Assignment 3

Kola Deepti(120CS0151)

CSMA-brodcast

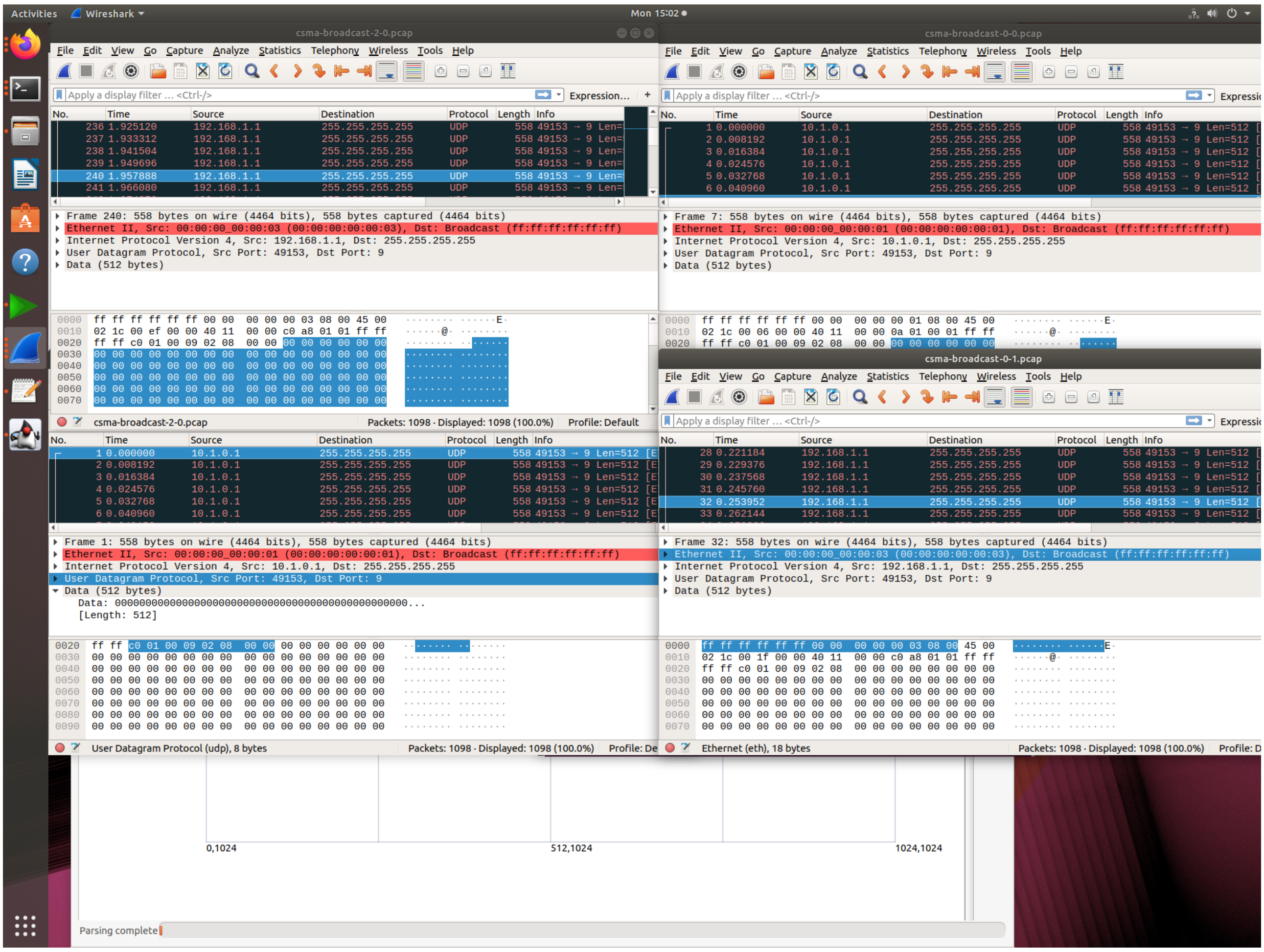
Code:

```
 #include <iostream>
   #include <fstream>
   #include <string>
   #include <cassert>

   #include "ns3/core-module.h"
   #include "ns3/network-module.h"
   #include "ns3/csma-module.h"
   #include "ns3/applications-module.h"
   #include "ns3/internet-module.h"
   #include "ns3/netanim-module.h"
   using namespace ns3;

   NS_LOG_COMPONENT_DEFINE ("CsmaBroadcastExample");

   int
   main (int argc, char *argv[])
   {
     // Users may find it convenient to turn on explicit debugging
     // for selected modules; the below lines suggest how to do this
   #if 0
     LogComponentEnable ("CsmaBroadcastExample", LOG_LEVEL_INFO);
   #endif
     LogComponentEnable ("CsmaBroadcastExample", LOG_PREFIX_TIME);
```

```
// Allow the user to override any of the defaults and the above
// Bind()s at run-time, via command-line arguments
CommandLine cmd;
cmd.Parse (argc, argv);

NS_LOG_INFO ("Create nodes.");
NodeContainer c;
c.Create (3);
NodeContainer c0 = NodeContainer (c.Get (0), c.Get (1));
NodeContainer c1 = NodeContainer (c.Get (0), c.Get (2));

NS_LOG_INFO ("Build Topology.");
CsmaHelper csma;
csma.SetChannelAttribute ("DataRate", DataRateValue (DataRate (5000000)));
csma.SetChannelAttribute ("Delay", TimeValue (MilliSeconds (2)));

NetDeviceContainer n0 = csma.Install (c0);
NetDeviceContainer n1 = csma.Install (c1);

InternetStackHelper internet;
internet.Install (c);

NS_LOG_INFO ("Assign IP Addresses.");
Ipv4AddressHelper ipv4;
ipv4.SetBase ("10.1.0.0", "255.255.255.0");
ipv4.Assign (n0);
ipv4.SetBase ("192.168.1.0", "255.255.255.0");
ipv4.Assign (n1);


// RFC 863 discard port ("9") indicates packet should be thrown away
// by the system.  We allow this silent discard to be overridden
// by the PacketSink application.
 uint16_t port = 9;

 // Create the OnOff application to send UDP datagrams of size
 // 512 bytes (default) at a rate of 500 Kb/s (default) from n0
 NS_LOG_INFO ("Create Applications.");
 OnOffHelper onoff ("ns3::UdpSocketFactory",
                    Address (InetSocketAddress (Ipv4Address ("255.255.255.255"), port)));
 onoff.SetConstantRate (DataRate ("500kb/s"));

 ApplicationContainer app = onoff.Install (c0.Get (0));
 // Start the application
 app.Start (Seconds (1.0));
app.Stop (Seconds (10.0));

// Create an optional packet sink to receive these packets
PacketSinkHelper sink ("ns3::UdpSocketFactory",
                       Address (InetSocketAddress (Ipv4Address::GetAny (), port)));
app = sink.Install (c0.Get (1));
app.Add (sink.Install (c1.Get (1)));
app.Start (Seconds (1.0));
app.Stop (Seconds (10.0));
AnimationInterface anim("csma-brodcast.xml");
// Configure ascii tracing of all enqueue, dequeue, and NetDevice receive
// events on all devices.  Trace output will be sent to the file
// "csma-one-subnet.tr"
AsciiTraceHelper ascii;
csma.EnableAsciiAll (ascii.CreateFileStream ("csma-broadcast.tr"));

// Also configure some tcpdump traces; each interface will be traced
// The output files will be named
// csma-broadcast-<nodeId>-<interfaceId>.pcap
```

```
  // and can be read by the "tcpdump -tt -r" command
  csma.EnablePcapAll ("csma-broadcast", false);


  NS_LOG_INFO ("Run Simulation.");
  Simulator::Run ();
  Simulator::Destroy ();
  NS_LOG_INFO ("Done.");
}
```

Wire Shark:



csma-broadcast-2-0.pcap

```
No.  Time       Source        Destination       Protocol  Length Info
 236 1.925120   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=
 237 1.933312   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=
 238 1.941504   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=
 239 1.949696   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=
 240 1.957888   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=
 241 1.966080   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=

Frame 240: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits)
Ethernet II, Src: 00:00:00_00:00:03 (00:00:00:00:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 49153, Dst Port: 9
Data (512 bytes)

0000  ff ff ff ff ff ff 00 00  00 00 00 03 08 00 45 00   ········ ······E·
0010  02 1c 00 ef 00 00 40 11  00 00 c0 a8 01 01 ff ff   ······@· ········
0020  ff ff c0 01 00 09 02 08  00 00 00 00 00 00 00 00   ········ ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········

csma-broadcast-2-0.pcap        Packets: 1098 · Displayed: 1098 (100.0%)    Profile: Default

No.  Time       Source        Destination       Protocol  Length Info
   1 0.000000   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E
   2 0.008192   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E
   3 0.016384   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E
   4 0.024576   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E
   5 0.032768   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E
   6 0.040960   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [E

Frame 1: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits)
Ethernet II, Src: 00:00:00_00:00:01 (00:00:00:00:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 10.1.0.1, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 49153, Dst Port: 9
Data (512 bytes)
   Data: 000000000000000000000000000000000000000000000000...
   [Length: 512]

0020  ff ff c0 01 00 09 02 08  00 00 00 00 00 00 00 00   ········ ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0090  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········

User Datagram Protocol (udp), 8 bytes     Packets: 1098 · Displayed: 1098 (100.0%)    Profile: De
```

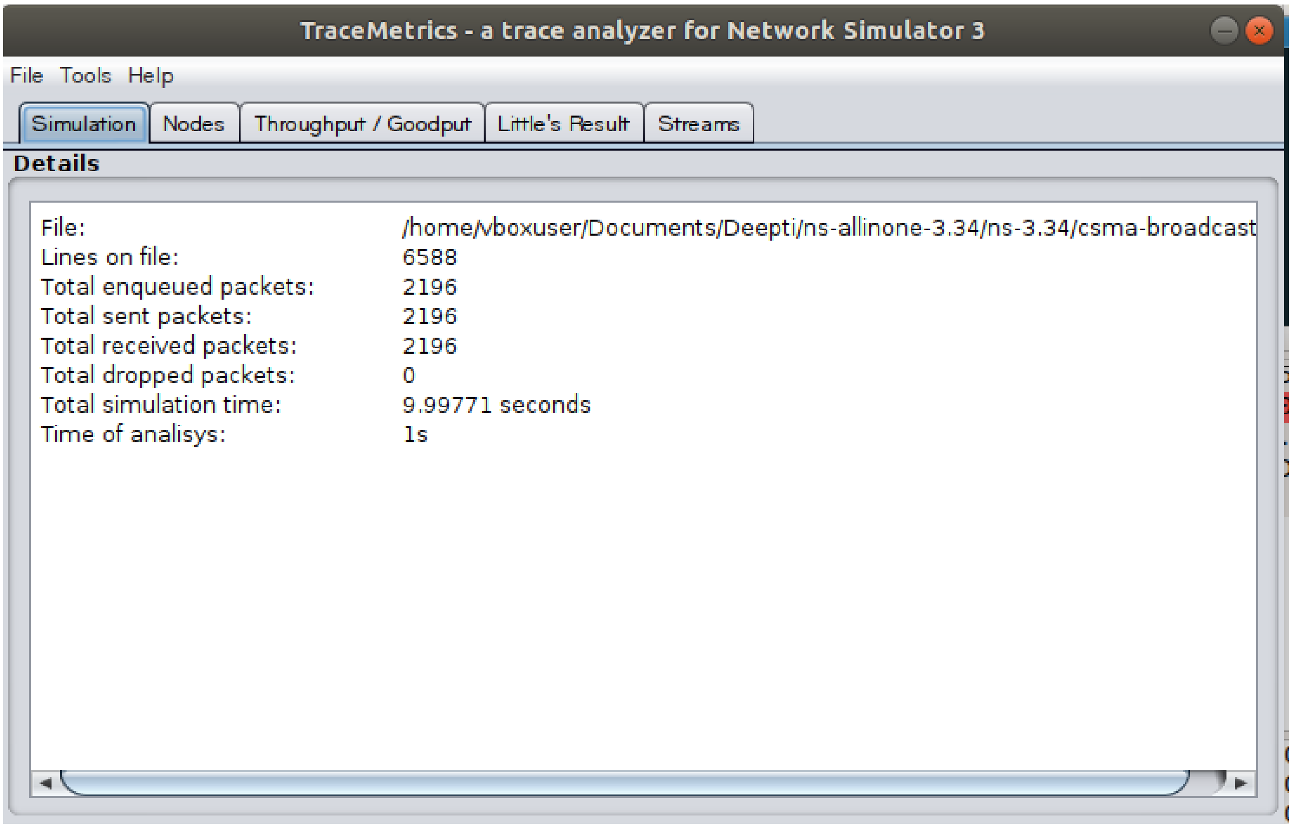
csma-broadcast-0-0.pcap

```
No.  Time       Source        Destination       Protocol  Length Info
   1 0.000000   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [
   2 0.008192   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [
   3 0.016384   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [
   4 0.024576   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [
   5 0.032768   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [
   6 0.040960   10.1.0.1      255.255.255.255   UDP       558 49153 → 9 Len=512 [

Frame 7: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits)
Ethernet II, Src: 00:00:00_00:00:01 (00:00:00:00:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 10.1.0.1, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 49153, Dst Port: 9
Data (512 bytes)

0000  ff ff ff ff ff ff 00 00  00 00 00 01 08 00 45 00   ········ ······E·
0010  02 1c 00 06 00 00 40 11  00 00 0a 01 00 01 ff ff   ······@· ········
0020  ff ff c0 01 00 09 02 08  00 00 00 00 00 00 00 00   ········ ········
```

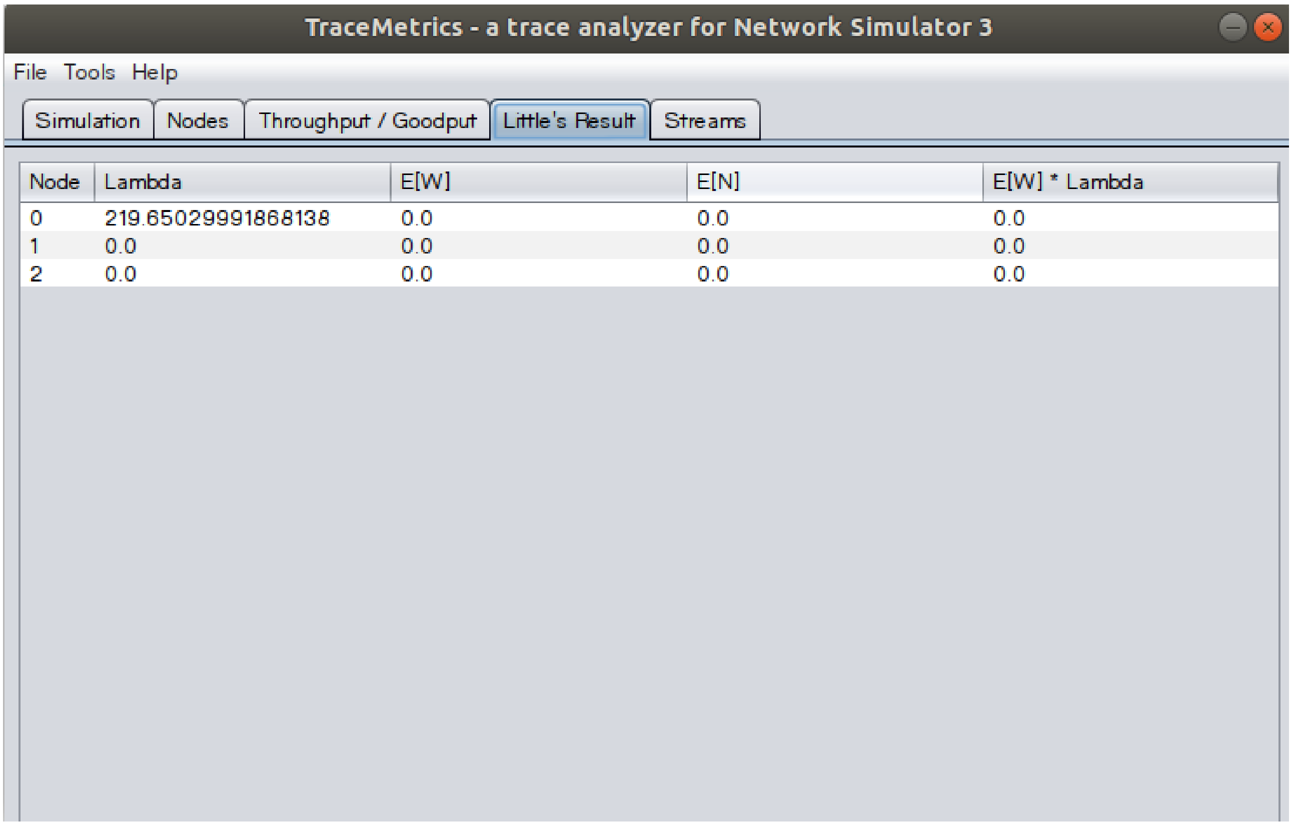
csma-broadcast-0-1.pcap

```
No.  Time       Source        Destination       Protocol  Length Info
  28 0.221184   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [
  29 0.229376   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [
  30 0.237568   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [
  31 0.245760   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [
  32 0.253952   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [
  33 0.262144   192.168.1.1   255.255.255.255   UDP       558 49153 → 9 Len=512 [

Frame 32: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits)
Ethernet II, Src: 00:00:00_00:00:03 (00:00:00:00:00:03), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 49153, Dst Port: 9
Data (512 bytes)

0000  ff ff ff ff ff ff 00 00  00 00 00 03 08 00 45 00   ········ ······E·
0010  02 1c 00 1f 00 00 40 11  00 00 c0 a8 01 01 ff ff   ······@· ········
0020  ff ff c0 01 00 09 02 08  00 00 00 00 00 00 00 00   ········ ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········

Ethernet (eth), 18 bytes     Packets: 1098 · Displayed: 1098 (100.0%)    Profile: D
```

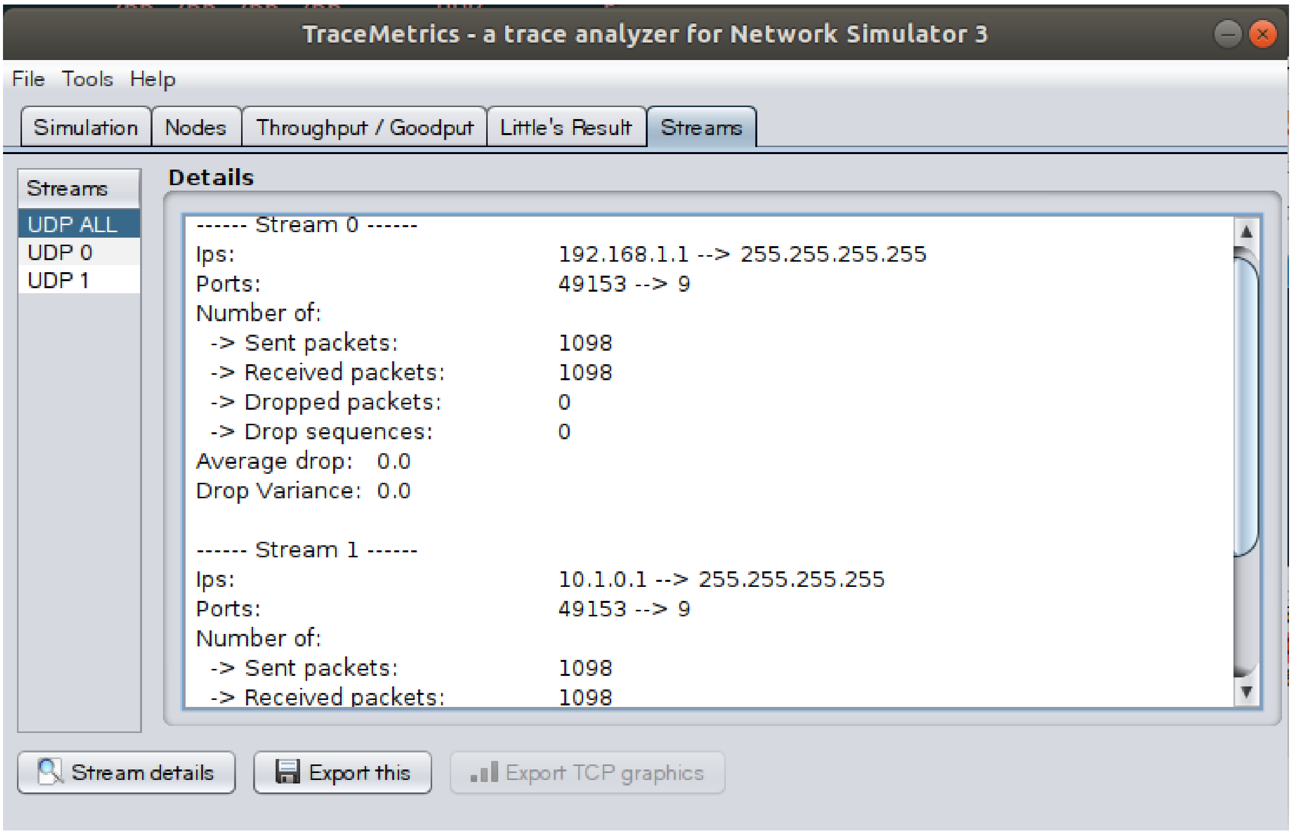
Tracematrix:



```
File:                       /home/vboxuser/Documents/Deepti/ns-allinone-3.34/ns-3.34/csma-broadcast
Lines on file:              6588
Total enqueued packets:     2196
Total sent packets:         2196
Total received packets:     2196
Total dropped packets:      0
Total simulation time:      9.99771 seconds
Time of analisys:           1s
```

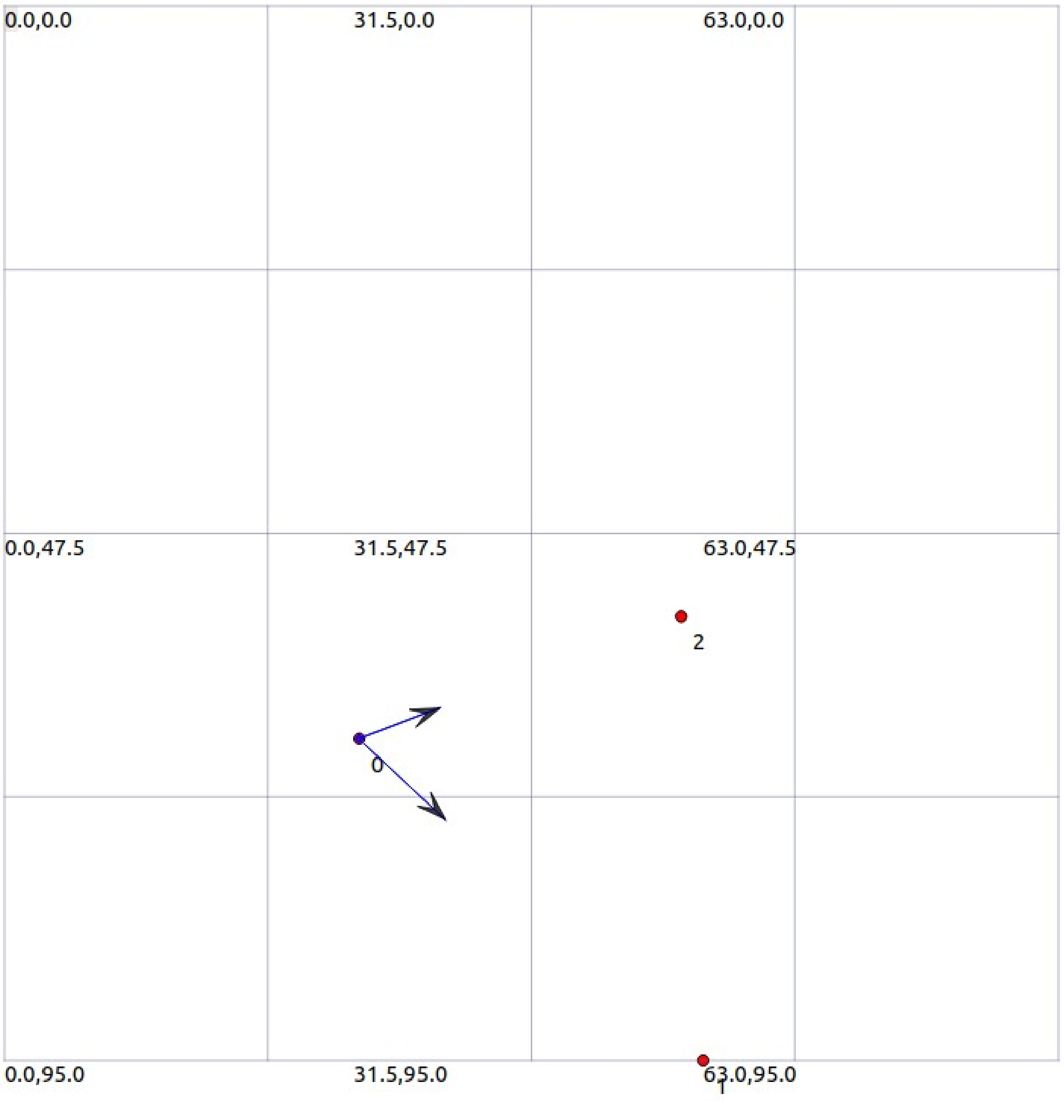


Node	Lambda	E[W]	E[N]	E[W] * Lambda
0	219.65029991868138	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0



```
------ Stream 0 ------
Ips:                      192.168.1.1 --> 255.255.255.255
Ports:                    49153 --> 9
Number of:
  -> Sent packets:        1098
  -> Received packets:    1098
  -> Dropped packets:     0
  -> Drop sequences:      0
Average drop:   0.0
Drop Variance:  0.0

------ Stream 1 ------
Ips:                      10.1.0.1 --> 255.255.255.255
Ports:                    49153 --> 9
Number of:
  -> Sent packets:        1098
  -> Received packets:    1098
```

NetAnim:



CSMA-multicast

Code:

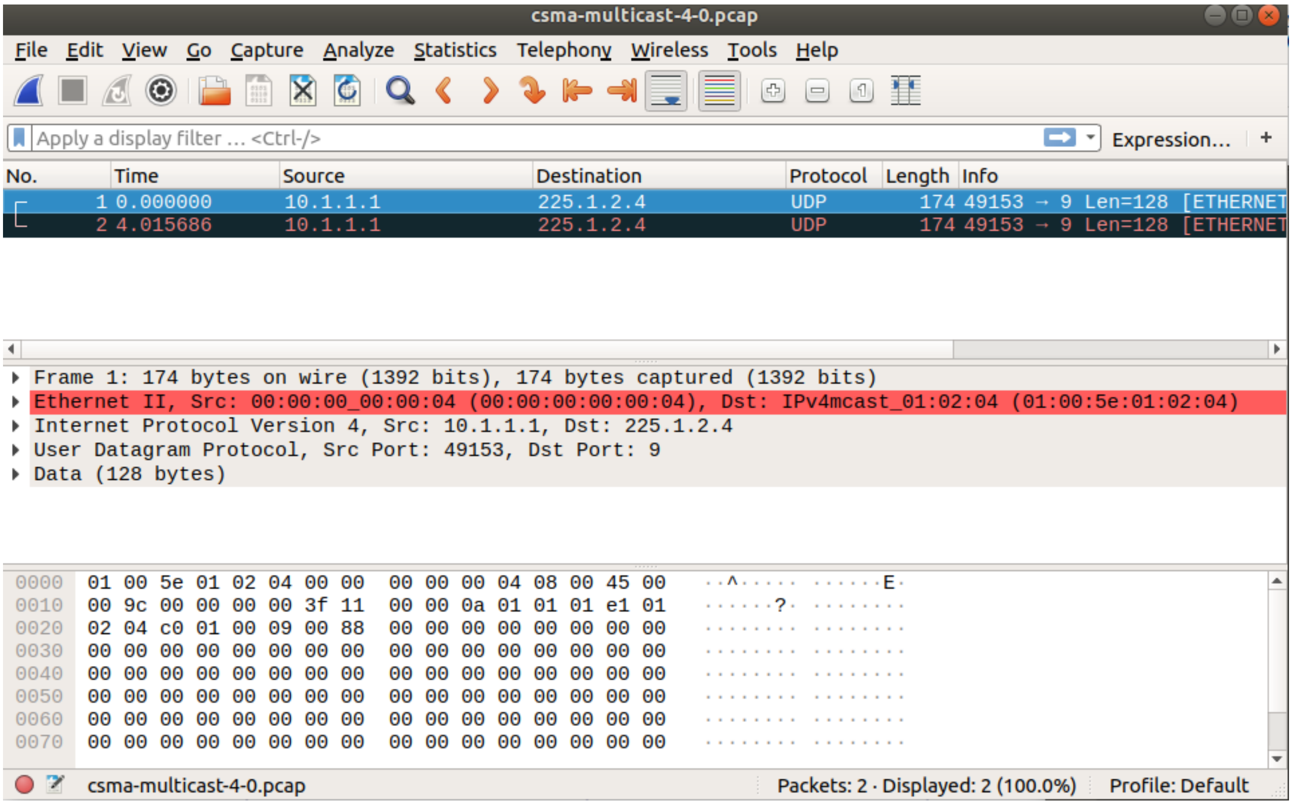
```
#include <iostream>

#include <fstream>


#include "ns3/core-module.h"

#include "ns3/network-module.h"

#include "ns3/csma-module.h"

#include "ns3/applications-module.h"
```

```
#include "ns3/internet-module.h"
#include "ns3/netanim-module.h"
using namespace ns3;

NS_LOG_COMPONENT_DEFINE ("CsmaMulticastExample");

int
main (int argc, char *argv[])
{
  Config::SetDefault ("ns3::CsmaNetDevice::EncapsulationMode", StringValue ("Dix"));
  // run-time, via command-line arguments
  CommandLine cmd;
  cmd.Parse (argc, argv);

  NS_LOG_INFO ("Create nodes.");
  NodeContainer c;
  c.Create (5);
  // We will later want two subcontainers of these nodes, for the two LANs
  NodeContainer c0 = NodeContainer (c.Get (0), c.Get (1), c.Get (2));
  NodeContainer c1 = NodeContainer (c.Get (2), c.Get (3), c.Get (4));

  NS_LOG_INFO ("Build Topology.");
  CsmaHelper csma;
  csma.SetChannelAttribute ("DataRate", DataRateValue (DataRate (5000000)));
  csma.SetChannelAttribute ("Delay", TimeValue (MilliSeconds (2)));

  // We will use these NetDevice containers later, for IP addressing
  NetDeviceContainer nd0 = csma.Install (c0);  // First LAN
  NetDeviceContainer nd1 = csma.Install (c1);  // Second LAN

  NS_LOG_INFO ("Add IP Stack.");
  InternetStackHelper internet;
  internet.Install (c);

  NS_LOG_INFO ("Assign IP Addresses.");
  Ipv4AddressHelper ipv4Addr;
  ipv4Addr.SetBase ("10.1.1.0", "255.255.255.0");
  ipv4Addr.Assign (nd0);
  ipv4Addr.SetBase ("10.1.2.0", "255.255.255.0");
 ipv4Addr.Assign (nd1);

  NS_LOG_INFO ("Configure multicasting.");

 Ipv4Address multicastSource ("10.1.1.1");
 Ipv4Address multicastGroup ("225.1.2.4");

Ipv4StaticRoutingHelper multicast;

 // 1) Configure a (static) multicast route on node n2 (multicastRouter)
 Ptr<Node> multicastRouter = c.Get (2);  // The node in question
 Ptr<NetDevice> inputIf = nd0.Get (2);  // The input NetDevice
 NetDeviceContainer outputDevices;  // A container of output NetDevices
 outputDevices.Add (nd1.Get (0));  // (we only need one NetDevice here)

 multicast.AddMulticastRoute (multicastRouter, multicastSource,
                              multicastGroup, inputIf, outputDevices);

 Ptr<Node> sender = c.Get (0);
 Ptr<NetDevice> senderIf = nd0.Get (0);
 multicast.SetDefaultMulticastRoute (sender, senderIf);

 NS_LOG_INFO ("Create Applications.");

 uint16_t multicastPort = 9;   // Discard port (RFC 863)
```

```
// Configure a multicast packet generator that generates a packet
// every few seconds
OnOffHelper onoff ("ns3::UdpSocketFactory",
               Address (InetSocketAddress (multicastGroup, multicastPort)));
onoff.SetConstantRate (DataRate ("255b/s"));
onoff.SetAttribute ("PacketSize", UintegerValue (128));
ApplicationContainer srcC = onoff.Install (c0.Get (0));

//
// Tell the application when to start and stop.
//
srcC.Start (Seconds (1.));
srcC.Stop (Seconds (10.));

// Create an optional packet sink to receive these packets
PacketSinkHelper sink ("ns3::UdpSocketFactory",
                  InetSocketAddress (Ipv4Address::GetAny (), multicastPort));

ApplicationContainer sinkC = sink.Install (c1.Get (2)); // Node n4
// Start the sink
sinkC.Start (Seconds (1.0));
sinkC.Stop (Seconds (10.0));

NS_LOG_INFO ("Configure Tracing.");
AnimationInterface anim("csma-multicast.xml");
AsciiTraceHelper ascii;
csma.EnableAsciiAll (ascii.CreateFileStream ("csma-multicast.tr"));

// Also configure some tcpdump traces; each interface will be traced.
// The output files will be named:
//     csma-multicast-<nodeId>-<interfaceId>.pcap
```

```
  // and can be read by the "tcpdump -r" command (use "-tt" option to
  // display timestamps correctly)
  csma.EnablePcapAll ("csma-multicast", false);

  //
  // Now, do the actual simulation.
  //
  NS_LOG_INFO ("Run Simulation.");
  Simulator::Run ();
  Simulator::Destroy ();
  NS_LOG_INFO ("Done.");
}
```

Wire Shark:



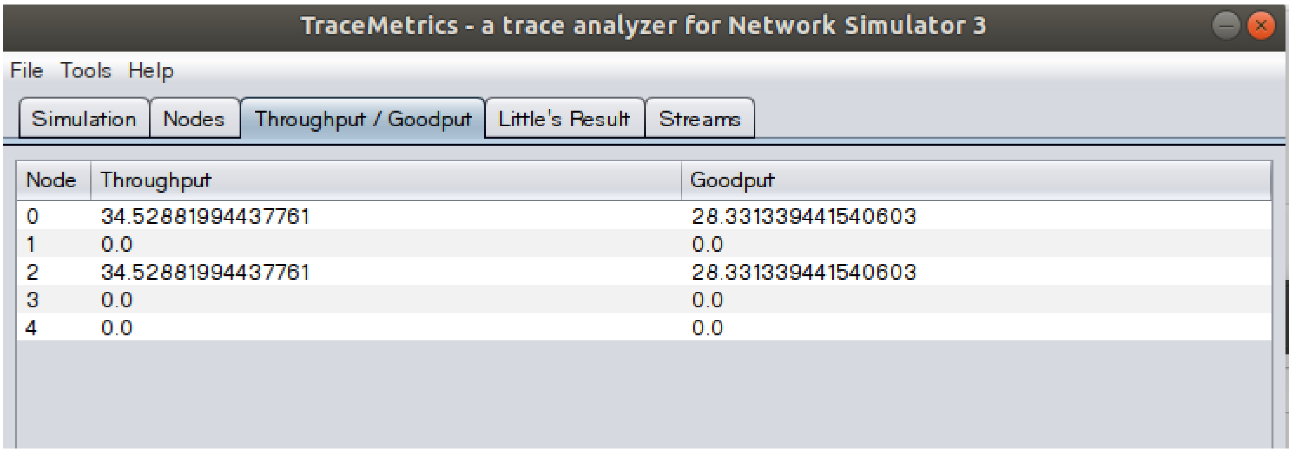
```
No.  Time        Source      Destination  Protocol  Length  Info
   1 0.000000    10.1.1.1    225.1.2.4    UDP          174  49153 → 9 Len=128 [ETHERNET
   2 4.015686    10.1.1.1    225.1.2.4    UDP          174  49153 → 9 Len=128 [ETHERNET
```

```
▸ Frame 1: 174 bytes on wire (1392 bits), 174 bytes captured (1392 bits)
▸ Ethernet II, Src: 00:00:00_00:00:04 (00:00:00:00:00:04), Dst: IPv4mcast_01:02:04 (01:00:5e:01:02:04)
▸ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 225.1.2.4
▸ User Datagram Protocol, Src Port: 49153, Dst Port: 9
▸ Data (128 bytes)
```

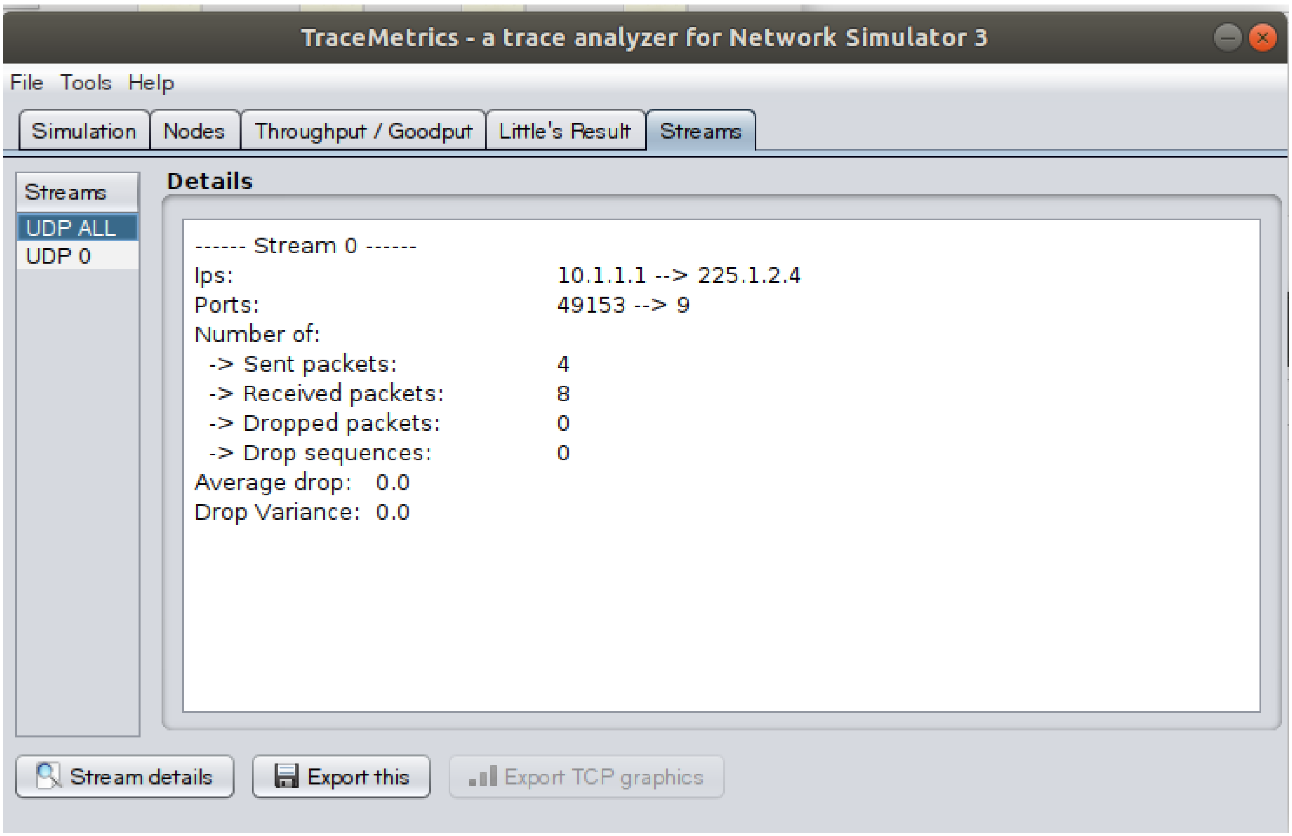
```
0000  01 00 5e 01 02 04 00 00  00 00 00 04 08 00 45 00   ··^·····  ······E·
0010  00 9c 00 00 00 00 3f 11  00 00 0a 01 01 01 e1 01   ······?·  ········
0020  02 04 c0 01 00 09 00 88  00 00 00 00 00 00 00 00   ········  ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········  ········
```

csma-multicast-4-0.pcap | Packets: 2 · Displayed: 2 (100.0%) | Profile: Default

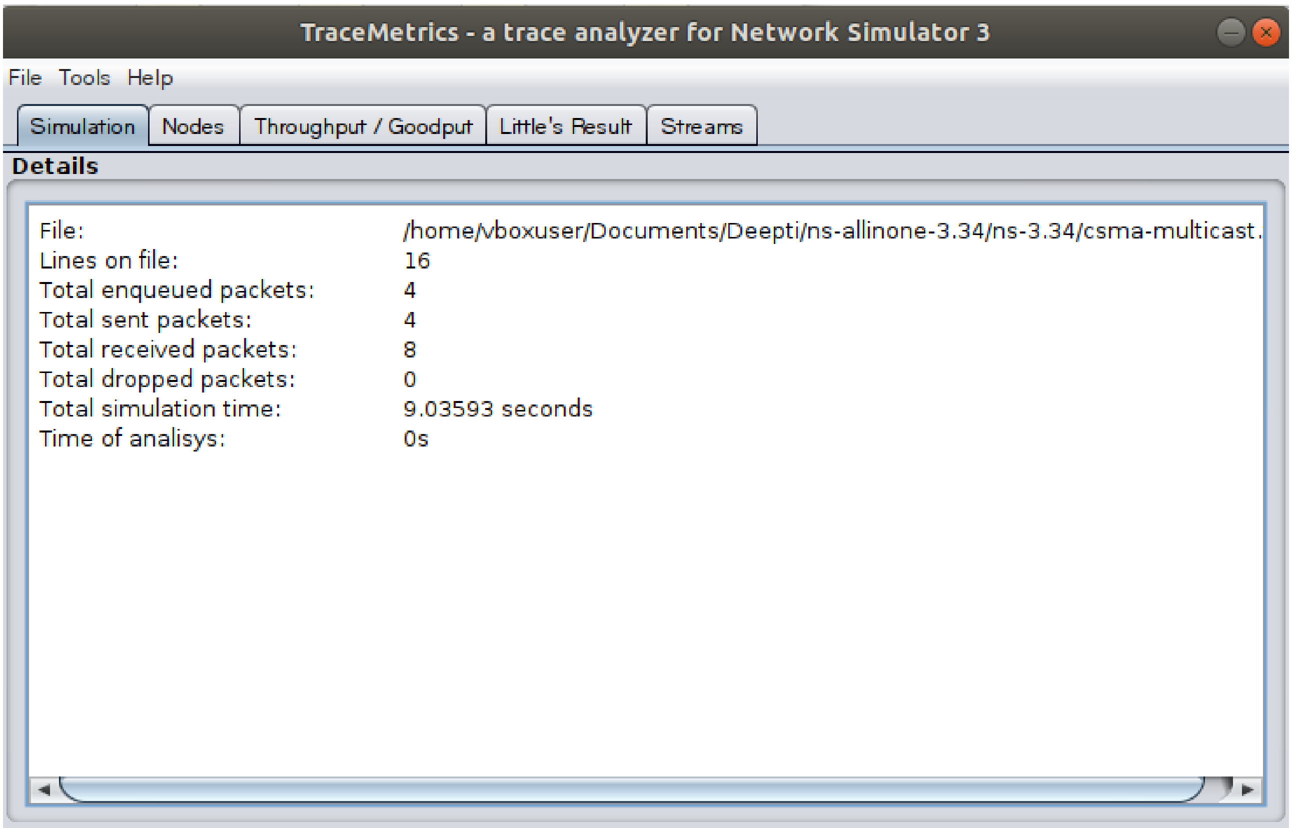
Tracematrix:



Node	Throughput	Goodput
0	34.52881994437761	28.331339441540603
1	0.0	0.0
2	34.52881994437761	28.331339441540603
3	0.0	0.0
4	0.0	0.0

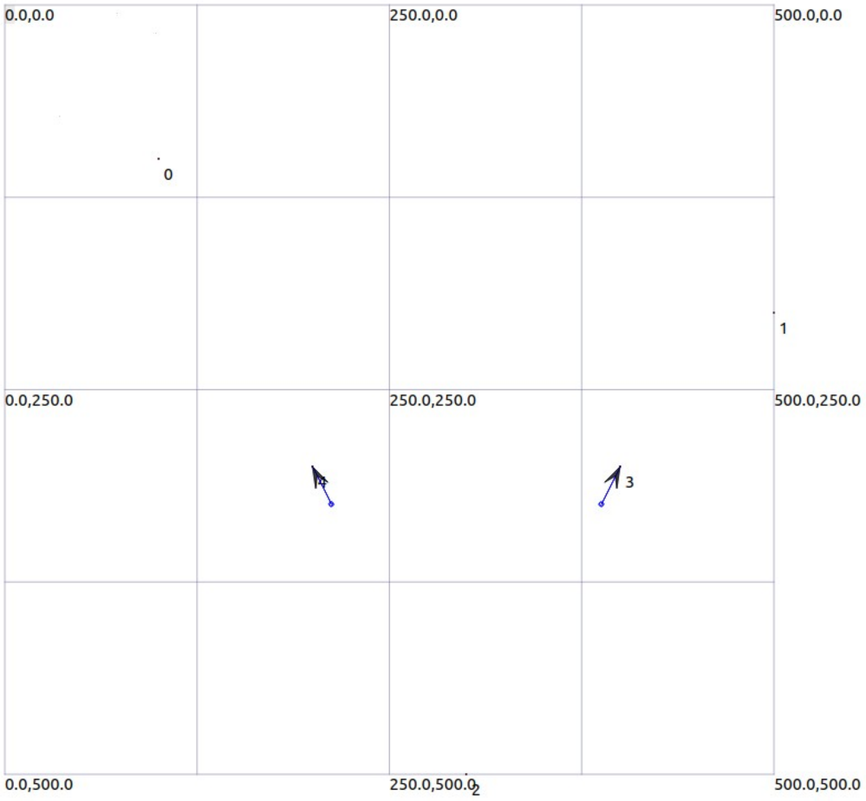


```
------ Stream 0 ------
Ips:                          10.1.1.1 --> 225.1.2.4
Ports:                        49153 --> 9
Number of:
  -> Sent packets:            4
  -> Received packets:        8
  -> Dropped packets:         0
  -> Drop sequences:          0
Average drop:   0.0
Drop Variance:  0.0
```



```
File:                       /home/vboxuser/Documents/Deepti/ns-allinone-3.34/ns-3.34/csma-multicast.
Lines on file:              16
Total enqueued packets:     4
Total sent packets:         4
Total received packets:     8
Total dropped packets:      0
Total simulation time:      9.03593 seconds
Time of analisys:           0s
```

NetAnim:



CSMA-one-subnet

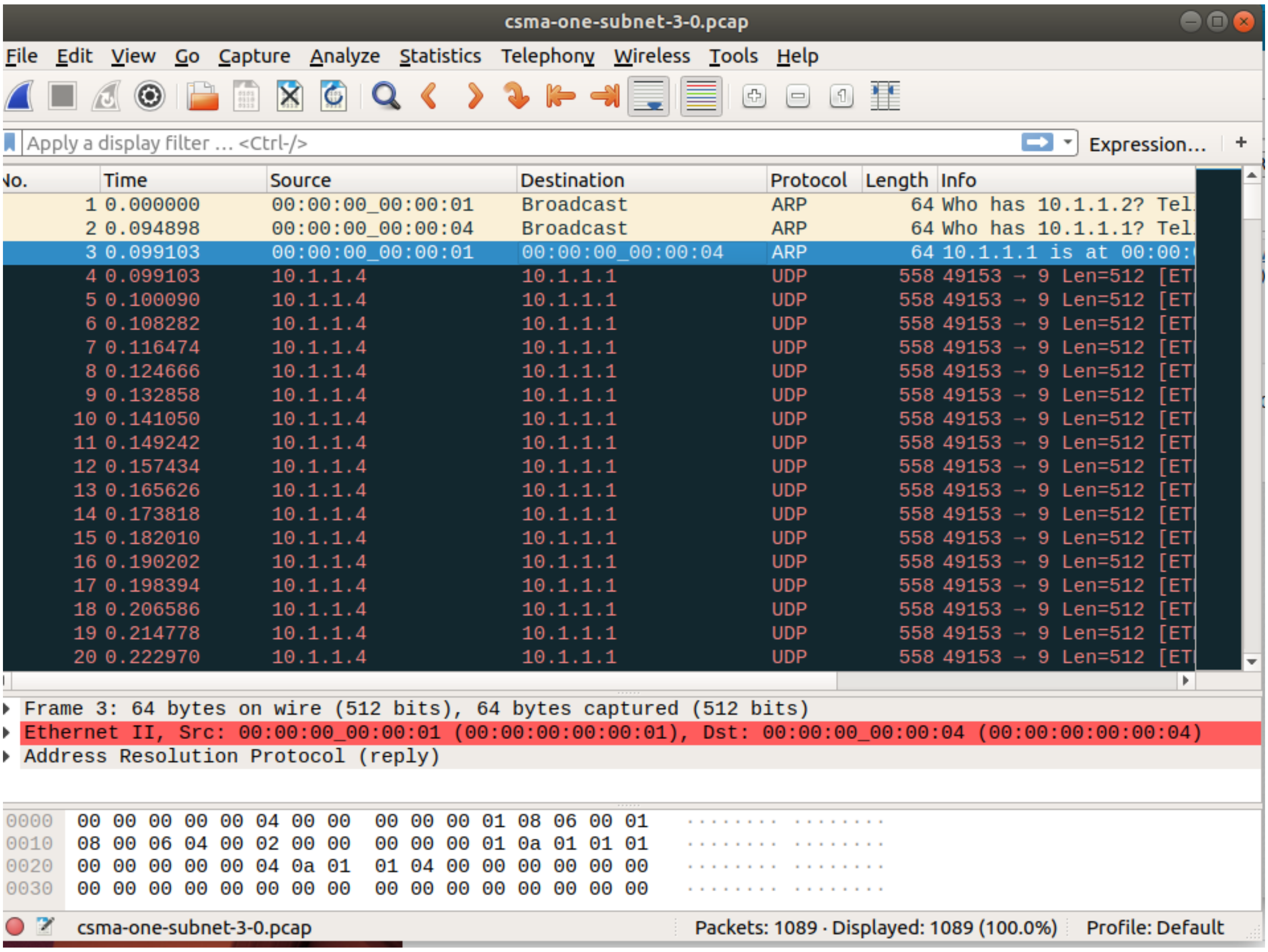
Code:

```
#include <iostream>

#include <fstream>

#include "ns3/core-module.h"

#include "ns3/network-module.h"

#include "ns3/csma-module.h"

#include "ns3/applications-module.h"
```

```
#include "ns3/internet-module.h"
#include "ns3/netanim-module.h"
using namespace ns3;

NS_LOG_COMPONENT_DEFINE ("CsmaOneSubnetExample");

int main (int argc, char *argv[]){

#if 0
LogComponentEnable ("CsmaOneSubnetExample", LOG_LEVEL_INFO);
#endif
CommandLine cmd;
cmd.Parse (argc, argv);

NS_LOG_INFO ("Create nodes.");
NodeContainer nodes;
nodes.Create (4);

NS_LOG_INFO ("Build Topology");
CsmaHelper csma;
csma.SetChannelAttribute ("DataRate", DataRateValue (5000000));
csma.SetChannelAttribute ("Delay", TimeValue (MilliSeconds (2)));

NetDeviceContainer devices = csma.Install (nodes);
InternetStackHelper internet;
internet.Install (nodes);

NS_LOG_INFO ("Assign IP Addresses.");
Ipv4AddressHelper ipv4;
ipv4.SetBase ("10.1.1.0", "255.255.255.0");
Ipv4InterfaceContainer interfaces = ipv4.Assign (devices);
```

```
NS_LOG_INFO ("Create Applications.");
uint16_t port = 9;

OnOffHelper onoff ("ns3::UdpSocketFactory",
                   Address (InetSocketAddress (interfaces.GetAddress (1), port)));
onoff.SetConstantRate (DataRate ("500kb/s"));

ApplicationContainer app = onoff.Install (nodes.Get (0));

app.Start (Seconds (1.0));
app.Stop (Seconds (10.0));

  PacketSinkHelper sink ("ns3::UdpSocketFactory",
                         Address (InetSocketAddress (Ipv4Address::GetAny (), port)));
  app = sink.Install (nodes.Get (1));
  app.Start (Seconds (0.0));

  onoff.SetAttribute ("Remote",
                      AddressValue (InetSocketAddress (interfaces.GetAddress (0), port)));
  app = onoff.Install (nodes.Get (3));
  app.Start (Seconds (1.1));
  app.Stop (Seconds (10.0));

  app = sink.Install (nodes.Get (0));
  app.Start (Seconds (0.0));

  NS_LOG_INFO ("Configure Tracing.");
  AnimationInterface anim("csma-one-subnet.xml");
  AsciiTraceHelper ascii;
  csma.EnableAsciiAll (ascii.CreateFileStream ("csma-one-subnet.tr"));

  csma.EnablePcapAll ("csma-one-subnet", false);
  NS_LOG_INFO ("Run Simulation.");
  Simulator::Run ();
  Simulator::Destroy ();
  NS_LOG_INFO ("Done.");
}
```

Wire Shark:



csma-one-subnet-3-0.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Expression…
+

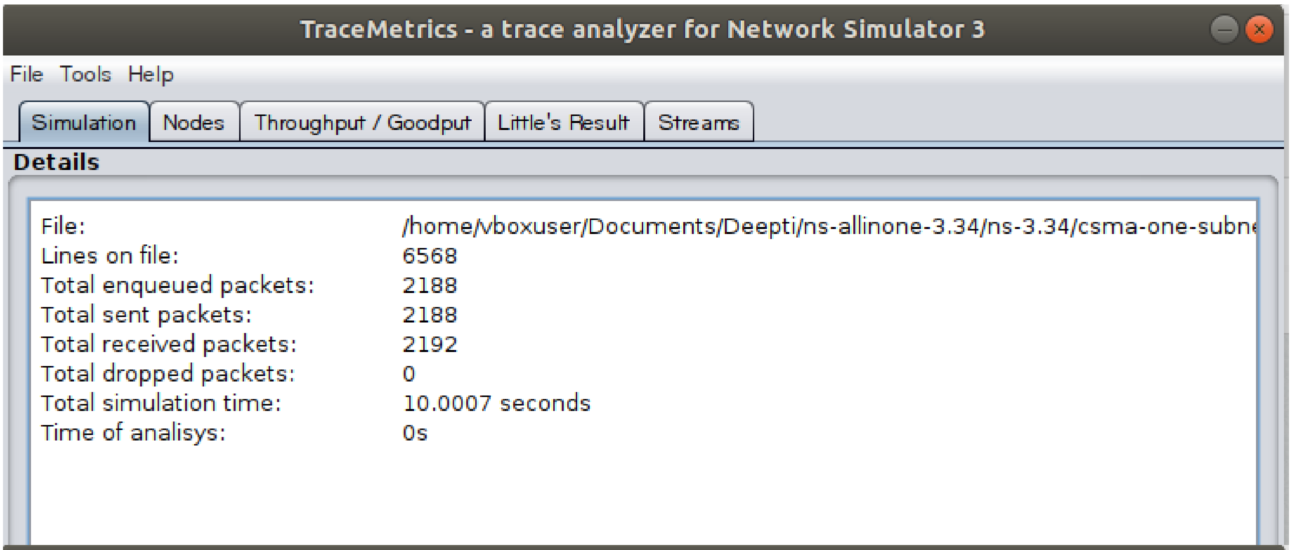
```
No.  Time      Source               Destination          Protocol Length Info
   1 0.000000  00:00:00_00:00:01    Broadcast            ARP        64 Who has 10.1.1.2? Tel
   2 0.094898  00:00:00_00:00:04    Broadcast            ARP        64 Who has 10.1.1.1? Tel
   3 0.099103  00:00:00_00:00:01    00:00:00_00:00:04    ARP        64 10.1.1.1 is at 00:00:
   4 0.099103  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
   5 0.100090  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
   6 0.108282  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
   7 0.116474  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
   8 0.124666  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
   9 0.132858  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  10 0.141050  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  11 0.149242  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  12 0.157434  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  13 0.165626  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  14 0.173818  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  15 0.182010  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  16 0.190202  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  17 0.198394  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  18 0.206586  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  19 0.214778  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
  20 0.222970  10.1.1.4             10.1.1.1             UDP       558 49153 → 9 Len=512 [ET
```

```
▸ Frame 3: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
▸ Ethernet II, Src: 00:00:00_00:00:01 (00:00:00:00:00:01), Dst: 00:00:00_00:00:04 (00:00:00:00:00:04)
▸ Address Resolution Protocol (reply)
```

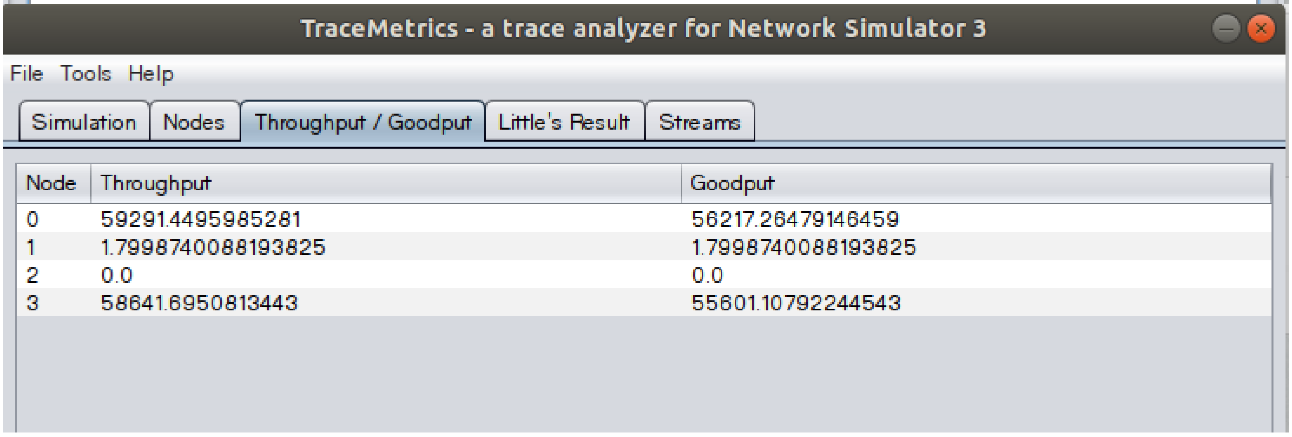
```
0000  00 00 00 00 00 04 00 00  00 00 00 01 08 06 00 01   ........ ........
0010  08 00 06 04 00 02 00 00  00 00 00 01 0a 01 01 01   ........ ........
0020  00 00 00 00 00 04 0a 01  01 04 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
```

csma-one-subnet-3-0.pcap   Packets: 1089 · Displayed: 1089 (100.0%)   Profile: Default

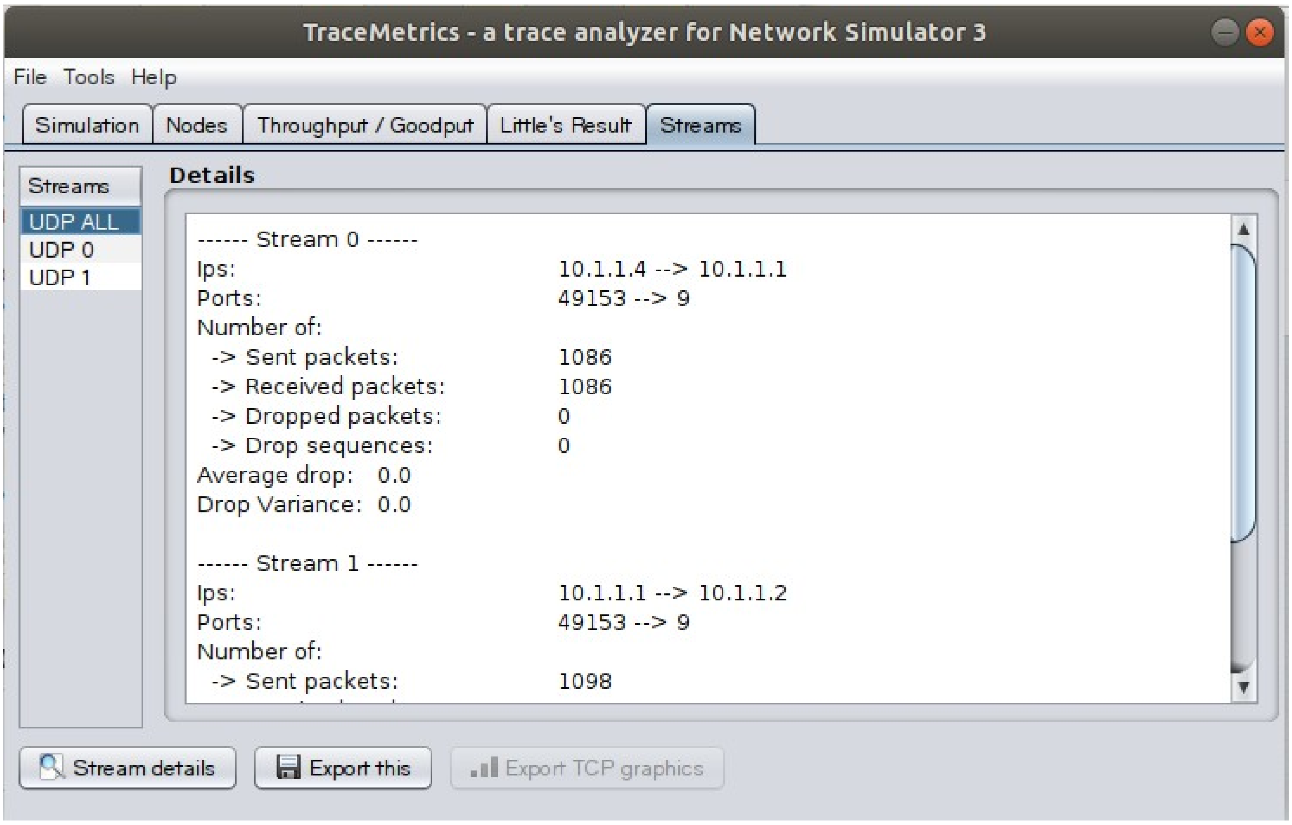
Tracematrix:



```
File:                       /home/vboxuser/Documents/Deepti/ns-allinone-3.34/ns-3.34/csma-one-subne
Lines on file:              6568
Total enqueued packets:     2188
Total sent packets:         2188
Total received packets:     2192
Total dropped packets:      0
Total simulation time:      10.0007 seconds
Time of analisys:           0s
```

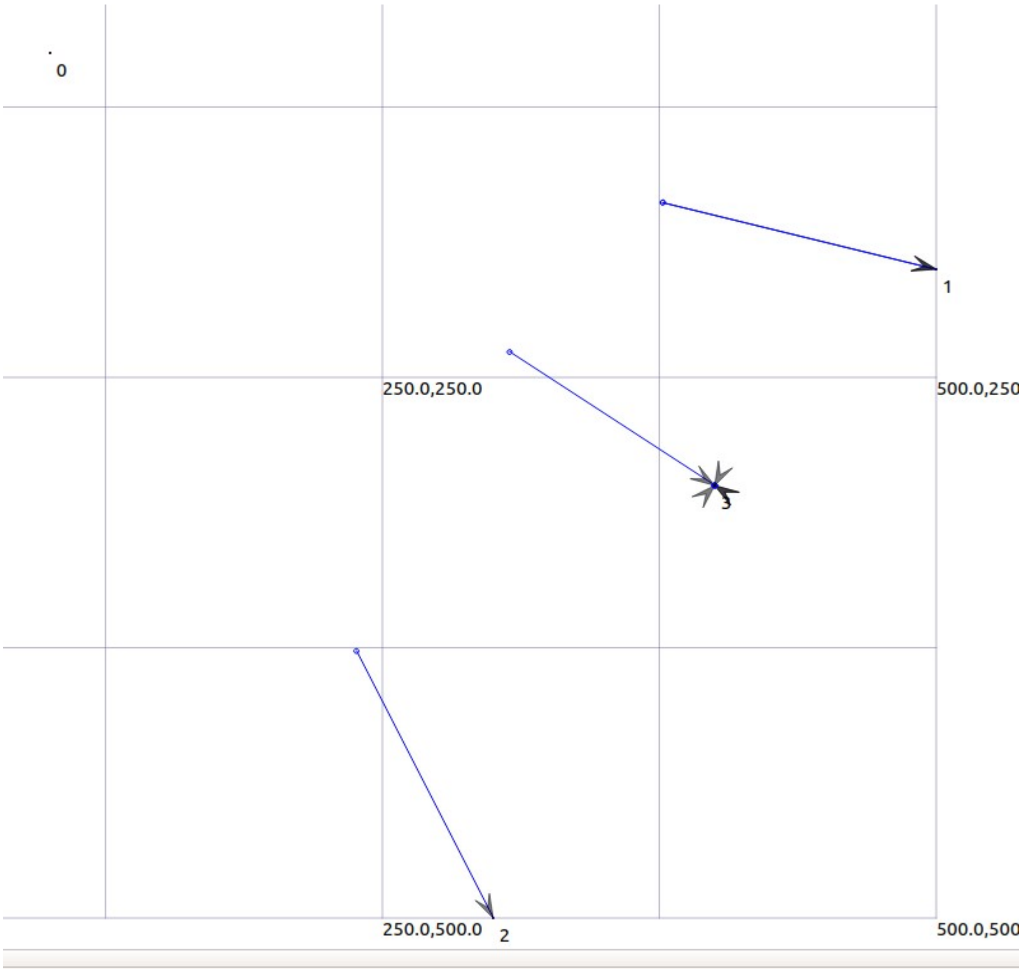


Node	Throughput	Goodput
0	59291.4495985281	56217.26479146459
1	1.7998740088193825	1.7998740088193825
2	0.0	0.0
3	58641.6950813443	55601.10792244543



```
------ Stream 0 ------
Ips:                          10.1.1.4 --> 10.1.1.1
Ports:                        49153 --> 9
Number of:
  -> Sent packets:            1086
  -> Received packets:        1086
  -> Dropped packets:         0
  -> Drop sequences:          0
Average drop:   0.0
Drop Variance:  0.0

------ Stream 1 ------
Ips:                          10.1.1.1 --> 10.1.1.2
Ports:                        49153 --> 9
Number of:
  -> Sent packets:            1098
```

NetAnim:



<END>