OBJECTIVES

The objective of the project is to build, simulate and configure different types of queuing like best effort, priority queue, round robin queue, weighted round robin queue and token-bucket queue. These queuing mechanisms are to be simulated to campare between all queuing types from different Quality of service (QOS) prespective. Traffic has to be classified, marked, and treated based on its characteristics such as time sensitive traffic voice and video and non-time sensitive traffic FTP, HTTP etc.

INTRODUCTION

This is Phase 3 of project in which *Project 1: Multiple Quality of Service Queuing* has been implemented and simulated in Omnet++ Simulator . In this project , we are deploying different types of queuing like best effort, priority queue, round robin queue and weighted round robin queue. The router traffic is being treated based on its characteristics such as time sensitive traffic voice and video and non-time sensitive traffic FTP, HTTP etc. The concept and theory behind the implementation are as follows :

Priority Queuing(PQ)

In Priority Queuing, instead of using a single queue, the router bifurcates the memory into multiple queues, based on some measure of priority. After this, each queue is handled in a FIFO manner while cycling through the queues one by one. The queues are marked as High, Medium, or Low based on priority. Packets from the High queue are always processed before packets from the Medium queue. Likewise, packets from the Medium queue are always processed before packets in the Normal queue, etc. As long as some packets exist in the High priority queue, no other queue's packets are processed. Thus, high priority packets cut to the front of the line and get serviced first. Once a higher priority queue is emptied, only then is a lower priority queue serviced.

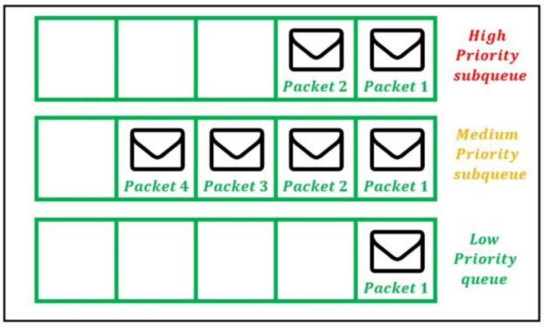


Figure 1 Priority Queuing model

First-In, First-Out Queuing (FIFO)

The default queuing scheme followed by most routers is FIFO. This generally requires little no configuration to be done on the server. All packets in FIFO are serviced in the same order as they arrive in the router. On reaching saturation within the memory, new packets attempting to enter the router are dropped (tail drop). Such a scheme, however, is not apt for real-time applications, especially during congestion. A real-time application such as VoIP, which continually sends packets, may be starved during times of congestion and have all its packets dropped.

Round Robin Queuing (RR)

RR is a queuing discipline that is quite a contrast to priority queuing. In simple RR, we have a few queues, and we assign traffic to them. The RR scheduler processes one packet from one queue and then a packet from the next queue and so on. Then it starts from the first queue and repeats the process. No queue has priority over the others, and if the packet sizes from all queues are (roughly) the same, effectively the interface bandwidth is shared equally among the RR queues. With RR, no queue is in real danger of starvation, but the limitation of RR is that it has no mechanism available for traffic prioritization.

Weighted Round Robin Queuing (WRR)

A modified version of RR, Weighted Round Robin (WRR), allows us to assign a "weight" to each queue, and based on that weight, each queue effectively receives a portion of the interface bandwidth, not necessarily equal to the others. Custom Queuing (CQ) is an example of WRR, in which we can configure the number of bytes from each queue that must be processed before it is the turn of the next queue.

Token-Bucket Queuing

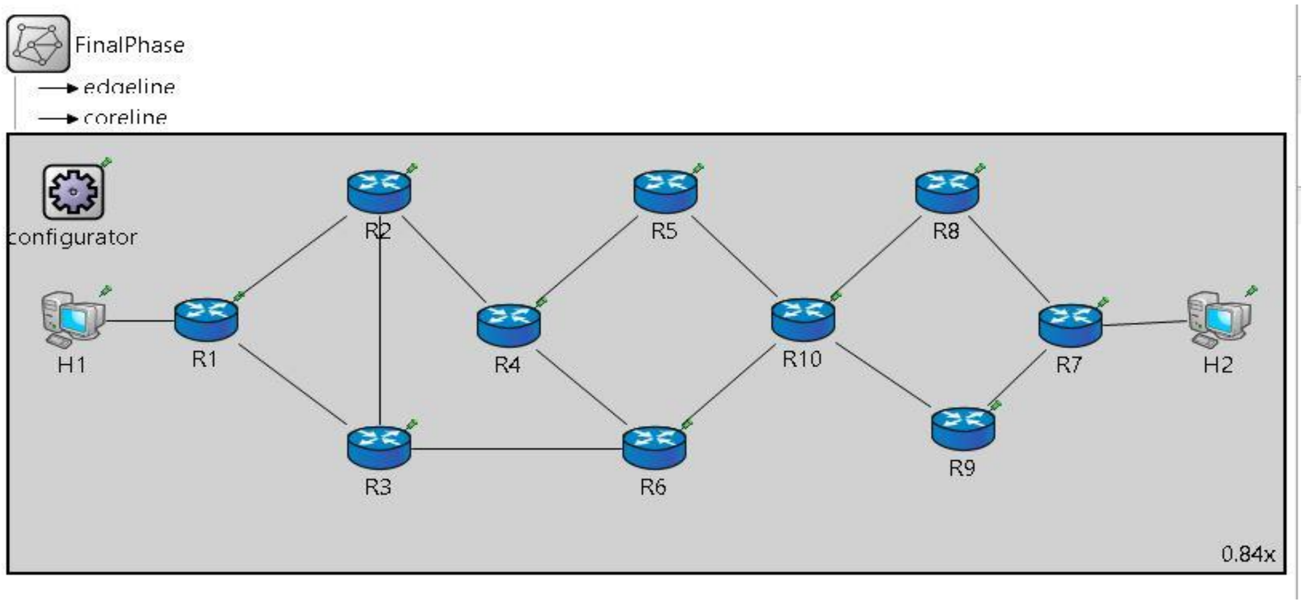
The Token-Bucket compound module implements the token bucket algorithm. By default, the module contains a drop-tail queue and a token based server. The queue capacity and the token consumption of the token-based server can be used to parameterize the token bucket algorithm. The module uses an external token generator for the token-based server. The packets are pushed into the bucket and a token generator generates the token periodically into the token bucket module.

These queuing and scheduling mechanism is applied on multiple routers in the topology which is further discussed in great detail in next section. To imitate the real network scenario, topology is adjusted in such a way that destination is reachable via multiple paths so that if any route fails OSPF routing can learn a new path to its destination. To examine the effectiveness of each scheduling and queuing mechanism three different types of applications(video, audio, and data) is being tested and corresponding to which metrics like 1) end to end delay, 2) queueing delay, 3) packets dropped, 4) packets send and received is measured

RESULTS

In this section, the setup of **topology** is discussed in detail.

The network to be simulated here is common for both Phase2 and Phase 3 simulations. It comprise of Ten routers (R1-R10) and two hosts (H1-H2) which is connected with interfaces of different speeds. The link between R2-R3, R2-R4,R3-R6, R4-R5, R4-R6, R5-R10, R10-R9 R6-R10 and R8-R10 are linked with core datarate links of 16Kbps and the links between R1-R2, R1-R3, R7-R8, R7-R9 are linked with interfaces of datarate 32Kbps



Multiple paths to destination :

- R1-R2-R4-R5-R10-R8-R7-H2
- R1-R3-R2-R4-R5-R10-R8-R7-H2
- R1-R3-R6-R10-R8-R7-H2

For simulation, the .ned files are made to configure the topology . After this, general parameters are configured like :

- Simulation time- 1250s
- Queue packet capacity- 100
- OSPF- enabled via xml doc
- Scalar and vector recordings- True

To simulate QOS parameters, different traffic has been set up called app(0,1,2) as UdpBasicBurst (to simulate a voice streaming), UdpBasicApp (to simulate a video streaming) and TcpBasicClientApp (to simulate a regular web streaming) respectively.

Now to measure the parameters and traffic passing through the node, Traffic classifiers and markers **TC1** are used at Router R1 and R7 and all result recording modes are enabled like :

- Sent packet
- Received packet
- Dropped packet
- Queuing time
- End to end delay

These Queueing and Scheduling mechanism, statistical measurements, apps and result recording modes are all configured in **Omnetpp.ini** files .

The results obtained after the simulation for each application are as follows :-

Exp1- Best effort Queuing, Exp2- Best effort with Policing and Shaping, Exp3- Weighted Round Robin

Exp4- Weighted Round Robin with Policing and shaping, Exp5- Priority Queueing

Exp6- Priority Queuing with Policing and Shaping, Exp7- Round Robin Queuing

Exp8- Round Robin with Policing and Shaping, Exp9- Token Bucken Queuing

Table 1 shows comparison between multiple Quality of service queuing for App[0}

	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6	Exp 7	Exp 8	Exp 9
Total Sent (packets)	21314	21314	21314	21314	21314	21314	21314	21314	21314
Total received (packets)	800	975	2151	1597	236	319	1332	1628	2109
End-to-End delay(sec)	34.9	37.54	3.3	2.53	4.396	3.817	4.94	5	0.66
Queueing time(R1)	7.98	7.92	7.268	7.227	9.49	9.51	10	9.84	0.006
Packets Dropped	20514	20339	19163	19717	21078	20995	19982	19686	19205

Table 2 shows comparison between multiple Quality of service queuing for App[1}

	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6	Exp 7	Exp 8	Exp 9
Total Sent (packets)	29961	29961	29961	29961	29961	29961	29961	29961	29961
Total received (packets)	4600	4052	4061	4063	4715	4070	4071	3930	499
End-to-End delay(sec)	45.16	49.19	66.38	44.36	52.56	52.2	87.39	91.164	1.38

Table 3 shows comparison between multiple Quality of service queuing for App[2}

	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6	Exp 7	Exp 8	Exp 9
Total Sent (packets)	2	3	3	3	1	2	2	2	79
Total received (packets)	1	2	2	1	1	1	2	2	79
End-to-End delay(sec)	0.215	6.97	6.90	0.2	0.2	0.215	2.416	2.24	1.811

Table 4 shows Queuing time(sec) for real time application on three different queues of priority Queuing

Exp 5	Queue[0]	Queue[1]	Queue[2]
App[0]- R1	0.22s	2.17s	26.08s
App[1]- R1	0.38s	5.92s	12.85s
App[2]- R2	0.13s	5.20s	16.68s

Table 5 shows Queuing time(sec) for real time application on three different queues of priority Queuing with Policing and shaping

Exp 6	Queue[0]	Queue[1]	Queue[2]
App[0]-R1	0.20s	2.10s	26.03s
App[1]-R1	0.35s	5.8s	13.6s
App[2]-R2	0.04s	0.2s	0.57s

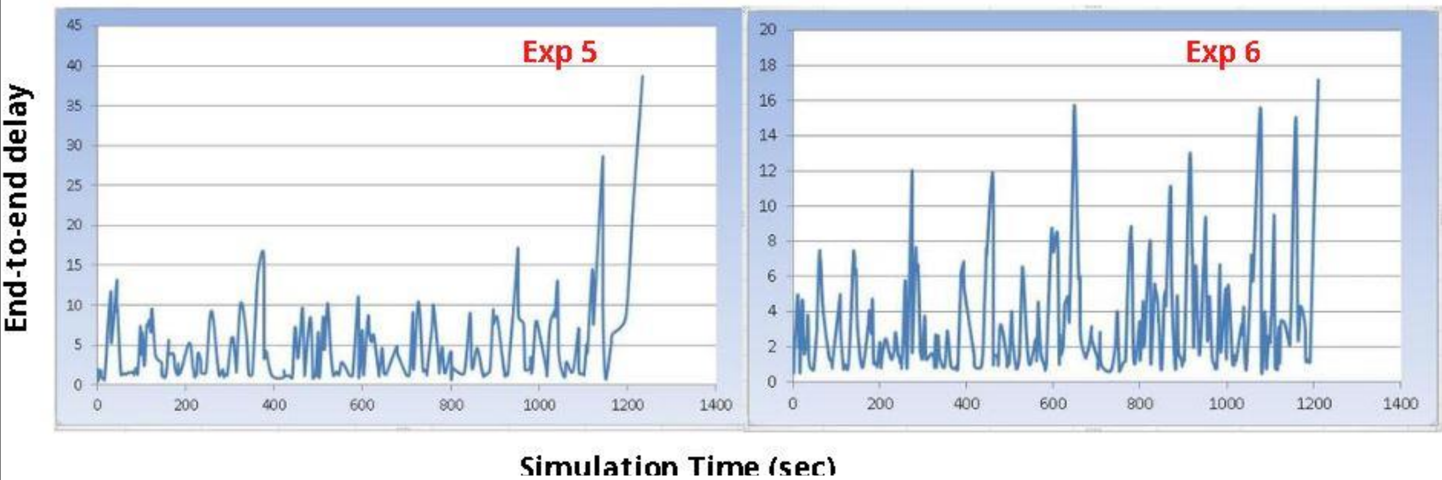


Figure 2 shows end-to-end delay comparison between Priority Queuing and priority Queuing with Policing and Shaping for App[0]

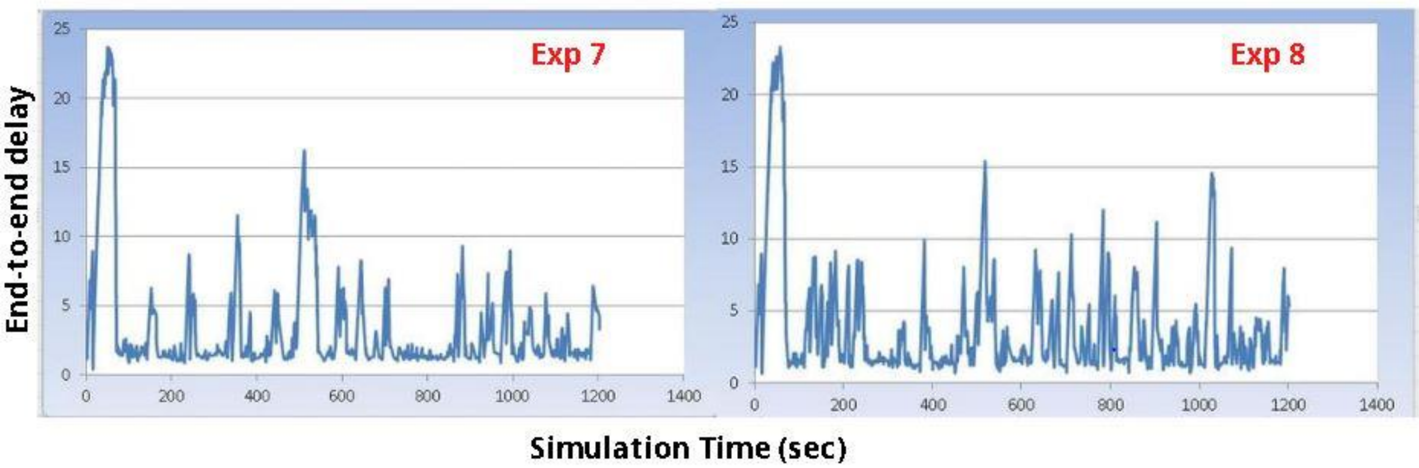


Figure 3 shows end-to-end delay comparison between Round Robin Queuing and Round Robin Queuing with Policing and Shaping for App[0]

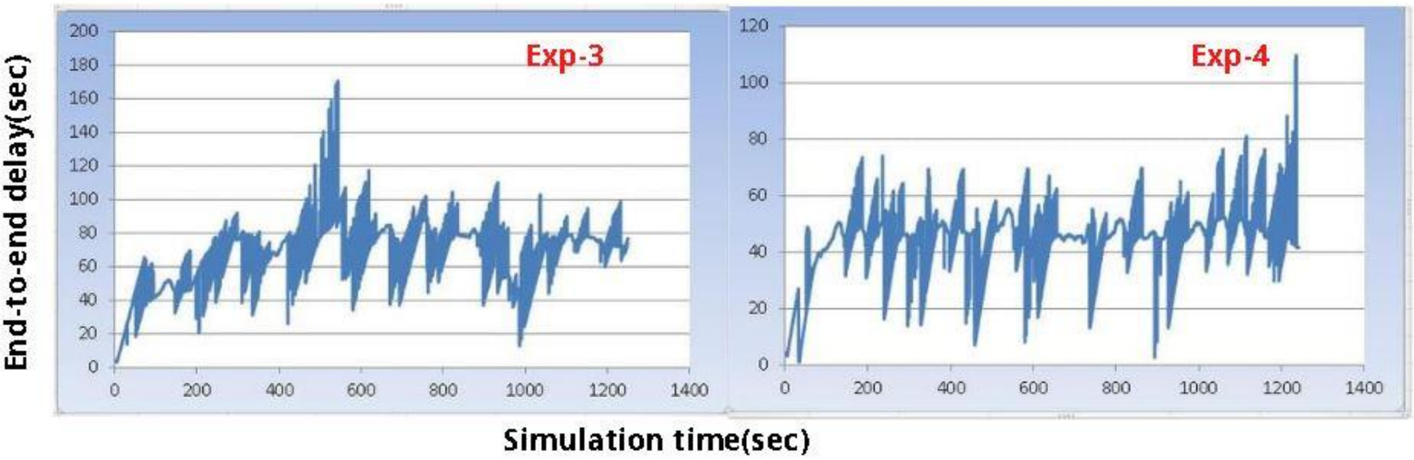


Figure 4 shows end-to-end delay comparison between weighted Round Robin Queuing and weighted Round Robin Queuing with Policing and Shaping for App[1]-Video streaming

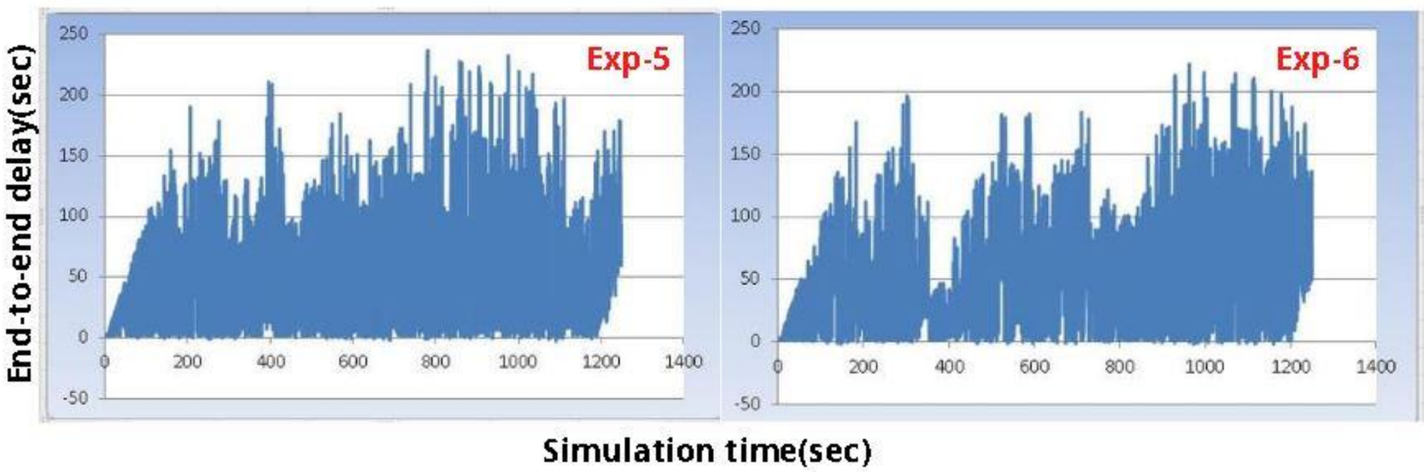


Figure 5 shows end-to-end delay comparison between Priority Queuing and Priority Queuing with Policing and Shaping for App[1]-Video streaming

Table 6 shows comparison between various queuing technique based on efficiency for App[0]

	Exp1 : Best effort	Exp :7 Round Robin	Exp :3 Weighted RR
Packets Sent	21314	21314	21314
Packets Received	800	1332	2151
End-to-end delay(sec)	34.9	4.94	3.3
Queuing time(R1)	7.98	10	7.268

DISCUSSION

Multiple metrics like packet sent, packet received, End-to-end delay and Queuing delay has been considered for comparing the efficacy of different Quality of sevice queuing based on the traffic type. These comparison is divided into multiple section to present more holistic view of the objective of the project . They are presented as follows :

- ❖ Comparison between multiple Queuing technique(without QOS) and their effect on different applications.
- ❖ Comparison based on Quality of service applied (with policing and shaping) to different application.
- ❖ Comparison based on the role and working models of different Queuing technique.

1. Comparison between multiple Queuing technique(without QOS) and their effect on different applications.

For App[0] : Voice streaming

- For simulating voice streaming application the total packets sent by host is 21314(count) and the total packets received by the end host depends on the type of queuing and scheduling applied. The maximum number of packets received is seen to be maximum in Exp3- Weighted Round Robin which is 2151 packets which can be seen from Table1 .
- There is a large amount of packet received seen because the weights in weighted round robin are configured as 40,10,5,1,1 which implies that more bandwith is provided for app[0] i.e voice streaming hence less packets were dropped.
- The end-to-end delay seen from simulation data mentioned in Table 1 implies that the least end-to-end delay is incured in Exp1- Best effort Queuing when compared to other Queuing counterparts(only considering without QOS case) which is 3.3sec.
- This low end-to end delay shows that packets are getting received faster at the end host which also impies that this method of Queuing is more efficient for Voice streaming.
- Queuing time here is measured on R1 and R7 since other routers are not taking much time on queuing. The queuing time seen is least in Exp3- Weighted Round Robin case which is also evident from table 1.

For App[1] : Video streaming

- For simulating voice streaming application the total packets sent by host is 29961(count) and the total packets received by the end host depends on the type of queuing and scheduling applied. The maximum number of packets received is seen to be maximum in Exp1- Best effort Queuing which is 4600 packets which can be seen from Table2 (without QOS).
- There is a large amount of packet received seen in Best effort queuing because of the use of UDP as transport protocol which provides best effort service.
- The end-to-end delay seen from simulation data mentioned in Table 2 implies that the least end-to-end delay is incured in Exp1- Best effort Queuing when compared to other Queuing counterparts(only considering without QOS case).
- This low end-to end delay shows that packets are getting received faster at the end host which also impies that this method of Queuing is more efficient for Video streaming.

For App[2] : Web streaming

- For simulating web streaming application the maximum packets sent by host is 3(count) and the total packets received by the end host depends on the type of queuing and scheduling applied. The maximum number of packets received is seen to be maximum in Exp5- priority Queuing which is 1 packet received compared to 1 packet sent which can be seen from Table3 .

- The end-to-end delay seen from simulation data mentioned in Table 3 implies that the least end-to-end delay is incured in Exp5- Priority Queuing when compared to other Queuing counterparts(only considering without QOS case).
- This low end-to end delay shows that packets are getting received faster at the end host which also impies that this method of Queuing is more efficient for Web streaming.
- Queuing time for Priority queuing in their sub-queues are mentioned in table 4 which shows that Priority queuing is getting very less queuing delay for App[2] which is 0.13sec at R2.

2. Comparison based on Quality of service applied (with policing and shaping) to different application.

For App[0] : Voice streaming

- When policing and shaping is applied to R2 ingress and R4 egress router a fair amount of increase in performance is witnessed in various queuing and scheduling mechanism as shown in Table 1, Figure 2 and figure 3.
- In figure 2 when Exp5- Priority Queueing and Exp6- Priority Queuing with Policing and Shaping, are compared . It can be seen from the scatter plot that end-to-end delay has greatly reduced when policing and shaping has been applied which shows that efficiency with QOS has increased compared to without QOS.
- In figure 3, Exp7- Round robin Queueing and Exp8- Round robin with Policing and Shaping, are compared . It can be seen from the scatter plot that end-to-end delay has greatly reduced in this case too when policing and shaping has been applied which shows that efficiency with QOS has increased compared to without QOS.
- From the graphs in Figure 2 and 3 we can observe that after applying policing and shaping the graph is more compact showing more readings of packets in a particular time of simulation. This means that we are receiving more packets after applying policing and shaping.

*For App[1] : Video streaming*When policing and shaping is applied to R2 ingress and R4 egress router a fair amount of increase in performance is witnessed in various queuing and scheduling mechanism as shown in Table 1, Figure 4 and figure 5.

- In figure 4 when Exp3- Weighted Round Robin and Exp4- Weighted Round Robin with Policing and shaping, are compared . It can be seen from the scatter plot that end-to-end delay has greatly reduced when policing and shaping has been applied which shows that efficiency with QOS has increased compared to without QOS.
- In figure 5, Exp5- Priority Queueing and Exp6- Priority with Policing and Shaping, are compared . It can be seen from the scatter plot that end-to-end delay has greatly reduced in this case too when policing and shaping has been applied which shows that efficiency with QOS has increased compared to without QOS.

3. Comparison based on the role and working models of different Queuing technique.

- Here queuing technique is compared between multiple queuing models presented in the project for example Best effort, round robin , Weighted round robin, Priority queuing etc.

- Best effort and priority queue is based on Droptail queuing mechanism and hence large packet drop and less packet received were seen for each application compared to Round robin queuing and Weighted round robin which is shown in Table 6.
- Refering to table 6 , weighted round robin efficiency is seen to be more compared to its counter-part round robin mechanism particularly for app[0] because weighted round robin comprises of weights that are assigned to provide more capacity to a particular type of traffic and hence less packet drop and small end-to-end delay.
- Comparing the queuing time at Router R1 it is also evident less queuing delay is incurred in weighted round robin compared to other methods.

CONCLUSION

Thus from all the data and experiments recorded it can be concluded that there is a great increase in network performance and efficiency seen with QOS compared to queuing models without QOS(Quality of service). It can also be inferred that overall Weighted round robin with Policing and Shaping has shown the highest performance in terms of low end-to-end delay for voice streaming, Video streaming and web streaming applications compared to all other queuing models.

While performing these experiments and simulations I learned various queuing and scheduling mechanism and the need for Quality of Service (QOS) to increase the network performance. I also come to know about various network parameters to take in account while considering network performance and efficacy and also the need for low latency and high efficiency for real time traffic like voice and video.

REFERENCES

[1] https://inet.omnetpp.org/docs/index.html.

[2] https://omnetpp.org/documentation/.

[3] https://users.soe.ucsc.edu/~brad/pubs/icccn08.pdf.

[4] https://www.ccexpert.us/qos-implementing/weighted-round-robin-1.html

[5] https://sites.google.com/a/seecs.edu.pk/network-technologies-tcp-ip-suite/home/performance-analysis-of-impact-of-various-queuing-mechanisms-on-mpeg-traffic/working-mechanism-of-fq-red-sfq-drr-and-drop-tail-queues

[6] https://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-policing/19645-policevsshape.html

[7] https://ipcisco.com/lesson/policing-and-shaping/

[8]https://moodle.concordia.ca/moodle/pluginfile.php/5304298/mod_resource/content/1/ELEC6181%20Project%20Phase%20One.pdf

[9] https://www.routerfreak.com/traffic-policing-vs-traffic-shaping/.
[10] https://www.tutorialspoint.com/packet-queuing-and-dropping-in-routers.