CHAPTER 5

PETRA CONGESTION CONTROL OVER LONG-TERM EVOLUTION (LTE) NETWORKS

Wireless technologies have experienced expansive improvements in the past decade, and new technologies have been rapidly introduced. Nowadays, wireless technologies are literally everywhere, with Wi-Fi signals, fast 4G LTE access points, and Bluetooth running everything from printers to security terminals at your house front gate. However, some new technologies have not yet shown their expected potential. One reason for this is that traditional techniques fail to support new requirements. For example, there is the poor performance of TCP's congestion control mechanisms over wireless links. Therefore, in this study we introduced a new TCP congestion control mechanism based on the modifications presented in chapter four.

This chapter will discuss implementing and evaluating the proposed modifications given in chapter four over LTE networks. We have presented three modifications to the TCPW, including enhancement of the slow start phase, enhanced fast retransmission and a faster recovery algorithm, and an enhanced timeout retransmission procedure. We have discussed the implementation and the evaluation for each modification using a modified TCPW implementation included in ns-3. As a result, the new modification showed better performance compared to TCPW and NewReno in terms of the total throughput and congestion window size.

The following section will introduce the LTE technology, implementing LTE in ns-3, and TCP performance over LTE. Then we will discuss the implementation of TCP PETRA in ns-3. The evaluation process will include more network performance metrics: throughput, congestion window size, average delay, and jitters. Also, we extend our study to test the fairness of the proposed implementation.

5.1 Long Term Evolution Networks

Cellular technologies are designed to provide communications between two mobile units called mobile stations, or between a mobile and one stationary unit called a land unit (Forouzan, 2007; Sesia & Toufik, 2011). Thus, the service provider has to locate, keep track, assign channels, and smoothly handover the mobile units, as they move all the time from one station to another. To locate a mobile device, each service area is divided into regions called "cells" which are then controlled by a base station. The base station is controlled by a switching centre, called a Mobile Switching Centre (MSC), which coordinates the communications between different base stations. Figure 5.1 shows a simple cellular system.

Figure 5.1: The Cellular System

The development of cellular technologies can be traced in sequential but different generations from the early 0G services like Mobile Telephone Service (MTS) and the Improved Mobile Telephone Service (IMTS), to the 1G cellular networks, to the 2G digital cellular networks, broadband data service in 3G, and the latest technology of 4G services with native-IP networks. Table 5.1 shows a brief description of these generations.

Table 5.1: Cellular Network Generations

	Commercially Used	Capacity	Evolution	Specification
1G	1980	2 - 5.6 Kbps	AMPS, NTM	Analog cellular networks for voice communication
G2	1990	10 – 14 Kbps	GSM,3GPP ,AMPS	Digital wireless networks, primarily for voice communication, but limited data transmission
2.5G	1998	50 – 144 Kbps	GPRS, HSCSD, EDGE, EGPRS	Interim step toward 3G services
3G	2000	2 – 3 Mbps	3GPP2, UMTS, WiMAX	Video, data, voice, and web suffering
4G	2010	100Mbps-1Gbps	LTE, LTE-Advance, WiMax-Advance	IP-based networks, full web browsing, video conferencing, etc.

The first version of LTE was introduced with the release of the 8.0 version of the third Partnership Project (3GPP) specification series in 2008. LTE is the leading of IP-based Orthogonal Frequency-Division Multiple Access (OFDMA) wireless mobile technology. According to 3GPP (2012), the objectives of producing LTE were:

- Reduced latency
- Higher user data rates
- Improved system capacity and coverage, and reduced overall cost for the operator
- Potential network and traffic cost reduction

- Flexible accommodation and deployment of existing and new access technologies with mobility by a common IP-based network.

LTE used radio access called Evolved UMTS Terrestrial Radio Access Network (E-UTRAN). 3GPP specifies the Evolved Packet Core (EPC) network architecture to support the E-UTRAN through a reduction in the number of network elements, simpler functionality, improved redundancy, and permission to connect and handover to another fixed line, as well as wireless access technologies which deliver seamless mobility (Ergen, 2009). Figure 5.2 shows the LTE structure.

Figure 5.2: LTE High-Level Architecture

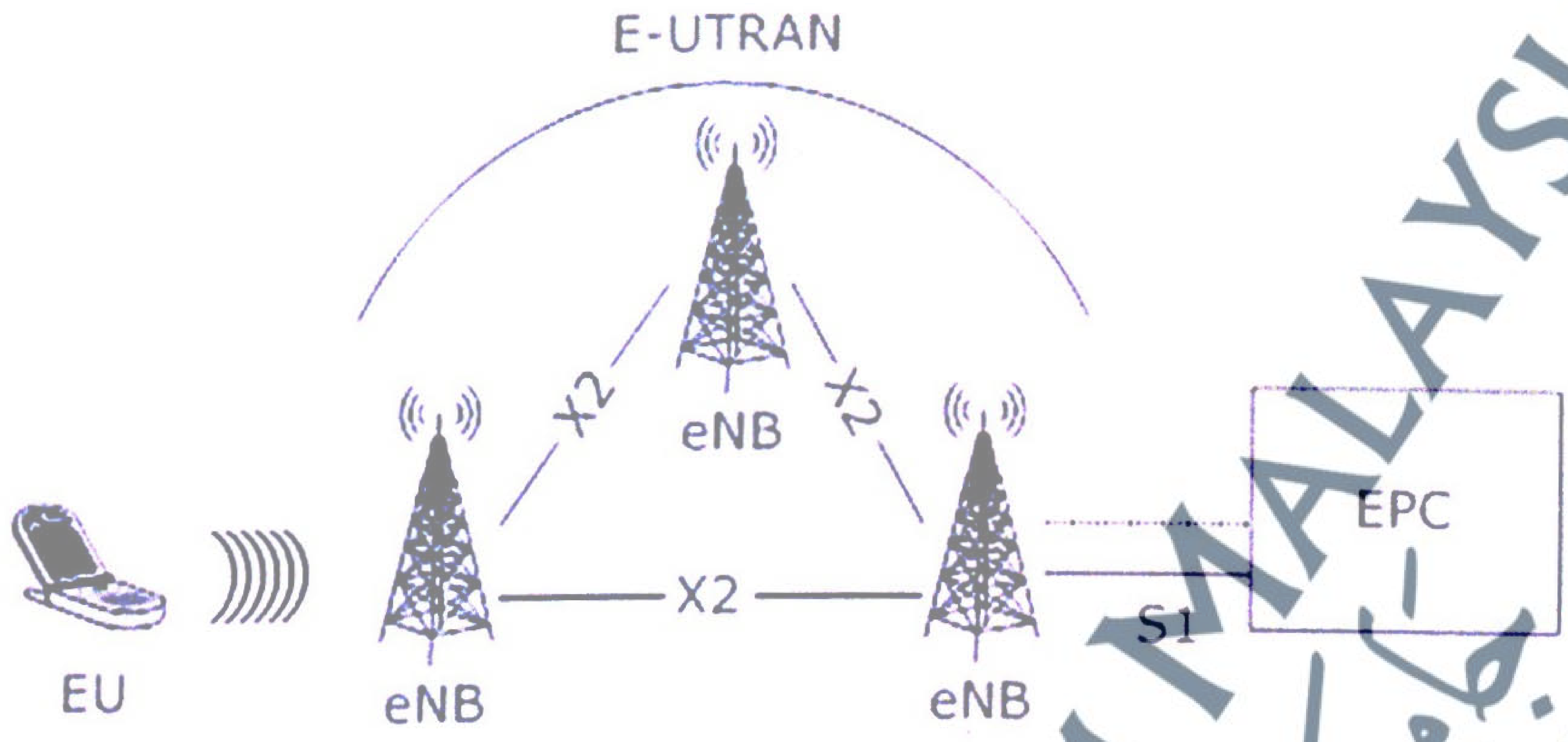
The LTE architecture is comprised of three main components:

- The user equipment (UE)
- The Evolved UMTS Terrestrial Radio Access Network (E-UTRAN)
- The Evolved Packet Core (EPC).

The user equipment in LTE has the same architecture as the one used by UMTS and the Global System for Mobile Communication (GSM), which is a Mobile Equipment (ME). ME is comprised of three modules: the Mobile Termination (MT), which handles all the communication functions; the Terminal Equipment (TE), which terminates the data streams; and the Universal Integrated Circuit Card (UICC), also known as the SIM card for LTE. UICC runs the Universal Subscriber Identity Module (USIM) where it keeps the user information as phone number, home network identity and security keys, etc.

On the other hand, the E-UTRAN (the access network) handles the radio communications between UE and EPC. Figure 5.3 shows the E-UTRAN architecture.

Figure 5.3: E-UTRAN architecture



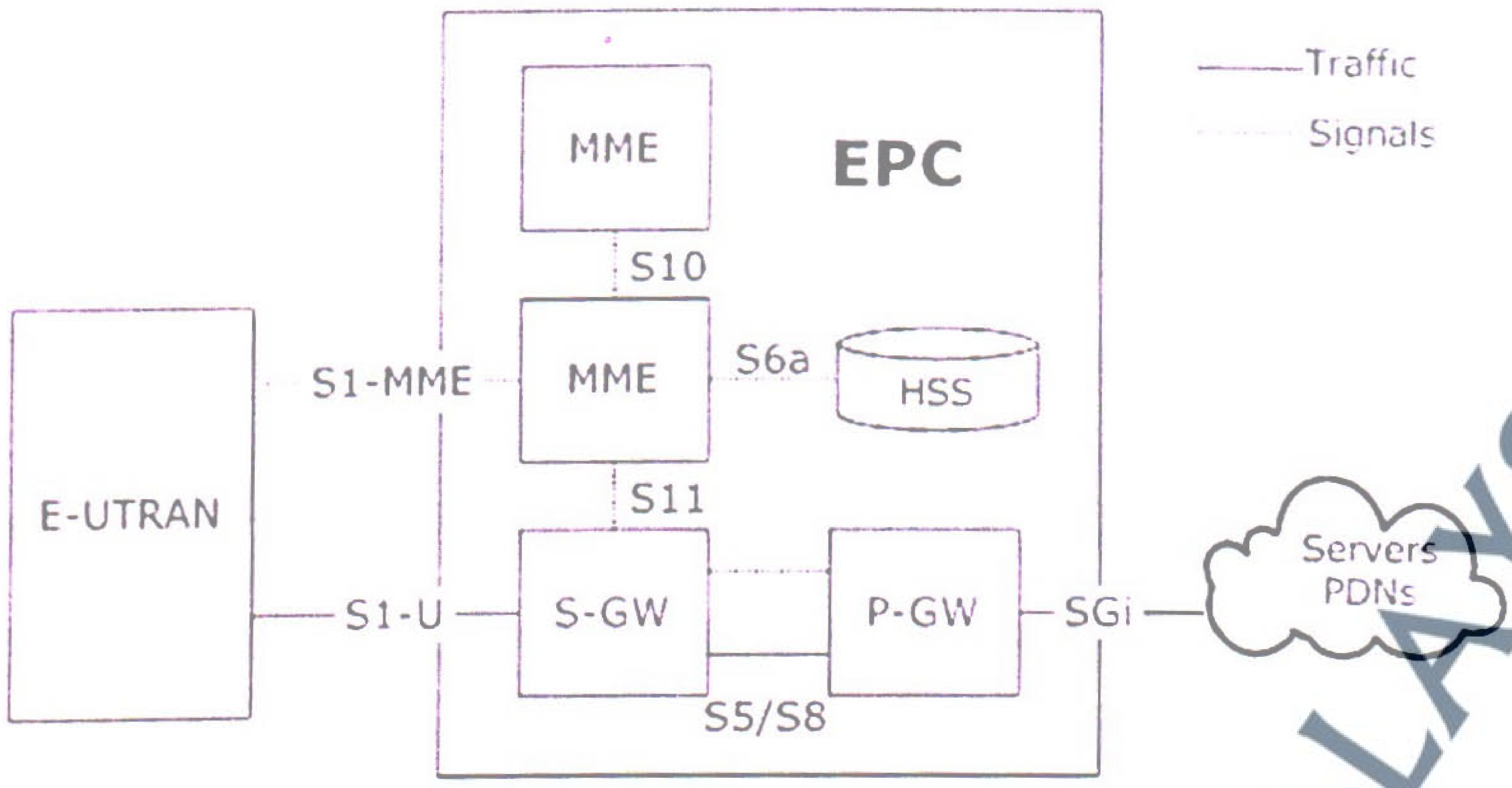
E-UTRAN has one component called the evolved Base Station (eNB). Each eNB controls the mobiles in one or more cells. However, the LTE Mobile communicates just one eNB and one cell at a time. The main functions of the eNB are to control:

- sending and receiving the LTE mobiles radio transmission
- LTE mobile low-level operations such as handover commands.

The eNBs connected to a nearby eNB uses X2 interfaces, and are connected to EPC by S1 interfaces. Another type of eNB is the Home eNB (HeNB) which is a base station that has been owned by a user to provide femtocell coverage to a Closed Subscriber Group (CSG) and can only be accessed by mobiles with a USIM belonging to the same CSG.

The last part of the LTE architecture is the Evolved Packet Core (EPC), which is the core part of the LTE network. The architecture of the EPC is shown below in Figure 5.4.

Figure 5.4: Evolved Packet Core (EPC)

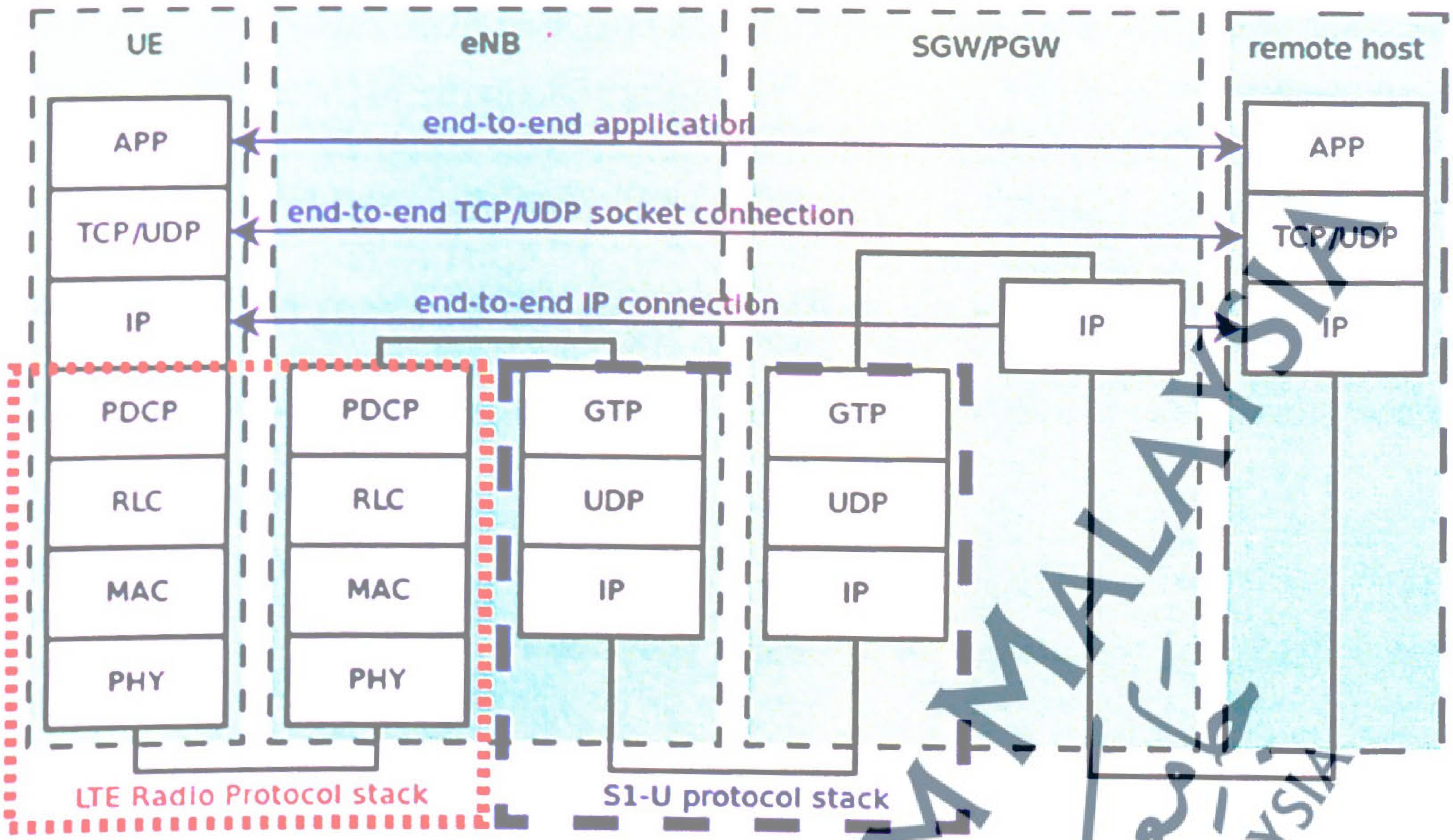


The EPC contains the Home Subscriber Server (HSS) which is a central database that contains the network operator subscriber's details. The EPC communicates with PDN via a Packet Data Network Gateway (P-GW) using SGi interfaces. Each P-GW is identified by its unique Access Point Name (APN). The data forwards from eNB to P-GW using a Server Gateway (S-GW), which acts as a router between them. The interfaces between S-GW and P-GW are either S5 if the two devices are in the same network or S8 if they are in different networks. The high-level operations of LTE mobiles are controlled and managed by the Mobility Management Entity (MME).

5.2 Implementing LTE in ns-3

In this section, we discuss implementing the LTE model in the ns-3 simulator. As presented in chapter 3, the LTE-LENA model has been adopted to implement the LTE-EPC simulation model in this study. The model includes the LTE model entities residing in UE and eNB nodes, and the core network interfaces, protocols and entities (SGW, PGW and MME nodes) in the EPC Model. Figure 5.5 shows the End-to-End LTE-EPC data plane protocol stack.

Figure 5.5: LTE-EPC data plane protocol stack (From LENA V8 Documentation, n.d)



In the above figure, it is clear that the SGW and PGW functionality is included within a single SGW/PGW node. This simplification will remove the need for the S5 or S8 interfaces; however, for other protocol stack (S1-U and LTE radio protocol stack) all the protocol layers specified by 3GPP are present (LENA v8 documentation).

The LTE-LENA model has been included in ns-3.18 version (Carneiro et al., 2009). The full details of the model can be found in (http://lena.cttc.es/manual/lte-design.html). Many Key Performance Indicators (KPIs) are available at different levels in the simulation output. This starts from the channels: PHY, MAC, Radio Link Control (RLC) protocols and Packet Data Convergence Protocol (PDCP), and finally to the IP and application status (as a FlowMonitor, PCAP traces, etc.). Details about the experimental topologies will be discussed in the coming sections of this chapter.

5.3 Implementing PETRA in ns-3

Chapter 4 introduced three main modifications to the TCPW congestion control mechanism. The first modification enhanced the slow start phase by introducing a new method to calculate the initial *ssthresh* value. Furthermore, the modification introduced a new method to accelerate the *cwnd* growth based on the connection link utilization.

The second modification enhanced the fast retransmission and fast recovery mechanism. Firstly, we introduced a new method to avoid waiting for the third duplicate acknowledgment to arrive as the condition to trigger the fast retransmission procedure. This method will prevent the fast retransmission procedure from waiting for an unachievable condition in the case of long *RTT* values. Secondly, we introduced a new method to enhance the fast recovery algorithm by using both the current bandwidth estimation and the *RTT* intervals to improve recovery of the *cwnd* size after receiving accumulative acknowledgment.

Finally, we have introduced a new modification to the TCPW's round trip timeout (*RTO*) procedure to avoid resting the *cwnd* size to its initial value every time a packet timeout expired. The modification uses the current bandwidth estimation and the last values of *RTT* to recognize the cause of a loss event (a congestion cause or a wireless packet dropped). For each modification, we have evaluated the performance of the proposed algorithm by comparing the simulation results of the new modification with TCPW and TCP NewReno implementations using ns-3 simulator. In this chapter, we evaluate the performance of the three modifications which have been implemented as a new TCP congestion control mechanism referred to as PETRA.

To implement PETRA, we used a modified version of the TCPW implementation which is included within ns-3.19 to avoid the complications of defining a new TCP for the simulator environment. The new implementation includes all the proposed modifications in chapter 4. The complete C++ source code of the proposed new implementation is appended within Appendix E. The new implementation uses the same TCPW's header file available in ("ns-allinone-3.19/ns-3.19/src/internet/model/tcp-westwood.h"). Figure 5.6 shows a transaction diagram of PETRA congestion control mechanism.

Figure 5.6: PETRA congestion control Transaction Diagram

5.4 Evaluating PETRA over LTE

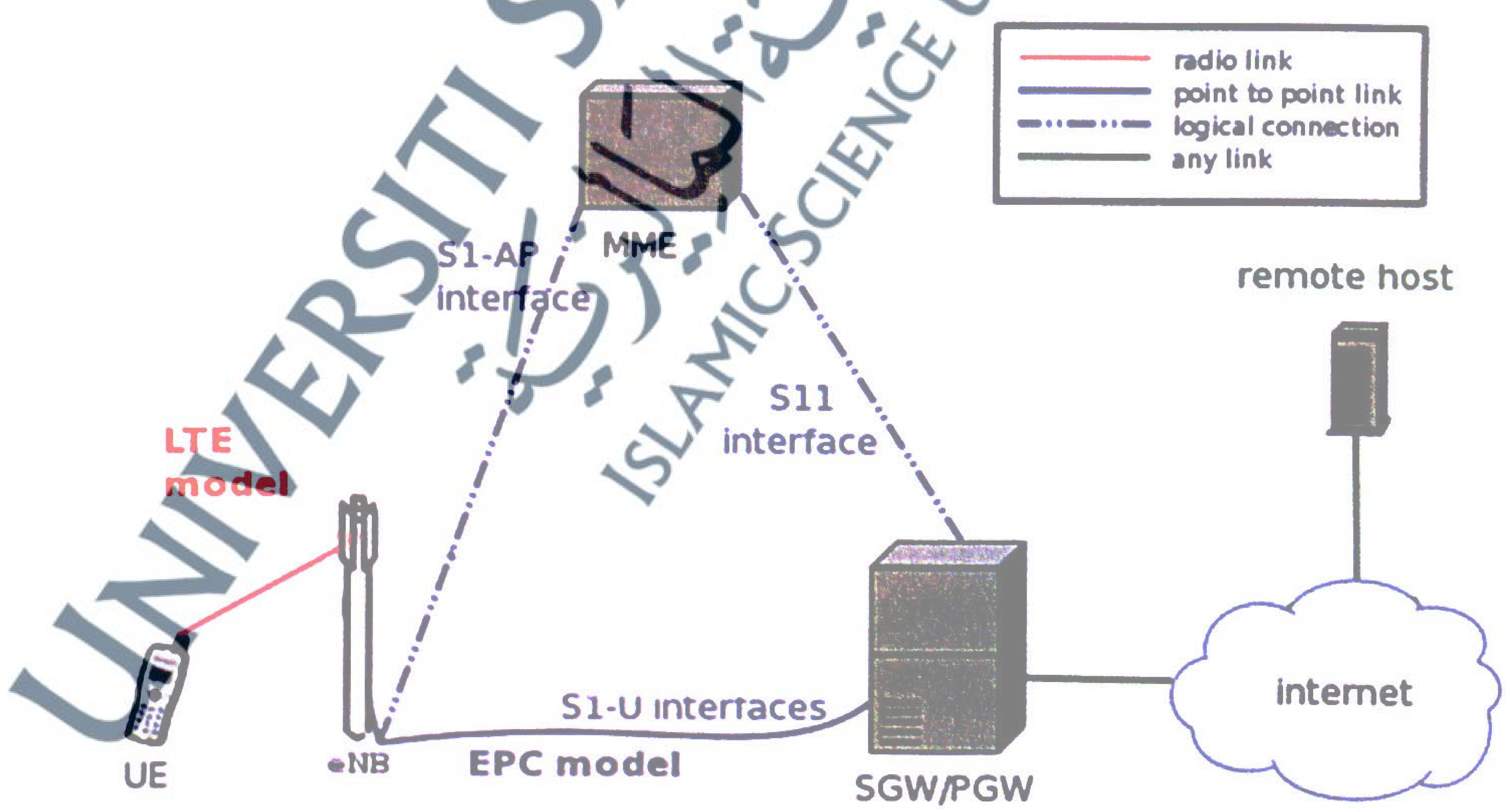
The following sections introduce the simulation results of evaluating the proposed PETRA congestion control mechanism over the LTE-LENA model using ns-3.19

simulator. We have designed three main simulation topologies to test the new modifications, as discussed in chapter 3. The first topology compares the results of congestion window size, throughput, packet loss, average delay, and the jitter sum of PETRA with TCPW and NewReno implementations. Furthermore, we test the fairness of PETRA over different flows in the second topology. In the third topology, we include the UE mobility model to the simulation topology to test the real behaviour of the new modifications over a realistic LTE environment.

5.4.1 Experiment 5-1

In this experiment, the performance of PETRA is evaluated using a simple LTE-LENA network topology as described in Figure 5.7. As discussed before, the LENA model is based on the 3GPP LTE Release 8 specifications. The topology will contain single user equipment (UE), an eNB, and a single remote host.

Figure 5.7: Experiment 5-1



The simulation scenario shown in Figure 5.7 represents a simple LENA LTE-EPC model. A single UE is connected to an eNB through the access-link using a radio link with a download transmission rate of 25 Mbps and latency of 5 ms as in (Abeta, 2010; Astely et al., 2009). The eNB is connected to the SGW/PGW and MME through the EPC S1-U, and S1-AP interfaces with a large transmission rate link of 1 Gbps and latency of 3 ms. The EPC is connected to the internet via SGW/PGW- remote host link with 100 Mbps transmission rate and 20 ms latency. The Drop-Tail queuing scheme was chosen as a queue management scheme for all the connection links. Table 5.2 summarizes the links parameters for the first experiment.

Table 5.2: Experiment 5-1 links parameters

Link	Transmission rate	Propagation Delay
Remote host-SGW/PGW	100 Mbps	20 ms
eNB-SGW/PGW & MME	1 Gbps	3 ms
MME-SGW/PGW	1 Gbps	3 ms
UE-eNB-Download Link	25 Mbps	5 ms

To simulate the wireless link over the UE-eNB link, we use the built-in ns-3's RateErrorModel class with value of 0.00005 to generate random transmission bit errors during the simulation period. A BulkSendApplication class is used to generate a single flow of traffic starting at the remote host and destined for the UE along the simulation period. Table 5.3 summarizes the simulation parameters.

Table 5.3: Experiment 5-1 simulation parameters

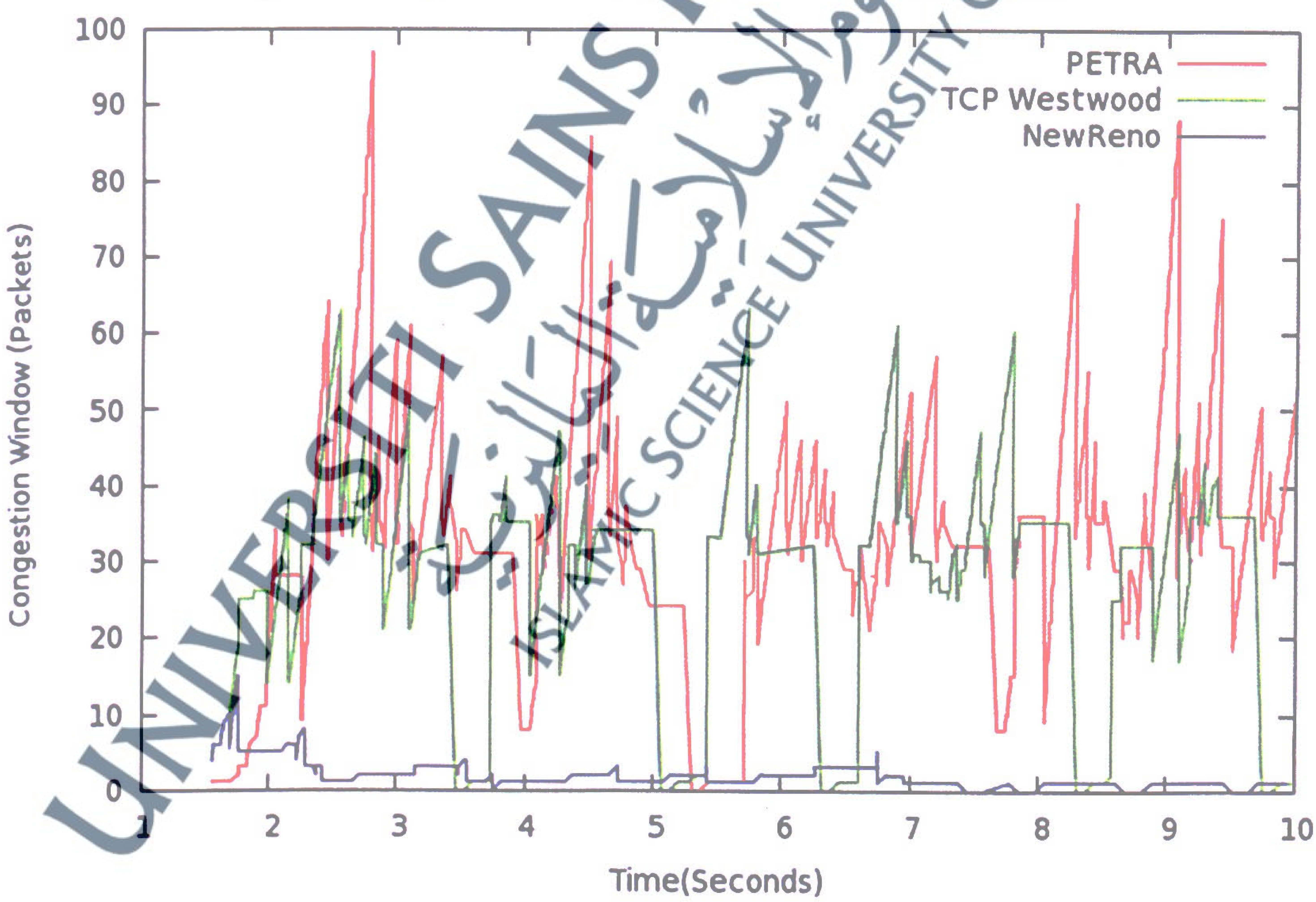
Parameter	Value
Number of UEs	1
Number of eNB	1
Packet Size	1500 bytes
Queue scheme	Drop-Tail

Simulation period	100 seconds
Error Rate	0.00005
TCP Protocols	NewReno, TCPW, PETRA

5.4.1.1 Simulation Results

We ran the first experiment using the same parameters listed in Table 5.2 and Table 5.3. The congestion window size was recorded for each transport protocol, NewReno, TCPW, and PETRA. Figure 5.8 shows the congestion window size for the three implementations for the first 10 seconds of the simulation period.

Figure 5.8: Experiement 5-1, Congestion window comparison



In the figure above, PETRA recorded higher *cwnd* values than other implementations, whereas, NewReno recorded a very small *cwnd* size. The behaviour of PETRA *cwnd* reflects the new enhancement mechanisms. One clear observation is that TCPW rested its *cwnd* size to the initial congestion window five times (at 3.5, 5.1, 6.4, 8.3, and 9.7 of the simulation period); however, PETRA rested its *cwnd* once during the first 10 seconds of the simulation period (almost at 5.3 second). This behaviour reflects the two modifications of the faster retransmission and fast recovery, and the modification of the time out retransmission procedure. A summary of the experimental results is listed in Table 5.4. The simulation results show better throughput improvements using PETRA compared to NewReno and TCPW. However, NewReno shows better performance in terms of the average delay because of the low bandwidth utilization over the links (i.e. reduction in the queuing time).

Table 5.4: Experiment 5-1 simulation results

	NewReno	**TCPW**	**PETRA**
Total Throughput (Mbps)	21.63	81.23	**99.33**
Average Delay (ms)	**33.69**	114.344	90.16
Jitter Average (ms)	3.37	3.39	**3.31**
Packet Loss (%)	3.1%	3.2%	**3.0%**

To test the LTE performance over high bit error links, we used different values of the Bit Error Rate (BER) (0.000075, 0.00009, 0.0001, 0.00025, and 0.0005) over the access link (UE-eNB link). We repeated the experiment above and plotted the total throughput achieved using TCPW, NewReno, and PETRA, as seen in Figure 5.9.

Figure 5.9: Throughput results using different PER values

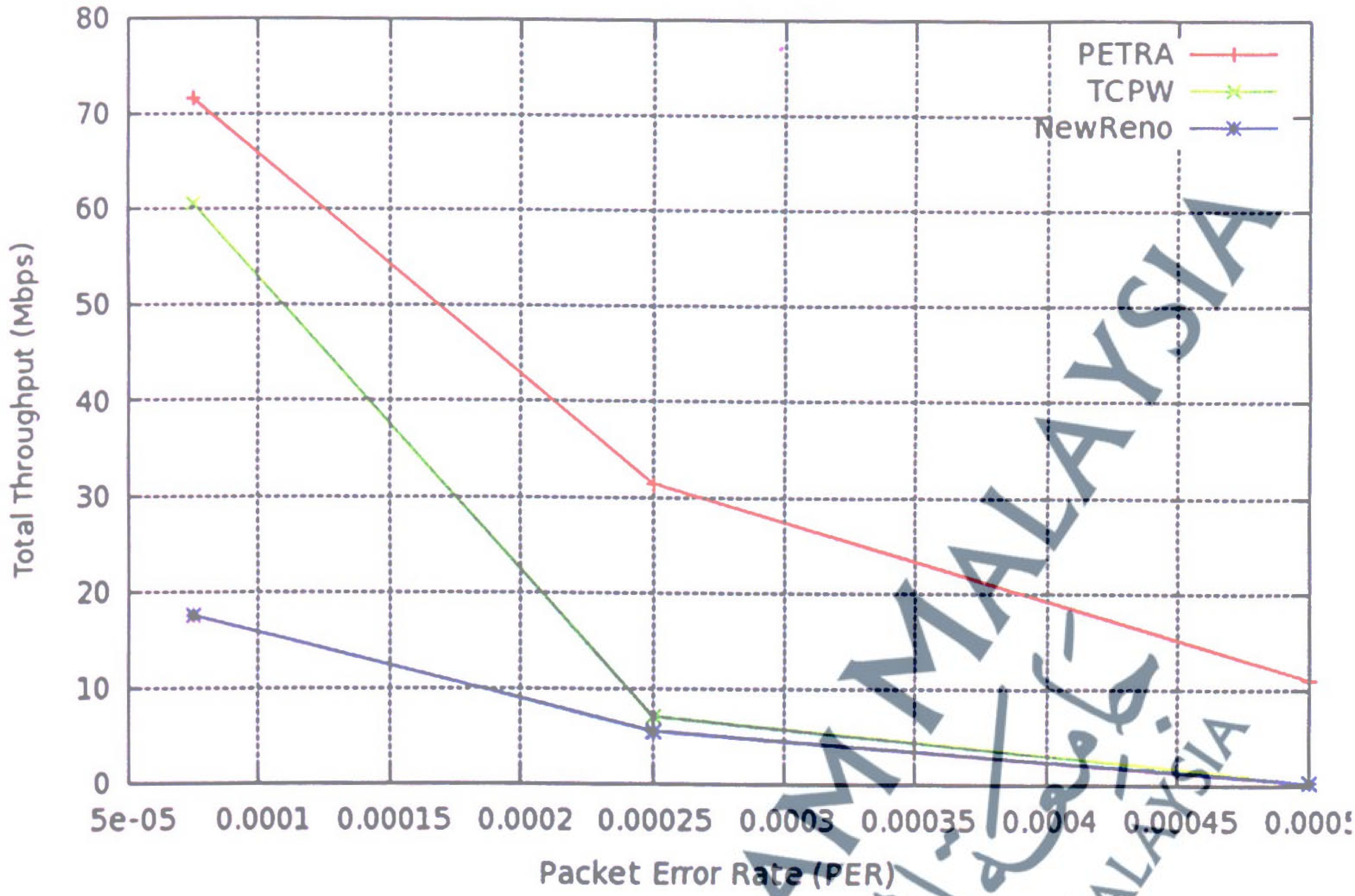


Figure 5.9 shows the significant throughput improvement that PETRA achieved over high PER values compared to other implementations. The throughput improvement is a result of the frequent use of the new fast retransmission and fast recovery besides the new retransmission procedures of the PETRA over loss links. Table 5.5 summarizes the experimental results.

Table 5.5: Simulation results using different PER values

PER	TCP Variants	Throughput (Mbps)	Average Delay (ms)	Average jitter (ms)	Loss ratio %
0.000075	PETRA	**71.62**	92.3	3.6	4.0%
	TCPW	62.53	104.8	3.37	4.2%
	NewReno	17.44	33.24	2.99	4.7%
0.00009	PETRA	**66.78**	93.7	3.49	4.4%
	TCPW	60.39	121.8	3.57	5.0%

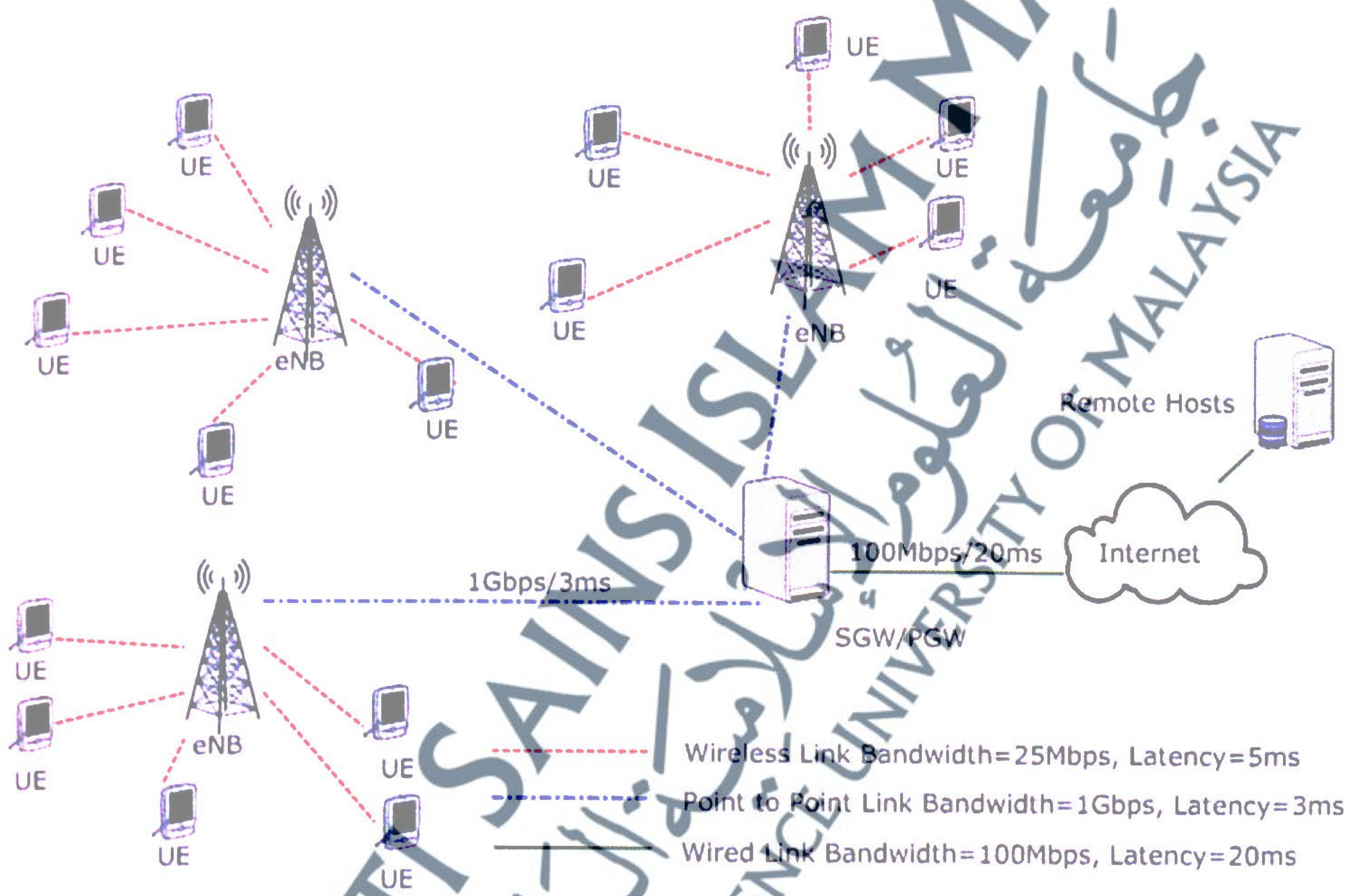
	NewReno	15.26	33.06	2.92	4.53%
0.0001	PETRA	**58.15**	94.5	3.36	5.23%
	TCPW	52.18	118.46	3.52	5.9%
	NewReno	14.15	32.95	2.88	5.4%
0.00025	PETRA	**31.35**	106.57	3.1	14.2%
	TCPW	7.02	131.48	3.74	15.02%
	NewReno	5.48	32.27	2.4	15.2%
0.0005	PETRA	**10.9**	85.4	3.6	30.9%
	TCPW	0.13	31.58	1.8	30%
	NewReno	0.13	31.5	1.8	40%

Table 5.5 shows the results of 15 different simulation experiments. The performance of PETRA was tested over different PER values. The above table shows better throughput values of PETRA over TCPW and NewReno. However, NewReno recorded lower values for average delay and jitter, which could be explained as the shorter queuing time due to the lower number of packets the NewReno injected to link, compared to PETRA and TCPW. On the other hand, PETRA shows better performance in term of packet loss ratio in comparison with other implementations. One more observation is that PETRA shows significant improvements over high PER values as in 0.00025 and 0.0005, where PETRA recoded at PER of 0.00025 a total throughput of 31.35mbps, which is 4.46 times as TCPW and 5.72 times as NewReno total throughput values.

5.4.2 Experiment 5-2

In this experiment, we extended the first topology used in experiment 5-1 to include more UEs and eNBs. The goal of this experiment is to test the PETRA congestion control mechanism in a congested LTE network. Also, in this experiment we tested PETRA fairness between the 15 flows shown in Figure 5.10.

Figure 5.10: Experiment 5-2 simulation topology



As we can see in the above figure, the simulation topology contains three eNBs connected to the SGW/PGW gateway using a point-to-point link with large data transmission rate of 1Gbps and latency of 3ms. Each eNB serves five UEs using a radio link with a download transmission rate of 25Mbps and latency of 5ms. A TCP bulk send application was started at the remote host to generate continuance traffic targeted to all UEs. The link between the local host and the LTE SGW/PGW is a wired link with a transmission rate of 100Mbps and latency of 20ms. A drop-tail queue scheme was applied to all the links in the proposed topology. Also, RateErrorModel is used over the radio links between the UEs and eNBs to generate uniform distributed transmission errors to simulate the wireless link medium. Table 5.6 summarizes the simulation parameters.

Table 5.6: Experiment 5-2 simulation parameters

Parameter	Value
Number of UEs	15
Number of eNB	3
Packet Size	1500 bytes
Queue scheme	Drop-Tail
Simulation period	100 seconds
Error Rate	0.00005
Application type	Bulk send application
TCP Protocols	NewReno, TCPW, PETRA

5.5.2.1 Simulation results

We used the same simulation parameters listed in Table 5.6 to compare the performance of PETRA with TCPW and NewReno. Table 5.7 summarizes the simulation results. We referred to every flow by a unique ID and recorded the total throughput, average delay, and loss ratio for the 15 flows using the three implementations of PETRA, TCPW and NewReno.

Table 5.7: Experiment 5-2 results

Flow ID	Total Throughput (Mbps)			Average Delay (ms)			Loss Ratio %		
	PETRA	TCPW	NewReno	PETRA	TCPW	NewReno	PETRA	TCPW	NewReno
1	21.57	18.78	4.92	45.60	45.55	45.50	0.41	0.42	0.43
2	21.42	18.46	4.49	45.61	45.59	45.49	0.42	0.44	0.47
3	20.27	18.08	4.97	45.62	45.63	45.50	0.44	0.48	0.41
4	21.32	18.02	4.05	45.62	45.63	45.49	0.43	0.48	0.51
5	20.99	18.33	4.56	45.66	45.57	45.50	0.49	0.45	0.47
6	21.36	17.80	3.77	45.61	45.64	45.52	0.43	0.50	0.66
7	21.04	17.35	4.07	45.67	45.66	45.50	0.45	0.51	0.51
8	21.16	19.24	4.12	45.68	45.50	45.50	0.44	0.39	0.50
9	21.06	16.85	4.07	45.67	45.70	45.50	0.45	0.52	0.51
10	21.55	18.72	4.06	45.60	45.55	45.50	0.41	0.42	0.51
11	21.13	18.74	4.66	45.68	45.55	45.49	0.44	0.42	0.47
12	21.25	18.06	5.50	45.62	45.63	45.48	0.44	0.48	0.37
13	21.07	18.04	3.79	45.67	45.64	45.50	0.45	0.48	0.65
14	21.19	18.61	4.60	45.68	45.56	45.49	0.44	0.43	0.46
15	20.76	18.35	5.24	45.67	45.62	45.49	0.48	0.45	0.34
	317.14	**273.43**	**66.87**	**684.66**	**684.02**	**682.45**	**6.62%**	**6.87%**	**7.27%**

The simulation results in the above table show that PETRA recorded the best throughput and the lower packets loss ratio of all the implementations; however, NewReno recorded the best average delay values for all the 15 flows.

The flow throughput values in Table 5.7 were used to test the fair bandwidth sharing of the new congestion control mechanism using Jain's fairness index, as discussed in chapter 3, which gives the following equation:

$$Fairness\,(throughput) = \frac{\left(\sum_{i=1}^{n} T_i\right)^2}{n\left(\sum_{i=1}^{n} T^2_i\right)} \tag{5.1}$$

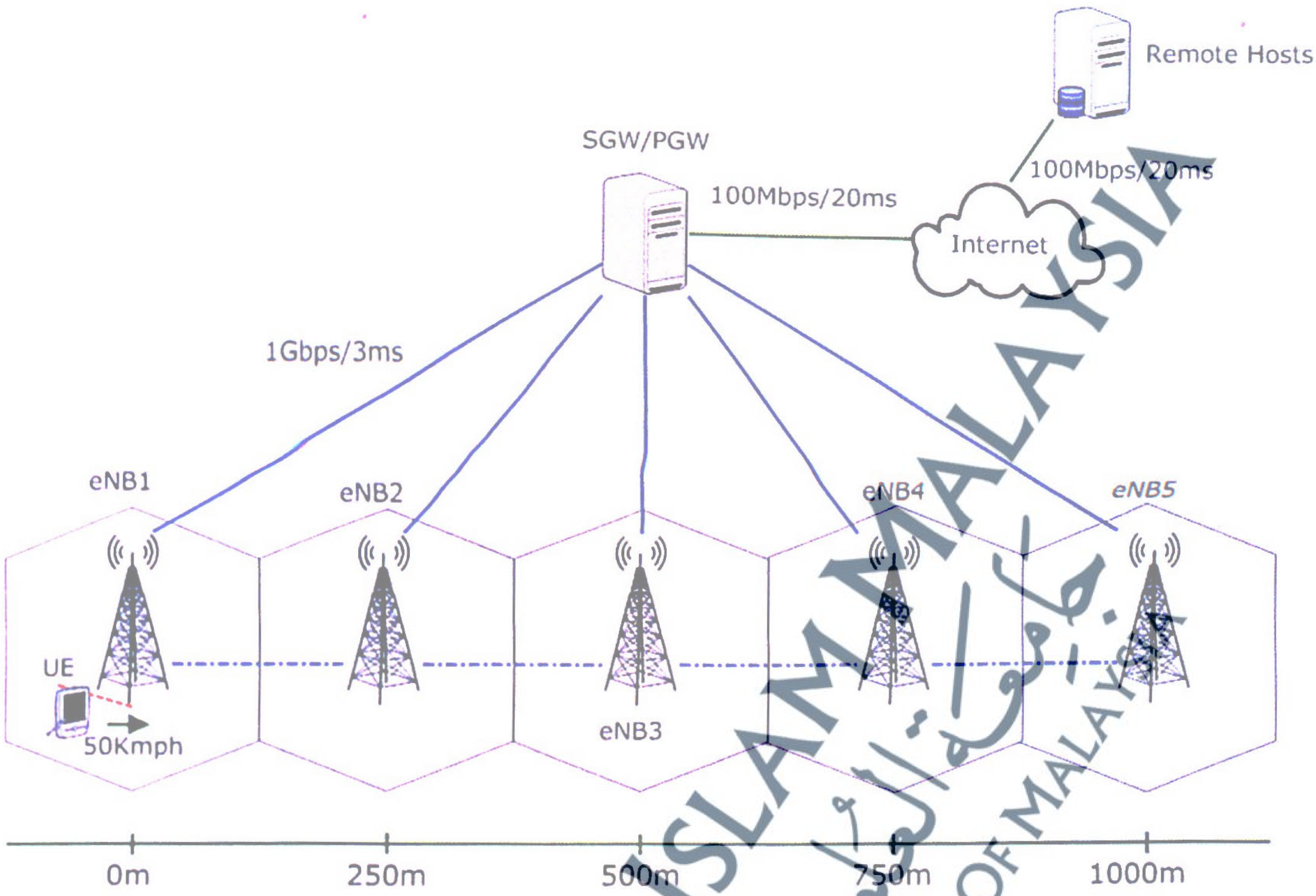
Thus, PETRA Fairness = 0.99978

According to the fairness index calculated above, 99.978% of the bandwidth is fairly shared among the 15 connections.

5.4.3 Experiment 5-3

In this experiment, we tested the performance of the new modification over a mobile UE in LTE network topology. We extended the first experiment topology to include the mobility feature to the UE. Four new eNBs will be added to the proposed topology so that the UE will be handover from eNB1, eNB2, eNB3, eNB4, to eNB5. The mobility of UE will be a linear movement along the x-axis. Figure 5.11 describes the proposed topology.

Figure 5.11: Experiment 5-3



The UE is initially attached to eNB1 and moves at a constant velocity of 50kmph along the x-axis. The UE will be handed over to the nearest eNB during the simulation period of 100 seconds. The links parameters will be the same as the first experiment parameters listed in Table 5.2. The simulation parameters are listed in the following table.

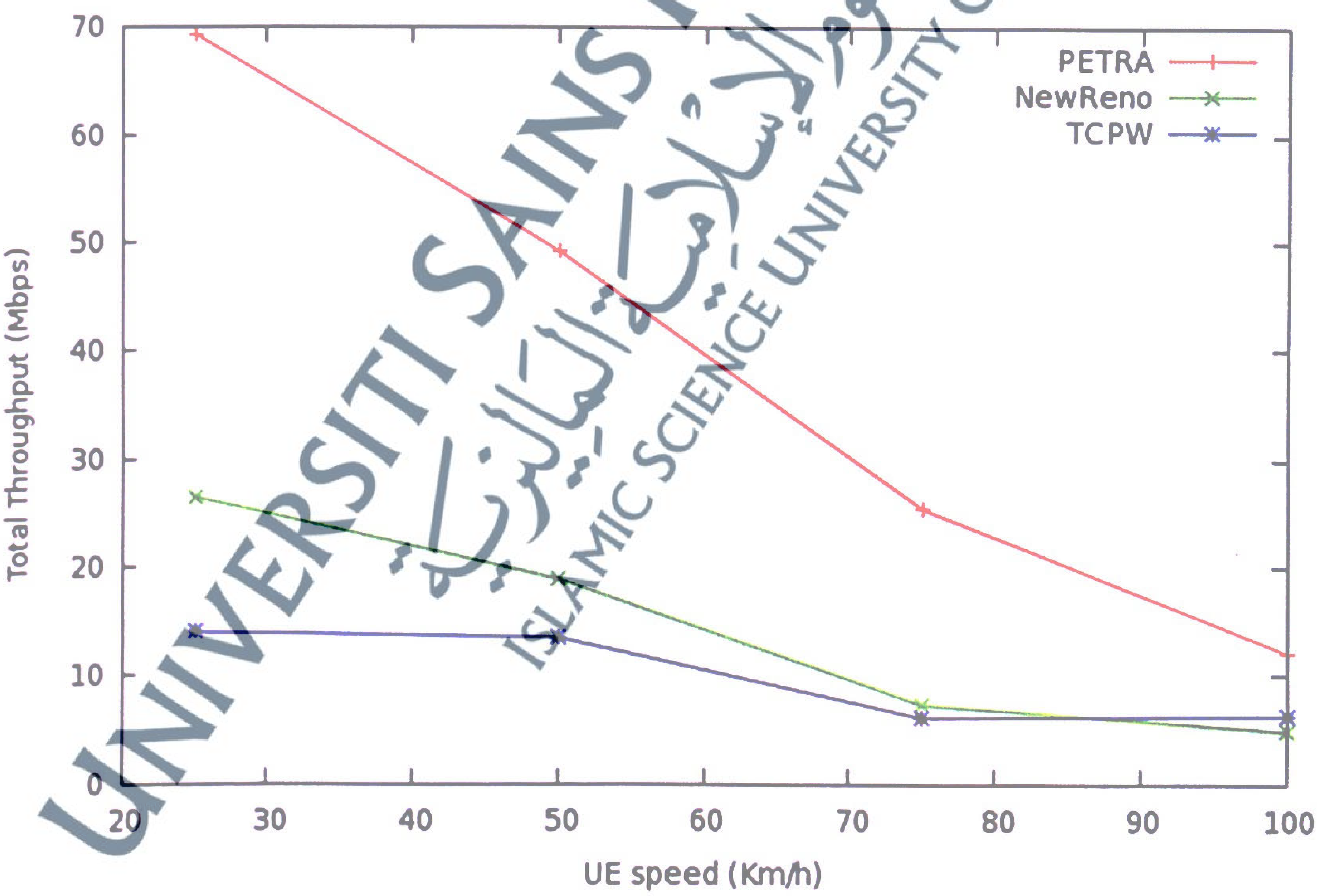
Table 5.8: Experiment 5-3 simulation parameters

Parameter	Value
Number of UEs	1
Number of eNB	5
Packet Size	1500 bytes
Queue scheme	Drop-Tail
Simulation period	100 seconds
Application type	Bulk send

	application
Mobility	Constant velocity at 25, 50, 75, 100 km/h
Cell dimension	250m
TCP Protocols	NewReno, TCPW, PETRA

Figure 5.12 shows the total throughput achieved using the three implementations over varies UE speeds. As expected, the total throughput is significantly degraded over the UE high mobility, as shown in the following figure.

Figure 5.12: Experiment 5-3 throughput results



The above figure shows significant throughput improvement of PETRA over other implementations. PETRA benefits from the new modifications by maximizing the bandwidth utilization over the high loss environment. Firstly, PETRA's faster start phase maximizes the bandwidth utilization every time a handover occurs. Secondly, the faster retransmission and best recovery is frequently triggered because of the high packet loss ratio. This will preserve greater *cwnd* size where the two modifications are based on the calculated bandwidth and the *RTTs* values. Finally, the time out procedure of the new modification keeps the *cwnd* size as large as the link status. Table 5.9 summarizes the results of experiment 5-3.

Table 5.9: Experiment 5-3 results

	Average Delay (ms)				Packet Loss Ratio (%)			
UE speed	**25**	**50**	**75**	**100**	**25**	**50**	**75**	**100**
PETRA	8.43	7.80	7.54	7.18	9.20	13.01	25.21	30.53
NewReno	6.90	4.13	3.35	3.30	7.37	11.08	30.45	35.24
TCPW	5.30	8.82	7.20	7.39	15.31	25.11	32.44	40.63

The simulation results in the above table show the packet loss ratio and the average delay values of the three implementations during the 100 second simulation period. PETRA recorded some longest delay values compared to NewReno and TCPW. One reason for this could be the long queuing time because of the large number of packets transmitted using PETRA compared to other implementations. On the other hand, PETRA recorded the lowest packet loss ratio for all the experiments.

5.5 Summary

This chapter introduced the implementation and evaluation of the new congestion control mechanism PETRA over LTE networks. Three different experiments have been discussed in this chapter to evaluate PETRA congestion control over LTE networks. The LENA-LTE model was used to implement the LTE simulated environment. Three different scenarios were used in experiments 5-1, 5-2, and 5-3. Simulation results showed significant improvement of PETRA performance compared to NewReno and TCPW implementations over high mobility, low-latency, and wide-bandwidth links.