

Δρομολόγηση σε Οπορτουμιστικά Δίκτυα με Αρχές Μηχανικής Μάθησης

Η Μεταπτυχιακή Εργασία Εξειδίκευσης

υποβάλλεται στην ορισθείσα

από τη Γενική Συνέλευση Ειδικής Σύνθεσης
του Τμήματος Μηχανικών Η/Υ και Πληροφορικής
Εξεταστική Επιτροπή

από τον

Βασίλειο Τουλατζή

ως μέρος των υποχρεώσεων για την απόκτηση του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ

ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

ΜΕ ΕΞΕΙΔΙΚΕΥΣΗ

ΣΤΑ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

Πανεπιστήμιο Ιωαννίνων

Ιούλιος 2017

Εξεταστική Επιτροπή:

- **Αριστείδης Λύκας**, Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- **Λυσίμαχος Παύλος Κόντης**, Αναπλ. Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- **Ευάγγελος Παπαπέτρου**, Επίκ. Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων (Επιβλέπων)

ΑΦΙΕΡΩΣΗ

Αφιερώνω αυτήν την εργασία στην οικογένεια μου και σε όλους τους ανθρώπους που με στήριξαν.

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά, θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέποντα καθηγητή της μεταπτυχιακής αυτής εργασίας, Επίκουρο Καθηγητή κ.Ευάγγελο Παπαπέτρου, για την εμπιστοσύνη που μου έδειξε αναθέτοντας μου την συγκεκριμένη εργασία, την πολύτιμη βοήθειά, την καθοδήγησή του αλλά και για τις σημαντικές γνώσεις που μου μετέδωσε καθ' όλη τη διάρκειά της συνεργασίας μας. Επιπρόσθετα, θα ήθελα να ευχαριστήσω τον Καθηγητή, κ.Λύκα Αριστείδη για την αποτελεσματική συνεργασία και συμβολή του στην ολοκλήρωση της παρούσας εργασίας.

Επιπλέον, θα ήθελα να εκφράσω τις εγκάρδιες ευχαριστίες μου στην οικογένειά μου, για την αμέριστη συναισθηματική, ηθική και οικονομική υποστήριξη που μου παρείχαν σε όλη την διάρκεια των σπουδών μου. Ένα τεράστιο ευχαριστώ στον πατέρα μου Ιωάννη και την μητέρα μου Κωνσταντινιά, γιατί χωρίς εκείνους δεν θα μπορούσε να ολοκληρωθεί η διπλωματική αυτή. Επίσης, θα ήθελα να ευχαριστήσω και τα δύο αδέρφια μου, Στέργιο και Μάριο που ήταν πάντα δίπλα μου και με υποστήριζαν.

Οφείλω σίγουρα ένα μεγάλο ευχαριστώ στην Αναστασία που δίχως την υπομονή, την κατανόηση, την αμέριστη αγάπη και τις συμβουλές της δεν θα ήταν όλα ίδια. Φυσικά, την ευχαριστώ και για όλες τις στιγμές που ήταν πάντα εκεί δίπλα μου για ότι χρειαζόμουν. Τέλος, θα ήθελα να ευχαριστήσω και όλους τους φίλους μου που πίστεψαν σε εμένα και με στήριξαν όλα αυτά τα χρόνια.

ΠΕΡΙΕΧΟΜΕΝΑ

Κατάλογος Σχημάτων	iii
Κατάλογος Πινάκων	v
Κατάλογος Αλγορίθμων	vii
Περίληψη	viii
Extended Abstract	x
1 Εισαγωγή	1
1.1 Οπορτουνιστικά Δίκτυα και Δρομολόγηση	1
1.2 Συμβολή της Διπλωματικής Εργασίας	3
1.3 Δομή της Διπλωματικής Εργασίας	5
2 Βασικές έννοιες & Σχετική Βιβλιογραφία	7
2.1 Οπορτουνιστικά Δίκτυα (Opportunistic Networks - OppNets)	7
2.2 Ανασκόπηση Αλγορίθμων Δρομολόγησης	13
3 Δρομολόγηση με Δυναμική Δημιουργία Αντιτύπων	18
3.1 Compare & Replicate (CNR)	20
3.2 Delegation Forwarding (DF)	22
3.3 Coordinated Delegation Forwarding (COORD)	26
4 Δρομολόγηση με Αρχές Μηχανικής Μάθησης	30
4.1 Κίνητρο και Κεντρική Ιδέα	30
4.2 Υπόβαθρο Μηχανικής Μάθησης	33
4.3 Δρομολόγηση με Εφαρμογή και Ενημέρωση του K-means	38
4.3.1 Δρομολόγηση με K-means	39

4.3.2	Ενημέρωση Πληροφορίας K-means (LVQ)	43
5	Αποτελέσματα	46
5.1	Βασικά Αποτελέσματα	47
5.1.1	Reality	48
5.1.2	Milano	51
5.1.3	Infocom 2005	54
5.1.4	Dartmouth2weeks	57
5.2	Παραμετροποίηση K-means & Αποτελέσματα	60
5.2.1	Επιλογή Πλήθους N Καταγραφών K-means	60
5.2.2	Προσαρμογή στις Μεταβολές - Επιλογή Ρυθμού Μάθησης η (LVQ)	67
6	Συμπεράσματα	74
	Βιβλιογραφία	76

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

2.1 Ένα παράδειγμα της τεχνικής Store, Carry & Forward κατά την δρομολόγηση ενός πακέτου τις χρονικές στιγμές t_1 , t_2 και t_3 από τον κόμβο A στον κόμβο D	9
2.2 Περιγραφή μιας Anti-entropy Session (συνεδρίας)	10
2.3 Κατηγοριοποίηση μετρικής ωφέλειας (Utility Metric).	11
2.4 Κατηγοριοποίηση των τεχνικών δρομολόγησης στα ομορτουμιστικά δίκτυα. Η κατηγοριοποίηση αυτή βασίζεται στην διατριβή [1])	15
3.1 CNR: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B	22
3.2 DF: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B	25
3.3 COORD: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B	29
4.1 Δρομολόγηση με Kmeans: Σύγκριση διαδικασίας προώθησης πακέτων με χρήση της πληροφορία ομαδοποίησης.	42
5.1 Reality: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.	49
5.2 Milano: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.	51
5.3 Infocom2005: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.	54
5.4 Dartmouth2weeks: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.	57

5.5	Reality - LTS: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.	61
5.6	Reality - DestEnc: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.	63
5.7	Reality - Prophet: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.	65
5.8	Reality - Betweeness: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.	66
5.9	Reality - LTS: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.	68
5.10	Reality - DestEnc: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.	70
5.11	Reality - Prophet: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.	71
5.12	Reality - Betweeness: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.	73

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

5.1	Πίνακας πληροφοριών των traces που χρησιμοποιήθηκαν.	48
5.2	Reality: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).	49
5.3	Reality: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).	50
5.4	Reality: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).	50
5.5	Reality: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).	50
5.6	Milano: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).	52
5.7	Milano: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).	52
5.8	Milano: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).	53
5.9	Milano: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).	53
5.10	Infocom2005: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).	55
5.11	Infocom2005: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).	55
5.12	Infocom2005: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).	56
5.13	Infocom2005: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).	56
5.14	Dartmouth2weeks: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).	58

5.15 Dartmouth2weeks: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).	58
5.16 Dartmouth2weeks: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).	59
5.17 Dartmouth2weeks: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).	59

ΚΑΤΑΛΟΓΟΣ ΑΛΓΟΡΙΘΜΩΝ

3.1	Compare & Replicate	21
3.2	Delegation Forwarding	23
3.3	Coordinated Delegation Forwarding	27
4.1	Δρομολόγηση με Αρχές Μηχανικής Μάθησης (K-means)	40
4.2	Learning Vector Quantization (LVQ) for Clustering Update	44

ΠΕΡΙΛΗΨΗ

Βασίλειος Τουλατζής, Μ.Δ.Ε. στην Πληροφορική, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων, Ιούλιος 2017.

Δρομολόγηση σε Οπορτουμιστικά Δίκτυα με Αρχές Μηχανικής Μάθησης.

Επιβλέπων: Ευάγγελος Παπαπέτρου, Επίκουρος Καθηγητής.

Η παρούσα μεταπτυχιακή εργασία επικεντρώνεται στην δρομολόγηση σε οπορτουμιστικά δίκτυα (OppNets). Τα οπορτουμιστικά δίκτυα περιγράφουν το παράδειγμα δικτύωσης για κινητούς χρήστες στα πλαίσια των δικτύων πέμπτης γενιάς. Πρόκειται για δίκτυα στα οποία δεν υπάρχει και δεν είναι απαραίτητη μία δικτυακή υποδομή για την επικοινωνία των κόμβων. Ένας κόμβος κινείται και επικοινωνεί κατά τις επαφές του με άλλους κόμβους του δικτύου. Η επαφή (contact) δύο κόμβων στο δίκτυο επιτυγχάνεται όταν ο ένας κόμβος εισέρχεται στην εμβέλεια του άλλου και είναι δυνατή η ανταλλαγή μηνυμάτων. Έτσι ένα μονοπάτι στο δίκτυο σχηματίζεται δυναμικά και αποτελείται από ένα πλήθος επαφών. Οι συνδέσεις μεταξύ των κόμβων είναι παροδικές και όχι μόνιμες. Αποτέλεσμα είναι ένα οπορτουμιστικό δίκτυο να μην είναι συνδεδεμένο από άκρη σε άκρη.

Όπως η επικοινωνία έτσι και η δρομολόγηση πακέτων επιτυγχάνεται με την συμβολή κάθε κόμβου στο δίκτυο. Ένας κόμβος κινείται, αποθηκεύει, μεταφέρει και προωθεί πακέτα δεδομένων στους άλλους κόμβους του δικτύου. Οι αλγόριθμοι δρομολόγησης χρησιμοποιούν είτε ένα είτε πολλαπλά αντίτυπα ενός πακέτου στο δίκτυο. Η δεύτερη κατηγορία αποτελεί την αποτελεσματικότερη εκδοχή αλγορίθμων αφού η ύπαρξη αντιτύπων οδηγεί σε μεγαλύτερα ποσοστά επιτυχούς παράδοσης και σε μικρότερες καθυστερήσεις. Ωστόσο, βασικό ζήτημα στην κατηγορία αυτή αποτελεί η κατανάλωση ενέργειας στην οποία οδηγεί ο μεγάλος αριθμός μεταδόσεων στο δίκτυο. Μια υποκατηγορία των αλγορίθμων δρομολόγησης με αντίτυπα, η οποία επικεντρώνεται στο παραπάνω ζήτημα είναι η δρομολόγηση με δυναμική δημιουργία αντιτύπων. Οι αλγόριθμοι αυτοί χρησιμοποιούν μια μετρική αξιολόγησης

των κόμβων του δικτύου, η οποία καλείται Utility (ωφέλεια). Η μετρική αυτή προσομοιώνει την ικανότητα ενός κόμβου για προώθηση ή/και παράδοση ενός πακέτου στον κόμβο προορισμό. Όσο μεγαλύτερη είναι η τιμή του Utility τόσο καταλληλότερος για παράδοση θεωρείται ένας κόμβος. Η παραπάνω μετρική χρησιμοποιείται στην πράξη για την απόφαση δημιουργίας και προώθησης αντιτύπων. Οι αλγόριθμοι με δυναμική δημιουργία αντιτύπων δημιουργούν αντίγραφα συγκρίνοντας την ωφέλεια του κόμβου που έρχεται σε επαφή με αυτή του κόμβου που μεταφέρει το πακέτο ή ακόμα και με τη μεγαλύτερη τιμή που έχει καταγραφεί κατά το παρελθόν. Ωστόσο, η παρατήρησή μας είναι ότι η στρατηγική αυτή μπορεί να οδηγήσει στη δημιουργία μεγάλου πλήθους αντιτύπων. Αυτό συμβαίνει γιατί αυτή η "συγκριτική" προσέγγιση δεν εξετάζει πόσο σημαντική είναι η βελτίωση που επιτυγχάνεται με την απόφαση δημιουργίας ενός αντιτύπου.

Η εργασία αυτή βασίστηκε στην παραπάνω παρατήρηση έτσι ώστε να ορίσουμε ένα καλύτερο κριτήριο απόφασης για την δημιουργία αντιτύπων στο δίκτυο. Στόχος μας είναι η περαιτέρω μείωση της κατανάλωσης ενέργειας, δίχως όμως να θυσιάζουμε την ικανότητα παράδοσης των πακέτων. Προτείνουμε μια επέκταση των μηχανισμών δρομολόγησης με δυναμική δημιουργία αντιτύπων, όπου χρησιμοποιούμε αρχές μηχανικής μάθησης. Στην προσέγγισή μας, ένας χρήστης που μεταφέρει ένα πακέτο χρησιμοποιεί αρχές μηχανικής μάθησης ώστε να αποφασίσει κατά πόσο η ωφέλεια ενός κόμβου που έρχεται σε επαφή τον κατατάσσει ως ένα σημαντικό κόμβο στο δίκτυο. Με τον τρόπο αυτό μπορεί να λάβει καλύτερες αποφάσεις για τη δημιουργία αντιτύπων. Κατά συνέπεια, καταφέρνουμε μέσω της μηχανικής μάθησης να εξάγουμε την επιθυμητή πληροφορία γενικής κατάταξης ενός κόμβου στο δίκτυο και επιτυγχάνουμε αξιοσημείωτες μειώσεις στην κατανάλωση ενέργειας σε κάθε περίπτωση αλγορίθμου. Δείχνουμε ότι η προσέγγισή μας μπορεί να βελτιώσει την απόδοση σε διαφορετικά δίκτυα και ανεξάρτητα του τρόπου κατασκευής του Utility.

EXTENDED ABSTRACT

Vasileios Toulatzis, M.Sc. in Computer Science, Department of Computer Science and Engineering, University of Ioannina, Greece, July 2017.

Routing Techniques in Opportunistic Networks with Principles of Machine Learning.
Advisor: Evangelos Papapetrou, Assistant Professor.

In the present thesis, we investigate routing techniques with low-level energy consumption in wireless opportunistic networks (OppNets). Such networks are foreseen to be key elements in future 5G networks. Networks of this type are self-organized with no need of fixed infrastructure. Communication is possible in a contact basis, i.e. when moving nodes come into range of one another. In opportunistic networks every node has to act like a router due to intermittent connectivity. Therefore, paths in this type of networks are dynamically formed and consist of a sequence of contacts. The path from source to destination does not necessarily exist or is randomly formed. This is the main reason why traditional routing techniques fail in OppNets.

Routing algorithms in opportunistic networks are based on store, carry and forward scheme. In many cases of routing algorithms, there is no control in forwarding procedure. Consequently, a technique that estimates in which candidate a node should forward a packet is necessary. The key to estimate the ability of a node to forward or/and deliver a packet is a metric called Utility. Each network node calculates locally its Utility value. This process is called Utility Estimation. In any contact with another network, node compares this value with the other node's value to resolve if it's time to forward a packet. Routing algorithms construct a Utility by different ways.

A routing strategy is also significant for routing process. Routing strategy is the way an algorithm uses and exploits the Utility. In particular, it focuses on creating forwarding decision rules by using a Utility. A routing strategy uses either a single or multi copy of a packet. This is the main categorization of routing algorithms. Most well-known routing strategy is packet replication which uses multiple copies

(replicas). Packet replication maximizes the delivery efficiency, but it can remarkably increase the number of transmissions. This consequently means a huge impact on the energy consumption.

A subcategory of multi-copy routing algorithms, which tends to decrease forwards is dynamic replication. Dynamic replication uses a Utility for making decisions of packet replication in every contact. A node creates a replica if and only if a contact with another network node exists and the Utility value of this node is higher than its value or higher than the highest Utility value (threshold) that this node has store. Let two network nodes A , B and their corresponding Utility values $U_A = 0.1$, $U_B = 0.11$. A $Contact(A, B)$ is up (active) and node A must forward a packet replica. Node B has a higher Utility value than A but how good is this value in the whole network. These routing algorithms can lead to huge energy consumption. This outcome is possible because there is no estimation of how beneficial is a replica creation. No routing algorithm uses this type of information.

The above observation is the main idea behind our proposal. In opportunistic networks a way of extracting the information about ranking of each node is a necessity. This node ranking comes out of a corresponding Utility value ranking formation. Our main goal is to extend routing algorithms with Principles of Machine Learning in order to further reduce the energy consumption without sacrificing the delivery efficiency. We are using a clustering technique to group the Utility values, known as Kmeans, to mine this extra information about the network. This information can be exploited to reduce replications of data packets. We introduce a simple method which can be attached to any routing algorithm that uses a Utility to provide better results in energy consumption. Furthermore, our method is also generic in the sense that it can be used with any routing algorithm and any utility metric for making replication decisions. In this thesis, we focus our investigation on three of the most famous and effective routing algorithms that use the dynamic replication. We validate the performance of our method through extensive simulations with different Utilities.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

-
- 1.1 Οπορτουνιστικά Δίκτυα και Δρομολόγηση
 - 1.2 Συμβολή της Διπλωματικής Εργασίας
 - 1.3 Δομή της Διπλωματικής Εργασίας
-

1.1 Οπορτουνιστικά Δίκτυα και Δρομολόγηση

Τα ασύρματα δίκτυα έχουν μελετηθεί εκτεταμένα τις τελευταίες δεκαετίες καθώς οι ασύρματες επικοινωνίες έχουν ενταχθεί σε μεγάλη κλίμακα στην καθημερινότητα μας. Σήμερα όλο και περισσότερο παρατηρούμε την μεγάλη σημασία της έρευνας αυτής. Οι χρήστες πλέον έχουν την ανάγκη να συνδέονται ασύρματα και σε διαφορετικές τοποθεσίες ακόμα και ενώ κινούνται. Ωστόσο, στα παραδοσιακά ασύρματα δίκτυα η συνδεσιμότητα επιτυγχάνεται μέσω μίας κατάλληλη δικτυακής υποδομής. Για την παροχή υπηρεσίας ασύρματης επικοινωνίας δίχως την παρουσία κάποιας απαραίτητης υποδομής, η ερευνητική κοινότητα εισήγαγε την έννοια των ασύρματων αδόμητων δικτύων πολλών αλμάτων (multihop adhoc networks). Σε τέτοιου τύπου ασύρματα δίκτυα οι χρήστες ουσιαστικά δημιουργούν και οργανώνουν ένα δίκτυο στο οποίο ο κάθε κόμβος συμμετέχει ενεργά στην διαδικασία της δρομολόγησης. Η επικοινωνία για έναν κόμβο επιτυγχάνεται με δύο τρόπους. Ο ένας αφορά την άμεση επαφή με τον κόμβο προορισμό ενός πακέτου και ο δεύτερος την δρομολόγηση με πολλαπλά άλματα μέσω ενδιάμεσων κόμβων. Έτσι ένα κόμβος προωθεί ένα πακέτο στον κόμβο προορισμό ή βασίζεται σε ενδιάμεσους κόμβους ώστε να

προωθήσει τα πακέτα του σε αυτόν. Η απόφαση σε ποιους κόμβους του δικτύου θα προωθήσει δεδομένα λαμβάνεται δυναμικά με βάση το ιστορικό συνδεσιμότητας.

Στην κατηγορία αυτών των δικτύων ανήκουν και τα ομοιομορφικά δίκτυα, στα οποία η συνδεσιμότητα μεταξύ των κόμβων είναι παροδική, υπό την έννοια ότι ένας κόμβος δεν βρίσκεται μόνιμα σε επικοινωνία με άλλους κόμβους του δικτύου. Δύο κόμβοι έρχονται σε επικοινωνία μόνο στην περίπτωση όπου ο ένας βρίσκεται στην εμβέλεια του άλλου και είναι δυνατή η αποστολή δεδομένων. Η παραπάνω κατάσταση ονομάζεται επαφή (contact) των δύο κόμβων. Παρά την δυνατότητα μετακίνησης ενός κόμβου, υπάρχουν περιπτώσεις χρονικών διαστημάτων στις οποίες ο κόμβος αυτός μένει απομονωμένος από το υπόλοιπο δίκτυο καθώς δεν έρχεται σε επαφή με κάποιο άλλο κόμβο του δικτύου. Αυτό καθιστά τα παραδοσιακά πρωτόκολλα δρομολόγησης μη αποτελεσματικά. Η λύση η οποία έχει προταθεί είναι η μέθοδος αποθήκευσης, μεταφοράς και προώθησης (Store, Carry & Forward), έναντι της παραδοσιακής Store & Forward τεχνικής. Κατά την διαδικασία αυτή, ένας κόμβος αποθηκεύει και μεταφέρει ένα πακέτο μέχρις ότου έρθει σε επαφή είτε με τον κόμβο προορισμό είτε με έναν κόμβο ο οποίος είναι κατάλληλος και ικανός ώστε να προωθήσει/παραδώσει το πακέτο στον κόμβο προορισμό.

Σημαντική συνιστώσα της διαδικασίας δρομολόγησης αποτελεί η αξιολόγηση της καταλληλότητας ενός κόμβου για την προώθηση ενός πακέτου. Η εκτίμηση αν ένας κόμβος είναι κατάλληλος για προώθηση επιτυγχάνεται με μία μετρική αξιολόγησης, η οποία ονομάζεται Utility. Το Utility αποτελεί ένα βαθμωτό μέγεθος το οποίο κάθε κόμβος του δικτύου υπολογίζει αξιοποιώντας το ιστορικό των επαφών του. Η τιμή αυτή αποτελεί και το μέτρο καταλληλότητας με το οποίο αξιολογείται ένας κόμβος. Σε μία επαφή του κόμβου, η απόφαση για προώθηση ενός πακέτου λαμβάνεται με κριτήριο την σύγκριση των Utilities των κόμβων οι οποίοι βρίσκονται σε επαφή. Υπάρχουν πολλοί διαφορετικοί τρόποι να κατασκευαστεί ένα Utility. Οι τρόποι κατασκευής του Utility προκύπτουν από οποιαδήποτε είδους πληροφορία την οποία μπορεί κανείς να εξάγει από το ιστορικό των επαφών. Παραδείγματα αποτελούν η κατασκευή του βάσει του πλήθους ή της συχνότητας των επαφών ενός κόμβου, η ομοιότητα των κόμβων (κοινοί γείτονες), ακόμα και μια πιθανοτική προσέγγιση που χρησιμοποιείται για πρόβλεψη παράδοσης ενός πακέτου και υπολογίζεται βάσει των επαφών με τον κόμβο προορισμό. Διαφορετικοί αλγόριθμοι χρησιμοποιούν διαφορετικά Utilities.

Ένα ακόμα σημαντικό κομμάτι της δρομολόγησης αποτελεί ο τρόπος με τον

οποίο αξιοποιείται η πληροφορία αξιολόγησης μέσω ενός Utility. Πως ουσιαστικά χρησιμοποιείται ένα Utility για την απόφαση προώθησης κάθε πακέτου στο δίκτυο. Η συνιστώσα αυτή της διαδικασίας δρομολόγησης ονομάζεται τακτική δρομολόγησης και αποτελείται από κανόνες προώθησης. Μια τακτική δρομολόγησης μπορεί να ακολουθεί δύο διαφορετικές στρατηγικές, την παραγωγή ενός ή πολλαπλών αντιτύπων. Η πρώτη στρατηγική, όπως είναι λογικό, δεν παρουσιάζει ικανοποιητικά ποσοστά παράδοσης λόγω της μεγάλης πιθανότητας απώλειας του μοναδικού αντιτύπου. Ακόμα και σε περιπτώσεις που ένα πακέτο φτάνει στον προορισμό του, η καθυστέρηση είναι πολύ μεγάλη. Αντίθετα, η δεύτερη στρατηγική τείνει να δίνει καλύτερα αποτελέσματα. Φυσικά, το κόστος είναι η δημιουργία περισσότερων αντιτύπων ενός πακέτου στο δίκτυο και συνεπώς και η μεγαλύτερη κατανάλωση της ενέργειας ενός κόμβου για την μετάδοση, την λήψη και αποθήκευση των αντιτύπων.

Στην περίπτωση των ομορτυνιστικών δικτύων, κατάλληλοι αλγόριθμοι για την διαδικασία της δρομολόγησης είναι εκείνοι οι οποίοι μπορούν να βοηθήσουν στην μείωση περιττών προωθήσεων αλλά ταυτόχρονα να παρέχουν υψηλά ποσοστά παράδοσης. Στην πραγματικότητα θα θέλαμε αλγορίθμους οι οποίοι να παρουσιάζουν χαμηλό πλήθος προωθήσεων (χαμηλή κατανάλωση ενέργειας) αλλά και ικανοποιητικά ποσοστά επιτυχούς παράδοσης πακέτων. Το επιθυμητό αυτό αποτέλεσμα το προσεγγίζει μια κατηγορία αλγορίθμων η οποία χρησιμοποιεί την δυναμική δημιουργία πολλαπλών αντιτύπων (replicas). Αυτή η κατηγορία αλγορίθμων αποτελεί και την αποτελεσματικότερη εκδοχή έως σήμερα των τεχνικών δρομολόγησης. Ωστόσο, η μείωση της κατανάλωσης ενέργειας παρότι είναι μεγάλη σε σχέση με άλλες τεχνικές, δεν αγγίζει τα επιθυμητά επίπεδα αφού παρατηρείται και εδώ αρκετά μεγάλος αριθμός προωθήσεων.

1.2 Συμβολή της Διπλωματικής Εργασίας

Στην μεταπτυχιακή εργασία αυτή μελετάμε αλγορίθμους οι οποίοι βασίζονται στη χρήση πολλαπλών αντιτύπων. Οι αλγόριθμοι αυτοί αντιμετωπίζουν ένα σημαντικό ζήτημα με τα αποθέματα ενέργειας των κόμβων του δικτύου καθώς υπάρχουν περιπτώσεις εκτεταμένης αύξησης του πλήθους των εκπομπών. Σημαντική βελτίωση ως προς την κατανάλωση ενέργειας παρουσιάζουν οι αλγόριθμοι της κατηγορίας δυναμικής δημιουργίας αντιτύπων. Οι αλγόριθμοι αυτοί χρησιμοποιούν ένα Utility ως

μέτρο αξιολόγησης των κόμβων και ακολουθούν μια "συγκριτική" στρατηγική των τιμών της μετρικής αυτής για την απόφαση προώθησης ενός πακέτου. Το κριτήριο σύγκρισης υλοποιείται με δύο τρόπους. Ο πρώτος αφορά την απλή σύγκριση των Utilities των κόμβων που ήρθαν σε επαφή, ενώ ο δεύτερος βασίζεται στην σύγκριση του Utility με την μέγιστη τιμή την οποία έχει καταγράψει ο κόμβος που μεταφέρει το πακέτο. Παρόλα αυτά και οι δύο εκδοχές ελέγχουν σε μία επαφή μόνο αν ένας κόμβος είναι καταλληλότερος από τον άλλο για παράδοση. Δεν λαμβάνουν υπόψιν τους όμως μια γενική εικόνα των κόμβων στο δίκτυο. Ουσιαστικά η συγκριτική λογική "παγιδεύεται" στα πλαίσια μιας επαφής, δίχως να λαμβάνεται υπόψιν μια γενική κατάταξη των κόμβων στο δίκτυο. Ένας κόμβος σε μια επαφή είναι πιθανόν να γνωρίζει μεγαλύτερη τιμή Utility όμως αυτό δεν τον ορίζει αυτομάτως ως έναν καλό και ικανό κόμβο για παράδοση ή/και προώθηση στο δίκτυο.

Στην πράξη οι αλγόριθμοι αυτοί εστιάζουν είτε μόνο στην πληροφορία για τον αν ένας κόμβος είναι πιο "αξιόπιστος" για να παραδώσει το πακέτο, είτε αν είναι πιο "αξιόπιστος" από όσους έχει καταγράψει ο κόμβος που μεταφέρει ένα πακέτο [2], [3]. Η "συγκριτική" αυτή στρατηγική μπορεί να οδηγήσει στη δημιουργία μεγάλου πλήθους αντιτύπων. Κατά συνέπεια, εστίασαμε στην ενσωμάτωση μιας τεχνικής η οποία χρησιμοποιεί αρχές μηχανικής μάθησης για την εξαγωγή πληροφορίας που αφορά την κατάταξη των κόμβων με βάση το μέτρο αξιολόγησης τους (Utility). Για να το επιτύχουμε αυτό βασιζόμαστε στο ιστορικό των επαφών και φυσικά στις καταγραφές των τιμών του Utility.

Η κεντρική ιδέα της εργασίας βασίζεται στην κατάταξη των κόμβων του δικτύου σε ομάδες καταλληλότητας. Η ομαδοποίηση (clustering) των κόμβων πραγματοποιείται με βάση τις αξιολογήσεις τους (τιμές Utility). Επομένως, κάθε ομάδα κόμβων αντιστοιχίζεται σε ένα διάστημα τιμών του Utility, οπότε ταυτόχρονα και σε ένα επίπεδο καταλληλότητας. Ο σχηματισμός ομάδων στο δίκτυο αποτελεί και την πληροφορία την οποία προσπαθούμε να εκμεταλλευτούμε για την δρομολόγηση. Οι περιπτώσεις όπου δημιουργούνται αντίτυπα είναι εκείνες οι οποίες κρίνονται κατάλληλες σύμφωνα με την πληροφορία κατάταξης ενός κόμβου. Καταλληλότερος πλέον θεωρείται ένας κόμβος, ο οποίος ανήκει σε μία ομάδα κόμβων με υψηλές τιμές Utility. Πιο συγκεκριμένα, σε μια επαφή δύο κόμβων, κάθε κόμβος σε ένα πρώτο βήμα αντιστοιχίζεται σε μία ομάδα με βάση το Utility. Εν συνεχεία, κριτήριο προώθησης αποτελεί η σύγκριση των επιπέδων των ομάδων των κόμβων. Στην πράξη σε μια επαφή του κόμβου A με τον B με τιμές Utility U_A και U_B αντίστοιχα, οι κόμβοι

A και B αρχικά αντιστοιχίζονται με βάση τις τιμές των U_A και U_B σε μία ομάδα (cluster). Κριτήριο προώθησης αποτελεί όπως είπαμε η σύγκριση των επιπέδων των δύο ομάδων. Εάν το $ClusterLevel_A$ είναι μικρότερο του $ClusterLevel_B$ τότε έχουμε προώθηση από τον κόμβο A στον B . Διαφορετικά δεν πραγματοποιείται προώθηση αντιτύπου. Κατά συνέπεια, η συγκριτική λογική περνάει πλέον σε επίπεδο σύγκρισης διαστημάτων των τιμών του Utility (επίπεδα ομάδων) και όχι σε απλή σύγκριση τιμών.

Εξετάσαμε και επικεντρωθήκαμε λοιπόν στον τρόπο με τον οποίο μία μέθοδος μηχανικής μάθησης και πιο συγκεκριμένα μία μέθοδος ομαδοποίησης, θα μπορούσε να οδηγήσει σε καλύτερες αποφάσεις δημιουργίας αντιτύπων. Για την εξαγωγή καλύτερης προβλεψιμότητας επεκτείνουμε τους αλγορίθμους δρομολόγησης δυναμικής δημιουργίας αντιτύπων με μία μέθοδο ομαδοποίησης (K-means) και πετυχαίνουμε σημαντική μείωση στην κατανάλωση ενέργειας. Αξιοσημείωτο είναι επίσης το φαινόμενο διατήρησης σταθερών ποσοστών επιτυχούς παράδοσης σε κάθε περίπτωση αλγορίθμου.

1.3 Δομή της Διπλωματικής Εργασίας

Η δομή της διπλωματικής εργασίας παρακάτω ορίζεται ως εξής:

Στο Κεφάλαιο 2, παρουσιάζουμε βασικές έννοιες για τα ομορτυνιστικά δίκτυα αλλά και την δρομολόγηση σε αυτά. Συγκεκριμένα, αναλύουμε τους μηχανισμούς της δρομολόγησης και περιγράφουμε τους βασικούς αλγορίθμους δρομολόγησης που έχουν προταθεί στη βιβλιογραφία.

Στο Κεφάλαιο 3, θα εστιάσουμε στην δρομολόγηση με δυναμική δημιουργία αντιτύπων. Θα αναλύσουμε εκτενέστερα τα χαρακτηριστικά και την διαδικασία που ακολουθούν οι αλγόριθμοι αυτού του τύπου. Τέλος, θα αναφερθούμε σε κάθε ένα από τους τρεις σημαντικότερους αλγορίθμους που ανήκουν στην συγκεκριμένη κατηγορία και θα περιγράψουμε την λειτουργία τους.

Στο Κεφάλαιο 4, θα εξετάσουμε και θα περιγράψουμε την δρομολόγηση σε ομορτυνιστικά δίκτυα με χρήση αρχών μηχανικής μάθησης. Θα αναλύσουμε την κεντρική ιδέα της επέκτασης καθώς και το πως μπορούμε να αξιοποιήσουμε την αξιολόγηση μέσω ομαδοποίησης για την εξαγωγή χρήσιμης πληροφορίας δρομολόγησης. Στόχος είναι να μειώσουμε την δημιουργία περιττών αντιτύπων στο δίκτυο. Οι βασικές έν-

νοιες της μηχανικής μάθησης, θα περιγραφούν επίσης στο κεφάλαιο αυτό. Τελικά θα επικεντρωθούμε στο πως μία τέτοια μέθοδος ομαδοποίησης (K-means) μπορεί στην πράξη να εφαρμοστεί στην διαδικασία δρομολόγησης στα ομορτουριστικά δίκτυα.

Στο Κεφάλαιο 5, θα παρουσιάσουμε τα αποτελέσματα της μεθόδου μας. Θα αναφερθούμε στην αποτελεσματικότητά της και στην επιπρόσθετη μείωση της καταναλισκόμενης ενέργειας που παρέχει. Θα δούμε επίσης ότι η μέθοδός μας δεν θυσιάζει την ικανότητα παράδοσης πακέτων και θα συγκρίνουμε την δουλειά μας με τους άλλους αλγόριθμους της κατηγορίας δυναμικής δημιουργίας αντιτύπων. Τέλος, θα αναλυθούν αποτελέσματα τα οποία αφορούν την κατάλληλη επιλογή παραμέτρων του αλγορίθμου K-means αλλά και της τεχνικής LVQ που χρησιμοποιείται για προσαρμογή της μεθόδου μας στις μεταβολές του δικτύου.

Τέλος, στο Κεφάλαιο 6 συνοψίζουμε την προτεινόμενη τεχνική και εξάγουμε τα συμπεράσματα σχετικά με τις ωφέλειες που αυτή παρέχει. Επίσης, συζητάμε πιθανές μελλοντικές επεκτάσεις.

ΚΕΦΑΛΑΙΟ 2

ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ & ΣΧΕΤΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

2.1 Οπορτουνιστικά Δίκτυα (Opportunistic Networks - OppNets)

2.2 Ανασκόπηση Αλγορίθμων Δρομολόγησης

Στο κεφάλαιο αυτό θα αναφερθούμε αρχικά σε βασικές έννοιες των οπορτουνιστικών δικτύων και θα επικεντρωθούμε κυρίως στην διαδικασία της δρομολόγησης. Συγκεκριμένα θα αναλύσουμε μηχανισμούς οι οποίοι είναι βασικοί για την λειτουργία της δρομολόγησης. Επιπλέον, θα παρουσιάσουμε και θα μελετήσουμε τις κατηγορίες αλγορίθμων δρομολόγησης που έχουν προταθεί στην βιβλιογραφία. Θα επισημάνουμε διαφορές μεταξύ των κατηγοριών αυτών και θα αναφέρουμε τα πλεονεκτήματα και τα μειονεκτήματα τους.

2.1 Οπορτουνιστικά Δίκτυα (Opportunistic Networks - OppNets)

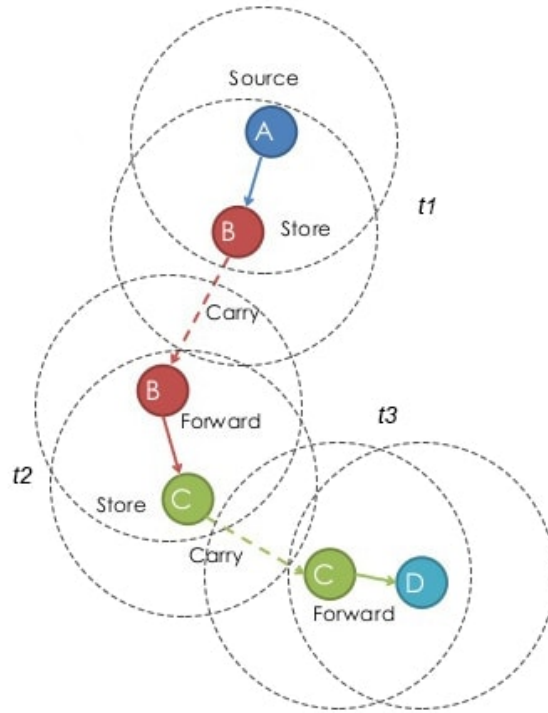
Η κλάση αυτή των ασύρματων αδόμητων δικτύων υποστηρίζει την πλήρη κινητικότητα των κόμβων του δικτύου προς οποιαδήποτε κατεύθυνση. Η επικοινωνία μεταξύ των κόμβων δεν βασίζεται σε μια προϋπάρχουσα δικτυακή υποδομή. Αντίθετα, η επικοινωνία των κόμβων του δικτύου επιτυγχάνεται με μικρής διάρκειας συνδέσεις όταν ένας κόμβος εισέρχεται στην εμβέλεια κάποιου άλλου κόμβου. Ως επαφή δύο κόμβων σε ένα οπορτουνιστικό δίκτυο ονομάζουμε τον παραπάνω τρόπο επικοινωνίας. Δύο κόμβοι έρχονται σε επαφή όταν ο ένας βρίσκεται στην εμβέλεια του

άλλου και μπορούν να αποστείλουν και να λάβουν πακέτα δεδομένων. Κάθε κόμβος βλέπει μία επαφή ως ευκαιρία προώθησης δεδομένων. Κατά συνέπεια, ένας κόμβος ο οποίος συμμετέχει σε ένα ομορτουμιστικό δίκτυο δρα ως δρομολογητής, παρέχοντας του πόρους του (ενέργεια, χώρος αποθήκευσης) για την επιτυχή λειτουργία της δρομολόγησης. Τα πακέτα ακολουθούν διαδρομές στο δίκτυο οι οποίες σχηματίζονται δυναμικά μέσω των επαφών αλλά για τις οποίες δεν υπάρχει καμία εγγύηση παράδοσης λόγω της διακοπτόμενης συνδεσιμότητας.

Αναφέραμε ότι σε ένα ομορτουμιστικό δίκτυο οι κόμβοι κινούνται προς οποιαδήποτε κατεύθυνση. Συνέπεια της πλήρους κινητικότητας είναι η ύπαρξη περιπτώσεων όπου ένας κόμβος είναι απομονωμένος και επικοινωνεί αραιά και παροδικά, με τους υπόλοιπους κόμβους του δικτύου. Η κινητικότητα των κόμβων ορίζει στοχαστικές (ευκαιριακές) συνδέσεις και επικοινωνίες κατά τις επαφές των κόμβων. Κατά συνέπεια είναι προφανές ότι οι παραδοσιακοί αλγόριθμοι δρομολόγησης στην περίπτωση αυτών των δικτύων αποτυγχάνουν. Ο κυριότερος λόγος είναι ότι η βασική τεχνική αποθήκευσης και προώθησης (Store & Forward) δεν είναι πλέον δυνατή αφού το δίκτυο δεν είναι συνδεδεμένο. Οι παραδοσιακοί αλγόριθμοι βασίζονται στην θεώρηση ότι την στιγμή δημιουργίας ενός πακέτου υπάρχει και μία διαδρομή (συνδεδεμένο δίκτυο) προς τον κόμβο προορισμού του. Λύση στο πρόβλημα αυτό αποτελεί η τεχνική αποθήκευσης, μεταφοράς και προώθησης (Store, Carry & Forward).

Όπως αναφέραμε και παραπάνω στα ομορτουμιστικά δίκτυα η επικοινωνία βασίζεται στην κινητικότητα των κόμβων. Επομένως, είναι απαραίτητη μία τεχνική η οποία να χρησιμοποιεί την ιδιότητα αυτή των κόμβων του δικτύου. Η τεχνική η οποία το κάνει πράξη είναι η τεχνική αποθήκευσης, μεταφοράς και προώθησης ενός πακέτου. Η μέθοδος αυτή ουσιαστικά ορίζει ότι ένας κόμβος αποθηκεύει πακέτα δεδομένων και τα μεταφέρει μέχρι να του δοθεί η ευκαιρία να τα προωθήσει σε κάποιον άλλο κόμβο του δικτύου. Η δυνατότητα προώθησης δίνεται σε ένα κόμβο όταν έρχεται σε επαφή με κάποιον άλλο κόμβο του δικτύου. Με τον τρόπο αυτό δημιουργείται μία διαδρομή προς τον κόμβο παραλήπτη για οποιοδήποτε πακέτο στο δίκτυο.

Το Σχήμα 2.1 παρουσιάζει ένα παράδειγμα χρήσης της τεχνικής αυτής κατά την διαδικασία δρομολόγησης και προώθησης ενός πακέτου μέχρι τον κόμβο προορισμό. Το πακέτο δημιουργείται στον κόμβο πηγή (source) την στιγμή t_0 και έχει ως προορισμό τον κόμβο D . Η διαδρομή του πακέτου είναι $A \rightarrow B \rightarrow C \rightarrow D$. Η διαδρομή αυτή δημιουργείται στις στιγμές t_1 , t_2 και t_3 , όπου οι ενδιάμεσοι κόμβοι έρχονται

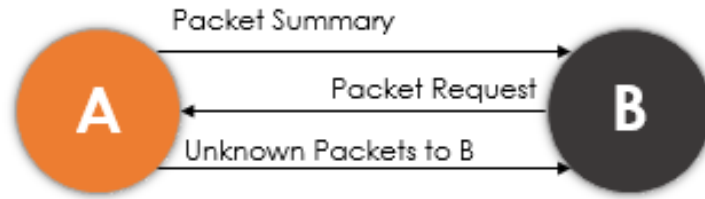


Σχήμα 2.1: Ένα παράδειγμα της τεχνικής Store, Carry & Forward κατά την δρομολόγηση ενός πακέτου τις χρονικές στιγμές t_1 , t_2 και t_3 από τον κόμβο A στον κόμβο D .

σε επαφή. Σε κάθε μία από τις επαφές, πραγματοποιείται προώθηση του πακέτου, ενώ τον υπόλοιπο χρόνο κάθε ενδιάμεσος κόμβος μεταφέρει το πακέτο. Τελικά, την στιγμή t_4 το πακέτο φτάνει στον προορισμό του.

Στο σημείο αυτό είναι σημαντικό να αναφέρουμε ότι υπάρχουν διαδρομές οι οποίες χρειάζονται αρκετό χρόνο έτσι ώστε να σχηματιστούν σε τέτοιου τύπου δίκτυα. Έτσι η αυξημένη καθυστέρηση παράδοσης του πακέτου είναι δυνατή και αποτελεί συχνό φαινόμενο στα ομορτουμιστικά δίκτυα. Για τον λόγο αυτό τα συγκεκριμένα δίκτυα ανήκουν στα δίκτυα ανεκτικά στην καθυστέρηση (Delay Tolerant Networks-DTNs). Ουσιαστικά DTN ονομάζεται ένα δίκτυο στο οποίο υπάρχει στοχαστικότητα ή περιοδικότητα στην συνδεσιμότητα. Τα ομορτουμιστικά δίκτυα όπως είδαμε εμφανίζουν στοχαστικότητα κατά τις επαφές λόγω της διακοπτόμενης συνδεσιμότητας. Αυτός είναι και ο λόγος για τον οποίο εντάσσονται στην κατηγορία αυτή.

Στα ομορτουμιστικά δίκτυα η δρομολόγηση πακέτων βασίζεται στις επαφές (Contact-based Routing) όπως είδαμε. Σε μία επαφή οι κόμβοι αρχικά ανταλλάσσουν πληροφορίες για τα πακέτα τα οποία επιθυμούν να μεταδοθούν. Η διαδικασία

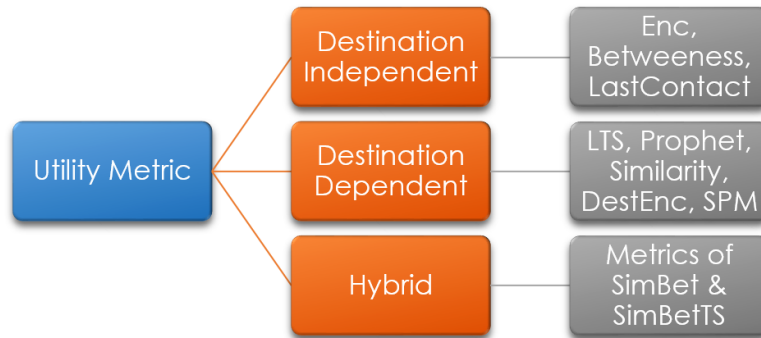


Σχήμα 2.2: Περιγραφή μιας Anti-entropy Session (συνεδρίας)

αυτή ονομάζεται Anti-entropy Session. Όπως φαίνεται και στο Σχήμα 2.2 όταν έρχονται σε επαφή δύο κόμβοι A και B τότε ο κόμβος A στέλνει στο κόμβο B μία σύνοψη των πακέτων τα οποία διαθέτει και μπορεί να αποστείλει στον B . Ο κόμβος B απαντά με μία αίτηση για τα πακέτα τα οποία επιθυμεί. Τέλος ο κόμβος A στέλνει τα πακέτα για τα οποία θεωρεί τον B "καταλληλότερο" δρομολογητή. Με την αποστολή των πακέτων έχουμε και την λήξη της διαδικασίας.

Σύμφωνα με την τεχνική Store, Carry & Forward κάθε κόμβος A ο οποίος μεταφέρει ένα πακέτο p με προορισμό έναν κόμβο D , πρέπει να αποφασίσει αν θα προωθήσει το πακέτο αυτό, όταν έρχεται σε επαφή με έναν άλλο κόμβο του δικτύου. Αν επιλέξει να προωθήσει ένα πακέτο σε έναν ακατάλληλο κόμβο είναι πιθανό να μην έχουμε επιτυχή παράδοση του πακέτου στον κόμβο προορισμό ή να επιτύχουμε παράδοση αλλά με μεγάλη απώλεια ενέργειας (πολλαπλές προωθήσεις). Για την εύρεση επομένως κατάλληλων ενδιαμέσων κόμβων, οι περισσότεροι αλγόριθμοι δρομολόγησης χρησιμοποιούν μία μετρική για να αξιολογήσουν την ικανότητα κάθε κόμβου για προώθηση ή/και παράδοση πακέτων στον προορισμό. Η μετρική αυτή είναι συνήθως ένα βαθμωτό μέγεθος το οποίο κάθε κόμβος του δικτύου υπολογίζει τοπικά και ονομάζεται μετρική ωφέλειας (Utility).

Σημαντικό είναι να αναφέρουμε ότι πλέον σε αλγορίθμους που χρησιμοποιούν ένα Utility, προστίθεται στο Anti-entropy Session (Σχήμα 2.2) πληροφορία που αφορά τις τιμές της μετρικής τις οποίες γνωρίζουν οι κόμβοι. Ουσιαστικά όταν οι δύο κόμβοι A και B έρχονται σε επαφή η διαδικασία αλλάζει λίγο. Κατά την αποστολή της αίτησης πακέτων που στέλνει ο B έχουμε και επισύναψη των Utilities για τα πακέτα που επιθυμεί να λάβει από τον A . Έτσι ώστε ο A να μπορεί να αποφασίσει ποια πακέτα θα προωθηθούν στον B και ποια όχι. Η απόφαση εξαρτάται απόλυτα από τον τρόπο με τον οποίο αξιοποιείται η πληροφορία που μας παρέχει το Utility. Η απόφαση βασίζεται στην λογική την οποία χρησιμοποιεί το



Σχήμα 2.3: Κατηγοριοποίηση μετρικής ωφέλειας (Utility Metric).

κάθε πρωτόκολλο δρομολόγησης και είναι κάτι στο οποίο θα αναφερθούμε εκτενώς παρακάτω.

Υπάρχουν διαφορετικοί τρόποι κατασκευής ενός Utility και κάθε κόμβος υπολογίζει την δική του τιμή χρησιμοποιώντας το ιστορικό των επαφών του. Πληροφορία που μπορεί να χρησιμοποιηθεί για την κατασκευή ενός Utility θα μπορούσαν να αποτελούν, οι κοινωνικές σχέσεις μεταξύ των κόμβων, η πιθανότητα επαφής ή η συχνότητα και η διάρκεια των επαφών ενός κόμβου. Διαφορετικά χαρακτηριστικά, που κανείς μπορεί να εξάγει από το ιστορικό των επαφών, χρησιμοποιούνται για την κατασκευή διαφορετικών Utilities. Η συχνότητα και η περιοδικότητα των επαφών που πραγματοποιεί ένας κόμβος ή μια περιγραφή κοινωνικών χαρακτηριστικών του, αποτελούν και αυτά στοιχεία τα οποία μπορεί κανείς να χρησιμοποιήσει για να δημιουργήσει ένα Utility. Σε ένα γενικότερο πλαίσιο τα Utilities μπορούν να κατηγοριοποιηθούν σε δύο βασικές κατηγορίες. Utilities τα οποία εξαρτώνται από τον κόμβο προορισμό (Destination Dependent) ενός πακέτου και Utilities τα οποία δεν εξαρτώνται (Destination Independent). Η κατηγορία των Destination Independent Utilities προσομοιώνει την γενική ικανότητα ενός κόμβου για προώθηση πακέτου σε οποιονδήποτε κόμβο του δικτύου. Η Destination Dependent κατηγορία προσομοιώνει την ικανότητα ενός κόμβου για παράδοση πακέτων συγκεκριμένα στον κόμβο προορισμό. Υπάρχει και μία τρίτη κατηγορία οι οποία είναι μια υβριδική προσέγγιση των δύο παραπάνω κατηγοριών. Συγκεκριμένα, χρησιμοποιεί ένα συνδυασμό από Utilities των δύο παραπάνω κατηγοριών για την κατασκευή ενός νέου. Η κατηγοριοποίηση αυτή παρουσιάζεται και στο Σχήμα 2.3.

Μία μετρική η οποία ανήκει στην κατηγορία των Destination Dependent Utilities, απαιτεί την αποθήκευση σε κάθε κόμβο του δικτύου διαφορετικών τιμών της μετρι-

κής για κάθε άλλο κόμβο του δικτύου. Αυτή η κατηγορία μετρικών ωφέλειας, είναι πολύ χρήσιμη και αποτελεσματική για κόμβους οι οποίοι ανήκουν στην "γειτονιά" του κόμβου προορισμού. Δεν είναι όμως τόσο αποτελεσματική όταν η δρομολόγηση ενός πακέτου αφορά απομακρυσμένους κόμβους, οι οποίοι μπορεί και να μην έχουν συναντηθεί ποτέ. Συχνές αναφορές στην βιβλιογραφία γίνονται για την μετρική LTS (Last Time Seen) [4, 5] η οποία ανήκει στην κατηγορία αυτή. Η μετρική LTS εξαρτάται από τον χρόνο που έχει μεσολαβήσει από την τελευταία επαφή με τον προορισμό. Η κεντρική ιδέα επικεντρώνεται στην θεώρηση ότι όταν ένας κόμβος έχει δει πρόσφατα τον κόμβο προορισμό τότε αποτελεί κατάλληλο υποψήφιο προώθησης. Η μετρική Similarity [6, 7] ανήκει επίσης στην κατηγορία αυτή, η οποία υπολογίζεται από τον αριθμό των γειτόνων που δύο κόμβοι μοιράζονται. Η πληροφορία αυτή υπολογίζεται με χρήση του τοπικού γραφήματος επαφών (ego network). Το ego network σχηματίζεται από το ιστορικό επαφών των κόμβων. Μια μετρική επίσης της κατηγορίας των Destination Dependent Utilities αποτελεί και η μετρική την οποία χρησιμοποιεί ο αλγόριθμος Prophet [8, 9]. Η μετρική αυτή είναι μια πιθανοτική μετρική η οποία βασίζεται στην πρόβλεψη παράδοσης. Συγκεκριμένα, υπολογίζεται με βάση την πιθανότητα συνάντησης με τον κόμβο προορισμό και αποσβένεται στον χρόνο. Η μετρική Prophet ενημερώνεται μόνο όταν υπάρξει μία εκ νέου συνάντηση με τον κόμβο προορισμό. Επιπλέον, η μετρικές DestEnc (Destination Encounters - επαφές με τον προορισμό) και SPM (Social Pressure Metric - Μετρική Κοινωνικής Πίεσης) [10, 11] ανήκουν και αυτές στην κατηγορία των Destination Dependent Utilities. Η πρώτη υπολογίζεται από τις επαφές ενός κόμβου με τον κόμβο προορισμό και η δεύτερη προσομοιώνει τις σχέσεις φιλίας των κόμβων, χρησιμοποιώντας τις έννοιες της συχνότητας, της διάρκειας και της τακτικότητας των επαφών μεταξύ των κόμβων.

Ένα Utility ανεξάρτητο του προορισμού (Destination Independent Utility) είναι μια μετρική η οποία απαθανατίζει την γενικότερη ικανότητα επαφής ενός κόμβου με τους υπόλοιπους κόμβους του δικτύου και υπολογίζεται ανεξάρτητα αν οι επαφές αυτές αφορούν τον κόμβο προορισμό του πακέτου. Γενικά προσομοιώνει την ικανότητα ενός κόμβου να προωθεί πακέτα σε οποιονδήποτε κόμβο του δικτύου. Στην περίπτωση αυτής της κατηγορίας, ο κάθε κόμβος του δικτύου ουσιαστικά εκτιμά μία μοναδική τιμή της μετρικής αυτής, η οποία δείχνει την σημαντικότητα του στο δίκτυο. Μία από τις γνωστότερες τέτοιες μετρικές αποτελεί η Enc (Encounters-Επαφές) [12, 13], η οποία απεικονίζει τον αριθμό των παλαιότερων επαφών ενός

κόμβου. Μία πιο πολύπλοκη μετρική αποτελεί η Betweenness Centrality [14], η οποία μετράει σε ποιο βαθμό ο κόμβος βρίσκεται στα συντομότερα μονοπάτια από όλους τους υπόλοιπους κόμβους του δικτύου. Η κατανεμημένη μορφή του ονομάζεται Ego Betweenness Centrality [6, 7] και υπολογίζεται τοπικά σε κάθε κόμβο, βάσει του τοπικού γραφήματος επαφών (ego network). Μία ακόμη μετρική της κατηγορίας αυτής είναι η μετρική LastContact (Τελευταία επαφή), η οποία αποτελεί την αντίστοιχη ανεξάρτητη από τον προορισμό μετρική της LTS και υπολογίζεται ως $1/(1 + LastTime)$. Όπου LastTime είναι ο χρόνος από την τελευταία επαφή με οποιοδήποτε άλλο κόμβο.

Τέλος, η υβριδική κατηγορία στην πράξη συνδυάζει μετρικές των δύο παραπάνω κατηγοριών για την παραγωγή ενός νέου Utility. Χαρακτηριστικά παραδείγματα τεχνικών δρομολόγησης οι οποίες χρησιμοποιούν τέτοιες μετρικές, αποτελούν οι αλγόριθμοι SimBet [6] και SimBetTS [7]. Ο αλγόριθμος SimBet χρησιμοποιεί μία μίξη του Similarity και του Ego Betweenness. Ο αλγόριθμος SimBetTS διαφέρει από τον SimBet καθώς χρησιμοποιεί και επιπρόσθετες μετρικές οι οποίες ουσιαστικά μετράνε το πόσο ισχυρές ή ανίσχυρες είναι οι σχέσεις μεταξύ των κόμβων.

2.2 Ανασκόπηση Αλγορίθμων Δρομολόγησης

Βασική συνιστώσα της διαδικασίας δρομολόγησης αποτελεί η τακτική δρομολόγησης των πακέτων. Τακτική δρομολόγησης ονομάζουμε στην πράξη τον αλγόριθμο ο οποίος χρησιμοποιείται για την διαδικασία της δρομολόγησης. Ο αλγόριθμος δρομολόγησης ορίζει σε μία επαφή ποια πακέτα θα προωθηθούν και ποια όχι βάσει κάποιων κανόνων προώθησης. Το Σχήμα 2.4 μας παρουσιάζει μία κατηγοριοποίηση των επικρατέστερων αλγορίθμων.

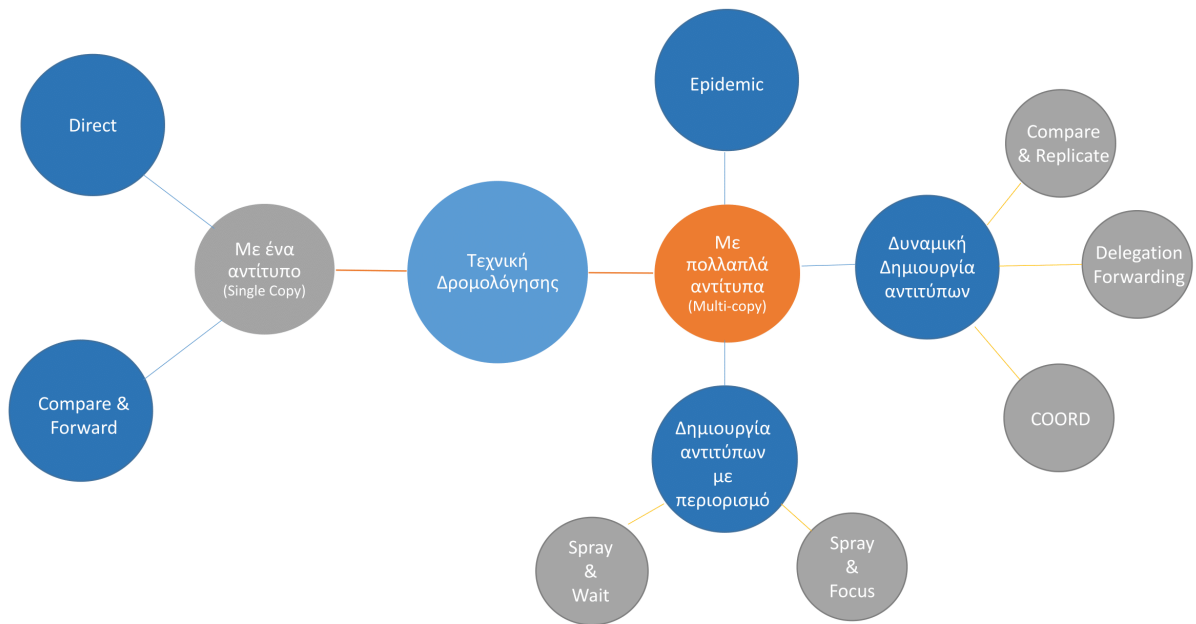
Οι αλγόριθμοι δρομολόγησης μπορούν να κατηγοριοποιηθούν αρχικά σε δύο βασικές κατηγορίες [1]. Η μία χρησιμοποιεί ένα αντίτυπο (single-copy), ενώ η δεύτερη χρησιμοποιεί πολλαπλά αντίτυπα (multi-copy) ενός πακέτου. Η πρώτη κατηγορία ουσιαστικά δεν επιτρέπει τη δημιουργία αντιτύπων στο δίκτυο. Κάθε πακέτο είναι μοναδικό και προωθείται κατά τις επαφές των κόμβων. Αντίθετα, η δεύτερη κατηγορία επιτρέπει την δημιουργία πολλαπλών αντιτύπων ενός πακέτου (packet replicas) και φυσικά την προώθηση αυτών στο δίκτυο. Η τελευταία κατηγορία παρουσιάζει καλύτερα αποτελέσματα καθώς κάθε ένα αντίτυπο ακολουθεί διαφορετικό μονο-

πάτι μέχρι τον κόμβο προορισμό. Αυτό διασφαλίζει σε μεγάλο βαθμό την παράδοση ενός πακέτου στον προορισμό του. Για τον λόγο αυτό οι αλγόριθμοι με πολλαπλά αντίτυπα παρουσιάζουν υψηλότερα ποσοστά επιτυχούς παράδοσης αλλά και μικρότερες καθυστερήσεις. Βάση της κατηγορίας με πολλαπλά αντίτυπα είναι η αύξηση της πιθανότητας τελικής λήψης ενός πακέτου από τον κόμβο προορισμό. Οι κυριότεροι λόγοι είναι δύο. Ο πρώτος έχει να κάνει με την διασφάλιση μικρής πιθανότητας απώλειας του πακέτου και ο δεύτερος αφορά την αποφυγή "εγκλωβισμού" ενός πακέτου σε μία γειτονία κόμβων (προωθήσεις μεταξύ των ίδιων κόμβων) που δεν βρίσκεται κοντά στον κόμβο προορισμού. Η κατηγορία με ένα αντίτυπο υστερεί ακριβώς στα δύο παραπάνω σημεία. Επιπλέον, παρουσιάζει μεγάλες καθυστερήσεις παράδοσης πακέτων ακόμα και στις επιτυχείς παραδόσεις.

Με την χρήση πολλαπλών αντιτύπων σίγουρα μπορούμε να επιτύχουμε καλύτερα ποσοστά επιτυχούς παράδοσης και μικρότερη καθυστέρηση. Παρόλα αυτά απαιτείται εκτεταμένη χρήση των πόρων (κατανάλωση ενέργειας, χώρος αποθήκευσης) των κόμβων. Ο κυριότερος λόγος κατανάλωσης ενέργειας είναι ο μεγάλος αριθμός προωθήσεων. Πολλά αντίτυπα επίσης σημαίνουν άμεσα και μεγαλύτερη χρήση αποθηκευτικού χώρου. Από την άλλη πλευρά, αν χρησιμοποιούμε ένα μοναδικό αντίτυπο μειώνουμε σημαντικά τις προωθήσεις και συνεπώς την κατανάλωση ενέργειας αλλά υστερούμε σε ποσοστά επιτυχούς παράδοσης και παρατηρούμε μεγαλύτερες καθυστερήσεις ακόμα και στις επιτυχείς παραδόσεις. Παρατηρούμε λοιπόν ένα ισοζύγιο (trade-off) μεταξύ παράδοσης και καθυστέρησης με τις προωθήσεις αντιτύπων στο δίκτυο.

Η κατηγορία αλγορίθμων με ένα αντίτυπο αποτελείται από δύο υποκατηγορίες. Η πρώτη ονομάζεται Direct [15]. Ουσιαστικά ένας κόμβος που ακολουθεί τον αλγόριθμο αυτό, προωθεί το μοναδικό αντίτυπο ενός πακέτου μόνο εάν έρθει σε επαφή με τον κόμβο προορισμό. Όπως είναι λογικό ο αλγόριθμος αυτός παρουσιάζει πολύ χαμηλά ποσοστά επιτυχούς παράδοσης. Ακόμη και όταν ένα πακέτο φτάνει στον προορισμό του, η καθυστέρηση είναι πολύ μεγάλη.

Η δεύτερη στρατηγική με ένα αντίτυπο που έχει προταθεί όπως φαίνεται και στο Σχήμα 2.4 είναι ο Compare & Forward αλγόριθμος. Αποτελεί ένα αλγόριθμο ο οποίος χρησιμοποιεί την διαδικασία σύγκρισης των Utilities για προώθηση. Μία προώθηση ενός πακέτου πραγματοποιείται από έναν κόμβο A σε έναν άλλο κόμβο B αν και μόνο αν η τιμή U_A είναι μικρότερη της αντίστοιχης τιμής της μετρικής U_B ($U_A < U_B$). Την πληροφορία για τις τιμές των Utilities όπως αναφέραμε παραπάνω



Σχήμα 2.4: Κατηγοριοποίηση των τεχνικών δρομολόγησης στα ομορτυνιστικά δίκτυα. Η κατηγοριοποίηση αυτή βασίζεται στην διατριβή [1]

ανταλλάσσουν οι δύο κόμβοι κατά την διάρκεια μιας επαφής και πιο συγκεκριμένα κατά το Anti-entropy Session. Ο Compare & Forward επιτυγχάνει πολύ καλύτερα αποτελέσματα στην παράδοση των πακέτων σε σχέση με τον Direct. Η απόδοση του είναι όμως και πάλι περιορισμένη λόγω της απόλυτης εξάρτησης από την ποιότητα του Utility. Ουσιαστικά εξαρτάται από το πόσο αποτελεσματικά προσομοιώνει ένα Utility την ικανότητα παράδοσης/προώθησης ενός κόμβου. Επίσης σημαντικός παράγοντας αποτυχίας αποτελεί σαφώς και η ύπαρξη ενός μοναδικού αντιτύπου για κάθε πακέτο στο δίκτυο.

Από την άλλη πλευρά, στη κατηγορία αλγορίθμων με πολλαπλά αντίτυπα υπάρχουν τρεις υποκατηγορίες. Η κάθε υποκατηγορία χρησιμοποιεί την διαδικασία δημιουργίας αντιτύπων (replication) για την αύξηση των ποσοστών επιτυχούς παράδοσης. Χαρακτηριστική αλλά ταυτόχρονα και ακραία περίπτωση αλγορίθμου με πολλαπλά αντίτυπα (multi-copy) αποτελεί ο αλγόριθμος Epidemic [16]. Ο αλγόριθμος αυτός παράγει αντίτυπα στο δίκτυο με έναν άπληστο τρόπο. Μία επαφή συνεπάγεται με δημιουργία αντιτύπου κάθε πακέτου το οποίο έχει στην κατοχή του ένας κόμβος. Ουσιαστικά ο αλγόριθμος αυτός υλοποιεί την τεχνική της πλημμύρας. Η περίπτωση αυτή είναι ακραία καθώς εμφανίζει τεράστιο κόστος (κατανάλωση ενέργειας, σπατάλη χώρου αποθήκευσης) για την παράδοση των πακέτων. Παρόλα

αυτά η πιθανότητα παράδοσης ενός πακέτου αυξάνεται κατακόρυφα.

Μία πολύ γνωστή τεχνική που υλοποιεί έναν μηχανισμό ελέγχου, είναι η κατηγορία τεχνικών δρομολόγησης με δημιουργία περιορισμένων αντιτύπων (Constrained Replication). Στην κατηγορία αυτή τίθεται ένα περιορισμός αντιτύπων για κάθε πακέτο το οποίο δημιουργείται. Ορίζεται στην πράξη ένα άνω όριο στα αντίτυπα κάθε πακέτου τα οποία μπορούν να δημιουργηθούν στο δίκτυο (L). Αρχικά έχουμε ανάθεση της τιμής L σε ένα πακέτο κατά την δημιουργία του. Σε δεύτερο επίπεδο, κατά την δρομολόγηση ενός πακέτου η τιμή αυτή μειώνεται. Η μείωση πραγματοποιείται κατά την προώθηση αντιτύπων. Τα αντίτυπα ενός πακέτου διαμοιράζονται είτε ισόποσα είτε άνισα με βάση κάποιο Utility. Η τεχνική αυτή ονομάζεται Spraying και διαρκεί μέχρι η τιμή L να μειωθεί σε ένα αντίτυπο. Από εκεί και πέρα ακολουθείται η λογική προώθησης ενός μοναδικού αντιτύπου με δύο τρόπους όπως είδαμε και παραπάνω. Κατά συνέπεια, μπορεί να υιοθετηθεί είτε η λογική του Direct είτε η λογική του Compare & Forward. Στην πρώτη περίπτωση γίνεται προώθηση του πακέτου μόνο στην περίπτωση επαφής με τον κόμβο προορισμό. Αντίθετα, στην δεύτερη περίπτωση χρησιμοποιείται η συγκριτική λογική των Utilities ως κριτήριο απόφασης για την προώθηση αντιτύπων. Η πρώτη περίπτωση ονομάζεται Spray & Wait [4, 17], ενώ η δεύτερη Spray & Focus [7, 12, 17].

Βασικό μειονέκτημα της κατηγορίας με περιορισμένο αριθμό αντιτύπων είναι ότι παρουσιάζει ένα ισοζύγιο (trade-off) μεταξύ κόστους και αποτελεσματικότητας καθώς όλα εξαρτώνται από το προκαθορισμένο άνω όριο αντιτύπων. Επιπρόσθετα, η κατάσταση του δικτύου συνεχώς αλλάζει με συνέπεια τον μη αξιόπιστο ορισμό της βέλτιστης τιμής του L ακόμη και αν γνωρίζουμε το τι θέλουμε να επιτύχουμε κατά την διαδικασία της δρομολόγησης. Δεν μπορεί κάποιος να γνωρίζει από πριν την δομή του δικτύου και πως αυτό θα εξελιχθεί στο μέλλον καθώς αναφερόμαστε σε δίκτυα στα οποία οι επαφές είναι τυχαίες. Δεν μπορεί επομένως εύκολα να προσδιοριστεί μία κατάλληλη τιμή για την παράμετρο L .

Μία εναλλακτική προσέγγιση είναι ένας δυναμικός έλεγχος των αντιτύπων στο δίκτυο. Οι αλγόριθμοι αυτοί χρησιμοποιούν ένα Utility για την αξιολόγηση των κόμβων για παράδοση ή/και προώθηση πακέτων. Κατά συνέπεια, η υποκατηγορία αυτή προτείνει έναν πιο ρεαλιστικό τρόπο προσέγγισης και υλοποίησης ενός μηχανισμού ελέγχου των αντιτύπων σε σύγκριση με τους αλγορίθμους δρομολόγησης με περιορισμένο αριθμό αντιτύπων. Ένα αρχικό τέτοιο σχήμα που έχει προταθεί αποτελεί ο αλγόριθμος Compare & Replicate (CNR). Στην πράξη χρησιμοποιεί την συγκρι-

τική λογική των Utilities (ακριβώς όπως ο Compare & Forward) δύο κόμβων σε μία επαφή, έτσι ώστε να αποφασιστεί η δημιουργία αντίτυπου ή όχι. Έχουμε ένα καλύτερο τρόπο ελέγχου δημιουργίας αντιτύπων καθώς έχουμε προσαρμογή στις αλλαγές του δικτύου μέσω της χρήσης ενός Utility. Συνεπώς, η αποδοτικότητα του CNR ως προς τις επιτυχείς παραδόσεις είναι πολύ καλύτερη συγκριτικά με τις Spray-based τεχνικές. Οι προωθήσεις επίσης μειώνονται αισθητά σε σύγκριση με τον αλγόριθμο Epidemic. Ωστόσο, η μείωση δεν έχει φτάσει στα επιθυμητά χαμηλά επίπεδα. Δύο καλύτερες και αποτελεσματικότερες τεχνικές δρομολόγησης με δυναμικό έλεγχο αποτελούν οι περιπτώσεις των Delegation Forwarding (DF) [2] και Coordinated Delegation Forwarding (COORD) [3]. Η λογική του DF σε έναν κόμβο του δικτύου είναι η εξής: σε μία επαφή δύο κόμβων A και B έχουμε προώθηση ενός πακέτου p αν και μόνο αν η τιμή του Utility του κόμβου B είναι μεγαλύτερη από την μέγιστη τιμή Utility (κατώφλι) την οποία έχει καταγράψει ο κόμβος A για το συγκεκριμένο πακέτο p . Η διαδικασία αυτή προϋποθέτει οπότε την καταγραφή ενός κατωφλίου (threshold) το οποίο προκύπτει από τις προγενέστερες επαφές του κόμβου. Ο αλγόριθμος COORD χρησιμοποιεί την ίδια λογική με τον DF αλλά προσθέτει και την ιδέα ενημέρωσης του κατωφλίου με ανταλλαγή επιπλέον πληροφορίας κατά το Anti-entropy Session. Στον DF, ο κόμβος A ενημερώνει το κατώφλι του με την τιμή του Utility του κόμβου B όταν προωθεί το πακέτο στον B . Αντίθετα στον COORD, υπάρχει επίσης ενημέρωση του κατωφλίου του κόμβου A , όταν ο κόμβος B έχει ήδη ένα αντίτυπο του πακέτου (έχει καταγράψει ένα κατώφλι) και η τιμή του κατωφλίου του ξεπερνάει την τιμή του αντίστοιχου κατωφλίου του κόμβου A . Με τον τρόπο αυτό ένας κόμβος είναι δυνατόν να επωφεληθεί με ενημέρωση του κατωφλίου του ακόμα και για πακέτα τα οποία δεν τίθενται προς αποστολή εκείνη την στιγμή. Η λογική του αλγορίθμου COORD βασίστηκε στην υπόθεση ότι είναι πιο αποτελεσματικό να υπάρχει ανταλλαγή πληροφορίας για τις τιμές των κατωφλίων. Πράγματι, ο αλγόριθμος COORD προάγει την αποτελεσματικότητα σε ότι αφορά την περαιτέρω μείωση των προωθήσεων σε σχέση με τον αλγόριθμο DF, χωρίς να θυσιάζει την επιτυχή παράδοση πακέτων. Η κατηγορία με δυναμικό έλεγχο δημιουργίας αντιτύπων είναι η κατηγορία που επικεντρωθήκαμε και εμείς, καθώς παρουσιάζει τα θετικότερα στοιχεία για την κατανάλωση ενέργειας αλλά εμφανίζει και υψηλά ποσοστά παράδοσης.

ΚΕΦΑΛΑΙΟ 3

ΔΡΟΜΟΛΟΓΗΣΗ ΜΕ ΔΥΝΑΜΙΚΗ ΔΗΜΙΟΥΡΓΙΑ ΑΝΤΙΤΥΠΩΝ

3.1 Compare & Replicate (CNR)

3.2 Delegation Forwarding (DF)

3.3 Coordinated Delegation Forwarding (COORD)

Η δρομολόγηση με δυναμική δημιουργία αντιτύπων αποτελεί μια υποκατηγορία της κατηγορίας δρομολόγησης με πολλαπλά αντίτυπα. Μας παρέχει ένα καλύτερο έλεγχο στην άσκοπη δημιουργία αντιτύπων στο δίκτυο με την χρήση μιας μετρικής ωφέλειας (Utility). Ουσιαστικά η σύγκριση διαφορετικών τιμών ενός Utility μας δίνει την δυνατότητα να ελέγχουμε την ποιότητα του κόμβου στον οποίο θέλουμε να προωθηθεί ένα πακέτο. Η παραπάνω διαδικασία παρουσιάζει σημαντικές βελτιώσεις στην κατανάλωση ενέργειας λόγω του περιορισμού των προωθήσεων αλλά ταυτόχρονα διατηρεί σε υψηλά επίπεδα τα ποσοστά επιτυχούς παράδοσης. Ανάλογα με το πως χρησιμοποιεί ένας αλγόριθμος αυτής της κατηγορίας την παραπάνω συγκριτική λογική έχουμε μεταβολές και στα αποτελέσματα που μας δίνει η διαδικασία της δρομολόγησης. Παρά την αξιοσημείωτη μείωση των προωθήσεων σε σχέση με άλλες κατηγορίες αλγορίθμων (όπως ο Epidemic), η κατανάλωση ενέργειας επιμένει να κινείται σε υψηλά επίπεδα στην περίπτωση του αλγορίθμου Compare & Replicate (CNR). Στις άλλες δύο περιπτώσεις των αλγορίθμων Delegation Forwarding

(DF) και Coordinated Delegation Forwarding (COORD) παρατηρούνται πολύ χαμηλότερα επίπεδα κατανάλωσης ενέργειας με τα ποσοστά επιτυχούς παράδοσης να κυμαίνονται σε υψηλά επίπεδα σε κάθε περίπτωση.

Η έννοια του Utility παίζει καθοριστικό ρόλο στην δρομολόγηση με δυναμική δημιουργία αντιτύπων. Η αποτελεσματικότητα όμως κάθε αλγορίθμου εξαρτάται μόνο από το πως χρησιμοποιείται η πληροφορία που μας δίνει ένα Utility. Ο CNR δεν πραγματοποιεί μία τόσο αποδοτική χρήση της μετρικής καθώς παρουσιάζει σημαντική εξάρτηση από την ποιότητα αυτής. Με τον όρο "ποιότητα" εννοούμε το πόσο αποδοτικά ορίζει ένα Utility την "καταλληλότητα" προώθησης ή/και παράδοσης ενός κόμβου. Η απόλυτη εξάρτηση του αλγορίθμου CNR με το Utility έχει ως συνέπεια να μην παρέχει εκτεταμένη μείωση της κατανάλωσης ενέργειας. Ο λόγος είναι το μεγάλος πλήθος αντιτύπων που δημιουργούνται και προωθούνται στο δίκτυο καθώς κανένα Utility δεν προσομοιώνει άριστα την "καταλληλότητα" των κόμβων. Αντίθετα, οι αλγόριθμοι DF και COORD χρησιμοποιούν το Utility ως "εργαλείο" για να επιτύχουν εκτεταμένη μείωση αντιτύπων στο δίκτυο. Η αποδοτικότητα τους εξαρτάται σε ελάχιστο βαθμό από το Utility καθώς χρησιμοποιούν την έννοια του κατωφλίου (μέγιστη καταγεγραμμένη τιμή του Utility). Η συγκριτική λογική στους αλγορίθμους αυτούς αλλάζει. Πλέον πραγματοποιείται σύγκριση τιμών του Utility με την τιμή του κατωφλίου για την απόφαση προώθησης ενός πακέτου. Έτσι οι δύο αλγόριθμοι απομακρύνονται από την απόλυτη εξάρτηση από την ποιότητα του Utility. Το αποτέλεσμα είναι ακόμα και σε περιπτώσεις χρήσης μη ποιοτικών Utilities να έχουμε αξιοσημείωτες και σημαντικές μειώσεις δημιουργίας αντιτύπων. Κατά συνέπεια, οι αλγόριθμοι DF και COORD δεν επιβαρύνουν το δίκτυο με επιπλέον περιττά αντίτυπα πακέτων που δεν βοηθάνε στην διαδικασία της τελικής παράδοσης στον κόμβο προορισμό. Για τον λόγο αυτό αποτελούν και τους αποτελεσματικότερους αλγορίθμους δρομολόγησης στα ομορτυνιστικά δίκτυα. Οι διαφορές τους έγκειται μόνο στον τρόπο ενημέρωσης του κατωφλίου. Κάτι το οποίο θα το αναλύσουμε παρακάτω.

Εμβαθύνοντας λίγο παραπάνω στην διαδικασία προώθησης πακέτων πρέπει να αναφέρουμε ότι η διαδικασία που περιγράψαμε εν συντομία για κάθε αλγόριθμο δεν είναι η μοναδική που πραγματοποιείται. Αρχικά ένας κόμβος A ο οποίος ήρθε σε επαφή με κάποιο άλλο κόμβο του δικτύου B , ερευνά τον Buffer του για πακέτα τα οποία ο κόμβος B είναι ο κόμβος προορισμός. Έτσι έχουμε μία φάση άμεσης αποστολής πακέτων (Direct Forwarding) στον κόμβο B . Η διαδικασία άμεσης απο-

στολής πακέτων πραγματοποιείται πάντοτε ανεξαρτήτως του αλγορίθμου δρομολόγησης με δυναμική δημιουργία αντιτύπων που χρησιμοποιείται. Η δεύτερη φάση είναι στην πράξη η φάση για την οποία ο εκάστοτε αλγόριθμος της κατηγορίας με δυναμική δημιουργία αντιτύπων (CNR, DF και COORD) ορίζει αν θα προωθηθούν στον κόμβο B τα πακέτα για τα οποία δεν είναι ο κόμβος προορισμός τους.

3.1 Compare & Replicate (CNR)

Ο αλγόριθμός αυτός, όπως ήδη έχουμε αναφέρει, μειώνει σημαντικά τις προωθήσεις αντιτύπων στο δίκτυο και παρουσιάζει υψηλά ποσοστά παράδοσης με χαμηλές καθυστερήσεις. Ο λόγος είναι η χρήση ενός Utility για την αξιολόγηση των κόμβων. Ωστόσο, η εξάρτηση του αλγορίθμου από την ποιότητα του Utility είναι απόλυτη. Η "συγκριτική λογική" των τιμών του Utility ως κριτήριο απόφασης για την προώθηση ενός πακέτου είναι η αιτία της άμεσης εξάρτησης της αποδοτικότητας του αλγορίθμου με την ποιότητα του Utility. Έτσι αν έχουμε μία ακατάλληλη τέτοια μετρική τότε οι προωθήσεις παραμένουν σε αρκετά υψηλά επίπεδα. Αυτό δεν σημαίνει όμως ότι μπορεί να συγκριθούν με τον αριθμό αντιτύπων που δημιουργούνται στο δίκτυο με την διαδικασία της πλημμύρας (Epidemic Routing). Με τον αλγόριθμο CNR παρά τις σημαντικές μειώσεις έχουμε αποτυχία εκτεταμένης μείωσης της κατανάλωσης ενέργειας αλλά και του γενικότερου κόστους παράδοσης (κατανάλωση ενέργειας και αποθηκευτικός χώρος). Επιπρόσθετα, να τονίσουμε ότι δεν υπάρχει κάποιο Utility που να προσομοιώνει σε απόλυτο βαθμό την ικανότητα παράδοσης/προώθησης ενός κόμβου στο δίκτυο. Έτσι ο αλγόριθμος CNR δεν μπορεί να επιτύχει την επιθυμητή μείωση των αντιτύπων στο δίκτυο σε ένα γενικότερο πλαίσιο.

Ο Αλγόριθμος Compare & Replicate 3.1 μας περιγράφει την διαδικασία που ακολουθεί ο κάθε κόμβος του δικτύου κατά την διάρκεια μίας επαφής και πιο συγκεκριμένα μετά την φάση του Direct Forwarding. Όταν ένα κόμβος N_i έρθει σε επαφή με ένα άλλο κόμβο N_j του δικτύου τότε για κάθε πακέτο (packet) p το οποίο έχει στην κατοχή πραγματοποιείται έλεγχος αν η τιμή του Utility $U_{i,p}$ είναι μεγαλύτερη ή μικρότερη της τιμής $U_{j,m}$. Σε περίπτωση που η τιμή του Utility του κόμβου N_j είναι μεγαλύτερη τότε ο κόμβος N_i προωθεί το πακέτο p .

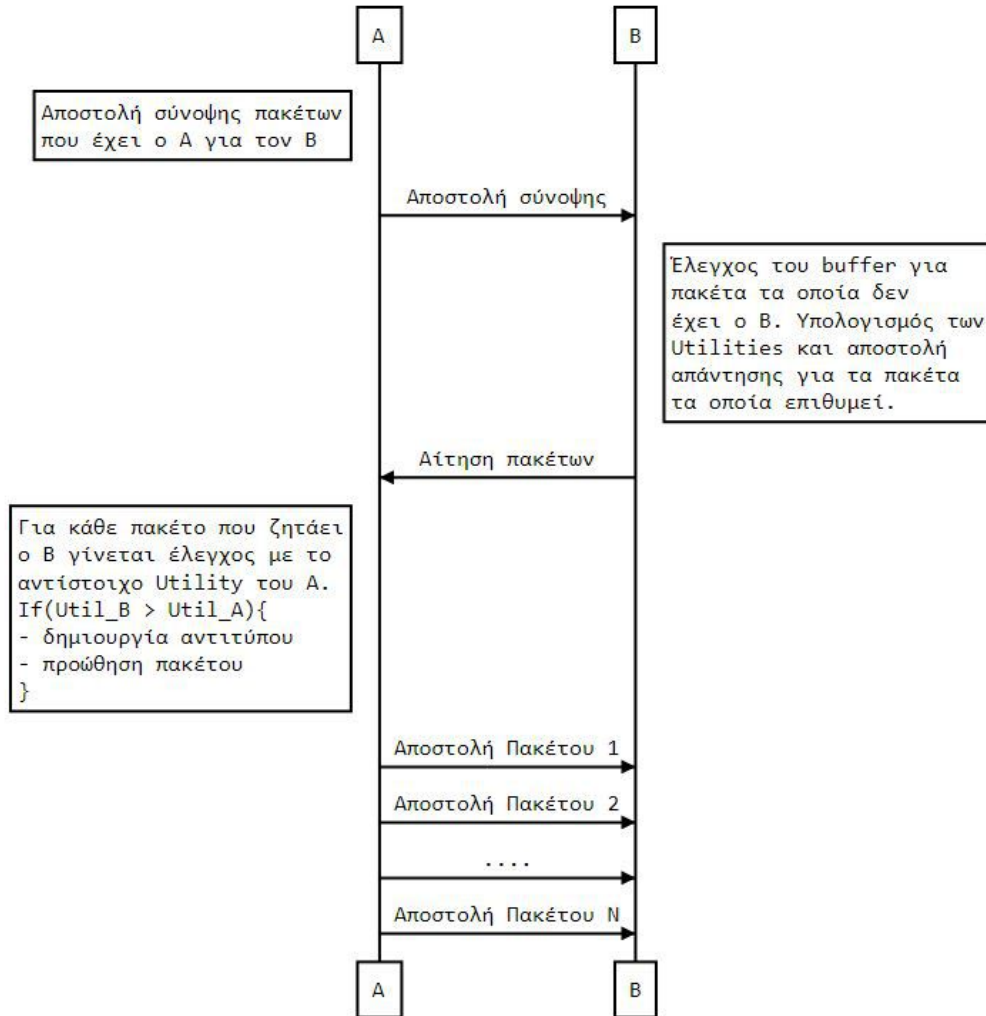
Όπως παρατηρεί κανείς στο Σχήμα 3.1 παρουσιάζεται και σχηματικά η διαδικασία βημάτων για την υλοποίηση της δρομολόγησης που παρέχει ο Compare

Αλγόριθμος 3.1 Compare & Replicate

```
1: Let  $N_1, N_2, \dots, N_N$  be nodes
2: Let  $Pkt_1, Pkt_2, \dots, Pkt_M$  be packets
3: Node  $N_i$  has utility  $U_{i,p}$  for  $Pkt_p$ .
4: On contact between  $N_i$  and node  $N_j$ :
5: for  $p$  in  $1, 2, \dots, M$  do
6:   if  $Pkt_p$  is currently held by  $N_i$  then
7:     if  $U_{i,p} < U_{j,p}$  then
8:       forward  $Pkt_p$  from  $N_i$  to  $N_j$ 
9:     end if
10:  end if
11: end for
```

& Replicate. Στο σχήμα περιγράφεται και η διαδικασία του Anti-entropy Session (ανταλλαγής πληροφορίας) που αναλύσαμε στην Ενότητα 2.1. Αρχικά ο κόμβος A συντάσσει και αποστέλλει μια σύνοψη των πακέτων που έχει στην διάθεση του και μπορεί να αποστείλει στον B . Τα πακέτα αυτά είναι πιθανό να τα έχει δημιουργήσει ο ίδιος είτε να είναι πακέτα τα οποία έχει λάβει από προηγούμενες επαφές που πραγματοποίησε. Ο κόμβος B λαμβάνει την σύνοψη και αναζητά ποια πακέτα διαθέτει και ποια όχι. Υπολογίζει τις αντίστοιχες τιμές του Utility για τα πακέτα τα οποία επιθυμεί και στέλνει μία αίτηση πακέτων με επισύναψη των αντίστοιχων τιμών του Utility. Εν συνεχεία, ο κόμβος A ακολουθεί την διαδικασία σύγκρισης των τιμών του Utility του κόμβου B με αυτές τις οποίες ο ίδιος έχει υπολογίσει για τα συγκεκριμένα πακέτα. Η απόφαση προώθησης λαμβάνεται με τον τρόπο που περιγράψαμε και στον Αλγόριθμο 3.1. Ελέγχει την τιμή του Utility για κάθε πακέτο ξεχωριστά και τελικά προωθεί τα πακέτα για τα οποία ο κόμβος B έχει μεγαλύτερη τιμή.

Με τον τρόπο που περιγράψαμε ο αλγόριθμος Compare & Replicate στοχεύει σε ποιοτικότερους, ικανότερους και καταλληλότερους κόμβους να προωθήσουν ή/και να παραδώσουν ένα πακέτο. Ο υπολογισμός νέων τιμών του Utility προκύπτει κατά τις επαφές ενός κόμβου. Με τον τρόπο αυτό ένας κόμβος εξελίσσει την εικόνα που έχει για τους άλλους κόμβους του δικτύου και γνωρίζει πότε πρέπει να προωθήσει ένα πακέτο και πότε όχι. Η απόλυτη εξάρτηση από την ποιότητα του Utility είναι το βασικό μειονέκτημα του αλγορίθμου. Το αποτέλεσμα είναι ο αλγόριθμος CNR



Σχήμα 3.1: CNR: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B.

να μη μπορεί να προσεγγίσει την επιθυμητή εκτεταμένη μείωση της κατανάλωσης ενέργειας.

3.2 Delegation Forwarding (DF)

Ο αλγόριθμος Delegation Forwarding [2] μειώνει σημαντικά τις προωθήσεις αντιτύπων στο δίκτυο σε σύγκριση με τον αλγόριθμο Compare & Replicate και παρουσιάζει υψηλά ποσοστά παράδοσης. Στην περίπτωση του αλγορίθμου αυτού χρησιμοποιείται η έννοια του κατωφλίου (threshold). Δηλαδή της μέγιστης τιμής του Utility την οποία καταγράφει και ανανεώνει κάθε κόμβος του δικτύου κατά τις επαφές του.

Αλγόριθμος 3.2 Delegation Forwarding

```
1: Let  $N_1, N_2, \dots, N_N$  be nodes
2: Let  $Pkt_1, Pkt_2, \dots, Pkt_M$  be packets
3: Node  $N_i$  has utility  $U_{i,p}$ , threshold  $\tau_{i,p}$  for  $M_{i,p}$ .
4: Initialization:  $\forall i, m : \tau_{i,p} \leftarrow U_{i,p}$ 
5: On contact between  $N_i$  and node  $N_j$  :
6: for  $p$  in  $1, 2, \dots, M$  do
7:   if  $Pkt_p$  is currently held by  $N_i$  then
8:     if  $\tau_{i,p} < U_{j,p}$  then
9:       if  $N_j$  does not have  $Pkt_p$  then
10:        forward  $Pkt_p$  from  $N_i$  to  $N_j$ 
11:       end if
12:        $\tau_{i,p} \leftarrow U_{j,p}$ 
13:     end if
14:   end if
15: end for
```

Η καταγραφή του κατωφλίου πραγματοποιείται είτε για κάθε κόμβο προορισμό (Destination Dependent Utility) είτε υπάρχει μία γενική καταγραφή (Destination Independent Utility) αναλόγως σε ποια κατηγορία ανήκει το Utility το οποίο χρησιμοποιείται (Ενότητα 2.1). Η έννοια του κατωφλίου βοηθάει στο να έχουμε μια ανεξάρτητη σχέση της αποτελεσματικότητας του αλγορίθμου με την ποιότητα του Utility. Έτσι ακόμα και σε περιπτώσεις που ένα Utility δεν προσομοιώνει αποτελεσματικά την καταλληλότητα ενός κόμβου για προώθηση/παράδοση έχουμε ένα αποδοτικότερο τρόπο αξιολόγησης των κόμβων μέσω της τιμής του κατωφλίου.

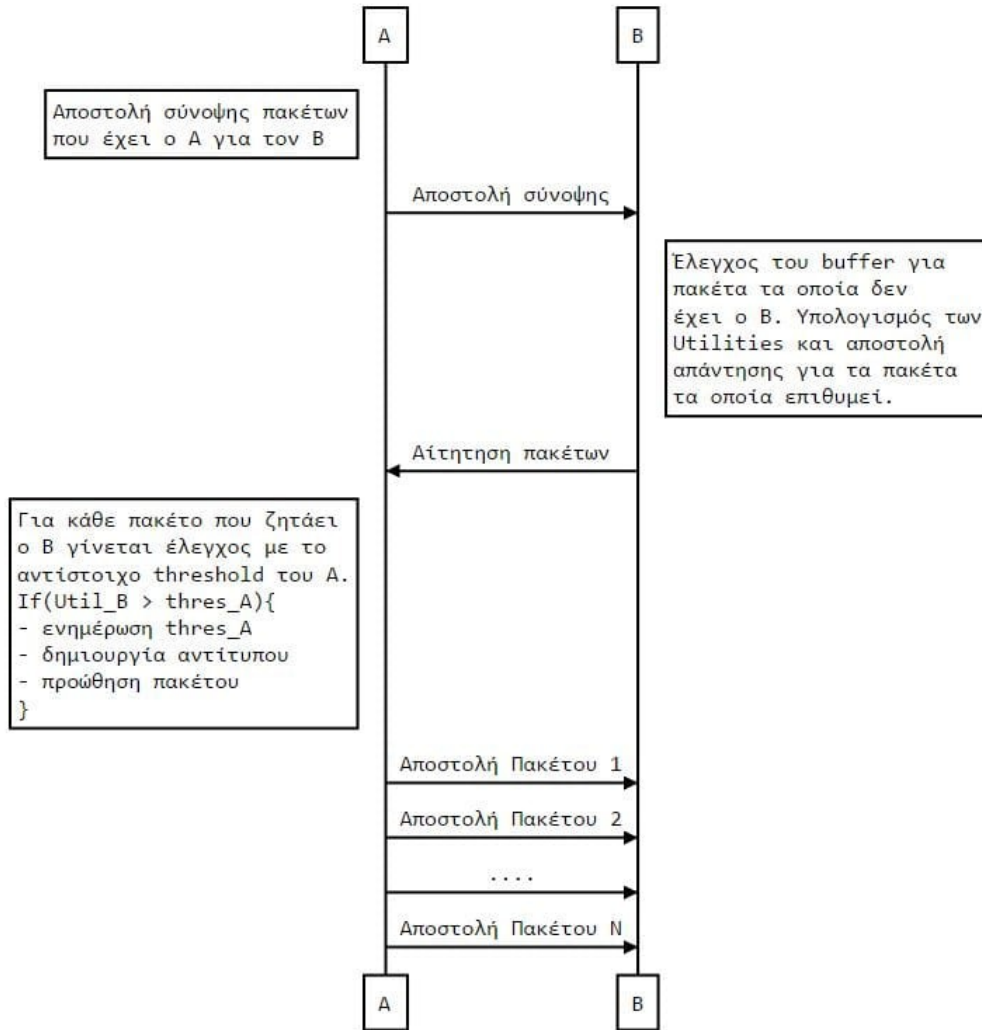
Ο Αλγόριθμος Delegation Forwarding 3.2 μας περιγράφει την διαδικασία που ακολουθεί ο κάθε κόμβος του δικτύου κατά την διάρκεια μίας επαφής, χρησιμοποιώντας την έννοια του κατωφλίου και ένα διαφορετικό κριτήριο προώθησης από τον CNR (Αλγόριθμος 3.1). Στην πραγματικότητα το κριτήριο αυτό περιγράφει και την γενικότερη ιδέα του αλγορίθμου. Κάθε κόμβος όπως αναφέραμε διατηρεί ένα κατώφλι για κάθε πακέτο το οποίο έχει στην κατοχή του. Ο κόμβος ανανεώνει την τιμή του κατωφλίου με την μέγιστη τιμή του Utility που μαθαίνει κατά τις επαφές του. Η πληροφορία ενημέρωσης του κατωφλίου προκύπτει από την διακίνηση πληροφορίας των τιμών του Utility που υπάρχουν στο δίκτυο και τις οποίες μαθαί-

νει ένας κόμβος μέσω των επαφών που πραγματοποιεί. Το πλήθος των κατωφλίων που καταγράφει ένας κόμβος εξαρτάται από το αν χρησιμοποιείται ένα Destination Dependent ή ένα Destination Independent Utility. Στην πρώτη περίπτωση έχουμε μία τιμή κατωφλίου για κάθε κόμβο τον οποίο γνωρίζει ο κόμβος που πράττει την καταγραφή. Αντίθετα, στην δεύτερη περίπτωση γίνεται καταγραφή ενός γενικού κατωφλίου για όλο το δίκτυο.

Το κριτήριο προώθησης στον αλγόριθμο DF είναι το εξής: για ένα πακέτο p δημιουργείται αντίτυπο στο δίκτυο από τον κόμβο A στον κόμβο B αν και μόνο αν $U_B > \max\{U_k\}$, $k \in N_A$, όπου N_A είναι το σύνολο των κόμβων με τους οποίους ο A έχει έρθει σε επαφή από την στιγμή που απέκτησε το πακέτο p . Έτσι η διαδικασία προώθησης πακέτων όπως περιγράφεται και στον Αλγόριθμο 3.2 έχει ως εξής: όταν ένα κόμβος N_i έρθει σε επαφή με ένα άλλο κόμβο N_j του δικτύου τότε για κάθε πακέτο (packet) p το οποίο έχει στην κατοχή του ο N_i πραγματοποιείται έλεγχος αν η τιμή του κατωφλίου $\tau_{i,p}$ είναι μεγαλύτερη ή μικρότερη της τιμής $U_{j,p}$. Σε περίπτωση που η τιμή του Utility του κόμβου N_j είναι μεγαλύτερη τότε ο κόμβος N_i ενημερώνει την τιμή του κατωφλίου του ($\tau_{i,p} \leftarrow U_{j,p}$) και προωθεί το συγκεκριμένο πακέτο p στον N_j .

Στο Σχήμα 3.2 παρουσιάζεται και σχηματικά η διαδικασία προώθησης του αλγορίθμου DF. Περιγράφεται επίσης η διαδικασία του Anti-entropy Session που έχουμε αναλύσει διεξοδικά στην Ενότητα 2.1. Αρχικά ο κόμβος A στέλνει μια σύνοψη των πακέτων που έχει στην διάθεση του και μπορεί να παρέχει στον κόμβο B . Ο κόμβος B λαμβάνει την σύνοψη και αναζητά τα πακέτα τα οποία δεν έχει στην κατοχή του. Υπολογίζει τις αντίστοιχες τιμές του Utility για κάθε πακέτο και στέλνει μία αίτηση με τα πακέτα τα οποία επιθυμεί να λάβει. Στην αίτηση επισυνάπτει τις αντίστοιχες τιμές του Utility για κάθε πακέτο. Στην συνέχεια, ο κόμβος A υλοποιεί την βασική διαδικασία απόφασης για προώθηση των αιτούμενων πακέτων. Έτσι ελέγχει την τιμή του Utility για κάθε πακέτο ξεχωριστά συγκρίνοντας την με το αντίστοιχο κατώφλι (threshold). Τελικά, προωθεί τα πακέτα για τα οποία ο κόμβος B έχει τιμή Utility μεγαλύτερη από το κατώφλι. Επιπλέον, ο κόμβος A σε μία τέτοια περίπτωση θα πρέπει να ανανεώσει και την τιμή του κατωφλίου του.

Με την ενσωμάτωση της έννοιας του κατωφλίου στην "συγκριτική λογική" ένας κόμβος εξελίσσει την εικόνα που έχει για τους κόμβους του δικτύου. Το αποτέλεσμα είναι ο αλγόριθμος Delegation Forwarding να "στοχεύει" σε πιο αξιόπιστους και ικανούς κόμβους να προωθήσουν ή/και να παραδώσουν ένα πακέτο. Κάθε κόμ-



Σχήμα 3.2: DF: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B.

βος μέσω του κατωφλίου μαθαίνει και εκμεταλλεύεται αποτελεσματικά την διακινούμενη πληροφορία που αφορά την αξιολόγηση των κόμβων. Έτσι αποκτά ένα "αυστηρότερο" κριτήριο προώθησης αντιτύπων και μια καλύτερη εικόνα για το πότε πρέπει να προωθήσει ή όχι ένα αντίτυπο από τον αλγόριθμο CNR. Σε ένα γενικότερο πλαίσιο αυτός είναι και ο λόγος που ο αλγόριθμος DF είναι αποτελεσματικός στην μείωση των περιττών αντιτύπων στο δίκτυο. Επιπρόσθετα, αξιοσημείωτο χαρακτηριστικό του αλγορίθμου αποτελούν τα ποσοστά επιτυχούς παράδοσης τα οποία διατηρούνται σε πολύ υψηλά επίπεδα παρά το μικρό πλήθος των αντιτύπων που δημιουργούνται κατά τις επαφές των κόμβων στο δικτύου.

3.3 Coordinated Delegation Forwarding (COORD)

Ο Coordinated Delegation Forwarding αλγόριθμος [3] μειώνει περαιτέρω τις προωθήσεις αντιτύπων στο δίκτυο από τον αλγόριθμο Delegation Forwarding. Παρουσιάζει επίσης πολύ υψηλά ποσοστά επιτυχούς παράδοσης. Ο αλγόριθμος COORD χρησιμοποιεί ένα κατώφλι και εκμεταλλεύεται την ιδέα ενημέρωσης του κατωφλίου με ανταλλαγή επιπλέον πληροφορίας κατά το Anti-entropy Session. Το βασικό ερώτημα από το οποίο προέκυψε η ιδέα του αλγορίθμου είναι γιατί ένας κόμβος να μην εκμεταλλευτεί και την εικόνα που προκύπτει από τις τιμές των κατωφλίων που γνωρίζουν οι υπόλοιποι κόμβοι με τους οποίους έρχεται σε επαφή. Έτσι ένας κόμβος μαθαίνει αν έχει ορίσει ορθά το κατώφλι του και αν οι άλλοι κόμβοι του δικτύου γνωρίζουν κάποια μεγαλύτερη τιμή κατωφλίου. Κατά συνέπεια, προκύπτουν ενημερώσεις του κατωφλίου με τιμές Utility που δεν ακολουθούν μόνο την λογική του DF αλλά προκύπτουν και από πληροφορία που υπάρχει στα ιστορικά των επαφών των γειτόνων ενός κόμβου. Ένας κόμβος επομένως ενημερώνει το κατώφλι του και με τιμές που στον DF δεν θα γνώριζε. Έτσι αποκτά μια βελτιωμένη εικόνα του δικτύου την οποία συνεχώς εξελίσσει με την πάροδο του χρόνου. Αυτό αποτελεί και το σημείο στο οποίο υπερτερεί και το οποίο εκμεταλλεύεται ο συγκεκριμένος αλγόριθμος με αποτέλεσμα να αποτελεί μία αποδοτικότερη περίπτωση αλγορίθμου δρομολόγησης και από τον DF.

Ο Αλγόριθμος Coordinated Delegation Forwarding 3.3 μας περιγράφει την διαδικασία που ακολουθεί ο κάθε κόμβος του δικτύου κατά την διάρκεια μίας επαφής χρησιμοποιώντας την έννοια του κατωφλίου. Η διαδικασία που θα περιγράψουμε παρακάτω αφορά την περίοδο μετά το Direct Forwarding όπως και στις δύο προηγούμενες περιπτώσεις αλγορίθμων. Ο COORD χρησιμοποιεί το κριτήριο προώθησης όπως ακριβώς ο αλγόριθμος DF. Η διαφορά των δύο αλγορίθμων ουσιαστικά έγκειται μόνο στην ενημέρωση του κατωφλίου. Κάθε κόμβος διατηρεί και στον COORD ένα κατώφλι για κάθε πακέτο που το ανανεώνει με την μέγιστη τιμή του Utility που μαθαίνει κατά τις επαφές του. Επιπλέον, ανανέωση του κατωφλίου πραγματοποιείται και για πακέτα για τα οποία δεν έχει ζητηθεί αποστολή από έναν κόμβο σε κάποιον άλλο. Έτσι για δύο κόμβους A και B οι οποίοι έρχονται σε επαφή, ο κόμβος B κατά το Anti-entropy Session ενημερώνει τον κόμβο A για τις τιμές των κατωφλίων που γνωρίζει για τα πακέτα τα οποία δεν επιθυμεί να λάβει από τον κόμβο A . Κατά συνέπεια, ο κόμβος A μπορεί να ενημερωθεί για τον αν έχει ορίσει

Αλγόριθμος 3.3 Coordinated Delegation Forwarding

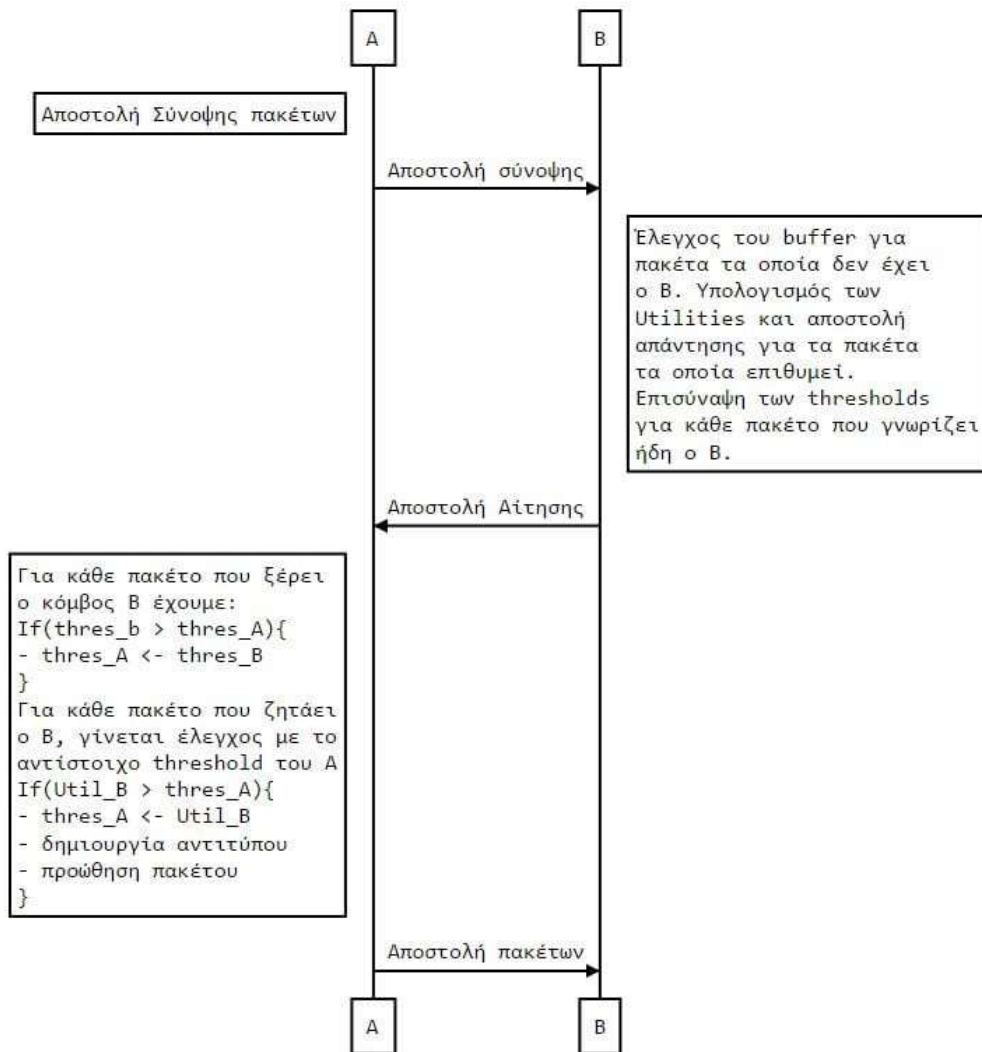
```
1: for every packet  $p$  exists in  $Bu\!f_v$  do
2:   if  $p$  exists in  $Bu\!f_v$  and  $c\tau_{v_t}^p < c\tau_{v_t}^p$  then
3:      $c\tau_{v_t}^p \leftarrow c\tau_{v_t}^p$ 
4:   else
5:     if  $c\tau_{v_t}^p < U_v$  then
6:       forward  $p$  to node  $v$ 
7:        $c\tau_{v_t}^p \leftarrow U_v$ 
8:     end if
9:   end if
10: end for
```

αποδοτικά το κατώφλι του ή αν θα πρέπει να το ενημερώσει με την αντίστοιχη τιμή που τον πληροφορεί ο B . Έτσι το κατώφλι του COORD είναι πάντοτε μεγαλύτερο ή ίσο του αντίστοιχου κατωφλίου του αλγορίθμου DF. Η λίγο διαφορετική διαδικασία ανανέωσης που ακολουθεί ο COORD παρέχει μεγαλύτερη μείωση της κατανάλωσης ενέργειας σε κάθε κόμβο. Η συνολική κατανάλωση ενέργειας στο δίκτυο αγγίζει ποσοστά έως και 55% μικρότερα σε σχέση με τον αλγόριθμο DF.

Επομένως, με τον αλγόριθμο COORD έχουμε κόμβους που μαθαίνουν και εκμεταλλεύονται καλύτερα την διακινούμενη πληροφορία αξιολόγησης των κόμβων (τιμές Utility) ακόμη και για πακέτα που δεν τίθενται προς έλεγχο για προώθηση. Το κατώφλι το οποίο χρησιμοποιεί ο αλγόριθμος COORD ονομάζεται Coordinated threshold και συμβολίζεται ως $c\tau$. Η διαδικασία προώθησης πακέτων όπως περιγράφεται και στον Αλγόριθμο 3.3 έχει ως εξής: όταν ένας κόμβος v έρθει σε επαφή με ένα άλλο κόμβο v την χρονική στιγμή t τότε για κάθε πακέτο p το οποίο έχει στην κατοχή του ο κόμβος v θα πρέπει ο κόμβος v αρχικά να ελέγξει εάν το πακέτο p υπάρχει στον Buffer του. Εάν αυτό συμβαίνει τότε ο κόμβος v ελέγχει αν η τιμή του κατωφλίου $c\tau_{v_t}^p$ που γνωρίζει ο κόμβος v είναι μεγαλύτερη της δικής του τιμής $c\tau_{v_t}^p$. Σε μια τέτοια περίπτωση έχουμε ανανέωση του κατωφλίου $c\tau_{v_t}^p$ με την τιμή $c\tau_{v_t}^p$. Αντίθετα, στην περίπτωση που ο κόμβος v δεν έχει στην κατοχή του το πακέτο η διαδικασία που ακολουθεί είναι η σύγκριση του κατωφλίου $c\tau_{v_t}^p$ με την τιμή του Utility U_v . Όπως ακριβώς υλοποιεί και ο αλγόριθμος DF, αν η τιμή του $c\tau_{v_t}^p$ είναι μικρότερη της U_v τότε ανανεώνουμε το κατώφλι με την τιμή αυτή και προωθούμε το πακέτο στον κόμβο v .

Συνοψίζοντας Coordinated threshold $c\tau_{v_t}^p$ ενός κόμβου v την χρονική στιγμή t για ένα πακέτο p είναι η υψηλότερη τιμή του Utility μεταξύ των κόμβων οι οποίοι μεταφέρουν το συγκεκριμένο πακέτο p και τους οποίους γνωρίζει άμεσα ή/και έμμεσα ο κόμβος v . Είναι ξεκάθαρο ότι ο αλγόριθμος DF χρησιμοποιεί μόνο την πληροφορία για την μέγιστη τιμή του Utility την οποία μαθαίνει μόνο από τους κόμβους με τους οποίους έρχεται σε επαφή. Αντίθετα, ο αλγόριθμος COORD χρησιμοποιεί και πληροφορία την οποία ένας κόμβος μαθαίνει για κόμβους που δεν έχει συναντήσει και πιθανόν να μην συναντήσει στο μέλλον. Αυτό παρέχει σε έναν κόμβο περισσότερη γνώση του δικτύου και τον καθιστά άμεσα "αυστηρότερο" και ταυτόχρονα αποδοτικότερο κατά τις προωθήσεις αντιτύπων στο δίκτυο. Αυτό έχει ως αποτέλεσμα σημαντική μείωση του κόστους παράδοσης πακέτων (κατανάλωση ενέργειας και χώρου αποθήκευσης). Εξετάζοντας το αλγόριθμο ως προς την ικανότητα παράδοσης βλέπουμε και πάλι τα ποσοστά επιτυχούς παράδοσης να κυμαίνονται σε πολύ υψηλά επίπεδα.

Μια πληρέστερη περιγραφή της διαδικασίας προώθησης του αλγόριθμου και πως υλοποιείται η όλη διαδικασία παρουσιάζεται και στο Σχήμα 3.3. Περιγράφεται και εδώ το Anti-entropy Session και πως στην πράξη εκμεταλλεύεται ο αλγόριθμος COORD την διαδικασία της ανταλλαγής πληροφορίας για την ενημέρωση του κατωφλίου. Παρατηρούμε στο σχήμα μία διαδικασία βημάτων για την ολοκλήρωση της προώθησης πακέτων. Όταν ο κόμβος A έρχεται σε επαφή με έναν κόμβο B τότε ο κόμβος A αποστέλλει μια σύνοψη των πακέτων που έχει στην διάθεση του και μπορεί να αποστείλει στον B . Ο κόμβος B δέχεται την σύνοψη και αναζητά ποια πακέτα διαθέτει και ποια επιθυμεί να λάβει από τον κόμβο A . Ο κόμβος B υπολογίζει την τιμή του Utility για κάθε πακέτο που επιθυμεί και στέλνει μία αίτηση με τα πακέτα αυτά. Στην αίτηση επισυνάπτονται και οι αντίστοιχες τιμές του Utility για κάθε πακέτο. Επιπλέον, έχει επισυνάψει στην αίτηση και τα κατώφλια για τα πακέτα που ήδη γνωρίζει και δεν επιθυμεί να λάβει. Κατά συνέπεια, ο κόμβος A ελέγχει τις τιμές των κατωφλίων για κάθε πακέτο που γνωρίζει και ο κόμβος B . Εάν το κατώφλι του κόμβου B ξεπερνάει την τιμή του A τότε πραγματοποιείται ανανέωση της τιμής του κατωφλίου του κόμβου A . Εν συνεχεία, για κάθε πακέτο το οποίο ο B δεν γνωρίζει η διαδικασία είναι η ίδια όπως και στον DF. Ο κόμβος A υλοποιεί την βασική διαδικασία απόφασης για προώθηση των πακέτων που αιτείται ο κόμβος B . Πραγματοποιείται από τον κόμβο A έλεγχος της τιμής του Utility για κάθε πακέτο ξεχωριστά. Συγκρίνει ουσιαστικά την τιμή του Utility του B με το



Σχήμα 3.3: COORD: Διαδικασία προώθησης πακέτων κατά την επαφή ενός κόμβου A με έναν κόμβο B.

αντίστοιχο κατώφλι του A. Τελικά, προωθεί τα πακέτα για τα οποία ο κόμβος B έχει μεγαλύτερη τιμή και ανανεώνει και την τιμή του κατωφλίου του. Ο αλγόριθμος Coordinated Delegation Forwarding πετυχαίνει εκτενέστερες μειώσεις των αντιτύπων. Κατά συνέπεια, μειώνεται σε μεγάλο βαθμό η κατανάλωση ενέργειας αλλά και ο περιττός χώρος που θα δεσμευόταν με την χρήση του αλγορίθμου DF.

ΚΕΦΑΛΑΙΟ 4

ΔΡΟΜΟΛΟΓΗΣΗ ΜΕ ΑΡΧΕΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

4.1 Κίνητρο και Κεντρική Ιδέα

4.2 Υπόβαθρο Μηχανικής Μάθησης

4.3 Δρομολόγηση με Εφαρμογή και Ενημέρωση του K-means

4.1 Κίνητρο και Κεντρική Ιδέα

Οι μέχρι σήμερα προσεγγίσεις των αλγορίθμων δρομολόγησης με δυναμική δημιουργία αντιτύπων χρησιμοποιούν ως κριτήριο είτε μια απλή σύγκριση των Utilities των κόμβων που ήρθαν σε επαφή είτε βασίζονται στην σύγκριση του Utility με την μέγιστη τιμή (κατώφλι) την οποία έχει καταγράψει ο κόμβος που μεταφέρει ένα πακέτο. Τα δύο κριτήρια προώθησης διασαφηνίζουν μόνο αν ένας κόμβος είναι καταλληλότερος από κάποιον άλλο κόμβο του δικτύου. Έτσι και οι τρεις αλγόριθμοι CNR, DF και COORD "παγιδεύονται" στα πλαίσια μιας επαφής και εξετάζουν μόνο αν ένας κόμβος θεωρείται "καταλληλότερος" για προώθηση ή/και παράδοση. Κατά συνέπεια, η "συγκριτική λογική" και των τριών αλγορίθμων δεν εξετάζει το "πόσο καταλληλότερος" είναι ο κόμβος ο οποίος αιτείται ένα πακέτο. Σε μία επαφή μία μεγαλύτερη τιμή Utility ορίζει αυτομάτως έναν ικανότερο κόμβο αλλά όχι απαραίτητα έναν γενικά ικανό κόμβο για παράδοση/προώθηση. Για παράδειγμα, εάν σε μια επαφή δύο κόμβων A και B με αντίστοιχα Utilities $U_A = 0.1$ και $U_B = 0.11$,

ο κόμβος B είναι σίγουρα καταλληλότερος από τον κόμβο A . Το μείζον ερώτημα που προκύπτει όμως είναι πόσο πραγματικά υψηλή είναι η τιμή 0.11 γενικά στο δίκτυο. Σε αυτό το μειονέκτημα στοχεύουμε και προτείνουμε την δική μας μέθοδο επέκτασης των αλγορίθμων της κατηγορίας με δυναμική δημιουργία αντιτύπων.

Η "συγκριτική λογική" στοχεύει στην εύρεση του καταλληλότερου κόμβου με πολλαπλά βήματα για την προώθηση/παράδοση ενός πακέτου. Ένα βήμα θεωρείται η δημιουργία και η προώθηση ενός αντιτύπου του πακέτου. Έτσι λόγω του εύρους των τιμών του Utility και της τυχαιότητας των επαφών μπορεί να οδηγηθούμε σε δημιουργία πολλών αντιτύπων και περιττών προωθήσεων ακόμα και με την χρήση της έννοιας του κατωφλίου. Για να αποφευχθεί μία τέτοια κατάσταση είναι απαραίτητη μια γενική κατάταξη των κόμβων βάσει της οποίας μπορεί ο κάθε κόμβος κατανεμημένα να λαμβάνει πιο αποδοτικές αποφάσεις για την προώθηση των πακέτων που μεταφέρει. Την πληροφορία αυτή μπορούμε να την εξάγουμε από την ύπαρξη ομάδων κόμβων στο δίκτυο με διαφορετικές ικανότητες προώθηση/παράδοσης. Οι ομάδες αυτές στην πράξη σχηματίζονται βάσει των τιμών του Utility που υπάρχουν στο δίκτυο.

Η διαδικασία εξαγωγής της συγκεκριμένης πληροφορίας ονομάζεται ομαδοποίηση των κόμβων και βασίζεται στην εξής ιδέα: για κάθε πακέτο το οποίο υπάρχει στο δίκτυο υπάρχουν ομάδες κόμβων όπου άλλες αφορούν κόμβους οι οποίοι είναι καταλληλότεροι για προώθηση/παράδοση και άλλες οι οποίες αφορούν λιγότερο κατάλληλους κόμβους. Έτσι υπάρχουν για ένα πακέτο "ομάδες καταλληλότητας" για παράδοση ή/και προώθηση. Τα κριτήρια προώθησης μέχρι τώρα ήταν ελαστικά καθώς ένα πακέτο προωθούνταν με βάση την σύγκριση τιμών του Utility ή μιας τιμής Utility με την τιμή του κατωφλίου. Έτσι ένα πακέτο για να φτάσει σε έναν πραγματικά "κατάλληλο" κόμβο μπορεί να ακολουθήσει ένα μεγάλο μήκος μονοπάτι. Αυτό σημαίνει και δημιουργία μεγάλου αριθμού περιττών αντιτύπων του συγκεκριμένου πακέτου. Με την ομαδοποίηση των κόμβων ουσιαστικά θέτουμε ένα "αυστηρότερο" κριτήριο προώθησης. Αυτό επιτυγχάνεται με μετάβαση της "συγκριτικής λογικής" από συγκρίσεις τιμών σε συγκρίσεις των επιπέδων των ομάδων. Στην πράξη μία ομάδα ορίζει ένα διάστημα τιμών του Utility. Έτσι μπορούμε επίσης να πούμε ότι μεταβαίνουμε σε συγκρίσεις διαστημάτων των τιμών του Utility.

Κατά συνέπεια, θέτουμε ένα επιλεκτικότερο και αυστηρότερο κριτήριο προώθησης. Με το κριτήριο αυτό ένας κόμβος στοχεύει σε κόμβους πραγματικά κατάλληλους και μάλιστα ακολουθεί πιο "σύντομα μονοπάτια" δίχως να δημιουργεί

περιττά αντίτυπα στο δίκτυο. Μια τέτοια τεχνική ομαδοποίησης η οποία επεκτείνει τους απλούς αλγόριθμους της δυναμικής δημιουργίας αντιτύπων μπορεί σίγουρα να μειώσει και άλλο τις περιπτώσεις των περιττών προωθήσεων. Η απάντηση στην ερώτηση "Γιατί αυτή η επέκταση δεν θα σημαίνει ταυτόχρονη και σημαντική μείωση στα ποσοστά επιτυχούς παράδοσης;" αποτελούν οι ίδιοι οι αλγόριθμοι δρομολόγησης με δυναμική δημιουργία αντιτύπων. Οι τρεις εκδοχές αλγορίθμων στοχεύουν στο ίδιο ακριβώς αποτέλεσμα. Δηλαδή την μείωση του κόστους αλλά να διατηρηθούν σε υψηλά επίπεδα τα ποσοστά επιτυχούς παράδοσης. Έτσι εξελίσσοντας το κριτήριο προώθησης μέχρι να φτάσουμε στον αλγόριθμο COORD εξελίσσουμε και την γνώση του δικτύου που αντιλαμβάνεται ένας κόμβος. Η γνώση του δικτύου είναι και το κλειδί της μείωσης των αντιτύπων αλλά ταυτόχρονα αποτελεί και τον κύριο λόγο διατήρησης υψηλών ποσοστών παράδοσης. Κατά συνέπεια, με την ομαδοποίηση των κόμβων στοχεύουμε και εμείς σε ένα ακόμα ισχυρότερο κριτήριο προώθησης και σε μια καλύτερη εικόνα του δικτύου που αντιλαμβάνεται ένας κόμβος. Η μέθοδος μας δίνει έτσι μια "αντικειμενική" εικόνα σε ένα κόμβο ο οποίος μπορεί να αποφασίζει με έναν πιο αποτελεσματικό τρόπο. Με την λογική σύγκρισης των επιπέδων των ομάδων (διαστημάτων) "κόβουμε" επιπλέον περιπτώσεις προωθήσεων που ουσιαστικά δεν βοηθούν πραγματικά στην παράδοση πακέτων αλλά καταναλώνουν πόρους του δικτύου.

Για την επιβεβαίωση της ιδέας μας εκτελέσαμε διαφορετικά πειράματα και προσομοιώσεις με διαφορετικούς αλγόριθμους και Utilities. Σε κάθε περίπτωση καταγράψαμε κατανεμημένα σε κάθε κόμβο τις τιμές του Utility. Επίσης, καταγράψαμε και την εικόνα των πραγματικών ομάδων που υπάρχουν στο δίκτυο για λόγους σύγκρισης με την κατανεμημένη εικόνα κάθε κόμβου. Μάλιστα οι καταγραφές έγιναν και σε διαφορετικές στιγμές των προσομοιώσεων. Παρατηρήσαμε ότι όντως δημιουργούνται ομάδες τιμών του Utility (συνεπώς και κόμβων) τις οποίες βλέπει ο κάθε κόμβος του δικτύου. Είδαμε επίσης ότι υπάρχουν κόμβοι οι οποίοι μπορεί να μην συναντήσουν κόμβους οι οποίοι ανήκουν σε υψηλού επιπέδου ομάδες. Το πρόβλημα αυτό αντιμετωπίζεται με την κινητικότητα των κόμβων και της τεχνικής Store, Carry & Forward. Επιπλέον, ακόμα και στην περίπτωση που ένας κόμβος βλέπει μόνο μία ομάδα κόμβων της γενικής κατάταξης έχουμε σχηματισμό υποομάδων στο εσωτερικό αυτής. Αυτό καθιστά δυνατή την προώθηση ενός πακέτου ακόμα και στην ίδια ομάδα κόμβων της γενικής εικόνας του δικτύου. Για να αυξήσουμε τα ποσοστά επιτυχούς παράδοσης της μεθόδου μας αξίζει να χαλαρώσουμε λίγο το

κριτήριο σύγκρισης των ομάδων που αντιλαμβάνεται κατανεμημένα ένας κόμβος. Αυτό το πετυχαίνουμε με το να αφήσουμε τα πακέτα τα οποία μόλις δημιουργήθηκαν να διακινούνται ακόμα και μέσα στην ίδια ομάδα. Ο βασικός λόγος είναι ότι δεν έχει νόημα να έχουμε ένα τόσο αυστηρό κριτήριο για τα πακέτα αυτά. Ως κριτήριο προώθησης για τα συγκεκριμένα πακέτα ορίζουμε το αντίστοιχο κριτήριο του εκάστοτε αλγορίθμου δρομολόγησης.

4.2 Υπόβαθρο Μηχανικής Μάθησης

Η μηχανική μάθηση αποτελεί αναπόσπαστο κομμάτι της επιστήμης της πληροφορικής. Η μηχανική μάθηση επικεντρώνεται στην μελέτη και την κατασκευή αλγορίθμων που μπορούν να μαθαίνουν από δεδομένα και να κάνουν προβλέψεις με βάση αυτά. Αφορά επίσης μεθόδους αναπαράστασης και αλγόριθμους για την εξαγωγή γνώσης από δεδομένα. Οι εφαρμογές της καλύπτουν ένα ευρύ φάσμα αναγκών, κυρίως στην ανάλυση δεδομένων και στα πλαίσια δημιουργίας μοντέλων για την λήψη αποφάσεων. Σίγουρα η έννοια της μηχανικής μάθησης είναι στενά συνδεδεμένη και συχνά συγχέεται με την υπολογιστική στατιστική καθώς και αυτός ο κλάδος στοχεύει στην πρόβλεψη. Επιπρόσθετα, συνδέεται σημαντικά με την τεχνητή νοημοσύνη και με την μαθηματική βελτιστοποίηση. Επιπλέον, εφαρμόζεται σε μια σειρά από υπολογιστικές εργασίες, όπου τόσο ο σχεδιασμός όσο και ο ρητός προγραμματισμός των αλγορίθμων είναι ανέφικτος. Η μηχανική μάθηση συνδέεται και με την εξόρυξη δεδομένων, όπου η τελευταία στοχεύει στην ανάλυση κυρίως των δεδομένων. Τα μοντέλα που προκύπτουν μέσω της μηχανικής μάθησης παρέχουν αποδοτικές αποφάσεις και αποτελέσματα μέσω συσχετίσεων και μοτίβων των δεδομένων.

Οι τύποι προβλημάτων και εργασιών της μηχανικής μάθησης ταξινομούνται σε τρεις κύριες κατηγορίες:

- Μάθηση με επίβλεψη (αλλιώς επιβλεπόμενη μάθηση ή επιτηρούμενη μάθηση) (Supervised Learning): Το σύστημα δέχεται τις εισόδους καθώς και τα επιθυμητά αποτελέσματα (δεδομένα). Ο στόχος του είναι να εκπαιδευτεί (να μάθει έναν γενικό κανόνα) έτσι ώστε να αντιστοιχίζει τις εισόδους με τα αποτελέσματα (εξόδους).
- Μάθηση χωρίς επίβλεψη (αλλιώς μη-επιβλεπόμενη μάθηση ή μη επιτηρούμενη

μάθηση (Unsupervised Learning): Ο ίδιος ο αλγόριθμος παράγει τον γενικό κανόνα αφού δεν του παρέχετε κάποια είσοδος ως γνώση και πρέπει να βρει μόνος του την δομή των δεδομένων εισόδου.

- Ενισχυτική μάθηση: Ένα σύστημα αλληλεπιδρά με έναν δυναμικό περιβάλλον στο οποίο πρέπει να πραγματοποιηθεί ένα έργο. Στο σύστημα όμως δεν αποσαφηνίζεται από κάποιον αν και πότε έχει φτάσει κοντά στο στόχο του.

Στην μεταπτυχιακή εργασία ερευνάται η δεύτερη κατηγορία στην οποία ανήκει και η ομαδοποίηση ή συσταδοποίηση (clustering) και συνεπώς και ο αλγόριθμος K-means ο οποίος χρησιμοποιήθηκε.

Ομαδοποίηση είναι η διαδικασία εκείνη κατά την οποία ένα σύνολο από δεδομένα διαχωρίζονται σε λογικές ομάδες. Η κατάταξη αντικειμένων στην ίδια ομάδα ερμηνεύεται ως ομοιότητα των δεδομένων και αντίστροφα (ανόμοια δεδομένα ανήκουν σε διαφορετικές ομάδες). Η ομοιότητα ή η ανομοιότητα των δεδομένων εξαρτάται από την μορφή τους αλλά και το πρόβλημα που προσπαθεί η ομαδοποίηση να επιλύσει. Η ομαδοποίηση όπως ήδη έχουμε αναφέρει ανήκει στην μάθηση χωρίς επίβλεψη. Τα δεδομένα συχνά αναφέρονται στην βιβλιογραφία ως πρότυπα ή διανύσματα. Συνοφίζοντας, ομαδοποίηση ονομάζεται η διαδικασία που διαχωρίζει πρότυπα σε ομάδες (συστάδες-clusters), όπου εφαρμόζεται κάποιο κριτήριο ομοιότητας για τον διαχωρισμό αυτό. Η ομαδοποίηση στοχεύει στην πράξη στον προσδιορισμό ομάδων με βάση την ομοιότητα των δεδομένων. Ένας πιο σαφής και μαθηματικός ορισμός αποτελεί ο παρακάτω.

Ορισμός 4.1. (Ομαδοποίηση-Clustering). Έστω ένα σύνολο διανυσμάτων $X = x_1, x_2, \dots, x_n$ και επιθυμούμε διαχωρισμό αυτών σε m ομάδες C_1, C_2, \dots, C_m , με $m \ll n$ έτσι ώστε: $C_i \neq \emptyset, \forall i = 1, 2, 3, \dots, m$. Η διαδικασία της διαμέρισης ονομάζεται ομαδοποίηση των διανυσμάτων (δεδομένων).

Η ταξινόμηση δεδομένων, η οποία ανήκει στην μάθηση με επίβλεψη συχνά συγχέεται με την ομαδοποίηση. Η διαφορά της ομαδοποίησης δεδομένων (data clustering) από την ταξινόμηση δεδομένων (data classification) είναι ότι στην ταξινόμηση οι ομάδες στις οποίες θα τοποθετηθούν τα δεδομένα είναι προκαθορισμένες. Αυτό σημαίνει ότι είναι εκ των προτέρων γνωστός ο αριθμός των ομάδων και οι ετικέτες (labels) τους. Η ταξινόμηση είναι και αυτό ένα σύστημα μάθησης καθώς οι ετικέτες που δίνονται από τα δεδομένα ενός dataset χρησιμοποιούνται ώστε να μάθει το μοντέλο ταξινόμησης την δομή κάθε κλάσης έτσι ώστε να είναι σε θέση να ταξινομήσει

σωστά αργότερα ένα νέο πρότυπο. Αντίθετα, στην ομαδοποίηση δεδομένων οι ομάδες δεν προϋπάρχουν αλλά αποφασίζονται από τον ίδιο τον αλγόριθμο δυναμικά. Δηλαδή υπάρχει ένα σύνολο δεδομένων το οποίο πρέπει να διαχωριστεί ώστε να προκύψουν δυναμικά οι ομάδες. Η διαμέριση των δεδομένων σε ομάδες οδηγείται από τα ίδια τα δεδομένα (data driven). Σκοπός είναι να δημιουργηθούν ομάδες που η καθεμία από αυτές θα συγκεντρώνει ομοιογενή στοιχεία. Κάθε μία από αυτές τις ομάδες διατηρεί ένα κέντρο, συνήθως το πιο κεντρικό και αντιπροσωπευτικό στοιχείο της.

Οι αλγόριθμοι και οι τεχνικές ομαδοποίησης μπορεί να είναι είτε ιεραρχικοί (Hierarchical) είτε μη ιεραρχικοί (non Hierarchical ή Partitional). Οι ιεραρχικοί αλγόριθμοι βρίσκουν διαδοχικές ομάδες, χρησιμοποιώντας κάθε φορά ήδη καθιερωμένες ομάδες και οργανώνουν την ιεραρχία αυτή σε μία δεντρική δομή (δενδρόγραμμα). Από την άλλη, οι μη ιεραρχικοί αλγόριθμοι καθορίζουν τις ομάδες άμεσα σε μη επικαλυπτόμενα υποσύνολα-ομάδες (clusters) του συνόλου των παρατηρήσεων (δεδομένων).

Ο αλγόριθμος K-means ανήκει στην τελευταία κατηγορία. Τα βασικά βήματα του αλγόριθμου είναι τα εξής:

- Επιλογή του αριθμού των ομάδων.
- Τυχαία επιλογή k κέντρων.
- Μεταβίβαση του κάθε σημείου στην κέντρο της κοντινότερης ομάδας.
- Υπολογισμός των νέων κέντρων των ομάδων.
- Επανάληψη μέχρι να συγκλίνει ο αλγόριθμος σε κάποιο κριτήριο.

Ο αλγόριθμος K-means είναι ένας από τους πιο απλούς και δημοφιλέστερους αλγορίθμους ομαδοποίησης που ανήκουν στην ευρύτερη κατηγορία των τεχνικών μάθησης χωρίς επίβλεψη. Ο αλγόριθμος αυτός είναι δημοφιλής εξαιτίας της απλότητας της υλοποίησης του και της γραμμικής του πολυπλοκότητας ($O(n)$, όπου n το σύνολο των διανυσμάτων). Η διαδικασία της ομαδοποίησης ενός συνόλου δεδομένων με βάση τον K-means είναι εύκολη, αρκεί να είναι εκ των προτέρων καθορισμένος ο αριθμός (k) των clusters (ομάδων) που θα προκύψουν. Η κύρια ιδέα είναι να προσδιοριστούν αρχικά k κέντρα, ένα για κάθε cluster. Αυτά τα αρχικά κέντρα πρέπει να επιλεγούν με σωστό τρόπο, γιατί διαφορετικές αρχικές θέσεις για τα κέντρα δίνουν

διαφορετικά τελικά αποτελέσματα. Δηλαδή η αρχική θέση των κέντρων επηρεάζει το αποτέλεσμα που θα δώσει ο αλγόριθμος. Έτσι μία αποτελεσματική τεχνική είναι οι πολλαπλές επαναλήψεις της διαδικασίας αρχικοποίησης και η επιλογή τελικά της αρχικοποίησης και κατά συνέπεια των κέντρων με την μικρότερη τιμή της συνάρτησης τετραγωνικού σφάλματος. Την τεχνική αυτή επιλέγουμε και χρησιμοποιούμε και εμείς στην υλοποίηση του αλγορίθμου K-means έτσι ώστε να μην εξαρτάται από την αρχικοποίηση. Πιο απλά επιλέγουμε την καλύτερη δυνατή αρχικοποίηση σε ένα πλήθος επαναλήψεων.

Το επόμενο βήμα είναι επιλογή κάθε στοιχείου από το σύνολο δεδομένων και η συσχέτιση του με το κοντινότερο σε αυτό κέντρο. Όταν αυτό πραγματοποιηθεί για όλα τα στοιχεία του συνόλου δεδομένων, το πρώτο βήμα έχει ολοκληρωθεί και μία πρώτη ομαδοποίηση έχει ήδη προκύψει. Στη συνέχεια απαιτείται να υπολογιστούν ξανά k νέα κέντρα για κάθε ένα cluster που προέκυψε από το προηγούμενο βήμα. Αφού οριστούν τα νέα k κέντρα, ακολουθεί και πάλι η ίδια διαδικασία ανάθεσης κάθε στοιχείου του συνόλου δεδομένων στο κοντινότερο νέο κέντρο. Επομένως, προκύπτει μια επανάληψη της ίδιας ακριβώς διαδικασίας. Αποτέλεσμα των συνεχών επαναλήψεων είναι σε κάθε βήμα τα κέντρα να αλλάζουν θέση και τα στοιχεία να ανατίθενται κάθε φορά στην κατάλληλη ομάδα (cluster) με βάση τις αποστάσεις τους από τα νέα αυτά κέντρα. Το κριτήριο τερματισμού του αλγορίθμου αποτελεί η κατάσταση στην οποία τα κέντρα των ομάδων σταματούν να μεταβάλλονται σε δύο συνεχόμενες επαναλήψεις. Το αποτέλεσμα που προκύπτει είναι η ομαδοποίηση του συνόλου δεδομένων σε k clusters. Ο αλγόριθμος στην πράξη στοχεύει να ελαχιστοποιήσει την αντικειμενική συνάρτηση (objective function):

$$E = \sum_{i=1}^N \min \{d(x_i, m_j)\}, \quad N \in \mathbb{N}. \quad (4.1)$$

Ένα σημαντικό σημείο το οποίο αξίζει περαιτέρω ανάλυση αποτελεί η επιλογή των ομάδων k το οποίο και ο αλγόριθμος K-means θεωρεί δεδομένο. Μια τεχνική η οποία χρησιμοποιείται κατά κόρον για την επιλογή του καλύτερου και αποδοτικότερου αριθμού ομάδων αποτελεί το κριτήριο σκιαγράφησης (Silhouette Criterion). Το κριτήριο σκιαγράφησης αποτελεί μία μέθοδο ερμηνείας και επικύρωσης της συνοχής των ομάδων δεδομένων που έχουν προκύψει σύμφωνα με έναν αλγόριθμο. Η τεχνική αυτή παρέχει μία ποιοτική εικόνα για το πόσο καλά το κάθε πρότυπο έχει αντιστοιχιστεί στην κατάλληλη ομάδα δεδομένων. Το Silhouette Criterion περιγράφηκε πρώτη φορά το 1986 από τον Peter J. Rousseeuw. Στην πράξη υπολογίζει πόσο όμοιο

είναι ένα πρότυπο (δεδομένο) με τα υπόλοιπα πρότυπα στην ομάδα στην οποία έχει καταταχθεί (συνεκτικότητα-cohesion) σε σύγκριση με την ομοιότητα του με τα πρότυπα των υπόλοιπων ομάδων δεδομένων (διαχωρισμός-separation). Το κριτήριο σκιαγράφησης κυμαίνεται από -1 έως και 1 , όπου μία υψηλή τιμή του δείχνει ότι τα αντικείμενα έχουν καταταχθεί κατάλληλα στις ομάδες τους και ταιριάζουν λιγότερο στις υπόλοιπες ομάδες. Αν έχουμε πολλές περιπτώσεις όπου παρουσιάζουν χαμηλές τιμές τότε αυτό σημαίνει αυτομάτως μία κακή τοποθέτηση των προτύπων, η οποία οφείλεται είτε σε επιλογή μεγάλου είτε μικρού αριθμού ομάδων. Το κριτήριο σκιαγράφησης μπορεί να υπολογιστεί χρησιμοποιώντας οποιαδήποτε μετρική απόστασης. Εμείς χρησιμοποιούμε την Ευκλείδεια απόσταση (Euclidean distance) για τον υπολογισμό του κριτηρίου. Ένας ποιο σαφής μαθηματικός ορισμός δίνεται παρακάτω.

Ορισμός 4.2. (Κριτήριο Σκιαγράφησης-Silhouette Criterion). Υποθέτουμε ότι τα δεδομένα μας έχουν ομαδοποιηθεί σε k ομάδες (clusters) με κάποια μέθοδο ομαδοποίησης (όπως είναι ο K-means). Για κάθε στοιχείο i , ονομάζουμε $a(i)$ την μέση τιμή της ανομοιότητας (dissimilarity) του στοιχείου αυτού από τα υπόλοιπα στοιχεία στην ομάδα. Το $a(i)$ το ερμηνεύουμε αλλιώς ως το πόσο καλά κάθε στοιχείο ταιριάζει στην ομάδα του. Όσο μικρότερη είναι τιμή τόσο καλύτερη είναι η ανάθεση του στην ομάδα. Την ανομοιότητα ενός σημείου i από μία ομάδα (cluster), την ορίζουμε ως την μέση τιμή της απόστασης από όλα τα άλλα σημεία τα οποία ανήκουν στην ομάδα αυτή. Ορίζουμε επίσης ως $b(i)$ την μικρότερη ανομοιότητα που εμφανίζει το στοιχείο i από οποιαδήποτε άλλη ομάδα στην οποία δεν ανήκει. Η ομάδα αυτή χαρακτηρίζεται ως γειτονική του σημείου i επειδή αποτελεί την ομάδα στην οποία ταιριάζει καλύτερα το σημείο μετά την δική του ομάδα. Ορίζουμε λοιπόν το κριτήριο σκιαγράφησης ως εξής:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}. \quad (4.2)$$

Κατά συνέπεια, από τον παραπάνω ορισμό ισχύει ότι το $s(i) \in [-1, 1]$. Για μια τιμή του $s(i)$ η οποία βρίσκεται κοντά στο 1 σημαίνει ότι έχουμε $a(i)$ πολύ μικρότερο του $b(i)$. Μια χαμηλή τιμή $a(i)$ μεταφράζεται ως σωστή κατάταξη του στοιχείου i στην ομάδα του, ενώ μεγάλη τιμή $b(i)$ ερμηνεύεται ως χαμηλή ομοιότητα του σημείου αυτού με την γειτονική ομάδα. Κατά συνέπεια μία τιμή του $s(i)$ κοντά στο 1 σημαίνει ότι το στοιχείο i έχει τοποθετηθεί κατάλληλα, ενώ μία τιμή κοντά στο -1 σημαίνει ακριβώς το αντίθετο. Αν υπολογίσουμε την μέση τιμή όλων των σημείων

i τότε μπορούμε να υπολογίσουμε πρακτικά το πόσο καλά έχουν καταταχθεί όλα τα δεδομένα μας στις ομάδες. Αν έχουμε μια χαμηλή μέση τιμή όλων των $s(i)$ σημαίνει ότι ο αριθμός k των ομάδων δεν είναι κατάλληλος. Στην διπλωματική αυτή χρησιμοποιήσαμε την μέση τιμή του Silhouette Criterion για όλα τα δεδομένα για να αξιολογήσουμε την καταλληλότητα των ομαδοποιήσεων που μας δίνει ο K-means σε κάθε περίπτωση.

4.3 Δρομολόγηση με Εφαρμογή και Ενημέρωση του K-means

Η ιδέα που αναλύθηκε στην Ενότητα 4.1 παρουσιάζεται εκτενέστερα παρακάτω με την τελική μορφή χρήσης της κατά την διαδικασία της δρομολόγησης. Η κεντρική ιδέα είναι να χωρίσουμε τη διαδικασία της δρομολόγησης σε δύο φάσεις. Η πρώτη φάση αφορά μια περίοδο μάθησης (training period), η οποία είναι απαραίτητη για την καταγραφή των δεδομένων που χρειάζεται ο αλγόριθμος K-means έτσι ώστε να προκύψει μια ομαδοποίηση. Πρέπει να τονιστεί εδώ ότι κατά την διάρκεια της φάσης αυτής ο αλγόριθμος δρομολόγησης χρησιμοποιεί κάποια από τις γνωστές τεχνικές για την απόφαση δημιουργίας αντιτύπου. Με άλλα λόγια στην περίοδο μάθησης το κριτήριο προώθησης ενός αντιτύπου παραμένει το ίδιο με τους απλούς αλγόριθμο δρομολόγησης (CNR, DF, COORD). Σε δεύτερη φάση (after training period) το κριτήριο προώθησης αλλάζει πλέον. Η έναρξη της δεύτερης φάσης πραγματοποιείται αφού ο K-means έχει καταγράψει τον επιθυμητό αριθμό από δεδομένα N και μας έχει δώσει τις σχηματιζόμενες ομάδες. Η δημιουργία και η προώθηση ενός αντιτύπου επιτυγχάνεται πλέον μόνο στην περίπτωση που ο κόμβος που αιτείται ένα πακέτο ανήκει σε μία υψηλότερου επιπέδου "ομάδα καταλληλότητας". Στην πράξη έχουμε προώθηση ενός αντιτύπου εάν η τιμή του Utility του κόμβου αυτού ανήκει σε μια ομάδα με υψηλότερες τιμές.

Η μείωση κατανάλωσης ενέργειας σε κάθε περίπτωση αλγορίθμου δρομολόγησης με δυναμική δημιουργία αντιτύπων, όπου ενσωματώθηκε η μέθοδος μας, είναι σημαντική και κυμαίνεται σε ποσοστά από 20% (παρουσιάζεται συνήθως στον COORD) μέχρι και σε πολλές περιπτώσεις και 60%. Αξίζει να αναφέρουμε ότι η μέθοδος αυτή μπορεί να εφαρμοστεί και σε αλγορίθμους που δεν ανήκουν στην κατηγορία με δυναμική δημιουργία αντιτύπων αλλά χρησιμοποιούν ένα Utility.

4.3.1 Δρομολόγηση με K-means

Στην ενότητα αυτή θα εξετάσουμε και θα αναλύσουμε την εφαρμογή της μεθόδου μας σε έναν αλγόριθμο δρομολόγησης που χρησιμοποιεί ένα Utility για να αξιολογεί την ικανότητα του κάθε κόμβου για προώθηση/παράδοση ενός πακέτου. Όσα αναφέραμε περιληπτικά παραπάνω περιγράφονται με περισσότερες λεπτομέρειες και στον Αλγόριθμο 4.1. Αρχικά ας θεωρήσουμε ότι έχουμε N κόμβους στο δίκτυο μας. Γνωρίζουμε ακόμα το είδος του Utility, δηλαδή αν χρησιμοποιείται κάποιο Utility ανεξάρτητο του προορισμού (Destination Independent-DI) είτε κάποιο το οποίο εξαρτάται από τον κόμβο προορισμό (Destination Dependent-DD). Επίσης, έχουμε μια δομή η οποία αποτελείται από αντικείμενα τύπου K-means (ένα ή πολλά ανάλογα το είδος του Utility) όπου καταγράφεται κάθε τιμή του Utility που μαθαίνει ένας κόμβος N_i κατά τις επαφές του.

Σε μία επαφή ενός κόμβου N_i με έναν κόμβο N_j , για κάθε πακέτο p το οποίο έχει στην κατοχή του ο N_i αρχικά ελέγχουμε σε ποια φάση της διαδικασίας δρομολόγησης βρισκόμαστε. Ο έλεγχος αυτός είναι απαραίτητος όχι μόνο για να οριστεί κατάλληλα το κριτήριο απόφασης για την προώθηση ενός αντιτύπου αλλά και για να υλοποιηθεί μια σειρά άλλων εργασιών. Εάν βρισκόμαστε στην φάση εκμάθησης ($TrainingPeriod = true$) τότε προωθούμε με βάση το πρωτόκολλο δρομολόγησης το οποίο χρησιμοποιείται. Πρέπει επίσης να καταγράψουμε την τιμή του Utility του κόμβου με τον οποίο πραγματοποιήθηκε η επαφή. Εάν χρησιμοποιείται ένα Utility το οποίο είναι DD τότε πρέπει να αναζητήσουμε το αντικείμενο K-means το οποίο αφορά τον κόμβο προορισμού d ($Km_{structure,d}$) του πακέτου και να καταγράψουμε εκεί την τιμή του Utility. Αντίθετα, όταν γίνεται χρήση ενός DI Utility έχουμε όλες τις καταγραφές σε ένα αντικείμενο ($Km_{structure,0}$) που αφορά γενικά όλο το δίκτυο. Και στις δύο περιπτώσεις αν στο αντίστοιχο αντικείμενο K-means είναι γεμάτος ο χώρος αποθήκευσης θα πρέπει να εκτελεστεί ο αλγόριθμος K-means έτσι ώστε να ομαδοποιηθούν τα δεδομένα. Με την εκτέλεση του K-means έχουμε και την άμεση λήξη της περιόδου μάθησης.

Το κριτήριο δημιουργίας και προώθησης αντιτύπων αλλάζει πλέον και σχετίζεται με τις ομάδες κόμβων που έχουν οριστεί από τον αλγόριθμο K-means. Έτσι για κάθε πακέτο ελέγχουμε σε ποια ομάδα ανήκουν οι αντίστοιχες τιμές του Utility $U_{j,p}$ του κόμβου N_j και $U_{i,p}$ του κόμβου N_i . Προωθούμε μόνο στην περίπτωση που η ομάδα του κόμβου N_j είναι υψηλότερου επιπέδου από ότι του κόμβου N_i .

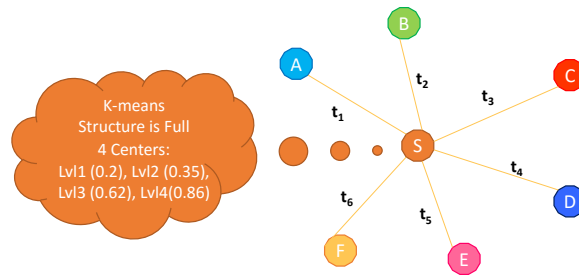
Αλγόριθμος 4.1 Δρομολόγηση με Αρχές Μηχανικής Μάθησης (K-means)

```
1: Let  $N_1, N_2, \dots, N_N$  be nodes
2: Let utilityType be the type of Utility
3: Let  $Km_{structure}$  be a structure of K-means objects holding useful network information
4: Node  $N_i$  has utility  $U_{i,p}$  for packet  $p$ .
5: On contact between  $N_i$  and node  $N_j$  :
6: for every packet  $p$  exists in  $Buf_{N_i}$  do
7:   if TrainingPeriod = true then
8:     if utilityType = DD then
9:       save  $U_{j,p}$  in  $Km_{structure,d}$  where  $d$  is the destination;
10:      if  $Km_{structure,d}$  is full then
11:        run K-means;
12:        TrainingPeriod  $\leftarrow$  false;
13:      end if
14:    else if utilityType = DI then
15:      save  $U_{j,p}$  in  $Km_{structure,0}$  for the whole network;
16:      if  $Km_{structure,0}$  is full then
17:        run K-means;
18:        TrainingPeriod  $\leftarrow$  false;
19:      end if
20:    end if
21:    node  $N_i$  reacts as an existing routing protocol commands (Forward or Not);
22:  else
23:    if ( $U_{j,p}Cluster > U_{i,p}Cluster$ ) OR ( $U_{j,p}Cluster = U_{i,p}Cluster$  but  $p$  has been
    just created and  $U_{j,p} > U_{i,p}$ ) then
24:      forward packet  $p$ ;
25:    end if
26:    LVQ: Online update of clusters's centers;
27:  end if
28: end for
```

Επειδή το κριτήριο αυτό είναι εν γένει αυστηρό το χαλαρώνουμε επιτρέποντας τις προωθήσεις στην ίδια ομάδα αν τηρείτε το απλό κριτήριο προώθησης της προηγούμενης φάσης ($U_{j,p} > U_{i,p}$, $U_{j,p} > \tau_{i,p}$, $U_{j,p} > c\tau_{i,p}$). Ο λόγος είναι ότι αυτό βελτιώνει σε μεγάλο βαθμό τα ποσοστά παράδοσης με ένα μικρό κόστος προωθήσεων εσωτερικά των ομάδων. Άλλωστε δεν έχει νόημα να "κόβουμε" προωθήσεις πακέτων τα οποία μόλις δημιουργήθηκαν. Με τον τρόπο αυτό αποφεύγουμε πιθανούς "εγκλωβισμούς" τέτοιων πακέτων σε μία ομάδα. Αυτό θα μπορούσε να συμβεί εάν ο κόμβος που δημιούργησε το πακέτο δεν έρθει ποτέ σε επαφή με κάποιο κόμβο ανώτερης "ομάδας καταλληλότητας". Το αποτέλεσμα θα ήταν μια σαφή πτώση του ποσοστού επιτυχούς παράδοσης. Να υπογραμμίσουμε ότι για την προώθηση πακέτων που βρίσκονται στο πρώτο άλμα (μόλις δημιουργήθηκαν) το κριτήριο προώθησης είναι ανάλογο του αλγορίθμου δρομολόγησης που χρησιμοποιείται. Στον Αλγόριθμο 4.1 χρησιμοποιείται το κριτήριο προώθησης του αλγορίθμου CNR. Εάν χρησιμοποιούμε ένα εκ των αλγορίθμων δρομολόγησης DF ή COORD συγκρίσεις με το κατώφλι είναι απαραίτητες στην φάση αυτή.

Ένα παράδειγμα δίνεται στο Σχήμα 4.1, το οποίο παρουσιάζει γιατί η μέθοδος που προτείνουμε λειτουργεί αποτελεσματικότερα στην απόφαση για δημιουργία και προώθηση αντιτύπων σε ένα ομορτυνιστικό δίκτυο. Βλέπουμε τις επαφές του κόμβου S κατά το πέρασμα του χρόνου και πόσο αποτελεσματικά τις εκμεταλλεύεται σε κάθε περίπτωση αλγορίθμου. Να τονίσουμε εδώ ότι για λόγους απλότητας και κατανόησης θεωρούμε ότι τα πακέτα που ελέγχονται για προώθηση δεν βρίσκονται στο πρώτο άλμα. Την χρονική στιγμή t_1 ο κόμβος S συναντάει τον κόμβο A , όπου σε περίπτωση χρήσης του αλγορίθμου CNR θα είχαμε αποστολή αντιτύπου του πακέτου p όπως και με την χρήση του CNR-Kmeans. Την χρονική t_2 στιγμή βλέπουμε οι δύο αλγόριθμοι να διαφοροποιούνται, αφού με σύγκριση απλά των τιμών του Utility ο CNR προωθεί το πακέτο ενώ ο CNR-Kmeans δεν το πράττει καθώς οι δύο κόμβοι ανήκουν στην ίδια ομάδα. Στις υπόλοιπες χρονικές στιγμές από t_3 έως t_6 οι δύο αλγόριθμοι προχωρούν στις ίδιες προωθήσεις.

Στους αλγορίθμους οι οποίοι χρησιμοποιούν την έννοια του κατωφλίου, θεωρούμε κάποιες τιμές ως κατώφλια οι οποίες έχουν προκύψει από προγενέστερες επαφές του κόμβου S . Κατά συνέπεια, για τον αλγόριθμο DF (όπως και DF-Kmeans) η τιμή του κατωφλίου, η οποία έχει καταγραφεί είναι 0.72, ενώ για την περίπτωση του αλγορίθμου COORD (όπως και COORD-Kmeans) η τιμή του αντίστοιχου κατωφλίου είναι 0.9. Την διαφορά των δύο αλγορίθμων την έχουμε αναλύσει



time	v	u	U_v	U_u	τ_v^p	τ_u^p	$\sigma_{\tau_v^p}$	$\sigma_{\tau_u^p}$	CNR	CNR Kmeans	DF	DF Kmeans	COORD	COORD Kmeans
t_1	S	A	0.4	0.5	0.72	0.5	0.9	0.5	✓	✓ (Lvl2 < Lvl3)	✗	✗ (Lvl4 > Lvl3)	✗	✗ (Lvl4 > Lvl3)
t_2	S	B	0.4	0.42	0.72	0.42	0.9	0.42	✓	✗ (same Lvl)	✗	✗ (Lvl4 > Lvl2)	✗	✗ (Lvl4 > Lvl2)
t_3	S	C	0.4	0.8	0.72	0.8	0.9	0.8	✓	✓ (Lvl2 < Lvl4)	✓	✓ (Lvl3 < Lvl4)	✗	✗ (same Lvl)
t_4	S	D	0.4	0.84	0.8	0.84	0.9	0.84	✓	✓ (Lvl2 < Lvl4)	✓	✓ (same Lvl)	✗	✗ (same Lvl)
t_5	S	E	0.4	0.89	0.84	0.89	0.9	0.89	✓	✓ (Lvl2 < Lvl4)	✓	✗ (same Lvl)	✗	✗ (same Lvl)
t_6	S	F	0.4	0.92	0.89	0.92	0.9	0.92	✓	✓ (Lvl2 < Lvl4)	✓	✗ (same Lvl)	✓	✗ (same Lvl)

Σχήμα 4.1: Δρομολόγηση με Kmeans: Σύγκριση διαδικασίας προώθησης πακέτων με χρήση της πληροφορία ομαδοποίησης.

διεξοδικά στο Κεφάλαιο 3. Συνεπώς, στον αλγόριθμο COORD το κατώφλι είναι υψηλότερο για να τονιστεί ότι μπορούμε να έχουμε ενημέρωση της τιμής του κατωφλίου και από πληροφορία η οποία προκύπτει έμμεσα από επαφές των γειτόνων του κόμβου S και όχι μόνο άμεσα από επαφές του ίδιου του κόμβου.

Παρατηρούμε ότι οι δύο αλγόριθμοι DF και DF-Kmeans διαφοροποιούνται την χρονική στιγμή t_4 , αφού την χρονική στιγμή t_3 ο κόμβος S ενημερώνει το κατώφλι του και μεταβαίνει σε ομάδα υψηλότερου επιπέδου. Την χρονική στιγμή t_4 ο κόμβος S έρχεται σε επαφή με τον κόμβο D και ενώ ο αλγόριθμος DF προωθεί ένα πακέτο, ο DF-Kmeans δεν προωθεί αντίτυπο στον κόμβο D . Ο λόγος είναι ξεκάθαρος αφού οι τιμές των κατωφλίων ανήκουν στην ίδια ομάδα (cluster). Λόγω της παραπάνω αλλαγής ομάδας δεν πραγματοποιούνται προωθήσεις στους κόμβους E και F στον αλγόριθμο DF-Kmeans τις αντίστοιχες χρονικές στιγμές t_5 και t_6 . Αντίθετα, λόγω των τιμών του κατωφλίου ο απλός αλγόριθμος DF δημιουργεί και προωθεί αντίτυπα. Τέλος στην περίπτωση των αλγορίθμων COORD και COORD-Kmeans βλέπουμε διαφοροποίηση μόνο στην χρονική στιγμή t_6 . Ο αλγόριθμος COORD δημιουργεί και προωθεί ένα αντίτυπο και ενημερώνει το κατωφλί του. Αντίθετα, ο αλγόριθμος COORD-Kmeans δεν δημιουργεί αντίτυπο του πακέτου γιατί οι κόμβοι S και F ανήκουν στην ίδια ομάδα.

Για τους τρεις αλγορίθμους δρομολόγησης με δυναμική δημιουργία αντιτύπων το φαινόμενο μείωσης των προωθήσεων είναι εμφανές. Έτσι με την κατανεμημένη διαδικασία ομαδοποίησης κόμβων οδηγούμαστε σε αποφάσεις περικοπής προωθήσεων σε κόμβους οι οποίοι ανήκουν σε μικρότερου αλλά και ίδιου επιπέδου ομάδας. Η τελευταία περίπτωση περιορισμού των προωθήσεων (ίδια ομάδα) βοηθάει στον να μην προωθούνται αντίτυπα σε κόμβους που έχουν παρόμοια ικανότητα παράδοσης.

4.3.2 Ενημέρωση Πληροφορίας K-means (LVQ)

Ένα σημαντικό σημείο στο οποίο επικεντρώνεται η συγκεκριμένη ενότητα αποτελεί το πως ανανεώνουμε την πληροφορία ομαδοποίησης. Αυτό είναι σημαντικό να συμβαίνει αφού η εικόνα του δικτύου πιθανόν να αλλάζει στο πέρασμα του χρόνου. Μάλιστα είναι σύνηθες φαινόμενο ο σχηματισμός των ομάδων να έχει μεγάλες αποκλίσεις με την πάροδο του χρόνου. Έτσι είναι απαραίτητος ένας online τρόπος ενημέρωσης των κέντρων των ομάδων όχι μόνο για να μπορούμε να αξιολογούμε καλύτερα τους κόμβους κατά τις επαφές αλλά και για να χρησιμοποιούμε αποδοτικότερα την πληροφορία που μας παρέχει η μέθοδός μας. Η καταγραφή και η εκτέλεση του αλγορίθμου K-means δεν είναι αρκετή όταν αναφερόμαστε σε ένα οπορτουνιστικό δίκτυο το οποίο εξελίσσεται και αλλάζει με τον χρόνο. Κατά συνέπεια, η εικόνα που έχει ένας κόμβος για το δίκτυο την χρονική στιγμή t είναι πολύ πιθανό να μην συμβαδίζει με την εικόνα την στιγμή $t+1$. Χρειαζόμαστε οπότε έναν μηχανισμό ο οποίος θα ανανεώνει την ομαδοποίηση που έχει προκύψει από την εκτέλεση του αλγορίθμου K-means. Αυτό μπορεί να πραγματοποιηθεί με αλλαγή της θέσης των κέντρων των ομάδων ανάλογα με τις τιμές Utility τις οποίες μαθαίνει ένας κόμβος. Ο μηχανισμός αυτός τον οποίο δανειζόμαστε από την μάθηση με επίβλεψη είναι η εκμάθηση με διάνυσμα κβαντισμού (Learning Vector Quantization - LVQ).

Αλγόριθμος 4.2 Learning Vector Quantization (LVQ) for Clustering Update

- 1: The algorithm's input is:
- 2: - how many centers we have K
- 3: - the centers \vec{C}_i for $i = 1, \dots, K$
- 4: - how fast our system is learning η (learning rate)
- 5: - and an online data input \vec{U} .
- 6: The algorithm's flow is:
- 7: - Step 1: For next input \vec{U} find the closest center \vec{C}_m
- 8: - Step 2: Update \vec{C}_m . A better explanation is get \vec{C}_m closer to the input \vec{U} .

$$\vec{C}_m \leftarrow \vec{C}_m + \eta \times (\vec{U} - \vec{C}_m) \quad (4.3)$$

- 9: terminate;
-

Στην επιστήμη των υπολογιστών, η τεχνική LVQ αποτελεί έναν αλγόριθμο ταξινόμησης (μάθηση με επίβλεψη), ο οποίος βασίζεται σε πρότυπα. Εμείς χρησιμοποιούμε την ιδέα της τεχνικής LVQ για την ενημέρωση της πληροφορίας ομαδοποίησης των κόμβων. Έστω ότι έχουμε K κέντρα $\vec{C}_1, \dots, \vec{C}_K$ τα οποία έχουν προκύψει από τον αλγόριθμο ομαδοποίησης. Για κάθε νέο πρότυπο \vec{U} (Utility στην περίπτωση μας) αποφασίζεται με βάση μία μετρική απόστασης το κέντρο το οποίο βρίσκεται πιο κοντά στην πρότυπο αυτό. Η θέση του κέντρου αυτού θα πρέπει να ανανεωθεί. Εμείς χρησιμοποιούμε την παραπάνω λογική για την ενημέρωση της πληροφορίας των ομάδων που βλέπει ο κάθε κόμβος του δικτύου. Η online ενημέρωση των κέντρων πραγματοποιείται μέσω του μηχανισμού που περιγράφεται στον Αλγόριθμο 4.2. Να σημειώσουμε ότι ο αλγόριθμος αυτός χρησιμοποιείται σε έναν κόμβο, κάθε φορά που μαθαίνει μία νέα τιμή Utility (κατά τις επαφές του). Είναι σημαντικό να αναφέρουμε επίσης ότι το πλήθος των ενημερώσεων εξαρτάται απόλυτα από την τυχασιότητα των επαφών αλλά και από τα πακέτα που μεταφέρει ένας κόμβος. Έτσι σε μία επαφή ένας κόμβος χρησιμοποιεί την κάθε τιμή του Utility που μαθαίνει για να ανανεώσει την εικόνα κατάταξης των κόμβων που αντιλαμβάνεται. Το κέντρο το οποίο ενημερώνεται, όπως ήδη αναφέραμε, είναι το κέντρο της ομάδας στην οποία ανήκει (βρίσκεται πιο κοντά) η "εισερχόμενη" τιμή του Utility. Ένα Utility είναι ένα βαθμωτό μέγεθος και όχι διάνυσμα όπως περιγράφεται στην γενική περίπτωση στον Αλγόριθμο 4.2. Κατά συνέπεια, η ανανέωση του κέντρου σε μια επαφή δύο κόμβων

A και B γράφεται ως:

$$C_m \leftarrow C_m + \eta \times (U_B - C_m) \quad (4.4)$$

όπου C_m το κέντρο στο οποίο ανήκει το Utility U_B και κατά συνέπεια και ο ίδιος ο κόμβος B .

ΚΕΦΑΛΑΙΟ 5

ΑΠΟΤΕΛΕΣΜΑΤΑ

5.1 Βασικά Αποτελέσματα

5.2 Παραμετροποίηση K-means & Αποτελέσματα

Στο κεφάλαιο αυτό θα παρουσιάσουμε και θα αναλύσουμε τα αποτελέσματα της έρευνας μας, η οποία βασίστηκε σε διαφορετικές προσομοιώσεις στον προσομοιωτή Adyton [18]. Ο προσομοιωτής αυτός είναι ανοιχτού κώδικα (open source) και προσομοιώνει ένα ομορτυνιστικό δίκτυο όπου μπορείς να επιλέξεις το trace, τον αλγόριθμο δρομολόγησης, το πλήθος των πακέτων που δημιουργούνται αρχικά στο δίκτυο, κ.α. Έτσι μπορείς να εξάγεις αποτελέσματα (ποσοστά επιτυχούς παράδοσης, προωθήσεις, ποσοστό περιττών προωθήσεων, καθυστέρηση, κ.τ.λ) από ήδη υλοποιημένους αλγορίθμους. Επιπλέον, σου δίνεται η δυνατότητα να εφαρμόσεις και να ενσωματώσεις μία δική σου ιδέα που αφορά την δρομολόγηση. Ουσιαστικά να υλοποιήσεις και να ενσωματώσεις στον προσομοιωτή τον δικό σου αλγόριθμο δρομολόγησης.

Στο κεφάλαιο αυτό αρχικά μελετάμε τα βασικά αποτελέσματα που αφορούν συγκρίσεις μεταξύ των αλγορίθμων CNR, DF και COORD με τις αντίστοιχες εκδοχές τους με εφαρμοσμένη την μέθοδο που προτείνουμε. Στην πράξη εφαρμόζουμε στους τρεις αλγορίθμους μια μέθοδο ομαδοποίησης των κόμβων όπως ακριβώς περιγράψαμε αναλυτικά στο Κεφάλαιο 4. Τις αντίστοιχες εκδοχές των τριών αλγορίθμων με υλοποιημένη την επέκταση ομαδοποίησης, τις έχουμε ονομάσει CNR-Kmeans, DF-

Kmeans και COORD-Kmeans. Σε ένα δεύτερο επίπεδο εξετάζουμε την μέθοδο μας και σε επίπεδο παραμετροποίησης. Έτσι ελέγχουμε για τον αλγόριθμο K-means το πλήθος των καταγραφών N . Επιπρόσθετα, εξετάζουμε την παράμετρο η (ρυθμός μάθησης) της τεχνικής LVQ η οποία όπως αναφέραμε αφορά την online ενημέρωση της ομαδοποίησης μετά την εκτέλεση του K-means. Για τις δύο παραμέτρους της μεθόδου μας προσπαθούμε μέσω των πειραμάτων να καταλήξουμε στην αποτελεσματικότερη επιλογή των τιμών τους. Τέλος, να σημειώσουμε ότι δεν παρατηρήθηκαν σημαντικές διαφοροποιήσεις των αποτελεσμάτων για διαφορετικό αριθμό επαναλήψεων για την ανεξαρτητοποίηση του αλγορίθμου από την αρχικοποίηση (περιγράφηκε στο Κεφάλαιο 4). Για τον λόγο αυτό και δεν παρουσιάζονται σχετικά αποτελέσματα καθώς δεν αποτελούν σημαντικό στοιχείο της αποτελεσματικότητας της μεθόδου μας. Μία τιμή που κυμαίνεται από 20 έως και 100 επαναλήψεις είναι ικανοποιητικές και δεν κοστίζουν σε χρόνο εκτέλεσης.

5.1 Βασικά Αποτελέσματα

Στην ενότητα αυτή παρουσιάζονται τα σχετικά αποτελέσματα συγκρίσεων μεταξύ των αλγορίθμων CNR, CNR-Kmeans, DF, DF-Kmeans, COORD και COORD-Kmeans. Εξετάσαμε και παρουσιάζουμε μέσω διαγραμμάτων και πινάκων την εκτίμηση της μεθόδου μας ως προς την κατανάλωση ενέργειας σε σχέση με τα ποσοστά επιτυχούς παράδοσης. Τα πειράματα έγιναν σε τέσσερα γνωστά traces, τα οποία και χρησιμοποιούνται κατά κόρον για την αποτίμηση νέων αλγορίθμων και τεχνικών δρομολόγησης. Τα traces αυτά τα οποία προέκυψαν από πραγματικά ομορτουριστικά δίκτυα είναι τα εξής: Reality [19], Milano [20], Infocom 2005 [21] και Dartmouth2weeks [22]. Περισσότερες λεπτομέρειες για τα traces παρουσιάζονται στον Πίνακα 5.1. Τα αποτελέσματα της μεθόδου μας (CNR-Kmeans, DF-Kmeans, COORD-Kmeans), όπως παρουσιάζονται σε όλα τα διαγράμματα, είναι κανονικοποιημένα (normalized) ως προς τις απλές εκδοχές των αλγορίθμων. Επιπρόσθετα, χρησιμοποιήθηκαν για την αποτίμηση της μεθόδου διαφορετικά Utilities τα οποία είναι τα εξής: LTS, DestEnc, Sim, Prophet, SPM, Enc, Bet και LastContact.

Πίνακας 5.1: Πίνακας πληροφοριών των traces που χρησιμοποιήθηκαν.

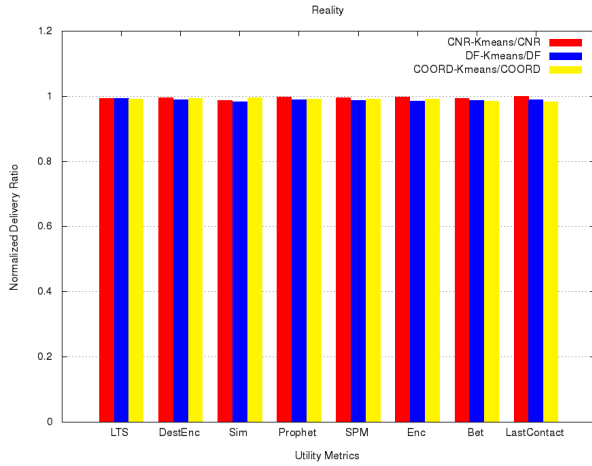
Trace	Περιοχή	Κόμβοι	Διάρκεια (μέρες)
MIT Reality	πανεπιστημιούπολη	97	83
Infocom 2005	συνέδριο	41	3
Milano pmtr	πανεπιστημιούπολη	44	18.9
Dartmouth2weeks	πανεπιστημιούπολη	739	14

5.1.1 Reality

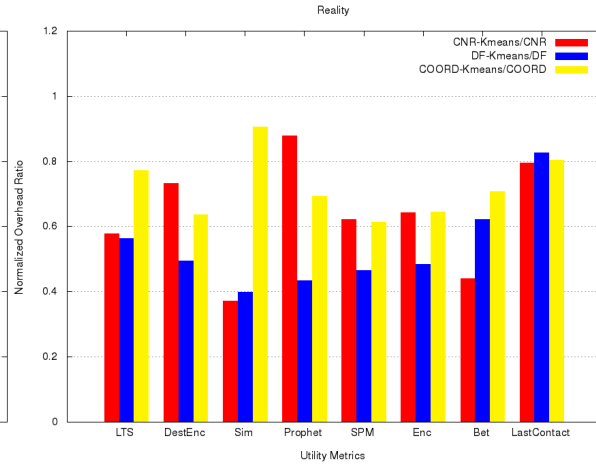
Το trace Reality έχει καταγραφεί στην πανεπιστημιούπολη του MIT, διήρκεσε 83 ημέρες και αποτελείται από 97 κόμβους. Στα αποτελέσματα τα οποία παρουσιάζονται στους πίνακες 5.2, 5.3, 5.4, 5.5 αλλά και στα δύο διαγράμματα 5.1α και 5.1β βλέπουμε ότι η μέθοδός την οποία και προτείνουμε επιτυγχάνει πολύ υψηλά ποσοστά παράδοσης με σημαντική μείωση της περιττής διακινούμενης πληροφορίας (Overhead).

Πιο συγκεκριμένα παρατηρώντας το διάγραμμα των ποσοστών επιτυχούς παράδοσης (Delivery Ratio 5.1α) βλέπουμε ότι οι αποκλίσεις στην ικανότητα παράδοσης είναι μηδαμινές σε κάθε σύγκριση με τους απλούς αλγόριθμους. Η ίδια εικόνα παρατηρείται και επιβεβαιώνεται από τους αντίστοιχους πίνακες ποσοστών επιτυχούς παράδοσης 5.2 και 5.4. Οι αποκλίσεις των αλγορίθμων που εφαρμόζουν την νέα τεχνική της ομαδοποίησης είναι της τάξης έως και 1.4% σε σύγκριση με τα ποσοστά των απλών αλγορίθμων CNR, DF και COORD. Το ποσοστό αυτό είναι ελαχίστης σημασίας αν αναλογιστεί κανείς την εκτεταμένη μείωση της περιττής πληροφορίας (Overhead Ratio 5.1β) που παρέχουν οι αλγόριθμοι με υλοποιημένη την μέθοδο μας.

Αναλυτικότερα, στην χειρότερη περίπτωση η μείωση που επιτυγχάνεται είναι της τάξης του 10%. Ο αλγόριθμος που πετυχαίνει την μείωση αυτή είναι ο COORD-Kmeans με Utility το Similarity. Εάν όμως δούμε τον Πίνακα 5.3 παρατηρούμε ότι στον απλό αλγόριθμο COORD για την περίπτωση χρήσης του Similarity, οι προωθήσεις βρισκόταν ήδη σε πολύ χαμηλά επίπεδα. Κατά συνέπεια, μια μείωση της τάξης του 10% είναι σαφώς σημαντική. Επιπρόσθετα, η δεύτερη χειρότερη περίπτωση (13%) του CNR-Kmeans με Utility το Prophet διερευνήθηκε μέσω επιπρόσθετης σχετικής έρευνας. Καταγράψαμε όλες τις τιμές της μετρικής σε κάθε κόμβο του δικτύου και παρατηρήσαμε απότομες αλλαγές στα επίπεδα των τιμών. Η φύση



(α) Reality: Delivery Ratio.



(β) Reality: Overhead Ratio.

Σχήμα 5.1: Reality: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.

Πίνακας 5.2: Reality: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	98.47916	97.77878	97.23832	98.89936	97.21846
CNR-Kmeans	97.8589	97.3786	95.97748	98.59926	96.73822
DF	96.83832	96.43798	95.17726	97.43854	95.35734
DF-Kmeans	96.09786	95.45744	93.5964	96.43808	94.07674
COORD	96.55814	95.75764	81.10878	97.07832	94.63696
COORD-Kmeans	95.67762	95.0973	80.70848	96.15776	93.71644

του Prophet περιορίζει τις δυνατότητες της μεθόδου μας καθώς έχουμε απότομες αλλαγές και στα κέντρα των ομάδων με την χρήση του μηχανισμού LVQ. Σημαντική παρατήρηση αποτελεί η εκτεταμένη μείωση στις περιπτώσεις των DF-Kmeans και COORD-Kmeans καθώς οι απλοί αλγόριθμοι DF και COORD αποτελούσαν τις αποτελεσματικότερες εκδοχές αλγορίθμων δρομολόγησης. Κοιτάζοντας συνολικά τα διαγράμματα για το συγκεκριμένο trace παρατηρούμε για την μέθοδο μας σταθερά ποσοστά επιτυχούς παράδοσης (Delivery Ratio) με μείωση της κατανάλωσης ενέργειας (Overhead Ratio) περίπου 40-45% στην μέση περίπτωση.

Πίνακας 5.3: Reality: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	746.1	494.546	466.389	801.164	431.652
CNR-Kmeans	430.712	362.475	173.422	703.4	268.494
DF	16.177	20.449	23.756	23.891	18.851
DF-Kmeans	9.109	10.114	9.456	10.394	8.789
COORD	8.583	13.284	3.737	11.957	12.268
COORD-Kmeans	6.629	8.461	3.388	8.282	7.517

Πίνακας 5.4: Reality: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	97.65864	96.5979	99.13952
CNR-Kmeans	97.4585	96.03748	99.03944
DF	95.15722	95.89752	85.95136
DF-Kmeans	93.75626	94.61686	85.07082
COORD	93.29616	93.55606	86.05152
COORD-Kmeans	92.55558	92.17538	84.61054

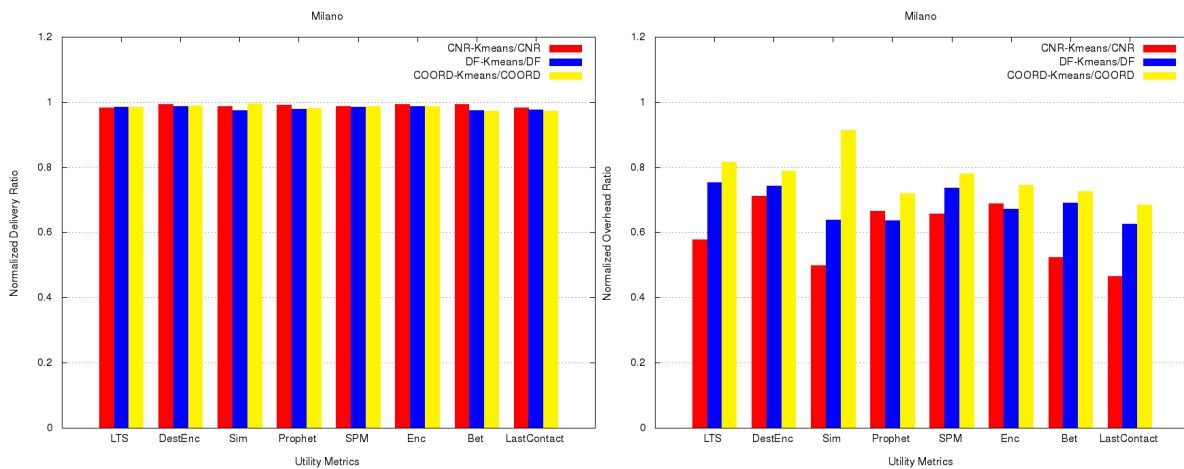
Πίνακας 5.5: Reality: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	428.88	410.47	819.004
CNR-Kmeans	275.285	180.981	651.957
DF	23.207	10.862	4.947
DF-Kmeans	11.245	6.749	4.091
COORD	13.079	6.363	4.875
COORD-Kmeans	8.422	4.5	3.915

5.1.2 Milano

Το trace Milano έχει καταγραφεί στην πανεπιστημιούπολη του πανεπιστημίου του Μιλάνο, διήρκεσε 18.9 ημέρες και αποτελείται από 44 κόμβους. Στο trace αυτό τα αποτελέσματα παρουσιάζονται στους πίνακες 5.6, 5.7, 5.8, 5.9 αλλά και στα δύο διαγράμματα 5.2α και 5.2β. Παρατηρούμε ότι και εδώ η μέθοδός μας επιτυγχάνει πολύ υψηλά ποσοστά παράδοσης με σημαντική μείωση των προωθήσεων περιττών αντιτύπων στο δίκτυο.

Αναλύοντας το διάγραμμα των ποσοστών επιτυχούς παράδοσης (Delivery Ratio 5.2α) βλέπουμε πολύ μικρές αποκλίσεις στην ικανότητα παράδοσης, οι οποίες και εδώ δεν είναι ιδιαίτερα σημαντικές. Οι πίνακες 5.6 και 5.8 επιβεβαιώνουν την παραπάνω εικόνα. Οι αποκλίσεις των αλγορίθμων στα ποσοστά παράδοσης που χρησιμοποιούν την νέα μέθοδο της ομαδοποίησης είναι της τάξης κάτω του 2% στη μέση περίπτωση. Αγγίζουν μέχρι και το πολύ 2.2% μείωση σε σύγκριση με τα ποσοστά των απλών αλγορίθμων. Το ποσοστό αυτό αποτελεί μικρής σημασίας διότι επιτυγχάνεται εκτεταμένη μείωση των περιττών προωθήσεων που "επιβαρύνουν" το δίκτυο.



(α) Milano: Delivery Ratio.

(β) Milano: Overhead Ratio.

Σχήμα 5.2: Milano: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.

Στην χειρότερη περίπτωση, η μείωση της κατανάλωσης ενέργειας η οποία επιτυγχάνεται στον αλγόριθμο COORD-Kmeans είναι της τάξης του 10%. Πάλι η μείωση αυτή επιτυγχάνεται με την χρήση του Similarity. Η μείωση αυτή είναι και εδώ σημαντική καθώς ο απλός αλγόριθμος COORD πετυχαίνει έτσι και αλλιώς πολύ μικρό

αριθμό προωθήσεων (Πίνακας 5.7). Στην μέση περίπτωση οι μειώσεις των αντιτύπων στο δίκτυο ξεπερνούν το 40% και σε κάποιες περιπτώσεις, αγγίζουν μέχρι και το 55%. Υπογραμμίζουμε ότι η εκτεταμένη μείωση της τάξης περίπου 25-27% στις περιπτώσεις των DF-Kmeans και COORD-Kmeans αποτελούν και εδώ τις μειώσεις μεγαλύτερης αξίας, καθώς στο trace αυτό οι απλοί αλγόριθμοι DF και COORD παρουσιάζουν ήδη πολύ χαμηλά ποσοστά κατανάλωσης ενέργειας.

Πίνακας 5.6: Milano: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	89.99328	88.24574	87.22116	92.06298	85.45336
CNR-Kmeans	88.50654	87.7034	86.09558	91.23924	84.42874
DF	83.24192	85.57378	84.38772	86.49704	80.97246
DF-Kmeans	82.01626	84.48874	82.1977	84.66878	79.7671
COORD	83.0813	84.95068	73.09554	86.15564	80.7716
COORD-Kmeans	81.85582	84.0868	72.69368	84.56834	79.68672

Πίνακας 5.7: Milano: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	93.482	41.242	35.756	90.357	29.956
CNR-Kmeans	54.06	29.346	17.83	60.064	19.715
DF	5.551	5.894	7.291	7.515	5.308
DF-Kmeans	4.184	4.385	4.657	4.776	3.909
COORD	4.607	5.249	2.515	6.322	4.797
COORD-Kmeans	3.757	4.136	2.298	4.555	3.745

Πίνακας 5.8: Milano: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	89.35078	86.57862	93.42956
CNR-Kmeans	88.82848	85.9559	91.8425
DF	83.9025	86.3171	89.41126
DF-Kmeans	82.84052	84.14748	87.4025
COORD	82.98136	79.14456	88.24624
COORD-Kmeans	81.85614	77.03464	85.79444

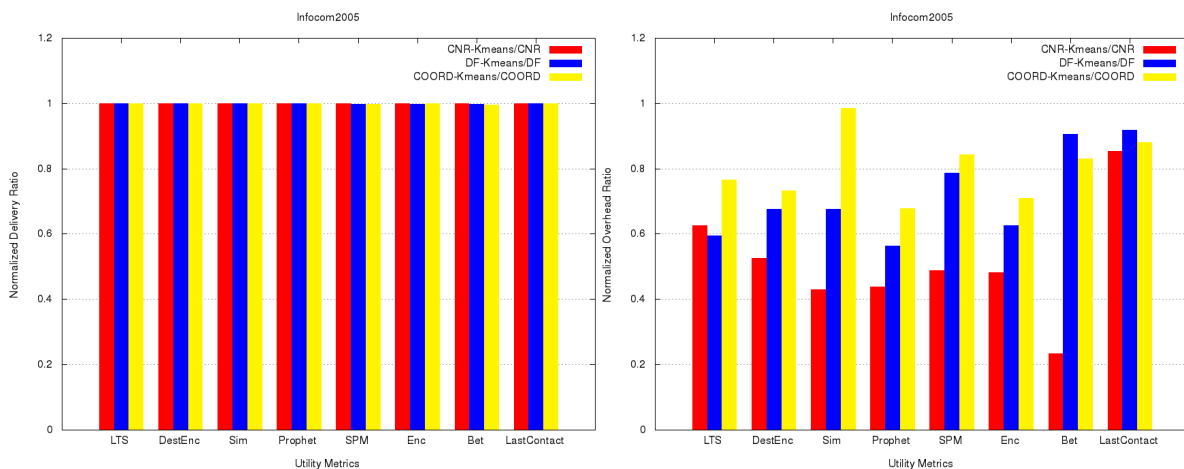
Πίνακας 5.9: Milano: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	36.042	59.493	113.147
CNR-Kmeans	24.815	31.105	52.556
DF	6.629	8.9	11.682
DF-Kmeans	4.46	6.153	7.321
COORD	5.367	4.976	7.954
COORD-Kmeans	3.998	3.618	5.441

5.1.3 Infocom 2005

Το trace Infocom 2005 έχει καταγραφεί σε ένα συνέδριο της IEEE στο Miami το οποίο διήρκεσε 3 ημέρες και αποτελείται από 41 κόμβους. Στην περίπτωση του Infocom 2005 τα αποτελέσματα, τα οποία βλέπουμε στους πίνακες 5.10, 5.11, 5.12, 5.13 αλλά και στα δύο διαγράμματα 5.3α και 5.3β, επιβεβαιώνουν και πάλι ότι η μέθοδός μας πετυχαίνει πολύ υψηλά ποσοστά παράδοσης με μεγάλη μείωση των περιττών προωθήσεων στο δίκτυο.

Το διάγραμμα των ποσοστών επιτυχούς παράδοσης (Delivery Ratio 5.3α) εμφανίζει προφανή σταθερότητα των ποσοστών σε κάθε περίπτωση σύγκρισης της μεθόδου μας με τους απλούς αλγορίθμους δρομολόγησης. Η ίδια εικόνα υπάρχει και στους αντίστοιχους πίνακες με τα ποσοστά επιτυχούς παράδοσης 5.10 και 5.12. Οι αποκλίσεις των αλγορίθμων που εφαρμόζουν την μέθοδο ομαδοποίησης με τους απλούς αλγορίθμους είναι της τάξης του 0.06%. Σε κάποιες περιπτώσεις έχουμε και μηδενικές μεταβολές (0%). Τα ποσοστά αυτά μπορούμε σαφώς να τα χαρακτηρίσουμε σταθερά.



(α) Infocom2005: Delivery Ratio.

(β) Infocom2005: Overhead Ratio.

Σχήμα 5.3: Infocom2005: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.

Οι μειώσεις των προωθήσεων παρουσιάζονται ξεκάθαρα εκτεταμένες και στο Διάγραμμα 5.3β αλλά και στους πίνακες 5.11 και 5.13. Το υψηλότερο ποσοστό μείωσης παρατηρείται για την περίπτωση χρήσης του Utility Ego-Betweeness στον αλγόριθμο CNR-Kmeans με ποσοστό 77%. Σημαντική μείωση έχουμε και στην περίπτωση χρήσης της μετρικής Prophet για τον ίδιο αλγόριθμο με ποσοστό 56%. Στις

περιπτώσεις των αλγορίθμων δρομολόγησης DF-Kmeans και COORD-Kmeans, οι μειώσεις του Overhead για Utilities: LTS, DestEnc, Prophet, SPM και Enc κυμαίνονται περίπου στο 40-45% στην μέση περίπτωση, ενώ έχουμε μικρότερες μειώσεις για τις υπόλοιπες μετρικές. Αξίζει να επισημανθεί επίσης ότι στις περιπτώσεις των αλγορίθμων DF-Kmeans και COORD-Kmeans παρά τις μικρές μεταβολές με χρήση των Similarity, Ego-Betweenness και LastContact οι προωθήσεις βρίσκονται ήδη σε πάρα πολύ χαμηλά επίπεδα στους απλούς αλγορίθμους (βλέπε Πίνακες 5.11, 5.13). Κατά συνέπεια, μία μείωση ακόμα και μικρής τάξης είναι ευπρόσδεκτη και αξίζει να αναφερθεί. Στην μέση περίπτωση η μέθοδος μας πετυχαίνει σημαντική μείωση των περιττών προωθήσεων της τάξης του 40-45% διατηρώντας σταθερά τα ποσοστά επιτυχούς παράδοσης.

Πίνακας 5.10: Infocom2005: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	97.88	97.76	94.0	98.1	97.48
CNR-Kmeans	97.82	97.72	94.0	98.06	97.4
DF	96.66	97.34	96.52	96.82	96.8
DF-Kmeans	96.6	97.32	96.44	96.7	96.64
COORD	96.62	97.28	91.46	96.76	96.76
COORD-Kmeans	96.54	97.26	91.46	96.66	96.62

Πίνακας 5.11: Infocom2005: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	264.087	99.543	36.105	228.675	43.545
CNR-Kmeans	165.589	52.2732	15.487	100.281	21.256
DF	10.22	7.256	5.033	8.307	5.25
DF-Kmeans	6.085	4.911	3.401	4.681	4.135
COORD	6.591	6.332	1.639	6.709	4.642
COORD-Kmeans	5.055	4.639	1.615	4.544	3.911

Πίνακας 5.12: Infocom2005: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	97.8	97.94	98.46
CNR-Kmeans	97.72	97.84	98.46
DF	97.32	97.4	97.32
DF-Kmeans	97.16	97.12	97.28
COORD	97.24	95.9	97.26
COORD-Kmeans	97.12	95.52	97.22

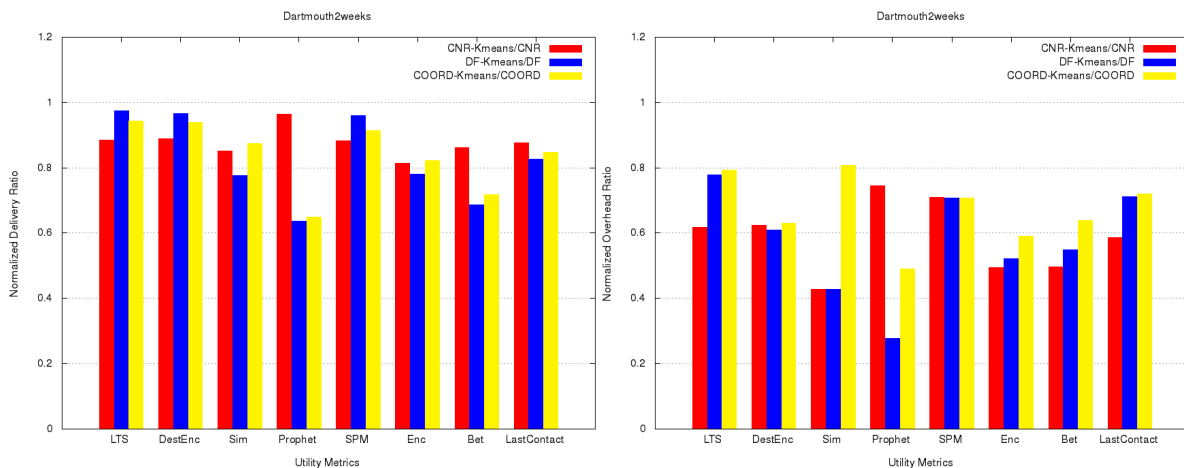
Πίνακας 5.13: Infocom2005: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	80.661	111.039	303.499
CNR-Kmeans	38.843	26.008	259.334
DF	8.59	4.966	3.39
DF-Kmeans	5.385	4.495	3.116
COORD	6.678	3.711	3.885
COORD-Kmeans	4.735	3.085	3.423

5.1.4 Dartmouth2weeks

Το trace Dartmouth2weeks έχει καταγραφεί στην πανεπιστημιούπολη του Dartmouth College, διήρκεσε 14 ημέρες και αποτελείται από 739 κόμβους. Στα αποτελέσματα τα οποία παρουσιάζονται στους πίνακες 5.14, 5.15, 5.16, 5.17 αλλά και στα δύο διαγράμματα 5.4α και 5.4β παρατηρούμε ότι η μέθοδός μας επιτυγχάνει πολύ διαφορετικά ποσοστά επιτυχούς παράδοσης. Οι μειώσεις παρόλα αυτά είναι σημαντικές σε κάθε περίπτωση αλγορίθμου που χρησιμοποιεί την μέθοδο μας και με οποιοδήποτε Utility.

Συγκεκριμένα για τις διακυμάνσεις των ποσοστών επιτυχούς παράδοσης ευθύνεται η δομή του trace. Το Dartmouth2weeks είναι ένα "trace πρόκληση" για κάθε αλγόριθμο δρομολόγησης λόγω του μεγάλου αριθμού των κόμβων και των επαφών στο δίκτυο. Επίσης, είναι ένα AP-based (Access Point-based) trace, κάτι το οποίο το διαφοροποιεί σημαντικά από τα υπόλοιπα traces που χρησιμοποιήσαμε. Στο Dartmouth2weeks βλέπουμε σημαντικές αποκλίσεις ακόμα και μεταξύ των απλών αλγορίθμων (βλέπε Πίνακες 5.14, 5.16). Κατά συνέπεια, αποτελεί μελλοντική δουλειά και πρόκληση η αντιμετώπιση του φαινομένου μείωσης των ποσοστών επιτυχούς παράδοσης της μεθόδου μας στο συγκεκριμένο trace.



(α) Dartmouth2weeks: Delivery Ratio.

(β) Dartmouth2weeks: Overhead Ratio.

Σχήμα 5.4: Dartmouth2weeks: Διαγράμματα ποσοστών επιτυχούς παράδοσης & περιττής πληροφορίας.

Η μείωση των ποσοστών δημιουργίας περιττής πληροφορίας (Overhead 5.4β) στο δίκτυο είναι και εδώ πολύ σημαντική με ποσοστά μείωσης έως και 75%. Στην

μέση περίπτωση η μείωση της κατανάλωσης ενέργειας που πετυχαίνουμε με την χρήση της μεθόδου μας κυμαίνεται περίπου στο 50% σε σχέση με τους απλούς αλγόριθμους με δυναμική δημιουργία αντιτύπων. Αυτό μας αποδεικνύει ότι η μέθοδος μας ακόμα και σε ιδιαίτερες περιπτώσεις ομορτυνιστικών δικτύων μπορεί να προσαρμοστεί και να πετύχει εν μέρει τον σκοπό της. Να αναφέρουμε επίσης ότι το Dartmouth2weeks έχει προκύψει από περικοπή του Dartmouth δικτύου του οποίου η διάρκεια είναι 5 χρόνια. Έτσι το Dartmouth2weeks είναι ένα ομορτυνιστικό δίκτυο το οποίο δεν έχει εξελιχθεί στον χρόνο. Να τονίσουμε ότι στις περιπτώσεις των προηγούμενων traces, οι κόμβοι του κάθε δικτύου τείνουν να έχουν έρθει σε επαφή με σχεδόν όλους του υπόλοιπους κόμβους του δικτύου. Πράγμα που σίγουρα δεν συμβαίνει στην περίπτωση αυτού του trace.

Πίνακας 5.14: Dartmouth2weeks: Πίνακας ποσοστών επιτυχούς παράδοσης (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	30.46952	30.06792	77.79992	95.1225	29.40502
CNR-Kmeans	26.97666	26.71562	66.19792	91.72986	25.95294
DF	27.25826	28.74298	69.16928	82.21606	27.03684
DF-Kmeans	26.57566	27.75952	53.69374	52.38924	25.95288
COORD	27.15788	28.7229	17.168225	78.3627	27.03684
COORD-Kmeans	25.59184	26.97664	15.01352	50.86366	24.6883

Πίνακας 5.15: Dartmouth2weeks: Πίνακας Overhead Ratio (Destination Dependent Utility Metrics).

Algorithm	LTS	DestEnc	Sim	Prophet	SPM
CNR	69.523	40.684	179.292	684.207	24.594
CNR-Kmeans	42.911	25.38	76.589	509.318	17.436
DF	2.699	4.479	17.588	40.2469	2.904
DF-Kmeans	2.101	2.733	7.511	11.188	2.052
COORD	2.124	3.598	1.178	18.165	2.283
COORD-Kmeans	1.686	2.264	0.952	8.901	1.614

Πίνακας 5.16: Dartmouth2weeks: Πίνακας ποσοτών επιτυχούς παράδοσης (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	70.13358	78.78306	96.7685
CNR-Kmeans	57.10624	67.88346	84.7857
DF	25.47166	49.37766	16.98052
DF-Kmeans	19.87094	33.94162	14.04986
COORD	20.41354	34.3235	15.57548
COORD-Kmeans	16.77976	24.66838	13.18702

Πίνακας 5.17: Dartmouth2weeks: Πίνακας Overhead Ratio (Destination Independent Utility Metrics).

Algorithm	Enc	Bet	LastContact
CNR	294.412	170.189	906.409
CNR-Kmeans	145.571	180.981	531.722
DF	16.8	18.586	9.63
DF-Kmeans	8.757	10.186	6.859
COORD	10.155	8.953	7.608
COORD-Kmeans	5.99	5.722	5.483

5.2 Παραμετροποίηση K-means & Αποτελέσματα

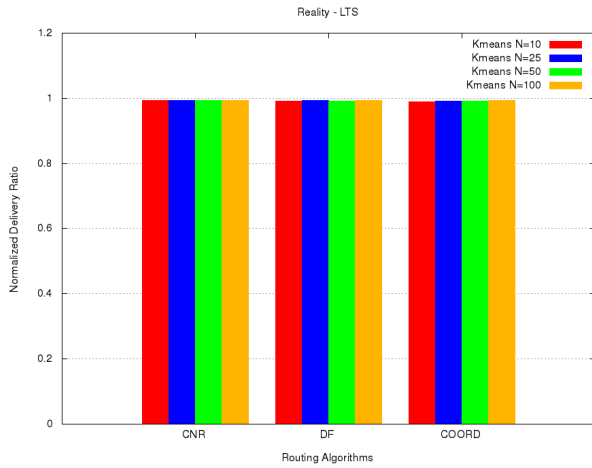
Σημαντικό και αναπόσπαστο κομμάτι αποτελεί και η κατάλληλη παραμετροποίηση της μεθόδου μας. Μπορούμε να αλλάξουμε σε κάποιο βαθμό την συμπεριφορά των αλγορίθμων δρομολόγησης τόσο σε ποσοστά παράδοσης όσο και σε αριθμό προωθήσεων. Παρακάτω παρουσιάζονται σε διαγράμματα και αναλύονται τα αποτελέσματα που προέκυψαν μετά από εκτενή έρευνα και πολλαπλές εκτελέσεις προσομοιώσεων στον Adyton [18]. Σε κάθε περίπτωση χρησιμοποιείται ένας μέσος όρος των ποσοστών επιτυχούς παράδοσης, των καθυστερήσεων αλλά και των περιττών προωθήσεων στο δίκτυο. Τα αποτελέσματα των διαγραμμάτων έχουν εξαχθεί με διαφορετικό πλήθος καταγραφών N του αλγορίθμου K-means 5.2.1 αλλά και για διαφορετικές τιμές του ρυθμού μάθησης η που αφορά τον μηχανισμό ενημέρωσης της ομαδοποίησης (LVQ) 5.2.2.

5.2.1 Επιλογή Πλήθους N Καταγραφών K-means

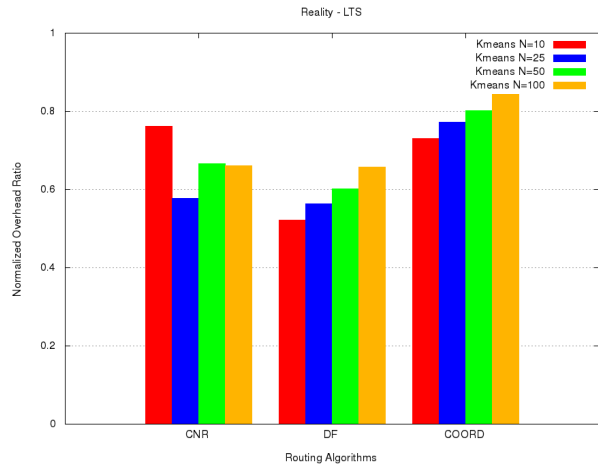
Στην υποενότητα αυτή εξετάζουμε το αντίκτυπο του πλήθους των καταγραφών του αλγορίθμου K-means. Η διαδικασία καταγραφής διαρκεί μέχρι να ξεπεραστεί το όριο (N) των καταγραφών. Τότε πραγματοποιείται η εκτέλεση του αλγορίθμου K-means για την ομαδοποίηση των κόμβων με βάση τις αξιολογήσεις τους (Utility values). Τα παρακάτω διαγράμματα προέκυψαν από εκτελέσεις των αλγορίθμων που εξετάζουμε στην εργασία αυτή χρησιμοποιώντας διαφορετικά Utilities. Οι μετρικές που χρησιμοποιήθηκαν και οι οποίες εμφανίζονται και στα διαγράμματα είναι οι εξής: LTS, DestEnc, Prophet και Ego-Betweeness. Τα πειράματα διεξήχθησαν στο trace Reality.

Χρήση μετρικής LTS (Last Time Seen)

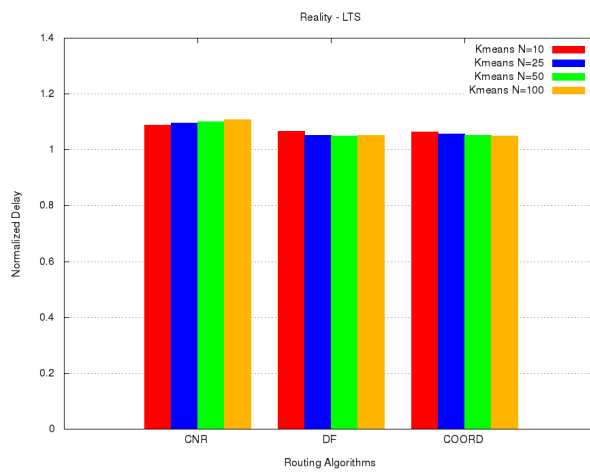
Αρχικά ας εξετάσουμε την περίπτωση χρήσης της μετρικής LTS και πως στην πράξη επηρεάζει η παράμετρος N τα αποτελέσματα της μεθόδου μας. Στο Διάγραμμα 5.5α βλέπουμε τα ποσοστά παράδοσης να κυμαίνονται στα ίδια επίπεδα για κάθε διαφορετική τιμή του πλήθους καταγραφών N . Τα ποσοστά επιτυχούς παράδοσης οπότε διατηρούνται σε υψηλό επίπεδο με πολύ μικρές αποκλίσεις. Κατά συνέπεια, αν δούμε μόνο το διάγραμμα αυτό μπορούμε να καταλήξουμε στο συμπέρασμα ότι η παράμετρος δεν επηρεάζει σχεδόν καθόλου την ικανότητα παράδοσης των κόμβων.



(α) Reality - LTS: Delivery Ratio.



(β) Reality - LTS: Overhead Ratio.



(γ) Reality - LTS: Delay (Days).

Σχήμα 5.5: Reality - LTS: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.

Παρατηρώντας τα διαγράμματα 5.5β και 5.5γ μπορούμε να αποσαφηνίσουμε την καλύτερη τιμή για το πλήθος καταγραφών του K-means.

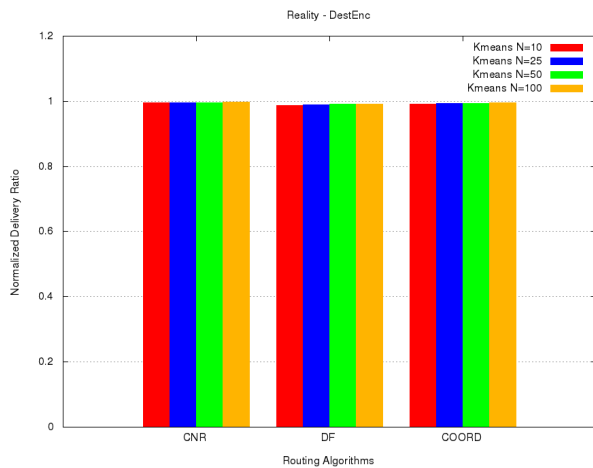
Για την περίπτωση του CNR-Kmeans βλέπουμε σαφέστατη μείωση των προωθήσεων για πλήθος καταγραφών $N = 25$. Η μείωση είναι της τάξης του 42% περίπου. Παρατηρούμε επίσης ότι για $N = 10$ προφανώς οι καταγραφές είναι λίγες για να λειτουργήσει αποδοτικά η χρήση της ομαδοποίησης. Παρόλα αυτά ακόμα και σε αυτήν την περίπτωση η μείωση των περιττών προωθήσεων είναι πολύ σημαντική. Από την άλλη πλευρά, εάν αυξήσουμε το πλήθος σε 50 και 100 καταγραφές βλέπουμε μία αύξηση της τάξης περίπου του 10% επάνω (άρα 35% μείωση από τον απλό

αλγόριθμο CNR) σε σχέση με την περίπτωση καταγραφής 25 τιμών της μετρικής LTS. Ο λόγος της αύξησης είναι ότι αφήνουμε τον αλγόριθμο να λειτουργεί όπως ο απλός CNR αλγόριθμος για περισσότερο χρονικό διάστημα. Έχουμε κατά συνέπεια μια μεγαλύτερης διάρκειας περίοδο εκμάθησης με αποτέλεσμα την αύξηση κατά 10% των προωθήσεων από την ιδανική περίπτωση (για $N = 25$).

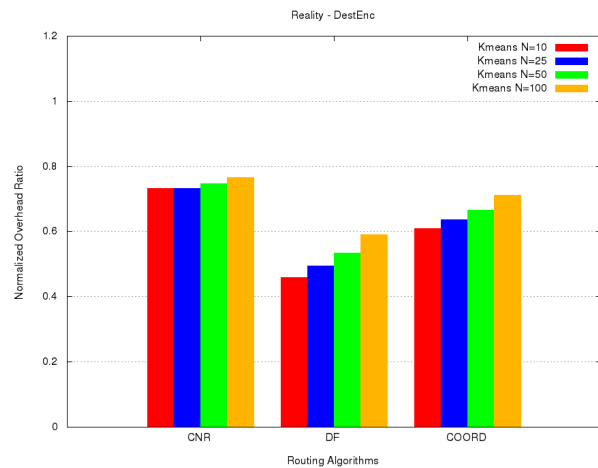
Στην περίπτωση του αλγορίθμου DF-Kmeans η χρήση της μεθόδου μας αποδίδει λιγότερες προωθήσεις σε σχέση με τον απλό DF αλγόριθμο από 35 έως και 46% ανάλογα με το πλήθος το καταγραφών N . Η μεγαλύτερη μείωση παρατηρείται για $N = 10$. Από εκεί και πέρα έχουμε μικρότερη μείωση προωθήσεων όσο αυξάνουμε την παράμετρο N . Παρατηρώντας όμως το διάγραμμα 5.5γ βλέπουμε διαφοροποιήσεις στην καθυστέρηση. Η μεγαλύτερη καθυστέρηση παρατηρείται κατά την καταγραφή 10 τιμών του Utility. Οι υπόλοιπες περιπτώσεις δίνουν την ίδια περίπου καθυστέρηση. Κατά συνέπεια, μια επιλογή $N = 25$ καταγραφών είναι και πάλι η βέλτιστη επιλογή κοιτάζοντας και τα δύο διαγράμματα 5.5β και 5.5γ. Ανάλογη εικόνα έχουμε και στις περιπτώσεις χρήσης του αλγορίθμου COORD-Kmeans. Τα ποσοστά μείωσης της περιττής πληροφορίας που επιτυγχάνει η μέθοδος μας κυμαίνονται από 16% έως και 28% σε σχέση με τον απλό αλγόριθμο. Παρατηρώντας και τα δύο διαγράμματα 5.5α και 5.5β για τις περιπτώσεις των δύο τελευταίων αλγορίθμων (DF-Kmeans και COORD-Kmeans) καταλήγουμε στο συμπέρασμα ότι η περίπτωση καταγραφής $N = 25$ είναι η καλύτερη επιλογή. Επιλέγουμε την τιμή αυτή γιατί θέλουμε να έχουμε μία ισορροπία της καθυστέρησης παράδοσης και των περιττών προωθήσεων.

Χρήση Μετρικής DestEnc (Destination Encounters)

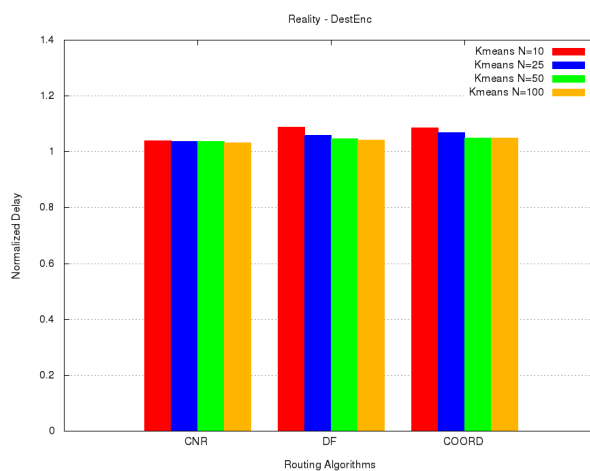
Με χρήση της μετρικής DestEnc καταλήγουμε στο συμπέρασμα ότι το πλήθος καταγραφών $N = 25$ είναι και πάλι η βέλτιστη επιλογή. Στο Διάγραμμα 5.6α βλέπουμε τα ποσοστά παράδοσης να κυμαίνονται στα ίδια επίπεδα για κάθε διαφορετική τιμή του πλήθους καταγραφών N . Άρα έχουμε σταθερά ποσοστά επιτυχούς παράδοσης για κάθε τιμή της παραμέτρου N . Παρατηρώντας τα διαγράμματα 5.6β και 5.6γ μπορούμε εύκολα να καταλήξουμε στην επιλογή της τιμής $N = 25$.



(α) Reality - DestEnc: Delivery Ratio.



(β) Reality - DestEnc: Overhead Ratio.



(γ) Reality - DestEnc: Delay (Days).

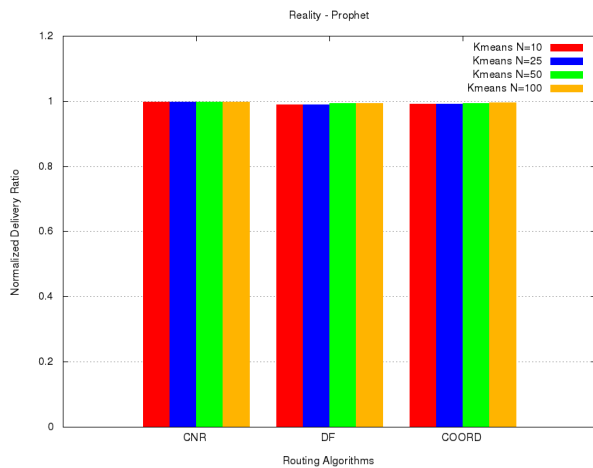
Σχήμα 5.6: Reality - DestEnc: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.

Στην περίπτωση του αλγορίθμου CNR-Kmeans παρατηρούμε σημαντική μείωση των προωθήσεων για πλήθος καταγραφών $N = 10$ και $N = 25$. Η μείωση είναι της τάξης του 25% περίπου και στις δυο περιπτώσεις. Παρατηρούμε στο Διάγραμμα 5.6γ ότι για όλες τις τιμές της παραμέτρου οι καθυστερήσεις στον CNR-Kmeans είναι περίπου ίδιες. Για τον αλγόριθμο DF-Kmeans η χρήση της μεθόδου μας αποδίδει πολύ λιγότερες προωθήσεις (Διάγραμμα 5.6β) σε σχέση με τον απλό αλγόριθμο. Η μείωση είναι της τάξης του 41-55% ανάλογα το πλήθος το καταγραφών N . Η μεγαλύτερη μείωση παρατηρείται για $N = 10$. Το ποσοστό μείωσης των προωθήσεων αυξάνεται όσο αυξάνουμε την τιμή της παραμέτρου N . Το Διά-

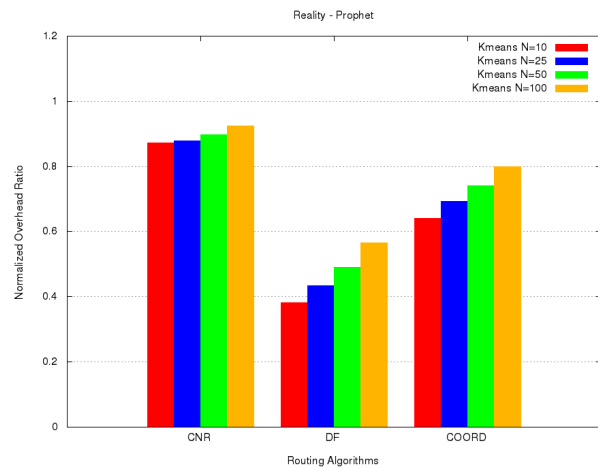
γραμμα 5.6γ όμως μας δείχνει διαφοροποιήσεις στην καθυστέρηση παράδοσης για διαφορετικές τιμές της παραμέτρου. Η μεγαλύτερη καθυστέρηση παρατηρείται για $N = 10$. Οι υπόλοιπες περιπτώσεις βρίσκονται σε φθίνουσα κατάταξη λόγω του μεγαλύτερου αριθμού προωθήσεων όπως φαίνεται και στο Διάγραμμα 5.6β. Η ίδια εικόνα για την καθυστέρηση υπάρχει και με την χρήση του αλγορίθμου COORD-Kmeans. Τα ποσοστά μείωσης της περιττής πληροφορίας κυμαίνονται από 30% ($N = 100$) έως και 40% ($N = 10$) σε σχέση με τον απλό αλγόριθμο COORD. Παρατηρώντας το Διάγραμμα 5.6γ βλέπουμε την ίδια εικόνα για την καθυστέρηση που περιγράψαμε για τον αλγόριθμο DF-Kmeans. Όσο αυξάνεται η τιμή N τόσο μειώνεται η καθυστέρηση παράδοσης λόγω των παραπάνω προωθήσεων. Κατά συνέπεια, κοιτάζοντας και τα δύο διαγράμματα (αφού έχουμε σταθερά ποσοστά επιτυχούς παράδοσης) 5.6β και 5.6γ καταλήγουμε ότι η περίπτωση καταγραφής $N = 25$ είναι η καλύτερη επιλογή. Αυτό ισχύει γιατί επιθυμούμε να επιτύχουμε ισορροπία μεταξύ της καθυστέρησης παράδοσης με τον αριθμό των περιττών προωθήσεων.

Χρήση Μετρικής Prophet

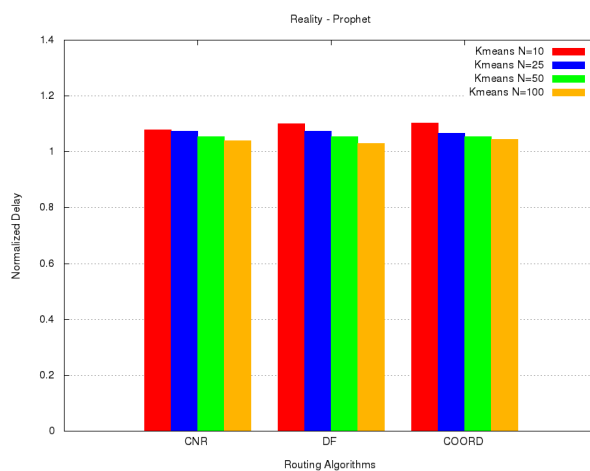
Με χρήση της μετρικής Prophet καταλήγουμε για τους ίδιους λόγους στο συμπέρασμα ότι το πλήθος καταγραφών $N = 25$ είναι και πάλι η καλύτερη επιλογή καταγραφών για τον αλγόριθμο K-means. Στο διάγραμμα 5.7α βλέπουμε σταθερά ποσοστά επιτυχούς παράδοσης για τις διαφορετικές τιμές του πλήθους των καταγραφών. Στα δύο διαγράμματα 5.7β και 5.7γ μπορούμε να καταλήξουμε στην επιλογή της τιμής $N = 25$ καθώς επιθυμούμε και πάλι μια ισορροπημένη κατάταξη μεταξύ καθυστέρησης και ποσοστού περιττής πληροφορίας (Overhead). Με την χρήση του Prophet παρατηρούμε μια πιο ξεκάθαρη μείωση της καθυστέρησης παράδοσης όσο αυξάνονται οι προωθήσεις στο δίκτυο.



(α) Reality - Prophet: Delivery Ratio.



(β) Reality - Prophet: Overhead Ratio.



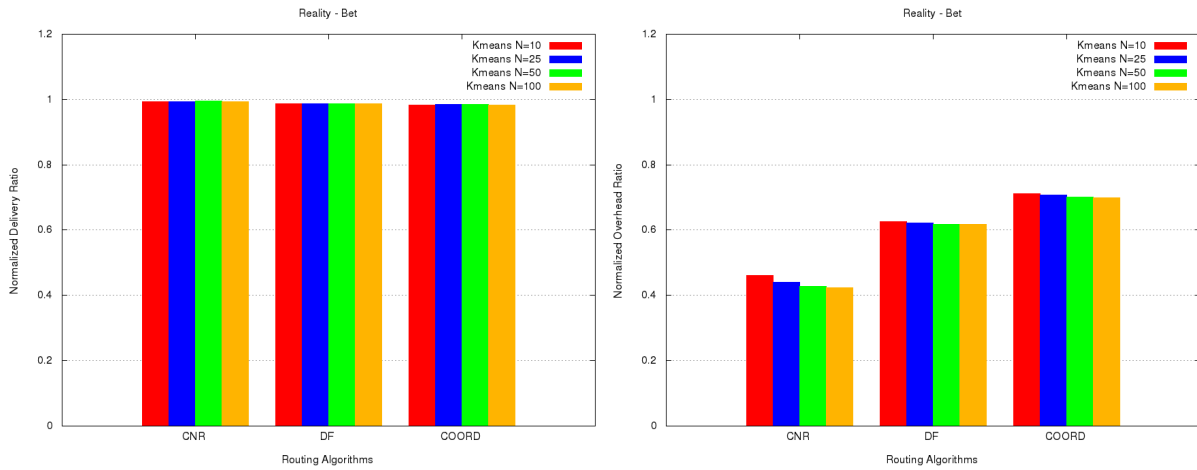
(γ) Reality - Prophet: Delay (Days).

Σχήμα 5.7: Reality - Prophet: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.

Χρήση Μετρικής Bet (Ego-Betweeness)

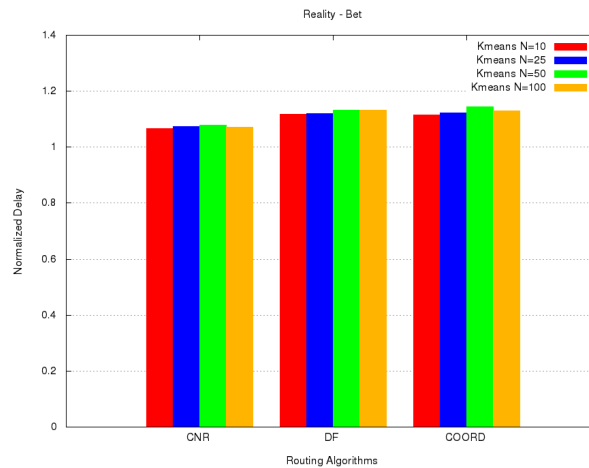
Τέλος, με χρήση της μετρικής Ego-Betweeness καταλήγουμε στο συμπέρασμα ότι το πλήθος καταγραφών $N = 25$ είναι και εδώ η καλύτερη επιλογή. Στο Διάγραμμα 5.8α βλέπουμε σταθερά ποσοστά επιτυχούς παράδοσης για τις διαφορετικές τιμές του πλήθους των καταγραφών. Στο Διάγραμμα 5.8β βλέπουμε μεγαλύτερη μείωση των περιττών αντιτύπων όσο αυξάνουμε την τιμή της παραμέτρου N . Παρόλα αυτά οι αποκλίσεις δεν είναι μεγάλες. Κοιτάζοντας το Διάγραμμα 5.8γ μπορούμε να λάβουμε μία καλύτερη απόφαση για την επιλογή της τιμής των καταγραφών. Έχοντας υπόψιν και τα δύο διαγράμματα μπορούμε να πούμε πράγματι ότι η τιμή $N = 25$

είναι και εδώ η πιο αποτελεσματική. Ωστόσο, δεν μπορούμε να πούμε ότι κάποια άλλη τιμή για την παράμετρο N είναι απαγορευτική στην περίπτωση χρήσης του Utility Ego-Betweeness.



(α) Reality - Betweeness: Delivery Ratio.

(β) Reality - Betweeness: Overhead Ratio.



(γ) Reality - Betweeness: Delay (Days).

Σχήμα 5.8: Reality - Betweeness: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικό πλήθος N καταγραφών του K-means.

Συνοψίζοντας, παρόμοια είναι η εικόνα που παρουσιάζεται σε κάθε περίπτωση. Τα ποσοστά επιτυχούς παράδοσης δεν παρουσιάζουν σημαντικές διακυμάνσεις. Αντίθετα, οι προωθήσεις στην μέση περίπτωση εξαρτώνται σε κάποιο βαθμό από το πλήθος καταγραφών N . Σε ένα γενικό πλαίσιο ισχύει ότι όσο αυξάνουμε το πλήθος καταγραφών τόσο αυξάνεται και η περίοδος μάθησης. Το αποτέλεσμα είναι ο

εκάστοτε αλγόριθμος να συμπεριφέρεται όλο και πιο κοντά στον απλό αλγόριθμο ο οποίος δεν εκμεταλλεύεται την πληροφορία της κατάταξης των κόμβων σε "ομάδες καταλληλότητας". Γενικά στην μέση περίπτωση φαίνεται η τιμή $N = 25$ να είναι αποτελεσματική λόγω του ισοζυγίου (trade-off) που υπάρχει μεταξύ καθυστέρησης παράδοσης και ποσοστών περιττής πληροφορίας.

5.2.2 Προσαρμογή στις Μεταβολές - Επιλογή Ρυθμού Μάθησης η (LVQ)

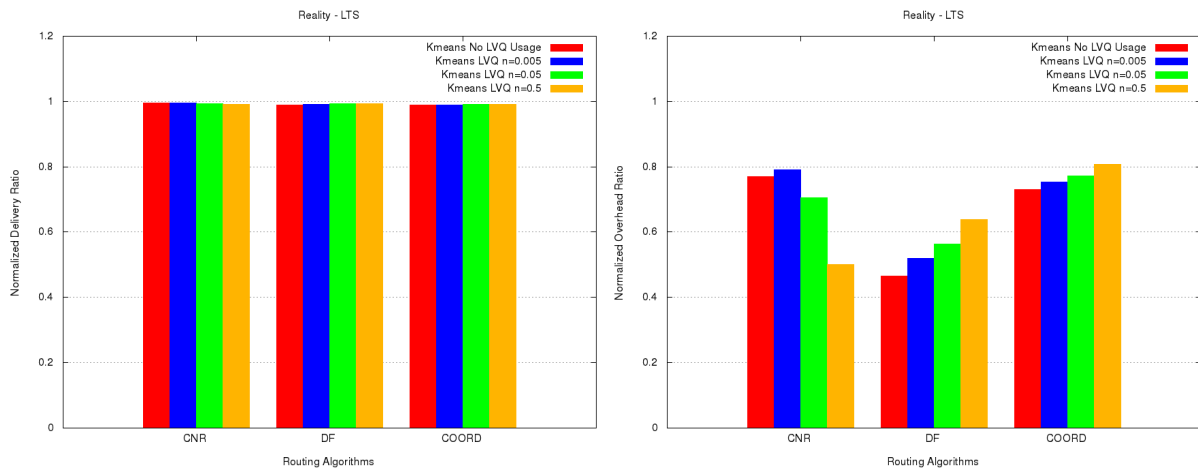
Στην υποενότητα αυτή εξετάζουμε το πόσο επηρεάζεται η απόδοση της μεθόδου μας από την παράμετρο που σχετίζεται με την ενημέρωση της ομαδοποίησης. Η διαδικασία ενημέρωσης εξαρτάται άμεσα από την παράμετρο η (ρυθμός μάθησης) του μηχανισμού LVQ. Για την παράμετρο χρησιμοποιήθηκαν διαφορετικές τιμές και εξετάστηκαν περιπτώσεις χρήσης διαφορετικών Utilities. Οι μετρικές οι οποίες χρησιμοποιήθηκαν είναι και εδώ οι LTS, DestEnc, Prophet και Ego-Betweeness. Τα πειράματα διεξήχθησαν και πάλι στο trace Reality με σταθερό πλήθος καταγραφών $N = 25$ του K-means. Στα παρακάτω διαγράμματα παρουσιάζονται αποτελέσματα των CNR-Kmeans, DF-Kmeans και COORD-Kmeans σε σύγκριση με τους αντίστοιχους απλούς αλγορίθμους. Επίσης, εξετάζεται και η περίπτωση όπου δεν χρησιμοποιείται καθόλου ο μηχανισμός LVQ.

Χρήση Μετρικής LTS (Last Time Seen)

Αρχικά ας εξετάσουμε τα αποτελέσματα με χρήση της μετρικής LTS. Στο Διάγραμμα 5.9α παρατηρούμε ότι τα ποσοστά παράδοσης είναι σχεδόν σταθερά για οποιαδήποτε περίπτωση που εξετάστηκε (No LVQ Usage, $\eta = 0.005$, $\eta = 0.05$ και $\eta = 0.5$). Κατά συνέπεια, δεν μπορούμε να λάβουμε μία απόφαση για την καλύτερη τιμή του ρυθμού μάθησης η .

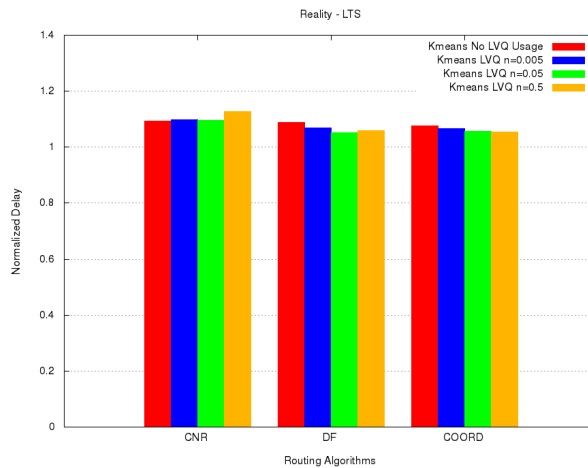
Στο Διάγραμμα 5.9β παρατηρούμε ότι όσο αυξάνουμε την τιμή της παραμέτρου η τόσο αυξάνονται οι προωθήσεις για τους αλγορίθμους DF-Kmeans και COORD-Kmeans. Αυτό συμβαίνει γιατί ο ρυθμός μάθησης ουσιαστικά ορίζει πόσο μεγάλη είναι η μεταβολή του κέντρου στο οποίο ανήκει ο κόμβος επαφής. Η απότομη μεταβολή των κέντρων των ομάδων οδηγεί σε λανθασμένη εικόνα κατάταξης των κόμβων. Κατά συνέπεια, υπάρχει και αλλοίωση του κριτηρίου προώθησης της μεθόδου μας. Επιπρόσθετα, όσο αυξάνουμε τον ρυθμό μάθησης μειώνεται η καθυστέρηση παράδο-

σης και για τους δύο αλγορίθμους. Η εικόνα αυτή φαίνεται και στο Διάγραμμα 5.9γ λόγω της αύξησης των προωθήσεων που αναφέραμε προηγουμένως.



(α) Reality - LTS: Delivery Ratio.

(β) Reality - LTS: Overhead Ratio.



(γ) Reality - LTS: Delay (Days).

Σχήμα 5.9: Reality - LTS: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.

Για τον αλγόριθμο CNR-Kmeans πρέπει να αναφέρουμε ότι δεν έχουμε αποσαφηνίσει σε απόλυτο βαθμό την εικόνα που παίρνουμε στο Διάγραμμα 5.9β. Ένας λόγος της ιδιαίτερης συμπεριφορά τους είναι ότι το συγκεκριμένο Utility υπολογίζεται σε πολλές χιλιάδες δευτερόλεπτα. Έτσι μια μικρή τιμή της παραμέτρου δεν μπορεί να μεταβάλει αισθητά ένα κέντρο. Επομένως, διαστρεβλώνεται και η εικόνα κατάταξης των κόμβων με συνέπεια την μη αποτελεσματική λήψη απόφασης για την προώθηση αντιτύπων. Στο Διάγραμμα 5.9γ παρατηρούμε μια αύξηση της

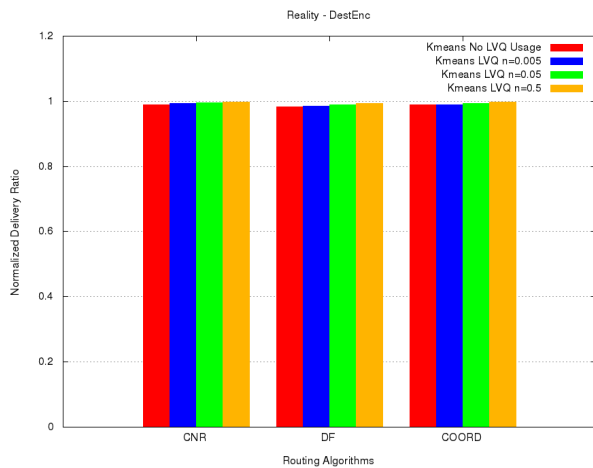
καθυστερήσης στον CNR-Kmeans για τιμή της παραμέτρου $\eta = 0.5$. Αυτό δικαιολογείται από τις πολύ μειωμένες προωθήσεις περιττών αντιτύπων όπως φαίνονται στο Διάγραμμα 5.9β.

Τελικά, εάν λάβουμε υπόψιν μας και τα τρία διαγράμματα, μια τιμή η κοντά στο 0.05 είναι στην μέση περίπτωση η αποδοτικότερη επιλογή. Ο λόγος είναι κυρίως ότι η μέθοδός μας πετυχαίνει έτσι μικρότερες καθυστερήσεις παράδοσης και δημιουργεί ένα ικανοποιητικά μικρό αριθμό προωθήσεων. Να σημειώσουμε ότι γενικότερα καμία τιμή δεν είναι απαγορευτική καθώς η μείωση της κατανάλωσης ενέργειας είναι αισθητά μεγάλη σε όλες τις περιπτώσεις σε σχέση με τους απλούς αλγορίθμους. Ακόμα και χωρίς την χρήση του μηχανισμού LVQ έχουμε σημαντική μείωση των προωθήσεων με διατήρηση υψηλών ποσοστών επιτυχούς παράδοσης.

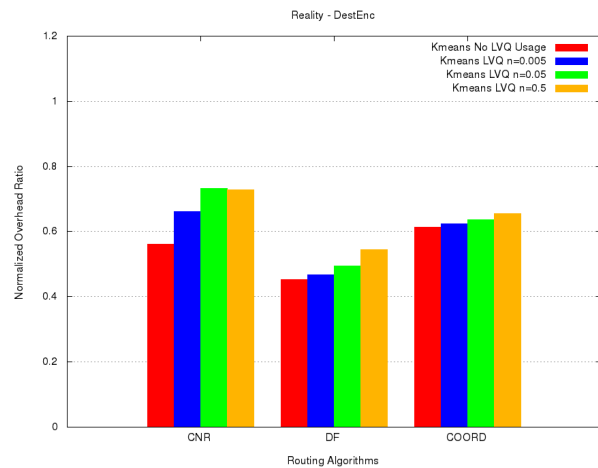
Χρήση Μετρικής DestEnc (Destination Encounters)

Με χρήση της μετρικής DestEnc παρατηρούμε στο Διάγραμμα 5.10α ότι και εδώ τα ποσοστά παράδοσης είναι σχεδόν σταθερά σε όλες τις περιπτώσεις. Ακόμα και όταν ο μηχανισμός LVQ δεν χρησιμοποιείται τα αποτελέσματα είναι ικανοποιητικά. Έτσι για να καταλήξουμε στην αποτελεσματικότερη τιμή η πρέπει να εξετάσουμε και τα διαγράμματα των ποσοστών περιττής πληροφορίας 5.10β και της καθυστέρησης παράδοσης 5.10γ.

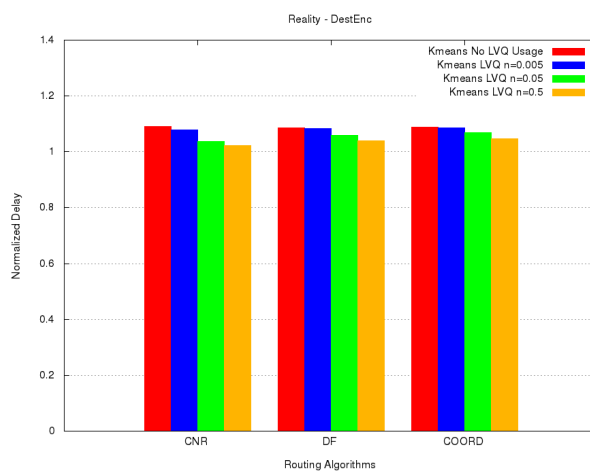
Στο Διάγραμμα 5.10β παρατηρούμε την ίδια εικόνα που περιγράψαμε και παραπάνω για τους αλγορίθμους DF-Kmeans και COORD-Kmeans. Μόνο που στην περίπτωση χρήσης της μετρικής DestEnc η εικόνα αυτή υπάρχει και στους τρεις αλγορίθμους που εφαρμόζουν την μέθοδο μας. Στην πράξη όσο αυξάνουμε την τιμή της παραμέτρου η τόσο αυξάνονται οι προωθήσεις. Η απότομη μεταβολή των κέντρων των ομάδων οδηγεί σε λανθασμένη εικόνα κατάταξης των κόμβων, αλλοιώνοντας το κριτήριο προώθησης. Στο Διάγραμμα 5.10γ παρατηρούμε την αναμενόμενη εικόνα μείωσης της καθυστέρησης παράδοσης όσο αυξάνουμε τον ρυθμό μάθησης του μηχανισμού LVQ. Η χειρότερη περίπτωση σαφώς παρουσιάζεται όταν δεν χρησιμοποιούμε καθόλου τον μηχανισμό LVQ.



(α) Reality - DestEnc: Delivery Ratio.



(β) Reality - DestEnc: Overhead Ratio.



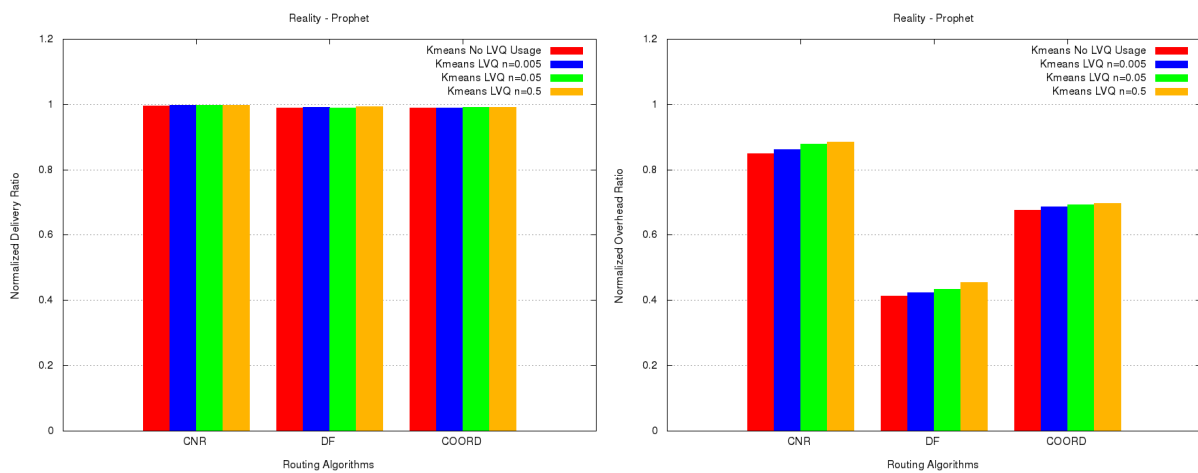
(γ) Reality - DestEnc: Delay (Days).

Σχήμα 5.10: Reality - DestEnc: Διαγράμματα ποσοτών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.

Αν λάβουμε υπόψιν μας συνολικά την εικόνα και των τριών διαγραμμάτων, μια τιμή η και πάλι κοντά στο 0.05 είναι αποτελεσματική στην μέση περίπτωση. Η επιλογή αυτή βασίζεται στον συνδυασμό των υψηλών ποσοτών επιτυχούς παράδοσης, των χαμηλών καθυστερήσεων και του αριθμού των περιττών προωθήσεων στο δίκτυο. Ωστόσο, και με την χρήση της μετρικής DestEnc δεν είναι απαγορευτική καμία τιμή.

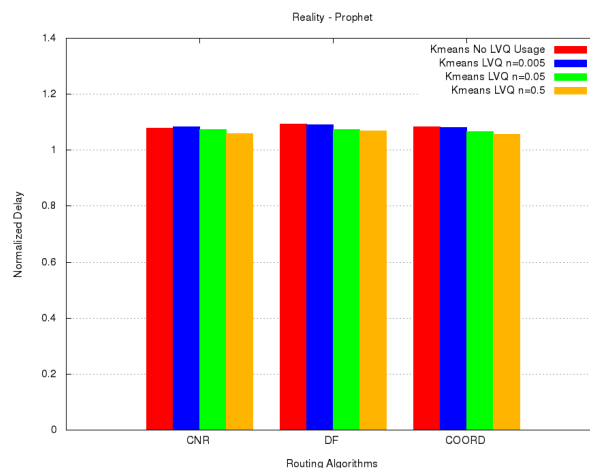
Χρήση Μετρικής Prophet

Με χρήση της μετρικής Prophet βλέπουμε στο Διάγραμμα 5.11α ότι τα ποσοστά επιτυχούς παράδοσης είναι σταθερά για οποιαδήποτε περίπτωση. Για να αποφασίσουμε ποια είναι η αποτελεσματικότερη τιμή της παραμέτρου η πρέπει να εξετάσουμε και τα διαγράμματα 5.11β και 5.10γ. Στο Διάγραμμα 5.11β παρατηρούμε την αναμενόμενη εικόνα που περιγράψαμε και παραπάνω. Όσο αυξάνουμε τον ρυθμό μάθησης η τόσο αυξάνονται και οι προωθήσεις. Στο Διάγραμμα 5.11γ παρατηρούμε, όπως είναι λογικό, την μείωση της καθυστέρησης παράδοσης όσο αυξάνουμε τον ρυθμό μάθησης του μηχανισμού LVQ λόγω της αύξησης των προωθήσεων.



(α) Reality - Prophet: Delivery Ratio.

(β) Reality - Prophet: Overhead Ratio.



(γ) Reality - Prophet: Delay (Days).

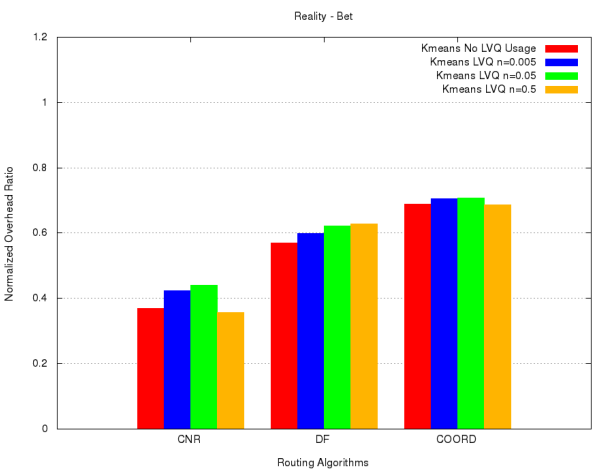
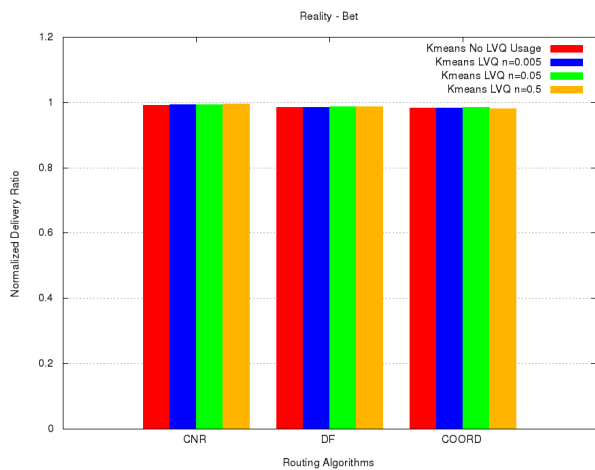
Σχήμα 5.11: Reality - Prophet: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.

Λαμβάνοντας υπόψιν την εικόνα και των τριών διαγραμμάτων, η τιμή $\eta = 0.05$ αποτελεί την αποδοτικότερη επιλογή. Η επιλογή αυτή βασίζεται στην διατήρηση μιας ισορροπίας μεταξύ των υψηλότερων ποσοστών επιτυχούς παράδοσης, των χαμηλών καθυστερήσεων και των ποσοστών περιττής πληροφορίας.

Χρήση Μετρικής Bet (Ego-Betweeness)

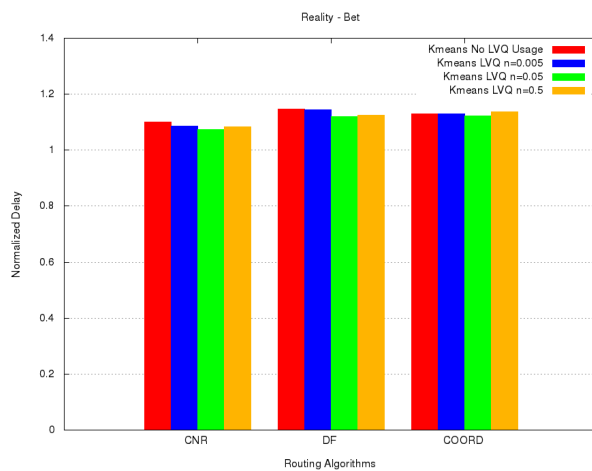
Στο Διάγραμμα 5.12α εμφανίζονται τα ποσοστά επιτυχούς παράδοσης για την περίπτωση χρήσης της μετρικής Ego-Betweeness. Παρατηρούμε και εδώ σταθερότητα των ποσοστών. Κατά συνέπεια, για την αποτελεσματική επιλογή του ρυθμού μάθησης εξετάζουμε τα διαγράμματα 5.12β και 5.12γ. Στο Διάγραμμα 5.12β παρατηρούμε μόνο για τον αλγόριθμο DF-Kmeans αύξηση των προωθήσεων όσο αυξάνουμε την τιμή της παραμέτρου η . Για τους δύο αλγορίθμους CNR-Kmeans και COORD-Kmeans ισχύει η ίδια εικόνα με εξαίρεση την τιμή 0.5, όπου και στις δύο περιπτώσεις έχουμε μείωση των προωθήσεων με μεγαλύτερη όμως καθυστέρηση παράδοσης (Διάγραμμα 5.12γ). Η εικόνα αυτή δεν έχει αποσαφηνιστεί απόλυτα καθώς έχει να κάνει και πάλι με την φύση της μετρικής Ego-Betweeness αλλά και με την τυχαιότητα των επαφών.

Παρόλα αυτά, εάν λάβουμε υπόψιν μας συνολικά την εικόνα και των τριών διαγραμμάτων μπορούμε να καταλήξουμε στο συμπέρασμα ότι στην μέση περίπτωση και πάλι η τιμή $\eta = 0.05$ φαίνεται να είναι η πιο αποτελεσματική. Δεν επιλέγουμε την τιμή 0.5 καθώς στις περιπτώσεις που υπερέχει έναντι της τιμής 0.05 παρατηρείται αύξηση της καθυστέρησης παράδοσης παρά την αισθητή μείωση των προωθήσεων.



(α) Reality - Betweenness: Delivery Ratio.

(β) Reality - Betweenness: Overhead Ratio.



(γ) Reality - Betweenness: Delay (Days).

Σχήμα 5.12: Reality - Betweenness: Διαγράμματα ποσοστών επιτυχούς παράδοσης, περιττής πληροφορίας και καθυστέρησης για διαφορετικές τιμές του ρυθμού μάθησης η της τεχνικής ενημέρωσης LVQ.

ΚΕΦΑΛΑΙΟ 6

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στα ομορτουμιστικά δίκτυα οι κόμβοι του δικτύου συνδέονται σε ζευγάρια για μικρά χρονικά διαστήματα και όχι μόνιμα. Η κινητικότητα τους σε συνεργασία με την τεχνική Store, Carry & Forward παίζει σημαντικό ρόλο στην διαδικασία δρομολόγησης κάθε πακέτου στο δίκτυο. Η υπάρχουσα τεχνολογία δεν μας δίνει την δυνατότητα να έχουμε φορητές συσκευές (κόμβους) με απεριόριστη ενέργεια και χώρο αποθήκευσης. Ο κυριότερος λόγος κατανάλωσης των πόρων ενός κόμβου είναι οι προωθήσεις περιττών αντιτύπων. Τα αντίτυπα αυτά στην πράξη δεν συμβάλλουν σημαντικά στην επιτυχή παράδοση ενός πακέτου.

Στην μεταπτυχιακή εργασία αυτή εξετάσαμε και αναλύσαμε μια νέα πρόταση, η οποία επεκτείνει τους υπάρχοντες αλγόριθμους δρομολόγησης στα ομορτουμιστικά δίκτυα. Οι αλγόριθμοι στους οποίους μπορεί να εφαρμοστεί η μέθοδος μας αφορά αλγόριθμους οι οποίοι χρησιμοποιούν μια μετρική αξιολόγησης (Utility). Η καινοτόμος μέθοδος αυτή παρουσιάζει σημαντική μείωση της κατανάλωσης ενέργειας καθώς επίσης και του χώρου αποθήκευσης στους κόμβους του δικτύου. Αυτό επιτυγχάνεται χρησιμοποιώντας την πληροφορία ομαδοποίησης των κόμβων που μας παρέχει ο αλγόριθμος K-means. Η online ενημέρωση της ομαδοποίησης προκύπτει μέσω του μηχανισμού LVQ κατά τις επαφές ενός κόμβου. Έτσι όταν ένας κόμβος έρχεται σε επαφή με έναν άλλο κόμβο του δικτύου μπορεί να ενημερώνει συνεχώς την πληροφορία ομαδοποίησης, μέσω των τιμών του Utility που μαθαίνει. Σε κάθε περίπτωση αλγόριθμου με δυναμική δημιουργία αντιτύπων, η εφαρμογή της μεθόδου μας πετυχαίνει εκτεταμένη μείωση των προωθήσεων αλλά και πολύ υψηλά

ποσοστά επιτυχούς παράδοσης. Ποσοστά τα οποία είναι ανάλογα των απλών αλγορίθμων CNR, DF και COORD.

Δοκιμάσαμε την μέθοδό μας στον προσομοιωτή Adyton [18]. Εφαρμόσαμε την μέθοδο μας στους αλγορίθμους CNR, DF και COORD, εξετάσαμε την συμπεριφορά της σε διαφορετικά traces και χρησιμοποιήσαμε διαφορετικά Utilities (Destination Dependent και Independent). Καταλήξαμε ότι οι αλγορίθμοι με εφαρμοσμένη την μέθοδο μας λειτουργούν πιο αποδοτικά ως προς την κατανάλωση ενέργειας ενώ ταυτόχρονα διατηρούν σταθερά και σε υψηλά επίπεδα τα ποσοστά παράδοσης. Στην μέση περίπτωση η μέθοδος μας επιφέρει μείωση της κατανάλωσης ενέργειας της τάξης του 40-45%. Ένα σημαντικό σημείο στο οποίο σταθήκαμε αποτελεί η αποδοτικότερη παραμετροποίηση του αλγορίθμου K-means και του μηχανισμού LVQ. Καταλήξαμε ότι ένα πλήθος καταγραφών $N = 25$ των τιμών του Utility είναι καθόλα ικανοποιητικό. Επίσης, για τον ρυθμό μάθησης του μηχανισμού LVQ οδηγηθήκαμε μέσω των αποτελεσμάτων στο συμπέρασμα ότι η τιμή $\eta = 0.05$ λειτουργεί ευεργετικά και αποτελεσματικά για την μέθοδο μας.

Κλείνοντας, μελλοντική δουλειά θα μπορούσε να αποτελεί η χρήση κάποιας άλλης μεθόδου ομαδοποίησης και η σύγκριση της με την αποτελεσματικότητα που παρέχει ο αλγόριθμος K-means. Επιπλέον, μια ακόμα μελλοντική πρόκληση αποτελεί η υλοποίηση της μεθόδου μας και σε άλλους αλγορίθμους δρομολόγησης που χρησιμοποιούν ένα Utility. Για παράδειγμα, μπορεί να εφαρμοστεί στις Spray-based τεχνικές. Τέλος, μια μελλοντική δουλειά θα μπορούσε να επικεντρωθεί και σε έναν διαφορετικό τρόπο ενημέρωσης των κέντρων. Ένα παράδειγμα αποτελεί η ενημέρωση της ομαδοποίησης με περιοδικές επανεκτελέσεις (σε χρονικά παράθυρα) του αλγορίθμου K-means.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] N. Papanikos, *Energy-efficient Networking in Wireless Ad Hoc Networks*. PhD thesis, 2017.
- [2] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, “Delegation forwarding,” 2008.
- [3] N. Papanikos and E. Papapetrou, “Coordinating replication decisions in multi-copy routing for opportunistic networks,” in *Proceedings of the 10th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2014*, pp. 8–13. DOI: 10.1109/WiMOB2014.6962142.
- [4] T. Spyropoulos, T. Turletti, and K. Obraczka, “Routing in delay-tolerant networks comprising heterogeneous node populations,” *IEEE Trans. Mobile Comput.*, 2009.
- [5] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, “Age matters: efficient route discovery in mobile ad hoc networks using encounter ages,” in *Proc. ACM Mobile Ad Hoc Netw. and Comput. (MobiHoc)*, 2003.
- [6] E. M. Daly and M. Haahr, “Social network analysis for routing in disconnected delay-tolerant manets,” in *ACM Proc. Int. Symposium Mobile Ad Hoc Netw. and Comput. (MobiHoc)*, 2007.
- [7] E. M. Daly and M. Haahr, “Social network analysis for information flow in disconnected delay-tolerant manets,” *IEEE Trans. Mobile Comput.*, 2009.
- [8] A. Lindgren, A. Doria, and O. Schelén, “Probabilistic routing in intermittently connected networks,” *ACM SIGMOBILE Mobile Comput. and Commun. Rev.*, 2003.

- [9] S. Grasic, E. Davies, A. Lindgren, and A. Doria, “The evolution of a dtn routing protocol - prophetv2,” in *Proc. ACM Workshop Challenged Netw. (CHANTS)*, 2011.
- [10] E. Bulut and B. K. Szymanski, “Friendship based routing in delay tolerant mobile social networks,” in *Global Telecommunications Conference (GLOBECOM 2010)*, 2010 *IEEE*, 2010.
- [11] E. Bulut and B. K. Szymanski, “Exploiting friendship relations for efficient routing in mobile social networks,” *IEEE Trans. Parallel Distrib. Syst.*, 2012.
- [12] S. C. Nelson, M. Bakht, and R. Kravets, “Encounter-based routing in dtns,” in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2009.
- [13] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, “Diversity of forwarding paths in pocket switched networks,” in *Proc. ACM SIGCOMM Internet Meas. Conf. (IMC)*, 2007.
- [14] L. C. Freeman, “A set of measures of centrality based on betweenness,” *Sociometry*, 1977.
- [15] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Efficient routing in intermittently connected mobile networks: The single-copy case,” *IEEE/ACM Trans. Netw.*, 2008.
- [16] D. B. A. Vahdat, “Epidemic routing for partially connected ad hoc networks,” in *Duke University*, 2000.
- [17] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Efficient routing in intermittently connected mobile networks: the multiple-copy case,” *IEEE/ACM Trans. Netw.*, 2008.
- [18] Nikolaos Papanikos, Dimitrios-Georgios Akestoridis, and Evangelos Pappetou, “Crawdad toolset tools/simulate/uoi/adyton (v. 2016-04-21).” downloaded from <http://crawdad.org/tools/simulate/uoi/adyton/20160421>, <https://doi.org/10.15783/C7BG6V>, Apr 2016.
- [19] N. Eagle and A. S. Pentland, “CRAWDAD dataset mit/reality (v. 2005-07-01).” Downloaded from <http://crawdad.org/mit/reality/20050701>, July 2005.

- [20] P. Meroni, S. Gaito, E. Pagani, and G. P. Rossi, "CRAWDAD dataset unimi/pmtr (v. 2008-12-01)." Downloaded from <http://crawdad.org/unimi/pmtr/20081201>, Dec. 2008.
- [21] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD dataset cambridge/haggle (v. 2009-05-29)." Downloaded from <http://crawdad.org/cambridge/haggle/20090529>, May 2009.
- [22] D. Kotz, T. Henderson, I. Abyzov, and J. Yeo, "CRAWDAD dataset dartmouth/campus (v. 2009-09-09)." Downloaded from <http://crawdad.org/dartmouth/campus/20090909>, Sept. 2009.

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ

Ο Βασίλειος Τουλατζής, γεννήθηκε στις 27 Νοεμβρίου στην πόλη της Καβάλας. Ολοκλήρωσε τις πρώτες σπουδές του στο τμήμα της Πληροφορικής του Πανεπιστημίου Ιωαννίνων το 2015. Ολοκληρώνει τις μεταπτυχιακές σπουδές του στο τμήμα Μηχανικών Η/Υ και Πληροφορικής του Πανεπιστημίου Ιωαννίνων. Επιπλέον, έχει πραγματοποιήσει δίμηνη πρακτική άσκηση στο Νοσοκομείο Καβάλας στο τμήμα Πληροφορικής (software και hardware engineering) κατά την καλοκαιρινή περίοδο Ιουλίου-Αυγούστου το έτος 2015, στα πλαίσια της πρακτικής άσκησης που προσφέρει το τμήμα Μηχανικών Η/Υ και Πληροφορικής. Τέλος, έχει εργαστεί ως IT Manager και Web Developer σε εταιρεία κατασκευής και φιλοξενίας ιστοσελίδων την περίοδο 2016-2017.