ENSC 833：NETWORK PROTOCOLS AND PERFORMANCE

Final Project

Performance analysis of video streaming (YouTube) by Riverbed Modeler

Project website

https://lichengb.wixsite.com/sp2022ensc833team07

Team 7

Prepared by:

Licheng Bao
301296884
lichengb@sfu.ca
Linqi Cheng
301271424
linqic@sfu.ca
Yiran Wang
301473391
ywa391@sfu.ca

Spring 2022 Project prepared for:

Prof. Ljiljana Trajković &

Mr. Zhida Li

April 30, 2022

SFU

Abstract

This project tries to use the Riverbed Modeler 17.5 to simulate and analyze YouTube video streaming performance. Implementing the project will use the data from YouTube and design a series of scenarios, including a relatively ideal scenario, a scenario of video browsing with other users, a scenario with the effect of different data rates, and a scenario with the movement. These scenarios will consider the evaluation criteria such as latency, throughput, etc.

Keywords: video browsing, mobile, throughput, delay 802.11g/n, movement

Content

Table of Contents

1.1 Introduction

The increasing popularity of multimedia applications will lead to substantial growth in video traffic on both wireline and wireless networks [1]. As a result, evaluating the performance of network topologies, protocols, and procedures for video traffic is critical. YouTube is the second most visited website [2]. In 2016, the number of hours of YouTube content reached 10 billion hours [3]. There are tons of videos related to all the subjects we learn, making YouTube a top choice when students face problems and searching for help. Therefore, there are usually some students watching YouTube videos in the library. The report mainly focuses on the simulation of this scenario. Since the user experience is closely related to the smoothness of video playback, the simulation's focus will be on throughput and delay.

As the most commonly used technology in home networks, Wi-Fi is also used widely on campus. To increase the dependability of audio and video transmission, the IEEE 802.11 standard specifies many multicast techniques [4]. We choose IEEE 802.11n and IEEE 802.11g as our main protocols.

Besides, one more scenario is added to find the influence of moving at different speeds. Mobile stations change positions in a real-time IEEE 802.11 WLAN scenario, resulting in imbalanced load distribution on available access points (APs)[29]. The movement will cause increased packet loss and unstable throughput [29].

This report will describe the background of video streaming, Wi-Fi standard, Riverbed Modeler, and quality of service briefly. Then, it will describe the overview of the project and related work in this field. Next, the project will select and design some possible scenarios around a mobile. Then the report will detail how to implement these scenarios through Riverbed Modeler. After that, it will show the results of each scenario and discuss them. Finally, the report will conclude and provide some possible suggestions for improvements and future work.

1.2 Background

1.2.1 Video streaming

Video streaming is a continuous transmission of video files from a server to a client. It enables users to view videos online without having to download them [5]. Video streaming on YouTube is accomplished through TCP (Transmission Control Protocol) at the transport layer [6]. At the application layer, YouTube takes advantage of DASH, which stands for Dynamic Adaptive Streaming over HTTP. YouTube makes use of a MPEG-DASH variation of the DASH protocol [7]. Footage streaming services like YouTube and Netflix employ HAS to break the video into chunks that are subsequently encoded using multiple video qualities, as seen in the Figure below [8].

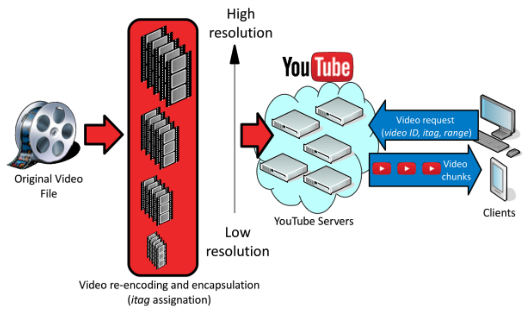


Figure 1-1 YouTube video service [8]

Each chunk is sent as a distinct web object via HTTP. The client continually checks available bandwidth during video playback and requests successful chunks for the specified data rate. Usually, the client stores a buffer of chunks in case of network difficulties (e.g. latency, packet loss, connection loss) [8]. To ensure that video files are compatible with all browsers, devices, bandwidth, and quality requirements, customers may choose from various encoding options for each video file [8]. In addition, numerical identification is known as "itag" is utilized to distinguish different video encoding techniques. It is sent along with the HTTP requests. Furthermore, since each chunk is sent individually as a distinct web object, [8] the client continually evaluates available bandwidth and requests chunks for the data rate sustained during video playback. To cope with future network difficulties (e.g., latency, packet loss, connection loss), the client tries to retain a buffer of video data; to do so, a buffering technique is required [8].

1.2.2 Wi-Fi Standard

The IEEE 802.11 Working Group2 created and maintained the IEEE 802.11 protocols [9].
That standard, IEEE 802.11a, has a theoretical maximum data rate of 54 Mbit/s. The frequency is around 5 GHz, which is unlicensed. Due to the 6cm-wavelength, it is easily absorbed by objects and walls. [9] However,802.11g Wi-Fi (2.4 GHz) is less readily absorbed. Currently, WLAN devices based on IEEE 802.11g technology provide data rates of 100-125Mbps [9]. IEEE 802.11n has a data rate of up to 540Mbps and works at 2.4 GHz and 5 GHz. The IEEE 802.11n standard, finally issued in 2009, aims to improve the MAC layer throughput over earlier standards. The IEEE 802.11n workgroup has been researching different upgrades to the physical and MAC layers to boost performance since its inception in 2002 [9]. Changes to signal encoding techniques, numerous antennas, smart antennas, and MAC protocols are only a few improvements [9]. To boost throughput, IEEE 802.11n included additional MAC

layer methods. The standard employs MIMO [9] and a variety of innovative modulation and coding methods to enhance data speeds. The standard has a fixed channel bandwidth of 20MHz, which aids backward compatibility with previous standards. A 40MHz channel is also available in IEEE 802.11n. Only four spatial streams utilizing a single 40 MHz-wide channel may reach data speeds of up to 600 Mbps [9].

Besides, "An Efficient Cooperative Retransmission MAC Protocol for IEEE 802.11n Wireless LANs" mentioned that 802.11n has frame aggregation mechanisms and block ACK mechanisms that can gather the frames together for enhancing the efficiency of throughput [19]. In the INET documentation, it said that 802.11n has MAC Service Data Unit (MSDU) Aggregation and MAC Protocol Data Unit (MPDU) Aggregation [20]. Furthermore, they showed that frame aggregation could enhance the performance of throughput.

1.2.3 Riverbed Modeler

The Riverbed Modeler Academic Edition provides a virtual environment for modeling, assessing, and forecasting the performance of information technology infrastructures, including applications, servers, and networking technologies[10]. Academic Edition, which is based on Riverbed's award-winning Modeler software, is intended to be used in conjunction with particular lab tasks that teach basic networking principles. Commercial versions of Modeler provide enhanced features that are intended to accelerate network research and development productivity, build proprietary wireless protocols and technologies, and assess modifications to standards-based protocols, among other things[10]. Riverbed software is utilized by thousands of commercial and government organizations throughout the globe and by more than 500 colleges and research institutions[10]. In addition to being designed for entry-level networking courses, the Modeler Academic Edition is also intended for use in the classroom. Incorporating tools for all stages of a project, Riverbed Modeler Academic Edition includes models for development, simulations for testing, data gathering, and data analysis[10].

1.2.4 Quality of service

The phrase 'Quality of Service' (QoS) refers to the ability of a network to measure its overall performance [11]. Our project mainly focuses on throughput and latency (packet delay) [12].

1.3 project overview

This project will select a mobile node as the core, and it aims to simulate “a user using a mobile to a watch YouTube video” under a series of different scenarios. Then analyze the performance of mobile node in each scenario mainly.
This project designs four main scenarios. The first scenario is to simulate the 1080P and 30 FPS video on YouTube at the **mobile** node and the fixed node (such as a desktop), then the throughput, latency, and retransmission of the two nodes will be compared. This object also aims to find the overall performance of video streaming on mobile in a relatively ideal environment, which will be regarded as the baseline for the followed scenarios.

The second scenario increases the number of fixed nodes to test video streaming performance when more users are on the network. This scenario simulates a YouTube video being watched in a library with some students present. In this circumstance, the project gives three sub-scenarios: two fixed nodes and one mobile node (Mobile node keeps the video browsing. Light browsing is used by one fixed node, while heavy browsing is used by the other). The report will call the first sub-scenario "mult11" in the followed contents.
Next, the total number of fixed nodes increases to four, and half of the fixed nodes are light browsing, and the rest are heavy browsing. Therefore, the report will call the second sub-scenario "mult22" in the followed contents.
The third sub-scenario is to increase the total size to eight and repeat the same distribution as the previous. Therefore, the report will call the third sub-scenario "mult44" in the followed contents.
The video streaming node's WLAN parameter, application, and profile remain unaltered. The light and heavy browsing parameters are the default parameters provided by Riverbed Modeler.

In the third scenario, the project experimented with various data speeds and Wi-Fi protocols to see how they affected the Video Streaming node's performance. This scenario consists of some sub-scenarios. Their topologies are similar to "mult22" in the scenario2 but different wireless LAN protocols. Therefore, the project selects 802.11g and 802.11n in this scenario.

The fourth scenario is based on the scenario3. Scenario4 will re-place the mobile far away from the router and add a trajectory. Scenario4 aims to find the maximum effective range of each wireless LAN protocol, what happens when the mobile goes back to the router's effective range, and the effect of moving and distance on video streaming performance.

1.4 Related Work

This section presents previous attempts at this problem. P. Casas [13] et al presented an analysis of the YouTube service performance in cellular networks from an end-user perspective, focusing on video flows download throughput and simplified QoE(Quality of Experience) metrics. Furthermore, they show that as end-users QoE improves, user engagement Increases. The YouTube download flow varies from 1.3Mbps (Low SNR flows) to 2.1Mbps (High SNR flows)[13].
S. Abdallah-Saleh et al in "Handover evaluation for mobile video streaming in heterogeneous wireless networks." [14] This study tested and evaluated the handover process and quality of service (QoS) of mobile video streaming in heterogeneous wireless networks from the network layer's perspective. It shows that handovers will cause a sharp drop in throughput, and a lower speed movement will have a better throughput for video streaming.
D. Jain [15] et al in “Prediction of quality degradation for mobile video streaming apps: A case study using YouTube” proposed a lightweight method for early detection of network capacity degradation. They provided some traffic patterns when facing different “bad quality” of the network, which helps analyze the performance of YouTube videos.

Michael Ng and Ching Ho Weng, in ‘Video Streaming over WiFi using Riverbed Modeler’ [16], built four scenarios to analyze the Average Throughput and Delay of video streaming on YouTube.
Amandeep Kaur, Haotian Ye, and Ashiv Rao Dhondea, in “Performance analysis of video streaming with WiFi” [12], tested four throughput and packet delay scenarios to see how WiFi performs video streaming.
From the two projects of Michael and Amandeep above, we learned some ideas about where to start to build the scenario and how to set the parameters. Besides, this report will thank “Performance analysis of video streaming with WiFi”[12] provide a guide on how to write a report.

Jay Kim, Jack Zheng, and Paniz Bertsch, in "Video Streaming over Wi-Fi," [21] built the scenario consisting of video streaming, VoIP, and web browsing. Then, they applied to video conferencing applications to substitute the role of video streaming. Besides, they test the influence of distance on video streaming performance first.

Sophia Calzada, Curtis Rietchel, and Tomasz Szajner, in their "Performance Analysis of a Wireless Home Network" [22], built the scenario consisting of video streaming, VoIP, gaming, and web browsing. Then they test the performance scenario with different data rates.

2. Design & Implementation

In this project, the critical core is the mobile node, all results and discussion will focus on the mobile performance, and fixed nodes are the supports in this project.

Note: Only mobile nodes will use the video browsing profile in Scenario2, Scenario3, Scenario4, so "video streaming performance" is equivalent to "video streaming performance of mobile" in this project

2.1. Scenario 1: Mobile node VS Computer (fixed) node

The network in Scenario1 contains two sub-scenarios:

1. A router, a server, and a mobile node. (aka, Mobile_single)
2. A router, a server, and a fixed workstation. (aka, Fixed_single)

The router is a "WLAN Ethernet router" connected to the "ethernet server" by a "1000BaseX Ethernet link". The Scenario1 is used to test some basic results with IEEE 802.11n (2.4G), and the data rate is set to 6.5Mbps/MAX 60Mbps. Next, we will simulate watching videos with 1080P and 30 FPS on YouTube. Another key idea is to compare video streaming performance with the mobile and fixed nodes.

Note: This Riverbed project started from "Wireless LAN Tutorial" provided by Prof Trajkovic [30].

The topologies are shown below:

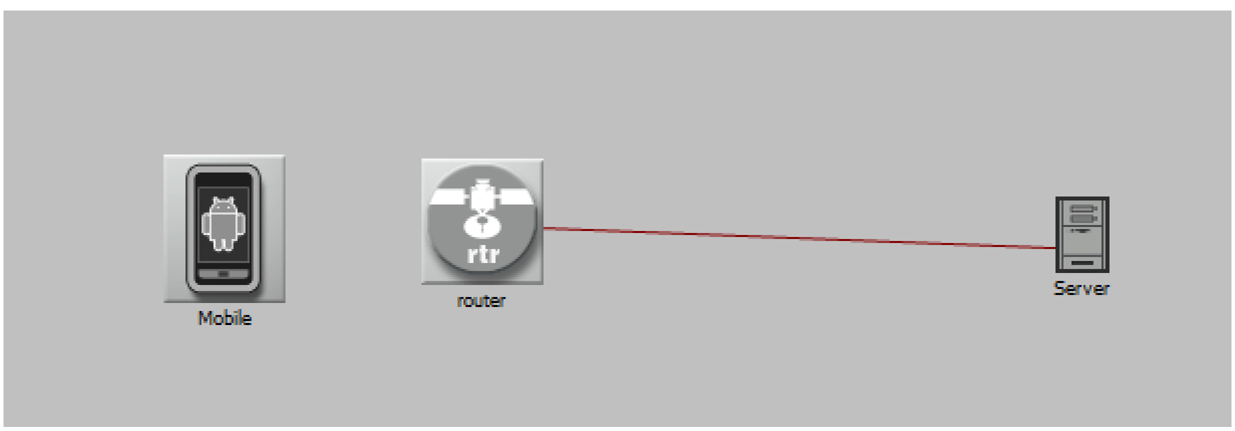


Figure 2-1. Topology with a mobile node.

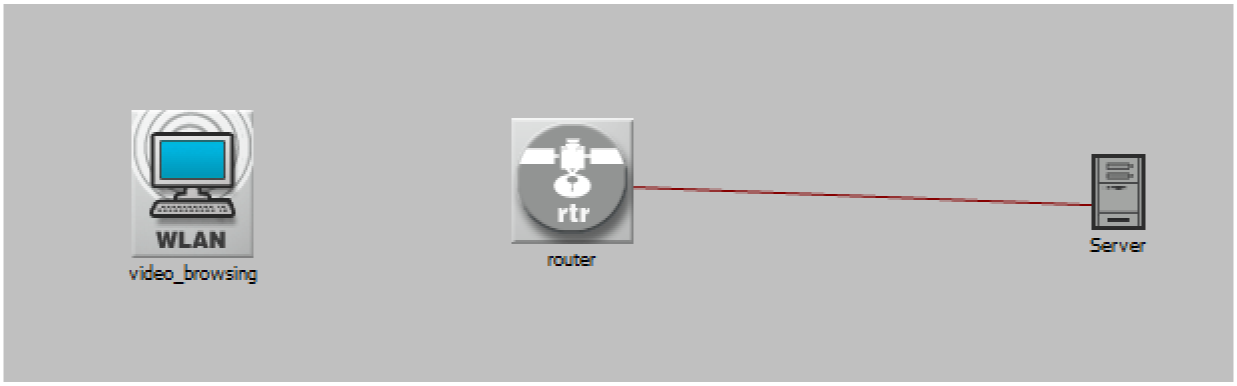
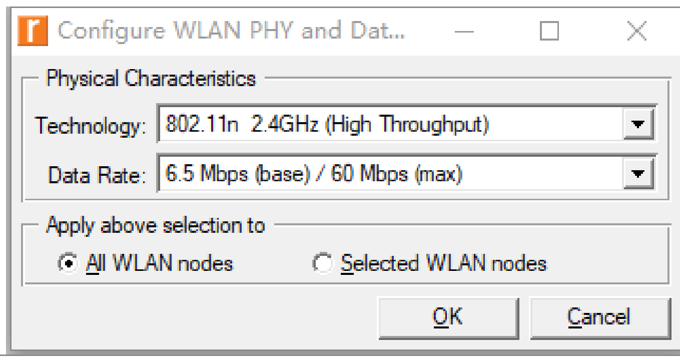


Figure 2-2. Topology with a fixed node.

Note: The application and profile are in the topologies, just not in the screenshots

The wireless LAN protocols of all nodes are set to “802.11n 2.4G - 6.5Mbps/60Mbps”. In addition, to ensure all the nodes are part of the same service set, the BSS Identifier for the router and node is set to 1.



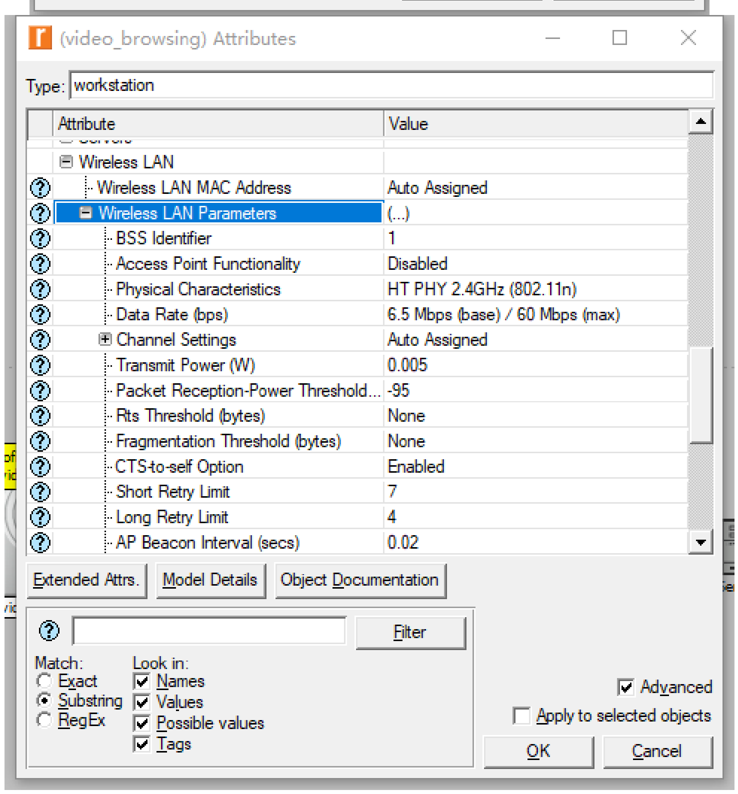


Figure 2-3. WLAN Configuration.

Here is the setting of the application:

The page Interarrival Time is used to simulate frames per second. According to the previous reports [12], videos on YouTube are usually 24 FPS to 30 FPS. We choose 30 FPS, so the ideal page Interarrival Time would be 1/30 = 0.0333s. Considering the potential frame drop, the FPS is set to uniform (29,30), that is “uniform(0.0333,0.0345)” for page Interarrival Time.

For simulating the video with 1080P and 30 FPS on YouTube:

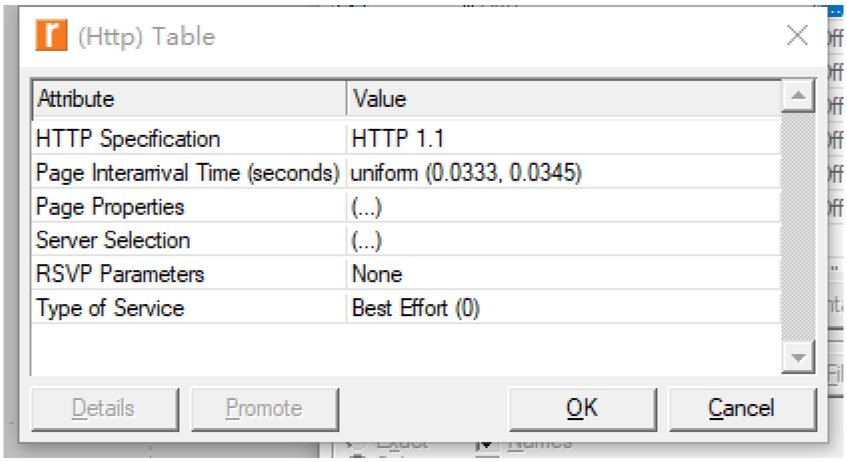


Figure 2-4. Setting of the Video Streaming application.

Wireshark was used to sniff the packets while streaming a 1080p YouTube video with a Wi-Fi connection. As is shown in fig 2-5, the packets are transferred to the user’s computer from a local web cache rather than the original server of YouTube. (Since the host of “2001:569:2:8::e” is “cache.google.com” in Burnaby.) The packets are transferred on UDP protocol with a standard packet size of 1230 bytes. Hence, we define the object size as constant (1230) bytes.

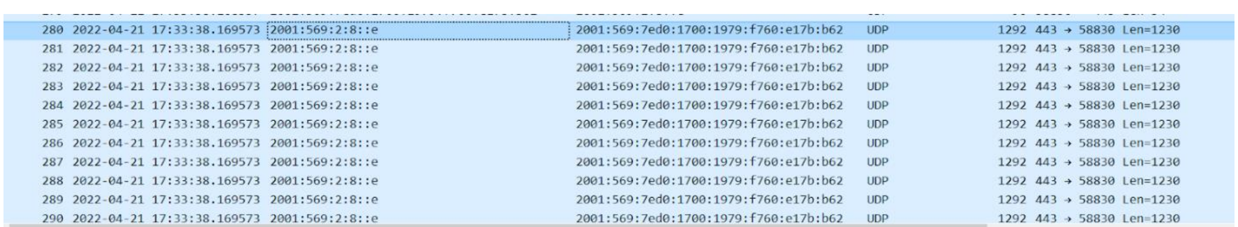


Figure 2-5. Wireshark trace of YouTube Video.

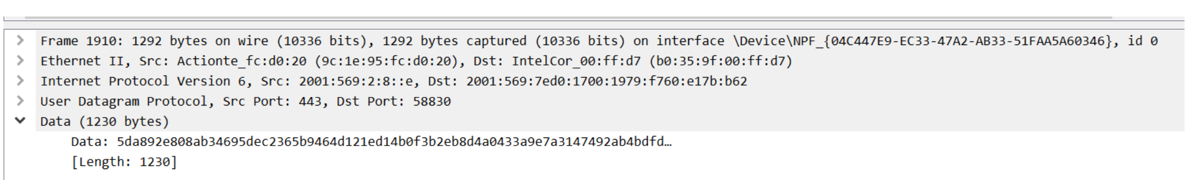


Figure 2-6. Details of a YouTube packet.

According to YouTube Help [17], to watch a video with good quality, the recommended continuous speeds for different resolutions are 20Mbps for 4K, 5 Mbps for 1080p, and 2.5 Mbps for 720p. So, we choose 5Mbps for this project.

Therefore, with the video bitrates and packet size, we can calculate the parameter of Objects per page in the following way:

5Mbps = 625 KBps.

In 30 FPS, 625 KBps = 20833.33 = 20834 byte/frame, 20834/1230 = 16.93 = 17 packets/frame.

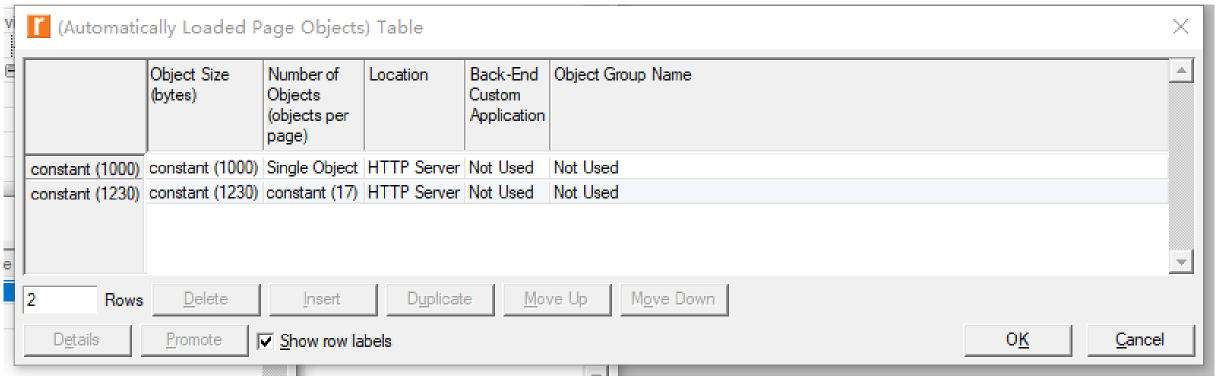


Figure 2-7. Parameters of automatically Loaded Page Objects

In the profile, the operation mode is set to be "Simultaneous," and the total duration of video_http_profile will be the whole simulation process.

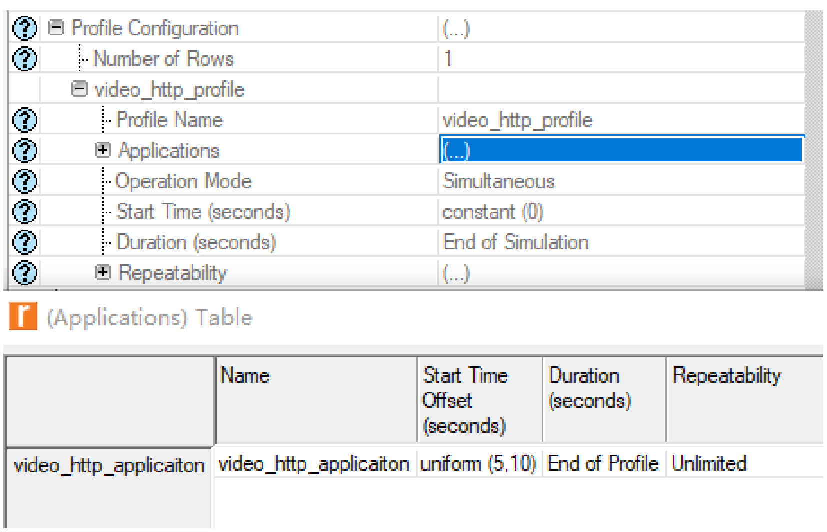
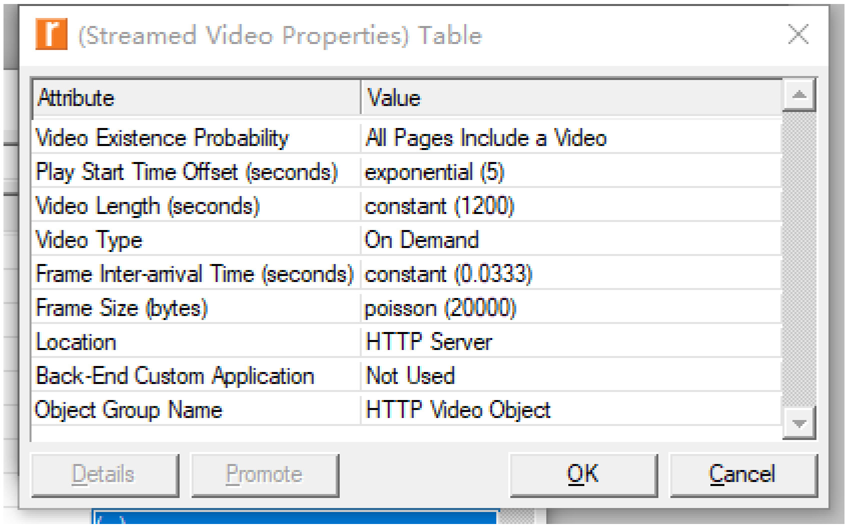


Figure 2-8. Profile setting

Finally, Add the video_http_profile to the mobile node and the fixed node. Add the video_http_application to the server node.

Aside:

Here is the setting of Streamed Video Properties:



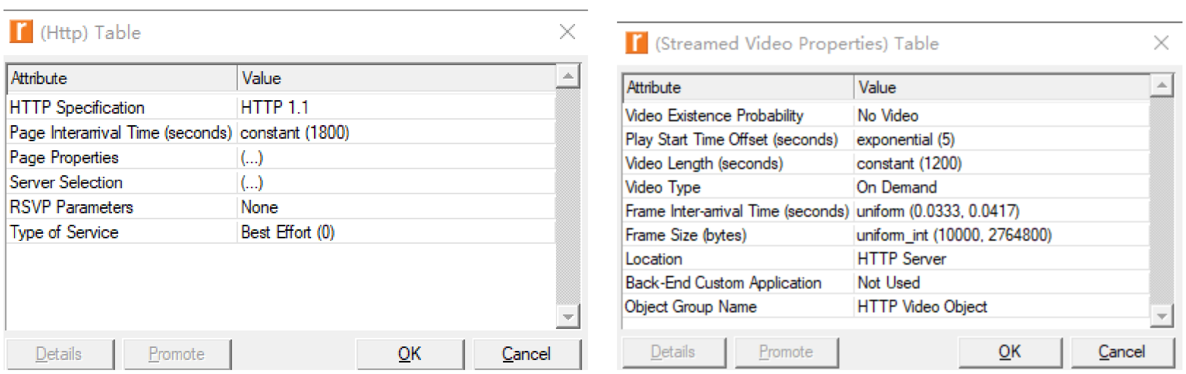
(Streamed Video Properties) Table

Attribute	Value
Video Existence Probability	All Pages Include a Video
Play Start Time Offset (seconds)	exponential (5)
Video Length (seconds)	constant (1200)
Video Type	On Demand
Frame Inter-arrival Time (seconds)	constant (0.0333)
Frame Size (bytes)	poisson (20000)
Location	HTTP Server
Back-End Custom Application	Not Used
Object Group Name	HTTP Video Object

Figure 2-9. setting of Streamed Video Properties

However, it seems this attribute will not influence the throughput and delay. We tested and compared the results of "All pages include a video" and "No video," finding no difference. Hence, we banned this attribute and redefined the frame size in Automatically Loaded Page Object.

Here are the test attributes and their results:



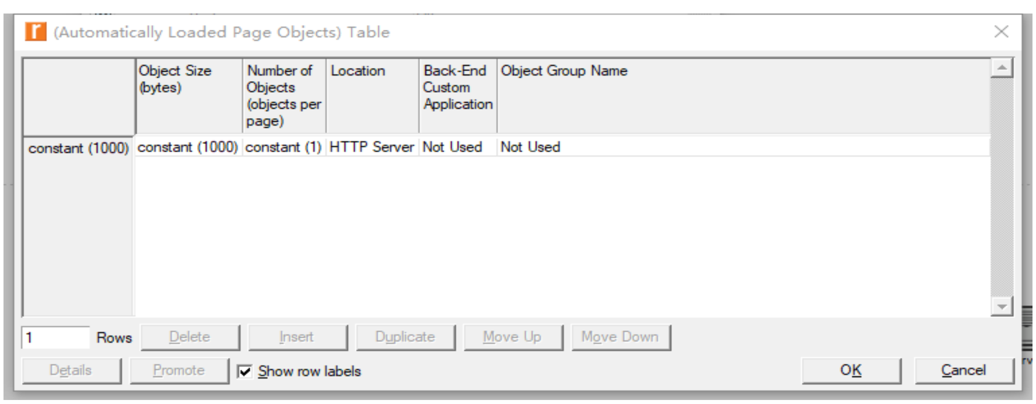
(Http) Table

Attribute	Value
HTTP Specification	HTTP 1.1
Page Interarrival Time (seconds)	constant (1800)
Page Properties	(...)
Server Selection	(...)
RSVP Parameters	None
Type of Service	Best Effort (0)

(Streamed Video Properties) Table

Attribute	Value
Video Existence Probability	No Video
Play Start Time Offset (seconds)	exponential (5)
Video Length (seconds)	constant (1200)
Video Type	On Demand
Frame Inter-arrival Time (seconds)	uniform (0.0333, 0.0417)
Frame Size (bytes)	uniform_int (10000, 2764800)
Location	HTTP Server
Back-End Custom Application	Not Used
Object Group Name	HTTP Video Object

Figure 2-10. Detailed settings of Streamed Video Properties



(Automatically Loaded Page Objects) Table

	Object Size (bytes)	Number of Objects (objects per page)	Location	Back-End Custom Application	Object Group Name
constant (1000)	constant (1000)	constant (1)	HTTP Server	Not Used	Not Used

Figure 2-11. Settings of automatically Loaded Page Objects

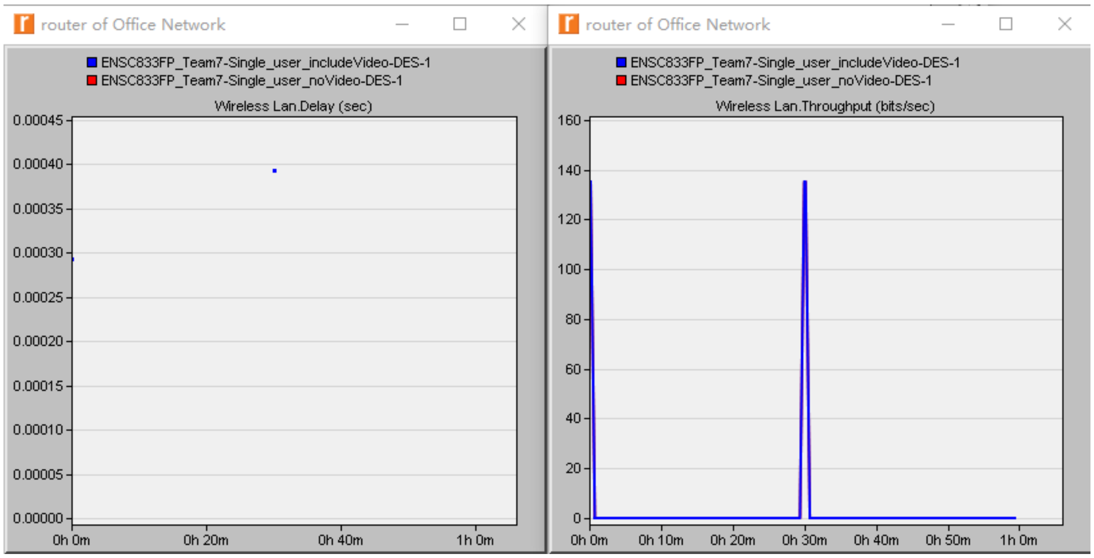


Figure 2-12. Delay and Throughput of noVideo and includeVideo

As is shown above, the plots overlap.

We define a 20-minute 720P 30 FPS video in streamed video properties in the attributes above. However, the results of "All pages include a video" and "No video" look the same.

Note: this is not a part of the scenario, so we show it here.

2.2. Scenario 2: Video streaming of Mobile with other users

In this scenario, we add more fixed nodes to find video streaming performance of mobile when there are more users in the same network. This scenario is the simulation of watching a YouTube video with some students in the library. We have three sub-scenarios in this scenario:

1. One mobile node and two fixed nodes. One fixed node uses light browsing, and the other uses heavy browsing. (aka, mult11)
2. One mobile node and four fixed nodes. Two for light browsing and the other two for heavy browsing. (aka, mult22)
3. One mobile node and eight fixed nodes in total. Four for light browsing and the other four for heavy browsing. (aka, mult44)

The wireless LAN parameter, application, and profile for the video streaming node (mobile) remain unchanged. Here are the topologies.

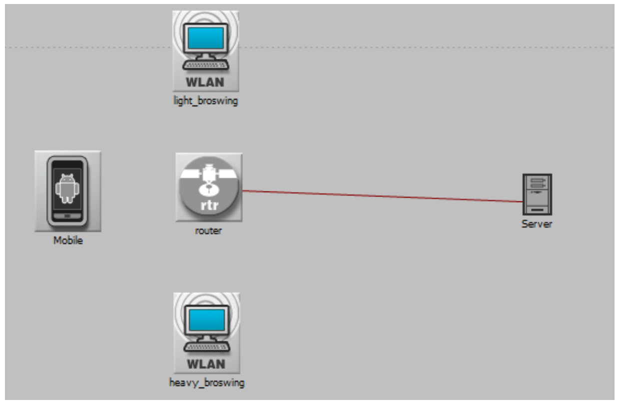


Figure 2-13. Case 1: One node of light browsing and one node of heavy browsing

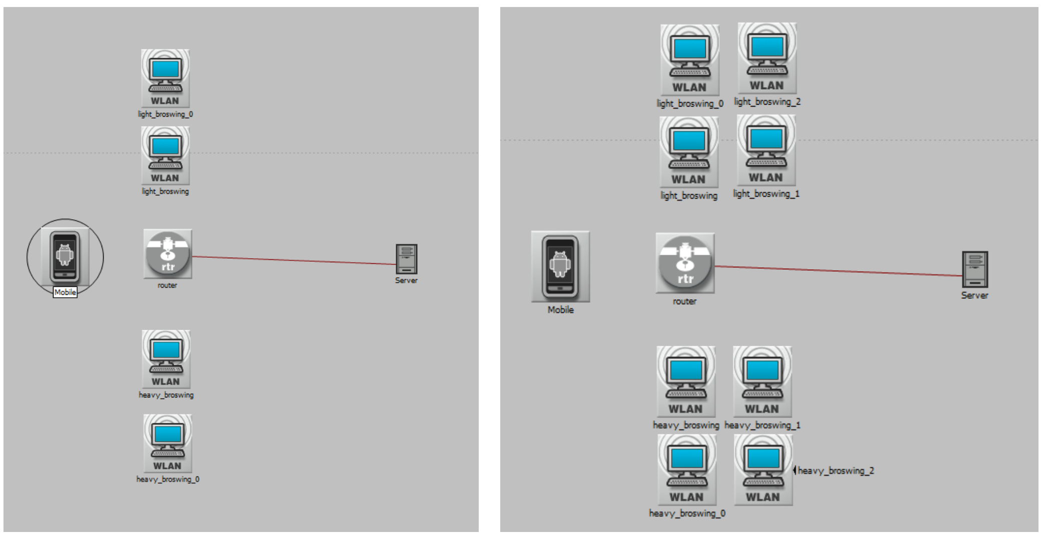


Figure 2-14. Case 2: Two nodes of light browsing and two nodes of heavy browsing; Case 3: Four nodes of light browsing and four nodes of heavy browsing

Since the application of light browsing and heavy browsing are not the focus of this report, we use the default settings in Riverbed. The object size is 500 bytes with just one object on a page or a "small Image" containing five objects per page in light browsing. This scenario simulates someone reading some news or articles with some small images. There will not be too much traffic with the page Interarrival Time set to exponential 720s.

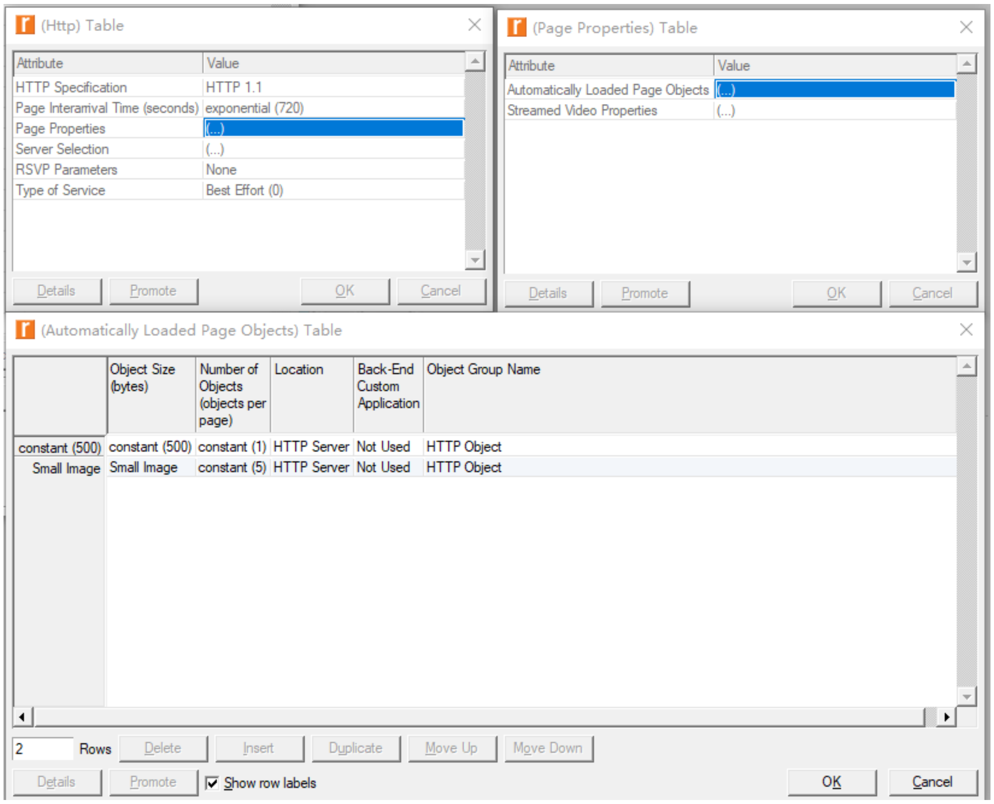


Figure 2-15. parameters of light browsing

In heavy browsing, the object size is larger as it contains five “Medium Images” and two “Short Videos.” In addition, the page Interarrival Time is set to exponential 50s to produce more traffic.

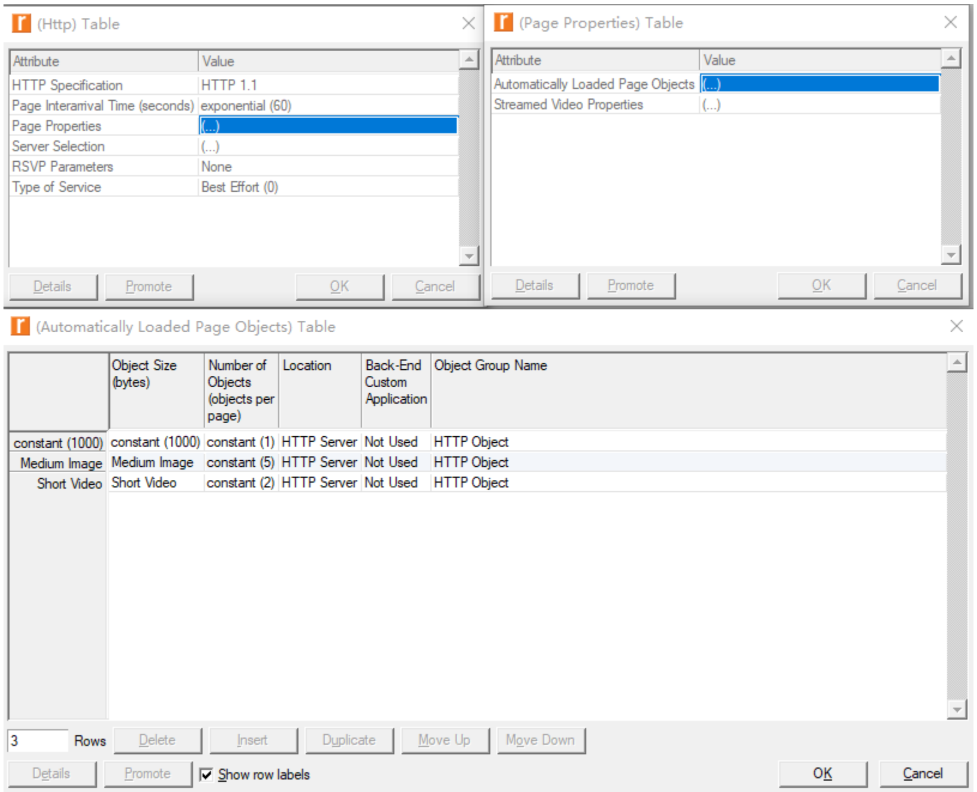


Figure 2-16. parameters of heavy browsing

The profile settings are like video browsing. Next, add the new applications to "application: supported service" on the server and assign the light profile and heavy profile to the respective nodes. The configuration of video browsing is the same as in scenario 1.

2.3. Scenario 3: Effect of data rate and Wi-Fi protocol

In this scenario, we tested different data rates and Wi-Fi protocols to find the influence on the performance of the Video Streaming node. We will use the second topology in Scenario2 (**mult22**):

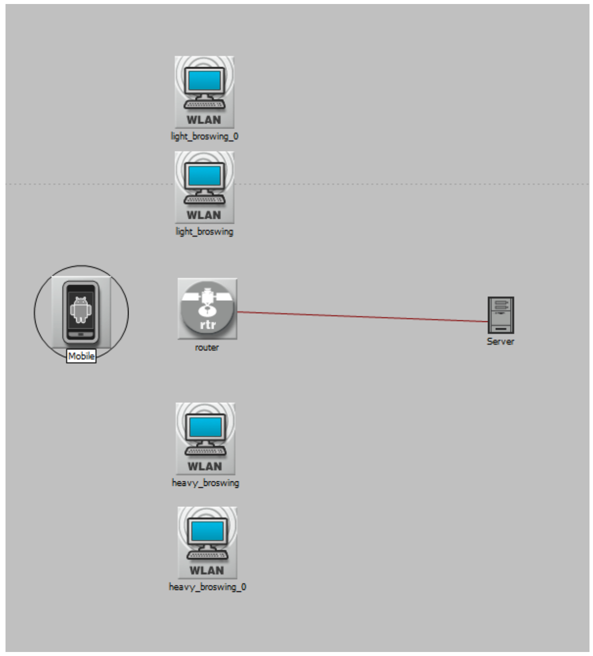


Figure 2-17. Topology for scenario 3.

In this scenario, we choose “mult22” (two light browsing nodes and two heavy browsing nodes) over “mult44” because of the limitation of the total event number in Riverbed Modeler Academic. Once the event number reaches 50,000,000, the simulation will be terminated automatically, which leaves around 7 minutes for the simulation with “mult44”. To get enough data, we trade time with the number of nodes.

Since 801.11g and 802.11n are the most commonly used protocol, we test 6 different data rates from each protocol. Here are the data rates we choose:

802.11g (Mbps)	802.11n (Mbps)
12	13
18	19.5
24	26
36	39
48	52
54	58.5

Table 1: Data rates

The video streaming (video browsing), heavy browsing, and light browsing application settings remain the same as in Scenario2.

2.4. Scenario 4: Effect of distance and movement on data rate and Wi-Fi protocol

This scenario aims at simulating someone walking while watching a video. We mainly focus on these three aspects:

1. The influence of the distance between the mobile node and the router.
2. The different ranges of different protocols and data rates.

3. The reactions of nodes after backing into the effective distance for different protocols and data rates.

The article "How far will your Wifi signal reach?" shows that "Wi-Fi routers operating on the traditional 2.4 GHz band reach up to 150 feet (46 m) indoors and 300 feet (92 m) outdoors."[18] However, changing the range of a router is not permitted in Riverbed Modeler Academic. To achieve our goal, the node must be 1000 meters away from the router, which is close to the range of 1200 meters.

Note: Since we do not know and cannot set the router's effective range, we tested 200, 400, 600, 800, 1000, and 1200 meters with 802.11n/6.5Mbps, and only 1200-meter result showed that the mobile was out of the effective range.

We add a trajectory to the mobile. Scenario4 employs the same WLAN parameter, application, and profile settings as Scenario3. There are two heavy browsing nodes and two light browsing nodes near the router as in Scenario3.

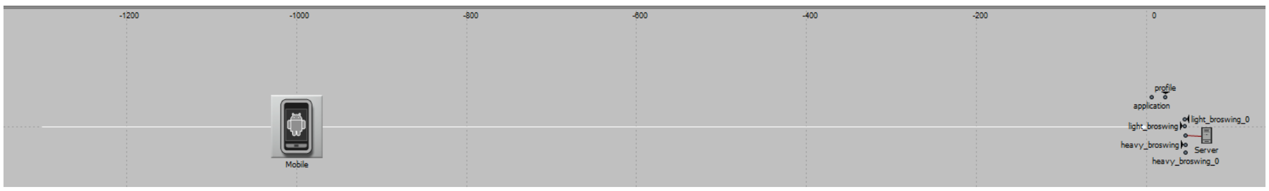
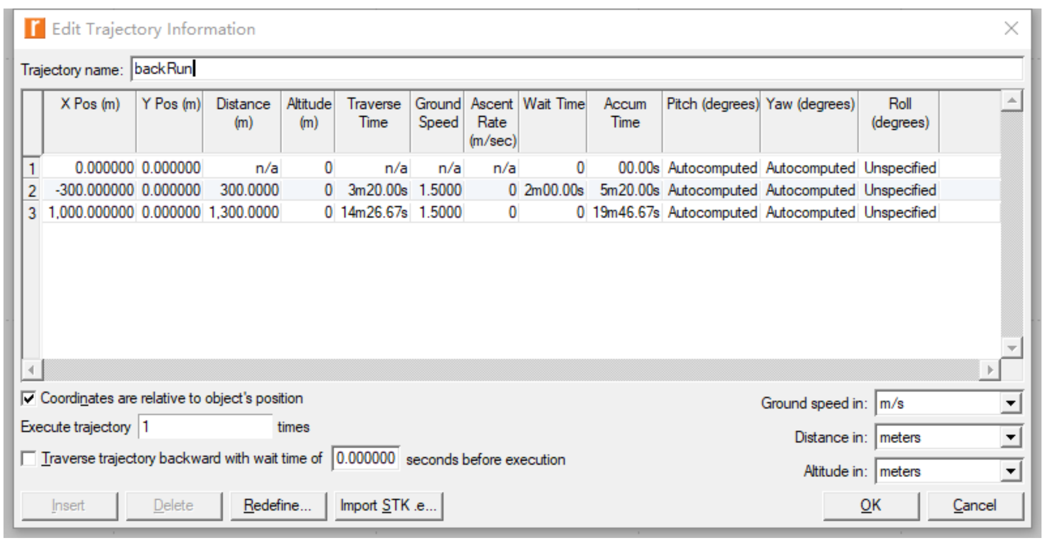


Figure 2-18. Topology for scenario 4.

Here are the trajectory data:



Edit Trajectory Information

Trajectory name: backRun

	X Pos (m)	Y Pos (m)	Distance (m)	Altitude (m)	Traverse Time	Ground Speed	Ascent Rate (m/sec)	Wait Time	Accum Time	Pitch (degrees)	Yaw (degrees)	Roll (degrees)
1	0.000000	0.000000	n/a	0	n/a	n/a	n/a	0	00.00s	Autocomputed	Autocomputed	Unspecified
2	-300.000000	0.000000	300.0000	0	3m20.00s	1.5000	0	2m00.00s	5m20.00s	Autocomputed	Autocomputed	Unspecified
3	1,000.000000	0.000000	1,300.0000	0	14m26.67s	1.5000	0	0	19m46.67s	Autocomputed	Autocomputed	Unspecified

☑ Coordinates are relative to object's position

Execute trajectory 1 times

☐ Traverse trajectory backward with wait time of 0.000000 seconds before execution

Ground speed in: m/s

Distance in: meters

Altitude in: meters

Insert | Delete | Redefine... | Import STK .e... | OK | Cancel

Figure 2-19. Setting of trajectory.

At the beginning of the scenario, the mobile node is placed to the left of the router at a distance of 1000 meters from the router. It first moves 300 meters to the left and stays in place for two minutes, and this step aims to find the maximum effective distance of the current wireless LAN protocol.

Next, the mobile walks to the right until the end of the simulation. The speed is set to 1.5m/s all the time. In this step, the mobile will connect to the router again at one moment. Then the mobile will keep walking to close the router. After that, we can find what happens when the mobile reconnects the router and the effect of the distance between the node and router.

3. Result and Discussion

Scenario 1: Mobile node VS Computer (fixed) node

This scenario is designed to find the baseline performance of video streaming on a mobile node and compare the performance of a mobile node and a fixed node.
Here is the comparison of one mobile node and one fixed node:

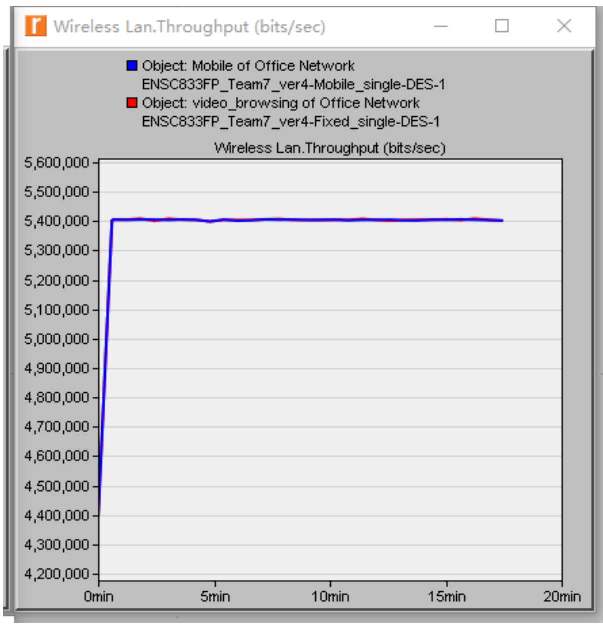


Figure 3-1 Throughput of mobile and fixed nodes

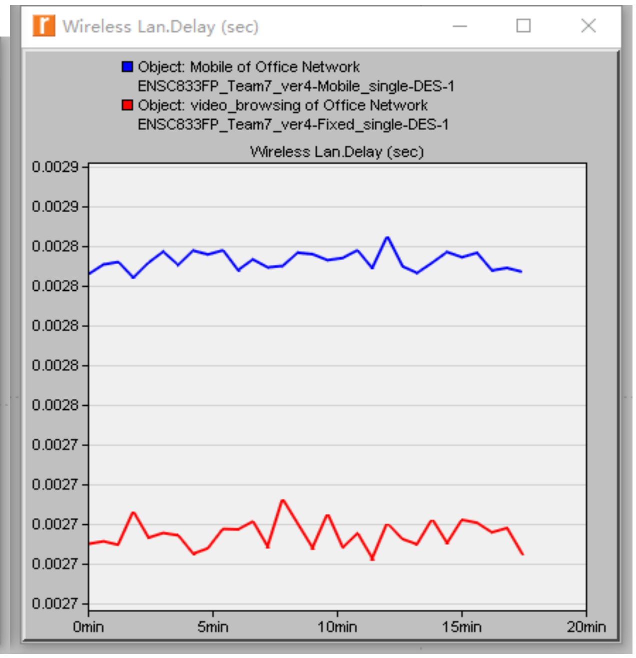


Figure 3-2 Delay of mobile and fixed nodes

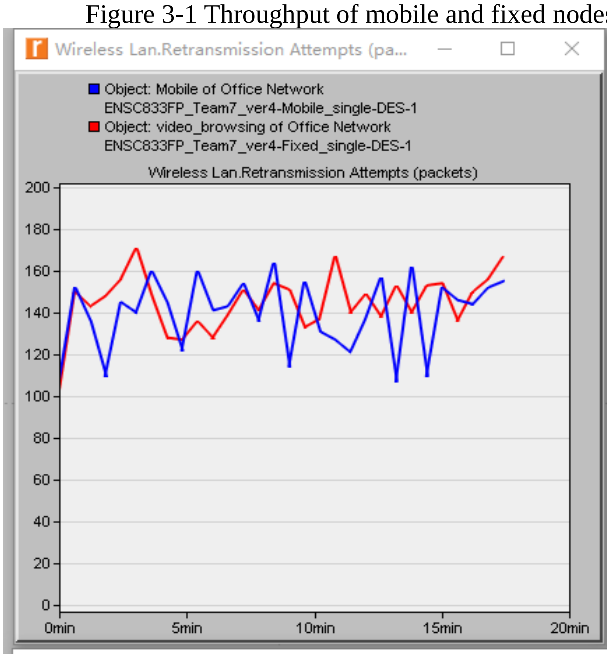


Figure 3-3 Retransmission of mobile and fixed nodes

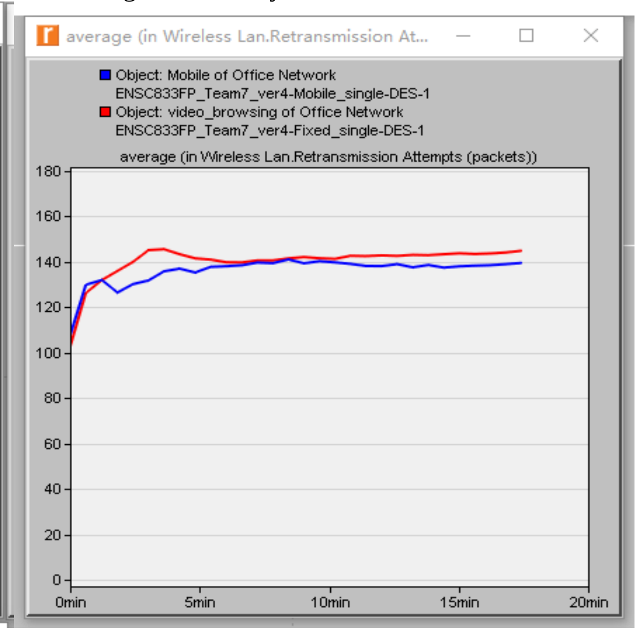


Figure 3-4 Retransmission(avg) of mobile and fixed nodes

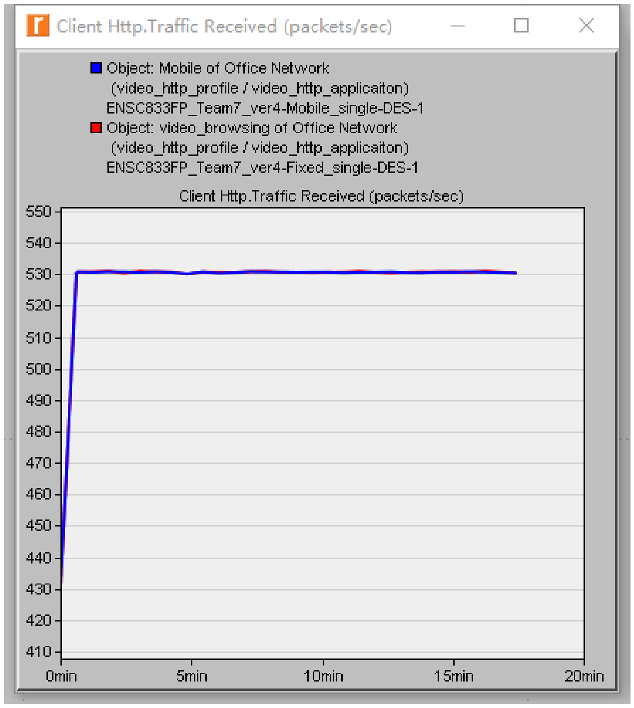


Figure 3-5 Traffic received from mobile and fixed nodes (packets/sec)

Scenario1 is designed to be an ideal environment where only one node in the network and only one video streaming application is used. The setting of the video streaming application is uniform [0.0333,0.0345] second for the page interarrival time and 17 packets with 1230 bytes per frame. Therefore, the overall throughput trend is stable at about 5.4Mbps, with only slight fluctuations (Figure 3-7 shows the detailed throughput performance, and its range is around 10000 bits, which is small for 5.4Mbps). Its delay is around 0.0028 seconds, and the retransmission attempts are about 100-160 packets, as shown in Figure 3.3.

In this case, 5.4Mbps means the maximum throughput (100% efficiency) of the mobile with this page property under this WLAN protocol (802.11n/6.5Mbps). Therefore, it will be regarded as the baseline for the followed scenarios. Similarly, the mobile baseline also contains delay (0.0028 sec) and retransmission attempts (100-160).

Compared to the mobile node, the fixed node has almost the same performance of throughput and traffic received as mobile. The fixed node has better performance in delay, and it has 0.0027 seconds. Both of their delays are good enough for watching a video. The retransmission attempts of fixed node are around 100-170 in Figure 3.3

In figure 3.4, we can see that the average retransmission attempts of the fixed node are higher than mobile overall. In chapter 1 of the lecture, the transmission may generate transmission delay [26]. The retransmission attempts may let the server re-send the packet, leading to a delay (spend more time). In this scenario, retransmission is not the main factor for delay (because mobile has a higher delay and fewer retransmission attempts)

Besides, according to Figures 3-6 and 3-7 below, the shape of throughput and traffic received (packets) in mobile are the same (regardless of value), also because of constant object size and numbers. Therefore, in this case, the analysis of throughput and packet are the same. Hence, the traffic received (packets) will not join the discussions of followed scenarios.

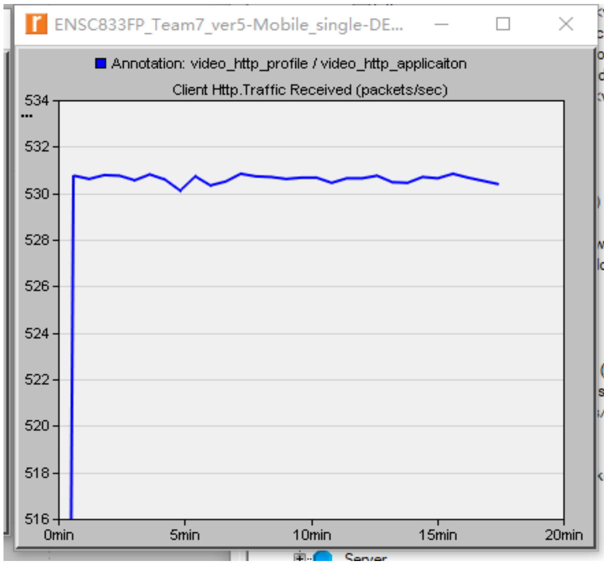


Figure 3-6 Traffic received from mobile

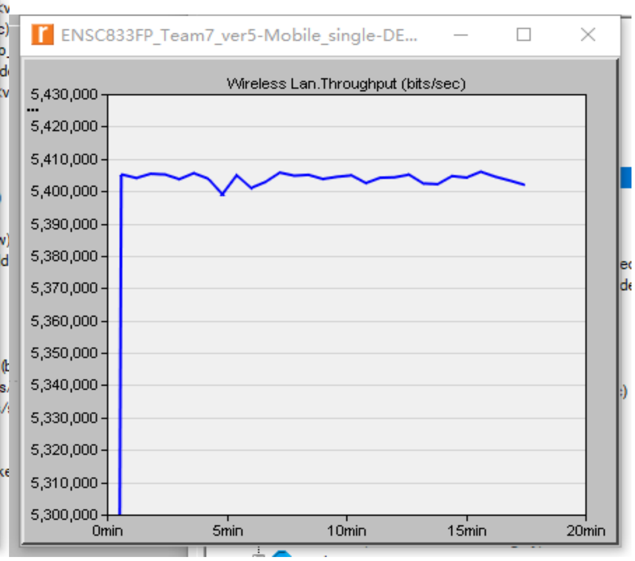
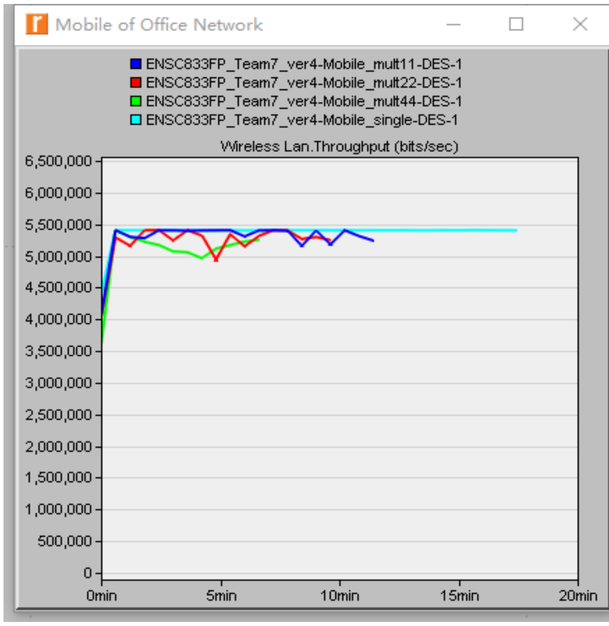


Figure 3-7 Throughput of mobile

Hence, the mobile's baseline data in this project are:
5.4Mbps throughput, 100-160 retransmission attempts, and 0.0028 seconds' delay.
Mobile's throughput performance is the same as the fixed node's.
Mobile's average retransmission attempts are fewer than the fixed node's.
Mobile's delay performance is worse than the fixed node's.
The following scenario will add more fixed nodes with light and/or heavy browsing to see what happens to the mobile node's performance.

Scenario 2: Video streaming of Mobile with other users

This scenario compares the mobile performance of mult11(one video streaming mobile node with one heavy browsing node and one light browsing node), mult22, mult44, and single mobile. (four sub-scenarios totally)
The result of single mobile comes from mobile performance in scenario 1.
This scenario tries to find the effect when there are more nodes around the mobile
Here are the results of throughput (figures 3-8, and 3-9) and one group of samples of heavy and light browsing (figures 3-12, and 3-13):



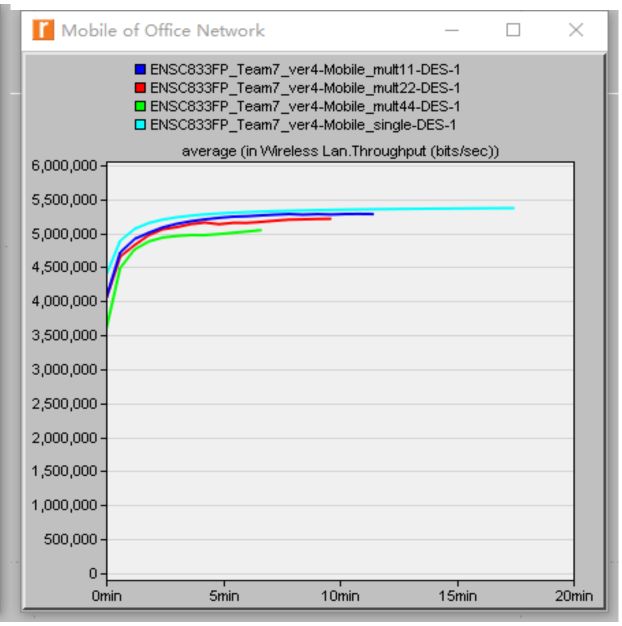
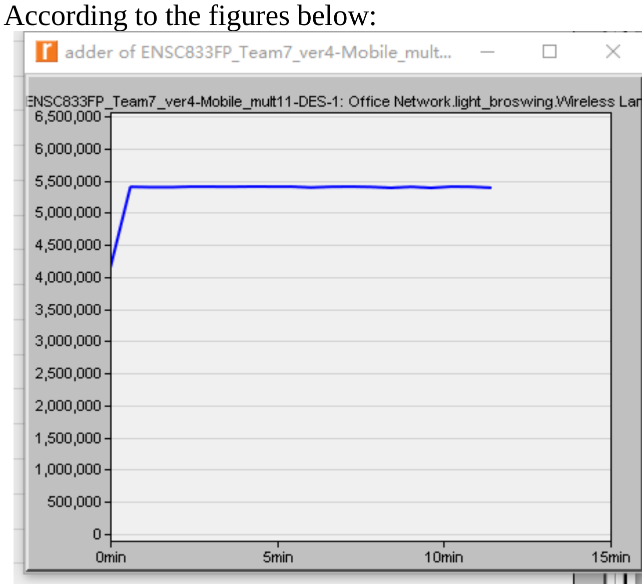


Figure 3-8 Throughput of scenario 2 Figure 3-9 Throughput average of scenario 2

According to the figures 3-8 and 3-9, when there is only one node in the network (Mobile_single), the throughput value stays stable at around 5.4 Mbps. When more nodes are added (from mult11 to mult44), the value of throughput starts to fluctuate. The graph of average throughput shows that the more nodes are joined, the smaller the throughput is.

According to the figures below:



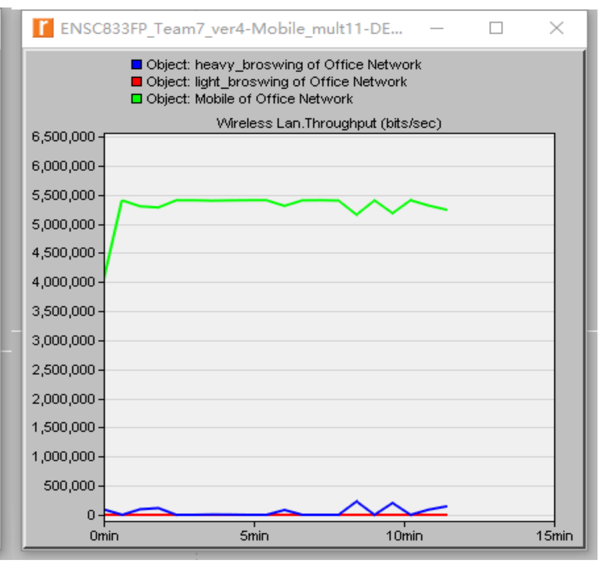


Figure 3-10 Throughput sum of mult22 Figure 3-11 Throughput of each node in mult22

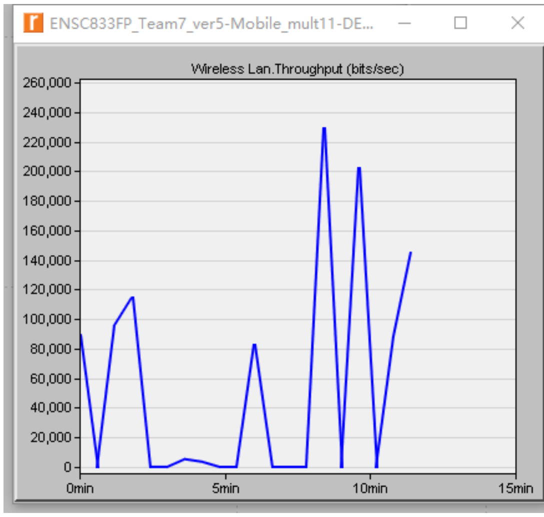


Figure 3-12 Throughput of heavy browsing

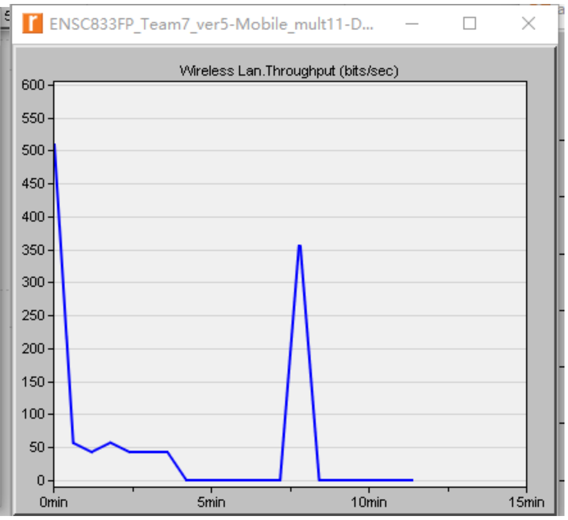


Figure 3-13 Throughput of light browsing

According to the Figures 3-10 and 3-11, we find that the reason for the decline in Mobile throughput is that the throughput is partially split by heavy browsing and light browsing users and while they still sum to 5.4Mbps, as shown in Figure 3-10**.**

Note: The figures may not display clearly, but it is indeed 5.4Mbps.

Due to processing the packets for other users, the efficiency of processing mobile is influenced. For example, here are the figures of single mult11 below:

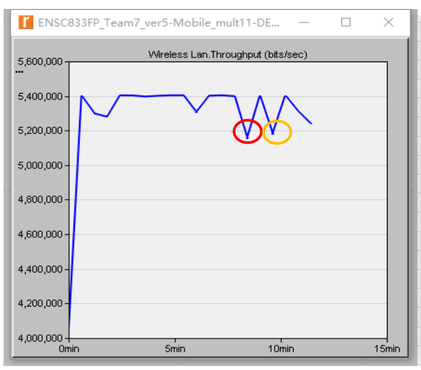


Figure3-14 Mobile throughput of mult11

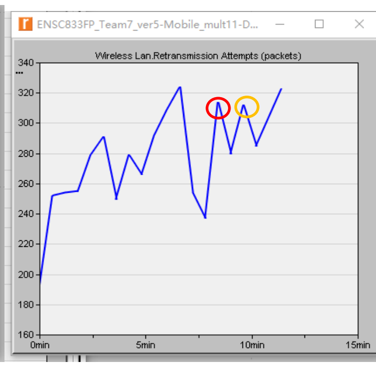


Figure3-15 Mobile retransmission of mult11

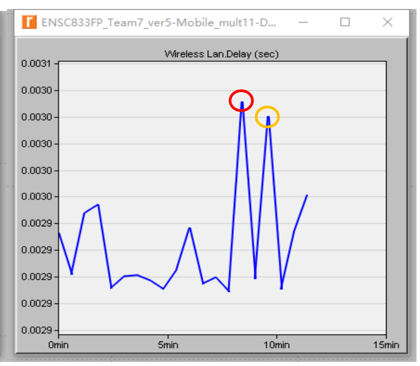


Figure3-16 Mobile delay of mult11

It shows that when the decline of throughput occurs, the value of retransmission attempts, and delays increases around the same time.

Here are the results of delay and retransmission attempts in this scenario:

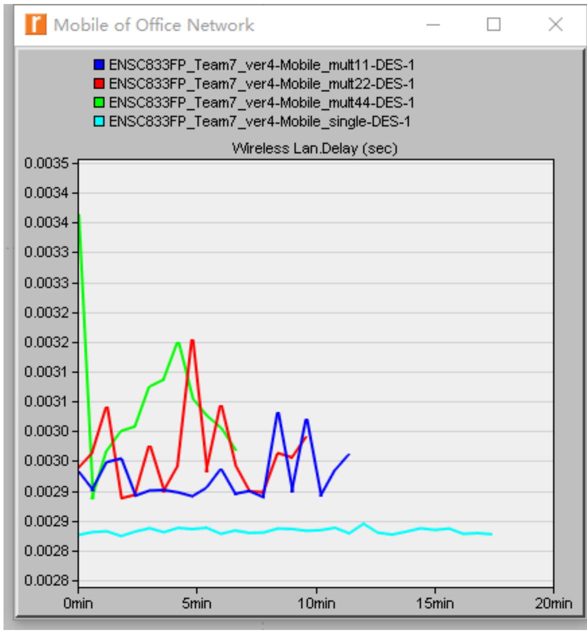


Figure3-17 Delay of scenario 2

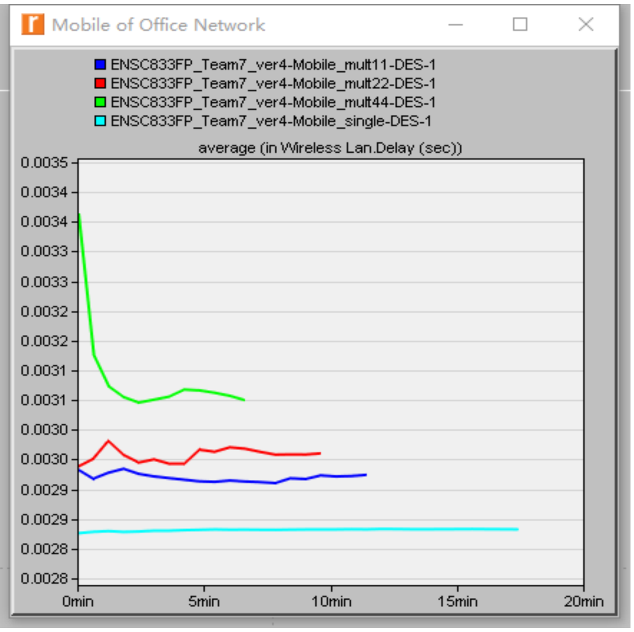


Figure3-18 Delay Average of scenario 2

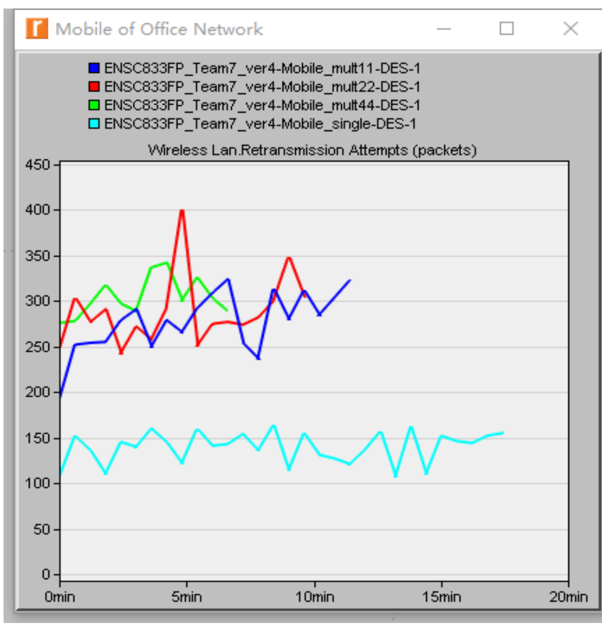


Figure 3-19 Retransmission of scenario 2

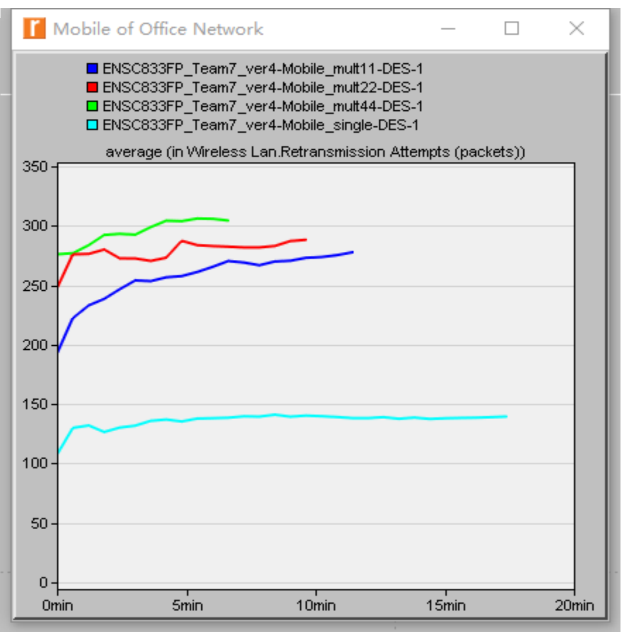


Figure 3-20 Retransmission average of scenario 2

In the lecture of chapter 3 [28], Prof Trajkovic described one case of congestion: When the buffer is large enough, the higher the sending rate, the higher the delay (exponential growth), the lower the throughput, and the lower the efficiency [28].

Although the project did not set an infinite buffer, the case of congestion may help to analyze the performance of throughput, delay, retransmission, and their relationship in this scenario.

As shown in Figures 3-14, 3-15,3-17, and 3-18, when the router must process packets for other users, it will quickly lead to fewer packets arrive the Mobile node in time. The mobile may resend more packets, which further reduces efficiency [28].

As shown in Figures 3-19 and 3-20, the delay increases because the router needs to process other packets in the delay part. Then, other packets generated by retransmission may increase the delay further.

More retransmission attempts may lead to higher delay, and higher delay may make mobile increase the number of retransmission attempts. They may influence each other and reduce the efficiency overall [28]. After that, when there are more other packets (more users) that need to process, the delay and retransmission attempt also increase.
Hence, the average retransmission attempts, and delays of mobile are increasing with the increase of users.
Besides, the decline in mobile throughput also represents that traffic is beyond the capacity of the current WLAN protocol (6.5Mbps). Therefore, it cannot keep a stable continuous data transmission in such a case. There are two methods to deal with this problem. The first way is to reduce the total packets, namely, fewer users. The second way is to enhance the performance of the WLAN protocol so that it can deal with more packets in unit time. Therefore, we will test different WLAN protocols with various data rates in the following scenario and compare their performance.
According to the figures (such as figures 3-8) above, the simulation time decreases as the number of users rises. The Riverbed Modeler Academic version will terminate the simulation once the events number is over 50,000,000, so the earlier the termination time, the more events are processed per unit time. (the more events with more users added)

Hence, the mobile's overall performance becomes worse with the increasing of nodes (especially the "heavy browsing" client). The next scenario will keep mobile with 2 light browsing nodes and 2 heavy browsing nodes and pursue to change the data rate and wireless LAN protocol.
Note: the "overall performance" in this project includes throughput, average retransmission attempts, and delay.

Scenario 3: Effect of data rate and Wi-Fi protocol

We select 802.11g and 802.11n with different data rates in this scenario and compare their performance. The base topology of Scenario3 is that the mult22 comes from Scenario2. This scenario assigns different data rates with different wireless LAN protocols to **mult22**. As a result, this scenario consists of six sub-scenarios for 802.11g, six sub-scenarios for 802.11n, and **mult22** (13 sub-scenarios in total).
Note: Only sub-scenarios within the same protocol will compare data rates. Different protocols will compare overall performance
Here are the results of throughput of each protocol:

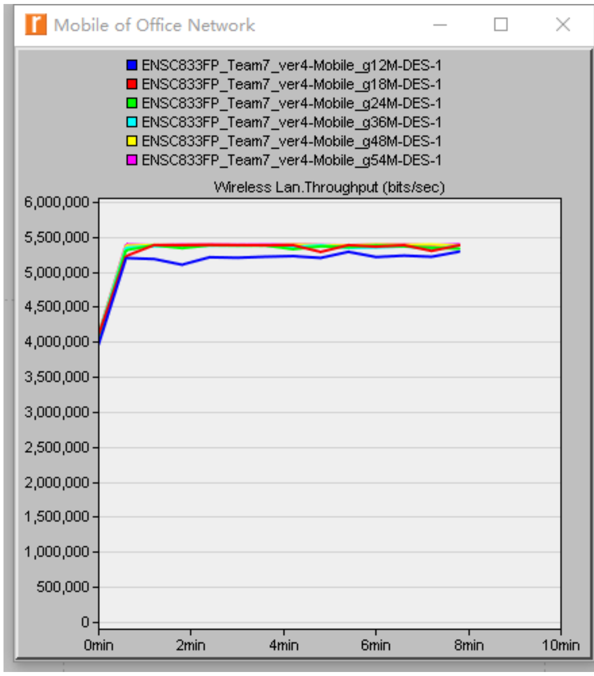


Figure 3-21 Throughput of 802.11g

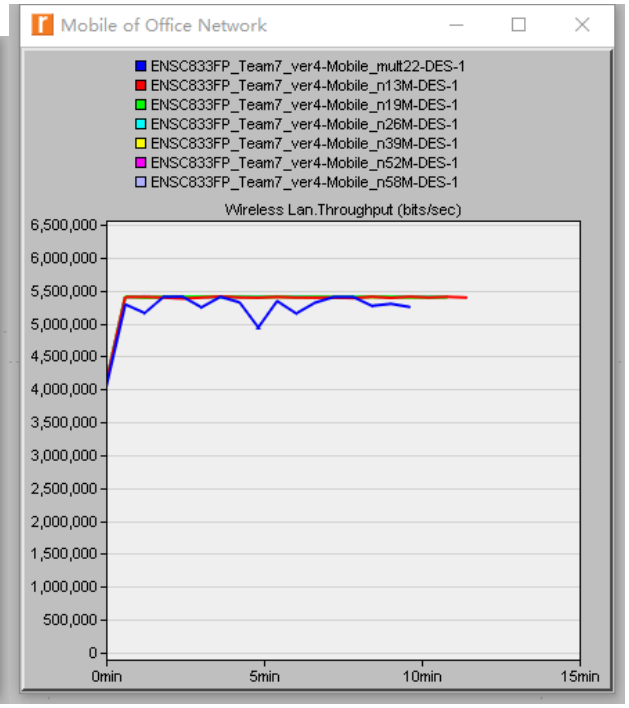


Figure 3-22 Throughput of 802.11n

In terms of termination time (such as figures 3-21 and 3-22), 802.11g usually stops running within 8 minutes, while 802.11n can run for about 12 minutes (except 6.5Mbps), which means 802.11g processes more events than 802.11n per unit time.
The paper [19]and documentation [20] said that 802.11n could have higher throughput using the MIMO and frame aggregation technique.

scenario 3.1 throughput part

As is shown in Figures 3-21 and 3-22, the general trend of throughput of 802.11g is "The higher the data rate, the higher the throughput." However, in 802.11n, almost all data rates maintain around 5.4Mbps throughput (except 6.5Mbps), which matches what we discussed in scenario 1, namely that the 5.4Mbps may be the maximum throughput (100% efficiency) the mobile can reach.

More detailed throughput figures 3-23 and 3-24 also represent that 802.11n/19Mbps is good enough to deal with a mobile node, two light browsing nodes, and two heavy browsing nodes. Only 48Mbps and 54Mbps for 802.11g to meet the exact requirement.

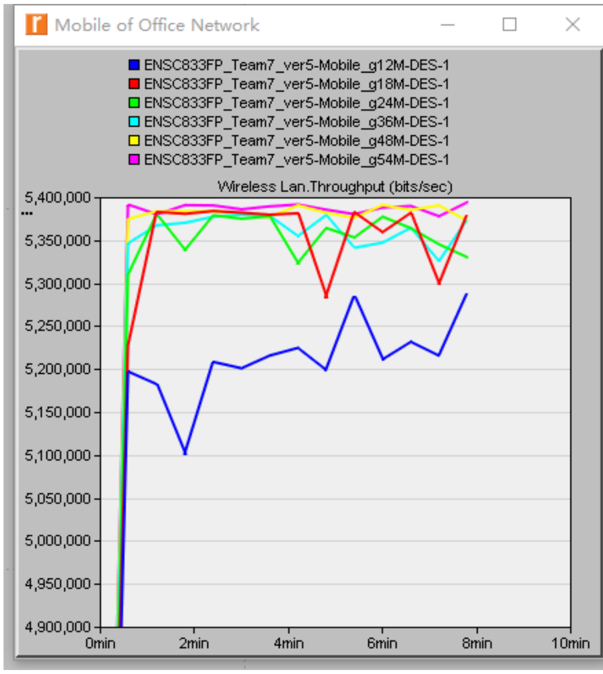


Figure 3-23 Detailed throughput of 802.11g

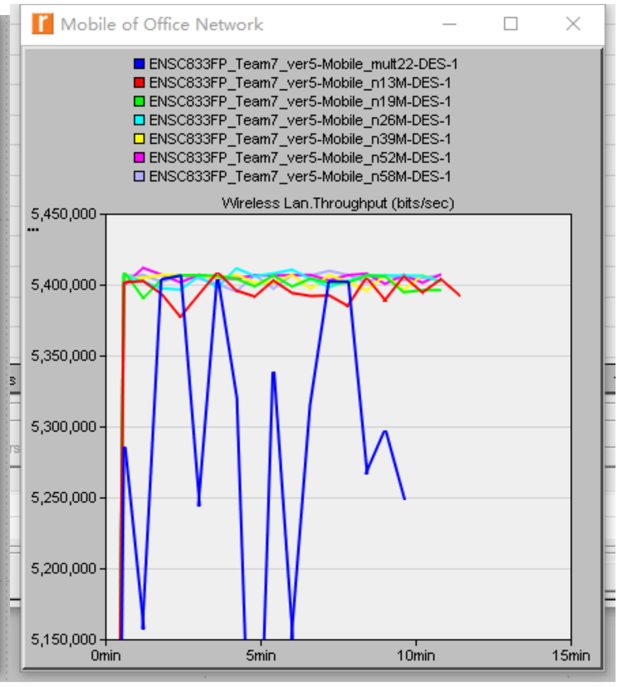


Figure 3-24 Detailed throughput of 802.11n

Here are the results of delay and retransmission attempts for each protocol:

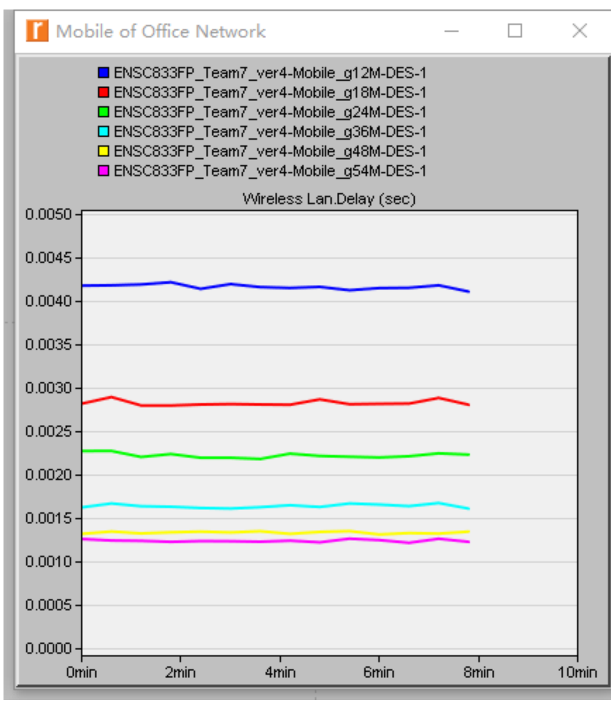


Figure 3-25 Delay of 802.11g

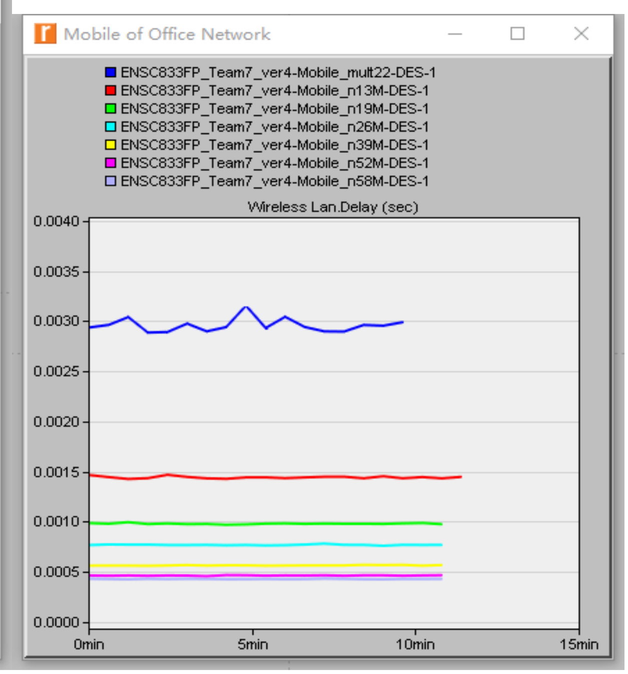


Figure3-26 Delay of 802.11n

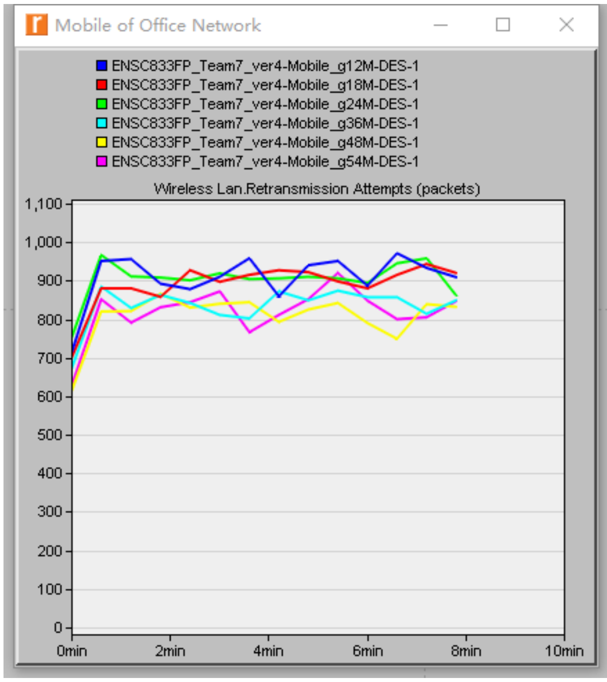


Figure 3-27 Retransmission of 802.11g

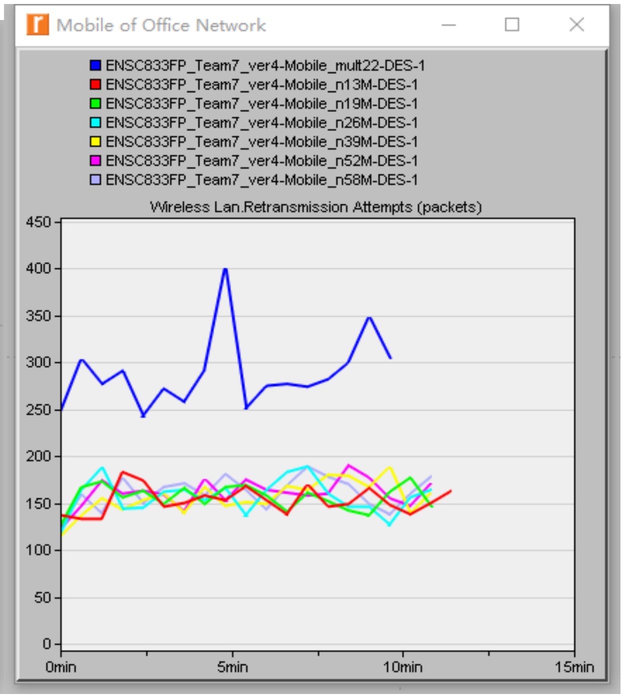


Figure 3-28 Retransmission of 802.11n

According to the figures 3-25,3-26,3-27, and 3-28, it is generally observed that throughput increase with data rates, while delays and retransmission attempts decrease as data rates increases. Here is a more detailed discussion:

scenario 3.2 Retransmission & delay part

In terms of retransmission, as the figures 3-27, and 3-28 are shown above, 802.11n generally has fewer retransmission attempts than 802.11g. However, the attempts vary from 800 to 1000 packets in 802.11g, while only 100 to 400 packets in 802.11n. According to the documentation of "frame aggregation" [20], one possible reason is that 802.11n can put a few frames into one big frame to enhance the performance of retransmission attempts. Another possible reason is that higher throughput in 802.11n reduces the retransmission attempts.

According to the figure (3-27, and 3-28) for 802.11g retransmission, the general trend is "the higher data rate, the fewer retransmission attempts." Due to the higher data rate can reach higher throughput (process more packets), which reduces the retransmission attempts from clients. 36Mbps, 48Mbps 54Mbps have fewer retransmission attempts than 12Mbps, 18Mbps 24Mbps

However, in 802.11n, there is no significant difference between the data rates other than 6.5Mbps. Therefore, all of the retransmission attempts are between 100 to 200. Compared with the baseline we discussed in Scenario1 (5.4Mbps, 100-160 retransmission attempts), we thought that since they have reached the maximum throughput (5.4Mbps, we discussed above), the 100-200 retransmission attempts are also the close to the minimum retransmission attempts they can reach. Another reason is that the retransmission attempts have already small when the data rate is 13Mbps, and then the effect of increasing to the higher data rate (19.5,26,39,52,58.5Mbps) is not apparent.

The reason is that they have met the requirement to transmit at the maximum rate, so there is no significant transmission blockage. Also, due to the inherently low retransmission count of 802.11n, a more extensive data rate does not have a noticeable effect on reducing the retransmission number.

In terms of delay, as the figures 3-25, and 3-26 are shown above, the general trends of both wireless LAN protocols are "the higher data rate, the lower delay." The higher data rate makes clients receive the packets faster, which reduces the delay. Based on the technique advantage of 802.11n [9][19][20], 802.11n has a faster delay than 802.11g generally. Besides, as we discussed in scenario 2, the fewer retransmission attempts of 802.11n may influence their delay and make it faster.

In this scenario, 802.11n performs much better than 802.11g on mobile devices.
In the same wireless LAN protocol, the overall performance of mobile becomes better as the increase of data rate.
So far, all the simulations have been done on the premise that the mobile is not moving. So next, we try to join the trajectory and place the mobile far away from the router.

Scenario 4: Effect of distance and movement on data rate and Wi-Fi protocol

The sub-scenarios of this scenario are modified by their corresponding sub-scenarios in Scenario3 (same date rate and protocol). All mobile nodes in Scenario3 will be added a trajectory and re-placed.
This scenario investigates three aspects of a mobile node watching a video while moving: the impact of distance between the user and the router; the range of different protocols and data rate; the action of different protocols and data rate when the user re-enters the range of the router.
Due to the completed results being more complex than in previous scenarios, we selected four separate data groups and discussed their performance first. Then, we discuss the overall performance of mobile with different data rates (802.11g and 802.11n) like what we discussed in scenario 3.
We select 802.11g/12Mbps, 802.11g/24Mbps, 802.11n/13Mbps, and 802.11n/39Mbps.
Here are their plots of “delay,” “delay vs. throughput,” and “delay vs. retransmission”:

802.11g/12Mbps

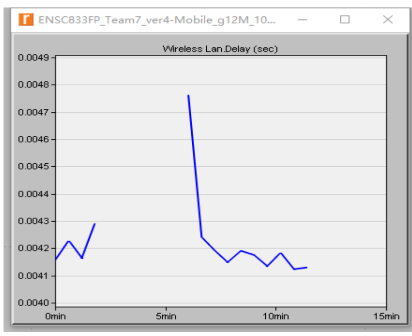


Figure 3-29 Delay

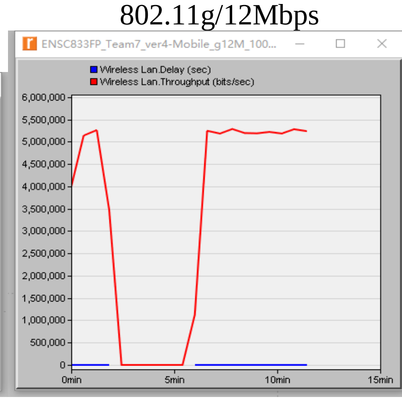


Figure 3-30 Delay vs Throughput

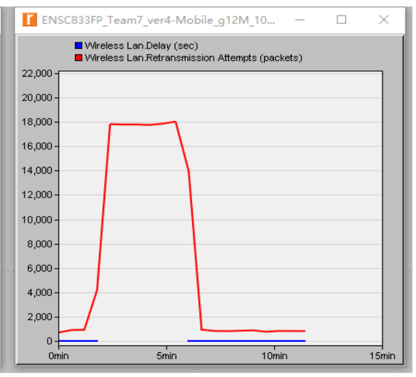


Figure 3-31 Delay vs Retransmission

802.11g/24Mbps

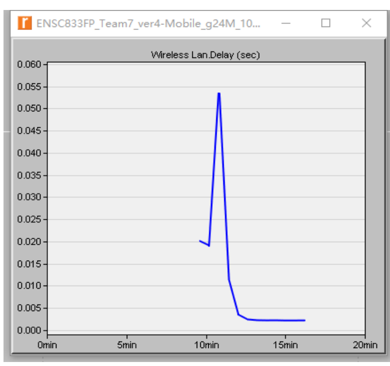


Figure 3-32 Delay

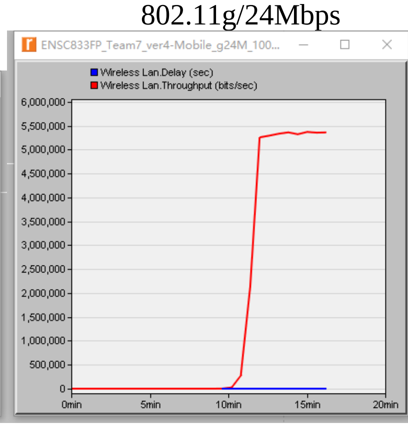


Figure 3-33 Delay vs Throughput

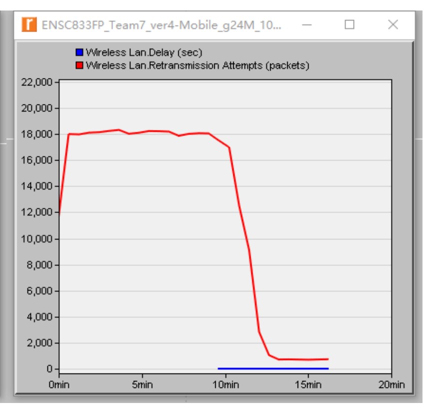


Figure 3-34 Delay&Retransmission

802.11n/13Mbps

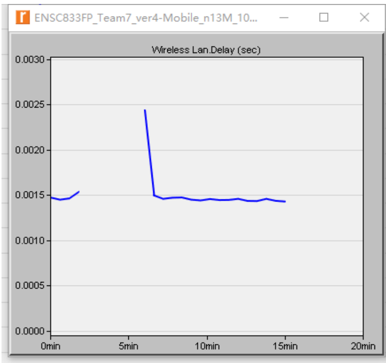


Figure 3-35 Delay

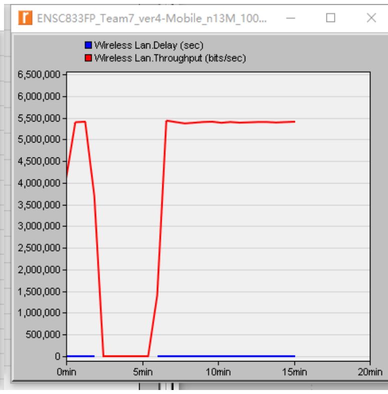


Figure 3-36 Delay vs Throughput

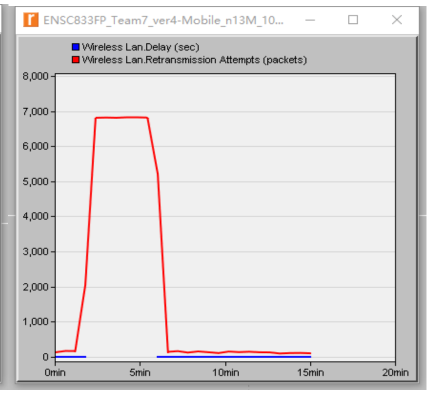


Figure 3-37 Delay&Retransmission

802.11n/39Mbps

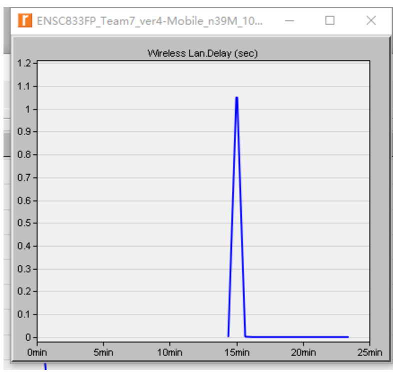


Figure 3-38 Delay

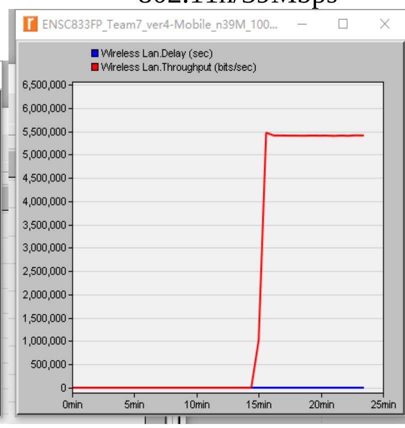


Figure 3-39 Delay vs Throughput

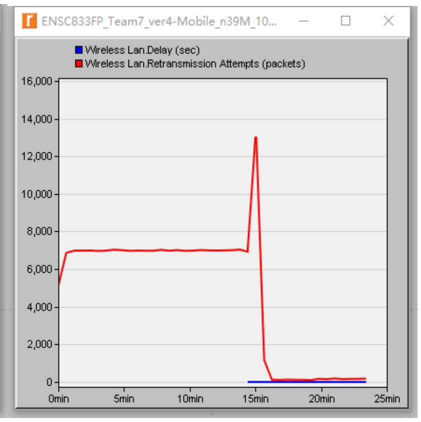


Figure 3-40 Delay&Retransmission

According to the four groups of figures (Figures from 3-29 to 3-40) shown above:
As figures 3-31, 3-34, 3-37, and 3-40 shown above, when the mobile is out of the router's effective range, the mobile will keep sending retransmission attempts, zero throughputs, and no delay. During this period, the maximum retransmission attempts (per unit time) in 802.11g are around 18000 packets, and 802.11n's are around 7000 packets. In this case, 18000 may be the upper limit of retransmission attempts that mobile can reach with the given page properties under 802.11g. Similarly, in this scenario, 7000 may be the upper limit of retransmission attempts in 802.11n.
Then, when the mobile goes back to the effective range of the router, the throughput increases to 5.4Mbps in a short time, and the delay and retransmission seriously increase and then decrease to the typical level in short. This phenomenon is serious in 802.11n.
There is a paper "Bufferbloat: Dark Buffers on the Internet" that describes a problem called "bufferbloat": This problem occurs when too many packets/requests are in a huge buffer, the packets become much harder to arrive in time and have to stay in the buffer [25]. Then the hosts will keep sending requests that may make the situation worse. Furthermore, the final result is a huge delay and a huge buffer. The papers "Resolving Bufferbloat in TCP Communication over IEEE 802.11n WLAN by Reducing MAC Retransmission Limit at Low Data Rate" [23], "An empirical evaluation of bufferbloat in IEEE 802.11n wireless networks" [24] mentioned the bufferbloat problem in 802.11n, and it may lead serious problem in real-time transfer such as VoIP.
Furthermore, during a short time, the mobile sends massive retransmission attempts. As a result, its corresponding delay also becomes very high, which matches the case of congestion we discussed in scenario2: “the higher the sending rate, the higher the delay.” [28]

In this case, the phenomenon of 802.11n we described above may mean the bufferbloat(congestion) occurs when the phone returns to the router's effective range. The huge cumulative requests of mobile need to be processed, which may lead to high retransmission and high delay in a short. Furthermore, when cumulative requests are completed, the retransmission and delay go back to the typical level like the plots in scenario 3. In 802.11g, their retransmission attempts keep high (18000). However, as the figures of delay of 802.11g (figures 3-29, 3-32, 3-35, and 3-38) shown above, the delay is still increased temporarily, representing that the bufferbloat(congestion) may occur, but not severe.
Hence, the mobile may occur bufferbloat(congestion) when it arrives in the router's range due to the cumulative retransmission attempts and resulting in a high delay in a short time. Furthermore, 802.11n may be affected more seriously than 802.11g from bufferbloat(congestion).

Here are the total complete figures of 802.11g and 802.11n with different data rates:

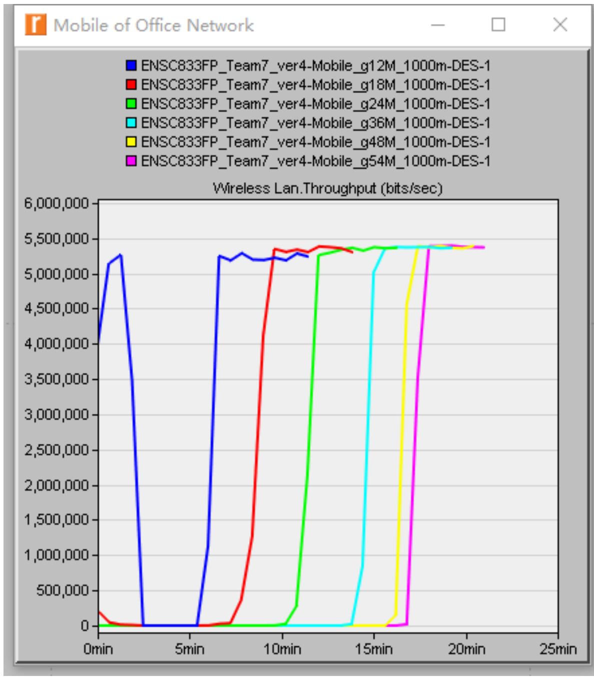


Figure3-41 Throughput of 802.11g (scenario 4)

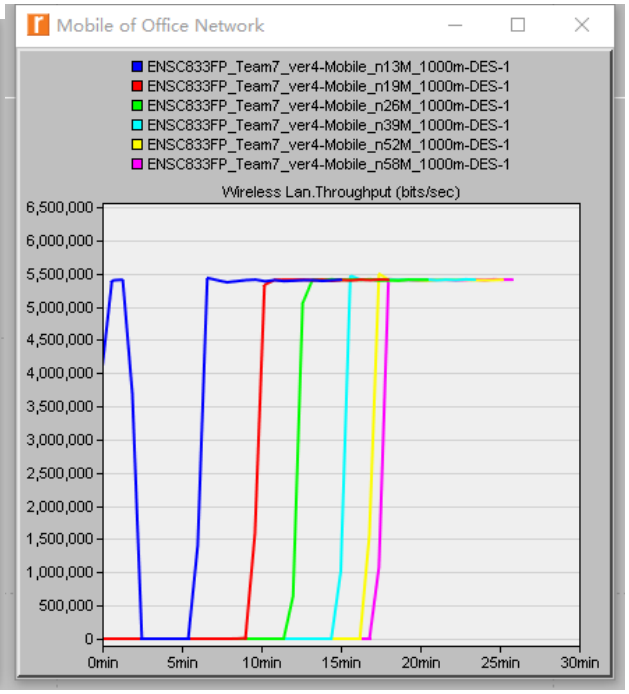


Figure3-42 Throughput of 802.11n (scenario 4)

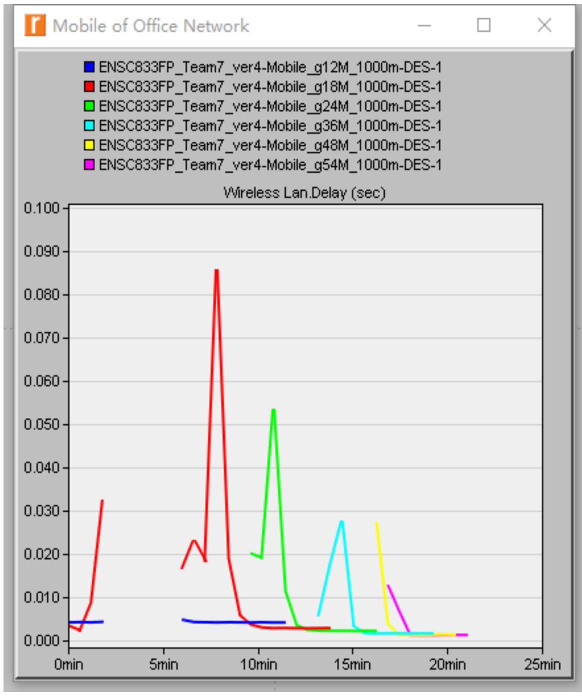


Figure3-43 Delay of 802.11g (scenario 4)

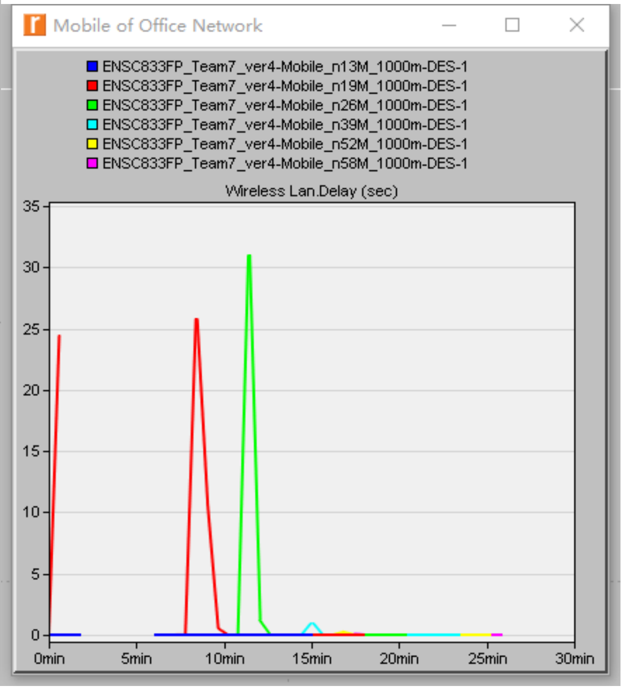


Figure3-44 Delay of 802.11n (scenario 4)

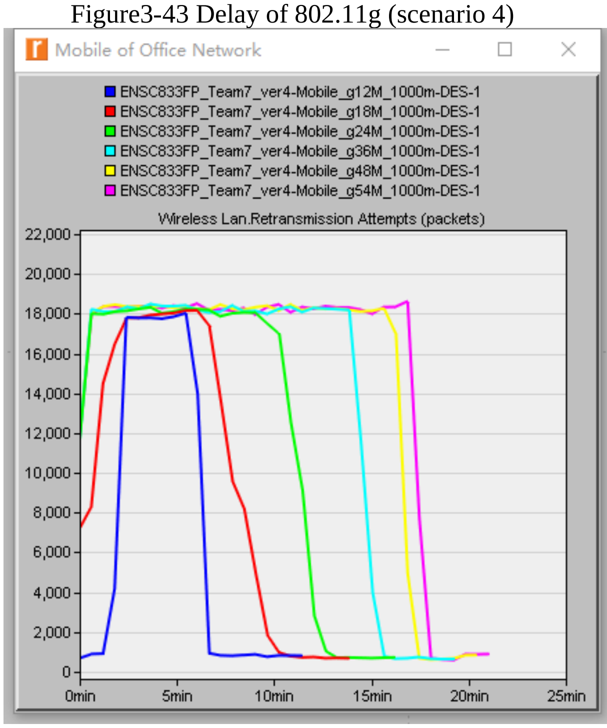


Figure3-45 Retransmission of 802.11g (scenario 4)

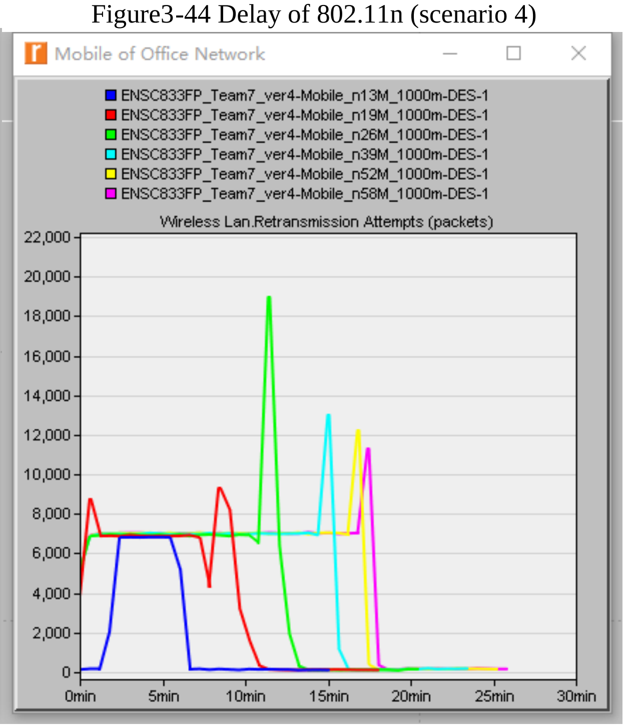


Figure3-46 Retransmission of 802.11n (scenario 4)

Based on any six figures shown above, the higher the data rate in the same wireless LAN protocol, the smaller the effective range. According to Figures 3-41 and 3-42, the later it returns to 5.4Mbp, the closer it is to the router, and the smaller the effective range is. Besides, the 802.11g/12Mbps and 802.11n/13Mbps's effective ranges are over 1000 meters because they have throughput at the beginning.

Notes: When the mobile is out of the router's range, there is no event for the mobile so the simulation time can be over 8 or 12 minutes.

When the mobile is back to the router's effective range in each wireless LAN protocol, the retransmission attempts, and delay will temporarily increase and then decrease to the typical level. As we described before, this phenomenon may mean that the bufferbloat(congestion) occurs.
As the figures 3-41 and 3-42 are shown above, when the mobiles with different data rates keep moving in the router's effective range, the throughput performance is almost the same as their performance in scenario 3.
Here are figures about detailed performance, delay and retransmission attempts when mobile is in the effective range:

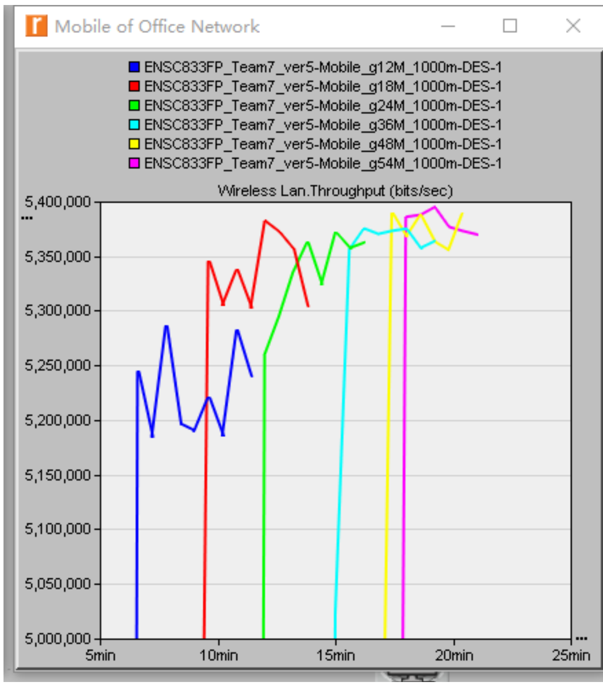


Figure3-47 Detailed throughput of 802.11g (scenario 4)

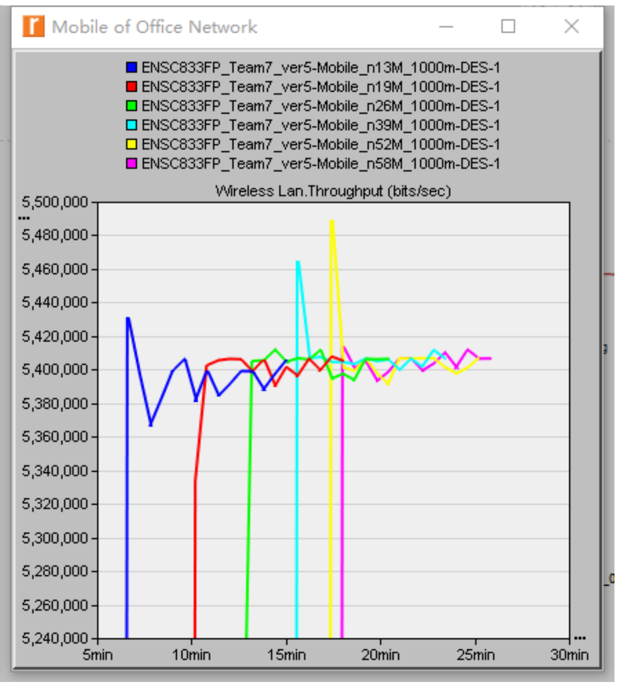
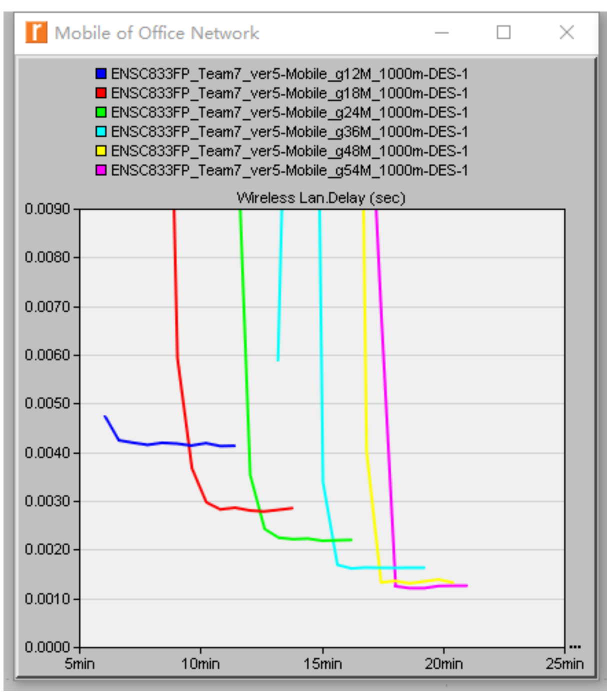


Figure3-48 Detailed throughput of 802.11n (scenario 4)



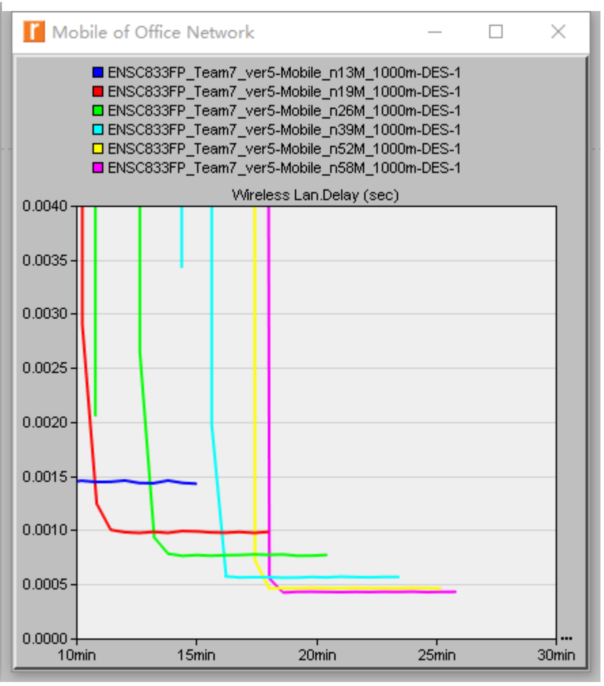
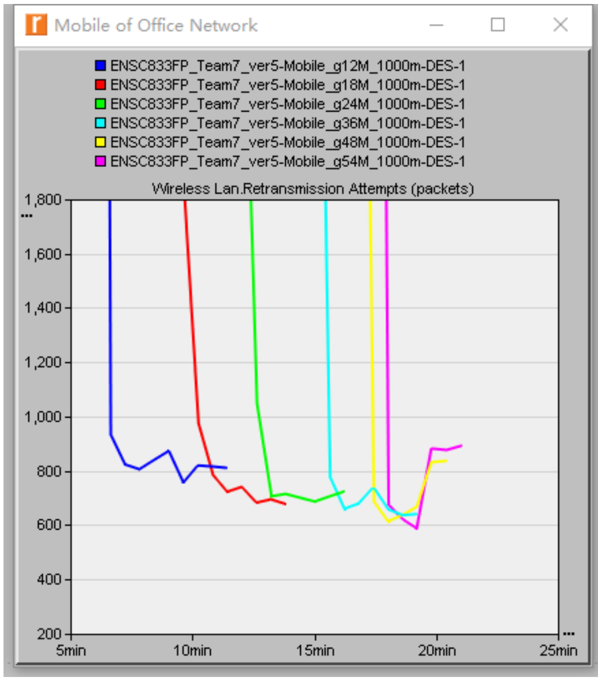


Figure3-49 Detailed delay of 802.11g (scenario 4) Figure3-50 Detailed delay of 802.11n (scenario 4)



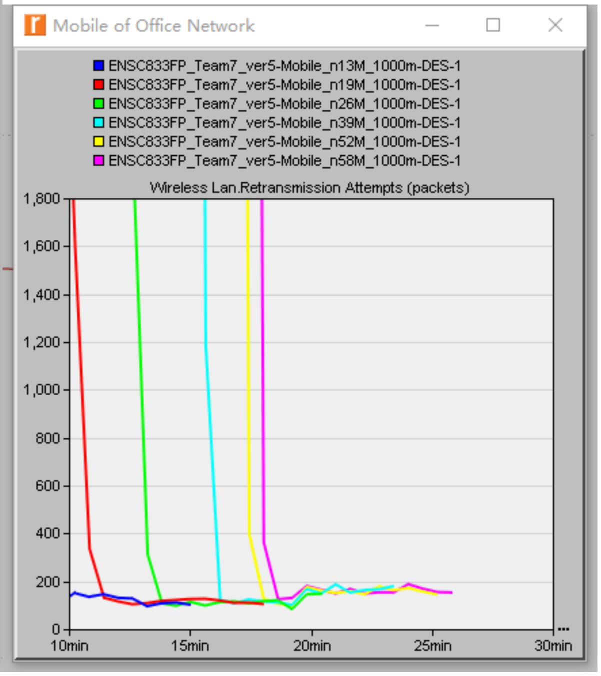


Figure3-51 Detailed retransmission of 802.11g (scenario 4) Figure3-52 Detailed retransmission of 802.11n (scenario 4)

As figures from 3-47 to 3-52, mobile's overall performance (throughput, delay, and retransmission attempts) in the range matches their corresponding overall performance in scenario3.
Since "the mobile keeps moving" is equivalent to "the mobile is approaching the router," the distance decreases. Hence, we can see that the change in distance does not affect throughput performance, and they have the same performance as their performance in scenario 3, which is not matched what the paper [29] said.
There are two possible reasons why distance does not influence the throughput:
First, the throughput reaches the maximum once the mobile goes back to the range.
Second, the distance will not influence the throughput in these Riverbed configurations.

In terms of retransmission attempts and delay, figures 3-43, 3-44, 3-45, and 3-46, all mobiles with 802.11g keep around 18000 retransmission attempts out of the range, and mobiles with 802.11n keep around 7000 attempts. According to the figures 3-43, 3-44, 3-45, and 3-46, when they go back to the range, the bufferbloat(congestion) may occur. In 802.11g, the bufferbloat(congestion) problem only results in a temporarily high delay.
Besides, the general trend is that the higher the data rate, the lower the maximum value of "high delay," which may be due to the higher data rate can process the requests more efficiently.
In 802.11n, the bufferbloat(congestion) problem represents the temporarily serious retransmission attempts and high delay, and their performance is worse than 802.11g when bufferbloat(congestion) occurs. Besides, the general trend is similar to the 802.11g except for 19Mbps and 39Mbps. Therefore, 802.11n/Mbps has the worst performance during this short time.
Hence, in this scenario, when the mobile is out of the range, they do not have throughput and delay and keep sending retransmission attempts. When the mobile arrives at the range, the bufferbloat(congestion) may occur. During the short time, the throughput increases to 5.4Mbps, and the retransmission and the delay increase seriously. After that, the throughput kept around 5.4Mbps, and the retransmission and the delay decreased. When the phone is near the router, the mobiles with different data rates have the same performance as in Scenario3, and the distance does not affect the performance.

4.Conclusion & Improvement

This project simulates a mobile watching a 1080P/30FPS YouTube video in different scenarios through Riverbed Modeler.
The first scenario shows the overall performance of mobile: about 5.4Mbps, around 100-160 retransmission attempts, and about 0.0028-second delay. On the other hand, the delay performance of fixed nodes (Computer) is better than mobile, and the rest performance is the same as mobile. Therefore, both mobile delay and fixed nodes' delay will not affect the QoS in video streaming.

In the second scenario, the mobile performance will be influenced due to the other users' requests that need to be handled. As a result, the mobile will send more retransmission, leading to congestion, delay, and lower throughput. Then the mobile may be extra influenced due to the sent retransmission by itself. The more the other nodes, the worse the overall mobile performance

In the third scenario, the mobile has a better overall performance with a higher data rate. Furthermore, mobile with 802.11n has better performance than mobile with802.11g. Besides, when a node's profile archives 100% efficiency, the higher data rate will keep throughput and improve delay (obvious) and retransmission attempts (slight).

In the fourth scenario, the router's effective range becomes smaller as its data rate becomes higher. When the mobile is out of the router's effective range, it will keep sending the retransmission attempts and has no throughput or delay. On the other hand, when the mobile arrives at the effective range, the massive retransmission may lead to bufferbloat(congestion) phenomenon and will be done quickly. Besides, the distance and movement do not affect the mobile performance in this scenario due to the configuration.

Here are possible suggestions for improving this project and future work:
Due to the academic version of the Riverbed Modeler, this project suffers from two significant limitations: First, the academic riverbed modeler has a cap of 50000000 events. Second, it cannot customize the router's valid range. If the project can be on the full version of the Riverbed Modeler, the project should be better.

This project only selects 1080p 30fps for simulation in terms of YouTube, but there is more resolution [720p 2k, 4k] and frame options (24fps, 60fps). Furthermore, the comparison between different resolutions and frame number combinations is one for future work.
Besides, the project set the attributes "uniform(0.0333,0.0345)" page interarrival time and "constant(1230) with constant(17) number" objects based on the data from YouTube Help[17].
Chapter 2 lecture describes how streaming stored video works [27]. In the future, the project can define other kinds of video browsing, such as "step-by-step data transfer" (Right now, it defines "continuous speed" video browsing). YouTube also uses DASH technology to do video streaming[7]. “A Framework for Generating HTTP Adaptive Streaming Traffic in ns-3” also mentioned that YouTube and Netflix applied HAS technology to transfer chunks.

In terms of scenarios, there is a large potential to build more complex and detailed topologies for new scenarios. For example, scenario4 found that distance and movement do not affect mobile performance. Therefore, the project can focus on scenario4 and build new topologies for improving it in the future. In other words, it is possible to apply the "cache" to the scenario as a new parameter.

Generally speaking, this project selects a group of YouTube parameters (1080P-30FPS) and builds a series of scenarios to test and analyze its performance.

Contribution

	Licheng	Linqi	Yiran
Reference and literature review	30	35	35
Project website	80	10	10
Simulation scenarios	33.3	33.3	33.3
implementation	40	30	30
analysis, and discussion of simulation results	40	30	30
Project presentation	30	30	40
Written final report	33.3	33.3	33.3

Note: the project website is very easy to build and maintain through WiX, so only one person is good enough to deal with it.

Reference

[1] J.-H. Huh, “Reliable User Datagram Protocol as a Solution to Latencies in Network Games,” Electronics, vol. 7, no. 11, p. 295, Nov. 2018, doi: 10.3390/electronics7110295.
[2] N. Zhukova, “Top 100: The Most Visited Websites in the US [2021 Top Websites Edition],” Semrush, Oct. 28, 2020. https://www.semrush.com/blog/most-visited-websites/.(accessed Apr. 19, 2022)
[3] C. Goodrow, “You know what’s cool? A billion hours,” YouTube Official Blog, Feb. 27, 2017. https://blog.youtube/news-and-events/you-know-whats-cool-billion-hours/. (accessed Apr. 19, 2022).
[4] Kim, & Park, S. (2017). Delay model of multicast protocols co-existing with legacy unicast in IEEE 802.11 wireless local area network. IET Communications, 11(9), 1438–1445. https://doi.org/10.1049/iet-com.2016.0660
[5] A. S. Gillis, “video streaming,” *TechTarget*, Nov. 12, 2021. Available: https://www.techtarget.com/searchunifiedcommunications/definition/streaming-video. (Accessed: Apr. 19, 2022.).
[6] A. Rao, A. Legout, Y. Lim, D. Towsley, C. Barakat, and W. Dabbous, “Network characteristics of video streaming traffic,” 2011. http://dx.doi.org/10.1145/2079296.2079321.
[7] “YouTube,” *Google Developers*. https://developers.google.com/youtube/v3/live/guides/encoding-with-dash (accessed Apr. 19, 2022).
[8] W. D. D. Maza, “A Framework for Generating HTTP Adaptive Streaming Traffic in ns-3,” *Archive ouverte HAL*, Aug. 22, 2016. https://hal.archives-ouvertes.fr/hal-01362445/ (accessed Apr. 19, 2022).
[9] S. Banerji and R. S. Chowdhury, “On IEEE 802.11: Wireless LAN Technology,” *arXiv.org*, Jul. 10, 2013. https://arxiv.org/abs/1307.2661.
[10] “Riverbed Modeler Academic Edition,” *Riverbed.com*. https://cms-api.riverbed.com/portal/community_home. (accessed Apr. 19, 2022).
[11] P. Krzyzanowski, “Quality of Service,” *Pk.org*, Sep. 29, 2012. https://people.cs.rutgers.edu/~pxk/417/notes/03-qos.html (accessed Apr. 19, 2022).
[12] Amandeep Kaur, Haotian Ye,Ashiv Rao Dhondea, “Performance analysis of video streaming with WiFi”, http://www.sfu.ca/~hdhondea/projectfiles/ENSC894_Spring2020_report_team2.pdf.
[13] P. Casas, P. Fiadino, A. Sackl and A. D'Alconzo, "YouTube in the move: Understanding the performance of YouTube in cellular networks," 2014 IFIP Wireless Days (WD), 2014, pp. 1-6, doi: 10.1109/WD.2014.7020798.
[14] S. Abdallah-Saleh, Q. Wang, C. Grecos and D. Thomson, "Handover evaluation for mobile video streaming in heterogeneous wireless networks," 2012 16th IEEE Mediterranean Electrotechnical Conference, 2012, pp. 23-26, doi: 10.1109/MELCON.2012.6196371.
[15] D. Jain, S. Agrawal, S. Sengupta, P. De, B. Mitra and S. Chakraborty, "Prediction of quality degradation for mobile video streaming apps: A case study using YouTube," 2016 8th International Conference on Communication Systems and Networks (COMSNETS), 2016, pp. 1-2, doi: 10.1109/COMSNETS.2016.7440005.
[16] Michael Ng, Ching Ho Weng, “Video Streaming over WiFi using Riverbed Modeler”, http://www.sfu.ca/~ljilja/ENSC427/Spring16/Projects/team1/ENSC427_team1_report.pdf
[17] “System requirements,” YouTube Help. https://support.google.com/youtube/answer/78358?hl=en (accessed Apr. 27, 2022).
[18] “How far will your Wifi signal reach?”, OpenWeb.co.za, Oct. 07, 2018. https://openweb.co.za/how-far-will-your-wifi-signal-reach/ (accessed Apr. 26, 2022).
[19] D. Gong, Y. Yang and H. Li, "An Efficient Cooperative Retransmission MAC Protocol for IEEE 802.11n Wireless LANs," 2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems, 2013, pp. 191-199, doi: 10.1109/MASS.2013.27.
[20] “IEEE 802.11 Frame Aggregation — INET 4.3.0 documentation,” inet.omnetpp.org. https://inet.omnetpp.org/docs/showcases/wireless/aggregation/doc/index.html (accessed Apr. 28, 2022).

[21] J. Kim, J. Zheng, and P. Bertsch, “Video Streaming over Wi-Fi,”
http://www.sfu.ca/~ljilja/ENSC427/Spring15/Projects/team2/ENSC427_team2_report.pdf
[22] S. Calzada, C. Rietchel, and T. Szajner, “Performance Analysis of a Wireless Home Network,"
https://www.sfu.ca/~ljilja/ENSC427/Spring14/Projects/team4/ENSC427_team4_report.pdf
[23] M. Nomoto, C. Wu, S. Ohzahata, and T. Kato, “Resolving bufferbloat in TCP communication over IEEE 802.11n Wlan by reducing Mac retransmission limit at low data rate,” ThinkMind(TM) Digital Library, 23-Apr-2017. [Online]. Available: https://www.thinkmind.org/index.php?view=article&articleid=icn_2017_4_30_30016. [Accessed: 30-Apr-2022].
[24] A. Showail, K. Jamshaid and B. Shihada, "An empirical evaluation of bufferbloat in IEEE 802.11n wireless networks," 2014 IEEE Wireless Communications and Networking Conference (WCNC), 2014, pp. 3088-3093, doi: 10.1109/WCNC.2014.6953002.
[25] J. Gettys, "Bufferbloat: Dark Buffers in the Internet," in IEEE Internet Computing, vol. 15, no. 3, pp. 96-96, May-June 2011, doi: 10.1109/MIC.2011.56.
[26] L.Trajkovic, (2022)“Chapter 1: Introduction” [PDF]
[27] L.Trajkovic, (2022)“Chapter 2: Application Layer” [PDF]
[28] L.Trajkovic, (2022)“Chapter 3: Transport Layer” [PDF]
[29] Krishan, & Laxmi, V. (2015). IEEE 802.11 WLAN Load Balancing for Network Performance Enhancement. Procedia Computer Science, 57, 493–499. https://doi.org/10.1016/j.procs.2015.07.371
[30] L.Trajkovic, “Wireless LAN Tutorial”, ENSC833, 2022 spring