Prajwal Sundar
106121092

CSLR52 NETWORKS LABORATORY-6

Simulate TCP Reno, Tahoe and Vegas for hight, medium and low traffics with differing number of nodes (10,20,30,40,50) and plot the following metrics for the differing traffics:

1. Control Overhead
2. Packet delivery ratio (PDR)
3. Retransmission rate (RT)

A) Reno:

1. Low Traffic

```
set ns [new Simulator]
set numOfNodes 50
set nSource 3

set nf [open tcp.nam w]
set tr [open reno_low_40.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15
```

```
# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Reno]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 40.1 "$ftp($i) stop"
}

$ns at 40.11 "finish"
$ns at 40.11 "finish"
$ns run

2.  Medium Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 7

set nf [open tcp.nam w]
set tr [open reno_mid_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
```

```
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Reno]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"
```

```
$ns run

3.  High Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 13

set nf [open tcp.nam w]
set tr [open reno_high_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15



# creating the source and sinks

for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Reno]
```

```
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"

$ns run

B)  Tahoe
1.  Low Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 3

set nf [open tcp.nam w]
set tr [open tahoe_low_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}
```

```
for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"
$ns run
2.  Medium Traffic

set ns [new Simulator]
```

```
set numOfNodes 50
set nSource 7

set nf [open tcp.nam w]
set tr [open tahoe_mid_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
```

```
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"
$ns run

3.  High Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 13

set nf [open tcp.nam w]
set tr [open tahoe_high_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
```

```
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"
$ns run

C)  Vegas
1.  Low Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 3

set nf [open tcp.nam w]
set tr [open vegas_low_50.tr w]
$ns trace-all $tr
$ns namtrace-all $nf
```

```
for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Vegas]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
```

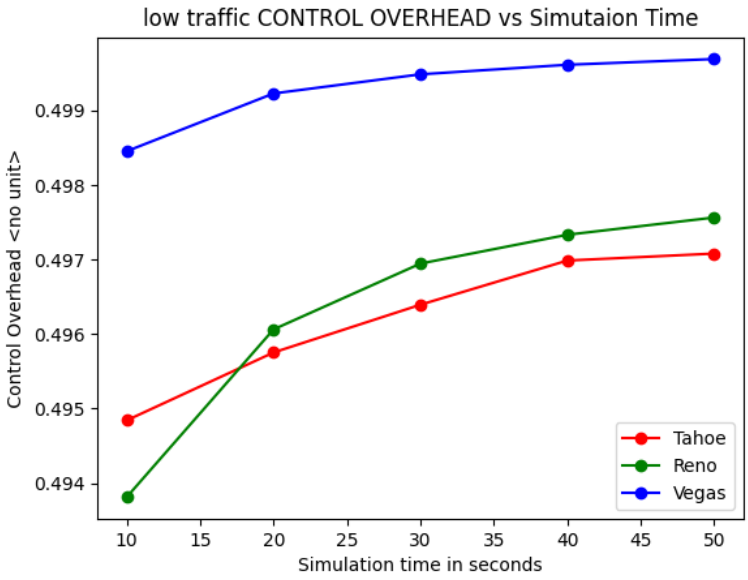
```
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 50.1 "$ftp($i) stop"
}

$ns at 50.11 "finish"
$ns at 50.11 "finish"
$ns run

2.  Medium Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 7

set nf [open tcp.nam w]
set tr [open vegas_mid_10.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
```

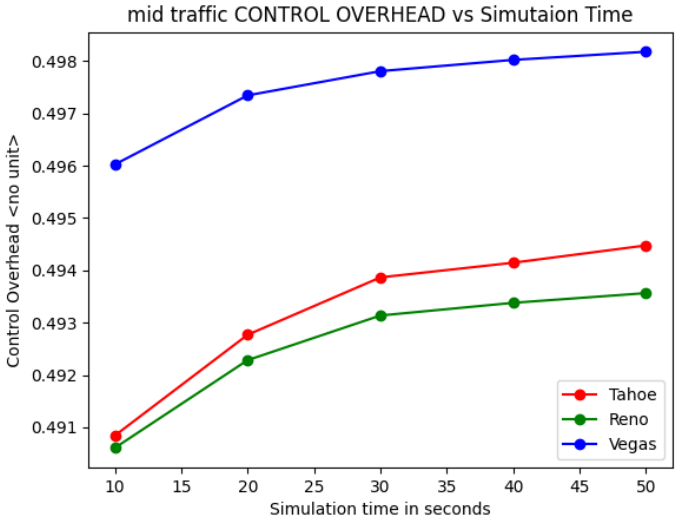
```
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Vegas]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 10.1 "$ftp($i) stop"
}

$ns at 10.11 "finish"
$ns at 10.11 "finish"
$ns run

3.  High Traffic

set ns [new Simulator]
set numOfNodes 50
set nSource 13

set nf [open tcp.nam w]
set tr [open vegas_high_10.tr w]
$ns trace-all $tr
$ns namtrace-all $nf

for { set i 0 } { $i < $numOfNodes } { incr i} {
    set ftp($i) [new Application/FTP]
}

proc finish {} {
    global ns nf
    $ns flush-trace
```

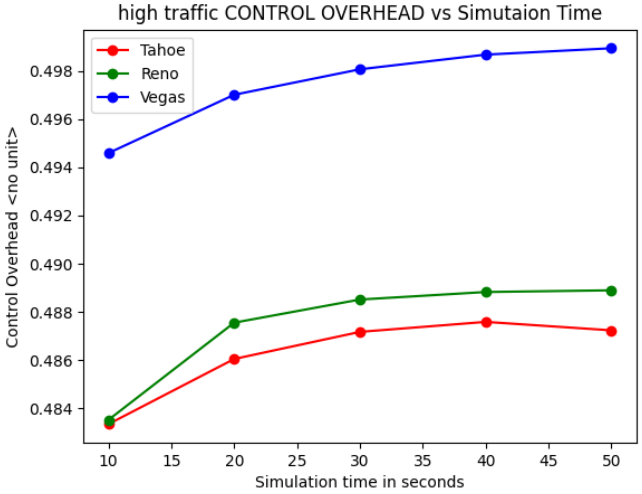
```
    close $nf
    exec nam tcp.nam &
    exit 0
}

for {set i 0} { $i < $numOfNodes } { incr i } {
    set n($i) [$ns node]
}

# Creating Star toplogy with 25 nodes in each star and a bus connection
between the stars
for { set i 1 } { $i < 25 } { incr i } {
    $ns duplex-link $n(0) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(0) $n($i) 20
}
for { set i 26 } { $i < 50 } { incr i } {
    $ns duplex-link $n(25) $n($i) 1Mb 10ms DropTail
    $ns queue-limit $n(25) $n($i) 10
}
$ns duplex-link $n(0) $n(25) 1Mb 10ms DropTail
$ns queue-limit $n(0) $n(25) 15

# creating the source and sinks
for { set i 1 } { $i <= $nSource } { incr i } {
    set source_node [expr $i * 3]
    set tcp($source_node) [new Agent/TCP/Vegas]
    $tcp($source_node) set class_ 2
    $ns attach-agent $n($source_node) $tcp($source_node)
    $ftp($i) attach-agent $tcp($source_node)
    $ftp($i) set type_ FTP

    set sink_node [expr 50 - $source_node]
    set sink($sink_node) [new Agent/TCPSink]
    $ns attach-agent $n($sink_node) $sink($sink_node)

    $ns connect $tcp($source_node) $sink($sink_node)
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at [ expr 0.01  * $i ] "$ftp($i) start"
}

for { set i 1} { $i <= $nSource } { incr i } {
    $ns at 10.1 "$ftp($i) stop"
}
$ns at 10.11 "finish"
$ns at 10.11 "finish"
$ns run
```

OUTPUT:

1. Control Overhead

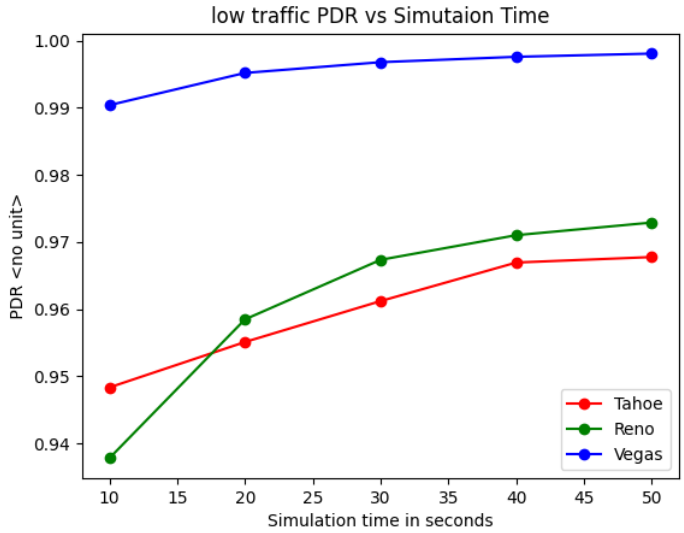


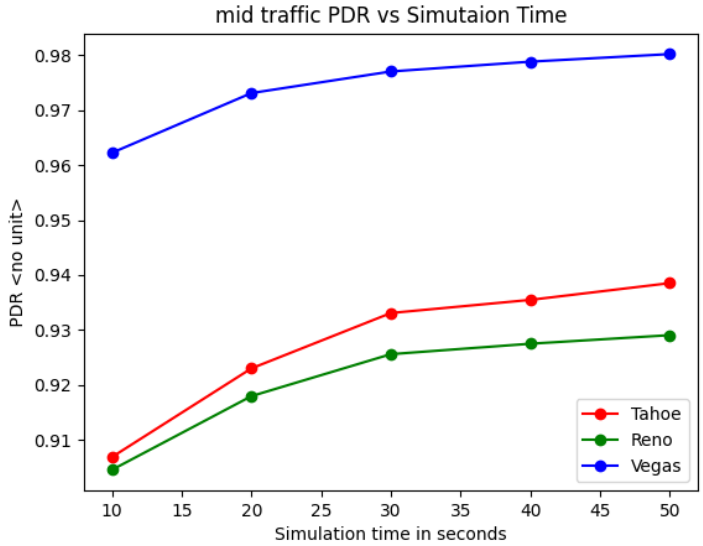


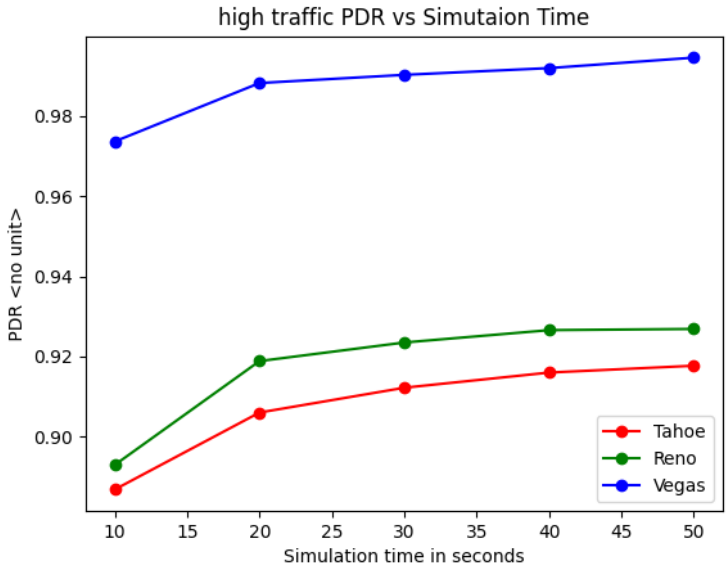


---

2. Packet Delivery Ratio (PDR)

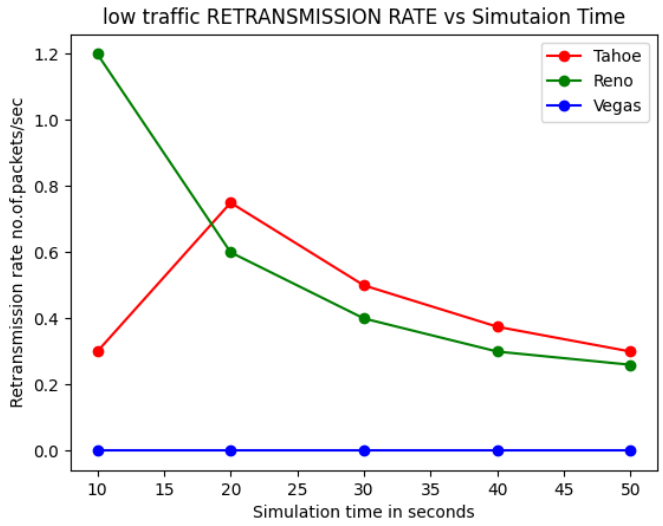


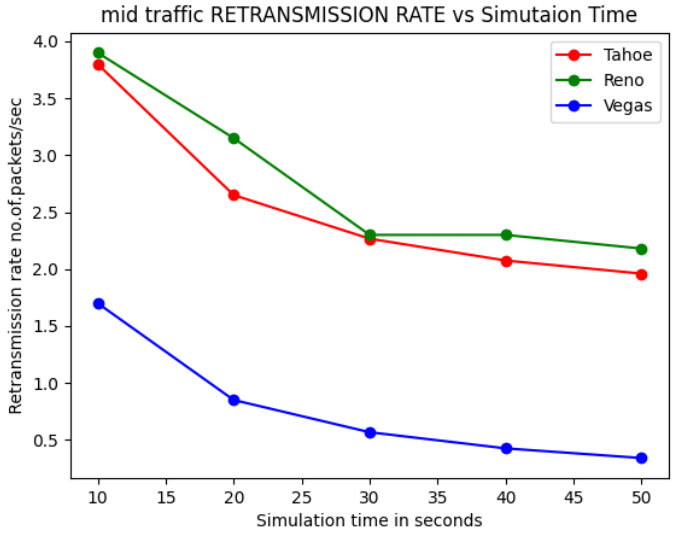


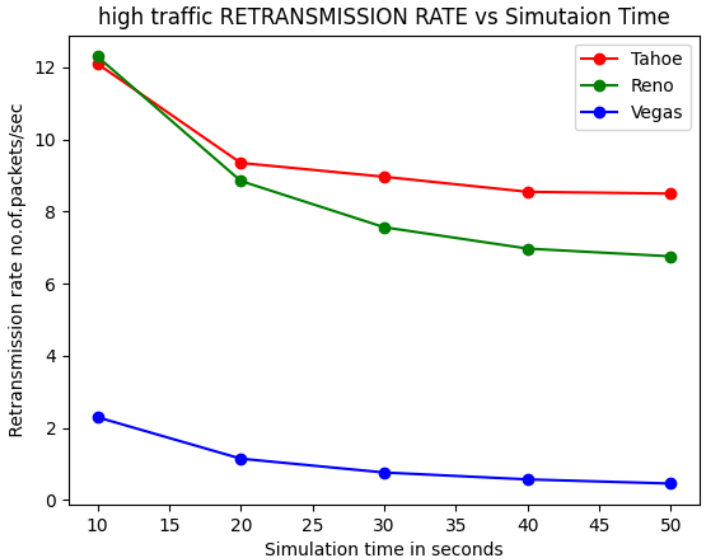


---

3. Retransmission Rate







Prajwal Sundar
106121092

PYTHON PLOT FILE

```
import matplotlib.pyplot as plt
paths = ["./Tahoe", "./Reno", "./Vegas"]
traffics = ["low" , "mid", "high"]
simulation_times = [10, 20, 30, 40, 50]
nodes = [ [[3, 6, 9], [47, 44, 41]],
          [[3, 6, 9, 12, 15, 18, 21], [47, 44, 41, 38, 35, 32, 29]],
          [[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39], [47, 44, 41, 38,
35, 32, 29, 26, 23, 20, 17, 14, 11]]
        ]

def readFile(filePath, Nodes) :
    sent = 0
    recv = 0
    packets = [0, 0] # ack_packets, total_packets
    retransmission = 0
    dropPackets = {}
    f = open(filePath, "r")
    for line in f:
        col = line.split()
        sender = eval(col[2])
        reciever = eval(col[3])
        packetID = col[2] + "-" + col[3] + "-" + col[10]
        if col[0] == "-":
            packets[1] += 1
            if col[4] == "ack" :
                packets[0] += 1
            if sender in Nodes[0]:
                sent += 1
        elif col[0] == "r" and reciever in Nodes[1]:
            recv += 1
        elif col[0] == "d":
            if packetID in dropPackets:
                retransmission += 1
            else:
                dropPackets[packetID] = True

    PDR = recv/sent
    control_overhead = packets[0]/packets[1]
    return PDR, control_overhead, retransmission

PDRS = []
CONT_OVRHD = []
RETRANS = []
for j in range(0,3): # traffic types
    TrafficPDR = []
    TrafficCO = []
    TrafficRT = []
```

```
    for i in range(0,3): # agent type
        AgentPDR = []
        AgentCO = []
        AgentRT = []
        for k in range(0, 5): #simulation time
            agent = paths[i].split('/')
            agent = agent[1].lower()
            filePath = paths[i] + "/" + traffics[j] + "_tr/" + agent + "_" +
traffics[j] + "_" + str(simulation_times[k]) + ".tr"
            PDR, CO, RT = readFile(filePath, nodes[j]) #PDR, CONTROL OVERHARD,
RETRANSMISSION
            RT /= simulation_times[k] # RETRANSMISSION RATE
            AgentPDR.append(PDR)
            AgentCO.append(CO)
            AgentRT.append(RT)
        TrafficPDR.append(AgentPDR)
        TrafficCO.append(AgentCO)
        TrafficRT.append(AgentRT)
    PDRS.append(TrafficPDR)
    CONT_OVRHD.append(TrafficCO)
    RETRANS.append(TrafficRT)
for i in range(0,3):
    print(traffics[i] + " traffic Values \n")
    for j in range(0,3):
        agentname = paths[j].split('/')
        agentname = agentname[1]
        print("  ", agentname)
        for k in range(0,5):
            print("\tSimulation Time ", simulation_times[k])
            print("\t  PDR = ", PDRS[i][j][k], "CTRL_OVRHD = ",
CONT_OVRHD[i][j][k], " R.R = ", RETRANS[i][j][k])
```