Προσομοίωση Δικτύων

2η Άσκηση

Δημιουργία, διαμόρφωση μελέτη επικοινωνιακών ζεύξεων

Δημιουργία ζεύξης

- Create a simulator object

  set ns [new Simulator]

- Open a file for writing that is going to be used for the nam trace data.

  set nf [open out.nam w]
  $ns namtrace-all $nf

- Open a file for writing that is going to be used for the event trace data.

  set tf [open out.tr w]
  $ns trace-all $tf

Δημιουργία ζεύξης

- Add a 'finish' procedure that closes the trace file and starts nam.

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

- Execute the 'finish' procedure after 5.0 seconds of simulation time.
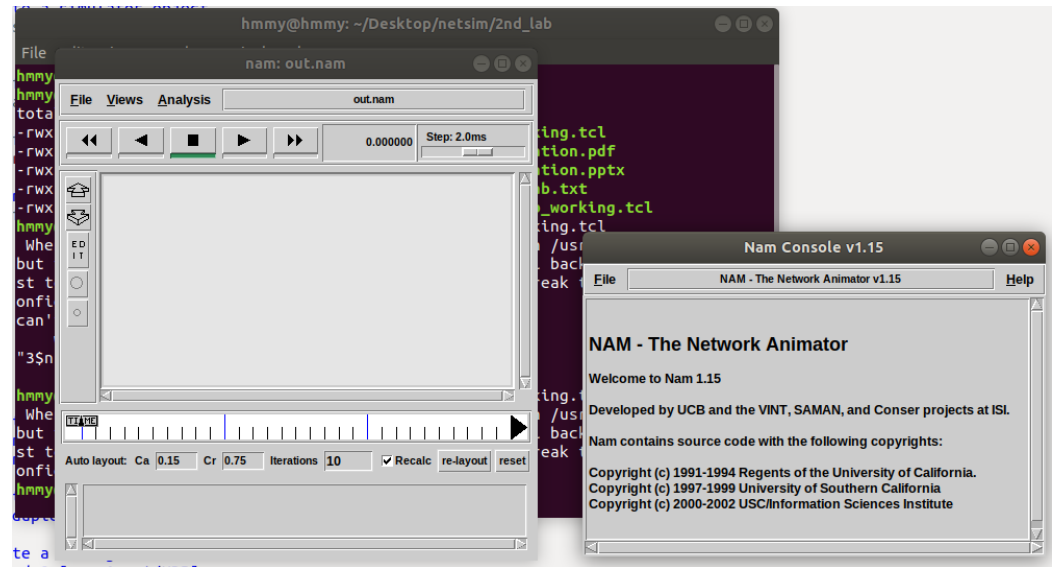
```
        $ns at 5.0 "finish"
```

- Starts the simulation.

```
        $ns run
```

Δημιουργία ζεύξης

- Run the simulation
  - If everything goes well, you should get
    - No error messages
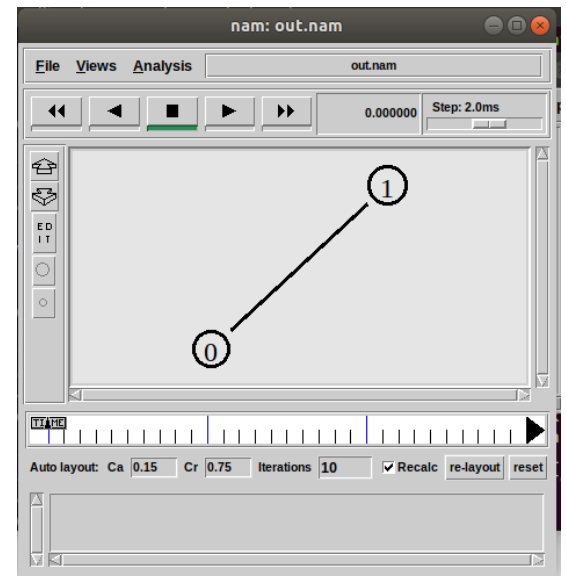   - And something
    like the following

- So, we have the simulation shell but nothing in it.
- Let's create a bidirectional link

Δημιουργία ζεύξης

- Create two nodes

  set n0 [$ns node]

  set n1 [$ns node]

- Create a bidirectional link between these nodes

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

- Run again the simulation

  – It should now look like this →

  – Now we have a network that does nothing

Δημιουργία ζεύξης

- Let put some traffic over the link
  - A CBR traffic source from n0

```
#Create a UDP agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0

# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.009
$cbr0 attach-agent $udp0
```

Δημιουργία ζεύξης

- Let put some traffic over the link
  - A traffic sink to n1
  set null0 [new Agent/Null]
  $ns attach-agent $n1 $null0

  - Connection at transport layer
  $ns connect $udp0 $null0

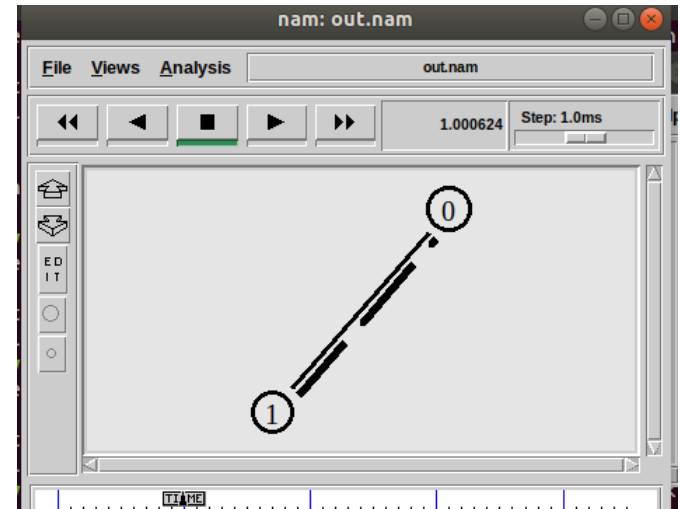 - Control the beginning and stop of the data flow
  $ns at 0.5 "$cbr0 start"
  $ns at 4.5 "$cbr0 stop"

- Re-run it and see the traffic at nam appllication
➢ Increase traffic rate until you see packet get dropped
  - Gradually increase packet creation rate
  - Gradually increase packet size

Performance Evaluation

- First make sure you have all the right files in the same folder

  - delay_2nd_lab.txt

```
hmmy@hmmy:~/Desktop/netsim/2nd_lab$ ls -l
total 792
-rwxrwxrwx 1 hmmy hmmy   1185 Οκτ  11  2016 2nd_lab_working.tcl
-rwxrwxrwx 1 hmmy hmmy 146391 Οκτ  23  2015 2nd_presentation.pdf
-rwxrwxrwx 1 hmmy hmmy 246056 Οκτ  17  2014 2nd_presentation.pptx
-rwxrwxrwx 1 hmmy hmmy    825 Οκτ  27  2014 delay_2nd_lab.txt
-rw-r--r-- 1 hmmy hmmy 289702 Νοε   5 15:37 out.nam
-rw-r--r-- 1 hmmy hmmy 108978 Νοε   5 15:37 out.tr
```

  - out.tr

  - out.nam

- 1st way: using the awk script "delay_2nd_lab.tx

  - Execute the following command:

    - awk –f delay_2nd_lab.txt out.tr

    - This command applies "delay_2nd_lab.txt" upon the "out.tr" trace file and should give something like

```
hmmy@hmmy:~/Desktop/netsim/2nd_lab$ awk -f delay_2nd_lab.txt out.tr
Mean Delay = 0.014000
Successfully transmitted packets = 801.000000
Dropped packets = 0.000000
```

Performance Evaluation

- 2nd way: Reading the out.tr file
  - You can open it with an editor and read it based on the following guideline

event	time	from node	to node	pkt type	pkt size	flags	fid	src addr	dst addr	seq num	pkt id

```
r : receive (at to_node)
+ : enqueue (at queue)              src_addr : node.port (3.0)
- : dequeue (at queue)              dst_addr : node.port (0.0)
d : drop     (at queue)
```

```
r 1.3556 3 2 ack 40 -------- 1 3.0 0.0 15 201
+ 1.3556 2 0 ack 40 -------- 1 3.0 0.0 15 201
- 1.3556 2 0 ack 40 -------- 1 3.0 0.0 15 201
r 1.35576 0 2 tcp 1000 -------- 1 0.0 3.0 29 199
+ 1.35576 2 3 tcp 1000 -------- 1 0.0 3.0 29 199
d 1.35576 2 3 tcp 1000 -------- 1 0.0 3.0 29 199
+ 1.356 1 2 cbr 1000 -------- 2 1.0 3.1 157 207
- 1.356 1 2 cbr 1000 -------- 2 1.0 3.1 157 207
```

Δημιουργία ζεύξης (**ΑΝΑΦΟΡΑ**)

- Calculate the packet delay through the trace file
- Study how packet interval affect performance through awk script (**ΑΝΑΦΟΡΑ**)
  - Packet size = 500
  - Packet interval 0.009 – 0.005 – 0.003 – 0.002 – 0.001
  - Κάντε γραφικές παραστάσεις για όλα τα performance metrics
  - Επαληθεύσατε τα delays από το trace file (print screens)
- Study how packet size affect performance through awk script (**ΑΝΑΦΟΡΑ**)
  - Packet interval 0.009
  - Packet size 500, 700, 1000, 1100, 1500
  - Κάντε γραφικές παραστάσεις για όλα τα performance metrics
  - Επαληθεύσατε τα delays από το trace file (print screens)

Δημιουργία ζεύξης (**ΑΝΑΦΟΡΑ**)

- Create an opposite data flow i.e. from n1 to n0
- Study performance effect when opposite flow exist
  - Calculate drop packets for 500Byte packets and 0.003 packet interval
  - Increase link bandwidth gradually by 0.1Mb until zero packet loss is recorded
  - Κάντε γραφικές παραστάσεις για όλα τα performance metrics
  - Επαληθεύσατε τα delays από το trace file (print screens)