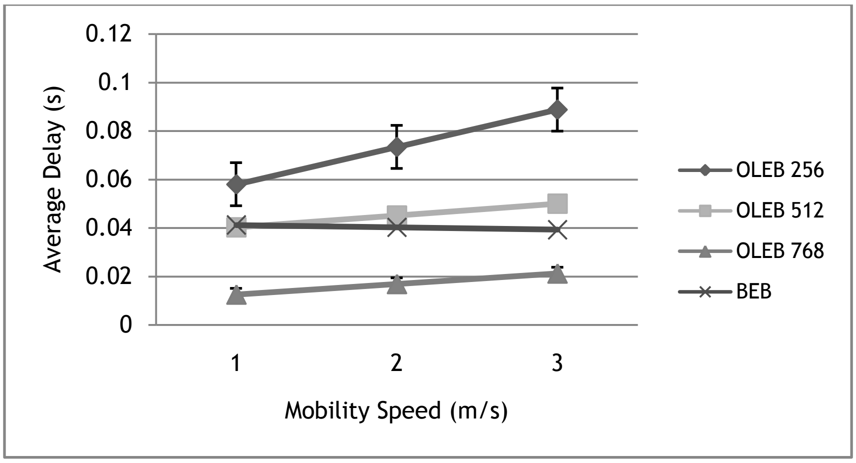


Figure 5.5: Average Packet Delay against Mobility Speed for different versions of OLEB

The two algorithms have also been evaluated in terms of average packet delay. Figure 5.5 presents average packet delay for different versions of OLEB. The linear increment used by OLEB produces less delay if allowed to work for longer time. This is provided by the version of OLEB that uses a turning point at 75% of the maximum CW size.

Figure 5.6 demonstrates the results of average packet delay for different versions of PLEB. The versions in this figure use different behavior changing point as discussed earlier. PLEB starts by using the exponential behavior. The version of PLEB that uses a turning point of 25% of maximum CW size allows the linear part to be used more than the exponential part. Therefore, the figure demonstrates the expected result which is the lowest average delay at the turning point set at 25% of the maximum CW size which is 256.

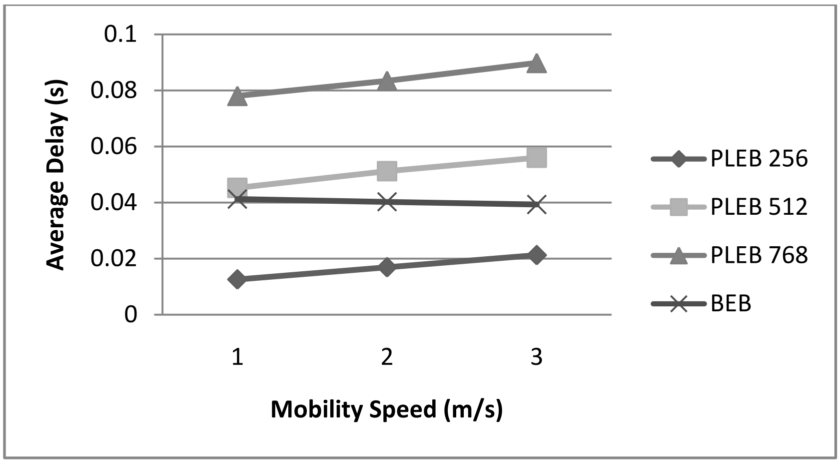


Figure 5.6: Average Packet Delay against Mobility Speed for different versions of PLEB

The different linear increment factors have been evaluated for average packet delay. Figure 5.7 demonstrates average delay for different versions of OLEB. The results report that the larger linear increment factor imposes longer average packet delays. The same result is drawn from evaluating PLEB with different linear increment factors. This can be seen in Figure 5.8.

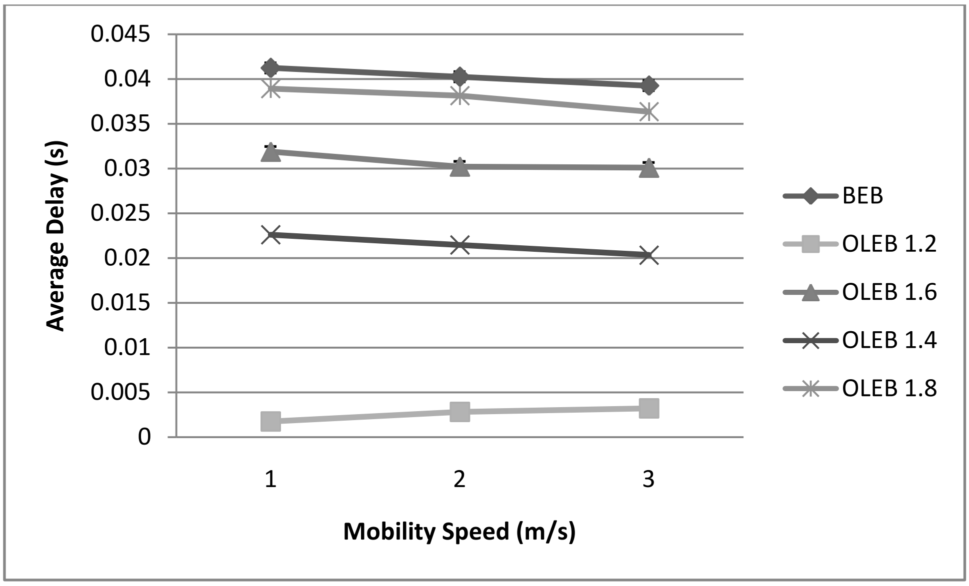


Figure 5.7: Average Packet Delay against Mobility Speed for different line versions of OLEB

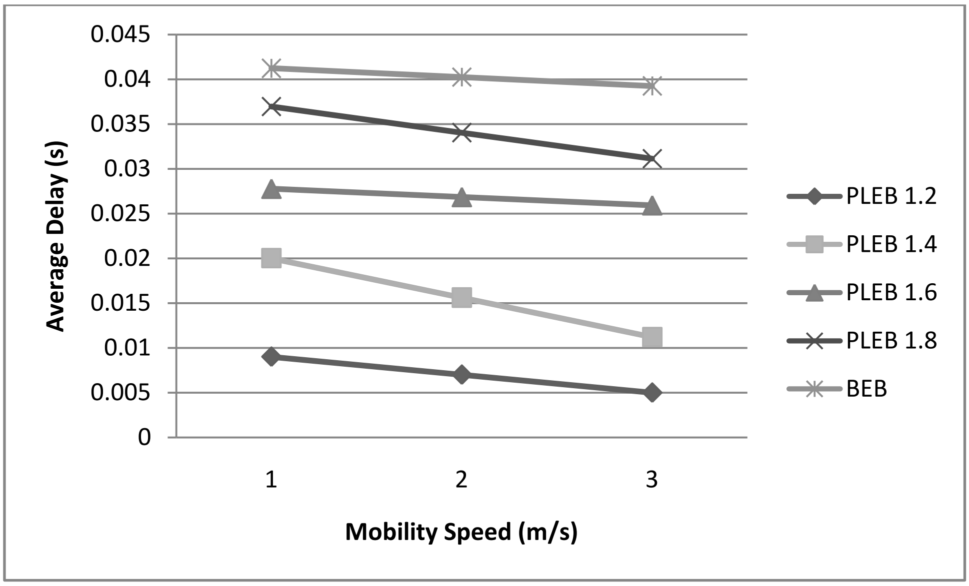


Figure 5.8: Average Packet Delay against Mobility Speed for different line versions of PLEB

5.4. Conclusions

In this chapter, the Pessimistic and the Optimistic Backoff algorithms introduced in Chapter 4 have been studied to analyse the changing point between the linear and the exponential increments on backoff Contention Window size. Results have shown that for OLEB, changing to the exponential increment behaviour at early stages does not allow the algorithm to achieve the best network throughput. For PLEB, a similar observation is made. The throughput results in this chapter suggest that the changing point should be chosen in a way that allows the linear increments to be used more than the exponential increments. The effect of behaviour changing point on average packet delay has also been studied. Results reported show that allowing the linear increments to be used more than the exponential increments reduces average packet delay.

The linear part of OLEB and PLEB has been studied also. Results show that there is a trade-off between throughput and delay when choosing the size of linear increments. Small linear increments achieve shorter average packet delay and larger linear increments provide better network throughput.

Chapter 6. Conclusions and Future Research Directions

6.1. Conclusions

This thesis has studied backoff mechanisms for MANETs. The first main objective of this work is to evaluate the performance of backoff in the presence of the new conditions introduced by MANETs. Such factors include network size, mobility speeds and traffic rates. Secondly, this work has aimed to gather enough evidence to help in developing backoff algorithms for MANETs. Moreover, this work has suggested new backoff mechanisms and has evaluated the performance of these algorithms under the mentioned factors.

The first part of this research has been presented in Chapter 3. In this chapter, simulations results have been presented to study the effects on network performance of changing both the increment and decrement behaviour of backoff algorithms. Changes applied to the algorithms modify increment behaviour upon a transmission failure and decrement behaviour after a successful transmission. Results from simulations have revealed that using different behaviours for increasing and decreasing contention window size, directly affects network performance metrics such as network throughput and average packet delay.

Changes applied to increment behaviours include both larger and smaller increments compared to the standard Binary Exponential Backoff (BEB). According to results, using large increments for contention windows sizes improves total network throughput. However, the large increments introduce extra packet delay. On the other hand, using smaller increment steps slightly improves the total network throughput and decreases packet delay as well. Although the improvements on network throughput are noticed even when the number of nodes and mobility speed are high, the improvement on network throughout is insignificant when taking the error margins of the simulations into account.

This work has addressed the increment behaviours. The second part of this research has been conducted and then presented in Chapter 4. In this part, two new backoff algorithms, referred to as the Pessimistic Linear Exponential Backoff (PLEB) and the Optimistic Linear Exponential Backoff (OLEB), have been introduced. PLEB is a combination of exponential and linear increment behaviours. In order to evaluate the performance of PLEB, this work has compared its performance against the existing backoff mechanism algorithm, Linear Multiplicative Increment Linear Decrement (LMILD). Simulation results have shown that PLEB achieves a lower network throughput, for a network of small size. This is not surprising since a network with such a small number of nodes is an ideal environment for LMILD.

When the number of nodes increases, PLEB provides a better network throughput than LMILD. A larger numbers of nodes (e.g. 50 nodes and over) makes it more difficult for LMILD to be able to update backoff timers after each collision due to the increased collision rate. Moreover, PLEB achieves better performance with low mobility speed. On the other hand, the performance advantage of PLEB is reduced with high mobility speed as this reduces the chance of a successful transmission after an exponentially-increased backoff timer expires. This is due to fact that when nodes move with a high speed there is high chance that a node leaves transmission range and thus breaks the link to the destination or the next hop in case the current destination is not the final destination of the packet.

PLEB has also been tested for average packet delay. Results have shown significant improvements in average packet delay when PLEB is implemented. This is valid for all network sizes at all traffic rates.

In the new OLEB algorithm, the exponential backoff is also combined with linear increment behaviour. The order of using the linear and the exponential increments is reversed in OLEB in comparison to PLEB. OLEB attempts to reduce redundant long backoff times by implementing less dramatic increments in the early backoff stages

The measurements of network throughput have revealed that for a small number of nodes, such as 10 nodes, the three algorithms addressed in Chapter 4 achieve same network throughputs for low traffic.

LMILD has higher throughput at high traffic rates and most node speeds. However, OLEB achieves higher network throughput than PLEB for high traffic. For a medium network size, PLEB has shown higher throughput than LMILD and OLEB. However, OLEB has higher network throughput than LMILD at low and medium mobility speeds. Finally, at a large network size of 100 nodes, PLEB has the best network throughput compared to OLEB and LMILD. At medium and high mobility speeds, LMILD achieves the best network throughput and OLEB has the lowest network throughput.

OLEB causes longer average packet delay compared to PLEB and LMILD at small network sizes. For a network size of medium size, OLEB produces shorter average packet delay at low traffic. However, at high traffic rates, OLEB has a higher delay than LMILD. In a network of large size, OLEB achieves shorter average packet delay than LMILD for medium and high mobility speeds.

In general, the results of this research indicate two main points. First, when designing the decrement behaviour of a backoff algorithm, larger decrement steps achieve better throughput compared to using smaller steps. For example, reducing CW size by 50%, results in significantly increasing the network throughputs when compared to linear decrements of CW size. Secondly, in most of the network scenarios used in this work, larger increments of CW at early stages of backoff sequence and then turning to smaller increments afterwards has proven to be the best increment behaviour when compared to smaller increment steps or implementing the small increments first and then turning to use larger increments.

Finally, this work has studied the effect of choosing the behaviour changing point between linear and exponential increments in OLEB and PLEB. Results have shown that increasing the number of times in which the linear increment is used increases network throughput. Moreover, using larger linear increments increase network throughput.

It is noteworthy to mention that the existing backoff algorithms have limitations in the sense that they all impose waiting time via increasing CW sizes. This is directly linked to the basic operation scheme of these backoff algorithms. Although the new proposed algorithms have improved network performance by increasing network throughout and decreasing average packet delay, these new algorithms use the same basics as the existing counterparts. Therefore, the increased CW sizes do add extra waiting time that might be wasted network idle time. Moreover, larger CW sizes can lead to long waiting times that end up in transmission drop especially in large network sizes. On the other hand, the information about other network nodes that is used by some existing algorithms, such as LMILD, limits the performance levels of these algorithms by the ability to obtain such information. However, the new proposed algorithms do not have this limitation since they do not use such information.

6.2. Future Research Directions

During the course of this research, many interesting issues have surfaced. The possible future directions of this work include addressing the following potential avenues.

- In this work, three network factors have been studied. However, other network factors also need to be considered. The most interesting among these is node transmission range. The network topology can be significantly affected by the node transmission range since it can lead to the network nodes being separated into groups.

- This research has used Constant Bit Rate (CBR) traffic. Future work should address using other traffic types such as Variable Bit Rate (VBR). Moreover, future work can possibly use traces of real traffic in order to achieve more credible measurements of network performance.

- This work has used simulation to evaluate the performance of backoff algorithms. This has also been the case with most of the performance-related work on MANETs [18, 74 and 17]. Another possible future direction of this work is to evaluate the algorithms using real practical MANETs in order to validate the findings of this research using real life data.

- The possible future directions of this work include studying the stability of the proposed algorithms since a part of results has revealed the incapability of the new algorithms to cope with increasing traffic rate. However, it should be mentioned here that addressing this point needs considerable amount of time and computation power since it involves injecting the network with extremely high traffic rates.

- All the simulations conducted by this work have assumed that nodes move according to the Random Waypoint model that has been widely used by previous researchers [7, 18, 17 and 74]. However, one possible direction of future work is to study backoff algorithms under different mobility models such as the Random Walk model [11]. Moreover, instead of using an individual node mobility model, a possible direction is to evaluate the algorithms under group mobility models that have been suggested in the literature [29]. Another possible future direction is to deploy real life data into simulations instead of relying totally on theoretically-generated data. Such real life data might include using mobility traces to build a realistic mobility model to use with the simulator.

- The set of possible values of network parameters used in this work has been limited due to time constraints and computation power. However, given the adequate time, one possible direction of this work is to evaluate the performance of backoff algorithms under a larger set of values for network parameters used.

- A final direction of this work might include developing an analytical model for backoff algorithms that relates the most critical factors together in order to build a sound validation tool for any future work on backoff mechanisms for MANETs.

Appendix A: List of Publications during the Course of This Work

[1] S. Manaseer and M. Masadeh, *"Pessimistic Backoff for Mobile Ad hoc Networks,"* The 4th International Conference on Information Technology ICIT, Jordan, 2009.

[2] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"On a Modified Backoff Algorithm for MAC Protocol in MANETs,"* International Journal of Business Data Communications and Networking, Vol. 5(1), pp. 60-73, 2009.

[3] M. Bani Yassein, S. Manaseer and A. Al-Turani, *"A Performance Comparison of Different Backoff Algorithms under Different Rebroadcast Probabilities for MANETs,"* 25th UK Performance Engineering Workshop, Leeds, UK, July, 2009.

[4] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"On a Modified Backoff Algorithm for MAC Protocol in MANETs,"* International Journal of Information Technology and Web Engineering pp 2(1), 34-47 Idea Group Publishing, 2007.

[5] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"Analytical Study of Backoff Algorithms for MAC Protocol in Mobile Ad Hoc Networks,"* 22nd UK Performance Engineering Workshop, Bournemouth University pp 89-94. UK, 2006.

[6] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"Thorough study of the logarithmic backoff algorithm for MAC protocols in MANETs,"* DCS Technical Report Series Dept of Computing Science, University of Glasgow, 2006.

[7] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"Fibonacci Backoff Algorithm for Mobile Ad Hoc Networks,"* DCS Technical Report Series Dept of Computing Science, University of Glasgow, 2006.

[8] S. Manaseer and M. Ould-Khaoua, *"Logarithmic Based Backoff Algorithm for MAC Protocol in MANETs,"* DCS Technical Report Series Dept of Computing Science, University of Glasgow, 2006.

[9] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *"Fibonacci Increment Backoff Algorithm for MAC Protocol in Mobile Ad Hoc Networks,"* Seventh Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, PGNET, Liverpool, UK pp 103-109, 2006.

[10] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *“On the Logarithmic Backoff Algorithm for MAC Protocol in MANETs,”* the 4th International Multiconference on Computer Science and Information Technology, Jordan pp 481-487, 2006.

[11] S. Manaseer, M. Ould-Khaoua and L. Mackenzie, *“A New Backoff Algorithm for MAC Protocol in MANETs,”* 21st Annual UK Performance Engineering Workshop pp 159-164, 2005.

Bibliography

[1] Liang-Jin Lin; Ortega, A., "Bit-rate control using piecewise approximated rate-distortion characteristics," Circuits and Systems for Video Technology, IEEE Transactions on , volume 8, issue 4, pp 446-459, 1998.

[2] “Mobile metropolitan ad hoc network (MobileMAN),” IST-2001-18113 Project, Funded by the EC FET-IST.

[3] A. Ali, L. Latiff and N. Fisal, “GPS-free indoor location tracking in mobile ad hoc network (MANET) using RSSI,” . Proceedings of RF and Microwave Conference RFM, pp 251-255, 2004.

[4] G. Anastasi, M. Conti, and E. Gregori, “IEEE 802.11 Ad Hoc Networks: Protocols, Performance and Open issues” In Mobile Ad Hoc Networking, S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, editors, New York: IEEE Press and John Wiley and Sons, Incorp

[5] C. Bettstetter, G. Resta, and Paolo Santi, “The node distribution

of the random waypoint mobility model for wireless ad hoc networks," IEEE Transactions on Mobile Computing, pp 257-269, 2003.

[6] C. Bettstetter, H. Hartenstein, and X. Pérez-Costa, "Stochastic properties of the random waypoint mobility model," ACM/Kluwer Wireless Networks: Special issue on Modeling and Analysis of Mobile Networks, volume 10, issue 5, 2004.

[7] C. Bettstetter and C. Wagner, "The spatial node distribution of the random waypoint mobility model," In Proceedings of German Workshop on Mobile Ad Hoc networks (WMAN), Ulm, Germany, 2002.

[8] V. Bharghavan, et al, "MACAW: a media access protocol for wireless LANs," in Proceedings of ACM SIGCOMM '94, pp. 212–225, 1994.

[9] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE Journal on Selected Areas in Communication , volume 18, issue 3, pp 535-547, 2000.

[10] E. Callaway, et al, “Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks,” IEEE Communications Magazine, volume 40, issue 8, pp 70-77, 2002.

[11] T. Camp, J. Boleng, and V. Davies, “A Survey of Mobility Models for Ad Hoc Network Research,” Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking Research, Trends and Applications, volume 2, issue 5, pp 483-502, 2

[12] P. Chatzimisios, A. Boucouvalas and V. Vitsas, "Effectiveness of RTS/CTS handshake in IEEE 802.11a Wireless LANs," Electronics Letters , volume 40, issue 14, pp 915-916, 2004.

[13] K. Chen and K. Nahrstedt, “Effective Location-Guided Tree Construction Algorithms for Small Group Multicast in MANET,” Proceedings of INFOCOM, pp. 1180–89, 2002.

[14] K. Chin, et al. “Implementation experience with MANET routing protocols,” SIGCOMM Comput. Commun, volume 32, issue 5, pp

49-59, 2002.

[15] I. Chlamtac, et al. “Mobile ad hoc networking: imperatives and challenges,” Ad Hoc Networks, volume 1, issue 1, pp 13-64, 2003.

[16] I. Chlamtac, et al, “Mobile ad hoc networking: imperatives and challenges,” Journal of Ad Hoc Networks. volume 1, issue 1, pp 13-64, 2003.

[17] J. Deng, P. Varshney, and Z. Haas, “A new backoff algorithm for the IEEE 802.11 distributed coordination function,” In Communication Networks and Distributed Systems Modeling and Simulation (CNDS '04), 2004.

[18] J. Deng and Z. J. Haas, “On Optimizing the Backoff Interval for Random Access Schemes,” IEEE Transactions on Communications, volume 51, issue 12, pp 2081-2090, 2003.

[19] J. DiMicco, “Mobile Ad Hoc Voting,” Proceedings of CHI Workshop on Mobile Ad-Hoc Collaboration, 2002.

[20] F. Dressler, "Self-Organization in Ad Hoc Networks: Overview and Classification," University of Erlangen, Department of Computer Science, Technical Report, 2006.

[21] M. Elizabeth, et al, "Transmission Range Effects on AODV Multicast Communication," Mobile Networks and Applications, volume 7, issue 6, pp 455-470, 2002.

[22] K. Fall and K. Varadhaa. Editors. "NS notes and Documentation." The V l N l Project UC Berkeley, 2002.

[23] Z. Fang and B. Bensaou, "A novel topology-blind fair medium access control for wireless LAN and ad hoc networks," IEEE International Conference on Communications, ICC '03 , volume 2, pp 1129-1134, 2003

[24] M. Frodigh, Per Johansson and Peter Larsson, "Wireless Ad Hoc Networking the Art of Networking without a Network," Ericsson Review No. 04, 2000.

[25] J. Gomez, et al, "Conserving Transmission Power in Wireless Ad Hoc Networks," Ninth International Conference on Network Protocols (ICNP'01), pp.0024, 2001.

[26] J. Goodman, et al., "Stability of Binary Exponential Backoff," Proceedings of the 17-th Annual ACM Symposium Theory of Computers, Providence, 1985.

[27] J. Goodman, et al, "Stability of binary exponential backoff," Journal of the ACM (JACM), volume 35, issue 3, pp 579-602, 1988.

[28] C. Guo, L. Zhong, and J. Rabaey, "Low-Power Distributed MAC for Ad Hoc Sensor Radio Networks," Proceedings of the Internet Performance Symposium (Globecom '01), 2001.

[29] X. Hong, "A Group Mobility Model for Ad Hoc Wireless Networks," Proceedings of MSWiM/99, pp 53-60, 1999.

[30] C. Hsu, Y. Tseng and J. Sheu, "An efficient reliable broadcasting protocol for wireless mobile ad hoc networks," Ad Hoc Networks,

volume 5, issue 3, pp 299-312, 2007.

[31] C. Hu, H. Kim, and J. Hou. “An Analysis of the Binary Exponential Backoff Algorithm in Distributed MAC Protocols”, Technical Report, University of Illinois – Urbana Champaign, 2005.

[32] J.-P. Hubaux, et al, “Towards self-organized mobile ad hoc networks: The Terminodes project,” IEEE Communications Magazine, 2001.

[33] E. Hyytiä, P. Lassila, and J. Virtamo, “Spatial Node Distribution of the Random Waypoint Mobility Model with Applications,” IEEE Transactions on Mobile Computing, 2006.

[34] E. Hyytiä, P. Lassila and J. Virtamo, “A Markovian Waypoint Mobility Model with Application to Hotspot Modeling,” In Proceedings of IEEE ICC, 2006.

[35] E. Hyytiä and J. Virtamo, “Random waypoint model in n-dimensional space,” Operations Research Letters, volume 33, issue 6, pp. 567-571, 2005.

[36] E. Hyytiä and J. Virtamo, “Random waypoint model in cellular networks,” Wireless Networks, 2006.

[37] IEEE, ANSI/IEEE standard 802.11, 1999 Edition (R2003), Part 11: “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”

[38] K. Jang, “A New Backoff Algorithm to Guarantee Quality of Service over IEEE 802.11 Wireless Local Area Networks,” LNCS 298, pp 371-376, 2004.

[39] J. Jang and M. Lee, "A Modified Backoff Algorithm to Improve Multiple Access Fairness for Optical Wireless LAN", ICACT, 2000.

[40] P. Joseph Macker and M. Scott Corson, “Mobile ad hoc networking and the IETF,” ACM SIGMOBILE Mobile Computing and Communications Review, volume 2, issue 4, 1998.

[41] A. Kanjanavapastit and B. Landfeldt, “A modified point coordination function in IEEE 802.11 Wireless LAN,” Proceedings

of IEEE ICON 03, pp 561-566, 2003.

[42] A. Kanjanavapastit and B. Landfeldt, “An analysis of a modified point coordination function in IEEE 802.11,” Proceedings on 14th IEEE Personal, Indoor and Mobile Radio Communications, volume 2, pp 1732-1736, 2003.

[43] W. Kiess and M. Mauve, “A survey on real-world implementations of mobile ad-hoc networks,” Ad Hoc Networks, volume 5, issue 3, pp 324-339, 2007.

[44] N. Komninos, D. Vergados and C. Douligeris, “Detecting unauthorized and compromised nodes in mobile ad hoc networks,” Ad Hoc Networks, volume 5, issue 3, pp 289-298, 2007.

[45] A. Koubâa, et al, “Improving the IEEE 802.15.4 slotted CSMA/CA MAC for time-critical events in wireless sensor networks,” In Proceedings of Workshop of Real-Time Networks, pp 270-277, 2006.

[46] S. Kumar, V. Raghavan and J. Deng, “Medium Access Control protocols for ad hoc wireless networks: A survey,” Ad Hoc Networks, volume 4, issue 3, pp 326-358, 2006.

[47] S. Kurkowski, T. Camp and M. Colagross, “Manet simulation studies: The current state and new simulation tools,” Technical Report MCS-05-02, The Colorado School of Mines, 2005.

[48] B. Kwak, N. Song and L. Miller, “Performance analysis of exponential backoff,” Networking, IEEE/ACM Transactions on , volume 13, issue 2, pp 343-355, 2005

[49] Q. Li, and D. Rus, “Sending messages to mobile users in disconnected ad-hoc wireless networks,” Proceedings of the 6th Annual international Conference on Mobile Computing and Networking MobiCom '00, ACM, pp 44-55, 200.

[50] J. Li, et al, “Performance Evaluation of Modified IEEE 802.11 MAC for Multi-Channel Multi-hop Ad Hoc Network,” Advanced Information Networking and Applications (AINA 2003) conference,

2003.

[51] Y. Li and A. Ephremides, “A joint scheduling, power control, and routing algorithm for ad hoc wireless networks,” Ad Hoc Networks, volume 5, issue 7, pp 959-973.2007

[52] W. Li and X. Chao, “Modelling and performance evaluation of a cellular mobile network,” IEEE Transactions on Networking, 12, 131–145. 2004.

[53] J. Liebeherr and G. Dong, “An overlay approach to data security in ad-hoc networks,” Ad Hoc Networks, volume 5, issue 7, pp 1055-1072, 2007.

[54] A. Lindgren, “Infrastructured Ad Hoc Networks,” Proc. 2002 Int'l Workshop on Ad Hoc Networking, Vancouver, August 2002.

[55] D. Malone, P. Clifford, and D. J. Leith, “MAC layer channel quality measurement in 802.11,” IEEE Communication, volume 11, issue 2, pp 143–145, 2007.

[56] S. McCanne and S. Floyd, “Network Simulator NS-2,” [online] Available at http://www.isi.edu/nsnam/ns/, 1997.

[57] R. Morris, et al., “Carnet: A scalable ad hoc wireless network system,” in 9th ACM SIGOPS European Workshop, 2000.

[58] T. Nandagopal, et al, “Achieving MAC layer fairness in wireless packet networks,” Proceedings of the 6th annual international conference on Mobile computing and networking, pp 87 – 98, 2000.

[59] A. Nasipuri and S. R. Das, “Multichannel CSMA with signal power-based channel selection for multihop wireless networks,” Proceedings of IEEE Fall Vehicular Technology Conference, 2000.

[60] W. Navidi and T. Camp, “Stationary distributions for the random waypoint mobility model,” IEEE Transactions on Mobile Computing, volume 3, issue 1, pp 99-108, 2004.

[61] S. Nesargi and R. Prakash, “MANETconf: configuration of hosts in a mobile ad hoc network,” INFOCOM 2002. Proceedings of IEEE

Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, volume 2, pp 1059-1068, 2002.

[62] M. Obaidat and D. Green, "An adaptive protocol model for IEEE 802.11 wireless LANs," Computer Communications, volume 27, issue 12 , pp 1131-1136, 2004.

[63] O. Ore, "Number Theory and its History," McGraw Hill, 1988.

[64] B.P. Crow, et al, "IEEE 802.11 Wireless Local Area Networks," Communications Magazine, IEEE, volume 35, issue 9, pp116-126, 1997.

[65] A. Pantazi and T. Antonakopoulos, "Equilibrium point analysis of the binary exponential backoff algorithm," Computer Communications, volume 24, issue 18, pp 1759-1768, 2001.

[66] J. Prokkola and T. Braysy, "A detailed study of a CDMA based approach to enhance ad hoc network performance," Ad Hoc Networks, volume 5, issue 7, pp 1149-1172, 2007.

[67] Finch, S. R. "The Golden Mean." Mathematical Constants. Cambridge University Press, pp. 5-12, 2003.

[68] R. Rajaraman, “Topology control and routing in ad hoc networks: a survey,” ACM SIGACT News, volume 33, issue 2, 2002

[69] R. Ramanathan and J. Redi, “A brief overview of ad hoc networks: challenges and directions,” IEEE Communications Magazine, volume: 40, issue 5, pp 20-22, 2002.

[70] G. Resta and P. Santi. “An analysis of the node spatial distribution of the random waypoint model for Ad Hoc networks,” In Proceedings of ACM Workshop on Principles of Mobile Computing (POMC), pp 44-50, 2002.

[71] E. M. Royer and C.-K. Toh, “A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks,” IEEE Communications, pp 46-55, 1999.

[72] C. Shen, C. Srisathapornphat and C. Jaikaeo, “An adaptive management architecture for ad hoc networks,” IEEE

Communication Magazine, volume 41, issue 2, pp 108-115, 2003.

[73] J. Shoch and J. Hupp, “Measured performance of an Ethernet local network,” Communications of the ACM, volume 23, issue 12, pp 711 – 721, 1980.

[74] N. Song, et al, "Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm," The 57th IEEE Semiannual Vehicular Technology Conference, volume 4, pp 2775-2778, 2003.

[75] P. Soni and A. Chockalingam, “Energy efficiency analysis of link layer backoff schemes on point-to-point Markov fading links,” The 11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2000 , volume 1, pp 416-420, 20

[76] I. Stojmenoic, “Handbook of wireless networks and mobile computing,” Wiley, New York 2002

[77] K. Sundaresan and R. Sivakumar, “A unified MAC layer framework

for ad-hoc networks with smart antennas," Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing, pp. 244 - 255, 2004.

[78] P. Tague and R. Poovendran, "Modelling adaptive node capture attacks in multi-hop wireless networks," Ad Hoc Networks, volume 5, issue 6, pp 801-814, 2007.

[79] M. Taifour, F. Nait-Abdesselam and D. Simplot-Ryl, "Neighbourhood backoff algorithm for optimizing bandwidth in single hop wireless ad-hoc networks" International Conference on Wireless Networks, Communications and Mobile Computing, volume: 1, pp 336- 3

[80] A. Tate and J. Dalton, "Using AI planning technology for army small unit operations," Proceedings of the Fifth International Conference on Planning and Scheduling Systems (AIPS), 2000.

[81] C-K. Toh, "Ad hoc mobile wireless networks, protocols and systems," Prentice-Hall, New York 2002.

[82] C. Tschudin, et al, “Lessons from experimental MANET research,” Ad Hoc Networking for Pervasive Systems, volume 3, issue 2, pp 221-233, 2005.

[83] Y. Tseng and T. Hsieh, “Fully power-aware and location-aware protocols for wireless multi-hop ad hoc networks,” Proceedings of the Eleventh International Conference on Computer Communications and Networks, pp 608-613, 2002.

[84] R. Wattenhofer, et al, “Distributed topology control for power efficient operation in multihop wireless ad hoc networks,” Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, volume: 3, pp 1388-1397, 2001.

[85] K. Weniger and M. Zitterbart, "Mobile ad hoc networks - current approaches and future directions," Network, IEEE , volume 18, issue 4, pp 6-11, 2004.

[86] K. Whitehouse, et al,” Exploiting the capture effect for collision detection and recovery,” IEEE Workshop on Embedded

Networked Sensors (EmNetS-II), 2005

[87] J. Wu and F. Dai, “A distributed formation of a virtual backbone in MANETs using adjustable transmission ranges," Proceedings of the 24th International Conference on Distributed Computing Systems, pp. 372-379, 2004.

[88] J. Wu and I. Stojmenovic, "Ad hoc networks," Computer, volume 37, issue 2, pp 29-31, 2004.

[89] H. Wu and X. Jia, “QoS multicast routing by using multiple paths/trees in wireless ad hoc networks,” Ad Hoc Networks, volume 5, issue 5, pp 600-612, 2007.

[90] H. Xia, Z. Zeng and W. Ding, “Adaptive Backoff Algorithm based on Network Congestion in Multi-hop Wireless Ad hoc Networks,” SNPD (3), pp 319-324, 2007.

[91] B. Yan, H. Gharavi, “Power Control in Multihop CSMA,” Proceedings of the 2006 International Symposium on World of Wireless, Mobile and Multimedia Networks, pp 432-434, 2006.

[92] Q. Zeng and D. P. Agrawal, "Handoff in wireless mobile networks," Handbook of wireless networks and mobile computing. New York: Wiley. 2002.

[93] H. Zhai, et al, "Medium access control in mobile ad hoc networks: challenges and solutions," Wireless Communications and Mobile Computing, volume 6, issue 2, pp 151-170, 2006.

[94] Y. Zhang, et al, "A secure incentive protocol for mobile ad hoc networks," Wireless Networks, volume 13, issue 5, pp 569-582, 2007.

[95] M. Bani Yassein, et al, "New Adaptive Counter Based Broadcast Using Neighborhood Information in MANETS," Proceeding of the 23rd IEEE International Parallel & Distributed Processing Symposium, Rome, Italy, May 25-29, 2009.

[96] S. Papanastasiou, "Investigating TCP Performance in Mobile Ad Hoc Networks," doctoral dissertation, University of Glasgow, Glasgow, UK, 2006.

[97] T. Camp, et al, "A survey of mobility models for ad hoc network research," Wireless Communications and Mobile Computing, 2002.

[98] N. Meghanathan, "Impact of simulation time and network shape on the hop count and stability of routes for mobile ad hoc networks," Proceedings of the 47th Annual Southeast Regional Conference, 2009.

[99] S. Kurkowski, et al, "Constructing MANET Simulation Scenarios That Meet Standards," IEEE International Conference on Mobile Ad hoc and Sensor Systems, 2007.

[100] IEEE Standard 802.11-1997 Information technology, Telecommunications and information exchange between systems, Local and metropolitan area networks- Specific requirements Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification, 1997.