README
GO-BACK-N
(EE20BTECH11015-DONTHULA SRINISH)

Go-Back-N:

The codes for this protocol are in folder named “go-back-n”.There are 2 files named “EE20BTECH11015_senderGBN.py”(sends image) and “EE20BTECH11015_receiverGBN.py”(receives image).
Before running the code, first we should run “sudo mn” commands in the terminal and create 2 hosts named h1 and h2 by running “xterm h1” and “xterm h2” respectively.
The commands to add/change the link speed, delay and packet loss are written in “h1-arp.sh” and “h2-arp.sh”(“sudo tc qdisc add dev h1-eth0 root netem rate 10Mbit limit 100 delay 5ms loss 5%” in h1-arp.sh and similarly in h2-arp.sh”).
To implement the link speed, delay and packet loss, we should run the bash commands (“bash h1-arp.sh” in xterm h1 and similarly in xterm h2)
We can check the packet loss and delay by running the command “h1 ping -c 10 h2”.
Now we can run the python codes.
I am sending a image from h1 to h2.I am breaking the image data into small chunks and sending a window size of chunks at a time by receiving the acknowledgements from the receiver.There is a socket timeout.If a packet takes more than timeout,then the sender will not receiver the acknowledgement.Then the sender will retransmit the packet.We can change the socket timeout in the “sender.py” and check for better efficiency(less number of retransmissions).
I am receiving the image in a file named “received.jpg”.
I have calculated the total time taken to send the image and throughput in the code itself by running the code for different window sizes(given in the table) and different delays(5ms, 50ms, 150 ms).
From the values in the table, I have plotted the graph (Throughput(KBps) vs window size).