بسم الله الرحمن الرحيم

Sudan University of Science and Technology (SUST)

College of Graduate Studies

Performance Evaluation of Cross layer Design in Multi-hop Relay WiMAX Networks

تقويم الأداء لتصميم تعاون الطبقات لشبكات الواي ماكس ذات المرحل متعددة القفزات

A Thesis Submitted in Partial Fulfillment of the Requirement for the Degree of Master of Science in Electronics Engineering (Computer Engineering)

By:

Thimar Alzubair Ibrahim Alzubair

Supervised By:

Dr. Fath Elrahman Ismael Khalifa

March 2015

الآيـــــة

قال تعالى:

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

نَرْفَعُ دَرَجَاتٍ مَّن نَّشَاءُ ۗ وَفَوْقَ كُلِّ ذِي عِلْمٍ عَلِيمٌ "

صَدَقَ اللَّهُ العَظِيم

(سورة يوسف:الآية رقم 76)

إهـــــــــــداء

إلى من فارقوني جسداً وبقيت أرواحهم تحرسني.. ترفرف حولي.. تمنحني الشعور بالأمان..

إلى روحك أمي وإلى روح جدتي الغالية..

إهداء إلى أبي الذي ناضل الحياة من أجلنا ومازال..

إهداء إلى من هم جزء مني.. لا تكتمل حياتي من دونهم..

إلى إخوتي..

إهداء إلى توائم روحي.. صديقاتي..

وإلى أولئك الذين ما ضنوا علي يومًا من ّ الله به عليهم وبما حازوا من المعرفة وكانت أولى أولوياتهم

في الحياة هي تنويري وتزويدي بخير الزاد..

إلى معلمي ّ .. وأخص بينهم د. فتح الرحمن إسماعيل..

وأخيراً وليس آخراً .. إهداء إليه هو..

ACKNOWLEDGEMENT

First and foremost I offer my sincerest gratitude to my supervisor, Dr. Fath Elrahman Ismael, who has supported me throughout my research with his patience and knowledge whilst allowing me the room to work in my own way. I attribute the level of my Master's degree to his encouragement and effort. One simply could not wish for a better or friendlier supervisor.

I also would like to express the outmost gratitude to Eng. Ahmed Ali from Alneel Center for Electronic Researches who always stands by me to achieve my goals in research field offering me the aids and tools that I need.

A special appreciation and thanks is to Eng. Abdalmjeed Ismaeel for his help and assistance in writing the MATLAB simulation code, I spent with him a great and extremely productive time.

My heartfelt thanks are also due to some of my most important sources of ongoing intellectual and spiritual nourishment: my close friends, the supportive and loving companions. Who have always inspired and then supported me to the fullest in all my pursuits.

I owe gratitude to several of my classmates and colleagues for their upholding and everlasting encouragement during writing this thesis.

Last, but not least, I would be remiss if I fail to express my profound gratitude to my family: my beloved father, siblings, nephews and nieces for the continuous love, and support, and for bringing me up to what I am now. And special thanks to my aunts and cousins for all the encouragement I have received from them and their immeasurable support.

ABSTRACT

Many cross-layer design schemes have been proposed to overcome the limitations of protocol architectures in following strict layering principles. The core idea is to maintain the functionalities associated to the original layers but to allow coordination, interaction and joint optimization of protocols crossing different layers. In this study, we focus on the overhead deriving from the overlap of Transmission Control Protocol (TCP) and Automatic Repeat reQuest(ARQ) functionalities at different layers, where several acknowledgements are generated for a single data block transmitted in wireless channels reducing theavailable bandwidth resources, and taking a long time to be delivered causing a delay. To the aim of reducing unnecessary burden on the wireless link, we propose a cross-layer ARQ approach, called ARQ Proxy. It is a software module located in the protocol stack of the wireless Base Station (BS) to locally generate the TCP ACK for every data packet destined to the Mobile Host (MH) confirming successful data reception, and releases it to the Fixed Host (FH) when requested by the ARQ client which is located in the protocol stack of the MH. Then there is no need for TCP ACK transmission over the wireless link, the saved time can be used by other nodes for data packet delivery which increases overall network capacity. A MATLAB code simulates 802.16j environment with two hops is written and the proposed solution is applied on it. Results show a clear improvement in the chosen network performance parameters. Where there is an increment in the average of throughput and the efficiency and there is a decrement in the Round Trip Time (RTT) and the number of time outs after applying ARQ proxy which improves the efficiency of the proposed solution.

المستخلص

تم إقتراح العديد من مخططات تصميم التداخل الطبقي أو التداخل بين الطبقات للتغلب على المشاكل الناتجة من اتباع معماريات البروتوكولات لمبادئ طبقية مقيّدة. تكمن فكرة هذه المخططات في الحفاظ على الوظائف المرتبطة بالطبقات الأصلية وفي ذات الوقت السماح بالتنسيق والتفاعل والتحسين المشترك لهذه البروتوكولات عبر الطبقات المختلفة. في هذه الدراسة، نحن نركز على العبء الناشئ من تقنيات "إعادة الطلب الأوتوماتيكي متعددة الطبقات" والتي تعمل في طبقتي الرابط والناقل. حيث أنه لكل حزمة بيانات واحدة يتم إرسال أكثر من إشعار تأكيد وصول في الوسط اللاسلكي مما يؤدي إلى تقليل المصادر المتاحة وزيادة زمن التأخير. بهدف تقليل العبء غير الضروري على الوسط اللاسلكي،تم اقتراح طريقة تسمى "وكيل إعادة الطلب الأوتوماتيكي"، عبارة عن برنامج صغير يتم إدراجه ضمن البروتوكولات في المحطة اللاسلكية الأساسية لتوليد إشعار التحكم في الإرسال محلياً لكل حزمة بيانات متوجهة نحو الجهاز المستقبل (النقال) مؤكداً نجاح وصول حزمة البيانات، ويتم إرسالها للجهاز المرسل (الثابت) عندما تُطلب بواسطة الوكيل الثاني والذي يتم إدراجه ضمن البروتوكولات في المستقبل. وبالتالي لا توجد حوجة لنقل إشعار بروتوكول التحكم في الإرسال عبر الوسط اللاسلكي ، وتتم الاستفادة من الزمن المتوفر في نقل حزم بيانات أخربمما يزيد من سعة الشبكة ويحسن أداءها بصورة عامة. تمت كتابة برنامج ماتلاب يحاكي بيئة شبكة الواي ماكس مع تطبيق الحل المقترح. أظهرت النتائج تحسناً ملحوظاً في معاملات قياس الأداء في الشبكة. فهناك زيادة ملحوظة في متوسط معدل البيانات والكفاءة؛ ونقصان واضح في زمن الرحلة وعدد مرات نفاذ الوقت بعد تطبيق "وكيل إعادة الطلب الأوتوماتيكي" مما يثبت فعالية الحل المقترح.

LIST OFCONTENTS

Title **Page**

CHAPTER ONE: INTRODUCTION

CHAPTER TWO: LITERATURE REVIEW

CHAPTER THREE: ARQ CROSS LAYER DESIGN

CHAPTER FOUR: RESULTS AND DISCUSSION

CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS

List of Tables

`

List of Figures

LIST OF ABBREVATIONS

Abbreviation	**Description**
ARQ	Automatic Repeat-Request
BER	Bit Error Ratio
BWA	Broadband Wireless Access
BS	Base Station
BSN	Block sequence Number
CID	Connection Identifier
CRC	Cyclic Redundancy Check
ECC	Error-Correcting Code
FCH	Frame Control Header
FEC	Forward Error Correction
FH	Fixed Host
HARQ	Hybrid Automatic Repeat request
LLC	Logical Link Control
MAC	Media Access Control
MIMO	Multiple Inputs Multiple Outputs
MMR	Mobile Multi-hop Relay
MS	Mobile Station
NACK	Non-ACK
NLOS	Non-Line Of Sight
OFDMA	Orthogonal Frequency Division Multiple Access
OSI	The Open Systems Interconnection Model

PDU	Packet Data Unit
PER	Packet Error Rate
PMP	Point to Multi-Point
QoS	Quality-of-Service
RS	Relay Station
RTO	Retransmission Timeout
RTT	Round Trip Time
SISO	Single Input Single Output
SN	Sequence Number
SS	Subscriber Station
ssthresh	Slow-Start Threshold
TCP	Transmission Control Protocol
TCP ACK	Transmission Control Protocol Acknowledgment
TCP ACK ID	Transmission Control Protocol Acknowledgment Identification
TCP/IP	Transmission Control Protocol/Internet Protocol
TCP NACK	Transmission Control Protocol Non Acknowledgment Negative Acknowledgement
UL	Uplink
VoIP	Voice over Internet Protocol
WiMAX	Worldwide interoperability For Microwave Access
WLAN	Wireless Local Area Networks
WMAN	Wireless Metropolitan Area Networks
WPAN	Wireless Personal Area Networks
WWAN	Wireless Wide Area Networks

CHAPTER ONE

INTRODUCTION

CHAPTER ONE

INTRODUCTION

1-1 Preface

Wireless communications clearly represent a fast-growing sector in the framework of data networks. Mainly, wireless technologies provide mobile access to networks and services – omitting the requirement for a cable and fixed infrastructure, thus enabling fast and cost-effective network organization, deployment and maintenance.

One of the leading technologies for wireless data communications with metropolitan coverage capability is the Worldwide Interoperability for Microwave Access (WiMAX) [1], it provides wireless internet service over longer distances than standard Wi-Fi. WiMAX is based on standard IEEE 802.16 technology.

The multiple-hop structure of Relay Stations (RSs) in WiMAX networks is defined in the IEEE 802.16j. It allows the extension of the radio coverage from the cellular Base Stations (BSs) [2]. It enables Mobile Stations (MSs) to communicate with a base station through intermediate RSs -Mobile Multi-hop relay station (MMR)- to provide additional coverage or performance.

MMR-WiMAX similar to other types of wireless networks suffers from high error rates at radio channel (physical) level: while Bit Error Rate (BER) varies from 10^{-8} to 10^{-6} for wired channels, it varies from 10^{-3} up to

10^{-1} for wireless channels [3][4]. Such error rates are unacceptable for the Transmission Control Protocol (TCP). In addition, data packets can be dropped at the intermediate relay stations due to buffer overflow. Meanwhile, throughput would experience dismal degradation as the number of hops increases, even in an error-free multi-hop environment. In order to meet the performance and reliability requirement and to counteract such variation of BER, Forward Error Correction (FEC) or Automatic Repeat request (ARQ) techniques can be applied. Considering both of them introduces Hybrid ARQ (HARQ) at the Link layer. By using these techniques it becomes possible to compensate the link error rate by several orders of magnitude[5][6].

However, the link layer is not the only layer which acknowledges packet delivery: TCP reliability is obtained through the utilization of a positive acknowledgement scheme which specifies TCP receiver to acknowledge data successfully received from the sender[6][7].

The Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite provides the set of protocols which is the de facto standard for communications in Internet today. Designed in early days of ARPANET for traditional wired networks, TCP/IP shows poor performance in wireless network environment. Among the main reasons of this performance degradation are limitations of wireless medium in terms of bandwidth, latency, information loss, and mobility [8].

Various approaches have been proposed to optimize TCP/IP performance in wireless networks. Most of them target at either a compensation of undesirable characteristics of wireless medium making it look like a wired one, or an introduction of direct modifications of higher layer protocols like TCP making them network-aware. In fact, just a few of the proposed solutions were adapted in final products mainly due to the low tradeoff between performance improvement and implementation complexity.

However, during the last several years an alternative approach for the design is gaining interest, namely Cross-Layer design. It overcomes layering principles employed in network architectures and protocol stacks allowing joint interlayer optimization. As a result, cross-layer design allows better performance optimization in such challenging scenarios as wireless and heterogeneous networks [4].

1-2 Problem Statement

Due to the overlap of TCP and ARQ functionalities at different layers; several acknowledgements are generated for a single data block transmitted in wireless channels. These acknowledgements correspond to the overhead reducing available bandwidth resources, and it takes a long time to be delivered causing a delay.

1-3 Proposed Solution

The proposed approach is called Cross-Layer ARQ or ARQproxy. It substitutes the transmission of TCP ACK packet with a short MAC layer request on the radio link. The protocol stacks of the wireless sender and receiver (BS and MH) are equipped with cross-layer ARQ agents. The agent in the BS assumes successful data packet reception and locally generates TCP ACK, in the MH after the TCP layer sends its ACK the agent substitutes it with a short MAC layer packet which triggers the agent in the sender to release the corresponding stored TCP ACK. Packet identification is achieved through the use of hash functions applied to the packet headers.

As a result, Cross-Layer ARQ approach avoids the transmission of TCP ACK packets over the wireless channel. The saved time can be used by other nodes for data packet delivery which increases overall network capacity.

1-4 Objectives

The main objective of using ARQproxy approach in WiMAX 802.16j is to improve the overall Performance and system capacity and:

- To reduce the Round Trip Time (RTT).
- To decrement the number of time outs.
- To reduce the amount of bandwidth resources; required for TCP ACK transmission in the wireless channel.
- To increase the system throughput.

1-5 Methodology

In this research thesis, the performance bounds and limitations present in wireless networks are identified firstly, then designed a proper cross-layer solution targeting the optimization of the acknowledgement scheme which brings performance improvement through the reduction of the medium-busy time and it requires interaction between the transport and link layers.

This is achieved through a design of ARQproxy agents and proved by a mathematical model. Evaluation results are obtained through a simulation code.

An executable M-file is written using MATLAB version 7.5.0 (R2007b) implementing the main physical and MAC layers features of 802.16j with one wireless BS and two Relay stations.

Each TCP packet is divided into four fragments and they are put in the ARQ window to be sent. Sender expects to receive the TCP ACK after one frame duration for the first hop and two frame duration for the second hop.

Different Packet Error Rates (PERs) are assumed for each link and a random function is used to select some packets to be acknowledged negatively.

The ARQproxy in the BS generates TCP ACK for every transmitted TCP data packet and stores it until requested by the ARQ client in case of positively received packets; it delivers the ACK to the link layer, otherwise; in case of receiving Non ACK (NACK) or the time of the packet is out; the packet is retransmitted and its corresponding ACK is dropped.

1-6 Thesis Outlines

The rest of the thesis is organized as follows: Chapter Two provides overview of MMR, TCP and other related works, besides it introduces other solutions which have already been applied.

Chapter Three addresses Cross-Layer Design in more details and introduces the proposed Cross-Layer ARQ solution designed for performance enhancements obtained from the introduced feedback between the link and the transport layers of the protocol stack in MMR network.

Chapter four includes simulation results and discussion.

Finally; Chapter Fivepresents a summary of the research work drawing conclusions and outlines directions for future research in the field.

CHAPTER TWO

LITERATURE REVIEW

CHAPTER TWO

LITRITURE REVIEW

2-1 Wireless Networks

The term refers to any kind of networking that does not involve cables. It is a technique that helps entrepreneurs and telecommunications networks to save the cost of cables for networking in specific premises in their installations. The transmission system is usually implemented and administrated via radio waves where the implementation takes place at physical level [9].

The interest in wireless communications has grown constantly for the past decades, leading to an enormous number of applications and services embraced by billions of users.

In order to meet the increasing demand for high-bandwidth network services, high data rate radio networks have recently been proposed, such as IEEE 802.11s and IEEE 802.16sfamilies of standards.They provide high-speed wireless communications to a large number of users across wide areas [10].

The types of wireless networks (shown in Figure (2-1)[11]) are defined on the bases of their size (that is the number of machines), their range and the speed of data transfer. Here is a brief description for them:

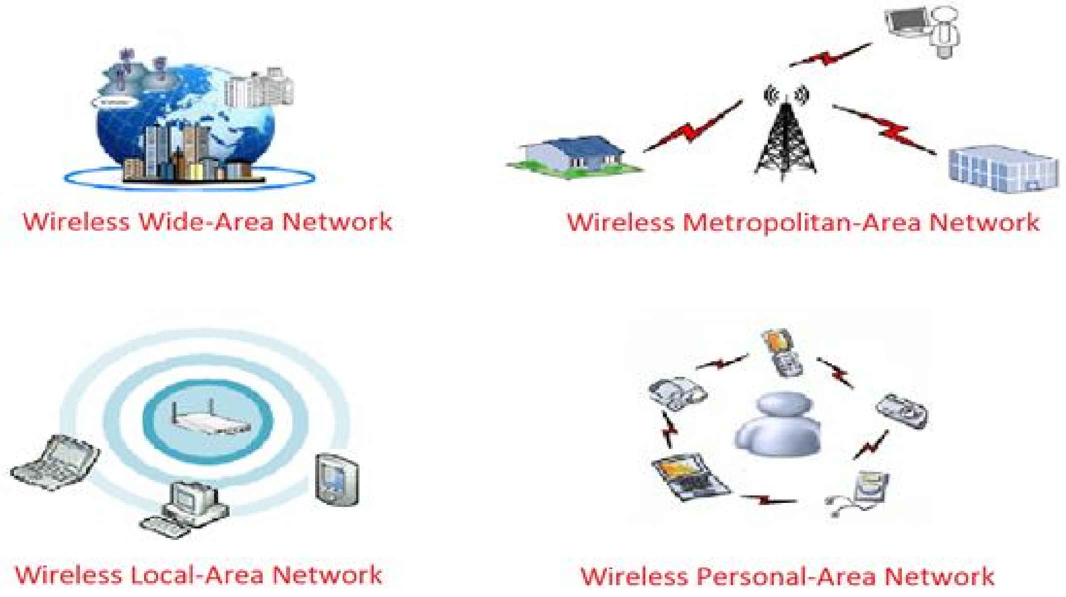
Wireless Wide-Area Network
Wireless Metropolitan-Area Network
Wireless Local-Area Network
Wireless Personal-Area Network

Figure (2-1): Wireless Network Architectures.

2-1-1 Wireless - Personal Area Network (WPAN)

WPANs are designed to connect user devices located within personal communication range which is typically considered of up to 10 meters from a person. Bluetooth is the leading industry standard for WPANs. Nowadays, WPANs are supported by mobile phones, PDAs, laptops and other wireless devices [11].

2-1-2 Wireless - Local Area Network (WLAN)

The simplest wireless distribution method that is used for interlinking two or more devices providing a connection to wider internet through an access point covering a small geographic area, like corporate or campus building.OFDM or spread-spectrum technologies give clients freedom to move within a local coverage area while remaining connected to the LAN. LAN’s data transfer speed is typically 10 Mbps for Ethernet and 1 Gbps for Gigabit Ethernet. Such networks could accommodate as many as hundred or even one thousand users. The IEEE 802.11, commonly known as WiFi, became the de facto standard for WLAN networking[12].

2-1-3 Wireless- Metropolitan Area Networks (WMAN)

The wireless network that is used to connect at high speed multiple locations within a metropolitan area. The network allows two or more nodes to communicate with each other as if they belong to the same LAN. The set up makes use of routers or switches for connecting with high-speed links such as fiber optic cables. WiMAX described as 802.16 standards by the IEEE is a type of WMAN [12].

2-1-4 Wireless - Wide Area Networks (WWAN)

WAN is the wireless network that usually covers large outdoor areas. The speed on such network depends on the cost of connection that increases with increasing distance. The technology could be used for interconnecting the branch offices of a business or public internet access system. Developed on 2.4GHz band these systems usually contain access points, base station gateways and wireless bridging relays. Their connectivity with renewable source of energy makes them stand alone systems. The most commonly available WWAN is internet [9].

2-2 WiMAX Networks

The most promising emerging broadband wireless technology, Worldwide Interoperability for Microwave Access (WiMAX) is the IEEE 802.16 standards-based wireless technology, which defines Air Interface for fixed and mobile Broadband Wireless Access (BWA) Systems, and it is a WMAN. The IEEE 802.16 standard includes MAC and PHY layer specifications and it is designed to achieve goals like easy deployment, high speed data rate, large spanning area and large frequency spectrum [5].

WiMAXtechnology provides higher speed connection up to 70 Mbps over the area of 30 miles. There is no need for line of sight connection between subscriber terminals and the base station in WiMAX technology and it can support hundreds if not thousands of subscribers from a single base station. It also supports low latency applications such as voice, video, and Internet access at the same time [13].

There are several IEEE 802.16 standards as illustrated bellow:

- **802.16:** This is the basic 802.16 standard that was released in 2001. It provided for basic high data links at frequencies between 11 and 60 GHz. (Now withdrawn).
- **802.16a:** This amendment addressed certain spectrum issues and enabled the standard to be used at frequencies below the 11 GHz minimum of the original standard. (Now withdrawn).
- **802.16b:** It increased the spectrum that was specified to include frequencies between 5 and 6 GHz while also providing for Quality of Service aspects.(Now withdrawn).

- **802.16c:** This amendment to 802.16 provided a system profile for operating between 10 and 66 GHz and provided more details for operations within this range. The aim was to enable greater levels of interoperability.(Now withdrawn).
- **802.16d (802.16-2004):** It was a major revision of the 802.16 standard and upon its release, all previous documents were withdrawn. The amendment provided a number of fixes and improvements to 802.16a including the use of 256 carrier OFDM. Profiles for compliance testing are also provided, and the standard was aligned with the ETSI HiperMAN standard to allow for global deployment. The standard only addressed fixed operation.
- **802.16e(802.16-2005):** This standard was provided for nomadic and mobile use. With lower data rates of 15 Mbps against to 70 Mbps of 802.16d, it enabled fullnomadic and mobile use including handover.
- **802.16f:**Management information base.
- **802.16g:**Management plane procedures and services;
- **802.16h:**Improved coexistence mechanisms for license-exempt operation.
- **802.16j:**Multi-hop relay specification.
- **802.16k:**802.16 bridging.
- **802.16m:**Advanced air interface. This amendment is looking to the future and it is anticipated it will provide data rates of 100 Mbps for mobile applications and 1 Gbps for fixed applications. It will allow

cellular, macro and micro cell coverage, with currently there are no restrictions on the RF bandwidth although it is expected to be 20 MHz or more.

2-3 Mobile Multi-hop Relay WiMAX

The Single hop WiMAX products such as IEEE802.16e have limited coverage and provide poor Quality-of-Service (QoS) for indoor users as well as for users at cell boundaries. To address this issue, since 2006 IEEE802.16j Multi-hop Relay Task Group has been working to define a new relay station which can be used as an extension to the base station and relay traffic between the BS and the MS. As a result, the channel quality received by the MS becomes better, and the overall network capacity increases dramatically. Additionally, RS significantly reduces the installation and operation cost compared with using micro- BS to cover these areas. The functionality of the BS should be extended in order to support incorporation of relays into the network. BS which incorporates these new functions is called a Mobile Multi-hop Relay Base Station (MMR-BS). However adding the RS increases the complexity and delay of relaying information both signaling and user data across multiple hops [15]. Figure (2-2)[16] presents a simple MMR network.

Multi-hop relay station (M-RS) is an optional deployment that may be used to provide additional coverage or performance advantage in an access network. The RS may be fixed in location or, in the case of an access RS, it may be mobile access RS. Most of the time the RS will act as a BS and should have its own physical cell identifier, and also it should be able to transmit its own synchronization channels and control information. There should be no difference between cell control in RS and BS.

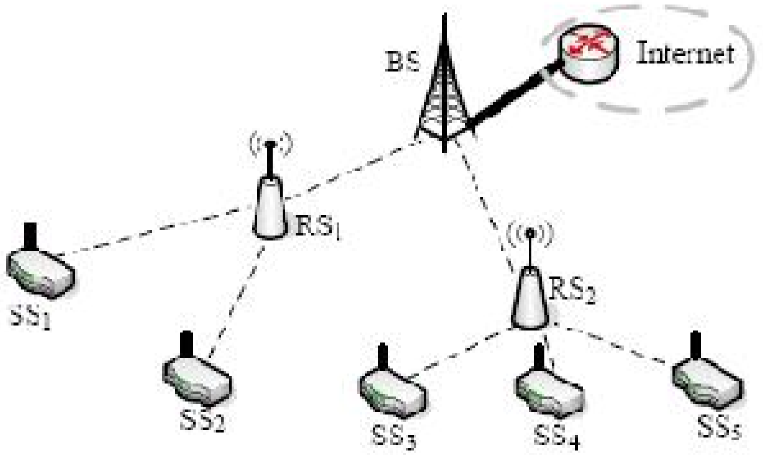

Figure (2-2): IEEE 802.16j Mobile Multi Hop Relay (MMR) network.

The radio link originating or terminating at an MS is named as the access link, but the link between BS and RS or between pair of RSs is called relay link. These access link and relay link can be used for uplink and downlink data transmission.

This standard defines the physical and the MAC layer specifications for MMR networks. The MAC layer supports functions such as network entry, bandwidth request, forwarding of PDUs, connection management and Hand over. The PHY layer adopts Orthogonal Frequency Division Multiple Access (OFDMA) as the primary channel access mechanism for Non-Line Of Sight (NLOS) communications in the frequency band below 11 GHz. Where multiple users are allocated separate set of slots, so that they can communicate in parallel. It supports Point to Multi-Point (PMP) network topology where resource allocation is performed by BS on a per connection basis and the SSs are treated equally.

Multiple Input Multiple Output techniques (MIMO) have ability to exploit NLOS channels and increase spectral efficiency compared to Single Input Single Output systems(SISO). It is able to provide high capacity and data rate without increasing bandwidth. The gain of MIMO is multiplexing gains, diversity gains and array gains.

Two different relay modes are defined in this standard, transparent mode and non- transparent mode.

a. Transparent Relay Mode

The transparent relay mode increases the throughput which facilities capacity increases within the BS coverage area. It has no support to coverage extension because it does not forward framing information to BS. It is operated in two hop network topology and supports centralized scheduling only as scheduling is done only in BS. It uses Connection IDentifier(CID) based forwarding scheme and supports embedded and explicit mode of path management [16].

b. Non-transparent Relay Mode

The non-transparent relay mode is to increase the coverage extension of BS, here RS generate its own framing information and forward it to MSs. It operates is 2 or more hops and uses centralized or distributed scheduling mode, as scheduling is done in BS and RSs. It used CID and Tunnel based forwarding scheme and supports embedded and explicit mode of path management [16](refer to Figure (2-3)[17]).

The transparent relay station does not transmit control message, permeable, Frame Control Header (FCH), and DL/UL-MAP, as it only increases system throughput. The non-transparent relay station transmit control message, permeable, FCH, and DL/UL-MAP, as it increases system throughput and increases cell coverage [16].

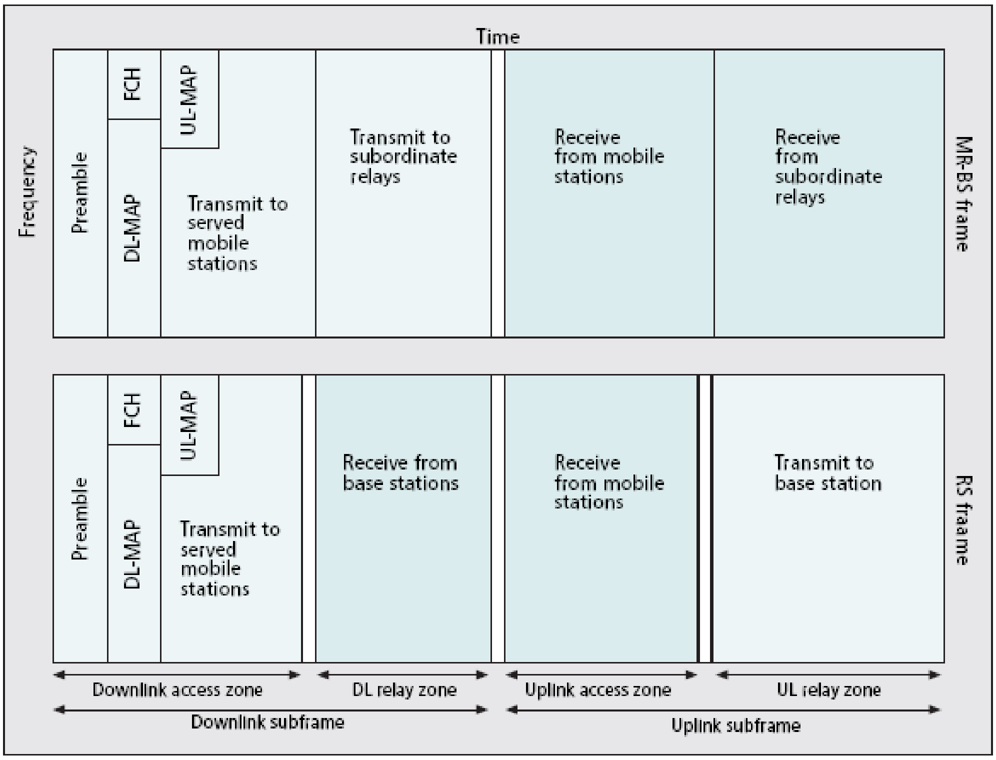
Time
Frequency
Preamble
FCH
UL-MAP
DL-MAP
Transmit to served mobile stations
Transmit to subordinate relays
Receive from mobile stations
Receive from subordinate relays
MR-BS frame
Preamble
FCH
UL-MAP
DL-MAP
Transmit to served mobile stations
Receive from base stations
Receive from mobile stations
Transmit to base station
RS fraame
Downlink access zone
DL relay zone
Uplink access zone
UL relay zone
Downlink subframe
Uplink subframe

Figure (2-3): The Non-transparent Relay Mode Frame.

2-4 Transmission Control Protocol/Internet Protocol (TCP/IP) Model

The protocol stack used on the Internet is the Internet Protocol Suite. It is usually called TCP/IP after two of its most prominent protocols, but there are other protocols as well. The *TCP/IP model* is based on a five-layer model for networking.

The TCP/IP stack is comprised of modules. Each module provides a specific function, but the modules are fairly independent. The TCP/IP layers (shown in figure (2-4)[18]) contain relatively independent protocols that can be used depending on the needs of the system to provide whatever function is desired. In TCP/IP, each higher layer protocol is supported by lower layer protocols [18].

• The *Physical layer* contains all the functions needed to carry the bit stream over a physical medium to another system.

• The *Data Link layer* provides the abstraction of a link and organizes the bit stream into a data unit called a “frame”. This layer is further split into two sub-layers:

- *Logical Link Control* (LLC) which multiplexes protocols running atop the data link layer and optionally provides flow control, acknowledgment, and error recovery.
- *Medium Access Control* (MAC) that establishes which users is allowed to access the media at any one time, and can be implemented by centralized and distributed algorithms [10].

- The *Network Layer* introduces the concept of source-destination path and handles the routing of data flow through the network.

- The *Transport layer* provides a virtual end-to-end channel.

- The *Application Layer*Concerned with differences in internal representation, userinterfaces, and anything else that the user requires [10].

Sometimes; the Network layer called Internet work layer and the Data link and Physical layers regarded as one layer called Network layer, keeping the same functions at both situations.

The TCP/IP or Internet model is not the only standard way to build a protocol suite or stack. The Open Standard Interconnection (OSI) reference model is a seven-layer model that loosely maps into the five layers of TCP/IP. Until the Web became widely popular in the 1990s, the OSI reference model was proposed as the standard model for all communication networks. Today, the OSI reference model is often used as a learning tool to introduce the functions of TCP/IP[18].

Modularity and simplification of the network design are guaranteed by per-layer communications, i.e., given any two nodes in the network, layers at the same level communicate with each other without knowing the inner working at the lower layers, Per-layer communication allows optimizing the operations of each layer separately. However, in many systems there exists an inherent coupling between the operation at different layers, e.g., in wireless networks adjusting the resource allocation at the physical and MAC layers, such as transmit power or transmission rights, changes the average link rates, influences the optimal routing, and alters the achievable network utility. Under such coupling, optimizing only within layers will not be enough to achieve the optimal network performance, thus the control mechanisms at the transport, network, data link and physical layers need to be jointly designed [10].

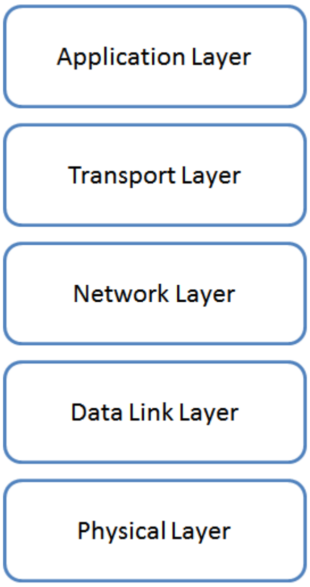

Figure (2-4): Layers of TCP/IP.

2-5 Error Control Techniques in Link and Physical Layers

In order to counteract wireless channel limitations such as fading, signal interference, limited bandwidth and available transmission power, MMR WiMAX, like most of other wireless technologies, employs error protection techniques implemented at the physical and the link layers. There are Three Techniques: FEC, ARQ and HARQ. By using them it becomes possible to compensate the link error rate by several orders of magnitude bringing it from $10^{-1} – 10^{-3}$ to $10^{-5} – 10^{-7}$. The achieved error rates become acceptable by the most widely used protocol reference model TCP/IP, originally designed for low error rate wired networks [6].

2-5-1 Forward Error Correction (FEC)

Is a technique used for controlling errors in data transmission over unreliable or noisy communication channels. The central idea is the sender encodes his message in a redundant way by using an Error-Correcting Code (ECC).

The redundancy allows the receiver to detect a limited number of errors that may occur anywhere in the message, and often to correct these errors without retransmission, but at the cost of a fixed, higher forward channel bandwidth [19].

The two main types of FECs are block codes and convolution codes.

2-5-1-1 Block codes

Work on fixed length blocks of bits or symbols of predetermined size. It encodes the data by multiple a complex matrix and in receiver, the decoder will reconstruct the data and correct the errors.

The code rate R = k/n(2-1)
where
- k is the length of useful information bits
- n is the total length codeword.

There are many types of block codes such as BCH codes, Hamming codes, Reed-Solomon, Block Turbo Code[20].

2-5-1-2 Convolution codes: work on bit or symbol streams, which have arbitrary length. They are used in numerous applications such as digital video, radio, mobile communication, and satellite communication. It is high speed and efficient.

Cyclic Redundancy Check (CRC) and Turbo Convolution Code are most important Convolution Code

In general; we can choose which types FECs can be used in our system by what we need. But Reed-Solomon coding is the most widely used because it strange ability in error control[20].

2-5-2 Automatic Repeat reQuest (ARQ)

Also known as Automatic Repeat Query, is an error-control method for data transmission that uses acknowledgements and timeouts to achieve reliable data transmission over an unreliable service. If the sender does not receive an acknowledgment before the timeout, it usually retransmits the frame/packet until the sender receives an acknowledgment or exceeds a predefined number of retransmissions.

The ARQ is mandatory for Mobile WiMAX systems. An ARQ block is assigned a sequence number (SN) or a Block Sequence Number (BSN) and is managed as a distinct entity by the ARQ state machines. The block size is a parameter negotiated during connection establishment [5].

Basically, there are three relay ARQs for multi-hop relay environment (shown in figure (2-5)[21]):

– End-to-End ARQ: ARQ is performed between BS and MS

– Two-Link ARQ:

In first link, ARQ is performed between BS and the last hop RS.

In second link, ARQ is performed between the last hop RS and MS[21].

– Hop-by-Hop ARQ: ARQ is initiated for each transmission hop, and each node requests the retransmission of the packet directly from the previous node[22].

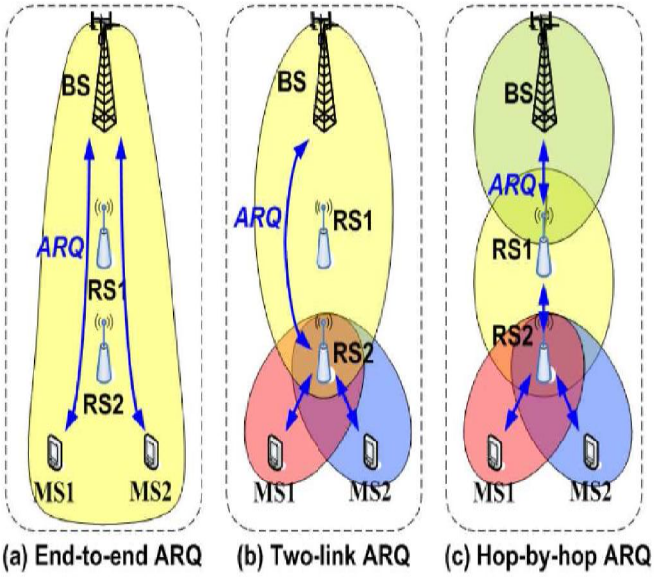

Figure (2-5): Types of ARQ for MMR Networks.

2-5-3 Hybrid ARQ (HARQ)

HARQ, implemented through a PHY-MAC cross layer mechanism in WiMAX, is a combination of FEC with ARQ technique. It uses not only retransmitted packets to reconstruct the original error free packets, but it also utilizes the packets received with errors. The original packet can be reconstructed by a combination of several versions of packet with errors[23].

HARQ implementation is organized according to a multichannel stop-and-wait approach where the sender should wait for the receiver feedback for every packet sent at the physical layer on any particular HARQ channel [6].

Error recovery technologies actually employed by MMR WiMAX are Automatic Repeat reQuest (ARQ) and Hybrid ARQ (HARQ)[5][6].

2-6 Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP) which is implemented at transport layer is used to retransmit the corrupted data segment. TCP is the most commonly transport protocol used in the Internet to provide end-to-end connection oriented and reliable services. Figure (2-6)[18] shows TCP packet header; where:

- **Source port**: Identifies the sending application.
- **Destination port**: Identifies the destination application.
- **Sequence number:** Used for assembling segmented data in the proper order at the receiving end.
- **Acknowledgement number**: Contains the sequence number of the next data octet that the TCP entity expects to receive.
- **Header length**: The TCP header length in 4-byte units.
- **Reserved**: Four bits are reserved for future use, should be set to zero.

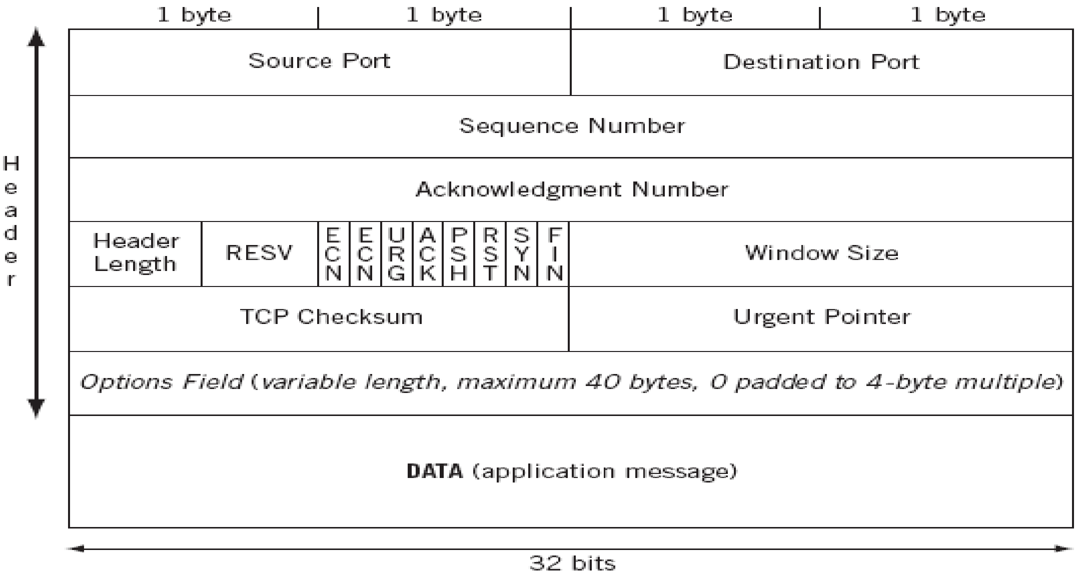

Figure (2-6): TCP Packet Header.

- **ECN flags:** The two explicit congestion notification (ECN) bits are used to tell the host when the network is experiencing congestion and send windows should be adjusted.
- **Flags** (also known as control bits): Contains six single-bit fields give the receiver more information on how to process the TCP segment:

  1- URG: Urgent pointer field is significant.

  2- ACK: Acknowledgement field is significant.

  3- PSH: Push function.

  4- RST: Reset the connection.

  5- SYN: Synchronize sequence numbers.

  6- FIN: No more data from sender.
- **Window size:** The number of bytes the sender is willing to receive starting from the acknowledgement field value. This field is used in TCP flow control and congestion control.
- **Checksum**: Used for error-checking of the header and data.

- **Urgent pointer**: If the URG flag is set, then this 16-bit field is an offset from the sequence number indicating the last urgent data byte.

Options and padding: TCP options are padded to a 4-byte boundary and can be a maximum of 40 bytes long. Generally, a 1-byte Type is followed by a 1-byte Length field (including these initial 2 bytes), and then the actual options

- **Data**: The contents of this field are the user data being transmitted between two application level entities[24].

TCP protocol is optimized to be used in wired networks where congestion is the only cause of packet loss. It responds to segments loss by retransmits the packets and shrinking its transmission window, and this reduces the throughput of TCP traffic. On the other hand, in wireless networks, packet loses are due to errors in link layer and disconnection due to handoff rather than congestion. In addition to that, TCP ACK reduces available bandwidth resources and takes a long time to be delivered causing a delay[15][25].

2-7 TCP Performance Enhancement Solutions for MMR Networks

Various approaches have been proposed to optimize the performance of IEEE 802.16j wireless networks. These can be broadly categorized into three groups:

2-7-1 Link Layer Solutions

The principle of this approach is to solve problems locally, with the transport layer not being made aware of the characteristics of the individual links. Such protocols attempt to hide losses in the wireless link to make it appear to be a highly reliable one. Link layer solutions require no changes in existing transport layer protocols. The proposed solutions for the link layer can be classified according to their awareness of the transport layer protocol [4]:

2-7-1-1 TCP-unaware Link Layer Protocol

Optimizes the link layer by hiding the differences between wired and wireless mediums so that the transport layer can operate as if it is installed in a wired net-work. This method does not violate the modularity of the protocol stack, however, since the necessary adaptations improve the reliability independent of higher-layer protocols. Nonetheless, this lack of awareness can affect performance under certain specific conditions.

2-7-1-2 TCP-aware Link Layer Protocol

The TCP-aware link layer protocol presents certain advantages since knowledge of the protocol operating at the transport level allows fine tuning of the performance. For instance, TCP-unaware approach may trigger retransmissions at both link and transport layers simultaneously.

2-7-2 Transport Layer Solutions

The theory underlying this approach is the modification of the transport protocol in order to achieve high throughput on wireless links. Since some packets may be lost, the modified transport protocol should implement congestion control as a reaction to packet losses, moreover, other schemes should be implemented to consider the peculiarities of the wireless environment. TCP was originally designed for wired networks, where packet losses are caused mostly to network congestion, rather than errors resulting from noisy channels, handoffs and node mobility. A reduction in the congestion window is thus the TCP reaction to packet loss of any kind. TCP modifications are logically divided into two groups according to the technique they introduce [4]:

2-7-2-1 Connection Splitting Approach

In this scheme, the end-to-end TCP connection is divided into fixed and wireless parts, so that more degrees of freedom are available for the optimization of TCP over both wired and wire-less links.

The disadvantages of connection splitting mainly involve the attempt to perform transparent from the wired node splitting of TCP. This leads to greater complexity in base station procedures, which is the most common and suitable place for splitting; the greater complexity involves not only handoff handling but also, prevention of end-to-end semantics of the TCP connection and, also greater software overhead caused by the TCP part of the stack involved at the intermediate point.

2-7-2-2 TCP Modifications

TCP Modification involves a group of solutions which promote small changes to TCP behavior, such as the mechanics of acknowledgement generation used by TCP. TCP modification does not require modification of the BS avoiding overhead in packet delivery and BS complexity increase.

2-7-3 Cross-Layer Solutions

Traditionally, protocol architectures follow strict layering principles, which ensure interoperability, fast deployment, and efficient implementations. However, lack of coordination between layers limits the performance of such architectures due to the specific challenges posed by wireless nature of the transmission links.

Cross-layer solutions break the principles of layering by allowing interdependence and joint development of protocols involving various different layers of the protocol stack. It enables layers to exchange state information to adapt and optimize the performance of the network. The sharing of information enables each layer to have a global picture of the constraints and characteristics of the network, leads to better coordination, and enables them to make decisions that would jointly optimize the performance of the network [26] considering two points:

1- What information should be exchanged across protocol layers, and how should that information be adapted?

2- How should the global system constraints and characteristics be factored into the protocol designs at each layer?

Two general approaches to cross-layer interactions are:

1- Bottom-to-top: higher layers are notified with details related to the network they operate on.
2- Top-to-bottom: allows lower layers to access information available at the upper layers.

2-8 Related Works on Cross Layer Design

BER in the physical layer can be used by a network layer as a guide in changing the interface.Ramachandran and Shanmugavel[27]identified the received signal strength of route reply (RREP); it's received signal strength information parameter is passed from the physical layer to the network layer. Hence, the nodes receiving the RREP packet calculate path loss experienced by the packet and then compute the minimum required transmission power stored in the routing table along with the next hop against the destination. This parameter mitigates unnecessary link/route failures as transmission power is updated on a per packet basis. In addition, the received signal strength information is used in choosing reliable links to form stable routes. This procedure enables the routing layer to find paths that offer the lowest levels of generated interference correctly, which increases reliability in terms of packet transmission success rates and highest achievable transmission rates. Hence, this method decreases packet loss.

In exchanging retransmission information among transport and link layers, the packet loss rate will be reduced, such as in [15], which introduced a cross-layer design for maintaining the TCP packet loss probability in IEEE

802.16j MMR WiMAX based on a given channelcondition and user location threshold. The result shows that TCP packet loss probability is reduced by determining a suitable ARQ packet length for a given number of retransmissions.

An Application to MAC cross-layer video adaptation solution used for optimizing network resource consumption and user experienced quality of video streaming in wireless networks is proposed [28]. The proposed solution utilizes the flexibility of the Scalable Video Coding (SVC) technology and combines fast and fair MAC layer packet scheduling with long-term application layer adaptation. Thus the usage of network resources is improved by dropping video data based on its priority when congestion occurs. As a result, application layer adaptation gains over 60% less base layer losses, because of SVC, than in the case without any adaptation.

A new Network to link/MAC cross-layer QoS provisioning framework for Wireless MultimediaSensor Networks is proposed in [29]. Although the objective of the network layer is to obtain QoS-routes with the application of specific QoS requirements, the link/MAC uses this routing information for actual packet classification and delivery.

The author in [30]proposes a time-synchronized sensors form on-off schedules that enable the sensors to be awake only when necessary.It exploits the time scale difference between sensor network reconfiguration periods and data forwarding periods. The schedule establishment is fully distributed and thus appropriate for large sensor networks. The author mainly focuses on link-level transmission schemes and network/MAC layer resource management policies.

CHAPTER THREE

ARQ CROSS LAYER DESIGN

CHAPTER THREE

ARQ CROSS LAYER DESIGN

3-1 Transmission Packet over MMR WiMAX Network

Error recovery technologies employed by MMR WiMAX are classified into ARQ and HARQ[5][6].

Figure (3-1) illustrates a single TCP DATA packet delivered from the fixed part of the network to the Mobile WiMAX Host (MH), passing through the BS forwarding it to the appropriate MH over the radio link.

This downlink transmission includes also heavy physical and link layer overhead. In case of successful reception, the MH sends an HARQ ACK response to the BS and forwards the received TCP data packet up to the TCP layer of the protocol stack. Consequently, the TCP layer generates a TCP ACK.

This one represents ordinary payload for the MMR WiMAX link layer: before it can be transmitted on the wireless link, uplink resources should be requested and a corresponding assignment grant should be received at the MAC layer. Moreover, TCP ACK transmission requires an HARQ ACK from the BS.

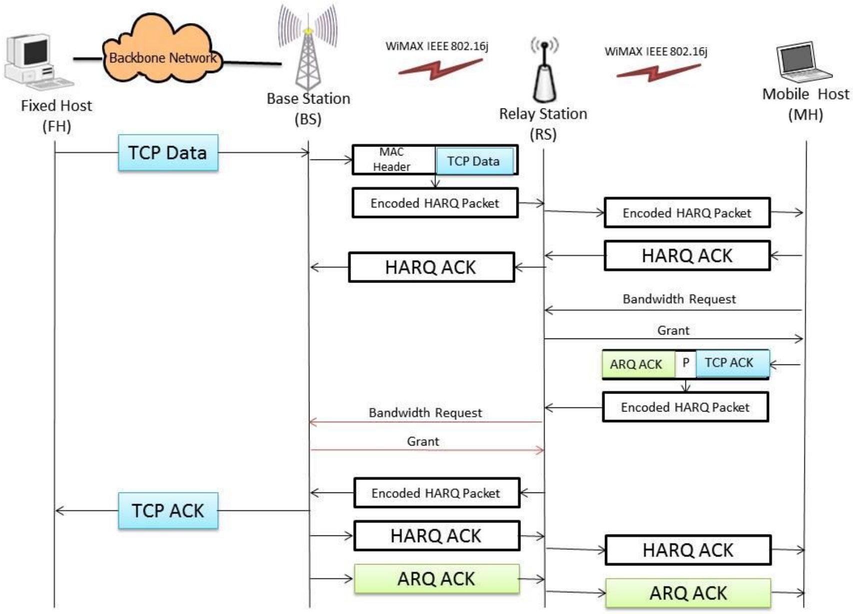

Figure (3-1): TCP Packet over MMR WiMAX

Summarizing, a single transport-layer data transmission is acknowledged three times: once at the transport layer and two times at lower layers. Moreover, while HARQ feedback is relatively small it is several bits, the TCP ACK along with the headers added at the network, link and physical layers consumes large amount of network resources. In case ARQ and HARQ are enabled on a MAC connection carrying a TCP flow, multilayer error recovery at the MAC and transport layers takes place. Although this increases reliability in the wireless environment, the ARQ acknowledgement feedback consumes a significant amount of radio resources.

More precisely:

- A standalone TCP ACK requires at least 46 bytes (20 for TCP, 20 for IP, 6 for MAC headers). If ARQ is enabled, there are another 4 bytes for the

ARQ CRC. Instead, a piggybacked TCP ACK entails no overhead since ACK information is provided inside TCP header.

• An UL HARQ acknowledgement occupies half a considerable portion of the uplink slot, while a downlink HARQ ACK is only one bit placed in the downlink HARQ ACK IE contained in the DL-MAP message.

• The size of an ARQ acknowledgement depends on the acknowledgement scheme employed and is quite variable. However its minimum value is 4 bytes[23].

For underlining this aspect, Figure (3-1) shows the transmission, reception and acknowledgement of a TCP (+IP) data packet in a WiMAX network when both ARQ and HARQ are enabled. In such scenario, a single TCP data packet is acknowledged five times: one at the transport layer and four at MAC layer. As a consequence, this multilayer error recovery consumes a significant amount of resources, especially in the uplink direction.

3-2 Proxy in MMR WiMAX Networks

The basic idea behind the proposed ARQ proxy approach is to avoid the transmission of standalone TCP ACK packets over the radio link between the BS and MH together with associated overhead added at the physical and link layers including request/response exchange at the MAC layer as illustrated in Figure (3-2). In order to achieve this goal; no changes are needed to the TCP protocol, but new software entities need to be introduced within the protocol stack: the ARQ Proxy and ARQ Client.

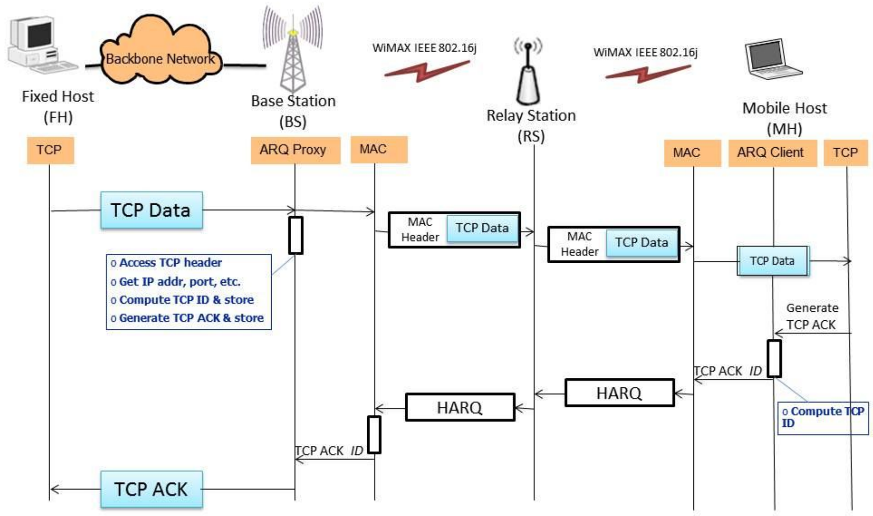

Figure (3-2): ARQ Proxy Approach in MMR WiMAX

ARQ Proxy is a software module located in the protocol stack of the wireless BS. It accesses TCP and IP headers of the transmitted packets and generates a TCP ACK for every TCP data packet destined to MH to confirm successful data reception up to the flow segment carried in this TCP data packet.

TCP ACK is generated using a simple memory copy operation applied to the fields (IP addresses, port numbers, and flow sequence numbers) (refer to figure 2-6) of the transmitted TCP data packet into a previously generated template of TCP ACK. As a result, there is no conventional TCP layer functionalities need to be supported at the BS, and no TCP state related information needs to be stored. This allows ARQ proxy approach scales well in large network systems supporting high load at the BSs.

Since not always all the segments are successfully received, TCP ACKs generated by ARQ Proxy module are not released to the FH immediately, but stored in BS memory until requested by the ARQ Client. Figure (3-3) shows what is going on insight the BS.

ARQ Client is a software module which located between the link and transport layers of the MH protocol stack. Its purpose is to suppress all outgoing standalone TCP ACK packets and replace them with MAC layer requests.

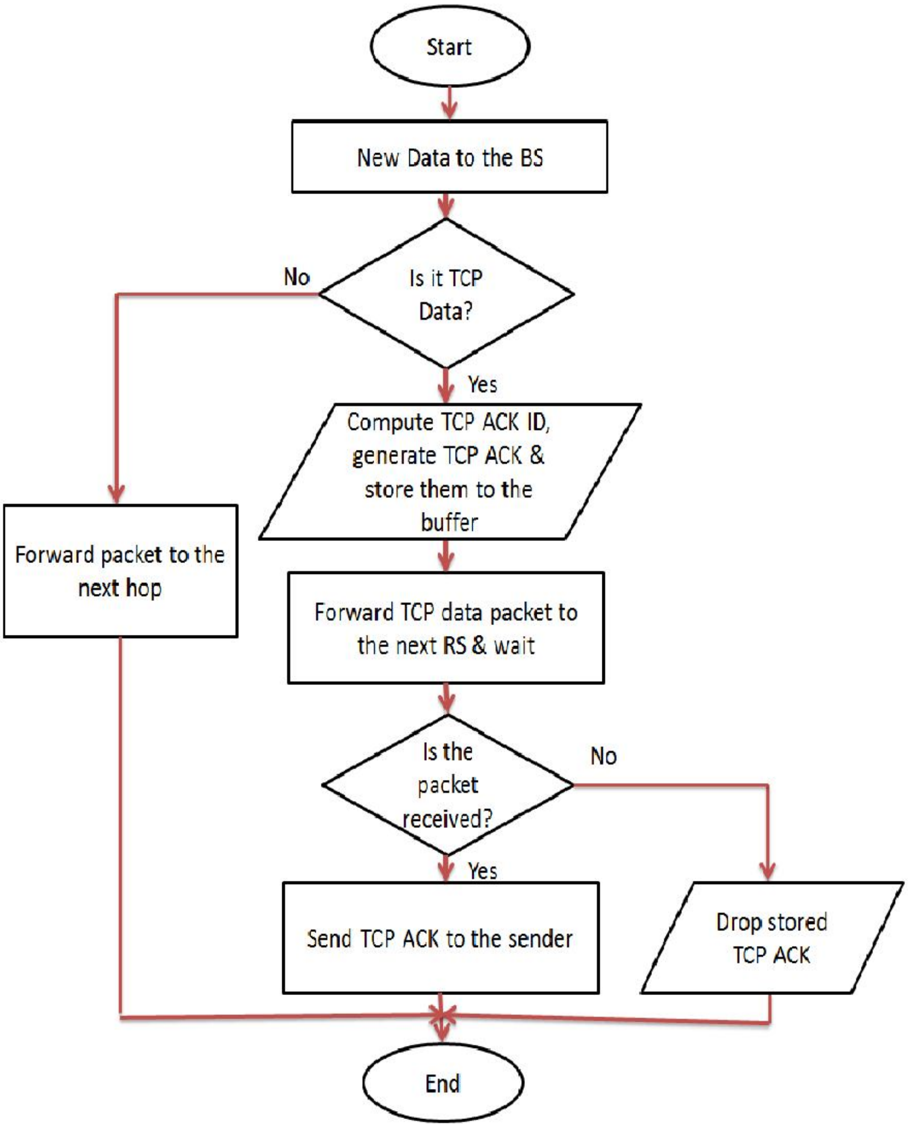
Start
New Data to the BS
Is it TCP Data?
No
Yes
Compute TCP ACK ID, generate TCP ACK & store them to the buffer
Forward packet to the next hop
Forward TCP data packet to the next RS & wait
Is the packet received?
No
Yes
Send TCP ACK to the sender
Drop stored TCP ACK
End

Figure (3-3): ARQ Agent Flowchart

In order to do so, whenever a standalone TCP ACK is produced at the MH transport layer, a TCP ACK suppression request is immediately scheduled for transmission at the link layer, while the original TCP ACK packet travels down the protocol stack which involves corresponding processing at each layer, output queuing delay, shared medium access and other procedures.

Whichever comes first to the physical layer: the TCP ACK or the corresponding suppression request; will be transmitted, while the other one cancelled.Figure (3-4) shows ARQ Client position and operation in MH's protocol stack.

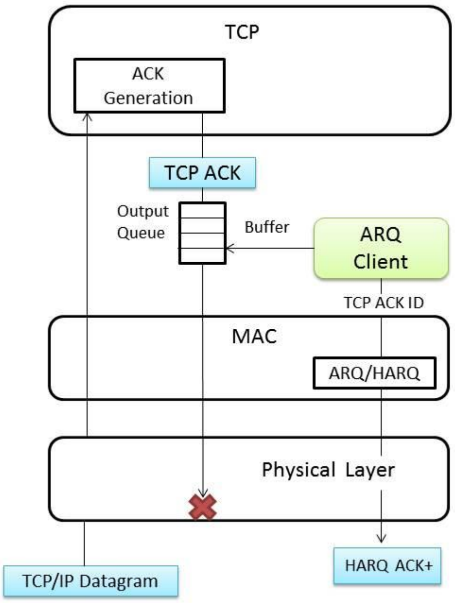

Figure (3-4): ARQ Clients Position within the MS's Protocol Stack [31].

In order to trigger TCP ACK transmission, the ARQ Client module implemented at MH should clearly specify which of the TCP ACKs generated by the ARQ Proxy should be transmitted to the fixed host. For that reason, TCP ACK suppression request contains a TCP ACK identification (afterwards referred to as TCP ACK ID).

TCP ACK ID should be easily computable at both BS and MH without the need for direct communication.

3-3 Performance Evaluation Metrics

In order to estimate the system performance after applying the proposed solution and to have tangible results some performance metrics are focused on: the round trip time, the number of timeouts, system throughput and the efficiency.Here are some equations used in the MATLAB code related to the considered performance metrics:

- **Delay Estimation:** In IEEE 802.16j MMRWiMAX networks the increased number of hops between the user and the MMR-BS introduce much delay and signaling overhead which affect the performance of services offered by WiMAX networks.The delay associated a data packet a cross n hops is given by [15]:

  $$D = 4 * t + (n - 1) * t \quad (3\text{-}1)$$

  Where t is the frame duration.

  Delay affects many other metrics such as RTT and the number of time outs.

- **Packet Error Rate:** The ARQ packet loss probability is a function of the link status measured as Bit Error Rate (BER) and packet size (k) in bits, as shown in (3-2)from [15].

$$P = 1 - (1 - BER)^{K} \quad (3\text{-}2)$$

- **Throughput Calculation:** We also have the formula (3-3) introduced in [32] to predict the TCP throughput. In which, M is the flow’s segment size, b is the number of TCP segments per new ACK, p is the packet loss rate, RTT is the Round Trip Time before the flow, and RTO is the TCP retransmission timeout period.

$$TH_{tcp} = \frac{M}{RTT\sqrt{\frac{2bp}{3} + RTO*\min\left(1,\frac{3bp}{8}\right)*p(1 + 32\,p^{2})}} \quad (3\text{-}3)$$

The TCP Retransmission Timeout (RTO) which calculated in the standard depending on the value of RTT as follow:

$$S(k) = (1 - a) * S(k - 1) + a * T(k) \quad (3\text{-}4)$$

$$V(k) = (1 - b) * V(k - 1) + b * |T(k) - S(k)| \quad (3\text{-}5)$$

$$RTO = S(k) + m * V(k) \quad (3\text{-}6)$$

Where:

S(k) : The smoothed mean deviation of the RTT

V(k) : Smoothed average of RTT

RTO : The Retransmission Out value:

a : exponential smoothing parameter (1/8).

ß : smoothing parameter (1/4).

m = 4 in the standard.

- **Spectral Efficiency:** Spectral Efficiency refers to the information rate that can be transmitted over a given bandwidth in a specific communication system [33]. The maximum efficiency that can be achieved is:

  $$\eta s = TH/B \qquad (3\text{-}7)$$

  where:

  TH: Throughput in bit/sec.

  B: Bandwidth in HZ.

CHAPTER FOUR

RESULTS AND DISCUSSION

CHAPTER FOUR

RESULTS AND DISCUSSION

ARQ_Proxy is a general solution designed for any network implementing multiple positive ARQ schemes at different layers. However, as a reference scenario for performance evaluation of the proposed approach, IEEE 802.16j is chosen as the most wide-spread and well-analyzed environment nowadays. Evaluation results are obtained through simulations in MATLAP code.

RTT and average throughput of a single TCP connection is evaluated during the simulation time in both situations: Normal TCP and Cross Layer design. Also; a comparison of timeouts number in the two hobs. In addition to that; the Spectral Efficiency is calculated every second.

4-1 System Simulation Model

The considered scenario in this study is for a one branch of a MMR WiMAX network (IEEE 802.16j) with a MMR-BS, two relays and three MSs distributed at all hop levelsas illustrated in Figure (4-1).

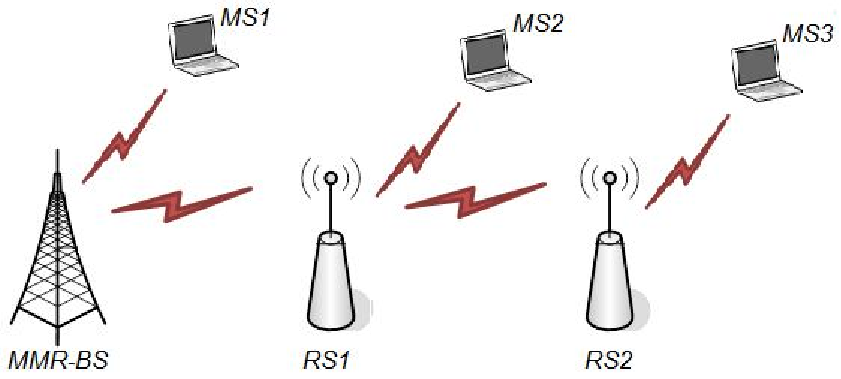

Figure (4-1): MMR WiMAX Simulation Model

4-2 Simulation Design

An executable M-file is written using MATLAB version 7.5.0 (R2007b) implementing the main physical and MAC layers features of 802.16j, considering TCP as a transport layer protocol and adjusting the parameters in table 4-1.

Table (4-1): Simulation Parameters

Parameter	Setting
Bandwidth	20 MHz
FFT size	2048
OFDMA frame duration	20 ms
Frames per second	50
OFDMA symbols	198

Duplexing Mode	TDD
TCP segment size	1024 bytes
Fragmentation/packing	On
CRC/ARQ	On
Simulation time	5 sec
ARQ block size	1024 bytes
ARQ window size for 1st/2nd hops	(3-20) packets
ARQ retry time for 1st/2nd hops	(40-80) ms
ARQ block lifetime for 1st/2nd hops	Adaptive ms

4-3 Simulation scenario

The simulation is implemented starting with setting the initial parameters such as Bandwidth, Frame Duration, ARQ Block Size … etc (as in Table (4-1)).

Window size is assigned in the beginning and then readjusted during the transmission.

Each TCP packet is divided into four fragments and they are put in the ARQ window to be sent. Sender expects to receive the TCP ACK after one frame duration for the first hop and two frame duration for the second hop.

Different Packet Error Rates (PERs) are assumed for each link and a random function is used to select some packets to be acknowledged negatively.

The end user specifies the type of ACKs sends them on a queue and the BS distinguishes them by what it receives from the end user where the value “One” is for positive ACK and “Two” for NACK and it puts “Three” for timeout.

The sender retransmits packets with NACK or its time is out. When it received three duplicate ACKs, it defines Slow-Start Threshold (ssthresh)and Congestion window andreadjusts its window size.

4-4 Simulation Results

The main objective of using ARQproxy approach in WiMAX 802.16j is to improve the overall Performance and system capacity. Measuring some of the performance parameters would show the improvement of the MMR WiMAX network.

4-4-1Round Trip Time RTT

The simulation of proposed ARQ optimization technique reduces the overhead associated with ARQ process implemented at higher layer based on the link layer ARQ feedback; it generates TCP acknowledgement based on the successful packet delivery indication received from the link layer. This eliminates the time associated with TCP ACK transmission over the radio link, including uplink bandwidth reservation delay, then it reduces RTT.

Figures (4-2a) and (4-2b) show RTT of the users in the first and second hop respectively; for cross layer ARQ and normal TCP, in which, Along the simulation time the cross layer ARQ gives less RTT by –in average- 73% in the first hop and 50% in the second hop as compared with the normal TCP.

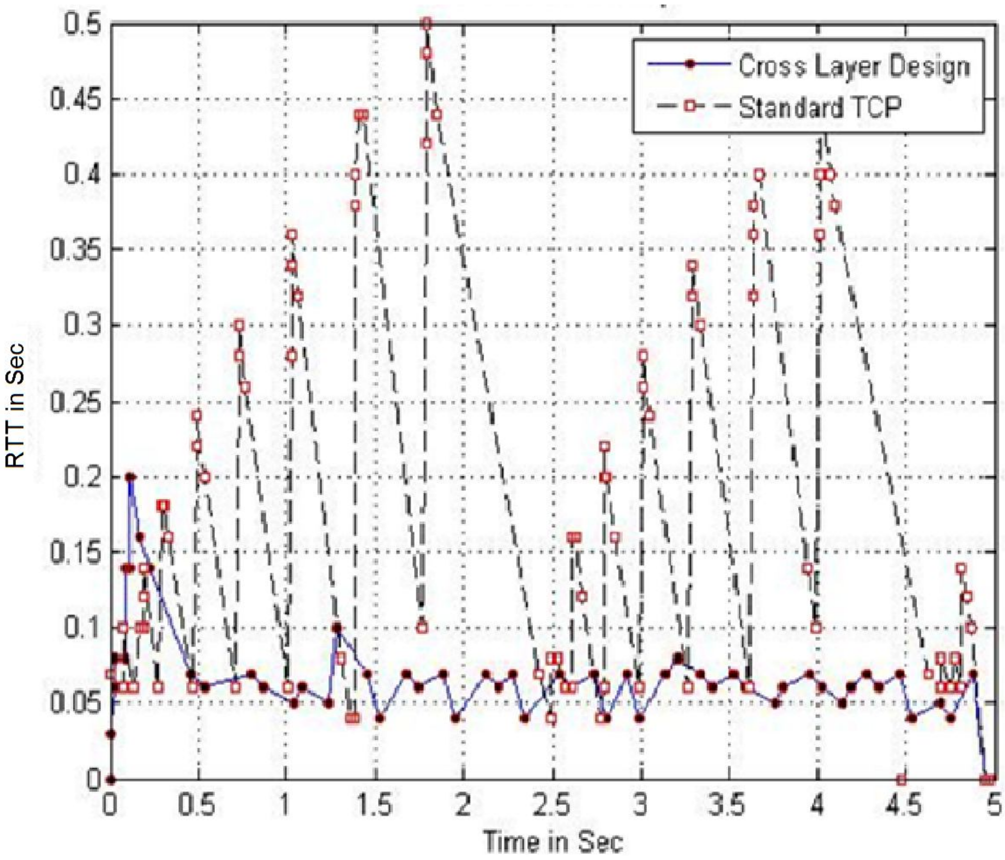

Figure (4-2a): RTT in the first hop

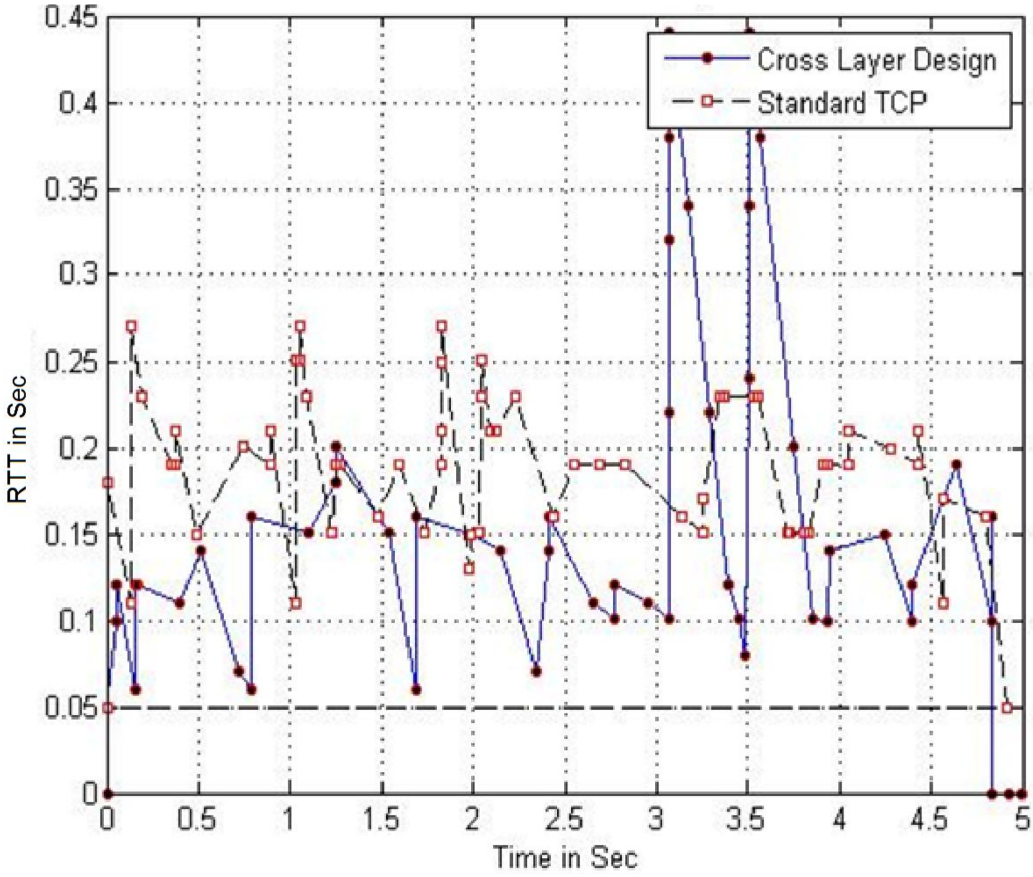

Figure (4-2b): RTT in the second hop

4-4-2 Average Throughput

TCP throughput depends on the RTT between the sender and the receiver as shown in equation (3-2) and that is logically since the reduction of the delay gives opportunity for sending more data packets.

Figures (4-3a) and (4-3b) represent the average throughput in cross layer design and standard TCP for the first and second hop respectively, as the comparison in percentage is calculated for the two hops in table 4-2.

But as noticed in figure (4-3b); sometimes throughput in the case of standard TCP exceeds throughput of cross layer, this approximately because the cross layer approach does not affect BER of the link, and also because of the wasted time in the case of NACK and timeout packets; that the BS had already spent time in manipulating them by generating there ACKs, storing them and then deleting them.

The existence of some very high values of throughput may refer to the non-accuracy of the MATLAB code.

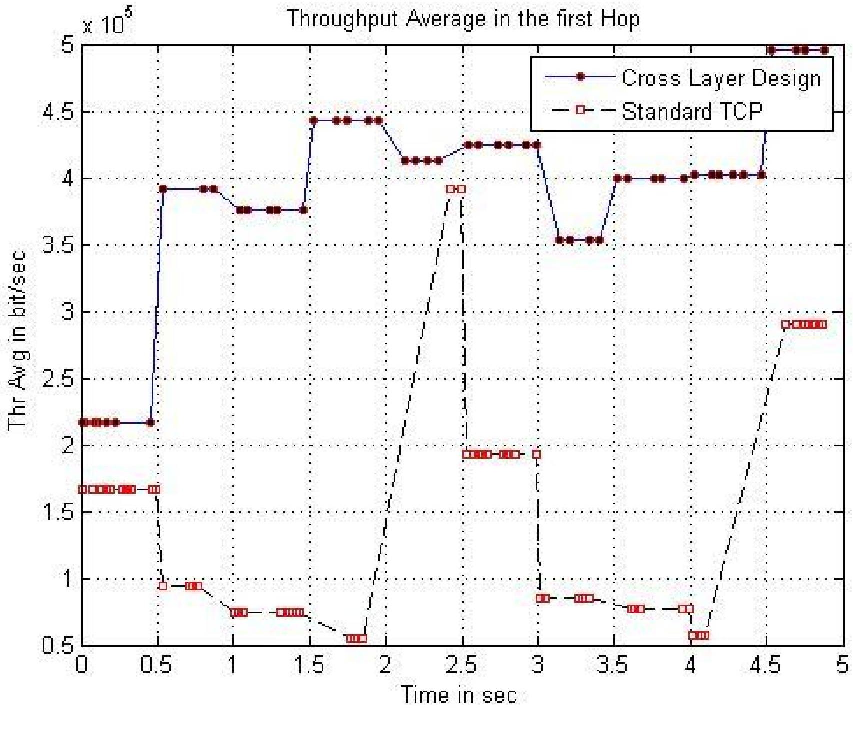

Figure (4-3a): A comparison of Throughput Avg. between normal TCP and ARQ cross layer design in the first hop

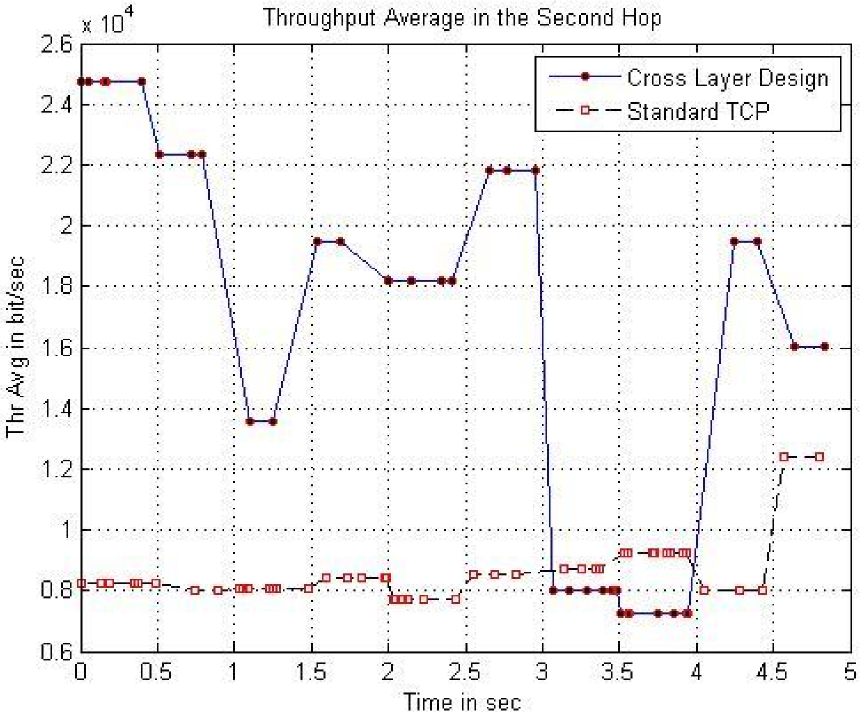

Figure (4-3b): A comparison of throughput avg. between normal TCP and ARQ cross layer design in the Second Hop

Table 4-2: A comparison of Throughput Avg. between normal TCP and ARQ cross layer design in percentage for the first and second hops

time in Sec / Hop	0-	0.5-	1-	1.5-	2-	2.5-	3-	3.5-	4-	4.5-
The first hop	30	316	403	715	5	120	319	422	606	71
The second hop	200	179	68	131	136	155	-7	-21	143	29

4-4-3 Number of Timeouts

Cross-Layer ARQ approach avoids the transmission of TCP ACK packets over the wireless channel, which releases network resources to send other ARQ ACKs before its timeout. In addition to that; the substitution of the TCP ACK with a short ARQ packet reduces the probability of losing ACK packets then reducing the timeout occurrence. This is obvious in figure (4-4) where we notice that the number of timeous in cross layer design is less Twelve Times than in standard TCPin the first hop and One and half in the second hop.

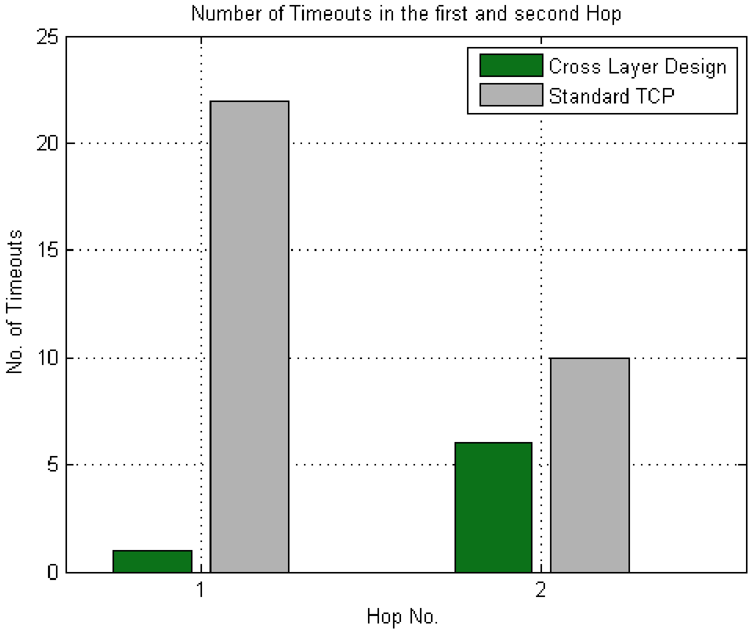

Figure (4-4): A comparison of Number of Timeouts between normal TCP and ARQ Cross Layer design in the First and Second Hops.

4-4-4 Spectral Efficiency

Since there are an increment in the system throughput and better resources utilization; it is expected that the Efficiency would be improved. Calculated results of the simulation in Figure (4-5a) and (4-5b) with the table 4-3 prove that there is improvement in the first hop and second hop.

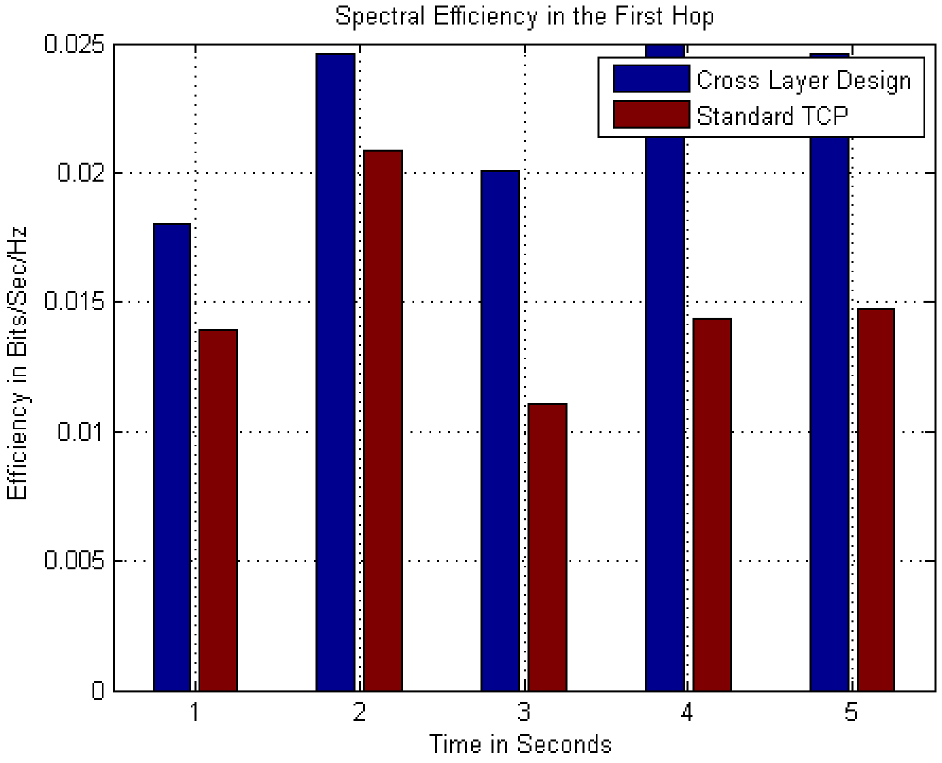

Figure (4-5a): A comparison of Spectral Efficiency between normal TCP and ARQ cross layer design in the first hop

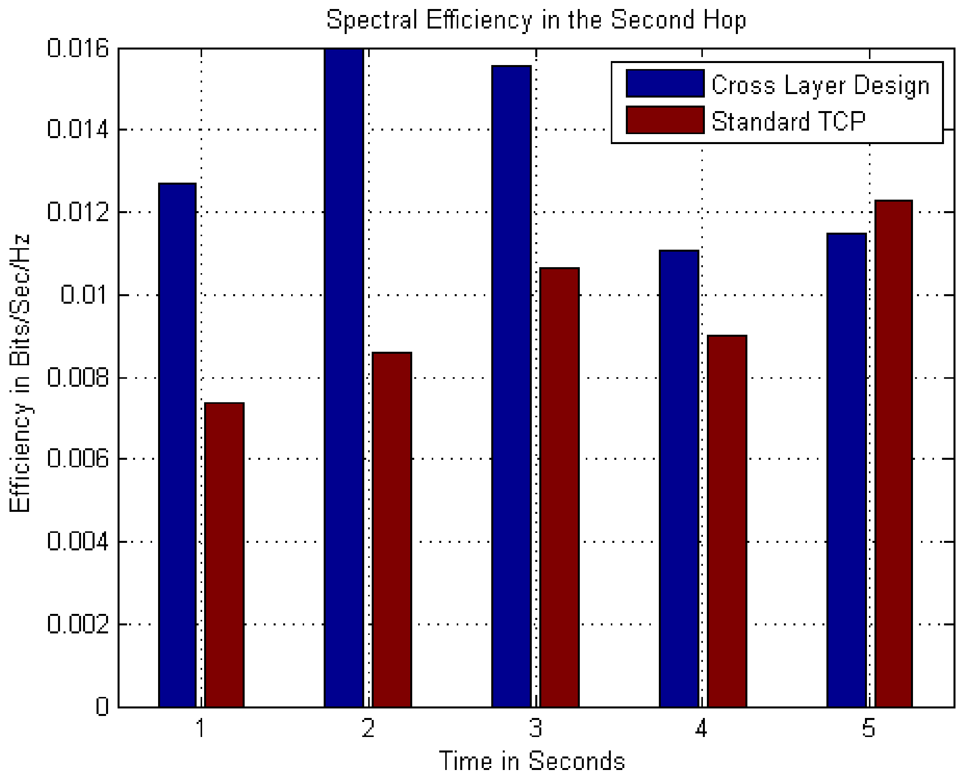

Figure (4-5b): A comparison of Spectral Efficiency between normal TCP and ARQ cross layer design in the Second Hop

Table 4-3: A comparison of Spectral Efficiencybetween normal TCP and ARQ cross layer design in percentage for the first and second hops

time Sec Hop	1	2	3	4	5
The first hop	28	16	81	78	68
The second hop	73	81	44	22	-4

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5-1 CONCLUSION

This thesis studies an approach for performance enhancement of TCP over MMR WiMAX network (802.16j). Performance improvement comes from cross-layer optimization of ARQ schemes employed at different layers of the protocol stack.

The proposed solution named Cross-Layer ARQ or ARQproxy avoids TCP ACK transmission over the wireless link through local generation of ACKs.

The evaluation of the proposed approach is performed via MATLAB Simulation Code written in M-file in IEEE 802.16j scenario. Results demonstrate an improvement in the throughput in the range of 25-40%.

Summarizing, main improvements of Cross-layer ARQ are:

- *Increment of system Throughput.*
- *Round Trip Time (RTT) reduction.*
- *Decrement in the Number of Timeouts.*
- *System Efficiency Enhancement.*

Cross-layer ARQ is not limited to wireless networks scenarios. In fact it can be implemented inside any protocol stack employing multiple ARQ schemes at different layers. However, the main benefits are achieved over wireless links due to heavy overheads added at the link and physical layers.

Wish that this thesis research would be a step forward in the tremendously growing field of cross-layer design. It is relatively new to the research community, and currently it evolves. However, there is a tradeoff between benefits of performance enhancements and future design complexity.

5-2 RECOMMENDATIONS

Finally; there are many area of research in the field of Cross Layer design in general and in ARQ Cross Layer specifically. For the future work in the same field we recommend:

- Tacking Mobility and handover into consideration.
- Considering QoS Parameters.
- Appling the design in a real network in order to have accurate results and study the effects of Cross layer design in other aspects that we could not observe with the simulation code.
- Initiating a quantitative approach for calculating the sensitivity of system performance with the respect to parameters across different layers for VoIP over Wireless WAN.

REFRENCES

[1] IEEE Std. 802.16e-2005, IEEE Standard for Local and Metropolitan Area Networks, part 16, "**Air Interface for Fixed and Mobile Broadband Wireless Access Systems**", IEEE Press, December 2005.

[2]ChutimaPrommak and ChitapongWechtaison," **Network Planning and Optimization for Multi-Hop Relay Placement in WiMAX Networks**", Journal of Computer Science, 2012.

[3] K.Pentikousis, "**TCP in Wired-Cum-Wireless**", IEEE Com-munications Surveys, 2000.

[4] DzmitryKliazovich, "**CROSS-LAYER PERFORMANCE OPTIMIZATION IN WIRELESS LOCAL AREA NETWORKS**", PhD Dissertation, 2007.

[5] Shwetha D1, Thontadharya H. J2, Subramanya Bhat.M3 and Devaraju J.T4, "**PERFORMANCE ANALYSIS OF ARQ MECHANISM IN WiMAX NETWORKS",** *International Journal of Computer Science & Communication Networks,Vol 1(2), 123-12, Oct 2011*.

[6] DzmitryKliazovich, TommasoBeniero, Sergio Dalsass, Federico Serrelli, Simone Redana, and Fabrizio, "**Cross-Layer Error Control Optimization in WiMAX**", IEEE "GLOBECOM", 2008.

[7]FabrizioGranelli, DzmitryKliazovich, "**Cross-Layering for Performance Improvement in Multi-Hop Wireless Networks**", Journal of Interconnection Networks, 2005.

[8] F. Granelli, D. Kliazovich and Nelson L. S. da Fonseca, "**Performance Limitations of IEEE 802.11 Networks and Potential Enhancements**", Nova Science Publishers, Hardbound, 2005.

[9]HalahMahmoud,“**Wireless network and its impact**”, Bloggers API, December 2012.

[10] Pablo Soldati, “**Cross-Layer Optimization of Wireless Multi-Hop Networks**”, *KTH School of Electrical Engineering, Stockholm, June 2007.*

[11] ProfNizar Zorba, CharalambosSkianis and Christos Verikoukis, “**CROSS LAYER DESIGNS IN WLAN SYSTEMS**”,*TroubadorPuplishing LTD, UK, 2010.*

[12]Dr.A.S.Sodiya, “**NETWORKING AND COMMUNICATION TECHNOLOGY**”, National OpenUniversityof Nigeria, 2012.

[13]Ajoy Paul, “**WARELESS INTERNET SERVICE**”, Academic Information, November 2011.

[14] http://www.radio-electronics.com/info/wireless/wimax/ieee-802-16-standards.php 5/1, 23/12/2014

[15] FathElrahman Ismael, Sharifah K. Syed Yusof, Mazlan Abbas, Adnan Shahid Khan, and NorsheilaFisal, ‘**TCP Packet Loss Rate Reduction for MMR WiMAX Networks using Cross Layer Design’**, *3rd International Graduate Conference on Engineering, Science and Humanities (IGCESH), School of Graduate Studies UniversitiTeknologi Malaysia, November 2010.*

[16] D.Satish Kumar, Dr.N.Nagarajan, “**Relay Technologies in IEEE 802.16j Mobile Multi-hop Relay (MMR) Networks”,** *Computer Engineering and Intelligent Systems*www.iiste.org*, Vol 2, No.3, ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online).*

[17]*Steven W. Peters and Robert W. Heath,* **“The Future of WiMAX: Multihop Relaying with IEEE 802.16j”,**IEEE Communications Magazine, January 2009.

[18] Walter Goralski, “**The Illustrated Network: How TCP Works in A modern Network**”, Morgan Kaufmann Publishers, 2008.

[19] Firoz Ahmed Mansuri, "**Performance Analysis of Acoustic Signal in OFDM Using FEC Codes**", International Journal of Digital Application & Contemporary research, September 2013.

[20] Bo Rong, Yiyan Wu, and Gilles Gagnon,Information Theory, **"Shannon Limit and Error Correction Codes for Terrestrial DTV Broadcasting"**, Communications Research Centre Canada, October 2011.

[21] Tzu-Ming Lin, **"Error Control Strategies for WiMAXMulti-Hop Relay Networks"**,IEEE GLOBECOM, 2009

[22] Liping Wang, Error Control in Relay Networks, EP2200 Course Project , November 2012

[23] DzmitryKliazovich, Simone Redana and FabrizioGranelli, "**Cross-Layer Error Control Optimization in WiMAX**", the IEEE "GLOBECOM", 2008.

[24] Behourz A. Forouzan, "**The TCP/IP Protocol Suite"**, Fujitsu Network Communications, December 2006.

[25] ArathiShamakumar, Ravi Kiran, Srinivas and SirivasaRaoSegu**, "TCP for Wireless Networks"**, University of Texas at Dallas, November 2001.

[26] Prof. Wei Liu, "**Wireless Ad hoc Network",** *Huazhong University of Science and Technology, Jun 2009.*

[27] B. Ramachandran and S. Shanmugavel, "**Received Signal Strength-based Cross-layer Designs for Mobile Ad Hoc Networks**", IETE Technical Review, Vol. 25, No. 4, pp. 192-200, 2008.

[28] TiiaSutinen, JanneVehkaperä, EsaPiri and MikkoUitto, **"Towards ubiquitous video services through scalable video coding and cross-layer optimization"**, EURASIP Journal on WirelessCommunications and Networking,, January 2012.

[29] NavratiSaxena, Abhishek Roy and Jitae Shin, “**Cross-Layer Algorithms for QoS Enhancement inWireless Multimedia Sensor Networks”**, IEICE Transactions on Communications, Vol.E91.B,No.8, pp.2716–2719, August 2008.

[30] Mihail L. Sichitiu, **“Cross-Layer Scheduling for Power Efficiency in Wireless Sensor Networks**”, IEEE INFOCOM 2004.

[31] DzmitryKliazovich, FabrizioGranelli and Mario Gerla, “**Performance improvement in wireless networksusing cross-layer ARQ**”, Computer Networks 51 (2007) 4396–4411 ,2007.

[32] FathElrahman Ismael, Sharifah K. Syed Yusof , Mazlan Abbas, Adnan Shahid Khan, and NorsheilaFisal , “**Throughput Enhancement in MMR WiMAX IEEE 802.16j Networks**” , 3rd International Graduate Conference on Engineering “Malaysia” , 2010.

[33] Guowang Miao and Guocong Song, “**Energy and Spectral Efficient Wireless Network Design**”, Kambridge University Press, 2015.

Appendix A

A1 RTT in the First and Second Hops

```
tx11(h1,3)=ttime;
tx11(h1,4)=tx11(h1,3)-tx11(h1,2);


figure ;
plot (tx11(:,2),tx11(:,4),'bo-','Linewidth', 1
,'MarkerEdgeColor','r','MarkerFaceColor','k','Markersize', 4);
holdon

plot (tx1(:,2),tx1(:,4),'ks--','Linewidth', 1
,'MarkerEdgeColor','r','MarkerFaceColor','w','Markersize', 4);
holdoff

title (' RRT in the First Hop');
xlabel (' Time in Sec')
ylabel ('RRT in Sec')
legend ('Cross Layer Design','Standard TCP')
gridon
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

tx1(h1,3)=ttime;
tx1(h1,4)=tx11(h1,3)-tx1(h1,2);


figure ;
plot (tx22(:,2),tx22(:,4),'bo-','Linewidth', 1 ,'MarkerEdgeColor','r'
,'MarkerFaceColor','k','Markersize', 4);
holdon

plot (tx2(:,2),tx2(:,4),'ks--','Linewidth', 1
,'MarkerEdgeColor','r','MarkerFaceColor','w','Markersize', 4);
holdoff


title (' RRT in the Second Hop');
xlabel (' Time in Sec')
ylabel ('RRT in Sec')
legend ('Cross Layer Design','Standard TCP')
gridon


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A2 Average of Throughput in the First and Second Hops

```
n1=0;
n2=0;
c=0;
s=1;
rr11=0;
m11=0;

while n2<=5
   n=0;      %%RTT
   m=0;      %%RTO
   x=0;
   n2=n1+0.5;
forjj=s:(length (tx11(:,4)))
if tx11(jj,4)~= 0 && tx11(jj,2)<n2

        n=n+tx11(jj,4);
        m=m+tx11(jj,7);
        x=x+1;
        s=s+1;
end
end
   c=c+1;
    rr11(c)=n/x;
    m11(c)=m/x;

    th_11(c,1) =
(256*8)/(rr11(c)*real((sqrt((2*PER1*20/3)+(1.1218*min(1,(2*PER1*20/3))*
PER1*(1+32*PER1*PER1))))))/0.5;

    n1=n2;
end

n1=0;
n2=0;
c=0;
s=1;
ss11=zeros(100);
ss_11=zeros(100);

for c=1:10
        n2=n1+0.5;
forjj= s:(length (tx11(:,4)))
if tx11(jj,4)~= 0 && ((tx11(jj,2)>=n1) && (tx11(jj,2)<n2))

s11(jj,1)=th_11(c,1);
s11(jj,2)=tx11(jj,2);
      s=s+1;
end
end
   n1=n2;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n1=0;
n2=0;
c=0;
rr1=0;
m1=0;
th_1=0;
s=1;
while n2<=5
   n=0;
   m=0;
   x=0;
   n2=n1+0.5;
forjj=s:(length (tx1(:,4)))
if tx1(jj,4)~= 0 && tx1(jj,2)<n2

        n=n+tx1(jj,4);
        m=m+tx1(jj,7);
        x=x+1;
        s=s+1;
end
end
   c=c+1;
    rr1(c)=n/x;
    m1(c)=m/x;

    th_1(c,1) =
(256*8)/(rr1(c)*real((sqrt((2*PER1*20/3)+(1.1238*min(1,(2*PER1*20/3))*P
ER1*(1+32*PER1*PER1))))))/0.5;

    n1=n2;
end


n1=0;
n2=0;
c=0;
s=1;
ss1=zeros (100);
ss_1=zeros (100);

for c=1:10
        n2=n1+0.5;
forjj= s:(length (tx1(:,4)))
if tx1(jj,4)~= 0 && ((tx1(jj,2)>=n1) && (tx1(jj,2)<n2))

s1(jj,1)=th_1(c,1);
s1(jj,2)=tx1(jj,2);
      s=s+1;
end
end
   n1=n2;
end
```

```
figure ;
plot (s11 (:,2),s11(:,1), 'bo-' , 'Linewidth' , 1 ,
'MarkerEdgeColor','r' , 'MarkerFaceColor','k', 'Markersize' , 4);
holdon

plot (s1(:,2),s1(:,1), 'ks--' , 'Linewidth' , 1 , 'MarkerEdgeColor','r'
, 'MarkerFaceColor','w', 'Markersize' , 4);
holdoff


title (' Throughput Average in the first Hop');
ylabel (' ThrAvg in bit/sec')
xlabel ('Time in sec')
legend ('Cross Layer Design','Standard TCP')
gridon

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n1=0;
n2=0;
c=0;
s=1;
rr22= 0;
th_22= 0;

while n2<=5
   n=0;
   x=0;
   n2=n1+0.5;

forjj=s:(length (tx22(:,4)))
if tx22(jj,4)~= 0 && tx22(jj,2)<n2

        n=n+tx22(jj,4);
        x=x+1;
        s=s+1;
end
end

   c=c+1;
    rr22(c)=n/x;

     th_22(c,1) =
(256*8)/(rr22(c)*real((sqrt((2*PER2*20/3)+(1.1238*min(1,(2*PER2*20/3))*
PER2*(1+32*PER2*PER2))))))/0.5;

   n1=n2;
end

n1=0;
n2=0;
```

```
c=0;
s=1;
ss22=zeros(100);
ss_22=zeros(100);

for c=1:10
        n2=n1+0.5;

forjj= s:(length (tx22(:,4)))
if tx22(jj,4)~= 0 && ((tx22(jj,2)>=n1) && (tx22(jj,2)<n2))

s22(jj,1)=th_22(c,1);
s22(jj,2)=tx22(jj,2);
      s=s+1;
end
end

   n1=n2;
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n1=0;
n2=0;
c=0;
rr2=0;
th_2=0;
s=1;

while n2<=5
   n=0;
   x=0;
   n2=n1+0.5;

forjj=s:(length (tx2(:,4)))
if tx2(jj,4)~= 0 && tx2(jj,2)<n2

        n=n+tx2(jj,4);
        n=n+tx2(jj,4)+0.0001;
        x=x+1;
        s=s+1;
end
end

   c=c+1;
    rr2(c)=n/x;

     th_2(c,1) =
(256*8)/(rr2(c)*real((sqrt((2*PER2*20/3)+(1.1238*min(1,(2*PER2*20/3))*P
ER2*(1+32*PER2*PER2))))))/0.5;

   n1=n2;
end
```

```
n1=0;
n2=0;
c=0;
s=1;
ss2=zeros (100);
ss_2=zeros (100);

for c=1:10
        n2=n1+0.5;

forjj= s:(length (tx2(:,4)))
if tx2(jj,4)~= 0 && ((tx2(jj,2)>=n1) && (tx2(jj,2)<n2))

s2(jj,1)=th_2(c,1);
s2(jj,2)=tx2(jj,2);
      s=s+1;
end
end

   n1=n2;
end



figure;
plot (s22 (:,2),s22(:,1),'bo-','Linewidth', 1
,'MarkerEdgeColor','r','MarkerFaceColor','k','Markersize', 4);
holdon

plot (s2(:,2),s2(:,1),'ks--','Linewidth', 1
,'MarkerEdgeColor','r','MarkerFaceColor','w','Markersize', 4);
holdoff

title ('Throughput Average in the Second Hop');
ylabel ('ThrAvg in bit/sec')
xlabel ('Time in sec')
legend ('Cross Layer Design','Standard TCP')


gridon

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A3 Number of Timeouts in the First and the Second hops

```
h1;
a=0.125;
m=4;
B=0.25;
Ift_select==1
tx11(h1,5)=(1-a)*(temp5_1)+a*temp4_1;           %%S(K)
tx11(h1,6)=(1-B)*(temp6_1)+B*(abs(temp4_1-tx11(h1,5)));      %%V(k)
tx11(h1,7)=tx11(h1,5)+m*tx11(h1,6)+0.008;       %%RTO
tx11(h1,8)=tx11(h1,2)+tx11(h1,7);

if bb<=3
tx11(h1,9)=tx11(h1,2)+0;
end

temp5_1=tx11(h1,5);
   temp6_1=tx11(h1,6);
end

ift_select==2
tx1(h1,5)=(1-a)*(temp5_1)+a*temp4_1;
tx1(h1,6)=(1-B)*(temp6_1)+B*(abs(temp4_1-tx1(h1,5)));
tx1(h1,7)=tx1(h1,5)+m*tx1(h1,6)+0.01;
tx1(h1,8)=tx1(h1,2)+tx1(h1,7);

if bb<=3 & h1>1
tx1(h1,9)=tx1(h1,2);
end

if h1>1 & tx1(h1,4)<tx1(h1-1,4)
    temp5_1=tx1(h1,5);
    temp6_1=tx1(h1,6);
end

if h1==1 | tx1(h1,4)>=tx1(h1-1,4)
    temp5_1=tx1(h1,5);
    temp6_1=tx1(h1,6);
end

end
timeout(i,1)= timeout(i,1)+1;
x1=[1 2];
figure;
bar(x1,[timeout([1 2],1),timeout([1 2],2)],'grouped','LineWidth',1)
barmap=[0.05 0.45 0.1; 0.7 0.7 0.7];
colormap(barmap);
title (' Number of Timeouts in the first and second Hop');
ylabel (' No. of Timeouts');
xlabel ('Hop No.');
legend ('Cross Layer Design','Standard TCP');
gridon;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A4 Efficiency in the First and Second Hops

```
throughput_time(th1,i)=throughput_time(th1,i)+ 1*1024*8;

forjj=1:5
theff(jj,1)=throughput_time(jj,2)/(20*10^6);
theff(jj,2)=throughput_time(jj,3)/(20*10^6);
theff(jj,3)=throughput_time(jj,5)/(20*10^6);
theff(jj,4)=throughput_time m(jj,6)/(20*10^6);
end

figure;
x2=[1 2 3 4 5];
bar(x2,[theff(:,1),theff(:,3)],'grouped','LineWidth',1)
title (' Spectral Efficiency in the First Hop');
ylabel (' Efficiency in Bits/Sec/Hz');
xlabel ('Time in Seconds');
legend ('Cross Layer Design','Standard TCP');
gridon;

figure;
bar(x2,[theff(:,2),theff(:,4)],'grouped','LineWidth',1)
title (' Spectral Efficiency in the Second Hop');
ylabel (' Efficiency in Bits/Sec/Hz');
xlabel ('Time in Seconds');
legend ('Cross Layer Design','Standard TCP');
gridon;
```