

**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**  
**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ**  
**ΥΠΟΛΟΓΙΣΤΩΝ**



**ΔΙΚΤΥΑ ΕΠΙΚΟΙΝΩΝΙΩΝ**

(2018-2019)

*6<sup>η</sup> ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΝΑΦΟΡΑ*

Ονοματεπώνυμο : Χρήστος Τσούφης

A.M. : 03117176

## 1. Εισαγωγή

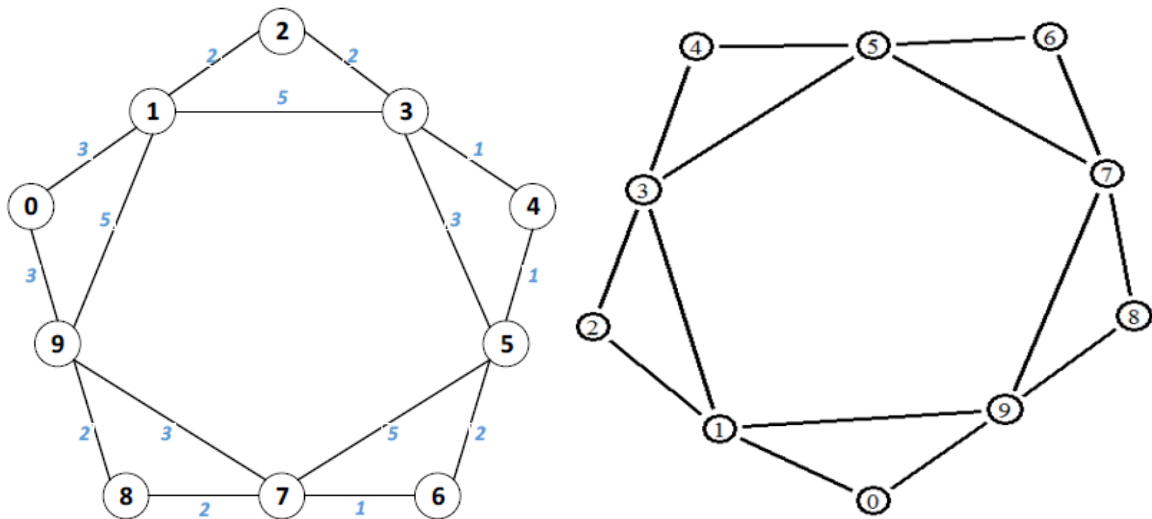
Σε αυτή την άσκηση γίνεται μελέτη της επίδοση του πρωτοκόλλου δρομολόγησης διανύσματος αποστάσεων (distance vector - DV).

Η άσκηση περιλαμβάνει δύο μέρη. Στο πρώτο μέρος θα δημιουργηθεί ένα δίκτυο με σύνθετη τοπολογία με την οποία θα επαληθευθεί η λειτουργία του πρωτοκόλλου δρομολόγησης DV. Στο δεύτερο μέρος θα παρατηρηθεί ότι για τοπολογίες συγκεκριμένης μορφής το πρωτόκολλο DV αποτυγχάνει.

Το Πρωτόκολλο Δρομολόγησης Διανύσματος Αποστάσεων (Distance Vector Routing Protocol ή DVRP) χρησιμοποιείται για την εύρεση ελαχίστων μονοπατιών με χρήση του κατανεμημένου αλγορίθμου Bellman-Ford/Floyd-Warshall. Ο αλγόριθμος λειτουργεί υποχρεώνοντας κάθε δρομολογητή να διατηρεί έναν πίνακα με τις βέλτιστες αποστάσεις προς τους άλλους κόμβους. Σε κάθε επανάληψη ο κάθε πίνακας του δικτύου  $G(V,E)$  ενημερώνεται με τις τρέχουσες εκτιμήσεις ελαχίστων διαδρομών βάσει της σχέσης  $\delta_x(y) = \min \{ w(x,u) + \delta_u(y) \}$ , όπου  $w: E \rightarrow \mathbb{R}^+$  η συνάρτηση βαρών του δικτύου.

## 2. Επαλήθευση πρωτοκόλλου DV

Αρχικά δημιουργείται το δίκτυο του Σχήματος, όπου όλες οι ζεύξεις είναι αμφίδρομες και έχουν ταχύτητα μετάδοσης 1 Mb/s. Ο αριθμός πάνω σε κάθε ζεύξη αναπαριστά το κόστος μετάδοσης μέσω αυτής (και προς τις δύο κατευθύνσεις), έτσι ώστε μία μονάδα κόστους να αντιστοιχεί σε καθυστέρηση ίση με 10 msec. Θεωρείται ότι οι ουρές αναμονής για όλες τις ζεύξεις είναι τύπου DropTail.



Χρησιμοποιείται δυναμική δρομολόγηση (routing protocol distance vector – DV), οπότε οι κόμβοι/δρομολογητές επιλέγουν αυτόματα τις συντομότερες διαδρομές.

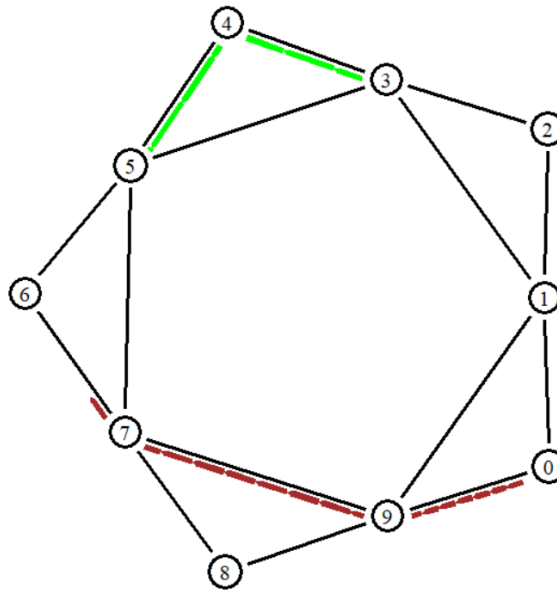
Ο κόμβος 0 στέλνει 150 πακέτα κίνησης FTP μεγέθους 540 Bytes (συμπεριλαμβανομένων των επικεφαλίδων) προς τον κόμβο 5 (ροή 1), ξεκινώντας τη μετάδοση τη χρονική στιγμή  $t=0.5$  sec. Επιπλέον, ο κόμβος 3 στέλνει άλλα 150 πακέτα κίνησης FTP μεγέθους 540 Bytes (συμπεριλαμβανομένων των επικεφαλίδων) προς τον κόμβο 9 (ροή 2), ξεκινώντας τη μετάδοση τη χρονική στιγμή  $t=0.8$  sec. Θεωρείστε ότι η προσομοίωση ολοκληρώνεται τη χρονική στιγμή  $t=4.0$  sec. Προσαρμόστε τον κώδικα στο Tcl script, ώστε να εμφανίσετε τη ροή 1 με καφέ χρώμα και τη ροή 2 με πράσινο χρώμα.

## 2.1 Ερωτήσεις

1. Ποιες διαδρομές ακολουθούν τα πακέτα δεδομένων για τις δύο μεταδόσεις; Ποιες διαδρομές ακολουθούν τα πακέτα επιβεβαιώσεων για τις δύο μεταδόσεις; Παραθέστε τον ολοκληρωμένο κώδικα προσομοίωσης.

Παρατηρώντας την προσομοίωση του nam προκύπτει ότι:

- Τα πακέτα δεδομένων της ροής 1 (0→5) ακολουθούν την διαδρομή: 0→9→7→6→5 (με συνολική καθυστέρηση: 90msec)
- Τα πακέτα δεδομένων της ροής 2 (3→9) ακολουθούν την διαδρομή: 3→4→5→6→7→9 (με συνολική καθυστέρηση: 80msec)
- Τα πακέτα επιβεβαιώσεων της ροής 1 (5→0) ακολουθούν την διαδρομή: 5→6→7→9→0
- Τα πακέτα επιβεβαιώσεων της ροής 2 (9→3) ακολουθούν την διαδρομή: 9→7→6→5→4→3



Παρατηρείται ότι η διαδρομή των πακέτων επιβεβαίωσης είναι ακριβώς αντίθετη από αυτή των πακέτων δεδομένων.

Πηγαίος κώδικας:

```
set ns [new Simulator]
set nf [open lab6.nam w]
$ns namtrace-all $nf
set trf [open lab6.tr w]
$ns trace-all $trf
```

```
Agent/rtrProto/Direct set preference_ 200
```

```
$ns rtrproto DV
```

```
for {set i 0} {$i < 10} {incr i} {
  set n($i) [$ns node]
}
```

```
$ns duplex-link $n(0) $n(1) 1Mb 30ms DropTail
```

```
$ns cost $n(0) $n(1) 3
```

```
$ns cost $n(1) $n(0) 3
```

```

$ns duplex-link $n(1) $n(2) 1Mb 20ms DropTail
$ns cost $n(1) $n(2) 2
$ns cost $n(2) $n(1) 2
$ns duplex-link $n(2) $n(3) 1Mb 20ms DropTail
$ns cost $n(2) $n(3) 2
$ns cost $n(3) $n(2) 2
$ns duplex-link $n(3) $n(4) 1Mb 10ms DropTail
$ns cost $n(3) $n(4) 1
$ns cost $n(4) $n(3) 1
$ns duplex-link $n(4) $n(5) 1Mb 10ms DropTail
$ns cost $n(4) $n(5) 1
$ns cost $n(5) $n(4) 1
$ns duplex-link $n(5) $n(6) 1Mb 20ms DropTail
$ns cost $n(5) $n(6) 2
$ns cost $n(6) $n(5) 2
$ns duplex-link $n(6) $n(7) 1Mb 10ms DropTail
$ns cost $n(6) $n(7) 1
$ns cost $n(7) $n(6) 1
$ns duplex-link $n(7) $n(8) 1Mb 20ms DropTail
$ns cost $n(7) $n(8) 2
$ns cost $n(8) $n(7) 2
$ns duplex-link $n(8) $n(9) 1Mb 20ms DropTail
$ns cost $n(8) $n(9) 2
$ns cost $n(9) $n(8) 2
$ns duplex-link $n(9) $n(0) 1Mb 30ms DropTail
$ns cost $n(9) $n(0) 3
$ns cost $n(0) $n(9) 3
$ns duplex-link $n(1) $n(3) 1Mb 50ms DropTail
$ns cost $n(1) $n(3) 5
$ns cost $n(3) $n(1) 5
$ns duplex-link $n(3) $n(5) 1Mb 30ms DropTail
$ns cost $n(3) $n(5) 3
$ns cost $n(5) $n(3) 3
$ns duplex-link $n(5) $n(7) 1Mb 50ms DropTail
$ns cost $n(5) $n(7) 5
$ns cost $n(7) $n(5) 5
$ns duplex-link $n(7) $n(9) 1Mb 30ms DropTail
$ns cost $n(7) $n(9) 3
$ns cost $n(9) $n(7) 3
$ns duplex-link $n(9) $n(1) 1Mb 50ms DropTail
$ns cost $n(9) $n(1) 5
$ns cost $n(1) $n(9) 5

```

```

set tcp1 [new Agent/TCP]
set sink1 [new Agent/TCPSink]
set ftp1 [new Application/FTP]

```

```

$tcp1 set packetSize_ 500
$tcp1 set fid_ 1
$ns attach-agent $n(0) $tcp1
$ns attach-agent $n(5) $sink1
$ns connect $tcp1 $sink1
$ftp1 attach-agent $tcp1
$ns color 1 brown
set tcp2 [new Agent/TCP]
set sink2 [new Agent/TCPSink]
set ftp2 [new Application/FTP]
$tcp2 set packetSize_ 500
$tcp2 set fid_ 2
$ns attach-agent $n(3) $tcp2
$ns attach-agent $n(9) $sink2
$ns connect $tcp2 $sink2
$ftp2 attach-agent $tcp2
$ns color 2 green

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exit 0
}
$ns at 0.5 "$ftp1 produce 150"
$ns at 0.8 "$ftp2 produce 150"
$ns at 4.0 "finish"
$ns run

```

2. Να εκτελέσετε με το χέρι τον αλγόριθμο διανύσματος αποστάσεων και να επαληθεύσετε την ορθή λειτουργία του πρωτοκόλλου που οδηγεί στην δρομολόγηση μέσω των διαδρομών ελαχίστου κόστους.

Μετά την εκτέλεση του αλγορίθμου Bellman-Ford, ο οποίος χρησιμοποιείται για την δρομολόγηση με την διαδρομή ελαχίστου κόστους, με το χέρι προκύπτουν τα εξής:

Για την ροή 1:

Αριθμός Επαναλήψεων	0	1	2	3	4	5	6	7	8	9
1 <sup>η</sup>	-	3	5	8	8	10	11	14	14	3
				7		9	7	12	5	
								6		
2 <sup>η</sup>							7			
3 <sup>η</sup>	Καμία μεταβολή									

	0	1	2	3	4	5	6	7	8	9
Προηγούμενος Κόμβος	-	0	1	2	3	4 ή 6	7	9	9	0

Επομένως η διαδρομή με το ελάχιστο κόστος είναι: 0→9→7→6→5 και 0→1→2→3→4→5.

Το πρωτόκολλο επιλέγει την πρώτη από τις δύο πιο πάνω, άρα επαληθεύεται πως είναι ορθή η λειτουργία του.

Για την ροή 2:

Αριθμός Επαναλήψεων	0	1	2	3	4	5	6	7	8	9
1 <sup>η</sup>	8	5	2	-	1	3	4	7	7	10
		4				2		5		8
2 <sup>η</sup>										
3 <sup>η</sup>	Καμία μεταβολή									

	0	1	2	3	4	5	6	7	8	9
Προηγούμενος Κόμβος	1	2	3	-	3	4	5	6	7	7

Επομένως η διαδρομή με το ελάχιστο κόστος είναι: 3→4→5→6→7→9.

Το πρωτόκολλο επιλέγει την παραπάνω διαδρομή, άρα επαληθεύεται πως είναι ορθή η λειτουργία του.

3. Χρησιμοποιώντας κατάλληλο script σε γλώσσα awk, βάσει του αρχείου ίχνους της προσομοίωσης, να προσδιοριστεί το πλήθος των πακέτων TCP που λαμβάνονται από τους κόμβους προορισμού για κάθε μία από τις δύο ροές. Ολοκληρώνεται η μετάδοση κάθε ροής κατά τη διάρκεια της προσομοίωσης; Εάν ναι, καταγράψτε τις χρονικές στιγμές ολοκλήρωσης των δύο μεταδόσεων. Παραθέστε τον κώδικα awk που χρησιμοποιήσατε.

Με την εντολή: awk95.exe -f lab6.awk < lab6.tr εκτελείται το awk.

Με την εντολή: awk95.exe -f lab6.awk < lab6.tr > results.txt αποθηκεύονται τα αρχεία στο results.txt.

Πηγαίος κώδικας:

```
BEGIN {
packets1 = 0;
packets2 = 0;
time1 = 0;
time2 = 0;
}
/^r/&&tcp/ {
if($4 == 5 && $6 == 540 && $8 == 1) {
packets1++;
time1 = $2;
}
if($4 == 9 && $6 == 540 && $8 == 2) {
packets2++;
time2 = $2;
}
}
END {
```

```
printf("Total Packets received for flow 1: %d\n", packets1);
printf("Total packets received for flow 2: %d\n", packets2);
printf("Last packet for flow 1 received at: %f sec\n", time1);
printf("Last packet for flow 2 received at: %f sec\n", time2);
}
```

Και μετά την εκτέλεση του awk προκύπτει:

```
C:\Users\Chris Tsoufis\Desktop>awk95.exe -f lab6.awk < lab6.tr
Total Packets received for flow 1: 150
Total packets received for flow 2: 150
Last packet for flow 1 received at: 2,700240 sec
Last packet for flow 2 received at: 2,861680 sec
```

Άρα επιβεβαιώνεται ότι και οι δύο ροές ολοκληρώνουν τη μετάδοση τους και αποστέλλουν και τα 150 πακέτα.

Η μετάδοση για τη ροή 1 ολοκληρώνεται για  $t=2.700240\text{sec}$ .

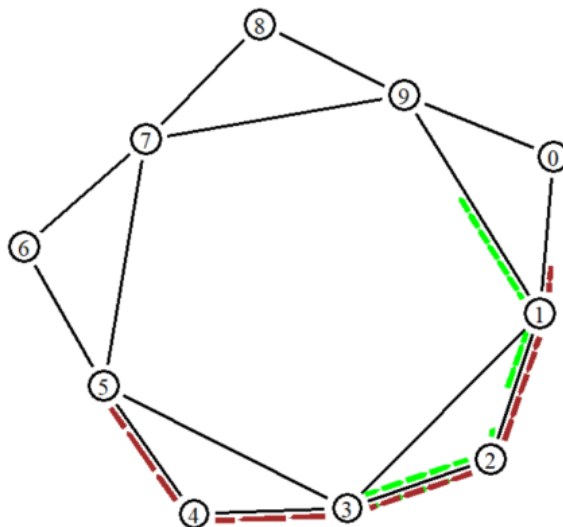
Η μετάδοση για τη ροή 2 ολοκληρώνεται για  $t=2.861680\text{sec}$ .

4. Ποιο είναι το μέγιστο κόστος μετάδοσης της ζεύξης 5-6, ώστε να διατηρούνται βέλτιστες οι διαδρομές που προσδιορίσατε στο προηγούμενο ερώτημα για τις δύο μεταδόσεις; Είναι οι μόνες συντομότερες διαθέσιμες διαδρομές σε αυτήν την περίπτωση; Επαληθεύστε την απάντησή σας τόσο θεωρητικά (εκτέλεση αλγορίθμου DV με το χέρι), όσο και πειραματικά με τη βοήθεια του NS2, υποδεικνύοντας όλες τις διαδρομές ελαχίστους κόστους για κάθε μετάδοση.

Από την εκτέλεση του αλγόριθμου με το χέρι, στο ερώτημα 2, προέκυψε ότι υπάρχει εναλλακτική διαδρομή για την ροή 1 με το ίδιο κόστος, η οποία δεν συμπεριλαμβάνει τη ζεύξη 5-6. Άρα οποιαδήποτε αύξηση στο κόστος στην ζεύξη 5-6 θα έχει ως αποτέλεσμα την αλλαγή της βέλτιστης διαδρομής. Για την επαλήθευση του πιο πάνω ισχυρισμού εκτελείται αρχικά η προσομοίωση αλλάζοντας το κόστος μετάδοσης της ζεύξης 5-6 από 2 σε 3 (30msec καθυστέρηση).

Τα πακέτα δεδομένων της ροής 1 (0→5) ακολουθούν τη διαδρομή: 0→1→2→3→4→5  
(συνολική καθυστέρηση: 90msec)

Τα πακέτα δεδομένων της ροής 2 (3→9) ακολουθούν τη διαδρομή: 3→2→1→9  
(συνολική καθυστέρηση: 90msec)



Παρατηρείται δηλαδή ότι και για τις δύο ροές η βέλτιστη διαδρομή έχει αλλάξει.

Για επαλήθευση, εκτελέσθηκε με το χέρι ο αλγόριθμος Bellman-Ford με το κόστος της ζεύξης 5-6 να είναι 3 οπότε προέκυψαν:

Για την ροή 1:

Αριθμός Επαναλήψεων	0	1	2	3	4	5	6	7	8	9
1 <sup>η</sup>	-	3	5	8	8	10	12	14	15	3
				7		9	7	13	5	
								6		
2 <sup>η</sup>							7			
3 <sup>η</sup>	Καμία μεταβολή									

	0	1	2	3	4	5	6	7	8	9
Προηγούμενος Κόμβος	-	0	1	2	3	4	7	9	9	0

Αρα η διαδρομή με το ελάχιστο κόστος είναι: 0→1→2→3→4→5.

Για την ροή 2:

Αριθμός Επαναλήψεων	0	1	2	3	4	5	6	7	8	9
1 <sup>η</sup>	8	5	2	-	1	3	5	7	8	10
		4				2		6		9
2 <sup>η</sup>	7									
3 <sup>η</sup>	Καμία μεταβολή									

	0	1	2	3	4	5	6	7	8	9
Προηγούμενος Κόμβος	1	2	3	-	3	4	5	6	7	7 ή 1

Αρα η διαδρομή με το ελάχιστο κόστος είναι: 3→4→5→6→7 ή 3→2→1→9.

Συνεπώς, επαληθεύεται ο ισχυρισμός ότι αν αυξηθεί το κόστος μετάδοσης της ζεύξης 5-6, θα αλλάξουν οι βέλτιστες διαδρομές άρα πρέπει να διατηρηθεί η καθυστέρηση στα 20msec.

5. Θεωρώντας το κόστος της ζεύξης 5-6 ίσο με την τιμή που προσδιορίστηκε στο προηγούμενο ερώτημα, να υπολογιστεί η ελάχιστη μεταβολή τους κόστους μετάδοσης της ζεύξης 1-3, η οποία θα οδηγήσει στη συμμετοχή της ζεύξης αυτής στη μία εκ των παραπάνω δύο βέλτιστων διαδρομών για τις δύο μεταδόσεις και άρα στην τροποποίησή της; Επαληθεύστε την απάντησή σας πειραματικά με τη βοήθεια του NS2 και παραθέστε τα σχετικά στιγμιότυπα από το NAM.

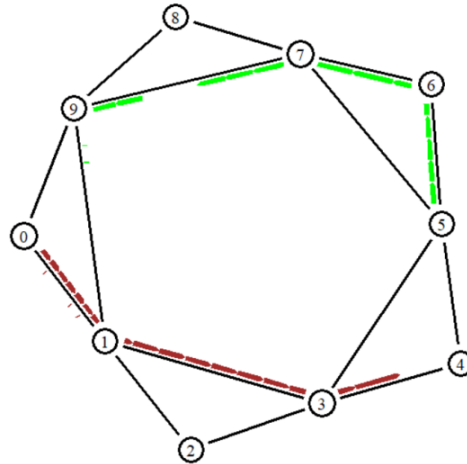
Αρχικά το κόστος της ζεύξης 5-6 διατηρείται στις 2 μονάδες αφού όπως αποδείχτηκε στο προηγούμενο ερώτημα, δεν μπορεί να αυξηθεί.



Παρατηρείται ότι, για τη ροή 1, η υπάρχουσα βέλτιστη διαδρομή έχει συνολικό κόστος 9 και η βέλτιστη διαδρομή με την ζεύξη 1-3 έχει συνολικό κόστος 10. Επίσης για τη ροή 2, η υπάρχουσα βέλτιστη διαδρομή έχει συνολικό κόστος 8 και η βέλτιστη διαδρομή με την ζεύξη 1-3 έχει συνολικό κόστος 10.

Επομένως, για τη συμμετοχή της ζεύξης 1-3 πρέπει να μειωθεί τουλάχιστον κατά 2 μονάδες ώστε να χρησιμοποιηθεί σίγουρα για τη ροή 1. Ακόμη, παρατηρείται ότι δεν χρησιμοποιείται από τη ροή 2 παρ' όλο που έχει συνολικό κόστος ίδιο με αυτό της διαδρομής που χρησιμοποιείται.

Ακολουθεί το στιγμιότυπο του nam:

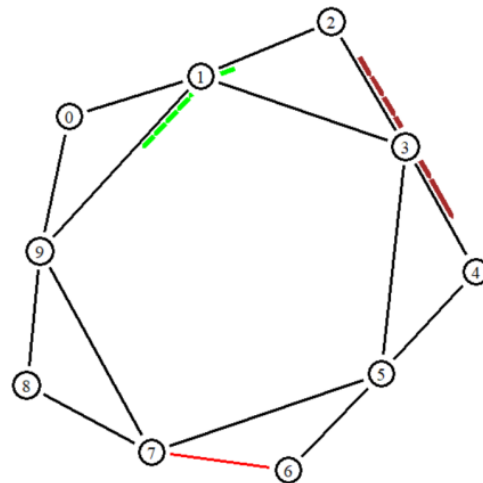


Αξιοσημείωτο είναι το γεγονός ότι παρά την μείωση του κόστους της ζεύξης 1-3 κατά 1 μονάδα δεν θα εξασφαλίζεται η χρησιμοποίηση της από τη ροή 1 διότι υπάρχει ισοβαθμία αλλά το πρωτόκολλο επιλέγει την προηγούμενη βέλτιστη διαδρομή κάτι το οποίο επαληθεύτηκε και με τη χρήση του NS2.

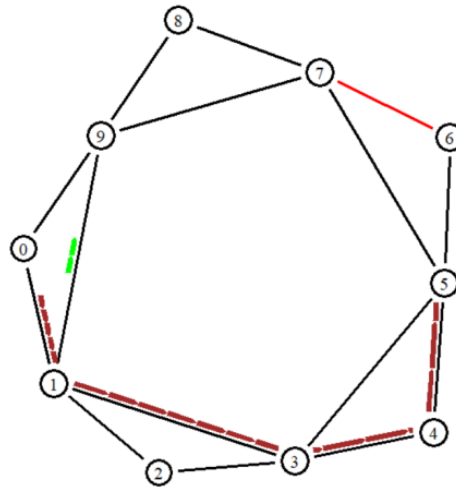
6. Την χρονική στιγμή 1 sec η γραμμή 6-7 τίθεται εκτός λειτουργίας. Ποιες διαδρομές ακολουθούν τώρα τα πακέτα για τις δύο ανωτέρω περιπτώσεις; Επαληθεύστε την απάντησή σας πειραματικά με τη βοήθεια του NS2 και παραθέστε τα σχετικά στιγμιότυπα από το NAM.

Προστίθεται το script: `$ns rtmodel-at 1.0 down $n(6) $n(7)`

Για το ερώτημα 4 (μετά την πτώση της ζεύξης 6-7) τα πακέτα δεδομένων της ροής 1 ακολουθούν την διαδρομή: 0→1→2→3→4→5 και τα πακέτα δεδομένων της ροής 2 ακολουθούν την διαδρομή: 3→2→1→9.

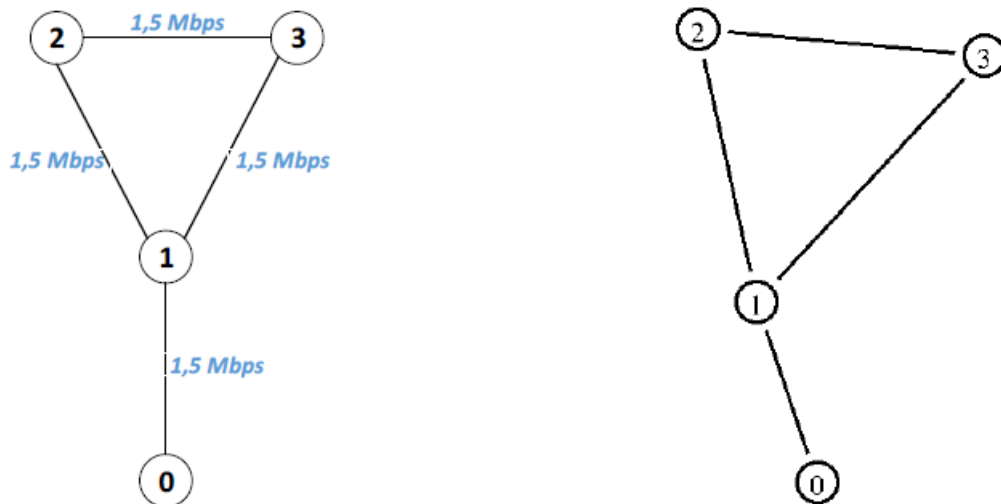


Για το ερώτημα 5 (μετά την πτώση της ζεύξης 6-7) τα πακέτα δεδομένων της ροής 1 ακολουθούν την διαδρομή: 0→1→3→4→5 και τα πακέτα δεδομένων της ροής 2 ακολουθούν την διαδρομή: 3→1→9.



### 3. Αποτυχία πρωτοκόλλου Distance Vector

Δημιουργείται η παρακάτω τοπολογία όπου όλες οι ζεύξεις είναι αμφίδρομες και έχουν ταχύτητα μετάδοσης 1,5 Mbps. Η καθυστέρηση μετάδοσης σε κάθε ζεύξη είναι ίση με 20 msec και οι ουρές αναμονής για όλες τις ζεύξεις είναι τύπου DropTail.



Ο κόμβος 2 αποστέλλει πακέτα UDP μεγέθους 540 Byte, ένα κάθε 4 msec (ροή 1 κίνησης CBR) προς τον κόμβο 0, ενώ ο κόμβος 3 αποστέλλει πακέτα UDP μεγέθους 540 Byte, ένα κάθε 4 msec (ροή 2 κίνησης CBR) προς τον κόμβο 1.

Η μετάδοση και για τις δύο ροές ξεκινά τη χρονική στιγμή  $t=0.4$  sec και τελειώνει τη χρονική στιγμή  $t=3.2$  sec. Θεωρείται ότι η προσομοίωση ολοκληρώνεται τη χρονική στιγμή  $t=3.8$  sec.

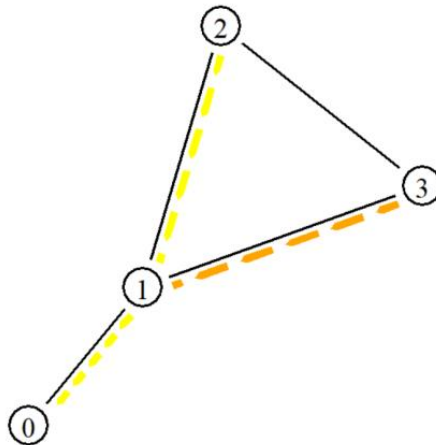
Θα πρέπει να χρησιμοποιηθεί διαδικασία καταγραφής (record) των δεδομένων της κίνησης για κάθε ροή.

Τη χρονική στιγμή  $t=1.5$  sec, η ζεύξη 0-1 τίθεται εκτός λειτουργίας, ενώ τη χρονική στιγμή  $t=2$  sec, αποκαθίσταται η λειτουργία της ζεύξης 0-1.

### 3.1 Ερωτήσεις

1. Ποια διαδρομή ακολουθούν τα πακέτα που στέλνει ο κόμβος 2 πριν και μετά τη διακοπή της ζεύξης 0-1; Εξηγήστε αυτή τη συμπεριφορά.

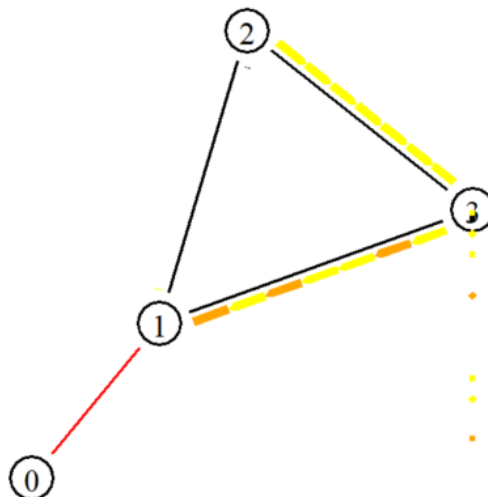
Πριν την διακοπή:



Η ροή 1 ( $2 \rightarrow 0$ ) ακολουθεί την διαδρομή:  $2 \rightarrow 1 \rightarrow 0$ .

Η ροή 2 ( $3 \rightarrow 1$ ) ακολουθεί την διαδρομή:  $3 \rightarrow 1$ .

Μετά την διακοπή:



Η ροή 1 ( $2 \rightarrow 0$ ) δεν δύναται να παραδώσει τα πακέτα στον κόμβο 0 και ακολουθεί αλλόκοτη διαδρομή στην προσπάθειά της να βρει λύση.

Η ροή 2 ( $3 \rightarrow 1$ ) ακολουθεί την διαδρομή:  $3 \rightarrow 1$ .

Αξίζει να σημειωθεί ότι η ασυνήθιστη διαδρομή που ακολουθεί η ροή 1 έχει αρχικά ως συνέπεια την προσωρινή αποθήκευση πακέτων που στοιβάζονται στις ουρές των κόμβων και τις ίδιες τις ζεύξεις. Το φαινόμενο απώλειας πακέτων παρατηρείται όταν αρχίσουν να γεμίζουν οι στοίβες. Αξίζει να σημειωθεί ότι παρ' όλο που η διαδρομή της ροής 2 δεν έχει αλλάξει, επηρεάζεται από την ροή 1 και παρατηρούνται και απώλειες δικών της πακέτων.

2. Να παραθέσετε τα διαγράμματα του *xgraph* που δείχνουν τον ρυθμό άφιξης δεδομένων στους προορισμούς τους. Σχολιάστε τους παρατηρούμενους ρυθμούς άφιξης δεδομένων για κάθε ροή.

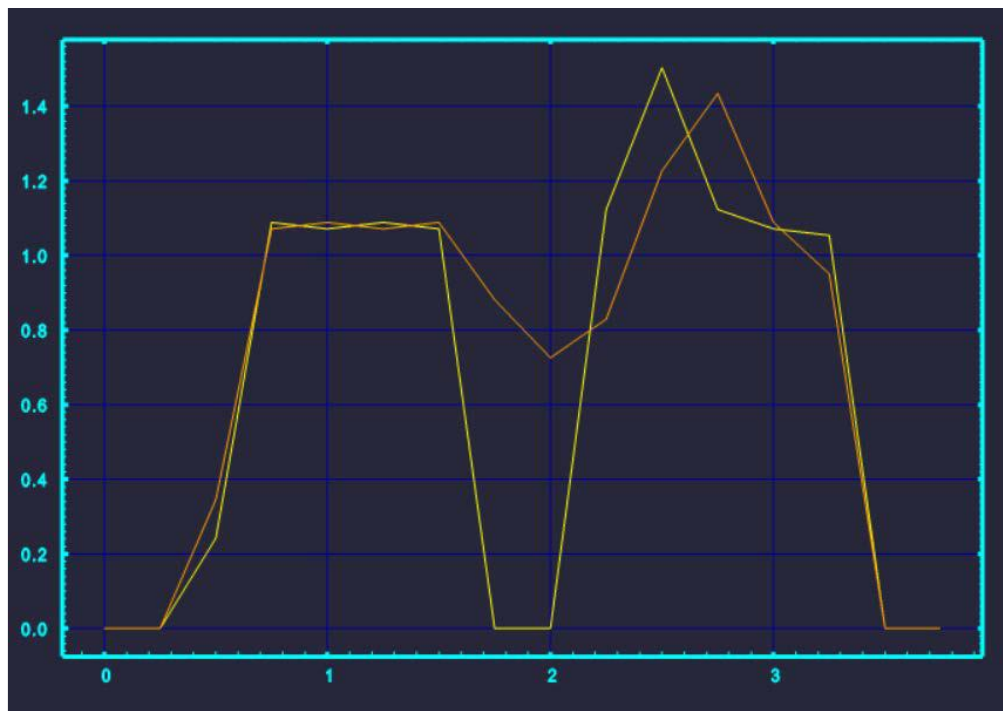
Για:

$t = 0.4 \text{ sec} - 0.75 \text{ sec}$ : ο ρυθμός μετάδοσης (PM) αυξάνεται μέχρι να σταθεροποιηθεί

$t = 0.75 \text{ sec} - 1.50 \text{ sec}$ : ο PM διατηρείται σχεδόν σταθερός στο 1Mbps

$t = 1.50 \text{ sec} - 2.00 \text{ sec}$ : ο PM πέφτει απότομα λόγω της πτώσης της ζεύξης 0-1. Για τη ροή 1 μηδενίζεται ενώ για τη ροή 2 έχει τιμή που μεταβάλλεται παραμένει όμως σταθερά κάτω από το 1Mbps

$t = 2.00 \text{ sec} - \text{τέλος}$ : ο PM αυξάνεται πάλι απότομα, ξεπερνά την προηγούμενη σταθερή τιμή και φτάνει στιγμιαία πάνω από τα 1.4Mbps. Έπειτα σταματά η αποστολή πακέτων έτσι ο PM φθίνει μέχρι που μηδενίζεται.



3. Επηρεάζεται η ροή πακέτων από τον κόμβο 3 προς στον 1 λόγω της αστοχίας της ζεύξης 0-1; Προκύπτει στο ερώτημα 1 και επιβεβαιώνεται στο ερώτημα 2 πως επηρεάζεται αρκετά η ροή 2 παρ' όλο που η διαδρομή της δεν μεταβάλλεται. Πρώτα, με τη πτώση της ζεύξης 0-1, διότι η χρησιμοποίηση της ζεύξης 3-1 από κίτρινα πακέτα προκαλεί μείωση του ρυθμού αποστολής των πορτοκαλί πακέτων. Εν τέλει, όταν συσσωρευθούν αρκετά πακέτα ώστε να γεμίσει η στοίβα του κόμβου 3 παρατηρείται ακόμα και απώλεια πορτοκαλί πακέτων.

4. Υπολογίστε το πλήθος των πακέτων που χάνεται για κάθε ροή, με τη βοήθεια του κατάλληλου script σε γλώσσα awk.

Από την εκτέλεση του awk προκύπτει:

```
Dropped packets on flow 1: 85
Dropped packets on flow 2: 17
```

Για την ροή 1: χάνονται 85 πακέτα.

Για την ροή 2: χάνονται 17 πακέτα.

Πηγαίος κώδικας:

```
BEGIN {  
  packets1 = 0;  
  packets2 = 0; }  
/^d/ && cbr/ {  
  if($8 == 1) {  
    packets1 ++ ;}  
  if($8 == 2) {  
    packets2 ++ ;}  
}  
END {  
  printf("Dropped Packets on flow 1: %d\n", packets1);  
  printf("Dropped Packets on flow 2: %d\n", packets2);}
```

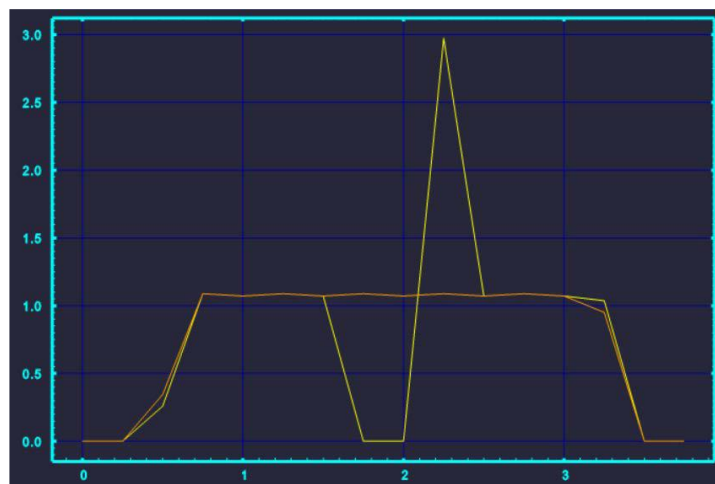
5. Προτείνετε κάποια λύση για την αντιμετώπιση αυτής της συμπεριφοράς.

Για την αντιμετώπιση αυτής της συμπεριφοράς θα μπορούσε να δημιουργηθεί μια εφεδρική ζεύξη 0-1 η οποία θα λαμβάνει τη θέση της κανονικής όταν τίθεται εκτός λειτουργίας. Ακόμη, μια άλλη λύση θα ήταν, το δίκτυο να ενημερώνει τον κόμβο 2 να σταματήσει να αποστέλλει πακέτα και να περιμένει ενημέρωση τότε να επανεκκινήσει την αποστολή για να μην παρατηρείται συσσώρευση πακέτων που δεν μπορούν να φτάσουν στον προορισμό τους. Μια πιο υλοποιήσιμη εναλλακτική λύση είναι το route poisoning ή η δρομολόγηση με link state.

6. Όσο η ζεύξη 0-1 είναι εκτός λειτουργίας, διαφοροποιείται καθόλου η κατάσταση εάν μεταβληθεί ο μέγιστος ρυθμός μετάδοσης που μπορεί να υποστηριχθεί από τις υπόλοιπες ζεύξεις; Αιτιολογείστε την απάντησή σας τόσο θεωρητικά, όσο και πειραματικά με τη βοήθεια του NS2.

Αν αυξηθεί ο ρυθμός μέγιστης μετάδοσης το δίκτυο είναι σε θέση να αποθηκεύσει μεγαλύτερο όγκο δεδομένων στις ζεύξεις του με αποτέλεσμα να διατηρείται ψηλά ο PM των πακέτων της ροής 2 και να μην συσσωρεύονται τόσο πολύ τα πακέτα της ροής 1 με αποτέλεσμα να μην χάνονται. Έτσι τα κίτρινα πακέτα έχουν τη δυνατότητα να φτάσουν στον προορισμό τους όταν αποκατασταθεί η βλάβη στη ζεύξη 0-1. Για επιβεβαίωση του πιο πάνω ισχυρισμού, τίθεται ο μέγιστος ρυθμός μετάδοσης στα 15Mbps (10πλάσιο από πριν) και εκτελείται το αρχείο awk και το xgraph με τα ακόλουθα αποτελέσματα:

```
Dropped packets on flow 1: 5  
Dropped packets on flow 2: 0
```



Παρατηρείται ότι ο PM της ροής 2 παραμένει αμετάβλητος, όμως χάνονται μόνο 5 κίτρινα πακέτα και κανένα πορτοκαλί.