TUHH
Technische Universität Hamburg-Harburg

ComNets
Institute of Communication Networks

Simulation and Modeling of Communication Networks

Final Report

Sajid Maqbool, Roy Ruiz

September 01, 2020

Final Report
Sajid Maqbool, Roy Ruiz
Matriculation 55085, 54705
ICS M.Sc.
Team: A2
Tutor: Leonard Fisser
Examiner: Prof. Dr.-Ing. Timm-Giel
Hamburg University of Technology
Institute of Communication Networks

Contents

Chapter 1

Introduction

A difficult situation caused by the ongoing pandemic forced all onsite activities to be postponed and left with a mere option for the students and teachers to work from home and conduct all the activities via online sources. However, these online activities are being disturbed due to some vulnerable points present in the network where bottlenecks are formed. In order to improve the situation, a deep analysis of the network will be performed and details will be described in this report. First, some theoretical bounds are established to set benchmarks, which would help us evaluate the actual simulations results. These theoretical limits provide a vague idea of the potential bottleneck points in the network. In the next step, these potential limiting points are tested by forming various scenarios. Finally, a few possible solutions are proposed to improve the current situation.

1.1 Problem to be Analysed

The lecturer, who lives in a country side, is engaged in a video conference with a student attending the lecture from the dormitory. In addition to the video conference, the student is also uploading his assignment to the lecturer's FTP server which continues during the entire simulation time. Moreover, the lecturer is using a wired connection and the student is connected to the internet via a wireless connection.

Apart from the student, other tenants in the dormitory are surfing the internet via the WLAN access point and are requesting files from the HTTP server after a consistent interval. We are provided with a trace file of web browsing behavior of other clients. Each web browsing client requests data from the HTTP server after 3 seconds.

The following points should be analyzed to determine the behavior of the network:

1. How can the web traffic data packet size be modeled based on the trace file?
2. How does the presence of the web users and FTP traffic affect the QoS of the video conference at both ends?
3. How does the number of web users influence the QoS of the video conference?

In addition, the analysis should also include some possible solutions to fix the network issues.

1.2 General Description of Network

An overview of the network has been depicted below(see figure 1.1). It consists of a student connected to the internet via a WLAN access point and is attending a video lecture and doing an upload to the lecturer's FTP server. There are some other tenants in the student dormitory doing web surfing via the same WLAN access point. The access point is connected to the dormitory router via a Fast Ethernet link, which offers negligible delay. The dormitory router is connected to the dormitory ISP router via a full duplex VDSL connection. The dormitory ISP router is connected to the HTTP server and the lecturer's ISP router via Fast Ethernet connections with a 60ms delay. An ADSL full duplex link with negligible delay connects lecturer's ISP router to the lecturer's router. The lecturer's FTP server and laptop are connected to the lecturer's router via a Fast Ethernet connection with negligible delay.

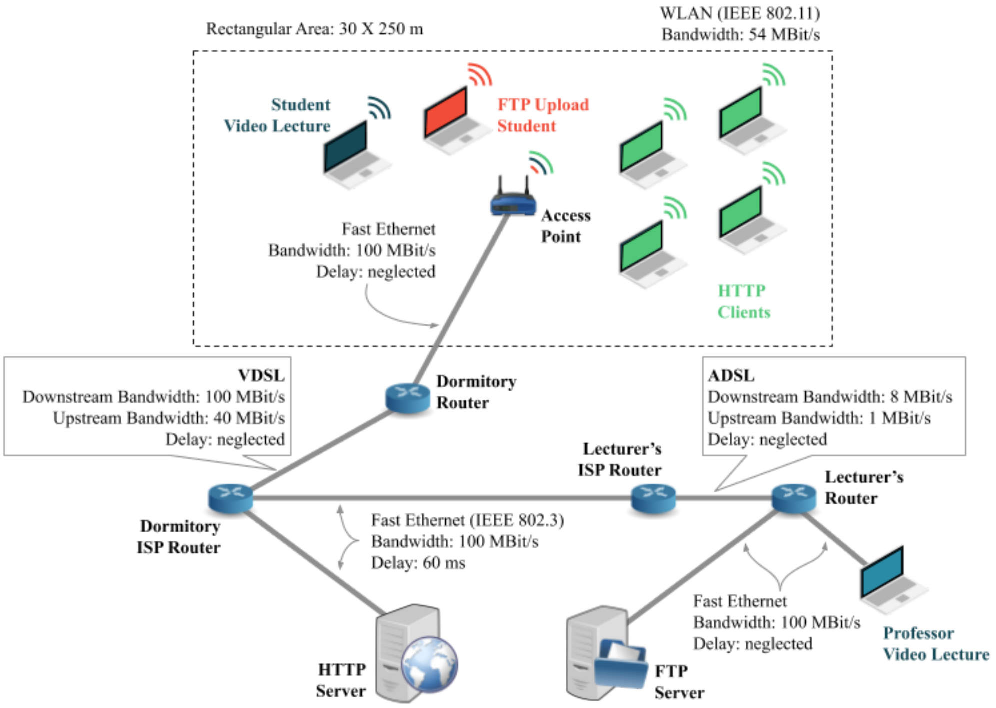

Figure 1.1: Network scenario depiction

A more detailed view of the network parameters given are listed in the table(see table 1.1):

Parameter / Description	Value / Comments
Wireless Local Area Network (WLAN) Protocol	IEEE 802.11g CSMA/CA Medium Access Control
WLAN Bit Rate	54 Mbps
WLAN Access Point Coverage Area	30 m ✕ 250 m
WLAN Request to Send Threshold	3000 B
Fast Ethernet Protocol	IEEE 802.3
Fast Ethernet Data Rate	100 Mbps

Very High Speed Digital Subscriber Line (VDSL) Downlink Data Rate	100 Mbps
VDSL Uplink Data Rate	40 Mbps
Asymmetric Digital Subscriber Line (ADSL) Downlink Data Rate	8 Mbps
ADSL Uplink Data Rate	1 Mbps
Link Delays	varies *shown in network scenario figure*
Router Output Port Buffer Maximum Packet Capacity	70 packets
Router Queue Management Algorithm	Drop Tail Queue
HyperText Transfer Protocol (HTTP) Response Size	$\mathcal{N}(658797, 64411)$ B *corresponds to the given trace file*
HTTP Waiting Time between Response Sessions	3 secs *corresponds to the given trace file*
HTTP Request Size	254 B
Transport Control Protocol Congestion Control Algorithm	TCP New Reno
User Datagram Protocol (UDP) Video Conference (VC) Packet Size	1400 B of Payload *plus protocol headers (RTP, UDP, IP)*
VC Send Interval Time	40 ms
VC Maximum Acceptable End-to-End Delay	150 ms
VC Maximum Acceptable Packet Loss Rate	10%
File Transfer Protocol (FTP) File Size	2 GB *must last during entire simulation time*
Maximum Segment Size (MSS)	1460 B
Receiver Advertised Window	1460000 B

Table 1.1: Network scenario parameters in detail

1.3 Network Services and Requirements

Any user of this network should be able to experience the following services:

- Video Streaming:
  - The lecturer gives a lecture via the university's video conferencing system.
  - A student attends the online lecture from the dormitory.
- FTP Service:
  - A student uploads his/her assignment to the lecturer's server during the lecture.
- Web Browsing:
  - Other tenants in the dormitory may access the Internet via WLAN.

As part of our network setup, our responsibility is to evaluate the video lecture's performance. For that, the network must provide the following Quality of Service (Qos) for the video lecture being streamed bidirectionally:

- Packets arriving beyond 150 ms shall be considered lost.
- Video packet loss rate must not go over 10% in order to avoid service degradation.

Chapter 2

Network Simulation Model

Before discussing the actual results, this section is used to establish theoretical expectations of the network. We were also provided with the behavior trace file of the web surfing clients in the network, which would be used to model the internet traffic.

2.1 General Assumptions

Given the virus pandemic situation we are currently in, the need to maintain a superb Quality of Service (QoS) while video lectures are happening is critical. As communication networks experts, we have been tasked to evaluate the network performance of a video lecture between the lecturer and the student. In order to visualize this, think of a Zoom call where both the student and lecturer are actively engaged during the video lecture. In order to simulate a realistic scenario, we assumed the video lecture would last a total of 60 minutes (3600 seconds). In parallel, the student is also uploading a file (from a previous assignment) to the lecturer's server via FTP (File Transfer Protocol). As we have been informed, all students, including the one engaged in the video lecture and uploading a file, are connected to the dormitory's WLAN (Wireless Local Area Network) via an access point. The other students not participating in the video lecture are browsing the web via HTTP (HyperText Transfer Protocol). These scenarios are all happening simultaneously, and for that, it's best to assume all activities are occurring at the same time.

2.2 Expected Network Behavior

In the given network, QoS between the lecturer and the student should be maintained, but in the dormitory, there are other clients who will be surfing the internet in parallel. Since the radio medium is shared among many users, each one may have to wait for a longer time before the access is granted. Thus, the inclusion of more HTTP clients can degrade QoS because more clients imply the access point has to wait for a longer duration as mentioned before. This wait can potentially lead to queues forming at the access point and, consequently, packet drops. However, the behavior with a few clients is expected to be better with a lower packet loss rate and, thus, better QoS.

2.2.1 HTTP Traffic Via TCP

Before evaluating the network performance of a video lecture, it is best to understand the web browsing behavior of the other students. Fortunately, the networking department of the dormitory captured web browsing behavior statistics of the students. Our responsibility as consultants is to identify the statistical behavior of the HTTP responses by analyzing the given recorded trace. The HTTP response is considered to be the reply length from the HTTP server to the student. Thus, we can consider the response statistics as download statistics from the students' perspective. The statistics provided contain 1000 different sample traces with a minimum and maximum of 455846 and 916033 bytes, respectively. Moreover, the trace file exhibits a mean of ~658.8 KBytes.

With the statistics provided, we perform a hypothesis test by performing a goodness-of-fit test to determine a suitable distribution that characterizes the observed data set. Our null hypothesis is as follows:

H_0: "The HTTP response sizes follow a **normal distribution**"

Furthermore, we performed a chi-square test as part of the distribution fitting analysis. In order to trust the results from the chi-square test, the observed sample traces are divided into equi-distant intervals and the following condition must be met:

$$min(n \cdot p_j) \geq \frac{5 \cdot y(5)}{k},\ 0 \leq j < k \quad [1] \qquad (2.1)$$

where,

$n \cdot p_j$: *expected value*

$k \geq 3$: *number of intervals*

$y(5)$: *number of intervals with* $n \cdot p_j < 5$

Our results indicate the observed sample traces can be divided into 8 equi-distant intervals. In other words, the above condition is met when $0 \leq j < 8$ (see equation 2.1). Consequently, after adding all chi-square values from the table, we obtain a measured chi-square value of ~7.2074. We compared this measured chi-square value to its critical value in order to determine whether we accept or reject the null hypothesis. As we know from distribution fitting analysis, the degrees of freedom depends on the distribution of the sample trace. Given the sample trace behaves like a normal distribution, the degrees of freedom is determined as follows:

$$d_f = k - m - 1 \quad [1] \qquad (2.2)$$

where,

d_f : *degrees of freedom*

k : *number of intervals*

m : *number of parameters of the distribution*

In our case, the degrees of freedom is 5 (see equation 2.2) and its critical chi-square value with a 5% statistical significance level is 11.0705. For this, we accept the null hypothesis since our measured chi-square value is less than the critical chi-square value. Thus, the observed sample traces follow a normal distribution centered around ~658.8 KBytes. Actual values shown in the table (see table 2.1):

j	*Interval*	*Observed* (N_j)	*Expected* $(n \cdot p_j)$	$^2 = \sum \frac{(Observed - Expected)^2}{Expected} = \sum_{0 \leq j < k} \frac{(N_j - n \cdot p_j)^2}{n \cdot p_j}$
0	(455845, 513369]	16	11.1647	2.0941
1	(513369, 570893]	72	74.1886	0.0646
2	(570893, 628416]	221	232.4146	0.5606
3	(628416, 685940]	350	344.6884	0.0819
4	(685940, 743463]	254	242.3871	0.5564
5	(743463, 800986]	78	80.7050	0.0907
6	(800986, 858510]	7	12.6722	2.5390
7	(858510, 916033]	2	0.9330	1.2201

Table 2.1: Response sizes binned in equi-distant intervals including formulas [1]

We further confirmed the trace behaves like a normal distribution by plotting a histogram (see figure 2.1) of the observed and expected values from the table above and the cumulative distribution function (see figure 2.2) of an expected normal distribution and the observed sample trace values. Both graphs show they behave similarly concluding the sample traces of the HTTP responses exhibit the statistical behavior of a normal distribution.

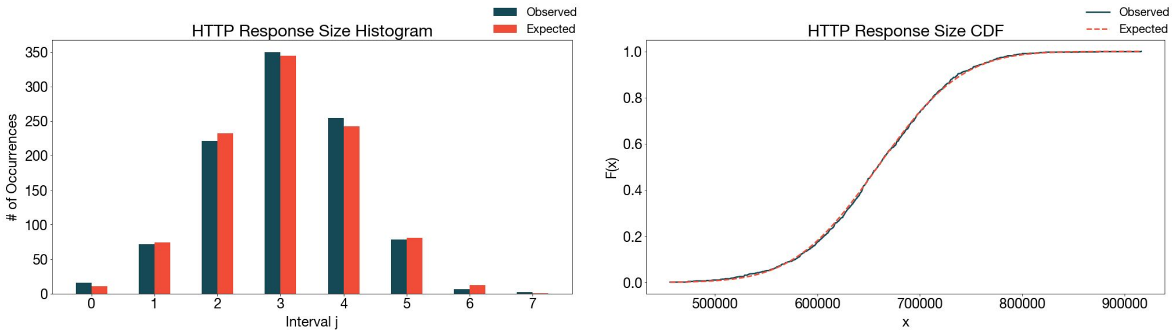

Figure 2.1, 2.2: Observed and expected HTTP response sizes comparison - Histogram (2.1) and CDF (2.2)

Since the HTTP service is modeled as a TCP application, we can estimate the amount of offered traffic assuming the channel is lossless, and the reply is completed within 3 seconds for both uplink (client to server) and downlink (server to client). For the uplink offered traffic, we note it would consist of both request and TCP acknowledgements (see equation 2.3). The details on how we estimated is shown below:

Since,

$$HTTP\ Uplink_{Offered\ Traffic} = HTTP\ Request_{Traffic} + TCP\ ACK_{Traffic} \tag{2.3}$$

$$HTTP\ Request_{Traffic} = \frac{(requestLength + Header\ Size) * 8\ bits}{3\ seconds} = \frac{(254 + 40) * 8\ bits}{3\ seconds} = 784\ bps \tag{2.3a}$$

$$TCP\ ACK_{Traffic} = \frac{TCP\ ACKs_{per\ request} * TCP\ ACK_{size}}{1\ request} * \frac{1\ request}{3\ seconds} = \frac{451 * 64 * 8\ bits}{3\ seconds} \approx 77.0\ kbps \tag{2.3b}$$

where,

$$TCP\ ACKs_{per\ request} = \frac{replyLength}{MSS} = \frac{658800\ B}{1460\ B} \approx 451\ ACKs_{per\ request}$$

$$TCP\ ACK_{Size} = \frac{ACK_{size_{ETH}} + ACK_{size_{P2P}} + ACK_{size_{ETH}}}{3} = \frac{72 + 47 + 72}{3} \approx 64\ B$$

Thus,

$$HTTP\ Uplink_{Offered\ Traffic} = 784\ bps + 77.0\ kbps \approx 77.8\ kbps$$

$$HTTP\ Downlink_{Offered\ Traffic} = \frac{replyLength * MTU * 8\ bits}{MSS * waiting\ time} = \frac{658800 * 1500 * 8\ bits}{1460 * 3\ seconds} \approx 1.8\ Mbps \tag{2.4}$$

2.2.2 FTP Traffic Via TCP

The student is uploading a file to the lecturer's FTP server throughout the simulation via FTP. The receiver side advertises a receive window size 1000 times the Maximum Segment Size (MSS). The MSS, in this case, is 1460 Bytes; therefore, we can estimate the maximum achievable theoretical TCP throughput (see equation 2.6) considering that we have a lossless path. Additionally, the network has links between Dormitory's ISP router and Lecturer's ISP router with a 60ms delay, but all other links have a negligible delay. For this, we can estimate the round trip time for the FTP downlink offered traffic to be two times the delay. In parallel, the FTP uplink offered traffic, which comprises of TCP acknowledgements, is determined as follows:

Since,

$$FTP\ Uplink_{Offered\ Traffic} = \frac{TCP\ ACKs_{Total} * TCP\ ACK_{size} * 8\ bits}{2000\ seconds} = \frac{1369863 * 64 * 8\ bits}{2000\ seconds} \approx 350.7\ kbps \tag{2.5}$$

where,

$$TCP\ ACKs_{Total} = \frac{sendBytes}{MSS} = \frac{2000000000\ B}{1460\ B} \approx 1369863\ ACKs_{Total}$$

$$TCP\ ACK_{Size} \approx 64\ B$$

Thus,

$$FTP\ Uplink_{Offered\ Traffic} = TCP\ ACK_{Traffic} \approx 350.7\ kbps$$

$$FTP\ Downlink_{Offered\ Traffic} = \frac{TCP\ Window\ Max}{Round\ Trip\ Delay} = \frac{1460 * 1000 * 8\ bits}{2 * 60\ milliseconds} \approx 97.3\ Mbps\ [2] \tag{2.6}$$

This is the maximum theoretical limit for the FTP upload, but we have links with less bandwidth in the path between the student and the FTP server. The link with the smallest bandwidth is between the Lecturer's ISP router and Lecturer's router which has bandwidth of 8Mbps. Therefore, this is the theoretical upper bound for the FTP traffic in the network. In other words, the network will never reach the maximum theoretical limit of 97.33Mbps (see equation 2.6). This provides us a benchmark to identify the bottlenecks in the network.

2.2.3 Video Traffic Via UDP

In order to compare our experimental results, we calculate the amount of offered traffic to be generated by the student and lecturer during the video conference. Since the video conference service transmits packets via User Datagram Protocol (UDP), several metrics should be taken into account in order to determine the offered traffic. This includes individual packet size and protocol headers, which can affect the overall bandwidth consumption. We estimate the amount of offered traffic as follows:

Since,

$$Video_{\ Offered\ Traffic} = Packet\ Rate * UDP\ Packet_{\ Size} \tag{2.7}$$

where,

$$UDP\ Packet_{\ Size} = Headers_{\ Size} + Payload_{\ Size}$$

$$Headers_{\ Size} = RTP + UDP + IP = 12 + 8 + 20 = 40\ B$$

$$Packet\ Rate = \frac{1\ packet}{40\ milliseconds} = 25\ pps$$

Thus,

$$Video_{\ Offered\ Traffic} = 25\ pps * (1440 + 40) * 8\ bits = 288\ Kbps$$

Therefore, the student and the lecturer both generate roughly 288Kbps of offered traffic (see equation 2.7) during the video conference service.

2.2.4 Expected Link Utilization

The following table (see table 2.2) lists the types of connections and traffic each link in the network handles. Moreover, the figure gives a visual representation of the expected traffic and respective link utilization of VDSL and ADSL connections. Further noted in the figure (see figure 2.3) are the offered traffic values calculated from the previous sections to have an understanding of what is to be expected from a link utilization standpoint.

Link	Type	Bandwidth	Downlink Traffic	Uplink Traffic
WLAN Access Point ⇋ Dormitory Router	ETH	100 Mbps	UDP + FTP + HTTP	UDP + FTP + HTTP
Dormitory Router → Dormitory ISP Router	P2P	40 Mbps	-	UDP + FTP + HTTP
Dormitory Router ← Dormitory ISP Router	P2P	100 Mbps	UDP + FTP + HTTP	-

Dormitory ISP Router ⇋ HTTP Server	ETH	100 Mbps	HTTP	HTTP
Dormitory ISP Router ⇋ Lecturer ISP Router	ETH	100 Mbps	UDP + FTP	UDP + FTP
Lecturer ISP Router → Lecturer Router	P2P	8 Mbps	UDP + FTP	-
Lecturer ISP Router ← Lecturer Router	P2P	1 Mbps	-	UDP + FTP
Lecturer Router ⇋ FTP Server	ETH	100 Mbps	FTP	FTP
Lecturer Router ⇋ Lecturer	ETH	100 Mbps	UDP	UDP

Table 2.2: Summary of all links and their corresponding bandwidths showing all Downlink and Uplink traffic

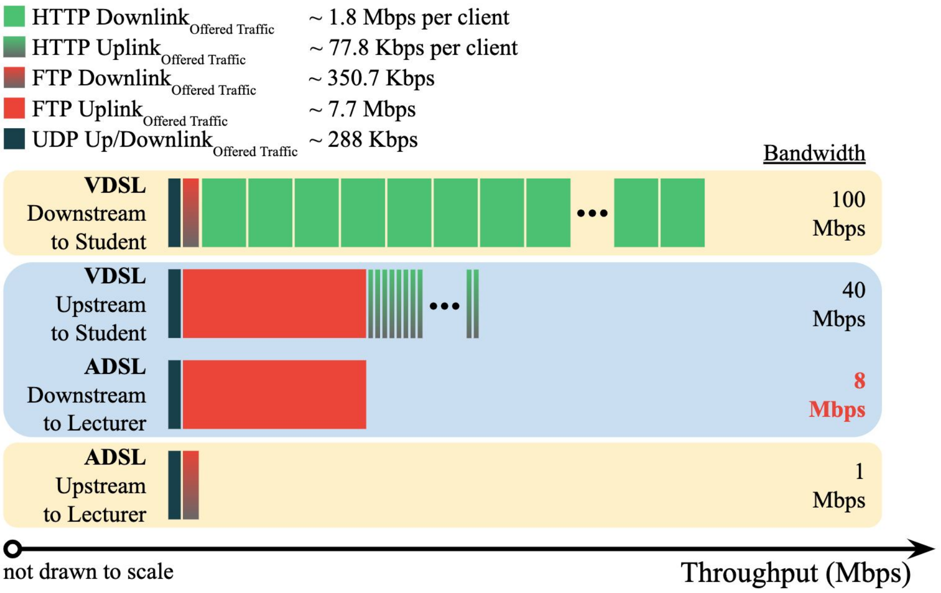
HTTP Downlink Offered Traffic ~ 1.8 Mbps per client
HTTP Uplink Offered Traffic ~ 77.8 Kbps per client
FTP Downlink Offered Traffic ~ 350.7 Kbps
FTP Uplink Offered Traffic ~ 7.7 Mbps
UDP Up/Downlink Offered Traffic ~ 288 Kbps
Bandwidth
VDSL Downstream to Student
100 Mbps
VDSL Upstream to Student
40 Mbps
ADSL Downstream to Lecturer
8 Mbps
ADSL Upstream to Lecturer
1 Mbps
not drawn to scale
Throughput (Mbps)

Figure 2.3: Point-to-Point expected full link utilization for VDSL and ADSL links

2.2.5 Expected Bottlenecks

Based on the theoretical values obtained from the calculations performed above, we expect to see a bottleneck at the Lecturer's ISP router due to the combined FTP and UDP traffic. Since the combined traffic of the FTP upload and video conference is expected to utilize the link to its full capacity, we expect to see packet drops and queuing delay at this point in the network.

The second bottleneck to be expected is at the access point where the network would have more HTTP clients while the server replies to each of them. Since packets are transmitted from both FTP and HTTP servers to their corresponding clients via the access point, this traffic is expected to affect the overall quality-of-service between the student and the lecturer. This would result in both dropped and delayed packets at the access point. The access point is delivering the received data via a shared channel, where the presence of more wireless users reduces the probability of being granted the channel access with every request. Moreover, it introduces delays resulting in queue formation.

2.2.5.1 Routers, Access Point (as HTTP clients increase)

As mentioned above, we expected to see bottlenecks at different points in the network. For the traffic generated by the student, who is engaged in a video conference with the lecturer, the path contains an ADSL link with a bitrate 8 Mbps between the Lecturer's ISP router and lecturer's router. Our theoretical calculation (see equation 2.6) for the FTP downlink offered traffic dictate we can achieve a maximum theoretical throughput of ~97 Mbps. For this, we expect to see a maximum link utilization and possible queues building at the lecturer's ISP router potentially inducing queue delays. We also expect the maximum uplink throughput would be determined by the weakest uplink bandwidth. In this case, it would be the lecturer's ADSL downstream bandwidth of 8 Mbps.

Access points and other wireless clients use Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) protocol for data transmission in 802.11 networks. It uses a multiple access scheme with a carrier sensing and collision avoidance mechanism. As soon as a node receives a packet that is to be transmitted via CSMA/CA, it checks the availability of the radio channel virtually, by means of a network allocation vector (NAV), and physically to ensure that no other node is transmitting at the same time. If both indicators are positive for a distributed interframe space (DIFS) time interval, the frame transfer begins immediately. The collision avoidance mechanism may introduce some additional random delay if there was an ongoing transmission while the transmit request was received. This random delay is known as the backoff interval and if the channel remains idle during this interval, the frame transmission is started.

The backoff interval is calculated as:

$$T_b = b * T_{slot} \text{ [3]} \qquad (2.8)$$

where,

$$b = Uniform\,[0,\ min(CW_{new},\ CW_{max})] \text{ [3]} \qquad (2.9)$$

The backoff interval, b, which is drawn from the contention window and T_{slot} is a duration time, which is a parameter dependent on the physical layer (see equation 2.8). If the channel is busy when the backoff interval reaches zero, the backoff period is set again and the process is repeated.

Each time a transmission fails, the medium access channel (MAC) layer assumes this occurred due to a collision and, thus, further reduces the collision probability by increasing the backoff interval and contention window (CW). The contention window (CW) is calculated as follows:

$$CW_{new} = 2 * (CW_{old} + 1) - 1 \text{ [3]} \qquad (2.10)$$

Initially, the contention window is set to 31, and a new contention window is determined (see equation 2.10). Please note a maximum contention window CW_{max} is defined to be 1023 in the IEEE 802.11 standard used for backoff interval determination (see equation 2.9). In case of a successful transmission, the contention window is reduced to its minimum value of 31.

A virtual carrier sense uses the Network Allocation Vector (NAV) to assess the availability of the channel. The NAV is set according to the duration field of a received frame and this time is measured in milliseconds. A station will not disturb an ongoing transmission even if the physical sensing finds the idle channel since the set NAV indicates an unavailable channel. The wireless nodes in our network also use Ready-to-Send/Clear-to-Send (RTS/CTS) mechanism to avoid the hidden node problem which can cause collision at the receiver. The RTS mechanism informs all the nodes in the vicinity of the sender about the start of a transmission. Accordingly, the intended receiver of the frame sends a CTS to inform its neighboring nodes. The surrounding nodes update their NAV accordingly and refuse to transmit as long as the NAV is set.

Chapter 3

Performance Evaluation

In order to evaluate the performance of the network, different scenarios were established targeting different metrics. These scenarios include:

- Network where the student is uploading a file via FTP and having a video conference while no clients are browsing the web via HTTP.
- Network where the student is uploading a file via FTP and having a video conference while the network experiences an increased number of HTTP clients (1 to 12 clients)
- Network where the student only has a video conference while clients browse the internet via HTTP (4 clients).

We understood there would be some undesired transient effects at the beginning of the simulation**.** These transient effects can potentially skew the results and, therefore, should be properly eliminated. This can be done by setting a warm-up period, which will temporarily disable recording of parameters during such period. This would ensure parameters are recorded after reaching steady-state assuming the simulation reaches steady-state after the warm-up period is over. However, we noticed the warm-up period is not an optimum option as it produces undesirable outliers right at the beginning of the recording phase. For this reason, we decided to perform the simulations for a longer period and agreed on 2000 seconds because it takes around 250 seconds to reach the steady state. Running each simulation for 2000 seconds allows for it to record a substantial amount of data and would help nullify the effect of the transient behavior.

Moreover, in order to visualize and have confidence in our data, we ran each configuration scenario for fifty different runs. This would ensure the data is sufficient and reliable. We would obtain the confidence in the simulation results. It is recommended to have at least thirty data points to make a good conclusion about the data with enough confidence. [4] Choosing fifty was a reasonable choice.

3.1 Video Conference App Metrics

To evaluate the network performance, we modeled the packet loss rate and end-to-end delay. In the following section, we are presenting the details about how we calculated these metrics.

3.1.1 Video Packet Loss Rate Calculation

As mentioned before, the network is required to meet a QoS requirement, i.e the packet loss rate for video conference should not exceed 10%. We implemented this functionality in the video conference application via a C++ file where packets exceeding the 150 ms delay limit are processed as late. The video conference application would only increment a counter but not process the packet via UDP on the receiver side. For this reason, we were able to quantify how many packets were late, received by the application for packets arriving before the 150 ms threshold, and lost in the network (see equation 3.1). This allows us to obtain an accurate representation of the overall packet loss rate for the both the student and lecturer (see equation 3.2):

$$LatePackets_{Total} = Number\ of\ packets\ arrived\ at\ time\ >\ 150\ ms$$
$$LostPackets_{Total} = SentPackets_{Total} - ReceivedPackets_{Total} - LatePackets_{Total} \qquad (3.1)$$

Thus,

$$Packet\ Loss\ Rate\ (\%) = \frac{LatePackets_{Total} + LostPackets_{Total}}{SentPackets_{Total}} * 100 \qquad (3.2)$$

3.1.2 Video End-to-End Delay Calculation

Furthermore, we also computed the time difference between times when every UDP packet was generated to when it was processed at the receiver (see equation 3.3). This gives the total time packet spent from source to destination:

$$Packet\ E2E\ Delay = time\ packet\ is\ processed - time\ packet\ is\ created \qquad (3.3)$$

As consultants, we believe a good evaluation method is to obtain the data as a vector and graph against simulation time. This gives us a more accurate representation of what’s occurring in the queues, especially at the lecturer’s ISP router. It should be noted all packets with an end-to-end delay were either received via UDP or considered late.

3.2 Simulation Results

In this section, a detailed analysis of the network was performed to identify the bottlenecks and test them with various scenarios. As mentioned earlier, we formulated different scenarios to identify the bottlenecks in the network and further tested these points with varying clients and types of users depending upon their usage. The analysis evaluates various metrics including the packet loss rate, end-to-end delay, queue length, etc.

3.2.1 Throughput Evaluation

The following table lists upstream and downstream traffic for the various scenarios which were established. Throughput values for three, six and ten HTTP clients at various links are listed in the table shown below. As shown in red, the link with highest utilization is the ADSL downstream link between the lecturer’s ISP router and lecturer’s router with over 92%

utilization in all cases. The next highest utilized link is the ADSL upstream link with over 51% utilization between the same two routers. Clearly, we can see the bottleneck is at the ADSL link and by investigating the maximum queue lengths at each of the routers, it will point us to the culprit.

			Throughput (Mbps)		**Link Utilization (Pctg.)**	
Link	**Bandwidth (Mbps)**	**HTTP Clients**	Downlink	Uplink	Downlink	Uplink
WLAN Access Point ⇋ Dormitory Router	100 Mbps	3	2.12	7.77	2.1%	7.8%
		6	3.59	7.72	3.6%	7.7%
		10	5.47	7.66	5.5%	7.7%
Dormitory Router ⟶ Dormitory ISP Router	40 Mbps	3	-	7.65	-	19.1%
		6	-	7.58	-	19.0%
		10	-	7.50	-	18.8%
Dormitory Router ⟵ Dormitory ISP Router	100 Mbps	3	1.98	-	2.0%	-
		6	3.43	-	3.4%	-
		10	5.29	-	5.3%	-
Dormitory ISP Router ⇋ HTTP Server	100 Mbps	3	1.48	0.06	1.5%	0.1%
		6	2.95	0.12	3.0%	0.1%
		10	4.84	0.20	4.8%	0.2%
Dormitory ISP Router ⇋ Lecturer ISP Router	100 Mbps	3	0.64	7.70	0.6%	7.7%
		6	0.64	7.60	0.6%	7.6%
		10	0.63	7.46	0.6%	7.5%
Lecturer Router ⟵ Lecturer ISP Router	8 Mbps	3	**7.60**	-	**95.1%**	-
		6	**7.50**	-	**93.8%**	-
		10	**7.37**	-	**92.1%**	-
Lecturer Router ⟶ Lecturer ISP Router	1 Mbps	3	-	**0.52**	-	**51.8%**
		6	-	**0.51**	-	**51.4%**
		10	-	**0.51**	-	**51.0%**
Lecturer Router ⇋ FTP Server	100 Mbps	3	0.35	7.41	0.3%	7.4%
		6	0.34	7.30	0.3%	7.3%
		10	0.34	7.17	0.3%	7.2%
Lecturer Router ⇋ Lecturer	100 Mbps	3	0.29	0.29	0.3%	0.3%
		6	0.29	0.29	0.3%	0.3%
		10	0.29	0.29	0.3%	0.3%

Table 3.1: Throughput and link utilization efficiency for downlink / uplink traffic with 3, 6, 10 HTTP clients

3.2.2 Video Packet Loss Rate Evaluation

As a means to observe the difference in packet loss rate between the student and lecturer, we perform the simulation with no students browsing the web, by setting the number of HTTP clients to zero. We observed the lecturer experiences a ~12.16% loss rate; whereas, the student experiences ~0.08%. This is the worst-case scenario for the lecturer.

3.2.2.1 Lecturer’s Video Packet Loss Rate

The loss rate for the lecturer shows a decreasing trend as the HTTP clients increased. However, this improvement at the lecture's end induces the degrading effect on the student’s QoS when HTTP clients are increased as explained in the next section.

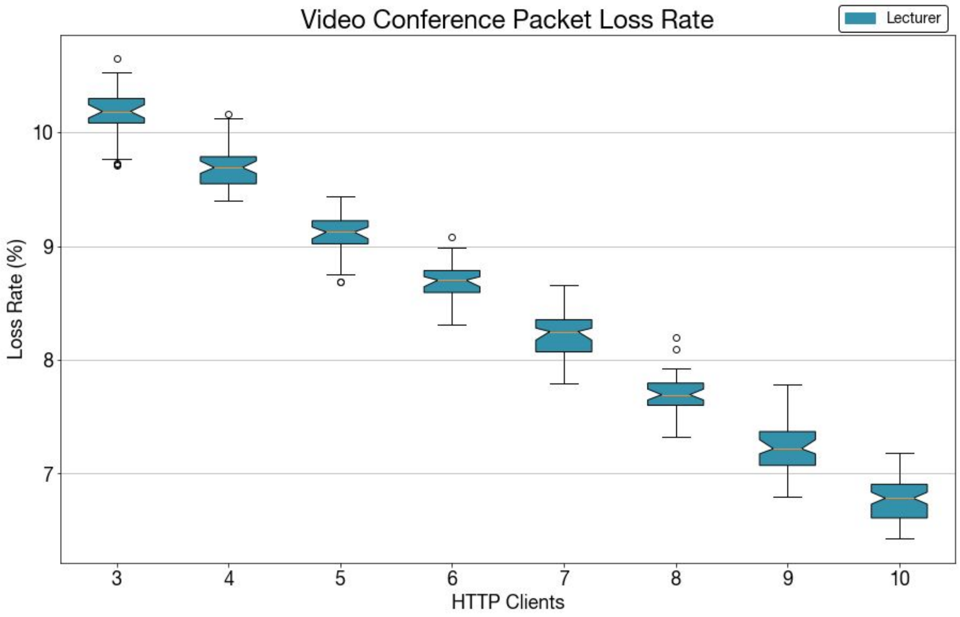

Figure 3.1: Video conference application packet loss rate for the lecturer by HTTP clients

The decreasing trend in the loss rate for the lecturer is evident (see figure 3.1). The loss rate is well above the threshold limit with a few HTTP clients i.e for three HTTP clients the average is about 10.2%. However this average further dips as more web clients are added. When there are four web clients, the loss rate averages 9.7% with some points from the distribution above the 10% limit. However, adding another client further improves the loss rate bringing the average down to 9.1% with all data points well below the required limit.

This trend appears due to the fact that when there are very few users using the shared medium, the FTP and UDP packets are processed faster at the access point. Less users in the network means that the probability to be granted the MAC channel access for the FTP upload is higher. This ensures faster transmission of packets at the lecturer’s router. This behavior leads to the queue formation at the bottleneck point, which in this particular route, is present at the lecturer’s ISP router, where the link’s bitrate reduces from 100Mbps to 8Mbps.

However, when the number of HTTP clients increases, probability to be granted the medium access decreases whenever requested by the FTP upload student. Probability of longer wait between successful attempts increases. More users use the shared medium; thus, the packets are spaced in the time domain resulting in a reduced packet flood at the lecturer's ISP router.

Late packets contribute maximum to the degraded QoS for the lecturer. Queue formation reduces at the lecturer's ISP router due to the inclusion of more HTTP clients, packets wait for relatively lower times before they are served; therefore, the number of late UDP packets also decreases which results in an improved QoS for the lecturer.

Moreover, it is also safe (see figure 3.2) to assume that when five HTTP clients are served in the network, the loss rate is well below the required limit resulting in a better QoS experience for the lecturer. This number is taken as an indication of the minimum number of HTTP clients surfing the web required for the lecturer to experience a better QoS.

3.2.2.2 Student's Video Packet Loss Rate

As mentioned in the previous chapter, a bottleneck is expected to appear at the access point when the network experiences more users surfing the web. Since each web user is requesting a file every three seconds, having more of them in the network increase the overall traffic volume directed towards the access point. The access point uses the MAC channel to transmit the data. More clients result in either queues overflowing or waiting to be served at the access point, which leads to dropped and delayed packets, respectively.

The packet loss rate for the student is directly proportional to the number of HTTP clients in the network with a linear behavior (see figure 3.2). The best QoS for the student occurs when the network contains no users surfing the web, which is not a realistic scenario. The presence of barely three HTTP clients begins to induce a loss rate at about 3.4%. This rate continues to rise and crosses the allowed limit of 10% when nine clients are web surfing (~10.1%). The student engaged in a video conference would experience an acceptable quality-of-service with at most eight users surfing the web simultaneously (~9.0%).

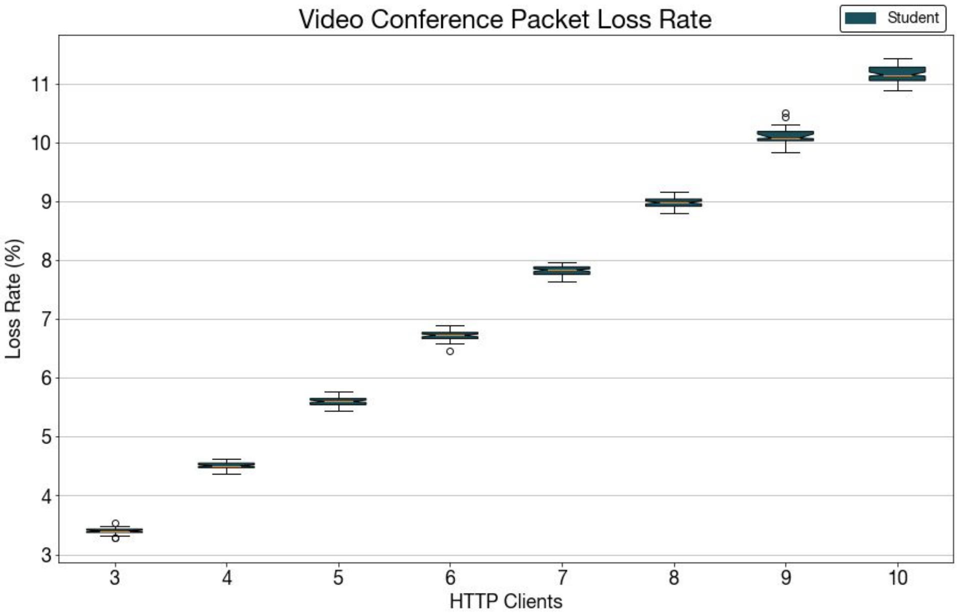

Figure 3.2: Video conference application packet loss rate for the student by HTTP clients

Degraded QoS is caused due to late and dropped packets at the access point. These drops occurred due to long pending queues at the access point resulting in new incoming packets being dropped. As we increase the number of HTTP clients, more packets arrive at the access point resulting in a bottleneck situation, where more and more packets are dropped. Thus, eight HTTP clients act as an upper bound of HTTP clients for the student to experience a better QoS based on the network scenario.

3.2.3 Video Packet Delay Evaluation

Based on the results from the previous section, it is clear both the student and lecturer can experience a better QoS when a certain number of web users are present in the network. We also established that too little or too many of these web surfing clients degrade the service for the lecturer and the student, respectively. Five to eight HTTP clients is a conservative range to ensure both lecturer and student experience a better quality-of-service. Furthermore, we established the presence of six or seven web users provides a more optimal QoS.

3.2.3.1 Lecturer’s Video End-to-End Delay

For this, we further investigated the scenario where the network exhibits six web users. We gathered end-to-end delay data for all video packets received by the student and lecturer (see equation 3.3). As already determined, the lecturer’s ISP router is a bottleneck due to almost max link utilization (see table 3.1) for downstream traffic with respect to the lecturer. The end-to-end delay graph (see figure 3.3) depicts late packets in red, i.e. above 150 ms, and correctly received packets in blue, i.e. below or at 150 ms.

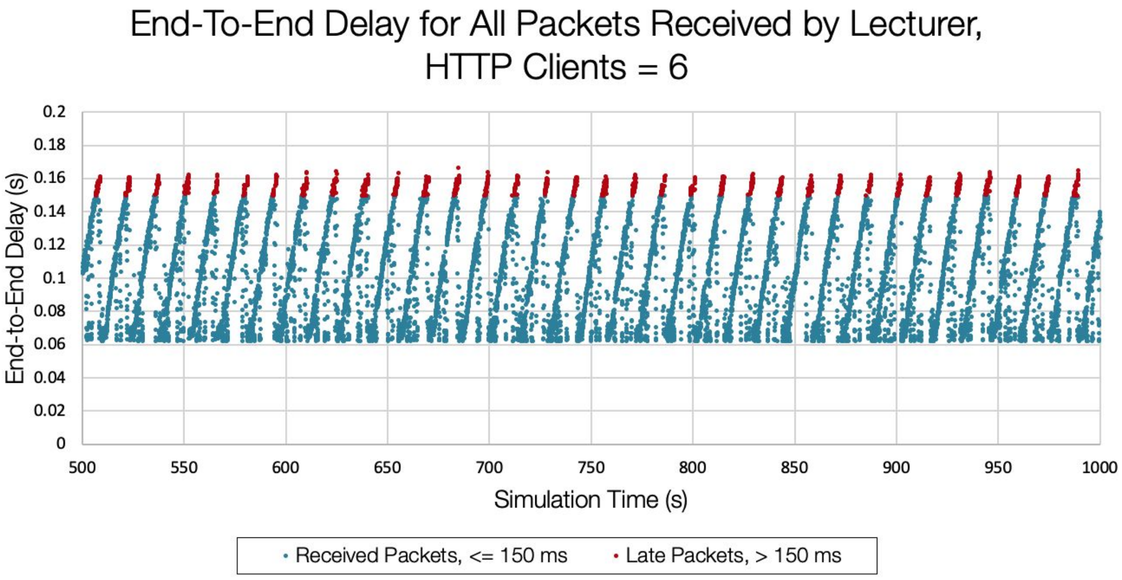

Figure 3.3: Video conference application end-to-end delay for the lecturer with 6 HTTP clients

To better see the behavior, we only showed a snapshot of the data between 500 and 1000 seconds of simulation time. Out of the total received packets by the lecturer, there are roughly ~9.5% late packets (in red) depicted at the top of the sawtooth pattern. The sawtooth behavior is observed because the queues are building at the lecturer's ISP router when the TCP window for FTP upload is increasing along with the UDP traffic (see figure 3.3). As a result, packets spend more time in the queues before they are served. A continuous rise in the waiting time appears as the queues build up. Then, when the queues are full, a drop of an FTP segment occurs. When a segment is dropped, TCP assumes this happened due to the congestion in the network and makes adjustments by decreasing the window. This decrease allows the router to flush its queues as observed from the graph.

3.2.3.2 Student's Video End-to-End Delay

The end-to-end delay behavior for the student (see figure 3.4) depicts a different situation. The bottleneck occurs at the access point where traffic from multiple sources is arriving randomly. The average delay for the UDP packets is ~75 ms; however, there are a few packets that take over 180 ms. These late packets are ~6% of the total received packets.

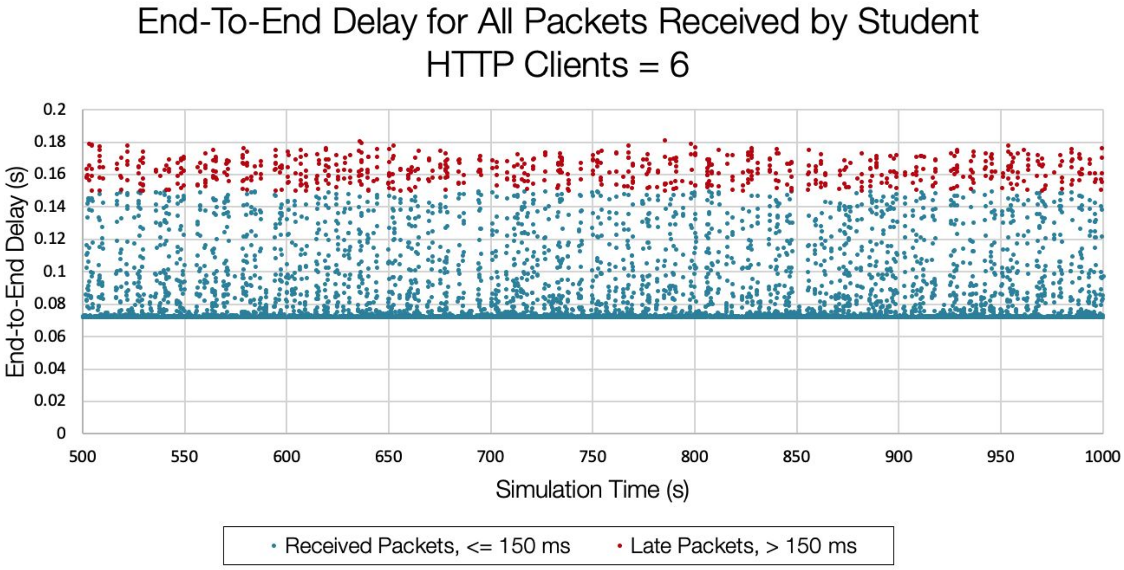

Figure 3.4: Video conference application end-to-end delay for the student with 6 HTTP clients

The traffic at the access point is arriving from multiple sources and is, comparatively, much higher than the traffic at the lecturer's ISP router. Therefore, in comparison to the lecturer, the lost packets in the network contribute more to the degraded QoS for the student. Since there is a shared wireless medium on the student's side and multiple clients are trying to access it to receive the data, the queues start to build. The packets arrive at a higher rate from all other sources. This includes replies from the HTTP server, UDP video packets from the lecturer and FTP acknowledgments from the FTP server. These packets are served whenever the shared medium is available resulting in large delays occurring randomly throughout the simulation. Consequently, this leads to filled queues at the access point and further incoming packets being dropped.

3.2.4 Further Look into Results

As analyzed in the previous section, the degraded service for both the student and lecturer is caused by the packet drops due to congestion and late packets resulting from large queues. In this section, we looked further into some other metrics to solidify our findings.

3.2.4.1 Queue Length Analysis

We observed the queues at the lecturer's ISP router being filled to its full capacity downstream to the lecturer (see figure 3.5). The same was not observed at the output port for the other routers downstream to the lecturer.

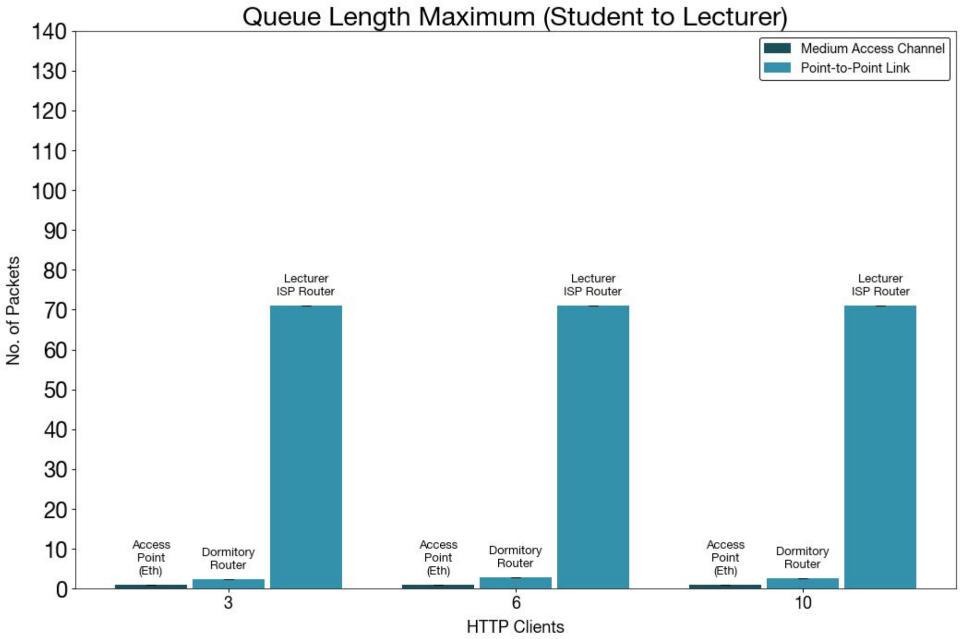

Figure 3.5: Maximum packet queue length at all routers and access point with data flowing from student to lecturer and 3, 6, 10 HTTP clients

Furthermore, it confirms the lecturer's ISP router presents a bottleneck for the UDP directed towards the lecturer. Even though increasing the number of HTTP clients improves the QoS for the lecturer, the queue at the lecturer's ISP router overflows due to TCP traffic sent to the FTP server.

On the other hand, we can see large pending queues building at the access point (see figure 3.6) for traffic downstream to the student. These queues not only cause packet drops but also induce delays resulting in a substantial amount of UDP packets received late by the student.

Since the access point serves multiple clients, UDP packets sit in the queues waiting to be served.

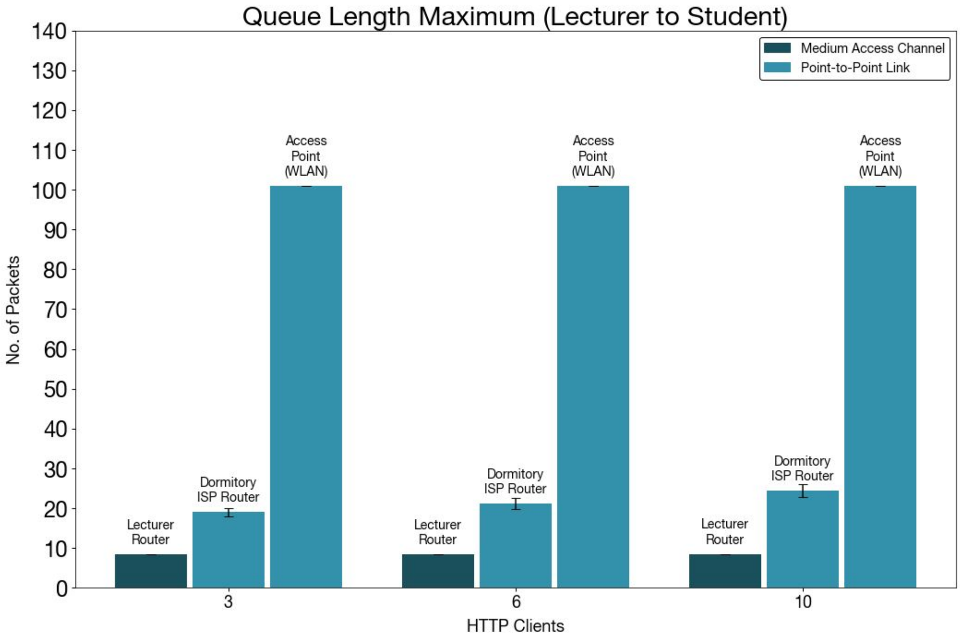

Figure 3.6: Maximum packet queue length at all routers and access point with data flowing from lecturer to student and 3, 6, 10 HTTP clients

As discussed earlier, the loss rate increases for the student as the number of web users increases. This happens because the rate at which packets arrive at the access point far exceeds the rate at which they are served resulting in the pending queues. The inclusion of more HTTP clients means that more data is requested from the server resulting in more TCP segments to arrive. On the other hand, the medium access channel has its own limitations where only one device can use it at a time. In this case, the access point experiences longer waits before it is granted access to the shared channel resulting in the packets being put into the queue waiting to be served.

Since the maximum queue length at the access point is set to 100 by default, it is not enough to buffer all the incoming packets resulting in packet drops. The access point uses a first-in-first-out (FIFO) mechanism. The TCP contention window continues to grow (see equation 2.10) adding more and more segments before negative acknowledgements are received. FIFO pushes all responsibility for congestion control and resource allocation out to the edges of the network. Thus, congestion control assumes no help from the routers rather TCP takes responsibility for detecting and responding to congestion. [5]

3.2.4.2 Dropped Packets Analysis

As a next step in our investigation, we performed a more in-depth analysis into the packet loss rate; therefore, we analyzed the total number of lost packets. There are two main factors contributing to the degraded services: packets lost due to congestion and delayed packets caused by the queuing. Following a similar approach, we considered three cases of HTTP clients, three, six, and ten, respectively. These three cases demonstrate either the lecturer or student experiencing a better QoS or both.

From all three scenarios, it is clear the contribution to the lecturer's degraded service is due to late packets (see figure 3.7). The number of lost packets is negligible.

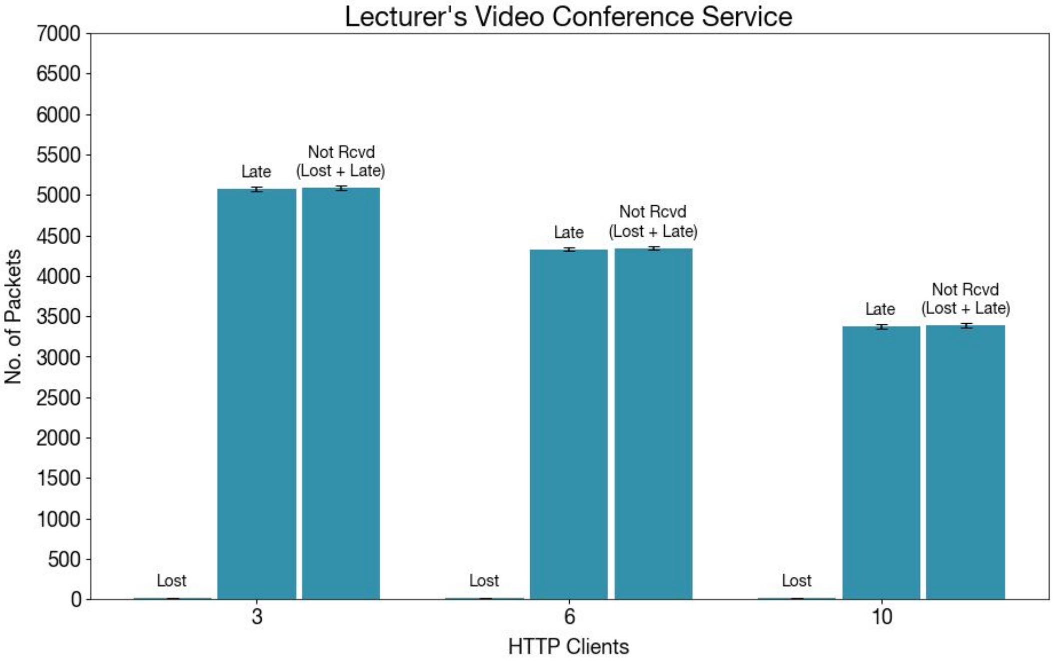

Figure 3.7: A comparison of late and lost packets for the lecturer's video conference service

For three HTTP clients, on average 12 packets are lost due to queue overflow, and this number remains constant for all scenarios; whereas, the number of late packets is substantially higher and reduces as the number of HTTP clients increases. Despite the lecturer experiencing a better QoS from a packet loss rate standpoint, there's still a substantial amount of late packets in the queue at the lecturer's ISP router due to the increasing FTP traffic. When HTTP clients are increased, the shared medium access channel is used by more users; thus, FTP traffic arrives with some delay allowing the lecturer's ISP router an opportunity to flush its queues even quicker. For this reason, we observe a decreasing trend in the late packets and continues to decrease as more HTTP clients are introduced.

On the other hand, the impact of the lost packets due to overflowing and pending queues at the access point is much more significant for the student. As we observed in the lecturer's case, although the student also experiences a degraded service due to late packets caused by the queues at the access point, packets lost in the network contribute higher in comparison to the lecturer (see figure 3.8). From the graph, we can observe the situation worsens for the student as the number of web users increases. Consequently, more traffic is present at the access point resulting in a higher frequency of overflowed queues increasing packet drops and delays.

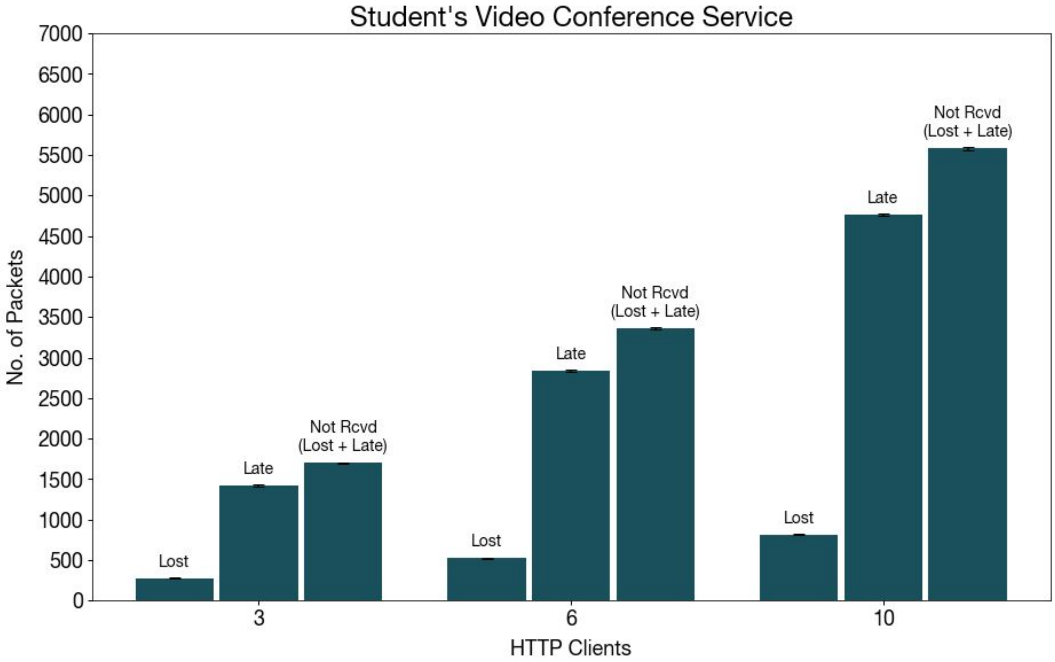

Figure 3.8: A comparison of late and lost packets for the student's video conference service

For the scenario with three HTTP clients, the total late packets are 1420 while 280 packets are lost due to queue overflow. Out of the total dropped packets (late and lost), roughly 16% are lost packets due to queue overflow. The student's QoS continues to degrade with the increasing number of HTTP clients. The buffer size at the access point is not enough to hold all packets waiting to be served and results in more and more losses as web users increase.

3.3 Special Scenarios

In order to complete our analysis, we analyzed some additional scenarios. These scenarios included:

- Removal of HTTP Clients
- Removal of FTP Clients

3.3.1 Removal of FTP and/or HTTP client

We began our analysis by looking at the packet loss rates for both the student and lecturer when only the FTP service is present. The student received almost all packets within the acceptable end-to-end delay limit. In fact, the loss rate for the student was ~0%. The average end-to-end delay indicated there was no late packet for the student.

However, the loss rate for the lecturer was found to be over 12%. Interestingly, the late packets contributed more towards the degraded service for the lecturer as we witnessed in the previous sections for other scenarios (see figure 3.9). Lecturer faces degraded QoS due to the queuing delay induced by the lecturer's ISP router and packets arriving beyond the 150ms threshold limit.

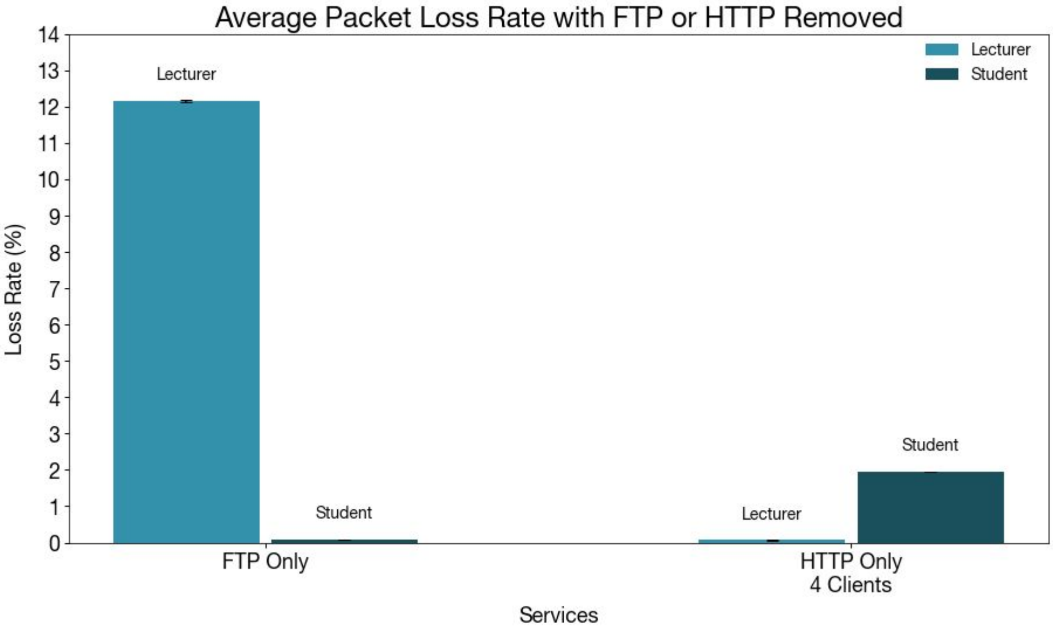

Figure 3.9: Packet loss rate for the lecturer and student after removal of FTP or HTTP service

In the second scenario, the FTP service was removed and replaced with HTTP service only (servicing four clients). As expected, the video packet loss rate for the lecturer was negligible. All the links have enough bandwidth to handle the generated traffic with negligible loss rates in the absence of the FTP upload. We start to see losses for the student as the number of web users increases. Since no TCP traffic is present in the links via FTP, we can assume the number of web users would be higher than previously suggested eight users in the previous section.

3.4 Conclusion

As observed in the evaluation, the video packet loss rate for the lecturer and student behave in an opposite manner. That is, the lecturer's packet loss rate decreases as the number of HTTP clients increase (more users surf the web); whereas, the packet loss rate for the student increases. The results were compelling with a 95% confidence and noticeable trend. Conservatively, if the number of web users are kept between five to eight, both the lecturer and student will have experienced a fairly acceptable quality-of-service since the overall packet loss rate is kept within 10%. However, both the lecturer and student will experience service degradation from time to time since there was a substantial amount of late and lost packets on both ends of the service. As a reminder, in order to maintain a good quality-of-service, the video conference application has a maximum acceptable delay of 150 ms.

The main factors attributing to this increase in delay resulting as late are due to both the lecturer's ISP router and WLAN access point experiencing a bottleneck. This was further discussed by showing the link utilization efficiency when the network experiences three, six, and ten HTTP clients, respectively. The lecturer's ISP router is over 90% utilized. Thus, the congestion observed and discussed in the performance evaluation section (see chapter 3).

Moreover, the bottleneck was observed at the WLAN access point resulting in both late and lost packets as discussed in the previous sections. This is attributed to the pending queues at the access point, in which packets are being delivered to all clients served via the access point using a CSMA/CA mechanism. Thus, in order to avoid packet collision, the medium must be free resulting in higher backoff interval, and consequently, delayed packets.

In the next chapter, we discuss improvements that can serve both the student and lecturer to experience a higher quality-of-service.

Chapter 4

Improvements

In chapter 3 of this report, we identified the bottlenecks in the network and tested the network behavior with various scenarios targeting these points. Each of these points behave differently depending on the presence of the users while utilizing different services. There are several ways to improve the QoS between the student and the lecturer, and it depends on the changes allowed. These include:

- Configuration Changes
- Resource Reservation
- Structural Changes
- Usage Changes

4.1 Configuration Changes

In configuration changes, Differentiated services (DiffServ) technique can be used to implement the QoS between the student and the lecturer. DiffServ gives the control of the traffic and allows to configure which traffic to be accepted and what traffic to be discarded. QoS requirements are used to classify the inbound traffic on a particular interface. A class consists of a set of rules that identify the class of the packets. Inbound traffic is separated into various classes based on Network layer and Transport layer headers and the Virtual LAN (VLAN) ID and is marked with a corresponding DSCP value. Furthermore, policy is used to define QoS attributes for various traffic types. These attributes may include the ability to mark the packets. Edge devices such as routers are responsible for separating the inbound traffic into sets of traffic classes and are responsible for determining a packet's classification.

Traffic condition policy is associated with an inbound traffic class and specifies the actions to be performed on packets meeting the class rules. All packets meeting a given header receive the same marking, and it does not depend on their arrival rate. This way the UDP packets from the student and lecturer can be marked and they can receive specific forwarding services i.e higher priority forwarding. These packets can be stored in a dedicated queue and are served first whenever requested. [6][7] Since UDP packets are now prioritized and placed in separate queues, they will not have to wait in the normal router queue before they are served as it was in the presence of other traffic types.

4.2 Resource Reservation

Resource Reservation Protocol (RSVP) can be used as a solution to achieve the required QoS between the student and the lecturer. It enables routers to dynamically manage their bandwidth and this includes reserving bandwidth for certain applications. The goal of this exercise is to keep the loss rate and the end-to-end delay below the specified threshold. Using RSVP, the bandwidth is reserved for real-time applications such as the video conference, and routers give priority to the UDP packets. This is achieved by using the connection-oriented data transmission on the packet-switched network by establishing a virtual path between the sender and the receiver.

An RSVP packet is sent to the receiver before establishing the connection. End devices inform the network about their bandwidth requirements and maximum delay time of the data packets. Every node in the path checks these requirements and if they can be met, a connection is established to create a virtual path in the network. Once a path is established, each packet always follows the same defined path. As long as the communication is in progress, the routers in the path continue to monitor and ensure the bandwidth requirements are met. [8]

If a virtual path can be established between the student and lecturer and resources are allocated along the dedicated virtual path before they begin their video conference, the required QoS can be achieved.

4.3 Infrastructural Changes

Another possible improvement is to either increase the bandwidth slightly or reduce the buffer size. The buffer size for a single TCP flow is calculated using the product of bandwidth and delay. It gives the amount of unacknowledged data required in the network to utilize the congested link to its full capacity. Buffering allows you to keep the link busy when TCP window is reduced as a result of the segment loss. While buffering improves link utilization and maximizes the throughput, it induces queuing delay which degrades the QoS for real time applications such as video conference.

Buffering is necessary to keep utilizing the congested link when TCP window halves as a result of a lost segment. If the buffer size is small, the sender halves the window after a loss and waits for the ACK's, there is insufficient reserve in the buffer to keep utilizing the bottleneck link. The buffer goes empty, the bottleneck link goes idle and throughput is reduced. On the other hand, if the buffer size is larger and TCP halves its window, the buffer is not completely empty. As a result, the queuing delay is increased because the buffer always has packets queued. [9]

In order to reduce the loss rate for the lecturer, we can either increase the bandwidth slightly so that packet queues are served faster at the lecturer's ISP router, or we can decrease the buffer capacity so that queuing is reduced and thus packets don't sit in the buffer waiting to be served. We tested these suggestions with FTP and video conference services because this scenario produces the maximum loss rate for the lecturer.

Aside from a few outliers, all packets received by the lecturer were within the maximum acceptable delay of 150 ms (see figure 4.1).

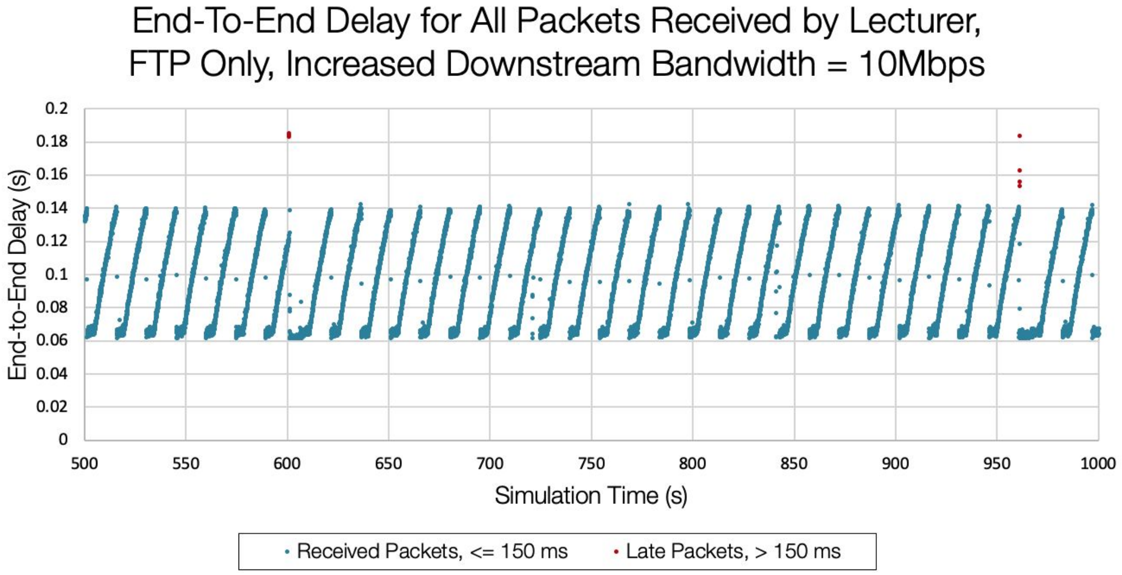

Figure 4.1: End-to-end delay with 10 Mbps bandwidth

When the buffer size is increased to accommodate more packets, the situation for the UDP worsens (see figure 4.2). Around 27% of the packets are late when they reach the lecturer.

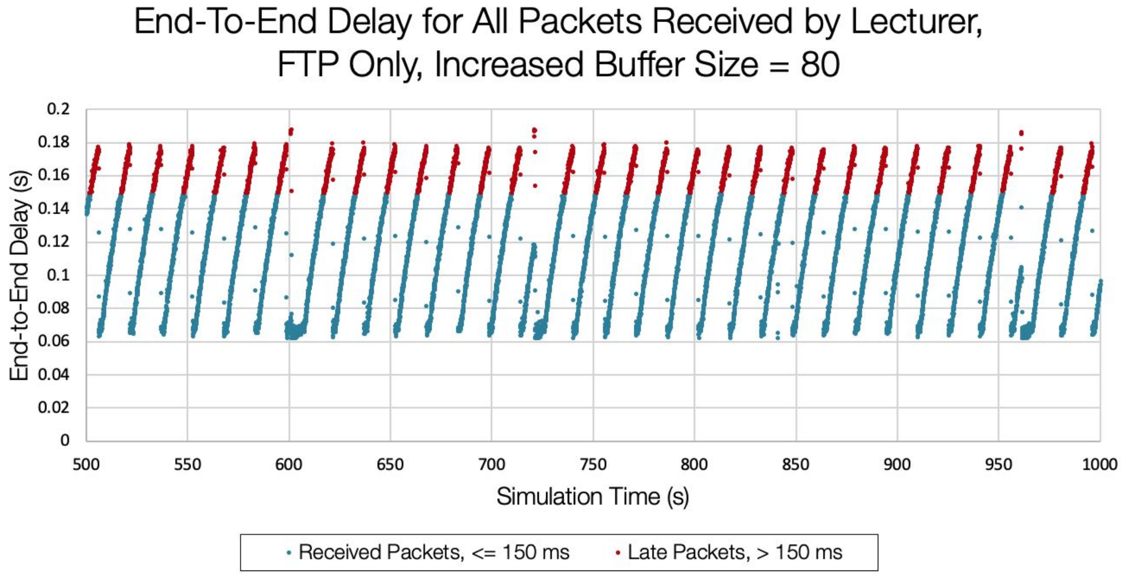

Figure 4.2: End-to-end delay with increased buffer at lecturer's ISP router

Furthermore, reducing the buffer size improves the end-to-end delay for the lecturer. However, the link utilization reduces the congested link, and the throughput slightly decreases to under 7 Mbps (see figure 4.3). The end-to-end delay for the UDP packets is below 120 ms for almost all packets.

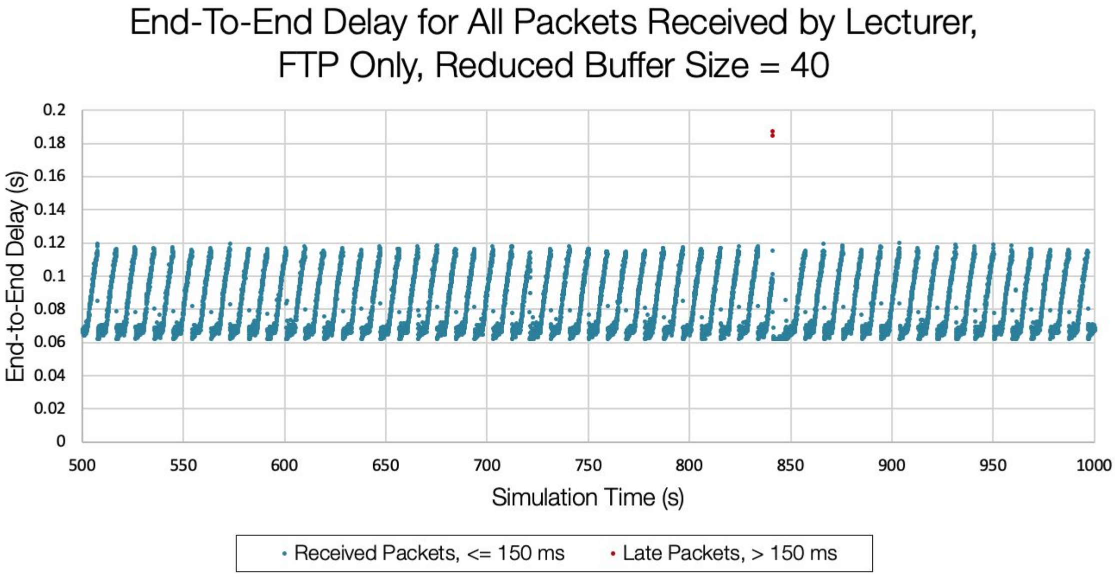

Figure 4.3: End-to-end delay with reduced buffer size

4.4 Usage Changes

In chapter 3, analysis of the network revealed that the main factor in degrading the services is the FTP service when there is a video conference in progress between the student and lecturer. A simple solution to the problem could be to change the usage behavior of the FTP service. If the student refrains from uploading a file via FTP during the video conference, the degraded QoS can be avoided.

Bibliography

[1] A. Timm Giel, Class Lecture, Topic: "Lecture 5 - Hypothesis Testing." 57767_S20, Institute of Communication Networks, Technische Universität Hamburg, Hamburg, May 2020.

[2] Terry Slattery. “TCP Performance and the Mathis Equation” Internet: https://netcraftsmen.com/tcp-performance-and-the-mathis-equation/, [Accessed: Aug. 21, 2020].

[3] A. Timm Giel, Class Lecture, Topic: "Lecture 7 - Wireless LAN - IEEE 802.11." 17943_S19, Institute of Communication Networks, Technische Universität Hamburg, Hamburg, Dec 2019.

[4] A. Timm Giel, Class Lecture, Topic: "Lecture 6 - Simulation Result Analysis." 57767_S20, Institute of Communication Networks, Technische Universität Hamburg, Hamburg, May 2020.

[5] L. Peterson and B. Davie, “Congestion Control: Queuing Disciplines,” in Computer Networks - A Systems Approach, 5th ed., Elsevier, Inc. Massachusetts: Morgan Kaufmann, 2012, ch. 6, sec. 6.2, pp. 264-268.

[6] “What is differentiated services (DiffServ) and how does it work with my managed switch?”. Internet: https://kb.netgear.com/21748/What-is-differentiated-services-DiffServ-and-how-does-it-work-with-my-managed-switch, [Accessed: Aug. 28, 2020]

[7] “Differentiated Services.” Internet: https://inet.omnetpp.org/docs/showcases/general/diffserv/doc/, [Accessed: Aug. 28, 2020].

[8] “Prioritization and queuing.” Internet: https://www.elektronik-kompendium.de/sites/net/1306121.htm, [Accessed: Aug 30, 2020]

[9] G. Appenzeller, I. Keslassy, and N. McKeown, “Sizing router buffers.” In Proceedings of ACM SIGCOMM'04, Aug 2004. [Online]. Available: http://yuba.stanford.edu/techreports/TR04-HPNG-060800.pdf [Accessed: Aug 30, 2020]