Lab Report				
Course Title: Computer Networks Lab				
Autumn 2024				
Section:7BF				
Lab No: 5				
Name of Labwork: Create your own message type i.e. packet which has 3 fields namely source, destination and hopCount. Run several configurations and show hop counts in a plot. Also calculate the mean, standard deviation of hop counts.				
Student's ID : C213246 Date of Performance : 0 2 .10.2024				
Date of Submission : 09.10.2024				
Team Name : ProtocolPros				
Marks .				
•				

**Name of Labwork:** Create your own message type i.e. packet which has 3 fields namely source, destination and hopCount. Run several configurations and show hop counts in a plot. Also calculate the mean, standard deviation of hop counts.

#### 1. Introduction:

In this lab, we will create a custom message type (packetPP) with three fields: source, destination, and hopCount. The objective is to simulate a network where packets are forwarded from one node to another, counting the number of hops until they reach their destination. After running several configurations, we will gather hop count data, calculate its mean and standard deviation, and display the results using a plot.

#### 2. Constructing Network(NED):

We will define a simple network of nodes using OMNeT++'s NED language. Each node will randomly forward messages through its connected gates until the message reaches its destination.

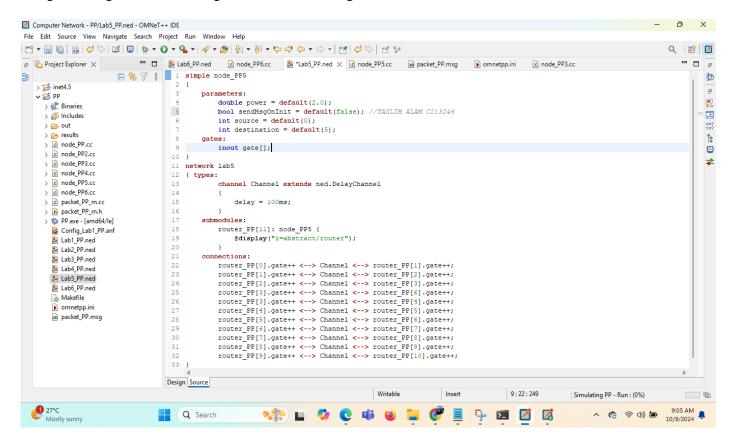


Figure 1: Laptop view of NED file

#### 3. Building Module(C++ file):

The custom packet type will be defined with three fields: source, destination, and hopCount. Nodes will randomly forward packets to other nodes until they reach the destination.

```
Computer Network - PP/node_PP5.cc - OMNeT++ IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Q 🔡 🗹
                                                                                                                                                                              - - -
                             □ □ Lab6_PP.ned ⓒ node_PP6.cc ጮ *Lab5_PP.ned ⓒ node_PP5.cc X ⋒ packet_PP.msg
₽ Project Explorer ×
                                                                                                                       omnetpp.ini
                                                                                                                                      node PP3.cc
                                        1⊖/*
                                              node_ProtocolPros.cc
Created by : TASLIM ALAM
Created on: OCT 10 2024
    > 🞏 inet4.5
    V 👺 PP
                                                                                                                                                                                    > 🐉 Binaries
                                           * Team Name : Protocol Pros
                                                                                                                                                                                    .P.
      > 🞒 Includes
                                                                                                                                                                                    X=?
Y=?
      > 🗁 out
                                           #include <omnetpp.h>
      > 🗁 results
                                           #include "packet_PP_m.h"
      > 🖟 node_PP.cc
                                           using namespace omnetop:
                                                                                                                                                                                    ▣
      > lc node_PP2.cc
                                      10⊖ class node_PP5 : public cSimpleModule
      > 🖻 node_PP3.cc
      > le node_PP4.cc
                                            private:
      > .c node_PP5.cc
> .c node_PP6.cc
                                               double counter;
                                               bool sendMsgOnInit;
      > 🖟 packet_PP_m.cc
      > h packet_PP_m.h
      > pp.exe - [amd64/le]
Config_Lab1_PP.anf
                                               virtual void initialize() override;
                                               virtual void handleMessage (cMessage *msg) override;
virtual void forwardMessage (packetPP *msg);
        Lab1 PP.ned
        Lab2_PP.ned
                                               virtual packetPP *generateMessage();
        Lab3 PP.ned
                                       22 Define Module(node PP5);
        Lab4_PP.ned
       Lab5_PP.ned
                                      △24⊖ void node_PP5::initialize()
        Lab6 PP.ned
       Makefile
        omnetpp.ini
                                               int src=par("source");
        packet_PP.msg
                                               int dst=par("destination");
                                               if (getIndex() == src && par("sendMsgOnInit").boolValue() == true) {
                                                   packetPP *msg=generateMessage();
                                                    scheduleAt(0.0,msg);
                                                                                             Writable
                                                                                                               Smart Insert
                                                                                                                                                 Simulating PP - Run : (0%)
                                                                                                                                                                               -
   1 27°C
                                                                                                                                                                           9:06 AM
                                                                                        Цi
                                                                                                                             Q Search
                                                                                                                                                      ^ ♠ ♠ ♠ ♠ ♠ 9:06 AM ♣ 10/9/2024 ♣
```

Figure 2.1: Laptop view of cc file

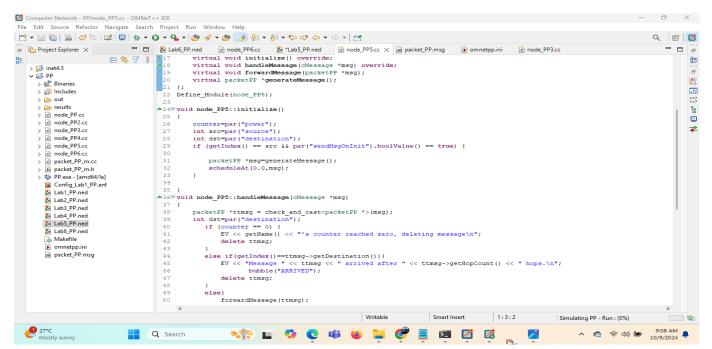


Figure 2.2: Laptop view of cc file

```
Computer Network - PP/node_PP5.cc - OMNeT++ IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Q 🔡 🗾
₽ Project Explorer ×
                         □ 🗈 Lab6_PP.ned 🗈 node_PP6.cc 🖺 *Lab5_PP.ned 🗈 node_PP5.cc 🗙 📠 packet_PP.msg 🕟 omnetpp.ini 🗈 node_PP3.cc
                                                                                                                                                                   - -
                                                                                                                                                                         8
                      □ 🕏 🎖 🕴
                                    47
                                                   delete timsg:
                                                                                                                                                                         ð
   > $\frac{1}{2}$ inet4.5

✓ $\frac{1}{2}$ PP
                                                                                                                                                                         8
                                    49
                                               else{
                                                   forwardMessage(ttmsg);
     > 🐉 Binaries
                                                                                                                                                                        X=?
Y=?
                                    51
     > 🔊 Includes
     > 海 out
                                    53
     > 🗁 results
                                    54@ void node_PP5::forwardMessage(packetPP *msg){
     > onode_PP.cc
                                    55
56
                                            counter = counter - 0.1;
                                                                                                                                                                         ⊜
     > lc node PP2.cc
                                            msg->setHopCount(msg->getHopCount()+1);
                                                                                                                                                                        *
     > @ node_PP3.cc
                                    57
58
                                            int n= gateSize("gate");
     > 🖻 node_PP4.cc
                                            int k=intuniform(0,n-1);
     > onode_PP5.cc
     > @ node_PP6.cc
                                    60
                                           EV<<"Forwarding message "<<msg<<"on gate["<<k<<"]\n";
     > i packet_PP_m.cc
     > h packet_PP_m.h
                                    62
     > pp.exe - [amd64/le]
       Config_Lab1_PP.anf
                                    64@packetPP*node PP5::generateMessage()
       Lab1 PP.ned
                                    66
                                            // Produce source and destination addresses.
       Lab2_PP.ned
                                            int src = par("source");
       Lab3_PP.ned
                                    68
69
                                            int dst = par("destination"); // our module index
       Lab4 PP.ned
       Lab5_PP.ned
       Lab6_PP.ned
                                            char msgname[20];
       Makefile
                                     72
73
                                            sprintf(msgname, "packetPP-%d-to-%d", src, dst);
       omnetpp.ini
       packet_PP.msg
                                            // Create message object and set source and destination field.
packetPP *msg = new packetPP(msgname);
                                     74
75
76
77
                                            msg->setSource(src);
                                            msq->setDestination(dst);
                                    78
79 }
                                                                                                                       1:3:2
                                                                                       Writable
                                                                                                        Smart Insert
                                                                                                                                       Simulating PP - Run : (0%)
                                                                                                                                             ^ ⑤ ۞ Φ) 🖆 9:07 AM 💂
                                  Q Search
                                                           💖 🔲 🥠 🙋 📫
                                                                                                                     🗷 🗹 📝
```

Figure 2.3: Laptop view of cc file

## 4. Initializing simulation(ini file):

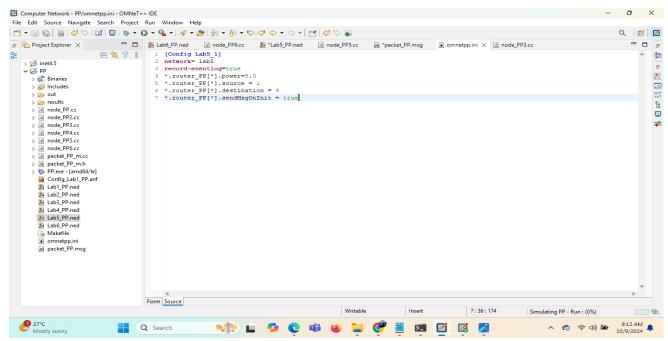


Figure 3: Laptop view of ini file

### 5. Msg file:

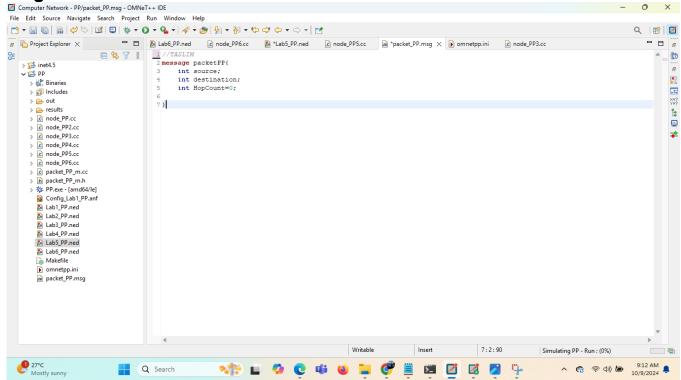


Figure 4: Laptop view of msg file

# 6. Experiment:

In this extended lab, we use the msg file to track the routing information of messages within a network or communication system. Also hop count to the number of intermediary nodes or hops that the message traverses while moving from the source to the destination. Each hop typically represents a network node or a routing point that forwards the message towards its final destination.

## 7. Result and Analysis:

## **Average Hop count:**

Serial no	Source	Destination	Hops
1	0	5	17
2	3	5	11
3	0	3	45
4	4	5	2
5	5	4	1

### **Average count:**

Sum = 76
Count=5

AVERAGE =15.2

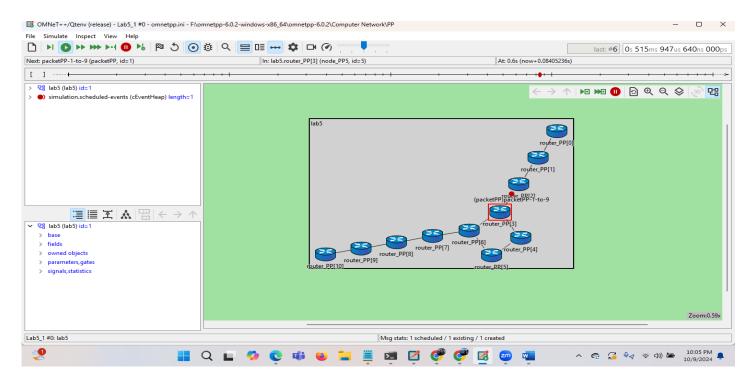


Figure 5.1: Laptop view of result

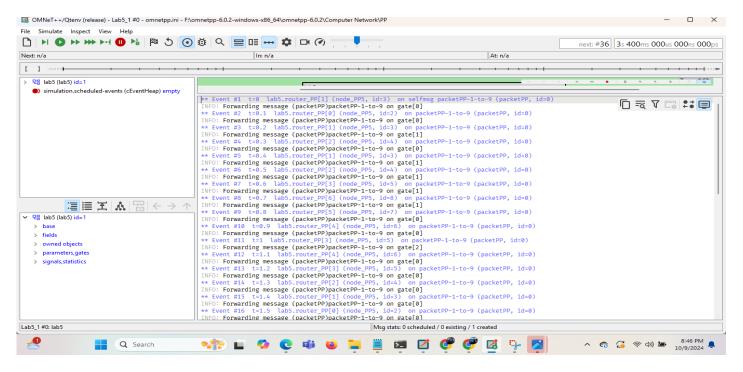


Figure 5.2: Laptop view of result

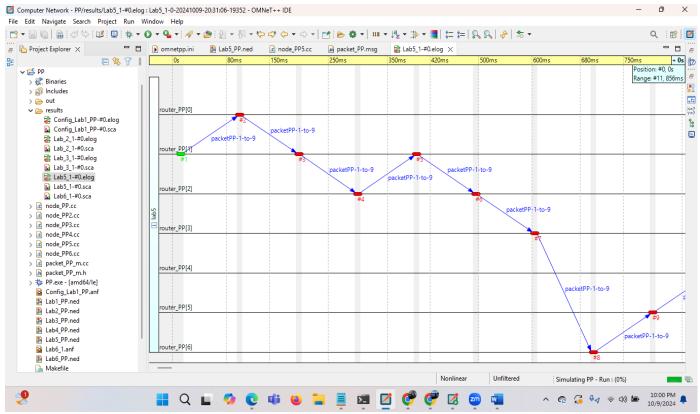


Figure 5.3: Laptop view of sequence diagram

#### **Result and Analysis:**

The average hop count is calculated by summing up all hop counts from the table and then dividing the total by the number of entries in the table. By analyzing the hop counts, we can identify potential areas for optimization, evaluate network congestion, and ensure reliable communication between sources and destinations. The average hop count provides a quick metric to gauge overall network performance, aiding in decision-making for network enhancements and improvements.

#### 8. Conclusion:

Doing this lab work showcased its associated hop counts provides actionable insights for improving the communication network's performance, reliability, and efficiency. It aids network administrators and engineers in making informed decisions to enhance the overall quality of service for message transmission.