COL 334/672 Assignment 4

Deadline: 11:59 P.M 10 Nov 2022

This assignment has 2 parts. The first part is on **Transport Layer**. The second part is on **Network Layer**. Part A carries 40% weightage and Part B carries 60% weightage.

Part A
Transport Layer

In this part, we will learn how to create our own congestion control protocol in ns3(3.29).

NS-3 by default uses TcpNewReno as the congestion control strategy (Link). It consists of the following functions of interest to us

1. **TcpNewReno::SlowStart:**
   "ns-allinone-3.29/ns-3.29/src/internet/model/tcp-congestion-ops.cc"

2. **TcpNewReno::CongestionAvoidance:**
   "ns-allinone-3.29/ns-3.29/src/internet/model/tcp-congestion-ops.cc"

Our goal will be to create a new congestion control algorithm called TcpNewRenoPlus. For this, you need to create 2 new files TcpNewRenoPlus.cc and TcpNewRenoPlus.h in *"ns-allinone-3.29/ns-3.29/src/internet/model"*. You can refer to the following link on how to write a new congestion control algorithm in NS-3.

You need to reuse most of the code of TcpNewReno from *"ns-allinone-3.29/ns-3.29/src/internet/model/tcp-congestion-ops.cc"* and *""ns-allinone-3.29/ns-3.29/src/internet/model/tcp-congestion-ops.h"* and the algorithmic changes need to be done only for the SlowStart and CongestionAvoidance functions in your TcpNewRenoPlus.cc.

The way you implement it by using inheritance of base class TcpCongestionOps / TcpNewReno /or any other way is your decision. The important point is that all functionalities in TcpNewRenoPlus except the below two should be of TcpNewReno.

1. TcpNewReno::SlowStart
2. TcpNewReno::CongestionAvoidance

In **Slow Start phase of TcpNewRenoPlus**, increase the congestion window size$Cwnd$ as:

$$Cwnd = Cwnd + (SegmentSize)^{1.91}/Cwnd$$

In the **Congestion Avoidance** phase of TcpNewRenoPlus, update the congestion window as

$$Cwnd = Cwnd + 0.51 * SegmentSize$$

In order to understand the behavior of the network using the newly designed congestion control protocol, we will use the topology described in Figure 1 below.

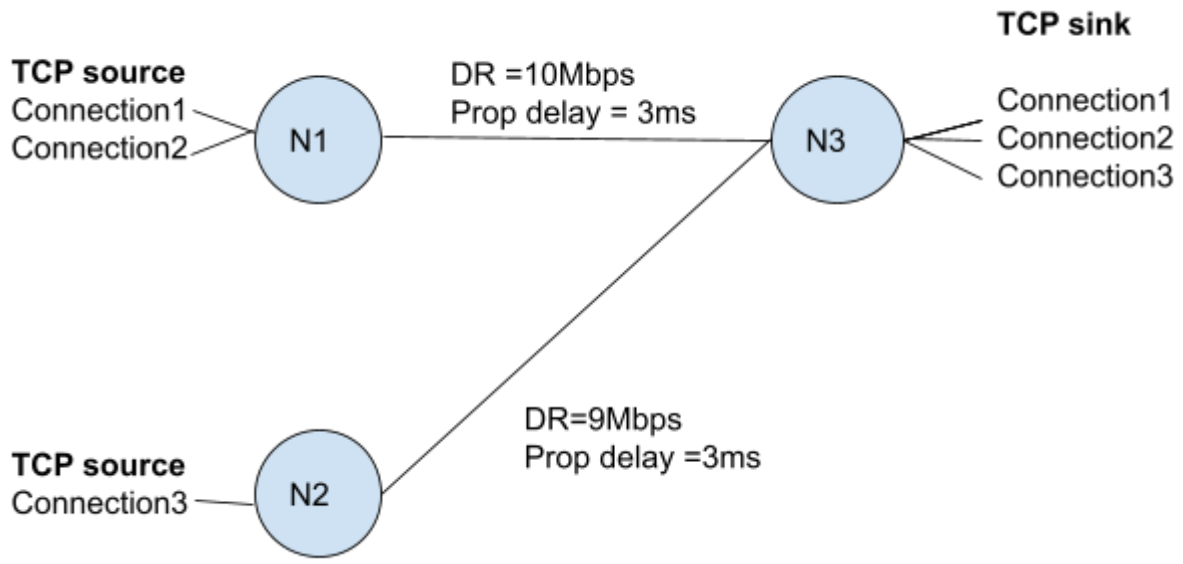

Figure 1

In Figure.1, N2-N3 and N1-N3 are connected via a point-to-point link. DR means Data Rate of the channel. Here we have TCP source at N1 and N2 and TCP sink at N3.

We will use the below 2 configurations of the network described in Fig.1 to analyze the behavior of congestion control algorithms:

- **Configuration1:** All senders use *TCPNewReno*.
- **Configuration2:** All senders use *TCPNewReno*Plus.

For both two configurations above, the following hold:

i) Application data rate: 1.5 Mbps

ii) Connection 1 starts at time 1sec, Connection 2 starts at time 5sec, Connection 3 starts at time 15sec.

iii)Connection 1 ends at time 20sec, Connection 2 ends at time 25sec, Connection 3 ends at time 30sec.

iv) Packet size is 3000 bytes.
v) At N3 use RateErrorModel as the error model with error rate as **0.00001**.

You need to answer the following questions:
For each **configuration** do the following at **each TCP source(each connection).**

1. Plot Congestion window size vs time (from t=1 to t=30 seconds)
2. How does the congestion avoidance phase vary on the same sender when using *TCPNewReno*Plus vs *TCPNewReno*? Explain the observed trends. How does it impact the entire network?

What to Submit for Part A

1. For each part, write your answers, observations, plots and hypotheses in a single report file.
2. Submission folder should be a single .zip or .tar.gz file containing a report file and code for each question. Follow the directory structure as given below.

PART A/-
Entry_number_Report_partA.pdf

Congestion/-
TcpNewRenoPlus.cc
TcpNewRenoPlus.h
Congestion_topology_files/-
First.cc etc.

In case you are using any other helper code/libraries to generate plots, submit that code for generating plots.

3. The plots can be generated either using the .cc files or any other tool you wish to choose such as Python. But make sure, that plot is generated just using commands and not by copy pasting data values manually in excel or something like that during the demo. (**Tip** : Save the output values in a separate file and write a script to process that file and generate plots).
4. For each code file/package/libraries you submitted in the assignment, clearly describe their purpose in the report (1 to 2 sentences are enough for each).

Part B:
Network Layer

Routing Information Protocol

For this part we will refer to the following as base file
"ns-allinone-3.29/ns-3.29/examples/routing/rip-simple-network.cc"

You need to make the following changes in the base networking model.

- Create a network topology of the form

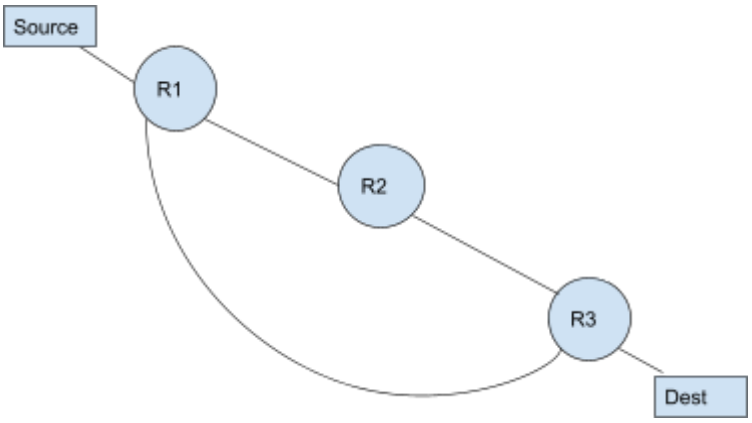

Fig.2

R1, R2, R3 are RIPng routers(similar to what used in the base file).

- All interfaces have metric =1(all networks have cost=1).
- Use split horizon as **SPLIT_HORIZON as** default.
- The **Source** and **Dest** will exchange packets as done in the base file.
- On similar lines as in the base file, for this topology(fig.2) R1 and R3(where R1 is connected to Source and R3 is connected to Dest), are configured with static addresses. Refer to base file and change if required.
- src and dst are using static routing in the base file and the messages are routed to A and D. In a similar way, src and dst should use static routing and the messages should be routed to R1 and R3 respectively. You shouldn't use static routing in R1 and R3.

Part B.i)

In this part we will study how the convergence time of routing table varies with different channel delays. The data rate is fixed for this experiment (5000000 i.e (csma.SetChannelAttribute ("DataRate", DataRateValue (5000000));).

Questions

a) Run the simulation with the following different delays in CSMA channel
[100, 1000, 20000, 80000]. All values in millisec
For each delay value, create a plot where x-axis represents time(in seconds) and y -coordinate shows whether all routing tables have converged or not at that point of time.(The y-coordinate will have value 1 if all routing tables have converged at that particular time else 0). For each plot, show the y-coordinate values at **Time stamps x = 0, 10, 50, 100, 500(all values in sec).**

For this question, you generate 4 plots(corresponding to each delay)

b) Explain the trend observed in part a). Why does this behavior happen?

For **PartB.ii** and **PartB.iii)** the data rate is fixed to 5000000(csma.SetChannelAttribute ("DataRate", DataRateValue (5000000));) and Delay is set to 2 Milliseconds.

Part B.ii): In this part we simulate link going down and its impact on routing tables and reachability.

Simulate the following:
The link between R1 and R2 goes down at t=50 sec. The link between R1 and R3 goes down at t=120 sec.

Questions

a) Print the routing table for all routers at t=121 sec and t=180 sec. For your tables, label which table is for which routing table(like R1, R2 R3). **No marks will be awarded without this.** First display the tables at t=121 sec then display the tables at t=180 sec.
b) Create a plot where on x-axis we have the time(t in sec) and on y-axis we plot whether the **destination(DEST) node** is reachable from the **source**

node(SOURCE). (y-coordinate has value 1 if dest at time t is reachable else 0). Do this for t = 49, 51, 60, 119, 121, 150(all values in sec).

c) In part b above, what is the similarity/difference between the reachability behavior at t=51sec vs at t=119sec vs at t=121sec? Explain why does it happen.

Part B.iii) Simulate the following(this question is independent of PartB.ii)

The link between R1 and R2 goes down at t=50sec. The link between R1 and R3 also goes down at t=50sec. The link between (R1 and R2) and between (R1 and R3) comes up at t= 120sec.

Questions

a) Print the routing table for all routers at t=70sec and t=180sec. For your routing tables, label which table is for which routing table(like R1, R2, R3). **No marks will be awarded without this.** First display the tables at t=70sec then display the tables at t=180sec.

b) Create a plot where on x-axis we have the time(t in sec) and on y-axis we plot whether the **destination node** is reachable from the **source node.** (y-coord has value 1 if dest is reachable from src else 0). Do this for t = 119, 121, 140, 200(All values in seconds.)

What to submit for Part B

Report(file name EntryNumber_Report_PartB) and your .cc file(s). Any other scripts such as .sh files required to call your program with arguments etc.

<u>Overall submission details for Assignment 4</u>

Create a folder(i.e home folder with name = Entry Number) having two sub folders (Part A, Part B).

Submit files(folders) of each part in its folder.

Zip home folder as EntryNumber.zip.