Aim:

To Simulate Congestion Control using Network Simulator

Algorithm:

The size of the sender window is determined by the following two

factors

1. Receiver window size
2. Congestion window size

1.Reciever Window Size:

Sender should not send data greater than receiver window size. • Otherwise, it leads to dropping the TCP segments which causes TCP Retransmission. • So, sender should always send data less than or equal to receiver window size. • Receiver dictates its window size to the sender through TCP Header

2.Congestion Window Size:

Sender should not send data greater than congestion window size. • Otherwise, it leads to dropping the TCP segments which causes TCP retransmission. • So, sender should always send data less than or equal to congestion window size. • Different variants of TCP use different approaches to calculate the size of congestion window. • Congestion window is known only to the sender and is not sent over the links.

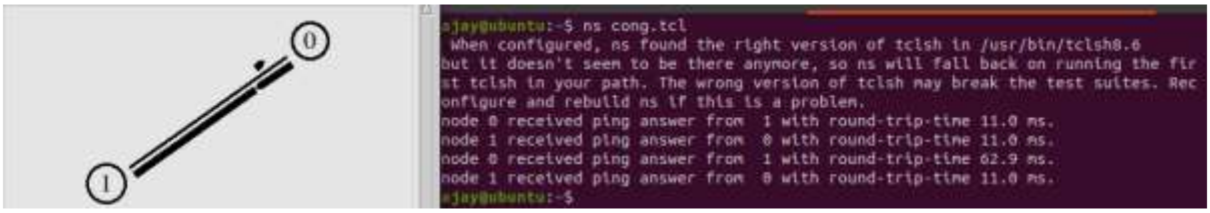
Code:

```
        set ns [new Simulator]
set f [open congestion.tr w]
$ns trace-all $f
set nf [open congestion.nam w]
$ns namtrace-all $nf
proc finish {} {
 exec nam congestion.nam &
exit 0
}
set n0 [$ns node]
set n1 [$ns node]
```

```
$ns duplex-link $n1 $n0 1Mb 5ms DropTail


set tcp1 [new Agent/TCP/Reno]

$ns attach-agent $n0 $tcp1

$tcp1 set fid_ 1



set sink1 [new Agent/TCPSink]

$ns attach-agent $n1 $sink1


$ns connect $tcp1 $sink1

set ftp1 [new Application/FTP]

$ftp1 attach-agent $tcp1

$ftp1 set type_ FTP


set p0 [new Agent/Ping]

$ns attach-agent $n0 $p0

set p1 [new Agent/Ping]

$ns attach-agent $n1 $p1

$ns connect $p0 $p1

Agent/Ping instproc recv {from rtt} {

$self instvar node_

puts "node [$node_ id] received ping answer from \

$from with round-trip-time $rtt ms."

}

$ns at 0.5 "$p0 send"

$ns at 0.8 "$p1 send"

$ns at 1.0   "$ftp1 start"

$ns at 70.0 "$ftp1 stop"

$ns at 70.1 "$p0 send"

$ns at 70.2 "$p1 send"
```

```
$ns at 80.0 "finish"



$ns run
```

Output:



```
ajay@ubuntu:~$ ns cong.tcl
 When configured, ns found the right version of tclsh in /usr/bin/tclsh8.6
but it doesn't seem to be there anymore, so ns will fall back on running the fir
st tclsh in your path. The wrong version of tclsh may break the test suites. Rec
onfigure and rebuild ns if this is a problem.
node 0 received ping answer from  1 with round-trip-time 11.0 ms.
node 1 received ping answer from  0 with round-trip-time 11.0 ms.
node 0 received ping answer from  1 with round-trip-time 62.9 ms.
node 1 received ping answer from  0 with round-trip-time 11.0 ms.
ajay@ubuntu:~$
```

Result :

Thus The TCP Congestion was simulated using ns2