

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ  
Τμήμα Πληροφορικής  
Μεταπτυχιακό Πρόγραμμα Σπουδών  
«Προηγμένα Συστήματα Πληροφορικής»



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Μελέτη φόρτισης δικτύων αισθητήρων με χρήση  
ασύρματων κινούμενων φορτιστών**

**Research about charging of Wireless Sensor Networks  
using wireless moving chargers**

*Παπαδοκαλάκης Γεώργιος  
Α.Μ.: ΜΠΣΠ14071*

Επιβλέπων: Δημήτριος Βέργαδος  
Άγγελος Μιχάλας

Πειραιάς, Οκτώβριος 2017





## Περίληψη

Ένα ασύρματο δίκτυο αισθητήρων (ΑΔΑ / Wireless Sensor Network - WSN ) αποτελείται από διασκορπισμένους αυτόνομους αισθητήρες για την παρακολούθηση φυσικών ή περιβαλλοντολογικών συνθηκών, όπως η θερμοκρασία, ο ήχος, η ατμοσφαιρική πίεση κτλ. και μέσω συνεργασίας να μεταφέρει τα δεδομένα μέσω του δικτύου σε μια συγκεκριμένη τοποθεσία.

Κατά το σχεδιασμό ενός ασύρματου δικτύου αισθητήρων, υπάρχουν πολλοί σημαντικοί παράγοντες, οι οποίοι θα πρέπει να ληφθούν υπ' όψιν. Μερικοί από αυτούς, είναι το περιβάλλον στο οποίο θα τοποθετηθεί το δίκτυο, οι περιορισμοί στο υλικό, η ανοχή σε σφάλματα και το κόστος παραγωγής. Έναν από τους σημαντικότερους παράγοντες, αποτελεί η διαχείριση των ενεργειακών πόρων των κόμβων των αισθητήρων, οι οποίοι απαρτίζουν αυτά τα δίκτυα.

Η τεχνολογία ασύρματης φόρτισης, είναι μία πολλά υποσχόμενη τεχνολογία. Οι σύγχρονες εξελίξεις στον τομέα αυτό, καθώς και οι εξελίξεις σχετικά με τα υλικά των μπαταριών που μπορούν να χρησιμοποιηθούν, ανοίγουν το δρόμο για νέες ευκαιρίες, οι οποίες αφορούν τον τρόπο διαχείρισης της ενέργειας στα δίκτυα αισθητήρων.

Η παρούσα πτυχιακή έχει ως αντικείμενο τον σχεδιασμό και την υλοποίηση ενός Αλγορίθμου Ενεργειακής Φόρτισης Σε Δίκτυο Αισθητήρων, ο οποίος θα λαμβάνει υπόψιν του όλα τα παραπάνω. Στο πρώτο μέρος, γίνεται μια παρουσίαση των δικτύων αισθητήρων καθώς επίσης περιγράφονται τα μέρη που το απαρτίζουν, οι παράγοντες που το επηρεάζουν και οι εφαρμογές για τις οποίες μπορεί να χρησιμοποιηθεί. Στη συνέχεια, γίνεται μια ανάλυση του προβλήματος της κατανάλωσης ενέργειας, το οποίο παρατηρείται σε τέτοιου είδους δίκτυα και παρουσιάζονται διάφορες προσεγγίσεις που έχουν προηγηθεί. Παράλληλα, γίνεται αναφορά σε θέματα δρομολόγησης και σε πρωτόκολλα που αφορούν τα δίκτυα αισθητήρων, καθώς αποτελούν βασικό συστατικό τους.

Στο τελευταίο μέρος της παρούσας διπλωματικής, γίνεται μια προσπάθεια υλοποίησης αλγορίθμου για την φόρτιση των κόμβων δικτύου αισθητήρων. Στην προσπάθεια αυτή, λαμβάνονται υπ' όψιν και συνδυάζονται δυο αλγόριθμοι φόρτισης από παλιότερες εργασίες. Ο ένας έχει να κάνει με τη φόρτιση των κόμβων από κινούμενους φορτιστές, ενώ ο δεύτερος με τη φόρτιση των φορτιστών από άλλους μεγαλύτερους. Η υλοποίηση γίνεται σε περιβάλλον Matlab και αφού παρουσιαστούν τα αποτελέσματα μέσω γραφικών παραστάσεων, παρατίθενται συμπεράσματα και προτάσεις πάνω στις οποίες θα μπορούσε να βασιστεί και να συνεχιστεί η μελέτη.



## Abstract

Wireless sensor networks (WSN), sometimes called wireless sensor and actuator networks (WSAN), are spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure etc and to cooperatively pass their data through the network to other locations. The more modern networks are bi-directional, also enabling control of sensor activity.

The design of a wireless sensor network requires many important factors that need to be taken into account. Some of them, are the environment in which the network will be placed, hardware constraints, fault tolerance, and production costs. One of the most important factors, is the management of the sensor node energy resources, which classify these networks.

Wireless charging technology is a very promising technology. Modern developments in this area, as well as advances in battery materials that can be used, pave the way for new opportunities in how to manage energy in network sensors.

This diploma thesis aims at design and implementation of a Load Energy Algorithm in a WSN, which will take into account all of the above. In the first part, we present the WSNs as well as the parts of them, the conditions that effect their function and the applications that they can be used for. Subsequently, we analyze the problem of the loss of energy, that is very important in WSNs and we present researches of the past. At the same time, we are referring to routing and other protocols about WSNs, because of their important role.

In the final part, we are trying to implement an algorithm for the charging of the nodes in WSNs. In our effort, we combine two algorithms of charging from two older papers. The first one is about charging the nodes by the chargers while the second one is about the charging of the chargers by bigger chargers. The implementation is taking place in Matlab and after the presentation of the results in graphs, conclusions and proposals are made to help the continuation of the research about charging in WSNs.



## Ευχαριστίες

Για την υλοποίηση της παρούσας διπλωματικής, θα ήθελα να ευχαριστήσω για τη συμβολή τους, τους ακόλουθους:

Τον κ. Άγγελο Μιχάλα, καθηγητή στο Τεχνολογικό Εκπαιδευτικό Ίδρυμα (ΤΕΙ) Δυτικής Μακεδονίας, για τη διαρκή επικοινωνία που είχαμε όλον τον καιρό υλοποίησης της εργασίας. Η συμβολή του ήταν καθοριστική στη συγγραφή και την ανάπτυξη της με συχνές συναντήσεις και συζητήσεις.

Τον κ. Δημήτριο Βέργαδο, αναπληρωτή πρόεδρο του τμήματος Πληροφορικής και πρόεδρο του Μεταπτυχιακού Προγράμματος Σπουδών «Προηγμένα Συστήματα Πληροφορικής» του Πανεπιστημίου Πειραιά, ο οποίος βοήθησε με εποικοδομητικά σχόλια στην ανάπτυξη της παρούσας διπλωματικής εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου και τους φίλους μου, οι οποίοι με στήριξαν ψυχολογικά όλον αυτόν τον καιρό.

22.10.2017

Παπαδοκαλάκης Γεώργιος



## Περιεχόμενα

Πίνακας Εικόνων .....	8
Εισαγωγή.....	9
Κεφάλαιο 1 <sup>ο</sup> : Ασύρματα δίκτυα αισθητήρων.....	10
1.1 Ορισμός Ασύρματων Δικτύων Αισθητήρων - Αρχιτεκτονική.....	10
1.2 Κόμβοι αισθητήρων .....	12
1.3 Δικτύων Αισθητήρων και Ad hoc Δίκτυα .....	15
1.4 Παράγοντες που επηρεάζουν τη σχεδίαση των δικτύων αισθητήρων .....	16
1.5 Εφαρμογές στα Ασύρματα Δίκτυα Αισθητήρων.....	20
Κεφάλαιο 2 <sup>ο</sup> : Ενεργειακή κατανάλωση στα WSN.....	23
2.1 Το πρόβλημα της κατανάλωσης ενέργειας στα WSN.....	23
2.2 Επαναφορτιζόμενα Δίκτυα Αισθητήρων .....	25
2.3 Ασύρματη Μεταφορά Ενέργειας .....	26
2.4 Ασύρματη επαναφόρτιση δικτύων αισθητήρων .....	28
2.5 Βιβλιογραφική Αναφορά .....	30
Κεφάλαιο 3 <sup>ο</sup> : Δρομολόγηση και Ενέργεια .....	32
3.1 Τι είναι δρομολόγηση .....	32
3.2 Πρωτόκολλο Δικτύων.....	33
3.3 Αλγόριθμοι Δρομολόγησης.....	33
3.4 Αλγόριθμος Εύρεσης Συντομότερων Διαδρομών Dijkstra's algorithm.....	36
3.5 Αλγόριθμος Εύρεσης Συντομότερης Διαδρομής Belman-Ford ή Ford-Fulkerson Algorithm ...	37
3.6 Στατική δρομολόγηση .....	38
3.7 Δυναμική Δρομολόγηση .....	39
3.8 Δρομολόγηση πλημμύρας (flooding).....	40
3.9 Δρομολόγηση διανυσμάτων απόστασης (Distance vector) .....	40
3.10 Δρομολόγηση κατάστασης συνδέσεων (Link State).....	41
Κεφάλαιο 4 <sup>ο</sup> : Δρομολόγηση σε δίκτυα Αισθητήρων .....	42
4.1 Cluster Based Routing Protocol.....	42
4.2 Global State Routing Πρωτόκολλο (GSR) .....	44



---

4.3 Fisheye State Routing protocol .....	45
4.4 Zone Routing Protocol.....	46
Κεφάλαιο 5 <sup>ο</sup> : Αλγόριθμος Ενεργειακής Φόρτισης Σε Δίκτυο Αισθητήρων .....	49
5.1 Το μοντέλο Φόρτισης.....	49
5.2 Αποτελέσματα προηγούμενων μελετών .....	49
5.3 Αποτελέσματα Εφαρμογής και Προσομοίωσης .....	69
Κεφάλαιο 6 <sup>ο</sup> : Συμπεράσματα.....	80
Βιβλιογραφία .....	85
Παράρτημα – Κώδικας Εφαρμογής σε Matlab .....	88



## Πίνακας Εικόνων

Εικόνα 1. Αρχιτεκτονική WSN .....	11
Εικόνα 2. Αρχιτεκτονική Ασύρματου Αισθητήρα .....	12
Εικόνα 3. Ακρίβεια πληροφορίας σε ένα πρωτόκολλο FSR .....	46
Εικόνα 4. Ένα παράδειγμα μιας ζώνης των δύο hops .....	46
Εικόνα 5. Παράδειγμα λειτουργίας του zone routing protocol .....	48
Εικόνα 6. Απεικόνιση Δικτύου.....	52
Εικόνα 7. Αποτελέσματα εφαρμογής του αλγορίθμου (Α) [4] .....	57
Εικόνα 8. Δίκτυο χωρισμένο σε τμήματα με ετερογενείς πυκνότητες [5] .....	59
Εικόνα 9. Ενεργοί κόμβοι με το πέρασμα του χρόνου [5] .....	63
Εικόνα 10: Routing robustness με το πέρασμα του χρόνου .....	63
Εικόνα 11. Απεικόνιση multihop δικτύου (Α) .....	72
Εικόνα 12. Απεικόνιση multihop δικτύου (Β).....	73
Εικόνα 13. Απεικόνιση multihop δικτύου (Γ).....	74
Εικόνα 14. Αποτελέσματα – Σύνολο ενέργειας .....	75
Εικόνα 15. Αποτελέσματα – Αριθμός ενεργών κόμβων .....	76
Εικόνα 16. Αποτελέσματα – Περίπτωση απλού δικτύου .....	77
Εικόνα 17. Αποτελέσματα – Περίπτωση Δικτύου με πρωτόκολλο DCLK .....	78
Εικόνα 18. Αποτελέσματα – Περίπτωση Δικτύου με πρωτόκολλο 1KDC υπερφορτιστών και με πρωτόκολλο φορτιστών DCLK .....	79





## Εισαγωγή

Η παρούσα διπλωματική εργασία, ασχολείται με τα δίκτυα αισθητήρων και πιο συγκεκριμένα, με την ενεργειακή τους φόρτιση. Προσπαθεί να προτείνει μια αξιόλογη λύση στο ενεργειακό πρόβλημα που αντιμετωπίζουν αυτού του τύπου τα δίκτυα. Στην προσπάθεια αυτή, γίνεται χρήση δυο μοντέλων φόρτισης, το ένα για τη φόρτιση των κόμβων και το άλλο για την φόρτιση των φορτιστών, τα οποία προτείνονται από δυο διαφορετικές εργασίες.

Αρχικά, γίνεται παρουσίαση των δικτύων αισθητήρων. Στο πρώτο κεφάλαιο, γίνεται επίσης ανάλυση των κόμβων αισθητήρων, και παρουσίαση των Ad hoc δικτύων, τα οποία θα χρησιμοποιηθούν στην υλοποίηση. Επιπλέον, σε αυτό το κεφάλαιο, γίνεται αναφορά στους παράγοντες, οι οποίοι θα πρέπει να λαμβάνονται υπ' όψιν κατά την υλοποίηση ενός δικτύου αισθητήρων, καθώς και εφαρμογές, από διάφορους τομείς, στις οποίες μπορεί να γίνει χρήση των δικτύων αυτών.

Στη συνέχεια, στο δεύτερο κεφάλαιο, γίνεται αναφορά στο θέμα που πραγματεύεται η συγκεκριμένη διπλωματική και δεν είναι άλλο από το πρόβλημα της ενεργειακής καταναλώσης. Αφού γίνει η παρουσίαση του, αναλύεται το θέμα της φόρτισης των αισθητήρων. Ειδικότερα, η ανάλυση αυτή, αφορά την ασύρματη φόρτιση των αισθητήρων, την οποία θα χρησιμοποιήσουμε και αργότερα στην υλοποίηση. Ακόμα, γίνεται βιβλιογραφική αναφορά σε διάφορες προγενέστερες εργασίες, σχετικές με το συγκεκριμένο θέμα.

Το τρίτο κεφάλαιο της εργασίας, αναφέρεται σε ένα σημαντικό θέμα των δικτύων υπολογιστών, το οποίο δε θα μπορούσε να μην αφορά και τα δίκτυα αισθητήρων, και δεν είναι άλλο από αυτό της δρομολόγησης. Εξηγούμε τι σημαίνει η έννοια της δρομολόγησης, παρουσιάζουμε τις διάφορες μορφές της, καθώς και σημαντικούς αλγορίθμους, οι οποίοι χρησιμοποιούνται κατά κόρον σε κάθε μορφή δικτύου.

Αφού παρουσιάσουμε το θέμα της δρομολόγησης, στο τέταρτο κεφάλαιο, επικεντρωνόμαστε στη δρομολόγηση στα δίκτυα αισθητήρων, και παρουσιάζουμε



διάφορα πρωτόκολλα, τα οποία βασίζονται στην ομαδοποίηση των αισθητήρων και κρίνονται κατάλληλα για τη συγκεκριμένη μορφή δικτύου.

Τέλος, στα δύο τελευταία κεφάλαια, παρουσιάζεται το μοντέλο φόρτισης το οποίο προτείνουμε, τα αποτελέσματα προηγούμενων εργασιών, σχετικές με το θέμα, αλλά και τα αποτελέσματα της συγκεκριμένης υλοποίησης. Επιπλέον, αναλύουμε τα συμπεράσματα μας και προτείνουμε ιδέες σχετικά με το πως θα πρέπει να συνεχιστεί η έρευνα πάνω στο συγκεκριμένο θέμα.

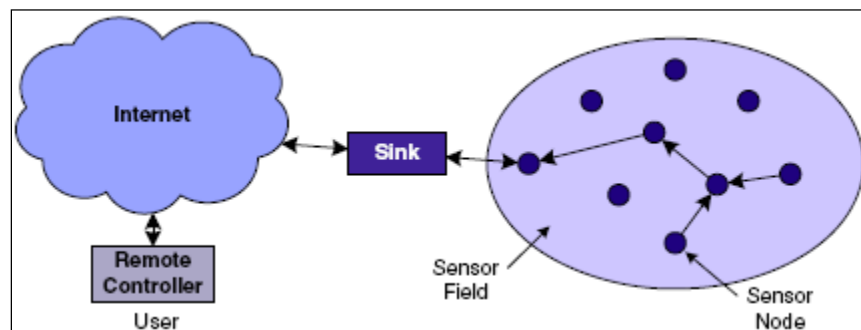
## ***Κεφάλαιο 1<sup>ο</sup>: Ασύρματα δίκτυα αισθητήρων***

Στο κεφάλαιο αυτό, εισάγουμε τον ορισμό των Ασύρματων Δικτύων Αισθητήρων, καθώς αναλύουμε και την αρχιτεκτονική τους. Περιγράφονται τα Δίκτυα Αισθητήρων και συγκεκριμένα τα Ad hoc Δίκτυα. Επίσης, αναφέρονται και αναλύονται οι παράγοντες, οι οποίοι επηρεάζουν τη σχεδίαση τους, ενώ γίνεται αναφορά στους τομείς, τους οποίους μπορούν να χρησιμοποιηθούν και για ποιό σκοπό.

### **1.1 Ορισμός Ασύρματων Δικτύων Αισθητήρων - Αρχιτεκτονική**

Η πρόοδος στη τεχνολογία των μικροηλεκτρομηχανικών συστημάτων, στις ασύρματες επικοινωνίες, τη ψηφιακή και αναλογική ηλεκτρονική, έχει επιτρέψει την ανάπτυξη πολυ-λειτουργικών κόμβων-αισθητήρων, χαμηλού κόστους και ισχύος. Οι αισθητήρες αυτοί, είναι πολύ μικροί σε μέγεθος και μπορούν να επικοινωνούν μεταξύ τους σε κοντινές αποστάσεις. Αποτελούνται από επιμέρους τμήματα αισθητήρων, επεξεργασίας δεδομένων και επικοινωνίας, και δημιούργησαν την ιδέα για ασύρματα δίκτυα αισθητήρων βασισμένα σε συνεργασία ενός μεγάλου αριθμού τέτοιων κόμβων. Τα ασύρματα δίκτυα αισθητήρων, αποτελούν μια σημαντική πρόοδο σε σχέση με τους παραδοσιακούς αισθητήρες και μας επιτρέπουν να έχουμε καλύτερη απεικόνιση του φαινομένου, το οποίο παρακολουθείται καθώς και μεγαλύτερη περιοχή κάλυψης [2].

Τα Ασύρματα Δίκτυα Αισθητήρων (WSNs), αποτελούνται από έναν ή περισσότερους σταθμούς βάσης (sink ή base station) και από μερικές δεκάδες ή χιλιάδες κόμβους-αισθητήρες (sensor nodes), οι οποίοι αναπτύσσονται σε ένα χώρο είτε μέσα στο φαινόμενο είτε πολύ κοντά σε αυτό [22]. Η αρχιτεκτονική ενός ασύρματου δικτύου αισθητήρων με ένα sink φαίνεται στην παρακάτω εικόνα.



Εικόνα 1. Αρχιτεκτονική WSN

Οι κόμβοι-αισθητήρες, συλλέγουν πληροφορίες από το περιβάλλον και τις μεταδίδουν στο sink μέσω του δικτύου, το οποίο σχηματίζουν οι ίδιοι. Στη συνέχεια, από το sink, μέσω του ίντερνετ, καταλήγουν στον απομακρυσμένο χρήστη που θα τις επεξεργαστεί. Η θέση των κόμβων-αισθητήρων δεν χρειάζεται να σχεδιαστεί ή να προκαθοριστεί.

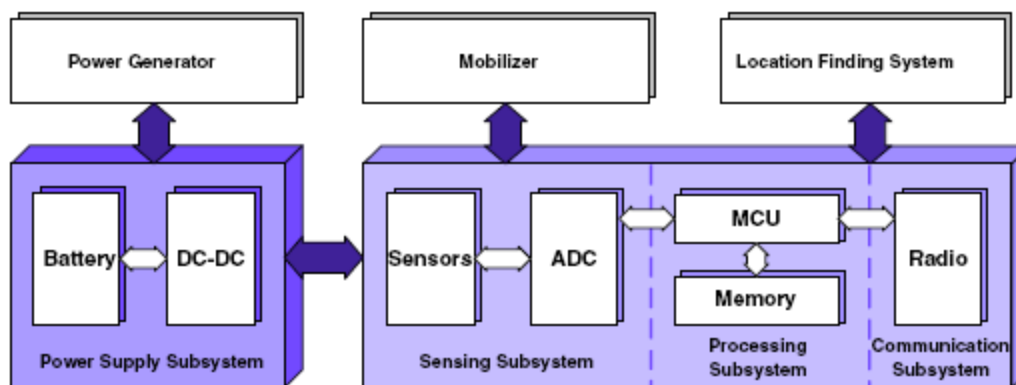
Αυτό επιτρέπει την τυχαία εναπόθεση των κόμβων σε περιοχές μη προσβάσιμες από τον άνθρωπο ή σε επιχειρήσεις αντιμετώπισης καταστροφών. Από την άλλη πλευρά, αυτό σημαίνει ότι τα πρωτόκολλα δικτύου και οι αλγόριθμοι των κόμβων, θα πρέπει να έχουν την ικανότητα να οργανώνονται από μόνα τους.

Ένα άλλο χαρακτηριστικό γνώρισμα των ασύρματων δικτύων αισθητήρων, είναι η συνεργασία και επομένως η συνεχής επικοινωνία που λαμβάνει χώρα, κυρίως μεταξύ των γειτονικών κόμβων. Κάθε κόμβος, ενσωματώνει και έναν επεξεργαστή, ο οποίος του δίνει την δυνατότητα, αντί να στείλει κατευθείαν τα δεδομένα σε έναν καθορισμένο κόμβο που έχει αναλάβει την επεξεργασία-μίξη τους, να χρησιμοποιεί ο ίδιος πρώτα τον επεξεργαστή

του, για την εκτέλεση καθορισμένων απλών υπολογισμών, και στη συνέχεια να αποστέλλει μόνο τα απαραίτητα και μερικώς επεξεργασμένα δεδομένα.

## 1.2 Κόμβοι αισθητήρων

Ένας κόμβος αισθητήρων, αποτελείται από τέσσερα βασικά συστατικά, τη μονάδα που «αισθάνεται» (sensing subsystem), τη μονάδα που επεξεργάζεται (processing subsystem), τη μονάδα της ασύρματης επικοινωνίας (communication subsystem) και τη μονάδα τροφοδότησης ενέργειας (power supply subsystem). Στην εικόνα 1.2 μπορούμε να δούμε τα βασικά υποσυστήματα, τα οποία συνθέτουν το κόμβο-αισθητήρα και περιγράφονται στη συνέχεια [6].



Εικόνα 2. Αρχιτεκτονική Ασύρματου Αισθητήρα

- Το **υποσύστημα αισθητήρων**, αποτελεί το μέσο με το οποίο ο κόμβος αντιλαμβάνεται το φυσικό κόσμο και σκοπός του είναι η μετατροπή ενός φυσικού ή χημικού μεγέθους σε ηλεκτρικό σήμα, το οποίο συνήθως πρόκειται για μια αναλογική τάση. Αποτελείται, κατά κανόνα, από τρία μέρη: τον ηλεκτρονικό αισθητήρα, τα απαραίτητα ηλεκτρονικά για την προετοιμασία του σήματος και την τελική του μετατροπή σε τάση αν αυτό απαιτείται (Signal-Conditioning), και τέλος



τον A/D μετατροπέα (Analog to Digital Converter) για την ψηφιακή μετατροπή του αναλογικού σήματος τάσης. Πολλές φορές, τα τρία αυτά μέρη, είναι δυσδιάκριτα, αφού μπορεί να ενσωματώνονται στο πακέτο του αισθητήρα ή/και του επεξεργαστή, αν πρόκειται για μικροελεγκτή.

- Το **υποσύστημα της επεξεργασίας των δεδομένων**, αποτελεί την καρδιά του συστήματος και ο ρόλος του είναι να ελέγχει και να εκτελεί τις περισσότερες λειτουργίες του ασύρματου κόμβου. Μερικές από αυτές είναι ο έλεγχος και η δειγματοληψία των αισθητήρων, η επεξεργασία, η προσωρινή αποθήκευση των δεδομένων και η λήψη αποφάσεων, η εκτέλεση των τηλεπικοινωνιακών πρωτοκόλλων και των αλγορίθμων δρομολόγησης και εξοικονόμησης ενέργειας. Η υπολογιστική απόδοση του κόμβου και οι ενεργειακές του απαιτήσεις, καθορίζονται σε μεγάλο βαθμό από την επεξεργαστική μονάδα, η οποία και αποτελεί τον δεύτερο σε σειρά καταναλωτή ενέργειας του κόμβου, μετά το υποσύστημα ασύρματης επικοινωνίας. Πολλοί επεξεργαστές, μικροελεγκτές, DSPs ακόμα και FPGA είναι διαθέσιμοι στην αγορά και προσφέρονται για χρήση στους κόμβους-αισθητήρες. Συνήθως χρησιμοποιούνται οι μικροελεγκτές. Στους σύγχρονους μικροελεγκτές, ενσωματώνονται μνήμες τύπου flash και RAM, A/D μετατροπείς και ψηφιακά I/O σε ένα ολοκληρωμένο κύκλωμα χαμηλού κόστους. Η επιλογή του επεξεργαστή, στηρίζεται σε παράγοντες όπως η κατανάλωση ενέργειας, οι απαιτήσεις σε τάση λειτουργίας, το κόστος, η υποστήριξη περιφερειακών, ο χρόνος αφύπνισης και η ταχύτητα του.
- Το **υποσύστημα της επικοινωνίας**, είναι το πλέον σημαντικό κομμάτι ενός κόμβου, μιας και αποτελεί τον μεγαλύτερο καταναλωτή ενέργειας, επηρεάζοντας έτσι την απόδοση του κόμβου αλλά και τη συνολική απόδοση του δικτύου. Θέματα που απασχολούν την έρευνα στον τομέα του υποσυστήματος μετάδοσης αφορούν την ακτίνα εκπομπής, τον τύπο διαμόρφωσης που χρησιμοποιείται, καθώς και τον ρυθμό μετάδοσης δεδομένων. Οι περισσότεροι ασύρματοι κόμβοι χρησιμοποιούν



πομποδέκτες που λειτουργούν στην ISM (Industrial, Scientific, Medical) μπάντα, στις συχνότητες των 433.5 – 437.9 MHz, 868.0 – 868.6 MHz και 2400 – 2483.5 MHz. Χρησιμοποιούν δημοφιλείς μεθόδους διαμόρφωσης, όπως OOK (On/Off Key), ASK (Amplitude Shift Key) και FSK (Frequency Shift Key). Η κατανάλωση ισχύος, είναι σχεδόν ίδια κατά την εκπομπή και τη λήψη δεδομένων και κυμαίνεται στα 15 – 300 mW, ανάλογα με το ολοκληρωμένο, τη συχνότητα και τη μέθοδο που χρησιμοποιεί για την επικοινωνία. Η ακτίνα επικοινωνίας κυμαίνεται από τα 25 – 200 m, με το μέγιστο να παρατηρείται σε εξωτερικούς χώρους, με οπτική επαφή. Οι ρυθμοί μετάδοσης που επιτυγχάνονται είναι της τάξης των 10 – 250 Kbps και πρόσφατα έως και 500Kbps.

- Το **υποσύστημα παροχής ισχύος**, αποτελείται από μία μπαταρία και συνήθως και από ένα dc-dc converter. Σε αυτό το υποσύστημα εντοπίζεται η μεγάλη αδυναμία των ασύρματων δικτύων αισθητήρων. Από τη στιγμή, που θα εγκατασταθεί ένα Ασύρματο Δίκτυο Αισθητήρων σε έναν χώρο, προβλέπεται να λειτουργήσει, για ένα μεγάλο χρονικό διάστημα, χωρίς την ανθρώπινη επίβλεψη. Αυτό σημαίνει, ότι θα πρέπει να διαχειριστεί τους ενεργειακούς του πόρους με τέτοιο τρόπο, ώστε να μεγιστοποιηθεί ο χρόνος ζωής του. Κύρια πηγή ενέργειας των κόμβων είναι οι μπαταρίες, οι οποίες συνήθως δεν είναι επαναφορτιζόμενες. Βέβαια, έχουν προταθεί και μέθοδοι παραγωγής ενέργειας επάνω στον κόμβο και αποθήκευσής της σε επαναφορτιζόμενες μπαταρίες και υπέρ-πυκνωτές, με σκοπό την παράταση του χρόνου ζωής του. Η ενεργειακή κατανάλωση είναι ο δείκτης εκείνος που θα καθορίσει την διάρκεια λειτουργίας του δικτύου.

Εκτός από τις προαναφερόμενες βασικές μονάδες, οι οποίες συνθέτουν τους κόμβους-αισθητήρες, οι κόμβοι μπορεί να ενσωματώνουν και επιπρόσθετες υπομονάδες, όπως γεννήτρια ισχύος (power generator), σύστημα κίνησης (mobilizer) και ένα σύστημα εντοπισμού θέσης (location finding system).



### 1.3 Δικτύων Αισθητήρων και Ad hoc Δίκτυα

Τα χαρακτηριστικά των κόμβων, τα οποία είδαμε στην προηγούμενη ενότητα, εξασφαλίζουν ένα ευρύ φάσμα εφαρμογών για τα δίκτυα αισθητήρων. Μερικοί από τους τομείς εφαρμογής, οι οποίοι θα αναφερθούν αναλυτικότερα και παρακάτω, είναι η υγεία, ο στρατιωτικός τομέας, και το σπίτι. Για παράδειγμα, τα φυσιολογικά στοιχεία ενός ασθενή μπορούν να ελεγχθούν εξ' αποστάσεως από ένα γιατρό. Αυτό διευκολύνει τον ασθενή, ενώ επιτρέπει στο γιατρό να καταλάβει καλύτερα την τρέχοντα κατάσταση του ασθενή. Τα δίκτυα αισθητήρων, μπορούν επίσης να χρησιμοποιηθούν για να ανιχνεύσουν τους ξένους χημικούς παράγοντες στον αέρα και στο νερό. Μπορούν να βοηθήσουν στο προσδιορισμό του τύπου, της συγκέντρωσης, και της θέσης των ρύπων. Στην ουσία, τα δίκτυα αισθητήρων θα παράσχουν στον τελικό χρήστη τη νοημοσύνη και μια καλύτερη κατανόηση του περιβάλλοντος [25].

Η υλοποίηση των παραπάνω, αλλά και άλλων εφαρμογών, απαιτούν τη χρήση τεχνικών ασύρματης ad hoc δικτύωσης. Παρ' όλο που έχουν προταθεί πολλά πρωτόκολλα και αλγόριθμοι για τα παραδοσιακά ασύρματα ad hoc δίκτυα, αυτά δεν είναι κατάλληλα για τα μοναδικά χαρακτηριστικά και τις απαιτήσεις των εφαρμογών των δικτύων αισθητήρων. Αυτό οφείλεται στις διαφορές μεταξύ των δικτύων αισθητήρων και των ad hoc δικτύων, οι οποίες συνοπτικά είναι:

- Ο αριθμός των κόμβων σε ένα δίκτυο αισθητήρων μπορεί να είναι πολλές τάξεις υψηλότερος από τους κόμβους σε ένα ad hoc δίκτυο.
- Οι κόμβοι αισθητήρες αναπτύσσονται σε μεγάλη πυκνότητα.
- Οι κόμβοι αισθητήρων έχουν τη προδιάθεση να αποτυγχάνουν.
- Η τοπολογία ενός δικτύου αισθητήρων αλλάζει πολύ συχνά.
- Οι κόμβοι αισθητήρες χρησιμοποιούν κυρίως επικοινωνία ευρείας εκπομπής (broadcast) , ενώ τα περισσότερα ad hoc δίκτυα είναι βασισμένα σε επικοινωνίες από σημείο σε σημείο.



- Οι κόμβοι αισθητήρες έχουν περιορισμένη ισχύ, υπολογιστική ικανότητα και μνήμη.
- Οι κόμβοι αισθητήρες μπορεί να μην έχουν ένα παγκόσμιο αναγνωριστικό ταυτότητας (ID) λόγω της ύπαρξης ενός μεγάλου αριθμού αισθητήρων.

Ένας από τους σημαντικότερους περιορισμούς στους κόμβους αισθητήρες, είναι η απαίτηση για χαμηλή κατανάλωσης ισχύος. Οι μονάδες αυτές χρησιμοποιούν περιορισμένες, συνήθως μη αντικαταστάσιμες κατά τη διάρκεια της λειτουργίας τους, ενεργειακές πηγές.. Επομένως, ενώ τα παραδοσιακά δίκτυα στοχεύουν να παρέχουν υψηλή ποιότητα υπηρεσίας (QoS), τα πρωτόκολλα δικτύων αισθητήρων πρέπει να εστιάσουν κυρίως, στην εξοικονόμηση ενέργειας. Πρέπει να έχουν ενσωματωμένους μηχανισμούς ανταλλαγής, οι οποίοι να δίνουν στον τελικό χρήστη τη δυνατότητα παράτασης της ζωής του δικτύου, με κόστος τη χαμηλότερη ρυθμαπόδοση ή την υψηλότερη καθυστέρηση μετάδοσης.

#### **1.4 Παράγοντες που επηρεάζουν τη σχεδίαση των δικτύων αισθητήρων**

Τη σχεδίαση ενός δικτύου αισθητήρων, την επηρεάζουν πολλοί παράγοντες. Μερικοί από αυτούς είναι ο χρόνος ζωής (Lifetime), η κάλυψη και επεκτασιμότητα (Coverage and Scalability), το κόστος παραγωγής και η ευκολία ανάπτυξης (Production Cost and Deployment), η αντοχή σε σφάλματα (Fault Tolerance), ο συγχρονισμός και ο χρόνος απόκρισης (Synchronization and Time Response) και η ασφάλεια (Security). Οι παράγοντες αυτοί είναι σημαντικοί, καθώς αποτελούν κατευθυντήριες γραμμές για τη σχεδίαση πρωτοκόλλων και αλγορίθμων για τα δίκτυα αισθητήρων. Επιπλέον, μπορούν να χρησιμοποιηθούν για τη σύγκριση διαφορετικών σχημάτων [13].





### 1.4.1 Χρόνος Ζωής

Αποφασιστικό χαρακτηριστικό στην ανάπτυξη ενός δικτύου αισθητήρων, είναι ο αναμενόμενος χρόνος ζωής του. Κυριότερος περιοριστικός παράγοντας στην διάρκεια ζωής του είναι η χωρητικότητα του συσσωρευτή ενέργειας του συστήματος. Στο σημείο αυτό πρέπει να τονιστεί ότι σε πολλές εφαρμογές, κρίσιμο χαρακτηριστικό δεν είναι ο μέσος χρόνος ζωής ενός κόμβου, αλλά ο ελάχιστος εκτιμώμενος χρόνος ζωής.

Για την αντιμετώπιση του προβλήματος της πεπερασμένης χωρητικότητας των συσσωρευτών, έχουν αναπτυχθεί διάφορες τεχνικές, οι οποίες αποσκοπούν στη μεγιστοποίηση του χρόνου ζωής του δικτύου. Στη σημαντικότερη από αυτές, ο στόχος επιτυγχάνεται με την ελαχιστοποίηση της κατανάλωσης ενέργειας στα συστατικά δομικά στοιχεία του κόμβου και ειδικότερα στον πομποδέκτη, ο οποίος αποτελεί το τμήμα του κόμβου που παρουσιάζει τη μεγαλύτερη κατανάλωση ενέργειας.

Τελευταία, μια νέα μέθοδος τοποθετεί στους κόμβους του δικτύου συσκευές, οι οποίες έχουν τη δυνατότητα να 'συλλαμβάνουν' και να εκμεταλλεύονται την ενέργεια που βρίσκεται διάχυτη στον περιβάλλοντα χώρο του κόμβου. Τέτοιες συσκευές είναι τα ηλιακά κύτταρα, οι πιεζοηλεκτρικές γεννήτριες, οι μικροανεμογεννήτριες κλπ [8].

### 1.4.2 Κάλυψη και Επεκτασιμότητα

Αμέσως επόμενοι σημαντικοί παράγοντες, μετά το χρόνο ζωής, είναι η κάλυψη και η επεκτασιμότητα. Όπως είναι προφανές, είναι σημαντικό για τον τελικό χρήστη να μπορεί να αναπτύξει δίκτυα τα οποία θα έχουν τη δυνατότητα να καλύψουν μια ευρεία περιοχή παρατήρησης. Στο σημείο αυτό, θα πρέπει να τονίσουμε ότι η κάλυψη του δικτύου δεν ταυτίζεται απαραίτητα με την ακτίνα κάλυψης των επικοινωνιακών συνδέσεων που χρησιμοποιεί ο κάθε κόμβος. Με τη χρήση multi-



hor τεχνικών, είναι εφικτή η επέκταση της κάλυψης αρκετά πιο μακριά από την ακτίνα που επιτρέπει ο χρησιμοποιούμενος πομπός. Θεωρητικά, η επέκταση της ακτίνας κάλυψης του δικτύου τείνει στο άπειρο. Αποδεικνύεται όμως, ότι μετά από έναν αριθμό από hops και μια συγκεκριμένη ακτίνα εκπομπής, το συνολικό ισοζύγιο κατανάλωσης ισχύος του δικτύου αυξάνεται ενώ ταυτόχρονα μειώνεται ο χρόνος αντίδρασής του.

Η επεκτασιμότητα, αποτελεί ένα από τα σημαντικότερα χαρακτηριστικά των δικτύων αισθητήρων. Ανάλογα με την εφαρμογή, ο αριθμός των κόμβων που συμμετέχουν κυμαίνεται από λίγους αισθητήρες έως μερικές εκατοντάδες αισθητήρων. Ακόμα, θα πρέπει να υπάρχουν κατάλληλοι μηχανισμοί που να επιτρέπουν την προσθήκη νέων κόμβων χωρίς να διαταράσσεται η λειτουργία του δικτύου.

### **1.4.3 Κόστος Παραγωγής και Ευκολία Ανάπτυξης**

Μιας και τα δίκτυα αισθητήρων αποτελούνται από πολλούς κόμβους, το κόστος κάθε κόμβου παίζει σημαντικό ρόλο στη διαμόρφωση του συνολικού κόστους του δικτύου. Είναι επιθυμητό, το κόστος του κάθε κόμβου να είναι χαμηλό, έτσι ώστε το συνολικό κόστος του δικτύου να είναι χαμηλότερο από το κόστος ενός συμβατικού δικτύου αντίστοιχων δυνατοτήτων.

Επίσης, ένα επιπλέον χαρακτηριστικό, είναι η ευκολία ανάπτυξης. Η περαιτέρω ανάπτυξη του δικτύου στο χώρο λειτουργίας του, πρέπει να είναι εφικτή και από μη εξειδικευμένο προσωπικό. Μια τέτοια δυνατότητα, προϋποθέτει να έχει το δίκτυο την ικανότητα να αυτορυθμίζεται. Στην ιδεατή περίπτωση, το σύστημα θα είναι ικανό να ρυθμίζεται αυτόματα, ανεξάρτητα από την κατάσταση που επικρατεί στο περιβάλλον όπου τοποθετείται.



#### 1.4.4 Αντοχή σε Σφάλματα

Σημαντικός δείκτης απόδοσης ενός συστήματος αισθητήρων, είναι και η αντοχή του στα σφάλματα. Κάποιοι κόμβοι μπορεί να πάψουν να λειτουργούν λόγω βλάβης, λόγω αστοχίας κάποιου υποσυστήματος τους ή λόγω παρεμβολών από κάποιο εξωτερικό αίτιο. Στην περίπτωση αυτή, το σφάλμα λειτουργίας ενός κόμβου ή μιας ομάδας κόμβων δε θα πρέπει να επηρεάζει τη συνολική λειτουργία του δικτύου. Η ιδιότητα αυτή λέγεται αξιοπιστία του συστήματος ή αντοχή σε σφάλματα και ορίζεται ως η ικανότητα διατήρησης των λειτουργιών του δικτύου χωρίς διακοπή, σε περιπτώσεις βλάβης κάποιων κόμβων του δικτύου [16].

Η αξιοπιστία  $R_k(t)$  ενός κόμβου είναι δυνατό να μοντελοποιηθεί χρησιμοποιώντας την κατανομή Poisson, έτσι η πιθανότητα της μη εμφάνισης βλάβης σε ένα χρονικό διάστημα  $(0,t)$  δίνεται από την σχέση:

$$R_k(t) = e^{-\lambda_k t}$$

όπου  $\lambda$  είναι ο ρυθμός εμφάνισης βλάβης στον κόμβο  $k$  σε χρονική περίοδο  $t$ .

#### 1.4.5 Συγχρονισμός και χρόνος απόκρισης

Σε ορισμένες κατηγορίες εφαρμογών, όπως σε εφαρμογές επιτήρησης και εντοπισμού, δεδομένα από πολλούς κόμβους πρέπει να συσχετισθούν χρονικά ώστε γίνει εφικτός ο εντοπισμός βασικών παραμέτρων του φαινομένου που παρατηρείται. Για να επιτευχθεί ο συγχρονισμός αυτός, πρέπει το δίκτυο να είναι ικανό να κατασκευάζει και να διατηρεί μια καθολική ώρα συστήματος, η οποία θα χρησιμοποιείται για τη χρονική ταξινόμηση των δεδομένων, τα οποία καταγράφονται σε κάθε κόμβο του δικτύου. Για να είναι δυνατή η παραπάνω διαδικασία απαιτείται η ύπαρξη ενός μηχανισμού, ο οποίος θα διαχειρίζεται τις



διαδικασίες διατήρησης και διασποράς των μηνυμάτων μεταξύ των κόμβων, με πληροφορίες συγχρονισμού.

Σε κάποιες εφαρμογές, όπως τα συστήματα έγκαιρης προειδοποίησης και συναγερμού, ο χρόνος αντίδρασης είναι ο σημαντικότερος παράγοντας σχεδίασης και αξιολόγησης ενός δικτύου. Ωστόσο, η ικανότητα του δικτύου να έχει μικρό χρόνο απόκρισης έρχεται σε σύγκρουση με άλλους δείκτες, όπως για παράδειγμα το χρόνο ζωής του συστήματος. Η αντίφαση αυτή, αίρεται με την ενσωμάτωση στο δίκτυο, κόμβων οι οποίοι βρίσκονται σε κατάσταση συνεχούς λειτουργίας [2].

#### **1.4.6 Ασφάλεια**

Τα ασύρματα δίκτυα αισθητήρων, θα πρέπει να είναι ικανά να κρατούν κρυφή από μη εξουσιοδοτημένους χρήστες, την πληροφορία που συλλέγουν. Έτσι, για να μπορέσει να διατηρηθεί η μυστικότητα, το δίκτυο πρέπει να υποστηρίζει μηχανισμούς κρυπτογράφησης και αυθεντικότητας.

Η χρήση τέτοιων τεχνικών επιδρούν αρνητικά τόσο στην κατανάλωση ισχύος όσο και στο διαθέσιμο εύρος ζώνης του δικτύου. Ακόμα, η ενσωμάτωση επιπλέον bits στα μεταφερόμενα πακέτα, τα οποία περιέχουν τις πληροφορίες αυθεντικότητας, μειώνουν τον αριθμό των πραγματικών δειγμάτων που μπορούν να μεταφερθούν από ένα κόμβο.

### **1.5 Εφαρμογές στα Ασύρματα Δίκτυα Αισθητήρων**

Το μεγάλο πλήθος των εφαρμογών, οι οποίες μπορούν να υλοποιηθούν μέσω των ασύρματων δικτύων αισθητήρων, οφείλεται κυρίως στη δυνατότητα που μας παρέχουν να χρησιμοποιήσουμε μια μεγάλη ποικιλία αισθητήρων κάθε τύπου. Μερικοί από τους τύπους αισθητήρων είναι οι σεισμικοί, οι θερμικοί, οι μαγνητικοί, οι οπτικοί, οι



αισθητήρες υπέρυθρης ακτινοβολίας και οι αισθητήρες ακουστικών συχνοτήτων. Ακόμα, υπάρχουν αισθητήρες, οι οποίοι είναι ικανοί να απεικονίσουν μια ποικιλία συνθηκών, όπως:

- Θερμοκρασία
- Υγρασία
- Κυκλοφορία οχημάτων
- Συνθήκες φωτός
- Πίεση
- Σύσταση εδάφους
- Επίπεδα θορύβου
- Η παρουσία ή απουσία συγκεκριμένων αντικειμένων στο χώρο
- Τρέχοντα χαρακτηριστικά όπως η ταχύτητα, η κατεύθυνση και το μέγεθος ενός αντικειμένου κ.ά.

Οι κόμβοι αισθητήρων, μπορούν να χρησιμοποιηθούν για διαρκή ανίχνευση κατάστασης και εντοπισμό συμβάντων καθώς και τον εντοπισμό θέσης. Η έννοια της μικρο-αίσθησης και η ασύρματη σύνδεση αυτών των κόμβων υπόσχονται πολλά πεδία εφαρμογών. Οι εφαρμογές αυτές μπορούν να κατηγοριοποιηθούν σε στρατιωτικές, περιβαλλοντικές, εφαρμογές στον τομέα της υγείας και εφαρμογές για έξυπνα σπίτια ή άλλες εμπορικές εφαρμογές. Μπορεί να επεκταθεί αυτή η ταξινόμηση σε περισσότερες κατηγορίες όπως η διαστημική εξερεύνηση, η χημική επεξεργασία και τη πρόληψη καταστροφών.

Στην κατηγορία των στρατιωτικών εφαρμογών έχουμε:

- Παρακολούθηση των κινήσεων του εχθρού
- Διαφύλαξη της ασφάλειας μίας περιοχής
- Απομακρυσμένο έλεγχο υλικού



Στην κατηγορία των περιβαλλοντικών εφαρμογών έχουμε:

- Την καταγραφή της εξελικτικής διαδικασίας ενός οικοσυστήματος (υδάτινου, χερσαίου, δασικού, αστικού)
- Την καταγραφή του μικροκλίματος σε εργασιακούς χώρους και άλλες μεγάλες Εγκαταστάσεις, για τη βελτιστοποίηση της χρήσης των κλιματιστικών συστημάτων.
- Πρόληψη και ανίχνευση εκδήλωσης φωτιάς σε υπαίθριους ή κλειστούς χώρους.
- Παρακολούθηση της εξέλιξης γεωργικών καλλιεργειών
- Καταγραφή γεωφυσικών φαινομένων.

Στην κατηγορία των εφαρμογών οικιακού αυτοματισμού έχουμε:

- Αυτόματη ενεργοποίηση και απενεργοποίηση του φωτισμού σε χώρους που υπάρχει δραστηριότητα.
- Αυτόματη ρύθμιση της θερμοκρασίας ή της έντασης του φωτισμού ανάλογα με τις εξωτερικές κλιματολογικές συνθήκες.

Στην κατηγορία για εφαρμογές ασφαλείας:

- Παρακολούθηση χώρων και για λόγους ασφαλείας και ενημέρωση κάποιας εποπτεύουσας εφαρμογής σε τακτά χρονικά διαστήματα ή όταν λάβει χώρα ένα περιστατικό ενδιαφέροντος, όπως παραβίαση χώρου ή πυρκαγιά.

Στην κατηγορία για εφαρμογές αντιμετώπισης φυσικών καταστροφών:

- Πρόληψη και διάγνωση σεισμικής δραστηριότητας
- Εντοπισμός παγιδευμένων ατόμων(Ray et al)
- Οι Kattapali et al ενσωματώνουν σε κατάλληλα σημεία κατά την κατασκευή



υποδομών αισθητήρες που καταγράφου την καταπόνηση του υλικού και τη μετακίνηση του.

Εφαρμογές στον χώρο της υγείας.

- Συστήματα καταγραφής κρίσιμων βιοσημάτων. Τέτοιου είδους βιοσήματα, είναι το ηλεκτροκαρδιογράφημα, το ηλεκτρομυογράφημα, το ηλεκτροεγκεφαλογράφημα, ο κορεσμός του οξυγόνου στο αίμα, η θερμοκρασία του ασθενούς, η αναπνοή, η πίεση κ.α.
- Σε συνεργασία με περιβαλλοντικούς αισθητήρες για επιδημιολογικές μελέτες.
- Η χορήγηση φαρμάκων

## ***Κεφάλαιο 2<sup>ο</sup>: Ενεργειακή κατανάλωση στα WSN***

Στο κεφάλαιο αυτό, αναλύουμε την ενεργειακή κατανάλωση των WSN, καθώς και περιγράφουμε αναλυτικά τα Επαναφορτιζόμενα Δίκτυα Αισθητήρων. Ακόμα, γίνεται αναλυτική περιγραφή στην Ασύρματη Μεταφορά Ενέργειας και την Ασύρματη επαναφόρτιση δικτύων αισθητήρων. Τέλος, γίνεται ιστορική και βιβλιογραφική αναφορά σε μελέτες που έχουν πραγματοποιηθεί σχετικά με την ασύρματη μεταφορά ενέργειας.

### **2.1 Το πρόβλημα της κατανάλωσης ενέργειας στα WSN**

Ένα ασύρματο δίκτυο αισθητήρων, όπως έχει ήδη αναφερθεί, αποτελείται από κόμβους-αισθητήρες, οι οποίοι απλώνονται σε μια γεωγραφική περιοχή, με σκοπό την παρακολούθηση των φυσικών φαινομένων και τη καταγραφή περιβαλλοντικών παραμέτρων όπως η θερμοκρασία, η υγρασία, δονήσεις, σεισμικά γεγονότα, και άλλα. Τυπικά, ένας κόμβος είναι μια μικροσκοπική συσκευή, η οποία περιλαμβάνει τρία βασικά συστατικά: ένα υποσύστημα αίσθησης για την απόκτηση δεδομένων από το φυσικό περιβάλλον, ένα υποσύστημα επεξεργασίας για την τοπική επεξεργασία και αποθήκευση



των δεδομένων, και ένα ασύρματο υποσύστημα επικοινωνίας για τη μετάδοση και λήψη αυτών των δεδομένων [11].

Επιπλέον, υπάρχει μια πηγή ισχύος, η οποία παρέχει την αναγκαία ενέργεια για τη συσκευή, ώστε να μπορεί να εκτελέσει την προγραμματισμένη λειτουργία της, όπως τη συλλογή δεδομένων, τη τοπική επεξεργασία τους και την αποστολή τους. Αυτή η πηγή ενέργειας, αποτελείται συνήθως από μια μπαταρία με περιορισμένη ενεργειακή ποσότητα, η οποία μπορεί να είναι αδύνατο ή πολύ δύσκολο να αντικατασταθεί ή ακόμα και να επαναφορτιστεί. Αυτό μπορεί να συμβαίνει, καθώς οι κόμβοι μπορεί να τοποθετηθούν σε ένα εχθρικό ή μη πρακτικό περιβάλλον.

Από την άλλη πλευρά, το δίκτυο αισθητήρων πρέπει να έχει μια διάρκεια ζωής αρκετά μεγάλη για να εκπληρώσει τις απαιτήσεις της εφαρμογής. Σε πολλές περιπτώσεις ο χρόνος ζωής της τάξεως των αρκετών μηνών, ή ακόμα και ετών, μπορεί να απαιτηθεί. Επομένως, η κρίσιμη ερώτηση είναι: "πώς μπορεί να επιμηκυνθεί η διάρκεια ζωής του δικτύου για ένα μακρύ χρονικό διάστημα;"

Σε μερικές περιπτώσεις, είναι δυνατό να συλλεχθεί η ενέργεια από το εξωτερικό περιβάλλον (π.χ., με τη χρησιμοποίηση ηλιακών κυττάρων ως πηγή ενέργειας). Εντούτοις, οι εξωτερικές πηγές παροχής ενέργειας συχνά εκθέτουν μια μη συνεχή συμπεριφορά έτσι ώστε ένας ενεργειακός απομονωτής (μια μπαταρία, αποθηκευτικός χώρος ενέργειας) να είναι επίσης απαραίτητος. Σε κάθε περίπτωση, η ενέργεια είναι ένας πολύ κρίσιμος πόρος και για το λόγο αυτόν, θα πρέπει να χρησιμοποιείται με πολύ οικονομία. Επομένως, η ενεργειακή συντήρηση είναι ένα βασικό ζήτημα στο σχεδιασμό των συστημάτων τα οποία βασίζονται σε ασύρματα δίκτυα αισθητήρων [19].

Επιπλέον, τα αμέσως επόμενα χρόνια, δεν αναμένονται καινοτομίες στην τεχνολογία των μπαταριών. Όσο νέες τεχνολογίες, όπως τα κύτταρα καυσίμων (fuel cells), δεν περνούν από το επίπεδο του πρωτοτύπου στη μαζική παραγωγή, οι μπαταρίες και άρα η κατανάλωση ενέργειας θα είναι ο περιοριστικός παράγοντας. Οι μπαταρίες συχνά αποτελούν πάνω από το 50% του βάρους και του όγκου όλης της συσκευής. Για παράδειγμα, ο κόμβος του Mica2 ζυγίζει 18 g. Αντίθετα, 2 AA μπαταρίες ζυγίζουν 20-30 gr έκαστη.





Εξαιτίας αυτής της περιορισμένης ενεργειακής πηγής, η κατανάλωση ενέργειας είναι ένας κρίσιμος σχεδιαστικός παράγοντας για σχεδιαστές τόσο υλικού, όσο και λογισμικού. Παρόλο όμως, που το υλικό και το λογισμικό είναι άρρηκτα συνδεδεμένα στην ανάπτυξη κινητών και ενσωματωμένων συσκευών, η κατανάλωση ενέργειας αποτελεί δευτερεύων ζήτημα για την ανάπτυξη λογισμικού.

Για να γίνει εφικτή μια αποτελεσματική σχεδίαση υλικού και λογισμικού τέτοιων συσκευών, είναι κρίσιμη μια σε βάθος αξιολόγηση εφαρμογών και συστημάτων, όσον αφορά στην κατανάλωση ενέργειας. Για τους σχεδιαστές, λοιπόν, σε επίπεδο λογισμικού είναι πολύ σημαντικό να αξιολογήσουν με ακρίβεια την κατανάλωση ενέργειας κάθε εφαρμογής, αφού αυτή μπορεί να επηρεαστεί αρκετά από την επιλογή των αλγορίθμων και της τεχνικής προγραμματισμού.

Έτσι λοιπόν, στη συνέχεια παρουσιάζονται σύγχρονοι μέθοδοι που έχουν προταθεί, για τον υπολογισμό, είτε μέσω hardware είτε μέσω software, της κατανάλωσης ενέργειας των κόμβων ενός ασύρματου δικτύου αισθητήρων και τεχνικές για τη μείωση της κατανάλωσης ενέργειας.

## 2.2 Επαναφορτιζόμενα Δίκτυα Αισθητήρων

Κατά το σχεδιασμό ενός ασύρματου δικτύου αισθητήρων, υπάρχουν πολλοί σημαντικοί παράγοντες, οι οποίοι θα πρέπει να ληφθούν υπ' όψιν. Μερικοί από αυτούς, είναι το περιβάλλον στο οποίο θα τοποθετηθεί το δίκτυο, οι περιορισμοί στο υλικό, η ανοχή σε σφάλματα και το κόστος παραγωγής. Ένας όμως από τους σημαντικότερους παράγοντες, αποτελεί η διαχείριση των ενεργειακών πόρων των κόμβων των αισθητήρων, οι οποίοι απαρτίζουν αυτά τα δίκτυα.

Κάθε κόμβος σε ένα ασύρματο δίκτυο αισθητήρων, είναι εξοπλισμένος με μία πηγή ενέργειας, η οποία συνήθως είναι μία μικρής χωρητικότητας μπαταρία. Κάθε αισθητήρας, καταναλώνει αρκετή ενέργεια προκειμένου να πραγματοποιήσει τις λειτουργίες του. Οι λειτουργίες αυτές περιλαμβάνουν την παρακολούθηση του περιβάλλοντός τους, την



παραγωγή δεδομένων, κ.α.. Ένα μεγάλο βέβαια ποσοστό ενέργειας, καταναλώνεται κατά την αποστολή και λήψη δεδομένων από έναν άλλο κόμβο.

Όπως γίνεται εύκολα αντιληπτό, εάν ένας κόμβος δεν έχει πλέον ενέργεια για να λειτουργήσει, τότε αυτό μπορεί να οδηγήσει στη δυσλειτουργία ή ακόμα και στην κατάρρευση ολόκληρου του δικτύου. Συνεπώς, η σωστή διαχείριση της ενέργειας είναι υψίστης σημασίας και αποτελεί κρίσιμο κομμάτι στο σχεδιασμό και στη λειτουργία του δικτύου [20].

Οι ερευνητές στον τομέα αυτό, κατευθύνονται κυρίως προς 3 κατευθύνσεις. Η μία κατεύθυνση είναι το πως μπορεί ένας αισθητήρας να καταναλώνει όσο το δυνατόν λιγότερη ενέργεια. Ο τομέας αυτός, ασχολείται κυρίως με την ανάπτυξη πρωτοκόλλων δρομολόγησης με κύριο σκοπό τους τη διατήρηση της ενέργειας. Ωστόσο, με τον τρόπο αυτό δε λύνεται οριστικά το πρόβλημα, καθώς οι αλγόριθμοι αυτοί, απλά μπορούν να καθυστερήσουν την κατάρρευση του δικτύου.

Η δεύτερη κατεύθυνση, αφορά το πως μπορεί η μπαταρία κάθε αισθητήρα να επαναφορτίζεται χρησιμοποιώντας ανανεώσιμες πηγές ενέργειας, όπως η ηλιακή ενέργεια ή η αιολική ενέργεια. Βέβαια, εδώ τίθενται αρκετοί περιορισμοί ως προς το περιβάλλον λειτουργίας του δικτύου αλλά και ως προς τη διαθεσιμότητα αυτών των πηγών ενέργειας, καθ' όλη τη διάρκεια λειτουργίας του. Για παράδειγμα, αν έχει δημιουργηθεί ένα δίκτυο αισθητήρων, το οποίο βασίζεται στην επαναφόρτιση με ηλιακή ενέργεια, και το δίκτυο αυτό έχει αναπτυχθεί σε περιοχή, στην οποία υπάρχει ηλιοφάνεια 8 μήνες το χρόνο, τότε θα υπάρχει πρόβλημα στη λειτουργία του δικτύου τους υπόλοιπους 4 μήνες.

Τέλος, τρίτη πολύ σημαντική κατεύθυνση, είναι η δημιουργία ασύρματων επαναφορτιζόμενων δικτύων, στα οποία η μεταφορά ενέργειας στους κόμβους γίνεται με ασύρματο τρόπο.

### **2.3 Ασύρματη Μεταφορά Ενέργειας**

Η φράση «ασύρματη μεταφορά ενέργειας», αναφέρεται στη διαδικασία εκείνη, κατά την οποία μεταφέρεται ηλεκτρική ενέργεια από μία πηγή σε μία συσκευή χωρίς τη χρήση



καλωδίων ή αγωγών. Είναι ένας γενικός όρος, ο οποίος αναφέρεται σε έναν αριθμό διαφορετικών τεχνολογιών μετάδοσης ισχύος, που χρησιμοποιούν χρονικώς μεταβαλλόμενα ηλεκτρομαγνητικά πεδία.

Το 1826, ο André-Marie Ampère ανέπτυξε το νόμο του Αμπέρ, όπου δείχνει ότι το ηλεκτρικό ρεύμα παράγει ένα μαγνητικό πεδίο. Μετά από 5 χρόνια, το 1831, ο Michael Faraday, ανέπτυξε το νόμο επαγωγής Faraday, ο οποίος περιγράφει την ηλεκτρομαγνητική δύναμη που προκαλείται σε έναν αγωγό, από μία χρονικά μεταβαλλόμενη μαγνητική ροή. Το 1862, ο James Clerk Maxwell, χρησιμοποιώντας τις προαναφερθείσες, καθώς και άλλες παρατηρήσεις, αποτελέσματα πειραμάτων και εξισώσεων πάνω στον ηλεκτρισμό, το μαγνητισμό και την οπτική, συνέθεσε μία θεωρία από την οποία απορρέουν οι εξισώσεις του Maxwell. Αυτό το σύνολο μερικών διαφορικών εξισώσεων, αποτελεί τη βάση για το σύγχρονο ηλεκτρομαγνητισμό, συμπεριλαμβανομένης και της ασύρματης μεταφοράς ενέργειας.

Στα μέσα της δεκαετίας του 1890, ο Νικόλαος Τέσλα, ήταν εκείνος ο οποίος πραγματοποίησε τα πρώτα πειράματα για την ασύρματη μεταφορά ενέργειας. Έχοντας στο μυαλό του το όνειρο του για έναν κόσμο με Ελεύθερη Ενέργεια για όλους, πήγε στο Κολοράντο Σπρινγκς, όπου αναζήτησε μία τοποθεσία απομονωμένη για να διεξάγει τα πειράματά του. Στις αρχές του 1899, εξασφάλισε νέες επενδύσεις από έναν αριθμό πλούσιων προσωπικοτήτων, μεταξύ των οποίων και ο John Jacob Astor, και τελικά κατόρθωσε να δημιουργήσει τεχνητές αστραπές μήκους 40 μέτρων αλλά και να ανάψει λαμπτήρες και να θέσει σε λειτουργία συσκευές σε απόσταση δεκάδων χιλιομέτρων. Φεύγοντας από το Κολοράντο Σπρινγκς, γεμάτος αισιοδοξία κι ελπίδες, ο Τέσλα έγραφε στο ημερολόγιό του: «Η δυνατότητα ασύρματης μεταφοράς ενέργειας διαμέσου του φυσικού περιβάλλοντος και σε μεγάλες αποστάσεις, θα δημιουργήσει ανεξάντλητες πηγές πλούτου και δύναμης, θέτοντας ακόμη και την ίδια την ηλιακή ενέργεια στην υπηρεσία του ανθρώπου».

Το 2007, επιστήμονες του MIT ανακοίνωσαν πως κατάφεραν να ανάψουν μια λάμπα των 60 watt, τροφοδοτώντας την με ενέργεια από μια ηλεκτρική πηγή, η οποία βρισκόταν σε απόσταση μεγαλύτερη των 2 μέτρων από αυτήν και χωρίς φυσικά να υπάρχει



οποιαδήποτε φυσική σύνδεση. Οι ερευνητές ονόμασαν το εγχείρημά τους «WiTricity» (από τις λέξεις Wireless και Electricity). Το πείραμα έγινε με δύο χάλκινα καλώδια, διαμέτρου 60 εκατοστών, εκ των οποίων το ένα χρησιμοποιήθηκε ως πομπός και είχε συνδεθεί στην παροχή του ηλεκτρικού ρεύματος, ενώ το άλλο ως δέκτης και τοποθετήθηκε 2 μέτρα μακριά, αφού πρώτα συνδέθηκε με τη λάμπα. Το πείραμα ήταν επιτυχές, αν και μόνο το 40% της εκπεμπόμενης ενέργειας κατάφερε να μεταφερθεί στη λάμπα. Σημειώνεται δε, ότι η λάμπα παρέμεινε αναμμένη ακόμη και όταν ανάμεσα σε αυτήν και τον πομπό τοποθετήθηκαν διάφορα φυσικά εμπόδια, όπως ξύλο, μέταλλο ή ακόμη και ηλεκτρονικές συσκευές. Σήμερα, οι προσπάθειες προς αυτή την κατεύθυνση όλο και πληθαίνουν και έχουν επιτύχει την ασύρματη μεταφορά ενέργειας, έστω και για πολύ κοντινές αποστάσεις [31].

## 2.4 Ασύρματη επαναφόρτιση δικτύων αισθητήρων

Η τεχνολογία ασύρματης φόρτισης, είναι μία πολλά υποσχόμενη τεχνολογία. Οι σύγχρονες εξελίξεις στον τομέα αυτό, καθώς και οι εξελίξεις σχετικά με τα υλικά των μπαταριών που μπορούν να χρησιμοποιηθούν, ανοίγουν το δρόμο για νέες ευκαιρίες, οι οποίες αφορούν τον τρόπο διαχείρισης της ενέργειας στα δίκτυα αισθητήρων. Όσον αφορά το πρώτο κομμάτι, έρευνες έχουν δείξει ότι μπορεί να επιτευχθεί αποδοτικότητα μέχρι και 40% κατά τη μεταφορά 60 Watts ενέργειας σε απόσταση έως και 2 μέτρων. Όσον αφορά το δεύτερο κομμάτι, έχει βρεθεί ένα νέο υλικό για κατασκευή μπαταριών το οποίο επιτρέπει εξαιρετικά γρήγορη φόρτιση.

Αυτή η εξαιρετικά γρήγορη φόρτιση, παρατηρήθηκε σε σχετικά πειράματα που έγιναν με το  $\text{LiFePO}_4$ . Αποδείχθηκε λοιπόν ότι υιοθετεί και συνδυάζει τα πλεονεκτήματα των συμβατικών μπαταριών Li-ion και των υπερ-πυκνωτών, με αποτέλεσμα υψηλή ενεργειακή πυκνότητα και να μπορεί να φορτιστεί μέχρι και 400C. Με αυτόν τον τρόπο, ο χρόνος για μία πλήρης φόρτιση της μπαταρίας, συρρικνώνεται σε μόλις λίγα δευτερόλεπτα [26].



Αυτές οι εξελίξεις, όπως είναι φυσικό, ανοίγουν το δρόμο για τα Ασύρματα Επαναφορτιζόμενα Δίκτυα Αισθητήρων. Αυτά αποτελούνται από αισθητήρες κόμβους, οι οποίοι μπορούν να είναι είτε σταθεροί είτε κινητοί, καθώς επίσης και από κινητούς κόμβους οι οποίοι είναι υπεύθυνοι για την παροχή ενέργειας στους κόμβους αισθητήρες. Στη συνέχεια, δίνονται ορισμένα πλεονεκτήματα χρήσης των Επαναφορτιζόμενων Δικτύων Αισθητήρων:

1. Το θέμα της ενέργειας σε τέτοια δίκτυα θα μπορεί να διευθετηθεί με μεγαλύτερη λεπτομέρεια και πιο αποδοτικά.
2. Το θέμα της διαχείρισης της ενέργειας μπορεί να πραγματοποιηθεί παθητικά από την πλευρά των κόμβων, χωρίς το υπολογιστικό και επικοινωνιακό βάρος που προστίθεται από πολύπλοκους αλγορίθμους διαχείρισης της ενέργειας.
3. Το κομμάτι της διαχείρισης της ενέργειας μπορεί να μελετηθεί και να σχεδιαστεί ανεξάρτητα από τα πρωτόκολλα δρομολόγησης που χρησιμοποιούνται για τη μεταφορά δεδομένων.
4. Δεν εξαρτάται η επαναφόρτιση του δικτύου από περιβαλλοντικούς παράγοντες όπως γίνεται στην περίπτωση χρήσης ανανεώσιμων πηγών ενέργειας.
5. Σε αντίθεση με τις περιπτώσεις όπου έχει προταθεί η αντικατάσταση των μπαταριών σε κόμβους που έχουν ξεμείνει από ενέργεια, εδώ δεν απαιτείται κάτι σχετικό. Το μεγαλύτερο μειονέκτημα που αφορά την αντικατάσταση των μπαταριών έχει να κάνει με το γεγονός ότι οι κόμβοι δεν είναι πάντα προσβάσιμοι από τους ανθρώπους ή από robots σε εύλογο χρονικό διάστημα. Ένα παράδειγμα είναι όταν το δίκτυο είναι τοποθετημένο στο βυθό της θάλασσας.
6. Η τεχνολογία ασύρματης μεταφοράς ενέργειας είναι φιλική προς το περιβάλλον.



## 2.5 Βιβλιογραφική Αναφορά

Παρόλο που τα πλεονεκτήματα από τα Ασύρματα Επαναφορτιζόμενα Δίκτυα Αισθητήρων είναι φανερά, η αλήθεια είναι ότι πολύ λίγη δουλειά έχει αναφερθεί στο πως θα μπορούσε να σχεδιαστεί και να υλοποιηθεί ένα ασύρματο σύστημα φόρτισης για τα δίκτυα αισθητήρων.

Σχεδόν σε όλες τις περιπτώσεις όμως, στο σχεδιασμό υποθέτουμε την ύπαρξη ενός ή περισσότερων φορτιστών, οι οποίοι επωμίζονται το έργο της επαναφόρτισης των κόμβων του δικτύου. Ένα από τα σημαντικότερα ζητήματα που πρέπει όμως να λυθεί, είναι να βρεθεί ο καλύτερος σχεδιασμός, ώστε να βελτιωθεί η ενεργειακή απόδοση αλλά και να επιμηκυνθεί ο χρόνος ζωής του δικτύου, δηλαδή ο χρόνος πριν την κατάρρευσή του.

Πρώτη φορά η ιδέα για τα Ασύρματα Επαναφορτιζόμενα Δίκτυα Αισθητήρων μελετήθηκε στην εργασία των Rahimi, Shah, Sukhatme, Heideman και Estrin [21]. Οι συγγραφείς της εργασίας αυτής, μελετούν ποια πρέπει να είναι τα χαρακτηριστικά σχετικά με την ενέργεια, τα οποία θα πρέπει να έχει ένα δίκτυο, με σκοπό να μπορεί να αυτοσυντηρείται, δηλαδή να μπορεί να δουλεύει επ' άπειρον. Ακολουθούν μία μέθοδο, σύμφωνα με την οποία ένα μικρό ποσοστό των κόμβων του δικτύου είναι κινητοί και αυτοί οι κόμβοι μπορούν και ψάχνουν για ενέργεια, επαναφορτίζονται και στη συνέχεια διανέμουν αυτή την ενέργεια και στους υπόλοιπους σταθερούς κόμβους του δικτύου.

Στην εργασία "On optimal scheduling in wireless rechargeable sensor networks for stochastic event capture" των Jiang, Cheng και Chen [14], οι συγγραφείς μελετούν το πρόβλημα τόσο της τροχιάς που θα ακολουθεί ο φορτιστής, όσο και την πολιτική της λειτουργίας του αισθητήρα κόμβου.

Υπάρχουν επίσης μελέτες, όπως αυτή των Zhao, Li και Yang [28], όπου υπάρχει ένας κινητός κόμβος, ο οποίος λειτουργεί ως μεταφορέας ενέργειας για τους σταθερούς κόμβους αλλά και ως συλλογέας δεδομένων. Στην περίπτωση αυτή, ακολουθούνται 2 βήματα. Στο πρώτο βήμα, η κινητή οντότητα βρίσκει τα σημεία εκείνα στα οποία υπάρχουν κόμβοι όπου η ενέργειά τους είναι χαμηλή, ενώ παράλληλα η συνολική διαδρομή που ακολουθεί δεν ξεπερνάει σε μήκος ένα κατώτατο όριο που έχει οριστεί. Στο



δεύτερο βήμα, προσπαθούν να λύσουν το θέμα της συγκέντρωσης των δεδομένων από τους κόμβους.

Ιδιαίτερο ενδιαφέρον παρουσιάζει η εργασία των Li, Peng, Zhang και Qiao [17], όπου χρησιμοποιείται η ύπαρξη, εκτός από έναν κινητό φορτιστή, ενός σταθμού-βάση. Ο σκοπός είναι να συνδυαστούν αποδοτικά για να λύσουν το πρόβλημα της δρομολόγησης δεδομένων και της φόρτισης των σταθερών κόμβων. Η τεχνική που ακολουθείται, είναι ότι κάθε αισθητήρας στέλνει τα δεδομένα του στο κέντρο ελέγχου χρησιμοποιώντας το Collection Tree Protocol (CTP). Παράλληλα όμως επισυνάπτονται και άλλες μετρικές μαζί με τα δεδομένα, όπως ο ρυθμός κατανάλωσης ενέργειας, το επίπεδο της ενέργειας, κα. Στη συνέχεια, ο σταθμός βάση, σύμφωνα με τις πληροφορίες από τους αισθητήρες, προγραμματίζει τις ενέργειες που πρέπει να γίνουν για την επαναφόρτιση και δίνει τις αντίστοιχες εντολές στο φορτιστή.

Γενικά, αυτό που παρατηρείται, είναι ότι υπάρχουν πολλοί παράμετροι που πρέπει να ρυθμιστούν στον τρόπο εφαρμογής της ασύρματης φόρτισης στα δίκτυα αισθητήρων. Είναι δηλαδή πολλές οι εναλλακτικές επιλογές, οι οποίες έχουν να κάνουν με το θέμα του σχεδιασμού και της υλοποίησης της πολιτικής που θα ακολουθηθεί για την ασύρματη φόρτιση. Μερικές από τις επιλογές αυτές είναι οι εξής:

- Η πολιτική φόρτισης που θα ακολουθηθεί, για παράδειγμα αν θα υπάρχει πλήρης ή μερική φόρτιση ενός κόμβου.
- Ο λόγος της ενέργειας που διατίθεται στον φορτιστή σε σχέση με τη συνολική ενέργεια που είχε αρχικά αποδοθεί στο δίκτυο.
- Η βελτιστοποίηση του τρόπου με τον οποίο ο φορτιστής διασχίζει το δίκτυο, δηλαδή να βρεθεί ποια είναι η καλύτερη τροχιά που πρέπει να ακολουθήσει ο φορτιστής.



## Κεφάλαιο 3<sup>ο</sup>: Δρομολόγηση και Ενέργεια

Στο κεφάλαιο αυτό αναλύουμε προβλήματα της Δρομολόγησης και της Ενέργειας. Δυο θέματα, τα οποία έχουν σημαντικό ρόλο στη λειτουργία των δικτύων αισθητήρων. Εξηγούμε λεπτομερώς τους ορισμούς της Δρομολόγησης και του Πρωτοκόλλου Δικτύων. Επίσης, αναλύεται ο Αλγόριθμος Εύρεσης Συντομότερων Διαδρομών Dijkstra's algorithm και ο Αλγόριθμος Εύρεσης Συντομότερης Διαδρομής Belman-Ford ή Ford-Fulkerson Algorithm, οι οποίοι χρησιμοποιούνται στην υλοποίηση. Τέλος, περιγράφουμε τη Στατική δρομολόγηση, την Δυναμική Δρομολόγηση, την Δρομολόγηση πλημμύρας (flooding) και την Δρομολόγηση διανυσμάτων απόστασης (Distance vector).

### 3.1 Τι είναι δρομολόγηση

Σε ένα οποιοδήποτε δίκτυο υπολογιστών, αυτό που ουσιαστικά γίνεται, είναι η μεταφορά μηνυμάτων από ένα κόμβο σε έναν άλλον. Σαν κόμβο ονομάζουμε κάθε συσκευή, η οποία βρίσκεται στο δίκτυο, όπως υπολογιστές, tablets, κινητά, switches, access points, routers κ.α. Ένα μήνυμα λοιπόν, το οποίο μεταφέρεται από τον έναν κόμβο σε άλλο, θα ακολουθήσει μία διαδρομή, η οποία θα ξεκινά από τον κόμβο πηγή και θα καταλήγει στον κόμβο προορισμού. Συνήθως, υπάρχουν περισσότερες από μια διαδρομές, τις οποίες μπορεί να ακολουθήσει ένα πακέτο, ώστε να φτάσει στον προορισμό του. Έτσι, στόχος είναι να βρεθεί η πιο σύντομη διαδρομή και η πιο ασφαλής. Η εξεύρεση του καλύτερου, όπως λέμε, μονοπατιού καλείται δρομολόγηση [27].

Η δρομολόγηση πακέτων από ένα κόμβο σε ένα άλλο, αποτελεί μια από τις βασικότερες λειτουργίες σε ένα δίκτυο υπολογιστών. Το δίκτυο, συνήθως αποτελείται από υποδίκτυα, όπου τα πακέτα χρειάζεται να πραγματοποιήσουν πολλά βήματα, μέχρι να φτάσουν στον προορισμό τους. Η μόνη αξιοσημείωτη εξαίρεση είναι τα δίκτυα εκπομπής, αλλά ακόμη και εδώ η δρομολόγηση είναι πρόβλημα εάν η πηγή και ο προορισμός δεν είναι στο ίδιο δίκτυο.





Υπάρχουν αρκετοί αλγόριθμοι, οι οποίοι χρησιμοποιούνται με στόχο την εύρεση της κατάλληλης διαδρομή [1] [3] [7] [24]. Σήμερα τα περισσότερα δίκτυα (σχεδόν το 98%), ακολουθούν το μοντέλο TCP-IP. Το μοντέλο αυτό, χωρίζει τις λειτουργίες ενός δικτύου σε επίπεδα με βάση τις πληροφορίες που περιλαμβάνει ένα πακέτο. Στη συνέχεια περιγράφονται πρωτόκολλα, τα οποία αφορούν ακριβώς αυτό το μοντέλο.

### 3.2 Πρωτόκολλο Δικτύων

Πρωτόκολλο, ονομάζεται ένα σύνολο από συμβάσεις, οι οποίες καθορίζουν το πώς πρέπει να πραγματοποιηθεί κάποια διαδικασία. Έτσι και στα δίκτυα, πρωτόκολλο είναι ένα σύνολο από συμβάσεις, οι οποίες καθορίζουν το πώς ανταλλάσσουν μεταξύ τους δεδομένα οι κόμβοι του δικτύου.

Ένα πρωτόκολλο δικτύου, ουσιαστικά καθορίζει το πώς διακινούνται τα δεδομένα, το πώς γίνεται ο έλεγχος και ο χειρισμός των λαθών, κλπ. Σε ένα δίκτυο μπορεί να έχουμε πολλά υποδίκτυα, όπως π.χ το Internet, και έτσι χρειάζεται ένα σύνολο από συμβάσεις, οι οποίες να καθορίζουν το πώς ανταλλάσσουν μεταξύ τους δεδομένα υπολογιστές, οι οποίοι μπορεί να είναι διαφορετικού τύπου και να ανήκουν σε διαφορετικά δίκτυα.

### 3.3 Αλγόριθμοι Δρομολόγησης

Όπως είπαμε παραπάνω, ο ρόλος της δρομολόγησης είναι να κατευθύνει, προωθεί, το πέρασμα των πακέτων από την πηγή τους προς τον προορισμό τους μέσω ενδιάμεσων κόμβων. Ένας αλγόριθμος δρομολόγησης είναι ουσιαστικά η διαδικασία μέσω της οποίας θα γίνει η δρομολόγηση, με στόχο να βρεθεί η καλύτερη διαδρομή. Η δουλειά ενός αλγόριθμου δρομολόγησης, είναι να ορίζει, με βάση μια μέθοδο, την κατασκευή των πινάκων δρομολόγησης ώστε να είναι αποτελεσματική η δρομολόγηση.

Υπάρχει μεγάλη ποικιλία αλγορίθμων δρομολόγησης, οι οποίοι ανάλογα το πρόβλημα δίνουν ανάλογους πίνακες δρομολόγησης σε κάθε δρομολογητή. Σε μικρά



δίκτυα οι πίνακες δρομολόγησης, μπορούν να συμπληρωθούν και με το χέρι. Σε μεγάλα δίκτυα, όπου εμπλέκονται πολύπλοκες και διαρκώς μεταβαλλόμενες τοπολογίες, η χειροκίνητη κατασκευή των πινάκων δρομολόγησης είναι προβληματική. Εντούτοις, τα περισσότερα δημόσια τηλεφωνικά δίκτυα μεταγωγής (*PSTN*), χρησιμοποιούν προϋπολογισμένους πίνακες δρομολόγησης, με εφεδρικές διαδρομές αν η πιο σύντομη μπλοκαριστεί.

Οι αλγόριθμοι δρομολόγησης, ουσιαστικά αφορούν την λεγόμενη δυναμική δρομολόγηση. Η δυναμική δρομολόγηση, προσπαθεί να λύσει αυτό το πρόβλημα, κατασκευάζοντας τους πίνακες δρομολόγησης αυτόματα, βασιζόμενη στις πληροφορίες τις οποίες μεταφέρονται από τα πρωτόκολλα δρομολόγησης, και αφήνει το δίκτυο να ενεργεί σχεδόν αυτόνομα, με σκοπό να αποφεύγει βλάβες και μπλοκαρίσματα. Χρησιμοποιείται κατά κόρον, σε μεγάλα δίκτυα, όπως το Ίντερνετ. Εντούτοις, η ρύθμιση των πρωτοκόλλων δρομολόγησης απαιτεί ικανότητες, αφού ακόμα δεν έχει υιοθετηθεί μια διαδικασία αυτοματοποίησης στην επιλογή του κατάλληλου πρωτοκόλλου δρομολόγησης.

Τα δίκτυα μεταγωγής πακέτων, όπως το Ίντερνετ, χωρίζουν τα δεδομένα σε πακέτα, καθένα από τα οποία περιέχει πληροφορίες για τον προορισμό του ώστε να γίνεται ξεχωριστή δρομολόγηση. Τα δίκτυα μεταγωγής κυκλώματος, όπως τα τηλεφωνικά δίκτυα, εκτελούν και αυτά δρομολόγηση, με σκοπό να βρουν διαδρομές για κυκλώματα (όπως τηλεφωνικές κλήσεις), πάνω από τις οποίες μπορούν να στείλουν μεγάλες ποσότητες δεδομένων, χωρίς να επαναλαμβάνουν συνεχώς τη διεύθυνση του προορισμού.

Ουσιαστικά, η εφαρμογή ενός αλγόριθμου δρομολόγησης γίνεται στο επίπεδο 3 του μοντέλου TCP\_IP (επίπεδο δικτύου), το οποίο είναι υπεύθυνο να αποφασίσει σε ποια γραμμή εξόδου θα πρέπει να μεταδοθεί ένα εισερχόμενο πακέτο.

Ο κάθε αλγόριθμος, καλείται να πάρει αποφάσεις, σχετικά με το αν ο πίνακας δρομολόγησης θα είναι σταθερός ή θα αλλάζει, αφού η βέλτιστη διαδρομή μπορεί να έχει αλλάξει κατά την διάρκεια της ζωής του δικτύου. Άσχετα με το αν οι διαδρομές επιλέγονται ανεξάρτητα για κάθε πακέτο ή μόνο όταν εγκαθίστανται νέες συνδέσεις,



υπάρχουν κάποια βασικά χαρακτηριστικά, τα οποία είναι επιθυμητά σε ένα αλγόριθμο δρομολόγησης.

Τα χαρακτηριστικά αυτά είναι:

- Η ορθότητα
- Η απλότητα
- Η ανθεκτικότητα
- Η σταθερότητα
- Η δικαιοσύνη
- Η βελτιστοποίηση

Η ορθότητα και η απλότητα, αφορούν το να γίνεται σωστά η δρομολόγηση και μάλιστα με τον πιο απλό τρόπο. Η λέξη απλότητα, συνεπάγεται μικρούς πίνακες δρομολόγησης, γεγονός που δίνει την δυνατότητα να χρησιμοποιούμε routers με μικρότερη μνήμη.

Η ανάγκη για ανθεκτικότητα, αφορά τη δρομολόγηση σε μεγάλα δίκτυα, τα οποία θα πρέπει να λειτουργούν με τις μικρότερες αστοχίες. Κατά τη διάρκεια της περιόδου λειτουργίας τους, θα υπάρχουν αστοχίες υλικού και λογισμικού όλων των ειδών. Ο αλγόριθμος δρομολόγησης, θα πρέπει να είναι σε θέση να αντιμετωπίζει αλλαγές της τοπολογίας και της κίνησης χωρίς να απαιτεί να διακόπτονται οι δρομολογητές και να εφαρμόζουν από την αρχή τις ρυθμίσεις τους.

Η σταθερότητα, είναι επίσης ένας σπουδαίος στόχος για τον αλγόριθμο δρομολόγησης. Υπάρχουν αλγόριθμοι δρομολόγησης, οι οποίοι δεν συγκλίνουν ποτέ σε κάποιο σημείο ισορροπίας, όσο χρόνο και να τρέξουν.

Τέλος, η δικαιοσύνη, εξασφαλίζει ότι τα πακέτα μοιράζονται ισομερώς σε περιπτώσεις κατά τις οποίες έχουν ισοδύναμες διαδρομές, ενώ η βελτιστοποίηση, αφορά την δυνατότητα που έχει ο αλγόριθμος, να αναζητά σε κάθε φάση την καλύτερη δυνατή λύση.



### 3.4 Αλγόριθμος Εύρεσης Συντομότερων Διαδρομών Dijkstra's algorithm

Ο αλγόριθμος του Ντάικστρα (Dijkstra), πήρε το όνομά του από τον Ολλανδό Έντογκερ Ντάικστρα, ο οποίος τον επινόησε το 1956 και τον δημοσίευσε το 1959. Πρόκειται για έναν αλγόριθμο εύρεσης συντομότερων διαδρομών (single-source shortest path problem) από κοινή αφετηρία, σε έναν (κατευθυνόμενο ή μη) γράφο με μη αρνητικά βάρη στις ακμές [23]. Ο αλγόριθμος του Dijkstra είναι άπληστος. Δηλαδή, σε κάθε βήμα επιλέγει την τοπικά βέλτιστη λύση, ώσπου στο τελευταίο βήμα συνθέτει μια συνολικά βέλτιστη λύση. Σε περίπτωση κατά την οποία ο γράφος περιέχει αρνητικά βάρη, ο αλγόριθμος του Ντάικστρα δε δίνει σωστό αποτέλεσμα. Για γράφους οι οποίοι μπορεί να έχουν αρνητικά βάρη στις ακμές, χρησιμοποιούνται πιο περίπλοκοι αλγόριθμοι, όπως αυτός των Bellman και Ford ή των Floyd-Warshall.

Ο αλγόριθμος του Dijkstra είναι πλέον ευρέως διαδεδομένος και χρησιμοποιείται σε πολλές εφαρμογές. Χρήση του αλγόριθμου αυτού, κάνει το πρωτόκολλο OSPF, το οποίο είναι το εσωτερικό πρωτόκολλο πύλης δικτύου του Διαδικτύου.

Πιο συγκεκριμένα ο αλγόριθμος Dijkstra ακολουθεί τα παρακάτω βήματα:

**Βήμα 1.** Σημείωσε σε κάθε κόμβο μια ετικέτα απόστασης ( $d[*]$ ) με τιμή 0 στον αρχικό κόμβο και τιμή άπειρο σε όλους τους υπόλοιπους. Επίσης, σημείωσε μια ετικέτα προηγούμενου κόμβου ( $prev[*]$ ) και βάλε της την κενή τιμή για όλους τους κόμβους. Η ετικέτα αυτή χρειάζεται για τον υπολογισμό της ζητούμενης διαδρομής στο τέλος.

**Βήμα 2.** Σημείωσε όλους τους μη-επεξεργασμένους κόμβους ( $S = \emptyset$ ). Ο τρέχων κόμβος είναι ο αρχικός.

**Βήμα 3.** Για τον τρέχων κόμβο, εξέτασε όλους τους μη-επεξεργασμένους γείτονές του και υπολόγισε το συνολικό άθροισμα απόστασής τους από τον αρχικό κόμβο. Για παράδειγμα, αν ο τρέχων κόμβος έχει απόσταση 6 από τον αρχικό και ο γείτονας του τρέχοντος κόμβου, που εξετάζει αυτή τη στιγμή ο αλγόριθμος, έχει απόσταση 2 το συνολικό άθροισμα απόστασης του γείτονα από τον αρχικό κόμβο είναι  $6+2=8$ . Αν αυτή η απόσταση είναι



μικρότερη από την ετικέτα απόστασης που είχε σημειωθεί, αντικατέστησε τη με τη νέα υπολογισμένη τιμή και σημείωσε τον τρέχων κόμβο στην ετικέτα προηγούμενου κόμβου.

**Βήμα 4.** Όταν τελειώσεις με την εξέταση όλων των γειτόνων του τρέχοντος κόμβου, σημείωσέ τον ως επεξεργασμένο. Ένας επεξεργασμένος κόμβος δεν εξετάζεται ποτέ ξανά από τον αλγόριθμο. Η ετικέτα απόστασής της είναι η ελάχιστη και θα παραμείνει σταθερή.

**Βήμα 5.** Ο επόμενος τρέχων κόμβος θα είναι ο μη-επεξεργασμένος κόμβος με τη μικρότερη ετικέτα απόστασης.

**Βήμα 6.** Αν όλοι οι κόμβοι έχουν σημειωθεί ως επεξεργασμένοι, προχώρα στο επόμενο βήμα. Διαφορετικά, συνέχισε από το βήμα 3.

**Βήμα 7.** Ξεκινώντας από τον κόμβο-προορισμό (ο οποίος είναι ο τελευταίος τρέχων κόμβος) εκτύπωσε τον κόμβο που αναγράφεται στην ετικέτα προηγούμενου κόμβου. Επανέλαβε μέχρι η ετικέτα προηγούμενου κόμβου που θα συναντήσεις να είναι άδεια.

### 3.5 Αλγόριθμος Εύρεσης Συντομότερης Διαδρομής Belman-Ford ή Ford-Fulkerson Algorithm

Ο αλγόριθμος Bellman-Ford, εφαρμόζεται κατά κύριο λόγο σε σταθμισμένους κατευθυνόμενους γράφους, για την επίλυση του προβλήματος εύρεσης συντομότερης διαδρομής από έναν κόμβο αφετηρίας προς έναν κόμβο προορισμού (single-source shortest path), στην περίπτωση που ένας γράφος περιλαμβάνει ακμές που έχουν θετικά και αρνητικά βάρη. Ο αλγόριθμος, επιστρέφει μια λογική τιμή (boolean value), η οποία υποδεικνύει εάν υφίσταται ή όχι ένας αρνητικού βάρους κύκλος στο γράφο, ο οποίος είναι προσβάσιμος από τον κόμβο αφετηρίας. Η ύπαρξη ενός τέτοιου κύκλου στο γράφο, σημαίνει ότι το πρόβλημα που εξετάζεται δεν έχει λύση [15].

Στην περίπτωση που δεν υπάρχει κύκλος αρνητικού βάρους, τότε ο αλγόριθμος δίνει ως αποτέλεσμα τις συντομότερες διαδρομές που υφίστανται μεταξύ των κόμβων του



γράφου και τα κόστη διάσχισής τους. Η πολυπλοκότητα του αλγόριθμου Bellman – Ford σε ένα γράφο με  $V$  κόμβους και  $E$  ακμές είναι ίση με  $O(V, E)$ .

Μία περιγραφή του αλγόριθμου φαίνεται στον παρακάτω ψευδοκώδικα:

```
BELLMAN-FORD (G, W, S)
1  INITIALIZE-SINGLE-SOURCE (G, S)
2  for i ← 1 to |V[G]| - 1
3      do for each edge (u,v) ∈ E[G]
4          do RELAX(u, v, w)
5  for each edge (u,v) ∈ E[G]
6      do if d[v] > d[u] + w(u,v)
7          then return FALSE
8  return TRUE
```

Στην πρώτη γραμμή του ψευδοκώδικα, γίνεται η αρχικοποίηση των μεταβλητών  $d$  και  $\pi$ , στις οποίες καταχωρούνται η απόσταση που διανύεται κατά τη διάρκεια τρεξίματος του αλγόριθμου και οι προκάτοχοι κόμβοι αντίστοιχα. Ο αλγόριθμος, υλοποιεί  $|V| - 1$  περάσματα από τις ακμές του γράφου, κάθε ένα από τα οποία αντιστοιχεί σε μια επανάληψη του βρόχου που περιλαμβάνεται στην εντολή for στις γραμμές 2-4 του ψευδοκώδικα. Αφού ολοκληρωθούν τα  $|V| - 1$  περάσματα από τις ακμές του γράφου, ο αλγόριθμος ελέγχει την ύπαρξη κύκλων με αρνητικά βάρη και επιστρέφει την αντίστοιχη boolean τιμή. Εάν υπάρχει κύκλος με αρνητικό βάρος το τρέξιμο του αλγόριθμου σταματά, σε διαφορετική περίπτωση ο αλγόριθμος δίνει ως αποτέλεσμα τα συντομότερα μονοπάτια και το κόστος τους.

### 3.6 Στατική δρομολόγηση

Στη στατική δρομολόγηση, αποφασίζουμε εξ αρχής το δρομολόγιο που θα ακολουθήσει ένα πακέτο για να φτάσει στον προορισμό του. Το δρομολόγιο δεν



προσαρμόζεται δυναμικά στα διάφορα χαρακτηριστικά αλλά ακολουθεί την διαδρομή όπως του την έχει προκαθορίσει ο διαχειριστής δικτύου.

Στατική δρομολόγηση χρησιμοποιείται είτε για λόγους ασφαλείας, όπου ο διαχειριστής έχει προκαθορίσει το δρόμο συγκεκριμένων κρίσιμων πακέτων, είτε σε απλά μικρά δίκτυα, όπου η πολυπλοκότητα είναι αρκετά μικρή ή δρομολόγηση γίνεται με μια λογική που δεν καθορίζεται από ένα αλγόριθμο, αλλά από μια εξειδικευμένη λογική του διαχειριστή ή του οργανισμού στον οποίο ανήκει το δίκτυο. Σε περιπτώσεις στις οποίες υπάρχει και στατική και δυναμική δρομολόγηση, η στατική δρομολόγηση έχει προτεραιότητα.

### 3.7 Δυναμική Δρομολόγηση

Η δυναμική δρομολόγηση, αφορά την αυτόματη δημιουργία του πίνακα δρομολόγησης σε κάθε router. Για να γίνει αυτό, χρησιμοποιούνται διάφοροι αλγόριθμοι δρομολόγησης, οι οποίοι είναι κατάλληλοι να εξερευνήσουν το δίκτυο και να εντοπίσουν τις πιθανές διαδρομές που μπορεί να ακολουθήσει ένα πακέτο για να ξεκινήσει από ένα σημείο του δικτύου και να φτάσει σε ένα οποιοδήποτε άλλο. Ακόμα, σε περίπτωση που μια συγκεκριμένη διαδρομή γίνει μη διαθέσιμη, οι υπάρχοντες κόμβοι πρέπει να αποφασίσουν μια εναλλακτική διαδρομή, την οποία μπορούν να χρησιμοποιήσουν για να στείλουν τα δεδομένα στον προορισμό τους.

Αυτό, επιτυγχάνεται μέσω της χρήσης πρωτοκόλλων δρομολόγησης, τα οποία χρησιμοποιούν μία από τις δυο ευρείες κλάσεις αλγορίθμων δρομολόγησης: αλγορίθμους διανύσματος απόστασης και αλγορίθμους κατάστασης συνδέσμων, οι οποίες περιέχουν σχεδόν κάθε αλγόριθμο δρομολόγησης που χρησιμοποιείται σήμερα στο Ίντερνετ.



### 3.8 Δρομολόγηση πλημμύρας (flooding)

Σε έναν αλγόριθμο δρομολόγησης, ο οποίος χρησιμοποιεί την τεχνική της πλημμύρας, κάθε εισερχόμενο πακέτο στέλνεται σε κάθε εξερχόμενη γραμμή εκτός από αυτή μέσω της οποίας έφτασε. Με τον τρόπο αυτό, δημιουργούνται άπειρα αντίγραφα του πακέτου και θα πρέπει να ληφθούν μέτρα για την ανακοπή της πλημμύρας. Ένα τέτοιο μέτρο, είναι να περιέχεται ένας μετρητής αλμάτων στην κεφαλίδα κάθε πακέτου IP. Σε αυτήν την περίπτωση, ο μετρητής αλμάτων θα πρέπει να έχει τιμή μικρότερη από το πλήθος των αλμάτων που χρειάζονται για να φτάσει το πακέτο από την πηγή στον προορισμό. Για κάθε άλμα, ο μετρητής μειώνεται κατά ένα. Όταν μηδενιστεί, το πακέτο δεν θα αναμεταδοθεί, αλλά θα απορριφθεί [12].

### 3.9 Δρομολόγηση διανυσμάτων απόστασης (Distance vector)

Οι αλγόριθμοι διανυσμάτων απόστασης, χρησιμοποιούν τον αλγόριθμο Bellman-Ford. Αυτή η διαδικασία αναθέτει έναν αριθμό, το κόστος, σε κάθε μια από τις συνδέσεις μεταξύ των κόμβων σε ένα δίκτυο. Οι κόμβοι θα στέλνουν πληροφορίες από το σημείο A στο σημείο B μέσω της διαδρομής που έχει το μικρότερο συνολικό κόστος, δηλαδή το αποτέλεσμα που προκύπτει από την άθροιση του κόστους μεταξύ των κόμβων που χρησιμοποιήθηκαν.

Ο αλγόριθμος λειτουργεί με πολύ απλό τρόπο. Όταν ξεκινάει ένας κόμβος, ξέρει μόνο τους άμεσους γείτονές του, και το κόστος που εμπλέκεται ώστε να φτάσει σε αυτούς. Αυτές οι πληροφορίες, δηλαδή η λίστα με τους προορισμούς, το εμπλεκόμενο κόστος για να φτάσει κανείς σε αυτόν και στον επόμενο κόμβο (hop), σχηματίζουν τον πίνακα δρομολόγησης ή πίνακα αποστάσεων. Κάθε κόμβος, σε τακτικά χρονικά διαστήματα, στέλνει σε κάθε γείτονά του την δική του αντίληψη για το κόστος που εμπλέκεται μέχρι να φτάσει σε όλους τους προορισμούς που του είναι γνωστοί. Οι γειτονικοί κόμβοι εξετάζουν αυτές τις πληροφορίες και τις συγκρίνουν με αυτές που ήδη 'ξέρουν'. Αν οι νέες





πληροφορίες βελτιστοποιούν τους χρόνους που ήδη γνωρίζουν, τις εισάγουν στον δικό τους πίνακα δρομολόγησης. Όσο επαναλαμβάνονται τα βήματα του αλγορίθμου, όλοι οι κόμβοι του δικτύου θα ανακαλύπτουν το καλύτερο επόμενο βήμα (hop) για όλους τους προορισμούς, και το καλύτερο συνολικό κόστος.

Ένα πρωτόκολλο, το οποίο χρησιμοποιεί αλγόριθμο διανυσμάτων απόστασης είναι το RIP, το αρχικό εσωτερικό πρωτόκολλο πύλης δικτύου του Internet. Αργότερα, αντικαταστάθηκε από το OSPF, το οποίο αποτελεί υλοποίηση ενός αλγορίθμου κατάστασης συνδέσεων.

### 3.10 Δρομολόγηση κατάστασης συνδέσεων (Link State)

Στην περίπτωση των αλγορίθμων κατάστασης συνδέσεων, ο κάθε κόμβος χρησιμοποιεί σαν αρχικά δεδομένα ένα χάρτη του δικτύου με την μορφή γράφου. Για να παραχθεί ο γράφος αυτός, κάθε κόμβος πλημμυρίζει ολόκληρο το δίκτυο με πληροφορίες σχετικά με το με ποιούς άλλους κόμβους μπορεί να συνδεθεί, εν συνεχεία κάθε κόμβος συγκεντρώνει όλες αυτές τις πληροφορίες και σχηματίζει έναν χάρτη. Χρησιμοποιώντας αυτό το χάρτη, κάθε δρομολογητής αποφασίζει ανεξάρτητα την καλύτερη διαδρομή από τον εαυτό του προς κάθε άλλο κόμβο.

Ο αλγόριθμος που χρησιμοποιείται για να επιλεγεί η βέλτιστη διαδρομή συνήθως είναι ο αλγόριθμος του Dijkstra, στον οποίο αναφερθήκαμε και παραπάνω. Ο αλγόριθμος Dijkstra, επιλέγει την πιο σύντομη διαδρομή, δημιουργώντας ένα δέντρο, με τον τρέχοντα κόμβο σαν ρίζα του δέντρου, το οποίο περιέχει όλους τους υπόλοιπους κόμβους του δικτύου. Ξεκινάει με ένα δέντρο που περιέχει μόνο τον εαυτό του και συνεχίζει, προσθέτοντας έναν τη φορά, από το σύνολο των κόμβων οι οποίοι δεν έχουν προστεθεί στο δέντρο. Κάθε φορά, προσθέτει τον κόμβο, ο οποίος έχει το μικρότερο κόστος για να φτάσει σε κάποιον γειτονικό κόμβο, ο οποίος ήδη υπάρχει στο δέντρο. Αυτό συνεχίζεται μέχρις ότου όλοι οι κόμβοι να υπάρχουν στο δέντρο.



Στη συνέχεια, το συγκεκριμένο δέντρο χρησιμοποιείται για την κατασκευή του πίνακα δρομολόγησης του κάθε κόμβου, δείχνοντας το καλύτερο επόμενο βήμα (hop), για να φτάσει από τον εαυτό του σε οποιονδήποτε άλλον κόμβο του δικτύου.

## **Κεφάλαιο 4<sup>ο</sup>: Δρομολόγηση σε δίκτυα Αισθητήρων**

Στο κεφάλαιο αυτό αναλύουμε προβλήματα της Δρομολόγησης σε δίκτυα Αισθητήρων. Καθώς, η δρομολόγηση είναι ένα από τα βασικά ζητήματα στα δίκτυα αισθητήρων, όπως προαναφέραμε, γίνεται αναφορά σε πρωτόκολλα δρομολόγησης, τα οποία ομαδοποιούν τους αισθητήρες, τεχνική την οποία χρησιμοποιούμε κατά την υλοποίηση του αλγορίθμου στο κεφάλαιο 5. Συγκεκριμένα, το «Cluster Based Routing Protocol», το «Global State Routing Πρωτόκολλο (GSR)», «Fisheye State Routing protocol» και το «Zone Routing Protocol»

### **4.1 Cluster Based Routing Protocol**

Στις περιπτώσεις των ασύρματων δικτύων, πολλές φορές χρησιμοποιούμε πρωτόκολλα και αλγόριθμους που χειρίζονται τους κόμβους ανά ομάδες (clusters). Στην περίπτωση αυτή λέμε ότι έχουμε cluster based πρωτόκολλα. Η μορφή του cluster που χρησιμοποιείται, δημιουργείται συνήθως μέσω του επόμενου αλγόριθμου. Όταν εμφανίζεται ένας κόμβος, βρίσκεται σε κατάσταση αμφιβολίας, ξεκινάει έναν χρονομετρητή και μεταδίδει ένα μήνυμα Hello. Όταν ο επικεφαλής cluster λαμβάνει αυτό το μήνυμα ανταποκρίνεται με ένα αυτόματο μήνυμα hello. Μόλις ο μη-καθορισμένος κόμβος δέχεται αυτό το μήνυμα από τον επικεφαλής cluster, καθορίζει το καθεστώς του ως «μέλος». Στη περίπτωση που ο χρόνος αναμονής του μη-καθορισμένου κόμβου λήξει, μετατρέπεται σε επικεφαλής cluster εάν διαθέτει διπλής κατεύθυνσης link προς κάποιον γείτονα, αλλιώς παραμένει σε μη-καθορισμένο καθεστώς και επαναλαμβάνει την διαδικασία. Οι επικεφαλείς cluster, είναι προτιμότερο να αλλάζουν όσο γίνεται σπανιότερα [9].



Ο κάθε κόμβος περιέχει έναν πίνακα γειτόνων. Για κάθε γείτονα, ο πίνακας γειτόνων περιέχει το καθεστώς της σύνδεσης (εάν είναι μιας ή δύο κατευθύνσεων) και το επίπεδο των γειτόνων (εάν είναι επικεφαλής cluster ή εάν αποτελούν μέλη). Ο επικεφαλής cluster συλλέγει πληροφορίες για τα μέλη του cluster του και επιπλέον διατηρεί έναν πίνακα γειτονικών cluster, ο οποίος περιέχει πληροφορίες για τους γειτονικούς cluster. Για κάθε γειτονικό cluster, ο πίνακας αυτός έχει μια εγγραφή, η οποία περιέχει το δρομολόγιο με το οποίο ο cluster αυτός μπορεί να εντοπιστεί, αλλά και τον επικεφαλής της συγκεκριμένης ομάδας.

Όταν μια πηγή θέλει να στείλει δεδομένα σε κάποιον προορισμό, μεταδίδει αίτημα δρομολόγησης. Λαμβάνοντας την απαίτηση αυτή ο επικεφαλής cluster ελέγχει για να δει εάν ο προορισμός βρίσκεται στην περιοχή του cluster του. Εάν όντως ο προορισμός βρίσκεται μέσα στην περιοχή του, στέλνει την απαίτηση κατευθείαν στον προορισμό που του ζητήθηκε. Στην αντίθετη περίπτωση στέλνει την απαίτηση σε όλους τους διπλανούς του επικεφαλής cluster.

Όταν ο προορισμός λαμβάνει το αίτημα δρομολόγησης, απαντάει πίσω με την διαδρομή η οποία καταγράφηκε μέσα στο πακέτο της απαίτησης. Εάν η πηγή δεν λάβει μια απάντηση μέσα σε ένα συγκεκριμένο χρονικό διάστημα, στέλνει ξανά μια απαίτηση διαδρομής.

Στα Cluster based πρωτόκολλα, η δρομολόγηση πραγματοποιείται με την χρησιμοποίηση της δρομολόγησης της συντομότερης διαδρομής. Δηλαδή, λαμβάνοντας ένα αίτημα δρομολόγησης, ο κόμβος προσπαθεί να βρει τον πιο κοντινό κόμβο στην διαδρομή και στέλνει το πακέτο σ' αυτόν, συμβάλλοντας με αυτόν τον τρόπο στην μείωση της διαδρομής.

Κατά την προώθηση του πακέτου, εάν ένας κόμβος παρατηρήσει την ύπαρξη μίας σπασμένης σύνδεσης, στέλνει ένα μήνυμα σφάλματος στην πηγή και στη συνέχεια χρησιμοποιεί έναν μηχανισμό τοπικής επιδιόρθωσης της σύνδεσης. Σύμφωνα με τον μηχανισμό αυτόν, όταν ένας κόμβος βρει πως ο επόμενος κόμβος δεν μπορεί να



εντοπιστεί, ελέγχει για να δει εάν ο επόμενος κόμβος μπορεί να εντοπιστεί μέσω ενός από τους γείτονες του. Εάν λειτουργήσει ένας από τους δύο τρόπους, το πακέτο θα σταλεί μέσω του διορθωμένου μονοπατιού.

## 4.2 Global State Routing Πρωτόκολλο (GSR)

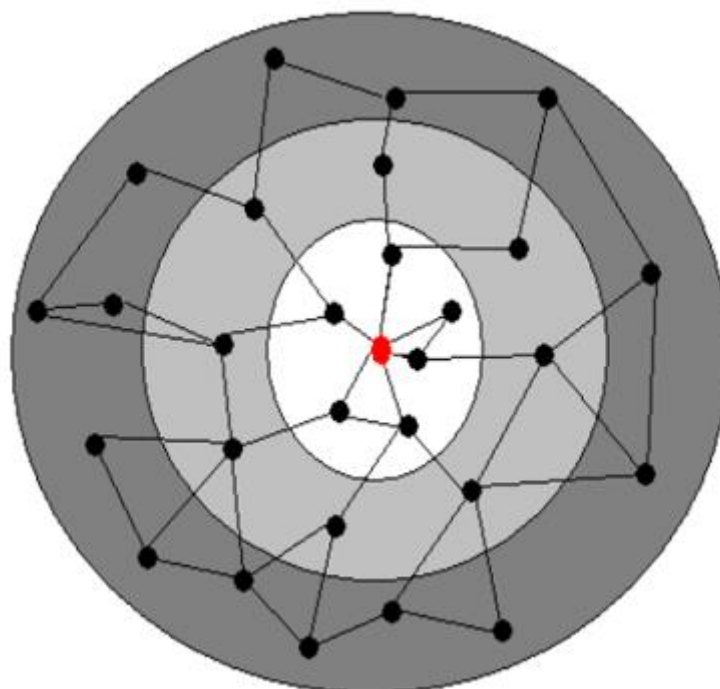
Το πρωτόκολλο Global-State Routing (GSR), είναι ένα πρωτόκολλο, το οποίο ουσιαστικά λειτουργεί με βάση την ιδέα του link state routing, βελτιώνοντας πολύ την συμπεριφορά του, με το να αποφεύγει τον καταιγισμό από μηνύματα δρομολόγησης [10]. Σε αυτόν τον αλγόριθμο, ο κάθε κόμβος περιέχει τα εξής στοιχεία: μια λίστα γειτόνων, έναν πίνακα τοπολογιών, έναν Next Hop πίνακα και τέλος έναν πίνακα απόστασης. Η λίστα των γειτόνων, περιέχει την λίστα με όλους τους γειτονικούς κόμβους, με την διαφορά ότι όσοι κόμβοι μπορούν να ακουστούν από έναν κόμβο υποτίθεται ότι είναι και γείτονες του. Για κάθε κόμβο προορισμού, ο πίνακας τοπολογιών, περιέχει πληροφορίες για την κατάσταση των γραμμών, όπως αναφέρεται από τον προορισμό και την χρονοσφραγίδα της πληροφορίας. Επιπλέον, ο πίνακας Next Hop, περιέχει τον επόμενο κόμβο, στον οποίο θα πρέπει να προωθηθούν τα πακέτα, ενώ ο πίνακας της απόστασης, περιέχει την κοντινότερη απόσταση για κάθε κόμβο προορισμού.

Τα μηνύματα δρομολόγησης, παράγονται με την αλλαγή μιας σύνδεσης, όπως ακριβώς και στα πρωτόκολλα link state. Όταν παραλαμβάνει ένα μήνυμα δρομολόγησης, ο κόμβος αναβαθμίζει τον πίνακα τοπολογιών του, μόνο στην περίπτωση που ο σειριακός αριθμός του μηνύματος είναι πιο πρόσφατος από τον αντίστοιχο αριθμό που είναι αποθηκευμένος στον πίνακα. Μετά από αυτήν την κατάσταση, ο κόμβος επαναπροσδιορίζει τον πίνακα δρομολόγησης του και μεταδίδει την πληροφορία στους γειτονικούς του κόμβους



### 4.3 Fisheye State Routing protocol

Το πρωτόκολλο Fisheye State Routing (FSR), αποτελεί μια διόρθωση του πρωτοκόλλου GSR. Το μεγάλο μέγεθος των μηνυμάτων αναβάθμισης στα πρωτόκολλα GSR σπαταλά μια αρκετά μεγάλη ποσότητα από το bandwidth του δικτύου. Για τον λόγο αυτόν, στα πρωτόκολλα FSR, κάθε μήνυμα αναβάθμισης δεν περιέχει πληροφορίες για όλους τους κόμβους. Αντιθέτως, ανταλλάσσει πληροφορίες για κοντινότερους κόμβους πολύ συχνότερα από ότι ανταλλάσσει για πιο μακρινούς με αποτέλεσμα να μειώνεται το μέγεθος του μηνύματος αναβάθμισης. Με αυτόν τον τρόπο, κάθε κόμβος αποκτά ακριβή πληροφόρηση για τους γείτονες του και αυτή η ακρίβεια στην πληροφόρηση μειώνεται καθώς η απόσταση από τον συγκεκριμένο κόμβο μεγαλώνει. Παρακάτω παρουσιάζεται τα πρωτόκολλο FSR. Ο score καθορίζεται σύμφωνα με τους κόμβους που μπορούν να έρθουν σε επαφή σε έναν συγκεκριμένο αριθμό βημάτων (hops). Ο κεντρικός κόμβος κατέχει την πιο ακριβή πληροφόρηση για όλους τους κόμβους που βρίσκονται στον άσπρο κύκλο του σχήματος και αυτό συνεχίζεται και στους υπόλοιπους κύκλους [18].

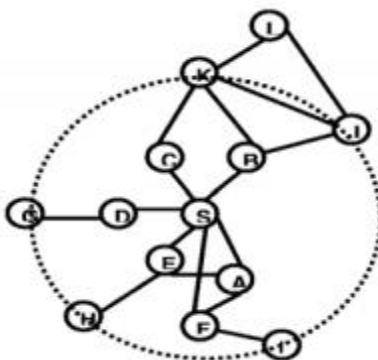


Εικόνα 3. Ακρίβεια πληροφορίας σε ένα πρωτόκολλο FSR

Αν και στα πρωτόκολλα FSR ένας κόμβος δεν έχει ακριβή πληροφόρηση για τους απομακρυσμένους κόμβους, τα πακέτα δρομολογούνται επακριβώς γιατί η πληροφόρηση του μονοπατιού γίνεται ολοένα και πιο ακριβής καθώς το πακέτο πλησιάζει στον προορισμό.

#### 4.4 Zone Routing Protocol

Το ZRP, είναι ένα υβριδικό reactive/proactive πρωτόκολλο, το οποίο βασίζει την λειτουργία του σε μία παράμετρο που ονομάζεται ζώνη δρομολόγησης ή απλά ζώνη (Routing Zone). Μία ζώνη, ακτίνας  $\rho$ , καθορίζεται για κάθε κόμβο και περιλαμβάνει όλους τους κόμβους για τους οποίους η απόσταση, σε αριθμό hops, από τον κόμβο που εξετάζεται, είναι το πολύ  $\rho$  hops. Ένα παράδειγμα μιας ζώνης των δύο hops απεικονίζεται στο παρακάτω σχήμα:



Εικόνα 4. Ένα παράδειγμα μιας ζώνης των δύο hops

Στο ZRP, κάθε κόμβος διατηρεί πληροφορίες δρομολόγησης μόνο για τους κόμβους οι οποίοι ανήκουν στην ζώνη του. Με αυτόν τον τρόπο, τα πακέτα ενημέρωσης περιορίζονται σε τοπικό επίπεδο και η επιβάρυνση στο διαθέσιμο εύρος ζώνης λόγω



ανταλλαγής πληροφοριών ελέγχου, δεν εξαρτάται από τον αριθμό των κόμβων που απαρτίζουν το δίκτυο, ο οποίος μπορεί να είναι αρκετά μεγάλος.

Κάθε κόμβος, διατηρεί τις απαραίτητες πληροφορίες για την ζώνη του, με την βοήθεια ενός πρωτοκόλλου, το οποίο ενεργεί μέσα στην ζώνη και ονομάζεται “Intrazone Routing Protocol” (IARP). Το IARP, είναι συνήθως ένας καθαρά proactive αλγόριθμος ή ένας reactive αλγόριθμος με πολλά χαρακτηριστικά proactive. Για τη δρομολόγηση σε προορισμούς, οι οποίοι δεν ανήκουν στην ζώνη του υπό εξέταση κόμβου, χρησιμοποιείται το “Interzone Routing Protocol” (IERP), το οποίο είναι ένα πρωτόκολλο που ανακαλύπτει με reactive τρόπο, διαδρομές εκτός ζώνης. Το IERP, διαφοροποιείται από τα κλασσικά reactive πρωτόκολλα, τα οποία βασίζονται στην μέθοδο πλημμυρίσματος του δικτύου με πακέτα αναζήτησης διαδρομής, καθώς εκμεταλλεύεται την δομή που δημιουργείται από τις ζώνες των κόμβων. Οι ζώνες μειώνουν αισθητά την επιβάρυνση στο διαθέσιμο εύρος ζώνης λόγω ανταλλαγής μηνυμάτων ελέγχου. Αυτό συμβαίνει, καθώς η γνώση της τοπολογίας της ζώνης ενός κόμβου, επιτρέπει την αποτελεσματικότερη προώθηση των πακέτων αναζήτησης διαδρομής σε όλο το δίκτυο. Αυτό επιτυγχάνεται με την χρήση μιας ειδικής υπηρεσίας προώθησης πακέτων, η οποία ονομάζεται “Bordercast Routing Protocol” (BRP) και η οποία επιτρέπει σε έναν κόμβο να στείλει ένα πακέτο, το οποίο απευθύνεται στους κόμβους που βρίσκονται στην άκρη της ζώνης του (ονομάζονται περιφερειακοί κόμβοι). Αυτή η διαδικασία, περιγράφεται πιο αναλυτικά παρακάτω:

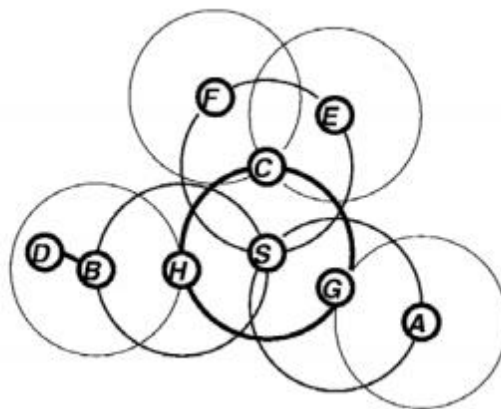
Αρχικά, ο αποστολέας, ο κεντρικός κόμβος της ζώνης που εξετάζουμε, ελέγχει αν ο προορισμός βρίσκεται μέσα στην ζώνη του ή όχι. Ο έλεγχος αυτός, γίνεται με τη χρήση του IARP πρωτοκόλλου και των πληροφοριών που του παρέχει. Στην περίπτωση που ο τελικός προορισμός βρίσκεται μέσα στην ζώνη, τότε η διαδρομή προς αυτόν είναι ήδη γνωστή λόγω του proactive χαρακτήρα του IARP. Αν ο τελικός προορισμός δεν ανήκει στην ζώνη του αποστολέα, τότε ενεργοποιείται ο IERP μηχανισμός. Στην περίπτωση αυτή, ο αποστολέας μεταδίδει ένα Route Request (RREQ) πακέτο σε όλους τους περιφερειακούς του κόμβους με την βοήθεια του BRP μηχανισμού. Ο κάθε περιφερειακός κόμβος με την σειρά του, εξετάζει αν ο τελικός προορισμός βρίσκεται μέσα στην ζώνη του.



Αν όντως συμβαίνει αυτό, τότε μεταδίδει μία θετική απόκριση πίσω στον αρχικό αποστολέα, υποδηλώνοντας την διαδρομή που θα πρέπει να χρησιμοποιηθεί (είναι εφικτή η χρήση είτε hop-by-hop δρομολόγησης είτε source δρομολόγησης). Αν ο τελικός προορισμός δεν βρίσκεται μέσα στην ζώνη των περιφερειακών κόμβων, τότε αυτοί με την σειρά τους χρησιμοποιούν το BRP για να προωθήσουν το πακέτο αναζήτησης της διαδρομής, στους δικούς τους περιφερειακούς κόμβους. Αυτή η διαδικασία συνεχίζεται μέχρι να βρεθεί ο τελικός προορισμός.

Η παραπάνω διαδικασία, περιγράφεται στο παρακάτω σχήμα, όπου ο αρχικός αποστολέας S επιθυμεί να μεταδώσει ένα πακέτο στο τελικό προορισμό D. Αρχικά ο S ελέγχει εάν ο D βρίσκεται μέσα στην ζώνη του. Διαπιστώνει ότι δεν βρίσκεται και τότε στέλνει ένα RREQ στους περιφερειακούς του κόμβους C,G και H. Με την ίδια διαδικασία καθένας από αυτούς τους κόμβους αφού διαπιστώσει ότι ο τελικός προορισμός D δεν βρίσκεται μέσα στην ζώνη του, προωθεί το πακέτο αναζήτησης διαδρομής στους δικούς του περιφερειακούς κόμβους.

Συγκεκριμένα, ο H το προωθεί στον B, ο G στον A και ο C στους F και E. Τελικά ο B διαπιστώνει ότι ο D βρίσκεται μέσα στην ζώνη δρομολόγησης του και αποκρίνεται θετικά στο πακέτο αναζήτησης διαδρομής, δημιουργώντας με αυτόν τον τρόπο την διαδρομή S-H-B-D.



Εικόνα 5. Παράδειγμα λειτουργίας του zone routing protocol





Όπως γίνεται αντιληπτό και από το προηγούμενο σχήμα, οι ζώνες των κόμβων επικαλύπτονται σε μεγάλο βαθμό. Είναι πολύ πιθανό, λοιπόν, ένα πακέτο αναζήτησης διαδρομής, τελικά να πλημμυρίσει το δίκτυο. Ακόμη πιο αποθαρρυντικό, είναι το γεγονός ότι η χρήση του συνδυασμού IERP/BRP μπορεί να οδηγήσει σε μεγαλύτερη επιβάρυνση στο διαθέσιμο εύρος ζώνης, από ότι η τεχνική του πλημμυρίσματος. Για να αποφευχθεί αυτό, χρησιμοποιούνται τεχνικές ανίχνευσης των RREQs, έτσι ώστε να στέλνεται το πακέτο αναζήτησης διαδρομής μόνο προς εκείνη την κατεύθυνση που είναι απαραίτητο.

## ***Κεφάλαιο 5<sup>ο</sup>: Αλγόριθμος Ενεργειακής Φόρτισης Σε Δίκτυο Αισθητήρων***

Στο κεφάλαιο αυτό, σχεδιάζουμε και υλοποιούμε τον Αλγόριθμο Ενεργειακής Φόρτισης σε Δίκτυο Αισθητήρων. Παρουσιάζονται το μοντέλο Φόρτισης, τα Αποτελέσματα προηγούμενων μελετών και τα Αποτελέσματα Εφαρμογής και Προσομοίωσης. Η υλοποίηση του έχει πραγματοποιηθεί σε Matlab και παρουσιάζονται τα αντίστοιχα γραφήματα και οι γραφικές παραστάσεις.

### **5.1 Το μοντέλο Φόρτισης**

Το δίκτυο, το οποίο σχεδιάζεται αρχικά στην εργασία αυτή, αποτελείται από ένα σύνολο κόμβων αισθητήρων, χωρίς να υπάρχει η δυνατότητα επαναφόρτισης τους. Στη συνέχεια, στο δίκτυο αυτό προστείνονται φορτιστές, οι οποίοι είναι υπεύθυνοι για την φόρτιση όσων κόμβων το έχουν ανάγκη, ενώ τέλος προσθέτουμε υπερφορτιστες, στους οποίους αναθέτουμε τη φόρτιση των φορτιστών, αλλά όχι των κόμβων.

Η λειτουργία του δικτύου, είναι η εξής: Οι κόμβοι αισθητήρων, εκτελούν τις καθιερωμένες τους διεργασίες, όπως είναι η παρακολούθηση του περιβάλλοντός τους και η παραγωγή δεδομένων. Παράλληλα, σε τακτά χρονικά διαστήματα, υπολογίζουν



δεδομένα, τα οποία αφορούν και τους ίδιους, όπως είναι το επίπεδο ενέργειας που έχει απομείνει στην μπαταρία του κάθε ενός. Σύμφωνα με τα δεδομένα αυτά, υλοποιείται η διαδικασία φόρτισης τους από τους αντίστοιχους φορτιστές.

Η παρούσα εργασία, στόχο έχει να μελετήσει την αποτελεσματικότερη διαχείριση της ενέργειας σε δίκτυα αισθητήρων και βασίζεται σε μια σειρά μελετών, με κύριες τις παρακάτω:

- Efficient, Distributed Coordination of Multiple Mobile, Chargers in Sensor Networks (Adelina Madhja, Sotiris Nikolettseas, Theofanis P. Raptis).
- Hierarchical, Collaborative, Wireless Charging in Sensors Networks (Adelina Madhja, Sotiris Nikolettseas, Theofanis P. Raptis).

Σε αυτές τις εργασίες, γίνεται προσπάθεια επαναφόρτισης των αισθητήρων που χάνουν ενέργεια, μέσω φορτιστών οι οποίοι πηγαίνουν με τη σειρά τους για φόρτιση στον ανάλογο φορτιστή. Πιο συγκεκριμένα, και στις δύο περιπτώσεις έχουμε ένα δίκτυο αισθητήρων, οι οποίοι έχουν γνώση της τοπολογίας και στέλνουν μηνύματα σε ένα Sink χρησιμοποιώντας δρομολόγηση. Η δρομολόγηση γίνεται με το να στέλνει ο κάθε κόμβος τα πακέτα στον πιο κοντινό κόμβο από αυτούς που βρίσκονται δίπλα του και ταυτόχρονα είναι πιο κοντά στον προορισμό. Οι αισθητήρες κανονικά, θα πρέπει να χάνουν ενέργεια τόσο για την ύπαρξη τους, δηλαδή το ότι είναι απλά ενεργοί, όσο και σε κάθε αποστολή ενός πακέτου.

Στην πρώτη εργασία, γίνεται μια προσπάθεια φόρτισης των αισθητήρων, με φορτιστές οι οποίοι ομαδοποιούν τους αισθητήρες και ο κάθε φορτιστής φορτίζει μόνο τους αισθητήρες που ανήκουν στην ομάδα του. Κάθε φορτιστής, επιλέγει να φορτίσει σε κάθε χρονική στιγμή, τον αισθητήρα ο οποίος είναι στην ομάδα του, έχει χάσει την μεγαλύτερη ενέργεια και είναι πιο κοντά σε αυτόν.

Στην δεύτερη εργασία, γίνεται και μια προσπάθεια για φόρτιση των φορτιστών μέσω μεγαλύτερων φορτιστών. Ανάλογα, οι φορτιστές χωρίζονται σε ομάδες και κάθε υπερφορτιστής διαθέτει την δική του ομάδα. Στην συγκεκριμένη εργασία, δεν αναφέρεται



ο τρόπος φόρτισης των αισθητήρων, και αναφέρεται μόνο στον τρόπο με τον οποίο φορτίζονται οι φορτιστές, οι οποίοι χάνουν ενέργεια κατά την φόρτιση ενός αισθητήρα, από την κίνηση τους αλλά και από την απλή λειτουργία τους.

Στην παρούσα μελέτη, συνδυάζουμε τις προτάσεις των δύο εργασιών και με τη χρήση τόσο φορτιστών όσο και υπερφορτιστών και προτείνουμε ένα πρωτόκολλο φόρτισης για ολόκληρη τη λειτουργία ενός δικτύου αυτής της μορφής. Εφαρμόσαμε απλή δρομολόγηση σε ένα απλό δίκτυο χωρίς φορτιστές, με φορτιστές σύμφωνα με το πρωτόκολλο Distributed Coordination Local Knowledge protocol (DCLK) και στην συνέχεια σε δίκτυο με υπερφορτιστές, σύμφωνα με το πρωτόκολλο 1 Level Knowledge Distributed Coordinates (1KDC) για τους υπερφορτιστές, ενώ οι φορτιστές κρατούν το πρωτόκολλο DCLK. Τα δύο αυτά πρωτόκολλα περιγράφονται αναλυτικά στην επόμενη ενότητα. Στις τρεις αυτές περιπτώσεις, μετρήσαμε τον χρόνο από την αρχή της λειτουργίας του δικτύου, μέχρι το χρόνο που χάνεται ο πρώτος κόμβος, τον αριθμό των αισθητήρων που μένουν ζωντανοί σε σχέση με τον χρόνο και το Coverage του δικτύου. Η εφαρμογή της παραπάνω μεθόδου, έγινε με τη χρήση Matlab, όπου εξήχθησαν τα αποτελέσματα που παρουσιάζονται στην ενότητα 5.3

## 5.2 Αποτελέσματα προηγούμενων μελετών

Έχουν πραγματοποιηθεί σημαντικές μελέτες, οι οποίες εισάγουν μια σειρά συστημάτων προς μελέτη, παρουσιάζοντας παράλληλα και τα αντίστοιχα αποτελέσματα. Μερικές από αυτές είναι οι ακόλουθες:

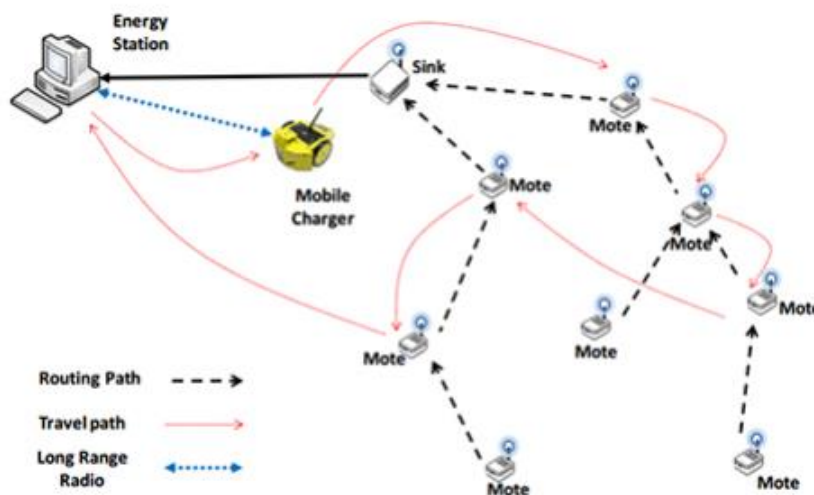
- Prolonging Sensor Network Lifetime Through Wireless Charging
- Efficient Energy Management in Wireless Rechargeable Sensor Networks [4]
- Adaptive, Limited Knowledge Wireless Recharging in Sensor Networks [5]

Στο σημείο αυτό, γίνεται μια αναλυτικότερη παρουσίαση των μελετών αυτών.

### **Prolonging Sensor Network Lifetime Through Wireless Charging**

Το δίκτυο, το οποίο σχεδιάζεται στην εργασία αυτή και φαίνεται στην παρακάτω εικόνα, αποτελείται από:

1. Έναν κινητό ασύρματο φορτιστή.
2. Ένα σύνολο κόμβων-αισθητήρων, οι οποίοι είναι εξοπλισμένοι με δέκτες ασύρματης ενέργειας.
3. Ένα σταθμό ενέργειας, ο οποίος είναι υπεύθυνος για την παρακολούθηση της ενεργειακής κατάστασης των αισθητήρων και ο οποίος αποφασίζει για τον τρόπο με τον οποίο θα επαναφορτιστούν οι αισθητήρες από το φορτιστή.



Εικόνα 6. Απεικόνιση Δικτύου

Το δίκτυο λειτουργεί με τον εξής τρόπο: Οι κόμβοι αισθητήρων εκτελούν τις καθιερωμένες τους διεργασίες, όπως είναι η παρακολούθηση του περιβάλλοντός τους και η παραγωγή δεδομένων, ενώ περιοδικά υπολογίζουν δεδομένα που αφορούν και τους ίδιους, όπως είναι το επίπεδο ενέργειας το οποίο έχει απομείνει στην μπαταρία τους. Όλες αυτές τις πληροφορίες τις αποστέλλουν στο κέντρο ελέγχου. Ωστόσο, μόνο οι πληροφορίες που αφορούν την ενέργεια των  $k$  κόμβων με τη λιγότερη ενέργεια προωθούνται στο σταθμό ενέργειας. Τελικά, ο σταθμός ενέργειας είναι αυτός που



αποφασίζει για το πως πρέπει να φορτιστούν αυτοί οι κόμβοι, τρέχοντας τον αλγόριθμο φόρτισης, και δίνει τις απαραίτητες εντολές στο φορτιστή.

Ο φορτιστής από τη μεριά του, εκτελεί τις εντολές, τις οποίες λαμβάνει και όταν λάβει εκ νέου εντολές, προσαρμόζει αντίστοιχα τη διαδικασία φόρτισης. Όπως λοιπόν αναφέρθηκε, ο σταθμός ενέργειας, τρέχει έναν αλγόριθμο φόρτισης για να προγραμματίσει τη διαδικασία φόρτισης, την οποία θα πραγματοποιήσει ο φορτιστής. Ο αλγόριθμος αυτός, αποφασίζει τόσο για την ακολουθία των κόμβων που θα φορτιστούν, όσο και για το ποσοστό της ενέργειας που θα διατεθεί σε κάθε κόμβο.

Στόχος της συγκεκριμένης εργασίας, είναι να βρεθεί η καλύτερη ακολουθία φόρτισης, ώστε ο χρόνος ζωής του δικτύου, δηλαδή ο χρόνος μέχρι να ξεμείνει από ενέργεια έστω και ένας κόμβος, να μεγιστοποιηθεί. Ένα πρόβλημα το οποίο αποδεικνύεται ότι είναι NP-Complete.

Η πιο απλοϊκή λύση για τον αλγόριθμο φόρτισης, θα ήταν προφανώς να φορτίζεται κάθε φορά εκείνος ο κόμβος αισθητήρας, ο οποίος έχει το χαμηλότερο επίπεδο ενέργειας. Ωστόσο, αυτό θα οδηγούσε το φορτιστή να κάνει μεγάλες διαδρομές και να αναγκάζεται να πηγαίνει μεταξύ των κόμβων, γεγονός το οποίο θα προσέθετε μεγάλο κόστος μετακίνησης του φορτιστή. Για το λόγο αυτό, προτάθηκε ο αλγόριθμος απληστίας ο οποίος έχει τα εξής δύο βήματα:

Βήμα 1. Ταξινόμησε όλους τους κόμβους αισθητήρες κατά αύξουσα σειρά, με κριτήρια το χρόνο ζωής που απομένει στον κάθε κόμβο.

Βήμα 2. Ξεκινώντας από τον πρώτο στην ταξινόμηση κόμβο, έλεγξε αν μπορείς να αυξήσεις το χρόνο ζωής του δικτύου στο χρόνο ζωής του επόμενου κάθε φορά κόμβου. Αν αυτό είναι εφικτό, σημαίνει ότι μπορεί να βρεθεί κατάλληλη ακολουθία φόρτισης. Συνεπώς, δοκίμασε αν μπορείς να κάνεις το ίδιο και με τον επόμενο στην σειρά κόμβο. Αν δε γίνεται, σταμάτα εκεί. Μόλις τελειώσει η παραπάνω διαδικασία, επιλέγεται εκείνη η ακολουθία φόρτισης η οποία μπορεί να επεκτείνει όσο παραπάνω γίνεται το χρόνο ζωής του δικτύου.



Αξίζει να αναφερθεί ότι ο αλγόριθμος αυτός είναι πολύ εύκολος να ενσωματωθεί και είναι αποτελεσματικός στην πράξη. Ένα όμως μειονέκτημα του αλγορίθμου είναι ότι μόλις ξεκινήσει να φορτίζει έναν κόμβο, τότε θα συνεχίζει να φορτίζει τον ίδιο κόμβο μέχρι να φτάσει στο χρόνο ζωής στόχο, χωρίς να υπολογίζει τι θα γίνει και με τους υπόλοιπους κόμβους προς φόρτιση. Αυτό το γεγονός, κάτω από συγκεκριμένες περιστάσεις μπορεί να υποβαθμίσει το δίκτυο και να μειώσει το χρόνο ζωής του.

Έχοντας υπ' όψιν αυτή την παρατήρηση, προτάθηκε ένας λιγότερο άπληστος και πιο ισορροπημένος αλγόριθμος, ο GreedyPlus. Ο συγκεκριμένος αλγόριθμος, εφαρμόζει δυαδική αναζήτηση για να βρει έναν πιο κατάλληλο στόχο για το χρόνο ζωής, όπου θα μπορούν να τον επιτύχουν όλοι οι κόμβοι.

Στα πειράματα αξιολόγησης που πραγματοποιήθηκαν, αποδείχθηκε ότι σε ένα δίκτυο αισθητήρων, όπου κάθε κόμβος ξεκινά με διαφορετικό επίπεδο ενέργειας και έχει διαφορετική κατανάλωση, το προτεινόμενο σχήμα φόρτισης μπορεί να βελτιώσει το χρόνο ζωής του δικτύου, αφού πρώτα εντοπιστούν οι κόμβοι με τα χαμηλότερα επίπεδα ενέργειας. Οι αντίστοιχες παρατηρήσεις από τα αποτελέσματα της εξομοίωσης έδειξαν ότι:

- Η ασύρματη φόρτιση είναι αποτελεσματική στο να παρατείνει το χρόνο ζωής του δικτύου.
- Αν γίνει προσεκτικός σχεδιασμός των κινήσεων του φορτιστή, θα καταναλώνεται λιγότερη ενέργεια μετακίνησης, γεγονός που σημαίνει ότι θα απομένει περισσότερη ενέργεια στο φορτιστή για να παρέχει στους κόμβους και άρα θα μεγαλώσει και ο χρόνος ζωής του δικτύου.
- Αποδεικνύεται ότι, όταν ο φορτιστής είναι ισχυρός, δηλαδή όταν μπορεί να μεταφέρει μεγάλο φορτίο ενέργειας, τότε ο αλγόριθμος ο οποίος στοχεύει στη δρομολόγηση με στόχο την αποδοτική φόρτιση είναι πιο αποτελεσματικός. Αντίθετα, όταν ο φορτιστής είναι μικρός, δηλαδή μπορεί να μεταφέρει μικρότερο φορτίο ενέργειας, τότε ο αλγόριθμος που στοχεύει στη δρομολόγηση με στόχο την ισορροπημένη φόρτιση είναι πιο αποδοτικός.

#### **Efficient Energy Management in Wireless Rechargeable Sensor Networks [4]**



Το σύστημα το οποίο περιγράφεται στη συγκεκριμένη εργασία απαρτίζεται από 3 ειδών συσκευές:

- Ένα κινητό ασύρματο φορτιστή.
- Ένα σύνολο στατικών κόμβων αισθητήρων.
- Ένα κέντρο ελέγχου.

Για το μοντέλο γίνονται οι εξής υποθέσεις:

- Οι στατικοί κόμβοι βρίσκονται σε μία κυκλική περιοχή με ακτίνα  $R$ .
- Το κέντρο ελέγχου βρίσκεται στο κέντρο της κυκλικής περιοχής.
- Ο φορτιστής δεν πραγματοποιεί καμία ενέργεια για συγκέντρωση δεδομένων
- Ο φορτιστής μπορεί να φορτίσει έναν κόμβο κάθε φορά και αφού έχει πλησιάσει τον κόμβο αισθητήρα σε πολύ κοντινή απόσταση, έτσι ώστε η διαδικασία της φόρτισης να έχει τη μέγιστη αποτελεσματικότητα.
- Ο χρόνος που κάνει ο φορτιστής για να μετακινηθεί από τον έναν κόμβο στον άλλο, είναι κατά πολύ μικρότερος του χρόνου τον οποίο χρειάζεται για να φορτίσει έναν κόμβο.
- Ο χρόνος φόρτισης, είναι ίσος για κάθε κόμβο και είναι ανεξάρτητος από το επίπεδο ενέργειας της μπαταρίας του.

Το πρόβλημα που απασχολεί τους συγγραφείς και αυτής της εργασίας, είναι η επιλογή της καλύτερης δυνατής τροχιάς για το φορτιστή, ώστε να μη χαθεί κανένα μήνυμα εξαιτίας χαμηλής ενέργειας. Το πρόβλημα αυτό, αποδεικνύεται ότι είναι NP-Complete. Προκειμένου να δοθεί μία λύση, θα πρέπει πρώτα να δοθούν κάποιες απαντήσεις σε 2 πολύ κρίσιμους παράγοντες. Ένας από αυτούς τους παράγοντες, έχει να κάνει με το ποσοστό ενέργειας, το οποίο θα δοθεί αρχικά στο φορτιστή. Η αιτία για αυτό είναι ότι υποθέτουμε ότι η συνολική ενέργεια που διατίθεται στο σύστημα είναι σταθερή και ένα ποσοστό από αυτή πάει στους κόμβους αισθητήρων, ενώ το υπόλοιπο ποσοστό διατίθεται στο φορτιστή. Ακόμα, θα πρέπει να αποφασιστεί, το πόσο θα είναι αυτό το ποσοστό, καθώς μεγαλύτερο ποσοστό ενέργειας στο φορτιστή, σημαίνει ότι οι κόμβοι δεν θα είναι πλήρως φορτισμένοι. Ο δεύτερος παράγοντας έχει να κάνει με το ποσοστό φόρτισης κάθε κόμβου, αν δηλαδή ο κόμβος θα φορτίζεται πλήρως ή απλά κατά ένα ποσοστό, το οποίο απλά θα τον βγάζει από την κρίσιμη, λόγω χαμηλής ενέργειας, κατάσταση.



Στις πειραματικές αξιολογήσεις, χρησιμοποιούνται 5 διαφορετικές τεχνικές διάσχισης του δικτύου. Οι τεχνικές αυτές αναλύονται παρακάτω:

- Διάσχιση με καθολική γνώση του δικτύου: Ο φορτιστής, έχει πλήρη γνώση της εικόνας του δικτύου και σε κάθε του βήμα υπολογίζει ποιον κόμβο θα πρέπει να φορτίσει με βάση ένα σύνολο παραγόντων, στους οποίους περιλαμβάνονται η τρέχουσα ενέργεια, η αρχική ενέργεια και η απόσταση από το φορτιστή για κάθε κόμβο.
- Σπειροειδής διάσχιση: Ο φορτιστής, ξεκινώντας από το κέντρο ελέγχου, το οποίο βρίσκεται στο κέντρο του δικτύου, διασχίζει ένα μονοπάτι που σχηματίζει ένα σύνολο ομόκεντρων κύκλων, με κέντρο το κέντρο ελέγχου αλλά με αυξανόμενη ακτίνα. Μόλις ο φορτιστής φτάσει στην άκρη του δικτύου, ακολουθεί το ίδιο μονοπάτι με την αντίθετη φορά.
- Διαμετρική διάσχιση: Ο φορτιστής ξεκινάει πάλι από το κέντρο ελέγχου και επιλέγει μία τυχαία κατεύθυνση με σκοπό να φτάσει στην περίμετρο του δικτύου. Κινείται πάνω σε αυτή τη διάμετρο και στη συνέχεια διανύει ένα κομμάτι της περιμέτρου. Στη συνέχεια επιλέγει εκ νέου μία νέα διάμετρο μέχρι να φτάσει στο άλλο άκρο του δικτύου. Επαναλαμβάνει την ίδια διαδικασία μέχρι να ξεμείνει από ενέργεια.
- Τυχαίος περίπατος: Κάθε κίνηση του φορτιστή γίνεται κατά τυχαίο τρόπο και είναι στοχαστικά ανεξάρτητη από προηγούμενες κινήσεις.
- Προσαρμοζόμενη κυκλική διάσχιση: Ο αλγόριθμος για τη διάσχιση αυτή, προτείνεται από τους συγγραφείς της εργασίας και αυτό που συμβαίνει είναι ο φορτιστής να ακολουθεί κυκλική τροχιά γύρω από το κέντρο ελέγχου. Η ακτίνα της τροχιάς διαφέρει κάθε φορά και προσαρμόζεται στην ενεργειακή εξάντληση κάθε υποπεριοχής του δικτύου. Ο φορτιστής, φορτίζει τους αισθητήρες που βρίσκονται μέσα στο δακτύλιο, τον οποίο οριοθετεί η κάθε τροχιά.

Κατά τη διάρκεια διεξαγωγής των πειραμάτων και προκειμένου να γίνει η αξιολόγηση των αποτελεσμάτων χρησιμοποιήθηκαν οι παρακάτω μετρικές:

- Ο αριθμός των κόμβων, οι οποίοι είχαν κάθε στιγμή αρκετή ενέργεια για να λειτουργήσουν κατά τη διάρκεια της πειραματικής διαδικασίας.
- Ο ενεργειακός χάρτης του δικτύου, δηλαδή η απεικόνιση ολόκληρου του δικτύου με γεωγραφικούς όρους, μετά την πραγματοποίηση ορισμένων συμβάντων.



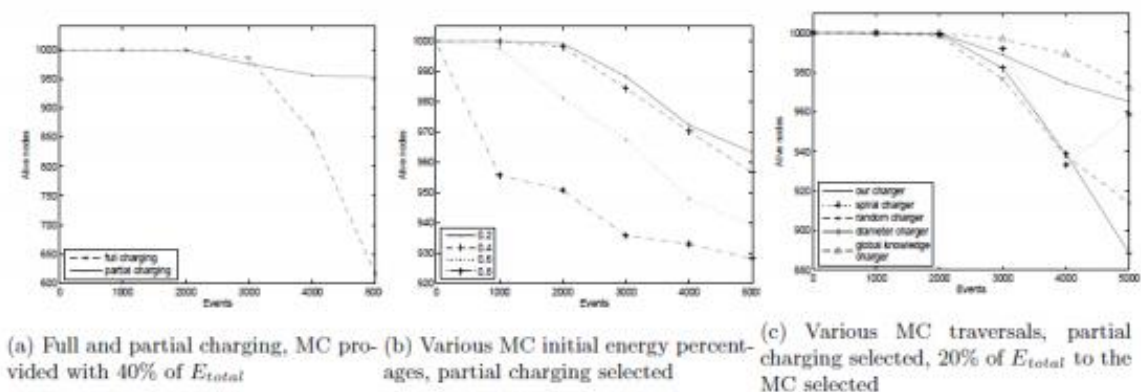


- Ο βαθμός κάλυψης, δηλαδή το μέσο αριθμό των κόμβων, οι οποίοι έχουν το φορτιστή στην ακτίνα τους.
- Η συνδεσιμότητα κάθε χρονική στιγμή που μετριέται με το μέσο όρο για όλους τους κόμβους.

Αρχικά, πραγματοποιήθηκαν πειράματα για να διαπιστωθεί ο βαθμός στον οποίο πρέπει να φορτίζεται κάθε κόμβος, αν δηλαδή θα πραγματοποιείται μερική ή ολική φόρτιση. Αυτό που αποδείχθηκε, είναι ότι η μερική φόρτιση έχει καλύτερα αποτελέσματα και αυτό είναι απόρροια του γεγονότος ότι στην περίπτωση της ολικής φόρτισης, ο φορτιστής ξοδεύει πολύ ενέργεια και άρα η ενέργειά του καταναλώνεται πολύ πιο γρήγορα.

Στη συνέχεια, πραγματοποιήθηκαν πειράματα για να βρεθεί το ποσοστό της ενέργειας, η οποία θα πρέπει αρχικά να δοθεί στο φορτιστή, γεγονός που όπως έχει ήδη αναφερθεί συνδέεται άμεσα με το ποσοστό ενέργειας το οποίο θα δοθεί στους κόμβους. Αυτό που παρατηρήθηκε, είναι ότι στο φορτιστή δε θα πρέπει να δίνεται ποσοστό πάνω από 40%, καθώς αυτό επηρεάζει αρνητικά το χρόνο ζωής του δικτύου, ανεξάρτητα από τον αλγόριθμο δρομολόγησης που μπορεί να ακολουθηθεί.

Στην παρακάτω εικόνα, φαίνεται με τις διαφορετικές τεχνικές ο αριθμός των κόμβων, οι οποίοι μένουν ενεργοί μετά από την εφαρμογή του αλγόριθμου στη συγκεκριμένη εργασία.



Εικόνα 7. Αποτελέσματα εφαρμογής του αλγορίθμου (A) [4]



Επιπλέον, έγιναν πειράματα ώστε να συγκριθούν οι διαφορετικές τεχνικές διάσχισης οι οποίες αναλύθηκαν. Για να γίνουν τα πειράματα αυτά, χρησιμοποιήθηκαν προφανώς τα αποτελέσματα από τις 2 προηγούμενες πειραματικές διαδικασίες και αποφασίστηκε να εφαρμοστεί η τεχνική, όπου ο φορτιστής παίρνει το 20% της αρχικής ενέργειας, ενώ κάθε φορά θα φορτίζει μερικώς τον αντίστοιχο κόμβο. Παράλληλα διεξήχθησαν πειράματα, όπου εφαρμόστηκαν 3 διαφορετικά πρωτόκολλα δρομολόγησης, ώστε να φανεί η επίδραση τους στην απόδοση του δικτύου. Σαν τελικό συμπέρασμα προέκυψε, ότι η καλύτερη στρατηγική για το δίκτυο που περιγράφηκε, περιλαμβάνει τη μερική επαναφόρτιση του δικτύου, την απόδοση ενός μέτριου ποσοστού ενέργειας στο φορτιστή σε σχέση με το υπόλοιπο δίκτυο και ο φορτιστής να ακολουθεί κυκλική τροχιά γύρω από το κέντρο ελέγχου, με ακτίνα η οποία προσαρμόζεται, με βάση τις διαφοροποιήσεις της ενέργειας σε γεωγραφικό επίπεδο.

### **Adaptive, Limited Knowledge Wireless Recharging in Sensor Networks [5]**

Το σύστημα, το οποίο υλοποιείται στη συγκεκριμένη εργασία απαρτίζεται από 3 ειδών συσκευές:

- Ένα κινητό ασύρματο φορτιστή.
- Ένα σύνολο στατικών κόμβων αισθητήρων.
- Ένα κέντρο ελέγχου.

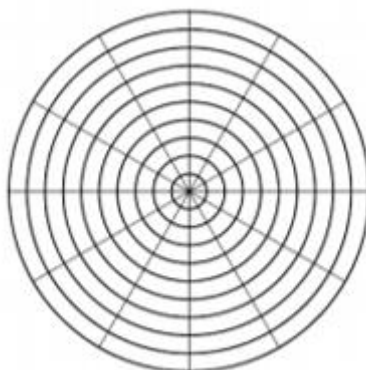
Για το μοντέλο, το οποίο αναλύεται γίνονται οι εξής υποθέσεις:

- Οι κόμβοι και το κέντρο ελέγχου είναι στατικοί.
- Η περιοχή ανάπτυξης του δικτύου θεωρείται ότι είναι κυκλική ακτίνας  $D$ .
- Το δίκτυο, χωρίζεται σε ομόκεντρους δακτυλίους, οι οποίοι διαφέρουν ως προς την ακτίνα τους και τον αριθμό τεμαχίων που περιέχουν. Οι τομές μεταξύ τους, είναι αυτές που καθορίζουν τους επιμέρους τομείς του δικτύου. Για παράδειγμα, το δίκτυο στην εικόνα που ακολουθεί, αποτελείται από 10 ομόκεντρους κύκλους και 12 τεμάχια, με αποτέλεσμα να σχηματίζονται συνολικά 120 τομείς.



- Οι κόμβοι αισθητήρες, τοποθετούνται τυχαία μέσα στο δίκτυο και κάθε κόμβος έχει ένα μοναδικό ID, γνωρίζει την τοποθεσία του και προφανώς μπορεί να ανήκει μόνο σε έναν τομέα.
- Ο φορτιστής διασχίζει το δίκτυο και φορτίζει τους κόμβους, όταν έρθει αρκετά κοντά σε αυτούς.
- Όπως και στην προηγούμενη εργασία, η συνολική ενέργεια του δικτύου προκύπτει από την ενέργεια με την οποία ξεκινάει ο φορτιστής και με τη συνολική ενέργεια, η οποία αποδίδεται στους κόμβους.

Στο παρακάτω σχήμα φαίνεται ο χωρισμός στη δομή του δικτύου, όπως παρουσιάζεται στην εργασία.



**Εικόνα 8. Δίκτυο χωρισμένο σε τμήματα με ετερογενείς πυκνότητες [5]**

Στόχος της εργασίας αυτής, είναι να δοθεί λύση στο πρόβλημα της εξεύρεσης της καλύτερης δυνατής τροχιάς, την οποία θα ακολουθεί ο φορτιστής, ώστε να αυξηθεί ο χρόνος ζωής του δικτύου. Εισάγεται λοιπόν για πρώτη φορά μία καινούρια παράμετρος, η οποία αφορά τους κόμβους και έχει να κάνει με το πόσο σημαντικός είναι ο κάθε ένας από αυτούς. Η παράμετρος αυτή, επηρεάζεται τόσο από την κίνηση, δηλαδή τον όγκο δεδομένων που εξυπηρετεί ο κόμβος, όσο και από την ενέργεια που καταναλώνει. Για παράδειγμα, ένας κόμβος ο οποίος έχει υψηλή κίνηση και έχει καταναλώσει μεγάλο



ποσοστό ενέργειας, θεωρείται ότι είναι κρίσιμος και θα πρέπει να του δοθεί υψηλή προτεραιότητα από το φορτιστή.

Όσον αφορά το ποσοστό ενέργειας, το οποίο θα δίνει ο φορτιστής σε κάθε κόμβο, η πιο λογική επιλογή, θα ήταν να φορτίζει τον κόμβο αναλογικά με την ενέργεια η οποία λείπει από τον κόμβο. Αυτή η πολιτική όμως, αγνοεί το γεγονός ότι υπάρχουν κόμβοι, οι οποίοι είναι πιο σημαντικοί για το δίκτυο εξαιτίας της τοποθεσίας τους, του ρυθμού παραγωγής τους, του ειδικού ρόλου που μπορεί να έχουν για το δίκτυο κ.α. Για το λόγο αυτό, στην εργασία αυτή, οι συγγραφείς θεωρούν υψηλής σημασίας το πόσο σημαντικός είναι ένας κόμβος για να καταλήξουν στο ποσοστό στο οποίο θα πρέπει να φορτιστεί.

Ακόμα, στη συγκεκριμένη εργασία, παρουσιάζονται 3 πρωτόκολλα, τα οποία αφορούν το πως θα πρέπει να διατρέχει ο φορτιστής το δίκτυο. Καθένα από αυτά έχει διαφορετικές υποθέσεις όσον αφορά το επίπεδο γνώσης του δικτύου. Το επίπεδο αυτό κυμαίνεται από πλήρης γνώση μέχρι περιορισμένη. Τα πρωτόκολλα αυτά είναι τα ακόλουθα:

#### *1. Πρωτόκολλο καθολικής γνώσης του δικτύου*

Στο πρωτόκολλο αυτό, ο φορτιστής έχει πλήρη γνώση της δομής και της κατάστασης του δικτύου και αποφασίζει ποιον κόμβο θα φορτίσει, δίνοντας προτεραιότητα σε κόμβους υψηλής κρισιμότητας, οι οποίοι είναι όμως παράλληλα και σε κοντινή απόσταση με το φορτιστή. Το πρωτόκολλο αυτό, από τη στιγμή που έχει γενική γνώση του δικτύου, προφανώς και θα υπερνικά της υπόλοιπες τεχνικές, ωστόσο δε μπορεί εύκολα να εφαρμοστεί σε ευρείας κλίμακας δίκτυα, καθώς έχει υψηλό επικοινωνιακό κόστος.

#### *2. Πρωτόκολλο περιορισμένης πληροφόρησης*

Σε αυτό το πρωτόκολλο, από κάθε τομέα, επιλέγονται ανά τακτά χρονικά διαστήματα κάποιοι κόμβοι, οι οποίοι λειτουργούν ως αντιπρόσωποι του τομέα αυτού. Ρόλος τους είναι να ενημερώνουν το κέντρο ελέγχου για την κατάσταση τους και για την



κρισιμότητά τους. Μόλις το κέντρο ελέγχου ενημερωθεί από τους αντιπροσώπους κάθε τομέα, δίνει τις απαραίτητες οδηγίες στο φορτιστή. Το αποτέλεσμα είναι το πρωτόκολλο να χρησιμοποιεί γνώση από όλο το δίκτυο, η οποία όμως περιορίζεται σε συγκεκριμένους κόμβους. Κρίσιμο παράγοντα εδώ, αποτελεί το ποσοστό των κόμβων οι οποίοι λειτουργούν ως αντιπρόσωποι. Αν ο αριθμός τους είναι μεγάλος, τότε ναι μεν το κέντρο ελέγχου θα έχει πιο αναλυτική πληροφόρηση για το δίκτυο, αλλά από την άλλη αυξάνεται το κόστος μεταφοράς μηνυμάτων. Αν ο αριθμός των αντιπροσώπων είναι μικρός, τότε το κόστος μηνυμάτων μειώνεται αλλά η αναπαράσταση κάθε τομέα είναι λιγότερο αναλυτική.

### *3. Πρωτόκολλο τροχιάς αντίδρασης*

Σε αυτή τη στρατηγική, κάθε φορά που η ενέργεια κάποιου κόμβου πέσει κάτω από το επιτρεπτό όριο, τότε ο κόμβος παράγει ένα προειδοποιητικό μήνυμα. Το μήνυμα αυτό πολλαπλασιάζεται και μεταφέρεται σε διπλανούς κάθε φορά κόμβους. Το αποτέλεσμα είναι να παράγεται για κάθε κόμβο με χαμηλή ενέργεια μία δενδρική δομή η οποία ξεκινάει από αυτόν τον κόμβο. Έτσι, όταν φορτιστής περάσει από κάποιον κόμβο του δέντρου, λαμβάνει το μήνυμα το οποίο έχει σταλεί από τον κόμβο στη ρίζα του δέντρου. Το δέντρο, το οποίο σχηματίζεται αυξάνεται σταδιακά, με ρυθμό που είναι ανάλογος της κρισιμότητας του κόμβου στη ρίζα του δέντρου.

Προτείνεται λοιπόν, μία στρατηγική για τη διάδοση του μηνύματος, η οποία στοχεύει σε μία σχετικά μεγάλη κάλυψη της περιοχής του δικτύου, ενώ παράλληλα η κατανάλωση ενέργειας εξαιτίας της επικοινωνίας παραμένει χαμηλή. Ο φορτιστής από την άλλη πλευρά, μπορεί να μεταβάλλει την κατάστασή του μεταξύ 2 επιλογών. Μπορεί είτε να μπει σε κατάσταση περιπολίας, είτε να μπει σε κατάσταση φόρτισης. Η εναλλαγή αυτή πραγματοποιείται με τον ακόλουθο τρόπο: Αρχικά, βρίσκεται σε κατάσταση περιπολίας και ακολουθεί σπειροειδή τροχιά με κέντρο το κέντρο ελέγχου, μέχρι να ειδοποιηθεί ότι η περιοχή που διασχίζει αποτελείται από κόμβους με χαμηλή ενέργεια. Τη στιγμή εκείνη,



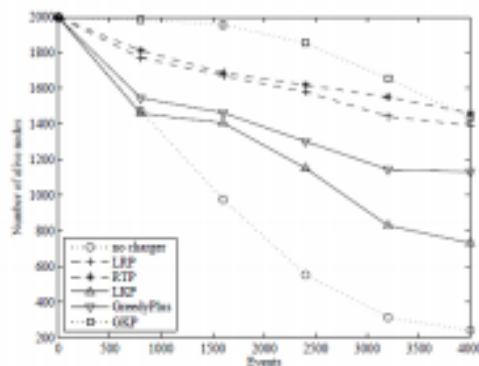
μπαίνει σε κατάσταση φόρτισης και ακολουθεί μία διαφορετική τροχιά με σκοπό να ολοκληρώσει τη διαδικασία φόρτισης στη συγκεκριμένη περιοχή. Αν ο φορτιστής διαπιστώσει ότι στην περιοχή αυτή υπάρχουν περισσότερα από ένα δέντρα ταυτόχρονα, τότε με έναν απλό έλεγχο του βάθους κάθε δέντρου μπορεί να διαπιστώσει ποιος κόμβος είναι πιο κρίσιμος για να φορτιστεί. Μόλις τελειώσει τη διαδικασία φόρτισης, ξαναμπαίνει σε κατάσταση περιπολίας.

Κατά τη διάρκεια διεξαγωγής των πειραμάτων και προκειμένου να γίνει η αξιολόγηση των αποτελεσμάτων, χρησιμοποιήθηκαν οι παρακάτω μετρικές:

- Ο αριθμός των κόμβων, οι οποίοι είχαν κάθε στιγμή αρκετή ενέργεια για να λειτουργήσουν κατά τη διάρκεια της πειραματικής διαδικασίας.
- Ο ενεργειακός χάρτης του δικτύου, δηλαδή η απεικόνιση ολόκληρου του δικτύου με γεωγραφικούς όρους, μετά την πραγματοποίηση ορισμένων συμβάντων.
- Η ευρωστία της δρομολόγησης.
- Η συνδεσιμότητα μεταξύ των κόμβων.

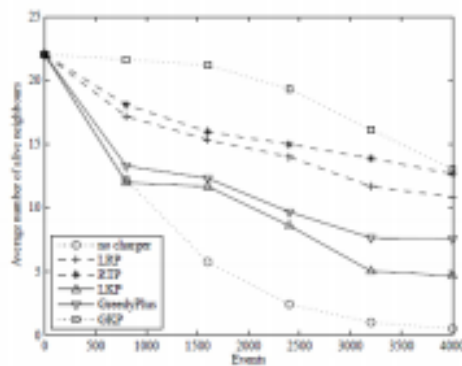
Όπως και στην προηγούμενη εργασία, πραγματοποιήθηκαν πειράματα, για να βρεθεί το ποσοστό της ενέργειας που θα πρέπει αρχικά να δοθεί στο φορτιστή, γεγονός που όπως έχει ήδη αναφερθεί συνδέεται άμεσα με το ποσοστό ενέργειας, το οποίο θα δοθεί στους κόμβους. Η χρυσή τομή που βρέθηκε, είναι να δοθεί στο φορτιστή ενέργεια που ισούται με το 30% της συνολικής ενέργειας του δικτύου.

Εξίσου κρίσιμη παράμετρος, είναι και ο αριθμός των κόμβων, οι οποίοι θα λειτουργούν ως αντιπρόσωποι. Έπειτα από σχετικά πειράματα το ιδανικό ποσοστό των αντιπροσώπων αποφασίστηκε να είναι 5% των συνολικών κόμβων. Τα αποτελέσματα, τα οποία παρουσιάζονται στην εργασία είναι τα ακόλουθα:



(a) Alive nodes over time.

Εικόνα 9. Ενεργοί κόμβοι με το πέρασμα του χρόνου [5]



(b) Average routing robustness.

Εικόνα 10: Routing robustness με το πέρασμα του χρόνου

Τα πειράματα που έγιναν, είχαν ως σκοπό να συγκρίνουν τα προτεινόμενα πρωτόκολλα με 2 άλλα πρωτόκολλα, τα οποία είχαν προταθεί και αναλυθεί νωρίτερα. Αυτά είναι ο αλγόριθμος απληστίας και το απλό πρωτόκολλο περιορισμένης πληροφόρησης, το οποίου δε λαμβάνει υπ' όψιν του το βαθμό κρισιμότητας κάθε κόμβου.

Το συμπέρασμα το οποίο βγήκε, ήταν ότι τα προτεινόμενα πρωτόκολλα πλησιάζουν σε απόδοση το πρωτόκολλο καθολικής γνώσης του δικτύου και εμφανώς υπερέρχουν του



απλού πρωτοκόλλου περιορισμένης πληροφόρησης. Υπεροχή βρέθηκε και στη σύγκριση με τον αλγόριθμο απληστίας GreedyPlus, όπου ο φορτιστής απαιτεί συνεχείς ενημερώσεις, με αποτέλεσμα την αυξημένη κατανάλωση ενέργειας για ανταλλαγή μηνυμάτων.

Μία πολύ ενδιαφέρουσα ιδέα, η οποία έχει προς το παρόν ελάχιστα μελετηθεί από τους ερευνητές, είναι η στρατηγική του να μην υπάρχει μόνο ένας φορτιστής στο δίκτυο αισθητήρων αλλά περισσότεροι. Αυτή η ιδέα λοιπόν, ενέπνευσε και τους συγγραφείς της εργασίας “Efficient, Distributed Coordination of Multiple Mobile Chargers in Sensor Networks” [30] και χρησιμοποιώντας την, προτείνουν 4 διαφορετικά πρωτόκολλα για να λύσουν το ενεργειακό πρόβλημα στο δίκτυο. Το μοντέλο που χρησιμοποιούν αποτελείται από:

- Ένα σύνολο στατικών κόμβων αισθητήρων.
- Ένα κέντρο ελέγχου.
- $K$  φορτιστές, οι οποίοι δεν εκτελούν καμία διεργασία στο κομμάτι της συγκέντρωσης των δεδομένων, και οι οποίοι αρχικά τοποθετούνται στο δίκτυο σε συντεταγμένες:

$$(x, y) = \left( \left( \frac{R}{2} * \cos\left(\frac{p}{K}\right)(2j - 1) \right), \frac{R}{2} * \sin\left(\left(\frac{p}{K}\right)(2j - 1)\right) \right)$$

Στο πρωτόκολλο αυτό γίνονται οι εξής υποθέσεις:

- Το δίκτυο είναι κυκλικό και εκτείνεται σε μία περιοχή ακτίνας  $R$ .
- Το κέντρο ελέγχου βρίσκεται στο κέντρο του δικτύου.
- Υπάρχουν  $k$  φορτιστές, οι οποίοι δεν εκτελούν καμία διεργασία στο κομμάτι της συγκέντρωσης των δεδομένων.
- Η συνολική ενέργεια του δικτύου είναι το άθροισμα της ενέργειας που αποδίδεται αρχικά στους κόμβους αισθητήρες και στους φορτιστές, δηλαδή  $E_{total} = E_{sensors} + EMC(tinit)$
- Κάθε φορτιστής μπορεί να φορτίζει έναν κόμβο κάθε φορά και αφού βρίσκεται σε πολύ κοντινή απόσταση με αυτόν.

Η συνολική διαδικασία φόρτισης χωρίζεται σε 2 φάσεις. Η πρώτη είναι η φάση συντονισμού και η δεύτερη είναι η φάση που διεξάγεται η φόρτιση. Για να εξηγηθεί η φάση συντονισμού, πρέπει πρώτα να αναφερθεί ότι ο ρυθμός που κάθε φορτιστής χάνει την ενέργειά του, δεν είναι ο ίδιος αλλά ποικίλει εξαιτίας του μη ομοιόμορφου ρυθμού με τον οποίο πραγματοποιούνται τα διάφορα γεγονότα στο δίκτυο. Για το λόγο αυτό λοιπόν, στη φάση του συντονισμού, οι διαφορετικοί φορτιστές επικοινωνούν μεταξύ τους με σκοπό να





καταναείμουν τις περιοχές φόρτισης δίκαια. Για παράδειγμα, αν ένας φορτιστής είναι αδύναμος, τότε θα πρέπει να αναλάβει μία μικρότερη περιοχή.

Η διαδικασία αυτή μπορεί να γίνει είτε με κεντρικό είτε με καταναεμημένο τρόπο. Στην πρώτη περίπτωση, ο συντονισμός επιτυγχάνεται με χρήση πληροφοριών που συγκεντρώνονται από όλους τους φορτιστές, ενώ στη δεύτερη περίπτωση ένας φορτιστής ενημερώνεται για την κατάσταση των γειτονικών φορτιστών και αυτός είναι υπεύθυνος για το συντονισμό με τους υπολοίπους.

Για τη φάση της διεξαγωγής της φόρτισης και στην περίπτωση που υπάρχει ένας μόνο φορτιστής, έχουν αναφερθεί ήδη κάποιες από τις πολιτικές που ακολουθούνται. Στην περίπτωση της συγκεκριμένης εργασίας, όπου υπάρχουν περισσότεροι φορτιστές, πρέπει να ακολουθηθεί διαφορετική στρατηγική. Για το λόγο αυτό, διακρίνονται 4 διαφορετικές περιπτώσεις, ως προς το ποσοστό γνώσης που υπάρχει για το δίκτυο. Έτσι υπάρχει η καθολική γνώση, όπου υποθέτουμε ότι κάθε φορτιστής μπορεί να χρησιμοποιήσει πληροφορία από όλο το δίκτυο, η τοπική γνώση όπου κάθε φορτιστής γνωρίζει περιορισμένη πληροφορία που αφορά μόνο τη γειτονική του περιοχή, η περίπτωση όπου ο φορτιστής δεν έχει καθόλου πληροφορία για το δίκτυο και τέλος η γνώση αντίδρασης, μία ειδική περίπτωση της καθολικής γνώσης, όπου ξοδεύεται ενέργεια για ανταλλαγή μηνυμάτων πληροφόρησης.

Τα 3 σημαντικότερα πρωτόκολλα που εφαρμόζονται στην περίπτωση που εξετάζουμε είναι:

### **Πρωτόκολλο κεντρικού συντονισμού (CC)**

Πραγματοποιεί κεντρικό συντονισμό, δηλαδή μπορεί να χρησιμοποιήσει πληροφορία από όλους τους φορτιστές, αλλά δε γνωρίζει τι συμβαίνει στο δίκτυο και στους κόμβους αισθητήρες. Κατά τη διάρκεια της φάσης συντονισμού, ανατίθεται σε κάθε φορτιστή μία περιοχή ανάλογα με την τρέχουσα ενεργειακή του κατάσταση και η οποία καθορίζεται από τη γωνία την οποία καλύπτει. Συγκεκριμένα χρησιμοποιείται η παρακάτω συνάρτηση:

$$\phi_j = 2\pi * \frac{E_j}{\sum_{j=1}^K E_j}$$



Όπου:

$$\sum_{j=1}^K \phi = 2\pi$$

Κατά τη διάρκεια της φάσης φόρτισης, ο φορτιστής διατρέχει την περιοχή του ξεκινώντας από το κέντρο ελέγχου μέχρι να φτάσει στα όρια της περιοχής του και κατά τη διάρκεια της διάσχισης φορτίζει τους κόμβους που χρειάζονται φόρτιση, μέχρι να ξεμείνει από ενέργεια.

### Πρωτόκολλο κατανεμημένου συντονισμού (DC)

Στη φάση συντονισμού, του πρωτοκόλλου DC, η περιοχή του δικτύου χωρίζεται σε τομείς και κάθε φορτιστής αναλαμβάνει έναν τομέα, ανάλογα με τη δυναμική του. Κάθε φορτιστής μπορεί να μετακινεί τα όρια της περιοχής του είτε προς τα δεξιά είτε προς τα αριστερά και έτσι μπορεί να μεγαλώνει ή να συρρικνώνει την περιοχή του. Με τον τρόπο αυτό τα όρια κάθε τομέα αποφασίζονται από κοινού μεταξύ των 2 γειτονικών φορτιστών. Η αλλαγή των ορίων της περιοχής ενός φορτιστή  $j$  ορίζεται ως η αλλαγή της γωνίας  $j$ . Για την εύρεση των ορίων της περιοχής ενός φορτιστή  $j$  χρησιμοποιούνται 2 κρίσιμες παράμετροι, η τρέχουσα ενέργεια του φορτιστή  $E_j$  και ο ρυθμός κατανάλωσης ενέργειας του φορτιστή από τον τελευταίο συντονισμό  $\rho_j$ . Η αλλαγή των ορίων της περιοχής για κάθε φορτιστή καθορίζεται από τους παρακάτω αλγόριθμους:

Για την αριστερή πλευρά θα είναι:

$$\begin{aligned} &\text{if } \min\{E_j, E_{j-1}\} = E_j \text{ then} \\ &\quad \Delta\phi_j^l = -\phi_j \cdot \frac{|\rho_{j-1} - \rho_j|}{\max\{\rho_{j-1}, \rho_j\}} \\ &\text{else} \\ &\quad \Delta\phi_j^l = \phi_{j-1} \cdot \frac{|\rho_{j-1} - \rho_j|}{\max\{\rho_{j-1}, \rho_j\}} \\ &\text{end if} \end{aligned}$$

Ενώ για την δεξιά:



```
if  $\min\{E_j, E_{j+1}\} = E_j$  then  
     $\Delta\phi_j^r = -\phi_j \cdot \frac{|\rho_j - \rho_{j+1}|}{\max\{\rho_j, \rho_{j+1}\}}$   
else  
     $\Delta\phi_j^r = \phi_{j+1} \cdot \frac{|\rho_j - \rho_{j+1}|}{\max\{\rho_j, \rho_{j+1}\}}$   
end if
```

Η φάση φόρτισης είναι ακριβώς η ίδια όπως περιγράφεται στο προηγούμενο πρωτόκολλο.

### Πρωτόκολλο καταναμημένου συντονισμού με τοπική γνώση (DCLK)

Τέλος, στο Πρωτόκολλο καταναμημένου συντονισμού με τοπική γνώση (DCLK), η φάση συντονισμού είναι ακριβώς η ίδια όπως είναι στο πρωτόκολλο καταναμημένου συντονισμού. Αυτό που διαφοροποιείται είναι η φάση φόρτισης. Συγκεκριμένα, αυτό που συμβαίνει είναι ότι κάθε τομέας που ανήκει σε έναν φορτιστή, διαχωρίζεται σε επιμέρους τομείς ίσων διαστάσεων. Ο φορτιστής δίνει προτεραιότητα στον τομέα που περικλύει τους περισσότερους κρίσιμους κόμβους, δηλαδή αυτούς που έχουν τη λιγότερη υπολειπόμενη ενέργεια. Συγκεκριμένα ο φορτιστής φορτίζει τον τομέα  $S_{jk}$  που μεγιστοποιεί το γινόμενο:

$$\max\{N(S_{jk}) * (E_{sensor}^{max} - E_{jk}^{min})\}$$

Όπου  $E_{\min,jk}$  είναι το ποσό της ενέργειας του κόμβου με τη χαμηλότερη ενέργεια, στον τομέα

$$S_{jk}, E_{jk}^{min+\Delta} = E_{jk}^{min} + \delta * \frac{E_{sensor}^{max}}{E_{jk}^{min}}, \delta \in (0, 1) \text{ και } N(S_{jk}) = \sum_{e=E_{jk}^{min}}^{E_{jk}^{min+\Delta}} N(e).$$

Όπως είδαμε, τα παραπάνω πρωτόκολλα αφορούν τη φόρτιση των αισθητήρων. Τι γίνεται όμως για την περίπτωση της φόρτισης των φορτιστών;

Το πρωτόκολλο που επιλέξαμε για την φόρτιση των φορτιστών από άλλους υπερφορτιστές, είναι το 1 Level Knowledge Distributed Coordinates (1KDC), όπως αναφέρθηκε και παραπάνω. Σύμφωνα με το συγκεκριμένο πρωτόκολλο, πραγματοποιείται μια καταναμημένη συνεργασία μεταξύ των υπερφορτιστών χρησιμοποιώντας πληροφορίες σχετικές μόνο με την ενεργειακή στάθμη των φορτιστών και όχι των κόμβων. Στη φάση της



συνεργασίας, υποθέτουμε ότι ο υπερφορτιστής γνωρίζει ποιού 2 φορτιστές είναι κοντά του και στα όρια της περιοχής του. Αν  $k$  είναι ο συγκεκριμένος υπερφορτιστής, θεωρούμε «προηγούμενο» τον τελευταίο φορτιστή που ανήκει στην περιοχή του  $k-1$  υπερφορτιστή και «επόμενο» τον πρώτο φορτιστή που ανήκει στην περιοχή του υπερφορτιστή  $k+1$ . Ο υπερφορτιστής  $k$ , για να συνεργαστεί με τους γειτονικούς του,  $k-1$  και  $k+1$ , υπολογίζει ποιός από τους δύο έχει περισσότερα αποθέματα ενέργειας για να φορτίσει τους φορτιστές της ομάδας του αλλά και έναν επιπλέον φορτιστή από την δεξιά ή αριστερή γειτονική του περιοχή. Έτσι, κάθε υπερφορτιστής υπολογίζει το υπολειπόμενο ποσοστό ενέργειας σύμφωνα με τα παρακάτω:

$$e_k^p = E_{SC_k} - \sum_{j \in C_k} E_{MC_j}^{lack} - E_{MC_{pk}}^{lack}$$

$$e_k^n = E_{SC_k} - \sum_{j \in C_k} E_{MC_j}^{lack} - E_{MC_{nk}}^{lack}$$

όπου  $E_{MC_j}^{lack} = E_{MC}^{max} - E_{MC_j}$  είναι το ποσοστό ενέργειας που μπορεί να λάβει ένας φορτιστής μέχρι να θεωρηθεί πλήρως φορτισμένος.

Μεταξύ δύο γειτονικών υπερφορτιστών, εκείνος με το μεγαλύτερο ενεργειακό απόθεμα, προσθέτει τους συνοριακούς φορτιστές της διπλανής ομάδας στην ομάδα του. Έτσι, ο υπερφορτιστής με την λιγότερη ενέργεια, είναι πλέον υπεύθυνος για μια μικρότερη περιοχή. Στην περίπτωση που το ενεργειακό απόθεμα και των δύο φορτιστών είναι στο ίδιο επίπεδο, δεν λαμβάνει χώρα καμία μεταβολή σχετικά με τις περιοχές. Πιο συγκεκριμένα, ο αλγόριθμος συνεργασίας μεταξύ των υπερφορτιστών είναι ο ακόλουθος:

Μεταξύ των υπερφορτιστών  $SC_k$  και  $SC_{k-1}$ :

```

( $SC_k, SC_{k-1}$ )
if ( $e_k^p > e_{k-1}^n$ ) then
     $C_k = C_k \cup \{MC_{pk}\}$ 
     $C_{k-1} = C_{k-1} \setminus \{MC_{pk}\}$ 
else if ( $e_k^p < e_{k-1}^n$ ) then
     $C_{k-1} = C_{k-1} \cup \{MC_{nk-1}\}$ 
     $C_k = C_k \setminus \{MC_{nk-1}\}$ 
end if
    
```

Μεταξύ των υπερφορτιστών  $SC_k$  και  $SC_{k+1}$ :



```
(SCk, SCk+1)  
if (ekn > ek+1p) then  
  Ck = Ck ∪ {MCnk}  
  Ck+1 = Ck+1 \ {MCnk}  
else if (ekn < ek+1p) then  
  Ck+1 = Ck+1 ∪ {MCpk+1}  
  Ck = Ck \ {MCpk+1}  
end if
```

Κάθε υπερφορτιστής, επιλέγει τον φορτιστή που θα φορτίσει ανάλογα με την απόσταση και το ποσοστό ενέργειας του φορτιστή. Δηλαδή, ο επόμενος προς φόρτιση φορτιστής δίνεται από τη σχέση:

$$m = \arg \min_{j \in C_k} \left\{ \left( 1 + \frac{E_{MC_j}}{E_{MC}^{max}} \right) \cdot \left( 1 + \frac{d_{k_j}}{2R} \right) \right\}$$

### 5.3 Αποτελέσματα Εφαρμογής και Προσομοίωσης

Όπως είδαμε στα προηγούμενα πρωτόκολλα γίνεται προσπάθεια επαναφόρτισης των κόμβων αισθητήρων, οι οποίοι χάνουν ενέργεια. Οι αισθητήρες γενικά χάνουν ενέργεια τόσο για την ύπαρξή τους, δηλαδή ότι είναι απλά ενεργοί, όσο και για την αποστολή κάθε πακέτου. Στόχος μας είναι να γίνει μια συγκριτική μελέτη, να εντοπιστούν τα πιθανά προβλήματα που υπάρχουν και να προταθεί ένα γενικότερο πρωτόκολλο φόρτισης, το οποίο θα έχει στόχο την μεγαλύτερη διάρκεια ζωής του δικτύου.

Για να συγκριθούν οι παραπάνω περιπτώσεις θα πραγματοποιηθεί σύγκριση με την απλή λειτουργία των αισθητήρων. Έτσι, αρχικά πρέπει να καθοριστεί ο τρόπος δρομολόγησης, όπου θα πρέπει να είναι ίδιος σε όλες τις περιπτώσεις. Παράλληλα, είναι γνωστό ότι εφόσον υπάρχει φόρτιση σίγουρα θα υπάρχει μεγαλύτερη διάρκεια ζωής στο δίκτυο αφού έτσι κι αλλιώς προσφέρεται περισσότερη ενέργεια. Αυτό έχει πολύ μεγάλη σημασία, καθώς όποιο πρωτόκολλο και να χρησιμοποιηθεί, αφού προσφέρεται μεγαλύτερη ενέργεια το σύστημα θα λειτουργήσει περισσότερο. Για το λόγο αυτό, είναι αναγκαία και η σύγκριση με ένα πρωτόκολλο, που χωρίς ομαδοποίηση, οι φορτιστές θα



φορτίζουν απλά τυχαία τον κοντινότερο αισθητήρα σε αυτούς, και ο οποίος θα έχει τη μικρότερη ενέργεια.

Το ίδιο θα πρέπει να γίνει και με τους υπερφορτιστές. Δηλαδή να γίνει σύγκριση με ένα πρωτόκολλο, σύμφωνα με το οποίο ένας υπερφορτιστής θα φορτίζει τον κοντινότερο του απλά φορτιστή. Πάλι θα θεωρηθεί ο ίδιος τρόπος δρομολόγησης, καθώς επίσης, ότι οι φορτιστές θα λειτουργούν είτε με απλή επιλογή του αισθητήρα φόρτισης είτε με ομαδοποίηση.

Είναι ακόμα σημαντικό, να καθοριστεί μια κανονικοποιημένη κατανάλωση ενέργειας για κάθε περίπτωση, έτσι ώστε να μπορεί εύκολα να προσαρμοστεί σε κάθε δίκτυο αισθητήρων, με ένα απλό πολλαπλασιασμό με την ανάλογη κατανάλωση ενέργειας. Εξάλλου η κατανάλωση ενέργειας γίνεται από:

- Μεταφορά πακέτου
- Κίνηση
- Λειτουργία

Η στρατηγική, η οποία θα ακολουθηθεί για τη διαδικασία της φόρτισης, είναι να γίνεται πάντα προσπάθεια να φορτιστεί πλήρως ο αισθητήρας ή ο φορτιστής.

Τελικά οι περιπτώσεις οι οποίες θα χρησιμοποιηθούν προκειμένου να γίνουν οι συγκρίσεις περιλαμβάνουν:

- Απλό δίκτυο αισθητήρων.
- Δίκτυο Αισθητήρων με απλούς φορτιστές, οι οποίοι φορτίζουν τον κοντινότερο μικρότερο σε ενέργεια αισθητήρα.
- Δίκτυο αισθητήρων με φορτιστές, οι οποίοι ομαδοποιούν τους αισθητήρες και φορτίζουν μόνο τους αισθητήρες στην δική τους ομάδα.
- Δίκτυο αισθητήρων με φορτιστές και υπερφορτιστές, όπου οι φορτιστές φορτίζουν τον κοντινότερο μικρότερο σε ενέργεια αισθητήρα και οι υπερφορτιστές τον κοντινότερο με μικρότερη ενέργεια φορτιστή.



- Δίκτυο αισθητήρων με φορτιστές και υπερφορτιστές, όπου οι φορτιστές ομαδοποιούν και φορτίζουν μόνο αισθητήρες στην ομάδα τους, και οι υπερφορτιστές τον κοντινότερο με μικρότερη ενέργεια φορτιστή
- Δίκτυο αισθητήρων με φορτιστές και υπερφορτιστές, όπου οι φορτιστές φορτίζουν τον κοντινότερο μικρότερο σε ενέργεια αισθητήρα και οι υπερφορτιστές ομαδοποιούν και φορτίζουν φορτιστές της ομάδας τους
- Δίκτυο αισθητήρων με φορτιστές και υπερφορτιστές, όπου οι φορτιστές ομαδοποιούν και φορτίζουν μόνο αισθητήρες στην ομάδα τους και οι υπερφορτιστές, ομοίως, ομαδοποιούν και φορτίζουν φορτιστές της ομάδας τους.

Αξίζει να αναφερθεί ότι και η δρομολόγηση παίζει σημαντικό ρόλο, καθώς δρομολογήσεις που στόχο έχουν τα λιγότερα hops, καταναλώνουν, όπως είναι φυσικό, λιγότερη ενέργεια. Άρα, σημαντικό είναι να εξεταστεί και αυτό το ενδεχόμενο, για όλες τις παραπάνω μετρήσεις.

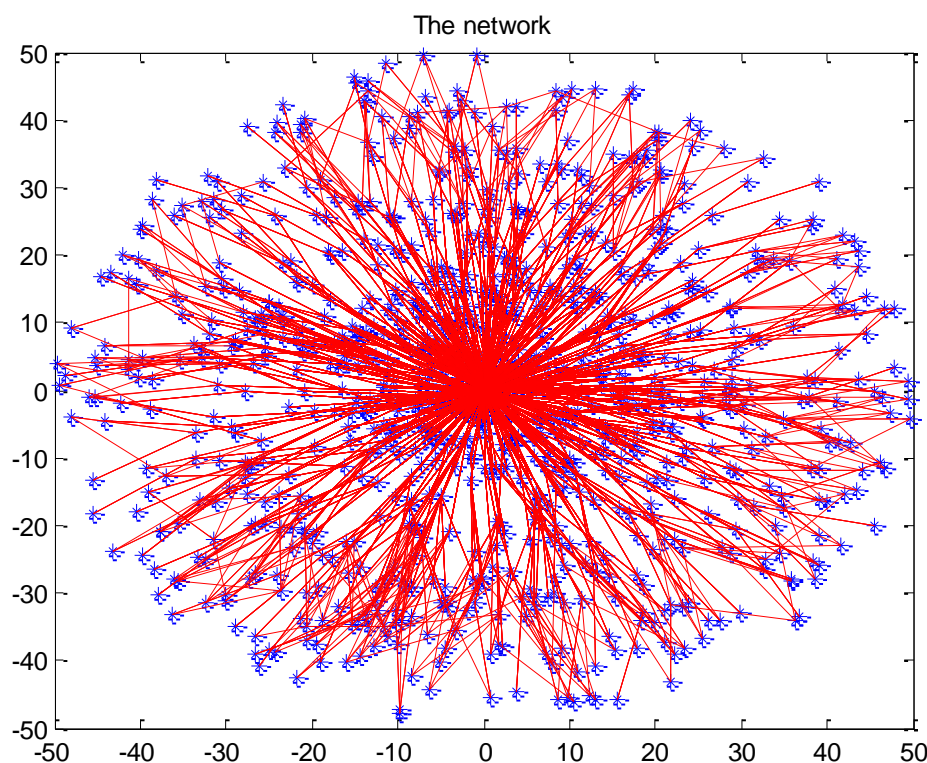
Οι μετρικές που θα χρησιμοποιηθούν, όπως είναι λογικό, θα πρέπει να είναι ίδιες σε όλες τις περιπτώσεις. Αναλυτικά θα χρησιμοποιηθούν οι παρακάτω μετρικές:

- Χρόνος μέχρι έστω και ένας αισθητήρας να βρεθεί χωρίς ενέργεια (χρόνος ζωής δηλαδή του δικτύου).
- Ο αριθμός αισθητήρων που είναι ζωντανοί σε σχέση με το χρόνο.
- Ο αριθμός πακέτων που απεστάλησαν στο χρόνο ζωής του δικτύου.

Ιδιαίτερα για την τελευταία μετρική, οι προτεινόμενοι αλγόριθμοι θα πρέπει να συγκριθούν με τον ίδιο αριθμό πακέτων (π.χ. 1000000 πακέτα), τα οποία θα ξεκινούν από τους ίδιους αισθητήρες και θα προορίζονται στους ίδιους αισθητήρες κάθε φορά, ώστε να υπάρχει σύγκριση με τους ίδιους όρους. Για να είναι δίκαιο όμως, θα πρέπει να επιλεγεί μια σειρά τυχαίων τέτοιων δρομολογήσεων και να μελετηθεί με όλες τις περιπτώσεις που εξετάζονται. Θα πρέπει τέλος, να εξεταστεί και η περίπτωση τυχαίων δρομολογήσεων, οι οποίες μπορεί να δημιουργηθούν σε κάθε χρονική στιγμή.

Ολοκληρώνοντας τη μελέτη των παραπάνω δικτύων θα υπάρχει μια πολύ καλύτερη εικόνα ώστε τελικά να μετρηθούν οι βελτιώσεις που θα γίνουν και πιθανά να οδηγήσουν σε νέες προτάσεις και βελτιώσεις.

Έτσι αρχικά το δίκτυο, το οποίο εξετάσαμε, είναι της μορφής που παρουσιάζεται στην εικόνα 11. Οι αισθητήρες στέλνουν μηνύματα σε ένα SINK, το οποίο βρίσκεται στο κέντρο του δικτύου. Το δίκτυο είναι multihop, όπως φαίνεται και στην παρακάτω εικόνα, η οποία δείχνει σειρά από μηνύματα από τους κόμβους να καταλήγουν στο SINK, περνώντας από ενδιάμεσους κόμβους. Οι κόκκινες γραμμές είναι οι τροχιές που ακολουθούν τα μηνύματα.

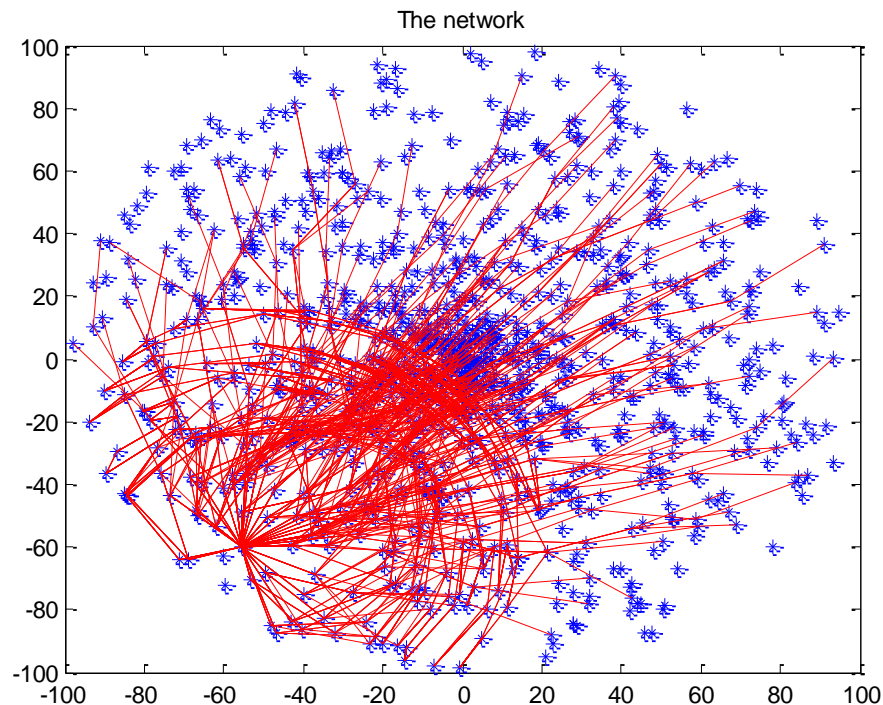


Εικόνα 11. Απεικόνιση multihop δικτύου (Α)

Ακόμα, στην εικόνα 12, παρουσιάζουμε την γενικότερη περίπτωση, κατα την οποία το SINK βρίσκεται σε μία τυχαία θέση μέσα στο δίκτυο. Απεικονίζονται οι διαδρομές για

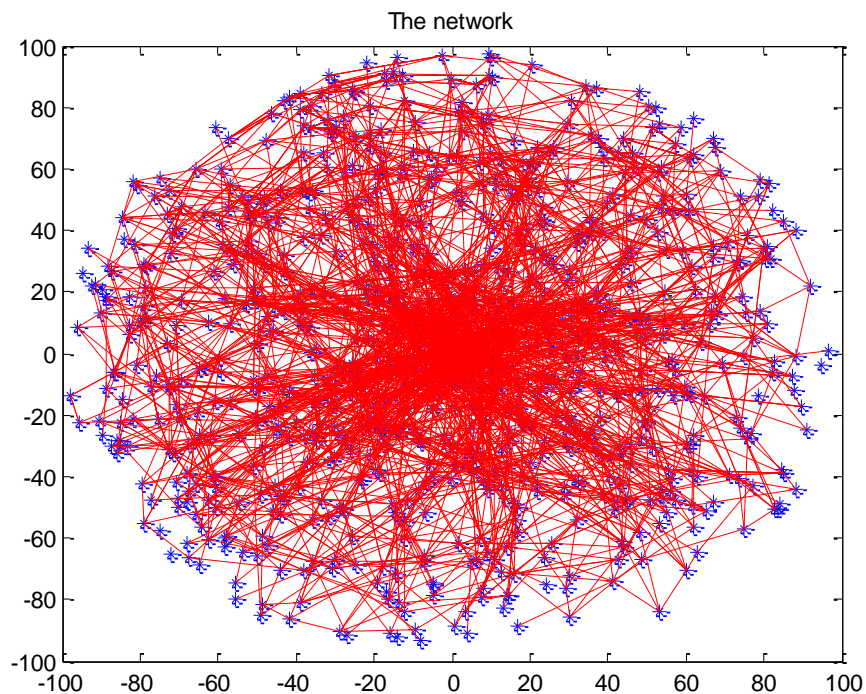


100 μηνύματα τα οποία ξεκινούν από 100 τυχαίους κόμβους και καταλήγουν στο SINK. Μπορεί εύκολα να γίνει αντιληπτό το που περίπου βρίσκεται το sink, από την πυκνότητα των μονοπατιών των μηνυμάτων τα οποία στάλθηκαν.



Εικόνα 12. Απεικόνιση multihop δικτύου (B)

Σε ένα δίκτυο, στο οποίο οι κόμβοι επικοινωνούν μεταξύ τους, και όχι μόνο με το sink, έχουμε τις τροχιές οι οποίες απεικονίζονται στην εικόνα 13. Και εδώ το δίκτυο είναι multihop και η πυκνότητα των μονοπατιών επηρεάζεται από την πυκνότητα των κόμβων στις εκάστοτε περιοχές του δικτύου.



Εικόνα 13. Απεικόνιση multihop δικτύου (Γ)

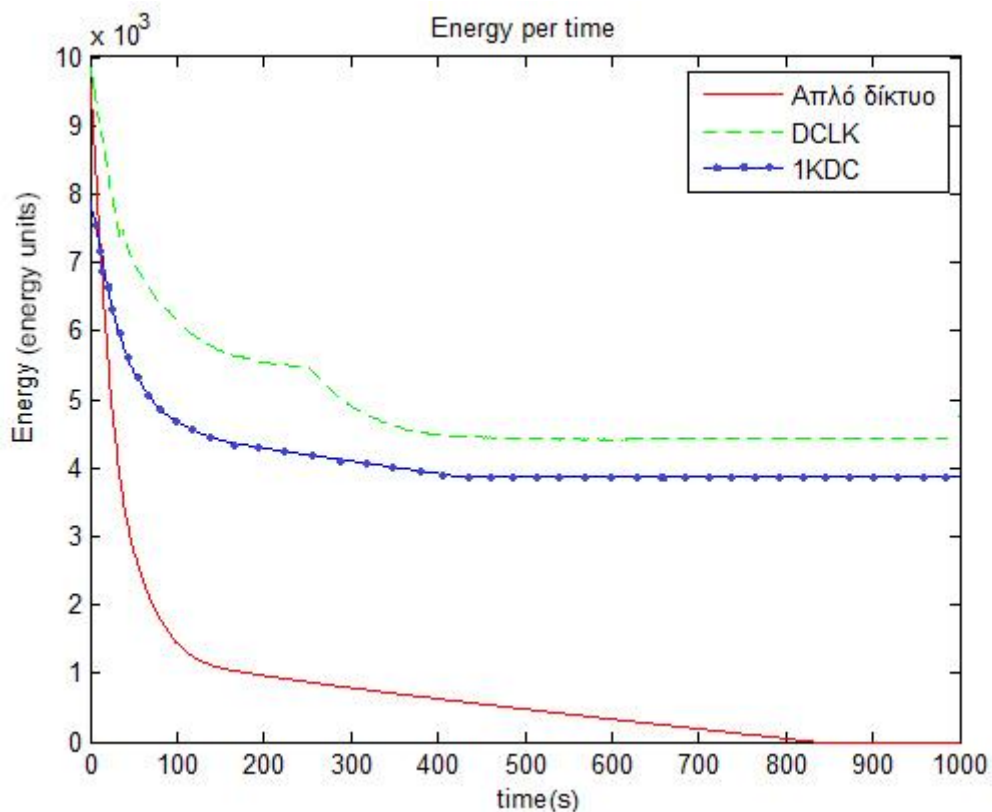
Ακολουθούν τα αποτελέσματα λειτουργίας του αλγορίθμου στις περιπτώσεις απλού δικτύου, δικτύου με χρήση φορτιστών και δικτύου με χρήση φορτιστών και υπερφορτιστών. Οι παράμετροι της υλοποίησης είναι οι εξής:

	Απλό δίκτυο	Δίκτυο με φορτιστές	Δίκτυο με φορτιστές και υπερφορτιστές
Αριθμός Nodes	1000	1000	1000
Συνολική ενέργεια (μονάδες ενέργειας)	10000	10000	10000
Διάμετρος δικτύου (m)	100	100	100
Time (s)	1000	1000	1000
Αριθμός φορτιστών	-	20	20
Αριθμός υπερφορτιστών	-	-	4



Όπως παρατηρούμε, οι παράμετροι οι οποίες είναι κοινές και για τις τρεις περιπτώσεις, έχουν την ίδια τιμή, έτσι ώστε να μπορούν να ληφθούν καλύτερα συμπεράσματα.

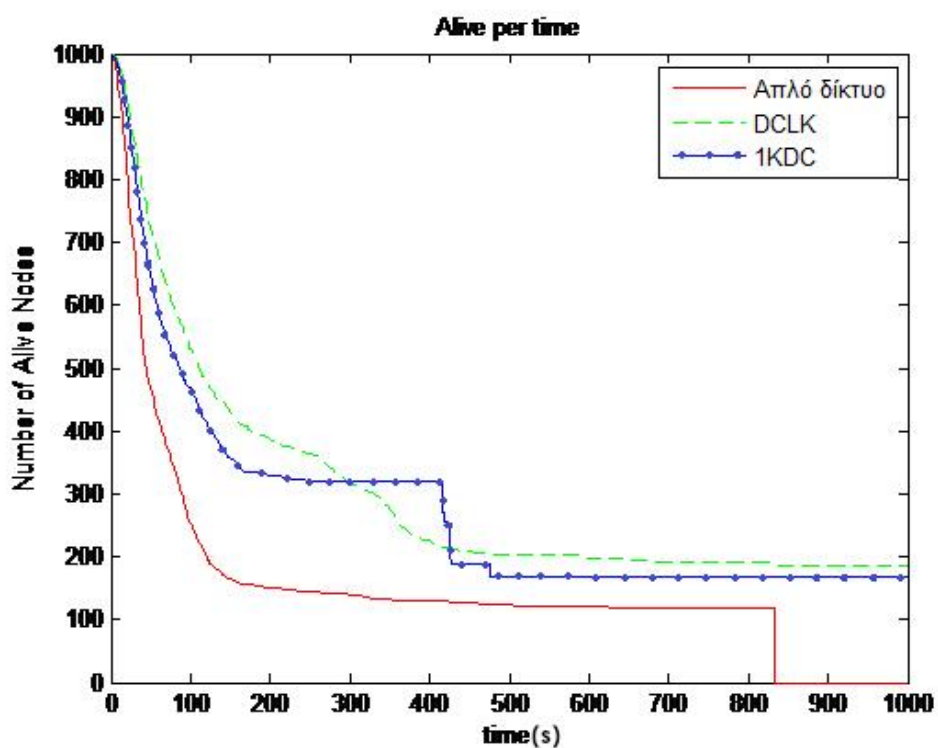
Στην παρακάτω γραφική παράσταση (εικόνα 14), απεικονίζεται η συνολική ενέργεια όπως αυτή χάνεται στο χρόνο. Όπως παρατηρούμε, η χρήση των αλγορίθμων αυξάνει αισθητά το χρόνο λειτουργίας του δικτύου. Πιο αποτελεσματική φαίνεται να είναι η χρήση φορτιστών μόνο για τη φόρτιση των κόμβων, με τη χρήση του πρωτοκόλλου DCLK, ενώ ικανοποιητική είναι και η βελτίωση με τη χρήση υπερφορτιστών.



Εικόνα 14. Αποτελέσματα – Σύνολο ενέργειας



Στην εικόνα 15, παρουσιάζονται οι ενεργοί κόμβοι με το πέρασμα του χρόνου. Όπως βλέπουμε υπάρχει αντιστοιχία με την προηγούμενη εικόνα και με την συνολική ενέργεια του δικτύου. Η βελτίωση είναι εμφανής με τη χρήση των φορτιστών και ειδικά στην περίπτωση του πρωτοκόλλου DCLK και της χρήσης μόνο φορτιστών για τους κόμβους.

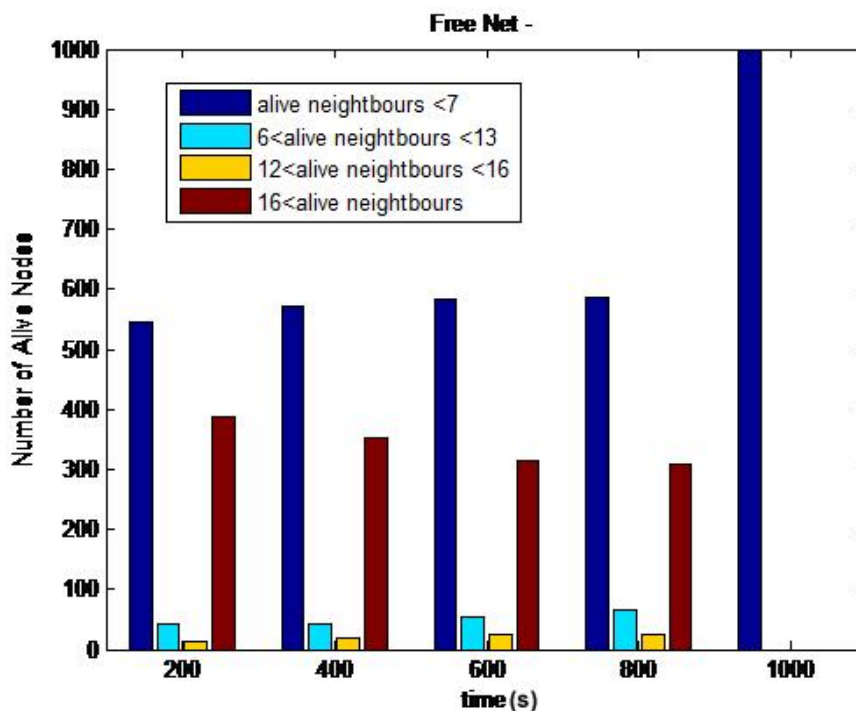


Εικόνα 15. Αποτελέσματα – Αριθμός ενεργών κόμβων

Στη συνέχεια, μέσω των επόμενων γραφικών παραστάσεων, παρουσιάζεται η κατανομή των ενεργών κόμβων σε σχέση με τον χρόνο. Κάθε περίπτωση από τις παραπάνω παρουσιάζεται σε διαφορετική γραφική παράσταση και κάθε στήλη αντιστοιχεί στον αριθμό των κόμβων που έχουν συγκεκριμένο αριθμό ενεργών γειτόνων. Στα αποτελέσματα αυτά, παίζει ρόλο και η τοπολογία του κάθε δικτύου, καθώς η τοποθέτηση των κόμβων είναι τυχαία.

## Περίπτωση απλού δικτύου

Στην περίπτωση απλού δικτύου, παρατηρούμε ότι η μείωση των κόμβων είναι αρκετά απότομη από ένα χρονικό σημείο και μετά. Αυτό φαίνεται φυσικολογικό καθώς ο ρυθμός μείωσης της ενέργειας είναι παρόμοιος για κάθε κόμβο, οπότε αποφορτίζονται μαζικά πολλοί κόμβοι.

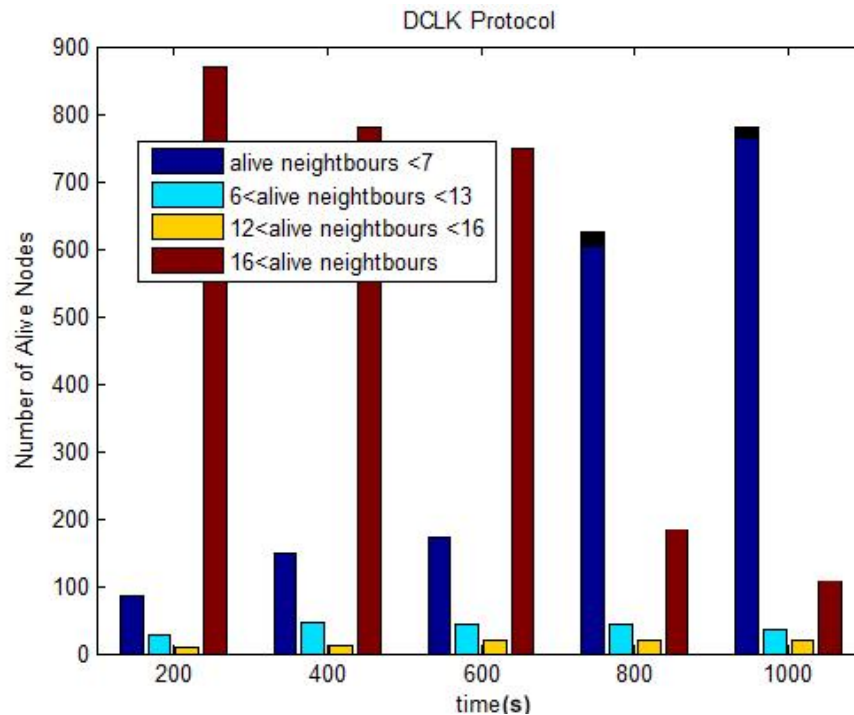


Εικόνα 16. Αποτελέσματα – Περίπτωση απλού δικτύου

## Περίπτωση Δικτύου με πρωτόκολλο DCLK

Όπως απρατηρούμε στην εικόνα 17, σε αυτήν την περίπτωση, η αποφόρτιση των κόμβων είναι πιο ομαλή. Αυτή τη φορά, καθώς υπάρχει η δυνατότητα επαναφόρτισης των κόμβων, δεν υπάρχει μαζική αποφόρτιση. Η επιμήκυνση της ζωής του δικτύου είναι

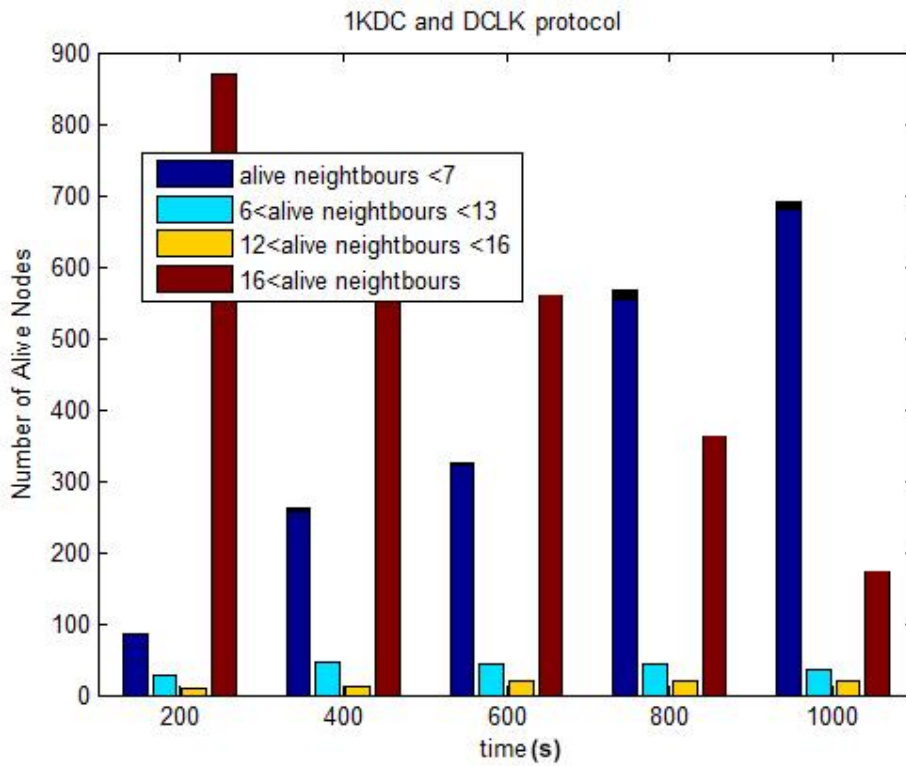
εμφανής καθώς στο ίδιο χρονικό σημείο παρατηρείται μεγαλύτερη πυκνότητα από ενεργούς κόμβους.



Εικόνα 17. Αποτελέσματα – Περίπτωση Δικτύου με πρωτόκολλο DCLK

### Περίπτωση Δικτύου με πρωτόκολλο 1KDC υπερφορτιστών και με πρωτόκολλο φορτιστών DCLK

Παρόμοια αποτελέσματα παίρνουμε και στην περίπτωση φορτιστών και υπερφορτιστών. Παρόλα αυτά βλέπουμε οι μη ενεργοί κόμβοι να γίνονται περισσότεροι όσο περνά ο χρόνος.



Εικόνα 18. Αποτελέσματα – Περίπτωση Δικτύου με πρωτόκολλο 1KDC υπερφορτιστών και με πρωτόκολλο φορτιστών DCLK



## Κεφάλαιο 6<sup>ο</sup>: Συμπεράσματα

Στην εργασία μας, εξετάστηκε το πρόβλημα για την φόρτιση των δικτύων αισθητήρων. Ένα ασύρματο δίκτυο αισθητήρων, όπως έχει ήδη αναφερθεί, αποτελείται από κόμβους-αισθητήρες, οι οποίοι απλώνονται σε μια γεωγραφική περιοχή, με σκοπό την παρακολούθηση των φυσικών φαινομένων και τη καταγραφή περιβαλλοντικών παραμέτρων, όπως η θερμοκρασία, η υγρασία, δονήσεις, σεισμικά γεγονότα, και άλλα. Μια πηγή ισχύος, παρέχει την αναγκαία ενέργεια για τις συσκευές, ώστε να εκτελέσουν την προγραμματισμένη λειτουργία τους, όπως τη συλλογή δεδομένων, τη τοπική επεξεργασία τους και την αποστολή τους.

Αυτή η πηγή ενέργειας, αποτελείται συνήθως από μια μπαταρία με περιορισμένη ενεργειακή ποσότητα, η οποία μπορεί να είναι αδύνατο ή πολύ δύσκολο να αντικατασταθεί ή ακόμα και να επαναφορτιστεί. Αυτό μπορεί να συμβαίνει για διάφορους λόγους, όπως επειδή οι κόμβοι μπορεί να τοποθετηθούν σε ένα εχθρικό ή μη πρακτικό περιβάλλον. Από την άλλη πλευρά, το δίκτυο αισθητήρων, θα πρέπει να έχει μια διάρκεια ζωής αρκετά μεγάλη για να εκπληρώσει τις απαιτήσεις της εφαρμογής. Σε πολλές περιπτώσεις, ο χρόνος ζωής της τάξεως των αρκετών μηνών, ή ακόμα και ετών, μπορεί να απαιτηθεί. Επομένως, η κρίσιμη ερώτηση είναι: "πώς να επιμηκυνθεί η διάρκεια ζωής του δικτύου για ένα μακρύ χρονικό διάστημα;"

Σε μερικές περιπτώσεις, είναι δυνατό να συλλεχθεί η ενέργεια από το εξωτερικό περιβάλλον, π.χ., με τη χρησιμοποίηση ηλιακών κυττάρων ως πηγή ενέργειας. Εντούτοις, οι εξωτερικές πηγές παροχής ενέργειας, συχνά εκθέτουν μια μη συνεχή συμπεριφορά, έτσι ώστε ένας ενεργειακός απομονωτής, μια μπαταρία, αποθηκευτικός χώρος ενέργειας, να χρειάζεται επίσης. Σε κάθε περίπτωση, η ενέργεια είναι ένας πολύ κρίσιμος πόρος και γι' αυτό πρέπει να χρησιμοποιείται με πολύ προσεγμένο τρόπο.

Επομένως, η ενεργειακή συντήρηση, είναι ένα βασικό ζήτημα στο σχεδιασμό των συστημάτων, τα οποία είναι βασισμένα στα ασύρματα δίκτυα αισθητήρων. Κατά καιρούς, έχουν υλοποιηθεί αρκετοί αλγόριθμοι, οι οποίοι αφορούν τον τρόπο με τον οποίο μπορεί να χρησιμοποιηθεί η τεχνολογία της ασύρματης φόρτισης στα δίκτυα αισθητήρων.





Αυτό όμως που προκύπτει, είναι ότι οι πολλές εναλλακτικές επιλογές που υπάρχουν, έχουν να κάνουν με πολλούς παράγοντες. Κάποιοι από αυτούς τους παράγοντες είναι:

- Η πολιτική φόρτισης, η οποία θα ακολουθηθεί κατά τη φόρτιση ενός κόμβου, αν θα είναι δηλαδή πλήρης ή μερική.
- Ο αριθμός των φορτιστών, οι οποίοι θα χρησιμοποιηθούν στο δίκτυο, αν θα είναι δηλαδή ένας ή περισσότεροι.
- Η εύρεση της καλύτερης δυνατής τροχιάς για κάθε φορτιστή, ώστε να αυξηθεί ο χρόνος ζωής του δικτύου.
- Το ποσοστό της ενέργειας, το οποίο θα διατεθεί στον ή στους φορτιστές, σε σχέση με το αντίστοιχο ποσοστό που θα δοθεί αρχικά στους κόμβους. Σημειώνεται εδώ ότι αυτό έχει να κάνει με το γεγονός ότι η συνολική ενέργεια του δικτύου είναι άθροισμα της αρχικής ενέργειας που διατίθεται στους φορτιστές και στους κόμβους.
- Ο βαθμός γνώσης, το οποίο θα έχουν οι φορτιστές για την ενεργειακή κατάσταση των κόμβων του δικτύου.
- Ο τρόπος με τον οποίο θα διαμοιράζεται το δίκτυο στους φορτιστές, εφόσον βέβαια υπάρχει αναφορά στην περίπτωση, όπου στο δίκτυο υπάρχουν περισσότεροι από έναν φορτιστές.

Έχοντας λοιπόν λάβει υπ' όψη αρκετούς από τους παραπάνω παράγοντες, στην παρούσα διπλωματική εργασία θα επιχειρηθεί μία συγκριτική μελέτη μεταξύ 2 σημαντικών πρωτοκόλλων.

Σημαντικές εργασίες στην μελέτη των παραπάνω προβλημάτων ήταν οι:

1. Efficient, Distributed Coordination of Multiple Mobile, Chargers in Sensor Networks (Adelina Madhja, Sotiris Nikolettseas, Theofanis P. Raptis).
2. Hierarchical, Collaborative, Wireless Charging in Sensors Networks (Adelina Madhja, Sotiris Nikolettseas, Theofanis P. Raptis).

Στόχος των παραπάνω εργασιών ήταν να δοθεί λύση στο πρόβλημα της εξεύρεσης της καλύτερης δυνατής τροχιάς, που θα πρέπει να ακολουθεί ο φορτιστής, ώστε να



αυξηθεί ο χρόνος ζωής του δικτύου. Για να γίνει αυτό πρέπει κάθε φορά να επιλέγεται ποιος φορτιστής θα κινηθεί , πως θα κινηθεί , ποιον αισθητήρα θα φορτίσει. Έτσι εισάγεται μία παράμετρος, η οποία αφορά τους κόμβους και έχει να κάνει με το πόσο σημαντικός είναι ο κάθε ένας από αυτούς. Με βάση την παράμετρος αυτή, επηρεάζεται τόσο από την κίνηση, δηλαδή τον όγκο δεδομένων που εξυπηρετεί ο κόμβος, όσο και από την ενέργεια που καταναλώνει. Έτσι ένας κόμβος ο οποίος έχει καταναλώσει μεγάλο ποσοστό ενέργειας, θεωρείται ότι είναι κρίσιμος και θα πρέπει να του δοθεί υψηλή προτεραιότητα από το φορτιστή. Παράλληλα θα πρέπει να εξεταστεί το ποσοστό ενέργειας, το οποίο θα δίνει ο φορτιστής σε κάθε κόμβο. Η πιο λογική επιλογή θα ήταν να φορτίζει τον κόμβο αναλογικά με την ενέργεια η οποία λείπει από τον κόμβο. Παρόλα αυτά η πολιτική αυτή αγνοεί το γεγονός ότι υπάρχουν κόμβοι, οι οποίοι είναι πιο σημαντικοί για το δίκτυο εξαιτίας της τοποθεσίας τους, του ρυθμού παραγωγής τους, του ειδικού ρόλου που μπορεί να έχουν για το δίκτυο κ.α.

Αφού μελετήσαμε τα προτεινόμενα πρωτόκολλα, εφαρμόσαμε μια απλή δρομολόγηση σε ένα απλό δίκτυο χωρίς φορτιστές, με φορτιστές σύμφωνα με το πρωτόκολλο Distributed Coordination Local Knowledge protocol (DCLK) και στην συνέχεια σε δίκτυο με υπερφορτιστές, σύμφωνα με το πρωτόκολλο 1 Level Knowledge Distributed Coordinates (1KDC) για τους υπερφορτιστές, ενώ οι φορτιστές κρατούν το πρωτόκολλο DCLK(1KDC).

Και στις τρεις αυτές περιπτώσεις, μετρήσαμε τον χρόνο από την αρχή της λειτουργίας του δικτύου, μέχρι το χρόνο που χάνεται ο πρώτος κόμβος, τον αριθμό των αισθητήρων που μένουν ζωντανοί σε σχέση με τον χρόνο και το Coverage του δικτύου.

Είδαμε ότι είχαμε αποτελεσματική φόρτιση και το δίκτυο διατηρούσε την ζωή του και συνεκτικότητα του, για σχετικά καλό χρονικό διάστημα. Πιο συγκεκριμένα είδαμε ότι έχουμε μια πιο αποτελεσματική διαχείριση της ενέργειας στο πρωτόκολλο DCLK, αφού φαίνεται να έχουμε μια πιο ομαλή καμπύλη και το δίκτυο να λειτουργεί για περισσότερο χρόνο. Παρόμοια αποτελέσματα, μας δίνει και το 1KDC, παρόλα αυτά βλέπουμε οι μη ενεργοί κόμβοι να γίνονται περισσότεροι όσο περνά ο χρόνος.



Σαν συμπέρασμα, προκύπτει ότι τα πρωτόκολλα δίνουν αποτελεσματικές λύσεις, παρόλα αυτά υπάρχουν αρκετά πράγματα που θα μπορούσαν να μελετηθούν και να εξεταστούν, τα οποία παρουσιάζονται παρακάτω.

## Προτεινόμενη συνέχεια και έρευνα

Κατά την διάρκεια των πειραμάτων μας, είδαμε ότι αλλάζοντας τις παραμέτρους, καταλήγουμε ακόμα και σε διαφορετικές μορφές των καμπύλων μας. Έτσι λοιπόν, θεωρήσαμε ότι δεν υπάρχει ενέργεια λειτουργίας των αισθητήρων, δεν υπάρχει ενέργεια λήψης μηνυμάτων παρά μόνο ενέργεια αποστολής, δεν εφαρμόζεται ενέργεια κίνησης των φορτιστών και υπερφορτιστών. Προκύπτουν όμως τα παρακάτω ερωτήματα:

- Τι συμβαίνει στα πρωτόκολλα αυτά αλλά και σε άλλα πιθανά πρωτόκολλα όταν αλλάζει το range των αισθητήρων;
- Τι συμβαίνει αν το δίκτυο αισθητήρων δεν ανταλλάσει μηνύματα μόνο με το Sink αλλά και οι αισθητήρες στέλνουν και μεταξύ τους μηνύματα;
- Τι συμβαίνει αν προσθέσουμε και ενέργεια λειτουργίας σε αισθητήρες, φορτιστές και υπερφορτιστές;
- Τι συμβαίνει αν προσθέσουμε ενέργεια κίνησης;

Ακόμα, για να ελέγξουμε την αποτελεσματικότητα κάθε πρωτοκόλλου, ορίσαμε ότι οι αισθητήρες κρατούν το 70% της ενέργειας και το 30% το έχουν οι φορτιστές. Αυτές οι δύο παράμετροι μπορούν να δοκιμαστούν με διάφορες τιμές και να φανεί η διαφορά σε σχέση και με τις υπόλοιπες παραμέτρους;

Τα παραπάνω αποτελούν αφετηρία για μια σειρά μελλοντικών μελετών, οι οποίες μπορούν να δώσουν σειρά πειραματικών αποτελεσμάτων.

Με βάση τα προηγούμενα μπορούμε να καταλήξουμε στα παρακάτω θέματα για περαιτέρω μελέτη:



- Είναι σημαντικό να ερευνηθεί τι γίνεται αν προσθέσουμε ενέργεια στην απλή λειτουργία των αισθητήρων.
- Στο αρχικό απλό δίκτυο, θα πρέπει να μελετηθούν οι παράμετροι του Range και πως αυτό επηρεάζει τελικά την ενέργεια.
- Με βάση το αποτέλεσμα, το οποίο προκύπτει πρέπει να μελετηθούν μια σειρά από πρωτόκολλα σε διαφορετικά range
- Θα πρέπει, να δοκιμαστεί η περίπτωση ανταλλαγής μηνυμάτων μεταξύ των αισθητήρων και όχι μόνο του απλού sink .
- Να δοκιμαστεί η συμπεριφορά του δικτύου και στην περίπτωση, κατά την οποία οι αισθητήρες καταναλώνουν ενέργεια για την απλή λειτουργία τους, καταναλώνουν ενέργεια για αποστολή αλλά και για λήψη.

Έτσι, αρχικά θα πρέπει να καθοριστεί ο τρόπος δρομολόγησης, έτσι ώστε να είναι ίδιος σε όλες τις περιπτώσεις. Μπορούν ακόμα, να δοκιμαστούν διάφοροι τρόποι δρομολόγησης.

Στην περίπτωση φορτιστών, καλό είναι να δοκιμαστεί και ένα πολύ απλό πρωτόκολλο, σύμφωνα με το οποίο, χωρίς ομαδοποίηση ή καθορισμό περιοχής οι φορτιστές, θα φορτίζουν απλά τυχαία τον κοντινότερο αισθητήρα από αυτούς με την μικρότερη ενέργεια. Σε αυτή την περίπτωση, θα πρέπει να δοκιμαστεί και με ενέργεια λειτουργίας αλλά και κίνησης των φορτιστών.

Επιπροσθέτως, το ίδιο πρέπει να γίνει και με τους υπερφορτιστές , δηλαδή να γίνει σύγκριση μεταξύ ενός πρωτοκόλλου, κατά το οποίο ένας υπερφορτιστής θα φορτίζει τον κοντινότερο του απλά φορτιστή. Στην περίπτωση αυτή πάλι, θα θεωρήσουμε τον ίδιο τρόπο δρομολόγησης καθώς επίσης, ότι οι φορτιστές θα λειτουργούν με απλή επιλογή του αισθητήρα φόρτισης ή με ομαδοποίηση.



## Βιβλιογραφία

- [1] Akkaya, K., & Younis, M. (2005). A survey on routing protocols for wireless sensor networks. *Ad hoc networks*, 3(3), 325-349..
- [2] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer networks*, 38(4), 393-422..
- [3] Al-Karaki, J. N., & Kamal, A. E. (2004). Routing techniques in wireless sensor networks: a survey. *IEEE wireless communications*, 11(6), 6-28.
- [4] Angelopoulos, C. M., Nikolettseas, S., Raptis, T. P., Raptopoulos, C., & Vasilakis, F. (2012, October). Efficient energy management in wireless rechargeable sensor networks. In *Proceedings of the 15th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems* (pp. 309-316). ACM.
- [5] Angelopoulos, C. M., Nikolettseas, S., & Raptis, T. P. (2013, November). Adaptive, limited knowledge wireless recharging in sensor networks. In *Proceedings of the 11th ACM international symposium on Mobility management and wireless access* (pp. 65-72). ACM.
- [6] Bonanno, G., Fantoni, R., Fichera, A., Fornetti, G., Moriconi, C., Poggi, C., ... & Fascioli, A. (2003, May). The sensing subsystem of RAS. In *Atti del Meeting Nazionale sulle Nuove Tecnologie, Frascati, Italy* (pp. 1-4).
- [7] Braginsky, D., & Estrin, D. (2002, September). Rumor routing algorithm for sensor networks. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (pp. 22-31). ACM.
- [8] Chang, J. H., & Tassiulas, L. (2004). Maximum lifetime routing in wireless sensor networks. *IEEE/ACM Transactions on networking*, 12(4), 609-619.
- [9] Chen, G., Li, C., Ye, M., & Wu, J. (2009). An unequal cluster-based routing protocol in wireless sensor networks. *Wireless Networks*, 15(2), 193-207.
- [10] Chen, T. W., & Gerla, M. (1998, June). Global state routing: A new routing scheme for ad-hoc wireless networks. In *Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on* (Vol. 1, pp. 171-175). IEEE.
- [11] Duarte-Melo, E. J., & Liu, M. (2002, November). Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks. In *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE* (Vol. 1, pp. 21-25). IEEE.
- [12] Guo, S., He, L., Gu, Y., Jiang, B., & He, T. (2014). Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. *IEEE Transactions on Computers*, 63(11), 2787-2802.
- [13] Howard, A., Mataric, M. J., & Sukhatme, G. S. (2002). Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. *Distributed autonomous robotic systems*, 5, 299-308.
- [14] Jiang, F., He, S., Cheng, P., & Chen, J. (2011, October). On optimal scheduling in wireless rechargeable sensor networks for stochastic event capture. In *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on* (pp. 69-74).



- IEEE.
- [15] Johnson, E. L. (1966). Networks and basic solutions. *Operations Research*, 14(4), 619-623.
  - [16] Krishnamachari, B., & Iyengar, S. (2004). Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks. *IEEE Transactions on Computers*, 53(3), 241-250.
  - [17] Li, Z., Peng, Y., Zhang, W., & Qiao, D. (2011, October). J-RoC: A joint routing and charging scheme to prolong sensor network lifetime. In *Network Protocols (ICNP), 2011 19th IEEE International Conference on* (pp. 373-382). IEEE.
  - [18] Pei, G., Gerla, M., & Chen, T. W. (2000). Fisheye state routing: A routing scheme for ad hoc wireless networks. In *Communications, 2000. ICC 2000. 2000 IEEE International Conference on* (Vol. 1, pp. 70-74). IEEE.
  - [19] Pirnia, S. (2010). *Energy consumption in wireless sensor networks* (Master's thesis).
  - [20] Prudvi Raj, Y., & Kumar, S. L. (2013). DISPERSED DISPENSATION OF APPARENT QUERIES IN CORDLESS SENSOR NETWORKS.
  - [21] Rahimi, M., Shah, H., Sukhatme, G. S., Heideman, J., & Estrin, D. (2003, September). Studying the feasibility of energy harvesting in a mobile sensor network. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on* (Vol. 1, pp. 19-24). IEEE.
  - [22] Rajendran, V., Obraczka, K., & Garcia-Luna-Aceves, J. J. (2006). Energy-efficient, collision-free medium access control for wireless sensor networks. *Wireless networks*, 12(1), 63-78.
  - [23] Skiena, S. (1990). Dijkstra's algorithm. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*, Reading, MA: Addison-Wesley, 225-227.
  - [24] Woo, A., Tong, T., & Culler, D. (2003, November). Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems* (pp. 14-27). ACM.
  - [25] Wu, J., & Stojmenovic, I. (2004). Ad hoc networks. *Computer*, 37(2), 29-31.
  - [26] Yamada, A., Chung, S. C., & Hinokuma, K. (2001). Optimized LiFePO<sub>4</sub> for lithium battery cathodes. *Journal of the electrochemical society*, 148(3), A224-A229.
  - [27] Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer networks*, 52(12), 2292-2330.
  - [28] Zhao, M., Li, J., & Yang, Y. (2011, September). Joint mobile energy replenishment and data gathering in wireless rechargeable sensor networks. In *Proceedings of the 23rd International Teletraffic Congress* (pp. 238-245). International Teletraffic Congress.
  - [29] Adelina Madhja, Sotiris Nikolettseas and Theofanis P. Raptis. Hierarchical, Collaborative Wireless Charging in Sensor Networks
  - [30] Adelina Madhja, Sotiris Nikolettseas and Theofanis P. Raptis. Efficient, Distributed



Coordination of Multiple Mobile Chargers in Sensor Networks

- [31] A. Kurs, A. Karalis, R. Moatt, J. D. Joannopoulos, P. Fisher, and M. Soljacic, “Wireless power transfer via strongly coupled magnetic resonances,” *Science*, 2007.
- [32] K. Kang, Y. S. Meng, J. Bréger, C. P. Grey, and G. Ceder, Electrodes with high power and high capacity for rechargeable lithium batteries. in *Science*, 2013.
- [33] S. Zhang, J. Wu, and S. Lu, Collaborative mobile charging for sensor networks, in *MASS*, 2012.
  
- [34] A. Madhja, S. Nikolettseas, and T. P. Raptis, Efficient, distributed coordination of multiple mobile chargers in sensor networks. in *MSWiM*, 2013.
- [35] H. Dai, X. Wu, L. Xu, G. Chen, and S. Lin, Using minimum mobile chargers to keep large-scale wireless rechargeable sensor networks running forever, in *ICCCN*, 2013.



## Παράρτημα – Κώδικας Εφαρμογής σε Matlab

Απλό σύστημα

```
%%  
%create network  
clear;  
  
%we supposed that we have a random network  
%all nodes can communicate with the closest neighbors  
N=1000; %number of nodes  
r=100; %diameter of the network  
TotalEnergy=10000; %total energy of the system  
MaxEnergyOfNode=TotalEnergy/N; %the maximum and initial Energy of a node  
dEtransmit=1; %energy loss send a packet per distance  
dEwork=0; %energy loss of a round working  
dErcv=0; %energy receive  
  
%number of messages  
Nmsg=50;  
  
%Total time of experimental  
Ttotal=1000;  
  
%define the range radius of a node as 10% of the diameter of the network  
range=r*0.3;  
%coordinates of nodes  
R=rand(N,1)*r;  
F=rand(N,1)*2*pi;  
Px=R.*cos(F);  
Py=R.*sin(F);  
  
%create a number of random points  
R1=rand(1000,1)*r;  
F1=rand(1000,1)*2*pi;  
Px1=R1.*cos(F1);  
Py1=R1.*sin(F1);  
  
%define the all the nodes as alive
```





```
Alive(1:N)=1;
E(1:N)=MaxEnergyOfNode;
%the max life time of a node
TimeEnd(1:N)=Ttotal;
%plot the network
figure(1)
plot(Px,Py, '*')
title ('The network');
hold

%%
%rooting

%we create a number of messages that they start from a random node to an
%other random node
for i=1:Nmsg
    n1=randi(N);
    n2=n1;
    while (n1==n2) % if the they are the same node then find another node
        n2=randi(N);
    end
    StartNode(i)=n1;
    EndNode(i)=n2;
end

%time per 100 time steps
te=1;

%for each time step

for t=1:Ttotal

    %we send the messages

    for i=1:Nmsg

        n1=StartNode(i);
        n2=EndNode(i);

        % if a message had found its target we create a new message
        if (n1==n2)

            %vector with alive nodes
            AliveS=find((1:N).*(Alive));
```



```
n1=randi(length(AliveS)); %find a random alive node

n2=n1;
while (n1==n2)
    n2=randi(length(AliveS)); %find a random alive node
end

StartNode(i)=AliveS(n1);
EndNode(i)=AliveS(n2);
end
n1=StartNode(i);
n2=EndNode(i);
d2=r;
oldn=n1;

%if the nodes are alive the message will send
if(Alive(n1)==1 && Alive(n2)==1)

    %if the destination is different of the source send the message
    if(n1~=n2)

        %found the minimum distance node from source and destination in the range to send the
message (rooting)
        mind=r*100;
        mindi=0;
        for j=1:N
            if (Alive(j)==1)
                if (j~=n1)

                    d=sqrt((Px(n1)-Px(j))^2+(Py(n1)-Py(j))^2);
                    if (d<range)
                        d2=sqrt((Px(n2)-Px(j))^2+(Py(n2)-Py(j))^2);

                        if (d2<mind)
                            mindi=j;
                            mind=d2;
                        end
                    end
                end
            end
        end

    end
end

end
```



```
n1=mindi;
if(n1==0)
    EndNode(i)=StartNode(i);
end

%send message and loss energy
if (n1~=0)
    disp([oldn n1])
    d=sqrt((Px(oldn)-Px(n1))^2+(Py(oldn)-Py(n1))^2);
    % E(n1)=E(n1)-dErcv;
    E(oldn)=E(oldn)-dEtransmit*d;

    %plot the route
    plot([Px(oldn),Px(n1)],[Py(oldn),Py(n1)],'r-');
    StartNode(i)=n1;
end

end

end

end

% decrease Energy for Working
for i=1:N
    E(i)=E(i)-dEwork;
end

%Make not Alive every node that have E<=0
for i=1:N

    if (E(i)<=0)
        if (TimeEnd(i)==Ttotal)
            TimeEnd(i)=t;
        end
        Alive(i)=0;
        E(i)=0;
    end
end

if(mod(t,100)==0)
    for i=1:1000
        NP(i)=0;
    end
end
```



```
for j=1:N
    d=sqrt((Px(j)-Px1(i)).^2+(Py(j)-Py1(i)).^2);
    if (d<range)
        if(Alive(j)==1)
            NP(i)=NP(i)+1;
        end
    end
end
end
end
%computing the alive neighbours
T(te)=t;
covarage(te,1)=sum(NP<7);
covarage(te,2)=sum(NP>6 & NP<13);
covarage(te,3)=sum(NP>12 & NP<15);
covarage(te,4)=sum(NP>16);
te=te+1;

end

%number of alive nodes per time
NAlive(t)=sum(Alive==1);

%Network Energy per time
NEnergy(t)=sum(E);
end

hold
% making the graphs for energy
figure(2)
title('Energy per time');
plot(NEnergy,'r');
xlabel('time');
ylabel('Energy');
%making the graphs for the alive nodes
figure(3);
title ('Alive per time');
plot(NAlive,'r')
xlabel('time');
```



```

ylabel('Number of Alive Nodes');
%%

figure(4);
bar(T,covarage);

title ('Free Net - ');
xlabel('time');
ylabel('Number of Alive Nodes');
legend('alive neighbours <7','6<alive neighbours <13','12<alive neighbours <16','16<alive neighbours');

disp(min(TimeEnd));

```

### Σύστημα με φορτιστές σύμφωνα με το πρωτόκολλο Distributed Coordination Local Knowledge protocol (DCLK)

```

%%
%create network

%we supposed that we have a random network
%all nodes can communicate with the closest neighbors
N=1000; %number of nodes
r=100; %diameter of the network
TotalEnergy=10000;
MaxEnergyOfNode=0.7*TotalEnergy/N; %the maximum and initial Energy of a node
dEtransmit=1; %energy loss send a packet per distance
dEwork=0; %energy loss of a round working
dErcv=0; %energy receive
NChargers=20; %number of the chargers
MaxEnergyOfChargers=0.3*TotalEnergy/NChargers; %initial energ for the chargers

```



```
%Slices in a Charger Area
KK=4;

te=1;

%number of messages
Nmsg=50;

%Total time of experimental
Ttotal=1000;

%define the range radius of a node as 10% of the diameter of the network
range=r*0.4;
%coordinates of nodes
%R=rand(N,1)*r;
%F=rand(N,1)*2*pi;
%Px=R.*cos(F);
%Py=R.*sin(F);

%initial coordinates for chargers
K=NChargers;
I=1:NChargers;
  Rcx=r/2*(cos(pi/K*(2*I-1)));
  Rcy=r/2*(sin(pi/K*(2*I-1)));
  rr(1:NChargers)=I*2*pi/K;

%define the all the nodes as alive
Alive(1:N)=1;
E(1:N)=MaxEnergyOfNode;
Echargers(1:NChargers)=MaxEnergyOfChargers;
%plot the network
%figure(1)
%plot(Px,Py,'*')
%title ('The network');
%hold

%%
%rooting
```



```
%we create a number of messages that they start from a random node to an
%other random node
for i=1:Nmsg
    n1=randi(N);
    n2=n1;
    while (n1==n2) %n1 must be a different node than n2
        n2=randi(N);
    end

    StartNode(i)=n1;
    EndNode(i)=n2;
end
%for each time step
for t=1:Ttotal

    %we send the messages

    for i=1:Nmsg
        n1=StartNode(i);
        n2=EndNode(i);

        % if a message had found its target we create a new message
        if (n1==n2)

            %vector with alive nodes
            AliveS=find((1:N).*(Alive));
            n1=randi(length(AliveS));

            n2=n1;
            while (n1==n2)
                n2=randi(length(AliveS)); %n1 must be a different node than n2 and alive
            end

            StartNode(i)=AliveS(n1);
            EndNode(i)=AliveS(n2);
        end
        n1=StartNode(i);
        n2=EndNode(i);
        d2=r;
        oldn=n1;

        %if the nodes are alive the message will send
        if(Alive(n1)==1 && Alive(n2)==1)
```



```
%if the destination is different of the source send the message
if(n1~=n2)
    %found the minimum distance node from source and destination in the range to send the
message (rooting)
    mind=r;
    mindi=0;
    for j=1:N
        if (Alive(j)==1)
            if (j~=n1)
                d=sqrt((Px(n1)-Px(j))^2+(Py(n1)-Py(j))^2);
                if (d<range)
                    d2=sqrt((Px(n2)-Px(j))^2+(Py(n2)-Py(j))^2);
                    if (d2<mind)
                        mindi=j;
                        mind=d2;
                    end
                end
            end
        end
    end

    end
    end

end
n1=mindi;

if(n1==0)
    EndNode(i)=StartNode(i);
end

%send message and loss energy
if (n1~=0)

    d=sqrt((Px(oldn)-Px(n1))^2+(Py(oldn)-Py(n1))^2);
    % E(n1)=E(n1)-dErcv;
    E(oldn)=E(oldn)-dEtransmit*d;

    %plot the route
    % plot([Px(oldn),Px(n1)],[Py(oldn),Py(n1)],'r-');
    StartNode(i)=n1;
end

end
```





```
end

end

% decrease Energy for Working
for i=1:N
    E(i)=E(i)-dEwork;
end

%%%%%%%% charging %%%%%%%%%

Emax=max(E); % maximus Energy Sensor
delta=0.1; %delta
%for each charger found the Emin_jk
for ic=1:NChargers

    % limits of sector jk
    if (ic==1)
        r1=0; r2=rr(ic);
        ric=rr(ic)/KK;
    else
        r1=rr(ic-1); r2=rr(ic);
        ric=(rr(ic)-rr(ic-1))/KK;
    end

    for ik=1:KK

        %calculate minimum Energy of Slice Sjk
        Emin=TotalEnergy;

        for i=1:N
            frad=atan(Py(i)/Px(i));
            if (frad>=r1+(ik-1)*ric && frad<r1+ik*ric) %if the node belong to the slice
                if Emin>=E(i) % if has less energy than the minimum
                    Emin=E(i); %make it minimum
                end
            end
        end
        EminD=Emin+delta*Emax/Emin;
        NSjk=0;
```



```
for i=1:N
    frad=atan(Py(i)/Px(i));
    if (frad>=r1+(ik-1)*ric && frad<r1+ik*ric)
        if (E(i)>=Emin && E(i)<=EminD)
            NSjk=NSjk+1;
        end
    end
end
ProdNS(ik)=NSjk*(Emax-Emin);
end

[mm , ik2]=max(ProdNS);
for i=1:N % choice of the node which need to be charged and charging
    frad=atan(Py(i)/Px(i));
    if (frad>=r1+(ik2-1)*ric && frad<r1+ik2*ric)
        if (Echargers(ik)>MaxEnergyOfNode-E(i))
            E(i)=MaxEnergyOfNode;
            Echargers(ik)=Echargers(ik)-(MaxEnergyOfNode-E(i));
        else
            E(i)=E(i)+Echargers(ik);
            Echargers(ik)=0;
        end
    end
end

end

%Make not Alive every node that have E<=0
for i=1:N

    if (E(i)<=0)

        Alive(i)=0;
        E(i)=0;
    end
end
```



```
end

if(mod(t,100)==0)
    for i=1:1000
        NP(i)=0;
        for j=1:N
            d=sqrt((Px(j)-Px1(i)).^2+(Py(j)-Py1(i)).^2);
            if (d<range)
                if(Alive(j)==1)
                    NP(i)=NP(i)+1;
                end
            end
        end
    end
end

%collect informations for the graph
T(te)=t;
covarage(te,1)=sum(NP<7);
covarage(te,2)=sum(NP>6 & NP<13);
covarage(te,3)=sum(NP>12 & NP<15);
covarage(te,4)=sum(NP>16);
te=te+1;

end

EChargers_all=sum(Echargers);
rv=0;
for ic=1:NChargers
    rr(ic)=rv+2*pi*Echargers(ic)/EChargers_all;
    Rcx=r/2*(cos(rv+rr(ic)/2));
    Rcy=r/2*(sin(rv+rr(ic)/2));
    rv=rv+rr(ic);
end

%number of alive nodes per time
NAlive(t)=sum(Alive==1);

%Network Energy per time
NEnergy(t)=sum(E)+sum(Echargers);
end
```



```
% hold

figure(2) % energy graph
plot(NEnergy,'r');

figure(3); %alive nodes graphs
plot(NAlive,'r')

%%

figure(5); %coverage graph
bar(T,covarage);

title ('DCFK Protocol');
xlabel('time');
ylabel('Number of Alive Nodes');
legend('alive neighbours <7','6<alive neighbours <13','12<alive neighbours <16','16<alive
neighbours');
```

Σύστημα με υπερφορτιστές σύμφωνα με το πρωτόκολλο 1 Level Knowledge Distributed Coordinates (1KDC)

```
%%
%create network

%we supposed that we have a random network
%all nodes can communicate with the closest neighbours
N=1000; %number of nodes
r=100; %diameter of the network
TotalEnergy=10000;
MaxEnergyOfNode=0.5*TotalEnergy/N; %the maximum and initial Energy of a node
dEtransmit=1; %energy loss send a packet per distance
dEwork=0; %energy loss of a round working
```



```
dErcv=0; %energy receive
NChargers=20;
MaxEnergyOfChargers=0.3*TotalEnergy/NChargers; %energy of the chargers
NSChargers=4;
MaxEnergyOfSChargers=0.2*TotalEnergy/NSChargers; %energy of the superchargers
% Slices in a Charger Area
KK=4;

%number of messages
Nmsg=50;

%Total time of experimental
Ttotal=1000;

%time per 100 time steps
te=1;

%define the range radius of a node as 10% of the diameter of the network
range=r*0.4;
%coordinates of nodes
%R=rand(N,1)*r;
%F=rand(N,1)*2*pi;
%Px=R.*cos(F);
%Py=R.*sin(F);

%initial coordinates for chargers
K=NChargers;
I=1:NChargers;
Rcx=r/2*(cos(pi/K*(2*I-1)));
Rcy=r/2*(sin(pi/K*(2*I-1)));
rr(1:NChargers)=I*2*pi/K;

%define the all the nodes as alive
Alive(1:N)=1;
E(1:N)=MaxEnergyOfNode; %energy of nodes
Echargers(1:NChargers)=MaxEnergyOfChargers; %energy of chargers
ESchargers(1:NChargers)=MaxEnergyOfSChargers; %energy of superchargers
%plot the network
%figure(1)
```



```
%plot(Px,Py,'*')
%title ('The network');
%hold

%%
%rooting

%we create a number of messages that they start from a random node to an
%other random node
for i=1:Nmsg
    n1=randi(N);
    n2=n1;
    while (n1==n2) %n2 must be different than n1
        n2=randi(N);
    end

    StartNode(i)=n1;
    EndNode(i)=n2;
end
%for each time step
for t=1:Ttotal

    %we send the messages

    for i=1:Nmsg
        n1=StartNode(i);
        n2=EndNode(i);

        % if a message had found its target we create a new message
        if (n1==n2)

            %vector with alive nodes
            AliveS=find((1:N).*(Alive));
            n1=randi(length(AliveS));

            n2=n1;
            while (n1==n2)
                n2=randi(length(AliveS)); %n2 must be different than n1 and alive
            end

            StartNode(i)=AliveS(n1);
            EndNode(i)=AliveS(n2);
        end
    end
end
```



```
n1=StartNode(i);
n2=EndNode(i);
d2=r;
oldn=n1;

%if the nodes are alive the message will send
if(Alive(n1)==1 && Alive(n2)==1)
    %if the destination is different of the source send the message
    if(n1~=n2)
        %found the minimum distance node from source and destination in the range to send the
message (rooting)
        mind=r;
        mindi=0;
        for j=1:N
            if (Alive(j)==1)
                if (j~=n1)
                    d=sqrt((Px(n1)-Px(j))^2+(Py(n1)-Py(j))^2);
                    if (d<range)
                        d2=sqrt((Px(n2)-Px(j))^2+(Py(n2)-Py(j))^2);
                        if (d2<mind)
                            mindi=j;
                            mind=d2;
                        end
                    end
                end
            end
        end

        end
        n1=mindi;

    if(n1==0)
        EndNode(i)=StartNode(i);
    end
    %send message and loss energy
    if (n1~=0)

        d=sqrt((Px(oldn)-Px(n1))^2+(Py(oldn)-Py(n1))^2);
        % E(n1)=E(n1)-dErcv;
        E(oldn)=E(oldn)-dEtransmit*mind;

        %plot the route
        % plot([Px(oldn),Px(n1)], [Py(oldn),Py(n1)], 'r-');
```



```
        StartNode(i)=n1;
    end

    end

    end

end

% decrease Energy for Working
for i=1:N
    E(i)=E(i)-dEwork;
end

%%%%% charging %%%%%

Emax=max(E); % maximus Energy Sensor
delta=0.1; %delta
%for each charger found the Emin_jk
for ic=1:NChargers

    % limits of sector jk
    if (ic==1)
        r1=0; r2=rr(ic);
        ric=rr(ic)/KK;
    else
        r1=rr(ic-1); r2=rr(ic);
        ric=(rr(ic)-rr(ic-1))/KK;
    end

    for ik=1:KK

        %calculate minimum Energy of Slice Sjk
        Emin=TotalEnergy;

        for i=1:N
            frad=atan(Py(i)/Px(i));
            if (frad>=r1+(ik-1)*ric && frad<r1+ik*ric)
```





```
        if Emin>=E(i)
            Emin=E(i);
        end
    end
end
EminD=Emin+delta*Emax/Emin;
NSjk=0;
for i=1:N
    frad=atan(Py(i)/Px(i));
    if (frad>=r1+(ik-1)*ric && frad<r1+ik*ric)
        if (E(i)>=Emin && E(i)<=EminD)
            NSjk=NSjk+1;
        end
    end
end
ProdNS(ik)=NSjk*(Emax-Emin);
end

[mm , ik2]=max(ProdNS);
for i=1:N
    frad=atan(Py(i)/Px(i));
    if (frad>=r1+(ik2-1)*ric && frad<r1+ik2*ric)
        if (Echargers(ic)>MaxEnergyOfNode-E(i))
            E(i)=MaxEnergyOfNode;
            Echargers(ic)=Echargers(ic)-(MaxEnergyOfNode-E(i));
        else
            E(i)=E(i)+Echargers(ic);
            Echargers(ic)=0;
        end
    end
end

end

end

%chargers charging by superchargers
SK=NChargers/NSChargers;
for ic=1:NSChargers
    C=(((ic-1)*SK)+1):ic*SK;
```



```
if (ic==1)
    Ep=Echargers(NChargers);
    icp=NChargers;
    icn=ic*SK+1;
else
    if (ic==NChargers)
        En=Echargers(1);
        icp=(ic-1)*SK;
        icn=1;
    else
        En=Echargers(ic+1);
        icp=(ic-1)*SK;
        icn=ic*SK+1;
    end
end

ep(ic)=ESchargers(ic)-sum(Echargers(C))-Ep;
en(ic)=ESchargers(ic)-sum(Echargers(C))-Ep;
end

for ic=1:NSChargers
    if (ic==1)
        icp=NChargers;
        icn=ic*SK+1;
    else
        if (ic==NChargers)

            icp=(ic-1)*SK;
            icn=1;
        else

            icp=(ic-1)*SK;
            icn=ic*SK+1;
        end
    end

    C=(((ic-1)*SK)+1):ic*SK;
    if (ic==1)
        ep1=NSChargers;
        en1=ep(ic+1);
    else
        if (ic==NSChargers)
```



```
    ep1=en(ic-1);
    en1=1;
    else
    ep1=en(ic-1);
    en1=ep(ic+1);

    end

end

if (ep(ic)>ep1)
    C=[icp C ];
else
    C=[C icn];
end
if (en(ic)>en1)
    C=[C icn];
else
    C=[icp C ];
end

k2=min(C);
dE=MaxEnergyOfChargers-Echargers(k2);
if (ESchargers(k2)>=dE)
    Echargers(k2)=Echargers(k2)+dE;
    ESchargers(k2)=ESchargers(k2)-dE;
else
    Echargers(k2)=Echargers(k2)+ESchargers(k2);
    ESchargers(k2)=0;
end

end

%Make not Alive every node that have E<=0
for i=1:N

    if (E(i)<=0)

        Alive(i)=0;
        E(i)=0;
```



```
end
end

if(mod(t,100)==0)
    for i=1:1000
        NP(i)=0;
        for j=1:N
            d=sqrt((Px(j)-Px1(i)).^2+(Py(j)-Py1(i)).^2);
            if (d<range)
                if(Alive(j)==1)
                    NP(i)=NP(i)+1;
                end
            end
        end
    end
end

%coverage informations
T(te)=t;
coverage(te,1)=sum(NP<7); %nodes with less than 7 alive neighbours
coverage(te,2)=sum(NP>6 & NP<13); %nodes with 7-13 alive neighbours
coverage(te,3)=sum(NP>12 & NP<15); %nodes with 13-15 alive neighbours
coverage(te,4)=sum(NP>16); %nodes with more than 16 alive neighbours
te=te+1;

end

EChargers_all=sum(Echargers);
rv=0;
for ic=1:NChargers
    rr(ic)=rv+2*pi*Echargers(ic)/EChargers_all;
    Rcx=r/2*(cos(rv+rr(ic)/2));
    Rcy=r/2*(sin(rv+rr(ic)/2));
    rv=rv+rr(ic);
end

%number of alive nodes per time
NAlive(t)=sum(Alive==1);

%Network Energy per time
NEnergy(t)=sum(E)+sum(Echargers);
end
```



```
% hold

figure(2) %energy graph
title('Energy per time');
plot(NEnergy,'g');
xlabel('time');
ylabel('Energy');
figure(3); %alive nodes graph
title ('Alive per time');
plot(NAlive,'g')
xlabel('time');
ylabel('Number of Alive Nodes');

%coverage graph%

figure(6);
bar(T,covarage);

title ('1KDC and DCFK protocol ');
xlabel('time');
ylabel('Number of Alive Nodes');
legend('alive neighbours <7','6<alive neighbours <13','12<alive neighbours <16','16<alive
neighbours');
```

## Περίπτωση ανταλλαγής μηνυμάτων κόμβου με SINK

### Απλό σύστημα

```
%%
%create network
clear;

%we supposed that we have a random network
%all nodes can communicate with the closest neighbours
```



```
N=1000; %number of nodes
r=100; %diameter of the network
TotalEnergy=100000; %total energy of the system
MaxEnergyOfNode=TotalEnergy/N; %the maximum and initial Energy of a node
dEtransmit=1; %energy loss send a packet per distance
dEwork=0; %energy loss of a round working
dErcv=0; %energy receive

%number of messages
Nmsg=100;

%Total time of experimental
Ttotal=2000;

%define the range radius of a node as 10% of the diameter of the network
range=r*0.3;
%coordinates of nodes
R=rand(N,1)*r;
F=rand(N,1)*2*pi;
Px=R.*cos(F);
Py=R.*sin(F);

%create a number of random points
R1=rand(1000,1)*r;
F1=rand(1000,1)*2*pi;
Px1=R1.*cos(F1);
Py1=R1.*sin(F1);

%define the all the nodes as alive
Alive(1:N)=1;
E(1:N)=MaxEnergyOfNode;
%the max life time of a node
TimeEnd(1:N)=Ttotal;
%plot the network
figure(1)
plot(Px,Py,'*')
title ('The network');
hold

%%
%rooting
```



```
%we create a number of messages that they start from a random node to an
%other random node
for i=1:Nmsg
    n1=randi(N-1)+2;
    % n2=n1;
    % while (n1==n2)
    %   n2=randi(N);
    % end
    StartNode(i)=n1;
    EndNode(i)=1;
    Alive(1)=1;
end

%time per 100 time steps
te=1;

%for each time step

for t=1:Ttotal

    %we send the messages

    for i=1:Nmsg

        n1=StartNode(i);
        n2=1;

        % if a message had found its target we create a new message
        if (n1==n2)

            %vector with alive nodes
            AliveS=find((1:N).*(Alive));
            n1=randi(length(AliveS));

            % n2=n1;
            % while (n1==n2)
            %   n2=randi(length(AliveS));
            % end

            StartNode(i)=AliveS(n1);
            EndNode(i)=1;
            Alive(1)=1;
        end
    end
end
```



```
n1=StartNode(i);
n2=1;
Alive(1)=1;
d2=r;
oldn=n1;

%if the nodes are alive the message will send
if(Alive(n1)==1 && Alive(n2)==1)

    %if the destination is different of the source send the message
    if(n1~=n2)

        %found the minimum distance node from source and destination in the range to send the
message (rooting)
        mind=r*100;
        mindi=0;
        for j=1:N
            if (Alive(j)==1)
                if (j~=n1)

                    d=sqrt((Px(n1)-Px(j))^2+(Py(n1)-Py(j))^2);
                    if (d<range)
                        d2=sqrt((Px(n2)-Px(j))^2+(Py(n2)-Py(j))^2);

                        if (d2<mind)
                            mindi=j;
                            mind=d2;
                        end
                    end

                end
            end

        end

        n1=mindi;
        if(n1==0)
            EndNode(i)=StartNode(i);
        end

        %send message and loss energy
        if (n1~=0)
            disp([oldn n1])
            d=sqrt((Px(oldn)-Px(n1))^2+(Py(oldn)-Py(n1))^2);
```





```
        % E(n1)=E(n1)-dErcv;
        E(oldn)=E(oldn)-dEtrasmit*d;

        %plot the route
        plot([Px(oldn),Px(n1)],[Py(oldn),Py(n1)],'r-');
        StartNode(i)=n1;
    end

    end

    end

    % decrease Energy for Working
    for i=1:N
        E(i)=E(i)-dEwork;
    end

    %Make not Alive every node that have E<=0
    for i=1:N

        if (E(i)<=0)
            if (TimeEnd(i)==Ttotal)
                TimeEnd(i)=t;
            end
            Alive(i)=0;
            E(i)=0;
        end
    end

    if(mod(t,100)==0)
        for i=1:1000
            NP(i)=0;
            for j=1:N
                d=sqrt((Px(j)-Px1(i)).^2+(Py(j)-Py1(i)).^2);
                if (d<range)
                    if(Alive(j)==1)
                        NP(i)=NP(i)+1;
                    end
                end
            end
        end
    end
end
```



```
%coverage informations
T(te)=t;
covarage(te,1)=sum(NP<7);
covarage(te,2)=sum(NP>6 & NP<13);
covarage(te,3)=sum(NP>12 & NP<15);
covarage(te,4)=sum(NP>16);
te=te+1;

end

%number of alive nodes per time
NAlive(t)=sum(Alive==1);

%Network Energy per time
NEnergy(t)=sum(E);
end

hold

figure(2) %energy graph
title('Energy per time');
plot(NEnergy,'r');
xlabel('time');
ylabel('Energy');
figure(3); %alive neighbours graph
title ('Alive per time');
plot(NAlive,'r')
xlabel('time');
ylabel('Number of Alive Nodes');
%%

figure(4); %coverage graph
bar(T,covarage);

title ('Free Net - ');
xlabel('time');
ylabel('Number of Alive Nodes');
```



```
legend('alive neighbours <7','6<alive neighbours <13','12<alive neighbours <16','16<alive  
neighbours');
```

```
disp(min(TimeEnd));
```