

Ατομική Διπλωματική Εργασία

**ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ
ΔΙΑΜΕΡΙΣΜΟΥ ΠΑΚΕΤΩΝ ΣΕ ΠΟΛΛΑΠΛΑ ΜΟΝΟΠΑΤΙΑ
ΔΙΚΤΥΩΝ MPLS**

Ιωάννου Ιωσήφ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2013

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ ΔΙΑΜΕΡΙΣΜΟΥ
ΠΑΚΕΤΩΝ ΣΕ ΠΟΛΛΑΠΛΑ ΜΟΝΟΠΑΤΙΑ ΔΙΚΤΥΩΝ MPLS

Ιωάννου Ιωσήφ

Επιβλέπων Καθηγητής
Γεωργίου Χρύσης
Βασιλείου Βάσος

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2013

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τους επιβλέποντες καθηγητές μου κ. Γεωργίου Χρύση και κ. Βασιλείου Βάσο για την καθοδήγηση και τις συμβουλές τους που μου πρόσφεραν καθ' όλη τη διάρκεια της Διπλωματικής Εργασίας, καθώς και την ευκαιρία που μου έδωσαν να συνεργαστώ μαζί τους.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένεια μου που έδειξαν κατανόηση κατά την περίοδο στην οποία υλοποιούσα την Ατομική Διπλωματική Εργασία και μου ανέβαζαν το ηθικό.

Τέλος, ευχαριστώ πολύ τους φίλους μου Γεωργίου Κυριακή, Θωμά Παντελή, Δήμου Παναγιώτα, Χριστοφόρου Μαρία, Τεμενέ Νάταλη και Μιχαήλ Μάριο που ήταν κοντά μου και μου συμπαράσταθηκαν καθώς και τους συναδέλφους μου στο εργαστήριο Δικτύων για τις ευχάριστες στιγμές που περάσαμε μαζί.

.

Περίληψη

Η ανάγκη για προσφορά και υποστήριξη καλύτερων υπηρεσιών διαδικτύου οδήγησε στην ανάπτυξη και εφαρμογή νέων τεχνολογιών οι οποίες θα είναι σε θέση να καλύψουν τις αυξημένες ανάγκες. Ο μηχανισμός MPLS παρουσιάζετε ως η πιο αποδοτική και αποτελεσματική επιλογή για την αντιμετώπιση των σύγχρονων προκλήσεων όπως είναι η ταχύτητα, η ποιότητα υπηρεσίας, η διαχείριση εύρους ζώνης, η επεκτασιμότητα και η μηχανική κυκλοφορία. Το MPLS είναι μια τεχνική μεταγωγής που βασίζεται σε ετικέτες, η οποία μπορεί να εφαρμοσθεί για την εξυπηρέτηση κίνησης διαφόρων πρωτοκόλλων επιπέδου δικτύου.

Η παρούσα διπλωματική εργασία ασχολείται με την εξέταση του αλγορίθμου TSS (k,n) με πολλαπλά μονοπάτια των Sahel Alouneh, Anjali Agarwal & Abdeslam En-Nouaary, ο οποίος συνδυάζει ανοχή σφαλμάτων και μηχανική κυκλοφορία. Ο αλγόριθμος αυτός δρομολογεί πακέτα σε δίκτυο MPLS βασισμένο με δρομολόγηση πολλαπλών μονοπατιών και με ένα τροποποιημένο αλλά καθορισμένο Σχήμα Μοιρασμού Κατωφλίου (TSS - Threshold Sharing Scheme). Σκοπός της διπλωματικής εργασίας είναι να εξεταστεί κατά πόσο ο προτεινόμενος αλγόριθμος συνδυάζει αποτελεσματικά την ανοχή σφαλμάτων και τη μηχανική κυκλοφορία.

Τέλος γίνεται σύγκριση του προτεινόμενου αλγορίθμου με ένα υβριδικό αλγόριθμο, που αναπτύχθηκε σε προηγούμενη ερευνητική εργασία, ο οποίος συνδυάζει τους καλύτερους αλγόριθμους ανοχής σφαλμάτων και μηχανικής κυκλοφορίας. Για την αξιολόγηση των αλγορίθμων έγινε χρήση του προσομοιωτή Network Simulator 2. Τα αποτελέσματα της προσομοίωσης αναλύθηκαν και συγκρίθηκαν και προέκυψαν ενδιαφέροντα πλεονεκτήματα και μειονεκτήματα των δυο αλγορίθμων.

Περιεχόμενα

Κεφάλαιο 1	1
Εισαγωγή	1
1.1 Κίνητρο	1
1.2 Σκοπός	2
1.3 Μεθοδολογία	2
1.4 Δομή Ατομικής Διπλωματικής Εργασίας	3
Κεφάλαιο 2	4
Μηχανισμός MPLS	4
2.1 Εισαγωγή στο Μηχανισμό MPLS	4
2.2 Βασικές Έννοιες	5
2.3 Πρωτόκολλα Σηματοδότησης MPLS	6
2.3.1 Πρωτόκολλο Διανομής Ετικετών	7
2.3.2 Πρωτόκολλο Διανομής Ετικετών με Περιορισμούς στη Δρομολόγηση	8
2.3.2 Πρωτόκολλο Κατακράτησης Πόρων με Επεκτάσεις Εφαρμοσμένης Μηχανικής Κυκλοφορίας	8
2.4 Λειτουργία του Μηχανισμού MPLS	9
2.4.1 Δομή ενός MPLS Δικτύου	9
2.4.2 Δημιουργία Ετικέτας και Πίνακα Δρομολόγησης	10
2.4.3 Δημιουργία LSP και Δρομολόγηση Πακέτων	11
Κεφάλαιο 3	12
Μηχανική Κυκλοφορία	12
3.1 Εισαγωγή στη Μηχανική Κυκλοφορίας	12
3.2 Μηχανική Κυκλοφορία σε Δίκτυα MPLS	13
Κεφάλαιο 4	15
Ανοχή Σφαλμάτων	15
4.1 Εισαγωγή στην Ανοχή Σφαλμάτων	15
4.2 Βασικές Έννοιες	16
4.3 Ανίχνευση Σφαλμάτων σε MPLS Δίκτυο	17
4.4 Μηχανισμοί Αποκατάστασης Δικτύου MPLS	18
Κεφάλαιο 5	23
Ανασκόπηση Υβριδικού Αλγορίθμου	23
5.1 Εισαγωγή	23
5.2 Αλγόριθμοι Ανοχής Σφαλμάτων Υβριδικού	24
5.2.1 Αλγόριθμος Otel	24
5.2.2 Αλγόριθμος GONFA	28
5.3 Αλγόριθμος Μηχανικής Κυκλοφορίας Υβριδικού	31

5.3.1 Αλγόριθμος LSFBS	31
5.4 Περιγραφή Υβριδικού Αλγορίθμου	32
5.4.1 Ψευδοκώδικας Υβριδικού Αλγορίθμου	35
Κεφάλαιο 6	39
Αλγόριθμος Κατωφλίου (k,n) σε Πολλαπλά Μονοπάτια	39
6.1 Εισαγωγή	39
6.2 Τεμαχισμός του Αρχικού Πακέτου σε Πολλαπλά Υπό-πακέτα	40
6.3 Ανακατασκευή του Αρχικού Πακέτου από τα Υπό-πακέτα	42
6.4 Πληροφορίες Δικτύου MPLS για την Υλοποίηση του Αλγορίθμου Κατωφλίου	45
6.4.1 Εύρεση n Ανεξάρτητων Διαδρομών Δρομολόγησης	45
6.4.2 Καθορισμός Ροών Δρομολόγησης και Buffer Τελικού Κόμβου	46
6.4.3 Καθορισμός Επικεφαλίδας Υπό-πακέτων	48
Κεφάλαιο 7	50
Πειραματική Αξιολόγηση	50
7.1 Προσομοιωτής Network Simulator 2	50
7.2 Παράμετροι Προσομοίωσης	52
7.2.1 Τοπολογίες Δικτύων Προσομοίωσης	52
7.2.2 Πληροφορίες Δικτύου	54
7.2.3 Σενάρια	55
7.3 Παράμετροι Αξιολόγησης	59
7.4 Αποτελέσματα Προσομοίωσης	60
7.4.1 Γραφικές Παραστάσεις για το Σενάριο 1	60
7.4.2 Γραφικές Παραστάσεις για το Σενάριο 2	64
7.4.3 Γραφικές Παραστάσεις για το Σενάριο 3	69
7.4.4 Γραφικές Παραστάσεις για το Σενάριο 4	73
7.4.2 Γραφικές Παραστάσεις για το Σενάριο 5	77
7.4.2 Γραφικές Παραστάσεις για το Σενάριο 6	80
7.5 Συμπεράσματα	83
Κεφάλαιο 8	89
Συμπεράσματα	89
8.1 Γενικά Συμπεράσματα	89
8.2 Μελλοντική Εργασία	91
Βιβλιογραφία	93
Παράρτημα Α	A-1
Παράρτημα Β	B-1

Κεφάλαιο 1

Εισαγωγή

1.1 Κίνητρο	1
1.2 Σκοπός	2
1.3 Μεθοδολογία	2
1.4 Δομή Ατομικής Διπλωματικής Εργασίας	3

1.1 Κίνητρο

Η ραγδαία ανάπτυξη του διαδικτύου είχε ως αποτέλεσμα την αύξηση της ζήτησης δεδομένων που μεταφέρονται στα δίκτυα αλλά και τη δημιουργία σφαλμάτων κατά μήκος των διαδρομών στο δίκτυο με αποτέλεσμα η ποιότητα των υπηρεσιών και κυκλοφορίας των δεδομένων να μειώνεται. Πολλαπλά σφάλματα στο δίκτυο μπορούν να συμβάλλουν στην κατάρρευση του δικτύου και στις εφαρμογές πραγματικού χρόνου οι απώλειες πακέτων ή οι μεγάλες καθυστερήσεις να επηρεάζουν αρνητικά στην διεκπεραίωση τους, αλλοιώνοντας έτσι και τα αποτελέσματα τους.

Υπάρχει ανάγκη από μηχανισμούς που να αντιμετωπίζουν τα σφάλματα που εμφανίζονται σε ένα δίκτυο καθώς και στην αξιοποίηση περισσότερων πόρων για τη δρομολόγηση των πακέτων έτσι ώστε τα δίκτυα να λειτουργούν με τον πιο αποδοτικό και αξιόπιστο τρόπο. Με αποτέλεσμα τα δεδομένα που μεταφέρονται ανά δευτερόλεπτο

μέσα στο δίκτυο θα φτάνουν στον προορισμό τους με τη μικρότερη δυνατή καθυστέρηση χωρίς να χάνονται.

Ένας από τους μηχανισμούς κυκλοφορίας, ο οποίος λειτουργεί με αποδοτικό και λειτουργικό τρόπο, είναι ο μηχανισμός MPLS [1] [2]. Η προσπάθεια για καλύτερη χρήση των πόρων του δικτύου και για εγγυήσεις όσο αφορά την παροχή ποιότητας υπηρεσιών (π.χ. βίντεο, ομιλία) μαζί με την ανάγκη για γρήγορη επαναφορά του δικτύου σε περίπτωση σφάλματος είναι μερικά από τα βασικά κίνητρα που οδήγησαν στην προσφυγή χρησιμοποίησης του μηχανισμού MPLS.

1.2 Σκοπός

Σκοπός της παρούσας διπλωματικής εργασίας είναι η μελέτη και υλοποίηση του αλγόριθμου διαμερισμού πακέτου σε πολλαπλά μονοπάτια [3]. Ο αλγόριθμος αυτός προσομοιώθηκαν σε τοπολογίες MPLS δικτύων χρησιμοποιώντας το εργαλείο προσομοίωσης δικτύων NS2. Στη συνέχεια τα αποτελέσματα που προέκυψαν από τις προσομοιώσεις επεξεργάστηκαν για τον καθορισμό του κατά πόσο ο αλγόριθμος αυτός συνδυάζει αποτελεσματικά τη μηχανική κυκλοφορίας και την ανοχή σφαλμάτων. Μέρος της αξιολόγησης του προτεινόμενου αλγόριθμου είναι και η σύγκριση του με τον υβριδικό αλγόριθμο (hybrid algorithm) [4], ο οποίος υλοποιήθηκε σε προηγούμενη ερευνητική εργασία και αποτελεί συνδυασμό των καλύτερων αλγόριθμων για την ανοχή σφαλμάτων και μηχανικής κυκλοφορίας.

1.3 Μεθοδολογία

Για την εκπόνηση αυτής της ερευνητικής εργασίας ακολουθήθηκε η εξής μεθοδολογία. Αρχικά έγινε μελέτη για τις βασικές έννοιες ενός MPLS δικτύου, τη βασική δομή του, τον τρόπο δρομολόγησης των δεδομένων και τις υπηρεσίες προστασίας στις συνδέσεις και στους κόμβους που προσφέρει ο μηχανισμός MPLS. Επιπλέον μελετήθηκαν οι τεχνικές Μηχανικής Κυκλοφορίας και Ανοχής Σφαλμάτων και με ποιο τρόπο μπορούν να εφαρμοστούν σε ένα δίκτυο MPLS. Ακολούθως μελετήθηκε ο υβριδικός αλγόριθμος, ο οποίος υλοποιήθηκε σε προηγούμενη ερευνητική εργασία, και ο αλγόριθμος κατωφλίου σε πολλαπλά μονοπάτια. Για την

πειραματική αξιολόγηση χρησιμοποιήθηκε ο προσομοιωτής δικτύων Network Simulator 2, όπου χρειάστηκε να γίνει μάθηση του εργαλείου. Τέλος έγινε η πειραματική αξιολόγηση και των δυο αλγορίθμων στον προσομοιωτή NS2 και από τα αποτελέσματα που προέκυψαν συγκρίθηκαν οι δυο αλγόριθμοι (εξαγωγή συμπερασμάτων).

1.4 Δομή Ατομικής Διπλωματικής Εργασίας

Στο Κεφάλαιο 2 γίνεται μια σύντομη εισαγωγή στο μηχανισμό MPLS και επεξήγηση των βασικών ορισμών που συσχετίζονται με το MPLS. Τέλος αναλύεται ο τρόπος λειτουργίας του.

Στο Κεφάλαιο 3 γίνεται εισαγωγή στη μηχανική κυκλοφορία και πώς υλοποιείται στο μηχανισμό MPLS. Επίσης καθορίζονται τα κριτήρια αξιολόγησης των αλγορίθμων μηχανικής κυκλοφορίας.

Το Κεφάλαιο 4 αναφέρεται για την ανοχή σφαλμάτων και παρουσιάζονται οι μηχανισμοί αποκατάστασης των δικτύων.

Στο Κεφάλαιο 5 γίνεται ανασκόπηση του Υβριδικού αλγορίθμου.

Στο Κεφάλαιο 6 παρουσιάζεται ο TSS και γίνεται αναφορά στον τρόπο που δρομολογεί τα πακέτα σε ένα δίκτυο MPLS.

Στο Κεφάλαιο 7 περιγράφεται το εργαλείο προσομοίωσης, οι παράμετροι αξιολόγησης των δυο αλγορίθμων καθώς και οι τοπολογίες των δικτύων που προσομοιώθηκαν (γίνεται αναλυτική αναφορά στα σενάρια της κάθε τοπολογίας). Επίσης γίνεται η παρουσίαση των αποτελεσμάτων κάθε σεναρίου για κάθε τοπολογία (μέσω γραφικών παραστάσεων).

Τέλος το Κεφάλαιο 8 ασχολείται με την ανάλυση των αποτελεσμάτων (τελικά συμπεράσματα) και περιγραφή της μελλοντικής εργασίας που μπορεί να συμπληρώσει το συγκεκριμένο θέμα μηχανικής κυκλοφορίας και ανοχής σφαλμάτων σε δίκτυα MPLS.

Κεφάλαιο 2

Μηχανισμός MPLS

2.1 Εισαγωγή στο Μηχανισμό MPLS	4
2.2 Βασικές Έννοιες	5
2.3 Πρωτόκολλα Σηματοδότησης MPLS	6
2.4 Λειτουργία του Μηχανισμού MPLS	9

2.1 Εισαγωγή στο Μηχανισμό MPLS

Ο μηχανισμός MPLS (Multi-Protocol Label Switching) είναι ένα πρωτόκολλο δρομολόγησης δεδομένων, το οποίο συνδυάζει τη διαχείριση του εύρου ζώνης (bandwidth) και τις απαιτήσεις εξυπηρέτησης των IP δικτύων [5] [6] [1] [7].

Στο παραδοσιακό μοντέλο OSI δικτύωσης το επίπεδο 2 (Link Layer) καλύπτει τα πρωτόκολλα Ethernet και SONET, τα οποία μεταφέρουν IP πακέτα μόνο σε απλά δίκτυα (LAN ή point-to-point WAN) ενώ το επίπεδο 3 (Network Layer) καλύπτει τη δρομολόγηση των πακέτων στις διευθύνσεις του διαδικτύου χρησιμοποιώντας το IP πρωτόκολλο. Ο μηχανισμός MPLS βρίσκεται μεταξύ των δυο αυτών επιπέδων προσφέροντας επιπλέον εφαρμογές για τη δρομολόγηση των δεδομένων δια μέσου του δικτύου. Γι' αυτό μπορεί να χαρακτηριστεί και ως το πρωτόκολλο δικτύου επιπέδου "2.5". Ως μηχανισμός στηρίζεται στη τεχνική ανταλλαγής ετικέτας, όπου σε κάθε πακέτο του δικτύου προστίθεται μια μοναδική ετικέτα και με βάση αυτή γίνεται η

δρομολόγηση του πακέτου. Η κάθε ετικέτα περιέχει όλες τις απαραίτητες πληροφορίες που χρειάζεται ένα δρομολογητής (ο οποίος υποστηρίζει MPLS) για να προωθήσει το πακέτο δεδομένων δια μέσο του δικτύου. Το κάθε πακέτο αποκτά μια μοναδική ετικέτα με την είσοδο του στο δίκτυο τύπου MPLS και αφαιρείται με την άφιξη του στον τελευταίο κόμβο. Οι πληροφορίες κάθε ετικέτας καθορίζουν τον επόμενο αποδέκτη του πακέτου καθώς και για άλλες υπηρεσίες.

2.2 Βασικές Έννοιες

Για καλύτερη κατανόηση της λειτουργίας ενός MPLS δικτύου δίνονται σε συντομία οι σημαντικότεροι ορισμοί [5].

Δίκτυο MPLS: Δίκτυο που αποτελείται από δρομολογητές, οι οποίοι υποστηρίζουν MPLS.

Label Switch Router (LSR): Είναι ο δρομολογητής, ο οποίος υποστηρίζει MPLS.

Ingress Label Switch Router: Είναι ο δρομολογητής ο οποίος βρίσκεται στην αρχή του MPLS δικτύου (είναι ο πρώτος δρομολογητής που δέχεται το πακέτο όταν αυτό εισέρχεται σε MPLS δίκτυο) και ενώνεται με δρομολογητές που δεν υποστηρίζουν MPLS.

Egress Label Switch Router: Είναι ο δρομολογητής ο οποίος βρίσκεται στο τέλος του MPLS δικτύου και ενώνεται με δρομολογητές που δεν υποστηρίζουν MPLS.

MPLS Label: Η ετικέτα (label) είναι μια μικρή επικεφαλίδα που χρησιμοποιείται για την προώθηση των πακέτων. Χρησιμοποιούνται μόνο τοπικά στα δίκτυα που υποστηρίζουν MPLS, κυρίως για τη δρομολόγηση των πακέτων. Με την εισαγωγή ενός πακέτου στο δίκτυο, ο αρχικός κόμβος αντί να βρει τον επόμενο κόμβο δρομολόγησης (next hop) βρίσκει τον τελικό κόμβο και δημιουργείται η επικεφαλίδα (label) με την διαδρομή που θα ακολουθήσει το πακέτο μέχρι να φτάσει στον τελικό κόμβο. Με μικρό σταθερό μέγεθος (συνολικά 32 bits) δεν επιβαρύνει το δίκτυο με επιπλέον φορτίο

δεδομένων και διαγράφεται μόλις το πακέτο εξέλθει του MPLS δικτύου. Η ετικέτα περιλαμβάνει τα ακόλουθα πεδία:

- ▲ Label (20 bits): Περιλαμβάνει την πραγματική τιμή της MPLS ετικέτας
- ▲ Exp (3 bits): Μπορεί να επηρεάζει τους αλγόριθμους δρομολόγησης του πακέτου κατά τη διάρκεια μετάδοσης του μέσα στο MPLS δίκτυο.
- ▲ S (1 bit): Υποστηρίζει ιεραρχική στοίβα ετικετών
- ▲ TTL (8 bits): Time to Live – Παρέχει τη συμβατική IP TTL

Label Switch Path (LSP): Το μονοπάτι (διαδρομή) που καθορίζεται από όλες τις ετικέτες που ανατίθενται μεταξύ δύο τελικών σημείων.

Label Distribution Protocol: Πρωτόκολλο που χρησιμοποιείται για την κατανομή των ετικετών μεταξύ των LSRs.

Forwarding Equivalence Class (FEC): Είναι ένα σύνολο από πακέτα τα οποία έχουν παρόμοια ή/και τα ίδια χαρακτηριστικά και δρομολογούνται με τον ίδιο τρόπο.

2.3 Πρωτόκολλα Σηματοδότησης MPLS

Η χρησιμοποίηση ενός LSP απαιτεί τη επικοινωνία των δρομολογητών, οι οποίοι θα αποτελούν το μονοπάτι στο οποίο θα δρομολογηθεί το πακέτο. Για να επιτευχθεί η επικοινωνία μεταξύ των δρομολογητών χρησιμοποιούνται τα πρωτόκολλα σηματοδότησης έτσι ώστε να μπορούν να ανταλλάξουν πληροφορίες μεταξύ τους. Με τον καθορισμό των ετικετών σε όλους τους MPLS δρομολογητές δημιουργούνται οι κατάλληλοι πίνακες και η δρομολόγηση πλέον στηρίζεται μόνο από τις ετικέτες [8]. Πιο κάτω παρουσιάζονται τα βασικά πρωτόκολλα δρομολόγησης:

- ▲ Πρωτόκολλο διανομής ετικετών (Label Distribution Protocol - LDP)
- ▲ Πρωτόκολλο διανομής ετικετών με περιορισμούς στη δρομολόγηση (Constraint Based LDP - CRLDP)
- ▲ Πρωτόκολλο κατακράτησης πόρων με επεκτάσεις εφαρμοσμένης μηχανικής για την κυκλοφορία (RSVP for Traffic Engineering – RSVP-TE)

2.3.1 Πρωτόκολλο Διανομής Ετικετών

Το πρωτόκολλο διανομής ετικετών χρησιμοποιείται για την ανταλλαγή ετικετών μεταξύ γειτονικών κόμβων, οι οποίοι θα δημιουργήσουν τη LSP για τη δρομολόγηση των πακέτων. Με τη δημιουργία του LSP γίνεται η σύνδεση του επίσης με ένα FEC, το οποίο διευκρινίζει ποια πακέτα συσχετίζονται με το συγκεκριμένο μονοπάτι ανταλλαγής ετικετών. Τα είδη των μηνυμάτων που χρησιμοποιούνται για την ανταλλαγή των ετικετών μεταξύ δυο κόμβων δρομολογητών είναι:

- ▲ **Μηνύματα ανακάλυψης:** Τα μηνύματα ανακάλυψης χρησιμοποιούνται για να γνωστοποιήσει ένας κόμβος την παρουσία του στο δίκτυο (μηνύματα χαιρετισμού).
- ▲ **Μηνύματα σύνδεσης:** Τα μηνύματα σύνδεσης χρησιμοποιούνται για τη σύνδεση, την εγκατάσταση και τον τερματισμό της επικοινωνίας μεταξύ δύο δρομολογητών. Όταν εγκατασταθεί επικοινωνία μεταξύ δύο δρομολογητών, ο κάθε κόμβος είναι υπεύθυνος για την ενημέρωση του άλλου δρομολογητή ότι ακόμα είναι ενεργός για να διατηρηθεί η σύνδεση μεταξύ τους. Σε περίπτωση που δεν υπάρχουν άλλα δεδομένα για να σταλούν από τον ένα κόμβο στον άλλο, στέλλεται μήνυμα για να τερματιστεί η υφιστάμενη σύνδεση μεταξύ τους.
- ▲ **Μηνύματα ανακοίνωσης:** Τα μηνύματα ανακοίνωσης χρησιμοποιούνται για τη δημιουργία, τροποποίηση και διαγραφή των ετικετών, οι οποίες είναι συσχετισμένες με ροές FECs.
- ▲ **Μηνύματα ενημέρωσης:** Τα μηνύματα ενημέρωσης χρησιμοποιούνται για την παροχή διάφορων πληροφοριών στους δρομολογητές LSR. Χωρίζονται στα μηνύματα λάθους και στα μηνύματα συμβουλής. Τα μηνύματα λάθους στέλλονται στις περιπτώσεις που κάποιος δρομολογητής LSR παύει να λειτουργεί έτσι ώστε οι υπόλοιποι δρομολογητές να τερματίσουν τη σύνδεση μαζί του. Ενώ με τα μηνύματα συμβουλής παρέχουν πληροφορίες στον LSR κόμβο για την LDP σύνδεση.

2.3.2 Πρωτόκολλο Διανομής Ετικετών με Περιορισμούς στη Δρομολόγηση

Το πρωτόκολλο διανομής ετικετών με περιορισμούς στη δρομολόγηση (CR-LDP) είναι μια επέκταση του απλού LDP πρωτοκόλλου. Αρχικά το CR-LDP πρωτόκολλο σημειώνει τους πόρους που απαιτούνται από το μονοπάτι δρομολόγησης των πακέτων για κάθε βήμα της διαδρομής. Αν μια διαδρομή δεν έχει τους απαιτούμενους πόρους, τότε αποδεσμεύονται για τη χρησιμοποίησή τους από κάποια άλλη διαδρομή που τους χρειάζεται. Το πρωτόκολλο CR-LDP εισάγει τα LSPID μηνύματα, τα οποία επιτρέπουν τον καθορισμό των κόμβων που θα ακολουθήσει ένα πακέτο σε μια διαδρομή. Έτσι γίνεται η εγκατάσταση ενός LSP βασισμένο στους περιορισμούς ποιότητας, υπηρεσίας και εύρους ζώνης.

2.3.2 Πρωτόκολλο Κατακράτησης Πόρων με Επεκτάσεις Εφαρμοσμένης Μηχανικής Κυκλοφορίας

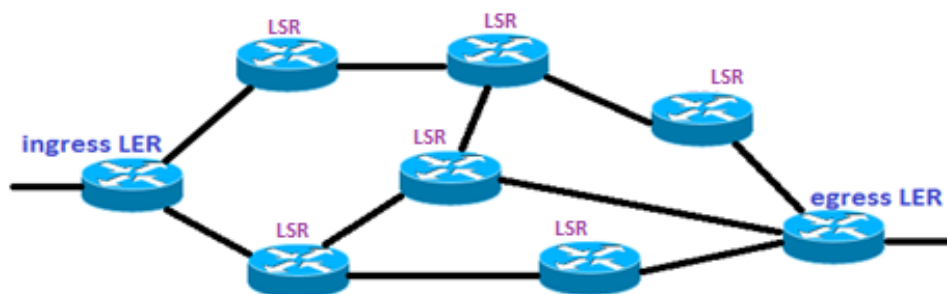
Το πρωτόκολλο κατακράτησης πόρων με επεκτάσεις εφαρμοσμένης μηχανικής για την κυκλοφορία (RSVP-TE) είναι μια επέκταση του πρωτοκόλλου κράτησης πόρων (RSVP) [9] [10] και εστιάζεται στη μηχανική κυκλοφορία. Υποστηρίζει την κράτηση πόρων σε ένα IP δίκτυο και οι εφαρμογές που υπάρχουν σε ένα IP δίκτυο μπορούν να χρησιμοποιήσουν αυτό τον αλγόριθμο έτσι ώστε να υποδείξουν στους άλλους κόμβους τα χαρακτηριστικά τους (όπως το εύρος ζώνης, το μέγιστο αριθμό πακέτων που μπορούν να σταλούν στο δίκτυο χωρίς αυτό να καταρρεύσει, κλπ). Η εγκαθίδρυση ενός LSP γίνεται με τον ίδιο τρόπο όπως γίνεται και στο απλό πρωτόκολλο κράτησης πόρων με τη διαφορά ότι στο RSVP-TE οι αναθέσεις των ετικετών γίνεται μέσω των μηνυμάτων Resv και Path. Τα μηνύματα Path στέλλονται κατά τη διάρκεια της εγκατάστασης του νέου μονοπατιού, όπου οι δρομολογητές που θα αποτελούν το νέο LSR να ενημερώνουν τους υπόλοιπους για την κατάσταση τους. Επίσης το RSVP-TE προσφέρει την επιλογή για τη δέσμευση των πόρων σε ένα LSP και αυτό επιτυγχάνεται με τα Resv μηνύματα.

2.4 Λειτουργία του Μηχανισμού MPLS

Μέσα από τη λειτουργία του μηχανισμού MPLS θα αναλύσουμε τη βασική δομή ενός MPLS δικτύου (τα βασικά μέρη που αποτελούν ένα δίκτυο MPLS), πως δημιουργείται μια ετικέτα και τι περιέχει ένας πίνακας δρομολόγησης καθώς και πως δημιουργείται ένα LSP και δρομολογούνται τα πακέτα μέσα στο δίκτυο [5] [11].

2.4.1 Δομή ενός MPLS Δικτύου

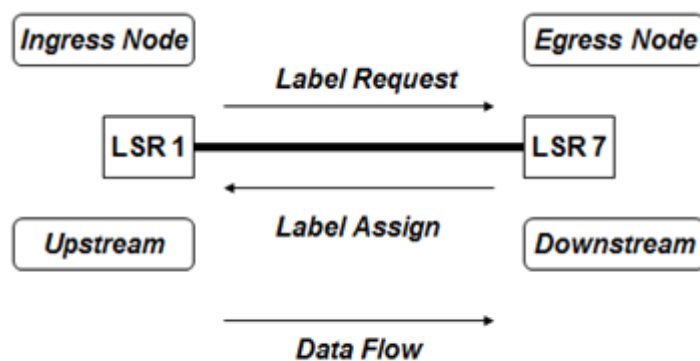
Το MPLS δίκτυο αποτελείται από δύο κυρίως τμήματα, τον πυρήνα και τα άκρα του δικτύου MPLS. Τα άκρα του δικτύου αποτελούνται από τους δύο κόμβους που βρίσκονται στην αρχή και στο τέλος του δικτύου και έχουν ως γείτονες του κόμβους του πυρήνα και τους κόμβους οι οποίοι δεν υποστηρίζουν MPLS. Ενώ ο πυρήνας αποτελείται μόνο από κόμβους που έχουν ως γείτονες άλλους κόμβους που υποστηρίζουν μόνο MPLS. Οι δρομολογητές στα άκρα του MPLS δικτύου ονομάζονται δρομολογητές των άκρων (Label Edge Routers - LER). Πιο συγκεκριμένα ο αρχικός δρομολογητής, ο κόμβος ο οποίος βρίσκεται στην είσοδο του δικτύου, ονομάζεται κόμβος εισόδου (ingress LER) και ο δρομολογητής που βρίσκεται στο τέλος του MPLS δικτύου ονομάζεται κόμβος εξόδου (egress LER). Οι δρομολογητές του πυρήνα ονομάζονται LSR μετάβασης (transit nodes). Ένα απλό παράδειγμα ενός MPLS δικτύου και τα τμήματα στα οποία χωρίζεται παρουσιάζεται στην πιο κάτω εικόνα (Εικόνα 2.4.1.1).



Εικόνα 2.4.1.1 - Δομή MPLS δικτύου

2.4.2 Δημιουργία Ετικέτας και Πίνακα Δρομολόγησης

Πριν τη δρομολόγηση των πακέτων μέσα στο δίκτυο οι δρομολογητές δεσμεύουν μια ετικέτα για να δημιουργήσουν τους πίνακες δρομολόγησης και να δεσμεύσουν FEC ετικέτα. Χρησιμοποιώντας το πρωτόκολλο LDP οι δρομολογητές διανέμουν τις ετικέτες κατά μήκος του δικτύου (για τη διανομή των ετικετών χρησιμοποιούνται μόνο οι downstream δρομολογητές, δηλαδή τα δεδομένα κατευθύνονται από τους δρομολογητές που βρίσκονται πιο κοντά στον ingress κόμβο προς αυτούς που καταλήγουν στον egress κόμβο – Εικόνα 2.4.2.1) [12]. Κατά τη διάρκεια της επικοινωνίας μεταξύ των LSR δημιουργείται ο πίνακας δρομολόγησης (Routing Table ή Label Information Base - LIB), ο οποίος περιέχει πληροφορίες σχετικά με τις ετικέτες και του FEC (πιο συγκεκριμένα οι πληροφορίες που περιέχει ο πίνακας δρομολόγησης είναι ο κόμβος εισδοχής (input port), η ετικέτα εισόδου (input label), ο προορισμός (destination), κόμβος εξόδου (output port), ετικέτα εξόδου (output label)). Σχετικό παράδειγμα ενός πίνακα δρομολόγησης παρουσιάζεται στην εικόνα 2.4.2.2.



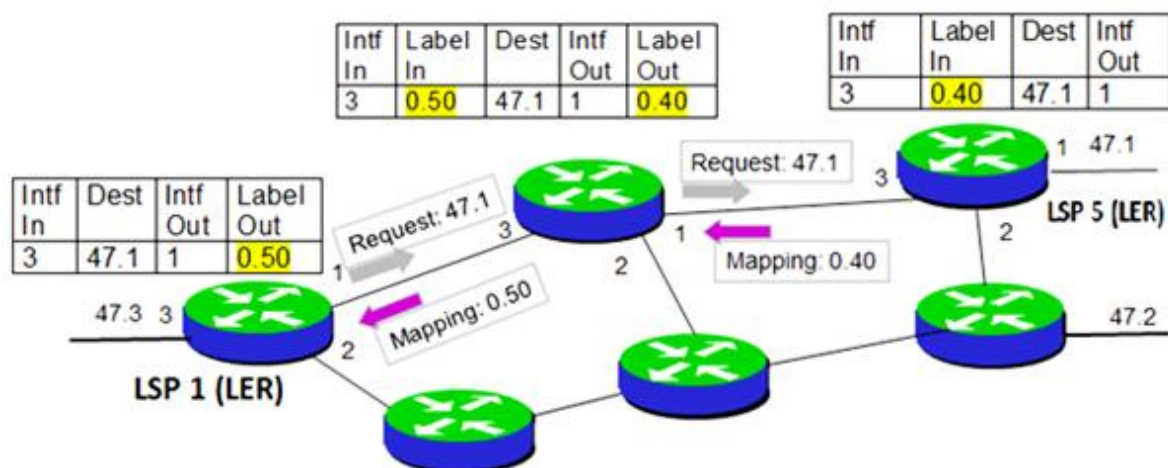
Εικόνα 2.4.2.1 - Upstream & Downstream δρομολογητές[12]

Intf In	Label In	Dest	Intf Out	Label Out
3	0.50	47.1	1	0.40

Εικόνα 2.4.2.2 - Πίνακας Δρομολόγησης

2.4.3 Δημιουργία LSP και Δρομολόγηση Πακέτων

Όπως προαναφέραμε προηγουμένως ο πίνακας δρομολόγησης περιέχει πληροφορίες σχετικά με τη δρομολόγηση του πακέτου (άφιξη και προορισμό). Έτσι όταν πακέτο εισέλθει στο MPLS δίκτυο, το μονοπάτι (LSP) που θα ακολουθήσει καθορίζεται από τις ετικέτες που ανατίθενται μεταξύ των δυο σημείων άφιξης και προορισμού. Χρησιμοποιώντας την εικόνα 2.4.3.1 θα εξηγήσουμε πως γίνεται η δρομολόγηση ενός πακέτου όταν εισέλθει στο δίκτυο. Όταν το πακέτο φτάσει στον κόμβο εισδοχής (LSP 1 - LER), ο δρομολογητής αυτός θα χρησιμοποιήσει τον πίνακα δρομολόγησης για να βρει τον επόμενο κόμβο στον οποίο πρέπει να σταλεί το πακέτο για το συγκεκριμένο FEC (στο πακέτο προστίθεται μια ετικέτα η οποία περιέχει πληροφορίες για τον προορισμό που πρέπει να φτάσει το πακέτο). Οι ενδιάμεσοι κόμβοι που θα χρησιμοποιηθούν για τη δρομολόγηση του πακέτου από τον αρχικό στον τελικό κόμβο θα χρησιμοποιούν την ετικέτα για να βρουν τον επόμενο δρομολογητή που πρέπει να παραλάβει το πακέτο μέχρι να φτάσει στον τελικό. Όταν το πακέτο φτάσει στον κόμβο εξόδου (LSP 5 - LER) η ετικέτα που του προστέθηκε στον κόμβο εισδοχής διαγράφεται και το πακέτο πλέον δρομολογείται ανάλογα με το πρωτόκολλο δρομολόγησης που χρησιμοποιείται από τον επόμενο κόμβο.



Εικόνα 2.4.3.1 - MPLS Label Distribution [12]

Κεφάλαιο 3

Μηχανική Κυκλοφορία

3.1 Εισαγωγή στη Μηχανική Κυκλοφορίας	12
3.2 Μηχανική Κυκλοφορία σε Δίκτυα MPLS	13

3.1 Εισαγωγή στη Μηχανική Κυκλοφορίας

Με την διάδοση του διαδικτύου αυξάνεται και η ζήτηση πληροφορίας από τους χρήστες με αποτέλεσμα πολλές φορές να υπάρχει αρκετή συμφόρηση σε μερικές ροές δρομολόγησης σε μια συγκεκριμένη περιοχή του δικτύου. Η μηχανική κυκλοφορίας (traffic engineering) [13] είναι η τεχνολογία που έχει ως απώτερο σκοπό τη βελτιστοποίηση της απόδοσης του δικτύου. Στις περιοχές που υπάρχει συμφόρηση λόγω μεγάλης διακίνησης δεδομένων, η μηχανική κυκλοφορίας ως τεχνική μετακινεί την κυκλοφορία του συγκεκριμένου σημείου στο δίκτυο σε άλλους συνδέσμους που έχουν διαθέσιμους πόρους. Με αυτό επιτυγχάνεται η αποδοτική χρήση όλων των διαθέσιμων πόρων στον πυρήνα του δικτύου.

Κύριο χαρακτηριστικό της μηχανικής κυκλοφορίας είναι η ανακατεύθυνση της ροής που τείνει να παρουσιαστεί συμφόρηση έτσι ώστε να αποφευχθεί η συμφόρηση και να αυξηθεί η αποδοτικότητα της. Έτσι προσφέρονται καλύτερες υπηρεσίες (καλύτερος έλεγχος και διαχείριση των ροών) στο δίκτυο και εφαρμόζει τον αλγόριθμο

του κοντινότερου μονοπατιού για τις ροές δρομολόγησης. Η μηχανική κυκλοφορία μπορεί να διαχωριστεί σε δυο είδη:

- ▲ Βελτιστοποίησης χρήσης των πόρων του δικτύου
- ▲ Βελτιστοποίησης της κυκλοφορίας στο δίκτυο

Η πρώτη κατηγορία έχει ως στόχο στη βελτιστοποίηση της χρήσης των πόρων του δικτύου ώστε να χρησιμοποιούνται αξιόπιστα και ισάξια από όλες τα τμήματα του δικτύου. Ενώ η δεύτερη κατηγορία έχει ως στόχο την ενίσχυση της ποιότητας των υπηρεσιών των ροών, την ελαχιστοποίηση της απώλειας και της καθυστέρησης των πακέτων καθώς και την αύξηση των δεδομένων που μεταφέρονται ανά δευτερόλεπτο.

3.2 Μηχανική Κυκλοφορία σε Δίκτυα MPLS

Η μηχανική κυκλοφορία εφαρμόζεται σε MPLS δίκτυα χρησιμοποιώντας το πρωτόκολλο κατακράτησης πόρων με επεκτάσεις εφαρμοσμένης μηχανικής (RSVP-TE) [9] [7] [14] [15] [2]. Με το πρωτόκολλο αυτό γίνεται η δέσμευση των πόρων που χρειάζονται για τη δρομολόγηση των πακέτων κατά μήκος του δικτύου. Χρησιμοποιώντας το πρωτόκολλο RSVP κάθε διαδρομή LSP έχει ένα μέρος των πόρων του δικτύου που μπορεί να χρησιμοποιήσει.

Το εύρος ζώνης για κάθε LSP καθορίζεται κυρίως με δυο τρόπους:

- ▲ Εξωτερικός υπολογισμός (Offline Calculation)
- ▲ Αυτόματος υπολογισμός (Auto Bandwidth)

Το εύρος ζώνης στον εξωτερικό υπολογισμό υπολογίζεται από μοντέλα προσομοίωσης (χρησιμοποιώντας scripts ή άλλα εργαλεία) εκτός του δικτύου. Με αυτό τον τρόπο βασίστηκε η πρώτη υλοποίηση του μηχανισμού MPLS, η οποία χρησιμοποιείται μέχρι και σήμερα. Ενώ το εύρος ζώνης στον αυτόματο υπολογισμό υπολογίζεται από τον ίδιο το δρομολογητή μετρώντας περιοδικά τον όγκο δεδομένων που στέλλεται μέσω του LSP. Όταν υπάρξει αλλαγή της τιμής του όγκου των δεδομένων ανανεώνεται και ο πίνακας του RSVP πρωτοκόλλου που αποθηκεύει τους πόρους που χρειάζεται το κάθε LDP.

Όταν ένα πακέτο εισέλθει στο MPLS δίκτυο γίνεται η εύρεση του συντομότερου μονοπατιού (shortest path LSP) που μπορεί να ακολουθήσει για να

φτάσει στον προορισμό του αλλά παράλληλα να υπάρχουν και οι απαιτούμενοι πόροι που χρειάζονται για τη μεταφορά αυτού του πακέτου. Αν οι πόροι που απαιτούνται καλύπτονται από το συντομότερο μονοπάτι γίνεται η δρομολόγηση του πακέτου και όσο το πακέτο δρομολογείται από το συγκεκριμένο LSP αφαιρούνται οι πόροι που χρησιμοποιούνται από το συνολικό αριθμό των διαθέσιμων πόρων. Όταν ένα δεύτερο πακέτο κριθεί ότι έχει ως συντομότερο μονοπάτι αυτό που ήδη δρομολογεί ένα πακέτο θα γίνει έλεγχος ότι υπάρχουν ακόμα διαθέσιμοι πόροι για τη δρομολόγηση του. Σε αντίθετη περίπτωση (που δεν υπάρχουν αρκετοί πόροι) το δεύτερο πακέτο θα δρομολογηθεί από την επόμενη πιο σύντομη διαδρομή η οποία ταυτόχρονα θα έχει και τους απαιτούμενους πόρους. Η μηχανική κυκλοφορία υποστηρίζεται στα MPLS δίκτυα για τρεις κύριους λόγους:

- ▲ Βελτιστοποίηση της χρήσης του δικτύου
- ▲ Διαχείριση σφαλμάτων σε κόμβους και συνδέσεις
- ▲ Διαχείριση απροσδόκητης συμφόρησης

Η βελτιστοποίηση της χρήσης του δικτύου επιτυγχάνεται όταν οι δρομολογητές προωθούν τα πακέτα σε διαδρομές που πληρούν τις απαιτήσεις (όπως είναι το συντομότερο μονοπάτι, η ύπαρξη διαθέσιμων πόρων κλπ). Έτσι με αυτό τον τρόπο θα γίνεται η μέγιστη δυνατή αξιοποίηση όλων των πόρων του δικτύου, θα αποφεύγετε η υπερβολική συμφόρηση δεδομένων σε μερικούς κόμβους του δικτύου και οι δρομολογητές που βρίσκονται στα άκρα του δικτύου θα κατανέμουν καλύτερα την κυκλοφορία των δεδομένων προς όλες τις ροές δρομολόγησης.

Η διαχείριση των σφαλμάτων σε κόμβους και συνδέσεις αναφέρεται στη γρήγορη επαναδρομολόγηση (fast reroute) των δεδομένων σε περίπτωση εμφάνισης σφάλματος σε κάποιο κόμβο ή σύνδεση. Η αποκατάσταση του δικτύου ικανοποιεί τους περιορισμούς συντομότερου μονοπατιού και δρομολόγησης σε LSP με διαθέσιμους πόρους. Αυτό το επιτυχαίνει προϋπολογίζοντας εναλλακτικά μονοπάτια για πιθανά σφάλματα σε κόμβους ή συνδέσεις.

Σε περιπτώσεις μεγάλης συμφόρησης ενεργείται η διαχείριση απροσδόκητης συμφόρησης, όπου γίνεται δρομολόγηση δεδομένων και σε ροές που ήταν αδρανείς. Δεν υπάρχει όμως αποδοτική διαχείριση των πόρων του δικτύου αλλά με αυτό τον τρόπο επιτυγχάνεται η κανονική λειτουργία του δικτύου και η μη ύπαρξη απώλειας πακέτων.

Κεφάλαιο 4

Ανοχή Σφαλμάτων

4.1 Εισαγωγή στην Ανοχή Σφαλμάτων	15
4.2 Βασικές Έννοιες	16
4.3 Ανίχνευση Σφαλμάτων σε MPLS Δίκτυο	17
4.4 Μηχανισμοί Αποκατάστασης Δικτύου MPLS	18

4.1 Εισαγωγή στην Ανοχή Σφαλμάτων

Μια από τις μεγαλύτερες προκλήσεις του μηχανισμού MPLS είναι η εξασφάλιση ποιότητας υπηρεσίας στις ροές δεδομένων. Η παρουσία σφάλματος σε ένα σύνδεσμο ή κόμβο στο δίκτυο θα έχει τεράστια επίδραση στην ποιότητα των υπηρεσιών και εφαρμογών πραγματικού χρόνου. Η τεχνολογία MPLS όμως, βασίζεται σε αρχιτεκτονική που προσανατολισμένη στη σύνδεση. Αυτό έχει ως αποτέλεσμα ότι σε περίπτωση σφάλματος ο μηχανισμός MPLS πρέπει να εγκαταστήσει ένα νέο μονοπάτι δρομολόγησης δεδομένων και να χαρακτηρίζεται ως ένας μηχανισμός με αργή αντίδραση στις περιπτώσεις ύπαρξης σφαλμάτων στις ροές δρομολόγησης. Γι' αυτό το λόγο η ύπαρξη μηχανισμού γρήγορης επαναδρομολόγησης είναι ζωτικής σημασίας σε δίκτυα παροχής υπηρεσιών πραγματικού χρόνου έτσι ώστε να διασφαλίζεται η ποιότητα υπηρεσίας και να μην επηρεάζεται από την παρουσία σφαλμάτων. Στον μηχανισμό MPLS όταν ένα υφιστάμενο μονοπάτι δρομολόγησης δεδομένων καταστεί

ακατάλληλο λόγο σφάλματος, τα δεδομένα αυτά πρέπει να επαναδρομολογηθούν σε ένα εναλλακτικό μονοπάτι με σκοπό να ελαχιστοποιηθεί η διάρκεια της διακοπής αποστολής δεδομένων και την ομαλή συνέχιση της λειτουργίας των υπηρεσιών.

4.2 Βασικές Έννοιες

Για καλύτερη κατανόηση της λειτουργίας της ανίχνευσης σφαλμάτων σε ένα δίκτυο δίνονται σε συντομία οι σημαντικότεροι ορισμοί [12] [16].

Κύριο Μονοπάτι (Primary or Working Path): Το μονοπάτι το οποίο μεταφέρει δεδομένα πριν την εμφάνιση σφάλματος.

"Downstream" Μονοπάτι: Η κατεύθυνση που ακολουθούν τα δεδομένα από ένα κόμβο εισόδου προς ένα κόμβο εξόδου.

"Upstream" Μονοπάτι: Η κατεύθυνση που ακολουθούν τα δεδομένα από ένα κόμβο εξόδου προς ένα κόμβο εισόδου.

Εναλλακτικό Μονοπάτι (Alternative Path): Το μονοπάτι στο οποίο μεταφέρεται η επηρεαζόμενη ροή δεδομένων μετά το σφάλμα από τους μηχανισμούς αποκατάστασης του δικτύου MPLS. Η κατεύθυνση που ακολουθούν τα δεδομένα παραμένει η ίδια (από τον αρχικό κόμβο εισόδου προς τον αρχικό κόμβο εξόδου).

Alert LSR (Alert Node): Είναι ο κόμβος που ανιχνεύει το σφάλμα.

Ανίχνευση Σφάλματος (Fault Detection): Οι μηχανισμοί του δικτύου οι οποίοι αναγνωρίζουν τα σφάλματα στις συνδέσεις.

Ειδοποίηση Σφάλματος (Fault Notification ή Fault Indication Signal - FIS): Είναι το σήμα που δείχνει ότι έχει εμφανιστεί ένα σφάλμα κατά μήκος της πορείας δρομολόγησης των δεδομένων.

Χρόνος Αποκατάστασης (Recovery Period): Το χρονικό διάστημα που διανύεται από την ανίχνευση του σφάλματος στο δίκτυο μέχρι το τέλος της χρησιμοποίησης του επηρεαζόμενου μονοπατιού δρομολόγησης δεδομένων.

Υπολογισμός Εναλλακτικού Μονοπατιού (Alternative Path Computation): Ο upstream κόμβος μόλις παραλάβει το σήμα ανίχνευσης σφάλματος υπολογίζει εναλλακτικό μονοπάτι.

Επαναδρομολόγηση Δεδομένων στο Εναλλακτικό Μονοπάτι (Reroute Traffic to Alternative): Είναι η διαδικασία παράκαμψης των δεδομένων από το κυρίως μονοπάτι στο εναλλακτικό.

Επαναδρομολόγηση Δεδομένων στο Κυρίως Μονοπάτι (Traffic Reverting): Είναι η διαδικασία κατά την οποία τα δεδομένα δρομολογούνται από το εναλλακτικό μονοπάτι στο κυρίως μετά την αποκατάσταση του σφάλματος.

4.3 Ανίχνευση Σφαλμάτων σε MPLS Δίκτυο

Τα είδη των σφαλμάτων σε MPLS δίκτυο διακρίνονται σε δυο κύριες κατηγορίες:

▲ Σφάλμα στη σύνδεση μεταξύ δυο κόμβων δρομολόγησης

Η σύνδεση μεταξύ δυο κόμβων δρομολόγησης χάνεται και με την ανίχνευση του σφάλματος καλούνται οι μηχανισμοί επαναδρομολόγησης της επηρεαζόμενης ροής στην εναλλακτική.

▲ Σφάλμα στον κόμβο δρομολόγησης

Ο κόμβος δρομολόγησης σταματά την κανονική λειτουργία του και πλέον δεν προωθεί τα πακέτα που στέλλονται σε αυτόν. Με την ανίχνευση του σφάλματος γίνεται η επαναδρομολόγηση της επηρεαζόμενης ροής που περιλαμβάνει τον κόμβο με σφάλμα στην εναλλακτική πορεία.

Για την αποτελεσματική αποκατάσταση του δικτύου MPLS πρέπει να γίνεται σύντομη και αξιόπιστη ανακοίνωση των σφαλμάτων και της σοβαρότητας τους. Αν ο κόμβος ή η σύνδεση παρουσιάσει σφάλμα πρέπει να σταλεί άμεσα μηνύματα FIS έτσι

ώστε να ανακοινώσει προς τα πάνω το σφάλμα που εντόπισε. Το μήνυμα αυτό μεταβιβάζεται με την ψηλότερη προτεραιότητα και έτσι θα φτάσει έγκαιρα στον LSR, ο οποίος είναι υπεύθυνος για την αποκατάσταση της επηρεαζόμενης ροής.

4.4 Μηχανισμοί Αποκατάστασης Δικτύου MPLS

Τα μοντέλα αποκατάστασης δικτύων μπορούν να κατηγοριοποιηθούν σε δυο κατηγορίες:

- Κατανεμημένα
- Κεντριοποιημένα

Τα κεντριοποιημένα μοντέλα αποκατάστασης δικτύων κεντρικό σύστημα διαχείρισης για την υλοποίηση της αποκατάστασης σφαλμάτων (ανίχνευση σφαλμάτων, επιλογή εναλλακτικής δρομολόγησης κλπ). Το μοντέλο αυτό λαμβάνει όλες τις παραμέτρους του δικτύου έτσι ώστε να μπορεί να βελτιστοποιεί τα μονοπάτια δρομολόγησης. Με αυτό τον τρόπο γίνεται χρήση του ελάχιστου δυνατού αριθμού των πόρων αλλά έχει ως μεγάλο μειονέκτημα το μεγάλο χρόνο αποκατάστασης του δικτύου που απαιτείται. Αυτό οφείλεται στο ότι ο κεντρικός διαχειριστής αναλαμβάνει μεγάλο φόρτο και καθυστερεί να βρει εναλλακτικά μονοπάτια.

Από την άλλη τα κατανεμημένα μοντέλα προσφέρουν τη δυνατότητα ο κάθε κόμβος να χειρίζεται τα σφάλματα που παρουσιάζονται περιορίζοντας έτσι το χρόνο καθυστέρησης που παρουσιάζεται στα κεντριοποιημένα μοντέλα. Με την παρουσία σφάλματος στο δίκτυο τα επηρεαζόμενα μονοπάτια δρομολογούνται από τους κόμβους που εμπλέκονται χωρίς να εμπλέκεται ο κεντρικός διαχειριστής.

Οι μηχανισμοί αποκατάστασης που προσπαθούν να επαναφέρουν το δίκτυο χωρίζονται στους:

- Δυναμικούς
- Προϋπολογισμένους

Οι δυναμικοί μηχανισμοί αποκατάστασης ανταλλάσσουν αρκετά μηνύματα κατά τη διάρκεια της προσπάθειας για επαναφοράς του δικτύου, όπου γίνεται ο καθορισμός

και η εγκαθίδρυση του εναλλακτικού μονοπατιού δρομολόγησης δεδομένων. Η διαδικασία αυτή χρειάζεται αρκετό χρόνο μέχρι να ολοκληρωθεί σε αντίθεση με τους προϋπολογισμένους μηχανισμούς οι οποίοι προϋπολογίζουν από πριν τα εναλλακτικά μονοπάτια για τη δρομολόγηση των δεδομένων (με την αρχική δρομολόγηση των δεδομένων ο μηχανισμός αυτός έχει καθορίσει επίσης και εναλλακτικά μονοπάτια για τη ροή αυτή για τις περιπτώσεις που υπάρξουν σφάλματα). Με αυτό τον τρόπο η αποκατάσταση του σφάλματος γίνεται πιο γρήγορα και πιο αποτελεσματικά.

Σε ένα δίκτυο MPLS χρησιμοποιούνται οι μηχανισμοί που προαναφέρθηκαν. Οι σύνδεσμοι και οι κόμβοι είναι υπεύθυνοι για την επαναδρομολόγηση σε περίπτωση σφάλματος στις ροές τις οποίες ανήκουν. Η αποκατάσταση σφάλματος σε ένα MPLS δίκτυο μπορεί να επιτευχθεί με τους ακόλουθους δυο τρόπους:

- ▲ Επαναδρομολόγηση (Rerouting)
- ▲ Γρήγορη Επαναδρομολόγηση (Fast Rerouting)

Η μέθοδος της επαναδρομολόγησης χρησιμοποιείται για τον ορισμό και της εγκατάστασης ενός νέου μονοπατιού δρομολόγησης δεδομένων για την αποκατάσταση του δικτύου κατόπιν της παρουσίας σφάλματος, είτε σε κόμβο είτε σε σύνδεση, στο δίκτυο. Το κύριο και εναλλακτικό μονοπάτι δρομολόγησης είναι ξεχωριστά και υπολογίζονται σύμφωνα με την πολιτική δρομολόγησης, τοπολογία και το σφάλμα του δικτύου. Σε περίπτωση εμφάνισης σφάλματος στο δίκτυο ακολουθούνται τα εξής κύρια στάδια για την αποκατάσταση του:

- ▲ Αναγνώριση του σφάλματος από τον πιο κοντινό κόμβο
- ▲ Ενημέρωση γειτονικών κόμβων δρομολόγησης για το σφάλμα
- ▲ Υπολογισμός εναλλακτικού μονοπατιού
- ▲ Επαναδρομολόγηση των δεδομένων από το επηρεαζόμενο μονοπάτι στο εναλλακτικό

Χρησιμοποιώντας τη μέθοδο της επαναδρομολόγησης προσφέρεται η δυνατότητα χρησιμοποίησης των πληροφοριών του δικτύου για την αναγνώριση σφαλμάτων, εύρεση και συνέχιση της δρομολόγησης δεδομένων σε πραγματικό χρόνο. Για το λόγο αυτό η διαχείριση των πόρων του δικτύου μπορεί να προσφέρει καλύτερα

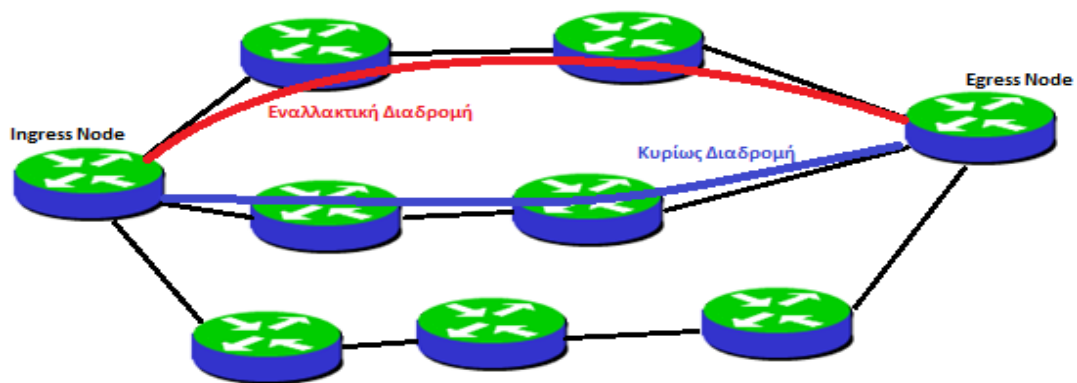
αποτελέσματα όσο αφορά το χρόνο που απαιτείται για την επαναδρομολόγηση των δεδομένων αφού αυτή θα γίνει με τις τρέχουσες πληροφορίες του δικτύου.

Όπως αναφέρθηκε και στο Κεφάλαιο 3, η γρήγορη επαναδρομολόγηση χρησιμοποιείται για την επαναδρομολόγηση των δεδομένων στο δίκτυο όταν εμφανιστεί κάποιο σφάλμα όπου οι εναλλακτικές διαδρομές εγκαθιδρύονται εκ των προτέρων (δηλαδή πριν την εμφάνιση του σφάλματος). Τα κριτήρια για τον καθορισμό των εναλλακτικών μονοπατιών που δημιουργούνται εκ των προτέρων βασίζεται στις παραμέτρους του δικτύου. Με την εμφάνιση κάποιου σφάλματος στο δίκτυο, ο δρομολογητής που είναι υπεύθυνος για την επαναδρομολόγηση εναλλάσσει το κύριο μονοπάτι στο εναλλακτικό μονοπάτι. Αξιοσημείωτο είναι το γεγονός ότι ο χρόνος που χρειάζεται για την αποκατάσταση του δικτύου είναι πολύ μικρότερος από την τεχνική της επαναδρομολόγησης λόγω του ότι τα εναλλακτικά μονοπάτια στην γρήγορη επαναδρομολόγηση υπολογίζονται πριν την εμφάνιση σφάλματος ενώ στην επαναδρομολόγηση υπολογίζονται δυναμικά σε πραγματικό χρόνο.

Για την ανάκτηση κάποιου σφάλματος στο δίκτυο υπάρχουν δυο κατηγορίες αποκατάστασης:

- ▲ Τοπική
- ▲ Καθολική

Η διόρθωση του σφάλματος στην περίπτωση της καθολικής αποκατάστασης καθορίζεται με το εναλλακτικό μονοπάτι να είναι ξεχωριστό από το κυρίως και να ξεκινά από τον αρχικό κόμβο (ingress node) και να τελειώνει στον τελικό κόμβο (egress node). Το εναλλακτικό μονοπάτι για τη δρομολόγηση των πακέτων μπορεί να υπολογιστεί δυναμικά (θα υπάρχει μεγαλύτερη καθυστέρηση μέχρι την επαναδρομολόγηση των πακέτων) ή να προϋπολογιστεί εκ των προτέρων. Στην πιο κάτω εικόνα (Εικόνα 4.4.1) είναι ένα παράδειγμα καθολικής αποκατάστασης όπου το εναλλακτικό μονοπάτι είναι προϋπολογισμένο ενώνοντας τον αρχικό κόμβο (ingress node) με τον τελικό (egress node) και οι κόμβοι που το αποτελούν είναι ανεξάρτητοι από το κυρίως μονοπάτι δρομολόγησης.



Εικόνα 4.4.1 - Καθολική Αποκατάσταση

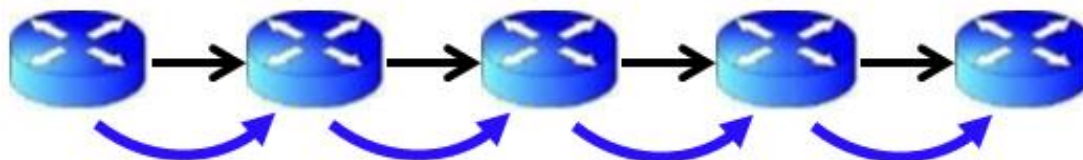
Η διόρθωση του σφάλματος στην περίπτωση της τοπικής αποκατάστασης στοχεύει στη διόρθωση του σφάλματος δημιουργώντας μια διαδρομή αποκατάστασης γύρω από το σφάλμα.

Το εναλλακτικό μονοπάτι μπορεί να είναι 1 προς 1 (One-to-One) ή πολλά προς 1 (Many-to-One). Στην περίπτωση όπου το εναλλακτικό μονοπάτι είναι 1 προς 1 για κάθε LSP δημιουργείται από το RSVP ένα ξεχωριστό εναλλακτικό μονοπάτι. Ενώ στην περίπτωση που το εναλλακτικό μονοπάτι είναι πολλά προς ένα σημαίνει ότι για πολλά LSP θα υπάρχουν κάποια εναλλακτικά μονοπάτια που θα είναι κοινά. Ως αποτέλεσμα στις περιπτώσεις που υπάρχουν σφάλματα στο δίκτυο και επηρεάζουν πολλαπλά μονοπάτια δρομολόγησης που έχουν το ίδιο εναλλακτικό μονοπάτι δεν προσφέρετε οποιαδήποτε εγγύηση ότι θα μπορούν να εξυπηρετηθούν όλα τα αιτήματα των ροών που έχουν επηρεαστεί.

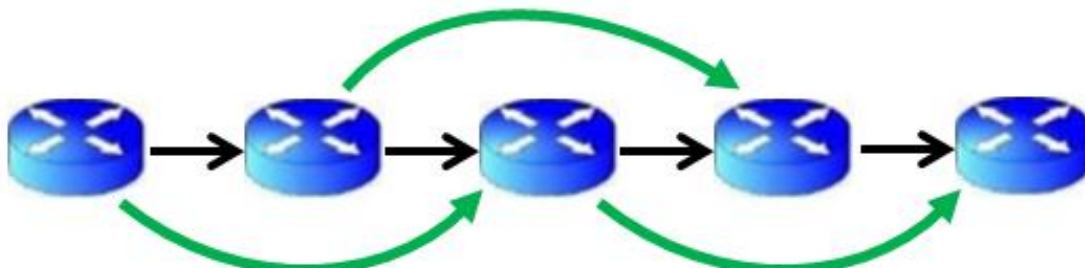
Πιο κάτω παρουσιάζονται τα είδη προστασίας σε ένα MPLS δίκτυο.



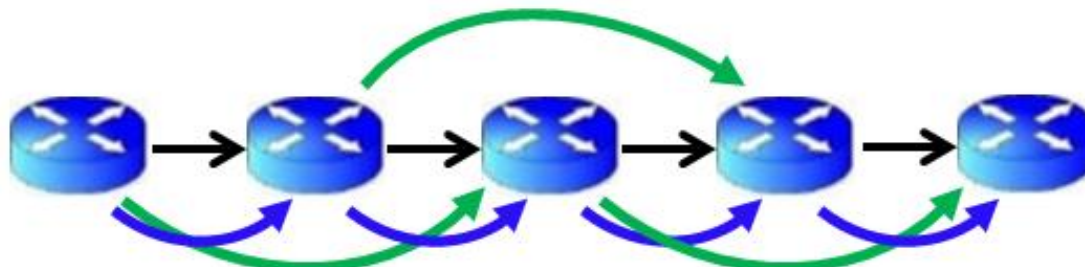
Εικόνα 4.4.2 - MPLS Δίκτυο Χωρίς Προστασία



Εικόνα 4.4.3 - MPLS Δίκτυο Με Προστασία στις Συνδέσεις



Εικόνα 4.4.4 - MPLS Δίκτυο Με Προστασία στους Κόμβους



Εικόνα 4.4.5 - MPLS Δίκτυο Με Προστασία στους Κόμβους και στις Συνδέσεις

Κεφάλαιο 5

Ανασκόπηση Υβριδικού Αλγορίθμου

5.1 Εισαγωγή	23
5.2 Αλγόριθμοι Ανοχής Σφαλμάτων Υβριδικού	24
5.3 Αλγόριθμος Μηχανικής Κυκλοφορίας Υβριδικού	31
5.4 Περιγραφή Υβριδικού Αλγορίθμου	32

5.1 Εισαγωγή

Ο Υβριδικός αλγόριθμος προτάθηκε σε προηγούμενη ερευνητική εργασία [4]. Ο αλγόριθμος αυτός αποτελεί συνδυασμό δυο ήδη υλοποιημένων αλγορίθμων, που κρίθηκαν ως οι καλύτεροι βασισμένοι στη ανοχή σφαλμάτων και μηχανικής κυκλοφορίας σε δίκτυα MPLS. Για την ανοχή σφαλμάτων ο υβριδικός αλγόριθμος συνδυάζει δυο αλγόριθμους ανοχής σφαλμάτων έτσι ώστε να μπορεί να προσφέρει γρήγορη επαναδρομολόγηση των πακέτων αλλά παράλληλα να αντέχει πολλαπλά σφάλματα. Ακολουθώντας με τις απαραίτητες αλλαγές στον υβριδικό αλγόριθμο ενσωματώθηκε με τον ένα από τους καλύτερους αλγόριθμους μηχανικής κυκλοφορίας. Στα επόμενα υποκεφάλαια θα παρουσιαστούν ξεχωριστά οι τρεις αλγόριθμοι που αποτελούν τον υβριδικό και στο τέλος ο υβριδικός αλγόριθμος.

5.2 Αλγόριθμοι Ανοχής Σφαλμάτων Υβριδικού

5.2.1 Αλγόριθμος Otel

Ο αλγόριθμος ανοχής σφαλμάτων Otel [4] [17] χρησιμοποιεί τη μέθοδο της τοπικής αποκατάστασης χρησιμοποιώντας πληροφορίες σχετικά με τη σύνδεση που παρουσιάστηκε το σφάλμα, τον πιο πάνω κόμβο της σύνδεσης καθώς και το δέντρο με τα μικρά μονοπάτια (Short Path Tree - SPT). Με τον εντοπισμό του σφάλματος από τον προς τα πάνω κόμβο εγκαθίσταται ένα τοπικό μονοπάτι, το οποίο λειτουργεί ως εναλλακτική ροή δρομολόγησης αποφεύγοντας έτσι το σημείο που εμφανίστηκε το σφάλμα. Όταν παρουσιαστεί κάποιο σφάλμα ενημερώνονται μόνο τα τμήμα του STP που επηρεάζονται άμεσα από το σφάλμα αυτό.

Ο αλγόριθμος διατηρεί τις εξής βασικές δομές δεδομένων:

- Το STP στο οποίο η ρίζα του είναι ο κόμβος που θα εκτελέσει του υπολογισμούς για την δρομολόγηση στο νέο εναλλακτικό μονοπάτι
- Ένα πίνακα που περιέχει τις αποστάσεις των πιο κοντινών μονοπατιών από τη ρίζα του STP προς όλους
- Ουρά προτεραιότητας για τους κόμβους, όπου κάθε κόμβος παρουσιάζεται ως μια τετράδα (n, Δ, p, l). Η μεταβλητή n είναι ο κόμβος, το Δ ορίζει μοναδικά την τετράδα, το p είναι ο πατέρας του κόμβου στο SPT και το l είναι η σύνδεση που ενώνει το p και το l .

Όταν μια τετράδα για ένα κόμβο εισαχθεί στην ουρά γίνεται έλεγχος αν υπάρχει ήδη μια τετράδα καταχωρημένη για το συγκεκριμένο κόμβο. Αν υπάρχει η υφιστάμενη τετράδα αντικαταστίτε από τη νέα τετράδα αν η μεταβλητή Δ της νέας τετράδας είναι μικρότερη από αυτή της υφιστάμενης.

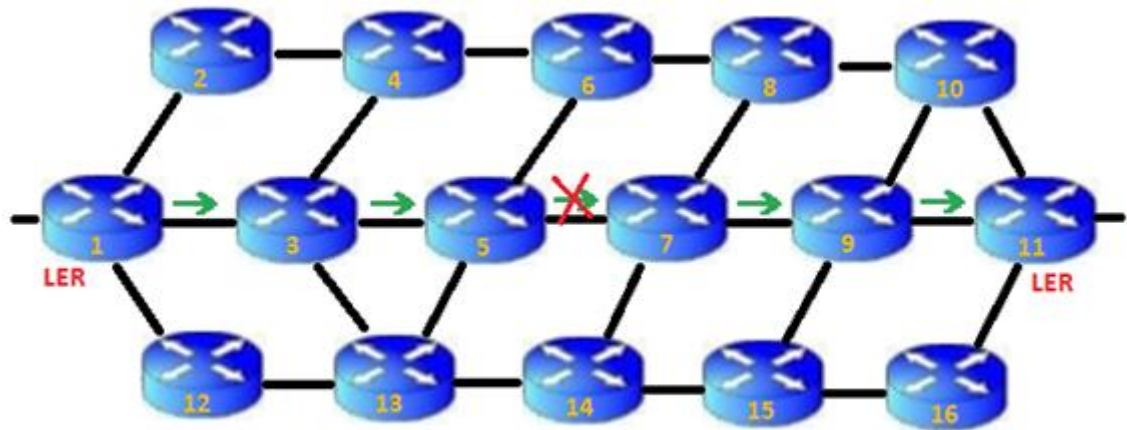
Τα βήματα του αλγορίθμου έχουν ως εξής:

- Σημαδεύουμε όλους τους κόμβους του υφιστάμενου STP ως προσεγγίσιμοι.
- Ορίζουμε σαν SPT υπό-δέντρο που έχει σαν ρίζα τον αποσυνδεδεμένο προς τα κάτω κόμβο. Αρχίζοντας από τη ρίζα του STP υπό-δέντρου σημαδεύουμε όλους τους κόμβους του STP υπό-δέντρου σαν μη προσεγγίσιμοι.

- ▲ Κατασκευάζουμε το σύνολο L , το οποίο περιέχει τις συνδέσεις του δικτύου που έχουν σαν κόμβο εισόδου ένα κόμβο προσεγγίσιμο και έχουν σαν κόμβο εξόδου ένα κόμβο που είναι μη προσεγγίσιμος.
- ▲ Για κάθε σύνδεση l του συνόλου L ,

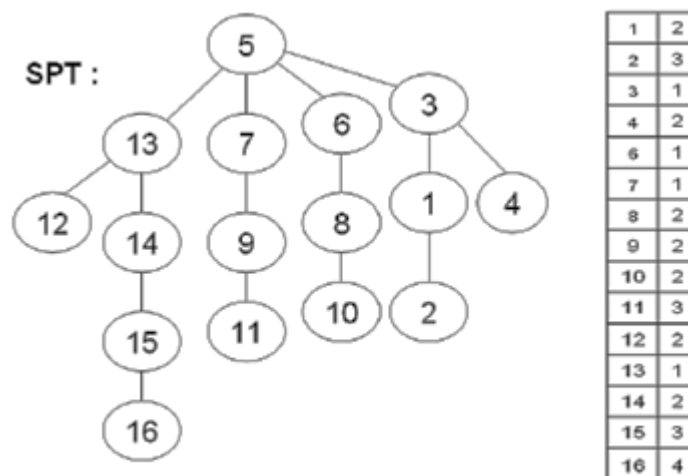
 - Ορίζουμε ως p και n τους κόμβους εισόδου και εξόδου της σύνδεσης l αντίστοιχα. Υπολογίζουμε την απόσταση της δοκιμαστικής πορείας από τη ρίζα του SPT μέχρι και τον κόμβο n που είναι ίσο με το άθροισμα του πιο μικρού μονοπατιού από τη ρίζα του SPT στον κόμβο n και το βάρος της σύνδεσης.
 - Υπολογίζουμε το Δ ως τη διαφορά μεταξύ της δοκιμαστικής πορείας και της απόστασης του μικρότερου μονοπατιού από τη ρίζα του SPT και του κόμβου n στην προηγούμενη μη ενημερωμένη τοπολογία.
 - Για τον κόμβο n εισάγουμε στην ουρά προτεραιότητας την τετράδα που έχει διαμορφωθεί από τις τιμές των n , p , Δ και l .
- ▲ Από την ουρά προτεραιότητας εξάγουμε μια τετράδα (n', p', Δ', l') . Ενημερώνουμε το SPT διαγράφοντας το κλαδί που ενώνει τον κόμβο n' με τον πατέρα του στο παλιό SPT και προσθέτοντας κλαδί με τη βοήθεια της σύνδεσης l' , ενώνοντας έτσι τον απροσέγγιστο n' και με τον προσεγγίσιμο κόμβο p' . Επίσης ενημερώνεται ο πίνακας που περιέχει τις αποστάσεις των πιο κοντινών μονοπατιών, υπολογίζοντας τη μικρότερη απόσταση από τη ρίζα του SPT στον κόμβο n' (δηλαδή το άθροισμα του πιο μικρού μονοπατιού από την ρίζα του SPT στο κόμβο p') και το βάρος της σύνδεσης l' . Αν ο κόμβος n' είναι ο κόμβος προορισμού ο αλγόριθμος τερματίζει.
- ▲ Σημάδεψε τον κόμβο n' και όλα τα παιδιά του στο SPT σαν προσεγγίσιμοι. Διαγραφούμε από την ουρά προτεραιότητας όλες τις τετράδες που αντιπροσωπεύουν τα παιδιά του. Ενημερώνουμε τις τιμές του συνόλου L που περιέχει τις συνδέσεις που έχουν σαν κόμβο εισόδου LSR τον κόμβο n' ή κάποιο από τα παιδιά του και σαν κόμβο εξόδου ένα κόμβο απροσέγγιστο. Πήγαινε στο βήμα 4.

Στην τοπολογία της Εικόνας 5.2.1.1 θα εξηγήσουμε αναλυτικά πως ο αλγόριθμος Otel λειτουργεί σε περίπτωση σφάλματος. Το σφάλμα που παρουσιάζεται στην τοπολογία είναι στο κυρίως μονοπάτι μεταξύ του LSR5 και LSR7.



Εικόνα 5.2.1.1 - Τοπολογία MPLS δικτύου με σφάλμα στη σύνδεση του κόμβου LSR5 & LSR7 [4]

Με την εφαρμογή του αλγόριθμου Otel δημιουργείται το SPT, όπου θέτει ως ρίζα τον κόμβο LSR5 αφού αυτός είναι ο πρώτος κόμβος που εμφανίζεται προς τα πάνω εκεί που εντοπίζεται άμεσα το σφάλμα. Η δομή που δημιουργείται παρουσιάζεται από το δέντρο της Εικόνας 5.2.1.2. Ο πίνακας που δημιουργείται περιέχει όλες τις αποστάσεις των πιο κοντινών μονοπατιών από τη ρίζα του SPT προς όλους τους υπόλοιπους κόμβους. Στο βήμα 2 σημαδεύουμε τους απροσέγγιστους κόμβους (στην περίπτωση της πιο πάνω τοπολογίας οι κόμβοι που είναι μη προσεγγίσιμοι είναι οι κόμβοι LSR7, LSR9 και LSR11). Στο βήμα 3 δημιουργούμε το σύνολο των συνδέσμων που αποτελείται από τους συνδέσμους 7-8, 7-14, 9-10, 9-15, 11-10 και 11-16.



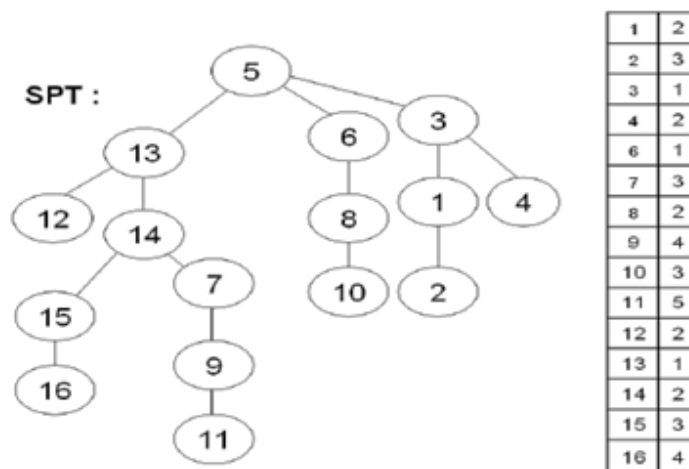
Εικόνα 5.2.1.2 - Δομή αλγόριθμου Otel [4]

Στο βήμα 4 για κάθε σύνδεση δημιουργείται οι ανάλογες τετράδες (n, p, Δ, l). Ακολούθως οι τετράδες με το μικρότερο Δ για κάθε κόμβο που δεν είναι προσεγγίσιμος εισάγονται στην ουρά προτεραιότητας. Με βάση το κριτήριο αυτό θα εισέρθουν οι τετράδες με σύνδεση 7-14, 9-15 και 11-10. Ο πίνακας των τετράδων (n, p, Δ, l) παρουσιάζεται στην Εικόνα 5.2.1.3.

n	p	Δ	l
7	8	2	7-8
7	14	0	7-14
9	10	2	9-10
9	15	1	9-15
11	10	1	11-10
11	16	1	11-16

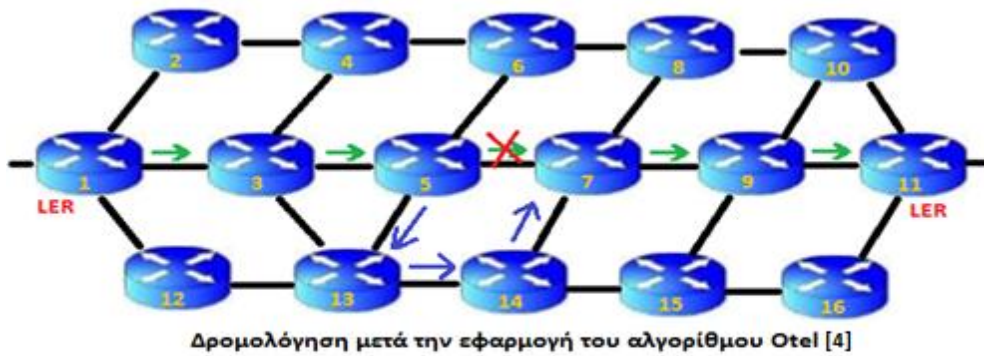
Εικόνα 5.2.1.3 - Πίνακας τετράδων που δημιουργείται στο Βήμα 4 [4]

Στο βήμα 5 γίνεται η επιλογή της τετράδας με το μικρότερο Δ και εξάγεται από την ουρά προτεραιότητας (η τετράδα αυτή αναφέρεται στον κόμβο 7, ο οποίος είναι μη προσεγγίσιμος). Το σύνολο L ενημερώνεται λόγω του ότι δεν υπάρχουν άλλοι μη προσεγγίσιμοι κόμβοι και ο αλγόριθμος σταματά. Στην Εικόνα 5.2.1.4 και 5.2.1.5 παρουσιάζεται το SPT μετά την υλοποίηση του αλγορίθμου και την επίλυση του σφάλματος ενώ στην τοπολογία που ακολουθεί παρουσιάζεται η δρομολόγηση των πακέτων μετά την εφαρμογή του αλγορίθμου Otel.



Δομή αλγορίθμου Otel [4]

Εικόνα 5.2.1.4



Εικόνα 5.2.1.4

5.2.2 Αλγόριθμος GONFA

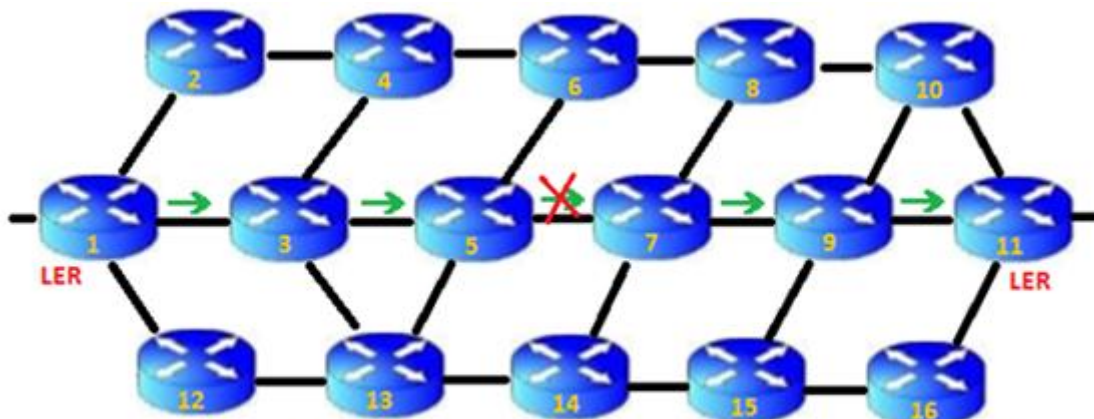
Ο αλγόριθμος ανοχής σφαλμάτων GONFA [4] [7] χρησιμοποιεί τοπική και καθολική αποκατάσταση συ συνδυασμό με την εναλλαγή σε εναλλακτικό μονοπάτι. Η καθολική αποκατάστασης χαρακτηρίζεται από ένα σύνολο LSRs, το οποίο προστατεύει ολόκληρο το κυρίως μονοπάτι. Η καθολική προστασία χωρίζεται σε πολλαπλές περιοχές προστασίας και η αποκατάσταση του σφάλματος γίνεται μέσα στην περιοχή του τμήματος παρέχοντας έτσι τμηματική προστασία. Με την τεχνική αυτή επιτυγχάνεται όλη η περιοχή του MPLS δικτύου να είναι ένα σύνολο από περιοχές προστασίας τμήματος και η εναλλακτική πορεία για ολόκληρη την περιοχή του MPLS δικτύου να δημιουργείται από τη συνένωση μερικών σημείων των τμηματικών αυτών περιοχών. Κατά τη διαδικασία της αποκατάστασης του σφάλματος από τον αλγόριθμο GONFA η εναλλακτική πορεία δρομολόγησης προκύπτει από την προσωρινή συνένωση του προς τα πίσω μονοπατιού αρχίζοντας από τον LSR που εντόπισε το σφάλμα και το προκαθορισμένο εναλλακτικό μονοπάτι. Η προσωρινή συνένωση του προς τα πίσω μονοπατιού γίνεται για να μεταφέρει τα πακέτα που δρομολογούνται από το κυρίως μονοπάτι δρομολόγησης στο εναλλακτικό.

Τα βήματα του αλγορίθμου έχουν ως εξής:

- ▲ Εγκατάσταση του κυρίου LSP
- ▲ Ρητή δρομολόγηση του εναλλακτικού μονοπατιού για ολόκληρο το κυρίως μονοπάτι δρομολόγησης

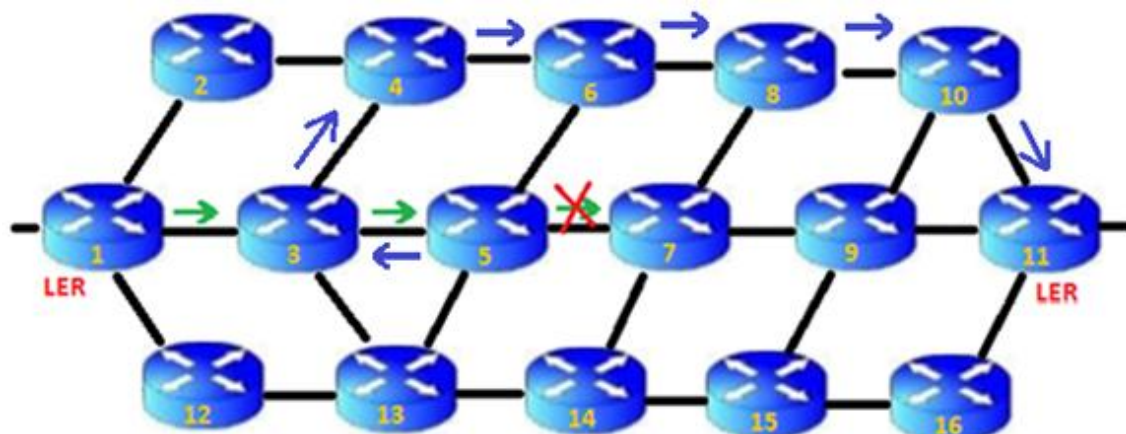
- ▲ Για την κάθε τμηματική περιοχή εγκαθίστανται ξεχωριστά προκαθορισμένα προς τα πίσω μονοπάτια

Στην τοπολογία της Εικόνας 5.2.2.1 θα εξηγήσουμε αναλυτικά πως ο αλγόριθμος GONFA λειτουργεί σε περίπτωση σφάλματος. Το σφάλμα που παρουσιάζεται στην τοπολογία είναι στο κυρίως μονοπάτι μεταξύ του LSR5 και LSR7.

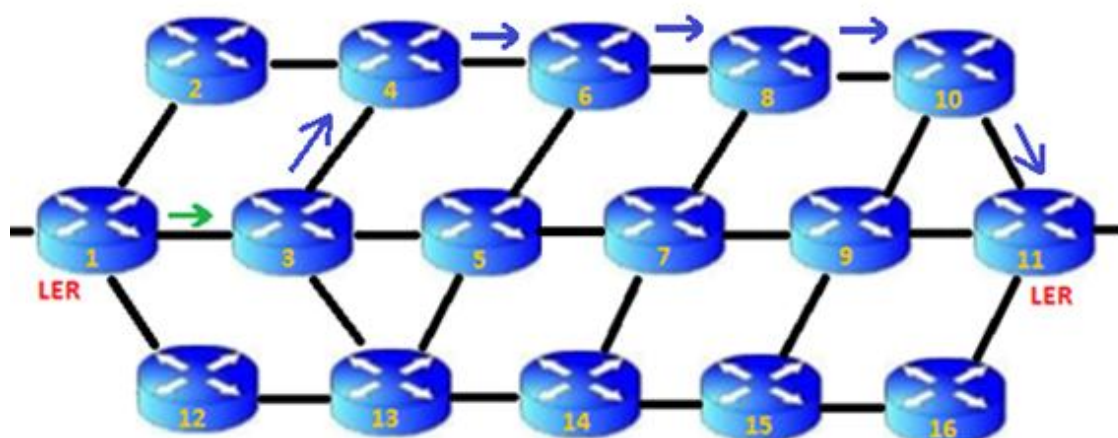


Εικόνα 5.2.2.1 - Τοπολογία MPLS δικτύου με σφάλμα στη σύνδεση του κόμβου LSR5 & LSR7 [4]

Όταν παρουσιαστεί το σφάλμα γίνεται εγκατάσταση του προς τα πίσω μονοπάτι LSR3 και LSR1 για την τμηματική προστασία 1-3 και 1-2-4-3. Το δεύτερο προς τα πίσω μονοπάτι είναι το 7-5-3 για τη τμηματική προστασία 3-5-7 και 3-4-6-8-7, το τρίτο προς τα πίσω μονοπάτι είναι το 11-9-7 για τη τμηματική προστασία 7-9-11 και 7-8-10-11. Σε περίπτωση σφάλματος μεταξύ του LSR5 - LSR7, η ροή θα δρομολογηθεί από τον LSR5 στο προς τα πίσω μονοπάτι LSR5, LSR3, LSR4 και στην συνέχεια θα μεταφερθεί στο εναλλακτικό μονοπάτι LSR4, LSR6, LSR8, LSR10 και LSR11, όπως φαίνεται στην πρώτη τοπολογία της Εικόνας 5.2.2.2 όταν ο LSR3 παραλαμβάνει την ροή μέσω του προς τα πίσω μονοπατιού, και καταλαβαίνει ότι υπάρχει σφάλμα στο υπόλοιπο μονοπάτι τότε το δρομολογεί απευθείας στο εναλλακτικό μονοπάτι. Αυτό παρουσιάζεται στην δεύτερη τοπολογία της Εικόνας 5.2.2.2 όπου η ροή δρομολογείται μέσω των LSR1, LSR3, LSR4, LSR6, LSR8, LSR10 και LSR11.



Δρομολόγηση προς τα πίσω σύμφωνα με τον αλγόριθμο GONFA [4]



Δρομολόγηση στο εναλλακτικό μονοπάτι [4]

Εικόνα 5.2.2.2

Στον αλγόριθμο GONFA χρησιμοποιείται επίσης η τεχνική της γρήγορης και αξιόπιστης δρομολόγησης, η οποία παρέχει προληπτικές ενέργειες για το προστατευμένο LSP ενάντια στην απώλεια πακέτων κατά τη διάρκεια σφάλματος (Reliable and Fast Rerouting - RFR). Η τεχνική επιτυγχάνει την αποφυγή της απώλειας πακέτων είτε αυτό οφείλεται σε σφάλμα σε σύνδεση είτε σε σφάλμα σε κόμβο. Το κάθε πακέτο σε κάθε LSR αντιγράφεται και αποθηκεύεται στη τοπική του μνήμη πριν αυτό να σταλεί στην κατάλληλη σύνδεση. Όταν κάποιο σφάλμα εμφανιστεί από κάποιο LSR η διαδικασία της αποστολής των πακέτων στο προς τα πίσω μονοπάτι ξεκινά αμέσως με τα πακέτα που είναι αποθηκευμένα στην τοπική μνήμη του LSR να στέλνονται πρώτα στο προς τα πίσω μονοπάτι. Οποιοδήποτε επόμενο πακέτο που παραλαμβάνεται από το κύριο μονοπάτι στέλνεται επίσης στο προς τα πίσω μονοπάτι. Ο κόμβος αυτός έχει

αντίγραφα των πακέτων που χάθηκαν από τη στιγμή που το σφάλμα προέκυψε στη σύνδεση ή στον κόμβο, επομένως δεν υπάρχει οποιαδήποτε απώλεια πακέτων. Μόλις κάθε προς τα πάνω κόμβος ανιχνεύσει το πρώτο πακέτο που δρομολογείται μέσω του προς τα πίσω μονοπατιού, αδειάζει όλα τα πακέτα τα οποία αποθηκεύτηκαν προηγουμένως για τυχόν σφάλμα στην σύνδεση εξόδου του κόμβου ή στον συνεργάτη κόμβο. Το επόμενο πακέτο που έρχεται κανονικά από το κύριο μονοπάτι, προστίθεται σε αυτό ένα στίγμα και δρομολογείται κανονικά στο κύριο μονοπάτι με αποτέλεσμα να μειώνεται η μέση καθυστέρηση πακέτων (average packet delay). Όταν το πακέτο με το στίγμα ανιχνευτεί κατά την πορεία του στο προς τα πίσω μονοπάτι δίνει την ένδειξη στον κόμβο που το ανιχνεύει ότι όλα τα επόμενα πακέτα μετά από αυτό θα είναι αυτά που έχουν αποθηκευτεί προηγουμένως και θα αποσταλούν μετά από αυτό, διατηρώντας έτσι τα πακέτα σε σειρά.

5.3 Αλγόριθμος Μηχανικής Κυκλοφορίας Υβριδικού

5.3.1 Αλγόριθμος LSFSB

Ο αλγόριθμος μηχανικής κυκλοφορίας Local State Fair Share Bandwidth (LSFSB) [4] [18] επιλέχτηκε ως ο πιο αποτελεσματικός αλγόριθμος ο οποίος προσπαθεί να δρομολογήσει όσο το δυνατό πιο κατανομημένα τις ροές λαμβάνοντας αποφάσεις αποκλειστικά βασιζόμενος στους ακρινούς δρομολογητές. Αντιθέτως με άλλους αλγορίθμους, ο αλγόριθμος LSFSB δε χαρακτηρίζεται με μεγάλη πολυπλοκότητα αλλά ούτε και ανταλλαγές μηνυμάτων μεταξύ των δρομολογητών έτσι ώστε να μπορούν να διαχειριστούν το διαθέσιμο εύρος ζώνης του δικτύου. Ο αλγόριθμος LSFSB χρησιμοποιεί τοπικές πληροφορίες και είναι based-router αλγόριθμος όπου οι δρομολογητές στις άκρες του δικτύου παίρνουν τις αποφάσεις όσο αφορά τις διαδρομές στις οποίες θα τοποθετηθεί η κυκλοφορία που εισάγεται. Με αυτό τον τρόπο εξασφαλίζεται μια γρήγορη και αποδοτική μέθοδος για τη μείωση της συμφόρησης του δικτύου καθώς αυξάνεται το επίπεδο της ποιότητας των υπηρεσιών του. Ως αλγόριθμος μηχανικής κυκλοφορίας, ο LSFSB είναι ένας απλός αλγόριθμος και χρειάζεται ελάχιστη γνώση της κατάστασης του δικτύου και αποτελείται από δύο στάδια.

▲ Στάδιο 1

Το πρώτο στάδιο του αλγορίθμου εκπληρώνει τον κατακερματισμό της χωρητικότητας του πυρήνα και για κάθε ροή κυκλοφορίας χρησιμοποιεί τον wildest shortest path αλγόριθμο για να επιλέξει τη διαδρομή. Όταν η χωρητικότητα είναι πλήρης ο αλγόριθμος προχωρά στο δεύτερο στάδιο χρησιμοποιώντας τα αποτελέσματα του πρώτου σταδίου.

▲ Στάδιο 2

Στο δεύτερο στάδιο ο LSFSB αλγόριθμος υπερφορτώνει το δίκτυο, χρησιμοποιώντας σαν πλεονέκτημα τις ικανότητες διάσπασης του IP. Στο δεύτερο στάδιο δεν μπορεί να χρησιμοποιηθεί ο αλγόριθμος wildest shortest path λόγω του ότι ο αλγόριθμος αυτός προσδιορίζεται μόνο για διαδρομές με την αναμενόμενη χωρητικότητα.

Σε γενικές γραμμές, ο LSFSB είναι ο μόνος αλγόριθμος μηχανικής κυκλοφορίας που προσπαθεί να βελτιώσει την χρήση των πόρων του δικτύου με εξισορρόπηση του φορτίου, χρησιμοποιώντας πολλαπλά μονοπάτια όταν παρουσιαστεί υπερφόρτωση στο δίκτυο. Αποθηκεύοντας τοπικές πληροφορίες κάνει πιο εύκολους του υπολογισμούς μονοπατιών και οι αποφάσεις για την εισαγωγή μιας ροής στο δίκτυο λαμβάνονται από τους ακρινούς κόμβους. Τέλος προσπαθεί για γρήγορη επαναφορά και μέγιστο ρυθμό μεταφοράς δεδομένων όπου παράλληλα να διατηρείται η ποιότητα υπηρεσιών και η αξιοπιστία του δικτύου.

5.4 Περιγραφή Υβριδικού Αλγορίθμου

Ο υβριδικός αλγόριθμος συνενώνει τη μηχανική κυκλοφορία με την ανοχή σφαλμάτων ξεκινώντας τον αλγόριθμο εφαρμόζοντας μηχανική κυκλοφορία για την αποδοτική δρομολόγηση των δεδομένων στις ροές και με την εμφάνιση σφάλματος στο δίκτυο ενεργοποιείται ο μηχανισμός ανοχής σφαλμάτων.

Έτσι εφαρμόζεται αρχικά η μηχανική του LSFSB αλγορίθμου, όπου υπολογίζονται τα κοντινά μονοπάτια. Στο πρώτο στάδιο του αλγορίθμου γίνεται ο

κατακερματισμός της χωρητικότητας του πυρήνα και χρησιμοποιείται ο wildest shortest path αλγόριθμος για την επιλογή της διαδρομής για κάθε ροή δρομολόγησης από ένα σύνολο διαδρομών καθώς και το εναλλακτικό μονοπάτι, το οποίο θα χρησιμοποιηθεί στην περίπτωση εμφάνισης σφάλματος και τα πακέτα θα δρομολογούνται από αυτό αντί από το αρχικό κύριο μονοπάτι. Όταν η χωρητικότητα είναι πλήρης ξεκινά το δεύτερο στάδιο του αλγόριθμου. Χρησιμοποιώντας τα αποτελέσματα από το πρώτο στάδιο υπολογίζει το εύρος ζώνης του πυρήνα. Οι κόμβοι εισόδου διανέμουν τη ροή που τους αναλογεί στις συνδέσεις τους που τους ενώνουν με τον πυρήνα του MPLS δικτύου ανάλογα με το δικό τους εύρος ζώνης. Επίσης στο δεύτερο στάδιο ορίζεται ότι οι κόμβοι εισόδου θα αποδεχτούν την QoS κυκλοφορία στα όρια των συνδέσεων ενώσω το εύρος ζώνης του πυρήνα είναι μεγαλύτερο από αυτό που μπορεί να μεταφέρει η σύνδεση μεταξύ τους. Όταν ο πυρήνας αρχίζει να γεμίζει υπάρχουν δύο περιπτώσεις που μπορούν να συμβούν:

- ▲ Να τεθεί σε λειτουργία το πρώτο στάδιο του αλγορίθμου και δεν θα επιτραπεί η είσοδος επιπρόσθετης κυκλοφορίας
- ▲ Να δεχτεί τη ροή και είτε να διανέμει την εισερχόμενη κυκλοφορία σύμφωνα με τη ροή δια μέσου των κατάλληλων διαδρομών σε αναλογία με τη συνολική διανομή των διαδρομών που έγινε στο πρώτο στάδιο του αλγορίθμου, είτε να παρακολουθήσει την αποστολή των ροών κατά το πρώτο στάδιο και να αντικαταστήσει τις ροές αυτές έτσι ώστε να ανανεωθεί η διανομή των διαδρομών

Επομένως χρειάζεται ένα επιπλέον στάδιο της υπερφόρτωσης όπου γίνεται η κατανομή της εισερχόμενης κυκλοφορίας. Με αυτό τον τρόπο κάθε ροή που θα εισέρχεται στο δίκτυο από τον κόμβο εισόδου θα εισάγεται σε ένα LSP σύμφωνα με τον LSFSB αλγόριθμο.

Για να επηρεαστεί η ροή δεδομένων από σφάλμα άμεσα ή έμμεσα πρέπει το σφάλμα να προκύψει σε μονοπάτι που δρομολογούνται τα πακέτα ή στο εναλλακτικό. Η αποκατάσταση του δικτύου θα γίνει με διαφορετικό τρόπο ανάλογα με το που και με ποια σειρά θα εντοπιστεί το σφάλμα. Οι δυο αυτές περιπτώσεις αναλύονται πιο κάτω.

▲ Πρώτη περίπτωση

Όταν εντοπιστεί σφάλμα από τον προς τα πάνω κόμβο, ο οποίος βρίσκεται στο αρχικό μονοπάτι και δεν έχει προηγηθεί σφάλμα στο εναλλακτικό μονοπάτι τότε ο ακρινός κόμβος, αφού ενημερωθεί για το σφάλμα, θα ελέγξει αν το εναλλακτικό μονοπάτι που έχει υπολογιστεί στην αρχή αν διαθέτει τους απαραίτητους πόρους για να φιλοξενήσει την επηρεαζόμενη ροή. Αν οι απαραίτητοι πόροι είναι διαθέσιμοι γίνεται η εναλλαγή. Σε αντίθετη περίπτωση επαναλαμβάνεται ο διαμοιρασμός του εύρου ζώνης για να υπολογιστεί ξανά το εναλλακτικό μονοπάτι. Ταυτόχρονα ενημερώνονται όλοι οι κόμβοι μέσω FIS μηνυμάτων που στέλνονται από τον προς τα πάνω κόμβο όταν εμφανιστεί το σφάλμα. Ο κόμβος εισόδου θα ενημερώσει τις δομές που διατηρεί ο αλγόριθμος Otel, έτσι ώστε όταν χρειαστεί να εφαρμοστεί οι ροές να είναι πλήρως ενημερωμένες σύμφωνα με τη νέα κατάσταση του δικτύου.

Σε περίπτωση που εντοπιστεί σφάλμα στο μονοπάτι που ορίστηκε ως το νέο μονοπάτι δρομολόγησης θα εφαρμοστεί ο Otel αλγόριθμος, ο οποίος θα μας δώσει το τοπικό εναλλακτικό μονοπάτι έτσι ώστε να αποκατασταθεί το σφάλμα. Ο Otel θα επιλέξει το σφάλμα ανάλογα με τα βάρη των συνδέσεων και το συνολικό διαθέσιμο εύρο ζώνης. Το μονοπάτι που επιλέγεται είναι αυτό που έχει το μεγαλύτερο διαθέσιμο εύρος ζώνης. Όσο τα σφάλματα εντοπίζονται στα εναλλακτικά μονοπάτια που χρησιμοποιούνται για την δρομολόγηση των πακέτων, η αποκατάσταση του δικτύου θα γίνεται σύμφωνα με τον τροποποιημένο αλγόριθμο Otel όπως ορίζει ο υβριδικός αλγόριθμος. Αυτό οφείλεται στο ότι ακολουθεί την τεχνική επαναδρομολόγησης και όλοι οι αλγόριθμοι που ακολουθούν την τεχνική αυτή είναι ικανοί να ανεχτούν πολλαπλά σφάλματα στο υφιστάμενο και στο εφεδρικό μονοπάτι δρομολόγησης. Όλοι οι κόμβοι θα ενημερωθούν μέσω των μηνυμάτων FIS που στέλνει ο προς τα πάνω κόμβος όταν εντοπιστεί το σφάλμα και ο κόμβος εισόδου θα ενημερώσει τις δομές που διατηρεί ο Otel. Τέλος λόγω της τροποποίησης του μπορεί να επιλέγει μη φορτωμένες συνδέσεις που θα μπορούν να μεταφέρουν την ροή και η διαδικασία αυτή εφαρμόζεται για κάθε ροή που θα εισέρχεται στο δίκτυο.

▲ Δεύτερη περίπτωση

Όταν το σφάλμα εντοπιστεί σε μονοπάτια που δε μεταφέρουν ροές κυκλοφορίας και δεν υπάρχει σφάλμα στα μονοπάτια που δρομολογούνται οι κυρίως ροές, η ροή δεν επηρεάζεται και συνεχίζει να δρομολογείται από το αρχικό κυρίως μονοπάτι. Όλοι οι

κόμβοι όμως θα ενημερωθούν μέσω FIS μηνυμάτων, τα οποία στέλλονται από τον προς τα πάνω κόμβο όταν εμφανιστεί το σφάλμα. Ο κόμβος εισόδου θα ενημερώσει τις δομές που διατηρεί ο Otel αλγόριθμος έτσι όταν εφαρμοστεί οι ροές θα είναι πλήρως ενημερωμένες.

Σε περίπτωση που υπάρξει σφάλμα θα εντοπιστεί από τον προς τα πάνω κόμβο που βρίσκεται στο αρχικό μονοπάτι και η αποκατάσταση του δικτύου θα γίνει βάση του αλγόριθμου Otel. Ο Otel χρησιμοποιεί τοπική εύρεση τοπικού εναλλακτικού μονοπατιού και επιλέγει τον εναλλακτικό μονοπάτι ανάλογα με τα βάρη των συνδέσεων και το εύρος ζώνης.

Το νέο μονοπάτι θα υπολογίζεται με βάση και τα δυο κριτήρια και έτσι το νέο μονοπάτι θα ικανοποιεί τα βάρη και το εύρος ζώνης. Σε περίπτωση που ακολουθήσουν και άλλα σφάλματα στο νέο μονοπάτι, το οποίο χρησιμοποιείται για τη δρομολόγηση πακέτων, η αποκατάσταση του δικτύου θα συνεχιστεί να γίνεται με βάση τον Otel αλγόριθμο. Η διαδικασία αυτή θα γίνεται και θα ισχύει για κάθε ροή που εισέρχεται στο δίκτυο.

5.4.1 Ψευδοκώδικας Υβριδικού Αλγορίθμου

Η λειτουργία του υβριδικού αλγόριθμου (hybrid algorithm) περιγράφεται από τον ακόλουθο ψευδοκώδικα.

Αρχικά:

Υπολογίζεται ένα σύνολο κοντινότερων μονοπατιών. Γίνεται ο καταμερισμός της χωρητικότητας του πυρήνα. Δίνεται στους ακρινούς κόμβους ένα ποσοστό του εύρους ζώνης του πυρήνα.. Κάθε ακρινός κόμβος διανέμει το εύρος ζώνης που του αναλογεί στις συνδέσεις του που τον ενώνουν με τον πυρήνα αναλογικά με το εύρος ζώνης αυτών των συνδέσεων.

Υπολογίζονται:

1. SPT
2. Πίνακας με τις αποστάσεις των κόμβων μεταξύ τους
3. Πίνακας με το αρχικό μονοπάτι

4. Βάρη κάθε σύνδεσης
5. Διαθέσιμο εύρος ζώνης κάθε σύνδεσης

Κάθε LSR εδραιώνει όσο το δυνατό πιο ανεξάρτητα μονοπάτια ανάμεσα σε αυτό και τους ακρινούς κόμβους και ανάμεσα σε αυτό και τους άλλους δρομολογητές μέσα στο δίκτυο, χρησιμοποιώντας την self-discovery διαδικασία με τις πληροφορίες της IP δρομολόγησης. Εδραιώνονται πριν από οποιαδήποτε λειτουργία του δικτύου, χωρίς όμως να δεσμεύεται εύρος ζώνης.

Με την είσοδο της ροής στο δίκτυο γνωρίζοντας τις απαιτήσεις της σε εύρος ζώνης εφαρμόζεται ο αλγόριθμος WSP όπου θα επιλέξει την πλέον κατάλληλη από τις υπάρχουσες διαδρομές σύμφωνα με τις απαιτήσεις της για εύρος ζώνης και το διαθέσιμο που έχει κάθε υπολογισμένο LSP και αποτελεί την κύρια διαδρομή ενώ επιλέγεται με τον ίδιο τρόπο και το εναλλακτικό μονοπάτι της ροής. Μόλις βρει την κατάλληλη ροή γίνεται και η εδραίωση του μονοπατιού και έτσι η ροή ξεκινά από τον κόμβο εισόδου προς τον κόμβο προορισμό.

Αν (η χωρητικότητα του πυρήνα είναι μεγαλύτερη από αυτό που μπορεί να στείλει η σύνδεση που ενώνει με τον πυρήνα) τότε

1. Θα αποδέχεται την ροή και θα την δρομολογήσει
2. Αλλαγή διαθέσιμου εύρους ζώνης των συνδέσεων
3. Αλλαγή βαρών των συνδέσεων

νΑ

Αν (η σύνδεση έχει εύρος ζώνης μεγαλύτερο από τη χωρητικότητα του πυρήνα – πλήρης χωρητικότητα) τότε

1. Μη επιτρεπτή η είσοδος επιπρόσθετης κυκλοφορίας ή
2. (α) Διανομή της εισερχόμενης κυκλοφορίας σύμφωνα με την ροή δια μέσου των κατάλληλων διαδρομών που έγινε κατά το πρώτο στάδιο του αλγορίθμου. Οι ροές δεν διαμοιράζονται.
 1. Αλλαγή διαθέσιμων εύρος ζώνης των συνδέσεων
 2. Αλλαγή βαρών κάθε σύνδεσης

ή

(β) Παρακολούθηση των ροών κατά το πρώτο στάδιο και αντικατάσταση των ροών που φεύγουν για να ανανεωθεί η διανομή των διαδρομών.

1. Αλλαγή διαθέσιμων εύρος ζώνης των συνδέσεων
2. Αλλαγή βαρών κάθε σύνδεσης

νΑ

Περίπτωση Λάθους:

Αν (το λάθος παρουσιαστεί στο κύριο μονοπάτι) τότε

Το κύριο μονοπάτι δεν είναι πλέον διαθέσιμο.

νΑ

Αν (το λάθος παρουσιαστεί στο εναλλακτικό μονοπάτι) τότε

Το εναλλακτικό μονοπάτι δεν είναι πλέον διαθέσιμο.

νΑ

Αν (το κύριο μονοπάτι είναι διαθέσιμο & το εναλλακτικό είναι διαθέσιμο) τότε

1. Το SPT διαμορφώνεται με τα νέα δεδομένα
2. Ο πίνακας με τα μεγέθη ενημερώνεται για κάθε κόμβο.
3. Αλλάζουν τα βάρη κάθε σύνδεσης.
4. Αλλαγή διαθέσιμων εύρος ζώνης των συνδέσεων

νΑ

Αν (το κύριο μονοπάτι δεν είναι διαθέσιμο & το εναλλακτικό είναι διαθέσιμο) τότε

1. Το SPT διαμορφώνεται με τα νέα δεδομένα , ο πίνακας με τα μεγέθη ενημερώνεται για κάθε κόμβο.
2. Αλλάζουν τα βάρη κάθε σύνδεσης
3. Το λάθος επιδιορθώνεται σύμφωνα με τον αλγόριθμο LSFSB.

νΑ

Ο ακρινός κόμβος ελέγχει αν το υπολογισμένο εναλλακτικό του μονοπάτι μπορεί να το φιλοξενήσει.

Αν ναι εναλλάσσουν τη ροή σε αυτό,

αλλιώς επαναλαμβάνει τους υπολογισμούς του LSFSB για να υπολογίζει ένα νέο μονοπάτι για να στείλει τη ροή.

Αλλαγή διαθέσιμων εύρους ζώνης των συνδέσεων

νΑ

Αν (το κύριο μονοπάτι είναι διαθέσιμο & το εναλλακτικό δεν είναι διαθέσιμο) τότε

1. Το SPT διαμορφώνεται με τα νέα δεδομένα
2. Ο πίνακας με τα μεγέθη ενημερώνεται για κάθε κόμβο.
3. Αλλαγή διαθέσιμων εύρους ζώνης των συνδέσεων
4. Αλλαγή των βαρών κάθε σύνδεσης

νΑ

Αν (το κύριο μονοπάτι & το εναλλακτικό δεν είναι διαθέσιμα) τότε

1. Το SPT διαμορφώνεται με τα νέα δεδομένα
2. Ο πίνακας με τα μεγέθη ενημερώνεται για κάθε κόμβο.
3. Αλλάζουν τα βάρη κάθε σύνδεσης.
4. Το λάθος επιδιορθώνεται σύμφωνα με τον αλγόριθμο Otel που επιλέγει την νέα ροή λαμβάνοντας υπόψη και το διαθέσιμο εύρος ζώνης των συνδέσεων.
5. Αλλαγή διαθέσιμων εύρους ζώνης των συνδέσεων

νΑ

Επιδιόρθωση λάθους:

1. Το SPT διαμορφώνεται με τα νέα δεδομένα , ο πίνακας με τα μεγέθη ενημερώνεται για κάθε κόμβο.
2. Αλλάζουν τα βάρη κάθε σύνδεσης.
3. Αλλαγή διαθέσιμων εύρους ζώνης των συνδέσεων

Κεφάλαιο 6

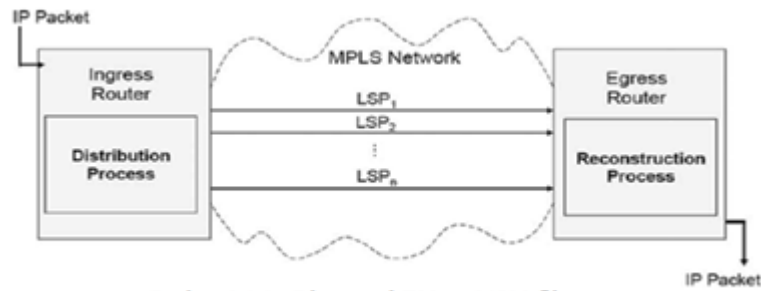
Αλγόριθμος Κατωφλίου (k,n) σε Πολλαπλά Μονοπάτια

6.1 Εισαγωγή	39
6.2 Τεμαχισμός του Αρχικού Πακέτου σε Πολλαπλά Υπό-πακέτα	40
6.3 Ανακατασκευή του Αρχικού Πακέτου από τα Υπό-πακέτα	42
6.4 Πληροφορίες Δικτύου MPLS για την Υλοποίηση του Αλγόριθμου Κατωφλίου	45

6.1 Εισαγωγή

Ο Αλγόριθμος Κατωφλίου (TSS) προτάθηκε από τους Sahel Alouneh, Anjali Agarwal & Abdeslam En-Nouaary και είναι μια εναλλακτική πρόταση για την ανοχή σφαλμάτων εφαρμοσμένη σε δίκτυα MPLS [3] [19]. Αναφέρεται στην προσαρμογή του αλγόριθμου TSS έτσι ώστε να είναι συμβατός σε δίκτυα MPLS. Ο αλγόριθμος TSS διαμοιράζει το αρχικό πακέτο σε n υπό-πακέτα, τα οποία στέλλονται σε πολλαπλά μονοπάτια και ο ελάχιστος αριθμός των υπό-πακέτων που απαιτείται για τη δημιουργία του αρχικού πακέτου ισούται με k ($k < n$). Ένα απλό παράδειγμα της τυπικής λειτουργίας του αλγορίθμου TSS σε MPLS δίκτυα παρουσιάζεται στην Εικόνα 6.1.1, όπου ένα πακέτο IP εισέρχεται στο MPLS δίκτυο και στον πρώτο κόμβο (ingress router) του MPLS δικτύου τεμαχίζεται σε n υπό-πακέτα και ανατίθενται για μεταφορά στα ανεξάρτητα μονοπάτια (τα ανεξάρτητα μονοπάτια στην εικόνα παρουσιάζονται ως τα LSPs). Όταν παραληφθούν k υπό-πακέτα στον τελικό κόμβο (egress router) του δικτύου

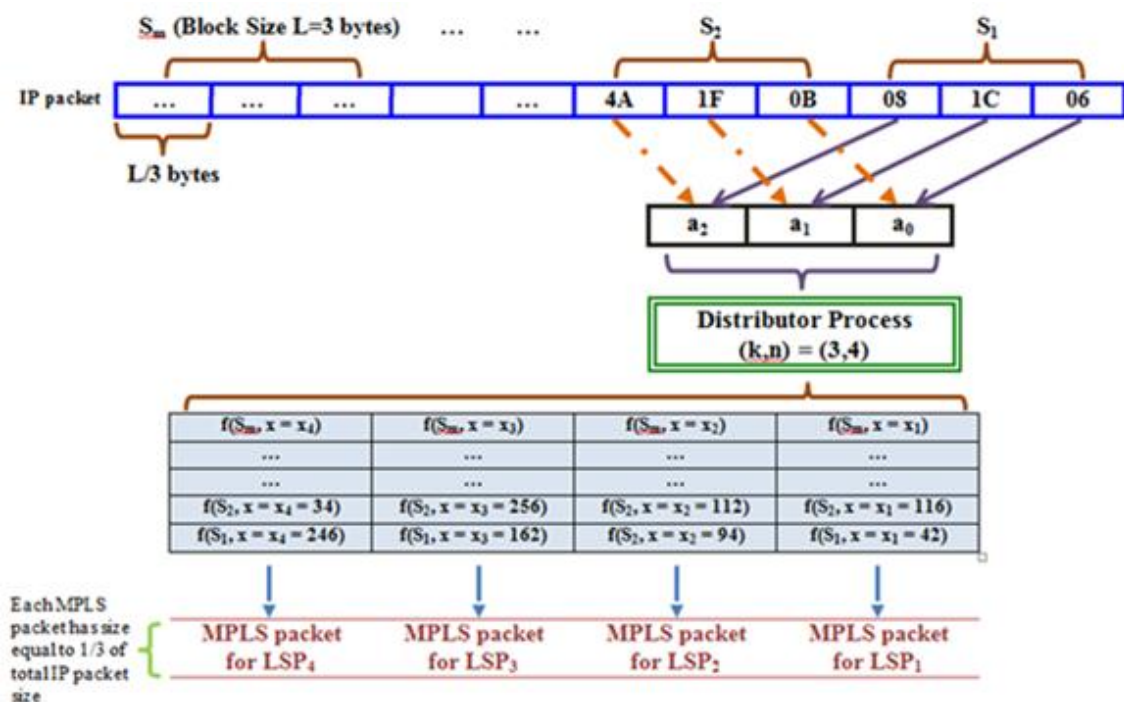
δημιουργείται το αρχικό πακέτο. Οι διαδικασίες για τον τεμαχισμό του πακέτου και τη συναρμολόγηση του θα περιγραφθούν στα επόμενα υποκεφάλαια της ενότητας.



Εικόνα 6.1.1 - Εφαρμογή TSS σε MPLS δίκτυο [3]

6.2 Τεμαχισμός του Αρχικού Πακέτου σε Πολλαπλά Υπό-πακέτα

Με την είσοδο ενός IP πακέτου στο MPLS δίκτυο γίνεται ο τεμαχισμός του σε πολλαπλά υπό-πακέτα. Η διαδικασία αυτή ονομάζεται Distribution Process. Μέσω της Εικόνας 6.2.1 θα γίνει αναλυτική περιγραφή της διαδικασίας τεμαχισμού του πακέτου όταν αυτό εισέλθει στον αρχικό κόμβο (ingress node) του MPLS δικτύου όταν εφαρμόσουμε τον αλγόριθμο κατωφλίου (3,4).



Εικόνα 6.2.1 – Διαδικασία τεμαχισμού του πακέτου στον αρχικό κόμβο του MPLS δικτύου εφαρμόζοντας τον προτεινόμενο αλγόριθμο κατωφλίου (3,4) [3]

Αρχικά το IP πακέτο χωρίζεται σε m μπλοκ (S_1-S_m), όπου το κάθε μπλοκ έχει μέγεθος L (bytes) ίσο με πολλαπλάσιο του αριθμού k . Ο αριθμός k καθορίζεται από τον αλγόριθμο και στην περίπτωση του παραδείγματος ισούται με 3 (αφού το κατώφλι είναι (3,4)). Το κάθε μπλοκ που θα δημιουργηθεί θα διαμοιραστεί στις ροές δρομολόγησης του δικτύου (κάθε ροή δρομολόγησης θα πρέπει να είναι ανεξάρτητη από τις άλλες). Η διεργασία κατανομής (distributor process) του πακέτου χρησιμοποιεί την ακόλουθη πολυωνυμική συνάρτηση για την κωδικοποίηση του περιεχομένου για το κάθε υπό-πακέτο.

$$f(x) = (a_{k-1}x^{k-1} + \dots + a_2x^2 + a_1x + a_0) \bmod p$$

Εικόνα 6.2.2 - Πολυωνυμική Συνάρτηση Κωδικοποίησης υπό-πακέτων [3]

- ▲ Ο αριθμός p είναι ένας πρώτος αριθμός,
- ▲ Οι αριθμοί a_1-a_{k-1} είναι οι συντελεστές της εξίσωσης και καθορίζονται από το κάθε μπλοκ (αντιθέτως από τον πρωτότυπο αλγόριθμο κατωφλίου, οποίος θέτει στους συντελεστές αυτούς τυχαίες τιμές),
- ▲ Ο αριθμός x ορίζει πόσες φορές θα εκτελεστεί η πολυωνυμική εξίσωση. Οι τιμές που παίρνει είναι από 1- n , όπου το n είναι ένας ακέραιος αριθμός που καθορίζεται από τους κόμβους εισόδου και εξόδου. Η μέγιστη τιμή που μπορεί να ισούται περιορίζεται στον αριθμό των ανεξάρτητων διαδρομών που μπορούν να σταλούν τα υπό-πακέτα.

Στην περίπτωση του κατωφλίου (3,4) οι συντελεστές a_1-a_3 για το πρώτο μπλοκ (S_1) ισούνται με $a_0=06$, $a_1=1C$ (σε δεκαδική μορφή είναι ο αριθμός 28) και $a_3=08$, για το δεύτερο μπλοκ (S_2) το $a_0=0B$ (σε δεκαδική μορφή είναι το 11), $a_1=1F$ (σε δεκαδική μορφή είναι το 31) και $a_3=4A$ (σε δεκαδική μορφή είναι το 74) κλπ. Για κάθε μπλοκ οι συναρτήσεις θα υπολογιστούν μέχρι και τέσσερις φορές αλλάζοντας κάθε φορά τη τιμή της μεταβλητής x ($1 \leq x \leq n$, όπου n είναι ο αριθμός των ανεξάρτητων διαδρομών). Έτσι για τα δυο πρώτα μπλοκ προκύπτουν τα εξής αποτελέσματα:

Μπλοκ 1	Μπλοκ 2
$f(S_1, x) = 8x^2 + 28x + 6 \bmod 257$	$f(S_2, x) = 74x^2 + 31x + 11 \bmod 257$
$f(S_{j=1}, x=1) = 42$	$f(S_{j=2}, x=1) = 116$
$f(S_{j=1}, x=2) = 94$	$f(S_{j=2}, x=2) = 112$
$f(S_{j=1}, x=3) = 162$	$f(S_{j=2}, x=3) = 256$
$f(S_{j=1}, x=4) = 246$	$f(S_{j=2}, x=4) = 34$

Με τον υπολογισμό όλων των n πολυωνυμικών συναρτήσεων για όλα τα m μπλοκ προκύπτουν τέσσερα υπό-πακέτα, τα οποία περιέχουν ένα μέρος από κάθε μπλοκ σε κωδικοποιημένη μορφή. Στη γενική περίπτωση το μέγεθος του κάθε υπό-πακέτου που δημιουργείται ισούται με το $1/k$ του αρχικού πακέτου (στην περίπτωση του παραδείγματος ισούται με το $1/3$ του μεγέθους του αρχικού IP πακέτου). Η διαδικασία τεμαχισμού του IP πακέτου ολοκληρώνεται με τη δρομολόγηση των υπό-πακέτων στις αντίστοιχες ροές δρομολόγησης.

Η πολυπλοκότητα του τεμαχισμού του πακέτου που εισέρχεται στον αρχικό κόμβο του δικτύου είναι της τάξεως $O(\frac{a}{b} * c)$, όπου το a ισούται με το μέγεθος του IP πακέτου που εισέρχεται στον κόμβο εισόδου, το b είναι το μέγεθος των μπλοκ που δημιουργούνται από τον τεμαχισμό του αρχικού πακέτου και το c είναι ο αριθμός των LSPs που χρησιμοποιούνται για τη δρομολόγηση των υπό-πακέτων από τον κόμβο εισόδου στον κόμβο εξόδου.

6.3 Ανακατασκευή του Αρχικού Πακέτου από τα Υπό-πακέτα

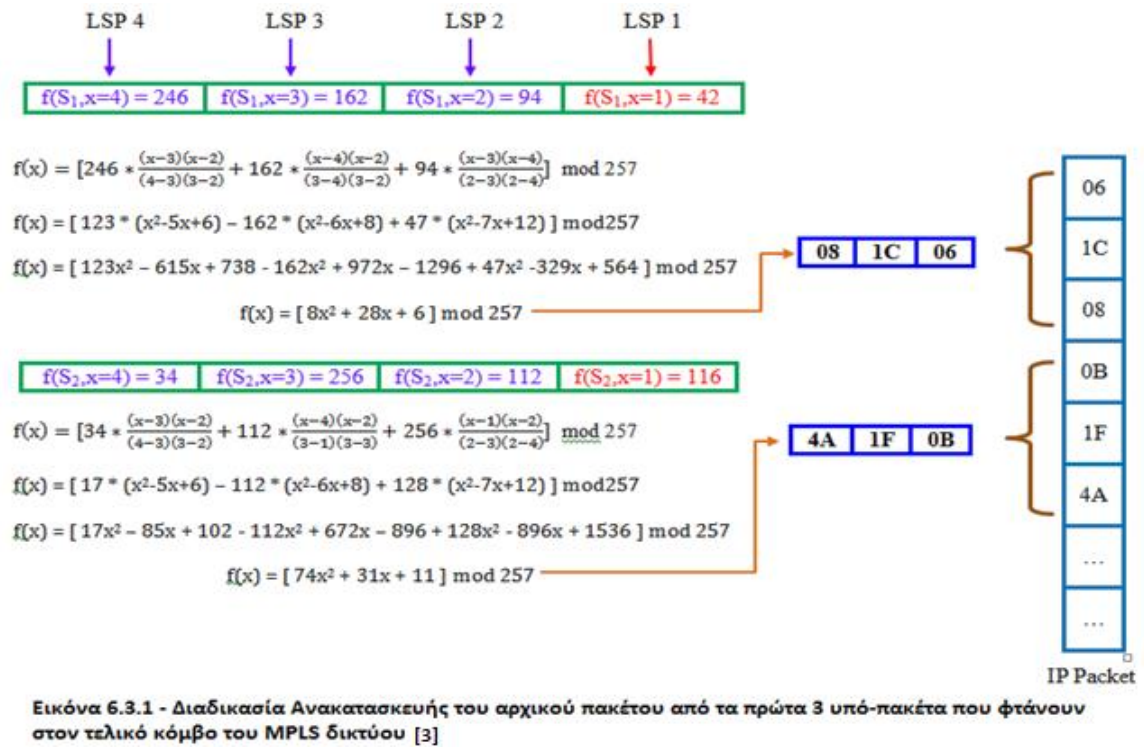
Η παραλαβή ενός πακέτου θεωρείται επιτυχής μόνο όταν παραληφθούν στον τελικό κόμβο (egress node) k υπό-πακέτα από το αρχικό πακέτο. Η παραλαβή αυτών των k υπό-πακέτων πρέπει να γίνεται εντός ενός χρονικού πλαισίου που καθορίζεται από προηγουμένως. Η μέγιστη τιμή που μπορεί να πάρει το πλαίσιο αυτό ισούται με το χρόνο που χρειάζεται το αργότερο μονοπάτι στο δίκτυο να δρομολογήσει το υπό-πακέτο από τον αρχικό στον τελικό κόμβο. Με την παραλαβή των k υπό-πακέτων στον τελικό κόμβο αρχίζει η διαδικασία ανακατασκευής του αρχικού πακέτου. Η διαδικασία αυτή ονομάζεται Reconstruction Process.

Για την ανακατασκευή του αρχικού πακέτου χρησιμοποιείται η απλουστευμένη μέθοδος της Lagrange Interpolation, η Lagrange Linear Interpolation [20] [21]. Η κεντρική ιδέα της μεθόδου αυτής είναι η επίλυση ενός πίνακα για ένα άγνωστο αριθμό x με βάση γνωστών συντελεστών για το x και αποτελεσμάτων των συναρτήσεων που αποτελούν λύση για το x . Δηλαδή είναι ένα σύνολο από εξισώσεις που αθροίζονται μαζί για την εύρεση όλων των τιμών που μπορεί να πάρει η μεταβλητή x για την

επίλυση μιας εξίσωσης. Έτσι χρησιμοποιώντας τη Lagrange Linear Interpolation οι πολυωνυμικές εξισώσεις μπορούν να παρουσιαστούν ως το ακόλουθο άθροισμα:

$$f(x) = \sum_{j=1}^k y_{ij} \prod \frac{x - x_{is}}{x_{ij} - x_{is}}$$

Μέσω της Εικόνας 6.3.1 θα γίνει αναλυτική περιγραφή της διαδικασίας ανακατασκευής του αρχικού πακέτου στον τελικό κόμβο (egress node) του MPLS δικτύου όταν εφαρμόσουμε τον αλγόριθμο κατωφλίου (3,4).



Στο παράδειγμα της εκτέλεσης του προτεινόμενου αλγόριθμου κατωφλίου (3,4) τα πρώτα k υπό-πακέτα (το k ισούται με 3) που φτάνουν στον τελικό κόμβο είναι από τις ροές δρομολόγησης LSP2, LSP3 και LSP4 (η LSP1 είναι η διαδρομή με το μεγαλύτερο χρόνο δρομολόγησης και έτσι δε συμμετέχει το υπό-πακέτο της στην διαδικασία ανακατασκευής). Οι πολυωνυμικές συναρτήσεις που χρησιμοποιούνται από τον αρχικό κόμβο (ingress node) είναι οι ίδιες και για τον τελικό κόμβο (λόγω του ότι στο σχεδιασμό τους συμβάλουν και οι δυο κόμβοι) έτσι οι συντελεστές $a_0 - a_{k-1}$ θα

υπολογιστούν χρησιμοποιώντας τη μέθοδο Lagrange Interpolation. Χρησιμοποιώντας τις τρεις εξισώσεις για τα υπό-πακέτα που παραλήφθηκαν από τον τελικό κόμβο,

- ♣ για το LSP2 η εξίσωση είναι $(\alpha_0 + \alpha_1(2) + \alpha_2(2^2)) = \text{LSP2_value mod } 257$,
- ♣ για το LSP3 η εξίσωση είναι $(\alpha_0 + \alpha_1(3) + \alpha_2(3^2)) = \text{LSP3_value mod } 257$,
- ♣ για το LSP4 η εξίσωση είναι $(\alpha_0 + \alpha_1(4) + \alpha_2(4^2)) = \text{LSP4_value mod } 257$ και

μαζί με τον τύπο της Lagrange μεθόδου θα προκύψουν οι τιμές για τους συντελεστές α_0 - α_2 .

Για το πρώτο μπλοκ οι εξισώσεις έχουν ως εξής:

- ♣ για το LSP2 η εξίσωση είναι $(\alpha_0 + \alpha_1(2) + \alpha_2(2^2)) = 94 \text{ mod } 257$,
- ♣ για το LSP3 η εξίσωση είναι $(\alpha_0 + \alpha_1(3) + \alpha_2(3^2)) = 162 \text{ mod } 257$,
- ♣ για το LSP4 η εξίσωση είναι $(\alpha_0 + \alpha_1(4) + \alpha_2(4^2)) = 246 \text{ mod } 257$

και χρησιμοποιώντας το τύπο της μεθόδου Lagrange προκύπτει η εξής εξίσωση $f(S_1, x) = 8x^2 + 28x + 6$, η οποία αντιπροσωπεύει τους αρχικούς συντελεστές του πρώτου μπλοκ. Το 8 είναι ο συντελεστής α_2 (στη δεκαεξαδική μορφή ισούται με 08), το 28 είναι ο συντελεστής α_1 (στη δεκαεξαδική μορφή είναι το 1C) και το 6 είναι ο συντελεστής α_0 (στη δεκαεξαδική μορφή είναι το 06).

Αντίστοιχα για το δεύτερο μπλοκ οι εξισώσεις έχουν ως εξής:

- ♣ για το LSP2 η εξίσωση είναι $(\alpha_0 + \alpha_1(2) + \alpha_2(2^2)) = 112 \text{ mod } 257$,
- ♣ για το LSP3 η εξίσωση είναι $(\alpha_0 + \alpha_1(3) + \alpha_2(3^2)) = 256 \text{ mod } 257$,
- ♣ για το LSP4 η εξίσωση είναι $(\alpha_0 + \alpha_1(1) + \alpha_2(1^2)) = 34 \text{ mod } 257$

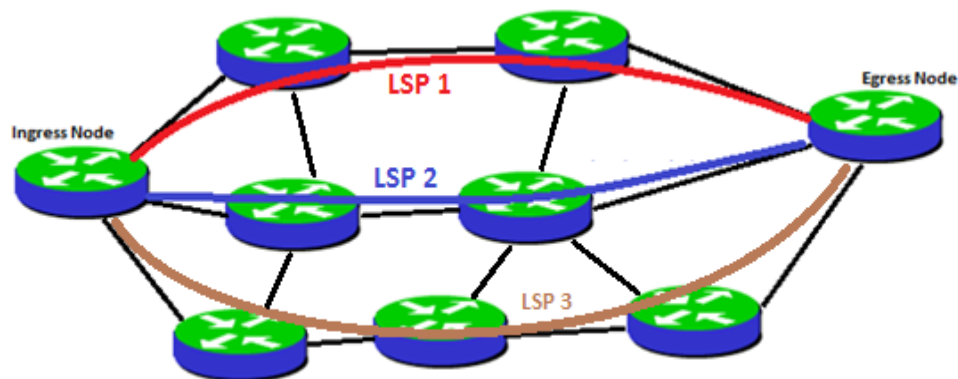
και χρησιμοποιώντας το τύπο της μεθόδου Lagrange προκύπτει η εξής εξίσωση $f(S_2, x) = 74x^2 + 31x + 11$, η οποία αντιπροσωπεύει τους αρχικούς συντελεστές του πρώτου μπλοκ. Το 74 είναι ο συντελεστής α_2 (στη δεκαεξαδική μορφή ισούται με 4A), το 31 είναι ο συντελεστής α_1 (στη δεκαεξαδική μορφή είναι το 1F) και το 11 είναι ο συντελεστής α_0 (στη δεκαεξαδική μορφή είναι το 0B).

Η πολυπλοκότητα της ανακατασκευής του αρχικού πακέτου είναι της τάξεως του $O(b \cdot a^3)$, όπου το a είναι ο αριθμός των υπό-πακέτων που χρειάζονται για την ανακατασκευή του αρχικού πακέτου και το b είναι ο αριθμός των μπλοκ που αποτελούν το αρχικό πακέτο. Η πολυπλοκότητα της Lagrange Linear Interpolation είναι $O(a^3)$.

6.4 Πληροφορίες Δικτύου MPLS για την Υλοποίηση του Αλγόριθμου Κατωφλίου

6.4.1 Εύρεση n Ανεξάρτητων Διαδρομών Δρομολόγησης

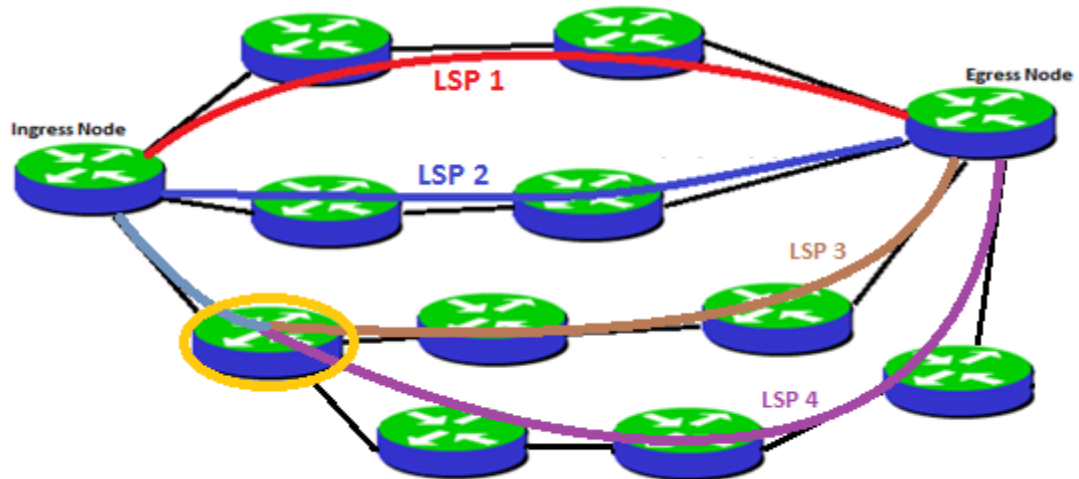
Η δρομολόγηση των υπό-πακέτων μέσα στο MPLS δίκτυο γίνεται μέσω των n ανεξάρτητων διαδρομών δρομολόγησης που προκύπτουν από το υφιστάμενο MPLS δίκτυο. Οι ανεξάρτητες διαδρομές δρομολόγησης αποτελούνται από τους κόμβους του MPLS δικτύου και δε μπορεί ένας κόμβος να ανήκει σε περισσότερες από μια διαδρομές. Για τον υπολογισμό n ανεξάρτητων διαδρομών δρομολόγησης μπορεί να χρησιμοποιηθεί οποιαδήποτε προσέγγιση, η οποία μπορεί να προσφέρει αυτή τη δυνατότητα σε ένα MPLS δίκτυο. Μερικοί από αυτούς είναι ο τροποποιημένος αλγόριθμος του Dijkstra, ο οποίος προτάθηκε από τον Bhandari [22], η πολυωνυμική επίλυση γράφου για ανεξάρτητα μονοπάτια από τον Shiloach [23]. Για παράδειγμα ο τροποποιημένος αλγόριθμος του Dijkstra επεκτείνει τη δυνατότητα του αρχικού αλγορίθμου του Dijkstra για εύρεση ανεξάρτητων μονοπατιών σε ένα δίκτυο από δυο μονοπάτια σε περισσότερα από δυο. Αυτό επιτυγχάνεται με το τέλος n επαναλήψεων εκτέλεσης του τροποποιημένου αλγορίθμου Dijkstra. Παράδειγμα ενός δικτύου MPLS με n ανεξάρτητες διαδρομές δρομολόγησης παρουσιάζεται στην Εικόνα 6.4.1.1 (ο αριθμός των μέγιστων ανεξάρτητων διαδρομών δρομολόγησης ισούται με 3).



Εικόνα 6.4.1.1 - MPLS Δίκτυο με n ανεξάρτητες διαδρομές δρομολόγησης ($n=3$)

Στις περιπτώσεις που δεν προσφέρονται n ανεξάρτητες διαδρομές από το δίκτυο χωρίς να υπάρχει καταπάτηση του κανόνα, ότι ο κάθε κόμβος πρέπει να ανήκει μόνο σε μια διαδρομή δρομολόγησης, εφαρμόζεται η τεχνική εύρεσης των μέγιστων δυνατών διαδρομών δρομολόγησης. Στην εύρεση μέγιστων δυνατών διαδρομών δρομολόγησης ένας κόμβος μπορεί να ανήκει σε περισσότερες από μια διαδρομές μόνο και μόνο αν

από τον κόμβο αυτό αρχίζουν δυο νέες διαδρομές δρομολόγησης. Οποιαδήποτε άλλη περίπτωση, όπως δυο ή περισσότερες διαδρομές δρομολόγησης να καταλήγουν σε ένα κοινό κόμβο του δικτύου, δεν επιτρέπεται. Στην Εικόνα 6.4.1.2 παρουσιάζεται ένα παράδειγμα όπου ένας κόμβος (ο κόμβος που βρίσκεται στο πορτοκαλί πλαίσιο) αποτελεί την αρχή για δυο ανεξάρτητες διαδρομές δρομολόγησης (τις LSP3 και LSP4).



Εικόνα 6.4.1.2 - MPLS Δίκτυο με μέγιστο αριθμό ανεξάρτητων διαδρομών δρομολόγησης

Επίσης στις περιπτώσεις αυτές (που δεν προσφέρονται η ανεξάρτητες διαδρομές από το δίκτυο χωρίς να υπάρχει καταπάτηση του κανόνα) συνεπάγεται ότι σε περίπτωση σφάλματος δεν εγγυάται η αξιόπιστη μετάδοση του μηνύματος όπως θα συνέβαινε σε περίπτωση με εντελώς ανεξάρτητα μονοπάτια. Αυτό οφείλεται στο ότι όταν υπάρξει σφάλμα σε κόμβο, ο οποίος αποτελεί την δημιουργία δύο ή περισσότερων διαδρομών δρομολόγησης (disjoint node) πλέον δεν επηρεάζεται μόνο μια διαδρομή δρομολόγησης πακέτων αλλά όσες δημιουργούνται από τον κόμβο αυτό. Με αποτέλεσμα όταν δεν ικανοποιείται το κριτήριο του κατωφλίου για την ανακατασκευή του αρχικού πακέτου, το πακέτο αυτό να χάνεται.

6.4.2 Καθορισμός Ροών Δρομολόγησης και Buffer Τελικού Κόμβου

Ένα από τα πιο σημαντικά προβλήματα που πρέπει να αντιμετωπίσει ο αλγόριθμος κατά τη δρομολόγηση των υπό-πακέτων στο δίκτυο είναι η μη ισότητα των μεγεθών των διαδρομών δρομολόγησης. Το πρόβλημα που δημιουργείται από αυτό είναι ότι κατά τη δρομολόγηση των υπό-πακέτων στο δίκτυο υπάρχουν μερικά υπό-

πακέτα που μπορεί να φτάσουν στον τελικό κόμβο πιο γρήγορα από άλλα και να περιμένουν εκεί μέχρι να καλυφθεί το απαιτούμενο κατώφλι. Έτσι για να μπορεί ο τελικός κόμβος να ανακατασκευάσει το αρχικό πακέτο από τα αντίστοιχα υπό-πακέτα πρέπει να αποθηκεύει στο buffer τα υπό-πακέτα αυτά. Το buffer που απαιτείται να έχει ο τελικός κόμβος πρέπει να ισούται με τον αριθμό όλων των υπό-πακέτων που μπορούν να φτάσουν σε αυτό μέχρι να φτάσει το πρώτο υπό-πακέτο από την πιο χρονοβόρα διαδρομή δρομολόγησης. Γι' αυτό στις περιπτώσεις όπου δεν πληρείται το κριτήριο του κατωφλίου για την ανακατασκευή του αρχικού πακέτου και ο χρόνος που είναι αποθηκευμένα τα υπό-πακέτα αυτού του πακέτου ξεπεράσει το χρόνο που χρειάζεται το υπό-πακέτο που δρομολογείται από την μεγαλύτερη διαδρομή να φτάσει στον τελικό κόμβο απορρίπτονται και χάνονται από το buffer του τελικού κόμβου. Όταν η διαδρομή δρομολόγησης με το μεγαλύτερο χρόνο δρομολόγησης γίνει για οποιοδήποτε λόγο (σφάλμα σε κόμβο ή σύνδεση κλπ) μη διαθέσιμη τότε ο χρόνος που θα περιμένει ο τελικός κόμβος πριν απορρίψει οποιαδήποτε αποθηκευμένα υπό-πακέτα από το buffer του θα ισούται με τα χρόνο της επόμενης χρονοβόρας διαδρομής δρομολόγησης από αυτή που έγινε μη διαθέσιμη.

Για παράδειγμα με βάση την Εικόνα 6.4.1.1 θα υπολογίσουμε το μέγεθος του buffer που πρέπει να έχει ο τελικός κόμβος βάση του χρόνου που απαιτείται για τη δρομολόγηση των υπό-πακέτων από τις διαδρομές δρομολόγησης.

Οι διαδρομές δρομολόγησης (LSP 1, LSP 2 και LSP 3) χρειάζονται τους εξής χρόνους για να μεταφέρουν ένα υπό-πακέτο από τον αρχικό κόμβο στον τελικό:

- ▲ η διαδρομή LSP 1 χρειάζεται 20ms,
- ▲ η διαδρομή LSP 2 χρειάζεται 20ms και
- ▲ η διαδρομή LSP 3 χρειάζεται 25ms.

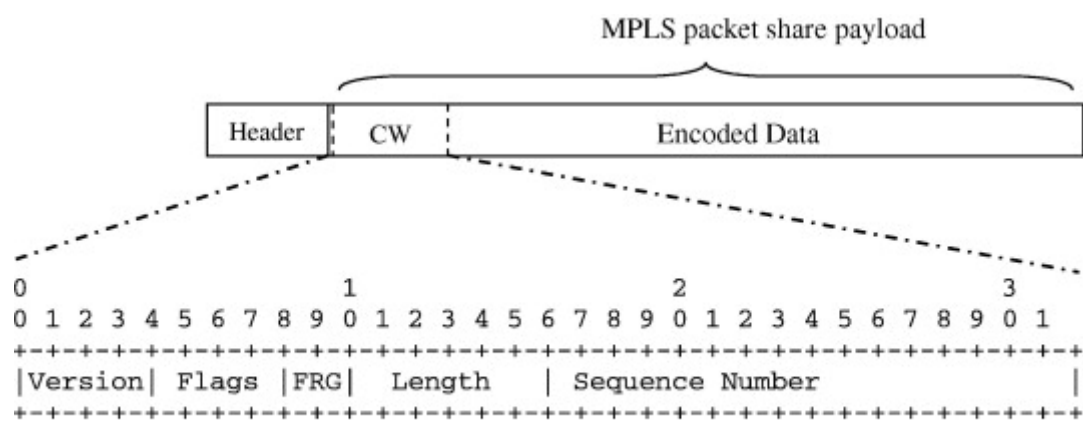
Ο μέγιστος χρόνος που απαιτείται για να φτάσει ένα υπό-πακέτο στον τελικό κόμβο είναι 25ms. Έτσι το buffer του τελικού κόμβου πρέπει να είναι σε θέση να αποθηκεύσει 1280 Megabits. Αυτό προκύπτει όπως αναφέραμε προηγουμένως από το ότι το μέγεθος του buffer πρέπει να ισούται με το συνολικό μέγεθος όλων των υπό-πακέτων που στέλλονται από τις άλλες διαδρομές (εξαιρείται η διαδρομή που χρειάζεται το μεγαλύτερο χρόνο για τη δρομολόγηση του υπό-πακέτου) μέχρι να φτάσει το πρώτο υπό-πακέτο από τη διαδρομή με το μεγαλύτερο χρόνο δρομολόγησης. Έτσι έχουμε για τις διαδρομές LSP 1 και LSP 2, το συνολικό μέγεθος των πακέτων που στέλλει η κάθε

διαδρομή στον τελικό κόμβο μέχρι το 25° ms που θα φτάσει το πρώτο πακέτο της χρονοβόρας διαδρομής δρομολόγησης ισούται με $(t_{\max} - t_{\text{cur}}) * P_{\text{size}}$, όπου $t_{\max}$ είναι ο χρόνος δρομολόγησης από τη μεγαλύτερη διαδρομή, t_{cur} είναι ο χρόνος δρομολόγησης από την υφιστάμενη διαδρομή που θέλουμε να υπολογίσουμε και P_{size} είναι το μέγεθος του κάθε πακέτου (η μονάδα μέτρησης των πακέτων ανάλογα με το κάθε δίκτυο που εφαρμόζεται ο προτεινόμενος αλγόριθμος μπορεί να είναι bits, Megabits, Gigabits κλπ). Για τη διαδρομή LSP 1 ισχύει $(25-20)*128 = 640$ Megabits και για τη διαδρομή LSP 2 ισχύει $(25-20)*128 = 640$ Megabits. Έτσι το μέγεθος του buffer για τον τελικό κόμβο θα πρέπει να ισούται με 1280 Megabits.

6.4.3 Καθορισμός Επικεφαλίδας Υπό-πακέτων

Όταν το αρχικό IP πακέτο τεμαχιστεί στον αρχικό κόμβο στο κάθε υπό-πακέτο που δημιουργείται προστίθεται μια επικεφαλίδα. Η επικεφαλίδα αυτή (μεγέθους 4 bytes) περιέχει πληροφορίες σχετικά με τη δρομολόγηση του από τον αρχικό κόμβο προς τον τελικό (πίνακας δρομολόγησης) καθώς και μια τεχνική που υιοθετήθηκε από την έρευνα Control Word for Use over an MPLS PSN [24]. Η επιπρόσθετη προσθήκη του Control Word (CW) μαζί με την επικεφαλίδα του υπό-πακέτου προσφέρει κυρίως τη δυνατότητα αριθμό ακολουθίας κάθε υπό-πακέτου που αποτελεί το αρχικό IP πακέτο. Αναλυτικά το CW περιέχει τις εξής πληροφορίες:

- ▲ Version (bits 0-3): Χρησιμοποιείται για τον καθορισμό της έκδοσης του πρωτοκόλλου διαδικτύου (στην περίπτωση του IPv4 παίρνει την τιμή 4, στην περίπτωση του IPv6 παίρνει την τιμή 6 κλπ)
- ▲ Flags (bits 4-7): Χρησιμοποιείται για τη σηματοδότηση per-pay-load
- ▲ FRG (bits 8-9): Χρησιμοποιείται όταν τα υπό-πακέτα μοιράζονται πόρους για τη δρομολόγηση τους (δηλαδή στις περιπτώσεις που κάποιος κόμβος δημιουργεί δυο ή περισσότερες ροές δρομολόγησης)
- ▲ Length (bits 10-15): Χρησιμοποιείται όταν το μονοπάτι δρομολόγησης μεταξύ του αρχικού και τελικού κόμβου περιέχει τμήμα τύπου Ethernet
- ▲ Sequence Number (bits 16-31): Χρησιμοποιείται για την αρίθμηση των υπό-πακέτων που αποτελούν το αρχικό πακέτο



Εικόνα 6.4.3.1 - Περιγραφή ετικέτας Control Word (CW) [3]

Κεφάλαιο 7

Πειραματική Αξιολόγηση

7.1 Προσομοιωτής Network Simulator 2	50
7.2 Παράμετροι Προσομοίωσης	52
7.3 Παράμετροι Αξιολόγησης	59
7.4 Αποτελέσματα Προσομοίωσης	60
7.5 Συμπεράσματα	83

7.1 Προσομοιωτής Network Simulator 2

Για την προσομοίωση της λειτουργίας του αλγόριθμου TSS σε πολλαπλά μονοπάτια δρομολόγηση χρησιμοποιήθηκε ο προσομοιωτής δικτύων **Network Simulator 2** [25] [26] [27] [28] [29] [30].

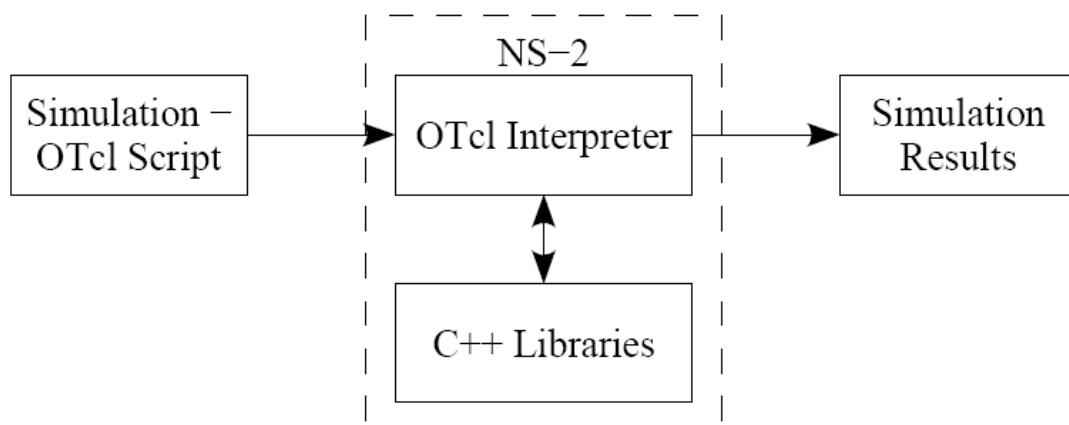
Ο προσομοιωτής NS 2 αναπτύχθηκε το 1996 από τον Steve McCanne και αποτελεί ένα από τους πιο δημοφιλείς προσομοιωτές δικτύων, ο οποίος δίνει τη δυνατότητα σε οποιονδήποτε χρήστη να τροποποιήσει τον πηγαίο κώδικα για να υλοποιήσει διαφορετικές λειτουργίες ή ακόμα να επεκτείνει μερικές από αυτές. Υλοποιημένος με δυο γλώσσες (τη C++ και τη script γλώσσα OTcl – αποτελεί επέκταση της Tcl) μπορούμε να ορίσουμε ταυτόχρονα τόσο τους εσωτερικούς μηχανισμούς της προσομοίωσης (γλώσσα C++) όσο και την δημιουργία, τροποποίηση

και εκτέλεση των σεναρίων προσομοίωσης αφού πρώτα οριστούν τα χαρακτηριστικά του δικτύου (τοπολογία, πρωτόκολλα δρομολόγησης, μηχανισμοί ουράς, κ.ά.).

Μερικές από τις βασικές λειτουργίες του NS 2 είναι:

- ▲ Η υλοποίηση πρωτοκόλλων δικτύου, όπως TCP και UDP
- ▲ Προσομοίωση κυκλοφορίας, όπως FTP
- ▲ Μηχανισμούς διαχείρισης ουράς, όπως Drop Tail
- ▲ Υλοποίηση μηχανισμών MPLS, Multi-Cast κ.ά.

Από την προοπτική των χρηστών (χωρίς να λαμβάνετε υπόψη η αρχιτεκτονική του προσομοιωτή NS 2), ο NS 2 είναι ένα διερμηνέας τύπου OTcl, ο οποίος λαμβάνει ως είσοδο ένα OTcl σενάριο και εξάγει ένα αρχείο (trace file) ως έξοδο (Εικόνα 7.1.1). Αυτά τα αρχεία (trace files) περιέχουν τα αποτελέσματα της προσομοίωσης, τα οποία χρησιμοποιούνται σε μετέπειτα στάδιο για την έκδοση γραφικών παραστάσεων με το εργαλείο XGraph καθώς και για την μελέτη πληροφοριών, όπως τη ρυθμαπόδοση, το ποσοστό επιτυχίας αποστολής πακέτων, το συνολικό αριθμό παραλαβής πακέτων κ.ά. Επίσης χρησιμοποιείται το εργαλείο NAM για τη γραφική αναπαράσταση του σεναρίου κάθε τοπολογίας.



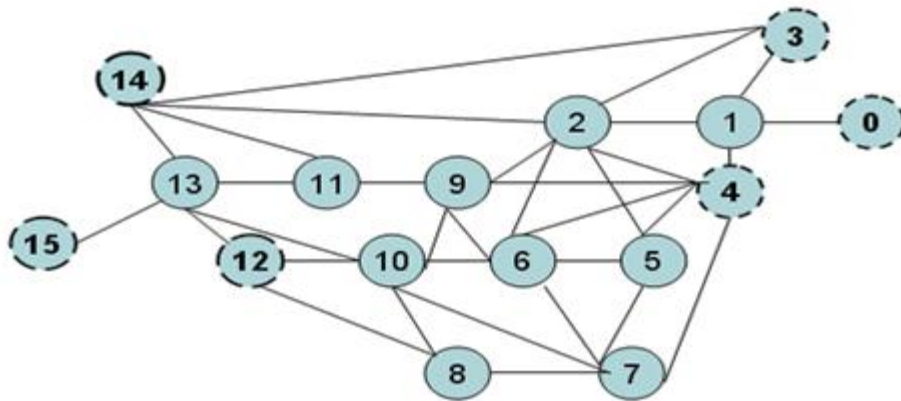
Εικόνα 7.1.1. - Λειτουργία NS 2

7.2 Παράμετροι Προσομοίωσης

7.2.1 Τοπολογίες Δικτύων Προσομοίωσης

Πιο κάτω παρουσιάζονται οι τοπολογίες των δικτύων που προσομοιώθηκαν για την εξόρυξη των αποτελεσμάτων και την περαιτέρω μελέτη τους. Όλες οι τοπολογίες υποστηρίζουν πολλαπλές ροές.

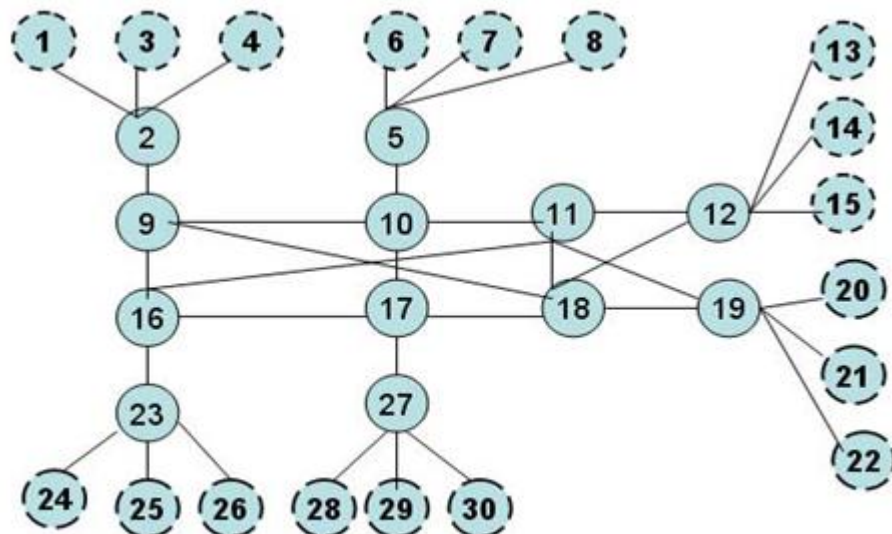
▲ Τοπολογία 1 – Επικοινωνιακό Δίκτυο ΗΠΑ



Εικόνα 7.2.1.1 - Τοπολογία του πρώτου δικτύου (ΗΠΑ)[4]

Σχόλια: Ακρινοί κόμβοι εισόδου θεωρήθηκαν οι κόμβοι 0, 3 & 4. Ακρινοί κόμβοι εξόδου θεωρήθηκαν οι κόμβοι 12, 14 & 15. Χρησιμοποιήθηκε για την προσομοίωση του υβριδικού και του TSS με πολλαπλά μονοπάτια αλγόριθμος.

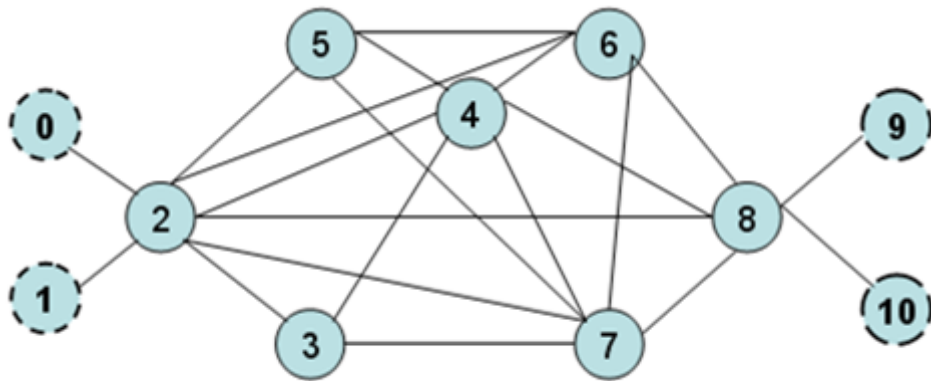
▲ Τοπολογία 2



Εικόνα 7.2.1.2 - Τοπολογία του δεύτερου δικτύου [4]

Σχόλια: Ακρινοί κόμβοι εισόδου θεωρήθηκαν οι 1, 3, 4, 6, 7, 8, 13, 14, 15, 20, 21 & 22 και κόμβοι εξόδου θεωρήθηκαν οι 24, 25, 26, 28, 29 & 30. Χρησιμοποιήθηκε για την προσομοίωση του υβριδικού και του TSS με πολλαπλά μονοπάτια αλγορίθμου.

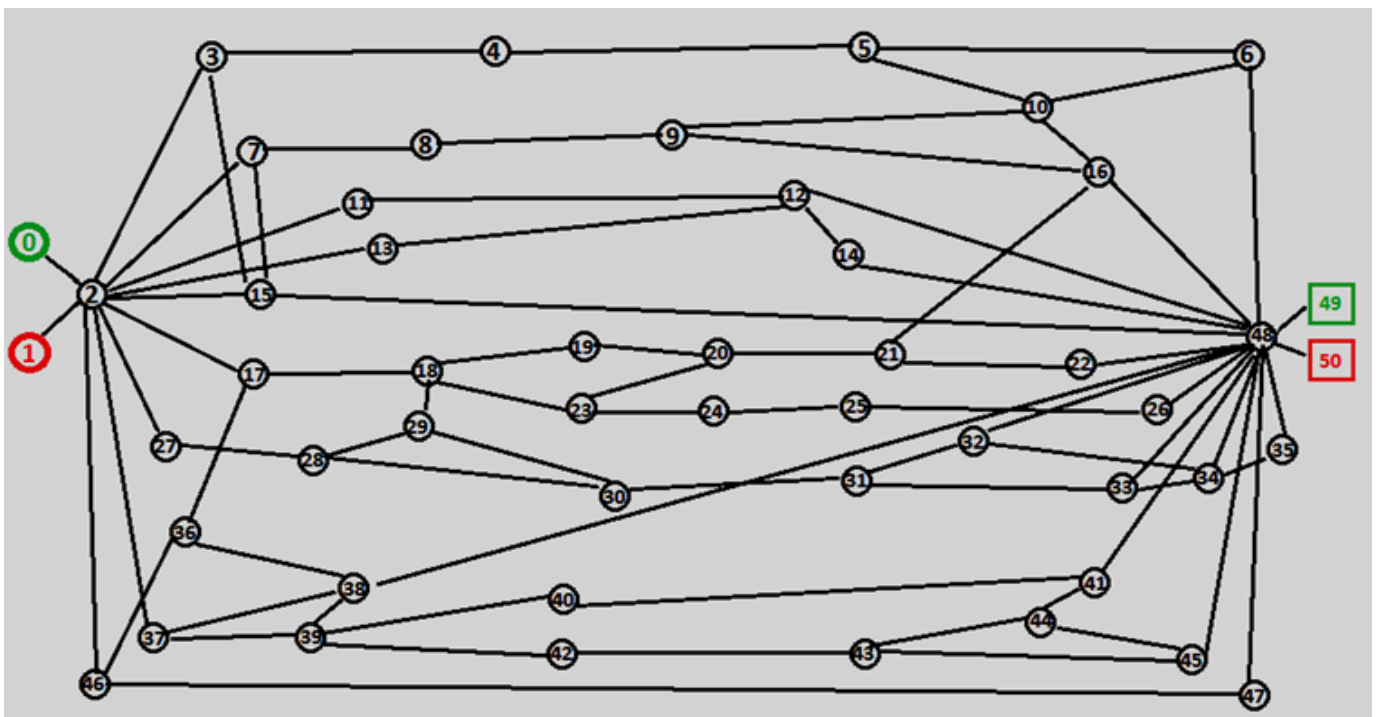
▲ Τοπολογία 3



Εικόνα 7.2.1.3 - Τοπολογία του τρίτου δικτύου [4]

Σχόλια: Ακρινοί κόμβοι εισόδου θεωρήθηκαν οι 0 & 1 και κόμβοι εξόδου θεωρήθηκαν οι 9 & 10. Χρησιμοποιήθηκε για την προσομοίωση του υβριδικού και του TSS με πολλαπλά μονοπάτια αλγορίθμου.

▲ Τοπολογία 4



Εικόνα 7.2.1.4 - Τοπολογία του τέταρτου δικτύου

Σχόλια: Ακρινοί κόμβοι εισόδου θεωρήθηκαν οι 0 & 1 και κόμβοι εξόδου θεωρήθηκαν οι 49 & 50. Χρησιμοποιήθηκε για την προσομοίωση του αλγορίθμου TSS με πολλαπλά μονοπάτια.

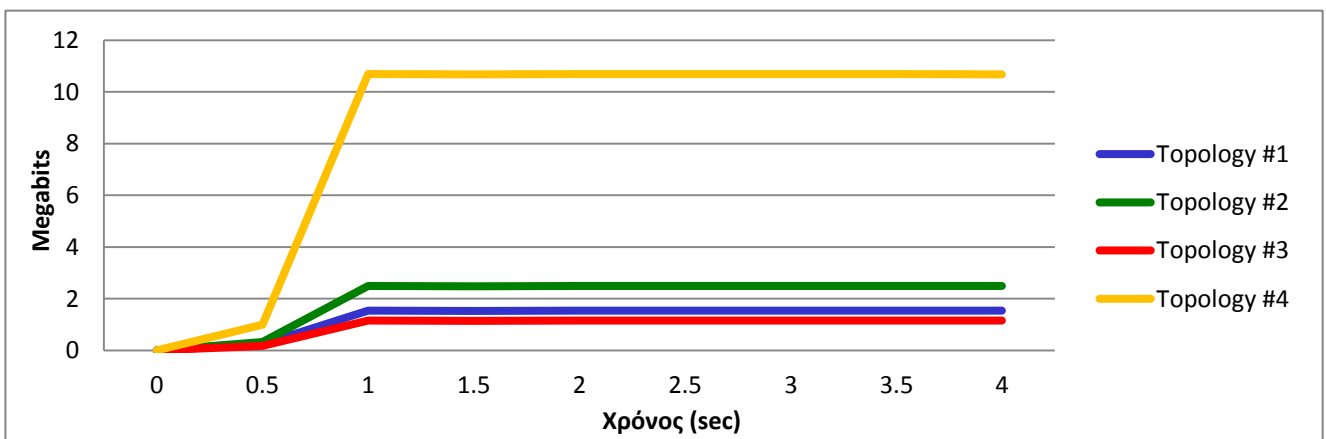
Η Τοπολογία 4 σχεδιάστηκε από εμένα και είχε ως σκοπό τη μελέτη του αλγορίθμου TSS με πολλαπλά μονοπάτια σε τοπολογία με μεγάλο αριθμό δρομολογητών να αποτελούν τον πυρήνα του δικτύου. Οι δρομολογητές του δικτύου σχηματίζουν δεκαπέντε ανεξάρτητα μονοπάτια (ελάχιστοι είναι οι κόμβοι που δεν ανήκουν σε κάποιο ανεξάρτητο μονοπάτι), όπου μερικά από αυτά τα μονοπάτια τα περισσότερα δημιουργούνται από κόμβους τύπου disjoint. Δηλαδή από ένα δρομολογητή του πυρήνα του δικτύου προέκυπταν δυο νέα μονοπάτια δρομολόγησης. Η εισαγωγή αυτού του τύπου κόμβων στο δίκτυο θα μας βοηθήσει να παρατηρήσουμε κατά πόσο ο TSS μπορεί να ανταπεξέλθει σε σφάλματα αυτού του τύπου (ένα σφάλμα σε κόμβο τύπου disjoint επηρεάζει δυο ροές δρομολόγησης).

7.2.2 Πληροφορίες Δικτύου

Πέραν των τοπολογιών που προαναφέραμε εξίσου σημαντικό για την προσομοίωση των δικτύων είναι και ο καθορισμός κάποιων παραμέτρων, οι οποίες συμβάλουν σημαντικά στον τρόπο με τον οποίο θα προσομοιωθεί το δίκτυο στο NS 2.

- ▲ Οι συνδέσεις έχουν εύρος ζώνης 2MB και ο χρόνος καθυστέρησης που δίνεται στις ροές είναι 1ms.
- ▲ Όλες οι ροές δημιουργούν πακέτα των 200 bytes και ο ρυθμός με τον οποίο στέλνονται τα πακέτα είναι 400KB.
- ▲ Οι κόμβοι υποστηρίζουν το πρωτόκολλο UDP και η εφαρμογή της ροής είναι CBR, έτσι με αυτό τον τρόπο επιτυγχάνουμε τα πακέτα να δημιουργούνται με σταθερό ρυθμό.
- ▲ Δεν χρησιμοποιήθηκε κάποιο συγκεκριμένο είδος ουράς, αφού με βάση τον προτεινόμενο αλγόριθμο (TSS (k,n) με πολλαπλά μονοπάτια) μια από τις υποθέσεις που γίνονται είναι ότι δεν υπάρχουν ουρές στους δρομολογητές, αφού όλες οι ροές έχουν την ίδια χωρητικότητα αποστολής πακέτων.

Πιο κάτω (Σχήμα 7.2.2.1) παρουσιάζουμε τον όγκο των δεδομένων που μεταφέρονται στις συνδέσεις και στις τέσσερις τοπολογίες σε σχέση με το χρόνο. Οι μετρήσεις κάθε τοπολογίας δείχνουν το μέγιστο όγκο δεδομένων που μεταφέρονται στις συνδέσεις του κάθε δικτύου όταν δεν υπάρχουν σφάλματα σε οποιαδήποτε σύνδεση μεταξύ των κόμβων (ο μέγιστος όγκος δεδομένων είναι μεγαλύτερος μόνο στην περίπτωση που το κατώφλι ισούται με ένα γιατί τα υπό-πακέτα θα έχουν μέγεθος ίσο με το μέγεθος του αρχικού πακέτου που στέλλεται από τις πηγές). Παρατηρούμε ότι ο μεγαλύτερος όγκος δεδομένων βρίσκεται στην 4^η τοπολογία που φτάνει μέχρι και τα 10,69 Megabits ενώ ο μικρότερος όγκος δεδομένων βρίσκεται στην 3^η τοπολογία με 1,178 Megabits. Στις τοπολογίες 1 και 2 ο όγκος δεδομένων ανέρχεται στο 1,535 και 2,488 Megabits αντίστοιχα.



Σχήμα 7.2.2.1 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στις τοπολογίες

Τα αποτελέσματα αυτά οφείλονται στο γεγονός ότι στην τοπολογία 4, υπάρχουν οι περισσότερες ανεξάρτητες διαδρομές στις οποίες ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια στέλνει τα υπό-πακέτα του αρχικού πακέτου. Με αποτέλεσμα να παρατηρείται και αυξημένος αριθμός των δεδομένων που μεταφέρονται στις συνδέσεις το δευτερόλεπτο.

7.2.3 Σενάρια

Για τις τοπολογίες ένα μέχρι τρία έχουν εφαρμοστεί τέσσερα διαφορετικά σενάρια, ενώ για την τοπολογία τέσσερα εφαρμόστηκαν δυο σενάρια. Σε όλες όμως τις

περιπτώσεις λόγω του ότι ελέγχθηκε και η μηχανική κυκλοφορία υπήρχαν περισσότερες από μια ροές (δυο πηγές αποστολής δεδομένων με τους αντίστοιχους παραλήπτες) καθώς υπήρξαν και μερικά "τεχνητά" σφάλματα (τερματισμός λειτουργίας μερικών κόμβων/συνδέσεων), έτσι ώστε να παρατηρήσουμε τη συμπεριφορά του δικτύου πριν και μετά τα σφάλματα.

Σκοπός των σεναρίων είναι η εξαγωγή συμπερασμάτων για την συμπεριφορά του αλγορίθμου TSS με πολλαπλά μονοπάτια όσον αφορά την ανοχή σφαλμάτων και τη μηχανική κυκλοφορίας. Μετέπειτα θα συγκριθούν με τα αποτελέσματα του υβριδικού αλγορίθμου που προαναφέραμε.

Για τα πρώτα τέσσερα σενάρια των τοπολογιών ένα μέχρι τρία επικρατεί η ίδια γενική ιδέα, αφού τα αποτελέσματα αυτών των σεναρίων θα συγκριθούν με τα αποτελέσματα του υβριδικού αλγορίθμου. Πιο συγκεκριμένα, σύμφωνα με τα σενάρια του υβριδικού αλγορίθμου σε κάθε τοπολογία τα λάθη συμβαίνουν σε συνδέσεις που είναι στα εναλλακτικά ή κυρίως μονοπάτια των ροών όπως αυτά έχουν οριστεί από τον αλγόριθμο LSFSB και όχι στις ίδιες συνδέσεις για όλα τα δίκτυα. Σύμφωνα με τον ορισμό του αλγορίθμου TSS (k, n) με πολλαπλά μονοπάτια διαμορφώσαμε την κάθε τοπολογία για όλα τα σενάρια να πληροί τα κριτήρια του TSS. Δηλαδή βρήκαμε το μέγιστο αριθμό διαδρομών από τον ingress κόμβο προς τον egress κόμβο, όπου κάθε διαδρομή αποτελείται από ανεξάρτητους κόμβους (κόμβοι οι οποίοι δεν ανήκουν σε οποιανδήποτε άλλη διαδρομή). Τα σφάλματα εμφανίζονται κατά τα χρονικά διαστήματα που γίνονται και στα σενάρια του υβριδικού αλγορίθμου.

Σενάριο Εκτέλεσης 1: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Σύμφωνα με το χρονοδιάγραμμα του σεναρίου του υβριδικού αλγορίθμου αρχικά παρουσιάζονται σφάλματα σε συνδέσεις στα κυρίως μονοπάτια των κυρίως ροών και μετέπειτα στα εναλλακτικά μονοπάτια. Στην περίπτωση του αλγορίθμου TSS με πολλαπλά μονοπάτια τα σφάλματα που εμφανίζονται στο δίκτυο επηρεάζουν μόνο αν επηρεαστεί κάποιος από τους κόμβους που αποτελούν τις n ανεξάρτητες διαδρομές.

Παρατηρήσεις: Τα σφάλματα συμβαίνουν στα μονοπάτια κάθε ροής. Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγορίθμου. Τα πακέτα που φτάνουν στον

egress κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

Σενάριο Εκτέλεσης 2: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Σύμφωνα με το χρονοδιάγραμμα του σεναρίου του υβριδικού αλγόριθμου αρχικά παρουσιάζονται σε συνδέσεις στα εναλλακτικά μονοπάτια των κυρίως ροών και ακολούθως στα κυρίως μονοπάτια. Στην περίπτωση του αλγορίθμου TSS με πολλαπλά μονοπάτια αλγορίθμου τα σφάλματα που εμφανίζονται στο δίκτυο επηρεάζουν μόνο αν επηρεαστεί κάποιος από τους κόμβους που αποτελούν τις n ανεξάρτητες διαδρομές.

Παρατηρήσεις: Τα σφάλματα συμβαίνουν στα μονοπάτια κάθε ροής. Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγόριθμου. Τα πακέτα που φτάνουν στον egress κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

Σενάριο Εκτέλεσης 3: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Σύμφωνα με το χρονοδιάγραμμα του σεναρίου του υβριδικού αλγόριθμου αρχικά παρουσιάζεται σφάλμα σε συνδέσεις στα κυρίως μονοπάτια των ροών. Μετέπειτα παρουσιάζεται σφάλμα σε κόμβο που επηρεάζει όλες τις ροές. Στην περίπτωση του αλγορίθμου TSS (k, n) με πολλαπλά μονοπάτια τα σφάλματα που εμφανίζονται στο δίκτυο επηρεάζουν μόνο αν επηρεαστεί κάποιος από τους κόμβους που αποτελούν τις k ανεξάρτητες διαδρομές.

Παρατηρήσεις: Τα σφάλματα συμβαίνουν στα μονοπάτια κάθε ροής. Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγόριθμου. Τα πακέτα που φτάνουν στον egress κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

Σενάριο Εκτέλεσης 4: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Παρουσιάζονται τυχαία σφάλματα σε κάθε ροή.

Παρατηρήσεις: Τα σφάλματα συμβαίνουν στα μονοπάτια κάθε ροής. Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγόριθμου. Τα πακέτα που φτάνουν στον egress κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

Για τα επόμενα σενάρια, θα μελετήσουμε πως λειτουργά ο αλγόριθμος TSS με πολλαπλά μονοπάτια μέσα από μια τοπολογία, η οποία περιέχει περισσότερους κόμβους (συνολικά περιέχει πενήντα κόμβους) να αποτελούν τις διαδρομές και μερικές από αυτές να δημιουργούνται από κόμβους disjoint (η τιμή του κατωφλίου θα μεταβάλλεται για να κατανοήσουμε καλύτερα πως το κατώφλι επηρεάζεται από τα πολλαπλά σφάλματα στις ανεξάρτητες ροές δρομολόγησης).

Σκοπός είναι η κατανόηση της λειτουργίας του αλγορίθμου TSS (k, n) με πολλαπλά μονοπάτια και κατά πόσο η μη δυναμική εύρεση εναλλακτικών μονοπατιών αποστολής πακέτων στην ύπαρξη σφαλμάτων, ο προκαθορισμός δρομολόγησης από πριν και η ύπαρξη του threshold επηρεάζουν την αποδοτικότητα του.

Σενάριο Εκτέλεσης 5: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Παρουσιάζονται τυχαία σφάλματα σε κόμβους από όλο το δίκτυο (εκτός από τους κόμβους που βρίσκονται αμέσως μετά τον αρχικό κόμβο και αμέσως προηγούμενως τον τελικό κόμβο).

Παρατηρήσεις: Τα σφάλματα συμβαίνουν στα μονοπάτια κάθε ροής. Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγόριθμου. Τα πακέτα που φτάνουν στον egress κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

Σενάριο Εκτέλεσης 6: Με την είσοδο των ροών στο δίκτυο, υπολογίζεται ο μέγιστος αριθμός n ανεξάρτητων διαδρομών για τις ροές. Παρουσιάζονται σφάλματα κυρίως σε αρχικούς κόμβους ή/και σε κόμβους disjoint.

Παρατηρήσεις: Τα σφάλματα που επηρεάζουν μονοπάτια που δημιουργούνται από κόμβους disjoint επηρεάζουν σε σημαντικό βαθμό την ικανοποίηση της συνθήκης του

κατωφλίου, κυρίως όταν το κατώφλι είναι μεγαλύτερο, αφού για κάθε disjoint κόμβο αφαιρούνται δυο πακέτα που θα χρησιμοποιούνταν για την ανακατασκευή του αρχικού πακέτου (λόγω του ότι αυτός ο κόμβος δημιουργό δυο μονοπάτια τα οποία ταυτόχρονα είναι και ανεξάρτητες διαδρομές). Τα σφάλματα δεν επιδιορθώνονται λόγω της φύσης του αλγόριθμου. Τα πακέτα που φτάνουν στον τελικό κόμβο για την ανακατασκευή του αρχικού πακέτου και δεν πληρούν το κριτήριο του κατωφλίου σε χρόνο ίσο με το μέγιστο χρόνο της μεγαλύτερης διαδρομής, απορρίπτονται.

7.3 Παράμετροι Αξιολόγησης

Για την αξιολόγηση του αλγορίθμου TSS (k,n) με πολλαπλά μονοπάτια καθορίστηκαν τα πιο κάτω κριτήρια

- ▲ **Πακέτα που παραλαμβάνονται:** Γραφική παράσταση με άξονα των x να είναι τα σενάρια και των y το ποσοστό των πακέτων που παραλαμβάνονται.
- ▲ **Πακέτα που χάνονται:** Γραφική παράσταση με άξονα των x να είναι οι πηγές κάθε σεναρίου και y το ποσοστό των πακέτων που χάνονται.
- ▲ **Πακέτα που λαμβάνονται με λανθασμένη σειρά (Received-Unordered-Lost):** Γραφική παράσταση με άξονα των x να είναι οι πηγές και y το ποσοστό των πακέτων που λαμβάνονται με λανθασμένη σειρά.
- ▲ **Χρόνος παραλαβής πακέτων (Delay):** Γραφική παράσταση με άξονα των x να είναι η ακολουθία των πακέτων και ο άξονας y να είναι ο χρόνος.
- ▲ **Δεδομένα που μεταφέρονται το δευτερόλεπτο της χρήσης των συνδέσεων (Throughput):** Γραφική παράσταση με άξονα των x το χρόνο και y τα Mb (Mega bit).
- ▲ **Χρόνος αποκατάστασης του κάθε σφάλματος (Recovery Time):** Γραφική παράσταση με άξονα των x να είναι οι πηγές για κάθε τοπολογία με τα λάθη τους και y να είναι ο χρόνος.

Τα πιο πάνω κριτήρια είναι τα ίδια με αυτά που αξιολογήθηκε ο υβριδικός αλγόριθμος σε προηγούμενη ερευνητική εργασία, έτσι ώστε να μπορέσουμε να συγκρίνουμε τους δύο αλγορίθμους.

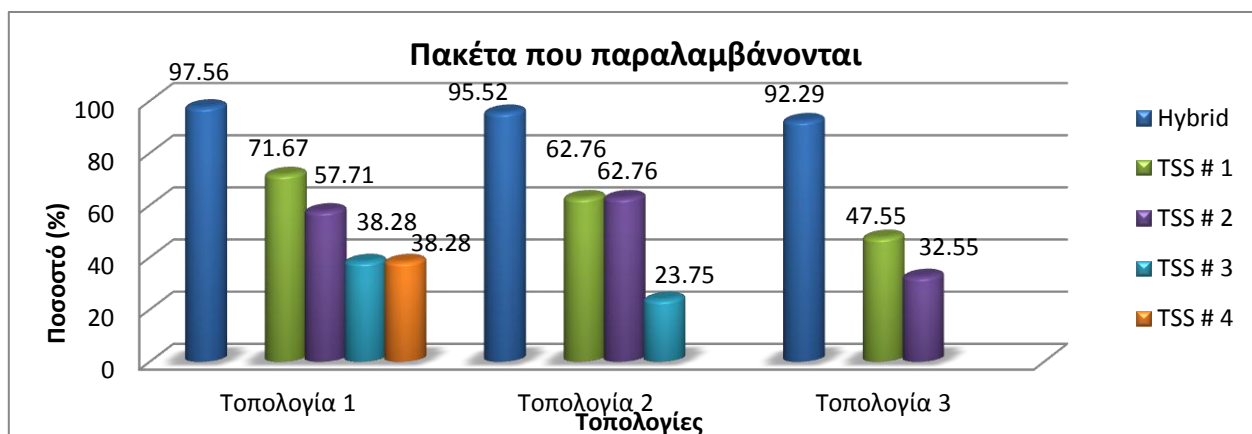
7.4 Αποτελέσματα Προσομοίωσης

Τα αποτελέσματα της προσομοίωσης του αλγορίθμου που συλλέξαμε παρουσιάζονται μέσω γραφικών παραστάσεων. Μέσω των γραφικών παραστάσεων θα παρουσιάσουμε τα αποτελέσματα από τα σενάρια υλοποίησης τόσο του αλγορίθμου TSS (k,n) με πολλαπλά μονοπάτια όσο και του υβριδικού (Hybrid Algorithm). Για κάθε σενάριο πάρθηκαν ξεχωριστές μετρήσεις έτσι θα υπάρχουν οι αντίστοιχες γραφικές παραστάσεις, όπου μέσα από αυτές θα μπορούμε να συγκρίνουμε τους δυο αλγόριθμους καθώς και να εξακριβώσουμε κάτω από ποιες συνθήκες ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια λειτουργεί αποδοτικά και πως συνδυάζει τη μηχανική κυκλοφορία με την ανοχή σφαλμάτων.

Οι γραφικές παραστάσεις παρουσιάζουν τα αποτελέσματα για τον μέσο όρο των ροών.

7.4.1 Γραφικές Παραστάσεις για το Σενάριο 1

Πακέτα που παραλαμβάνονται

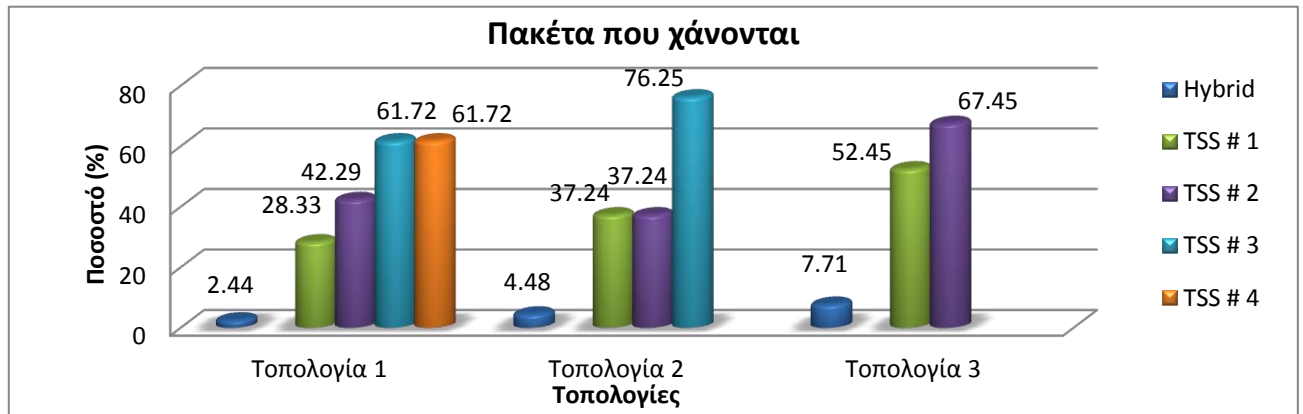


Σχήμα 7.4.1.1 – Ποσοστό πακέτων που παραλήφθηκαν

Από την πάνω γραφική παράσταση παρατηρούμε ότι το ποσοστό των πακέτων που παραλαμβάνονται διατηρούνται σε ψηλά επίπεδα για τον υβριδικό αλγόριθμο. Ελαττώνεται για τα δίκτυα που έχουν τις λιγότερες συνδέσεις όταν παρουσιαστεί σφάλμα (υπάρχει δυσκολία στην εύρεση εναλλακτικών ροών δρομολόγησης με αποτέλεσμα να αυξάνεται ο χρόνος επαναφοράς και να χάνονται περισσότερα πακέτα). Στην περίπτωση του TSS αλγορίθμου το ποσοστό των πακέτων που παραλαμβάνονται είναι πολύ μικρότερος από αυτό του υβριδικού αλγορίθμου λόγω του ότι όταν προκύψει

σφάλμα στο δίκτυο δεν βρίσκει εναλλακτικές ροές δρομολόγησης με αποτέλεσμα να μην παραλαμβάνονται τα υπό-πακέτα για την ικανοποίηση του κατωφλίου έτσι ώστε να δημιουργηθεί το αρχικό πακέτο.

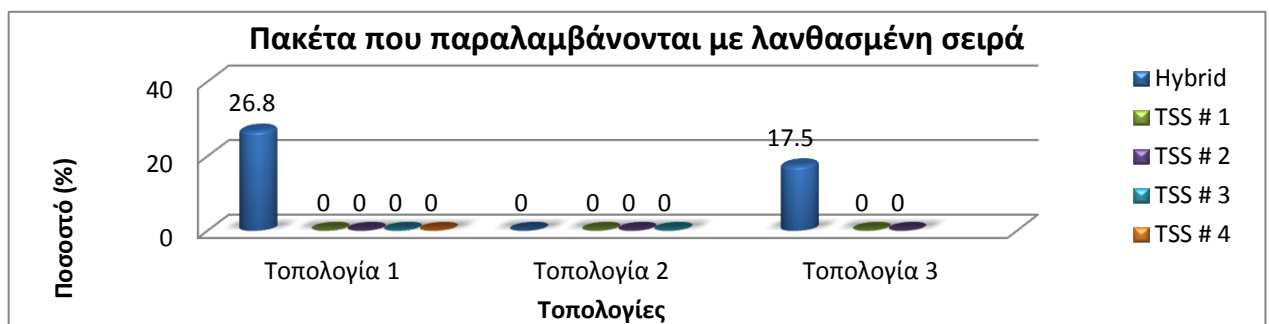
Πακέτα που χάνονται



Σχήμα 7.4.1.2 – Ποσοστό πακέτων που χάθηκαν

Από την πιο πάνω γραφική παράσταση παρατηρούμε ότι για τον υβριδικό αλγόριθμο το ποσοστό των πακέτων που χάνονται διατηρούνται σε χαμηλά επίπεδα. Λίγο περισσότερα είναι για το δίκτυο που έχει τις λιγότερες συνδέσεις, λόγω του ότι υπάρχει δυσκολία στην εύρεση νέων ροών δρομολόγησης. Ενώ στον αλγόριθμο TSS το ποσοστό των πακέτων που χάνονται είναι μεγαλύτερο δια το λόγω ότι βάση της δομής του αλγορίθμου στην περίπτωση σφάλματος σε ροή στην οποία γίνεται δρομολόγηση πακέτων δεν γίνεται εύρεση εναλλακτικής ροής για αυτήν ως αποτέλεσμα όταν δεν πληρείται το απαιτούμενο κατώφλι για την δημιουργία του αρχικού πακέτου, το πακέτο αυτό να χάνεται.

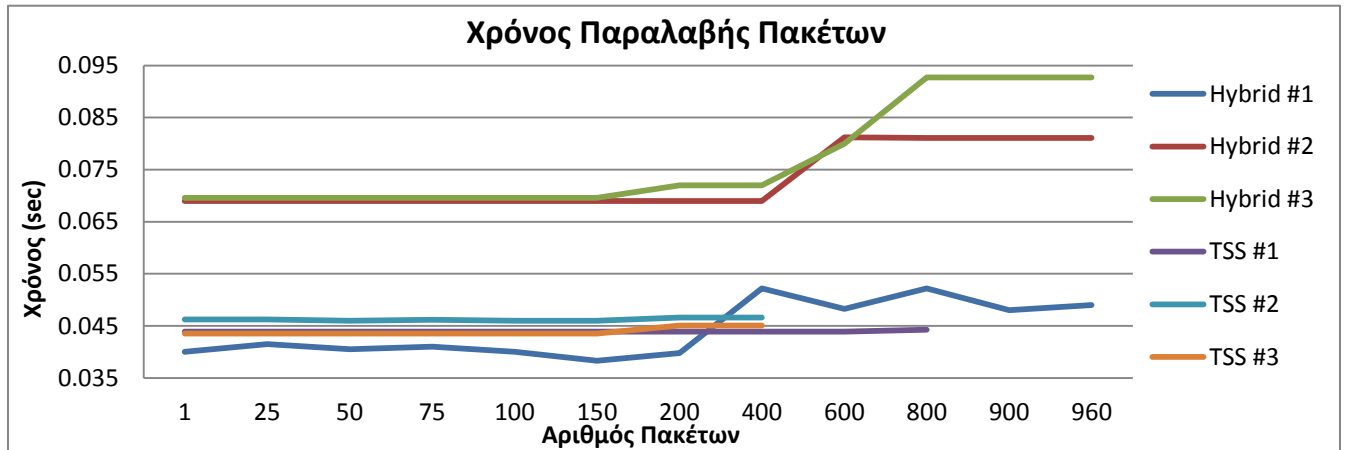
Πακέτα που λαμβάνονται με λανθασμένη σειρά



Σχήμα 7.4.1.3 – Ποσοστό πακέτων που παραλήφθηκαν με λανθασμένη σειρά

Ο υβριδικός αλγόριθμος παρουσιάζει χαμηλό ποσοστό πακέτων τα οποία λαμβάνονται με λανθασμένη σειρά ενώ ο αλγόριθμος TSS έχει μηδενικά ποσοστά για όλες τις τοπολογίες.

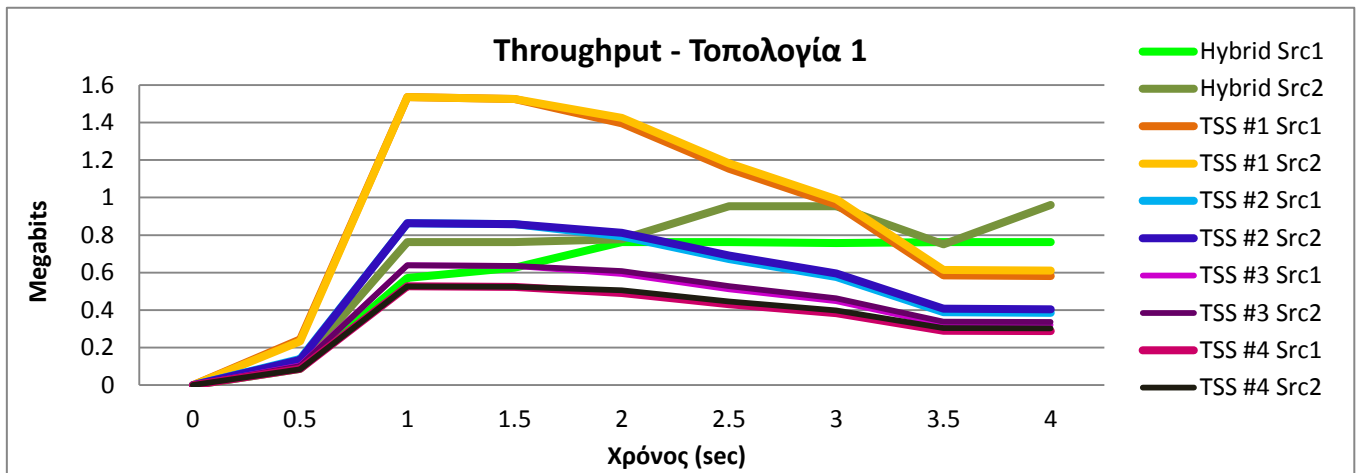
Χρόνος παραλαβής πακέτων (Delay)



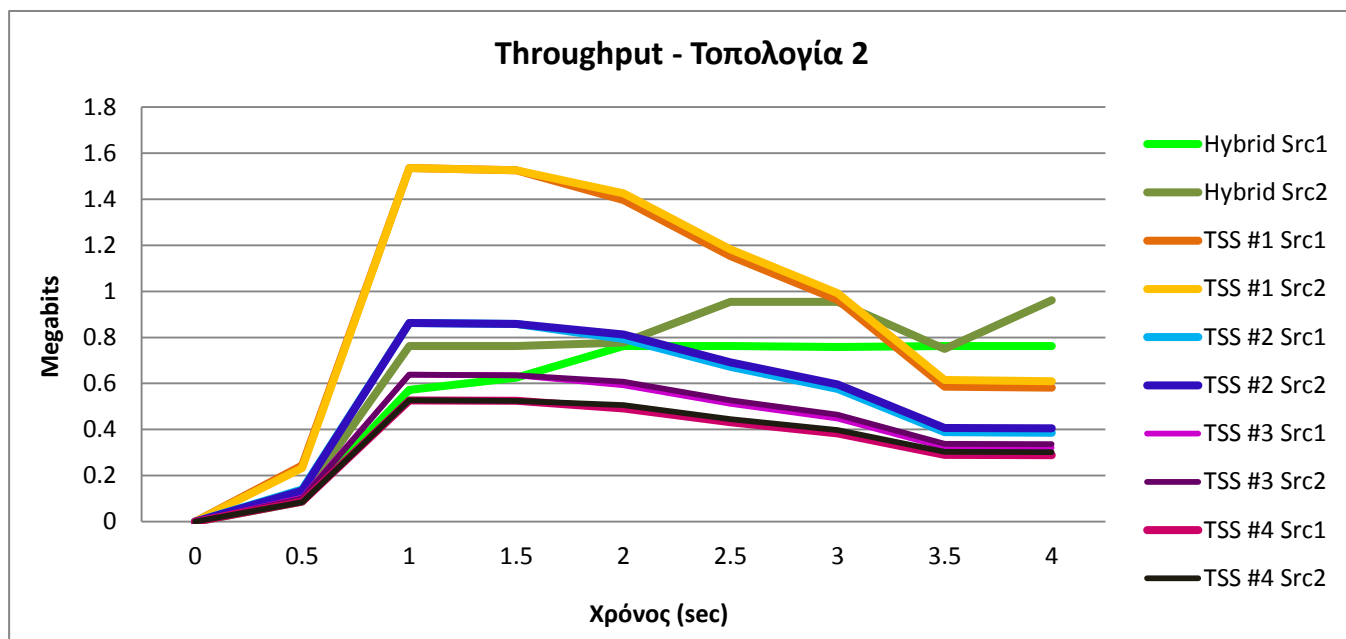
Σχήμα 7.4.1.4 – Χρόνος παραλαβής των πακέτων

Παρατηρούμε ότι ο χρόνος παραμένει σταθερός για τις τοπολογίες ένα και δύο, ενώ για την τοπολογία τρία αυξάνεται λόγω της χρήσης των ροών στα εναλλακτικά μονοπάτια που υπολογίζονται από τον υβριδικό αλγόριθμο. Για τις περιπτώσεις του TSS αλγορίθμου ο χρόνος παραμένει σταθερός μέχρι τη στιγμή που τα σφάλματα που παρουσιάζονται επηρεάζουν όλες τις ροές δρομολόγησης και πλέον δεν μπορεί να παραλάβει άλλα υπό-πακέτα για τη δημιουργία του αρχικού πακέτου.

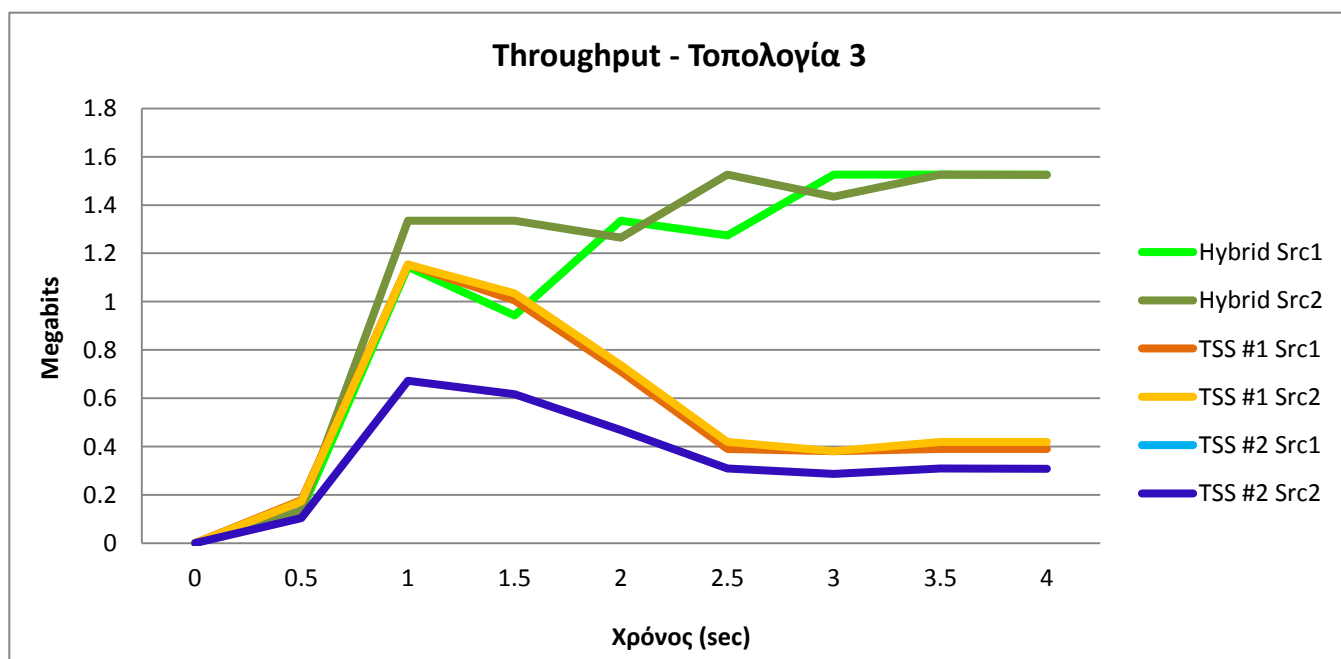
Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.1.5.1 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 1



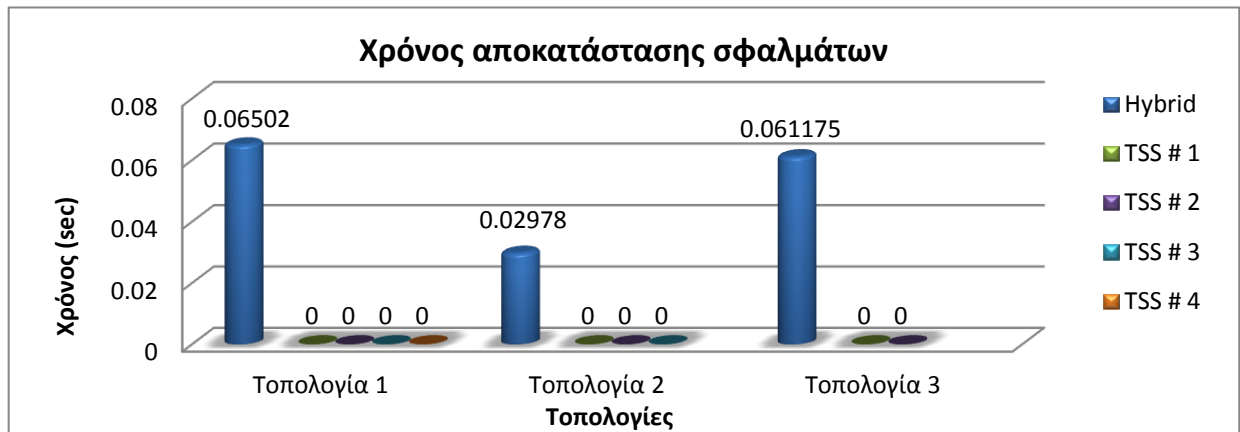
Σχήμα 7.4.1.5.2 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 2



Σχήμα 7.4.1.5.3 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 3

Από τις πιο πάνω γραφικές παραστάσεις παρατηρούμε ότι τα δεδομένα που μεταφέρονται ανά δευτερόλεπτο για κάθε ροή διατηρούνται σε ψηλά επίπεδα (max utilization). Για τον υβριδικό αλγόριθμο πέφτει ελάχιστα για κάθε ροή μόνο όταν παρουσιαστεί σφάλμα, μέχρι να γίνει η επαναφορά. Ενώ για τον αλγόριθμο TSS όταν παρουσιαστεί κάποιο σφάλμα ισούται με το μέγιστο αριθμό των δεδομένων που μεταφέρονται στις υπόλοιπες ροές αφού δεν γίνεται επαναφορά ροών διαδρομής.

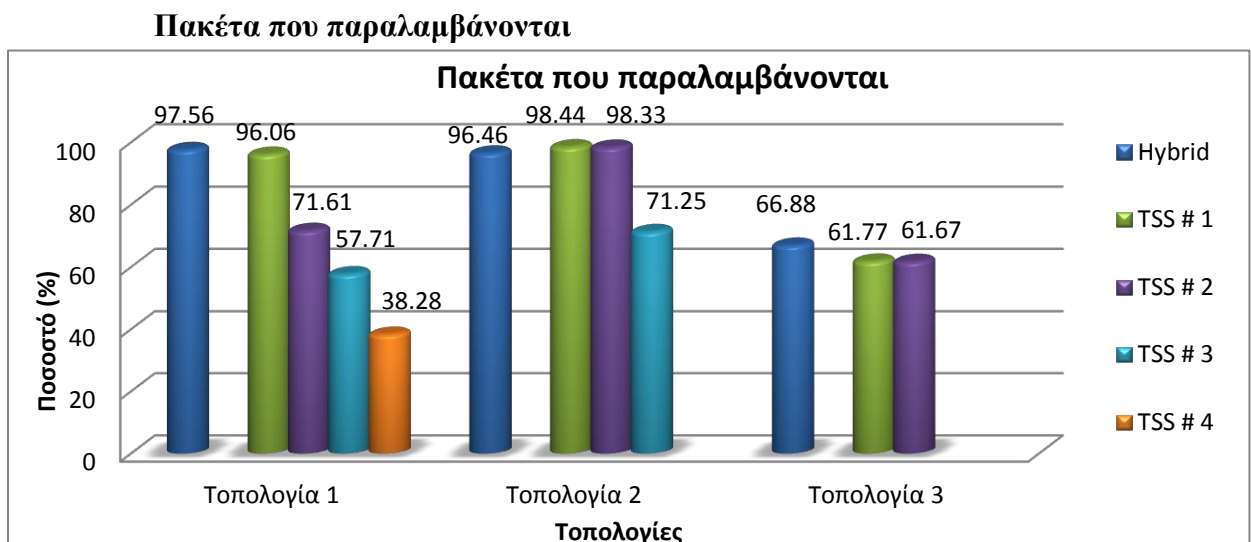
Χρόνος αποκατάστασης του κάθε σφάλματος (Recovery Time)



Σχήμα 7.4.1.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

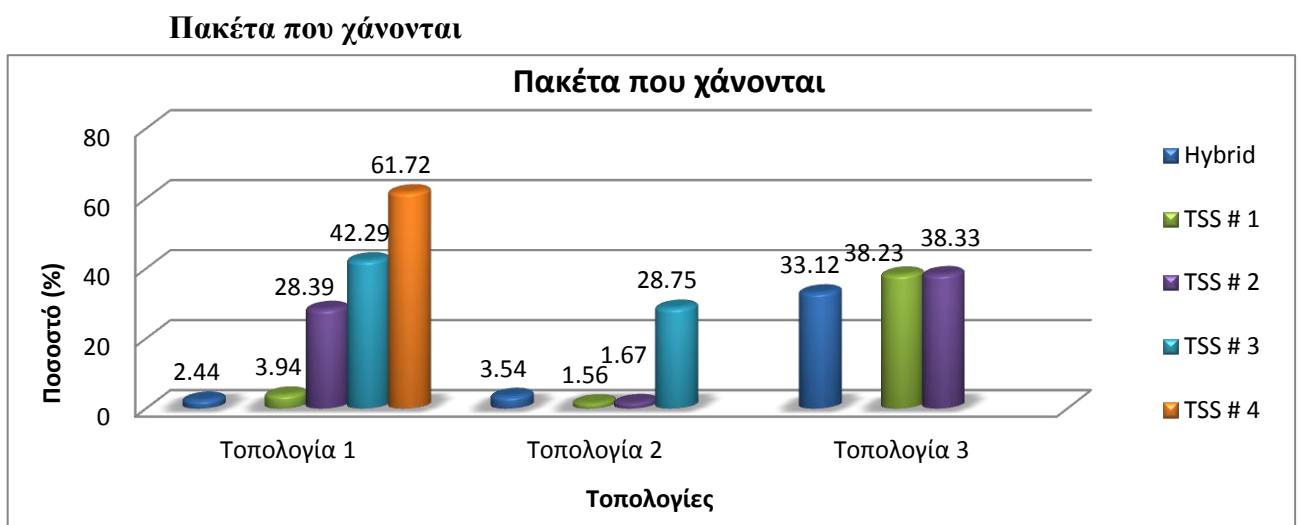
Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης. Ενώ για τον υβριδικό αλγόριθμο βρίσκεται σε χαμηλά επίπεδα. Τις μεγαλύτερες τιμές τις έχουν οι τοπολογίες με τους λιγότερους κόμβους, στις οποίες ο υβριδικός αλγόριθμος δυσκολεύεται να βρει εναλλακτικές ροές δρομολόγησης δεδομένων που να πληρούν τα κριτήρια που ορίζει.

7.4.2 Γραφικές Παραστάσεις για το Σενάριο 2



Σχήμα 7.4.2.1 – Ποσοστό πακέτων που παραλήφθηκαν

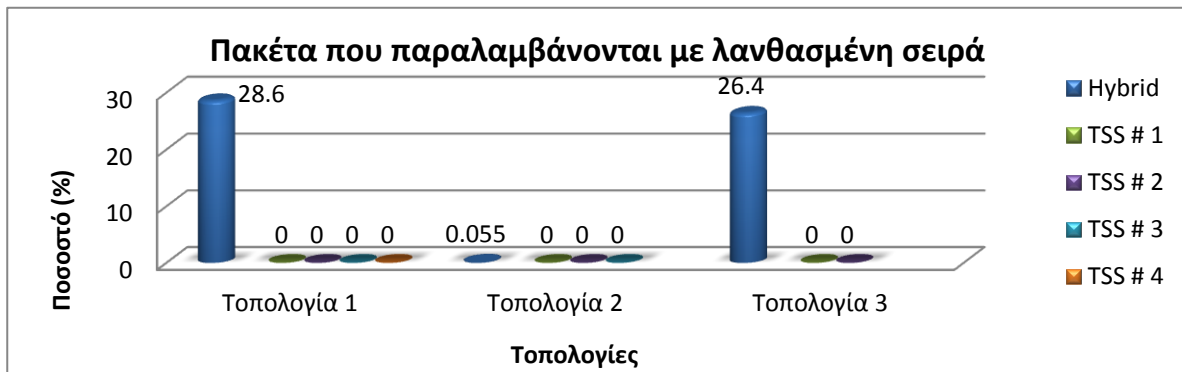
Από την πάνω γραφική παράσταση παρατηρούμε ότι το ποσοστό των πακέτων που παραλαμβάνονται διατηρούνται σε ψηλά επίπεδα για τον υβριδικό αλγόριθμο. Ελαττώνεται για την τοπολογία 3 όταν παρουσιαστεί σφάλμα (υπάρχει δυσκολία στην εύρεση εναλλακτικών ροών δρομολόγησης με αποτέλεσμα να αυξάνεται ο χρόνος επαναφοράς και να χάνονται περισσότερα πακέτα). Στην περίπτωση του TSS αλγορίθμου το ποσοστό των πακέτων που παραλαμβάνονται είναι μεγάλο στις περιπτώσεις που το κατώφλι είναι ίσο ή μικρότερο του μισού από τον αριθμό των ανεξάρτητων ροών δρομολόγησης αφού τα σφάλματα συμβαίνουν σε αυτές τις ροές δρομολόγησης.



Σχήμα 7.4.2.2 – Ποσοστό πακέτων που χάθηκαν

Από την πάνω γραφική παράσταση παρατηρούμε ότι για τον υβριδικό αλγόριθμο το ποσοστό των πακέτων που χάνονται διατηρούνται σε χαμηλά επίπεδα και για τις τρεις τοπολογίες. Για τον αλγόριθμο TSS το ποσοστό των πακέτων που χάνονται είναι μεγαλύτερο στις περιπτώσεις που το κατώφλι είναι μεγαλύτερο από το μισό αριθμό των ανεξάρτητων ροών δρομολόγησης γιατί όταν τα σφάλματα εμφανίζονται στις ροές αυτές δεν γίνεται επαναδρομολόγηση με αποτέλεσμα τα πακέτα αυτά να χάνονται. Στην τοπολογία τρία που έχουμε τις λιγότερες συνδέσεις τα σφάλματα προκύπτουν προς το τέλος της προσομοίωσης γι' αυτό και το ποσοστό των πακέτων που χάθηκαν δεν είναι αρκετά μεγάλο.

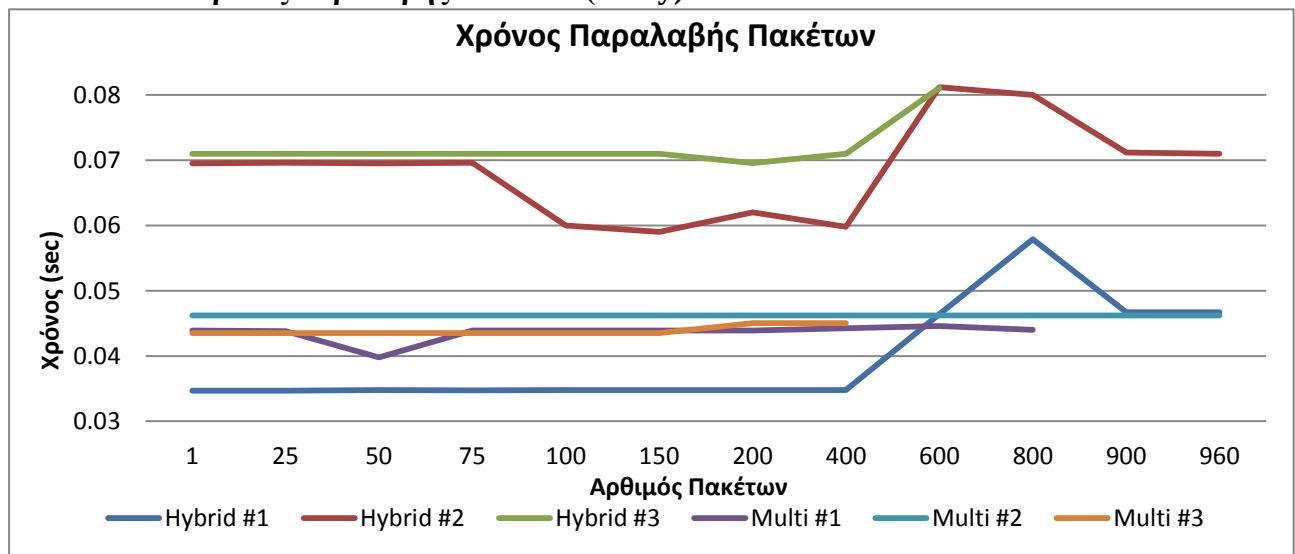
Πακέτα που λαμβάνονται με λανθασμένη σειρά



Σχήμα 7.4.2.3 – Ποσοστό πακέτων που παραλήφθηκαν με λανθασμένη σειρά

Ο υβριδικός αλγόριθμος παρουσιάζει χαμηλό ποσοστό πακέτων τα οποία λαμβάνονται με λανθασμένη σειρά ενώ ο αλγόριθμος TSS έχει μηδενικά ποσοστά για όλες τις τοπολογίες.

Χρόνος παραλαβής πακέτων (Delay)

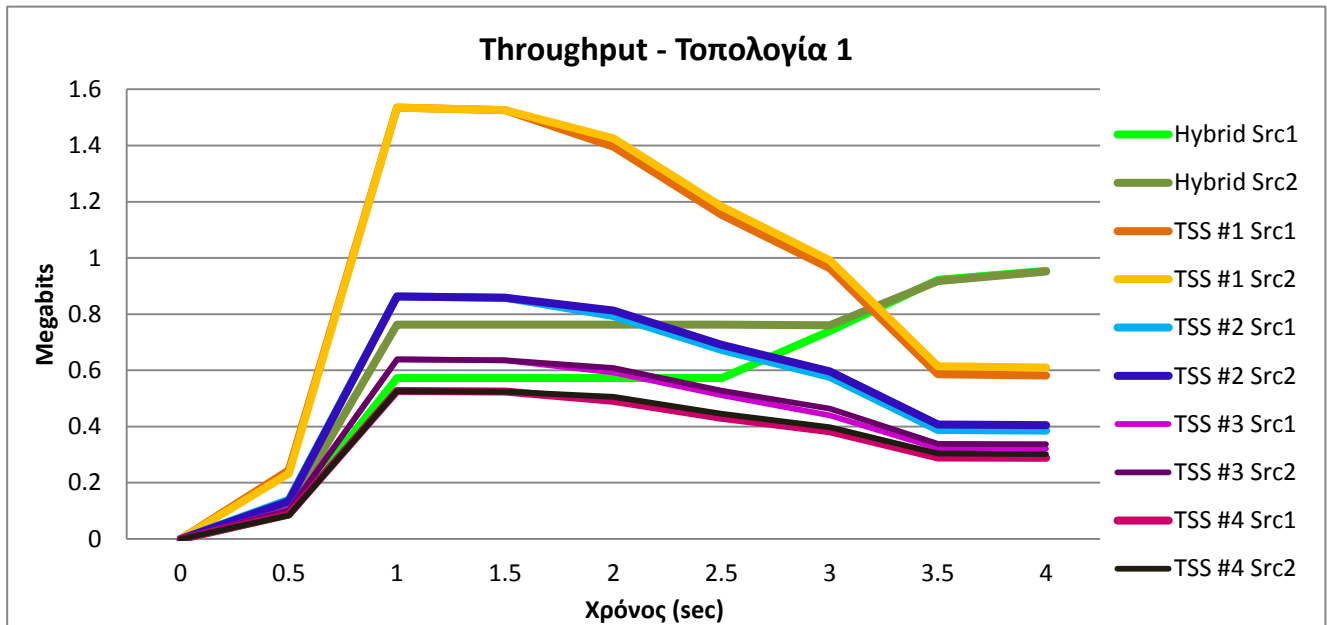


Σχήμα 7.4.2.4 – Χρόνος παραλαβής των πακέτων

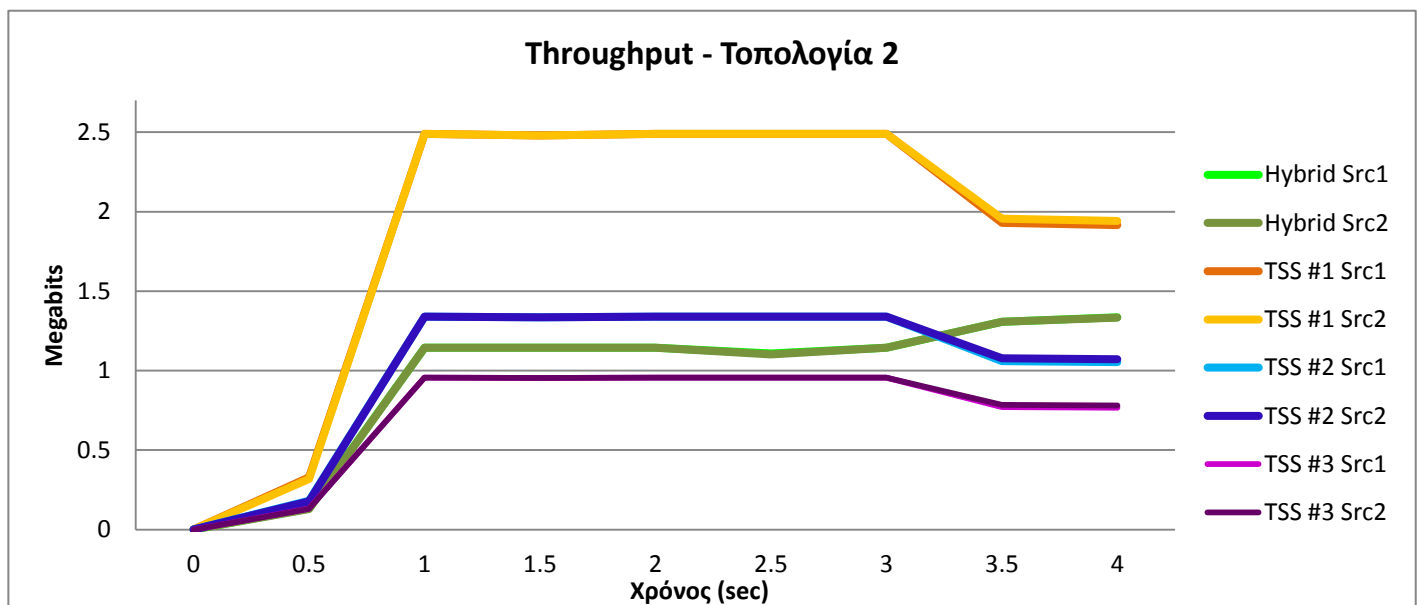
Ο χρόνος παραλαβής των πακέτων παραμένει σταθερός για τις τοπολογίες του υβριδικού αλγόριθμου. Προς το τέλος της προσομοίωσης παρατηρείται μικρή αύξηση γιατί λόγω των σφαλμάτων που γίνονται στο δίκτυο ο αλγόριθμος δυσκολεύεται να βρει εναλλακτικά μονοπάτια ροής δρομολόγησης. Για τις περιπτώσεις του TSS αλγορίθμου ο χρόνος παραμένει σταθερός για τις τρεις τοπολογίες για όλη τη χρονική διάρκεια

αποστολής των πακέτων. Στο τέλος των σεναρίων λόγω σφαλμάτων στις κύριες ροές δρομολόγησης δεν παραλαμβάνονται άλλα πακέτα.

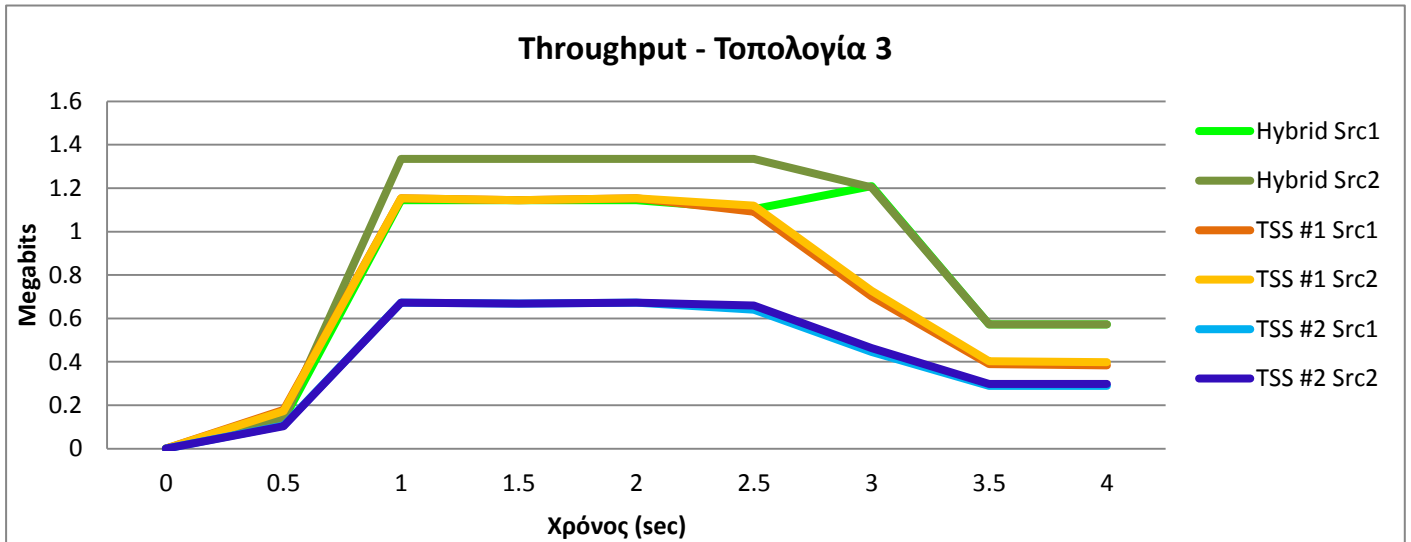
Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.2.5.1 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 1



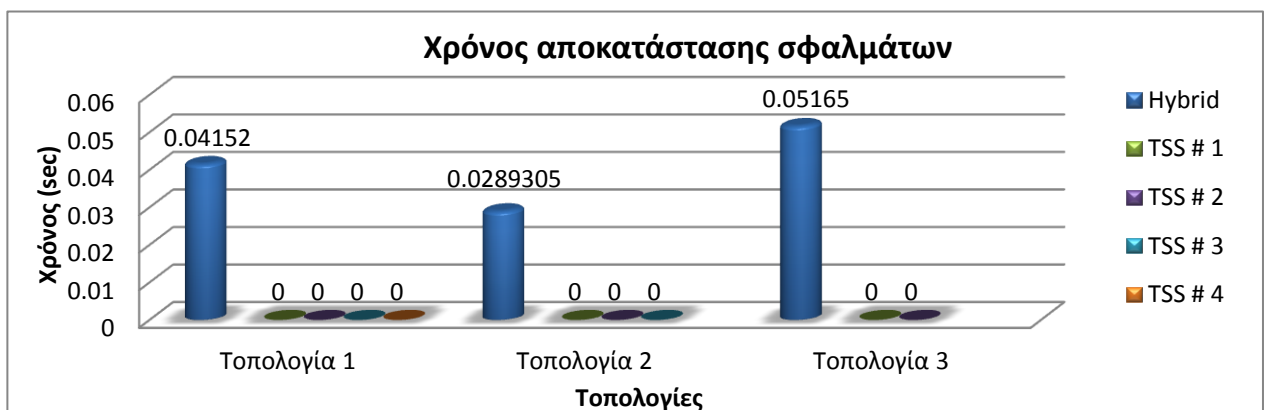
Σχήμα 7.4.2.5.2 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 2



Σχήμα 7.4.2.5.3 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 3

Από τις πιο πάνω γραφικές παραστάσεις παρατηρούμε ότι τα δεδομένα που μεταφέρονται ανά δευτερόλεπτο για κάθε ροή διατηρούνται σε ψηλά επίπεδα (max utilization). Για τον υβριδικό αλγόριθμο πέφτει ελάχιστα για κάθε ροή μόνο όταν παρουσιαστεί σφάλμα, μέχρι να γίνει η επαναφορά. Ενώ για τον αλγόριθμο TSS όταν παρουσιαστεί κάποιο σφάλμα ισούται με το μέγιστο αριθμό των δεδομένων που μεταφέρονται στις υπόλοιπες ροές αφού δεν γίνεται επαναφορά ροών διαδρομής. Στις πρώτες δυο τοπολογίες ο αλγόριθμος TSS για μικρό κατώφλι έχει πολύ μεγαλύτερο όγκο δεδομένων που μεταφέρονται στις ροές από τον υβριδικό αλγόριθμο, ενώ στην τοπολογία τρία ο υβριδικός αλγόριθμος έχει το μεγαλύτερο όγκο δεδομένων διότι όταν προκύψουν τα σφάλματα στις ροές δρομολόγησης ο TSS αλγόριθμος δεν βρίσκει εναλλακτικά μονοπάτια δρομολόγησης των πακέτων.

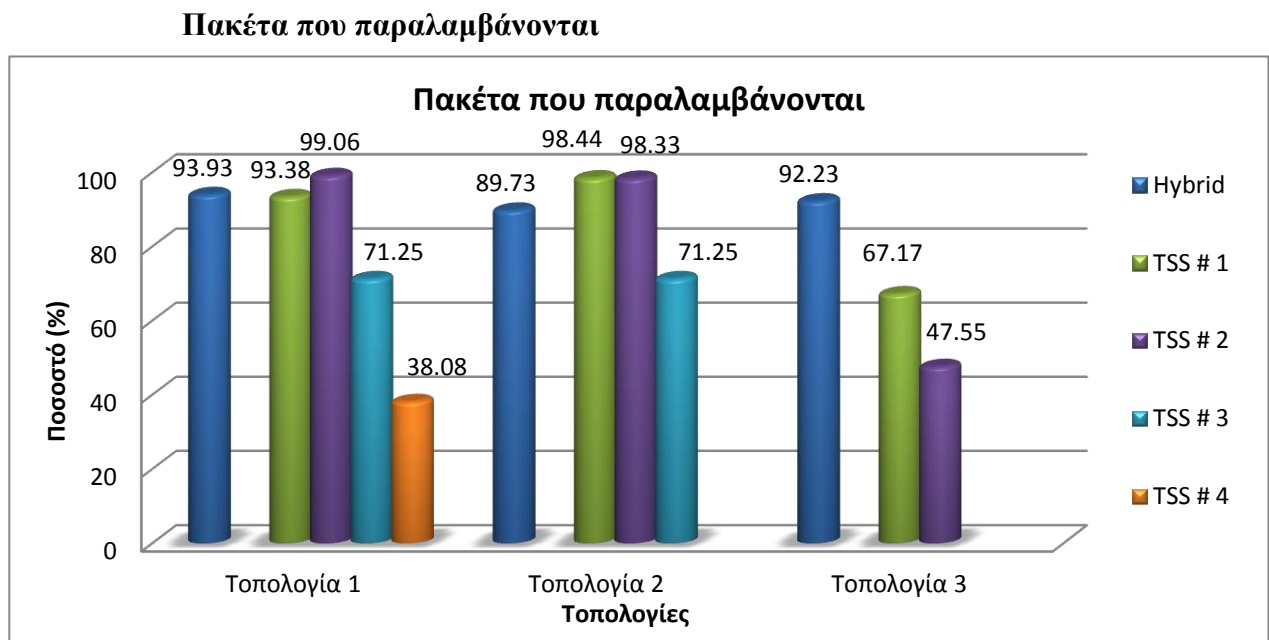
Χρόνος αποκατάστασης του κάθε σφάλματος (Recovery Time)



Σχήμα 7.4.2.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης. Ενώ για τον υβριδικό αλγόριθμο βρίσκεται σε χαμηλά επίπεδα. Τις μεγαλύτερες τιμές τις έχουν οι τοπολογίες με τους λιγότερους κόμβους, στις οποίες ο υβριδικός αλγόριθμος δυσκολεύεται να βρει εναλλακτικές ροές δρομολόγησης δεδομένων που να πληρούν τα κριτήρια που ορίζει.

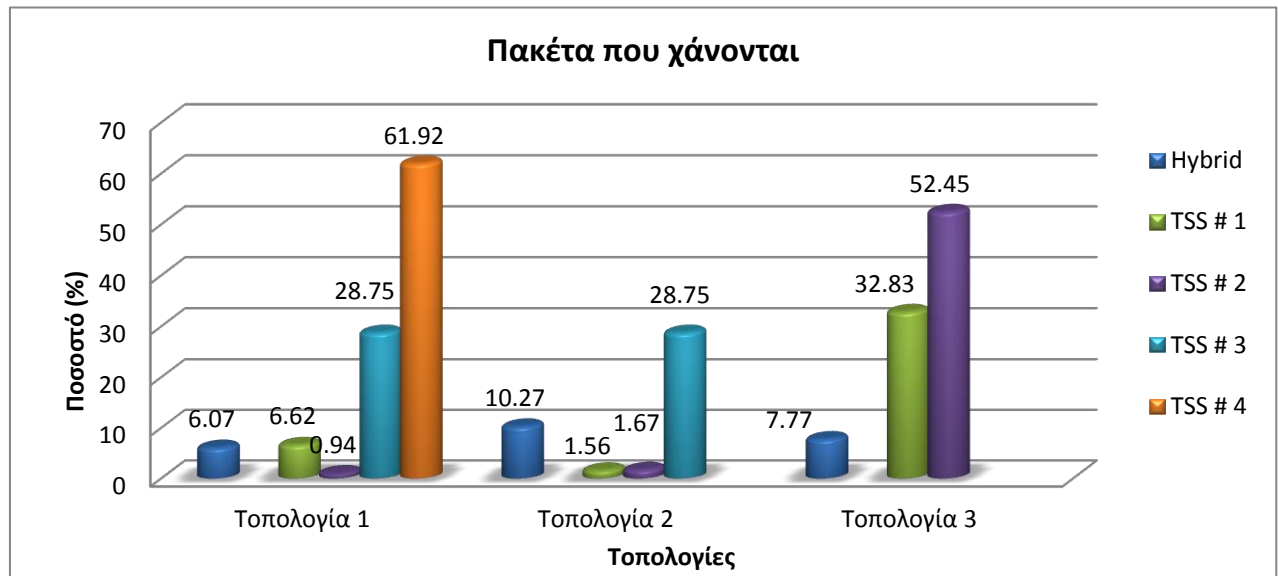
7.4.3 Γραφικές Παραστάσεις για το Σενάριο 3



Σχήμα 7.4.3.1 – Ποσοστό πακέτων που παραλήφθηκαν

Από την πάνω γραφική παράσταση παρατηρούμε ότι το ποσοστό των πακέτων που παραλαμβάνονται διατηρούνται σε ψηλά επίπεδα για τον υβριδικό και τον TSS αλγόριθμο. Ελαττώνεται το ποσοστό του TSS αλγορίθμου στην τοπολογία ένα και τρία όταν το κατώφλι ισούται με τον αριθμό των n ανεξάρτητων ροών δρομολόγησης πακέτων.

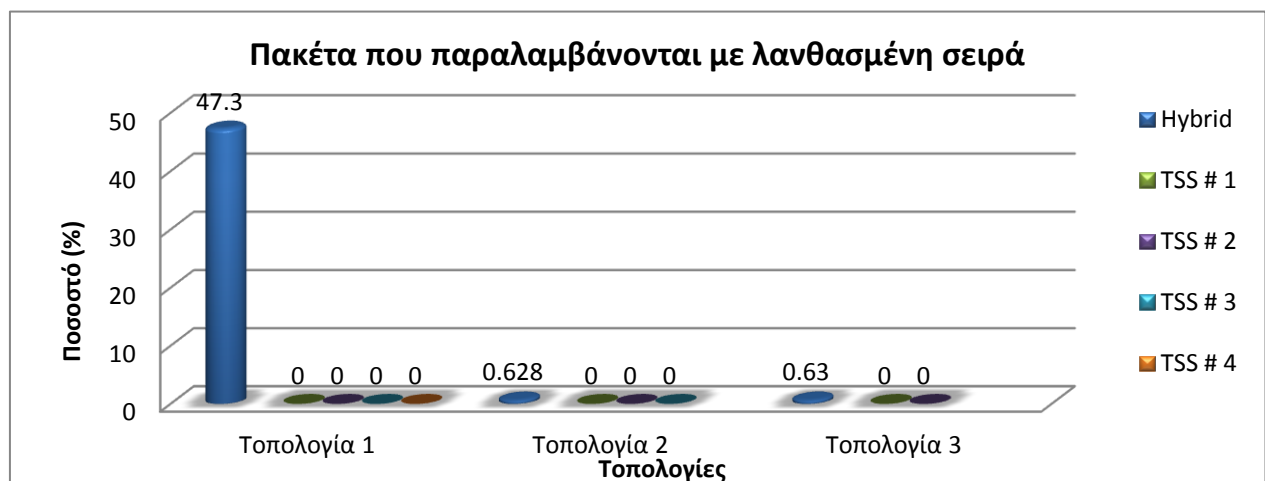
Πακέτα που χάνονται



Σχήμα 7.4.3.2 – Ποσοστό πακέτων που χάθηκαν

Από την πάνω γραφική παράσταση παρατηρούμε ότι και για τους δύο αλγόριθμους το ποσοστό των πακέτων που χάνονται διατηρούνται σε χαμηλά επίπεδα και για τις τρεις τοπολογίες. Το ποσοστό των πακέτων που χάθηκαν για τον TSS αλγόριθμο αυξάνεται στην τοπολογία ένα και τρία όταν το κατώφλι ισούται με τον αριθμό των n ανεξάρτητων ροών δρομολόγησης πακέτων. Φτάνει μέχρι και το 61,92% γιατί λόγω το σφαλμάτων που συμβαίνουν στις ροές δρομολόγησης τα υπό-πακέτα που χρειάζονται για την δημιουργία του αρχικού πακέτου δεν φτάνουν στον τελικό κόμβο.

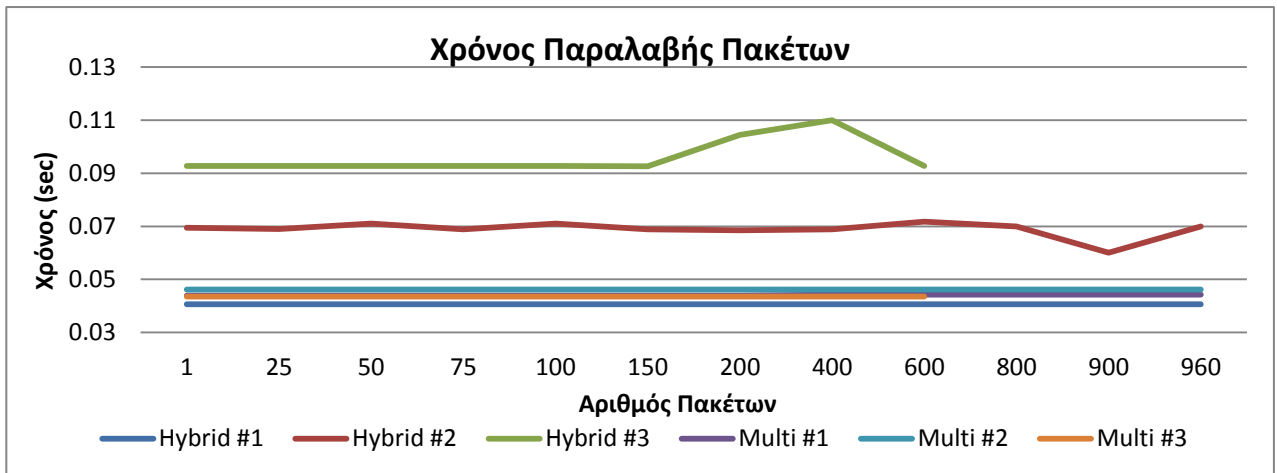
Πακέτα που λαμβάνονται με λανθασμένη σειρά



Σχήμα 7.4.3.3 – Ποσοστό πακέτων που παραλήφθηκαν με λανθασμένη σειρά

Ο υβριδικός αλγόριθμος παρουσιάζει χαμηλό ποσοστό πακέτων τα οποία λαμβάνονται με λανθασμένη σειρά για τις τοπολογίες δύο & τρία και πολύ ψηλό για την τοπολογία ένα, ενώ ο αλγόριθμος TSS έχει μηδενικά ποσοστά για όλες τις τοπολογίες.

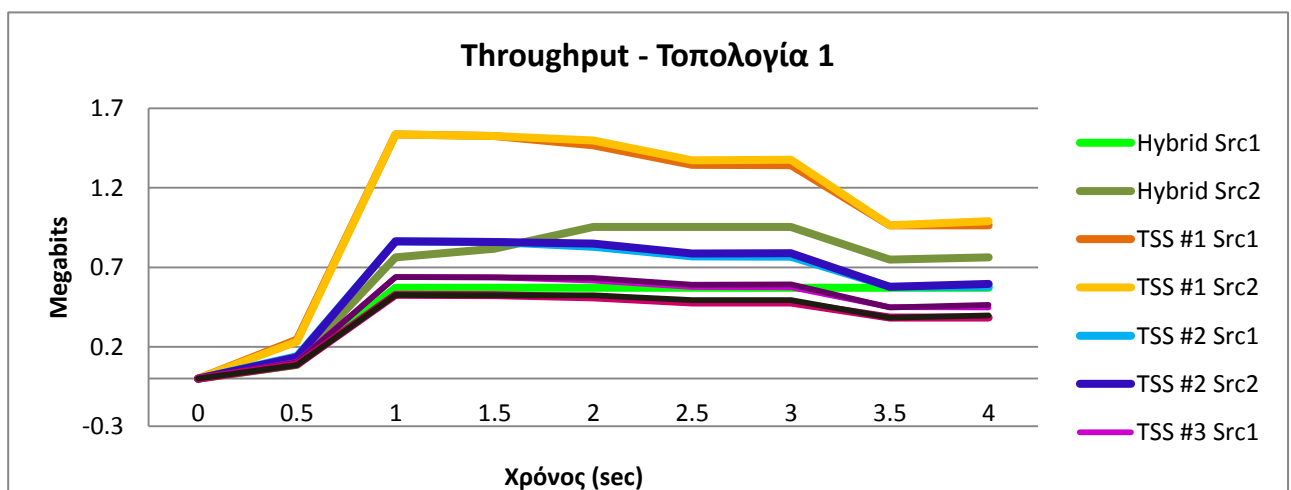
Χρόνος παραλαβής πακέτων (Delay)



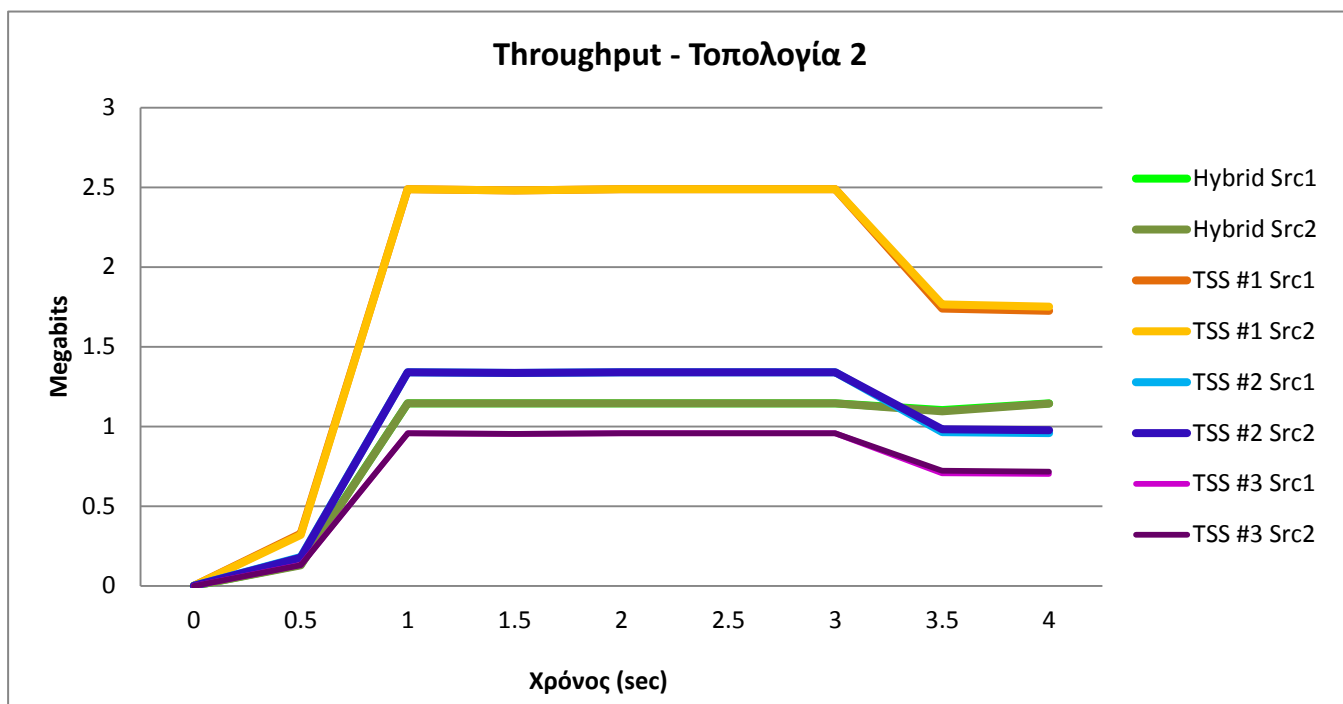
Σχήμα 7.4.3.4 – Χρόνος παραλαβής των πακέτων

Ο χρόνος παραλαβής των πακέτων παραμένει σταθερός για όλες τις τοπολογίες του TSS αλγόριθμου. Για τις τοπολογίες δύο και τρία του υβριδικού αλγορίθμου ο χρόνος παραλαβής των πακέτων παρουσιάζει μικρή αυξομείωση γιατί παρουσιάζονται σφάλματα στις κυρίως ροές δρομολόγησης του.

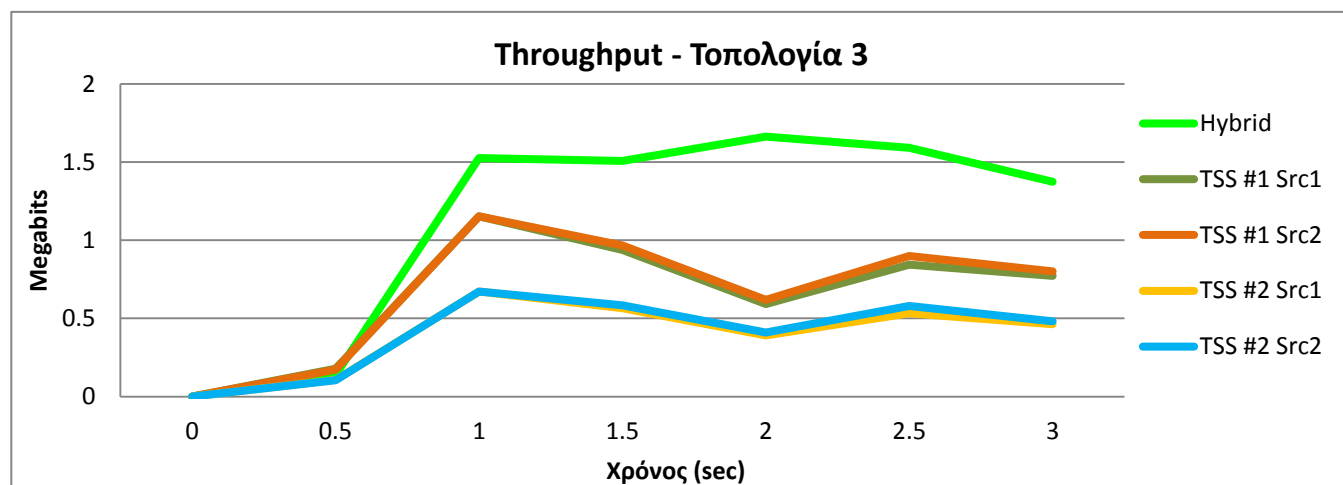
Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.3.5.1 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 1



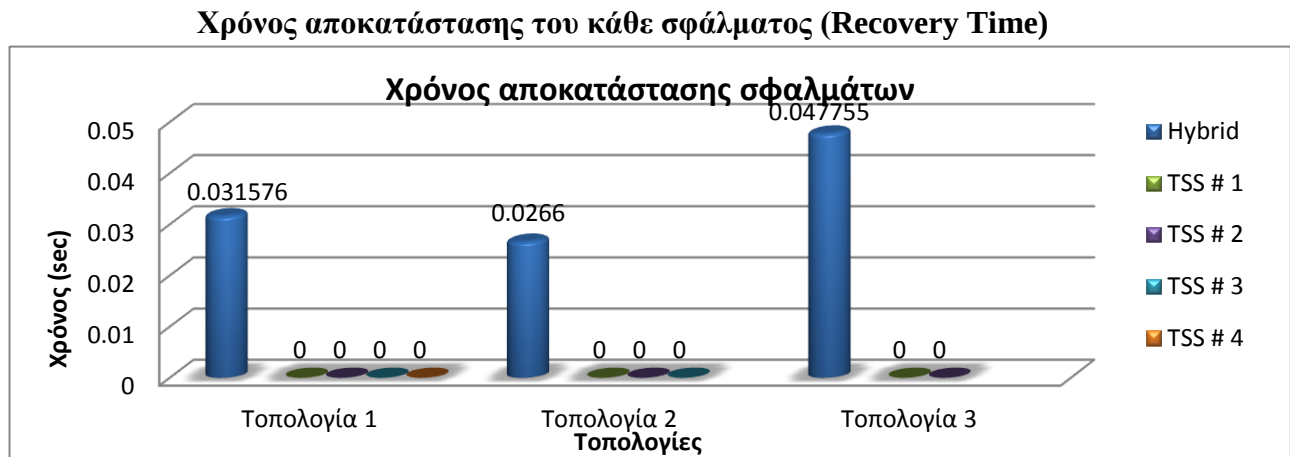
Σχήμα 7.4.3.5.2 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 2



Σχήμα 7.4.3.5.3 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 3

Από τις πιο πάνω γραφικές παραστάσεις παρατηρούμε ότι τα δεδομένα που μεταφέρονται ανά δευτερόλεπτο για κάθε ροή διατηρούνται σε ψηλά επίπεδα (max utilization). Για τον υβριδικό αλγόριθμο πέφτει ελάχιστα μόνο όταν παρουσιαστεί σφάλμα στη ροή δρομολόγησης, μέχρι να γίνει η επαναφορά. Ενώ για τον αλγόριθμο TSS όταν παρουσιαστεί κάποιο σφάλμα ισούται με το μέγιστο αριθμό των δεδομένων που μεταφέρονται στις υπόλοιπες ροές αφού δεν γίνεται επαναφορά ροών διαδρομής.

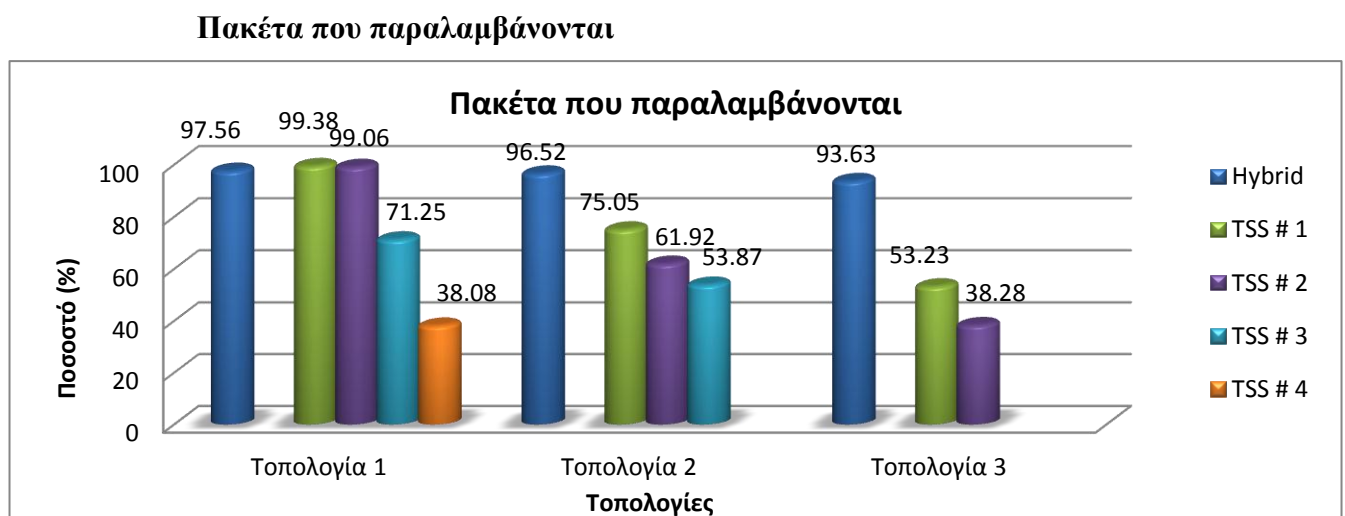
Στην τοπολογία τρία ο υβριδικός αλγόριθμος έχει μεγαλύτερο όγκο δεδομένων γιατί σε αντίθεση με τον TSS αλγόριθμο βρίσκει δυναμικά εναλλακτικό μονοπάτι για τη δρομολόγηση πακέτων. Στις δυο πρώτες τοπολογίες ο TSS έχει μεγαλύτερο όγκο δεδομένων που μεταφέρονται στις ροές για μικρό κατώφλι.



Σχήμα 7.4.3.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης. Ενώ για τον υβριδικό αλγόριθμο βρίσκεται σε χαμηλά επίπεδα. Τις μεγαλύτερες τιμές τις έχουν οι τοπολογίες με τους λιγότερους κόμβους, στις οποίες ο υβριδικός αλγόριθμος δυσκολεύεται να βρει εναλλακτικές ροές δρομολόγησης δεδομένων που να πληρούν τα κριτήρια που ορίζει.

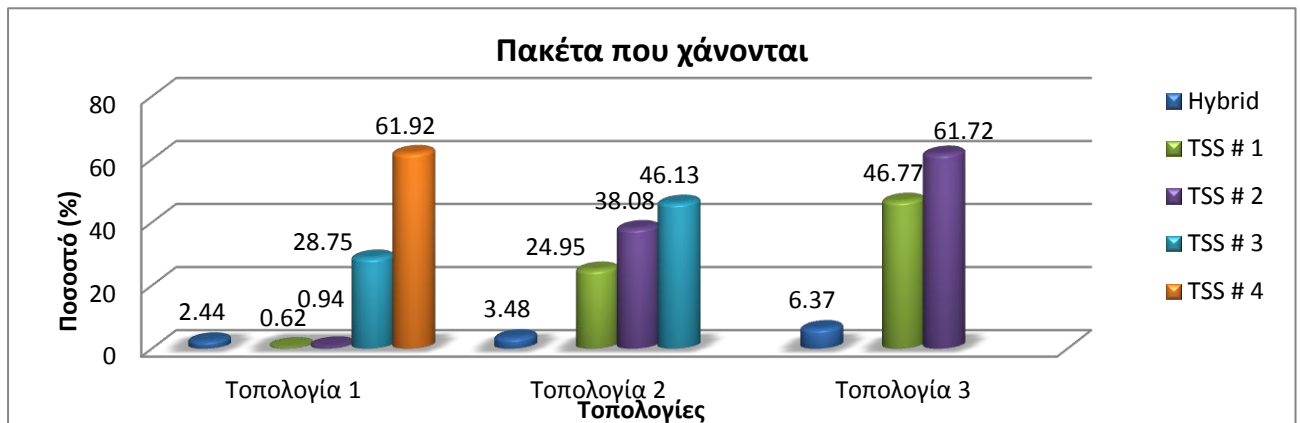
7.4.4 Γραφικές Παραστάσεις για το Σενάριο 4



Σχήμα 7.4.4.1 – Ποσοστό πακέτων που παραλήφθηκαν

Από την πάνω γραφική παράσταση παρατηρούμε ότι το ποσοστό των πακέτων που παραλαμβάνονται διατηρούνται σε ψηλά επίπεδα για τον υβριδικό και τον TSS αλγόριθμο. Ελαττώνεται το ποσοστό του TSS αλγορίθμου όταν το κατώφλι ισούται με τον αριθμό των n ανεξάρτητων ροών δρομολόγησης πακέτων.

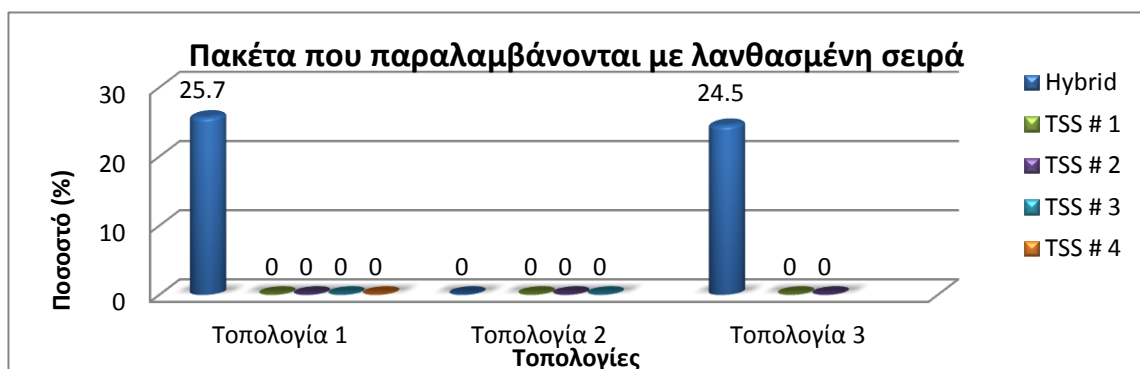
Πακέτα που χάνονται



Σχήμα 7.4.4.2 – Ποσοστό πακέτων που χάθηκαν

Από την γραφική παράσταση παρατηρούμε ότι το ποσοστό των πακέτων που χάνονται για τον υβριδικό αλγόριθμο βρίσκεται σε πολύ χαμηλά επίπεδα σε αντίθεση με το ποσοστό των πακέτων που χάθηκαν για τον TSS αλγόριθμο. Αυτό οφείλεται στο γεγονός ότι ο υβριδικός βρίσκει εναλλακτικές ροές δρομολόγησης όταν εμφανίζονται σφάλματα (τα σφάλματα στις ροές δρομολόγησης του TSS αλγορίθμου εμφανίζονται προς το τέλος της προσομοίωσης γι' αυτό και δεν έχει μεγαλύτερο ποσοστό αποτυχίας παραλαβής πακέτων).

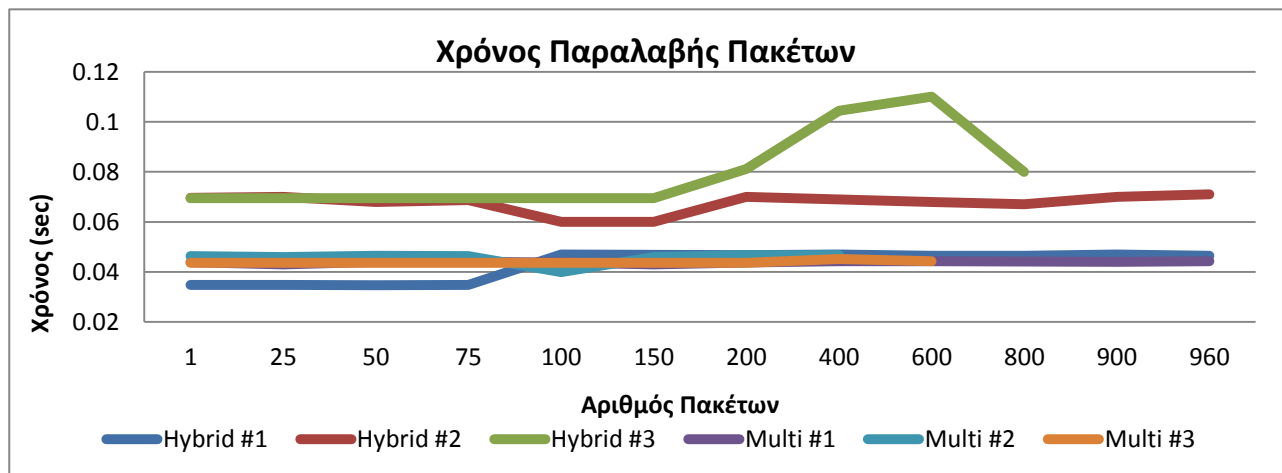
Πακέτα που λαμβάνονται με λανθασμένη σειρά



Σχήμα 7.4.4.3 – Ποσοστό πακέτων που παραλαμβάνονται με λανθασμένη σειρά

Ο υβριδικός αλγόριθμος παρουσιάζει χαμηλό ποσοστό πακέτων τα οποία λαμβάνονται με λανθασμένη σειρά στην τοπολογία δύο ενώ στις τοπολογίες ένα και τρία το 25% των συνολικών πακέτων που παραλαμβάνονται είναι με λανθασμένη σειρά. Ο αλγόριθμος TSS έχει μηδενικά ποσοστά για όλες τις τοπολογίες.

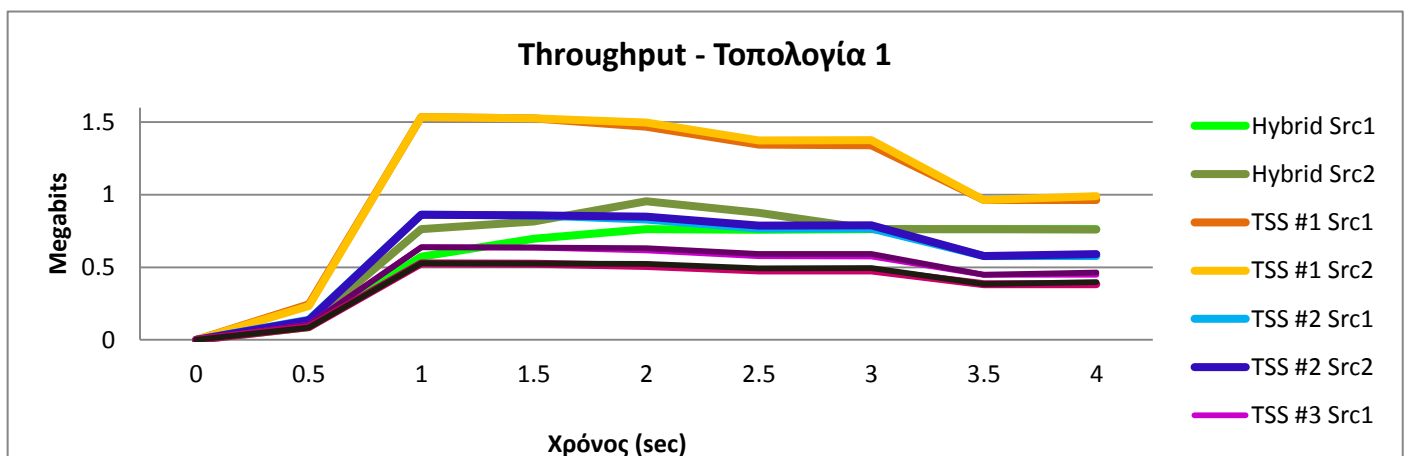
Χρόνος παραλαβής πακέτων (Delay)



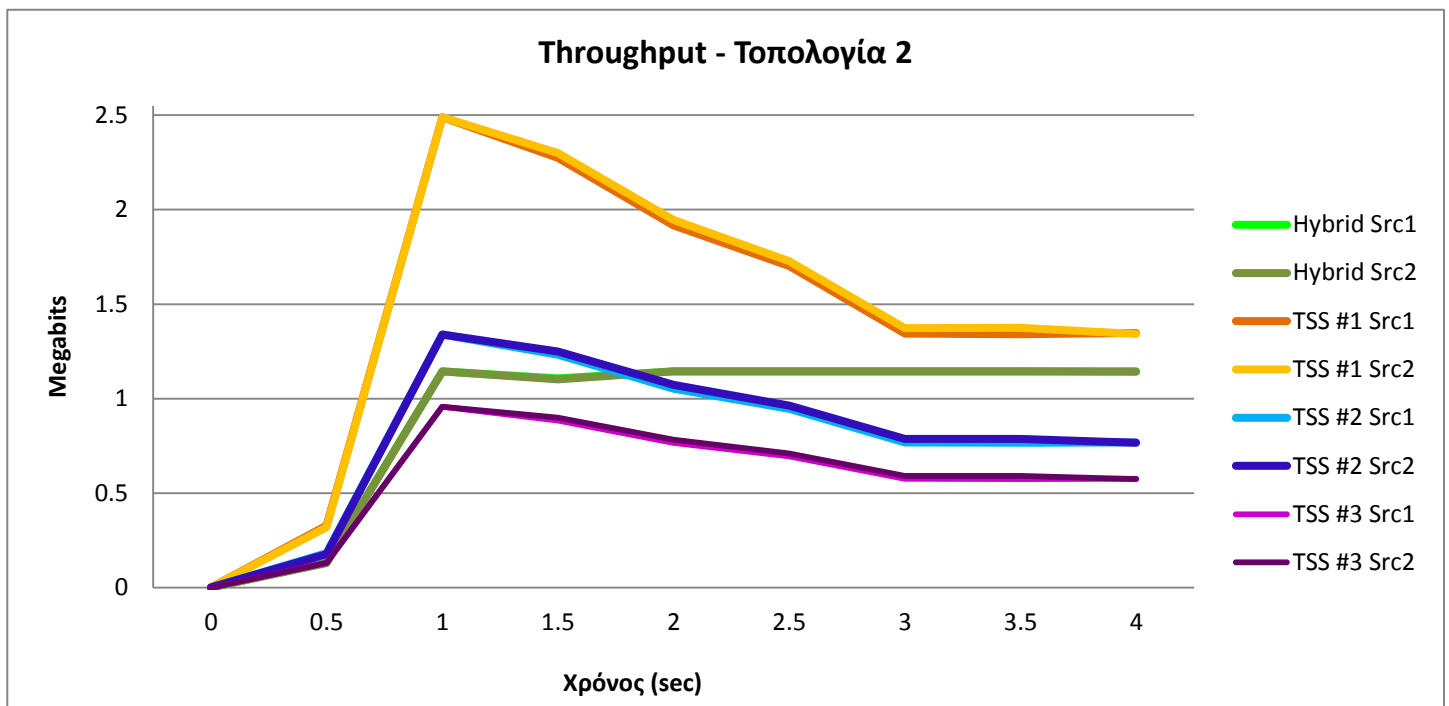
Σχήμα 7.4.4.4 – Χρόνος παραλαβής των πακέτων

Ο χρόνος παραλαβής των πακέτων παραμένει σταθερός για όλες τις τοπολογίες του TSS αλγόριθμου (τις στιγμές που εμφανίζονται τα σφάλματα δεν γίνεται παραλαβή άλλων πακέτων). Για τις τοπολογίες δύο και τρία του υβριδικού αλγορίθμου ο χρόνος παραλαβής των πακέτων παρουσιάζει μικρή αυξομείωση γιατί παρουσιάζονται σφάλματα στις κυρίως ροές δρομολόγησης του.

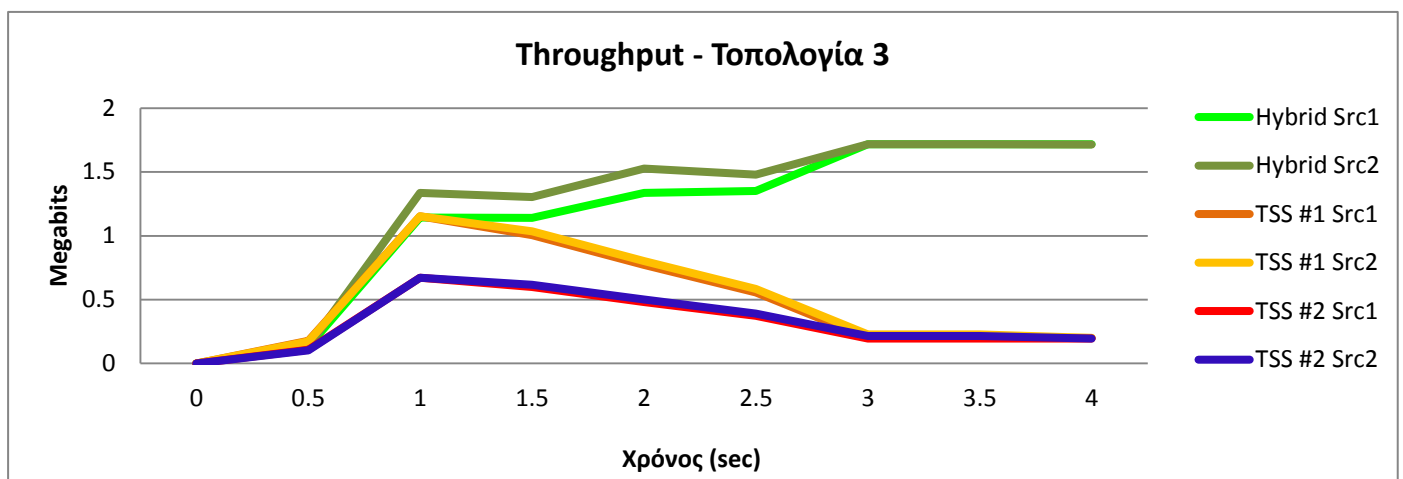
Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.4.5.1 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 1



Σχήμα 7.4.4.5.2 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 2

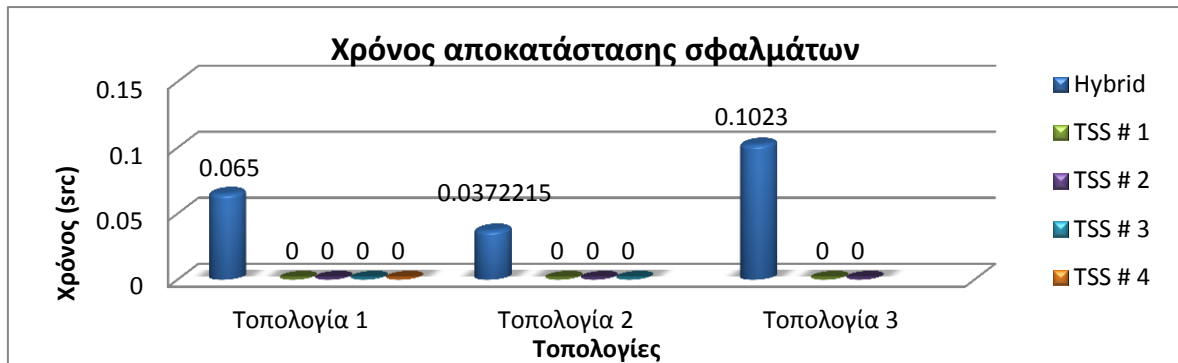


Σχήμα 7.4.4.5.3 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 3

Από τις πιο πάνω γραφικές παραστάσεις παρατηρούμε ότι τα δεδομένα που μεταφέρονται ανά δευτερόλεπτο για κάθε ροή διατηρούνται σε ψηλά επίπεδα (max utilization). Για τον υβριδικό αλγόριθμο πέφτει ελάχιστα μόνο όταν παρουσιαστεί σφάλμα στη ροή δρομολόγησης, μέχρι να γίνει η επαναφορά. Ενώ για τον αλγόριθμο TSS όταν παρουσιαστεί κάποιο σφάλμα ισούνται με το μέγιστο αριθμό των δεδομένων που μεταφέρονται στις υπόλοιπες ροές αφού δεν γίνεται επαναφορά ροών διαδρομής. Στην τοπολογία τρία ο υβριδικός αλγόριθμος έχει μεγαλύτερο όγκο δεδομένων γιατί σε

αντίθεση με τον TSS αλγόριθμο βρίσκει δυναμικά εναλλακτικό μονοπάτι για τη δρομολόγηση πακέτων. Στις δυο πρώτες τοπολογίες ο TSS έχει μεγαλύτερο όγκο δεδομένων που μεταφέρονται στις ροές για μικρό κατώφλι.

Χρόνος αποκατάστασης του κάθε σφάλματος (Recovery Time)

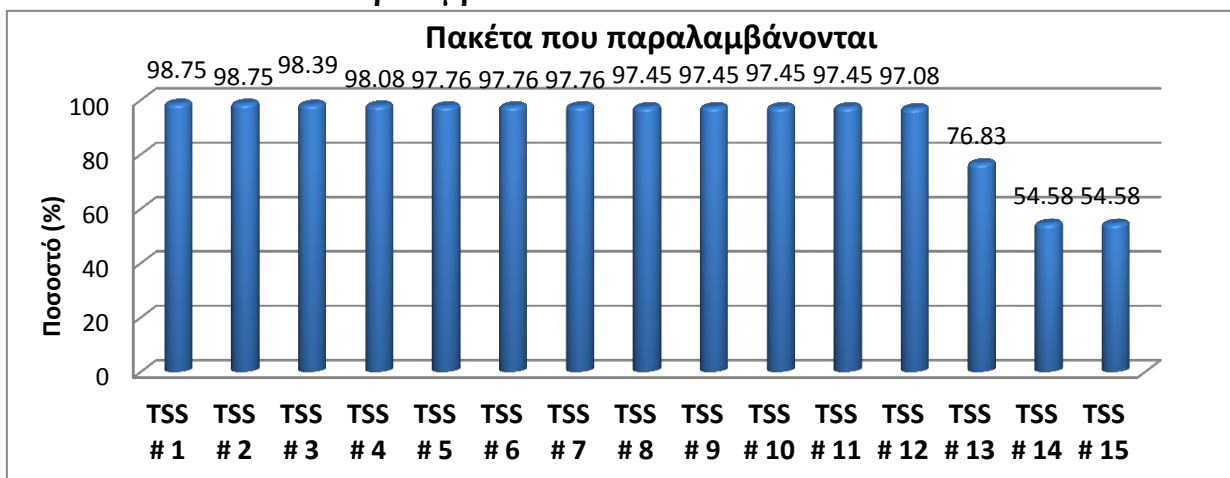


Σχήμα 7.4.4.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης. Ενώ για τον υβριδικό αλγόριθμο βρίσκεται σε χαμηλά επίπεδα. Τις μεγαλύτερες τιμές τις έχουν οι τοπολογίες με τους λιγότερους κόμβους, στις οποίες ο υβριδικός αλγόριθμος δυσκολεύεται να βρει εναλλακτικές ροές δρομολόγησης δεδομένων που να πληρούν τα κριτήρια που ορίζει.

7.4.2 Γραφικές Παραστάσεις για το Σενάριο 5

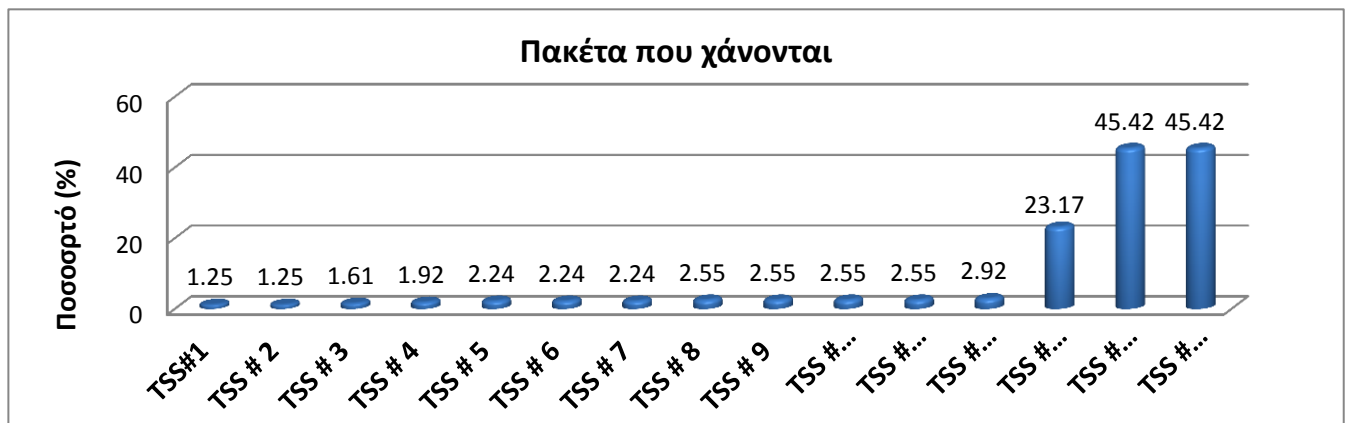
Πακέτα που παραλαμβάνονται



Σχήμα 7.4.5.1 – Ποσοστό πακέτων που παραλήφθηκαν

Το ποσοστό των πακέτων που παραλαμβάνονται στο σενάριο 5 της τοπολογίας 4 βρίσκεται σε πολύ ψηλά επίπεδα για τις πλείστες τιμές που παίρνει η τιμή του κατωφλίου. Αυτό οφείλεται στο γεγονός ότι τα σφάλματα που προκύπτουν στο δίκτυο επηρεάζουν μόνο δύο από τους κόμβους που βρίσκονται στις ροές δρομολόγησης των πακέτων και δεν είναι κόμβοι του τύπου disjoint. Το πιο μικρό ποσοστό το έχουν οι περιπτώσεις όπου η τιμή του κατωφλίου ισούται με 14 και 15, δηλαδή όσο είναι και ο αριθμός των n ανεξάρτητων ροών δρομολόγησης.

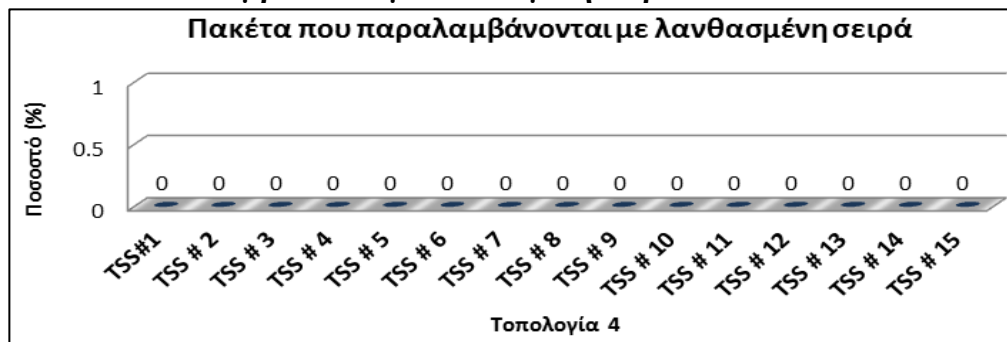
Πακέτα που χάνονται



Σχήμα 7.4.5.2 – Ποσοστό πακέτων που χάθηκαν

Από την πιο πάνω γραφική παράσταση το ποσοστό των πακέτων που χάνονται είναι πολύ χαμηλό αφού τα σφάλματα που προκύπτουν στο δίκτυο επηρεάζουν μόνο δύο από τους κόμβους που βρίσκονται στις ροές δρομολόγησης. Το ποσοστό όμως αυξάνεται για τις τιμές του κατωφλίου 14 & 15 γιατί τα υπό-πακέτα που απαιτούνται από τις ροές που έχουν σφάλματα δεν φτάνουν ποτέ.

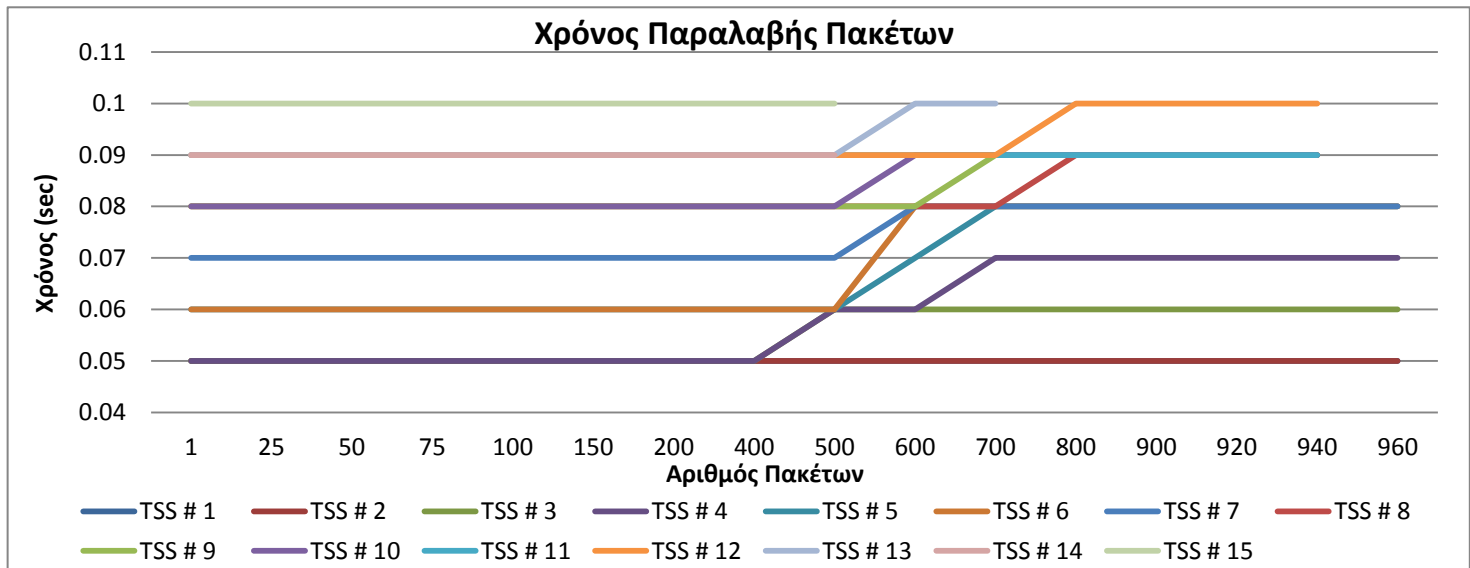
Πακέτα που λαμβάνονται με λανθασμένη σειρά



Σχήμα 7.4.5.3 – Ποσοστό πακέτων που παραλήφθηκαν με λανθασμένη σειρά

Για όλες τις διαφορετικές τιμές του κατωφλίου, ο αλγόριθμος TSS έχει μηδενικά ποσοστά για τα πακέτα που παραλαμβάνονται με λανθασμένη σειρά.

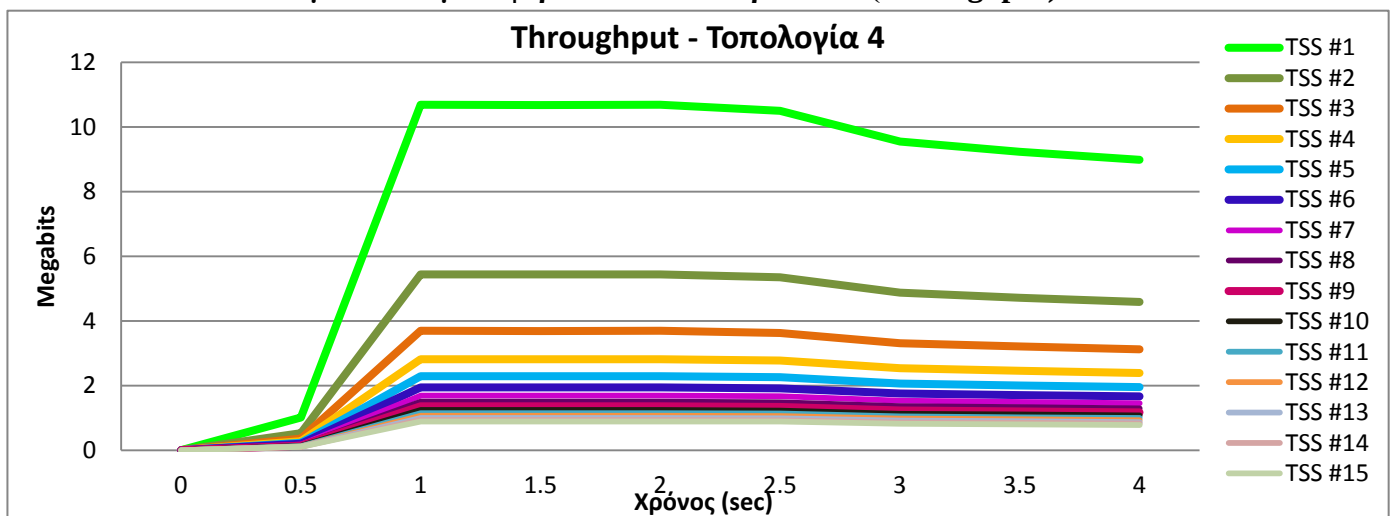
Χρόνος παραλαβής πακέτων (Delay)



Σχήμα 7.4.5.4 – Χρόνος παραλαβής των πακέτων

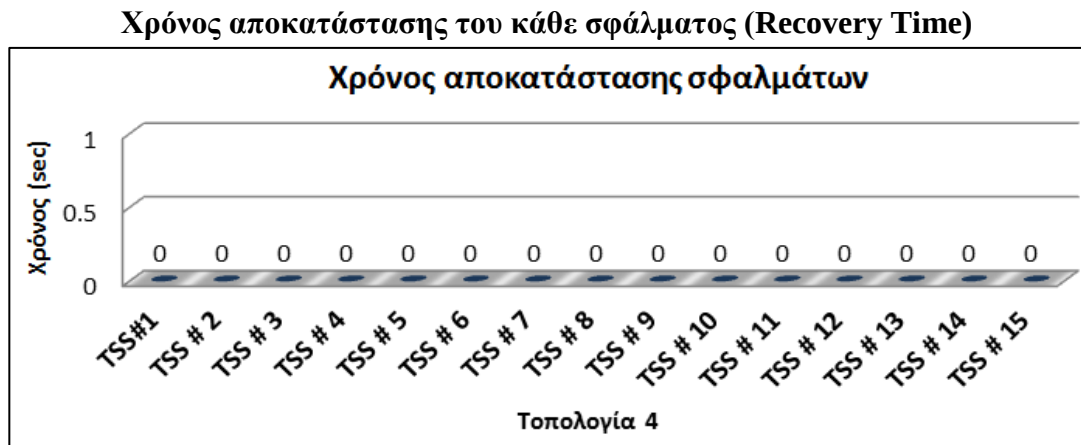
Ο χρόνος παραλαβής των πακέτων για οποιανδήποτε τιμή του threshold ισούται με το μέγιστο χρόνο που χρειάζεται το τελευταίο υπό-πακέτο να φτάσει στον egress κόμβο έτσι ώστε να μπορεί το αρχικό πακέτο να ανακατασκευαστεί. Λόγω του ότι τα σφάλματα προκύπτουν σε κόμβους που δεν αποτελούν τις ροές δρομολόγησης των πακέτων (μόνο δύο σφάλματα συμβαίνουν σε δυο ροές δρομολόγησης) δεν επηρεάζεται ο χρόνος παραλαβής των πακέτων εκτός στα σενάρια που το κατώφλι κοντεύει να ισούται με τον αριθμό των n ανεξάρτητων ροών δρομολόγησης.

Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.5.5 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 4

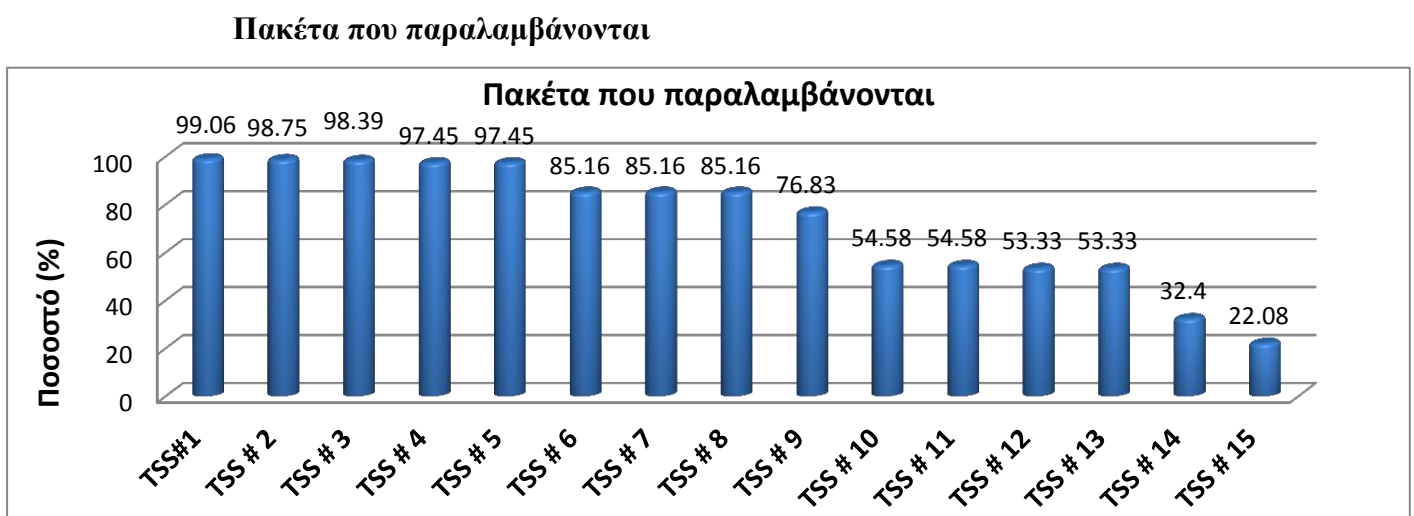
Ο όγκος δεδομένων που μεταφέρονται στις συνδέσεις εξαρτάται από την τιμή του κατωφλίου που ορίζεται. Για μικρότερη τιμή του threshold ο όγκος των δεδομένων είναι πολύ μεγαλύτερος από τον επόμενο (σχεδόν διπλάσιος από αυτόν όταν το κατώφλι ισούται με δυο). Αυτό οφείλεται στο ότι το μέγεθος των υπό-πακέτων θα ισούνται με το αποτέλεσμα της διαίρεσης του μεγέθους του αρχικού πακέτου δια τον αριθμό του κατωφλίου που ορίζεται.



Σχήμα 7.4.5.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

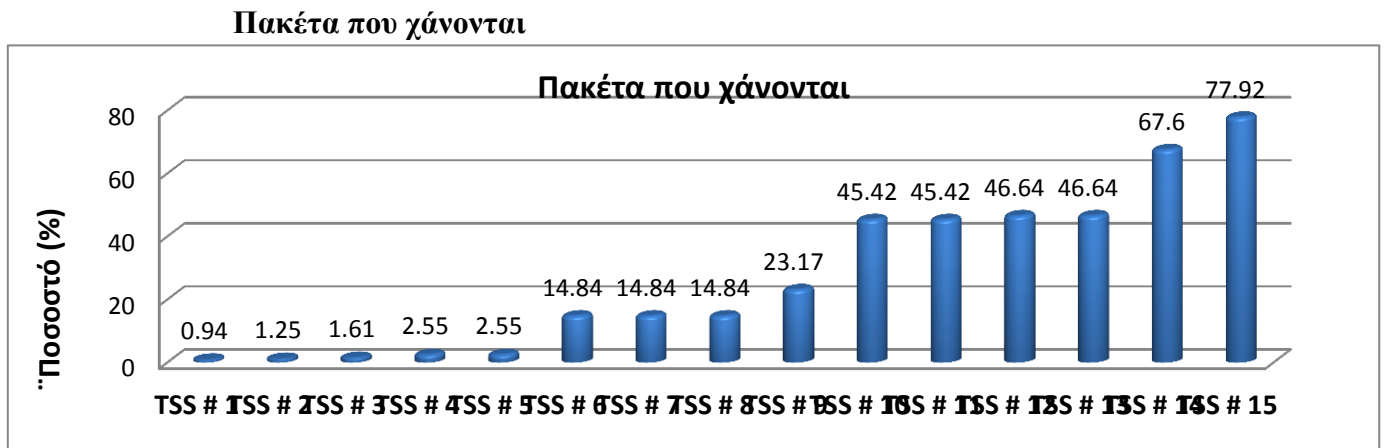
Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης.

7.4.2 Γραφικές Παραστάσεις για το Σενάριο 6



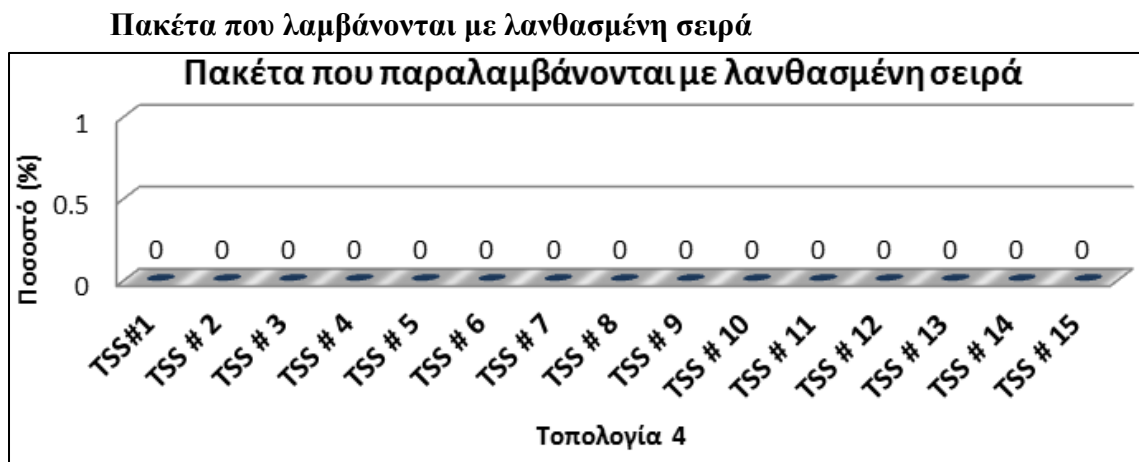
Σχήμα 7.4.6.1 – Ποσοστό πακέτων που παραλήφθηκαν

Από την πιο πάνω γραφική παράσταση παρατηρούμε πως το ποσοστό των πακέτων που παραλήφθηκαν είναι ψηλό μόνο για μικρές τιμές του threshold. Λόγω του ότι τα σφάλματα προκύπτουν σε κόμβους του τύπου disjoint, επηρεάζονται περισσότερες από μια ροές δρομολόγησης ως αποτέλεσμα να μειώνεται και το ποσοστό επιτυχίας.



Σχήμα 7.4.6.2 – Ποσοστό πακέτων που χάθηκαν

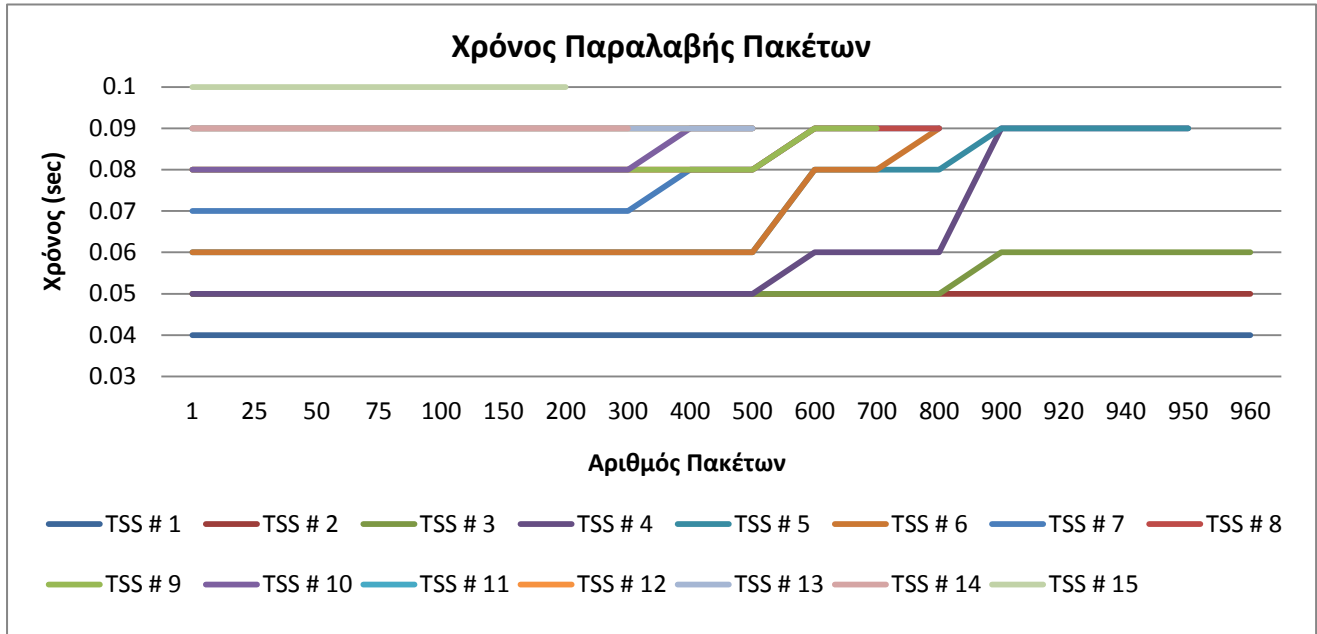
Το ποσοστό των πακέτων που χάνονται είναι χαμηλό μόνο για τις πρώτες πέντε τιμές του threshold. Μετά σταδιακά αυξάνεται και φτάνει μέχρι το 77.93%. Αυτό οφείλεται στο γεγονός ότι τα σφάλματα επηρεάζουν περισσότερες από μια ροή δρομολόγησης αφού οι κόμβοι που επηρεάζονται είναι τύπου disjoint.



Σχήμα 7.4.6.3 – Ποσοστό πακέτων που παραλήφθηκαν με λανθασμένη σειρά

Για όλες τις διαφορετικές τιμές του κατωφλίου, ο αλγόριθμος TSS έχει μηδενικά ποσοστά για τα πακέτα που παραλαμβάνονται με λανθασμένη σειρά.

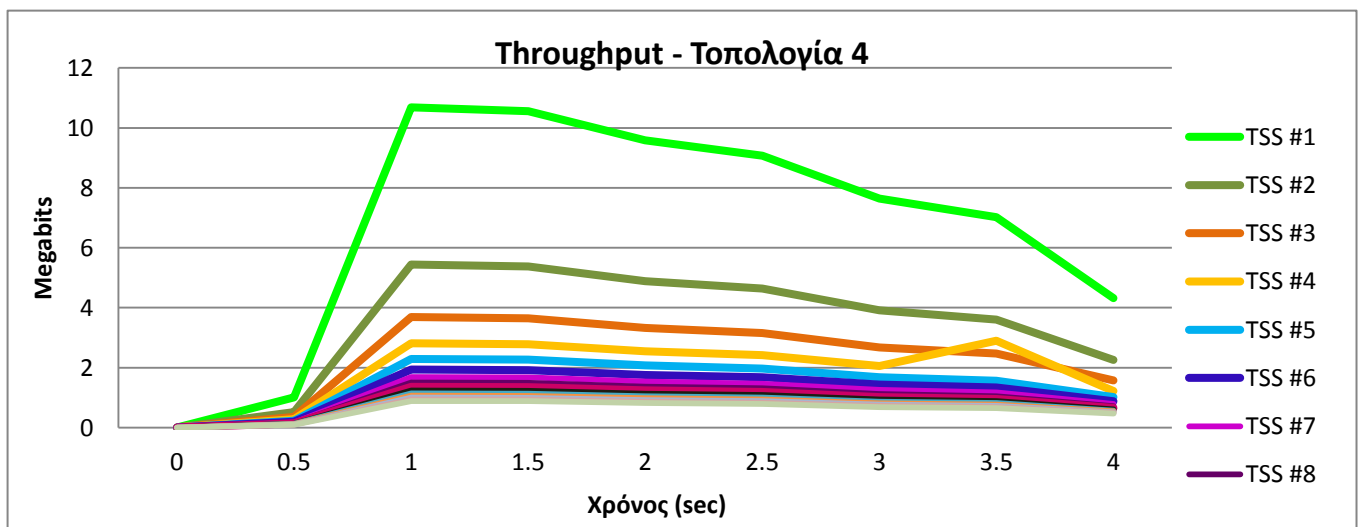
Χρόνος παραλαβής πακέτων (Delay)



Σχήμα 7.4.6.4 – Χρόνος παραλαβής των πακέτων

Ο χρόνος παραλαβής των πακέτων για οποιανδήποτε τιμή του κατωφλίου ισούται με το μέγιστο χρόνο που χρειάζεται το τελευταίο υπό-πακέτο να φτάσει στον τελικό κόμβο έτσι ώστε να μπορεί το αρχικό πακέτο να ανακατασκευαστεί. Λόγω του ότι τα σφάλματα προκύπτουν σε κόμβους που αποτελούν τις ροές δρομολόγησης πακέτων (κυρίως σε κόμβους που είναι τύπου disjoint) επηρεάζεται ο χρόνος παραλαβής πακέτων και στα σενάρια που το κατώφλι είναι σχετικά μικρό (το μισό των ανεξάρτητων ροών δρομολόγησης).

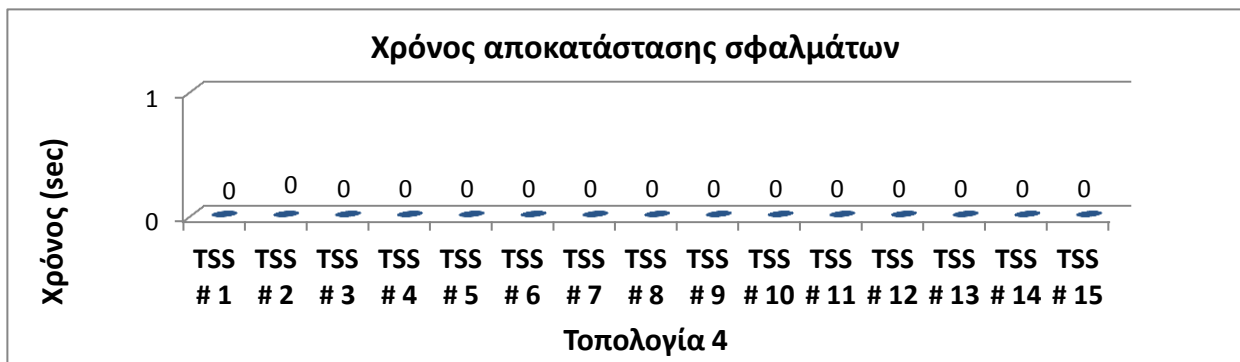
Δεδομένα που μεταφέρονται το δευτερόλεπτο (Throughput)



Σχήμα 7.4.6.5 – Δεδομένα που μεταφέρονται το δευτερόλεπτο στην Τοπολογία 4

Ο όγκος δεδομένων που μεταφέρονται στις συνδέσεις εξαρτάται από την τιμή του κατωφλίου που ορίζεται. Για μικρότερη τιμή του κατωφλίου ο όγκος των δεδομένων είναι πολύ μεγαλύτερος από τον επόμενο (σχεδόν διπλάσιος από αυτόν όταν το κατώφλι ισούται με δυο). Αυτό οφείλεται στο ότι το μέγεθος των υπό-πακέτων θα ισούνται με το αποτέλεσμα της διαίρεσης του μεγέθους του αρχικού πακέτου δια τον αριθμό του κατωφλίου που ορίζεται.

Χρόνος αποκατάστασης του κάθε σφάλματος (Recovery Time)



Σχήμα 7.4.6.6 – Χρόνος που χρειάστηκε για την αποκατάσταση σφαλμάτων

Ο χρόνος αποκατάστασης του δικτύου για τον TSS αλγόριθμο είναι μηδενικός λόγω του ότι δεν υπάρχει επαναφορά των ροών δρομολόγησης.

7.5 Συμπεράσματα

Ο αλγόριθμος TSS (k, n) με πολλαπλά μονοπάτια είναι αποδοτικός όταν χρησιμοποιείται σε δίκτυα με πολλές συνδέσεις/κόμβους, οι οποίες σχηματίζουν μεγάλο αριθμό διακριτών διαδρομών. Αξιοσημείωτο είναι το γεγονός ότι ο προτεινόμενος αλγόριθμος έχει υψηλό ποσοστό επιτυχίας όταν η τιμή του κατωφλίου (k) είναι μικρότερη από τον αριθμό των ροών διαδρομής n . Το γεγονός ότι ο αλγόριθμος αυτός βρίσκει από πριν τις n διαδρομές που θα δρομολογηθούν τα πακέτα δίνει πλεονέκτημα έναντι άλλων αλγορίθμων, συγκεκριμένα με τον υβριδικό αλγόριθμο που θα συγκρίνουμε αργότερα, στις μετρικές του χρόνου επαναφοράς και της παραλαβής πακέτων με λανθασμένη σειρά με τις τιμές τους να είναι ίσες με το μηδέν. Δηλαδή κανένα πακέτο δεν παραλαμβάνεται σε λανθασμένη σειρά και δεν υπάρχει η επαναφορά οποιασδήποτε διαδρομής στην οποία εμφανιστεί σφάλμα.

Επίσης ο προτεινόμενος αλγόριθμος προσφέρει ανοχή πολλαπλών σφαλμάτων όσο μεγαλύτερος είναι ο αριθμός των n ανεξάρτητων διαδρομών. Τα επίπεδα προστασίας είναι τα εξής:

- ▲ $n - k = 0$ – δεν προσφέρεται προστασία (μηδενική ανοχή σφαλμάτων)
- ▲ $n - k = 1$ – προσφέρεται ενοχή ενός σφάλματος μόνο
- ▲ $n - k > 1$ – προσφέρεται ανοχή πολλαπλών σφαλμάτων

Για να πετύχουμε όμως προστασία για πολλαπλά σφάλματα απαιτείται η προσθήκη και χρησιμοποίηση περισσότερων LSPs μεταξύ του ingress κόμβου και του egress κόμβου, γεγονός το οποίο θα αυξήσει όμως σε μεγάλο βαθμό το ρυθμό απόδοσης (throughput) για να μπορέσει να ικανοποιήσει τις ανάγκες δρομολόγησης που θα υπάρχουν.

Από τις γραφικές παραστάσεις των σεναρίων του προηγούμενου υποκεφαλαίου μπορούμε να παρατηρήσουμε ότι ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια έχει πολύ καλές επιδόσεις στα σενάρια των τοπολογιών, όπου τα σφάλματα συμβαίνουν σε κόμβους, οι οποίοι δεν αποτελούν τις προκαθορισμένες n διαδρομές. Στα σενάρια αυτά τα ποσοστά των πακέτων που παραλήφθηκαν με επιτυχία έχουν μεγάλη τιμή, η οποία σε μερικές περιπτώσεις ξεπερνά τα ποσοστά επιτυχίας του υβριδικού αλγορίθμου. Κυρίως στις τοπολογίες 1 και 2 στα σενάρια 3 και 4, το **ποσοστό επιτυχίας** του TSS (k,n) με πολλαπλά μονοπάτια αλγορίθμου φτάνει μέχρι το 99,38% ενώ του υβριδικού μόλις το 97%. Χωρίς να σημαίνει ότι ο υβριδικός αλγόριθμος υστερεί σε μεγάλο βαθμό, αφού αυτό ισχύει κυρίως για τις τιμές του κατωφλίου που είναι οι μισές των n ανεξάρτητων διαδρομών. Για μεγάλες τιμές κατωφλίου παρατηρούμε ότι το ποσοστό επιτυχίας μειώνεται μέχρι και το 38%.

Ο υβριδικός αλγόριθμος είναι πιο αποδοτικός όταν χρησιμοποιείται σε δίκτυα με πολλές συνδέσεις. Όσες λιγότερες είναι οι συνδέσεις στα δίκτυα, τόσο περισσότερο δυσκολεύεται να βρει εναλλακτικά μονοπάτια που να πληρούν τους περιορισμούς που ορίζονται από τη δομή του αλγορίθμου. Ως αποτέλεσμα ο χρόνος επαναφοράς του δικτύου και το ποσοστό των πακέτων που φτάνουν με λανθασμένη σειρά να αυξάνονται. Όπως επίσης επηρεάζεται και το ποσοστό επιτυχίας των πακέτων που παραλαμβάνονται επειδή χρειάζεται μερικός χρόνος για την έναρξη της επαναδρομολόγησης των πακέτων στις ροές. Ο υβριδικός αλγόριθμος μπορεί να

επιτυγχάνει μηχανική κυκλοφορία και ανοχή σφαλμάτων, με το να κατανέμει τις ροές που εισέρχονται στο δίκτυο χρησιμοποιώντας διαφορετικά μονοπάτια (εξισορροπείται το φορτίο) και να μεταφέρει μεγάλο αριθμό δεδομένων ανά δευτερόλεπτο, προσφέρεται η δυνατότητα στον αλγόριθμο να έχει πάντα διαθέσιμους πόρους και να εξασφαλίζει την επαναφορά της ροής. Έτσι με αποδοτικό τρόπο μπορεί να αντιμετωπίσει τυχαία σφάλματα που συμβαίνουν στο δίκτυο αλλά με κόστος να υπάρχει απώλεια πακέτων και λανθασμένη σειρά παραλαβής άλλων. Αυτό είναι μεγάλο μειονέκτημα στον υβριδικό αλγόριθμο γιατί πακέτα, τα οποία θεωρούνται σημαντικά, μπορεί να χαθούν λόγω σφαλμάτων στους κόμβους είτε να καθυστερήσουν να φτάσουν στον προορισμό λόγω του χρόνου απόκλισης που χρειάζεται ο αλγόριθμος για να επαναδρομολογήσει τα πακέτα στις νέες ροές και επηρεάζοντας τις διεργασίες που τα χρειάζονται άμεσα.

Στην αντίθετη περίπτωση ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια κάνει χρήση του δικτύου στο μέγιστο δυνατό. Όταν υπάρξει σφάλμα σε μια ροή δρομολόγησης του δικτύου, θα χαθεί μόνο το υπό-πακέτο που ακολουθούσε τη ροή αυτή. Τα υπόλοιπα υπό-πακέτα του αρχικού πακέτου δεν θα επηρεαστούν, με την προϋπόθεση ότι δεν υπήρχε σφάλμα στις δικές τους διαδρομές (μιας και το κάθε υπό-πακέτο ακολουθεί διαφορετική διαδρομή). Θεωρείτε ως επιτυχημένη παραλαβή ενός πακέτου μόλις φτάσουν τα πρώτα k υπό-πακέτα του, μιας και από αυτά μπορεί το αρχικό πακέτο να ξαναδημιουργηθεί. Αυτό έχει ως σκοπό σε περίπτωση που στέλλονται σημαντικά πακέτα μέσω των ροών και υπάρξει οποιοδήποτε σφάλμα να μην επηρεαστεί η διεργασία που χρειάζεται άμεσα τις πληροφορίες αυτές.

Όσα προαναφέρθηκαν μπορούν να επιβεβαιωθούν και από τις γραφικές παραστάσεις της προηγούμενης υποενότητας, παρατηρώντας ότι ο **χρόνος αποκατάστασης** του δικτύου για την επαναδρομολόγηση των πακέτων και το **ποσοστό παραλαβής των πακέτων με λανθασμένη σειρά** για τον υβριδικό αλγόριθμο να είναι σε χαμηλά επίπεδα ενώ για τον αλγόριθμο TSS (k,n) με πολλαπλά μονοπάτια τα αποτελέσματα για όλα τα σενάρια να είναι μηδέν. Ο μέγιστος χρόνος αποκατάστασης του δικτύου που χρειάστηκε ο υβριδικός αλγόριθμος σε όλα τα σενάρια των τοπολογιών να φτάνει μέχρι το 0,1 δευτερόλεπτα και το ποσοστό παραλαβής πακέτων με λανθασμένη σειρά να φτάνει και το 47%.

Με τον υβριδικό αλγόριθμο επιτυγχάνεται η μέγιστη χρήση των πόρων. Δηλαδή μπορούν να μεταφερθούν πολλά δεδομένα ανά δευτερόλεπτο, ιδιαίτερα στις ροές που υπάρχουν ψηλές προτεραιότητες. Μικρή μείωση υπάρχει την ώρα που υπολογίζονται τα εναλλακτικά μονοπάτια για μια ροή αλλά αυτό ισχύει μέχρι να αλλάξει το μονοπάτι στο οποίο θα δρομολογηθεί. Με τον αλγόριθμο TSS (k,n) με πολλαπλά μονοπάτια υπάρχει ταυτόχρονη αλλά και μέγιστη χρήση των πόρων στις ροές που γίνεται η δρομολόγηση των πακέτων (περισσότερη από όσο γίνεται από τον υβριδικό αλγόριθμο). Όσο πιο μικρό είναι το κατώφλι που απαιτείται για την ανακατασκευή του αρχικού πακέτου, τόσο μεγαλύτερος είναι ο όγκος δεδομένων που υπάρχει στο δίκτυο. Αυτό οφείλεται κυρίως στο γεγονός ότι αν απαιτείται μικρότερο κατώφλι τα υπό-πακέτα του αρχικού πακέτου που θα δημιουργηθούν θα έχουν μέγεθος ίσο με το μέγεθος του αρχικού πακέτου δια το κατώφλι που ορίστηκε. Για παράδειγμα εάν έχουμε αρχικό πακέτο μεγέθους 200 bytes και το κατώφλι ισούται με ένα, τότε για κάθε ανεξάρτητη k διαδρομή θα δρομολογηθεί ένα πακέτο μεγέθους 200 bytes. Στην περίπτωση που το κατώφλι ισούται με δύο, για κάθε ανεξάρτητη διαδρομή θα δρομολογηθεί ένα πακέτο μεγέθους 100 bytes. Αυτό μπορούμε να το παρατηρήσουμε και από τις γραφικές παραστάσεις του **throughput**, όπου για τις μικρότερες τιμές του κατωφλίου τα δεδομένα που μεταφέρονται στις διαδρομές έχουν το μεγαλύτερο όγκο. Μείωση υπάρχει μόνο στις χρονικές στιγμές που δημιουργείται κάποιο σφάλμα και επηρεάζει κάποιο κόμβο που ανήκει στις ροές δρομολόγησης των πακέτων. Όταν προκύψει κάποιο σφάλμα στις ροές δρομολόγησης η μέγιστη χρήση των πόρων μειώνεται οριστικά στο επόμενο μέγιστο σημείο χρήσης των πόρων. Δηλαδή όταν έχουμε τέσσερα μονοπάτια δρομολόγησης η μέγιστη χρήση των πόρων ισούνται με το άθροισμα των πόρων που χρειάζονται και οι τέσσερις ροές. Στην περίπτωση σφάλματος που επηρεάζεται μια ή περισσότερες από τις τέσσερις ροές η μέγιστη χρήση των πόρων θα ισούται πλέον με το συνολικό άθροισμα πόρων που χρησιμοποιούν οι εναπομείναντες διαδρομές.

Από τα σενάρια των τοπολογιών 1 μέχρι 4, παρατηρήσαμε πως λειτουργεί ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια σε περιπτώσεις που η τοπολογία αποτελείται από μικρό αριθμό κόμβων και σαφώς μικρό αριθμό n ανεξάρτητων διαδρομών. Το αποτέλεσμα από τις τοπολογίες αυτές είναι ότι στην ύπαρξη σφαλμάτων σε κόμβους που βρίσκονται στις ροές δρομολόγησης των πακέτων ο προτεινόμενος

αλγόριθμος δεν αποδίδει καλά σε σχέση με τον υβριδικό αλγόριθμο, κυρίως όταν το απαιτούμενο κατώφλι ισούται με το μισό αριθμό των ανεξάρτητων διαδρομών δρομολόγησης. Το ιδανικό σε τέτοια δίκτυα (με μικρό αριθμό ανεξάρτητων διαδρομών) είναι να υπάρχει πολύ χαμηλό απαιτούμενο κατώφλι.

Ενώ από την τοπολογία 4 (σενάρια 5 και 6), παρατηρήσαμε πως λειτουργεί ο αλγόριθμος TSS (k, n) με πολλαπλά μονοπάτια σε τοπολογία με μεγάλο αριθμό κόμβων και n ανεξάρτητων διαδρομών (στην τοπολογία αυτή υπάρχουν και disjoint κόμβοι, οι οποίοι δημιουργούν δυο ή περισσότερες ροές δρομολόγησης). Στο σενάριο στο οποίο τα σφάλματα γίνονται σε συνδέσεις κόμβων που βρίσκονται στις ροές δρομολόγησης και δεν είναι disjoint (σενάριο 5) παρουσιάζει αποτελέσματα με ψηλό το ποσοστό παραλαβής πακέτων με επιτυχία. Τα πιο χαμηλά αποτελέσματα παρουσιάζονται στις περιπτώσεις που το κατώφλι είναι 14 και 15, δηλαδή όταν ισούται με το μέγιστο αριθμό των k ανεξάρτητων διαδρομών. Το throughput της τοπολογίας, στο σενάριο 5, βρίσκεται σε ψηλά επίπεδα όταν το κατώφλι είναι πολύ χαμηλό, αφού για πολύ μικρό κατώφλι το μέγεθος των υπό-πακέτων ισούται σχεδόν με το μέγεθος του αρχικού πακέτου.

Στο σενάριο 6 που εμφανίζονται σφάλματα σε κόμβους disjoint το ποσοστό αποτυχίας παραλαβής πακέτων αυξάνεται όταν το κατώφλι ισούται ή είναι μεγαλύτερο από το μισό αριθμό των k ανεξάρτητων διαδρομών. Το throughput του σεναρίου όταν προκύψουν τα σφάλματα στους disjoint κόμβους μειώνεται μέχρι και στα 0,47 Megabits. Τα αποτελέσματα αυτά οφείλονται στο γεγονός ότι όταν υπάρξει σφάλμα σε κόμβο τύπου disjoint επηρεάζονται δυο ή περισσότερες ροές δρομολόγησης των πακέτων, επηρεάζοντας έτσι και τον αριθμό των υπό-πακέτων που χρειάζονται για την κατασκευή του αρχικού πακέτου.

Βάση της δομής του αλγορίθμου κανένα πακέτο δεν παραλαμβάνεται με λανθασμένη σειρά ούτε υπάρχει οποιαδήποτε καθυστέρηση στην επαναφορά του δικτύου, με την προϋπόθεση ότι όταν φτάσουν στον τελικό κόμβο k υπό-πακέτα μπορεί να δημιουργηθεί ξανά το αρχικό πακέτο. Αυτό οφείλεται στην αρμοδιότητα του τελικού κόμβου, ο οποίος μπορεί να αποθηκεύσει στο buffer του αριθμό υπό-πακέτων ίσο με συνολικό αριθμό υπό-πακέτων που στέλλονται από όλες τις διαδρομές. Έτσι όλα τα πακέτα καταλήγουν στον τελικό κόμβο με την αρχική σειρά που στέλλονται και μόλις παραληφθεί το k -οστό υπό-πακέτο (το οποίο θα φτάσει από τη διαδρομή με το

μεγαλύτερο χρόνο δρομολόγησης από τις k διαδρομές) που χρειάζεται για την ανακατασκευή του αρχικού πακέτου θα ανακατασκευαστεί το αρχικό με επιτυχία.

Κεφάλαιο 8

Συμπεράσματα

8.1 Γενικά Συμπεράσματα	89
8.2 Μελλοντική Εργασία	91

8.1 Γενικά Συμπεράσματα

Οι δυο αλγόριθμοι, Υβριδικός και TSS (k,n) με πολλαπλά μονοπάτια, είναι εξίσου αποτελεσματικοί σε δίκτυα με μεγάλο αριθμό συνδέσεων και συνεπώς διακριτών ροών δρομολόγησης. Χαρακτηρίζονται από τις καλές επιδόσεις που έχουν στον αριθμό των πακέτων παραλαβής, στο μικρό χρόνο αποκατάστασης σφαλμάτων που συμβαίνουν στο δίκτυο, στο μικρό ποσοστό πακέτων που παραλαμβάνονται σε λανθασμένη σειρά και ένα μεγάλο ποσοστό χρησιμοποίησης των πόρων που τους προσφέρεται από την τοπολογία.

Ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια υπερτερεί σε ένα πολύ μικρό βαθμό έναντι του υβριδικού στις υπηρεσίες παραλαβής πακέτων με λανθασμένη σειρά και στο χρόνο αποκατάστασης σφαλμάτων, διότι λόγω της δομής του δεν κάνει επαναδρομολόγηση πακέτων όταν παρουσιαστούν σφάλματα στο δίκτυο με την προϋπόθεση ότι στο τέλος θα φτάσουν στον τελικό κόμβο k υπό-πακέτα που χρειάζονται για τη δημιουργία του αρχικού πακέτου. Αυτό δίνει το πλεονέκτημα γιατί τα πακέτα που χαρακτηρίζονται ως σημαντικά για την άμεση διεκπεραίωση μιας

εργασίας έχουν τη δυνατότητα να φτάσουν στον προορισμό τους στο δίκτυο και να σχηματίσουν το αρχικό πακέτο αντί να χαθούν ή να φτάσουν αργοπορημένα (αξιοποίηση των πόρων που προσφέρει το δίκτυο στο μέγιστο δυνατό).

Το σημείο που κάνει τον υβριδικό αλγόριθμο να υπερτερεί από τους δυο είναι ότι δυναμικά θα προσπαθήσει να βρει εναλλακτικά μονοπάτια για τη δρομολόγηση των πακέτων σε περίπτωση σφαλμάτων ακόμα και στην κατάρρευση πολλών κόμβων. Σε αντίθεση ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια επειδή δεν χρησιμοποιεί τη δυναμική εύρεση μονοπατιού για την δρομολόγηση των πακέτων, όταν παρουσιαστούν σφάλματα σε $n-k+1$ μονοπάτια, αυτόματα ο αλγόριθμος αδυνατεί να δημιουργήσει το αρχικό πακέτο και το δίκτυο να καταρρέει. Η χειρότερη περίπτωση είναι όταν το δίκτυο αποτελείται από πολλούς disjoint κόμβους και τα σφάλματα εμφανίζονται σε αυτούς τους κόμβους, όπου ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια λόγω του ότι υπολογίζει από πριν τα μονοπάτια ροής δεν θα μπορεί να λειτουργεί με επιτυχία. Ενώ ο υβριδικός αλγόριθμος θα μπορεί σε κάθε σφάλμα να βρίσκει δυναμικά διαφορετική ανεξάρτητη ροή για τη δρομολόγηση πακέτων πριν τη παρουσίαση του επόμενου πιθανού σφάλματος. Η ανεξάρτητη ροή που θα καθορίζει ο υβριδικός αλγόριθμος ισχύει μόνο στις περιπτώσεις όπου το δίκτυο δημιουργείται από πολλούς κόμβους και ο αλγόριθμος θα μπορεί να βρει δυο ροές, κύρια και εναλλακτική, που να αποτελούνται από κόμβους που δεν ανήκουν και στις δυο ροές.

Σε περίπτωση που το δίκτυο αποτελείται από μικρό αριθμό κόμβων τότε υπάρχει το ενδεχόμενο η κύρια και εναλλακτική ροή δρομολόγησης πακέτων να έχουν μερικούς κοινούς κόμβους. Σε περιπτώσεις σφάλματος σε αυτούς τους κόμβους ο χρόνος παραλαβής των πακέτων θα αυξάνεται γιατί θα πρέπει να καθοριστεί νέα ροή δρομολόγησης των πακέτων τόσο κύρια όσο και εναλλακτική. Ενώ ο αλγόριθμος TSS (k,n) με πολλαπλά μονοπάτια στις περιπτώσεις πολλαπλών σφαλμάτων θα κινδυνεύει να μη μπορέσει να λειτουργεί σωστά και να έχει μεγάλα ποσοστά αποτυχίας γιατί δεν θα μπορεί να υπολογίσει δυναμικά εναλλακτικά μονοπάτια.

Η πολυπλοκότητα του τροποποιημένου αλγορίθμου κατωφλίου TSS (k,n) με πολλαπλά μονοπάτια είναι πάντα της τάξεως του $O\left(\frac{(a*c)+(b^2*a^3)}{b}\right)$, όπου το a ισούται με το μέγεθος του IP πακέτου που εισέρχεται στον κόμβο εισόδου, το b είναι το μέγεθος των μπλοκ που δημιουργούνται από τον τεμαχισμό του αρχικού πακέτου και το c είναι ο αριθμός των LSPs που χρησιμοποιούνται για τη δρομολόγηση των υπό-πακέτων από τον κόμβο εισόδου στον κόμβο εξόδου. Στην περίπτωση που η μεταβλητή

$c > b^2 \cdot a^2$ τότε η πολυπλοκότητα είναι της τάξεως του $O(\frac{a \cdot c}{b})$ διαφορετικά είναι $O(b \cdot a^3)$. Ενώ η πολυπλοκότητα του υβριδικού αλγορίθμου είναι μεγαλύτερη και αυξάνεται κάθε φορά που παρουσιάζονται σφάλματα στο δίκτυο.

8,2 Μελλοντική Εργασία

Στην παρούσα διπλωματική εργασία έγινε προσπάθεια για να συγκριθούν δυο από τους καλύτερους αλγόριθμους που συνδυάζουν μηχανική κυκλοφορία και ανοχή σφαλμάτων. Ο αλγόριθμος κατωφλίου TSS (k,n) με πολλαπλά μονοπάτια προσφέρει συνεχή δρομολόγηση των δεδομένων στο δίκτυο με μηδενικό χρόνο επαναφοράς αλλά ταυτόχρονα η εγγύηση στην ανοχή σφαλμάτων περιορίζεται όταν ο αριθμός του κατωφλίου ξεπερνά το μισό αριθμών των ανεξάρτητων ροών δρομολόγησης. Στις περιπτώσεις όμως που δεν υπάρχουν σφάλματα στα κυρίως μονοπάτια ο αλγόριθμος αυτός έχει άριστες επιδόσεις. Επίσης ο αλγόριθμος κατωφλίου εκμεταλλεύεται όλους τους πόρους του δικτύου. Από την άλλη ο υβριδικός αλγόριθμος προσφέρει καλούς χρόνους επαναφοράς του δικτύου στις περιπτώσεις που δεν επηρεάζεται το εναλλακτικό μονοπάτι δρομολόγησης και παράλληλα υπάρχουν οι απαιτούμενοι πόροι για να φιλοξενήσουν τη ροή δρομολόγησης.

Τέλος γίνεται εισήγηση να τροποποιηθεί ο υβριδικός αλγόριθμος έτσι ώστε να συνδυάζεται με τον τροποποιημένο αλγόριθμο κατωφλίου με πολλαπλά μονοπάτια. Μπορεί να υιοθετήσει την πολιτική του αλγορίθμου κατωφλίου για τη μέγιστη δυνατή χρησιμοποίηση των πόρων που προσφέρονται από το δίκτυο έτσι ώστε σε περίπτωση σφάλματος ο χρόνος αποκατάστασης του δικτύου και η παραλαβή των πακέτων με λανθασμένη σειρά να έχουν μηδενικές τιμές. Χρησιμοποιώντας τους αλγόριθμους ανοχής σφαλμάτων (Otel και GONFA) που χρησιμοποιεί ο υβριδικός αλγόριθμος θα μπορεί μετά την εμφάνιση σφάλματος σε μια κύρια ροή δρομολόγησης να γίνει δυναμικά (εάν είναι εφικτή) η εύρεση ενός νέου εναλλακτικού μονοπατιού που να χρησιμοποιεί τους κόμβους οι οποίοι δεν χρησιμοποιούνται στα υπόλοιπα κυρίως μονοπάτια έτσι ώστε να μπορεί να γίνεται η δρομολόγηση και η ανακατασκευή των πακέτων με όσο το δυνατό λιγότερες απώλειες ακόμα και στις περιπτώσεις που το

κατώφλι τείνει να ισούται με τον αριθμό των ανεξάρτητων διαδρομών δρομολόγησης. Συνδυάζοντας αυτές τις πολιτικές μπορούμε να πετύχουμε μια γρήγορη και αξιόπιστη πολιτική δρομολόγησης πακέτων σε MPLS δίκτυα με υψηλό το ποσοστό επιτυχίας και χαμηλό χρόνο αποκατάστασης όπου παράλληλα θα αξιοποιούμε στο μέγιστο δυνατό όλους τους πόρους του δικτύου.

Βιβλιογραφία

- [1] A. Viswanathan & R. Callon E. Rosen, "Multiprotocol Label Switching Architecture," January 2001.
- [2] CISCO, "Advanced Topics in MPLS-TE Deployment," p. 93, 1992-2002.
- [3] A. Agarwal & A. En-Nouaary S. Alouneh, "A novel path protection scheme for MPLS networks using multi-path routing," February 2009.
- [4] Μ. Παπά, "Ανοχή Σφαλμάτων και Μηχανική Κυκλοφορίας σε Δίκτυα MPLS - Υβριδικός Αλγόριθμος," Πανεπιστήμιο Κύπρου, Λευκωσία, 2009.
- [5] Inc Juniper Networks. (1999-2010) How MPLS Works. [Online].
<http://www.juniper.net/techpubs/software/erx/junose72/swconfig-bgp-mpls/html/mpls-config5.html>
- [6] S. Syed, "How MPLS works - The Journey of a Network Engineer," February 2011.
- [7] J. Ahmed Zubairi, "An Automated Traffic Engineering Algorithm for MPLS-Diffsev Domain".
- [8] L. Hundessa, "Enhanced Fast Rerouting Mechanisms For Protected Traffic In MPLS Networks," p. 148, February 2003.
- [9] Cariden Technologies Inc, "IP/MPLS Traffic Engineering," May 2012.
- [10] D. Awduche, "RSVP-TE, Extensions to RSVP for LSP Tunnels," December 2001.
- [11] B. Davie & Y. Rekhter, "MPLS: Technology and Applications," 2000.
- [12] Δρ. Πιτσιλλίδης Ανδρέας. (EPL324, 2011) MPLS - Multi Protocol Label Switching.
- [13] CISCO. MPLS Traffic Engineering. [Online].
http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/TE_1208S.html
- [14] Cisco Systems, "MPLS Traffic Engineering v2.1," 2002.
- [15] J. Patrao, C. Lopes, A. de Sousa & R. Valadas L. Cardoso, "A Tool for MPLS Traffic Engineering,".
- [16] Juniper. [Online]. http://www.juniper.net/techpubs/en_US/junos9.5/information-products/topic-collections/config-guide-mpls-applications/mpls-configuring-primary-and-secondary-lsps.html#id-23856
- [17] D. Otel, "On Fast Computing Bypass Tunnel Routes in MPLS- Based Local Restoration," pp. 234 - 238, 2002.
- [18] Henry Owen, David Barlow Sven Owen, "Evaluation of the Local State Fair Share Bandwidth Algorithm".
- [19] A. Shamir, "How to share a secret," vol. 22, no. 11, p. 613, November 1979.
- [20] Inc Wikimedia Foundation. (2013, February) Lagrange polynomial. [Online].
http://en.wikipedia.org/wiki/Lagrange_polynomial#Main_idea
- [21] CE 341/441. (2004) Langrange Interpolation. [Online].
<http://www3.nd.edu/~jjwteach/441/PdfNotes/lecture6.pdf>
- [22] R. Bhandari, *Survivable Networks, Algorithms for Diverse Routing.*: Kluwer

- Academic Publishers, 1999.
- [23] Y. Shiloach, "A Polynomial Solution to the undericted Two Path Problems," pp. 445-446, 1980.
 - [24] G. Swallow, L. Martini & D. McPherson S. Bryant, "Pseudowire Emulation Edge-toEdge Control Word for use over an MPLS PSN," vol. RFC4385, no. IETF, February 2006.
 - [25] J. Chung & M. Claypool. NS by Example. [Online]. <http://nile.wpi.edu/NS/>
 - [26] M. Greis & NS users. Tutorial for the Network Simulator "NS". [Online]. <http://www.isi.edu/nsnam/ns/tutorial/>
 - [27] T. Issariyakul. (2010-2013) NS2 Ultimate. [Online]. <http://www.ns2ultimate.com/>
 - [28] T. Issariyakul & E. Hossain. (2008) introduction to Network Simulator NS2. [Online]. <http://www.ece.ubc.ca/~teerawat/publications/NS2/15-Packets.pdf>
 - [29] B. Ghaddar, S. Khawam, H. Safa, H. Artail & Z. Dawy J. Naoum-Sawaya. (2005, August) Adaptive Approach for QoS Support in IEEE 802.11e Wireless LAN. [Online]. <http://hpds.ee.ncku.edu.tw/~smallko/ns2/wireless-udp-1.htm>
 - [30] E. Altman & T. Jimenez, "NS Simulator for beginners," 2003-2004.

Παράρτημα Α

Στο παράρτημα Α παρουσιάζουμε πως υλοποιήσαμε τη δρομολόγηση των πακέτων στο δίκτυο για τον προτεινόμενο αλγόριθμο κατωφλίου (k,n) με πολλαπλά μονοπάτια στο εργαλείο Network Simulator 2 στη γλώσσα OTcl. Υπάρχουν συνολικά πέντε τοπολογίες με τέσσερα διαφορετικά σενάρια. Τα σενάρια αυτά έχουν το δικό τους χρονοδιάγραμμα στο οποίο συμβαίνουν τα αντίστοιχα σφάλματα. Ακολουθεί ο κώδικας για ένα παράδειγμα της υλοποίησης του προτεινόμενου αλγορίθμου, όπου παρουσιάζεται η βασική δομή του προγράμματος.

```
# We use multi cast instead of multi path [MPLS] technology for the packet routing because
# we had to modified the existing files of the Network Simulator 2, so it can route
# simultaneously multiple packets from different routes. To implement each scenario,
# necessary changes will be made [nodes, routes, links, receivers]
#-----
# Create an event scheduler wit multicast turned on
set ns [new Simulator -multicast on]
$ns multicast

# Turn on Tracing
set tf [open file_trace.tr w]
$ns trace-all $tf

# Turn on nam Tracing
set fd [open out.nam w]
$ns namtrace-all $fd

# Create a text that contains info for each starting node
# and there packet routing
set init_file [open file_init.tr w]
set init_file2 [open file_init2.tr w]
#-----

# Finish procedure to stop the emulator
proc finish {} {
    global ns tf init_file init_file2
```

```

        $ns flush-trace
        close $tf
        close $init_file
        exec nam out.nam &
        exit 0
    }
#-----

# Create nodes
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 n8 [$ns node]
set n10 [$ns node]
#-----

# Create links between the nodes. The links of the nodes are the simplest form of the
# network so we can apply the modified TSS (k,n) algorithm
$ns duplex-link $n0 $n2 1Mb 10ms DropTail
$ns duplex-link $n1 $n2 1Mb 10ms DropTail
$ns duplex-link $n2 $n3 1Mb 10ms DropTail
$ns duplex-link $n2 $n4 1Mb 10ms DropTail
$ns duplex-link $n2 $n6 1Mb 10ms DropTail
$ns duplex-link $n2 $n8 1Mb 10ms DropTail
$ns duplex-link $n4 $n5 1Mb 10ms DropTail
$ns duplex-link $n6 $n7 1Mb 10ms DropTail
$ns duplex-link $n8 $n10 1Mb 10ms DropTail
#-----

# Manual Nodes Layout [Path Layouts]
$ns duplex-link-op $n0 $n2 orient right-down

```

\$ns duplex-link-op \$n1 \$n2 orient right-up

From the ingress node to the path/nodes and finally to the egress nodes

\$ns duplex-link-op \$n2 \$n3 orient right-up

\$ns duplex-link-op \$n2 \$n4 orient right

\$ns duplex-link-op \$n2 \$n6 orient right-down

\$ns duplex-link-op \$n2 \$n8 orient down

\$ns duplex-link-op \$n4 \$n5 orient down-right

\$ns duplex-link-op \$n6 \$n7 orient down-right

\$ns duplex-link-op \$n8 \$n10 orient down-right

#-----

routing protocol [set distance vector]

set mproto DM

set mrthandle [\$ns mrtproto \$mproto {}]

allocate group addresses

set group1 [Node allocaddr]

set group2 [Node allocaddr]

UDP Transport agent for the traffic source & Setup a CBR over UDP connection

Assign values to the packetSize and to the rate as well

set udp0 [new Agent/UDP]

\$ns attach-agent \$n0 \$udp0

\$udp0 set dst_addr_ \$group1

\$udp0 set dst_port_ 0

set cbr1 [new Application/Traffic/CBR]

\$cbr1 set packetSize_ 200

\$cbr1 set rate_ 400Kb

\$cbr1 attach-agent \$udp0

Transport agent for the traffic source

set udp1 [new Agent/UDP]

\$ns attach-agent \$n1 \$udp1

\$udp1 set dst_addr_ \$group2

\$udp1 set dst_port_ 0

```
set cbr2 [new Application/Traffic/CBR]
```

```
$cbr2 set packetSize_ 200
```

```
$cbr2 set rate_ 400Kb
```

```
$cbr2 attach-agent $udp1
```

```
#-----
```

```
# the egress node will be formed by a number of different receivers and each one will receive
```

```
# the packets of each route.
```

```
# create receivers for source node A
```

```
set rcvr1 [new Agent/Null]
```

```
$ns attach-agent $n3 $rcvr1
```

```
$ns at 0.4 "$n3 join-group $rcvr1 $group1"
```

```
set rcvr2 [new Agent/Null]
```

```
$ns attach-agent $n5 $rcvr2
```

```
$ns at 0.4 "$n5 join-group $rcvr2 $group1"
```

```
set rcvr3 [new Agent/Null]
```

```
$ns attach-agent $n7 $rcvr3
```

```
$ns at 0.4 "$n7 join-group $rcvr3 $group1"
```

```
set rcvr4 [new Agent/Null]
```

```
$ns attach-agent $n10 $rcvr4
```

```
$ns at 0.4 "$n10 join-group $rcvr4 $group1"
```

```
# Create receivers for source node B
```

```
set rcvr5 [new Agent/Null]
```

```
$ns attach-agent $n3 $rcvr5
```

```
$ns at 0.4001 "$n3 join-group $rcvr5 $group2"
```

```
set rcvr6 [new Agent/Null]
```

```
$ns attach-agent $n5 $rcvr6
```

```
$ns at 0.4001 "$n5 join-group $rcvr6 $group2"
```

```
set rcvr7 [new Agent/Null]
```

```
$ns attach-agent $n7 $rcvr7
```



```
$ns at 0.4001 "$n7 join-group $rcvr7 $group2"
```

```
set rcvr8 [new Agent/Null]
```

```
$ns attach-agent $n10 $rcvr8
```

```
$ns at 0.4001 "$n10 join-group $rcvr8 $group2"
```

```
#-----
```

```
# Schedule events
```

```
$ns at 0.4 "$cbr1 start"
```

```
$ns at 4.0 "$cbr1 stop"
```

```
$ns at 0.4001 "$cbr2 start"
```

```
$ns at 4.0 "$cbr2 stop"
```

```
# Set the ERRORS that will occurred during the simulation. Each one error is the same that it
```

```
# will be happened in hybrids scenario/topology. Each scenario have different errors.
```

```
$ns rtmodel-at 1.8 down $n2 $n4
```

```
$ns rtmodel-at 2.5 down $n2 $n3
```

```
$ns rtmodel-at 3.0 down $n2 $n6
```

```
$ns at 4.0 "finish"
```

```
#-----
```

```
# For nam
```

```
# Colors for packets from two mcast groups
```

```
$ns color 10 red
```

```
$ns color 11 green
```

```
$ns color 30 purple
```

```
$ns color 31 green
```

```
# Source & Receiver Node Labels
```

```
$udp0 set fid_ 10
```

```
$n0 color red
```

```
$n0 label "Src-I"
```

```
$udp0 set fid_ 11
```

```
$n1 color green
```

```
$n1 label "Src-II"
$n3 label "Rcvr-A"
$n3 color blue
$n3 shape box
$n5 label "Rcvr-B"
$n5 color blue
$n5 shape box
$n7 label "Rcvr-C"
$n7 color blue
$n7 shape box
$n10 label "Rcvr-D"
$n10 color blue
$n10 shape box
```

```
# Write to the file the init & end time as well the init node, the final nodes
puts $init_file "0.4 4.0 0 3 5 7 9"
puts $init_file2 "0.4001 4.0 1 3 5 7 9"
```

```
# Animation rate [simulation speed]
$ns set-animation-rate 1.0ms
```

```
# Start the simulation
$ns run
```

Παράρτημα Β

Στο παράρτημα Β παρουσιάζουμε πως υπολογίζουμε τον όγκο των δεδομένων που μεταφέρονται στις ροές δρομολόγησης. Ακολουθεί ο κώδικας για ένα παράδειγμα της υλοποίησης του υπολογισμού αυτού, όπου παρουσιάζεται στα αρχεία υλοποίησης awk.

```
BEGIN {  
    # set the threshold that is needed  
    packetSplit;  
    fromNode;  
  
    # init the variables  
    toNode3 = 3;  
    toNode5 = 5;  
    toNode7 = 7;  
    toNode9 = 9;  
    totalByte = 0;  
    totalByte3 = 0;  
    totalByte5 = 0;  
    totalByte7 = 0;  
    totalByte9 = 0;  
    packetCount = 0;  
    time = 1;  
  
    print "";  
    print " == Start == packetSplit " packetSplit;  
}  
  
# takes all the packets that are sent from the source node and reach successfully the egress  
/^r/&&($8==fromNode && ($4==toNode3 || $4==toNode5 || $4==toNode7 || $4==toNode9))  
{  
    if( ($2 > 0.5 * time) ){  
        # Conver to Megabit from Bytes  
        totalByte = (totalByte3 + totalByte5 + totalByte7 + totalByte9) / (131072);  
        # print the results
```

```

        print time*0.5;
        print " - Total transmitted = " totalByte " Megabits";

        packetCount = 0;
        totalByte3 = 0;
        totalByte5 = 0;
        totalByte7 = 0;
        totalByte9 = 0;
        time++;
    }
    packetCount++;

    # calculate the packet sizes
    if($4==toNode3)
        totalByte3 += ($6/packetSplit);
    else if($4==toNode5)
        totalByte5 += ($6/packetSplit);
    else if($4==toNode7)
        totalByte7 += ($6/packetSplit);
    else if($4==toNode9)
        totalByte9 += ($6/packetSplit);
};

END{
    # print the results
    print time*0.5;

    totalByte = (totalByte3 + totalByte5 + totalByte7 + totalByte9) / (131072);
    print " - Total transmitted = " totalByte " Megabits";

    print "";
    print " == End == ";
    print "";
};

```