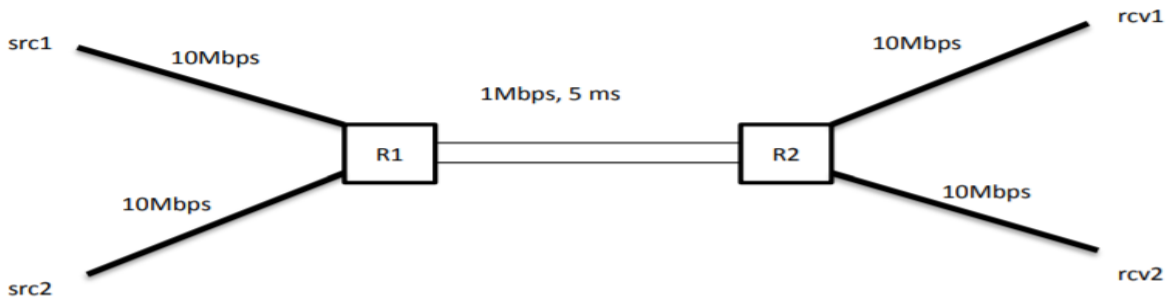
NS-1 NETWORK SIMULATION

TEAM 21: Akshay Gajanan Hasyagar, Karthikeyan Kathiresan

This is submitted as a part of ECEN 602 Network Simulation Assignment-1

The following setup configuration was done on NAM.
Setup configuration:

- The sources (src1 and src2), routers (R1 and R2), and receivers (rcv1 and rcv2) are set up using standard ns2 commands
- Duplex links are established with DropTail mechanism
- End to end delay between sources and routers varies according to the problem statement. End to end RTTs are increasing in the ratio 1:2 1:3 1:4
- Simulation was run for 400ms
- First 100ms was ignored and then throughput and relative throughput was tabulated
- Application sender is FTP over TCP



TCP version = (TCP SACK | TCP VEGAS)

Case 1:

•src1-R1 and R2-rcv1 end-2-end delay = 5 ms

•src2-R1 and R2-rcv2 end-2-end delay = 12.5 ms

Case 2:

•src1-R1 and R2-rcv1 end-2-end delay = 5 ms

•src2-R1 and R2-rcv2 end-2-end delay = 20 ms

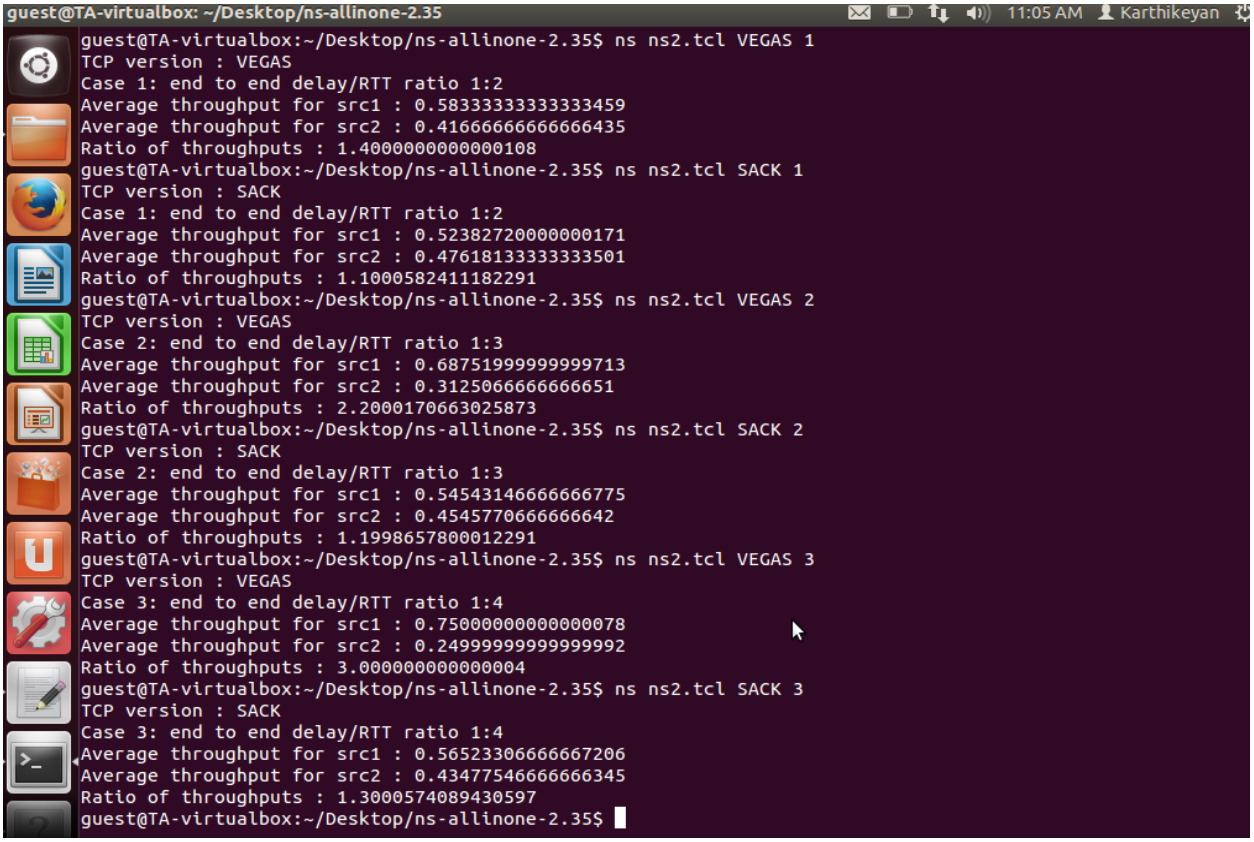
Case 3:

•src1-R1 and R2-rcv1 end-2-end delay = 5 ms

•src2-R1 and R2-rcv2 end-2-end delay = 27.5 ms

(i) For each of the TCP flavors (VEGAS and SACK) simulate the three RTT cases and find the ratio of the average throughput of src1 to src2. Make two separate tables (one for each TCP flavor) showing the throughput for each test case.

Output



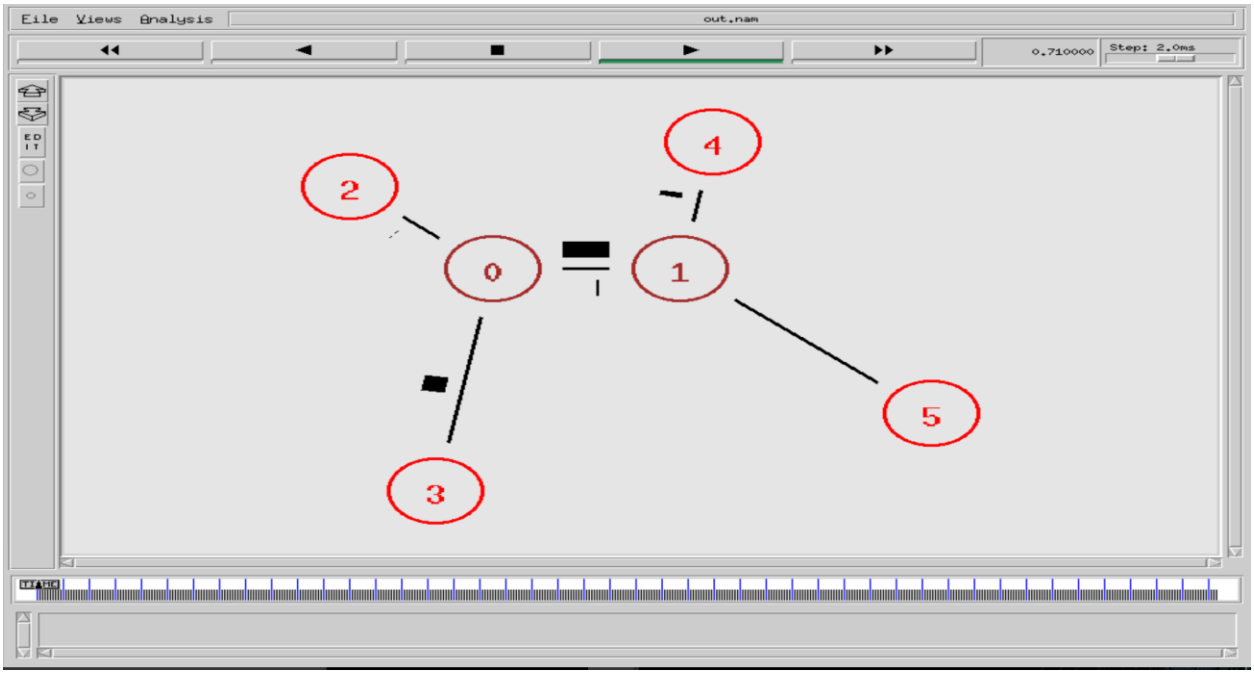


Table of values

VEGAS

Case Number	Average Throughput for SRC1 (in Mbps)	Average Throughput for SRC2 (in Mbps)	Ratio of throughputs
1	0.5833	0.4166	1.4
2	0.6875	0.3125	2.2
3	0.75	0.2499	3.0

SACK

Case Number	Average Throughput for SRC1 (in Mbps)	Average Throughput for SRC2 (in Mbps)	Ratio of throughputs
1	0.5238	0.4761	1.1
2	0.5454	0.4545	1.19
3	0.5652	0.4347	1.3

ii)

For each TCP Flavor, we can see the difference in throughputs increasing as the RTT difference increase (from 1:2 to 1:3 to 1:4). This is expected: as the SRC2's delay increases with every case (ACK delay also increases), SRC1 gets to transmit more packets in a particular time period as compared to SRC1 since its quickly acknowledged.

We can also see that the increase in the ratio of throughput in TCP VEGAS is higher compared to TCP SACK for the same increase in RTT ratio. Choosing case 1 as the reference, the relative ratio of throughput for TCP SACK is 1.10 and for VEGAS it is 1.4.

And in all cases, TCP Vegas outperforms SACK.

Why TCP Vegas performs better than SACK?

TCP Vegas outperform SACK with a better utilization of bandwidth and lesser congestion. In its estimation of incipient congestion, and its efficient estimation of congestion by measuring change in throughput rather than packet loss.

Also, Vegas is more stable than SACK. The reason for this being that SACK uses packet losses to denote congestion. So that the sender continually increases sending rate until there is congestion and then they cut back. This cycle continues, and the system keeps on oscillating. TCP Vegas flattens out its sending rate at the optimal bandwidth utilization point thus inducing stability.