

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

**Αλγόριθμοι δρομολόγησης οπτικών δικτύων σε
επίπεδο IP**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΑΝΤΖΑΡΕΑ ΑΝΑΣΤΑΣΙΑ

Επιβλέπων : Εμμανουήλ Βαρβαρίγος
Καθηγητής Ε.Μ.Π.

Αθήνα, Δεκέμβριος 2016



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αλγόριθμοι δρομολόγησης οπτικών δικτύων σε επίπεδο IP

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΑΝΤΖΑΡΕΑ ΑΝΑΣΤΑΣΙΑ

Επιβλέπων : Εμμανουήλ Βαρβαρίγος
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την Δεκεμβρίου 2016 .

(Υπογραφή)

.....
Εμμανουήλ Βαρβαρίγος
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....
Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

(Υπογραφή)

.....
Βασίλειος Μάγκλαρης
Καθηγητής Ε.Μ.Π.

Αθήνα, Δεκέμβριος 2016

(Υπογραφή)

.....
ΜΑΝΤΖΑΡΕΑ ΑΝΑΣΤΑΣΙΑ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας αποτελεί η ανάπτυξη αλγορίθμων βέλτιστης δρομολόγησης για οπτικά δίκτυα σε επίπεδο IP και η ενσωμάτωση τους στο εργαλείο σχεδιασμού πολυεπίπεδων οπτικών δικτύων MANTIS. Λόγω της αλματώδους διάδοσης του ARPANET, του πρώτου δικτύου μεταγωγής πακέτου του κόσμου, στα μέσα του 1970, προέκυψε η ανάγκη να βρεθεί μια αυτοματοποιημένη μέθοδος που να αποφασίζει για την βέλτιστη δρομολόγηση των μηνυμάτων που διαβιβάζονταν καθώς και για την βέλτιστη διάταξη του δικτύου. Η μέθοδος Flow deviation (FD) που προτάθηκε για τον σκοπό αυτό, εφαρμόστηκε τελικά σε πολλές περιπτώσεις προβλημάτων που μπορούν να μοντελοποιηθούν ως προβλήματα ανάθεση ροών σε ένα δίκτυο.

Η λύση στο πρόβλημα βέλτιστης δρομολόγησης δίνεται από βασικές κατηγορίες προσεγγιστικών μεθόδων, την Frank-Wolfe και Projection methods, τις οποίες εξετάζουμε και υλοποιούμε στην παρούσα εργασία. Χρησιμοποιώντας αληθοφανή σύγχρονα οπτικά δίκτυα και κίνηση παρουσιάζουμε αποτελέσματα σύγκλισης αυτών των αλγορίθμων. Η επίτευξη βέλτιστης δρομολόγησης, η επίδοση καθώς και η ομαλή λειτουργία των αλγορίθμων γίνεται εμπράκτως αντιληπτή μέσα από την ενσωμάτωση τους στο εργαλείο σχεδιασμού πολυεπίπεδων οπτικών δικτύων MANTIS.

Λέξεις Κλειδιά: αλγόριθμοι δρομολόγησης, βέλτιστη δρομολόγηση, βελτιστοποίηση, flow deviation, projection method, flow models, οπτικά δίκτυα, πολυεπίπεδα οπτικά δίκτυα

Abstract

The scope of this diploma thesis was the development of algorithms for optimal routing for optical networks at IP level and integrating them into designing multi-level optical networks tool MANTIS. Due to ARPANET's rapidly growing in 1970's, world's first packet switching network, it was essential to find an automated method in order to decide message's optimal routing and to design the optimal network topology. The Flow deviation method, developed by researchers for this cause, has been finally applied to many problems which can be modeled as network multicommodity flow problems.

The solution to the optimal routing problem came from the application of two categories of approximate methods, the Frank-Wolfe and Projection methods, both of which we examine and develop in this thesis. Using plausible modern optical networks and traffic we present the results of convergence of these algorithms. Achieving optimal routing, the performance and the smooth operation of these algorithms is perceived in practice by integrating them into the MANTIS designing multi-level optical networks tool .

Keywords: routing algorithms, optimal routing, optimization, flow deviation, projection method, flow models, optical networks, multi-level optical networks

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία εκπονήθηκε κατά το ακαδημαϊκό έτος 2016-2017 στα πλαίσια του τομέα Επικοινωνιών, Ηλεκτρονικής και Πληροφοριακών συστημάτων της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου.

Αρχικά, θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου κ.Εμμανουήλ Βαρβαρίγο για την εμπιστοσύνη που μου έδειξε με την ανάθεση της διπλωματικής, δίνοντάς μου την ευκαιρία να ασχοληθώ με ένα προσωπικώς ιδιαίτερα ενδιαφέρον θέμα.

Επίσης, οφείλω ένα μεγάλο ευχαριστώ στον κ. Κωνσταντίνο Χριστοδουλόπουλο για τη συνεχή καθοδήγηση, την υπομονή και την ανεκτίμητη βοήθεια που μου παρείχε κατά τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας.

Επιπλέον, θα ήθελα να εκφράσω τις ευχαριστίες μου στους φίλους και συναδέλφους για όλη την υποστήριξη στην φοιτητική μου σταδιοδρομία. Η βοήθειά σας σε κάθε επίπεδο ήταν πολύτιμη. Για τη συνεισφορά της θα ήθελα να ευχαριστήσω ειδικά την φίλη και συνεργάτιδα Ελένη Μάρκου.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου για την αμέριστη συμπαράσταση της καθόλη τη διάρκεια των σπουδών μου, καθώς και για την ευκαιρία να φοιτήσω σε αυτήν την σχολή.

Πίνακας περιεχομένων

1	Εισαγωγή.....	13
1.1	Σκοπός.....	13
1.2	Οργάνωση κειμένου.....	14
1.3	Η έννοια της δρομολόγησης.....	15
1.4	Δρομολόγηση και ARPANET.....	16
1.5	Εφαρμογές flow deviation method	18
1.5.1	Πλοήγηση οχημάτων σε οδικά δίκτυα - NAVOPT.....	18
1.5.2	Ελαχιστοποίηση ενεργειακού κόστους σε δίκτυο ISP.....	20
1.5.3	Μελλοντικές προεκτάσεις - SDN tunnel routing.....	21
1.6	Δρομολόγηση και Οπτικά δίκτυα.....	22
2	Βέλτιστη δρομολόγηση, μοντέλα ροής και τοπολογική σχεδίαση	26
2.1	Βέλτιστη δρομολόγηση βάσει μοντέλων ροής.....	26
2.2	Ορισμός του προβλήματος.....	27
2.2.1	Συναρτήσεις κόστους βέλτιστης δρομολόγησης.....	27
2.2.2	Διατύπωση του προβλήματος	28
3	Αλγόριθμοι βέλτιστης δρομολόγησης	32
3.1	General flow deviation.....	32
3.3.1	Feasible direction μέθοδοιγια βέλτιστη δρομολόγηση.....	32
3.2	The Frank-Wolfe (Flow Deviation) Method.....	34
3.3	Projection methods για βέλτιστη δρομολόγηση.....	37
3.3.1	Εφαρμογή για βέλτιστη δρομολόγηση.....	38
4	Υλοποίηση και πειράματα προσομοίωσης	41
4.1	Περιγραφή πειραματικής διαδικασίας και παραδοχές.....	41
4.2	Προσομοίωση 1 ^{ης} τοπολογίας δικτύου.....	42
4.2.1	Παράδειγμα 1 ^ο	42
4.2.2	Παράδειγμα 2 ^ο	44
4.3	Προσομοίωση 2 ^{ης} τοπολογίας δικτύου.....	47
4.3.1	Παράδειγμα 1 ^ο	46
4.3.2	Παράδειγμα 2 ^ο	48

4.4	Συγκρίσεις – Συμπεράσματα	49
4.4.1	Συμπεράσματα 1 ^{ης} κατηγορίας πειραμάτων	50
4.4.2	Συμπεράσματα 2 ^{ης} κατηγορίας πειραμάτων	50
5	Ενσωμάτωση αλγορίθμων στο MANTIS	53
5.1	MANTIS: Εργαλείο σχεδιασμού και λειτουργίας αποδοτικών οπτικών δικτυων	53
5.2	Προγραμματιστική διεπαφή εργαλείου και ενσωμάτωσης νέων αλγορίθμων στο επίπεδο εκτέλεσης	55
5.2.1	Επίπεδο εκτέλεσης και ενσωμάτωση νέων αλγορίθμων	55
5.2.2	Υλοποίηση και προσθήκη αλγορίθμων Optimal routing	57
5.3	Διεπαφή χρήστη με το εργαλείο MANTIS	59
5.4	Εκτέλεση σεναρίου αξιολόγησης	66
6	Βιβλιογραφία	69
ΠΑΡΑΡΤΗΜΑ Α		71
ΠΑΡΑΡΤΗΜΑ Β		82

1

Εισαγωγή

1.1 Σκοπός

Η παρούσα διπλωματική εξετάζει το πρόβλημα βέλτιστης δρομολόγησης δεδομένων, ανάθεσης δηλαδή ροών, σε ένα δίκτυο επικοινωνίας με στόχο την ελαχιστοποίηση του κόστους και/ή της καθυστέρησης για δεδομένη τοπολογία με δεδομένες απαιτήσεις - χαρακτηριστικά του δικτύου. Για το σκοπό αυτό, επικεντρωθήκαμε στην υλοποίηση της μεθόδου flow deviation (FD).

Ο αλγόριθμος FD προτάθηκε στις αρχές δημιουργίας του ARPANET για να συμβάλει στον αποτελεσματικό σχεδιασμό τοπολογιών δικτύων μεταγωγής πακέτου (packet switching). Το βασικό πλεονέκτημα του αλγορίθμου ήταν η απλότητα και η ικανότητα διαισθητικής ερμηνείας του σε κάθε στάδιο βελτιστοποίησης, στοιχείο που βοήθησε τους ερευνητές να αναπτύξουν τεχνικές σχεδιασμού άρρηκτα συνδεδεμένες με τις φυσικές ιδιότητες του συστήματος. Αυτή η απλότητα αλλά και η επιστημονική ακρίβεια έκανε τον αλγόριθμο αρκετά δημοφιλή σε πολλά διαφορετικά πεδία.

Ο αλγόριθμος FD είναι αποτελεσματικός σε στατικές ή αργά μεταβαλλόμενες ροές κίνησης. Μετά από τα πρώτα χρόνια λειτουργίας του ARPANET η τοπολογία μεγάλωσε και η κίνηση αυξήθηκε σημαντικά και ήταν πολύ εκρηκτική, οπότε η μέθοδος FD έπαψε να είναι αποδοτική για τον άμεσο έλεγχο της κίνησης του δικτύου. Ωστόσο, κατά τις ώρες αιχμής σε ένα οδικό δίκτυο, μπορεί να αποδειχθεί ιδιαίτερα βοηθητικός αφού η ταχύτητα των οχημάτων είναι πολύ πιο αργή από ότι η ανταλλαγή μηνυμάτων και η

υπολογιστική ικανότητα του αλγορίθμου. Ένα άλλο πεδίο όπου μπορεί να εφαρμοστεί με επιτυχία ο FD είναι η δρομολόγηση ροών δεδομένων σε δίκτυα καθοριζόμενα από λογισμικό (Software Defined Networks), όπου η μέθοδος FD μπορεί να προσφέρει νέες προοπτικές για τους σχεδιαστές στη δυναμική αναδιαμόρφωση των SDN tunnels.

Τέλος, ένα σημαντικό πεδίο εφαρμογής της μεθόδου FD αποτελούν τα οπτικά δίκτυα κορμού και τα μητροπολιτικά δίκτυα. Συνήθως αναφερόμαστε σε αυτά ως πολυεπίπεδα (multi-layer) οπτικά δίκτυα, όταν λαμβάνουμε υπόψη μας και παίρνουμε αποφάσεις για τη ρύθμιση των IP δρομολογητών (routers) στις παρυφές του οπτικού δικτύου. Η κίνηση στα δίκτυα αυτά δεν είναι πολύ δυναμική, αφού αποτελεί συνάθροιση κίνησης από πολλά δίκτυα πρόσβασης. Η εφαρμογή σε αυτά τα δίκτυα οδήγησε την παρούσα εργασία στη μελέτη και υλοποίηση αλγορίθμων που βασίζονται στη μέθοδο FD και την ενσωμάτωση τους στο εργαλείο σχεδιασμού οπτικών δικτύων MANTIS.

1.2 Οργάνωση κειμένου

Στο **1^ο κεφάλαιο** παρουσιάζεται η έννοια της δρομολόγησης ως ένα γενικότερο πρόβλημα βελτιστοποίησης. Παράλληλα, γίνεται μία σύντομη αναδρομή στους αλγόριθμους δρομολόγησης που συνέβαλαν στην ανάπτυξη και διάδοση του ARPANET επικεντρώνοντας το ενδιαφέρον μας στη μέθοδο βέλτιστης δρομολόγησης flow deviation (FD) method. Οι εφαρμογές όμως της μεθόδου αυτής δεν περιορίζονται μόνο σε δίκτυα ηλεκτρονικών υπολογιστών αλλά και σε άλλα επιστημονικά προβλήματα. Η πλοήγηση οχημάτων σε οδικά δίκτυα με την χρήση της πλατφόρμας του NAVOPT, η ελαχιστοποίηση του ενεργειακού κόστους ISP δικτύων αλλά και η μελλοντική εφαρμογή SDN tunnels αποτελούν μόνο μερικές από τις ήδη υπάρχουσες εφαρμογές της μεθόδου. Το κεφάλαιο κλείνει με μια σύντομη παρουσίαση των πολυεπίπεδων οπτικών δικτύων στα οποία η μέθοδος FD έχει σημαντική εφαρμογή.

Στο **2^ο κεφάλαιο** της διπλωματικής εργασίας γίνεται προσπάθεια μαθηματικής περιγραφής του προβλήματος βελτιστοποίησης, ορίζοντας τις απαιτούμενες μεταβλητές και τις παραμέτρους που πρέπει να ικανοποιούνται. Η βέλτιστη δρομολόγηση που επιχειρείται βασίζεται σε «μοντέλα ροής» της προσέγγισης Kleinrock (Kleinrock independence approximation).

Η λύση στο πρόβλημα της δρομολόγησης των δεδομένων δίνεται στο επόμενο και **3^ο κεφάλαιο** της εργασίας, όπου εξετάζουμε και αναλύουμε δύο βασικές προσεγγιστικές μεθόδους, την Frank Wolfe Method και την Projection Method.

Στη συνέχεια, υλοποιούμε τους δύο αυτούς αλγορίθμους, εφαρμόζοντας όσα προηγήθηκαν σε δύο διαφορετικές τοπολογίες δικτύων δεδομένων.

Παρουσιάζουμε τα αποτελέσματα αυτά στο **4^ο κεφάλαιο**, όπου διατυπώνονται και οι βασικές μας παρατηρήσεις όσον αφορά τη λειτουργία, την καταλληλότητα και την ακρίβεια των μεθόδων στα εκάστοτε πειράματα. Οι προσομοιώσεις πραγματοποιήθηκαν σε προγραμματιστικό περιβάλλον matlab.

Τέλος, στο **5^ο κεφάλαιο** παρουσιάζουμε την διαδικασία ενσωμάτωσης των δύο αλγορίθμων στο εργαλείο σχεδιασμού πολυεπίπεδων οπτικών δικτύων MANTIS και δίνουμε κάποια στιγμιότυπα της λειτουργία τους.

Στο **παράρτημα Α** της εργασίας διατίθενται οι κώδικες υλοποίησης των αλγορίθμων που χρησιμοποιήθηκαν για τις προσομοιώσεις που εκτελέστηκαν στα παραδείγματα του 4^{ου} κεφαλαίου, ενώ στο **παράρτημα Β** τα δεδομένα εισόδων τους για κάθε δίκτυο.

1.3 Η έννοια της δρομολόγησης

Στην ενότητα αυτή επιχειρείται μια πρώτη εισαγωγή στην έννοια της δρομολόγησης και εξηγείται γιατί η αντιμετώπιση του προβλήματος της δρομολόγησης εμπίπτει στην αντιμετώπισή του ως συνολικό πρόβλημα βελτιστοποίησης.

Ο όρος της δρομολόγησης αναφέρεται στη γενικότητά του στην κατάστρωση και σχεδίαση της διαδρομής πάνω σε ένα γράφο, του οποίου οι ακμές ,που συνδέουν ζεύγη κόμβων, χαρακτηρίζονται από κόστη. Στη γενική περίπτωση, τα κόστη αυτά μπορεί να υπολογίζονται ως συνάρτηση της απόστασης, του εύρους ζώνης, της μέσης κίνησης, του κόστους επικοινωνίας, του μέσου μήκους της ουράς αναμονής, της μετρούμενης καθυστέρησης και άλλων παραγόντων. Σκοπός, επομένως, ενός αλγορίθμου «βέλτιστης» δρομολόγησης αποτελεί η εύρεση του «συντομότερου» μονοπατιού (shortest path) του γράφου σύμφωνα με οποιοδήποτε μεμονωμένο κριτήριο ή σύμφωνα με ένα συνδυασμό κριτηρίων, δηλαδή η εύρεση εκείνης της ακολουθίας κόμβων που ελαχιστοποιούν το κόστος μετάβασης από έναν κόμβο πηγής σε ένα κόμβο προορισμού.

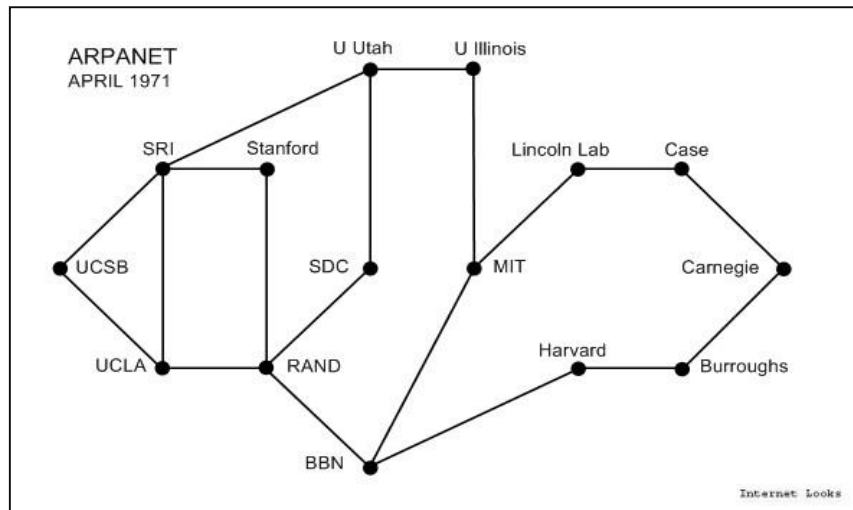
Πράγματι, οι αλγόριθμοι και οι τεχνικές που χρησιμοποιούνται για τη εύρεση ή προσέγγιση της βέλτιστης διαδρομής απαντώνται τακτικά σε προβλήματα Συνδυαστικής Βελτιστοποίησης, Στοχαστικής Βελτιστοποίησης και Βέλτιστου Ελέγχου και αποτελούν ένα NP πλήρες πρόβλημα. Οι νέες αλγοριθμικές εξελίξεις των τελευταίων ετών και η ολοένα αυξανόμενη υπολογιστική δύναμη των συστημάτων οδήγησαν στην δραματική αύξηση των δραστηριοτήτων της βελτιστοποίησης.

1.4 Δρομολόγηση και ARPANET

Το ARPANET (Advanced Research Projects Agency Network) ήταν το πρώτο στον κόσμο δίκτυο μεταγωγής πακέτου και το δίκτυο πυρήνας ενός συνόλου που θα συνέθετε το παγκόσμιο Διαδίκτυο (internet). Το δίκτυο χρηματοδοτήθηκε από το Γραφείο ερευνών Αμύνης (Defense Advanced Research Projects Agency (DARPA)) του τμήματος άμυνας των Ηνωμένων Πολιτειών για χρήση στα πανεπιστήμια και εργαστήρια ερευνών στις Η.Π.Α. Η μεταγωγή πακέτου (Packet switching), η οποία αποτελεί σήμερα την κυρίαρχη βάση για την επικοινωνία δεδομένων παγκοσμίως, ήταν μια νέα αντίληψη την στιγμή της σύλληψης της δημιουργίας του ARPANET. Οι επικοινωνίες δεδομένων είχαν βασιστεί στη μεταγωγή κυκλώματος, όπως στο παραδοσιακό τηλεφωνικό δίκτυο.

Το 1971 το ARPANET αναπτυσσόταν ταχύτατα. Κατά συνέπεια, προέκυψε η ανάγκη να βρεθεί μία αυτοματοποιημένη μέθοδος που θα αποφασίζει για την βέλτιστη διάταξη και τοπολογία του δικτύου. Συγχρόνως, η θεωρία δικτύων πρωταγωνιστούσε στην έρευνα, κυρίως σε ό,τι αφορούσε τα «Στοχαστικά συστήματα αναμονής» (stochastic networks of queues) και την «βελτιστοποίηση ανάθεσης ροών» (network flow optimization) σε δίκτυο.

Μετά την ένταξή του στο UCLA, ο Kleinrock μαζί με δύο άλλους νέους ερευνητές, τους Luigi Fratta και Mario Gerla, διατύπωσε δύο βασικά προβλήματα για τον σχεδιασμό ενός δικτύου μεταγωγής πακέτου, όπως ήταν το ARPANET. Αυτά τα προβλήματα αφορούσαν την δρομολόγηση του μηνύματος στο δίκτυο και την ανάθεση χωρητικότητας στο κανάλι. Η μέθοδος που προτάθηκε για την επίλυση αυτών ήταν η flow deviation method (FD), η διαδικασία δηλαδή κατά την οποία αναθέτουμε ροές σε ένα δίκτυο επικοινωνίας προκειμένου να ελαχιστοποιήσουμε το κόστος ή/και την καθυστέρηση για μία δεδομένη τοπολογία και για δεδομένες απαιτήσεις του δικτύου. Λόγω του ότι η μέθοδος αυτή μπορεί να εφαρμοστεί σε κάθε πρόβλημα ροής, με περιορισμούς ή χωρίς, αποτέλεσε ένα ικανοποιητικό και αποτελεσματικό εργαλείο για τον σχεδιασμό δικτύων μεταγωγής πακέτου και ειδικά για την ανάπτυξη του ARPANET στις αρχές του 1970. Επίσης, ο απαιτούμενος χρόνος υπολογισμού ανά επανάληψη ήταν συγκρίσιμος με το χρόνο των αντίστοιχων τεχνικών που εφαρμοζόντουσαν εκείνη την περίοδο.



1.1. Τοπολογία ARPANET 1971

Η μέθοδος FD επηρέασε πολλές πτυχές του σχεδιασμού των πρωτοκόλλων του ARPANET. Η πρώτη πρόκληση ήταν ο σχεδιασμός ενός κατάλληλου αλγορίθμου δρομολόγησης για το ARPANET, καθώς ο ήδη υπάρχων και χρησιμοποιούμενος αλγόριθμος Bellman-Ford και οι διάφορες επεκτάσεις του δεν επαρκούσαν. Ο αλγόριθμος FD υποσχέθηκε μια βελτιωμένη λύση με βάση τη συστηματική επιλογή βελτίωσης πολλαπλών μονοπατιών. Ωστόσο, η μέθοδος FD σχεδιάστηκε για ημιστατικά δίκτυα και έτσι δεν μπορούσε να εφαρμοστεί άμεσα στο ARPANET όπου η κίνηση είναι πολύ δυναμική. Ακόμη, ο αλγόριθμος χρησιμοποιήθηκε κυρίως στην διαχείριση της τοπολογίας του ARPANET, δηλαδή προκειμένου να επιβεβαιώσει ότι οι νέες προσθήκες συνδέσμων στο δίκτυο θα παρέχουν τη βέλτιστη δυνατή λύση.

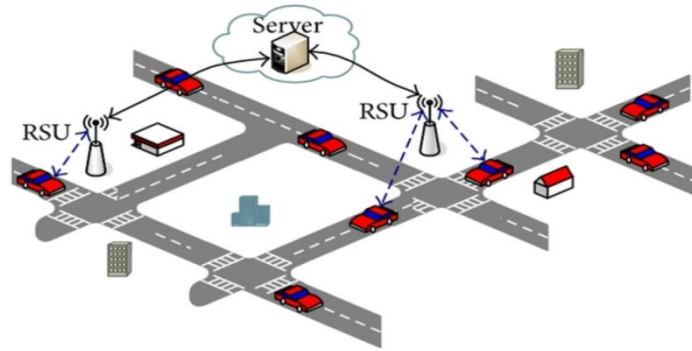
Η πρόκληση της ενσωμάτωσης της συνεχούς βελτίωσης των ροών στα πρωτόκολλα του ARPANET ελήφθη από τον Callagher. Το 1977 ο Callagher σχεδίασε μία πολύ κομψή μέθοδο που παρατηρούσε τον ρυθμό μετάδοσης των δεδομένων και τις καθυστερήσεις σε πραγματικό χρόνο βελτιστοποιώντας ακολουθώντας τους πίνακες δρομολόγησης. Η μέθοδος του υλοποιήθηκε και επεκτάθηκε με την συμβολή του Golestani. Οι δύο ερευνητές χρησιμοποίησαν τεχνικές FD επιδιώκοντας να βελτιστοποιήσουν μια κυρτή αντικειμενική συνάρτηση. Ωστόσο, παρά την εννοιολογική κομψότητα της προσέγγισης τους, η μέθοδος Gallager δεν βρήκε σημαντική έγκριση και εφαρμογή στα πρωτόκολλα δρομολόγησης λόγω της δυναμικότητας των ροών του ARPANET, της σχετικής πολυπλοκότητας του συστήματος και των περιορισμένων υπολογιστικών πόρων των δρομολογητών εκείνες τις μέρες. Θα χρειαστούν αρκετά χρόνια πριν οι FD τεχνικές ροής χρησιμοποιηθούν για την on-line διαχείριση της κυκλοφορίας.

1.5 Εφαρμογές flow deviation method

Μετά τις εφαρμογές στο ARPANET, το ενδιαφέρον για τον αλγόριθμο FD παρέμεινε στην δικτυακή ερευνητική κοινότητα και ενέπνευσε πολλούς ερευνητές να αναπτύξουν παρόμοιες μεθόδους για ένα ευρύ φάσμα προβλημάτων που δεν αφορούσαν μόνο δίκτυα μεταγωγής πακέτου. Στην πραγματικότητα, καθώς η τεχνολογία των δικτύων εξελισσόταν, οι ερευνητές αντιμετώπισαν νέα «προβλήματα ροών» τα οποία θεώρησαν βολικό να τα κατατάξουν ως τέτοια καθώς και να τα επιλύσουν με χρήση αλγορίθμου FD, εκμεταλλευόμενοι την απλότητα, την γενικότητα και την ευελιξία χρήσης του. Αξίζει να σημειώσουμε ότι μέχρι σήμερα περισσότερες από 500 επιστημονικές δημοσιεύσεις έχουν αναφορές στην αρχική δημοσίευση του FD, γεγονός που καταδεικνύει το εύρος εφαρμογών του αλγορίθμου. Στην ενότητα που ακολουθεί παρουσιάζουμε εφαρμογές της μεθόδου FD οι οποίες δεν σχετίζονται με δρομολόγηση πακέτων δεδομένων.

1.5.1 Πλοήγηση οχημάτων σε οδικά δίκτυα - NAVOPT

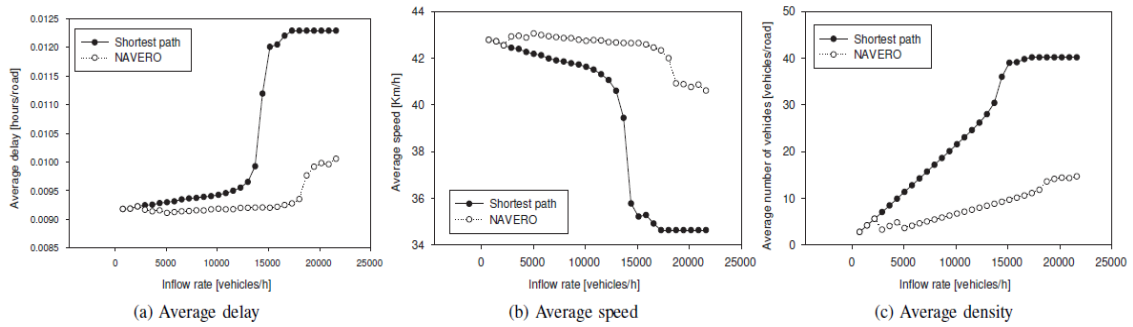
Ο αριθμός των οχημάτων στις περισσότερες αστικές περιοχές έχει ξεπεράσει την οδική χωρητικότητα, οδηγώντας σε σοβαρά προβλήματα κυκλοφοριακής συμφόρησης κατά τις ώρες αιχμής. Οι πληροφορίες για την κυκλοφορία που λαμβάνουμε από τα οχήματα σε πραγματικό χρόνο μπορούν να βοηθήσουν στη μείωση της κυκλοφοριακής συμφόρησης, αξιοποιώντας την ασύρματη αρχιτεκτονική DSRC (Dedicated short-range communications) που έχει ως στόχο την ασφάλεια και τον έλεγχο της κυκλοφορίας. Για το σκοπό αυτό, ερευνητές του UCLA υλοποίησαν το NAVOPT (Navigator Assisted Vehicular route OPTimizer), έναν μηχανισμό πλοήγησης και δρομολόγησης κίνησης οχημάτων. Στο NAVOPT, υποθέτουμε ότι όλα τα οχήματα είναι εφοδιασμένα με ένα έξυπνο σύστημα πλοήγησης το οποίο υποστηρίζει ασύρματη επικοινωνία (Wi-Fi ή 3G) και χάρτη με γεωγραφικές πληροφορίες. Το σύστημα πλοήγησης διαθέτει επίσης GPS (Global Positioning System) ή DSRC διεπαφές. Στην παρούσα αρχιτεκτονική, τα οχήματα μπορούν να επικοινωνούν μεταξύ τους καθώς με διακομιστές στο Διαδίκτυο (servers) μέσω ειδικών σημείων πρόσβασης, τα RSU's (Road side units).



1.3. Αρχιτεκτονική NAVOPT

Το NAVOPT επιδιώκει να αντιμετωπίσει το πρόβλημα σε συνολικό επίπεδο. Επομένως, η πόλη υποδιαιρείται σε διαμερίσματα. Κάθε γεωγραφικό διαμέρισμα έχει έναν διακομιστή πλοήγησης (server) που παρέχει τη βέλτιστη ταξιδιωτική διαδρομή για κάθε οδηγό. Επίσης, δίκτυα οχημάτων μεγάλης κλίμακας όπως εθνικά δίκτυα μπορούν να υποδιαιρεθούν σε περισσότερα δίκτυα μικρότερης κλίμακας (π.χ. δίκτυα πόλεων) σε μια ιεραρχική δομή που διατηρεί την επεκτασιμότητα του συστήματος (scalability). Τα οχήματα συγκεντρώνουν πληροφορίες κυκλοφορίας μέσω αισθητήρων, κάμερας και GPS, όπως η πυκνότητα των οχημάτων, η ταχύτητα ή η θέση κάποιου ατυχήματος, με τις οποίες ενημερώνουν τον server ανά τακτά χρονικά διαστήματα. Βασιζόμενος σε αυτές τις πληροφορίες, ο server-πλοηγός μπορεί να εκτιμήσει τον χρόνο ταξιδιού συναρτήσει του μέσου αριθμού οχημάτων στο δρόμο και της καθυστέρησης σε κάθε τμήμα της διαδρομής. Στην καθυστέρηση μπορεί επίσης να συνυπολογιστεί η οδική χωρητικότητα βάσει πειραματικών μαθηματικών μοντέλων. Ωστόσο, κάποιοι αστάθμητοι παράγοντες, όπως ατυχήματα, μπορεί να διαταράξουν αυτή τη σχέση. Στη συνέχεια, ο server επιστρέφει στον πλοηγό του οχήματος τη συντομότερη διαδρομή (minimum cost path), αυτή δηλαδή με τον ελάχιστο απαιτούμενο χρόνο ταξιδιού σύμφωνα με τις τρέχουσες κυκλοφοριακές συνθήκες. Στο NAVOPT, ο server χρησιμοποιεί τον αλγόριθμο FD (flow deviation method) προκειμένου να υπολογίσει την βέλτιστη ταξιδιωτική διαδρομή η οποία θα ισορροπήσει την συμφόρηση του δικτύου.

Παρακάτω παρουσιάζουμε κάποια αποτελέσματα του NAVOPT τα οποία προέκυψαν μέσω του προσομοιωτή SUMO με χρήση της FD μεθόδου που εξετάζουμε σε δίκτυο του Μανχάταν.



1.4. Αποτελέσματα απόδοσης FD μέσω προσομοίωσης SUMO

Συμπερασματικά, μπορούμε να υποστηρίξουμε ότι ο αλγόριθμος FD αποδείχθηκε αρκετά αποτελεσματικός στην εύρεση εναλλακτικών διαδρομών για τα οχήματα του δικτύου. Το NAVOPT μπορεί να βελτιώσει την ταχύτητα και την απόδοση κατά περίπου 25% σε σύγκριση με συντομότερη διαδρομή δρομολόγησης της «εγωιστικής» αντιμετώπισης του προβλήματος (τη συντομότερη δηλαδή διαδρομή που υπολογίζει κάθε μεμονωμένος οδηγός). Επίσης, το NAVOPT μειώνει το συνολικό χρόνο ταξιδιού κατά 40%, γεγονός που αποδεικνύει ότι η μέθοδος FD αποτελεί μία έγκυρη μέθοδο δρομολόγησης οχημάτων. Στην πραγματικότητα, η μέθοδος FD λειτουργεί καλύτερα με ροές οχημάτων από ότι με ροές πακέτων δεδομένων του ARPANET, αφού οι ταχύτητες των οχημάτων και γενικότερα οι πάσης φύσεως μεταβολές του συστήματος είναι πολύ πιο αργές από αυτές των δικτύων ηλεκτρονικών υπολογιστών και κατά συνέπεια, πολύ πιο εύκολα διαχειρίσιμες από τον αλγόριθμο.

1.5.2 Ελαχιστοποίηση κόστους ενέργειας σε δίκτυο ISP

Το διαρκώς αυξανόμενο κόστος της ενέργειας και η ανάγκη μείωση της παγκόσμιας εκπομπής αερίων του θερμοκηπίου (όπως το CO₂) για την προστασία του περιβάλλοντος, έχουν αναδείξει την ενεργειακή απόδοση σε μία από τις κυριότερες τεχνολογικές προκλήσεις του αιώνα μας. Αξίζει να σημειώσουμε ότι, σύμφωνα με μελέτες, η κατανάλωση ενέργειας που σχετίζεται με το Διαδίκτυο ξεπερνά το 10% της παγκόσμιας κατανάλωσης ενέργειας και αυξάνεται με αλματώδη ταχύτητα. Στο πλαίσιο αυτό, οι τεχνολογίες επικοινωνιών και πληροφοριών ICT (Information and Communication Technologies) αναμένεται να παίξουν καθοριστικό ρόλο στην μείωση των ενεργειακών απαιτήσεων μέσω της βελτιστοποίησης των ενεργειακών διαδικασιών. Δεν αποτελεί έκπληξη ότι μόνο το 20% των εκπομπών του διοξειδίου του άνθρακα των ICT προέρχεται από την παραγωγή, ενώ το υπόλοιπο 80% από τη χρήση του εξοπλισμού. Επιπλέον, το 37% των συνολικών εκπομπών των ICT οφείλονται στις τηλεπικοινωνιακές υποδομές και στις

συσκευές, ενώ τα κέντρα δεδομένων και τα τερματικά των χρηστών είναι υπεύθυνα για το υπόλοιπο ποσοστό. Στην Ιταλία, για παράδειγμα, Η Telecom Italia είναι ο δεύτερος μεγαλύτερος καταναλωτής ηλεκτρικής ενέργειας μετά το σύστημα Εθνικών Σιδηροδρόμων, και καταναλώνει περισσότερο από 2 TWh ανά έτος, η οποία είναι ισοδύναμη με την ενέργεια που καταναλώνεται από 660.000 οικογένειες μέσα σε ένα χρόνο. Παρόμοιες, και ακόμα πιο απαισιόδοξες σκέψεις ισχύουν και για άλλες ανεπτυγμένες χώρες, όπως για παράδειγμα η Ιαπωνία. Στο βαθμό αυτό, συσκευές δικτύωσης, όπως δρομολογητές IP (IP routers) καταναλώνουν το μεγαλύτερο μέρος της ενέργειας. Επομένως, δεν αποτελεί έκπληξη το γεγονός ότι ερευνητές, κατασκευαστές και οι πάροχοι των δικτύων διενεργούν σημαντικές προσπάθειες για τη μείωση της κατανάλωσης ενέργειας των συστημάτων ICT.

Η κεντρική ιδέα, επομένως, πολλών δημοσιεύσεων στο συγκεκριμένο θέμα αφορά τη μείωση της συνολικής ισχύος κατανάλωσης ενός Internet service provider (ISP) στη βάση ενός δικτύου, θεωρώντας το ως ένα απλό, μεγάλο και διανεμημένο σύστημα. Δεν μας ενδιαφέρει η μείωση της κατανάλωσης ισχύος κάθε συσκευής ξεχωριστά, αλλά στοχεύουμε στον έλεγχο ολόκληρου του δικτύου, έτσι ώστε να επιτύχουμε τον ελάχιστο αριθμό χρησιμοποιούμενων συσκευών που απαιτούνται για την κάλυψη της πραγματικής ζήτησης της κυκλοφορίας. Αυτό επιτυγχάνεται με τη διαχείριση των πόρων που απαιτούνται για την ικανοποίηση της ποιότητας της παρεχόμενης υπηρεσίας QoS (quality-of-service) κατά τις ώρες αιχμής. Για το σκοπό, έχει προταθεί η ενεργοποίηση και απενεργοποίηση συσκευών του δικτύου για την εξοικονόμηση ενέργειας, διασφαλίζοντας ταυτόχρονα τις απαιτήσεις της QoS. Αυτή η βελτιστοποίηση μπορεί να οριστεί επακριβώς από ακέραιο γραμμικό προγραμματισμό (ILP) σε διάφορα επίπεδα πολυπλοκότητας. Σε αυτό το πλαίσιο επομένως, όπου εμπλέκονται μόνο πραγματικές μεταβλητές, ο αλγόριθμος FD μπορεί εύκολα να δώσει τη βέλτιστη λύση στην δρομολόγηση της συγκεντρωτικής ροής.

1.5.3 Μελλοντικές προεκτάσεις – SDN tunnel routing

Όπως έχουμε ήδη αναφέρει, σε «παραδοσιακά» δίκτυα μεταγωγής πακέτου η κίνηση μπορεί να είναι ιδιαίτερος εκρηκτική και η ανάθεση ροών να πρέπει να πραγματοποιείται σε millisecond. Αυτός άλλωστε είναι και ένας από τους λόγους για τους οποίους το σχήμα δρομολόγησης που αναπτύχθηκε από τον Gallager στα μέσα του 1970 και που βασίστηκε στην μέθοδο FD δεν βρήκε άμεση πρακτική εφαρμογή.

Πρόσφατα, ωστόσο, μία νέα τεχνολογία στο τομέα των δικτύων έχει αρχίσει να εμφανίζεται, το Software Defined Network (SDN), η οποία βασίζεται σε μοντέλα ροής όπου η μονάδα δεν είναι το πακέτο, όπως στα «παραδοσιακά»

δίκτυα, αλλά η ροή των πακέτων ενός ζεύγους πηγής-προορισμού. Στα SDN, το πρόβλημα δρομολόγησης περιλαμβάνει όχι μόνο το πρόβλημα δρομολόγησης μεμονωμένων πακέτων πληροφορίας αλλά ολόκληρων διόδων (tunnels). Οι ροές στα tunnels είναι αρκετά σταθερές στον χρόνο. Μικρές αλλαγές στην κίνηση είναι διαχειρίσιμες και πραγματοποιούνται με αναδιανομή της κίνησης πηγής-προορισμού πάνω από τα tunnels. Ενώ, μεγαλύτερες αλλαγές απαιτούν την εγκατάσταση νέων tunnels. Σε κάθε περίπτωση χρησιμοποιείται η προσέγγιση της FD μεθόδου, προκειμένου να «μεταφερθεί» ροή από ένα tunnel σε ένα άλλο, το «shortest path» tunnel. Επομένως, λόγω της ιδιαιτέρως υψηλής κίνησης στα σύγχρονα data centers , όπως Google, Amazon και Facebook, οι λύσεις που προσφέρει η τεχνολογία SDN έχουν αρχίσει να γίνονται πολύ δημοφιλείς.

1.6 Δρομολόγηση και Οπτικά δίκτυα

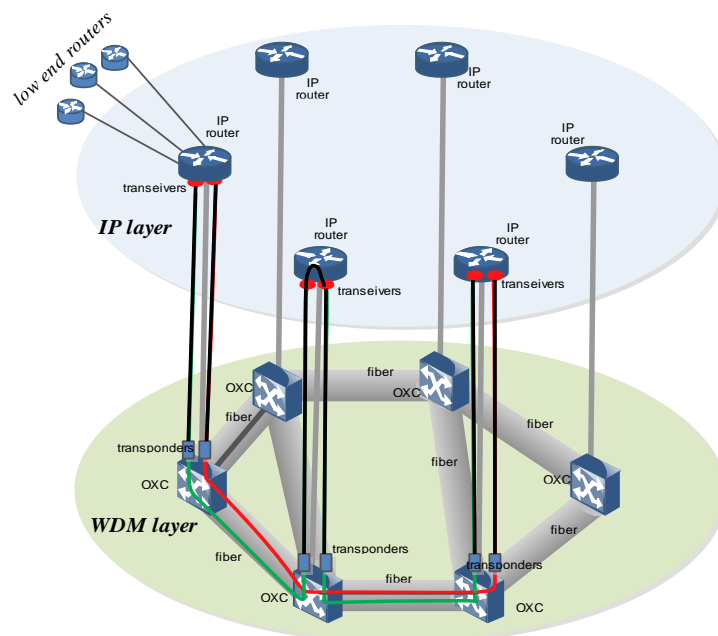
Όπως είδαμε παραπάνω η μέθοδος FD υλοποιήθηκε αρχικά για το ARPANET, αλλά μετά από τα πρώτα χρόνια λειτουργία του ARPANET η τοπολογία μεγάλωσε και η κίνηση αυξήθηκε και ήταν πολύ εκρηκτική, οπότε η μέθοδος FD έπαψε να είναι αποδοτική για τον άμεσο έλεγχο της κίνησης του δικτύου αυτού. Οι ερευνητές χρησιμοποίησαν τη μέθοδο FD και σε άλλα προβλήματα δρομολόγησης και βελτιστοποίησης ροών, όπου παρουσιάζουν στατική ή αργά μεταβαλλόμενη κίνηση.

Επιστρέφοντας στα δίκτυα τηλεπικοινωνιών, ένα πεδίο που η μέθοδος FD έχει ακόμα και σήμερα εφαρμογή είναι τα οπτικά δίκτυα κορμού και τα μητροπολιτικά οπτικά δίκτυα. Το δίκτυο ARPANET εξελίχθηκε στο διαδίκτυο που συνενώνει πολλά μικρότερα δίκτυα. Το διαδίκτυο είναι ιεραρχικό και αποτελείται από διάφορα τμήματα, τα δίκτυα πρόσβασης, τα μητροπολιτικά δίκτυα και τα δίκτυα κορμού. Η κίνηση κυρίως στα δίκτυα κορμού αλλά και στα μητροπολιτικά δίκτυα είναι αρκετά σταθερή αφού αποτελεί συνάθροιση κίνησης από πολλά μικρότερα δίκτυα πρόσβασης. Σήμερα τα δίκτυων κορμού και μητροπολιτικών δικτύων υλοποιούνται με οπτικά δίκτυα που χρησιμοποιούν πολυπλεξία μήκους κύματος (Wavelength Division Multiplexing –WDM). Σε αυτήν την ενότητα, εξηγείται η έννοια της δρομολόγησης σε ένα WDM οπτικό δίκτυο καθώς και ο τρόπος με τον οποίο μπορεί να εφαρμοστεί η μέθοδος FD σε αυτό.

Η τεράστια χωρητικότητα των οπτικών ινών μπορεί να αξιοποιηθεί μέσω της τεχνολογίας WDM με μετάδοση πολλαπλών σημάτων σε μόνο μία ίνα χρησιμοποιώντας διαφορετικά μήκη κύματος. Κάθε κανάλι (μήκος κύματος) λειτουργεί σε πολύ υψηλή ταχύτητα μετάδοσης της τάξης των 1-10 Gbps. Πολλά εμπορικά συστήματα που έχουν ήδη αναπτυχθεί , επιτυγχάνουν 32-40

διαφορετικά μήκη κύματος ανά ίνα, ενώ συστήματα που υποστηρίζουν μέχρι και 100 μήκη κύματος βρίσκονται υπό σχεδιασμό.

Ένα οπτικό δίκτυο δρομολόγησης μήκους κύματος, όπως αυτό που παρουσιάζεται στο σχήμα, αποτελείται από οπτικούς κόμβους μεταγωγής (optical cross connect –OXC ή reconfigurable add/drop multiplexers ROADM). Οι οπτικοί κόμβοι OXC /ROADM εκτελούν τις λειτουργίες της προσθαφαίρεσης κυκλωμάτων (add-drop) αλλά και της διαφανής (transparent) μεταγωγής των διερχόμενων οπτικών κυκλωμάτων, όπου με τον όρο διαφανή εννοούμε ότι το σήμα παραμένει οπτικό. Οι οπτικοί κόμβοι διασυνδέονται μέσω οπτικών ινών. Οπτικοί πομποδέκτες (transponders) χρησιμοποιούνται για την μετάδοση του σήματος στο οπτικό δίκτυο και οπτικοί ενισχυτές χρησιμοποιούνται στις ίνες/συνδέσμους (ενισχυτές γραμμής), αλλά και στους οπτικούς κόμβους για να αντισταθμίζουν τις απώλειες διάδοσης των οπτικών ινών, όπου αυτό κρίνεται απαραίτητο.



1.2. Παράδειγμα οπτικού δικτύου τεχνολογίας WDM

Τα WDM δίκτυα συναντώνται στον κορμό (core/backbone) και σε μητροπολιτικά δίκτυα (metro), με σκοπό να συνενώσουν IP υπο-δίκτυα στις παρυφές τους που είναι γεωγραφικά κατανομημένα και σε μεγάλες αποστάσεις μεταξύ τους. Στην ουσία, ηλεκτρονικοί IP δρομολογητές (routers) ενώνονται με τους οπτικούς κόμβους (ROADM/OXC) και επικοινωνούν μέσω του οπτικού WDM δικτύου που δημιουργούν οι οπτικοί κόμβοι και οι ίνες. Οπτικοί πομποδέκτες transponders μετατρέπουν τα πακέτα που προωθούνται από τους IP routers σε οπτικό σήμα το οποίο μεταφέρετε μέσω οπτικών κυκλωμάτων πάνω από το WDM δίκτυο στον IP router προορισμού. Τα οπτικά πακέτα, μπορεί να μετατραπούν σε ηλεκτρικά σε κάποιους ενδιάμεσους οπτικούς κόμβους, απαιτώντας την χρήση επιπλέον πομποδεκτών σε αυτούς τους

κόμβους ή τα πακέτα μπορούν να περάσουν διαφανώς (transparent) κάποιους ενδιάμεσους κόμβους (κόκκινη σύνδεση).

Στο παρελθόν η WDM τεχνολογία χρησιμοποιείτο μόνο για μεταδόσεις από-σημείο-σε-σημείο, που σημαίνει ότι όλες οι συνδέσεις τερματίζονταν σε κάθε ενδιάμεσο κόμβο στο ηλεκτρονικό επίπεδο, χωρίς να υπάρχει η δυνατότητα να περνάνε διαφανώς από τον ενδιάμεσο κόμβο. Στα σημερινά οπτικά δίκτυα (δεύτερης και τρίτης γενιάς), το αν θα τερματίζει ή όχι το σήμα ηλεκτρονικά και σε ποιους ενδιάμεσους κόμβους αποτελεί πλέον μια από τις επιλογές του σχεδιαστή του δικτύου. Αυτές οι επιλογές συνεισφέρουν άμεσα ή έμμεσα στο κόστος και την ενεργειακή κατανάλωση του δικτύου: μια διαφανή σύνδεση δεν χρησιμοποιεί πομποδέκτες και μειώνει το φόρτο του ηλεκτρονικού IP router. Από την άλλη υπάρχει περιορισμός λόγω των φυσικών εξασθενήσεων (θόρυβος από ενισχυτές, παρεμβολή από άλλα κανάλια) της διαφανής απόστασης μετάδοσης, οπότε οπτικοί αναγεννητές ή τερματισμός στο ηλεκτρικό επίπεδο χρησιμοποιείται στις πολύ μακρινές οπτικές συνδέσεις. Όταν εξετάζουμε το οπτικό δίκτυο μεταγωγής μήκους κύματος και τους IP δρομολογητές στις παρυφές του, έχουμε ένα πολυεπίπεδο (multi-layer) δίκτυο το οποίο στη βιβλιογραφία ονομάζεται ως IP over WDM.

Στα IP over WDM δίκτυα υπάρχουν 2 προβλήματα δρομολόγησης και για λόγους απλούστευσης λύνονται ξεχωριστά. Υπάρχει η δρομολόγηση στο οπτικό WDM δίκτυο, το οποίο συνήθως ονομάζεται και πρόβλημα δρομολόγησης και ανάθεσης μήκους κύματος (routing and wavelength assignment). Το πρόβλημα της εγκατάστασης μιας σύνδεσης, δηλαδή της εύρεσης ενός οπτικού μονοπατιού (lightpath), ανάμεσα σε έναν αρχικό και έναν τελικό κόμβο ανάγεται σε δύο μέρη: στον προσδιορισμό της διαδρομής (route) και στον προσδιορισμό του μήκους κύματος (wavelength assignment). Οι περιορισμοί που ορίζουν αυτό το πρόβλημα είναι ότι ένα μήκος κύματος μπορεί να χρησιμοποιηθεί μόνο από ένα οπτικό μονοπάτι σε κάθε οπτικό σύνδεσμο (distinct wavelength assignment constraint) και ότι, όταν δεν υπάρχουν μετατροπείς μήκους κύματος, το οπτικό μονοπάτι πρέπει να χρησιμοποιεί το ίδιο μήκος κύματος σε όλους τους συνδέσμους που το αποτελούν (wavelength continuity constraint). Το RWA πρόβλημα έχει εκθετική πολυπλοκότητα (NP-hard) και μοντελοποιείται συνήθως σαν ένα πρόβλημα ακέραιου γραμμικού προγραμματισμού (integer linear programming - ILP). Επειδή οι μεταβλητές του προβλήματος είναι ακέραιες (μήκη κύματος και σύνδεσμοι που αποτελούν το μονοπάτι), η μέθοδος FD δεν έχει άμεση εφαρμογή σε αυτό το πρόβλημα. Όμως ένας τρόπος επίλυσης προβλημάτων ακέραιου γραμμικού προγραμματισμού χρησιμοποιούν την τεχνική του γραμμικού προγραμματισμού (linear programming - LP) και πιο συγκεκριμένα χρησιμοποιούν την LP χαλάρωση (LP-relaxation) όπου οι μεταβλητές γίνονται συνεχείς και μετά εφαρμόζονται τεχνικές για να πάρουμε

ακέραιες λύσεις .Οπότε η FD μέθοδος εφαρμόστηκε για την επίλυση του RWA προβλήματος σε αλγόριθμους που βασίζονται σε LP χαλάρωση.

Τα οπτικά μονοπάτια ορίζουν κατ'ουσίαν μια εικονική τοπολογία (virtual topology) πάνω στη οποία εκτελείται η IP δρομολόγηση. Οι IP δρομολογητές δεν βλέπουν τις οπτικές ίνες, αλλά οι σύνδεσμοι που βλέπουν είναι τα οπτικά μονοπάτια. Για την δρομολόγηση στο IP επίπεδο η τεχνική βέλτιστης δρομολόγησης FD έχει άμεση εφαρμογή. Η ιδιαιτερότητα είναι ότι η τοπολογία του δικτύου είναι η εικονική τοπολογία που κατασκευάζεται από τα οπτικά μονοπάτια και την επίλυση του RWA προβλήματος.

Η εφαρμογή της μεθόδου FD στα πολυεπίπεδα οπτικά δίκτυα οδήγησε την παρούσα εργασία στη μελέτη και υλοποίηση FD αλγορίθμων και την ενσωμάτωση τους στο εργαλείο σχεδιασμού οπτικών δικτύων MANTIS.

2

Βέλτιστη δρομολόγηση, μοντέλα ροής και τοπολογική σχεδίαση

2.1 Βέλτιστη δρομολόγηση βάσει μοντέλων ροής

Για την αξιολόγηση της απόδοσης ενός αλγορίθμου δρομολόγησης, θα πρέπει να προσδιοριστεί ποσοτικά η έννοια της κυκλοφοριακής συμφόρησης. Στο κεφάλαιο αυτό, εισάγουμε την έννοια των μοντέλων επίδοσης βάση του ρυθμού άφιξης της κίνησης στους κόμβους του δικτύου. Τα μοντέλα αυτά ονομάζονται «μοντέλα ροής» (flow models) λόγω της σχέσης τους με τα μοντέλα βελτιστοποίησης της ροής ενός δικτύου και χρησιμοποιούνται για να διαμορφώσουν και να ορίσουν τα προβλήματα βέλτιστης δρομολόγησης, τα οποία εξετάζουμε στην παρούσα διπλωματική, αλλά και στον σχεδιασμό τμημάτων της τοπολογίας του δικτύου.

Η συμφόρηση σε ένα δίκτυο δεδομένων μπορεί να ποσοτικοποιηθεί βάσει των στατιστικών κατά τη διαδικασία άφιξης των δεδομένων στις ουρές του δικτύου. Σύμφωνα με αυτά, καθορίζονται οι κατανομές του μήκους της ουράς και ο χρόνος αναμονής των πακέτων σε κάθε σύνδεσμο χωρίς αυτό να είναι πάντα εύκολο να υπολογιστεί με ακρίβεια. Μία βολική αλλά κάπως ατελής εναλλακτική λύση είναι να μετρηθεί η συμφόρηση με βάση τη μέση κίνηση του συνδέσμου. Ακριβέστερα, υποθέτουμε ότι τα στατιστικά κατά τη διαδικασία άφιξης σε κάθε σύνδεσμο (i, j) αλλάζουν μόνο εξαιτίας των αναδιαμορφώσεων

στη δρομολόγηση. Επίσης μετρούμε την συμφόρηση στον (i, j) σύνδεσμο μέσω του δείκτη κίνησης στην άφιξη F_{ij} .

2.2 Ορισμός του προβλήματος

2.2.1 Συναρτήσεις κόστους βέλτιστης δρομολόγησης

Ονομάζουμε F_{ij} τη ροή του συνδέσμου (i, j) και το εκφράζουμε σε μονάδες δεδομένων (data units)/sec, όπου μονάδες δεδομένων μπορεί να είναι bits, πακέτα, μηνύματα, κτλ. Υποθέτουμε επίσης ότι ο ρυθμός άφιξης των δεδομένων στις ουρές του δικτύου διατηρείται σταθερός συναρτήσει του χρόνου.

Μία έκφραση της μορφής $\sum_{i,j} D_{ij}(F_{ij})$, όπου κάθε συνάρτηση D_{ij}

είναι γνησίως αύξουσα, είναι συνήθως κατάλληλη ως συνάρτηση του κόστους για βελτιστοποίηση.

Αν χρησιμοποιηθεί η προσέγγιση του Kleinrock (Kleinrock independence approximation), με την υπόθεση δηλαδή ότι κάθε ουρά συμπεριφέρεται ως μια M/M/1 ουρά από πακέτα, ο αναμενόμενος αριθμός πακέτων που βρίσκονται στην ουρά του συνδέσμου (i, j) , ή είναι υπό εξυπηρέτηση, ή είναι «εν πτήση», ή υπό επεξεργασία στον σύνδεσμο αυτό είναι

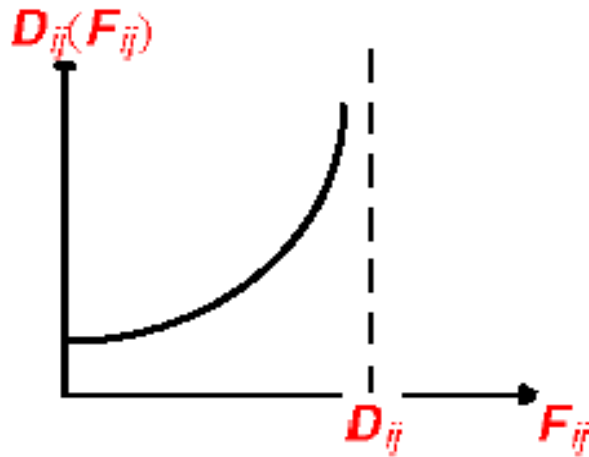
$$D_{ij}(F_{ij}) = \frac{F_{ij}}{C_{ij} - F_{ij}} + d_{ij} F_{ij}$$

όπου F_{ij} = ο αναμενόμενος ρυθμός δεδομένων (σε packets/sec) στον (i, j)

C_{ij} = η χωρητικότητα του συνδέσμου (i, j) (σε packets/sec)

και d_{ij} = η καθυστέρηση διάδοσης και επεξεργασίας.

Παρόλο που η υπόθεση αυτή συνήθως παραβιάζεται στα πραγματικά δίκτυα, η συνάρτηση κόστους αναπαριστά μια χρήσιμη μέτρηση της απόδοσης στην πράξη, κυρίως γιατί εκφράζει ποιοτικά ότι συμφόρηση πραγματοποιείται όταν μια ροή F_{ij} προσεγγίζει τον αντίστοιχο σύνδεσμο με χωρητικότητα C_{ij} .

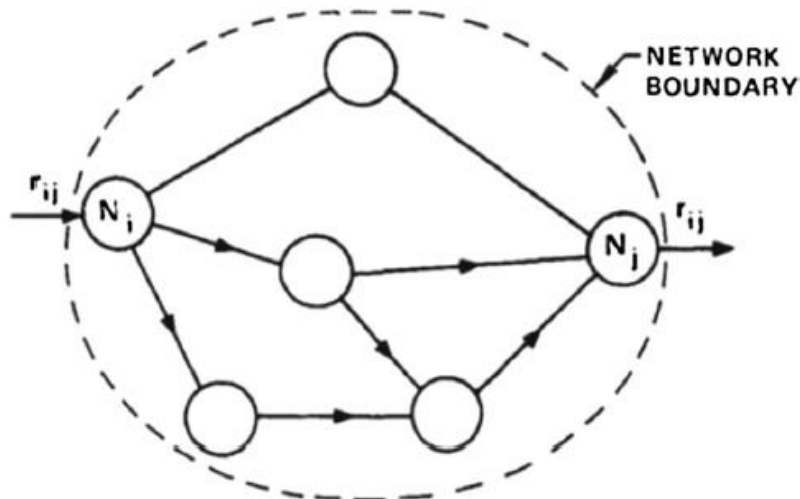


2.1. Συνάρτηση κόστους βάσει Kleinrock

Μια άλλη συνάρτηση κόστους με παρόμοιες ποιοτικές ιδιότητες δίνεται από τον τύπο $\max_{(i,j)} \left\{ \frac{F_{ij}}{C_{ij}} \right\}$ (μέγιστη χρησιμοποίηση συνδέσμου) .Ωστόσο, υπάρχει μικρή διαφορά ανάμεσα στα αποτελέσματα που δίνονται από τους δύο τύπους, όπως έχει δείξει σχετική υπολογιστική μελέτη. Αυτό σημαίνει πως μπορούμε να επιλέξουμε τον τύπο της συνάρτησης κόστους με γνώμονα την ευκολία κατά την διαδικασία της βελτιστοποίησης. Στην παρούσα διπλωματική, θα επικεντρωθούμε στις συναρτήσεις κόστους της μορφής $\sum D_{ij}(F_{ij})$.

2.2.2 Διατύπωση του προβλήματος

Προκειμένου να ορίσουμε το πρόβλημα, ας υποθέσουμε ένα δίκτυο με N το πλήθος κόμβους και μία ποσότητα πληροφορίας r_{ij} που πρέπει να δρομολογηθεί από τον κόμβο πηγής i στον κόμβο προορισμού j σε μία δοσμένη τοπολογία δικτύου, όπως αυτή του σχήματος που ακολουθεί.



2.2. Παράδειγμα δρομολόγησης ροής για το OD ζεύγος (i,j)

Για κάθε ζεύγος $w = (i, j)$ διακριτών κόμβων i και j (origin-destination ή αλλιώς OD pair), ο ρυθμός άφιξης της κίνησης υποτίθεται σταθερός, όπως ήδη έχουμε αναφέρει, με δείκτη r_w και μετράται σε data units/sec. Ο στόχος της δρομολόγησης είναι να χωρίσει κάθε r_w ανάμεσα στα πολλά μονοπάτια από την πηγή στον προορισμό με τρόπο που η προκύπτουσα ολική ροή στο σύνδεσμο ελαχιστοποιεί τη συνάρτηση κόστους.

Ακριβέστερα, συμβολίζουμε με

W = το σύνολο όλων των συνόδων που πρέπει να δρομολογηθούν, με άλλα λόγια το σύνολο όλων των OD ζευγαριών

P_w = το σύνολο όλων των κατευθυνόμενων μονοπατιών που συνδέουν τους κόμβους πηγής και προορισμού των OD ζευγαριών w (σε μια παραλλαγή του προβλήματος, το P_w είναι ένα δοσμένο υποσύνολο όλων των κατευθυνόμενων μονοπατιών που συνδέουν την πηγή με τον προορισμό του w).

x_p = η ροή σε data units/sec του μονοπατιού p .

Στη συνέχεια, απαιτούμε η ροή όλων των μονοπατιών $\{x_p \mid w \in W, p \in P_w\}$ να ικανοποιεί τους ακόλουθους περιορισμούς :

$$x_p \geq 0, \quad \text{για όλα τα } p \in P_w, w \in W$$

$$\sum_{p \in P_w} x_p = r_w, \quad \text{για όλα τα } w \in W,$$

Η ολική ροή F_{ij} του συνδέσμου (i, j) ορίζεται ως το άθροισμα των ροών όλων των μονοπατιών που διατρέχουν το σύνδεσμο, δηλαδή

$$F_{ij} = \sum_{\text{all paths } p \text{ containing } (i,j)} x_p$$

Θεωρώντας λοιπόν μια συνάρτηση κόστους της μορφής $\sum D_{ij}(F_{ij})$, το πρόβλημα εύρεσης του συνόλου των ροών των μονοπατιών που ελαχιστοποιούν τη συνάρτηση κόστους υπόκεινται στους περιορισμούς που παρουσιάζονται παρακάτω.

Συνοψίζοντας, το πρόβλημα βελτιστοποίησης λαμβάνει την μορφή:

$$\text{minimize } D(x) = \sum D_{(i,j)} \left[\sum_{\text{all paths } p \text{ containing } (i,j)} x_p \right]$$

$$\text{constraints } x_p \geq 0, \quad \text{για όλα τα } p \in P_w, w \in W$$

$$\sum_{p \in P_w} x_p = r_w, \quad \text{για όλα τα } w \in W.$$

Έτσι, το πρόβλημα διαμορφώνεται με άγνωστη μεταβλητή την ροή κάθε μονοπατιού του δικτύου x_p . Αυτό είναι το βασικό πρόβλημα δρομολόγησης που θα εξετάσουμε. Είναι όμως χρήσιμο σε αυτό το σημείο να εισάγουμε την έννοια της πρώτης παραγώγου D'_{ij} της συνάρτησης D_{ij} . Υποθέτουμε ότι κάθε συνάρτηση D_{ij} είναι παραγωγίσιμη ως προς F_{ij} και ορίζεται στο διάστημα $[0, C_{ij})$, όπου C_{ij} είναι είτε κάποιος θετικός αριθμός είτε τείνει στο άπειρο, ενώ συνήθως παριστάνει την χωρητικότητα συνδέσμου. Θέτω $x = x_p$ και σύμφωνα με τα προηγούμενα λαμβάνω

$$D(x) = \sum D_{(i,j)} \left[\sum_{\text{all paths } p \text{ containing } (i,j)} x_p \right]$$

και υπολογίζω την μερική παράγωγο του κόστους D ως προς το x_p . Οπότε ισχύει

$$\frac{\partial D(x)}{\partial x_p} = \sum_{\text{all paths } p \text{ containing } (i,j)} D'_{ij}$$

Έστω ότι $x^* = \{x_p^*\}$ είναι ένα βέλτιστο διάνυσμα ροών για κάποιο ζεύγος πηγής-προορισμού w με σύνολο μονοπατιών P_w . Τότε, οποιαδήποτε μετακίνηση κυκλοφορίας δ (δ πολύ μικρή ποσότητα, $\delta > 0$) από ένα μονοπάτι p σε ένα άλλο μονοπάτι p' δεν δύναται να μειώσει το συνολικό κόστος $D(x)$, αφού το x^* είναι

το βέλτιστο. Επομένως, η αλλαγή στο κόστος εξαιτίας μίας τέτοιας μετακίνησης μπορεί να εκφραστεί ως

$$\delta \frac{\partial D(x^*)}{\partial x'_p} - \delta \frac{\partial D(x^*)}{\partial x_p}$$

Και επειδή αυτή η αλλαγή πρέπει εξ'ορισμού να είναι μην αρνητική, προκύπτει ότι

$$x_p^* \geq 0 \quad \Rightarrow \quad \frac{\partial D(x)}{\partial x_p} \geq \frac{\partial D(x)}{\partial x_p}, \text{ για όλα τα } p' \in P_w$$

Γίνεται επομένως κατανοητό ότι το βέλτιστο διάνυσμα ροής είναι θετικό μόνο στα μονοπάτια που η τιμή της συνάρτησης της πρώτης παραγώγου του κόστους είναι ελάχιστη. Ακόμη, η παραπάνω σχέση αποτελεί ικανή και αναγκαία συνθήκη για την εύρεση του βέλτιστου μονοπατιού σε αλγόριθμους δρομολόγησης.

3

Αλγόριθμοι βέλτιστης δρομολόγησης

Στη γενική περίπτωση, η διαδικασία εύρεσης λύσης στο πρόβλημα της δρομολόγησης μπορεί να αποδειχθεί δύσκολη και να χρειαστεί η συμβολή πολλών διαφορετικών μεθόδων για το σκοπό αυτό. Ένα κρίσιμο σημείο αποτελεί η συνεχής μείωση του κόστους με παράλληλη διατήρηση της ικανοποίησης όλων των απαιτήσεων/περιορισμών. Στο προηγούμενο κεφάλαιο είδαμε ότι βέλτιστη δρομολόγηση επιτυγχάνεται μόνο όταν η πληροφορία ρέει σε μονοπάτια ελάχιστης πρώτης παραγώγου (Minimum First Derivative Path) MDFL για κάθε κόμβο πηγής - προορισμού. Αυτό υποδηλώνει ότι μπορούμε να βελτιστοποιήσουμε τη δρομολόγηση σε ένα δίκτυο ελαχιστοποιώντας ταυτόχρονα τη συνάρτηση κόστους, μεταφέροντας ροή σε ένα MDFL μονοπάτι από τα υπόλοιπα για κάθε ζευγάρι πηγής-προορισμού, ικανοποιώντας τους περιορισμούς / διατηρώντας την εφικτότητα (feasibility) της λύσης. Σε αυτήν ακριβώς την ιδέα, της μετακίνησης δηλαδή ενός τμήματος της ροής των άλλων μονοπατιών στο MDFL συντομότερο, βασίζονται οι δύο μέθοδοι που θα εξετάσουμε και θα υλοποιήσουμε στη συνέχεια.

3.1 General flow deviation

3.1.1 Feasible direction μέθοδοι για βέλτιστη δρομολόγηση

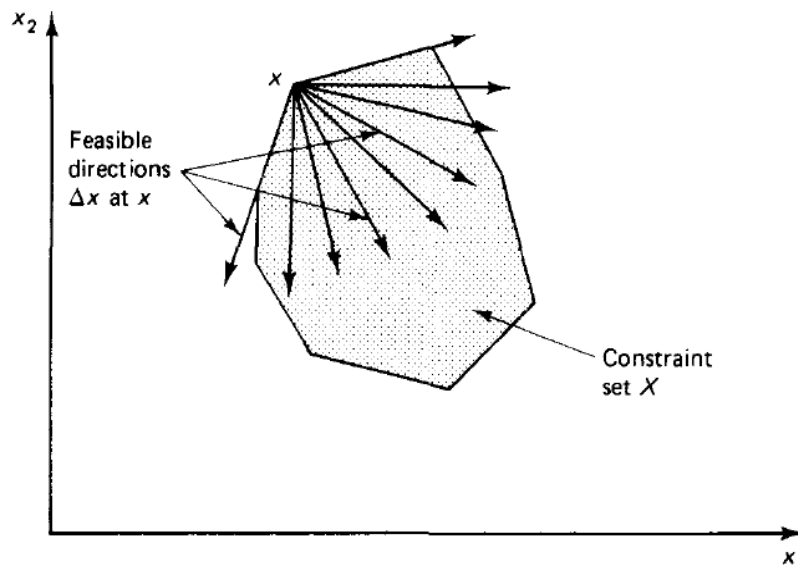
Δεδομένου ενός αρχικού διανύσματος ροής $x = x_p$, το οποίο ικανοποιεί τους περιορισμούς του προβλήματος, θέλουμε να εξετάσουμε την περίπτωση

μεταβολής του x κατά μία διεύθυνση $\Delta x = \Delta x_p$. Διακρίνουμε δύο περιορισμούς που θα πρέπει να επιβληθούν για την διεύθυνση Δx .

1. Η πρώτη και βασική απαίτηση που θα πρέπει να ικανοποιείται, εφόσον το αρχικό διάνυσμα $x = x_p$ αποτελεί μία feasible (εφικτή) λύση, είναι η διατήρηση της ιδιότητας. Πρέπει, δηλαδή η διεύθυνση Δx να είναι εφικτή (feasible direction). Με άλλα λόγια, όπως φαίνεται και στο επόμενο σχήμα, μικρές μεταβολές του x κατά τη διεύθυνση του Δx δεν διαταράσσουν την feasibility του συστήματος.

Η μαθηματική έκφραση αυτής της πρώτης απαίτησης που αφορά τελικώς τη διατήρηση της συνολικής ροής δίνεται παρακάτω.

$$\sum_{p \in P_w} \Delta x_p = 0, \text{ για όλα τα } w \in W$$



3.1. Παράδειγμα feasible διευθύνσεων Δx

2. Σε δεύτερο στάδιο, απαιτούμε η Δx να είναι μία «καθοδική» διεύθυνση (descent direction), με την έννοια ότι μικρές μεταβολές του διανύσματος x κατά την Δx μειώνουν τη συνάρτηση κόστους.

Η μαθηματική έκφραση της δεύτερης απαίτησης προκύπτει από τη συνθήκη ότι το εσωτερικό γινόμενο της κλίσης $\nabla D(x)$ και του διανύσματος Δx πρέπει να είναι αρνητικό, δηλαδή

$$\sum_{w \in W} \sum_{p \in P_w} \frac{\partial D(x)}{\partial x_p} \Delta x_p < 0$$

Ένας τρόπος ικανοποίησης της τελευταίας συνθήκης, που χρησιμοποιείται συχνά σε αλγορίθμους δρομολόγησης, είναι να απαιτήσουμε

$$\sum_{p \in P_w} \Delta x_p = 0 \text{ και}$$

$\Delta x_p \leq 0$, για όλα τα μη-συντομότερα (nonshortest) μονοπάτια (με την έννοια

$$\text{ότι } \frac{\partial D(x)}{\partial x_p} \geq \frac{\partial D(x)}{\partial x_p} \text{ για κάποιο } p \text{ του ίδιου OD pair)}$$

$\Delta x_p > 0$, για τουλάχιστον ένα μη-συντομότερο (nonshortest) μονοπάτι p .

Με πιο απλά λόγια, για την ικανοποίηση των (1) και (2) , εξασφαλίζουμε ότι κάποιο μέρος θετικής ροής μεταφέρεται από τα μη-συντομότερα (nonshortest) μονοπάτια στο συντομότερο (με βάση τα μήκη $D'_{ij}(F_{ij})$) ενώ δεν συμβαίνει το αντίστροφο.

Στην κατηγορία αλγορίθμων βέλτιστης δρομολόγησης που εξετάζουμε στην επόμενη ενότητα, η βασική επανάληψη δίνεται από την ακόλουθη σχέση.

$$x := x + a \Delta x$$

όπου Δx είναι μία feasible descent κατεύθυνση όπως ορίστηκε παραπάνω. Το βήμα a μπορεί να διαφέρει σε κάθε επανάληψη ή ακόμα για κάθε μονοπάτι κάθε OD ζεύγους.

3.2 The Frank-Wolfe (Flow Deviation) Method

Στην παράγραφο αυτή παρουσιάζουμε μία ειδική περίπτωση της μεθόδου Frank-Wolfe, η οποία ενδείκνυται για την επίλυση γενικών, μη γραμμικών προγραμματιστικών προβλημάτων με κυρτά σύνολα περιορισμών. Ο αλγόριθμος αυτός ονομάζεται «μέθοδος απόκλισης ροής» ή αλλιώς Flow deviation (FD) method και μπορεί να αποδειχθεί ότι ελαχιστοποιεί την τιμή της συνάρτησης κόστους στο όριο του βέλτιστου παρόλο που ο δείκτης σύγκλισής του κοντά σε αυτό τείνει να είναι πολύ αργός.

Δεδομένου ενός αρχικού feasible διάνυσματος ροής $x = \{x_p\}$, βρίσκουμε ένα minimum first derivative length (MDFL) μονοπάτι για κάθε OD ζεύγος (οι πρώτες παράγωγοι D'_{ij} υπολογίζονται φυσικά στο διάνυσμα x). Έστω ότι $\bar{x} = \{\bar{x}_p\}$ είναι το διάνυσμα των ροών των μονοπατιών το οποίο θα προέκυπτε εάν όλη η είσοδος r_w για κάθε OD ζεύγος $w \in W$ δρομολογηθεί-μεταφερθεί από το αντίστοιχο MDFL μονοπάτι. Επίσης, έστω ότι a^* είναι το βήμα που ελαχιστοποιεί την $D[x + a(\bar{x} - x)]$ για όλα τα $a \in [0,1]$, οπότε έχουμε ότι

$$D[x + a^*(\bar{x} - x)] = \min_{a \in [0,1]} D[x + a(\bar{x} - x)].$$

Το νέο σύνολο ροών των μονοπατιών προκύπτει από τη σχέση

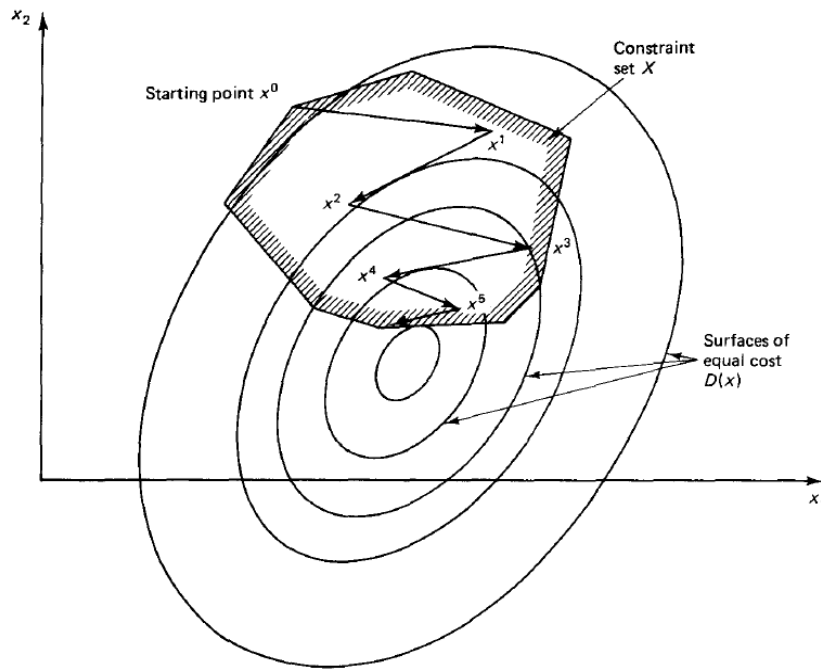
$$x_p := x_p + a(\bar{x}_p - x_p), \text{ για όλα τα } p \in P_w, w \in W$$

και η διαδικασία επαναλαμβάνεται.

Η χαρακτηριστική ιδιότητα αυτής της μεθόδου είναι ότι η ροή κατευθύνεται από τα μη-συντομότερα (nonshortest) μονοπάτια σε ίσες ποσότητες. Αυτό το στοιχείο άλλωστε την διαχωρίζει από τις gradient projection μεθόδους που συζητούνται στη συνέχεια. Η εύρεση του βέλτιστου βήματος a^* που καθορίζει αυτές τις ποσότητες ροής απαιτεί μία μονοδιάστατη ελαχιστοποίηση στο $[0,1]$ και μπορεί να πραγματοποιηθεί διαμέσου μιας από τις αρκετές ήδη υπάρχουσες μεθόδους. Ένας απλός τρόπος είναι να επιλεγεί το βήμα a^* μέσω της σχέσης

$$a^* = \min \left[1, - \frac{\sum_{(i,j)} (\bar{F}_{ij} - F_{ij}) D'_{ij}}{\sum_{(i,j)} (\bar{F}_{ij} - F_{ij})^2 D''_{ij}} \right]$$

Εδώ, τα $\{F_{ij}\}$ και $\{\bar{F}_{ij}\}$ είναι τα σύνολα των συνολικών ροών συνδέσμων που αντιστοιχούν στα $\{x_p\}$ και $\{\bar{x}_p\}$, δηλαδή η F_{ij} (ή \bar{F}_{ij}) προκύπτει προσθέτοντας όλες τις ροές x_p (ή αντίστοιχα \bar{x}_p) των μονοπατιών p που διατρέχουν τον σύνδεσμο (i, j) . Οι πρώτες και δεύτερες παράγωγοι D'_{ij} και D''_{ij} υπολογίζονται βάσει των F_{ij} . Για τον τύπο των συναρτήσεων κόστους που εξετάζουμε στην παρούσα διπλωματική (π.χ. το μοντέλο καθυστέρησης M/M/1), φαίνεται πως η επιλογή του παραπάνω βήματος συνήθως οδηγεί σε σύγκλιση ακόμα και αν το αρχικό διάνυσμα ροών απέχει πολύ από το βέλτιστο.



3.2. Εφαρμογή flow deviation method σε αρχικό διάνυσμα x^0

Σε περίπτωση που η αρχική λύση (starting point x^0 του σχήματος) $x = \{x_p\}$ είναι μη εφικτή (infeasible) προσπαθούμε να εφαρμόσουμε την παραπάνω μέθοδο τμηματικά. Πιο συγκεκριμένα, ένας εύκολος τρόπος υλοποίησης της παραπάνω ιδέας είναι να ορίσουμε μια παράμετρο $a^{\text{int}} = \min(C_{ij}/F_{ij})$ (προφανώς ισχύει $a^{\text{int}} < 1$ στους συνδέσμους όπου η αρχική λύση χαρακτηρίζεται infeasible) και να δρομολογήσουμε βέλτιστα μέρος του αρχικού διανύσματος ρών όπως προηγουμένως, δηλαδή $x_p = x_p \cdot a^{\text{int}}$. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να βρεθεί κάποια feasible αρχική λύση, έστω x_p^{new} ($a^{\text{int}} > 1$). Σε διαφορετική περίπτωση, θα αποφανθούμε ότι δεν υπάρχει feasible λύση στο πρόβλημα. Στη συνέχεια, εφαρμόζουμε την flow deviation μέθοδο με αρχικό διάνυσμα ρών το x_p^{new} , όπως ακριβώς την περιγράψαμε παραπάνω.

Όσον αφορά τώρα την ταχύτητα σύγκλισης, αντιλαμβανόμαστε ότι ο ρυθμός με τον οποίο συγκλίνει η μέθοδος κοντά στη βέλτιστη λύση είναι αργός. Η αναλογία των διαδοχικών λαθών της συνάρτησης κόστους δίνεται από τη σχέση

$$\frac{D(x^{k+1}) - D(x^*)}{D(x^k) - D(x^*)}$$

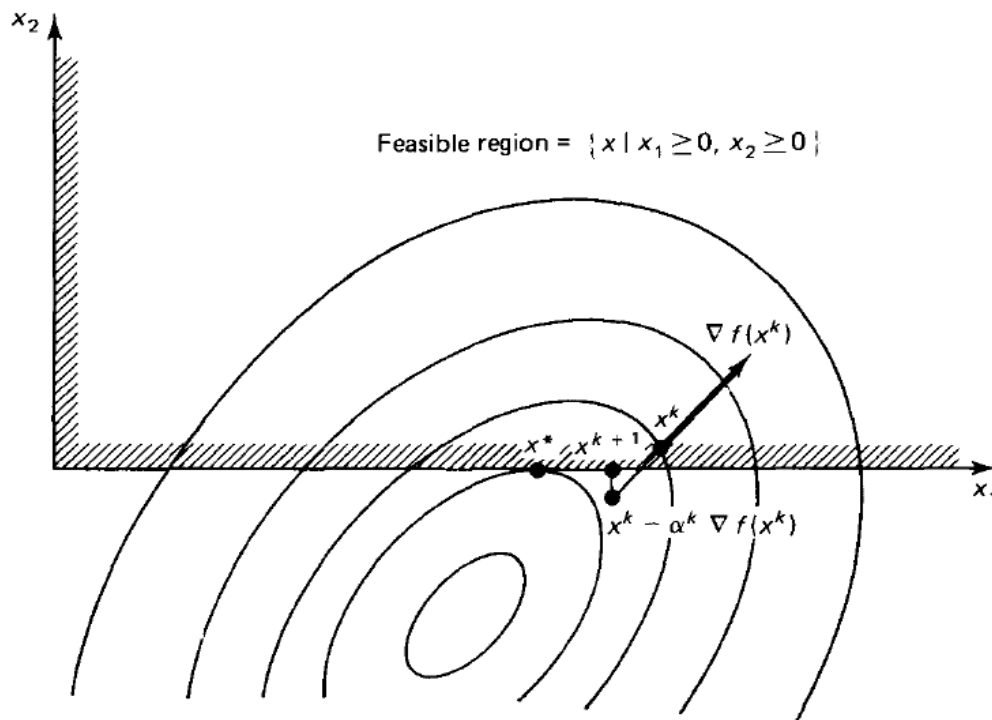
και συγκλίνει στο 1 καθώς $k \rightarrow \infty$.

Τέλος, αξίζει να αναφέρουμε μια κατάσταση όπου η μέθοδος Frank Wolfe εμφανίζει πλεονέκτημα σε σχέση με άλλες μεθόδους, όπως οι projection methods που θα συζητηθούν στη συνέχεια. Υποθέτουμε ότι δεν ενδιαφερόμαστε να

μάθουμε το βέλτιστο διάνυσμα ροών των μονοπατιών αλλά μας ενδιαφέρει μόνο η βέλτιστη συνολική ροή F_{ij} ή ακόμη μόνο η ελάχιστη τιμή της συνάρτησης κόστους. Σε αυτήν την περίπτωση, η μέθοδος Frank Wolfe μπορεί να εφαρμοστεί με τέτοιο τρόπο ώστε μόνο η τρέχουσα συνολική ροή με τα αντίστοιχα συντομότερα μονοπάτια να διατηρείται στη μνήμη σε κάθε επανάληψη. Οι απαιτούμενοι πόροι για την αποθήκευση αυτών των δεδομένων είναι σχετικά λίγοι, επιτρέποντας έτσι την επίλυση προβλημάτων σε μεγάλα δίκτυα.

3.3 Projection methods για βέλτιστη δρομολόγηση

Στο κεφάλαιο αυτό θα εξετάσουμε ένα σύνολο αλγορίθμων βέλτιστης δρομολόγησης και feasible κατεύθυνσης, που είναι γρηγορότεροι από τη μέθοδο Frank-Wolfe και προσφέρονται ευκολότερα για διαμοιρασμένη εφαρμογή. Οι projection method («μέθοδοι των προβολών») βασίζονται επίσης στην εύρεση ενός MDFL μονοπατιού για κάθε OD ζεύγος σε κάθε προσέγγιση. Μία μικρή ποσότητα αλλαγής της ροής υπολογίζεται για κάθε μονοπάτι βάσει των αντίστοιχων μηκών στα μονοπάτια και μερικές φορές, βάσει των δεύτερων παραγώγων των συναρτήσεων κόστους. Αν η ποσότητα αυτή είναι αρκετά μεγάλη ώστε η ροή του μονοπατιού να γίνει αρνητική, τότε η ροή του μονοπατιού λαμβάνεται μηδέν.



3.3. Εφαρμογή projection method για την k επανάληψη

3.3.1 Εφαρμογή σε βέλτιστη δρομολόγηση

Ας επικεντρωθούμε όμως τώρα στο πρόβλημα βέλτιστης δρομολόγησης, στο οποίο επιδιώκουμε να ελαχιστοποιήσουμε την

$$D(x) = \sum_{(i,j)} D_{ij}(F_{ij})$$

$$\text{ως προς } \sum_{p \in P_w} x_p = r_w \text{ για όλα τα } p \in P_w, w \in W$$

όπου κάθε συνολική ροή συνδέσμου F_{ij} εκφράζεται βάσει του διανύσματος ροών των μονοπατιών $x = \{x_p\}$ ως άθροισμα των ροών των μονοπατιών που διατρέχουν τον σύνδεσμο (i, j) . Υποθέτουμε επίσης πως οι δεύτερες παράγωγοι της D_{ij} , τις οποίες συμβολίζουμε ως $D''_{ij}(F_{ij})$ είναι θετικές για κάθε F_{ij} . Έστω ότι $x^k = \{x^k\}$ είναι το διάνυσμα ροών των μονοπατιών που προκύπτουν μετά από k επαναλήψεις της μεθόδου και $\{F_{ij}^k\}$ είναι το αντίστοιχο σύνολο των συνολικών ροών στους συνδέσμους. Για κάθε OD ζεύγος w , έστω \bar{p}_w ένα MFDL μονοπάτι ως προς τα μήκη του αντίστοιχου συνδέσμου $D'_{ij}(F_{ij})$.

Το παραπάνω πρόβλημα βέλτιστης δρομολόγησης μπορεί να μετατραπεί σε ένα πρόβλημα που περιλαμβάνει μόνο θετικούς περιορισμούς εκφράζοντας τις ροές των MFDL μονοπατιών \bar{p}_w ως προς τις ροές των άλλων μονοπατιών, εξαλείφοντας παράλληλα τους περιορισμούς ισότητας, όπως τον περιορισμό

$$\sum_{p \in P_w} x_p = r_w .$$

Για κάθε w , το $x_{\bar{p}_w}$ αντικαθίσταται στη συνάρτηση κόστους $D(x)$ χρησιμοποιώντας την εξίσωση

$$x_{\bar{p}_w} = r_w - \sum_{\substack{p \in P_w \\ p \neq \bar{p}_w}} x_p$$

και κατά συνέπεια προκύπτει ένα πρόβλημα της μορφής

$$\text{minimize } \tilde{D}(\bar{x}),$$

$$\text{για } x_p \geq 0, \text{ για κάθε } w \in W, p \in P_w, p \neq \bar{p}_w,$$

όπου το \bar{x} είναι το διάνυσμα για όλες τις ροές μονοπατιών που δεν είναι MFDL μονοπάτια.

Στη συνέχεια, υπολογίζουμε τις παραγώγους που θα χρειαστούν ώστε να εφαρμόσουμε την projection method στο προηγούμενο πρόβλημα. Χρησιμοποιώντας τις παραπάνω σχέσεις και τον ορισμό της $\tilde{D}(\bar{x})$ παίρνουμε:

$$\frac{d\tilde{D}(\bar{x}^k)}{dx_p} = \frac{dD(x^k)}{dx_p} - \frac{dD(x^k)}{dx_{\bar{p}_w}}, \text{ για κάθε } p \in P_w, p \neq \bar{p}_w$$

για όλα τα $w \in W$. Έχουμε δείξει πως $\frac{dD(x)}{dx_p}$ είναι η πρώτη παράγωγος του μήκους του μονοπατιού p , και ισχύει:

$$\frac{dD(x^k)}{dx_p} = \sum_{\substack{\text{all links } (i,j) \\ \text{on path } p}} D'_{ij}(F_{ij}^k).$$

Όσον αφορά τις δεύτερες παραγωγούς, μια απευθείας παραγωγή των παραπάνω εκφράσεων της πρώτης παραγωγού δείχνει ότι:

$$\frac{d^2 D(x^k)}{(dx_p)^2} = \sum_{(i,j) \in L_p} D''_{ij}(F_{ij}^k), \text{ για όλα τα } w \in W, p \in P_w, p \neq \bar{p}_w,$$

όπου, για κάθε p :

$L_p =$ είναι το σύνολο των συνδέσμων που ανήκουν είτε στο p είτε στο αντίστοιχο MFDL μονοπάτι \bar{p}_w όμως όχι και στα δύο.

Οι εκφράσεις και για την πρώτη αλλά και για την δεύτερη παράγωγο του μειωμένου κόστους $\tilde{D}(\bar{x})$ έχουν τώρα βρεθεί και επομένως η projection method μπορεί να εφαρμοσθεί. Η επαναληπτική σχέση λαμβάνει τη μορφή

$$x_p^{k+1} = \max \{0, x_p^k - \alpha^k H_p^{-1} (d_p - d_{\bar{p}_w})\}, \text{ για όλα τα } w \in W, p \in P_w, p \neq \bar{p}_w,$$

όπου d_p και $d_{\bar{p}_w}$ είναι τα μήκη των πρώτων παραγωγών των μονοπατιών p και \bar{p}_w που ορίζονται ως:

$$d_p = \sum_{\substack{\text{all links } (i,j) \\ \text{on path } p}} D'_{ij}(F_{ij}^k), \quad d_{\bar{p}_w} = \sum_{\substack{\text{all links } (i,j) \\ \text{on path } p}} D'_{ij}(F_{ij}^k),$$

και το H_p είναι το «μήκος» της δεύτερης παραγωγού

$$H_p = \sum_{(i,j) \in L_p} D''_{ij}(F_{ij}^k)$$

Το βήμα α^k είναι μια θετική μεταβλητή που μπορεί να επιλεχθεί με πολλές μεθόδους. Μία δυνατότητα είναι να διατηρήσουμε το $\alpha^k = \alpha$ σταθερό για κάθε επανάληψη k . Αποδεικνύεται ότι αν η αρχική προσέγγιση είναι feasible, τότε υπάρχει $\alpha^k \in (0, \bar{\alpha})$, δηλαδή υπάρχει ένα σύνολο δυνατών επιλογών βήματος για το οποίο η μέθοδος συγκλίνει στη βέλτιστη λύση δρομολόγησης. Η εμπειρία και η έρευνα έχουν δείξει ότι η επιλογή ενός α^k κοντά στο 1 συνήθως λειτουργεί καλά ανεξάρτητα από τις τιμές της εισόδου r_w . Ακόμη καλύτερη

απόδοση επιτυγχάνουμε στην περίπτωση που οι επαναλήψεις πραγματοποιούνται για μόνο ένα OD ζεύγος, ρυθμίζοντας τις συνολικές ροές των συνδέσμων προκειμένου να ληφθεί υπόψη η αλλαγή αυτή. Η διαδικασία αυτή επαναλαμβάνεται για κάθε OD ζεύγος. Τέλος, η κατάλληλη επιλογή του α^k επηρεάζει έως ένα ποσοστό και την ταχύτητα σύγκλισης του αλγορίθμου.

Ακόμη καλύτερη απόδοση επιτυγχάνουμε συνήθως εάν η επανάληψη πραγματοποιείται για ένα OD ζεύγος (ή για έναν μόνο κόμβο προέλευσης) κάθε φορά (βλ. Projection method – one-at-a-time mode, κεφ.4). Πιο αναλυτικά, πρώτα πραγματοποιείται η επανάληψη με βήμα $\alpha^k = 1$ για ένα OD ζεύγος και στη συνέχεια ρυθμίζεται το αντίστοιχο διάνυμα ροών σύμφωνα με την προηγούμενη αλλαγή. Η διαδικασία αυτή επαναλαμβάνεται κυκλικά μέχρις ότου εξαντληθούν όλα τα OD ζεύγη του δικτύου.

Καταλήγουμε επομένως στις εξής παρατηρήσεις για τον projection αλγόριθμο :

1. Εφόσον ισχύει ότι $d_p \geq d_{\bar{p}_w}$ για όλα τα $p \neq \bar{p}_w$, όλες οι θετικές ροές των μη-συντομότερων (nonshortest) μονοπατιών θα μειωθούν με το αντίστοιχο ποσοστό της ροής που μεταβαίνει στο MFDL μονοπάτι \bar{p}_w . Αν το βήμα α^k είναι αρκετά μεγάλο, όλη η ροή από τα μη-συντομότερα μονοπάτια θα μεταβεί στο συντομότερο μονοπάτι.
2. Οι ροές αυτών των μη-συντομότερων μονοπατιών $x_p, p \neq \bar{p}_w$ που είναι μηδενικές, θα παραμείνουν μηδενικές. Επομένως, η προσέγγιση για τη ροή των μονοπατιών που αναφέραμε προηγουμένως θα πρέπει να εφαρμόζεται μόνο για μονοπάτια που μεταφέρουν θετική ροή.
3. Μόνο τα μονοπάτια που μετέφεραν θετική ροή στο αρχικό μοτίβο ροής ή ήταν MFDL μονοπάτια σε κάποια προηγούμενη προσέγγιση μπορούν να μεταφέρουν θετική ροή στην εκκίνηση μιας προσέγγισης. Αυτό είναι σημαντικό, αφού τείνει να διατηρεί μικρό τον αριθμό των μονοπατιών που μεταφέρουν ροή, με αντίστοιχη μείωση στον αριθμό των υπολογισμών που χρειάζονται σε κάθε προσέγγιση.

Τέλος, συγκριτικά με τον Frank-Wolfe αλγόριθμο, αξίζει να σημειώσουμε ότι οι Projection μέθοδοι αποδίδουν συνήθως ταχεία σύγκλιση κοντά στην περιοχή της βέλτιστης λύσης. Από τη στιγμή που πλησιάζει κοντά σε μία λύση (το πόσο «κοντά» εξαρτάται από το προς επίλυση πρόβλημα) τείνει να επιβραδύνει. Η πρόοδος του είναι συχνά ικανοποιητική κοντά σε μία λύση και συνήθως πολύ καλύτερη από αυτήν της μεθόδου Frank-Wolfe.

4

Υλοποίηση και πειράματα προσομοίωσης

4.1 Περιγραφή πειραματικής διαδικασίας και παραδοχές

Στο κεφάλαιο αυτό παρουσιάζουμε τα αποτελέσματα ορισμένων παραδειγμάτων δικτύων δεδομένων, προκειμένου να γίνει άμεσα η λειτουργία των δύο αλγορίθμων που αναπτύχθηκαν και παρουσιάστηκαν θεωρητικώς στα προηγούμενα κεφάλαια της εργασίας, Frank-Wolfe και Projection method. Οι προσομοιώσεις αυτές πραγματοποιήθηκαν σε προγραμματιστικό περιβάλλον matlab. Επίσης, παρατίθεται μία σύντομη περιγραφή της πειραματικής διαδικασίας που ακολουθήθηκε καθώς και λεπτομέρειες σχετικά με την υλοποίηση των αλγορίθμων όπου αυτό κρίνεται απαραίτητο.

Ειδικότερα, δεδομένης μιας τοπολογίας G και επίσης δεδομένων των υπολοίπων παραμέτρων του δικτύου, όπως η μέγιστη χωρητικότητα των συνδέσμων (*capacity*) και οι χιλιομετρικές αποστάσεις μεταξύ των κόμβων (*dist*), εκτελούμε τις μεθόδους βέλτιστης δρομολόγησης για μία δεδομένη ποσότητα πληροφορίας (*traffic*). Η πειραματική διαδικασία εφαρμόστηκε για δύο διαφορετικές τοπολογίες δικτύων με διαφορετικές παραμέτρους και εξάχθηκαν αποτελέσματα και συμπεράσματα για κάθε περίπτωση. Σε αυτό το σημείο να σημειώσουμε επίσης ότι όλα τα δεδομένα που χρησιμοποιήθηκαν ως είσοδοι των αλγορίθμων για την εκτέλεση των προσομοιώσεων διατίθενται στο Παράρτημα Β, ενώ οι κώδικες εκτέλεσης σε matlab στο Παράρτημα Α της εργασίας.

Σχετικά με τον αλγόριθμο αναζήτησης συντομότερου μονοπατιού που χρησιμοποιήθηκε, επιλέξαμε αλγόριθμο Dijkstra και στα δύο παραδείγματα, ο οποίος έχει τη δυνατότητα υπολογισμού της συντομότερης διαδρομής μεταξύ ενός δεδομένου κόμβου του δικτύου προς όλους τους άλλους. Ο συγκεκριμένος αλγόριθμος χρησιμοποιείται τακτικά στην επίλυση προβλημάτων δρομολόγησης και συγκεκριμένα όσον αφορά τη μεταφορά πακέτων δεδομένων στα δίκτυα υπολογιστών, αφού αποτελεί τμήμα γνωστών πρωτοκόλλων δρομολόγησης όπως το IS-IS και το OSPF (Open Shortest Path First).

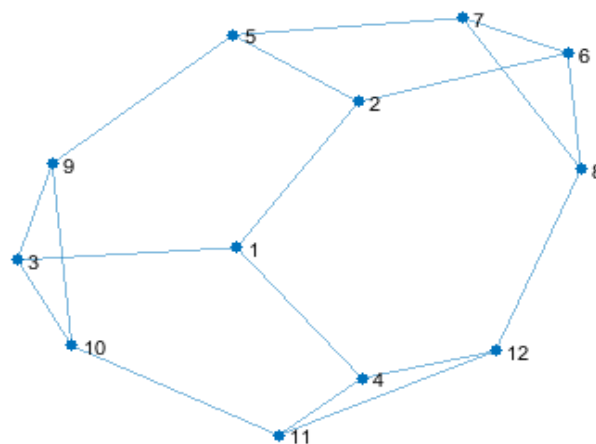
4.2 Προσομοίωση 1ης τοπολογίας δικτύου

4.2.1 Παράδειγμα 1ο

Για την αναπαράσταση των τοπολογιών των δικτύων που εξετάζουμε θα χρησιμοποιήσουμε συμμετρικό πίνακα γειτνίασης G_{-1} , $n \times n$, όπου $n=12$ το πλήθος των κόμβων.

$$G_{-1} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Η παραπάνω τοπολογία μπορεί να αναπαρασταθεί ισοδυνάμως με χρήση ενός μη κατευθυνόμενου γράφου.



4.1. 1st Topology

Σε αυτό το παράδειγμα θα θεωρήσουμε ότι τα χιλιόμετρα κόστη των ακμών του γράφου είναι όλα ίσα, και μάλιστα θέτουμε $dist = 1$ για όλες τις ακμές (i, j) , υπολογίζοντας με αυτόν τον τρόπο το συντομότερο μονοπάτι με κριτήριο τον ελάχιστο αριθμό απαιτούμενων βημάτων (min hops).

Η ποσότητα των δεδομένων που καλούμαστε να δρομολογήσουμε βέλτιστα αναπαριστάται επίσης από πίνακα διαστάσεων $n \times n$, όπου ο αποστολέας i προτίθεται να στείλει $tr_matrix(i, j)$ packets/sec στον παραλήπτη j .

$tr_matrix1 =$

0	7,966	1,464	27,79	5,337	1,903	0,668	2,87	5,378	10,88	1,917	0,489
11,5	0	50,08	26,55	22,34	49,99	35,12	16,3	10,01	5,798	10,63	1,628
0,225	3,552	0	5,358	0,445	0,241	0,356	0,239	0,853	0,344	0,543	0,166
56,53	32,48	43,06	0	101,3	43,55	34,23	75,74	141,2	99,29	125,7	28,22
11,33	3,101	1,269	6,414	0	3,51	1,25	0,775	1,958	0,764	1,297	0,286
7,039	11,16	2,162	18,69	8,941	0	1,567	5,109	5,457	3,301	3,899	0,673
0,263	2,789	0,5	5,022	0,684	0,401	0	0,393	0,981	0,387	0,727	0,192
8,689	2,003	1,345	8,292	2,476	2,645	1,193	0	5,258	2,809	2,366	0,544
4,138	8,601	2,925	33,52	11,06	8,55	6,5	6,212	0	5,832	11,11	3,405
0,388	0,358	0,609	18,8	1,158	0,72	0,545	2,141	2,693	0	1,052	0,437
0,89	0,701	0,687	14,81	1,695	1,21	0,84	2,706	5,968	1,871	0	0,637
0,39	0,407	0,423	4,775	0,789	0,503	0,432	1,176	1,351	0,473	1,076	0

Οφείλουμε επίσης να αναφέρουμε ότι χρησιμοποιήσαμε την προσέγγιση Kleinrock σε ότι αφορά την συνάρτηση κόστους που καλούμαστε να ελαχιστοποιήσουμε, ενώ έχουμε αγνοήσει την επίδραση της καθυστέρησης διάδοσης και επεξεργασίας (propagation delay). Αυτές οι παρατηρήσεις ισχύουν για όλα τα παραδείγματα και τις εφαρμογές που εκτελέσαμε στην παρούσα εργασία.

Επιλέγουμε τώρα την μέγιστη χωρητικότητα των συνδέσμων να ισούται με $capacity = 400$ packets/sec, οπότε εκτελούμε τους δύο αλγορίθμους βέλτιστης δρομολόγησης για αυτά τα δεδομένα. Όσον αφορά τον Projection method αλγόριθμο, εκτελούμε δύο εκδοχές του, *all-at-a-time* και *one-at-a-time*. Τα αποτελέσματα που προκύπτουν αφορούν την ελαχιστοποίηση της συνάρτησης κόστους (ολικό κόστος) σε k επαναληπτικά βήματα της εκάστοτε μεθόδου (iteration).

Iterations #k	Cost FD (Frank-Wolfe)	Cost Projection method (all-at-a-time)	Cost Projection method (one-at-a-time)
0	25.8292	25.8292	25.8292
10	16.9211	13.9107	13.5334
20	14.6825	13.5711	13.4581
35	13.9690	13.4682	13.4579
50	13.7621	13.4580	13.4579
100	13.5763	13.4579	13.4579

250	13.5269	13.4579	13.4579
500	13.5148	13.4579	13.4579
750	13.5111	13.4579	13.4579
1000	13.5093	13.4579	13.4579

Iterations #k	Total time FD	Total time Projection method all-at-a-time mode	Total time Projection method one-at-a-time mode
0	0.163 s	0.134 s	0.113 s
10	0.096 s	0.170 s	0.344 s
20	0.123 s	0.199 s	0.788 s
35	0.135 s	0.211 s	1.094 s
50	0.168 s	0.268 s	1.360 s
100	0.161 s	0.315 s	2.256 s
250	0.279 s	0.498 s	4.560 s
500	0.385 s	0.755 s	9.473 s
750	0.508 s	1.008 s	14.274 s
1000	0.714 s	1.230 s	18.790 s

4.2.2 Παράδειγμα 2^ο

Σε αυτό το σημείο της πειραματικής διαδικασίας οφείλουμε να καταγράψουμε δύο σημαντικές παρατηρήσεις σχετικά με την υλοποίηση των δύο μεθόδων.

1.Ο αλγόριθμος Projection method, εν αντιθέσει με την Flow deviation method, οδηγεί σε σύγκλιση υπό την βασική προϋπόθεση ότι το αρχικό διάγραμμα ροών είναι feasible, όπως επισημάνθηκε και στην θεωρητική περιγραφή της μεθόδου. Στην υλοποίηση και των δύο αλγορίθμων, έχουμε επιλέξει η αρχική λύση να προκύπτει διαιρώντας ισομερώς την κίνηση $tr_matrix(i, j)$ στα k διαθέσιμα μονοπάτια για κάθε OD ζεύγος (i, j) .

2. Ο αλγόριθμος Projection method εκτελείται για 4 διαφορετικές τιμές της μεταβλητής a^k , $a^k = \{1, 0.5, 0.25, 0.125\}$ και στη συνέχεια επιλέγεται αυτή που οδηγεί σε καλύτερη σύγκλιση, δηλαδή αυτή που ελαχιστοποιεί ταχύτερα τη συνάρτηση κόστους.

Επιλέγουμε στην συνέχεια να διατηρήσουμε σταθερές όλες τις χαρακτηριστικές μεταβλητές του δικτύου, μεταβάλλοντας ωστόσο την χωρητικότητα των συνδέσμων *capacity*. Θεωρούμε επίσης ως ικανοποιητικό κριτήριο σύγκλισης και τερματισμού των αλγορίθμων, όταν

$$Cost(D^{k+1}) - Cost(D^k) \leq 0.1\%$$

όπου k το πλήθος των επαναληπτικών βημάτων (iterations).

Καταγράφουμε επίσης, κατά την διαδικασία, των αριθμών των απαιτούμενων βημάτων, την μέγιστη ροή στους συνδέσμους $\max_{ij}(F_{ij})$, καθώς και τον συνολικό χρόνων εκτέλεσης των δύο μεθόδων. Για την Flow deviation method σημειώνεται επίσης το πλήθος των επαναλήψεων που εκτελούνται κατά την διαδικασία εύρεσης feasible αρχικής λύσης. Οπότε προκύπτουν τα παρακάτω αποτελέσματα.

Capacity	Cost FD (Frank-Wolfe)	Cost PR (Projection method) all-at-a-time mode	Cost PR (Projection method) one-at-a-time mode
35	no feasible solution	no feasible solution	no feasible solution
280	57.7868	non feasible initial solution	non feasible initial solution
300	33.2627	non feasible initial solution	non feasible initial solution
320	24.7536	non feasible initial solution	non feasible initial solution
350	18.5783	18.3997	18.3856
400	13.5930	13.4648	13.4586
500	9.2475	9.0041	9.0041
600	6.8522	6.8225	6.8221
900	3.9874	3.9764	3.9764
1200	2.8193	2.8128	2.8128
1500	2.1827	2.1781	2.1781
2000	1.5874	1.5841	1.5841

Capacity	loops of non feasible solution	Steps FD	Steps PR All-at-a-time mode	Steps PR One-at-a-time mode
35	-	-	-	-
280	773	2391	-	-
300	29	674	-	-
320	14	306	-	-
350	-	193	147	33
400	-	89	63	17
500	-	18	27	8
600	-	33	18	5
900	-	14	7	3
1200	-	9	5	3
1500	-	7	5	2
2000	-	4	4	2

Capacity	Total time FD	Total time PR all-at-a- time mode	Total time PR one-at-a- time mode	max(F) FD	max(F) PR all-at-a- time mode	max(F) PR One-at-a- time Mode
35	-	-	-	-	-	-
280	2.019 s	-	-	261.3273	-	-
300	0.619 s	-	-	262.2579	-	-
320	0.344 s	-	-	263.0365	-	-
350	0.160 s	0.238 s	0.743 s	264.4984	264.7867	265.1555
400	0.124 s	0.117 s	0.473 s	267.9500	269.3623	269.0860
500	0.092 s	0.113 s	0.303 s	290.9354	277.2537	276.9629
600	0.095 s	0.095 s	0.228 s	282.9454	285.5345	285.0396
900	0.091 s	0.070 s	0.158 s	307.3290	303.9903	304.2259
1200	0.089 s	0.095 s	0.168 s	314.8368	307.4785	307.5205
1500	0.108 s	0.067 s	0.124 s	324.6725	307.5718	307.5718
2000	0.073 s	0.079 s	0.150 s	331.3709	307.6538	307.6221

Παρατηρούμε αρχικά ότι για συγκεκριμένο εύρος τιμών της χωρητικότητας των συνδέσμων δεν υπάρχει feasible λύση στο πρόβλημα, καθώς η συνολική κίνηση του δικτύου υπερβαίνει τη μέγιστη συνολική του χωρητικότητα, δηλαδή

$$\sum_{ij} tr_matrix > \sum_{ij} capacity$$

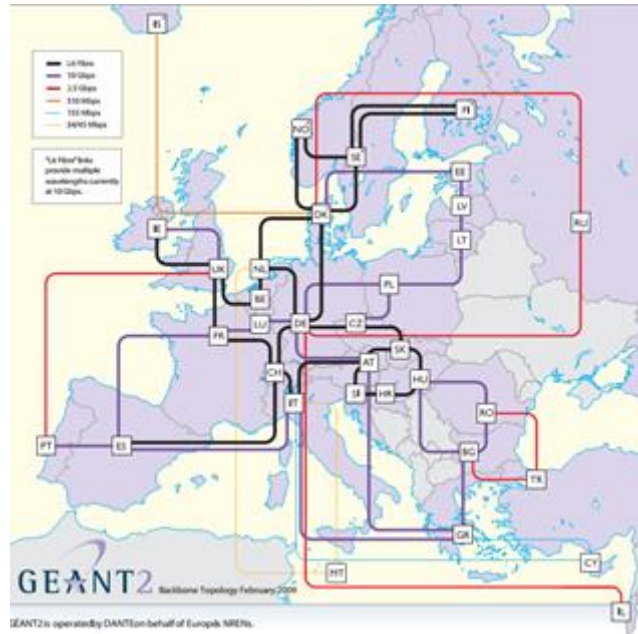
Η τιμή αυτή υπολογίζεται σε $capacity = 39.5790 \text{ packets / sec}$ για τις συγκεκριμένες απαιτήσεις του δικτύου.

4.3 Προσομοίωση 2^{ης} τοπολογίας δικτύου

4.3.1 Παράδειγμα 1^ο

Σε αυτήν την ενότητα εφαρμόζουμε την προηγούμενη πειραματική διαδικασία σε δίκτυο μεγαλύτερης κλίμακας, δηλαδή σε δίκτυο συνθετότερης τοπολογίας και μεγαλύτερου προς δρομολόγηση όγκο δεδομένων ανά δευτερόλεπτο.

Πιο συγκεκριμένα, η νέα τοπολογία του δικτύου απεικονίζεται στο επόμενο σχήμα, $n=34$ κόμβων. Πρόκειται για το GEANT2, το οποίο αποτελεί σήμερα το μεγαλύτερο και γρηγορότερο διακρατικό δίκτυο παγκοσμίως και έχει σχεδιαστεί ώστε να προωθήσει τη συνεργασία των επιστημονικών και εκπαιδευτικών κοινοτήτων και να επιτρέψει το χειρισμό τεράστιου όγκου ερευνητικών δεδομένων. Μεταξύ άλλων, το GEANT2, πρόκειται να χρησιμοποιηθεί για τη μετάδοση των δεδομένων από τον Μεγάλο Επιταχυντή Αδρονίων (LHC) που κατασκευάζεται στο ευρωπαϊκό ινστιτούτο σωματιδιακής φυσικής CERN αλλά και σε άλλες εφαρμογές που αφορούν τεράστιο όγκο πληροφορίας, όπως τη διασύνδεση μεγάλου αριθμού ραδιοτηλεσκοπίων σε όλη την Ευρώπη.



4.2. Δίκτυο GEANT 2

Σε αυτό το παράδειγμα, κριτήριο με βάσει το οποίο θα εφαρμοστεί ο αλγόριθμος του Dijkstra και στις δύο μεθόδους θα αποτελέσει η χιλιομετρική απόσταση μεταξύ των 34 κόμβων του δικτύου. Τα κόστη αυτά, που είναι διαθέσιμα στον πίνακα *dist2* του παραρτήματος, δίνονται ως είσοδος του αλγορίθμου.

Επίσης, για τον προσδιορισμό των k διαθέσιμων μονοπατιών για κάθε OD ζεύγος, επιλέχθηκαν disjoint μονοπάτια, μονοπάτια δηλαδή τα οποία δεν είχαν κοινούς συνδέσμους.

Δεδομένης λοιπόν και της κίνησης του δικτύου (*tr_matrix2*, παράρτημα Β), επιλέγουμε *capacity* = 700 *packets/sec*, εκτελούμε όπως ακριβώς στο παράδειγμα 1 της ενότητας 4.2.1 και συγκρίνουμε τα αποτελέσματα που προκύπτουν.

Iterations #k	Cost FD (Frank-Wolfe)	Cost Projection method (all-at-a-time)	Cost Projection method (one-at-a-time)
0	69.3329	69.3329	69.3329
2	39.4272	33.5277	30.6861
5	31.2643	31.1505	30.5993
10	30.7529	30.6273	30.5988
15	30.7372	30.6180	30.5987
20	30.7333	30.6119	30.5987
40	30.7317	30.6019	30.5987
60	30.7314	30.5985	30.5987
100	30.7312	30.5976	30.5987
150	30.7311	30.5975	30.5987

Iterations #k	Total time FD	Total time PR all-at-a-time mode	Total time PR one-at-a-time mode
0	2.304 s	2.423 s	0.931 s
2	2.560 s	2.453 s	1.352 s
5	2.521 s	2.566 s	1.765 s
10	2.440 s	2.638 s	2.484 s
15	2.484 s	2.790 s	3.077 s
20	2.564 s	2.836 s	3.921 s
40	2.597 s	3.308 s	6.434 s
60	2.675 s	3.556 s	9.527 s
100	2.854 s	4.197 s	14510 s
150	3.065 s	4.908 s	21.842 s

Οφείλουμε να σημειώσουμε ότι κατά την εκτέλεση της δεύτερης μεθόδου (Projection method) επιλέξαμε την παράμετρο a^k μέσα από ένα εύρος δυνατών τιμών $a^k = \{0.4, 0.2, 0.1, 0.005\}$, για λόγους που σχετίζονται με τα χαρακτηριστικά του δικτύου με γνώμονα πάντοτε την επίτευξη ταχύτερης σύγκλισης στη βέλτιστη λύση.

4.3.2 Παράδειγμα 2^ο

Σε αυτό το τελευταίο πείραμα, επιλέγουμε ως κριτήριο σύγκλισης και των δύο μεθόδων το ακόλουθο, καθώς ο υπολογιστικός χρόνος αυξάνεται σημαντικά για δίκτυα μεγαλύτερης κλίμακας όπως αυτό που εξετάζουμε.

$$\text{Cost}(D^{k+1}) - \text{Cost}(D^k) \leq 0.2\%$$

Στη συνέχεια εκτελούμε τους δύο αλγορίθμους για διαφορετικές τιμές της χωρητικότητας *capacity*, ενώ παράλληλα μετρούμε την τιμή της συνάρτησης κόστους, τη μέγιστη ποσότητα ροής σε σύνδεσμο του δικτύου, το πλήθος των επαναληπτικών βημάτων της μεθόδου, τον χρόνο εκτέλεσης της, καθώς και το πλήθος των βημάτων στην περίπτωση μη ύπαρξης αρχικής *feasible* λύσης (non feasible initial solution). Παραθέτουμε ενδεικτικό πίνακα αποτελεσμάτων.

Capacity	Cost FD (Frank-Wolfe)	Cost PR (Projection method) all-at-a-time mode	Cost PR (Projection method) one-at-a-time mode
40	no feasible solution	no feasible solution	no feasible solution
280	219.7314	non feasible initial solution	non feasible initial solution
320	132.0838	non feasible initial solution	non feasible initial solution
360	95.7263	non feasible initial solution	non feasible initial solution
480	53.2228	non feasible initial solution	non feasible initial solution
550	44.7033	44.3942	44.3558
600	38.8410	38.5998	38.5663
750	27.8566	27.7210	27.7078
880	22.3852	22.2692	22.2501

1000	18.9233	18.8374	18.8237
1300	13.6294	13.5928	13.5805
1500	11.4818	11.4736	11.4497

Capacity	loops of non feasible solution	Steps FD	Steps PR all-at-a-time mode	Steps PR one-at-a-time mode
40	-	-	-	-
280	275	341	-	-
320	141	132	-	-
360	91	127	-	-
480	14	62	-	-
550	-	36	25	7
600	-	30	23	6
750	-	16	36	4
880	-	11	17	4
1000	-	10	24	3
1300	-	9	28	2
1500	-	9	14	2

Capacity	Total time FD	Total time PR all-at-a-time mode	Total time PR one-at-a-time mode	max(F) FD	max(F) PR all-at-a-time mode	max(F) PR one-at-a-time mode
40	-	-	-	-	-	-
280	3.312 s	-	-	242.4902	-	-
320	2.798 s	-	-	251.5033	-	-
360	2.650 s	-	-	258.7895	-	-
480	2.168 s	-	-	284.3919	-	-
550	2.677 s	2.636 s	2.819 s	333.9263	333.6984	332.1566
600	2.601 s	2.646 s	2.609 s	345.7633	349.3520	346.3365
750	2.521 s	2.798 s	2.080 s	387.0964	384.9445	385.3764
880	2.501 s	3.809 s	2.152 s	402.7644	407.8654	407.8654
1000	2.527 s	3.727 s	2.071 s	410.3124	417.3715	416.7406
1300	2.553 s	2.699 s	1.759 s	438.5055	459.1890	463.3144
1500	2.523 s	2.577 s	1.768 s	465.6540	487.9898	487.2600

Υπολογίζουμε την ελάχιστη τιμή της χωρητικότητας των συνδέσμων προκειμένου να έχουμε αποδεκτή feasible λύση δρομολόγησης σε $capacity = 59.3206 \text{ packets / sec}$, όπως και στο προηγούμενο παράδειγμα.

4.4 Συγκρίσεις - Συμπεράσματα

Σε αυτήν την ενότητα συνοψίζουμε τα συμπεράσματα που προέκυψαν μέσα από τις προσομοιώσεις που εκτελέστηκαν προηγουμένως. Ως μία γενική εισαγωγική παρατήρηση θα αναφέρουμε ότι και οι δύο αλγόριθμοι επιλύουν

επιτυχώς το πρόβλημα βελτιστοποίησης που τέθηκε στις προηγούμενες ενότητες της εργασίας.

4.4.1 Συμπεράσματα 1^{ης} κατηγορίας πειραμάτων

Παρουσιάζουμε σε πρώτη φάση τα συμπεράσματα που προέκυψαν από την εκτέλεση του 1^{ου} και 3^{ου} παραδείγματος και που επιβεβαιώνονται από την εκτέλεση των υπολοίπων καθώς επίσης και από το θεωρητικό υπόβαθρο του 3^{ου} κεφαλαίου της εργασίας.

Αρχικά, οφείλουμε να αναγνωρίσουμε ότι και οι δύο μέθοδοι τείνουν να ελαχιστοποιούν το συνολικό κόστος δρομολόγησης, ενώ η διαφορά στα αποτελέσματα των δύο αλγορίθμων μετά από ικανοποιητικό αριθμό επαναληπτικών βημάτων δεν ξεπερνά το 2% .

Δεδομένου, τώρα, ενός αρχικού feasible διανύσματος ροών, επαληθεύεται πειραματικώς ότι η ταχύτητα σύγκλισης της Projection μεθόδου είναι μεγαλύτερη από αυτήν της Flow deviation , τόσο στα πρώτα επαναληπτικά βήματα όσο και κοντά στην προσέγγιση της βέλτιστης λύσης. Με άλλα λόγια, επιβεβαιώνουμε ότι ο αλγόριθμος Projection method ελαχιστοποιεί ταχύτερα την τιμή της συνάρτησης κόστους , και μάλιστα σε συγκρίσιμο χρόνο εκτέλεσης, ακόμη και σε δίκτυα σύνθετης τοπολογίας. Συνεπώς, παρόλο που δεν γνωρίζουμε σε καμία από τις περιπτώσεις που μελετήσαμε την βέλτιστη λύση του εκάστοτε προβλήματος δρομολόγησης ώστε να υπολογίζουμε με μαθηματική ακρίβεια την ταχύτητα σύγκλισης και το σφάλμα προσέγγισης των μεθόδων, είμαστε βέβαιοι ότι ο Projection αλγόριθμος είναι ταχύτερος, άρα και πιο ακριβείς όσον αφορά τα αποτελέσματα.

Συγκρίνοντας τώρα τις δύο εκδοχές του τελευταίου, διαπιστώνουμε ότι ο one-at-a-time mode αλγόριθμος επιτυγχάνει καλύτερη απόδοση στις πρώτες επαναλήψεις και στις δύο τοπολογίες. Όμως, επιβραδύνει όσο προσεγγίζει γειτονικές περιοχές της βέλτιστης λύσης και γίνεται πολύ αργός όσον αφορά τον χρόνο εκτέλεσης.

4.4.2 Συμπεράσματα 2^{ης} κατηγορίας πειραμάτων

Η βασική παρατήρηση που διαμορφώνεται από τα πειράματα 2, 4 αφορά πάλι την ταχύτητα σύγκλισης των μεθόδων, την ικανοποιητική δηλαδή προσέγγιση που επιτυγχάνουν στην βέλτιστη λύση, καθώς και τον χρόνο στο οποίο αυτό πραγματοποιείται.

Υποθέτουμε ότι υπάρχει αρχική feasible λύση στο πρόβλημα. Τότε, εξάγουμε τις εξής παρατηρήσεις:

1. Η δεύτερη μέθοδος (Projection method) επιτυγχάνει σε όλες τις περιπτώσεις εφαρμογής της καλύτερη προσέγγιση του βέλτιστου, δηλαδή ελαχιστοποιεί ταχύτερα την τιμή της συνάρτησης κόστους από ότι η πρώτη μέθοδος (Flow deviation method).

2. Ένα ακόμη ισχυρό πλεονέκτημα της 2^{ης} μεθόδου αποτελεί η παρατήρηση ότι αυτή η ακριβέστερη προσέγγιση επιτυγχάνεται εν γένει σε λιγότερα επαναληπτικά βήματα σε σύγκριση με την 1^η μέθοδο. Αυτό το συμβαίνει για την πλειοψηφία των περιπτώσεων εφαρμογής.
3. Καθοριστικής σημασίας για την επαλήθευση των παραπάνω αποτελεί η κατάλληλη επιλογή της παραμέτρου a^k κατά την εκτέλεση της 2^{ης} μεθόδου. Μπορεί να φανεί ότι για συγκεκριμένο μέγεθος του βήματος a^k , η σύγκλιση σε μία γειτονική περιοχή της λύσης είναι πολύ γρήγορη και για τις δύο εκδοχές του projection αλγορίθμου, all-at-a-time και one-at-a-time mode. Καθώς το μέγεθος του βήματος όμως αυξάνεται, αυξάνεται παράλληλα και ο κίνδυνος απόκλισης, πρώτα για την all-at-a-time λειτουργία και στη συνέχεια για την one-at-a-time.
4. Όσον αφορά τον χρόνο εκτέλεσης, δεν μπορούμε να εξάγουμε κάποιο ασφαλές συμπέρασμα, αφού όλοι οι μετρούμενοι χρόνοι δεν παρουσιάζουν κάποια αξιόλογη διαφορά στην προσπάθεια σύγκρισης των δύο μεθόδων. Ωστόσο, βλέπουμε ότι σε σύνθετες τοπολογίες δικτύων, ο χρόνος εκτέλεσης των αλγορίθμων αυξάνεται σημαντικά, ειδικά για τον one-at-a-time mode projection αλγόριθμο.

Στην περίπτωση τώρα μη feasible αρχικής λύσης, η Flow deviation μέθοδος μας δίνει μία ικανοποιητική πρώτη προσέγγιση για το διάλυμα ροών σε ένα μικρό σχετικά αριθμό βημάτων. Εάν αναζητούμε ακόμα καλύτερη προσέγγιση, θα μπορούσαμε στη συνέχεια να εφαρμόσουμε την 2^η μέθοδο με αρχικό διάλυμα ροής το αποτέλεσμα που δόθηκε προηγουμένως.

5

Ενσωμάτωση αλγορίθμων στο MANTIS

5.1 Mantis: Εργαλείο σχεδιασμού και λειτουργίας αποδοτικών οπτικών δικτύων

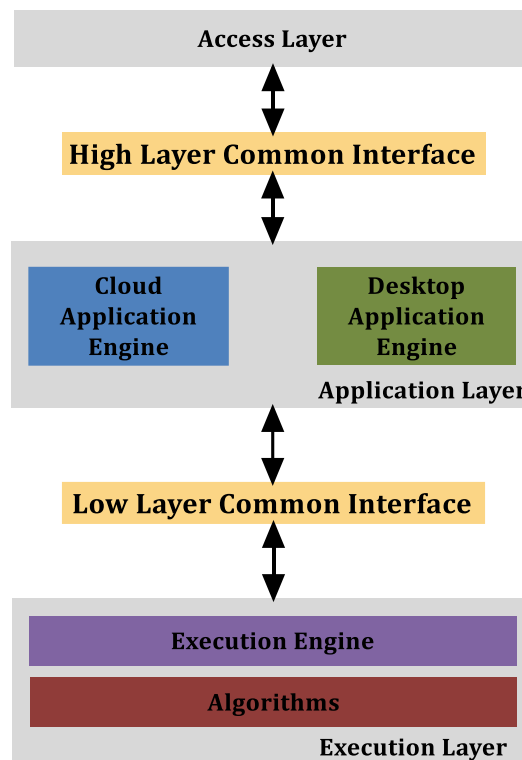
Το Mantis αποτελεί ένα ολοκληρωμένο εργαλείο για το σχεδιασμό και τον έλεγχο λειτουργίας των οπτικών δικτύων επόμενης γενιάς. Έχει σχεδιαστεί ώστε να εξασφαλίζει καλούς χρόνους εκτέλεσης για τα απαιτητικά υπολογιστικά προβλήματα σχεδιασμού και λειτουργίας οπτικών δικτύων, αποτελεσματική αξιοποίηση των διαθέσιμων υπολογιστικών πόρων, να παρέχει ανοχή στα σφάλματα των αξιοποιούμενων υπολογιστικών πόρων, να κλιμακώνεται με βάση την υπάρχουσα ζήτηση καθώς και να μπορεί εύκολα και άμεσα να επεκταθεί με νέα χαρακτηριστικά, ενεργειακά μοντέλα και αλγόριθμους.

Ένα βασικό χαρακτηριστικό του Mantis είναι η δυνατότητα λειτουργίας είτε ως κλασική εφαρμογή (Desktop Application) είτε ως υπηρεσία (Software as a Service) σε υποδομές Υπολογιστικού Νέφους. Η λειτουργία ως υπηρεσία είναι ιδιαίτερα σημαντική, αφού τα ζητήματα κλιμάκωσης και καλής απόδοσης μπορούν να βελτιωθούν μέσω της αξιοποίησης των πλεονεκτημάτων που προσφέρουν οι υποδομές Υπολογιστικού Νέφους. Επιπλέον, το εργαλείο έχει σχεδιαστεί με αρθρωτό τρόπο προκειμένου να υπάρχει η μέγιστη δυνατή επαναχρησιμοποίηση των διαφόρων λειτουργικών τμημάτων μεταξύ των δύο

εκδόσεων. Η αρχιτεκτονική επιτρέπει την επέκταση οποιουδήποτε λειτουργικού τμήματος χωρίς να επηρεάζεται η συνολική λειτουργικότητα της εφαρμογής.

Σύμφωνα με την αρχιτεκτονική του Mantis τα λειτουργικά τμήματα του εργαλείου οργανώνονται σε τρία επίπεδα: το *επίπεδο πρόσβασης* (Access Layer), το *επίπεδο εφαρμογής* (Application Layer) και το *επίπεδο εκτέλεσης* (Execution Layer). Επιπλέον, υπάρχουν δύο κοινές διεπαφές των οποίων ο πρωταρχικός σκοπός είναι η εξασφάλιση χαλαρής διασύνδεσης μεταξύ του επιπέδου εφαρμογής και των άλλων δύο επιπέδων. Στο **Error! Reference source not found.** παρουσιάζεται η αρχιτεκτονική του Mantis καθώς και τα βασικά λειτουργικά τμήματα του κάθε επιπέδου.

Το επίπεδο πρόσβασης χειρίζεται την αλληλεπίδραση με τους χρήστες μέσω μιας web-based διεπαφής και του διαθέσιμου RESTful API. Μέσω της web-based διεπαφής οι χρήστες έχουν πρόσβαση σε όλες τις λειτουργίες του εργαλείου, εκτελώντας με ευκολία όλες τις υποστηριζόμενες λειτουργίες ενώ μπορούν να συνεργάζονται με άλλους χρήστες. Το επίπεδο εφαρμογής υλοποιεί τη βασική λογική της εφαρμογής και αναλαμβάνει να εκτελέσει όλες τις αναγκαίες ενέργειες προκειμένου να εξυπηρετηθούν τα αιτήματα των χρηστών. Αποτελεί το μοναδικό επίπεδο ο σχεδιασμός και η υλοποίηση του οποίου διαφέρει μεταξύ των δυο εκδόσεων του Mantis, καθώς σε κάθε περίπτωση λειτουργίας υπάρχουν διαφορετικές απαιτήσεις και διαδικασίες που πρέπει να εκτελεστούν. Τέλος, το επίπεδο εκτέλεσης περιλαμβάνει τους αναγκαίους μηχανισμούς που επιτρέπουν την εξυπηρέτηση των αιτημάτων των χρηστών σχετικά με την εκτέλεση κάποιου αλγορίθμου. Το συγκεκριμένο επίπεδο χειρίζεται αποκλειστικά τους αλγορίθμους που περιλαμβάνει η εφαρμογή καθώς και την πλήρη εκτέλεσή τους στους διαθέσιμους πόρους.



5.1 Αρχιτεκτονική και βασικά λειτουργικά τμήματα Mantis

5.2 Προγραμματιστική διεπαφή εργαλείου και ενσωμάτωση νέων αλγορίθμων στο επίπεδο εκτέλεσης

5.2.1 Επίπεδο εκτέλεσης και ενσωμάτωση νέων αλγορίθμων

Το επίπεδο εκτέλεσης του Mantis αποτελείται από τη μηχανή εκτέλεσης (Execution Engine) και τη βιβλιοθήκη των διαθέσιμων αλγορίθμων για το σχεδιασμό και τη λειτουργία οπτικών δικτύων. Το επίπεδο περιλαμβάνει τους αναγκαίους μηχανισμούς που επιτρέπουν την εξυπηρέτηση των αιτημάτων των χρηστών σχετικά με την εκτέλεση των αλγορίθμων. Πιο συγκεκριμένα, ο plug-ins μηχανισμός που διαθέτει επιτρέπει στη μηχανή εκτέλεσης να διαχειρίζεται την εκτέλεση των αλγορίθμων, οι οποίοι μπορεί να έχουν υλοποιηθεί σε διάφορες γλώσσες προγραμματισμού. Ο plug-ins μηχανισμός εξασφαλίζει τη δυνατότητα άμεσης προσθήκης νέων αλγορίθμων, ακόμη και από τρίτους χρήστες, χωρίς να επηρεάζεται η υπάρχουσα λειτουργικότητα ούτε να χρειάζεται πλήρης γνώση του τρόπου λειτουργίας της εφαρμογής.

Οι αλγόριθμοι που μπορούν να ενσωματωθούν στο εργαλείο με βάση την τρέχουσα υλοποίηση του επιπέδου εκτέλεσης μπορεί να είναι υλοποιημένοι σε: (i) C/C++, (ii) Java και (iii) Python/Cython.

Γενικότερα, η διαδικασία για την ενσωμάτωση κάποιου αλγορίθμου στην εφαρμογή είναι σχετικά απλή και περιλαμβάνει τα ακόλουθα βήματα:

1. Υλοποίηση του αλγορίθμου με χρήση του plug-ins integration API.
2. Περιγραφή των παραμέτρων εισόδου και των αποτελεσμάτων με βάση το αντίστοιχο Mantis JSON σχήμα και προσδιορισμός των πιθανών εξαρτήσεων από άλλα πακέτα λογισμικού και βιβλιοθήκες.
3. Εισαγωγή των απαραίτητων πληροφοριών για το νέο αλγόριθμο (όνομα, σύντομη περιγραφή, περιγραφή παραμέτρων εισόδου και αποτελεσμάτων) στη βάση δεδομένων της εφαρμογής.
4. Προσθήκη του νέου αλγορίθμου στη βιβλιοθήκη της μηχανής εκτέλεσης.

Επιπλέον, ο plug-ins μηχανισμός καθορίζει μια κοινή διεπαφή, ανεξάρτητα από την τεχνολογία υλοποίησης του κάθε αλγορίθμου, η οποία προσδιορίζει σαφώς τη σύνταξη των παραμέτρων εισόδου και των αποτελεσμάτων για όλους τους αλγορίθμους. Η κοινή διεπαφή χρησιμοποιεί το πρότυπο JavaScript Object Notation (JSON) ως μορφή ανταλλαγής δεδομένων για την περιγραφή των αποτελεσμάτων και των απαραίτητων παραμέτρων εισόδου για όλους τους αλγορίθμους. Πιο συγκεκριμένα, κάθε αλγόριθμος στην εφαρμογή επιστρέφει τα αποτελέσματα εξόδου ως ένα αντικείμενο JSON και δέχεται ως είσοδο τις ακόλουθες παραμέτρους, χρησιμοποιώντας το προκαθορισμένο σχήμα JSON, με τη σειρά που παρουσιάζονται:

1. δικτυακή τοπολογία
2. απαιτήσεις κίνησης
3. παράμετροι spectrum grid
4. παράμετροι CAPEX/OPEX για transponders
5. παράμετροι CAPEX/OPEX για οπτικούς κόμβους
6. παράμετροι CAPEX/OPEX για routers
7. επιλογές και περιορισμοί σχετικά με τις διαθέσιμες δικτυακές συσκευές
8. παράμετροι εισόδου αλγορίθμου

Το επίπεδο πρόσβασης και η μηχανή εκτέλεσης πρέπει να γνωρίζουν για κάθε αλγόριθμο τη λεπτομερή περιγραφή των παραμέτρων εισόδου και των αποτελεσμάτων προκειμένου να είναι σε θέση να προετοιμάσουν την εκτέλεση του αλγορίθμου και να διαχειριστούν τα αποτελέσματα. Ωστόσο, σε αντίθεση με την περιγραφή της δικτυακής τοπολογίας, των απαιτήσεων κίνησης καθώς και των διαφόρων CAPEX/OPEX παραμέτρων οι παράμετροι εισόδου και τα αποτελέσματα μπορεί να διαφέρουν σημαντικά για κάθε αλγόριθμο. Αυτό έχει ως αποτέλεσμα να μην μπορούν να προσδιοριστούν αυτόματα από τα λειτουργικά τμήματα των επιπέδων εφαρμογής και εκτέλεσης.

Για αυτό το λόγο, κάθε φορά που ενσωματώνουμε ένα νέο αλγόριθμο στο επίπεδο εκτέλεσης του Mantis είναι αναγκαίο να παρέχουμε την περιγραφή των παραμέτρων εισόδου και των αποτελεσμάτων χρησιμοποιώντας μια προκαθορισμένη περιγραφή. Με την αξιοποίηση των συγκεκριμένων πληροφοριών, που βρίσκονται αποθηκευμένες στη βάση δεδομένων του εργαλείου, ο plug-ins μηχανισμός της μηχανής εκτέλεσης μπορεί να εκτελέσει

οποιοδήποτε αλγόριθμο χωρίς να είναι αναγκαία οποιαδήποτε άλλη τροποποίηση ή παρέμβαση.

Το JSON σχήμα που χρησιμοποιείται για την περιγραφή των παραμέτρων εισόδου κάθε αλγορίθμου αποτελείται από ένα αντικείμενο που περιλαμβάνει τις παραμέτρους:

- *container_name*
- *execution_wrapper*
- *input_parameters*
- *output_parameters*

Η πρώτη προσδιορίζει το όνομα του module ή της δυναμικής βιβλιοθήκης που περιέχει την υλοποίηση του αλγορίθμου, η δεύτερη τη τεχνολογία υλοποίησης που καθορίζει παράλληλα την κλάση που θα χειριστεί την εκτέλεση, ενώ οι άλλες δύο παράμετροι που αποτελούνται από λίστες αντικειμένων περιγράφουν αντίστοιχα τις παραμέτρους εισόδου και τα αποτελέσματα του αλγορίθμου.

5.2.2 Υλοποίηση και προσθήκη αλγορίθμου *OptimalRouting*

Υστερα από την αρχική υλοποίηση του αλγορίθμου στη Matlab και την αξιολόγησή του προχωρήσαμε στις απαραίτητες ενέργειες ώστε να ενσωματωθεί στο εργαλείο Mantis. Για το σκοπό αυτό, το πρώτο βήμα είναι η υλοποίηση του αλγορίθμου σε μία από τις 3 διαφορετικές γλώσσες προγραμματισμού (C++, Python και Java) που υποστηρίζει η μηχανή εκτέλεσης της εφαρμογής. Στην δική μας περίπτωση επιλέξαμε ως τεχνολογία υλοποίησης την γλώσσα προγραμματισμού C++ μιας και η Matlab παρέχει τα κατάλληλα εργαλεία και τις απαραίτητες βοηθητικές βιβλιοθήκες που επιτρέπουν την επιτυχή μετατροπή προγραμμάτων από Matlab σε C++.

Για την περίπτωση προσθήκης αλγορίθμων σε C++ έχουμε αξιοποιήσει τις δυνατότητες που μας παρέχει το πλαίσιο προγραμματισμού Qt για την επέκταση της λειτουργικότητας μιας εφαρμογής μέσω των Qt plug-ins. Πιο συγκεκριμένα, το τμήμα της μηχανής εκτέλεσης που διαχειρίζεται την εκτέλεση των συγκεκριμένων αλγορίθμων (QtPluginWrapper) καθορίζει μια σαφή διεπαφή που πρέπει να ακολουθεί κάθε αλγόριθμος, ο οποίος στην πράξη υλοποιείται ως μια κοινόχρηστη βιβλιοθήκη (shared library). Ο συγκεκριμένος τρόπος οργάνωσης εξασφαλίζει την άμεση πρόσβαση του QtPluginWrapper στις μεθόδους οποιoδήποτε αλγορίθμου υλοποιεί την διεπαφή καθώς και την διαφανή εκτέλεση και διαχείριση των αποτελεσμάτων.

Σχετικά με τις λεπτομέρειες υλοποίησης, αρχικά μετατρέψαμε την αρχική υλοποίηση του αλγορίθμου σε C++ βιβλιοθήκη ενώ στη συνέχεια καθορίσαμε την απαραίτητη κλάση που επιτρέπει την ενσωμάτωσή του στη μηχανή εκτέλεσης του Mantis. Πιο συγκεκριμένα, η διεπαφή του plug-ins integration API καθορίζει ότι η βασική κλάση που περιλαμβάνει τον ορισμό κάθε αλγορίθμου θα πρέπει να κληρονομεί από την γενική κλάση PluginInterfaceImpl, ενώ κάθε αλγόριθμος θα πρέπει να υλοποιεί όλες τις μεθόδους που περιλαμβάνει η γενική κλάση PluginInterfaceImpl. Ο Πίνακας 5.1 περιλαμβάνει τον ορισμό της βασικής

κλάσης που υλοποιεί όλες τις απαραίτητες μεθόδους για τη διασύνδεση του αλγορίθμου με το plug-ins μηχανισμό του επιπέδου εκτέλεσης.

```
#ifndef OPTIMALROUTING_H
#define OPTIMALROUTING_H

#include <QObject>
#include <QtCore/qplugin.h>
#include "pluginInterfaceImpl.h"

class OptimalRouting : public QObject, PluginInterfaceImpl
{
    Q_OBJECT
    Q_PLUGIN_METADATA(IID "Mantis.Plugin.PluginInterfaceImpl")
    Q_INTERFACES(PluginInterfaceImpl)
public:
    QString execute(QString topology, QString traffic, QString grid, QString
transponders, QString optical_nodes, QString ip_routers, QStrint constraints,
QString input_params);
    QString name();
private:
    QString sanitizeResults(QString results);
}
#endif // OPTIMALROUTING H
```

Πίνακας 5.1: Ορισμός κλάσης που υλοποιεί τη διασύνδεση του αλγορίθμου με το plug-ins μηχανισμό του επιπέδου εκτέλεσης

Η διαδικασία ενσωμάτωσης της υλοποίησης του αλγορίθμου στο εργαλείο Mantis ολοκληρώθηκε αρχικά με την εγκατάστασή του στη μηχανή εκτέλεσης, ενώ στη συνέχεια ενημερώθηκαν οι κατάλληλοι πίνακες στη βάση δεδομένων της εφαρμογής σχετικά με τις περιγραφές των παραμέτρων εισόδου και εξόδου. Τέλος ο Πίνακας 2 περιέχει την περιγραφή των παραμέτρων εισόδου και των αποτελεσμάτων για τον νέο αλγόριθμο.

```
{
    "container_name": "OptimalRouting",
```

```

"execution_wrapper": "QtPluginWrapper",
"input_parameters": [ ],
"output_parameters": [
  { "name": "optical_connections", "description": "Established
    transparent lightpaths", "type": "list-2"},
  { "name": "traffic_to_connections", "description": " Traffic demands
    over the established lightpaths", "type": "list-2"}
]
}

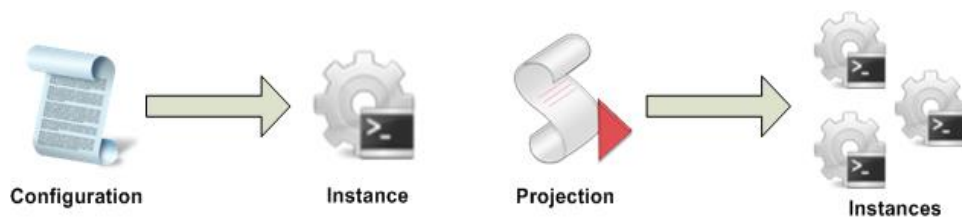
```

Πίνακας 2: Περιγραφή παραμέτρων εισόδου και αποτελεσμάτων για τον νέο αλγόριθμο

5.3 Διεπαφή χρήστη με το εργαλείο MANTIS

Το εργαλείο Mantis παρέχει μια σύγχρονη και απλή web-based διεπαφή μέσω της οποίας οι χρήστες μπορούν να έχουν πρόσβαση σε όλες τις υποστηριζόμενες λειτουργίες. Το **Error! Reference source not found.** παρουσιάζει την κεντρική σελίδα του Mantis μαζί με τις βασικές περιοχές.

Οι βασικές οντότητες στο Mantis είναι η περιγραφή εκτέλεσης (configuration), η εργασία (instance) και οι προβολές (projections). Μια περιγραφή εκτέλεσης αποτελείται από το σύνολο των παραμέτρων που προσδιορίζουν την εκτέλεση ενός συγκεκριμένου σεναρίου αξιολόγησης. Οι προβολές αποτελούν παραμετροποιημένες εκδοχές των περιγραφών εκτέλεσης, η εκτέλεση των οποίων έχει ως αποτέλεσμα τη δημιουργία μιας σειράς εργασιών οι οποίες διαφέρουν μόνο σε μία από τις αρχικές παραμέτρους (π.χ., αριθμό των μηκών κύματος). Το **Error! Reference source not found.** απεικονίζει την ακριβή σχέση μεταξύ περιγραφής εκτέλεσης, προβολής και εργασιών.



5.2. Σχέσεις μεταξύ περιγραφής εκτέλεσης, προβολής και εργασιών (instance)

Γενικότερα, μέσω της web-based διεπαφής του εργαλείου οι χρήστες μπορούν να:

1. δημιουργήσουν δικτυακές τοπολογίες με όλες τις απαραίτητες πληροφορίες τους
2. περιγράψουν απαιτήσεις κινήσεις που κάποια τοπολογία καλείται να εξυπηρετήσει

3. διαμορφώσουν τις παραμέτρους σχετικά με το κόστος απόκτησης και λειτουργίας των δικτυακών συσκευών, με ιδιαίτερη έμφαση στην κατανάλωση ενέργειας
4. χρησιμοποιήσουν μια βιβλιοθήκη αλγορίθμων για το σχεδιασμό, λειτουργία ή αξιολόγηση κάποιου οπτικού δικτύου που τους ενδιαφέρει.

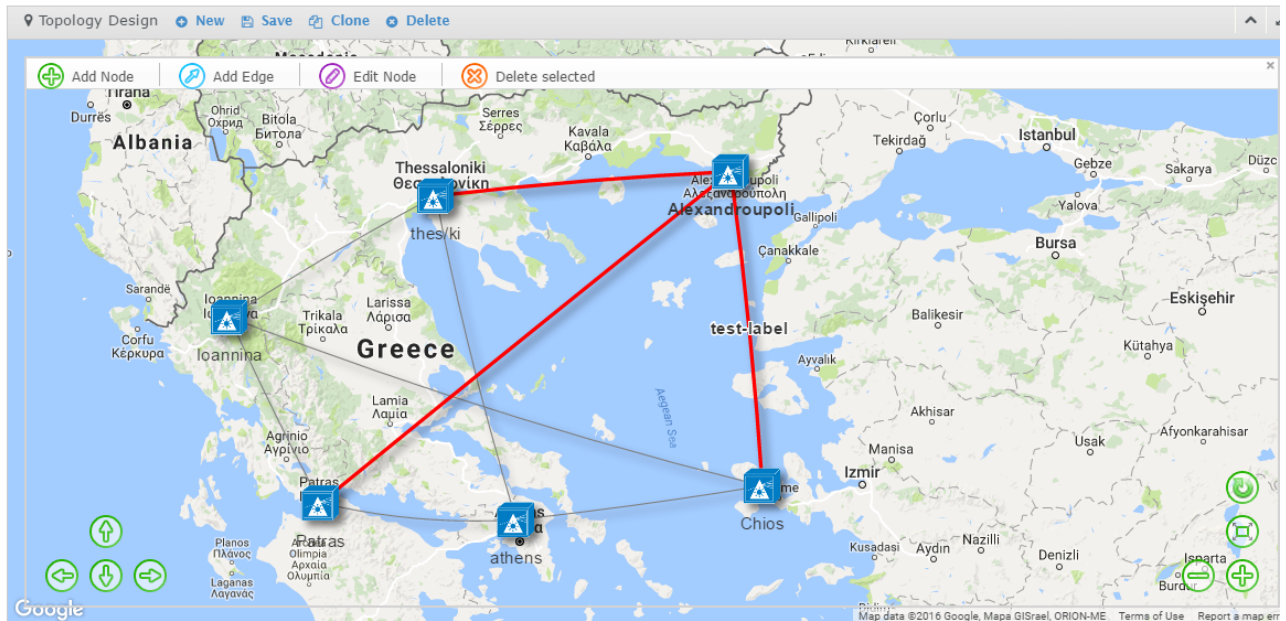
Οι διαθέσιμοι αλγόριθμοι αξιολογούν το σχεδιασμό ή τη λειτουργία ενός δικτύου χρησιμοποιώντας διάφορα μελλοντικά σενάρια λειτουργίας και απαιτήσεις κίνησης προτείνοντας μια λεπτομερή λύση για την κάθε περίπτωση. Στην προτεινόμενη λύση κάθε αλγόριθμος περιλαμβάνεται το απαιτούμενο εύρος ζώνης για την εξυπηρέτηση των απαιτήσεων κίνησης, ο αριθμός και οι ακριβείς διαμορφώσεις των αναγκαίων δικτυακών συσκευών (αναμεταδότες, αναγεννητές), το συνολικό κόστος υλοποίησης για την προτεινόμενη λύση, η συνολική απαιτούμενη ενέργεια για τη λειτουργία του δικτύου καθώς και οι συνδέσεις που δεν μπορούν να εξυπηρετηθούν είτε λόγω φυσικών εξασθενήσεων είτε λόγω μη διαθέσιμου εύρους ζώνης.

Name	Configurations	Experiments	Topologies
Testing	31	4	5
Default	11	0	10

5.3 Η αρχική σελίδα του Mantis και οι βασικές περιοχές της γραφικής διεπαφής

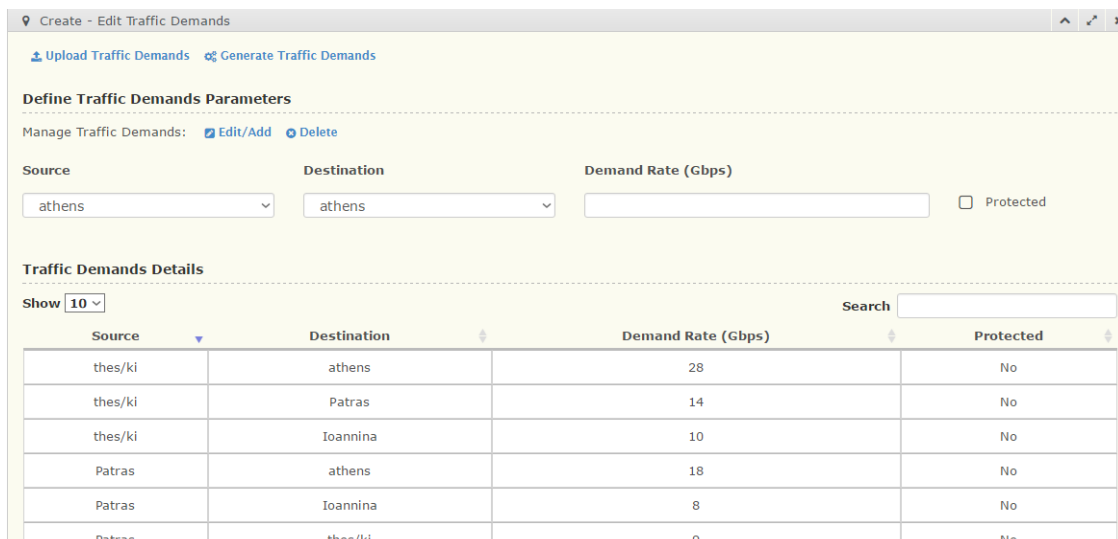
Η διεπαφή επιτρέπει στους χρήστες να σχεδιάζουν και να επεξεργάζονται δικτυακές τοπολογίες καθορίζοντας τους κόμβους και τους συνδέσμους μαζί με τα αντίστοιχα μήκη τους. Το **Error! Reference source not found.** απεικονίζει τη διαθέσιμη γραφική διεπαφή για την επεξεργασία δικτυακών τοπολογιών. Οι χρήστες μπορούν να δημιουργούν και να επεξεργάζονται εύκολα τις τοπολογίες μιας και κάθε κόμβος ή σύνδεσμος παρέχει ένα μενού με τις διαθέσιμες ενέργειες. Επιλέγοντας κάποιον κόμβο οι χρήστες μπορούν να επεξεργαστούν το όνομά του, να το διαγράψουν, να προσθέσουν ένα σύνδεσμο που ξεκινά από το συγκεκριμένο κόμβο ή να προσθέσουν ένα νέο κόμβο στην τοπολογία. Ομοίως, υπάρχει διαθέσιμο ένα μενού για την αλληλεπίδραση με τους συνδέσμους της

τοπολογίας, το οποίο παρέχει στους χρήστες τη δυνατότητα να προσθέτουν ένα σύνδεσμο, να τροποποιούν το μήκος ή να το διαγράφουν.

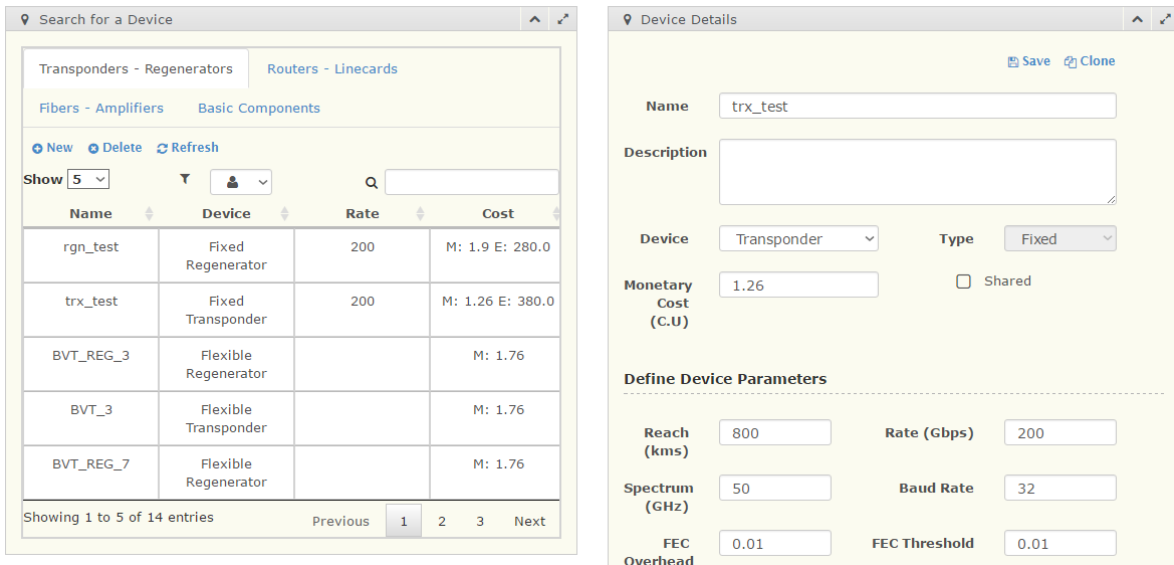


5.4. Δημιουργία και διαχείριση δικτυακών τοπολογιών

Οι απαιτήσεις κίνησης μπορούν να οριστούν είτε γραφικά μέσω της αντίστοιχης γραφικής διεπαφής (**Error! Reference source not found..5**) είτε μπορούν να φορτωθούν από αρχεία στα οποία περιγράφονται ακολουθώντας το πρότυπο Comma-Separated Values (CSV). Ακόμη, οι χρήστες μπορούν να καθορίζουν τις διάφορες δικτυακές συσκευές που θα είναι διαθέσιμες για την δημιουργία τοπολογιών αλλά και το σχεδιασμό των δικτύων. Η αντίστοιχη διεπαφή (**Error! Reference source not found.**) παρουσιάζει στους χρήστες τις διαθέσιμες συσκευές, ενώ παράλληλα επιτρέπει την τροποποίηση των παραμέτρων τους (χαρακτηριστικά λειτουργίας, ενεργειακό κόστος λειτουργίας, κόστος αγοράς) ή τον ορισμό νέων συσκευών. Οι αλγόριθμοι αξιοποιούν τις συγκεκριμένες πληροφορίες προκειμένου να υπολογίσουν το συνολικό κόστος υλοποίησης (CAPEX), λειτουργίας (OPEX) και την ενεργειακή κατανάλωση για κάθε λύση που προτείνουν.



5.5 Δημιουργία και διαχείριση απαιτήσεων κυκλοφορίας για δεδομένη δικτυακή τοπολογία



The screenshot displays two windows from a network management application. The left window, titled 'Search for a Device', shows a list of devices with columns for Name, Device, Rate, and Cost. The right window, titled 'Device Details', shows the configuration for a device named 'trx_test'.

Name	Device	Rate	Cost
rgn_test	Fixed Regenerator	200	M: 1.9 E: 280.0
trx_test	Fixed Transponder	200	M: 1.26 E: 380.0
BVT_REG_3	Flexible Regenerator		M: 1.76
BVT_3	Flexible Transponder		M: 1.76
BVT_REG_7	Flexible Regenerator		M: 1.76

The 'Device Details' window for 'trx_test' includes the following fields:

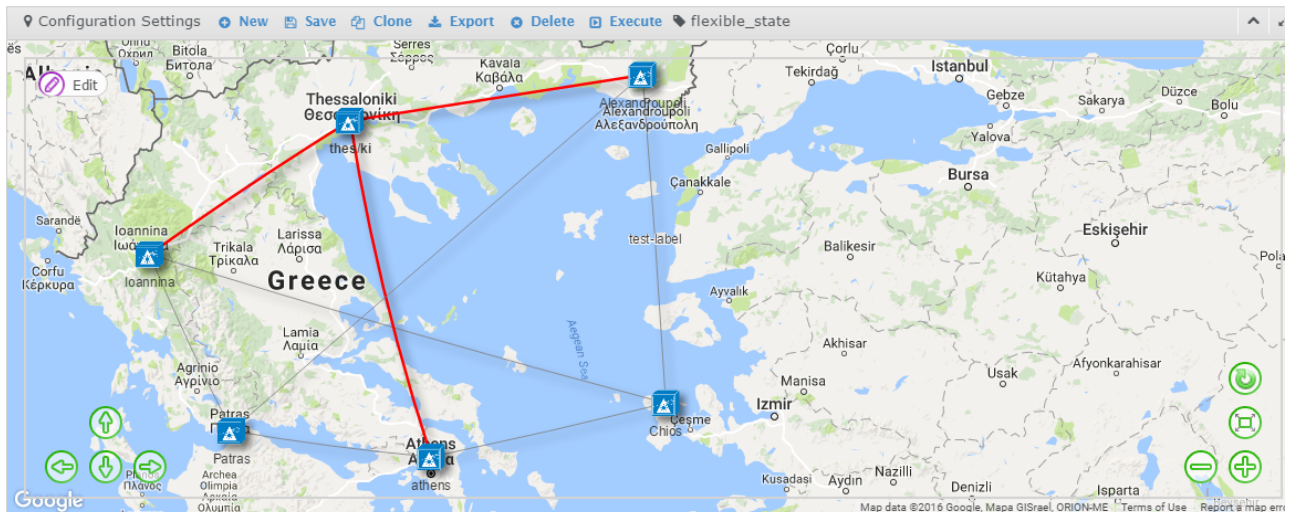
- Name: trx_test
- Description: (empty)
- Device: Transponder
- Type: Fixed
- Monetary Cost (C.U): 1.26
- Shared: (unchecked)
- Define Device Parameters:
 - Reach (kms): 800
 - Rate (Gbps): 200
 - Spectrum (GHz): 50
 - Baud Rate: 32
 - FEC Overhead: 0.01
 - FEC Threshold: 0.01

5.6 Ορισμός και διαχείριση δικτυακών συσκευών

Για την δημιουργία μιας περιγραφής πειράματος αρχικά είναι να αναγκαίο να επιλέξουμε κάποια από τις υπάρχουσες δικτυακές τοπολογίες ή να δημιουργήσουμε μια νέα τοπολογία. Επίσης στο αναδυόμενο παράθυρο ο χρήστης προσδιορίζει το όνομα, τη περιγραφή και επιλέγει μερικά βασικά στοιχεία για τη συγκεκριμένη περιγραφή πειράματος.

Στη συνέχεια, πατώντας το κουμπί δημιουργία μεταβαίνει στο βασικό χώρο εργασίας (**Error! Reference source not found.**) για να συνεχίσει την περαιτέρω επεξεργασία της περιγραφής και την εκτέλεση του πειράματος. Στο χώρο εργασίας διακρίνεται η βασική περιοχή εργασίας που φιλοξενεί την γραφική αναπαράσταση της τοπολογίας με το διαθέσιμο οριζόντιο μενού για την δημιουργία νέας περιγραφής πειράματος, την αποθήκευση, την εκτέλεση και την αντιγραφή της προς επεξεργασία περιγραφής. Στο οριζόντιο μενού εμφανίζονται επίσης μερικά βασικά στοιχεία της περιγραφής. Επιπλέον, υπάρχουν οι επιμέρους καρτέλες, κάτω από την τοπολογία, μέσω των οποίων προσδιορίζονται όλα τα απαραίτητα στοιχεία μιας περιγραφής πειράματος. Στο παράδειγμα που παραθέτουμε παρουσιάζονται οι βασικές πληροφορίες για την συγκεκριμένη περιγραφή που μπορούμε να επεξεργαστούμε.

Οι χρήστες μπορούν μέσω της κατάλληλης διεπαφής (**Error! Reference source not found.**) να ελέγχουν την κατάσταση των πειραμάτων που εκτελούνται ή που έχουν ήδη ολοκληρωθεί και να έχουν πρόσβαση σε χρήσιμα στοιχεία όπως το όνομα της περιγραφής πειράματος, την τρέχουσα κατάσταση και την ημερομηνία εκτέλεσης. Επιπλέον, μπορούν να τερματίσουν την εκτέλεση κάποιου πειράματος και να έχουν πρόσβαση στα αποτελέσματα όσων έχουν ολοκληρωθεί επιτυχώς ή να φιλτράρουν τα διαθέσιμα πειράματα με βάση διάφορα κριτήρια. Για τις περιγραφές πειραμάτων που έχουν ολοκληρωθεί επιτυχώς οι χρήστες μπορούν να έχουν πρόσβαση στα αναλυτικά αποτελέσματα της κάθε εκτέλεσης καθώς και να εξάγουν την προτεινόμενη λύση που περιλαμβάνει αναλυτικά στοιχεία για κάθε σύνδεση για περαιτέρω ανάλυση.



Configuration Settings **flexible_state**

Name
Description
Project

Wavelengths
Granularity
Architecture Flexible

5.7 Βασικός χώρος εργασίας (Workspace) για την δημιουργία μιας περιγραφής πειράματος

Experiments Details

Show
 Status
 Projection
 Search

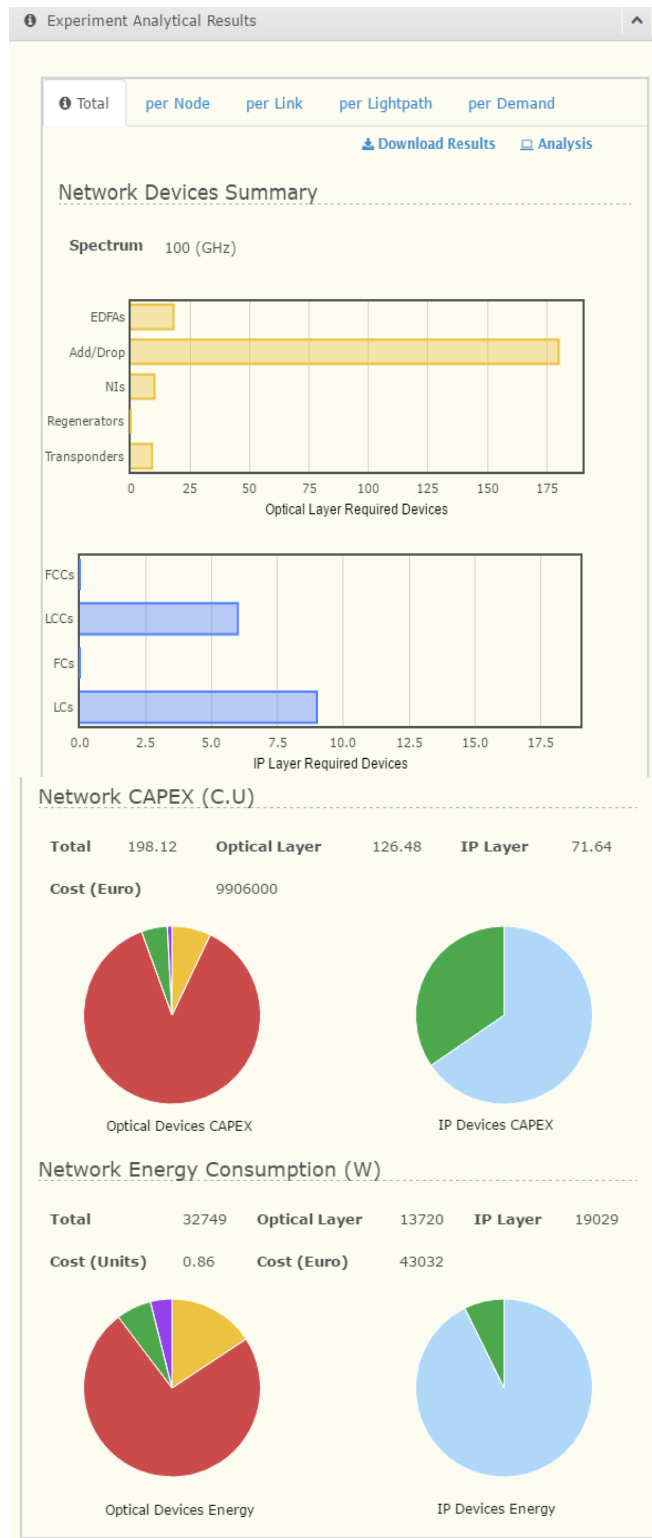
	Name	Configuration	Project	Status	Reason	Created
<input type="checkbox"/>	test_multiperiod_14 81899611	test_multiperiod	Testing	Done	COMPLETED	16/12/16 16:46:51
<input type="checkbox"/>	OptimalRouting_148 1712844	OptimalRouting	Testing	Done	COMPLETED	14/12/16 12:54:04
<input type="checkbox"/>	OptimalRouting_148 1712747	OptimalRouting	Testing	Done	COMPLETED	14/12/16 12:52:27
<input type="checkbox"/>	6-node- IP&Optical_14817073 94	6-node-IP&Optical	Testing	Done	COMPLETED	14/12/16 11:23:14
<input type="checkbox"/>	optical_translucent_ 1481705982	optical_translucent	Testing	Done	COMPLETED	14/12/16 10:59:42
<input type="checkbox"/>	optical_translucent_ 1481221548	optical_translucent	Testing	Done	COMPLETED	08/12/16 20:25:48
<input type="checkbox"/>	IPoFlex_1480951929	IPoFlex	Testing	Done	COMPLETED	05/12/16 17:32:09
<input type="checkbox"/>	increment_test_1480	increment_test	Testing	Done	COMPLETED	29/11/16 12:03:03

5.8 Διεπαφή επισκόπησης της κατάστασης των πειραμάτων και των αποτελεσμάτων τους

Για τις εργασίες που έχουν ολοκληρωθεί επιτυχώς οι χρήστες μπορούν να έχουν πρόσβαση στα αναλυτικά αποτελέσματα της κάθε εκτέλεσης καθώς και

να εξάγουν την προτεινόμενη λύση που περιλαμβάνει αναλυτικά στοιχεία για κάθε σύνδεση για περαιτέρω ανάλυση. Η εφαρμογή επιστρέφει στους χρήστες αναλυτικές λύσεις για κάθε διαθέσιμο αλγόριθμο, όπου κατά περίπτωση μπορεί να περιλαμβάνουν:

- το αναγκαίο εύρος ζώνης ώστε να εξυπηρετηθούν οι απαιτήσεις κίνησης
- τα χρησιμοποιούμενα οπτικά μονοπάτια
- το πλήθος και τις διαμορφώσεις για τους απαραίτητους αναμεταδότες και αναγεννητές
- τους κόμβους που πρέπει να τοποθετηθούν οι αναμεταδότες και αναγεννητές
- το συνολικό κόστος υλοποίησης και τη συνολική απαιτούμενη ενέργεια για τη λειτουργία της προτεινόμενης λύσης
- τις συνδέσεις που δεν μπορούν να εξυπηρετηθούν είτε λόγω φυσικών εξασθενήσεων είτε λόγω έλλειψης εύρους ζώνης



5.9. Απόσπασμα αναλυτικής λύσης, περιλαμβάνει τους απαιτούμενους αναμεταδότες, αναγεννητές καθώς και ανάλυση του συνολικού κόστους

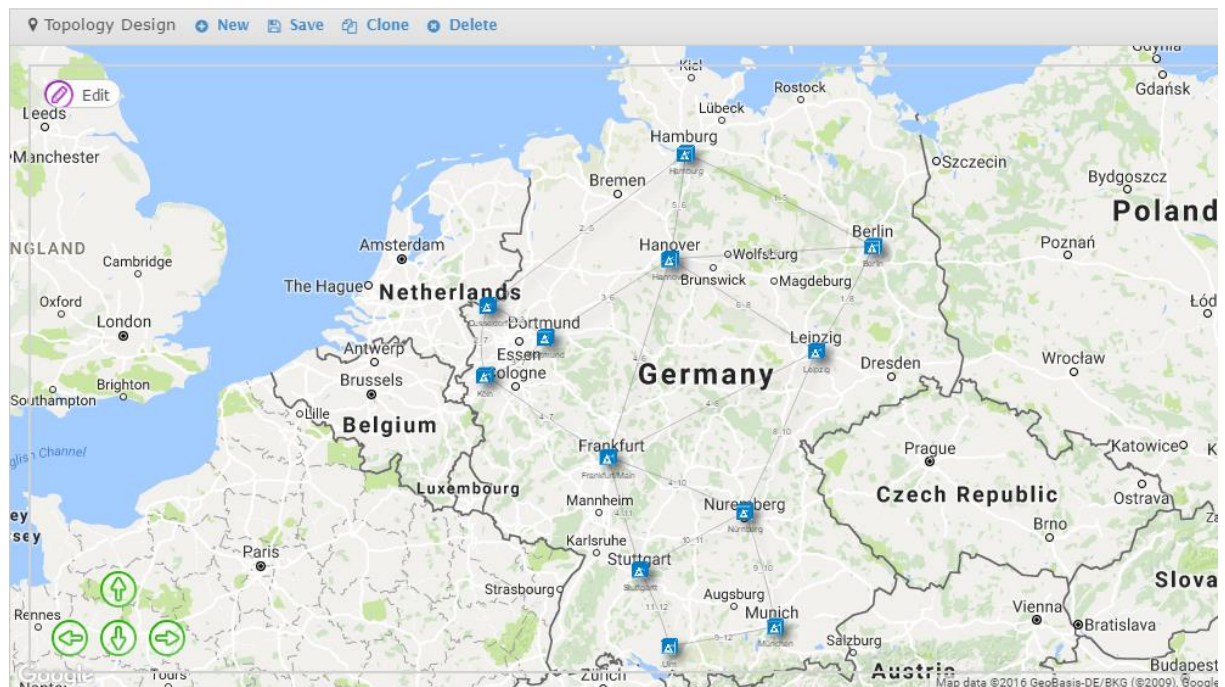
5.4 Εκτέλεση σεναρίου αξιολόγησης

Στη συνέχεια εκτελέσαμε ένα απλό σενάριο για να αξιολογήσουμε την απόδοση του νέου αλγορίθμου βέλτιστης δρομολόγησης. Πιο συγκεκριμένα, συγκρίναμε τον αλγόριθμο projection method βέλτιστης δρομολόγησης (που παρουσιάσαμε στο κεφάλαιο 3.3) για τη δρομολόγηση στο IP επίπεδο ενός οπτικού δικτύου με τον αλγόριθμο IP over Flex (IPoFlex) που σχεδιάζει από κοινού το IP και το οπτικό επίπεδο.

Ο αλγόριθμος IPoFlex εξετάζει το πρόβλημα σχεδιασμού ενός ευέλικτου οπτικού δικτύου πολλαπλών επιπέδων (multi-layer), στοχεύοντας στην ελαχιστοποίηση του χρησιμοποιούμενου φάσματος και του κόστους εγκατάστασης του δικτύου. Για την επίλυση του ενοποιημένου προβλήματος σχεδιασμού δικτύου πολλαπλών επιπέδων χρησιμοποιεί μοντελοποίηση Ακέραιου Γραμμικού Προγραμματισμού, (Integer Linear Programming – ILP) η οποία είναι γενική και μπορεί να χρησιμοποιηθεί για δίκτυα ευέλικτου ή σταθερού φάσματος. Το μοντέλο κόστους που υιοθετήσαμε στα πλαίσια του αλγορίθμου θεωρεί IP/MPLS δρομολογητές κλιμακωτής αρχιτεκτονικής οι οποίοι συνδέονται στις παρυφές του οπτικού δικτύου με οπτικούς πομποδέκτες (transponders) μεταβλητού ή σταθερού φάσματος.

Στον αλγόριθμο βέλτιστης δρομολόγησης περνάμε σαν είσοδο τα οπτικά μονοπάτια που αποφασίζει ο IPoFlex αλγόριθμος. Αυτοί οι εικονικοί σύνδεσμοι είναι οι σύνδεσμοι που βλέπει ο αλγόριθμος βέλτιστης δρομολόγησης. Ο αλγόριθμος αποφασίζει τη δρομολόγηση της IP κίνησης πάνω από αυτούς τους εικονικούς συνδέσμους με στόχο την ελαχιστοποίηση της χρησιμοποίησης των εικονικών αυτών συνδέσμων.

Η δικτυακή τοπολογία (**Error! Reference source not found.**) που χρησιμοποιούμε αντιστοιχεί στο εθνικό δίκτυο της Γερμανίας (Deutsche Telecom) το οποίο αποτελείται από 12 κόμβους και από 40 διπλούς συνδέσμους μονής κατεύθυνσης. Επιπλέον, οι απαιτήσεις κίνησης (**Error! Reference source not found.**) που χρησιμοποιήσαμε αντιστοιχούν στην πραγματική κίνηση που είχε να διαχειριστεί το συγκεκριμένο δίκτυο το 2014.



5.10. Δικτυακή τοπολογία για τον έλεγχο ενσωμάτωσης του αλγορίθμου στο Mantis

Traffic Demands Details

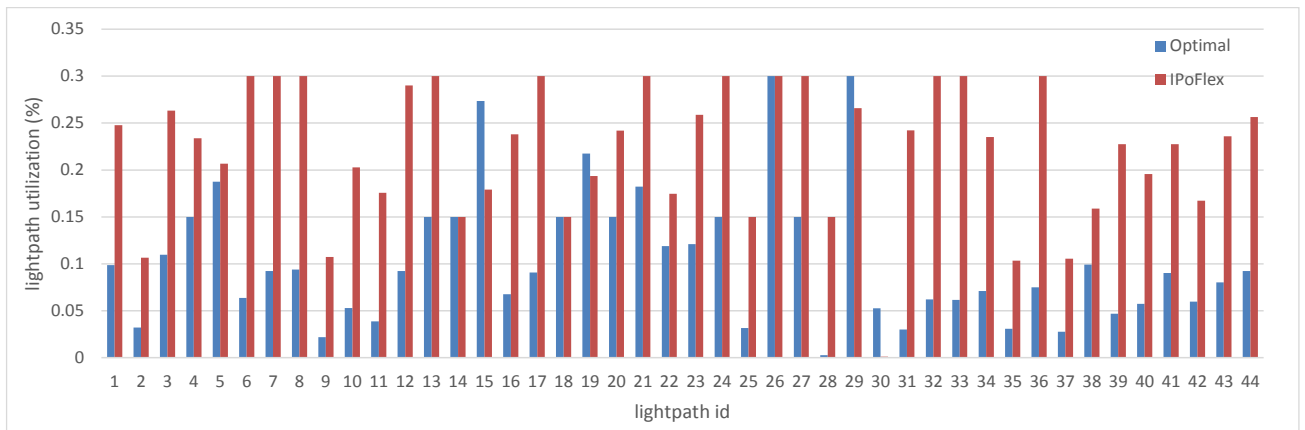
Show Search

Source	Destination	Demand Rate (Gbps)	Protected
Berlin	Dusseldorf	19.599	No
Berlin	Dortmund	3.602	No
Berlin	Frankfurt/Main	68.364	No
Berlin	Hamburg	13.131	No
Berlin	Hannover	4.682	No
Berlin	Köln	1.644	No
Berlin	Leipzig	7.061	No
Berlin	München	13.232	No
Berlin	Nürnberg	26.761	No
Berlin	Stuttgart	4.717	No

Showing 1 to 10 of 132 entries Previous 2 3 4 5 ... 14 Next

5.11. Απαιτήσεις κίνησης για το σενάριο αξιολόγησης

Αρχικά, για την εκτέλεση του τελικού ελέγχου ενσωμάτωσης και αξιολόγησης της απόδοσης δημιουργήσαμε μέσω της web-based διεπαφής του Mantis την κατάλληλη δικτυακή τοπολογία και τις αντίστοιχες απαιτήσεις κίνησης. Στη συνέχεια, δημιουργήσαμε για κάθε διαθέσιμο αλγόριθμο μια περιγραφή πειράματος (Configuration) χρησιμοποιώντας τη τοπολογία αναφοράς και επιλέγοντας κατάλληλα τις επιμέρους παραμέτρους εισόδου του αλγορίθμου. Πλέον, έχοντας δημιουργήσει τις απαιτούμενες περιγραφές προχωρήσαμε στην εκτέλεση κάθε επιμέρους αλγορίθμου μέσω της web-based διεπαφής.



5.1: Ποσοστό χρησιμοποίησης των οπτικών μονοπατιών (lightpaths)

Στο παραπάνω σχήμα παρατηρούμε το ποσοστό χρησιμοποίησης των οπτικών μονοπατιών (lightpaths) για τις δύο μεθόδους που εξετάσαμε: τη μέθοδο βέλτιστης δρομολόγησης για το IP επίπεδο και τη μέθοδο από κοινού σχεδιασμού του οπτικού και του IP και του οπτικού δικτύου (IPoFlex). Παρατηρούμε ότι ο αλγόριθμος βέλτιστης δρομολόγησης επιτυγχάνει μικρότερο ποσοστό χρησιμοποίησης των περισσότερων από τα οπτικά μονοπάτια.

6

Βιβλιογραφία

- 1) L. Chiaraviglio, M. Mellia, F. Neri, Minimizing isp network energy cost: formulation and solutions, IEEE/ACM Trans Netw (TON) 20 (2012) 463–476.
- 2) L. Fratta, M. Gerla, L. Kleinrock, The flow deviation method: an approach to store-and-forward communication network design, Networks 3 (2) (1973) 97–133.
- 3) Wooseong Kim, Mario Gerla, NAVOPT: Navigator Assisted Vehicular route OPTimizer, in: IMIS, June 2011.
- 4) Flow Deviation: 40 years of incremental flows for packets, waves, cars and tunnels , Luigi Fratta , Mario Gerla , Leonard Kleinrock
- 5) Data Networks, Dimitri Bertsekas, Robert Gallager, 2nd edition
- 6) Δικτύωση υπολογιστών: Προσέγγιση από Πάνω προς τα Κάτω, KUROSE | ROSS, 6^η έκδοση
- 7) A. Kretsis, K. Christodoulopoulos, P. Kokkinos, E. Varvarigos, "Planning and Operating Flexible Optical Networks: Algorithmic Issues and Tools", IEEE Communications Magazine, special issue on Advances in Network Planning, Vol. 52(1), pp. 61-69, 2014.
- 8) P. Papanikolaou, K. Christodoulopoulos, E. Varvarigos, "Multilayer flex-grid network planning", International Conference on Optical Network Design and Modeling, (ONDM), 2015.

Ηλεκτρονική βιβλιογραφία

[1] NAVOPT : <http://nrlweb.cs.ucla.edu/publication/download/791/navopt.pdf>

[2] **Energyconsumption:**

<http://www.telematica.polito.it/oldsite/chiaraviglio/papers/GreenTon.pdf>

[3] **Optical networks - WDM :**

<http://web.mit.edu/modiano/www/papers/J15.pdf>

[4] ARPANET Wikipedia : <https://en.wikipedia.org/wiki/ARPANET>

[5] **Routing in Data nets :**

http://web.mit.edu/dimitrib/www/Routing_Data_Nets.pdf

[6] Geant2 : <http://geant2.archive.geant.net>

[7] **Qt - Cross-platform software development for embedded & desktop:**

<https://www.qt.io>

ΠΑΡΑΡΤΗΜΑ Α

Flow deviation method

```
tic

nodes=length(G);
delay_prop=0;

%Δημιουργώ τους k_paths, alt_path
%k_paths: περιέχει όλα τα διαθέσιμα disjoint μονοπάτια για κάθε OD
pair
%alt_path: πλήθος αυτών των διαθέσιμων μονοπατιών (k)
k_paths=zeros(nodes, nodes);
alt_path=zeros(nodes,1);
m=1;
x1=0;
for i=1:nodes
    for j=1:nodes
        if i~=j
            %ο k_paths1 είναι βοηθητικός (=είναι ο k_paths για ένα OD
pair)
            [k_paths1,k]=k_paths_algo_DISJ_PATHS(G,km,i,j);
            alt_path(m)=k;
            m=m+1;
            for x=1:k
                for y=1:nodes

                    k_paths(x1+x,y)=k_paths1(x,y);

                end
            end
            x1=x1+k;
        end
    end
end
alt_path=nonzeros(alt_path);
k_paths;

%ftiaxnw to arxiko x_p
[n,m]=size(k_paths);

x_p=zeros(n,1);

tr1=zeros(n,1); %(βοηθητικός): περιέχει την κίνηση για κάθε OD pair
του k_paths
for i=1:n
    j=m;
    while k_paths(i,j)==0
        j=j-1;
    end
    tr1(i)=tr_matrix(k_paths(i,1),k_paths(i,j));
end
%
%χωρίζω την κίνηση ισομερώς στα k διαθέσιμα paths
%x_p: το αρχικό διάνυσμα ρών
```

```

l=0;
i=1;
while i<=n
    l=l+1;
    k=1;
    while k<=alt_path(l)
        x_p(i)=(1/alt_path(l))*tr1(i);
        k=k+1;
        i=i+1;
    end

end

x_arx=x_p;
if sum(sum(tr_matrix))>sum(sum(capacity))
    disp ('no feasible solution')

end

%αρχικοποιώ τους D:κόστος Kleinrock
%           D_1_der: πρώτη παράγωγος (για κάθε link)
%           D_2_der: δεύτερη παράγωγος (για κάθε link)

a_int=Ypologismos_a(F, capacity);
D=zeros(nodes, nodes);
D_1_der=zeros(nodes, nodes);
D_2_der=zeros(nodes, nodes);

while a_int<1
    disp ('non feasible initial solution')
    x_p=x_p*a_int;

    %υπολογίζω κόστη-παραγωγούς σύμφωνα με kleinrock
    for i=1:nodes
        for j=1:nodes

            if F(i,j)>0
                D(i,j)=(F(i,j)/(capacity(i,j)-F(i,j))) +
(delay_prop*F(i,j));
                D_1_der(i,j)= capacity(i,j)/((capacity(i,j)-
F(i,j))^2)+delay_prop;
                D_2_der(i,j)= (2*capacity(i,j))/((capacity(i,j)-F(i,j))^3);
            end
        end
    end
sum(sum(D));

D_1_path=zeros(n,1);%πρώτη παράγωγος του path (άθροισμα των πρώτων
παραγωγών των links του path)
x_p_bar=zeros(n,1); %διάνυσμα ρών με δρομολόγηση όλης της ροής στο
mfd path κάθε OD
for i=1:n
    d=m;
    while k_paths(i,d)==0 % βρίσκω origin (i) - destination (j) για
κάθε path
        d=d-1; %
    end %
    for j=1:(d-1)

```



```

    D_1_path(i)=D_1_path(i)+D_1_der(k_paths(i,j),k_paths(i,j+1));
end
end
D_1_path;
k=0;
i=1;
while i<=length(alt_path)
    k=k+1; %
    min=D_1_path(k); % βρίσκω το min forst derivative path και
την θέση του
    thesis=k;
    l=1;
    while l<alt_path(i) % για όλα τα od το min ανάμεσα στα
διαθέσιμα alt_path(i) paths
        k=k+1; %
        if D_1_path(k)<min
            thesis=k;
        end
        l=l+1;
    end
    x_p_bar(thesis)=trl(thesis);
    i=i+1;
end
x_p_bar;
%υπολογίζω utilization των inks σύμφωνα με x_p_bar
F_bar=F_util(x_p_bar,k_paths);
%υπολογίζω παράμετρο α
a1=sum(sum((F_bar-F).*D_1_der)); %αριθμητης
a2=sum(sum(((F_bar-F).*(F_bar-F)).*D_2_der)); %παρονομαστης
a3= - (a1/a2);
a=1; %min(1,a3);
if a3<a
    a=a3;
end
%υπολογίζω το νέο διάνυσμα ρών
x_p=x_p+ (a*(x_p_bar-x_p));
%
F=F_util(x_p,k_paths);
%θα ελέξω εάν είναι ακόμη non feasible
a_int=Ypologismos_a(F,capacity);

end

steps=1;
while steps<150

F=F_util(x_p,k_paths);

%υπολογίζω κόστη με βάση kleinrock
D_1_der=zeros(nodes,nodes);
D_2_der=zeros(nodes,nodes);

for i=1:nodes
    for j=1:nodes

        if F(i,j)>0
            D(i,j)=(F(i,j)/(capacity(i,j)-F(i,j))) +
(delay_prop*F(i,j));
            D_1_der(i,j)= capacity(i,j)/((capacity(i,j)-
F(i,j))^2)+delay_prop;

```

```

        D_2_der(i,j)= (2*capacity(i,j))/((capacity(i,j)-F(i,j))^3);
    end
end
end
sum(sum(D))

%vriskw to x_p_bar

D_1_path=zeros(n,1);
x_p_bar=zeros(n,1);
for i=1:n
    d=m;
    while k_paths(i,d)==0
        d=d-1;
    end
    for j=1:(d-1)
        D_1_path(i)=D_1_path(i)+D_1_der(k_paths(i,j),k_paths(i,j+1));
    end
end
D_1_path;
k=0;
i=1;
while i<=length(alt_path)
    k=k+1;
    min=D_1_path(k);
    thesis=k;
    l=1;
    while l<alt_path(i)
        k=k+1;
        if D_1_path(k)<min
            thesis=k;
        end
        l=l+1;
    end
    x_p_bar(thesis)=trl(thesis);
    i=i+1;
end
x_p_bar;
%ypologizw F_bar
F_bar=F_util(x_p_bar,k_paths);
%ipologizw to a
a1=sum(sum((F_bar-F).*D_1_der));
a2=sum(sum(((F_bar-F).*(F_bar-F)).*D_2_der));
a3= - (a1/a2);
a=1;
if a3<a
    a=a3;
end
%ipologizw to neo xp
x_p=x_p+ (a*(x_p_bar-x_p));
steps=steps+1;
end
sum(sum(D))
max(max(F))
steps=steps-1

```

Projection method

```
tic

nodes=length(G);
%α(k)=par = {1,0.5,0.25,0.125}
lisi=zeros(4,1);
par=1;
delay_prop=0;

%Δημιουργώ τους k_paths, alt_path
%k_paths: περιέχει όλα τα διαθέσιμα disjoint μονοπάτια για κάθε OD
pair
%alt_path: πλήθος αυτών των διαθέσιμων μονοπατιών (k)
k_paths=zeros(nodes, nodes);
alt_path=zeros(nodes,1);
m=1;
x1=0;
for i=1:nodes
    for j=1:nodes
        if i~=j
            %ο k_paths1 είναι βοηθητικός (=είναι ο k_paths για ένα OD
pair)
            [k_paths1,k]=k_paths_algo_DISJ_PATHS(G,km,i,j);
            alt_path(m)=k;
            m=m+1;
            for x=1:k
                for y=1:nodes

                    k_paths(x1+x,y)=k_paths1(x,y);

                end
            end
            x1=x1+k;
        end
    end
end
alt_path=nonzeros(alt_path);
k_paths;

%ftiaxnw to arxiko x_p
[n,m]=size(k_paths);

x_p=zeros(n,1);

tr1=zeros(n,1); %(βοηθητικός): περιέχει την κίνηση για κάθε OD pair
του k_paths
for i=1:n
    j=m;
    while k_paths(i,j)==0
        j=j-1;
    end
    tr1(i)=tr_matrix(k_paths(i,1),k_paths(i,j));
end
%
%χωρίζω την κίνηση ισομερώς στα k διαθέσιμα paths
```

```

%x_p: το αρχικό διάνυσμα ροών
l=0;
i=1;
while i<=n
    l=l+1;
    k=1;
    while k<=alt_path(l)
        x_p(i)=(1/alt_path(l))*tr1(i);
        k=k+1;
        i=i+1;
    end
end

x_arx=x_p;
if sum(sum(tr_matrix))>sum(sum(capacity))
    disp ('no feasible solution')
end

%αρχικοποιώ τους D:κόστος Kleinrock
%           D_1_der: πρώτη παράγωγος (για κάθε link)
%           D_2_der: δεύτερη παράγωγος (για κάθε link)
%           d_p_w_1: πρώτη παράγωγος του minimum first
derivative path
%           (αναφέρεται σε paths)

D=zeros(nodes,nodes);
D_1_der=zeros(nodes,nodes);
D_2_der=zeros(nodes,nodes);
d_p_w_1=zeros(n,1);

%x_p_k: το νέο διάνυσμα ροών
x_p_k=zeros(n,1);

for s=1:4 %par=(1, 0.5, 0.25, 0.125)

steps=1;
while steps<2
    if steps==1
        x_p=x_arx;
    else
        x_p=x_p_k;
    end

    %utilization των links
    F=F_util(x_p,k_paths);

    %υπολογίζω κόστιη-παραγωγούς σύμφωνα με kleinrock
    for i=1:nodes
        for j=1:nodes
            if F(i,j)>0

                D(i,j)=(F(i,j)/(capacity(i,j)-F(i,j))) + (delay_prop*F(i,j));
                D_1_der(i,j)= capacity(i,j)/((capacity(i,j)-
F(i,j))^2)+delay_prop;
                D_2_der(i,j)= (2*capacity(i,j))/((capacity(i,j)-F(i,j))^3);
            end
        end
    end
end

```

```

end

D_1_path=zeros(n,1); %πρώτη παράγωγος του path (άθροισμα των πρώτων
παράγωγων των links του path)
D_2_path=zeros(n,1);
for i=1:n
    d=m; % βρίσκω origin (i) - destination (j) για κάθε
path
    while k_paths(i,d)==0 %
        d=d-1; %
    end
    for j=1:(d-1)
        D_1_path(i)=D_1_path(i)+D_1_der(k_paths(i,j),k_paths(i,j+1));
%αθροίζω τα links κάθε path
        D_2_path(i)=D_2_path(i)+D_2_der(k_paths(i,j),k_paths(i,j+1));
    end
end

h_p_w=zeros(n,1); % άθροισμα δευτέρων παραγώγων του mdf για κάθε od
pair
d_p=D_1_path;
k=0;
i=1;
thesis=zeros(length(alt_path),1); %θέση των mfd μονοπατιών στον
k_paths για κάθε OD pair
while i<=length(alt_path)
    k=k+1;
    min=D_1_path(k);
    thesis(i)=k;
    l=1;
    while l<alt_path(i) %ψάχνω το min σε κάθε set k διαθέσιμων
μονοπατιών
        k=k+1;
        if D_1_path(k)<min
            thesis(i)=k;
            min=D_1_path(k);
        end
        l=l+1;
    end
    for j=(k-alt_path(i)+1):k
        d_p_w_1(j)=min;
        a=m;
        while k_paths(thesis(i),a)==0
            a=a-1;
        end
        for b=1:(a-1)
            h_p_w(j)=h_p_w(j)+D_2_der(k_paths(thesis(i),b),k_paths(thesis(i),b+1)
);
        end
    end
    i=i+1;
end

H_p=h_p_w+D_2_path; %επειδή τα paths είναι disjoint, μόνο αθροίζω

```

```

for i=1:n % για όλα τα paths που δεν είναι mdf (τα mdf τα μηδενίζω
αρχικά)
    H_p(i)=1/H_p(i);
    a1(i)=par*H_p(i)*(d_p(i)-d_p_w_1(i));
    if (x_p(i)-a1(i))>0
        x_p_k(i)=x_p(i)-a1(i);
    else
        x_p_k(i)=0;
    end
end

%υπολογίζω το υπόλοιπο της ροής προκειμένου να υπολογίσω τη ροή στα
mdf
%μονοπάτια
f=x_p-x_p_k;
k=1;
for i=1:length(alt_path)
    l=1;
    sum_rois=0; %άθροισμα υπολοίπου ροής για κάθε OD pair
    while l<=alt_path(i)
        sum_rois =sum_rois+ f(k);

        l=l+1;
        k=k+1;
    end

    x_p_k(thesis(i))=sum_rois+x_p_k(thesis(i)); %ροή για τα mdf
μονοπάτια
end

steps=steps+1;

end
lisi(s)=sum(sum(D))

par=par/2;
end
min=+inf;
for i=1:4
    if lisi(i)>0 & lisi(i)<min %κρατάω μόνο τις θετικές λύσεις
(αρνητικά έχω όταν το par θέτει τη λύση non feasible
        min=lisi(i);
    end
end
min
max(max(F))
steps=steps-2
toc

```

k_paths function

```
function [ k_paths, k ] =k_paths_algo_DISJ_PATHS( G,km,s,dest )
```

```

[n,m]=size(G);

nodes=n;
k1max=sum(G(s,1:nodes));
k2max=sum(G(dest,1:nodes));
kmax= max(k1max,k2max);

newkm=zeros(nodes,nodes);
k_paths=zeros(kmax,nodes);
y= zeros(kmax,1);
[a]=Dijkstra_path2(G,km,s,dest);
for j=1:length(a)
    k_paths(1,j)=a(j);
end
for i=2:kmax
    [x,y]=find(k_paths==dest);
    y2=zeros(nodes);
    for l=1:i-1
        b= find(x==l);
        y(b);
        y2(l)=y(b);
    end

    for j=1:(y2(i-1,1)-1)

        G(k_paths(i-1,j),k_paths(i-1,j+1))=0;
        G(k_paths(i-1,j+1),k_paths(i-1,j))=0;
    end
    [a]=Dijkstra_path2(G,km,s,dest);
    for j=1:length(a)
        k_paths(i,j)=a(j);
    end
end
k=kmax;
while (k_paths(k,1)==s) & (k_paths(k,2)==dest) & (k>1)

k=k-1;
end
k_paths=k_paths(1:k,1:nodes);

end

```

Dijkstra_path function

```

function [ path ] = Dijkstra_path2( G, km, s, dest )
[n,m]=size(G);
nodes=n;
%initialization
P=zeros(nodes,nodes);
w=zeros(nodes,1);
w(1)=s;
mincost=zeros(nodes,1);

for i=1:nodes
    for j=1:nodes
        D(i,j)=inf;
    end
    P(1,i)=s;
    if G(s,i)==1

```

```

        D(1,i)=km(s,i);
    elseif i==s
        D(1,i)=inf;
    else
        D(1,i)=10000;
    end
end
for i=2:nodes

M = min(D,[],2);
[a,b]=find(D==M(i-1));
thesis=find(a==(i-1));
w(i)=b(thesis(1));
mincost(i-1)= M(i-1);
D(i,w(i))=inf;
for j=1:nodes
    if j==w(i)
        D(i,w(i))=inf;
    elseif D(i-1,j)==inf
        D(i,j)=inf;
    else
        if G(w(i),j)==1

            if D(i-1,j)> mincost(i-1) + km(w(i),j)

D(i,j)=mincost(i-1) + km(w(i),j);
P(i,j)=w(i);
                else
                    D(i,j)=D(i-1,j);
                end
            else
                D(i,j)=D(i-1,j);
            end
        end
    end
end
end
path=zeros(nodes,1);
path(nodes)=dest;
m=dest;
j=1;

while m~=s
    i=nodes;
    while P(i,m)==0
        i=i-1;
    end
    path(nodes-j)=P(i,m);
    m=P(i,m);
    j=j+1;
end
path=nonzeros(path);
end

```

F_util function

```

function [ F ] = F_util( x_p,k_paths )
[n,m]=size(k_paths);

```



```

F=zeros(m,m);
for i=1:n
    d=m;
    while k_paths(i,d)==0
        d=d-1;
    end
    for j=1:(d-1)
        F(k_paths(i,j),k_paths(i,j+1))=F(k_paths(i,j),k_paths(i,j+1))+x_p(i);
    end
end
F;
end

```

Ypologismos_a function (fon non feasible initial solutions)

```

function [ a ] = Ypologismos_a( F,C )
k=length(F);
a_n=zeros(k,k);
for i=1:k
    for j=1:k
        if C(i,j)>0
            a_n(i,j)=C(i,j)/F(i,j);
        end
    end
end
a_n=nonzeros(a_n);
a=min(min(a_n));
end

```

ΠΑΡΑΡΤΗΜΑ Β

1^η τοπολογία δικτύου

G =

0	1	1	1	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1	0	0
1	0	0	0	0	0	0	0	0	0	1	1
0	1	0	0	0	0	1	0	1	0	0	0
0	1	0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	1	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	1
0	0	1	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1	0
0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	1	0	0	1	0

tr_matrix1 =

0	7.9660	1.4640	27.7860	5.3370	1.9030	0.6680	2.8700	5.3780	10.8770	
1.9170	0.4890									
11.4980	0	50.0810	26.5460	22.3380	49.9850	35.1180	16.3040	10.0060	5.7980	
10.6310	1.6280									
0.2250	3.5520	0	5.3580	0.4450	0.2410	0.3560	0.2390	0.8530	0.3440	0.5430
0.1660										
56.5330	32.4750	43.0560	0	101.2940	43.5480	34.2330	75.7360	141.1880	99.2920	
125.7210	28.2170									
11.3310	3.1010	1.2690	6.4140	0	3.5100	1.2500	0.7750	1.9580	0.7640	1.2970
0.2860										
7.0390	11.1570	2.1620	18.6920	8.9410	0	1.5670	5.1090	5.4570	3.3010	
3.8990	0.6730									
0.2630	2.7890	0.5000	5.0220	0.6840	0.4010	0	0.3930	0.9810	0.3870	0.7270
0.1920										
8.6890	2.0030	1.3450	8.2920	2.4760	2.6450	1.1930	0	5.2580	2.8090	2.3660
0.5440										
4.1380	8.6010	2.9250	33.5160	11.0580	8.5500	6.5000	6.2120	0	5.8320	
11.1090	3.4050									
0.3880	0.3580	0.6090	18.7960	1.1580	0.7200	0.5450	2.1410	2.6930	0	1.0520
0.4370										
0.8900	0.7010	0.6870	14.8060	1.6950	1.2100	0.8400	2.7060	5.9680	1.8710	0
0.6370										
0.3900	0.4070	0.4230	4.7750	0.7890	0.5030	0.4320	1.1760	1.3510	0.4730	
1.0760	0									

2^η τοπολογία δικτύου

G2 =

Columns 1 through 20

```

0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0
1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1
0 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 1 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1
1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0

```

Columns 21 through 34

```

0 0 0 0 0 0 0 0 0 0 1 1 0 0
1 0 0 1 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 1 0 0
1 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 0 0 0 0 1 0 0 0 0
0 1 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 1 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 1 0 0 0 1 0 0
0 0 0 0 0 0 1 0 0 0 0 0 1

```

```

0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 1 0 0
0 1 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0
0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0

```

```

dist2 =
Columns 1 through 10

    0     0     0     0     0    353     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0     0     0     0     0     326     0     0    1363
    0     0     0     0     0     0     0     0     0     0
353     0     0     0     0     0     0    379     820     0     0
    0     0     0    326     0    379     0     0     0     0
    0     0     0     0     0    820     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0     0    1363     0     0     0     0     0     0
    0     0     0     0     0     0     0     0     238     0
    0     0     0     0     0     0     0     0     0    1497
    0     0    803     0    1618     0     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0    746     0     0     0     0     0     0     0
    0     0     0     0     0     0     0    1597     0     0
    0     0     0     0    387     0     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0     0    255     0     0    406     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0    269     0     0     0     0    479     0     0     0
    0     0     0     0     0     0     0     0    224     0
    0     0     0     0     0     0     0     0     0     0
    0    219     0     0     0     0    743    935     0     0
    0     0     0     0     0     0     0    613     0     0
    0     0     0     0     0    342     0     0     0     0
    0     0     0     0     0     0     0     0     0    583
    0     0    414     0     0     0     0     0     0     0
    0     0     0     0     0     0     0     0     0     0
    0     0     0     0     0     0     0    561    581     0
320     0     0     0     0     0     0     0     0     0
 67     0     0     0     0    366     0     0     0     0
    0     0    714     0     0     0     0     0     0     0
    0    380     0     0     0     0     0     0     0     0

```

Columns 11 through 20

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	803	0	746	0	0	0	0	0
0	0	0	0	0	0	0	0	255	0
0	0	1618	0	0	0	387	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	406	0
0	0	0	0	0	1597	0	0	0	0
238	0	0	0	0	0	0	0	0	0
0	1497	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1342	0	0	0	0	0	0
0	0	1342	0	386	0	0	0	0	0
0	0	0	386	0	0	0	0	0	0
0	0	0	0	0	0	0	2053	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	2053	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	379	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	547
0	0	947	0	0	0	1666	0	1725	0
0	0	0	0	0	0	0	0	0	0
947	0	0	0	0	0	0	2361	0	0
0	0	0	0	0	0	0	0	0	556
0	0	0	0	0	1863	0	0	0	0
0	0	0	0	698	0	0	0	0	0
1184	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	179	0	0	0	0	673	0
0	0	0	0	223	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	484	0	0	0	597	0	0	0	0

Columns 21 through 30

0	0	0	0	0	0	0	0	0	0
269	0	0	219	0	0	0	0	0	0
0	0	0	0	0	0	0	414	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	342	0	0	0	0
479	0	0	743	0	0	0	0	0	0
0	0	0	935	613	0	0	0	0	561
0	224	0	0	0	0	0	0	0	581
0	0	0	0	0	0	583	0	0	0
0	0	0	0	947	0	0	0	1184	0
379	0	0	0	0	0	0	0	0	0
0	0	947	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	698	0	0
0	0	0	0	0	0	1863	0	0	0
0	0	1666	0	0	0	0	0	0	0
0	0	0	0	2361	0	0	0	0	0
0	0	1725	0	0	0	0	0	0	0

0	547	0	0	0	556	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	956	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1341	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1341	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	956	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	534	0	0
0	0	0	401	0	0	0	0	0	0

Columns 31 through 34

320	67	0	0
0	0	0	380
0	0	714	0
0	0	0	0
0	0	0	0
0	366	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	484
0	0	0	0
179	0	0	0
0	223	0	0
0	0	0	597
0	0	0	0
0	0	0	0
673	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	401
0	0	0	0
0	0	0	0
0	0	0	0
0	0	534	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

tr_matrix_2 =

Columns 1 through 12

0	6.2749	12.5497	6.2749	0	25.0994	18.8246	6.2749	0	0	0	6.2749
6.2749	0	0	6.2749	0	69.0234	12.5497	6.2749	0	0	0	12.5497
12.5497	0	0	0	0	75.2982	87.8479	0	0	0	0	0
6.2749	6.2749	0	0	0	43.9240	12.5497	6.2749	0	6.2749	0	6.2749
0	0	0	0	0	0	0	0	0	0	0	0
12.5497	18.8246	56.4737	18.8246	0	0	37.6491	6.2749	6.2749	0	0	0
31.3743	12.5497	18.8246	75.2982	6.2749	0	87.8479	0	18.8246	0	56.4737	0
25.0994	6.2749	6.2749	0	6.2749	0	18.8246	12.5497	0	0	0	6.2749
0	0	0	0	0	31.3743	0	0	0	0	0	0
0	0	0	6.2749	0	0	163.1461	0	0	0	0	12.5497
0	0	0	0	0	0	0	6.2749	0	0	0	0
6.2749	12.5497	0	6.2749	0	6.2749	94.1228	0	0	12.5497	0	0
0	0	81.5731	0	0	0	0	0	0	0	0	0
6.2749	0	25.0994	25.0994	0	6.2749	6.2749	18.8246	0	0	0	0
6.2749	0	6.2749	6.2749	0	6.2749	6.2749	6.2749	0	0	0	0
0	6.2749	0	0	0	0	6.2749	6.2749	0	0	0	6.2749
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
6.2749	43.9240	0	6.2749	0	6.2749	31.3743	6.2749	0	75.2982	0	0
56.4737	0	0	0	0	0	6.2749	0	0	0	0	0
12.5497	6.2749	0	6.2749	0	6.2749	6.2749	0	0	31.3743	0	18.8246
0	0	0	0	0	62.7485	0	6.2749	0	0	37.6491	0
0	0	6.2749	0	0	0	0	0	0	0	0	0
6.2749	6.2749	0	18.8246	0	6.2749	12.5497	6.2749	0	0	0	25.0994
6.2749	6.2749	0	0	0	6.2749	6.2749	6.2749	0	0	6.2749	0
6.2749	6.2749	18.8246	6.2749	0	6.2749	12.5497	6.2749	6.2749	0	6.2749	0
6.2749	0	0	0	0	0	0	0	0	6.2749	0	0
6.2749	0	6.2749	0	0	6.2749	6.2749	0	0	0	0	0
0	0	0	0	0	0	0	0	6.2749	0	6.2749	0
0	0	0	0	0	0	0	0	0	0	0	0
6.2749	6.2749	0	0	0	6.2749	6.2749	6.2749	0	0	6.2749	0
6.2749	0	43.9240	0	0	0	6.2749	0	0	0	0	0
6.2749	6.2749	6.2749	6.2749	0	6.2749	6.2749	6.2749	0	0	0	6.2749
6.2749	0	106.6725	0	0	0	0	0	0	0	0	0
0	12.5497	0	6.2749	0	6.2749	25.0994	6.2749	0	0	0	31.3743

Columns 13 through 24

0	6.2749	6.2749	0	0	0	6.2749	0	0	0	0	0
0	0	0	6.2749	0	0	6.2749	0	6.2749	0	0	6.2749
56.4737	0	6.2749	0	0	0	0	0	0	0	6.2749	0
0	18.8246	6.2749	0	0	0	6.2749	0	0	0	0	6.2749
0	0	0	0	0	0	0	0	0	0	0	0
0	12.5497	18.8246	0	0	0	18.8246	6.2749	0	125.4970	0	12.5497
0	18.8246	18.8246	37.6491	0	0	31.3743	18.8246	6.2749	0	0	0
25.0994	0	12.5497	6.2749	6.2749	0	0	6.2749	0	0	6.2749	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	12.5497	0	6.2749	0	0	0
0	0	0	0	0	0	0	0	0	56.4737	0	0
0	0	0	6.2749	0	0	25.0994	0	6.2749	0	0	12.5497

0	25.0994	6.2749	0	0	0	0	0	0	0	0	0	0
43.9240	0	6.2749	0	0	0	119.2222	0	0	0	0	0	0
6.2749	6.2749	0	0	0	0	31.3743	0	6.2749	0	0	18.8246	0
0	0	0	0	0	0	0	62.7485	0	0	18.8246	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	125.4970	6.2749	0	0	0	0	6.2749	12.5497	0	6.2749	56.4737	0
0	0	0	0	0	0	31.3743	0	0	0	0	6.2749	0
0	0	12.5497	31.3743	0	0	12.5497	0	0	0	0	25.0994	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	6.2749	0	0	0	0	0	0
0	0	18.8246	18.8246	0	0	131.7719	12.5497	6.2749	0	0	0	0
0	0	0	0	0	0	0	0	0	6.2749	0	75.2982	0
0	6.2749	6.2749	0	0	0	12.5497	6.2749	6.2749	18.8246	0	6.2749	0
0	0	0	6.2749	0	0	0	0	0	0	0	0	0
6.2749	69.0234	6.2749	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	6.2749	0	37.6491	0	0	0
0	0	0	0	0	0	0	6.2749	0	6.2749	0	81.5731	0
12.5497	6.2749	6.2749	0	0	0	37.6491	12.5497	69.0234	0	0	0	0
6.2749	6.2749	6.2749	0	0	0	18.8246	0	0	0	0	6.2749	0
6.2749	37.6491	6.2749	0	0	0	0	0	0	0	0	0	0
0	0	0	6.2749	0	0	18.8246	0	81.5731	0	0	31.3743	0

Columns 25 through 34

0	0	0	0	0	0	0	0	0	0	0	0	0
6.2749	6.2749	0	0	0	6.2749	0	6.2749	0	12.5497	0	0	0
0	18.8246	0	6.2749	0	0	75.2982	6.2749	100.3976	0	0	0	0
0	6.2749	0	0	0	0	0	6.2749	0	6.2749	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
18.8246	6.2749	0	25.0994	0	18.8246	6.2749	6.2749	0	25.0994	0	0	0
25.0994	18.8246	0	18.8246	0	25.0994	18.8246	12.5497	0	43.9240	0	0	0
6.2749	6.2749	0	0	0	6.2749	0	6.2749	0	6.2749	0	0	0
0	6.2749	0	0	6.2749	0	0	0	0	0	0	0	0
0	0	6.2749	0	0	0	0	0	0	0	0	0	0
6.2749	6.2749	0	0	6.2749	6.2749	0	0	0	0	0	0	0
0	6.2749	0	0	0	0	0	6.2749	0	31.3743	0	0	0
0	0	0	6.2749	0	0	0	6.2749	6.2749	0	0	0	0
0	6.2749	0	31.3743	0	0	0	6.2749	50.1988	0	0	0	0
0	6.2749	0	6.2749	0	0	6.2749	6.2749	6.2749	0	0	0	0
0	0	6.2749	0	0	0	0	0	0	6.2749	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	6.2749	0	0	0	0	12.5497	6.2749	0	56.4737	0	0	0
0	6.2749	0	0	6.2749	6.2749	12.5497	0	0	0	0	0	0
0	6.2749	0	0	0	0	69.0234	6.2749	0	18.8246	0	0	0
6.2749	18.8246	0	0	106.6725	6.2749	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
43.9240	6.2749	0	0	0	50.1988	0	6.2749	0	25.0994	0	0	0
0	6.2749	0	0	0	6.2749	0	0	0	0	0	0	0
6.2749	0	0	6.2749	0	6.2749	6.2749	6.2749	0	6.2749	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	6.2749	0	0	0	0	18.8246	6.2749	6.2749	0	0	0	0
0	0	0	0	0	6.2749	0	0	0	0	0	0	0
6.2749	6.2749	0	0	6.2749	0	0	6.2749	0	0	0	0	0
0	6.2749	0	25.0994	0	0	0	0	0	0	0	0	0
0	6.2749	0	6.2749	0	6.2749	0	0	6.2749	0	0	0	0
0	0	0	6.2749	0	0	0	6.2749	0	0	0	0	0
0	6.2749	0	0	0	0	0	0	0	0	0	0	0