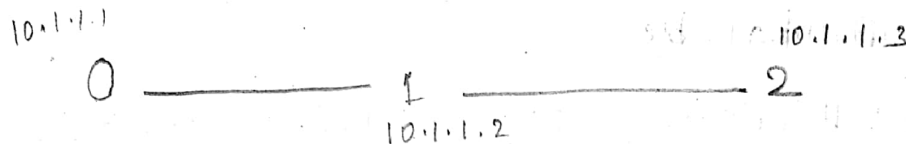




Name :	Samyuktha H R	Branch:	
USN/Roll No. :	1MST6CS089	Sem/Sec:	
Subject :	CN LAB	Subject Code:	

Stimulate a three node point-to-point network with duplex links between them. set queue, vary size the bandwidth and find number of packets dropped.

Topology



→ Copy the file from example, traffic control

Before int main() and after NS_LOG - UNCOND add

void

Tcpackets Queue

and

void

Device Packet Queue Trace

Remove void Time Trace()

Change the port from TCP → UDP

Remove and add value

change no of nodes from 2 → 3

Create 2 devices instead of 1, ~~copy~~

devices1 \rightarrow nodes.Get(0) \rightarrow 1

devices2 \rightarrow nodes.Get(1) \rightarrow 2

Remove traffic controller helper before IPv4 Address helper

Set 2 addresses "10.1.1.0", "10.1.2.0" and 2 interfaces.

Add GlobalRoutingTable

IPv4 GlobalRoutingHelper :: populateRoutingTables();

In Application Container, change nodes.Get(0) to nodes.Get(1)

In address, value remote address change the interface GetAddress(0) 0 \rightarrow 1

\rightarrow In nodes.Get(1) change 1 \rightarrow 0.

- Remove from uint32_t packets dropped by queue.

After Destroy remove everything till return.

Output

Tx packets	30482	For 50 Mbps	payload
	69759	100	1428
	109038	150	1448
	127649	200	2500

with flow monitor code

Tx Packets : 30801

Flow-id : t

Src Address : 10.1.1.1

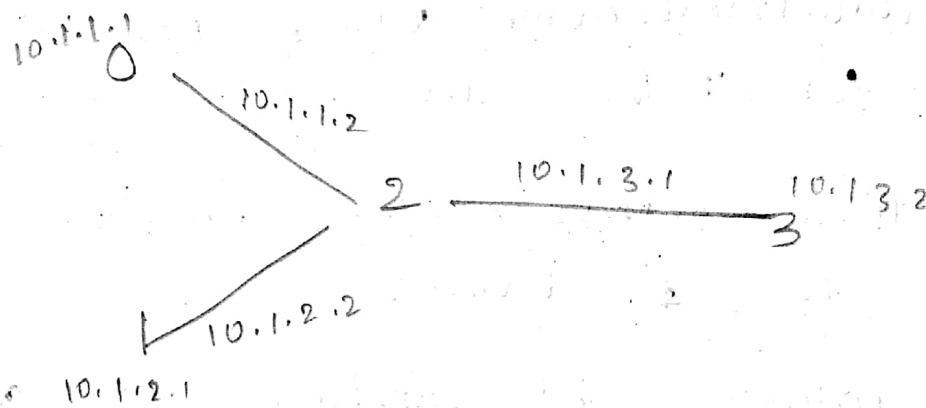
Destination : 10.1.2.2

No. of packets lost : 3928

Name :	Jamyuktha H R	Branch:	
USN/Roll No. :	IMS16CS089	Sem/Sec:	
Subject :	CN Lab	Subject Code:	

Stimulate a four node point-to-point network and connect links as follows $n0-n2$, $n1-n2$, $n2-n3$. Apply TCP agent between $n0-n3$ and UDP agent between $n1-n3$. Apply relevant applications over TCP and UDP agents by changing the parameters.

Topology



Remove initial 2 void functions

change the create node(2) → create node(4)

Remove cmd add value

Change the 3rd point to point set queue max value to mode and string value of to "QUEUE-MODE-PACKETS"

Create 3 devices for NetDevice connector, copy paste the same.

NetDevice connect Device1 = point to point (nodes Get(0) → (2))
Similarly 2 & 3 (1 → 2) & (2 → 3) respectively.

After declaring Tcb, in host add vint16-t handle
= tcb.SetRouteQueueDrivers ("ns3: RedQueue")

Add internal Queue used by ctd as tcb.AddInternalQueue(handle, 1, "ns3: DropTailQueue",
"max-packets", UintegerValue(10000));

Copy paste QueueDiskContainer devices = tcb.install
(devices). for all the 3 devices.

Copy paste ipv4 address for creating 3
such address for 3 devices.

create the interface and similarly 2nd and
3rd device.

Update starts (seconds(1.0))

Similarly make tcp connections

set the port to 9.

Print per flow statistics.

monitor → checkForLostPackets();

ptr → classifier

IPv4 flow classifier → five tuple t

NS-LOG-UNCOND("Flow ID", src, dest address) using
iterator.

NS-LOG-UNCOND("Tx Packets");

Output:

Flow ID: 1 / Src Addr 10.1.1.1 Destin-Address 10.1.3.2

No. of packets transmitted = 39278

Flow ID: 2 / Src Addr 10.1.2.1 Des Address 10.1.3.2

No of packets transmitted = 3

Flow ID: 3 / Src Addr 10.1.3.2 Des Address 10.1.3.1

No of packets transmitted = 3



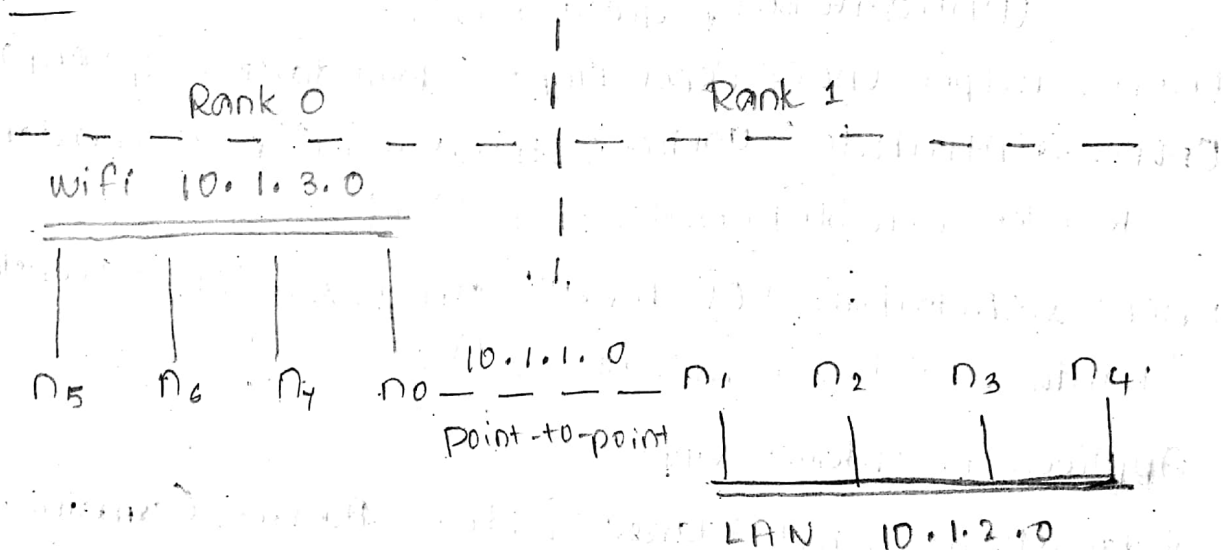
MARKS :

NS 3 Program 3

Name :	Samyuktha H R	Branch:	
USN/Roll No. :	IMS16CS089	Sem/Sec:	
Subject :	CN Lab	Subject Code:	

Stimulate simple extended service set with transmitting nodes in wireless LAN and determine the performance with respect to transmission of packets.

Network Topology :



Modifications :

① In tutorial / third.cc

add UDP flow as below

// UDP flow

uint16_t port = 7

Address localAddress(Inet socketAddress (Ipv4Address::GetAny(), port));

```

PacketSinkHelper packetSinkHelper(SocketType, localAddress);
ApplicationContainer sinkApp = PacketSinkHelper::Install(
    CsmNodes.Get(nCsm));
SinkApp.start(Seconds(0.0));
SinkApp.stop(Seconds(simulationTime + 0.1));
uint32_t payloadSize = 33448;
Config::setDefault("ns3::TcpSocket::SegmentSize",
    UIntegerValue(payloadSize));
OnoffHelper onoff(SocketType, IPv4Address::GetAny());
onoff.SetAttribute("OnTime", StringValue("ns3::ConstantRandomVariable[Constant=1]"));
onoff.SetAttribute("OffTime", StringValue("ns3::ConstantRandomVariable[Constant=0]"));
ApplicationContainer apps;
AddressValue remoteAddress(InetSocketAddress(CsmInterface::GetAddress(nCsm), port));
onoff.SetAttribute("Remote", remoteAddress);
apps.Add(onoff, Install(WifiStationNodes, Get(nWifi-1)));
apps.start(Seconds(1.0));
apps.stop(Seconds(simulationTime + 0.1));
// Print per flow statistics.

```


Add throughput

$$\text{Throughput} = \frac{\text{Citer} \rightarrow \text{second} \cdot \text{RxPacket} \times 8}{\text{Citer} \rightarrow \text{second} \cdot \text{RxPacket} - \text{iter} \rightarrow \text{second} \cdot \text{TxPacket}} \cdot 1024 ;$$

Output :

Flow ID : 1

Src Address : 10.1.3.3

Destination Address : 10.1.2.4

Tx Packets : 33846



NS 3 Prog 4

Name :	Samyuktha H R	Branch:	
USN/Roll No. :	IMS16C8089	Sem/Sec:	
Subject :	CN Lab	Subject Code:	

Stimulate a wireless network, generate traffic and analyze its performance.

- In the main function, remove both the void function which are before int main()
- In int main remove packet size & num packets and verbose value.
- Remove the cmd.add value.
- Remove the convert to time object
- create 5 nodes instead of 2.
- Do not include the wifi.EnableComponents().
- MobilityHelper for list positional allocation.
Mobility. Set Positional Allocator ("ns3::GridPositionalAllocator", "Minx", DoubleValue(0.0),
"Miny", DoubleValue(0.6),
"Deltax", DoubleValue(5.0);
"LayoutType", StringValue("RowFirst");

```

→ Add Ipv4 GlobalRoutingHelper::PopulateRoutingTables();
→ packetSinkHelper = ns3::UdpSocketFactory
→ CInetSocketAddress(i.GetAddress(3), port))) ;
  Apps.AddConoff.Install(C.getCo());
→ // Add visualization using Netanim
  AnimationInterface anim("ex4.xml");
→ // Print per Flow stats
  Add the throughput
  NS_LOG_UNCOND("Throughput" << iter->second.TxBytes * 8 /
    (iter->second.timeLastRxPacket.GetSeconds() -
    iter->second.timeFirstTxPacket.GetSeconds()) / 1024);
  Simulator Destroy();

```

Output:

Flow ID : 1

src Add : 10.1.2.4

Tx Packet : 33846

Rx Packet : 41328

Throughput : 678.4 -



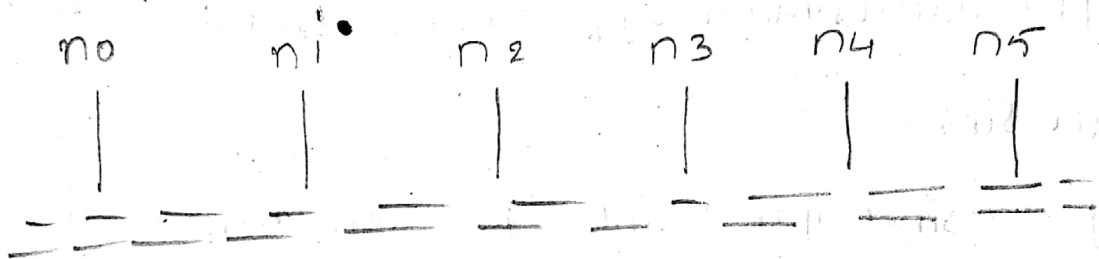
N83

Prog. 5

Name :	Samyuktha H. R	Branch:	
USN/Roll No. :	IMS16CS089	Sem/Sec:	
Subject :	CN Lab	Subject Code:	

Stimulate a transmission of ping message over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion?

Topology



node n0, n1, n2, n3, n5 pings to node n4

node n0 generates protocol 2 (ICMP) to node n5.

Delete the static void sink Rx and ping Rtt functions

In the int main

include a variable Time InterPacketInterval = Seconds(1.0);

Instead of LOG_INFO → NS_LOG_UNCOND

Create 6 nodes

Connect all nodes to shared channel using
CsmatHelper csma and csma install (c);

Add ip stack to all nodes

NS-LOG-UNCOND("Add ip stack");

ipstack.Install(c);

Assign IP address

ip.Assign(device);

Create Source

InetSocketAddress dst = InetSocketAddress(Add.GetAddress());

ApplicationContainer app = Onoff.Install(c.getCo());

Create Sink

PacketSinkHelper Sink = PacketSinkHelper("ns3::UdpSocketFactory", dst);

Create Pingers

ping.SetAttribute("Interval", TimeValue(InterPacketInterval));

Flow monitor

Calculate the throughput

throughput = (iter -> second. rxbytes * 8) / (iter -> second, timeLastRxPackets.GetSeconds() - timeFirstTxPacket) 1024;

Output:

Create Node

Connect Node

Add ip stack

Create source

Create Sink

Flow ID: 1 / Src Addr 10.1.3.3

Destⁿ addr : 10.1.2.6

TxPackets : 33846

RxPackets : 54380

Throughput : 831.21