Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

L3 + L4

Principles of Computer Communication: ECE3030

Allen Ben Philipose – 18BIS0043

LAB TASKS

Comparison of Voice Codecs

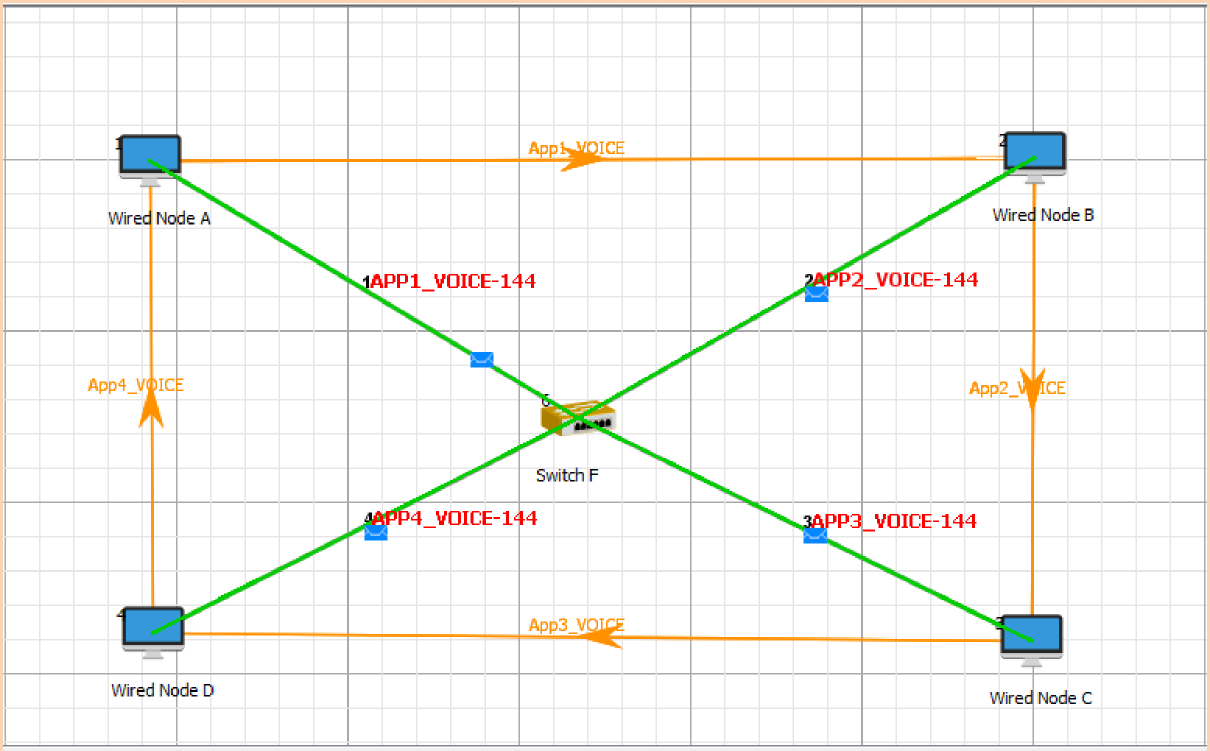
Aim

To simulate Networks using various voice codecs in Netsim and compare their performance.

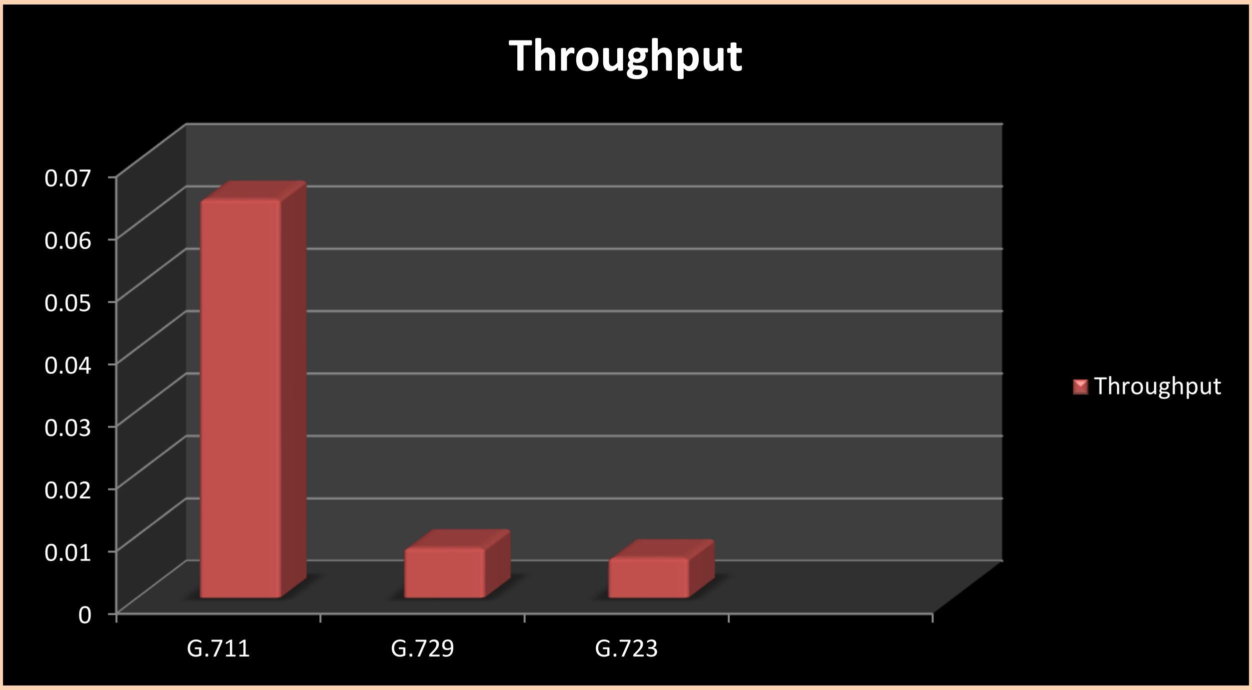
Tools Required

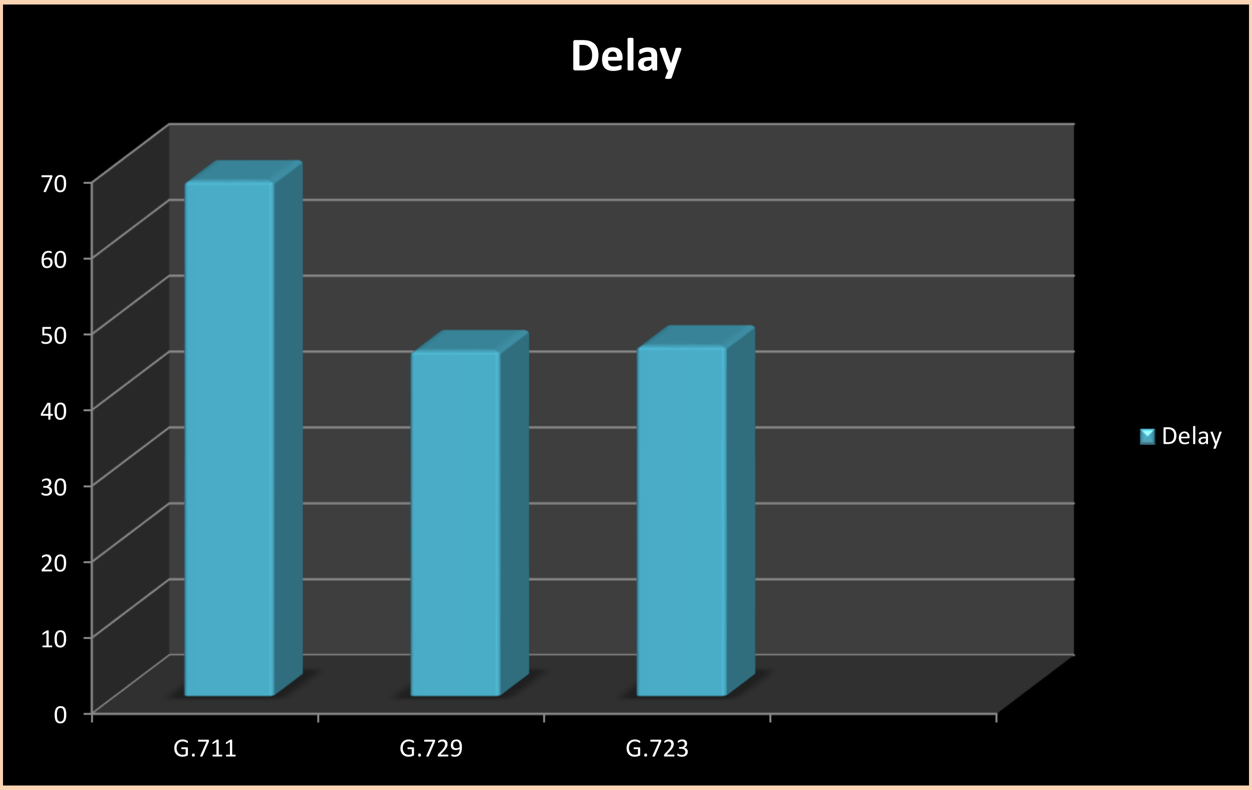
Netsim Software

Circuit Diagram

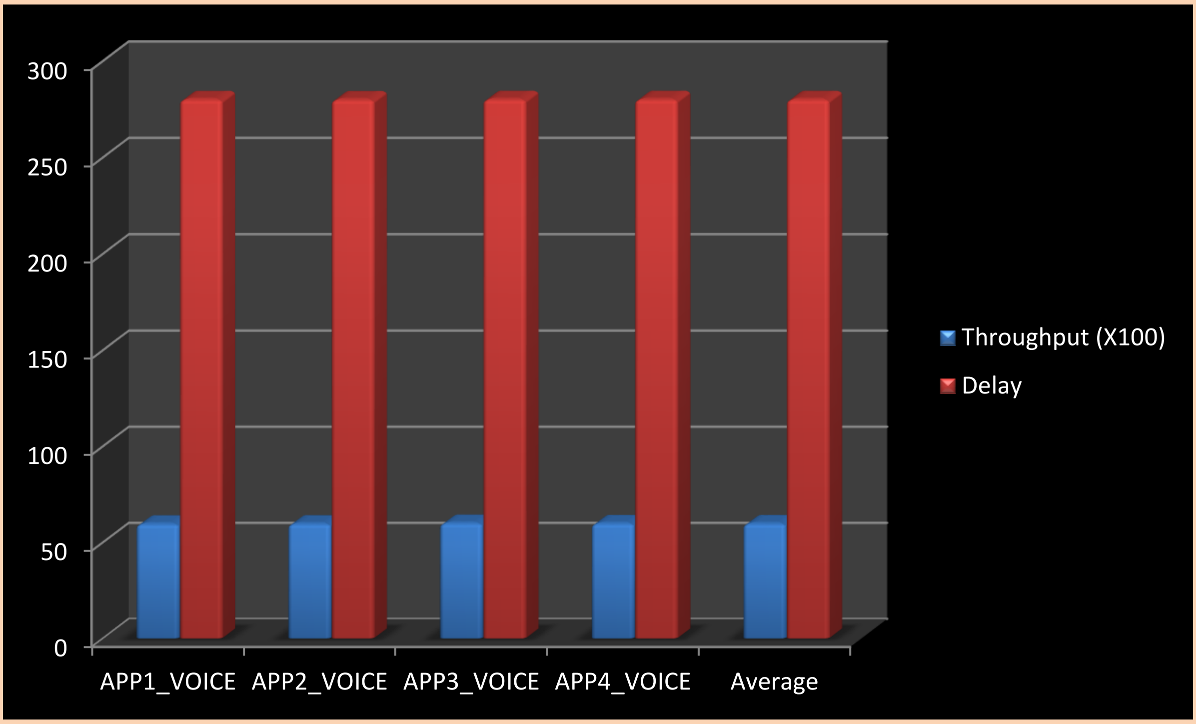


Graphs





"Custom" given as a separate graph and table



Tables

G.711		Throughput	Delay
	APP1_VOICE	0.063872	67.86
	APP2_VOICE	0.063744	67.86
	APP3_VOICE	0.063872	67.86
	APP4_VOICE	0.063872	67.86
	Average	0.06384	67.86

G.729		Throughput	Delay
	APP1_VOICE	0.007989	45.46
	APP2_VOICE	0.007979	45.46
	APP3_VOICE	0.007989	45.46
	APP4_VOICE	0.007989	45.46
	Average	0.0079865	45.46

G.723		Throughput	Delay
	APP1_VOICE	0.006387	46.1
	APP2_VOICE	0.006374	46.1
	APP3_VOICE	0.006387	46.1
	APP4_VOICE	0.006387	46.1
	Average	0.00638375	46.1

Custom		Throughput (X100)	Delay
	APP1_VOICE	59.5307	280.038111
	APP2_VOICE	59.6064	279.956343
	APP3_VOICE	59.92	280.1
	APP4_VOICE	59.7643	280.105668
	Average	59.70535	280.0500305

Calculations – G.711

Base Band frequency f_m = 4KHz

Bitrate = n x f_s

f_s = 2 x f_m = 8 KHz

Bitrate=8 x 8KHz = 64kbps

Transmission Time = X = L/R

= (160 x 8)/ 64 kbps

=20 ms

Packet Size = (Data Rate x Transmission Time)/8

= (64 x 20)/8

= 160 bytes

Payload = (Packet Transmitted x Packet Size)

= 499 x 160

= 79840

Throughput = (Payload x 8)/ (10 x 106)

= 64 kbps

Calculations – G.723

Base Band frequency f_m = 400 Hz

Bitrate = n x f_s

$f_s = 2 \times f_m = 800$ Hz

Bitrate =10 x 800Hz = 8 kbps

Transmission Time = X = L/R

= (24 x 8)/ 8 kbps

= 24 ms

Packet Size = (Data Rate x Transmission Time)/8

= (8 x 24)/8

= 24 bytes

Payload = (Packet Transmitted x Packet Size)

= 333 x 24

= 7992

Throughput = (Payload x 8) / (10 x 10^6)

= 6.394 kbps

Calculations – G.729

Base Band frequency f_m= 500 Hz

Bitrate = n x f_s

$f_s = 2 \times f_m = 1000$ Hz

Bitrate =8 x 1000Hz = 8 kbps

Transmission Time = X = L/R

= (20 x 8)/ 8 kbps

= 20 ms

Packet Size = (Data Rate x Transmission Time)/8

= (8 x 20)/8

= 20 bytes

Payload = (Packet Transmitted x Packet Size)

= 499 x 20

= 9980

Throughput = (Payload x 8) / (10×10^6)

= 7.984 kbps

Inference

Higher the throughput and the lower delay shows which is the better codec. The 'custom' codec has a significant difference of throughput when compared to the other codecs, but the delay increase it has also is very high. Hence, the best alternative that we have is the G.711 codec.

Voice codec G.711 has a very high throughput value and the delay, despite being high, is very comparable with the other codecs.

Custom:

Case 1: *Data Rate = 200kbps*

Assume,

Transmission Time = 20ms

X = L/R

L = 20 ms x 200 kbps = 4000 bits

Packet size = 4000/8 = 500 bytes

Case 2: *Data Rate = 60kbps*

Assume,

Transmission Time = 20ms

X = L/R

L = 20 ms x 60 kbps = 1200 bits

Packet size = 1200/8 = 150 bytes

Demonstration of switches and hubs

Aim

To simulate Networks using switches and hubs in Netsim and compare their performance.

Tools Required

Netsim Software

Important Formulae

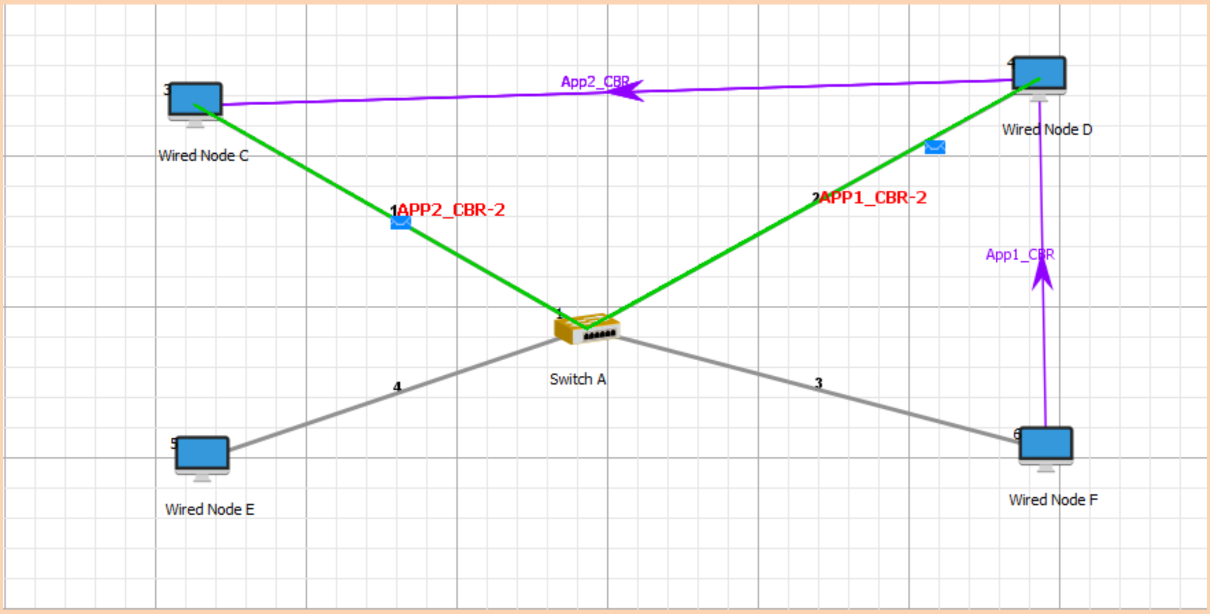
i. Payload Received = Packet Received x Packet Size
ii. Throughput (Mbps) = (Payload Received x 8) / (Simulation Time x 10^6)

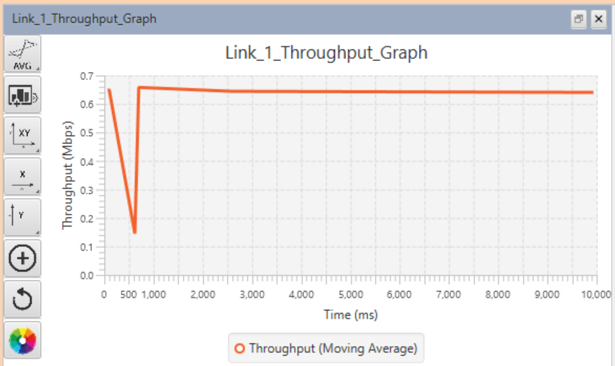
Abstract and Inference

From the previous experiment, it can be inferred that the hub connections have more collision of packets and hence the delay is very high. Also, the structure reduces the throughput, which essentially makes the switch a much efficient system in comparison among the both.

Hub works on a physical layer while the switch works on a data link layer. So, the hub transmits the data to all the destinations and the receivers must decide whether to accept the data or not. Switches on the other hand can control the channel of the message transmission.

Circuit Diagram with SWITCH – 4 Nodes





Application_Metrics_Table

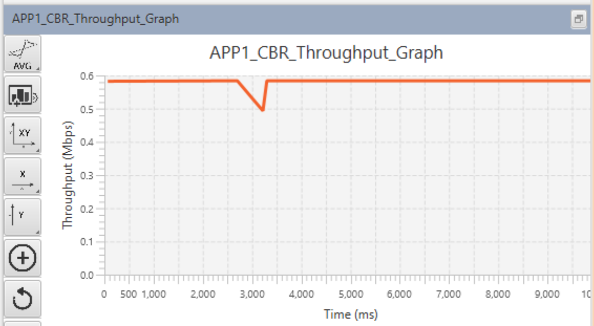
Application_metrics Detailed View

plication Id	Throughput Plot	Application Name	Throughput (Mbps)	Delay(microse
	Application_throughput_plot	APP1_CBR	0.582832	13414.544008
	Application_throughput_plot	APP2_CBR	0.582832	13430.371138

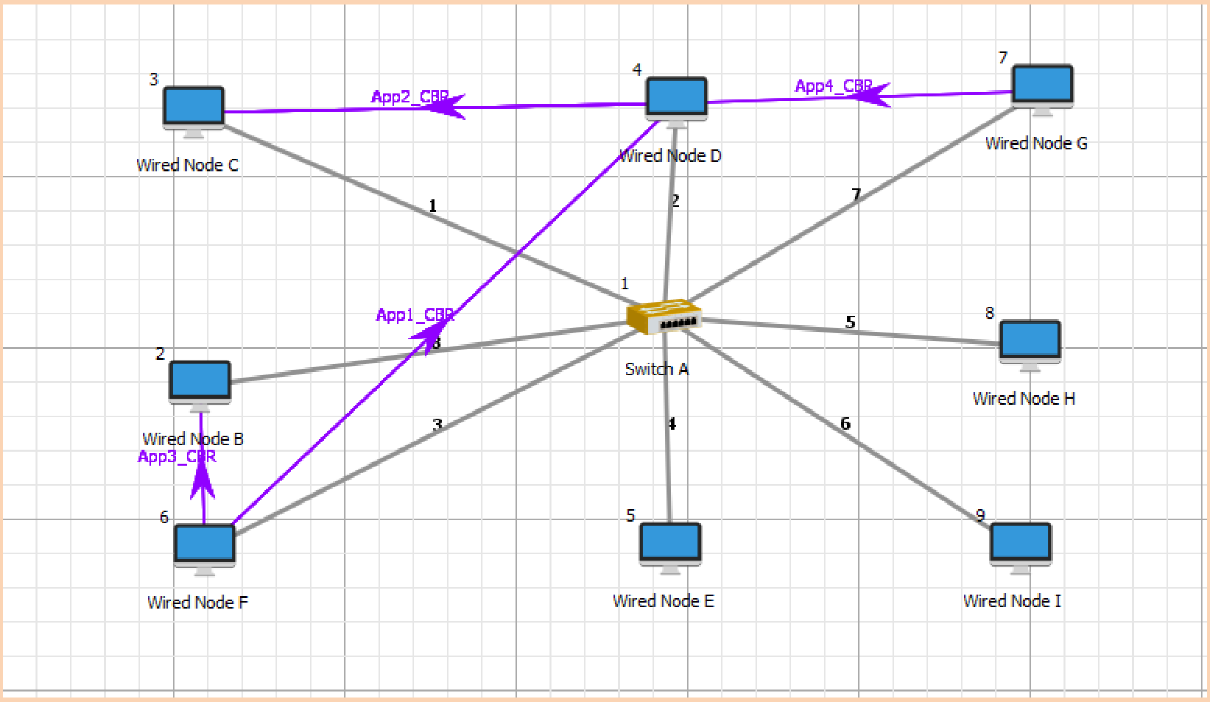
Network_Metrics_Table

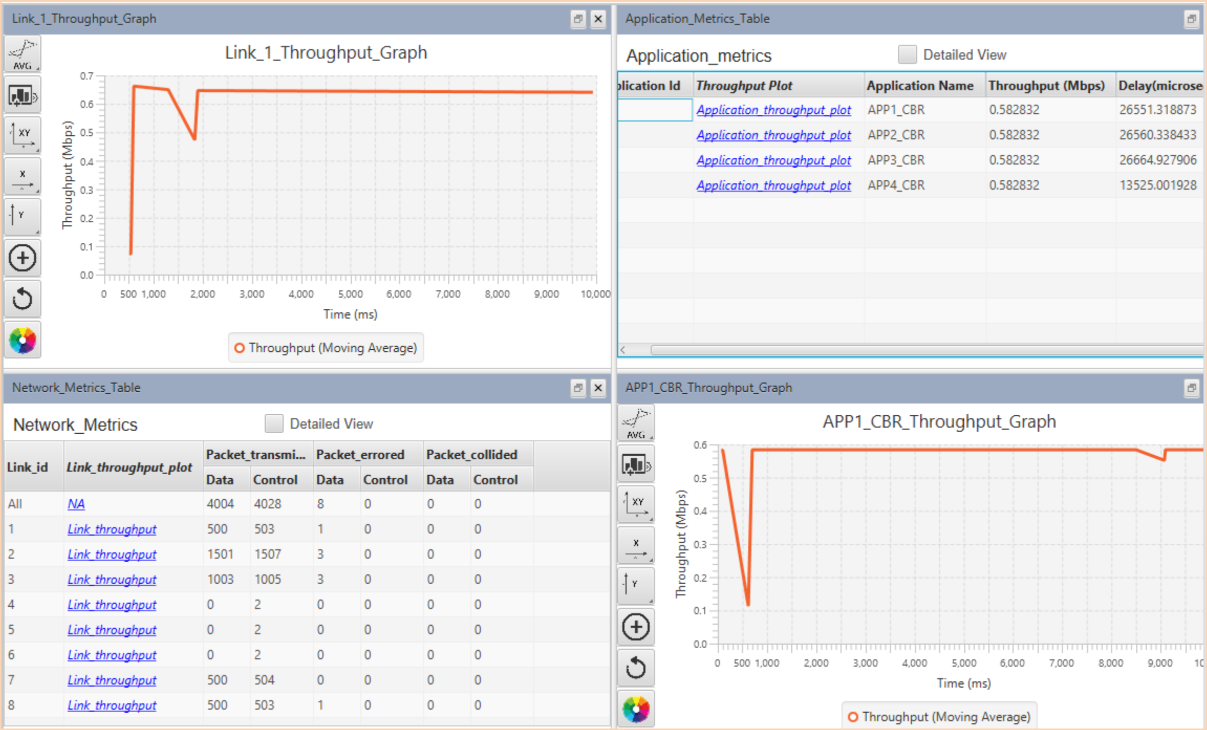
Network_Metrics Detailed View

Link_id	Link_throughput_plot	Packet_transmi...		Packet_errored		Packet_collided	
		Data	Control	Data	Control	Data	Control
All	NA	2001	2010	3	0	0	0
1	Link_throughput	500	502	1	0	0	0
2	Link_throughput	1001	1004	2	0	0	0
3	Link_throughput	500	503	0	0	0	0
4	Link_throughput	0	1	0	0	0	0

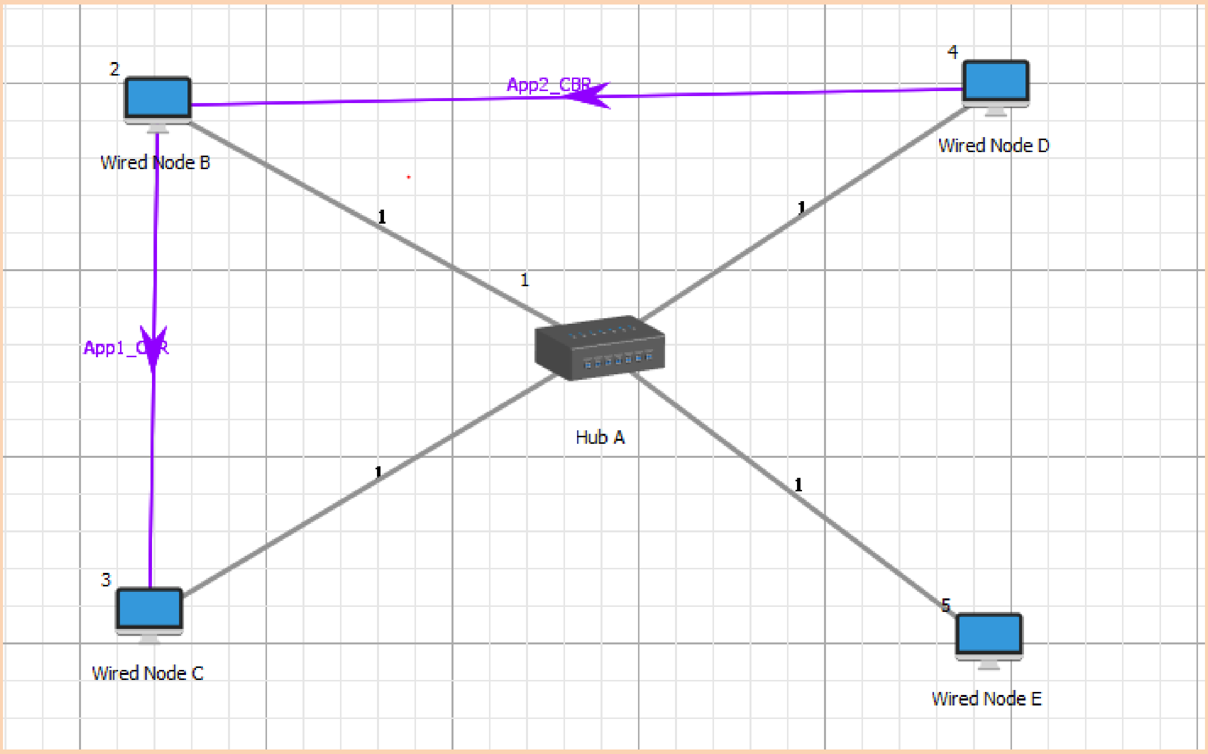


Circuit Diagram with SWITCH – 8 Nodes





Circuit Diagram with HUB – 4 Nodes



Application_metrics Detailed View

Application Id	Application Name	Throughput (Mbps)	Delay(microsec)
1	APP1_CBR	0.582832	3401.385170
2	APP2_CBR	0.578160	3370.214141

Network_Metrics_Table

Network_Metrics Detailed View

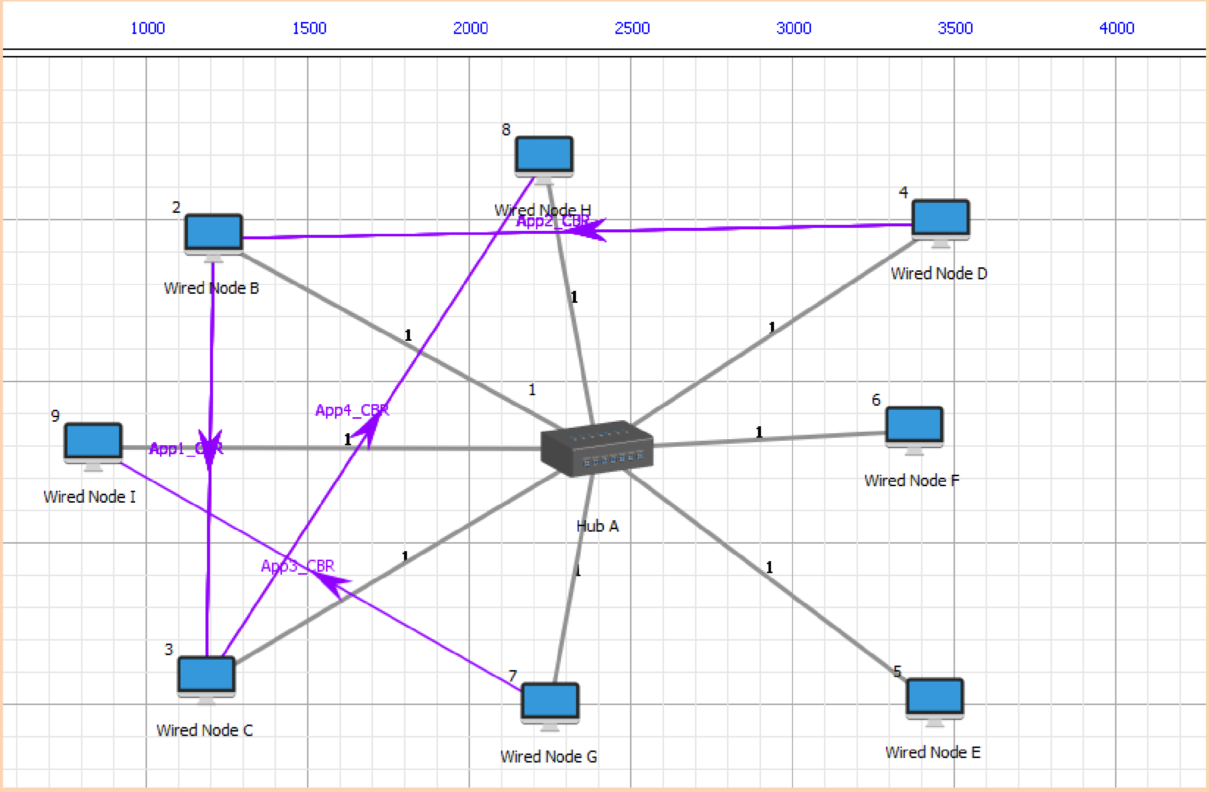
Link_id	Link_throughput_plot	Packet_transmi...		Packet_errored		Packet_collided	
		Data	Control	Data	Control	Data	Control
All	NA	2070	0	4	0	1072	0
1	NA	2070	0	4	0	1072	0

IP_Metrics_Table

IP_Metrics Detailed View

Device Id	Packet sent	Packet forwarded	Packet received
2	499	0	495
3	0	0	499
4	499	0	0
5	0	0	0

Circuit Diagram with HUB – 8 Nodes



Queue_Metrics_Table

Queue_Metrics ☐ Detailed View

Device_id	Port_id	Queued_pa...	Dequeued_...	Dropped_p...

No content in table

Application_Metrics_Table

Application_metrics ☐ Detailed View

Application Id	Application Name	Throughput (Mbps)	Delay(microsec)
1	APP1_CBR	0.374928	2036866.746418
2	APP2_CBR	0.456688	1147022.215857
3	APP3_CBR	0.414640	1474829.056902
4	APP4_CBR	0.369088	1795649.397469

Network_Metrics_Table

Network_Metrics ☐ Detailed View

Link_id	Link_throughput_plot	Packet_transmit...		Packet_errored		Packet_collided	
		Data	Control	Data	Control	Data	Control
All	NA	16825	0	6	0	15436	0
1	NA	16825	0	6	0	15436	0

IP_Metrics_Table

IP_Metrics ☐ Detailed View

Device Id	Packet sent	Packet forwarded	Packet received
2	499	0	391
3	499	0	321
4	499	0	0
5	0	0	0
6	0	0	0
7	499	0	0
8	0	0	316
9	0	0	355

SPANNING TREE - 1

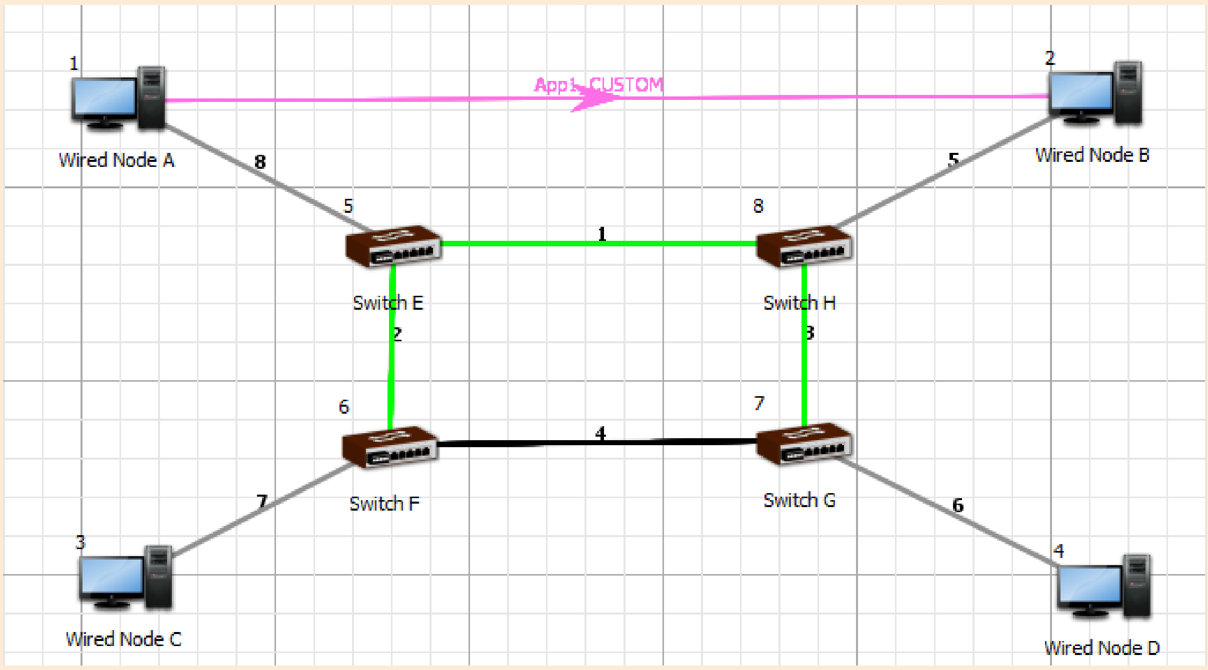
Aim

To simulate Networks using switch priorities in Netsim and compare their performance.

Tools Required

Netsim Software

Network Diagram – 1111



Switch E:

1. D560 – H
2. 1241 – F - Root Port
3. 1F41 – Node A

Switch F:

1. 62C2 – E
2. D67C – G
3. 3611 – Node C - Root Port

Switch G:

1. B4A1 – H
2. 1CB6 – F - Root port
3. FC01 – Node D

Switch H: **- Root switch**

1. 0140 – E - Root port
2. 28D9 – G
3. 8E99 – Node B

Inference:

Smallest MAC address of each switch shows their root port

And the smallest MAC address among the root ports, show the root switch

SPANNING TREE - 2

Aim

To simulate Networks using switch priorities in Netsim and compare their performance.

Tools Required

Netsim Software

Inference:

Smallest MAC address of each switch shows their root port

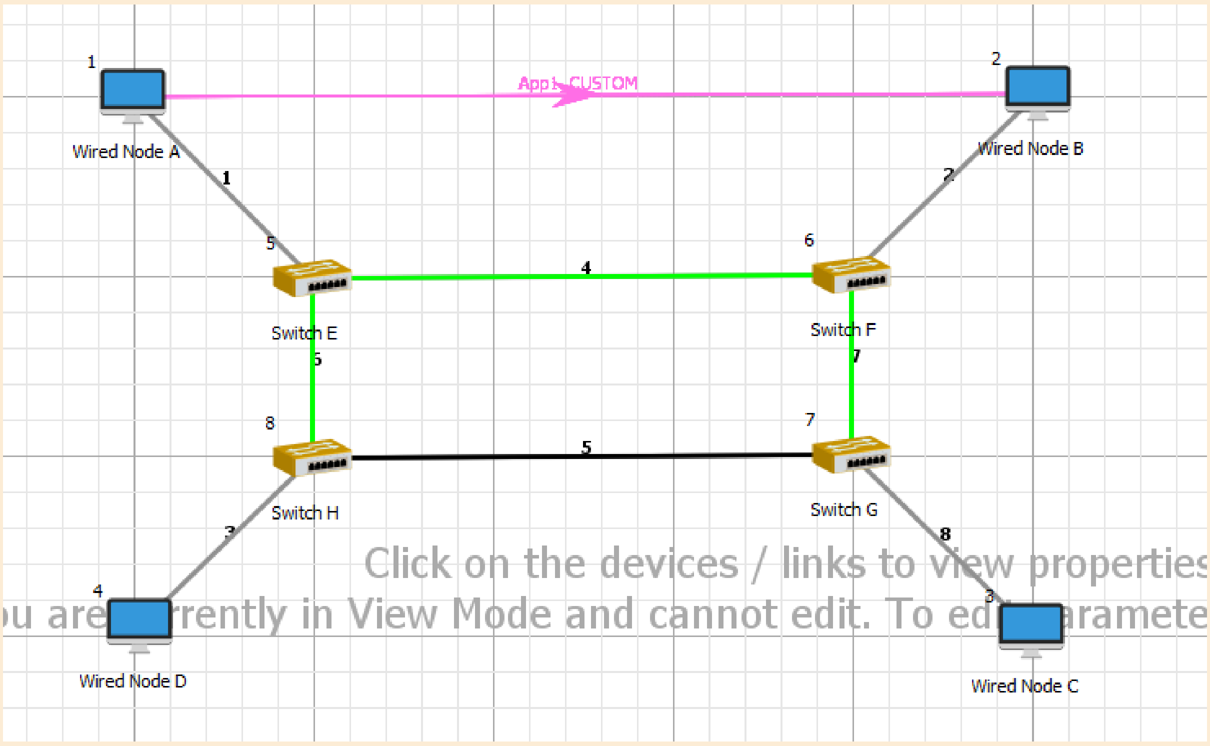
And the smallest MAC address among the root ports, show the root switch

The message packet is transferred usually through the ports which has the lowest MAC address for switches of **equal priority**, but since the priority of the switches have been changed manually – new routes of packet transmission is made automatically.

Hence, the following experiment has been observed with multiple sequences of priority arrangement among the switches. This changes the root switch, the root ports and the blocked ports.

So, "1234" means that switch 'E' has been given **1st** priority, 'F' got the **2nd** priority, 'G' got the **3rd** priority and 'H' got the **4th** priority. 4 different sequences have been listed below.

Network Diagram – 1234



Root switch: E

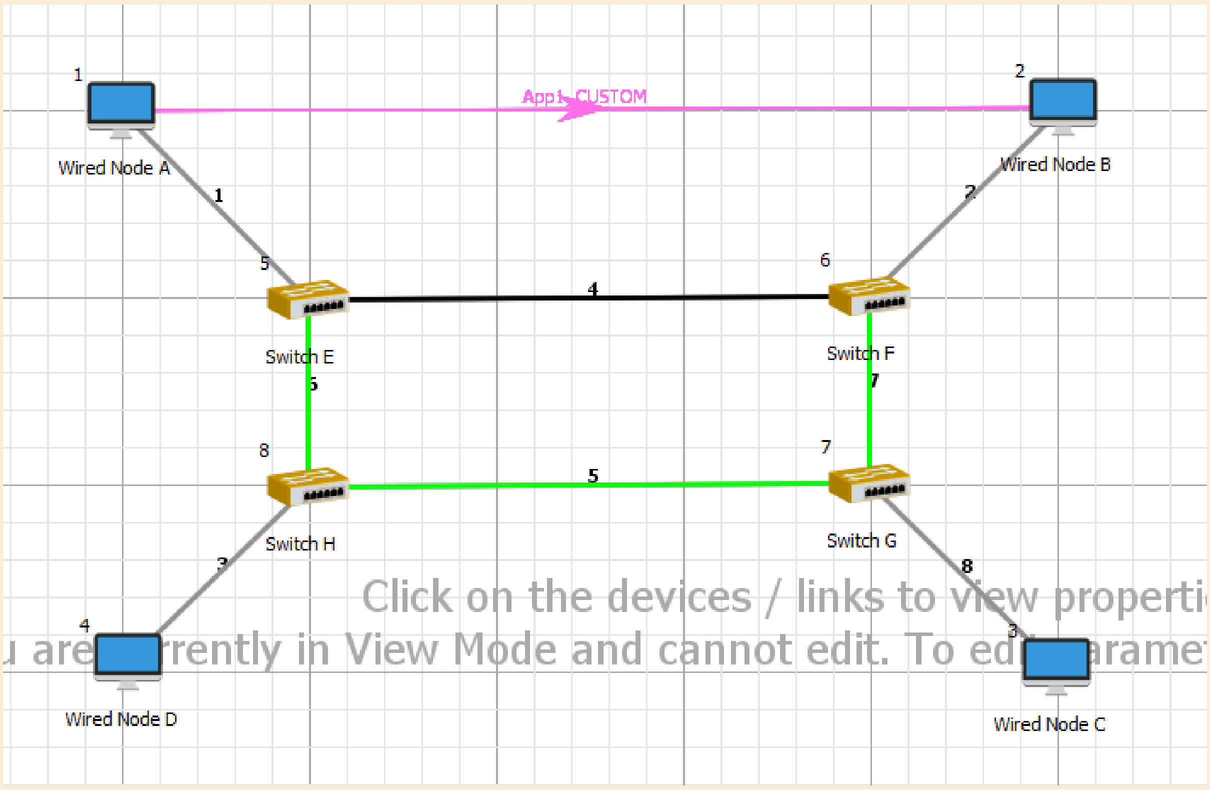
Root ports: G(7), F(4), H(6)

Designated ports: E(4), E(6), F(7)

Blocked ports: H(5), G(5)

Forward paths: 4, 6, 7

Network Diagram – 4312



Root switch: G

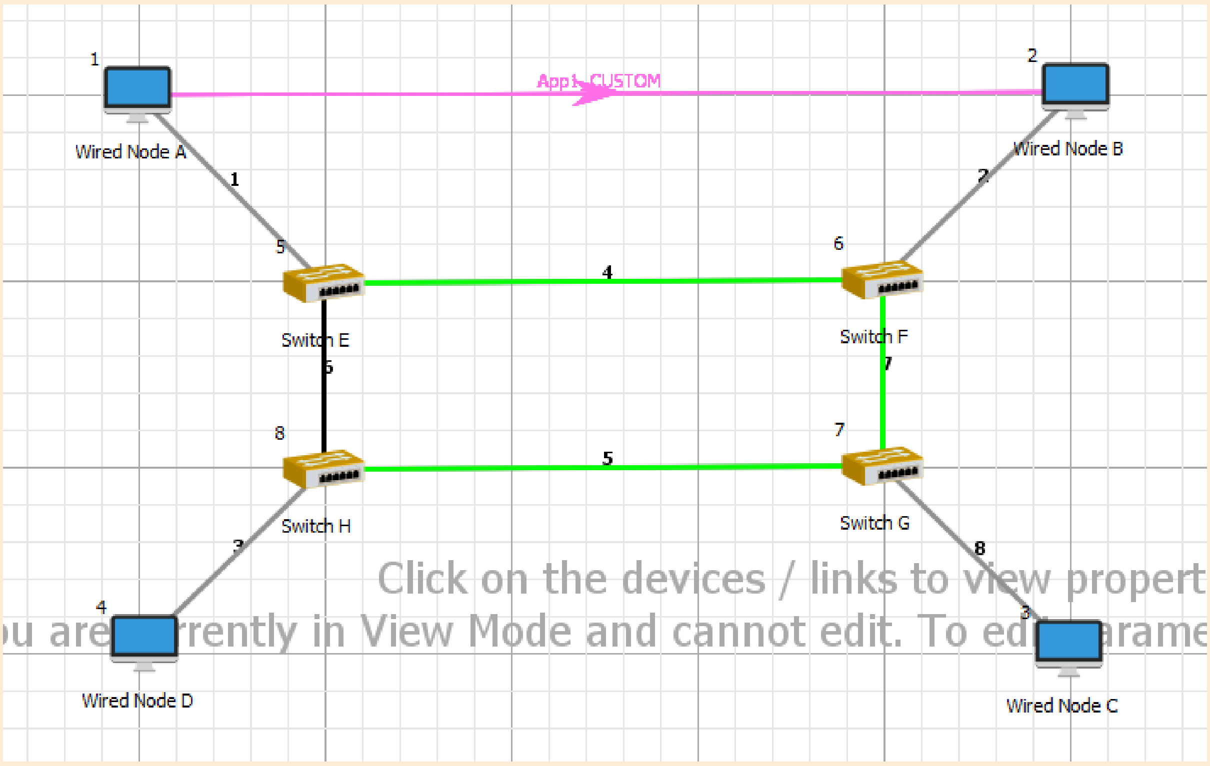
Root ports: F(7), H(5), E(6)

Designated ports: G(7), G(5), H(6)

Blocked ports: E(4), F(4)

Forward paths: 5, 6, 7

Network Diagram – 3124



Root switch: F

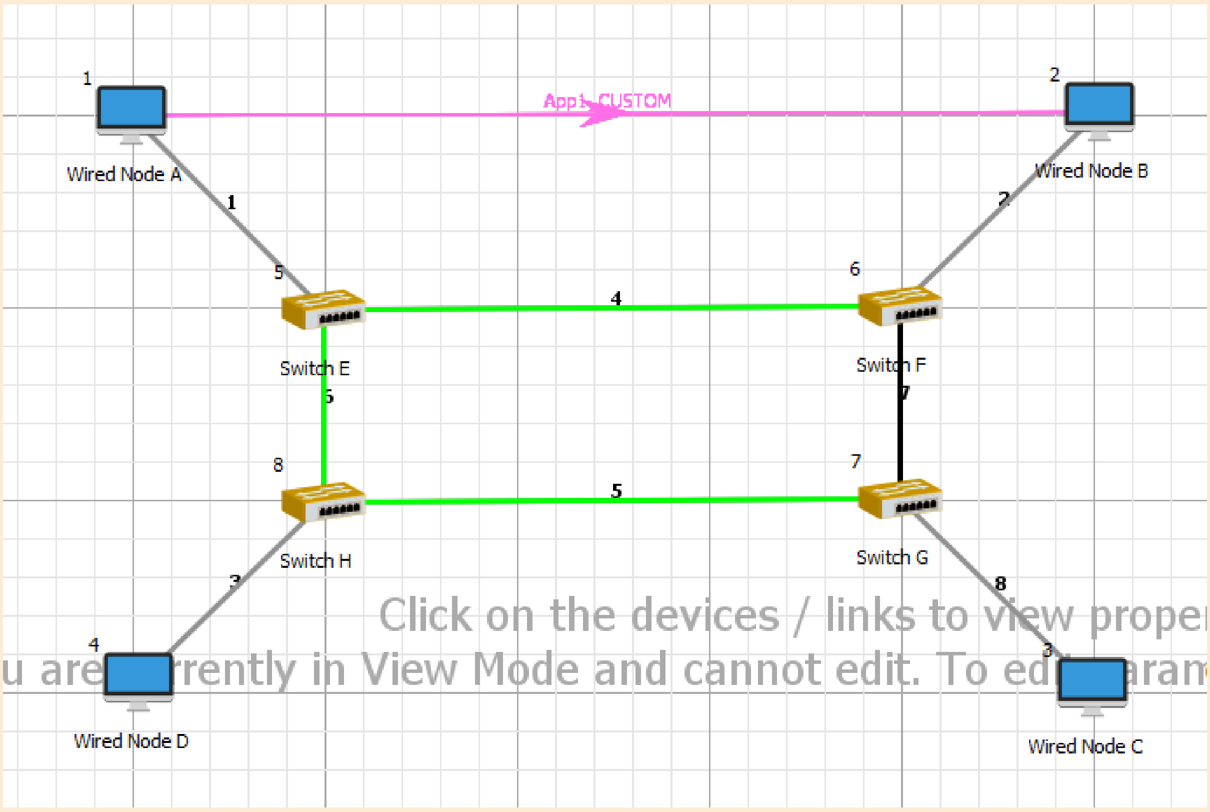
Root ports: E(4), H(5), G(7)

Designated ports: F(7), F(4), G(5)

Blocked ports: H(5), G(5)

Forward paths: 4, 5, 7

Network Diagram – 2431



Root switch: H

Root ports: E(6), G(5), F(4)

Designated ports: H(5), H(6), E(4)

Blocked ports: H(5), G(5)

Forward paths: 4, 5, 6

SIMPLE AND BUSY NETWORKS

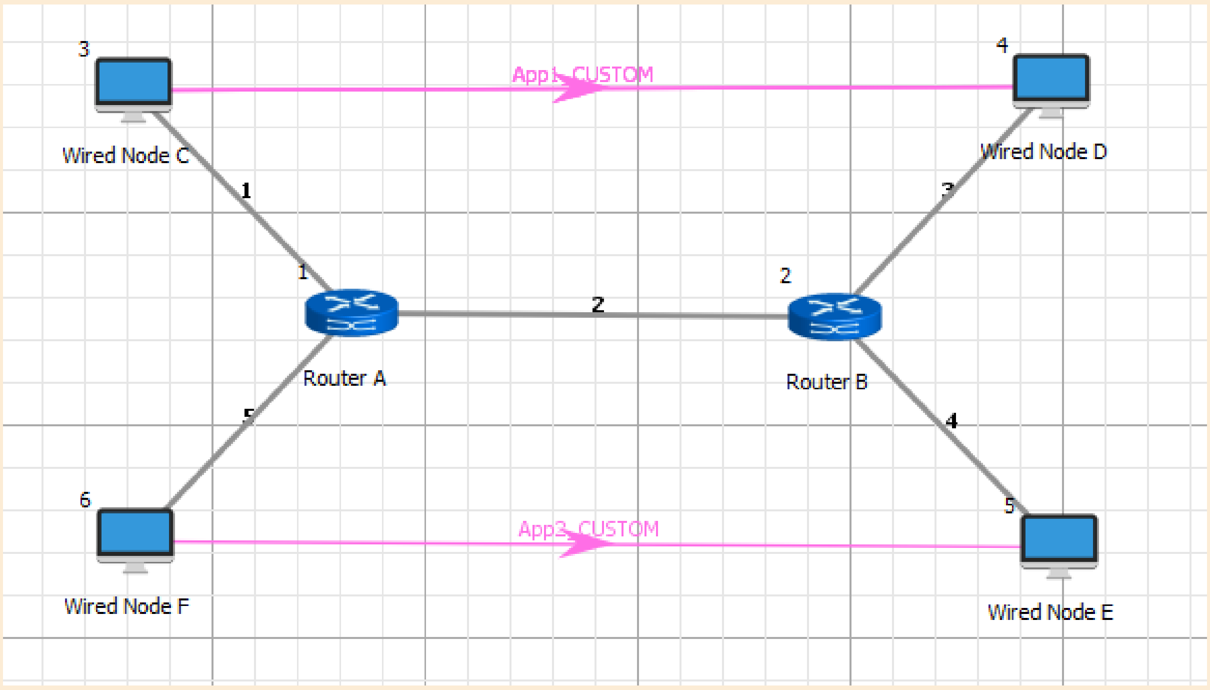
Aim

To study performance comparison between busy and simple networks

Tools Required

Netsim Software

Network Diagram



1:

Fast network: 100 Mbps

Throughput: 0.582832

Delay: 26737

2:

Simple network: 10 Mbps

Throughput: 0.582832

Delay: 32816

3:

Busy network: 1 Mbps

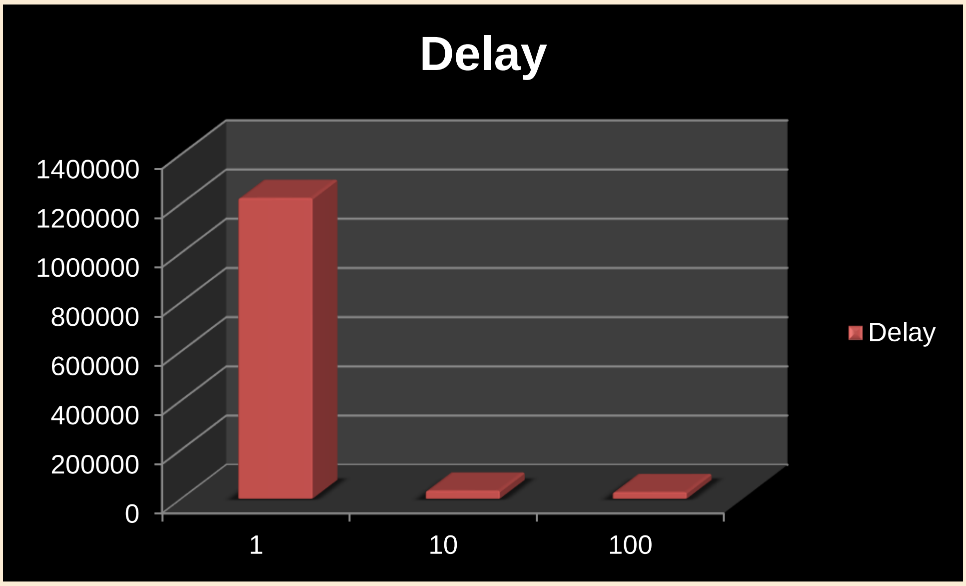
Throughput: 0.462528

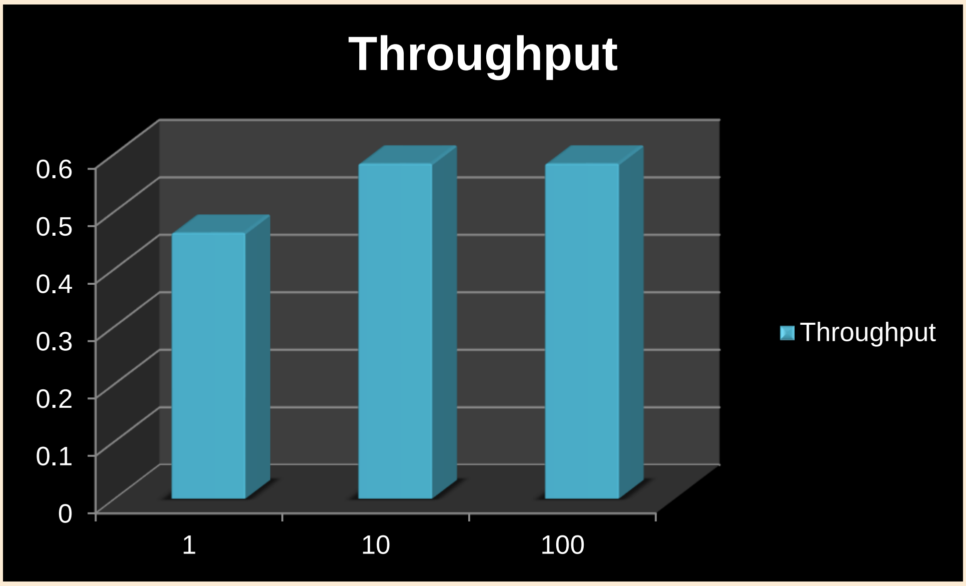
Delay: 1222234

Inference

From the given graphs, we understand the fact that the delay is very high for a busy network with a lower uplink and downlink speed. As an additional measure, I have also calculated the same for 100 Mbps and found nearly similar results to 10 Mbps, implying that the increase in speed results in a logarithmic decrease in the delay time.

Graphs


Delay
1400000
1200000
1000000
800000
600000
400000
200000
0
1
10
100
Delay


Throughput
0.6
0.5
0.4
0.3
0.2
0.1
0
1
10
100
Throughput

UDP & TCP

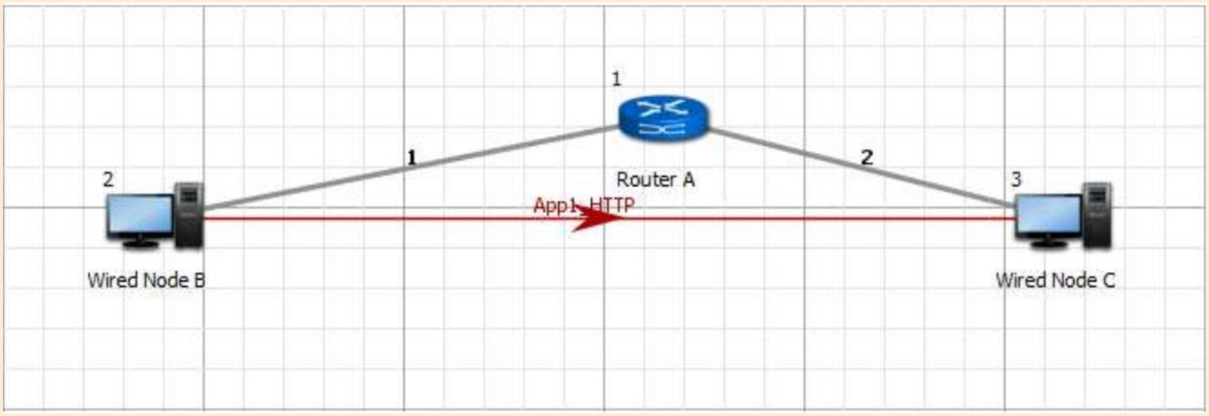
Aim

A study of performance between TCP and UDP busy and simple networks

Tools Required

Netsim Software

Network Diagram



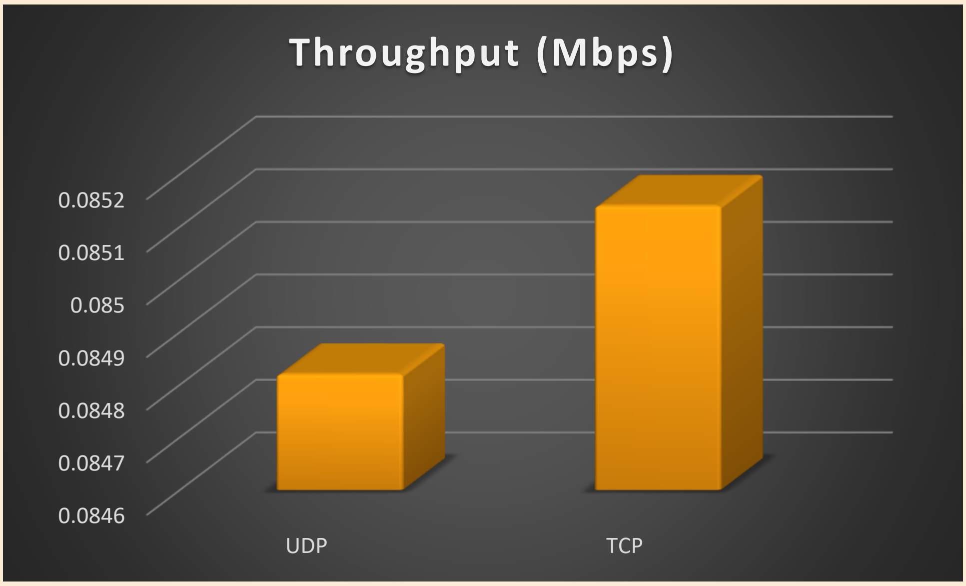
UDP:

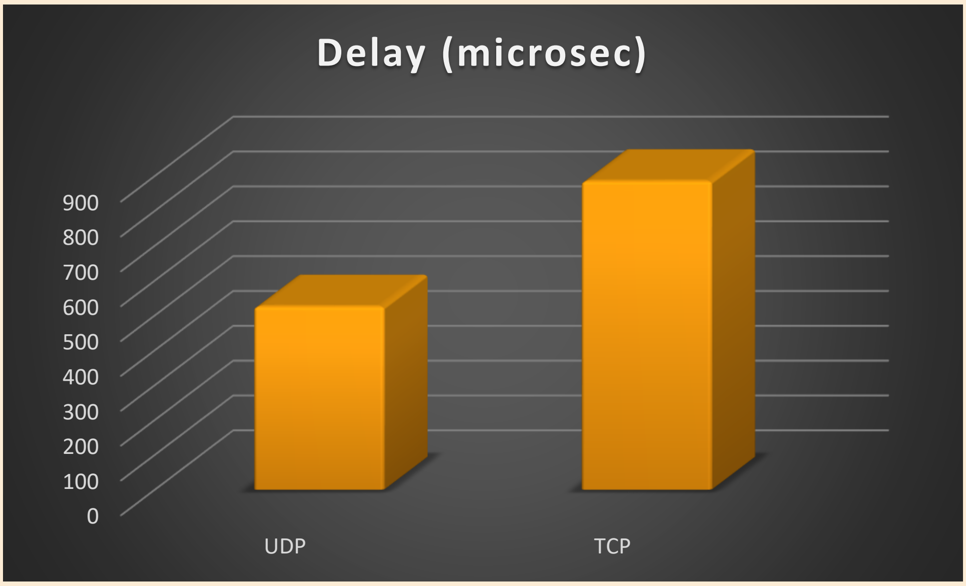
Packet Transmitted	Packet Received	Payload Transmitted	Payload Receiverd	Throughput (Mbps)	Delay (microsec)
1089	1085	1064250	1060250	0.08482	525.575742

TCP:

Packet Transmitted	Packet Received	Payload Transmitted	Payload Receiverd	Throughput (Mbps)	Delay (microsec)
1089	1085	1064250	1064250	0.08514	885.27056

Graphs





Inference:

TCP is a connection-oriented protocol while UDP is a connectionless protocol. So, the delay should be more in TCP as, it takes some extra time to establish connection. This can be inferred from the Delay graph above. On the other hand, even the throughput of TCP is greater, showing that the extra delay makes it more reliable. But the difference is very minute when comparing with UDP

IP ADDRESSING

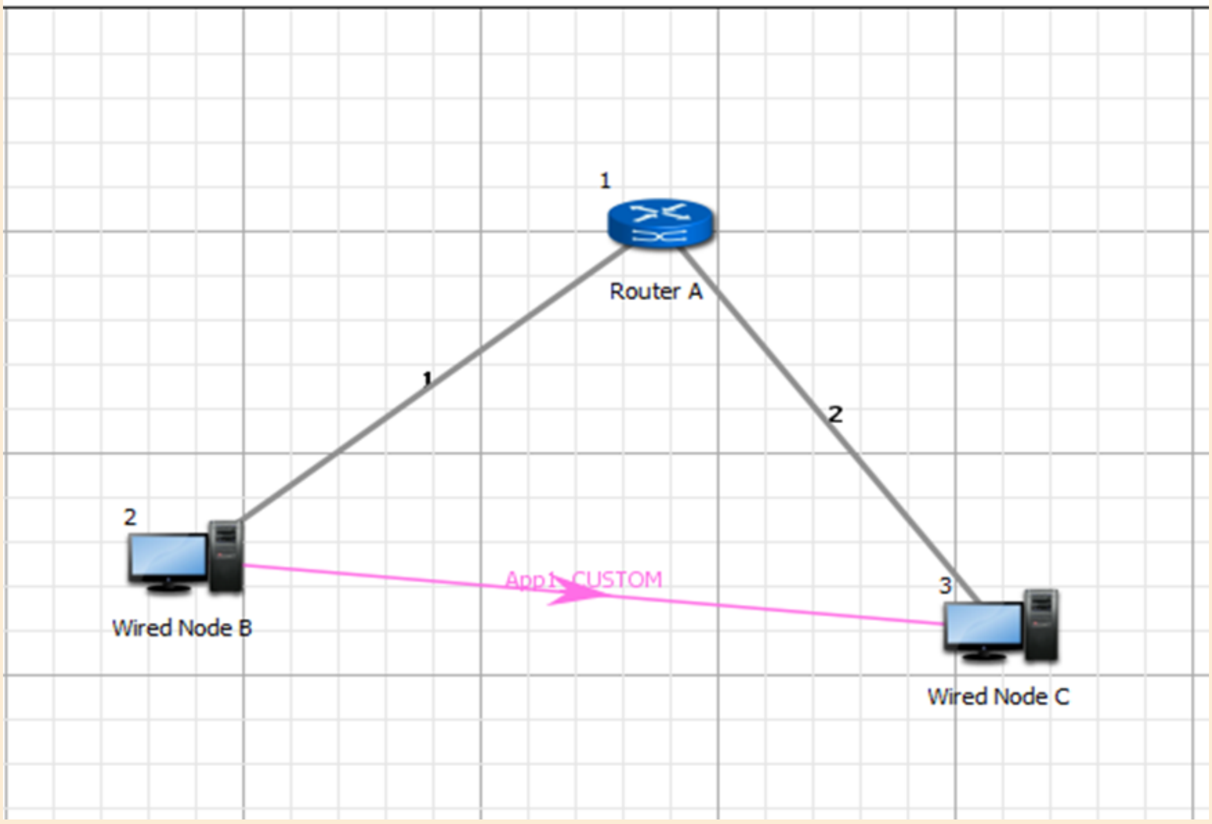
Aim

To understand IPv4Addressing with classes A and B

Tools Required

Netsim Software

Network Diagram



Observations

CLASS A WITH SUBNETTING

Router A:

IP Address: 11.1.1.1 (Interface with B), 11.2.1.1 (Interface with C)

Subnet Mask: 255.255.0.0

Wired Node B:

IP Address: 11.1.1.2

Subnet Mask: 255.255.0.0

Default Gateway: 11.1.1.1

Wired Node C:

IP Address: 11.2.1.2

Subnet Mask: 255.255.0.0

Default Gateway: 11.2.1.1

DeviceId	IP_Address	MAC_Address	Type
1	11.1.1.1	43522E199F12	STATIC
	11.2.1.1	890F23F5A261	STATIC
	192.2.1.2	24203B32BB36	STATIC
	11.2.1.2	3C09E34C40A6	STATIC
2	11.1.1.1	43522E199F12	STATIC
	11.2.1.1	890F23F5A261	STATIC
	192.2.1.2	24203B32BB36	STATIC
	11.2.1.2	3C09E34C40A6	STATIC
3	11.1.1.1	43522E199F12	STATIC
	11.2.1.1	890F23F5A261	STATIC
	192.2.1.2	24203B32BB36	STATIC
	11.2.1.2	3C09E34C40A6	STATIC

CLASS B WITH SUBNETTING

Router A:

IP Address: 190.1.1.1 (Interface with B),

190.2.1.1 (Interface with C)

Subnet Mask: 255.255.255.0

Wired Node B:

IP Address: 190.1.1.2

Subnet Mask: 255.255.255.0

Default Gateway: 190.1.1.1

Wired Node C:

IP Address: 190.2.1.2

Subnet Mask: 255.255.255.0

Default Gateway: 190.2.1.1

DeviceId	IP_Address	MAC_Address	Type
1	190.1.1.1	F683FEC92A83	STATIC
	190.2.1.1	49E615E9C226	STATIC
	190.1.1.2	D61B67CDD181	STATIC
	190.2.1.2	6A11FFA324BC	STATIC
2	190.1.1.1	F683FEC92A83	STATIC
	190.2.1.1	49E615E9C226	STATIC
	190.1.1.2	D61B67CDD181	STATIC
	190.2.1.2	6A11FFA324BC	STATIC
3	190.1.1.1	F683FEC92A83	STATIC
	190.2.1.1	49E615E9C226	STATIC
	190.1.1.2	D61B67CDD181	STATIC
	190.2.1.2	6A11FFA324BC	STATIC

Results:

Case 1: Throughput and Delay are more than zero

Case 2: Throughput and Delay are zero

Case 3: Throughput and Delay are more than zero

Inference:

Networks in which the IP address of the nodes are of the same class, have an established connection as seen by the positive values of throughput and delay. When the class of one of the nodes is changed, by changing its IP address, there is no established connection or the server crashes, as seen by the zero value of the throughput or delay