

ΠΕΡΙΛΗΨΗ

Στην παρούσα Διατριβή Μάστερ σχεδιάσαμε και υλοποιήσαμε ένα αλγόριθμο για τα Ασύρματα Δίκτυα Αισθητήρων (ΑΔΑ) εμπνευσμένο από τη βιολογία, ο οποίος υλοποιεί δρομολόγηση πακέτων σε συνδυασμό με μηχανισμό για αποφυγή της συμφόρησης. Ο μηχανισμός αυτός αναπτύχθηκε στη βάση του μαθηματικού πληθυσμιακού μοντέλου ανταγωνισμού Lotka-Volterra.

Ο αλγόριθμος αυτός είναι απλός αφού υλοποιείται στους κόμβους αισθητήρων, οι οποίοι έχουν περιορισμένους πόρους, όπως μνήμη και υπολογιστική δύναμη. Επίσης, ο αλγόριθμος είναι κατανεμημένος έτσι ώστε να μην υπάρχει η ανάγκη για μια κεντρική οντότητα που να διευθύνει όλο το δίκτυο. Η οντότητα αυτή θα απαιτούσε περισσότερα μηνύματα ελέγχου, οπότεν περισσότερη ενέργεια και θα ήταν ένα κεντρικό σημείο αποτυχίας στο δίκτυο. Το πρωτόκολλο που υλοποιήσαμε αφορά εφαρμογές με **συνεχή ροή δεδομένων**. Δηλαδή, όταν σε κάποιο κόμβο ενεργοποιηθεί ένα γεγονός τότε υπάρχει συνεχής ροή δεδομένων. Σταματά (ή μειώνει) μόνο αν ο κόμβος πιάσει εντολή να σταματήσει (ή να μειώσει) τη ροή. Στα Ασύρματα Δίκτυα Αισθητήρων είναι απαραίτητο να γίνεται έλεγχος της συμφόρησης. Η συμφόρηση προκαλεί το χάσιμο των πακέτων άρα δημιουργεί σπατάλη ενέργειας, τόσο για τα πακέτα που δημιουργήθηκαν και χάθηκαν όσο και για τα πακέτα που πρέπει να δημιουργηθούν ξανά. Το πρόβλημα γίνεται εντονότερο ειδικά σε εφαρμογές με συνεχή ροή δεδομένων, αφού οι χρήστες τέτοιων εφαρμογών θα αντιμετωπίζουν διακοπές στην εφαρμογή που χρησιμοποιούν λόγω των χαμένων πακέτων.

Στα αποτελέσματα δείξαμε ότι δημιουργήσαμε ένα αλγόριθμο με τις μικρές ανάγκες σε αποθήκευση δεδομένων και με μικρό αριθμό πακέτων ανταλλαγής μεταξύ των κόμβων (όπως είδαμε στα αποτελέσματα), για να ανταποκρίνεται στις ανάγκες των Ασύρματων Δικτύων Αισθητήρων αφού οι αισθητήρες έχουν μικρό μέγεθος μνήμης. Ακόμα, δείξαμε ότι υπάρχει δικαιοσύνη σε όλους τους κόμβους και ειδικότερα σε αυτούς που βρίσκονται μακριά από το sink. Τέλος, είδαμε ότι οι κόμβοι είναι αυτόνομοι και ανεξάρτητοι και στον αλγόριθμο μας υπάρχει προσαρμοστικότητα και κλιμακωσιμότητα. Παρόλα αυτά, μπορεί

ακόμα να γίνει σύγκριση του αλγορίθμου μας με κάποιο άλλο, να υπάρχει κίνηση και προσαρμοστικότητα στους κόμβους όσο αφορά τις τιμές των μεταβλητών α , β και γ του μηχανισμού.

**ΕΛΕΓΧΟΣ ΡΟΗΣ ΓΙΑ ΑΠΟΦΥΓΗ ΣΥΜΦΟΡΗΣΗΣ ΣΕ
ΔΙΚΤΥΑ ΑΙΣΘΗΤΗΡΩΝ, ΒΑΣΙΣΜΕΝΟΣ ΣΤΟ ΜΟΝΤΕΛΟ
ΑΝΤΑΓΩΝΙΣΜΟΥ LOTKA-VOLTERRA**

Παναγιώτα Κ. Κούλλουρου

Η Διατριβή αυτή
Υποβλήθηκε προς Μερική Εκπλήρωση των
Απαιτήσεων για την Απόκτηση
Τίτλου Σπουδών Master
σε Προηγμένες Τεχνολογίες Πληροφορικής
στο
Πανεπιστήμιο Κύπρου

Συστήνεται προς Αποδοχή
από το Τμήμα Πληροφορικής
Ιούνης, 2009

ΣΕΛΙΔΑ ΕΓΚΡΙΣΗΣ

Διατριβή Master

**ΕΛΕΓΧΟΣ ΡΟΗΣ ΓΙΑ ΑΠΟΦΥΓΗ ΣΥΜΦΟΡΗΣΗΣ ΣΕ ΔΙΚΤΥΑ
ΑΙΣΘΗΤΗΡΩΝ, ΒΑΣΙΣΜΕΝΟΣ ΣΤΟ ΜΟΝΤΕΛΟ ΑΝΤΑΓΩΝΙΣΜΟΥ
ΛΟΤΚΑ-VOLTERRA**

Παρουσιάστηκε από

Παναγιώτα Κ. Κούλλουρου

Ερευνητικός Σύμβουλος

Ανδρέας Πιτσιλλίδης

Μέλος Επιτροπής

Χριστόδουλος Χριστοδούλου

Μέλος Επιτροπής

Βάσος Βασιλείου

Πανεπιστήμιο Κύπρου

Ιούνης, 2009

ΕΥΧΑΡΙΣΤΙΕΣ

Τελειώνοντας τη Διατριβή Μάστερ, θα ήθελα να ευχαριστήσω το σύμβουλο μου, Καθηγητή Δρ. Ανδρέα Πιτσιλλίδη για την πολύτιμη και ουσιαστική βοήθεια και καθοδήγησή του σε κάθε φάση της δημιουργίας της Διατριβής. Ακόμα, είμαι ευγνώμων στον κύριο Πιτσιλλίδη για όλες τις πολύτιμες υποδείξεις του σε όλη τη διάρκεια της εκπόνησης της εργασίας μου.

Επίσης, θα ήθελα να ευχαριστήσω τον υποψήφιο διδάκτορα Παύλο Αντωνίου, που στάθηκε δίπλα μου συμβουλευόντάς με και δίνοντάς μου λύσεις σε όλες τις δυσκολίες που αντιμετώπισα κατά την υλοποίηση του πρωτοκόλλου σε ένα εργαλείο δύσκολο και απαιτητικό, όπως το NS-2. Σε μεγάλα αδιέξοδα ήταν «εκεί» δίνοντάς μου τις καλύτερες συμβουλές, καθοδηγώντας με σε σωστές αποφάσεις.

Ένα μεγάλο ευχαριστώ θέλω να εκφράσω στην αδελφή μου και τους φίλους μου που όλο αυτό τον καιρό ήταν δίπλα μου και με στήριζαν με κάθε τρόπο. Τέλος, είμαι ευγνώμων στους γονείς μου, Κώστα και Μαρία για την ολόψυχη αγάπη και υποστήριξη τους όλα αυτά τα χρόνια.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

| | |
|--|----------|
| Κεφάλαιο 1 | 1 |
| 1.1 Υπόβαθρο..... | 2 |
| 1.2 Οργάνωση της Διπλωματικής Εργασίας..... | 4 |
| Κεφάλαιο 2 | 5 |
| 2.1 Εισαγωγή..... | 6 |
| 2.2 Διαφορές Δικτύων Αισθητήρων - Ad hoc δικτύων..... | 6 |
| 2.3 Εφαρμογές στα Ασύρματα Δίκτυα Αισθητήρων..... | 8 |
| 2.4 Προβλήματα και Περιορισμοί των Ασύρματων Δικτύων Αισθητήρων..... | 9 |
| 2.5 Κίνητρα και Στόχοι των Ασύρματων Δικτύων Αισθητήρων..... | 11 |
| 2.5.1 Χαρακτηριστικά των απαιτήσεων..... | 11 |
| 2.5.2 Μηχανισμοί των απαιτήσεων..... | 13 |
| 2.5.3 Στόχοι των Ασύρματων Δικτύων Αισθητήρων..... | 16 |
| 2.6 Στοιβα Πρωτοκόλλων στα Ασύρματα Δίκτυα Αισθητήρων..... | 18 |
| 2.6.1 Επίπεδο Δικτύου στα Ασύρματα Δίκτυα Αισθητήρων..... | 19 |
| 2.6.1.1 Η Ανάγκη για Πρωτόκολλα Δρομολόγησης..... | 19 |
| 2.6.1.2 Ιδιότητες των Πρωτοκόλλων Δρομολόγησης των Δικτύων Αισθητήρων..... | 20 |
| 2.6.1.3 Τα Επισημοποιημένα Πρωτόκολλα της IETF MANET (Mobile Ad-hoc Networks)..... | 21 |
| 2.6.2 Επίπεδο Μεταφοράς στα Ασύρματα Δίκτυα Αισθητήρων..... | 22 |
| 2.7 Έλεγχος Συμφόρησης (Congestion Control)..... | 25 |
| 2.7.1 Έλεγχος Συμφόρησης στα Ενσύρματα και Ασύρματα Δίκτυα..... | 26 |
| 2.7.2 Έλεγχος Συμφόρησης Στα Ασύρματα Δίκτυα Αισθητήρων..... | 27 |

| | |
|--|-----------|
| 2.7.3 Τρόποι Αντιμετώπισης της Συμφόρησης στα Ασύρματα Δίκτυα Αισθητήρων | 31 |
| 2.7.4 Έλεγχος Συμφόρησης στο Διαδίκτυο με Χρήση του Μοντέλου Lotka-Volterra .. | 34 |
| 2.7.4.1 Μοντέλο Lotka – Volterra | 35 |
| Κεφάλαιο 3..... | 40 |
| 3.1 Περιγραφή Αλγορίθμου Δρομολόγησης..... | 41 |
| 3.2 Επεξήγηση Μηχανισμού για Αποφυγή της Συμφόρησης..... | 45 |
| 3.3 Περιγραφή Υλοποίησης..... | 54 |
| Κεφάλαιο 4..... | 61 |
| 4.1 Περιβάλλον Προσομοίωσης..... | 62 |
| 4.2 Δημιουργία Δέντρου | 63 |
| 4.3 Μηχανισμός Συμφόρησης..... | 65 |
| Κεφάλαιο 5..... | 82 |
| Βιβλιογραφία | 85 |
| APPENDIX A | 88 |
| Δομή του NS-2..... | 89 |
| Κόμβοι | 90 |
| Πράκτορες..... | 92 |

ΚΑΤΑΛΟΓΟΣ ΜΕ ΠΙΝΑΚΕΣ

Πίνακας 2-1. Παράγοντες εξισώσεων

Πίνακας 4-1. Βασικές ρυθμίσεις του NS-2 για τις προσομοιώσεις

Πίνακας 4-2. Αποτελέσματα παραγόντων για τη δημιουργία δέντρου

Πίνακας 4-3. Χρονική στιγμή ενεργοποίησης κόμβων αισθητήρων

Πίνακας 4-4. Περιγραφή πειραμάτων που υλοποιήσαμε

Πίνακας 4-5. Τιμές των μεταβλητών α και β για την ψηλότερη τιμή του Packet Delivery Ratio
με χώρο προσωρινής μνήμης = 700

Πίνακας 4-6. Τιμές των μεταβλητών α και β για την ψηλότερη τιμή του Packet Delivery Ratio
με χώρο προσωρινής μνήμης = 3000

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

- Σχήμα 3-1. Το δέντρο που δημιουργείται με τον αλγόριθμο δρομολόγησης
- Σχήμα 3-2. Μια τοπολογία με τυχαία κατανομή των κόμβων
- Σχήμα 3-3. Η διαδικασία δημιουργίας του δέντρου πριν ακόμα ολοκληρωθεί
- Σχήμα 3-4. Οι κόμβοι πηγής συναγωνίζονται για τους περιορισμένους πόρους του πατέρα τους
- Σχήμα 3-5. Ο Κόμβος Σύνδεσης δημιουργεί μια δική του ροή η οποία συναγωνίζεται για την προσωρινή μνήμη του κόμβου – πατέρα του
- Σχήμα 3-6. Σχεδιάγραμμα αλγορίθμου
- Σχήμα 4-1. Δέντρο 25 κόμβων με $A=0.5$ και $B=0.5$
- Σχήμα 4-2. Καθορισμένη τοπολογία 25 κόμβων
- Σχήμα 4-3. (α) Data Delivery Ratio για το Πείραμα 1 του Πίνακα 4-4 (3 ενεργοί κόμβοι)
(β) Delay για το Πείραμα 1 του Πίνακα 4-4
- Σχήμα 4-4. Ρυθμός σύγκλισης για 3 ενεργούς κόμβους
- Σχήμα 4-5. (α) Data Delivery Ratio για το Πείραμα 4 του Πίνακα 4-4 (10 ενεργοί κόμβοι)
(β) Delay για το Πείραμα 4 του Πίνακα 4-4
- Σχήμα 4-6. Ρυθμός σύγκλισης για 10 ενεργούς κόμβους
- Σχήμα 4-7. (α) Data Delivery Ratio για το Πείραμα 6 του Πίνακα 4-4 (20 ενεργοί κόμβοι)
(β) Delay για το Πείραμα 6 του Πίνακα 4-4
- Σχήμα 4-8. Ρυθμός σύγκλισης για 20 ενεργούς κόμβους
- Σχήμα 4-9. (α) Υπολογισμός του Rate για τα Πειράματα με 3, 10 και 20 ενεργούς κόμβους
(αντίστοιχα)

(β) Bandwidth για τα Πειράματα με 3, 10 και 20 ενεργούς κόμβους (αντίστοιχα)

Σχήμα 4-10. (α) Rate για 5, 7 και 10 ενεργούς κόμβους και $r=0.5$

(β) Rate για 5, 7 και 10 ενεργούς κόμβους και $r=1$

Σχήμα 4-11. Ποσοστό πακέτων δεδομένων για 3, 5, 7, 10, και 15 ενεργούς κόμβους

Σχήμα 4-12. (α) Rate για πείραμα με τυχαία τοπολογία με 3 ενεργούς κόμβους

(β) Rate για πείραμα με τυχαία τοπολογία με 10 ενεργούς κόμβους

Σχήμα 4-13. (α) Rate όπου ο κόμβος 6 σταματά την αποστολή δεδομένων με αρχικά 3 ενεργούς κόμβους

(β) Rate όπου ο κόμβος 6 σταματά την αποστολή δεδομένων με αρχικά 10 ενεργούς κόμβους

Σχήμα Appendix A-1. Δομή του Unicast Κόμβου

Σχήμα Appendix A-2. Ιεραρχία των Πρακτόρων με βάση το Πρωτόκολλο TCP στο NS-2

Κεφάλαιο 1

Εισαγωγή

Τα τελευταία χρόνια προωθείται πολύ η χρήση μικροσκοπικών συσκευών σε όλες τις πτυχές της τεχνολογικής ανάπτυξης. Δηλαδή, απλές και χαμηλής ισχύος συσκευές που ενσωματώνουν ασύρματες τηλεπικοινωνιακές ζεύξεις υψηλής απόδοσης και βελτιωμένες μικρής κλίμακας πηγές ενέργειας έχουν συνδυαστεί με μειωμένα κόστη παραγωγής για να κάνουν ένα νέο τεχνολογικό όνειρο πραγματικότητα. Τα Ασύρματα Δίκτυα Αισθητήρων (WSNs).

Τα Ασύρματα Δίκτυα Αισθητήρων αποτελούνται από αρκετούς μικροσκοπικούς κόμβους αισθητήρων οι οποίοι αναπτύσσονται σε μια περιοχή τυχαία κατανεμημένοι. Κάθε κόμβος αισθητήρα έχει περιορισμένες δυνατότητες επεξεργασίας και επικοινωνίας, μικρή χωρητικότητα μνήμης, αλλά όταν συνεργάζεται με άλλους κόμβους ανταλλάσσοντας πληροφορίες μεταξύ τους, έχουν τη δυνατότητα να μετρούν τα δεδομένα του φυσικού περιβάλλοντος με μεγάλη λεπτομέρεια ή να εκτελούν ένα στόχο με σύνθετες λειτουργίες. Γι' αυτό το λόγο ένα δίκτυο αισθητήρων μπορεί να περιγραφεί σαν μια συλλογή από κόμβους αισθητήρων που συνεργάζονται μεταξύ τους για να εκτελούν συγκεκριμένες πράξεις. Από τη στιγμή που κάθε κόμβος είναι ενσωματωμένος πάνω σε μια πινακίδα επεξεργαστή μπορεί να χρησιμοποιεί τις δικές του δυνατότητες επεξεργασίας για να βρίσκει απλούς υπολογισμούς και να μεταφέρει μόνο τα απαραίτητα δεδομένα. Αυτές οι ιδιαιτερότητες επιτρέπουν στα

δίκτυα αισθητήρων να χρησιμοποιούνται σε διάφορες εφαρμογές όπως παρακολούθηση απομακρυσμένων αγροτικών περιοχών, ανίχνευση κίνησης οχημάτων και ομάδων στρατιωτών στο πεδίο της μάχης, έλεγχο βιομηχανικών μονάδων και εργοστασίων, περιβαλλοντικό έλεγχο, παρακολούθηση ασθενών και άλλα.

1.1 Υπόβαθρο

Τα Ασύρματα Δίκτυα Αισθητήρων έχουν κάποιες ιδιαιτερότητες που τα διαφοροποιούν από τα παραδοσιακά δίκτυα, και αναλύονται στο επόμενο Κεφάλαιο πιο διεξοδικά. Κάτω από αυτές τις συνθήκες, δε θα ήταν δυνατό να εφαρμοστούν τα πρωτόκολλα που μέχρι σήμερα χρησιμοποιούνταν στα Ασύρματα Τοπικά Δίκτυα, ή ακόμα και στα Ad hoc δίκτυα. Γιατί αρχικά, τα δίκτυα αυτά έχουν περιορισμένους πόρους, οπότε κατά τη δημιουργία ενός νέου πρωτοκόλλου πρώτιστος στόχος είναι να αποφεύγεται οτιδήποτε μπορεί να προκαλέσει κατανάλωση της ενέργειας, όπως χάσιμο πακέτων, και ανταλλαγή περιττής πληροφορίας. Επίσης, πρέπει να λαμβάνεται υπόψη ότι οι κόμβοι είναι τοποθετημένοι σε τυχαίες θέσεις και έχουν τη δυνατότητα να επικοινωνούν μόνο με τους γείτονες τους, μέσω των οποίων μπορούν να μεταφέρουν τα δεδομένα τους, μέσω άλλων κόμβων (hop-by-hop) και στους πιο απόμακρους κόμβους. Δηλαδή, δεν υπάρχει μια κεντρική οντότητα που να διαχειρίζεται όλους τους κόμβους, όλοι οι κόμβοι είναι οι ίδιοι και είναι αυτόνομοι (self-organization). Όλα αυτά μαζί με την ανάγκη για ευρωστία (robustness) και ανθεκτικότητα (resilience) είναι τα συστατικά που πρέπει να περιγράφουν ένα Ασύρματο Δίκτυο Αισθητήρων. Για να ικανοποιούνται αυτά τα χαρακτηριστικά πρέπει ένας αλγόριθμος να είναι πολύ απλός και αρκετά εξελίξιμος αφού οι κόμβοι αισθητήρων δεν μπορούν από μόνοι τους να τα ικανοποιήσουν. Στη βιολογία μπορεί κάποιος να συναντήσει τα χαρακτηριστικά που απαιτούνται στα υπολογιστικά περιβάλλοντα, όπως το Ασύρματο Δίκτυο Αισθητήρων. Δηλαδή, απλές, αποκεντρωτικές, αυτόνομες και προσαρμοστικές μέθοδοι χρησιμοποιούνται, κυρίως για επιβίωση και βελτιστοποίηση του είδους. Επίσης, η περιοριστική φύση των

δικτύων αυτών απαιτεί απλές στρατηγικές υλοποίησης σε ξεχωριστούς κόμβους με την ελάχιστη ανταλλαγή πληροφοριών, όπως και σε πολλαπλά βιολογικά συστήματα (π.χ. οικοσύστημα μυρμηγκιών - ant colonies).

Τα μαθηματικά μοντέλα πληθυσμών [5] βοηθούν στη μοντελοποίηση βιολογικών διαδικασιών χρησιμοποιώντας αναλυτικές τεχνικές και εργαλεία που χρησιμοποιούνται συχνά για τη μελέτη μη-γραμμικών συστημάτων. Η ανάπτυξη πληθυσμών είναι ένα θέμα που παραδοσιακά μελετάται από τη μαθηματική βιολογία, η οποία παρακολουθεί την αλλαγή του πληθυσμού των ειδών, σε ένα ανταγωνιστικό περιβάλλον, σε χρόνο και χώρο και τι προκαλούν αυτές οι αλλαγές. Στην υλοποίησή μας χρησιμοποιούμε ένα μοντέλο εμπνευσμένο από τη βιολογία για το σχεδιασμό ενός μηχανισμού για τον έλεγχο της συμμόρφωσης μεταξύ ανταγωνιστικών ροών, που να είναι εξελικτικό, προσαρμόσιμο και οδηγείται από τα χαρακτηριστικά που είδαμε πιο πάνω και υπάρχουν στο οικοσύστημα.

Σε αυτή τη Διπλωματική Εργασία θα δούμε τη υλοποίηση ενός αλγορίθμου δρομολόγησης, ο οποίος χρησιμοποιεί ένα μηχανισμό για αποφυγή της συμμόρφωσης. Να σημειώσουμε ότι η υλοποίηση μας αφορά ένα αλγόριθμο στον οποίο υπάρχει ανοχή σφαλμάτων (fault tolerance) και μπορεί να χαρακτηριστεί σαν μια αποκεντρωτική προσέγγιση τόσο για το κτίσιμο ενός ισορροπημένου δέντρου με κορυφή το κόμβο sink, όσο και τη ρύθμιση του ρυθμού αποστολής δεδομένων, η οποία βασίζεται στο υπολογιστικό μοντέλο Lotka-Volterra. Το μοντέλο Lotka-Volterra εγγυάται ισορροπία στη συνύπαρξη όλων των ροών δεδομένων, και σταθερότητα και δικαιοσύνη μεταξύ των ενεργών ροών όταν κάποιες συνθήκες ικανοποιούνται. Επίσης, να αναφέρουμε ότι ο αλγόριθμός μας αφορά εφαρμογές όπου απαιτείται συνεχής ροή δεδομένων πολυμέσων (multimedia streaming). Στο Κεφάλαιο 4 θα αξιολογήσουμε την υλοποίησή μας μέσα από διαφορετικά πειράματα.

Στο επόμενο Κεφάλαιο θα δούμε διάφορα πρωτόκολλα δρομολόγησης που προτάθηκαν για τα Ασύρματα Δίκτυα αισθητήρων, αλλά και διάφορους μηχανισμούς, οι οποίοι στόχο έχουν είτε την αποφυγή συμμόρφωσης, είτε να αντιμετωπίζουν τη συμμόρφωση. Επίσης, θα δούμε έναν συγγενή αλγόριθμο που δημιουργήθηκε για το Διαδίκτυο και χρησιμοποιεί το μαθηματικό μοντέλο Lotka-Volterra. Κατά τη διάρκεια της διπλωματικής αυτής εργασίας, θα

κάνουμε τη δική μας πρόταση για ένα αλγόριθμο δρομολόγησης αλλά και για ένα μηχανισμό για αποφυγή της συμφόρησης [1] που βασίζεται στο μοντέλο Lotka-Volterra.

1.2 Οργάνωση της Διπλωματικής Εργασίας

Αρχικά, στο Κεφάλαιο 2 θα μιλήσουμε γενικά για τα Ασύρματα Δίκτυα Αισθητήρων. Θα δούμε τους περιορισμούς τους, τα χαρακτηριστικά των απαιτήσεών τους, καθώς και παραδείγματα άλλων πρωτοκόλλων Δρομολόγησης και μηχανισμών για έλεγχο της συμφόρησης. Όλα αυτά αποτελούν συστατικά τα οποία είναι απολύτως απαραίτητα να γνωρίζουμε όταν θέλουμε να δημιουργήσουμε το δικό μας αλγόριθμο δρομολόγησης, ο οποίος θα κάνει και έλεγχο της συμφόρησης για τα δίκτυα αυτά. Στη συνέχεια, στο Κεφάλαιο 3 θα δούμε μια περιγραφή τόσο του αλγορίθμου δρομολόγησης όσο και του μηχανισμού που χρησιμοποιούμε για αποφυγή της συμφόρησης. Στο Κεφάλαιο 4, θα αποτιμήσουμε τα αποτελέσματα των διάφορων πειραμάτων που πήραμε κατά την προσομοίωση του αλγορίθμου για να μελετήσουμε την απόδοση της υλοποίησής μας. Τέλος, στο Κεφάλαιο 5, θα παρουσιάσουμε τα συμπεράσματα και τη μελλοντική εργασία. Στο Appendix A θα δούμε μια μικρή περιγραφή του εργαλείου (NS-2) που είναι το εργαλείο στο οποίο υλοποιήσαμε τον αλγόριθμό μας.

Κεφάλαιο 2

Σχετική Εργασία

Η γρήγορη ανάπτυξη της μικροηλεκτρονικής και των υλικών επέτρεψε την κατασκευή πολύ μικρών αισθητήρων, οι οποίοι έχουν την ικανότητα να μετρούν και να καταγράφουν μια μεγάλη σειρά από περιβαλλοντολογικά ή βιολογικά μεγέθη, όπως τη θερμοκρασία, την ατμοσφαιρική πίεση, την υγρασία, τη φωτεινότητα, τη στάθμη υδάτων, την ωρίμανση καρπών, την ανίχνευση χημικών στοιχείων, την πίεση αίματος, τους σφυγμούς καρδιάς, την κίνηση αντικειμένων και ανθρώπων και πολλές άλλες παραμέτρους. Αξιοσημείωτο είναι ότι σε μία διάταξη ίση με ένα νόμισμα 2 ευρώ μπορούμε να συμπεριλάβουμε πολλά από τα παραπάνω αισθητήρια και να καταμετρούμε συγχρόνως διάφορα μεγέθη. Επίσης, η τεχνολογία ασύρματης τεχνολογίας θέσπισε στους μικροσκοπικούς κόμβους αισθητήρων τη δυνατότητα να επικοινωνούν μεταξύ τους σε κοντινές αποστάσεις. Στο Κεφάλαιο που ακολουθεί θα δούμε τις διαφορές των Ασύρματων Δικτύων Αισθητήρων από τα Ad-Hoc Δίκτυα, θα δούμε τους περιορισμούς τους, τα χαρακτηριστικά των απαιτήσεων τους, κάποιες εφαρμογές τους καθώς και παραδείγματα άλλων πρωτοκόλλων Δρομολόγησης και μηχανισμών για έλεγχο της συμφόρησης.

2.1 Εισαγωγή

Τα Ασύρματα Δίκτυα Αισθητήρων (WSNs) αποτελούνται από ένα ή περισσότερα sink (ή base station) και από μερικές δεκάδες ή χιλιάδες κόμβους αισθητήρες (sensor nodes), οι οποίοι διασκορπίζονται σε ένα χώρο. Οι κόμβοι αυτοί συλλέγουν πληροφορίες από το περιβάλλον και ανάλογα με την εφαρμογή, είτε επεξεργάζονται τις πληροφορίες και τις στέλνουν, είτε τις στέλνουν χωρίς καμιά επεξεργασία. Οι κόμβοι αυτοί, συνήθως αισθάνονται τη θερμοκρασία, το φως, τη δόνηση, τον ήχο, την ακτινοβολία κ.α. Οι πληροφορίες αυτές “ταξιδεύουν” μέσα στο δίκτυο, από κόμβο-σε-κόμβο, έχοντας σαν τελικό προορισμό τους κόμβους sink. Ανάλογα με την εφαρμογή, τα sink ενδέχεται να αποστείλουν κάποια ερωτήματα (queries) προς τους κόμβους, με σκοπό να μαζέψουν χρήσιμες πληροφορίες.

2.2 Διαφορές Δικτύων Αισθητήρων - Ad hoc δικτύων

Παρόλο που τα Ασύρματα Δίκτυα Αισθητήρων έχουν αρκετές ομοιότητες με τα ήδη υπάρχοντα Ad hoc δίκτυα, διαφέρουν σε αρκετές και συγκεκριμένες ιδιότητες. Κάποια σημαντικά σημεία που κάνουν τα Ασύρματα Δίκτυα Αισθητήρων να διαφέρουν θα αναφέρουμε πιο κάτω:

- **Application specific.** Στα Ασύρματα Δίκτυα Αισθητήρων υπάρχουν διάφοροι τύποι εφαρμογών. Αυτό συμβαίνει λόγω των διαφορετικών εφαρμογών των αισθητήρων (αραιή ή πυκνή τοπολογία, συνεχής ή διακριτή κίνηση), λόγω των πολλών δυνατών συνδυασμών των αισθητήρων (υπολογιστικών και επικοινωνιακών) είναι δυνατό να προκύψουν πολλά διαφορετικά σενάρια για τα οποία θα απαιτούνται διαφορετικά πρωτόκολλα ή πρωτόκολλα που θα προσαρμόζονται ανάλογα.
- **Environment interaction.** Τα Ασύρματα Δίκτυα Αισθητήρων δέχονται επίδραση από το περιβάλλον και έτσι η κίνηση στα δίκτυα (συνεχής ή διακριτή) αναμένεται να διαφέρει από τα δίκτυα όπου η κίνηση καθορίζεται από τον άνθρωπο.

- **Scale.** Τα Ασύρματα Δίκτυα Αισθητήρων έχουν περισσότερες δυνατότητες για κλιμακωτή ανάπτυξη σε μεγαλύτερους αριθμούς (χιλιάδες ή ακόμα και εκατοντάδες χιλιάδες) από οντότητες από τα τρέχοντα δίκτυα ad hoc, τα οποία χρειάζονται διαφορετικές, πιο κλιμακωτές λύσεις.
- **Energy.** Αντίθετα με τα δίκτυα ad hoc όπου ο ανεφοδιασμός ενέργειας μπορεί να γίνει εύκολα, στα Ασύρματα Δίκτυα Αισθητήρων ο ανεφοδιασμός είναι σπάνιος και γι' αυτό η κατανάλωση ενέργειας είναι κύρια μετρική που πρέπει να λαμβάνεται υπόψη. Συχνά, η μπαταρία στους αισθητήρες των Ασύρματων Δικτύων Αισθητήρων είναι μη επαναφορτιζόμενη και η ανάγκη για να παρατείνουμε τη ζωή ενός κόμβου αισθητήρα έχει μεγάλη επίδραση στο σύστημα και στην αρχιτεκτονική του δικτύου.
- **Self configurability.** Σχεδόν παρόμοια με τα δίκτυα ad hoc, τα δίκτυα ασύρματων αισθητήρων είναι πιθανό να απαιτούν self-configure σε ένα συνδεδεμένο δίκτυο αλλά η διαφορά στην κίνηση και στο συμβιβασμό για την ενέργεια θα απαιτούν καινούριες λύσεις.
- **Dependability and QoS.** Αυτά τα δίκτυα θα επιδείξουν μεγάλες διαφορές στις ιδέες για την αξιοπιστία και την ποιότητα υπηρεσίας. Σε μερικές περιπτώσεις γίνεται περιστασιακή μεταφορά πακέτων (όποτε αισθανθούν κάποιο πακέτο στο περιβάλλον), ενώ σε κάποιες άλλες περιπτώσεις η μεταφορά πακέτων είναι συνεχής (streaming). Σε κάποιες απαιτείται να υπάρχει υψηλή αξιοπιστία, ενώ σε άλλες όχι.
- **Data centric.** Η σημαντικότητα κάποιου συγκεκριμένου κόμβου μειώνεται σημαντικά σε σύγκριση με τα παραδοσιακά δίκτυα, όπου κάποιος χρήστης θέλει να συνδέσει τον κινητό του υπολογιστή με ένα συγκεκριμένο εξυπηρετητή web. Το πιο σημαντικό είναι τα δεδομένα που μπορούν αυτοί οι κόμβοι να παρατηρήσουν. Αυτό μετακινεί τη σημαντικότητα από την αρχιτεκτονική που έχει τον κόμβο σαν κέντρο, στην αρχιτεκτονική όπου το κέντρο είναι τα δεδομένα.
- **Simplicity.** Εφόσον οι κόμβοι αισθητήρων είναι μικροί και η ενέργεια τους λιγοστή, το λειτουργικό και το δικτυακό λογισμικό πρέπει να είναι πολύ απλό σε σύγκριση με

τους σημερινούς υπολογιστές. Αυτή η απλότητα, πρέπει επίσης να σπάξει τα συμβατικά επίπεδα του δικτυακού λογισμικού, εφόσον η αφαιρετικότητα συνήθως κοστίζει χρόνο και χώρο.

Επίσης, τα Ασύρματα Δίκτυα Αισθητήρων έχουν ακόμα κάποιες διαφορές με δίκτυα ad hoc, όπως το ότι οι κόμβοι αισθητήρων είναι αρκετά μεγαλύτεροι σε αριθμό, έχουν περιορισμένη δύναμη, υπολογιστικές δυνατότητες και μνήμη, δεν έχουν καθολική ταυτότητα (global ID), κυρίως χρησιμοποιούν επικοινωνία με βάση την οποία η πληροφορία κοινοποιείται σε όλους (broadcast), είναι πολύ επιρρεπείς σε αποτυχίες και ακόμα, η τοπολογία του δικτύου μπορεί να αλλάζει πολύ συχνά.

2.3 Εφαρμογές στα Ασύρματα Δίκτυα Αισθητήρων

Βασίζόμενοι σε ένα τέτοιο τεχνολογικό όραμα, νέοι τύποι εφαρμογών γίνονται πιθανοί. Εφαρμογές που περιλαμβάνουν έλεγχο του περιβάλλοντος όπως η παρακολούθηση της εξέλιξης μιας φωτιάς, εγκατάσταση αισθητήρων σε γέφυρες ή κτίρια για να ελέγχονται σε περιπτώσεις σεισμού, παρακολουθήσεις για τυχόν ανεπιθύμητες παρουσίες σε κάποιο χώρο, αλλά ακόμα και ενσωμάτωση αισθητήρων βαθιά μέσα σε μηχανήματα όπου οι ενσύρματοι αισθητήρες δεν θα ήταν δυνατόν να εφαρμοστούν. Σε τέτοιες εφαρμογές οι κόμβοι αισθητήρων περιμένουν μέχρι να συμβεί κάποιο γεγονός και να στείλουν μήνυμα στο sink για να αναφέρουν το γεγονός.

Εκτός από αυτό το είδος εφαρμογών υπάρχουν και οι εφαρμογές όπου υπάρχει συνεχής ροή δεδομένων (streaming). Τέτοιου είδους εφαρμογές απαιτούνται όταν υπάρχει ανάγκη να γίνεται συνεχής παρακολούθηση (π.χ. με κάμερες) σε κάποιο κτίριο, σε δημόσια γεγονότα ή ακόμα και σε ιδιωτική περιουσία για προστασία είτε από κλέφτες, είτε από τρομοκρατικές επιθέσεις. Ακόμα, μπορεί να χρησιμοποιούνται για παρακολούθηση δρόμων σε μεγάλες πόλεις και να γίνεται η κατάλληλη ενημέρωση στους οδηγούς για να αποφεύγουν την κίνηση στους δρόμους. Τα Ασύρματα Δίκτυα Αισθητήρων μπορούν να χρησιμοποιηθούν

ακόμα και στην ιατρική με τη βοήθεια των δικτύων 3G για να προσφέρουν υπηρεσίες υγείας παντού. Δηλαδή, κάποιος ασθενής να μεταφέρει πάνω του κόμβους αισθητήρων, οι οποίοι θα κάνουν συνεχείς μετρήσεις της θερμοκρασίας τους, της πίεσής τους κ.α. και να τα στέλνουν συνεχώς σε ένα κέντρο υγείας όπου θα γίνεται απομακρυσμένη παρακολούθηση ασθενών μέσω καμερών ή ακουστικών που θα βρίσκονται τοποθετημένα πιθανώς πάνω στον καρπό του ασθενή.

Η υλοποίησή μας σε αυτή τη Διπλωματική Εργασία αφορά το δεύτερο είδος εφαρμογών. Να προσθέσουμε ότι υπάρχουν πολλές πιθανότητες τα Ασύρματα Δίκτυα Αισθητήρων να κάνουν μια αναστάτωση στην τεχνολογία, όπου τα βασικά προβλήματα του κόστους, του μεγέθους, της μεγάλης ταχύτητας αλλά και της ευλυγισίας των συστημάτων να επιλύονται. Παρόλα αυτά, όπως θα δούμε και πιο κάτω, τα ασύρματα δίκτυα αισθητήρων έχουν να αντιμετωπίσουν κάποια άλλα προβλήματα.

2.4 Προβλήματα και Περιορισμοί των Ασύρματων Δικτύων Αισθητήρων

Τα Ασύρματα Δίκτυα Αισθητήρων παρουσιάζουν ένα εντελώς διαφορετικό σύνολο από περιορισμούς σε σύγκριση με τους περιορισμούς που παρουσιάζονται στα παραδοσιακά δίκτυα. Το πιο σημαντικό από αυτά είναι η ενέργεια. Αυτά τα δίκτυα αποτελούνται από συλλογές από συσκευές που πρέπει να είναι ενεργές αρκετή ώρα με μικρές μπαταρίες. Έρευνες που έχουν γίνει τελευταίως, έδειξαν ότι εάν κόμβος τρέχει με τη μέγιστη του δύναμη, η διάρκεια ζωής του κόμβου είναι τέσσερις μέρες. Αυτές οι τέσσερις μέρες πρέπει να διασκορπιστούν σε αρκετά χρόνια ζωής. Κάτι τέτοιο, για τα υπόλοιπα συστήματα δεν είναι κάτι που πρέπει να λαμβάνουν υπόψη, αλλά για τα Ασύρματα Δίκτυα Αισθητήρων είναι κάτι πολύ σημαντικό. Το δεύτερο πολύ σημαντικό είναι η τοποθέτησή τους. Αντίθετα με τους υπόλοιπους υπολογιστές, οι αισθητήρες συχνά πρέπει να τοποθετούνται σε δύσκολες περιοχές, όπου οι τεχνικοί που θα τους εφαρμόσουν έχουν μειωμένο πεδίο ορατότητας.

Ακόμα κάτι που γίνεται δύσκολα στους κόμβους αισθητήρων είναι το να καθοριστεί για ποιο λόγο χάθηκε ένα πακέτο. Οι πιθανοί λόγοι είναι λόγω των συγκρούσεων στο επίπεδο MAC, λόγω υπερχείλισης της ουράς, λόγω έλλειψης ενέργειας, ή ακόμα λόγω έλλειψης ασφάλειας. Για παράδειγμα, κάποιος εξωγενής παράγοντας μετακίνησε τον αισθητήρα, όπως κάποιο πουλί. Ακόμα κάποιος πιθανός λόγος για το χάσιμο πακέτων είναι η συμφόρηση στο δίκτυο, επομένως κάποια πακέτα να μην μπορούν να μεταφερθούν και ειδικότερα, αυτά που βρίσκονται πιο μακριά από το sink.

Ένα ακόμα πρόβλημα που έχουν να αντιμετωπίσουν τα Ασύρματα Δίκτυα Αισθητήρων είναι το coverage. Δηλαδή, το πόσο καλά παρακολουθείται μια περιοχή από τους κόμβους αισθητήρες.

Επίσης, αντιμετωπίζουν κάποια προβλήματα λόγω των περιορισμών που υπάρχουν στους αισθητήρες, όπως για παράδειγμα οι περιορισμένοι πόροι, οι χαμηλές υπολογιστικές δυνατότητες, η μικρή μνήμη και η περιορισμένη -και πολλές φορές μη επαναφορτιζόμενη- μπαταρία, λόγω του μεγέθους τους και των περιορισμών που αναφέραμε πιο πάνω. Και το χαμηλό ασύρματο επικοινωνιακό bandwidth λόγω της ασύρματης επικοινωνίας τους.

Λόγω αυτών και άλλων περιορισμών και προβλημάτων που έχουν να αντιμετωπίσουν τα Ασύρματα Δίκτυα Αισθητήρων υπάρχει ανάγκη για απλά, καινοτόμα, συστήματα, πρωτόκολλα, και αλγορίθμους. Λόγω του ότι στα δίκτυα αυτά δεν υπάρχει η σαφής έννοια της διαστρωμάτωσης, τα πιο πάνω προβλήματα επιλύονται, σε ένα ή περισσότερα από τα κλασσικά επίπεδα (φυσικό, εφαρμογής, δικτύου κ.τ.λ.). Για παράδειγμα, ο έλεγχος συμφόρησης μπορεί να εμπεριέχεται στο επίπεδο δικτύου και μεταφοράς όπως θα μελετήσουμε στη συνέχεια.

2.5 Κίνητρα και Στόχοι των Ασύρματων Δικτύων Αισθητήρων

Τα Ασύρματα Δίκτυα Αισθητήρων, όπως είδαμε και στην προηγούμενη παράγραφο, έχουν να αντιμετωπίσουν ένα μεγάλο αριθμό από διαφορετικές εφαρμογές και αυτό δεν τους δίνει τη δυνατότητα να μπορούν να διαχειρίζονται με ένα απλό τρόπο. Εν τούτοις, ορισμένα κοινά γνωρίσματα εμφανίζονται, ειδικά όσον αφορά τα χαρακτηριστικά τους και τους μηχανισμούς που είναι απαραίτητοι σε τέτοια συστήματα. Η ικανοποίηση αυτών των χαρακτηριστικών με τους νέους μηχανισμούς είναι η σημαντικότερη πρόκληση του οράματος των Ασύρματων Δικτύων Αισθητήρων.

2.5.1 Χαρακτηριστικά των απαιτήσεων

Οι περισσότερες εφαρμογές στα Ασύρματα Δίκτυα Αισθητήρων μοιράζονται τα πιο κάτω κοινά χαρακτηριστικά [16]:

- **Τύπος Υπηρεσίας (Type of service).** Ο τύπος υπηρεσιών που δίνεται από ένα δίκτυο επικοινωνίας είναι εμφανής - κινεί τα bits από μια θέση προς μια άλλη. Για ένα Ασύρματο Δίκτυο επικοινωνίας, η κίνηση των bits είναι μόνο ένας τρόπος, ένα μέσο μεταφοράς μέχρι το τέρμα, αλλά όχι ο πραγματικός σκοπός. Ένα Ασύρματο Δίκτυο Αισθητήρων αναμένεται να παρέχει σημαντικές πληροφορίες ή/και ενέργειες για ένα στόχο, ο οποίος μεταβάλλεται ανάλογα με την εφαρμογή.
- **Ποιότητα Υπηρεσίας (Quality of service).** Πολύ σχετική με τον τύπο υπηρεσίας ενός δικτύου είναι η ποιότητα της συγκεκριμένης υπηρεσίας. Παραδοσιακά η ποιότητα των απαιτήσεων των υπηρεσιών, όπως την οριακή καθυστέρηση (bounded delay) ή το ελάχιστο εύρος ζώνης (bandwidth) είναι άσχετες όταν οι εφαρμογές είναι ανεκτικές στην καθυστέρηση ή το εύρος ζώνης των δεδομένων που πρέπει να μεταφερθούν, όταν αυτό είναι πολύ μικρό από την αρχή. Βασικά σε κάποιες εφαρμογές η περιστασιακή παράδοση ενός πακέτου μπορεί να είναι υπέρ αρκετή ενώ σε κάποιες άλλες υπάρχουν πολύ υψηλές απαιτήσεις αξιοπιστίας. Ακόμα σε κάποιες

άλλες εφαρμογές η καθυστέρηση μπορεί να είναι σημαντική. Πολλές φορές μια μετρική μόνο μπορεί να είναι ανεπαρκής.

- **Ανοχή Σφαλμάτων (Fault tolerance).** Στα Ασύρματα Δίκτυα Αισθητήρων είναι δεδομένο ότι υπάρχει μεγάλη πιθανότητα οι κόμβοι να μείνουν χωρίς ενέργεια ή να καταστραφούν, ή ακόμα η ασύρματη επικοινωνία μεταξύ δύο κόμβων μπορεί να διακοπεί μόνιμα. Γι' αυτούς τους λόγους είναι πολύ σημαντικό στα Ασύρματα Δίκτυα Αισθητήρων να υπάρχει ανοχή στην αποτυχία κάποιων κόμβων και αυτό μπορεί να επιτυγχάνεται μέσω του αλγορίθμου.
- **Χρόνος Ζωής (Lifetime).** Η διάρκεια ζωής ενός Ασύρματου Δικτύου Αισθητήρων είναι ένα πολύ σημαντικό χαρακτηριστικό για αυτό και σε πολλά σενάρια, οι κόμβοι στηρίζονται μόνο σε ένα περιορισμένο ανεφοδιασμό ενέργειας, ο οποίος υποστηρίζεται από τις μπαταρίες. Βέβαια η αντικατάσταση των μπαταριών είναι ένα δύσκολο και χρονοβόρο γεγονός γι' αυτό και γίνεται μια στροφή προς περιορισμένες πηγές ενέργειας μέσω ηλιακών κυττάρων που μπορούν να λειτουργούν σαν συμπλήρωμα στα αποθέματα ενέργειας αφού μπορούν να επαναφορτίζονται. Με αυτό τον τρόπο η διάρκεια ζωής του δικτύου γίνεται ιδανικά άπειρη.
 Να προσθέσουμε ότι η διάρκεια ζωής ενός δικτύου έχει άμεσες επιδράσεις πάνω στην ποιότητα της υπηρεσίας. Έτσι κατά το σχεδιασμό ενός πρωτοκόλλου για τα Ασύρματα Δίκτυα Αισθητήρων πρέπει η ενέργεια να έχει ένα καθοριστικό ρόλο και να αποφεύγεται ο,τιδήποτε μπορεί να προκαλέσει σπατάλη ενέργειας, όπως αποστολή επιπλέον πακέτων. Επίσης, πρέπει να υπάρχει δυνατότητα να συνεχίζει η λειτουργία του δικτύου παρά την αποτυχία κάποιων κόμβων στο δίκτυο.
- **Κλιμακωσιμότητα (Scalability).** Δεδομένου ότι ένα Ασύρματο Δίκτυο Αισθητήρων μπορεί να περιλαμβάνει έναν μεγάλο αριθμό κόμβων, οι υπάρχουσες αρχιτεκτονικές και πρωτόκολλα πρέπει να είναι σε θέση να αναπτύσσονται και σε αυτούς τους αριθμούς.
- **Wide range of densities.** Σε ένα Ασύρματο Δίκτυο Αισθητήρων ο αριθμός των κόμβων ανά περιοχή, δηλαδή η πυκνότητα του δικτύου, μπορεί να ποικίλει αρκετά,

- **Programmability.** Οι κόμβοι δεν πρέπει μόνο να υποβάλλουν σε επεξεργασία τις πληροφορίες, αλλά πρέπει να μπορούν να αντιδρούν ελαστικά στις διάφορες αλλαγές για να μπορούν να πετυχαίνουν τους στόχους τους. Οι κόμβοι αισθητήρες πρέπει να έχουν τη δυνατότητα να προγραμματίζονται και ο προγραμματισμός τους πρέπει να είναι μεταβλητός κατά τη διάρκεια της λειτουργίας τους, έτσι ώστε όταν οι νέοι στόχοι γίνονται σημαντικοί να μπορούν να επιτυγχάνονται. Ένας σταθερός τρόπος επεξεργασίας πληροφοριών είναι ανεπαρκής.
- **Maintainability.** Το σύστημα πρέπει να προσαρμόζεται δεδομένου ότι, τόσο το περιβάλλον ενός Ασύρματου Δικτύου Αισθητήρων όσο και το ίδιο, είναι πιθανό να αλλάξουν (μειωμένες μπαταρίες, κόμβοι που αποτυγχάνουν, νέοι στόχοι). Το σύστημα πρέπει να παρακολουθεί την κατάστασή του για να μπορεί να αλλάζει τις λειτουργικές παραμέτρους του ή για να μπορεί να επιλέγει τις διάφορες παραμέτρους (π.χ. για να παρέχουν τη χαμηλότερη ποιότητα όταν ο ενεργειακός πόρος γίνεται λιγιστός) για να μπορεί να διατηρείται. Θα πρέπει επίσης, να είναι ικανό να αλληλεπιδρά με τους εξωτερικούς μηχανισμούς συντήρησης για να εξασφαλίζει την εκτεταμένη λειτουργία του σε ένα απαραίτητο επίπεδο ποιότητας.

2.5.2 Μηχανισμοί των απαιτήσεων

Για να πραγματοποιήσουμε αυτές τις απαιτήσεις πρέπει να βρούμε καινοτόμους μηχανισμούς για το επικοινωνιακό δίκτυο. Όπως νέα πρωτόκολλα και νέες αρχιτεκτονικές. Ο πραγματικός στόχος είναι να βρεθούν καινούριοι μηχανισμοί που να είναι ακριβείς στην ιδιοσυγκρασία μιας δεδομένης εφαρμογής για να μπορούν να υποστηρίζουν τη συγκεκριμένη ποιότητα υπηρεσίας, τη ζωή του δικτύου και τη συντήρηση των απαιτήσεων. Από την άλλη,

αυτοί οι μηχανισμοί πρέπει να είναι αρκετά γενικοί για ένα ευρύ φάσμα εφαρμογών και να μην γίνονται για κάθε μεμονωμένη εφαρμογή απαραίτητα, γιατί αυτό θα καθιστούσε τα Ασύρματα Δίκτυα Αισθητήρων ως μια τεχνολογική έννοια που είναι οικονομικά απραγματοποίητη. Πιο κάτω θα δούμε μερικούς από τους μηχανισμούς που αποτελούν τα χαρακτηριστικά μέρη των Ασύρματων Δικτύων Αισθητήρων όπως περιγράφονται στο [16]:

- **Ασύρματη Επικοινωνία Πολλαπλών Βημάτων (Multihop wireless communication).** Η επικοινωνία multihop στα Ασύρματα Δίκτυα Αισθητήρων είναι ένα απαραίτητο συστατικό και αυτό γιατί η επικοινωνία σε πολύ μεγάλες αποστάσεις είναι δυνατή χρησιμοποιώντας πολύ μεγάλη δύναμη, η οποία είναι απαγορευτική για τέτοιου είδους δίκτυα. Έτσι η χρήση των ενδιάμεσων κόμβων μπορεί να μειώσει τη συνολική δύναμη που απαιτείται.
- **Energy-efficient operation.** Η λειτουργία για αποδοτική ενέργεια είναι μια βασική τεχνική για να υποστηρίζεται η μακροχρόνια ζωή των κόμβων αισθητήρων. Υπάρχουν διάφορες επιλογές, όπως να περιλαμβάνεται αποδοτική ενέργεια κατά τη μεταφορά δεδομένων μεταξύ δύο κόμβων ή να προσδιορίζεται η αποδοτική ενέργεια για τα ζητούμενα δεδομένα.
- **Αυτοδιαμόρφωση (Auto-configuration).** Ένα Ασύρματο Δίκτυο Αισθητήρων πρέπει να είναι σε θέση να διαμορφώνει τις περισσότερες λειτουργικές του παραμέτρους αυτόνομα και ανεξάρτητα από οποιαδήποτε εξωτερική διαμόρφωση. Για παράδειγμα, οι κόμβοι αισθητήρες πρέπει να είναι σε θέση να ανέχονται τις αποτυχίες κάποιων κόμβων, λόγω έλλειψης μπαταρίας κ.ά.
- **Συνεργασία και ενδοδιαδικτυακή επεξεργασία (Collaboration and in-network processing).** Σε μερικές εφαρμογές, ένας μόνο αισθητήρας δεν είναι ικανός να αποφασίσει εάν ένα γεγονός έχει συμβεί, οπότε πολλοί αισθητήρες συνεργάζονται για να ανιχνεύσουν ένα γεγονός και μόνο η κοινή τους απόφαση παρέχει αρκετές πληροφορίες. Οι πληροφορίες είτε επεξεργάζονται και μετά στέλνονται, είτε απλά στέλνονται στο sink όπου γίνεται όλη η επεξεργασία. Για παράδειγμα, για να καθοριστεί η υψηλότερη ή η μέση θερμοκρασία μέσα σε μια περιοχή και να

αναφερθεί η θερμοκρασία στο sink. Για να λυθεί αποτελεσματικά αυτός ο στόχος μπορεί η ανάγνωση του κάθε μεμονωμένου αισθητήρα να αθροίζεται και έτσι να μειώνεται η ποσότητα των δεδομένων που πρέπει να διαδοθούν μέσα στο δίκτυο και να βελτιώνεται και η ενεργειακή απόδοση ταυτόχρονα.

- **Data centric.** Τα παραδοσιακά δίκτυα επικοινωνίας χαρακτηρίζονται από τη μεταφορά δεδομένων μεταξύ δύο συγκεκριμένων συσκευών, όπου η κάθε μια έχει τη δική της μοναδική διεύθυνση. Στα Ασύρματα Δίκτυα Αισθητήρων οι κόμβοι αισθητήρες διασκορπίζονται σε μια περιοχή, συνήθως αρκετοί λόγω των συχνών αποτυχιών από έλλειψη μπαταρίας. Στα δίκτυα αυτά, σημαντικό είναι οι τιμές και όχι ο κόμβος που δίνει αυτή την τιμή. Σε συνέχεια με το προηγούμενο παράδειγμα, αυτό που είναι σημαντικό είναι η θερμοκρασία και όχι ο κόμβος ή οι κόμβοι που έχουν τη συγκεκριμένη θερμοκρασία.
- **Τοπικότητα (Locality).** Οι κόμβοι αισθητήρες, όπως αναφέραμε και προηγουμένως, έχουν περιορισμένους πόρους. Έτσι, για αποφυγή σπατάλης των πόρων τους οι γνώσεις τους περιορίζονται στις πληροφορίες που παίρνουν από τους γείτονες τους. Με αυτό τον τρόπο όταν προστεθούν περισσότεροι κόμβοι δε χρειάζεται επιπρόσθετη επεξεργασία στον κάθε κόμβο ξεχωριστά για να μάθει τους νέους κόμβους.
- **Εκμετάλλευση ανταλλαγμάτων (Exploit trade-offs).** Τα Ασύρματα Δίκτυα Αισθητήρων θα πρέπει να στηρίζουν σε μεγάλο βαθμό την εκμετάλλευση των διαφόρων ανταλλαγών μεταξύ των συγκρουόμενων στόχων, τόσο κατά τη διάρκεια του σχεδιασμού συστημάτων ή πρωτοκόλλων, όσο και στο χρόνο εκτέλεσης. Για παράδειγμα, οι ψηλότερες ενεργειακές δαπάνες επιτρέπουν την υψηλότερη ακρίβεια αποτελέσματος, ανταλλάζοντας μια πιο μακροχρόνια διάρκεια ζωής του δικτύου με τη διάρκεια ζωής μεμονωμένων κόμβων. Μια άλλη σημαντική ανταλλαγή είναι η πυκνότητα κόμβων: ανάλογα με την εφαρμογή, την επέκταση, και τις αποτυχίες των κόμβων κατά την εκτέλεση, η πυκνότητα του δικτύου μπορεί να αλλάξει αρκετά. Τα πρωτόκολλα θα πρέπει να χειριστούν τις πολύ διαφορετικές καταστάσεις.

Η εκμετάλλευση αυτών των μηχανισμών είναι μια πολύ σημαντική πρόκληση αφού η αναχώρηση από την ιδέα του παραδοσιακού δικτύου με τις διευθύνσεις απαιτεί νέες διεπαφές προγραμματισμού, που υπερβαίνουν την απλή σημασιολογία της συμβατικής διεπαφής υποδοχών και εισάγουν έννοιες, όπως την απαραίτητη ακρίβεια, την ανταλλαγή μεταξύ ενέργειας και ακρίβειας.

2.5.3 Στόχοι των Ασύρματων Δικτύων Αισθητήρων

Τα Ασύρματα Δίκτυα Αισθητήρων, όπως είπαμε και προηγουμένως, έχουν πολύ διαφορετικό τρόπο λειτουργίας από τα παραδοσιακά δίκτυα. Αυτό οδηγεί και σε διαφορετικές λύσεις που πρέπει να δοθούν ανάλογα και με τις διαφορετικές εφαρμογές. Από όλα αυτά προκύπτουν τα εξής ερωτήματα: πώς να βελτιστοποιηθεί ένα δίκτυο, πώς να αποφασίσει ποια προσέγγιση υποστηρίζει καλύτερα μια δεδομένη εφαρμογή, και πώς να μετατρέψει τους σχετικά ανακριβείς στόχους βελτιστοποίησης σε μετρήσιμους αριθμούς αξίας; Μια γενική απάντηση φαίνεται αδύνατη για τα πιο πάνω ερωτήματα, όμως εξετάζοντας τη μεγάλη ποικιλία των πιθανών εφαρμογών, μερικές πτυχές είναι αρκετά εμφανείς. Στη συνέχεια, θα δούμε μόνο μερικά παραδείγματα εφαρμογών για τους παράγοντες που αφορούν τους στόχους βελτιστοποιήσεων των Ασύρματων Δικτύων Αισθητήρων.

Τα δίκτυα αυτά ουσιαστικά κινούν τα bits από μια θέση προς άλλη. Για την **ποιότητα υπηρεσίας** που προσφέρουν (QoS) υπάρχουν πρόσθετες απαιτήσεις. Η ποιότητα υπηρεσίας μπορεί να θεωρηθεί είτε ως χαμηλού επιπέδου, το οποίο παρατηρείται από τις συσκευές δικτύου και σχετίζεται με το εύρος ζώνης, την καθυστέρηση, το ποσοστό απώλειας πακέτων είτε υψηλού επιπέδου, το οποίο παρατηρείται από το χρήστη και σχετίζεται με την υποκειμενική ιδιότητα όπως την αντιληπτή ποιότητα μιας μετάδοσης φωνής ή μιας τηλεοπτικής μετάδοσης. Ενώ το πρώτο είδος ιδιοτήτων ισχύει ως ένα βαθμό στα Ασύρματα Δίκτυα Αισθητήρων, για παράδειγμα το εύρος ζώνης είναι αρκετά ασήμαντο, το δεύτερο είδος δεν είναι σαφές, αλλά είναι αρκετά σημαντικό. Παρόλα αυτά, οι υψηλού επιπέδου ιδιότητες που αντιστοιχούν στις υποκειμενικές ιδιότητες είναι κάτι που απαιτείται στα

Ασύρματα Δίκτυα Αισθητήρων. Αλλά ακριβώς όπως στα παραδοσιακά δίκτυα, οι ιδιότητες αυτές για την ποιότητα υπηρεσίας στα Ασύρματα Δίκτυα Αισθητήρων εξαρτώνται ιδιαίτερα από την εφαρμογή. Για παράδειγμα, για μια εφαρμογή που ανιχνεύει γεγονότα και τα αναφέρει, η μη υποβολή εκθέσεων ενός συναγερμού πυρκαγιάς σε έναν σταθμό επιτήρησης θα ήταν μια αυστηρή ανεπάρκεια.

Σε ένα μεγάλο μέρος της μελέτης αυτής, είδαμε ότι η **ενέργεια** είναι ένας πολύτιμος πόρος στα Ασύρματα Δίκτυα Αισθητήρων και ότι η ενεργειακή αποδοτικότητα πρέπει επομένως να κάνει έναν εμφανή στόχο βελτιστοποίησης. Είναι σαφές ότι με ένα αυθαίρετο ποσό ενέργειας, οι περισσότερες από τις μετρικές της ποιότητας υπηρεσίας μπορούν να αυξηθούν. Ως εκ τούτου, για την κατανόηση της ενεργειακής αποδοτικότητας χρειάζεται η ποιότητα υπηρεσίας που παραδίδεται και η ενέργεια που απαιτείται. Ο όρος "ενεργειακή αποδοτικότητα" είναι στην πραγματικότητα, ένας όρος που ποικίλει για διαφορετικές πτυχές ενός συστήματος, το οποίο πρέπει να διακριθεί προσεκτικά για να διαμορφώσει τους πραγματικούς, μετρήσιμους αριθμούς της αξίας. Για παράδειγμα, μια από τις πιο συνηθισμένες πτυχές είναι η ενέργεια που καταναλώνεται ανά κάθε σωστά παραλαμβανόμενο bit. Αυτή η μετρική είναι συχνά χρήσιμη για τις περιοδικές εφαρμογές ελέγχου. Προφανώς, όσο περισσότερος είναι αυτός ο χρόνος τόσο πιο καλά εκτελείται ένα δίκτυο. Σε πολλά Ασύρματα Δίκτυα Αισθητήρων και πιο συγκεκριμένα σε πρωτόκολλα, δεδομένου ότι τείνουν να θυσιάσουν τη μακροχρόνια διάρκεια ζωής σε αντάλλαγμα για μια βελτίωση στη σύντομη διάρκεια ζωής.

Η δυνατότητα να διατηρηθούν τα χαρακτηριστικά απόδοσης ανεξάρτητα από το μέγεθος του δικτύου αναφέρεται ως **εξελιξιμότητα** (scalability). Τα Ασύρματα Δίκτυα Αισθητήρων που πιθανό να αποτελούνται από χιλιάδες κόμβους, η εξελιξιμότητα είναι προφανώς μια αναπόφευκτη απαίτηση. Η εξελιξιμότητα πρέπει να εξυπηρετείται από οποιοδήποτε κατασκευάσμα που απαιτεί μια συνολικά συνεπής κατάσταση, όπως οι διευθύνσεις ή οι επιτραπέζιες καταχωρήσεις δρομολόγησης που πρέπει να διατηρηθούν. Ως εκ τούτου, η ανάγκη να περιοριστούν τέτοιες πληροφορίες επιβάλλεται μαζί με τους περιορισμούς των πόρων των κόμβων αισθητήρων, ειδικά όσον αφορά τη μνήμη. Οι

αρχιτεκτονικές και τα πρωτόκολλα πρέπει να εφαρμόσουν την «κατάλληλη» υποστήριξη εξελιξιμότητας.

Τα Ασύρματα Δίκτυα Αισθητήρων πρέπει να εκθέτουν μια κατάλληλη **ευρωστία**, αφού αυτή είναι σχετική με την ποιότητα υπηρεσίας και επίσης, με τις απαιτήσεις εξελιξιμότητας. Τα δίκτυα αυτά δεν πρέπει να αποτύχουν επειδή ένας περιορισμένος αριθμός κόμβων έχουν έλλειψη ενέργειας, ή επειδή το περιβάλλον τους αλλάζει και χωρίζει τις υπάρχουσες ράδιο συνδέσεις μεταξύ δύο κόμβων. Εάν είναι δυνατόν, αυτές οι αποτυχίες πρέπει να αντισταθμιστούν, για παράδειγμα, με την εύρεση άλλων διαδρομών. Μια ακριβής αξιολόγηση της ευρωστίας είναι δύσκολη στην πράξη και εξαρτάται συνήθως από τα μοντέλα αποτυχίας, και για τους κόμβους αλλά και για τις συνδέσεις επικοινωνίας.

2.6 Στοιβά Πρωτοκόλλων στα Ασύρματα Δίκτυα Αισθητήρων

Στα Ασύρματα Δίκτυα Αισθητήρων δεν ακολουθείται το μοντέλο της παραδοσιακής στοιβάς πρωτοκόλλων OSI (Open Systems Interconnection Basic Reference Model) με το σαφή διαχωρισμό των επιπέδων και των πρωτοκόλλων του κάθε επιπέδου. Ο λόγος που παρατηρείται αυτό στα Ασύρματα Δίκτυα Αισθητήρων είναι επειδή οι περιορισμοί και οι ανάγκες των δικτύων αυτών, όπως τους είδαμε και σε προηγούμενο Κεφάλαιο, δεν μπορούν να ανταποκριθούν σε πρωτόκολλα που θα ακολουθούσαν το σαφή διαχωρισμό των επιπέδων και αυτό φαίνεται σε διάφορα πρωτόκολλα για έλεγχο της συμμόρφωσης που δημιουργήθηκαν βασισμένα στο διαχωρισμό των επιπέδων όπως και στο [30], όπου ο έλεγχος της συμμόρφωσης υλοποιείται στο επίπεδο Μεταφοράς.

Να αναφέρουμε ότι έχουν υλοποιηθεί και κάποια πρωτόκολλα όπου δεν υπάρχει ο διαχωρισμός των στρωμάτων αλλά χρησιμοποιούν το cross-layer interaction. Να αναφέρουμε ότι το cross-layer interaction χρησιμοποιείται ευρέως στα Δίκτυα Αισθητήρων [19] και συχνά αντιμετωπίζεται σαν αναγκαιότητα [10]. Επιπρόσθετα με το cross-layer interaction δίνεται η

δυνατότητα να γίνεται χειρισμός των ειδικών ιδιοτήτων των Ασύρματων Δικτύων Αισθητήρων, οι οποίες δεν μπορούν να διαχειρίζονται πλήρως από την αρχιτεκτονική όπου υπάρχει αυστηρός καθορισμός των επιπέδων. Επομένως, στα Δίκτυα Αισθητήρων η παραδοσιακή τεχνική καθορισμένων επιπέδων αντικαταστάθηκε από την αρχιτεκτονική που βασίζεται στα συστατικά (component based), όπου το cross-layer interaction υλοποιείται ευκολότερα. Στη συνέχεια θα δούμε αναλυτικότερα το ‘επίπεδο’ μεταφοράς και το ‘επίπεδο’ Δικτύου.

2.6.1 Επίπεδο Δικτύου στα Ασύρματα Δίκτυα Αισθητήρων

Όπως αναφέραμε και πιο πάνω, στα Ασύρματα Δίκτυα Αισθητήρων για να φτάσει ένα πακέτο στον προορισμό του μπορεί να «ταξιδέψει» μέσα από διάφορους κόμβους αισθητήρων. Ο κυριότερος στόχος ενός πρωτοκόλλου δρομολόγησης είναι να εγκαθιδρύσει και να συντηρεί ένα πίνακα δρομολόγησης, ο οποίος θα περιέχει πληροφορίες για το που πρέπει να προωθηθούν τα πακέτα για να φτάσουν τον προορισμό τους που συνήθως είναι το sink. Στη συνέχεια, θα δούμε γιατί χρειάζονται τα πρωτόκολλα δρομολόγησης και επίσης, θα αναφέρουμε κάποια πρωτόκολλα δρομολόγησης ad-hoc.

2.6.1.1 Η Ανάγκη για Πρωτόκολλα Δρομολόγησης

Η δρομολόγηση δεν είναι καινούριο ζήτημα στα Δίκτυα. Τα πρωτόκολλα δρομολόγησης link-state, όπως το OSPF ([21]) και distance vector, όπως το RIP ([14,15]) χρησιμοποιούνται για μεγάλα χρονικά διαστήματα στα Δίκτυα (Διαδίκτυο). Παρόλα αυτά, τα πρωτόκολλα αυτά δεν μπορούν να χρησιμοποιηθούν στα Ασύρματα Δίκτυα Αισθητήρων για διάφορους λόγους, όπως οι πιο κάτω: Τα πρωτόκολλα αυτά σχεδιάστηκαν για στατικές τοπολογίες, πράγμα που δεν ισχύει στα Ασύρματα Δίκτυα Αισθητήρων, αφού υπάρχουν συχνές αλλαγές στην τοπολογία των αισθητήρων. Επίσης, τα πρωτόκολλα αυτά σχεδιάστηκαν για wired δίκτυα και σε αυτά τα δίκτυα υποθέτεται ότι είναι διπλή κατεύθυνσης (bidirectional). Στα Ασύρματα Δίκτυα Αισθητήρων αυτό δεν ισχύει πάντα αφού στους

κόμβους αισθητήρων η τεχνολογία που χρησιμοποιείται για να εκπέμπονται τα σήματα μπορεί να επιτρέπει μόνο uni-directional κανάλια, τα οποία μπορούν να στέλνουν δεδομένα ή να παραλαμβάνουν, δηλαδή μόνο από τη μια κατεύθυνση. Ακόμα κάτι που γίνεται στα πρωτόκολλα αυτά είναι να συντηρούν τις διαδρομές προς όλες τις κατευθύνσεις. Στα Ασύρματα Δίκτυα Αισθητήρων με το μεγάλο αριθμό κόμβων αυτό μπορεί να είναι ακατόρθωτο λόγω του μεγάλου αριθμού των πλειάδων που θα πρέπει να συντηρούνται στον πίνακα δρομολόγησης και επίσης, αυτό θα προκαλεί αχρείαστη κατανάλωση της ενέργειας των αισθητήρων που είναι τόσο σημαντική για τους κόμβους. Επομένως, φτάνουμε στην ανάγκη για ειδικά πρωτόκολλα δρομολόγησης για τα Ασύρματα Δίκτυα Αισθητήρων στα οποία θα λαμβάνονται υπόψη οι ιδιότητες των δικτύων αυτών όπως τις είδαμε και πιο πάνω.

2.6.1.2 Ιδιότητες των Πρωτοκόλλων Δρομολόγησης των Δικτύων Αισθητήρων

Όπως είδαμε και πιο πάνω τα χαρακτηριστικά των Δικτύων Αισθητήρων δημιούργησαν την ανάγκη για πρωτόκολλα δρομολόγησης ειδικά για αυτά τα Δίκτυα. Σύμφωνα με το [22], μερικές από τις ιδιότητες που πρέπει να έχουν τα πρωτόκολλα δρομολόγησης είναι τα εξής:

- **Κατανεμημένη (Distributed) Λειτουργία:** Αυτό είναι ένα δομικό στοιχείο για τα Ασύρματα Δίκτυα Αισθητήρων. Εφόσον τα δίκτυα αυτά δεν έχουν μια καθορισμένη υποδομή η λειτουργία τους πρέπει να είναι δυναμική, κατανεμημένη και αποκεντρωμένη.
- **Αποφυγή Βρόγχων Επανάληψης (Loop-freedom):** Αυτή είναι μια ιδιότητα που είναι επιθυμητή να υπάρχει αλλά συνήθως δεν εφαρμόζεται αυστηρώς. Η εξασφάλιση της αποφυγής των βρόγχων επανάληψης βοηθά στην αποφυγή των χειρότερων σεναρίων, όπως πακέτα να βρίσκονται στο δίκτυο για ακαθόριστο χρόνο.
- **Demand-based (reactive) Λειτουργία:** Αντί να συντηρούνται οι πίνακες δρομολόγησης με πληροφορίες για όλες τις κατευθύνσεις μεταξύ όλων των κόμβων στο Δίκτυο, το πρωτόκολλο δρομολόγησης αναζητεί τις κατευθύνσεις μόνο όταν χρειάζονται. Με αυτό τον τρόπο αποφεύγεται η άσκοπη χρήση του δικτύου.

- **Proactive Λειτουργία:** Αυτή είναι η αντίθετη λειτουργία από την reactive. Αν το κόστος για αναζήτηση σε on demand-based δεν είναι αποδεκτό για μια εφαρμογή, και αν οι πόροι του δικτύου, όπως το εύρος ζώνης και οι κόμβοι, όπως η περιορισμένη ενέργεια το επιτρέπουν, η proactive λειτουργία προτιμάται.
- **Ασφάλεια:** Είναι επιθυμητό το πρωτόκολλο δρομολόγησης να έχει τη δυνατότητα να χρησιμοποιεί τεχνικές ασφάλειας για να αποφεύγεται η διάσπαση ή η διαφοροποίηση της λειτουργίας του πρωτοκόλλου.
- **Λειτουργία Περιόδου Ανάπαυσης (Sleep period):** Επειδή συνήθως οι κόμβοι στα Ασύρματα Δίκτυα Αισθητήρων έχουν περιορισμένη ενέργεια, είναι καλό τα πρωτόκολλα δρομολόγησης να έχουν τη δυνατότητα να αντιμετωπίζουν καταστάσεις όπου οι κόμβοι προσωρινά θα σβήνουν με σκοπό να διαφυλάξουν ενέργεια.
- **Υποστήριξη για Μονής Κατεύθυνσης Κανάλια:** Στα Ασύρματα Δίκτυα Αισθητήρων δεν υπάρχει εγγύηση ότι τα κανάλια είναι διπλής κατεύθυνσης και έτσι γίνεται η χρήση ξεχωριστών καναλιών μονής κατεύθυνσης και για τις δύο κατευθύνσεις σε τέτοια περίπτωση.

Στη συνέχεια, θα δούμε κάποια πρωτόκολλα που χρησιμοποιούνται στα MANET, εφόσον τα δίκτυα αυτά να έχουν κάποιες ομοιότητες και ιδιότητες με τα πρωτόκολλα που χρειάζεται να κτιστούν για τα Ad-hoc δίκτυα.

2.6.1.3 Τα Επισημοποιημένα Πρωτόκολλα της IETF MANET (Mobile Ad-hoc Networks)

Οι προδιαγραφές για πολλά υπάρχοντα πρωτόκολλα δρομολόγησης σε MANET έχουν αναπτυχθεί από την IETF (Internet Engineering Task Force), η οποία επικεντρώνει την προσοχή της στα MANET [20]. Επίσης, σκοπός της ομάδας αυτής είναι να επισημοποιεί ένα ή περισσότερα intra-domain unicast πρωτόκολλα δρομολόγησης. Μέχρι τώρα έχουν δημοσιεύσει τα πιο κάτω εννέα πρωτόκολλα τα οποία θα αφαιρούμε χωρίς περαιτέρω επεξήγηση αφού αφορούν τα MANET και όχι τα Ασύρματα Δίκτυα Αισθητήρων:

- AODV (Ad hoc On-demand Distance Vector) [24]
- DSR (Dynamic Source Routing) [15]
- ZRP (Zone Routing Protocol) [13]
- OLSR (Optimized Link State Routing) [6]
- LANMAR (Landmark Routing Protocol) [8]
- FSR (Fisheye State Routing) [9]
- IERP (Interzone Routing Protocol) [17]
- IARP (Intrazone Routing Protocol) [12]
- TBRPF (Topology Broadcast based on Reverse-Path Forwarding) [23]

2.6.2 Επίπεδο Μεταφοράς στα Ασύρματα Δίκτυα Αισθητήρων

Στη μελέτη αυτή, θα δώσουμε ιδιαίτερη σημασία, στο μηχανισμό του ‘επιπέδου’ μεταφοράς των Ασύρματων Δικτύων Αισθητήρων για έλεγχο της συμφόρησης, αφού θεωρείται από πολλούς ερευνητές αναγκαίο για την αξιόπιστη μεταφορά δεδομένων και διατήρηση ενέργειας.

Γενικά, τα κλασσικά πρωτόκολλα μεταφοράς συμπεριλαμβάνουν δύο πολύ σημαντικές λειτουργικότητες: τον έλεγχο συμφόρησης και την αξιόπιστη μεταφορά μέσω ανάκτησης χαμένων πακέτων (loss recovery). Για τον έλεγχο συμφόρησης, αρχικά αυτό που πρέπει να γίνει είναι το πώς εντοπίζεται αν έχει γίνει συμφόρηση ή όχι, και το πότε και πού έχει γίνει. Στο Διαδίκτυο αυτό που γίνεται για την αντιμετώπιση της συμφόρησης είναι να πετάγονται (drop) τα πακέτα στο σημείο που έγινε η συμφόρηση και η αυξομείωση της ροής με την βοήθεια αλγορίθμων όπως AIMD στο TCP. Στα Ασύρματα Δίκτυα Αισθητήρων, αυτή η μέθοδος με την οποία θα εντοπιστεί και στην συνέχεια, αντιμετωπιστεί η συμφόρηση πρέπει να μελετηθεί πολύ προσεκτικά γιατί τα Ασύρματα Δίκτυα Αισθητήρων έχουν περιορισμένους πόρους. Τα πρωτόκολλα αυτά πρέπει να χαρακτηρίζονται από απλότητα και δυνατότητα ανάπτυξης για να αποθηκεύουν ενέργεια και παρατείνουν έτσι τη ζωή όλου του δικτύου.

Το χάσιμο πακέτων στα Ασύρματα Δίκτυα Αισθητήρων είναι σύνηθες φαινόμενο λόγω της κακής ποιότητας του ασύρματου καναλιού, της αποτυχίας των αισθητήρων ή και της συμφόρησης. Όπως μας είναι γνωστό και από το Διαδίκτυο, ανάλογα με τις εφαρμογές υπάρχουν και διαφορετικές ανάγκες. Κάποιες εφαρμογές μπορεί να χρειάζονται εγγύηση για την ανάκτηση χαμένων πακέτων, αξιόπιστη μεταφορά του κάθε πακέτου ή ακόμα και αξιόπιστη μεταφορά όλων των πακέτων. Αρχικός στόχος θα είναι να ανιχνευτεί το χάσιμο κάποιου πακέτου, έτσι ώστε να είναι δυνατή η αντικατάστασή του. Στα Ασύρματα Δίκτυα Αισθητήρων είναι δυνατή η ανίχνευση χαμένων πακέτων με τις ίδιες μεθόδους που χρησιμοποιούνται στο Διαδίκτυο, αλλά η υλοποίηση δεν είναι τόσο εύκολη. Για παράδειγμα, κάθε πακέτο μπορεί να έχει σειριακό αριθμό (sequence number) και ο παραλήπτης να ανακαλύψει το χάσιμο κάποιου πακέτου από τη μη συνέχεια των αριθμών και με τη βοήθεια των ACK ή NACK να αντικαταστήσει το χαμένο πακέτο. Η χρήση αριθμών και επιπλέον πακέτων δεν είναι επιθυμητό γεγονός στα Ασύρματα Δίκτυα Αισθητήρων, έστω και εάν η πληροφορία αυτή στέλνεται μέσω άλλων μηνυμάτων (piggyback). Εδώ αξίζει να σημειωθεί ότι στα Ασύρματα Δίκτυα Αισθητήρων δεν υπάρχει η έννοια του origin-destination ζευγαριών και επίσης, οι μοναδικές διευθύνσεις (ειδικά του τύπου A) δεν υπάρχουν όπως υπάρχει στο Διαδίκτυο.

Η ποιότητα υπηρεσίας (QoS) είναι ακόμα ένας μηχανισμός που υλοποιείται στο πρωτόκολλο μεταφοράς του Διαδικτύου. Η ποιότητα υπηρεσίας εξασφαλίζει συγχρονισμό και δικαιοσύνη. Το μεγαλύτερο πρόβλημα που έχουν να αντιμετωπίσουν τα Ασύρματα Δίκτυα Αισθητήρων κατά το σχεδιασμό πρωτοκόλλων μεταφοράς είναι ο αποδοτικός έλεγχος συμφόρησης, η εξασφάλιση αξιοπιστίας, η ποιότητα υπηρεσίας και ταυτόχρονα η διαφύλαξη ενέργειας.

Στη συνέχεια, θα δούμε δύο πολύ γνωστά πρωτόκολλα στο Διαδίκτυο, το TCP και το UDP, και ειδικότερα θα πούμε γιατί αυτά τα δύο πρωτόκολλα είναι κακή επιλογή για τα Ασύρματα Δίκτυα Αισθητήρων. Αρχικά να δούμε τα χαρακτηριστικά του TCP:

- Το TCP πρωτόκολλο είναι connection-oriented και πριν να ξεκινήσουν να στέλνονται δεδομένα υπάρχει μια διαδικασία, 3-way handshake, όπως ονομάζεται κατά την οποία

ανταλλάσσονται 3 μηνύματα μεταξύ του αποστολέα και του παραλήπτη, για να εγκαθιδρυθεί με αυτό τον τρόπο η συνομιλία τους. Στα Ασύρματα Δίκτυα Αισθητήρων τα δεδομένα που αισθάνονται οι αισθητήρες είναι μερικά bytes ή ακόμα και μια τιμή. Η εγκαθίδρυση της επικοινωνίας θα προκαλέσει η ίδια υπερφόρτωση στο δίκτυο για την αποστολή πολύ μικρών δεδομένων. Επίσης, στα ασύρματα δίκτυα ο χρόνος για την εγκαθίδρυση επικοινωνίας είναι πολύ μεγάλος και έτσι τα δεδομένα θα είναι ξεπερασμένα (outdated) μετά την εγκατάσταση της επικοινωνίας. Επίσης η έννοια της καθολικής διεύθυνσης IP πιθανώς να μην υπάρχει στα Ασύρματα Δίκτυα Αισθητήρων.

- Στο TCP χρησιμοποιείται επικοινωνία μεταξύ αποστολέα και παραλήπτη ενώ στα Ασύρματα Δίκτυα Αισθητήρων είναι πολλοί προς ένα. Δηλαδή οι κόμβοι αισθητήρες, που είναι πολλοί σε αριθμό στέλνουν τα δεδομένα τους σε ένα κοινό sink.
- Στο TCP πρωτόκολλο, υποθέτουμε ότι όλα τα πακέτα που χάνονται είναι λόγω συμφόρησης, και οι μηχανισμοί αντιμετώπισης είναι έλεγχος ροής (flow) και συμφόρησης. Αν το TCP πρωτόκολλο εφαρμοστεί στα Ασύρματα Δίκτυα Αισθητήρων, θα έχει σαν συνέπεια να μειώνεται ο ρυθμός αποστολής δεδομένων ακόμα και σε περιπτώσεις που το χάσιμο πακέτων προκλήθηκε λόγω bit-error. Η κατάσταση αυτή θα οδηγήσει σε χαμηλό throughput επηρεάζοντας περισσότερο τα sensors που βρίσκονται μακριά από τα sinks, κάνοντάς τα μη ικανά να υποστηρίξουν εφαρμογές με συνεχή ροή δεδομένων.
- Το TCP πρωτόκολλο χρησιμοποιεί end-to-end τεχνική για να ελέγχει τη συμφόρηση και αυτό απαιτεί μεγάλο χρόνο ανταπόκρισης όταν υπάρχει συμφόρηση, με αποτέλεσμα πακέτα να χάνονται. Για τα Ασύρματα Δίκτυα Αισθητήρων αυτό σημαίνει χάσιμο ενέργειας και μεγάλο χρόνο απόκρισης μέχρι να γεμίσει ξανά το ασύρματο κανάλι μετά τη συμφόρηση.
- Το TCP πρωτόκολλο εξασφαλίζει αξιοπιστία στην αποστολή δεδομένων με τα ACK και επαναμετάδοση δεδομένων (retransmission). Με αυτό τον τρόπο το throughput θα γίνει πιο χαμηλό και θα χρειάζεται μεγαλύτερο χρόνο αποστολής των δεδομένων,

ειδικότερα για μεγάλα Ασύρματα Δίκτυα Αισθητήρων, αφού ο αποστολέας δεδομένων θα πρέπει να σταματά και να περιμένει για το ACK για κάθε αποστολή.

- Στα Ασύρματα Δίκτυα Αισθητήρων τα sensors έχουν διαφορετικό αριθμό hops από το sink και το TCP θα είναι άδικο σε τέτοιο περιβάλλον. Δηλαδή, τα sensors που είναι πιο κοντά στο sink θα έχουν περισσότερες ευκαιρίες να στείλουν δεδομένα με αποτέλεσμα να σπαταλήσουν γρηγορότερα την ενέργειά τους και να προκαλέσουν το διαχωρισμό ολόκληρου του δικτύου.

Σε αντίθεση με το TCP το UDP είναι connection-less πρωτόκολλο. Όπως το TCP έτσι και αυτό το πρωτόκολλο δεν είναι κατάλληλο για TCP για τους εξής λόγους:

- Το UDP δεν κάνει έλεγχο για συμφόρηση και ροής δεδομένων και αν χρησιμοποιηθεί στα Ασύρματα Δίκτυα Αισθητήρων θα προκαλέσει το χάσιμο πολλών πακέτων όταν γίνει συμφόρηση.

Κανένα από τα δύο πρωτόκολλα δεν είναι κατάλληλο για WSNs. Να προσθέσουμε ότι, στα Ασύρματα Δίκτυα Αισθητήρων τα κατώτερα επίπεδα αναμένεται να δίνουν στο πρωτόκολλο μεταφοράς βοηθητικές πληροφορίες με αποτέλεσμα να βελτιστοποιεί την απόδοση του συστήματος.

2.7 Έλεγχος Συμφόρησης (Congestion Control)

Αρχικά να πούμε, ότι ο έλεγχος συμφόρησης στο TCP και στο ISO/OSI μοντέλο είναι ένας μηχανισμός που υλοποιείται στο πρωτόκολλο μεταφοράς μαζί με το μηχανισμό για αξιόπιστη μεταφορά δεδομένων. Ο έλεγχος συμφόρησης αφορά στον έλεγχο της κίνησης σε ένα δίκτυο. Σε ένα μηχανισμό για έλεγχο της συμφόρησης γίνεται προσπάθεια για αποφυγή της ενεργής συμμετοχής οποιασδήποτε διεργασίας αλλά και η χρήση όλων των δυνατοτήτων του καναλιού ή ακόμα όλων των ενδιάμεσων κόμβων του δικτύου. Βασικά ακολουθούνται κάποια βήματα περιορισμού των πόρων με σκοπό να μειωθεί ο ρυθμός αποστολής πακέτων.

Στη συνέχεια, θα δούμε πολύ σύντομα τον έλεγχο συμφόρησης στα ενσύρματα και ασύρματα δίκτυα και αναλυτικότερα τον έλεγχο συμφόρησης στα Ασύρματα Δίκτυα Αισθητήρων. Θα δούμε το τι προκαλεί τη συμφόρηση, αλλά και τις επιπτώσεις που έχει η συμφόρηση στα δίκτυα αυτά.

2.7.1 Έλεγχος Συμφόρησης στα Ενσύρματα και Ασύρματα Δίκτυα

Στα ενσύρματα και ασύρματα δίκτυα, η συμφόρηση συμβαίνει όταν η κίνηση στο δίκτυο ξεπερνά τη χωρητικότητα σε οποιοδήποτε σημείο στο δίκτυο. Ο έλεγχος της συμφόρησης στα ενσύρματα δίκτυα, συνήθως γίνεται χρησιμοποιώντας μηχανισμούς από άκρο σε άκρο (end-to-end) όπως στο TCP, οι οποίοι είναι απλοί και εύρωστοι και επιπέδου δικτύου. Παρόλα αυτά, η μέθοδος αυτή δεν επιλύει το πρόβλημα στα ασύρματα δίκτυα γιατί οι ταυτόχρονες μεταδόσεις στα διάφορα «κανάλια» αλληλεπιδρούν και επηρεάζουν το ένα το άλλο, και αυτό επειδή η ποιότητα του καναλιού δείχνει μεγάλη μεταβλητότητα όσο προχωρά ο χρόνος.

Ο μηχανισμός TCP από άκρο σε άκρο δεν μπορεί να χρησιμοποιηθεί στα Ασύρματα Δίκτυα Αισθητήρων γιατί δεν είναι αρκετά αποδοτικός στα δίκτυα αυτά. Οι λόγοι είναι οι εξής: τα μηνύματα ελέγχου που χρησιμοποιούνται για τον εντοπισμό των χαμένων πακέτων στο μηχανισμό από άκρο σε άκρο χρησιμοποιούν ένα επιστρεφόμενο μονοπάτι που αποτελείται από πολλά hops και αυτό δεν είναι αποδοτικό από πλευράς ενέργειας. Επίσης, τα μηνύματα ελέγχου που «ταξιδεύουν» διαμέσου πολλών hops έχουν πολύ μεγάλη πιθανότητα να χαθούν είτε λόγω λαθών στο σύνδεσμο είτε λόγω συμφόρησης. Και τέλος, ο μηχανισμός από άκρο σε άκρο για την ανακάλυψη κάποιου χαμένου πακέτου, αναπόφευκτα χρησιμοποιεί από άκρο σε άκρο επαναμεταδόσεις που σπαταλούν ενέργεια. Σε επόμενο κεφάλαιο θα δούμε διάφορα πρωτόκολλα Ασύρματων Δικτύων Αισθητήρων να χρησιμοποιούν αυτούς τους μηχανισμούς.

2.7.2 Έλεγχος Συμφόρησης Στα Ασύρματα Δίκτυα Αισθητήρων

- **Εντοπισμός συμφόρησης (congestion detection)**, με ένα κοινό τρόπο όπως ελέγχοντας το μήκος της ουράς ή ο ρυθμός εξυπηρέτησης των πακέτων στους ενδιάμεσους κόμβους από τον αποστολέα μέχρι το sink.

Πιο πάνω είδαμε, ότι η συμφόρηση στα ασύρματα δίκτυα διαφέρει από τη συμφόρηση στα ενσύρματα δίκτυα και επίσης, είδαμε ότι οι κόμβοι αισθητήρες στα Ασύρματα Δίκτυα Αισθητήρων αντιμετωπίζουν περιορισμούς όπως περιορισμένες υπολογιστικές δυνατότητες, περιορισμένο χώρο μνήμης, επικοινωνιακό εύρος ζώνης και πάνω από όλα ενέργεια. Συνοψίζοντας, αυτοί είναι και οι κυριότεροι λόγοι για τους οποίους πρέπει να σχεδιαστούν αποδοτικοί μηχανισμοί. Πιο κάτω θα δούμε τέσσερις τρόπους με τους οποίους μπορούμε να αντιμετωπίσουμε τη συμφόρηση από τις προσεγγίσεις που έχουν προταθεί σε κλασσικά δίκτυα όπως το Διαδίκτυο:

- **Γνωστοποίησης συμφόρησης (congestion notification)**, αφού εντοπιστεί η συμφόρηση υπάρχουν δύο τρόποι για να ενημερωθεί το δίκτυο. Αυτονόητα (implicit), δηλαδή π.χ. με ένα δυαδικό bit (CN) που γίνεται piggyback σε άλλα πακέτα ενημερώνεται όλο το δίκτυο για τη συμφόρηση ή κατηγορητικά (explicit) στέλνονται ειδικά μηνύματα για να ενημερώσουν το δίκτυο, όπως στο [29].
- **Αποφυγής συμφόρησης (congestion avoidance)**. Ο έλεγχος συμφόρησης ψάχνει να βρει ένα βέλτιστο (και δίκαιο) τρόπο δέσμευσης ρυθμού αποστολής δεδομένων ανά αισθητήρα (optimal fair rate allocation) ο οποίος να είναι αποδοτικός, έτσι ώστε όταν οι αισθητήρες στέλνουν με αυτό το ρυθμό, το δίκτυο να χρησιμοποιείται πλήρως και ο χρήσιμος ρυθμός (goodput) μετάδοσης δεδομένων ανά αισθητήρα να είναι κοντά στο ρυθμό αποστολής (όχι πολλές επαναμεταδόσεις πακέτων). Με άλλα λόγια τα πρωτόκολλα ελέγχου συμφόρησης για τα WSNs πρέπει να περιορίζουν το πέταγμα πακέτων αφού το πέταγμα πακέτων σημαίνει και απώλεια ενέργειας. Αυτό μπορεί να γίνει με τον ενεργό έλεγχο συμφόρησης (Active Congestion Control), δηλαδή με μια διαδικασία κατά την οποία κάνουμε αποφυγή συμφόρησης με πολύ λίγο κόστος.

- **Μείωσης συμφόρησης (congestion mitigation):** Έστω ένα δίκτυο στο οποίο ο κάθε αισθητήρας στέλνει με σταθερό ρυθμό. Όταν ο συνολικός ρυθμός δεδομένων ξεπεράσει τη χωρητικότητα του δικτύου (network capacity), η τεχνική της μείωσης συμφόρησης θα καθορίσει το πώς θα πρέπει οι αισθητήρες να ρυθμίσουν τη μετάδοση δεδομένων τους, έτσι ώστε ο χρήσιμος ρυθμός δεδομένων του δικτύου (network goodput) να ελαττωθεί χαριτωμένα (graceful degradation). Δηλαδή, αφού γίνει εντοπισμός της συμφόρησης και ενημερωθεί το δίκτυο για τη συμφόρηση, πρέπει κάποιοι μηχανισμοί να ενεργοποιηθούν για να ελέγξουν τη συμφόρηση και έτσι να μπορέσει να λειτουργήσει ξανά σωστά το δίκτυο.

Η συμφόρηση παρατηρείται ακριβώς στο σημείο που όλα τα πακέτα δεδομένων φτάνουν στο sink.

Σύμφωνα με το [7], μπορούμε να κατηγοριοποιήσουμε τη συμφόρηση σε δύο τύπους :

- **Πρώτος τύπος:** σε μια συγκεκριμένη περιοχή πολλοί κόμβοι προσπαθούν να μεταδώσουν ταυτόχρονα προκαλώντας έτσι το χάσιμο των δεδομένων λόγω των παρεμβολών (interference) που θα συμβαίνουν στους κόμβους που ακούν και βρίσκονται στο ίδιο πεδίο με τους κόμβους που μεταδίδουν δεδομένα. Με αυτό τον τρόπο μειώνουν το throughput όλων των κόμβων στην περιοχή. Κάποιου είδους συγχρονισμός μπορεί να μειώσει το χάσιμο πακέτων αλλά όχι πλήρως, επειδή κάποιοι μη γειτονικοί κόμβοι μπορεί να εξακολουθούν να επηρεάζονται από τις παρεμβολές κατά τη μετάδοση.
- **Δεύτερος τύπος:** σε ένα συγκεκριμένο κόμβο η ουρά (queue) ή ο ενταμιευτικός χώρος (buffer space) που χρησιμοποιείται για να φυλάει τα πακέτα που θα μεταδοθούν μπορούν να υπερχειλίσουν (overflow). Αυτός ο τύπος συμφόρησης παρατηρείται ευρέως στα ενσύρματα δίκτυα και το χάσιμο πακέτων αφορά στην επιτυχημένη παραλαβή του πακέτου, αλλά το πακέτο πρέπει να πεταχτεί λόγω της υπερχειλίσης.

Να πούμε ότι και οι δύο τύποι συμφόρησης μπορεί να συμβαίνουν ταυτόχρονα. Στη συνέχεια, θα δούμε κάποια συμπτώματα, δηλαδή το τι προκαλεί τη συμφόρηση στα Ασύρματα Δίκτυα Αισθητήρων.

Ο ρυθμός με τον οποίο ωθούνται τα δεδομένα. Δηλαδή, για κάποιες εφαρμογές κάποιο γεγονός μπορεί να είναι περιοδικό ενώ για κάποιες άλλες μπορεί να είναι τυχαίο. Για κάποιες μπορεί να συχνό ή σε κάποιες άλλες θα υπάρχει ανάγκη για συνεχή αποστολή δεδομένων (όπως αυτή που θα εξετάσουμε στην υλοποίησή μας). Επίσης, σε κάποιες εφαρμογές μπορεί το γεγονός να εμφανίζεται σε μια μεμονωμένη περιοχή του δικτύου ή σε κάποια άλλη εφαρμογή να αφορά ολόκληρη την περιοχή και πάντα με τον ίδιο στόχο. Τα δεδομένα να φτάσουν στο sink. Παρόλα αυτά, και στις δύο περιπτώσεις μπορεί να παρατηρηθεί συμφόρηση γιατί το κανάλι μετάδοσης αλλάζει στο χρόνο, συχνά δραματικά, και ταυτόχρονες μεταφορές δεδομένων σε διαφορετικά «κανάλια» μεταδόσεων αλληλεπιδρούν, κάνοντας έτσι την ποιότητα να εξαρτάται όχι μόνο από το θόρυβο αλλά και από την πυκνότητα των κόμβων αισθητήρων στο δίκτυο.

Η πρόσθεση κόμβων αισθητήρων για σκοπούς αντικατάστασης κάποιων κόμβων που απέτυχαν (πιθανώς λόγω απώλειας ενέργειας) ή για αύξηση της πυκνότητάς τους για να μπορούν να έχουν καλύτερη απόδοση (να αισθάνονται ένα γεγονός γρηγορότερα).

Στα παραδοσιακά ενσύρματα και ασύρματα δίκτυα, η υπερχειλίση των ενταμιευτικών χώρων και το πέταγμα των πακέτων, και η αύξηση της καθυστέρησης είναι τα κύρια συμπτώματα που προκαλούν τη συμφόρηση. Οι διάφοροι ερευνητές έχουν καταφέρει με συνδυασμό της αναπροσαρμογής του ρυθμού στα άκρα του δικτύου και το πέταγμα πακέτων που γίνεται στο επίπεδο δικτύου, να εγγυώνται ότι αυτά τα δίκτυα θα μπορούν να λειτουργούν χωρίς να καταρρέουν λόγω συμφόρησης. Στα Ασύρματα Δίκτυα Αισθητήρων, εκτός από την υπερχειλίση των ενταμιευτικών χώρων, ένα σημαντικό σύμπτωμα της συμφόρησης είναι η μείωση της ποιότητας του καναλιού μετάδοσης που προκαλείται από την αύξηση της ποσότητας της κίνησης που στέλνεται από άλλα κομμάτια του δικτύου. Επειδή τα «κανάλια» μετάδοσης δεν προστατεύονται το ένα από το άλλο όπως γίνεται στα ενσύρματα δίκτυα, η κίνηση ταξιδεύει σε ένα δεδομένο κομμάτι του δικτύου έχοντας επιβλαβή επίδραση

στην ποιότητα του καναλιού σε άλλα κομμάτια του δικτύου. Η φτωχή και εναλλασσόμενη στο χρόνο ποιότητα του καναλιού, η ασυμμετρία στο επικοινωνιακό κανάλι και οι κρυμμένοι κόμβοι (hidden terminal) δουλεύουν όλα μαζί εναντίον μιας καλά οργανωμένης κίνησης έτσι ώστε να κάνουν τη μεταφορά και παράδοση των δεδομένων δύσκολη.

Πιο πάνω αναφέραμε τα πιθανά συμπτώματα που μπορεί να προκαλέσουν συμφόρηση σε ένα δίκτυο ασύρματων αισθητήρων. Τώρα πρέπει να δούμε και τις επιπτώσεις που έχει η συμφόρηση στα δίκτυα αυτά. Στην κορυφή της πυραμίδας των επιπτώσεων που προκαλεί η συμφόρηση στα Ασύρματα Δίκτυα Αισθητήρων βρίσκεται η σπατάλη ενέργειας. Η ενέργεια είναι ένα από τα σημαντικότερα προβλήματα που έχουν να αντιμετωπίσουν τα δίκτυα αυτά, αφού η ενέργεια στους κόμβους αισθητήρων είναι μειωμένη και η συμφόρηση κάνει ακόμα πιο έντονο το πρόβλημα. Η σπατάλη ενέργειας μπορεί να κάνει κάποιο κόμβο να αποτύχει λόγω έλλειψης ενέργειας και να οδηγήσει ακόμα και σε διαχωρισμό του δικτύου αν πολλοί κόμβοι αποτύχουν λόγω έλλειψης ενέργειας. Δηλαδή, η συμφόρηση προκαλεί και μείωση της διάρκειας ζωής των κόμβων αισθητήρων ή ακόμα, και ολόκληρου του δικτύου.

Ένα πιθανώς πιο σημαντικό σύμπτωμα μαζί με την έλλειψη ενέργειας και το διαχωρισμό του δικτύου είναι η μείωση της ακρίβειας των πληροφοριών ή ακόμα και χάσιμο σημαντικών πληροφοριών. Για παράδειγμα, σε περίπτωση που η πληροφορία που μεταφέρεται σε ένα δίκτυο ασύρματων αισθητήρων ήταν για να ειδοποιηθεί για πυρκαγιά, και αυτό το μήνυμα χάθηκε λόγω της συμφόρησης, αυτό μπορεί να προκαλέσει ολέθριες καταστροφές αφού δεν θα ενημερωθούν ποτέ οι πυροσβέστες για την πυρκαγιά.

Η συμφόρηση προκαλεί μείωση του throughput και αύξηση της καθυστέρησης. Σε συνέχεια με το προηγούμενο παράδειγμα, για την πυρκαγιά καθυστερημένες πληροφορίες για φωτιά μπορεί να οδηγήσουν και πάλι σε ολέθριες καταστροφές. Ακόμα σε εφαρμογές όπου υπάρχει ανάγκη για συνεχή ροή δεδομένων αυτό θα προκαλέσει δυσαρέσκεια στους χρήστες της εφαρμογής αφού θα υπάρχουν ανεπιθύμητες διακοπές. Είναι σημαντικό να αναφέρουμε ότι όλες οι επιπτώσεις από τη συμφόρηση, δηλαδή η σπατάλη ενέργειας, η μείωση ζωής του δικτύου ή των κόμβων, η μείωση της ακρίβειας των πληροφοριών, το χάσιμο πακέτων, η

μείωση του throughput και η αύξηση της καθυστέρησης είναι όλα απαιτήσεις για την ποιότητα υπηρεσιών στα Ασύρματα Δίκτυα Αισθητήρων, όπως είδαμε και προηγουμένως.

2.7.3 Τρόποι Αντιμετώπισης της Συμφόρησης στα Ασύρματα Δίκτυα Αισθητήρων

Στα Ασύρματα Δίκτυα Αισθητήρων έχουν σχεδιαστεί και αναπτυχθεί ειδικά πρωτόκολλα που να υλοποιούν έλεγχο της συμφόρησης. Ειδικότερα, έχουν σχεδιαστεί ειδικοί μηχανισμοί για εντοπισμό της συμφόρησης, για γνωστοποίηση της συμφόρησης στο υπόλοιπο δίκτυο και για αποφυγή της συμφόρησης, όπως είδαμε και προηγουμένως. Σημαντικό είναι το γεγονός ότι τα περισσότερα πρωτόκολλα που αναπτύχθηκαν αφορούν τον έλεγχο της συμφόρησης στην περιοχή του sink, αφού έχει αποδειχτεί από πολλούς ερευνητές ότι η συμφόρηση συμβαίνει πιο συχνά στη συγκεκριμένη περιοχή, όπου φτάνουν όλα τα πακέτα δεδομένων των κόμβων αισθητήρων. Επίσης, οι μηχανισμοί που έχουν αναπτυχθεί μπορούν να κατηγοριοποιηθούν με βάση τις εφαρμογές στις οποίες μπορούν να χρησιμοποιηθούν. Δηλαδή, κάποιοι μηχανισμοί έχουν αναπτυχθεί για να χρησιμοποιούνται σε εφαρμογές, οι οποίες οδηγούνται από τα γεγονότα. Σε αυτές τις εφαρμογές δεδομένα στέλνονται μόνο όταν οι κόμβοι αισθητήρες αισθανθούν κάποιο γεγονός, όπως σε περίπτωση πυρκαγιάς. Βέβαια, όπως αναφέραμε και σε προηγούμενη παράγραφο, υπάρχουν και οι εφαρμογές όπου γίνεται συνεχή ροή δεδομένων. Η υλοποίηση μας αφορά το δεύτερο είδος εφαρμογών. Πιο κάτω θα δούμε κάποια από τα πρωτόκολλα που αναπτύχθηκαν και για τα δύο είδη εφαρμογών.

Αρχικά, να αναφέρουμε το Event-to-Sink Reliable Transport (ESRT) [7] το οποίο έχει αναπτυχθεί για εφαρμογές όπου οι κόμβοι στέλνουν δεδομένα μόνο όταν αισθανθούν κάποιο γεγονός. Το ESRT ρυθμίζει το ρυθμό αποστολής των αισθητήρων όταν βρεθεί συμφόρηση στο δίκτυο. Όταν παραληφθούν πακέτα στα οποία υπάρχει το bit για γνωστοποίηση της συμφόρησης, το sink ρυθμίζει τον αναφερόμενο ρυθμό στέλνοντας σε όλους ένα πακέτο χρησιμοποιώντας μεγάλη ενέργεια στο σήμα ελέγχου έτσι ώστε να φτάσει σε όλους τους κόμβους πηγές. Αυτό το σήμα μεγάλης ενέργειας μπορεί να διακόψει την

αποστολή άλλων πακέτων. Άλλο ένα πρωτόκολλο που χρησιμοποιείται στις ίδιες εφαρμογές είναι το Congestion Detection and Avoidance (CODA) [29]. Το CODA χρησιμοποιεί ένα συνδυασμό του παρόντος και του παρελθόντος φορτίου του καναλιού και τη τρέχουσα χωρητικότητα του προσωρινού χώρου αποθήκευσης για να αναφέρει τον ακριβή εντοπισμό της συμφόρησης σε κάθε παραλήπτη με χαμηλό κόστος. Μόλις κάποιος κόμβος εντοπίσει συμφόρηση στέλνει ένα backpressure μήνυμα στους upstream κόμβους για να ελέγξουν το ρυθμό αποστολής hop-by-hop. Επίσης, μπορεί να διεκδικεί έλεγχο της συμφόρησης μεταξύ πολλαπλών πηγών από ένα απλό sink σε μια συνεχή συμφόρηση. Να αναφέρουμε ότι το CODA ξεπερνά κάποιους περιορισμούς του ESRT, αλλά δε λαμβάνει υπόψη του την αξιοπιστία των πακέτων και τη δικαιοσύνη των γεγονότων.

Ένα πρωτόκολλο που μπορεί να χρησιμοποιηθεί για πολλά είδη εφαρμογών είναι το Pump Slowly Fetch Quickly (PSFQ) [28]. Το PSFQ είναι ένα κλιμακωτό και αξιόπιστο πρωτόκολλο που εγγυάται αυστηρή μεταφορά των δεδομένων παρά αξιοπιστία (όπως γίνεται στο ESRT) μόνο στα πακέτα δεδομένων. Το PSFQ βασίζεται στην αργή παροχή πακέτων στο δίκτυο και εκτελεί hop-by-hop ανάκαμψη σε περίπτωση που κάποιο πακέτο χαθεί. Η προσέγγιση του PSFQ συνδυάζει μια εξειδικευμένη παράμετρο και ένα ακριβή χρονικό συντονισμό. Με αυτά επιτυγχάνει τη χρήση του από πολλές εφαρμογές. Το PSFQ έχει αρκετά μειονεκτήματα. Δεν μπορεί να εντοπίσει το χάσιμο ενός μικρού αριθμού πακέτων αφού χρησιμοποιεί NACK μηνύματα. Επίσης, χρησιμοποιεί στατική και αργή παροχή πακέτων και αυτό έχει σαν αποτέλεσμα να υπάρχει μεγάλη καθυστέρηση στο δίκτυο και τέλος, απαιτεί περισσότερο χώρο προσωρινής μνήμης λόγω του μηχανισμού hop-by-hop που χρησιμοποιεί. Τα πρωτόκολλα που είδαμε μέχρι τώρα μπορούν να χρησιμοποιηθούν είτε μόνο σε εφαρμογές που οδηγούνται από τα γεγονότα, είτε και στα δύο είδη εφαρμογών. Η δική μας υλοποίηση αφορά μόνο εφαρμογές με συνεχή ροή δεδομένων. Στη συνέχεια θα δούμε ένα πρωτόκολλο που χρησιμοποιείται στον ίδιο τύπο εφαρμογών όπως και στη δική μας υλοποίηση.

Στο πρωτόκολλο Interference-Minimized Multipath Routing (I2MR) [25] αρχικά γίνεται καθορισμός του μονοπατιού από τον κόμβο πηγής μέχρι τον προορισμό. Για την ακρίβεια καθορίζονται ένα πρωτεύον και ένα δευτερεύον μονοπάτι καθώς και ένα επιπλέον

εφεδρικό, το οποίο χρησιμοποιείται μόνο σε περίπτωση που τα άλλα δύο αποτύχουν. Οι κόμβοι πηγής ξεκινούν να στέλνουν δεδομένα με βάση τον αρχικό ρυθμό στον οποίο αρχικοποιούνται. Τη συμφόρηση εντοπίζουν οι ενδιάμεσοι κόμβοι και με ειδικό μήνυμα ενημερώνουν τους κόμβους πηγής να μειώσουν το ρυθμό τους. Να αναφέρουμε ότι το πρωτόκολλο αυτό δουλεύει με βάση κάποιους περιορισμούς. Ενδεικτικά να αναφέρουμε τους εξής: το βεληνεκές της παρεμβολής (interference) πρέπει να είναι δύο φορές μεγαλύτερο από το επικοινωνιακό βεληνεκές, οι τοποθεσίες των κόμβων πηγής και των gateway κόμβων πρέπει να είναι γνωστές και οι κόμβοι αυτοί, πρέπει να έχουν περισσότερη ενέργεια από τους υπόλοιπους. Στη δική μας υλοποίηση θα προσπαθήσουμε να υλοποιήσουμε ένα αλγόριθμο με τους λιγότερο δυνατούς περιορισμούς, τους οποίους θα δούμε στο Κεφάλαιο 3.

Στο σημείο αυτό ολοκληρώνοντας την αναφορά των πρωτοκόλλων τόσο του ‘επιπέδου’ δρομολόγησης όσο και του ‘επιπέδου’ μεταφοράς, να τονίσουμε τη μεγάλη αναγκαιότητα των αποδοτικών μηχανισμών για έλεγχο και αποφυγή της συμφόρησης στα Ασύρματα Δίκτυα Αισθητήρων. Το χάσιμο πακέτων λόγω συμφόρησης διακινδυνεύει το σκοπό της εφαρμογής και ακόμα, προκαλεί κατανάλωση της ενέργειας η οποία είναι πολύτιμη στα δίκτυα αυτά. Όταν κάποιο πακέτο χαθεί, χάνεται μαζί του όλη η ενέργεια που χρειάστηκε για να δημιουργηθεί, να σταλεί και να παραληφθεί. Όλες οι μεθοδολογίες, όπως τις έχουμε δει σε αυτό το Κεφάλαιο, προσπάθησαν να δουν το πρόβλημα από διαφορετικές πλευρές. Δηλαδή, οι μεθοδολογίες διαφοροποιούνται ως προς τις εφαρμογές στις οποίες χρησιμοποιούνται και ως προς τους περιορισμούς που θέτονται σε κάθε μια.

Σε αυτή τη Διπλωματική εργασία θα υλοποιήσουμε ένα αλγόριθμο στον οποίο γίνεται συνδυασμός τόσο του ‘επιπέδου’ δρομολόγησης, όσο και του ‘επιπέδου’ μεταφοράς για εφαρμογές όπου υπάρχει συνεχής ροή δεδομένων. Δηλαδή, θα δημιουργούμε ένα ισορροπημένο δέντρο και θα κάνουμε έλεγχο της συμφόρησης για να αποφεύγουμε το χάσιμο πακέτων, και αυτό θα δείξουμε μέσα από τα αποτελέσματα μας στο Κεφάλαιο 4. Να προσθέσουμε ότι ο μηχανισμός που χρησιμοποιούμε για αποφυγή της συμφόρησης είναι εμπνευσμένος από τη βιολογία. Χρησιμοποιούμε το μαθηματικό μοντέλο Lotka–Volterra στην προσπάθειά μας να εκμεταλλευτούμε τα πλεονεκτήματα που μας προσφέρει η βιολογία.

Τη σταθερότητα, ευρωστία και προσαρμοστικότητα που μπορεί να έχει η κάθε οντότητα όταν δρα ανεξάρτητα, αυτόνομα και κατανεμημένα, αλλά και σε συνεργασία με τις άλλες οντότητες για επίτευξη του κοινού στόχου. Χαρακτηριστικά τα οποία λάβαμε υπόψη μας κατά το σχεδιασμό της δική μας υλοποίησης. Να αναφέρουμε ότι κάποια από αυτά τα χαρακτηριστικά επαληθεύονται στο Κεφάλαιο 4, όπου αξιολογούμε την υλοποίηση μας μέσα από διάφορες μετρικές.

2.7.4 Έλεγχος Συμφόρησης στο Διαδίκτυο με Χρήση του Μοντέλου Lotka-Volterra

Στην παράγραφο αυτή θα δούμε ένα αλγόριθμο, ο οποίος έχει σκοπό να κάνει έλεγχο της συμφόρησης στο Διαδίκτυο χρησιμοποιώντας το μαθηματικό μοντέλο Lotka-Volterra (το ίδιο μαθηματικό μοντέλο που χρησιμοποιούμε και εμείς στην υλοποίησή μας). Κάποιες άλλες προτάσεις που τέθηκαν για αντιμετώπιση της συμφόρησης αφορούν είτε καλυτέρευση του τρέχον μηχανισμού ελέγχου συμφόρησης του TCP, είτε παρέχεται ένας νέος τρόπος αντιμετώπισης της συμφόρησης.

Στο [14] υλοποιείται ένα καινούριος μηχανισμός ελέγχου συμφόρησης του TCP, το οποίο χρησιμοποιεί πληροφορίες που παίρνει από το φυσικό και διαθέσιμο bandwidth με τη βοήθεια μιας τεχνικής μετρήσεων. Βασικά, στο [14] έχουν σχεδιάσει ένα αλγόριθμο ο οποίος ελέγχει το μέγεθος του παραθύρου (window size). Στόχος είναι να γίνεται γρήγορη η αλλαγή του μεγέθους του παραθύρου. Η νέα τιμή πρέπει να είναι μια επαρκής τιμή που βασίζεται στο bandwidth, έτσι ώστε να γίνεται μια δίκαιη κατανομή του bandwidth μεταξύ των συναγωνιζόμενων κόμβων. Αυτός ο αλγόριθμος βασίζεται πάνω στο μαθηματικό μοντέλο του Lotka-Volterra. Οι συγγραφείς του [14] χρησιμοποιούν αυτό το μαθηματικό μοντέλο βασιζόμενοι στην ουσιαστική του φύση. Τη σταθερότητα και ευρωστία που μπορεί να προσφέρει όταν η κάθε οντότητα δρα ανεξάρτητα, αυτόνομα και κατανεμημένα.

Στη συνέχεια, θα δούμε την αναλογία μεταξύ του μοντέλου του Lotka-Volterra και του μηχανισμού (οι παράγοντες του μοντέλου Lotka-Volterra θα αναλυθούν στην παράγραφο που ακολουθεί). Ο πληθυσμός των ειδών είναι το μέγεθος του παραθύρου στη σύνδεση TCP,

η ικανότητα μεταφοράς του περιβάλλοντος (carrying capacity) είναι το φυσικό bandwidth και ο interspecific συναγωνισμός μεταξύ των ειδών είναι το bandwidth που μοιράζονται οι συναγωνιζόμενοι κόμβοι μεταξύ των συνδέσεων.

Στην επόμενη παράγραφο θα δούμε αναλυτικά το μαθηματικό μοντέλο του Lotka-Volterra, αφού όπως αναφέραμε είναι το μαθηματικό μοντέλο που χρησιμοποιείται και στο μηχανισμό για έλεγχο της συμφοράς που χρησιμοποιούμε στην υλοποίησή μας.

2.7.4.1 Μοντέλο Lotka – Volterra

Το μοντέλο Lotka-Volterra [4] είναι το μοντέλο που χρησιμοποιείται στο μηχανισμό ελέγχου συμφοράς που θα δούμε πιο κάτω. Να αναφέρουμε ότι συχνά χρησιμοποιούνται απλά μαθηματικά μοντέλα εμπνευσμένα από τη βιολογία [5] για διάφορα πρωτόκολλα στα Ασύρματα Δίκτυα Αισθητήρων, αφού ικανοποιούν πολλούς από τους περιορισμούς τους (ευρωστία, αυτονομία κ.α.) και αυτό στηρίζεται στο γεγονός ότι τόσες χιλιάδες χρόνια ο πλανήτης συνεχίζει να υπάρχει παρά τις κακοπροαίρετες επεμβάσεις του ανθρώπου.

Βασικά, με δεδομένο το γεγονός ότι κανένα είδος δε ζει απομονωμένο εξετάζεται η αλληλεπίδραση μεταξύ των ειδών που μελετούνται, αλλά και μεταξύ των υπόλοιπων ειδών που μοιράζονται το ίδιο περιβάλλον. Για αρκετό χρονικό διάστημα στην Οικολογία ο ανταγωνισμός ήταν η σημαντικότερη κινητήρια δύναμη στις κοινότητες – φυσική επιλογή, ανταγωνιστικός αποκλεισμός κ.τ.λ.. Παρόλα αυτά, τώρα αποδεικνύεται πιο περίπλοκη.

Υπάρχουν διάφορα είδη αλληλεπιδράσεων [31], οι οποίες χωρίζονται σε διάφορες κλάσεις βασισμένες είτε στα αποτελέσματα, είτε στο μηχανισμό της αλληλεπίδρασης. Στην κατηγορία των αλληλεπιδράσεων που βασίζονται στο αποτέλεσμα υπάρχουν έξι δυνατοί συνδυασμοί, οι οποίοι κυμαίνονται από ευεργετικές μέχρι επιβλαβείς αλληλεπιδράσεις. Το επίπεδο ωφέλειας ή ζημιάς είναι συνεχόμενο και όχι διακριτό, όπως για παράδειγμα μια αλληλεπίδραση μπορεί να είναι επιτόλεια επιβλαβής μέχρι και υπερβολικά. Είναι σημαντικό να σημειωθεί ότι οι αλληλεπιδράσεις δεν είναι στατικές, αφού τα είδη σε πολλές περιπτώσεις αλληλεπιδρούν διαφορετικά κάτω από διαφορετικές συνθήκες. Αυτό ισχύει σε πολλές

περιπτώσεις, αλλά δεν περιορίζεται σε αυτό αφού υπάρχουν και περιπτώσεις όπου τα είδη έχουν πολλαπλά, δραστικά και διαφορετικά στάδια ζωής. Οι έξι διαφορετικοί συνδυασμοί είναι οι εξής:

- Ουδετερότητα (neutralism): περιγράφει τη σχέση μεταξύ δύο ειδών τα οποία αλληλεπιδρούν χωρίς να επηρεάζουν το ένα το άλλο.
- Αμενσαλισμός (amensalism): συμπεριλαμβάνει μεταξύ δύο ειδών την επικείμενη ή περιοριστική επιτυχία τους ενός στο άλλο είδος, χωρίς να επηρεάζεται θετικά ή αρνητικά από τη παρουσία του άλλου.
- Συναγωνισμός (competition): είναι η αλληλεπίδραση μεταξύ ατόμων ή πληθυσμών, οι οποίοι είναι αμοιβαία επιβλαβείς.
- Ανταγωνισμός (antagonism): είναι η αλληλεπίδραση όπου το ένα είδος ευεργετεί εις βάρος του άλλου.
- Παρασιτισμός (commensalism): είναι η αλληλεπίδραση όπου ο ένας οργανισμός ευεργετεί και ο άλλος ούτε ευεργετείται, αλλά ούτε και πληγώνεται.
- Αμοιβαιότητα (mutualism): είναι η αλληλεπίδραση μεταξύ δύο ή περισσότερων ειδών, όπου τα είδη παράγουν ένα αμοιβαίο όφελος. Για παράδειγμα, μια αυξανόμενη ικανότητα μεταφοράς.

Στην κατηγορία των αλληλεπιδράσεων που βασίζονται στο μηχανισμό της αλληλεπίδρασης εμπίπτουν η συμβίωση και ο συναγωνισμός. Η συμβίωση (symbiosis) είναι η αλληλεπίδραση όπου όλα τα είδη ευεργετούν ή μπορεί να χρησιμοποιηθεί γενικότερα για να περιγράψει όλες τις στενές σχέσεις μεταξύ των οργανισμών, ακόμα και τον παρασιτισμό που είδαμε πιο πάνω. Ο συναγωνισμός (competition) είναι η αλληλεπίδραση μεταξύ των ειδών όπου η ικανότητα τους ενός είδους μειώνεται από την παρουσία του άλλου είδους [3]. Όπως θα δούμε πιο κάτω, ο μηχανισμός για αποφυγή της συμφόρησης που χρησιμοποιούμε στην

υλοποίησή μας βασίζεται στο συναγωνισμό σε σύγκριση με τα υπόλοιπα είδη αλληλεπιδράσεων.

Ο συναγωνισμός [11] είναι προφανώς η αλληλεπίδραση μεταξύ δύο ή περισσότερων ειδών για κάποιο περιορισμένο πόρο στο περιβάλλον. Ο πόρος αυτός μπορεί να είναι τρόφιμα ή διάστημα ή ταίρι ή οτιδήποτε για το οποίο η ζήτηση είναι περισσότερη από τη διάθεση. Αυτό ονομάζεται ανταγωνισμός πόρων. Όταν κάποιο είδος είναι καλύτερος ανταγωνιστής σε ένα ανταγωνισμό μεταξύ των ειδών, τα υπόλοιπα είδη επηρεάζονται αρνητικά μειώνοντας το μέγεθος του πληθυσμού τους ή μειώνοντας τους ρυθμούς ανάπτυξης τους. Αυτό έχει ως αντίκτυπο να επηρεάζει τη δύναμη των πληθυσμών του ανταγωνιστή. Ο συναγωνισμός μεταξύ των ειδών του ίδιου είδους ονομάζεται *intraspecific*, ενώ ο συναγωνισμός μεταξύ οντοτήτων διαφορετικού είδους ονομάζεται *interspecific* [32]. Επίσης, ο συναγωνισμός μπορεί να είναι άμεσος ή έμμεσος. Οι όροι που θα δούμε πιο κάτω περιγράφουν τους μηχανισμούς με τους οποίους συμβαίνει ο συναγωνισμός. Αυτοί οι μηχανισμοί εφαρμόζονται το ίδιο τόσο στο *intraspecific* όσο και στο *interspecific* συναγωνισμό και είναι οι εξής:

- Συναγωνισμός Παρεμβολής (*interference competition*): συμβαίνει άμεσα μεταξύ των οντοτήτων μέσω επιθετικότητας κ.α. όταν οι οντότητες παρεμποδίζονται στον να μεταφέρουν τροφή, να επιβιώνουν ή να αναπαράγονται από άλλες ή ακόμα, όταν αποτρέπονται άμεσα στον καθιερωθούν σε μια μερίδα κάποιου βιότοπου.
- Συναγωνισμός Εκμετάλλευσης (*exploitation competition*): συμβαίνει έμμεσα μέσω ενός κοινού περιορισμένου πόρου και δρα σαν ενδιάμεσος. Για παράδειγμα, η χρήση του πόρου μειώνει τη διαθέσιμη ποσότητα στους άλλους ή συναγωνίζονται το διαθέσιμο χώρο.
- Προφανής Συναγωνισμός (*apparent competition*): συμβαίνει έμμεσα μεταξύ δύο ειδών, τα οποία είναι και τα δύο επάνω από τον ίδιο καταστροφέα. Για παράδειγμα, τα είδη A και B είναι και τα δύο θηράματα για το αρπακτικό Γ. Η αύξηση τους είδους A θα προκαλέσει τη μείωση του είδους B γιατί η

αύξηση του είδους A θα μειώσει τον αριθμό του αρπακτικού Γ, τα οποία με τη σειρά τους θα κυνηγήσουν περισσότερες οντότητες του είδους B.

Οι συναγωνιστικές αλληλοεπιδράσεις μεταξύ των οργανισμών μπορούν να έχουν μεγάλη επίδραση στην εξέλιξη των ειδών, τη δομή των κοινοτήτων, των οποίων τα είδη είτε συνυπάρχουν είτε όχι και της κατανομής των ειδών. Μοντελοποιώντας αυτές τις αλληλεπιδράσεις προβάλλεται μια χρήσιμη δομή για να προβλέπεις τα αποτελέσματα. Το μοντέλο του συναγωνισμού μεταξύ των ειδών Lotka-Volterra, είναι ένα απλό μαθηματικό μοντέλο που βασίζεται στη λογιστική ανάπτυξη και το οποίο μπορεί να χρησιμοποιηθεί για να κατανοήσουμε πώς οι διαφορετικοί παράγοντες επηρεάζουν το αποτέλεσμα των συναγωνιστικών αλληλοεπιδράσεων.

Το μοντέλο βασίζεται σε δύο εξισώσεις ανάπτυξης πληθυσμού. Η κάθε μία εξίσωση είναι για το κάθε συναγωνιστικό είδος. Τα είδη αναφέρονται ως είδος 1 και ως είδος 2. Οι αριθμοί 1 και 2 χρησιμοποιούνται για να δηλώνονται τα είδη και σε ποιο αναφέρονται. Πιο κάτω μπορούμε να δούμε τις εξισώσεις και στον Πίνακα 2-1 βλέπουμε τους όρους των εξισώσεων.

Εξίσωση για την ανάπτυξη πληθυσμού του είδους 1:

$$\frac{dN_1}{dt} = r_1 N_1 \left(\frac{K_1 - N_1 - a_{12} N_2}{K_1} \right)$$

Εξίσωση για την ανάπτυξη πληθυσμού του είδους 2:

$$\frac{dN_2}{dt} = r_2 N_2 \left(\frac{K_2 - N_2 - a_{21} N_1}{K_2} \right)$$

| | |
|---|----------------------------|
| N | population size |
| t | Time |
| K | carrying capacity |
| r | intrinsic rate of increase |
| a | competition coefficient |

Πίνακας 2-1. Παράγοντες εξισώσεων

Τα κυριότερα σύμβολα των εξισώσεων είναι τα a :

- Το a_{12} αναφέρεται στην επίδραση της ανάπτυξη πληθυσμού του είδους 1 στο ξεχωριστό είδος 2, σχετικά με την επίδραση στο ξεχωριστό είδος 1.
- Το a_{21} αναφέρεται στην επίδραση της ανάπτυξη πληθυσμού του είδους 2 στο ξεχωριστό είδος 1, σχετικά με την επίδραση στο ξεχωριστό είδος 2.

Στο Κεφάλαιο 3 θα δούμε πως το μαθηματικό μοντέλο Lotka-Volterra χρησιμοποιείται για αποφυγή της συμφόρησης στα Ασύρματα Δίκτυα Αισθητήρων.

Κεφάλαιο 3

Βασική Λειτουργία και Οντότητες του Μηχανισμού Ελέγχου Συμφόρησης

Σε αυτό το Κεφάλαιο παρουσιάζεται η βασική λειτουργία του μηχανισμού ελέγχου συμφόρησης (που ενσωματώνει λειτουργίες δρομολόγησης) ο οποίος αναπτύχθηκε στα πλαίσια της διατριβής αυτής. Δίνεται έμφαση στις διάφορες οντότητες του δικτύου και στους ρόλους της κάθε οντότητας. Το πλαίσιο εργασίας μας αποτελείται στην ουσία από δύο μέρη. Το ένα μέρος αφορά στη δρομολόγηση και το άλλο μέρος αφορά στον έλεγχο συμφόρησης. Στο Κεφάλαιο αυτό θα περιγράψουμε αναλυτικά τα δύο μέρη.

Οι οντότητες που θα περιγραφούν στη συνέχεια, υλοποιήθηκαν στον ανοικτό κώδικα εξομοιωτή δικτύων (NS2 network simulator [Appendix A]) με σκοπό να μελετηθεί η συμπεριφορά του σε συνθήκες πραγματικού δικτύου, ώστε να μετρηθεί η απόδοσή του, να ελεγχθεί η αξιοπιστία του στη βάση συγκεκριμένων μετρικών απόδοσης. Τα αποτελέσματα παρουσιάζονται στο επόμενο Κεφάλαιο.

Η υλοποίηση μας αποτελείται όπως έχουμε αναφέρει από δύο μέρη. Αρχικά γίνεται κτίσιμο του δέντρου έτσι ώστε όλοι οι κόμβοι αισθητήρων που βρίσκονται στο δίκτυο να μπορούν να έχουν επικοινωνία με το sink. Επίσης, μέσα από τον αλγόριθμο μας γίνονται οι απαραίτητες διαδικασίες έτσι ώστε να γίνεται και συντήρηση του δέντρου δηλαδή, σε

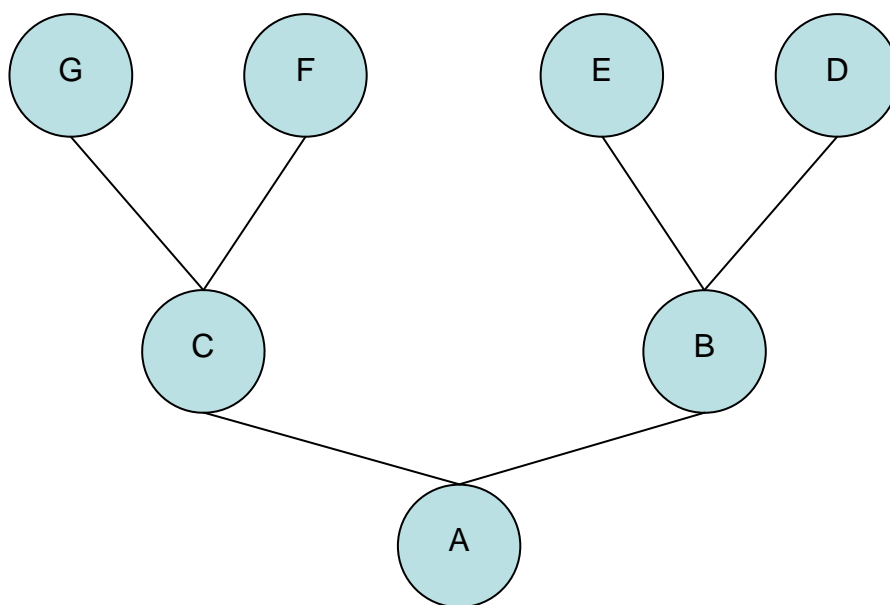
περίπτωση αποτυχίας να μπορεί το δέντρο να ξανακτίζεται. Το δεύτερο μέρος της υλοποίησης μας αφορά τον έλεγχο συμφόρησης. Να επισημάνουμε ότι ο έλεγχος της συμφόρησης γίνεται στη βάση του μαθηματικού μοντέλου Lotka – Volterra που περιγράψαμε στο προηγούμενο Κεφάλαιο για να υπολογίζουμε το ρυθμό με τον οποίο θα αποστέλλουν οι κόμβοι αισθητήρων τα δεδομένα τους με στόχο να αποφεύγουν τη συμφόρηση. Ο μηχανισμός αυτός δίνει έμφαση στις εφαρμογές με συνεχή ροή δεδομένων (streaming). Προσπαθήσαμε να κάνουμε μια υλοποίηση λαμβάνοντας υπόψη όλους τους περιορισμούς των Ασύρματων Δικτύων Αισθητήρων και επίσης, προσπαθήσαμε να δημιουργήσουμε ένα αλγόριθμο με τις λιγότερο δυνατές υποθέσεις τόσο όσο αφορά στους κόμβους αισθητήρων ως μονάδες αλλά και στις ανάγκες μεταξύ τους ως δίκτυο. Δηλαδή, προσπαθήσαμε να δημιουργήσουμε ένα αλγόριθμο όπου οι κόμβοι να λειτουργούν αυτόνομα, προσδίδοντας έτσι ευρωστία και σταθερότητα στο δίκτυο. Ένα δίκτυο το οποίο μπορεί να κτίζεται και να συντηρείται μόνο του χωρίς την ανάγκη κεντρικής οντότητας ανταλλάσσοντας ένα μικρό αριθμό μηνυμάτων, έτσι ώστε να μη γίνεται σπατάλη ενέργειας. Όλα αυτά θα προσπαθήσουμε να τα δείξουμε τόσο κατά την υλοποίησή μας, όσο και στις μετρήσεις που θα παρουσιάσουμε στο επόμενο Κεφάλαιο.

Στη συνέχεια θα δούμε ξεχωριστά τη διαδικασία που αφορά τη δρομολόγηση και το μηχανισμό για έλεγχο συμφόρησης, αλλά και το πώς τα δύο αυτά μέρη συνενώνονται στην υλοποίησή μας.

3.1 Περιγραφή Αλγορίθμου Δρομολόγησης

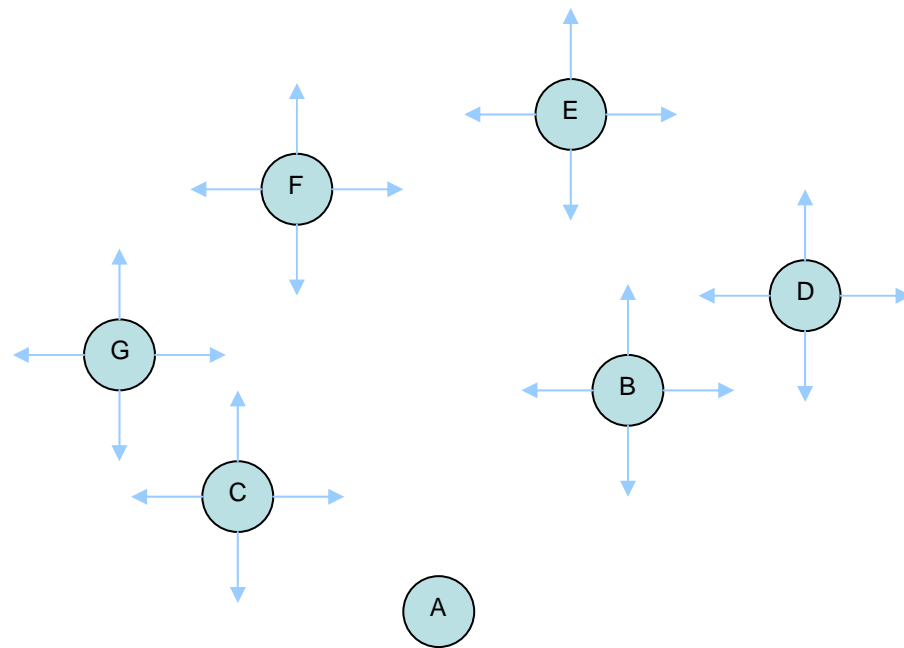
Αρχικά θα δούμε τον τρόπο με τον οποίο κτίζεται το δέντρο στην υλοποίησή μας, αλλά και τον τρόπο με τον οποίο συντηρείται το δέντρο μέσα από ένα παράδειγμα. Στο Σχήμα 3-1 στη θέση του κόμβου A βρίσκεται πάντα το sink. Ο κόμβος B έχει για παιδιά τους κόμβους D και E, και ο κόμβος C έχει για παιδιά τους κόμβους F και G. Στόχος όλων των κόμβων

αισθητήρων είναι να συνδεθούν στο δέντρο που κτίζεται με κορυφή το sink. Ο τρόπος με τον οποίο κτίζεται το δέντρο είναι ο εξής:



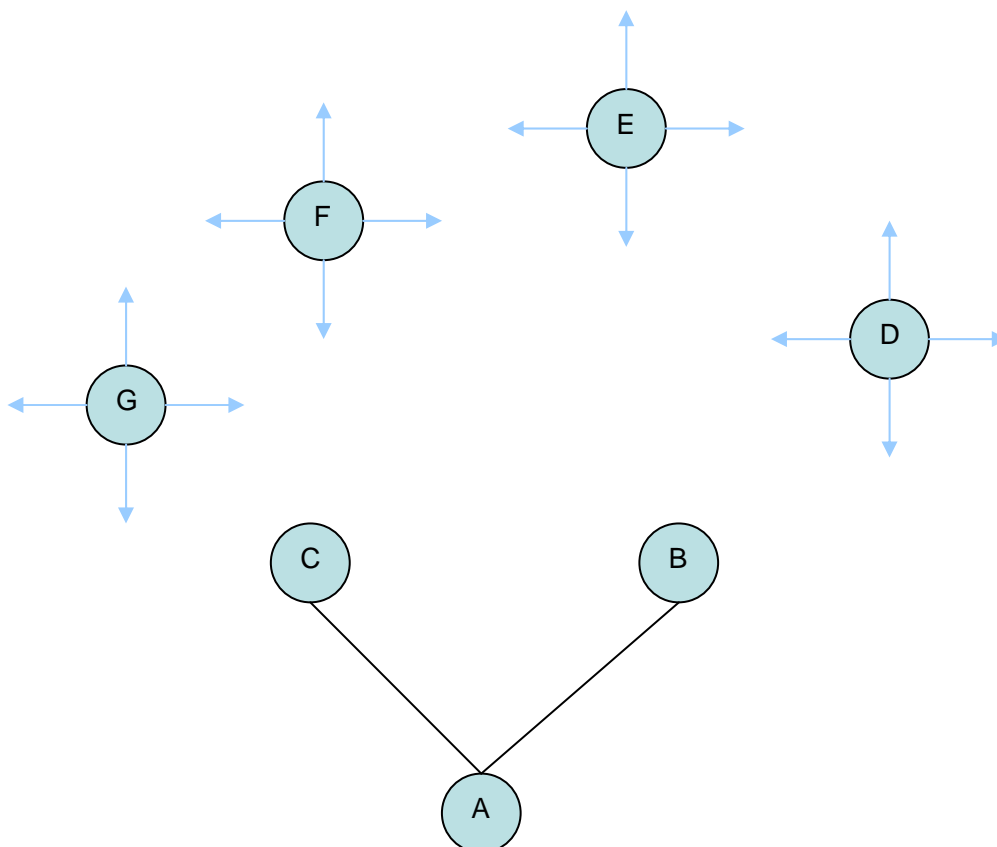
Σχήμα 3-1. Το δέντρο που δημιουργείται με τον αλγόριθμο δρομολόγησης

- Όλοι οι κόμβοι στέλνουν περιοδικά μηνύματα με τα οποία αναζητούν να συνδεθούν πάνω σε ένα κόμβο αισθητήρα ο οποίος είναι ήδη συνδεδεμένος πάνω στο δέντρο. Δηλαδή οι κόμβοι βρίσκονται σκορπισμένοι όπως στο Σχήμα 3-2 και παραλαμβάνουν (αλλά και στέλλουν) περιοδικά μηνύματα από τους γείτονές τους, οι οποίοι προσπαθούν να συνδεθούν στο δέντρο.



Σχήμα 3-2. Μια τοπολογία με τυχαία κατανομή των κόμβων

- Στο βήμα αυτό κόμβος A (sink) που είναι ο μόνος κόμβος που είναι συνδεδεμένος στο δέντρο θα απαντήσει στα μηνύματα που παρέλαβε από τους B και C. Δηλαδή ο κόμβος A είναι ο μόνος υποψήφιος πατέρας στο παράδειγμά μας. Στο μήνυμα αυτό θα τους απαντήσει ότι μπορεί να είναι ο πατέρας τους. Οπότε με αυτό το τρόπο οι κόμβοι B και C θα συνδεθούν στο δέντρο και το δίκτυο μας θα φαίνεται όπως στο Σχήμα 3-3.



Σχήμα 3-3. Η διαδικασία δημιουργίας του δέντρου πριν ακόμα ολοκληρωθεί

- Τώρα είναι η σειρά των κόμβων G, F, E και D να συνδεθούν στο δέντρο, αφού τα δικά τους μηνύματα παραλήφθηκαν από τους κόμβους C και B που τώρα είναι συνδεδεμένοι στο δέντρο και τους απάντησαν αυτοί με τη σειρά τους ότι μπορούν να είναι πατέρες τους. Σε αυτό το σημείο το δέντρο μας δημιουργείται και η τελική του μορφή φαίνεται στο Σχήμα 3-1. Να αναφέρουμε ότι τα μηνύματα που στέλνει κάποιος κόμβος που δεν είναι συνδεδεμένος στο δέντρο, υπάρχει πιθανότητα να παραληφθούν από περισσότερους από ένα κόμβους που είναι συνδεδεμένοι στο δέντρο. Κατά συνέπεια σε αυτή την περίπτωση πρέπει να επιλεγθεί ένας από τους υποψήφιους ο οποίος θα είναι ο πατέρας του κόμβου που προσπαθεί να συνδεθεί. Ο τρόπος με τον οποίο γίνεται η επιλογή μεταξύ των κόμβων που είναι υποψήφιοι πατέρες θα τον δούμε πιο κάτω, όπου γίνεται η περιγραφή της υλοποίησής μας. Στο

Κεφάλαιο 4 θα δείξουμε ένα παράδειγμα με περισσότερους κόμβους και πώς γίνεται πραγματικά η σύνδεση των κόμβων στο δέντρο.

Να αναφέρουμε ότι το δέντρο μας συντηρείται γιατί τα μηνύματα αυτά στέλνονται περιοδικά. Δηλαδή, στην περίπτωση που κάποιος πατέρας αποτύχει (π.χ. λόγω εξάντλησης αποθεμάτων μπαταρίας) τα παιδιά του μπορούν να αναζητήσουν καινούργιο πατέρα για να συνδεθούν. Οι κόμβοι αλλάζουν πατέρα μόνο αν ο πατέρας τους αποτύχει. Επίσης, να αναφέρουμε ότι για να μπορέσουν οι κόμβοι να στέλνουν δεδομένα πρέπει πρώτα να συνδεθούν στο δέντρο. Για την επεξήγηση του αλγορίθμου θα χρησιμοποιούμε τις έννοιες πιο κάτω στο δέντρο (downstream) και θα αναφερόμαστε για την κίνηση από τους κόμβους E και D στον κόμβο B και αντίθετα, πιο πάνω στο δέντρο (upstream) η κίνηση από τον κόμβο B στους κόμβους E και D. Στην επόμενη παράγραφο θα περιγράψουμε το μηχανισμό που χρησιμοποιούμε για να αποφεύγεται η συμφόρηση στο δίκτυο.

3.2 Επεξήγηση Μηχανισμού για Αποφυγή της Συμφόρησης

Για την επεξήγηση του μηχανισμού που χρησιμοποιούμε για την αποφυγή της συμφόρησης θα ξεκινήσουμε βλέποντας την αναλογία ενός οικοσυστήματος με το Ασύρματο Δίκτυο Αισθητήρων. Ένα οικοσύστημα αποτελείται από πολλά είδη, τα οποία ζουν μαζί και αλληλεπιδρούν τόσο μεταξύ τους, όσο και με τους πόρους που βρίσκονται στο περιβάλλον τους, για να μπορέσουν να ικανοποιήσουν τις ανάγκες τους και να συνυπάρξουν. Παρόμοια, παρατηρούμε ένα Δίκτυο Αισθητήρων, όπου υπάρχει ένας μεγάλος αριθμός κόμβων αισθητήρων. Σε ένα τέτοιο δίκτυο κάθε κόμβος έχει το δικό του προσωρινό χώρο αποθήκευσης (buffer) για να μπορεί να αποθηκεύει τα πακέτα του, αλλά και για να μπορεί να αρχίζει τη ροή των δεδομένων του.

Όλες οι ροές δεδομένων συναγωνίζονται η μια την άλλη για τους διαθέσιμους πόρους του δικτύου, έτσι ώστε να μπορέσουν να διοχετεύσουν τα πακέτα τους στο sink διαμέσου των

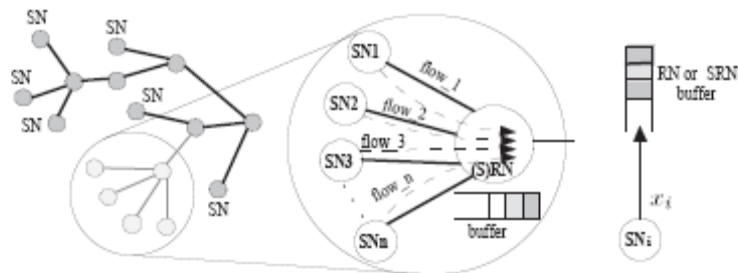
ενδιάμεσων κόμβων. Ακριβώς όπως και στο οικοσύστημα, ο στόχος είναι οι ροές δεδομένων των διαφόρων κόμβων να μπορέσουν να συνυπάρχουν όπως συνυπάρχουν και τα διάφορα είδη.

Για τον υπολογισμό του ρυθμού αποστολής δεδομένων με τη βοήθεια του μηχανισμού για αποφυγή της συμφόρησης κάποιος κόμβος στο δίκτυο μπορεί να έχει διαφορετικούς ρόλους. Ένας κόμβος μπορεί ο ίδιος να αρχίσει μια ροή δεδομένων, οπότε σε αυτή την περίπτωση έχει το ρόλο του Source Node. Ένας κόμβος έχει το ρόλο του Relay Node όταν εξυπηρετεί κάποια ροή δεδομένου κάποιου άλλου κόμβου ή Source-Relay Node όταν εξυπηρετεί και τους δύο ρόλους. Συνήθως τα Source Node βρίσκονται στην άκρη του δικτύου, είναι δηλαδή φύλλα στα δέντρα. Σαν Relay Node είναι οι κόμβοι που είναι ενδιάμεσοι κόμβοι όπως για παράδειγμα οι κόμβοι C και B στο παράδειγμα που είδαμε πιο πάνω.

Ο μηχανισμός που χρησιμοποιούμε εστιάζεται σε μια τεχνική που βασίζεται στην βέλτιστη προσαρμογή του ρυθμού που γίνεται σε κάθε hop για έλεγχο της συμφόρησης και είναι εμπνευσμένος από τη βιολογία, όπως είπαμε και πιο πάνω. Ο μηχανισμός για έλεγχο της συμφόρησης πρέπει να έχει τη δυνατότητα να ρυθμίζει τη ροή των δεδομένων ξεχωριστά για κάθε κόμβο που κάνει αποστολή δεδομένων. Σύμφωνα με το μηχανισμό κάθε κόμβος από μόνος του ρυθμίζει και προσαρμόζει τη ροή δεδομένων του, δηλαδή το ρυθμό με τον οποίο δημιουργεί ή προωθεί τα πακέτα. Όλες οι ροές δεδομένων συναγωνίζονται για το διαθέσιμο χώρο της προσωρινής μνήμης του κόμβου-παραλήπτη που βρίσκεται ένα hop μακριά τους. Δηλαδή, οι κόμβοι E και D συναγωνίζονται για το διαθέσιμο χώρο της προσωρινής μνήμης του πατέρα τους B. Οι κόμβοι πρέπει να καθορίζουν με τέτοιο τρόπο τις ροές δεδομένων τους χρησιμοποιώντας τις ελάχιστες ανάγκες σε χωρητικότητα της προσωρινής μνήμης του πατέρα τους κάθε φορά, έτσι ώστε σε όλη τη διαδρομή προς το sink να αποφεύγονται τα μη επιθυμητά φαινόμενα, όπως η συμφόρηση. Πολλές φορές οι κόμβοι πρέπει να αλληλεπιδρούν μεταξύ τους, όπως και τα διάφορα είδη στο οικοσύστημα, για να μπορούν να κερδίσουν ένα δίκαιο μέρος στη χωρητικότητα της προσωρινής μνήμης του πατέρα τους.

Ο μηχανισμός για αποφυγή της συμφόρησης για τα Ασύρματα Δίκτυα Αισθητήρων, όπως είπαμε και προηγουμένως πρέπει να δίνει τη δυνατότητα στους κόμβους να ρυθμίζουν από μόνοι τους το ρυθμό των ροών δεδομένων τους υιοθετώντας τις αλλαγές του δικτύου ανά πάσα στιγμή. Επίσης, με το μηχανισμό αυτό το δίκτυο πρέπει να λειτουργεί σε ένα αποδεκτό επίπεδο απόδοσης όταν οι απαιτήσεις υπερβαίνουν ή είναι πολύ κοντά στο να υπερβούν τους διαθέσιμους πόρους του δικτύου. Η αποφυγή της συμφόρησης αναμένεται ότι θα αποτρέπει τα μη επιθυμητά φαινόμενα που προκαλεί η συμφόρηση ή αν όχι να προσφέρει μια επιθυμητή μείωση. Επιπλέον, θα κάνει μια δίκαιη κατανομή της προσφερόμενης κίνησης προβάλλοντας μια καλή απόκριση, θα μειώνει τον αριθμό των πακέτων που χάνονται και θα παρέχει μια ικανοποιητική ασφάλεια όσο αφορά την ποιότητα της εφαρμογής που θα χρησιμοποιείται ανάλογα με τις ανάγκες του χρήστη. Με στόχο μια αποδοτική και λειτουργική προσέγγιση θα επιτυγχάνεται μειωμένη κατανάλωση ενέργειας, αύξηση της ευρωστίας του συστήματος για επέκταση του χρόνου ζωής του δικτύου και της διαθεσιμότητάς του. Πάνω σε αυτά θα βασιστούν και οι μετρήσεις που θα δείξουμε στο επόμενο Κεφάλαιο, όπου θα παραθέσουμε τα αποτελέσματα των μετρήσεων μας με στόχο να αποδείξουμε τα πιο πάνω.

Με τη καινούρια προσέγγιση αποφυγής της συμφόρησης που χρησιμοποιείται από το μηχανισμό που υλοποιήσαμε, γίνεται ρύθμιση του ρυθμού των ροών δεδομένων σε κάθε κόμβο χρησιμοποιώντας τις τοπικές του πληροφορίες χωρίς την ανάγκη μιας κεντρικής οντότητας – ρυθμιστή του ρυθμού ροής των δεδομένων. Η ρύθμιση γίνεται ανάλογα με την προσφορά πόρων που δίνεται από το δίκτυο και τη ζήτηση πόρων που γίνεται από τους κόμβους του δικτύου. Επίσης, η ρύθμιση γίνεται σε όλους τους κόμβους από τον κόμβο πηγής μέχρι να φτάσει στον κόμβο sink, πάντα χρησιμοποιώντας τις τοπικές πληροφορίες που μπορεί να πάρει κάποιος κόμβος. Όπως αναφέραμε και πιο πάνω, ένας κόμβος μπορεί να έχει τρεις διαφορετικούς ρόλους (Source Node, Relay Node και Source-Relay Node) τους οποίους θα δούμε αναλυτικότερα πιο κάτω:



Σχήμα 3-4. Οι κόμβοι πηγής συναγωνίζονται για τους περιορισμένους πόρους του πατέρα τους

Source Node (Κόμβος Πηγής - ΚΠ): Οι ΚΠ είναι απλοί κόμβοι, οι τερματικές οντότητες που συνδέονται στο Δίκτυο Αισθητήρων μέσω κάποιου άλλου κόμβου, όπως Relay Node ή Source-Relay Node οι οποίοι είναι τοποθετημένοι πιο μακριά από το sink σε σύγκριση με τους υπόλοιπους κόμβους. Δηλαδή, οι κόμβοι G, F, E και D του Σχήματος 3-1. Παράδειγμα κόμβου πηγής είναι τα φύλλα του δέντρου, όπως μπορούμε να δούμε και στο Σχήμα 3-4 οι οποίοι ονομάζονται SN. Αναμένεται ότι κάθε ΚΠ θα ξεκινά κάποια ροή δεδομένων όταν ενεργοποιείται κάποιο γεγονός. Ο ρυθμός της κάθε ροής υπολογίζεται λύνοντας την Εξίσωση 1, η οποία δίνει τον αριθμό των bytes που στέλνονται x_i για κάθε ροή i . Για να μπορέσουμε να επιλύσουμε την Εξίσωση 1 για ένα κόμβο i πρέπει απαραίτητα να γνωρίζουμε το άθροισμα του αριθμού των

$$\frac{dx_i}{dt} = r_i x_i \left[1 - \frac{\beta_i}{K_i} x_i - \frac{1}{K_i} \left(\sum_{j=1, j \neq i}^n a_{ij} x_j \right) \right] \quad (1)$$

Bytes που στέλνονται από τους υπόλοιπους κόμβους που συναγωνίζονται. Δηλαδή, πρέπει να γνωρίζουμε το $\sum_{j=1, j \neq i}^n x_j$, το οποίο συμβολίζουμε σαν C_i . Αν και η γνώση της πληροφορίας αυτής είναι μη ρεαλιστική σε αποκεντρωτικές αρχιτεκτονικές, ένας κόμβος μπορεί να πάρει έμμεσα την πληροφορία αυτή από μηνύματα που στέλνονται περιοδικά από τον πατέρα στα παιδιά. Δηλαδή, να ενημερωθεί από τον πατέρα του το συνολικό αριθμό δεδομένων που

στέλνονται από όλους τους κόμβους που βρίσκονται πιο κοντά στο sink και αυτό το συμβολίζουμε ως BS. Η ενημέρωση μπορεί να γίνεται είτε μέσω κάποιου άλλου πακέτου (piggybacked), είτε μπορεί να στέλνεται ειδικό πακέτο με μοναδικό σκοπό να μεταφέρει τη συγκεκριμένη πληροφορία. Να τονίσουμε ότι με το δεύτερο τρόπο αυξάνεται ο αριθμός των πακέτων που ανταλλάσσονται στο δίκτυο ενώ στόχος στα Ασύρματα Δίκτυα Αισθητήρων είναι να γίνεται η ελάχιστη ανταλλαγή πληροφοριών. Κάθε κόμβος για να υπολογίσει τη συμβολή των γειτόνων του C_i πρέπει να αφαιρέσει τη δική του συμβολή από τη συνολική συμβολή BS, δηλαδή $C_i = \sum_{j=1, j \neq i}^n x_j = BS - x_i$. Έτσι η Εξίσωση 1 γίνεται:

$$\frac{dx_i}{dt} = rx_i \left(1 - \frac{\beta}{K} x_i - \frac{a}{K} C_i \right), i = 1, \dots, n \quad (2)$$

Για να πάρουμε το x_i , ολοκληρώνουμε την Εξίσωση 2:

$$x_i(t) = \frac{wx_i(0)}{\beta x_i(0) + [w - \beta x_i(0)]e^{-\frac{wr}{K}t}}, w = K - a C_i \quad (3)$$

Σε ένα δίκτυο (ή οικοσύστημα) ροών (ή ειδών) που συναγωνίζονται για ένα πόρο ο αριθμός των bytes που στέλνονται ρυθμίζονται από την Εξίσωση 3. Υπάρχει μια γενική, μη αρνητική και ασυπτοτικά σταθερή ισορροπία όταν ο inter-specific συναγωνισμός είναι πιο αδύνατος από τον intra-specific συναγωνισμό. Για παράδειγμα, $\beta > a$ ($a, \beta > 0$). Κάτω από αυτές τις συνθήκες, η σειρά των τιμών που παράγονται από κάθε ΚΠ συγκλίνει σε μια λύση συνύπαρξης γενική και ασυπτοτικά σταθερή, η οποία δίνεται από την Εξίσωση 4. Λεπτομερής απόδειξη της ιδέας αυτής αναφέρεται στο [2].

$$x_i^* = \frac{K}{a(n-1) + \beta}, i = 1, \dots, n \quad (4)$$

Η εγκυρότητα της Εξίσωσης 3 βασίζεται στην υπόθεση ότι το $K - a C_i > x_i$. Επιλέγουμε να θέσουμε το $a = 1$, γι' αυτό σύμφωνα με την ανισότητα απαιτούμε ο αριθμός

των bytes που στέλνονται από κάθε κόμβο i να μην ξεπερνά τον άδειο χώρο που μένει στην προσωρινή μνήμη του κόμβου που βρίσκεται πάνω από τον κόμβο μας ($K - C_i$) έτσι ώστε να αποφεύγεται η υπερχειλίση της προσωρινής του μνήμης. Η ικανοποίηση του περιορισμού αυτού μπορεί να καθοριστεί στη βάση της Εξίσωσης 4. Για να αποφύγουμε την υπερχειλίση της προσωρινής μνήμης πρέπει να είμαστε σίγουροι ότι όταν ένα σύστημα από n ενεργούς κόμβους συγκλίνει σε μια λύση συνύπαρξης κάθε κόμβος i θα μπορεί να στέλνει λιγότερα ή ίσα από K/n bytes. Χρησιμοποιώντας την Εξίσωση 4 αυτό επιτυγχάνεται όταν $\alpha(n - 1) + \beta \geq n$. Θέτοντας το $\alpha = 1$ ο περιορισμός γίνεται $\beta > 1$, το οποίο ικανοποιεί τη συνθήκη για την ύπαρξη μιας λύσης γενικής και ασυμπτικά σταθερής.

Κάθε ΚΠ, ο οποίος είναι μια συσκευή περιορισμένων πόρων και μη ικανή να χειριστεί την πολυπλοκότητα, αποτιμά την Εξίσωση 3 σε μια επαναληπτική διαδικασία για να ελέγχει το ρυθμό αποστολής για κάθε ροή σε πραγματικό χρόνο. Η επαναληπτική διαδικασία αναφέρεται στον αριθμό των πακέτων που στέλνονται σε μια περίοδο T . Για παράδειγμα, στην επανάληψη $k + 1$ στο χρόνο $(k + 1)T$ η τιμή $x((k + 1)T)$ υπολογίζεται με βάση την προηγούμενη τιμή $x(kT)$, δηλαδή $x((k + 1)T) = f(x(kT))$. Η επαναληπτική διαδικασία της Εξίσωσης 3 εκφράζεται ως:

$$x_i((k + 1)T) = \frac{w(kT)x_i(kT)}{\beta x_i(kT) + [w(kT) - \beta x_i(kT)]e^{-\frac{w(kT)r_T}{K}}} \quad (5)$$

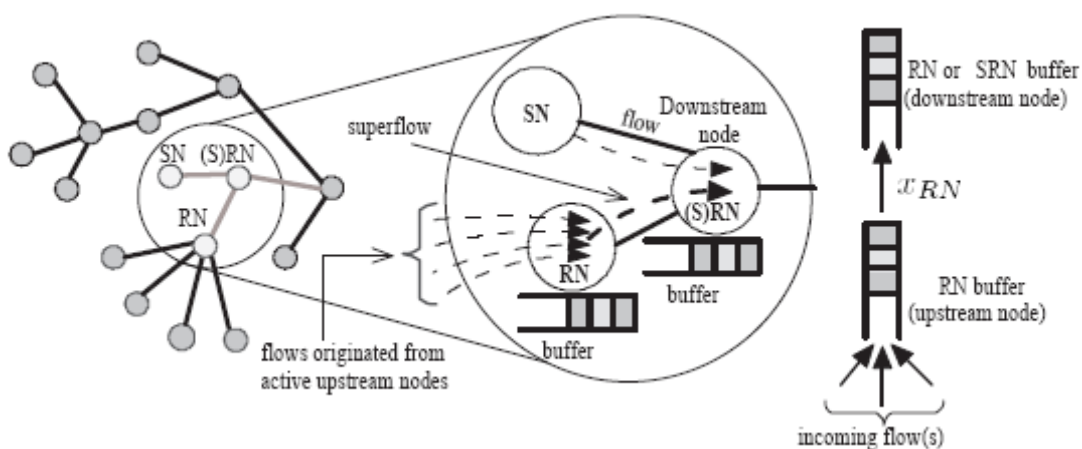
Ο υπολογισμός αυτός πρέπει να γίνει γρήγορα, με περιορισμένη χρησιμοποίηση της μνήμης και χαμηλή κατανάλωση ενέργειας. Ο υπολογισμός του εκθετικού παράγοντα της Εξίσωσης 5 είναι μια διαδικασία κατανάλωσης χρόνου, μνήμης και ενέργειας, η οποία απαιτεί από το λειτουργικό σύστημα κάθε κόμβου να υλοποιεί μια συγκεκριμένη βιβλιοθήκη μαθηματικών. Εναλλακτικά, αυτός ο υπολογισμός μπορεί να εφαρμοστεί με το πολυώνυμο του Taylor βαθμού n γύρω στο $T = t_0$, όπως ακολουθεί:

$$e^{-\frac{wr}{K}T} \sim e^{-\frac{wr}{K}t_0} \sum_{k=0}^n \left(-\frac{wr}{K}\right)^k \frac{(T-t_0)^k}{k!} \quad (6)$$

Όπου το t_0 είναι το ακέραιο τμήμα του T . Η παράμετρος T αναμένεται να εμπίπτει σε ένα μικρό πεδίο πραγματικών τιμών, για παράδειγμα 0.1 – 2 δευτερόλεπτα. Για αυτό το λόγο κάθε κόμβος στη μνήμη του πρέπει να έχει ένα πίνακα που περιέχει $e^{-(wr/k)t_0}$ για ένα περιορισμένο αριθμό τιμών από $(wr t_0)/K$. Η απόδοση μιας καλής προσέγγισης της εκθετικής συνάρτησης αν αυτή εφαρμοστεί στους n πρώτους όρους της σειράς Taylor, εξαρτάται από το αν το υπόλοιπο είναι πραγματικά μικρό. Ο υπόλοιπος όρος δείχνει τη διαφορά μεταξύ του πολυώνυμου του Taylor βαθμού n και την πραγματική εκθετική συνάρτηση. Πρέπει να σημειωθεί ότι αν το T πάρει ακέραια τιμή τότε το e^x είναι ίσο με τις σειρές του Taylor. Για το μηχανισμό που χρησιμοποιούμε υπολογίζουμε το πολυώνυμο 4^{ου} βαθμού. Ένα άνω όριο του λάθους μας δίνει η φόρμα του Lagrange:

$$\text{Λάθος Προσέγγισης} \leq R_4(T) = \frac{w^5 r^5 e^{-\frac{wr}{K}t_0} (T-t_0)^5}{5! K^5} \quad (7)$$

Παραγοντοποιώντας το άνω όριο του R_4 μπορούμε να θέσουμε ένα όριο για τη μεταβλητή r . Οι μεταβλητές K και T έχουν προκαθορισμένες τιμές. Τις τιμές των μεταβλητών αυτών θα τις δούμε σε επόμενο Κεφάλαιο.



Σχήμα 3-5. Ο Κόμβος Σύνδεσης δημιουργεί μια δική του ροή η οποία

συναγωνίζεται για την προσωρινή μνήμη του κόμβου – πατέρα του

Relay Node (Κόμβος Σύνδεσης - ΚΣ): Οι ΚΣ είναι ενδιάμεσοι κόμβοι οι οποίοι συνδέονται είτε με άλλους ΚΣ είτε με ΚΠ. Κάθε ΚΣ λειτουργεί απλά σαν ΚΣ, δεν δημιουργεί δικά του πακέτα απλά προωθεί πακέτα που ανήκουν σε διάφορες ροές και προωθούνται από αυτό τον κόμβο. Οι ροές αυτές συναγωνίζονται για τους πόρους του ΚΣ.

Η κύρια λειτουργία του ΚΣ είναι να ενώνει τις διάφορες ροές δεδομένων που φτάνουν σε αυτό σε μια ροή δεδομένων, την οποία συνδέει με τον κόμβο που βρίσκεται πιο κάτω του, δηλαδή πιο κοντά στο sink, όπως μπορούμε να δούμε και στο Σχήμα 3-5. Σαν επόμενο βήμα ο ΚΣ πιθανώς θα συναγωνιστεί με άλλες ροές δεδομένων για τους πόρους του κόμβου που βρίσκεται πιο κάτω. Δηλαδή ο κόμβος E συναγωνίζεται με τον κόμβο D για τον προσωρινό χώρο μνήμης του κόμβου που βρίσκεται πιο κάτω τους δηλαδή του B, όπως μπορούμε να δούμε από το Σχήμα 3-1. Κάθε ΚΣ επιλύει τη γενική εξίσωση του μοντέλου Lotka – Volterra. Με αυτό τον τρόπο υπολογίζει το ρυθμό με τον οποίο θα στείλει τα πακέτα του στον κόμβο – πατέρα του. Αντίθετα με τον ΚΠ, φαίνεται ότι είναι πιο δύσκολο να διαμοιραστούν δίκαια οι πόροι του κόμβου που βρίσκεται πιο κάτω επειδή η πιο κάτω ροή αποτελείται από πολλαπλές ροές. Για παράδειγμα, στο Σχήμα 3-5 κάθε μια από τις τέσσερις ροές αποτελούν την πιο κάτω ροή του κόμβου με την ονομασία RN. Η ροή που δημιουργήθηκε αρχικά από το ΚΠ πρέπει να δεσμεύσει ίσο μέρος από τους περιορισμένους πόρους του πιο κάτω κόμβου όπως και οι υπόλοιπες ροές του ΚΣ. Για αυτό κάθε ΚΣ πρέπει να δρα προς συμφέρον όλων των ενεργών ροών που διαπερνούν τον ΚΣ, υπολογίζοντας το ρυθμό αποστολής της πιο κάτω ροής (αριθμός των bytes που στέλνονται από το ΚΣ σε μια περίοδο T). Για την ακρίβεια, κάθε ΚΣ υπολογίζει με μια φόρμα τη δέσμευση πόρων όλων των ενεργών κόμβων που βρίσκονται πιο πάνω από το ΚΣ χρησιμοποιώντας μια ελαφρώς αλλαγμένη έκφραση της Εξίσωσης 5, η οποία είναι:

$$x_{RN}((k+1)T) = n \left(\frac{w(kT)H(kT)}{\beta H(kT) + [w(kT) - \beta H(kT)]e^{-\frac{w(kT)T}{K}}} \right) \quad (8)$$

Όπου $H(kT) = \frac{x_{RN}(kT)}{n}$, $w(kT) = K - aC_{RN}^*(kT)$ και n είναι ο συνολικός αριθμός των ενεργών κόμβων που βρίσκονται πιο πάνω από το ΚΣ, η οποία ανήκει στο δέντρο που έχει σαν ρίζα το ΚΣ. Στην πραγματικότητα ο ΚΣ πρέπει να στέλνει n περισσότερες φορές bytes σε σύγκριση με μια απλή ροή. Κάθε ΚΣ μπορεί να υπολογίσει τον αριθμό των ενεργών κόμβων (n) που βρίσκονται πιο πάνω του εξετάζοντας το πεδίο πηγής των πακέτων που τον διαπερνούν.

Ο αριθμός των bytes που στέλνονται από την πιο κάτω ροή σε μια περίοδο kT , ονομάζεται $x_{RN}(kT)$ και είναι ίση με το σύνολο του αριθμού των bytes που στέλνονται από τους n ΚΠ που είναι πιο πάνω από το ΚΣ και συναγωνίζονται τη προσωρινή του μνήμη (ο υπολογισμός γίνεται με την Εξίσωση 8). Το $C_{RN}^*(kT)$ απεικονίζει το συνολικό αριθμό των bytes που στέλνονται (BS) στον πιο κάτω κόμβο (ο κόμβος ΚΣ(Π) του Σχήματος) από όλους τους κόμβους που είναι ένα hop μακριά και συναγωνίζονται για τους πόρους του αφαιρώντας τη συνεισφορά μιας ροής που ανήκει στην πιο κάτω ροή. Το $C_{RN}^*(kT)$ εκφράζεται σαν :

$$C_{RN}^* = BS - \frac{x_{RN}(kT)}{n} \quad (9)$$

Η Εξίσωση 8 δίνει τη δυνατότητα σε κάθε ΚΣ να δεσμεύει τους πόρους του πιο κάτω κόμβου δίκαια για κάθε μια από τις ροές που βρίσκονται πιο πάνω από το ΚΣ. Αν ο υπολογιζόμενος ρυθμός αποστολής της πιο κάτω ροής ξεπερνά το συνολικό ρυθμό αποστολής όλων των εισερχόμενων ροών που αποτελούν τη ροή αυτή, αναμένεται ότι ένας αριθμός πακέτων θα συσσωρευτούν στην προσωρινή μνήμη του ΚΣ. Ενώ, ο διαθέσιμος χώρος στη προσωρινή μνήμη του ΚΣ θα μειωθεί προκαλώντας τους πιο πάνω κόμβους να μειώσουν

προοδευτικά το ρυθμό αποστολής τους χρησιμοποιώντας είτε την Εξίσωση 5 για τους ΚΠ είτε την Εξίσωση 8 για τους ΚΣ/ΚΣΠ.

Source-Relay Node (Κόμβος Σύνδεσης – Πηγής -ΚΣΠ): Ένας ΚΣΠ λειτουργεί και σαν ΚΠ αλλά και σαν ΚΣ, χρησιμοποιώντας και τις δύο λειτουργικότητες που είδαμε πιο πάνω ταυτόχρονα.

3.3 Περιγραφή Υλοποίησης

Στην παράγραφο αυτή θα κάνουμε αναλυτική περιγραφή του αλγορίθμου δρομολόγησης που υλοποιήσαμε και επίσης, θα περιγράψουμε πώς χρησιμοποιούμε τις εξισώσεις που είδαμε πιο πάνω για να κάνουμε έλεγχο για αποφυγή της συμφόρησης. Αρχικά να αναφέρουμε ότι στον αλγόριθμο δρομολόγησης χρησιμοποιήσαμε το μικρότερο δυνατό αριθμό μηνυμάτων που χρειάζεται ο αλγόριθμος που προτείνουμε (χρησιμοποιούνται 2 είδη μηνυμάτων και στέλνονται όταν είναι απαραίτητο για τη λειτουργία του πρωτοκόλλου) με το μικρότερο δυνατό μέγεθος για να εξυπηρετούμε τους περιορισμούς των Ασύρματων Δικτύων Αισθητήρων. Επίσης, δεν υπάρχει κάποιος κόμβος-συντονιστής στο δίκτυο, αλλά ο κόμβος sink είναι ο πρώτος που υπάρχει μέσα στο δίκτυο (ο πρώτος που θεωρείται συνδεδεμένος) και δύναται να στέλνει πακέτα προς αναζήτηση γειτονικών κόμβων. Οι υπόλοιποι κόμβοι συνδέονται διαδοχικά στο δίκτυο αφού επιλέξουν ένα πατέρα από τους είδη συνδεδεμένους κόμβους. Με τον τρόπο αυτό κτίζεται το δίκτυο και κάθε νεοεισελθής κόμβος αφού επιλέξει τον πατέρα του γίνεται μέρος του δικτύου. Στο δίκτυο ένας κόμβος που είναι ήδη συνδεδεμένος ενεργοποιείται όταν είναι σε θέση να στέλνει δεδομένα (τη διαδικασία θα τη δούμε πιο κάτω).

Στον αλγόριθμο χρησιμοποιούνται δύο διαφορετικά μηνύματα. Το πρώτο είναι το Request For Child (RQFC) και έχει τα εξής πεδία: τη διεύθυνση του κόμβου που το στέλνει, τη διεύθυνση του κόμβου παραλήπτη, τον αριθμό των παιδιών του, τον αριθμό των hops από το sink, την ώρα που στέλνεται το πακέτο, το χρόνο που θα πρέπει να υπολογίσει το ρυθμό

του με σκοπό να συγχρονιστεί με τους υπόλοιπους στο δέντρο. Επίσης, στο πακέτο αυτό υπάρχουν και κάποια άλλα πεδία τα οποία χρησιμοποιούνται για να θέτονται κάποιες μεταβλητές μεταξύ των κόμβων. Αυτό το μήνυμα στέλνεται σε όλους τους κόμβους, δηλαδή γίνεται broadcast σε όλους για να μπορέσουν οι κόμβοι που δεν είναι συνδεδεμένοι στο δίκτυο να συνδεθούν και οι κόμβοι που είναι συνδεδεμένοι στο δίκτυο είτε να ανανεώσουν τις πληροφορίες που έχουν από τον πατέρα τους, είτε να ενημερωθούν ότι ο πατέρας τους ακόμα υπάρχει. Δηλαδή, αν για οποιοδήποτε λόγο ο πατέρας τους δεν υπάρχει πλέον για να μπορέσουν να επιλέξουν άλλο πατέρα. Επίσης, το μήνυμα αυτό στέλνεται από κάποιον κόμβο όταν γίνει ενεργός για να ενημερώσει τον πατέρα του για την κατάσταση του (από μη ενεργός σε ενεργός) και στη συνέχεια, να του σταλεί η απάντηση για να μπορεί να ξεκινήσει να στέλνει και δεδομένα. Το μήνυμα αυτό χρησιμοποιείται σε περισσότερες από μία περιπτώσεις με στόχο τη χρήση λίγων πακέτων, έτσι ώστε να γίνεται γρήγορα και αποδοτικά η διαχείριση των μηνυμάτων που φτάνουν σε κάποιο κόμβο.

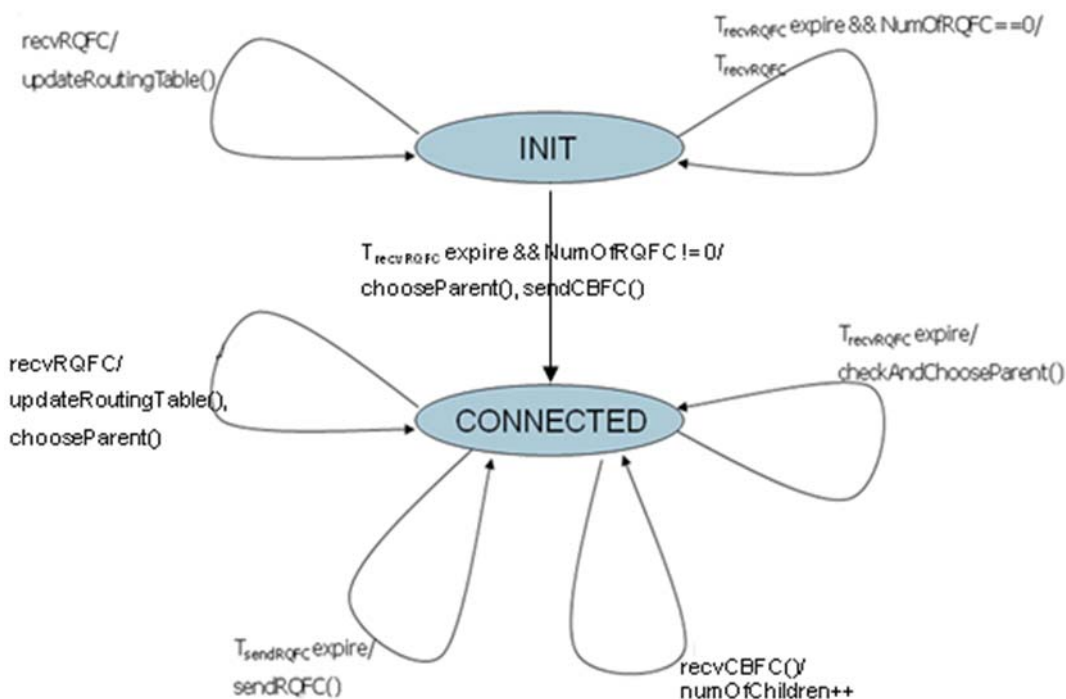
Το δεύτερο μήνυμα, είναι το Communication Between Father and Child (CBFC) με τα εξής πεδία: τη διεύθυνση του κόμβου που το στέλνει, τη διεύθυνση του κόμβου παραλήπτη, την ώρα που στέλνεται το πακέτο καθώς και ένα πεδίο που χρησιμοποιείται και πάλι για να θέτονται κάποιες μεταβλητές μεταξύ των κόμβων. Το μήνυμα αυτό στέλνεται από κάποιο υποψήφιο παιδί στον πατέρα του για να τον ενημερώσει ότι αποδέχεται να τον επιλέξει σαν πατέρα του. Όταν κάποιος κόμβος παραλάβει αυτό το μήνυμα τότε προσθέτει τον αποστολέα στη λίστα με τα παιδιά του. Επίσης, το μήνυμα αυτό χρησιμοποιείται από τον πατέρα σε κάποιο παιδί του για να το ενημερώσει ότι μπορεί να ξεκινήσει να στέλνει δεδομένα.

Εκτός από τα δύο είδη μηνυμάτων χρησιμοποιούνται και κάποιοι χρονιστές (timers). Δύο από τους timers χρησιμοποιούνται για να συνδεθούν οι κόμβοι αισθητήρων και να αποτελέσουν ένα συνδεδεμένο δέντρο (όπως το είδαμε και στο Σχήμα 3-1) που θα είναι στη συνέχεια το δίκτυο στο οποίο θα μεταφέρονται τα μηνύματα και οι άλλοι timer χρησιμοποιούνται για να ελέγχεται η συχνότητα με την οποία στέλνονται τα μηνύματα. Και τα δύο είδη timer θα τα δούμε αναλυτικότερα πιο κάτω.

Στο πρώτο είδος που αναφέραμε πιο πάνω υπάρχουν δύο timers. Ο ένας ονομάζεται T_{recvRQFC} και ο άλλος T_{sendRQFC} . Ο πρώτος timer χρησιμοποιείται για δύο λόγους. Ο πρώτος λόγος είναι όταν κάποιος κόμβος δεν είναι συνδεδεμένος στο δέντρο παίρνει λίγο χρόνο για να μαζέψει πληροφορίες από τους κόμβους-γείτονες του έτσι ώστε να μπορέσει να επιλέξει τον καλύτερο δυνατό πατέρα. Ο δεύτερος λόγος που χρησιμοποιείται από τους κόμβους που είναι συνδεδεμένοι για να τους δίνεται η δυνατότητα περιοδικά να ελέγχουν τους γείτονες τους οι οποίοι μπορεί να είναι πατέρας τους και τελικά, να επιλέγουν τον καλύτερο αλλά και να ελέγχουν τον αριθμό των παιδιών του κάθε γείτονα που είναι ενεργά (που στέλλουν δεδομένα). Ο timer T_{sendRQFC} χρησιμοποιείται από τους κόμβους που είναι συνδεδεμένοι στο δέντρο για να στέλλουν περιοδικά το μήνυμα RQFC για να μπορούν οι κόμβοι που δεν είναι συνδεδεμένοι στο δέντρο να συνδεθούν. Αυτό το μήνυμα στέλνεται από το sink αρχικά για να μπορέσει έτσι να εντοπίσει τους υπόλοιπους κόμβους και να δημιουργηθεί τελικά το δέντρο.

Οι timer του δεύτερου είδους, όπως αναφέρουμε σε προηγούμενη παράγραφο χρησιμοποιούνται για να ελέγχεται ο ρυθμός με τον οποίο στέλνονται τα πακέτα. Βασικά η διαδικασία που ακολουθούν όλοι οι κόμβοι είναι η εξής: δημιουργούν τα πακέτα που θέλουν να στείλουν και τα αποθηκεύουν στην ουρά τους. Επίσης, στην ουρά αυτή προσθέτονται και τα πακέτα που φτάνουν σε ένα κόμβο με σκοπό να μεταφερθούν σε άλλο κόμβο, δηλαδή ο κόμβος αυτός είναι ο ενδιάμεσος σταθμός για να φτάσουν στον προορισμό τους. Σύμφωνα με τον ένα timer περιοδικά υπολογίζεται ο ρυθμός με τον οποίο θα στέλνουν τα πακέτα τους ανάλογα με το ρόλο που έχει ο κάθε κόμβος. Παίρνοντας και πάλι το δέντρο του Σχήματος 3-1 οι κόμβοι D, E, F και G θα υπολογίζουν το ρυθμό τους σύμφωνα με την Εξίσωση 5 εφόσον όλοι στέλλουν δεδομένα. Οι κόμβοι B και C υπολογίζουν το ρυθμό τους με την Εξίσωση 8 αν οι ίδιοι δεν αποστέλλουν δεδομένα. Στην περίπτωση που και οι ίδιοι αποστέλλουν δεδομένα τότε υπολογίζουν και τις δύο Εξισώσεις (5 και 8) για να ικανοποιούν και τους δύο ρόλους τους ως Κόμβοι Πηγής και ως Κόμβοι Σύνδεσης. Δηλαδή, ο κάθε κόμβος υπολογίζει το ρυθμό με τον οποίο αποστέλλει τα δεδομένα του με σκοπό να αποφεύγεται η συμφόρηση. Οι κόμβοι που δε στέλλουν οι ίδιοι δεδομένα και δεν έχουν παιδιά που να στέλλουν δεδομένα δεν κάνουν υπολογισμό του ρυθμού αποστολής δεδομένων (δε χρειάζεται να γίνει

υπολογισμός). Ο άλλος timer χρησιμοποιείται για να ρυθμίζει την αποστολή πακέτων στους κόμβους. Δηλαδή, ο πρώτος timer υπολογίζει πόσα πακέτα μπορεί να στείλει κάποιος κόμβος σε μια περίοδο και ο δεύτερος timer υπολογίζει το ρυθμό με τον οποίο θα στέλνονται τα πακέτα σε αυτή την περίοδο.



Σχήμα 3-6. Σχεδιάγραμμα αλγορίθμου

Οι πληροφορίες που κρατούν όλοι οι κόμβοι τοπικά είναι ο αριθμός των hops τους από το sink, ο αριθμός των παιδιών τους, ο αριθμός των μηνυμάτων RQFC που παρέλαβαν, το χρόνο που παρέλαβαν το τελευταίο μήνυμα RQFC, τον πατέρα τους, το ρυθμό αποστολής δεδομένων του πατέρα τους, σε ποια από τις δύο καταστάσεις είναι (οι δύο καταστάσεις φαίνονται στο πιο κάτω Σχήμα 3-6 και επίσης, πιο κάτω περιγράφεται η χρησιμότητά τους) και μια λίστα με τα παιδιά τους. Εκτός από τις πληροφορίες που έχουν τοπικά οι κόμβοι

αισθητήρες χρησιμοποιούν και ένα πίνακα πληροφοριών που περιλαμβάνει τον αριθμό των παιδιών ενός κόμβου και τον αριθμό των hops από το sink.

Ο αλγόριθμος, όπως φαίνεται και στο Σχήμα 3-6, αποτελείται από δύο καταστάσεις την Init και την Connected. Δηλαδή, οι κόμβοι στο δίκτυο μπορούν να βρεθούν σε δύο καταστάσεις. Όλοι οι κόμβοι είναι αρχικοποιημένοι να βρίσκονται στην κατάσταση Init εκτός από το sink που βρίσκεται στην κατάσταση Connected. Στην κατάσταση Connected μπορούν να βρεθούν όταν αποκτήσουν πατέρα, δηλαδή όταν συνδεθούν στο δίκτυο.

Αρχικά το sink, αφού μόνο αυτό είναι στην κατάσταση Connected, στέλνει σε όλους τους κόμβους το μήνυμα RQFC. Το μήνυμα αυτό στη συνέχεια στέλνεται από τους κόμβους που βρίσκονται στην κατάσταση Connected, έτσι ώστε να μπορέσουν και οι δικό τους γείτονες που δεν είναι συνδεδεμένοι στο δίκτυο να συνδεθούν. Επίσης, όλοι οι κόμβοι για να ξεκινήσουν να στέλνουν δεδομένα πρέπει να παραλάβουν το ανάλογο μήνυμα από τον πατέρα τους. Για τη διαδικασία αυτή χρησιμοποιείται και πάλι το μήνυμα RQFC (αυτή είναι η δεύτερη χρήση του πακέτου RQFC, την πρώτη την είδαμε σε προηγούμενη παράγραφο) στον πατέρα τους (επιπρόσθετο από το περιοδικό μήνυμα που στέλνεται) για να πάρουν άδεια και έτσι να ξεκινήσουν να στέλνουν δεδομένα. Δηλαδή κάποιος κόμβος αρχικά πρέπει να συνδεθεί στο δέντρο και στη συνέχεια, για να μπορέσει να στείλει δεδομένα πρέπει να του επιτραπεί από τον πατέρα του.

Όταν κάποιος κόμβος είναι στην κατάσταση Init και ο timer του T_{recvRQFC} λήξει ελέγχει τον αριθμό των μηνυμάτων RQFC που έχει παραλάβει και αν αυτός είναι μηδέν τότε απλά μηδενίζει το timer και περιμένει ξανά. Αν ο αριθμός είναι διάφορος του μηδέν τότε επιλέγει πατέρα, στέλνει το μήνυμα CBFC στον πατέρα που επέλεξε και υπολογίζει το χρόνο στον οποίο θα πρέπει να υπολογίσει το ρυθμό του. Τον τρόπο επιλογής του πατέρα θα το δούμε πιο κάτω. Επίσης, όταν κάποιος κόμβος βρίσκεται στην κατάσταση Init και παραλάβει το μήνυμα RQFC, αρχικά από το sink και σε επόμενο στάδιο από κάποιους άλλους κόμβους γείτονες του που είναι συνδεδεμένοι στο δίκτυο, ανανεώνει τις πληροφορίες που έχει αποθηκευμένες στον πίνακα πληροφοριών του.

Σε περίπτωση που κάποιος κόμβος βρίσκεται στην κατάσταση Connected, δηλαδή είναι συνδεδεμένος στο δίκτυο και παραλάβει το μήνυμα RQFC αρχικά θα πρέπει να ελέγξει από ποιο κόμβο παρέλαβε το μήνυμα. Αν ο κόμβος είναι πατέρας του τότε ανανεώνει τις τοπικές μεταβλητές με τις πληροφορίες που παρέλαβε. Αν ο κόμβος είναι κάποιο από τα αδέρφια του τότε ανανεώνει και πάλι τις ανάλογες τοπικές μεταβλητές. Και στις δύο περιπτώσεις ανανεώνει και τις πληροφορίες του πίνακα πληροφοριών. Αν ένας κόμβος παραλάβει ένα μήνυμα RQFC επιπλέον από τα μηνύματα που στέλνονται περιοδικά τότε πρέπει να στείλει επιπρόσθετο μήνυμα RQFC στον πατέρα του και να ενημερώσει τις τοπικές του μεταβλητές. Αν ένας πατέρας είναι ήδη ενεργός για να στέλνει δεδομένα τότε απαντάει στο παιδί του μέσω του μηνύματος CBFC ότι μπορεί να ξεκινήσει να στέλνει δεδομένα.

Όταν κάποιος κόμβος που δεν του επιτρέπεται να στέλνει δεδομένα παραλάβει το μήνυμα CBFC τότε απλά αυξάνει τον αριθμό των παιδιών του, ανανεώνει τις πληροφορίες του πίνακα πληροφοριών και στέλλει ένα επιπρόσθετο μήνυμα RQFC για να ενημερώσει άμεσα την αύξηση του αριθμού των παιδιών έτσι ώστε κάποιος κόμβος που προσπαθεί να επιλέξει τη συγκεκριμένη στιγμή πατέρα να ενημερωθεί. Το μήνυμα αυτό στέλνεται με σκοπό οι κόμβοι να έχουν πάντα την πιο ανανεωμένη πληροφορία για τον αριθμό των παιδιών του πατέρα. Αν κάποιος κόμβος που μπορεί να στέλνει δεδομένα παραλάβει το πακέτο CBFC από τον πατέρα του τότε ξεκινά να υπολογίζει το ρυθμό με τον οποίο μπορεί να στέλνει πακέτα και με τη σειρά του ενημερώνει τα παιδιά του ότι μπορούν να στέλνουν δεδομένα.

Όταν ο timer του T_{recvRQFC} λήξει, επιλέγει καινούριο πατέρα με τη διαδικασία που θα δούμε πιο κάτω και στέλνει το μήνυμα CBFC και πάλι μόνο αν ο δικός του πατέρας έχει αποτύχει για παράδειγμα λόγω έλλειψης της μπαταρίας. Τέλος, όταν ο timer του T_{sendRQFC} λήξει τότε στέλνεται το μήνυμα RQFC με σκοπό να ενημερώσει τους κόμβους γείτονες του ότι ακόμα υπάρχει αφού σε περίπτωση που κάποιος κόμβος δεν παραλάβει αυτό το μήνυμα για τρεις περιόδους θα καταλάβει ότι ο συγκεκριμένος κόμβος για οποιοδήποτε λόγο δεν υπάρχει πλέον στο δίκτυο.

Για την επιλογή του πατέρα ακολουθείται η πιο κάτω διαδικασία. Χρησιμοποιούμε μια εξίσωση της μορφής της Εξίσωσης 10,

$$Y_i = Ax_i + Bz_i, A+B=1 \quad (10)$$

όπου η μεταβλητή x_i αντιστοιχεί στον αριθμό των παιδιών του υποψήφιου πατέρα i που διαιρείται με το συνολικό αριθμό παιδιών των υποψήφιων πατέρων στην εμβέλεια ενός κόμβου, και η μεταβλητή z_i αντιστοιχεί στον αριθμό hops του υποψήφιου πατέρα i διαιρώντας τον με το συνολικό αριθμό hops των υποψήφιων πατέρων. Οι μεταβλητές A και B επηρεάζουν τον αριθμό των παιδιών και το βάθος του δέντρου αντίστοιχα. Δηλαδή, αν θέλουμε το δέντρο μας να κτίζεται με μεγάλο βάθος τότε το B πρέπει να είναι μεγαλύτερο από το A και το αντίθετο αν θέλουμε το δέντρο μας να είναι πλατύ και όχι βαθύ, τότε το A είναι μεγαλύτερο από το B . Για τον κόμβο i με την μικρότερη τιμή y_i που υπολογίζουμε βρίσκουμε ένα τυχαίο αριθμό που εμπίπτει στους πιο πάνω περιορισμούς και τότε αυτός επιλέγεται σαν πατέρας. Με αυτό τον τρόπο μπορούμε να κτίζουμε δέντρα ανάλογα με το αν θέλουμε να είναι βαθιά ή πλατιά αλλάζοντας απλά τις τιμές στις δύο μεταβλητές (A και B) που αναφέραμε πιο πάνω.

Τελειώνοντας, να αναφέρουμε ότι ο μηχανισμός για αποφυγή της συμφόρησης και ο αλγόριθμος για το κτίσιμο του δέντρου που είδαμε πιο πάνω, είναι και τα δύο σχεδιασμένα λαμβάνοντας υπόψη τους περιορισμούς των Ασύρματων Δικτύων Αισθητήρων, όπως η περιορισμένη ενέργεια κ.α.. Στο επόμενο Κεφάλαιο θα δούμε κάποια πειράματα που υλοποιήσαμε για να δείξουμε την αποδοτικότητα του αλγορίθμου κάτω από τους περιορισμούς των Ασύρματων Δικτύων Αισθητήρων καθώς και κάποιες άλλες μετρικές που να αποδεικνύουν τόσο την καλή λειτουργία του αλγορίθμου όσο και τους περιορισμούς τους.

Κεφάλαιο 4

Πειραματικό Μέρος

Στο πειραματικό μέρος της εργασίας, εξομοιώνεται ένα δίκτυο και μετρούνται κάποια από τα μεγέθη που το χαρακτηρίζουν. Η εξομοίωση γίνεται με το πρόγραμμα Network Simulator (NS-2), το οποίο θα δούμε αναλυτικότερα στο Appendix A. Το μοντέλο δικτύου που χρησιμοποιούμε αποτελείται από ασύρματους κόμβους αισθητήρων και ασύρματη σύνδεση για την μεταξύ τους επικοινωνία. Όλοι οι κόμβοι είναι οι ίδιοι και μπορεί να είναι είτε κόμβοι σύνδεσης είτε κόμβοι πηγής είτε ακόμα μπορούν να έχουν και τους δύο ρόλους όπως είδαμε στο προηγούμενο Κεφάλαιο.

Στην αξιολόγηση μας θα δούμε ένα μέρος που αφορά τον αλγόριθμο δρομολόγησης, ένα ακόμα μέρος που αφορά την αξιολόγηση του μηχανισμού και τέλος, θα αξιολογήσουμε την υλοποίηση μας σαν ένα ενιαίο κομμάτι. Με στόχο να αξιολογήσουμε την απόδοση του πρωτοκόλλου καθώς και άλλες μετρικές δημιουργήσαμε διάφορα πειράματα τα οποία θα δούμε αναλυτικότερα στη συνέχεια του Κεφαλαίου. Εδώ να σημειώσουμε ότι τη γραφική απεικόνιση των αποτελεσμάτων που θα δούμε πιο κάτω έγινε με τη χρήση του Matlab.

| | |
|----------------------------------|-------------------|
| Channel type | WirelessChannel |
| Radio propagation model | TwoRayGround |
| Network interface type | WirelessPhy |
| MAC type | 802.11 |
| MAC data rate | 1Mbps |
| Interface queue (IFQ) type | DropTail/PriQueue |
| Link layer type | LL |
| Antenna model | OmniAntenna |
| Maximum number of packets in IFQ | 700 |

Πίνακας 4-1. Βασικές ρυθμίσεις του NS-2 για τις προσομοιώσεις

4.1 Περιβάλλον Προσομοίωσης

Η εξομοίωση μας αποτελείται μόνο από ασύρματους κόμβους αισθητήρων. Ο αριθμός τους αλλάζει ανάλογα με την προσομοίωση που τρέχουμε. Επίσης, σε κάποια από τα πειράματα που θα δούμε πιο κάτω οι κόμβοι τοποθετούνται είτε σε τυχαίες θέσεις είτε σε καθορισμένες. Στον Πίνακα 4-1 φαίνονται κάποια από τα χαρακτηριστικά που χρησιμοποιούνται σε όλες τις προσομοιώσεις μας μαζί με τις τιμές με τις οποίες είναι αρχικοποιημένες. Σε όλα τα πειράματα που τρέξαμε σαν πρωτόκολλο δρομολόγησης χρησιμοποιούσαμε το πρωτόκολλο που δημιουργήσαμε εμείς και το οποίο ενσωματώνει και το μηχανισμό για την αποφυγή της συμφόρησης.

Όλοι οι κόμβοι που βρίσκονται στο δίκτυο έχουν σαν στόχο αρχικά να συνδεθούν με το sink είτε απευθείας, αν υπάρχει δυνατότητα για απευθείας σύνδεση είτε μέσω των γειτόνων τους. Επίσης, όλοι οι κόμβοι που ενεργοποιούνται για να αποστέλλουν δεδομένα τα αποστέλλουν με προορισμό το sink χρησιμοποιώντας τη σύνδεση που έχουν με το sink

(απευθείας ή μέσω των γειτόνων τους). Δηλαδή με την τοπολογία του Σχήματος 4-1, αν ο κόμβος με αριθμό 16 ενεργοποιηθεί να στέλνει δεδομένα στο sink θα τα αποστέλλει μέσω του κόμβου με αριθμό 20. Αν όμως ο κόμβος 20 ενεργοποιηθεί για να αποστέλλει δεδομένα στο sink θα τα στέλνει απευθείας. Στην επόμενη ενότητα θα δούμε πως οι παράμετροι A και B επηρεάζουν τη δημιουργία του δέντρου.

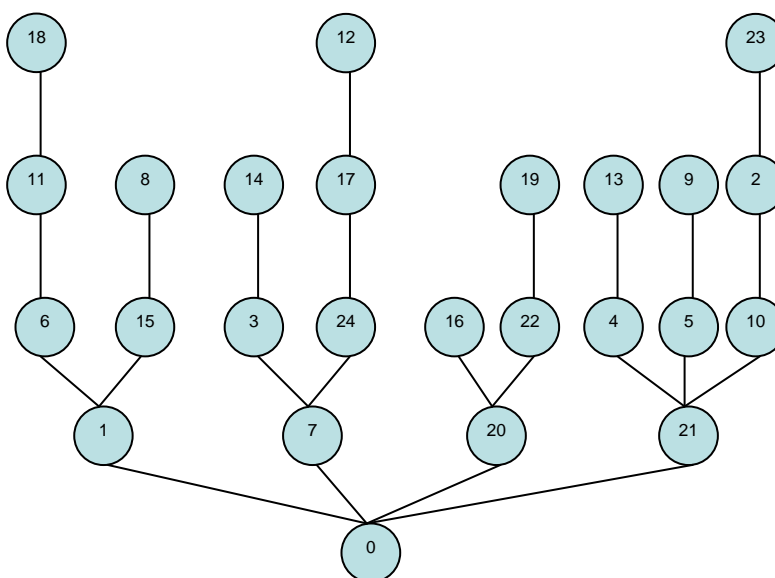
4.2 Δημιουργία Δέντρου

Στο προηγούμενο Κεφάλαιο είδαμε αναλυτικά και τα δύο μέρη από τα οποία αποτελείται το πρωτόκολλο μας. Το πρώτο μέρος αφορά τη δρομολόγηση των πακέτων, δηλαδή πως οι κόμβοι συνδέονται μεταξύ τους για να μπορούν να στέλνουν τα πακέτα τους με προορισμό τον κόμβο sink. Το δεύτερο μέρος αφορά το μηχανισμό για αποφυγή της συμφόρησης, δηλαδή είδαμε πως οι κόμβοι πρέπει να ελέγχουν το ρυθμό με τον οποίο αποστέλλουν τα πακέτα τους έτσι ώστε να αποφεύγεται η συμφόρηση στους κόμβους αισθητήρων και όλες οι αρνητικές συνέπειες που προκαλούνται από το χάσιμο πακέτων. Στην ενότητα αυτή θα δούμε πως οι παράγοντες A και B της Εξίσωσης 10 που είδαμε στην Ενότητα 3-3 επηρεάζουν το κτίσιμο του δέντρου. Στον Πίνακα 4-2 βλέπουμε το βάθος και το πλάτος των δέντρων όταν προσομοιώσαμε ένα πείραμα όπου είχαμε 25 κόμβους αισθητήρων να τοποθετούνται σε τυχαίες θέσεις σε ένα τετράγωνο 1000 * 1000 μέτρων και αλλάζαμε μόνο τις τιμές των μεταβλητών A και B.

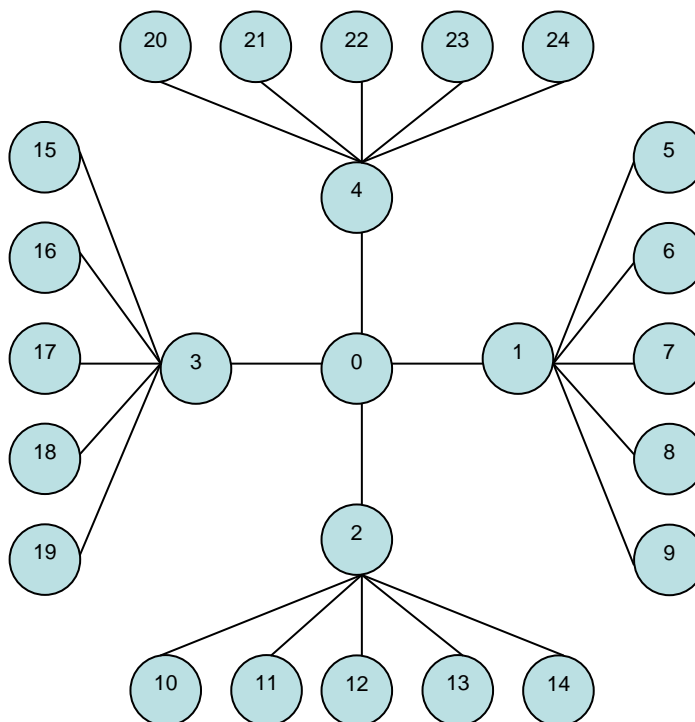
| A | B | Πλάτος | Βάθος |
|-----|-----|--------|-------|
| 0 | 1 | 2 | 11 |
| 0.1 | 0.9 | 3 | 7 |
| 0.3 | 0.7 | 2 | 7 |
| 0.5 | 0.5 | 4 | 4 |

| | | | |
|-----|-----|----|---|
| 0.7 | 0.3 | 5 | 4 |
| 0.9 | 0.1 | 7 | 3 |
| 1 | 0 | 11 | 3 |

Πίνακας 4-2. Αποτελέσματα παραγόντων για τη δημιουργία δέντρου



Σχήμα 4-1. Δέντρο 25 κόμβων με $A=0.5$ και $B=0.5$



Σχήμα 4-2. Καθορισμένη τοπολογία 25 κόμβων

Ενδεικτικά στο Σχήμα 4-1 θα δούμε τη δημιουργία του δέντρου όταν το A ήταν ίσο με 0.5 και το B ίσο με 0.5. Η αλλαγή στις τιμές των μεταβλητών A και B μπορεί να μας δώσει το πλάτος (αριθμός παιδιών) και το βάθος (απόσταση από το sink) του δέντρου που θέλουμε να δημιουργήσουμε ανάλογα με την εφαρμογή που θέλουμε να το χρησιμοποιήσουμε. Στο σημείο αυτό να αναφέρουμε ότι για τα υπόλοιπα πειράματα που θα δούμε πιο κάτω χρησιμοποιήσαμε την τοπολογία που φαίνεται στο Σχήμα 4-2. Στα υπόλοιπα πειράματα κρατούμε σταθερή την τοπολογία με $A=1.0$ και $B=0.0$, και μεταβάλλουμε τις τιμές των υπόλοιπων παραμέτρων για να μελετήσουμε το μηχανισμό αποφυγής της συμφόρησης.

Από τον Πίνακα 4-2 παρατηρούμε ότι όταν το A είναι μεγάλο και το B μικρό τότε το δέντρο που δημιουργείται έχει μεγάλο βάθος και μικρό πλάτος. Όσο το A μειώνεται ενώ το B αυξάνεται τότε συμβαίνει ακριβώς το αντίθετο. Δηλαδή, το βάθος μειώνεται και το πλάτος αυξάνεται. Στο Σχήμα 4-1 όπου το A και το B ισούνται με 0.5, το δέντρο που δημιουργείται έχει ίσο βάθος και πλάτος. Στο Σχήμα 4-2 όπου το A ισούται με 1 και το B ισούται με 0. Το βάθος του δέντρου που δημιουργείται είναι ίσο με 2 και το πλάτος ισούται με 5. Ο τρόπος με τον οποίο είναι τοποθετημένοι οι κόμβοι στο Σχήμα 4-2 είναι διαχωρισμένοι σε ομάδες (clusters), όπου ο αρχηγός κάθε ομάδας από τις τέσσερις ομάδες που σχηματίστηκαν είναι οι κόμβοι 1, 2, 3 και 4.

4.3 Μηχανισμός Συμφόρησης

Στην ενότητα αυτή θα δούμε τα αναλυτικά αποτελέσματα από τα πειράματα όπου αλλάζαμε τις διάφορες παραμέτρους που αφορούν το μηχανισμό μας για αποφυγή της συμφόρησης. Όπως έχουμε αναφέρει και στην προηγούμενη παράγραφο για τα πειράματα μας δημιουργήσαμε μια τοπολογία όπως του Σχήματος 4-2 η οποία αποτελείται από 25 κόμβους αισθητήρων. Η αξιολόγηση του μηχανισμού γίνεται σε στατικό και χωρίς αποτυχίες περιβάλλον. Όλοι οι κόμβοι έχουν την ίδια χωρητικότητα προσωρινής μνήμης, δηλαδή $K =$

35KB και έχουμε θέσει την περιοδικότητα του συστήματος T σε 1 δευτερόλεπτο. Δηλαδή, αυτός είναι ο χρόνος στον οποίο γίνεται ο υπολογισμός του ρυθμού με τον οποίο οι κόμβοι θα στέλνουν τα δεδομένα τους. Και επίσης, με βάση το χρόνο αυτό στέλνονται και τα περιοδικά μηνύματα που είδαμε στο προηγούμενο Κεφάλαιο. Όπως έχουμε αναφέρει ο αλγόριθμος αφορά εφαρμογές με συνεχή ροή δεδομένων με καθορισμένα επίπεδα και σε κάθε επίπεδο υπάρχει καθορισμένο διαφορετικό CBR. Είναι σημαντικό να αναφέρουμε ότι για να μπορούν οι κόμβοι να συγκλίνουν είναι απαραίτητο να ικανοποιούνται οι συνθήκες $\beta > \alpha$ όπου α και β να είναι μεγαλύτερα του 0 και $\alpha(n-1)+\beta \geq n$. Η τιμή στην οποία συγκλίνουν οι κόμβοι υπολογίζεται από την εξίσωση $x_i^* = \frac{K}{\alpha(n-1)+\beta}$ (Εξίσωση 4 του Κεφαλαίου 3). Τις συνθήκες αυτές καθώς και την τιμή στην οποία οι κόμβοι συγκλίνουν θα τα δούμε αναλυτικά στα πειράματα που ακολουθούν.

| Αριθμός Κόμβου | Χρόνος Ενεργοποίησης |
|----------------|-------------------------|
| 5 | 3.9 |
| 6 | 73.9 |
| 7 | 180.9 |
| 8 | 187.9 |
| 9 | 450.9 |
| 10 | 143.9 |
| 11 | 390.9 |
| 12 | 280.9 |
| 13 | 240.9 |
| 14 | 203.9 |
| 15 | 240.9 |
| 16 | 90.9 |

| | |
|----|-------|
| 17 | 340.9 |
| 18 | 410.9 |
| 19 | 190.9 |
| 20 | 155.9 |
| 21 | 340.9 |
| 22 | 110.9 |
| 23 | 310.9 |
| 24 | 350.9 |

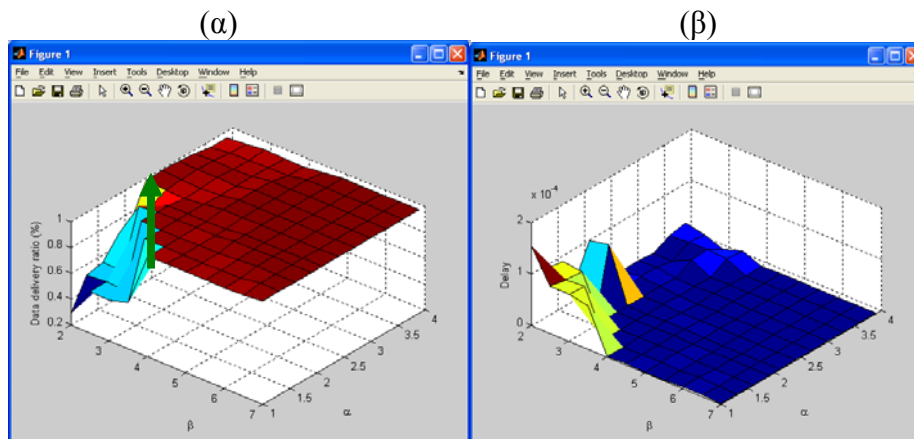
Πίνακας 4-3. Χρονική στιγμή ενεργοποίησης κόμβων αισθητήρων

| Αριθμός Πειράματος | Αριθμός Ενεργών Κόμβων | Ενεργοί Κόμβοι |
|--------------------|------------------------|--|
| 1 | 3 | 5, 6, 10 |
| 2 | 5 | 5, 6, 10, 13,14 |
| 3 | 7 | 5, 6, 10, 12, 13, 14, 21 |
| 4 | 10 | 5, 6, 10, 11, 12, 13, 14, 18, 21, 24 |
| 5 | 15 | 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 18, 21, 23, 24 |
| 6 | 20 | 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 |

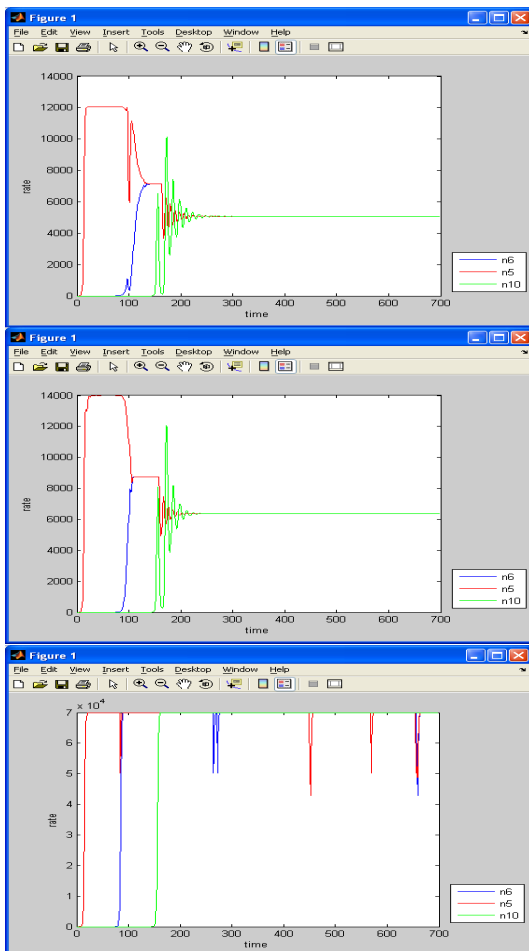
Πίνακας 4-4. Περιγραφή πειραμάτων που υλοποιήσαμε

Στον Πίνακα 4-3 θα δούμε τους κόμβους που ενεργοποιούνται καθώς και τη χρονική στιγμή που ενεργοποιούνται οι διάφοροι κόμβοι σε όλα τα πειράματα που θα δούμε πιο κάτω. Η χρονική στιγμή στην οποία ενεργοποιούνται οι κόμβοι είναι η στιγμή στην οποία οι κόμβοι ξεκινούν να στέλνουν δεδομένα με προορισμό τον κόμβο sink, δηλαδή ο κόμβος με αριθμό 0. Επίσης, στον Πίνακα 4-3 δεν υπάρχουν οι κόμβοι από 0 έως 4 γιατί οι κόμβοι αυτοί δεν

ενεργοποιούνται ποτέ. Όλοι οι κόμβοι που ενεργοποιούνται πρέπει να είναι στα άκρα του δικτύου. Στον Πίνακα 4-4 βλέπουμε τα διάφορα πειράματα που υλοποιήσαμε. Δηλαδή, βλέπουμε τον αριθμό των ενεργών κόμβων καθώς και ποιο ήταν οι ενεργοί κόμβοι σε κάθε πείραμα.



Σχήμα 4-3. (α) Data Delivery Ratio για το Πείραμα 1 του Πίνακα 4-4 (3 ενεργοί κόμβοι)
 (β) Delay για το Πείραμα 1 του Πίνακα 4-4



| | | | |
|----------|---------|------------------|------------------------------|
| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
| 2,0 | 2,9 | ✓ | $6.9 \geq 3$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{6.9} = 5072.5$$

(α)

| | | | |
|----------|---------|------------------|------------------------------|
| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
| 1.5 | 2,5 | ✓ | $5.5 \geq 3$ ✓ |

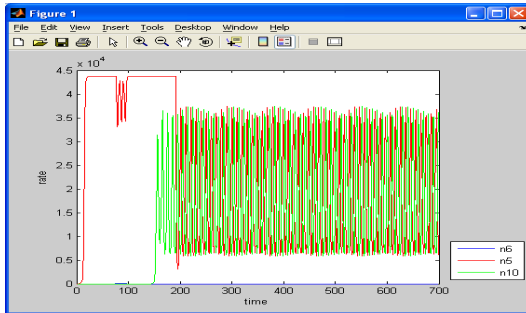
$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{5.5} = 6363.6$$

(β)

| | | | |
|----------|---------|------------------|------------------------------|
| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
| 0.25 | 0.5 | ✓ | $1 \geq 3$ X |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{1} = 35000$$

(γ)



| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 1.0 | 0.8 | X | $2.8 \geq 3$ X |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{2.8} = 12500$$

(δ)

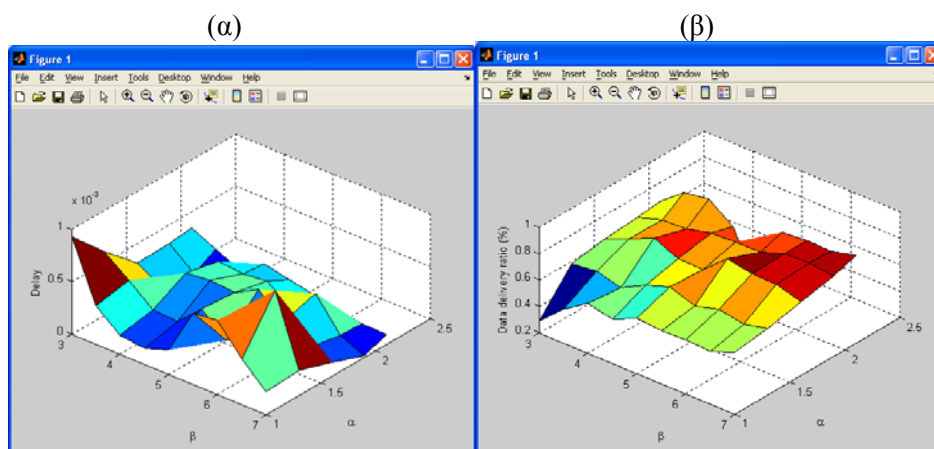
Σχήμα 4-4. Ρυθμός σύγκλισης για 3 ενεργούς κόμβους

Στις γραφικές παραστάσεις του Σχήματος 4-3 μεταβάλλουμε τις τιμές των μεταβλητών α και β και παρατηρούμε το Data Delivery Ratio και το Delay για το Πείραμα 1 που είδαμε στον Πίνακα 4-4 όπου ο αριθμός των ενεργών κόμβων είναι 3. Το Data Delivery Ratio υπολογίζεται με βάση το συνολικό αριθμό πακέτων δεδομένων που παραλήφθηκαν από το sink διαιρώντας τον αριθμό πακέτων που στάλθηκαν από τους ενεργούς κόμβους. Στο σημείο αυτό πρέπει να αναφέρουμε ότι σε όλα τα πειράματα όλοι οι κόμβοι του δικτύου μας που αποστέλλουν δεδομένα σταματούν να δημιουργούν πακέτα δεδομένων για λίγο χρονικό διάστημα πριν να τελειώσει η προσομοίωση έτσι ώστε να μπορέσουν να μεταφερθούν όλα τα πακέτα δεδομένων που υπάρχουν στο δίκτυο στο sink. Και το delay για κάθε πακέτο υπολογίζεται αφαιρώντας από τη χρονική στιγμή που παραλήφθηκε το πακέτο τη χρονική στιγμή που δημιουργήθηκε. Προσθέτουμε τα delay όλων των πακέτων δεδομένων που παραλήφθηκαν από τον κόμβο 0 και διαιρούμε με το συνολικό αριθμό των πακέτων.

Στο Πείραμα 1 του Σχήματος 4-3 παρατηρούμε ότι όσο αυξάνονται οι τιμές των μεταβλητών α και β αυξάνεται το Data Delivery Ratio ενώ το Delay μειώνεται. Αυτό εξηγείται με βάση την Εξίσωση (4) που είδαμε στο Κεφάλαιο 3. Δηλαδή, όσο πιο μικρές είναι οι τιμές των παραμέτρων α και β η τιμή στην οποία το σύστημα συγκλίνει είναι μεγαλύτερη. Άρα όσο λιγότερα πακέτα υπάρχουν στο δίκτυο τόσο μεγαλύτερη είναι η τιμή του Data Delivery Ratio και τόσο μικρότερη είναι το Delay, και το αντίστροφο.

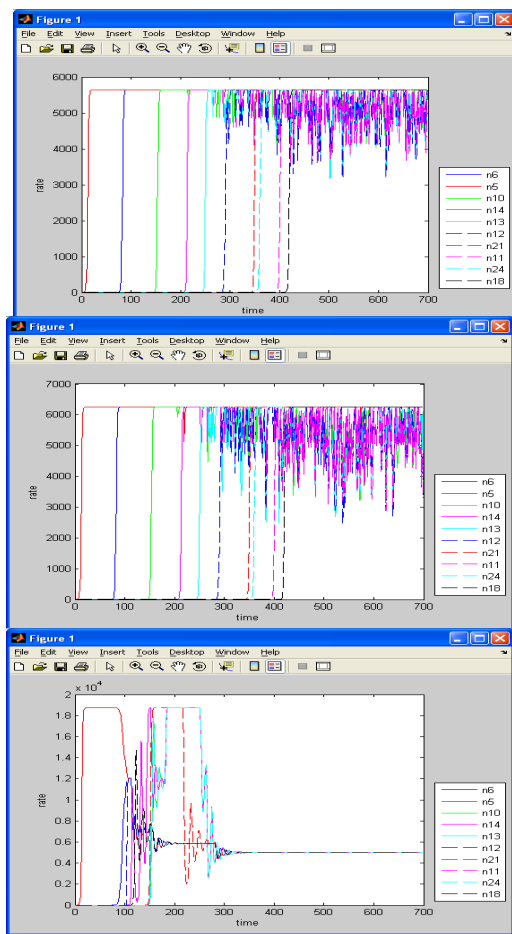
Στις γραφικές παραστάσεις του Σχήματος 4-4 βλέπουμε το ρυθμό στον οποίο συγκλίνει το σύστημα κάτω από διαφορετικές τιμές των παραμέτρων α και β . Στη γραφική παράσταση 4-4(α) βλέπουμε το ρυθμό σύγκλισης όταν και οι δύο συνθήκες του μηχανισμού

επαληθεύονται, για τις μικρότερες τιμές α και β για τις οποίες έχουμε τη μεγαλύτερη τιμή του Packet Delivery Ratio. Δηλαδή, πήραμε ένα συνδυασμό τιμών α και β που ανήκουν πάνω στην κίτρινη γραμμή της γραφικής παράστασης του Σχήματος 4-3(α). Στη γραφική παράσταση του Σχήματος 4-4(β) βλέπουμε το ρυθμό σύγκλισης για κάποιο συνδυασμό τιμών των παραμέτρων α και β για τις οποίες η τιμή του Data Delivery Ratio είναι μικρότερο αλλά και πάλι επαληθεύονται οι δύο συνθήκες. Στις γραφικές παραστάσεις των Σχημάτων 4-4(γ) και 4-4(δ) βλέπουμε το ρυθμό σύγκλισης όταν η μια συνθήκη δεν επαληθεύεται και όταν και οι δύο συνθήκες δε συγκλίνουν, αντίστοιχα. Στην πρώτη περίπτωση όπου η μια συνθήκη δεν επαληθεύεται βλέπουμε ότι υπάρχουν αρκετές αυξομειώσεις στο ρυθμό που δεν παρατηρήθηκαν στις προηγούμενες γραφικές όπου και οι δύο συνθήκες επαληθεύονται. Στην περίπτωση όπου δεν επαληθεύεται καμιά από τις δύο συνθήκες τότε το σύστημα δεν μπορεί να συγκλίνει.



Σχήμα 4-5. (α) Data Delivery Ratio για το Πείραμα 4 του Πίνακα 4-4 (10 ενεργοί κόμβοι)

(β) Delay για το Πείραμα 4 του Πίνακα 4-4



Σχήμα 4-6. Ρυθμός σύγκλισης για 10 ενεργούς κόμβους

| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 1.4 | 6.2 | ✓ | $18.8 \geq 10$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{18.8} = 1861.7$$

(α)

| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 1.7 | 5.6 | ✓ | $20.9 \geq 10$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{20.9} = 1470.6$$

(β)

| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 2.2 | 4.0 | ✓ | $23.8 \geq 10$ ✓ |

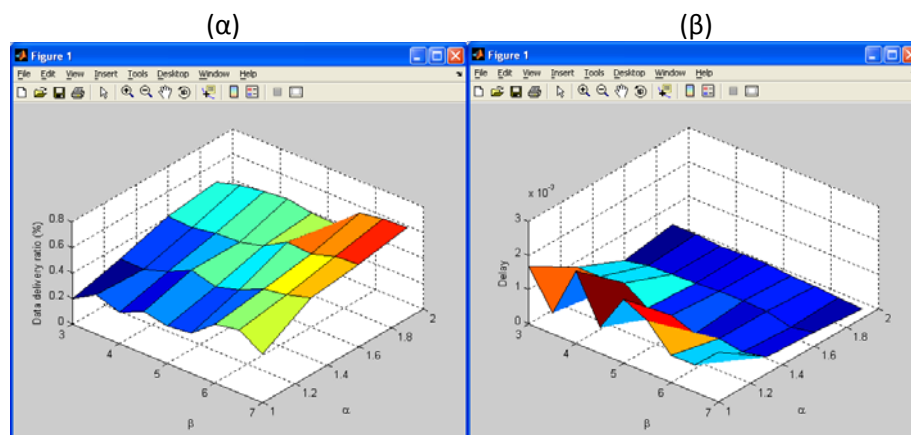
$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{2500 * 50}{23.8} = 5252.1$$

(γ)

Στο Σχήμα 4-5 βλέπουμε τις γραφικές παραστάσεις για το Data Delivery Ratio και το Delay για το Πείραμα 4 του Πίνακα 4-4, όπου ο αριθμός των ενεργών κόμβων είναι 10. Στις γραφικές παρατηρούμε ότι όσο αυξάνονται οι τιμές των μεταβλητών α και β αυξάνεται το Data Delivery Ratio ενώ το Delay μειώνεται. Αυτό εξηγείται όπως και στο προηγούμενο πείραμα με βάση την Εξίσωση (4) που είδαμε στο Κεφάλαιο 3. Δηλαδή, όσο πιο μικρές είναι οι τιμές των παραμέτρων α και β η τιμή στην οποία το σύστημα συγκλίνει είναι μεγαλύτερη. Άρα όσο λιγότερα πακέτα υπάρχουν στο δίκτυο τόσο μεγαλύτερη είναι η τιμή του Data Delivery Ratio και τόσο μικρότερη είναι το Delay, και το αντίστροφο. Επίσης, παρατηρούμε ότι το Packet Delivery Ratio στη γραφική του Σχήματος 4-5 η ψηλότερη τιμή του είναι μικρότερη σε σύγκριση με τη ψηλότερη τιμή του Packet Delivery Ratio στο πείραμα όπου ο αριθμός των ενεργών κόμβων είναι 3, και είδαμε στο Σχήμα 4-3. Όταν ο αριθμός των κόμβων

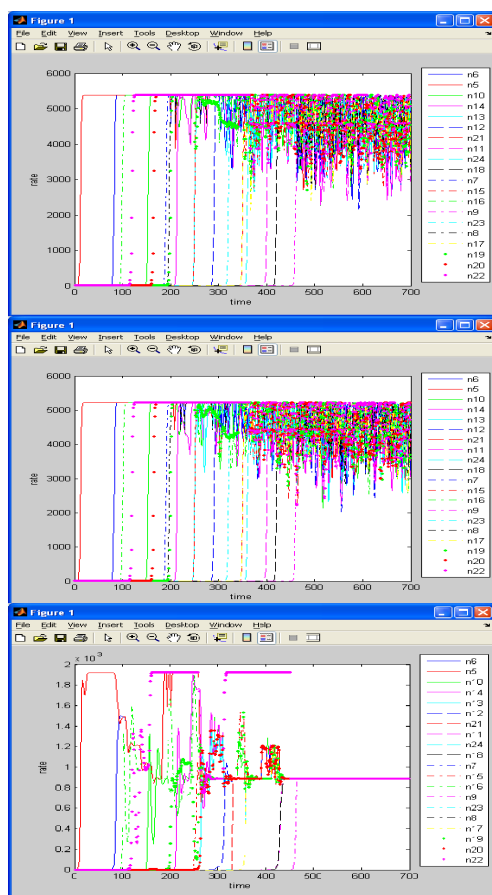
που είναι ενεργοί στο δίκτυο τότε ο ρυθμός στον οποίο συγκλίνουν είναι μικρότερος σύμφωνα με την Εξίσωση 4 που είδαμε στο Κεφάλαιο 3. Παρόλα αυτά τα πακέτα δεδομένων που κυκλοφορούν στο δίκτυο είναι περισσότερα αφού και ο αριθμός των ενεργών κόμβων είναι μεγαλύτερος και αυτός είναι και ο λόγος που η τιμή του Packet Delivery Ratio είναι μικρότερη.

Στις γραφικές παραστάσεις του Σχήματος 4-6 βλέπουμε το ρυθμό στον οποίο συγκλίνει το σύστημα κάτω από διαφορετικές τιμές των παραμέτρων α και β . Στη γραφική παράσταση 4-6(α) βλέπουμε το ρυθμό σύγκλισης όταν και οι δύο συνθήκες του μηχανισμού επαληθεύονται, για τις μικρότερες τιμές α και β για τις οποίες έχουμε τη μεγαλύτερη τιμή του Packet Delivery Ratio. Δηλαδή, πήραμε τις μικρότερες τιμές των παραμέτρων α και β για τις οποίες παίρνουμε τη ψηλότερη τιμή του Packet Delivery Ratio από τη γραφική παράσταση του Σχήματος 4-5(α). Στη γραφική παράσταση του Σχήματος 4-6(β) βλέπουμε το ρυθμό σύγκλισης για κάποιο συνδυασμό τιμών των παραμέτρων α και β για τις οποίες η τιμή του Data Delivery Ratio είναι μικρότερη αλλά και πάλι επαληθεύονται οι δύο συνθήκες. Στις δύο αυτές γραφικές παραστάσεις βλέπουμε ότι οι κόμβοι δε συγκλίνουν στην αναμενόμενη τιμή σύγκλισης αλλά υπάρχουν αυξομειώσεις στο ρυθμό. Οι αυξομειώσεις αυτές οφείλονται στην υπερχειλίση της προσωρινής μνήμης των κόμβων. Σύμφωνα με το μηχανισμό μας για αποφυγή της συμφόρησης απαιτείται να ισχύει η εξής ανισότητα: $\alpha(n - 1) + \beta > n$. Έτσι το β πρέπει να είναι αρκετά μεγαλύτερο του 1 και του α και το α πρέπει να είναι πολύ κοντά στο 1 για να υπάρχει δυνατότητα στον κάθε κόμβο να δεσμεύει λιγότερο χώρο προσωρινής μνήμης από το K / n με σκοπό να αποφεύγεται η μεγάλη χρησιμοποίηση της προσωρινής μνήμης. Στη γραφική παράσταση του Σχήματος 4-6(γ) αυξάνουμε το K από 700 σε 2500, δηλαδή αυξάνουμε το χώρο της προσωρινής μνήμης των κόμβων και πράγματι βλέπουμε ότι οι κόμβοι συγκλίνουν.



Σχήμα 4-7. (α) Data Delivery Ratio για το Πείραμα 6 του Πίνακα 4-4 (20 ενεργοί κόμβοι)

(β) Delay για το Πείραμα 6 του Πίνακα 4-4



| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 1.6 | 6.5 | ✓ | $36.5 \geq 20$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{36.5} = 959$$

(α)

| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 1.7 | 6.7 | ✓ | $39 \geq 20$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{700 * 50}{39} = 897.4$$

(β)

| α | β | $\beta > \alpha$ | $\alpha(n-1) + \beta \geq n$ |
|----------|---------|------------------|------------------------------|
| 2.0 | 7.0 | ✓ | $45 \geq 20$ ✓ |

$$x^* = \frac{K}{\alpha(n-1) + \beta} = \frac{900 * 50}{45} = 1000$$

(γ)

Σχήμα 4-8. Ρυθμός σύγκλισης για 20 ενεργούς κόμβους

Στο Σχήμα 4-7 βλέπουμε τις γραφικές παραστάσεις για το Data Delivery Ratio και το Delay για το Πείραμα 6 του Πίνακα 4-4, όπου ο αριθμός των ενεργών κόμβων είναι 20. Στις

γραφικές παρατηρούμε ότι όσο αυξάνονται οι τιμές των μεταβλητών α και β αυξάνεται το Data Delivery Ratio ενώ το Delay μειώνεται. Αυτό εξηγείται όπως και στα προηγούμενα πειράματα με βάση την Εξίσωση (4) που είδαμε στο Κεφάλαιο 3. Δηλαδή, όσο πιο μικρές είναι οι τιμές των παραμέτρων α και β η τιμή στην οποία το σύστημα συγκλίνει είναι μεγαλύτερη. Άρα όσο λιγότερα πακέτα υπάρχουν στο δίκτυο τόσο μεγαλύτερη είναι η τιμή του Data Delivery Ratio και τόσο μικρότερη είναι το Delay, και το αντίστροφο. Επίσης, παρατηρούμε ότι το Packet Delivery Ratio στη γραφική του Σχήματος 4-7 η ψηλότερη τιμή του είναι μικρότερη σε σύγκριση με τη ψηλότερη τιμή του Packet Delivery Ratio στα πειράματα όπου ο αριθμός των ενεργών κόμβων είναι 3 και 10, και είδαμε στο Σχήμα 4-3 και 4-5, αντίστοιχα. Όταν ο αριθμός των κόμβων που είναι ενεργοί στο δίκτυο τότε ο ρυθμός στον οποίο συγκλίνουν είναι μικρότερος σύμφωνα με την Εξίσωση 4 που είδαμε στο Κεφάλαιο 3, δηλαδή οι κόμβοι στέλνουν μικρότερο αριθμό δεδομένων. Παρόλα αυτά τα πακέτα δεδομένων που κυκλοφορούν στο δίκτυο είναι περισσότερα αφού και ο αριθμός των ενεργών κόμβων είναι μεγαλύτερος και αυτός είναι και ο λόγος που η τιμή του Packet Delivery Ratio είναι μικρότερη.

Στις γραφικές παραστάσεις του Σχήματος 4-8 βλέπουμε το ρυθμό στον οποίο συγκλίνει το σύστημα κάτω από διαφορετικές τιμές των παραμέτρων α και β . Στη γραφική παράσταση 4-8(α) βλέπουμε το ρυθμό σύγκλισης όταν και οι δύο συνθήκες του μηχανισμού επαληθεύονται, για τις μικρότερες τιμές α και β για τις οποίες έχουμε τη μεγαλύτερη τιμή του Packet Delivery Ratio. Δηλαδή, πήραμε τις μικρότερες τιμές των παραμέτρων α και β για τις οποίες παίρνουμε τη ψηλότερη τιμή του Packet Delivery Ratio από τη γραφική παράσταση του Σχήματος 4-7(α). Στη γραφική παράσταση του Σχήματος 4-8(β) βλέπουμε το ρυθμό σύγκλισης για κάποιο συνδυασμό τιμών των παραμέτρων α και β για τις οποίες η τιμή του Data Delivery Ratio είναι μικρότερη αλλά και πάλι επαληθεύονται οι δύο συνθήκες. Στις δύο αυτές γραφικές παραστάσεις βλέπουμε ότι οι κόμβοι δε συγκλίνουν στην αναμενόμενη τιμή σύγκλισης αλλά υπάρχουν αυξομειώσεις στο ρυθμό. Οι αυξομειώσεις αυτές οφείλονται στην υπερχειλίση της προσωρινής μνήμης των κόμβων. Όπως και στο προηγούμενο πείραμα όπου ο αριθμός των ενεργών κόμβων είναι 10 δημιουργήσαμε ένα πείραμα όπου αυξήσαμε το K

από 700 σε 900, δηλαδή αυξάνουμε το χώρο της προσωρινής μνήμης των κόμβων. Ο χώρος της προσωρινής μνήμης αυξήθηκε σε 900 και όχι σε 2500, όπως στο προηγούμενο πείραμα επειδή όπως έχουμε δει από την Εξίσωση 4 του Κεφαλαίου 3 η τιμή του K επηρεάζει την τιμή σύγκλισης. Όσο αυξάνεται η τιμή K αυξάνεται και η τιμή σύγκλισης. Έτσι στο πείραμα όπου ο αριθμός των ενεργών κόμβων είναι 20 η τιμή σύγκλισης με το K να ισούται με 2500 είναι πολύ μεγάλη τιμή και δε μπορούν όλοι οι κόμβοι να συγκλίνουν στην υπολογιζόμενη τιμή σύγκλισης. Στη γραφική παράσταση του Σχήματος 4-8(γ) πράγματι βλέπουμε ότι οι κόμβοι συγκλίνουν όταν αυξάνεται ο χώρος της προσωρινής μνήμης των κόμβων.

Αυτό που παρατηρούμε για όλα τα πειράματα είναι ότι όσο αυξάνονται οι τιμές των μεταβλητών a και b αυξάνεται το Data Delivery Ratio ενώ το Delay μειώνονται επίσης, μειώνεται και ο ρυθμός με τον οποίο αποστέλλουν τα δεδομένα τους οι κόμβοι στο δίκτυο. Αυτό όπως είδαμε και πιο πάνω οφείλεται στην Εξίσωση 4 του Κεφαλαίου 3.

| a | b | r | K | Active Nodes | Packet Delivery Ratio |
|----------|----------|----------|----------|---------------------|------------------------------|
| 1.6 | 3.3 | 1 | 700 | 3 | 0.99 |
| 1.5 | 4.5 | 1 | 700 | 5 | 0.99 |
| 1.7 | 6.0 | 1 | 700 | 7 | 0.88 |
| 1.4 | 6.2 | 1 | 700 | 10 | 0.70 |
| 1.6 | 6.5 | 1 | 700 | 15 | 0.64 |
| 1.9 | 6.5 | 1 | 700 | 20 | 0.62 |

Πίνακας 4-5. Τιμές των μεταβλητών a και b για την ψηλότερη τιμή του Packet Delivery Ratio με χώρο προσωρινής μνήμης = 700

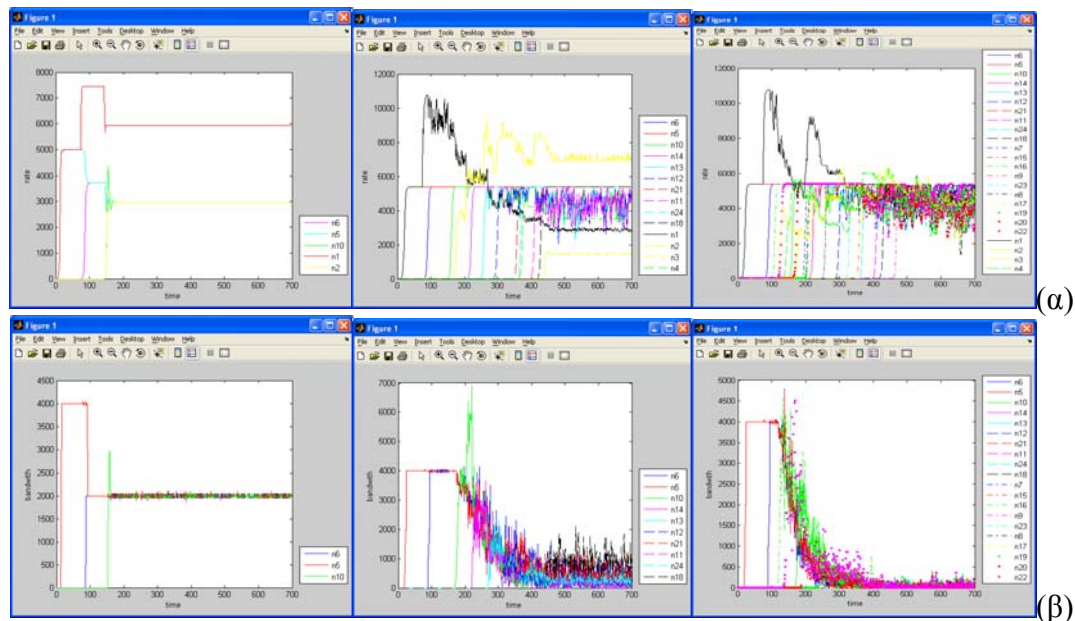
| a | b | r | K | Active Nodes | Packet Delivery Ratio |
|----------|----------|----------|----------|---------------------|------------------------------|
| 1.6 | 3.3 | 1 | 700 | 3 | 0.99 |
| 1.5 | 4.5 | 1 | 700 | 5 | 0.99 |
| 1.7 | 6.0 | 1 | 700 | 7 | 0.96 |
| 1.4 | 6.2 | 1 | 700 | 10 | 0.87 |
| 1.6 | 6.5 | 1 | 700 | 15 | 0.76 |
| 1.9 | 6.5 | 1 | 700 | 20 | 0.76 |

Πίνακας 4-6. Τιμές των μεταβλητών a και b για την ψηλότερη τιμή του Packet Delivery Ratio με χώρο προσωρινής μνήμης = 3000

Στον Πίνακα 4-5 βλέπουμε συγκεντρωτικά για όλα τα Πειράματα τις τιμές των παραμέτρων που επηρεάζουν το μηχανισμό για αποφυγή της συμφόρησης όταν παρατηρούμε την καλύτερη (ψηλότερη) τιμή του Packet Delivery Ratio. Η τιμή της παραμέτρου K και της παραμέτρου Maximum number of packets in IFQ που είδαμε στον Πίνακα 4-1 ισούνται με 700 (όπως και σε όλα τα πιο πάνω πειράματα). Αυτό που παρατηρούμε είναι ότι όσο αυξάνεται ο αριθμός των ενεργών κόμβων η καλύτερη τιμή του Packet Delivery Ratio μειώνεται. Αυτό οφείλεται στο γεγονός ότι όσο αυξάνονται οι ενεργοί κόμβοι στο δίκτυο αυξάνονται και τα δεδομένα που υπάρχουν στο δίκτυο άρα χαμηλότερη τιμή του Packet Delivery Ratio παρόλο που σύμφωνα με την Εξίσωση 4 από το Κεφάλαιο 3 στέλλουν λιγότερα δεδομένα ο κάθε κόμβος ξεχωριστά. Στον Πίνακα 4-5 παρατηρούμε επίσης, ότι όσο ο αριθμός των ενεργών κόμβων αυξάνεται, αυξάνεται και η διαφορά μεταξύ των παραμέτρων α και β . Αυτό εξηγείται σύμφωνα με το μηχανισμό μας για αποφυγή της συμφόρησης από την εξής ανισότητα: $\alpha(n - 1) + \beta > n$. Το β πρέπει να είναι αρκετά μεγαλύτερο του 1 και του α και το α πρέπει να είναι πολύ κοντά στο 1 για να υπάρχει δυνατότητα στον κάθε κόμβο να δεσμεύει λιγότερο χώρο προσωρινής μνήμης από το K/n με σκοπό να αποφεύγεται η μεγάλη χρησιμοποίηση της προσωρινής μνήμης. Έτσι όσο αυξάνεται ο αριθμός των ενεργών κόμβων η διαφορά στις παραμέτρους α και β αυξάνεται για να έχουμε λιγότερα χαμένα πακέτα άρα ψηλότερες τιμές του Packet Delivery Ratio.

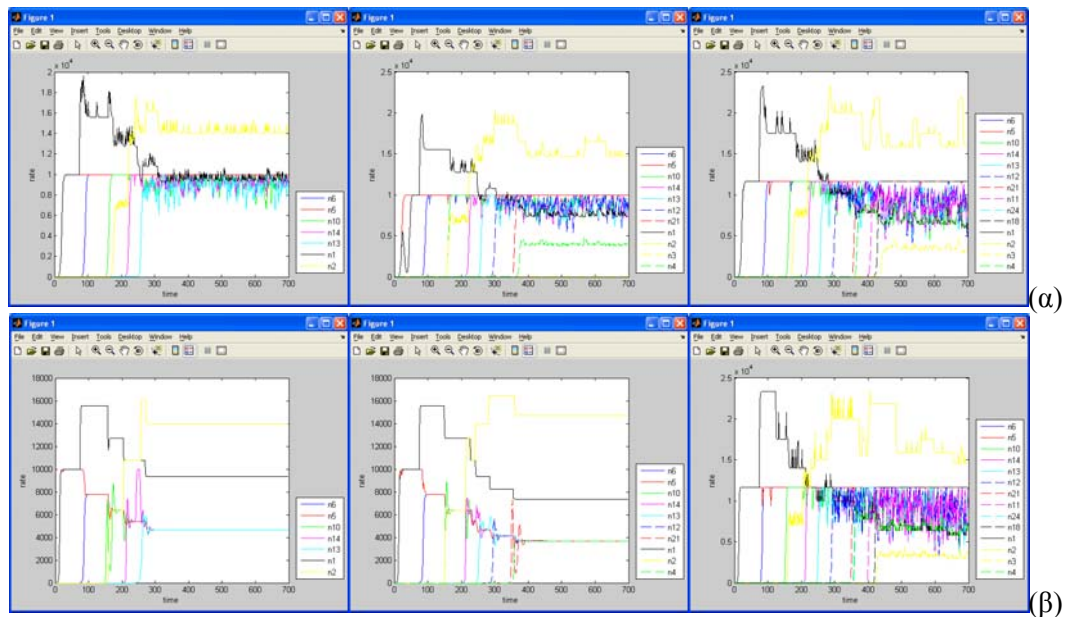
Στον Πίνακα 4-6 τρέξαμε όλα τα πειράματα με τις ίδιες παραμέτρους που επηρεάζουν τόσο τις συνθήκες σύγκλισης όσο και την τιμή σύγκλισης, δηλαδή τις παραμέτρους α , β , γ και K . Στα πειράματα αυτά αλλάξαμε την τιμή της παραμέτρου που επηρεάζει το χώρο προσωρινής μνήμης των κόμβων, δηλαδή την παράμετρο Maximum number of packets in IFQ που είδαμε στον Πίνακα 4-1. Στον Πίνακα 4-6 παρατηρούμε τα ίδια αποτελέσματα με τον Πίνακα 4-5 όσο αφορά την αλλαγή στην τιμή του Packet Delivery Ratio μεταξύ των πειραμάτων με διαφορετικό αριθμό ενεργών κόμβων. Όσο αφορά τη σύγκριση στην τιμή του Packet Delivery Ratio μεταξύ των αντίστοιχων πειραμάτων (για τον ίδιο αριθμό των ενεργών κόμβων) βλέπουμε ότι αυξάνεται εφόσον περισσότερα πακέτα παραλαμβάνονται από τον

κόμβο sink. Άρα λιγότερα πακέτα χάνονται λόγω υπερχείλισης της προσωρινής μνήμης των κόμβων.



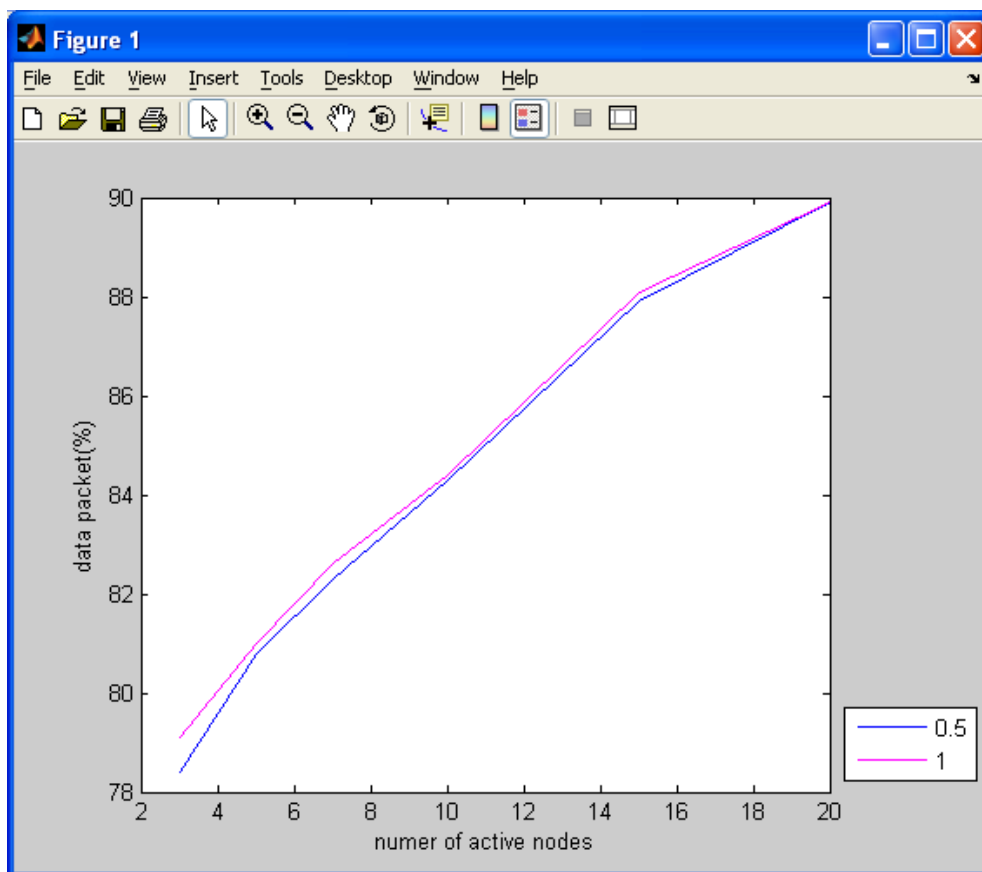
Σχήμα 4-9. (α) Υπολογισμός του Rate για τα Πειράματα με 3, 10 και 20 ενεργούς κόμβους (αντίστοιχα)
(β) Bandwidth για τα Πειράματα με 3, 10 και 20 ενεργούς κόμβους (αντίστοιχα)

Στο Σχήμα 4-9 βλέπουμε τις γραφικές παραστάσεις που αφορούν το Rate και το Bandwidth. Το Rate υπολογίζεται από το μηχανισμό αποφυγής συμφόρησης κάθε 1 δευτερόλεπτο για την αποστολή των πακέτων από κάθε κόμβο και το Bandwidth είναι ο πραγματικός ρυθμός με τον οποίο τα πακέτα δεδομένων παραλαμβάνονται από τον κόμβο 0 για τον κάθε κόμβο ξεχωριστά. Στις γραφικές παραστάσεις για το Bandwidth (Σχήμα 4-9 (β)), η τιμή του Bandwidth είναι λίγο χαμηλότερη από το υπολογιζόμενο Rate και αυτό οφείλεται στο χάσιμο κάποιων πακέτων λόγω υπερχείλισης της προσωρινής μνήμης αποθήκευσης. Επιπρόσθετα, στα πειράματα όπου οι ενεργοί κόμβοι είναι 10 και 20 το Bandwidth είναι πολύ χαμηλότερο. Αυτό συμβαίνει επειδή οι κόμβοι που αποστέλλουν δεδομένα είναι περισσότεροι οπότε η προσωρινή μνήμη των κόμβων υπερχειλίζει γρηγορότερα με αποτέλεσμα περισσότερα πακέτα να πετάγονται και τελικά, λιγότερα πακέτα να φτάνουν στον κόμβο 0.



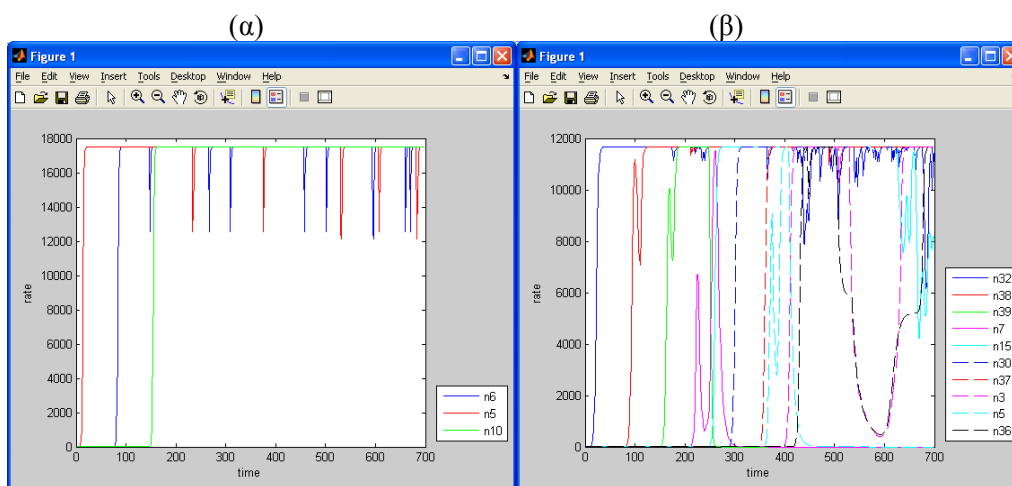
Σχήμα 4-10. (α) Rate για 5, 7 και 10 ενεργούς κόμβους και $r=0.5$
 (β) Rate για 5, 7 και 10 ενεργούς κόμβους και $r=1$

Στο Σχήμα 4-10 μπορούμε να δούμε πως η μεταβλητή r επηρεάζει το ρυθμό σύγκλισης των κόμβων όταν ο αριθμός των ενεργών κόμβων είναι 5, 7 και 10. Η μεταβλητή r επηρεάζει το πόσο γρήγορα το σύστημα φτάνει στο ρυθμό στον οποίο οι ενεργοί κόμβοι συγκλίνουν (σύμφωνα με την Εξίσωση 4 από το Κεφάλαιο 3). Στα πειράματα όπου ο αριθμός των ενεργών κόμβων 5 και 7 παρατηρήσαμε ότι όταν η μεταβλητή r ισούται με 0.5 οι κόμβοι δε συγκλίνουν. Υπάρχουν δηλαδή αυξομειώσεις στο ρυθμό ενώ στο πείραμα όπου ο αριθμός των ενεργών κόμβων είναι 10 υπάρχουν μεγαλύτερες αυξομειώσεις όταν η μεταβλητή r ισούται με 0.5. Στο Σχήμα 4-10(β), όπου η μεταβλητή r ισούται με 1 στα πειράματα όπου ο αριθμός των ενεργών κόμβων είναι 5 και 7 οι κόμβοι συγκλίνουν ακριβώς στο ρυθμό που υπολογίζεται από την Εξίσωση 4 που είδαμε στο Κεφάλαιο 3. Στο πείραμα όπου ο αριθμός των ενεργών κόμβων είναι 10 υπάρχουν αυξομειώσεις στο ρυθμό αλλά είναι λίγο σταθερότερος ο ρυθμός σε σύγκριση με το ρυθμό όταν η μεταβλητή r ισούται με 0.5.



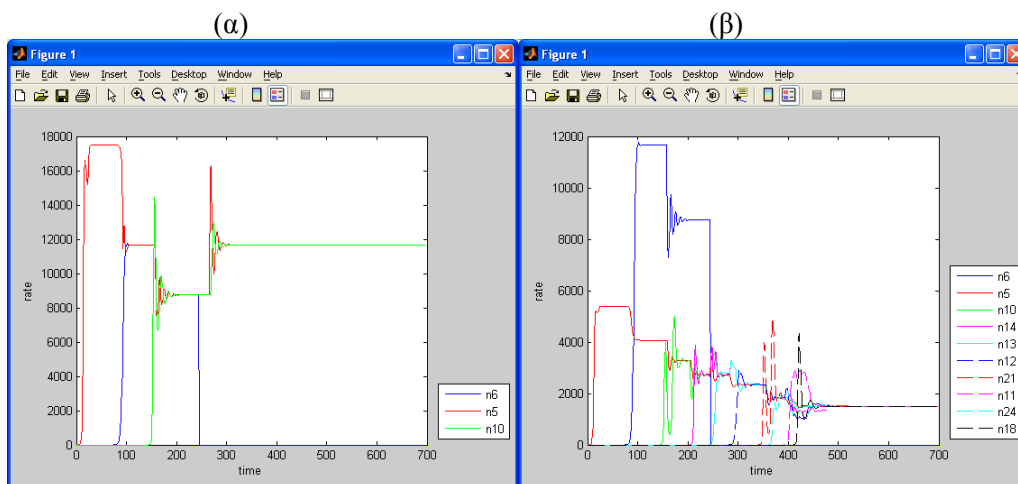
Σχήμα 4-11. Ποσοστό πακέτων δεδομένων για 3, 5, 7, 10, και 15 ενεργούς κόμβους

Στη γραφική του Σχήματος 4-11 παρατηρούμε το ποσοστό πακέτων δεδομένων που υπάρχουν στο δίκτυο σε σύγκριση με τα πακέτα που χρησιμοποιούνται για δρομολόγηση. Παρατηρούμε ότι όσο αυξάνεται ο αριθμός των ενεργών κόμβων αυξάνεται και το ποσοστό πακέτων δεδομένων που υπάρχουν στο δίκτυο. Τα πακέτα δρομολόγησης παραμένουν περίπου τα ίδια σε όλα τα πειράματα εφόσον και η τοπολογία που χρησιμοποιούμε είναι η ίδια ενώ τα πακέτα δεδομένων αυξάνονται αφού αυξάνεται και ο αριθμός των κόμβων που αποστέλλουν δεδομένα. Στη γραφική παράσταση του Σχήματος 4-11 παρατηρούμε επίσης ότι όταν η μεταβλητή γ ισούται με 1 τότε το ποσοστό πακέτων δεδομένων που στέλνονται είναι μεγαλύτερο σε σύγκριση με το ποσοστό δεδομένων που στέλνονται όταν η μεταβλητή γ ισούται με 0.5.



Σχήμα 4-12. (α) Rate για πείραμα με τυχαία τοπολογία με 3 ενεργούς κόμβους
(β) Rate για πείραμα με τυχαία τοπολογία με 10 ενεργούς κόμβους

Στο Σχήμα 4-12 παρατηρούμε τις γραφικές παραστάσεις για το Rate όταν ο αριθμός των ενεργών κόμβων είναι 3 (Σχήμα 4-12 (α)) και όταν ο αριθμός των ενεργών κόμβων είναι 10 (Σχήμα 4-12 (β)). Και στις δύο γραφικές παρατηρούμε ότι υπάρχουν αυξομειώσεις στο ρυθμό και αυτό είναι εντονότερο στη γραφική παράσταση όπου ο αριθμός των ενεργών κόμβων είναι 10. Αυτό συμβαίνει επειδή οι κόμβοι αλλάζουν πατέρα όποτε σε αυτή την περίπτωση πρέπει να υπολογίσουν ξανά το ρυθμό τους με βάση τα καινούρια δεδομένα του νέου τους πατέρα. Στο Σχήμα 4-12(β) οι κόμβοι που υπάρχουν στο δίκτυο είναι περισσότεροι, άρα περισσότερα είναι και τα πακέτα δεδομένων που υπάρχουν στο δίκτυο, οπότε περισσότερα πακέτα χάνονται. Αυτό έχει ως αποτέλεσμα να παρατηρούμε πιο συχνές αλλαγές στην τοπολογία, δηλαδή στην επιλογή του πατέρα και τελικά στη δομή του δέντρου.



Σχήμα 4-13. (α) Rate όπου ο κόμβος 6 σταματά την αποστολή δεδομένων με αρχικά 3 ενεργούς κόμβους
(β) Rate όπου ο κόμβος 6 σταματά την αποστολή δεδομένων με αρχικά 10 ενεργούς κόμβους

Στο Σχήμα 4-38 παρατηρούμε τις γραφικές παραστάσεις για το Rate όταν ο αριθμός των ενεργών κόμβων είναι αρχικά 3 (Σχήμα 4-13(α)) και όταν ο αριθμός των ενεργών κόμβων είναι αρχικά 10 (Σχήμα 4-13(β)). Και στις δύο γραφικές μπορούμε να παρατηρήσουμε ότι μόλις ο κόμβος 6 σταματά να στέλνει δεδομένα τότε ο ρυθμός αλλάζει για τους υπόλοιπους κόμβους του δικτύου που είναι ενεργοί. Δηλαδή, στο Σχήμα 4-13 (α) ο ρυθμός των κόμβων n5 και n10 αυξάνεται όταν ο κόμβος n6 σταματά να στέλνει δεδομένα αφού μειώνεται ο αριθμός των ενεργών κόμβων. Στο Σχήμα 4-13(β) ο ρυθμός των κόμβων n5, n10, n14, n12, n13, n21, n11, n24 και n18 αυξάνεται όταν ο κόμβος n6 σταματά να στέλνει δεδομένα, όπως και στο Σχήμα 4-13(α). Η αύξηση του ρυθμού επηρεάζεται από τον αριθμό των ενεργών κόμβων σύμφωνα πάντα με την Εξίσωση 4 που είδαμε στο Κεφάλαιο 4.

Αυτό που μπορούμε γενικά να παρατηρήσουμε για τις μεταβλητές του μηχανισμού για αποφυγή της συμφόρησης είναι ότι όταν ο αριθμός των ενεργών κόμβων αλλάζει τότε μεταβάλλονται και οι τιμές των παραμέτρων του (a, b, r) για τις οποίες παίρνουμε τις ψηλότερες τιμές για το Packet Delivery Ratio.

Κεφάλαιο 5

Συμπεράσματα και Μελλοντική Εργασία

Τα Ασύρματα Δίκτυα Αισθητήρων αποτελούνται από αρκετούς μικροσκοπικούς κόμβους αισθητήρων, οι οποίοι αναπτύσσονται σε μια περιοχή τυχαία κατανομημένοι. Σε προηγούμενο Κεφάλαιο αναφέραμε ότι οι κόμβοι αισθητήρων έχουν πολλούς περιορισμούς, όπως μειωμένη και πολλές φορές μη επαναφορτιζόμενη ενέργεια. Αυτό έχει σαν αποτέλεσμα να φτάνουμε στην ανάγκη για δημιουργία καινούριων πρωτοκόλλων τόσο για τη δρομολόγηση των πακέτων, όσο και για έλεγχο της συμφόρησης (σημαντικός παράγοντας για την κατανάλωση ενέργειας). Στη Διπλωματική εργασία αυτή προσπαθήσαμε να δημιουργήσουμε ένα εύρωστο και αξιόπιστο αλγόριθμο ελέγχου συμφόρησης για τα ασύρματα δίκτυα αισθητήρων που να βασίζεται στην αυτονομία και ανεξαρτησία των κόμβων αισθητήρων με αποδοτικό και αποτελεσματικό τρόπο. Ο μηχανισμός για αποφυγή της συμφόρησης που χρησιμοποιήσαμε είναι εμπνευσμένος από το μαθηματικό μοντέλο ανταγωνισμού Lotka-Volterra. Δηλαδή, εμπνευστήκαμε από τον τρόπο που τα είδη ενός οικοσυστήματος ανταγωνίζονται τους πόρους του, όπως την τροφή για να επιβιώσουν για να προσομοιώσουμε τον τρόπο με τον οποίο οι κόμβοι αισθητήρων ανταγωνίζονται για την προσωρινή μνήμη του πατέρα τους. Οι κόμβοι αισθητήρων σχεδιάστηκαν όλοι με τον ίδιο τρόπο χωρίς πολύπλοκους υπολογισμούς έτσι ώστε και πάλι να επαληθεύουν τους περιορισμούς των δικτύων αυτών.

Το ένα κομμάτι της Διπλωματικής αφορούσε τη δρομολόγηση των πακέτων, δηλαδή τη δημιουργία ενός δέντρου μέσω του οποίου οι κόμβοι αισθητήρων θα χρησιμοποιούν για να δρομολογούν τα πακέτα τους. Το άλλο κομμάτι αφορά το μηχανισμό για αποφυγή της συμφόρησης και είναι επηρεασμένος από τη βιολογία αφού είναι μια παραλλαγμένη μορφή του μαθηματικού μοντέλου ανταγωνισμού Lotka-Volterra. Έτσι, σχεδιάσαμε ένα αλγόριθμο για έλεγχο της ροής δεδομένων με αποφυγή της συμφόρησης σε κάθε κόμβο αισθητήρα. Υλοποιήσαμε ανάλυση του μοντέλου που αφορά τόσο τις παραμέτρους που επηρεάζουν τη δημιουργία του δέντρου, όσο και τις παραμέτρους του μαθηματικού μοντέλου για να μελετήσουμε τη σταθερότητα, σύγκλιση και δικαιοσύνη του αλγορίθμου μας. Τα πειράματα που υλοποιήσαμε επαλήθευσαν την ορθότητα των θεωρητικών αποτελεσμάτων και δείξαμε ότι το μοντέλο μας επιτυγχάνει προσαρμοστικότητα, δικαιοσύνη, κλιμακωσιμότητα και ικανοποιητική απόδοση.

Μέσα από τα διάφορα πειράματα που υλοποιήσαμε είδαμε ότι το σύστημα ενεργεί διαφορετικά όταν ο αριθμός των ενεργών κόμβων αλλάζει. Δηλαδή, κάτω από διαφορετικές τιμές των παραμέτρων που επηρεάζουν το μηχανισμό αποφυγής της συμφόρησης (a και b). Επίσης, είδαμε ότι το Packet Delivery Ratio μειώνεται όσο αυξάνεται ο αριθμός των ενεργών κόμβων για τον καλύτερο συνδυασμό τιμών των παραμέτρων σε κάθε πείραμα. Αυτό συμβαίνει επειδή όσο αυξάνεται ο αριθμός των ενεργών κόμβων στο δίκτυο αυξάνεται και ο αριθμός των δεδομένων που υπάρχουν στο δίκτυο με αποτέλεσμα να χάνονται περισσότερα πακέτα και άρα το Packet Delivery Ratio να μειώνεται. Ακόμα, όσο αυξάνεται ο αριθμός των ενεργών κόμβων αυξάνονται και οι τιμές των παραμέτρων a και β για να πιάνουμε τη ψηλότερη τιμή του Packet Delivery Ratio, όπως είδαμε και στο συγκεντρωτικό πίνακα στο προηγούμενο Κεφάλαιο (Πίνακα 4-5). Αυξάνοντας τις παραμέτρους μειώνεται η τιμή σύγκλισης (σύμφωνα με την Εξίσωση 4 στο Κεφάλαιο 3) και έτσι μπορούν οι κόμβοι να συγκλίνουν σε μια τιμή. Επίσης, μέσα από τα πειράματα μας είδαμε ότι οι κόμβοι συγκλίνουν σε ένα ρυθμό σε περιπτώσεις όπως όταν η τοπολογία των κόμβων μας είναι τυχαία και όταν κάποιος κόμβος σταματά την αποστολή δεδομένων. Τέλος, στα πειράματα μας είδαμε ότι η

κίνηση που υπάρχει στο δίκτυο δημιουργείται περισσότερο από τα πακέτα δεδομένων παρά από τα πακέτα δρομολόγησης.

Σαν μελλοντική εργασία θα μπορούσε να γίνει σύγκριση του δικού μας αλγορίθμου με κάποιο άλλο από αυτά που ήδη υπάρχουν όπως το ESRT με απαραίτητη προϋπόθεση και τα δύο να είναι δημιουργημένα στο ίδιο εργαλείο. Στο προηγούμενο Κεφάλαιο είδαμε ότι η καλή λειτουργία του αλγορίθμου μας εξαρτάται από τον αριθμό των ενεργών κόμβων. Δηλαδή, οι τιμές των παραμέτρων αλλάζουν ανάλογα με τον αριθμό των ενεργών κόμβων και αυτό φαίνεται και στο συγκεντρωτικό πίνακα που είδαμε στο προηγούμενο Κεφάλαιο. Έτσι θα μπορούσε να δημιουργηθεί ένας καινούριος αλγόριθμος, ο οποίος θα μπορεί να είναι προσαρμόσιμος ανάλογα με τον αριθμό των ενεργών κόμβων. Να προσαρμόζει με άλλα λόγια τις τιμές των παραμέτρων ανάλογα με τη κατάσταση του δικτύου. Τέλος, θα μπορούσε να τροποποιηθεί ο αλγόριθμος έτσι ώστε να λαμβάνει υπόψη την κινητικότητα των κόμβων. Δηλαδή, οι κόμβοι αισθητήρων θα μπορούν να είναι όχι μόνο ασύρματοι, αλλά και κινητοί.

Βιβλιογραφία

- [1] P. Antoniou and A. Pitsillides, "Congestion Control in Autonomous Decentralized Networks Based on the Lotka-Volterra Competition Model" Proceedings of the 19th International Conference on Artificial Neural Networks (ICANN'09), Limassol, Cyprus.
- [2] P. Antoniou and A. Pitsillides, "Towards a Scalable and Self-adaptable Congestion Control Approach for Autonomous Decentralized Networks," NiSIS, 2007.
- [3] M. Begon, J. L. Harper, and C. R. Townsend, "Ecology", Blackwell Science, 1996.
- [4] M. Begon, J. L. Harper, and C. R. Townsend, "Ecology: Individuals, Populations, and Communities," 3rd edition, Cambridge, 1996.
- [5] F. Brauer and C. Castillo-Chavez, "Mathematical Models in Population Biology and Epidemiology", New York, USA, 2000.
- [6] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, "Optimized Link State Routing Protocol," *Internet Draft, draft-ietf-manet-olsr-06.txt*, September 2001.
- [7] C. T. Ee and R. Bajcsy, "Congestion Control and Fairness for Many-to-One Routing in Sensor Networks," *Proceedings ACM Sensys '04, Baltimore, MD*, (2004) 3–5.
- [8] M. Gerla, X. Hong, L. Ma, and G. Pei. "Landmark Routing Protocol (LANMAR) for Large Scale Ad Hoc Networks," *Internet Draft, draft-ietf-manet-lanmar-04.txt*, June 2002.
- [9] M. Gerla, X. Hong, and G. Pei, "Fisheye State Routing Protocol (FSR) for Ad Hoc Networks," *Internet Draft, draft-ietf-manet-fsr-03.txt*, June 2002.
- [10] A. J. Goldsmith and S. B. Wicker, "Design challenges for energy constrained ad hoc wireless networks," *IEEE Wireless Communications* 9(4) (2002), 8–27.
- [11] N. J. Gotelli, "A Primer of Ecology", 2nd edition, Sunderland, 1998.

- [12] Z. J. Haas, M. R. Pearlman, and P. Samar, "The Interzone Routing Protocol (IERP) for Ad Hoc Networks," *Internet Draft, draft-ietf-manet-zone-ierp-02.txt*, July 2002.
- [13] Z. J. Haas, M. R. Pearlman, and P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks," *Internet Draft, draft-ietf-manet-zone-zrp-02.txt*, July 2002.
- [14] G. Hasegawa and M. Murata, "Self-adaptive and Scalable TCP Congestion Control based on Inline Network Measurement," *IEEE/ACM Transactions on Networkin*, 2006
- [15] D. B. Johnson, D. A. Maltz, Y-C Hu, and J. G. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR) ", *Internet Draft, draft-ietf-manet-dsr-07.txt*, February 2002.
- [16] H. Karl and A. Willig, "Protocols and Architectures for Wireless Sensor Networks," Wiley, 2005.
- [17] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross layer design," *IEEE Wireless Communications* 12(1) (2005), 3–11.
- [18] S. Keshav, "Real 5.0 overview", August 1997.
- [19] P. Levis, S. Madden, D. Gay, J. Polastre, R. Szewczyk, A. Woo, E. Brewer, and D. Culler, "The emergence of networking abstractions and techniques in TinyOS," *Proceedings of the 1st Symp. On Network Systems Design and Implementation*, 2004.
- [20] G. Malkin, "*The Official IETF working group Manet webpage - RIP Version 2*," <http://www.ietf.org/html.charters/>.
- [21] J. Moy, "OSPF Version 2," *Request For Comments (RFC) 2328*, April 1998.
- [22] NRL's Sensor Network Extension to NS-2, Networks and Communication Systems Branch, U.S. Naval Research Lab, <http://pf.itd.nrl.navy.mil/nrlsensorsim>.
- [23] R. G. Ogier, F. L. Templin, B. Bellur, and M. G. Lewis, "Topology Broadcast Based on Reverse - Path Forwarding (TBRPF)," *Internet Draft, draft-ietf-manet-tbrpf-05.txt*, March 2002.
- [24] C. Perkins, E. Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," *Internet Draft, draft-ietf-manet-aodv-11.txt*, June 2002.
- [25] J. Y. Teo, Y. Ha and C. K. Tham, "Interference-Minimized Multipath Routing with Congestion Control in Wireless Sensor Network for High-Rate Streaming," *IEEE Transactions on Mobile Computing*, 7(9)(2008), 1124-1137.
- [26] The network simulator NS2 – Mailing list, <http://mailman.isi.edu/mailman/listinfo/ns-users>
- [27] The NS Manual, <http://www.isi.edu/nsnam/ns/>.

[28] C.Y. Wan, A.T. Campbell, L. Krishnamurthy, “Pump-slowly, fetch-quickly (PSFQ): a reliable transport protocol for sensor networks,” *IEEE Journal of Selected Areas in Communication*, 23(4) (2005), 862–872

[29] C. Y .Wan, S. B. Eisenman, and A. T. Campbell, “CODA: Congestion Detection and Avoidance in Sensor Networks,” *Proceedings of ACM SenSys’03, Los Angeles , USA* , (2003), 5-7.

[30] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, “A survey of transport protocols for wireless sensor networks,” *IEEE Network* 20(3)(2006), 34-40.

[31] Wikipedia official website - Biological interaction,
http://en.wikipedia.org/wiki/Biological_interaction.

[32] Wikipedia official website – Competition,
[http://en.wikipedia.org/wiki/Competition_\(biology\)](http://en.wikipedia.org/wiki/Competition_(biology)).

APPENDIX A

Το NS-2 [27] είναι ένα νετερμινιστικό διακριτών γεγονότων εργαλείο προσομοίωσης δικτύων το οποίο ξεκίνησε από το Lawrence Berkeley National Laboratory (LBNL) διαμέσω του DARPA που χρηματοδοτείται από το ερευνητικό έργο του Virtual InterNetwork Testbed (VINT). Το ερευνητικό έργο VINT είναι μια συνεργασία μεταξύ του Information Sciences Institute (ISI) στο University of Southern California (USC), του Xerox's Palo Alto Research Center (Xerox PARC), του πανεπιστημίου University of California Berkeley (UCB) και του LBNL.

Το NS-2 δημιουργήθηκε το 1989. Σκοπός της δημιουργίας του ήταν να χρησιμοποιηθεί σαν εναλλακτικός προσομοιωτής δικτύου του REAL [18]. Από τότε το εύρος του και η χρησιμοποίηση του ερευνητικού έργου NS έχει αυξηθεί σημαντικά. Αν και υπάρχουν πολλοί διαφορετικοί προσομοιωτές δικτύων διαθέσιμοι σήμερα το NS-2 είναι αυτό που χρησιμοποιείται πιο πολύ. Διαφέρει από τα περισσότερα εργαλεία επειδή είναι ένα λογισμικό ανοικτού κώδικα, προμηθεύοντας τον κώδικα πηγής σε οποιοδήποτε τον χρειάζεται χωρίς κανένα κόστος. Η απελευθέρωση του κώδικα πηγής επιτρέπει στους χρήστες να δημιουργούν τις δικές τους συναρτήσεις και υπό-προγράμματα και επίσης, γίνεται πιο εύκολο για τους χρήστες να τα υλοποιούν στο περιβάλλον του NS-2. Ένα από τα κυριότερα πλεονεκτήματα που προσφέρει η απελευθέρωση του κώδικα πηγής στην ομάδα του ερευνητικού έργου του NS είναι ότι ανεξάρτητοι ερευνητές βοηθούν στην ανάπτυξη του NS-2. Επίσης, κάποιος ερευνητής συμβάλλει υλοποιώντας τον κώδικα ενός μη υλοποιημένου πρωτοκόλλου ή

αλγόριθμοι για τις σπουδές του προσθέτοντας στον συνολικών αριθμό των υλοποιημένων πρωτοκόλλων το δικό του αλγόριθμο .

Αυτό που είναι σημαντικό να τονίσουμε για το NS-2 είναι ότι αποτελεί ερευνητικό εργαλείο και όχι κάποιο εμπορικό λογισμικό. Μια κύρια διαφορά είναι ότι υπάρχει ένας πολύ μικρός αριθμός ατόμων στην ερευνητική ομάδα του NS-2 για να υποστηρίξουν όλους τους χρήστες σε σύγκριση με κάποιο κανονικό λογισμικό. Οι χρήστες μπορούν να βρίσκουν λύσεις για διάφορα προβλήματα που πιθανό να αντιμετωπίσουν κατά την υλοποίηση τους στα mailing list [26]. Βασικά, στα mailing list μπορεί κάποιος χρήστης να βρει προβλήματα και λύσεις που αντιμετώπισαν και άλλοι. Να προσθέσουμε ότι τα πιο κοινά πρωτόκολλα έχουν χρησιμοποιηθεί και ελεγχθεί πολλές φορές άρα δεν υπάρχει ανησυχία για την ορθότητα των συγκεκριμένων πρωτοκόλλων. Οι νέες όμως υλοποιήσεις συνήθως ξεκινούν σαν ερευνητικές εργασίες ανεξάρτητες της ομάδας του NS-2 και λόγω κάποιων προβλημάτων που αντιμετωπίζουν οι προγραμματιστές ενθαρρύνουν τη βοήθεια από οποιοδήποτε.

Δομή του NS-2

Το NS-2 αποτελείται από εκατοντάδες ξεχωριστά μικρότερα προγράμματα, τα οποία βοηθούν το χρήστη να βρίσκει οτιδήποτε ψάχνει. Το κάθε ξεχωριστό πρωτόκολλο, όπως και οι παραλλαγές του, έχουν τα δικά τους αρχεία. Αν στο NS-2 υπήρχαν μόνο μερικά γιγάντια αρχεία τότε θα ήταν αδύνατο κάποιος να καταφέρει να κατανοήσει τον κώδικα.

- C++

Η C++ είναι η κυρίαρχη γλώσσα προγραμματισμού του NS-2. Είναι η γλώσσα προγραμματισμού που χρησιμοποιείται από όλα τα μικρότερα προγράμματα που αποτελούν την ιεραρχία του NS-2.

Η C++ είναι μια από τις πιο κοινές γλώσσες προγραμματισμού και σχεδιάστηκε για αντικειμενοστραφή προγραμματισμό, κάνοντας την μια λογική επιλογή, για να χρησιμοποιείται στο NS-2. Αυτό βοηθά είτε όταν ο χρήστης θέλει να κατανοήσει

κάποιο κώδικα είτε όταν θέλει να κάνει κάποιες αλλαγές στον κώδικα. Υπάρχουν πολλά βιβλία για τη C++ και εκατοντάδες αν όχι χιλιάδες σελίδες στο Διαδίκτυο απλουστεύοντας τη διαδικασία αναζήτησης βοήθειας ή απαντήσεων όσο αφορά κώδικα στο NS-2.

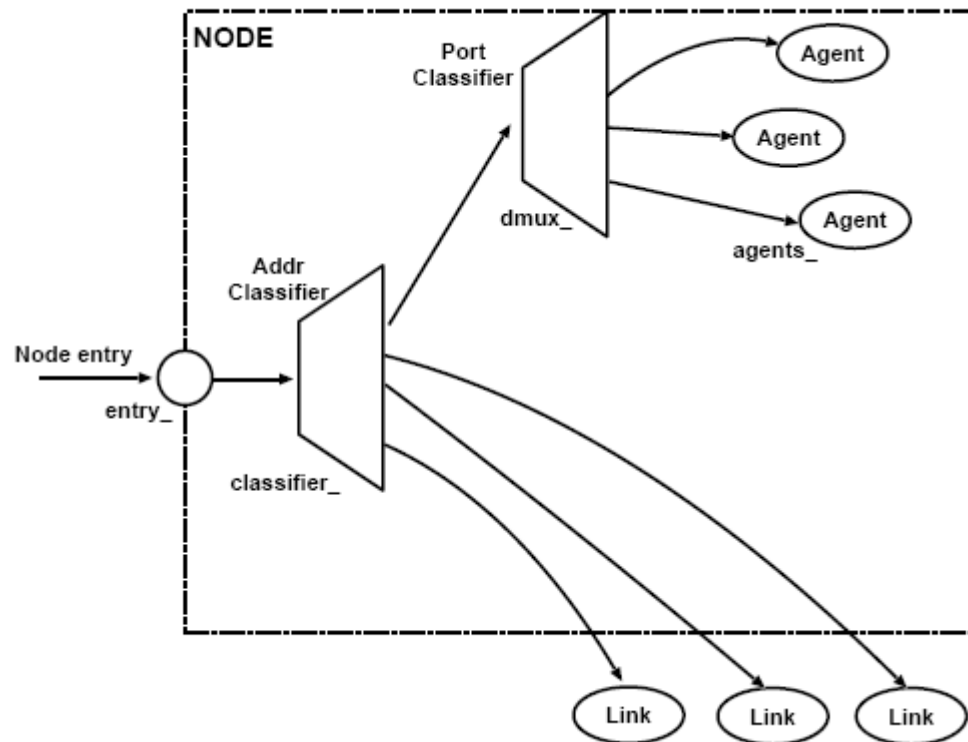
- OTCL

Η Object Tcl (OTcl) είναι μια αντικειμενοστραφή έκδοση της γλώσσας προγραμματισμού Tool Command Language (Tcl), η οποία οδηγείται από εντολές και σύνταξη. Αυτή είναι η δεύτερη από τις δύο γλώσσες προγραμματισμού που χρησιμοποιούνται στο NS-2. Η OTcl χρησιμοποιείται σαν η μπροστινή διασύνδεση του NS-2. Επίσης, γίνεται και διασύνδεση της γλώσσας τύπου script Tcl με τη C++. Μαζί οι δύο γλώσσες προγραμματισμού δημιουργούν ένα περιβάλλον script ελεγχόμενο από τη C++. Αυτό βοηθά κατά τη δημιουργία προσομοιώσεων αφού απλά γράφεται ένα script, το οποίο θα εκτελεστεί όταν τρέξει η προσομοίωση. Αυτά τα script είναι η συνταγή της προσομοίωσης και πρέπει να καθορίζονται οι προδιαγραφές για την ίδια την προσομοίωση. Αν το script δεν καθοριστεί όπως πρέπει, δηλαδή, να καθοριστούν η τοπολογία του δικτύου, οι ροές δεδομένων τότε δεν θα γίνει τίποτα.

Κόμβοι

Ένας κόμβος στο NS-2 είναι ακριβώς όπως ένας κόμβος στο δίκτυο. Ένας κόμβος μπορεί να είναι είτε ο τελευταίος κόμβος είτε ένας ενδιάμεσος κόμβος στο δίκτυο. Όλοι οι πράκτορες (agents), όπως θα δούμε και στην επόμενη παράγραφο, και τα καλώδια πρέπει να είναι συνδεδεμένα σε ένα κόμβο για να μπορούν να δουλεύουν. Επίσης, υπάρχουν διάφοροι τύποι κόμβων ανάλογα με το είδος του δικτύου στο οποίο θα προσομοιωθούν. Οι κυριότεροι τύποι είναι οι απλοί κόμβοι και κινητοί κόμβοι. Οι πρώτοι χρησιμοποιούνται κυρίως για

ενσύρματα δίκτυα ενώ οι δεύτεροι για ασύρματα δίκτυα. Στο Σχήμα Appendix A -1 μπορούμε να δούμε τη γενική δομή ενός κόμβου unicast (όπως τον κόμβο που χρησιμοποιούμε στην υλοποίηση



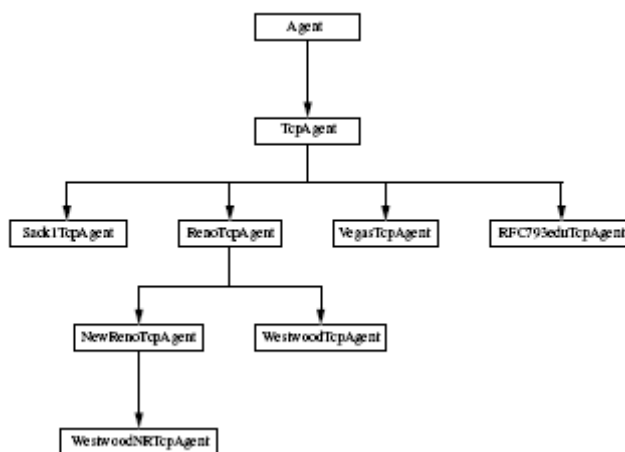
Σχήμα Appendix A-1. Δομή του Unicast Κόμβου

μας) στο NS-2. Ο κόμβος από μόνος του είναι μια κλάση του OTcl και γι'αυτό τα περισσότερα από τα συστατικά του είναι από μόνα τους TclObjects. Η απλή δομή του κόμβου που βλέπουμε στο Σχήμα Appendix A-1 αποτελείται από δύο TclObjects: το address classifier (classifier_) και το port classifier (dmux_). Η λειτουργικότητα και των δύο αυτών classifiers είναι να κατανέμουν τα εισερχόμενα πακέτα στο σωστό πράκτορα ή στο σωστό εξερχόμενο καλώδιο. Όλοι οι κόμβοι περιέχουν τα εξής συστατικά:

- Διεύθυνση ή id_, τα οποία αυξάνονται κατά 1 αρχίζοντας από το 0 όπως δημιουργούνται οι κόμβοι στην προσομοίωση.
- Λίστα με τους γείτονες (neighbor_).
- Λίστα με τους πράκτορες (agent_).

- Ένα αναγνωριστή κόμβου (nodetype_).
- Και τέλος, μια μονάδα δρομολόγησης.

Στο NS-2 υπάρχουν πολλές διαφορετικές εντολές με τις οποίες καθορίζονται τα πρωτόκολλα που χρησιμοποιούνται. Για παράδειγμα, το είδος του πρωτοκόλλου δρομολόγησης που θα χρησιμοποιηθεί ή αν υπάρχει κάποια ιδιαιτερότητα στην επιλογή του μονοπατιού – όχι το πιο κοντινό μονοπάτι. Περισσότερες λεπτομέρειες για το κόμβους μπορούν να βρεθούν στο [27].



Σχήμα Appendix A-2. Ιεραρχία των Πράκτορων με βάση το Πρωτόκολλο TCP στο NS-2

Πράκτορες

Οι πράκτορες είναι ένα συγκεντρωτικό όνομα για τα περισσότερα πρωτόκολλα που μπορείς να βρεις στο επίπεδο Μεταφοράς. Στο [27] καθορίζονται σαν τερματικά όπου τα πακέτα δημιουργούνται και καταναλώνονται. Οι πράκτορες στο NS-2 είναι όλοι συνδεδεμένοι στην κλάση πατέρας τους, η οποία ονομάζεται “agent”. Σε αυτή την κλάση καθορίζεται η γενικότερη συμπεριφορά και οι κλάσεις απόγονοι βασίζονται σε κάποιες αλλαγές στις κληρονομημένες συναρτήσεις της κλάσης πατέρας. Οι αλλαγμένες συναρτήσεις θα αντικαταστήσουν τις παλιές και με αυτό τον τρόπο γίνεται αλλαγή στην απόδοση για να

προσομοιωθεί το επιθυμητό πρωτόκολλο. Στη συνέχεια, θα δούμε ένα παράδειγμα από ένα γνωστό πρωτόκολλο όπως το TCP Tahoe. Ο πράκτορας του TCP Tahoe είναι ο πατέρας όλων των υλοποιήσεων του TCP. Το TCP Tahoe κληρονομεί τις βασικές συναρτήσεις από την κλάση πατέρα την “agent” και προσθέτει κάποιες απαραίτητες συναρτήσεις για να επιτευχθεί η συμπεριφορά του TCP. Ο πράκτορας του TCP Reno κληρονομεί τη συμπεριφορά από το TCP Tahoe και αντικαθιστά τις συναρτήσεις που χρειάζεται να αλλαχθούν για να προσομοιώσει τη συμπεριφορά του TCP Reno παρά να χρειαστεί να γράψει από την αρχή καινούργιο πρόγραμμα. Στο NS-2 υπάρχουν υλοποιημένα πολλά πρωτόκολλα παραλλαγές του TCP και στο Σχήμα Appendix A-2 μπορούμε να δούμε την ιεραρχία των πρωτοκόλλων που υπάρχουν στο NS-2.