

MAKERERE

P.O. Box 7062 Kampala Uganda
<https://www.cs.mak.ac.ug>



UNIVERSITY

Telephone: 256-414-534560/1-9
E-mail: cs@cis.mak.ac.ug

COLLEGE OF COMPUTING AND INFORMATION SCIENCES



Priority Scheduling Schemes in Mobile Ad-Hoc Networks

BY

Mukakanya Abel Muwumba
Reg No: 2019/HD05/31234U
Msc. DCSE, Bsc. IT (Mak)

A thesis submitted to the Directorate of Research and Graduate Training in fulfillment of the requirement for the award of Doctor of Philosophy in Computer Science of Makerere University

26th August 2024

Declaration

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution..

Signature..... *J. Abel* Date *26.05.2024*

Mukakanya Abel Muwumba

PhD Candidate, Department of Computer Science

School of Computing and Informatics Technology,

Makerere University,

Kampala.

Approval

This report has been submitted with the approval of the following supervisors.

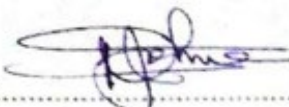
Dr. John Ngubiri

Department of Computer Science

School of Computing and Informatics Technology,

Makerere University,

Kampala.

Signature.......... Date 26/8/24


Dr. Odongo Steven Eyobu

Department of Networks

School of Computing and Informatics Technology,

Makerere University,

Kampala.

Signature.......... Date 2024.08.26

Date of PhD defense: 28th March 2024.

Jury Members

1. Prof. BAGULA, Bigomokero Antoine, University of Western Cape, Cape Town, South Africa.
2. Dr. Kibombo Joseph Balikuddembe Makerere University.
3. Dr. Abubaker Matovu Waswa Makerere University.
4. Dr. Tonny Eddie Bulega Makerere University.
5. Dr. Julianne Sansa-Otim Makerere Univeristy.
6. Dr. Rose Nakibuule Makerere Univeristy.
7. Dr. Marriette Atuhuriire Katarahweire Makerere Univeristy.

Dedication

This project is dedicated to my wife Hardy Mary Najjuko, whose love and encouragement made this work possible, and in loving memory of my parents (Papa Mzee Dison Mukakanya and Mama Beatrice Nakandha Mukakanya), who always stood by me in everything I attempted.

Acknowledgements

This PhD work has been made possible because of the contribution of a number of people whose effort cannot be overlooked but deserves to be acknowledged. In a special way I would like to extend my sincere appreciation to my thesis supervisors Dr. John Ngubiri and Dr. Odongo Steven Eyobu for their patience, encouragement and professional guidance, which made this work complete. You have not only mentored but also modelled me into a strong academician and writer. My deep gratitude to the Doctoral Committee members Dr. Micheal Kizito, Dr. Daudi Jjingo and Associate Professor, Engineer Bainomugisha; your support and encouragement has guided me along the path to the apex of my academic career.

This work also benefited from the support and cooperation of administrators from the College of Computing and Information Sciences most especially Liliane Maureen Ayebare and Pearl Arinda. To the management and staff of Uganda Business and Technical Examinations Board (UBTEB) most especially CPA Onesmus Oyesigye (Executive Secretary), Dr. Wilfred Karukuza Nahamya (Deputy Executive Secretary Examinations Management), Mr. Kawanguzi Geoffrey (Deputy Executive Secretary Finance Human Resource and Administration) thank you for standing in for and with me during the course of this study.

To my beloved children Mukisa Abel Kabaale, Mukakanya Elijah Mwesigwa, Lugwire Eric Mugabi and Muwumba Elisha Tsubira a lot of your time (day, nights and holidays) I used to immerse in deep reading and writing this PhD thesis cannot go unrecognized. Many thanks to my dear wife Najjuko Hardy Mary whose love and friendship kept this PhD work on-going to completion.

Finally, I am highly indebted to Government of Uganda for the PhD grant provided through Makerere University Research and Innovations Fund (grant number: MAK-RIF Round 4, 2022/23). I am also grateful for the funding support from UBTEB in preparation of this manuscript. Without your financial support, undertaking PhD studies would have remained a delusion.

Acronyms

AOMDV: Ad-hoc On-Demand Multipath Distance Vector.

BS: Base Station.

BP: Bounded Pareto.

CaSMA: Channel Aware Scheduling.

CBWFQ: Class Based Weighted Fair Queuing.

CoV: Coefficient of Variation.

EDF: Earliest Deadline First.

EEDF-I: Enhanced Earliest Deadline First-1.

EEDF-II: Enhanced Earliest Deadline First-II.

ELLQ: Extended Low Latency Queuing algorithm.

EWRR: Existing Weighted Round Robin.

IID: Independent and Identically Distributed.

GPS: General Processor Sharing.

IWRR: Improved Weighted Round Robin.

MANET: Mobile Ad-Hoc Network.

MMPP: Markov Modulated Poisson Process.

MS: Mobile Station.

LAS: Least Attained Service.

LCFS: Last Come First Served.

LLQ: Low Latency Queueing.

PALM: Power-Aware Link Maintenance.

PDA: Personal Digital Assistant.

PQ: Priority Queuing.

WRR: Weighted Round Robin.

WFQ: Weighted Fair Queuing.

SRPT: Shortest Remaining Processing Time.

Technical Terms

- Adapted is to make (something) suitable for a new use or purpose by modifying it.
- Adopted means to take on, or accept as ones own.
- Average waiting time is the time between when the packet arrives and when the packet first receives service
- Coefficient of variability is the ratio of standard deviation to the mean of the distribution.
- Conditional mean response time, $T(x)$ is mean response time and is a function of job size under a given policy.
- Conditional mean slowdown, $S(x)$ are defined as $T(x)/x$.
- Latency is the sum of transmission delay and propagation delay.
- Response time refers to the total time a packet spends in the system.
- Scheduling scheme: Is an algorithm with a set rules that determine the task to be executed at a particular moment.
- Slowdown: is the ratio of the response time of a packet to the size of that packet.
- Transmission delay is time needed by one bit to travel through a network.
- Propagation delay is time it will take a signal to pass through the transmission delay.

Table of Contents

Declaration	ii
Approval	iii
Dedication	iv
Acknowledgements	v
List of Acronyms	vi
Technical Terms	vii
Abstract	36
1 Introduction	37
1.1 Background	37
1.2 MANETs challenges	40
1.3 Statement of the problem	41
1.4 Objectives	41
1.4.1 General objective	41
1.4.2 Specific objectives	41
1.5 Research questions	41
1.6 Scope	42
1.7 Significance and Relevance of the work	42
1.8 Thesis contributions	42
1.9 Publications	42
1.10 Thesis outline	43
2 Literature Review	43
2.1 Classification of scheduling algorithms in MANETs	44
2.1.1 Channel Aware Scheduling Algorithms	44
2.1.2 Packet scheduling algorithms	45
2.2 Desired properties of packet scheduling algorithms in MANETs	47
2.3 Challenges of scheduling in wireless network	47
2.4 Applications of scheduling in communication network infrastructure	48
2.5 Related work on EDF	49
2.5.1 General structure of EDF	49
2.5.2 Advantages and limitations of EDF scheduling	49
2.5.3 The earlier EDF analytical models	50
2.5.4 The recent EDF analytical models	50
2.6 Related work on LLQ	51
2.6.1 General structure of LLQ	51
2.6.2 Common variations	51
2.6.3 The earlier LLQ schemes	51
2.6.4 Recent improvements of the LLQ schemes	52
2.7 Related work on WRR	53
2.7.1 How WRR works	53

2.7.2 Strengths and weakness of WRR	53
2.7.3 The earlier WRR schemes	53
2.8 Limitations in accessing literature	54
2.9 Research gaps	54
3 Research Methodology	55
3.1 Queuing theory	55
3.1.1 Introduction	55
3.1.2 Kendalls notation	55
3.2 The MANET queuing model	55
3.2.1 The service distributions	56
3.2.2 The model input	56
3.2.3 The model output and performance metrics	56
3.3 Methods and tools	57
3.3.1 Experiments	57
3.3.2 Analytical modelling	57
3.3.3 Simulations	57
3.4 Conclusion	58
4 EDF Scheduling in MANETs	58
4.1 Introduction to EDF schemes	59
4.1.1 The EDF algorithm	59
4.1.2 EDF timing requirements	59
4.2 Assumptions	60
4.3 The design of EDF schemes	60
4.3.1 The generic EDF algorithm	60
4.3.2 Weakness of the generic EDF Abhaya algorithm	62
4.3.3 The performance metrics	62
4.4 Adaption of the EDF to MANETs	62
4.5 The EEDF-II model	65
4.5.1 Modifications	65
4.5.2 Expressions for average delay	65
4.6 Implementation and Results	67
4.6.1 Theoretical evaluation	67
4.6.2 Performance of the EEDF-I model	70
4.6.3 Performance of the EEDF-I and EEDF-II models	71
4.6.4 Discussions of the results	71
4.7 Conclusion	72
5 LLQ Scheduling in MANETs	72
5.1 Introduction	72
5.2 The mathematical notations and expressions	72
5.3 Assumptions	73
5.4 Existing LLQ model	73

5.5 The adopted LLQ algorithm	74
5.6 The proposed LLQ algorithm	75
5.7 The performance evaluation	77
5.7.1 Experimental set-up	77
5.7.2 Results of the adopted LLQ algorithm	77
5.7.3 Performance under exponential distribution	79
5.7.4 Performance under BP distribution	81
5.8 Conclusion	82
6 Extended LLQ Scheduling in MANETs	83
6.1 Introduction	83
6.2 The ELLQ model	83
6.3 Results and Discussions	85
6.3.1 Analysis under exponentially distributed workloads	85
6.3.2 Analysis under heavy tailed workloads	91
6.4 Conclusion	96
7 WRR Scheduling in MANETs	96
7.1 The generic WRR scheduling algorithm	96
7.2 Adapting the WRR algorithm into MANETs	97
7.3 The improved WRR algorithm	100
7.4 Numerical results	102
7.4.1 Experimental set-up and analysis software	102
7.4.2 Evaluation of the EWRR algorithm	103
7.4.3 Evaluation of EWRR & IWRR under exponential distribution	105
7.4.4 Evaluation of EWRR & IWRR under heavy tailed distribution	107
7.5 Conclusion	109
8 Conclusion and Future Work	109
8.1 Summary of findings	109
8.1.1 EDF algorithms	109
8.1.2 LLQ algorithms	109
8.1.3 WRR algorithms	110
8.2 Areas of further research	110
References	110

Priority Scheduling Schemes in Mobile Ad-Hoc Networks

Type: PhD Thesis

Received: September 14, 2024

Published: November 01, 2024

Citation:

Mukakanya Abel Muwumba.
"Priority Scheduling Schemes
in Mobile Ad-Hoc Networks".
PriMera Scientific Engineering
5.5 (2024): 36-116.

Copyright:

© 2024 Mukakanya Abel Muwumba. This is an open-access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mukakanya Abel Muwumba*

Department of Computer Science, Makerere University, Kampala, Uganda

***Corresponding Author:** Mukakanya Abel Muwumba, Department of Computer Science, Makerere University, Kampala, Uganda.

Abstract

Delay is a major Quality of Service (QoS) metric in mission critical applications. Some applications run on Mobile Ad-Hoc Network (MANET) set ups which comes with transmission challenges arising from the size of traffic packets and environmental conditions. These challenges cause transmission delays, packet loss and hence a degraded network performance. This study investigated the performance of: Earliest Deadline First (EDF); Low Latency Queueing (LLQ) and Weighted Round Robin (WRR) scheduling algorithms in MANETs.

Firstly, the study investigated the Abhaya pre-emptive EDF scheduler. The study improved and adopted EDF algorithm to the MANETs environment, and formulated the Enhanced Earliest Deadline First-I and II (EEDF-I & EEDF-II) algorithms respectively. The numerical results showed that the EEDF-II model shortened the waiting times of packets of the different queues at various system loads compared with the EEDF-I model.

Secondly, the study adopted and improved the existing model to LLQ algorithm in the M/G/1 queue system. The numerical results revealed that the proposed algorithm performed better than the adopted in transmitting video packets. The study extended further the proposed LLQ algorithm to formulate the Extended Low Latency Queueing algorithm (ELLQ). The numerical results revealed that the video packets experienced the least conditional mean response time/slowdown; followed by voice packets and lastly text packets.

Thirdly, the study enhanced and studied the Existing (EWRR) service strategy; and then proposed an Improved (IWRR) model in the M/G/1 queue system under varying workloads distributions. The numerical results showed that video packets performed poorly compared to voice packets in the EWRR algorithm.

In conclusion, we studied three algorithms namely: EDF, LLQ & WRR, and proposed three novel variants i.e., EEDF-II, ELLQ plus IWRR for MANETs.

Chapter 1 Introduction

This Chapter presents a brief overview of the thesis. It introduces the problem and its background, the expected tasks, scope as well as the outcome of the research, Section 1.1, discusses the background and motivation of this study. Section 1.2 presents the MANET challenges. The statement of the problem is presented in Section 1.3. The objectives and research questions of the study are stated in Sections 1.4 and 1.5 respectively; Section 1.6 defines the scope of the study; The benefits of this work are provided in Section 1.7; Section 1.8, presents the research contributions; followed by Section 1.9 which presents the publications; and lastly, Section 1.10, provides the thesis outline.

1.1 Background

Wireless communication has been around for over a century and has within the recent past positioned its self as the most popular mode of communication in people's daily lives. The traditional wireless networks are usually infrastructured with fixed Base-Stations (BS). Installing such an infrastructure is often either too expensive or technically impossible for some remote localities [1]. Therefore, Mobile Ad Hoc Networks (MANETs) evolved out of the need by researchers/practitioners to eliminate the use of infrastructures in wireless communication. This concept of MANETs dates back to the Defense Advanced Research Projects Agency (DARPA) packet radio network program in the 1970's [2]. Communication networks first came into existence in 1876 with the invention of telephone by Alexander Graham Bell. Broadly, communication networks are classified into two categories: wired and wireless. Wired networks make use of cables or wires as the main transmission media whereas wireless networks use free space as transmission media. Some of the advantages of wired networks are: (i) Stability and reliability-When configured and utilised properly, wired networks are capable of providing unparalleled reliability. Furthermore, if the hubs, switches, and cables are connected, you can have a reliable network at your premises. Wired connections experience minimal interference from other network connections in the vicinity. This is why you will find that wired networking-based infrastructures are generally more stable. (ii) Faster speeds and high Connectivity Wired networks allow for high-speed data communication than wireless networks. With the evolution in technology, the speeds have kept on improving since the use of Gigabit routers became a common practice. In addition to that, a wired network has a limited set of users connecting to it at any time, so its rarely bogged down by unexpected traffic delivering nearly constant high speeds at all times. The major disadvantage of these wired networks is they are very inflexible when it comes to mobility. To be able to access a wired network at a different location, there is no other option but to run extra cables and install switches at that location which may or may not always be inconvenient, depending on how much mobility your business or staff demands. Also, wired networks are considered to be expensive because of the high setup costs of infrastructure and maintenance costs. Since wired networks are expensive, wireless networks have been sought to be the most suitable alternative choice. Wireless networks are popular because they have characteristics such as mobility, reachability, simplicity, maintainability, roaming services and new services.

Wireless networks have internal classifications depending on transmission technologies, data carried and services offered. There are two main types of wireless networks namely: Infrastructured and Infrastructure-less [3]. In Infrastructured wireless networks, mobile nodes can move while communicating, the BS are fixed and as the node goes out of the range of a BS, it gets into the range of another BS. Infrastructure-less or Ad-Hoc wireless networks, the mobile node can move while communicating, there are no fixed base stations and the nodes in the network act as routers.

Wireless networks have evolved into different generations. The First Generation (1G), commenced in the early 1980s. The 1G were analog but used point-to-point digital microwave for the backhaul links connected to or close to the BS. They used time division multiplexed (TDM) technology to transmit data. These links operated initially in or close to the 2 GHz band but migrated slowly to the 18 to 23 GHz range and only voice is supported [4].

This was followed by the Second Generation (2G) systems, that started in the early 1990s. Just, like 1G, 2G systems primary aim was to provide voice services, but they utilized digital modulation, permitting a higher voice capacity and support of low-rate data applications. The 2G systems operating frequencies ranged from 18 to 38 GHz, higher data rate was possible through fiber providing backhaul

connection. The Third Generation (3G) systems, commenced service in the early 2000s. These systems were aimed at delivering wide range of services, including telephony, higher speed data than available with 2G, video, paging, and messaging. The 3G systems utilized Code Division Multiple Access (CDMA) technology to provide much higher data rates over both circuit-switched and packet switched bearers. Through innovations there was evolution in technology to Universal Mobile Telecommunications System (UMTS) whose continued development led to extremely high data rates [4].

The Fourth Generation (4G) systems, started in early 2010s. The 4G backhaul networks are all Internet Protocol (IP)/Ethernet-based. The connection accomplished via Ethernet links e.g., Fast Ethernet (100 Mb/s) or Gigabit Ethernet (1Gb/s) [4]. In the recent past, the wireless transport evolution has been pushing for the next-generation cellular systems, commonly referred to as Fifth Generation (5G) commenced service in 2019. 5G backhaul networks are all IP/Ethernet-based, but now, the transport network may also comprise IP/Ethernet-based midhaul and fronthaul [4].

The 5G systems just like the earlier generations are confronted with limitations of failure to provide highly reliable connectivity in scenarios when the wireless transport infrastructure has been destroyed by natural disasters like lightning and earth quakes, man-made catastrophes such as war, riots [4]. Therefore, MANETs remain one of the viable solutions that is able to provide connectivity even when wireless infrastructure is inexistant or disabled. MANETs are autonomously self-organized infrastructure less networks without fixed topology and each node acts as both a router and host at the same time [5]. The basic structure of MANETs constitutes mobile devices such as laptops and Personal Digital Assistants (PDA) that are connected to transfer packets from source to destination mobile nodes and there is an intermediary node between transmitting and receiving node acting as a router [6].

Figure 1.1 shows MANET technology with no fixed BS and every node must cooperate in forwarding packets in the network [7].

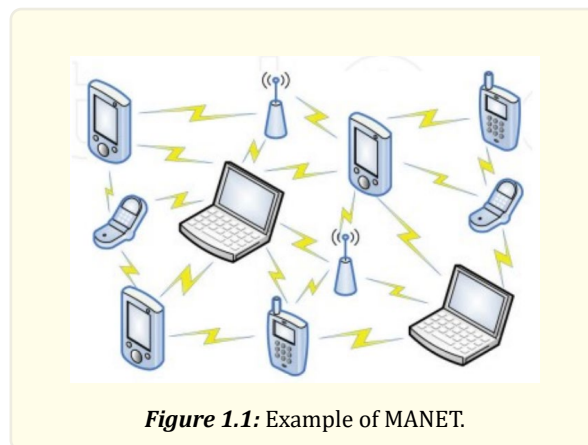


Figure 1.1: Example of MANET.

MANETs have unique characteristics that make them popular everywhere at any time.

Infrastructure-less nature

MANETs are formed based on the collaboration between independent peer-to-peer nodes to communicate with other nodes for a particular purpose [8].

Easy and rapid deployment

MANETs technology comes with several advantages over wireless networks, including ease of deployment, speed of deployment, and decreased dependence on a fixed infrastructure. MANETs are becoming popular because of their ability to provide an instant network formation without the presence of fixed BS and system administration [9].

Bandwidth constraints and variable link capacity

The MANET nodes are connected by wireless links that have much smaller bandwidth than those with wires [8].

Multi-hop communication

A message from source node to destination is transmitted via multiple nodes because of limited transmission range [10]. Within the MANETs every node act as a router and forwards packets from other nodes in order to facilitate multi-hop routing [11].

Constrained resources (light-weight terminals)

A large proportion of the MANET nodes are small hand-held devices characterized with limited power (battery operated), processing capabilities and storage capacities. Some of the notable examples range from laptops, smart phones and PDA.

Short range connectivity

MANET depends on Radio Frequency (RF) or Infrared (IR) technology for connectivity, both of which are generally used for short range communications.

The following are the application areas that benefit from MANETs [12]:

- **Military tactical operations**
A communication network that is largely dependent on fixed infrastructure is highly risky for military tactical operations, because it is susceptible target in hostile environments. Elimination of fixed infrastructure makes MANETs the most ideal choice for military tactical operations.
- **Search and rescue missions**
Quite often search and rescue missions occur in remote locations with no communication infrastructure, like at the top of a mountain, in the middle of a forest or inside a cave, inside a tunnel. MANETs are easy to use communication systems for such environments.
- **Disaster relief**
MANETs provide communication in problematic environments such as locations where existing infrastructure has been destroyed or left inoperable due to thunder and rain storm, floods, war, riots.
- **Law enforcement**
Law enforcement operations can be extended to include locations with no communication infrastructure. MANET systems support fast and secure communication in such environments.
- **Commercial use**
MANETs are used to aid data exchange between people and applications in large meetings and conventions.

The following are the strengths associated with MANETs:

- **Self configuration**
MANETs are self-configuring and do not depend on a particular node as a central controller and dynamically adjust as nodes join or leave the network due to free node mobility. The network is constituted from simple battery-operated handheld devices relaying data in multi-hop. Therefore, the strength of MANETs include: low installation costs, easy deployment, accessibility.
- **Flexibility**
MANETs are both flexible and robust, hence, due to their flexibility and robustness, they can be quickly deployed for the support of many applications. In addition, MANETs bypass the need for a router by connecting the devices directly to each other using their wireless network adapters which is an added advantage on saving for hardware acquisition.
- **Robustness:** Also, failure of a node(s) does not necessarily result into total blackout in transmission or exchange of data.

Despite the strengths, there are inherent weaknesses exhibited by MANETs for instance

- The ever rapidly changing MANETs topology can result into frequent link failure.
- MANETs are unable to provide faster connection speeds for data and therefore require innovative scheduling techniques to increase the amount of bandwidth per user.

1.2 MANETs challenges

Despite the attractive applications and different characteristics of MANETs, there are several challenges and issues that must be studied carefully before a wide commercial deployment as summarized below [13, 14, 12]:

Limited bandwidth

Wireless links continue to have significantly lower capacity than infrastructure networks. The erroneous channel characteristics further decrease the channel capacity, making bandwidth a valuable resource for MANETs [12].

Dynamic topology

The topology of a MANET can change due to the mobility of the nodes in the network. Also, the dynamic topology membership may disturb the trust relationship among nodes.

Time varying links

The wireless link characteristics are time-varying in nature. The communication channel between the nodes in the network is highly unreliable. The terminals communicate via a channel which is subjected to fading, noise, interference and path loss as compared to wired networks.

Battery constraints and power management

The nodes that constitute these networks have restrictions on the power source in order to maintain portability of the device.

QoS guarantees

There is a need to provide QoS requirements due to demanding applications. MANETs originally used to transmit ordinary data. Scheduling of this data was always easy because this data was homogeneous. Of recent, commercial usage of multimedia transmission over MANETs has been growing rapidly with respect to the number of users and the amount and type of traffic. This convergence of multimedia traffic with traditional data traffic creates yet another challenge because the former requires strict delay whereas the latter is delay-tolerant. Moreover, it is not unusual to have delay sensitive and non-delay-sensitive applications coexisting in the same network, hence making QoS provisioning to be a critical issue [15].

Real-time traffic is delay sensitive, therefore, the design of an efficient priority scheduling scheme that will ensure that the mobile nodes in MANETs transmit traffic to the desired expectations of the users under strict deadline constraints becomes crucial. In designing priority-based scheduling policies, the ultimate goal is to avoid job starvation [16].

The dominant data types transmitted over the internet like video and voice are also popular with MANETs. Cisco predicted that by 2022 online video would comprise more than 82% of the internet traffic [17]. Covid-19 pandemic and work from home policies completely revolutionized the way it is done and greatly increased the volume of data transported via the Internet with strict QoS demand for real-time voice and video services [18]. During the pandemic the networks witnessed growth in volume of traffic demands for high QoS for realtime voice and video services. In MANETs, a group of nodes constitute a network where the individual node can play sender, receiver or router roles. Therefore, scheduling is an important part to this kind of network for efficient data packet transfer within this network [19]. Intuitively, the different users/applications of the MANETs demand for different QoS requirements and this

is yet another critical challenge. Transmitting real time video contents such as video streaming or video conferencing over MANETs is a challenging task, because multimedia applications are delay sensitive and require an acceptable level of QoS to provide multimedia services. Whereas scheduling algorithms are vital to the MANETs in providing QoS to the different user requests, unfortunately, MANETs technology does not specify any specific scheduling scheme leaving it open to researchers and scholars to innovate in this area. There are resources e.g., bandwidth allocation in MANETs that need to be shared fairly amongst users in MANETs.

Resource scheduling in modern Information Technology (IT) systems serves mainly three goals; the first goal is to schedule system resources for example CPU and disk such that there is fair share allocation of resources; the second goal is to ensure that there is efficient use of the system resources; and the third goal of scheduling: scheduling to provide shorter mean response times across all requests in a system. The study focused on a third aspect to provide shorter mean response times across all packets in MANETs. In particular, studied three popular scheduling Algorithms namely: Abhaya et al. [20, 21] Earliest Deadline First (EDF); Kakuba et al. [22] Improved Low Latency Queueing (LLQ); and Hottmar et al. [23] mathematical model of Weighted Round Robin (WRR) strategy in order to solve the QoS provision challenges.

1.3 Statement of the problem

Nowadays, MANETs are widely used in a range of applications because of their ability to provide communication without fixed infrastructure. However, MANETs exhibit unique challenges among others is QoS provision due to the many demanding applications and limited bandwidth. Transmitting real time multimedia traffic over MANETs is a challenging task, because multimedia applications are delay sensitive and require an acceptable QoS guarantee [24, 25]. The need for improved network traffic flow and bandwidth management becomes critical when there are increasing amounts of real time applications are transmitted within MANETs. Recent developments within the research community provide numerous scheduling algorithms namely: EDF, LLQ and WRR. Unfortunately, a large proportion of these existing algorithms are effective only under certain system loads. Intuitively, there are exceptions, where the algorithms are unable to satisfy the QoS needs of different users, implying that they cannot guarantee the service quality of high priority flows. The existing algorithms offer temporary priority to delay sensitive applications. By providing temporary priority to delay sensitive flows then this creates another problem of other data types being delayed or starved. In particular the low priority packets are penalized when using EDF algorithm; starvation of video packets in the LLQ algorithm; and poor performance during overload conditions in the WRR strategy. Precisely, there is a scheduling problem in MANETs.

1.4 Objectives

1.4.1 General objective

The main objective of this study was to develop novel efficient scheduling algorithms to improve QoS in MANETs.

1.4.2 Specific objectives

Specifically the other objectives of the study were:

- i. To study the performance of EDF, LLQ and WRR algorithms under varying workloads;
- ii. To model novel scheduling algorithms that are scalable and efficient based on the existing EDF LLQ and WRR in MANETs; and
- iii. To evaluate the performance improvement of the modelled scheduling algorithms against the existing algorithms.

1.5 Research questions

The study was guided by the following three research questions:

- i. What is the performance of the existing EDF, LLQ and WRR scheduling algorithms in MANETs under varying workloads?
- ii. How can novel EDF, LLQ and WRR scheduling algorithms be modelled with improved QoS over the existing ones in MANETs?
- iii. What is the performance improvement of the modelled scheduling EDF, LLQ and WRR algorithms?

1.6 Scope

The study considered a MANET with heterogeneous hand-held portable mobile devices that were restricted from getting out of the transmission range of each other. The nature of data transported over the network comprised of mainly voice, video and text.

1.7 Significance and Relevance of the work

The improved and proposed new scheduling algorithms shall benefit, researchers, scholars and innovators in scheduling real time traffic in MANETs by reducing high delays that lead to unacceptable QoS performance in terms of latency, therefore supporting improvements of systems. The results of the study reveal, that we did a good job in reducing waiting time for real-time applications. In relation to the evolution of the 5G and 6G networks are looking at real-time applications in video or Internet of Things (IOT), the study results clearly indicate a good contribution in next generation networking and mobility. Even IOT the most popular device is the mobile phone which also works well in the MANETs.

1.8 Thesis contributions

The following are the main contributions of this study. The study:

- i. Adopted the generic pre-emptive Abhaya [20, 21] EDF model to the MANET environment to formulate a non-pre-emptive EEDF-I algorithm;
- ii. Enhanced the EEDF-I algorithm to formulate the EEDF-II algorithm. The EEDF-II algorithm addressed the following specific gaps
 - a) The starvation of low priority queue packets.
 - b) The poor performance of the EEDFI model at high network traffic loads.
- iii. Enhanced the Kakuba [22] LLQ algorithm by shifting from the MMPP/G/1 to the M/G/1 queue system.
- iv. Proposed an LLQ algorithm that utilizes the technique of splitting video packets while receiving service where one part of the video packet is transmitted along-side with the voice packets.
- v. Further extended the proposed LLQ algorithm and investigated the performance variation of the ELLQ scheduling algorithm with three queues (voice, video and text) under different workloads.
- vi. Adopted the Hottmar [23] mathematical WRR strategy into the M/G/1 queue system to the MANET environment to formulate the EWRR algorithm.
- vii. Enhanced the EWRR algorithm to formulate the IWRR algorithm. The proposed IWRR that utilizes the technique of computing the partial average waiting times of the small/large voice/video packets and in turn shorten the conditional average response time and slowdown.

1.9 Publications

The contributions in Section 1.8 resulted in the following publications:

- i. Mukakanya Abel Muwumba, Godfrey Njulumi Justo, Libe Valentine Massawe and John Ngubiri (2020). Priority EDF Scheduling Scheme for MANETs. In: Gao, H., Feng, Z., Yu, J., Wu, J. (eds) Communications and Networking. ChinaCom, Springer, Cham. https://doi.org/10.1007/978-3-030-41114-5_6
- ii. Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri (2023). An Improved Low Latency Queueing Scheduling Algorithm for MANETs. In: Arai, K. (eds) Advances in Information and Communication. FICC 2023. Springer, Cham. https://doi.org/10.1007/978-3-031-28076-4_9
- iii. Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri (2023). An Improved WRR Scheduling Algorithm for MANETs. In: Arai, K. (eds) Intelligent Computing. SAI 2023. Springer, Cham. https://doi.org/10.1007/978-3-031-377174_66
- iv. Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri The Performance Analysis of Low Latency Queueing Scheduling Algorithm for MANETs". PriMera Scientific Engineering 2.6 (2023): 03-12. <https://doi.org/10.56831/PSEN-02-054>

1.10 Thesis outline

The thesis is organized into eight Chapters. Chapter 1, provides the introduction to MANETs; discuss their evolution; the application gap they fill; their strength and the inherent weakness of MANETs; the motivation and background of the study, defines the problem and states the thesis contributions.

The remainder of the thesis is organized in seven Chapters. Chapter 2, reviews the various scheduling techniques that have been proposed in literature for QoS of MANETs, the perceived strengths and limitations, what can be done to improve the technique. The Chapter presents the packet scheduling algorithms in MANETs. The desired properties and classification of algorithms are discussed exhaustively. The challenges of scheduling in wireless networks are discussed and some scheduling solutions in communication network infrastructure are presented. The Chapter also presents the related work on EDF schemes, the LLQ algorithm and the WRR strategy. The research gaps conclude this Chapter.

Chapter 3 presents queuing theory, the MANETs queuing model, the two main service distributions that are used in the analysis, the performance metrics measured. and the three main methods used in the MANETs performance evaluation.

Chapter 4, adopts the generic EDF algorithm proposed by Abhaya [20, 21]; enhance the adopted algorithm of Abhaya; and carry out performance evaluation of the algorithms at various system loads. The Chapter introduces EDF, explains how it works, and discusses the types of EDF deadlines. It describes the generic EDF and the Kleinrock [26] framework used to compute the main performance measure. The weakness and the main performance measure of the generic EDF algorithm are also presented. The EEDF-I algorithm is discussed plus adoption steps and the justification for improvements. The Chapter further describes the EEDF-II algorithm and indicates the improvements. It then presents the theoretical and analytical results plus discussion of the results.

Chapter 5, presents the adopted Kakuba et al. [22] model; improved on the adopted Kakuba LLQ model; and discusses the performance evaluation of the algorithms at varying workloads. The Chapter presents the mathematical notations and expressions; describes the existing Kakuba LLQ and the adopted LLQ model to justify the improvements made; plus highlight the changes be made. Also, the proposed LLQ scheduling algorithm its improvements plus the steps followed in the modelling the algorithm are discussed. The results of the performance evaluation of the LLQ algorithms are presented in other parts of the Chapter.

Chapter 6, presents the extended work on the proposed LLQ scheduling algorithm to a three priority queue (consisting of voice, video and text packets) model. A brief introduction of the existing LLQ is provided, then the design and development of the ELLQ algorithm is discussed. The results and discussions on the performance of the ELLQ algorithm are presented to show the performance gains.

Chapter 7, presents the adopted WRR algorithm proposed by Hottmar [23]; discusses the improved adopted algorithm of Hottmar; and the performance evaluation of the algorithms at varying workloads. The generic WRR algorithm plus the mathematical notations and expressions are first introduced. The Chapter presents the adopted WRR algorithm into MANETs environment and indicates the changes made. Then the Improved WRR algorithm plus justifying the reasons for the improvements are discussed. Also, the numerical results on the performance of the WRR algorithms are presented.

The conclusion and future research are finally presented in Chapter 8.

Chapter 2 Literature Review

This Chapter provides a review of the various scheduling techniques that have been proposed in literature for QoS of MANETs by looking at what has been done by other scholars. Section 2.1 presents the classifications of packet scheduling algorithms. Section 2.2 presents the desired properties of packet scheduling algorithms for MANETs. Section 2.3 discusses the challenges of scheduling in wireless network. In Section 2.4, we present some applications of scheduling in communication network infrastructure. Section 2.5 presents the related work on EDF schemes, discusses the general structure, the common variations; their earlier and recent solutions.

Section 2.6 presents the related work on the LLQ algorithms, their general structure, the common variations; the earlier and recent improvements in the LLQ solutions. Section 2.7 presents the WRR algorithm, discusses how it works; its structure; the generic strengths and weakness of WRR; and the earlier WRR solutions. Section 2.8 discusses the challenges encountered when getting literature review. Section 2.9 presents the research gaps and concludes this Chapter.

2.1 Classification of scheduling algorithms in MANETs

For the scheduling process to be implemented in MANETs the wireless medium plays a key role in transporting data. However, studies have revealed that the wireless medium is a shared and scarce resource, which is used by all nodes in the MANET. The task of efficiently controlling the access to this scarce resource requires resource management schemes. There are broadly two resource management schemes namely: channel aware scheduling algorithms and packet scheduling algorithms [27].

2.1.1 Channel Aware Scheduling Algorithms

Channel aware scheduling is a resource management scheme that controls which node should get access among the set of competing nodes in a contention region.

Channel Aware Scheduling for Mobile Ad hoc networks (CaSMA)

CaSMA scheduling technique for MANETs takes into consideration both the congestion state and end-to-end latency [28]. This scheduling scheme is channel aware, which refers to having the knowledge/information about the channel conditions such as the quality of the channel which is expressed in terms of suitable metrics for example delay, throughput and fairness.

CaSMA focuses on end-to-end channel awareness and considers the end-to-end channel condition which is represented as residual lifetime for channel-awareness. A queue size parameter is included to make the scheduling scheme congestion aware. As consequence of this combination of parameters the accumulation of packets in the network is reduced and congestion is avoided [28].

Power-Aware Link Maintenance (PALM)

The Power-Aware Link Maintenance protocol was proposed, and is responsible for the power control along with route maintenance Cho et al. [29]. The protocol is based on Ad-hoc On Demand Distance Vector(AODV) Routing algorithm [30]. In the proposed scheme the nodes uninterruptedly monitor the received signals and determine the transmission power based on the previously received channel state information. The nodes compute the gain in the channel and the hop distance after receiving the data. While scheduling a link transmission, the proposed PALM algorithm locks the current link and monitors the routing table for the next link entry. The next link is also possibly locked for certain amount of time thus creating a power efficient, loop free routes in the network. As periodic beaconing leads to misuse of channel resources the route maintenance in PALM is done by power control with beaconing done only for active nodes [31]. The following are main drawbacks of channel dependent scheduling in MANETs [31]:

- Survival of the fittest

The survival of fittest policy states that the user with good channel conditions gets frequent access to the shared resources. This implies that the users located near the BS have more access and those furthest from it are starved in spite of good channel quality leading to unfair scheduling in channel dependent techniques.

- Delay in Reporting the Channel Quality Indicator (CQI)

Another key issue in channel dependent scheduling is the correctness of the CQI. An active user should constantly provide its CQI information to the BS in the network. The duration taken by the user to provide the CQI information varies from the time taken to receive it in the BS. A small time differs helps the BS to get a better understanding about the channel conditions and perform the scheduling. Sometimes this can create an overhead for users with limited power consumption. If the time difference is large, then the user benefits but leads to problem in the BS. This might cause faulty transmissions when the users location is changed but the BS maintains scheduling the channel conditions for the location yet the user location has changed.

2.1.2 Packet scheduling algorithms

This packet scheduling technique controls the allocation of bandwidth among multiple flows in MANETs. The scheme selects the flow which should be served among the set of backlogged flows within a node. The nodes are distributed in the serving area and data is to be sent directly between them [32].

No-Priority scheduling

First-in First-out (FIFO) scheduling algorithm is the most basic queue scheduling algorithm. This scheduling technique places all the packets in a single queue and are processed based on the order of arrival [33]. This queuing technique is predictable and requires a very small computational load. The maximum delay in the network is determined by the maximum size of the queue. The disadvantage of FIFO queuing is the inability to provide differentiated services for different traffic classes. As all the packets are placed in a single queue, a bursty flow occupies the entire buffer preventing the service for other flows till the queue gets emptied.

Priority Queuing (PQ) algorithm

Priority queue requires a small computational load and provides differentiated services based on the nature of the packets. The incoming packets are classified and placed in different priority queues. Packets with the highest priority are serviced before the packets with lower priority [34, 35]. In MANETs the control and data packets are maintained in the separate queues in FIFO order and high priority is assigned to control packets. When the high priority traffic is in excess compared to the low priority traffic, then the low priority traffic is likely to be dropped as the buffer space allocated to it starts to overflow resulting in denial of resources to low priority traffic.

Weighted Fair Queuing (WFQ) algorithm

Flows with different bandwidth requirements are scheduled by WFQ algorithm using a processor sharing (PS) system. WFQ allocates proportionate share of the bandwidth for each flow according to the weight attributed to it [36]. The incoming packets are placed in the respective flow queues and are time stamped with a finish time. The packets with smallest finish time are selected as the next packet for transmission on the output port by the WFQ scheduler [34, 35]. This algorithm guarantees minimum level of output bandwidth for each service class and is independent of the other service classes. The main problem of WFQ is the complexity involved in computation of virtual time. The virtual time associated with the discipline needs to be updated each time a queue empties or a packet of a class is queued. These events are repeated several times in a packet transmission time. Hence, complexity increases as a function of the number of packets completing service. Also, WFQ does not provide bandwidth guarantee for low traffic; and separate queue cannot be assigned for user defined classes.

Class Based Weighted Fair Queuing (CBWFQ) algorithm

CBWFQ expands the functionality of WFQ and supports the user-defined traffic classes. This algorithm explains about each traffic classes which are based on the access control lists, protocols and input interfaces. The packets with the matching criteria are added to the respective traffic classes. A queue is created for each class and the traffic related to that class is added to the corresponding queue. CBWFQ services the class queue fairly based on the weight assigned to the queued packets [34, 35] The amount of bandwidth allocated for each traffic class is clearly stated and the use of access control lists and protocols describes the traffic classification. However, there is no proper method to provide a strict priority queue and reduce latency in real time traffic such as VoIP.

Weighted-hop scheduling

Weighted-hop scheduling gives high priority to the data packet which has only a few hops remaining to traverse. When a packet has got fewer hops to traverse then it requires more potential to quickly reach the destination. A packet scheduler usually serves the packet in a WRR fashion. A WRR scheduler is utilized instead of a static priority scheduler to avoid starvation by providing opportunities to all service classes. WFQ [37] or deficit queuing [38] is used to allocate the right amount of bandwidth.

Weighted-distance scheduling

Weighted distance scheduling algorithm considers the physical distance using a GPSR in MANETs [34, 35]. Each data packet in a GPSR protocol contains the destination address. Nodes which are close to the physical distance are also closer in the network topology. When the physical distance to the destination decreases the remaining hops to the destination in the network topology also decreases. A weighted-distance scheduler can also be referred to as a WRR scheduler where higher weights are assigned to the data packets with shorter remaining physical distances to the destinations.

Round Robin (RR) scheduling

The newly arrival packets are queued up by flow in which each flow has its corresponding queue. The scheduler polls each flow queue in the cyclic order and serves the packet from any empty buffer encountered. RR scheduling greatest strength is that of having low computational complexity. The RR method is widely used for time-sharing systems because they provide higher fairness and better bandwidth utilization. However, one limitation with RR, is its lack of flexibility in an attempt to treat all flows identically [39].

WRR scheduling

This scheduling scheme is based on the RR and priority scheduling algorithms. WRR is a fair queuing algorithm where packets receive service based on the queue to which they belong. In every round of service, the number of packets served from a queue is proportional to its associated weight and the mean packet size. Under WRR discipline, if the queues are continuously backlogged there is fairness among different classes. However, WRR has shortcomings within a single round of service when a queue is empty; the empty queue will not receive extra service in later service rounds. Also, it is unaware of the true size of the packets in each queue while performing scheduling.

Greedy Scheduling

In greedy scheduling algorithm, each node forwards its own data packets before forwarding those of other nodes. The other nodes data packets are serviced in FIFO order. Suppose, there are two queues in a class following greedy scheduling. The queue N1 has its own data packets and has a strict priority over the queue N2 which has the data packets of the other nodes. This greedy scheduling scheme is very common in MANETs.

LLQ algorithm

LLQ algorithm operates with a single strict priority queue in which different traffic classes are placed. Strict priority queuing in LLQ guarantees delay sensitive traffic such as voice to be processed before the other queues, which makes it ideal for delay and jitter sensitive applications [34, 35]. Special preference is given to one or more delay sensitive traffic classes. All traffic classes from different priority classes are queued in the same, strict priority queue. These LLQ strict-priority queue does not starve all other queues and are regulated by the percentage of bandwidth or only bandwidth [40]. In this scheme, the LLQ strict priority queue is policed. This means that the LLQ strict priority queue is a priority with a minimum bandwidth guarantee, but at the time of congestion, it cannot transmit more data than what its bandwidth permits. The LLQ algorithm has the following limitations.

- It cannot guarantee the expected QoS level, if sensitive audio and video packets are processed in the single priority queue due to resource sharing between many application
- The presence of bursty video packet interferes with the voice traffic from being transmitted successfully [41]. The reason is that the behavior of the voice traffic is controllable whereas the video traffic is uncontrollable.

EDF scheduling

One of the most well-known scheduling algorithms for real-time network services like multimedia applications is EDF. The EDF is used for scheduling the packets in greedy manner where the packets with the nearest deadline are always selected. It provides

time-dependent priority for each eligible packet compared to the strict priority scheme. EDF allows the guarantee of QoS if the traffic characteristic of each flow follows the particular constraint. One of the major drawbacks with EDF is that this policy performs very poorly in overloaded conditions because it gives highest priority to packets that are close to missing their deadlines, thus delaying other packets that can still meet their deadlines.

2.2 Desired properties of packet scheduling algorithms in MANETs

Scheduling in MANETs refers to two problems: Channel access scheduling and Packet scheduling [42]. Channel access scheduling refers to which node should get access among the set of competing nodes in a contention region. In communication networks transmission resources are always shared. Therefore, the process of assigning users packets to appropriate shared resource to achieve some performance guarantee is called packet scheduling. There are specific *desired properties of packet scheduling algorithms*. The mobility of nodes and the error prone nature of the wireless media introduce many challenges in MANETs. These problems increase the packet delays and decrease the throughput. This calls for design of efficient scheduling algorithms to provide solutions for performance degradation. This study focuses on packet scheduling algorithms because they provide QoS guarantees in MANETs. Precisely, the main idea behind scheduling is to minimize starvation. The main goal of packet scheduling algorithms is to maximize the system capacity while satisfying the QoS of users and achieving certain level of fairness. Specifically, a packet scheduling algorithm should satisfy the following desired properties [2].

Efficiency

The basic function of packet scheduling algorithms is to schedule the transmission order of packets queued in the system based on the available shared resource in a way that satisfies the set of QoS of each user. In highly loaded conditions, efficiency (measured in terms of total achieved throughput) is one of the most significant performance criteria. Therefore, a packet scheduling algorithm should perform efficiently by providing same QoS guarantees even in overloaded conditions.

Flexibility

Besides QoS guarantees, another desired property is a packet scheduling algorithm should be able to support users with different QoS requirements. It should be able to provide service to best effort, data and real-time traffic. Real time applications have specific timing requirements. Examples are videoconferencing and video-on-demand, in which the timely delivery of packets must be maintained to ensure continuity of the image and sound. Thus, a packet scheduling algorithm should provide QoS support.

Low complexity

A packet scheduling algorithm should have reasonable computational complexity to be implemented. There is fast growing demand for bandwidth and faster transmission rates in today's communication systems, thus processing speed of packets becomes more and more critical. Thus, the complexity of the packet scheduling algorithms is of great concern.

Protection

A packet scheduling algorithm should treat the flows like providing individual virtual channels, thus traffic characteristics of one flow will have as little effect to the service quality of other flows as possible.

2.3 Challenges of scheduling in wireless network

Time varying channel conditions

In fixed IP networks packet scheduling plays a crucial role in providing QoS for the end users. A number of scheduling techniques have been proposed with different intentions for instance for providing fairness among the users [43, 44, 37] or guaranteeing a certain delay bound [45, 46]. However, scheduling techniques for fixed networks cannot be directly implemented in a wireless environment because wireless networks have certain unique characteristics such as: limited bandwidth, and location-dependent and time-varying

channel conditions that set additional requirements for the scheduling technique [47]. The channel conditions vary in a short time scale due to path-loss variation, slow log-normal shadowing and fast multipath-fading phenomena [48]. In a longer time-scale channel quality variation is caused by mobility and interference from the surroundings. Therefore, also the service quality perceived by the users is highly time-varying unless the scheduling technique takes the channel conditions into account

User dissatisfaction

It is possible to do scheduling that generate a QoS metric and while a metric might give an impression that the algorithm is doing well, the user who is the owner of the job is not satisfied with the service. For instance, if a metric is user waiting time and the scheduling discipline is Last Come First Served (LCFS) it means the job that arrives last has almost zero waiting time, But the earlier arrivals will be starved in the queue waiting for the late arrivals to be served.

Low utilization of the system

At low arrival rates the utilization of the system is low. Any job which arrives is served therefore the details of the scheduler is irrelevant since there is NO waiting. When we are to assess schedulers, we always compare them at worst case scenarios to test the robustness.

Mismodelling of arrival rates

Modelling the workloads, assumes a distribution of the arrivals of the jobs and the distribution of their service times. But there is always the distribution of the means during the overall working time. The arrival rate is not fixed it fluctuates depending on the peaks. Therefore, when we fail to correctly model the variations of the means, we mis model the system especially at overloads.

2.4 Applications of scheduling in communication network infrastructure

Shortest Remaining Processing Time (SRPT) scheduling for web servers and Least Attained Service (LAS) scheduling for network routers was used to illustrate how scheduling can tremendously enhance user experienced system performance, without necessarily purchasing additional system hardware or software. SRPT-based scheduling solution that significantly improves both server stability and client experience during transient overload conditions was proposed [49]. A solution that compared LAS to FIFO for different network topologies, namely a chain, a grid and a wireless LAN topology and different workloads, namely longlived flows or flow size distributions with various levels of skewness was proposed [32]. A simulation approach was used to show that: LAS solves all known fairness issues in wireless networks. The result revealed further that fairness is not obtained at the expense of performance in a wireless LAN.

Harchol-Balter [50] compared the mean response time under the different policies (First Come First Served, Processor sharing, Shortest Job First, Pre-emptive Shortest Job First Shortest Remaining Processing Time) as a function of load for an M/G/1 under the Weibull job size distribution and showed that conditional mean response time increased with load. However, one of the differences between the current study and that of that Harchol-Balter [50] is that it considered the exponential and BP distributions instead of Weibull job size distribution in the analysis of the adopted LLQ and WRR algorithms.

Rai and Okopa [51] proposed SWAP models and evaluated the scheduling policy using workloads under two service distributions. The numerical results obtained from the derived models reveal that SWAP approximates Shortest Job First (SJF) better for heavy-tailed workloads than for exponentially distributed workloads. The results also showed that SWAP performs significantly better than First Come First Served (FCFS) and Processor Sharing (PS) policies regardless of the distribution of the workload. Although the solution utilized the M/G/1 queue system, it did not focus on applications in MANETS.

Briefly the WFQ and WRR queue scheduling algorithms are discussed because they are widely employed for implementing differentiated services among multiple classes. The WFQ algorithm and some earlier solutions are presented because they are closely related to the WRR. The WFQ is a sophisticated algorithm designed by Demers et al. in [35], building on the work of Nagle [52]. The WFQ

algorithm partitions the available bandwidth among queues of traffic based on their weights. The algorithm assigns the bandwidth for each service based on the weight assigned to each queue and not based on the number of packets. In other words, it can guarantee each class bandwidth share proportional to its assigned weight but comes at a cost of greatly increased complexity to implement the scheduling discipline. The weight for each packet is calculated by multiplying the packet size with the inverse of weight for the associated queue. The WFQ is based on the system virtual definition [35]. The WFQ scheduler assigns a start tag and a finish tag to each arriving packet and serves packets in the increasing order of their finish tags. Although WFQ-like schedulers are very popular, there are very few analytical results available in the literature [53]. In this study we indicate only a few of them for the benefit of our readers. The Idealized Wireless Fair Queueing (IWFQ) algorithm, proposed by [54] is one of the earliest representative packet scheduling algorithms for wireless access networks and to handle the characteristic of location-dependent burst error in wireless links. The difference between IWFQ and WFQ is that when the wireless link, which is in bad (or error) state a packet will not be transmitted (because the packets transmitted will be corrupted). The packet with the next smallest virtual finish time will be picked and the process will repeat until the scheduler finds a packet with a good link state. One limitation with the IWFQ is that it does not consider the delay/jitter requirements in real-time applications.

Amongst the other earlier solutions was the WFQ [43] and its variant Worst-case Fair Weighted Fair Queueing (WF2Q) [44] that have good delay and fairness properties but have high implementation complexity. A newly localized and fully distributed fair queuing model provides measurable plus effective solutions to the queuing problems faced in MANETs was proposed [55].

2.5 Related work on EDF

2.5.1 General structure of EDF

The general structure of the EDF scheduler is that the packets are arranged in one queue according to their deadlines. The packets with the closest deadline are served first while those with longer deadlines wait. There is a possibility of new arrivals with shorter deadlines joining the queue and receive service before the earlier arrivals whose deadline is longer. The deadlines are fixed i.e., they do not change while the packets are in the queue. As such the EDF scheduler [56] is one of the simplest multiplexing algorithms, especially well-suited for real-time traffic, as it is able to adhere to the delay deadlines that traffic classes require. Voice and video are examples of real-time traffic, as opposed to best-effort traffic, which includes e-mail, ftp and http.

2.5.2 Advantages and limitations of EDF scheduling

- i. Meeting Deadlines
EDF ensures that tasks with the earliest deadlines are executed first. By prioritizing tasks based on their deadline EDF minimizes the chances of missing deadlines and helps to meet real-time requirements.
- ii. Optimal Utilization
EDF maximizes CPU utilization by allowing tasks to execute as soon as their deadline arrive as long as the CPU is available. It optimizes the use of system resources by maximizing idle time.
- iii. Responsiveness
EDF provides a high level of responsiveness for time critical tasks. It ensures that tasks are scheduled and executed promptly, reducing the response times and improving system performance.
- iv. Predictability
EDF provides predictability in terms of task execution times and deadline. The scheduling decisions are deterministic and can be analyzed and predicted in advance, which is crucial for real-time systems.
- v. Flexibility. EDF can handle both periodic and aperiodic tasks, making it suitable for wide range of real-time systems. It allows for dynamic tasks creation and scheduling without disrupting the execution of existing tasks.
- vi. However, this algorithm has significant complexity deriving from an incremental cost of classification packets, which increases with the queues length, furthermore, it experiences efficient implementation problem in case of high load.
- vii. The notable limitations include: Transient overload problem; Resource sharing problem;

2.5.3 The earlier EDF analytical models

Telecommunication companies around the world expressed interest in finding ways to merge real-time traffic, such as voice conversations, video-conferencing and video-on-demand with best-effort traffic, on the same IP network [57]. The goal was to offer a cost-effective and more adaptable solution than carrying real-time and best-effort traffic on separate networks. In order to engineer the future networks, analytical models were required to predict the behavior of the various networking components. Unfortunately, for the EDF scheduler, an exact analysis, as offered by conventional queueing theory, cannot be used to model modern networks carrying heterogeneous traffic, as these networks are generally too complicated. The problem with the EDF scheduler is that packets are not served in order of arrival, as is the case with FIFO.

It is a known fact that Liu and Layland [45] were the first to innovate the EDF as a real-time scheduling algorithm. The EDF scheme assigns a deadline to each packet, which is used by the scheduler to define the order of service. The highest priority job is the one with the earliest deadline. It essentially schedules the jobs in a greedy manner which always picks the jobs with the closest deadline. A solution that finds an approximate expression for the mean value of the queueing delay in an EDF queueing system was proposed [58]. This solution is non-preemptive and work-conserving M/G/1/.EDF model which is supported by general workloads. Andrews [59] relies on a Chernoff bound to find an expression for the deadline violation probability. Sivaraman and Chiussi [60] compares a similar chernoff bound to an expression derived using the effective bandwidth theory and large deviation principles and finds that for heavy loads the chernoff bound is looser than the second method.

Some earlier solutions that offer analysis of the EDF algorithm include [61] which extends the classic response time analysis techniques to analyze tasks scheduled either with fixed priorities, or with an EDF scheduler running on top of an underlying fixed priority scheduler. Albers and Slomka [62] present the fast exact feasibility tests for uniprocessor real-time systems using preemptive EDF scheduling. An integrated schedulability theory for EDF is presented. Finally, for the multiprocessor scenario, Baker [63] derives a new schedulability test for preemptive.

2.5.4 The recent EDF analytical models

More recent solutions that offer analysis of the EDF algorithm include [64], an analytical method for approximating the performance of a two-class priority M/M/1 system. In this model the prioritized class-1 jobs were considered to be real-time and served according to the EDF scheduling policy, and the non-real-time class-2 jobs were served according to the FIFO policy. One limitation with this model is that it is not an exact analytical solution for the analysis of EDF, even for a system with purely real-time jobs. A multi-queue EDF and its variant Flexible Earliest Deadline First (F-EDF) was proposed [65].

A pre-emptive M/M/1/EDF and non-pre-emptive M/M/m/EDF model that investigated mean sojourn times in multi-class queues with feedback and their application to packet scheduling in communication networks was proposed [66]. The model assumed exponentially distributed service times and these may not suitably represent web services workloads because the services could be used in exposing any type of system. A packet scheduling algorithm consisting of EDF algorithm and least slack time algorithm is proposed for scheduling the various multimedia applications [67]. This scheduling algorithm is used to reduce the transmission delay and to achieve better QoS requirements.

While reviewing the literature, the study found out that analytical EDF Priority schedulers are not common in MANETs, though there are a few that exist. An EDF scheduler which permits service differentiation for real time traffic is proposed scheduling [68]. Every packet is assigned a deadline that is used by the scheduler to determine the order of service. It is not possible to serve all the packets before the assigned deadlines based on the server load. An efficient QoS architecture using inter layer communication with a highly efficient real time scheduler design at the network layer with improved rate monotonic algorithm and EDF scheduling that efficiently schedules multiple real time applications without missing any of their deadline was proposed [69].

A pre-emptive EDF scheduling scheme that approximates the mean waiting time for a given class based on the higher and lower priority tasks receiving service prior to the target and the mean residual service time experienced was proposed [20, 21]. The goal of this EDF scheduler was to favour higher priority packets thereby reducing their waiting times. The limitation with this approach is that favouring higher priority queues yielded increased waiting times of lower priority queues. The study found out that this EDF algorithm could be adopted and studied in a MANETs environment. This motivated the study to investigate the performance of EDF algorithms in a MANETs environment.

2.6 Related work on LLQ

2.6.1 General structure of LLQ

The general structure of the LLQ scheduler is that the packets are arranged in two or more queues according to their data types or size of packets. There is a high and low priority queue. Delay sensitive voice and video traffic is placed in the high priority queues and hence it is scheduled first before best effort traffic such as email, ftp, http in the low priority queues. Within the individual queues packets are served in FIFO order and there is normally NO preemption of jobs.

2.6.2 Common variations

The variations come in the following forms, for instance, there could be bursty video packets which arrive at the system when the system is highly loaded which might lead to starvation of low priority packets. What decision does the LLQ scheduler have to make? Is bursty video traffic served uninterrupted to completion or pre-empt the bursty video traffic to serve the small size low priority packets normally who consist the majority of internet traffic.

2.6.3 The earlier LLQ schemes

There are a few papers that have studied the behavior of LLQ. Semeria [34] developed an LLQ algorithm that utilizes a single priority queue in which separate traffic classes are placed. The strict priority queuing allows delay sensitive traffic such as voice to be processed prior to the other queues. A novel Low Latency and Efficient Packet Scheduling (LLEPS) algorithm was developed to ensure low latency for real time audio and video streaming applications [70]. A model was developed to give a higher priority to voice and video traffic which is the most sensitive [71]. This model monitors and classifies all incoming traffic based on the level of their sensitivity. The highest priority is assigned to voice and video traffic and a lower priority to other traffic which are delay tolerant. This sequence occurs so that high priority traffic can be delivered to the destination directly without considering a congestion avoidance technique.

Farzad et al. [72] proposed a scheduling algorithm that takes traffic types of flows into consideration when scheduling packets and also it is provided scheduling flexibility by trading off video quality to meet the playback deadline. Shaimaa et al. [73] proposed a scheme that combines CBWFQ and LLQ. Simulation results revealed that applying LLQ for voice improves the performance of overall real-time and non real-time application.

An urgency-based packet scheduling technique was proposed to deliver delay sensitive data in mobile networks effectively [74]. Packet urgency, route urgency and node urgency were defined, based on the end-to-end delay requirements and the number of hops over a route. The urgency metrics determined the order of packet scheduling for dropping the packets.

A model to support real time service in e-learning environment was developed [75]. The model is a multi-tiered e-learning system based on web services whose architecture took into consideration QoS requirements such as: traffic dropped, traffic received and packet end to end delay. Jui-Chi [76] proposed a batch arrival model to analyze multimedia packet scheduling for next generation mobile networks. The model provides favorable scheduling and maximum bandwidth utilization according to specific QoS constraints. A scheduling algorithm that categorizes and prioritizes the real-time traffic was developed [77]. Although this technique offers preferential treatment to real-time traffic, its use of only a single strict priority queue (predominantly for voice traffic) renders the solution unsuitable for other data types. If more traffic arrives more than what the strict priority queue can transmit (due to strict bandwidth limit) it is dropped. Hence at times of congestion other queues do not starve, and get their share of the bandwidth to transmit their

traffic. Therefore, traffic in the strict priority queue is penalized at the expense of other data types.

In MANETs, each node is connected through wireless links with other nodes and multihop transmission is used to send and receive data across a wireless network. Communication within the network is made possible with the aid of routing protocols to discover paths between the nodes. Zakaria et al. [78] proposed an analytical model that attempted to utilize queuing theory to analyze the performance of a MANET protocol by determining its arrival times, average waiting times, and response times.

2.6.4 Recent improvements of the LLQ schemes

Ali et al. [79] proposed an analytical model to determine the utilization, mean waiting time, mean queue length and distribution of the number of packets in the queue and average response time of the cluster heads when applying flushing technique in MANETs.

Kacem et al. [80] introduced a new routing method based on the fuzzy synchronized Petri Net model and optimization algorithm for MANETs. Specifically, they use Fuzzy Petri Nets to discover and decision-making select optimal routes, while the algorithm is used to find a solution for uncertain events in the network. Experiment results demonstrated the efficiency of the proposed solution in terms of performance and QoS compared to existing solutions.

The results on mean response time on scheduling policies for a single queue, the M/G/1 queue are presented [81]. In the proposed solution, the mean response time under the different policies as a function of load and as a function of the squared coefficient of variation of job size, C_x^2 were compared.

Zhang et al. [82] proposed a new QoS management multicast model based on genetic algorithms in MANETs. Specifically, the model guarantees the duration time of a link in a multicast tree is longer than the delay time from the source node. Experimental results demonstrated that the proposed model improved QoS flows and the delay compared to existing methods in MANETs environments. Sivaram et al. [83] proposed a Re-transmission Dual Busy Tone Multiple Access (RDBTMA) protocol to address the problems associated with hidden and exposed terminals. The problems exist due to non-transitivity in media access control schemes, and affect the channel utilization plus throughput in media access control protocols. The simulated results show that the proposed RDBTMA protocol is effective in terms of the improved QoS namely; network throughput and packet delivery ratio than the existing methods.

Chen et al. [84] proposed a multipath routing protocol to guarantee QoS for high mobile MANETs scenarios. Specifically, they proposed a mechanism to predict the disconnect of links. Then, they use a multi-metric routing algorithm based on parameters, such as remain energy, bandwidth, length of the queue, reliability of the link. Experimental results demonstrated that the proposed protocol improved performance and QoS compared to existing protocols in scenarios the movement speed of nodes up to 108 km/h. Khan et al. [85] proposed a routing approach relying on the game theory for guarantee QoS in MANETs. Specifically, they evaluate the reputes of each node based on the collaboration level of a node aims to encourage the positive of nodes jointly in the data forward. Experiment results demonstrated the proposed solution improved performance and QoS compared to existing solutions.

Harchol-Balter and Scheller-Wolf [86] introduced SOAP policies with a very broad class of scheduling policies for the M/G/1 queue, and analyzed some notable policies like the Gittins index policy, which has long been known to minimize mean response time in settings where exact job sizes are not known.

Rukmani et al. [87] proposed an enhanced LLQ with an additional Primary Strict Priority Queuing (PSPQ) for scheduling the video applications separately along with the dedicated Secondary (SSPQ) for voice applications. The performance of the proposed algorithm is compared with other existing algorithms through simulations using the OPNET modeler. The simulation and statistical results revealed that the proposed algorithm recorded a performance improvement in terms of throughput and delay than the existing algorithms for the real time audio and video applications. It is not possible to schedule the video packets from the SSPQ before scheduling the voice packets from the PSPQ by considering the nature of the application.

Sohail et al. [88] proposed an improved packet scheduling algorithm which utilizes a mapping criterion and an efficient queuing mechanism for voice, video and other traffic in separate queues. It was proven, that the algorithm enhanced the throughput and fairness along with a reduction in the delay and packet loss factors for smooth and worst traffic conditions.

Kakuba et al. [22] proposed an improved LLQ algorithm modelled in a non-preemptive priority MMPP/G/1 queue system. In this model, real-time traffic is classified into voice and video packets. The LLQ provides priority for voice packets to ensure that they are not stuck behind the large video packets.

2.7 Related work on WRR

2.7.1 How WRR works

WRR is a network scheduler for data flows and is also used to schedule processes. This scheduling scheme is based on the RR and priority scheduling algorithms. WRR is a fair queuing algorithm where packets receive service based on the queue to which they belong. The general structure of the WRR scheduler is that the packets are arranged in a RR manner with each queue assigned weight. The packets are classified according to their data types or size of packets. If all packets have the same size, WRR is the simplest approximation of General Processor Sharing (GPS).

In every round of service, the number of packets served from a queue is proportional to its associated weight and the mean packet size. Under WRR discipline, if the queues are continuously backlogged there is fairness among different classes.

2.7.2 Strengths and weakness of WRR

- a. The following are the generic strengths of the WRR scheduler.
 - i. WRR can be implemented in hardware, therefore it is used in high-speed interfaces in both the core and at the edge of the network.
 - ii. WRR queuing provides a coarse control over the percentage of output port bandwidth allocated to each service class.
 - iii. WRR queuing ensures that all service classes have access to at least some configured amount of network bandwidth to avoid bandwidth starvation.
 - iv. WRR queuing provides an efficient mechanism to support the delivery of differentiated service classes for a reasonable number of highly aggregated traffic flows.
 - v. Classification of traffic by service class provides more equitable management and more stability for network applications than the use of priorities or preferences.
- b. On the other hand, WRR has some limitations
 - i. The greatest limitation of WRR is that it provides the correct percentage of bandwidth to each service class only if all the packets in all of the queues are the same size or when the mean packet size is known in advance. Also, it is unaware of the true size of the packets in each queue while performing scheduling.
 - ii. Another limitation with WRR is that within a single round of service when a queue is empty, the empty queue will not receive extra service in later service rounds.

2.7.3 The earlier WRR schemes

The WRR algorithm has been implemented in various ICT infrastructure for example: Katevenis et al. [89] present the architecture of a general-purpose Broadband Integrated Service Digital Networks (B-ISDN) switch chip implements its scheduling function consisting of a weighted round-robin multiplexing scheme. The WRR allows differentiating in service class handling. Packets are first classified into service classes and assigned to specified queues. Each queue is visited by the scheduler and packets are sent from the queue. The mechanisms built into the chip can be applied by the network manager to offer guaranteed service performance to the real-time traffic, and to fully utilize the spare capacity of the links by serving the lower-priority traffic and minimize congestion.

Chaskar and Madhow [90] proposed the WRR schemes that are described and analyzed for fixed packet sizes. These particular type of WRR schedulers are applicable for cell scheduling in Asynchronous Transfer Mode (ATM) networks. They are also used in certain other scheduling scenarios where the schedulable unit of bandwidth is intrinsically of a fixed size. For instance, the time slot scheduling on wireless link in (GPRS) networks, and frame scheduling on the air interface in third generation CDMA wireless networks.

Qian, et.al [91] used network calculus to study the Strict Priority Queuing (SPQ) and WRR scheduling with respect to the worst-case timing behavior of individual flows in the network. On comparing, the service behavior, it was revealed that WRR serves traffic in fair manner whereas SPQ is unfair because the flows with low priority are starved. Also, WRR is more flexible for QoS provision since it enables to balance the allocation of shared network bandwidth to different traffic flows respecting their delay constraints. It was further noted that WRR is capable of providing isolation to individual flows. A new iterative mathematical model for calculating the average bandwidth assigned to traffic flows using a WRR scheduler in IP networks is presented [92]. The bandwidth assignment estimation is based on the average packet length, link speed and the arrival rate. Balogh and Medvecky [93] present a packet scheduling technique based on parallel usage of multiple WRR schedulers, rate limiters and output bandwidth calculation for modern NGN networks, whose focus is to provide queueing fairness within queues. Most of the earlier WRR algorithms provide only fairness in the output allocation between queues but fairness in the queue is mostly not taken into account.

Gautam et al. [94] proposed an algorithm that improves the QoS by focusing on three parameters such as packet drop ratio, throughput and time delay. Hottmar and Adamec s [23] mathematical model was aimed at solving the challenge of QoS in converged IP networks. Unfortunately, starvation video packets were observed at the expense of voice packets.

2.8 Limitations in accessing literature

In this study we encountered a challenge of referring to papers that were beyond 10 years old. The explanation for this is that sometimes it happens, it depends on the type of research you are doing. While dealing with simulation and theoretical analysis or analytical modelling it makes sense because this type of work does not take time. We are working with formulars, it is not a kind of architecture. The formulars can be used in all infrastructure including next generation infrastructure.

2.9 Research gaps

From reviewing the literature, the study identified the following research gaps:

- i. Many of the existing algorithms were working in different ICT infrastructure. The review also revealed that most of the earlier WRR algorithms provide only fairness in the output allocation between queues but fairness within a given queue is mostly not taken into account. In addressing the above gap, the study made improvements to the classical Abhaya EDF model and the Hottmar WRR strategy by adopting them to the MANETs environment.
- ii. The study established that analytical priority schedulers are not common in MANETs. Although some few scholars who made attempts to exploit the use of queueing theory in packet scheduling, the size-based schedulers in MANETs have remained a virgin area. In order to bridge this gap, the study employed analytical size-based approaches in the LLQ and WRR models
- iii. There are a few works that have been published that attempt to analyze the behavior of EDF [45, 58-60, 95-97]. Though for many applications, an exact analysis real-time system is possible but not very popular because the exact tests take long execution times and are limited to simple models [2, 3, 11]. On the other hand, there are several approximation methods have been used to model EDF. The study also established that the existing EDF solutions failed to provide QoS guarantees to jobs which have long deadlines having waited and are about to receive service but there is a possibility of missing service due to pre-emptions. The study bridges this gap by proposing a non-pre-emptive EEDF-II scheduler. This was done on the basis that pre-emption is wasteful in terms of network resources.

Chapter 3 Research Methodology

This Chapter first presents queueing theory in Section 3.1. The main idea of discussing queueing theory is that our schemes are modelled based on M/G/1 and M/G/c queueing systems. The MANETs queueing model is presented in Section 3.2. The Section further presents the two main service distributions that are used in the analysis and also discusses the performance metrics measured. Section 3.3 presents the three main methods used in the MANETs performance evaluation. This Section also explains the strengths and limitations of these methods plus the analysis tool used in this research. Section 3.4 concludes the Chapter.

3.1 Queueing theory

3.1.1 Introduction

Queueing theory is a branch of mathematics that studies and models the act of waiting in lines [98]. The study of queues (waiting lines) is popularly referred to as queueing theory. The history of queueing theory and networks dates back to the beginning of 20th (twentieth) century and the details of how it has evolved is presented in [99, 100]. Queues (or waiting lines) play a crucial role in facilities or businesses to provide service in an orderly manner. In real life environments, queues (or waiting lines) are noticed when customers are waiting to be served at banks, supermarkets, airports, service centres, food cafeteria, hospitals. There are also queues of inquiries waiting to be processed by an interactive computer system, queue of database requests, queues of I/O requests [101].

Queues are also common in complex systems like computer networks, telecommunication systems (wired and wireless networks including MANETs) and call centres. In queueing theory, the unit demanding service, whether it is human or otherwise is identified as the customer whereas the unit providing service is known as the server. In computer networks, telecommunication systems the customers that are common are: user requests, jobs, flows, packets, transactions or programmes [99].

3.1.2 Kendall's notation

Queueing theory, makes use of standard system popularly referred to as Kendall's notation to describe and classify the queueing events [102, 103]. The notation takes on the form A/B/c/K, where A denotes the interarrival time distribution; B represents the service time distribution; c is the number of servers; and K is the size of the system capacity. Quite often the symbols A and B are represented by M for exponential distribution (M denotes for Markov); D for deterministic distribution; and G for general distribution.

There are several queueing systems but the most common examples are: M/M/1, M/M/1/K, M/M/c, M/G/1 and G/M/1 [101]. The M/M/1 queueing system is characterized by: exponentially distributed interarrival times; exponentially distributed service times; One server and the service discipline is FCFS. The customers arrival process is poisson with rate λ . For the M/M/c queueing system there are $c \geq 1$ servers and the waiting room is of infinite capacity. Inria [101] asserts that queues are always finite in practice, hence a new customer is dropped when he finds the queue(s) full (e.g., telephone calls). For the case of the M/M/1/K the maximum allowable number of customers, including the customer in the service facility, if any is K. The M/G/1 general queue system, the arrival process is poisson, with rate, $\lambda > 0$, and the service times of the customers are independent with the same, arbitrary, cumulative density function (cdf) $G(x)$.

3.2 The MANET queueing model

A MANETs queueing system can be described as customers arriving for service, waiting for service if it is not immediate (waiting room), and if having waited for service, leaving the system after being served. In this system the arrivals place demands upon a finite capacity resource. The components of queueing system are inputs, queue and one or more servers for serving customers arriving in some manner and having some service requirements.

3.2.1 The service distributions

The study considered two distributions i.e., exponential and BP. The exponential distribution was used to represent low Coefficient of Variation (CoV) empirical packet sizes. The short form of the exponential distribution is denoted as exp; The exponential distribution with rate λ is defined by the probability density function (pdf) is given [51, 104] as: $f(x) = \lambda e^{-\lambda x}$, $x \geq 0, \lambda \geq 0$.

If T is a random variable that represents interarrival times with the exponential distribution, then $P(T \leq x) = 1 - e^{-\lambda x}$ and $P(T > x) = e^{-\lambda x}$. This distribution was used in modeling the packet interarrival times or service times for the following reasons. Firstly, the exponential function is a strictly decreasing function of t . This implies that after an arrival has occurred, the amount of waiting time until the next arrival is more likely to be small than large. Secondly, the exponential distribution is known as to have memoryless property. The no-memory property means that the time until the next arrival does not depend on how much time has already passed. This makes intuitive sense for a model where we were measuring packet arrivals because the packets actions are clearly independent of one another events [98].

While the BP distribution was used as a typical example of high CoV empirical packet sizes for large P values. The short form of the BP distribution is denoted as; $BP(k, L, \alpha)$ where k and L are the minimum and the maximum packet sizes, and α is the exponent of the power law (is a theoretical or empirical relationship governed by a power function). The pdf of the Pareto is given [51, 104] as;

$$f(x) = \frac{\alpha k^\alpha}{1 - (k/L)^\alpha} x^{-\alpha-1} \quad k \leq x \leq L, \quad 0 \leq \alpha \leq 2.$$

The BP that emerge in computer system applications typically have $\alpha \in (0.9; 1.3)$ [104].

3.2.2 The model input

The MANET consists of N interconnected nodes. In this model, traffic arrives according to a Poisson process and consists of packets. Each node corresponds to one packet at any one time. The nodes are modelled by queuing network in which each of the n identical site is represented by an M/G/1 system. Packet arrivals to the nodes are modelled with parameter, λ for exponential distribution. The exponential and BP distributions were used to generate workloads.

3.2.3 The model output and performance metrics

The study considers user-based performance metrics. The prime interest is to look at performance (and fairness) from the users point of view. The adopted user based metrics are average waiting time, conditional mean response time is denoted as $T(x)$ and the conditional mean slowdown, $S(x)$.

Response time (sojourn time or flow time), T refers to the total time a packet spends in the system or the time between when a packet arrives and when the packet completes service. Therefore, the study considered $T(x)$, response time for packet conditioned on that packet being of size x referred to as conditional mean response time and its units are seconds.

Slowdown (or stretch) is the ratio of the response time of a packet to the size of that packet, denoted, $S(x)$, is equal to $T(x)/x$. The slowdown metric is important because it helps to evaluate unfairness. For example, in an M/G/1 system with PS scheduling, all packets (long and short) experience the same expected slowdown (hence PS is "fair"). $S(x)$ was the third performance metric we considered and its units were seconds per byte.

In this study the metric average waiting time was used to evaluate the EDF schemes;

While the metrics $T(x)$ and $S(x)$ were used in the evaluation of the LLQ and WRR algorithms.

3.3 Methods and tools

There are three main methods applied to evaluate MANETs performance namely measurement/experiments, simulations and theoretical analysis/analytical modelling [105]. The selection of the method was based on the following considerations i.e., the life-cycle stage in which the system is (e.g. does the system exist at all), the time and resources available and the level of detail needed. The study adopted analytical modelling and simulation approaches. The scheduling problem of the EDF, LLQ, and WRR algorithms were represented mathematically using expressions. However, the analytical models could not be used to generate the results because the solution is Non-deterministic Polynomial (NP) hard. For the simulation approaches the study adopted MATLAB simulation tool to get numerical results. The justification for this philosophy is that queuing theory has a foundation in mathematics, hence a scheduling problem can be solved mathematically.

3.3.1 Experiments

The method of measurement requires conducting experiments on a real system and provides the most direct means of network performance evaluation. In scenarios where attributes of the physical layer (namely; real world properties) are defining for the system performance or other investigated features, experiments with real equipment in the desired conditions is the preferred method [105]. Although this method is practiced by many network vendors, it has the following limitations.

- Performing experiments with more than a few nodes requires a great effort from participants in the experiment, especially in order to support realistic mobility. The work load prior to the experiment is high, since the equipment must be prepared with the correct software versions, charged batteries and so on.
- Schormans et al. [106-108], also ref. [109] have explored the inaccuracy inherent in packet level measurement, which are caused mainly by inappropriate probing patterns and rates. It has been discovered that even for static traffic in simple buffering scenarios there are practical load limits beyond which measurement accuracy degrades very rapidly.
- Additionally, it would not normally be accurate to draw general conclusions from measurement results, as many of the environmental parameters (like system configuration, time of measurement) may be unique to the experiment and may not represent the range of variables present in the real world.
- Measurement is expensive and cannot be done until the real system is built. Even before the experiment, the workload can be in higher order of magnitude because all the equipment must be installed with the correct software versions, charged batteries, etc. Thus, the infrastructure required to perform large experiments makes it more cost effective to perform research using simulation and analytical modelling.

Owing to the substantial work effort required to perform large experiments, the study sought it more efficient to perform research using simulators and theoretical analysis within the limits of a PhD program.

3.3.2 Analytical modelling

Analytical modelling involves constructing a mathematical model of the system behaviour [78]. Mathematical model can be used to abstract the essential characteristics of a computer system and analyse the system behaviour [110]. Analytical models can generally be solved rather quickly; however, a tractable analytical model often restricts the range of system characteristics that can be explicitly considered. Nonetheless, they can be effective when carefully applied. For example, this method can be considered as the best approach to determine the effects of various parameters and their interactions (provided a valid analytical model is available). Analytical modelling is usually the cheapest and fastest of the three techniques.

3.3.3 Simulations

Simulators are software tools used to create a virtual environment that supports researchers to set up and test a network's performance under different conditions [?].

Simulations enable the investigation of the dynamics occurring when the distributed interaction is too complex to model using theoretical analysis, especially in combination with mobility [105]. Simulation techniques (namely: using a computer to model the operations of a real-world system) can model a network to an arbitrary degree of detail, and may therefore be closer to reality than analytical models. Simulator is a very popular and important tool for research community not only for MANETs but for other networks also. Simulators provide an inexpensive method to evaluate any research without the use of actual hardware. Some of the simulation tools that are used to investigate MANETs are: Network Simulator-2 (NS-2), GloMosim, CSIM, OPNET, QualNet and MATLAB.

The performance results presented in this thesis were obtained using MATLAB simulations. MATLAB is one simple and popular tool for generating the numerical results from the analytical models. The implementation of the algorithms was done in MATLAB version R2021a that was installed on Windows Operating System platform. The experiments were run on a Dell Laptop Latitude 3520 with the following specifications: System Microsoft Windows 11 Pro, Version 22H2; Processor, 11th Gen Intel(R) Core (TM) i3-1115G4 @ 3.00GHz 2.19 GHz and 4.00 GB (3.74 GB usable) of RAM. The advantage of MATLAB implementations of the algorithms is that it allows for fast numerical performance evaluation. The tool offers the following benefits:

- i. *Flexibility* - MATLAB provides an intuitive language and a flexible environment for technical computations which integrates mathematical computing and visualization tools for data analysis and development of algorithms and applications.
- ii. *Ease of Use* - MATLAB is an interpreted language. Program may be easily written and modified with the built-in integrated development environment and debugged with the MATLAB debugger.
- iii. *Platform Independence* - MATLAB is supported on many different computer systems, providing a large measure of platform independence.
- iv. *Predefined Function* - MATLAB comes complete with an extensive library of predefined functions that provide tested and pre-packaged solutions to many basic technical tasks. For example, the arithmetic mean, standard deviation, median and so on.
- v. *Device-Independent Plotting* - Unlike most other computer languages, MATLAB has many integral plotting and imaging commands.
 - Even though simulations provide an easy way to investigate the distributed properties of algorithms, simulators cannot simulate the world in its entirety, but have different areas where they are strong and weak.
 - The solution of a simulation model requires significantly more computer resources than an analytical model, despite this limitation, in some cases, simulation is the only viable modelling approach and is the most common approach to the evaluation of network performance [111, 112].

In some cases, both analytical and simulation models may be used. For example, simulation can be used to check the impact of the assumptions needed in an analytical model, while an analytical model can suggest appropriate parameters to investigate in a simulation study [113].

3.4 Conclusion

This Chapter presented an overview of the research methodology employed during the work. A more thorough presentation of the research methodology on the modelling of the the EDF, LLQ and WRR schemes can be found in the respective Chapters 4, 5, 6 and 7.

Chapter 4 EDF Scheduling in Manets

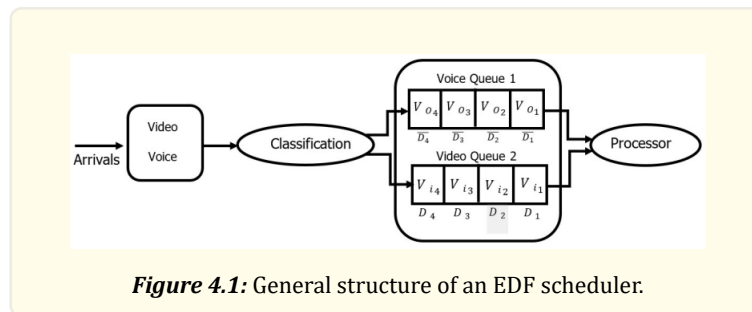
In this Chapter, we adopt the generic EDF algorithm proposed by Abhaya to MANETs; enhance the adopted algorithm of Abhaya; and carry out a performance evaluation of the algorithms at various system loads. In Section 4.1, we introduce EDF, explain how it works, and discuss the types of EDF deadlines. Section 4.2 describes the assumptions made in EDF development. Section 4.3 describes the generic EDF and contains the framework used to compute the main performance measure. The weaknesses and the main performance measure of the generic EDF algorithm are also presented. Section 4.4 describes the EEDF-I algorithm. It discusses the adaption steps and the justification for improvements. In Section 4.5, we describe the EEDF-II algorithm and indicates the improvements. Section 4.6 presents the theoretical and analytical results. Also, the discussion of the results is presented. Section 4.7 concludes the Chapter.

The results have been published in: Mukakanya Abel Muwumba, Godfrey Njulumi Justo, Libe Valentine Massawe and John Ngubiri (2020). Priority EDF Scheduling Scheme for MANETs. In: Gao, H., Feng, Z., Yu, J., Wu, J. (eds) Communications and Networking. China-Com, Springer, Cham. https://doi.org/10.1007/978-3-03041114-5_6

4.1 Introduction to EDF schemes

4.1.1 The EDF algorithm

The EDF scheduler is designed for real-time applications. It operates with the principle of sorting packets in order of their absolute deadlines [66]. When a packet arrives at a node (router or switch), the EDF scheduling algorithm assigns an absolute deadline equal to its arrival time in the node plus the relative deadline of the flow to which it belongs. Its philosophy is that a job should be scheduled before the deadline expires, thus EDF considers the packets relative deadline being priority key, such that the packet that is closer to expire gets higher priority than other arrived packets to the scheduler's queue [114]. Figure 4.1 shows the general structure of the EDF scheduler. V_{o1} to V_{o4} are voice packets in priority queue 1; V_{i1} to V_{i4} are video packets in priority queue 2; and D_1 to D_4 are the respective deadlines assigned to the voice and video packets. When voice and video packets arrive at the EDF scheduler, they are classified in priority queue 1 and 2 packets. In each priority queue, the packets are sorted and scheduled in order of their deadlines. A packet with short deadline is scheduled first and those with long deadlines are scheduled last.



4.1.2 EDF timing requirements

In designing real-time systems two different types of timing requirements are catered for: hard and soft deadlines.

- Hard deadlines are applicable to hard real-time systems. In these systems all jobs must meet their deadlines with a missed deadline being treated as a fatal fault. Hard real-time systems are commonly found in control systems, especially in avionics systems or automobile engine control applications, safety or mission critical applications [115]. They are designed to ensure that there are no missed deadlines often at the expense of resource utilization and average performance.
- Soft deadlines have a much larger class of applications, referred to as soft real-time systems. These systems allow some jobs to miss their deadlines in order to improve resource usage or average performance. In other words, the value of output lowers with missed deadline. Soft deadlines tolerate lateness and use the jobs completed after the deadline [115]. Thus, it may be necessary to schedule a job that has missed its deadline, it allows some degree of flexibility in that it can be extended. Soft real-time systems have a wide range of applications like audio and video transmissions, where the end user is able to tolerate a small lack of continuity in the sound or image being transmitted. The goal of a soft real-time system is to meet as many deadlines as possible before attempting to maximize the average performance. Applications with firm deadlines are permitted to miss their deadlines. However, once the deadline has been missed, there is no value in completing a job whose deadline has expired.

The study of the EDF scheduling is interesting, especially since this algorithm has been proved optimal in a mono-processor environment where the moments activation of the packets is not known a priori [116-118]. EDF is an ideal scheduler for real-time flows because the optimality of this algorithm has been proved for number criteria [66]. Its implementation in real networks has been in numerous earlier studies [59, 119-123] and in the recent past [20, 21, 64-67]. It has also been implemented to a small extent in MANETs

studies [68, 69]. The study focused on the soft deadline case in order to address the problem of EDF scheduling in MANETs by reducing on the delay of the low priority packets in queue 2 have to wait before being serviced.

4.2 Assumptions

- i. For evaluation purposes, we consider a system of 4 sources, 2 routers and 10 destination mobile nodes as the three main components. A router can send the packet to the destination mobile nodes via MANETs.
- ii. The system has identical nodes with battery life of at least 12 hours.
- iii. The study assumes that the arrival process for each priority class is an independent Poisson process with rate λ_i and that the service times are exponentially distributed with mean $\frac{1}{\mu_i}$.
- iv. Links (or connections) between stations are assumed error free.

4.3 The design of EDF schemes

4.3.1 The generic EDF algorithm

The aim of this Section is to describe the generic EDF Abhaya et al. [20, 21] algorithm. Our description is closely linked to the Kleinrock [26] system because it is the foundation of the generic Abhaya EDF algorithm. The Kleinrock [26] system contains a generic framework used to compute the average waiting times. The framework is introduced as follows: Assuming a newly arriving tagged packet from queue i . N_{ji} refers to the average number of packets of queue j arriving prior to the tagged packet of queue i , and are serviced before the tagged packet does. M_{ji} refers to the average number of packets of queue j arriving after the tagged packet of queue i , and are serviced before the tagged packet does. \bar{W}^0 refers to the average residual service time and is defined as the mean value of time required to finish the service of the packet currently being served when the tagged packet arrives. In an M/G/1 queue,

$$\bar{W}_i = \sum_{i=1}^N \rho_i \frac{E(x_i^2)}{2E(x_i)} \quad (4.1)$$

Where ρ_i represents the total load experienced by the system, $E(x_i)$ represents the mean of the service time distribution, and $E(x_i^2)$ represents the second moment of the service time distribution.

The average waiting time is given by:

$$\bar{W}_i = \bar{W}^0_i + \sum_{i=1}^N 2E(x_i)[N_{j,i} + M_{j,i}] \quad (4.2)$$

In Kleinrock [26], it is assumed that queue 1 has the highest priority and N the lowest. For queue 1, due to the strict priority and the FCFS discipline within a same queue, we get $N_{ji} = 0$ for $j > i$; $N_{ji} = \lambda_j \bar{W}_j$, where λ_j is the arrival rate of packets in queue j . for $j \leq 1$ represents the average number of packets in queue j . $M_{ji} = 0$ for $j \geq i$; $M_{ji} = \lambda_j \bar{W}_i$ for $j < i$ which is the number of packets of higher priorities coming during our tagged customer's waiting time.

The study first considers the EDF discipline with $N=2$ (independent poisson input queues). Equation 4.2 is directly applied to analyze the parameters N_{ji} and M_{ji} in relation to the EDF discipline. The EDF discipline has the following noticeable characteristics packets from queue 2 can be allowed to be serviced before some packets in queue 1. Therefore, the waiting time of queue 1 and queue 2 packets ordered by EDF policy is closer to each other. Packets in the same queue have the same deadline offset because the service order is FCFS. Let us follow a newly arriving tagged packet from queue 1.

- Since the service is FCFS within the same queue, all packets found in queue 1 before the arrival of the tagged packet receive service before it does. Late arrivals will be serviced after. Hence, $N_{1,1} = \lambda_1 \bar{W}_1$ and $M_{1,1} = 0$.
- Since $d_2 > d_1$, all packets of queue 2 that arrive after the tagged packet receive service after it, hence $M_{2,1} = 0$. Among the packets

present, which are $\lambda_2 \bar{W}_2$ on average, only those with a deadline ($D_{2,1}$) lower than the tagged packet will be serviced before. Thus $N_{2,1} = \max(0, \rho_2(\bar{W}_2 - D_{2,1}))$

Let us now follow a newly arriving tagged packet from queue 2.

- Since $d_1 < d_2$, all packets of queue 1 present receive service before the tagged packet does. Thus $N_{1,2} = \lambda_1 \bar{W}_1$
- For $M_{1,2}$ packets of queue 1 arriving not later than $D_{2,1}$ after the tagged packet will have shorter deadlines, receive service before the latter does. The tagged packet stays in the queue for less than $D_{2,1}$, therefore, $M_{1,2} = \rho_1 \min(\bar{W}_2, D_{2,1})$, recall when it is FCFS discipline within the same queue then $N_{2,2} = \lambda_2 \bar{W}_2$ and $M_{2,2} = 0$.

When we directly apply Equation 4.2, where ρ_1 and ρ_2 are loads due packets in queues 1 & 2 respectively; noting that $\lambda_i E(x_i) = \rho_i$ the expressions for average waiting time \bar{W}_1 and \bar{W}_2 are as follows:

$$\begin{cases} \bar{W}_1 = \bar{W}_1^0 + \rho_1 \bar{W}_1 + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) \\ \bar{W}_2 = \bar{W}_2^0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_1 \min(\bar{W}_2, D_{2,1}) \end{cases}$$

For queue i

$$\begin{cases} N_{k,i} = \lambda_k \bar{W}_k & 1 \leq k \leq i \\ N_{k,i} = \lambda_k \max(0, \bar{W}_k - D_{k,i}) & 1 < k \leq N \end{cases}$$

and

$$\begin{cases} M_{k,i} = \lambda_k \min(0, \bar{W}_i, D_{i,k}) & 1 \leq k < i \\ M_{k,i} = 0 & 1 \leq k \leq N \end{cases}$$

Assuming N=3, we have:

$$\begin{cases} \bar{W}_1 = \bar{W}_1^0 + \rho_1 \bar{W}_1 + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) + \rho_3 \max(0, \bar{W}_3 - D_{3,1}) \\ \bar{W}_2 = \bar{W}_2^0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_3 \max(0, \bar{W}_3 - D_{3,2}) + \rho_1 \min(\bar{W}_2, D_{2,1}) \\ \bar{W}_3 = \bar{W}_3^0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_3 \bar{W}_3 + \rho_1 \min(\bar{W}_3, D_{3,1}) + \rho_2 \min(\bar{W}_3, D_{3,2}) \end{cases}$$

Given the pre-emptive resume scheduling discipline considered, the mean waiting times must satisfy the conservation law [26, 124] for pre-emptive resume M/G/1 queues.

$$\sum_{k=1}^i \rho_k W_k = \frac{\sigma_i \bar{W}_i^0}{(1 - \sigma_i)} \quad (4.3)$$

According to Kleinrock [26], the mean residual service time, \bar{W}_0^i is given by:

$$\bar{W}_0^i = \sum_{k=1}^i (P_k \bar{R}_k) \quad (4.4)$$

Therefore, the generic equation of the average waiting time for queue i is given by;

$$\bar{W}_i = \left[\frac{\bar{W}_i^0}{(1 - \sigma_i)} + \sum_{k=i+1}^N \rho_k \max(0, \bar{W}_k - D_{k,i}) \right] + \sum_{k=1}^{i-1} \rho_k \min(\bar{W}_i, D_{i,k}) \quad (4.5)$$

Equation 4.5 represents the generic Abhaya EDF [20, 21] algorithm that applies to web services middleware. The symbols in equation 4.5 are explained as follows: \overline{W}_i , is the mean waiting time for a packet of stream/priority i for adopted EDF Abhaya model; \overline{W}_k , is mean waiting time for a packet of stream/priority k ; \overline{W}_0^i is mean time delay experienced by an arrival from stream i (or mean residual service time), from the packets already in progress; $D_{k,i}$ is the difference in the deadline offsets of streams i and k ; $D_{i,k}$ is the difference in the deadline offsets of streams k and i ; N is the number of independent streams through which requests arrive at the system following a poisson process; and ρ_k is the system load due to queue k packets.

Algorithm 1 The generic EDF Abhaya model

Consider a pre-emptive M/G/1 queue Mean waiting time for any class i For all incoming jobs classify into priority classes;
 Assign a deadline to each job;
 $i = 1 \leftarrow N \quad j = i + 1 \leftarrow N \quad D_{i,j} = d_j - d_i$;
 Compute the;
 Service times, \overline{X}_i ;
 Second moments, \overline{X}_i^2 ;
 System load, ρ ;
 Mean residual service, \overline{W}_0^i ;
 Probability of a request from stream;
 Mean delay experienced by a new arrival;
 Mean waiting time, \overline{W} ;

4.3.2 Weakness of the generic EDF Abhaya algorithm

- i. Was designed for middleware yet it is powerful algorithm that can be customised for MANETs.
- ii. The second weakness is that the generic Abhaya EDF scheme is pre-emptive, and failed to provide QoS guarantees to jobs which have long deadlines having waited and close to being served but there is a possibility of missing service due to pre-emptions. Pre-empting a job in transmission is wasteful in terms of network resources. In the study, we shift from pre-emptive to non-pre-emptive EDF scheduling.

4.3.3 The performance metrics

The mean waiting time experienced by a given priority class is considered to be the main performance measure approximated by the models. The study considered mean waiting time as a function of load in these models as Abhaya et al. [20, 21] urges these are the most commonly used metrics to measure the performance of a system.

4.4 Adaption of the EDF to MANETs

In the generic Abhaya EDF model the requests, jobs or tasks are received by the middleware, and those selected are serviced at each server using the EDF scheduling algorithm. It is possible to have multiple servers in a single MANETs environment. The study exploits this property of MANETs in the adopted EDF Abhaya model. The study refers to the adopted Abhaya EDF model as Enhanced Earliest Deadline First-1 (EEDF-1).

Adaption steps and justifications

The following changes in the generic Abhaya model are made. The study:

- i. Adopts an M/G/m queue which is a multi-server system. Specifically, we use the M/M/m queueing system with arrival rates λ and service time \overline{X} .
- ii. Computes the waiting probability following a non-pre-emptive M/M/m queue for server utilization.
- iii. Computes the mean residual service time for a request of stream/priority i is for a M/M/m system.

The modelling of the EEDF-I algorithm goes through the following steps:

- i. The study obtained the parameter for the average number of packets of the given queue arriving prior to the tagged packet that are serviced before the tagged packet does.
- ii. This was followed by obtaining the parameter for the average number of packets of a given queue arriving after the tagged packet that are serviced before the tagged packet does.
- iii. The parameter for the average residual service time was also computed.
- iv. From these three parameters we got the expression for the average waiting time of given queue for an M/G/1 system.
- v. Next, the study briefly explains the steps for modelling the EEDF-I algorithm. For simplicity, the study first considers the EEDF-I algorithm with $N=2$ queues with service order of FCFS within a same queue.
- vi. The expression for average waiting time in step iv is directly applied to analyze the parameters in steps i and ii in relation to the EEDF-I algorithm.
- vii. For queue 1, the study followed up a tagged packet, the parameters for packets found in queue 1 before the arrival of the tagged packet receive service before it does, and that for late arrivals serviced after the tagged were obtained
- viii. The study considered a scenario when the deadline of queue 2 is longer than for queue 1 to obtain parameters for all packets of queue 2 that arrive after the tagged packet receive service is obtained. Then a newly arriving tagged packet from queue 2 was followed.
- ix. The study considered a scenario when the deadline of queue 1 is shorter than for queue 2 to obtain the parameters for all packets of queue 1 present receive service before the tagged packet does.
- x. For queue 2, the study also followed up a tagged packet, and then considered packets of queue 1 arriving not later than deadline after the tagged packet having shorter deadlines, receive service before the latter does and obtained the parameters. The equation in step 4 was applied to derive the expression for average waiting times for queue 1 and queue 2.
- xi. The results in step x were generalized to the case of more than two queues. Hence the generic expression of average waiting time of any queue is obtained.

The EEDF-I model is presented in algorithm 2. The EEDF-I scheduling algorithm determines the way packets are processed by the M/G/m scheduling system depending on deadline priority factor. Four priority queues i.e., $P1$ -high, $P2$ -medium, $P3$ -normal and $P4$ -low are considered at the intermediary node (router). Routers transmit packets by selecting the packet with shortest deadlines in the high priority queue. If any packet exists in the high priority queue, then it is selected and transmitted. Else, a packet with the shortest deadline is selected from the medium priority queue and transmitted. If there does not exist any packet in the medium priority queue also, then normal priority queue is considered. Finally, the low priority queue is taken into account. This procedure is continued for every packet in the MANET traffic.

Algorithm 2 The EEDF-I model

Consider a non-preemptive M/G/m

For all incoming jobs classify into priority queues (queue 1 and queue 2 Assign a deadline

$i = 1 \leftarrow N \quad j = i + 1 \leftarrow N \quad D_{i,j} = d_j - d_i$

Compute the;

Service times, \bar{X}_i ;

Second moments, \bar{X}_i^2 ;

System load, ρ ;

Waiting probability of a request, P_i ;

Mean residual service time, \bar{W}_0^i ;

Mean delay experienced by a new arrival;

Mean waiting time, \bar{W} ;

Mean waiting time for any queue, \bar{W}_i ;

The generic expression of mean waiting time for any queue i for the EEDF-I model is given by Equation 4.6.

$$\bar{W}_i(EEDF - I) = \left[\frac{\bar{W}_i^0}{m(1-\sigma_i)} + \sum_{k=i+1}^N \rho_k \max(0, \bar{W}_k - D_{k,i}) \right] + \sum_{k=1}^{i-1} \rho_k \min(\bar{W}_i(EEDF - I), D_{i,k}) \quad (4.6)$$

All the variables in Equation 4.6 have been defined in Section 4.3, except the other additional variables m that stands for the number of identical servers in an M/G/m queue, and \bar{W}_i (EEDF - I) is the mean waiting time of the EEDF-I algorithm. We present the generic Abhaya EDF model in algorithm 1. From the EEDF-I generic model in Equation 4.6, the expressions for the *average waiting times* of for any number of priority queues can be deduced. In this study, the expressions for average waiting times $N=2, 3$ and 4 are as follows: For instance, when $N=2$ i.e., P1-high and P2-medium, then

$$\begin{cases} \bar{W}_1 = \frac{\bar{W}_1^0}{m(1-\sigma_1)} + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) \\ \bar{W}_2 = \frac{\bar{W}_2^0}{m(1-\sigma_2)} + \rho_1 \min(\bar{W}_2, D_{2,1}) \end{cases}$$

For instance, when $N=3$ i.e., P1-high, P2-medium and P3-normal;

$$\begin{cases} \bar{W}_1 = \frac{\bar{W}_1^0}{m(1-\sigma_1)} + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) + \rho_3 \max(0, \bar{W}_3 - D_{3,1}) \\ \bar{W}_2 = \frac{\bar{W}_2^0}{m(1-\sigma_2)} + \rho_3 \max(0, \bar{W}_3 - D_{3,2}) + \rho_1 \min(\bar{W}_2, D_{2,1}) \\ \bar{W}_3 = \frac{\bar{W}_3^0}{m(1-\sigma_3)} + \rho_1 \min(\bar{W}_3, D_{3,1}) + \rho_2 \min(\bar{W}_3, D_{3,2}) \end{cases}$$

For instance, when $N=4$ i.e., P1-high, P2-medium, P3-normal and P4-low; ρ_1, ρ_2, ρ_3 and ρ_4 are loads due packets in queues 1, 2, 3 & 4 respectively, then;

$$\begin{cases} \bar{W}_1 = \frac{\bar{W}_1^0}{m(1-\sigma_1)} + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) + \rho_3 \max(0, \bar{W}_3 - D_{3,1}) + \rho_4 \max(0, \bar{W}_4 - D_{4,1}) \\ \bar{W}_2 = \frac{\bar{W}_2^0}{m(1-\sigma_2)} + \rho_3 \max(0, \bar{W}_3 - D_{3,2}) + \rho_4 \max(0, \bar{W}_4 - D_{4,2}) + \rho_1 \min(\bar{W}_2, D_{2,1}) \\ \bar{W}_3 = \frac{\bar{W}_3^0}{m(1-\sigma_3)} + \rho_4 \max(0, \bar{W}_4 - D_{4,3}) + \rho_1 \min(\bar{W}_3, D_{3,1}) + \rho_2 \min(\bar{W}_3, D_{3,2}) \\ \bar{W}_4 = \frac{\bar{W}_4^0}{m(1-\sigma_4)} + \rho_1 \min(0, \bar{W}_4 - D_{4,1}) + \rho_2 \min(\bar{W}_4, D_{4,2}) + \rho_3 \min(\bar{W}_4, D_{4,3}) \end{cases} \quad (4.7)$$

The *average waiting time*, \bar{W}_1 for a tagged packet in P1 queue is given as;

$$\bar{W}_1 = \frac{\bar{W}_1^0}{m(1-\sigma_1)} + \rho_2 \max(0, \bar{W}_2 - D_{2,1}) + \rho_3 \max(0, \bar{W}_3 - D_{3,1}) + \rho_4 \max(0, \bar{W}_4 - D_{4,1}) \quad (4.8)$$

The *average waiting time*, \bar{W}_2 for a tagged packet in P2 queue is given as;

$$\bar{W}_2 = \frac{\bar{W}_2^0}{m(1-\sigma_2)} + \rho_3 \max(0, \bar{W}_3 - D_{3,2}) + \rho_4 \max(0, \bar{W}_4 - D_{4,2}) + \rho_1 \min(\bar{W}_2, D_{2,1}) \quad (4.9)$$

The *average waiting time*, \bar{W}_3 for a tagged packet in P3 queue is given as;

$$\bar{W}_3 = \frac{\bar{W}_3^0}{m(1-\sigma_3)} + \rho_4 \max(0, \bar{W}_4 - D_{4,3}) + \rho_1 \min(\bar{W}_3, D_{3,1}) + \rho_2 \min(\bar{W}_3, D_{3,2}) \quad (4.10)$$

The average waiting time, \overline{W}_4 for a tagged packet in $P4$ queue is given as;

$$\overline{W}_4 = \frac{\overline{W}_0^4}{m(1 - \sigma_4)} + \rho_1 \min(0, \overline{W}_4 - D_{4,1}) + \rho_2 \min(\overline{W}_4, D_{4,2}) + \rho_3 \min(\overline{W}_4, D_{4,3}) \quad (4.11)$$

4.5 The EEDF-II model

4.5.1 Modifications

The study indicates the following improvements in the EEDF-I model. The improved adopted EDF Abhaya is referred to as Enhanced Earliest Deadline First II (EEDF-II). The component:

- i. $\sum_{k=i+1}^N \rho_k \max(0, \overline{W}_k - D_{k,i})$ is removed to avoid excessive waiting time for low priority packets.
- ii. $\sum_{k=1}^{i-1} \rho_k \min(\overline{W}_i (EEDF - I), D_{i,k})$ is changed to $\sum_{i=2}^{N-1} \rho_i \min(\overline{W}_i (EEDF - II), D_{i+1,i})$ to improve network performance.

It is these modifications that are responsible for shortening the waiting times for the low priority packets.

4.5.2 Expressions for average delay

In the following the expressions for average delay of $P1$ to $P4$ packets are presented

- i. In the EEDF-II model, traffic consists of N number of independent queues that are classified by the scheduler into priority queues. Each queue is identified by i where $i = 1, 2, \dots, N$ and is associated with a different deadline. Packets from the same queue get assigned a constant deadline offset. Using EEDF-II algorithm, scheduling of packets among different queues, the requests from the same queue are serviced in a First-Come First-Served basis. The study considers the scenario of four priority queues where $N = 4$ i.e., $P1$ -high, $P2$ -medium, $P3$ -normal and $P4$ -low. Packets from $P1$ are assumed to have shorter deadlines than packets from $P2$, $P3$ and $P4$. In a typical priority-based system, higher priority packets are always serviced ahead of lower priority packets, since the priority is determined by the absolute deadline in the system under consideration. Since packets from $P1$ have higher priority over $P2$, $P3$ and $P4$ packets, under a high arrival rate of $P1$ packets; $P2$, $P3$ and $P4$ packets are served after $P1$. For the case of four priority queues $N = 4$, there are four view points of the average delay of a tagged packet in a specific queue after delay.
- ii. *The expression for average delay of $P1$ packets*

The tagged $P1$ packet will experience the following delays: $P1$ packets found in the queue will be serviced before the tagged packet, in this case $\overline{N}_{1,1} = \lambda_1 \overline{W}_1$. Where λ_1 is the arrival rate of $P1$ queue packets, and \overline{W}_1 is the mean waiting time for $P1$ queue packet. $P1$ packets from stream 1 arriving at the system after the tagged request will be served later i.e., $M_{1,1} = 0$. Therefore, the average waiting time, \overline{W}_1 for a tagged packet in $P1$ queue is given as;

$$\overline{W}_1 = \overline{W}_1^0 + \rho_1 \overline{W}_1 \quad (4.12)$$

- iii. *The expression for average delay of $P2$ packets*

The following are the delays experienced by the tagged $P2$ packet; delay due to $P1$ packets found in the queue when the tagged packet arrives, these packets will experience delay given by; $N_{1,2} = \lambda_1 \overline{W}_1$. Delay due to $P2$ packets found in queue $N_{2,2} = \lambda_2 \overline{W}_2$. Where λ_2 is the arrival rate of $P2$ queue packets, and \overline{W}_2 is the mean waiting time for $P2$ queue packet. Once the tagged packet arrives at the system, it will not wait for a portion of $P1$ packets to be served before it. These packets will have deadlines earlier than the tagged packet. Because packets from $P1$ that arrive after the tagged packet, they will be served after the tagged packet. Therefore $M_{1,2} = 0$, and the delay experienced by the tagged packet can be expressed by; $\overline{W}_2 = \overline{W}_0 + \overline{X}_1 \lambda_1 \overline{W}_1 + \overline{X}_2 \lambda_2 \overline{W}_2$. The average waiting time, \overline{W}_2 for a tagged packet in $P2$ queue is given as;

$$\overline{W}_2 = \overline{W}_2^0 + \rho_1 \overline{W}_1 + \rho_2 \overline{W}_2 \quad (4.13)$$

iv. *The expression for average delay of P3 packet*

The following are the delays experienced by the tagged P3 packet: delay due to P1 packets in queue $N_{1,3} = \lambda_1 \bar{W}_1$; P2 packets in queue $N_{2,3} = \lambda_2 \bar{W}_2$; and P3 packets in queue $N_{3,3} = \lambda_3 \bar{W}_3$. Where λ_3 is the arrival rate of P3 queue packets, and \bar{W}_3 is the mean waiting time for P3 queue packet. Given the waiting time of P3, the tagged packet may be in the queue for a time period less than $D_{3,2}$ given that $W_3 < D_{3,2}$. The delay can be estimated as $M_{2,3} = \lambda_2 \min(\bar{W}_3, D_{3,2})$.

$\bar{W}_3 = \bar{W}_0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_3 \bar{W}_3 + \bar{X}_2 \lambda_2 \min(\bar{W}_3, D_{3,2})$. $D_{3,2}$ is the difference in the deadline offsets of streams 3 and 2. The average waiting time, \bar{W}_3 for a tagged packet in P3 queue is given as;

$$\bar{W}_3 = \bar{W}_3^0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_3 \bar{W}_3 + \rho_2 \min(\bar{W}_3, D_{3,2}) \quad (4.14)$$

v. *The expression for average delay of P4 packet*

The following are the delays experienced by the tagged P4 packet: delay due to P1 packets in queue $N_{1,4} = \lambda_1 \bar{W}_1$; delay due to P2 packets in queue $N_{2,4} = \lambda_2 \bar{W}_2$; delay due to P3 packets in queue $N_{3,4} = \lambda_3 \bar{W}_3$; and delay due to P4 packets in queue $N_{4,4} = \lambda_4 \bar{W}_4$. Where λ_4 is the arrival rate of P4 queue packets, and \bar{W}_4 is the mean waiting time for P4 queue packet. Given the waiting time of stream 4, the tagged packet may be in the queue for a time period less than $D_{4,3}$ given that $W_4 < D_{3,2}$ and less than $D_{4,3}$ given that $W_4 < D_{4,3}$. The delay can be estimated as $M_{2,3} = \lambda_2 \min(\bar{W}_3, D_{3,2})$ and $M_{3,4} = \lambda_3 \min(\bar{W}_4, D_{4,3})$. $D_{4,3}$ is the difference in the deadline offsets of streams 4 and 3. $\bar{W}_4 = \bar{W}_3 + \rho_4 \bar{W}_4 + \bar{X}_3 \lambda_3 \min(\bar{W}_4, D_{4,3})$. The average waiting time, \bar{W}_4 for a tagged packet in P4 queue is given as;

$$\bar{W}_4 = \bar{W}_4^0 + \rho_1 \bar{W}_1 + \rho_2 \bar{W}_2 + \rho_3 \bar{W}_3 + \rho_4 \bar{W}_4 + \rho_2 \min(\bar{W}_3, D_{3,2}) + \rho_3 \min(\bar{W}_4, D_{4,3}) \quad (4.15)$$

Given the scheduling discipline considered, the mean waiting times must satisfy the conservation law for M/G/m queues [26, 124]. Because of the m servers;

$$\sum_{k=1}^i \rho_k W_k = \frac{\bar{W}_i^0}{m(1 - \sigma_i)} \quad (4.16)$$

Note: $\sigma_i = \sum_{k=1}^i \rho_k$. Substituting Equation 4.16 into Equations 4.12, 4.13, 4.14 and 4.15, we have the average waiting times, $\bar{W}_1, \bar{W}_2, \bar{W}_3$ and \bar{W}_4 for the tagged packets in P1, P2, P3 and P4 queues respectively as follows:

$$\bar{W}_1 = \frac{\bar{W}_1^0}{m(1 - \sigma_1)} \quad (4.17)$$

The average waiting time, \bar{W}_2 for a tagged packet in P2 queue is:

$$\bar{W}_2 = \frac{\bar{W}_2^0}{m(1 - \sigma_2)} \quad (4.18)$$

The average waiting time, \bar{W}_3 for a tagged packet in P3 queue is:

$$\bar{W}_3 = \frac{\bar{W}_3^0}{m(1 - \sigma_3)} + \rho_2 \min(\bar{W}_3, D_{3,2}) \quad (4.19)$$

And the average waiting time, \bar{W}_4 for a tagged packet in P4 queue is:

$$\bar{W}_4 = \frac{\bar{W}_4^0}{1 - \sigma_4} + \rho_2 \min(\bar{W}_3, D_{3,2}) + \rho_3 \min(\bar{W}_4, D_{4,3}) \quad (4.20)$$

$$\begin{cases} \bar{W}_1 = \frac{\bar{W}_1^0}{m(1-\sigma_1)} \\ \bar{W}_2 = \frac{\bar{W}_2^0}{m(1-\sigma_2)} \\ \bar{W}_3 = \frac{\bar{W}_3^0}{m(1-\sigma_3)} + \rho_2 \min(\bar{W}_3, D_{3,2}) \\ \bar{W}_4 = \frac{\bar{W}_4^0}{m(1-\sigma_4)} + \rho_2 \min(\bar{W}_3, D_{3,2}) + \rho_3 \min(\bar{W}_4, D_{4,3}) \end{cases} \quad (4.21)$$

The generic Equation for the *average waiting time* of the EEDF-II algorithm for P_i queue is given by;

$$\bar{W}_i(EEDF - II) = \bar{W}_i^0 + \frac{\sigma_i \bar{W}_i^0}{m(1-\sigma_i)} + \sum_{i=2}^{N-1} \rho_i \min(\bar{W}_i(EEDF - II), D_{i+1,i}) \quad (4.22)$$

Where $\bar{W}_i(EEDF - II)$ is e *average waiting time* for the EEDF-II algorithm. The EEDF-II model is presented in algorithm 3.

4.6 Implementation and Results

The EDF models were implemented in MATLAB to evaluate their performance.

Algorithm 3 The EEDF-II model

Consider a non-pre-emptive M/G/m queue

For all incoming jobs classify into priority queues

Assign a deadline

$i = 1 \leftarrow N \quad j = i + 1 \leftarrow N \quad D_{i,j} = d_j - d_i$

Compute the;

Service times, \bar{X}_i ;

Second moments, \bar{X}_i^2 ;

System load, ρ ;

Waiting Probability of a request, P_i ;

Mean residual service time, \bar{W}_0^i ;

Mean delay experienced by a new arrival after removal of excessive delay components;

Mean waiting time, \bar{W} ;

Mean waiting time for any queue, \bar{W}_i

4.6.1 Theoretical evaluation

The study applied the following steps in theoretical evaluation of the EEDF-I and EEDF-II algorithms

- i. Considered a given number of priority queues say $N = 2, 3$ or 4 .
- ii. Initiated the process by calculating the parameters needed to find the mean delay incurred by packets in execution, at an arrival.
- iii. Computed the waiting probability of a packet from a given queue being in service at an arrival.
- iv. Then computed the mean delay incurred by packets in service, for each queue.
- v. The values of the waiting probabilities in step ii above were used in the final step of calculating the individual average waiting time.
- vi. The computation of average waiting time is an iterative process which was done in the reverse order of priorities that is to say starting from the lowest priority then backwards for the EEDF-1 and vice versa for the EEDF-II.
- vii. The process was repeated for varying loads to get different readings of average waiting times.
- viii. For the performance evaluation, we made graphs of average waiting time vs load to analyze the M/G/1/. /EDF systems.

Deadlines	Deadline Differences	Service Times	System Loads	Arrival Rates	Second Moments
$d_1=1500\text{ms}$	$D_{2,1}=2500\text{ms}$	$\bar{X}_1 = 502.5\text{ms}$	$\rho_1=0.2311$	$\lambda_1=0.000460$	$\bar{X}_1^2=335673$
$d_2=4000\text{ms}$	$D_{3,1}=4500\text{ms}$	$\bar{X}_2 = 1502.5\text{ms}$	$\rho_2=0.3005$	$\lambda_2=0.000200$	$\bar{X}_2^2=2340673$
$d_3=6000\text{ms}$	$D_{4,1}=7500\text{ms}$	$\bar{X}_3 = 2502.5\text{ms}$	$\rho_3=0.2202$	$\lambda_3=0.000088$	$\bar{X}_3^2=6345673$
$d_4=9000\text{ms}$	$D_{4,2}=5000\text{ms}, D_{4,3}= 3000\text{ms}$	$\bar{X}_4=3502.5\text{ms}$	$\rho_4=0.1506$	$\lambda_4=0.000043$	$\bar{X}_4^2=12350673$

Table 4.1: Sample parameters.

In the study the system considered for the evaluation has four priority queues. As the EEDF-I and EEDF-II models are based on the work by Chen et al. [58] and Abhaya et al. [20, 21] similar sample parameters are used in the evaluation as shown in Table 4.1. The deadlines d_1, d_2, d_3 and d_4 are the same as those used in the Abhaya et al. Briefly some of the values were got as follows: For instance d_2 minus d_1 gives deadline difference $D_{2,1}=2500\text{ms}$; d_3 minus d_1 gives deadline difference $D_{3,1}=4500\text{ms}$; d_3 minus d_1 gives deadline difference $D_{3,1}=4500\text{ms}$; d_4 minus d_1 gives deadline difference $D_{4,1}=7500\text{ms}$; similar steps were repeated to compute the other remaining deadline differences. The study adopted Abhaya service time, $\bar{X}_1=502.5$ for the highest priority queue, the other remaining ones for the lower priority queues had an incremental interval of 1000, hence $\bar{X}_2=1502.5, \bar{X}_3=2502.5$ and $\bar{X}_4=3502.5$ respectively. In the table the values of the service times multiplied by arrival rates gave us the system load. Thus $502.5 \times 0.000460 = 0.2311$; $1502.5 \times 0.000200=0.3005$. This was repeated for the remaining parameters of service times and arrival rates. In the table the values for second moments were computed as follows: $E(x_i^2) = \bar{X}_i^2 + \sigma^2$. We assumed $\sigma^2=83166.75$ for demonstration purposes; Then $E(x_1^2) = 502.5^2 + \sigma^2=335673$; $E(x_2^2)= 1502.5^2 + \sigma^2=2340673$; $E(x_3^2) = 2502.5^2 + \sigma^2=6345673$; and $E(x_4^2) = 3502.5^2 + \sigma^2=12350673$: The system loads $\rho_1=0.2311$; $\rho_2=0.3005$; $\rho_3=0.2202$ and $\rho_4=0.1506$ were also assumed. Utilisation of the server by packets belonging to queue i is given by $\rho_i = \lambda_i \bar{X}_i$. Hence, the arrival rates were computed as follows: $\lambda_1 = \frac{502.5}{0.2311} = 0.000460$; $\lambda_2 = \frac{1502.5}{0.3005} = 0.000200$; $\lambda_3 = \frac{2502.5}{0.2202} = 0.000088$; and $\lambda_4 = \frac{3502.5}{0.2202} = 0.000043$. In particular the study used Equation 4.21 to compute the average waiting times for four priority queues for load, $\rho=0.6$ for the EEDF-II model. To do so, the study first computed the waiting probability of a packet from a given queue being in service at an arrival. For the M/M/m and M/G/m queues a simple yet good approximation of waiting probability is given by:

$$P_i \approx \begin{cases} \frac{\rho^m + \rho}{2} & \rho > 0.7 \\ \frac{\rho^{m-1}}{2} & \rho \leq 0.7 \end{cases} \quad (4.23)$$

Using condition 2 in Equation 4.23: $P_1=0.195^{\frac{1}{2}}=0.4416$; $P_2=0.215^{\frac{1}{2}}=0.4636$; $P_3=0.180^{\frac{1}{2}}=0.4226$; and $P_4=0.01^{\frac{1}{2}}=0.1$. The mean delay incurred by packets in service, for each queue are computed using Equation 4.4 as follows;

$$\begin{aligned} \bar{W}_1^0 &= P_1 \bar{R}_1 = 0.4416 \times \frac{335673}{2 \times 2 \times 502.5} = 73.748 \\ \bar{W}_2^0 &= P_1 \bar{R}_1 + P_2 \bar{R}_2 = 0.4416 \times \frac{335673}{2 \times 2 \times 502.5} + 0.4636 \times \frac{2340673}{2 \times 2 \times 1502.5} = 254.303 \\ \bar{W}_3^0 &= \bar{W}_2^0 + P_3 \bar{R}_3 = 0.4416 \times \frac{335673}{2 \times 2 \times 502.5} + 0.4636 \times \frac{2340673}{2 \times 2 \times 1502.5} + 0.4226 \times \frac{6345673}{2 \times 2 \times 2502.5} = 267.900 \\ \bar{W}_4^0 &= \bar{W}_3^0 + P_4 \bar{R}_4 = 0.4416 \times \frac{335673}{2 \times 2 \times 502.5} + 0.4636 \times \frac{2340673}{2 \times 2 \times 1502.5} + 0.4226 \times \frac{6345673}{2 \times 2 \times 2502.5} + \\ & \quad 0.1 \times \frac{12350673}{2 \times 2 \times 3502.5} = 356.056 \end{aligned}$$

Using Equation 4.16, the component $\frac{\bar{W}_i^0}{1-\sigma_i}$ for the packets in each queue is computed as follows:

$$\frac{\bar{W}_1^0}{1-\sigma_1} = \frac{73.748}{1-0.195} = 91.612$$

$$\frac{\bar{W}_2^0}{1 - \sigma_2} = \frac{254.303}{1 - 0.195 - 0.2150} = 431.022$$

$$\frac{\bar{W}_3^0}{1 - \sigma_3} = \frac{267.900}{1 - 0.195 - 0.2150} = 653.415$$

$$\frac{\bar{W}_4^0}{1 - \sigma_4} = \frac{356.056}{1 - 0.195 - 0.2150 - 0.180} = 840.141$$

The values got for the component $\frac{\bar{W}_i^0}{1 - \sigma_i}$ are fed directly into Equation 4.21 to calculate the individual waiting times as follows: This is done using an iterative process starting with the highest order of priority. Hence, for $i=1$.

$$\bar{W}_1 = \frac{\bar{W}_1^0}{1 - \sigma_1} = \frac{73.748}{1 - 0.195} = 91.612$$

For $i=2$

$$\bar{W}_2 = \frac{\bar{W}_2^0}{1 - \sigma_2} = \frac{254.303}{1 - 0.195 - 0.2150} = 431.022$$

For $i=3$

$$\bar{W}_3 = \frac{\bar{W}_3^0}{m(1 - \sigma_3)} + \rho_2 \min(\bar{W}_3, D_{3,2}) \quad 1st \quad Int.$$

$$\bar{W}_3 = 653.415 + 0.2 \min(653.415, 2000) = 784.098$$

2nd Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 784.098 = 810.235$$

3rd Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 810.235 = 815.462$$

4th Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 815.462 = 816.507$$

5th Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 816.507 = 816.716$$

6th Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 816.716 = 816.756$$

7th Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 816.756 = 816.767$$

8th Int.

$$\bar{W}_3 = 653.415 + 0.2 \times 816.767 = 816.768$$

For $i=4$

$$\bar{W}_4 = \frac{\bar{W}_4^0}{m(1 - \sigma_4)} + \rho_2 \min(\bar{W}_3, D_{3,2}) + \rho_3 \min(\bar{W}_4, D_{4,3})$$

1st Int.

$$\bar{W}_4 = 890.141 + 0.215 \min(816.768, 2000) + 0.18 \min(0, 3000) = 1065.746$$

2nd Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1065.746 = 1257.580$$

3rd Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1257.580 = 1292.110$$

4th Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1292.110 = 1298.326$$

5th Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1298.326 = 1299.445$$

6th Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1299.445 = 1299.682$$

7th Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1299.682 = 1298.689$$

8th Int.

$$\bar{W}_4 = 890.141 + 0.215 \times 816.768 + 0.18 \times 1299.689 = 1299.689$$

This process is repeated for $\rho=0.3, 0.45, 0.75$ and 0.9 to get the corresponding values of average waiting times.

4.6.2 Performance of the EEDF-I model

The study first looked at the performance of the EEDF-I model under the M/M/m queueing system. The goal here is to show the weakness of the EEDF-I in terms of penalizing low priority class packets in favor of higher priority packets. Only the delay performances of the EEDF-I for four priority classes are showed. More results of the EEDF-I performance when the algorithm is compared with the EEDF-II are presented in Section 4.6.3. Figure 4.2 shows the waiting time as function of total load. We observe that:

- i. $P1$ packets have a better performance overall among the all compared priority packets making the EEDF-I model to penalize low priority packet
- ii. Waiting time for $P3$ and $P4$ packets increases uniformly with increasing load
- iii. Waiting time for $P1$ and $P2$ packets increases uniformly with increasing load up to the 0.6 network load, beyond this load it drops up to 0.75 network load; and again it gradually increases till the end.

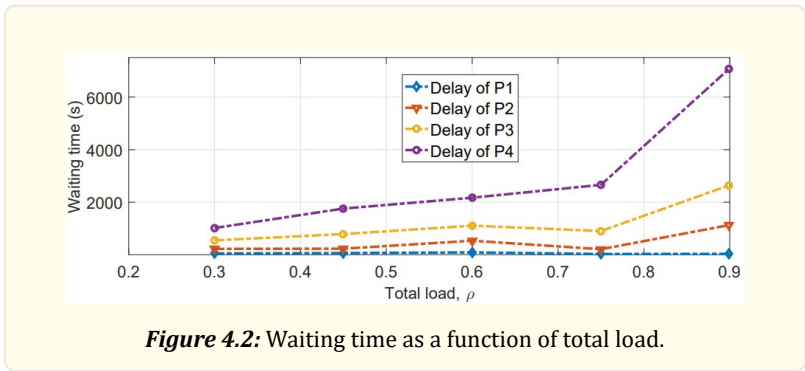


Figure 4.2: Waiting time as a function of total load.

4.6.3 Performance of the EEDF-I and EEDF-II models

The intention of this subsection 4.6.3 is to experiment the EEDF-II and benchmark it against the EEDF-I model in Section 4.4. The evaluation of the models was carried out using analytical methods. The main metrics measured are average waiting time. The waiting times of the four priority queues were computed for both models using an iterative process at system loads, $\rho = 0.3, 0.45, 0.6, 0.75$ and 0.9 . Table 4.2 shows the estimated waiting times the four priority queues for the EEDF-I and EEDF-II models. We assumed the same parameters for deadlines, deadline differences, service times and second moments for various system loads.

	EEDF-I	EEDF-I	EEDF-I	EEDF-I	EEDF-II	EEDF-II	EEDF-II	EEDF-II
Load	P1	P2	P3	P4	P1	P2	P3	P4
0.30	51.335	225.843	549.792	1019.489	51.335	207.775	500.911	876.030
0.45	64.232	232.496	788.239	1758.080	64.232	205.624	672.913	1354.04
0.60	91.612	535.430	1107.482	2171.020	91.612	431.022	816.768	1299.689
0.75	29.225	212.829	902.871	2661.403	29.225	170.263	663.700	1392.291
0.90	36.283	1127.756	2639.266	7067.499	36.283	258.811	1175.739	4324.818

Table 4.2: Waiting times for EEDFI and EEDF-II models- four priority queues.

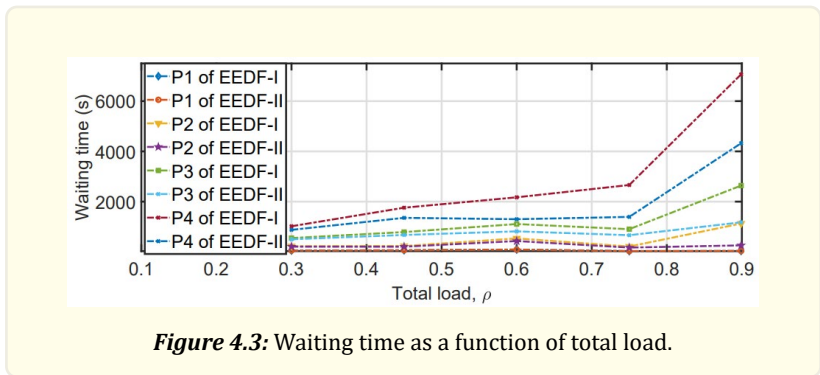


Figure 4.3: Waiting time as a function of total load.

4.6.4 Discussions of the results

Figure 4.3 is a variation of waiting time with total load for four priority queues of the EEDF-I and EEDF-II. Results obtained show that:

- i. The highest priority P1 packets had the lowest waiting times at all loads for the EEDF-I and EEDF-II models. Packets that arrive are serviced on first come first serve basis with minimal delay. This validates the claim that in multi-server queuing systems when the average service rate is more than average arrival rate then there are no or minimal delays.

- ii. The waiting time for $P2$, $P3$ and $P4$ packets increases with increasing total load in both the adopted and improved adopted Abhaya EDF models. This confirms fact that in multi-server queuing systems increasing total load, and when the number of packets in the system is more than or equal to the number of servers then all servers will be busy resulting into increasing longer waiting times. We observe closer rates of deterioration for EEDF-II, hence fairness and equitable scheme.
- iii. At any instant the waiting time for $P4$ is significantly higher than $P3$, $P2$ and $P1$ packets in the EEDF-I model; We further note that the EEDF-II model provides bigger relative improvements in waiting times for $P4$, $P3$ and $P2$ packets. This validates the our two claims that:
 - By EEDF-I favoring higher priority packets ends up increasing the waiting times of lower priority packets.
 - Low-priority queue packet starvation is avoided by the EEDF-II model.
- iv. At higher system loads, EEDF-II model provides higher improvements in waiting times for packets compared to EEDF-I. High system loads are associated with high arrival rates resulting into long delays for lower priority class packets.

The brief explanation why EEDF-II is giving a better performance basing on its scheduling set up is the component $\sum_{k=i+1}^N \rho_k \max(0, \bar{W}_k - D_{k,i})$ is removed to avoid excessive waiting time for low priority packets; and component $\sum_{k=1}^{i-1} \rho_k \min(\bar{W}_i(EEDF - I), D_{i+1,i})$ is changed to $\sum_{i=2}^{N-1} \rho_i \min(\bar{W}_i(EEDF - II), D_{i+1,i})$ to improve network performance.

4.7 Conclusion

In this Chapter, we adopted the generic EDF algorithm proposed by Abhaya; enhanced the adopted algorithm of Abhaya; did a performance evaluation of the algorithms at various system loads. The results revealed a better performance in terms of average waiting times for the enhanced (EEDF-II) compared to the adopted (EEDF-I) scheme.

Chapter 5 LLQ Scheduling in Manets

In this Chapter, we adopt the LLQ algorithm proposed by Kakuba et al; and improve on the adopted Kakuba LLQ model; and then carry out a performance evaluation of the algorithms at varying workloads. The mathematical notations and expressions are presented in Section 5.2. Section 5.3 presents the main assumptions followed in modelling the LLQ schemes. Section 5.4 describes the existing Kakuba LLQ model. In Section 5.5, we describe the adopted LLQ model, justify the improvements made; and highlight the changes to be made. Section 5.6 presents the proposed LLQ scheduling algorithm and indicate the improvements. The steps followed in the modelling of the proposed algorithm are discussed. Section 5.7 presents the performance evaluation of the algorithms. Section 5.8 concludes the Chapter.

The results have been published in: Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri (2023). An Improved Low Latency Queueing Scheduling Algorithm for MANETs. In: Arai, K. (eds) Advances in Information and Communication. FICC 2023. Springer, Cham. https://doi.org/10.1007/978-3-031-28076-4_9

5.1 Introduction

The LLQ algorithm is a queuing scheme that was developed by Cisco to bring strict PQ to CBWFQ [22]. The LLQ technique gives the opportunity to real-time traffic to receive service over other traffic by allowing the delay sensitive traffic to be dequeued and sent first. The LLQ scheme makes use of a strict priority queue as high priority while others are low queues, which makes it ideal for delay and jitter sensitive applications.

5.2 The mathematical notations and expressions

The study considered the following mathematical notations and expressions [51]: λ is the average arrival rate of all the packets to the queue; ρ_{vs} is load due to voice packets; ρ is the total load of all packets in the system; $T(x)$ is the conditional average response time; \bar{x}_L is the mean service time of a video packet; $W(x_s)$ is delay due to voice packet; $W(x_l)$ is delay due to video packet;

The other expressions for the performance metrics that were considered are: $f(x)$ is the probability density function (pdf) of the packet size X ; $F(x) = \int_0^x f(t)dt$ is the cumulative distribution function (cdf); and $F^c(x) = 1 - F(x)$ is the survival function. Define $\overline{x_x^n} = \int_0^x t^n f(t)dt$ to be the n^{th} moment of the packet size distribution with size less than or equal to x . ρ_L is load due to video packets. Let: $\overline{x_{x_S}}$ and $\overline{x_{x_S}^2}$ be the mean service time and the second moment of voice packets respectively; $\overline{x_{x_L}}$ and $\overline{x_{x_L}^2}$ be the mean service time and the second moment of the video packets respectively. It follows that $\overline{x_{x_L}^n} = \int_x^\infty t^n f(t)dt$ is the n^{th} moment of the packet size distribution with size greater than x . The load due to voice packets is $\rho_{x_S} = \lambda \int_0^x t f(t)dt$, while the load due to video packets is $\rho_{x_L} = \lambda \int_x^\infty t f(t)dt$, also $\rho_{x_L} = \rho - \rho_{x_S}$ where $\rho =$ is the total load in the system.

5.3 Assumptions

- i. The study assumes that the arrival process for each priority class is an independent Poisson process with rate λ_i and that the service times are exponentially distributed with mean $\frac{1}{\mu_i}$.
- ii. When two nodes are within a communication range, they can form a link and share data. Links (or connections) between stations are assumed error free.
- iii. The nodes are identical with battery life of 12 hours.
- iv. Although there is mobility, nodes may not go beyond the communication range of each other which would lead to link failure.
- v. Nodes are communicating through bidirectional links.
- vi. All nodes have constant and similar transmission range.
- vii. We assume that scheduling of packets at the destination can be handled without loss of information. Thus, a video packet may be split and its partial packets may be transferred over voice and video queues.

5.4 Existing LLQ model

In this model the authors employed queueing theory. They made use of Kendall's notation. For the benefit of our readers, we first describe the poisson processes. The processes are often used to model the number of arrivals over a given Interval for example number of packet-arrivals to a queue, number of queries to a database (over a time interval t). Assume the process shown in Figure 5.1 with arrivals at time t_1, t_2 and so on. If the inter-arrival times " $t_2 - t_1$ ", " $t_3 - t_2$ " and so on are IID and exponentially distributed, the number of arrivals over a given time interval has a poisson distribution and this is referred to as a poisson process. The work of A. K. Erlang in 1919 was one of the original applications of the poisson process in communications to model the arrivals of calls to a telephone exchange Inria [101]. The traffic was modelled using the MMPP/G/1 queue, where MMPP is Markov Modulated Poisson Process. Lee and Jeon [125] defines the MMPP model as doubly stochastic poisson process where arrival rate is a function of the state of a given continuous time Markov process. The MMPP model has been widely used in some works [126-128] to capture the typical characteristics of the incoming traffic such as self-similar behavior (correlated traffic), burstiness behavior, and long range dependency.

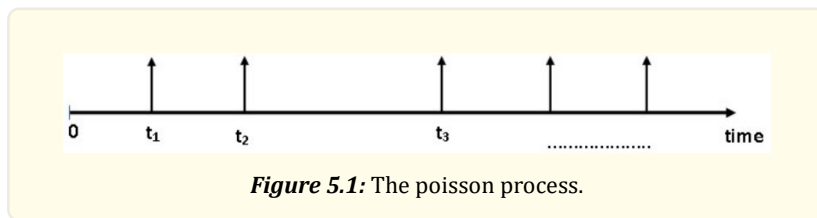


Figure 5.1: The poisson process.

Kakuba et al. [22] considered an MMPP/G/1 queue in the improved LLQ algorithm to schedule video and voice packets on downlinks of MS in MANETs by giving priority to voice packets first. The Kakuba LLQ model allows, all incoming traffic to a given arrival rate at the classifier, from outside the system following an MMPP or a poisson process. Real time voice and video packets are sent to the strict priority queue, while the other types of data are sent to the CBWFQ queue. We note that for this model even in strict priority queue,

during service, voice packets are prioritized over video. Therefore, the video packets are penalized while scheduling the voice packets. This happens at high loads, and get worse in highly overloaded system conditions.

While analyzing the behavior of voice and video packets in the Kakuba LLQ model, the traffic was modelled using the MMPP/G/1 queue under four conditions which lead to the derivation of expressions for average waiting times.

- a. The first expression was for voice packets under MMPP/M/1 queue system before delay.
- b. The second expression was for video packets under MMPP/M/1 queue system before delay.
- c. The third and fourth expressions plus their variants are of significant importance to our study, therefore it is appropriate to elaborate them further. The third expression was for voice packets under:
 - i. MMPP/M/1 queue system after being delayed once is given by;

$$E[W(x_S)] = \sum_{i=1}^2 x_i \frac{R}{(1-\rho_{x_S})} + \frac{\rho_{x_L} R}{(1-\rho_{x_S})^2(1-\rho_{x_S}-\rho_{x_L})} \quad (5.1)$$

- ii. MMPP/BP/1 queue system after being delayed once is given by;

$$E[W(x_S)] = \sum_{i=1}^2 x_i \frac{CoV^2 + 1}{2} \frac{R}{(1-\rho_{x_S})} + \frac{\rho_{x_L} R}{(1-\rho_{x_S})^2(1-\rho_{x_S}-\rho_{x_L})} \quad (5.2)$$

- d. The fourth expression was for video packets under:
 - i. MMPP/M/1 queue system after being delayed once is given by;

$$E[W(x_L)] = \sum_{i=1}^2 x_i \frac{R}{(1-\rho_{x_S})(1-\rho_{x_L})} \quad (5.3)$$

- ii. MMPP/BP/1 queue system after being delayed once is given by;

$$E[W(x_L)] = \sum_{i=1}^2 x_i \frac{CoV^2 + 1}{2} \frac{R}{(1-\rho_{x_S})(1-\rho_{x_L})} \quad (5.4)$$

5.5 The adopted LLQ algorithm

The study adopts Kakuba model and then improves the adopted Kakuba model. The following changes to the Kakuba et al. [22] algorithm are made:

- i. In the first adaption is a shift from the MMPP used to model self-similar behavior of the incoming traffic to where arrivals are Markovian (Poisson) arrival process because not all traffic transmitted over MANETs possess characteristics of burstiness behavior. In telecommunication, a burst transmission or data burst is the broadcast of a relatively high bandwidth transmission over a short period [129]. Burst transmission occurs as intentional, broadcasting a compressed message at a very high data signaling rate within a very short transmission time. Bursty traffic is uneven pattern of data transmission. Data rate changes in a very short period of time. This is the most difficult type of traffic to handle because the profile is very unpredictable and entails a low utilization of network resources for long times, but suddenly causes congestion in network buffers. Examples are HTTP, FTP downloads graphic, video content. Data burst can occur naturally, such as when the download of the data from the internet briefly experiences higher speeds. It can also occur in the computer network where data transmission is interrupted at intervals. The components namely: $\sum_{i=1}^2 x_i$ and $\sum_{i=1}^2 x_i \frac{CoV^2+1}{2}$ in Equations 5.1, 5.2, 5.3 and 5.4 were removed, where x_i is the weight of the delay in each state and CoV is the coefficient of variation of the service time distribution. Therefore, expression for the average waiting time of voice packets, $E[W(x_s)]$ after delaying once is given as;

$$E [W (x_S)] = \frac{R}{(1 - \rho_{x_S})} + \frac{\rho_{x_L} R}{(1 - \rho_{x_S})^2 (1 - \rho_{x_S} - \rho_{x_L})} \quad (5.5)$$

And the average waiting time of video packets, $E [W (x_L)]$ after delaying the voice packets once is given as;

$$E [W (x_L)] = \frac{R}{(1 - \rho_{x_S})(1 - \rho_{x_L})} \quad (5.6)$$

The average service time, R can also be represented in terms of the second moment and the average arrival rate of all the packets;

$$R = \frac{\lambda \overline{x^2}}{2} \quad (5.7)$$

- ii. The second change is the component R is represented in terms of the second moment and the average arrival rate of the packets. This change is done because the study utilizes the second moments and the average arrival rate of all the packets in the analysis for the exponential and BP distributions.

$$E [W (x_S)] = \left[\frac{\lambda \overline{x_S^2}}{2(1 - \rho_{x_S})} + \frac{\lambda \overline{x_S^2} \rho_{x_L}}{2(1 - \rho_{x_S})^2 (1 - \rho_{x_S} - \rho_{x_L})} \right] \quad (5.8)$$

$$E [W (x_L)] = \frac{\lambda \overline{x_L^2}}{2(1 - \rho_{x_S})(1 - \rho_{x_L})} \quad (5.9)$$

In the M/G/m queue because of the m servers, an arriving packet has to wait, on average for only the service of $E [W (x)]/m$ packets.

- iii. The third change introduces the component to represent the average time taken by the packet to be serviced. The result of ρ , the total load of all packets in the system divided by λ , the average arrival rate of all the packets gives us this component. Therefore, the expression for conditional mean response time of a voice packet of size x_S if a voice packet is delayed once in queue 1 be given by;

$$E [T (x_S)] = \frac{\rho}{\lambda} + \left[\frac{\lambda \overline{x_S^2}}{2m(1 - \rho_{x_S})} + \frac{\lambda \overline{x_S^2} \rho_{x_L}}{2m(1 - \rho_{x_S})^2 (1 - \rho_{x_S} - \rho_{x_L})} \right] \quad (5.10)$$

And the expression for conditional mean response time of a video packet of size x_L in queue 2 if a voice packet is delayed once is be given by;

$$E [T (x_L)] = \frac{\rho}{\lambda} + \frac{\lambda \overline{x_L^2}}{2m(1 - \rho_{x_S})(1 - \rho_{x_L})} \quad (5.11)$$

5.6 The proposed LLQ algorithm

The modelling of the LLQ algorithm goes through the following steps:

- i. The LLQ algorithms were analytically modelled with two or more queues using queuing theory.
- ii. The tagged packet technique was used to analyze the conditional mean response time by tracking the experience of a tagged arrival (voce video or text) to derive the models.
- iii. The expressions for the conditional mean response time under M/G/1/FCFS system were defined; and used these expressions to compute the conditional mean response time for packets for the different priority queues.

- iv. The study considered two job size distributions that is, exponential and BP in the analysis.
- v. The technique of splitting video packets while receiving service under the job size distributions was used in the design of the proposed LLQ algorithm.
- vi. The derived models were implemented in MATLAB to write the main function MATLAB-code that was used to generate values of packet sizes, loads, conditional mean response time and slowdown under varying workloads.
- vii. The derived LLQ models were used in the performance evaluation while comparing the adopted and the proposed LLQ algorithms in terms of conditional mean response time and slowdown .
- viii. The results were presented in form of graphic representation of conditional mean response time or slowdown Vs packet size or load. The discussion of the results followed each graph.

Given the natural packet size in data networks, it is possible to have data packets to be delayed more than once on the LLQ scheduler. When voice packets are encountered, the LLQ scheduler serves them un-interrupted until service is completed. Consequently, depending on the workload distribution, video packets may experience considerable delays at the expense of favoring voice packets. Therefore, we propose the technique of splitting video packets while receiving service. We assume data packets can be split, luckily enough, voice and video packets can be split. We use the Pollaczek-KhinChine (PK) formula, where the mean waiting time voice packet of size x_s is given by;

$$E [W (x_s)] = \frac{\overline{\lambda x_s^2}}{2(1 - \rho_{x_s})} \quad (5.12)$$

When a video packet of size x_l is encountered in any queue, it is split into partial packets $x_{pr1}, x_{pr2}, x_{pr3}, \dots, x_{pri}$. Then the first partial video is serviced in queue 1 following the earlier arrivals of the voice packets then the other partial video packets are serviced in queue 2 to completion. Hence, the expression for mean waiting time of the first partial video packet in queue 1;

$$E [W (x_{pr1})] = \frac{\overline{\lambda x_{pr1}^2}}{2(1 - \rho_{pr1})} \quad (5.13)$$

And the expression for mean waiting time of the other partial video packets, $x_{pr1}, x_{pr2}, x_{pr3}$ to x_{pri} in queue 2;

$$E [W (x_{pri})] = \frac{\overline{\lambda x_{pr2}^2}}{2(1 - \rho_{pr2})} + \frac{\overline{\lambda x_{pr3}^2}}{2(1 - \rho_{pr3})} \dots + \frac{\overline{\lambda x_{pri}^2}}{2(1 - \rho_{pri})} \quad (5.14)$$

$$= \int_2^i \frac{\overline{\lambda x_{pr_i}^2} dx}{2(1 - \rho_{pr_i})} \quad (5.15)$$

Assuming a tagged voice packet of size x_s arriving at queue 1. This voice packet will be delayed by: mean residual time of the packets found in service, the mean waiting time of the voice packets found in the queue, the mean waiting time of first partial video packet found in the queue. Recall that an M/G/m queue and also MANETS there m servers, hence; The expression for conditional mean response time of a voice packet of size x_s in queue 1 if a voice packet is delayed once is given by;

$$E [T (x_s)] = \frac{\rho}{\lambda} + \frac{\overline{\lambda x_s^2}}{2m(1 - \rho_{x_s})} + \frac{\overline{\lambda x_{pr1}^2}}{2m(1 - \rho_{pr1})} \quad (5.16)$$

Assuming a tagged video packet of size x_l arriving at queue 2. This video packet will be delayed by: mean residual time of the packets found in service and the mean waiting time of the other partial video packets found in the queue. Therefore the expression for conditional mean response time of a video packet of size x_l in queue 2 if a voice packet is delayed once is be given by;

$$E [T(x_L)] = \frac{\rho}{\lambda} + \int_2^i \frac{\lambda \overline{x_{pr_i}^2} dx}{2m(1 - \rho_{pr_i})} \quad (5.17)$$

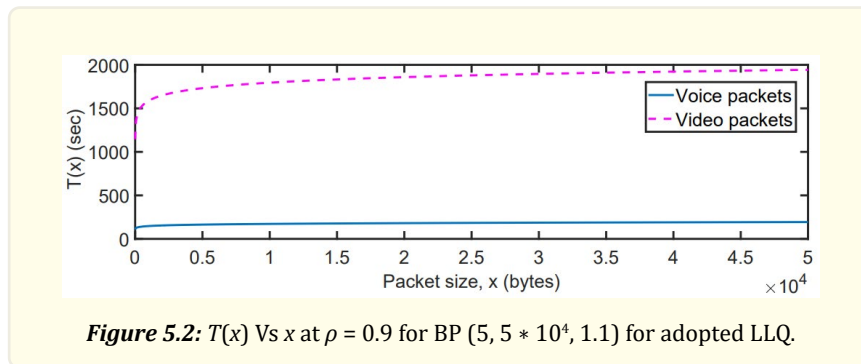
5.7 The performance evaluation

5.7.1 Experimental set-up

The study used the LLQ scheduling algorithms developed in Sections 5.5 & 5.6 to obtain the MATLAB codes for the analysis. The exponential and BP distributions were used to generate the workloads, and the number of iterations used were 300, 0000. Table 5.1 shows the experiment set-up configuration for the study. The study considered exponentially distributed packet sizes to see the performance difference when LLQ is analyzed under M/G/1 queue model. The *CoV* value of an exponential distribution is 1. To obtain different values of *CoV* for the $BP(k, L, \alpha)$, one or more of the parameters of the BP distribution was changed. The parameters were chosen for illustration purposes otherwise these can be varied depending on the user requests. The corresponding values of conditional mean response times and slowdowns were obtained and plotted against the packet sizes as shown next. The study experimentally set up the algorithm under both exponential and BP distributions. Our goal here is to show the weakness of this algorithm in terms of penalizing large video packets for workloads that have highly varying sizes.

Parameter	Values
Average packet arrival rate, λ for exp	$\frac{1}{2000}$
Average packet arrival rate, λ for BP	0.0124
C values for BP	2, 6, 20, 284, 288
System Loads, ρ	0.5, 0.75, 0.9
Range of values for x BP	x=10:0.01:300,000;
Range of values for x exp	x =200:0.1:300,000;
$BP(k, L, \alpha)$	$BP(10, 5000, 1.1)$
Threshold value, x_{th} for BP	1526.7 bytes
Threshold value, x_{th} for exp	2500 bytes

Table 5.1: Experiment configuration set-up.



5.7.2 Results of the adopted LLQ algorithm

The study evaluated the performance of the adopted LLQ algorithm using the performance metrics of the mean conditional response time and mean conditional slowdowns. The adopted LLQ model were implemented in MATLAB to evaluate its performance.

The study only shows conditional mean response time and conditional mean slowdown performances of the algorithm at $\rho = 0.9$. More results of the adopted algorithm performance when compared with the improved LLQ are presented in Section 5.6. Now the

results of the adopted LLQ algorithm under BP distribution are presented. The study evaluates the conditional mean response time incurred by packets along a node in the MANET. Figure 5.2 shows the conditional mean response time for BP ($5, 5 * 10^4, 1.1$) as a function of packet size for the adopted LLQ scheduling algorithm at $\rho = 0.9$. Observe that the conditional mean response time gently increases for all the packet sizes. Voice packets have a small conditional mean response time in comparison to video packets at high system loads. The results clearly show that video packets are penalized at the expense of voice packets for heavy tailed workloads. In MANETs transmission of packets is via nodes and packets experience delays at the nodes. When the delay at a given node is small then the conditional mean response time is also small while the delay is big then the conditional mean response time is large. For the same packet size video packets perform worse than voice packets. This can be explained from the structure of video packets that is to say it has one part voice and the other is graphics. The structure of a packet comprises of packet type and protocol. When data transferred over the internet it is broken down into IP packets, which can be voice and video [4]. Transmitting a video packet requires disassembling the packet, sending the voice and graphics parts sequentially then re-assembling the packet at the end of transmission. The voice part is held in the buffers waiting for the graphics to be sent then re-synchronized. This process consumes more resources in terms of bandwidth and results into excessive delays in transmission of the video packets hence higher conditional mean response time for the video packets in comparison to the voice packets Figure 5.3 shows the conditional mean slowdown time for BP ($5, 5 * 10^4, 1.1$) as a function of packet size for the adopted LLQ scheduling algorithm at $\rho = 0.9$. Observe that the conditional mean slowdown decreases with increase in packet size. Further observe that voice packets have a small conditional mean slowdown in comparison to video packets at high system load. We again note that video packets are penalized at the expense of voice packets under the BP distribution. Recall, the definition of conditional mean slowdown, is the conditional mean response time divided by the size of that packet. Obviously, the trend for graphs of conditional mean slowdown time is expected to follow a similar pattern to that of conditional mean response time. The explanation for this trend is again related to the packet structure of voice and video packets. The results of the adopted LLQ algorithm for the conditional mean response time for the exponentially distributed workloads as a function of packet size at $\rho = 0.9$ are shown in Figure 5.4. We again observe that the conditional mean response time gently increases for all the packet sizes. We further note that voice packets have a small conditional mean response time in comparison to video packets at high system load. Again, video packets are penalized at the expense of voice packets. The explanation for this trend is again related to the packet structure of voice and video packets. While transmitting voice and video packets in MANETs there is de-assembling and re-assembling of packets at the end of the transmission. Video packets consume more resources in terms of bandwidth and require more time to re-assemble. The difference in separation graphs of voice and video packets in Figure 5.4 is narrow compared to that in Figure 5.3. The explanation for this is that exponentially distributed packet sizes have low CoV values while the BP packet sizes have high CoV values for large L values. In Figure 5.5, we present the results of the adopted LLQ algorithm for the conditional mean slowdown time for the exponentially distributed workloads as a function of packet size at $\rho = 0.9$. We again observe that video packets are penalized at the expense of voice packets. The trend for graphs of conditional mean slowdown time rises then gradually slope with increase in packet size. The explanation for this trend is again related to the packet structure of voice and video packets.

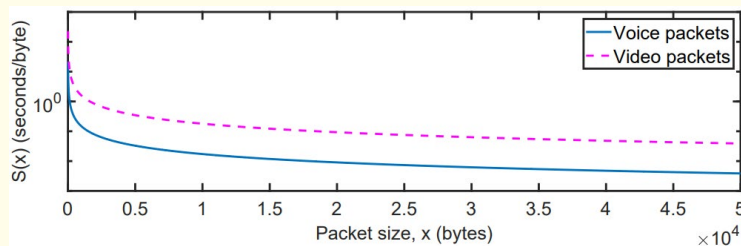


Figure 5.3: $S(x)$ Vs x at $\rho = 0.9$ for BP ($5, 5 * 10^4, 1.1$) for adopted LLQ.

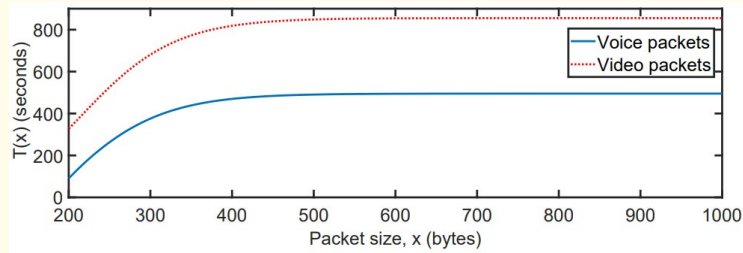


Figure 5.4: $T(x)$ Vs x at $\rho = 0.9$ for voice and video packets.

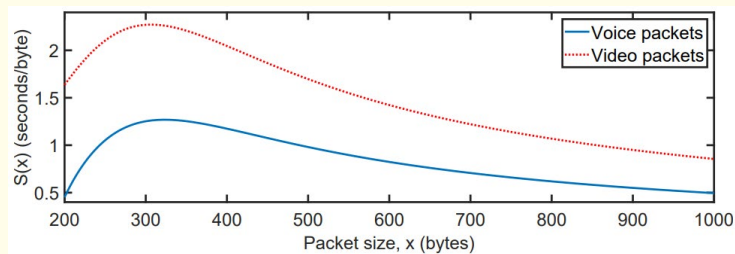


Figure 5.5: $S(x)$ Vs x at $\rho = 0.9$ for voice and video packets.

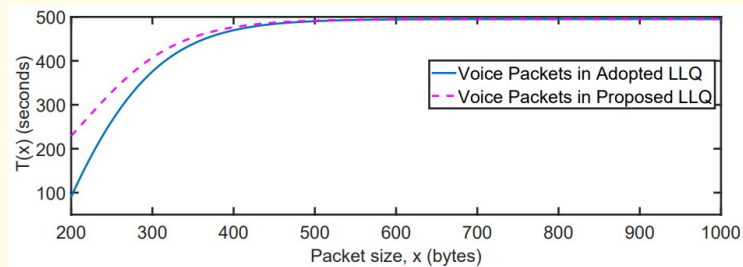


Figure 5.6: $T(x)$ Vs x at $\rho = 0.9$ for exp workloads.

5.7.3 Performance under exponential distribution

Figure 5.6 presents the results of the adopted and proposed LLQ algorithms for the conditional mean response time for the exponential workloads as a function of packet size at $\rho = 0.9$ for voice packets. We further observe that an improvement in video leads to decline in voice. Recall, the proposed algorithm utilizes the technique of splitting video where a partial video packet is transmitted along-side with the voice packets. While the added partial video packet is being transmitted, there is some added delay for the voice packets and a decline in their performance in the proposed algorithm.

Figure 5.7 presents the results of the adopted and proposed LLQ algorithms for the conditional mean slowdown time for the exponential workloads as a function of packet size at $\rho = 0.9$ for voice packets. We again note that the proposed algorithm performs poorer

than the adopted in transmitting voice packets. The explanation for this is splitting the video packets and transmits the partial video packet along-side with the voice packet in the proposed algorithm is responsible for the degradation in performance. Figure 5.8 presents the results of the adopted and proposed LLQ algorithms for the conditional mean response time for the exponential workloads as a function of packet size at $\rho = 0.9$. We observe that the adopted algorithm performs poorer than the proposed in transmitting video packets. Recall, the proposed algorithm utilizes the technique of splitting video where a partial video packet is transmitted along-side with the voice packets. While the added partial video packet is being transmitted, there is delay created for the voice packets and performance gain on the side of video packets in the proposed algorithm. The performance gain in the proposed algorithm shows significant contribution that is to say, in this study, we improve the performance of video packets while voice packets suffer some form of performance degradation. Figure 5.9 presents the results of the adopted and proposed LLQ algorithms for the conditional mean slowdown time for the exponential workloads as a function of packet size at $\rho = 0.9$. We note that the adopted algorithm performs poorer than the proposed in transmitting video packets. The explanation for this is splitting the video packets and transmits the partial video packet along-side with the voice packet in the proposed algorithm is responsible for improved performance. We can rightly conclude that the proposed LLQ algorithm is a more suitable than the adopted LLQ algorithm in scheduling video packets.

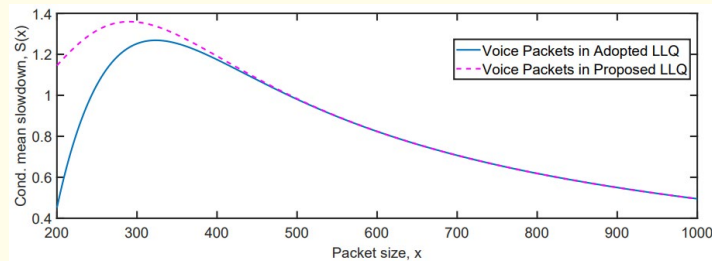


Figure 5.7: $S(x)$ Vs x at $\rho = 0.9$ under adopted & proposed LLQ.

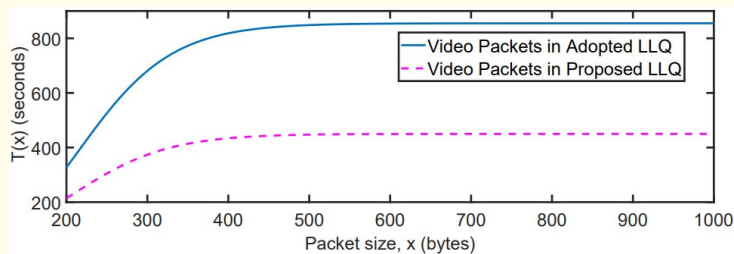


Figure 5.8: $T(x)$ Vs x at $\rho = 0.9$ under adopted & proposed LLQ.

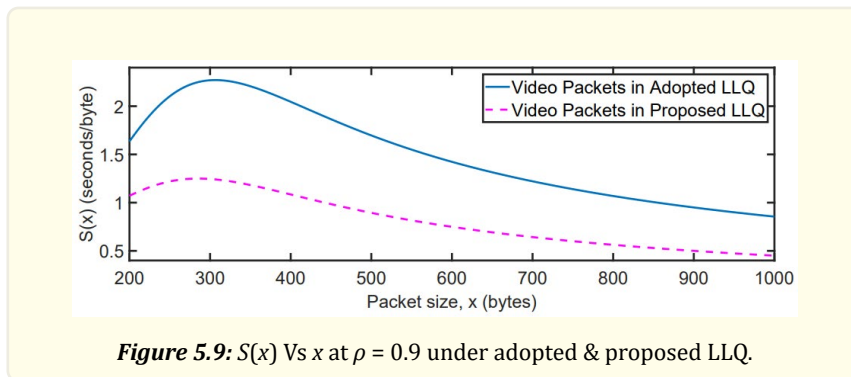


Figure 5.9: $S(x)$ Vs x at $\rho = 0.9$ under adopted & proposed LLQ.

5.7.4 Performance under BP distribution

In Figure 5.10, we present the results of the proposed and adopted LLQ algorithms for the conditional mean response time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ for voice packets. We observe that an improvement in video leads to decline in voice performance. This is as a result of the technique of splitting the video packet that is utilized by the proposed LLQ algorithm where one part of the video packet is transmitted along-side with the voice packets. This added partial video packet creates some form of delay for the voice packets hence the decline in performance of voice packets in the proposed algorithm. The results of the adopted and proposed LLQ algorithms for the conditional mean slowdown time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ are presented in Figure 5.11 for voice packets. We again observe that an improvement in video leads to decline in voice. The explanation is that the proposed LLQ algorithm utilizes the technique of splitting the video packets and transmits the partial video packet along-side with the voice packets, hence some form of added delay is created for the voice packets and this leads to the decline in performance of voice in the proposed algorithm. In Figure 5.12, we present the results of the adopted and proposed LLQ algorithms for the conditional mean response time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ for video packets. We observe that the adopted algorithm performs poorer than the proposed in transmitting video packets. This is as a result of the technique of splitting the video packet that is utilized by the proposed LLQ algorithm where one part of the video packet is transmitted along-side with the voice packets. This added partial video packet creates delay for the voice packets while enhancing the performance of video in the proposed algorithm. The results of the adopted and proposed LLQ algorithms for the conditional mean slowdown time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ are presented in Figure 5.13. Observe that the adopted algorithm performs worse than the proposed in transmitting video packets. The explanation is that the proposed LLQ algorithm utilizes the technique of splitting the video packets and transmits the partial video packet along-side with the voice packets, hence some form of added delay is created for the voice packets while enhancing the performance of video in the proposed algorithm.

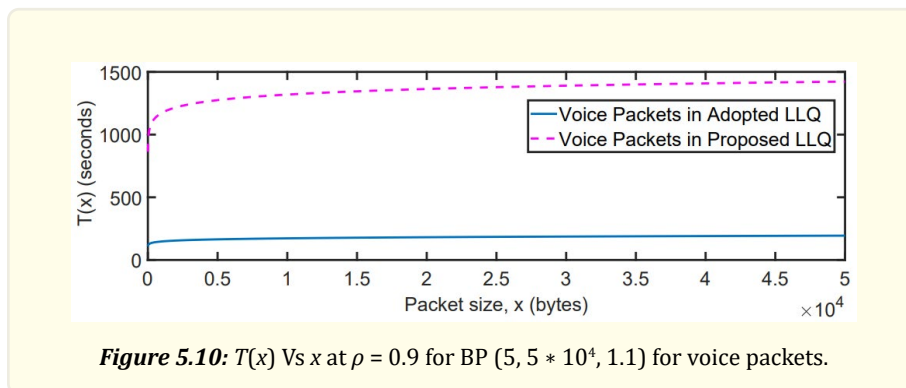


Figure 5.10: $T(x)$ Vs x at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$ for voice packets.

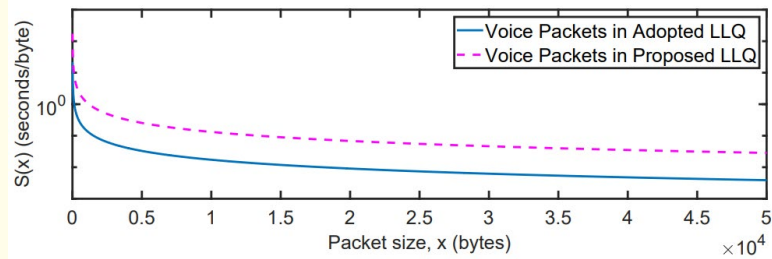


Figure 5.11: $S(x)$ Vs x at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$ for voice packets.

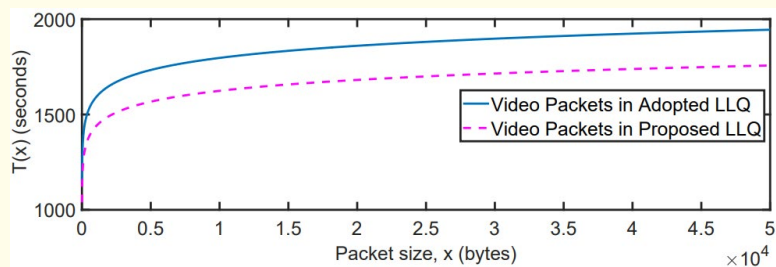


Figure 5.12: $T(x)$ Vs x at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$ for video packets.

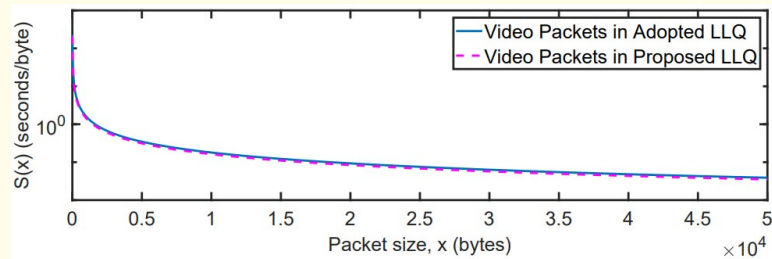


Figure 5.13: $S(x)$ Vs x at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$ for video packets.

5.8 Conclusion

In this Chapter, we developed and derived the LLQ models, discussed the scheduling of traffic (voice and video) and presented the results for LLQ scheduling algorithms. We adopted the LLQ algorithm from an MMPP/G/1 queue to an M/G/1 queue system. We proposed an improved LLQ scheduling algorithm based on the adopted an LLQ algorithm. Experiments were conducted to compare the performance of the adopted and proposed LLQ algorithms under varying workloads. The results revealed that the proposed LLQ algorithm performed better than the adopted LLQ algorithm in terms of conditional mean response times and slowdowns of video packets.

Chapter 6 Extended LLQ Scheduling in Manets

In this Chapter, we extend the work on the proposed LLQ scheduling algorithm to a three-priority queue (consisting of voice, video and text packets) model. In Section 6.1, we provide a brief introduction of the Chapter. Section 6.2 describes the design and development of the ELLQ algorithm. The results and discussions are presented in Section 6.3. Then Section 6.4 concludes the Chapter.

The results have been published in: Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri "The Performance Analysis of Low Latency Queueing Scheduling Algorithm for MANETs". *PriMera Scientific Engineering* 2.6 (2023): 03-12. <https://doi.org/10.56831/PSEN-02-054>

6.1 Introduction

In the previous Chapter 5, the adopted and proposed LLQ algorithms considered two priority queues. However, in MANETs and even other networks, network traffic consists of more than two classes of traffic like text, http, ftp e-mails and so on. Therefore, because of that research gap, we are motivated to extend the LLQ scheme to multiple queues and analyze the performance of algorithms under varying workloads. We extend and improve the proposed LLQ algorithm to formulate the Extended Low Latency Queueing Algorithm (ELLQ). We considered the mathematical notations as presented in Section 5.2 except we introduced some new notations like $T(x_p)$ for the conditional mean response time of text packets and CoV is the Coefficient of Variation. Also, the two service distributions as presented in Section 3.2 were considered in the analysis of the ELLQ algorithm. We introduced a new optimization technique for computing the size of the partial video packet to be transmitted alongside voice packets in queue 1 in order to fully maximize system utilization.

6.2 The ELLQ model

Network traffic is usually classified as voice, video and text. Assume an LLQ model with three queues that is to say queue 1 is for voice packets, queue 2 is for video packets and queue 3 is for text packets as represented in Figure 6.1. The rationale for the choice of three queues is for demonstration purposes otherwise one can extend the LLQ model beyond this specified number.

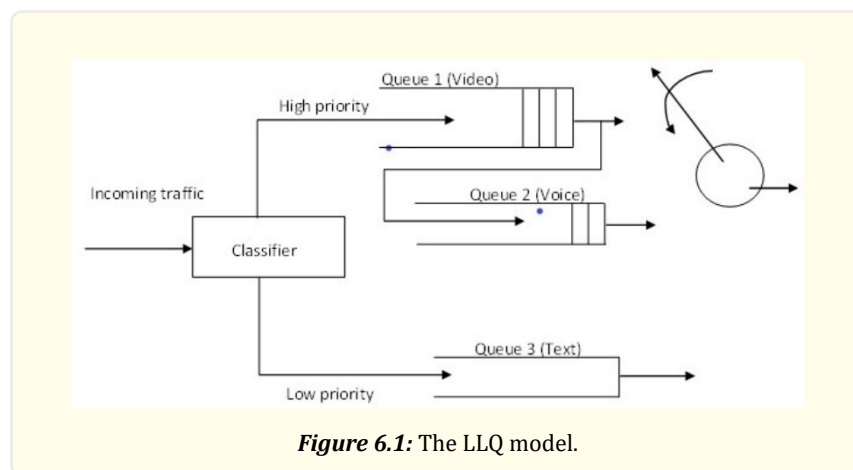


Figure 6.1: The LLQ model.

Scenario i: When voice packet is delayed once

We begin from a novel point of view with the expressions of conditional mean response time for voice and video packets [130]. Using the Pollaczek-KhinChine (PK) formula the mean waiting time of the text packet is given by;

$$W(x_D) = \frac{\overline{\lambda x_D^2}}{2(1 - \rho_{x_D})} \quad (6.1)$$

The change indicated in the expressions for conditional mean response times voice, video and text packets is the introduction of the component for the *CoV*. The squared coefficient of variation is used to quantify the variability of a distribution.

$$CoV^2[X] = \frac{Var[X]}{E[X]^2} \quad (6.2)$$

The squared coefficient of variability is a normalized measure of variability under which $CoV^2[X] = 1$, when X is an exponential random variable [131]. Thus, all distributions with $CoV^2 < 1$ are less exhibit varying characteristics than the exponential and all distributions with $CoV^2 > 1$ are exhibit more varying characteristics than the exponential distribution. Assuming a tagged voice packet of size x_s arriving at queue 1. This voice packet will be delayed by: mean residual time of the packets found in service, the mean waiting time of the voice packets found in the queue, the mean waiting time of first partial video packet found in the queue. Recall that an M/G/m queue and also MANETs there m servers, hence; The expression for conditional mean response time of a voice packet of size x_s in queue 1 if a voice packet is delayed once and piggy backed with video on transmission is given by;

$$T(x_S) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\frac{\overline{\lambda x_S^2}}{2m(1 - \rho_{x_S})} + \frac{\overline{\lambda x_{pr_1}^2}}{2m(1 - \rho_{pr_1})} \right] \quad (6.3)$$

In the simplest understanding we use the term piggybacking to refer to a process where a voice packet transmitted by any node consists of at least its own state information and a header but also includes information of the partial video packet. Assuming a tagged video packet of size x_L arriving at queue 2. This video packet will be delayed by: mean residual time of the packets found in service and the mean waiting time of the other partial video packets found in the queue. Therefore the expression for conditional mean response time of a video packet of size x_L in queue 2 if a voice packet is delayed once is given by;

$$T(x_L) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\int_2^i \frac{\overline{\lambda x_{pr_i}^2} dx}{2m(1 - \rho_{pr_i})} \right] \quad (6.4)$$

Assuming a tagged text packet of size x_D arriving at queue 3. This data packet will be delayed by: mean residual time of the packets found in service, the mean waiting time of the voice packets found in the queue and the mean waiting time of the video packets found in the queue. Therefore the expression for conditional mean response time of text packet of size x_D in queue 3 if a voice packet is delayed once is given by;

$$T(x_D) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\frac{\overline{\lambda x_S^2}}{2m(1 - \rho_{x_S})} + \frac{\overline{\lambda x_L^2}}{2m(1 - \rho_{x_L})} \right] \quad (6.5)$$

Equations 6.3, 6.4 and 6.5 represent the ELLQ algorithm with three queues under the first scenario.

Scenario ii: When voice packet is delayed only if there is a partial video packet being transmitted

Assuming a tagged voice packet of size x_S arriving at queue 1. This voice packet will be delayed by: mean residual time of the packets found in service and the mean waiting time of first partial video packet found in the queue. The expression for conditional mean response time of a voice packet of size x_S in queue 1 if a voice packet is delayed only if there is a partial video packet being transmitted is given by;

$$T(x_S) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\frac{\overline{\lambda x_{pr_1}^2}}{2m(1 - \rho_{pr_1})} \right] \quad (6.6)$$

Assuming a tagged video packet of size x_L arriving at queue 2. This video packet will be delayed by: mean residual time of the packets found in service, the mean waiting time of the voice packets found in the queue and the mean waiting time of the other partial video packets found in the queue. Therefore the expression for conditional mean response time of a video packet of size x_L in queue 2 if a voice packet is delayed only if there is a partial video packet being transmitted is given by

$$T(x_L) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\frac{\overline{\lambda x_S^2}}{2m(1 - \rho_{xS})} + \int_2^i \frac{\overline{\lambda x_{pr_i}^2} dx}{2m(1 - \rho_{pr_i})} \right] \quad (6.7)$$

Assuming a tagged text packet of size x_D arriving at queue 3. This data traffic packet will be delayed by: mean residual time of the packets found in service, the mean waiting time of the voice packets found in the queue and the mean waiting time of the video packets found in the queue and the mean waiting time of the text packets found in the queue. Therefore the expression for conditional mean response time of text packet of size x_D in queue 3 if a voice packet is delayed only if there is a partial video packet being transmitted is given by;

$$T(x_D) = \frac{\rho}{\lambda} + \frac{CoV^2 + 1}{2} \left[\frac{\overline{\lambda x_S^2}}{2m(1 - \rho_{xS})} + \frac{\overline{\lambda x_L^2}}{2m(1 - \rho_{xL})} + \frac{\overline{\lambda x_F^2}}{2m(1 - \rho_{xD})} \right] \quad (6.8)$$

Equations 6.6, 6.7 and 6.8 represent the LLQ algorithm with three queues under the second scenario.

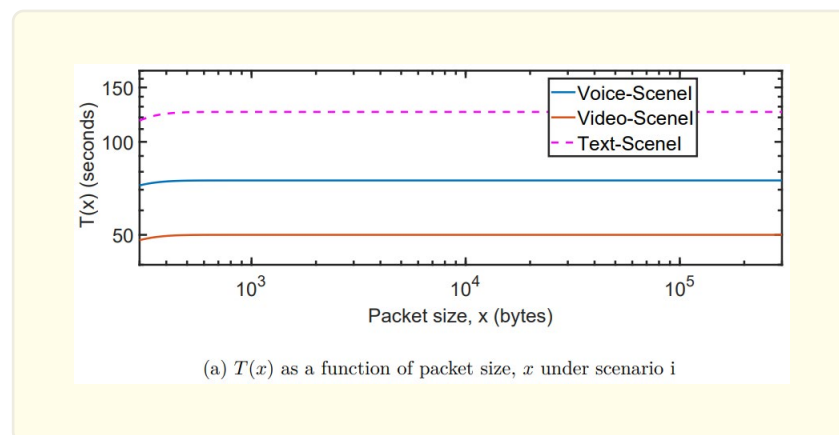
6.3 Results and Discussions

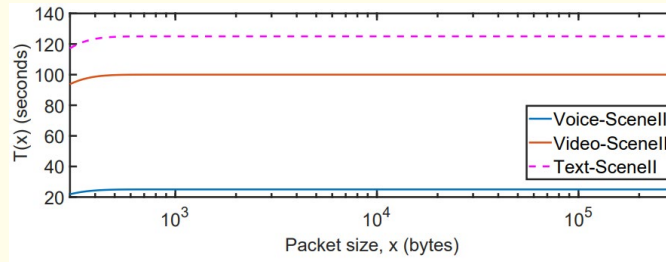
We evaluate the performance of ELLQ scheduling algorithm under exponentially and heavy tailed distributed workloads for different values of load. The ELLQ is analyzed under the experimental conditions as in Section 5.7. We carry out a comparative performance analysis to show which traffic class performs best under the two scenarios while being scheduled by the ELLQ algorithm. We show the result for high load, $\rho = 0.9$ in addition to the results for low load, $\rho = 0.5$ to complete the comparison exercise. The CoV value of an exponential distribution is 1. To obtain different values of CoV for the BP (k, L, α) we changed one or more of the parameters of the BP distribution. The parameters were chosen for illustration purposes otherwise these can be varied depending on the user requests. The corresponding values of conditional mean response times and slowdowns were obtained and plotted against the packet sizes as shown next.

6.3.1 Analysis under exponentially distributed workloads

We study the performance of the algorithm under scenarios i and ii for the exponential distribution. In this study, we write the terms ScenI and ScenII to refer to the short forms of scenarios i and ii respectively in the legends of the graphs. Figure 6.2 shows the $T(x)$ performance for ELLQ under exponentially distributed workloads at $\rho = 0.5$. Figures 6.2(a) and 6.2(b) show the $T(x)$ performance of voice, video and text packet as a function of packet size, x . We observe that the conditional mean response time increases with increase in packet size for all the three traffic types (voice, video and text packets). Video packets experienced the best performance and text had the worst performance. Figures 6.3(a) and 6.3(b) show the performance of voice, video and text packet for the ELLQ scheduling algorithm in terms of the conditional mean response time, $T(x)$ as a function of packet size, x under the exponential distribution. We note that the conditional mean response time increases with increase in packet size for all the three traffic types (voice, video and text packets). This result confirms what earlier studies [131] indicated that the conditional mean response time, $T(x)$ of a packet depends on the size of a packet. $T(x)$ grows linearly with x , since the response time of a packet includes (at minimum) the packet size. Voice packets registered the best performance, followed by video and again text packet was the least. The explanation for this trend is that when the partial video packet is being transmitted, there is delay created for the voice packets and performance gain for video packets. The result is spontaneous in that a delay in transmitting voice packets triggers a delay for data packets. For scenario ii: voice packet is delayed only if there is a partial video packet being transmitted; voice packets registered the best performance, followed by video and again text packets were the least. Figures 6.4(a) and 6.4(b) show the performance of voice, video and data text packet for the E

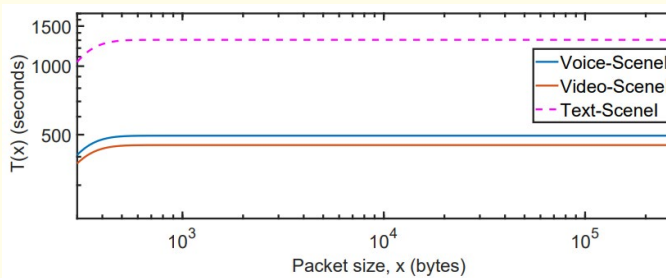
LLQ scheduling algorithm in terms of the conditional mean response time, $T(x)$ as a function of load, ρ when $x = 5000$ bytes under the exponential distribution. We observe that the conditional mean response time increases with increase in load for all the three traffic types (voice, video & text packets). For scenario i: When voice packet is delayed once and piggy backed with video on transmission; We can also observe from the figure that the ELLQ algorithm performs much better for video packets compared to voice and text packets regardless of the load. For scenario ii: When voice packet is delayed only if there is a partial video packet being transmitted; It can be observed that voice packets experience lower conditional mean response time under ELLQ algorithm. Video packets experience a slightly higher conditional mean response time, and text packets experience the highest conditional mean response time under ELLQ algorithm. A similar trend is observed when the packet size $x = 20000$ bytes under the exponential distribution as shown in Figures 6.5(a) and 6.5(b). Figure 6.6 shows the $S(x)$ performance for ELLQ under exponentially distributed workloads at $\rho = 0.5$. Figures 6.6(a) and 6.6(b) show the $S(x)$ performance of voice, video and text packets as a function of packet size, x . We observe video packets registered the lowest $S(x)$ followed by voice and text had the highest. Figures 6.7(a) and 6.7(b) show the performance of voice, video and text packets for the ELLQ scheduling algorithm in terms of the conditional mean slowdown, $S(x)$ as a function of packet size, x under the exponential distribution. We note that the conditional mean slowdown decreases with increase in packet size for all the three traffic types (voice, video and text packets). Just like Wierman [131], we use the figures showing ratios $S(x) = T(x)/x$ in order to contrast the behavior of conditional response time across different packet sizes. This is a useful measure because, in many cases, it is appropriate for response times to be proportional to job size (that is to say small packets should have small response times and large packets should have large response times). For scenario i: when voice packet is delayed once and piggy backed with video on transmission; we can also observe from Figure 6.7 that the ELLQ algorithm performs much better for video packets compared to voice and text packets. For scenario ii: when voice packet is delayed only if there is a partial video packet being transmitted; it can be observed that voice packets experience lower conditional mean slowdown under ELLQ algorithm. Video packets experience a slightly higher conditional mean slowdown, and text packets experience the highest conditional mean slowdown under ELLQ algorithm. Figures 6.8(a) and 6.8(b) show the performance of voice, video and text packets for the ELLQ scheduling algorithm in terms of the conditional mean slowdown, $S(x)$ as a function of load, ρ when $x = 5000$ bytes under the exponential distribution. For all the three traffic types (voice, video and text packets), we note that the conditional mean slowdown increases with increase in load. For scenario i: when voice packet is delayed once and piggy backed with video on transmission; from the figure it is observed that the ELLQ algorithm performs much better for video packets compared to voice and text packets regardless of the load. For scenario ii: when voice packet is delayed only if there is a partial video packet being transmitted; it can be observed that voice packets experience lower conditional mean slowdown under ELLQ algorithm. Video packets experience a slightly higher conditional mean slowdown, and text packets experience the highest conditional mean slowdown under ELLQ algorithm. A similar trend is observed when the packet size $x = 20000$ bytes under the exponential distribution as shown in Figures 6.9(a) and 6.9(b).



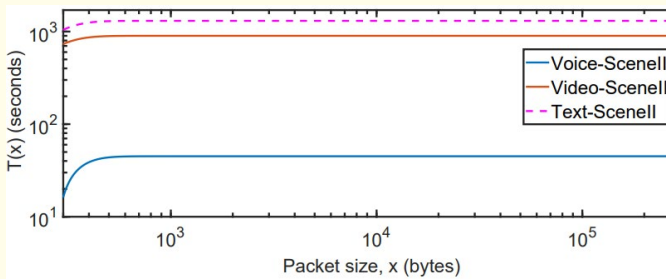


(b) $T(x)$ as a function of packet size, x under scenario ii

Figure 6.2: $T(x)$ performance for ELLQ at $\rho = 0.5$ for exp distribution.



(a) $T(x)$ as a function of packet size, x under scenario i



(b) $T(x)$ as a function of packet size, x under scenario ii

Figure 6.3: $T(x)$ performance for ELLQ at $\rho = 0.9$ for exp distribution.

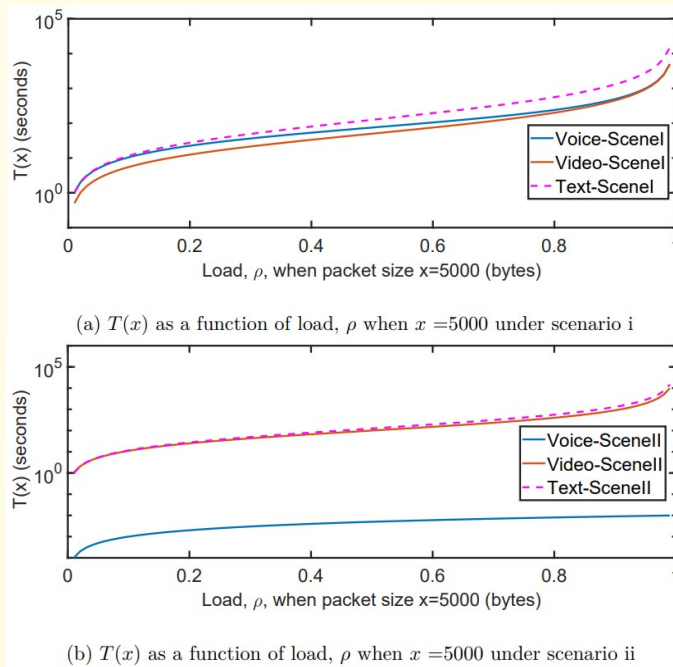


Figure 6.4: $T(x)$ performance for ELLQ at $\rho = 0.9$ for exp.

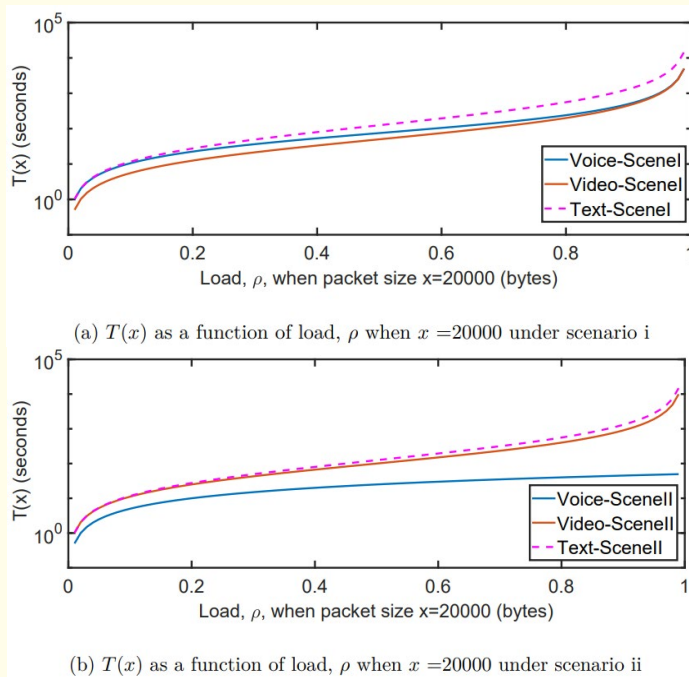
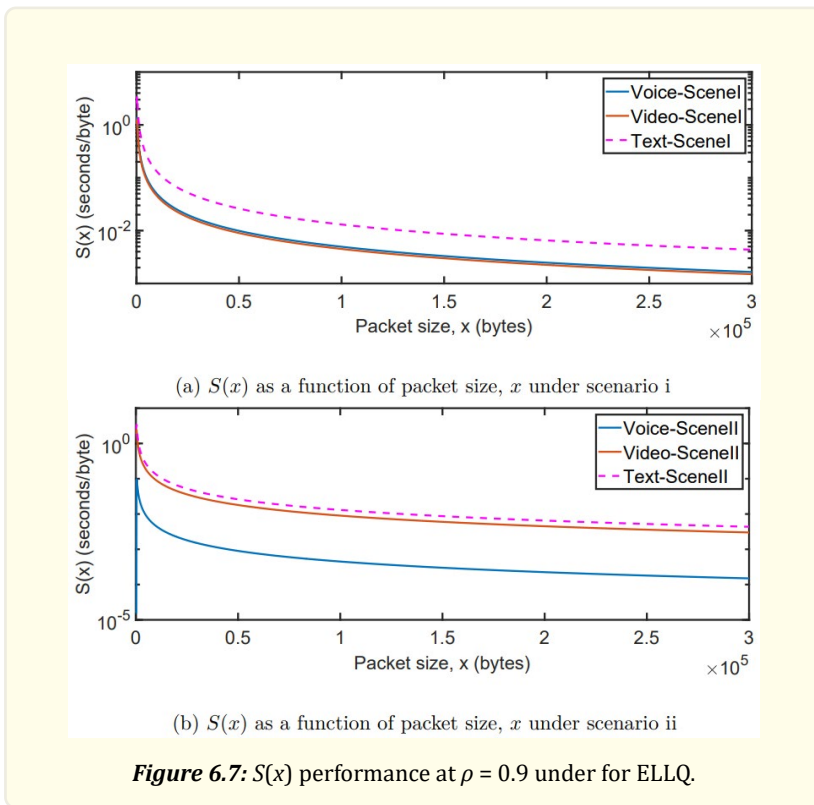
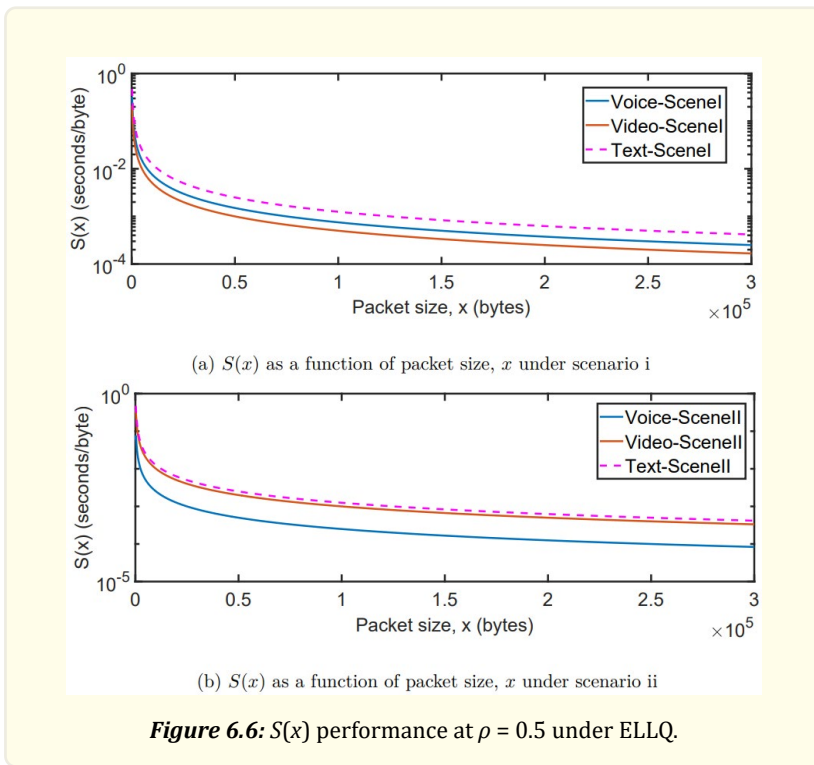
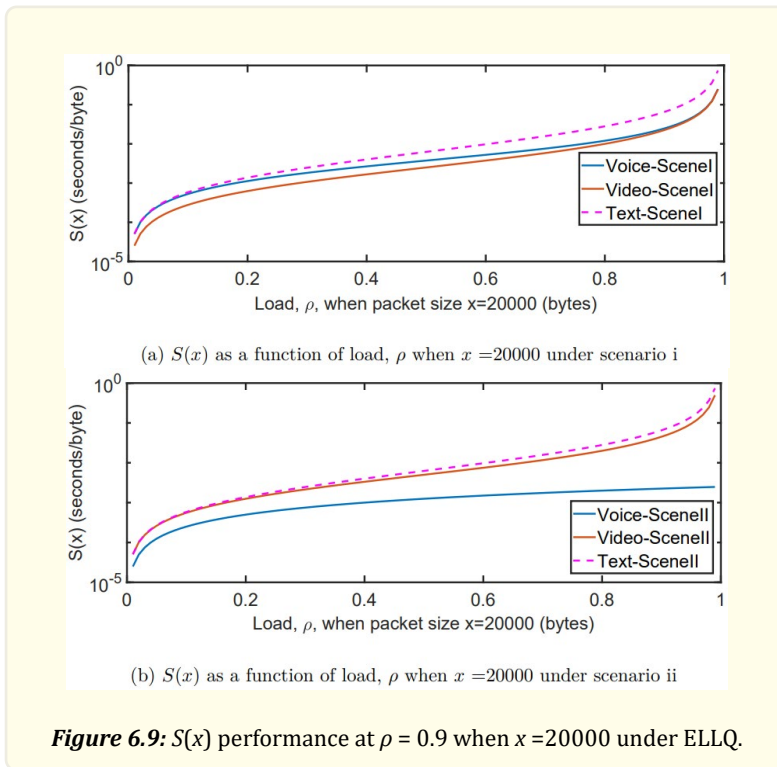
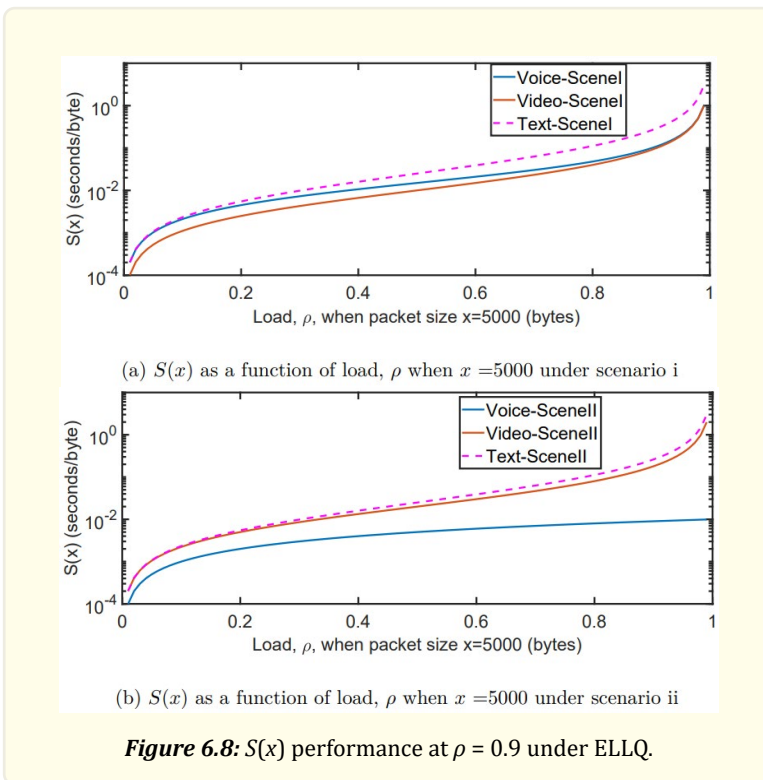


Figure 6.5: $T(x)$ performance for ELLQ at $\rho = 0.9$ when $x = 20000$.





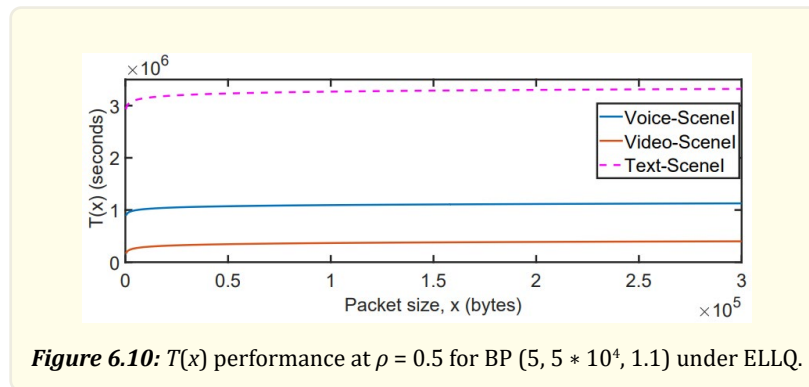


Figure 6.10: $T(x)$ performance at $\rho = 0.5$ for BP $(5, 5 * 10^4, 1.1)$ under ELLQ.

6.3.2 Analysis under heavy tailed workloads

We now study the performance of the algorithms under scenarios i and ii for the BP distribution. Recent traffic measurements, have shown that the traffic of the internet exhibit the heavy-tailedness characteristic [132]. Heavy tailed distributions refer to distributions with tails that decay slower than the exponential distribution [133]. The most commonly used heavy-tailed distributions are pareto, weibull and lognormal. It is a well-known fact that heavy-tailed traffic significantly degrades the performance of networks. Also there have been few works for analyzing queueing systems with heavy-tailed input traffic. Therefore, in this study, we found it necessary to consider the BP to analyze the performance of the ELLQ algorithm. Figure 6.10 shows the $T(x)$ performance for the ELLQ algorithm under the BP $(5, 5 * 10^4, 1.1)$ at $\rho = 0.5$ for voice, video and text packets. We note that even at low load that the conditional mean response time increases with increase in packet size for all the three traffic types (voice, video and text packets). In Figure 6.11, we present the results of the ELLQ algorithm for the conditional mean response time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ for voice, video and text packets. We again note that the conditional mean response time increases with increase in packet size for all the three traffic types (voice, video and text packets). It is a known fact that $T(x)$ grows linearly with x , since the response time of a packet includes (at minimum) the packet size. We observe that video packets registered the best performance, followed by voice and again text packets are the least. The explanation that follows is the partial video packet is transmitted alongside the voice packets in queue1 and this creates additional delay for voice packets. Text packets by default are delay tolerant and have to wait for voice and video packets to be serviced before they receive service. Figures 6.12(a) and 6.12(b) show the $T(x)$ performance of voice, video and text packet for the ELLQ at, $\rho=0.5$ when $x=5000$ bytes under the BP $(5, 5 * 10^4, 1.1)$. We note that even at low load the conditional mean response time increases with increase in load for all the three traffic types (voice, video and text packets). Figures 6.13(a) and 6.13(b) show the $T(x)$ performance of voice, video and text packet for the ELLQ at, $\rho=0.5$ when $x=20000$ bytes under the BP $(5, 5 * 10^4, 1.1)$. We again note that the conditional mean response time increases with increase in load for all the three traffic types (voice, video and text packets). In Figures 6.14(a) and 6.14(b) we show the performance of voice, video and text packet for the ELLQ scheduling algorithm in terms of the conditional mean response time, $T(x)$ as a function of load, ρ when $x=5000$ bytes under the BP $(5, 5 * 10^4, 1.1)$. We note that the conditional mean response time increases with increase in load for all the three traffic types (voice, video and text packets). For scenario i: when voice packet is delayed once and piggy backed with video on transmission; we can also observe from the figure that the ELLQ algorithm performs much better for video packets compared to voice and text packets regardless of the load. For scenario ii: when voice packet is delayed only if there is a partial video packet being transmitted; it can be observed that voice packets experience lower conditional mean response time under ELLQ algorithm. We observe that the performance of video packets is poor, and text packets experience the poorest performance amongst the three traffic types under ELLQ algorithm. Obviously, the poor performance of text packets is attributed to the original design concepts internet traffic that do not guarantee any strict QoS because this class of traffic is delay tolerant. A similar trend is observed when the packet size $x=20000$ bytes under the BP $(5, 5 * 10^4, 1.1)$ as shown in Figures 6.15(a) and 6.15(b). The results of the ELLQ algorithm for the conditional mean slowdown time for BP $(5, 5 * 10^4, 1.1)$ as a function of packet size at $\rho = 0.9$ are presented in Figure 6.16 for voice, video and text packets. We note that the conditional mean slowdown

decreases with increase in packet size for all the three traffic types (voice, video and text packets). We can also observe from the figure that video packets perform much better compared to voice and text packets under the ELLQ algorithm. The explanation for this is the ELLQ algorithm does not permit pre-emption of any job that is already receiving service. In Figures 6.17(a) and 6.17(b) we show the performance of voice, video and data packets for the ELLQ scheduling algorithm in terms of the conditional mean slowdown, $S(x)$ as a function of load, ρ when $x = 5000$ bytes under the BP $(5, 5 * 10^4, 1.1)$. We note that the conditional mean slowdown decreases with increase in packet size for all the three traffic types (voice, video and text packets). As earlier indicated [131], we have used the figures showing ratios $S(x) = T(x)/x$ in order to contrast the behavior of conditional response time across different packet sizes. This is a useful measure because, in many cases, it is appropriate for response times to be proportional to job size (that is to say small packets should have small response times and large packets should have large response times). For scenario i: when voice packet is delayed once and piggy backed with video on transmission; we can also observe from the figure that video packets perform much better compared to voice and text packets under the ELLQ algorithm. This is expected because splitting the video packets comes with some performance gains on the side of video and degradation in performance on the side of voice packets. For scenario ii: when voice packet is delayed only if there is a partial video packet being transmitted; it can be observed that voice packets experience lower conditional mean slowdown under ELLQ algorithm. Video packets experience a slightly higher conditional mean slowdown, and text packets experience the highest conditional mean slowdown under ELLQ algorithm. The poor performance of text packets in both scenarios is attributed the design concept of treating this class as best effort, and also not allowing pre-emption of voice and video packets. A similar trend is observed in Figures 6.18(a) and 6.18(b) for the performance of voice, video and text packets under the ELLQ scheduling algorithm in terms of the conditional mean slowdown, $S(x)$ as a function of load, ρ when $x = 20000$ bytes under the BP $(5, 5 * 10^4, 1.1)$.

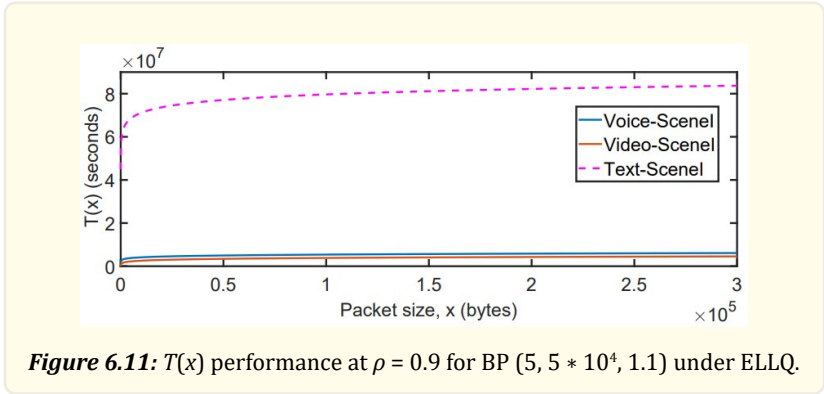
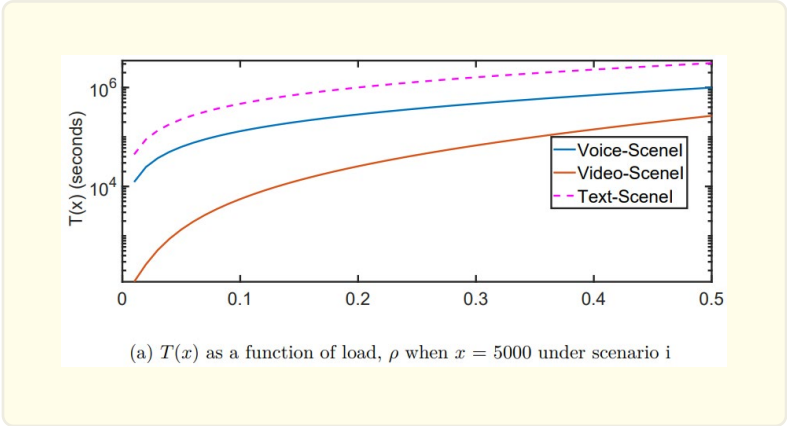


Figure 6.11: $T(x)$ performance at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$ under ELLQ.



(a) $T(x)$ as a function of load, ρ when $x = 5000$ under scenario i

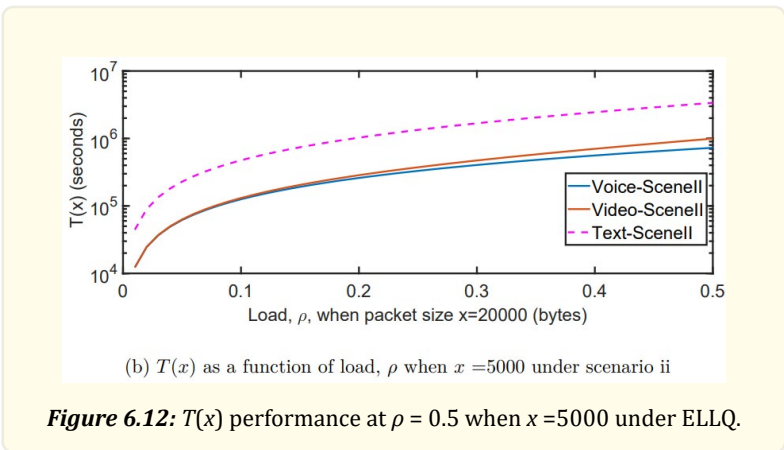


Figure 6.12: $T(x)$ performance at $\rho = 0.5$ when $x = 5000$ under ELLQ.

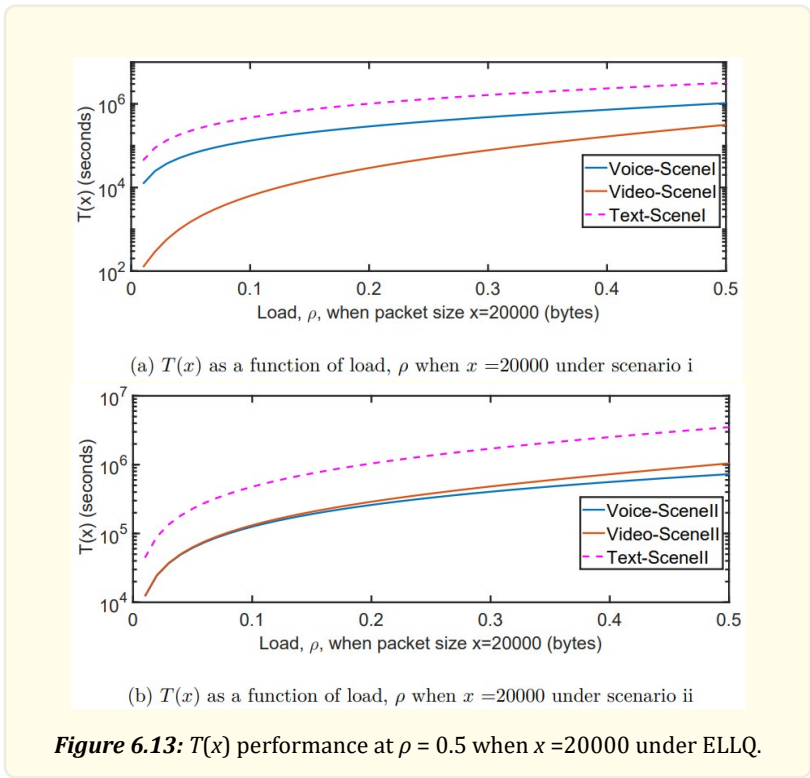


Figure 6.13: $T(x)$ performance at $\rho = 0.5$ when $x = 20000$ under ELLQ.

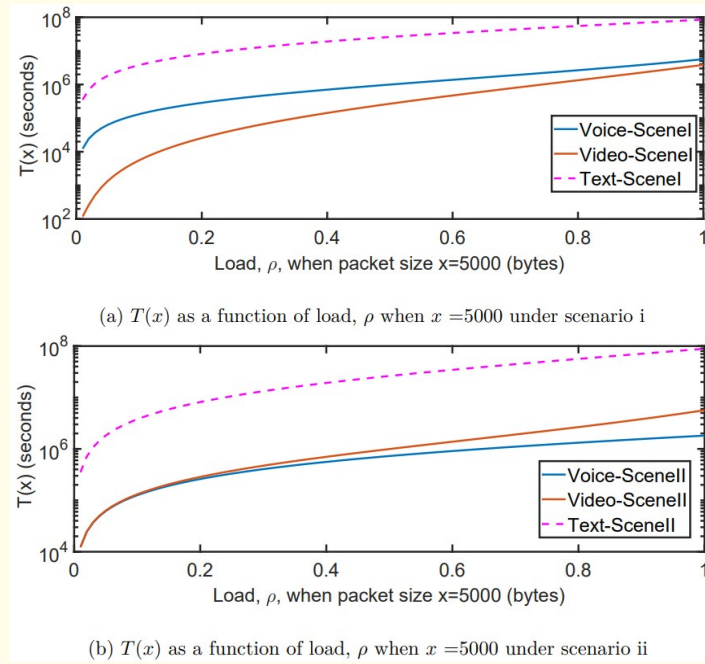


Figure 6.14: $T(x)$ performance at $\rho = 0.9$ when $x = 5000$ under ELLQ.

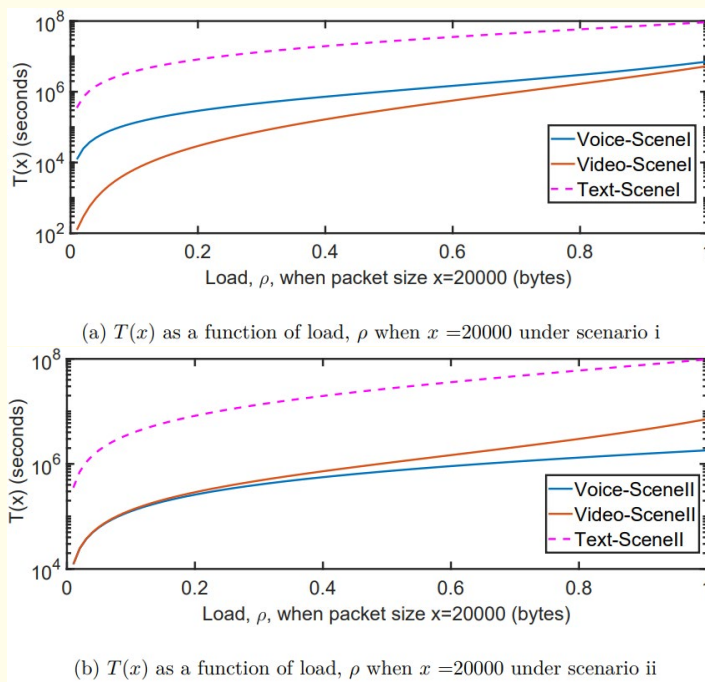


Figure 6.15: $T(x)$ performance at $\rho = 0.9$ when $x = 20000$ under ELLQ.

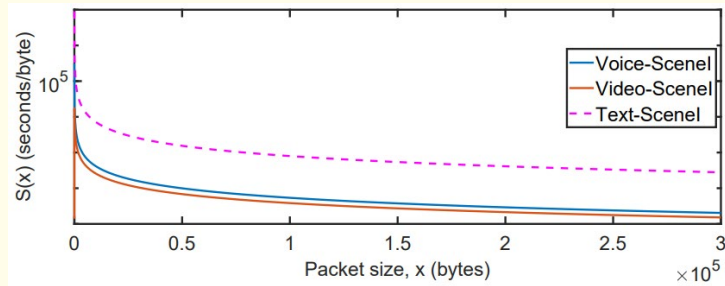
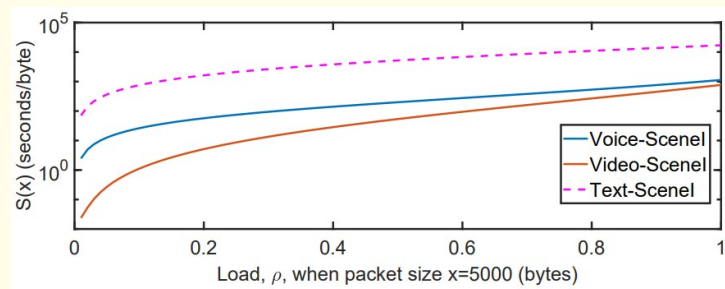
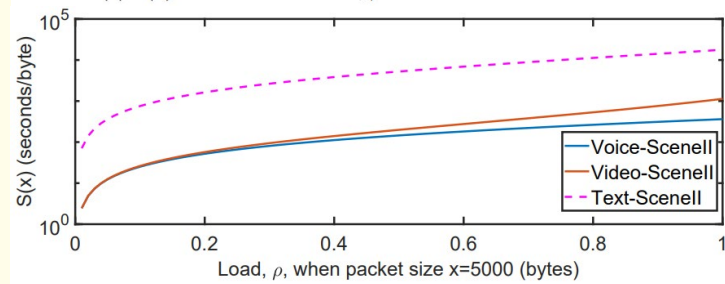


Figure 6.16: $S(x)$ Vs x at $\rho = 0.9$ for BP $(5, 5 * 10^4, 1.1)$.

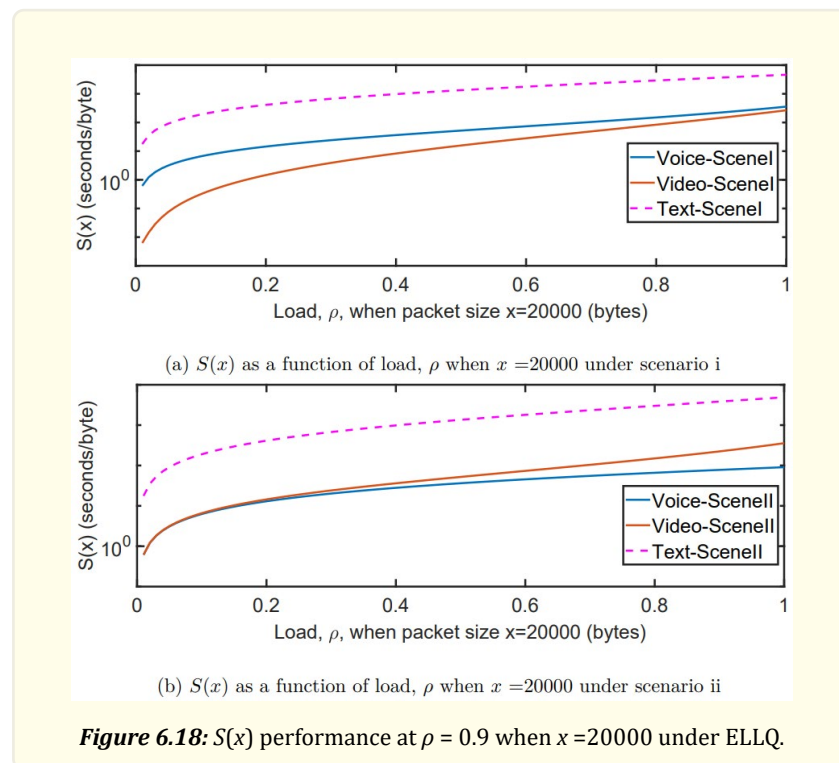


(a) $S(x)$ as a function of load, ρ when $x = 5000$ under scenario i



(b) $S(x)$ as a function of load, ρ when $x = 5000$ under scenario ii

Figure 6.17: $S(x)$ performance at $\rho = 0.9$ when $x = 5000$ under ELLQ.



6.4 Conclusion

In this chapter, a three priority ELLQ queue model was developed. We evaluated the performance of ELLQ algorithm under varying workloads for two scenarios. The results revealed that the video packets experienced the least conditional mean response time/conditional mean slowdown, followed by voice and least were text packets under ELLQ algorithm; and vice versa for scenario ii.

Chapter 7 WRR Scheduling in Manets

In this Chapter, we adopt the WRR algorithm proposed by Hottmar; improve the adopted algorithm of Hottmar; and carry out a performance evaluation of the algorithms at varying workloads. Section 7.1 introduces the generic WRR algorithm plus the mathematical notations and expressions. In Section 7.2, we adapt the WRR algorithm into MANETs environment and indicate the changes made. In Section 7.3, we present the Improved WRR algorithm and justify the reasons for the improvements. The numerical results are presented in Section 7.4. We conclude the Chapter in Section 7.5.

The results are published in: Mukakanya Abel Muwumba, Odongo Steven Eyobu and John Ngubiri (2023). An Improved WRR Scheduling Algorithm for MANETs. In: Arai, K. (eds) Intelligent Computing. SAI 2023. Springer, Cham. https://doi.org/10.1007/978-3-031-37717-4_66

7.1 The generic WRR scheduling algorithm

In WRR queuing packets are classified into service classes (for example realtime, interactive and file transfer) and then allocated a proportion of bandwidth. The packets are assigned to queues specifically dedicated to them. The queues are serviced in a RR order. Figure 7.1 shows WRR queuing, the output port bandwidth allocation is as follows: 25 percent is for real-time traffic; 25 percent is for interactive traffic; and 50 percent is for file transfer traffic. With the above allocations it implies that during each service round, the WRR visits the file transfer queue twice. The WRR algorithm is not aware of the true sizes of the packets in the buffers that are to be

scheduled. The queues and scheduling are generally optimized for mean packet size. The sizes are estimates and have no true meaning with regard to the actual traffic mix in each queue. Recall, the drawback of WRR is that it is blind when it comes of bandwidth allocation and suffers severe limitations when scheduling variable-sized packets besides this issue, its application in MANETs has been given little attention. This research gap serves as the motivation of the study.

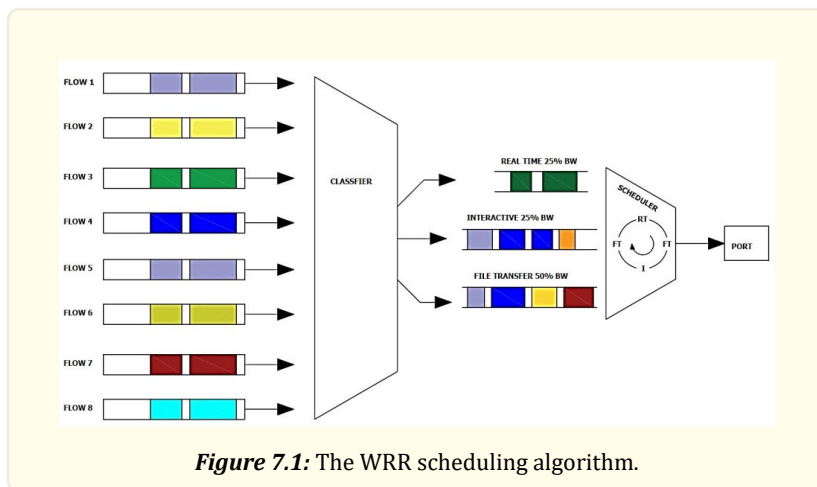


Figure 7.1: The WRR scheduling algorithm.

Preliminaries

The conditional average response time and slowdown are the main performance metrics to be used in the analysis. Table 7.1, presents the mathematical notations and expressions that are closely related to those used in Rai and Okopa [51].

7.2 Adapting the WRR algorithm into MANETs

The study started from the novel point of view, and adopted the general relationship for the mean waiting time of the existing WRR strategy [23] as given in Equation 7.1.

$$W_i = \frac{[(BR - BR_i) * E(X_i)]}{[BR_i * 2(1 - \rho_{iS}) * BR]} \tag{7.1}$$

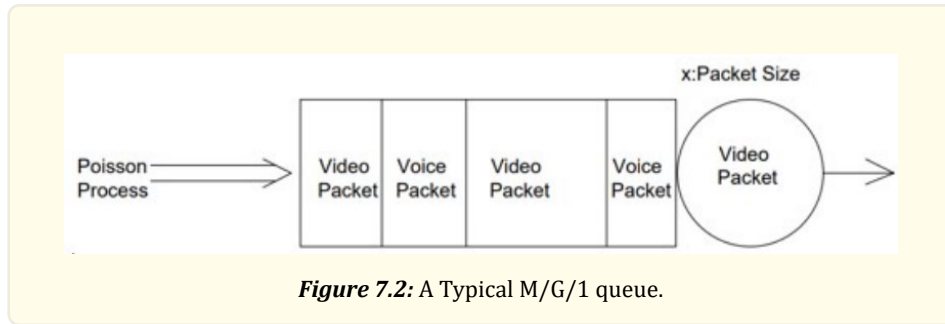
Description	Notation
Average packet arrival rate	λ
Load for voice packets	$\rho_{x_{vO}}$
Load for video packets	ρ_{vI}
Total load for all packets in the system	ρ
Second moment of voice packets of the service-time distribution	$\frac{x_{vO}^2}{2}$
Second moment of video packets of the service-time distribution	$\frac{x_{vI}^2}{2}$
Conditional average response time	$T(x)$
Conditional average response time of voice packets	$T(x_{vO})$
Conditional average response time of video packets	$T(x_{vI})$
Average waiting time of voice packets	$W(x_{vO})$
Average waiting time of video packets	$W(x_{vI})$
Probability density function (pdf)	$f(x)$
Cumulative distribution function (cdf)	$F(x)$
Survival function.	$F^c(x) = 1 - F(x)$

Table 7.1: Mathematical notations and expressions.

The study indicates the following changes and assumptions:

- a. The first adaption is that we assume two queues, i.e., queue 1 for voice and queue 2 for video packets. Basing on Equation 7.1, W_i is the average waiting time of packets of i^{th} queue; BR overall transfer capacity of output interface is replaced with a continuous random variable, x (which also represents packet size); Depending on the traffic type and the service time distribution, $E[X_i]$, the average size of the packets is replaced with the second moment $\overline{x_{vO}^2}$ and $\overline{x_{vI}^2}$ of voice and video packets of the service-time distribution ; BR_p , the proportionate ratio sizes out of the overall capacity BR are replaced with weights, $\omega_1 = 7808$ or $\omega_1 = 1712$ respectively; the load coefficient of processed traffic, ρ_{is} is replaced with ρ_{xvO} or ρ_{xvI}
- b. The second adaption is that we consider an M/G/1 queue. The M/G/1 queue is shown in the Figure 7.2 and we assume that the user requests (voice and video packets) arrive according to a poisson process with mean rate λ packets per second. Also, Figure 7.2 depicts user requests with large and small sized boxes implying that voice and video packets are large size and others are small size meaning that their service demands are different. We use a continuous random variable X which is said to have an exponential distribution with the probability density function (pdf) [51, 104] given as follows:

$$f(x) = \lambda e^{-\lambda x}, \quad x \geq 0, \quad \lambda \geq 0.$$



- i. We use the exponential distribution to get an expression for the second moment for voice packets of size less than or equal to x_{vO} ; and for video packets of size less than or equal to x_{vI} respectively is given as:

$$\overline{x_{xvO}^2} = \int_0^{x_{vO}} t^2 \lambda e^{-\lambda t} dt \tag{7.2}$$

$$\overline{x_{xvI}^2} = \int_0^{x_{vI}} t^2 \lambda e^{-\lambda t} dt \tag{7.3}$$

The load, ρ_{xvO} associated with voice packets of sizes less than or equal to x_{vO} is given as:

$$\rho_{xvO} = \lambda \int_0^{x_{vO}} t f(t) dt \tag{7.4}$$

Likewise the load, ρ_{xvI} associated with video packets of sizes less than or equal to x_{vI} is given as:

$$\rho_{xvI} = \lambda \int_0^{x_{vI}} t f(t) dt \tag{7.5}$$

- ii. We also use a continuous random variable X which is said to have the BP distribution. We write the pdf of the Pareto [51, 104]

as follows;

$$f(x) = \frac{\alpha k^\alpha}{1 - (k/L)^\alpha} x^{-\alpha-1} \quad k \leq x \leq L, \quad 0 \leq \alpha \leq 2.$$

For voice packets, the second moment of sizes less than or equal to x_{vO} and video packets of sizes less than or equal to x_{vI} under the BP distribution are given respectively by;

$$\overline{x_{x_{vO}}^2} = \int_k^{x_{vO}} t^2 f(t) dt \quad (7.6)$$

$$\overline{x_{x_{vI}}^2} = \int_k^{x_{vI}} t^2 f(t) dt \quad (7.7)$$

Note: load,

$$\rho = \lambda \bar{x},$$

The load, $\rho_{x_{vO}}$ for voice packets of sizes less than or equal to x_{vO} under the BP distribution is given as:

$$\rho_{x_{vO}} = \rho_{x_{vI}} = \lambda \int_k^{x_{vO}} t f(t) dt \quad (7.8)$$

The load, $\rho_{x_{vI}}$ for video packets of sizes less than or equal to x_{vI} under the BP distribution is given as:

$$\rho_{x_{vO}} = \rho_{x_{vI}} = \lambda \int_k^{x_{vI}} t f(t) dt \quad (7.9)$$

- iii. The third adaption is we use Hottmar et al. [23] configuration parameters i.e., $E[X1] = 1712$ bit and $E[X2] = 7808$ bit to determine the weights assigned to the queues. We find it appropriate to determine the queue weights to be assigned. using these parameters instead of the guaranteed proportions from the overall capacity. It follows that the expressions for the average waiting times of voice and video packets respectively are given by:

$$W(x_{vO}) = \frac{[x - (\omega_1 * \overline{x_{vO}^2})]}{[\omega_1 * 2(1 - \rho_{x_{vO}})]} \quad (7.10)$$

and

$$W(x_{vI}) = \frac{[x - (\omega_2 * \overline{x_{vI}^2})]}{[\omega_2 * 2(1 - \rho_{x_{vI}})]} \quad (7.11)$$

- iv. The fourth adaption is that we assume the residence time is $\frac{x}{(1-\rho_{x_v})}$. Quite often the conditional average response time consists of two components that is to say the waiting time and the residence time. It follows that the expressions for the conditional average response times of voice and video respectively are given as;

$$T_{(vO)} = \frac{x}{(1 - \rho_{x_{vO}})} + \frac{[x - (\omega_1 * \overline{x_{vO}^2})]}{[\omega_1 * 2(1 - \rho_{x_{vO}})]} \quad (7.12)$$

and

$$T_{(vI)} = \frac{x}{(1 - \rho_{x_{vI}})} + \frac{[x - (\omega_2 * \overline{x_{vI}^2})]}{[\omega_2 * 2(1 - \rho_{x_{vI}})]} \quad (7.13)$$

We summarize the EWRR scheduling in pseudo code algorithm 4.

Algorithm 4 The pseudo code of EWRR

Require: Consider an M/G/1 queue system
 Classify incoming traffic into queue 1(voice) and queue 2(video);
if traffic in the queues is voice or video **then**
 Find the second moment of queue 1 and queue 2 packets;
 Find the load due to queue 1 and queue 2 packets;
 Determine the weights assigned to each queue;
 Determine the average waiting time to each queue;
for each queue **do**
 Compute the cond. average response time;
 Compute the conditional slowdown;
end for
end if

7.3 The improved WRR algorithm

The reasons to improve the EWRR are due to the following deficiencies in the WRR. According to Gautam et al. [94] WRR behaves like a blind scheduling policy because it is ignorant about how different packet lengths are scheduled, the scheduling strategy does not favor queues that have different sized packets. The Improved WRR (IWRR) algorithm is based on the EWRR algorithm with the aim of addressing the above stated deficiencies. We indicate the following improvements in the EWRR algorithm:

- i. The first change is that besides assuming the two queues, one for voice and two for video packets, within each queue the packets are classified into small and large packets.
- ii. The second change is that we use an M/G/1 system where the arrival rate is λ and X is a continuous random variable of the service-time distribution, to get the mean waiting time for the small voice packets of sizes less than or equal to x_{vO} ; and video packets of size less than or equal to x_{vI} respectively as given by Pollaczek-KhinChine (PK) formula.

$$E [W (x_{vO})] = \frac{\overline{\lambda x_{vO}^2}}{2(1 - \rho_{x_{vO}})} \quad (7.14)$$

$$E [W (x_{vI})] = \frac{\overline{\lambda x_{vI}^2}}{2(1 - \rho_{x_{vI}})} \quad (7.15)$$

Where $\overline{x_{vO}^2}$ and $\overline{x_{vI}^2}$ are second moments of voice and video packets of the service-time distribution.

- iii. The third change is that the average size of a large voice and video packet under an exponential and the BP distribution is derived as follows:

$$\rho_{x_{LvO}} = \overline{\lambda x_{LvO}}$$

$$\rho_{x_{LvI}} = \overline{\lambda x_{LvI}}$$

since $\rho_{xL} = \rho - \rho_x$,

$$\overline{x_{LvO}} = \frac{1}{\lambda} (\rho - \rho_{x_{vO}})$$

$$\overline{x_{LvI}} = \frac{1}{\lambda} (\rho - \rho_{x_{vI}})$$

- iv. The fourth change is that we let the survival function (or reliability function), $F^c(x_{th}) = 1 - F(x_{th})$, the probability that a large video or voice packet is found in the queue multiplied by the average size of a large voice $\overline{x_{LvI}}$ or video $\overline{x_{LvO}}$ packet to get the workload due to the large packets. Hence, the expressions for average waiting time for the large packets (voice and video) respectively are:

$$W(x_{vO}) = \overline{x_{LvO}} * F^c(x_{th}) \quad (7.16)$$

$$W(x_{vI}) = \overline{x_{LvI}} * F^c(x_{th}) \quad (7.17)$$

- v. The fifth change is that we add the residence time $\frac{x}{(1-\rho_{x_{vO}})}$ plus equations 7.14 and 7.16 together. We do the same for the residence time $\frac{x}{(1-\rho_{x_{vI}})}$ for Equations 7.15 and 7.17 and then perform some mathematical manipulations to the combined Equations as well as assigning the queue weights. The aim of this operation is to get the expressions for the conditional average response time of the voice and video packets. We use the configuration parameters for queue 1 and 2 in Table 7.3 to determine the queue weights (ω_1 and ω_2) to be assigned. Therefore, the expressions for the conditional average response time of the voice and video packets respectively are:

$$T_{(vO)} = \frac{x}{(1-\rho_{x_{vO}})} + \left[\frac{x - \omega_1 * (\lambda \overline{x_{vO}^2} + 2(1-\rho_{x_{vO}})\overline{x_{LvO}} * F^c(x_{th}))}{\omega_1 * 2(1-\rho_{x_{vO}})} \right] \quad (7.18)$$

and

$$T_{(vI)} = \frac{x}{(1-\rho_{x_{vI}})} + \left[\frac{x - \omega_2 * (\lambda \overline{x_{vI}^2} + 2(1-\rho_{x_{vI}})\overline{x_{LvI}} * F^c(x_{th}))}{\omega_2 * 2(1-\rho_{x_{vI}})} \right] \quad (7.19)$$

Note: The main aim behind the changes (i) to (v) is to enable us compute the partial average waiting times of the small and large voice or video packets. These partial average waiting times are added together to get a resulting average waiting time which is subtracted from the continuous random variable x as shown in expressions 7.18 and 7.19 to obtain the average waiting time of voice and video packets. It is this operation that is responsible for the reduction in conditional average response time and slowdown for the proposed WRR scheduling algorithm as indicated in the results in Section 7.4. We summarize the proposed WRR scheduling in pseudo code algorithm 5.

Algorithm 5 The pseudo code of the IWRR

Require: Consider an M/G/1 queue system

Classify incoming traffic into queue 1(voice) and 2(video);

Classify traffic in each queue into small and large packets;

if packet size in the queue is small or large **then**

Find the second moment of each packet size;

Find the load due to each packet size;

Determine the weights assigned to each queue;

Determine the average waiting time due to each packet size for each queue;

for each queue **do**

 Compute the cond. average response time for each queue;

 Compute the conditional slowdown for each queue;

end for

end if

In Table 7.2, we illustrate traffic by IWRR, where X is as defined earlier.

Step	Typical Output
1	$X = [2.9988 \ 2.9989 \ 2.9990 \ 2.9991 \ 2.9993 \ 2.9994 \ 2.9995 \ 2.9997 \ 2.9998 \ 2.9999 \ 3.0000]$
2	$\bar{x^2} = [9.1733 \ 9.1747 \ 9.1808 \ 9.1809 \ 9.1810 \ 9.1811 \ 9.1812 \ 9.1813 \ 9.1814 \ 9.1815 \ 9.1816]$
3	$\rho = [0.0033 \ 0.0066 \ 0.0100 \ 0.0133 \ 0.0166 \ 0.0199 \ 0.0232 \ 0.0265 \ 0.0299 \ 0.0332 \ 0.0365]$
4	$T(x) = [0.0123 \ 0.0247 \ 0.0373 \ 0.0502 \ 0.0632 \ 0.0764 \ 0.0899 \ 0.1036 \ 0.1176 \ 0.1318]$

Table 7.2: Scheduling of packets by IWRR.

Steps followed in WRR modelling

- i. The WRR algorithms were analytically modelled with two queues using queuing theory in an M/G/1 queue system.
- ii. The tagged packet technique was used to analyze the conditional mean response time by tracking the experience of a tagged arrival (voce or video) to derive the models.
- iii. We defined the expressions for the conditional mean response time under WRR M/G/1/FCFS system; and used these expressions to compute the conditional mean response time for packets for the different priority queues.
- iv. We again considered two job size distributions that is, exponential and BP distributions in the analysis.
- v. We used the technique of computing the partial mean waiting times of the small/large voice/video packets while receiving service under the job size distributions in the design of the WRR algorithm.
- vi. The derived models were implemented in MATLAB to write the codes that consisted of one main function. The main function was used to generate values of packet sizes, loads, conditional mean response time and slowdown under varying workloads.
- vii. The derived WRR models were used in the performance evaluation while comparing the adopted and the proposed WRR algorithms in terms of conditional mean response time and slowdown.
- viii. The results were presented in form of graphic representation of conditional mean response time or slowdown Vs packet size or load. The discussion of the results followed each graph.

7.4 Numerical results

This section presents the performance evaluation of the EWRR and IWRR algorithm under the exponential and BP distributions. In this analysis we first show the weakness of the enhanced WRR algorithm in terms of starving video packets. The study then evaluates the performance of the WRR algorithms with the goal of depicting the performance gains of the IWRR.

7.4.1 Experimental set-up and analysis software

The implementation of WRR scheduling algorithms was done using MATLAB tool already described in Section 3.3. We used the WRR scheduling algorithms developed in Section 7.3 to obtain the MATLAB codes for the study. During the experimental set up, we specified the number of iterations the algorithms were expected to execute. The exponential and BP distributions were used to generate the workloads. We assumed the following typical values of 300,000 iterations; mean packet arrival rates, λ of $\frac{1}{2000}$ and 0.0124, threshold values, x_{th} of 1526.7 and 2500 for exponential and BP distributions respectively; queue weights ($\omega_1 = 7808$ and $\omega_2 = 1712$) as shown in Table 7.3 for the experiments. These parameters were chosen for illustration purposes otherwise these can be varied depending on the user requests. The corresponding values of conditional mean response times and slowdowns were obtained and plotted against the packet sizes as shown next.

Parameter	Values
Average packet arrival rate, λ for exp	$\frac{1}{2000}$
Average packet arrival rate, λ for BP	0.0124
Queue weights (ω_1 and ω_2)	7808 and 1712
System load, ρ	0.9
Range of values for x BP	x=10:0.01:300,000;
Range of values for x exp	x =0:0.1:300,000;
$BP(k, L, \alpha)$	$BP(10, 5000, 1.1)$
Threshold value, x_{th} for BP	1526.7
Threshold value, x_{th} for exp	2500

Table 7.3: Parameters for the Experiments.

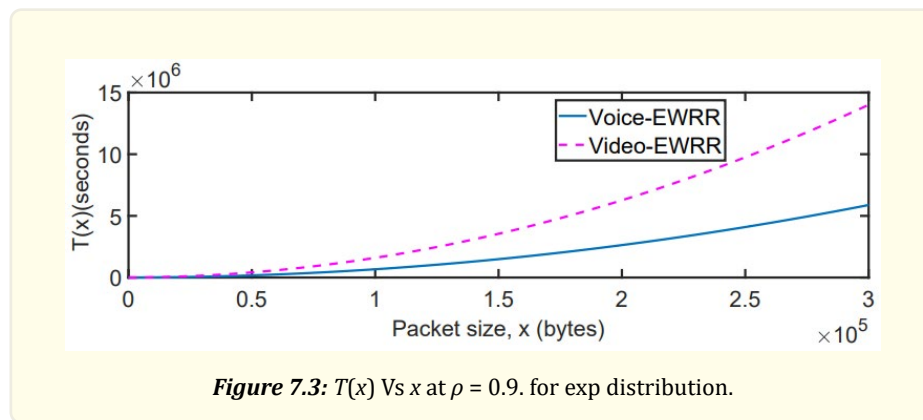
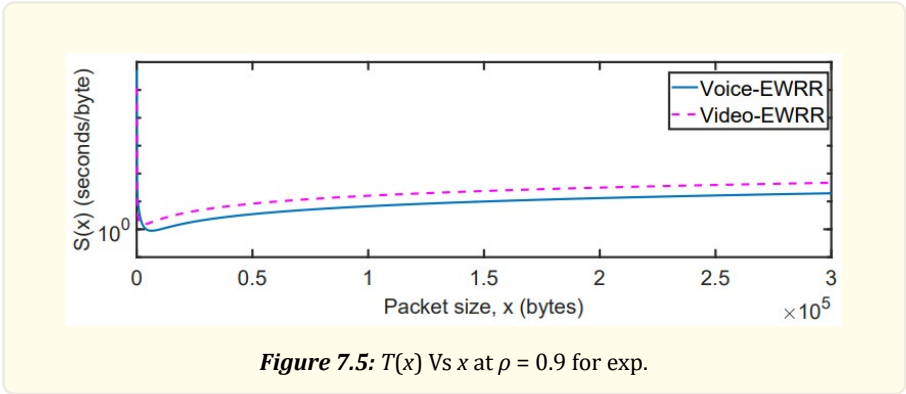
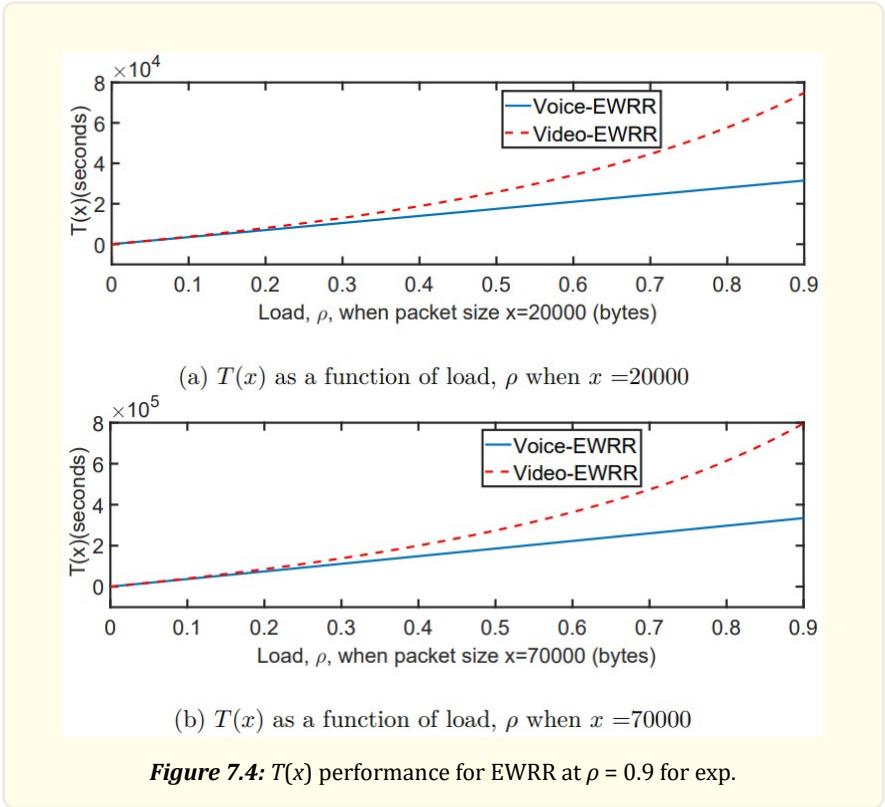


Figure 7.3: $T(x)$ Vs x at $\rho = 0.9$. for exp distribution.

7.4.2 Evaluation of the EWRR algorithm

Figure 7.3 shows $T(x)$ Vs x under the exponential distribution for EWRR scheduling algorithm when ρ is 0.9. We observe that $T(x)$ increases rapidly in the direction of the growth of video packet size, but there is a gentle increase of $T(x)$ for the voice packet size. Obviously, the result reveals that video packets are starved at the expense of voice packets and this is a weakness of this EWRR scheduling algorithm. This result also confirms what the previous solution like Hottmar et al. [23] found out that the size of the weight has a direct relationship with the average waiting time. In Figure 7.4(a) and 7.4(b) we show the performance of voice and video packets for EWRR algorithm in terms of $T(x)$ Vs load, ρ when x is 20000 bytes and 70000 bytes respectively under the exponential distribution. We observe that with fixed packet sizes, the conditional mean response time grows most rapidly in the direction of the growth of load while a gentle increase is realized for the voice packet size. The trend of the graphs obtained shows a closer relationship with those other previous solutions like in Harchol-Balter [50]. The results obtained further revealed that video packets are starved at the expense of voice packets. Figure 7.5 shows $S(x)$ Vs x under the exponential distribution for EWRR scheduling algorithm when ρ is 0.9. We observe that $S(x)$ increases with increase in packet size. We note that $S(x)$ for video packets is higher than that of voice packets. The performance of video packets is poorer than that for voice packets. Obviously, this trend is expected since $S(x)$ is a ratio of conditional average response time divided by packet size. Figure 7.6 shows $T(x)$ Vs x under the BP (10, 5×10^4 , 1.1) for EWRR scheduling algorithm when ρ is 0.9. The results reveal a higher $T(x)$ for video packets compared to voice packets. This is a clear indication that video is being starved at the expense of voice. The difference is much more pronounced when the packet sizes are large. We further note that for the same packet size, $T(x)$ for video packets is still higher than that of voice packets. This result again confirms Hottmar et al. [23] solution that the size of the weight has a relationship with the average waiting time. In Figure 7.7 we present the result for $S(x)$ Vs x under the BP (10, 5×10^4 , 1.1) for the EWRR scheduling algorithm when ρ is 0.9. We observe that video packets registered a lower $S(x)$ compared to that of

voice packets. Recall, $S(x)$ is a ratio of $T(x)/x$. The results clearly show that under the EWRR scheduling algorithm, the video packets are performing poorly in terms of $T(x)$ and $S(x)$ under the two service distributions at high system load. We can therefore conclude that video packets are starved under the EWRR strategy and this algorithm is not scalable.



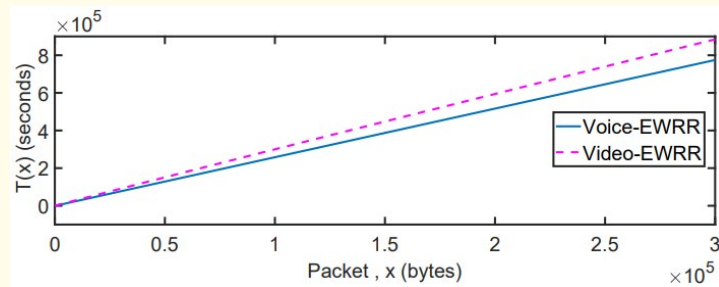


Figure 7.6: $T(x)$ Vs x at $\rho = 0.9$ for BP (10, $5 * 10^3$, 1.1) under EWRR.

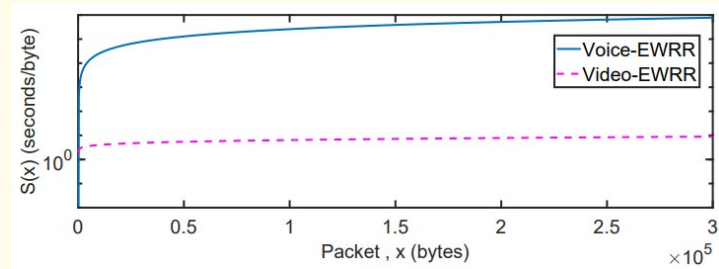


Figure 7.7: $S(x)$ Vs x at $\rho = 0.9$ for BP (10, $5 * 10^3$, 1.1) under EWRR.

7.4.3 Evaluation of EWRR & IWRR under exponential distribution

Figures 7.8(a) and 7.8(b) show the $T(x)$ performance for EWRR and IWRR at $\rho = 0.5$ for exponential workloads. We observe that $T(x)$ increases with increase in packet size. In Figures 7.9(a) and 7.9(b), the performances of voice and video packets for WRR and IWRR scheduling algorithms when ρ is 0.9 is compared in terms of $T(x)$ Vs x under the exponential distribution. We note that $T(x)$ increases with increase in packet size. The graph for $T(x)$ voice packets for EWRR scheduling algorithm is steeper compared to that of IWRR scheduling algorithm. From the result, we can rightly conclude that the IWRR algorithm out-performs EWRR in terms of scheduling voice and video packets. This is so because the EWRR scheduling algorithm is unaware of the packet sizes in the queues, and the mean waiting time is computed as one combined entity for the packets. while in the IWRR scheduling algorithm is aware of the sizes, the voice and video packet sizes are known in advance because the traffic within each queue is classified into small and large packets therefore, the average waiting time is computed as two entities that are later combined.

In Figure 7.10 we compare the performance of voice and video packets for IWRR scheduling algorithm and the results for $T(x)$ Vs x when x under the exponential distribution when ρ is 0.9 are presented. The results are very interesting because they depict that voice and video packets performance is very close or nearly the same. This dramatic improvement in performance of video packets, at the cost of no additional resources indicates that the IWRR scheduling algorithm is scalable. Hottmar et al. [23] points out that one advantage of classification of traffic by service class is that it promotes more equitable management and increased stability for network applications rather than the use of priorities or preferences, and this benefit is clearly observed from this result. Gautam et al. [94] solution on WRR algorithm revealed that a queue that has mostly small packets while another has mostly big packets, then more bandwidth allocation is given to the queue with big packets. On the other hand, if less bandwidth is allocated to the queue with big packets, then the big packets are starved at the expense of small packets. Also, services that have a very strict demand on delay and jitter can be

affected by the scheduling order of other queues because WRR offers no priority levels in its scheduling. However, our result seems to invalidate J. Gautam et al claim. From this result, we are right to say that the IWRR scheduling algorithm is superior compared to the EWRR scheduling algorithm. In Figure 7.11, we present the results of the IWRR scheduling algorithm $T(x)$ Vs x when ρ is 0.9 for voice and video packets under exponentially distributed workloads. We observe that there is no significant variation in performance for all classes of packets. It clearly revealed that at high system load there is no significant performance degradation for voice and video packets occurs. Therefore, we are right to say that the IWRR scheduling algorithm is scalable under exponentially distributed workloads.

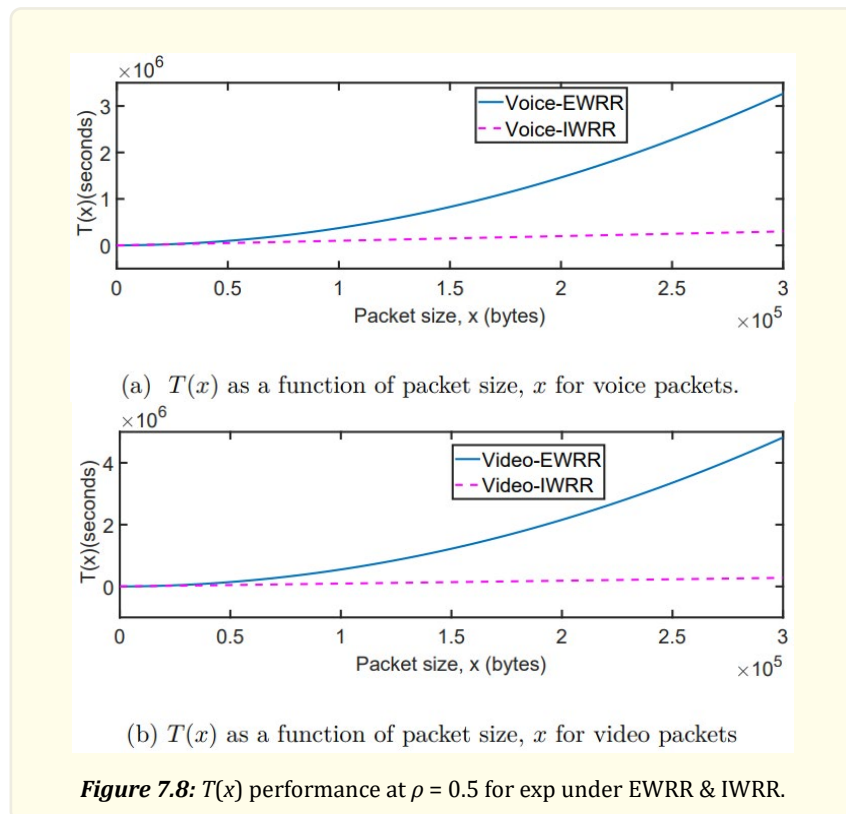
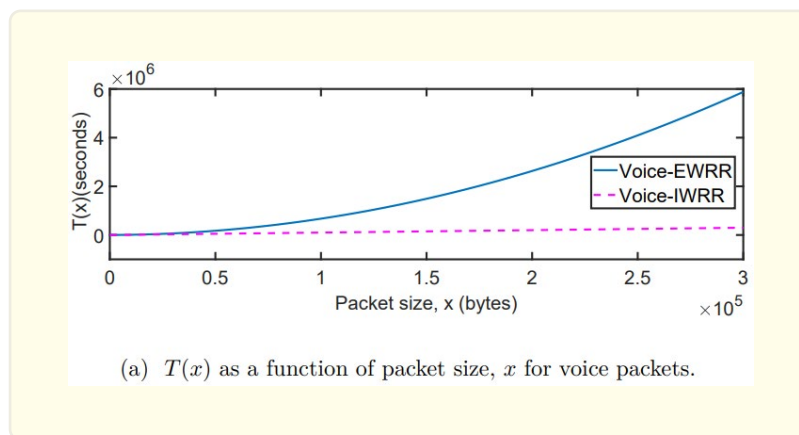
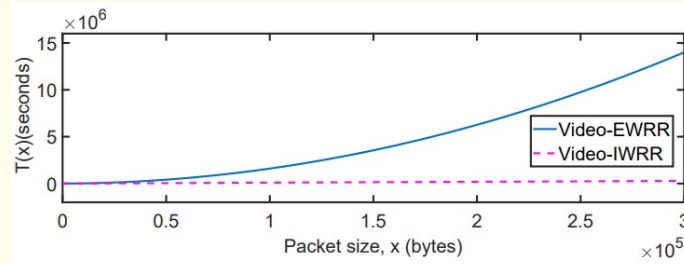


Figure 7.8: $T(x)$ performance at $\rho = 0.5$ for exp under EWRR & IWRR.



(a) $T(x)$ as a function of packet size, x for voice packets.



(b) $T(x)$ as a function of packet size, x for video packets

Figure 7.9: $T(x)$ performance at $\rho = 0.9$ for exp under EWRR & IWRR.

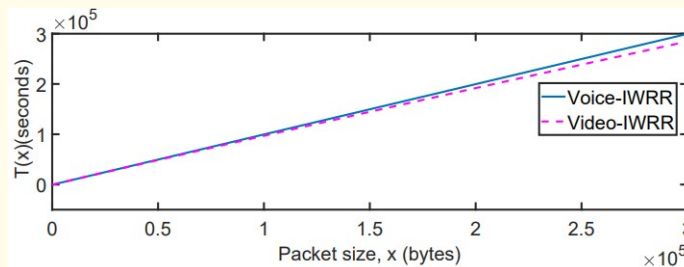


Figure 7.10: $T(x)$ Vs x at $\rho = 0.9$ for exp under IWRR.

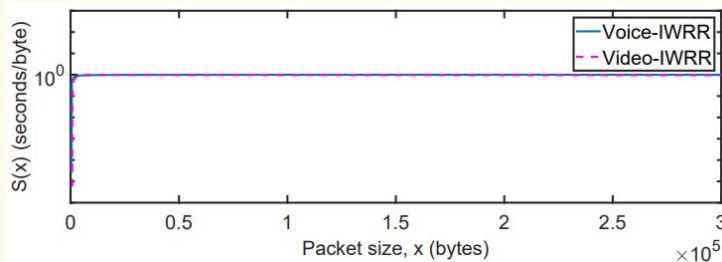


Figure 7.11: $S(x)$ Vs x at $\rho = 0.9$ for exp.

7.4.4 Evaluation of EWRR & IWRR under heavy tailed distribution

In Figures 7.12(a) and 7.12(b), the performances of voice and video packets for the EWRR and IWRR scheduling algorithm are compared in terms of $T(x)$ Vs x when ρ is 0.9 under heavy tailed workloads. As expected, $T(x)$ increases with increase in packet size. We observe a better performance of video and voice packets for the IWRR algorithm compared to EWRR scheduling algorithm. We note that there is performance degradation for voice and video packets in the EWRR scheduling algorithm. This is attributed to the fact the IWRR algorithm is aware of the packet size and utilizes a technique of computing the mean waiting times of small and large packets as separate components, which are later combined into one component. Obviously, the sum of the resulting average waiting time is higher than in the EWRR scheduling algorithm. The resulting average waiting time is got from the difference between the continu-

ous random variable X as shown in Equations 7.18 and 7.19 and the average waiting times of small plus large packets. The resulting average waiting time for the voice and video packets is lower in the IWRR algorithm. Definitely a reduction in average waiting time results into low $T(x)$ for voice and video packets in the IWRR algorithm. We further note that the slight disparity for the video packets compared to voice packets. One strong aspect of the IWRR algorithm is that it can take bursty traffic that conforms to the exponential and BP distribution. By splitting the average waiting times of the small and large packets in this algorithm, we enhance the capability of handling bursty traffic into our analytical model. Figure 7.13 shows the performance comparison of voice and video packets for IWRR scheduling algorithm. The results for $T(x)$ Vs x under the BP $(10, 5 * 10^3, 1.1)$ when ρ is 0.9 are presented. We note that the slight disparity in performance of video and voice packets is small. This result clearly indicates that the IWRR algorithm can take on bursty traffic in conformity with exponentially and heavy tailed workloads. The results of the IWRR scheduling algorithm for $S(x)$ Vs x when ρ is 0.9 for voice and video packets under the BP $(10, 5 * 10^3, 1.1)$ presented in Figure 7.14. We again observe that there is no significant variation in performance for both voice and video packets.

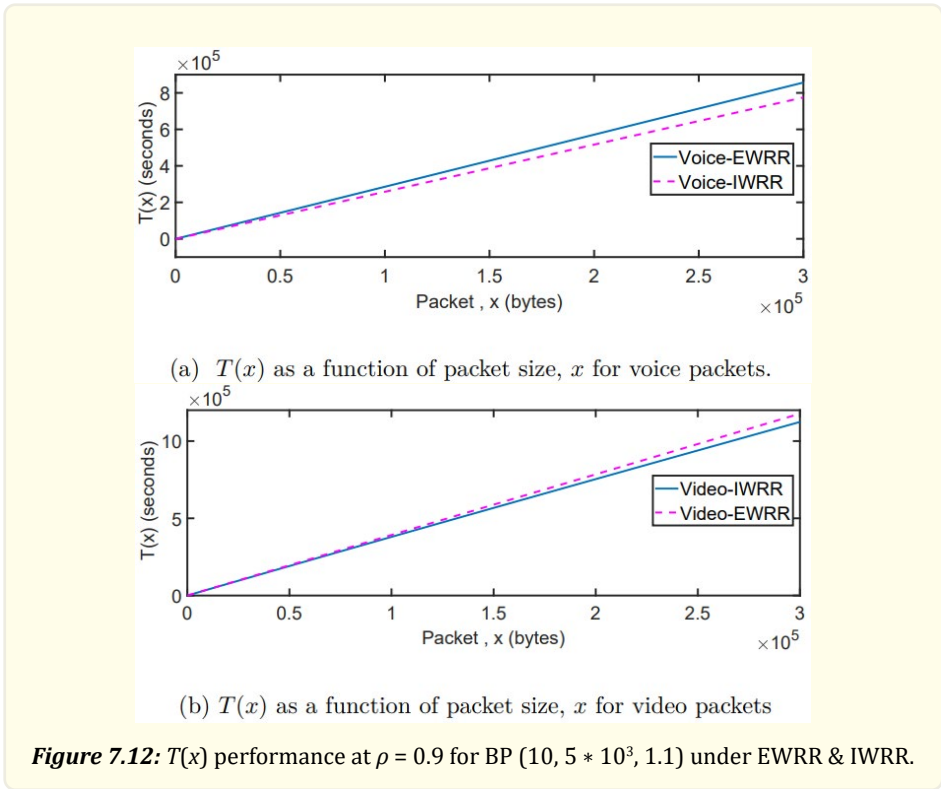


Figure 7.12: $T(x)$ performance at $\rho = 0.9$ for BP $(10, 5 * 10^3, 1.1)$ under EWRR & IWRR.

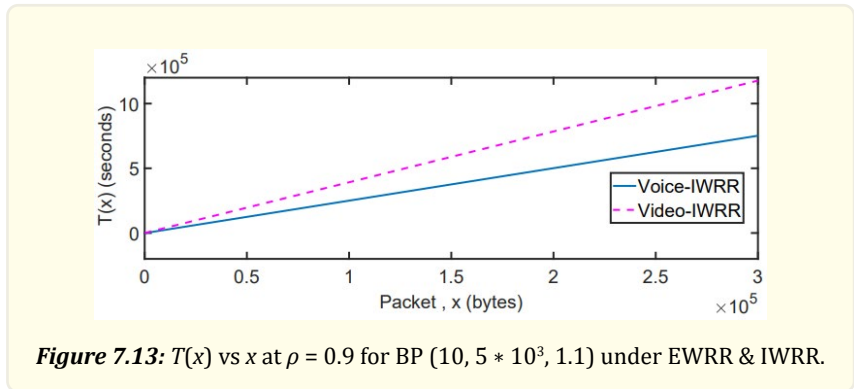
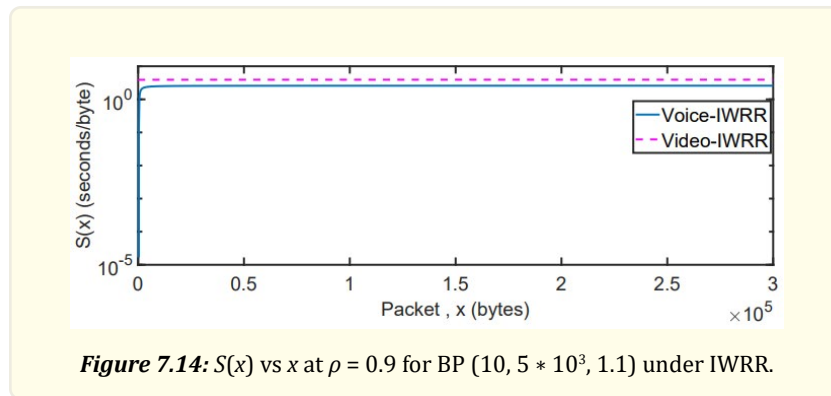


Figure 7.13: $T(x)$ vs x at $\rho = 0.9$ for BP $(10, 5 * 10^3, 1.1)$ under EWRR & IWRR.



7.5 Conclusion

In this Chapter, we adopted the WRR algorithm proposed by Hottmar; improved the adopted algorithm of Hottmar; did performance evaluation of the algorithms at varying workloads. The numerical results were presented. It was observed that the IWRR performed better than the EWRR.

Chapter 8 Conclusion and Future Work

This Chapter highlights the research findings and future work. Section 8.1 summarizes the findings for the EDF, LLQ and WRR algorithms. Lastly Section 8.2 presents the suggested areas for further research.

8.1 Summary of findings

To summarize, we studied three algorithms namely: EDF, LLQ & WRR, and proposed three novel variants i.e., EEDF-II, ELLQ plus IWRR for MANETs.

8.1.1 EDF algorithms

The study first developed a novel EEDFII scheduling scheme that reduces the average waiting time of the priority queue packets. We compared average waiting times of four priority queues i.e., $P1$ -high, $P2$ -medium, $P3$ -normal and $P4$ -low at various system loads for the EEDFI and EEDFII models. From the results obtained, the EEDF-II shortens the waiting times of packets in all queues as compared to EEDF-I.

8.1.2 LLQ algorithms

Next, we studied LLQ algorithms under exponentially and heavy tailed distributed workloads. We adopted an LLQ algorithm from an MMPP/G/1 queue to an M/G/1 queue and then studied it under varying packet size distributions in MANETs. We used the numerical results to show that the adopted LLQ algorithm penalized video packets at high system load depending on the packet size distribution. We proposed an LLQ scheduling algorithm based on the adopted LLQ algorithm. Experiments were conducted to compare the performance of the adopted and proposed LLQ algorithms under exponential and BP distributions at low and high system load. The numerical results showed that the proposed LLQ algorithm outperformed the adopted LLQ algorithm in terms of reducing the conditional mean response times and slowdowns of video packets.

The study further extended the work of proposed LLQ algorithm by classifying traffic into three priority queues namely; queue 1 (voice packets); queue 2 (video packets); and queue 3 (text packets). We considered two scenarios: firstly, when voice packet is delayed once and piggy backed with video on transmission; The results revealed that the video packets experienced the least conditional mean response time/conditional mean slowdown, followed by voice and least were text packets under ELLQ algorithm.

And secondly, when voice packet is delayed only if there is a partial video packet being transmitted. It was observed that voice packets experienced the least conditional mean response time/conditional mean slowdown, followed by video packets and then text packets in that order under ELLQ algorithm. From the results we can rightly conclude that video packets performed best in scenario i while voice packets performed best in scenario ii and text packets registered the least performance in both scenarios under the ELLQ algorithm.

8.1.3 WRR algorithms

The study lastly, proposed an IWRR algorithm that utilizes the technique of computing the partial average waiting times of the small/large voice/video packets. The IWRR algorithm performs better than the EWRR algorithm in terms of conditional mean response time and slowdown. From the numerical results we rightfully conclude that the aim and objectives of the study were achieved.

8.2 Areas of further research

The work presented in this thesis is an investigation of three algorithms i.e., EDF, LLQ and WRR under the simplifying assumption of Poisson arrivals processes and exponential service times. However, the assumption is unrealistic because network traffic is not Poisson, in some cases it is long-range dependent (LRD), and the distribution of packet sizes is not exponential.

To effectively schedule LRD traffic, it might be possible to develop an EDF model with batch arrival processes. The new EDF model should be able to minimize relative performance gaps and starvation trends of lower priority queue packets. The current EDF model could not be adapted readily because it depends upon the iterative process to approximate the mean waiting times. A major limitation of the proposed LLQ is the lack on restriction on the threshold allowable video packets sizes to be split. It is anticipated that if the threshold value is not set and checked this might result into poor performance for the LLQ model. In future work a completely new scheduling LLQ algorithm with multiple queue classes be developed to analyze fairness. Lastly, the limitation with the improved WRR model is that it utilizes the survival function (o reliability function) and average size of a large voice and video packets to approximate the work load due to a large packet. Hence, restricting the to study from investigating the optimal performance of the WRR model. In future work a study to extend and optimize the WRR scheduler beyond two classes of traffic be sought.

References

1. JN Al-Karaki. "Infrastructureless wireless networks: Cluster-based architectures and protocols". Ph.D. dissertation, Iowa State University, Ames, Iowa (2004).
2. J Jubin and JT now. "The DARPA packet radio network protocols". in Proc. of IEEE 75.1 (1987): 21-32.
3. T Sunil and K Ashwani. "A Survey of Routing Protocols in Mobile Ad Hoc Networks". International Journal of Innovation, Management and Technology.
4. DH Morais. 5G and Beyond Wireless Transport Technologies Enabling Backhaul, Midhaul, and Fronthaul. Gewerbestrasse 11, 6330 Cham, Switzerland: Springer Nature Switzerland AG (2021).
5. M Rath, B Pattanayak and B Pati. "Energy Efficient MANET Protocol Using Cross Layer Design for Military Applications". Defence Science Journal 66.2 (2016): 146-150.
6. V Kumar. "Improving Quality of Service in Mobile Ad-Hoc Networks (MANETs) Using Adaptive Broadcast Scheduling Algorithm with Dynamic Source Routing Protocol". Journal of Computational and Theoretical Nanoscience 14.5 (2017).
7. AM Fahad., et al. "Ns2 based performance comparison study between dsr and aodv protocols". Int. J. Adv. Trends Comput. Sci. Eng 8 (2019): 379-393.
8. H Al-Bahadili. "An optimized scheduling scheme in OFDMA WiMax networks". (2012).
9. J Loo, JL Mauri and JH Ortiz. Mobile ad hoc networks: current status and future trends: CRC Press (2016).
10. D Taniar. "Mobile Computing: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications". IGI Global 1 (2008).
11. S Dhar. "MANET: Applications, Issues, and Challenges for the Future". International Journal of Business Data Communications

- and Networking (IJBCDN) 1 (2005): 66-92.
12. B Karaoglu. "Efficient Use of Resources in Mobile Ad Hoc Networks". Ph.D. dissertation, University of Rochester, New York (2013).
 13. MB Sedrati. "Multipath Routing to Improve Quality of Service for Video Streaming Over Mobile Ad Hoc Networks". *Wireless Personal Communications Springer US* 99 (2018): 999-1013.
 14. ML Raja and CDSS Babooi. "An Overview of MANET: Applications, Attacks and Challenges". *International Journal of Computer Science and Mobile Computing (IJCSMC)* 3 (2014): 408-417.
 15. W Chen., et al. "Joint qos provisioning and congestion control for multihop wireless networks". *EURASIP Journal on Wireless Communications and Networking* (2016).
 16. S Malik., et al. "An Adaptive Emergency First Intelligent Scheduling Algorithm for Efficient Task Management and Scheduling in Hybrid of Hard Real-Time and Soft Real-Time Embedded IoT Systems" (2019).
 17. Cisco. "Cisco Visual Networking Index: Forecast and Trends, 2017-2022". 2018, white Paper. [Online]. <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html>
 18. TA Assegie and HD Bizuneh. "Improving network performance with an integrated priority queue and weighted fair queue scheduling". *Indonesian Journal of Electrical Engineering and Computer Science* 19.1 (2020): 241-247.
 19. A Sufian, A Banerjee and P Dutta. "Survey of various real and nonreal-time scheduling algorithms in mobile ad hoc networks". *Industry Interactive Innovations in Science, Engineering and Technology* (2018).
 20. V Abhaya., et al. "Performance Analysis of EDF Scheduling in a Multi-Priority Preemptive M/G/1 Queue". in *IEEE Transactions on Parallel and Distributed Systems* 25.8 (2014).
 21. G Abhaya. "Towards Achieving Execution Time Predictability in Web Services Middleware". Ph.D. dissertation, School of Computer Science and Information Technology, College of Science, Engineering, and Health, RMIT University, Melbourne, Victoria (2012).
 22. S Kakuba, K Kawaase and M Okopa. "Modeling Improved Low Latency Queuing Scheduling Scheme for Mobile AdHoc Networks". in *International Journal of Digital Information and Wireless Communication* (2017).
 23. V Hottmar and B Adamec. "Analytical Model of a Weighted Round Robin Service System". *Journal of Electrical and Computer Engineering* (2012).
 24. Y Yang., et al. "Traffic Agents for Improving QoS in Mixed Infrastructure and Ad Hoc Modes Wireless LAN". *EURASIP Journal on Wireless Communications and Networking* (2005): 1-7.
 25. K WU and J Harms. "QoS Support in Mobile Ad Hoc Networks". *Crossing Boundaries an interdisciplinary journal* 1.1 (2001).
 26. G Bolch., et al. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. Wiley-Blackwell (2006).
 27. A Mohammed., et al. "Weighted Round Robin Scheduling Algorithms in Mobile AD HOC Network".
 28. Chen X, HM Jones and D Jayalath. "Channel Aware Routing in MANETs with Route Handoff". in *Proc. of IEEE Transactions on Mobile Computing* 10.1 (2011): 108-121.
 29. NB Ramantt and I Guptat. "On Demand Routing Protocols for Mobile Ad Hoc Networks: A Review". in *Proc. of IEEE International Advance Computing Conference (IACC)* (2009): 586-591.
 30. D Bruin., et al. "Fair channel dependent scheduling in CDMA systems". in *Proc. of IST Mobile & Wireless Communications Summit* (2003): 737-741.
 31. R Nandakumar. "Review of Packet Scheduling Algorithm in MANET". *Infokara Research* 9 (2019).
 32. D Ferrero and G Urvoy-Keller. "Size-based scheduling to improve fairness and performance in 802.11 networks". *Research Report RR-06183*.
 33. BC Sherin and EM Anita. "A Survey of Scheduling Algorithms for Wireless Ad-hoc Networks". *International Journal of Advanced Science and Engineering* 4.4 (2018): 776-787.
 34. C Semeria. "Supporting Differentiated Service Classes: Queue Scheduling Disciplines, Juniper Networks". (2001): 11-14.
 35. A Demers, S Keshav and S Shenker. "Analysis and simulation of a fair queueing algorithm". (1989). [Online].

36. T Balogh and M Medvecký. "Average Bandwidth Allocation Model of WFQ". *Modelling and Simulation in Engineering*, vol. 2012.
37. M Shreedhar and G Varghese. "Efficient Fair Queueing using Deficit Round Robin". in *Proc. of ACM SIGCOMM* (1995).
38. "Low Latency Queueing Algorithm (LLQ)." [Online]. http://www.cisco.com/en/US/docs/ios/12_0t/12_0t7/feature/guide/pqc-bwfg.pdf
39. A Raj and PB Prince. "Round robin based secure-aware packet scheduling in wireless networks". *International Journal of Engineering Science and Technology* 5.3 (2013).
40. Z Chen, Z Ge and M Zhao. "Congestion aware scheduling algorithm for MANET". *WiCOM* (2006).
41. D Brunonas, A Tomas and B Aurelijus. "Analysis of QoS Assurance using Weighted Fair Queueing (WFQ) Scheduling Discipline with Low Latency Queue (LLQ)". *Proc. In: 28th International Conference Information Technology Interfaces June 19-22, IEEE, Croatia* (2006).
42. NIM Enzai, SS Rais and R Darus. "An Overview of Scheduling Algorithms in Mobile Ad-Hoc Networks". in *Proc. 2010 International Conference on Computer Applications and Industrial Electronics (ICCAIE 2010), Kuala Lumpur, Malaysia* (2010).
43. AK Parekh and RG Gallager. "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks". in *Proc. IEEE/ACM Transaction on Networking* 2.2 (1994).
44. C Bennett and H Zhang. "Wf2q: Worst-case fair weighted fair queueing". in *Proc. IEEE INFOCOM 96* (1996): 120-128.
45. C Liu and JW Layland. "Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment". in *Journal of ACM* 20.1 (1973).
46. V Sivaramani. "End-to-End Delay Service in High Speed Packet Networks using Earliest Deadline First Scheduling". Ph.D. dissertation (2000).
47. H Paloheimo., et al. "Challenges in Packet Scheduling in 4G Wireless Networks". (2006).
48. X Liui. "Opportunistic Scheduling in Wireless Communication Networks". Ph.D. dissertation (2002).
49. B Schroeder and M Harchol-Balter. "Web Servers Under Overload: How Scheduling Can Help". *ACM Transactions on Internet Technology* 6 (2002): 2052.
50. M Harchol-Balter. "Queueing Disciplines". *Wiley Encyclopedia of Operations Research and Management Science* (2009).
51. IA Rai and M Okopa. "Modeling and Evaluation of SWAP Scheduling Policy Under Varying Job Size Distributions". in *Proc. The Tenth International Conference on Networks* (2011).
52. J Nagle. *SIGCOMM Comput. Commun. Rev* 14.4 (1984): 61. [Online].
53. DA Mahmood and G Ath. "A Simple Approximation for the Response times in the Two-class Weighted Fair Queueing System". in *Proc. Conference Paper in Lecture Notes in Computer Science* (2017).
54. S Lu and V Bharghavan. "Fair Scheduling in Wireless Packet Networks". in *Proc. IEEE/ACM Trans. Networking* 7.4 (1999): 473-489.
55. H Luo., et al. "A Self-Coordinating Approach to Distributed Fair Queueing in Ad Hoc Wireless Networks".
56. H Zhang and S Keshav. "Comparison of rate-based service disciplines". in *SIGCOMM* (1991): 113-121.
57. SM Scriba. "Analysis of the EDF family of schedulers". Ph.D. dissertation, University of KwaZulu-Natal, Durban, South Africa (2009).
58. K Chen and L Decreusefond. "An Approximate Analysis of Waiting Time in Multi-Classes M/G/1./EDF Queues". in *Proc. the 1996 ACM SIGMETRICS international conference on Measurement and modeling of com, Las Vegas, NY,USA* (1996).
59. M Andrews. "Probabilistic end-to-end delay bounds for earliest deadline first scheduling". in *Proc. IEEE Infocom* (2000): 603-612.
60. V Sivaraman and FM Chiussi. "Statistical analysis of delay bound violations at an earliest deadline first (EDF) scheduler". *Performance Evaluation* 36-37.1-4 (1999): 457-470.
61. MG Harbour and J Palencia. "Response time analysis for tasks scheduled under EDF within fixed priorities". in *Proc. 24th IEEE RealTime Systems Symposium* (2003): 200-209.
62. K Albers and F Slomka. "Efficient feasibility analysis for real-time systems with EDF scheduling". in *Proc. IEEE Computer Society, Proceedings of the Design, Automation and Test in Europe Conference and Exhibition* (2005): 492-497.

63. TP Baker. "An analysis of EDF schedulability on a multiprocessor". *IEEE Transactions on Parallel and Distributed Systems* 16 (2005).
64. M Kargahi and A Movaghar. "A Two-Class M/M/1 System with Preemptive Non Real-Time Jobs and Prioritized Real-Time Jobs under Earliest-Deadline-First Policy". in *Scientia Iranica* 15.2 (2008): 252-265.
65. Y Dehbi and N Mikou. "Priority assignment for multimedia packet scheduling in MANET". in *Proc. International Conference on Signal Image Technology and Internet Based Systems* (2008).
66. R Barhoun and A Namir. "Packet Scheduling of Two Classes Flow". in *International Journal of Computer Science and Information Technology (IJCSIT)* 3.4 (2011).
67. B Arunkumar, R Avudaiammal and A Swarnalatha. "QoS Based Packet Scheduler for Hybrid Wireless Networks". in *International Journal of Networks (IJN)* 1 (2015).
68. Chun and M Baker. "Evaluation of Packet Scheduling Algorithms in Mobile Ad-Hoc Networks". *ACM SIGMOBILE Mobile Computing and Communications Review* (2018): 3649.
69. M Rath, B Pati and BK Pattanayak. "Cross layer based QoS platform for multimedia transmission in MANET". in *Proc. 11th International Conference on Intelligent Systems and Control (ISCO)* (2017).
70. H Eric, IH Ming and L Hsu-Te. "Low Latency and Efficient Packet Scheduling for Streaming Applications." *Journal of Computer Communications* 29.9 (2006): 1413-1421.
71. A Jesus, V Perez and C Christian. *A Network and Data Link Layer QoS Model to Improve Traffic Performance. Emerging Directions in Embedded and Ubiquitous Computing. Lecture Notes in Computer Science, Springer Berlin* (2006).
72. A Farzad, K Sahar and S Bahram. "A New Scheduling Algorithm Based on Traffic Classification using Imprecise Computation". *International Journal of Computer, Control, Quantum and Information Engineering* 2.9 (2008): 78-82.
73. B Shaimaa, B Fatma and D Gamal. "QoS Adaptation in Real Time Systems based on CBWFQ". in *Proc. 28th National Radio Science Conference (NRSC), Cairo* (2011): 1-8.
74. J Hyunchul, AK Jin and S Hwangjun. "Urgency-based Packet Scheduling and Routing Algorithms for Video Transmission over MANETS." *IET International Communication Conference on Wireless Mobile and Computing* (2011): 78-82.
75. B Shaimaa, B Fatma and D Gamal. "Simulation based performance evaluation of queuing for e-learning real time system". in *Proc. International Conference on Education and e-learning innovations, IEEE* (2012).
76. C Jui-Chi. "Optimized Packet Scheduling Management: Maximizing Bandwidth Utilization for Next-Generation Mobile Multimedia Communications". *Wireless Peers Communication* (2012): 613-630.
77. P Rukmani and R Ganesen. "Scheduling Algorithm for Real Time Applications in Mobile Ad-Hoc Network with OPNET Modeler". *Procedia Engineering Journal* 64 (2013): 94-103.
78. AH Zakaria., et al. "Performance Analysis of Mobile Ad Hoc Networks using Queuing Theory". *Malaysia* (2014).
79. A Ali, N Singh and P Verma. "M/M/1/n+Flush/n Model to Enhance the QoS for Cluster Heads in MANETS". *International Journal of Advanced Computer Science and Applications* 9.5 (2018).
80. Kacem., et al. "A New Routing Approach for Mobile Ad Hoc Systems Based on Fuzzy Petri Nets and Ant System". *IEEE Access* 6 (2018): 65705-65720.
81. M Balter. "Queueing Disciplines". in *Wiley Encyclopedia of Operations Research and Management Science* (2009).
82. Q Zhang, L Ding and Z Liao. "A Novel Genetic Algorithm for Stable Multicast Routing in Mobile Ad Hoc Networks". *China Communications* 16.8 (2019): 24-37.
83. M Sivaram, V Porkodi and AS Mohammed. "Re-transmission DBTMA Protocol with Fast Re-transmission Strategy to Improve the Performance of MANETS". *IEEE Access* 7 (2019): 85098-85109.
84. Z Chen., et al. "An Adaptive on Demand Multipath Routing Protocol with QoS Support for High Speed MANET". *IEEE Access* 8 (2020): 44760-44773.
85. BUI Khan., et al. "A Game Theory-Based Strategic Approach to Ensure Reliable Data Transmission with Optimized Network Operations in Futuristic Mobile Adhoc Networks". *IEEE Access* 8 (2020): 124097-124109.
86. Z Scully, M Harchol-Balter and A Scheller-Wolf. *SOAP: One Clean Analysis of All Age-Based Scheduling Policies. Carnegie Mellon*

- University (2018).
87. P Rukmani and R G. "Enhanced Low Latency Queuing Algorithm for Real Time Applications in Wireless Networks". *International Journal of Technology* (2016): 663-672.
 88. A Sohail, et al. "Implementation of class-based low latency fair queueing (cblfq) packet scheduling algorithm for hsdpa core network". *KSII Transactions on Internet and Information Systems* 14.1 (2020): 473-494.
 89. M Katevenis, S Sidiropoulos and C Courcoubetis. "Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip". *IEEE Journal on Selected Areas in Communications* 9.8 (1991).
 90. HM Chaskar and U Madhow. "Fair Scheduling with Tunable Latency: A Round-Robin Approach". *IEEE/ACM Transactions on Networking* 11.4 (2003).
 91. Y Qian, Z Lu and Q Dou. "QoS Scheduling for NoCs: Strict Priority Queuing versus Weighted Round Robin". in *Proc. IEEE International Conference on Computer Design* (2010): 52-59.
 92. B Tomá and M Martin. " Mean Bandwidth Allocation Model of WRR for IP Networks". in *Proc.35rd International Conference on Telecommunications and Signal Processing* (2012): 156-160.
 93. Tomas Balogh and Martin Medvecky. "Weighted Round Robin and Rate Limiter based Fair Queuing for WRR". *I. J. Computer Network and Information Security* 5 (2015): 51-60.
 94. J Gautam, et al. "Efficient Traffic Scheduling and Congestion Control Mechanism in Wireless Networks ". *International Journal for Scientific Research & Development* 7 (2019).
 95. K Elsayed. "Enhancing the end-to-end schedulability condition of EDF scheduling for real-time applications". in *Proc.IEEE ATM Workshop Proceedings* (1998): 75-79.
 96. K Zhu, Y Zhuang and Y Viniotis. "Achieving end-to-end delay bounds by EDF scheduling without traffic shaping". *IEEE INFO COM* (2001): 1493-1501.
 97. J Lopez, et al. "Worst-case utilization bound for edf scheduling on real-time multiprocessor systems". in *Proc.12th Euromicro Conference on RealTime Systems, 2000 (Euromicro RTS 2000)* (2000): 25-33.
 98. WL Winston. *Operations Research: Applications and Algorithms*, 2nd edition. PWS-Kent Publishing, Boston (1991).
 99. B Filipowicz and J Kwicien. "Queueing systems and networks. Models and applications". Department of Automatics, AGH University of Science and Technology, 30 Mickiewicza Ave., 30-059 Kraków, Poland (2008).
 100. S. Stidham, "Analysis, design and control of queueing systems". *Operations Research* 50.1 (2002): 197-216.
 101. PN Inria. "Basic elements of queueing theory application to the modelling of computer systems". Department of Networks, Faculty of Computing and Information Technology, Makerere University, 2004 route des Lucioles 06902 Sophia Antipolis, France, lecture Notes (2004).
 102. JG Han and Y Qian. "Queueing Theory Based Co-Channel Interference Analysis Approach for High-Density Wireless Local Area Networks". *Sensors* (2016).
 103. J Zhang, GH and Y Qian. "Queueing Theory Based Co-Channel Interference Analysis Approach for High-Density Wireless Local Area Networks". *Sensors* (2016).
 104. IA Rai. "QoS Support in Edge Routers". Ph.D. dissertation, Paris Telcom, France (2004).
 105. E Larsen. "Increasing the Performance of MANETs Throughput and QoS Performance Enhancing Mechanisms for Unicast and Group Communication in Proactive Mobile Ad Hoc Networks". Ph.D. dissertation, Norwegian University of Science and Technology (2011).
 106. M Hasib, J Schormans and T Timotijevic. "Accuracy of packet loss monitoring over networked CPE". *IET Communications* 1 (2007): 507-513.
 107. JA Schormans and CM Leung. Measurement for guaranteeing QoS in broadband multiservice networks.
 108. T Timotijevic, CM Leung and J Schormans. "Accuracy of measurement techniques supporting QoS in packet-based intranet and extranet VPNs". in *Proc. IEE Proceedings-Communications* 151 (2004): 89-94.
 109. M Roughan. "Fundamental bounds on the accuracy of network performance measurements". in *Proc. in Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, Banff, Alberta, Canada:

- ACM (2005).
110. L Kleinrock. *Queueing Systems, Volume 1: Theory*. John Wiley & Sons (1975).
 111. RK Jain. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*: Wiley-Interscience (1991).
 112. JF Kurose and HT Mouftah. "Computer-aided modeling, analysis, and design of communication networks". *IEEE Journal on Selected Areas in Communications* 6 (1988): 130-145.
 113. AM Law and WD Kelton. *Simulation modeling and analysis*, 3rd ed. London: McGraw-Hill (2000).
 114. SM Saleh. "Adaptive Security-Aware Scheduling for Packet Switched Networks Using Real-Time Multi-Agent Systems". Ph.D. dissertation, Western Michigan University, Graduate Colleges (2012).
 115. MKGJW Liu. "Performance of Algorithms for Scheduling RealTime Systems with Overrun and Overload". in Proc. in the Proceedings of the Eleventh Euromicro Conference on Real-Time Systems, held at University of York, England (1999).
 116. L George, P Muhlethaler and N Rivierre. "Optimality and Nonpreemptive Real-time Scheduling Revisited". technical Report 2516, INRIA (1995).
 117. L Georgiadis, R Guerin and AK Parekh. "Optimal Multiplexing on a Single Link: Delay and Buffer Requirements". in Proc. Proceedings of the IEEE Transactions on Information Theory (1997).
 118. V Firoiu, J Kurose and D Towsley. "Efficient Admission Control for EDF Schedulers". in Proc. Proceedings of IEEE INFOCOM (1997): 310-317.
 119. JKR Chipalkatti and D Towsley. "Scheduling Policies for Real-Time and Non-Real-Time Traffic in a Statistical Multiplexer". in Proc. In Proceedings of INFOCOM89 3 (1989): 774-783.
 120. J Peha and F Tobagi. "Evaluation scheduling algorithms for traffic with heterogenous performance objectives". in Proc. In Proceedings of GlobeCom 90 1 (1990): 21-27.
 121. V Sivaraman, FM Chiussi and M Gerlai. "End-to-end statistical delay service under GPS and EDF scheduling: A comparison study". in Proc. In Proceedings of IEEE INFOCOM 01 (2001): 1113-1122.
 122. A Grilo, M Macedo and M Nunes. "A scheduling algorithm for QoS support in IEEE802.11e networks". in Proc. In IEEE Wireless Communications 10 (2003): 36-43.
 123. MD Natale and A Mesch. "Scheduling messages with earliest deadline techniques". In *Journal Real-Time Syst* 20 (2001): 255-285.
 124. L Kleinrock. *Queueing Systems Volume 2: Computer Applications* (1976).
 125. G Lee and J Jeon. "Analysis of an MMPP/G/1/K Finite Queue with Two-Level Threshold Overload Control". *Comm. Korean Math. Soc* 14 (1999): 805-813.
 126. BK Asingwire, M Okopa and T Bulega. "Performance of VoIP Traffic over 802.11 Wireless Mesh Network Under Correlated Interarrival Times". In the *International Journal of Digital Information and Wireless Communications (IJDIWC)* 6.2 (2016): 122-138.
 127. W Fischer and K Meier-Hellstern. "The Markov modulated Poisson process (MMPP) cookbook". in Proc. Proceedings of the Performance Evaluation 18.2 (1993): 149-171.
 128. B Ciciani, A Santoro and P Romano. "Approximate Analytical Models for Networked Servers Subject to MMPP Arrival Processes". in Proc. Proceedings of the 6th IEEE International Symposium on Network Computing and Applications (2007).
 129. PJ Kotian, P Vaishnavi and S Begum. "Review on Data Traffic in Real Time for MANETs". *International Research Journal of Engineering and Technology (IRJET)* 4 (2017).
 130. MA Muwumba, OS Eyobu and J Ngubiri. "An Improved Low Latency Queueing Scheduling Algorithm for MANETs". In K. Arai (Eds.) *Advances in Information and Communication FICC 2023, Lecture Notes in Networks and Systems* 651 (2023).
 131. A Wierman. "Scheduling for Today's Computer Systems: Bridging Theory and Practice". Ph.D. dissertation, Pittsburgh, PA 15213 (2007).
 132. L Byeongchan. "Asymptotic tail distribution analysis of queueing systems with heavy-tailed input traffic". Ph.D. dissertation, Korea Advanced Institute of Science and Technology, College of Natural Sciences (2018).

133. L Clavier, et al. "Experimental Evidence for Heavy Tailed Interference in the IoT ". IEEE Communications Letters (2021): 692-695.