

ΔΡΟΜΟΛΟΓΗΣΗ ΑΝΕΚΤΙΚΗ ΣΤΗΝ ΚΑΘΥΣΤΕΡΗΣΗ ΣΕ ΔΙΚΤΥΑ ΜΕ ΜΕΙΚΤΑ
ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΣΥΝΔΕΣΙΜΟΤΗΤΑΣ

Η
ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΕΞΕΙΔΙΚΕΥΣΗΣ

Υποβάλλεται στην

ορισθείσα από την Γενική Συνέλευση Ειδικής Σύγκλησης
του Τμήματος Μηχανικών Η/Υ και Πληροφορικής
Εξεταστική Επιτροπή

από την

Αναστασία Κόντου

ως μέρος των Υποχρεώσεων

για τη λήψη

του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ
ΜΕ ΕΞΕΙΔΙΚΕΥΣΗ ΣΤΑ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

Φεβρουάριος 2014

ΑΦΙΕΡΩΣΗ

Στην οικογένειά μου...

ΕΥΧΑΡΙΣΤΙΕΣ

Η επιτυχημένη ολοκλήρωση της παρούσας εργασίας δεν θα ήταν δυνατή χωρίς την ουσιαστική αρωγή ορισμένων προσώπων, τα οποία νιώθω την ανάγκη να ευχαριστήσω.

Πρώτον απ' όλους θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Παπαπέτρου, για την πολύτιμη βοήθειά του και την υπομονή του σε όλη τη διάρκεια εκπόνησης της εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω τους γονείς μου για την ηθική και οικονομική τους βοήθεια όλα αυτά τα χρόνια.

Τέλος, επιθυμώ να εκφράσω τις ευχαριστίες μου στους φίλους μου, για την συναισθηματική υποστήριξη που μου προσέφεραν και ειδικά τον Παναγιώτη Οβρένοβιτς που ήταν πάντα δίπλα μου.

ΠΕΡΙΕΧΟΜΕΝΑ

	Σελ
ΑΦΙΕΡΩΣΗ	ii
ΕΥΧΑΡΙΣΤΙΕΣ	iii
ΠΕΡΙΕΧΟΜΕΝΑ	iv
ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ	vi
ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ	vii
ΠΕΡΙΛΗΨΗ	x
EXTENDED ABSTRACT IN ENGLISH	xii
ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ	1
1.1. Κινητά Ασύρματα Δίκτυα	1
1.2. Αντικείμενο της Διατριβής	5
1.3. Δομή της Διατριβής	8
ΚΕΦΑΛΑΙΟ 2. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΔΡΟΜΟΛΟΓΗΣΗΣ	9
2.1. Δρομολόγηση στα MANET	9
2.1.1. Destination- Sequenced Distance-Vector- DSDV	12
2.1.2. Optimized Link State Routing- OLSR	13
2.2. Δρομολόγηση στα DTN	15
2.2.1. Μετρικές Κοινωνικής Δικτύωσης	17
2.2.2. Ο Αλγόριθμος Simbet	19
2.3. Υβριδικοί Αλγόριθμοι	21
2.3.1. Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks- CAR	23
2.3.2. Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks -HYMAD	25
2.3.3. A Mobile Ad Hoc and Disruption Tolerant Routing Protocol for Tactical Military Networks- NOMAD	27

ΚΕΦΑΛΑΙΟ 3. ΠΡΟΤΕΙΝΟΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ	28
3.1. Περιγραφή του Προβλήματος	28
3.2. Ο Αλγόριθμος Hybrid Synchronous/Asynchronous Routing with Social Incentives- SASI	32
3.2.1. Σύγχρονη Λειτουργία	34
3.2.2. Ασύγχρονη Λειτουργία	38
3.3. Παραλλαγές του Αλγορίθμου SASI	45
ΚΕΦΑΛΑΙΟ 4. ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ	47
4.1. Περιβάλλον Προσομοίωσης- Εργαλεία	47
4.2. Κίνηση Κόμβων	48
4.3. Αποτελέσματα	56
4.3.1. Σενάριο Κίνησης Α	58
4.3.2. Σενάριο Κίνησης Β	63
4.3.3. Σενάριο Κίνησης Γ	67
4.3.4. Σενάριο Κίνησης Δ	71
4.3.5. Σενάριο Κίνησης Ε	75
4.3.6. Σενάριο Κίνησης ΣΤ	78
ΚΕΦΑΛΑΙΟ 5. ΣΥΜΠΕΡΑΣΜΑΤΑ	82
5.1. Ανασκόπηση	82
5.2. Προτάσεις για Περαιτέρω Μελέτη	84
ΑΝΑΦΟΡΕΣ	85
ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ	88

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

Πίνακας	Σελ
Πίνακας 3.1 Τιμές των μετρικών για τον κόμβο D	30
Πίνακας 3.2 Ενέργειες αλγορίθμων μετά τη λήψη Hello μηνύματος	42
Πίνακας 3.3 Ενέργειες αλγορίθμων μετά τη λήψη TC μηνύματος	43
Πίνακας 3.4 Ενέργειες αλγορίθμων μετά τη λήψη Encounter Vector	43
Πίνακας 3.5 Ενέργειες αλγορίθμων μετά τη λήψη Summary Vector	44
Πίνακας 3.6 Ενέργειες αλγορίθμων μετά τη λήψη Message Request	44
Πίνακας 3.7 Ενέργειες αλγορίθμων μετά τη λήψη μηνύματος δεδομένων	45
Πίνακας 4.1 Παράμετροι δημιουργίας σεναρίων κίνησης	50
Πίνακας 4.2 Ποσοστό επιτυχούς παράδοσης και Μέση καθυστέρηση για όλα τα σενάκια κίνησης	58

ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ

Σχήμα	Σελ
Σχήμα 1.1 Multi-hop δρομολόγηση (E.M. Daly, M. Haahr, [5])	2
Σχήμα 1.2 Εξέλιξη ενός δικτύου στο χρόνο (Zhenseng Zhang, [21])	4
Σχήμα 1.3 Η κινητικότητα σχέση με την πυκνότητα των κόμβων (Σχήμα παραφρασμένο WHITBECK, J.α σε and CONAN, V. [19])	6
Σχήμα 3.1 Παράδειγμα partitioned δικτύου	29
Σχήμα 3.2 Παράδειγμα partitioned δικτύου	34
Σχήμα 3.3 Ανακάλυψη γειτόνων με τη βοήθεια Hello μηνυμάτων	36
Σχήμα 3.4 Επιλογή MPR κόμβων	36
Σχήμα 3.5 Παράδειγμα partitioned δικτύου	41
Σχήμα 4.1 Ο κοινωνικός γράφος του σεναρίου κίνησης Α	51
Σχήμα 4.2 Ο κοινωνικός γράφος του σεναρίου κίνησης Β	51
Σχήμα 4.3 Αριθμός μοναδικών επαφών ανά κόμβο (Σενάριο κίνησης Α)	52
Σχήμα 4.4 Αριθμός μοναδικών επαφών ανά κόμβο (Σενάριο κίνησης Β)	52
Σχήμα 4.5 Συνολικός αριθμός επαφών ανά κόμβο (Σενάριο κίνησης Α)	53
Σχήμα 4.6 Συνολικός αριθμός επαφών ανά κόμβο (Σενάριο κίνησης Β)	53
Σχήμα 4.7 Ο κοινωνικός γράφος του σεναρίου κίνησης Γ	54
Σχήμα 4.8 Ο κοινωνικός γράφος του σεναρίου κίνησης Δ	54
Σχήμα 4.9 Ο κοινωνικός γράφος του σεναρίου κίνησης Ε	55
Σχήμα 4.10 Ο κοινωνικός γράφος του σεναρίου κίνησης ΣΤ	56
Σχήμα 4.11 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Α)	59
Σχήμα 4.12 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Α)	60
Σχήμα 4.13 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Α)	62
Σχήμα 4.14 Αριθμός προωθήσεων (Σενάριο κίνησης Α)	62

Σχήμα 4.15 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Α)	63
Σχήμα 4.16 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Β)	64
Σχήμα 4.17 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Β)	64
Σχήμα 4.18 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Β)	65
Σχήμα 4.19 Αριθμός προωθήσεων (Σενάριο κίνησης Β)	66
Σχήμα 4.20 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Β)	67
Σχήμα 4.21 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Γ)	68
Σχήμα 4.22 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Γ)	69
Σχήμα 4.23 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Γ)	69
Σχήμα 4.24 Αριθμός προωθήσεων (Σενάριο κίνησης Γ)	70
Σχήμα 4.25 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Γ)	71
Σχήμα 4.26 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Δ)	72
Σχήμα 4.27 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Δ)	72
Σχήμα 4.28 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Δ)	73
Σχήμα 4.29 Αριθμός προωθήσεων (Σενάριο κίνησης Δ)	74
Σχήμα 4.30 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Δ)	75
Σχήμα 4.31 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Ε)	76
Σχήμα 4.32 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Ε)	76
Σχήμα 4.33 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Ε)	77
Σχήμα 4.34 Αριθμός προωθήσεων (Σενάριο κίνησης Ε)	77
Σχήμα 4.35 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Ε)	78
Σχήμα 4.36 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης ΣΤ)	79
Σχήμα 4.37 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης ΣΤ)	79

Σχήμα 4.38 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης ΣΤ)	80
Σχήμα 4.39 Αριθμός προωθήσεων (Σενάριο κίνησης ΣΤ)	81
Σχήμα 4.40 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης ΣΤ)	81

ΠΕΡΙΛΗΨΗ

Αναστασία Κόντου του Γεωργίου και της Θεοδώρας, MSc. Τμήμα Μηχανικών Η/Υ και Πληροφορικής. Πανεπιστήμιο Ιωαννίνων. Φεβρουάριος, 2014. Δρομολόγηση Ανεκτική στην Καθυστερήση σε Δίκτυα με Μεικτά Χαρακτηριστικά Συνδεσιμότητας. Επιβλέπωντας: Ευάγγελος Παπαπέτρου

Αντικείμενο μελέτης της παρούσας διατριβής είναι οι αρχές δρομολόγησης σε μία κατηγορία κινητών ασύρματων δικτύων όπου οι κόμβοι του δικτύου σχηματίζουν ομάδες. Στα δίκτυα αυτά παρατηρείται περιορισμένη συνδεσιμότητα όχι μόνο μεταξύ μεμονωμένων κόμβων του δικτύου αλλά και μεταξύ ομάδων κόμβων. Αυτού του είδους τα χαρακτηριστικά συνδεσιμότητας οφείλονται στο γεγονός ότι η κίνηση των κόμβων σχετίζεται με την κίνηση των ανθρώπων που μεταφέρουν τις ασύρματες συσκευές. Η κοινωνική συμπεριφορά των ανθρώπων συνήθως τους κάνει να κινούνται σε ομάδες και μεταξύ ομάδων και όχι μεμονωμένα.

Το μεγαλύτερο μέρος των αλγορίθμων δρομολόγησης που έχουν προταθεί, απευθύνονται είτε σε MANET, είτε σε ομορτυνιστικά δίκτυα. Στις περιπτώσεις, όμως, δικτύων που μελετάμε οι αλγόριθμοι αυτοί δεν μπορούν να λειτουργήσουν ικανοποιητικά. Η λύση βρίσκεται στους υβριδικούς αλγορίθμους που χρησιμοποιούν κατά περίπτωση MANET και ομορτυνιστικούς αλγορίθμους. Οι αλγόριθμοι αυτοί εκμεταλλεύονται τις περιπτώσεις όπου υπάρχει συνδεσιμότητα μεταξύ των κόμβων εντός των ομάδων ενώ ταυτόχρονα να μπορούν να ανεχτούν την έλλειψη συνδεσιμότητας μεταξύ αυτών.

Στην διατριβή αυτή προτείνουμε έναν νέο υβριδικό αλγόριθμο, τον SASI, που συνδυάζει τις αρχές της σύγχρονης δρομολόγησης από τα δίκτυα MANET με τις

αρχές της ασύγχρονης δρομολόγησης από τα ομορτουριστικά δίκτυα. Σύμφωνα με τον αλγόριθμο, όταν υπάρχει συνδεσιμότητα μεταξύ μιας ομάδας κόμβων τα δεδομένα δρομολογούνται με τις αρχές της σύγχρονης δρομολόγησης ενώ όταν δεν υπάρχει, προωθούνται χρησιμοποιώντας τις αρχές της ασύγχρονης δρομολόγησης. Για τη σύγχρονη δρομολόγηση χρησιμοποιείται το πρωτόκολλο δρομολόγησης OLSR. Δεδομένου ότι η κίνηση των κόμβων εξαρτάται από την κοινωνική συμπεριφορά των ανθρώπων που κουβαλούν τις ασύρματες συσκευές, επιλέχθηκε για την ασύγχρονη λειτουργία ένα πρωτόκολλο κοινωνικής δικτύωσης, ο Simbet. Ωστόσο, για να μεγιστοποιηθεί η απόδοση αυτής της προσέγγισης, διαδίδουμε τις μετρικές κοινωνικής δικτύωσης της ασύγχρονης δρομολόγησης μέσω των μηχανισμών του OLSR.

Η νέα προσέγγιση που παρουσιάζεται απευθύνεται σε δίκτυα που παρουσιάζουν κατά περίπτωση χαρακτηριστικά MANET και ομορτουριστικών δικτύων, καλύπτοντας το κενό που δημιουργείται μεταξύ τους. Τα πειραματικά αποτελέσματα αποδεικνύουν ότι ο προτεινόμενος αλγόριθμος SASI, παρουσιάζει καλύτερη απόδοση σε σχέση με άλλους υβριδικούς αλγόριθμους που έχουν προταθεί μέχρι σήμερα.

EXTENDED ABSTRACT IN ENGLISH

Kontou, Anastasia, G. MSc, Department of Computer Science and Engineering, University of Ioannina, Greece. February, 2014. Delay Tolerant Routing in opportunistic wireless networks with mixed features of connectivity. Thesis Supervisor: Evaggelos Papapetrou.

The subject of this thesis is the routing problem in a special category of mobile ad hoc networks. Mobile devices consisting the network are carried by humans, so nodes' movement depends on human decisions and social behavior. Usually humans move in groups and between groups rather than individual, so in real network nodes form connected groups of nodes.

Most of the routing protocols that have been proposed are either for MANET or opportunistic networks. An abundance of routing protocols have been proposed for MANET, that implicitly assume that the network is connected and there is a contemporaneous end-to-end path between any source and destination pair. However, in a physical ad hoc network, the assumption that there is a contemporaneous end-to-end path between any source and destination pair may not be true. The type of network that nodes have occasional connectivity and the network behavior is random and not known is called opportunistic network. The routing protocols have been proposed for opportunistic networks are suitable for sparse networks and address the issues of eventual connectivity by the use of a store and forward mechanism [21].

In dense networks, conventional MANET protocols start to break down under high mobility even if the network is almost always fully connected. DTNs/opportunistic

protocols on the other hand, can handle high mobility regardless of the density of the network. However by narrowly focusing on per-encounter events, they ignore a lot of available information. The hybrid DTN-MANET approach is the one that aims at filling the gap for efficient routing in highly connected and highly mobile networks such as inside concentration points [19].

In this thesis, we propose a new hybrid routing protocol, SASI which combines synchronous routing principles with asynchronous routing. The core idea of SASI is to use a synchronous MANET routing protocol when connectivity is present between a group of nodes while using asynchronous routing when no end-to-end path between source and destination exists. For the synchronous routing we use OLSR protocol while asynchronous routing is based on social-based routing algorithm, Simbet. OLSR is used not only to deliver data when possible but also to spread information about nodes' social relationships. If there isn't connectivity between source and destination, SASI uses metrics based on social network analysis techniques to forward packets. We have demonstrated through simulation that SASI has better outcome concerned delivering packets compared to other hybrid protocols.

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

- 1.1 Κινητά Ασύρματα Δίκτυα
 - 1.2 Αντικείμενο της Διατριβής
 - 1.3 Δομή της Διατριβής
-

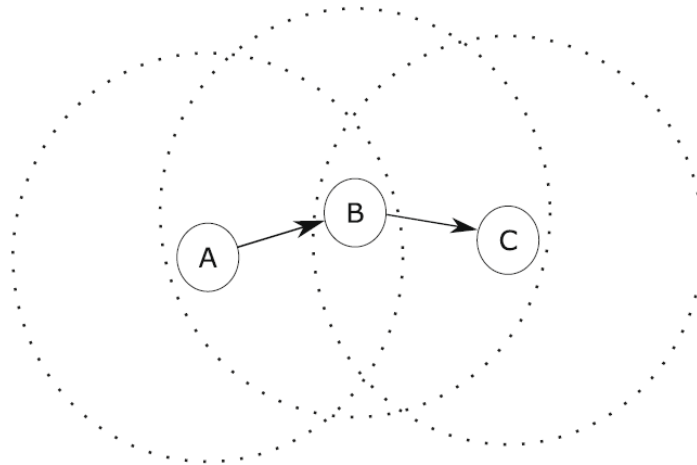
1.1. Κινητά Ασύρματα Δίκτυα

Ένα ασύρματο δίκτυο, είναι ένα δίκτυο στο οποίο οι κόμβοι συνδέονται μεταξύ τους ασύρματα. Αν οι κόμβοι αυτοί έχουν τη δυνατότητα φυσικής κίνησης στο χώρο, τότε έχουμε ένα κινητό ασύρματο δίκτυο. Τα κινητά δίκτυα χωρίζονται σε δύο κατηγορίες: τα ασύρματα δίκτυα σταθερής υποδομής και τα ad hoc δίκτυα.

Στα ασύρματα δίκτυα σταθερής υποδομής η γεωγραφική περιοχή που καλύπτει το δίκτυο μοιράζεται στους σταθμούς βάσης. Οι σταθμοί βάσης συνδέονται μεταξύ τους σε ένα δίκτυο κορμού. Ο κάθε κινητός σταθμός μπορεί να επικοινωνήσει μόνο με τον σταθμό βάσης της περιοχής όπου ανήκει και η επικοινωνία μεταξύ δύο κινητών κόμβων εξυπηρετείται μέσω των σταθμών βάσης.

Αντίθετα, τα κινητά ad hoc δίκτυα (MANETs) χαρακτηρίζονται από έλλειψη σταθερής υποδομής και οι κόμβοι λειτουργούν ως δρομολογητές που συνδέονται μεταξύ τους μέσω ασύρματων ζεύξεων οι οποίες προκύπτουν από αυθαίρετη τοπολογία. Οι συνδέσεις πραγματοποιούνται δια μέσου πολλών κόμβων. Κάθε κόμβος του δικτύου έχει τη δυνατότητα να λάβει ή να προωθήσει δεδομένα σε άλλους κόμβους, εφόσον βρίσκονται εντός της ακτίνας επικοινωνίας του. Η επικοινωνία σε απόσταση μεγαλύτερη από αυτή την ακτίνα γίνεται χρησιμοποιώντας τους

ενδιάμεσους κόμβους για την αναμετάδοση των δεδομένων. Στο παράδειγμα του σχήματος, ο κόμβος A δεν βρίσκεται στην ακτίνα επικοινωνίας του C, αλλά μπορεί να χρησιμοποιήσει σαν μεσολαβητή τον ενδιάμεσο κόμβο B [5].



Σχήμα 1.1 Multi-hop δρομολόγηση (E.M. Daly, M. Haahr, [5])

Οι κόμβοι όμως, αυτοί έχουν τη δυνατότητα να εισέρχονται ή να αποχωρούν από το δίκτυο οποιαδήποτε στιγμή και από οποιοδήποτε σημείο του. Επομένως, η τοπολογία του δικτύου μεταβάλλεται δυναμικά και απρόβλεπτα, καθιστώντας την δρομολόγηση ιδιαίτερα απαιτητική. Οι αλγόριθμοι δρομολόγησης που έχουν επικρατήσει στα ενσύρματα δίκτυα θεωρούν ότι υπάρχει συνεχής συνδεσιμότητα, πολύ μικρό ποσοστό απώλειας πακέτων και σχετικά χαμηλή καθυστέρηση διάδοσης. Αντίθετα στα κινητά ασύρματα δίκτυα, οι αλγόριθμοι δρομολόγησης έχουν να αντιμετωπίσουν τις συνεχείς μεταβολές της τοπολογίας του δικτύου, την πυκνότητα των κόμβων και την περιορισμένη διαθέσιμη ενέργεια.

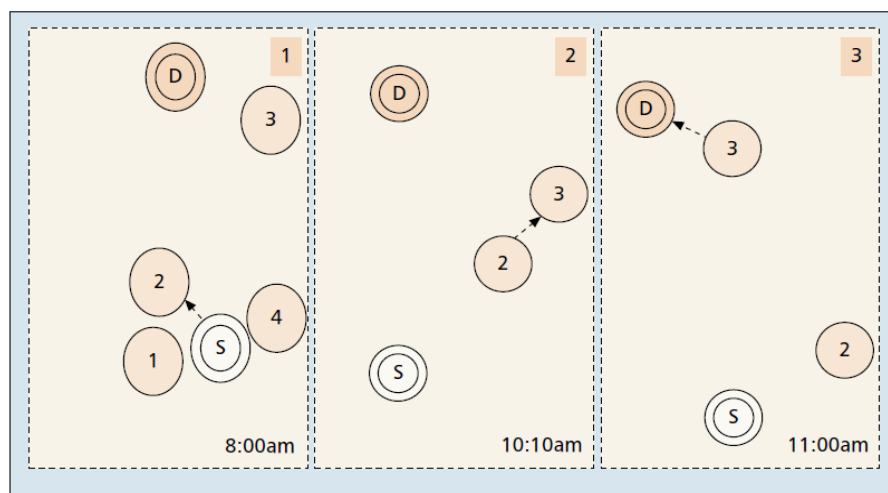
Στη βιβλιογραφία διακρίνονται δύο βασικές κατηγορίες αλγορίθμων δρομολόγησης για MANET δίκτυα, ανάλογα με τα χαρακτηριστικά τους. Η πρώτη κατηγορία είναι οι περιοδικοί αλγόριθμοι ή βασιζόμενοι σε πίνακες (proactive or table-driven). Οι αλγόριθμοι αυτοί βασίζονται στις αρχές των παραδοσιακών αλγορίθμων για ενσύρματα δίκτυα. Το χαρακτηριστικό τους είναι ότι χρησιμοποιούν μία περιοδική διαδικασία ανταλλαγής ενημερώσεων που επιτρέπουν στους κόμβους να γνωρίζουν για τις αλλαγές στην τοπολογία του δικτύου. Κάθε κόμβος διατηρεί έναν πίνακα

δρομολόγησης, με διαδρομές για κάθε άλλο κόμβο του δικτύου, που ανανεώνεται χρησιμοποιώντας τις πληροφορίες που παρέχονται από τις περιοδικές ενημερώσεις. Από την άλλη πλευρά, οι αλγόριθμοι δρομολόγησης που ανήκουν στην δεύτερη κατηγορία και ονομάζονται αιτούμενης ανανέωσης (on-demand), δεν διατηρούν κάποια περιοδική διαδικασία. Αντίθετα, ανακαλύπτουν μία διαδρομή μόνο όταν υπάρχει ανάγκη για την μετάδοση ενός πακέτου δεδομένων και δεν υπάρχει γνωστή διαδρομή για τον προορισμό. Για την ανακάλυψη της διαδρομής απαιτείται μία διαδικασία ερωτήσεων μέχρι να εντοπιστεί ο προορισμός και να δημιουργηθεί μία απάντηση. Αφού βρεθεί, η διαδρομή διατηρείται μέχρι είτε ο προορισμός να καταστεί απρόσιτος είτε να μην χρησιμοποιείται πλέον.

Και οι δύο κατηγορίες αλγορίθμων δρομολόγησης για MANET που αναφέρθηκαν, υποθέτουν ότι υπάρχει συνεχής συνδεσιμότητα στο δίκτυο. Οι αλγόριθμοι που έχουν προταθεί για MANET δίκτυα υποθέτουν ότι τη στιγμή της δρομολόγησης υπάρχει διαθέσιμη από άκρο σε άκρο διαδρομή μεταξύ αποστολέα και παραλήπτη, είτε ενός είτε πολλών αλμάτων. Αν δεν υπάρχει, τότε αποτυγχάνουν να δρομολογήσουν το πακέτο δεδομένων. Στα πραγματικά δίκτυα όμως, μπορεί σε κάποιο στιγμιότυπο του δικτύου να μην υπάρχει πλήρες μονοπάτι από έναν κόμβο προς κάποιον άλλο, όμως τμήματα του μονοπατιού μπορεί να εμφανίζονται σε διαφορετικές χρονικές στιγμές. Αυτή η κατηγορία δικτύων ονομάζεται DTN (δίκτυα ανεκτικά σε καθυστερήσεις). Όταν δε η κίνηση των κόμβων είναι εντελώς τυχαία τότε γίνεται λόγος για μία ειδική κατηγορία DTN, τα opportunistic δίκτυα. Στα opportunistic δίκτυα οι κόμβοι προωθούν τα δεδομένα σε ενδιάμεσους κόμβους προσπαθώντας να προσεγγίσουν τον προορισμό, χρησιμοποιώντας τη συνδεσιμότητα που είναι διαθέσιμη κάθε στιγμή σε κάποιο υπομήμα του δικτύου. Όπως είναι αναμενόμενο, οι καθυστερήσεις στη μετάδοση των δεδομένων λόγω της απουσίας συνδεσιμότητας αποτελούν συχνό φαινόμενο και αυτός είναι ο λόγος που τους δόθηκε η ονομασία δίκτυα ανεκτικά σε καθυστερήσεις. Συχνά στην βιβλιογραφία ο όρος DTN χρησιμοποιείται για να περιγράψει τα opportunistic δίκτυα γι' αυτό και στο υπόλοιπο της παρούσας εργασίας θεωρούμε ότι οι όροι ταυτίζονται.

Για να μπορέσουν να ανταποκριθούν στις ιδιαίτερες απαιτήσεις των δικτύων αυτών, οι αλγόριθμοι δρομολόγησης για opportunistic δίκτυα εισάγουν την τεχνική “store-

carry-forward". Σύμφωνα με αυτή, όταν ένας κόμβος θέλει να προωθήσει ένα πακέτο δεδομένων αλλά το επόμενο άλμα δεν είναι διαθέσιμο, ο κόμβος αποθηκεύει το πακέτο μέχρι να βρεθεί η ευκαιρία για την προώθησή του. Στο παράδειγμα του σχήματος δεν υπάρχει μονοπάτι από τον κόμβο S στον κόμβο D σε καμία χρονική στιγμή αλλά το πακέτο μπορεί να παραδοθεί μέσω των ενδιάμεσων κόμβων (ο κόμβος S στέλνει το πακέτο στον κόμβο 2, ο οποίος το προωθεί στον κόμβο 3 και εκείνος με τη σειρά του στον D).



Σχήμα 1.2 Εξέλιξη ενός δικτύου στο χρόνο (Zhenseng Zhang, [21])

Για να καταστεί αποδοτική η δρομολόγηση με αυτόν τον τρόπο, ο αλγόριθμος δρομολόγησης θα πρέπει να μπορεί να καθορίσει την καλύτερη χρονική στιγμή ώστε να προωθήσει το πακέτο στον καλύτερο κόμβο. Όταν ένα πακέτο δεδομένων δεν μπορεί να παραδοθεί στον παραλήπτη λόγω διάσπασης του δικτύου, οι καλύτεροι μεταφορείς του είναι οι κόμβοι που έχουν τη μεγαλύτερη πιθανότητα να το παραδώσουν επιτυχώς. Αν οι κόμβοι δεν έχουν πληροφορίες σχετικά με την κατάσταση του δικτύου, τότε αυτό που μπορούν να κάνουν είναι να προωθούν με τυχαίο τρόπο πακέτα στους γείτονές τους. Οι αλγόριθμοι δρομολόγησης που ανήκουν σε αυτή την κατηγορία ονομάζονται αλγόριθμοι βασισμένοι στην πλημμύρα. Αν όμως, οι κόμβοι μπορούν να εκτιμήσουν την πιθανότητα των γειτόνων να προωθήσουν επιτυχώς το πακέτο στον παραλήπτη, μπορούν να αποφασίσουν σε ποιους κόμβους να το προωθήσουν ή αν θα πρέπει να το αποθηκεύσουν και να περιμένουν μία πιο κατάλληλη στιγμή. Οι αλγόριθμοι αυτοί ονομάζονται estimation-

based ή history-based γιατί βασίζονται στο ιστορικό των επαφών των χρηστών προκειμένου να υπολογίσουν τις πιθανότητες επιτυχούς παράδοσης. Τέλος, υπάρχουν αλγόριθμοι που για να καθορίσουν το καλύτερο επόμενο άλμα του πακέτου δεδομένων στηρίζονται στην ανάλυση των κοινωνικών σχέσεων των χρηστών. Οι αλγόριθμοι αυτοί ονομάζονται αλγόριθμοι κοινωνικής δικτύωσης και χρησιμοποιούν μετρικές ώστε να αξιολογήσουν τη σημαντικότητα των κόμβων [21].

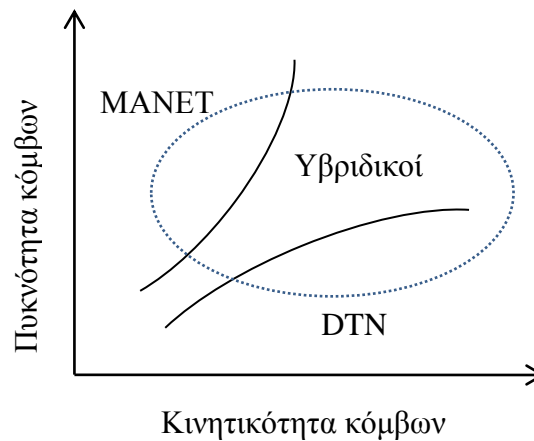
1.2. Αντικείμενο της Διατριβής

Όπως αναφέραμε, στα κινητά ασύρματα δίκτυα οι κόμβοι κινούνται σε περιοχές χωρίς γεωγραφικούς περιορισμούς με αποτέλεσμα να μεταβάλλεται συνεχώς η τοπολογία του δικτύου. Οι ασύρματες συσκευές που απαρτίζουν το δίκτυο συνήθως ανήκουν σε ανθρώπους άρα η κίνηση των συσκευών εξαρτάται από τις ανθρώπινες αποφάσεις και κοινωνικές συμπεριφορές. Είναι συνηθισμένο οι άνθρωποι να κινούνται σε ομάδες και μεταξύ ομάδων και όχι μεμονωμένα [13]. Επομένως, στα πραγματικά δίκτυα παρατηρούνται περιοχές όπου υπάρχει συνδεσιμότητα εντός μιας συγκεκριμένης περιοχής αλλά όχι στο σύνολο του δικτύου. Οι περιοχές αυτές ονομάζονται σημεία συγκέντρωσης (concentration points).

Οι αλγόριθμοι δρομολόγησης που έχουν προταθεί για τα MANET υποθέτουν ότι υπάρχει συνεχής συνδεσιμότητα στο δίκτυο και η δρομολόγηση γίνεται μέσω ενός από άκρη σε άκρη μονοπατιού μεταξύ πηγής και προορισμού. Το κοινό χαρακτηριστικό των αλγορίθμων αυτών είναι ότι υποθέτουν ένα πυκνό δίκτυο, με τους κόμβους να έχουν μία μέτρια κινητικότητα ώστε να μπορούν να διασφαλίσουν ότι υπάρχει συνεχής συνδεσιμότητα. Τα χαρακτηριστικά αυτά είναι απαραίτητα για οποιονδήποτε MANET αλγόριθμο. Όσο όμως μειώνεται η πυκνότητα των κόμβων, τόσο η συνδεσιμότητα τείνει να εξαφανίζεται. Τότε οι κόμβοι έχουν πολύ λίγους ή μπορεί και καθόλου κόμβους στην ακτίνα επικοινωνίας τους. Αυξάνοντας όμως την κινητικότητα οι κόμβοι συναντούν νέους γείτονες αξιοποιώντας ευκαιρίες προώθησης των μηνυμάτων. Τότε, αναφερόμαστε στα DTN δίκτυα [19].

Στις περιπτώσεις, όμως, ασύρματων δικτύων που παρουσιάζουν περιοχές συγκέντρωσης κόμβων, παρατηρείται το φαινόμενο να υπάρχει αυξημένη πυκνότητα

κόμβων εντός των περιοχών ενώ μεταξύ αυτών να υπάρχει περιορισμένη συνδεσιμότητα. Στις περιπτώσεις αυτές, ούτε οι αλγόριθμοι που έχουν προταθεί για MANET αλλά ούτε και αυτοί για opportunistic δίκτυα μπορούν να προσφέρουν ικανοποιητικά αποτελέσματα.



Σχήμα 1.3 Η κινητικότητα σχέση με την πυκνότητα των κόμβων (Σχήμα παραφρασμένο Whitbeck, J. and Conan, V. 19])

Προς αντιμετώπιση του προβλήματος αυτού έχουν προταθεί διάφοροι υβριδικοί αλγόριθμοι που συνδυάζουν στοιχεία και από τις δύο κατηγορίες και προσπαθούν να εκμεταλλευτούν τα πλεονεκτήματα της κάθε μιας. Επιχειρούν, δηλαδή, να εκμεταλλευτούν τη συνδεσιμότητα που παρατηρείται στα σημεία συγκέντρωσης χρησιμοποιώντας έναν σύγχρονο αλγόριθμο ενώ ταυτόχρονα να μπορούν να ανεχτούν τις διασπάσεις του δικτύου.

Οι υβριδικοί αλγόριθμοι μπορούν να χωριστούν σε δύο μεγάλες κατηγορίες, ανάλογα με τον τρόπο που συλλέγεται πληροφορία σχετικά με την συνδεσιμότητα, σε αιτούμενης ανανέωσης και περιοδικούς. Στους αλγόριθμους που χρησιμοποιούν ως βάση έναν αιτούμενης ανανέωσης (on-demand) αλγόριθμο για MANET [3], [6], [9], [14], παρατηρείται ένα συχνό και σημαντικό πρόβλημα, η εναλλαγή μεταξύ σύγχρονης και ασύγχρονης λειτουργίας. Συνήθως οι αλγόριθμοι αυτοί όταν ανιχνεύσουν ότι δεν υπάρχει συνδεσιμότητα μεταπίπτουν σε ασύγχρονη λειτουργία. Το πρόβλημα προκύπτει κατά την επιλογή του χρόνου ανάκαμψης από αυτήν αφο

εξαρτάται από τις συνθήκες του δικτύου. Τίθενται λοιπόν ερωτήματα σχετικά με το πότε θα πρέπει να γίνει προσπάθεια ανάκαμψης χωρίς να προκαλείται επιβάρυνση στο δίκτυο και ταυτόχρονα να αξιοποιούνται οι ευκαιρίες που προκύπτουν για τη δρομολόγηση των δεδομένων. Από την άλλη πλευρά, οι υβριδικοί αλγόριθμοι που στηρίζονται σε ένα περιοδικής ανανέωσης πρωτόκολλο για MANET [7], [12], [19], δεν αντιμετωπίζουν το πρόβλημα που αναφέρθηκε παραπάνω. Αδυνατούν όμως να προσφέρουν ικανοποιητικά αποτελέσματα στις περιπτώσεις δικτύων που μελετάμε.

Γι' αυτό το λόγο στην παρούσα εργασία παρουσιάζεται ένας νέος υβριδικός αλγόριθμος που συνδυάζει τις αρχές της σύγχρονης δρομολόγησης από τα δίκτυα MANET με τις αρχές της ασύγχρονης δρομολόγησης από τα DTN δίκτυα. Σύμφωνα με τον αλγόριθμο, όταν υπάρχει συνδεσιμότητα μεταξύ μιας ομάδας κόμβων τα δεδομένα δρομολογούνται με τις αρχές της σύγχρονης δρομολόγησης ενώ όταν δεν υπάρχει, προωθούνται χρησιμοποιώντας τις αρχές της ασύγχρονης δρομολόγησης. Για τη σύγχρονη δρομολόγηση χρησιμοποιείται το πρωτόκολλο δρομολόγησης OLSR. Δεδομένου ότι η κίνηση των κόμβων εξαρτάται από την κοινωνική συμπεριφορά των ανθρώπων που κουβαλούν τις ασύρματες συσκευές, επιλέχθηκε για την ασύγχρονη λειτουργία ένα πρωτόκολλο κοινωνικής δικτύωσης, ο Simbet. Ωστόσο, για να μεγιστοποιηθεί η απόδοση αυτής της προσέγγισης, διαδίδουμε τις μετρικές κοινωνικής δικτύωσης της ασύγχρονης δρομολόγησης μέσω των μηχανισμών του OLSR.

Η νέα προσέγγιση που παρουσιάζεται απευθύνεται σε δίκτυα που παρουσιάζουν κατά περίπτωση χαρακτηριστικά MANET και DTN δικτύων, καλύπτοντας το κενό που δημιουργείται μεταξύ τους. Ταυτόχρονα, όμως, μπορεί να ανταποκριθεί αποτελεσματικά σε δίκτυα που ανήκουν αποκλειστικά σε μία από τις δύο κατηγορίες. Η διαφοροποίηση του προτεινόμενου αλγορίθμου σε σχέση με άλλους υβριδικούς αλγορίθμους, έγκειται στο γεγονός ότι εισάγονται για πρώτη φορά στοιχεία κοινωνικής δικτύωσης σε έναν σύγχρονο αλγόριθμο. Η εφαρμογή δε των μετρικών και ο τρόπος υπολογισμού τους, γίνεται αξιοποιώντας πληροφορίες που μπορεί να παρέχει η σύγχρονη δρομολόγηση.

Αναλύοντας τα πειραματικά αποτελέσματα του αλγορίθμου παρατηρείται ότι παρουσιάζει καλύτερη απόδοση σε σχέση τόσο με τους παραδοσιακούς αλγορίθμους όσο και με τους υβριδικούς που έχουν προταθεί μέχρι στιγμής.

1.3. Δομή της Διατριβής

Η δομή της διατριβής έχει την παρακάτω δομή. Στο κεφάλαιο 2 παρουσιάζονται οι σημαντικότεροι αλγόριθμοι δρομολόγησης σε MANET και DTN. Έπειτα ασχολούμαστε με εργασίες που έχουν γίνει στο αντικείμενο που μελετάμε. Στο Κεφάλαιο 3 περιγράφεται το θεωρητικό υπόβαθρο του προτεινόμενου αλγορίθμου και τα προβλήματα που λύνει. Επίσης γίνεται αναλυτική περιγραφή του αλγορίθμου. Στο Κεφάλαιο 4 παρουσιάζεται το περιβάλλον της προσομοίωσης, τα εργαλεία και τα δεδομένα που χρησιμοποιήσαμε για την αξιολόγηση του αλγορίθμου. Επιπρόσθετα, παραθέτονται τα πειραματικά αποτελέσματα και γίνεται ο σχολιασμός τους. Τέλος στο Κεφάλαιο 5 καταγράφονται τα συμπεράσματα που προκύπτουν από την μελέτη.

ΚΕΦΑΛΑΙΟ 2. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΔΡΟΜΟΛΟΓΗΣΗΣ

2.1 Δρομολόγηση στα MANET

2.2 Δρομολόγηση στα DTN

2.3 Υβριδικοί Αλγόριθμοι

2.1. Δρομολόγηση στα MANET

Οι αλγόριθμοι δρομολόγησης των MANETs στηρίχθηκαν στους αλγορίθμους ενσύρματων δικτύων κάνοντας κάποιες απαραίτητες τροποποιήσεις ώστε να μπορούν να ανταποκριθούν στα ιδιαίτερα χαρακτηριστικά τους. Αυτά είναι η απουσία σταθερής υποδομής, οι συνεχείς και απρόβλεπτες αλλαγές στην τοπολογία του δικτύου, το περιορισμένο εύρος ζώνης καθώς και οι αυστηροί περιορισμοί στην κατανάλωση ενέργειας, που καθιστούν δύσκολη και περίπλοκη τη δρομολόγηση των δεδομένων μεταξύ των κόμβων. Στα ενσύρματα αλλά και στα ασύρματα δίκτυα έχουν επικρατήσει αλγόριθμοι συντομότερης διαδρομής που είναι βασισμένοι σε πίνακες δρομολόγησης, οι οποίοι αποθηκεύουν διαδρομές προς όλους τους πιθανούς προορισμούς. Για την ενημέρωση των πινάκων αυτών, οι κόμβοι ανταλλάσσουν ενημερώσεις και επαναυπολογίζουν διαδρομές. Οι αλγόριθμοι αυτοί χωρίζονται σε δύο μεγάλες κατηγορίες: αλγόριθμοι διανυσμάτων απόστασης (distance vector) και καταστάσεων συνδέσμων (link state).

Στους αλγόριθμους διανυσμάτων απόστασης κάθε κόμβος διατηρεί έναν πίνακα δρομολόγησης με εγγραφές που περιέχουν την διεύθυνση του κόμβου προορισμού, τη διεύθυνση του επόμενου άλματος και το κόστος της διαδρομής. Κάθε κόμβος στέλνει

ένα αντίγραφο του πίνακα δρομολόγησής του προς κάθε γείτονά του περιοδικά. Όταν ένας κόμβος δέχεται ένα νέο τέτοιο μήνυμα από έναν από τους γείτονές του, ενημερώνει τον δικό του πίνακα δρομολόγησης και αν υπάρχουν αλλαγές αποστέλλει με τη σειρά του τις ενημερωμένες εγγραφές στους γείτονές του. Ο κάθε γείτονας θα επαναλάβει την ίδια διαδικασία. Η διαδικασία λήψης ενημερώσεων από τους γείτονες, υπολογισμού και ανανέωσης των πινάκων δρομολόγησης και πληροφόρησης των γειτόνων για τις αλλαγές, συνεχίζεται έως ότου να μην αποσταλεί κανένα άλλο μήνυμα ενημέρωσης. Σε αυτό το σημείο, ο αλγόριθμος έχει συγκλίνει.

Ένας αλγόριθμος καταστάσεων συνδέσμων υπολογίζει τη διαδρομή ελάχιστου κόστους από τον αποστολέα στον προορισμό γνωρίζοντας την τοπολογία του δικτύου και το κόστος όλων των ζεύξεων. Αυτό γίνεται με την ανταλλαγή μηνυμάτων που περιγράφουν τις συνδέσεις του δικτύου, ποιοι είναι οι γείτονες του αποστολέα, το αντίστοιχο κόστος της ζεύξης, καθώς και έναν ακολουθιακό αριθμό που αυξάνεται σε κάθε ενημέρωση και χρησιμοποιείται για την αποφυγή βρόχων. Τα πακέτα αυτά ονομάζονται Link State packets (LSP) και αποστέλλονται από έναν κόμβο προς όλους τους άλλους κόμβους του δικτύου περιοδικά. Κάθε κόμβος, λοιπόν, έχοντας πλήρη εικόνα του δικτύου, μπορεί να υπολογίσει τη διαδρομή με το μικρότερο κόστος προς κάθε άλλο κόμβο εφαρμόζοντας ένα κεντρικοποιημένο αλγόριθμο, συνήθως τον αλγόριθμο Dijkstra [1].

Ο πρωταρχικός στόχος της δρομολόγησης είναι η εύρεση της συντομότερης διαδρομής για ένα ζεύγος κόμβων, ώστε τα μηνύματα να παραδοθούν αξιόπιστα και έγκαιρα. Η δρομολόγηση θα πρέπει να γίνει με το ελάχιστο κόστος το οποίο εξαρτάται από το κόστος των συνδέσεων της διαδρομής. Έχουν προταθεί πολυάριθμα πρωτόκολλα δρομολόγησης και έχει μελετηθεί η επίδοσή τους σε διάφορα περιβάλλοντα και συνθήκες κίνησης αλλά κανένα δεν είναι αποτελεσματικό σε όλες τις πιθανές συνθήκες. Τα πρωτόκολλα δρομολόγησης των MANETs κατατάσσονται σε τρεις μεγάλες κατηγορίες: περιοδικής ανανέωσης (proactive), αιτούμενης ανανέωσης (reactive ή on-demand) και υβριδικά (hybrid).

Το χαρακτηριστικό των proactive πρωτοκόλλων είναι ότι διατηρούν μια περιοδική διαδικασία που ανιχνεύει πιθανές αλλαγές στην τοπολογία του δικτύου. Το

πλεονέκτημα είναι ότι υπάρχει άμεσα διαθέσιμη διαδρομή προς κάθε πιθανό προορισμό. Αυτό μειώνει σημαντικά την καθυστέρηση στην δρομολόγηση αφού δημιουργούν εκ των προτέρων τις προϋποθέσεις που θα τους επιτρέψουν να κάνουν αργότερα τη δρομολόγηση. Τα proactive πρωτόκολλα μπορούν να ανήκουν στην κατηγορία των διανυσμάτων απόστασης, ή των καταστάσεων συνδέσμων ή να συνδυάζουν και τα δύο. Χαρακτηριστικά πρωτόκολλα της κατηγορίας αυτής είναι ο DSDV (Destination Sequenced Distance Vector) [16] και ο OLSR (Optimized Link State Routing) [8].

Αντίθετα, τα reactive πρωτόκολλα αρχίζουν να επιλύουν το πρόβλημα της δρομολόγησης τη στιγμή που θα εμφανιστεί. Στην προσέγγιση αυτή, δεν υπάρχει ανάγκη αποστολής ενημερώσεων για τις αλλαγές στην τοπολογία του δικτύου. Υπολογίζουν μία διαδρομή πριν τη μετάδοση των δεδομένων. Συνήθως χαρακτηρίζονται από τρεις φάσεις: ανακάλυψη διαδρομής, διατήρηση διαδρομής και διαγραφή διαδρομής. Η ανακάλυψη της διαδρομής ενεργοποιείται όταν υπάρχει ανάγκη για την μετάδοση ενός πακέτου δεδομένων και δεν υπάρχει γνωστή διαδρομή για τον προορισμό. Τότε ξεκινά μία διαδικασία ερωτήσεων μέχρι να εντοπιστεί ο προορισμός και να δημιουργηθεί μία απάντηση. Οι διαδρομές που ανακαλύπτονται διατηρούνται μέχρι να μην χρησιμοποιούνται πλέον ή να διαγραφούν.

Για την προώθηση των πακέτων χρησιμοποιούνται δύο τεχνικές, η δρομολόγηση πηγής (source routing), τεχνική κατά την οποία ο αποστολέας εισάγει όλη τη διαδρομή στο πακέτο δεδομένων, και hop by hop, όπου αντίθετα οι κόμβοι διατηρούν πίνακες δρομολόγησης βάση των οποίων προωθούν το πακέτο από το ένα άλμα στο επόμενο. Χαρακτηριστικά πρωτόκολλα της κατηγορίας είναι το DSR (Dynamic Source Routing) [22] και το AODV (Ad Hoc On-demand Distance Vector) [15].

Τα proactive πρωτόκολλα μπορούν να χρησιμοποιηθούν αποτελεσματικά σε δίκτυα μικρού μεγέθους ή σε δίκτυα με χαμηλή κινητικότητα κόμβων. Επιτυγχάνουν μικρότερη επιβάρυνση στο δίκτυο κατά της φάση της αναζήτησης διαδρομών, εφόσον η πληροφορία υπάρχει ήδη στους πίνακες δρομολόγησης, αλλά επιβαρύνουν το δίκτυο με τα μηνύματα ενημερώσεων που απαιτούνται για τον υπολογισμό των πινάκων δρομολόγησης. Αντίθετα τα reactive πρωτόκολλα δημιουργούν μεγάλη

επιβάρυνση κατά την αναζήτηση αλλά αποφεύγουν την άσκοπη επιβάρυνση του δικτύου ότι δεν υπάρχει κίνηση δεδομένων. Έτσι λοιπόν, είναι περισσότερο αποτελεσματικοί για δίκτυα με μεγαλύτερη κινητικότητα. Η ανάγκη για ικανοποίηση αντικρουόμενων απαιτήσεων οδήγησε στα υβριδικά πρωτόκολλα που συνδυάζουν χαρακτηριστικά και των δύο κατηγοριών. Ένα τέτοιο πρωτόκολλο είναι το ZRP (Zone Routing Protocol) [20].

2.1.1. Destination- Sequenced Distance-Vector- DSDV

Ο αλγόριθμος DSDV βασίζεται στις τεχνικές των παραδοσιακών πρωτοκόλλων διανυσμάτων απόστασης, όπου κάθε κόμβος διατηρεί μία διαδρομή προς κάθε άλλο κόμβο του δικτύου. Ο πίνακας δρομολόγησης περιέχει για κάθε εγγραφή την διεύθυνση του κόμβου προορισμού, τη διεύθυνση του επόμενου άλματος, τον αριθμό αλμάτων που απαιτούνται μέχρι τον προορισμό και έναν αριθμό ακολουθίας. Ο αριθμός ακολουθίας που χαρακτηρίζει κάθε διαδρομή, χρησιμοποιείται για την αποφυγή δημιουργίας βρόχων και δείχνει πόσο πρόσφατη είναι η συγκεκριμένη διαδρομή.

Κάθε κόμβος αποστέλλει περιοδικά στους γείτονές του τον πίνακα δρομολόγησής του στη μορφή «διεύθυνση προορισμού, αριθμό αλμάτων, αριθμός ακολουθίας». Όταν ένας κόμβος λάβει την πληροφορία αυτή τη χρησιμοποιεί για να ανανεώσει τον δικό του πίνακα δρομολόγησης. Οι κόμβοι επιλέγουν πάντοτε τις διαδρομές με το μεγαλύτερο ακολουθιακό αριθμό. Σε περίπτωση που δύο διαδρομές έχουν τον ίδιο ακολουθιακό αριθμό επιλέγεται αυτή με την καλύτερη μετρική (μικρότερο αριθμό αλμάτων). Ένας κόμβος αυξάνει τον αριθμό ακολουθίας του όποτε υπάρχει αλλαγή στην τοπική γειτονιά του. Χρησιμοποιείται περιττός αριθμός ακολουθίας και μετρική ίση με άπειρο όταν ανιχνεύεται απώλεια σύνδεσης, και άρτιος αριθμός για ενημέρωση της διαδρομής από τον ίδιο τον κόμβο. Με αυτόν τον τρόπο γίνεται διάκριση μεταξύ παλαιών και νέων διαδρομών και αποφεύγεται η δημιουργία βρόχων δρομολόγησης. Ακόμη, για την αποφυγή άσκοπων ενημερώσεων, το πρωτόκολλο εισάγει κάποια χρονικά όρια μεταξύ ενός συμβάντος, όπως η απώλεια μία σύνδεσης, και της αποστολής μιας ενημέρωσης για αυτό.

Εκτός από την περιοδική αποστολή των μηνυμάτων ενημερώσεων ένας κόμβος μπορεί να στείλει και εξαναγκασμένες ενημερώσεις ή ενημερώσεις οδηγούμενες από γεγονότα. Τα γεγονότα αυτά μπορούν να είναι η απώλεια μιας ζεύξης ή η ανίχνευση μίας καινούριας. Για τη μείωση του όγκου της πληροφορίας στα πακέτα ενημέρωσης, υπάρχουν δύο τύποι ενημερώσεων: full dump και incremental. Τα full dump περιέχουν πληροφορίες για όλες τις διαθέσιμες διαδρομές, ενώ τα incremental dump περιέχουν μόνο τις διαδρομές του πίνακα δρομολόγησης που έχουν τροποποιηθεί από την τελευταία πλήρη ενημέρωση.

Η αποτελεσματικότητα του πρωτοκόλλου εξαρτάται από το είδος του δικτύου. Σε μικρού μεγέθους δίκτυα με χαμηλή κινητικότητα επιτυγχάνεται γρήγορη σύγκλιση του αλγορίθμου και υψηλά ποσοστά επιτυχούς παράδοσης. Όσο όμως μεγαλώνει το μέγεθος του δικτύου, τόσο αυξάνεται και το μέγεθος των μηνυμάτων ενημερώσεων προκαλώντας συμφόρηση. Επιπλέον, σε μία συχνά εναλλασσόμενη τοπολογία οι κόμβοι οδηγούνται σε μία συνεχή ανταλλαγή πληροφοριών, πιθανών μη έγκυρων, επιβαρύνοντας το δίκτυο.

2.1.2. Optimized Link State Routing- OLSR

Το πρωτόκολλο OLSR είναι μία βελτιστοποίηση του κλασικού link state πρωτοκόλλου. Όπως έχουμε αναφέρει η λειτουργία ενός link state αλγορίθμου στηρίζεται στα μηνύματα ελέγχου Link State packets (LSP) που περιέχουν πληροφορίες για όλες τις ζεύξεις του δικτύου και διαδίδονται με τη μέθοδο της πλημμύρας σε όλους τους κόμβους του δικτύου. Έπειτα κάθε κόμβος, γνωρίζοντας την πλήρη εικόνα του δικτύου, μπορεί να υπολογίσει μια διαδρομή προς κάθε άλλο κόμβο. Το πρωτόκολλο OLSR μειώνει το μέγεθος των μηνυμάτων αυτών και ελαχιστοποιεί την πλημμύρα των μηνυμάτων ελέγχου άρα και την επιβάρυνση του δικτύου.

Το OLSR χρησιμοποιεί για την διάδοση των Link State packets τους MultiPoint Relay (MPR) κόμβους. Δηλαδή, αντί ένας κόμβος να ανακοινώνει όλες τις ζεύξεις του σε όλους τους κόμβους του δικτύου, ανακοινώνει μόνο τις ζεύξεις του με τους

Multipoint relay selectors γείτονές του. Οι Multipoint Relays είναι οι μόνοι που προωθούν τα μηνύματα ελέγχου, μειώνοντας το φόρτο του δικτύου.

Κάθε κόμβος επιλέγει ένα σύνολο κόμβων μεταξύ των άμεσων γειτόνων του, που ονομάζονται Multipoint relays, έτσι ώστε να καλύπτονται όλοι οι γείτονες σε απόσταση δύο αλμάτων. Αυτοί είναι υπεύθυνοι να προωθούν τα μηνύματα ελέγχου του κόμβου. Γι' αυτό το σκοπό, κάθε κόμβος διατηρεί έναν πίνακα με τους γείτονες που καλούνται MPR selectors του κόμβου. Κάθε μήνυμα που προέρχεται από τους MPR selectors πρέπει να αναμεταδοθεί από τον κόμβο. Κάθε κόμβος αποστέλλει περιοδικά τη λίστα με τους κόμβους που τον έχουν επιλέξει ως MPR. Όταν ένας κόμβος λάβει αυτή τη λίστα με τους MPR selectors, υπολογίζει και ανανεώνει τις διαδρομές για όλους τους γνωστούς προορισμούς επιλέγοντας ως ενδιάμεσους κόμβους τους MPR κόμβους.

Κάθε κόμβος, για να ανακαλύψει τους άμεσα γείτονές του εκπέμπει περιοδικά μηνύματα *Hello* που περιέχουν τους γείτονες που γνωρίζει μέχρι εκείνη τη στιγμή και την κατάσταση της ζεύξης, συμμετρική (bi-directional), ασύμμετρη (uni-directional) ή MPR. Τα μηνύματα αυτά αποστέλλονται μία φορά ανά χρονική περίοδο και απευθύνονται μόνο στους άμεσους γείτονες και δεν προωθούνται περαιτέρω. Με αυτόν τον τρόπο κάθε κόμβος μαθαίνει τη γειτονιά του σε απόσταση δύο αλμάτων (2-hop) και κατασκευάζει έναν «πίνακα γειτόνων» που περιέχει του άμεσους γείτονες και την κατάσταση της ζεύξης που τους ενώνει, καθώς και τους γείτονες δύο αλμάτων. Κάθε εγγραφή του πίνακα γειτόνων περιέχει τη χρονική στιγμή μετά από την οποία η εγγραφή λήγει και δεν θεωρεί πλέον έγκυρη.

Γνωρίζοντας την 2-hop γειτονιά ο κόμβος επιλέγει ανάμεσα στους απευθείας γείτονές του το ελάχιστο δυνατό σύνολο κόμβων που καλύπτουν όλους τους γείτονες σε απόσταση δύο αλμάτων. Το σύνολο αυτών των κόμβων είναι οι MultiPoint relays του κόμβου. Για την αποφυγή προβλημάτων επιλέγονται σαν MPR μόνο οι κόμβοι με συμμετρικές ζεύξεις.

Για την διατήρηση πληροφορίας σχετικά με την τοπολογία του δικτύου, κάθε κόμβος που έχει επιλεγεί ως MPR, εκπέμπει μηνύματα ελέγχου που ονομάζονται Topology

Control (TC) μηνύματα και προωθούνται σε όλο το δίκτυο μέσω των MPR κόμβων. Τα TC μηνύματα αποστέλλονται περιοδικά και περιέχουν τη λίστα των κόμβων που έχουν επιλέξει τον κόμβο που αποστέλλει το μήνυμα ως Multipoint relay, καθώς και ένας ακολουθιακό αριθμό. Τα μηνύματα αυτά προωθούνται σε όλο το δίκτυο μέσω των MPR κόμβων.

Με την πληροφορία που παρέχεται από τα TC, οι κόμβοι διατηρούν ένα πίνακα τοπολογίας με εγγραφές για τους MultiPoint relays των υπόλοιπων κόμβων. Κάθε εγγραφή περιέχει: τη διεύθυνση του προορισμού (τον MPR selector που περιέχεται στο TC), τη διεύθυνση του τελευταίου άλματος (ο αποστολέας του TC) και τον ακολουθιακό αριθμό.

Επιπροσθέτως, κάθε κόμβος διατηρεί έναν πίνακα δρομολόγησης με τις πληροφορίες που αντλεί από τον πίνακα τοπολογίας και τον πίνακα γειτόνων. Οι εγγραφές είναι της μορφής: διεύθυνση προορισμού, επόμενο άλμα, εκτιμώμενη απόσταση και υπάρχει μία εγγραφή για κάθε γνωστό προορισμό. Ο πίνακας δρομολόγησης επαναυπολογίζεται όταν ανιχνευτεί κάποια αλλαγή στη γειτονιά του κόμβου ή στον πίνακα τοπολογίας. Όλα τα πακέτα δεδομένων προωθούνται βάση του πίνακα δρομολόγησης.

Το OLSR είναι ένα κατανεμημένο πρωτόκολλο κατάλληλο για μεγάλα δίκτυα, πυκνά και με μεγάλο φόρτο. Επίσης, είναι αποτελεσματικό σε περιπτώσεις όπου δεν επιτρέπονται μεγάλες καθυστερήσεις στη δρομολόγηση των δεδομένων.

2.2. Δρομολόγηση στα DTN

Το περιβάλλον στο οποίο καλούνται να ανταποκριθούν οι αλγόριθμοι δρομολόγησης σε DTN χαρακτηρίζεται από πολλές προκλήσεις και δυσκολίες. Έχει προταθεί πληθώρα αλγορίθμων οι οποίοι χωρίζονται σε τρεις μεγάλες κατηγορίες: αλγόριθμοι βασισμένοι στην πλημμύρα (flooding-based), αλγόριθμοι βασισμένοι στο ιστορικό (history-based) και αλγόριθμοι κοινωνικής δικτύωσης (social-based).

Οι αλγόριθμοι βασισμένοι στην πλημμύρα προσπαθούν να χρησιμοποιήσουν τις ευκαιριακές συνδέσεις των κόμβων προκειμένου να προωθήσουν τα πακέτα δεδομένων και να μεγιστοποιήσουν την πιθανότητα επιτυχούς παράδοσης. Το πιο χαρακτηριστικό παράδειγμα της κατηγορίας αυτής είναι ο αλγόριθμος Epidemic [18]. Σύμφωνα με τον αλγόριθμο, οι κόμβοι συνεχώς αντιγράφουν και μεταδίδουν μηνύματα σε νέους κόμβους οι οποίοι δεν έχουν αποκτήσει ακόμα κάποιο αντίγραφο του μηνύματος. Μειονέκτημα του αλγορίθμου αυτού είναι ότι κατά την ανταλλαγή των μηνυμάτων δεν λαμβάνεται υπόψη πληροφορία για την επιτυχή παράδοση του μηνύματος από τον νέο κόμβο που παραλαμβάνει το μήνυμα. Δημιουργούνται, λοιπόν, πολλά αντίγραφα που μπορεί να μη βοηθούν στην παράδοση στον τελικό προορισμό με αποτέλεσμα να επιβαρύνεται το δίκτυο και να καταναλώνονται πολύτιμοι πόροι όπως αποθηκευτικός χώρος, επεξεργαστική ισχύς, μπαταρία.

Οι αλγόριθμοι βασισμένοι στο ιστορικό των επαφών στηρίζονται στην υπόθεση ότι οι χρήστες δεν κινούνται και τόσο τυχαία όσο πιστεύεται αλλά ακολουθούν διαδρομές οι οποίες μπορούν προβλεφθούν. Δηλαδή αν ένας κόμβος έχει συναντήσει τον κόμβο-προορισμό πολλές φορές στο παρελθόν είναι πολύ πιθανό να τον συναντήσει και στο μέλλον. Χαρακτηριστικός αλγόριθμος της κατηγορίας αυτής είναι ο PRoPHET (Probabilistic Routing Protocol using History of Encounters and Transitivity) [10], ο οποίος χρησιμοποιεί το ιστορικό επαφών για να υπολογίσει την πιθανότητα επιτυχούς παράδοσης για κάθε γνωστό προορισμό. Όταν δύο κόμβοι συναντηθούν ανταλλάσσουν την πιθανότητα πρόβλεψης επιτυχούς παράδοσης και αναλαμβάνει την προώθηση του μηνύματος ο κόμβος με την μεγαλύτερη πιθανότητα [21].

Στους αλγορίθμους κοινωνικής δικτύωσης γίνεται προσπάθεια εντοπισμού και απεικόνισης των σχέσεων των χρηστών αφού θεωρούν ότι οι σχέσεις αυτές αντικατοπτρίζονται και στο δίκτυο. Για την ανάλυση των κοινωνικών δικτύων χρησιμοποιούνται μετρικές κοινωνικής δικτύωσης, οι οποίες λαμβάνονται υπόψη για την δρομολόγηση. Από τους πιο γνωστούς αλγορίθμους της κατηγορίας αυτής είναι ο Simbet [4].

2.2.1. Μετρικές Κοινωνικής Δικτύωσης

Οι κοινωνική δικτύωση στηρίζεται στις πραγματικές κοινωνικές σχέσεις μεταξύ των χρηστών των ασύρματων συσκευών, για παράδειγμα συνάδελφοι που εργάζονται στον ίδιο χώρο. Οι κοινωνικές αυτές σχέσεις μπορούν να αναπαρασταθούν από έναν κοινωνικό γράφο (social graph), όπου ο κάθε κόμβος του γράφου αναπαριστά ένα χρήστη και οι ακμές την κοινωνική σχέση των χρηστών. Η συσχέτιση δύο χρηστών ονομάζεται επαφή. Επομένως, ο κοινωνικός γράφος αποτελεί μία αναπαράσταση όλων των επαφών μεταξύ των κόμβων του δικτύου.

Το πρόβλημα προκύπτει από το γεγονός ότι δεν συνεπάγονται όλες οι επαφές κοινωνικές σχέσεις αφού κάποιες από αυτές προκύπτουν από τυχαίες συναντήσεις των κόμβων. Γι' αυτό το λόγο για την ανάλυση του κοινωνικού γράφου χρησιμοποιούνται μετρικές κοινωνικής δικτύωσης που αξιολογούν τη σημαντικότητα της επαφής και του κόμβου μέσα στο κοινωνικό δίκτυο. Οι πιο γνωστές μετρικές είναι το centrality και το similarity.

Η έννοια του centrality εκφράζει τη σημαντικότητα ενός κόμβου στον κοινωνικό γράφο. Ένας κόμβος με μεγάλο centrality έχει και μεγάλη συνδεσιμότητα με άλλους κόμβους τους δικτύου και θα μπορούσαμε να τον παρομοιάσουμε με «γέφυρα» μεταξύ διαφορετικών ομάδων κόμβων. Οι τρεις πιο χρησιμοποιούμενες μετρικές του centrality είναι το degree centrality, closeness centrality και betweenness centrality.

- *Degree centrality*: ορίζεται ως το σύνολο των άμεσων συνδέσεων που έχει ένας κόμβος με άλλους κόμβους. Ένας κόμβος με μεγάλη τιμή degree centrality διατηρεί επαφές και πολλούς άλλους κόμβους, δηλαδή θα μπορούσαμε να τον θεωρήσουμε ως δημοφιλή κόμβο. Η τιμή του degree centrality δίνεται από τον τύπο:

$$C_D(p_i) = \sum_{k=1}^N a(p_i, p_k) \quad \text{Εξ. 2.1}$$

όπου $a(p_i, p_k) = 1$ αν υπάρχει άμεση σύνδεση μεταξύ των p_i και p_k . Για τον υπολογισμό του degree centrality απαιτείται μόνο τοπική πληροφορία του κόμβου.

- *Closeness centrality*: εκφράζει το μέσο μήκος των συντομότερων μονοπατιών μεταξύ του κόμβου p_i και όλων των άλλων προσβάσιμων κόμβων του δικτύου. Μπορεί να θεωρηθεί μετρική του πόσο χρόνο χρειάζεται για να μεταδοθεί η πληροφορία από έναν κόμβο προς άλλους κόμβους του δικτύου. Η τιμή του closeness centrality υπολογίζεται από τον τύπο:

$$C_C(p_i) = \frac{N - 1}{\sum_{k=1}^N d(p_i, p_k)} \quad \text{Εξ. 2.2}$$

όπου N ο αριθμός των κόμβων του δικτύου. Για τον υπολογισμό του closeness centrality χρειάζονται πληροφορίες που δεν είναι τοπικά διαθέσιμες στον κόμβο.

- *Betweenness centrality*: προσδιορίζει τη σημασία της θέσης του κόμβου σε σχέση με το υπόλοιπο δίκτυο, δηλαδή το κατά πόσο μπορεί να μεταδοθεί η πληροφορία μέσω του κόμβου αυτού. Η μετρική αυτή εκφράζει σε πόσα συντομότερα μονοπάτια από ένα σημείο του δικτύου προς ένα άλλο συμμετέχει ως ενδιάμεσος ο κόμβος που μας ενδιαφέρει. Η τιμή του betweenness centrality δίνεται από τον τύπο:

$$C_B(p_i) = \sum_{j=1}^N \sum_{k=1}^{j-1} \frac{g_{jk}(p_i)}{g_{jk}} \quad \text{Εξ. 2.3}$$

όπου g_{jk} είναι ο συνολικός αριθμός των συντομότερων μονοπατιών που ενώνουν δύο οποιουσδήποτε κόμβους του δικτύου j και k , και $g_{jk}(p_i)$ είναι το πλήθος των συντομότερων μονοπατιών στα οποία συμπεριλαμβάνεται ο κόμβος p_i . Για τον υπολογισμό του betweenness centrality απαιτούνται πληροφορίες για ολόκληρο τον κοινωνικό γράφο.

Η μετρική similarity εκφράζει το πλήθος των κοινωνικών επαφών που έχει κάποιος κόμβος με κάποιον άλλο κόμβο. Αν υπάρχουν πολλές κοινές επαφές μεταξύ δύο κόμβων σημαίνει ότι έχουν και πολλές πιθανότητες να συναντηθούν. Όσο μεγαλύτερο είναι το similarity ενός κόμβου με τον προορισμό, τόσο περισσότερες πιθανότητες έχει να παραδώσει πακέτα που έχει γι' αυτόν. Η τιμή της μετρικής υπολογίζεται από τον τύπο:

$$P_{(x,y)} = |N_{(x)} \cap N_{(y)}|$$

Εξ. 2.4

όπου $N_{(x)}$ και $N_{(y)}$ είναι το σύνολο των επαφών του x και y αντίστοιχα [4].

2.2.1.1. Ego networks

Για τον υπολογισμό κάποιων από τις μετρικές που αναφέραμε απαιτείται πληροφορία για όλο τον κοινωνικό γράφο, επομένως γίνεται δύσκολος και χρονοβόρος ο υπολογισμός τους όσο αυξάνει το μέγεθος του δικτύου. Γι' αυτό το λόγο χρησιμοποιείται η έννοια του ego network. Το ego network μπορεί να οριστεί ως ένα δίκτυο ενός κόμβου (ego) με τους γειτονικούς του κόμβους (alters) και όλες τις μεταξύ τους συνδέσεις. Η κοινωνική ανάλυση του ego network μπορεί να γίνει τοπικά από κάθε κόμβο χωρίς να είναι απαραίτητη η γνώση όλου του κοινωνικού γράφου.

Το degree centrality μπορεί να υπολογιστεί ως ο αριθμός των επαφών ενώ το closeness centrality δεν έχει νόημα εφόσον όλοι οι κόμβοι του ego network απέχουν απόσταση ενός άλματος από τον ego κόμβο. Η μετρική betweenness centrality είναι αυτή παρουσιάζει ενδιαφέρον και προσδιορίζει πόσο σημαντικός είναι ο ego κόμβος για την επικοινωνία των γειτόνων του. Η διαδικασία υπολογισμού του ego betweenness είναι ίδια με αυτή για τον υπολογισμό του betweenness centrality μόνο που χρησιμοποιείται μόνο τοπική πληροφορία. Στην εργασία του ο Marsden υπολόγισε το ego betweenness και το συνέκρινε με τις μετρικές του Freeman για το sociocentric betweenness και βρέθηκε ότι η κατάταξη των δύο αυτών μετρικών είναι ίδια σε μεγάλο ποσοστό σε διαφορετικά κοινωνικά δίκτυα [11]. Αυτό σημαίνει ότι δύο κόμβοι μπορούν να συγκρίνουν την τιμή των τοπικά υπολογισμένων betweenness. Αυτό έδωσε τη δυνατότητα να χρησιμοποιηθούν οι μετρικές αυτές σε αλγορίθμους ως κριτήριο για αποφάσεις δρομολόγησης [4].

2.2.2. *Ο Αλγόριθμος Simbet*

Ο αλγόριθμος Simbet [4] χρησιμοποιεί τις μετρικές ego betweenness ή απλά betweenness και similarity για να αποφασίσει ποιος κόμβος θα αναλάβει την

προώθηση του πακέτου δεδομένων. Το *betweenness* εκφράζει το πλήθος των συντομότερων μονοπατιών από τον αποστολέα έως τον παραλήπτη στα οποία ανήκει ο κόμβος, ενώ το *similarity* εκφράζει το πλήθος των κοινών επαφών που έχει ένας κόμβος με τον παραλήπτη. Οι μετρικές αυτές συνδυάζονται γραμμικά δίνοντας μία νέα μετρική την *SimbetUtil* ως εξής:

$$SimUtil_A(d) = \frac{Sim_A(d)}{Sim_A(d) + Sim_B(d)} \quad \text{Εξ. 2.5}$$

$$BetUtil_A = \frac{Bet_A}{Bet_A + Bet_B} \quad \text{Εξ. 2.6}$$

$$SimBetUtil_A(d) = a * SimUtil_A(d) + b * BetUtil_A \quad \text{Εξ. 2.7}$$

Αν A και B είναι οι κόμβοι, $Sim_A(d)$ και $Sim_B(d)$ είναι οι τιμές του *similarity* για τον προορισμό d και Bet_A και Bet_B οι τιμές του *betweenness*. Επίσης, πρέπει να ισχύει $a+b=1$, με συνηθισμένες τιμές τις $a=b=0,5$.

Για τη λειτουργία του αλγορίθμου οι κόμβοι εκπέμπουν περιοδικά μηνύματα Hello που χρησιμεύουν στην αναγνώριση των γειτόνων. Επίσης διατηρούν μία λίστα (*encounter vector*) που περιέχει όλους τους κόμβους που έχουν συναντήσει μέχρι στιγμής.

Ο αλγόριθμος ακολουθεί τα παρακάτω βήματα:

- 1) Όταν ένας κόμβος n λάβει ένα Hello μήνυμα από έναν κόμβο m, εάν είναι καινούριος γείτονας, του παραδίδει όσα μηνύματα έχει αποθηκευμένα με προορισμό τον m. Έπειτα ζητά από τον m να του στείλει τη λίστα με τις επαφές του (*encounter vector*).
- 2) Όταν ένα κόμβος n λάβει *encounter vector* από τον κόμβο m, υπολογίζει τις μετρικές *betweenness* και *similarity* χρησιμοποιώντας την πληροφορία που περιέχεται στο *encounter vector* για όλους τους προορισμούς για τους οποίους διαθέτει πακέτο δεδομένων. Στη συνέχεια, ο n στέλνει στον m μία λίστα που

περιέχει όλους τους προορισμούς των πακέτων που διαθέτει και τις μετρικές που υπολόγισε (summary vector).

- 3) Όταν ένας κόμβος n λάβει summary vector από τον κόμβο m , υπολογίζει για κάθε προορισμό που περιέχεται στη λίστα τις δικές του μετρικές betweenness και similarity. Έπειτα υπολογίζει το SimbetUtil δικό του και του m και για κάθε προορισμό για τον οποίο ο n έχει μεγαλύτερη μετρική από τον m , ζητά από τον m να του αποστείλει τα πακέτα (message request).
- 4) Όταν ένας κόμβος n λάβει message request από τον κόμβο m , προωθεί στον m όσα πακέτα έχει αποθηκευμένα για τους προορισμούς που περιέχονται στο message request.

Στόχος, λοιπόν, του αλγορίθμου είναι σε κάθε άλμα να προωθούνται τα πακέτα στον κόμβο που έχει περισσότερες πιθανότητες να παραδώσει το μήνυμα στον προορισμό του, δηλαδή έναν κεντρικός κόμβος με όσους δυνατόν περισσότερους κοινούς γείτονες με τον προορισμό.

2.3. Υβριδικοί Αλγόριθμοι

Οι αλγόριθμοι που έχουν παρουσιαστεί μέχρι στιγμής ανήκουν ή στην κατηγορία των MANETs ή των DTNs. Πολλές φορές όμως στα πραγματικά δίκτυα παρατηρούνται περιοχές όπου υπάρχει συνδεσιμότητα εντός της συγκεκριμένης περιοχής αλλά όχι στο σύνολο του δικτύου. Στις περιπτώσεις αυτές δεν ανταποκρίνονται ικανοποιητικά ούτε οι αλγόριθμοι δρομολόγησης των MANET ούτε εκείνοι των DTN. Γι' αυτό το λόγο έχουν προταθεί αλγόριθμοι που συνδυάζουν στοιχεία και από τις δύο κατηγορίες.

Όπως και στα MANETs έτσι και τους υβριδικούς αλγόριθμους μπορούμε να τους χωρίσουμε σε δύο μεγάλες κατηγορίες: περιοδικής ανανέωσης (proactive) και αιτούμενης ανανέωσης (reactive ή on-demand). Στην πρώτη κατηγορία ανήκουν αλγόριθμοι που στηρίζονται σε έναν proactive MANET αλγόριθμο, δηλαδή διατηρούν μια περιοδική διαδικασία που ανιχνεύει πιθανές αλλαγές στην τοπολογία του δικτύου, ενώ στη δεύτερη αλγόριθμοι που στηρίζονται σε reactive πρωτόκολλα.

Ένα παράδειγμα υβριδικού reactive αλγορίθμου είναι το “Integrating DTN and MANET Routing” [14], το οποίο χρησιμοποιεί σύγχρονη δρομολόγηση όταν είναι διαθέσιμη ενώ ταυτόχρονα προσφέρει ασύγχρονη δρομολόγηση σε επιλεγμένους κόμβους, αφήνοντας την απόφαση της επιλογής ποιος από τους δύο τρόπους θα χρησιμοποιηθεί στην εφαρμογή του χρήστη. Σύμφωνα με το πρωτόκολλο χρησιμοποιείται το route discovery του AODV [15] για να ανακαλυφθεί τυχόν διαθέσιμο από άκρη σε άκρη μονοπάτι και ταυτόχρονα ο κόμβος μπορεί να πληροφορηθεί για τους διαθέσιμους κόμβους που υποστηρίζουν ασύγχρονη λειτουργία και είναι προκαθορισμένοι. Σε περίπτωση που είναι διαθέσιμες και σύγχρονες και ασύγχρονες διαδρομές ο κόμβος θα πρέπει να επιλέξει ποια θα χρησιμοποιήσει. Αυτό ακριβώς είναι το σημαντικότερο μειονέκτημα του αλγορίθμου, ότι η επιλογή μεταξύ των δύο τρόπων δρομολόγησης δεν γίνεται από το πρωτόκολλο αλλά πρέπει η εφαρμογή να πάρει μία απόφαση που εξαρτάται από τις τρέχουσες συνθήκες του δικτύου. Επιπλέον, η δυνατότητα δρομολόγησης με ασύγχρονο τρόπο προσφέρεται μόνο σε συγκεκριμένους προκαθορισμένους κόμβους, κάτι που σημαίνει ότι είναι δυνατό να μην μπορεί να χρησιμοποιηθεί ασύγχρονη δρομολόγηση ακόμη και αν δεν υπάρχει από άκρη σε άκρη μονοπάτι.

Ένα άλλο παράδειγμα πρωτοκόλλου αιτούμενης ανανέωσης είναι το DT-DYMO: Delay-tolerant Dynamic MANET On-demand Routing [9] το οποίο συνδυάζει το πρωτόκολλο DYMO [2], που μοιάζει με τον AODV, με τον αλγόριθμο PROPHET. Χρησιμοποιούνται οι φάσεις ανακάλυψης, διατήρησης και διαγραφή διαδρομής του AODV καθώς και μία πιθανότητα παράδοσης (delivery probability) η οποία ανανεώνεται σε κάθε συνάντηση δύο κόμβων και μειώνεται με την πάροδο του χρόνου. Οι πιθανότητες παράδοσης γίνονται broadcast στο δίκτυο περιοδικά. Ο αποστολέας στέλνει ένα Route Request (RREQ) στο οποίο αναφέρει την ελάχιστη πιθανότητα παράδοσης που θεωρεί αποδεκτή. Στο RREQ απαντάει ή ο προορισμός ή όποιος κόμβος έχει πιθανότητα παράδοσης για τον προορισμό μεγαλύτερο από την τιμή που αναφέρεται στο RREQ. Σε περίπτωση που βρεθεί ο προορισμός, ο αποστολέας προωθεί το μήνυμα απευθείας σε αυτόν. Διαφορετικά, ο αποστολέας επιλέγει είτε τον κόμβο με τη μεγαλύτερη πιθανότητα παράδοσης, είτε έναν τυχαίο κόμβο να στείλει το μήνυμα, είτε μοιράζει το μήνυμα σε πολλούς κόμβους. Η

απόφαση εξαρτάται από τη λειτουργία δρομολόγησης που έχει επιλεγεί κατά στην προσομοίωση.

Το κοινό μειονέκτημα των reactive υβριδικών πρωτοκόλλων είναι η μετάβαση μεταξύ των δύο τρόπων δρομολόγησης. Συνήθως οι αλγόριθμοι μεταβαίνουν από σύγχρονη σε ασύγχρονη δρομολόγηση όταν δεν υπάρχει διαθέσιμη διαδρομή από άκρο σε άκρο, αλλά αντιμετωπίζουν προβλήματα στην ανάκαμψη. Η φύση των reactive πρωτοκόλλων είναι να αρχίζουν να επιλύουν το πρόβλημα της δρομολόγησης τη στιγμή που θα εμφανιστεί. Έτσι, λοιπόν, εγείρονται ερωτήματα σχετικά με το κάθε πότε είναι κατάλληλο να ξεκινήσει η διαδικασία ανακάλυψης μιας από άκρο σε άκρο διαδρομής όταν το πρωτόκολλο λειτουργεί σε ασύγχρονη λειτουργία. Υπό κανονικές συνθήκες η διαδικασία αυτή ενεργοποιείται όταν υπάρχει ανάγκη για την μετάδοση ενός πακέτου δεδομένων και δεν υπάρχει γνωστή διαδρομή για τον προορισμό. Όταν όμως το πακέτο δρομολογείται hop-by-hop σε κανένα από τους μέχρι τώρα προτεινόμενους αλγορίθμους δεν υπάρχει σαφής απάντηση σχετικά με την προσπάθεια μετάβασης από ασύγχρονη σε σύγχρονη λειτουργία. Το δίλλημα που αντιμετωπίζουν είναι ότι αν η προσπάθεια αυτή είναι πολύ συχνή προκαλείται μεγάλη επιβάρυνση στο δίκτυο κατά την αναζήτηση, ενώ αν είναι λιγότερο συχνή είναι πιθανόν να μην αξιοποιηθούν διαδρομές που προκύπτουν.

Το πρόβλημα αυτό μπορεί να αντιμετωπιστεί με τη χρήση proactive πρωτοκόλλων των οποίων το πλεονέκτημα είναι ότι διατηρούν μια περιοδική διαδικασία που ανιχνεύει πιθανές αλλαγές στην τοπολογία του δικτύου. Σε αυτή την περίπτωση όταν το πρωτόκολλο λειτουργεί σε ασύγχρονη λειτουργία μπορεί να ενημερωθεί άμεσα για τυχόν διαθέσιμη από άκρο σε άκρο διαδρομή. Παρακάτω θα παρουσιάσουμε κάποια proactive υβριδικά πρωτόκολλα.

2.3.1. Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks- CAR

Το πρωτόκολλο CAR [12] στηρίζεται στην υπόθεση ότι οι κόμβοι του δικτύου είναι χωρισμένοι σε ομάδες εντός των οποίων υπάρχει συνδεσιμότητα και χρησιμοποιεί κόμβους που κινούνται για να μεταφέρει μηνύματα σε κόμβους που βρίσκονται σε απομακρυσμένες ομάδες. Το βασικό μοντέλο καλύπτει δύο περιπτώσεις: Αν ο

παραλήπτης του μηνύματος είναι στην ίδια ομάδα με τον αποστολέα (υπάρχει συνδεσιμότητα), χρησιμοποιείται για τη δρομολόγηση ένας αλγόριθμος distance vector (DSDV). Διαφορετικά, ο καλύτερος μεταφορέας του μηνύματος είναι ο κόμβος με τη μεγαλύτερη πιθανότητα επιτυχημένης παράδοσης (delivery probability). Η delivery probability υπολογίζεται τοπικά και εξαρτάται από τη συχνότητα αλλαγής των συνδέσεων και την διαθέσιμη ενέργεια (μπαταρία).

Κάθε κόμβος διατηρεί και στέλνει περιοδικά τον πίνακα δρομολόγησης του αλγορίθμου DSDV (destination, next hop, cost) και μία λίστα με τις delivery probabilities για κάθε γνωστό προορισμό (destination, best host, delivery probability). Όταν ένας κόμβος λάβει αυτές τις πληροφορίες ανανεώνει τους δικούς του πίνακες δρομολόγησης και τη λίστα με τις πιθανότητες επιτυχούς παράδοσης. Οι κόμβοι υπολογίζουν τις δικές τους delivery probabilities χρησιμοποιώντας μελλοντικές πιθανότητες παράδοσης συνδυάζοντας Kalman filters για πιο ακριβείς προβλέψεις.

Η διαδικασία της πρόβλεψης εκτελείται κατά τη διάρκεια προσωρινών αποσυνδέσεων και συνεχίζεται μέχρις ότου να μπορεί να εγγυηθεί συγκεκριμένη ακρίβεια. Οι παράγοντες που επηρεάζουν την τιμή της πιθανότητας επιτυχημένης παράδοσης είναι η συχνότητα αλλαγής συνδέσεων και η πιθανότητα ο αποστολέας και ο παραλήπτης να ανήκουν στην ίδια ομάδα. Η συχνότητα αλλαγής των συνδέσεων υπολογίζεται τοπικά εξετάζοντας το ποσοστό των γειτόνων ενός κόμβου που έχουν αλλάξει την κατάσταση σύνδεσης τους ανάμεσα σε δύο χρονικές στιγμές. Για τον υπολογισμό την πιθανότητας παραλήπτης και αποστολέας να ανήκουν στην ίδια ομάδα χρησιμοποιείται η διάρκεια της μεταξύ τους επαφής και ένα φίλτρο Kalman.

Όταν ένας κόμβος επιλέγεται σαν μεταφορέας (host) αποθηκεύει το μήνυμα στη μνήμη του. Αν ο αποστολέας και ο παραλήπτης δεν είναι στην ίδια ομάδα, ο αποστολέας στέλνει το μήνυμα στον κόμβο με το μεγαλύτερο delivery probability για τον παραλήπτη. Σε περίπτωση που ο κόμβος δεν έχει πληροφορία για τον παραλήπτη, στέλνει το μήνυμα στον κόμβο με την μεγαλύτερη κινητικότητα. Αν ο κόμβος που μεταφέρει το μήνυμα συναντήσει καθώς κινείται έναν κόμβο με μεγαλύτερο delivery probability από τον ίδιο, προωθεί το μήνυμα σε αυτόν.

Το μειονέκτημα του CAR είναι ότι στηρίζεται στον αλγόριθμο DSDV, ο οποίος δεν λειτουργεί αποτελεσματικά σε δίκτυα με συχνά εναλλασσόμενη τοπολογία. Επιπλέον, οι κόμβοι δεν έχουν πλήρη εικόνα του δικτύου αλλά στηρίζονται στις μετρικές που τους στέλνουν οι άμεσοι γείτονές τους. Αυτό μπορεί να προκαλέσει λανθασμένες αποφάσεις δρομολόγησης. Για παράδειγμα, μπορεί να υπολογίζεται καλύτερη *delivery probability* για έναν κόμβο επειδή αυτός έχει συνεχή συνδεσιμότητα με την ομάδα του αλλά στην πραγματικότητα να μην είχε ποτέ επαφή με τον παραλήπτη.

2.3.2. *Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks -HYMAD*

Το πρωτόκολλο HYMAD [19] χρησιμοποιεί κατά περίπτωση MANET και DTN. Το πρωτόκολλο σκανάρει περιοδικά την τοπολογία του δικτύου και δημιουργεί προσωρινές ομάδες κόμβων, οι οποίοι συνδέονται μεταξύ τους έτσι ώστε η διάμετρος της ομάδας να μην ξεπερνά μία συγκεκριμένη τιμή D_{max} . Η δρομολόγηση εντός της ομάδας (*intra-group routing*) γίνεται με κάποιον αλγόριθμο *distance vector*, ενώ η δρομολόγηση μεταξύ των ομάδων (*inter-group routing*) γίνεται με DTN αλγόριθμο.

Ο αλγόριθμος “*enhanced distance vector*” στηρίζεται στον DSDV με τη διαφορά ότι αν η σύνδεση με το επόμενο άλμα διακοπεί τότε ο κόμβος αποθηκεύει τα μηνύματα στη μνήμη του. Επίσης, χρησιμοποιεί επιπλέον πεδίο προτεραιότητας. Για την ασύγχρονη δρομολόγηση χρησιμοποιείται ο αλγόριθμος *Spray and Wait* [17] που αποτελείται από δύο φάσεις. Στη φάση “*spray*” ο αποστολέας του μηνύματος δημιουργεί ένα συγκεκριμένο αριθμό αντιτύπων L που προωθούνται από τον ίδιο αλλά και τους κόμβους που έλαβαν αντίτυπα σε L κόμβους. Στην επόμενη φάση που ονομάζεται “*wait*” αν το μήνυμα δεν παραδοθεί στον προορισμό του οι κόμβοι που κατέχουν ένα αντίτυπο του μηνύματος μπορούν να το προωθήσουν μόνο στον κόμβο-προορισμό.

Κάθε κόμβος κάνει περιοδικά *broadcast* δύο μηνύματα: ένα “*enhanced distance vector*” για όλα τα μέλη της ομάδας (*destination, next hop, cost, seq. Number*) και μία λίστα με τα μηνύματα που έχουν τα μέλη της ομάδας (*destination, num of copies, custodian*). Χρησιμοποιώντας την πληροφορία από τα δύο αυτά μηνύματα οι *border*

nodes, δηλαδή κόμβοι που βρίσκονται στα σύνορα των ομάδων, είναι σε θέση να γνωρίζουν την τοπολογία δύο γειτονικών ομάδων καθώς και τα μηνύματα που αυτές έχουν.

Όταν ο border node μαθαίνει ότι η ομάδα του έχει κάποιο μήνυμα που η γειτονική ομάδα δεν έχει τότε: αν ο παραλήπτης βρίσκεται στη γειτονική ομάδα, το ζητά από τον μεταφορέα (custodian), αν η ομάδα του έχει περισσότερα από ένα αντίτυπα του μηνύματος, ζητά από ένα έως $\frac{\text{αριθμός αντιτύπων}}{\text{αριθμός border nodes group}+1}$ αντίγραφα αλλιώς δεν κάνει τίποτα. Όταν ο border node λαμβάνει ένα νέο μήνυμα, αν ο παραλήπτης είναι στην ομάδα του, τότε το προωθεί (MANET), αλλιώς επιλέγει τυχαία έναν custodian.

Όταν ένας κόμβος θέλει να στείλει ένα νέο μήνυμα απλά το προσθέτει στη λίστα του και οι border nodes θα ενημερωθούν για αυτό μέσω του distance vector και θα ζητήσουν αντίγραφα για να το προωθήσουν στις γειτονικές ομάδες. Για να παραμείνει σταθερός ο αριθμός των αντιτύπων ενός μηνύματος χρησιμοποιούνται δύο μηνύματα ελέγχου “Copy Request message” για να ζητηθεί από έναν κόμβο να προωθήσει ένα αντίτυπο και “Reduce Number of Copies message” για να ενημερώσει έναν κόμβο ότι πρέπει να μειώσει τον αριθμό των αντιτύπων του μηνύματος που κατέχει.

Το HYMAD χρησιμοποιεί έναν αλγόριθμο clustering προκειμένου να χωρίσει το δίκτυο σε ομάδες ώστε η διάμετρος τους να μην ξεπερνά μία συγκεκριμένη τιμή. Δεδομένου ότι το δίκτυο δημιουργεί από τη φύση του ομάδες κόμβων είναι μία περιττή διαδικασία που καταναλώνει πολύτιμη υπολογιστική ισχύ. Επιπρόσθετα, το πρωτόκολλο ουσιαστικά προσπαθεί να διανείμει ένα αντίτυπο του μηνύματος σε κάθε ομάδα κόμβων χωρίς να θέτει κάποιο κριτήριο αξιολόγησης της ομάδας και της πιθανότητας να παραδώσει το μήνυμα στον προορισμό του, δηλαδή γίνεται μία διαδικασία παρόμοια με τους αλγορίθμους βασισμένους στην πλημμύρα.

2.3.3. *A Mobile Ad Hoc and Disruption Tolerant Routing Protocol for Tactical Military Networks- NOMAD*

Το πρωτόκολλο NOMAD [7] συνδυάζει το πρωτόκολλο OLSR με ένα single-copy DTN πρωτόκολλο. Χρησιμοποιεί το κλασικό OLSR εμπλουτίζοντάς το με λειτουργία ανάκαμψης σε περίπτωση διάσπασης του δικτύου. Θεωρεί ότι υπάρχουν πραγματικές (real) και φανταστικές (imaginary) διαδρομές. Οι πραγματικές προκύπτουν από τα γνωστά TC messages του OLSR ενώ οι φανταστικές από ένα νέο είδος TC message, το οποίο ενημερώνει το δίκτυο ότι ο παραλήπτης δεν είναι διαθέσιμος.

Σε περίπτωση που χαθεί η σύνδεση μεταξύ δύο one-hop γειτονικών κόμβων, οι κόμβοι αυτοί δρουν με τη λογική store-carry-forward ο ένας για τον άλλο αλλά και για όσους κόμβους τους έχουν επιλέξει ως MPR, γίνονται δηλαδή DTN Selectors. Οι DTN Selectors δρουν όπως και οι MPR Selectors με τη διαφορά ότι τα TC messages που παράγουν είναι σηματοδοτημένα ως φανταστικά. Όταν ο κόμβος είναι ξανά διαθέσιμος επανέρχεται η πραγματική διαδρομή. Οι πραγματικές διαδρομές προτιμώνται πάντα σε σχέση με τις φανταστικές.

Για την ασύγχρονη δρομολόγηση χρησιμοποιείται η μετρική closeness. Το πεδίο των TC messages που σηματοδοτεί ότι πρόκειται για φανταστική διαδρομή, περιλαμβάνει και την απόσταση μεταξύ των κόμβων. Για τον υπολογισμό των πινάκων δρομολόγησης χρησιμοποιούνται και οι φανταστικές διαδρομές καθώς και διαδρομές για κόμβους σε απόσταση μικρότερη των δυο αλμάτων. Οι κόμβοι με βάση τους πίνακες δρομολόγησης προωθούν τα μηνύματα σε εκείνους τους κόμβους που είναι πιο κοντά στον disrupted destination, με την ελπίδα ότι θα φτάσουν στον προορισμό τους.

Τα δύο κυριότερα μειονεκτήματα του NOMAD είναι πρώτον ότι εισάγει επιπλέον μηνύματα σηματοδοσίας κάτι που προκαλεί την επιβάρυνση του δικτύου και δεύτερον η μετρική closeness που χρησιμοποιείται δεν αποτελεί σημαντικό κριτήριο κοινωνική σχέσης.

ΚΕΦΑΛΑΙΟ 3. ΠΡΟΤΕΙΝΟΜΕΝΟΣ ΑΛΓΟΡΙΘΜΟΣ

3.1 Περιγραφή του Προβλήματος

3.2 Ο Αλγόριθμος Hybrid Synchronous/Asynchronous Routing with Social Incentives- SASI

3.3 Παραλλαγές του Αλγορίθμου SASI

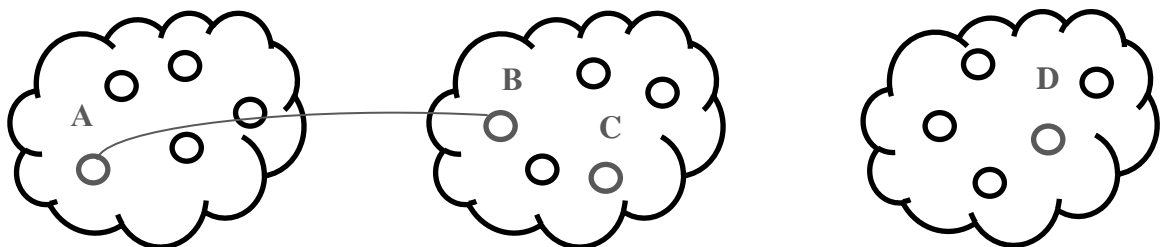
3.1. Περιγραφή του Προβλήματος

Όπως έχουμε ήδη αναφέρει το σημαντικότερο χαρακτηριστικό των MANETs είναι ότι υποθέτουν ένα πυκνό δίκτυο, με τους κόμβους να έχουν μία μέτρια κινητικότητα ώστε να μπορούν να διασφαλίσουν ότι υπάρχει συνεχής συνδεσιμότητα. Αυτό συμβαίνει διότι η επικοινωνία μεταξύ δύο κόμβων απαιτεί να υπάρχει πλήρης μονοπάτι μεταξύ τους. Αντίθετα, τα DTN/opportunistic δίκτυα δεν κάνουν καμία υπόθεση ύπαρξης μονοπατιού μεταξύ αποστολέα και παραλήπτη. Αξιοποιούν τμήματα του μονοπατιού που μπορεί να εμφανίζονται σε διαφορετικές χρονικές στιγμές λόγω της κινητικότητας των κόμβων.

Το μεγαλύτερο μέρος της έρευνας μέχρι τώρα υποθέτει είτε το ένα είδος δικτύου είτε το άλλο. Υπάρχουν όμως περιπτώσεις δικτύων που παρουσιάζουν μεικτά χαρακτηριστικά. Σε αυτά τα δίκτυα, παρατηρούνται σημεία συγκέντρωσης κόμβων, δηλαδή περιοχές όπου υπάρχει συνδεσιμότητα εντός της συγκεκριμένης περιοχής με αυξημένη πυκνότητα κόμβων. Μεταξύ, όμως, των περιοχών υπάρχει περιορισμένη συνδεσιμότητα. Σε αυτές τις περιπτώσεις ούτε οι αλγόριθμοι που έχουν προταθεί για MANET αλλά ούτε και αυτοί για DTN δίκτυα επιτυγχάνουν ικανοποιητικά

αποτελέσματα. Οι αλγόριθμοι για MANET αποτυγχάνουν να δρομολογήσουν τα δεδομένα επειδή δεν υπάρχει συνδεσιμότητα σε όλο το δίκτυο άρα και πλήρες μονοπάτι μεταξύ αποστολέα και παραλήπτη. Από την άλλη πλευρά, οι αλγόριθμοι για DTN μπορεί σε κάποιες περιπτώσεις να πετυχαίνουν σχετικά ικανοποιητικά ποσοστά παράδοσης αλλά με πολύ μεγάλες καθυστερήσεις και αυτό γιατί έχουν σχεδιαστεί για αραιά δίκτυα όπου παρατηρείται κατά διαστήματα συνδεσιμότητα.

Στο παράδειγμα του σχήματος 3.1 παρουσιάζεται ένα υπομήμη δικτύου χωρισμένο σε ομάδες κόμβων ή partitions. Ο κόμβος A μεταφέρει ένα πακέτο δεδομένων για τον κόμβο D για τον οποίο δεν γνωρίζει κάποια διαδρομή. Καθώς ο κόμβος A κινείται συναντά τον κόμβο B ο οποίος επίσης δεν γνωρίζει μία από άκρη σε άκρη διαδρομή προς τον κόμβο D. Οι αλγόριθμοι που έχουν προταθεί για MANET αδυνατούν να δρομολογήσουν το πακέτο καθώς δεν ισχύει η σημαντικότερη προϋπόθεση, να υπάρχει πλήρες μονοπάτι από τον κόμβο A στον κόμβο D.



Σχήμα 3.1 Παράδειγμα partitioned δικτύου

Από την άλλη πλευρά οι αλγόριθμοι για DTN συνήθως χρησιμοποιούν κάποιες μετρικές για να αποφασίσουν αν ένας κόμβος θα προωθήσει το πακέτο είτε αυτές στηρίζονται στο ιστορικό των επαφών, στις κοινωνικές σχέσεις ή αλλού. Οπότε η δρομολόγηση σε αυτή την περίπτωση θα εξαρτηθεί από τις τιμές των μετρικών. Έστω ότι οι τιμές των μετρικών είναι αυτές που παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 3.1 Τιμές των μετρικών για τον κόμβο D

Κόμβος	Utility _D
A	0,4
B	0,2
C	0,8

Από τις τιμές των μετρικών για τον κόμβο D προκύπτει ότι ο κόμβος με τις μεγαλύτερες πιθανότητες να παραδώσει το πακέτο είναι ο κόμβος C. Όταν ο A θα συναντήσει τον κόμβο B θα συγκρίνει τις τιμές των μετρικών και δεν θα του προωθήσει το πακέτο. Αν καθώς κινείται ο κόμβος A συναντήσει τον κατάλληλο κόμβο για να μεταφέρει τον μήνυμα στον D το πακέτο θα παραδοθεί τελικώς στον προορισμό του διαφορετικά ίσως όχι. Σε κάθε περίπτωση έχει προσπεράσει μία πολύ καλή ευκαιρία προώθησης του πακέτου προς τον παραλήπτη. Σε αυτές τις περιπτώσεις των δικτύων χρειάζεται κάποιο υβριδικό πρωτόκολλο που θα εναλλάσσεται μεταξύ σύγχρονης και ασύγχρονης λειτουργίας ανάλογα με τις συνθήκες του δικτύου. Θα εκμεταλλεύεται την συνδεσιμότητα όταν είναι διαθέσιμη και θα μεταβαίνει σε hop-by-hop μορφή όταν δεν είναι.

Όπως έχουμε ήδη αναφέρει τα reactive υβριδικά πρωτόκολλα αντιμετωπίζουν σοβαρά προβλήματα λειτουργίας. Το σημαντικότερο πρόβλημα είναι η εναλλαγή μεταξύ σύγχρονου και ασύγχρονου τρόπου δρομολόγησης. Συνήθως οι αλγόριθμοι αυτοί όταν ανιχνεύσουν ότι δεν υπάρχει συνδεσιμότητα μεταπίπτουν σε ασύγχρονη λειτουργία. Το πρόβλημα προκύπτει κατά την επιλογή του χρόνου ανάκαμψης από αυτήν αφού εξαρτάται από τις συνθήκες του δικτύου. Εγείρονται, επομένως, ερωτήματα σχετικά με το πότε θα πρέπει να γίνει προσπάθεια ανακάλυψης μίας από άκρο σε άκρο διαδρομής όταν το πρωτόκολλο λειτουργεί με ασύγχρονο τρόπο. Σύμφωνα με τη φύση των reactive πρωτοκόλλων, η διαδικασία αυτή ενεργοποιείται όταν υπάρχει ανάγκη για την μετάδοση ενός πακέτου δεδομένων και δεν υπάρχει γνωστή διαδρομή για τον προορισμό. Το δίλλημα που αντιμετωπίζουν είναι ότι αν η προσπάθεια αυτή είναι πολύ συχνή προκαλείται μεγάλη επιβάρυνση στο δίκτυο κατά

την αναζήτηση, ενώ αν είναι λιγότερο συχνή είναι πιθανόν να μην αξιοποιηθούν ευκαιρίες που προκύπτουν για τη δρομολόγηση των δεδομένων.

Στο προηγούμενο κεφάλαιο παρουσιάσαμε proactive υβριδικά πρωτόκολλα και τα προβλήματα που αυτά αντιμετωπίζουν. Το HYMAD, για παράδειγμα, προσπαθεί να διανείμει ένα αντίτυπο του μηνύματος σε κάθε ομάδα κόμβων, εφαρμόζοντας ουσιαστικά τη μέθοδο της πλημμύρας μεταξύ των ομάδων καταναλώνοντας πολύτιμους πόρους. Στο παράδειγμα του σχήματος 3.1 ο αλγόριθμος θα προσπαθούσε να διανείμει ένα αντίτυπο του μηνύματος σε κάθε ομάδα κόμβων εφόσον αυτό επιτρέπεται από την κίνηση των κόμβων. Η διάδοση αντιτύπων του μηνύματος σε όλες τις ομάδες κόμβων χωρίς κριτήρια αξιολόγησης τους δεν αποτελεί σε καμία περίπτωση την επιθυμητή προσέγγιση.

Ούτε το πρωτόκολλο NOMAD μπορεί να ανταποκριθεί στην συγκεκριμένη περίπτωση αφού δεν υπάρχει συνδεσιμότητα για να χρησιμοποιηθεί η σύγχρονη λειτουργία. Επιπλέον, η μετρική closeness που χρησιμοποιείται για την ασύγχρονη δρομολόγηση δεν αποτελεί σημαντικό κριτήριο κοινωνική σχέσης. Παράλληλα, το πρωτόκολλο στηρίζεται στην υπόθεση ότι ο κόμβος που έχει επιλεγεί λόγω της μετρικής να μεταφέρει τα μηνύματα για συγκεκριμένο παραλήπτη, θα συναντηθεί κάποια στιγμή με τον παραλήπτη αυτό. Το γεγονός αυτό όμως είναι αυθαίρετο και δεν ανταποκρίνεται απαραίτητα στην πραγματικότητα.

Τέλος, σύμφωνα με το CAR ο κόμβος θα προωθήσει το μήνυμα στον κόμβο με τη μεγαλύτερη πιθανότητα παράδοσης που βρίσκεται στην ομάδα του. Το πρόβλημα όμως είναι ότι δεν ενεργοποιείται κάποια διαδικασία ανταλλαγής μηνυμάτων όταν δύο κόμβοι έρθουν σε επαφή. Έτσι, όταν ο κόμβος A συναντήσει τον B, δεν θα ενημερωθεί ότι υπάρχει στη δεύτερη ομάδα κόμβων με καλύτερη πιθανότητα παράδοσης και το μήνυμα δεν θα περάσει στη δεύτερη ομάδα κόμβων. Επιπρόσθετα, προβλήματα φαίνεται να παρουσιάζει και ο τρόπος υπολογισμού της πιθανότητας παράδοσης. Για παράδειγμα, μπορεί να υπολογίζεται καλύτερη πιθανότητα παράδοσης για έναν κόμβο επειδή αυτός έχει συνεχή συνδεσιμότητα με την ομάδα του αλλά στην πραγματικότητα να μην είχε ποτέ επαφή με τον παραλήπτη.

Το πρόβλημα της δρομολόγησης που παρουσιάστηκε θα μπορούσε να λυθεί αν ο κόμβος B γνώριζε ότι υπάρχει στην ομάδα του ο κόμβος C που έχει τη μεγαλύτερη πιθανότητα να παραδώσει το μήνυμα. Αν το γνώριζε, θα μπορούσε να το ανακοινώσει στον A κατά τη συνάντησή τους και το μήνυμα να περνούσε στο δεύτερο partition. Θα μπορούσε λοιπόν ένας αλγόριθμος να χρησιμοποιεί έναν σύγχρονο αλγόριθμο για την δρομολόγηση των δεδομένων όταν υπάρχει συνδεσιμότητα ενώ ταυτόχρονα θα διατηρεί και τις μετρικές δρομολόγησης ενός ασύγχρονου αλγόριθμου. Οι τιμές των μετρικών μπορούν να διαδοθούν στους υπόλοιπους κόμβους του partition χρησιμοποιώντας τη σύγχρονη λειτουργία του αλγόριθμου.

3.2. Ο Αλγόριθμος Hybrid Synchronous/Asynchronous Routing with Social Incentives- SASI

Στην παρούσα εργασία προτείνεται ένας νέος υβριδικός αλγόριθμος που λειτουργεί με ένα πρωτόκολλο για MANET όταν αυτό είναι δυνατό και μεταπίπτει σε ασύγχρονη λειτουργία όταν απαιτείται, αντιμετωπίζοντας τα προβλήματα που παρουσιάστηκαν στην προηγούμενη παράγραφο.

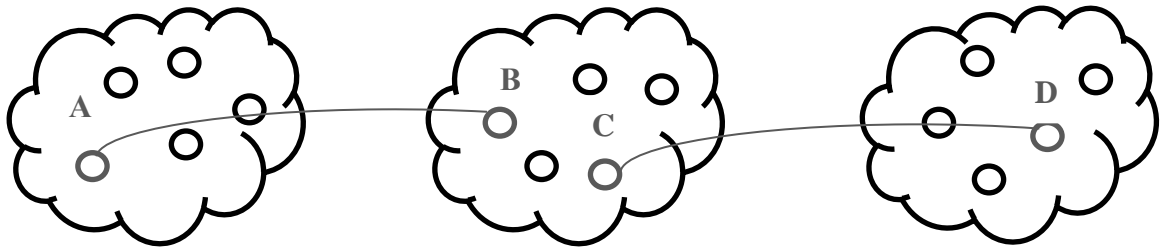
Από τα MANET πρωτόκολλα που έχουν προταθεί θεωρούμε ότι το OLSR ταιριάζει καλύτερα στις απαιτήσεις του είδους των δικτύων που μελετάμε. Πρώτον, πρόκειται για proactive αλγόριθμο οπότε δεν αντιμετωπίζουμε προβλήματα χρονοπρογραμματισμού της μετάβασης μεταξύ των πρωτοκόλλων όπως τα reactive υβριδικά πρωτόκολλα. Επιπροσθέτως, είναι κατάλληλο για πυκνά δίκτυα με μεγάλο φόρτο, κάτι που μπορεί να ισχύει στις περιοχές συγκέντρωσης των κόμβων. Τέλος επιτυγχάνει μικρούς χρόνους καθυστέρησης στη δρομολόγηση των δεδομένων. Για αυτούς τους λόγους θα το χρησιμοποιήσουμε ως τη βάση του πρωτοκόλλου που θα διαχειριστεί την από άκρο σε άκρο δρομολόγηση. Όσο αφορά την ασύγχρονη δρομολόγηση, θεωρήθηκε καλό να χρησιμοποιηθεί ένας αλγόριθμος κοινωνικής δικτύωσης λόγω του ότι στηρίζεται στις πραγματικές κοινωνικές σχέσεις μεταξύ των χρηστών των ασύρματων συσκευών. Ίσως ο πιο γνωστός αλγόριθμος της κατηγορίας αυτής είναι ο Simbet.

Ο αλγόριθμος Simbet χρησιμοποιεί για τη δρομολόγηση των δεδομένων μετρικές κοινωνικές δικτύωσης που υπολογίζονται με τη βοήθεια του ego network. Ο λόγος που χρησιμοποιείται το ego network είναι ότι για τον υπολογισμό των μετρικών για όλο τον κοινωνικό γράφο απαιτείται πληροφορία που είναι δύσκολο και χρονοβόρο να βρεθεί. Από την άλλη πλευρά ο OLSR χρησιμοποιεί μηνύματα σηματοδοσίας (Topology Control- TC messages), για να ανακοινώσει τους κόμβους που έχει επιλέξει ως MPR Selectors, τα οποία διαχέονται σε όλο το δίκτυο. Μπορούμε, λοιπόν, να χρησιμοποιήσουμε τα μηνύματα αυτά για να μεταδώσουμε πληροφορίες σχετικά με τις κοινωνικές σχέσεις των κόμβων σε απόσταση μεγαλύτερη από αυτή του ego network, χωρίς να δημιουργήσουμε επιπλέον μηνύματα σηματοδοσίας. Είναι δηλαδή πλέον εφικτό ο υπολογισμός των μετρικών κοινωνικής δικτύωσης να γίνει πιο ακριβής και γνωστός σε όλους τους κόμβους για κάθε άλλο κόμβο με τον οποίο υπάρχει συνδεσιμότητα.

Έχοντας ως δεδομένο τις περιπτώσεις δικτύων που παρατηρούνται περιοχές συγκέντρωσης κόμβων ή partitions, τα Topology Control messages διαχέονται μόνο εντός του partition του κόμβου, λόγω έλλειψης συνδεσιμότητας έξω από αυτό. Προσθέτοντας οι κόμβοι πληροφορίες σχετικά με τις κοινωνικές τους σχέσεις στο TC message, επιτρέπει σε κάθε κόμβο εντός του partition να μπορεί να αναπαραστήσει τον κοινωνικό γράφο όλου του partition με αντίστοιχο τρόπο όπως αναπαριστά την τοπολογία. Κατ' επέκταση μπορεί να γνωρίζει τις τιμές των μετρικών κοινωνικής δικτύωσης κάθε άλλου κόμβου. Με αυτό τον τρόπο κάθε κόμβος αντί να συγκρίνει τη μετρική του με έναν άλλο όπως στον αλγόριθμο Simbet μπορεί πλέον να συγκριθεί με όλους τους κόμβους του partition.

Το πρόβλημα της δρομολόγησης που παρουσιάστηκε στην προηγούμενη παράγραφο λύνεται συνδυάζοντας τα δύο πρωτόκολλα OLSR και Simbet. Όταν συναντηθούν οι κόμβοι A και B, θα ενεργοποιηθεί η διαδικασία ανταλλαγής μηνυμάτων που ορίζεται από τον αλγόριθμο Simbet. Ταυτόχρονα, ο B γνωρίζει ότι υπάρχει στο partition του ο κόμβος C που έχει μεγαλύτερη μετρική από τον A άρα και αυξημένες πιθανότητες να παραδώσει το πακέτο στον D. Επομένως, ο B θα ζητήσει από τον A να του αποστείλει το πακέτο και αυτός με τη σειρά του θα το προωθήσει στον κόμβο C,

χρησιμοποιώντας την σύγχρονη δρομολόγηση του OLSR, ο οποίος με τη σειρά του μπορεί να το παραδώσει επιτυχώς στον D.



Σχήμα 3.2 Παράδειγμα partitioned δικτύου

3.2.1. Σύγχρονη Λειτουργία

Όπως αναφέραμε στο 2ο κεφάλαιο, το πρωτόκολλο OLSR είναι μία βελτιστοποίηση του κλασικού link state πρωτοκόλλου. Το OLSR χρησιμοποιεί για την διάδοση των Link State packets τους MultiPoint Relay (MPR) κόμβους. Δηλαδή, αντί ένας κόμβος να ανακοινώνει όλες τις ζεύξεις του σε όλους τους κόμβους του δικτύου, ανακοινώνει μόνο τις ζεύξεις του με τους Multipoint relay selectors γείτονές του. Οι Multipoint Relays είναι οι μόνοι που προωθούν τα μηνύματα ελέγχου, μειώνοντας το φόρτο του δικτύου. Δεδομένου ότι το OLSR είναι ένα table-driven πρωτόκολλο, δηλαδή χρησιμοποιεί πίνακες δρομολόγησης, όλη του η λειτουργία στηρίζεται σε πίνακες-δομές δεδομένων. Αυτές είναι:

- Multiple Interface Association: το OLSR χρησιμοποιεί τις IP διευθύνσεις για να χαρακτηρίσει μοναδικά κάθε κόμβο. Κάθε κόμβος όμως μπορεί να έχει μία βασική διεύθυνση και πολλές άλλες δευτερεύουσες. Η λίστα αυτή περιέχει όλες τις διευθύνσεις του κόμβου.
- Link Set: περιέχει τις συνδέσεις του κόμβου με τους άμεσους (1-hop) γείτονές του.
- Neighbor Set: περιέχει τους άμεσους γείτονες του κόμβου.
- 2-hop Neighbor Set: περιέχει του γείτονες που είναι προσβάσιμοι μέσω των άμεσων γειτόνων.
- MPR Set: οι κόμβοι που έχουν επιλεγεί ως MPR.
- MPR Selector Set: περιέχει όλους τους γείτονες που έχουν επιλέξει τον κόμβο ως MPR.

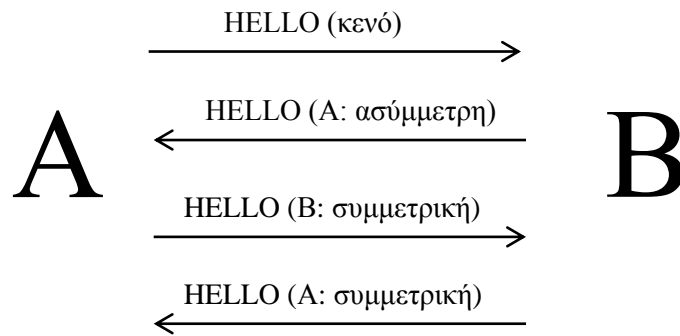
- Duplicate Set: η δομή αυτή περιέχει τα μηνύματα που επεξεργάστηκε ή προώθησε πρόσφατα ο κόμβος.
- Topology Set: περιέχει πληροφορίες σχετικά με την τοπολογία του δικτύου που προέρχεται από τα μηνύματα TC.

Τα περισσότερα από τα δεδομένα που διατηρούνται σε αυτές τις δομές συσχετίζονται με ένα χρονικό όριο μετά από το οποίο δεν θεωρούνται πλέον ότι ισχύουν.

Η λειτουργία του OLSR στηρίζεται στα μηνύματα σηματοδοσίας που είναι τριών ειδών: Multiple Interface Declaration (MID), Hello και Topology Control (TC).

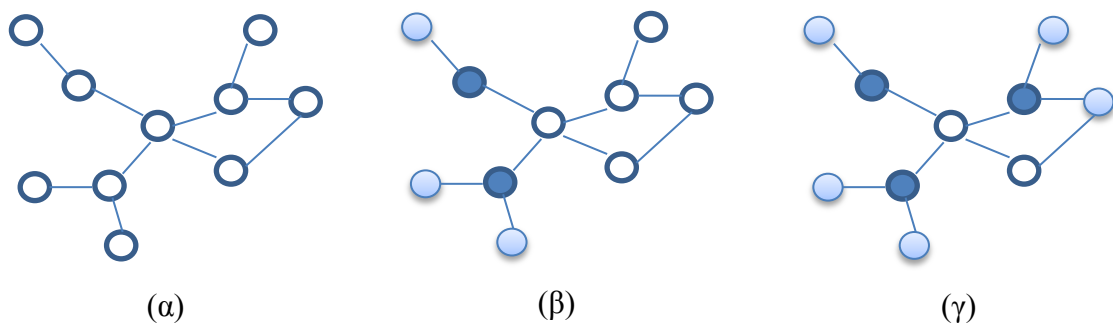
Όταν χρησιμοποιείται το OLSR με πολλαπλές διεπαφές, τα μηνύματα MID διαχέονται με τη μέθοδο της πλημμύρας σε όλο το δίκτυο για να ενημερώσουν τους υπόλοιπους κόμβους σχετικά με τη βασική και τις λοιπές διευθύνσεις που χρησιμοποιεί ο κόμβος.

Τα μηνύματα Hello εξυπηρετούν πολλούς σκοπούς στο πρωτόκολλο OLSR. Μεταδίδονται περιοδικά ανά σταθερά χρονικά διαστήματα και περιέχουν τους κόμβους που ανήκουν στην τοπική γειτονιά του καθώς και το είδος της σύνδεσης συμμετρική (bi-directional), ασύμμετρη (uni-directional) ή MPR. Η διαδικασία παρουσιάζεται απλοποιημένη στο σχήμα 3.3. Αρχικά ο κόμβος A στέλνει ένα μήνυμα Hello που δεν περιέχει τίποτα. Λαμβάνοντάς το ο κόμβος B καταγράφει τον A ως ασύμμετρο γείτονά του γιατί δεν βλέπει την δική του διεύθυνση στο Hello. Έπειτα ο B στέλνει Hello που περιέχει την διεύθυνση του A. Όταν το λάβει ο A βρίσκει ως ασύμμετρη ζεύξη τη διεύθυνσή του, οπότε θεωρεί πλέον τη ζεύξη συμμετρική και προσθέτει την πληροφορία στο επόμενο Hello μήνυμα. Τα μηνύματα Hello τα λαμβάνουν όλοι οι 1-hop γείτονες και δεν προωθούνται στο υπόλοιπο δίκτυο. Χρησιμοποιώντας τις πληροφορίες από τα μηνύματα Hello οι κόμβοι μπορούν να κατασκευάσουν τις δομές Link Set, Neighbor Set και 2-hop Neighbor Set. Λαμβάνοντας Hello από έναν γείτονα με συμμετρική ζεύξη, όλες οι συμμετρικές ζεύξεις των κόμβων που περιέχονται στο Hello και δεν ανήκουν στους ενός άλματος γείτονες του κόμβου θεωρούνται γείτονες δύο αλμάτων, καταγράφονται στο 2-hop Neighbor Set.



Σχήμα 3.3 Ανακάλυψη γειτόνων με τη βοήθεια Hello μηνυμάτων

Χρησιμοποιώντας το 2-hop Neighbor Set ο κόμβος επιλέγει το ελάχιστο δυνατό σύνολο γειτονικών κόμβων που καλύπτουν όλους τους γείτονες σε απόσταση δύο αλμάτων, δηλαδή τους MultiPoint relays του. Η διαδικασία της επιλογής παρουσιάζεται στο σχήμα 3.4 και έχει ως εξής: Ο κόμβος αρχικά επιλέγει ως MPR όλους τους γείτονες που είναι μοναδικοί γείτονες ενός κόμβου σε απόσταση δύο αλμάτων (Σχήμα 3.4-β). Έπειτα διαλέγει σαν MPR έναν γείτονα που καλύπτει το μεγαλύτερο αριθμό κόμβων δύο αλμάτων που δεν καλύπτονται ακόμη από το σύνολο MPR (Σχήμα 3.4-γ). Τέλος, ελέγχεται κάθε κόμβος που έχει επιλεγεί ως MPR. Αν αποκλειστεί από το MPR σύνολο και συνεχίζουν να καλύπτονται όλοι οι κόμβοι, τότε απορρίπτεται. Οι κόμβοι που επέλεξε ως MPR θα καταγραφούν στη δομή MPR Set και θα τους δοθεί ο αντίστοιχος χαρακτηρισμός στο επόμενο Hello μήνυμα που θα αποσταλεί, ενημερώνοντας τους κόμβους ώστε να καταρτίσουν τη λίστα των MPR selectors. Επαναυπολογισμός του συνόλου MPR γίνεται όταν διαπιστωθεί αλλαγή στην τοπική γειτονιά (ανίχνευση ή απώλεια σύνδεσης σε συμμετρική ζεύξη) ή ανιχνευτεί κάποια συμμετρική ζεύξη σε απόσταση δύο αλμάτων.



Σχήμα 3.4 Επιλογή MPR κόμβων

Κάθε κόμβος που έχει επιλεγεί ως MPR, εκπέμπει μηνύματα ελέγχου που ονομάζονται Topology Control (TC) μηνύματα και προωθούνται σε όλο το δίκτυο μέσω των MPR κόμβων. Τα TC μηνύματα περιέχουν τη λίστα των κόμβων που έχουν επιλέξει τον κόμβο που αποστέλλει το μήνυμα ως Multipoint relay, καθώς και έναν ακολουθιακό αριθμό. Ο ακολουθιακός αριθμός χρησιμοποιείται για να υποδείξει πόσο πρόσφατη είναι η πληροφορία που περιέχει το TC. Ο ακολουθιακός αριθμός δεν αυξάνεται σε κάθε νέα αποστολή TC μηνύματος αλλά όταν υπάρχει αλλαγή στη γειτονιά του κόμβου. Τα μηνύματα αυτά αποστέλλονται περιοδικά, αν και γίνονται και έκτακτες αποστολές όταν διαπιστωθούν αλλαγές στην τοπική γειτονιά και προωθούνται σε όλο το δίκτυο μέσω των MPR κόμβων, εφόσον λαμβάνονται για πρώτη φορά.

Όταν ένας κόμβος λάβει ένα TC μήνυμα ανανεώνεται η δομή Topology Set ως εξής:

- Αν δεν υπάρχει εγγραφή σχετικά με τον αποστολέα του TC, δημιουργείται μία.
- Αν υπάρχει εγγραφή με ακολουθιακό αριθμό μικρότερο από αυτόν του TC, τότε γίνεται ανανέωση σύμφωνα με τις πληροφορίες του TC.
- Αν υπάρχει εγγραφή με ακολουθιακό αριθμό ίσο με αυτόν του TC, τότε ανανεώνεται το χρονικό όριο μετά από το οποίο η εγγραφή λήγει.

Κάθε εγγραφή του πίνακα τοπολογίας περιέχει: τη διεύθυνση του προορισμού (τον MPR selector που περιέχεται στο TC), τη διεύθυνση του τελευταίου άλματος (ο αποστολέας του TC), τον ακολουθιακό αριθμό και το χρόνο λήξης της εγγραφής.

Ο πίνακας δρομολόγησης κάθε κόμβου κατασκευάζεται χρησιμοποιώντας τις δομές Neighbor Set και 2-hop Neighbor Set για τους γείτονες έως δύο άλματα απόσταση και το Topology Set για μεγαλύτερη απόσταση. Κάθε εγγραφή του πίνακα δρομολόγησης έχει τη μορφή <προορισμός, επόμενο άλμα, αριθμός αλμάτων>. Για τον υπολογισμό του ακολουθούνται τα παρακάτω βήματα:

- 1) Πρόσθεσε όλους τους άμεσους γείτονες με χαρακτηρισμό σύνδεσης «συμμετρική» και θέσε ως αριθμό αλμάτων 1.

2) Για κάθε συμμετρικό άμεσο γείτονα, πρόσθεσε όλους τους γείτονες του κόμβου (γείτονες δύο αλμάτων) που:

- Δεν έχουν ήδη προστεθεί.
- Έχουν συμμετρική ζεύξη.

Οι εγγραφές αυτές προστίθενται με αριθμό αλμάτων δύο και επόμενο άλμα τον τρέχοντα γείτονα.

3) Έπειτα, για κάθε κόμβο N που έχει προστεθεί στον πίνακα δρομολόγησης με αριθμό αλμάτων $n = 2$ πρόσθεσε όλες τις εγγραφές από το Topology Set όπου:

- η διεύθυνση του προορισμού στο Topology Set είναι ίση με N
- ο προορισμός δεν έχει ήδη προστεθεί

Οι νέες εγγραφές προστίθενται με αριθμό αλμάτων $n + 1$ και επόμενο άλμα τη διεύθυνση που αναφέρεται στο Topology Set.

4) Αύξησε το n κατά ένα και επανέλαβε το βήμα 3 έως ότου να μην υπάρχουν άλλες εγγραφές με αριθμό αλμάτων $n + 1$.

Όταν ένας κόμβος θέλει να προωθήσει ένα πακέτο, αναζητά στον πίνακα δρομολόγησης εγγραφή για τον συγκεκριμένο προορισμό προκειμένου να προωθήσει το μήνυμα στο επόμενο άλμα. Αν δεν βρεθεί εγγραφή, σημαίνει ότι ο κόμβος δεν έχει συνδεσιμότητα με τον προορισμό και απορρίπτει το πακέτο.

3.2.2. Ασύγχρονη Λειτουργία

Ο αλγόριθμος που προτείνεται επεκτείνει το πρωτόκολλο OLSR εμπλουτίζοντάς το με ασύγχρονη λειτουργία και πιο συγκεκριμένα με τον αλγόριθμο Simbet. Όπως είπαμε, σύμφωνα με τον αλγόριθμο OLSR κάθε κόμβος εκπέμπει περιοδικά Hello μηνύματα. Κάθε φορά που ο κόμβος λαμβάνει Hello μήνυμα από κάποιον νέο γείτονα παραδίδει καταρχήν όσα αποθηκευμένα πακέτα έχει γι' αυτόν και έπειτα ανανεώνει τη λίστα επαφών του. Η λίστα επαφών είναι μία δομή δεδομένων που περιέχει τους κόμβους με τους οποίους έχει έρθει σε επαφή. Επιπλέον, διατηρείται και το σύνολο των επαφών κάθε κόμβου που περιέχεται σε αυτή λίστα, έτσι ώστε να μπορεί να αναπαρασταθεί ο κοινωνικός γράφος σε απόσταση δύο αλμάτων.

Έπειτα ακολουθεί ανταλλαγή μηνυμάτων όπως ορίζεται από τον αλγόριθμο Simbet. Δηλαδή το κόμβος θα στείλει ένα Request encounter μήνυμα ζητώντας από τον νέο γείτονα να του στείλει τις δικές του επαφές. Όταν ο κόμβος λάβει ο ίδιος ένα Request encounter μήνυμα αποστέλλει τις επαφές του. Όταν λάβει λοιπόν ο κόμβος το μήνυμα που περιέχει τις επαφές του γείτονά του το χρησιμοποιεί για να ανανεώσει τη λίστα των επαφών. Έπειτα, μπορεί ο κόμβος να αναπαραστήσει το ego network του σε απόσταση δύο αλμάτων. Το ego network μπορεί να οριστεί ως ένα δίκτυο ενός κόμβου (ego) με τους γειτονικούς του κόμβους (alters) και όλες τις μεταξύ τους συνδέσεις. Πάνω σε αυτό το ego network μπορεί πλέον να υπολογίσει τις μετρικές κοινωνικής δικτύωσης. Οι μετρικές που χρησιμοποιούνται είναι το betweenness centrality και similarity που υπολογίζονται ως εξής:

- Betweenness centrality:

$$C_B(p_i) = \sum_{j=1}^N \sum_{k=1}^{j-1} \frac{g_{jk}(p_i)}{g_{jk}} \quad \text{Εξ. 3.1}$$

όπου g_{jk} είναι ο συνολικός αριθμός των συντομότερων μονοπατιών που ενώνουν δύο οποιουσδήποτε κόμβους του δικτύου j και k , και $g_{jk}(p_i)$ είναι το πλήθος των συντομότερων μονοπατιών στα οποία συμπεριλαμβάνεται ο κόμβος p_i .

- Similarity:

$$P_{(x,y)} = |N_{(x)} \cap N_{(y)}| \quad \text{Εξ. 3.2}$$

όπου $N_{(x)}$ και $N_{(y)}$ είναι το σύνολο των επαφών του x και y αντίστοιχα.

Αφού ο κόμβος υπολογίσει τις τιμές των μετρικών για κάθε προορισμό εν συνεχεία θα τις στείλει στον γείτονα κόμβο με μήνυμα Summary vector.

Οι μετρικές αυτές κοινοποιούνται επίσης και σε όλο το δίκτυο χρησιμοποιώντας τα Topology Control (TC) μηνύματα του OLSR. Τα TC μηνύματα περιέχουν πλέον τη λίστα των γειτόνων που έχουν επιλέξει τον αποστολέα του TC ως Multipoint Relay, τον ακολουθιακό αριθμό που δείχνει πόσο πρόσφατη είναι η επιλογή και τις τιμές των μετρικών betweenness και similarity.

Όταν ένας κόμβος λάβει ένα TC μήνυμα εκτός από τις ενέργειες που αναφέρθηκαν παραπάνω, θα αποθηκεύσει τις τιμές των μετρικών betweenness και similarity σε κατάλληλες δομές δεδομένων και θα τις διατηρήσει μέχρι να ανανεωθούν.

Σύμφωνα με τον αλγόριθμο Simbet, όταν ένας κόμβος λάβει τις μετρικές με Summary vector μήνυμα υπολογίζει τη μετρική SimbetUtil του κόμβου που στέλνει το Summary vector ως εξής:

Αν A και B είναι οι κόμβοι, Bet_A και Bet_B οι τιμές του betweenness και $Sim_A(d)$ και $Sim_B(d)$ είναι οι τιμές του similarity για τον προορισμό d.

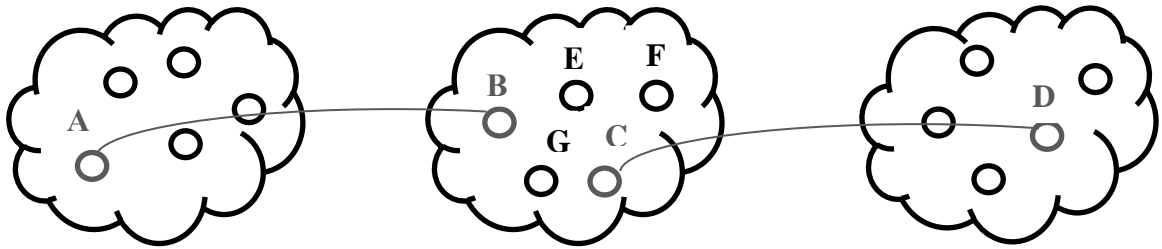
$$\circ \quad SimUtil_A(d) = \frac{Sim_A(d)}{Sim_A(d)+Sim_B(d)} \quad \text{Εξ. 3.3}$$

$$\circ \quad BetUtil_A = \frac{Bet_A}{Bet_A + Bet_B} \quad \text{Εξ. 3.4}$$

$$\circ \quad SimBetUtil_A(d) = a * SimUtil_A(d) + b * BetUtil_A \quad \text{Εξ. 3.5}$$

όπου $a+b=1$, με συνηθισμένες τιμές τις $a=b=0,5$

Έπειτα γνωρίζοντας όλες τις μετρικές των κόμβων του partition του υπολογίζει το καλύτερο SimbetUtil του partition. Στο παράδειγμα του σχήματος 3.5 ο κόμβος B γνωρίζει τους υπόλοιπους κόμβους που βρίσκονται στο partition του. Η πληροφορία αυτή παρέχεται από τον πίνακα δρομολόγησης του OLSR. Επίσης γνωρίζει και τις τιμές των μετρικών τους betweenness και similarity αφού τις λαμβάνει περιοδικά μέσω των TC μηνυμάτων που αποστέλλουν. Όταν λοιπόν συναντηθούν οι κόμβοι A και B, ο B θα υπολογίσει τις τιμές της μετρικής SimbetUtil για τον ίδιο ($SimBetUtil_B(D)$), για τον A ($SimBetUtil_A(D)$) αλλά και για όλους τους υπόλοιπους κόμβους C, E, F και G ($SimBetUtil_C(D)$, $SimBetUtil_E(D)$, $SimBetUtil_F(D)$, $SimBetUtil_G(D)$). Έπειτα συγκρίνει τις μετρικές που υπολόγισε και επιλέγει αυτή με τη μεγαλύτερη τιμή που στο παράδειγμα που παρουσιάζουμε είναι του κόμβου C.



Σχήμα 3.5 Παράδειγμα partitioned δικτύου

Αν η καλύτερη μετρική $SimbetUtil$ του partition είναι μεγαλύτερη από αυτή του κόμβου που στέλνει το Summary vector, στο συγκεκριμένο παράδειγμα αν $SimBetUtil_C(D) > SimBetUtil_A(D)$, στέλνει ένα μήνυμα message request ζητώντας να του προωθήσει το πακέτο, διαφορετικά δεν κάνει τίποτα. Αφού λάβει το πακέτο, το προωθεί στον κόμβο με την καλύτερη μετρική με σύγχρονο τρόπο, αξιοποιώντας τον πίνακα δρομολόγησης. Δηλαδή, όταν ο B λάβει τον πακέτο από τον A θα αναζητήσει μία διαδρομή στον πίνακα δρομολόγησής του για τον C και θα του προωθήσει το πακέτο.

Όταν ένας κόμβος θέλει να στείλει ένα νέο μήνυμα προσπαθεί αρχικά να το προωθήσει χρησιμοποιώντας τον πίνακα δρομολόγησης του OLSR. Αν δεν υπάρχει εγγραφή για τον συγκεκριμένο προορισμό αποθηκεύει το πακέτο στη μνήμη του. Κάθε φορά που προστίθεται μία καινούρια εγγραφή στον πίνακα δρομολόγησης γίνεται αναζήτηση στη μνήμη που κρατούνται τα πακέτα και σε περίπτωση που έγινε διαθέσιμη διαδρομή προς τον παραλήπτη, το πακέτο προωθείται άμεσα προς αυτόν με χρήση του πίνακα δρομολόγησης.

Συνδυάζοντας τα δύο πρωτόκολλα OLSR και Simbet αξιοποιούμε τη σύγχρονη λειτουργία όταν υπάρχει συνδεσιμότητα επιτυγχάνοντας μικρή καθυστέρηση παράδοσης αλλά ταυτόχρονα ανταποκρινόμαστε σε περιπτώσεις διάσπασης με τα πακέτα να προωθούνται στον κόμβο που έχει περισσότερες πιθανότητες να παραδώσει το μήνυμα στον προορισμό του, επιλέγοντάς τον από το σύνολο των κόμβων της ομάδας.

Παρακάτω παρουσιάζονται οι βασικές διαδικασίες που εκτελεί ο προτεινόμενος αλγόριθμος SASI σε αντιπαράθεση με τους Simbet και OLSR.

Πίνακας 3.2 Ενέργειες αλγορίθμων μετά τη λήψη Hello μηνύματος

Όταν ο κόμβος n λάβει Hello μήνυμα από τον m		
SIMBET	OLSR	SASI
<ul style="list-style-type: none"> ▪ Εάν είναι καινούριος γείτονας, του παραδίδει όσα μηνύματα έχει αποθηκευμένα με προορισμό τον m. ▪ Ζητά από τον m να του στείλει τη λίστα με τις επαφές του (encounter vector) 	<ul style="list-style-type: none"> ▪ Ανανεώνει τον πίνακα γειτόνων ▪ Ανανεώνει τον πίνακα MPR Selector ▪ Επιλέγει τους MPR Relays 	<ul style="list-style-type: none"> ▪ Εάν είναι καινούριος γείτονας, του παραδίδει όσα μηνύματα έχει αποθηκευμένα με προορισμό τον m. ▪ Ζητά από τον m να του στείλει τη λίστα με τις επαφές του (encounter vector) ▪ Ανανεώνει τον πίνακα γειτόνων ▪ Ανανεώνει τον πίνακα MPR Selector ▪ Επιλέγει τους MPR Relays

Πίνακας 3.3 Ενέργειες αλγορίθμων μετά τη λήψη TC μηνύματος

Όταν ο κόμβος n λάβει TC μήνυμα από τον m		
SIMBET	OLSR	SASI
	<ul style="list-style-type: none"> ▪ Ανανεώνει τον πίνακα τοπολογίας ▪ Ανανεώνει τον πίνακα δρομολόγησης 	<ul style="list-style-type: none"> ▪ Ανανεώνει τον πίνακα τοπολογίας ▪ Ανανεώνει τον πίνακα δρομολόγησης ▪ Ανανεώνει τις τιμές των betweenness και similarity των υπολοίπων κόμβων (περιέχονται στο TC) ▪ Αν υπάρχει νέα εγγραφή στον πίνακα δρομολόγησης προωθεί όσα μηνύματα έχει αποθηκευμένα με προορισμό τη νέα εγγραφή

Πίνακας 3.4 Ενέργειες αλγορίθμων μετά τη λήψη Encounter Vector

Αν ο κόμβος n λάβει encounter vector από τον κόμβο m		
SIMBET	OLSR	SASI
<ul style="list-style-type: none"> ▪ Ανανεώνει το ego network ▪ Υπολογίζει betweenness και similarity για όλους τους προορισμούς για τους οποίους διαθέτει πακέτο ▪ Ανταλλαγή της λίστας που περιέχει όλους τους προορισμούς και τις μετρικές που υπολόγισε (summary vector) 		<ul style="list-style-type: none"> ▪ Ανανεώνει το ego network ▪ Υπολογίζει betweenness και similarity για όλους τους προορισμούς για τους οποίους διαθέτει πακέτο ▪ Ανταλλαγή της λίστας που περιέχει όλους τους προορισμούς και τις μετρικές που υπολόγισε (summary vector)

Πίνακας 3.5 Ενέργειες αλγορίθμων μετά τη λήψη Summary Vector

Όταν ένας κόμβος n λάβει summary vector από τον κόμβο m		
SIMBET	OLSR	SASI
<ul style="list-style-type: none"> ▪ για κάθε προορισμό που περιέχεται στο summary vector, υπολογίζει $\text{SimBetUtil}_n(\text{dest})$ και $\text{SimBetUtil}_m(\text{dest})$ ▪ αν $\text{SimBetUtil}_m(\text{dest}) < \text{SimBetUtil}_n(\text{dest})$ ζητά από τον m να του αποστείλει τα πακέτα (message request) 		<ul style="list-style-type: none"> ▪ για κάθε προορισμό που περιέχεται στο summary vector, υπολογίζει $\text{SimBetUtil}_n(\text{dest})$ και $\text{SimBetUtil}_m(\text{dest})$ ▪ υπολογίζει το $\text{SimBetUtil}(\text{dest})$ όλων των κόμβων ανήκουν στο partition του και επιλέγει το καλύτερο ▪ για κάθε προορισμό για τον οποίο $\text{SimBetUtil}_m(\text{dest}) < \text{BestSimBetUtil}(\text{dest})$, ζητά από τον m να του αποστείλει τα πακέτα ορίζοντας τον καλύτερο μεταφορέα του μηνύματος (message request)

Πίνακας 3.6 Ενέργειες αλγορίθμων μετά τη λήψη Message Request

Όταν ένας κόμβος n λάβει message request από τον κόμβο m		
SIMBET	OLSR	SASI
<ul style="list-style-type: none"> ▪ προωθεί στον m όσα πακέτα έχει αποθηκευμένα για τους προορισμούς που περιέχονται στο message request 		<ul style="list-style-type: none"> ▪ προωθεί στον m όσα πακέτα έχει αποθηκευμένα για τους προορισμούς που περιέχονται στο message request

Πίνακας 3.7 Ενέργειες αλγορίθμων μετά τη λήψη μηνύματος δεδομένων

Όταν ένας κόμβος n λάβει ένα πακέτο δεδομένων από τον κόμβο m		
SIMBET	OLSR	SASI
<ul style="list-style-type: none"> ▪ αποθηκεύει το πακέτο 	<ul style="list-style-type: none"> ▪ αναζητεί στον πίνακα δρομολόγησης μία διαδρομή για τον παραλήπτη του πακέτου ▪ αν υπάρχει διαδρομή το προωθεί διαφορετικά το απορρίπτει 	<ul style="list-style-type: none"> ▪ αναζητεί στον πίνακα δρομολόγησης μία διαδρομή για τον παραλήπτη του πακέτου ▪ αν υπάρχει διαδρομή το προωθεί, διαφορετικά ελέγχει αν έχει οριστεί καλύτερος μεταφορέας του μηνύματος ▪ αν έχει οριστεί το προωθεί χρησιμοποιώντας τον πίνακα δρομολόγησης, διαφορετικά αποθηκεύει το πακέτο

3.3. Παραλλαγές του Αλγορίθμου SASI

Έχει μελετηθεί και μία παραλλαγή του προτεινόμενου αλγορίθμου SASI. Η διαφορά έγκειται στις πληροφορίες που προσθέτουμε το TC μήνυμα. Αντί για τις τιμές των μετρικών *betweenness* και *similarity* που έχει υπολογίσει τοπικά ο κόμβος με τη βοήθεια του *ego network*, σε αυτή την περίπτωση συμπεριλαμβάνουμε όλες τις επαφές του κόμβου. Λαμβάνοντας, λοιπόν, ένα TC μήνυμα ο κόμβος γνωρίζει πλέον και όλες τις επαφές του αποστολέα του TC. Με αυτό τον τρόπο μπορεί να κατασκευάσει όλο τον κοινωνικό γράφο του *partition* στο οποίο ανήκει.

Η έννοια του *ego network* χρησιμοποιείται επειδή για τον υπολογισμό κάποιων από τις μετρικές κοινωνικής δικτύωσης, όπως το *Betweenness centrality* απαιτείται πληροφορία για όλο τον κοινωνικό γράφο, επομένως γίνεται δύσκολος και χρονοβόρος ο υπολογισμός τους όσο αυξάνει το μέγεθος του δικτύου. Πλέον όμως στον αλγόριθμο μας, έχουμε μία αναπαράσταση όλου του κοινωνικού γράφου και

μπορούμε να υπολογίσουμε το Betweenness centrality χωρίς τη χρήση του ego network.

Διατηρώντας όλες τις υπόλοιπες λειτουργίες του αλγορίθμου ίδιες και διαφοροποιώντας μόνο το περιεχόμενο των μηνυμάτων TC και τον τρόπο υπολογισμού του betweenness, παρατηρούνται ελάχιστες διαφορές στα πειραματικά αποτελέσματα ενισχύοντας την άποψη ότι η κατάταξη των sociocentric betweenness και ego betweenness είναι ίδια σε μεγάλο ποσοστό [11]. Δεδομένου του υπολογιστικού κόστους που προκαλεί ο υπολογισμός του sociocentric betweenness σε σχέση με το ego betweenness, για τον υπολογισμό των συντομότερων μονοπατιών, αλλά και τον όγκο δεδομένων που προστίθενται στο TC, όταν προστίθενται οι τιμές των μετρικών το μέγεθος είναι σταθερό ενώ όταν προστίθενται οι επαφές το μέγεθος εξαρτάται από το δίκτυο και μπορεί να είναι πολύ μεγάλο, η συγκεκριμένη εκδοχή χρησιμοποιείται ως παραλλαγή και όχι ως ο κύριος αλγόριθμος.

ΚΕΦΑΛΑΙΟ 4. ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

4.1 Περιβάλλον Προσομοίωσης- Εργαλεία

4.2 Κίνηση Κόμβων

4.3 Αποτελέσματα

4.1. Περιβάλλον Προσομοίωσης- Εργαλεία

Για την αξιολόγηση του προτεινόμενου αλγορίθμου χρησιμοποιήθηκε ο προσομοιωτής Network Simulator (NS-2), ευρέως διαδεδομένος για πειράματα τόσο σε ενσύρματα όσο και σε ασύρματα δίκτυα. Ειδικά για τα ασύρματα δίκτυα υποστηρίζει την ad-hoc δρομολόγηση. Στον NS-2 είναι ενσωματωμένα αρκετά γνωστά πρωτόκολλα αλλά δίνει και τη δυνατότητα στον χρήστη να ενσωματώσει τα δικά του πρωτόκολλα ή να μεταβάλλει τα ήδη υπάρχοντα.

Ο προσομοιωτής NS-2 είναι ένας αντικειμενοστρεφής προσομοιωτής δικτύων προσανατολισμένος από γεγονότα, που χρησιμοποιείται για να εκτελούνται διαφορετικά σενάρια που εισάγει ο χρήστης. Ο χρήστης γράφει ένα σενάριο που καθορίζει τις παραμέτρους του δικτύου και το πρωτόκολλο που θα χρησιμοποιήσει. Το σενάριο έπειτα αξιοποιεί ο NS κατά τη διάρκεια της προσομοίωσης. Το αποτέλεσμα αυτής της προσομοίωσης είναι ένα αρχείο καταγραφής (trace file) που μπορεί έπειτα να χρησιμοποιηθεί για την επεξεργασία των αποτελεσμάτων της προσομοίωσης.

Για να εκτελέσει, λοιπόν, ο χρήστης μία προσομοίωση εισάγει το simulation script που αποτελεί την καρδιά της προσομοίωσης αφού καθορίζει τις παραμέτρους της προσομοίωσης όπως ο αριθμός των κόμβων, η περιοχή στην οποία κινούνται και ο

συνολικός χρόνος της προσομοίωσης. Επίσης, χρησιμοποιεί το αρχείο που περιγράφει την κίνηση των κόμβων (mobility scenario) και το αρχείο που ορίζει τη δημιουργία των πακέτων δεδομένων (traffic scenario).

Στα πειράματα που εκτελέσαμε ο συνολικός χρόνος προσομοίωσης είναι πέντε ημέρες (432000 sec), ο αριθμός των κόμβων είναι 50 και η περιοχή στην οποία κινούνται 5 επί 5 χιλιόμετρα. Ο αριθμός των πακέτων που παράγονται στη διάρκεια της προσομοίωσης είναι σταθερός και ίσος με 2450. Κάθε κόμβος παράγει ένα πακέτο για κάθε άλλο κόμβο σε τυχαία χρονική στιγμή. Εκτελέσαμε κάθε πείραμα πέντε φορές διαφοροποιώντας τη χρονική στιγμή κατά την οποία οι κόμβοι παράγουν τα πακέτα και για την αξιολόγηση του αλγορίθμου χρησιμοποιήσαμε το μέσο όρο των πέντε πειραμάτων.

4.2. Κίνηση Κόμβων

Σκοπός των πειραμάτων ήταν η δοκιμή του αλγορίθμου σε δίκτυα όπου οι κόμβοι δημιουργούν σημεία συγκέντρωσης. Γι' αυτό το λόγο χρησιμοποιήθηκαν συνθετικά μοντέλα κίνησης.

Για τη δημιουργία των αρχείων κινητικότητας χρησιμοποιήθηκε ο αλγόριθμος των Micro Musolesi και Cecilia Mascolo [13]. Ο αλγόριθμος αυτός μπορεί να πάρει ως είσοδο τον κοινωνικό γράφο ενός δικτύου ή να το κατασκευάσει χρησιμοποιώντας παραμέτρους που δίνει ο χρήστης. Το μοντέλο επιτρέπει οι κόμβοι να ομαδοποιούνται με τρόπο που βασίζεται στις μεταξύ τους κοινωνικές σχέσεις. Ο αλγόριθμος χρησιμοποιεί ένα πίνακα που ονομάζεται Interaction Matrix, ο οποίος αναπαριστά τις κοινωνικές σχέσεις των κόμβων χρησιμοποιώντας τιμές στο πεδίο [0,1]. Ο Interaction Matrix χρησιμοποιείται για την κατασκευή του Connectivity Matrix. Ελέγχεται κάθε στοιχείο του πίνακα Interaction Matrix. Αν η τιμή του ξεπερνά ένα προκαθορισμένο κατώφλι, ο αλγόριθμος θέτει την τιμή 1 στην αντίστοιχη θέση του Connectivity Matrix, διαφορετικά θέτει 0.

Η ομαδοποίηση των κόμβων και η τοποθέτησή τους στο χώρο καθορίζονται είτε από τον αλγόριθμο Caveman είτε από τον Newman and Grivan. Σύμφωνα με τον

αλγόριθμο Caveman το δίκτυο ξεκινά από K πλήρως συνδεδεμένους γράφους. Κάθε ακμή του αρχικού δικτύου “δείχνει” προς έναν κόμβο σε διαφορετικό δίκτυο με πιθανότητα p (rewiringProb). Ο αλγόριθμος Newman and Grivan χρησιμοποιεί το Connectivity Matrix για να σχηματίσει ομάδες κόμβων που χαρακτηρίζονται από σημαντική συνδεσιμότητα. Θεωρεί ότι η κοινωνική σχέση είναι ισχυρή εάν η τιμή του interaction matrix είναι μεγαλύτερη από ένα κατώφλι (threshold).

Ο χώρος τον οποίο μπορούν να κινούνται οι κόμβοι χωρίζεται σε τετράγωνα (squares) και η κίνηση των κόμβων εξαρτάται από την κοινωνική έλξη (social attractivity) που ασκείται. Η κοινωνική έλξη του τετραγώνου $S_{p,q}$ προς τον κόμβο i ορίζεται ως

$$SA_{p,qi} = \frac{\sum_{j \in C_{S_{p,q}}} m_{i,j}}{w} \quad \text{Εξ. 4.1}$$

όπου $m_{i,j}$: η κοινωνική σχέση των i και j και w : αριθμός των κόμβων που βρίσκονται στο τετράγωνο.

Η κίνηση των κόμβων είναι είτε ντετερμινιστική, δηλαδή επιλέγεται ο προορισμός του κόμβου εντός του τετραγώνου με το μεγαλύτερο social attractivity, είτε πιθανοκρατική, δηλαδή η πιθανότητα να επιλεγεί ως επόμενος προορισμός το τετράγωνο S_{pq} είναι

$$P(s = S_{p,qi}) = \frac{SA_{p,qi} + d}{\sum_{j=1}^{p \times q} (SA_{p,qj} + d)} \quad \text{Εξ. 4.2}$$

όπου d : τυχαία τιμή.

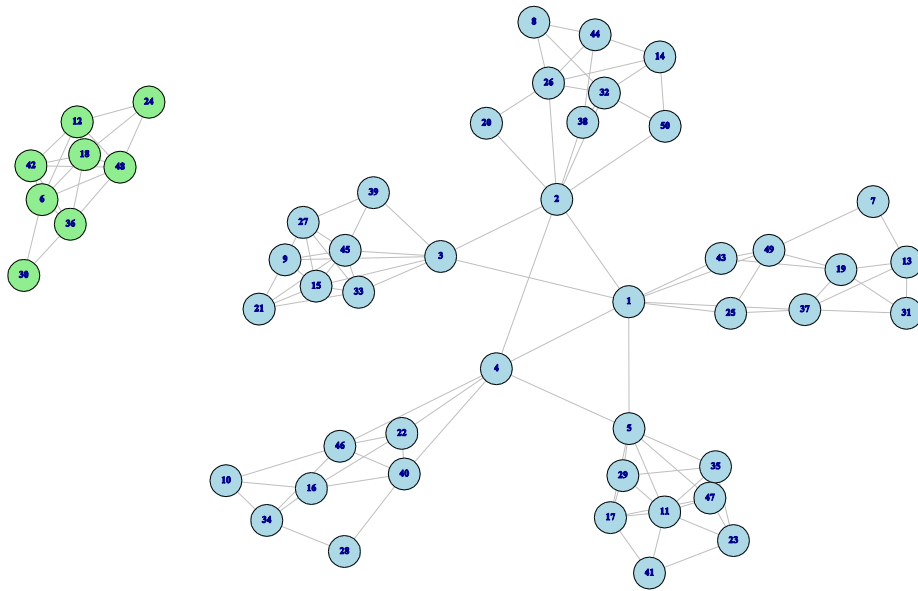
Όπως αναφέραμε, ο συνολικός χρόνος προσομοίωσης είναι πέντε ημέρες (432000 sec), ο αριθμός των κόμβων είναι 50 και η περιοχή στην οποία κινούνταν 5 επί 5 km. Για την παραγωγή διαφορετικών σεναρίων κίνησης χρησιμοποιήθηκαν οι παράμετροι που φαίνονται στον παρακάτω πίνακα.

Πίνακας 4.1 Παράμετροι δημιουργίας σεναρίων κίνησης

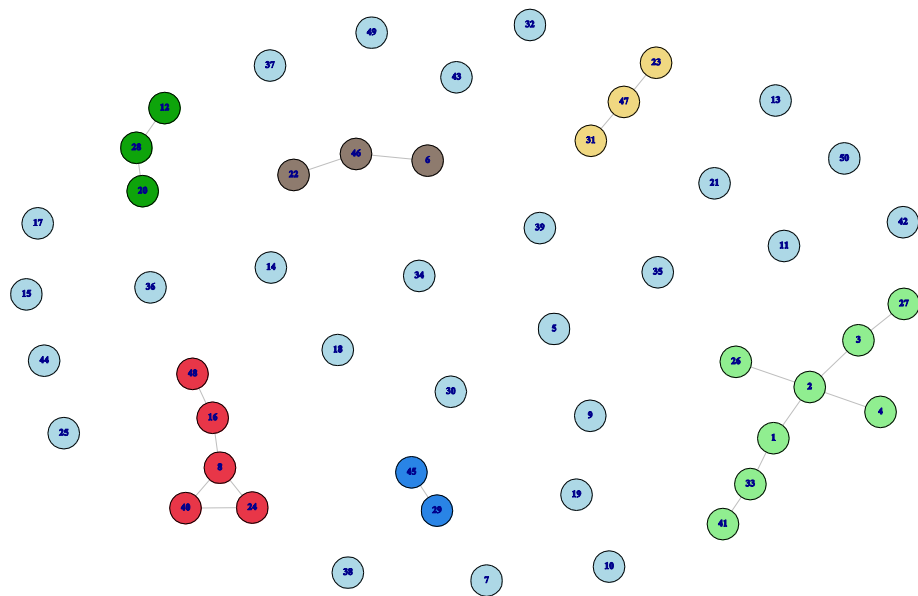
	A	B	Γ	Δ	E	ΣΤ
threshold	0,5	0,7	0,5	0,7	0,5	0,5
squares	100	1225	625	625	625	625
minSpeed	1	1	1	1	1	1
maxSpeed	3	4	5	5	5	5
Rewiring Prob	0.4	0.7	0.4	0.7	0.2	0.2
groups	4	8	4	8	8	4
αρχικές θέσεις	Caveman	Newman and Grivan	Caveman	Newman and Grivan	Newman and Grivan	Newman and Grivan
κίνηση κόμβων	deterministic	deterministic	deterministic	deterministic	probabilistic	probabilistic

Προκειμένου να γίνει κατανοητή η διαφορά των σεναρίων κίνησης που προκύπτουν από τις διαφορετικές παραμέτρους, παρουσιάζονται ενδεικτικά για τα δύο πρώτα σεναρία κάποια στατιστικά στοιχεία σχετικά με τον αριθμό των μοναδικών επαφών αλλά και των μη μοναδικών που έχει κάθε κόμβος στο τέλος της προσομοίωσης. Επιπλέον, παρατίθενται μία γραφική αναπαράσταση του αρχικού κοινωνικού γράφου για κάθε σενάριο κίνησης.

Βλέποντας τα σχήματα 4.1 και 4.2 παρατηρούμε ότι ο αρχικός κοινωνικός γράφος των δύο σεναρίων παρουσιάζει σημαντικές διαφορές. Το σενάριο κίνησης A παρουσιάζει διακριτές ομάδες κόμβων, μία από αυτές τελείως απομονωμένη ενώ οι υπόλοιπες συνδέονται μεταξύ τους με κάποιους κόμβους- γέφυρες. Αντίθετα, το σενάριο κίνησης B παρουσιάζει λίγες μικρές ομάδες χωρίς καμία κοινωνική σχέση μεταξύ τους και με τους υπόλοιπους κόμβους να είναι ανεξάρτητοι και διάσπαρτοι στο χώρο.

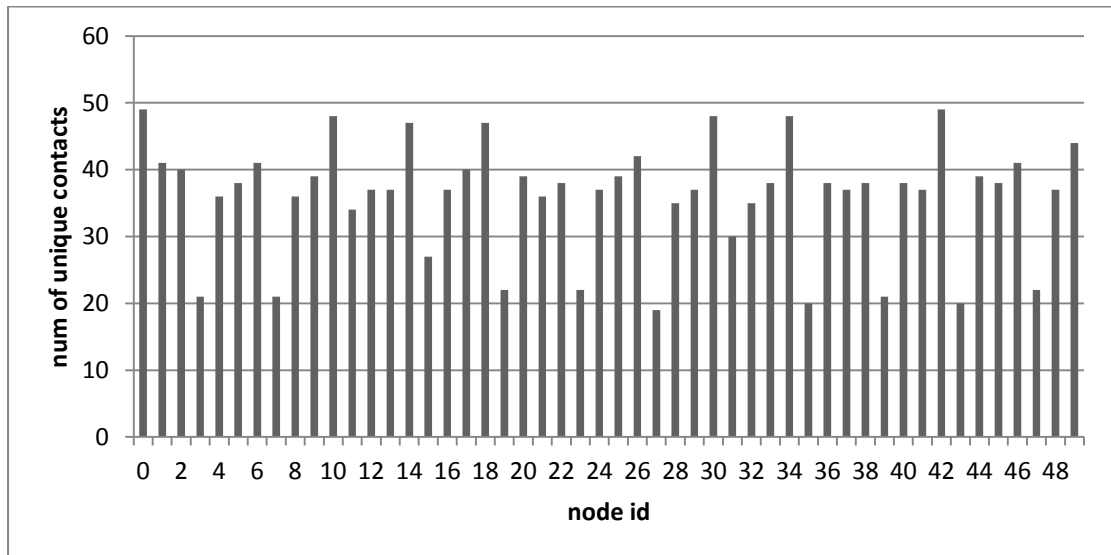


Σχήμα 4.1 Ο κοινωνικός γράφος του σεναρίου κίνησης A

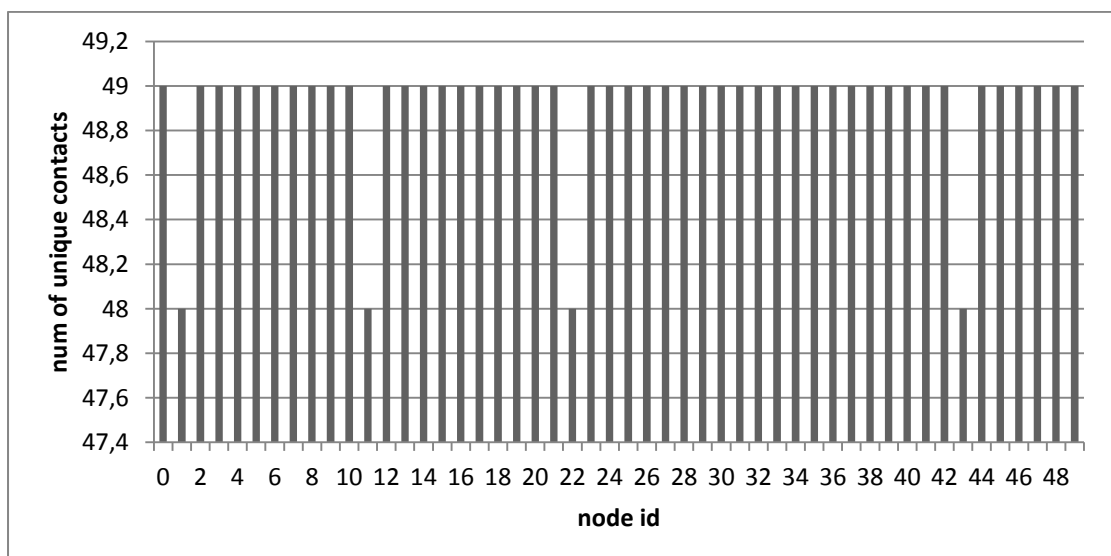


Σχήμα 4.2 Ο κοινωνικός γράφος του σεναρίου κίνησης B

Ο αριθμός των μοναδικών επαφών που έχει κάθε κόμβος υπολογίζεται στο τέλος της προσομοίωσης. Παρατηρώντας το σχήμα 4.3 βλέπουμε ότι στο σενάριο κίνησης A υπάρχουν αρκετές διαβαθμίσεις. Αντίθετα στο σενάριο κίνησης B στο σχήμα 4.4, εκτός ελαχίστων εξαιρέσεων, όλοι οι κόμβοι συναντιούνται με όλους τους άλλους έστω και μία φορά κατά τη διάρκεια των πέντε ημερών της προσομοίωσης.

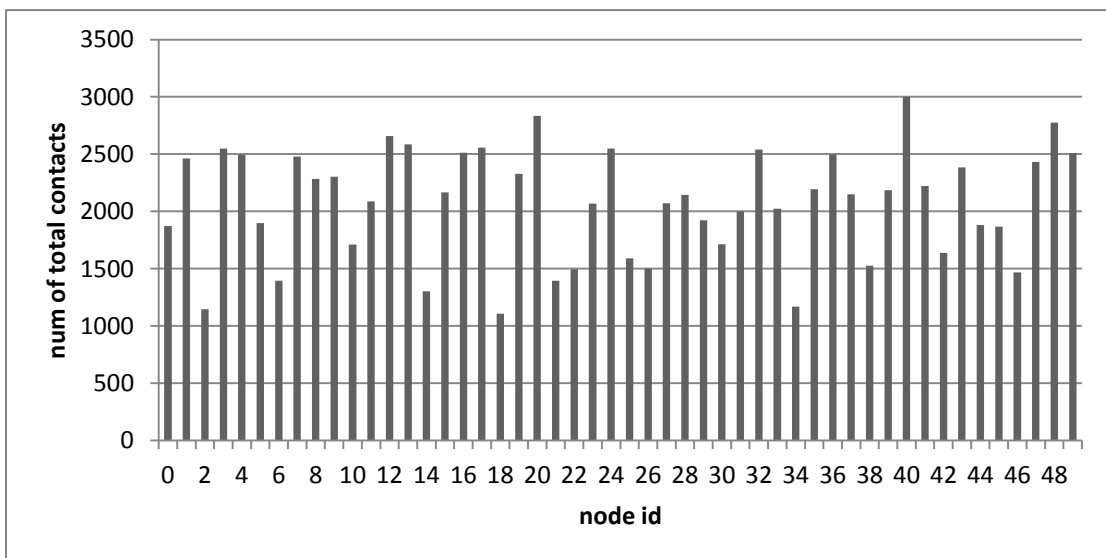


Σχήμα 4.3 Αριθμός μοναδικών επαφών ανά κόμβο (Σενάριο κίνησης A)

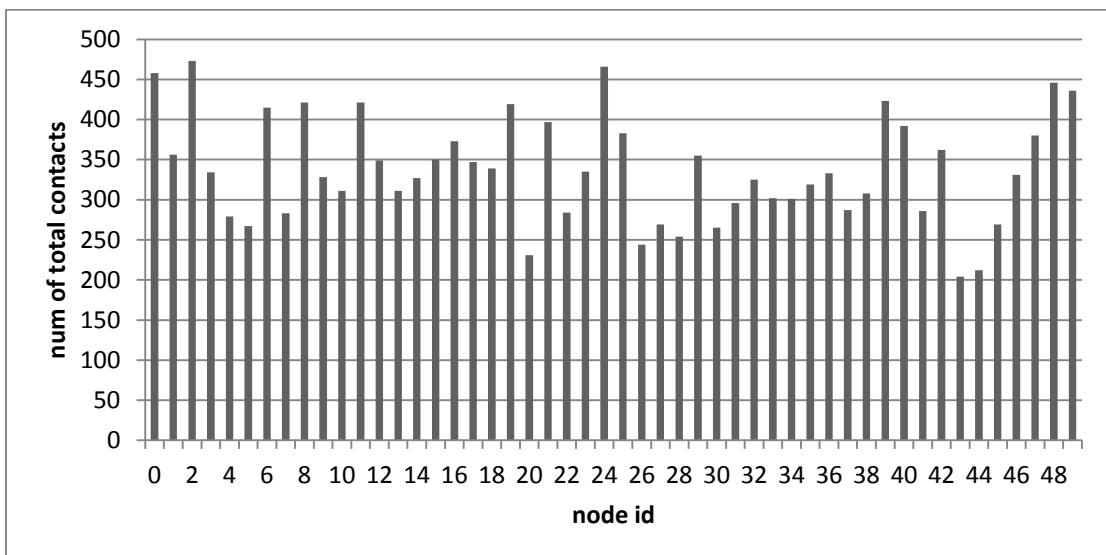


Σχήμα 4.4 Αριθμός μοναδικών επαφών ανά κόμβο (Σενάριο κίνησης B)

Αντίθετα με ότι ίσως περιμέναμε, ο συνολικός αριθμός των επαφών ανά κόμβο είναι πολύ μεγαλύτερος κατά μέσο όρο στο σενάριο κίνησης Α με μέγιστη τιμή τις 3000 επαφές σε σχέση με το σενάριο Β με μέγιστη τιμή τις 473 επαφές. Από αυτό μπορούμε να συμπεράνουμε ότι στην πρώτη περίπτωση υπάρχει αυξημένη κινητικότητα των κόμβων προκαλώντας συνεχείς αλλαγές στις συνδέσεις τους.

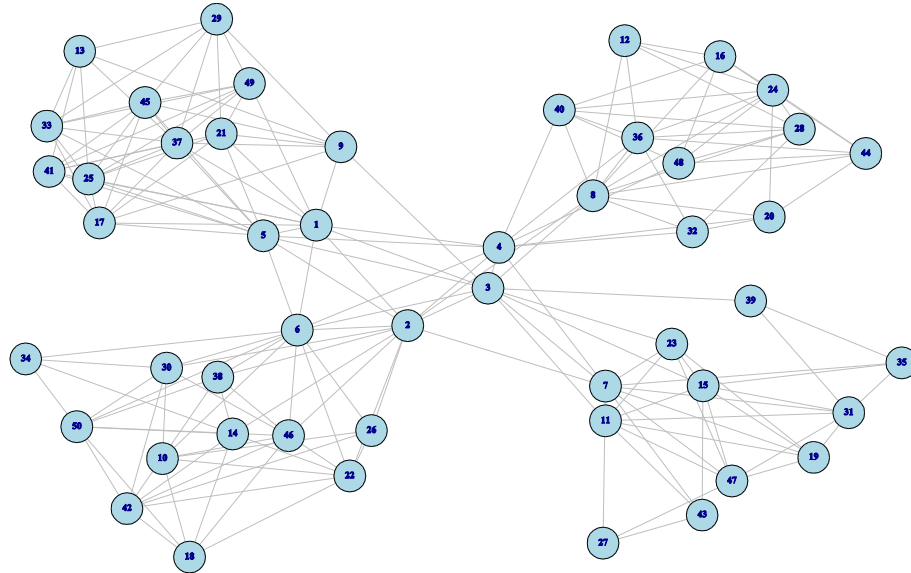


Σχήμα 4.5 Συνολικός αριθμός επαφών ανά κόμβο (Σενάριο κίνησης Α)

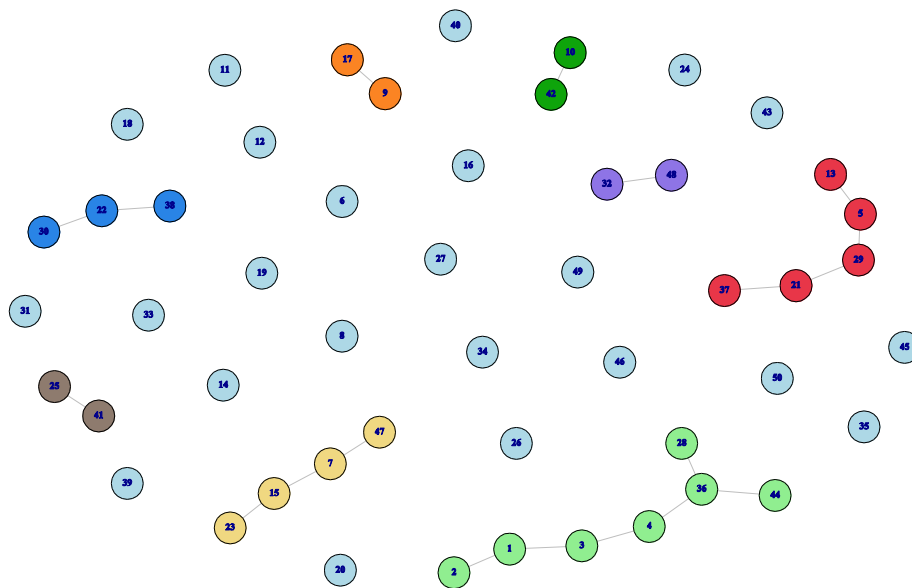


Σχήμα 4.6 Συνολικός αριθμός επαφών ανά κόμβο (Σενάριο κίνησης Β)

Στο κοινωνικό γράφο του σεναρίου κίνησης Γ, στο σχήμα 4.7, παρατηρούμε ότι οι κόμβοι σχηματίζουν τέσσερις μεγάλες ομάδες, εντός των οποίων υπάρχουν αρκετές συνδέσεις. Τις ομάδες αυτές ενώνουν μερικοί κόμβοι που παίζουν το ρόλο της γέφυρας. Αντίθετα, το τέταρτο σενάριο κίνησης παρουσιάζει ομοιότητες με το δεύτερο. Υπάρχουν μερικές μικρές ομάδες κόμβων που όμως δεν έχουν κοινωνική σχέση μεταξύ τους. Οι υπόλοιποι κόμβοι είναι διάσπαρτοι στο χώρο.

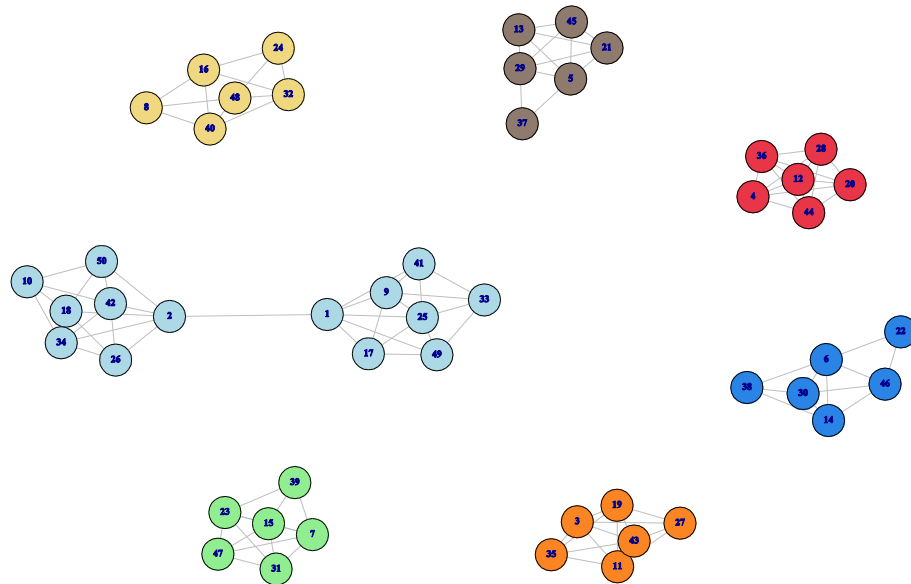


Σχήμα 4.7 Ο κοινωνικός γράφος του σεναρίου κίνησης Γ



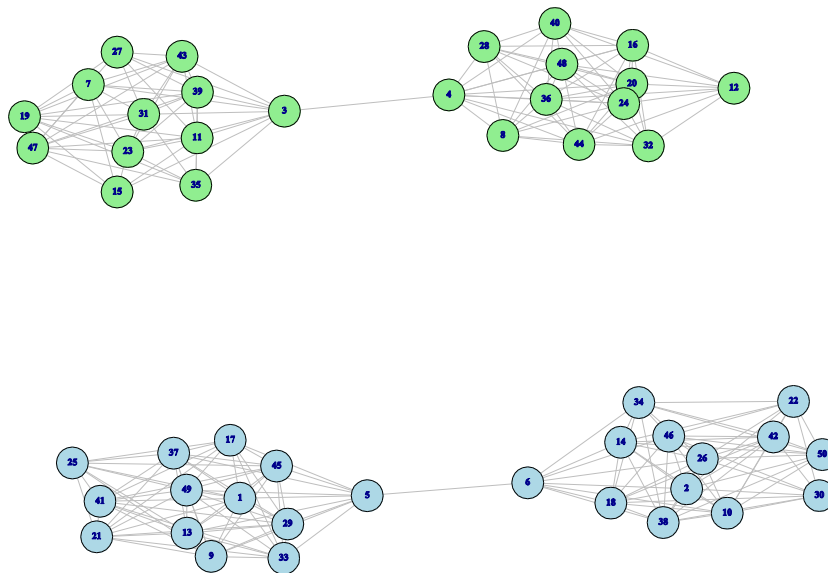
Σχήμα 4.8 Ο κοινωνικός γράφος του σεναρίου κίνησης Δ

Στον αρχικό κοινωνικό γράφο του πέμπτου σεναρίου κίνησης που παρουσιάζεται στο σχήμα 4.9, παρατηρούμε ότι δεν υπάρχει κανένας απομονωμένος κόμβος. Βλέπουμε διακριτές ομάδες κόμβων με κάθε κόμβο να παρουσιάζει κοινωνικές σχέσεις με μία μικρή ομάδα. Μόνο δύο από αυτές τις ομάδες φαίνεται να συσχετίζονται μεταξύ τους. Οι υπόλοιπες είναι αποκομμένες από τις άλλες ομάδες του δικτύου.



Σχήμα 4.9 Ο κοινωνικός γράφος του σεναρίου κίνησης E

Στο σενάριο κίνησης ΣΤ, σχήμα 4.10, οι κόμβοι σχηματίζουν τέσσερις ομάδες. Υπάρχει συσχέτιση των ομάδων σε ζευγάρια. Διαφορετικά, θα μπορούσαμε να πούμε ότι ο κοινωνικός γράφος χωρίζεται σε δύο μεγάλες ομάδες κόμβων, κάθε μία από τις οποίες αποτελείται από δύο μικρότερες. Τις μικρότερες αυτές ομάδες, ενώνουν μεταξύ τους δύο κόμβοι που κατέχουν στρατηγικές θέσεις στο δίκτυο. Αναμένουμε οι κόμβοι αυτοί να έχουν αυξημένες τιμές *Betweenness centrality*.



Σχήμα 4.10 Ο κοινωνικός γράφος του σεναρίου κίνησης ΣΤ

4.3. Αποτελέσματα

Στην παράγραφο αυτή θα παρουσιάσουμε τα πειραματικά αποτελέσματα της προσομοίωσης. Στα πειράματα συγκρίνουμε τον αλγόριθμο SASI με τους αλγορίθμους Simbet, OLSR, DSDV, Direct και CAR. Στον κλασικό αλγόριθμο OLSR έχει προστεθεί μία δομή που αποθηκεύονται τα πακέτα όταν δεν υπάρχει διαθέσιμη διαδρομή προς τον παραλήπτη αντί να απορρίπτονται όπως ορίζει το πρωτόκολλο. Τα πακέτα ανακτώνται όταν υπάρξει διαθέσιμη διαδρομή. Ο αλγόριθμος Direct είναι ένας απλός DTN αλγόριθμος σύμφωνα με τον οποίο ο κόμβος προωθεί το πακέτο μόνο στον παραλήπτη, δεν υπάρχουν δηλαδή ενδιάμεσοι κόμβοι.

Οι μετρικές που χρησιμοποιήθηκαν για τη σύγκρισή των πρωτοκόλλων είναι οι εξής:

- *Ποσοστό επιτυχούς παράδοσης (Delivery Ratio)*: ορίζεται ως το ποσοστό των πακέτων δεδομένων που παραδόθηκαν επιτυχώς στους παραλήπτες τους σε σχέση με το συνολικό πλήθος των πακέτων που δημιουργήθηκαν στη διάρκεια της προσομοίωσης. Η μετρική αυτή αποτελεί κύριο κριτήριο αξιολόγησης των αλγορίθμων. Όσο μεγαλύτερο είναι το ποσοστό αυτό τόσο

μεγαλύτερη είναι η ικανότητα του αλγορίθμου να παραδώσει τα πακέτα στους προορισμούς τους.

- *Μέση καθυστέρησης παράδοσης (Mean Delay)*: δείχνει το μέσο χρόνο που απαιτείται για να παραδοθεί ένα πακέτο στον προορισμό του. Στόχος είναι να επιτευχθεί όσο το δυνατόν μικρότερη μέση καθυστέρηση.
- *Μέσο πλήθος αλμάτων ανά πακέτο (Hops per packet)*: η μετρική αυτή είναι η μέση τιμή του αριθμού των ενδιάμεσων κόμβων από τους οποίους χρειάστηκε να περάσει το πακέτο για να φτάσει στον προορισμό του.
- *Αριθμός προωθήσεων (Number of forwards)*: ορίζεται ως το άθροισμα των πακέτων που προώθησαν οι κόμβοι στη διάρκεια της προσομοίωσης. Ο αριθμός των προωθήσεων υπολογίζεται στο σύνολο των πακέτων ανεξάρτητα με το αν παραδόθηκαν τελικώς στον παραλήπτη ενώ το μέσο πλήθος αλμάτων αφορά μόνο τα παραδοτέα πακέτα.

Παρακάτω, στον πίνακα 4.2, παρουσιάζονται συνοπτικά τα αποτελέσματα των πειραμάτων. Οι δύο ίσως σημαντικότερες μετρικές αξιολόγησης είναι το ποσοστό επιτυχούς παράδοσης και η μέση καθυστέρηση παράδοσης. Οι τιμές αφορούν το μέσο όρο των πέντε πειραμάτων που εκτελέσαμε για κάθε σενάριο κίνησης διαφοροποιώντας τη χρονική στιγμή που παράγονται τα πακέτα. Παρατηρούμε ότι ο αλγόριθμος SASI επιτυγχάνει υψηλότερο ποσοστό παράδοσης για όλα τα σενάρια κίνησης. Παράλληλα, βέβαια, εμφανίζει και μεγάλη μέση καθυστέρηση παράδοσης, γεγονός που δικαιολογείται δεδομένου ότι παραδίδει επιτυχώς περισσότερα πακέτα

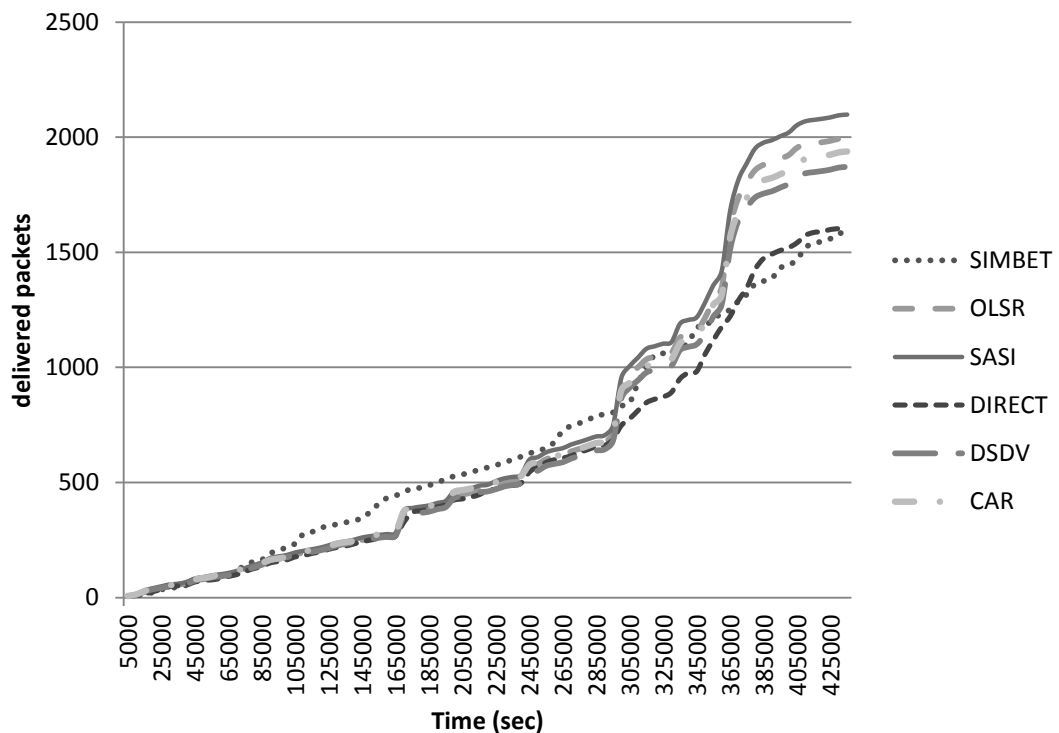
Πίνακας 4.2 Ποσοστό επιτυχούς παράδοσης και Μέση καθυστέρηση για όλα τα σενάρια κίνησης

		A	B	Γ	Δ	E	ΣΤ
Simbet	Delivery Ratio	62,57	73,53	76,91	84,20	84,99	90,99
	Mean Delay	44527	44070	37631	32537	36508	36156
OLSR	Delivery Ratio	80,86	88,55	82,04	90,65	90,35	91,93
	Mean Delay	83497	38544	54452	27426	27666	25885
SASI	Delivery Ratio	85,39	90,20	87,76	93,13	92,87	93,93
	Mean Delay	85132	38723	56731	27514	27877	26077
Direct	Delivery Ratio	65,81	85,33	85,09	89,40	88,41	89,60
	Mean Delay	75349	51741	64195	38595	41134	39832
DSDV	Delivery Ratio	75,99	86,25	76,26	87,98	87,35	89,30
	Mean Delay	81871	38391	52985	27140	27212	25335
CAR	Delivery Ratio	79,18	89,04	83,60	91,47	91,44	92,54
	Mean Delay	78071	37956	51907	26811	27020	25305

4.3.1. Σενάριο Κίνησης A

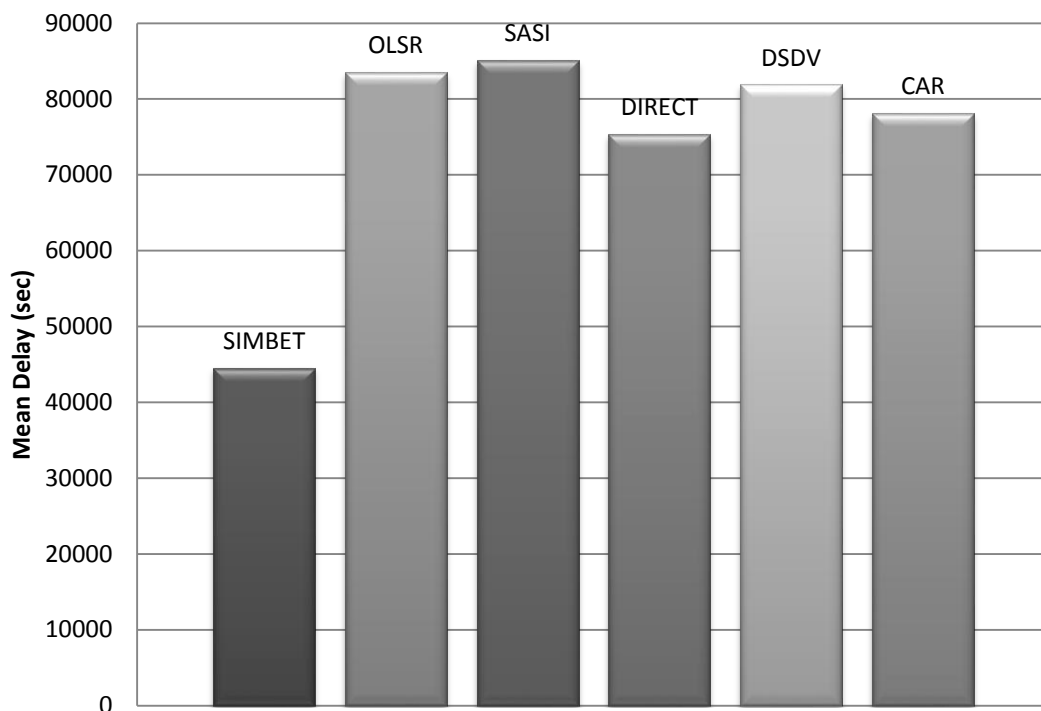
Στο σχήμα 4.11 βλέπουμε πως διαμορφώνεται το πλήθος των μηνυμάτων που παραδόθηκαν επιτυχώς σε συνάρτηση με το χρόνο για όλη τη διάρκεια της προσομοίωσης. Τα αποτελέσματα αφορούν το πρώτο από τα πέντε πειράματα που εκτελέσαμε. Όπως είναι αναμενόμενο το πλήθος των μηνυμάτων που παραδόθηκαν αυξάνεται με την πάροδο του χρόνου. Αυτό συμβαίνει κυρίως λόγω της κινητικότητας των κόμβων. Τα πακέτα προωθούνται σε όλο και καλύτερους μεταφορείς αυξάνοντας σε κάθε άλμα τις πιθανότητες να φτάσουν στον παραλήπτη. Επιπλέον, όταν ένας κόμβος διατηρεί στη μνήμη του ένα πακέτο, ακόμη και αν δεν

συναντήσει καλύτερο μεταφορέα είναι πιθανό να συναντήσει τον παραλήπτη, δεδομένου ότι η περιοχή που κινούνται οι κόμβοι είναι περιορισμένη στα 5 x 5 km. Ακόμη, όσον αφορά τους αλγορίθμους που στηρίζονται σε μετρικές αξιολόγησης με την πάροδο του χρόνου, οι κόμβοι έχουν περισσότερες πληροφορίες για το δίκτυο και μπορούν να κάνουν πιο ακριβείς υπολογισμούς. Παρατηρούμε ότι οι αλγόριθμοι SASI, OLSR, CAR και DSDV έχουν σε μεγάλο ποσοστό παρόμοια συμπεριφορά κάτι που οφείλεται στη σύγχρονη λειτουργία τους και στη συνδεσιμότητα του δικτύου. Στη χρονική στιγμή 365000 sec οι αλγόριθμοι έχουν σχεδόν ίδιο πλήθος παραδοτέων πακέτων ενώ στη συνέχεια παρατηρείται μεγαλύτερη ανοδική πορεία του SASI παραδίδοντας τελικώς περισσότερα πακέτα σε σχέση με όλους τους υπόλοιπους. Η διαφορά του ποσοστού επιτυχούς παράδοσης είναι μόλις 4% σε σχέση με τον OLSR αλλά παρουσιάζει βελτίωση κατά 23% σε σχέση με τον Simbet. Συγκρίνοντας τον αλγόριθμο μας με τον επίσης υδρικό CAR η διαφορά του τελικού ποσοστού επιτυχούς παράδοσης φτάνει το 6,5%. Η διαφορά αυτή οφείλεται στο πλεονέκτημα του αλγορίθμου SASI κάθε κόμβος να γνωρίζει τις μετρικές όλων των υπόλοιπων κόμβων που ανήκουν στο partition του.



Σχήμα 4.11 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης A)

Στο σχήμα 4.12 παρουσιάζεται η μέση καθυστέρηση παράδοσης, επίσης για πρώτο από τα πέντε πειράματα. Ο αλγόριθμος μας παρουσιάζει τη μεγαλύτερη καθυστέρηση παράδοσης αλλά είναι κάτι που αναμένουμε δεδομένου ότι παραδίδει επιτυχώς περισσότερα πακέτα. Τα πακέτα που δεν προωθούνται μένουν για περισσότερο χρόνο αποθηκευμένα στη μνήμη των κόμβων αυξάνοντας την καθυστέρηση παράδοσης. Παρατηρώντας για παράδειγμα τις τιμές των μετρικών, διαπιστώνουμε ότι ο αλγόριθμος SASI που παραδίδει 6,5% περισσότερα πακέτα έχει και περίπου 7% μεγαλύτερη μέση καθυστέρηση σε σχέση με τον CAR. Καθυστέρηση που είναι ανεκτή για τα δεδομένα του δικτύου.



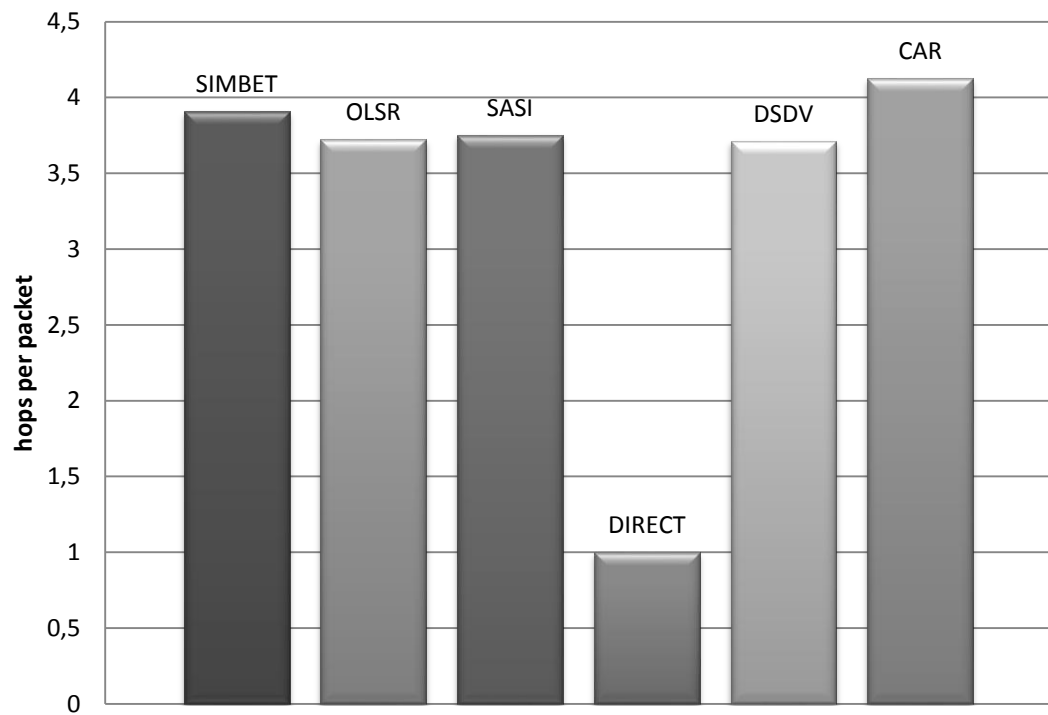
Σχήμα 4.12 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Α)

Στο σχήμα 4.13 βλέπουμε το μέσο αριθμό αλμάτων που χρειάστηκε να κάνει ένα πακέτο μέχρι να παραδοθεί στον παραλήπτη κατά μέση τιμή στο σύνολο των πέντε πειραμάτων που εκτελέσαμε. Ενδιαφέρον παρουσιάζει το γεγονός ότι στον αλγόριθμο

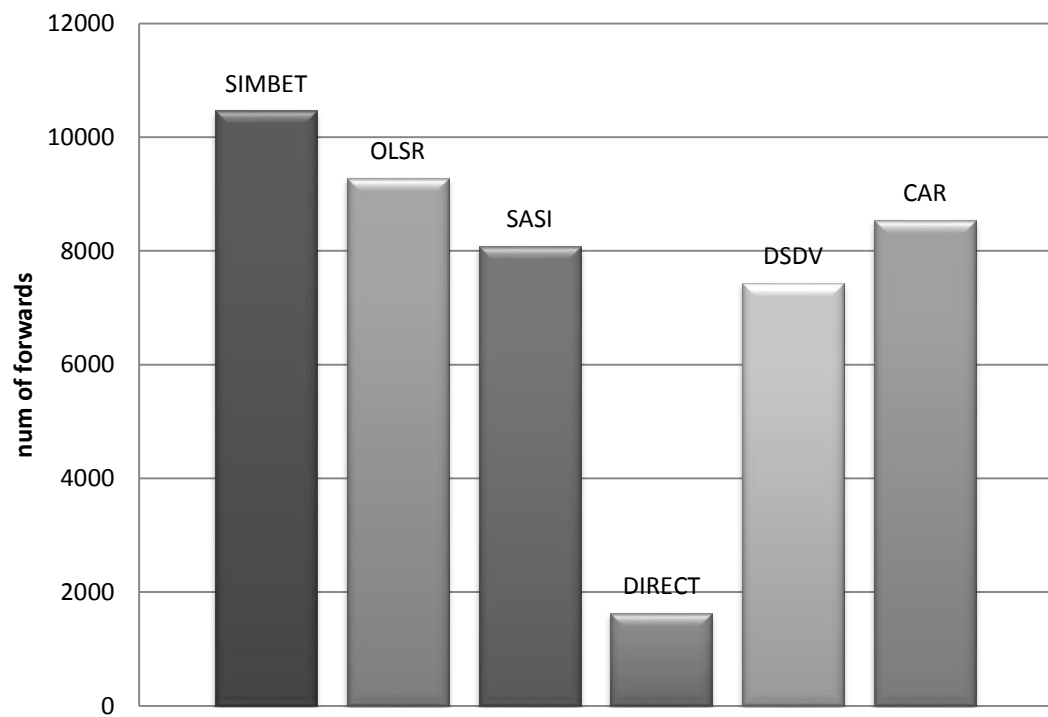
SASI τα πακέτα χρειάζεται να κάνουν λιγότερα άλματα σε σχέση με τον Simbet παρόλο που ο δεύτερος παραδίδει κατά 23% λιγότερα πακέτα. Παρόμοια συμπεριφορά παρατηρείται αν συγκρίνουμε τον αλγόριθμο μας με τον CAR στον οποίο παρατηρείται ο μεγαλύτερος μέσος αριθμός αλμάτων. Από αυτό μπορούμε να συμπεράνουμε είναι ότι η διαδρομή που ακολουθεί ένα πακέτο στον CAR προκειμένου να φτάσει στον προορισμό του, δεν είναι βέλτιστη, ακολουθώντας αρκετές φορές λανθασμένη διαδρομή. Αυτό οφείλεται στο γεγονός ότι οι κόμβοι δεν έχουν πλήρη εικόνα του δικτύου αλλά στηρίζονται στις μετρικές που ανταλλάσσουν οι άμεσοι γείτονές τους, κάτι που μπορεί να προκαλέσει λανθασμένες αποφάσεις δρομολόγησης.

Ο μέσος αριθμός αλμάτων συνδέεται με τον αριθμό προωθήσεων με τη διαφορά ότι αριθμός των προωθήσεων υπολογίζεται στο σύνολο των πακέτων ανεξάρτητα με το αν παραδόθηκαν τελικώς στον παραλήπτη. Στο σχήμα 4.14 παρουσιάζεται ο αριθμός των προωθήσεων που έκαναν όλοι οι κόμβοι στη διάρκεια της προσομοίωσης. Ο αλγόριθμος Direct όπως είναι αναμενόμενο έχει μέσο αριθμό αλμάτων ένα και τον μικρότερο αριθμό προωθήσεων (ίσο με τα πακέτα που παραδόθηκαν επιτυχώς). Εξαιρώντας τον Direct, ο αλγόριθμος SASI έχει το δεύτερο μικρότερο αριθμό προωθήσεων μετά από τον DSDV. Ο αλγόριθμος DSDV, όμως, παραδίδει σχεδόν 10% λιγότερα πακέτα σε σχέση με τον SASI. Από την άλλη πλευρά, βλέπουμε τον Simbet να έχει το μεγαλύτερο αριθμό προωθήσεων και τους υπόλοιπους αλγορίθμους να βρίσκονται στο ενδιάμεσο. Από τα αποτελέσματα μπορούμε να συμπεράνουμε ότι ο αλγόριθμος μας διαχειρίζεται καλύτερα τις επαφές των κόμβων προωθώντας τα πακέτα σε κόμβους που πραγματικά έχουν αυξημένες πιθανότητες επιτυχούς παράδοσης. Το γεγονός ότι κάθε κόμβος γνωρίζει τις μετρικές αξιολόγησης των υπολοίπων βοηθά να αξιολογήσει καλύτερα τον επόμενο παραλήπτη του πακέτου αποφεύγοντας άσκοπες μεταδόσεις που φορτώνουν το δίκτυο.

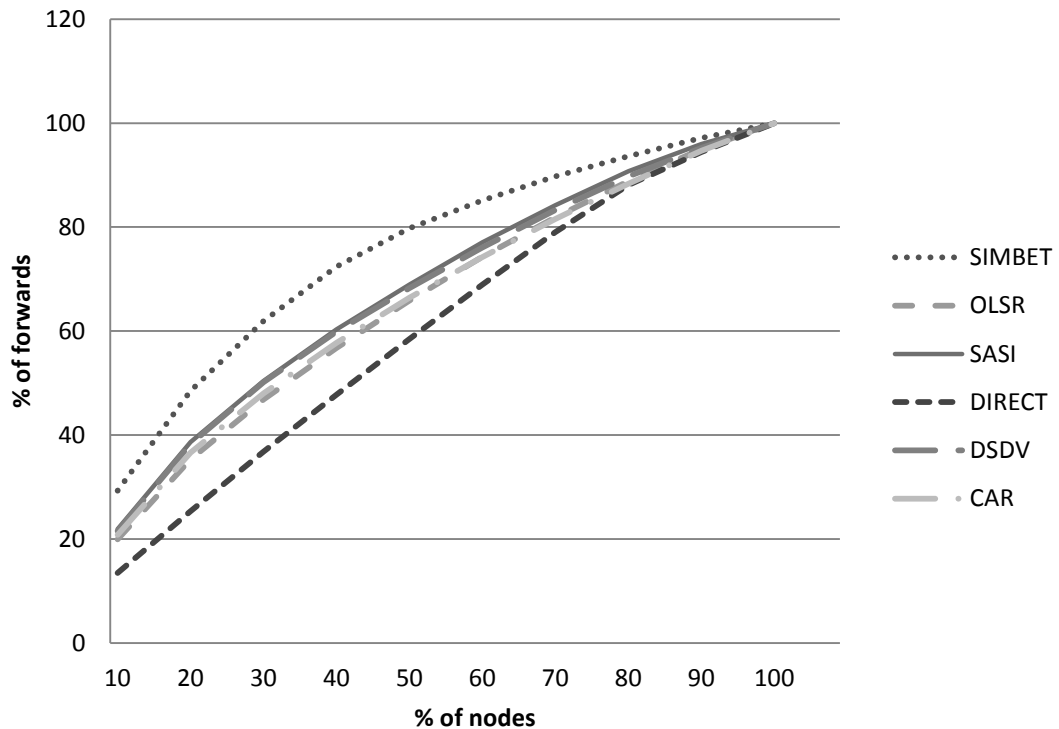
Τέλος, στο σχήμα 4.15 βλέπουμε πως κατανέμονται οι προωθήσεις στους κόμβους του δικτύου. Ο αλγόριθμος Direct παρουσιάζει την ιδανική συμπεριφορά κάτι που οφείλεται στο γεγονός ότι κάθε κόμβος παράγει ένα πακέτο για κάθε άλλο. Ο Simbet έχει τη μεγαλύτερη απόκλιση ενώ οι υπόλοιποι αλγόριθμοι έχουν παρόμοια συμπεριφορά.



Σχήμα 4.13 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης A)



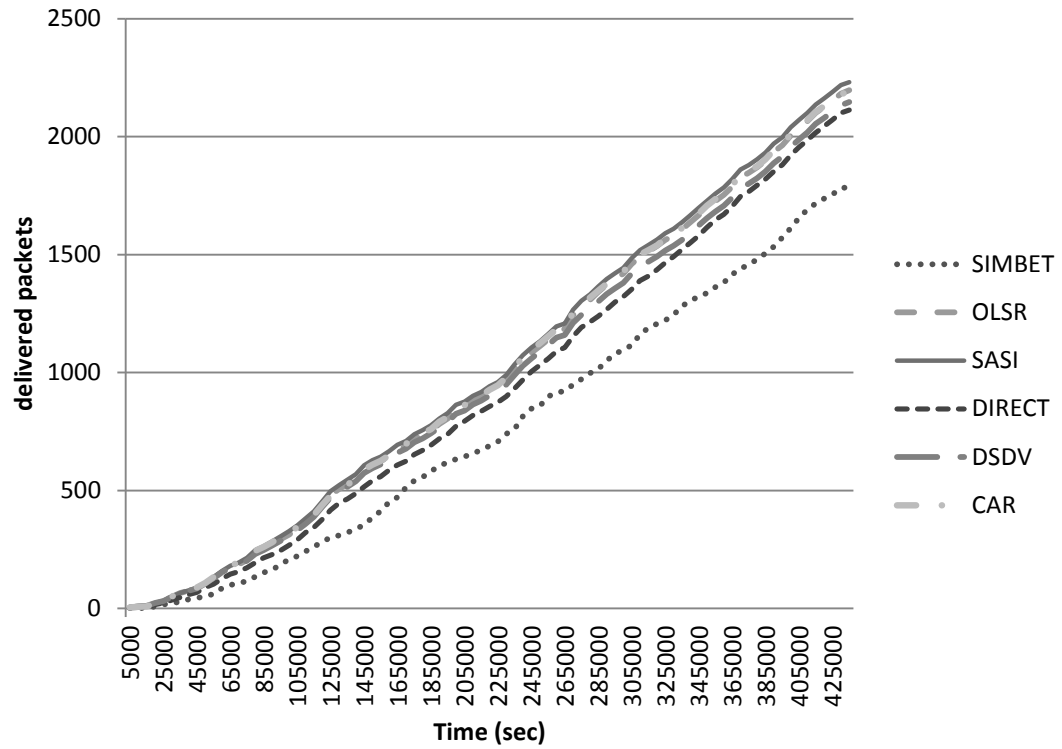
Σχήμα 4.14 Αριθμός προωθήσεων (Σενάριο κίνησης A)



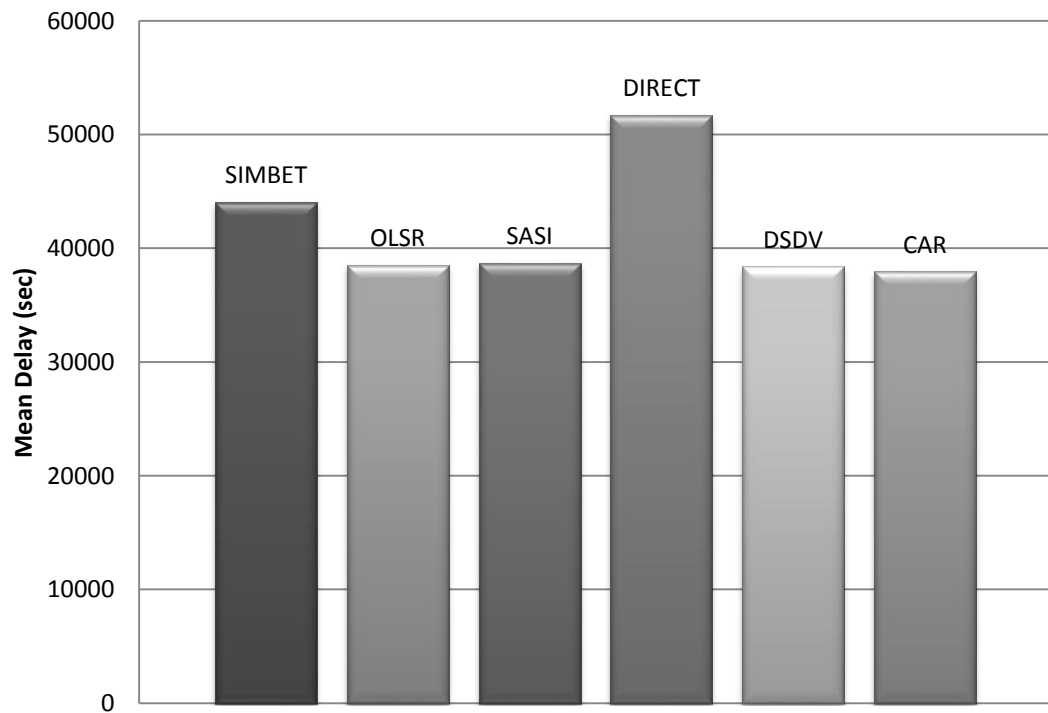
Σχήμα 4.15 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Α)

4.3.2. Σενάριο Κίνησης Β

Στο δεύτερο σενάριο κίνησης βλέπουμε ότι και εδώ ο αλγόριθμος μας παραδίδει το μεγαλύτερο πλήθος πακέτων (σχήμα 4.16), σχεδόν σε όλη τη διάρκεια της προσομοίωσης με μικρότερες όμως διαφορές από τους υπόλοιπους αλγορίθμους. Ο OLSR πετυχαίνει σχεδόν ίδιο ποσοστό επιτυχούς παράδοσης με τον CAR με μόλις λιγότερο από 1% διαφορά σε σχέση με τον SASI. Μεγάλη όμως παραμένει η υπεροχή του SASI σε σχέση με τον Simbet, που παραδίδει 18% λιγότερα πακέτα. Από τα αποτελέσματα μπορούμε να συμπεράνουμε ότι στο συγκεκριμένο σενάριο κίνησης παρατηρείται μεγαλύτερη συνδεσιμότητα αφού και ο αλγόριθμος Direct παραδίδει επιτυχώς μεγάλο αριθμό πακέτων. Στόχος της παρούσας εργασίας είναι μεν να προτείνει έναν υβριδικό αλγόριθμο, αλλά όπως βλέπουμε μπορεί κάλλιστα να λειτουργήσει αποτελεσματικά σε πυκνά δίκτυα όπως και ένα MANET πρωτόκολλο.



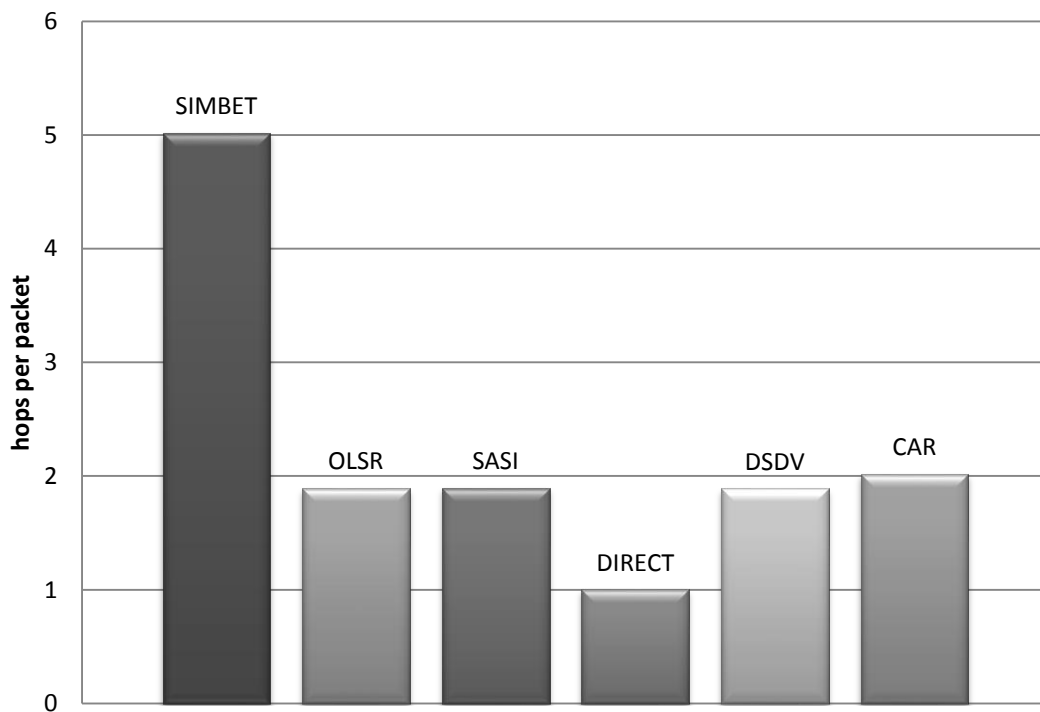
Σχήμα 4.16 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης B)



Σχήμα 4.17 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης B)

Η μέση καθυστέρηση παράδοσης στο σχήμα 4.17 έχει επίσης ελάχιστες διαφορές μεταξύ των πρωτοκόλλων OLSR, SASI, DSDV και CAR, όπως και το ποσοστό επιτυχούς παράδοσης. Ο αλγόριθμος μας σε αυτή την περίπτωση δικτύου χρησιμοποιεί περισσότερο τη σύγχρονη λειτουργία παρά την ασύγχρονη.

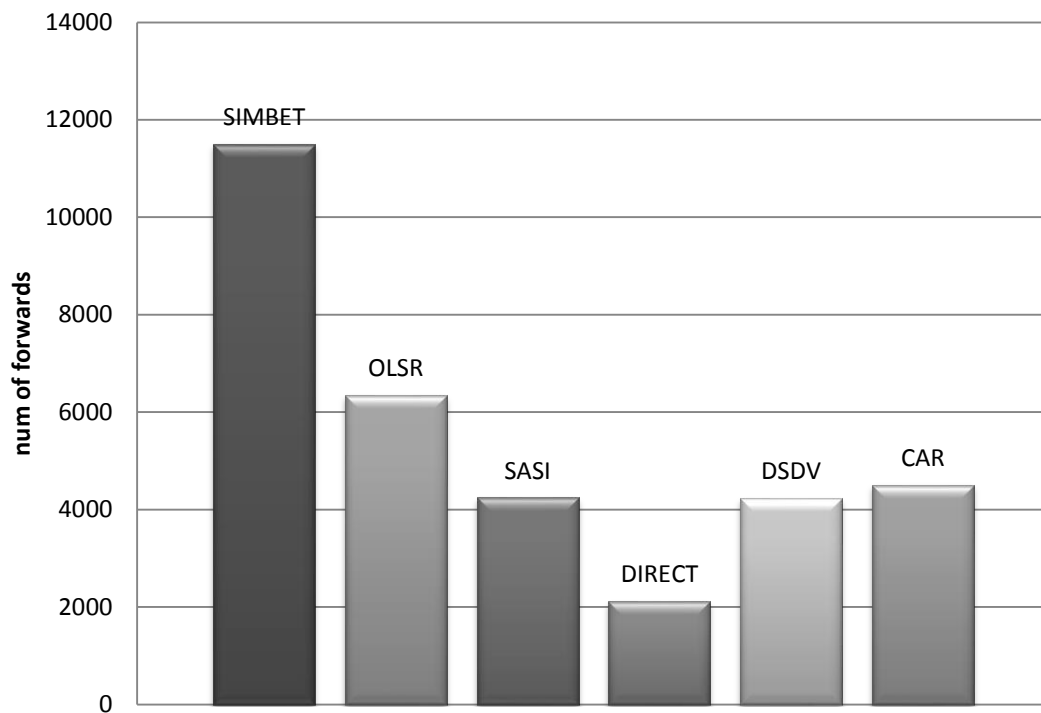
Παρόμοια συμπεριφορά παρατηρούμε και στο σχήμα 4.18 που απεικονίζεται ο μέσος αριθμός αλμάτων ανά πακέτο. Εντύπωση προκαλεί η πολύ μεγάλη διαφορά του Simbet σε σχέση με τα υπόλοιπα πρωτόκολλα, με αριθμό αλμάτων περίπου 2,5 φορές μεγαλύτερο σε σχέση με το SASI.



Σχήμα 4.18 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης B)

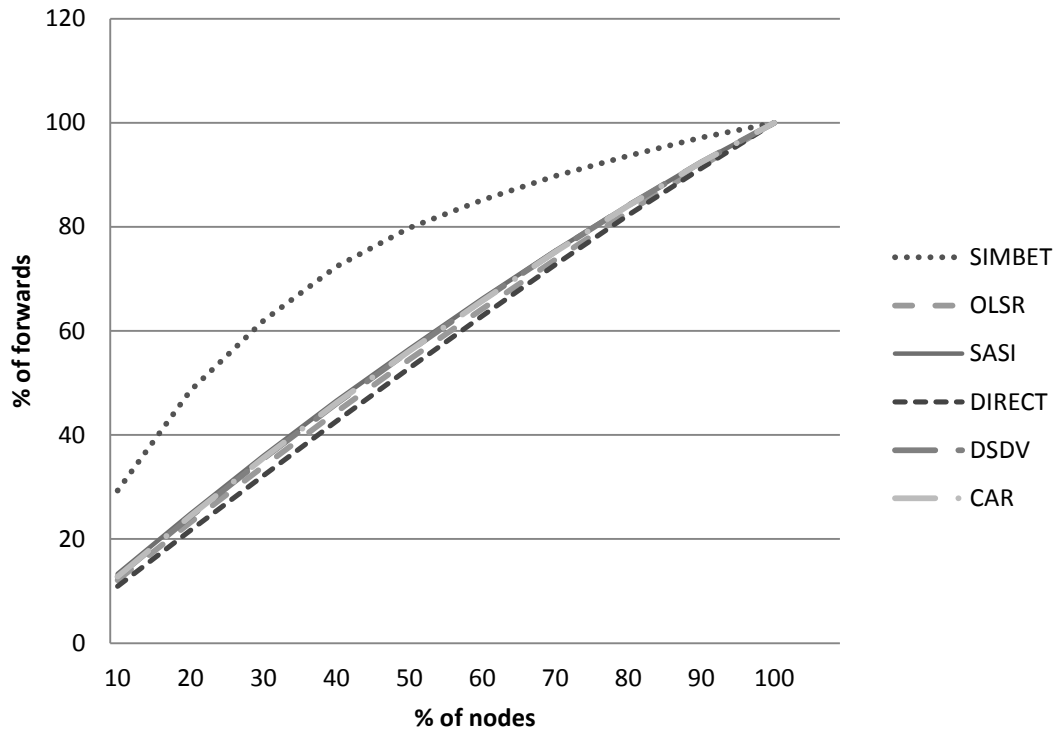
Στο σχήμα 4.19 βλέπουμε τον αριθμό των προωθήσεων για κάθε πρωτόκολλο. Όπως και στο προηγούμενο σχήμα με το μέσο αριθμό αλμάτων έτσι και σε αυτό, ο Simbet

έχει τις μεγαλύτερες τιμές. Από την άλλη πλευρά, παρατηρούμε ότι και ο αριθμός προωθήσεων του OLSR είναι αυξημένος κατά περίπου 30% σε σχέση του SASI. Σε αυτό το σενάριο κίνησης μπορεί ο αλγόριθμός μας να μην διαφοροποιείται σε μεγάλο ποσοστό από τον OLSR όσον αφορά το ποσοστό επιτυχούς παράδοσης και τη μέση καθυστέρηση αλλά καταφέρνει να μειώσει αρκετά τον αριθμό των προωθήσεων. Από ότι φαίνεται ο αλγόριθμός μας καταφέρνει να μειώσει τις περιττές μεταδόσεις.



Σχήμα 4.19 Αριθμός προωθήσεων (Σενάριο κίνησης Β)

Παρατηρώντας το σχήμα 4.20, μπορούμε να δούμε την ιδανική κατανομή των προωθήσεων στους κόμβους του δικτύου σε όλα τα πρωτόκολλα πλην του Simbet κάτι που οφείλεται στην συνδεσιμότητα των κόμβων. Όπως είδαμε και στην παράγραφο 4.2, στην εικόνα 4.4 όλοι οι κόμβοι συναντιούνται με όλους τους άλλους έστω και μία φορά κατά τη διάρκεια των πέντε ημερών της προσομοίωσης.

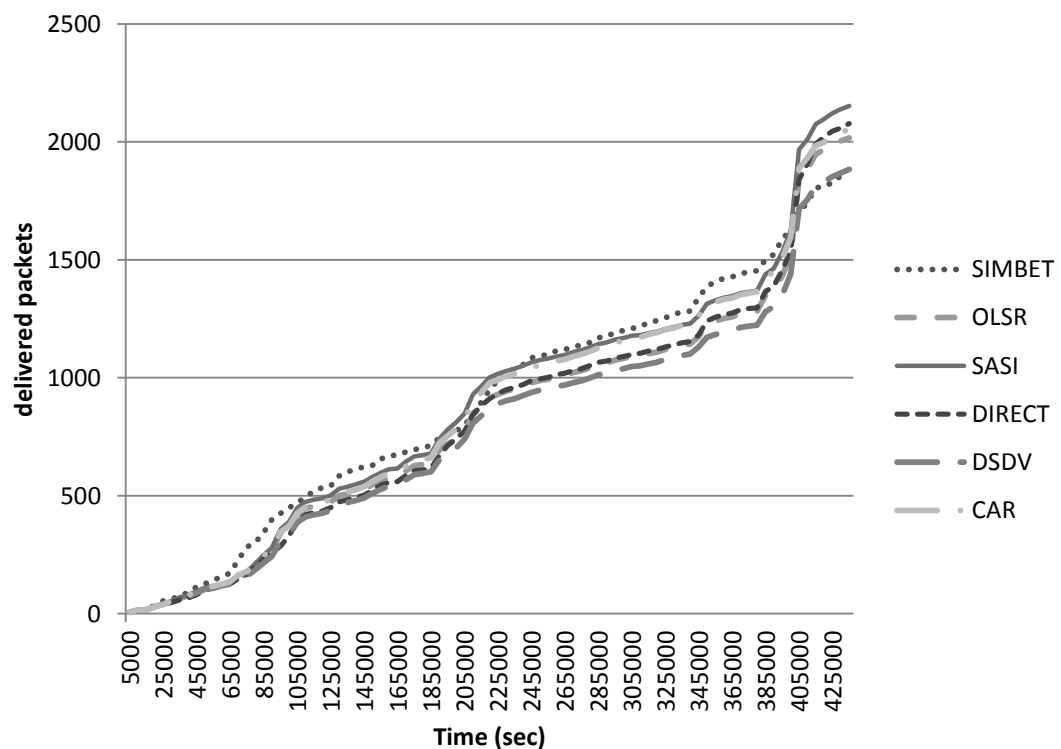


Σχήμα 4.20 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Β)

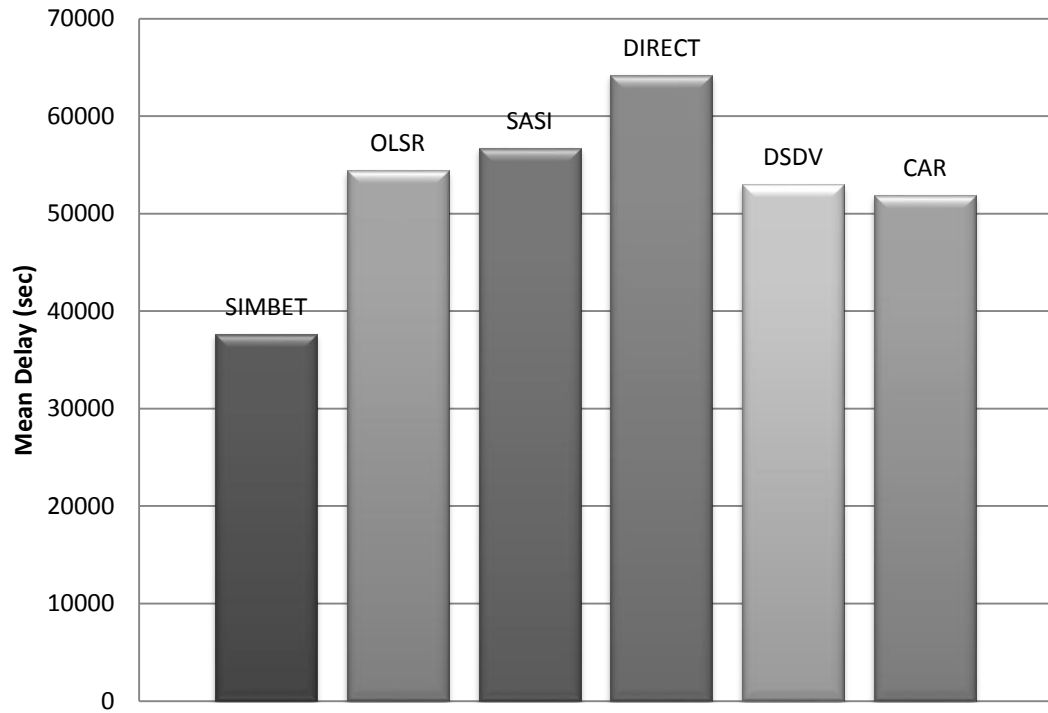
4.3.3. Σενάριο Κίνησης Γ

Για το τρίτο σενάριο κίνησης το πλήθος των πακέτων που παραδόθηκαν σε σχέση με το χρόνο παρουσιάζεται στο σχήμα 4.21. Το γεγονός που προκαλεί εντύπωση είναι ότι ο αλγόριθμος Direct έχει αρκετά υψηλά ποσοστά επιτυχούς παράδοσης που τον κατατάσσουν δεύτερο στη σειρά μετά τον SASI με διαφορά 3%. Το γεγονός αυτό είναι μία ένδειξη ότι στο συγκεκριμένο σενάριο κίνησης στο τέλος της προσομοίωσης οι κόμβοι θα έχουν γίνει κάποια στιγμή γείτονες με σχεδόν όλους τους άλλους. Παρατηρώντας όμως και το σχήμα 4.22, βλέπουμε ότι το πρωτόκολλο Direct έχει και τη μεγαλύτερη, με διαφορά, μέση καθυστέρηση σε σχέση με τα υπόλοιπα, προφανώς επειδή οι κόμβοι διατηρούν για μεγάλο χρονικό διάστημα τα πακέτα στη μνήμη τους περιμένοντας να συναντήσουν τον παραλήπτη. Αυτό μας δείχνει ότι μπορεί οι κόμβοι να έχουν πολλούς γείτονες, δεν έχουν όμως πολλές επαφές κατά τη διάρκεια της προσομοίωσης, δηλαδή υπάρχει μία περιορισμένη κινητικότητα στο σενάριο αυτό.

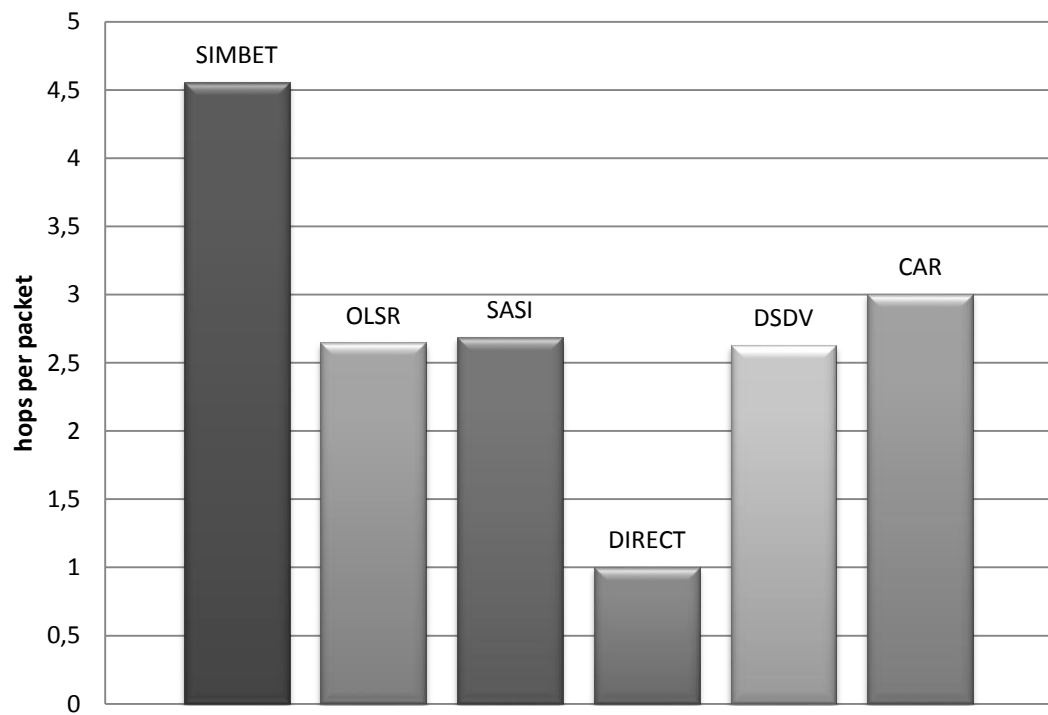
Η συμπεριφορά των υπόλοιπων αλγορίθμων είναι αντίστοιχη με τα προηγούμενα σενάρια, με τον OLSR να κυμαίνεται πολύ κοντά σε ποσοστά επιτυχούς παράδοσης με τον CAR και ακολουθούν ο DSDV με τον Simbet. Παρόλο που παρατηρούνται διακυμάνσεις, από τα μισά του χρόνου της προσομοίωσης παρατηρούμε παρόμοια συμπεριφορά για όλους τους αλγορίθμους, καθώς και τις τιμές τους να ταυτίζονται σχεδόν (εκτός από τον Simbet και τον DSDV) λίγο πριν το τέλος της προσομοίωσης. Σύμφωνα με τη γραφική παράσταση της μέσης καθυστέρησης παράδοσης, ο αλγόριθμός μας τερματίζει τελικώς με τη δεύτερη μεγαλύτερη καθυστέρηση μετά τον Direct που δικαιολογείται λόγω του αριθμού των πακέτων που παραδίδει και είναι λίγο μεγαλύτερη από του OLSR.



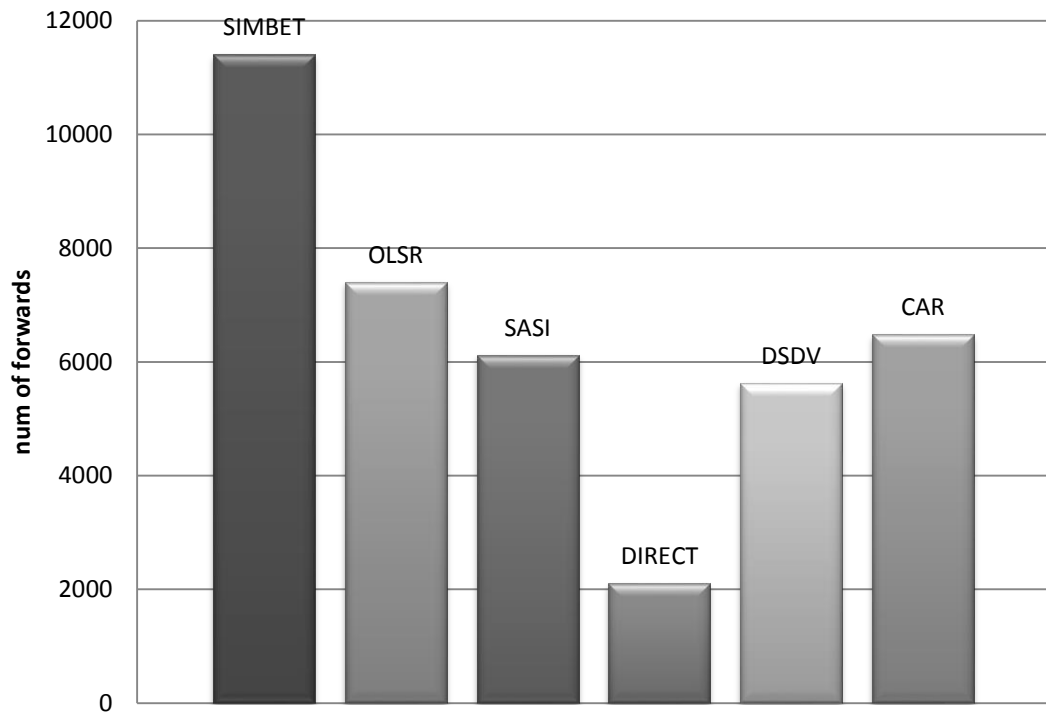
Σχήμα 4.21 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Γ)



Σχήμα 4.22 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Γ)

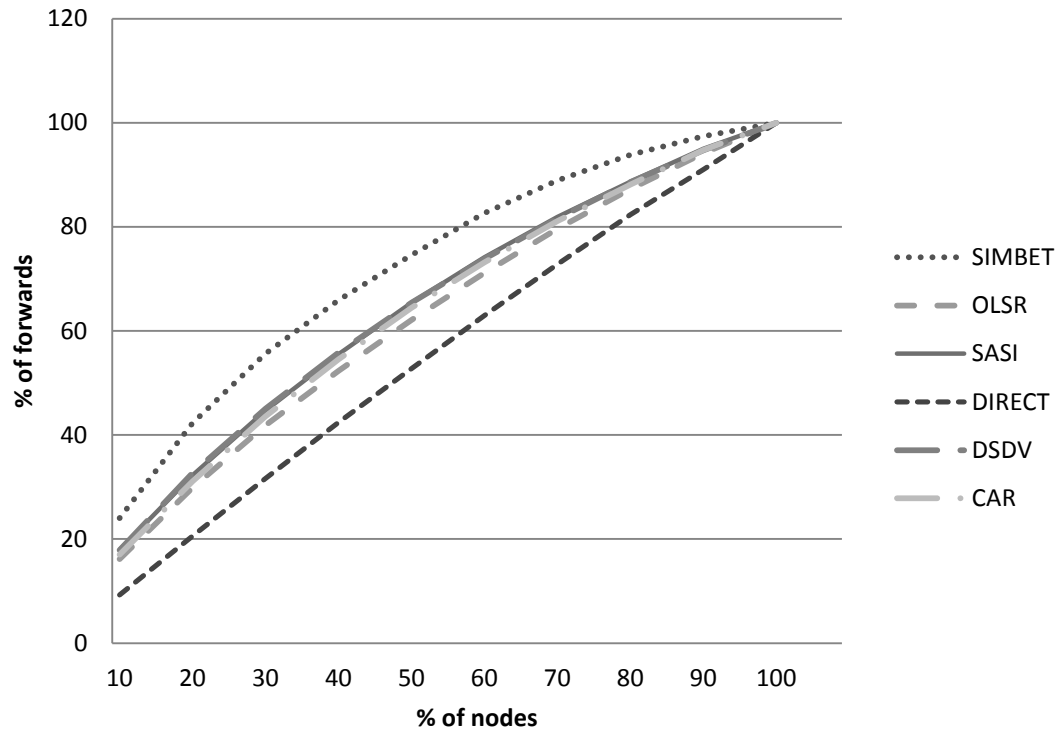


Σχήμα 4.23 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Γ)



Σχήμα 4.24 Αριθμός προωθήσεων (Σενάριο κίνησης Γ)

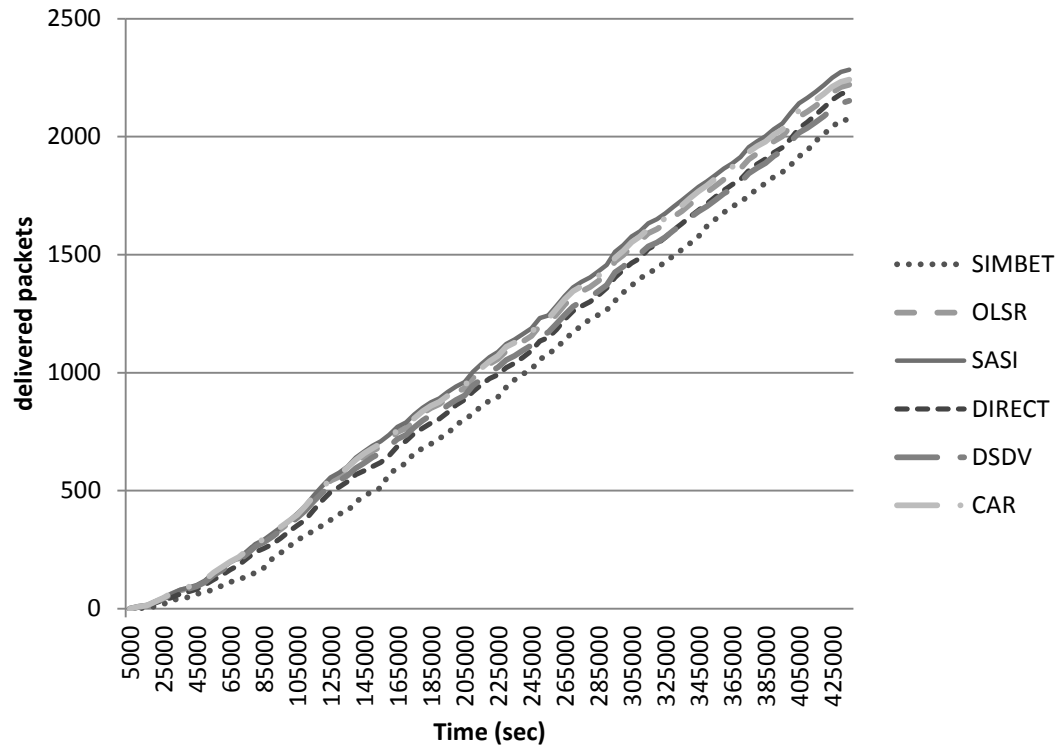
Ο μέσος αριθμός αλμάτων (σχήμα 4.23) και αριθμός προωθήσεων (σχήμα 4.24) παρουσιάζουν την ίδια εικόνα με του δεύτερου σεναρίου με μικρές διαφορές στην απόλυτη τιμή των μετρικών. Επίσης η κατανομή των προωθήσεων στους κόμβους του δικτύου δεν είναι τόσο ιδανική όσο στο προηγούμενο σενάριο.



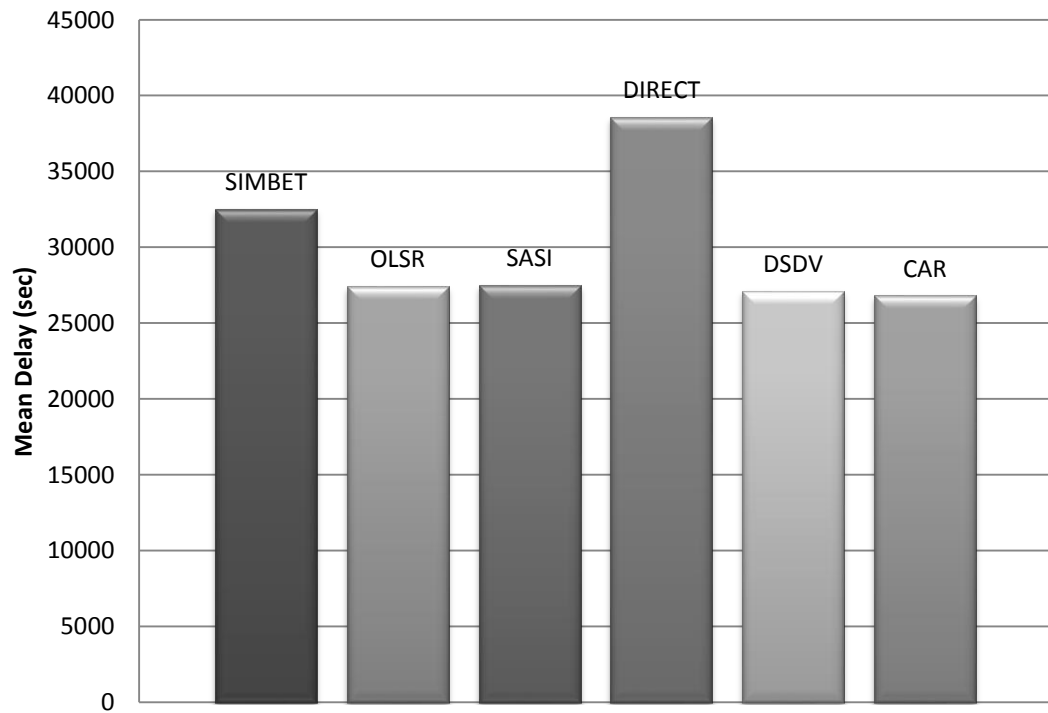
Σχήμα 4.25 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Γ)

4.3.4. Σενάριο Κίνησης Δ

Στο σχήμα 4.26 βλέπουμε το πλήθος πακέτων που παραδόθηκαν σε συνάρτηση με το χρόνο για το τέταρτο σενάριο κίνησης. Η γραφική παράσταση είναι ανοδική και μοιάζει αρκετά για όλα τα πρωτόκολλα. Σε αυτό το σενάριο οι τιμές μεταξύ των αλγορίθμων δεν παρουσιάζουν τόσο μεγάλες διακυμάνσεις όσο στα προηγούμενα, αφού έχουν όλοι αρκετά υψηλά ποσοστά επιτυχούς παράδοσης. Παρ' όλα αυτά ο αλγόριθμος SASI παραμένει πρώτος παραδίδοντας 2% περισσότερα πακέτα από τον CAR. Ακολουθούν κατά σειρά κατάταξης ο OLSR, Direct, DSDV και τελευταίος ο Simbet. Ο OLSR παραδίδει 3% λιγότερα πακέτα από τον SASI ενώ ο Simbet 8,5%.



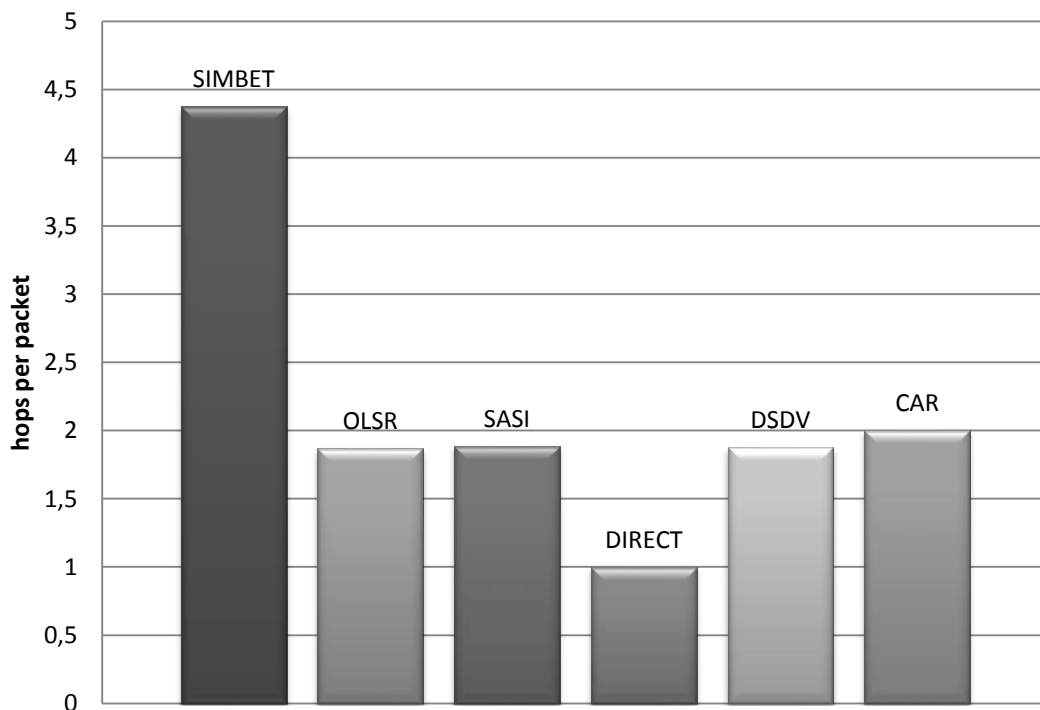
Σχήμα 4.26 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης Δ)



Σχήμα 4.27 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης Δ)

Η εικόνα της γραφικής παράστασης της μέσης καθυστέρησης παράδοσης είναι αυτή που αναμέναμε, με τα πρωτόκολλα που έχουν μικρές διαφορές στο ποσοστό επιτυχούς παράδοσης, να παρουσιάζουν την ίδια συμπεριφορά και εδώ. Ο αλγόριθμος Direct έχει και εδώ τη μεγαλύτερη μέση καθυστέρηση και ακολουθεί ο αλγόριθμος Simbet.

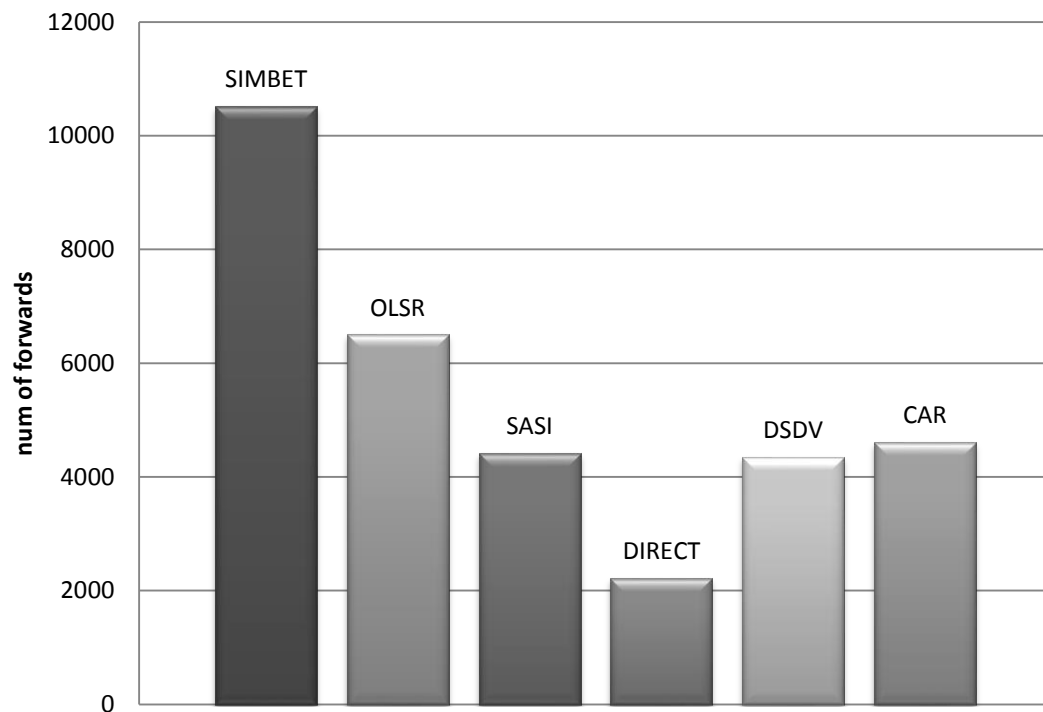
Όσον αφορά το μέσο αριθμό αλμάτων που έκανε ένα πακέτο μέχρι να φτάσει στον παραλήπτη, ο Simbet χρειάστηκε περισσότερο από τα διπλά άλματα σε σχέση με τον SASI. Οι υπόλοιποι αλγόριθμοι έχουν περίπου ίδιες τιμές αλλά παραμένει θετικό το γεγονός ότι ο αλγόριθμος μας παρουσιάζει μικρή διαφορά προς τα κάτω σε σχέση με τον CAR διατηρώντας ταυτόχρονα τη μικρή διαφορά προς τα πάνω όσον αφορά το ποσοστό επιτυχούς παράδοσης.



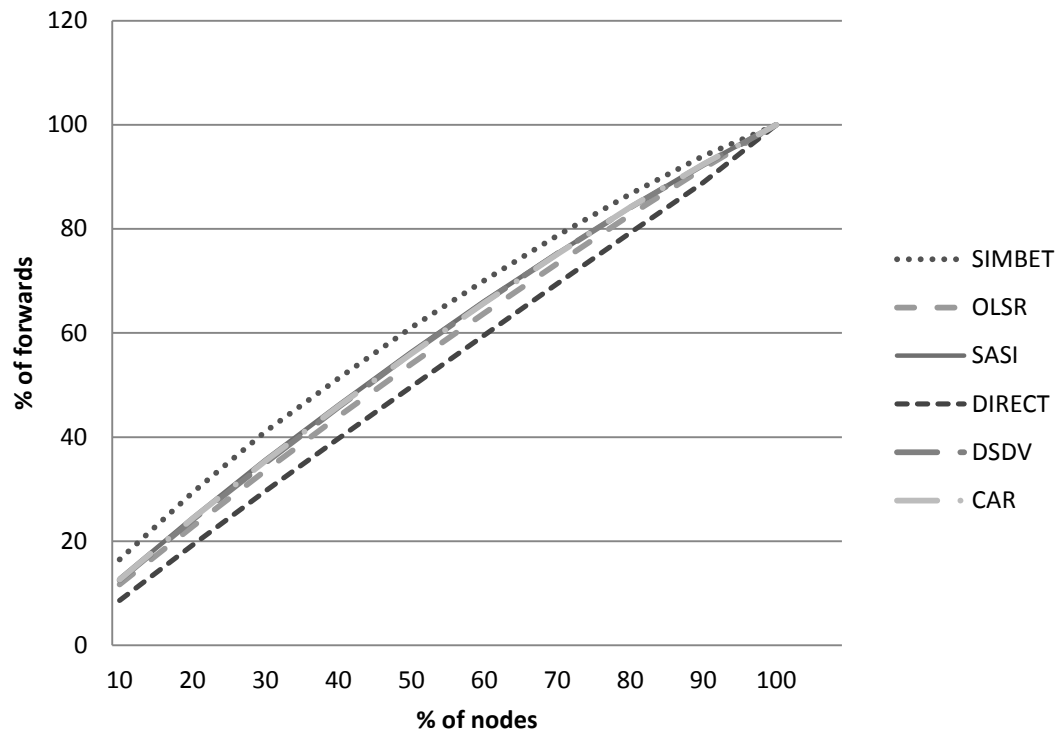
Σχήμα 4.28 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης Δ)

Στο σχήμα 4.29 βλέπουμε τον αριθμό των προωθήσεων για κάθε αλγόριθμο. Μικρότερη τιμή έχει φυσικά ο Direct, ακολουθεί ο DSDV και τρίτος στην κατάταξη

είναι ο SASI. Εδώ παρατηρούμε για ακόμη μία φορά ότι ο αλγόριθμος μας πετυχαίνει τις λιγότερες δυνατές προωθήσεις σε σχέση με τα πακέτα που παραδίδει, ενισχύοντας την άποψη ότι η δρομολόγηση γίνεται όσο αποτελεσματικά είναι εφικτό. Η κατανομή των προωθήσεων στο σχήμα 4.30 γίνεται σχεδόν όπως στην ιδανική κατάταξη του Direct.



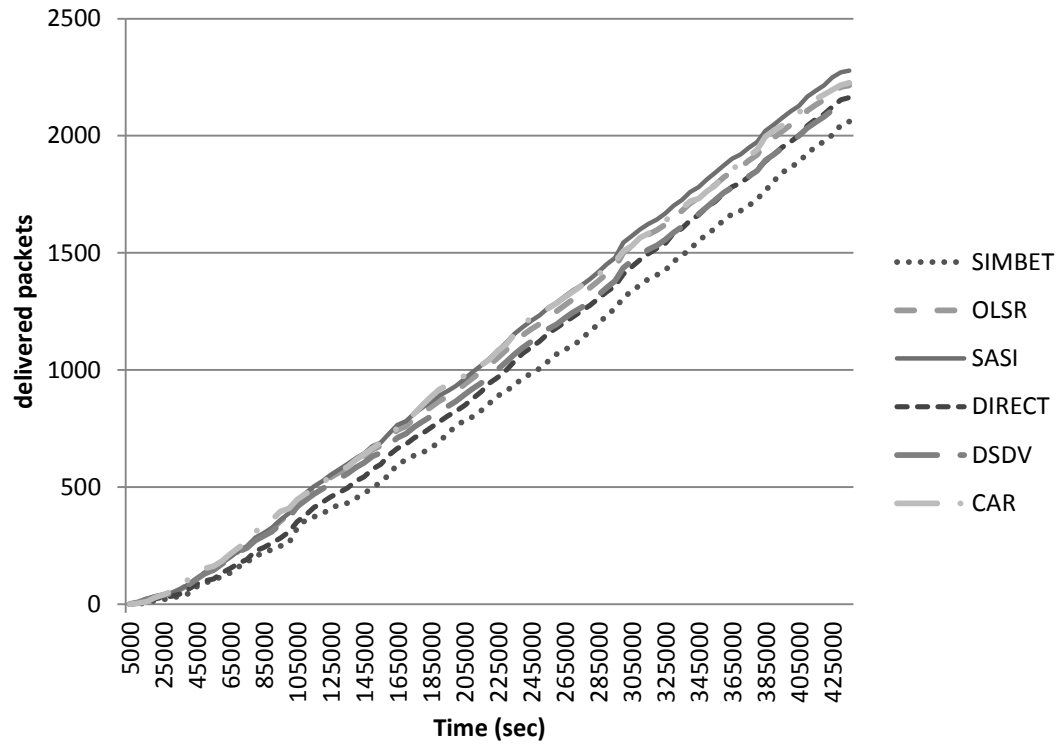
Σχήμα 4.29 Αριθμός προωθήσεων (Σενάριο κίνησης Δ)



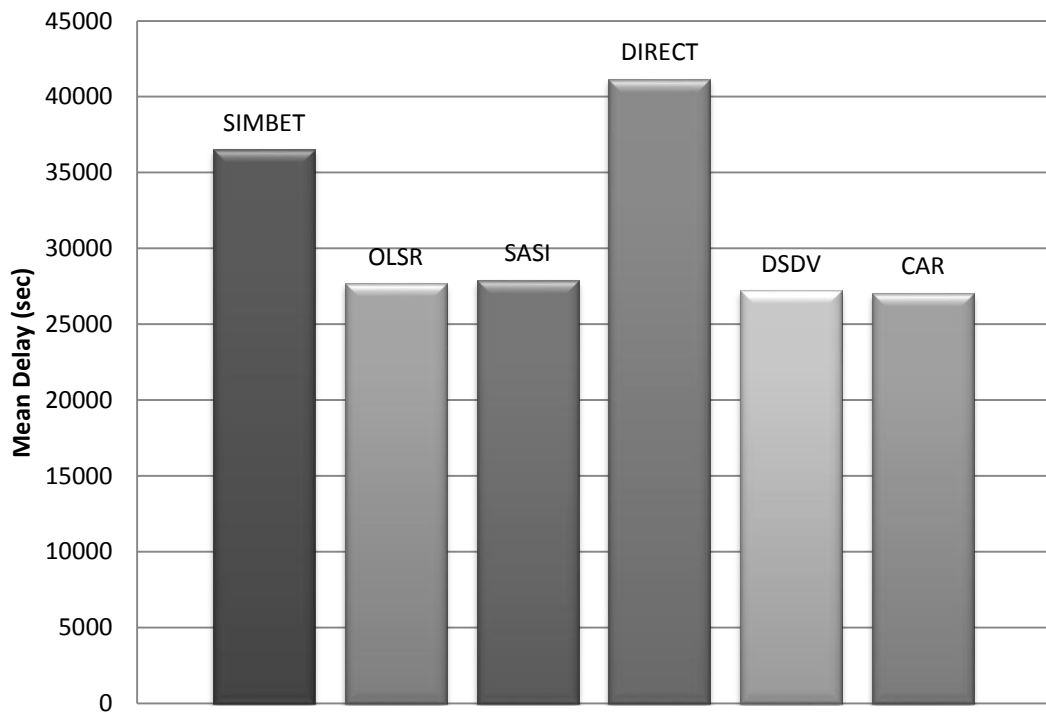
Σχήμα 4.30 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης Δ)

4.3.5. Σενάριο Κίνησης E

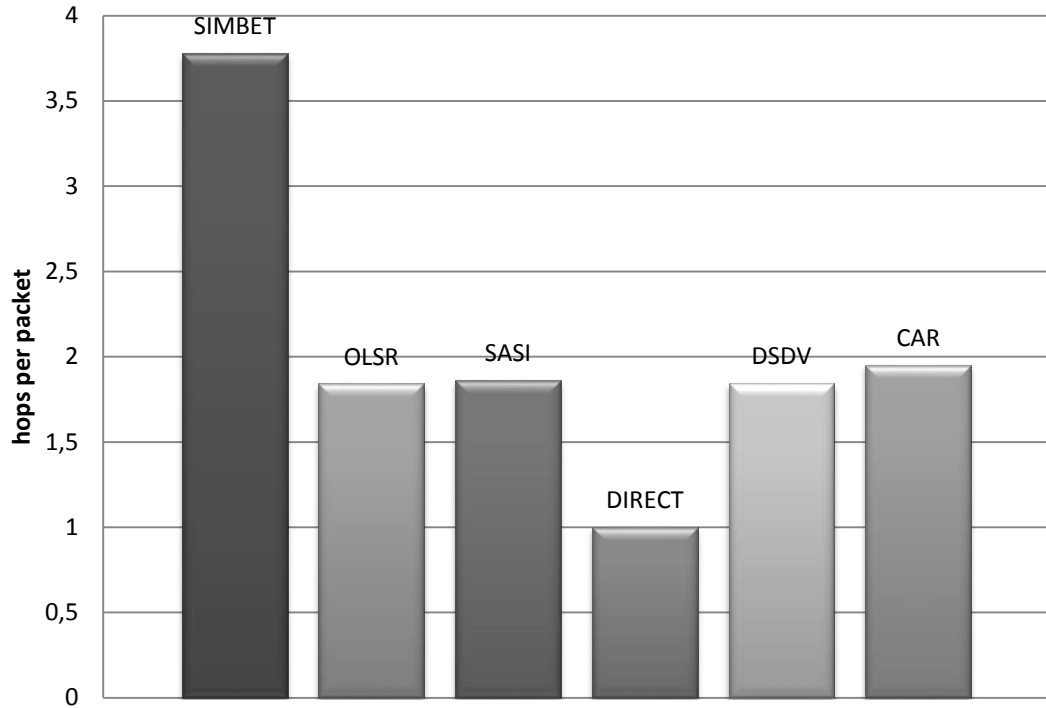
Όπως μπορούμε να παρατηρήσουμε στα επόμενα σχήματα τα αποτελέσματα του πέμπτου σεναρίου κίνησης μοιάζουν πολύ με του τέταρτου, με όλες τις γραφικές παραστάσεις να εμφανίζουν παρόμοιες εικόνες. Η διαφορά εδώ είναι ότι στο ποσοστό επιτυχούς παράδοσης ο CAR πλησιάζει πολύ το αλγόριθμό μας και σε κάποιες χρονικές στιγμές, περίπου στο 1/3 της προσομοίωσης παρουσιάζει καλύτερα αποτελέσματα. Τελικώς όμως ο SASI προηγείται κατά 2%.



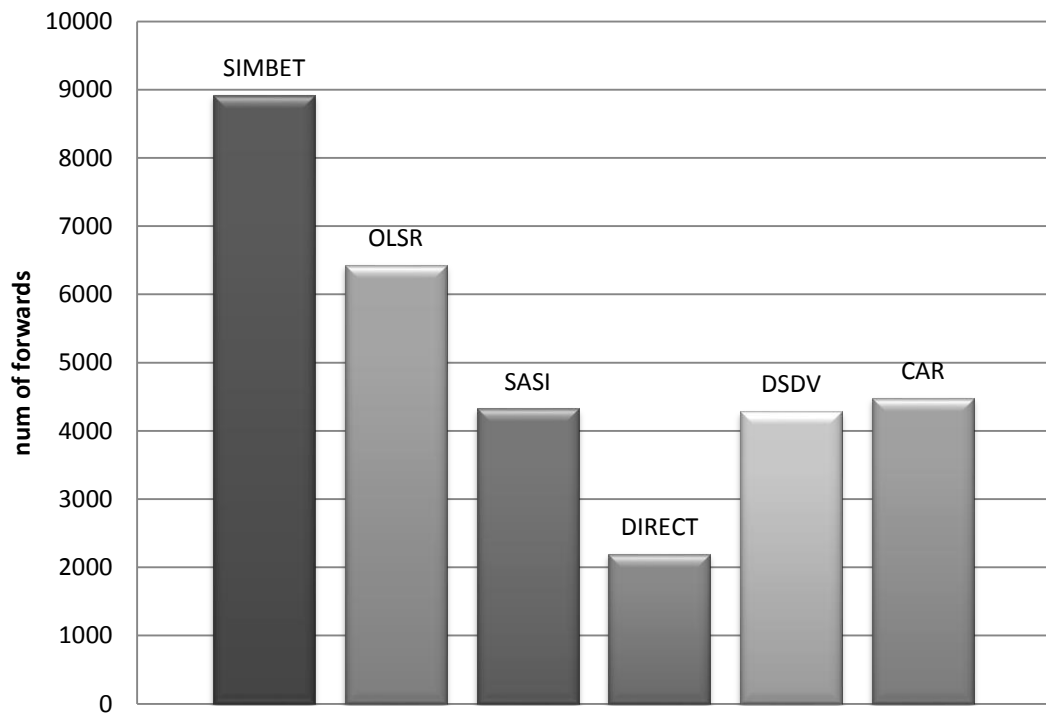
Σχήμα 4.31 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης E)



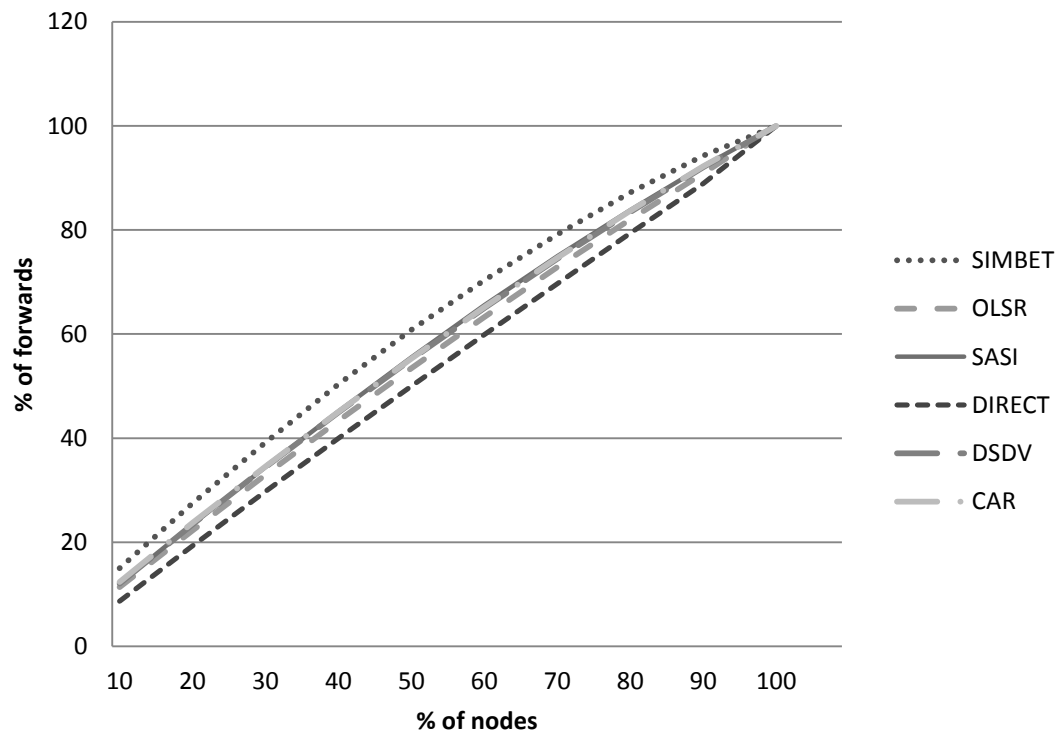
Σχήμα 4.32 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης E)



Σχήμα 4.33 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης E)



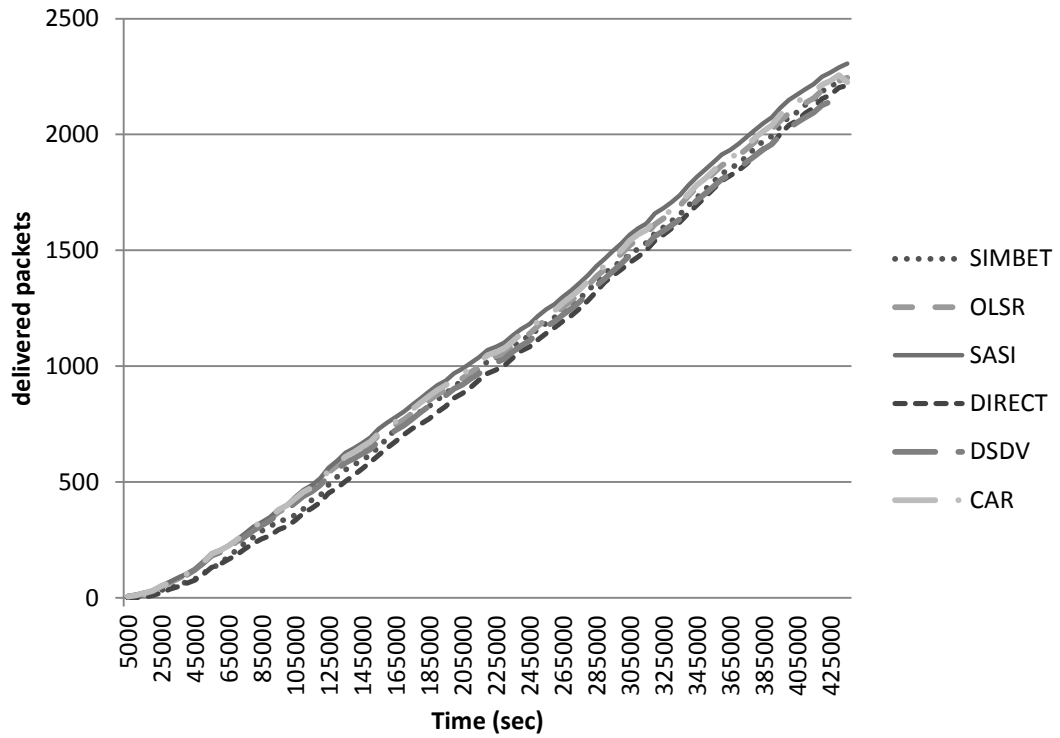
Σχήμα 4.34 Αριθμός προωθήσεων (Σενάριο κίνησης E)



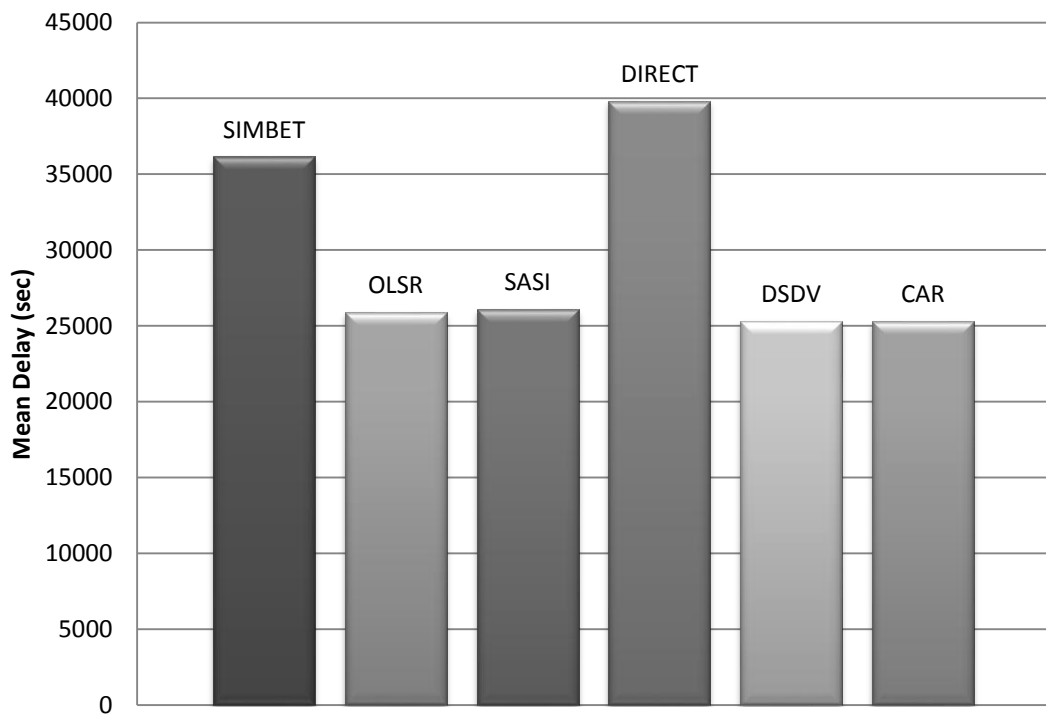
Σχήμα 4.35 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης E)

4.3.6. Σενάριο Κίνησης ΣΤ

Το έκτο σενάριο κίνησης είναι το μόνο που ο αλγόριθμος Simbet παρουσιάζει τόσο μεγάλο αριθμό πακέτων όπως βλέπουμε και στο σχήμα 4.36, με τις τιμές να βρίσκονται στα ίδια επίπεδα με του OLSR. Ο CAR κατατάσσεται στην επόμενη θέση με διάφορα από τον πρώτο σε κατάταξη SASI περίπου 3% και σχεδόν 1% κάτω από τον Simbet. Αυτό που μπορούμε να συμπεράνουμε από τα αποτελέσματα, είναι ότι σε αυτό το σενάριο κίνησης εμφανίζεται το φαινόμενο οι κόμβοι να συναντούν σχεδόν όλους τους άλλους, αφού και ο Direct έχει υψηλά ποσοστά. Ταυτόχρονα όμως υπάρχουν και πολλές επαφές μεταξύ τους ώστε να καταφέρνει και ο Simbet να παραδίδει αρκετά πακέτα. Παρόλα αυτά η γραφική παράσταση της μέσης καθυστέρησης δεν παρουσιάζει διαφοροποιήσεις σε σχέση με τα δύο προηγούμενα σενάρια. Μόνο που εδώ μπορούμε να δικαιολογήσουμε την αυξημένη καθυστέρηση που παρουσιάζει ο Simbet λόγω του αριθμού των πακέτων που παραδίδει.

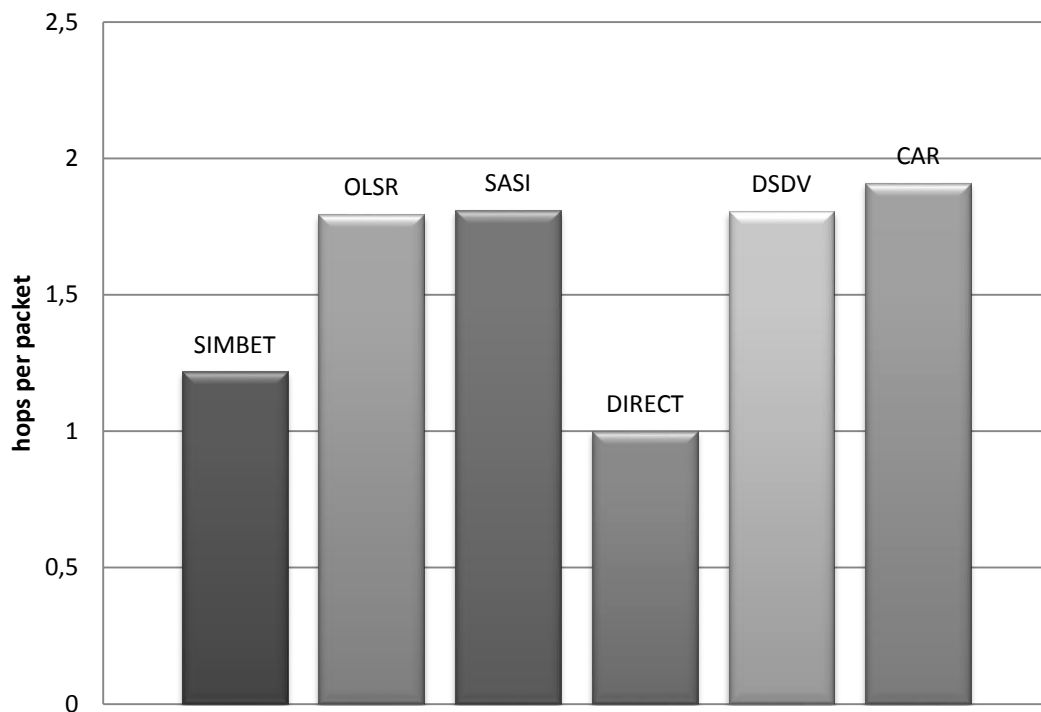


Σχήμα 4.36 Πλήθος μηνυμάτων που παραδόθηκαν σε σχέση με το χρόνο (Σενάριο κίνησης ΣΤ)



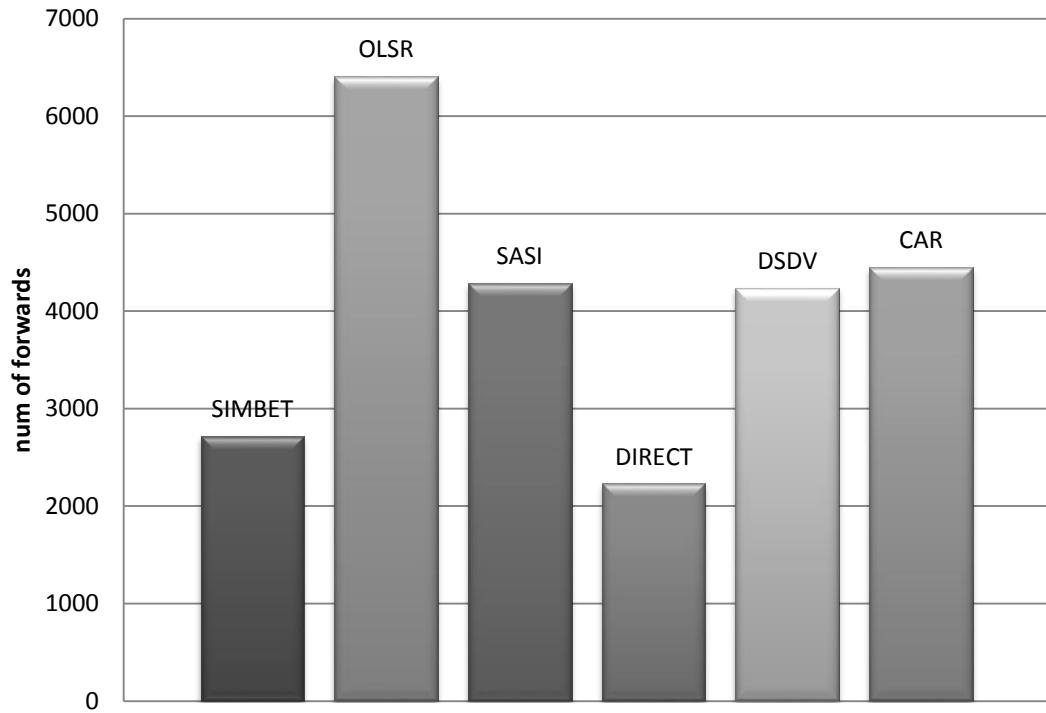
Σχήμα 4.37 Μέση καθυστέρηση παράδοσης (Σενάριο κίνησης ΣΤ)

Μία σημαντική διαφορά σε σχέση με τα προηγούμενα σενάρια παρουσιάζει ο μέσος αριθμός αλμάτων ανά πακέτο. Το μικρότερο αριθμό έχει όπως πάντα ο Direct και ακολουθεί ο Simbet. Οι αλγόριθμοι OLSR, SASI και DSDV έχουν σχεδόν ίδιες τιμές.

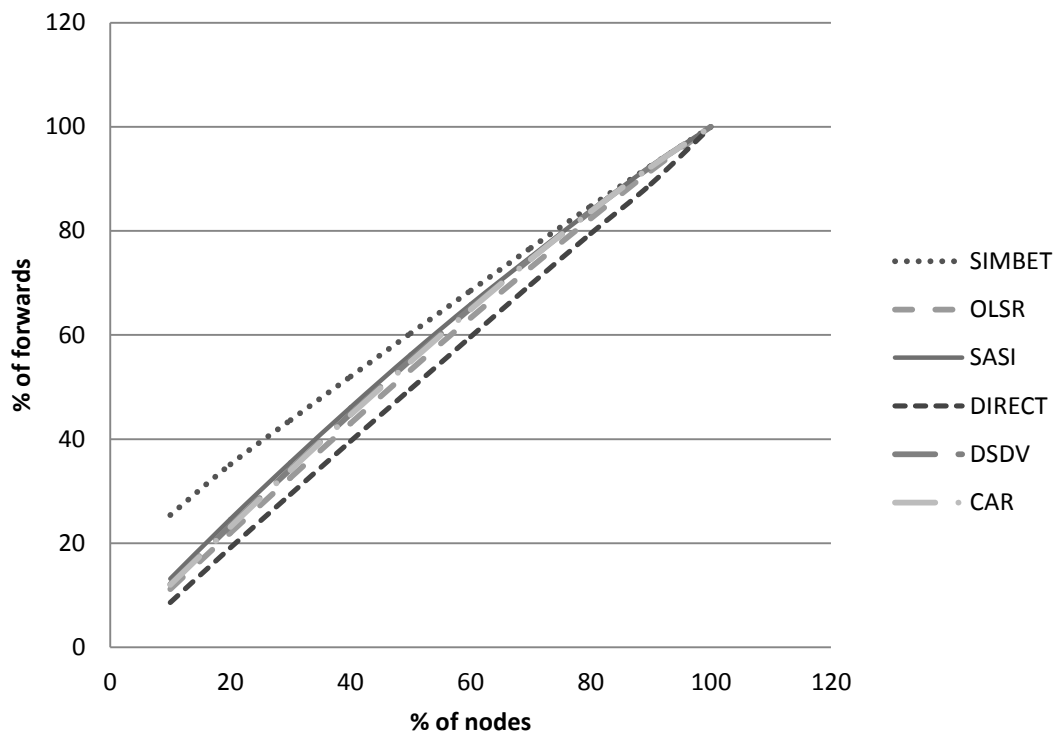


Σχήμα 4.38 Μέσος αριθμός αλμάτων ανά πακέτο (Σενάριο κίνησης ΣΤ)

Ο αλγόριθμος Simbet βρίσκεται στη δεύτερη χαμηλότερη θέση και στον αριθμό των συνολικών προωθήσεων. Οι τιμές του OLSR αυξάνονται κατά πολύ στο συγκεκριμένο σενάριο, ενώ ο SASI βρίσκεται περίπου στη μέση μεταξύ OLSR και Simbet. Φαίνεται λοιπόν από τα αποτελέσματα ότι ο αλγόριθμός μας χρησιμοποιεί περισσότερο την σύγχρονη λειτουργία για τη δρομολόγηση των δεδομένων αλλά καταφέρνει να αξιολογεί καλύτερα τους κόμβους χρησιμοποιώντας τις μετρικές κοινωνικής δικτύωσης πετυχαίνοντας να μειώσει πολύ τις περιττές μεταδόσεις του OLSR και ταυτόχρονα να παραδώσει κατά 2,5% περισσότερα πακέτα από τον Simbet.



Σχήμα 4.39 Αριθμός προωθήσεων (Σενάριο κίνησης ΣΤ)



Σχήμα 4.40 Ποσοστό προωθήσεων σε σχέση με το ποσοστό των κόμβων (Σενάριο κίνησης ΣΤ)

ΚΕΦΑΛΑΙΟ 5. ΣΥΜΠΕΡΑΣΜΑΤΑ

5.1 Ανασκόπηση

5.2 Προτάσεις για Περαιτέρω Μελέτη

5.1. Ανασκόπηση

Στα πλαίσια της παρούσας εργασίας ασχοληθήκαμε με την μελέτη μίας κατηγορίας δικτύων που εμφανίζουν μεικτά χαρακτηριστικά συνδεσιμότητας. Παρουσιάζουν, δηλαδή, περιοχές συγκέντρωσης κόμβων με χαρακτηριστικό την συνδεσιμότητα εντός των περιοχών αλλά όχι στο σύνολο του δικτύου. Σε αυτές τις περιπτώσεις δικτύων ούτε οι αλγόριθμοι που έχουν προταθεί για MANET αλλά ούτε και αυτοί για DTN δίκτυα μπορούν να προσφέρουν ικανοποιητικά αποτελέσματα. Στις ενότητες που προηγήθηκαν μελετήσαμε το θεωρητικό υπόβαθρο των αλγορίθμων που έχουν προταθεί για MANET και DTN. Έπειτα, ασχοληθήκαμε με τους υβριδικούς αλγορίθμους που χρησιμοποιούν κατά περίπτωση σύγχρονους και ασύγχρονους αλγορίθμους και έχουν προταθεί μέχρι στιγμής καθώς και τα προβλήματα που αυτοί αντιμετωπίζουν.

Στην διατριβή αυτή προτείνουμε έναν νέο υβριδικό αλγόριθμο, τον SASI, που συνδυάζει τις αρχές της σύγχρονης δρομολόγησης από τα δίκτυα MANET με τις αρχές της ασύγχρονης δρομολόγησης από τα DTN δίκτυα. Για τη σύγχρονη δρομολόγηση χρησιμοποιείται το πρωτόκολλο δρομολόγησης OLSR. Δεδομένου ότι η κίνηση των κόμβων εξαρτάται από την κοινωνική συμπεριφορά των ανθρώπων που κουβαλούν τις ασύρματες συσκευές, επιλέχθηκε για την ασύγχρονη λειτουργία ένα πρωτόκολλο κοινωνικής δικτύωσης, ο Simbet. Σκοπός είναι να διατηρήσουμε τη σύγχρονη λειτουργία του OLSR αλλά προσθέτοντας στοιχεία που θα επιτρέπουν να γίνει η δρομολόγηση σε περιπτώσεις διάσπασης του δικτύου. Η βασική ιδέα είναι να

χρησιμοποιήσουμε μετρικές κοινωνικής δικτύωσης και να τις διαδώσουμε στους υπόλοιπους κόμβους αξιοποιώντας τη λειτουργία του OLSR.

Προκειμένου να μην επιβαρύνουμε το δίκτυο με επιπλέον μηνύματα σηματοδοσίας, χρησιμοποιούμε τα TC messages που εκπέμπει κάθε κόμβος για να ανακοινώσει τους κόμβους που έχει επιλέξει ως MPR. Δεδομένου ότι οι κόμβοι σχηματίζουν partitions, τα TC messages προωθούνται μόνο εντός του partition του κόμβου μέσω των MPR κόμβων. Προσθέτοντας, λοιπόν, τις τιμές των μετρικών κοινωνικής δικτύωσης betweenness και similarity που χρησιμοποιεί ο Simbet στα TC μηνύματα, κάθε κόμβος μπορεί να γνωρίζει τις κοινωνικές σχέσεις των υπολοίπων. Παράλληλα, εισάγουμε και τη διαδικασία ανταλλαγής πακέτων του Simbet κάθε φορά που γίνεται ανακάλυψη ενός νέου γείτονα. Σύμφωνα με αυτήν οι κόμβοι ανταλλάσσουν μηνύματα, μεταξύ αυτών και τις μετρικές τους. Έπειτα ο κόμβος συγκρίνει τις μετρικές προκειμένου να αποφασίσει για το επόμενο άλμα του πακέτου. Στον SASI τροποποιούμε τη διαδικασία αυτή ώστε η σύγκριση να μην γίνεται πλέον μεταξύ δύο κόμβων αλλά μεταξύ του τρέχοντος μεταφορέα του μηνύματος και όλων των κόμβων που ανήκουν στο partition. Με αυτόν τον τρόπο το μήνυμα προωθείται κάθε φορά στον κόμβο που έχει την καλύτερη μετρική κοινωνικής δικτύωσης σε σχέση με όλους όσους έχει συνδεσιμότητα.

Στα πειράματα που διεξαγάγαμε για να αξιολογήσουμε τον αλγόριθμό μας χρησιμοποιήσαμε συνθετικά μοντέλα κίνησης ώστε να μπορούμε να κατασκευάσουμε δίκτυα στα οποία οι κόμβοι θα παρουσιάζουν σημεία συγκέντρωσης. Συγκρίνουμε τον SASI με παραδοσιακούς αλγορίθμους για MANET και DTN αλλά και με τον υβριδικό CAR. Αναλύοντας τα αποτελέσματα των πειραμάτων παρατηρείται ότι σε όλα τα σενάρια κίνησης που εκτελέσαμε, ο SASI πετυχαίνει να παραδώσει επιτυχώς περισσότερα πακέτα σε σχέση όλους τους συγκρινόμενους αλγορίθμους, άλλες φορές σε μεγαλύτερο και άλλες σε μικρότερο ποσοστό. Επίσης είχε σε όλα τα σενάρια κίνησης τη μεγαλύτερη μέση καθυστέρηση παράδοσης αλλά είναι κάτι αναμενόμενο αφού παραδίδει περισσότερα πακέτα που μένουν για περισσότερο χρόνο αποθηκευμένα στη μνήμη των κόμβων.

Επιπρόσθετα, στα αποτελέσματα σχετικά με τον μέσο αριθμό αλμάτων που χρειάστηκε ένα πακέτο μέχρι να παραδοθεί αλλά και το συνολικό αριθμό των προωθήσεων, παρατηρούμε ότι ο αλγόριθμος μας στην πλειοψηφία των περιπτώσεων πετυχαίνει καλύτερες τιμές σε σχέση με τους υπόλοιπους αλγορίθμους. Από αυτό μπορούμε να συμπεράνουμε ότι το γεγονός ότι κάθε κόμβος γνωρίζει τις μετρικές αξιολόγησης των υπολοίπων βοηθά τον SASI να διαχειρίζεται καλύτερα τις επαφές των κόμβων προωθώντας τα πακέτα σε κόμβους που πραγματικά έχουν αυξημένες πιθανότητες επιτυχούς παράδοσης, μειώνοντας με αυτό τον τρόπο τις περιττές μεταδόσεις των πακέτων και προωθώντας το όλο και πιο κοντά στον παραλήπτη.

5.2. Προτάσεις για Περαιτέρω Μελέτη

Σαν επέκταση της παρούσας εργασίας θα μπορούσε κάποιος να τροποποιήσει τον προτεινόμενο αλγόριθμο ώστε να αξιολογούνται καλύτερα οι επαφές των κόμβων. Αυτό θα μπορούσε να γίνει είτε χρησιμοποιώντας παράθυρο ιστορίας, διατηρώντας μόνο τις πιο πρόσφατες επαφές, είτε αποσβένοντας τη σημαντικότητα των παλαιότερων επαφών με την πάροδο του χρόνου, με παρόμοιο ίσως τρόπο με τον PROPHET [10]. Επιπλέον, αν λάβουμε υπόψη μας την παραλλαγή του SASI όπου κάθε κόμβος προσθέτει στο TC μήνυμα το σύνολο των επαφών του και άρα οι κόμβοι έχουν τη δυνατότητα να αναπαραστήσουν όλο τον κοινωνικό γράφο, θα μπορούσαμε να χρησιμοποιήσουμε διαφορετικές μετρικές κοινωνικής δικτύωσης από τις betweenness και similarity, προκειμένου να γίνει καλύτερη αξιολόγηση των κόμβων.

ΑΝΑΦΟΡΕΣ

- [1] M. Abolhasan, T. Wysocki and E. Dutkiewicz, “A review of routing protocols for mobile ad hoc networks”, *Ad Hoc Networks*, Volume 2, Issue 1, pp. 1-22, 2004
- [2] Chakeres and C. Perkins, “Dynamic MANET On-demand (DYMO) Routing”, *IETF Internet Draft*, draft-ietf-manet-dymo-14, 2008
- [3] X. Chen , “Enabling Disconnected Transitive Communication Mobile Ad Hoc Networks”, *Workshop on Principles of Mobile Computing*, colocated with *PODC’01*, 2001
- [4] E. Daly and Mads Haahr, “Social Network Analysis for Information Flow in Disconnected Delay-Tolerant MANETs”, *Mobile Computing, IEEE Transactions*, Volume:8, Issue: 5), pp. 606–621, 2009
- [5] E. M. Daly and Mads Haahr, “The challenges of disconnected delay-tolerant MANETs”, *Ad Hoc Networks*, Volume 8, Issue 2, pp. 241-250, 2010
- [6] Deng, J., Hui Zeng ; Xu, R., Xudong Wang, “Dynamic and robust routing protocol for orbit access networks in lunar exploration”, *Military Communications Conference*, 2009. *MILCOM 2009*, pp. 1-7, 2009
- [7] Holliday, P., ” *NOMAD A Mobile Ad Hoc and Distruption Tolerant Routing Protocol for Tactical Military Networks*”, *ICDCSW '09 Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems Workshops*, pp. 488-492, 2009

- [8] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International, pp. 62-68, 2001
- [9] Kretschmer, C., Ruhrup, S. and Schindelbauer, C., "DT-DYMO: Delay-Tolerant Dynamic MANET On-demand Routing", ICDCSW '09 Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems Workshops, pp. 493-498, 2009
- [10] Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks", ACM SIGMOBILE Mobile Computing and Communications Review, pp. 19-20, 2003
- [11] P. V Marsden, "Egocentric and sociocentric measures of network centrality", Social Networks, Volume 24, Issue 4, pp. 407-422, 2002
- [12] Musolesi, M., Hailes, S. and Mascolo, C., "Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks", Sixth IEEE International Symposium on World of Wireless Mobile and Multimedia Networks, pp. 183-189, 2005
- [13] M. Musolesi and C. Mascolo, "Designing Mobility Models based on Social Network Theory", ACM SIGMOBILE Mobile Computing and Communication Review, pp. 59 – 70, 2007
- [14] J. Ott, D. Kutscher and C. Dwertmann, "Integrating DTN and MANET routing", CHANTS '06 Proceedings of the 2006 SIGCOMM workshop on Challenged networks, pp. 221-228 , 2006
- [15] Perkins, C.E. and Royer, E.M., "Ad-hoc on-demand distance vector routing", Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop, pp 90 – 100, 1999

- [16] C.Perkins and P.Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers”, SIGCOMM '94 Proceedings of the conference on Communications architectures, protocols and applications, pp. 234-244, 1994
- [17] T. Spyropoulos , K. Psounis and Cauligi S. Raghavendra, “Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks”, Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, pp. 252-259, 2005
- [18] Vahdat , D. Becker, “Epidemic routing for partially-connected ad hoc networks”, Technical Report CS-200006, Duke University, 2000
- [19] Whitbeck, J. and Conan, V., “HYMAD: Hybrid DTN-MANET routing for dense and highly dynamic wireless networks”, Journal Computer Communications, Volume 33 Issue 13, pp. 1483-1492, 2010
- [20] Zhang, X. and Jacob, L., “Adapting zone routing protocol for heterogeneous scenarios in ad hoc networks”, Parallel Processing, 2003. Proceedings. 2003 International Conference, pp. 341-348, 2003
- [21] Z. Zhang, “Routing in Intermittently Connected Mobile Ad Hoc Networks and Delay Tolerant Networks: overview and Challenges”, Communications Surveys & Tutorials, IEEE, pp. 24–37, 2006
- [22] Y. Zhong and D. Yuan, “Dynamic source routing protocol for wireless ad hoc networks in special scenario using location information”, Communication Technology Proceedings, 2003. ICCT 2003. International Conference on (Volume:2), pp. 1287-1290, 2003

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ

Η Αναστασία Κόντου γεννήθηκε το έτος 1987 στην πόλη των Ιωαννίνων. Το 2005 εισήχθη στο τμήμα Ψηφιακών Συστημάτων του Πανεπιστημίου Πειραιώς. Το 2010 αποφοίτησε και συνέχισε τις σπουδές της στο μεταπτυχιακό πρόγραμμα σπουδών του τμήματος Πληροφορικής του Πανεπιστημίου Ιωαννίνων.

