

Εθνικό Μετσόβιο Πολυτεχνείο



Σχολη Ηλεκτρολογων Μηχανικων
Και
Μηχανικων Υπολογιστων

Δίκτυα Υπολογιστών

Πρώτη αναφορά - Network Simulator v2

Δήμητρα Γιαντσίδη (Α.Μ : 03112014)

9 Απριλίου 2015

Μετάδοση δεδομένων σε δίκτυο με σύνθετη τοπολογία

Σε αυτή την άσκηση θα ορίσουμε ένα δίκτυο που αποτελείται από 9 κόμβους. Επιλέγοντας 2 από αυτούς, οι οποίοι δεν συνδέονται άμεσα (διαδοχικοί), θα προσπαθήσουμε να στείλουμε τα δεδομένα από και προς αυτούς, δηλαδή οι κόμβοι να ανταλλάσουν ταυτόχρονα δεδομένα. Ο κώδικας για τον ορισμό της τοπολογίας και της ροής δεδομένων από και προς κόμβους 0 και 3 παρατίθεται παρακάτω.

```
set ns [new Simulator]

set nf [open lab3.nam w]
$ns namtrace-all $nf

for {set i 0} {$i < 9} {incr i} {
    set n($i) [$ns node]
}

for {set i 0} {$i < 7} {incr i} {
    $ns duplex-link $n($i) $n([expr ($i+1)%7]) 2Mb 40ms DropTail
}

$ns duplex-link $n(7) $n(1) 2Mb 20ms DropTail
$ns duplex-link $n(7) $n(5) 2Mb 10ms DropTail
$ns duplex-link $n(8) $n(5) 2Mb 10ms DropTail
$ns duplex-link $n(8) $n(2) 2Mb 40ms DropTail

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exit 0
}

set udp0 [new Agent/UDP]
$udp0 set packetSize_ 1500
$ns attach-agent $n(0) $udp0
$udp0 set fid_ 0
$ns color 0 green
set sink0 [new Agent/LossMonitor]
$ns attach-agent $n(0) $sink0

set udp3 [new Agent/UDP]
$udp3 set packetSize_ 1500
$ns attach-agent $n(3) $udp3
$udp3 set fid_ 3
$ns color 3 yellow
set sink3 [new Agent/LossMonitor]
$ns attach-agent $n(3) $sink3

$ns connect $udp0 $sink3
```

```

$ns connect $udp3 $sink0

set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 1500
$cbr0 set interval_ 0.015
$cbr0 attach-agent $udp0

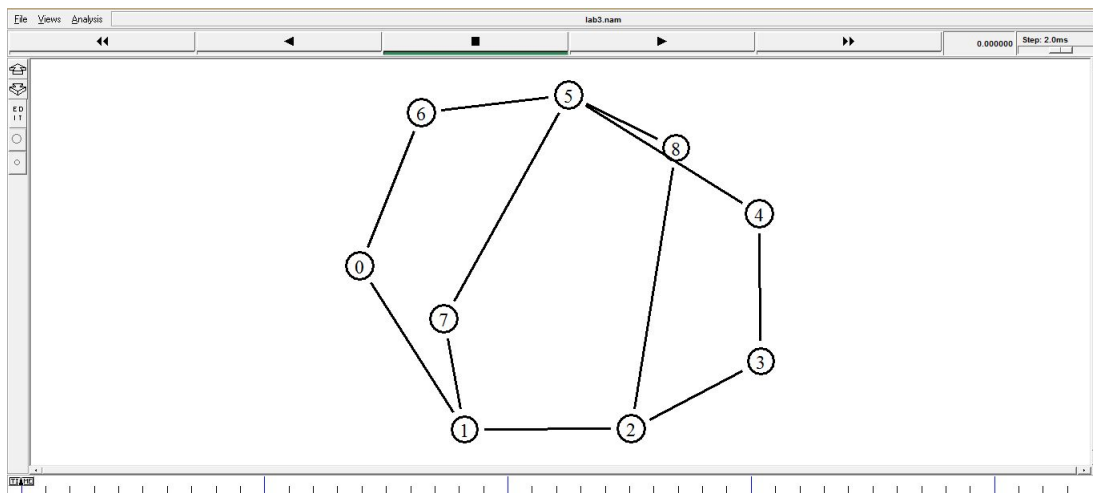
set cbr3 [new Application/Traffic/CBR]
$cbr3 set packetSize_ 1500
$cbr3 set interval_ 0.015
$cbr3 attach-agent $udp3

$ns at 0.2 "$cbr0_start"
$ns at 0.7 "$cbr3_start"
$ns at 3.7 "$cbr3_stop"
$ns at 4.2 "$cbr0_stop"

$ns at 4.5 "finish"
$ns run

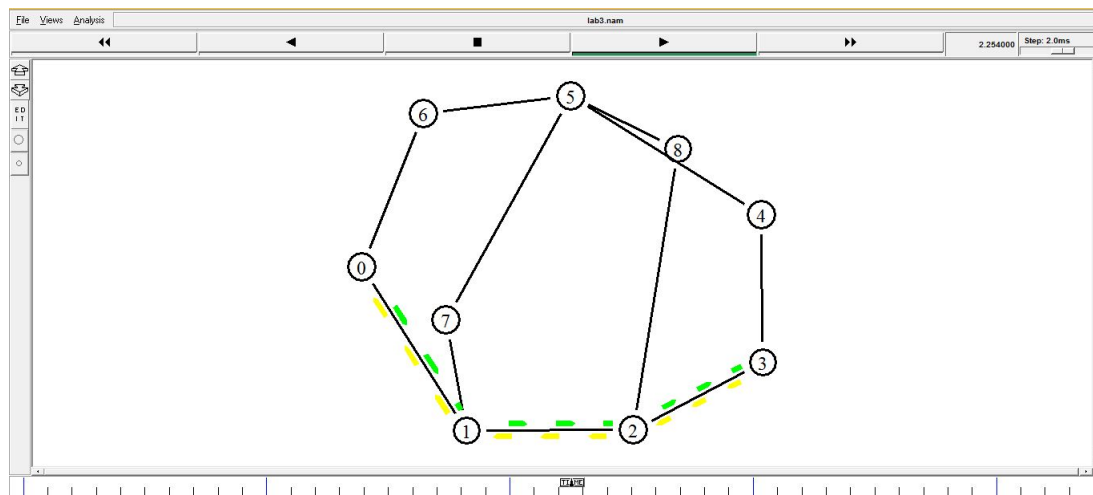
```

Κατά τα γνωστά τρέχουμε το script και η τοπολογία του δικτύου που λαμβάνουμε από την εκτέλεση του lab3.nam φαίνεται στο επόμενο σχήμα. Παραθέτουμε ακόμα, στιγμιότυπο από την προσομοίωση της λειτουργίας του δικτύου.



Σχήμα 1: Σύνθετη τοπολογία δικτύου 9 κόμβων

Σε ό,τι αφορά την προσομοίωση, πρώτα, τη χρονική στιγμή 0.2 sec, ξεκινούν τα πράσινα πακέτα από τον κόμβο 0 προς τον κόμβο 3. Στη συνέχεια, τη χρονική στιγμή 0.7 sec ξεκινάει η αποστολή των κίτρινων πακέτων από τον κόμβο 3 στον κόμβο 0. Παρατηρούμε, όπως επαληθεύει και το στιγμιότυπο (σχήμα 2) ότι γίνεται ταυτόχρονη αποστολή πράσινων και κίτρινων πακέτων μεταξύ των κόμβων 0 και 3. Τέλος, σταματάει πρώτα η αποστολή των κίτρινων πακέτων κατά τη χρονική στιγμή 3.7 sec και αργότερα, τη χρονική στιγμή 4.2 sec, η αποστολή των πράσινων. Η προσομοίωση τελειώνει στα 4.5 sec.



Σχήμα 2: Στιγμιότυπο προσομοίωσης

Ερωτήσεις

- Ποια διαδρομή ακολουθούν τα πακέτα;

Τα πράσινα πακέτα με "πηγή" τον κόμβο 0 και "προορισμό" τον κόμβο 3 ακολουθούν τη διαδρομή 0-1-2-3. Αντιστοίχως, τα κίτρινα πακέτα με "πηγή" τον κόμβο 3 και "προορισμό" τον κόμβο 0 ακολουθούν τη διαδρομή 3-2-1-0.

- Ελέγξτε αν η ροή των πακέτων και από τις δύο πλευρές ακολουθεί τη διαδρομή με τα λιγότερα βήματα.

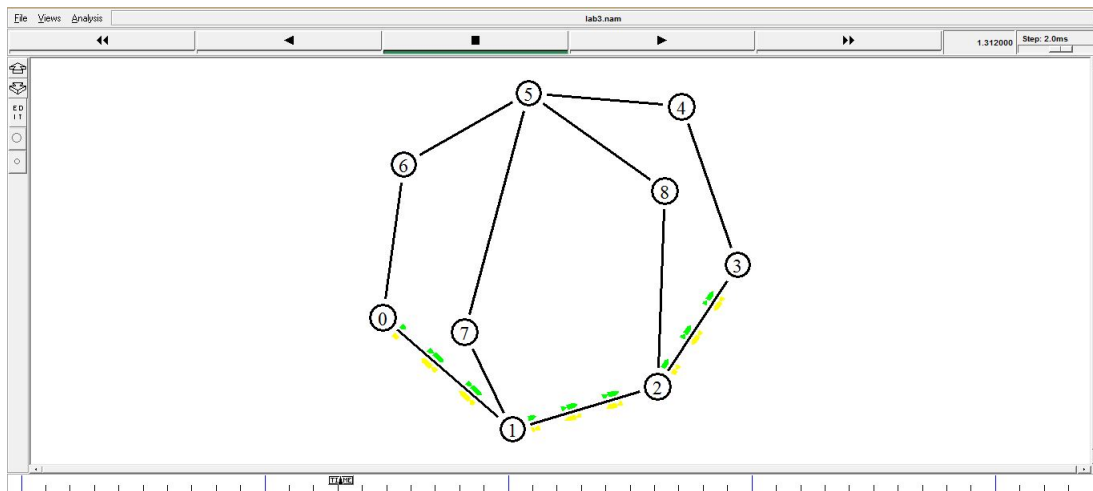
Όπως φαίνεται στο σχήμα 2 όλες οι υπόλοιπες πιθανές διαδρομές απαιτούν περισσότερα βήματα. Επομένως η επιλεχθείσα είναι η βέλτιστη.

- Υπάρχει συντομότερη ροή από αυτή που ακολουθούν, όσον αφορά τη συνολική καθυστέρηση κάθε ροής;

Παρατηρούμε από το σχήμα 2 ότι ο ελάχιστος αριθμός βημάτων από τον κόμβο 0 στον κόμβο 3 αντιστοιχεί στην διαδρομή που υποδεικνύει το σχήμα με χρονική καθυστέρηση $40 \cdot 3 = 120 \text{ ms}$ (αφού στους συγκεκριμένους κλάδους έχουμε καθυστέρηση 40 ms ανά κλάδο). Ωστόσο μιας και ορίσαμε και κάποιες συνδέσεις κόμβων με διαφορετικές καθυστερήσεις υπάρχει πιθανότητα να έχουμε μικρότερη χρονική καθυστέρηση με περισσότερα, ωστόσο, βήματα (διαδρομή μέσω περισσότερων κόμβων). Αρχικά, διαπιστώνουμε ότι οι διαδρομές 0-6-5-4-3, 0-6-5-8-2-3 δεν είναι συμφέρουσες γιατί έχουν συνολική καθυστέρηση $40 \cdot 4 = 160 \text{ ms}$ και $40 + 40 + 10 + 40 + 40 = 170 \text{ ms}$ αντίστοιχα. Άρα η ενδεχομένως καλύτερη διαδρομή θα είναι μέσω του κλάδου 1. Παρατηρούμε ότι ο χρόνος μεταξύ των κόμβων 0 και 1 απευθείας είναι 40 ms και είναι ο γρηγορότερος τρόπος αφού η σύνδεση του κόμβου 0 με τον 1 μέσω των κόμβων 6, 5 και 7 έχει καθυστέρηση $40 + 40 + 10 + 20 = 110 \text{ ms}$. Επομένως ως εδώ έχουμε αποδείξει ότι ο συντομότερος χρονικά δρόμος είναι μέσω της απευθείας σύνδεσης των κόμβων 0 και 1. Αντιστοίχως, αποκλείουμε την πιθανή διαδρομή 0-1-2-8-5-4-3, αφού η χρονική καθυστέρηση ξεπερνάει κατά πολύ την αρχική. Τελικά, η γρηγορότερη διαδρομή είναι αυτή που αρχικά επιλέχθηκε και από την άποψη των βημάτων και από την άποψη του χρόνου, αφού έχει τη μικρότερη δυνατή χρονική καθυστέρηση.

- Ποιος είναι ο ρόλος των εντολών `$udp0 set packetSize_ 1500` και `$udp3 set packetSize_ 1500`; Τι παρατηρείτε στις ροές των πακέτων αν αφαιρεθούν οι γραμμές αυτές από τον κώδικα της προσομοίωσης;

Αφαιρέσαμε τις συγκεκριμένες εντολές, τρέξαμε το script και το στιγμιότυπο που λάβαμε από το NAM, ήταν το εξής:

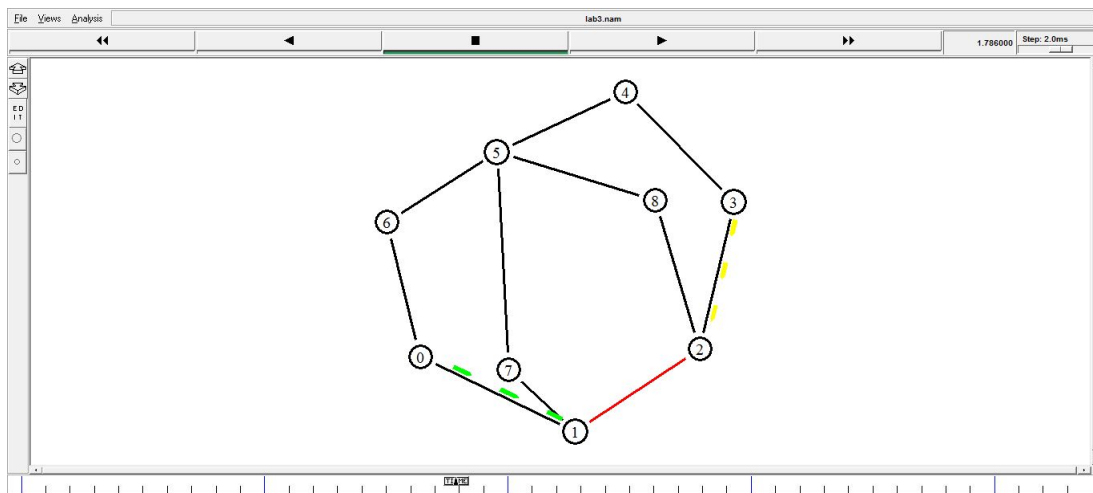


Σχήμα 3: Μέγιστο μήκος πακέτου : 1000 bytes

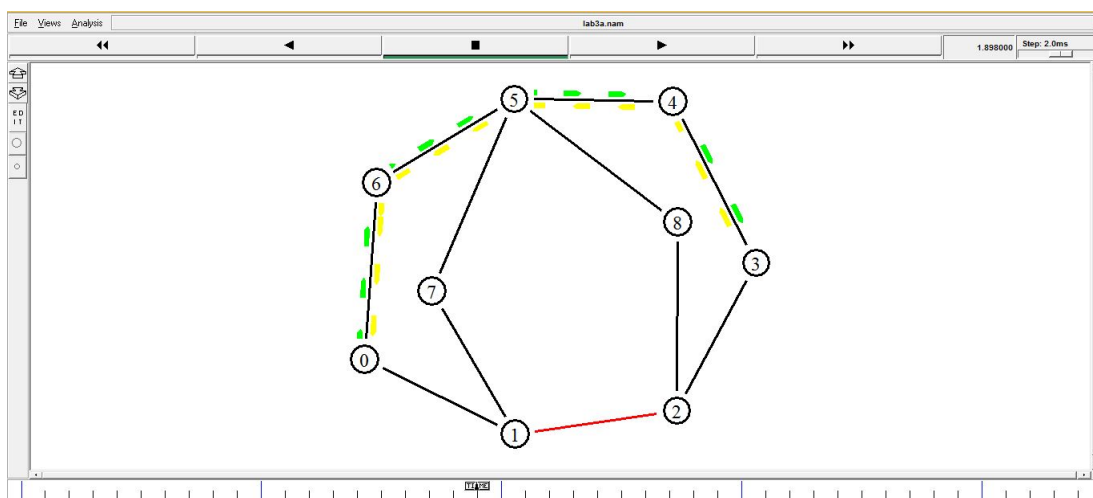
Βλέποντας το *σχήμα 3* διαπιστώνουμε ότι τα πακέτα σε αυτή την περίπτωση δεν στέλνονται εννιάα, αλλά χωρίζονται σε δύο άνισα μέρη. Αυτό συμβαίνει γιατί οι εντολές που διαγράψαμε ορίζουν ως το μέγιστο μήκος πακέτου που μεταφέρεται στα 1500 bytes, όσο είναι δηλαδή και το μήκος του πακέτου από την πηγή. Αυτός είναι και ο λόγος που στην πρώτη εκτέλεση βλέπουμε ότι τα πακέτα στέλνονται ολόκληρα (*σχήμα 2*). Αν το μέγιστο μήκος πακέτου στη ζεύξη δεν ορίζεται τότε αυτό θεωρείται 1000 bytes. Επομένως, αφού το μήκος του πακέτου είναι 1500 bytes, τα πακέτα στέλνονται χωρισμένα σε μέρη των 1000 bytes και 500 bytes.

Στατική και δυναμική δρομολόγηση

Σε αυτό το κομμάτι θα επιχειρήσουμε να διακόψουμε τη σύνδεση των κόμβων 1 και 2 για 1 sec. Το επιτυγχάνουμε προσθέτοντας το σύνολο εντολών στο script όπως υποδεικνύει η εκφώνηση της άσκησης. Στην πρώτη προσομοίωση η δρομολόγηση είναι στατική. Εν συνεχεία, επιχειρούμε να αλλάξουμε τη δρομολόγηση σε δυναμική. Αυτό σημαίνει ότι οι κόμβοι - "πηγές" θα πρέπει να αντιμετωπίζουν από μόνοι τους προβλήματα, όπως π.χ διακοπή της ζεύξης δύο ενδιαμέσων κόμβων, ώστε να συνεχιστεί η μεταφορά των δεδομένων. Οι εικόνες για τις δύο διαφορετικές προσομοιώσεις που τρέχουμε είναι :



Σχήμα 4: Προσομοίωση με στατική δρομολόγηση



Σχήμα 5: Προσομοίωση με δυναμική δρομολόγηση

Ερωτήσεις

- Εξηγήστε γιατί, με τη στατική δρομολόγηση, οι κόμβοι εξακολουθούν να στέλνουν πακέτα και μετά τη διακοπή της ζεύξης.

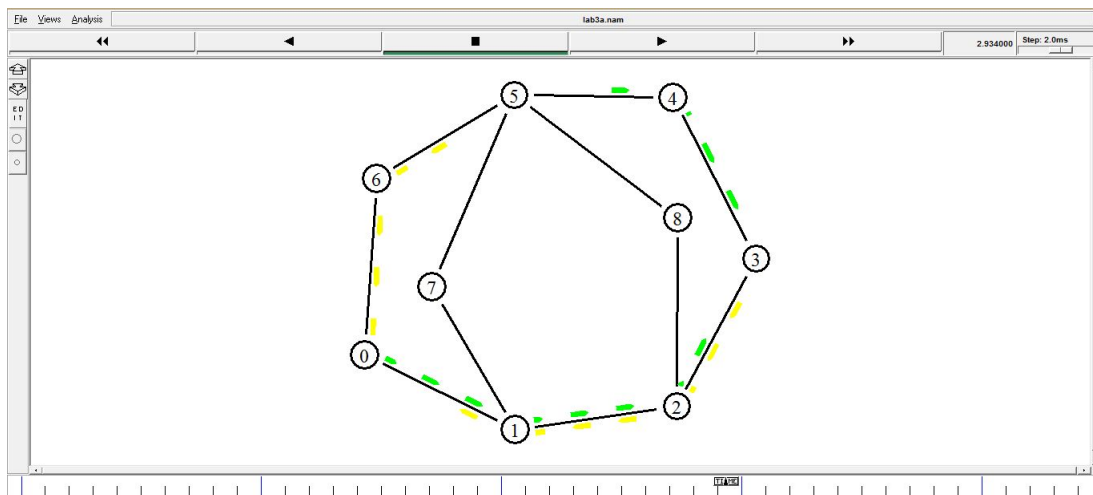
Όπως φαίνεται στο σχήμα 4 στη στατική δρομολόγηση, οι κόμβοι 0 και 3 εξακολουθούν να στέλνουν πακέτα παρότι η σύνδεση των 1 και 2 έχει απενεργοποιηθεί. Αυτό συμβαίνει γιατί στη στατική δρομολόγηση οι κόμβοι - "πηγές" δεν ενημερώνονται για τα προβλήματα που ενδεχομένως να παρουσιάζονται στη διαδρομή (σύνδεση μεταξύ των κόμβων).

- Τα πακέτα που χάθηκαν, θα ξαναμεταδοθούν από τους αντίστοιχους κόμβους, όταν επανέλθει η σύνδεση;

Όχι, γιατί τα πακέτα που έχουν σταλθεί κατά τη διάρκεια της ανενεργής ζεύξης, έχουν χαθεί. Επιπλέον, οι πηγές (κόμβοι 0 και 3) δεν έχουν ενημερωθεί για την αποτυχία λήψης πακέτων από τους παραλήπτες (κόμβοι 3 και 0 αντίστοιχα) και για αυτό το λόγο δεν θα ξαναστείλουν αυτά τα ήδη "χαμένα" πακέτα.

- Στο σχήμα 5 βλέπουμε την αρχική κατάσταση του δικτύου αμέσως μόλις κοπεί η ζεύξη. Παρατηρούμε ότι όταν κόβεται η ζεύξη 1-2 τα πακέτα που βρίσκονται στον κόμβο 1 ακολουθούν τη διαδρομή 1-7-5-4-3 και κατ' αντιστοιχία, τα πακέτα που έχουν σταλθεί στον κόμβο 2 την διαδρομή 2-8-5-6-0 για να φτάσουν στους αντίστοιχους κόμβους - προορισμούς. Παράλληλα ενημερώνονται οι κόμβοι - πηγές και τα πακέτα θα στέλλονται από εδώ και μπρος μέσω των διαδρομών 0-6-5-4-3 και 3-4-5-6-0. Κατά την επαναφορά της ζεύξης 1-2 τα πακέτα στέλλονται μέσω της αρχικής διαδρομής. Για ένα διάστημα μετά την επαναφορά της ζεύξης, παρατηρούμε στο animation ότι οι κόμβοι προορισμοί λαμβάνουν πακέτα και από τις δύο διαδρομές. Αυτό σταματάει όταν το τελευταίο πακέτο που στάλθηκε από την "εναλλακτική" οδό φτάσει στον προορισμό του.

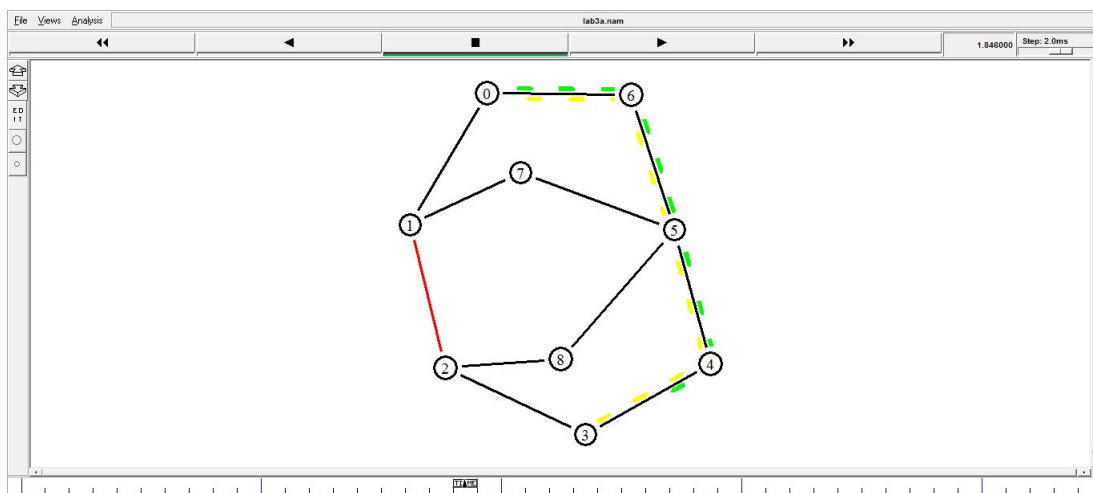
6



Σχήμα 7: Δυναμική δρομολόγηση - Επαναφορά της ζεύξης των κόμβων 1 και 2

- Με βάση το animation, προσδιορίστε για κάθε ροή τη χρονική στιγμή όπου παρατηρείται η μόνιμη διαδρομή δρομολόγησης κατά τη διάρκεια διακοπής.

Από το animation υπολογίζουμε ότι η μόνιμη κατάσταση επέρχεται τη χρονική στιγμή 1.85 sec.



Σχήμα 8: Μόνιμη κατάσταση στη χρονική στιγμή 1.85 sec

- Για ποιο λόγο τα πακέτα ακολουθούν τις συγκεκριμένες διαδρομές από-του πέσει η σύνδεση, στην αρχική και τη μόνιμη κατάσταση;

Ο λόγος που ακολουθούνται οι συγκεκριμένες διαδρομές είναι ότι είναι οι πιο σύντομες από την άποψη των βημάτων. Στην αρχική κατάσταση, τα πακέτα που είναι στον κόμβο 1 θα σταλούν μέσω της διαδρομής 1-7-5-4-3 και αντίστοιχα αυτά που είναι στον κόμβο 2 μέσω της διαδρομής 2-8-5-6-0 γιατί είναι οι συντομότερες διαδρομές στην παρούσα φάση που η ζεύξη 1-2 δεν λειτουργεί. Όμως, όταν οι κόμβοι - πηγές ενημερώνονται για το πρόβλημα στο δίκτυο (μόνιμη κατάσταση) και καλούνται να "αποφασίσουν" την διαδρομή μέσω της οποίας

θα στείλουν τα πακέτα, διαπιστώνουν ότι οι διαδρομές 0-6-5-4-3 και 3-4-5-6-0 είναι κατά ένα βήμα συντομότερες από τις 0-1-7-5-4-3 και 3-4-5-8-2-3. Για αυτό το λόγο επιλέγονται οι πρώτες.

- Θα μπορούσαν να δρομολογηθούν από άλλους κόμβους;

Ναι, θα μπορούσαν να δρομολογηθούν από οποιουδήποτε κόμβους, ωστόσο, η τάση είναι να επιλέγεται η συντομότερη σε βήματα διαδρομή.

- Ποιος από όλους τους κόμβους καθορίζει από ποια διαδρομή θα προωθηθούν κάθε φορά τα πακέτα;

Γενικά οι κόμβοι που "αποφασίζουν" για την διαδρομή είναι οι πηγές, στην περίπτωση μας δηλαδή, οι κόμβοι 0 και 3. Ωστόσο, στην αρχική μεταβατική κατάσταση παρατηρήσαμε ότι οι κόμβοι 1 και 2 αντίστοιχα δρομολόγησαν τα δεδομένα. Αυτό είναι πιθανό να συμβεί σε οποιοδήποτε κόμβο που φέρει δεδομένα και εκείνη τη στιγμή γίνονται αλλαγές στις ζεύξεις του δικτύου.

Καθορισμός κόστους ζεύξης

Μέχρι αυτό το σημείο έχουμε ορίσει τη συντομότερη διαδρομή ως εκείνη με τα λιγότερα βήματα (στην ουσία στα προηγούμενα θεωρούταν ότι όλες οι ζεύξεις είχαν ίσα κόστη), δηλαδή με τους λιγότερους κόμβους. Ωστόσο, κάθε ζεύξη έχει διαφορετική καθυστέρηση, επομένως χρονικά η συντομότερη διαδρομή μπορεί να είναι διαφορετική. Ο ολοκληρωμένος κώδικάς του script είναι :

```
set ns [new Simulator]

set nf [open lab3a.nam w]
$ns namtrace-all $nf

for {set i 0} {$i < 9} {incr i} {
    set n($i) [$ns node]
}

for {set i 0} {$i < 7} {incr i} {
    $ns duplex-link $n($i) $n([expr ($i+1)%7]) 2Mb 40ms DropTail
}

$ns duplex-link $n(7) $n(1) 2Mb 20ms DropTail
$ns duplex-link $n(7) $n(5) 2Mb 10ms DropTail
$ns duplex-link $n(8) $n(5) 2Mb 10ms DropTail
$ns duplex-link $n(8) $n(2) 2Mb 40ms DropTail

for {set i 0} {$i < 7} {incr i} {
    $ns cost $n($i) $n([expr ($i+1)%7]) 4
```

```

}

for {set i 0} {$i < 7} {incr i} {
    $ns cost $n([expr ($i+1)%7]) $n($i) 4
}

$ns cost $n(7) $n(1) 2
$ns cost $n(7) $n(5) 1
$ns cost $n(8) $n(5) 1
$ns cost $n(8) $n(2) 4

$ns cost $n(1) $n(7) 2
$ns cost $n(5) $n(7) 1
$ns cost $n(5) $n(8) 1
$ns cost $n(2) $n(8) 4

Agent/rtProto/Direct set preference_ 200
$ns rtproto DV

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exit 0
}

set udp0 [new Agent/UDP]
$udp0 set packetSize_ 1500
$ns attach-agent $n(0) $udp0
$udp0 set fid_ 0
$ns color 0 green
set sink0 [new Agent/LossMonitor]
$ns attach-agent $n(0) $sink0

set udp3 [new Agent/UDP]
$udp3 set packetSize_ 1500
$ns attach-agent $n(3) $udp3
$udp3 set fid_ 3
$ns color 3 yellow
set sink3 [new Agent/LossMonitor]
$ns attach-agent $n(3) $sink3

$ns connect $udp0 $sink3
$ns connect $udp3 $sink0

set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 1500
$cbr0 set interval_ 0.015
$cbr0 attach-agent $udp0

set cbr3 [new Application/Traffic/CBR]

```

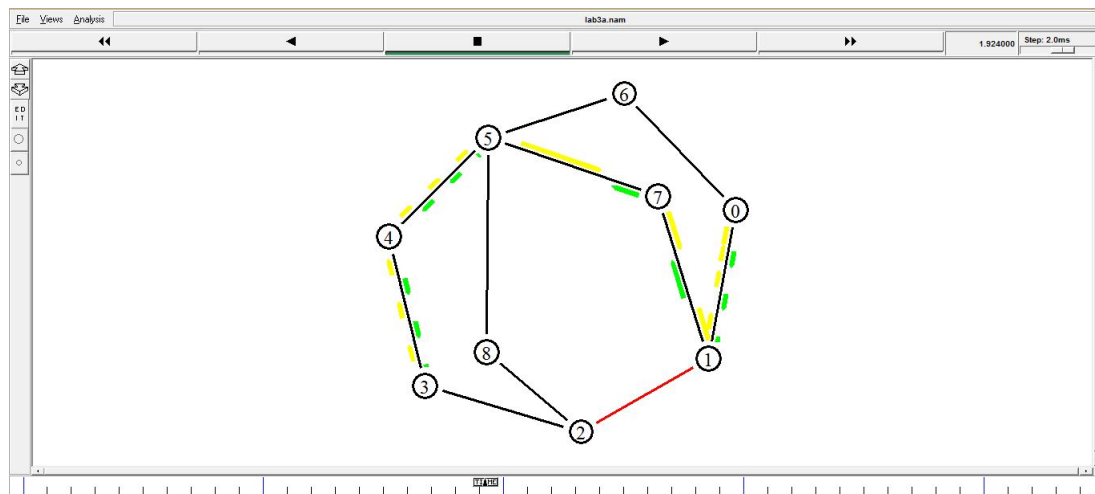
```

$cbr3 set packetSize_ 1500
$cbr3 set interval_ 0.015
$cbr3 attach-agent $udp3

$ns at 0.2 "$cbr0_start"
$ns at 0.7 "$cbr3_start"
$ns rtmodel-at 1.7 down $n(1) $n(2)
$ns rtmodel-at 2.7 up $n(1) $n(2)
$ns at 3.7 "$cbr3_stop"
$ns at 4.2 "$cbr0_stop"
$ns at 4.5 "finish"
$ns run

```

Τρέξαμε το παραπάνω κώδικα, ο οποίος λαμβάνει υπόψη και τα κόστη ζεύξεων και το αποτέλεσμα είναι το παρακάτω στιγμιότυπο.



Σχήμα 9: Στιγμιότυπο προσομοίωσης με δυναμική δρομολόγηση και με άνισα κόστη ζεύξεων

Ερωτήσεις

- Ποιες διαδρομές ακολουθούν τα πακέτα πριν, κατά τη διάρκεια και μετά την πτώση της σύνδεσης για τις δύο ροές;

Τα πακέτα σε αυτή την περίπτωση θα ακολουθήσουν την συντομότερη διαδρομή. Πριν την πτώση της σύνδεσης ακολουθούν την ίδια διαδρομή με πριν, δηλαδή για τα πράσινα πακέτα της 0-1-2-3 και για τα κίτρινα 3-2-1-0. Όταν πέφτει η σύνδεση παρατηρούμε ότι οι νέες διαδρομές στη μόνιμη κατάσταση είναι, για τα πράσινα πακέτα 0-1-7-5-4-3 και για τα κίτρινα είναι 3-4-5-7-1-0. Παρατηρούμε ότι στην αρχική μεταβατική κατάσταση τα πράσινα πακέτα ακολουθούν την ίδια διαδρομή εν αντιθέσει με τα κίτρινα που ακολουθούν την 3-2-8-5-6-0. Υπολογίζοντας το συνολικό κόστος της διαδρομής 0-1-7-5-4-3 διαπιστώνουμε ότι είναι $4 + 2 + 1 + 4 + 4 = 15$ δηλαδή μικρότερο από $4 + 4 + 4 + 4 = 16$ που είναι το κόστος της διαδρομής 0-6-5-4-3 που ακολουθούνταν στα προηγούμενα ερωτήματα και μικρότερο από το

κόστος της 3-2-8-5-6-0 που είναι $4 + 4 + 1 + 4 + 4 = 17$. Όταν επανέλθει η σύνδεση η διαδρομή είναι η 0-1-2-3 με κόστος $4 + 4 + 4 = 12$.

- Για ποιον λόγο τα πακέτα ακολουθούν τις συγκεκριμένες διαδρομές;

Είναι οι συντομότερες. Στο παραπάνω ερώτημα το αποδείξαμε και υπολογιστικά.

- Θα μπορούσαν να δρομολογηθούν από άλλους κόμβους;

Είναι δυνατό να συμβεί, ωστόσο επιλέχθηκαν οι διαδρομές με το λιγότερο δυνατό κόστος.

- Πριν την αποκατάσταση της ζεύξης μεταξύ των κόμβων "1" και "2", προσδιορίστε με βάση το animation τη χρονική στιγμή όπου παρατηρείται η μόνιμη διαδρομή δρομολόγησης για κάθε ροή.

Τα πράσινα πακέτα με το που πέσει η ζεύξη μεταξύ των κόμβων 1 και 2 ακολουθούν αμέσως την 0-1-7-5-4-3. Ωστόσο, τα κίτρινα πακέτα που είναι στον κόμβο 2 πριν η πηγή - κόμβος 3 αλλάξει τη ροή δεδομένων για να φτάσουν στον προορισμό τους ακολουθούν την διαδρομή 2-8-5-6-0. Επέρχεται η μόνιμη κατάσταση όταν και το τελευταίο από εκείνα τα πακέτα φτάσει στον προορισμό του. Αυτό παρατηρώντας την προσομοίωση συμβαίνει τη χρονική στιγμή 1.85 sec, όπως και την προηγούμενη φορά. Το αποτέλεσμα είναι αναμενόμενο γιατί οι καθυστερήσεις στις ζεύξεις και το μέγεθος των κίτρινων πακέτων παραμένουν ίδια.

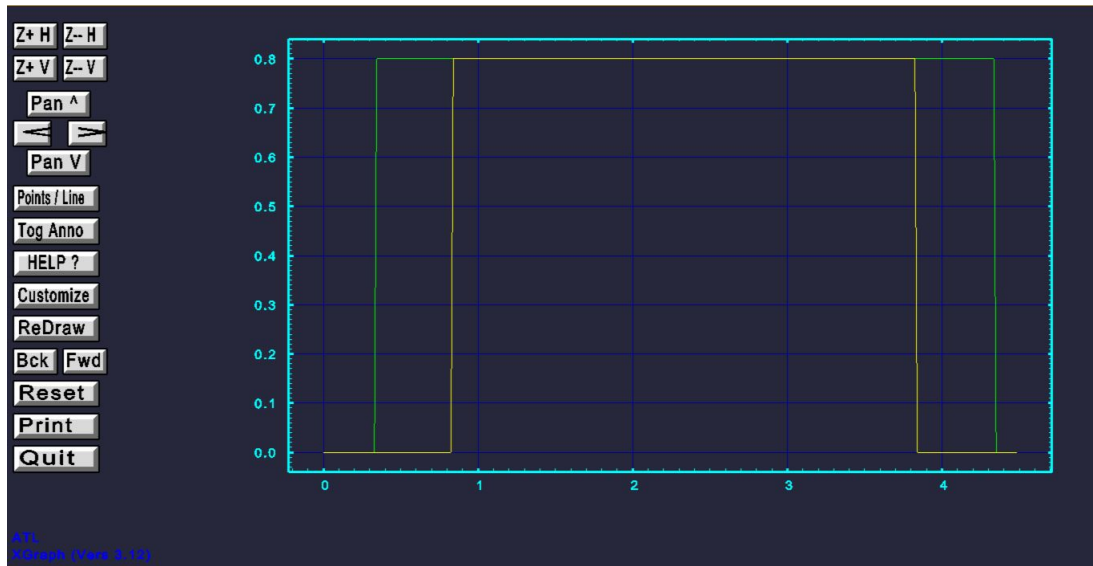
item Μετά την αποκατάσταση της ζεύξης μεταξύ των κόμβων "1" και "2", προσδιορίστε με βάση το animation τη χρονική στιγμή όπου παρατηρείται η μόνιμη διαδρομή δρομολόγησης για κάθε ροή.

Παρατηρούμε την προσομοίωση και συμπεραίνουμε ότι ο χρόνος κατά τον οποίο παρατηρείται πλήρης επαναφορά της αρχικής κατάστασης, δηλαδή τα πακέτα ρέουν μόνο μέσω της διαδρομής 0-1-2-3 είναι 2.95 sec.

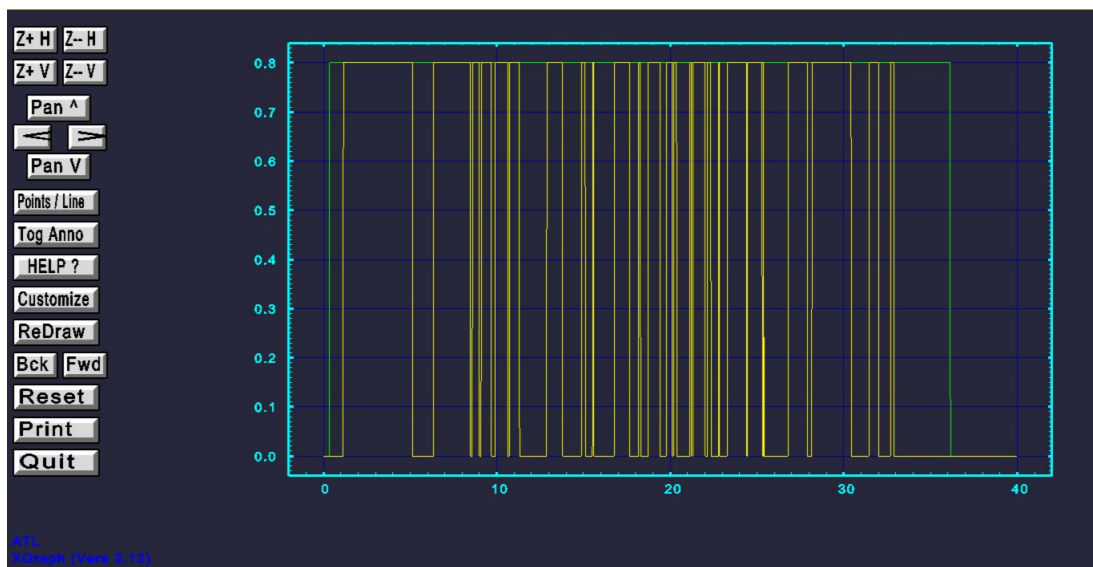
- Ποιος είναι ο ρόλος της εντολής Agent/rtProto/Direct set preference_ 200; Τι παρατηρείτε στη δρομολόγηση των πακέτων αν αφαιρεθεί η εντολή αυτή από τον κώδικα της προσομοίωσης; Αιτιολογείστε γιατί συμβαίνει αυτό.

Παρακολούθηση εκθετικής κίνησης με το Xgraph

Με χρήση του Xgraph θα δημιουργήσουμε τις γραφικές παραστάσεις των ροών του δικτύου. Στην πρώτη προσομοίωση η ροή είναι σταθερή (CBR), ενώ στην δεύτερη ρυθμίζουμε εκθετική (Exponential) την κίνηση των κίτρινων πακέτων και διατηρούμε σταθερή (CBR) την κίνηση των πράσινων πακέτων.



Σχήμα 10: CBR και οι δύο πηγές



Σχήμα 11: CBR η πηγή κόμβος 0 και Exponential η πηγή κόμβος 3

Ερωτήσεις

- Ποιος είναι ο μέγιστος ρυθμός μετάδοσης που επιτυγχάνεται για τις δύο περιπτώσεις κίνησης, βάσει των γραφικών παραστάσεων που σχεδιάσατε;

Στην πρώτη εκδοχή, όπου έχουμε σταθερή ροή και από τις δύο πηγές, ο μέγιστος ρυθμός μετάδοσης είναι και για τις δύο ροές $\frac{1500}{0.015} = 100000 \text{ bytes/sec} \Rightarrow 800000 \text{ bits/sec} = 0.8 \text{ Mbps}$. Επαληθεύουμε το αποτέλεσμα και από το σχήμα 10.

Στην δεύτερη εκδοχή έχουμε σταθερή ροή στην πηγή 0 και εκθετική στην πηγή 3. Ο μέγιστος ρυθμός μετάδοσης είναι από το σχήμα 11 ίσος με 0.8 Mbps.

- Αιτιολογείστε τις μέγιστες τιμές που προσδιορίσατε παραπάνω, χρησιμοποιώντας τις παραμέτρους που θέσατε για τη διαμόρφωση των δύο πηγών κίνησης (CBR και Exponential).

Για τη ροή CBR αποδείξαμε υπολογιστικά ότι η θεωρητική τιμή συμπίπτει με την πραγματική. Για τη δεύτερη περίπτωση ομοίως δείχνουμε ότι ο μέγιστος ρυθμός μετάδοσης για τα πράσινα πακέτα (CBR) είναι 0.8 Mbps. Εντούτοις, πρέπει να επιβεβαιώσουμε το συμπέρασμα και για τα κίτρινα πακέτα. Η εντολή `$exp3 set rate_ 800k` ορίζει κατευθείαν το μέγιστο ρυθμό μεταφοράς στα 800 Kbits/sec = 0.8 Mbps

- Υπολογίστε το πλήθος των bytes που λαμβάνονται επιτυχώς στον προορισμό για κάθε ροή, θεωρώντας ότι και οι δύο ροές ολοκληρώνονται σε χρόνο $t = 20 + \alpha \cdot 10 \text{ sec}$, όπου α τα δύο τελευταία ψηφία του αριθμού μητρώου σας.

$\alpha = 14$: Ο χρόνος μέσα στον οποίο ολοκληρώνονται και οι δύο ροές είναι $t = 20 + 14 \bmod 10 \text{ sec} = 24 \text{ sec}$. Για τα πράσινα πακέτα ο αριθμός των bytes είναι : $0.8 \cdot \frac{\text{Mbits}}{\text{sec}} \cdot 24 \cdot \text{sec} = 19200000 \text{ bits} \Rightarrow \frac{19200000}{8} = 2400000 \text{ bytes}$.