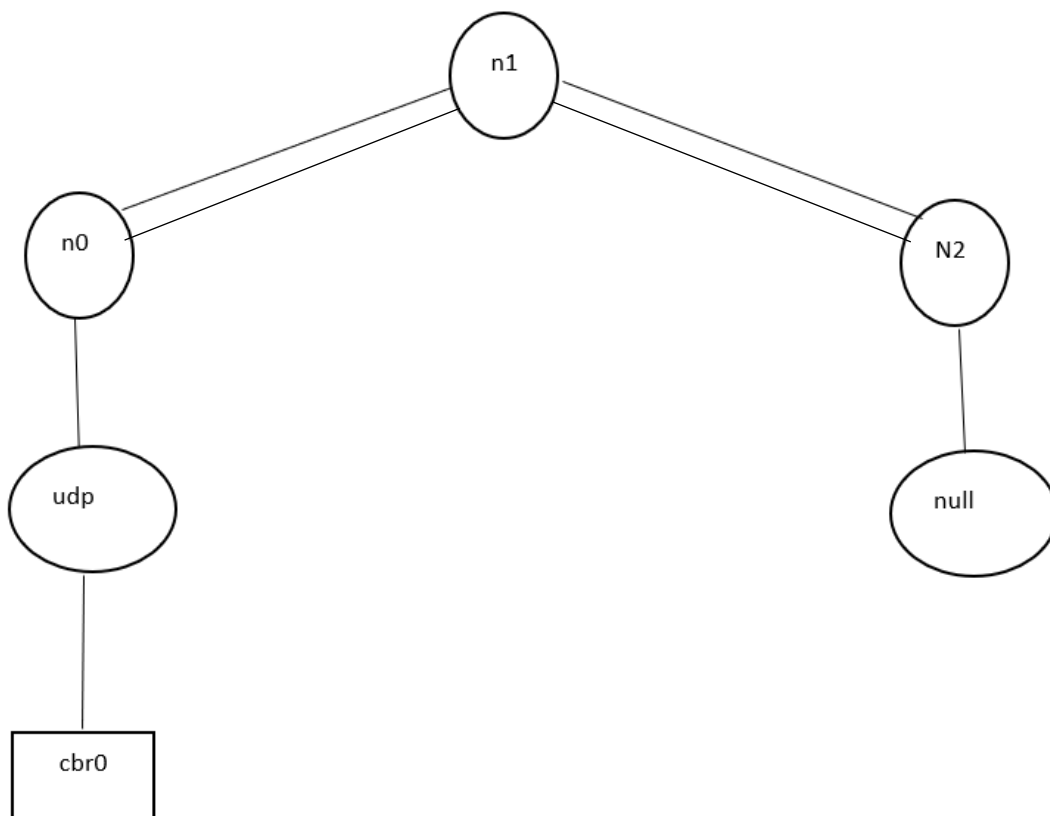


Experiment No 1

Problem statement

Simulate a three nodes point-to-point network with duplex links between them. Set the queue size vary the bandwidth and find the number of packets dropped

Topology:



Program:

```
#create simulator
set ns [new Simulator]

#open Trace file and NAM file
set ntrace [open prog1.tr w]
$ns trace-all $ntrace
set namfile [open prog1.nam w]
$ns namtrace-all $namfile

#create 3 nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]

#create duplex link b/w the nodes
#you need to modify the bandwidth to observe the variation in pkts drop
$ns duplex-link $n0 $n1 100Mb 10ms DropTail
$ns duplex-link $n1 $n2 100Mb 10ms DropTail

#set queue size
#you can modify the queue length as well to observe variation in packet drops
$ns queue-limit $n0 $n1 10
$ns queue-limit $n1 $n2 10

#set up a transport layer connection
set udp [new Agent/UDP]
$ns attach-agent $n0 $udp
set null [ new Agent/Null]
$ns attach-agent $n2 $null
$ns connect $udp $null

#set up an application layer traffic
set cbr0 [new Application/Traffic/CBR]
$cbr0 set type_ CBR
$cbr0 set packetSize_ 100
$cbr0 set rate_ 1Mb
$cbr0 set random_ false
$cbr0 attach-agent $udp
```

```
#finish procedure
proc finish {} {
global ns ntrace namfile

#dump all trace data and close the files
$ns flush-trace
close $ntrace
close $namfile

#execute the NAM Animation file
exec nam prog1.nam &

#show the number of pkts dropped
exec echo "the no of pkts dropped is" &
exec grep -c "^d" prog1.tr &
exit 0
}

#schedule events
$ns at 0.0 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"
#run the simulation
$ns run
```

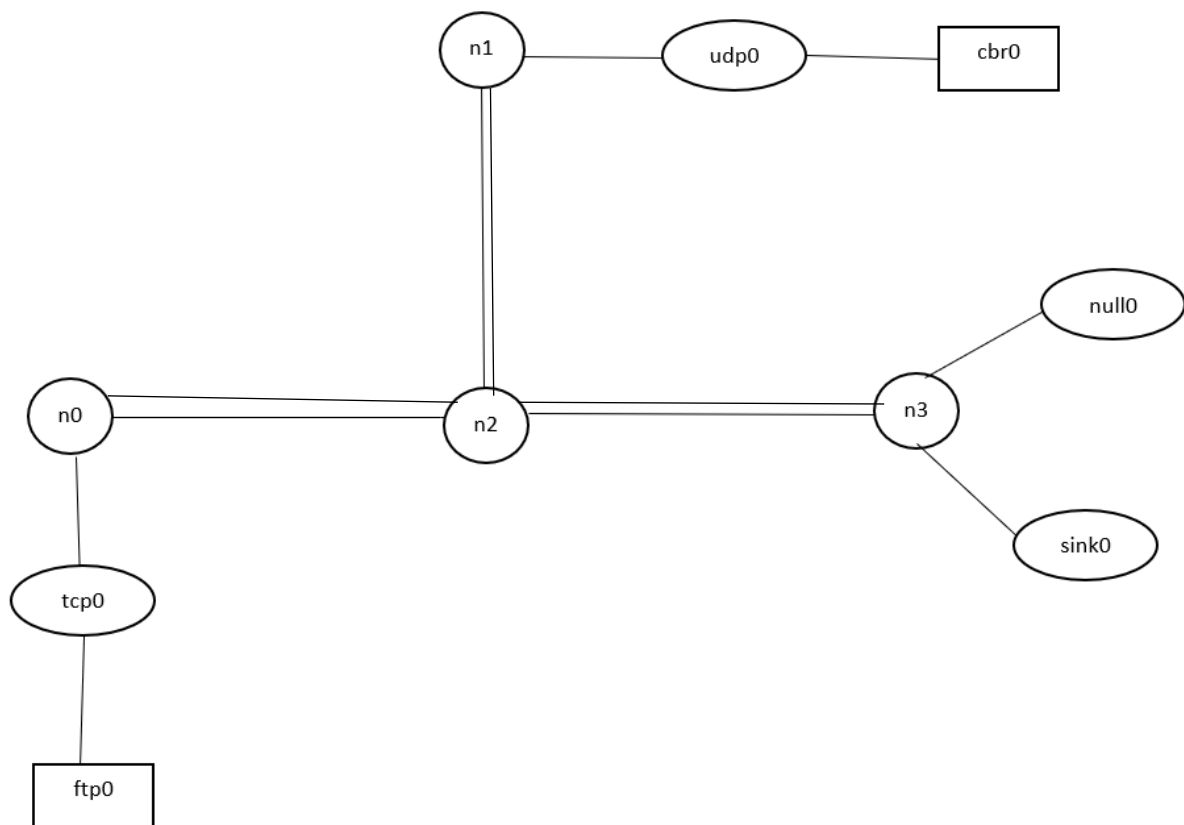
Output:

Experiment No 2

Problem statement

Simulate a four-node point-to-point network, and connect the links as follows: n0-n2, n1-n2 and n2-n3. Apply TCP agent between n0-n3 and UDP n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets sent by TCP/UDP.

Topology:



Program

```
set ns [new Simulator]
set ntrace [open prog2.tr w]
$ns trace-all $ntrace
set namfile [open prog2.nam w]
$ns namtrace-all $namfile
proc finish { } {
    global ns ntrace namfile
    $ns flush-trace
    close $ntrace
    close $namfile
    exec nam prog2.nam &
    exec echo "The no of TCP pkts sent are" &
    exec grep "^+" prog2.tr | cut -d " " -f 5 | grep -c "tcp" &
    exec echo "The no of UDP pkts sent are" &
    exec grep "^+" prog2.tr | cut -d " " -f 5 | grep -c "cbr" &
    exit 0
}
```

```
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$ns duplex-link $n0 $n2 2Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms DropTail
$ns duplex-link $n2 $n3 0.07Mb 20ms DropTail
```

```
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink0 [new Agent/TCPSink]
$ns attach-agent $n3 $sink0
$ns connect $tcp0 $sink0
```

```
set ftp0 [new Application/FTP]
$ftp0 set type_ FTP
$ftp0 attach-agent $tcp0
```

```
set udp0 [new Agent/UDP]
$ns attach-agent $n1 $udp0
set null0 [new Agent/Null]
$ns attach-agent $n3 $null0
$ns connect $udp0 $null0

set cbr0 [new Application/Traffic/CBR]
$cbr0 set type_ CBR
$cbr0 set packetSize_ 1000
$cbr0 set rate_ 0.01Mb
$cbr0 set random_ false
$cbr0 attach-agent $udp0

$ns at 0.1 "$cbr0 start"
$ns at 6.0 "$ftp0 start"
$ns at 5.0 "$cbr0 stop"
$ns at 10.0 "$ftp0 stop"
$ns at 11.0 "finish"
$ns run
```

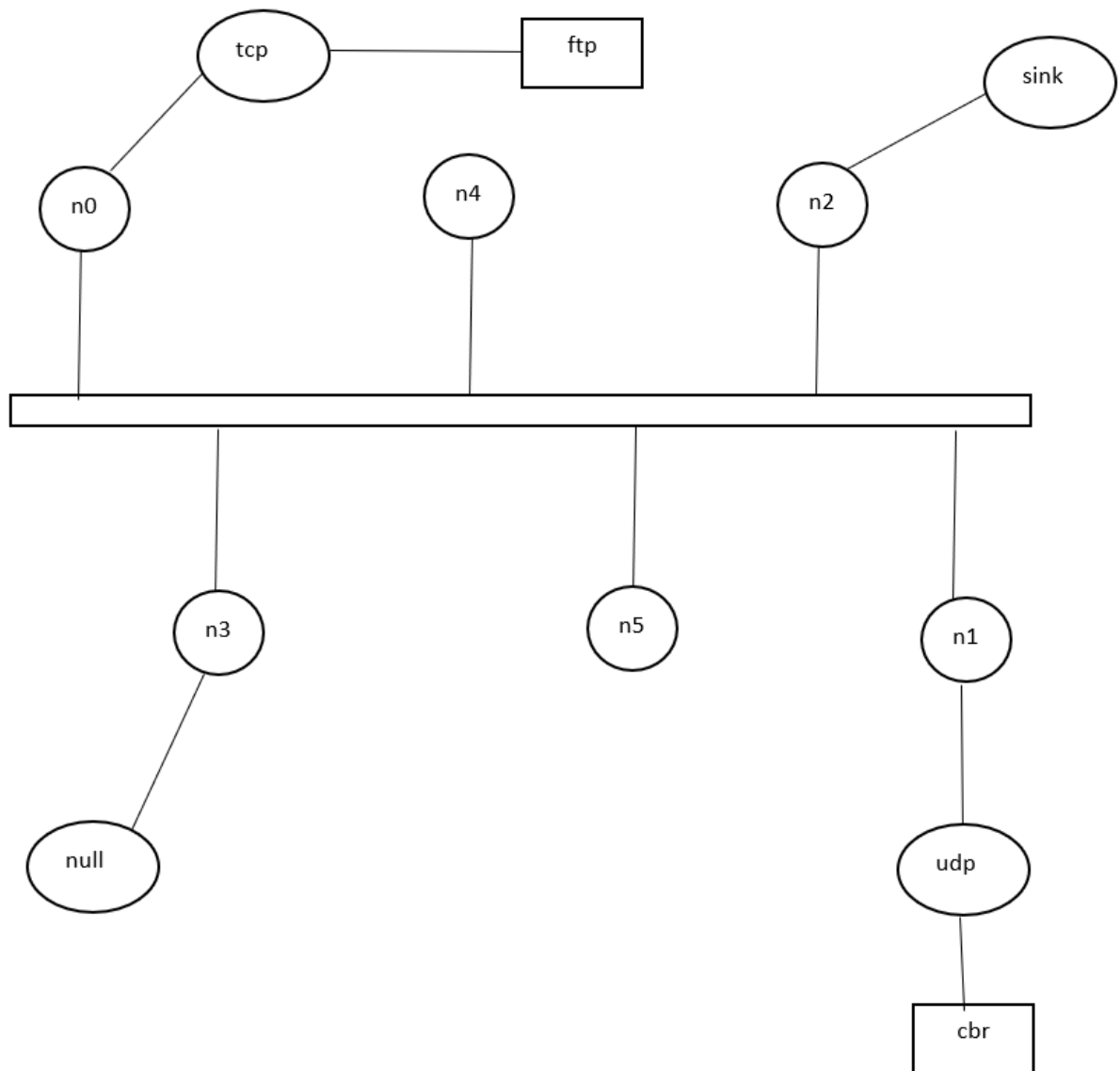
Output:

Experiment No 3

Problem statement

Simulate an Ethernet LAN using N nodes. Determine collision across different nodes

Topology



Program

```

set ns [new Simulator]
set ntrace [open prog3.tr w]
$ns trace-all $ntrace
set namfile [open prog3.nam w]
$ns namtrace-all $namfile
proc finish {} {
    global ns ntrace namfile
    $ns flush-trace
    close $ntrace
    close $namfile
    exec nam prog3.nam &
    exec echo "the no of tcp pkts dropped due to collision is" &
    exec grep -c "^d" prog3.tr &
    exit 0
}
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set lan [$ns newLan "$n0 $n1 $n2 $n3 $n4 $n5" 5Mb 10ms LL Queue/DropTail channel]

set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink]
$ns attach-agent $n2 $sink
$ns connect $tcp $sink

set ftp [new Application/FTP]
$ftp attach-agent $tcp

set udp [new Agent/UDP]
$ns attach-agent $n1 $udp
set null [new Agent/Null]
$ns attach-agent $n3 $null
$ns connect $udp $null

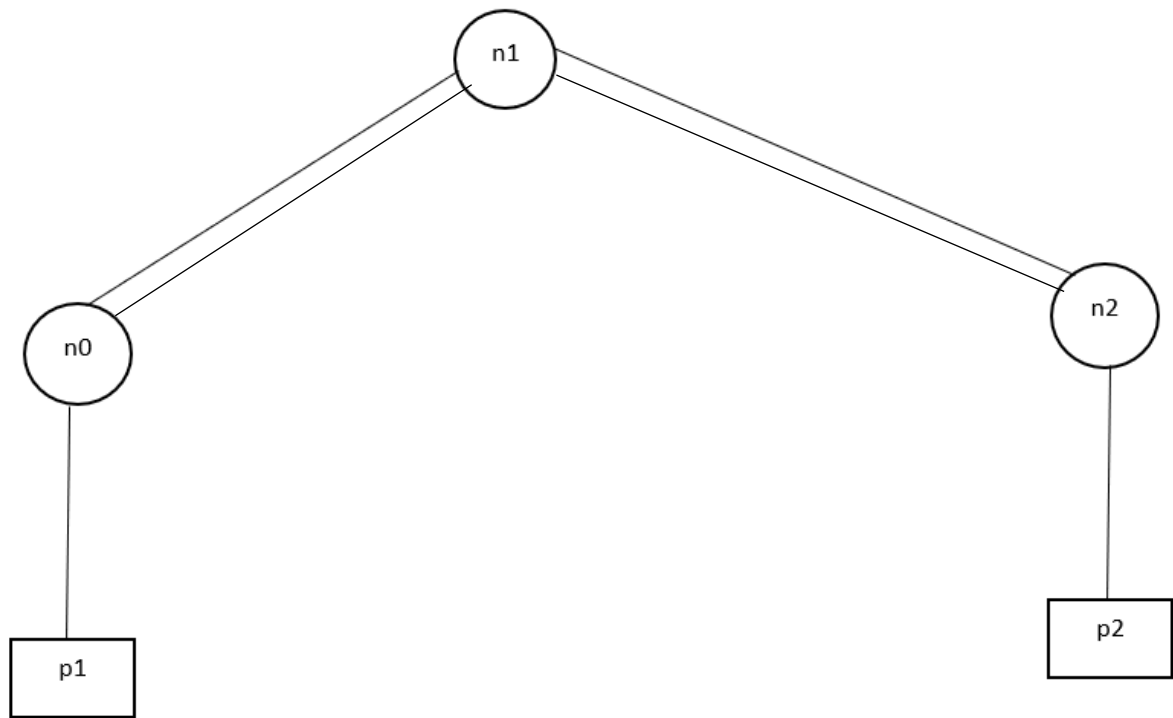
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp

$ns at 0.1 "$cbr start"
$ns at 2.0 "$ftp start"
$ns at 1.9 "$cbr stop"
$ns at 4.3 "$ftp stop"
$ns at 6.0 "finish"
$ns run

```


Experiment No 4

To Simulate transmission of ping messages over a network topology using ns-2



```

set ns [new Simulator]
set ntrace [open prg4.tr w]
$ns trace-all $ntrace
set namfile [open prg4.nam w]
$ns namtrace-all $namfile

proc finish {} {
global ns ntrace namfile
$ns flush-trace
close $ntrace
close $namfile
exec nam prg4.nam &
exit 0
}

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]

$ns duplex-link $n0 $n1 1Mb 10ms DropTail
$ns duplex-link $n1 $n2 1Mb 10ms DropTail

Agent/Ping instproc recv {from rtt} {
$self instvar node_
puts "$from recieved ping answer from node [$node_ id] with round trip time $rtt ms"
}

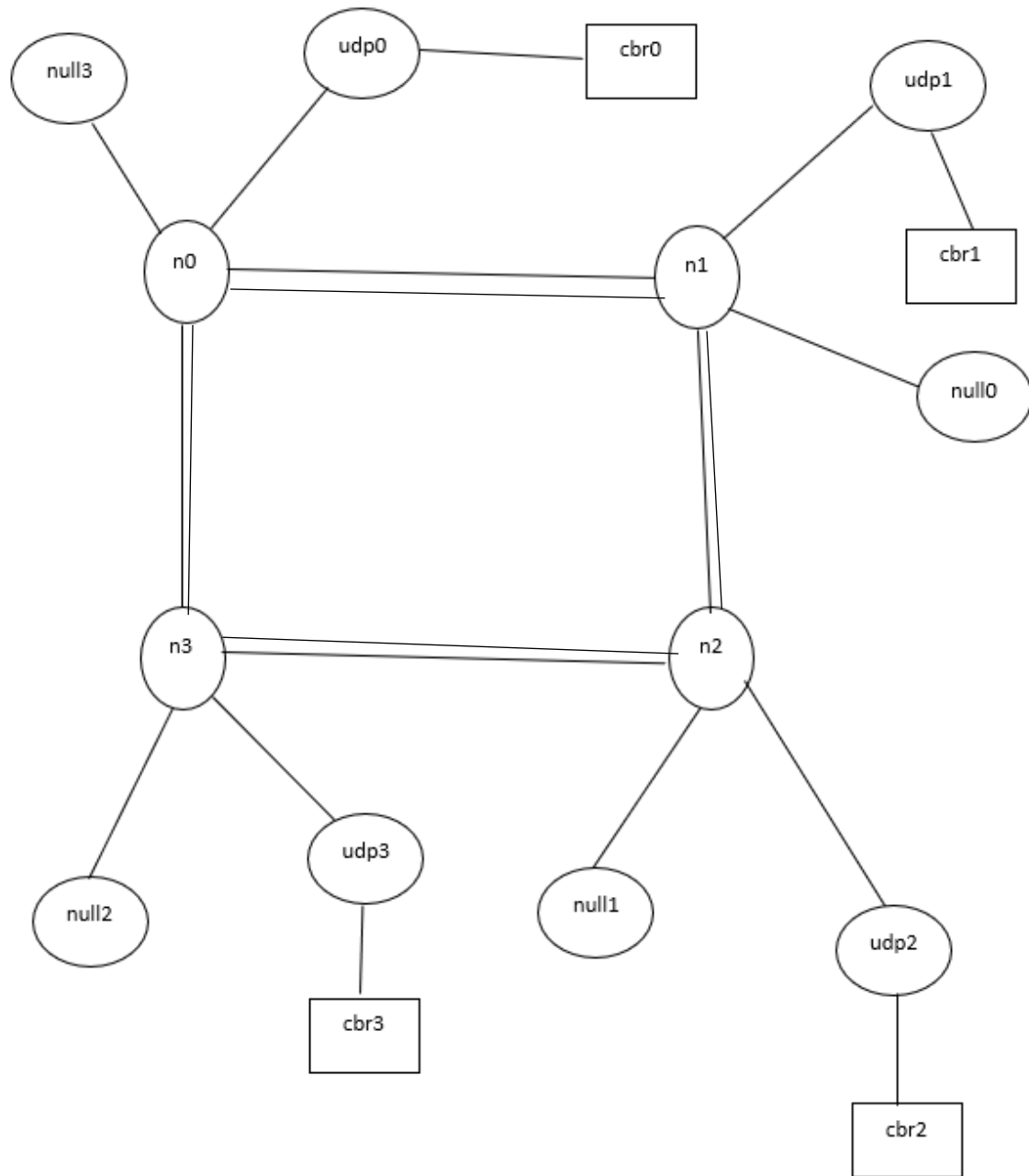
set p0 [new Agent/Ping]
$ns attach-agent $n0 $p0
set p1 [new Agent/Ping]
$ns attach-agent $n2 $p1
$ns connect $p0 $p1
$ns at 0.2 "$p0 send"
$ns at 0.4 "$p1 send"
$ns at 1.2 "$p0 send"
$ns at 1.7 "$p1 send"
$ns at 1.8 "finish"
$ns run

```

Experiment No 5

Implement the method of cyclic data transmission using UDP protocol

Topology:



```

set ns [new Simulator]
set ntrace [open prog6.tr w]
$ns trace-all $ntrace
set namfile [open prog6.nam w]
$ns namtrace-all $namfile
proc finish {} {
global ns ntrace namfile
$ns flush-trace
close $ntrace
close $namfile
exec nam prog6.nam &
exit 0
}
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$ns duplex-link $n0 $n1 2Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms DropTail
$ns duplex-link $n2 $n3 2Mb 10ms DropTail
$ns duplex-link $n3 $n0 2Mb 10ms DropTail

set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0
$ns connect $udp0 $null0
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0

set udp1 [new Agent/UDP]
$ns attach-agent $n1 $udp1
set null1 [new Agent/Null]
$ns attach-agent $n2 $null1
$ns connect $udp1 $null1
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $udp1

```

```
set udp2 [new Agent/UDP]
$ns attach-agent $n2 $udp2
set null2 [new Agent/Null]
$ns attach-agent $n3 $null2
$ns connect $udp2 $null2
set cbr2 [new Application/Traffic/CBR]
$cbr2 attach-agent $udp2
|
set udp3 [new Agent/UDP]
$ns attach-agent $n3 $udp3
set null3 [new Agent/Null]
$ns attach-agent $n0 $null3
$ns connect $udp3 $null3
set cbr3 [new Application/Traffic/CBR]
$cbr3 attach-agent $udp3

$ns at 0.0 "$cbr0 start"
$ns at 2.0 "$cbr1 start"
$ns at 3.0 "$cbr2 start"
$ns at 4.0 "$cbr3 start"

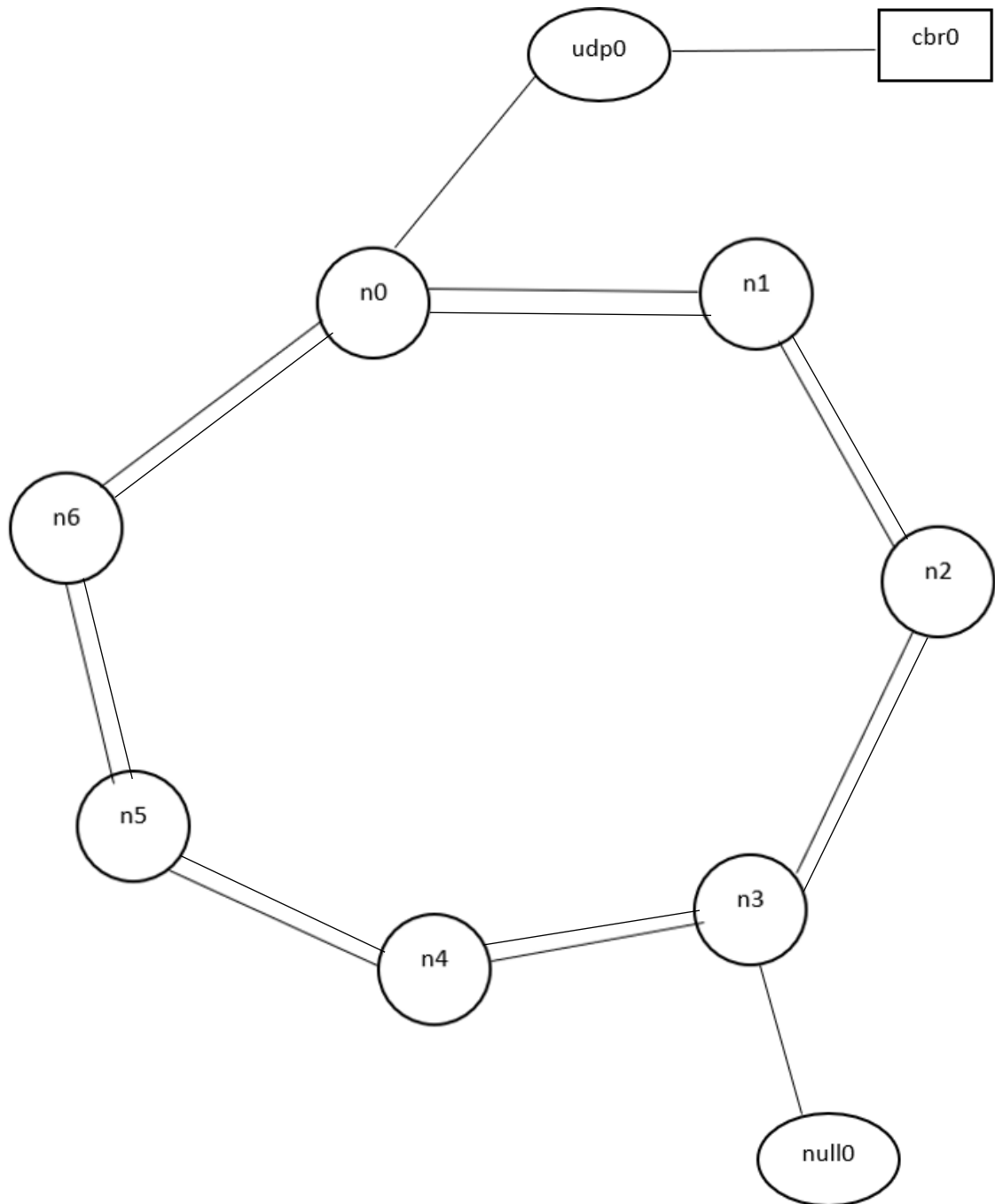
$ns at 2.0 "$cbr0 stop"
$ns at 3.0 "$cbr1 stop"
$ns at 4.0 "$cbr2 stop"
$ns at 5.0 "$cbr3 stop"

$ns at 10.0 "finish"
$ns run
```

Experiment No 6

Develop a program that implements dynamic routing algorithm using 7 nodes

Topology



```

set ns [new Simulator]
set ntrace [open prog7.tr w]
$ns trace-all $ntrace
set namfile [open prog7.nam w]
$ns namtrace-all $namfile
proc finish {} {
    global ns ntrace namfile
    $ns flush-trace
    close $ntrace
    close $namfile
    exec nam prog7.nam &
    exit 0
}

for {set i 0} {$i < 7} {incr i} {
    set n($i) [$ns node]
}

for {set i 0} {$i < 7} {incr i} {
    $ns duplex-link $n($i) $n([ expr ($i+1)%7 ]) 1Mb 10ms DropTail
}

$ns rtproto DV
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
$ns connect $udp0 $null0
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0

$ns at 0.0 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 3.0 "$cbr0 stop"

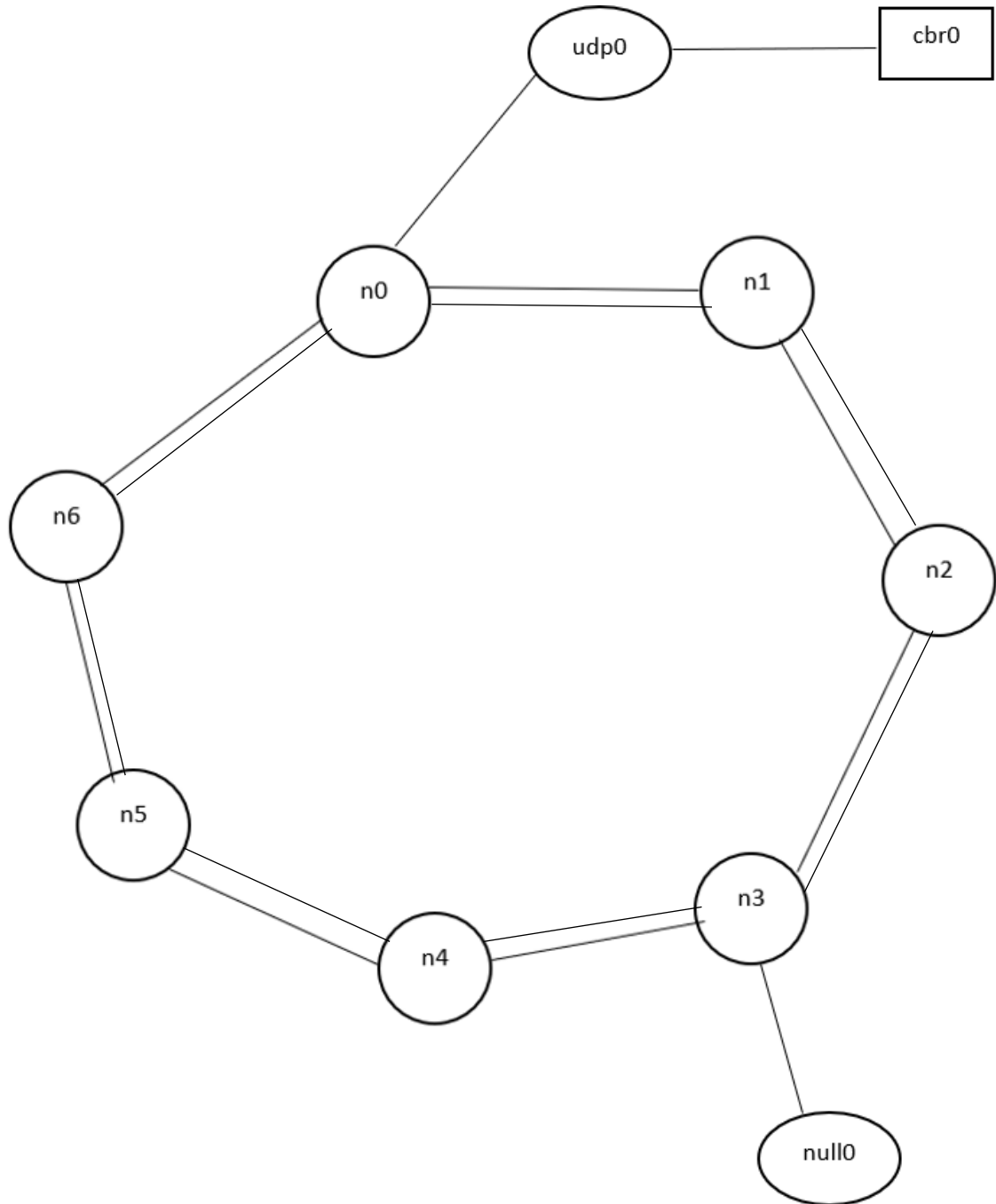
$ns at 5.0 "finish"
$ns run

```

Experiment No 7

Develop a program that implements link state routing algorithm using 7 nodes

Topology




```

set ns [new Simulator]
set ntrace [open prog7.tr w]
$ns trace-all $ntrace
set namfile [open prog7.nam w]
$ns namtrace-all $namfile
proc finish {} {
    global ns ntrace namfile
    $ns flush-trace
    close $ntrace
    close $namfile
    exec nam prog7.nam &
    exit 0
}

for {set i 0} {$i < 7} {incr i} {
    set n($i) [$ns node]
}

for {set i 0} {$i < 7} {incr i} {
    $ns duplex-link $n($i) $n([ expr ($i+1)%7 ]) 1Mb 10ms DropTail
}

$ns rtproto LS
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
$ns connect $udp0 $null0
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0

$ns at 0.0 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 3.0 "$cbr0 stop"

$ns at 5.0 "finish"
$ns run

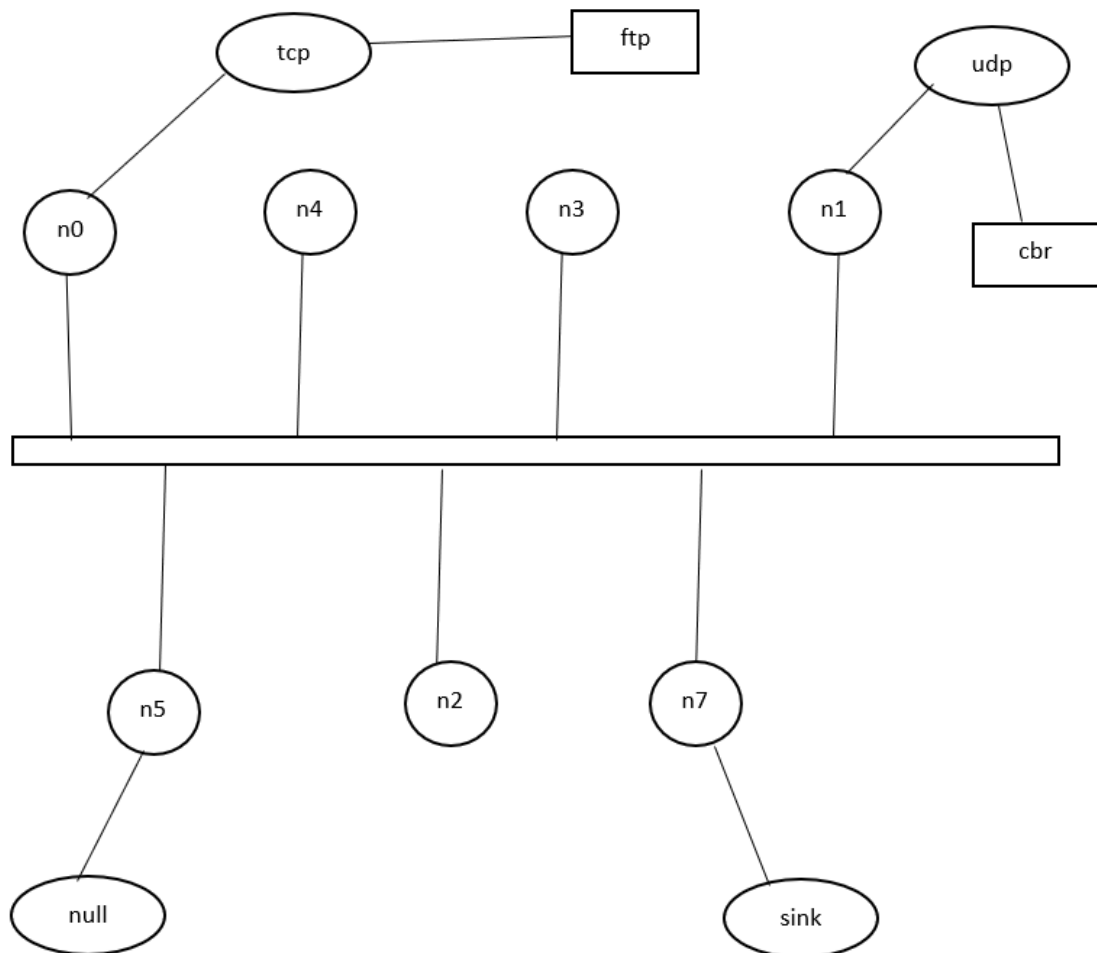
```

Experiment No 8

Problem Statement

Simulate an Ethernet LAN using N-nodes, and determine the throughput

Topology: -



```
set ns [new Simulator]
set trf [open prog5.tr w]
$ns trace-all $trf
set naf [open prog5.nam w]
$ns namtrace-all $naf
```

```
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
set n6 [$ns node]
set n7 [$ns node]
```

```
set lan [$ns newLan "$n0 $n1 $n2 $n3 $n4 $n5 $n6 $n7" 5Mb 10ms LL Queue/DropTail Channel]
```

```
set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
set ftp [new Application/FTP]
$ftp attach-agent $tcp
set sink [new Agent/TCPSink]
```

```
$ns attach-agent $n7 $sink
$ns connect $tcp $sink
```

```
set udp [new Agent/UDP]
$ns attach-agent $n1 $udp
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
set null [new Agent/Null]
$ns attach-agent $n5 $null
$ns connect $udp $null
```

```

proc finish {} {
    global ns naf trf
    $ns flush-trace
    exec nam prog5.nam &
    close $trf
    close $naf

    set tcpsize [ exec grep "^r" prog5.tr | grep "tcp" | tail -n 1 | cut -d " " -f 6]
    set numtcp [ exec grep "^r" prog5.tr | grep -c "tcp"]
    set tcptime 4.0
    set udpsize [ exec grep "^r" prog5.tr | grep "cbr" | tail -n1 | cut -d " " -f6]
    set numudp [ exec grep "^r" prog5.tr | grep -c "cbr"]
    set udptime 4.0

    puts "The throughput of FTP is"
    puts "[ expr ($numtcp*$tcpsize)/$tcptime] bytes per second"
    puts "The throughput of CBR is"
    puts "[ expr ($numudp*$udpsize)/$udptime] bytes per second"
    exit 0
}

$ns at 0.1 "$cbr start"
$ns at 2.0 "$ftp start"
$ns at 1.9 "$cbr stop"
$ns at 4.3 "$ftp stop"
$ns at 6.0 "finish"
$ns run

```