

**Τμήμα
Μηχανικών
Πληροφορικής τ.ε.**
Τεχνολογικό Εκπαιδευτικό Ίδρυμα
Δυτικής Ελλάδας

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

“ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ MESH”

ΦΟΙΤΗΤΗΣ: ΑΠΟΣΤΟΛΟΥ ΝΙΚΟΛΑΟΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Δρ. ΒΑΣΙΛΕΙΟΣ ΤΡΙΑΝΤΑΦΥΛΛΟΥ

ΑΝΤΙΡΡΙΟ 2015

Περιεχόμενα

Περιεχόμενα.....	2
Γενικό πλαίσιο – Σκοπός της πτυχιακής εργασίας.....	4
ΚΕΦΑΛΑΙΟ 1.ΑΡΧΙΤΕΚΝΟΚΕΣ.....	5
1.1 Εισαγωγή και εφαρμογές των ασύρματων δικτύων Mesh.....	6
1.2 Σύγκριση μεταξύ ασύρματων AD HOC και MESH δικτύων.....	8
1.3 Προκλήσεις στα ασύρματα MESH δίκτυα.....	11
1.3.1 Διεκπεραιωτικότητα και χωρητικότητα σε ένα WMN με string τοπολογία.....	11
1.3.2 Διεκπεραιωτικότητα δικαιοσύνης.....	12
1.3.3 Αξιοπιστία και Στιβαρότητα.....	14
1.3.4 Διαχείριση πηγών.....	14
1.4 Θέματα σχεδιασμού στα ασύρματα δίκτυα mesh.....	15
1.4.1 Αρχιτεκτονικά θέματα σχεδιασμού δικτύου.....	15
1.4.1.1 Flat ασύρματο δίκτυο mesh.....	15
1.4.1.2 Hierarchical ασύρματο δίκτυο mesh.....	16
1.4.1.3 Hybrid ασύρματο δίκτυο mesh.....	16
1.4.2 Θέματα σχεδιασμού πρωτοκόλλου δικτύου.....	16
1.4.2.1 Θέματα σχεδιασμού φυσικού επιπέδου.....	17
1.4.2.2 Επίπεδο ελέγχου πρόσβασης μέσου.....	18
1.4.2.3 Επίπεδο δικτύου.....	18
1.4.2.4 Επίπεδο μεταφοράς.....	19
1.4.2.5 Επίπεδο εφαρμογής.....	20
1.4.2.6 Θέματα σχεδιασμού στάθμης συστήματος.....	20
1.5 Θέματα σχεδιασμού στα multi-radio ασύρματα δίκτυα mesh.....	20
1.5.1 Αρχιτεκτονικά θέματα σχεδιασμού.....	21
1.5.2 Θέματα σχεδιασμού ελέγχου πρόσβασης μέσου.....	22
1.5.3 Θέματα σχεδιασμού πρωτοκόλλου δρομολόγησης.....	23
1.5.4 Θέματα σχεδιασμού μετρικής δρομολόγησης.....	25
1.5.5 Θέματα σχεδιασμού τοπολογίας.....	27
ΚΕΦΑΛΑΙΟ 2.Η ΔΡΟΜΟΛΟΓΗΣΗ ΣΤΑ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ MESH.....	28
2.1 Εισαγωγή	29
2.2 Ιδιότητες των ασύρματων δικτύων MESH.....	30
2.3 Γενικές έννοιες πρωτοκόλλων δρομολόγησης.....	31
2.3.1 Ταξινόμηση των πρωτοκόλλων δρομολόγησης.....	31
2.3.2 Η δρομολόγηση στο επίπεδο 2.....	33
2.3.3 Απαιτήσεις δρομολόγησης στα ασύρματα MESH δίκτυα.....	34
2.3.4 Πολύδρομική δρομολόγηση για υπόλοιπο φόρτου και ανοχή βλαβών..	34
2.3.5 QoS δρομολόγηση.....	35
2.4 Μετρικές δρομολόγησης.....	35
2.5 Πρωτόκολλα δρομολόγησης.....	37
2.5.1 Ad hoc On-demand Distance Vector πρωτόκολλο δρομολόγησης.....	37
2.5.2 Dynamic Source Routing Protocol (DSR).....	39
2.5.3 Optimized Link State Routing Protocol (OLSR).....	40
2.5.4 Cross-Layer προσέγγιση δρομολόγησης.....	42

2.5.5	Ενήμερο εύρος ζώνης δρομολόγησης.....	42
2.5.6	MR-LQSR πρωτόκολλο.....	43
2.5.7	Άλλα πρωτόκολλα δρομολόγησης βασισμένη-τοπολογία για WMNs.....	44
2.5.8	Πρωτόκολλα δρομολόγησης βασισμένης-θέσης.....	45
2.6	Προτεινόμενη δρομολόγηση για IEEE 802.11s WLAN mesh networking.....	46
2.6.1	Μετρική δρομολόγησης ραδιοχρόνου.....	47
2.6.2	Hybrid Wireless Mesh Protocol (HWMP).....	47
2.6.3	Radio Aware Optimized Link State Routing (RA-OLSR).....	50
2.6.4	Επεκτασιμότητα.....	51
2.7	Κοινή δρομολόγηση και εκχώρηση καναλιού.....	51
2.7.1	Συνδυασμένη Load-aware δρομολόγησης και εκχώρησης καναλιού.....	52
2.7.2	Κοινή LP-based δρομολόγηση και εκχώρηση καναλιού.....	53
2.7.3	Μοντέλα και παραδοχές.....	54
2.8	Προοπτικές και ανοικτά ζητήματα.....	55
ΚΕΦΑΛΑΙΟ 3.ΤΟ ΠΡΟΤΖΕΚΤ ΤΗΣ ΠΤΥΧΙΑΚΗΣ.....		57
3.1	NS2 versus NS3.....	58
3.2	AODV versus HWMP.....	60
3.3	Υλοποίηση πρότζεκτ.....	61
3.3.1	Υλοποίηση AODV.....	62
3.3.2	Υλοποίηση HWMP.....	63
3.4	Τα πειράματα του πρότζεκτ.....	64
3.5	NS-3 Scripts.....	67
ΑΝΑΦΟΡΕΣ-BIBΛΙΟΓΡΑΦΙΑ.....		88

Γενικό πλαίσιο – Σκοπός της πτυχιακής εργασίας

Η παρούσα πτυχιακή εργασία επικεντρώνεται στη μελέτη των Δικτύων MESH. Η ασύρματη επικοινωνία δεδομένων, η οποία εξελίσσεται και τεχνολογικά αλλά και στη χρήση/διάδοσή της, είναι μια κινητήρια δύναμη χάρη στο Internet και την επιτυχία της τέταρτης γενιάς των δικτύων με κυψέλες. Στο άμεσο μέλλον, ο ρόλος και οι δυνατότητες της ανταλλαγής δεδομένων σε κοντινή απόσταση αναμένεται να μεγαλώσουν, και να εξυπηρετούν σαν συμπλήρωμα στην παραδοσιακή μεγάλης κλίμακας επικοινωνία. Ακόμη είναι μια ριζοσπαστική δικτυακή μορφή των ασύρματων δικτύων που εξελίσσονται συνεχώς, αποκλίνουν όμως από τα συνηθισμένα κεντρικά ασύρματα συστήματα όπως τα κυτταρικά δίκτυα και τα LANs. Οι πελάτες μπορεί να είναι φορητοί υπολογιστές (laptop), κινητά τηλέφωνα ή tablets. Τα οφέλη των κατακεκομμένων δικτύων πλέγματος είναι πολλαπλά: είναι φθηνά, είναι εύκολα να εγκατασταθούν, είναι γρήγορα, και μπορούν να είναι ιδιωτικά. Αυτό σημαίνει βελτιωμένο εύρος ζώνης, βελτιωμένο λανθάνων χρόνο, βελτιωμένη ασφάλεια και λιγότερες αναταραχές πάνω από την ουδετερότητα του δικτύου. Εξαιτίας αυτών των πλεονεκτημάτων έχουν επίσης αποδειχθεί δημοφιλείς στην ανάπτυξη της διασυνδεσιμότητας σε πιο απομακρυσμένες περιοχές του πλανήτη. Ο σκοπός αυτής της πτυχιακής εργασίας είναι η σύγκριση ανάμεσα σε δύο πρωτόκολλα του AODV και HWMP μέσα από ένα αριθμό σεναρίων καθώς και η σύγκριση της απόδοσης που έχει κάθε αρχιτεκτονική αντίστοιχα.

Abstract

This thesis focuses on the study of MESH Networks. The wireless data communication, which is evolving technologically but also grows in use / dissemination, is a driving force thanks to the Internet and the success of fourth-generation cellular networks. In the near future, the role and the possibilities of data exchange within walking distances are expected to grow and to serve as a complement in traditional large-scale communication. Even a radical network form of wireless networks that are evolving, deviate from the usual centralized wireless systems such as cellular networks and LANs. Clients can be laptops (laptop), mobile phones or tablets. The benefits of distributed mesh network are manifold such as: it is inexpensive, it is easily installed, it is fast, and may be private. As an outcome, it has improved bandwidth, improved latency, improved safety and fewer disturbances over the net neutrality. Because of these advantages, they have been proven popular in the development of connectivity in remote areas of the planet. The purpose of this thesis is the comparison between two protocols which are the AODV and the HWMP through a number of scenarios and also the comparison of the performance that every architecture respectively has.

ΚΕΦΑΛΑΙΟ 1

ΑΡΧΙΤΕΚΝΟΝΙΚΕΣ

ΕΙΣΑΓΩΓΗ

1.1 Εισαγωγή και εφαρμογές των Ασύρματων Δικτύων Mesh

Τα ασύρματα δίκτυα πλέγματος έχουν τη δυνατότητα να παρέχουν ευρυζωνική πρόσβαση στο Διαδίκτυο, ασύρματο τοπικό δίκτυο κάλυψης και συνδετικότητας δικτύου για σταθερούς ή κινητούς κόμβους σε χαμηλό κόστος τόσο για τους φορείς εκμετάλλευσης δικτύου όσο και για τους πελάτες. Η βασική τεχνολογία περιλαμβάνει ένα δίκτυο ασύρματων routers αναμετάδοσης πακέτων μεταξύ τους με έναν πολλαπλών μεταδόσεων τρόπο. Αναδυόμενες εταιρείες σε αυτόν τον ανερχόμενο τομέα προσφέρουν διαφορετικές παραλλαγές και επιλογές υλοποίησής τους για την καθημερινή εφαρμογή. Η ασύρματη δικτύωση πλέγματος είναι μια πολλά υποσχόμενη τεχνολογία για πολλές εφαρμογές που απευθύνεται ειδικά σε εκείνες τις εφαρμογές που δεν μπορούν να υποστηριχθούν άμεσα από άλλες ασύρματες τεχνολογίες. Η εμπορική επιτυχία του Wi-Fi και τις εξελίξεις σε πολλές ασύρματες τεχνολογίες έχουν εν μέρει τονώσει την ανάπτυξη των ασύρματων δικτύων πλέγματος. Η επίμονη κινητήρια δύναμη, όμως, προέρχεται από τα πλεονεκτήματα των ίδιων των ασύρματων mesh τεχνικών, συμπεριλαμβανομένων εκτεταμένη κάλυψη, στιβαρότητα, αυτο-διάθρωση, εύκολη συντήρηση, και το χαμηλό κόστος. Ασύρματα δίκτυα πλέγματος μπορούν να υλοποιηθούν με διάφορες ασύρματες τεχνολογίες, συμπεριλαμβανομένων 802.11, 802.15, 802.16, κυψελοειδή τεχνολογίες ή συνδυασμούς περισσότερων από έναν τύπο. Πολλά mesh δίκτυα λειτουργούν σε πολλές ράδιο-μπάντες.

Για παράδειγμα Firetide και Wave Relay δίκτυα πλέγματος έχουν τη δυνατότητα να επικοινωνούν κόμβο σε κόμβο για 5,2 GHz ή 5,8 GHz, αλλά επικοινωνούν κόμβο με πελάτη σε 2,4GHz(802.11).

- Το SolarMESH project εξέτασε τη δυνατότητα να τροφοδοτούν τα δίκτυα 802.11 με βάση πλέγμα χρήση ηλιακής ενέργειας και επαναφορτιζόμενες μπαταρίες. Το 802.11s τυποποίησης IEEE γίνονται προσπάθειες θεωρώντας επιλογές εξοικονόμησης ενέργειας, αλλά και ηλιακή ενέργεια εφαρμογές μπορεί να περιλαμβάνει μόνο ράδιο κόμβους όπου relay-link εξοικονόμησης ενέργειας θα είναι ανεφάρμοστο.
- SMesh είναι ένα 802.11 multi-hop ασύρματο δίκτυο πλέγματος που αναπτύχθηκε από Distributed System και Networks Lab στο Πανεπιστήμιο Johns Hopkins. Ένα γρήγορο σχήμα μεταβιβάσεως επιτρέπει τους κινητούς πελάτες να περιφέρονται στο δίκτυο χωρίς διακοπή στη συνδεσιμότητα, ένα χαρακτηριστικό κατάλληλο για εφαρμογές real-time, όπως VoIP.
- FabFi είναι ένα open-source, city-scale, σύστημα ασύρματης δικτύωσης πλέγματος αρχικά αναπτύχθηκε το 2009 στο Τζαλαλαμπάντ, στο Αφγανιστάν για την

παροχή Internet υψηλής ταχύτητας σε τμήματα της πόλης και έχει σχεδιαστεί για υψηλές επιδόσεις σε πολλαπλά hops.

- Είναι ένα φθηνό πλαίσιο για την ανταλλαγή ασύρματου internet από ένα κεντρικό πάροχο σε ένα χωριό ή μια πόλη. Μια δεύτερη μεγαλύτερη εφαρμογή ακολούθησε ένα χρόνο αργότερα, κοντά στο Ναϊρόμπι, Κένυα με χαμηλόκοστο μοντέλο αμοιβής για την υποστήριξη της ανάπτυξης του δικτύου. Και τα δύο έργα έγιναν από τους χρήστες FabLab των αντίστοιχων πόλεων.

- Η Ναυτική Μεταπτυχιακή Σχολή, Monterey CA, όπως απέδειξαν τα ασύρματα δίκτυα πλέγματος για την ασφάλεια των συνόρων. Σε ένα πιλοτικό σύστημα, εναέριες κάμερες πάνω σε μπαλόνια αναμεταδίδουν σε πραγματικό χρόνο βίντεο υψηλής ανάλυσης για το προσωπικό εδάφους μέσω ενός δικτύου πλέγματος.

- Το MIT Media Lab έχει αναπτύξει το XO-1 laptop ή "OLPC" (One Laptop per Child) που προορίζεται για τα μειονεκτούντα σχολεία στις αναπτυσσόμενες χώρες και χρησιμοποιεί δικτύωση πλέγματος (με βάση το πρότυπο IEEE 802.11s) για να δημιουργήσει ένα ισχυρό και ανέξοδο υποδομής έργο. Οι στιγμιαίες συνδέσεις πραγματοποιούνται από τους φορητούς υπολογιστές για να μειώσει την ανάγκη για μια εξωτερική υποδομή, όπως το Διαδίκτυο για να φθάσουν σε όλους τους τομείς, επειδή ένας συνδεδεμένος κόμβος θα μπορούσε να μοιραστεί τη σύνδεση με τους κόμβους σε κοντινή απόσταση. Μια παρόμοια ιδέα έχει επίσης τεθεί σε εφαρμογή από Greenpacket με την εφαρμογή της, που ονομάζεται SONbuddy.

- Στο Cambridge, UK, στις 3 Ιουνίου 2006, η δικτύωση πλέγματος χρησιμοποιήθηκε στο "Strawberry Fair" για να εκτελεστεί ζωντανή τηλεόραση,ραδιόφωνο και Internet services σε περίπου 80.000 άτομα.

- Το project Champaign-Urbana Community Wireless Network (CUWiN) αναπτύσσει λογισμικό δικτύωσης πλέγματος με βάση εφαρμογών ανοιχτού κώδικα του Hazy-Sighted Link State Routing Protocol και μετρικής Expected Transmission Count. Επιπλέον, ο Wireless Networking Group στο Πανεπιστήμιο του Illinois στο Urbana-Champaign αναπτύσσουν μια πολυκαναλική, πολυ-ραδιοεπικοινωνία ασύρματου πλέγματος πλατφόρμα δοκιμών, ονομάζεται Net-X εφαρμόζουν ορισμένα από τα πολυκαναλικά πρωτόκολλα που αναπτύσσονται σε αυτή την ομάδα. Οι εφαρμογές που βασίζονται σε μια αρχιτεκτονική που επιτρέπει σε κάποια από τις ραδιοεπικοινωνίες για να αλλάξετε κανάλια για να διατηρήσετε τη συνδεσιμότητα δικτύου, και περιλαμβάνει πρωτόκολλα για την κατανομή των καναλιών και δρομολόγησης.

1.2 Σύγκριση μεταξύ ασύρματων AD HOC και MESH δικτύων

Mesh δίκτυο μπορεί να είναι ένα οποιοδήποτε ασύρματο δίκτυο που έχει δικτυακή τοπολογία είτε μερικής είτε πλήρης πλέγματος. Εξαιτίας της μερικής τοπολογίας πλέγματος ένα mesh χρησιμοποιεί multihop μετάδοση παρόμοια με τα ad hoc δίκτυα. Παρόλο που τα ad hoc είναι παρόμοια με τα mesh τα πρωτόκολλα και η αρχιτεκτονική ενός ad hoc έχουν κακή επίδοση όταν εφαρμόζονται στα mesh. Επιπλέον, τα κριτήρια σχεδιασμού είναι διαφορετικά για τα δύο δίκτυα. Αυτές οι διαφορές στον σχεδιασμό κυρίως κατάγονται από την εφαρμογή ή την εξάπλωση των αντικειμένων και από τους περιορισμούς στους πόρους μέσα σ' αυτά τα δίκτυα.

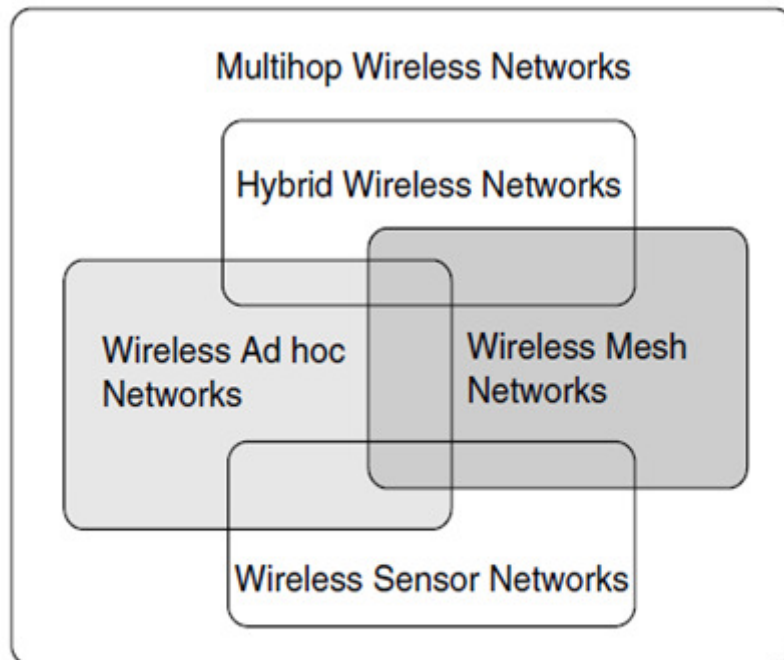
Παράδειγμα ένα ad hoc είναι σε γενικές γραμμές σχεδιασμένο για περιβάλλον υψηλού multihop κινητικότητας από την άλλη πλευρά ένα mesh είναι σχεδιασμένο για ένα στατικό ή περιορισμένης κινητικότητας περιβάλλον. Τα mesh είναι πιο πλούσια σε πόρους σε σύγκριση με τα ad hoc. Σε μερικές εφαρμογές του mesh το δίκτυο μπορεί να έχει μια συγκεκριμένη τοπολογία οπότε το πρωτόκολλο και οι αλγόριθμοι θα πρέπει να σχεδιαστούν για να επωφεληθούν από τέτοιες τοπολογίες.

<i>Issue</i>	<i>Wireless Ad Hoc Networks</i>	<i>Wireless Mesh Networks</i>
Network topology	Highly dynamic	Relatively static
Mobility of relay nodes	Medium to high	Low
Energy constraint	High	Low
Application characteristics	Temporary	Semipermanent or permanent
Infrastructure requirement	Infrastructureless	Partial or fully fixed infrastructure
Relaying	Relaying by mobile nodes	Relaying by fixed nodes
Routing performance	Fully distributed on-demand routing preferred	Fully distributed or partially distributed with table-driven or hierarchical routing preferred
Deployment	Easy to deploy	Some planning required
Traffic characteristics	Typically user traffic	Typically user and sensor traffic
Popular application scenario	Tactical communication	Tactical and civilian communication

- Network topology στα ad hoc είναι υψηλά δυναμική ενώ στα mesh είναι στατική έως ελάχιστα κινητική γιατί στα mesh χρησιμοποιούμε την backbone τοπολογία.

- Mobility of relay nodes στα ad hoc δίκτυα είναι μεσαία έως υψηλή ενώ στα mesh είναι στατική έως ελάχιστα κινητή γιατί χρησιμοποιούμε μια backbone τοπολογία όπου έχουμε κατευθυντικές κεραίες.
- Energy constraint (Ο περιορισμός ενέργειας) στα ad hoc είναι υψηλός ενώ στα mesh είναι χαμηλός λόγω της στατικής τοπολογίας και της στατικότητας των κόμβων.
- Application characteristics στα ad hoc είναι προσωρινό ενώ στα mesh είναι ήμι-μόνιμο ή μόνιμο γιατί έχουμε στατική τοπολογία,στατικούς κόμβους, χρησιμοποιείται κυρίως το 802.11, η μέριμνα των κόμβων είναι η ενεργειακή κατανάλωση,μέσω του δικτύου κορμού που έχουμε πρόσβαση στο internet και σύνδεση με άλλα ασύρματα δίκτυα,βελτιώνουν την επίδοση του δικτύου με την εύελικτη αρχιτεκτονική και μηχανισμούς ρύθμισης επέκτασης και ανοχής σε δυσλειτουργίες, multihop επικοινωνία οπότε υψηλό throughput λιγότερες παρεμβολές αποδοτικότερη χρησιμοποίηση συχνότητας.
- Infrastructure requirement (απαίτηση υποδομής) τα ad hoc δεν έχουν ενώ τα mesh έχουν μερική ή πλήρης σταθερή υποδομή λόγω της στατικής τοπολογίας και της στατικότητας των κόμβων.
- Relaying (αναμετάδοση) στα ad hoc γίνεται από κινητούς κόμβους ενώ στα mesh γίνεται από σταθερούς κόμβους λόγω της στατικότητας των κόμβων και λόγω της τοπολογίας backbone.
- Routing performance (επίδοση δρομολόγησης) στα ad hoc προτιμάται η πλήρως κατανεμημένη δρομολόγηση ενώ στα mesh προτιμάται η πλήρως κατανεμημένη ή μερικώς κατανεμημένη με πίνακα δρομολόγησης ή ιεραρχική δρομολόγηση λόγω του αλγόριθμου δρομολόγησης djikstra που προτιμάται στα δίκτυα mesh αλλά και ότι έχουν καλύτερη επίδοση.
- Deployment (ανάπτυξη) τα ad hoc είναι εύκολα να αναπτυχθούν ενώ τα mesh απαιτείται λίγος σχεδιασμός γιατί το δίκτυο βασίζεται σε μια τοπολογία και πρέπει να πάρουμε κάποιες παραμέτρους.

- Traffic characteristics (χαρακτηριστικά κίνησης) στα ad hoc γίνεται τυπικά η κίνηση του χρήστη ενώ στα mesh γίνεται τυπικά η κίνηση του χρήστη αλλά και του κόμβου.
- Popular application scenario (δημοφιλέσ σενάριο εφαρμογής) στα ad hoc είναι η στρατηγική επικοινωνία ενώ στα mesh έχουμε στρατηγική και πολιτική επικοινωνία λόγω του δωρεάν Internet που παρέχεται στους πολίτες αλλά και την στρατιωτική εφαρμογή τους.



Classification of multihop wireless networks.

1.3 Προκλήσεις στα ασύρματα MESH δίκτυα

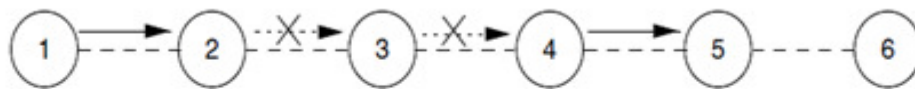
Τα παραδοσιακά ασύρματα δίκτυα ad hoc και WMNs ήταν βασισμένα σε μια single-channel ή single-radio διασύνδεση. Τα WMNs έχουν μια υψηλή ανεκτικότητα στα λάθη αλλά αντιμετωπίζουν έναν περιορισμό περιορισμένης δικτυακής χωρητικότητας. Το throughput capacity γίνεται απαράδεκτα χαμηλό όταν έχουμε αυξημένο αριθμό κόμβων. Για να βελτιωθεί το throughput capacity ενός WMNs είναι η χρησιμοποίηση multiple radio interfaces(διασυνδέσεις). Το throughput capacity μπορεί να αυξηθεί με την χρήση multiple interfaces και από τον καλό συντονισμό των πρωτοκόλλων.

1.3.1 Διεκπεραιωτικότητα και χωρητικότητα σε ένα WMN με string τοπολογία

Table 1.2 Throughput Degradation in a WMN with String Topology

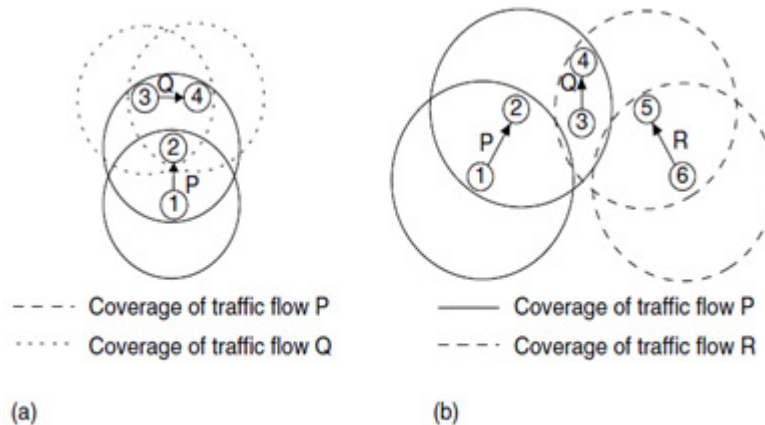
	<i>1 Hop</i>	<i>2 Hops</i>	<i>3 Hops</i>	<i>4 Hops</i>	<i>5 Hops</i>	<i>>5 Hops</i>
Normalized throughput	1	0.47	0.32	0.23	0.15	0.14
$\frac{1}{\text{Hoplength}}$	1	0.5	0.33	0.25	0.2	0.16

Όταν ο κόμβος 1 προσπαθεί να μεταδώσει στον κόμβο 2, ειδικά όταν χρησιμοποιούν το βασικό MAC πρωτόκολλο που βασίζεται στην μέθοδο CSMA-CA, οι κόμβοι 2 και 3 δεν μπορούν να ξεκινήσουν άλλη μια μετάδοση. Στα περισσότερα WMNs έχουμε half-duplex μετάδοση έτσι ο κόμβος 2 δεν έχει ταυτόχρονη μετάδοση και είναι εγκλωβισμένος στην συνεχόμενη μετάδοση μεταξύ του κόμβου 1 και 2, οπότε έχουμε χαμηλό throughput στα WMNs σε διαδρομή αναμετάδοσης multihop. Μια μετάδοση two-hop ανάμεσα στον 1 και 3 πρέπει να μοιραστεί το bandwidth μεταξύ των δύο οπότε το end-to-end throughput για two-hop διαδρομή είναι 47% της single-hop throughput.



An example of string topology and exposed node problem in a wireless mesh network.

1.3.2 Διεκπεραιωτικότητα δικαιοσύνης



Traffic flows and throughput unfairness in WMNs.

Οι εικόνες δείχνουν άπλες τοπολογίες μέσα σε ένα WMN που προκαλούν ανισότητα throughput.

Η πρώτη εικόνα δείχνει την ανισότητα throughput ανάμεσα στους κόμβους. Αυτό συμβαίνει σ'ένα single-radio WMN, εάν όλοι οι κόμβοι έχουν ίσο throughput κάτω από παρόμοιες καταστάσεις source traffic και network load. Έχουμε υψηλή ανισότητα throughput ανάμεσα στις διαμαχώμενες ροές κίνησης όταν το CSMA/CA βασισμένο σε MAC protocol χρησιμοποιούνται για να λυθεί η διαμάχη.

Δυο σημαντικές ιδιότητες που σχετίζονται με CSMA/CA βασισμένο σε MAC protocol σένα WMN είναι:

1. Ασυμμετρία πληροφοριών στην πρώτη εικόνα
2. Location-dependent(τοποθεσία-εξαρτημένη) διαμάχης στην δεύτερη εικόνα

Όταν ο κόμβος 1 έχει πακέτα για μετάδοση, ανιχνεύει ένα κενό κανάλι, μπορεί να αρχίσει την μετάδοση στέλνοντας RTS πακέτα στον κόμβο 2. Σε αυτό το σημείο, εάν υπάρχει μια συνεχής μετάδοση ανάμεσα στους κόμβους 3 και 4, ο κόμβος 2 δεν ανταποκρίνεται στο RTS, αφήνοντας τον κόμβο 1 εκθετικά να κάνει πίσω και να ξαναπροσπαθήσει. Οι επαναλαμβανόμενες οπισθοχωρήσεις και οι αρκετές προσπάθειες αναμετάδοσης οδηγούν στο να έχουμε χαμηλό throughput ανάμεσα στους κόμβους 3 και 4. Η ασυμμετρία πληροφοριών προκαλείται από έλλειψη πληροφοριών σε ορισμένους κόμβους, οι υπερβολικές πληροφορίες μπορούν να συμβάλουν επίσης στην ανισότητα throughput.

Στην εικόνα 1(b) οι ροές P και R, δεν έχουν πληροφορίες για οποιεσδήποτε άλλες ροές στο δίκτυο όπου η ροή Q έχει πληροφορίες και για τις δύο άλλες ροές. Η ροή Q πρέπει να ρυθμίσει το vector(NAV) και να απέχει από μετάδοση, όποτε βλέπει μετάδοση πακέτων έλεγχου ή πακέτων δεδομένων που ανήκουν στις ροές P και R. Η ροή Q περιμένει για ένα

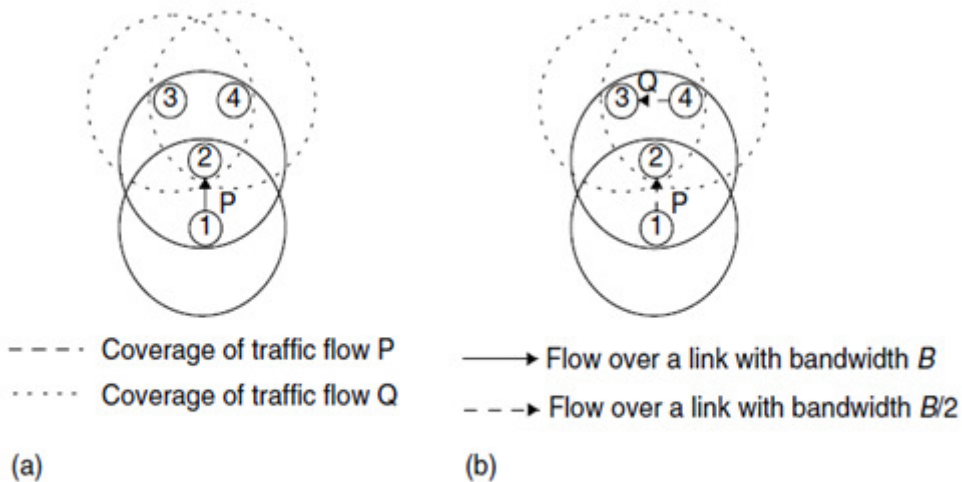
κενό κανάλι που εξαρτάται από την περίπτωση που και οι 2 ροές P και R συγχρόνως γίνουν κενές. Η θέση της ροής Q έχει πολύ περισσότερο ανταγωνισμό από τις υπόλοιπες ροές.

Άλλο ένα αποτέλεσμα από την location-dependent διαμάχη είναι perceived collision(αντιληπτή σύγκρουση) η οποία μπορεί να συμβεί στην ροή Q. Εξαιτίας της παρουσίας ανταγωνιζομένων ροών,η προσπάθεια να έχουμε πρόσβαση στο κανάλι,η ταυτόχρονη μετάδοση πακέτων ελέγχου,RTS-CTS-ACK και από τις 2 ροές P και R,μπορεί να έχει ως αποτέλεσμα μια σύγκρουση στη ροή Q. Αυτό το αποτέλεσμα είναι μιας λάθος αντίληψης της σύγκρουσης στην ροή Q, που στην πραγματικότητα μπορεί να μην είναι σύγκρουση μια σύγκρουση και για τις 2 ροές P και R. Αυτή η αντιληπτή σύγκρουση μπορεί να μειώσει την ποσότητα πληροφοριών,για τις ροές P και R,διαθέσιμες στην ροή Q,οπότε οδηγεί σε μεγαλύτερη υποβάθμιση της ισότητας throughput.

Εξάλλου η ασυμμετρία πληροφοριών και η διαμάχη location-dependent,η half-duplex ενός single-interface συστήματος είναι άλλη μια ιδιότητα που προκαλεί υψηλή ανισότητα throughput σε ένα single-radio WMN. Εξαιτίας των χαρακτηριστικών της half-duplex, κανένας κόμβος δεν μπορεί σύγχρονος να λάβει και να μεταδώσει.

Αυτό απεικονίζεται στην εικόνα 2(A) στην οποία ένα single half-duplex radio,μόνο μια επικοινωνία μπορεί να επιτραπεί οποιαδήποτε στιγμή. Όποτε η ροή P μεταδίδει καθώς οι άλλοι κόμβοι περιμένουν.

Στην εικόνα 2(B) ο κάθε κόμβος χρησιμοποιεί 2 radio interfaces,οπότε 2 ροές συγχρόνως P και Q μπορούν να υπάρξουν. Σε αυτήν την περίπτωση παρόλο που το κάθε κανάλι έχει το μισό bandwidth η throughput fairness αυξάνεται όπως ανακαλύπτεται σε πειραματικές μελέτες.



Half-duplex radio interfaces in WMNs.

1.3.3 Αξιοπιστία και Στιβαρότητα

Άλλο ένα σημαντικό κίνητρο να χρησιμοποιήσεις WMN και ειδικότερα MR-WMN είναι να βελτιώσουμε την αξιοπιστία και την στιβαρότητα της επικοινωνίας. Η μερική τοπολογία mesh(πλέγματος) σ'ένα WMN παρέχει υψηλή αξιοπιστία και διαμερισμό διαδρομής κατά των κόμβων και αποτυχίες σύνδεσης. Τα MR-WMNs παρέχουν το πιο σημαντικό συστατικό για στιβαρότητα σε διαφορετικότητα-επικοινωνιών. Στα ασύρματα συστήματα τα σφάλματα καναλιού μπορεί να είναι πάρα πολλά σε σύγκριση με τα καλωδιακά δίκτυα,οπότε η σταδιακή υποβάθμιση της ποιότητας της επικοινωνίας κατά την διάρκεια των υψηλών σφαλμάτων καναλιού είναι απαραίτητη.

Αυτό είναι σημαντικό όταν το WMN σύστημα χρησιμοποιεί χωρίς άδεια φάσμα συχνοτήτων. Για να αποκτήσουμε αναβάθμιση ποιότητας αντί για πλήρης έλλειψη συνδετικότητας,τα WMNs μπορούν να χρησιμοποιήσουν ποικιλομορφία συχνότητας,χρησιμοποιώντας multiple radio interfaces,το οποίο είναι δύσκολο να γίνει σε ένα single-radio WMN system. Τα MR-WMNs μπορούν να χρησιμοποιήσουν κατάλληλα μοντέλα radio-switching για να αποκτήσουν ανεκτικότητα σφάλματος στην επικοινωνία είτε αλλάζοντας τα radio,channels, ή χρησιμοποιώντας ταυτόχρονος multiple radio.

1.3.4 Διαχείριση πηγών

Η διαχείριση πηγών αναφέρεται στην αποτελεσματική διαχείριση των δικτυακών πηγών όπως η ενεργεία,bandwidth,συνδέσεις και αποθήκευση. Οι πηγές ενεργείας μπορούν να χρησιμοποιούνται αποτελεσματικά σ'ένα WMN με περιορισμένο απόθεμα ενέργειας εάν ο κάθε κόμβος στο σύστημα έχει μια καινούργια low-power διεπαφή επιπροσθέτως με την κανονική διεπαφή. Η συνολική κατανάλωση ενέργειας ακόμα και σε κατάσταση αναμονής,εξαρτάται από τον τύπο της διεπαφής. Οπότε σ'ένα WMN βασισμένο σε 802,11 με περιορισμένο απόθεμα ενέργειας,με low-power και low-data διεπαφή ρυθμού μπορεί να μεταφέρει out-of-band πληροφορία σηματοδότησης για να ελέγξει την high-data διεπαφή ρυθμού δεδομένων. Πηγές bandwidth μπορούν να διαχειρίζονται σ'ένα περιβάλλον multriradio.

Η εξισορρόπηση φορτίου(load balancing)σε πολλαπλές διεπαφές μπορούν να βοηθήσουν στην εμπόδιση οποιοδήποτε συγκεκριμένου καναλιού από το να έχει βαριά συμφόρηση και ως εκ τούτου να γίνεται μπουτλιάρισμα. Εκτός από την εξισορρόπηση του φορτίου,το bandwidth που αποκτάται μέσα από την κάθε διεπαφή μπορεί να συσσωρευτεί για να αποκτήσουμε ένα υψηλό βαθμό αποτελεσματικού ρυθμού δεδομένων. Σε έναν τέτοιο μηχανισμό συσσώρευσης bandwidth(or bandwidth stripping), ένα δυναμικό πακέτο προγραμματισμού μπορεί να χρησιμοποιηθεί για να αποκτήσουμε μια καλύτερη επίδοση. Ένα πλεονέκτημα ενός multi-radio σε WMN είναι η πιθανότητα να πραγματοποιήσουμε ποιότητα παροχής υπηρεσίας μέσα από την διαφοροποίηση υπηρεσίας.

1.4 Θέματα σχεδιασμού στα ασύρματα δίκτυα mesh

Υπάρχουν πολλά θέματα που χρειάζονται σκέψη όταν ένα WMN σχεδιάζεται για μια συγκεκριμένη εφαρμογή. Αυτά τα θέματα σχεδιασμού μπορούν ευρέως να ταξινομηθούν σε αρχιτεκτονικά θέματα και θέματα πρωτοκόλλου.

1.4.1 Αρχιτεκτονικά θέματα σχεδιασμού δικτύου

Ένα WMN μπορεί να σχεδιαστεί με 3 διαφορετικές αρχιτεκτονικές δικτύου βασισμένες στην δικτυακή τοπολογία:

1. Επίπεδο (flat) WMN
2. Ιεραρχικό (hierarchical) WMN
3. Υβριδικό (hybrid) WMN

1.4.1.1 Flat ασύρματο δίκτυο mesh

Στο Flat WMN, το δίκτυο σχηματίζεται από μηχανήματα πελατών (client machines) που δρουν ως hosts και δρομολογητές. Ο κάθε κόμβος είναι στο ίδιο επίπεδο με τους ομότιμους του. Οι ασύρματοι client nodes συντονίζονται μεταξύ τους για να παρέχουν δρομολόγηση, σχηματισμό δικτύου, παροχή υπηρεσιών, και άλλες παροχές εφαρμογής. Αυτή η αρχιτεκτονική είναι πιο κοντά σε ένα ασύρματο δίκτυο ad hoc και είναι η πιο απλή περίπτωση ανάμεσα σε 3 WMN αρχιτεκτονικές. Το κύριο πλεονέκτημα της είναι η απλότητα της και τα μειονεκτήματά της είναι η έλλειψη network scalability (κλιμακοθετησιμότητα δικτύου) και υψηλούς περιορισμούς πόρων.

Τα κύρια θέματα στο σχεδιασμό ενός flat WMN είναι το σχέδιο διευθυνσιοδότησης, η δρομολόγηση και τα σχέδια αποκάλυψης υπηρεσιών. Σε ένα flat network, η διευθυνσιοδότηση είναι ένα από τα θέματα που μπορεί να γίνει ένα εμπόδιο ενάντια επεκτασιμότητα.

1.4.1.2 Hierarchical ασύρματο δίκτυο mesh

Σένα ιεραρχικό WMN το δίκτυο έχει πολλαπλές στρώσεις ή ιεραρχικά επίπεδα στα οποία οι WMN client nodes σχηματίζουν το χαμηλότερο στην ιεραρχία. Οι client nodes μπορούν να επικοινωνήσουν με ένα backbone WMN δίκτυο που σχηματίζεται από WMN δρομολογητές. Στις περισσότερες περιπτώσεις ,οι WMN nodes είναι αποκλειστικοί κόμβοι που σχηματίζουν ένα backbone WMN δίκτυο. Αυτό σημαίνει ότι οι κόμβοι backbone μπορεί να μην προέρχονται ή να μην τερματίζουν την κίνηση δεδομένων όπως οι WMN client nodes. Η ευθύνη να οργανωθεί από μόνο του και να διατηρήσει το backbone δίκτυο παρέχεται στους WMN δρομολογητές,μερικοί από τους οποίους στο backbone δίκτυο μπορεί να έχουν εξωτερική διεπαφή στο Intranet και τέτοιοι κόμβοι αποκαλούνται ως gateway nodes.

1.4.1.3 Hybrid ασύρματο δίκτυο mesh

Αυτή είναι μια ειδική περίπτωση ιεραρχικών WMN όπου το WMN χρησιμοποιεί άλλα ασύρματα δίκτυα για επικοινωνία. Όπως η χρήση άλλων WMNs υποδομών-βασισμένων σε κυψελοειδής δίκτυα, WiLL δίκτυα, WiMAX δίκτυα, ή δορυφορικά δίκτυα. Παραδείγματα από τέτοια υβρίδια WMNs περιλαμβάνουν multihop κυψελοειδή δίκτυα, throughput βελτιωμένο ασύρματο σε τοπικό βρόχο (TWiLL) δίκτυα, και ενοποιημένο κυψελοειδές ad hoc δίκτυα.

Μια πρακτική λύση για ένα τέτοιο υβρίδιο WMN για επείγουσα ανταπόκριση εφαρμογών είναι η πλατφόρμα CalMesh. Αυτά τα υβρίδια WMNs μπορούν να χρησιμοποιούν πολλαπλές τεχνολογίες και για το WMN backbone και για το back haul. Καθώς η ανάπτυξη των WMN εξαρτάται κυρίως στο πως λειτουργεί με άλλες υπάρχων ασύρματες δικτυακές λύσης, αυτή η αρχιτεκτονική γίνεται πολύ σημαντική στην ανάπτυξη των WMN.

1.4.2 Θέματα σχεδιασμού πρωτοκόλλου δικτύου

Τα θέματα σχεδιασμού για τα πρωτόκολλα μπορούν να περιγράψουν με έναν τρόπο layer-wise ξεκινώντας από το φυσικό επίπεδο έως το επίπεδο εφαρμογής.

1.4.2.1 Θέματα σχεδιασμού φυσικού επιπέδου

Στο φυσικό επίπεδο, το κύριο θέμα σχεδιασμού είναι η επιλογή μιας κατάλληλης ραδιοεπικοινωνία τοπολογίας (radio topology). Η επιλογή μιας ραδιοεπικοινωνία τεχνολογίας μπορεί να βασίζεται:

- i. Τεχνολογική μελέτη
- ii. Οικονομική μελέτη

Η κύρια τεχνολογική μελέτη περιλαμβάνει την φασματική απόδοση, ρυθμός δεδομένων φυσικού επιπέδου, και την ικανότητα να λειτουργούν με την παρουσία των παρεμβολών. Για παράδειγμα η επιλογή τεχνολογιών όπως CDMA, UWB, MIMO είναι πιο κατάλληλα για το φυσικό επίπεδο ενός WMN από τις πιο δημοφιλή τεχνολογίες φυσικού επιπέδου, OFDM χρησιμοποιείται στα σημερινά WMNs. Για παράδειγμα η σημερινή τεχνολογία φυσικού επιπέδου, κυρίως βασίζεται στο OFDM που παρέχει το μεγαλύτερο στο φυσικό επίπεδο ροή δεδομένων στα 54Mbps. Σ' ένα πυκνό δίκτυο με υψηλές παρεμβολές, αυτή η χωρητικότητα μπορεί να μην είναι επαρκής.

Οπότε, η εξέλιξη μιας καινούργιας με υψηλό ρυθμό δεδομένων στο φυσικό επίπεδο όπως το UWB είναι μια πρόκληση στο φυσικό επίπεδο. Εξάλλου η επιλογή μιας ειδικής τεχνολογίας φυσικού επιπέδου, προγραμματιζόμενες ραδιοεπικοινωνίες ή γνωστικές ραδιοεπικοινωνίες προσθέτουν μια άλλη διάσταση στον σχεδιασμό φυσικού επιπέδου ενός WMN. Αυτό τονίζεται από μερικές εφαρμογές των WMNs όπως η επείγουσα ανταπόκριση και στρατιωτικές όπου το φάσμα που χρησιμοποιείται για επικοινωνία εξαρτάται από το αχρησιμοποίητο όπου το φάσμα σε μια συγκεκριμένη τοποθεσία.

Σε τέτοιες εφαρμογές, ένα software-defined ραδιοεπικοινωνίας με γνωστικές ικανότητες θα ήταν μια ιδανική επιλογή. Καθώς και οι τεχνολογικές μελέτες που προαναφέρθηκαν, η δεύτερη πιο σημαντική απαίτηση είναι οικονομική ή κοινωνική όπου η απλότητα τεχνολογίας του φυσικού επιπέδου θα οδηγήσει σε φθηνές συσκευές και οπότε καλύτερη κοινωνική προσιτότητα των WMNs.

Ένα παράδειγμα είναι η σημερινή επιτυχία του 802,11b που βασίζονται τα WMNs όπου οι φθηνές συνδέσεις κάρτες δικτύου συνείσφεραν στην επιτυχία του πολλαπλασιασμού των WMNs. Οπότε, καθώς επιλέγουμε την τεχνολογία φυσικού επιπέδου, ένας σχεδιαστής δικτύου πρέπει να κοιτάζει την εφαρμογή και το σενάριο του χρήστη επίσης.

1.4.2.2 Επίπεδο ελέγχου πρόσβασης μέσου

Ο σχεδιασμός του MAC γραμμής δεδομένων πρωτόκολλο είναι σημαντικό σ'ένα WMN επειδή η χωρητικότητα που αποκτά βασίζεται στην επίδοση του MAC πρωτοκόλλου. Τα βασικά θέματα που αντιμετωπίζει το δημοφιλές CSMA/CA-based στο IEEE 802,11 (DCF-συντονισμός λειτουργίας διανομής)είναι:

- I. Πρόβλημα κριμένου κόμβου
- II. Πρόβλημα εκτεθειμένου κόμβου
- III. Διαμάχη location-dependent
- IV. Υψηλή πιθανότητα λάθους στο κανάλι

Για να αυξήσουμε την χωρητικότητα του δικτύου,χρησιμοποιούνται multiple radios σε multiple channels. Οπότε,τα καινούργια MAC πρωτόκολλα σχεδιάζονται για να λειτουργούν σε multichannel MR-WMN συστήματα. Τα MAC πρωτόκολλα πρέπει επίσης να σχεδιαστούν για να λειτουργούν σε διαφορετικές τεχνολογίες φυσικού επιπέδου όπως UWB και MIMO. Άλλο ένα δημοφιλή θέμα έρευνας για καλύτερη επίδοση MAC είναι η χρήση των cross-layer αλληλεπίδρασης μηχανισμών που διευκολύνουν το MAC πρωτόκολλο να χρησιμοποιεί πληροφορίες από άλλα επίπεδα.

Στα παραδοσιακά ασύρματα ή ενσύρματα δίκτυα,το κάθε επίπεδο λειτουργεί με δικές του πληροφορίες κάνοντάς τα ανέκανα να εκμεταλλευτούν τις ιδιότητες ενός network-centric. Γενικό,η σχεδίαση του MAC γραμμής δεδομένων επιπέδου πρωτοκόλλου πρέπει να περιλαμβάνει μεθόδους και λύσεις για να παρέχει καλύτερη δικτυακή κλιμακοθετησιμότητα(scalability) και χωρητικότητα throughput.

1.4.2.3 Επίπεδο δικτύου

Αντίθετα με τα πρωτόκολλα δρομολόγησης για ασύρματα δίκτυα ad hoc,τα πρωτόκολλα δρομολόγησης ανάλογα με το δικτυακό σενάριο δρομολόγησης, αντιμετωπίζουν θέματα σχεδιασμού σ'ένα WMN. Καθώς το WMN είναι σχετικά ένα στατικό δίκτυο,η δρομολόγηση μπορεί να γίνεται από προσεγγίσεις table-driven όπως στα ενσύρματα δίκτυα ή σε ασύρματα δίκτυα ad hoc.

Τα κύρια θέματα που αντιμετωπίζονται από το πρωτόκολλο δρομολόγησης σ'ένα WMN είναι:

- I. Σχεδιασμό του μετρητή δρομολόγησης
- II. Ελάχιστη επιβάρυνση δρομολόγησης
- III. Ευρωστία διαδρομής

IV. Αποτελεσματική χρήση της στήριξης υποδομής

V. Load balancing (εξισορρόπηση φορτίου)

VI. Προσαρμογή διαδρομής

Ο σχεδιασμός του μετρητή δρομολόγησης παίζει σημαντικό ρόλο για να αποκτήσει μια καλή επίδοση. Ο καλύτερος μετρητής δρομολόγησης μπορεί να διαφέρει στην επίδοση του. Για παράδειγμα, στο WMN το σχέδιο του μετρητή δρομολόγησης πρέπει να λάβει υπόψη την ποιότητα σήματος του επιπέδου ζεύξης για καλύτερη ποιότητα επίδοση από την αρχή ως το τέλος(end-to-end). Τα πρωτόκολλα δρομολόγησης των WMNs, παρέχουν μια καλή επίδοση end-to-end, πρέπει επίσης να καταναλώνουν το ελάχιστο bandwidth για τη δημιουργία διαδρομών. Επιπλέον, η χρήση ασύρματου μέσου απαιτεί γρήγορη ικανότητα ανασχηματισμού διαδρομής για να διατηρήσει την ευρωστία της διαδρομής. Άλλη μια σημαντική όψη είναι η ικανότητα load-balancing που χρειάζεται να ενσωματωθεί με το πρωτόκολλο δρομολόγησης. Τελικά, ένα πρωτόκολλο δρομολόγησης για WMN πρέπει να προσαρμόζεται στην δυναμική δικτύου. Το πρωτόκολλο δρομολόγησης μπορεί να ταξινομηθεί είτε ως flat routing protocol ή Hierarchical routing protocol βασισμένο στο είδος δικτύου όπου εφαρμόζεται το πρωτόκολλο δρομολόγησης.

1.4.2.4 Επίπεδο μεταφοράς

Στο επίπεδο μεταφοράς, είναι η επίδοση των πρωτοκόλλων μεταφοράς σ'ένα WMN. Καθώς ένα WMN έχει μεγάλο **χρόνο μετάβασης μετ' επιστροφής(RTT)** παραλλαγές και αυτές οι παραλλαγές RTT εξαρτώνται στον αριθμό των hops της διαδρομής, η end-to-end TCP throughput υποβαθμίζεται ταχέως με το throughput.

Η απώλεια πακέτου, σύγκρουση, ασυμμετρία δικτύου, και η αποτυχία ζεύξης μπορούν επίσης να συμβάλλουν στην υποβάθμιση επίδοσης του πρωτοκόλλου επιπέδου μεταφοράς. Το δημοφιλές επίπεδο μεταφοράς για το Internet, TCP, έχει φτωχή απόδοση στην αρχική του μορφή σ'ένα WMN. Το επίπεδο μεταφοράς χρειάζεται να ξαναγραφτεί για να το κάνουν πιο αποτελεσματικό σ'ένα WMN.

Μερικά από τα θέματα σχεδιασμού για ένα πρωτόκολλο επιπέδου μεταφοράς WMN είναι:

- i. Αξιοπιστία end-to-end
- ii. Throughput
- iii. Ικανότητα διαχείρισης δικτυακής ασυμμετρίας
- iv. Ικανότητα διαχείρισης δικτυακού δυναμισμού

1.4.2.5 Επίπεδο εφαρμογής

Η πιο δημοφιλής εφαρμογή των WMNs είναι η υπηρεσία πρόσβασης Internet. Ένα WMN χρειάζεται να παρέχει υπηρεσίες Internet για περιοχές κατοικίας ή επιχειρησιακές. Σε μια τέτοια περίπτωση, αν και οι υπηρεσίες δεδομένων κάνουν την κύρια υπηρεσία με ένα WMN, οι φωνητικές υπηρεσίες όπως VoIP είναι επίσης σημαντικές. Οπότε είναι απαραίτητο να παρέχουμε υποστήριξη για κινήσεις time-sensitive και best-effort. Μαζί με τα βασικά δεδομένα και υποστήριξη φωνητικής κίνησης, το δίκτυο παρέχει μηχανισμούς ανακάλυψης υπηρεσιών. Καθώς οι περισσότερες δικτυακές υπηρεσίες είναι σε πλήρως κατανομημένη μορφή, οι μηχανισμοί στατικής ανακάλυψης υπηρεσιών μπορεί να μην είναι αποτελεσματικοί σ' ένα WMN. Άλλη μια σημαντική απαίτηση για το σχέδιο πρωτοκόλλου επιπέδου εφαρμογής είναι να χειρίζεται την ετερογένεια των δικτύων καθώς τα δεδομένα μπορεί να περνούν μέσα από μια ποικιλία δικτύων πριν σταλούν στην τελική εφαρμογή.

1.4.2.6 Θέματα σχεδιασμού στάθμης-συστήματος

Μαζί με τα θέματα σχεδιασμού πρωτοκόλλου, ένα WMN χρειάζεται λύσεις System-level. Μερικά παραδείγματα για θέματα System-level είναι:

- i. Σχεδιασμός συστήματος cross-layer
- ii. Σχεδιασμός για ασφάλεια και εμπιστοσύνη
- iii. Συστήματα διαχείρισης δικτύου
- iv. Θέματα επιβίωσης δικτύου

Μερικές από τις κύριες προκλήσεις που αντιμετωπίζει ένα WMN μπορούν να αντιμετωπιστούν με την χρήση ενός MR-WMN.

1.5 Θέματα σχεδιασμού στα multi-radio ασύρματα δίκτυα mesh

Τα κύρια πλεονεκτήματα στην χρήση ενός MR-WMN είναι η βελτιωμένη χωρητικότητα, κλιμακοθετησιμότητα (scalability), αξιοπιστία, ευρωστία, και αρχιτεκτονική ευλυγισία. Παρόλα τα πλεονεκτήματα της χρήσης ενός multi-radio συστήματος για τα WMNs, υπάρχουν πολλές προκλήσεις για να σχεδιάσεις ένα αποτελεσματικό MR-WMN σύστημα. Τα κύρια θέματα μπορούν να ταξινομηθούν σε θέματα αρχιτεκτονικής σχεδίασης, θέματα σχεδιασμού MAC, θέματα σχεδιασμού πρωτοκόλλου σχεδίασης.

1.5.1 Αρχιτεκτονικά θέματα σχεδιασμού

Η δικτυακή αρχιτεκτονική παίζει σημαντικό ρόλο για να αποκτήσουμε επίδοση αντικειμένων ενός MR-WMN όταν ένα δίκτυο αναπτύσσεται. Γενικά, η δικτυακή αρχιτεκτονική ενός MR-WMN σχεδιάζεται με βάση τον τύπο εφαρμογής ή το σενάριο ανάπτυξης.

Οι βασικές αρχιτεκτονικές που πρέπει να λαμβάνονται υπόψη είναι:

- a) Topology-based
- b) Technology-based
- c) Node-based

Βασισμένο στην τοπολογία ένα MR-WMN μπορεί να σχεδιαστεί είτε ως flat-topology-based είτε ως hierarchical-topology-based. Οι κατηγορίες σχεδιασμού κάτω από Technology-based είναι ομογενής ή ετερογενής. Η πιο δημοφιλής μορφή του συστήματος MR-WMN είναι η ομογενούς τύπου που χρησιμοποιεί μόνο ένα είδος ραδιοτεχνολογίας όπως το δημοφιλή WLAN τεχνολογία IEEE 802,11, είναι εφικτό όμως να αναπτύξουμε ένα MR-WMN με ετερογενή τεχνολογίες που χρησιμοποιούν μια ποικιλία από τεχνολογίες επικοινωνιών.

Το κριτήριο σχεδιασμού Node-based μπορεί να ταξινομηθεί είτε σε host-nodes, infrastructure-based, ή Hybrid MR-WMNs. Στην περίπτωση του host-based MR-WMNs, το δίκτυο σχηματίζεται από τους hosts nodes και είναι το ίδιο με ένα ad hoc ασύρματο δίκτυο με περιορισμένη ή με καμία κινητικότητα. Στα infrastructure-based MR-WMNs, το WMN σχηματίζεται από κόμβους σε σταθερές υποδομές ή κτίρια. Ένα παράδειγμα αυτού του αρχιτεκτονικού είδους είναι το δίκτυο πάνω σε σκέπες που σχηματίζονται τοποθετώντας ασύρματους κόμβους αναμετάδοσης mesh (πλέγματος) πάνω στην σκεπή κάθε σπιτιού για να χτίσουμε ένα δίκτυο επικοινωνίας κατοικίας. Ένα υβρίδιο MR-WMN έχει και infrastructure-based backbone και wireless mesh hosts. Οι hosts επικοινωνούν πάνω από το wireless mesh backbone. Η backbone τοπολογία μπορεί να οργανωθεί είτε ως μια flat τοπολογία είτε ως Hierarchical τοπολογία. Σε κάποια περιβάλλοντα εφαρμογής, οι hosts είναι κινητοί και αυτοί επίσης αναμεταδίδουν κίνηση εκ μέρους των άλλων hosts στο δίκτυο.

Παράδειγμα τέτοιου υβριδίου MR-WMN είναι τα WMNs οχημάτων που επικοινωνούν με μια ασύρματη υποδομή πλέγματος. Οπότε, το σχέδιο ενός συστήματος MR-WMN πρέπει να λάβει υπόψη του το είδος της εφαρμογής ή το περιβάλλον ανάπτυξης για επιλογή κατάλληλης αρχιτεκτονικής λύσης.

1.5.2 Θέματα σχεδιασμού ελέγχου πρόσβασης μέσου

Το επίπεδο MAC για τα MR-WMNs αντιμετωπίζει αρκετές προκλήσεις. Οι κύριες προκλήσεις ανάμεσα τους είναι:

- i. Διακαναλική παρεμβολή
- ii. Διαραδιοεπικοινωνιακή παρεμβολή
- iii. Καταχώριση καναλιού
- iv. Σχέδιο πρωτοκόλλου MAC

Η Διακαναλική παρεμβολή αναφέρεται στην παρεμβολή που βιώνεται σένα κανάλι εξαιτίας της δραστηριότητας στα γειτονικά κανάλια. Παράδειγμα το IEEE 802,11b η βόρεια Αμερική έχει 11 κανάλια,η Ευρώπη έχει 13 και η Ιαπωνία έχει 14,αλλά η βόρεια Αμερική μπορεί να χρησιμοποιήσει ταυτόχρονα σε όλες τις ηπείρους μόνο 3 από τα 11(1-6-11)κανάλια. Τα multiradios πρέπει να λάβουν υπόψιν την Διακαναλική παρεμβολή για την χρήση ενός καινούργιου καναλιού,μια δεύτερη παρεμβολή που εμποδίζει το υπάρχων κανάλι θα οδηγήσει σε υποβάθμιση επίδοσης.

Οπότε σε τέτοιες περιπτώσεις τα multiradio πρέπει να χρησιμοποιήσουν κανάλια που δεν συμπίπτουν. Η Διαραδιοεπικοινωνιακή παρεμβολή προκύπτει εξαιτίας του σχεδίου και εφαρμογής του σε ραδιοδιεπαφές. Αυτή η παρεμβολή βιώνεται σε μια συγκεκριμένη ραδιοεπικοινωνία εξαιτίας της δραστηριότητας του καναλιού σε άλλη διεπαφή μέσα στον ίδιο WMN κόμβο. Τέτοιες παρεμβολές συμβαίνουν όταν και οι δυο διεπαφές χρησιμοποιούν κανάλια που συμπίπτουν. Για παράδειγμα όταν οι διεπαφές A και B στον κόμβο WMN χρησιμοποιούν τα κανάλια 1 και 11 αντίστοιχα,μπορεί να παρουσιαστούν παρεμβολές μέσα στην Διαραδιοεπικοινωνία.

Αυτή η παρεμβολή είναι κυρίως εξαιτίας του σχεδιασμού των hardware components(συνιστώσες υλικού) και την ίδια την διεπαφή όπου συνήθως ένας αριθμός από χαμηλού-κόστους φίλτρα και συσχετισμένα RF εξαρτήματα χρησιμοποιούνται. Ο φυσικός χωρισμός των διεπαφών μπορεί να βοηθήσει να αποφύγουμε αυτό το θέμα έως κάποιο βαθμό,σε ορισμένες περιπτώσεις ο χωρισμός ίσως να είναι δύσκολος ιδιαίτερα σε φορητούς κόμβους. Η χρήση χαμηλού-κόστους καρτών διεπαφής οδηγεί σε παρεμβολές ακόμα και όταν χωρίζονται με μικρή απόσταση. Η Καταχώριση καναλιού αυτή είναι μια μεγάλου-δικτύου επεξεργασία όπου η καταχώριση των καναλιών που δεν εμποδίζουν θα οδηγήσει σε σημαντικό throughput και απόδοσης των μέσων πρόσβασης. Η καταχώριση καναλιού πρέπει να λάβει υπόψιν της τον αριθμό διαθέσιμων καναλιών και τον αριθμό διαθέσιμων διεπαφών. Οπότε,τεχνικές όπως ο χρωματισμός γραφήματος χρησιμοποιούνται για να παράγουν στρατηγικές καναλιού καταχώρισης.

Τελικά,το πιο σημαντικό είναι ο σχεδιασμός των MAC πρωτοκόλλων. Η διαθεσιμότητα των πολλαπλών διεπαφών και πολλαπλών καναλιών οδηγεί σε καινούργια σχέδια για πρωτόκολλο πρόσβασης μέσου(medium access protocol) που μπορούν να επωφεληθούν στην παρουσία των multipleradios.

Παραδείγματα τέτοιων πρωτοκόλλων είναι το MCSMA,ICSMA,2P-TDMA. Αυτα τα πρωτόκολλα χρησιμοποιούν πολλαπλά κανάλια ταυτόχρονα και επίσης προσπαθούν να λύσουν το θέμα πρόσβασης μέσου στα MR-WMNs.

1.5.3 Θέματα σχεδιασμού πρωτοκόλλου δρομολόγησης

Ο σχεδιασμός του πρωτοκόλλου δρομολόγησης εξαρτάται από το αρχιτεκτονικό σχέδιο του WMN και σε μερικές περιπτώσεις εξαρτάται και από την εφαρμογή του δικτύου και από το σενάριο ανάπτυξης.

Το σχέδιο πρωτοκόλλου δρομολόγησης μπορεί να ταξινομηθεί σε:

- A. Τοπολογία δρομολόγησης
- B. Η χρήση της δρομολόγησης backbone
- C. Προσέγγιση για πληροφορίες συντήρησης της δρομολόγησης

Βασισμένο στην τοπολογία δρομολόγησης, τα πρωτόκολλα δρομολόγησης μπορούν να σχεδιαστούν είτε ως ένα flat πρωτόκολλο δρομολόγησης είτε ως hierarchical πρωτόκολλο δρομολόγησης. Στην ιεραρχική δρομολόγηση, μια ιεραρχική δρομολόγηση χτίζεται μέσα σε κόμβους με τέτοιο τρόπο ώστε η ευθύνη της ανίχνευσης αναθέτετε σε κόμβους που βρίσκονται σε υψηλότερο επίπεδο στην ιεραρχία όταν οι κόμβοι στο χαμηλότερου επιπέδου δεν καταφέρνουν να αποκτήσουν μια διαδρομή, πχ hierarchical state routing (HCR).

Ένα flat σύστημα δρομολόγησης δεν έχει καμία ιεραρχία και ο κάθε κόμβος έχει ευθύνη να βρει μια διαδρομή στον προορισμό και να συμμετέχει στην διαδικασία της ανίχνευσης άλλων κόμβων. Η επιλεγμένη διαδρομή μπορεί να περιλαμβάνει οποιονδήποτε αυθαίρετο κόμβο στο δίκτυο χωρίς να ακολουθεί καμία συγκεκριμένη ιεραρχία των κόμβων.

Η δεύτερης κατηγορία είναι η δρομολόγηση που βασίζεται σε δρομολόγηση backbone και ταξινομείται σε:

1. Tree-based backbone δρομολόγηση
2. Mesh-based backboneless δρομολόγηση
3. Hybrid τοπολογία δρομολόγησης

Αντίθετα με ένα ασύρματο ad hoc δίκτυο, ένα WMN είναι σχετικά στατικό ή έχει περιορισμένη δικτυακή κινητικότητα. Οπότε για να αυξηθεί η αποτελεσματικότητα της δρομολόγησης, η δρομολόγηση backbone μπορεί να χτιστεί. Ένα παράδειγμα για αυτήν την προσέγγιση της δρομολόγησης είναι η WMN δρομολόγηση που εκτελείται από τον IP μηχανισμό δρομολόγησης πάνω σε ένα spanning tree protocol (STP)-based tree backbone. Στην περίπτωση του STP, το επίπεδο γραμμής δεδομένων θα σχηματίσει μια τοπολογία δέντρου ανάμεσα στους WMN κόμβους παρόμοιο με το wireless distributed system (WDS) και στο επίπεδο δικτύου, η δρομολόγηση διεξάγεται με παραδοσιακό IP-based μέθοδο δρομολόγησης. Παρόλο που αυτή είναι μια από τις πιο απλές προσεγγίσεις για το WMN, έχει αρκετά θέματα όπως κακή αξιοπιστία και έλλειψη δικτυακής κλιμακοθετησιμότητας.

Από την άλλη πλευρά τα πρωτόκολλα δρομολόγησης που έχουν σχεδιαστεί και εφαρμοστεί στο επίπεδο δικτύου μπορεί να ακολουθήσουν μια backboneless προσέγγιση δρομολόγησης πλέγματος(mesh).

Μια τρίτη προσέγγιση είναι να χρησιμοποιείται σε επίπεδο δικτύου μια backbone τοπολογία,ένα υποσύνολο από κόμβους που σχηματίζουν μια σαν πλέγματος backbone μέσα στο WMN,βελτιστοποιημένο για ορισμένες παραμέτρους όπως throughput,ποιότητα καναλιού,ή δικτυακή κλιμακοθετησιμότητα που μπορεί να χρησιμοποιηθεί για να βοηθήσει ένα πρωτόκολλο δρομολόγησης πλέγματος backboneless. Μια τέτοια προσέγγιση δρομολόγησης που χρησιμοποιεί μια δυναμική τοπολογία backbone σε ένα συγκεκριμένο κομμάτι του δικτύου ονομάζεται πρωτόκολλο δρομολόγησης υβριδικής τοπολογίας. Τελικά τα πρωτόκολλα δρομολόγησης μπορούν να σχεδιαστούν με βάση την προσέγγιση δρομολόγησης συντήρησης του.

Παραδείγματα από τέτοια σχέδια δρομολόγησης είναι:

1. Ενεργητικά (προληπτικά) ή πρωτόκολλα δρομολόγησης από πίνακα-δρομολόγησης(table-driven)
2. Αντιδραστικά ή πρωτόκολλα δρομολόγησης από απαίτηση (on-demand)
3. Υβριδικά πρωτόκολλα δρομολόγησης

Αναλυτικότερα:

1. Τα ενεργητικά ή πρωτόκολλα δρομολόγησης από πίνακα-δρομολόγησης(table-driven),ο κάθε κόμβος ανταλλάσσει τις πληροφορίες δρομολόγησης του περιοδικά και συντηρεί ένα πίνακα δρομολόγησης,το οποίο περιέχει πληροφορίες δρομολόγησης που χρησιμοποιούν για να φτάσουν σε κάθε κόμβο στο δίκτυο. Παραδείγματα πρωτοκόλλων που χρησιμοποιούν αυτήν την δρομολόγηση σχεδιασμού είναι: DSDV,WRP,STAR.
2. Τα αντιδραστικά ή πρωτόκολλα δρομολόγησης από απαίτηση(on-demand),ένας κόμβος απαιτεί πληροφορίες δρομολόγησης και διατηρεί τις πληροφορίες διαδρομής μόνο όταν χρειάζεται να επικοινωνήσει με άλλον κόμβο. Παραδείγματα πρωτοκόλλων που χρησιμοποιούν αυτήν την δρομολόγηση σχεδιασμού είναι:AODV,DSR,MRLQSR.
3. Τα υβριδικά πρωτόκολλα δρομολόγησης επωφελούνται από τα table-driven και από τα on-demand. Παραδείγματα πρωτοκόλλων που χρησιμοποιούν αυτήν την δρομολόγηση σχεδιασμού είναι:ZPR,το οποίο χρησιμοποιεί μια προσέγγιση δρομολόγησης table-driven μέσα σε μια ζώνη προσέγγισης on-demand πέραν της ζώνης. Δηλαδή ο κάθε κόμβος χρησιμοποιεί μια ενεργητική προσέγγιση μέσα σε μια k-hop ζώνη δρομολόγησης και χρησιμοποιεί μια αντιδραστική προσέγγιση δρομολόγησης πέρα από τη ζώνη δρομολόγησης.

1.5.4 Θέματα σχεδιασμού μετρικής δρομολόγησης

Εκτός από τον σχεδιασμό ενός πρωτοκόλλου δρομολόγησης, ένα άλλο σημαντικό θέμα είναι ο σχεδιασμός μιας μετρικής δρομολόγησης. Μια μετρική δρομολόγησης είναι η παράμετρος δρομολόγησης, βάρος, ή η αξία που σχετίζεται με μια ζεύξη ή διαδρομή, βασισμένο σε μια απόφαση δρομολόγησης παρθεί. Το hop count είναι ο πιο απλός μετρητής δρομολόγησης και είναι ένας πρόσθετος μετρητής δρομολόγησης. Εξαιτίας των ειδικών χαρακτηριστικών των WMNs, το hop count ως ένας μετρητής δρομολόγησης έχει κακή επίδοση. Οπότε ο σχεδιασμός του μετρητή δρομολόγησης είναι πολύ σημαντικός για τα MR-WMNs. Ο μετρητής δρομολόγησης παίζει σημαντικό ρόλο στην επίδοση ενός πρωτοκόλλου δρομολόγησης και ο σχεδιασμός των μετρητών δρομολόγησης πρέπει να λάβει υπόψιν του αρκετούς παράγοντες όπως:

- a. Την αρχιτεκτονική του δικτύου
- b. Το περιβάλλον του δικτύου
- c. Την ένταση δυναμισμού του δικτύου
- d. Τα βασικά χαρακτηριστικά του πρωτοκόλλου δρομολόγησης

Πρώτον, η αρχιτεκτονική ιδιότητα του δικτύου χρειάζεται να την λάβουμε υπόψιν για τον σχεδιασμό του μετρητή δρομολόγησης. Για παράδειγμα το δίκτυο μπορεί να σχεδιαστεί είτε σε μια flat τυπολογική αρχιτεκτονική είτε σε μια πολυβαθμιδωτή ιεραρχική αρχιτεκτονική. Επιπλέον, το αρχιτεκτονικό σχέδιο μπορεί να συμπεριλαμβάνει ένα δίκτυο χωρίς υποδομή, εν μέρη ένα δίκτυο που υποστηρίζεται από υποδομή, ή ένα δίκτυο που στηρίζεται από υποδομή. Ο μετρητής δρομολόγησης, για να σχεδιαστεί πρέπει να λάβει υπόψιν το αρχιτεκτονικό σχέδιο του δικτύου.

Έναν δεύτερο παράγοντα που πρέπει να λάβουμε υπόψιν είναι το περιβάλλον του δικτύου. Για παράδειγμα, εξαιτίας της παρουσίας της διαμάχης σχετικά με την εξάρτηση της τοποθεσίας (location-dependent), υψηλές κυμαινόμενες και απρόβλεπτες συνθήκες καναλιών, και το υψηλό ποσοστό λάθους ανά bit (BER), τα χαρακτηριστικά ενός WMN περιβάλλον είναι ριζικά διαφορετικό από εκείνο ενός ενσύρματου δικτύου. Οπότε, το σχέδιο του πρωτοκόλλου δρομολόγησης και μετρητή δρομολόγησης πρέπει να λάβει υπόψιν το συγκεκριμένο δικτυακό περιβάλλον. Η ένταση δυναμισμού του δικτύου εξαιτίας της κινητικότητας που βιώνεται από το δίκτυο, για ένα WMN σχεδιασμένο για στατικούς ή ελάχιστα κινητικούς κόμβους, τα ενεργητικά πρωτόκολλα μπορεί να είναι κατάλληλα όπου η προσέγγιση δρομολόγησης που έχει ζήτηση είναι κατάλληλη για ένα WMN που χειρίζεται κόμβους με υψηλή κινητικότητα.

Για να σχεδιάσεις έναν αποτελεσματικό μετρητή δρομολόγησης, τα βασικά χαρακτηριστικά μιας προσέγγισης δρομολόγησης είναι σημαντικά. Για παράδειγμα, καθώς ένα μη-ισοτονικό πρωτόκολλο δρομολόγησης δουλεύει καλά με μια πηγή on-demand βασισμένη σε ένα πρωτόκολλο δρομολόγησης, μπορεί να αποτύχει ή να έχει κακή επίδοση εξαιτίας του σχηματισμού των loops (βρόχος) δρομολόγησης όταν χρησιμοποιείται με πρωτόκολλο δρομολόγησης table-driven hop-by-hop.

Τα αντικείμενα σχεδιασμού για ένα πρωτόκολλο δρομολόγησης και μετρική είναι:

- 1)Αποτελεσματικότητα πόρων
- 2)Throughput
- 3)Ελευθερία από βρόχους δρομολόγησης
- 4)Σταθερότητα διαδρομής
- 5)Ικανότητα οργάνωσης για γρήγορη διαδρομή
- 6)Αποτελεσματική συντήρηση της διαδρομής

Οι προκλήσεις για να σχεδιάσεις πρωτόκολλο δρομολόγησης για MR-WMNs είναι:

- a)Διαραδιοεπικοινωνιακή παρεμβολή
- b)(interflow) παρεμβολή μεταξύ γειτονικών δρομολογητών που ανταγωνίζονται για τον ίδιο απασχολημένο κανάλι
- c)(intraflow) παρεμβολές μεταξύ των ενδιάμεσων δρομολογητών που μοιράζονται την ίδια διαδρομή ροής
- d)Πρόβλημα κρυμμένου τερματικού κόμβου
- e)Πρόβλημα εκτεθειμένου τερματικού κόμβου
- f)Διαμάχη location-depented
- g)Υψηλά δυναμισμού χαρακτηριστικά καναλιού
με τους κόμβους του δικτύου αυτόματα ιδρύεται ένα δίκτυο ad hoc και διατηρεί την «πλέγματος» συνδεσιμότητα. Κάθε ατομική δρομολόγηση μετρική εξετάζει κάποια χαρακτηριστικά και είναι δύσκολο να ικανοποιούν όλες τις απαιτήσεις των WMNS με τη χρήση μιας μόνο μετρικής. Αρκετές μετρήσεις απόδοσης που έχουν ήδη σχεδιαστεί για WMNs όπως Expected Transmission Count (ETX), Expected Transmission Time (ETT), Interference Aware Routing Metric (iAWARE).

1.5.5 Θέματα σχεδιασμού τοπολογίας

Η επίδοση του δικτύου σε ένα WMN επηρεάζεται από την τοπολογία του δικτύου, ελέγχοντας την τοπολογία του δικτύου η δικτυακή επίδοση μπορεί να βελτιωθεί. Ο έλεγχος της τοπολογίας ορίζεται από την ικανότητα του δικτύου να χειραγωγήσουν τις παραμέτρους του όπως η θέση των κόμβων, η κινητικότητα των κόμβων, η ισχύς εκπομπής, οι ιδιότητες της κεραίας, και η κατάσταση των διεπαφών.

Η τοπολογία μπορεί να ελέγχεται είτε ως μια μοναδική δραστηριότητα κατά την διάρκεια της δικτυακής φάσης αρχειοθέτησης ή ως μια περιοδική δραστηριότητα throughput σε όλη την διάρκεια της δικτυακής ζωής. Η αποτελεσματική χρήση του δικτύου τοπολογικού έλεγχου μπορεί να βοηθήσει να βελτιωθεί η χωρητικότητα.

Στην πράξη, η θέση κόμβου και η κινητικότητα δεν είναι υπό απευθείας έλεγχο του δικτυακού συστήματος, αφήνοντας τους υπόλοιπους παράγοντες όπως η ισχύς εκπομπής, οι ιδιότητες της κεραίας, και η κατάσταση των καρτών δικτυακής διεπαφής. Τα αντικείμενα των τοπολογικού έλεγχου είναι η συνδετικότητα, χωρητικότητα, αξιοπιστία και ανεκτικότητα σφάλματος, και κάλυψη δικτύου.

ΚΕΦΑΛΑΙΟ 2

Η ΔΡΟΜΟΛΟΓΗΣΗ ΣΤΑ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ MESH

2.1 Εισαγωγή

Τα ασύρματα δίκτυα πλέγματος (WMNs) είναι μια καινούργια τάση στην ασύρματη επικοινωνία και υπόσχεται μεγαλύτερη ευλυγισία,αξιοπιστία,και επίδοση στα ασύρματα δίκτυα τοπικών περιοχών(WLANs). Υπάρχουν πολλές εταιρείες που πουλούν λύσεις για τα WMN. Επίσης υπάρχει υποστήριξη από ομάδες τυποποίησης όπως IEEE 802.11,802.15,802.16. Τα WMNs και τα ad hoc δίκτυα χρησιμοποιούν την ίδια έννοια επικοινωνίας ανάμεσα σε κόμβους πάνω σε ασύρματο multiple hops σένα γράφημα δικτύου mesh. Όμως τονίζουν διαφορετικές πτυχές. Τα κινητά ad hoc δίκτυα(MANETs) έχουν ένα ακαδημαϊκό ιστορικό και επικεντρώνονται στις συσκευές που χρησιμοποιεί ο χρήστης,κινητικότητα και ad hoc ικανότητες.

Τα WMNs έχουν ένα επιχειρηματικό ιστορικό και επικεντρώνονται κυρίως στις στατικές συσκευές(αναλόγως της υποδομής),αξιοπιστία,χωρητικότητα δικτύου και πρακτική ανάπτυξη.

Η κύρια λειτουργία των ad hoc και των mesh είναι η ικανότητα δρομολόγησης. Τα πρωτόκολλα δρομολόγησης παρέχουν τις απαραίτητες διαδρομές μέσα σένα WMN,έτσι ώστε οι κόμβοι να επικοινωνούν σε καλή ή βέλτιστη διαδρομή πάνω σε πολλαπλά ασύρματα hops. Τα πρωτόκολλα δρομολόγησης πρέπει να λάβουν υπόψιν το δύσκολο περιβάλλον της ραδιοεπικοινωνίας με τις συνθήκες της να αλλάζουν συχνά και πρέπει να υποστηρίζουν ένα αξιόπιστο και αποτελεσματικό περιβάλλον στο δίκτυο πλέγματος.

Καθώς τα WMNs μοιράζονται κοινά χαρακτηριστικά με τα ad hoc,τα πρωτόκολλα δρομολόγησης που αναπτύχθηκαν για MANETs μπορούν να εφαρμοστούν στα WMNs. Για παράδειγμά, Microsoft Mesh Networks φτιάχνονται με βάση την Dynamic Source Routing(DSR),και πολλές άλλες εταιρείες χρησιμοποιούν ad hoc On-demand Distance Vector(AODV). Μερικές φορές οι έννοιες των υπαρχόντων πρωτοκόλλων δρομολόγησης επεκτείνονται για να ανταποκριθούν στις ειδικές απαιτήσεις των WMNs,για παράδειγμα με τους ραδιο-ενημερους μετρητές δρομολόγησης όπως την IEEE 802.11s WLAN mesh networking standardization.

Παρ όλη την διαθεσιμότητα αρκετών πρωτοκόλλων δρομολόγησης για τα ad hoc,το σχέδιο δρομολόγησης για τα WMNs είναι ακόμα ένα ενεργό κομμάτι έρευνας για αρκετούς λόγους.

- ◆ Στα περισσότερα WMNs,πολλοί κόμβοι είναι είτε σταθεροί ή έχουν ελάχιστη κινητικότητα και δεν βασίζονται σε μπαταρίες. Οπότε, η εστίαση των αλγορίθμων δρομολόγησης είναι να βελτιωθεί το δικτυακό throughput ή η επίδοση των ατομικών μεταφορών,αντί να αντιμετωπίζουν την κινητικότητα ή να ελαχιστοποιήσουν της χρήσης ενέργειας.

- ◆ Η απόσταση μεταξύ των κόμβων μπορεί να μειωθεί σένα WMN, το οποίο αυξάνει την ποιότητα της σύνδεσης και το ρυθμό μετάδοσης. Ωστόσο,μικρές αποστάσεις αυξάνουν επίσης την παρεμβολή μεταξύ των hops,η οποία μειώνει το διαθέσιμο bandwidth για κάθε σύνδεση. Ως εκ τούτου, καινούργιοι μετρητές δρομολόγησης πρέπει

να ανακαλυφθούν και να χρησιμοποιηθούν για να βελτιώσουν την απόδοση των πρωτοκόλλων δρομολόγησης σε WMN multihop Multiradio.

◆ Σένα multiradio/multichannel WMN, το πρωτόκολλο δρομολόγησης όχι μόνο πρέπει να επιλέξει μια πορεία ανάμεσα σε διαφορετικούς κόμβους, αλλά χρειάζεται επίσης να επιλέξει το πιο κατάλληλο κανάλι ή radio επάνω στην πορεία για τον κάθε κόμβο πλέγματος. Οπότε, μετρητές δρομολόγησης πρέπει να ανακαλυφθούν και να χρησιμοποιηθούν για να επωφεληθούν των multiple radios σε ένα ασύρματο δίκτυο πλέγματος.

◆ Σένα WMN, το σχέδιο επιπέδου γραμμής δεδομένων (cross-layer) είναι απαραίτητο επειδή η αλλαγή μιας διαδρομής δρομολόγησης περιλαμβάνει την αλλαγή καναλιού ή radio σένα multiradio multichannel κόμβο πλέγματος.

2.2 Ιδιότητες των ασύρματων δικτύων MESH

Τα ασύρματα δίκτυα πλέγματος στέλνουν πακέτα δεδομένων στοιχείων σε πολλαπλά ασύρματα hops. Ο κάθε κόμβος πλέγματος δρα ως αναμεταδότης σημείο/δρομολόγησης για τους άλλους κόμβους πλέγματος. Τα WMNs μπορούν συχνά να βρεθούν σε εμπορικά σενάρια χρήσης, δηλαδή, κάποιος να πουλάει το δίκτυο mesh. Τα πιο δεσπόζουσα είναι τα δίκτυα που έχουν πρόσβαση το κοινό και τα δημοτικά ασύρματα δίκτυα, όπου τα σημεία πρόσβασης είναι οι κόμβοι των ασυρμάτων δικτύων πλέγματος.

Η αξιοπιστία και η δικτυακή επίδοση είναι σημαντικοί στόχοι για τα WMNs, ιδιαίτερα στο δύσκολο ασύρματο περιβάλλον. Η κινητικότητα των κόμβων συνήθως δεν λαμβάνεται υπόψη ($u=0$). Οι στατικοί κόμβοι μπορούν να τοποθετηθούν πάνω σε στύλους φώτος, όπου υπάρχει αρκετή παροχή ρεύματος. Με αυτές τις υποθέσεις και το σενάριο της χρήσης στο μυαλό, τα πρωτόκολλα δρομολόγησης ad hoc μπορούν να μεγιστοποιηθούν σχετικά με την αξιοπιστία και την απόδοση του δικτύου.

Μπορούν να επεκταθούν για να χρησιμοποιήσουν ειδικούς μετρητές δρομολόγησης και μπορούν ακόμα να εντοπιστούν στο επίπεδο 2 για να έχουν καλύτερη πρόσβαση στις πληροφορίες του MAC και στα φυσικά επίπεδα. Οι κόμβοι πλέγματος μπορούν να έχουν πολλαπλές ασύρματες συνδέσεις για να αυξηθεί η χωρητικότητα του δικτύου mesh.

Οι πολλαπλές συνδέσεις μειώνουν την υποβαθμίσει του throughput εξαιτίας της διαδοχικής λήψης και την μεταφορά πακέτων στους κόμβους mesh με μια μοναδική ασύρματη σύνδεση. Μπορεί επίσης να είναι εφικτό να χρησιμοποιηθούν multiple channels. Οι ad hoc ικανότητες των WMNs είναι περιορισμένες άλλα η σταθερή απλή εγκατάσταση και η ευκαμψία είναι ένα πλεονέκτημα.

Έχει γίνει όλο και πιο φυσικό ότι οι (ισχυρές) συσκευές-πελάτες μπορούν να είναι κόμβοι mesh. Αυτό επεκτείνει τα WMNs προς και μέσα στην περιοχή των κλασικών κινητών ad hoc δικτύων. Αυτό δεν είναι προβληματικό, επειδή οι γενικές έννοιες είναι ίδιες ανάμεσα στα MANETs και WMNs. Αυτοί χρησιμοποιούν μόνο διαφορετικές "αξίες" για τους δικτυακούς παραμέτρους:

Κόμβους με κινητικότητα από "στατικούς" έως "κινούμενους με u " χρησιμοποιούν ασύρματη επικοινωνία μέσα από μια ή περισσότερες συνδέσεις σε multiple wireless

hops, όπου οι πορείες καθορίζονται από self-organizing πρωτόκολλα δρομολόγησης που δουλεύουν με διαφορετικούς μετρητές δρομολόγησης.

Τρεις τύποι ασύρματων δικτύων mesh μπορούν να αναγνωριστούν:

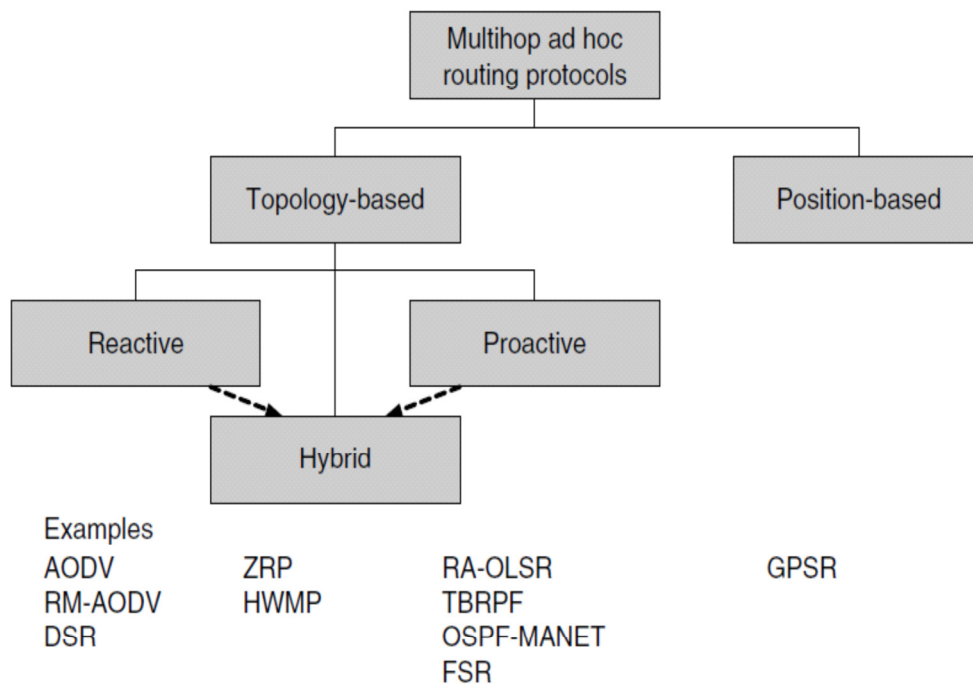
Δίκτυα πλέγματος υποδομής αποτελούνται από ειδικές συσκευές της δικτυακής υποδομής, όπως σημεία πρόσβασης ή αναμετάδοσης. Οι συσκευές-πελάτης δεν συμμετέχουν στην mesh δρομολόγηση. Αντί γιαυτό συνδέονται με τα σημεία πρόσβασης στα δίκτυα mesh με την παραδοσιακή ασύρματη τεχνολογία πρόσβασης.

Τα δίκτυα πλέγματος πελάτης αποτελούνται από συσκευές-πελάτης όπως τα laptops. Οι συσκευές-πελάτης συμμετέχουν στην mesh δρομολόγηση. Επίσης μπορούν να εκτελέσουν λειτουργίες συσκευών υποδομής.

Τα υβρίδια δίκτυα πλέγματος αποτελούνται από συσκευές υποδομής και από συσκευές-πελάτης.

2.3 Γενικές έννοιες πρωτοκόλλων δρομολόγησης

2.3.1 Ταξινόμηση των πρωτοκόλλων δρομολόγησης



Το κύριο καθήκον των πρωτοκόλλων δρομολόγησης είναι η επιλογή διαδρομής ανάμεσα στον κόμβο πηγής και στον κόμβο προορισμού. Αυτό πρέπει να γίνει αξιόπιστα, γρήγορα και με ελάχιστα έξοδα. Πρέπει να υπάρχει μια διαδρομή υπολογίζεται εάν υπάρχει μια. Τα πρωτόκολλα δρομολόγησης μπορούν να ταξινομηθούν σε:

• Τα πρωτόκολλα δρομολόγησης βασισμένα στην τοπολογία επιλεγούν διαδρομές που βασίζονται σε τοπολογικές πληροφορίες, όπως συνδέσεις ανάμεσα στους κόμβους.

• Τα πρωτόκολλα δρομολόγησης βασισμένα στην τοποθεσία επιλέγουν διαδρομές που βασίζονται σε γεωγραφικές πληροφορίες με γεωμετρικούς αλγορίθμους.

Υπάρχουν πρωτόκολλα δρομολόγησης που συνδυάζουν αυτές τις δυο έννοιες.

1. Reactive
2. Proactive
3. Hybrid

Τα Reactive πρωτόκολλα εκτελούν μια διαδρομή μόνο όταν χρειάζεται. Αυτό μειώνει τον έλεγχο επίγεια αλλά εισάγει ένα σφάλμα για να σταλθεί το πρώτο πακέτο εξαιτίας του χρόνου που απαιτείται για την on-demand ρύθμιση δρομολόγησης.

Στα Proactive ο κάθε κόμβος γνωρίζει μια διαδρομή σε όλους τους κόμβους όλη την ώρα. Δεν υπάρχει λάθος, αλλά μόνιμη συντήρηση των αχρησιμοποίητων διαδρομών αυξάνει την επιβάρυνση του ελέγχου.

Τα Hybrid προσπαθούν να συνδυάσουν τα πλεονεκτήματα από τις δυο φιλοσοφίες: το Reactive χρησιμοποιείται για τους κοντινούς κόμβους ή διαδρομές που χρησιμοποιούνται συχνά, ενώ τα Proactive χρησιμοποιούνται για μακρινούς κόμβους ή πορείες που χρησιμοποιούνται λιγότερο.

Άλλες πιθανότητες για την ταξινόμηση των πρωτοκόλλων δρομολόγησης είναι flat vs. Hierarchical, distance vector vs. link state, source routing vs. hopby-hop routing, single-path vs. multipath, ή βασισμένες στο σενάριο χρήσης.

Κυρίως τα δίκτυα πλέγματος μπορούν να παρατάξουν οπουδήποτε πρωτόκολλο δρομολόγησης από οποιαδήποτε κατηγορία που αναφέρθηκε νωρίτερα.

Όμως δεν μπορεί να λειτουργήσει καλά το κάθε πρωτόκολλο. Η επιλογή ενός κατάλληλου πρωτοκόλλου δρομολόγησης εξαρτάται από το προσλαμβάνων σενάριο εφαρμογής και από τις απαιτήσεις της επίδοσης.

2.3.2 Η δρομολόγηση στο επίπεδο 2

Σύμφωνα με το μοντέλο επιπέδου OSI και το μοντέλο TCP/IP, η λειτουργία δρομολόγησης είναι τοποθετημένη στο επίπεδο 3, το επίπεδο δικτύου, που χρησιμοποιεί συνήθως το πρωτόκολλο Internet (IP). Πρόσφατα, υπάρχουν προσπάθειες να αναπτύξουν πρωτόκολλα δρομολόγησης για ad hoc mesh δίκτυα στο επίπεδο 2.

Παρόλο που αυτό παραβιάζει το τωρινό επίπεδο δικτύου, αναμένονται τα εξής οφέλη:

- a) Πιο γρήγορη πρόσβαση σε περισσότερες πληροφορίες κατάστασης του επιπέδου γραμμής δεδομένων και του φυσικού επιπέδου
- b) Ταχύτερη προώθηση
- c) Βελτιώσεις στην πρόσβαση στα μέσα επικοινωνίας σχετικά με την wireless multihop communication
- d) Συνεργίες ανάμεσα στους μηχανισμούς (π.χ. περιοδικές εκπομπές δεν είναι απαραίτητες και στα δυο επίπεδα, μόνο μια φορά)

Φυσικά τα οφέλη δεν είναι δωρεάν. Η δρομολόγηση στο επίπεδο 2 είναι πιο δύσκολη να εφαρμοστεί, οι επιπλέον πληροφορίες στην δομή του δικτύου όπως είναι γνωστό από τις IP διευθύνσεις δεν είναι διαθέσιμη στις MAC διευθύνσεις, και είναι πιο δύσκολο να κάνεις “Internet” εργασία ανάμεσα σε ετερογενή δίκτυα.

Όμως ιδιαίτερα τα πλεονεκτήματα της καλύτερης πρόσβασης στα χαμηλότερα επίπεδα, τα οποία θα αυξήσουν την αξιοπιστία των wireless ad hoc mesh Networks λόγω της ταχύτερης και πιο κατάλληλης αντιδράσεις στις αλλαγές μέσα στο περιβάλλον του radio και των ασυρμάτων συνδέσμων, είναι ένα ισχυρό κίνητρο για τα πρωτόκολλα δρομολόγησης του επιπέδου 2.

Οι έννοιες για την επιλογή διαδρομής είναι ίδιες είτε στο επίπεδο 2 είτε στο 3. Ο τελευταίος χρησιμοποιεί μόνο MAC διευθύνσεις. Σημαίνει επίσης ότι μερικοί μηχανισμοί, μέχρι τώρα άγνωστες στο επίπεδο 2 πρέπει να εισαχθούν:

- Χρόνος ζωής (TTL) για πρόληψη των loop
- Πηγή διεύθυνσης και διεύθυνση προορισμού ως σημεία τέρματος μιας wireless multihop διαδρομής
- Η επέκταση μηχανισμών που σχεδιάστηκαν αρχικά για ένα single wireless hop to multiple hops

2.3.3 Απαιτήσεις δρομολόγησης στα ασύρματα MESH δίκτυα

Βασισμένο στην επίδοση των υπάρχων πρωτοκόλλων δρομολόγησης για ad hoc δίκτυα και τις συγκεκριμένες απαιτήσεις των WMNs, ένα μέγιστο πρωτόκολλο δρομολόγησης για τα WMNs πρέπει να συλλαμβάνει τα παρακάτω χαρακτηριστικά:

- **Fault Tolerance**(ανεκτικότητα λάθους): Η επιβίωση είναι η ικανότητα του δικτύου να λειτουργήσει στην περίπτωση που αποτύχει ένας κόμβος ή ένας σύνδεσμος. Τα WMNs μπορούν να εξασφαλίσουν στιβαρότητα ενάντια αποτυχιών συνδέσμων εκ φύσεως. Αντίστοιχα τα πρωτόκολλα δρομολόγησης πρέπει επίσης να υποστηρίζουν την επανεπιλογή διαδρομής που υπόκειται σε αποτυχίες συνδέσμου.
- **Load Balancing**(εξισορρόπηση φόρτου): Οι ασύρματοι δρομολογητές δραστηριοποίησης-mesh είναι καλοί στην εξισορρόπηση φόρτου επειδή μπορούν να επιλέξουν την πιο αποτελεσματική διαδρομή για τα δεδομένα.
- **Reduction of Routing Overhead**(μείωση της επιβαρυνμένης δρομολόγησης): Η διατήρηση του bandwidth είναι επιτακτική για την επιτυχία οποιουδήποτε ασύρματου δικτύου. Είναι σημαντικό να μειωθεί η επιβάρυνση δρομολόγησης, ιδιαίτερα αυτά που προκαλούνται από επανεκπέμονται.
- **Scalability**(κλιμακοθετησιμότητα): Ένα δίκτυο mesh είναι κλιμακωτό και μπορεί να χρειαστεί εκατοντάδες ή χιλιάδες κόμβους. Επειδή η λειτουργία του δικτύου δεν εξαρτάται από ένα κεντρικό σημείο έλεγχου, προσθέτοντας πολλαπλά σημεία περισυλλογής δεδομένων, ή gateways(πύλες) μετάβασης είναι βολικό. Δεδομένου ότι υπάρχουν χιλιάδες κόμβοι σένα WMN, η υποστήριξη της κλιμάκωσης στα πρωτόκολλα δρομολόγησης είναι σημαντική.
- **QoS Support**: Εξαιτίας της δυνατότητας περιορισμένων καναλιών, η επιρροή της παρεμβολής, ο μεγάλος αριθμός χρηστών και η εμφάνιση των πραγματικού χρόνου multimedia εφαρμογών, η υποστήριξη της ποιότητας εξυπηρέτησης(QoS) έχει γίνει μια κρίσιμη απαίτηση σε τέτοια δίκτυα.

2.3.4 Πολυδιαδρομική δρομολόγηση για υπόλοιπο φόρτου και ανοχή βλαβών

Η ανθεκτικότητα είναι μια ιδιότητα κλειδί στα WMNs. Ένα δίκτυο πλέγματος μπορεί να υποστηρίξει πολλαπλές πορείες ανάμεσα σε δικτυακούς κόμβους, οπότε είναι πιο γερο ενάντια στις αποτυχίες. Οι κόμβοι πλέγματος μπορούν να προστεθούν για να αυξηθεί ο πλεονασμός. Πολλαπλές διαδρομές επιλέγονται μεταξύ κόμβου πηγής και προορισμού. Όταν ένας σύνδεσμος διακόπτεται σε μια πορεία εξαιτίας κακής ποιότητας καναλιού ή κινητικότητας μια άλλη διαδρομή στο σύνολο των υπάρχων διαδρομών μπορεί να επιλεγθεί.

Οπότε χωρίς να περιμένουμε να στηθεί μια καινούργια διαδρομή δρομολόγησης, η καθυστέρηση end-to-end, throughput, και η ανθεκτικότητα σε λάθη μπορεί να βελτιωθούν. Έτσι, με έναν μετρητή επίδοσης, η βελτίωση εξαρτάται από την διαθεσιμότητα των διαδρομών αποκόλλησης κόμβων ανάμεσα στον κόμβο πηγής και προορισμού.

Άλλο αντικείμενο για την χρήση των πολλαπλών διαδρομών δρομολόγησης είναι να έχουμε καλύτερο load balancing(φόρτοςεξισορρόπησης) για να εμποδίσουμε συμφόρηση και να κατευθύνουμε την κίνηση σε άλλη διαδρομή γύρω από τους κόμβους με πολύ κίνηση.

2.3.5 QoS δρομολόγηση

Η QoS δρομολόγηση στα multihop wireless Networks χρειάζεται να παρέχει εγγυημένο bandwidth για μια αίτηση σύνδεσης. Μια αίτηση σύνδεσης θα μπλοκαριστεί εάν δεν μπορούμε να βρούμε μια διαδρομή με εγγυημένο bandwidth.

Το πρόβλημα είναι περίπλοκο όταν το bandwidth για μια διαδρομή επηρεάζεται από την παρεμβολή άλλων διαδρομών μέσα στο δίκτυο. Υπάρχουν δυο διαφορετικά είδη παρεμβολών σένα wireless multihop Networks:

- Interflow παρεμβολή
- Intraflow παρεμβολή

Για μια διαδρομή P, η Interflow παρεμβολή συμβαίνει όταν ένας σύνδεσμος P χρησιμοποιεί το ίδιο κανάλι με άλλον έναν σύνδεσμο που δεν είναι P μέσα στο δικό τους πεδίο παρεμβολής.

Ενώ η Intraflow παρεμβολή συμβαίνει όταν δυο σύνδεσμοι P μέσα στο δικό τους πεδίο παρεμβολής χρησιμοποιεί το ίδιο κανάλι.

Η Intraflow παρεμβολή είναι πιο δύσκολη να αντιμετωπίζεται επειδή η παρεμβολή εξαρτάται από την ίδια την δρομολόγηση, η οποία δεν είναι γνωστή πριν καθοριστεί η δρομολόγηση.

2.4 Μετρικές δρομολόγησης

Τα πρωτόκολλα δρομολόγησης καταγράφουν ή ανακαλύπτουν το ελάχιστο κόστος ή το ελάχιστο βάρος διαδρομών ανάμεσα στον κόμβο πηγής και προορισμού. Το κόστος/βάρος ορίζεται μέσω του μετρητή δρομολόγησης. Η κάθε διαδρομή έχει έναν μετρητή διαδρομής που είναι συνήθως το σύνολο όλων των μετρητών συνδέσμων πάνω στην διαδρομή. Άλλες αλληλουχίες των μετρητών συνδέσμων είναι πιθανά επίσης.

Οι μετρητές δρομολόγησης για τα WMNs πρέπει να εκπληρώσουν 4 απαιτήσεις:

- 1) Να εξασφαλίσουν σταθερότητα διαδρομών, δηλαδή, όχι συχνές αλλαγές διαδρομών
- 2) Καθορισμένες ελάχιστου κόστους/βάρους έχουν καλή επίδοση

- 3) Αποτελεσματικοί αλγόριθμοι για τον υπολογισμό του ελάχιστου κόστους/βάρους διαδρομής
- 4) Διασφάλιση loop χωρίς προώθηση

Διαφορετικοί μετρητές δρομολόγησης είναι πιθανόν. Η εκμετάλλευση μιας ορισμένης ιδιότητας ενός δικτύου πλέγματος μπορεί να χρειαστεί έναν ειδικό μετρητή δρομολόγησης,για παράδειγμα για την χρήση multiple interfaces ή multiple channels. Μερικοί μετρητές μπορεί να μην λειτουργήσουν με όλο των ειδών πρωτοκόλλων δρομολόγησης. Τα παρακάτω είναι μια λίστα μερικών υπάρχων μετρητές δρομολόγησης για τα WMNs:

- Hopcount είναι ο κλασσικός μετρητής δρομολόγησης,ο οποίος είναι εύκολος να καθορίσουμε. Όμως δεν δίνει πληροφορίες για το ασύρματο περιβάλλον, εκτός ότι δυο κόμβοι έχουν απευθείας σύνδεσμο.
- Expected Transmission Count (ETX) μετρητής προβλέπει τον αριθμό των απαιτούμενων μεταδόσεων για να στείλεις ένα πακέτο δεδομένων στον σύνδεσμο,το οποίο περιλαμβάνει αναμεταδόσεις. Το ETX υπολογίζεται από την προς και αντίστροφη αναλογία διαμονής ενός συνδέσμου.
- Expected Transmission Time (ETT) μπορούμε να θεωρήσουμε ότι είναι εφάμιλλη με την ETX μετρική όπου τι S είναι το μέγεθος του πακέτου. Ο ρυθμός μετάδοσης είναι το B και μπορεί να εκτιμηθεί με τα ζεύγη των πακέτων. Κάθε κόμβος στέλνει 2 back-to-back ανιχνευτικά πακέτα σε κάθε έναν από τους γείτονές του περιοδικά. Το πρώτο ανιχνευτικό πακέτο είναι πολύ μικρό, ενώ το δεύτερο είναι μάλλον μεγαλύτερο. Ο χρόνος παραλαβής των 2 ανιχνευτικών πακέτων μετρείται και αποστέλεται στον κεντρικό. Το μέγεθος του δεύτερου ανιχνευτικού πακέτου διαιρείται με το μικρότερο από τα 10 διαδοχικά δείγματα προκειμένου να εκτιμηθεί ο ρυθμός δεδομένων. Η διαδρομή της μετρικής είναι η πρόσθεση όλων των ETT της γραμμής της σύνδεσης.
- Weighted Cumulative Expected Transmission Time (WCETT) είναι μια επέκταση του ETT μετρητή και προσπαθεί να ελαχιστοποιήσει την Intraflow παρεμβολή επιβάλλοντας ποινή σε διαδρομές που έχουν περισσότερες μεταδόσεις στο ίδιο κανάλι ή radio. Το WCETT αναπτύχθηκε για multi-radio linkquality source routing (MR-LQSR) πρωτόκολλο δρομολόγησης.
- Metric of Interference and Channel-Switching (MIC) βελτιώνει τον WCETT και επίσης αιχμαλωτίζει την Interflow παρεμβολή.
- Airtime Link Metric είναι ένα μέτρο δρομολόγησης για τους πόρους του καναλιού που έχουν καταναλωθεί όταν μεταδίδεται ένα πλαίσιο σε έναν συγκεκριμένο σύνδεσμο. Αυτός ο μετρητής προτείνεται για το ανερχόμενο IEEE 802.11s WLAN mesh networking

2.5 Πρωτόκολλα δρομολόγησης

Κυρίως το κάθε πρωτόκολλο δρομολόγησης μπορεί να επεκταθεί για να λειτουργήσει με πολλαπλές ασύρματες συνδέσεις σε έναν κόμβο πλέγματος. Το ίδιο ισχύει για τους μετρητές δρομολόγησης, αλλά είναι πιο περίπλοκο. Ανάλογα με την πραγματική μετρική δρομολόγησης, πιο ουσιαστικές επεκτάσεις μπορεί να είναι απαραίτητες, όπως μπορεί να φανεί στην υπό on-demand μέρος του Hybrid Wireless Mesh Protocol (HWMP).

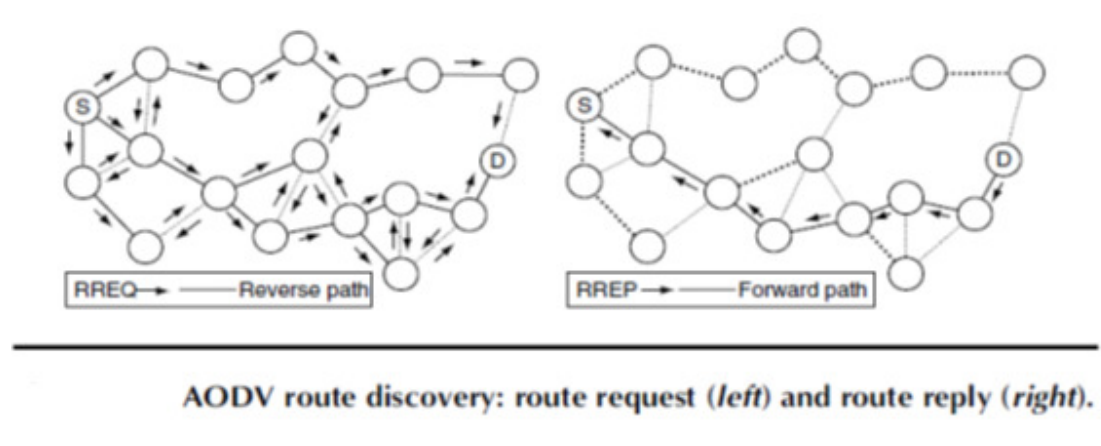
2.5.1 Ad hoc On-demand Distance Vector πρωτόκολλο δρομολόγησης (AODV)

Το AODV είναι ένα πολύ δημοφιλή πρωτόκολλο δρομολόγησης στα MANETs. Είναι ένα Reactive πρωτόκολλο δρομολόγησης. Οι διαδρομές στήνονται μετά από ζήτηση, και μόνο οι ενεργές διαδρομές διατηρούνται. Αυτό μειώνει την επιβάρυνση της δρομολόγησης αλλά εισάγει κάποια αρχική λανθάνουσα κατάσταση εξαιτίας της on-demand ρύθμισης δρομολόγησης. Πρόσφατα μια προσαρμογή του AODV έχει προταθεί για τα WLAN mesh networking.

Το AODV χρησιμοποιεί ένα απλό μηχανισμό request-reply για την ανακάλυψη των διαδρόμων. Μπορεί να χρησιμοποιεί hello μηνύματα για πληροφορίες σύνδεσης και σήματα επαναλαμβανόμενα σύνδεσης σε ενεργές διαδρομές με μηνύματα σφάλματος.

Η κάθε πληροφορία δρομολόγησης έχει ένα τέλος χρόνου που σχετίζεται με αυτό καθώς και έναν αριθμό ακολουθίας. Η χρήση των αριθμών ακολουθίας επιτρέπει να ενσωματώσουμε τα ξεπερασμένα δεδομένα, έτσι ώστε μόνο οι πιο πρόσφατες διαθέσιμες πληροφορίες δρομολόγησης να χρησιμοποιούνται.

Αυτό εξασφαλίζει ελευθερία των loops δρομολόγησης και αποφεύγει προβλήματα γνωστά από τα κλασικά πρωτόκολλα distance vector, όπως το “counting to infinity”.



Όταν ένας κόμβος πηγής S θέλει να στείλει πακέτα δεδομένων σε έναν κόμβο προορισμού D αλλά δεν έχει μια διαδρομή D στον πίνακα δρομολόγησης του, τότε μια ανακάλυψη διαδρομής πρέπει να γίνει από τον S. Τα πακέτα δεδομένων αποθηκεύονται κατά την διάρκεια της ανακάλυψης διαδρομών. Ο κόμβος πηγής S εκπέμπει αίτημα διαδρομής (RREQ) throughout σόλο το δίκτυο. Καθώς και πολλές σημαίες, ένα πακέτο RREQ περιέχει Hopcount, ένα αναγνωριστικό RREQ, την διεύθυνση προορισμού και τον αριθμό ακολουθίας του προορισμού, και την διεύθυνση αποστολέα και τον αριθμό ακολουθίας του αποστολέα.

Ο τομέας Hopcount περιέχει την απόσταση αποστολέα των RREQ, τον κόμβο πηγής S. Είναι ο αριθμός των hops που το RREQ έχει ταξιδέψει μέχρι τώρα. Το RREQ ID σε συνδυασμό με την διεύθυνση αποστολέα αναγνωρίζει μοναδικά μια διαδρομή αιτήματος. Αυτό χρησιμοποιείται για να βεβαιωθούμε ότι ένας κόμβος κάνει αναμετάδοση μιας διαδρομής αιτήματος μόνο μια φορά για να αποφύγει καταιγίδες εκπομπών, ακόμα και όταν ένας κόμβος λαμβάνει το RREQ πολλές φορές από τους γείτονες του.

Όταν ένας κόμβος λαμβάνει ένα RREQ πακέτο επεξεργάζεται τα εξής:

- Την διαδρομή στο προηγούμενο hop από το οποίο το RREQ πακέτο έχει ληφθεί δημιουργείται ή ενημερώνεται.
- Το RREQ ID και η διεύθυνση αποστολέα ελέγχεται για να δει εάν αυτό το RREQ έχει ήδη ληφθεί. Εάν έχει ληφθεί, το πακέτο απορρίπτεται.
- Το Hopcount αυξάνεται κατά 1.
- Η αντίστροφη διαδρομή στην πηγή, κόμβος S, δημιουργείται ή ενημερώνεται.
- Εάν ο κόμβος είναι απαιτούμενος προορισμός, δημιουργεί μια διαδρομή απάντησης (RREP) και στέλνει το RREP πακέτο πίσω στον αποστολέα κατά μήκος της αντίστροφης διαδρομής που έχει δημιουργηθεί στον κόμβο S.
- Εάν ο κόμβος δεν είναι προορισμός αλλά έχει μια έγκυρη διαδρομή στο D, στέλνει ένα RREP στην πηγή που εξαρτάται από την σημαία προορισμού. Εάν οι ενδιάμεσοι κόμβοι απαντούν στο RREQ, υπάρχει περίπτωση ο προορισμός να μην ακούσει κανένα RREQ, έτσι ώστε να μην έχει μια πορεία επιστροφής στην πηγή. Εάν τα χαρακτηριστικά RREP σημαίας που έχει δοθεί τοποθετείται στο RREQ, ο ενδιάμεσος κόμβος που ανταποκρίνεται θα στείλει RREP στον προορισμό. Αυτό θέτει την διαδρομή στον αποστολέα του RREQ στον προορισμό.
- Εάν ο κόμβος δεν παράγει RREP, το RREQ ενημερώνεται και εκπέμπεται ξανά εάν TTL είναι ≥ 1 .

Μόλις ληφθεί ένα RREP μήνυμα, ένας κόμβος θα δημιουργήσει ή θα ενημερώσει την διαδρομή του στον προορισμό D. Η Hopcount αυξάνεται με 1, και το αναβαθμισμένο RREP θα σταλθεί στον αποστολέα του αντίστοιχου RREQ. Τελικά ο κόμβος πηγής S θα λάβει ένα RREP εάν υπάρχει μια διαδρομή για τον προορισμό. Τα αποθηκευμένα πακέτα δεδομένων μπορούν να σταλούν τώρα στον προορισμό D στη νέα διαδρομή που έχει ανακαλυφθεί.

Πληροφορίες συνδεσιμότητας που παρέχονται και συντηρούνται από περιοδική μετάδοση του πρωτοκόλλου δρομολόγησης μηνυμάτων. Εάν ένας κόμβος δεν έχει στείλει ένα μήνυμα εκπομπής, π.χ ένα RREQ μήνυμα, μέσα στο τελευταίο hello διαστήματος, ο κόμβος μπορεί να εκπέμψει ένα hello message. Ένα hello είναι πραγματικά RREP με TTL=1 και ο ίδιος ο κόμβος ως προορισμό. Εάν ένας κόμβος δεν λάβει κανένα πακέτο από ένα γειτονικό κόμβο για κάποιο καθορισμένο χρόνο, ο κόμβος λαμβάνει υπόψιν το σύνδεσμο με εκείνο τον διακεκομμένο γείτονα. Όταν συμβαίνει μια αστοχία σύνδεσης, ο κόμβος που βρίσκεται πριν

από τον διακεκομμένο σύνδεσμο ελέγχει πρώτα εάν οπουδήποτε ενεργή διαδρομή είχε χρησιμοποιήσει αυτόν τον σύνδεσμο. Εάν δεν συμβεί αυτό, δεν θα γίνει τίποτα. Από την άλλη μεριά, εάν υπήρχαν ενεργές διαδρομές, ο κόμβος μπορεί να προσπαθήσει να κάνει τοπική επιδιόρθωση. Στέλνει ένα RREQ για να καθιερώσει ένα καινούργιο δεύτερο μισό της διαδρομής προς τον προορισμό. Ο κόμβος που εκτελεί την τοπική επιδιόρθωση buffers τα πακέτα δεδομένων καθώς περιμένει οποιαδήποτε απάντηση διαδρομής. Εάν η τοπική επιδιόρθωση αποτύχει ή δεν έχει αποπειραθεί, ο κόμβος παράγει ένα μήνυμα σφάλματος διαδρομής (RERR). Περιεχει τις διευθύνσεις και τους αντίστοιχους αριθμούς σειράς προορισμού όλων των ενεργών προορισμών που έχουν γίνει απρόσιτοι εξαιτίας της αστοχίας της σύνδεσης. Το RERR μήνυμα το στέλνουν σε όλους γείτονες που είναι πρόδρομοι των απρόσιτων προορισμών σε αυτόν τον κόμβο. Ένας κόμβος που λαμβάνει ένα RERR ακυρώνει τους αντίστοιχους εισόδους στον πίνακα δρομολόγησης του. Αφαιρεί όλους τους προορισμούς που δεν έχουν τον αναμεταδότη του RERR ως το επόμενο hop από την λίστα των απρόσιτων προορισμών. Εάν υπάρχουν πρόδρομοι των προορισμών σε αυτό αποκόπτονται από την λίστα, το ενημερωμένο RERR μήνυμα προωθείται σε αυτούς.

2.5.2 Dynamic Source Routing Protocol (DSR)

Το DSR είναι ένα από τα πρωτοποριακά πρωτόκολλα δρομολόγησης για τα MANETs. Ακόμη το DSR τροποποιείται από το IETF MANET ομάδα εργασίας. Το DSR είναι ένα γνωστό Reactive πρωτόκολλο δρομολόγησης. Υπολογίζει μια πορεία μόνο εάν χρειάζεται. Η ανακάλυψη διαδρομής αποτελείται από την ζήτηση διαδρομής και από την απάντηση διαδρομής. Η ζήτηση διαδρομής εκπέμπεται στο ασύρματο δίκτυο. Όμως αντί να τοποθετηθούν οι (αντίστροφες) διαδρομές στους πίνακες δρομολόγησης των κόμβων, η ζήτηση της διαδρομής μαζεύει τις διευθύνσεις των κόμβων που έχουν ταξιδέψει στο δρόμο προς τον προορισμό. Η απάντηση διαδρομής στέλνει αυτήν την διαδρομή πίσω στην πηγή όπου όλες οι διαδρομές αποθηκεύονται σένα cache διαδρομής. Η διαδρομή, δηλ. η λίστα των διευθύνσεων από την πηγή έως τον προορισμό, περιλαμβάνονται στην επικεφαλίδα κάθε πακέτου από τον κόμβο πηγής. Ο κάθε κόμβος προωθεί ένα πακέτο που έχει ληφθεί στο επόμενο hop στην λίστα των διευθύνσεων της επικεφαλίδας (δρομολόγηση πηγής). Το DSR χρησιμοποιεί RERR μηνύματα για την ειδοποίηση των διαλειμμάτων διαδρομής.

2.5.3 Optimized Link State Routing Protocol (OLSR)

Το OLSR είναι γνωστό Proactive πρωτόκολλο δρομολόγησης για ασύρματα ad hoc δίκτυα. Το OLSR χρησιμοποιεί την κλασική πιο σύντομη αλγορίθμου διαδρομή που βασίζεται στην Hopcount μετρική για τον υπολογισμό διαδρομών στο δίκτυο. Όμως η κυρία έννοια του OLSR είναι ένας μέγιστος μηχανισμός εκπομπής για την διανομή σόλο το δίκτυο των απαραίτητων πληροφοριών link-state.

Ο κάθε κόμβος επιλεγεί τα multipoint relays (MPRs) ανάμεσα στους γείτονες του με τέτοιο τρόπο ώστε όλοι οι 2-hop γείτονες λαμβάνουν μηνύματα εκπομπής ακόμα κι αν τα MPRs αναμεταδιδόσουν τα μηνύματα. Η προώθηση των μηνυμάτων μετάδοσης από τα MPRs μόνο μπορεί να μειώσει σημαντικά τον αριθμό των μηνυμάτων που μεταδίδονται.



Multipoint relay selection in OLSR

Στην εικόνα φαίνεται ο αριθμός των μηνυμάτων που έχουν αναμεταδοθεί μειώνεται στην μέση.

Αυτός ο μηχανισμός μέγιστης προώθησης χρησιμοποιείται για όλες τις εκπομπές σένα OLSR δίκτυο. Επίσης, η ποσότητα των πληροφοριών link-state που διανέμεται μέσα στο δίκτυο μπορεί να μειωθεί με το OLSR, επειδή μόνο οι link-state πληροφορίες σ όλους τους επιλεγείς MPR είναι απαραίτητες για τον απολογισμό μικρότερης διαδρομής.

Ο κάθε κόμβος περιοδικά εκπέμπει hello μηνύματα για τοπικό τοπολογικό εντοπισμό. Τα hello μηνύματα δεν προωθούνται (TTL=1) και περιέχουν μια λίστα των γειτόνων του αποστέλλοντος κόμβου. Ο κάθε κόμβος στο WMN θα γνωρίζει τον δικό του 2-hop γείτονα μέσω αυτού του hello μηχανισμού. Είναι επίσης πιθανό να επαληθεύσουμε δυο κατευθύνσεις των συνδέσεων. Το OLSR αποδίδει την κατάσταση (asymmetric, symmetric) στον κάθε σύνδεσμο. Επιπλέον, ο κάθε κόμβος ανακοινώνει την προθυμία του να προωθήσει πακέτα μέσα στα hello μηνύματα. Οι πληροφορίες των hello μηνυμάτων αποθηκεύονται σε αρκετές

αποθήκες πληροφοριών: το σύνολο συνδέσμων, το σύνολο γειτόνων και το 2-hop σύνολο γειτόνων.

Με αυτήν την γνώση ο κάθε κόμβος μπορεί τώρα να υπολογίσει το σύνολο multipoint αναμετάδοσης. Ο κάθε κόμβος υπολογίζει το σύνολο MPR του ανεξάρτητα από τους άλλους κόμβους κυρίως με βάση την τοπολογία που έχει ληφθεί τοπικά. Η μόνη απαίτηση είναι ότι η ολοκληρωμένη 2-hop γειτονιά θα λάβει μηνύματα μετάδοσης εάν τα MPR τα στέλνουν μόνο και οι συμμετρικοί σύνδεσμοι ληφθούν υπόψη. Δεν είναι απαραίτητο ότι το σύνολο των MPR να είναι ελάχιστο, αλλά ένα μικρότερο σύνολο MPR να κρατάει χαμηλά την επιβάρυνση του πρωτοκόλλου. Το OLSR προτείνει ένα απλό ευρετήριο για το MPR άλλα και άλλοι αλγόριθμοι είναι πιθανοί.

Οι επιλεγμένοι multipoint κόμβοι αναμετάδοσης αποθηκεύονται στο σύνολο των MPR. Οι γείτονες που έχουν επιλεγθεί ως MPR, θα έχουν μια κατάσταση συνδέσμου που δείχνει την MPR επιλογή hello μηνύματα. Ένας κόμβος που λαμβάνει ένα hello μήνυμα μπορεί να πάρει από αυτές τις πληροφορίες από εκείνους τους κόμβους που το επέλεξαν ως MPR. Αυτοί οι MPR επιλογείς αποθηκεύονται στο MPR σύνολο επιλογής.

Ο κάθε κόμβος περιοδικά εκπέμπει τις πληροφορίες κατάστασης συνδέσμων του μεσοί ολοκλήρου του OLSR δικτύου με μηνύματα έλεγχου (TC). Ένα TC μήνυμα περιέχει μια λίστα γειτόνων του αρχικού κόμβου. Αυτή η λίστα γειτόνων πρέπει να περιέχει τουλάχιστον όλους τους MPR επιλογείς αυτού του κόμβου για να εγγυηθεί πιο σύντομες διαδρομές σε σχέση με το Hopcount. Κάθε TC μήνυμα έχει έναν διαφημισμένο αριθμός ακολουθίας γειτόνων που σχετίζεται με την λίστα γειτόνων που επιτρέπει να απορρίψουμε ξεπερασμένες πληροφορίες τοπολογίας. Οι πληροφορίες των TC μηνυμάτων αποθηκεύονται στο σύνολο τοπολογίας.

Ο OLSR πίνακας δρομολόγησης που περιεχί καταχωρήσεις για όλους τους προσβάσιμους προορισμούς με το δίκτυο πλέγματος (Proactive routing protocol) υπολογίζεται από το σύνολο των συνδέσμων, το σύνολο γειτόνων, το σύνολο 2-hop γειτόνων, και το σύνολο τοπολογίας με ένα κλασσικό αλγόριθμο πιο σύντομης διαδρομής (π.χ Dijkstra algorithm). Εάν οποιοδήποτε από τα παραπάνω σύνολα έχει αλλάξει, ο πίνακας δρομολόγησης πρέπει να υπολογίζεται ξανά.

Επιπλέον, μπορεί να είναι χρήσιμο να στείλουμε ένα hello ή TC μήνυμα για να διαδώσουμε την αλλαγή της τοπολογίας αμέσως. Όλες οι καταχωρήσεις των αποθηκευμένων πληροφοριών, π.χ το σύνολο γειτόνων, έχουν ένα χρόνο λήξης που σχετίζεται με αυτές. Αυτός ο μηχανισμός μαλακής κατάστασης παρέχει μερική στιβαρότητα ενάντια της απώλειας OLSR πακέτων έλεγχου.

Ο OLSR μπορεί επίσης να αντιμετωπίσει πολλαπλές (OLSR) διεπαφές σένα κόμβο. Ένας τέτοιος κόμβος επιλέγει την διεύθυνση από οποιαδήποτε από τις διευθύνσεις του, ως η κύρια διεύθυνση και εκπέμπει περιοδικά πολλαπλή διεπαφή διασύνδεσης (MID) μηνύματα. Τα MID μηνύματα διανέμουν τη σχέση ανάμεσα της κύριας διεύθυνσης και άλλες διεπαφές διευθύνσεων. Προφανώς, ένας κόμβος με μια μοναδική διεπαφή OLSR δεν είναι αναγκασμένο να στείλει MID μηνύματα.

2.5.4 Cross-Layer προσέγγιση δρομολόγησης

Η παρεμβολή στα ασύρματα δίκτυα υποβαθμίζει δραματικά την επίδοση του δικτύου. Η παρεμβολή σχετίζεται απευθείας με την τάση της μετάδοσης. Μεγαλύτερη ισχύ μετάδοσης σημαίνει πιο αξιόπιστες συνδέσεις με μεγαλύτερη χωρητικότητα.

Από την άλλη μεριά, η μεγαλύτερη ισχύ μετάδοσης σημαίνει επίσης περισσότερες παρεμβολές, οπότε λιγότερο throughput στο δίκτυο. Οπότε, για την παροχή της δρομολόγησης επιπέδου δικτύου με τις πληροφορίες των κατώτερων επιπέδων μπορεί να βοηθήσει να βρούμε πιο αξιόπιστες διαδρομές με μεγαλύτερη χωρητικότητα.

Ένας αλγόριθμος δρομολόγησης cross-layer, ονομάζεται mesh routing strategy (MRS), για να βρεθούν μονοπάτια με υψηλό throughput με μειωμένη παρεμβολή και αυξημένη αξιοπιστία ελέγχοντας την ισχύ της μετάδοσης.

Παρατηρείται ότι όσο περισσότερη (λιγότερη) ισχύ χρησιμοποιείται, τόσο χαμηλότερο (υψηλότερο) είναι το packet error rate (PER), αλλά υψηλότερο (χαμηλότερο) είναι η παρεμβολή. Το MRS ψάχνει την βέλτιστη συναλλαγή ρυθμίζοντας μια βέλτιστη μετάδοση ισχύ επιπέδου που ελαχιστοποιεί την απόσταση από το ιδανικό ανώτερο σημείο.

Το MRS επεξεργάζεται την βελτιστοποίηση της τοπικής ισχύς και την ανακάλυψη της δρομολόγησης ξεχωριστά.

Μια τέτοια στρατηγική two-step λειτουργεί όπως παρακάτω:

- Αρχικά, μέσα από το πρωτόκολλο ανακάλυψης γειτόνων, ο κάθε κόμβος εξερευνά την γειτονιά του, υπολογίζει τους μετρητές, όπως ρυθμός μετάδοσης, παρεμβολή και PER, και καθορίζει την τοπική μετάδοση. Μετά από αυτό οι τοπικοί σύνδεσμοι διαφημίζονται.
- Οποτεδήποτε ένα γεγονός ενεργοποιεί μια αλλαγή στις μετρικές δρομολόγησης ενός ή περισσότερων συνδέσμων, η βελτιστοποίηση ενέργειας εκτελείται στον σύνδεσμο που αφορά και ξεκινά η ενημέρωση της διαδρομής.
- Μόλις αναγνωριστεί η καλύτερη μετρική (η οποία είναι μια σχετικά σταθερή κατάσταση) η σύνδεση διαφημίζεται. Το πρωτόκολλο δρομολόγησης MRS επιλέγει τα βέλτιστα μονοπάτια για να φτάσει οποιαδήποτε άλλη ασύρματη mesh δρομολόγηση του δικτύου με μια προσέγγιση distance vector.

2.5.5 Ενήμερο εύρος ζώνης δρομολόγησης

Σε μια εργασία πάνω σε QoS δρομολόγηση για WMN, οι συγγραφείς συζητούν για ενήμερη-παρεμβολή έλεγχου τοπολογίας και για δρομολόγηση QoS σε multichannel WMN.

Παρουσιάζουν μια έννοια της co-channel παρεμβολής και αναπτύσσουν έναν αλγόριθμο εύρεσης για να στήσουν ένα WMN που να έχει την ελάχιστη παρεμβολή μέσα σε όλες τις K-συνδεδεμένες τοπολογίες. Έπειτα, αυτοί εισάγουν το bandwidth-aware πρόβλημα δρομολόγησης για QoS δρομολόγηση με απαιτήσεις bandwidth.

Εάν η κίνηση για μια αίτηση σύνδεσης είναι διαιρούμενη, δηλ. να χρησιμοποιούν πολλαπλές διαδρομές για να ικανοποιήσουν την ζήτηση για bandwidth, αυτοί δείχνουν ότι το πρόβλημα μπορεί να λυθεί με μια γραμμικού προγραμματισμού (LP) διατύπωση.

Για την περίπτωση όπου το bandwidth για ένα αίτημα μπορεί να ικανοποιηθεί με μια μοναδική διαδρομή, ένας αλγόριθμος εύρεσης αναπτύσσεται προσδιορίζοντας την μεγίστη χωρητικότητα συμφόρησης διαδρομής.

Αποτελέσματα προσομοιώσεων παρέχονται επίσης για να δείξουν την αποτελεσματικότητα του προτεινόμενου σχεδίου.

2.5.6 Multi-Radio Link-Quality Source Routing (MR-LQSR)

Σε ένα WMN, κάποια υποβάθμιση του throughput μπορεί να αναμένεται μετά από 5 ή 6 hops. Η παρεμβολή καναλιού μπορεί να έχει ως αποτέλεσμα χαμηλότερο throughput εάν οι κόμβοι είναι πολύ κοντά ο ένας με τον άλλον ή εάν η ισχύ είναι πολύ υψηλή για την περιοχή.

Τα WMN πρωτόκολλα δρομολόγησης πρέπει να επιλέγουν διαδρομές με βάση την παρατηρούμενη λανθάνουσα κατάσταση και το ασύρματο περιβάλλον καθώς και άλλους παράγοντες επίδοσης, που έχει ως αποτέλεσμα το καλύτερο δυνατόν throughput σόλο το δίκτυο. Για να αυξήσουμε την χωρητικότητα των WMN, οι κόμβοι μπορεί να έχουν πολλαπλά radios, προτιμότερο δουλεύοντας σε διαφορετικά κανάλια ή σε διαφορετικές ζώνες.

Όμως υπάρχουν μερικά θέματα με την ποικιλία καναλιών μέσα σε πολλαπλά radios, τα οποία τα πρωτόκολλα δρομολόγησης πρέπει να λάβουν υπόψιν. Διαφορετικές ταχύτητες μετάδοσης και ζώνες έχουν διαφορετικές εμβέλειες μετάδοσης. Τα πρωτόκολλα δρομολόγησης μικρότερης διαδρομής θα προτιμήσουν τους συνδέσμους με μεγαλύτερη εμβέλεια μετάδοσης, τα οποία είναι συνήθως οι σύνδεσμοι με το χαμηλότερο ποσοστό μετάδοσης. Επίσης, δεν θα υπάρχει αύξηση στην αποθήκευση εάν τα multiple radios του mesh κόμβου χρησιμοποιεί το ίδιο κανάλι εξαιτίας της παρεμβολής. Επομένως, οι διαδρομές με ποικιλομορφία καναλιών (radio) πρέπει να προτιμάται.

Ο μετρητής WCETT λαμβάνει υπόψιν του την ποιότητα του συνδέσμου, την ποικιλία καναλιών, και το ελάχιστο hopcount. Μπορεί να επιτύχει μια καλή συναλλαγή μεταξύ στην καθυστέρησης και throughput επειδή λαμβάνει υπόψιν κανάλια με καλή ποιότητα και ποικιλία καναλιού την ίδια στιγμή. Το MR-LQSR έχει αναπτυχθεί για multiradio multihop WMNs με βάση τον μετρητή WCETT.

Το MR-LQSR αναθέτει ένα βάρος στο κάθε σύνδεσμο, το οποίο είναι ο αναμενόμενος χρόνος που χρειάζεται να μεταδώσουμε επιτυχώς ένα πακέτο σταθερού μεγέθους S πάνω σε αυτό τον σύνδεσμο. Αυτός ο χρόνος εξαρτάται από τον ρυθμό μετάδοσης του συνδέσμου και τον ρυθμό απώλειας. Εάν δοθεί μια σύνδεση i από τον κόμβο x στον κόμβο y , ο αναμενόμενος χρόνος μετάδοσης ETT του πακέτου σε αυτήν την σύνδεση μετράται. Δηλώνουμε αυτή την αξία με ETT $_i$.

Η μετρική δρομολόγησης στο MR-LQSR προσπαθεί να ισορροπήσει την συναλλαγή ανάμεσα στο throughput και στην καθυστέρηση. Όμως, το σύνολο των ETT όλων των συνδέσεων μιας διαδρομής αντικατοπτρίζει μόνο την καθυστέρηση end-to-end. Η επιρροή της διαφορετικότητας καναλιών δεν λαμβάνεται υπόψιν, και όποτε, πρέπει να επεκταθεί.

Παρατηρείται ότι ο χρόνος μετάδοσης πάνω σε μια ζεύξη καθορίζεται από το διαθέσιμο bandwidth, το οποίο καθορίζεται περισσότερο από την παρεμβολή ενός καναλιού.

Συντηρητικά, λαμβάνοντας υπόψιν μια n-hop διαδρομή, υποθέτουμε ότι οποιαδήποτε δυο hops ανάμεσα στα n hops εμποδίζουν ο ένας τον άλλον εάν μοιραστούν ένα κανάλι.

Θέτουμε το X_j ως:

$$X_j = \sum_{\text{Hop } i \text{ is on channel } j} \text{ETT}_i, \quad 1 \leq j \leq k$$

Όποτε X_j είναι το άθροισμα των χρόνων μετάδοσης των hops στο κανάλι j . Το υψηλότερο X_j δείχνει χαμηλότερο διαθέσιμο bandwidth στην κάθε ζεύξη που χρησιμοποιεί το κανάλι j . Το συνολικό throughput διαδρομής θα κυριαρχήσει από το κανάλι που στέλνει, το οποίο έχει το μεγαλύτερο X_j .

Στο MR-LQSR, WCETT ορίζεται ως:

$$\text{WCETT} = (1 - \beta) \times \sum_{i=1}^n \text{ETT}_i + \beta \times \max_{1 \leq j \leq k} X_j$$

Ο πρώτος όρος αντικατοπτρίζει το λάθος αυτής της διαδρομής. Ο δεύτερος όρος αντιπροσωπεύει το throughput της διαδρομής. Ο σταθμισμένος μέσος όρος προσπαθεί να εξισορροπήσει τα δύο.

2.5.7 Άλλα πρωτόκολλα δρομολόγησης βασισμένη-τοπολογία για WMNs

Πολλά πρωτόκολλα δρομολόγησης που έχουν προταθεί για wireless multihop networks. Θα αναφερθούν λίγα πρωτόκολλα δρομολόγησης. Για περισσότερες πληροφορίες ο χρήστης μπορεί να συνδεθεί στο internet (World Wide Web).

1. Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) είναι ένα proactive πρωτόκολλο δρομολόγησης. Το TBRPF χρησιμοποιείται σε μερικές εγκαταστάσεις και προϊόντα WMNs.
2. Dynamic on-demand MANET routing protocol (DYMO) είναι ένα reactive πρωτόκολλο δρομολόγησης και περιέχει την βασική ανακάλυψη διαδρομής και

χαρακτηριστικά συντήρησης παρόμοια με το AODV και έναν μηχανισμό για μελλοντικές βελτιώσεις.

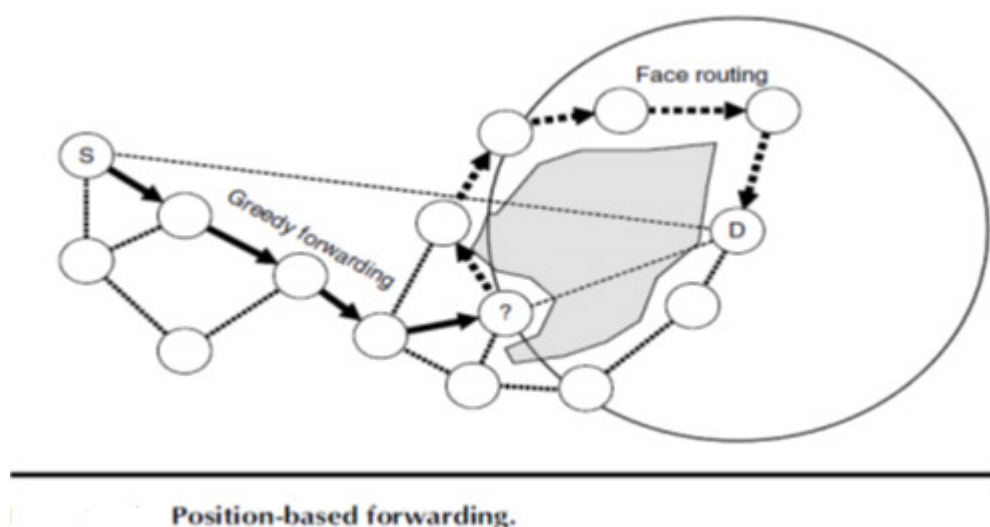
3. OSPF-MANET είναι μια συνεχής προσπάθεια του IETF να προσαρμόσουν το Open Shortest Path First (OSPF) στα wireless multihop networks. Το πλεονέκτημα του OSPF-MANET είναι η εύκολη ενσωμάτωση των WMNs και των MANETs σε υπάρχον(ασύρματα) OSPF δίκτυα. Παρόμοιο με το OLSR,το OSPF-MANET,η υπερχειλίση με καταστάσεις ζεύξης δημοσιεύσεις θα μειωθεί. Όμως,το OSPF-MANET συγκεντρώνεται στο connected dominating sets (CDS) για την μείωση των επανεπέμψων. Προφανώς,το OSPF-MANET είναι ένα proactive πρωτόκολλο δρομολόγησης

4. Fisheye State Routing (FSR) είναι ένα proactive πρωτόκολλο δρομολόγησης που χρησιμοποιεί την “fisheye” έννοια για μείωση των απαραίτητων μηνυμάτων που μεταδίδονται για την διανομή τοπολογιών πληροφοριών. Οι κόμβοι που είναι πιο κοντά σε έναν κόμβο N λαμβάνουν τοπολογικές πληροφορίες πιο συχνά από τους μακρινούς κόμβους. Αυτό γίνεται αυξάνοντας το TTL των μηνυμάτων για την κάθε υπερχειλίση μέχρι την μέγιστη αξία πριν συνεχίσει με την αρχική,μικρή αξία.

5. Zone Routing Protocol (ZRP) έχει υβριδική έννοια και αποτελείται από proactive δρομολόγηση στην κοντινή γειτονιά των κόμβων και της reactive δρομολόγησης για πιο μακρινούς κόμβους,το οποίο μειώνει τα μειονεκτήματα και των δυο μεθόδων καθώς αξιοποιεί τα πλεονεκτήματα τους.

2.5.8 Πρωτόκολλα δρομολόγησης βασισμένης-θέσης

Σε αυτήν την κατηγορία των αλγορίθμων δρομολόγησης,τα πακέτα προωθούνται με βάση τις γεωγραφικές τοποθετήσεις του προωθημένου κόμβου,των γειτόνων του,και τον προορισμό.



Αυτό απαιτεί ο κάθε κόμβος να γνωρίζει τη δική του γεωγραφική τοποθεσία. Η τοποθεσία του προορισμού πρέπει να παρέχεται από μια υπηρεσία τοποθεσίας. Ένας απλός αλγόριθμος προώθησης όπως η άπληστη προώθηση μπορεί να χρησιμοποιηθεί με αυτές τις πληροφορίες τοποθεσίας. Το πακέτο στέλνεται στον πιο κοντινό γείτονα του προορισμού.

Όμως, ο απλός αλγόριθμος προώθησης μπορεί να κολλήσει σένα τοπικά ελάχιστο και να μην μπορεί να φτάσει στον προορισμό παρόλο που υπάρχει μια διαδρομή προς τον προορισμό, όπως απεικονίζεται στην εικόνα.

Face routing συνήθως χρησιμοποιείται ως μια εφεδρική στρατηγική σε μια τέτοια περίπτωση. Το δικτυακό γράφημα χωρίζεται λογικά στις λεγόμενες προσόψεις, όπου οι ζεύξεις που λαμβάνουμε υπόψη δεν διασταυρώνονται. Αυτό το πλάνο του γραφήματος δικτύου μπορεί να γίνει τοπικά με μοιρασμένους αλγορίθμους. Τα πακέτα μπορεί να προχωρήσουν έξω από ένα τοπικό ελάχιστο προωθώντας έξω από αυτές τις προσόψεις προς τον προορισμό.

Η δρομολόγηση πρόσοψης είναι αποδεδειγμένη ότι φθάνει στον προορισμό εάν υπάρχει μονοπάτι, αλλά μπορεί να μην είναι η πιο κοντινή εξαιτίας της "παράκαμψης" στο τοπικό ελάχιστο.

Ένα από τα πρώτα πρακτικά πρωτόκολλα δρομολόγησης με βάση την τοποθεσία για τα ασύρματα δίκτυα είναι το Greedy Perimeter Stateless Routing (GPSR). Συνδυάζει την άπληστη δρομολόγηση με την δρομολόγηση πρόσοψης ως εναλλακτική. Landmark guided forwarding συνδυάζει τοπολογία βασισμένη σε proactive δρομολόγηση για τους μακρινούς κόμβους. Mauve et al. παρέχουν μια καλή επισκόπηση για την δρομολόγηση με βάση την τοποθεσία και για τις υπηρεσίες τοποθεσίας.

2.6 Προτεινόμενη δρομολόγηση για IEEE 802.11s WLAN mesh networking

Τον Ιούλιο του 2004, η WLAN mesh networking ομάδα έρευνας του 802.11 ομάδα εργασίας είχαν την πρώτη τους συνάντηση ως ομάδα εργασίας "s".

Ο στόχος είναι να αναπτύξουν ένα πρότυπο για IEEE 802.11 mesh networks με μέχρι τους 50 mesh κόμβους που μπορούν να αναπτυχθούν σε διαφορετικά σενάρια χρήσης:

- ◆ Περιοχή με κατοικίες
- ◆ Γραφείο
- ◆ Δημόσια πρόσβαση
- ◆ Κοινοτικά δίκτυα
- ◆ Δίκτυα δημόσιας ασφάλειας

Η δρομολόγηση είναι μια από τις βασικές λειτουργίες εκτός από MAC βελτιώσεις για multihop επικοινωνία και ασφάλεια. Η τυποποίηση των WLAN mesh networks στο IEEE 802.11s ήταν ακόμα σε εξέλιξη τη στιγμή που γραφόταν. Οι παρακάτω περιγραφές βασίζονται στην πρόχειρη εκδοχή 0.01 από τον Μάρτιο 2006 IEEE 802.11 της συνάντησης

στο Denver, CO (USA). Καθώς οι γενικές έννοιες του πρωτοκόλλου δρομολόγησης φαίνονται να είναι αρκετά σταθερές, αλλαγές στην λεπτομέρεια είναι πιθανές.

Το πρωτόκολλο δρομολόγησης του ανερχόμενου προτύπου IEEE 802.11s τοποθετείται στο επίπεδο 2 και χρησιμοποιεί επίσης διευθύνσεις MAC. Τα πλαίσια δεδομένων mesh χρησιμοποιούν το 4-διεύθυνση μορφή πλαισίου του IEEE προτύπου 802.11 1999 (Reaff 2003) που επεκτείνεται από το IEEE 802.11e QoS του τομέα της επικεφαλίδας και από καινούργιες επεκτάσεις mesh. Το τελευταίο περιλαμβάνει τον τομέα χρόνου επιβίωσης για την απόλυτη αποφυγή των loops και ένα mesh end-to-end αριθμό ακολουθίας για τον έλεγχο της εκπεμπόμενης υπερχειλίσισης και για να επιτρέψει κατά σειρά διανομή των πλαισίων δεδομένων.

Το 4-διεύθυνσεων σχέδιο παρέχει πεδία διεύθυνσης για μετάδοση και λήψη (τωρινής ζεύξης) καθώς και για πηγή και προορισμό (διαδρομή). Τα μηνύματα δρομολόγησης στέλνονται ως πλαίσια δραστηρικής διαχείρισης. Κληροδοτημένες IEEE 802.11 συσκευές μπορούν να συνδεθούν με ένα πλέγμα μέσω σημείου πρόσβασης πλέγματος χρησιμοποιώντας τις συνηθισμένες μεθόδους του προτύπου. Το σημείο πρόσβασης πλέγματος δρα ως ένας εξουσιοδοτημένος κόμβος πλέγματος για αυτές τις συσκευές. Τα σημεία πρόσβασης πλέγματος είναι κόμβοι πλέγματος με επιπλέον σημείο πρόσβασης λειτουργικότητας.

2.6.1 Μετρική δρομολόγησης ραδιοχρόνου

Η ζεύξη μετρικής *airtime* προτείνεται ως προτερόθετη δρομολόγηση μετρικής *radio-aware* για την βασική λειτουργία ανάμεσα στις συσκευές IEEE 802.11s. Αντικατοπτρίζει την ποσότητα των πόρων καναλιού που καταναλώνεται για την μετάδοση ενός πλαισίου σε μια συγκεκριμένη ζεύξη. Το μονοπάτι με το μικρότερο άθροισμα των μετρικών συνδέσμου χρόνου ομιλίας είναι το καλύτερο μονοπάτι.

Το κόστος *Airtime* c_a για την κάθε ζεύξη υπολογίζεται με την παρακάτω φόρμουλα:

$$c_a = \left[O_{ca} + O_p + \frac{B_t}{r} \right] \frac{1}{1 - e_{fr}}$$

Η επιβάρυνση πρόσβασης καναλιού O_{ca} , τα πρωτόκολλα MAC με επιβάρυνση O_p , και ο αριθμός των bits B_t σε ένα δοκιμασμένο πλαίσιο σταθεροποιούν ποιες τιμές εξαρτώνται από την χρησιμοποιημένη τεχνολογία μετάδοσης IEEE 802,11. Ο ρυθμός μετάδοσης των bit rate r in Mbit/s είναι ο ρυθμός στον οποίο ο κόμβος πλέγματος θα μεταδώσει ένα πλαίσιο μεγέθους B_t που βασίζεται στις τωρινές συνθήκες με τον ρυθμό σφάλματος πλαισίου e_{fr} .

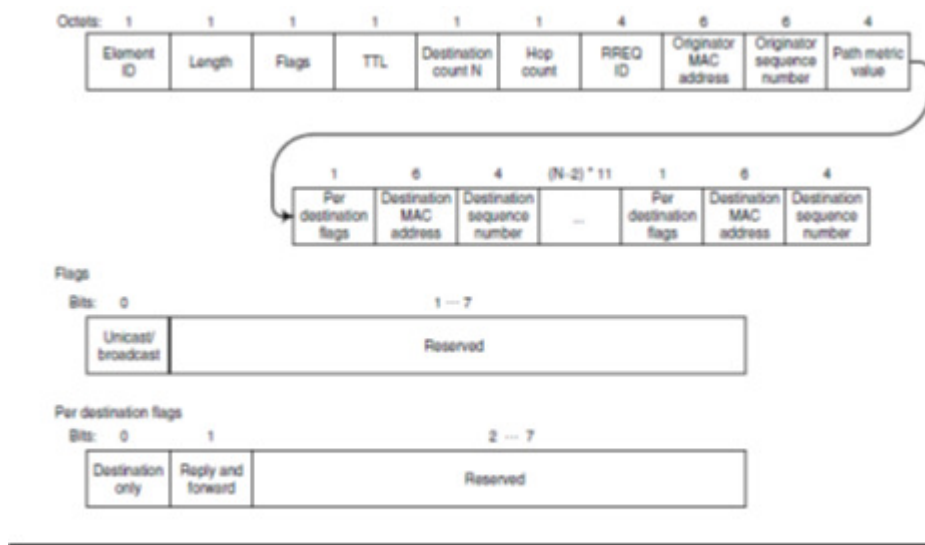
2.6.2 Hybrid Wireless Mesh Protocol (HWMP)

Το HWMP είναι προτερόθετο πρωτόκολλο δρομολόγησης για δικτύωση πλέγματος WLAN. Κάθε συσκευή συμβατή με IEEE 802.11s θα μπορεί να χρησιμοποιήσει αυτό το πρωτόκολλο δρομολόγησης. Η υβριδική φύση και η παραμετροποίηση του HWMP παρέχουν

καλή απόδοση σε όλο το αναμενόμενο σενάριο χρήσης. Το θεμέλιο του HWMP είναι μια προσαρμογή του reactive πρωτόκολλο δρομολόγησης AODV στο επίπεδο 2 και στις radio-aware μετρικές που ονομάζονται radio metric AODV (RM-AODV). Ένας κόμβος πλέγματος, συνήθως ένα mesh portal, μπορεί να ρυθμιστεί να μεταδίδει περιοδικά ανακοινώσεις, το οποίο θέτει ένα δέντρο που επιτρέπει proactive δρομολόγηση προς την κατεύθυνση αυτής της δικτυακής πύλης mesh portal.

Το reactive κομμάτι του HWMP ακολουθεί τις γειτονικές έννοιες του AODV. Χρησιμοποιεί την μεθοδο distance vector και την πασίγνωστη route discovery process με route request και route reply. Ο προορισμός των αριθμών ακολουθίας χρησιμοποιούνται για να αναγνωρίσουν όλες τις παλιές πληροφορίες δρομολόγησης.

Όμως υπάρχουν μερικές σημαντικές διαφορές στις λεπτομέρειες. Το σχήμα δείχνει τη δομή ενός HWMP route request για να τονίσει τα νέα χαρακτηριστικά.



HWMP route request.

Το HWMP χρησιμοποιεί MAC διευθύνσεις όπως ένα επίπεδο 2 πρωτόκολλο δρομολόγησης αντί των διευθύνσεων IP.

Επιπλέον, το HWMP μπορεί να χρησιμοποιήσει πιο περίπλοκους μετρητές δρομολόγησης παρά hopcount όπως μετρικές radio-aware. Ένα καινούργιο μονοπάτι τομέα μετρικής περιλαμβάνεται στα μηνύματα RREQ/RREP που περιέχουν την συνολική αξία των μετρητών ζεύξης του μονοπατιού μέχρι τώρα. Ο προτερόθετος μετρητής δρομολόγησης του HWMP είναι το airtime metric όπου οι ξεχωριστές μετρικές ζεύξης προσθέτονται για να πάρουν το μονόπατη μετρικής.

Καθώς μια μετρική radio-aware αλλάζει πιο συχνά από την μετρική hopcount, είναι προτιμότερο να έχουμε μόνο τον προορισμό για να απαντήσουμε σε ένα RREQ έτσι ώστε το μονοπάτι της μετρικής να είναι ενημερωμένο. Για αυτόν τον λόγο ο προορισμός only flag ρυθμίζεται (DO=1) από προτερόθετο HWMP.

Ρυθμίζοντας με ακρίβεια τον προορισμό only flag στο DO=0, είναι πιθανόν να επιτρέψει τους μεσαίους κόμβους να απαντήσουν. Αυτό δίνει μια πιο σύντομη λανθάνουσα καθυστέρηση της ανακάλυψης της διαδρομής, αλλά η διαδρομή μετρικής δεν είναι ενημερωμένη.

Οπότε,ο μεσαίος κόμβος που απάντησε με ένα RREP θα προωθήσει το RREQ στον προορισμό. Αυτό ελέγχεται από την απάντηση και την flag προώθησης. Ρυθμίζεται από την προτερόθετη (RF=1),αλλά μπορεί να ξερυθμιστεί για να έχει την παραδοσιακή συμπεριφορά AODV.

Ο προορισμός only flag στο προωθημένο RREQ πρέπει να είναι ρυθμισμένο σε (DO=1). Αυτό εμποδίζει περισσότερους μεσαίους κόμβους από το να παράγουν απαντήσεις διαδρομής οι οποίες μπορεί να είναι πολλές.

Οποιοσδήποτε πληροφορίες που έχουν ληφθεί (RREQ/RREP) ελέγχονται για εγκυρότητα με μια σειρά αριθμών ακολουθίας σύγκρισης. Οι πληροφορίες δρομολόγησης είναι έγκυρες εάν ο αριθμός ακολουθίας δεν είναι μικρότερος από τον αριθμό ακολουθίας στις προηγούμενες πληροφορίες.

Εάν ο αριθμός ακολουθίας είναι ίδιος και οι πληροφορίες δρομολόγησης,οι οποίες είναι το μονοπάτι μετρικής,είναι καλύτερες,τότε οι καινούργιες πληροφορίες θα χρησιμοποιηθούν και το καινούργιο μήνυμα θα επεξεργαστεί. Το HWMP μπορεί να χρησιμοποιήσει για περιοδική συντήρηση RREQs για να συντηρήσει μια καλύτερη διαδρομή μετρικής μεταξύ στην πηγή και στον προορισμό των ενεργών διαδρόμων. Αυτό είναι ένα προαιρετικό χαρακτηριστικό.

Το HWMP επιτρέπει πολλαπλούς προορισμούς στα μηνύματα RREQ,το οποίο μειώνει την επιβαρυνόμενη δρομολόγηση όταν ένας κόμβος πλέγματος πρέπει να βρει διαδρομές σε αρκετούς κόμβους ταυτόχρονα. Το ίδιο συμβαίνει για την επισκευή διακεκομμένης ζεύξης και για την συντήρηση των RREQs. Μερικές flags μπορεί να έχουν διαφορετικές τιμές για τον κάθε προορισμό. Οπότε,οι flags ανά προορισμό σχετίζονται με τον κάθε προορισμό και τον κάθε αριθμό ακολουθίας. Αυτές είναι οι flags που σχετίζονται συγκεκριμένα με την γενιά των μηνυμάτων απάντησης διαδρομής.

Ένας σαφώς τομέας time to live (TTL) είναι απαραίτητος καθώς δεν υπάρχει καθόλου στην επικεφαλίδα όπως στο παραδοσιακό AODV. Η χρήση της proactive επέκτασης στο RM-AODV διαμορφώνεται. Η proactive επέκταση χρησιμοποιεί την ίδια μεθοδολογία distance vector όπως στο RM-AODV και χρησιμοποιεί τα μηνύματα δρομολόγησης του RM-AODV.

Για να χρησιμοποιήσεις την proactive επέκταση,τουλάχιστον ένα mesh portal πρέπει να διαμορφωθεί για να εκπέμπει περιοδικά mesh portal ανακοινώσεις. Αυτό ενεργοποιεί μια επιλογή διαδρομής και τη διεργασία διαιτησίας,εκ των οποίων ένα μόνο root portal εξελίσσεται. Το root portal ρυθμίζει το τύπο αναγγελίας flag στο 1 (root) στην περιοδική της mesh portal ανακοινώσεις.

Όταν ληφθεί μια τέτοια αναγγελία root portal,ένας mesh node θα στήσει μια διαδρομή στο root portal μέσα από τον mesh node που έλαβε την root portal αναγγελία με την καλύτερη μετρική μονοπατιού. Ένα μονοπάτι που ανακοινώνει mesh portal μπορεί επίσης να ρυθμιστεί για την παραλαβή της portal ανακοίνωσης με της ανακοίνωση flag ρυθμισμένη στο 0 (portal).Το στήσιμο της διαδρομής θα οδηγήσει σε ένα δέντρο rooted στο (mesh) portal.

Εάν η flag εγγραφή δεν είναι ρυθμισμένη στην ανακοίνωση μηνύματος (non-registration mode),η επεξεργασία των root ανακοινώσεων σταματάει εδώ. Όταν ένας mesh node θέλει να στείλει πλαίσιο δεδομένων στο root portal,μπορεί να στείλει ένα χαριστικό RREP στο root portal αμέσως πριν από το πρώτο πακέτο δεδομένων. Αυτό θα στήσει την προς τα πίσω διαδρομή από το root portal έως τον κόμβο πηγής.

Εάν η flag εγγραφή είναι ρυθμισμένη στην ανακοίνωση μηνυμάτων (registration mode),ο κόμβος πλέγματος περιμένει ένα ορισμένο διάστημα για περισσότερα μηνύματα ανακοινώσεων root να φτάσουν ή μπορεί επίσης να εκδώσει ένα RREQ με TTL=1 για να ζητήσει ξεκάθαρα από τους γειτονικούς κόμβους διαδρομές για την root portal. Ο κόμβος πλέγματος επιλέγει την διαδρομή με την καλύτερη μετρική διαδρομής μέχρι την root portal. Η εγγραφή πρέπει να γίνει κάθε φορά που ο κόμβος αλλάζει το γονικό κόμβο του.

2.6.3 Radio Aware Optimized Link State Routing (RA-OLSR)

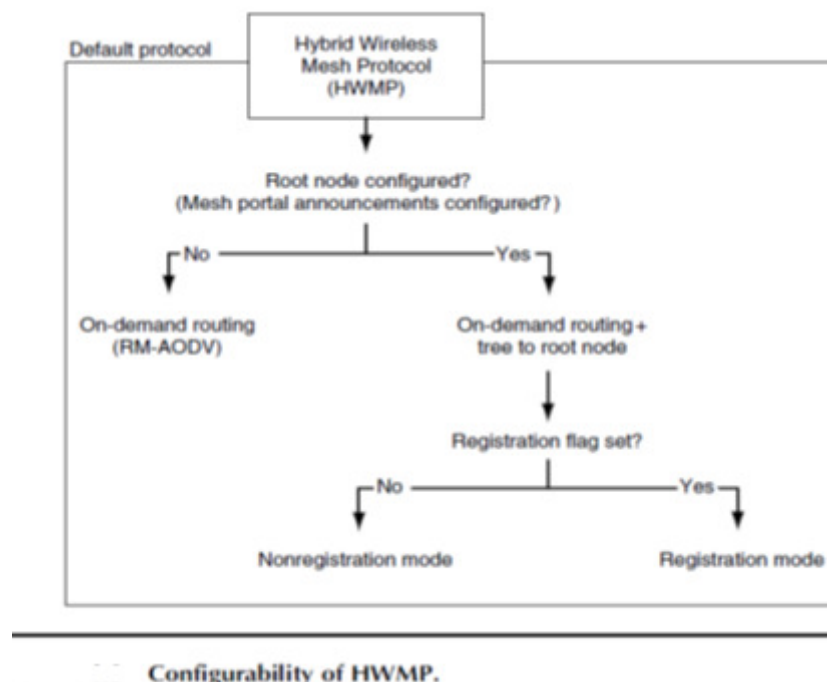
Το RA-OLSR πρωτόκολλο είναι ένα προαιρετικό, proactive πρωτόκολλο δρομολόγησης του αναδυόμενου προτύπου IEEE 802.11s. Ακολουθεί στενά την διευκρίνηση του OLSR πρωτοκόλλου. Αντί για IP διευθύνσεις χρησιμοποιεί MAC διευθύνσεις και μπορεί να λειτουργήσει με αυθαίρετους μετρητές δρομολόγησης όπως της air-time μετρικής.

Επιπλέον, ορίζει έναν μηχανισμό για την διανομή διευθύνσεων των μη πλέγματος πελατών WLAN στο RA-OLSR mesh. Η κατάσταση ζεύξης είναι η τιμή της μετρικής ζεύξης και χρησιμοποιείται στο ποιο κοντινό υπολογισμό μονοπατιού.

Οπότε, ένας τομέας μετρικής ζεύξης σχετίζεται με το κάθε αναφερόμενο γείτονα στα hello μηνύματα και TC μηνύματα. Η τιμή της μετρικής ζεύξης επίσης αποθηκεύεται στις αντίστοιχες αποθήκες πληροφοριών, δηλ. το σύνολο ζεύξεων και το σύνολο τοπολογίας.

Η μετρική ζεύξης χρησιμοποιείται επίσης στην εύρεση για την επιλογή των multipoint αναμεταδόσεων. Το κάθε σημείο πρόσβασης πλέγματος διατηρεί μια local association base (LAB) που περιέχει όλους τους σταθμούς IEEE 802.11 κληρονομιάς που σχετίζονται με αυτό το πλεγματο AP. Αυτό εκπέμπει local association base advertisement (LABA) μηνύματα περιοδικά, προκειμένου να διανεμίει τα στοιχεία σύνδεσης στο δίκτυο πλέγματος.

Οι πληροφορίες που λαμβάνονται από τα LABA μηνύματα αποθηκεύονται στην global association base (GAB) στον κάθε κόμβο.



Οι πληροφορίες του LABA και του GAB χρησιμοποιούνται στην κατασκευή του πίνακα δρομολόγησης και παρέχει πορείες σε σταθμούς κληρονομιάς που σχετίζονται με τα σημεία πρόσβασης πλέγματος.

Για να εξοικονομήσουμε το bandwidth, είναι πιθανό να προκηρύσσουμε δημοσίως μόνο το άθροισμα ελέγχου των τμημάτων του LAB. Εάν υπάρχει αναντιστοιχία μεταξύ του

αθροίσματος έλαβε και το checksum στο GAB,ο κόμβος ζητά μια ενημέρωση του αντίστοιχου τμήματος του LAB του αρχικού κόμβου.

Το RA-OLSR επίσης χρησιμοποιεί τον έλεγχο συχνότητας για κατάσταση ζεύξης υπερχειλίσης όπως είναι γνωστό από το Fisheye State Routing για την διανομή των μηνυμάτων τοπολογίας έλεγχου.

Η ιδέα είναι ότι οι πιο κοντινοί κόμβοι πλέγματος λαμβάνουν τοπολογικές πληροφορίες πιο συχνά από τους απομακρυσμένους κόμβους. Οπότε,το TTL στα μηνύματα έλεγχου τοπολογίας ρυθμίζεται στο 2, 4, και μέγιστο TTL διαδοχικά.

2.6.4 Επεκτασιμότητα

Ένα από τα συμπεράσματα από έρευνα στην κινητή ad hoc δικτύωση είναι ότι δεν θα υπάρχει ένα πρωτόκολλο δρομολόγησης που θα είναι βέλτιστο σε κάθε χρήσιμο σενάριο.

Οπότε,ένας μηχανισμός επέκτασης προτείνεται,ο οποίος επιτρέπει ευλυγισία,αλλά ακόμα παρέχει διαλειτουργικότητα ανάμεσα στους κόμβους πλέγματος διαφορετικών πωλητών.

Το κάθε δίκτυο πλέγματος ανακοινώνει το δικό του χρησιμοποιημένο πρωτόκολλο δρομολόγησης στους καινούργιους κόμβους με αντίστοιχα IDs. Μόνο εκείνοι οι κόμβοι υποστηρίζουν το χρησιμοποιημένο πρωτόκολλο δρομολόγησης και μετρική δρομολόγησης επιτρέπεται να ενταχθούν στο δίκτυο πλέγματος.

Όλες οι συσκευές IEEE 802.11s θα μπορέσουν να χρησιμοποιήσουν το προτερόθετο πρωτόκολλο δρομολόγησης και τον προτερόθετο μετρητή δρομολόγησης. Άλλα πρωτόκολλα δρομολόγησης ή μετρικές δρομολόγησης,τα οποία είναι πιο κατάλληλα για μερικά σενάρια,μπορούν να χρησιμοποιηθούν με τα προτερόθετα.

2.7 Κοινή δρομολόγηση και εκχώρηση καναλιού

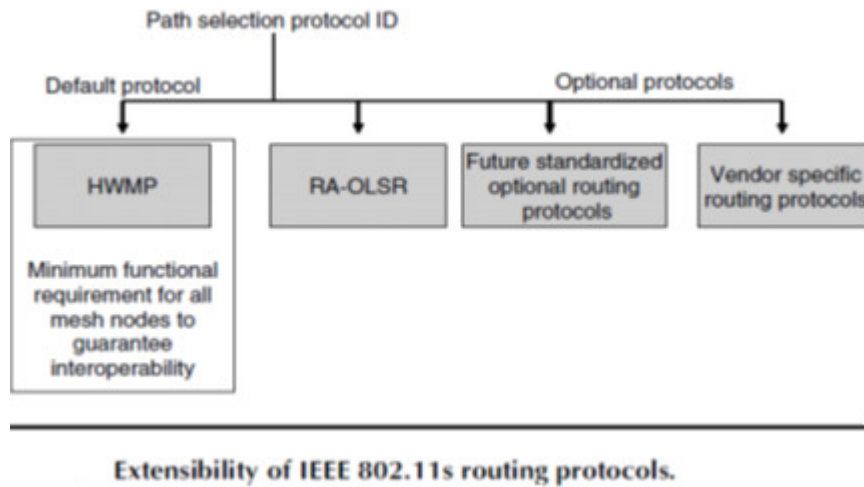
Σε ένα multiradio, multichannel WMN,εκτός από δρομολόγηση,άλλο ένα κύριο πρόβλημα σχεδιασμού είναι να αναθέσεις ένα radio channel σε κάθε δικτυακή διεπαφή για να αποκτήσεις αποτελεσματική χρήση των διαθέσιμων καναλιών,η λεγόμενη ανάθεση καναλιού.

Φιλολογικά,υπάρχουν πολυάριθμες μελέτες που λαμβάνουν υπόψιν τους την ανάθεση δρομολόγησης και καναλιού. Μερικές λαμβάνουν υπόψιν τους την μόνο την δρομολόγηση για να μεγιστοποιήσουν το throughput. Μερικές λαμβάνουν υπόψιν τους μόνο την ανάθεση καναλιού για να ελαχιστοποιήσουν τις παρεμβολές. Μερικές μελετούν και τις δυο αναθέσεις δρομολόγησης και καναλιού.

Οι Raniwala,και Chiueh λαμβάνουν υπόψιν τους την δρομολόγηση πρώτα και στην συνέχεια ανάθεση καναλιού. Καθώς η ανάθεση καναλιού γίνεται μετά από την δρομολόγηση,δίνεται το φορτίο στην κάθε ζεύξη.

Οπότε,μια τέτοια ανάθεση καναλιού ονομάζεται επίσης load-aware ανάθεσης καναλιού. Tang et al. Λαμβάνουν υπόψιν τους την ανάθεση καναλιού πρώτα,και στην συνέχεια την δρομολόγηση. Όταν δοθεί μια ανάθεση καναλιού,η παρεμβολή ανάμεσα στους συνδέσμους

καθορίζεται,οπότε η δρομολόγηση κάτω από ένα τέτοιο περιορισμό ονομάζεται επίσης interference-aware δρομολόγησης.



Όμως σε ένα multiradio multichannel WMN,η ανάθεση δρομολόγησης και καναλιού είναι ανεξάρτητα. Η δρομολόγηση εξαρτάται από την ικανότητα της εικονικής ζεύξης η οποία καθορίζεται από την ανάθεση καναλιού.

Από την άλλη πλευρά,η ανάθεση καναλιού εξαρτάται από το αναμενόμενο φόρτο της εικονικής ζεύξης,το οποίο επηρεάζεται από την δρομολόγηση. Εξαιτίας της κυκλικής εξάρτησης ανάμεσα στην ανάθεση δρομολόγησης και καναλιού,ένας κοινός μηχανισμός ανάθεσης δρομολόγησης και καναλιού για να μεγιστοποιηθεί το δικτυακό throughput είναι επιθυμητή.

Εδώ,πρώτα παρουσιάζουμε το συνδυασμένο load-aware δρομολόγηση και ανάθεση καναλιού μηχανισμό,το οποίο επαναλαμβάνει πάνω από την ανάθεση καναλιού και αλγορίθμους δρομολόγησης μέχρι η ικανότητα της κάθε ζεύξης να αντιστοιχεί με το αναμενόμενο φόρτο του όσο περισσότερο γίνεται. Τότε παρουσιάζουμε την κοινή ανάθεση καναλιού και δρομολόγησης.

Εξαιτίας της δυσκολίας της κοινής ανάθεσης καναλιού και δρομολόγησης και οι δυο εργασίες λύνουν το γραμμικό προγραμματισμό (LP) χαλάρωση του προβλήματος βέλτιστα,μετά κάνουν περισσότερη επεξεργασία για να έχουν μια εφικτή ανάθεση καναλιού.

2.7.1 Συνδυασμένη Load-aware δρομολόγησης και εκχώρησης καναλιού

Η combined load-aware routing and channel assignment ξεκινάει με έναν αρχικό υπολογισμό του αναμενόμενου φορτίου σε κάθε εικονική ζεύξη άσχετα με την χωρητικότητα της ζεύξης, μετά επαναλαμβάνει πάνω στην ανάθεση καναλιού και το αλγόριθμο δρομολόγησης μέχρι να είναι εφικτή η ανάθεση καναλιού.

Οι εισαγωγές στην συνδυασμένη ανάθεση καναλιού και αλγόριθμο δρομολόγησης περιλαμβάνουν έναν υπολογισμένου φόρτο κίνησης μετρητή, μια WMN τοπολογία, και τα

διαθέσιμα radios στον κάθε κόμβο καθώς και ο αριθμός των radios καναλιών που δεν συμπίπτουν το ένα με το άλλο. Όταν σταματάει ο αλγόριθμος, η κάθε διεπαφή 802.11 δένεται με ένα κανάλι και το κάθε ζευγάρι κόμβου επικοινωνίας έχει μια διαδρομή στα WMNs.

Οι λεπτομέρειες της συνδυασμένης load-aware δρομολόγησης και καναλιού ανάθεσης παρουσιάζονται:

(1) Βήμα : Ο αλγόριθμος δρομολόγησης υπολογίζει τις αρχικές πορείες για το κάθε ζευγάρι κόμβων που δίνεται ένα σύνολο από ζευγάρια κόμβων και το αναμενόμενο φορτίο κίνησης ανάμεσα στο κάθε ζευγάρι κόμβων.

(2) Βήμα : Δίνοντας την εισαγωγή από το βήμα δρομολόγησης, το radio κανάλι ανάθεσης αλγορίθμου αναθέτει ένα radio κανάλι στην κάθε διεπαφή έτσι ώστε το διαθέσιμο bandwidth σε κάθε εικονική ζεύξη δεν είναι λιγότερο από το αναμενόμενο φορτίο του.

(3) Βήμα : Το καινούργιο κανάλι ανάθεσης επανεισάγεται στον αλγόριθμο δρομολόγησης για να πάρει περισσότερες πληροφορίες δρομολόγησης.

(4) Βήμα : Επανυπολογίζουν το φορτίο ζεύξης με βάση τις πληροφορίες δρομολόγησης. Εάν μερικά από τα φορτία ζεύξης είναι περισσότερα από τις ικανότητες τους, πήγαινε στο Βήμα 2, αλλιώς, Σταμάτα.

Ο παραπάνω συνδυασμένος μηχανισμός μπορεί να χρησιμοποιηθεί ως μια γενική προσέγγιση για να συνδέσει οποιαδήποτε ανάθεση καναλιού και λύσεις δρομολόγησης μαζί για να αποκτήσουμε καλύτερο δικτυακό throughput.

Για παράδειγμα, στο Βήμα 2, οποιαδήποτε load-aware ανάθεση καναλιού μπορούν να εφαρμοστούν. Στο βήμα 3, οποιαδήποτε interference-aware αλγορίθμοι δρομολόγησης μπορούν να εφαρμοστούν.

2.7.2 Κοινή LP-based δρομολόγηση και εκχώρηση καναλιού

Ο Kodialam, Nandagopal και Alicherry et al., προτείνουν τα LP βασισμένα στην κοινή δρομολόγηση και ανάθεση καναλιού. Υποθέτουν ότι υπάρχει πιθανώς ζήτηση για κίνηση για οποιαδήποτε ζευγάρι κόμβων, υποθέτουν ότι η επικοινωνία συμμετέχει μόνο προς και από τις καλωδιακές πύλες, παρά από το να περιλαμβάνει ζευγάρια από κόμβους τέρματος.

Σε ένα WMN, οι ασύρματοι χρήστες ενδιαφέρονται κυρίως να συνδεθούν στο Internet, οπότε, ένα ασυμμετρικό σχέδιο κίνησης είναι λογικό και πράγματι απλοποιεί το LP μοντέλο.

2.7.3 Μοντέλα και παραδοχές

Μπορούμε να περιγράψουμε ένα WMN από ένα γράφημα $G=O(N, E)$ όπου το N αντιπροσωπεύει μια σειρά από κόμβους, το καθένα εξοπλίζεται με multiple wireless radios και E αντιπροσωπεύει ζεύξεις απευθείας επικοινωνίας ανάμεσα σε ένα ζευγάρι από κόμβους. Υπάρχει ένας gateway κόμβος $u_0 \in N$ που είναι συνδεδεμένο με το Internet, και η κίνηση από οποιοδήποτε άλλο κόμβο στο Internet κατευθύνεται μέσα από το u_0 .

Πρακτικά μπορεί να υπάρχουν multiple gateway κόμβοι στο σύστημα, και τα αποτελέσματα μας μπορούν εύκολα να γενικευτούν σε μια τέτοια περίπτωση. Για το στόχο της απλότητας, υποθέτουμε ότι υπάρχει μόνο ένας μοναδικός gateway κόμβος.

Ένας κόμβος μπορεί να επικοινωνήσει με κάποιους άλλους κόμβους μέσω ασυρμάτων ζεύξεων επικοινωνίας. Πιο συγκεκριμένα, υπάρχει μια ζεύξη επικοινωνίας $e=(u, v) \in E$ μόνο εάν οι κόμβοι u, v είναι μέσα στην έγκυρη εμβέλεια μετάδοσης ενός radio. Ας υποθέσουμε ότι υπάρχουν K ορθογωνικά ασύρματα κανάλια, και το bandwidth του κάθε καναλιού είναι c . Υποθέτουμε ότι το σύστημα λειτουργεί μέσω περιοδικό συγχρονισμένο χρονικό τρόπο, όπου ο κάθε κύκλος περιεχί T χρονοθυρίδες.

Προφανώς, το μοντέλο που παρουσιάζουμε εδώ περιεχί μια άνοδο για ένα σύστημα σε ασύγχρονη λειτουργία. Για τα προβλήματα στην ανάθεση δρομολόγησης και καναλιού, υπάρχουν τρεις αποφάσεις που μπορούμε να πάρουμε: Να αναθέσουμε μια σειρά ασυρμάτων καναλιών στην κάθε ζεύξη $e \in E$, να καθορίσουμε εάν το κάθε (κανάλι, ζεύξη) ζευγάρι είναι ενεργό για την κάθε χρονοθυρίδα $\tau = 1, \dots, T$, και να αναθέσουμε την επικοινωνία κίνησης στα ενεργά (κανάλι, ζεύξη) ζευγάρια σε διαφορετικές χρονοθυρίδες.

Το αντικείμενο της ανάθεσης δρομολόγησης και καναλιού είναι να μεγιστοποιήσουμε το throughput του συστήματος με τις παραπάνω αποφάσεις. Πιο συγκεκριμένα υποθέτουμε ότι ο κάθε κόμβος $u \in N$ έχει μια κατά μέσο όρο ζήτηση d_u που χρειάζεται να δρομολογηθεί στον gateway κόμβο u_0 και θέλουμε να εξυπηρετήσουμε αυτήν την κίνηση όσο μπορούμε.

Η προσέγγιση περιεχί 2 βήματα: Η interference-aware δρομολόγηση που ακολουθείται από μετά-επεξεργασία της ανάθεσης καναλιού που προσαρμόζεται.

Πρώτα λύνουμε ένα LP (γραμμικό προγραμματισμό) που καθορίζει μια λύση δρομολόγησης λαμβάνοντας υπόψιν το φορτίο κίνησης και την επιρροή της παρεμβολής. Μια τέτοια λύση επίσης παρέχει μια ανάθεση καναλιού που μπορεί να μην είναι εφικτή εξαιτίας της χαλάρωσης στο LP (γραμμικό προγραμματισμό).

Δεύτερον, στην μετά-επεξεργασία, προσαρμόζουμε την ανάθεση καναλιού για να αποκτήσουμε μια εφικτή λύση. Κάνοντας έτσι, κρατάμε την λύση δρομολόγησης έτσι ώστε το throughput κίνησης δεν θα μειωθεί.

2.8 Προοπτικές και ανοικτά ζητήματα

Η ασύρματη πλεγματοκή δικτύωση είναι ένα θέμα που τώρα προσελκύει μεγαλύτερη προσοχή από βιομηχανικές εταιρείες και πανεπιστήμια. Μεγάλες προσπάθειες γίνονται για να αναπτύξουν λειτουργικά πλεγματοκή πρωτότυπα. Υπάρχουν πολλές εταιρείες που πωλούν ασύρματες συσκευές πλέγματος. Και ήδη υπάρχουν πολλές εγκαταστάσεις σε λειτουργία ασύρματων δικτύων πλέγματος σε όλο τον κόσμο.

Η κυρία ιδέα των ασυρμάτων δικτύων πλέγματος-προώθησης πακέτων πάνω σε multiple wireless hops-είναι μια καινούργια ποιότητα στην ασύρματη επικοινωνία. Μπορεί να φτιάξει συσκευές 'πραγματικά ασύρματες'. Τα απαραίτητα πρωτόκολλα δρομολόγησης έχουν μια γερή βάση, η οποία είναι ένα καλό θεμέλιο για την συνεχή ανάπτυξη του αριθμού των αναπτύξεων των WMNs.

Εξαιτίας της μεγάλης ευλυγισίας και της δύναμης τους, τα ασύρματα δίκτυα πλέγματος θα είναι ένα σημαντικό μέρος των μελλοντικών(ασυρμάτων)δικτυακών αρχιτεκτονιών. Παρόλο που ο τομέας της ασύρματης δικτύωσης πλέγματος μπορεί να αναπτυχθεί με τη μεγάλη ποσότητα αποτελεσμάτων από μια δεκαετία της έρευνας στην κινητή δικτύωση ad hoc, υπάρχουν ακόμα πολλά ανοιχτά θέματα έρευνας.

Οι ειδικές ιδιότητες των WMNs απαιτούν και επιτρέπουν βελτιστοποιήσεις για να ανταποκριθεί με τους στόχους απόδοσης για την χρήση ασυρμάτων δικτύων πλέγματος. Το κάθε σενάριο εφαρμογής μπορεί να χρειαστεί διαφορετική βελτιστοποίηση. Υψηλό δικτυακό throughput και χωρητικότητα δικτύου είναι οι σημαντικότερες απαιτήσεις στις πρακτικές εφαρμογές.

Καινούργιες μετρικές δρομολόγησης πρέπει να αναπτυχθούν και να χρησιμοποιηθούν για να υποστηρίξουν τις απαραίτητες βελτιώσεις. Η κινητικότητα όταν οι client συσκευές ενσωματώνονται στα ασύρματα δίκτυα πλέγματος.

Καλύτερες και πιο ισχυρές συσκευές μπορούν να έχουν multiple radio διεπαφές και μπορούν να χρησιμοποιήσουν την ποικιλότητα καναλιού. Αυτό πρέπει να υποστηριχθεί από πρωτόκολλα δρομολόγησης και μετρικές δρομολόγησης.

Τελευταίο αλλά όχι ασήμαντο, σχέδιο cross-layer είναι σημαντικό για να έχουμε καλύτερη πρόσβαση στα επίπεδα που έχουν μεγάλη επιρροή στην δρομολόγηση MAC και φυσικό επίπεδο.

ΚΕΦΑΛΑΙΟ 3

ΤΟ ΠΡΟΤΖΕΚΤ ΤΗΣ ΠΤΥΧΙΑΚΗΣ

3.1 NS2 versus NS3

Το πρότζεκτ της πτυχιακής στηρίχθηκε στον προσομοιωτή NS3 διότι είναι ένας open-source και εξελισσόμενος προσομοιωτής Το project NS3 ξεκίνησε το 2006. Ο NS3 έχει αναπτυχθεί για να παρέχει ένα ανοιχτό, επεκτάσιμο δίκτυο πλατφόρμας προσομοίωσης, για την δικτύωση την έρευνα και εκπαίδευση. Επίσης υλοποιείται με την ευρέως διαδεδομένη γλώσσα προγραμματισμού C++ και python. Είναι πιο εύχρηστος απο τον NS2.

Οι χρήστες θα παρατηρήσουν ότι το διαθέσιμο πρότυπο που παρατίθεται στο NS3 εστιάζει στην μοντελοποίηση των πρωτοκόλλων του Διαδικτύου και των δικτύων εργασίας, αλλά ο NS3 δεν περιορίζεται σε συστήματα Internet. Οι χρήστες χρησιμοποιούν NS3 για τη μοντελοποίηση των συστημάτων non-Internet-based.

Πρωτίστως χρησιμοποιείται σε συστήματα Linux,αν και υποστηρίζεται απο FreeBSD, Cygwin (for Windows),και Windows Visual Studio είναι ακόμα σε πορεία ανάπτυξης. Ο NS3 δεν υποστηρίζεται επίσημα ως προϊόν λογισμικού της κάθε εταιρείας.

	NS2	NS3
Programming languages	<ul style="list-style-type: none"> • NS2 εφαρμόζεται χρησιμοποιώντας TCL (περιγράφουν την τοπολογία του δικτύου) and C++ (πυρήνα του προσομοιωτή). • Αυτό το σύστημα επιλέχθηκε στις αρχές του 1990 για να αποφευχθεί η επαναμεταγλώττιση of C++ δεδομένου ότι ήταν πολύ χρονοβόρα χρήση των διαθέσιμων εκείνη τη στιγμή hardware ,TCL αναμεταγλώττιση διαρκεί λιγότερο χρόνο από ό, τι C++. • TCL μειονέκτημα: δεν υπάρχουν γενικά εισάγεται με μεγάλες προσομοιώσεις. • TCL είναι η μόνη διαθέσιμη scripting γλώσσα. 	<ul style="list-style-type: none"> • NS3 εφαρμόζεται χρησιμοποιώντας C++ • Με τις σύγχρονες δυνατότητες του hardware, ρόνος μεταγλώττισης δεν ήταν ζήτημα, όπως για τον NS2, NS3 μπορούν να αναπτυχθούν με C ++ εντελώς. • Ένα σενάριο προσομοίωσης μπορεί να γραφτεί ως ένα πρόγραμμα C ++, η οποία δεν είναι δυνατόν σε NS2. • Υπάρχει περιορισμένη υποστήριξη για την Python στο scripting και οπτικοποίησης.
Memory Management	<ul style="list-style-type: none"> • NS2 απαιτεί βασικό εγχειρίδιο της C ++ λειτουργίες διαχείρισης μνήμης. 	<ul style="list-style-type: none"> • Επειδή NS3 υλοποιείται σε C++,όλες οι συνήθειες C++ διαχείριση μνήμης λειτουργίας, όπως η new, delete, malloc,και δωρεάν είναι ακόμα διαθέσιμα. • Αυτόματη ανακατανομή των αντικειμένων υποστηρίζεται χρησιμοποιώντας μέτρηση αναφοράς (αριθμός κομματιού δείκτες σε μια αντικείμενο) Αυτό είναι

		χρήσιμο όταν ασχολείται με Packet αντικείμενα.
Packets	<ul style="list-style-type: none"> • Ένα πακέτο αποτελείται από 2 διαφορετικές περιοχές ένα για τις κεφαλίδες, και το δεύτερο αποθηκεύει στοιχεία ωφέλιμου φορτίου • NS2 ποτέ δεν ελευθερώνει μνήμη χρησιμοποιείται για την αποθήκευση πακέτων μέχρι η προσομοίωση να τερματιστεί, απλά επαναχρησιμοποιεί το καταμετρηθεί πακέτα επανειλημμένα, ως αποτέλεσμα, η επικεφαλίδα περιοχή του κάθε πακέτου περιλαμβάνει όλες τις επικεφαλίδες που ορίζονται ως μέρος του χρησιμοποιημένου πρωτόκολλο ακόμη και αν το συγκεκριμένο πακέτο δεν θα χρησιμοποιήσει το συγκεκριμένο επικεφαλίδα, αλλά μόνο και μόνο για να είναι διαθέσιμα όταν αυτό κατανομή του πακέτου επαναχρησιμοποιείται. 	<ul style="list-style-type: none"> • Ένα πακέτο αποτελείται από ένα buffer bytes και προαιρετικά μια συλλογή μικρές ετικέτες που περιέχουν meta-data. • Το buffer αντιστοιχεί ακριβώς στο ρεύμα των bits που θα αποστέλλονται μέσω ενός πραγματικού δικτύου. • Πληροφορίες προστίθενται στο πακέτο με χρησιμοποιώντας subclasses, Header, η οποία προσθέτει πληροφορίες για την έναρξη του buffer, ουρά, το οποίο προσθέτει στο τέλος. • Σε αντίθεση με NS2, υπάρχει γενικά εύκολος τρόπος για να προσδιοριστεί εάν μια συγκεκριμένη επικεφαλίδα επισυνάπτεται.
Performance	<ul style="list-style-type: none"> • Ο συνολικός χρόνος υπολογισμού που απαιτείται για την εκτέλεση μιας προσομοίωσης μεταξύ NS2 και NS3, στον NS3 εκτελείται πιο γρήγορα. • Αυτό οφείλεται στην απομάκρυνση του γενικά συνδέεται με τη διεπαφή Tcl με C ++, και η επιβάρυνση που σχετίζονται με τον διερμηνευτή Tcl. 	<ul style="list-style-type: none"> • NS3 αποδίδει καλύτερα από ό, τι σε NS2 όσον αφορά τη διαχείριση της μνήμης. • Το σύστημα συνάθροισης αποτρέπει περιττούς παραμέτρους από το να αποθηκεύονται, και τα πακέτα δεν περιέχουν αχρησιμοποίητο χώρο που προορίζεται για την επικεφαλίδα.
Simulation output	<ul style="list-style-type: none"> • NS2 έρχεται με ένα πακέτο που ονομάζεται NAM (Network Animator), είναι ένα Tcl με βάση το σύστημα animation που παράγει μια οπτική αναπαράσταση του δικτύου όπως περιγράφεται. 	<ul style="list-style-type: none"> • NS3 χρησιμοποιεί ένα πακέτο γνωστό ως PyViz, το οποίο είναι ένα πακέτο σε πραγματικό χρόνο απεικόνισης με βάση την Python

3.2 AODV versus HWMP

Characteristic	AODV	HWMP
	On-demand Υπερχειλιση Mobile nodes Hello packets No loops Ttl No counting to infinity Time slots Hop count	
Update information	PERR	PREQ
Update destination	Source	Root
Method	Unicast	Tree broadcast
Update period	Even driven	Periodically
Stored information	Next hop for destination	Next hop for root
Structure	Flat	Tree
Route computation	reactive	Proactive
Routing metric	Alm	Alm

Το πρωτόκολλο HWMP βασίζεται στο πρωτόκολλο AODV αλλά χρησιμοποιείται στο επίπεδο 2 σε αντίθεση με το επίπεδο 3 που χρησιμοποιείται ο AODV για την δρομολόγηση. Έτσι, ένα ερώτημα που γεννάται είναι αν θα μπορούσαμε να γνωρίζουμε αν αυτό το νέο επίπεδο 2 δρομολόγησης που προτείνονται για τα MESH δίκτυα θα λειτουργήσει καλύτερα από ό, τι το παραδοσιακό επίπεδο 3 δρομολόγησης.

Για να απαντηθεί αυτό το ερώτημα αυτά τα δύο πρωτόκολλα δρομολόγησης ρυθμίστηκαν να δουλέψουν κάτω από τις ίδιες συνθήκες και να αξιολογηθούν οι επιδόσεις τους όσον αφορά την επεκτασιμότητα(κόμβοι και σύνδεση).

3.3 Υλοποίηση πρότζεκτ

Σε αυτή την ενότητα γίνεται μία περιγραφή της υλοποίησης της εφαρμογής. Συγκεκριμένα θα γίνει μία περιγραφή της λειτουργίας των συναρτήσεων καθώς και της μεταξύ τους αλληλεπίδρασης.

Το δίκτυό μας αποτελείται από ένα σύνολο 64 κόμβων 8*8 και στο οποίο μπορούν να συνδεθούν οι πελάτες-συσκευές για την πρόσβασή τους στο διαδίκτυο αλλά και την ανταλλαγή διαφόρων άλλων πολυμεσικών εφαρμογών όπως βίντεο,φωνής.

Ο gateway είναι ένας συγκεκριμένος κόμβος απο τους σταθερούς κόμβους. Οι πελάτες υποθέτουμε πως θέλουν να στείλουν πακέτα προς όλο το διαδίκτυο. Για το λόγο αυτό επικοινωνούν με τον gateway. Κάθε φορά ως gateway επιλέγετε ο σταθερός κόμβος που έχει δημιουργηθεί τελευταίος. Αυτός βρίσκεται στη γωνία του πλέγματος..

Η επιλογή της κεραίας στην οποία θα συνθεθούν οι κόμβοι γίνεται με βάση τα πρωτόκολλα αντίστοιχα. Στην περίπτωση που έχουμε το AODV κάθε φορά γίνεται μία διαδικασία broadcast. Όταν φτάσει το αρχικό μήνυμα στον προορισμό τότε ακολουθούνται τα μηνύματα ανάποδα για να δημιουργηθεί η διαδρομή. Οπότε κάθε φορά δημιουργείται εκ νέου το μονοπάτι και ο κόμβος συνδέεται όπου θα έχει γρηγορότερη ανταπόκριση. Στην περίπτωση του HWMP η διερεύνηση δεν γίνεται κάθε φορά αλλά διατηρείται ένας πίνακας δρομολόγησης παρόμοιος με αυτό του Distance Vector. Με αυτό τον τρόπο δεν επιλέγεται η ταχύτερη διαδρομή γιατί μπορεί ο κόμβος να έχει μετακινηθεί και να υπάρχει καλύτερη διαδρομή. Όμως αποφεύγονται οι συνεχείς αποστολές μηνυμάτων για την ανίχνευση της διαδρομής.

Η αποστολή των πακέτων sink γίνεται προς τον κόμβο gateway. Αυτό γίνεται ώστε να προσομοιωθεί η κίνηση προς το διαδίκτυο που θα έχουν αυτοί οι κόμβοι.

Η αποστολή δεδομένων γίνεται από όλους τους πελάτες ώστε να δημιουργείται κίνηση. Αν κάποια δεν έστελναν τότε θα ήταν το ίδιο με το δίκτυα αν λειτουργούσε με λίγότερους κόμβους.

Εμβέλεια-Ενεργειακό μοντέλο

EnergyDetectionThreshold, DoubleValue (-89.0)

CcaMode1Threshold, DoubleValue (-62.0)

TxGain, DoubleValue (1.0)

RxGain, DoubleValue (1.0)

TxPowerLevels, UintegerValue (1)

TxPowerEnd, DoubleValue (18.0)

TxPowerStart, DoubleValue (18.0)

RxNoiseFigure, DoubleValue (7.0)

3.3.1 Υλοποίηση AODV

Κόμβοι	Throughput	packet Delay	Packet Lost	Bandwidth
8	48017,4	1,18191	4047	6

Πλήθος Πελατών	Throughput	packet Delay	Packet Lost
10	37193,5	0,946923	2327
20	65500	0,753281	4499
40	93564,7	0,812501	11021
60	190916	0,921937	15966
80	137138	0,291468	21551
100	127474	0,497624	28952
150	123300	0,148598	45300
200	28580,8	0,0924324	60402

Κλάση AodvExample

Είναι η βασική κλάση του προγράμματος. Αποθηκεύει όλα τα χαρακτηριστικά για την εφαρμογή. Δημιουργεί τους κόμβους με τα απαιτούμενα τεχνικά χαρακτηριστικά και τους τοποθετεί σε κατάλληλες θέσεις. Επιπλέον βάζει τους κινητούς κόμβους να κινούνται στην περιοχή τυχαία. Τέλος, φτιάχνει τις ροές δεδομένων ώστε να γίνουν τα πειράματα.

Συνάρτηση AodvExample::Run ()

Η συγκεκριμένη συνάρτηση τρέχει όλη την εξομοίωση. Αρχικά δημιουργεί τους κόμβους με την `createNodes()`, στη συνέχεια δημιουργεί τις συσκευές πάνω στους κόμβους με την συνάρτηση `CreateDevices()`. Έπειτα, δημιουργεί σε κάθε κόμβο την στοίβα πρωτοκόλλων που θα εκτελεί ο κόμβος και τέλος δημιουργεί τις εφαρμογές που θα τρέξουν με την `InstallApplications`. Ο εξομοιωτής ξεκινάει με την συνάρτηση `Simulator::Run` ενώ πριν του έχουμε πει να σταματήσει σε προκαθορισμένο χρονικό διάστημα.

Η συγκεκριμένη συνάρτηση αφού τελειώσει η προσομοίωση παίρνει από κάθε ροή τα χαρακτηριστικά που μας ενδιαφέρουν. Αυτά είναι το Delay, το throughput, και το packetlost. Αφού υπολογίσει τα αθροιστικά στη συνέχεια γράφει σε αρχεία τους μέσους όρους αυτών.

Συνάρτηση AodvExample::CreateNodes ()

Η συνάρτηση `createNodes`, δημιουργεί τους κόμβους όπως λέει και το όνομά της. Αρχικά φτιάχνει τους σταθερούς κόμβους που είναι τοποθετημένοι πάνω σε ένα πλέγμα. Στη συνέχεια δημιουργεί τους κόμβους που θα είναι κινητοί. Αυτοί κινούνται σε μία περιοχή μέσα στο πλέγμα, με τυχαία αρχική θέση και μετακίνηση μέσα σε αυτό.

Συνάρτηση AodvExample::CreateDevices ()

Η συγκεκριμένη συνάρτηση δημιουργεί τις συσκευές και τις συνδέσεις που θα

έχουν οι κόμβοι. Οι σταθεροί κόμβοι έχουν μία σύνδεση wifi μεταξύ τους και άλλο ένα interface για την επικοινωνία με τους κόμβους πελάτες που κινούνται στην περιοχή. Οι σταθεροί κόμβοι επικοινωνούν μεταξύ τους με 802.11a ενώ η επικοινωνία με τους πελάτες γίνεται με 802.11b. Η εγκατάσταση των διεπαφών στους κόμβους γίνεται αφού έχουν δημιουργηθεί χρησιμοποιώντας τις μεταβλητές που μας επιστρέφθηκαν κατά την δημιουργία.

Συνάρτηση AodvExample::InstallInternetStack ()

Η συγκεκριμένη συνάρτηση βάζει στους κόμβους την στοίβα πρωτοκόλλων που θα εκτελούν. Τόσο οι σταθεροί όσο και οι κινητοί κόμβοι εκτελούν τα ίδια πρωτόκολλα. Επιπλέον, βρίσκονται στο ίδιο υποδίκτυο IP. Έχει επιλεγεί να χρησιμοποιούν ένα εύρος private (ιδιωτικών) διευθύνσεων του δικτύου 10.0.0.0/8

Συνάρτηση AodvExample::InstallApplications ()

Η εν λόγω συνάρτηση φτιάχνει τις εφαρμογές που θα τρέχουν οι κόμβοι του δικτύου. Όλοι οι πελάτες ξεκινάνε να στέλνουν δεδομένα προς ένα κόμβο που θεωρείται ο gateway της υποδομής. Αυτό προσομοιώνει την κίνηση που θα είχαν οι συγκεκριμένοι κόμβοι αν ήθελαν να επικοινωνήσουν με κάποιον κόμβο εκτός του δικτύου. Σε αυτή την περίπτωση όλοι χρησιμοποιούν τον gateway για να περάσει από εκεί η πληροφορία και να επικοινωνήσουν με το διαδίκτυο.

3.3.2 Υλοποίηση HWMP

Κόμβοι	Throughput	packet Delay	Packet Lost	Bandwidth
8	2362,79	0,991892	0	6

Πλήθος Πελατών	Throughput	packet Delay	Packet Lost
10	2658,62	0,600002	0
20	2875,39	0,989545	0
40	1022,45	1,35615	0
60	5271,37	1,89124	0
80	310,936	3,28315	0
100	17,355	3,60057	0
150	14,0174	2,44898	0
200	510,978	0,0493274	0

Οι συναρτήσεις που χρησιμοποιούνται για την υλοποίηση της συγκεκριμένης εφαρμογής με την χρήση του υβριδικού μοντέλου είναι σχεδόν οι ίδιες. Η διαφοροποίηση γίνεται στις συσκευές που γίνονται εγκατάσταση στους κόμβους. Η βασικότερη σχεδιαστική αρχή των δικτύων και του θεωρητικού μοντέλου OSI είναι η εύκολη τροποποίηση των επιμέρους πρωτοκόλλων χωρίς να χρειάζεται αλλαγές στα υπόλοιπα πρωτόκολλα. Οπότε και οι αλλαγές στην εφαρμογή που υλοποιήθηκε δεν χρειάζεται να είναι παραπάνω.

Για τον λόγο αυτό διαφοροποιείται η συνάρτηση: **AodvExample::CreateDevices ()**. Οι αλλαγές στη συγκεκριμένη συνάρτηση είναι πως οι κόμβοι όλοι τρέχουν το υβριδικό μοντέλο HWMP.

3.4 Τα πειράματα του πρότζεκτ

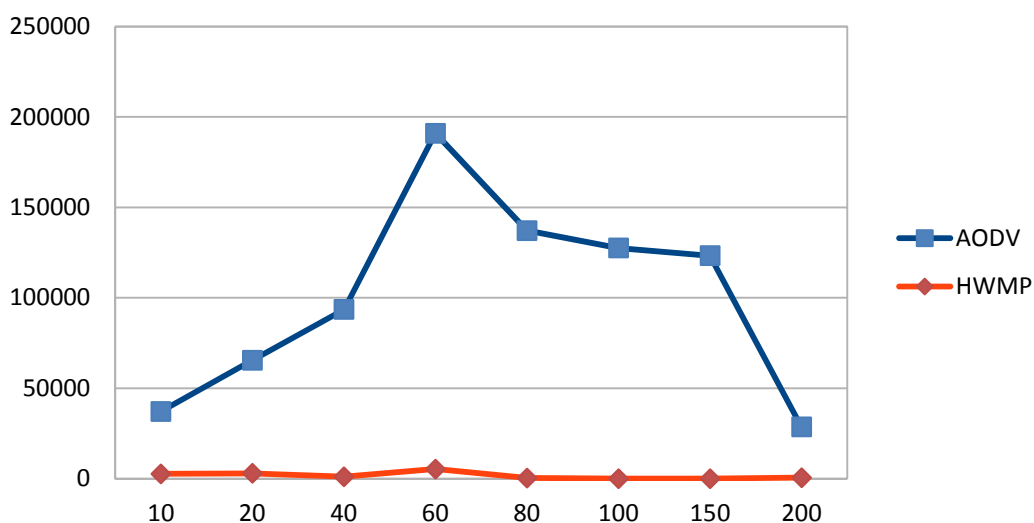
Στη συγκεκριμένη ενότητα περιγράφονται τα πειράματα που έγιναν και τα συμπεράσματα που μπορούμε να εξάγουμε από αυτά.

Σε όλα τα πειράματα έγινε σύγκριση των δύο πρωτοκόλλων. Για τα πειράματα τοποθετήσαμε 64 κόμβους σε ένα πλέγμα 8x8 που είναι οι σταθεροί κόμβοι. Κάθε φορά μεταβάλλεται το πλήθος των υπολογιστών πελατών που κινούνται μέσα στο πλέγμα. Έγιναν πειράματα για 10,20,40,60,80,100,150 και 200 κόμβους πελάτες. Στη συνέχεια παρουσιάζονται γραφικές παραστάσεις για το throughput, το packetdelay και το packetloss.

Throughput

Στην επόμενη εικόνα φαίνεται η γραφική παράσταση του throughput του δικτύου. Το throughput που επιτυγχάνεται από το HWMP είναι μικρότερο. Με μία πρώτη ανάγνωση αυτό θα φαινόταν να είναι άσχημο. Όμως αυτό που παρατηρούμε είναι η συνολική κίνηση που περνάει από τους κόμβους. Επειδή το AODV κάθε φορά θέλει να ανακαλύψει ένα νέο μονοπάτι (ή το ίδιο εξ'αρχής) πρέπει να στείλει πολλά πακέτα κάθε φορά. Αντίθετα, το υβριδικό τα αποθηκεύει και τα ξαναχρησιμοποιεί. Για το λόγο αυτό δεν αυξάνει η κίνηση και παραμένει σχετικά σταθερή.

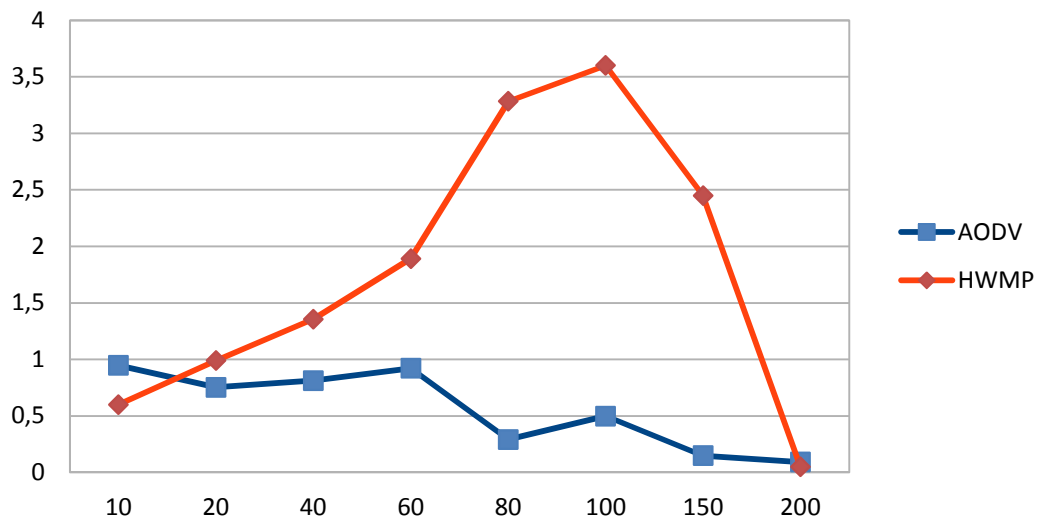
	Δίκτυο 8*8	
Num Clients	AODV	HWMP
10	37193,5	2658,62
20	65500	2875,39
40	93564,7	1022,45
60	190916	5271,37
80	137138	310,936
100	127474	17,355
150	123300	14,0174
200	28580,8	510,978



PacketDelay

Στη επόμενη εικόνα φαίνεται η καθυστέρηση των πακέτων. Για το AODV Παρατηρούμε πως αυτή μειώνεται όσο αυξάνει το πλήθος των κόμβων. Για την ακρίβεια μειώνεται η μέση καθυστέρηση. Αυτό συμβαίνει γιατί αυξάνονται οι κόμβοι άρα υπάρχουν περισσότερα μονοπάτια να ακολουθήσει το πακέτο χωρίς να υπάρχει μοναδική διαδρομή η οποία να εμφανίζει συμφόρηση. Αντίθετα η μέση καθυστέρηση αυξάνει στην περίπτωση που έχουμε το πρωτόκολλα HWMP, αυτό συμβαίνει μέχρι ένα σημείο. Από εκεί και έπειτα η μέση καθυστέρηση πάλι αρχίζει να μειώνεται. Τα πακέτα που καθυστερούν περισσότερο είναι πλέον λίγο μπροστά στα συνολικά πακέτα. Έτσι φαίνεται πως μειώνεται η μέση καθυστέρηση.

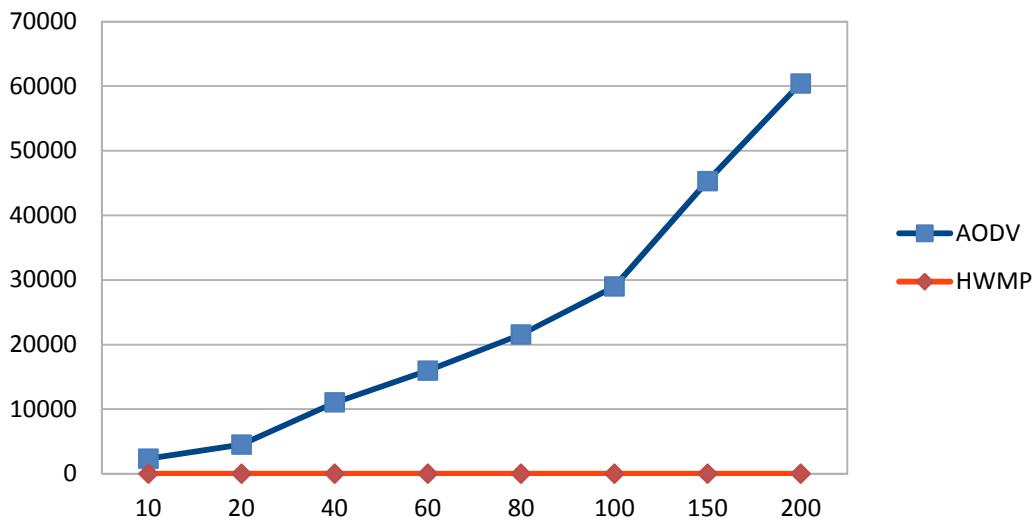
	Δικτυο 8*8	
Num Clients	AODV	HWMP
10	0,946923	0,600002
20	0,753281	0,989545
40	0,812501	1,35615
60	0,921937	1,89124
80	0,291468	3,28315
100	0,497624	3,60057
150	0,148598	2,44898
200	0,0924324	0,0493274



PacketLoss

Όπως φαίνεται στο AODV όσο αυξάνει το πλήθος των κόμβων τόσο αυξάνει η πιθανότητα λάθους. Αντίθετα, το HWMP δεν χάνει πακέτα.

Num Clients	Δικτυο 8*8	
	AODV	HWMP
10	2327	0
20	4499	0
40	11021	0
60	15966	0
80	21551	0
100	28952	0
150	45300	0
200	60402	0



3.5 NS-3 Scripts

HWMP

```

#include "ns3/aodv-module.h"
#include "ns3/core-module.h"
#include "ns3/network-module.h"
#include "ns3/internet-module.h"
#include "ns3/mobility-module.h"
#include "ns3/point-to-point-module.h"
#include "ns3/wifi-module.h"
#include "ns3/v4ping-helper.h"
#include <iostream>
#include <cmath>
#include "ns3/mesh-module.h"
#include "ns3/mesh-helper.h"
#include "ns3/mesh-module.h"
#include "ns3/wifi-phy.h"
#include "ns3/flow-monitor.h"
#include "ns3/flow-monitor-helper.h"
#include "ns3/ipv4-flow-classifier.h"
#include "ns3/random-variable.h"
#include "ns3/on-off-helper.h"
#include "ns3/packet-sink-helper.h"

```

```

#include <iostream>
#include <sstream>
#include <fstream>
#include <vector>
#define size_clients 10

using namespace ns3;

/**
 * \brief Test script.
 *
 * This script creates 1-dimensional grid topology and then ping last node from the first
one:
 *
 * [10.0.0.1] <-- step --> [10.0.0.2] <-- step --> [10.0.0.3] <-- step --> [10.0.0.4]
 *
 * ping 10.0.0.4
 */
class AodvExample
{
public:
    AodvExample ();
    bool Configure (int argc, char **argv);
    void Run ();
    void Report (std::ostream & os);

private:
    //\name parameters
    /\{
    /// Number of nodes
    uint32_t size;
    /// Distance between nodes, meters
    double step;
    /// Simulation time, seconds
    double totalTime;
    /// Write per-device PCAP traces if true
    bool pcap;
    /// Print routes if true
    bool printRoutes;

    uint16_t m_packetSize;
    /\}

    //\name network
    /\{
    NodeContainer nodes;
    NetDeviceContainer devices;

```

```

Ipv4InterfaceContainer interfaces;
// MeshHelper. Report is not static methods
MeshHelper mesh;
MeshHelper mesh_clients;
double randomStart;

//gia tous clients
NodeContainer nodes_clients;
NetDeviceContainer devices_clients;
Ipv4InterfaceContainer interfaces_clients;
WifiHelper wifi_clients;

/*
WifiHelper wifi2;
NetDeviceContainer devices2;
Ipv4InterfaceContainer interfaces2;

*/

private:
void CreateNodes ();
void CreateDevices ();
void InstallInternetStack ();
void InstallApplications ();
};

int main (int argc, char **argv)
{
AodvExample test;
if (!test.Configure (argc, argv))
NS_FATAL_ERROR ("Configuration failed. Aborted.");

test.Run ();
test.Report (std::cout);
return 0;
}

//-----
AodvExample::AodvExample () :
size (16*16),
step (100),
totalTime (10),
pcap (true),
printRoutes (true),
m_packetSize (1024),
randomStart (0.1)
{
}

```

```

bool
AodvExample::Configure (int argc, char **argv)
{
    // Enable AODV logs by default. Comment this if too noisy
    // LogComponentEnable("AodvRoutingProtocol", LOG_LEVEL_ALL);

    SeedManager::SetSeed (12345);
    CommandLine cmd;

    cmd.AddValue ("pcap", "Write PCAP traces.", pcap);
    cmd.AddValue ("printRoutes", "Print routing table dumps.", printRoutes);
    cmd.AddValue ("size", "Number of nodes.", size);
    cmd.AddValue ("time", "Simulation time, s.", totalTime);
    cmd.AddValue ("step", "Grid step, m", step);

    cmd.Parse (argc, argv);
    return true;
}

void AodvExample::Run ()
{
    // Config::SetDefault ("ns3::WifiRemoteStationManager::RtsCtsThreshold",
    UIntegerValue (1)); // enable rts cts all the time.
    CreateNodes ();
    CreateDevices ();
    InstallInternetStack ();
    InstallApplications ();

    std::cout << "Starting HWMP simulation for " << totalTime << " s ...\n";
    FlowMonitorHelper flowmon;
    Ptr<FlowMonitor> monitor = flowmon.InstallAll();

    Simulator::Stop (Seconds (totalTime));
    Simulator::Run ();

    // Define variables to calculate the metrics
    int k=0;
    int totalTxPackets = 0;
    int totalRxPackets = 0;
    double totalTxBytes = 0;
    double totalRxBytes = 0;
    double totalDelay = 0;
    double totalRxBitrate = 0;
    double diffTx, diffRx;
    double pdf_value, txbitrate_value, delay_value;
    double rxbitrate_value;

```

```

double pdf_total, rxbitrate_total, delay_total;
int totalPacketLost=0;

//Print per flow statistics
monitor->CheckForLostPackets ();
Ptr<Ipv4FlowClassifier> classifier = DynamicCast<Ipv4FlowClassifier>
(flowmon.GetClassifier ());
std::map<FlowId, FlowMonitor::FlowStats> stats = monitor->GetFlowStats ();
for (std::map<FlowId, FlowMonitor::FlowStats>::const_iterator i = stats.begin (); i !=
stats.end (); ++i)
{
    Ipv4FlowClassifier::FiveTuple t = classifier->FindFlow (i->first);
    difftx = i->second.timeLastTxPacket.GetSeconds() -
i->second.timeFirstTxPacket.GetSeconds();
    diffrx = i->second.timeLastRxPacket.GetSeconds() - i-
>second.timeFirstRxPacket.GetSeconds();
    pdf_value = (double) i->second.rxPackets / (double) i->second.txPackets *
100;
    txbitrate_value = (double) i->second.txBytes * 8 / 1024 / difftx;
    if (i->second.rxPackets > 0){
        if(diffrx>0){
            rxbitrate_value = (double)i->second.rxPackets *
m_packetSize * 8 / 1024 / diffrx;
        }
        else{
            rxbitrate_value = 0.0;
        }
        delay_value = (double) i->second.delaySum.GetSeconds() / (double)
i->second.rxPackets;
    }
    else{
        rxbitrate_value = 0;
        delay_value = 0;
    }
    // We are only interested in the metrics of the data flows
    if ((!t.destinationAddress.IsSubnetDirectedBroadcast("255.255.255.0")))
    {
        k++;
        // Plot the statistics for each data flow
        std::cout << "\nFlow " << k << " (" << t.sourceAddress << " -> " <<
t.destinationAddress << ") \n";
        //std::cout << "Tx Packets: " << i->second.txPackets << "\n";
        //std::cout << "Rx Packets: " << i->second.rxPackets << "\n";
        std::cout << "Lost Packets: " << i->second.lostPackets << "\n";
        //std::cout << "Dropped Packets: " << i-
>second.packetsDropped.size() << "\n";
        std::cout << "PDF: " << pdf_value << " %\n";
        std::cout << "Average delay: " << delay_value << "s\n";
    }
}

```

```

        std::cout << "Rx bitrate: " << rxbitrate_value << " kbps\n";
        std::cout << "Tx bitrate: " << txbitrate_value << " kbps\n\n";
        // Accumulate for average statistics
        totalPacketLost += i->second.lostPackets;
        totaltxPackets += i->second.txPackets;
        totaltxbytes += i->second.txBytes;
        totalrxPackets += i->second.rxPackets;
        totaldelay += i->second.delaySum.GetSeconds();
        totalrxbitrate += rxbitrate_value;
        totalrxbytes += i->second.rxBytes;
    }
}
// Average all nodes statistics
if (totaltxPackets > 0){
    pdf_total = (double) totalrxPackets / (double) totaltxPackets * 100;
}
else{
    pdf_total = 0;
}
if (totalrxPackets != 0){
    rxbitrate_total = totalrxbitrate;
    delay_total = (double) totaldelay / (double) totalrxPackets;
}
else{
    rxbitrate_total = 0;
    delay_total = 0;
}
//print all nodes statistics
std::cout << "\nTotal PDF: " << pdf_total << " %\n";
std::cout << "Total Rx bitrate: " << rxbitrate_total << " kbps\n";
std::cout << "Total Delay: " << delay_total << " s\n";
//print all nodes statistics in files
std::ostringstream os;
os << "1_AODV_PDF.txt";
std::ofstream of (os.str().c_str(), std::ios::out | std::ios::app);
of << pdf_total << "\n";
of.close ();
std::ostringstream os2;
os2 << "1_AODV_Delay.txt";
std::ofstream of2 (os2.str().c_str(), std::ios::out | std::ios::app);
of2 << delay_total << "\n";
of2.close ();
std::ostringstream os3;
os3 << "1_AODV_Throu.txt";
std::ofstream of3 (os3.str().c_str(), std::ios::out | std::ios::app);
of3 << rxbitrate_total << "\n";
of3.close ();

```

```

std::ostringstream os4;
os4 << "1_AODV_PacketLost.txt";
std::ofstream of4 (os4.str().c_str(), std::ios::out | std::ios::app);
of4 << totalPacketLost << "\n";
of4.close ();
Simulator::Destroy ();
//m_timeEnd=clock();
//m_timeTotal=(m_timeEnd - m_timeStart)/(double) CLOCKS_PER_SEC;
//std::cout << "\n*** Simulation time: " << m_timeTotal << "s\n\n";
//return 0;

}

void AodvExample::Report (std::ostream &)
{
    // Using this function we print detailed statistics of each mesh point device
    // These statistics are used later with an AWK files to extract routing metrics
    unsigned n (0);
    for (NetDeviceContainer::Iterator i = devices.Begin (); i != devices.End (); ++i,
    ++n){
        std::ostringstream os;
        //os << "mp-report1-" << n << ".xml";
        os << "mp-report.xml";
        std::ofstream of;
        of.open (os.str().c_str(), std::ios::out | std::ios::app);
        if (! of.is_open ())
        {
            std::cerr << "Error: Can't open file " << os.str() << "\n";
            return;
        }
        mesh.Report (*i, of);
        of.close ();
    }
    n = 0;
}

static void CourseChange (std::string foo, Ptr<const MobilityModel> mobility)
{
    /*
    *   Vector pos = mobility->GetPosition ();
    *   Vector vel = mobility->GetVelocity ();
    *   std::cout << Simulator::Now () << ", model=" << mobility << ", POS: x=" <<
pos.x << ", y=" << pos.y
    *   << ", z=" << pos.z << "; VEL:" << vel.x << ", y=" << vel.y
    *   << ", z=" << vel.z << std::endl;
    */
}

```



```

Void AodvExample::CreateNodes ()
{
    std::cout << "Creating " << (unsigned)size << " nodes " << step << " m apart.\n";
    nodes.Create (size);
    // Name nodes
    for (uint32_t i = 0; i < size; ++i)
    {
        std::ostringstream os;
        os << "node-" << i;
        Names::Add (os.str (), nodes.Get (i));
    }
    // Create static grid
    MobilityHelper mobility;
    mobility.SetPositionAllocator ("ns3::GridPositionAllocator",
        "MinX", DoubleValue (0.0),
        "MinY", DoubleValue (0.0),
        "DeltaX", DoubleValue (step),
        "DeltaY", DoubleValue (step),
        "GridWidth", UIntegerValue (16),
        //"GridHeight", UIntegerValue(200),
        "LayoutType", StringValue ("RowFirst"));
    mobility.SetMobilityModel ("ns3::ConstantPositionMobilityModel");
    mobility.Install (nodes);

    //gia tous clients
    std::cout << "Creating " << (unsigned)size_clients << " Clients " << step/2 << " m
    apart.\n";
    nodes_clients.Create (size_clients);

    // Name nodes
    for (uint32_t i = 0; i < size_clients; ++i)
    {
        std::ostringstream os;
        os << "Client-" << i;
        Names::Add (os.str (), nodes_clients.Get (i));
    }
    // Create static grid
    MobilityHelper mobility_clients;

    mobility_clients.SetPositionAllocator ("ns3::RandomDiscPositionAllocator",
        "X", StringValue ("800.0"),
        "Y", StringValue ("800.0"),
        "Rho", StringValue
("ns3::UniformRandomVariable[Min=0|Max=800]"));
    mobility_clients.SetMobilityModel ("ns3::RandomWalk2dMobilityModel",
        "Mode", StringValue ("Time"),

```

```

        "Time", StringValue ("2s"),
        "Speed", StringValue
("ns3::ConstantRandomVariable[Constant=1.0]"),
        "Bounds", StringValue ("0|800|0|800"));
    mobility_clients.Install (nodes_clients);
    Config::Connect ("/NodeList/*/ns3::MobilityModel/CourseChange", MakeCallback
(&CourseChange));

}

Void AodvExample::CreateDevices ()
{
    double m_txpower = 18.0;

    YansWifiPhyHelper WifiPhy = YansWifiPhyHelper::Default ();

    WifiPhy.Set("EnergyDetectionThreshold", DoubleValue (-89.0) );
    WifiPhy.Set("CcaMode1Threshold", DoubleValue (-62.0) );
    WifiPhy.Set("TxGain", DoubleValue (1.0) );
    WifiPhy.Set("RxGain", DoubleValue (1.0) );
    WifiPhy.Set("TxPowerLevels", UIntegerValue (1) );
    WifiPhy.Set("TxPowerEnd", DoubleValue (m_txpower) );
    WifiPhy.Set("TxPowerStart", DoubleValue (m_txpower) );
    WifiPhy.Set("RxNoiseFigure", DoubleValue (7.0) );

    YansWifiChannelHelper WifiChannel;
    WifiChannel.SetPropagationDelay ("ns3::ConstantSpeedPropagationDelayModel");
    WifiChannel.AddPropagationLoss
("ns3::LogDistancePropagationLossModel", "Exponent",
    StringValue ("2.7"));
    WifiPhy.SetChannel (WifiChannel.Create ());
    // Configure the parameters of the Peer Link
    Config::SetDefault ("ns3::dot11s::PeerLink::MaxBeaconLoss", UIntegerValue (20));
    Config::SetDefault ("ns3::dot11s::PeerLink::MaxRetries", UIntegerValue (4));
    Config::SetDefault ("ns3::dot11s::PeerLink::MaxPacketFailure", UIntegerValue (5));

    //Config::SetDefault
("ns3::dot11s::PeerManagementProtocol::EnableBeaconCollision-
Avoidance", BooleanValue (false));
    // Configure the parameters of the HWMP
    Config::SetDefault
("ns3::dot11s::HwmpProtocol::Dot11MeshHWMPActivePathTimeout", TimeValue
(Seconds (100)));
    Config::SetDefault
("ns3::dot11s::HwmpProtocol::Dot11MeshHWMPActiveRootTimeout", TimeValue
(Seconds (100)));
    Config::SetDefault
("ns3::dot11s::HwmpProtocol::Dot11MeshHWMPmaxPRERetries", UIntegerValue (5));

```

```

    Config::SetDefault ("ns3::dot11s::HwmpProtocol::UnicastPreqThreshold",
UIntegerValue (10));
    Config::SetDefault
("ns3::dot11s::HwmpProtocol::UnicastDataThreshold",UIntegerValue (5));
    Config::SetDefault ("ns3::dot11s::HwmpProtocol::DoFlag", BooleanValue (true));
    Config::SetDefault ("ns3::dot11s::HwmpProtocol::RfFlag", BooleanValue (false));

    mesh = MeshHelper::Default ();
    mesh.SetStandard (WIFI_PHY_STANDARD_80211a);
    mesh.SetMacType ("RandomStart", TimeValue (Seconds(randomStart)));
    mesh.SetRemoteStationManager ("ns3::ConstantRateWifiManager", "DataMode",
StringValue ("OfdmRate6Mbps"), "RtsCtsThreshold", UintegerValue (2500));
    // Set number of interfaces - default is single-interface mesh point
    mesh.SetNumberOfInterfaces (1);
    mesh.SetStackInstaller ("ns3::Dot11sStack");
    devices = mesh.Install (WifiPhy, nodes);

    /*
    //gia tous clients
    mesh_clients = MeshHelper::Default ();
    mesh_clients.SetStandard (WIFI_PHY_STANDARD_80211a);
    mesh_clients.SetMacType ("RandomStart", TimeValue (Seconds(randomStart)));
    mesh_clients.SetRemoteStationManager ("ns3::ConstantRateWifiManager",
"DataMode",
StringValue ("OfdmRate6Mbps"), "RtsCtsThreshold",
UIntegerValue (2500));
    // Set number of interfaces - default is single-interface mesh point
    mesh_clients.SetNumberOfInterfaces (1);
    mesh_clients.SetStackInstaller ("ns3::Dot11sStack");
    */
    devices_clients = mesh.Install (WifiPhy, nodes_clients);

}

void
AodvExample::InstallInternetStack ()
{
    // you can configure AODV attributes here using aodv.Set(name, value)
    InternetStackHelper stack;
    stack.Install (nodes);
    stack.Install(nodes_clients);
    Ipv4AddressHelper address;
    address.SetBase ("10.0.0.0", "255.0.0.0");
    interfaces = address.Assign (devices);

    //interfaces2 = address.Assign (devices2);
    interfaces_clients = address.Assign(devices_clients);
}

```

```

Void AodvExample::InstallApplications ()
{
    ApplicationContainer apps[size_clients];
    for(int i=0; i<size_clients; i++){
        OnOffHelper onoff ("ns3::UdpSocketFactory", Address (InetSocketAddress
(interfaces.GetAddress (size/2-1), 49000)));
        apps[i] = onoff.Install (nodes_clients.Get(i));
        apps[i].Start (Seconds (2));
        apps[i].Stop (Seconds (totalTime-1));
    }

    // Create a packet sink to receive the packets
    PacketSinkHelper sink
("ns3::UdpSocketFactory",InetSocketAddress(interfaces.GetAddress (size/2-1), 49000));
    for(int i=0; i<size_clients; i++){
        apps[i] = sink.Install (nodes.Get (size/2-1));
        apps[i].Start (Seconds (1.0));
    }

    // move node away
    //Ptr<Node> node = nodes.Get (size/2);
    //Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
    //Simulator::Schedule (Seconds (totalTime/3), &MobilityModel::SetPosition, mob,
Vector (1e5, 1e5, 1e5));
    for(uint32_t i=0; i<size; i++){
        Ptr <Node> node = nodes.Get(i);
        Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
        Vector pos = mob->GetPosition();
        std::cout<<"Node: "<<i<<" Position ("<< (pos.x)
<<","<<(pos.y)<<")<<"\tIP: "<< interfaces.GetAddress(i) <<std::endl;
    }

    for(uint32_t i=0; i<size_clients; i++){
        Ptr <Node> node = nodes_clients.Get(i);
        Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
        Vector pos = mob->GetPosition();
        std::cout<<"Client: "<<i<<" Position ("<< (pos.x)
<<","<<(pos.y)<<")<<"\tIP: "<< interfaces_clients.GetAddress(i) <<std::endl;
    }
}

```

AODV

```
#include "ns3/aodv-module.h"
#include "ns3/core-module.h"
#include "ns3/network-module.h"
#include "ns3/internet-module.h"
#include "ns3/mobility-module.h"
#include "ns3/point-to-point-module.h"
#include "ns3/wifi-module.h"
#include "ns3/v4ping-helper.h"
#include <iostream>
#include <cmath>
#include "ns3/mesh-module.h"
#include "ns3/mesh-helper.h"
#include "ns3/mesh-module.h"
#include "ns3/wifi-phy.h"
#include "ns3/flow-monitor.h"
#include "ns3/flow-monitor-helper.h"
#include "ns3/ipv4-flow-classifier.h"
#include "ns3/random-variable.h"
#include "ns3/on-off-helper.h"
#include "ns3/packet-sink-helper.h"
#include <iostream>
#include <sstream>
#include <fstream>
#include <vector>
#define size_clients 10

using namespace ns3;

/**
 * \brief Test script.
 *
 * This script creates 1-dimensional grid topology and then ping last node from the first one:
 *
 * [10.0.0.1] <-- step --> [10.0.0.2] <-- step --> [10.0.0.3] <-- step --> [10.0.0.4]
 *
 * ping 10.0.0.4
 */
class AodvExample
{
public:
```

```

AodvExample ();
bool Configure (int argc, char **argv);
void Run ();
void Report (std::ostream & os);

private:
//\name parameters
//\{
// Number of nodes
uint32_t size;
// Distance between nodes, meters
double step;
// Simulation time, seconds
double totalTime;
// Write per-device PCAP traces if true
bool pcap;
// Print routes if true
bool printRoutes;

uint16_t m_packetSize;
//\}

//\name network
//\{
NodeContainer nodes;
NetDeviceContainer devices;
Ipv4InterfaceContainer interfaces;
WifiHelper wifi_aodv;

WifiHelper wifi2;
NetDeviceContainer devices2;
Ipv4InterfaceContainer interfaces2;

//gia tous clients
NodeContainer nodes_clients;
NetDeviceContainer devices_clients;
Ipv4InterfaceContainer interfaces_clients;
WifiHelper wifi_clients;

private:
void CreateNodes ();
void CreateDevices ();
void InstallInternetStack ();
void InstallApplications ();
};

```

```

int main (int argc, char **argv)
{
    AodvExample test;
    if (!test.Configure (argc, argv))
        NS_FATAL_ERROR ("Configuration failed. Aborted.");

    test.Run ();
    test.Report (std::cout);
    return 0;
}

//-----
AodvExample::AodvExample () :
    size (16*16), /*Prwth allagh gia ta peiramata*/
    step (100),
    totalTime (10),
    pcap (true),
    printRoutes (true),
    m_packetSize (1024)
{
}

bool
AodvExample::Configure (int argc, char **argv)
{
    // Enable AODV logs by default. Comment this if too noisy
    // LogComponentEnable("AodvRoutingProtocol", LOG_LEVEL_ALL);

    SeedManager::SetSeed (12345);
    CommandLine cmd;

    cmd.AddValue ("pcap", "Write PCAP traces.", pcap);
    cmd.AddValue ("printRoutes", "Print routing table dumps.", printRoutes);
    cmd.AddValue ("size", "Number of nodes.", size);
    cmd.AddValue ("time", "Simulation time, s.", totalTime);
    cmd.AddValue ("step", "Grid step, m", step);

    cmd.Parse (argc, argv);
    return true;
}

void AodvExample::Run ()
{
    // Config::SetDefault ("ns3::WifiRemoteStationManager::RtsCtsThreshold",
    UIntegerValue (1)); // enable rts cts all the time. Config::SetDefault
    ("ns3::RandomWalk2dMobilityModel::Mode", StringValue ("Time"));
}

```

```

std::cout<<"Test!!\n";
CreateNodes ();
CreateDevices ();
InstallInternetStack ();
InstallApplications ();

std::cout << "Starting AODV simulation for " << totalTime << " s ...\n";
FlowMonitorHelper flowmon;
Ptr<FlowMonitor> monitor = flowmon.InstallAll();

Simulator::Stop (Seconds (totalTime));
Simulator::Run ();

// Define variables to calculate the metrics
int k=0;
int totalTxPackets = 0;
int totalRxPackets = 0;
double totalTxBytes = 0;
double totalRxBytes = 0;
double totalDelay = 0;
double totalRxBitrate = 0;
double diffTx, diffRx;
double pdf_value, txbitrate_value, delay_value;
double rxbitrate_value;
double pdf_total, rxbitrate_total, delay_total;
int totalPacketLost=0;
//Print per flow statistics
monitor->CheckForLostPackets ();
Ptr<Ipv4FlowClassifier> classifier = DynamicCast<Ipv4FlowClassifier>
(flowmon.GetClassifier ());
std::map<FlowId, FlowMonitor::FlowStats> stats = monitor->GetFlowStats ();
for (std::map<FlowId, FlowMonitor::FlowStats>::const_iterator i = stats.begin (); i !=
stats.end (); ++i)
{
    Ipv4FlowClassifier::FiveTuple t = classifier->FindFlow (i->first);
    diffTx = i->second.timeLastTxPacket.GetSeconds() -
i->second.timeFirstTxPacket.GetSeconds();
    diffRx = i->second.timeLastRxPacket.GetSeconds() - i-
>second.timeFirstRxPacket.GetSeconds();
    pdf_value = (double) i->second.rxPackets / (double) i->second.txPackets * 100;
    txbitrate_value = (double) i->second.txBytes * 8 / 1024 / diffTx;
    if (i->second.rxPackets > 0){
        if(diffRx>0){
            rxbitrate_value = (double)i->second.rxPackets * m_packetSize * 8 /
1024 / diffRx;
        }
        else{

```



```

        rxbitrate_value = 0.0;
    }
    delay_value = (double) i->second.delaySum.GetSeconds() / (double) i-
>second.rxPackets;
    }
    else{
        rxbitrate_value = 0;
        delay_value = 0;
    }
    // We are only interested in the metrics of the data flows
    if (!(t.destinationAddress.IsSubnetDirectedBroadcast("255.255.255.0")))
    {
        k++;
        // Plot the statistics for each data flow
        std::cout << "\nFlow " << k << " (" << t.sourceAddress << " -> " <<
t.destinationAddress << ")\n";
        //std::cout << "Tx Packets: " << i->second.txPackets << "\n";
        //std::cout << "Rx Packets: " << i->second.rxPackets << "\n";
        std::cout << "Lost Packets: " << i->second.lostPackets << "\n";
        //std::cout << "Dropped Packets: " << i->second.packetsDropped.size()
<< "\n";

        std::cout << "PDF: " << pdf_value << " %\n";
        std::cout << "Average delay: " << delay_value << "s\n";
        std::cout << "Rx bitrate: " << rxbitrate_value << " kbps\n";
        std::cout << "Tx bitrate: " << txbitrate_value << " kbps\n\n";
        // Acumulate for average statistics
        totalPacketLost += i->second.lostPackets;
        totaltxPackets += i->second.txPackets;
        totaltxbytes += i->second.txBytes;
        totalrxPackets += i->second.rxPackets;
        totaldelay += i->second.delaySum.GetSeconds();
        totalrxbitrate += rxbitrate_value;
        totalrxbytes += i->second.rxBytes;
    }
}
// Average all nodes statistics
if (totaltxPackets > 0){
    pdf_total = (double) totalrxPackets / (double) totaltxPackets * 100;
}
else{
    pdf_total = 0;
}
if (totalrxPackets != 0){
    rxbitrate_total = totalrxbitrate;
    delay_total = (double) totaldelay / (double) totalrxPackets;
}
else{
    rxbitrate_total = 0;
}

```

```

        delay_total = 0;
    }
    //print all nodes statistics
    std::cout << "\nTotal PDF: " << pdf_total << " %\n";
    std::cout << "Total Rx bitrate: " << rxbitrate_total << " kbps\n";
    std::cout << "Total Delay: " << delay_total << " s\n";
    //print all nodes statistics in files
    std::ostringstream os;
    os << "1_AODV_PDF.txt";
    std::ofstream of (os.str().c_str(), std::ios::out | std::ios::app);
    of << pdf_total << "\n";
    of.close ();
    std::ostringstream os2;
    os2 << "1_AODV_Delay.txt";
    std::ofstream of2 (os2.str().c_str(), std::ios::out | std::ios::app);
    of2 << delay_total << "\n";
    of2.close ();
    std::ostringstream os3;
    os3 << "1_AODV_Throu.txt";
    std::ofstream of3 (os3.str().c_str(), std::ios::out | std::ios::app);
    of3 << rxbitrate_total << "\n";
    of3.close ();

    std::ostringstream os4;
    os4 << "1_AODV_PacketLost.txt";
    std::ofstream of4 (os4.str().c_str(), std::ios::out | std::ios::app);
    of4 << totalPacketLost << "\n";
    of4.close ();
    Simulator::Destroy ();
    //m_timeEnd=clock();
    //m_timeTotal=(m_timeEnd - m_timeStart)/(double) CLOCKS_PER_SEC;
    //std::cout << "\n*** Simulation time: " << m_timeTotal << "s\n\n";
    //return 0;
}

void AodvExample::Report (std::ostream &)
{
}

static void CourseChange (std::string foo, Ptr<const MobilityModel> mobility)
{
    /*
    Vector pos = mobility->GetPosition ();
    Vector vel = mobility->GetVelocity ();
    std::cout << Simulator::Now () << ", model=" << mobility << ", POS: x=" << pos.x
    << ", y=" << pos.y
    << ", z=" << pos.z << "; VEL:" << vel.x << ", y=" << vel.y
    << ", z=" << vel.z << std::endl;
    */
}

```

```

        */
    }

Void AodvExample::CreateNodes ()
{
    std::cout << "Creating " << this->size << " nodes " << step << " m apart.\n";
    nodes.Create (size);

    // Name nodes
    for (uint32_t i = 0; i < size; ++i)
    {
        std::ostringstream os;
        os << "node-" << i;
        Names::Add (os.str (), nodes.Get (i));
    }
    // Create static grid
    MobilityHelper mobility;
    mobility.SetPositionAllocator ("ns3::GridPositionAllocator",
        "MinX", DoubleValue (0.0),
        "MinY", DoubleValue (0.0),
        "DeltaX", DoubleValue (step),
        "DeltaY", DoubleValue (step),
        "GridWidth", UIntegerValue (16),
        "LayoutType", StringValue ("RowFirst"));
    mobility.SetMobilityModel ("ns3::ConstantPositionMobilityModel");
    mobility.Install (nodes);

    //gia tous clients
    std::cout << "Creating " << (unsigned)size_clients << " Clients " << step/2 << " m apart.\n";
    nodes_clients.Create (size_clients);
    // Name nodes
    for (uint32_t i = 0; i < size_clients; ++i)
    {
        std::ostringstream os;
        os << "Client-" << i;
        Names::Add (os.str (), nodes_clients.Get (i));
    }
    // Create static grid
    MobilityHelper mobility_clients;

    mobility_clients.SetPositionAllocator ("ns3::RandomDiscPositionAllocator",
        "X", StringValue ("50.0"),
        "Y", StringValue ("100.0"),
        "Rho", StringValue ("ns3::UniformRandomVariable[Min=0|Max=50]"));
    mobility_clients.SetMobilityModel ("ns3::RandomWalk2dMobilityModel",
        "Mode", StringValue ("Time"),
        "Time", StringValue ("2s"),
        "Speed", StringValue ("ns3::ConstantRandomVariable[Constant=1.0]"),

```

```

        "Bounds", StringValue ("0|100|0|200"));
    mobility_clients.Install (nodes_clients);
    Config::Connect ("/NodeList/*/ns3::MobilityModel/CourseChange", MakeCallback
(&CourseChange));
}

Void AodvExample::CreateDevices ()
{
    double m_txpower = 18.0;

    // Configure YansWifiChannel
    YansWifiPhyHelper WifiPhy = YansWifiPhyHelper::Default ();

    WifiPhy.Set("EnergyDetectionThreshold", DoubleValue (-89.0) );
    WifiPhy.Set("CcaMode1Threshold", DoubleValue (-62.0) );
    WifiPhy.Set("TxGain", DoubleValue (1.0) );
    WifiPhy.Set("RxGain", DoubleValue (1.0) );
    WifiPhy.Set("TxPowerLevels", UIntegerValue (1) );
    WifiPhy.Set("TxPowerEnd", DoubleValue (m_txpower) );
    WifiPhy.Set("TxPowerStart", DoubleValue (m_txpower) );
    WifiPhy.Set("RxNoiseFigure", DoubleValue (7.0) );
    YansWifiChannelHelper WifiChannel;
    WifiChannel.SetPropagationDelay ("ns3::ConstantSpeedPropagationDelayModel");
    WifiChannel.AddPropagationLoss
("ns3::LogDistancePropagationLossModel", "Exponent", StringValue ("2.7"));
    WifiPhy.SetChannel (WifiChannel.Create ());
    wifi_aodv.SetStandard (WIFI_PHY_STANDARD_80211b);
    wifi_aodv.SetRemoteStationManager ("ns3::ConstantRateWifiManager",
    "DataMode", StringValue ("DsssRate2Mbps"), "RtsCtsThreshold", UIntegerValue (2500));
    // Install protocols and return container
    NqosWifiMacHelper wifiMac = NqosWifiMacHelper::Default ();
    wifiMac.SetType ("ns3::AdhocWifiMac");
    devices = wifi_aodv.Install (WifiPhy, wifiMac, nodes);
    //gia ta 2a interface
    wifi2.SetStandard (WIFI_PHY_STANDARD_80211a);
    wifi2.SetRemoteStationManager ("ns3::ConstantRateWifiManager", "DataMode",
StringValue ("OfdmRate6Mbps"), "RtsCtsThreshold", UIntegerValue (2500));
    // Install protocols and return container
    NqosWifiMacHelper wifiMac2 = NqosWifiMacHelper::Default ();
    wifiMac2.SetType ("ns3::AdhocWifiMac");
    devices2 = wifi2.Install (WifiPhy, wifiMac2, nodes);

    //gia tous clients
    wifi_clients.SetStandard (WIFI_PHY_STANDARD_80211b);
    wifi_clients.SetRemoteStationManager ("ns3::ConstantRateWifiManager",
    "DataMode", StringValue ("DsssRate1Mbps"), "RtsCtsThreshold", UIntegerValue (2500));
    // Install protocols and return container

```

```

NqosWifiMacHelper wifiMac3 = NqosWifiMacHelper::Default ();
wifiMac3.SetType ("ns3::AdhocWifiMac");
devices_clients = wifi_clients.Install (WifiPhy, wifiMac3, nodes_clients);

}

Void AodvExample::InstallInternetStack ()
{
    AodvHelper aodv;

    aodv.Set ("AllowedHelloLoss", UIntegerValue (20));
    aodv.Set ("HelloInterval", TimeValue (Seconds (3)));
    aodv.Set ("RreqRetries", UIntegerValue (5));
    aodv.Set ("ActiveRouteTimeout", TimeValue (Seconds (100)));
    aodv.Set ("DestinationOnly", BooleanValue (true));

    // you can configure AODV attributes here using aodv.Set(name, value)
    InternetStackHelper stack;
    stack.SetRoutingHelper (aodv); // has effect on the next Install ()
    stack.Install (nodes);
    stack.Install(nodes_clients);
    Ipv4AddressHelper address;

    address.SetBase ("10.0.0.0", "255.0.0.0");
    interfaces = address.Assign (devices);
    interfaces2 = address.Assign (devices2);

    interfaces_clients = address.Assign(devices_clients);

    if (printRoutes)
    {
        Ptr<OutputStreamWrapper> routingStream = Create<OutputStreamWrapper>
("aodv.routes", std::ios::out);
        aodv.PrintRoutingTableAllAt (Seconds (8), routingStream);
    }
}

void
AodvExample::InstallApplications ()
{
    ApplicationContainer apps[size_clients];
    for(int i=0; i<size_clients; i++){
        OnOffHelper onoff ("ns3::UdpSocketFactory", Address (InetSocketAddress
(interfaces.GetAddress (size-1), 49000)));

```

```

        apps[i] = onoff.Install (nodes_clients.Get(i));
        apps[i].Start (Seconds (2));
        apps[i].Stop (Seconds (totalTime-1));
    }

    // Create a packet sink to receive the packets
    PacketSinkHelper sink
("ns3::UdpSocketFactory",InetAddress(interfaces.GetAddress (size-1), 49000));
    for(int i=0; i<size_clients; i++){
        apps[i] = sink.Install (nodes.Get (size-1));
        apps[i].Start (Seconds (1.0));
    }

    // move node away
    //Ptr<Node> node = nodes.Get (size/2);
    //Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
    //Simulator::Schedule (Seconds (totalTime/3), &MobilityModel::SetPosition, mob,
Vector (1e5, 1e5, 1e5));
    for(uint32_t i=0; i<size; i++){
        Ptr <Node> node = nodes.Get(i);
        Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
        Vector pos = mob->GetPosition();
        std::cout<<"Node: "<<i<<" Position ("<< (pos.x)
<<","<<(pos.y)<<")"<<"\tIP: "<< interfaces.GetAddress(i) <<std::endl;
    }

    for(uint32_t i=0; i<size_clients; i++){
        Ptr <Node> node = nodes_clients.Get(i);
        Ptr<MobilityModel> mob = node->GetObject<MobilityModel> ();
        Vector pos = mob->GetPosition();
        std::cout<<"Client: "<<i<<" Position ("<< (pos.x)
<<","<<(pos.y)<<")"<<"\tIP: "<< interfaces_clients.GetAddress(i) <<std::endl;
    }
}

```

ΑΝΑΦΟΡΕΣ-ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Raniwala, A. Dept. of Comput. Sci., State Univ. of New York, Stony Brook, NY, USA
Tzi-cker Chiueh Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network
- [2] Ian F. Akyildiz, Xudong Wang, Weilin Wang Wireless mesh networks: a survey
- [3] John Bicket, Daniel Aguayo, Sanjit Biswas, Robert Morris Architecture and evaluation of an unplanned 802.11b mesh network
- [4] Mansoor Alicherry, Randeep Bhatia, Li (Erran) Li Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks
- [5] George Aggelou Wireless Mesh Networking
- [6] Y Zhang, J Luo, H Hu Wireless mesh networking: architectures, protocols and standards
- [7] Ieee Campista, M.E.M. Univ.Fed. do Rio de Janeiro,Rio de Janeiro Esposito,P.M. Moraes, I.M. Costa, L.H.M.K. Duarte, O.C.M.B. Passos, D.G. de Albuquerque, C.V.N. Saade, D.C.M. Rubinstein, M.G.Routing Metrics and Protocols for Wireless Mesh Networks
- [8] Tzu-Jane Tsai Comput. & Commun. Res. Labs., Ind. Technol. Res. Inst., Hsinchu, Taiwan Ju-Wei Chen IEEE 802.11 MAC protocol over wireless mesh networks: problems and perspectives
- [9] Skalli, H. IMT Lucca Inst. for Adv. Studies, Lucca Ghosh, S. Das, S.K. Lenzini, L. Conti, M. Channel Assignment Strategies for Multiradio Wireless Mesh Networks: Issues and Solutions
- [10] Brian Zill, Jitendra Padhye, Richard Draves Routing in multi-radio, multi-hop wireless mesh networks
- [11] Jane-Hwa Huang
Dept. of Commun. Eng., Nat. Chiao Tung Univ., Hsinchu, Taiwan
Li-Chun Wang ; Chung-Ju Chang Coverage and capacity of a wireless mesh network
- [12] Chun-Yen Hsu, Jean-Lien C. Wu, Shun-Te Wang, Chi-Yao Hong Survivable and delay-guaranteed backbone wireless mesh network design
- [13] Pirzada, A.A. Nat. ICT Australia Ltd., Brisbane Portmann, M. High Performance AODV Routing Protocol for Hybrid Wireless Mesh Networks
- [14] Kai Yang Dept. of Comput. Sci., Univ. of Xidian, Xian, China Jian-feng Ma Zi-hui Miao Hybrid Routing Protocol for Wireless Mesh Network