A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC

TEERAPHOL SUKPRAPAIPAT

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2014**

Thesis
entitled

A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC

……………………………………………
Mr. Teeraphol Sukprapaipat
Candidate

……………………………………………
Lect. Noppadol Wanichworanant,
Ph.D. (Electrical Engineering)
Major advisor

……………………………………………
Asst. Prof. Suratose Tritilanunt,
Ph.D. (Information Technology)
Co-advisor

……………………………………………
Asst. Prof. Konglit Hunchangsith,
Ph.D. (Information and Communication Technology management)
Co-advisor

……………………………………………
Asst. Prof. Auemphorn Mutchimwong,
Ph.D. (Air Quality Assessment)
Acting Dean
Faculty of Graduate Studies
Mahidol University

…………………………………………...
Lect. Supaporn Kiattisin,
Ph.D. (Electrical and Computer Engineering)
Program Director
Master of Science Program in Technology of Information System Management
Faculty of Engineering
Mahidol University

Thesis
entitled

A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science
(Technology of Information System Management)
on
May 19, 2014

……………….….…..……………………
Mr. Teeraphol Sukprapaipat
Candidate

……………….….…..……………………
Lect. Sotarat Thammaboosadee,
Ph.D. (Information Technology)
Chair

……………………….….…..………
Lect. Noppadol Wanichworanant,
Ph.D. (Electrical Engineering)
Member

….…………..….…..……………………...…
Asst. Prof. Suratose Tritilanunt,
Ph.D. (Information Technology)
Member

……………………….….…..………
Asst. Prof. Konglit Hunchangsith,
Ph.D. (Information and Communication Technology management)
Member

……………….….…..……………………
Asst. Prof. Kairoek Choeychuen,
Ph.D. (Electrical and Computer Engineering)
Member

……………………….….…..………
Asst. Prof. Auemphorn Mutchimwong,
Ph.D. (Air Quality Assessment)
Acting Dean
Faculty of Graduate Studies
Mahidol University

……………………….….…..……………
Lect. Worawit Israngkul,
M.S. (Technical Management)
Dean
Faculty of Engineering
Mahidol University

ACKNOWLEDGEMENTS

The success of this thesis is attributed by attentive support from my major advisor, Lect. Noppadol Wanichworanant, my co-advisor, Asst. Prof. Suratose Tritilanunt and Asst. Prof. Konglit Hunchangsith. I would like to express my sincere gratitude for all their invaluable encouragement, guidance, supervision and suggestion throughout this research.

A special thank goes to my co-advisor, Lect. Sotarat Thammaboosadee, committee chair and Asst. Prof. Kairoek Choeychuen, committee for their comments and useful suggestions to this research.

Special acknowledgement goes to all the lecturers and staff of the Technology of Information System Management Program, Faculty of Engineering, Mahidol University for their service and support. I would like to thank all of my friends for their helps and encouragements.

Finally, the most important and indispensable persons are my family members. I am very grateful for their entire support, caring and love throughout my whole life. This has inspired me to fight and beat the obstacles to finish this research as well.

Teeraphol Sukprapaipat

A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC

TEERAPHOL SUKPRAPAIPAT 5236777 EGTI/M

M.Sc.(TECHNOLOGY OF INFORMATION SYSTEM MANAGEMENT)

THESIS ADVISORY COMMITTEE : NOPPADOL WANICHWORANANT, Ph.D., SURATOSE TRITILANUNT, Ph.D., KONGLIT HUNCHANGSITH, Ph.D.

ABSTRACT

In today’s modern world, the use of multimedia is important for communication in business and daily life. Many people use real-time applications such as Voice over Internet Protocol (VoIP) for communication, so they need high-quality multimedia traffic when using these applications.

This research presents a hybrid network transport protocol for real-time applications and multimedia traffic. The name of the hybrid protocol is the New User Datagram Protocol or NUDP. This protocol is based on UDP and has been improved by using the retransmission method with modification to compensate for the packet loss rate. Furthermore, a function called “Sequencer” was added for re-ordering packets before sending to the application. This function can also reduce the application’s workload.

The results of this research is the NUDP can transmit data as quick as UDP when there is no data loss occurring in network, and it also has a data loss rate that is less than UDP in every situation. For this reason, NUDP is a suitable protocol for real-time applications and multimedia traffic because both real-time applications and multimedia traffic concern delay and data loss rate.

KEY WORDS: NETWORK PROTOCOL / MULTIMEDIA TRAFFIC / NS-2 / UDP MODIFICATION / TRANSPORT LAYER PROTOCOL

156 pages

โพรโทคอลเครือข่ายแบบผสมสำหรับการส่งข้อมูลแบบมัลติมีเดีย

A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC

ธีระพล สุขประไพพัฒน์ 5236777 EGTI/M

วท.ม. (เทคโนโลยีการจัดการระบบสารสนเทศ)

คณะกรรมการที่ปรึกษาวิทยานิพนธ์: นพดล วณิชวรนันท์, Ph.D., สุรทศ ไตรติลานันท์, Ph.D., ผ.ศ.คงฤทธิ์ หันจางสิทธิ์, Ph.D.

บทคัดย่อ

ในปัจจุบัน การส่งข้อมูลที่อยู่ในรูปแบบของมัลติมีเดียนั้นมีความจำเป็นทั้งในด้านของธุรกิจ และในด้านของการใช้ชีวิตประจำวัน แต่เนื่องจากในปัจจุบันนั้น การส่งข้อมูลที่อยู่ในรูปแบบของมัลติมีเดียจะใช้โพรโทคอล UDP ในการรับส่งข้อมูล ซึ่งลักษณะการทำงานของโพรโทคอล UDP นั้นจะมุ่งเน้นไปในเรื่องของความรวดเร็วในการรับส่งข้อมูล แต่จะมีอัตราการสูญหายของข้อมูลที่สูง และไม่มีกระบวนการใดๆ ที่จะใช้ในการลดอัตราการสูญหายของข้อมูลที่เกิดขึ้นระหว่างการรับส่งข้อมูล ซึ่งการสูญหายของข้อมูลนั้นจะส่งผลให้คุณภาพของภาพ, เสียง, และภาพเคลื่อนไหวนั้นลดลง ด้วยเหตุผลนี้เอง จึงเป็นที่มาของงานวิจัยชิ้นนี้ที่ได้ทำการศึกษาและพัฒนาโพรโทคอลที่เหมาะสมสำหรับใช้ในการรับส่งข้อมูลในรูปแบบของมัลติมีเดีย โดยในงานวิจัยชิ้นนี้ได้นำเอาข้อดีของโพรโทคอล TCP และ UDP มาใช้ร่วมกัน ซึ่งในงานวิจัยชิ้นนี้เราได้ในเอาวิธีการส่งข้อมูลใหม่ของโพรโทคอล TCP มาปรับปรุงใช้กับ UDP เพื่อลดอัตราการสูญหายของข้อมูลที่เกิดขึ้นระหว่างทำการส่งข้อมูล ผลลัพธ์ที่ได้คือ โพรโทคอลที่ถูกพัฒนาขึ้นมานั้นสามารถรับส่งข้อมูลได้รวดเร็วเทียบเท่ากับโพรโทคอล UDP ในสถานการณ์ที่ไม่มีการสูญหายของข้อมูลเกิดขึ้น และโพรโทคอลนี้มีอัตราการสูญหายของข้อมูลดีกว่าโพรโทคอล UDP ซึ่งเป็นโพรโทคอลที่ใช้ในการรับส่งข้อมูลในรูปแบบของมัลติมีเดียในปัจจุบันในทุกๆ สถานการณ์ ด้วยเหตุผลดังกล่าวจึงทำให้โพรโทคอลในงานวิจัยชิ้นนี้เหมาะสมสำหรับใช้ในการรับส่งข้อมูลในรูปแบบของมัลติมีเดีย เพราะการรับส่งข้อมูลในรูปแบบของมัลติมีเดียนั้นจะมุ่งเน้นในเรื่องของความรวดเร็วในการรับส่งข้อมูลและเรื่องอัตราการสูญหายของข้อมูล

156 หน้า

CONTENTS

CONTENTS (cont.)

CONTENTS (cont.)

LIST OF TABLE

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF TABLE (cont.)

LIST OF FIGURE

LIST OF FIGURE (cont.)

LIST OF FIGURE (cont.)

CHAPTER I
INTRODUCTION

1.1 Background and statement of problem

In today's modern world, the use of multimedia is important for communications in both business and personal life. Many people use real-time applications such as Voice over Internet Protocol(VoIP) for communication. For example, Skype has more than 80 million subscribers [5]. Most subscribers require high quality multimedia traffic when using these applications. The quality of multimedia traffic from these applications generally depends on key factors such as latency, jitter and data loss [15]. Some factors such as data loss can be managed by protocol in the transport layer of the Open Systems Interconnection (OSI) model.

The transport layer has two favorite protocols namely, Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). The aforementioned protocols are important part for transmitting data and both protocols have different advantages for use in different situations. TCP is used by internet applications requiring guaranteed delivery such as File Transfer Protocol (FTP) and the TELNET. UDP is used by other internet applications that do not require guaranteed delivery such as the Network File System (NFS), Simple Network Management Protocol (SNMP) [16] and real-time applications.

TCP is one of the main protocols in transmission control protocol/Internet Protocol (TCP/IP) that is the suite of communication protocols used to connect a computer to the internet. It is a reliable protocol which means this protocol guarantees the destination will receive all data. TCP has a mechanism called "Retransmission" for retransmitting packets when packet loss occurs. Other mechanisms, such as windowing for congestion control, are also available. These mechanisms improve transmitting performance. However, these mechanisms have overhead that increases the delay making transmission slower. Therefore, TCP is not suitable for use in multimedia traffic and real-time applications because multimedia and real-time applications are concerned about delay and quality can also be reduced when TCP is used to transmit multimedia and real-time traffic.

On the other hand, UDP is an unreliable protocol because there is no guarantee of successfully receiving the packets and UDP has no mechanism for managing packet loss. Furthermore, UDP does not wait for an acknowledgment before sending data. For this reason, it can send data faster than TCP. Nevertheless, data can be dropped, arrive out of order or be replicated. For these reasons, UDP is more suitable for applications concerned with delay than TCP. At present, UDP is used for transmitting multimedia and real-time traffic.

Although UDP's low delay makes the protocol suitable for real-time applications and multimedia traffic, significant problems are encountered due to the nature of UDP's packet loss resulting in lower quality for real-time applications. UDP offers no method for reducing or compensating for data loss. If many packets are dropped during a transmission and there is no method to compensate for packet loss, the performance will degrade.

Both TCP and UDP have different advantages. TCP has many features for making reliable transmissions such as retransmission. UDP can quickly transmit data. A combination of both TCP and UDP advantages is found in the concept of hybrid protocol.

This research presents a hybrid network transport protocol for use in real-time applications and multimedia traffic. The name of a hybrid protocol is the New User Datagram Protocol or NUDP. This protocol is based on UDP and has been improved by using the retransmission method with modification to compensate for the packet loss rate. Furthermore, a function called "Sequencer" was added for re-ordering packets before sending to the application. This function can also reduce the application's workload.

1.2 Objective of Study

TCP is a reliable protocol for transmitting data and UDP is a protocol capable of transmitting data quickly. The advantages of both TCP and UDP have been applied to the formation of a new protocol that is suitable for real-time applications and multimedia traffic. Accordingly, the new protocol must have a data loss rate less than UDP because UDP is the protocol currently used for real-time applications and multimedia traffic. Therefore the objectives of this research are as follows:

1. To design a process for reducing the data loss rate.
2. To develop a protocol with a lower data loss rate than UDP in every situation.
3. To develop a protocol capable of re-ordering the data before sending to the application.

1.3 Scope of Work

1. Improving User Datagram Protocol (UDP).
2. Improving UDP performance by reducing data loss.
3. Simulating with a Network Simulator 2 (NS-2).

1.4 Expected Results

1. A process for reducing the data loss rate.
2. A protocol with a lower data loss rate than UDP in every situation.
3. A protocol capable of re-ordering data before sending to the application.

CHAPTER II
LITERATURE REVIEW

2.1 Computer network meaning

A definition of computer network usually is a collection of computers interconnected for gathering, processing, and distributing information. These computers include devices such as servers, router, modems, workstations, notebook, etc. These computers are connected by medium or link such as fiber optic cables, copper cables, microwave, radio, and satellite links. The good example of the computer networks is the internet [12].

When we knew about meaning of computer network, we studied about OSI and TCP/IP model for understanding about how communication occurs across the network.

2.2 OSI model and TCP/IP model

We studied OSI and TCP/IP for understanding how communication occurs across the network. We use layer models as a framework for showing and describing networking concepts and technologies. Both models have many benefits such as reduce complexity, standardize interfaces, assist understanding, promote rapid product development, support interoperability, and facilitate modular engineering.

2.2.1 OSI model

The first reference model was developed by International Standards Organization (ISO) was the Open Systems Interconnection (OSI) model to provide a standard framework to describe the protocol stacks in a computer network. The OSI model just tells what each layer should do. It does not show the exact services and protocols to be used in each layer. Though the OSI model does not implemented in current, the OSI model philosophy is a basic for development in computer networking [12].

There are seven layers in OSI model where each layer is intended to perform a well-defined function including physical layer, data link layer, network layer, transport layer, session layer, presentation layer, and application layer. Figure 2.1 presented the OSI model and table 2.1 presented functions of the OSI model.

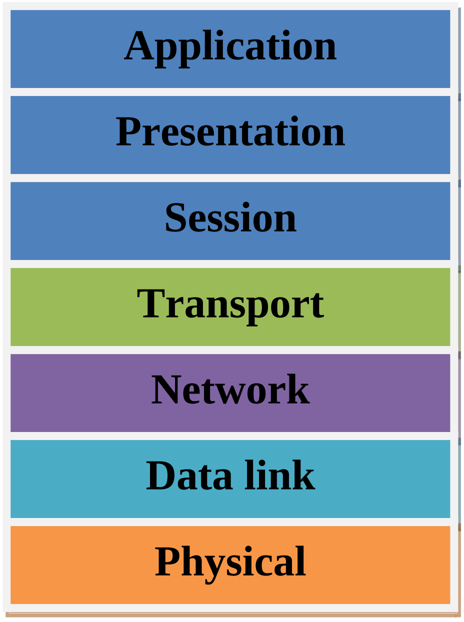

Figure 2.1 OSI model

2.2.2 TCP/IP model

The TCP (Transmission Control Protocol)/IP (Internet Protocol) model that is based on the two primary protocols which is TCP and IP that is used in current internet. It was developed for ARPANET which is a research networks sponsored by the U.S. Department of Defense which is considered as the grandparent of all computer networks [12]. The TCP/IP model defines four groups of functions that must occur for communication to be successful. Most protocol models describe a vendor-specific protocol stack. However, it does not control the definition of the model because TCP/IP model is an open standard [1]. Figure 2.2 presented the TCP/IP model compared with OSI model and table 2.2 presented functions and example protocols of the TCP/IP model.

Table 2.1 Functions of the OSI model

Layer	Function Description
Application (7)	Refers to interfaces between network and application software. Also includes authentication services.
Presentation (6)	Defines the format and organization of data. Includes encryption.
Session (5)	Establishes and maintains end-to-end bidirectional flows between endpoint. Includes managing transaction flows.
Transport (4)	Provides a variety of services between two host computers including connection establishment and termination, flow control, error recovery, and segmentation of large data blocks into smaller parts for transmission.
Network (3)	Refers to logical addressing, routing, and path determination.
Data link (2)	Formats data into frames appropriate for transmission onto some physical medium. Defines rules for when the medium can be used. Defines means by which to recognize transmission errors.
Physical (1)	Defines the electrical, optical, cabling, connectors, and procedural details required for transmitting bits, represented as some form of energy passing over a physical medium.

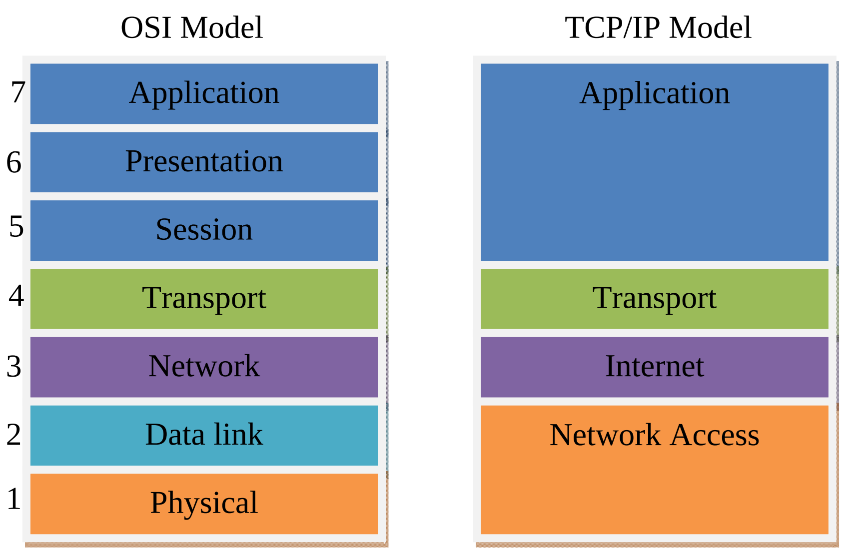

Figure 2.2 OSI model and TCP/IP model

Table 2.2 Functions and example protocols of TCP/IP model

TCP/IP Layer	Function	Example Protocols
Application	Represents data to the user and controls dialog.	DNS, Telnet, SMTP, POP3, IMAP, DHCP, HTTP, FTP, SNMP
Transport	Supports communication between diverse devices across diverse networks.	TCP, UDP
Internet	Determines the best path through the network.	IP, ARP, ICMP
Network access	Controls the hardware devices and media that make up the network	Ethernet, Frame Relay

In this research, we focused on Transport layer which has two favorite protocols that is TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). Both protocols have important for network communication. We presented details, advantage and disadvantage of both protocols.

2.3 Transmission Control Protocol

Transmission Control Protocol (TCP) takes large blocks of information from an application and divides them into segments. It numbers and sequences each segment so that the destination's TCP stack can put the segments back into the order the application intended. After these segments are sent, TCP from source host waits for an acknowledgment of the receiving end's TCP virtual circuit session, retransmitting those that are not acknowledged.

Before a transmitting host starts to send segments down the model, the sender's TCP stack contacts the destination's TCP stack to establish a connection. What is created is known as a virtual circuit. This type of communication is called connection-oriented. During this initial handshake, the two TCP layers also agree on the amount of information that is going to be sent before the recipient's TCP sends back an acknowledgment. With everything agreed upon in advance, the path is paved for reliable communication to take place.

TCP is a full-duplex, connection-oriented, reliable, and accurate protocol, but establishing all these terms and conditions, in addition to error checking, is no small task. TCP is very complicated and, not surprisingly, costly in terms of network overhead. And since today's networks are much more reliable than those of yore, this added reliability if often unnecessary [13]. In figure 2.3 presented format of TCP header.

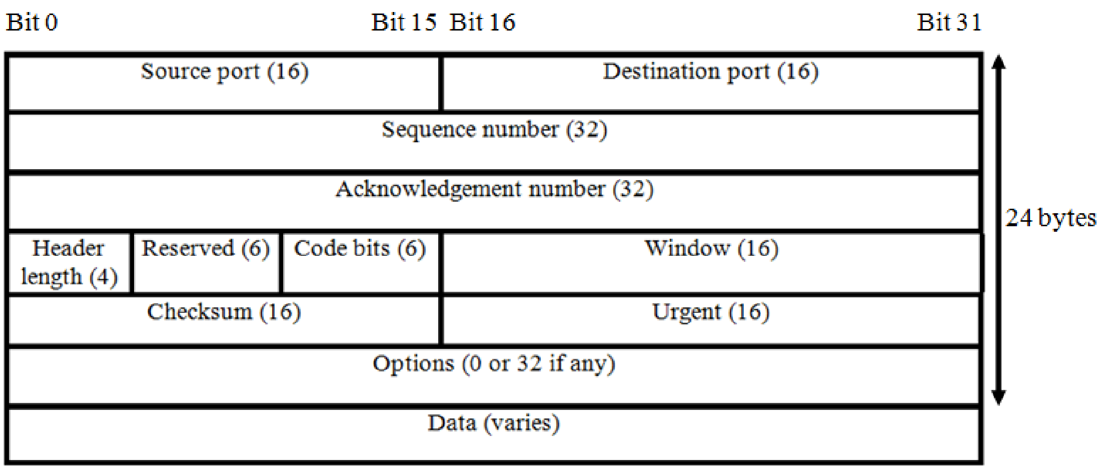

Figure 2.3 TCP header format

The TCP header is 20 bytes long and it can up to 24 bytes with options. The details of each field in the TCP header as follow: [13]

- Source port. The port number of the application on the host sending the data.
- Destination port. The port number of the application requested on the destination host.
- Sequence number. A number used by TCP that puts the data back in the correct order or retransmits missing or damaged data, a process called sequencing.
- Acknowledgment number. The TCP octet that is expected next.
- Header length. The number of 32-bit words in the TCP header. This indicates where the data begins. The TCP header is an integral number of 32 bits in length.

- Reserved. Always set to zero.
- Code bits. Control functions used to set up and terminate a session.
- Window. The window size the sender is willing to accept, in octets.
- Checksum. The cyclic redundancy check (CRC), because TCP does not trust the lower layers and checks everything. The CRC checks the header and data fields.
- Urgent. A valid field only if the Urgent pointer in the code bits is set. If so, this value indicates the offset from the current sequence number, in octets, where the first segment of non-urgent data begins.
- Options. May be zero or a multiple of 32 bits, if any. What this means is that no options have to be present (options size of zero). However, if any options are used that do not cause the option field to total a multiple of 32 bits, padding of 0s must be used to make sure the data begins on a 32-bit boundary.
- Data. Handed down to the TCP protocol at the Transport layer which includes the upper layer headers.

2.4 User Datagram Protocol

User Datagram Protocol (UDP) does not sequence the segments and does not care in which order the segments arrive at the destination. But after that, UDP sends the segments off and forgets about them. It does not follow through, check up on them, or even allow for an acknowledgment of safe arrival. Because of this, it is referred to as an unreliable protocol. This does not mean that UDP is ineffective, only that it does not handle issues of reliability.

UDP does not create a virtual circuit, nor does it contact the destination before delivering information to it. Because of this, it is also considered a connectionless protocol. Since UDP assumes that the application will use its own reliability method, it does not use any. This gives an application developer a choice when running the Internet Protocol stack: TCP for reliability or UDP for faster transfers [13]. UDP has low overhead when compared with TCP as we presented in figure 2.4, it presented format of UDP header.

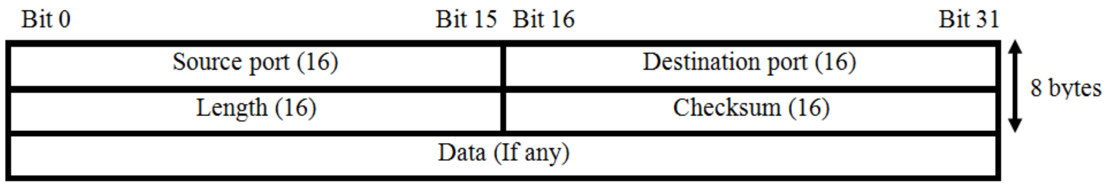

Figure 2.4 UDP header format

The details of each fields of UDP header as follow:

- Source port. Port number of the application on the host sending the data.
- Destination port. Port number of the application requested on the destination host.
- Length. Length of UDP header and UDP data.
- Checksum. Checksum of both the UDP header and UDP data fields.
- Data. Upper-layer data.

When we compared protocols in transport layer between TCP and UDP, we can saw a different thing from both protocols that is the key for reducing packet loss. It is retransmission which is a key function of TCP. A positive ACK is expected from the receiving TCP for the sender's successful data transfer. If the ACK is not received within a timeout period, the data are retransmitted. Retransmission can compensate packet loss or corrupted packet. TCP has another work that is the responsibility to ensure the correct order of received packets. Nevertheless, TCP has overhead for guarantee reliable and accurate data transfer that is reduces the network's throughput. Moreover, TCP increases the average transmission delay and the worse jitter [8].

We knew about details of both protocols in transport layer, we studied about application for using these protocols. Akyildiz and McNair [6] said multimedia traffic and real-time application such as digital voice and video normally using Constant bit rate (CBR) for generating traffic.

2.5 Constant Bit Rate

The Constant Bit Rate (CBR) is used for connections that transport traffic at a consistent bit rate, where there is an inherent reliance on time synchronization between the traffic sender and receiver. CBR suitable for any type of data for which the end-systems require predictable response time and a static amount of bandwidth continuously. The suitable applications for CBR include video conferencing, telephony, and any type of on-demand service such as interactive voice and audio [9].

Normally when we transmit multimedia data or use real-time application between networks, Quality of Service (QoS) is a main factor which we should concern it.

2.6 Quality of Service

Quality of Service (QoS) is a main issue on transmitting multimedia data that is how to guarantee that packet traffic for a voice or other media connection will not be delayed of dropped due interference from other lower priority traffic [17]. There are three things to consider including latency, jitter, and packet loss.

Now we knew about factor in QoS that is packet loss, latency (packet delay) and jitter (packet delay variation). Quality of voice and video on network depend on these factors [3]. If values of these factors increase, quality of voice and video on network will decrease. The detail of these factors as follow:

Latency also called packet delay is the time between a packet being transmitted from the source and received by the destination [10]. Large latency or large delays are bad thing and it can because voice echoes that it is hard to have a conversation with too large delays.

Jitter also called packet delay variation means the difference in end-to-end delay between selected packets transferred from the source to the destination [10]. Jitter can cause abnormal sound when have to conversation. The example is if it takes 40 ms (milliseconds) for the first voice packet to arrive and 90 ms for the second packet to arrive, the jitter level is 50ms.

Finally packet loss is the number of packets sent from the source minus the number of packets received from the destination [10]. Packet loss may occur at any stage of a network transmission. It usually occur when link or medium is failure, when network congestion is high, router mechanism called random early detection (RED), and Ethernet problems. Packet loss is the main problem for reducing quality of voice and video [4]. Packet loss can cause interrupts and make sound lousy.

Normally, real-time application used UDP for transfer and deliver voice data to destination because UDP has low delay but UDP has disadvantage that it has high packet loss rate because UDP does not have method for compensating packet loss. The real-time applications such as VoIP can only allow a small number of packet losses. The performance of the system will be increase if the packet loss rate can be reduced. Many researchers tried to find techniques for compensating packet loss.

Bova and Krivoruchk [11] presented the Reliable UDP (RUDP) which modified from UDP. They add functions that make packet delivery reliable and in-order such as acknowledgment when received data, windowing and congestion control, retransmission of lost packets. The RUDP retransmission algorithm is the sender enables retransmission at the time-out of the retransmission timer of when it receives and extended acknowledgment (EACK) segment. When EACK segment is received, the sender checks this segment and determines which segments should be retransmitted. An EACK segment contains the ACK number and the last out-of-sequence ACK number. Every segment between these two sequences numbers that are on the unacknowledged sent queue to be retransmitted. In the condition of a retransmission time-out, all messages on the unacknowledged sent queue are retransmitted. The time-out value is configurable in a retransmission timer. The timer is initialized every time a segment is sent and will reset when receiving an acknowledgement. 600 milliseconds is the recommended time-out value. Working together with the retransmission timer is the retransmission counter. This counter maintains the number of times a segment has been retransmitted. There is a configurable maximum value for the counter. Once the counter more than its maximum, the virtual connection is considered broken and a connection auto reset will be performed.

RUDP uses a SYN segment to establish the connection and synchronize sequence numbers between two hosts. Both the value of the retransmission timer and the maximum number of the retransmission counter are specified in the SYN segment.

These two values are defined before the build of the connection and they are constant over the connection's lifetime. For this reasons the RUDP does not take into account changes in the network's condition. When the network's condition turns bad, inappropriate retransmission will burden the link and cause a worse network condition, and lead to degradation of the overall performance of communication.

Le et al [14] modified RUDP by adding function that is the dynamic retransmission control. The name of this protocol is The Enhanced RUDP (E-RUDP). E-RUDP enables dynamic retransmission by using the data packet aging parameter, round-trip times (RTT) and the maximum cumulative ACK count. Data packet aging specifies the maximum delay the application can tolerate.

Gui [7] presented protocol which modify from UDP for using in wireless network control system. This protocol is named the Conditional Retransmission Enabled Transport Protocol (CRETP). CRETP has a conditional retransmission mechanism which improves data transmission reliability. Packet lost will be re-transferred for certain of times to compensate for data losses. It has function for checking data effectiveness and guarantee that every packet delivered to the application.

Thammadi [2] presented the protocol for using in reliable communication. This protocol modified from UDP. This protocol used positive acknowledgement and negative acknowledgement for provide reliability. Furthermore, this protocol uses the sliding window (selective repeat) for transmitting data.

Every protocol that described above used retransmission method for guarantee reliable transmits data. They must receive acknowledgement before sending next data. For this reason, theses protocols have high delay that is not suitable for real-time application and multimedia traffic because they concern about delay.

Therefore, we designed and developed protocol that suitable for real-time applications and multimedia traffic that we will describe in next chapter.

CHAPTER III
RESEARCH METHODOLOGY

This chapter presents the processes for developing a new protocol called the New User Datagram Protocol (NUDP) for the purpose of reducing the packet loss rate. This protocol is based on UDP and has been improved by adding a retransmission method to compensate for the packet loss rate. The research was conducted in four phases, including the Studying Phase, the Design and Coding Phase, the Evaluation Phase and the Conclusion Phase. Figure 3.1 presents the steps of this research.

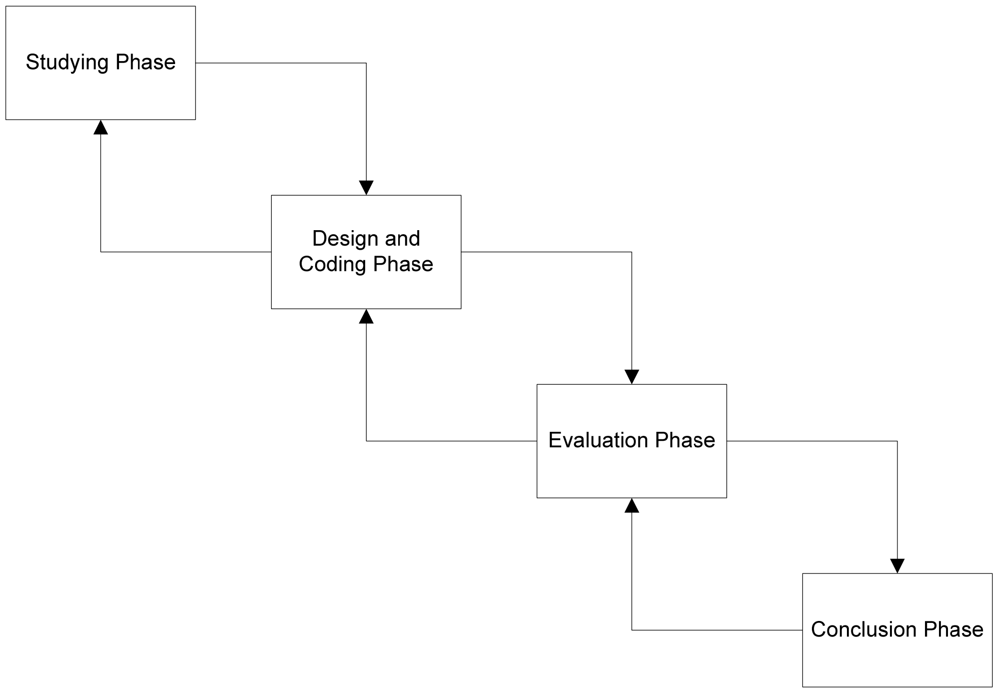

Figure 3.1 Research methodology

3.1 Studying phase

The studying phase is a very important phase in the research because the researchers need to study everything in relation to the research topic in order to accurately design and develop NUDP.

In this phase, the network system was studied according to the OSI model concerning the function of each layer with focus on the transport layer. Two popular protocols in the transport layer are TCP and UDP. The focus of the present study was centered on UDP. The researchers studied the characteristics, structure, advantages and disadvantages of UDP.

The researchers then studied the network simulator because protocols are difficult and expensive to develop or improve in the real world. Therefore, a simulator is good option for this situation. For the above reasons, the researchers decided to use a simulator to develop the protocol for the current study.

Two simulators are well-known among network researcher, namely, Network Simulator 2 (NS2) and OPNet. Both NS2 and OPNeT are perfect for the development and simulation of NUDP. The researchers decided to choose NS2 for developing NUDP in this research because NS2 is an open source and non-commercial program, while OPNeT is a commercial program.

3.2 Design and Coding phases

Once the researchers were familiar with the details concerning the transport layer of the OSI model and to the method for using NS2, which is the network simulator used in the current study. This phase required the application of knowledge from the study phase to design and develop NDUP into the NS2.

In the design and coding phases, the researchers designed the mechanisms for NUDP, the NUDP packet header, mechanisms for data effectiveness detection, acknowledgement mechanisms and conditional retransmission mechanisms, etc. Next, the researchers implemented NUDP in NS-2. The researchers were then able to go return to the study phase when the researchers had problems concerned with the simulator or network theory.

3.3 Evaluation phase

In the evaluation phase, the researchers evaluated NUDP by implementing NUDP in a case study. Two cases were investigated. In Case One, the situation was marked by numerous transitions in sending from source node to a destination node. The situation in Case Two involved one user watching a video online at a website. The results of this phase were the data loss rate and the average delay of NUDP compared with UDP in different situations. The researchers were able to go back to the design and coding phase for to design and modify certain processes or features of the protocol.

3.4 Conclusion phase

This phase concluded all of results from the simulation. The researchers summarized the results by using tables and graphs. The researchers were able to return to the Evaluation Phase when the results or other information was insufficient.

3.5 NUDP Logical Design

NUDP is a UDP-based protocol with the ability to retransmit data loss. Figure 3.2 shows the correlations of NUDP with the other protocols and layers of the TCP/IP model. Similar to UDP, NUDP is between the application layer and the IP layer.

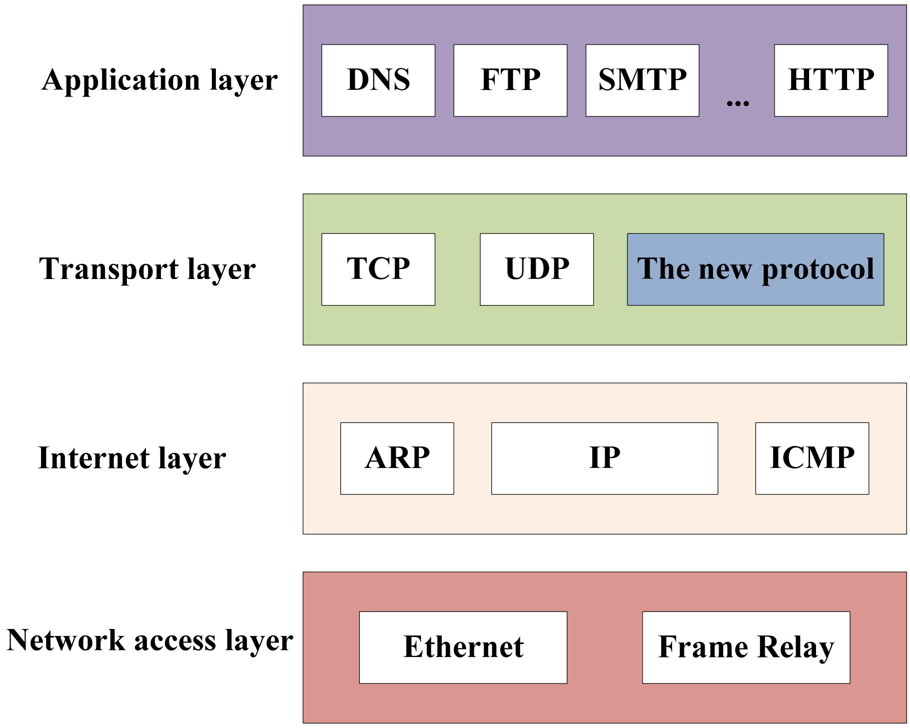

Figure 3.2 Position of NUDP in the TCP/IP model

3.5.1 NUDP philosophy

The researchers' philosophy for implementing NUDP was based on the following aspects:

- Retaining the simplicity of UDP in an unchanged state to maintain a similar level of rapid transmission.
- Using a retransmission method to compensate for packet loss.
- Check the effectiveness of the packets received to guarantee data timeliness.

3.5.2 Connectionless service with acknowledgement

NUDP was based on UDP. Hence, NUDP is a connectionless protocol and there is no need to establish a connection with three-way handshaking. Furthermore, each packet can travel on a different path. There is no flow control in NUDP. These simplicities enable the rapid data transmission by NUDP.

NUDP adds a sequence number to each data packet and enables acknowledgement for informing the source when packet losses have occurred in the network. The sequence number shows the order of a packet but the sequence number of the acknowledgement packet shows the sequence number of the lost packet. Figure 3.3 shows the successful delivery of a data packet and Figure 3.4 shows when packet loss has occurred when the receiver sent acknowledgement to inform the sender.

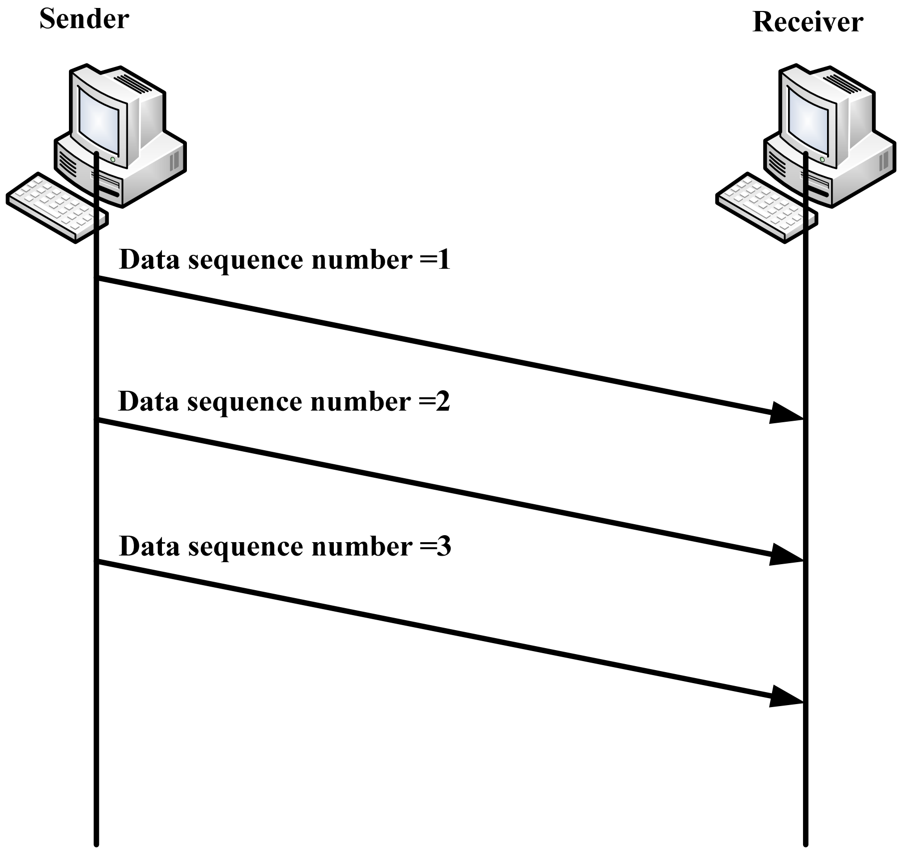

Figure3.3 Successful data packet delivery

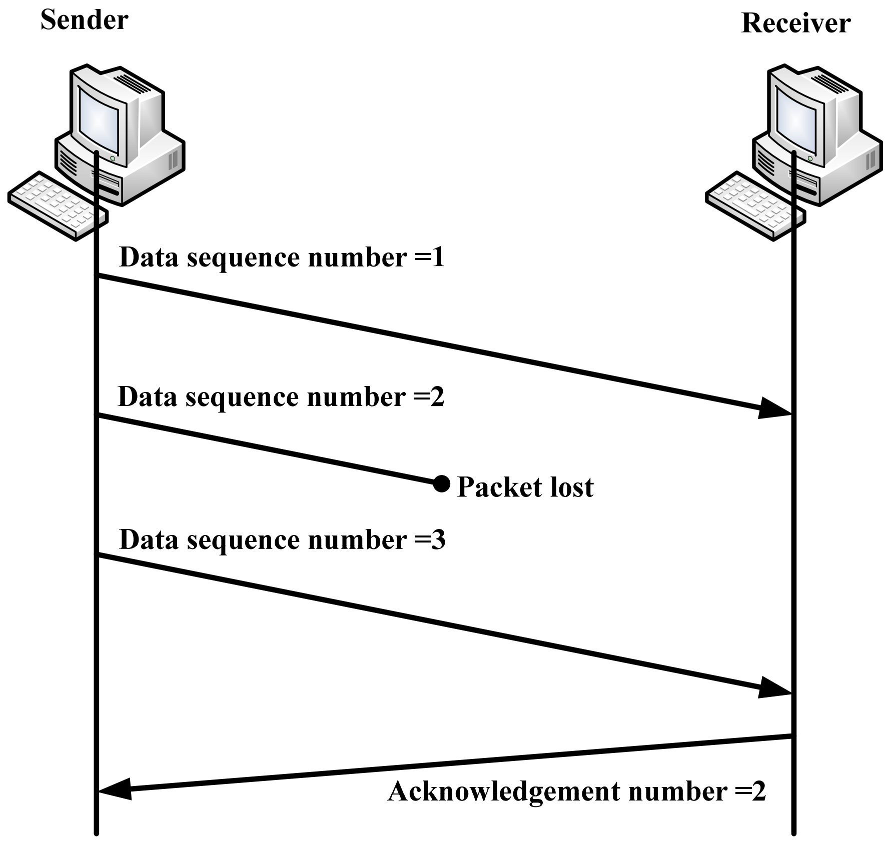

Figure 3.4 Receiver sends acknowledgement packet

3.5.3 Conditional retransmission services

Retransmission is the method used by NUDP to compensate packet loss. A packet is retransmitted when the sender has received an acknowledgement from the receiver. The acknowledgement packet contains the sequence number of the lost packet and number of packets lost. Figure 3.5 shows the data retransmission of a lost packet and Figure 3.6 shows the data retransmission of various packets.

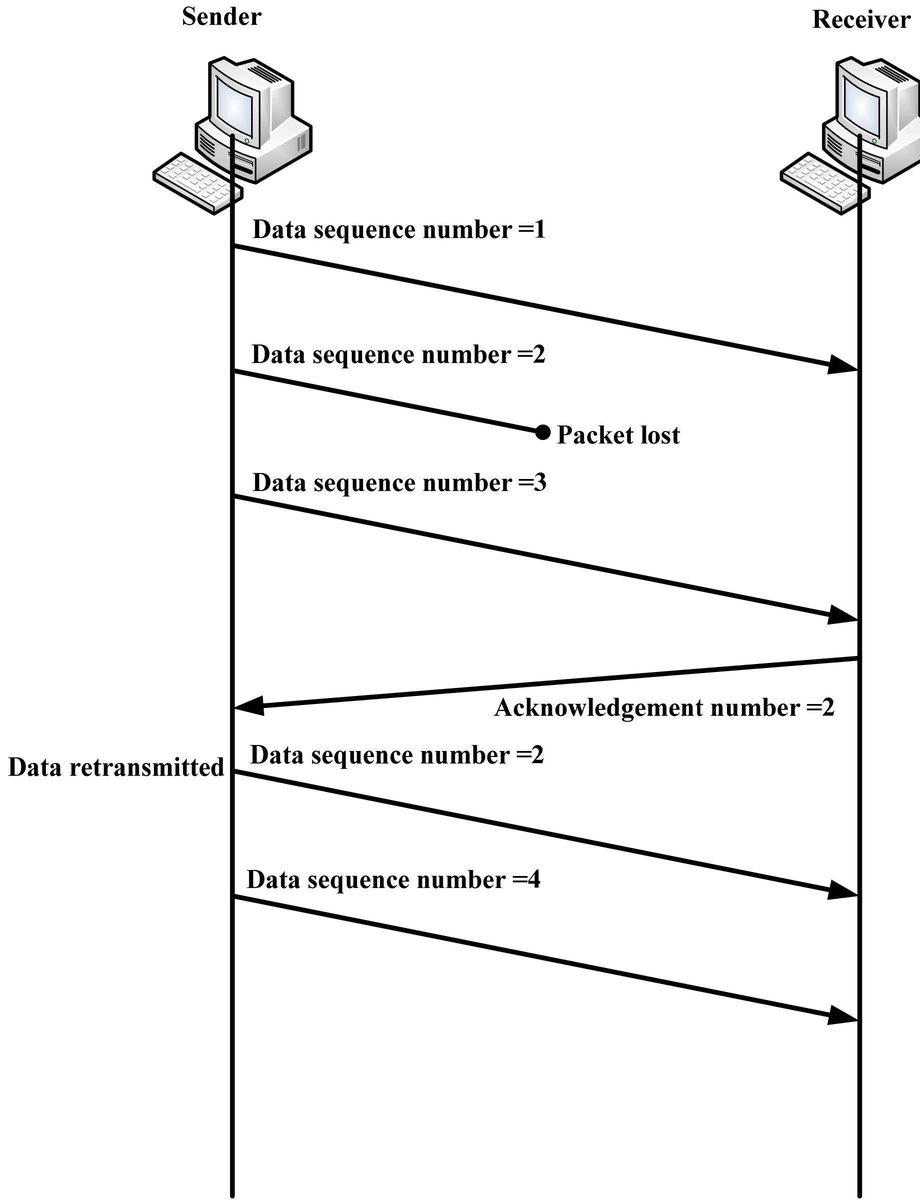

Figure 3.5 Data retransmission of a lost packet

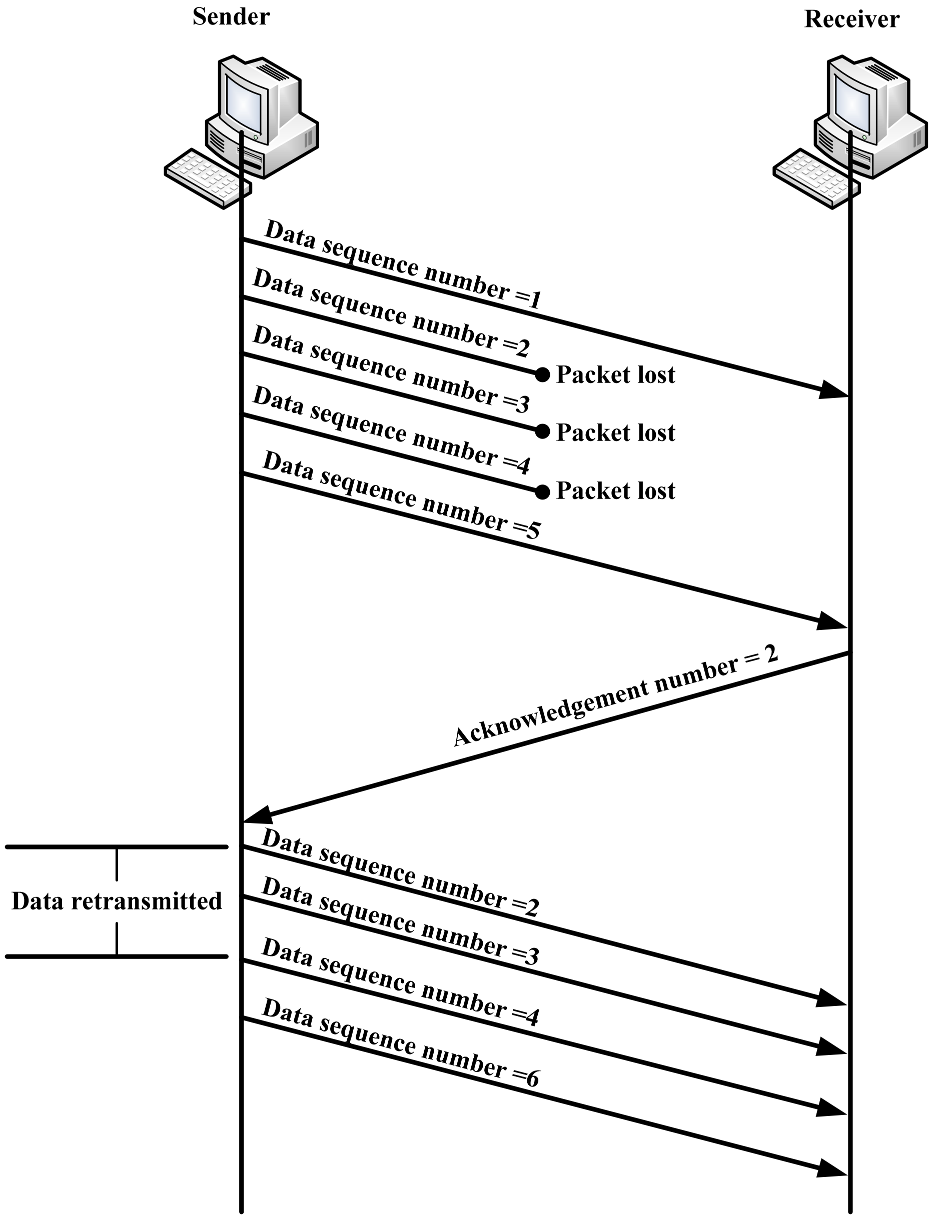

Figure 3.6 Data retransmission of various lost packets

3.5.4 Detection of ineffective data packets

NUDP should have the capability to check the effectiveness of received packets. Only an effective packet can be sent to the upper layer. Out-of-data packets and duplicated packets are types of ineffective packets.

When packet loss occurs, the receiver sends an acknowledgement to inform the sender who will then retransmit the packet. If these packets arrive later than the group of packets in the sequence sent to the application, these retransmission packets are out-of-date. And when sender retransmission packets reach the receiver but have the same packet sequence numbers in the sequencer, the retransmission packets are duplicate data. These two types of packets are rejected by the NUDP receiver. Figure 3.7 shows the situation of out-of-date packets and Figure 3.8 shows the situation of duplicate data.

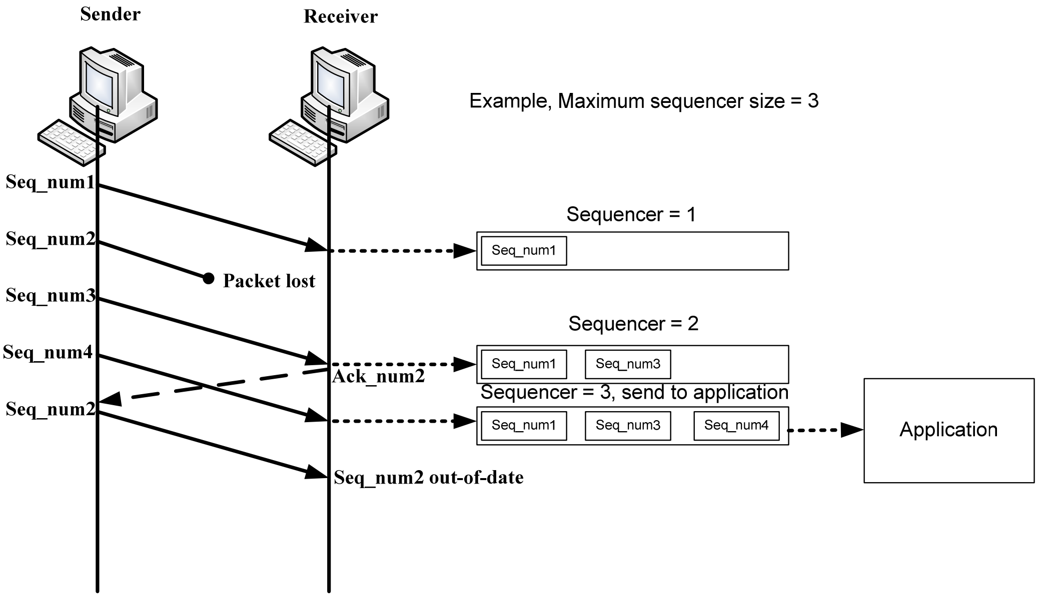

Figure 3.7 Situation Out-of-Date packets

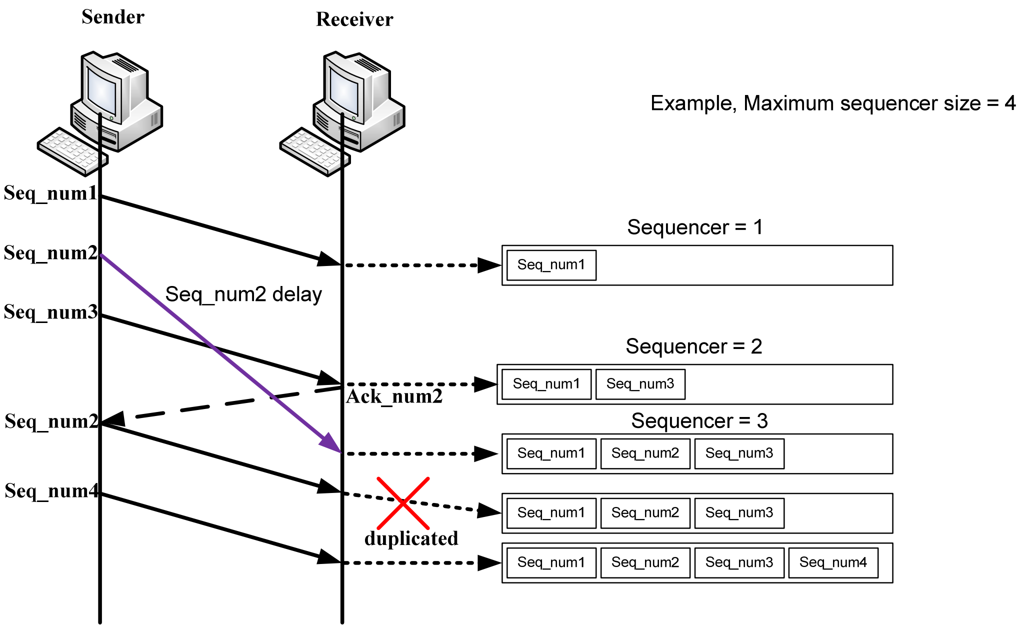

Figure 3.8 Situation duplicate data

3.5.5 NUDP Source Buffer

NUDP uses the retransmission method to compensate for the packet loss rate but NUDP is based on the original UDP which does not have a mechanism for supporting this technique. For this reason, the researchers needed to use a method capable of supporting the retransmission technique. In the present study, the aforementioned method is called a “buffer”. The buffer mechanism is very simple. It receives a datagram from the transport layer, duplicates the datagram and stores it. Finally, the original datagram is transmitted to the destination. Inside buffers use the First-In First-Out algorithm for managing datagram in buffer. First-In First-Out helps control the buffer size limit. Figure 3.9 shows the buffer mechanism and Figure 3.10 shows the First-In First-Out mechanism for use with the buffer mechanism.

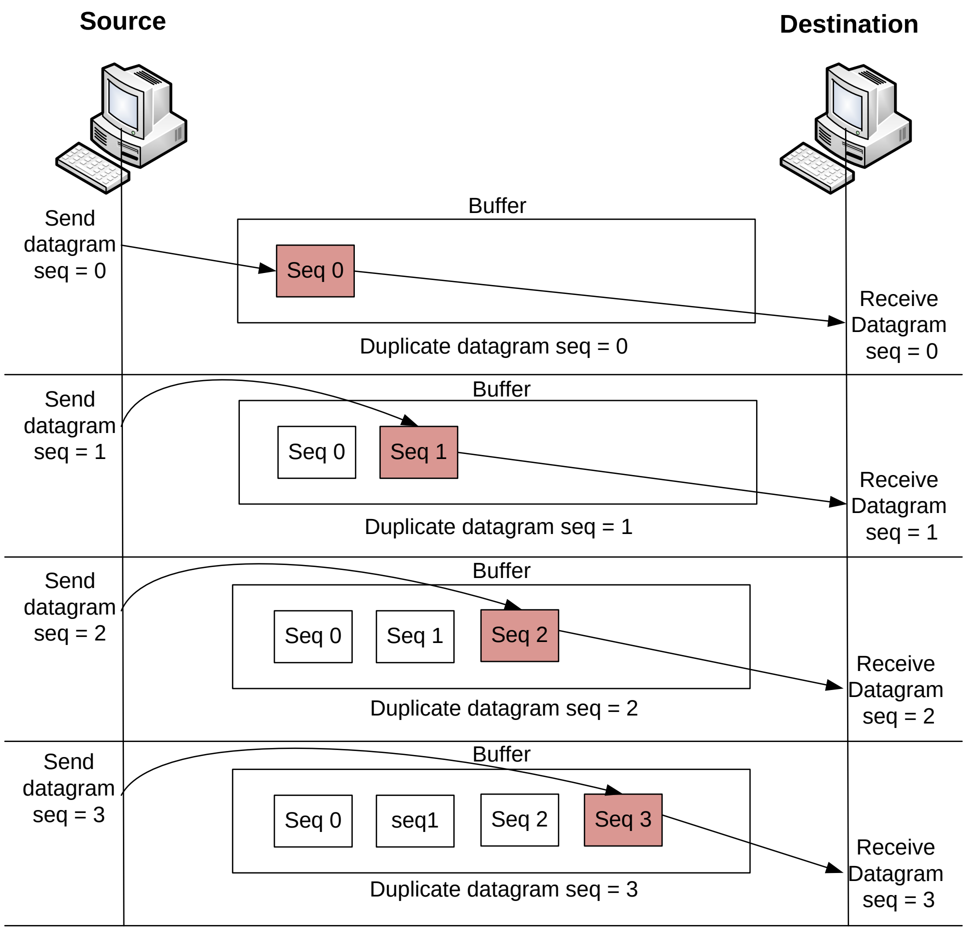

Figure 3.9 Buffer mechanisms

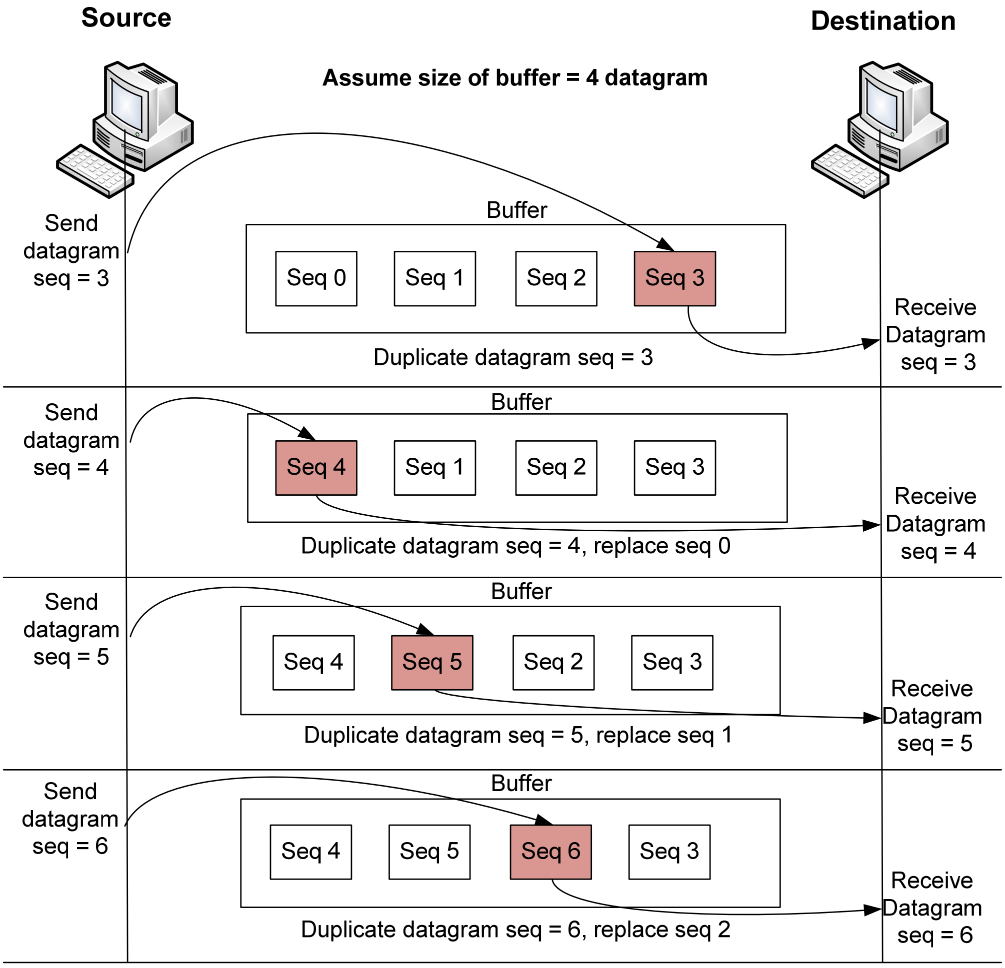

Figure 3.10 First-In First-Out algorithm

The buffer will retransmit datagram when the source receives the acknowledgement packet and searches the datagram by comparing the sequence number of the acknowledgement packet with the sequence number of the datagram contained in the buffer. Next, the datagram is retransmitted to the destination. If, however, the datagram cannot be found, nothing will occur. Figure 3.11 shows the mechanism for retransmitting datagram.

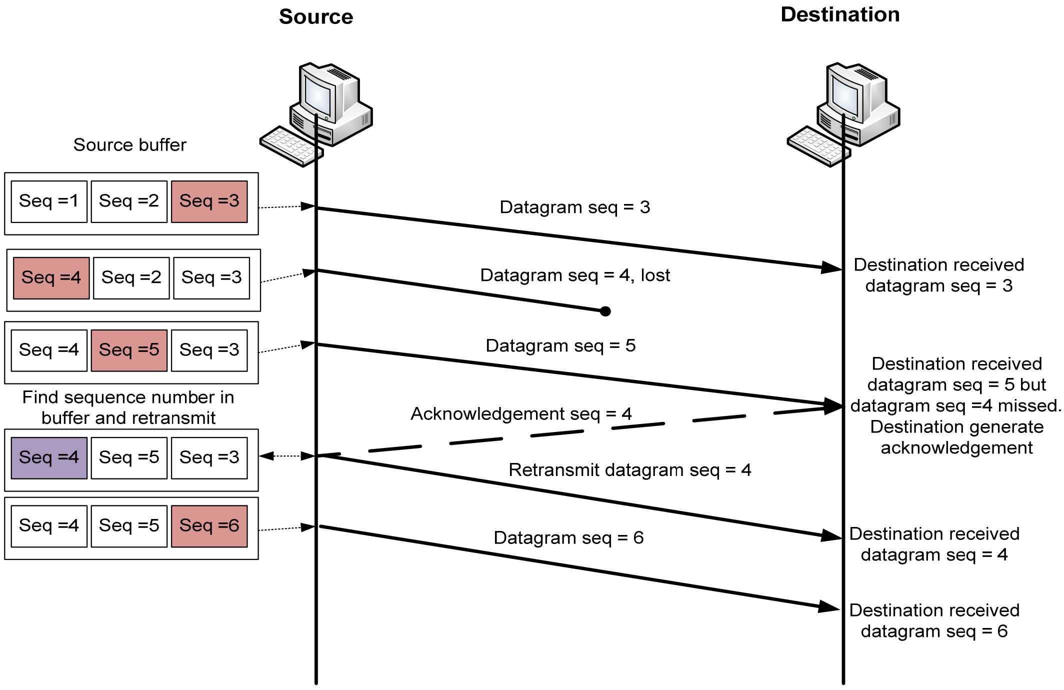

Figure 3.11 Mechanism for retransmitting datagram

3.5.6 Sequencer

Normally of original UDP when the destination received datagram that sent from source, the destination will immediately send a datagram to the application layer. The researchers cannot use this concept in NUDP because NUDP use retransmission method for compensating data lost. The researchers needed to create a method for controlling the retransmission data and that is the sequencer. The sequencer is designed for control, detection of ineffective transmission and re-ordering of data packets before sending to the application layer.

In controlling, the sequencer directs the datagram upon receipt at the destination and will send the datagram to the sequencer. The sequencer holds those datagram until the sequencer is full. Next, the sequencer sends these datagram to the application layer. The controlling process of the sequencer is shown in Figure 3.12 while Figure 3.13 shows how the sequencer control packet is retransmitted from the source.

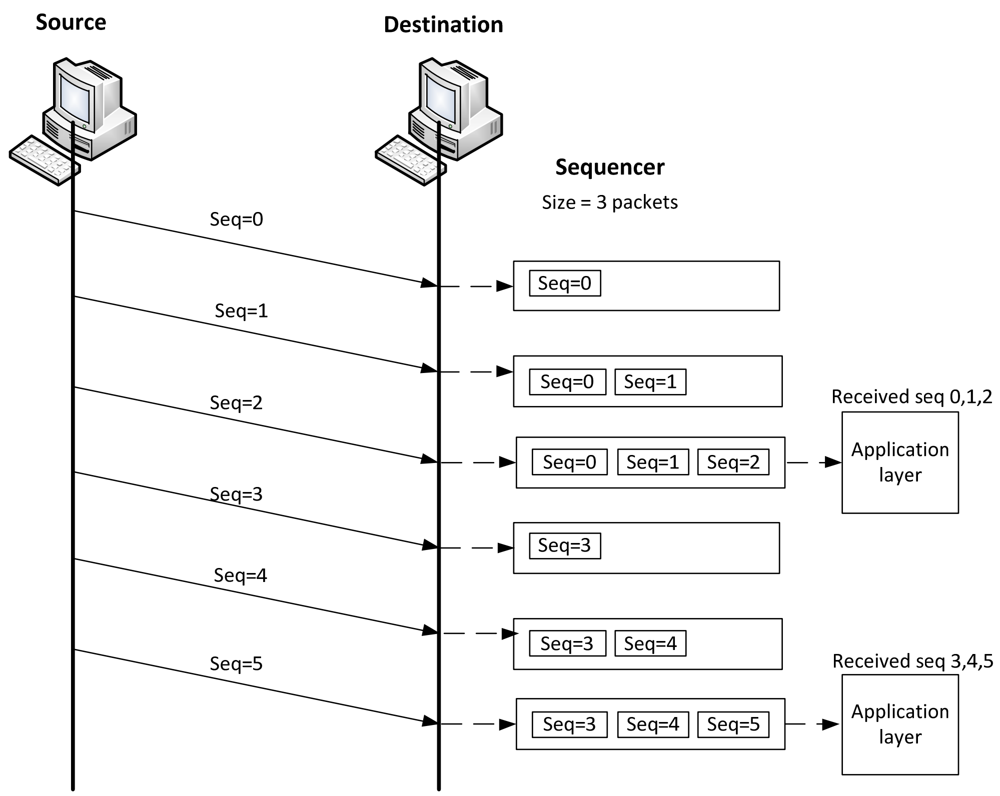

Figure 3.12 Sequencer packet control

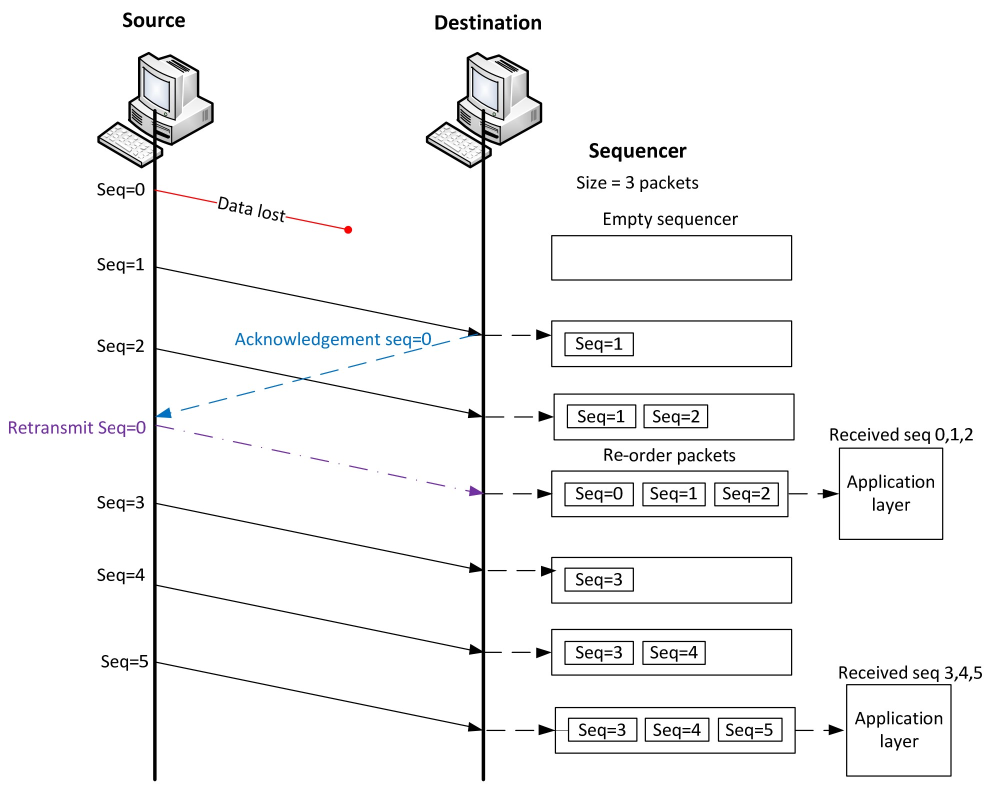

Figure 3.13Retransmitted sequencer control packet

Figure 3.12 shows the situation when the source transmits packets to the destination without retransmitting packets. The sequencer receives packets from the destination and holds them until the sequencer is full. Next, the sequencer sends every packet contained to the application layer. Figure 3.13 shows the situation lost packets occur during packet transmission. The sequencer receives the packets from the destination and holds them until the sequencer is full as Figure 3.12. However, when the sequencer receives the retransmission packets, it will re-order the packets before sending them to the application layer. The detection of ineffective parts is shown in Figures 3.7 and 3.8.

3.6 NUDP packet format

The NUDP packet has a fixed size header of 12 bytes. The header is an extension of the original UDP header. Figure 3.14 shows the NUDP packet format.

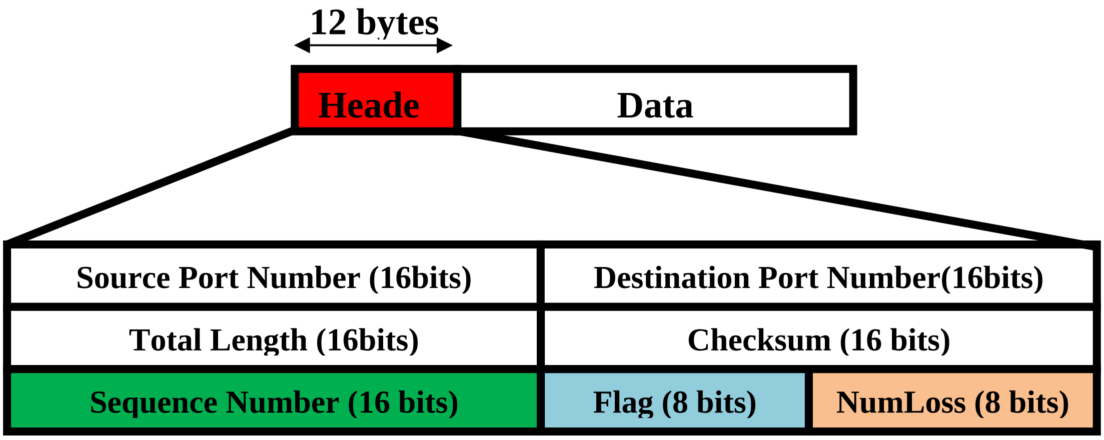

Figure 3.14 NUDP packet formats

According to Figure 3.14, the NUDP keeps the first eight bytes of its header the same as in the original UDP header and adds 4 bytes for use in compensating for the packet loss mechanism that includes the sequence number field, flag field and NumLoss field. The details are as follows:

- Source port number - This is the port number used by the process running on the source host. It is 16 bits long which means that it can range from 0 to 65535.
- Destination port number - This is the port number used by the process running on the destination host. It is 16 bits long, which means that it can range from 0 to 65535.
- Total length - This is a 16-bit field that defines the length of the datagram and header plus data. The minimum length of the datagram is 16 bytes, which is indicates a datagram with only a header and no data.
- Checksum - This field is used to detect transmission errors over the datagram (header plus data). The checksum calculation is exactly the same as that of UDP. When the receiver detects an error through the checksum, the datagram is silently discarded.
- Sequence number - This 16-bit field contains the sequence number of a datagram. It is a necessary part of this research because it is used for the retransmission method. When the NUDP is called by an application to start a data transfers, it will create a sequence number for each datagram according to the incoming sequence. On the other hand, when packet loss occurs, the receiver will create an acknowledgement to inform the sender about the packet loss. The sequence numbers filed for the acknowledgement datagram are the sequence numbers of lost datagram.
- Flag - The flag field of the datagram is provided to identify the type of the datagram. In the present study, the following three types of datagram have been employed: the normal datagram, the acknowledgement datagram and the retransmit datagram. The files inform both the sender and receiver about the type of datagram being transmitted.
- NumLoss – This field contains the number of packets which loss before sending the acknowledgement packet back to sender. This field will have the value when receiver sending the acknowledgment packet back to the sender for informing there is the packet loss occurs in the network.

3.7 NUDP Mechanisms

This part describes NUDP mechanisms. In the original UDP the source only received user demands from the application, divided it into datagram and sent the datagram to the destination in a very simple concept. In the NUDP objective, the researchers needed to retain the simple concept of the original UDP. NUDP receives user demands from the application and divides them into datagram in the same way as the original UDP. Before the data is sent out, however, the source of the NUDP needs to set the sequence number and the flag of the datagram for use in compensating data loss and sending the datagram to the source buffer, not directly to the destination as in the original UDP.

When the source buffer receives the datagram sent from the source, it will check the empty space. If the space is not full, it will duplicate the datagram and keep it in space and send the original datagram to the destination. On the other hand, if there is insufficient space for a datagram, the First In First Out algorithm (FIFO) will be used to manage the space. Next, the algorithm will duplicate the datagram and keep it in the space to send the original datagram to the destination. This process is shown in Figure 3.15.

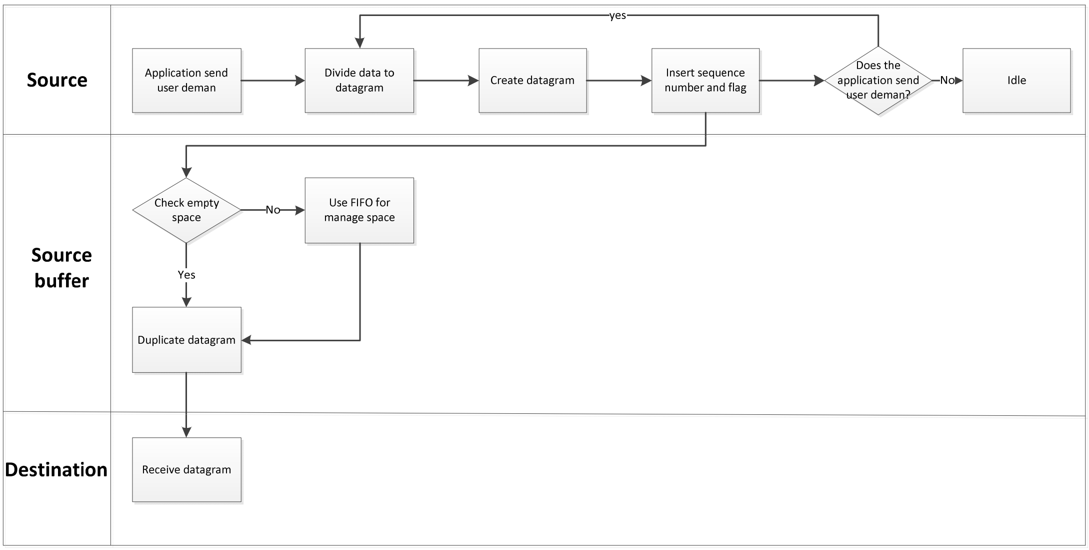

Figure 3.15 Source transmission data mechanism

When the destination receives the datagram, it will immediately send the datagram to the application in the simple process of the original UDP. In NUDP, however, the destination will check to determine whether or not data loss has occurred during the datagram transmission. If data loss has occurred, the destination will check to determine how much data was lost before receiving the datagram. When the destination knows the number of datagram lost, it will create an acknowledgement packet, set a flag to inform the source, namely, the acknowledgement packet, and set the number of data lost. Next, it will send the acknowledgement packet back to the source and send the datagram received to the sequencer as the situation with no packet loss. These processes are shown in Figure 3.16.

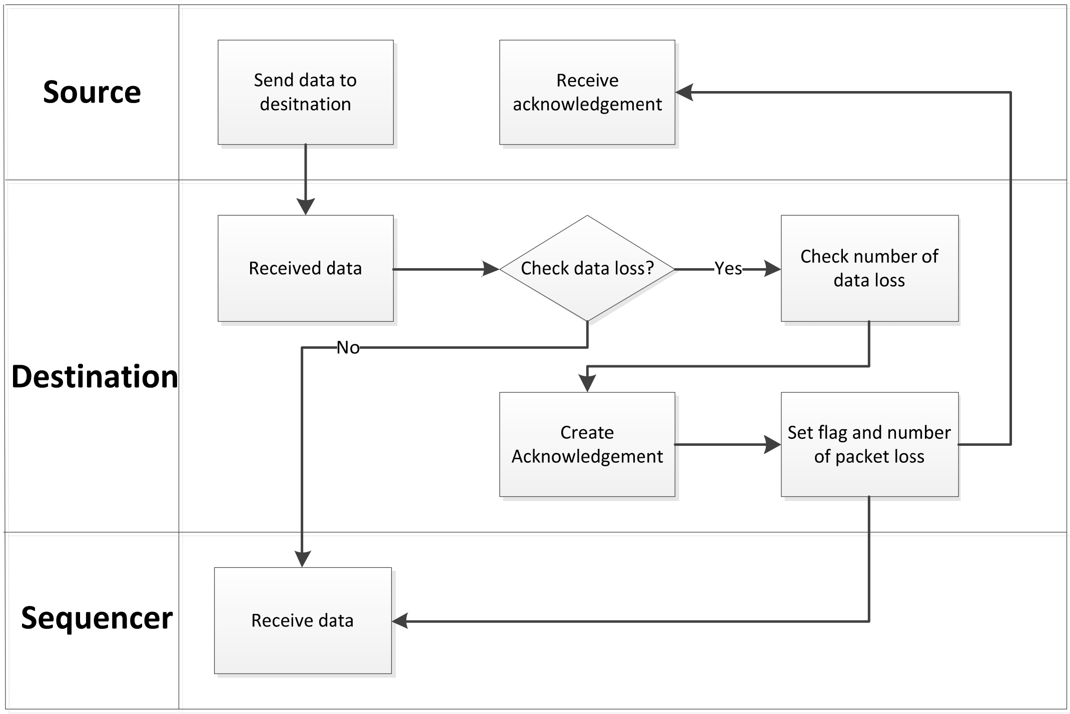

Figure 3.16 Destination mechanism

When the source receives the acknowledgment packet, it will compare the sequence number of the acknowledgement packet with the sequence number of the datagram contained in the source buffer. If they match, a flag will be set for the datagram that is a match informing the destination, namely, the retransmission datagram, and re-sends to the destination. On the other hand, however, if the datagram does not matches the sequence number of the acknowledgement packet, the source will drop the acknowledgement packet. The processes are shown in Figure 3.17.

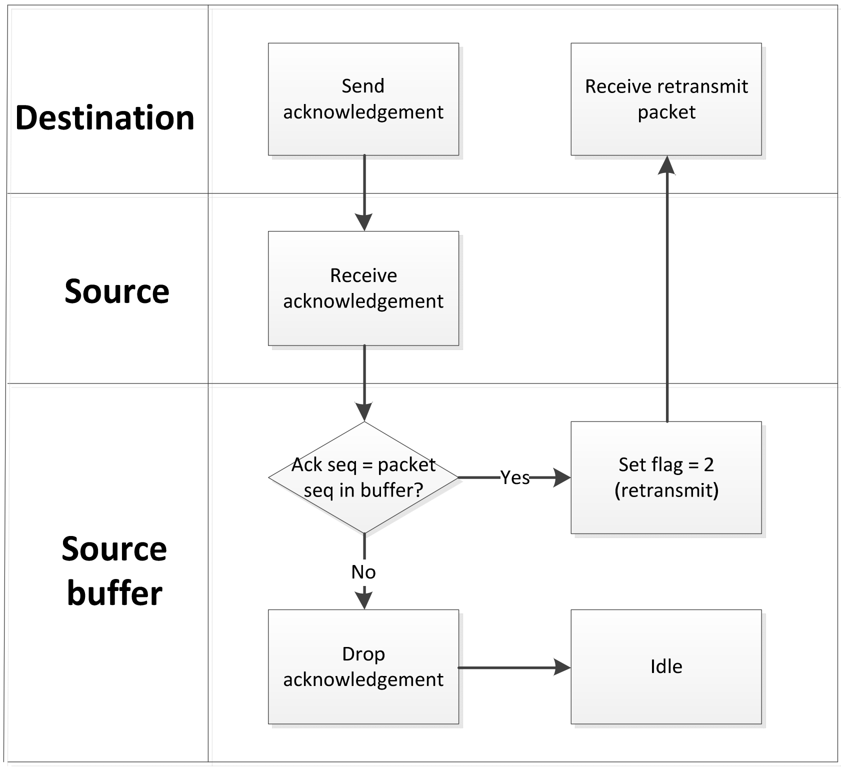

Figure 3.17 Source retransmission mechanisms

When the source sends the datagram to the destination, the destination will send normal and retransmission datagram to the sequencer. When the sequencer receives the datagram, it will check to determine whether or not the datagram is an out-of-data or duplicate datagram. If the datagram is an out-of-date or duplicate datagram, the datagram will be immediately dropped. If the datagram is not an out-of-date or duplicate datagram, it will be re-ordered and stored. The sequencer will store the datagram until the sequencer is full or receives a datagram flagged as last and will send every datagram contained therein to the application. The aforementioned processes are shown in Figure 3.18.

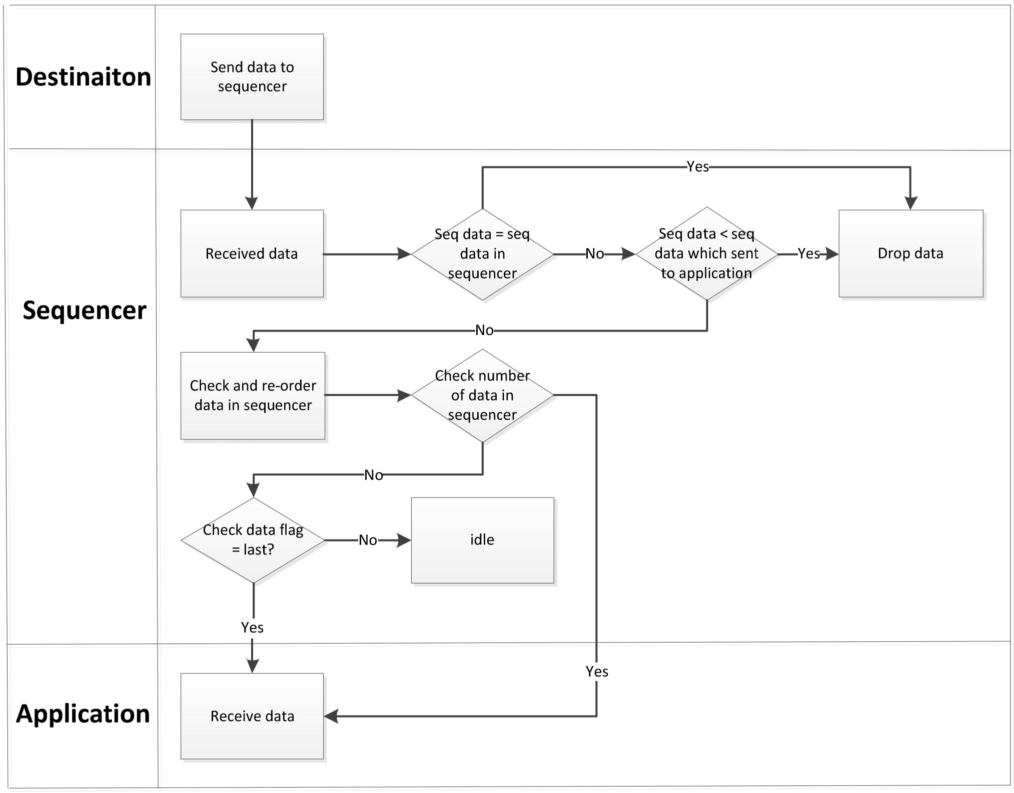

Figure 3.18Sequencer mechanisms

CHAPTER IV
RESULTS

4.1 Network specification

We used sample network topology from project "Mapnet" [18] that was a tool used for visualizing the geographic topology of multiple network infrastructures. This research chose Cable & Wireless Plc. backbone network topology in U.S.[19] for evaluating NUDP performance which showed details in Figure 4.1 and Figure 4.2 showed details network bandwidth of network lines in every nodes.

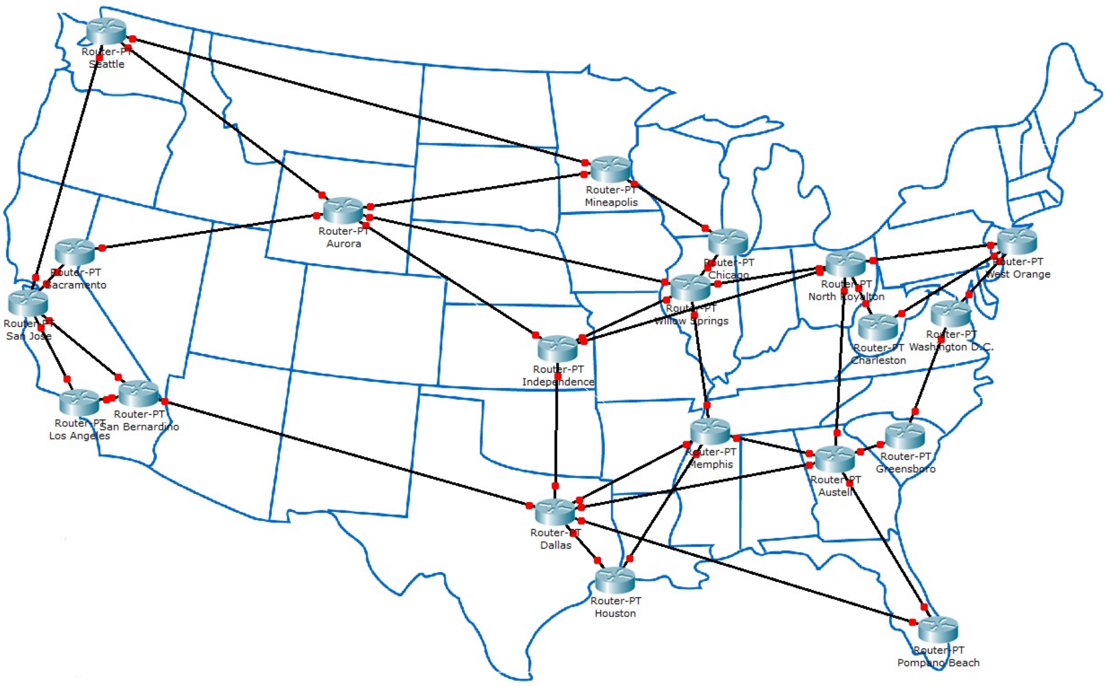

Figure 4.1Cable & Wireless Plc. backbone network topology in U.S

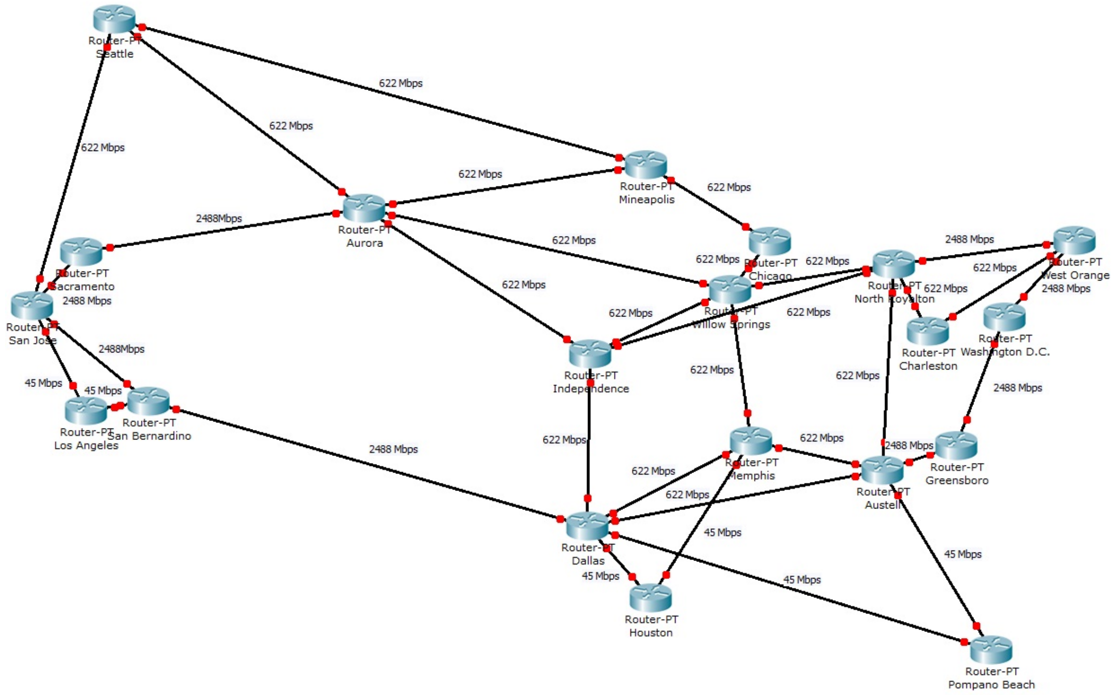

Figure 4.2 Bandwidth of Cable & Wireless Plc. backbone network topology in U.S

4.2 Experiment design

We created above topology in NS-2. At first, we created number of nodes are 20. After that, we defined the link bandwidth by using information from Figure 4.2 we calculated the link propagation delay and setting in simulation. We defined the type of the queues between nodes is DropTail. In the part of router queue size, we did not know about router band and router specification that used in real topology. For this reason, we assumed that we used Cisco Nexus 5500 Platform router [20] in every nodes which means we define the router queue size is 23 packets. We set routing protocol in this topology is link state protocol. Size of packets for using in this topology is 1000 bytes. We run the simulation 30 times with different random seeds (seed = 0) and calculate the mean of results. We summarize the parameters for using in this research following:

Table 4.1 Experimental parameters

Parameters	Values
Nodes number	20 nodes
Node queue type	DropTail
Node queue size	23 packets
Routing protocol	Link state protocol
Packets size	1000 bytes
Number of simulation runs	30 times
Simulation seed	0 means random seed
Intended traffic generator	CBR and Pareto
Background traffic generator	CBR, Pareto, and FTP
Simulation time	5 minute or 600 second

4.3 Performance evaluation

The objective of this research creates the transport layer protocol that can compensate packet lost rate for multimedia traffic. Packet lost rate and delay performance are two essential criteria for transmission multimedia traffic. Therefore, we evaluate performance of NUDP by using the formula. The formula consists of two parts for calculating packet lost rate and delay performance as define below:

4.3.1 Packet lost calculation

Packet lost happen in network communications due to many reasons. Packet lost may occur often when the network's condition gets worse. The quality of multimedia traffic depends on packet lost rate. For this reason, we need to use formula for calculating a packet lost rate. Formula 4.1 shows how to calculate packet lost rate.

$$Packet\ lost\ rate = (L \div S) \times 100 \tag{4.1}$$

In the above formula, L is the number of packets lost and S is the number of packets that send from sender. Normally, S must be larger value than L and L can be zero in normal situation that means there is nothing packet lost.

4.3.2 Average delay calculation

The network delay means how long it takes for a data to travel across the network from sender to receiver. The multimedia traffic main requirement is a small delay. The formula for calculating an average delay shows in Formula 4.2

$$Average\ delay\ = \frac{\sum(Tr_n - Ts_n)}{n} \tag{4.2}$$

In the above formula, Tr is the time when the packet arrived to destination, Ts is the time leave from source and n is number of packets. Normally, Tr must be larger value than Ts.

4.4 Simulation by using TCP

Before we use the NUDP and UDP for transmitting data in every situation in this experiment, we need to proof the TCP does not suitable for transmitting multimedia traffic or using in real-time application.

In this experiment, we used TCP at the source node for sending Intended traffic to destination node. We generated Intended traffic by using FTP.The transmission rate of Intended traffic is 25% of network bandwidth.

In the part of Background traffic, we used Pareto distribution for generating Background traffic and transport layer protocol is UDP.The transmission rate of Background traffics are started at 10% of network bandwidth in every path and increased by 10% in every experiment until the transmission rate of Background traffics equal 100% of network bandwidth.

Table 4.2 showed the packet loss rates result when Intended traffic is TCP and these results are also showed graphically in Figure 4.3. In the part of Table 4.3

showed the average delay result when Intended traffic is TCP and these results are also showed graphically in Figure 4.4.

Table 4.2 TCP Intended traffic packet lost rate

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

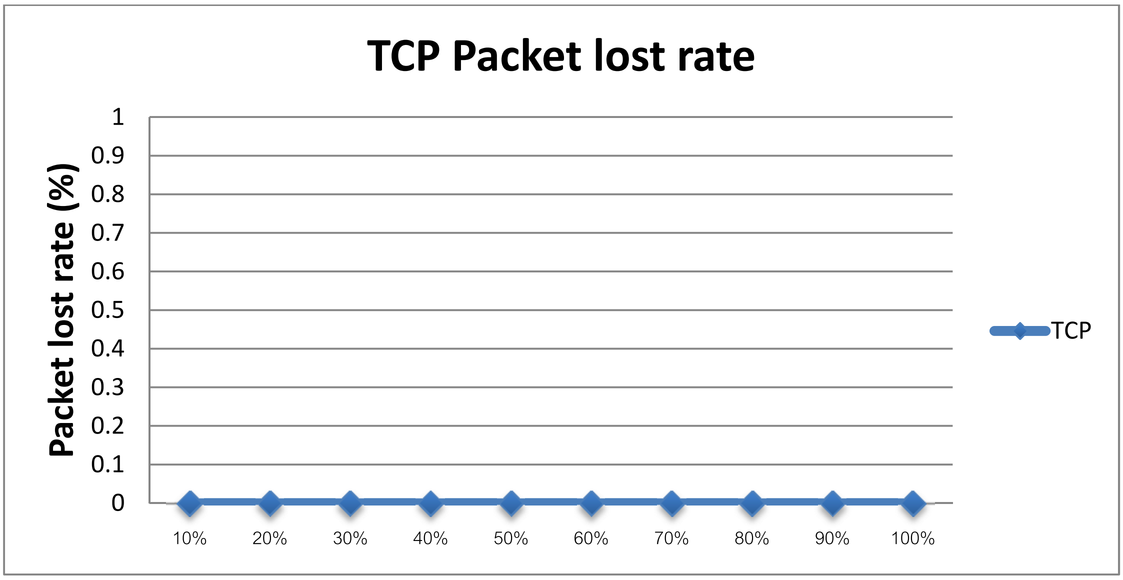

Figure 4.3 TCP packet lost rate

Table 4.3 TCP Intended traffic average delay

Background traffic	Average delay(sec)
10.00%	0.05001448725
20.00%	0.0500163594
30.00%	0.05001823155
40.00%	0.0500201037
50.00%	0.0500201037
60.00%	0.05002572015
70.00%	0.0500425695
80.00%	0.0950969534758
90.00%	0.2346505738381
100.00%	0.39345596654

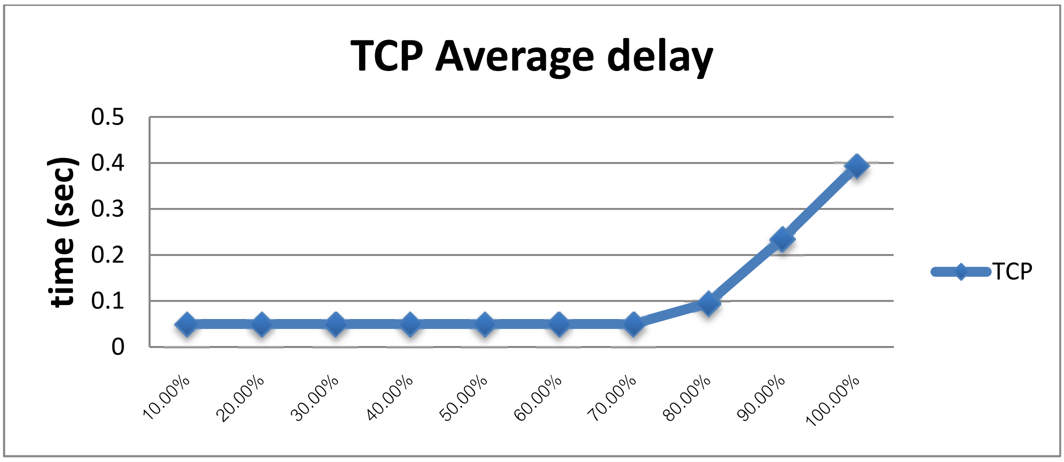

Figure 4.4 TCP average delay

From above results, we can see when Intended traffic is TCP, there is no packet lost occur in every background traffic transmission rate because TCP is a reliable protocol. TCP has mechanism for guarantee the data will transmit to destination. But in the other hand, in average delay term, TCP has huge average delay especially when background traffic transmission rate over than 70%. For this reason TCP does not suitable for transmitting multimedia traffic or using in real-time application because they concern about delay.

4.5 Simulation case one

In this case, we assume there are many transitions sending from source node to destination node. In this research, these transitions called Intended traffic. We specify the source of Intended traffic which is node number 16 and the destination of Intended traffic is node number 18.

The path of Intended traffic start from node number 16 to node number 0, node number 0 to node number 7, node number 7 to node number 11, and final path is node number 11 to node number 18. This path came from routing protocol that was the best path for communication between source node to destination node.

We generate traffic in the every path that Intended traffic passes it when source node sends Intended traffic to destination node at the same time. These traffics called Background traffic. The Background traffics have four paths consist of:

1. Sending from node number 16 to node number 0.
2. Sending from node number 0 to node number 7.
3. Sending from node number 7 to node number 11.
4. Sending from node number 11 to node number 18.

We divided transmission rate of Intended traffic in three situations consists of 25%, 50%, and 75% of network bandwidth. The transmission rate of Background traffics are started at 10% of network bandwidth in every path and increased by 10% in every experiment until the transmission rate of Background traffics equal 100% of network bandwidth.

In this experiment, we used both UDP and NUDP at the source node for sending Intended traffic to destination node. We generated Intended traffic by using CBR and Pareto distribution.

In the part of Background traffic, we used FTP and Pareto distribution for generating Background traffic when the transport layer protocol is TCP and used CBR and Pareto distribution when used UDP. Table 4.4 lists the condition of Intended traffic and table 4.5 lists the condition of Background traffic.

Table 4.4 Intended traffic conditions

Intended traffic transmission rate	Intended traffic transport layer protocol	Intended traffic generator
25% = 155.5 Mbps 50% = 311 Mbps 75% = 466.5 Mbps	UDP	CBR
		Pareto
	NUDP	CBR
		Pareto

Table 4.5 Background traffic condition

Background traffic transmission rate	Background traffic transport layer protocol	Background traffic generator	
10% 20% 30% 40% 50% 60% 70% 80% 90% 100%	UDP	CBR	Pareto
	TCP	FTP	Pareto

The simulation time for this case is 5 minutes or 300 seconds. The source node of Intended traffic starts communication with the destination node at 2.0s and stop at 302.0s. All of the source nodes of Background traffic start communication with the destination node at 0.0s and stop at 302.5s.

4.5.1 Result of the simulation

We divided simulation results into two parts including packet lost rate and average delay which shows in below:

4.5.1.1 Packet lost rate result

This part showed the simulation results in the part of the packet lost rate. We divided results of the simulation in this part into three parts depend on Intended traffic transmission rate including:

1) Intended traffic transmission rate is 25% of the network bandwidth.

2) Intended traffic transmission rate is 50% of the network bandwidth.

3) Intended traffic transmission rate is 75% of the network bandwidth.

1) Intended traffic transmission rate is 25% of the network bandwidth

In this part, we specify transmission rate of the Intended traffic into 25% of the network bandwidth that means the transmission rate of the Intended traffic is 155.5 Mbps. The result divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

1.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.6 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.7 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.5

Table 4.6 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	32.2772611
90%	45.70435767
100%	64.79168496

Table 4.7 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	32.2772611	15.12902476
90%	45.70435767	36.96682546
100%	64.79168496	57.72162736

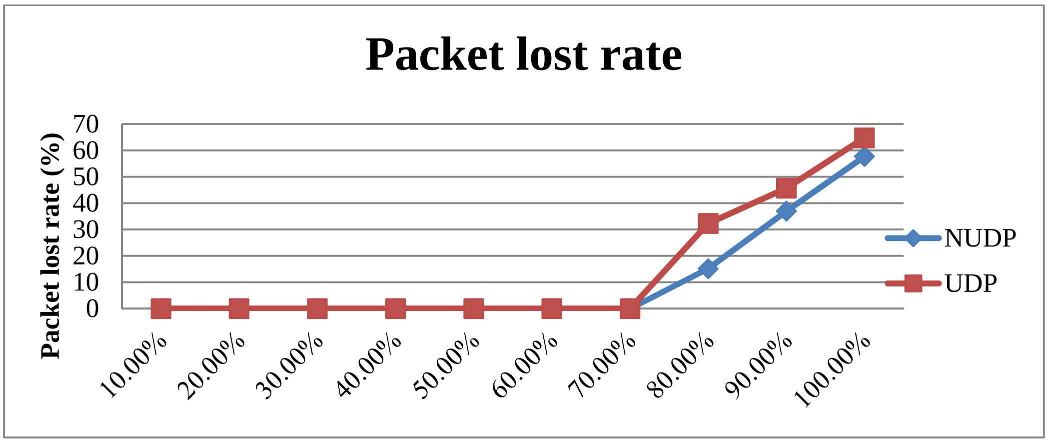

Figure 4.5 Packet loss rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.8 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.9 showed packet lost rate when

transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.6

Table 4.8 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	9.1416566
90%	33.182009
100%	56.99999

Table 4.9 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	9.1416566	3.8590091
90%	33.182009	20.731345
100%	56.99999	37.642891

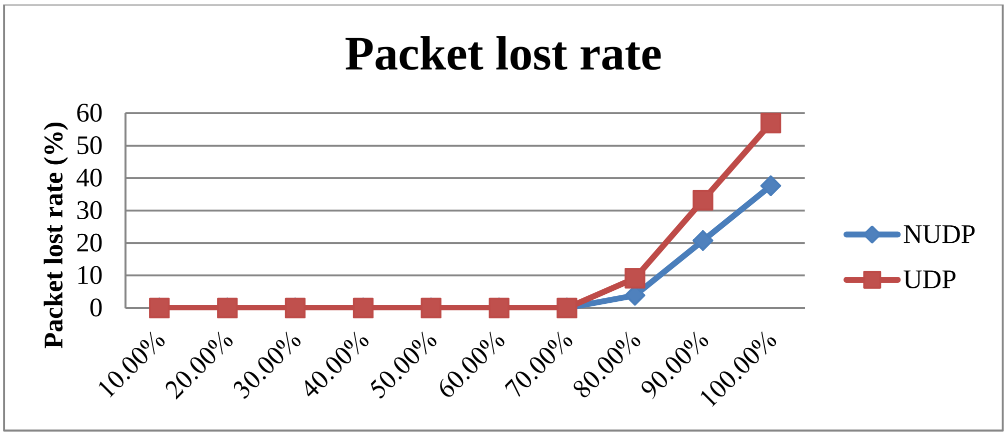

Figure 4.6 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.10 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.11 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.10 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.11 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.12 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.13 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.12 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.13 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

.

1.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generate by CBR. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.14 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.15 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.7

Table 4.14 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	21.484153
90%	47.723256
100%	68.211148

Table 4.15 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	21.484153	8.8406972
90%	47.723256	35.37729
100%	68.211148	61.655912

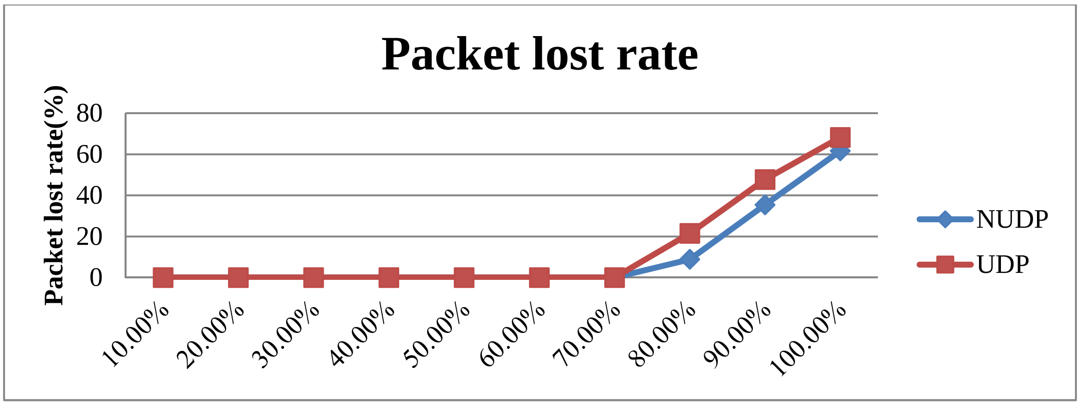

Figure 4.7 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.16 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.17 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.8

Table 4.16 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	10.592129
90%	35.890237
100%	51.016766

Table 4.17 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	10.592129	3.6957036
90%	35.890237	23.159338
100%	51.016766	43.744096

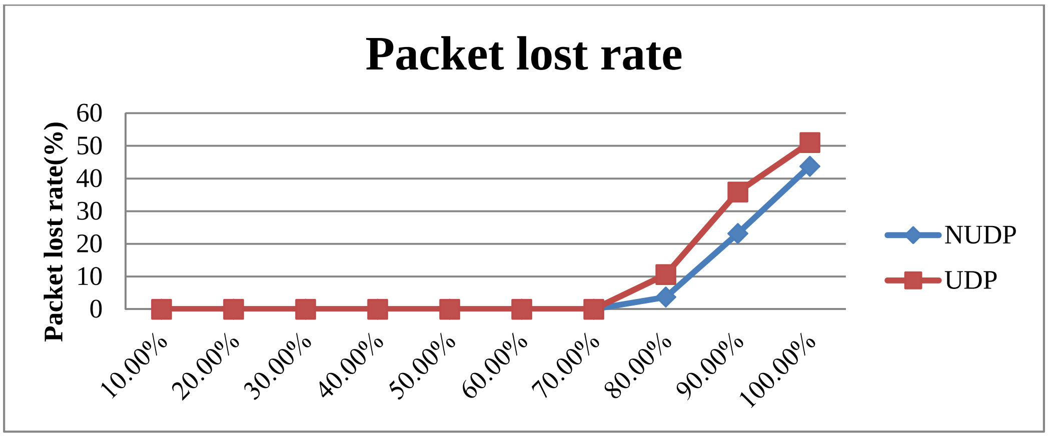

Figure 4.8 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.18 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.19 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.18 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.19 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.20 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.21 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.20 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.21 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

2) Intended traffic transmission rate is 50% of network bandwidth

In this part we specify transmission rate of the Intended traffic into 50% of the network bandwidth that means the transmission rate of the Intended traffic is 311 Mbps. The result divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

2.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.22 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.23 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.9

Table 4.22 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	1.571157
60%	23.92001
70%	36.86713
80%	55.3476
90%	62.26885
100%	72.6492

Table 4.23 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	1.571157	0.163894
60%	23.92001	14.69724
70%	36.86713	17.30883
80%	55.3476	50.13609
90%	62.26885	56.59135
100%	72.6492	67.30289

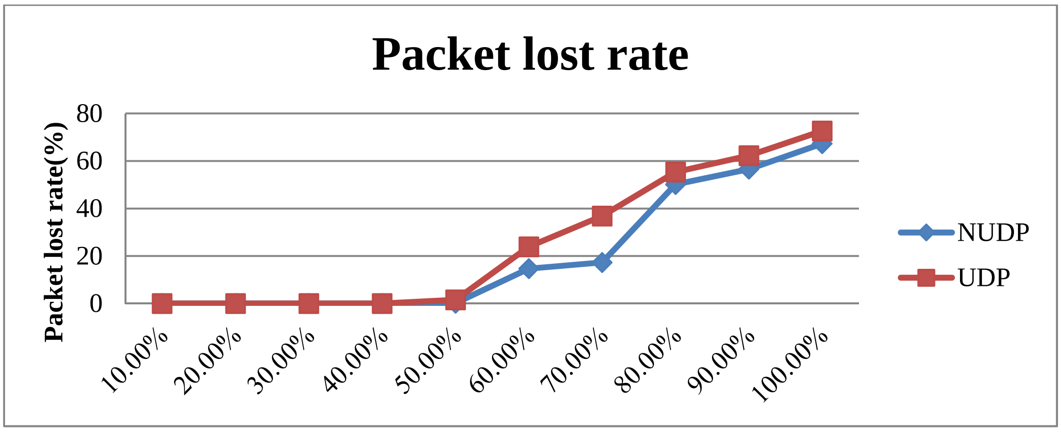

Figure 4.9 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.24 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.25 showed packet lost rate when

transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.10

Table 4.24 UDP Intended traffic packet lost rate over Pareto distribution and UDP background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0.149486
60%	16.41657
70%	29.71546
80%	37.82404
90%	47.89939
100%	59.76013

Table 4.25 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0.149486	0.010041
60%	16.41657	6.756473
70%	29.71546	17.48471
80%	37.82404	25.60763
90%	47.89939	35.3923
100%	59.76013	45.44601

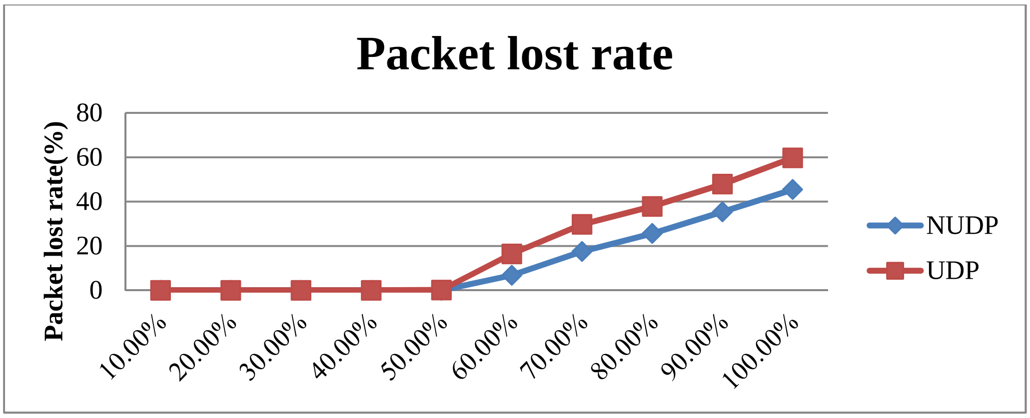

Figure 4.10 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.26 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.27 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.26 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.27 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.28 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.29 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.28 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.29 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

2.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generate by Pareto distribution. We specify the transport layer protocol of the Intended traffic both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.30 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.31 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.11

Table 4.30 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0.810819
60%	29.22269
70%	45.06885
80%	56.29804
90%	63.16148
100%	79.18396

Table 4.31 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0.810819	0.060267
60%	29.22269	12.79063
70%	45.06885	28.90386
80%	56.29804	42.9207
90%	63.16148	54.65321
100%	79.18396	75.54939

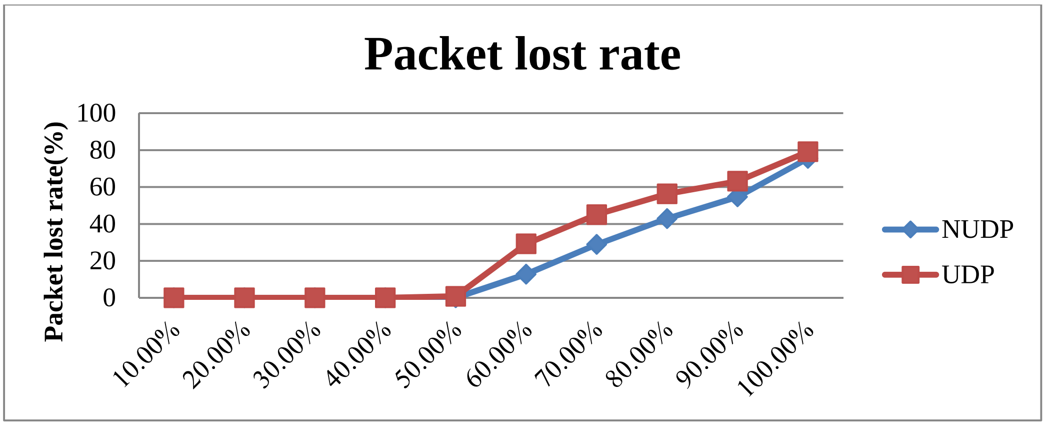

Figure 4.11 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.32 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.33 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.12

Table 4.32 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	19.12888
70%	32.03928
80%	40.71166
90%	52.84175
100%	63.44014

Table 4.33 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	19.12888	6.163701
70%	32.03928	16.11235
80%	40.71166	26.938
90%	52.84175	42.86756
100%	63.44014	56.71062

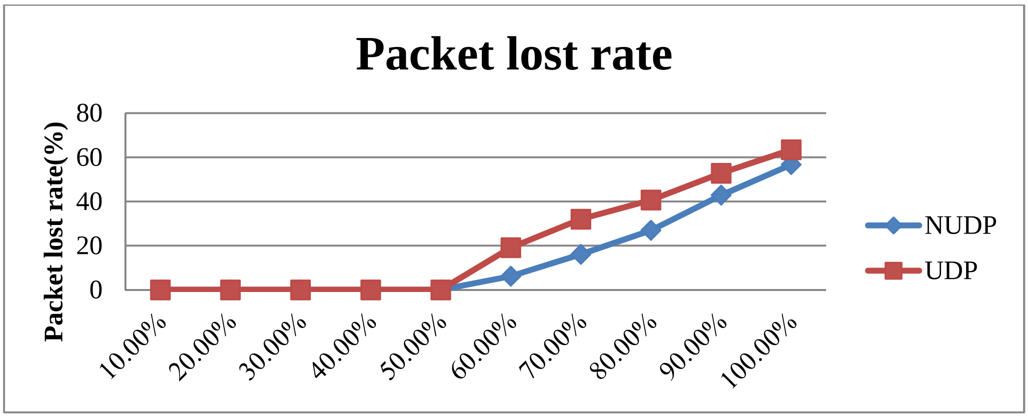

Figure 4.12 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.34 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.35 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.34 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.35 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.36 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.37 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.36 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	**Packet loss rate (%)**
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.37 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

3) Intended traffic transmission rate is 75% of network bandwidth

In this part we specify transmission rate of the Intended traffic into 75% of the network bandwidth that means the transmission rate of the Intended traffic is 466.5 Mbps. The result divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

3.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the transport layer protocol of the Intended traffic both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.38 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.39 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.13

Table 4.38 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	14.591499
40%	30.122876
50%	27.379694
60%	53.814269
70%	62.378767
80%	74.196618
90%	79.2036
100%	90.572265

Table 4.39 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	14.591499	6.0852886
40%	30.122876	17.804474
50%	27.379694	15.945088
60%	53.814269	45.087447
70%	62.378767	56.585271
80%	74.196618	66.130778
90%	79.2036	76.392565
100%	90.572265	89.999802

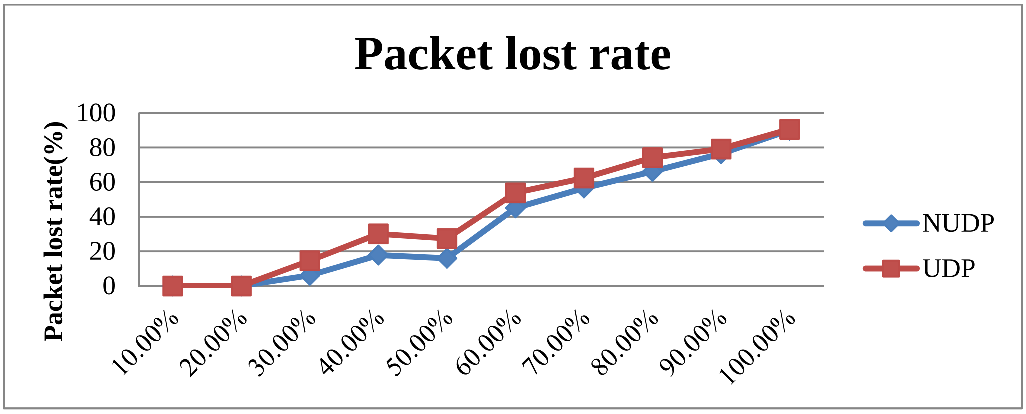

Figure 4.13 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.40 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.41 showed packet lost rate when

transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.14

Table 4.40 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	9.3696044
40%	20.098547
50%	29.590274
60%	40.084025
70%	47.891286
80%	52.744544
90%	64.159728
100%	75.625826

Table 4.41 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	9.3696044	2.7071294
40%	20.098547	9.5522336
50%	29.590274	19.158477
60%	40.084025	26.847294
70%	47.891286	36.814509
80%	52.744544	42.951161
90%	64.159728	55.422578
100%	75.625826	63.107525

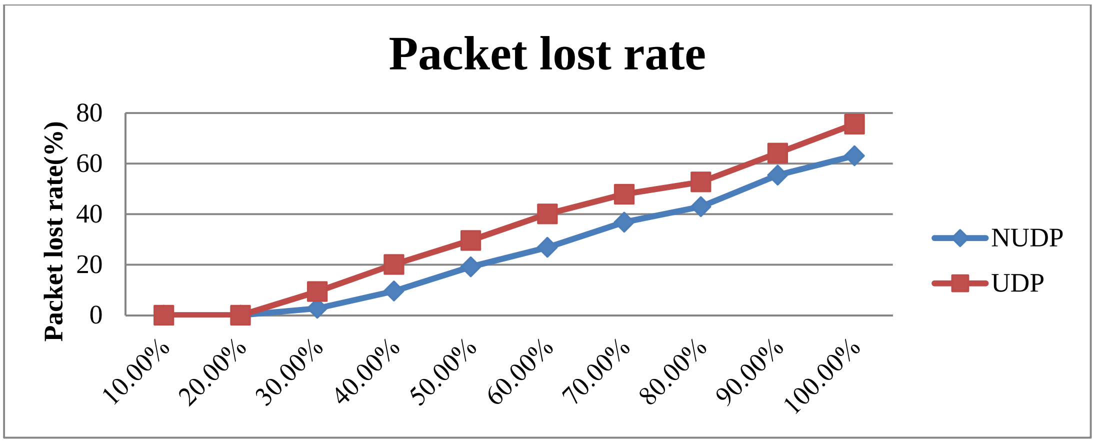

Figure 4.14 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is UDP. Table 4.42 showed packet lost rate when transport layer protocol of Intended traffic is TCP and Table 4.43 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.42 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.43 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.44 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.45 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.44 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.45 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

3.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generate by Pareto distribution. We specify the transport layer protocol of the Intended traffic both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.46 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.47 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.15

Table 4.46 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	15.87656
40%	30.39178
50%	42.93319
60%	50.79408
70%	62.11814
80%	70.33691
90%	75.65355
100%	81.98364

Table 4.47 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	15.87656	4.369791
40%	30.39178	15.618811
50%	42.93319	25.944909
60%	50.79408	37.295175
70%	62.11814	51.91118
80%	70.33691	62.584794
90%	75.65355	70.066481
100%	81.98364	79.476919

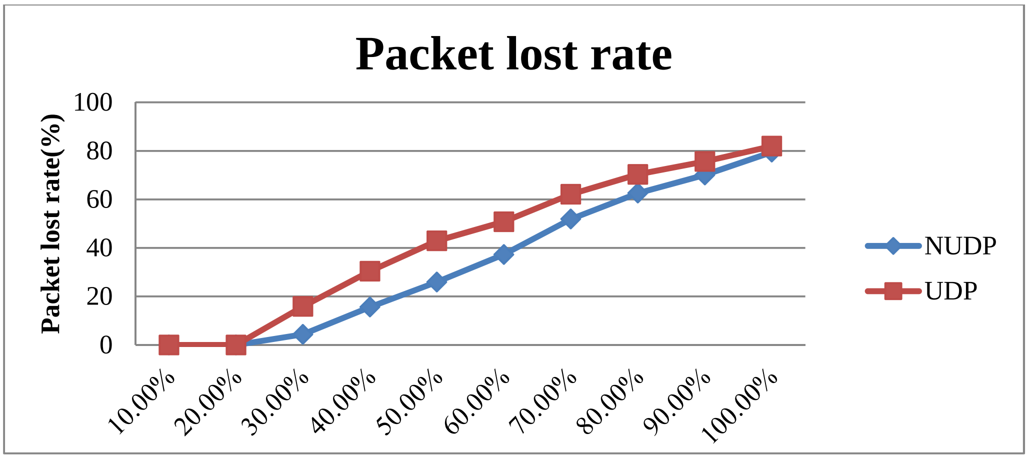

Figure 4.15 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.48 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.49 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.16

Table 4.48 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	10.32211
40%	21.46588
50%	26.66661
60%	39.57759
70%	45.87465
80%	49.53433
90%	60.46972
100%	65.59186

Table 4.49 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	10.32211	2.6710817
40%	21.46588	8.8080432
50%	26.66661	15.524124
60%	39.57759	23.09738
70%	45.87465	32.728407
80%	49.53433	37.891921
90%	60.46972	50.996463
100%	65.59186	60.872491

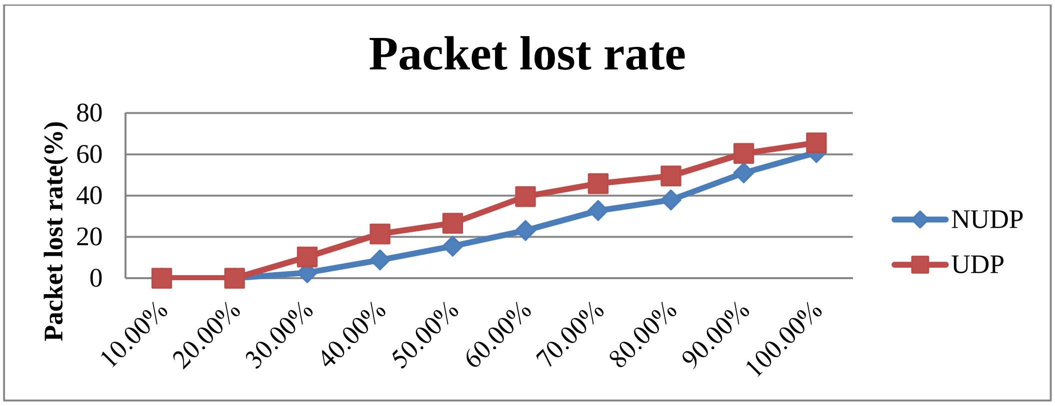

Figure 4.16 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.50 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.51 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.50 UDP Intended traffic packet lost rate over FTP and TCP background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.51 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate lost rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic packet lost rate when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.52 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.53 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.52 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10%	0
20%	0
30%	0
40%	0
50%	0
60%	0
70%	0
80%	0
90%	0
100%	0

Table 4.53 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10%	0	0
20%	0	0
30%	0	0
40%	0	0
50%	0	0
60%	0	0
70%	0	0
80%	0	0
90%	0	0
100%	0	0

4.5.1.2 Average delay result

This part showed the simulation result in the part of the average delay. In this part we calculate the average delay from the 10000 packets that receiver can receive it. We divided results of the simulation in this part into three parts depend on Intended traffic transmission rate including:

1) Intended traffic transmission rate is 25% of the network bandwidth.

2) Intended traffic transmission rate is 50% of the network bandwidth.

3) Intended traffic transmission rate is 75% of the network bandwidth.

1) Intended traffic transmission rate is 25% of network bandwidth

In this part we specify transmission rate of the Intended traffic into 25% of the network bandwidth that means the transmission rate of the Intended traffic is 155.5 Mbps. The result divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

1.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.54 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.55 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.17

Table 4.54 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026717
30.00%	0.026719
40.00%	0.026721
50.00%	0.026724
60.00%	0.026728
70.00%	0.026748
80.00%	0.027816
90.00%	0.027874
100.00%	0.027949

Table 4.55 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026717
30.00%	0.026719
40.00%	0.026721
50.00%	0.026724
60.00%	0.026728
70.00%	0.026748
80.00%	0.037551
90.00%	0.034712
100.00%	0.038795

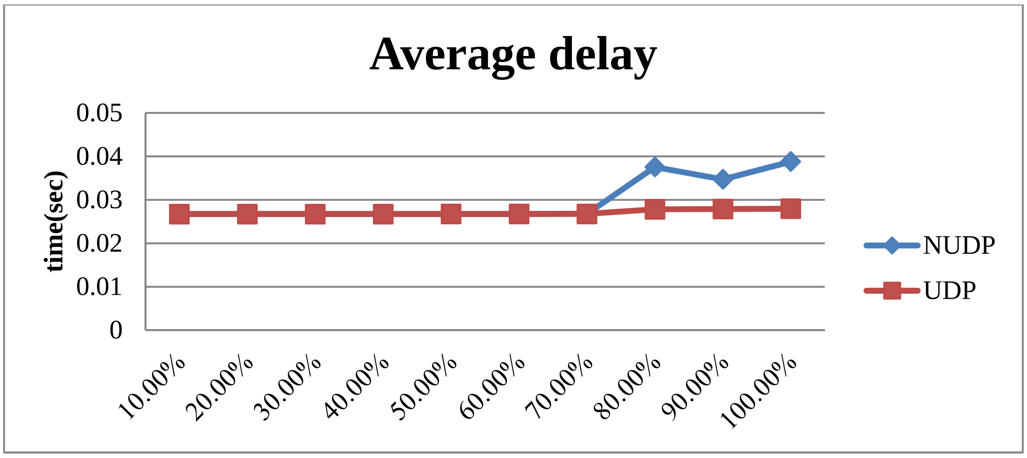

Figure 4.17 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.56 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.57 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.18

Table 4.56 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.02716
30.00%	0.026717
40.00%	0.026718
50.00%	0.026718
60.00%	0.026721
70.00%	0.02673
80.00%	0.027613
90.00%	0.027671
100.00%	0.027608

Table 4.57 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026716
30.00%	0.026717
40.00%	0.026718
50.00%	0.026718
60.00%	0.026721
70.00%	0.02673
80.00%	0.034133
90.00%	0.045209
100.00%	0.051961

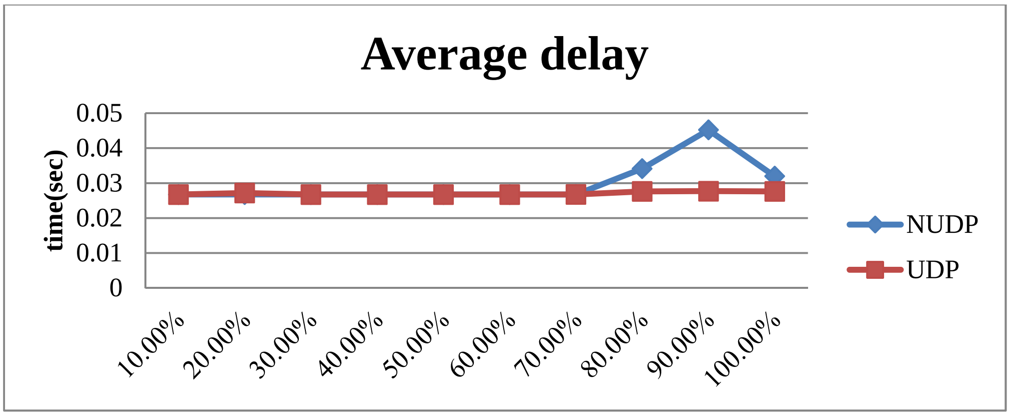

Figure 4.18 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.58 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.59 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.58 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026714
70.00%	0.026714
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

Table 4.59 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026714
70.00%	0.026714
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.60 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.61 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.60 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026715
70.00%	0.026714
80.00%	0.026714
90.00%	0.026715
100.00%	0.026715

Table 4.61 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026715
70.00%	0.026714
80.00%	0.026714
90.00%	0.026715
100.00%	0.026715

1.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generates by Pareto distribution. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.62 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.63 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.19

Table 4.62 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026717
30.00%	0.026719
40.00%	0.02672
50.00%	0.026723
60.00%	0.026726
70.00%	0.026743
80.00%	0.027787
90.00%	0.02786
100.00%	0.027943

Table 4.63 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026717
30.00%	0.026719
40.00%	0.02672
50.00%	0.026723
60.00%	0.026726
70.00%	0.026743
80.00%	0.033638
90.00%	0.03621
100.00%	0.039756

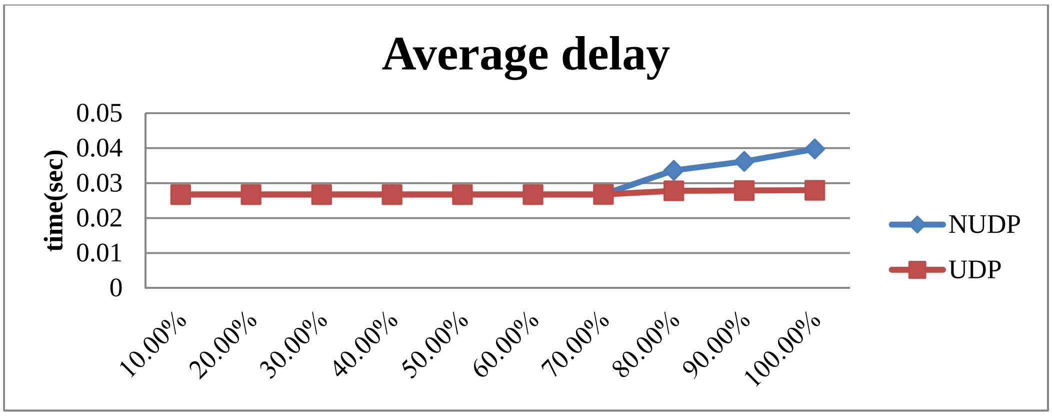

Figure 4.19 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.64 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.65 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.20

Table 4.64 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026716
30.00%	0.026717
40.00%	0.026719
50.00%	0.026714
60.00%	0.026722
70.00%	0.02673
80.00%	0.027612
90.00%	0.027701
100.00%	0.027683

Table 4.65 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026716
30.00%	0.026717
40.00%	0.026719
50.00%	0.026714
60.00%	0.026722
70.00%	0.02673
80.00%	0.032739
90.00%	0.041013
100.00%	0.029441

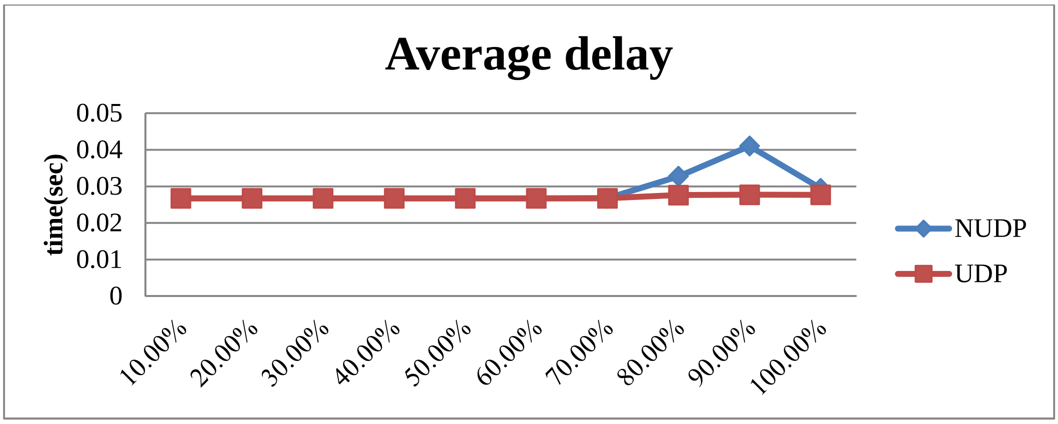

Figure 4.20 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.66 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.67 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.66 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026714
70.00%	0.026714
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

Table 4.67 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026714
50.00%	0.026714
60.00%	0.026714
70.00%	0.026714
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.68 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.69 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.68 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026715
50.00%	0.026714
60.00%	0.026714
70.00%	0.026715
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

Table 4.69 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026714
30.00%	0.026714
40.00%	0.026715
50.00%	0.026714
60.00%	0.026714
70.00%	0.026715
80.00%	0.026714
90.00%	0.026714
100.00%	0.026714

2) Intended traffic transmission rate is 50% of network bandwidth

In this part we specify transmission rate of the Intended traffic into 50% of the network bandwidth that means the transmission rate of the Intended traffic is 311 Mbps. The result divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

2.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.70 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.71 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.21

Table 4.70 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026719
30.00%	0.026723
40.00%	0.026732
50.00%	0.027244
60.00%	0.027845
70.00%	0.027872
80.00%	0.027878
90.00%	0.027886
100.00%	0.027956

Table 4.71 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026719
30.00%	0.026723
40.00%	0.026732
50.00%	0.027508
60.00%	0.03214
70.00%	0.038013
80.00%	0.033012
90.00%	0.034447
100.00%	0.036125

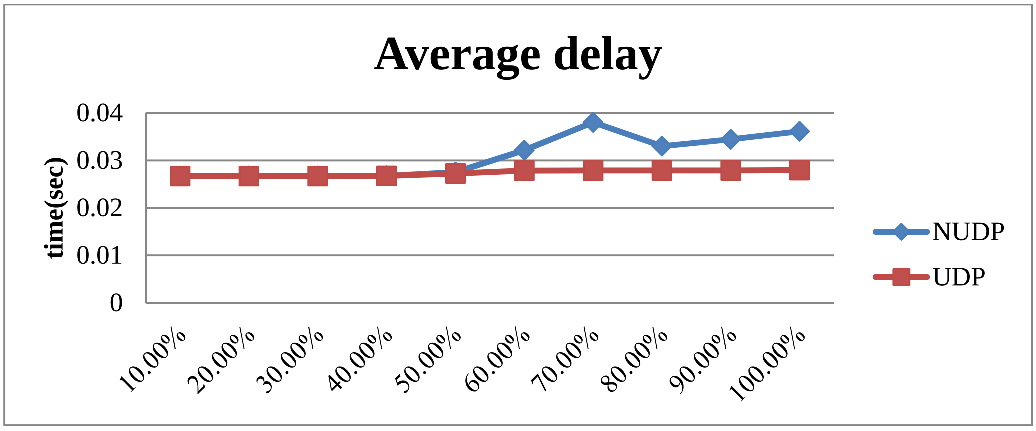

Figure 4.21 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.72 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.73 showed average delay when

transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.22

Table 4.72 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026717
30.00%	0.026718
40.00%	0.026725
50.00%	0.026918
60.00%	0.027648
70.00%	0.027653
80.00%	0.027626
90.00%	0.027625
100.00%	0.02765

Table 4.73 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026717
30.00%	0.026718
40.00%	0.026725
50.00%	0.026918
60.00%	0.037931
70.00%	0.035785
80.00%	0.036423
90.00%	0.044866
100.00%	0.032178

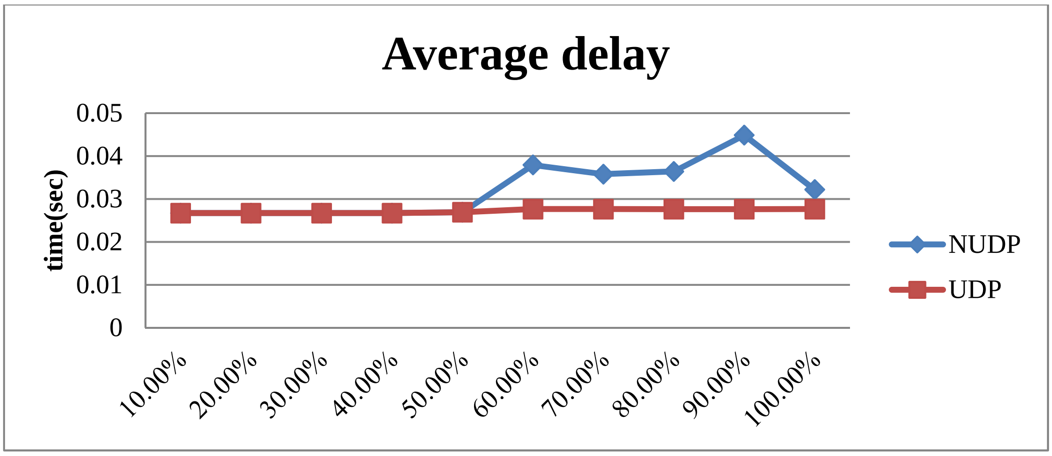

Figure 4.22 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.74 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.75 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.74 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026716
30.00%	0.026716
40.00%	0.026716
50.00%	0.026716
60.00%	0.026716
70.00%	0.026716
80.00%	0.026716
90.00%	0.026716
100.00%	0.026716

Table 4.75 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026716
30.00%	0.026716
40.00%	0.026716
50.00%	0.026716
60.00%	0.026716
70.00%	0.026716
80.00%	0.026716
90.00%	0.026716
100.00%	0.026716

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.76 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.77 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.76 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026716
30.00%	0.026716
40.00%	0.026716
50.00%	0.026716
60.00%	0.026716
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

Table 4.77 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026716
30.00%	0.026716
40.00%	0.026716
50.00%	0.026716
60.00%	0.026716
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

2.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generate by Pareto distribution. We specify the Intended traffic transport layer protocol is both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.78 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.79 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.23

Table 4.78 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026718
30.00%	0.02672
40.00%	0.026726
50.00%	0.027047
60.00%	0.027714
70.00%	0.027864
80.00%	0.02785
90.00%	0.02787
100.00%	0.027953

Table 4.79 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026716
20.00%	0.026718
30.00%	0.02672
40.00%	0.026726
50.00%	0.027114
60.00%	0.037849
70.00%	0.030479
80.00%	0.030204
90.00%	0.037901
100.00%	0.034567

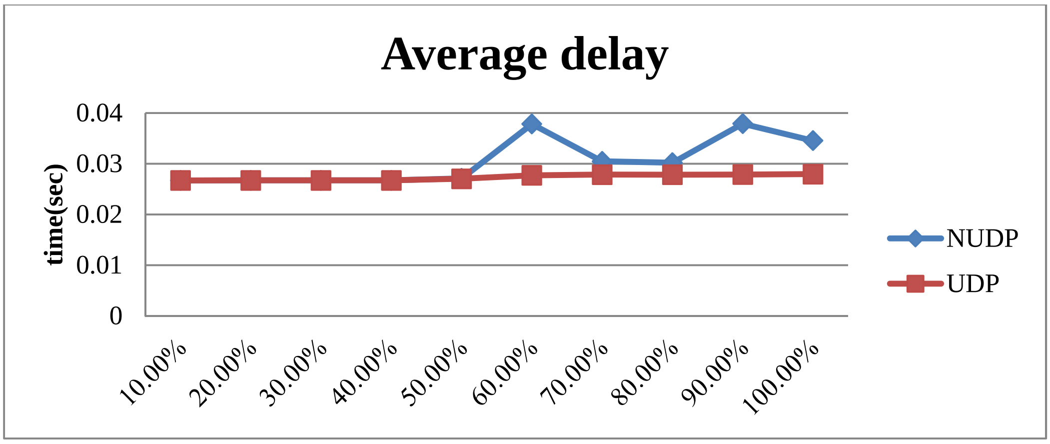

Figure 4.23 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.80 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.81 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.24

Table 4.80 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026716
30.00%	0.026718
40.00%	0.02672
50.00%	0.026714
60.00%	0.027596
70.00%	0.027633
80.00%	0.027635
90.00%	0.027726
100.00%	0.027614

Table 4.81 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026714
20.00%	0.026716
30.00%	0.026718
40.00%	0.02672
50.00%	0.026714
60.00%	0.028823
70.00%	0.02955
80.00%	0.041473
90.00%	0.042398
100.00%	0.028307

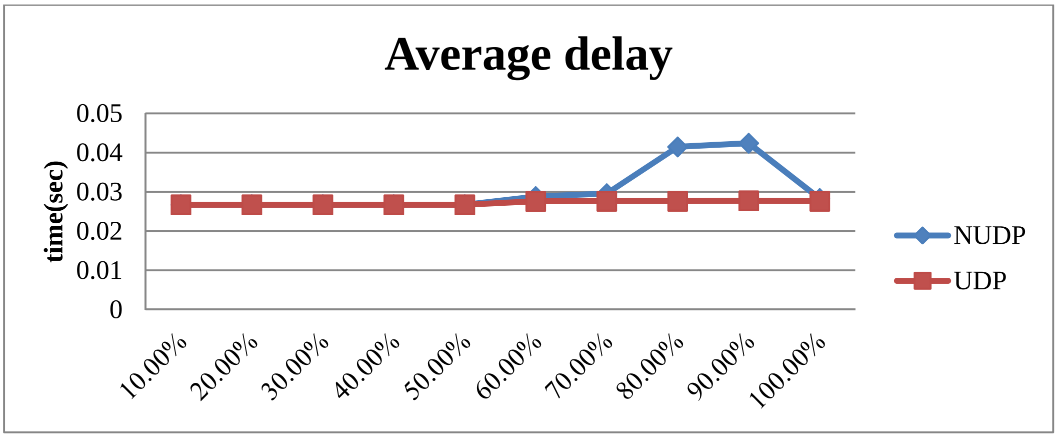

Figure 4.24 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.82 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.83 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.82 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026715
30.00%	0.026715
40.00%	0.026715
50.00%	0.026715
60.00%	0.026715
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

Table 4.83 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026715
30.00%	0.026715
40.00%	0.026715
50.00%	0.026715
60.00%	0.026715
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the

Background traffic is TCP. Table 4.84 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.85 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.84 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026715
30.00%	0.026715
40.00%	0.026715
50.00%	0.026715
60.00%	0.026715
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

Table 4.85 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026715
20.00%	0.026715
30.00%	0.026715
40.00%	0.026715
50.00%	0.026715
60.00%	0.026715
70.00%	0.026715
80.00%	0.026715
90.00%	0.026715
100.00%	0.026715

3) Intended traffic transmission rate is 75% of network bandwidth

In this part we specify transmission rate of the Intended traffic into 75% of the network bandwidth that means the transmission rate of the Intended traffic is 466.5 Mbps. The result is divided into two parts depending on the Intended traffic generator including CBR and Pareto distribution.

3.1) Intended traffic generator is CBR

In this part, we specify the Intended traffic generator is CBR that means the Intended traffic generate by CBR. We specify the transport layer protocol of the Intended traffic both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.86 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.87 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.25

Table 4.86 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.02674
30.00%	0.027626
40.00%	0.027743
50.00%	0.027875
60.00%	0.027757
70.00%	0.027821
80.00%	0.027875
90.00%	0.027882
100.00%	0.027945

Table 4.87 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.02674
30.00%	0.029837
40.00%	0.032028
50.00%	0.033095
60.00%	0.035072
70.00%	0.034693
80.00%	0.038997
90.00%	0.033838
100.00%	0.030877

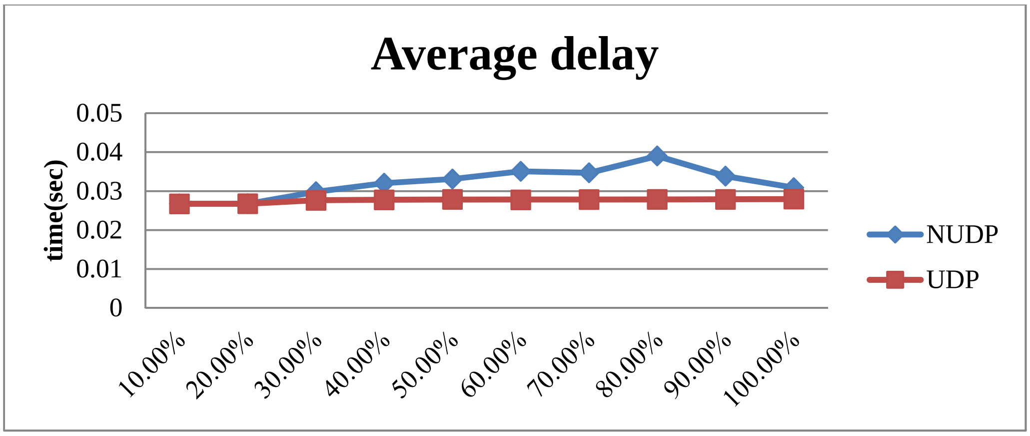

Figure 4.25 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.88 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.89 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.26

Table 4.88 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026729
30.00%	0.02758
40.00%	0.027624
50.00%	0.027627
60.00%	0.027692
70.00%	0.027665
80.00%	0.02763
90.00%	0.027686
100.00%	0.027343

Table 4.89 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026729
30.00%	0.029006
40.00%	0.034173
50.00%	0.038339
60.00%	0.044015
70.00%	0.038372
80.00%	0.034006
90.00%	0.042219
100.00%	0.033096

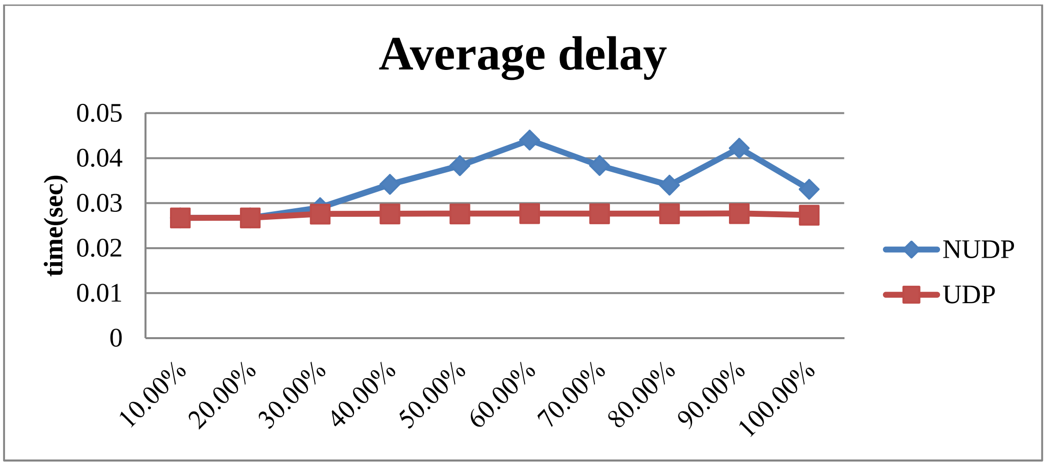

Figure 4.26 Average delays with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.90 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.91 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.90 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026724
20.00%	0.026724
30.00%	0.026724
40.00%	0.026724
50.00%	0.026724
60.00%	0.026724
70.00%	0.026724
80.00%	0.026724
90.00%	0.026724
100.00%	0.026724

Table 4.91 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026724
20.00%	0.026724
30.00%	0.026724
40.00%	0.026724
50.00%	0.026724
60.00%	0.026724
70.00%	0.026724
80.00%	0.026724
90.00%	0.026724
100.00%	0.026724

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the

Background traffic is TCP. Table 4.92 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.93 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.92 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026723
20.00%	0.026723
30.00%	0.026722
40.00%	0.026723
50.00%	0.026723
60.00%	0.026723
70.00%	0.026723
80.00%	0.026723
90.00%	0.026723
100.00%	0.026723

Table 4.93 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026723
20.00%	0.026723
30.00%	0.026722
40.00%	0.026723
50.00%	0.026723
60.00%	0.026723
70.00%	0.026723
80.00%	0.026723
90.00%	0.026723
100.00%	0.026723

3.2) Intended traffic generator is Pareto distribution

In this part, we specify the Intended traffic generator is Pareto distribution that means the Intended traffic generate by Pareto distribution. We specify the transport layer protocol of the Intended traffic both UDP and NUDP. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.94 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.95 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.27

Table 4.94 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026719
20.00%	0.02673
30.00%	0.027777
40.00%	0.027849
50.00%	0.027867
60.00%	0.027795
70.00%	0.027817
80.00%	0.027872
90.00%	0.027886
100.00%	0.027951

Table 4.95 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026719
20.00%	0.02673
30.00%	0.03027
40.00%	0.031403
50.00%	0.035036
60.00%	0.042308
70.00%	0.032991
80.00%	0.03323
90.00%	0.037607
100.00%	0.036014

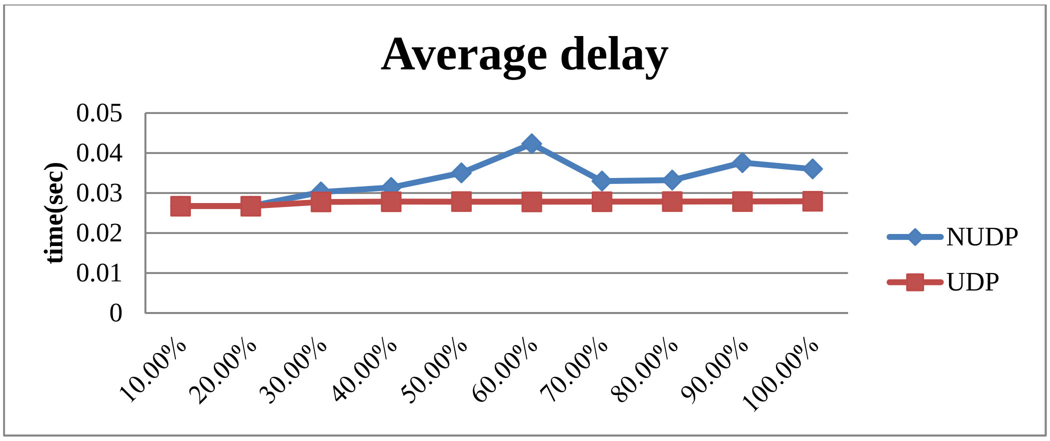

Figure 4.27 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.96 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.97 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.28

Table 4.96 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026717
20.00%	0.026719
30.00%	0.027563
40.00%	0.027572
50.00%	0.027595
60.00%	0.027603
70.00%	0.027631
80.00%	0.027638
90.00%	0.027697
100.00%	0.027626

Table 4.97 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026717
20.00%	0.026719
30.00%	0.027563
40.00%	0.030947
50.00%	0.042953
60.00%	0.03989
70.00%	0.033255
80.00%	0.036854
90.00%	0.041074
100.00%	0.032624

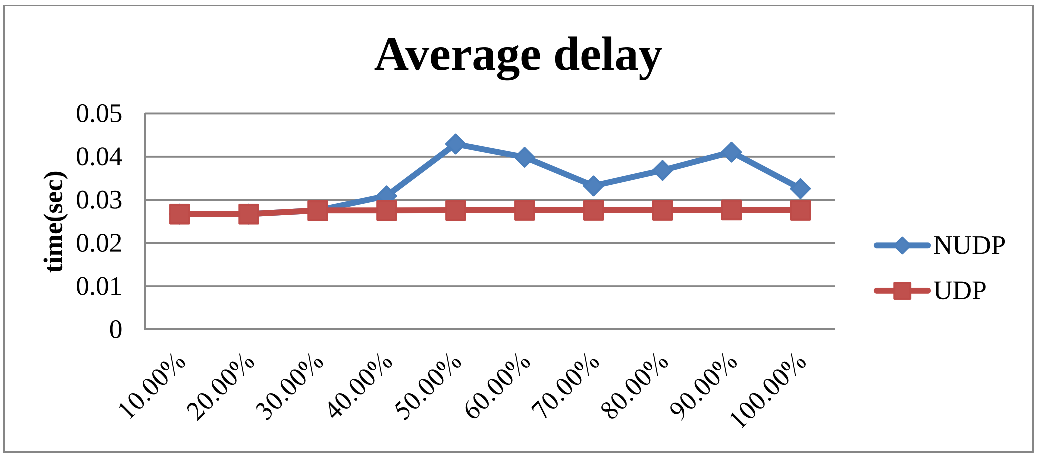

Figure 4.28 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.98 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.99 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.98 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026722
30.00%	0.026722
40.00%	0.026722
50.00%	0.026722
60.00%	0.026722
70.00%	0.026722
80.00%	0.026722
90.00%	0.026722
100.00%	0.026722

Table 4.99 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026722
30.00%	0.026722
40.00%	0.026722
50.00%	0.026722
60.00%	0.026722
70.00%	0.026722
80.00%	0.026722
90.00%	0.026722
100.00%	0.026722

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the

Background traffic is TCP. Table 4.100 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.101 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.100 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026722
30.00%	0.026722
40.00%	0.026722
50.00%	0.026722
60.00%	0.026722
70.00%	0.026722
80.00%	0.026722
90.00%	0.026722
100.00%	0.026722

Table 4.101 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026722
20.00%	0.026722
30.00%	0.026722
40.00%	0.026722
50.00%	0.026722
60.00%	0.026722
70.00%	0.026722
80.00%	0.026722
90.00%	0.026722
100.00%	0.026722

4.6 Simulation case two

We showed situation when there had one user watch video online on website such as youtube for comparing between UDP and NUDP. In this case, we have to convert sample video file to trace file by using tool set called Evalvid. We used trace file from Evalvid at NS-2 for generating traffic in simulation. The sample video for using in this case has detail below:

1. Video length = 6.39 minutes.
2. Video size = 640 × 480 pixels.
3. Frame rate = 24 frame/sec.

4.6.1 Result of the simulation

We divided result of the simulation into two parts including packet lost rate and average delay which showed in below:

4.6.1.1 Packet lost rate result

This part shows the result of the simulation in the part of the packet lost rate. We specify both UDP and NUDP are the Intended traffic transport layer protocol. The result of this part is divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the packet lost rate of Intended traffic when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.102 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.103 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.29

Table 4.102 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10.00%	0.252901
20.00%	0.258412
30.00%	0.261474
40.00%	0.270659
50.00%	0.462937
60.00%	0.860966
70.00%	1.540063
80.00%	2.747007
90.00%	4.81063
100.00%	44.99005

Table 4.103 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10.00%	0.252901	0
20.00%	0.258412	0
30.00%	0.261474	0
40.00%	0.270659	0
50.00%	0.462937	0
60.00%	0.860966	0.001225
70.00%	1.540063	0.001225
80.00%	2.747007	0.004286
90.00%	4.81063	0.025106
100.00%	44.99005	20.90016

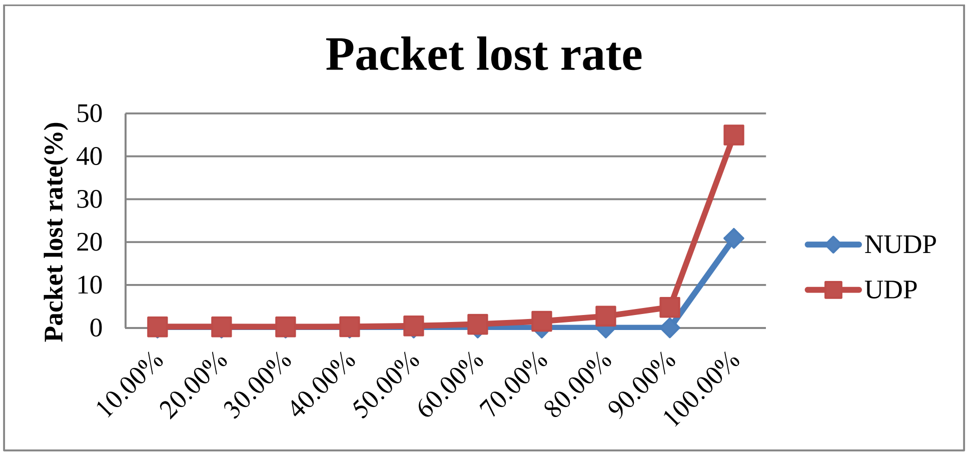

Figure 4.29 Packet lost rate with different protocols

From Figure 4.29, the packet lost rate results both UDP and NUDP increased between Background traffic transmission rate 90% to 100%. For this reason, we focused on Background traffic transmission rate 90% to 100%. Table 4.104 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.105 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.30

Table 4.104 UDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate (%)
90.00%	4.81063
91.00%	5.131502
92.00%	5.422982
93.00%	5.805701
94.00%	6.225162
95.00%	6.651358
96.00%	7.324944
97.00%	8.310217
98.00%	9.889471
99.00%	14.63029
100.00%	44.99005

Table 4.105 NUDP Intended traffic packet lost rate over CBR and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
90.00%	4.81063	0.025106
91.00%	5.131502	0.025719
92.00%	5.422982	0.026943
93.00%	5.805701	0.034904
94.00%	6.225162	0.03919
95.00%	6.651358	0.058786
96.00%	7.324944	0.116959
97.00%	8.310217	0.152475
98.00%	9.889471	0.262699
99.00%	14.63029	0.64603
100.00%	44.99005	20.90016

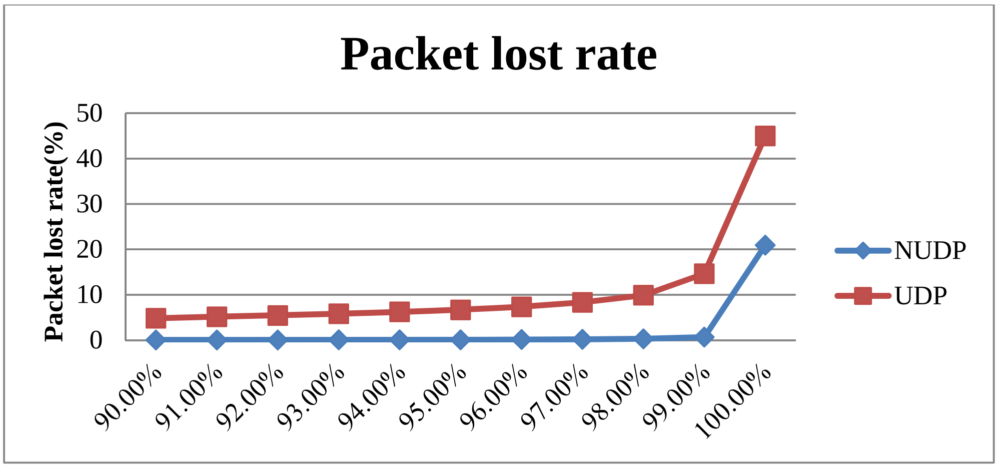

Figure 4.30 Packet lost rate with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the packet lost rate of Intended traffic when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.106 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.107 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.31

Table 4.106 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
10.00%	0.249839
20.00%	0.252289
30.00%	0.258412
40.00%	0.2578
50.00%	0.40109
60.00%	0.693794
70.00%	1.137748
80.00%	1.868283
90.00%	2.787422
100.00%	47.12103

Table 4.107 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10.00%	0.249839	0
20.00%	0.252289	0
30.00%	0.258412	0
40.00%	0.2578	0
50.00%	0.40109	0.001225
60.00%	0.693794	0.000612
70.00%	1.137748	0.000612
80.00%	1.868283	0.002449
90.00%	2.787422	0.003674
100.00%	47.12103	37.53774

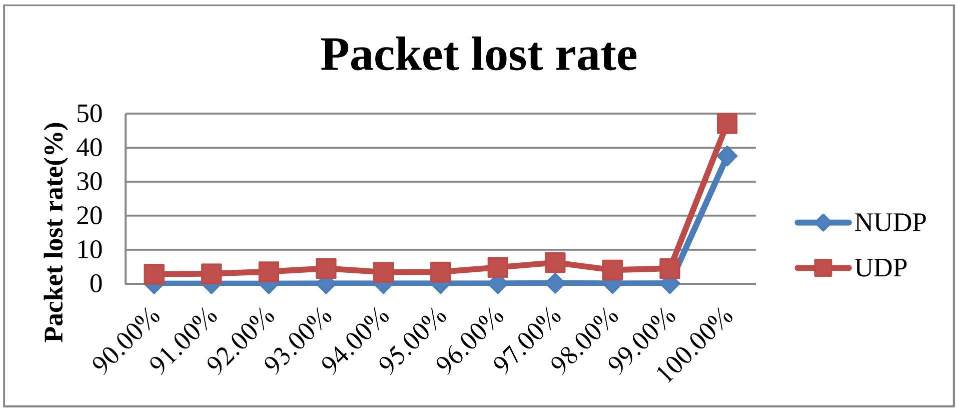

Figure 4.31 Packet lost rate with different protocols

From Figure 4.31, the packet lost rate results both UDP and NUDP increased between Background traffic transmission rate 90% to 100%. For this reason, we focused on Background traffic transmission rate 90% to 100%. Table 4.108 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.109 showed packet lost rate when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.32

Table 4.108 UDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate (%)
90.00%	2.787422
91.00%	2.910505
92.00%	3.522244
93.00%	4.473837
94.00%	3.380178
95.00%	3.412633
96.00%	4.817366
97.00%	6.231285
98.00%	4.006613
99.00%	4.445669
100.00%	47.12103

Table 4.109 NUDP Intended traffic packet lost rate over Pareto distribution and UDP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
90.00%	2.787422	0.003674
91.00%	2.910505	0.012859
92.00%	3.522244	0.009798
93.00%	4.473837	0.031842
94.00%	3.380178	0.042252
95.00%	3.412633	0.042865
96.00%	4.817366	0.061235
97.00%	6.231285	0.169009
98.00%	4.006613	0.057561
99.00%	4.445669	0.077156
100.00%	47.12103	37.53774

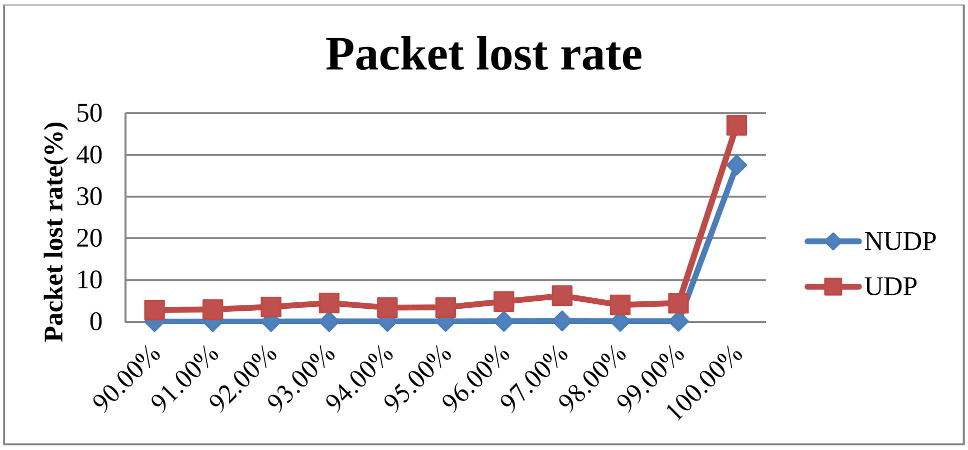

Figure 4.32 Packet lost rate with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the packet lost rate of Intended traffic when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.110 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.111 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.110 UDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10.00%	0.249839
20.00%	0.249839
30.00%	0.249839
40.00%	0.249839
50.00%	0.249839
60.00%	0.249839
70.00%	0.249839
80.00%	0.249839
90.00%	0.249839
100.00%	0.249839

Table 4.111 NUDP Intended traffic packet lost rate over FTP and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10.00%	0.249839	0
20.00%	0.249839	0
30.00%	0.249839	0
40.00%	0.249839	0
50.00%	0.249839	0
60.00%	0.249839	0
70.00%	0.249839	0
80.00%	0.249839	0
90.00%	0.249839	0
100.00%	0.249839	0

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the packet lost rate of Intended traffic when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.112 showed packet lost rate when transport layer protocol of Intended traffic is UDP and Table 4.113 showed packet lost rate when transport layer protocol of Intended traffic is NUDP.

Table 4.112 UDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate (%)
10.00%	0.249227
20.00%	0.249227
30.00%	0.249227
40.00%	0.249227
50.00%	0.249227
60.00%	0.249227
70.00%	0.249227
80.00%	0.249227
90.00%	0.249227
100.00%	0.249227

Table 4.113 NUDP Intended traffic packet lost rate over Pareto distribution and TCP Background traffic

Background transmission rate	Packet loss rate before retransmit (%)	Packet loss rate after retransmit (%)
10.00%	0.249227	0
20.00%	0.249227	0
30.00%	0.249227	0
40.00%	0.249227	0
50.00%	0.249227	0
60.00%	0.249227	0
70.00%	0.249227	0
80.00%	0.249227	0
90.00%	0.249227	0
100.00%	0.249227	0

4.6.1.2 Average delay per packet result

This part showed the simulation result in the part of the average delay. In this part we calculate the average delay from the 10000 packets that receiver can receive it. The results of this part are divided into four parts depending on type of the Background traffic generator and Background traffic transport layer protocol including:

A. Background traffic generator is CBR, Transport layer protocol of Background traffic is UDP.

B. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is UDP.

C. Background traffic generator is FTP, Transport layer protocol of Background traffic is TCP.

D. Background traffic generator is Pareto distribution, Transport layer protocol of Background traffic is TCP.

A. Background traffic generator is CBR, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is CBR and transport layer protocol of the Background traffic is UDP. Table 4.114 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.115 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.33

Table 4.114 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026819
20.00%	0.026833
30.00%	0.026847
40.00%	0.026864
50.00%	0.026883
60.00%	0.026902
70.00%	0.026923
80.00%	0.026945
90.00%	0.026975
100.00%	0.027497

Table 4.115 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026819
20.00%	0.026833
30.00%	0.026847
40.00%	0.026864
50.00%	0.026883
60.00%	0.027058
70.00%	0.027272
80.00%	0.027896
90.00%	0.028671
100.00%	0.081434

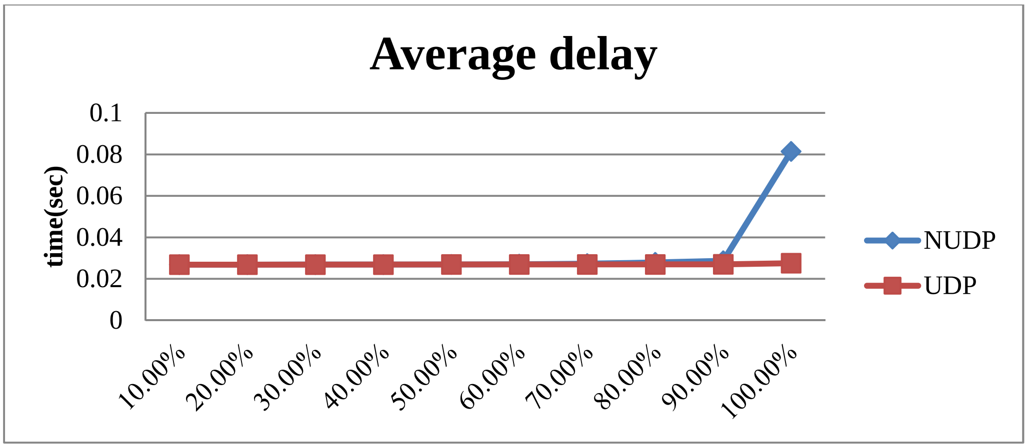

Figure 4.33 Average delay with different protocols

From Figure 4.33, the average delay results both UDP and NUDP increased between Background traffic transmission rate 90% to 100%. For this reason, we focused on Background traffic transmission rate 90% to 100%. Table 4.116 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.117 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.34

Table 4.116 UDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
90.00%	0.026975
91.00%	0.026979
92.00%	0.026985
93.00%	0.026992
94.00%	0.026998
95.00%	0.027008
96.00%	0.027019
97.00%	0.02704
98.00%	0.027066
99.00%	0.02714
100.00%	0.027497

Table 4.117 NUDP Intended traffic average delay over CBR and UDP Background traffic

Background traffic	Average delay(sec)
90.00%	0.028671
91.00%	0.028614
92.00%	0.029057
93.00%	0.028791
94.00%	0.02943
95.00%	0.029739
96.00%	0.029752
97.00%	0.030354
98.00%	0.032006
99.00%	0.036755
100.00%	0.081434

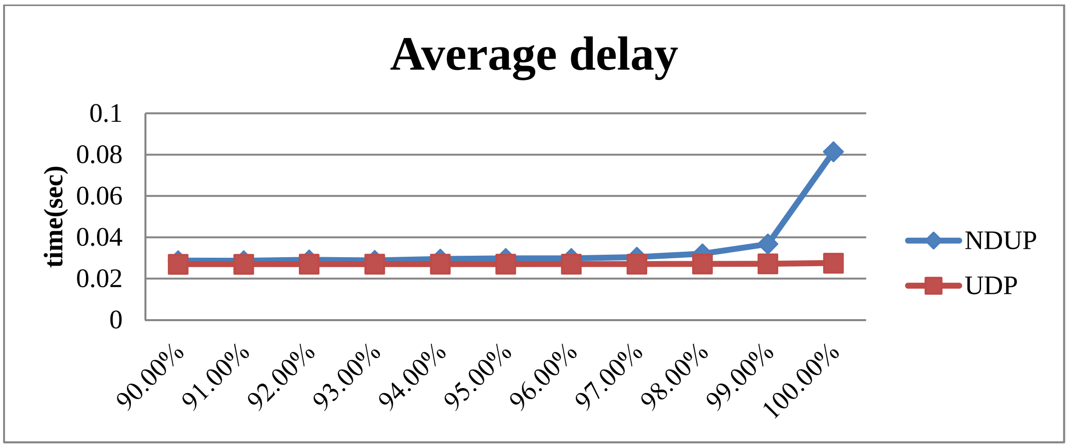

Figure 4.34 Average delay with different protocols

B. Background traffic generator is Pareto distribution, Transport layer protocol is UDP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is UDP. Table 4.118 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.119 showed average delay when

transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.35

Table 4.118 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026814
20.00%	0.026824
30.00%	0.026834
40.00%	0.026845
50.00%	0.026857
60.00%	0.026868
70.00%	0.02688
80.00%	0.026891
90.00%	0.026904
100.00%	0.02701

Table 4.119 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026814
20.00%	0.026824
30.00%	0.026834
40.00%	0.026845
50.00%	0.026857
60.00%	0.027001
70.00%	0.027209
80.00%	0.027496
90.00%	0.027893
100.00%	0.200282

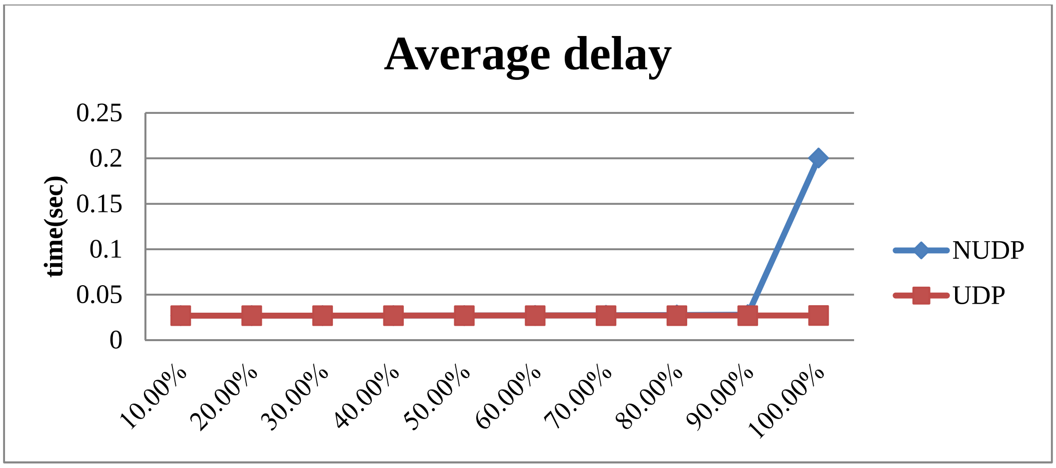

Figure 4.35 Average delay with different protocols

From Figure 4.35, the average delay results both UDP and NUDP increased between Background traffic transmission rate 90% to 100%. For this reason, we focused on Background traffic transmission rate 90% to 100%. Table 4.120 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.121 showed average delay when transport layer protocol of Intended traffic is NUDP. These results are also showed graphically in Figure 4.36

Table 4.120 UDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
90.00%	0.026904
91.00%	0.026905
92.00%	0.026906
93.00%	0.026907
94.00%	0.026909
95.00%	0.026915
96.00%	0.026914
97.00%	0.026911
98.00%	0.026917
99.00%	0.026921
100.00%	0.02701

Table 4.121 NUDP Intended traffic average delay over Pareto distribution and UDP Background traffic

Background traffic	Average delay(sec)
90.00%	0.027893
91.00%	0.028018
92.00%	0.028527
93.00%	0.028699
94.00%	0.029317
95.00%	0.027912
96.00%	0.02859
97.00%	0.031611
98.00%	0.030129
99.00%	0.028903
100.00%	0.200282

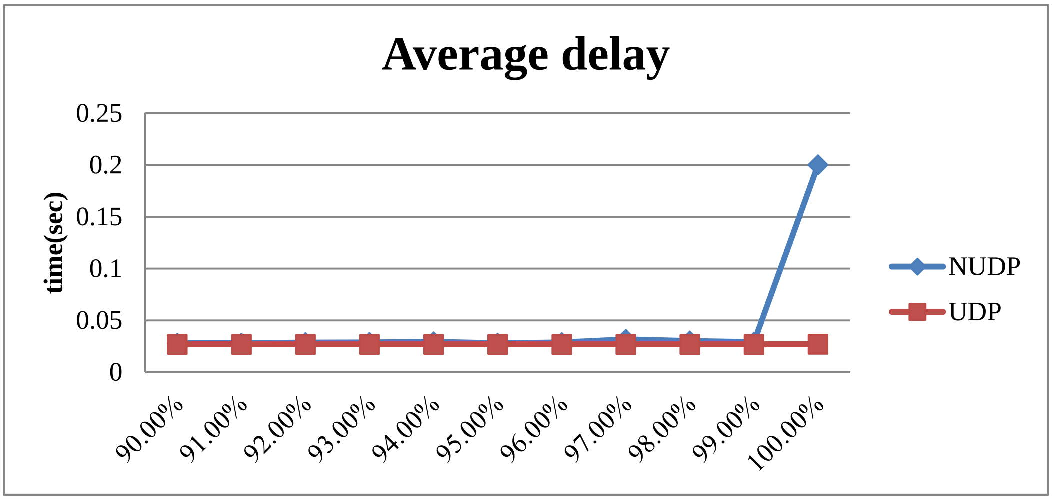

Figure 4.36 Average delay with different protocols

C. Background traffic generator is FTP, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is FTP and transport layer protocol of the Background traffic is TCP. Table 4.122 showed average delay when transport layer protocol of

Intended traffic is UDP and Table 4.123 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.122 UDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026808
20.00%	0.026808
30.00%	0.026808
40.00%	0.026808
50.00%	0.026808
60.00%	0.026808
70.00%	0.026808
80.00%	0.026808
90.00%	0.026808
100.00%	0.026808

Table 4.123 NUDP Intended traffic average delay over FTP and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026808
20.00%	0.026808
30.00%	0.026808
40.00%	0.026808
50.00%	0.026808
60.00%	0.026808
70.00%	0.026808
80.00%	0.026808
90.00%	0.026808
100.00%	0.026808

D. Background traffic generator is Pareto distribution, Transport layer protocol is TCP

This part is the Intended traffic average delay when Background traffic generator is Pareto distribution and transport layer protocol of the Background traffic is TCP. Table 4.124 showed average delay when transport layer protocol of Intended traffic is UDP and Table 4.125 showed average delay when transport layer protocol of Intended traffic is NUDP.

Table 4.124 UDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026809
20.00%	0.026808
30.00%	0.026808
40.00%	0.026808
50.00%	0.026808
60.00%	0.026808
70.00%	0.026808
80.00%	0.026808
90.00%	0.026808
100.00%	0.026808

Table 4.125 NUDP Intended traffic average delay over Pareto distribution and TCP Background traffic

Background traffic	Average delay(sec)
10.00%	0.026809
20.00%	0.026808
30.00%	0.026808
40.00%	0.026808
50.00%	0.026808
60.00%	0.026808
70.00%	0.026808
80.00%	0.026808
90.00%	0.026808
100.00%	0.026808

CHAPTER V
DISCUSSION AND CONCLUSION

This chapter discusses the findings of the previous chapter. According to the findings, the researchers divided the simulations into two cases. Hence, this chapter divides the discussion of the findings into two parts.

5.1 Discussion: Simulation Case One

In simulation case one, the researchers assumed many video transitions sending from the source node to the destination node. The researchers specified three intended traffic transmission rates, namely, 25%, 50% and 75%from the network bandwidth. The researchers used CBR and Pareto distribution for generating intended traffic.

The researchers have summarized the findings for when the intended traffic transmission rate is 25% of the network bandwidth. The intended traffic is generated by the CBR as shown in Table 5.1 which shows both the average packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table 5.1 Summary of the findings when the Intended traffic rate is 25% and CBR is used for generating

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	Transport layer	UDP	NUDP	UDP	NUDP
CBR	UDP	47.5911	36.6058	0.0270711	0.029813
Pareto	UDP	33.1079	20.7444	0.0270371	0.0298338
FTP	TCP	0	0	0.026714	0.026714
Pareto	TCP	0	0	0.0267143	0.0267143

According to Table 5.1, when the intended traffic transmission rate is 25% of the network bandwidth, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP because NUDP has a retransmission mechanism to compensate for packet loss but UDP has no mechanism for such compensation. On the other hand, the packet loss rate of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm. This algorithm is a technique aimed at avoiding network congestion. In this simulation, there was no packet loss as long as there was no network congestion. Hence, the researchers conclude that NUDP has a better average packet loss rate than UDP by 11.67%when the intended traffic transmission rate is 25% and generated by CBR.

Concerning average delay, UDP has better average delay than NUDP when the transport layer protocol of the background traffic is UDP because UDP is the fastest protocol and NUDP uses the retransmission method to compensate for packet loss increasing the overall average delay. On the other hand, the average delay of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm which means this simulation has no network congestion. In this simulation, the packet loss does not occur during transmission of packets when there is no network congestion. NUDP does not use the retransmission method when there is no packet loss and that is why the average delay of NUDP and UDP are equal. Hence, the researchers concluded that NUDP can transmit packet slower than UDP by 10.24%when the intended traffic transmission rate is 25% and generated by CBR

After the researchers summarized the findings when the intended traffic generator was CBR, the researchers summarized the results when the intended traffic generator was the Pareto distribution in Table 5.2 which shows both the average packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table 5.2 Summary of the findings when the Intended traffic rate is 25% and the Pareto distribution is used for Generating

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	Transport layer	UDP	NUDP	UDP	NUDP
CBR	UDP	45.8062	35.2913	0.0270653	0.0296667
Pareto	UDP	32.4997	23.533	0.0270028	0.0290225
FTP	TCP	0	0	0.026714	0.026714
Pareto	TCP	0	0	0.0267142	0.0267142

According to Table 5.2, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP, even though the researchers changed the intended traffic generator from CBR to Pareto distribution. The packet loss rate of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP as in the situation when CBR is the intended traffic generator. Hence, the researchers concluded that NUDP has a better average packet loss rate than UDP by 9.74%when the intended traffic transmission rate is 25% and generated by Pareto distribution.

Concerning average delay, UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP and equal when the transport layer protocol of the background traffic is TCP. This finding resembles the situation when the intended traffic generator is CBR. Hence, the researchers have concluded that NUDP can transmit packets more slowly than UDP by 8.55%when the intended traffic transmission rate is 25% and generated by Pareto distribution.

Looking at the packet loss rate in Table 4.6, Table 4.7, Table 4.8, Table 4.9, Table 4.14, Table 4.15, Table 4.16, Table 4.17, Figure 4.5, Figure 4.6, Figure 4.7, and Figure 4.8, the researchers realized that packet losses occurred when the background transmission rate was 80%. In other words, packet loss occurs when all of the network traffic exceeds the network bandwidth because the intended traffic transmission rate in this situation is 25% of the network bandwidth and the background traffic transmission rate is 80%, thereby making size of the network traffic105%.

The researchers concluded in the situation where the intended traffic transmission rate is 25% of the network bandwidth that NUDP has a better packet loss rate better than UDP when the transport layer protocol of the background traffic is UDP. In the same situation, however, UDP has an average delay that is less than NUDP, thereby meaning that a mechanism to compensate for the packet loss of NUDP affects transmission delay. The researchers visualize a situation where the transport layer protocol of the background traffic is TCP and there is no packet loss whether NUDP or UDP is used for the transport layer protocol of the intended traffic. In this situation, the average delay of both protocols is equal because NUDP does not use a mechanism to compensate for packet loss. NUDP has a better packet loss rate than UDP by 10.71% but is capable of transmitting packets more slowly than UDP by 9.39%. When UDP is the transport layer protocol of the Background traffic, packet loss occurs when the size of the network traffic more than the size of the network bandwidth. However, this situation does not occur when TCP is the transport layer protocol of the background traffic because TCP has a congestion avoidance algorithm with the effect of no congestion occurring in the network.

After the researchers had summarized the findings when the intended traffic transmission rate is 25% of the network bandwidth, the researchers summarized the findings when the intended traffic transmission rate is 50% of the network bandwidth and the intended traffic is generated by CBR in Table 5.3 which shows both the packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table 5.3 Summary of the findings when the Intended traffic rate is 50% and CBR is used for generating.

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	Transport layer	UDP	NUDP	UDP	NUDP
CBR	UDP	42.104	34.3667	0.0273571	0.0308135
Pareto	UDP	45.6243	35.813	0.0271996	0.0320977
FTP	TCP	0	0	0.026716	0.026716
Pareto	TCP	0	0	0.0267156	0.0267156

According to Table 5.3, when the intended traffic transmission rate is 50% of the network bandwidth, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP because NUDP has a retransmission mechanism to compensate for packet loss but UDP has no such mechanism to compensate for packet loss. On the other hand, the packet loss rates for NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm. This algorithm is a technique aimed at avoiding network congestion. In this simulation, there is no packet loss when there is no network congestion. Hence, the researchers have concluded that NUDP has a better average packet loss rate than UDP by8.77%when the intended traffic transmission rate is 50% and generated by CBR.

Concerning the average delay, UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP because UDP is the fastest protocol and NUDP uses the retransmission method to compensate for packet loss increasing the overall average delay. On the other hand, the average delays of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm meaning this simulation has no network congestion. In this simulation, the packet loss does not occur during packet transmission when there is no network congestion. NUDP does not use the retransmission method when there is no packet loss and that is why the average delays of NUDP and UDP are equal. Hence, the researchers have concluded that NUDP can transmit packets more slowly than UDP by 15.32% when the intended traffic transmission rate is 50% and generated by CBR.

After the researchers had summarized the results when the intended traffic generator is CBR, the researchers summarized the findings when the intended traffic transmission rate is 50% of the network bandwidth and the intended traffic is generated by Pareto distribution as in Table 5.4 which shows both packet loss rate and average delay when the intended traffic is UDP and NUDP.

Table 5.4 Summary of the findings when the Intended traffic rate is 50% and Pareto distribution is used for generating

Background traffic		**Intended traffic**			
		Packet loss rate (%)		**Average delay (sec)**	
Traffic generator	**Transport layer**	**UDP**	**NUDP**	**UDP**	**NUDP**
CBR	UDP	31.9608	21.7829	0.0273178	0.0304994
Pareto	UDP	41.6323	29.7584	0.0271786	0.0304133
FTP	TCP	0	0	0.026715	0.026715
Pareto	TCP	0	0	0.026715	0.026715

According to Table 5.4, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP, even though the researchers changed the intended traffic generator from CBR to Pareto distribution. The packet loss rates of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP as in the situation when CBR is the intended traffic generator. Hence, the researchers have concluded that NUDP has a better average packet loss rate than UDP by11.03%when the intended traffic transmission rate is 50% and generated by Pareto distribution.

Concerning the average delay, UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP and equal when the transport layer protocol of the background traffic is TCP. The aforementioned findings resemble the situation where the intended traffic generator is CBR. Hence, the researchers have concluded that NUDP can transmit packet more slowly than UDP by

11.77%when the intended traffic transmission rate is 50% and generated by Pareto distribution.

Looking at the packet loss rate in Table 4.22, Table 4.23, Table 4.24, Table 4.25, Table 4.30, Table 4.31, Table 4.32, Table 4.33, Figure 4.9, Figure 4.10, Figure 4.11, and Figure 4.12 the researchers realized packet losses when the background transmission rate was 50%, thereby meaning that packet loss occurred when all of the network traffic exceeded the network bandwidth because the intended traffic transmission rate in this situation was 50% of the network bandwidth and the background traffic transmission rate was 50%; hence, the size of the network traffic was 100%.

The researchers concluded in the situation where the intended traffic transmission rate is 50% of the network bandwidth that NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP. In the same situation, however, UDP had a lower average delay than NUDP. Therefore, the NUDP mechanism compensating for packet loss affected transmission delay. The researchers were able to visualize the situation where the transport layer protocol of the background traffic is TCP and no packet loss occurred whether NUDP or UDP was used for the transport layer protocol of the intended traffic. In this situation, the average delays of both protocols were equal because NUDP does not use a mechanism to compensate for packet loss. NUDP has a better packet loss rate than UDP by 9.9% but transmits packet more slowly than UDP by 13.55%. When UDP is the transport layer protocol of the background traffic, packet loss occurs when the size of the network traffic exceeds the size of the network bandwidth. However, this situation does not occur when TCP is the transport layer protocol of the background traffic because TCP has a congestion avoidance algorithm with the effect of no congestion occurring in the network.

After the researchers had summarized the findings where the intended traffic transmission rate was 25% and 50% of the network bandwidth, the researchers were eventually able to summarize the results when the intended traffic transmission rate was 75% of the network bandwidth and the intended traffic was generated by CBR in Table 5.5 which shows both the packet loss rate and average delay when the intended traffic is UDP and NUDP.

Table 5.5 Summary the findings when the Intended traffic rate is 75% and CBR is used for generating

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	**Transport layer**	**UDP**	**NUDP**	**UDP**	**NUDP**
CBR	UDP	54.0324	46.7538	0.0275986	0.0321899
Pareto	UDP	53.761	43.4085	0.0274298	0.0346677
FTP	TCP	0	0	0.026724	0.026724
Pareto	TCP	0	0	0.0267229	0.0267229

According to Table 5.5, when the intended traffic transmission rate is 75% of the network bandwidth, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP because NUDP has a retransmission mechanism to compensate for packet loss but UDP has no such mechanism to compensate for packet loss. On the other hand, the packet loss rate of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm. This algorithm is a technique aimed at avoiding network congestion. In this simulation, there is no packet loss when there is no network congestion. Hence, the researchers have concluded that NUDP has a better average packet loss rate than UDP by8.82%when the intended traffic transmission rate is 75% and generated by CBR.

Concerning average delay, UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP because UDP is the fastest protocol and NUDP uses the retransmission method to compensate for packet loss increasing overall average delay. On the other hand, the average delays of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm. Thus, the simulation has no occurrence of network congestion. In this simulation, packet loss does not occur during packet transmission when there is no network congestion. NUDP does not use the retransmission method when there is no packet loss this is the reason why the average delays for NUDP and UDP are equal. Hence, the researchers have concluded that NUDP can transmit packets more slowly than UDP by 21.51%when the intended traffic transmission rate is 75% and generated by CBR.

After the researchers had summarized the results when the intended traffic generator is CBR, the researchers summarized the results when the intended traffic transmission rate was 75% of the network bandwidth and the intended traffic was generated by Pareto distribution as in Table 5.6 which shows both the packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table 5.6 Summary of the findings when the Intended traffic rate is 75% and Pareto distribution is used for generating

Background traffic		**Intended traffic**			
		Packet lost rate (%)		**Average delay (sec)**	
Traffic generator	**Transport layer**	**UDP**	**NUDP**	**UDP**	**NUDP**
CBR	UDP	42.4455	32.0701	0.0276263	0.0332308
Pareto	UDP	39.9378	29.0737	0.0274361	0.0338596
FTP	TCP	0	0	0.026722	0.026722
Pareto	TCP	0	0	0.026722	0.026722

According to Table 5.6, NUDP has a better packet loss rate than UDP when the transport layer protocol of the background traffic is UDP, even when the researchers changed the intended traffic generator from CBR to Pareto distribution. The packet loss rates of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP as in the situation when CBR is the intended traffic generator. Hence, the researchers concluded that NUDP has a better average packet loss rate than UDP for 10.62%when the intended traffic transmission rate is 75% and generated by Pareto distribution.

UDP has an average delay better than NUDP when the transport layer protocol of the background traffic is UDP and equal when the transport layer protocol of the background traffic is TCP. The aforementioned finding resembles the situation when the intended traffic generator is CBR. Hence, the researchers concluded that NUDP can transmit packets more slowly than UDP by21.85%when the intended traffic transmission rate is 75% and generated by Pareto distribution,

Viewing the packet loss rate in Table 4.38, Table 4.39, Table 4.40, Table 4.41, Table 4.46, Table 4.47, Table 4.48, Table 4.49, Figure 4.13, Figure 4.14, Figure 4.15, and Figure 4.16 the researchers noticed that packet losses occurred when the Background transmission rate was 30%. Thus, the packet loss occurred when all of the network traffic exceeded the network bandwidth because the intended traffic transmission rate in this situation was 75% of the network bandwidth and the background traffic transmission rate was 30%; thus, the size of the network traffic is 105%.

The researchers summarized the situation when the intended traffic transmission rate was 75% of the network bandwidth, finding NUDP to have a better packet loss rate than UDP when the transport layer protocol of the background traffic was UDP. In the same situation, however, UDP has an average delay less than NUDP. That means the NUDP mechanism compensating for packet loss affected the transmission delay. Hence, the researchers observed no packet loss in the situation where the transport layer protocol for the background traffic was TCP, even though NUDP or UDP were used for the transport layer protocol of the intended traffic. In this situation, the average delay of both protocols were equal because NUDP does not use a mechanism to compensate for packet loss. NUDP has a better packet loss rate than UDP by 9.72% but can transmit packets more slowly than UDP by 21.68%. When UDP is the transport layer protocol of the background traffic, packet loss occurs when the size of the network traffic exceeds the size of the network bandwidth. However, this situation does not occur when TCP is the transport layer protocol of the background traffic because TCP has a congestion avoidance algorithm that causes no congestion to occur in the network.

5.2 Conclusion: Simulation Case One

This part concludes the simulation for Case One. According to the above findings, NUDP is better than UDP in terms of packet loss rate. The ideal situation for transmitting packets by using both UDP and NUDP is when the transport layer protocol of the background traffic is TCP because no packet loss occurs in the network when the transport layer protocol of the background traffic is TCP due to TCP's congestion avoidance algorithm. This algorithm is aimed at avoiding network congestion and TCP decreases the background traffic transmission rate when network congestion occurs. For example, when the intended traffic transmission rate is 25% of the network bandwidth and the transport layer protocol of the background traffic is TCP, the highest transmission rate for the background traffic is no more than 75% of the network bandwidth and that is why there is no packet loss occurring in the network when the transport layer protocol of the background traffic is TCP.

With regard to average delay, however, UDP is better than NUDP because NUDP uses the retransmission method to compensate for packet loss. The retransmission method affects transmission delay because NUDP will retransmit data when packet loss occurs in the network, but UDP has no mechanism to compensate for packet loss and that is why the average delay of NUDP is higher than UDP.

5.3 Discussion: Simulation Case Two

In simulation case two, the researchers assumed that one user was watching a video on a website for comparison between UDP and NUDP. Hence, the researchers used the sample video and converted it to generate the intended traffic.

The researchers summarized the findings on this case in Table 5.7 which shows both the average packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table 5.7 Summary of case two findings

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	Transport layer	UDP	NUDP	UDP	NUDP
CBR	UDP	5.64551	2.0932	0.0269488	0.0326577
Pareto	UDP	5.50277	3.75463	0.0268727	0.0444055
FTP	TCP	0.24984	0	0.026808	0.026808
Pareto	TCP	0.24923	0	0.0268081	0.0268081

According to Table 5.7, NUDP has a better packet loss rate than UDP in every situation because NUDP has a retransmission mechanism to compensate for packet loss but UDP has no such mechanism to compensate for packet loss. Hence, the researchers concluded that NUDP has a better average packet loss rate than UDP by 2.65% when there is one user watching a video on a website.

UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP because UDP is the fastest protocol and NUDP uses the retransmission method to compensate for packet loss which increases overall average delay. On the other hand, the average delays of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm which means this simulation has no network congestion. In this simulation, the packet loss does not occur during transmission of packets when there is no network congestion. NUDP does not use the retransmission method when there is no packet loss and that is why the average delays for NUDP and UDP are equal. Hence, the researchers concluded that NUDP can transmit packets more slowly than UDP by 43.21%when there is one user watching a video on a website.

In this case, the packet loss rate results in increases for both UDP and NUDP between the background traffic transmission rates at 90% to 100%. For this reason, the researchers focused on a background traffic transmission rate of 90% to 100%. Table 5.8 shows both the average packet loss rate and the average delay when the intended traffic is UDP and NUDP.

Table5.8 Summary of case two findings when the Background traffic transmission rate is 90% to 100%.

Background traffic		Intended traffic			
		Packet loss rate (%)		Average delay (sec)	
Traffic generator	Transport layer	UDP	NUDP	UDP	NUDP
CBR	UDP	10.8357	2.02627	0.0270635	0.0349639
Pareto	UDP	7.91898	3.45873	0.0269199	0.0445346
FTP	TCP	0.24984	0	0.026808	0.026808
Pareto	TCP	0.24923	0	0.0268081	0.0268081

According to Table 5.8, NUDP has a better packet loss rate than UDP in every situation because NUDP has a retransmission mechanism to compensate for packet loss, but UDP has no such mechanism to compensate for packet loss. Hence, the researchers concluded that NUDP has a better average packet loss rate than UDP for 6.63% when there is one user watching a video on a website.

UDP has a better average delay than NUDP when the transport layer protocol of the background traffic is UDP because UDP is the fastest protocol and NUDP uses the retransmission method to compensate for packet loss which increases overall average delay. On the other hand, the average delays of NUDP and UDP are equal when the transport layer protocol of the background traffic is TCP because TCP has a congestion avoidance algorithm which means the simulation has no network congestion. In this simulation, packet loss does not occur during packet transmission when there is no network congestion. NUDP does not use the retransmission method when there is no packet loss that is why the average delays for NUDP and UDP are equal. Hence, the researchers concluded that NUDP can transmit packets more slowly than UDP by 47.31%when there is one user watching a video on a website.

5.4 Conclusion: Simulation Case Two

This part concludes about simulation case two. According to the above findings, NUDP is better than UDP in terms of packet loss rates. The best situation for transmitting packets by using both UDP and NUDP is when the transport layer protocol of the background traffic is TCP because there is no packet loss in the network when the transport layer protocol of the Background traffic is TCP due to TCP's congestion avoidance algorithm. This algorithm is aimed at avoiding network congestion. TCP decreases the background traffic transmission rate when network congestion occurs. For example, when the intended traffic transmission rate is 25% of the network bandwidth and the transport layer protocol of the background traffic is TCP, the highest transmission rate for the background traffic exceeds 75% of the network bandwidth and that is why there is no packet loss in the network when the transport layer protocol of the background traffic is TCP.

With regard to average delay, UDP is better than NUDP because NUDP uses the retransmission method to compensate for packet loss. The retransmission method affects transmission delay because, NUDP retransmits data when packet loss occurs in the network, but UDP has no such mechanism to compensate for packet loss that is why the average delay for NUDP is higher than UDP.

5.5 Conclusion

The main objective of the present study was to improve network performance in terms of data loss. Therefore, the researchers modified UDP and added some features to improve performance, using the retransmission method to compensate for data loss. This technique can reduce data loss rate. Therefore, the researchers used a technique called “sequencer” for re-ordering and checking for duplicate data before sending to the application.

The researchers evaluated the performance of NUDP through simulations and comparative studies between NUDP and the original UDP. According to the findings, NUDP had a better packet loss rate than the original UDP in every situation. The average delay of NUDP equaled the average delay of UDP when no data loss occurred in the network. For this reason, NUDP is the most suitable in situations where the transport layer protocol of the background traffic is TCP. In the real world, however, network traffic does not use only TCP. Hence, the researchers simulated other situations when the background traffic was UDP. According to the findings, NUDP is suitable when the network is slightly congested. Thus, NUDP is suitable when the background traffic transmission rate is 25% because the packet loss rate of NUDP is better than UDP and the average delay of NUDP is only slightly higher than UDP. The results in case two showed the packet loss rate of NUDP to be slightly better than UDP while the average delay of NUDP is much higher than UDP, thereby meaning that NUDP is not suitable in situations where the size of the intended traffic is small.

NUDP is the protocol that has advantages of both TCP and UDP. It can transmit data quickly and there is mechanism for reducing data loss. It can transmit data quickly same as UDP in situation when there is no data loss occur in network and it also has the data loss rate less than UDP in every situation. For this reason NUDP is a suitable protocol for using with real-time applications and multimedia traffic because both real-time applications and multimedia traffic concern about delay and data loss rate.

REFERENCES

1. A. Johnson, “31 Days Before Your CCNA Exam Second Edition”, ISBN: 978-1-58713-197-4
2. A. Thammadi, “Reliable User Datagram Protocol (RUDP)”, Master thesis, Department of Computing and Information Sciences College of Engineering, Kansas state university.
3. F. A. Shaikh, S. McClellan, M. Singh, and S. K. Chakravarthy, “End-to-end testing of ip qos mechanisms,” *Computer*, vol. 35, no. 5, pp. 80–87, 2002.
4. G. Epiphaniou, C. Maple, and P. Sant, “Affects of Queuing Mechanisms on RTP Traffic Comparative Analysis of Jitter, End-to-End Delay and Packet Loss”, 2010 International Conference on Availability, Reliability and Security, pp.33-40.
5. K. Dileep, A. Saleem, R. Yeonseung, “Quality of Service (QoS) of Voice over MAC Protocol 802.11 using NS-2”, JCIT: Journal of Convergence Information Technology, Vol. 3, No. 4, pp. 76-83, 2008
6. L. F. Akyildiz, J. McNair,”Medium Access Control Protocols for Multimedia Traffic in Wireless Networks” IEEE Network, July/August, 1999, pp.39-47.
7. L. Gui, “A Transport Protocol for Real-Time Applications in Wireless Networked Control Systems”, Master thesis, Faculty of Science and Technology, Queensland University of Technology.
8. Liu, P. X., Meng, M., Ye, X. F., and Gu, J. (2002), “An UDP-based protocol for internet robots”, Proceedings of the 4^{th} World Congress on Intelligent Control and Automation, volumn 1, pp. 59-95
9. P. Ferguson, G. Huston, “Quality of Service: Delivering QoS on the Internet and in Corporate Networks”, John Wiley & Sons, 1998
10. S. W. Liu, S. L. Wu, and J. L. Chen, Adaptive Cross-layer QoS Mechanism for Cognitive Network Applications”, ICACT 2010, pp. 1389-1393, 2010.

11. T. Bova, and T. Krivoruchk, “RELIABLE UDP PROTOCOL”, http://www.ietf.org/proceedings/44/I-D/draft-ietf-sigtranreliable-udp-00.txt, access on 11/05/55
12. T. Issariyakul, E. Hossain, “Introduction to Network Simulator NS2”, ISBN: 978-0-387-71759-3
13. T. Lammle, “CCNA: Cisco Certified Network Associate Study Guide Sixth Edition”, ISBN: 978-0-470-11008-9
14. T. Le, G. Kuthethoor, C. Hansupichon, P. Sesha, J. Strohm, G. Hadynski, D. Kiwior, and D. Parker, “Reliable User Datagram Protocol for airborne network”, In IEEE Military Communications Conference (MILCOM2009), pages 1-6, Boston, Massachusetts, USA.
15. Y. Amir, C. Danilov, S. Goose, D. Hedqviat, A. Terzis, “An Overlay Architecture for High-Quality VoIP Streams” IEEE TRANSACTION ON MULTIMEDIA,Vol. 8, No.6, pp. 1250-1262, 2006
16. http://tools.ietf.org/htmp/rfc1180, last accessed 16/05/55
17. http://www.voip-info.org/wiki/view/QoS, last accessed 05/30/2012

BIOGRAPHY

NAME	Mr. Teeraphol Sukprapaipat
DATE OF BIRTH	17 April 1985
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTIONS ATTENDED	Suan Dusit Rajabhat University,2006-2008 Bachelor of Business Computer Mahidol University, 2009-2013 Master of Science (Technology of Information System Management)
HOME ADDRESS	29/47 Moo 3 Kanjanapisek Rd. Plai Bang Bang Kruai Nonthaburi 11130 Tel. 089-789-2558 E-mail : teeraphol.suk@gmail.com
PUBLICATION / PRESENTATION	Teeraphol Sukprapaipat, A HYBRID NETWORK TRANSPORT PROTOCOL FOR MULTIMEDIA TRAFFIC, ICSEC2013, May 10-11, 2013, Chonburi Thailand