

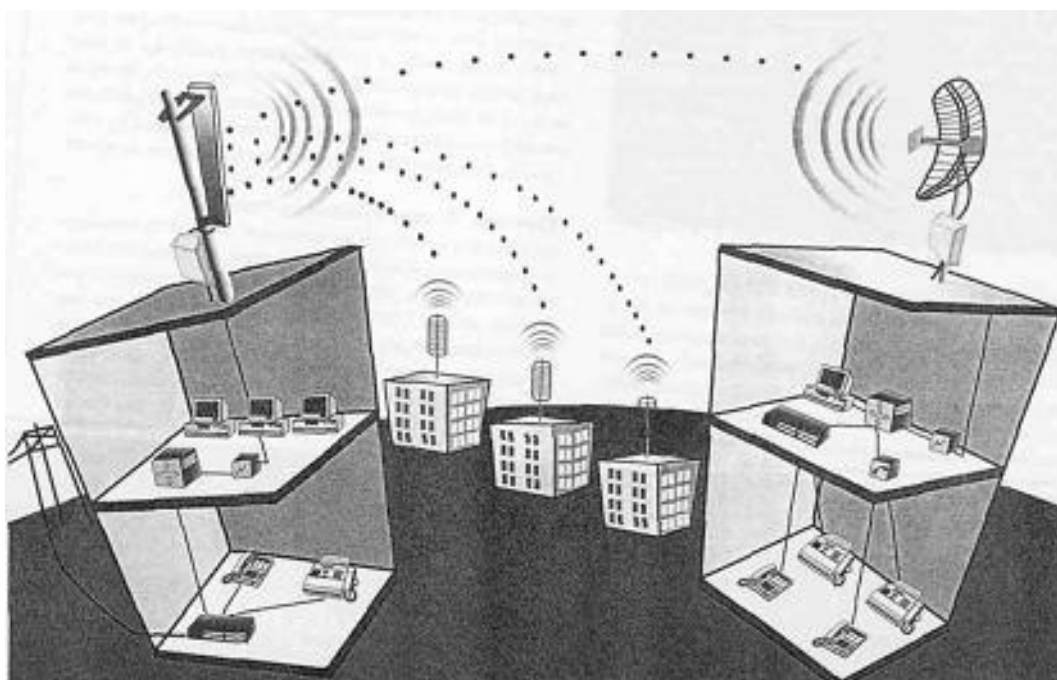


**ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**



## Πτυχιακή εργασία

**«Υλοποίηση πρωτοκόλλου ασυρμάτων τοπικών δικτύων  
IEEE 802.11 στον προσομοιωτή ns/2»**



**Του φοιτητή :  
Ασπροφρύδη Αλέξανδρου**

**Επιβλέπων καθηγητής  
Πασχάλης Ράπτης**

**Αρ. Μητρώου :  
02/2057**

**Θεσσαλονίκη 2011**

## **Ευχαριστίες**

Καταρχάς θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον κ. Πασχάλη Ράπτη , επιβλέποντα αυτής της πτυχιακής εργασίας για την καθοδήγηση, τις συμβουλές, καθώς και για το υλικό που μου πρόσφερε για την εκπόνηση της εργασίας αυτής. Έπειτα ένα μεγάλο ευχαριστώ στην οικογένειά μου για την ηθική και ψυχολογική συμπαράσταση όλα τα χρόνια των σπουδών μου, και στους φίλους και συμφοιτητές μου για τη βοήθεια τους.

# Περίληψη

Η ανάπτυξη προτύπων για τα ασύρματα τοπικά δίκτυα σε συνδυασμό με το χαμηλό κόστος της ασύρματης τεχνολογίας επέτρεψε την ραγδαία εξάπλωση των ασύρματων δικτύων σε κοινόχρηστους και ιδιωτικούς χώρους .

Η παρούσα διπλωματική εργασία έχει ως θέμα την «Υλοποίηση πρωτοκόλλου ασυρμάτων τοπικών δικτύων IEEE 802.11 στον προσομοιωτή ns/2» και μελετά το πρωτόκολλο IEEE 802.11 το οποίο κυριαρχεί στα τοπικά ασύρματα δίκτυα . Ειδικότερα ερευνά την απόδοση της κατανεμημένης διαδικασίας πρόσβασης DCF του υποεπιπέδου MAC του IEEE 802.11 . Η εργασία επικεντρώνεται στο πρόβλημα της εξομοίωσης ασύρματων δικτύων και αξιολογεί την καταλληλότητα εργαλείων ανοικτού κώδικα (Open Source) όπως είναι το NS. Με τα κατάλληλα προγράμματα εξομοιώνονται πολλά διαφορετικά σενάρια από άποψη τοπολογίας δικτύου.

Το πρώτο κεφάλαιο περιγράφει γενικά για τα δίκτυα Η/Υ ,τον σκοπό των δικτύων αυτών και την αρχιτεκτονική . Επίσης περιγράφονται τα διάφορα είδη δικτύων και κάποιες βασικές τοπολογίες των δικτύων Η/Υ.

Το δεύτερο κεφάλαιο περιγράφει γενικά την αρχιτεκτονική των ασύρματων δικτύων και τις κατηγορίες των τοπολογιών που μπορούν να υλοποιηθούν με την χρήση ασύρματων δικτύων. Αφού γίνει μια εισαγωγή στα ασύρματα δίκτυα παρουσιάζεται αναλυτικά το πρότυπο IEEE 802.11. Επίσης το δεύτερο κεφάλαιο επικεντρώνεται στο υποεπίπεδο Ελέγχου Πρόσβασης στο Μέσο (MAC), μέσω της λειτουργικής περιγραφής του. Ειδικότερα περιγράφονται οι Λειτουργίες Συντονισμού του προτύπου IEEE 802.11, με έμφαση στην Κατανεμημένη Λειτουργία Συντονισμού (DCF) και τις υπομεθόδους που αυτή διαθέτει (Βασική και RTS/CTS).

Το τρίτο κεφάλαιο περιγράφει γενικά τον προσομοιωτή NS-2 , την δομή του και τα χαρακτηριστικά . Παρουσιάζει λεπτομερή οδηγίες για την εγκατάσταση του σε λειτουργικά Windows και Linux. Επίσης περιγράφει το αρχείο NAM και το αρχείο trace .

Το τέταρτο κεφάλαιο περιγράφει την σύνταξη σεναρίων σε TCL και παρουσιάζει το βασικό σενάριο προσομοίωσης που θα χρησιμοποιήσουμε στην πτυχιακή εργασία .

Το πέμπτο κεφάλαιο περιγράφει τα στατιστικά στοιχεία και τα αποτελέσματα από την προσομοίωση των σεναρίων.

## Abstract

The development of standards for wireless local area networks, combined with the low cost of wireless technology has enabled the rapid deployment of wireless networks in public and private spaces.

This thesis is on "Implementation of wireless protocol IEEE 802.11 local area networks in the simulator ns / 2 » and studies the IEEE 802.11 protocol which dominates the local wireless networks. Specifically investigates the performance of distributed access procedure DCF MAC sublevel of the IEEE 802.11. The paper focuses on the problem of simulation of wireless networks and evaluates the adequacy of open source tools (Open Source) such as NS. With the appropriate programs many different scenarios are assimilated in terms of network topology.

The first chapter describes in general terms the computer networks, the purpose of these networks and architecture. It also describes the different types of network topologies and some basic network PC.

The second chapter describes the general architecture of wireless networks and the types of topologies that can be implemented using wireless networks. After an introduction to wireless networks, IEEE 802.11 is presented in detail. Also, the second chapter focuses on the sublevel medium access control (MAC), through the functional description. Specifically, the functions of coordinating the standard IEEE 802.11 are described, with an emphasis on Distributed Coordination Function (DCF) and the ypomethodous that it has (Basic and RTS / CTS).

The third chapter describes the general simulator NS-2, the structure and characteristics. It presents detailed instructions for installing on Windows and Linux. It also describes the file and the NAM trace file

The fourth chapter describes the syntax of scripts to TCL. It presents the baseline simulation that we will use in this thesis.

The fifth chapter describes the statistics and the results of the simulated scenarios.

# ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Ευχαριστίες .....	2
Περίληψη .....	3
Abstract .....	5
Πίνακας περιεχομένων .....	7
Κατάλογος Σχημάτων .....	11
Κατάλογος Πινάκων .....	12
<b>1.ΔΙΚΤΥΑ Η/Υ .....</b>	<b>13</b>
1.1 Τι είναι Δίκτυο Υπολογιστών .....	13
1.2 Σκοπός των Δικτύων .....	13
1.3 Αρχιτεκτονική των Δικτύων .....	14
1.4 Είδη Δικτύων .....	14
1.4.1 Δίκτυα με βάση την γεωγραφική ανάπτυξη .....	14
1.4.2 Δίκτυα με βάση τον τηλεπικοινωνιακό φορέα εξυπηρέτησης .....	15
1.5 Τοπολογία δικτύων υπολογιστών .....	16
1.5.1 Αρτηρίας (BUS) .....	16
1.5.2 Δακτυλίου ( RING) .....	17
1.5.3 Αστέρα (STAR ) .....	17
<b>2.ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ .....</b>	<b>18</b>
2.1 Πλεονεκτήματα ασύρματων δικτύων .....	19
2.2 Μειονεκτήματα ασύρματων δικτύων .....	20
2.3 Κατηγορίες Ασύρματων Δικτύων.....	22
2.3.1 Ασύρματα σωματικά δίκτυα (Wireless Body Area Network - WBANs). .....	22
2.3.2 Ασύρματα προσωπικά δίκτυα. (Wireless Personas Networks - WPANs).....	22
2.3.3 Ασύρματα Τοπικά Δίκτυα (Wireless Local Area Network - WLANs).....	22
2.3.4 Ασύρματα Μητροπολιτικά Δίκτυα (Wireless Metropolitan Area Network - WMANs).....	23

2.3.5 Ασύρματα δίκτυα ευρείας περιοχής (Wireless Wide Area Network - WWANs).....	23
2.4 Η Οικογένεια των Πρωτοκόλλων IEEE 802.11.....	23
2.4.1 IEEE 802.11.....	23
2.4.2 IEEE 802.11a.....	23
2.4.3 IEEE 802.11b.....	24
2.4.4 IEEE 802.11g.....	24
2.4.5 IEEE 802.11c.....	24
2.4.6 IEEE 802.11d.....	24
2.4.7 IEEE 802.11e.....	24
2.4.8 IEEE 802.11f.....	24
2.4.9 IEEE 802.11i.....	25
2.4.10 IEEE 802.11j.....	25
2.4.11 IEEE 802.11k.....	25
2.4.12 IEEE 802.11m.....	25
2.4.13 IEEE 802.11n.....	25
2.4.14 IEEE 802.11p.....	25
2.4.15 IEEE 802.11r.....	25
2.4.16 IEEE 802.11s.....	25
2.4.17 IEEE 802.11,2.....	25
2.4.18 IEEE 802.11u.....	25
2.5 Η πιστοποίηση Wi-Fi.....	26
2.6 Τοπολογία ασύρματων δικτύων. ....	27
2.6.1 BSS (Basic Service Set) .....	27
2.6.2 IBSS (Independent Basic Service Set) .....	28
2.6.3 ESS (Extended Service Set) .....	28
2.7 Διαστρωμάτωση του IEEE 802.11.....	29
2.7.1 Φυσικό Στρώμα (PHY) .....	30
2.7.2 Υπόστρωμα MAC .....	31
2.8 Τύποι πλαισίων .....	32
2.9 Μορφή πλαισίων .....	32
2.9.1 Preamble. ....	32
2.9.2 PLCP Header . ....	34
2.9.3 MAC Data.....	34



2.9.4 CRC.....	34
2.10 Χρονοθυρίδα και χρόνοι αναμονής .....	35
2.11 Τρόποι Έρευνας της Απόδοσης .....	36
2.12 Διαδικασία DCF του 802.11.....	37
2.12.1 Βασική μέθοδος πρόσβασης .....	39
2.12.2 RTS/CTS Μέθοδος πρόσβασης .....	40
2.12.3 Το πρόβλημα κρυμμένου τερματικού.....	42
<b>3. Network Simulator-2 .....</b>	<b>44</b>
3.1 Γενικά.....	44
3.2 <u>Εγκατάσταση του NS2</u> .....	44
3.2.1 Εγκατάσταση του NS2 στα Windows μέσω του Cygwin.....	45
3.2.2 Εγκατάσταση του NS2 στα Windows μέσω του VMWare.....	46
3.2.3 Εγκατάσταση του NS2 στα Linux (Ubuntu).....	47
3.3 Αρχεία εξόδου.....	48
3.3.1 Αρχεία NAM .....	48
3.3.2 Αρχεία TR.....	49
<b>4. ΠΕΙΡΑΜΑΤΙΚΟ ΜΕΡΟΣ.....</b>	<b>53</b>
4.1 Απλό παράδειγμα.....	53
4.2 Παράδειγμα αρχείου προσομοίωσης.....	60
<b>5. ΜΕΤΡΗΣΗΣ – ΣΥΜΠΕΡΑΣΜΑΤΑ.....</b>	<b>66</b>
5.1 Throughput (Ρυθμοαπόδοση) .....	66
5.2 Network Delay (Καθυστέρηση δικτύου).....	66
5.3 Εξαγωγή Στατιστικών .....	67
5.3.1 Throughput.awk.....	67
5.3.2 Delay.awk.....	68
5.3.3 Results.awk.....	69
5.4 Στατιστικά Αποτελέσματα.....	70
5.4.1 Διάγραμμα 1 . Μέση Ρυθμοαπόδοση (Average Throughput).....	72
5.4.2 Διάγραμμα 2 . Μέση Καθυστέρηση (Average Delay).....	72
5.4.3 Διάγραμμα 3 . Σύνολο Απορριφθέντων Πακέτων (Drop Packet)....	73

5.4.4 Διάγραμμα 4 . Σύνολο Συγκρουόμενων Πακέτων(Collision Packet).....	74
5.4.5 Διάγραμμα 5 . Σύνολο Σταλμένων Πακέτων(Sent Packet).....	74
5.4.6 Διάγραμμα 6 . Σύνολο Ληφθέντων Πακέτων(Receive Packet) ...	75
<b>Παράρτημα</b> .....	<b>76</b>
<b>Βιβλιογραφία</b> .....	<b>97</b>

# ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

<u>Αρ.</u>	<u>Περιγραφή</u>	<u>Σελ.</u>
Σχήμα 1.1.	Δίκτυο Υπολογιστών .....	10
Σχήμα 1.2.	WAN, LAN, MAN.....	12
Σχήμα 1.3.	Τοπολογία Δικτύων Υπολογιστών.....	13
Σχήμα 2.1.	Ασύρματο Δίκτυο.....	15
Σχήμα 2.2	Λογότυπο πιστοποίησης.....	23
Σχήμα 2.3	Basic Service Set.....	24
Σχήμα 2.4	Independent Basic Service Set.....	25
Σχήμα 2.5	Extended Service Set.....	26
Σχήμα 2.6	Διαστρωμάτωση του IEEE 802.11.....	27
Σχήμα 2.7	Long and Short Preamble.....	30
Σχήμα 2.8	Μορφή του MAC πλαισίου.....	31
Σχήμα 2.9	Χρόνοι αναμονής και η σχέση μεταξύ τους.....	33
Σχήμα 2.10	Βασική μέθοδος πρόσβασης.....	37
Σχήμα 2.11	RTS/CTS μέθοδος πρόσβασης.....	39
Σχήμα 2.12	πρόβλημα κρυμμένου τερματικού.....	40
Σχήμα 3.1	. Αρχεία NAM.....	46
Σχήμα 3.2	Απεικόνιση αρχείο.tr με το xgraph.....	47

## ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

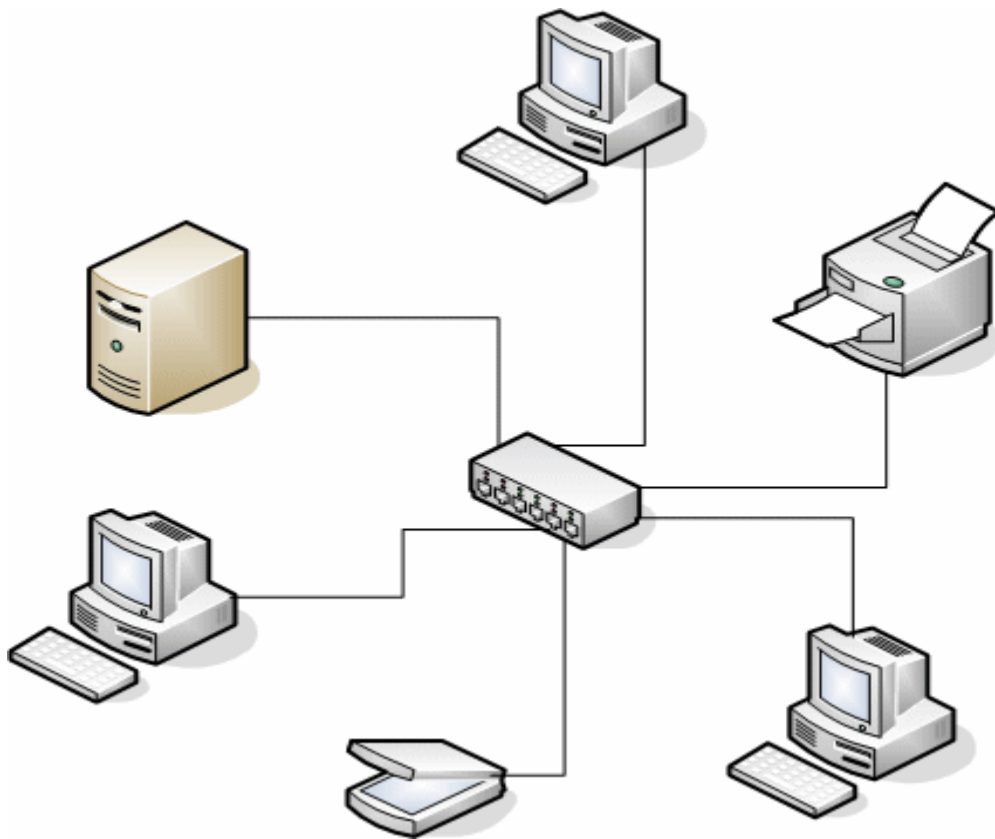
<u>Αρ.</u>	<u>Περιγραφή</u>	<u>Σελ.</u>
Πίνακας 5.1	Χαρακτηριστικά προσομοίωσης .....	70
Πίνακας 5.2	Τιμές για την μέση ρυθμοαπόδοση και μέση καθυστέρηση.....	68
Πίνακας 5.3	Τιμές για το σύνολο συγκρούσεων και απορριφθέντων πακέτων.....	69
Πίνακας 5.4	Τιμές για το σύνολο των σταλμένων και ληφθέντων πακέτων.	69

# Κεφάλαιο 1

## ΔΙΚΤΥΑ Η/Υ

### 1.1 Τι είναι Δίκτυο Υπολογιστών.

Ένα δίκτυο υπολογιστών είναι ένα σύστημα επικοινωνίας δεδομένων που συνδέει δύο ή περισσότερους αυτόνομους και ανεξάρτητους υπολογιστές και περιφερειακές συσκευές. Δύο υπολογιστές θεωρούνται διασυνδεδεμένοι όταν μπορούν να ανταλλάσσουν μεταξύ τους πληροφορίες.



Σχήμα 1.1. Δίκτυο Υπολογιστών

### 1.2 Σκοπός των Δικτύων.

Τα δίκτυα δημιουργήθηκαν για να εξυπηρετήσουν τις ανάγκες που προέκυψαν από την εξάπλωση της χρήσης των υπολογιστών. Βασικός σκοπός της ύπαρξης των δικτύων είναι ο διαμερισμός των πόρων του συστήματος και η ανταλλαγή πληροφοριών κάθε μορφής (προγράμματα,

video, μουσική, δεδομένα). Πόροι του συστήματος μπορούν να είναι είτε υλικό (hardware), π.χ. υπολογιστές, εκτυπωτές, plotters, σκληροί δίσκοι είτε λογισμικό (software), π.χ. δεδομένα, προγράμματα εφαρμογών, υπηρεσίες. Τα προγράμματα, τα δεδομένα και οι συσκευές (σκληροί δίσκοι, εκτυπωτές, κλπ) είναι διαθέσιμα σε οποιονδήποτε είναι συνδεδεμένος στο δίκτυο, ανεξάρτητα από τη φυσική του θέση. Με τον τρόπο αυτό επιτυγχάνεται εξοικονόμηση χρημάτων, αύξηση της απόδοσης του συστήματος, κεντρικός έλεγχος και εύκολη επεκτασιμότητα. Σε ένα δίκτυο μπορούμε να έχουμε ανταλλαγή δεδομένων, προγραμμάτων, χρήση κοινών βάσεων δεδομένων, αρχείων, αποστολή μηνυμάτων (electronic mail). Επιπλέον, ανεξάρτητα της τεχνολογίας, ένα δίκτυο είναι ένα πανίσχυρο μέσο επικοινωνίας ανθρώπων που βρίσκονται σε διαφορετικά μέρη.

### **1.3 Αρχιτεκτονική των Δικτύων.**

Η αρχιτεκτονική των δικτύων καθορίζει τον τρόπο με τον οποίο οι υπολογιστές και οι λοιπές συσκευές συνδέονται μεταξύ τους για να σχηματίσουν ένα σύστημα επικοινωνίας που θα επιτρέπει στους χρήστες να διαμοιράζονται πληροφορίες και συσκευές του δικτύου. Σε ένα δίκτυο δεδομένων περιλαμβάνονται:

1. Τερματικοί Κόμβοι.

Ελέγχουν τους πόρους του δικτύου (λογισμικό και υλικό).

2. Υποδίκτυα.

Φυσικά μέσα μετάδοσης, πρωτόκολλα επικοινωνίας, τοπολογία, τερματικοί κόμβοι, πόροι που μπορούν να διαφέρουν πολύ ανά υποδίκτυο.

3. Συσκευές Διασύνδεσης.

Διασυνδέουν τα ετερογενή υποδίκτυα έτσι ώστε να εξασφαλίζεται η επικοινωνία τερματικών κόμβων που βρίσκονται σε διαφορετικά υποδίκτυα.

### **1.4 Είδη Δικτύων.**

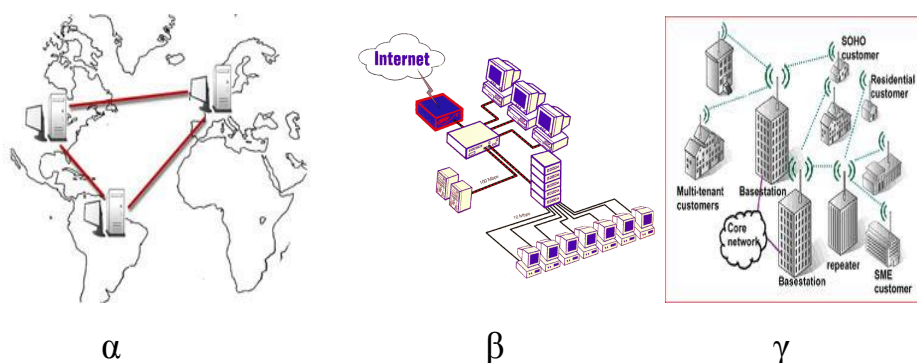
#### **1.4.1 Με βάση την γεωγραφική ανάπτυξη διακρίνονται σε :**

**Δίκτυα ευρείας περιοχής (Wide Area Networks, WAN),** που καλύπτουν αποστάσεις μερικών χιλιομέτρων (συνήθως άνω των 5 km) στην ίδια πόλη, μέχρι χιλιάδων χιλιομέτρων σε διαφορετικές πόλεις - κράτη ηπείρους.

Αποτελούνται από υπολογιστές, τηλεπικοινωνιακές συσκευές και γραμμές. Παραδείγματα τέτοιων δικτύων είναι τα δίκτυα των αεροπορικών εταιρειών,, τα τραπεζικά δίκτυα, τα δημόσια δίκτυα δεδομένων κλπ.(Σχήμα 1.2α). (Α. Αλεξόπουλος, Γ. Λαγογιαννης , Douglas E. Comer)

**Δίκτυα μικρών αποστάσεων ή τοπικά δίκτυα (Local Area Networks, LAN)** που καλύπτουν μικρές αποστάσεις (μερικών εκατοντάδων μέτρων ή λίγων χιλιομέτρων) και περιορίζονται στα πλαίσια μιας επιχείρησης. Ο διαχωρισμός τους από τα δίκτυα ευρείας περιοχής οφείλεται στο ότι χρησιμοποιούν διαφορετικές τεχνικές λειτουργίας. (Σχήμα 1.2β). (Α. Αλεξόπουλος, Γ. Λαγογιαννης , Douglas E. Comer)

**Αστικά Δίκτυα (Metropolitan Area Networks, MAN)**, που καλύπτουν δίκτυα που δεν ξεπερνούν τα σύνορα μιας πόλης. Είναι ταχύτερα από τα τοπικά δίκτυα και μπορούν να μεταδίδουν εικόνα, φωνή και δεδομένα αποδοτικότερα. (Σχήμα 1.2γ). (Α. Αλεξόπουλος, Γ. Λαγογιαννης , Douglas E. Comer)



Σχήμα 1.2. WAN, LAN, MAN

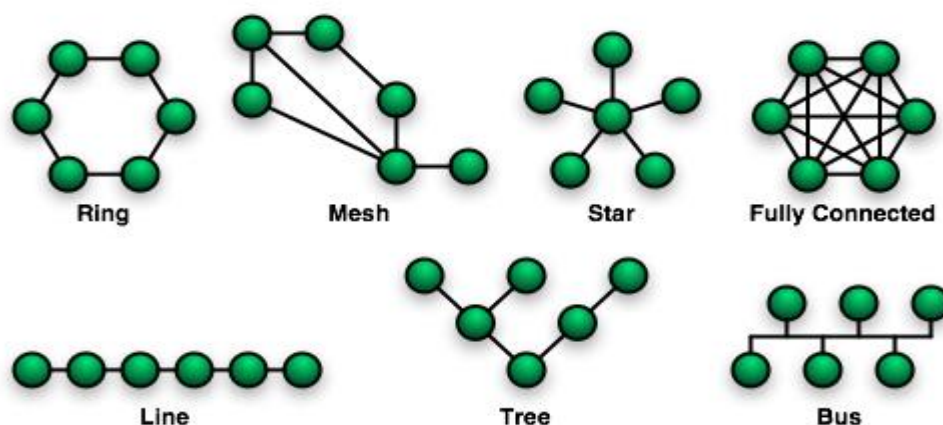
#### 1.4.2 Με βάση τον τηλεπικοινωνιακό φορέα εξυπηρέτησης διακρίνονται σε:

**Ιδιωτικά δίκτυα (Private Networks).** Ανήκουν εξ'ολοκλήρου σε ιδιωτικούς οργανισμούς και χρησιμοποιούν είτε αποκλειστικές γραμμές επικοινωνίας δημόσιων τηλεπικοινωνιακών φορέων (leased lines) χωρίς να τις μοιράζονται με άλλους χρήστες ή ιδιόκτητες γραμμές επικοινωνίας.

**Δημόσια δίκτυα** (Public Networks) που εξυπηρετούν τις διασυνδέσεις μεταξύ απομακρυσμένων σημείων. Χρησιμοποιούνται όταν η απόσταση είναι μεγάλη και καθίσταται απαγορευτική, λόγω κόστους, η χρήση αποκλειστικών γραμμών ή όταν ο φόρτος μεταξύ των σημείων δεν είναι μεγάλος και επιτυγχάνεται έτσι μεγάλη ταχύτητα μεταφοράς.

## 1.5 Τοπολογία δικτύων υπολογιστών

**Τοπολογία δικτύου** ονομάζεται η μορφή της σύνδεσης μεταξύ των κόμβων ενός δικτύου. Οι τοπολογίες είναι είτε φυσικές είτε λογικές. Τα είδη τοπολογιών είναι η γραμμική, τύπου διαύλου, δακτυλίου, αστέρα, κατανεμημένη, πλήρως κατανεμημένη και τύπου δένδρου. (Σχήμα 1.3).



Σχήμα 1.3. Τοπολογία Δικτύων Υπολογιστών.

Παρακάτω θα αναλύσουμε τα τρία βασικά: Αρτηρίας (BUS), Δακτυλίου (RING), Αστέρα (STAR).

### 1.5.1 Αρτηρίας (BUS)

Όλοι οι σταθμοί συνδέονται πάνω στο μοναδικό μέσο μετάδοσης την **αρτηρία**. Κάθε σταθμός έχει μια μοναδική ταυτότητα (διεύθυνση). Ένας σταθμός για να επικοινωνήσει με κάποιον άλλο τοποθετεί τη διεύθυνση του παραλήπτη στο μήνυμα και το αποστέλλει στην αρτηρία. Οι υπόλοιποι σταθμοί ακούνε το μήνυμα αλλά μόνο αυτός που έχει τη διεύθυνση του παραλήπτη κάνει χρήση των δεδομένων του εκτός αν η διεύθυνση προσδιορίζει πολλούς παραλήπτες (broadcast & multicast). Το πιο



συνηθισμένο μέσο υλοποίησης της αρτηρίας είναι το ομοαξονικό καλώδιο . Η συνηθέστερη υλοποίηση δικτύου με τοπολογία αρτηρίας είναι το **Ethernet** . Μια γνωστή παραλλαγή της τοπολογίας αρτηρίας είναι η τοπολογία δένδρου.

### 1.5.2 Δακτυλίου ( RING)

Η τοπολογία δακτυλίου είναι μια σύνδεση από σημείο σε σημείο ( point to point ) , η οποία δημιουργεί **κλειστό κύκλωμα** . Η ροή της πληροφορίας είναι πάντα προς μία κατεύθυνση και ο σταθμός που θέλει να μεταδώσει σε κάποιο κόμβο του δικτύου μεταφέρει το μήνυμα με τη διεύθυνση του παραλήπτη στο γειτονικό του κόμβο . Εκείνος αν είναι ο τελικός αποδέκτης , το παραλαμβάνει και το αναμεταδίδει στον επόμενο , αλλιώς , απλώς το μεταδίδει στον επόμενο. Τελικά , το μήνυμα φτάνει στον αρχικό κόμβο ο οποίος και το αποσύρει από το δακτύλιο μιας και ο παραλήπτης το έλαβε . Αν το μήνυμα είχε χαθεί , τότε ο αρχικός κόμβος δεν θα το λάμβανε ποτέ πίσω και συνεπώς θα μπορούσε να προχωρήσει στην επανεκπομπή του . Ο δακτύλιος συνήθως υλοποιείται με θωρακισμένα συνεστραμμένα ζεύγη καλωδίων ( STP – Shielded Twisted Pair Wire ). Μια άλλη συνηθισμένη μορφή είναι ο διπλός δακτύλιος οπτικών ινών ( FDDI - Fiber Distributed Data Interface ) με ταχύτητα 100 Mbits /sec.

### 1.5.3 Αστέρα (STAR )

Η τοπολογία αστέρα τείνει να υποκαταστήσει τις περισσότερες μορφές τοπικών δικτύων. Όλοι οι σταθμοί εργασίας ( workstations ) συνδέονται με τον **κεντρικό κόμβο** του αστέρα , που συνήθως είναι ένας κατανεμητής καλωδίων . Η μετάδοση από οποιονδήποτε κόμβο σε κάποιον άλλο γίνεται μέσω του κεντρικού κόμβου . Αν υπάρχει πρόβλημα σε οποιονδήποτε περιφερειακό κόμβο , το δίκτυο συνεχίζει ομαλά τη λειτουργία του .Φυσικά η απόδοση του αστέρα εξαρτάται από τις δυνατότητες του κεντρικού κόμβου , που σε αρκετές περιπτώσεις μπορεί να είναι ένας μεταγωγέας ( PBX – Private Branch extension ) με δυνατότητες μεταφοράς και δεδομένων και φωνής με μεταγωγή μηνύματος ή μεταγωγή κυκλώματος .

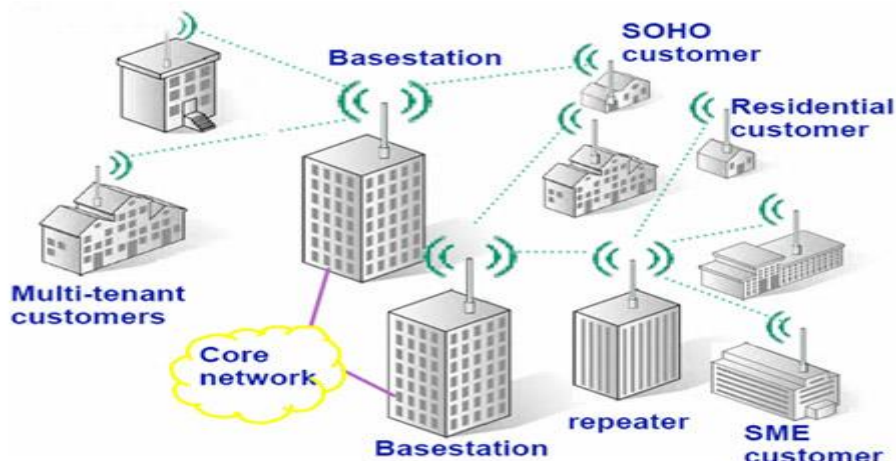
# Κεφάλαιο 2

## ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ

Είναι εμφανές ότι η αρχή της τρίτης χιλιετίας μπορεί να χαρακτηριστεί ως δικτυακή εποχή. Η εξάπλωση των δικτύων υπολογιστών συντελείται με τέτοιο ρυθμό, που πλέον έχει γίνει αναπόσπαστο κομμάτι της καθημερινής ζωής του σύγχρονου ανθρώπου. Οι υπολογιστές μετατρέπονται από μεμονωμένες μονάδες σε μέρη ενός ευρύτερου συνόλου. Αιτία αυτής της εξάπλωσης είναι η διαρκής ανάγκη για γρηγορότερη και πιο αποτελεσματική διακίνηση της πληροφορίας.

Η σύνδεση των τερματικών ενός δικτύου μπορεί να είναι ενσύρματη ή ασύρματη. Ένα ασύρματο δίκτυο δεν χρησιμοποιεί καλώδια για τις συνδέσεις των υπολογιστών. Αντί του καλωδίου χρησιμοποιείται μετάδοση μέσω ειδικά διαμορφωμένων οπτικών, υπέρυθρων ή ραδιοκυματικών σημάτων. Η έννοια του ασύρματου δικτύου ορίζεται ως εξής:

**Ορισμός:** Ως ασύρματο δίκτυο χαρακτηρίζεται το τηλεπικοινωνιακό δίκτυο, συνήθως τηλεφωνικό ή δίκτυο υπολογιστών, το οποίο χρησιμοποιεί ραδιοκύματα ως φορείς πληροφορίας. Τα δεδομένα μεταφέρονται μέσω ηλεκτρομαγνητικών κυμάτων, με συχνότητα φέροντος.



Σχήμα 2.1. Ασύρματο Δίκτυο

Η συνηθισμένη ακτίνα δράσης ενός ασύρματου δικτύου εκτείνεται σε αρκετά μέτρα, η οποία είναι ικανή να διασυνδέσει από τους ορόφους μιας πολυκατοικίας μέχρι τα κτήρια μιας πανεπιστημιούπολης.(Σχήμα 2.1). Η

διασύνδεση ενός ασύρματου τοπικού δικτύου με ένα αντίστοιχο ενσύρματο μπορεί να αυξήσει σημαντικά την απόσταση μετάδοσης του ενσύρματου δικτύου. Τα ασύρματα δίκτυα τοπικής περιοχής, συχνά αναφέρονται ως WLAN(Wireless Local Area Network ) ή Wi-Fi δίκτυα.

Ο τομέας των ασύρματων δικτύων είναι ένας από τους ταχύτερα αναπτυσσόμενους κλάδους των τηλεπικοινωνιών και αποτελεί ένα πεδίο εξαιρετικά πρόσφορο για έρευνα. Η εξέλιξη των ασύρματων δικτύων ακολουθεί αλματώδεις ρυθμούς, καθιστώντας τα διαρκώς πιο αποτελεσματικά και επικερδή. Οι άνθρωποι πλέον τα εγκαθιστούν όλο και περισσότερο στα γραφεία τους, στα ξενοδοχεία, τις καφετέριες και τα σπίτια τους. Τα ασύρματα δίκτυα προσφέρουν εμπιστευτικότητα και κινητικότητα. Πολλές φορές η ασύρματη δικτύωση μπορεί να είναι και λιγότερο δαπανηρή για την εφαρμογή από ότι η ενσύρματη.

Η παρούσα πτυχιακή εργασία μελετά το πρωτόκολλο IEEE 802.11 των ασύρματων τοπικών δικτύων και ειδικότερα ερευνά την απόδοση της κατανεμημένης διαδικασίας πρόσβασης DCF που χρησιμοποιεί τον αλγόριθμο της δυαδικής εκθετικής οπισθοχώρησης BEB (Binary Exponential Backoff) του υποεπιπέδου MAC του IEEE 802.11 για μετρήσεις της ρυθμοαπόδοσης και καθυστερήσεων των πακέτων.

## **2.1 Πλεονεκτήματα ασύρματων δικτύων**

Τα ασύρματα δίκτυα παρουσιάζουν διάφορα πλεονεκτήματα όταν συγκρίνονται με τα ενσύρματα τοπικά δίκτυα. Σε ένα ασύρματο δίκτυο είναι πιο εύκολο να προσθέσουμε ή να μετακινήσουμε ένα σταθμό ή να εγκαταστήσουμε ένα σημείο πρόσβασης, για να παρέχουμε συνδεσιμότητα σε περιοχές που είναι δύσκολο να καλωδιωθούν. Τα πλεονεκτήματα των ασύρματων δικτύων παρατίθενται παρακάτω:

- Ευκολία (Convenience): Η ασύρματη φύση αυτών των δικτύων επιτρέπει στους χρήστες να έχουν πρόσβαση στους πόρους ενός δικτύου, από σχεδόν οποιαδήποτε τοποθεσία χωρίς να πρέπει να βρίσκονται στο σπίτι ή στο

γραφείο. Με την αύξηση της χρησιμοποίησης φορητών υπολογιστών, αυτό είναι ιδιαίτερα σημαντικό.

- Κινητικότητα (mobility): Τα WLAN μπορούν να παρέχουν την δυνατότητα στους χρήστες για πρόσβαση σε πληροφορίες ενώ βρίσκονται εν κίνηση εντός της εμβέλειάς τους. Αυτή η ευχέρεια στην κίνηση υποστηρίζει την παραγωγικότητα και τις ευκαιρίες για εξυπηρέτηση, οι οποίες δεν είναι δυνατές με τα ενσύρματα δίκτυα.
- Ταχύτητα και ευελιξία εγκατάστασης: Η εγκατάσταση ενός WLAN εξαλείφει την ανάγκη της χρήσης των καλωδίων, η οποία συνήθως απαιτεί κόπο και χρόνο, ενώ η ασύρματη τεχνολογία επιτρέπει τη διασύνδεση δικτύων η οποία υπό άλλες συνθήκες θα ήταν αδύνατη.
- Μειωμένο κόστος κτήσης: Ενώ η αρχική επένδυση που απαιτείται για τον εξοπλισμό σε ένα WLAN μπορεί σε μερικές περιπτώσεις να είναι υψηλότερη από το αντίστοιχο κόστος για μια ενσύρματη σύνδεση, το συνολικό κόστος λειτουργίας μπορεί να είναι σημαντικά χαμηλότερο καθώς τα μακροπρόθεσμα κέρδη είναι πολύ μεγαλύτερα σε δυναμικά περιβάλλοντα όπου απαιτούνται πολύ συχνές μετακινήσεις και αλλαγές.
- Συμβατότητα: Τα WLAN μπορούν να διαμορφωθούν σε διάφορους τύπους δικτύων ,ανάλογα με τις ανάγκες συγκεκριμένων εγκαταστάσεων και εφαρμογών. Οι διαμορφώσεις αλλάζουν εύκολα και επεκτείνονται από μικρά δίκτυα κατάλληλα για ένα μικρό αριθμό χρηστών μέχρι πλήρως ανεπτυγμένα δίκτυα που καλύπτουν εκατοντάδες χρήστες.

## **2.2 Μειονεκτήματα ασύρματων δικτύων**

Η χρήση των ηλεκτρομαγνητικών κυμάτων για την μετάδοση των σημάτων κάνουν τα WLAN ευπαθή σε πολλά φαινόμενα παρεμβολής τα οποία αλλοιώνουν σε μικρότερο ή μεγαλύτερο βαθμό την επικοινωνία των ασύρματων χρηστών.

Τα κυριότερα μειονεκτήματα που παρουσιάζουν είναι:

- Παρεμβολή λόγω πολλαπλών διαδρομών: Τα μεταδιδόμενα σήματα μπορούν να συνδυαστούν με τα ανακλώμενα από διάφορες επιφάνειες ή εμπόδια με αποτέλεσμα την φθορά ή την καταστροφή του σήματος που

ανιχνεύεται από τον δέκτη. Το φαινόμενο αυτό είναι γνωστό ως “παρεμβολή λόγω πολλαπλών διαδρομών” ή “πολύοδη διάδοση”.

- Path Loss: Το φαινόμενο path loss μεταξύ πομπού και δέκτη είναι ένα από τα σημαντικότερα στοιχεία που πρέπει να ληφθούν υπόψη κατά τον σχεδιασμό ενός WLAN. Τα αναμενόμενα επίπεδα του path loss, τα οποία βασίζονται στην απόσταση μεταξύ του πομπού και του δέκτη, παρέχουν πολύτιμες πληροφορίες για τον καθορισμό των επιπέδων στην ισχύ εκπομπής, στην ευαισθησία του δέκτη και στον λόγο σήματος προς θόρυβο (SNR). Το πραγματικό path loss εξαρτάται από τη συχνότητα μετάδοσης και αυξάνει εκθετικά με την αύξηση της απόστασης μεταξύ του πομπού και του δέκτη.
- Παρεμβολές ραδιοσημάτων: Η διαδικασία της εκπομπής και λήψης ραδιοσημάτων μέσω του αέρα καθιστά τα ασύρματα συστήματα ευπαθή από το θόρυβο της ατμόσφαιρας και από τις μεταδόσεις άλλων συστημάτων που λειτουργούν στην ίδια μπάντα συχνοτήτων και λειτουργούν στον ίδιο φυσικό χώρο.
- Διαχείριση ενέργειας: Οι περισσότερες WLAN συσκευές από την πλευρά του χρήστη λειτουργούν με μπαταρίες που έχουν καθορισμένη διάρκεια ζωής. Η χρήση τους σε αυτές τις τηλεπικοινωνιακές εφαρμογές μειώνουν την αυτονομία τους.
- Ασυμβατότητα συστημάτων: Στην κατασκευή ενός WLAN θα πρέπει να ληφθεί υπόψη η ασυμβατότητα μεταξύ προϊόντων διαφορετικών κατασκευαστών, διαφορετικά το δίκτυο δεν θα λειτουργεί σωστά. Λόγοι ασυμβατότητας είναι οι εξής: χρήση διαφορετικής τεχνολογίας, η χρήση διαφορετικού φάσματος συχνοτήτων και η διαφορετική υλοποίηση.
- Ασφάλεια δικτύου: Η λειτουργία ενός ασύρματου δικτύου αντιστοιχεί στα χαμηλότερα επίπεδα της αρχιτεκτονικής ενός δικτύου και δεν εμπεριέχει άλλες λειτουργίες όπως εγκατάσταση σύνδεσης από άκρο σε άκρο ή άλλες υπηρεσίες (πχ login), που προσφέρουν τα ανώτερα στρώματα. Για τον λόγο αυτόν το μόνο θέμα που σχετίζεται με την ασφάλεια και απασχολεί τα ασύρματα δίκτυα έχει να κάνει με θέματα ασφαλείας χαμηλότερων στρωμάτων όπως η κρυπτογράφηση των δεδομένων. Έχουν υλοποιηθεί διάφορες τεχνικές κωδικοποίησης, οι οποίες καθιστούν εξαιρετικά δύσκολη τη

λήψη της μεταδιδόμενης πληροφορίας από κάποιον χρήστη πέραν του προοριζόμενου.

- Πρόβλημα του κρυμμένου κόμβου: Ένας συνηθισμένος περιορισμός στην απόδοση των WLAN είναι το πρόβλημα που προκύπτει από την περιορισμένη ακτίνα δράσης των ραδιοκυμάτων και είναι γνωστό ως “hidden node problem”. Το φαινόμενο αυτό προκύπτει όταν στο σύστημα υπάρχει ένας σταθμός που δεν μπορεί να ανιχνεύσει την μετάδοση ενός άλλου σταθμού ώστε να αναγνωρίσει ότι το μέσο χρησιμοποιείται (βλέπε το πρόβλημα κρυμμένου τερματικού).

## **2.3 Κατηγορίες Ασύρματων Δικτύων.**

Τα ασύρματα δίκτυα μπορούν να κατηγοριοποιηθούν ανάλογα με την περιοχή οπoία καλύπτουν .

**2.3.1 Ασύρματα σωματικά δίκτυα (Wireless Body Area Network - WBANs).** Τα WBAN δίκτυα αποτελούνται από αισθητήρες που είναι εμφυτευμένοι ή τοποθετημένοι πάνω στο σώμα ή στα ρούχα ενός ανθρώπου και επιβλέπουν διάφορες ζωτικές λειτουργίες του ανθρώπινου οργανισμού ή μετρούν διάφορες δείκτες υγείας . Το πρωτόκολλο 802.15.6 προορίζεται για το WBAN δίκτυα. (IEEE 802.15 TG6 2007)

**2.3.2 Ασύρματα προσωπικά δίκτυα.(Wireless Personal Networks - WPANs).** Τα WPAN είναι δίκτυα μικρής εμβέλειας, εξυπηρετούν έναν χρήστη και καλύπτουν χώρο π.χ ενός γραφείου ή ενός δωματίου και συνολικά απόσταση 10 μέτρων. Το bluetooth είναι WPAN δίκτυο και ορίζεται από το πρωτόκολλο IEEE 802.15.1. (IEEE 802.15.1 TG1a 2007)

**2.3.3 Ασύρματα Τοπικά Δίκτυα(Wireless Local Area Network - WLANs).** Τα WLAN δίκτυα έχουν μεγαλύτερη εμβέλεια από τα WPAN και συνήθως καλύπτουν χώρους ενός ορόφου ή μιας οικίας . Τα WLAN είναι ο πλέον δημοφιλής τρόπος ασύρματης επικοινωνίας δεδομένων . Το πρωτόκολλο που έχει επικρατήσει στα WLAN είναι το IEEE 802.11.

**2.3.4 Ασύρματα Μητροπολιτικά Δίκτυα (Wireless Metropolitan Area Network - WMANs)** Τα WMAN δίκτυα καλύπτουν γεωγραφικές περιοχές όπως μικρών πόλεων ή αποστάσεις μερικών χιλιομέτρων. Το WMAN στην Βόρειο Αμερική χρησιμοποιούν το πρωτόκολλο IEEE 802.16(τεχνολογία WiMAX).

**2.3.5 Ασύρματα δίκτυα ευρείας περιοχής (Wireless Wide Area Network - WWANs).** Τα WWAN καλύπτουν περιοχές μιας ή περισσότερων χωρών και είναι ευρέως διαδεδομένα στην κινητή τηλεφωνία και παρέχουν δυνατότητα μεταφοράς δεδομένων.

## **2.4 Η Οικογένεια των Πρωτοκόλλων IEEE 802.11.**

### **2.4.1 IEEE 802.11**

Δημοσιεύτηκε το 1997 από την IEEE, μετά από επτά χρόνια μελέτης, είναι το πρώτο πρότυπο για ασύρματη δικτύωση. Προβλέπει ρυθμούς μετάδοσης από 1 έως 2 Mbps. Στο φυσικό επίπεδο προβλέπει τεχνική FHSS (Frequency Hopping Spread Spectrum) ή DSSS (Direct-Sequence Spread Spectrum) σε ζώνες συχνοτήτων 915 MHz, 2.4 GHz και 5.2 GHz ή υπέρυθρη μετάδοση στα 850 nm έως 900 nm. Το πρότυπο γνώρισε περιορισμένη επιτυχία λόγω των χαμηλών ρυθμών μετάδοσης.

### **2.4.2 IEEE 802.11a**

Το πρότυπο αυτό εισήλθε στην αγορά αφού το 802.11b είχε ήδη ένα μεγάλο μερίδιο αυτής. Χρησιμοποιεί τα 5GHz ως συχνότητα εκπομπής με μέγιστη μεταφορά δεδομένων 54 Mbps . Επιλέχθηκε η λειτουργία σε μια υψηλότερη ζώνη συχνοτήτων, για να μπορούν να υποστηριχθούν οι μεγαλύτεροι ρυθμοί μετάδοσης.

### **2.4.3 IEEE 802.11b**

Η μετάδοση γίνεται στη ζώνη συχνοτήτων των 2.4 GHz αλλά με μέγιστη μεταφορά δεδομένων 11 Mbit/s και μέση ρυθμοαπόδοση 4 Mbit/s . Είναι το πιο δημοφιλές από όλα τα πρότυπα και το πρότυπο με τη μεγαλύτερη

διαλειτουργικότητα, όντας ένα στιβαρό, αποτελεσματικό και δοκιμασμένο πρότυπο. Οι προσθήκες του 802.11b σε σχέση με το 802.11 αφορούν μόνο τον τρόπο μετάδοσης ενώ ο τρόπος πρόσβασης των συσκευών και ο τρόπος λειτουργίας μένουν οι ίδιοι.

#### **2.4.4 IEEE 802.11g**

Αποτελεί επέκταση στο 802.11b, ώστε να υποστηρίζει μεγαλύτερους ρυθμούς μετάδοσης. Έτσι εκτός από τους ρυθμούς μετάδοσης του 802.11b, υποστηρίζει και ρυθμούς μέχρι και 54 Mbps. Οι αντίστοιχες συσκευές εργάζονται στη ζώνη συχνοτήτων των 2.4 GHz, διατηρώντας συμβατότητα προς τα πίσω με το 802.11b.

Πέρα των βασικών πρωτοκόλλων η οικογένεια προτύπων 802.11, περιλαμβάνει έναν αριθμό συμπληρωματικών προτύπων, που προσθέτουν επιπλέον λειτουργικότητα στα ασύρματα δίκτυα.

#### **2.4.5 IEEE 802.11c**

Παρέχει λειτουργία γεφύρωσης (bridging) 802.11 πλαισίων.

#### **2.4.6 IEEE 802.11d**

Παρέχει επεκτάσεις στο φυσικό επίπεδο, ώστε να λειτουργεί σε επιπλέον ρυθμιστικά πλαίσια (άλλες ζώνες συχνοτήτων).

#### **2.4.7 IEEE 802.11e**

Υποστήριξη QoS στο MAC επίπεδο (EDCF, Enhanced DCF και HCF) και ενίσχυση των μηχανισμών ασφάλειας.

#### **2.4.8 IEEE 802.11f**

Συνιστώμενη πρακτική για το πρωτόκολλο IAPP, Inter Access Point Protocol, που αφορά την επικοινωνία μεταξύ των σημείων πρόσβασης.

#### **2.4.9 IEEE 802.11i**

Επεκτάσεις στο MAC επίπεδο για ενισχυμένη ασφάλεια. Περιγραφή πρωτοκόλλων 802.1X, TKIP, AES.

#### **2.4.10 IEEE 802.11j**



Παρέχει ενίσχυση στο φυσικό επίπεδο του μηχανισμού IEEE 802.11a, ώστε να προσαρμοστεί με τις Ιαπωνικές απαιτήσεις.

#### **2.4.11 IEEE 802.11k**

Βελτιώσεις στην μέτρηση των πόρων του ραδιοφώνου, για την παροχή διασύνδεσης στα υψηλότερα επίπεδα για μετρήσεις δικτύων.

#### **2.4.12 IEEE 802.11m**

Συντήρηση του IEEE 802.11-1999 προτύπου, με τεχνικές και συντακτικές διορθώσεις.

#### **2.4.13 IEEE 802.11n**

Ενίσχυση στο φυσικό επίπεδο και στο επίπεδο MAC για την επίτευξη υψηλότερου ρυθμού μετάδοσης δεδομένων.

#### **2.4.14 IEEE 802.11p**

Στο φυσικό επίπεδο και στο MAC , παροχή ασύρματης πρόσβασης σε περιβάλλοντα που βρίσκονται σε τροχιά μεταξύ τους.

#### **2.4.15 IEEE 802.11r**

Στο φυσικό και στο MAC επίπεδο παρέχει γρήγορη περιαγωγή (γρήγορη μετάβαση BSS).

#### **2.4.16 IEEE 802.11s**

Παρέχει δικτύωση των ESS.

#### **2.4.17 IEEE 802.11,2**

Συνιστώμενη πρακτική για την αξιολόγηση της ασύρματης απόδοσης των 802.11.

#### **2.4.18 IEEE 802.11u**

Βοηθάει στη συνεργασία με διάφορα εξωτερικά δίκτυα.

## **2.5 Η πιστοποίηση Wi-Fi**

Με την ανάπτυξη των προτύπων από την IEEE και την εμφάνιση μεγάλου αριθμού κατασκευαστών, αντίστοιχων συσκευών, εμφανίστηκε από νωρίς η ανάγκη διασφάλισης της συμβατότητας μεταξύ των διαφόρων συσκευών και προστασίας του καταναλωτή. Γι' αυτό το σκοπό ιδρύθηκε το 1999 η WECA (Wireless Ethernet Compatibility Alliance), ένας μη κερδοσκοπικός οργανισμός, που σκοπό έχει την πιστοποίηση των 802.11 ασύρματων

συσκευών. Αυτός ο οργανισμός στη συνέχεια μετονομάστηκε σε Wi-Fi (Wireless Fidelity) Alliance και δημιούργησε ένα πακέτο από δοκιμές, για να πιστοποιεί τη διαλειτουργικότητα των 802.11b προϊόντων.

Σε αυτό τον οργανισμό μετέχουν κατασκευαστές ολοκληρωμένων κυκλωμάτων, παροχής υπηρεσιών WLAN, κατασκευαστές υπολογιστών και λογισμικού. Μερικές από τις εταιρίες που μετέχουν είναι η 3COM, Aironet, Apple, Breezecom, Cabletron, Compaq, Dell, Fujitsu, IBM, Intersil, Lucent Technologies, No Wires Needed, Nokia, Samsung, Symbol Technologies, Wayport, Zoom.



Σχήμα2.2 Λογότυπο πιστοποίησης

Η ένωση αυτή δημιούργησε μια ακολουθία από δοκιμές, προκειμένου να δοκιμαστεί η συμβατότητα των IEEE 802 προϊόντων. Οι συσκευές που περνούσαν με επιτυχία τις δοκιμές, αποκτούσαν το λογότυπο Wi-Fi (Wireless Fidelity) που αποτελεί κατά συνέπεια μια πιστοποίηση, για τον υποψήφιο αγοραστή μιας συσκευής και μια εγγύηση της επένδυσής του. Ο καταναλωτής αγοράζοντας μια συσκευή με το λογότυπο Wi-Fi, έχει την εγγύηση ότι η συσκευή θα συνεργαστεί με οποιαδήποτε άλλη συσκευή που φέρει το ίδιο λογότυπο.

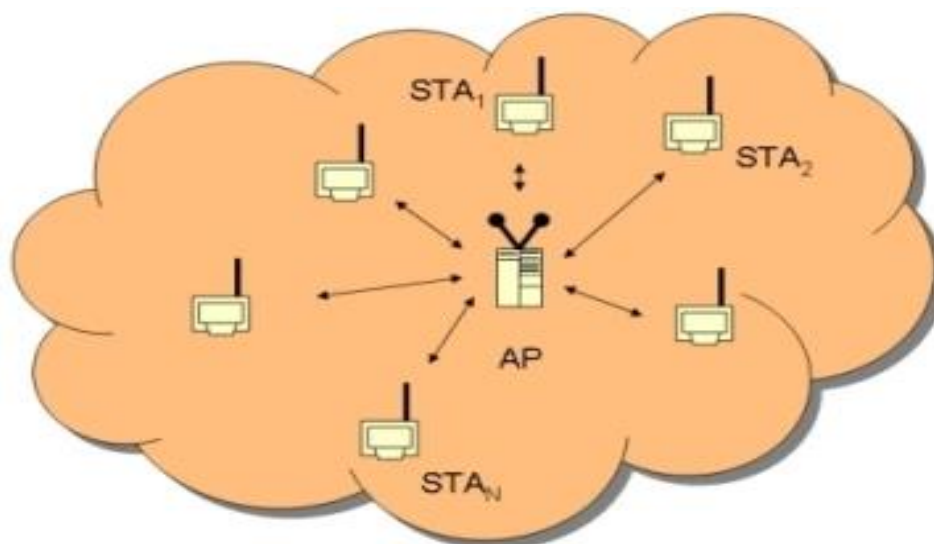
Τέλος να αναφερθεί ότι η πιστοποίηση αφορά τη λειτουργία 802.11b, 802.11g, 802.11a καθώς και τη WPA δυνατότητα. Να σημειωθεί ότι η Wi-Fi πιστοποίηση στο 802.11g απαιτεί την υποστήριξη του ρυθμού 54 Mbps, ενώ το επίσημο πρότυπο θέτει σαν υποχρεωτικούς ρυθμούς 1, 2, 5.5, 11, 6, 12, 24 Mbps, καθώς οι ανώτεροι ρυθμοί των 36, 48, 54 Mbps ορίζονται ως προαιρετικοί. Η διαδικασία πιστοποίησης που έχει αναπτυχθεί για τα 802.11a προϊόντα, ονομάζεται Wi-Fi5.

## 2.6 Τοπολογία ασύρματων δικτύων.

Το πρότυπο 802.11 ορίζει τρεις τρόπους επικοινωνίας μεταξύ κόμβων ενός δικτύου, τον BSS (Basic Service Set), τον IBSS (Independent Basic Service Set) και τον ESS (Extended Service Set).

### 2.6.1 BSS (Basic Service Set)

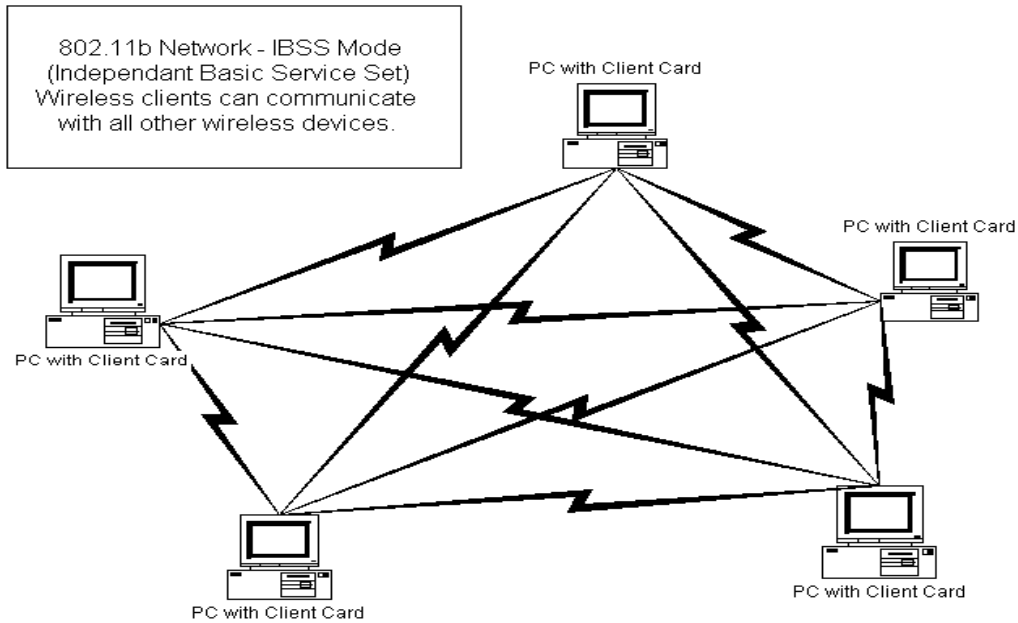
Το βασικό δομικό στοιχείο ενός IEEE 802.11 είναι το BSS (Basic Service Set). Ένα BSS μπορεί να αποτελείται για παράδειγμα από δύο ή περισσότερους σταθμούς οι οποίοι επικοινωνούν μέσω Access Point (σημείο πρόσβασης), όπως φαίνεται στην παρακάτω εικόνα, οι οποίοι είναι τα μέλη του. Αν ένας σταθμός μετακινηθεί έξω από το BSS στο οποίο ανήκει, δεν μπορεί πλέον να επικοινωνήσει άμεσα με τα μέλη του συγκεκριμένου BSS.



Σχήμα 2.3 Basic Service Set

### 2.6.2 IBSS (Independent Basic Service Set)

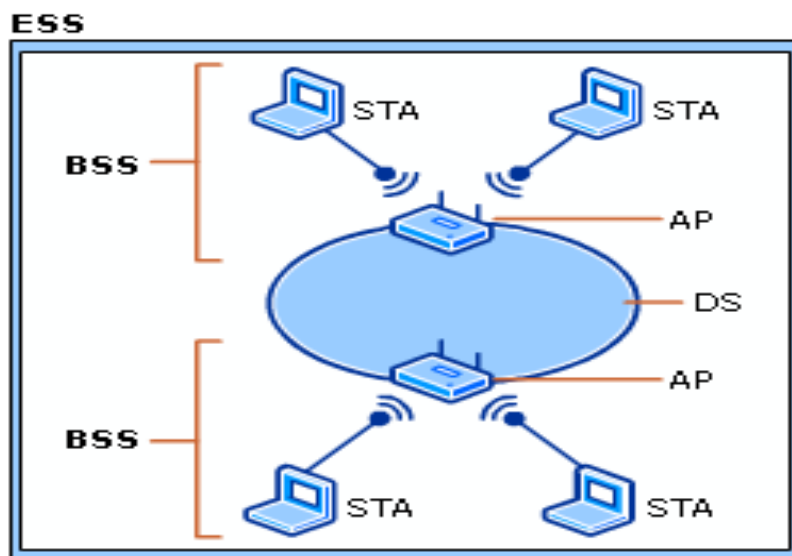
Είναι η πιο βασική και η πιο απλή τοπολογία ασύρματης δικτύωσης. Οι ασύρματοι σταθμοί επικοινωνούν κατευθείαν μεταξύ τους ένας προς έναν (peer-to-peer), χωρίς να υπάρχει κεντρικός σταθμός (AP). Όλοι οι σταθμοί είναι ισότιμοι μεταξύ τους. Στο παρακάτω σχήμα φαίνεται ένα δίκτυο IBSS ή όπως αλλιώς ονομάζονται Ad-hoc δίκτυα ή Peer-to-Peer.



Σχήμα 2.4 Independent Basic Service Set

### 2.6.3 ESS (Extended Service Set)

Όταν οι ανάγκες της διαδικτύωσης ξεπερνούν τα όρια του IBSS, το 802.11 καθορίζει τη δομή ενός πιο σύνθετου τοπικού δικτύου που ονομάζεται ESS (Extended Service Set) και στο οποίο είναι δυνατή η διασύνδεση και η επικοινωνία πολλών BSS μεταξύ τους. Το στοιχείο που χρησιμοποιείται για την διασύνδεση των BSS ονομάζεται σύστημα διανομής (Distributed System-DS). Το 802.11 κάνει διαχωρισμό του ασύρματου μέσου (wireless medium) από το DS.



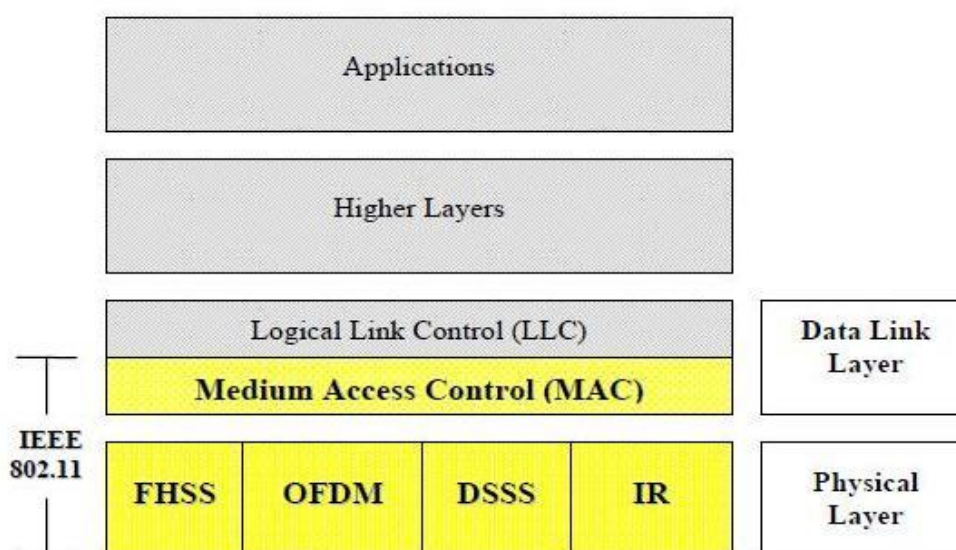
Σχήμα 2.5 Extended Service Set

Η πρόσβαση στο DS γίνεται με τη βοήθεια ενός σταθμού που καλείται AP (Access Point) και ο οποίος παρέχει ουσιαστικά τη διασύνδεση των σταθμών που βρίσκονται σε διάφορα BSS στο DS. Η διασύνδεση αυτή φαίνεται στο Σχήμα 2.5.

## 2.7 Διαστρωμάτωση του IEEE 802.11

Ο σχεδιασμός πρωτοκόλλων των ασύρματων τοπικών δικτύων χρησιμοποιεί αρχές των πρωτοκόλλων που έχουν αναπτυχθεί για τα ευρέως χρησιμοποιούμενα ενσύρματα τοπικά δίκτυα και περιλαμβάνει την ανάπτυξη μονό λίγων στρωμάτων.

Το IEEE 802.11 ορίζει το φυσικό στρώμα (Physical Layer -PHY) και το υπόστρωμα ελέγχου πρόσβασης μέσου (Medium Access Control - MAC) του στρώματος ζεύξης δεδομένων (Data Link Layer - DLL) . Το σχήμα 2.6 απεικονίζει την διαστρωμάτωση του πρωτοκόλλου IEEE 802.11 .Το υπόστρωμα ελέγχου λογικής ζεύξης (Logical Link Control - LLC) του στρώματος DLL ορίζεται από το πρότυπο IEEE 802.2 και χρησιμοποιείται για να προσαρμόσει (interface) ομοιόμορφα όλα τα διαφορετικά MAC (Ethernet , Token Ring , 802.11 , κτλ) με τα υψηλότερα στρώματα . (G. Bianchi , Π. Ράπτης 2010)



Σχήμα 2.6 Διαστρωμάτωση του IEEE 802.11

### 2.7.1 Φυσικό Στρώμα (PHY)

Το φυσικό επίπεδο καθορίζει μεθόδους και διαδικασίες εκπομπής και λήψης δεδομένων μεταξύ των ασύρματων σταθμών όπως είναι η συχνότητα εκπομπής, ο τρόπος διαμόρφωσης των σημάτων, οι κώδικες εντοπισμού και διόρθωσης λαθών. Στο φυσικό επίπεδο προστίθενται οι κεφαλίδες του PHY στις πληροφορίες που προέρχονται από τα υψηλότερα επίπεδα (στρώματα) και σχηματίζονται πακέτα έτοιμα για εκπομπή, και αντίστοιχα στο φυσικό επίπεδο των δεκτών γίνεται ο διαχωρισμός των κεφαλίδων του PHY και των πληροφοριών των υψηλότερων επιπέδων.

Η ασύρματη επικοινωνία πραγματοποιείται με τη χρήση ραδιοκυμάτων, ή υπέρυθρων ακτίνων. Το φυσικό επίπεδο του IEEE 802.11 ορίζει τους παρακάτω τρόπους διαμόρφωσης των σημάτων:

- Διασπορά φάσματος και εναλλαγή συχνότητας (Frequency Hopping Spread Spectrum - FHSS). Στην FHSS το σήμα δεδομένων διαμορφώνεται με ένα στενής-ζώνης φερόμενο σήμα που αλλάζει συχνότητα εκπομπής τυχαία αλλά με γνωστή σειρά πάνω σε μια ευρεία ζώνη συχνοτήτων. Η μέθοδος FHSS χρησιμοποιεί την περιοχή συχνοτήτων 2.4 – 2.4835 GHz και ταχύτητα εκπομπής δεδομένων είναι 1 Mbit/s και 2 Mbit/s.
- Εξάπλωση φάσματος ευρείας ακολουθίας (Direct sequence Spread Spectrum - DDSS). Στην DDSS το σήμα δεν αλλάζει συχνότητα εκπομπής αλλά περνά από μια διαδικασία εξάπλωσης και κατανέμεται σε όλη την ζώνη συχνοτήτων με μιας. Η μέθοδος DDSS χρησιμοποιεί την περιοχή συχνοτήτων 2.4 - 2.4835 GHz και οι ταχύτητες εκπομπής δεδομένων είναι 1 Mbit/s και 2 Mbit/s.
- Υπέρυθρες ακτίνες (Infrared - IR) Η IR είναι εκπομπή ηλεκτρομαγνητικής ακτινοβολίας σε μήκη κύματος από 750 nm έως 100μm. Οι IR ακτίνες έχουν μικρή διάχυση (μετάδοση με οπτική επαφή), δεν μπορούν να διαπεράσουν τοίχους, για αυτόν τον λόγο επικράτησαν οι μέθοδοι με ραδιοκύματα. Ταχύτητα εκπομπής δεδομένων 1 Mbit/s και 2 Mbit/s.
- Πολυπλεξία συχνοτήτων ορθογωνίων φερουσών (Orthogonal Frequency Division Multiplexing – OFDM). Στην μέθοδο διαμόρφωσης

OFDM γίνεται αξιοποίηση του διαθέσιμου εύρους συχνοτήτων και της μετάδοσης μέσω ορθογωνίων πολλαπλών φερουσών (sub-carrier) . Η OFDM χρησιμοποιείται από τα πρωτόκολλα IEEE 802.11a/g/n . Το IEEE 802.11a χρησιμοποιεί την περιοχή συχνοτήτων 5GHz με ταχύτητα εκπομπής δεδομένων έως 54 Mbit/s , το IEEE 802.11g την περιοχή 2.4 GHz με ταχύτητα εκπομπής δεδομένων έως 54 Mbit/s και το IEEE 802.11n τις περιοχές 2.4 GHz και 5 GHz με ταχύτητα εκπομπής δεδομένων έως 600 Mbit/s.

### **2.7.2 Υπόστρωμα MAC**

Ο κύριος σκοπός του υποστρώματος MAC είναι να εξασφαλίσει πρόσβαση σε κάθε σταθμό του δικτύου με έναν τρόπο δίκαιο και αποδοτικό . Το πρωτόκολλο IEEE 802.11 στο MAC καθορίζει δυο τρόπους πρόσβασης στο μέσον : α) την διαδικασία DCF (Distributed Coordination Function) , που ορίζει ως τρόπο πρόσβασης τον ανταγωνισμό μεταξύ των κόμβων σε ένα δίκτυο και β) τη προαιρετική διαδικασία PCF (Point coordination Function – την κεντρική διαχείριση πρόσβασης) που ορίζει έναν τρόπο πρόσβασης χωρίς ανταγωνισμό . Η PCF προσφέρει στους σταθμούς πρόσβαση στο μέσον με δίκαιο τρόπο δηλ. οι σταθμοί χρησιμοποιούν το μέσον ανάλογα με τις απαιτήσεις τους . Η διαδικασία PCF δεν έχει εξαπλωθεί ευρέως διότι είναι περίπλοκη , προαιρετική και δεν υποστηρίχθηκε από τους κατασκευαστές . Τα πρότυπα IEEE 802.11a,b,g χρησιμοποιούν το ίδιο MAC υπόστρωμα και έχουν την ίδια διαδικασία πρόσβασης μέσω DCF (το IEEE 802.11e χρησιμοποιεί μια εξελιγμένη DCF).

### **2.8 Τύποι πλαισίων**

Το πρότυπο IEEE 802.11 έχει ορίσει τρεις βασικούς τύπους πλαισίων , τα οποία είναι :

- Πλαίσιο δεδομένων (data frame) , τα οποία χρησιμεύουν για εκπομπή δεδομένων
- Πλαίσια ελέγχου (Control frame) , τα οποία χρησιμεύουν για την πρόσβαση στο μέσον (π.χ RTS,CTS και ACK).

- πλαίσια διαχείρισης (management frame) , είναι πλαίσια τα οποία μεταδίδονται με τον ίδιο τρόπο όπως και τα πλαίσια δεδομένων , για να ανταλλάξουν πληροφορίες διαχείρισης , αλλά δεν προωθούνται στα ανώτερα επίπεδα .

Το καθένα από τους παραπάνω τύπους πλαισίων είναι υποκατηγορίες άλλων πλαισίων , ανάλογα με την συγκεκριμένη λειτουργία που εκτελούν .

## 2.9 Μορφή πλαισίων

Σε αυτό το σημείο θα αναλύσουμε τα βασικά συστατικά του 802.11 πλαισίου. Όλα τα πλαίσια του πρωτοκόλλου IEEE 802.11 συμπεριλαμβάνουν τα παρακάτω πεδία , τα οποία θα τα αναλύσουμε παρακάτω.

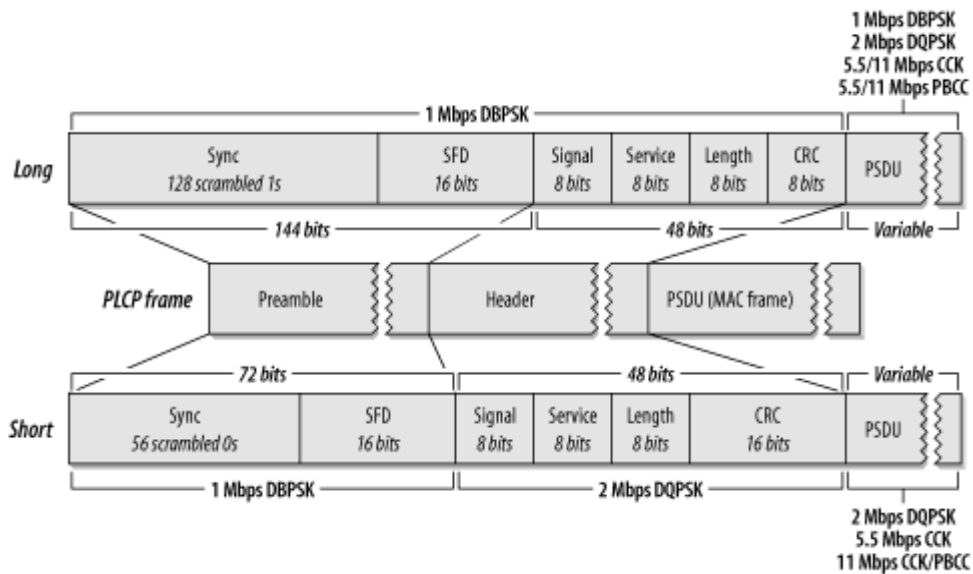
Preamble	PLCP Header	MAC Data	CRC
----------	-------------	----------	-----

### 2.9.1 Preamble.

Είναι το πρώτο πεδίο του PLCP(Physical Layer Convergence Protocol) του PDU(Protocol Data Unit) Κάθε πλαίσιο στέλνεται με ένα πρόθεμα(preamble) το οποίο είναι μια γνώστη ακολουθία από bits στην αρχή κάθε πλαισίου , έτσι ώστε ο παραλήπτης να μπορεί να συγχρονιστεί με τον πομπό και να είναι έτοιμος να δεχτεί πλαίσια δεδομένων . Η ταχύτητα που πρέπει να στέλνεται το πρόθεμα (preamble) είναι 1 Mbps σύμφωνα με το επίσημο πρότυπο , αλλά υπάρχουν δύο διαφορετικά είδη preamble : βραχυπρόθεσμα (short) και μακροπρόθεσμα (long) τα οποία αφορούν το μήκος του πεδίου συγχρονισμού. Το μακροπρόθεσμο πρόθεμα έχει μήκος πεδίου 128 bits ενώ του βραχυπρόθεσμου μόνο 56 bits.

(<http://www.intel.com/support/wireless/wlan/sb/CS-025317.htm>)





Σχήμα 2.7 Long and Short Preamble

### Long Preamble.

- Συμβατό με τα κληροδοτημένα συστήματα του IEEE 802.11 που λειτουργούν σε 1 και 2 Mbps.
- Τα PLCP με long preamble μεταδίδεται σε ταχύτητα 1 Mbps ανεξάρτητα από τον ρυθμό μετάδοσης των πλαισίων δεδομένων.
- Ο συνολικός χρόνος στο long preamble είναι σταθερός στα 192 usec (microseconds)

### Short Preamble:

- Δεν είναι συμβατό με τα κληροδοτημένα συστήματα του IEEE 802.11 που λειτουργούν σε 1 και 2 Mbps
- Τα PLCP με short preamble : το preamble μεταδίδεται με ταχύτητα 1 Mbps και κεφαλίδα με 2 Mbps.
- Ο συνολικός χρόνος στο long preamble είναι σταθερός στα 96 usec (microseconds).

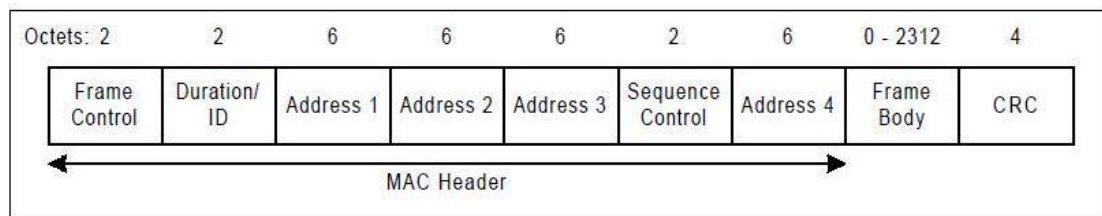
### 2.9.2 PLCP Header .

Το PLCP Header μεταδίδεται πάντα με ταχύτητα 1 Mbps και περιέχει λογικές πληροφορίες που θα χρησιμοποιηθεί από το φυσικό επίπεδο για να αποκωδικοποιεί τα πλαίσια και αποτελείται από:

- **PLOP\_PUT Length Word:** το οποίο αντιπροσωπεύει των αριθμό των bytes στο πακέτο, και είναι χρήσιμο για το φυσικό επίπεδο να καθορίσει το τέλος του πακέτου.
- **PLOP\_Signaling Field :** το οποίο περιέχει μόνο πληροφορίες για το ρυθμό μετάδοσης , κωδικοποιεί σε 0.5 Mbps προσαυξανόμενο από 1 Map/s σε 4.5 Map/s
- **Header Error Check Field :** το οποίο είναι ένα πεδίο εντοπισμού λαθών μήκους 16 bits.

### 2.9.3 MAC Data

Στο σχήμα 2.8 φαίνεται η μορφή του MAC πλαισίου.



Σχήμα 2.8 Μορφή του MAC πλαισίου

### 2.9.4 CRC

Το μέγεθος του CRC είναι 32 bit και περιέχει ένα 32-bit Cyclic Redundancy Check.

## 2.10 Χρονοθυρίδα και χρόνοι αναμονής

Η χρονοθυρίδα (slot time) είναι μια σταθερή χρονική διάρκεια που ορίζεται από το φυσικό στρώμα ανάλογα με την μέθοδο διαμόρφωσης ,για την FSSS μέθοδο έχει διάρκεια 50 μs και για την DSSS 20 μs .Μια χρονοθυρίδα είναι η χρονική διάρκεια που χρειάζεται ένας σταθμός να καταλάβει ότι ένας άλλος σταθμός εξέπεμψε .

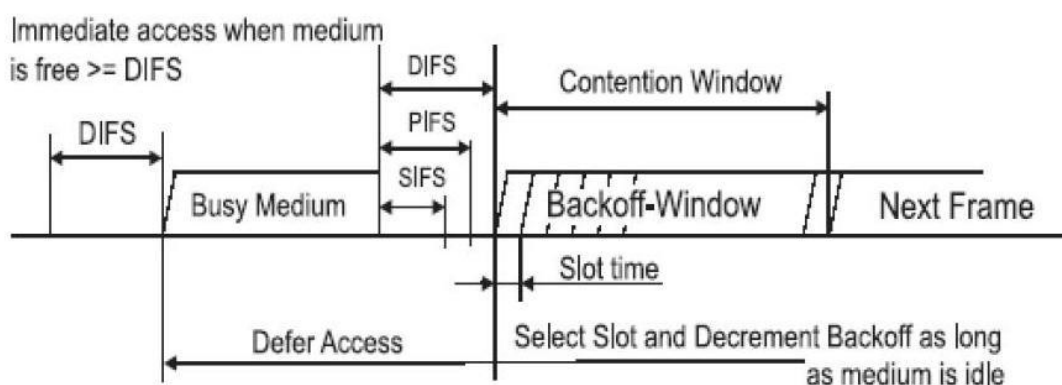
Οι δυο αλγόριθμοι DCF και PCF χρησιμοποιούν διάφορους χρόνους αναμονής (Interframe Space - IFS) για τον έλεγχο πρόσβασης στο μέσον. Κάθε σταθμός όταν πρόκειται να μεταδώσει ένα πλαίσιο ελέγχου ή δεδομένα πρέπει να περιμένει ένα χρονικό διάστημα και αν δεν ανιχνεύσει άλλη μετάδοση σε αυτό το χρονικό διάστημα τότε μπορεί να προχωρήσει στην

εκπομπή του. Το 802.11 χρησιμοποιεί τέσσερις τύπους χρόνων αναμονής ανάλογα με την διαδικασία DCF ή PCF που χρησιμοποιείται:

- α) SIFS – Short interframe space
- β) PIFS – PCF interframe space
- γ) DIFS – DCF interframe space
- δ) EIFS – Extended interframe space

Στο σχήμα 2.9 παρουσιάζεται η σχέση που έχουν οι χρόνοι IFS μεταξύ τους.

- ο χρόνος SIFS είναι ο ελάχιστος χρόνος αναμονής και χρησιμοποιείται για εκπομπές υψηλής προτεραιότητας όπως πλαίσια ελέγχου RTS, CTS και πλαίσια επιβεβαίωσης ACK. Η χρονική διάρκεια του SIFS είναι σταθερή και εξαρτάται από το φυσικό στρώμα. Το SIFS στο 802.11b είναι 10  $\mu$ s, στο 802.11a είναι 16  $\mu$ s, και στο 802.11g είναι 10  $\mu$ s.
- Ο χρόνος PIFS χρησιμοποιείται από την διαδικασία PCF ή τον AP για να έχουν μεγαλύτερη προτεραιότητα πρόσβασης στο μέσον σε σχέση με τους σταθμούς που χρησιμοποιούν την διαδικασία DCF και υπολογίζεται ως  $PIFS = SIFS + \text{slot time}$ .
- Ο χρόνος DIFS χρησιμοποιείται από την διαδικασία DCF και είναι ο ελάχιστος χρόνος αναμονής για να αρχίσει ο ανταγωνισμός διεκδίκησης του μέσου και υπολογίζεται ως  $DIFS = SIFS + 2 * (\text{slot time})$ .
- Ο χρόνος EIFS χρησιμοποιείται όταν συμβαίνει ένα λάθος στην εκπομπή ενός πλαισίου και υπολογίζεται ως  $EIFS = SIFS + (\text{διάρκεια του ACK}) + \text{slot time}$ .



Σχήμα 2.9 Χρόνοι αναμονής και η σχέση μεταξύ τους .

Στην διαδικασία DCF ένας σταθμός πριν εκπέμψει ανιχνεύει το μέσον εάν είναι ενεργό δηλ. εάν υπάρχει εκπομπή από άλλον σταθμό ή είναι ανενεργό. Εάν το μέσον είναι ανενεργό για χρονική διάρκεια DIFS τότε ο σταθμός εκπέμπει .

Ο χρόνος PIFS χρησιμοποιείται από τον AP για αν αποκτήσει πρόσβαση στο μέσο. Καθώς ο χρόνος PIFS είναι μικρότερος από τον DIFS ο AP καταλαμβάνει το μέσον πριν από κάθε άλλον σταθμό που χρησιμοποιεί την διαδικασία DCF .Επειδή ο χρόνος που μεσολαβεί μεταξύ ενός πλαισίου δεδομένων και της αποστολής του πλαισίου επιβεβαίωσής του (ACK) από τον παραλήπτη είναι SIFS και επειδή ο χρόνος SIFS είναι μικρότερος από τον PIFS και τον DIFS ο παραλήπτης έχει μεγαλύτερη προτεραιότητα και από τον AP και από άλλους σταθμούς στο δίκτυο .

## 2.11 Τρόποι Έρευνας της Απόδοσης

Η έρευνα για την απόδοση του πρωτοκόλλου IEEE 802.11 μπορεί να γίνει με χρήση προσομοιώσεων και αναλυτικών (μαθηματικών) μοντέλων , πειραμάτων .

- **Προσομοίωση** . Η προσομοίωση έχει χρησιμοποιηθεί ευρέως στην μελέτη της απόδοσης των ασύρματων δικτύων και στην επαλήθευση αναλυτικών μοντέλων . Η χρήση προσομοιώσεων στα δίκτυα παρέχει αποτελέσματα μετρήσεων με χαμηλό κόστος, και η αλλαγή συνθηκών και παραμέτρων γίνεται γρήγορα και πιο εύκολα. Λόγο του ότι οι προσομοιώσεις απαιτούν μεγάλους υπολογιστικούς χρόνους από τους Η/Υ για αυτό και τα αποτελέσματα στις προσομοιώσεις δεν δίνονται σε πραγματικό χρόνο . Κάποιοι ερευνητές χρησιμοποίησαν προσομοιωτές που ανέπτυξαν οι ίδιοι , άλλοι χρησιμοποίησαν το εμπορικό πακέτο OPNET , ενώ άλλοι χρησιμοποίησαν τον προσομοιωτή ανοικτού κώδικα ns-2 (network simulation).Στην παρούσα πτυχιακή εργασία θα χρησιμοποιήσουμε το ns-2 .
- **Αναλυτικά (Μαθηματικά) Μοντέλα**. Καθώς η απόδοση των ασύρματων δικτύων επηρεάζεται από πολλούς παράγοντες , η μοντελοποίησή τους προσφέρει έναν γρήγορο και χαμηλού κόστους τρόπο ανάλυσης των συνθηκών που επικρατούν σε διαφορετικούς τρόπους οργάνωσης των

ασύρματων δικτύων .Η ακρίβεια των μοντέλων και η μοντελοποίηση των ασύρματων δικτύων σε σύνθετες συνθήκες είναι ένα από τα αντικείμενα σύγχρονης έρευνας στον χώρο της απόδοσης των ασύρματων δικτύων.

- **Πειράματα.** Τα πειράματα παρέχουν αξιόπιστα αποτελέσματα αλλά με μεγάλο κόστος σε σχέση με τις δυο προηγούμενες μεθόδους μετρήσεων. Η αγορά υλικού εξοπλισμού (ασύρματες κάρτες δικτύου , συσκευές AP), του κατάλληλου λογισμικού , η ενασχόληση ειδικών επιστημόνων , η εύρεση χώρων κ.α αυξάνουν τον βαθμό δυσκολίας οργάνωσης πειραμάτων ασύρματων δικτύων .

## • 2.12 Διαδικασία DCF του 802.11

### Περιγραφή

Τα τελευταία χρόνια , μεγάλο ενδιαφέρον έχει παρουσιάσει η σχεδίαση των ασύρματων δικτύων για επικοινωνίες τοπικών περιοχών .Η ομάδα μελέτης του 802.11 διαμορφώθηκε σύμφωνα με το IEEE Project 802 με σκοπό να παρουσιάσει ένα διεθνές πρότυπο για ασύρματα τοπικά δίκτυα (WLAN's). Η τελική έκδοση του προτύπου εμφανίστηκε πρόσφατα, και παρέχει λεπτομερείς περιγράφει του έλεγχου μέσου πρόσβασης (MAC) και τις προδιαγραφές φυσικού στρώματος (PHY) για WLAN's.Στο πρωτόκολλο 802.11 ο θεμελιώδης μηχανισμός για πρόσβαση στο μέσο ονομάζεται distributed coordination function (DCF).

Η διαδικασία DCF που ελέγχει την πρόσβαση στο μέσο χρησιμοποιεί τον μηχανισμό carrier sense multiple access with collision avoidance (CSMA/CA) για πρόσβαση στο μέσον που βασίζεται στον ανταγωνισμό μεταξύ των κόμβων (σταθμών). Ο μηχανισμός αυτός ορίζει κανόνες για την χρήση του μέσου και χρησιμοποιείται για να ελαττωθεί η πιθανότητα των συγκρούσεων σε ένα ασύρματο δίκτυο .Όλοι οι σταθμοί που θέλουν να μεταδώσουν ανιχνεύουν το μέσο για άλλες εκπομπές .Εάν υπάρχουν εκπομπές κάθε σταθμός αναβάλλει την εκπομπή του.

Για να ελαττωθεί η πιθανότητα να συμβούν συγκρούσεις ,όταν το μέσο ελευθερωθεί , κάθε σταθμός αναβάλλει την εκπομπή του για μια τυχαία χρονική

περίοδο . Ο σταθμός επιλέγει τυχαία έναν ακέραιο αριθμό από το 0 έως CW-1 (όπου το CW είναι το εύρος του παραθύρου ανταγωνισμού (Contention Window) ) . Ο αριθμός αυτός είναι ο μετρητής οπισθοχώρησης (backoff counter) και μετρά τις χρονοθυρίδες που ο σταθμός αναβάλλει την εκπομπή του . Όταν δύο ή περισσότεροι σταθμοί τυχαίνει να επιλέξουν τον ίδιο χρόνο αναβολής τότε δεν αποφεύγεται η σύγκρουση . Όσο μεγαλύτερος είναι ο αριθμός των σταθμών στο δίκτυο τόσο μεγαλύτερη είναι και η πιθανότητα οι σταθμοί να επιλέξουν το ίδιο backoff counter .

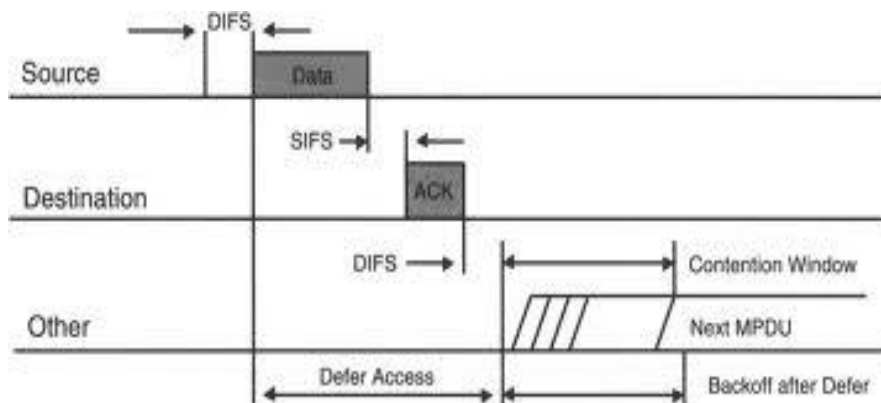
Η DCF βασίζεται σε δύο μεθόδους (τεχνικές) εκπομπής των πακέτων δεδομένων ,α) την Βασική πρόσβαση κατά την οποία γίνεται ανταλλαγή δύο πλαισίων (DATA-ACK) και β) την RTS/CTS (Request-To-Send /Clear-To-Send) μέθοδο πρόσβασης κατά την οποία γίνεται ανταλλαγή τεσσάρων πλαισίων (RTS-CTS-DATA-ACK).

Στην Βασική πρόσβαση ο αποστολέας εκπέμπει ένα πακέτο δεδομένων (DATA) και ο παραλήπτης μετά την επιτυχή λήψη του πακέτου δεδομένων βεβαιώνει την λήψη του πακέτου με την εκπομπή ενός πλαισίου επιβεβαίωσης (ACK) . Στην RTS/CTS πρόσβαση απαιτείτε η ανταλλαγή μικρών πλαισίων μεταξύ του αποστολέα (RTS)και του παραλήπτη (CTS) πριν την εκπομπή του πακέτου δεδομένων. (G. Bianchi , H. Wu, Y. Peng, K. Long, S. Cheng )

### **2.12.1 Βασική μέθοδος πρόσβασης**

Εάν ένας σταθμός έχει να στείλει ένα πακέτο δεδομένων και έχει ανιχνεύσει ότι το κανάλι είναι αδρανές για μια χρονική περίοδο ίση με το DIFS ,τότε ο σταθμός προχωρεί στην εκπομπή του πακέτου .Εάν το κανάλι είναι κατειλημμένο ο σταθμός περιμένει μέχρις ότου ανιχνεύσει ότι το κανάλι είναι ανενεργό για χρόνο ίσο με το DIFS , και κατόπιν ο σταθμός οπισθοχωρεί και αναβάλλει την εκπομπή του για μια τυχαία χρονική περίοδο . Αυτή η τυχαία χρονική περίοδος οπισθοχώρησης δημιουργείται από τον ίδιο τον σταθμό για να αποφύγει τις συγκρούσεις και μετράτε σε ακέραιες μονάδες χρονοθυρίδων (slot time). Ο μετρητής χρονικής οπισθοχώρησης (backoff counter)

ελαττώνεται κατά ένα όσο το κανάλι είναι ανενεργό για χρονικό μιας χρονοθυρίδας .Ο μετρητής σταματά ("παγώνει") να ελαττώνεται όταν το κανάλι είναι ενεργό και αρχίζει να ελαττώνεται όταν το κανάλι είναι ανενεργό για χρόνο μεγαλύτερο του DIFS . Ο σταθμός εκπέμπει ένα πακέτο όταν ο μετρητής οπισθοχώρησης φτάσει στο μηδέν . Εάν ο παραλήπτης λάβει με επιτυχία το πακέτο περιμένει για μια σύντομη χρονική περίοδο SIFS και μετά εκπέμπει ένα πακέτο ACK με το οποίο βεβαιώνει την επιτυχή λήψη του πακέτου. Εάν ο αποστολέας δεν λάβει ένα ACK πακέτο εντός μιας καθορισμένης χρονικής περιόδου ACK-timeout ,τότε θεωρεί ότι το πακέτο δεδομένων χάθηκε και ο σταθμός δρομολογεί επανάληψη της εκπομπής του πακέτου δεδομένων . Κάθε σταθμός διατηρεί έναν μετρητή επαναληπτικών προσπαθειών για κάθε πακέτο , ο οποίος αυξάνεται κατά ένα κάθε φορά που ένα πακέτο δεδομένων εκπέμπεται ανεπιτυχώς . Εάν ο μετρητής φτάσει στο μέγιστο όριο επαναληπτικών εκπομπών  $R$  τότε το πακέτο απορρίπτεται .Στο Σχήμα 2.10 παρουσιάζεται η Βασική μέθοδος μετάδοσης .



Σχήμα 2.10 Βασική μέθοδος πρόσβασης

Η ακέραια τιμή του μετρητή χρονικής οπισθοχώρησης επιλέγεται τυχαία με ομοιόμορφη κατανομή στο διάστημα  $[0, W_j - 1]$  , όπου  $j$  είναι η βαθμίδα οπισθοχώρησης (backoff stage δηλ. είναι ο αριθμός των επαναληπτικών εκπομπών ενός πακέτου)  $j$  ανήκει  $[0, R]$  και  $W_j$  είναι το μέγεθος του τρέχοντος παραθύρου ανταγωνισμού (CW) . Το παράθυρο ανταγωνισμού CW για την πρώτη εκπομπή ενός πακέτου λαμβάνει τιμή ίση με την ελάχιστη τιμή  $W_0 = W$ . Μετά από μια αποτυχημένη εκπομπή το παράθυρο ανταγωνισμού CW

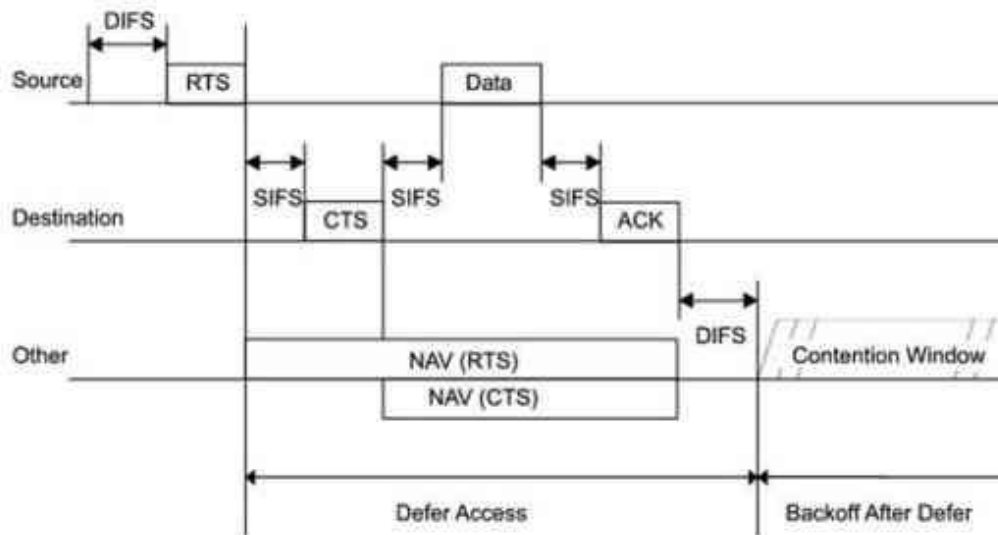
διπλασιάζεται . Ο αλγόριθμος διπλασιασμού του παραθύρου ανταγωνισμού μετά από μια αποτυχημένη εκπομπή πακέτου δεδομένων ονομάζεται BEB (Binary Exponential Backoff –δυναδικός εκθετικός αλγόριθμος οπισθοχώρησης) και ο διπλασιασμός γίνεται για να ελαττωθεί η πιθανότητα σύγκρουσης πακέτων . Η μέγιστη τιμή που μπορεί να πάρει το CW είναι  $2^m W$ , όπου  $m$  είναι ο αριθμός των διαφόρων μεγεθών του παραθύρου ανταγωνισμού .Από την στιγμή που το CW λάβει την μέγιστη τιμή , παραμένει σε αυτήν την τιμή μέχρι να τοποθετηθεί στην αρχική ελάχιστη τιμή. Το CW λαμβάνει την αρχική ελάχιστη τιμή μετά από μια επιτυχημένη εκπομπή πακέτου ή μετά από μια απόρριψη πακέτου το οποίο συμβαίνει όταν ο αριθμός των επαναληπτικών εκπομπών ενός πακέτου φτάσει στην τιμή (R. G. Bianchi H. Wu, Y. Peng, K. Long, S. Cheng)

## 2.12.2 RTS/CTS Μέθοδος πρόσβασης

Στα ασύρματα δίκτυα υπάρχουν δυο μηχανισμοί ανίχνευσης φέροντος σήματος . Ο πρώτος είναι ο φυσικός μηχανισμός ανίχνευσης του φέροντος (physical carrier sense) σήματος ο οποίος ελέγχει την ραδιοσυχνότητα του φέροντος σήματος για να εντοπίσει εάν υπάρχει κάποιος σταθμός που εκπέμπει .Ο δεύτερος είναι ο εικονικός μηχανισμός ανίχνευσης φέροντος (virtual carries sense) . Ο μηχανισμός αυτός χρησιμοποιεί έναν μετρητή χρόνου ο οποίος ονομάζεται NAV (Network Allocation Vector) και υπάρχει σε κάθε σταθμό του δικτύου. Εάν η τιμή του NAV είναι μηδέν τότε επιτρέπεται στον σταθμό να ανταγωνιστεί για την κατάληψη του μέσου . Εάν η τιμή του NAV είναι μεγαλύτερη από το μηδέν τότε ο σταθμός πρέπει να περιμένει μέχρις ότου ο μετρητής να μηδενιστεί και μετά να ανταγωνιστεί για το μέσο. Οι σταθμοί ενημερώνουν τους μετρητές NAV για την διάρκεια των εκπομπών σύμφωνα με τις πληροφορίες που υπάρχουν στα πεδία των πλαισίων RTS και CTS . Όταν ένας σταθμός πρόκειται να εκπέμψει ένα RTS υπολογίζει την διάρκεια κατάληψης του μέσου σύμφωνα με το μέγεθος του πακέτου , την ταχύτητα εκπομπής και τους χρόνους αναμονής . Η διάρκεια κατάληψης ενσωματώνεται στο πλαίσιο RTS .Με αυτόν τον τρόπο οι σταθμοί που ανιχνεύουν το RTS ενημερώνουν τους μετρητές NAV. Αντίστοιχα και ο παραλήπτης ενημερώνει τους σταθμούς μέσω της εκπομπής του πλαισίου



CTS για την διάρκεια κατάληψης του μέσου αφού προηγουμένως έχει ενσωματώσει την διάρκεια στο πλαίσιο . Στο σχήμα 2.11 παρουσιάζεται η RTS/CTS μέθοδος μετάδοσης .



Σχήμα 2.11 RTS/CTS μέθοδος πρόσβασης.

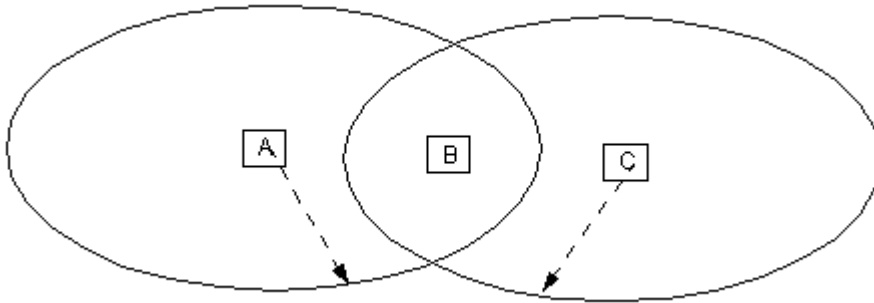
Η RTS/CTS πρόσβαση ακολουθεί τους ίδιους κανόνες οπισθοχώρησης που ισχύουν και στην Βασική μέθοδο . Ο αποστολέας εκπέμπει ένα πακέτο πλαίσιο RTS αντί του πακέτου δεδομένων . Ο παραλήπτης απαντά με ένα CTS πλαίσιο μετά από μια χρονική περίοδο ενός SIFS . Ο αποστολέας επιτρέπεται να εκπέμπει το πακέτο δεδομένων μόνο όταν λάβει ένα έγκυρο CTS πακέτο . Ο παραλήπτης αφού λάβει με επιτυχία το πακέτο δεδομένων εκπέμπει ένα πακέτο επιβεβαίωσης ACK .Εάν ο σταθμός αποστολέας δεν λάβει ένα CTS , ο μετρητής επαναληπτικών εκπομπών αυξάνεται κατά ένα .

(G. Bianchi H. Wu, Y. Peng, K. Long, S. Cheng)

### 2.12.3 Το πρόβλημα κρυμμένου τερματικού

Η RTS/CTS πρόσβαση αντιμετωπίζει με επιτυχία το πρόβλημα του κρυμμένου τερματικού (Hidden Station Problem) . Το πρόβλημα του κρυμμένου τερματικού συμβαίνει όταν το σήμα ενός τερματικού δεν λαμβάνεται από όλους τους σταθμούς στο δίκτυο . Στο σχήμα 2.12 παρουσιάζεται μια απλή περίπτωση κρυμμένου τερματικού , ο σταθμός B

λαμβάνει τα σήματα των σταθμών A και C , ενώ ο C και ο A είναι εκτός εμβέλειας ο ένας από τον άλλον . (Wu, Y. Peng, K. Long, S. Cheng).



Σχήμα 2.12 πρόβλημα κρυμμένου τερματικού

Το πρόβλημα μπορεί να δημιουργηθεί από τους A και C καθώς όταν ο ένας εκπέμπει ο άλλος θεωρεί ότι το μέσο είναι ελεύθερο με αποτέλεσμα να υπάρχει μεγάλη πιθανότητα να εκπέμπουν ταυτόχρονα ο A και ο C και ο B να μην μπορεί να διακρίνει κανένα σήμα . Με την RTS/CTS πρόσβαση ο B ενημερώνει με το CTS του το A ή τον C για πόσο χρόνο το μέσο θα είναι ενεργό . Ο A ή ο C αναβάλλει την εκπομπή του αφού προηγουμένως έχει ενημερωθεί ο μετρητής NAV.

Στη μέθοδο RTS/CTS η σύγκρουση μπορεί να συμβεί μονό στα πλαίσια RTS και ανιχνεύεται από την έλλειψη του πλαίσιο CTS από τον παραλήπτη . Για τον λόγο αυτό ο μηχανισμός RTS/CTS επιτρέπει την αύξηση της απόδοσης του συστήματος με την μείωση της διάρκειας των συγκρούσεων . Η αύξηση της απόδοσης είναι μεγάλη όταν υπάρχουν πολλοί σταθμοί στο δίκτυο ή ανταλλάσσονται μεγάλα πακέτα δεδομένων.

Στη διαδικασία DCF , όταν ένα πακέτο σε μέγεθος είναι μεγαλύτερο από μια δοθείσα καθορισμένη τιμή ορίου RTS-όριο(RTS-threshold),τότε χρησιμοποιείται η RTS/CTS μέθοδος , αλλιώς το πακέτο εκπέμπεται ακολουθώντας τους κανόνες της Βασικής πρόσβασης .

([http://www.joshuarobinson.net/docs/ns-802\\_11b.html](http://www.joshuarobinson.net/docs/ns-802_11b.html))

# Κεφάλαιο 3

## Network Simulator-2

### 3.1 Γενικά

Network Simulator -2 (NS-2) είναι ένας αντικειμενοστραφής προσομοιωτής δικτύων, γραμμένος σε C++ και OTcl (αντικειμενοστραφής έκδοση της γλώσσας TCL) . Στον NS-2 είναι ενσωματωμένα τα γνωστότερα δικτυακά πρωτόκολλα , όπως π.χ τα πρωτόκολλα TCP και UDP , πρωτόκολλα εφαρμογής (FTP,Telnet,Web,CBR) ,προσομοιώσεις διαχείρισης ουρών δρομολογητών (DropTail,RED), δρομολόγηση κτλ.

Οι προγραμματιστές του NS-2 προκειμένου να ισορροπήσουν ανάμεσα στην υψηλού επιπέδου υλοποίηση και στην ευκολία προγραμματισμού χώρισαν τον προσομοιωτή σε δυο τμήματα . Το ένα τμήμα (κυρίως για την διασύνδεση με τον χρήστη) είναι γραμμένο στη γλώσσα OTcl ώστε να μην χρειάζεται ο απλός χρήστης να γνωρίζει την δυσκολότερη αλλά πανίσχυρη C++ προκειμένου να τον χρησιμοποιήσει . Ο υπόλοιπος κώδικας του NS-2 είναι γραμμένος στην γλώσσα C++ ώστε το αποτέλεσμα να είναι υψηλού επιπέδου και να είναι αποδοτικός σε σχέση με την ταχύτητα .

Δηλαδή ένα δικτυακό πρόβλημα μπορεί να εκφραστεί μονό με OTcl ,μόνο με την C++ ή και με συνδυασμό των δυο. Συνήθως η παραμετροποίηση του προβλήματος γίνεται με την OTcl για μεγαλύτερη ταχύτητα και ευκολία , ενώ το υπόλοιπο πρόβλημα χρησιμοποιεί κώδικα C++ . τα αντικείμενα της C++ έχουν αναπαρασταθεί και σαν αντικείμενα της OTcl , έτσι ώστε να μπορεί να εκφράσει κάποιος ένα πρόβλημα και με τους δυο τρόπους .

<http://www.isi.edu/nsnam/ns/tutorial/index.html>)

### 3.2 Εγκατάσταση του NS2

Ο NS2 μπορεί να εγκατασταθεί σε συστήματα UNIX (FreeBSD, Linux, SunOS, Solaris) αλλά σε συστήματα Windows χρησιμοποιώντας το Cygwin, το οποίο είναι ένα περιβάλλον Linux, αλλά και το VMWare, το οποίο είναι ένα

πρόγραμμα εξομοίωσης ενός προσωπικού υπολογιστή μέσα από ένα ήδη υπάρχον λειτουργικό σύστημα (Windows, Linux).

(<http://allaboutns2.wordpress.com>)

### 3.2.1 Εγκατάσταση του NS2 στα Windows μέσω του Cygwin.

Αρχικά πρέπει να κατεβάσετε το **Cygwin** .

Κατά την εγκατάσταση του Cygwin:

1. σας ζητάει να επιλέξετε τον τύπο εγκατάστασης. Υπάρχουν τρεις επιλογές. Από αυτές εσείς διαλέγεται την πρώτη, εγκατάσταση μέσω του Internet.
2. επιλέγεται τον φάκελο εγκατάστασης και τον φάκελο στον οποίο θα αποθηκεύσει τα αρχεία που θα κατεβάσει. Ο φάκελος πρέπει να είναι ο ίδιος, αφήνεται τον προεπιλεγμένο (C:\cygwin).
3. επιλέγεται τον τρόπο που συνδέεστε στο Internet.
4. επιλέγεται από τη λίστα με τις διευθύνσεις μία.
5. επιλέγεται τα πακέτα που απαιτούνται για την εγκατάσταση του NS2. Αυτά είναι τα XFree86-base, XFree86-bin, XFree86-prog, XFree86-lib, XFree86-etc, make, patch, perl, gcc-g++, gawk, gnuplot, tar και gzip. Αν δεν υπάρχουν τα πακέτα XFree86 τότε εγκαταστήστε τα xorg-x11-bin, xorg-x11-bin-dlls, xorg-x11-devel, xorg-x11-libs-data και xorg-x11-etc.

Στη συνέχεια κατεβάζετε το **NS2 allinone**.

Εγκατάσταση του NS2:

1. Αντιγράφετε το αρχείο ns-allinone-x.xx.x.tar.gz στον φάκελο C:\cygwin\home\- 2. Εκτελείτε τον Cygwin και μπαίνετε στον φάκελο όπου αντιγράψατε το αρχείο (χρησιμοποιώντας την εντολή *cd folder\_name*).
- 3. Αποσυμπιέζετε το αρχείο με την εντολή *tar xvfz ns-allinone-x.xx.x.tar.gz*. Μετά την αποσυμπίεση δημιουργείται ο φάκελος ns-allinone-x.xx.x.

4. Μπαίνετε μέσα στον φάκελο αυτό με την εντολή `cd ns-allinone-x.xx.x` και μετά εκτελείτε την εντολή `./install`.
5. Αφού τελειώσει η εγκατάσταση του NS2 κάνετε επανεκκίνηση του Cygwin για να δημιουργηθεί το αρχείο `“.bashrc”`.
6. Αφού ξαναεκτελέσετε το Cygwin πηγαίνετε στον φάκελο `C:\cygwin\home\ και δίνετε την εντολή ls -a, αν δεν σας εμφανίσει το αρχείο “.bashrc” τότε δημιουργείστε το εσείς.`
7. Ανοίγετε το αρχείο `“.bashrc”` με κάποιον editor (πχ το Notepad) και στο τέλος του προσθέστε τις παρακάτω γραμμές:

```
export NS_HOME="pwd"/ns-allinone-2.28
export
PATH=$NS_HOME/tcl8.4.5/unix:$NS_HOME/tk8.4.5/unix:$NS_HOME/bin:$PATH
export
LD_LIBRARY_PATH=$NS_HOME/tcl8.4.5/unix:$NS_HOME/tk8.4.5/unix:$NS_HOME/otcl-1.9:$NS_HOME/lib:$LD_LIBRARY_PATH
export TCL_LIBRARY=$NS_HOME/tcl8.4.5/library
```

Σε κάθε export να βάζετε τις σωστές εκδόσεις. Στο πρώτο αλλάξετε την έκδοση του ns-allinone, στο δεύτερο την έκδοση του tcl και του tk, στο τρίτο την έκδοση του tcl, του tk και του otcl και στο τελευταίο την έκδοση του tcl.

Μέσω του Cygwin πηγαίνετε μέσα στον φάκελο `C:\cygwin\home\ και εκτελείτε την εντολή ./validate.`

Αν όλα τα test ολοκληρωθούν με επιτυχία τότε ο ns-2 εγκαταστάθηκε επιτυχώς.

### 3.2.2 Εγκατάσταση του NS2 στα Windows μέσω του VMWare.

1. Κατεβάζετε το [VMWare Workstation](#) και στη συνέχεια το [Ubuntu](#).
2. Κάνετε εγκατάσταση το VMWare Workstation.
3. Μέσω του VMWare κάνετε εγκατάσταση το Ubuntu.
4. Ακολουθήστε τα βήματα εγκατάστασης του NS-2 στα Ubuntu.

### 3.2.3 Εγκατάσταση του NS2 στα Linux (Ubuntu)

1. Ανοίγετε ένα terminal και γράφετε την παρακάτω εντολή για να κατεβάσετε τα απαιτούμενα πακέτα.

```
sudo apt-get install gcc g++ libxt-dev libx11-dev libxmu-dev build-essential  
autoconf automake
```

Αποσυμπιέζετε το αρχείο με την εντολή `tar xvfz ns-allinone-x.xx.x.tar.gz`. Μετά την αποσυμπίεση δημιουργείται ο φάκελος `ns-allinone-x.xx.x`.

2. Μπαίνετε μέσα στον φάκελο αυτό με την εντολή `cd ns-allinone-x.xx.x` και μετά εκτελείτε την εντολή `./install`.
3. Πηγαίνετε έναν φάκελο πίσω γράφοντας `cd..`.
4. Δίνεται την εντολή `gedit .bashrc` και σας ανοίγει ένα αρχείο, μετά πηγαίνετε στο τέλος του αρχείου και γράφετε το παρακάτω αν δεν υπάρχει.

```
# LD_LIBRARY_PATH  
OTCL_LIB=/home/manos/ns-allinone-2.34/otcl-1.13  
NS2_LIB=/home/manos/ns-allinone-2.34/lib  
X11_LIB=/usr/X11R6/lib  
USR_LOCAL_LIB=/usr/local/lib  
export  
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$OTCL_LIB:$NS2_LIB:$X11_LI  
B:$USR_LOCAL_LIB  
# TCL_LIBRARY  
TCL_LIB=/home/manos/ns-allinone-2.34/tcl8.4.18/library  
USR_LIB=/usr/lib  
export TCL_LIBRARY=$TCL_LIB:$USR_LIB  
# PATH  
XGRAPH=/home/manos/ns-allinone-2.34/bin:/home/sucha/ns-allinone-  
2.34/tcl8.4.18/unix:/home/manos/ns-allinone-2.34/tk8.4.18/unix  
NS=/home/manos/ns-allinone-2.34/ns-2.34/  
NAM=/home/manos/ns-allinone-2.34/nam-1.14/  
PATH=$PATH:$XGRAPH:$NS:$NAM
```

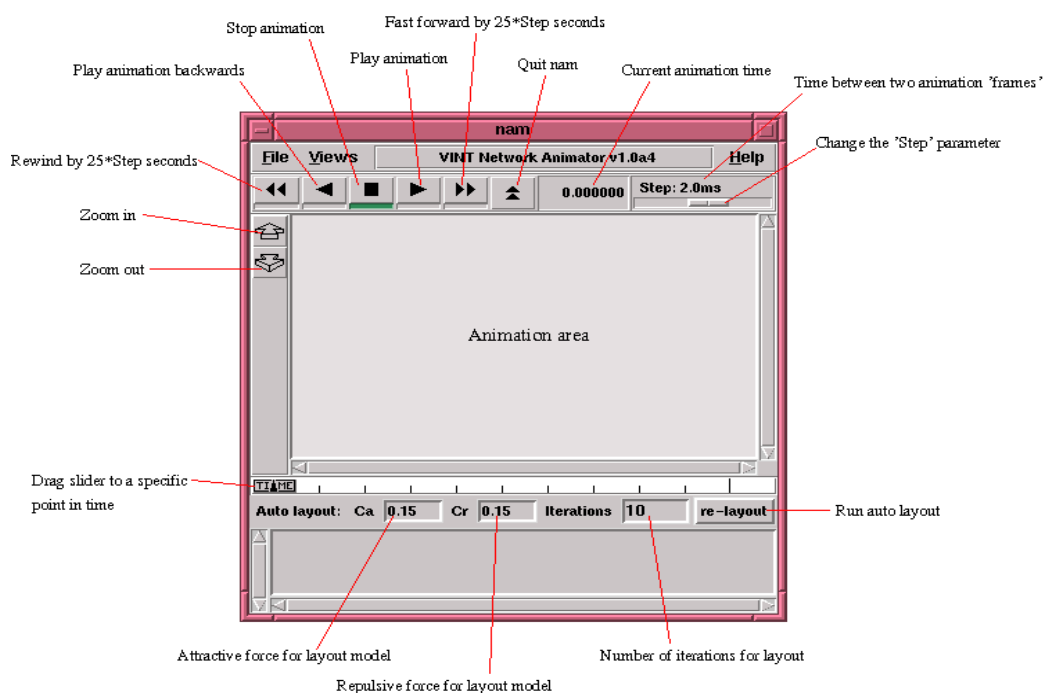
5. Δηλώστε τη διαδρομή όπου βρίσκεται ο φάκελος ns-allinone-x.xx.x αλλάζοντας το /home/manos. Επίσης αν χρειάζεται αλλάξτε την έκδοση των πακέτων (ns-allinone-2.34, tc18.4.18,...).
6. Μπείτε στον φάκελο ns-allinone-x.xx.x/nsx.xx και τρέξτε την εντολή `./validate`.
7. Αν όλα τα test ολοκληρωθούν με επιτυχία τότε ο ns-2 εγκαταστάθηκε ΕΠΙΤΥΧΩΣ.

### 3.3 Αρχεία εξόδου

Όταν τρέξουμε με το Network Simulator τον κώδικα δημιουργούνται δύο αρχεία εξόδου. Το ένα αρχείο αφορά την γραφική απεικόνιση της τοπολογίας και έχει κατάληξη αρχείο.nam. Το άλλο αρχείο καταγράφει την κίνηση των πακέτων που «κυκλοφορούν» σε αυτήν την τοπολογία και έχει κατάληξη αρχείο.tr.

#### 3.3.1 Αρχεία NAM

Τα αρχεία .nam κάνουν γραφική αναπαράσταση των αποτελεσμάτων με τη βοήθεια του παράθυρου διεπαφής του NAM. Παραθέτουμε ενδεικτικά την παρακάτω εικόνα παραθύρου (Σχήμα 3.1).

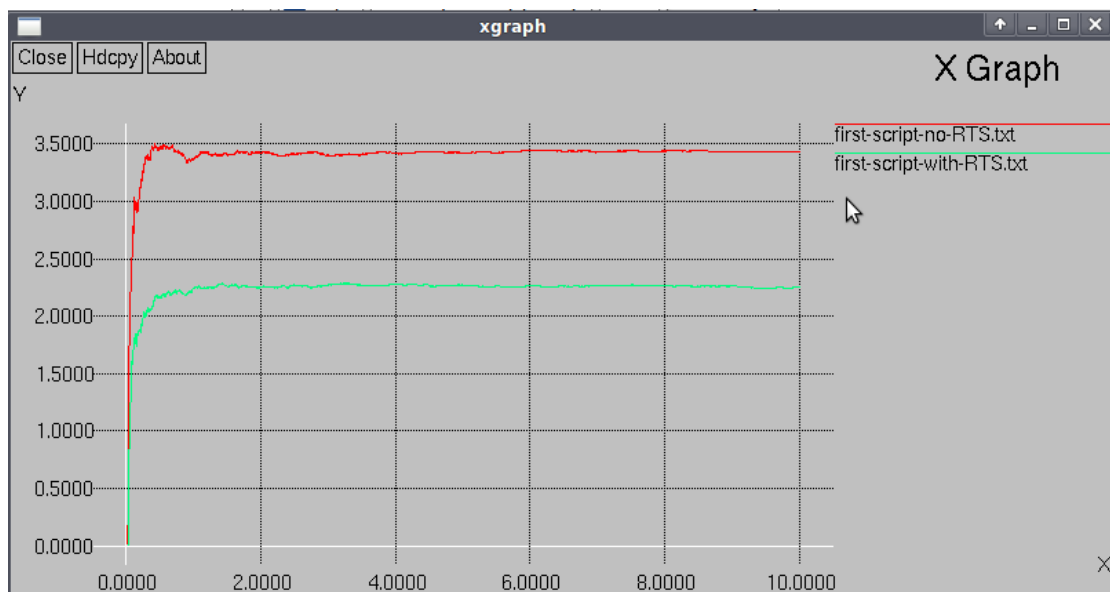


Σχήμα 3.1 . Αρχεία NAM

### 3.3.2 Αρχεία TR

Αν και το NAM είναι χρήσιμο για να έχουμε μια γενική εικόνα της εξομοίωσης, την πλήρη εικόνα της εξομοίωσης την έχουμε με τα αρχεία .tr. Είναι αρχεία κειμένου και σε ελεύθερη μετάφραση (trace file) είναι αρχεία κειμένου τα οποία καταγράφουν την κίνηση των πακέτων μεταξύ των κόμβων.

Όσο μεγαλύτερος είναι ο χρόνος της εξομοίωσης τόσο μεγαλύτερα είναι τα αρχεία αυτά , για αυτό το λόγο δεν μπορούμε να έχουμε κάποια εικόνα της κίνησης του δικτύου με μια απλή ματιά, καθώς το μόνο που βλέπουμε είναι σειρές από γράμματα και αριθμούς. Έτσι είναι μια αρκετά επίπονη διαδικασία να δούμε τις επιδόσεις του δικτύου, όμως με την βοήθεια της εντολής xgraph μπορούμε να απεικονίσουμε τις πληροφορίες που θέλουμε. Σχήμα 3.2



Σχήμα 3.2 Απεικόνιση αρχείο.tr με το xgraph

Κάθε γραμμή του αρχείου περιλαμβάνει ορισμένες πληροφορίες για τις ενέργειες που γίνονται στο δίκτυο, οι οποίες πληροφορίες είναι τοποθετημένες σε στήλες. Η πρώτη στήλη δείχνει αν η ενέργεια γίνεται από πηγή ή τον προορισμό του πακέτου. Η δεύτερη στήλη τη χρονική στιγμή της ενέργειας. Η τρίτη στήλη δείχνει τον κόμβο όπου λαμβάνει χώρα η ενέργεια αυτή. Έπειτα αναφέρεται το επίπεδο (layer) που γίνεται η ενέργεια. Ακολουθεί ο αριθμός σειράς του πακέτου και ο τύπος του. Τέλος αναφέρεται το μέγεθος του



πακέτου. Υπάρχουν και ορισμένες πληροφορίες έπειτα από όλα αυτά, οι οποίες είναι προαιρετικές και είναι συνήθως πληροφορίες σημαιών.

Παρακάτω γίνεται αναλυτική παρουσίαση του trace file.

(<http://www.cs.binghamton.edu/~kliu/research/ns2code/#trace>)

**ACTION:** [s|r|D]:

s -- sent  
r -- received  
D - dropped

**WHEN:** the time when the action happened

**WHERE:** the node where the action happened

**LAYER:**

AGT -- application,  
RTR -- routing,  
LL -- link layer (ARP is done here)  
IFQ -- outgoing packet queue (between link and mac layer)  
MAC -- mac,  
PHY -- physical

**flags:** The different reasons for dropping a packet are given below:

"END" DROP\_END\_OF\_SIMULATION  
"COL" DROP\_MAC\_COLLISION  
"DUP" DROP\_MAC\_DUPLICATE  
"ERR" DROP\_MAC\_PACKET\_ERROR  
"RET" DROP\_MAC\_RETRY\_COUNT\_EXCEEDED  
"STA" DROP\_MAC\_INVALID\_STATE  
"BSY" DROP\_MAC\_BUSY  
"NRTE" DROP\_RTR\_NO\_ROUTE i.e no route is available.  
"LOOP" DROP\_RTR\_ROUTE\_LOOP i.e there is a routing loop  
"TTL" DROP\_RTR\_TTL i.e TTL has reached zero.  
"TOUT" DROP\_RTR\_QTIMEOUT i.e packet has expired.  
"CBK" DROP\_RTR\_MAC\_CALLBACK  
"IFQ" DROP\_IFQ\_QFULL i.e no buffer space in IFQ.  
"ARP" DROP\_IFQ\_ARP\_FULL i.e dropped by ARP  
"OUT" DROP\_OUTSIDE\_SUBNET i.e dropped by base stations on receiving routing updates from nodes outside its domain.

**SEQNO:** the sequence number of the packet

**TYPE:** the packet type

cbr -- CBR data stream packet  
DSR -- DSR routing packet (control packet  
generated by routing)

```
RTS -- RTS packet generated by MAC 802.11
ARP -- link layer ARP packet
```

**SIZE:** the size of packet at current layer, when packet goes down, size increases, goes up size decreases

**[a b c d]:** a -- the packet duration in mac layer header  
b -- the mac address of destination  
c -- the mac address of source  
d -- the mac type of the packet body

**flags:**

**[.....]:**

```
[
source node ip : port_number
destination node ip (-1 means broadcast) : port_number
ip header ttl
ip of next hop (0 means node 0 or broadcast)
]
```

```
s 76.0000000000 _98_ AGT --- 1812 cbr 32 [0 0 0 0] -----
- [98:0 0:0 32 0]
```

Η εφαρμογή 0 (port number) στον κόμβο 98 στέλνει ένα CBR πακέτο του οποίου ο κωδικός πακέτου (ID) είναι 1812 και το μέγεθος του είναι 32 bytes, την χρονική στιγμή 76.0 δευτερόλεπτα , στην εφαρμογή 0 του κόμβου 0 με το TTL είναι 32 hops.

# Κεφάλαιο 4

## Πειραματικό Μέρος

Ο πιο εύκολος τρόπος να γνωριστούμε με την γλώσσα TCL , την οποία χρησιμοποιούμε για τα σενάρια των προσομοιώσεων , είναι να μελετήσουμε την δομή ενός απλού παραδείγματος . Στη συνέχεια του κεφαλαίου αυτού θα αναλύσουμε το βασικό σενάριο που χρησιμοποίησα για στην παρούσα πτυχιακή εργασία.

### 4.1 Απλό παράδειγμα

Το ακόλουθο παράδειγμα είναι ένα καλό ξεκίνημα για να εξοικειωθεί κάποιος με την γλώσσα Tcl και είναι αυτό που αναφέρεται σε κάθε εγχειρίδιο του NS-2. Δεν θα το παρεκκλίνω από την εκπαιδευτική του αξία και θα το παραθέσω ελαφρώς βελτιωμένο με κάποια δικά μου σχόλια .

Τα Tcl script μπορούν να γραφούν σε οποιοδήποτε text editor με κατάληξη “.tcl”.

Ας ονομάσουμε το παράδειγμα μας «example.tcl» .Πρώτη ενέργεια μας είναι να δημιουργήσουμε ένα αντικείμενο της τάξης Simulator .Αυτό γίνεται με την εντολή :

```
set ns [new Simulator]
```

Στην συνέχεια ανοίγουμε ένα αρχείο στο οποίο θα αποθηκευτούν τα στοιχεία (data trace) που είναι απαραίτητα για το NAM .Οι εντολές είναι οι ακόλουθες :

```
set nf [open out.nam w]  
$ns $ns namtrace-all $nf
```

Η πρώτη γραμμή ανοίγει το αρχείο , το οποίο ονομάζεται «out.nam» και του δίνει το file handle ‘nf’ . Στη δεύτερη γραμμή λέμε στο αντικείμενο Simulator

που δημιουργήσαμε πιο πάνω να γράψει όλα τα παραπάνω δεδομένα προσομοίωσης , που είναι σημαντικά και χρήσιμα για το NAM , σε αυτό το αρχείο .

Το επόμενο βήμα είναι να προσθέσουμε μια διαδικασία που θα κλείνει το αρχείο και θα ξεκινάει το NAM . Θα ονομάσουμε την διαδικασία αυτή «**stop**». Οι εντολές είναι οι ακόλουθες :

```
proc stop {} {  
    global ns nf  
    $ns flush-trace  
    close $nf  
    exec nam out.nam &  
    exit 0  
}
```

Στην παρούσα φάση αν δεν έχετε προγραμματιστικό ταλέντο ή ιδιαίτερη προϋπηρεσία δεν είναι εύκολο να καταλάβετε τι θέλουμε να πετύχουμε με αυτό το απόσπασμα κώδικα .Θα γίνει ξεκάθαρο στην πορεία όταν θα δούμε τον κώδικα ολοκληρωμένο .

Η επόμενη γραμμή λέει στο αντικείμενο Simulator να εκτελέσει την διαδικασία «stop» 5,0 δευτερόλεπτα μετά από την έναρξη της προσομοίωσης . Παρατηρήστε ότι το Ns μας παρέχει έναν πολύ απλό τρόπο για να προγραμματιστούν τα γεγονότα με την “at” εντολή .

```
$ns at 5.0 “stop”
```

Η τελευταία γραμμή είναι και αυτή που αρχίζει την προσομοίωση :

```
$ns run
```

Ολοκληρωμένο το αρχείο μέχρι στιγμής :

```
#Create a Simulator object  
set ns [new Simulator]  
#Open the nam trace file  
set nf [open out.nam w]  
$ns $ns namtrace-all $nf
```

```

#Define a 'stop' procedure
proc stop {} {
    global ns nf
    $ns flush-trace
#Close the trace file
    close $nf
#Execute nam on the trace file
    exec nam out.nam &
    exit 0
}
#Call the stop procedure after 5 seconds simulation time
$ns at 5.0 "stop"
#Run the simulation
$ns run

```

Στο επόμενο βήμα του παραδείγματος μας θα προσθέσουμε στο σενάριο της προσομοίωσης , που αυτήν την στιγμή είναι κενό , δύο κόμβους και θα δημιουργήσουμε μια σύνδεση μεταξύ τους .

Οι εντολές που δημιουργούν τους δύο κόμβους θα πρέπει να εισαχθούν πριν από την γραμμή που ορίζει την διάρκεια της προσομοίωσης και είναι οι ακόλουθες :

```

set n0 [$ns node]
set n1 [$ns node]

```

Βλέπουμε ότι ένα αντικείμενο τύπου **node** (κόμβος) δημιουργείται με την εντολή «**\$ns node**».Στον παραπάνω κώδικα δημιουργεί δύο κόμβους και θα τους δίνει handles «**n0**» και «**n1**».

Η επόμενη γραμμή είναι αυτή που συνδέει τους δύο κόμβους .

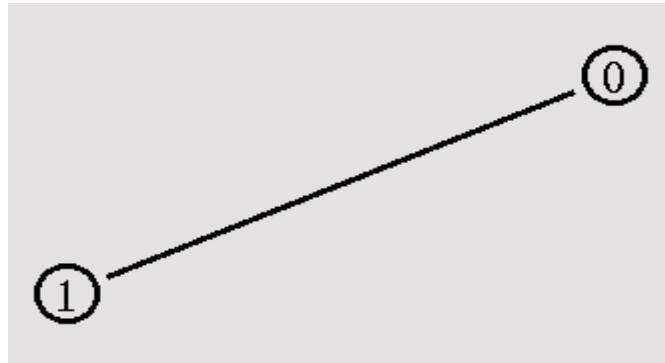
```

$ns duplex-link $no $n1 1Mb 10ms DropTail

```

Η γραμμή αυτή λέει στο αντικείμενο **Simulator** να συνδέσει τους κόμβους «**n0**» και «**n1**» με μια duplex γραμμή ,με το εύρος ζώνης **1Mbit** , delay **10 ms** και μια ούρα **DropTail** .

Αν αποθηκεύσουμε τώρα το αρχείο και το τρέχουμε με την εντολή «**ns example.tcl**» θα ξεκινήσει αυτόματα το NAM και θα πρέπει να δούμε ένα σχήμα που θα μοιάζει με την παρακάτω εικόνα και θα είναι ουσιαστικά αυτό που περιγράψαμε στο σενάριο μας .



Ολοκληρωμένο το αρχείο μέχρι στιγμής :

```
#Create a Simulator object
set ns [new Simulator]
#Open the nam trace file
set nf [open out.nam w]
$ns $ns namtrace-all $nf
#Difine a 'stop' proxedure
proc stop {} {
    global ns nf
    $ns flush-trace
#Close the trace file
    close $nf
#Execute nam on the trace file
    exec nam out.nam &
    exit 0
}
#Create two nodes
set n0 [$ns node]
set n1 [$ns node]
#Create a duplex link between the nodes
```

```

$ns duplex-link $no $n1 1Mb 10ms DropTail
#Call the stop procedure after 5 seconds simulation time
$ns at 5.0 "stop"
#Run the simulation
$ns run

```

Μέχρι στιγμής έχουμε φτιάξει μόνο την τοπολογία . Επόμενο βήμα είναι να βάλουμε κίνηση δεδομένων μεταξύ κόμβων . Θα ορίσουμε λοιπόν κίνηση στέλνοντας δεδομένα από τον κόμβο «n0» στον κόμβο «n1» .Αυτό θα το πετύχουμε δημιουργώντας ένα αντικείμενο τύπου agent στο n0 που θα στέλνει τα δεδομένα και αντικείμενο τύπου agent n1 που θα λαμβάνει . Οι εντολές για να πετύχουμε αυτό είναι οι ακόλουθες :

```

set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

```

Πιο αναλυτικά . Οι 2 πρώτες γραμμές δημιουργούν έναν **UDP agent (udp0)** τον επιθέτουν στον κόμβο **n0** . Η Τρίτη γραμμή δημιουργεί μια γεννήτρια CBR κίνησης , δηλαδή μια γεννήτρια παραγωγής σταθερού ρυθμού bit (constant bit rate generator) . Η τέταρτη και η πέμπτη γραμμή ορίζουν το packetSize στα 500 byte και τον ενδιάμεσο χρόνο μεταξύ των πακέτων που αποστέλλονται τα 0.005 sec (δηλαδή 200 πακέτα ανά δευτερόλεπτο) . Η έκτη γραμμή εναποθέτει την γεννήτρια CBR στον UDP agent udp0  
 Οι επόμενες γραμμές θα δημιουργήσουν έναν **Null agent** , τον **null0** ο οποίος θα λαμβάνει την κίνηση (**traffic sink**) και τον επιθέτουμε στον κόμβο n1 .

```

set null0 [new Agent/Null]
$ns attach-agent $n1 $null0

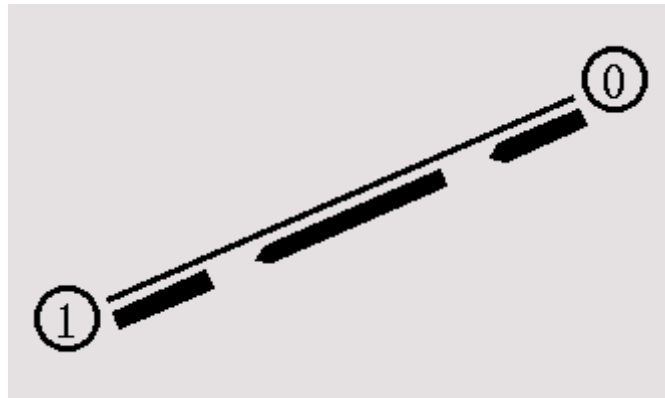
```

Έπειτα από αυτό μένει να συνδέσουμε τους agent μεταξύ τους , να ορίσουμε το χρονικό σημείο στο οποίο θα αρχίσει η αποστολή δεδομένων και το

χρονικό σημείο που θα σταματήσει . στο παράδειγμα μας ξεκινά από 0,5 και τελειώνει στο 4,5 . Οι εντολές για αυτό είναι οι ακόλουθες :

```
$ns connect $udp0 $null0  
$ns at 0.5 "$cbr0 start"  
$ns at 4.5 "$cbr0 stop"
```

Τώρα μπορούμε να αποθηκεύσουμε το αρχείο και να ξεκινήσουμε πάλι την προσομοίωση . Αυτή την φορά όταν κάνουμε κλικ στο «**play**» , του παραθύρου του NAM θα δούμε ότι μετά από 0,5 δευτερόλεπτα προσομοίωσης , ο κόμβος n0 αρχίζει να στέλνει πακέτα στον κόμβο n1. Η εικόνα θα έχει μορφή παρόμοια με την εικόνα που ακολουθεί .



Η τελική έκδοση του αρχείου θα είναι η εξής :

```
#Create a Simulator object  
set ns [new Simulator]  
#Open the nam trace file  
set nf [open out.nam w]  
$ns $ns namtrace-all $nf  
#Difine a 'stop' proxedure  
proc stop {} {  
    global ns nf  
    $ns flush-trace  
#Close the trace file  
    close $nf  
#Execute nam on the trace file
```



```

        exec nam out.nam &
        exit 0
    }
#Create two nodes
set n0 [$ns node]
set n1 [$ns node]
#Create a duplex link between the nodes
$ns duplex-link $no $n1 1Mb 10ms DropTail
#Create a UDP agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
#Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
#Create a Null agent (a traffic sink) and attach it to node n1
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0
#Connect the traffic source with the traffic sink
$ns connect $udp0 $null0
#Schedule events for the CBR traffic
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"
#Call the stop procedure after 5 seconds simulation time
$ns at 5.0 "stop"
#Run the simulation
$ns run

```

## 4.2 Παράδειγμα αρχείου προσομοίωσης

Για να γίνει κατανοητό το αρχείο που περιγράφει την προσομοίωση ενός ασύρματου δικτύου θα το παραθέσω αυτούσιο με κάποια σχόλια όπου κρίνω ότι χρειάζονται. Στην Tcl ότι ακολουθεί το σύμβολο “#” θεωρείτε σχόλιο. τα

υπόλοιπα σενάρια προσομοιώσεων θα τα παραθέσω σε παράρτημα στο τέλος της παρούσας πτυχιακής εργασίας.

# Παράμετροι κόμβων

```
set val(chan)      Channel/WirelessChannel      ;# Τύπος καναλιού
set val(prop)      Propagation/TwoRayGround     ;# Τύπος ράδιο-μετάδοσης
# Οι τιμές που μπορεί να πάρει είναι : FreeSpace , TwoRayGround ,
#Shadowing.Στην 1η περίπτωση υποθέτουμε ότι η διαδρομή ανάμεσα στον
#πομπό και στον δέκτη είναι άμεση , στην 2η περίπτωση συνυπολογίζεται και
#η ανάκλαση των σημάτων στο έδαφος τέλος στην 3η περίπτωση
#προσομοιώνεται και η απόσβεση του σήματος .
set val(netif)     Phy/WirelessPhy             ;# Ορίζει να είναι ασύρματο το
#interface του δικτύου
set val(mac)       Mac/802_11                  ;# Ορίζει τον τύπο του MAC
#υποστρώματος
set val(ifq)       Queue/DropTail/PriQueue     ;# Ορίζει τον τύπο interface της
#ουράς
set val(ll)        LL                          ;# link layer type
set val(ant)       Antenna/OmniAntenna        ;# Ορίζει το μοντέλο της κεραίας
set val(ifqlen)    50                          ;# Ορίζει το μέγεθος της ουράς σε ένα interface.έχει
#μεγάλη επιρροή στην καθυστέρηση των πακέτων και drop rate.
set val(rp)        AODV                        ;# Ορίζει το πρωτόκολλο δρομολόγησης
#π.χ DSR, DSDV, AODV, DumbAgent . Το DumbAgent χρησιμοποιούμε όταν
#όλοι οι κόμβοι ανήκουν στο ίδιο collision domain ,όπου δεν χρειάζεται
#πρωτόκολλο δρομολόγησης .
# Παράμετροι τοπολογίας
set val(x)         1000                        ;# Ορίζει την διάσταση X της τοπογραφίας
set val(y)         1000                        ;# Ορίζει την διάσταση Y της τοπογραφίας
#set val(cp)       ""                          ;# node movement model file
#set val(sc)       ""                          ;# traffic model file
set val(nn)        2                          ;#Ορίζουμε των αριθμός τον κόμβων
set val(seed)      0.0                        ;# move speed of node
set val(stop)      60                         ;# Συνολικός χρόνος προσομοίωσης
```

### # Παραμετροποίηση του Link Layer

```
LL set mindelay_      0us                ;# 50us (default value)
LL set delay_         0us                ;# 25us (default value)
LL set bandwidth_     0                  ;# not used
LL set debug_         false
LL set avoidReordering_ false
```

### # Παραμετροποίηση του υποστρώματος MAC

```
Mac/802_11 set CWMin_      31
Mac/802_11 set CWMax_      1023
Mac/802_11 set SlotTime_   0.000020    ;# 802.11b = 20us (default
#value) , 802.11a = 16us
Mac/802_11 set SIFS_       0.000010    ;# 802.11b = 10us (default
#value) , 802.11a = 9us
Mac/802_11 set PreambleLength_ 144      ;# 144 bit=18 byte
Mac/802_11 set PLCPHeaderLength_ 48     ;# 48 bits=6 byte
Mac/802_11 set PLCPDataRate_ 1.0e6     ;# 1Mbps
Mac/802_11 set RTSThreshold_ 0          ;# unit : byte
Mac/802_11 set rtxLimit_   6            ;# ~/ns/tcl/lan/ns-mac.tcl
Mac/802_11 set ShortRetryLimit_ 7      ;#is the retransmission #limit for RTS
Mac/802_11 set LongRetryLimit_ 1       ;#is the retransmission limit for DATA
#packet longer than RTS_THRESHH
Mac/802_11 set basicRate_   1Mb         ;# 802.11 basic rate
Mac/802_11 set dataRate_    11Mb        ;# 802.11 data rate
# Mac/802_11 set basicRate_ 1Mb ; set this to 0 if want to use bandwidth_ for
# Mac/802_11 set dataRate_ 1Mb ; both control and data pkts
```

### # Other default settings

```
Agent/Null set sport_      0
Agent/Null set dport_      0
Agent/CBR set sport_        0
Agent/CBR set dport_        0
Agent/TCPSink set sport_    0
Agent/TCPSink set dport_    0
Agent/TCP set sport_        0
Agent/TCP set dport_        0
```

```
Agent/TCP set packetSize_ 1023
```

```
Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1
```

```
# Ρύθμιση της κεραίας ώστε να είναι στο κέντρο κάθε κόμβου και 1.5 πάνω  
#από αυτό
```

```
Antenna/OmniAntenna set X_ 0
```

```
Antenna/OmniAntenna set Y_ 0
```

```
Antenna/OmniAntenna set Z_ 1.5
```

```
Antenna/OmniAntenna set Gt_ 1.0
```

```
Antenna/OmniAntenna set Gr_ 1.0
```

```
# Παραμετροποίηση του κοινού μέσου.
```

```
Phy/WirelessPhy set CPTresh_ 10.0 ;# capture threshold (db)
```

```
Phy/WirelessPhy set CSTresh_ 1.559e-11 ;# carrier sense threshold (W)
```

```
Phy/WirelessPhy set RXThresh_ 3.652e-10 ;# receive power threshold (W)
```

```
Phy/WirelessPhy set bandwidth_ 2e6 ;# 802.11b = 2 Mbps (default  
#value) , 802.11a = 54 Mbps
```

```
Phy/WirelessPhy set Pt_ 0.28183815 ;# transmitted signal power (W)
```

```
Phy/WirelessPhy set freq_ 914e+6 ;# frequency
```

```
Phy/WirelessPhy set L_ 1.0 ;# system loss factor
```

```
#Δημιουργία αντικειμένου ns_
```

```
set ns_ [new Simulator]
```

```
# Δημιουργία αρχείου για το NAM
```

```
set namfd [open 2-node.nam w]
```

```
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)
```

```
#Δημιουργία αρχείου ίχνους (trace file)
```

```
set tracefd [open 2-node.tr w]
```

```
$ns_ trace-all $tracefd
```

```
# Δημιουργία αντικειμένου τοπολογίας
```

```
set topo [new Topography]
```

```
# Μέγεθος τοπολογίας : 1000m x 1000m
```

```
$topo load_flatgrid $val(x) $val(y)
```

```

# Δημιουργία καναλιού
set chan [new $val(chan)]

# Δημιουργία God
set god_ [create-god $val(nn)]

#Random Mode
$defaultRNG seed 1 ;# ON/OFF the ranodm mode. (seed 0 is ON)

# set node parameter
puts "ns_ node-config"
$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channel $chan

# Δημιουργία κόμβων
for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0 ;# disable random motion
}

# Ορίζουμε τις συντεταγμένες κόμβων
$node_(0) set X_ 250.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0

```

```

$node_(1) set X_ 400.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0

# Ορίζουμε το μέγεθος των κόμβων στο nam
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 80
}
#Δημιουργία ροη δεδομένων FTP/TCP.
#Ροη δεδομένων από τον κόμβο 0 στο 1
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink

set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 0.0 "$ftp start"

#Διαδικασία λήξης προσομοίωσης
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}
$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 2-node.nam &
    exit 0
}
puts "Starting Simulation..."
$ns_ run

```

# Κεφάλαιο 5

## Μετρήσεις – Συμπεράσματα

Στο κεφάλαιο αυτό θα μελετήσουμε τις μετρήσεις κάποιων βασικών ιδιοτήτων κάθε δικτύου του Throughput και του Delay.

### 5.1 Throughput (Ρυθμαπόδοση)

Στην πληροφορική ονομάζουμε **διαμεταγωγή** ή **ρυθμαπόδοση** ή **ταχύτητα μετάδοσης πληροφορίας** τον χρονικό ρυθμό με τον οποίο ένας υπολογιστής, ένα δίκτυο υπολογιστών ή ένα υποσύστημα ενός υπολογιστή αποστέλλει ή λαμβάνει δεδομένα. Το **Throughput** αποτελεί ένα μέτρο της χωρητικότητας ενός τηλεπικοινωνιακού καναλιού ή μιας επικοινωνιακής σύνδεσης. Οι συνδέσεις αυτές συνήθως χαρακτηρίζονται από το bit rate τους, δηλαδή πόσα bit μεταδίδονται μέσω αυτών των συνδέσεων σε ένα δευτερόλεπτο (bit/sec).

### 5.2 Network Delay (Καθυστέρηση δικτύου)

Η πρώτη σημαντική ιδιότητα δικτύου που μπορεί να μετρηθεί ποσοτικά είναι το Delay. Η καθυστέρηση ενός δικτύου καθορίζει το πόσο χρονικό διάστημα χρειάζεται ένα bit για να ταξιδέψει από έναν υπολογιστή σε έναν άλλο μέσω του δικτύου. Η καθυστέρηση μετριέται σε δευτερόλεπτα ή κλάσματα δευτερολέπτου. Μπορεί να διαφέρει ελαφρώς, ανάλογα με τη θέση του ζεύγους των υπολογιστών που επικοινωνούν.

Αν και ο χρήστης ενδιαφέρεται μόνο για την συνολική καθυστέρηση του δικτύου οι μηχανική κάνουν πιο ακριβείς μετρήσεις. Έτσι, συνήθως έχουμε την μέγιστη και μέση καθυστέρηση, και επίσης μπορούν να διαχωρίζουν την καθυστέρηση σε άλλες υποκατηγορίες:

- **Processing delay** : η ώρα που χρειάζεται ένας δρομολογητής για να αναλάβει την διαδικασία του πακέτου κεφαλίδας.

- Queuing delay: ο χρόνος που το πακέτο βρίσκεται σε ουρά δρομολόγησης.
- Transmission delay: ο χρόνος που χρειάζεται για να προωθηθεί το πακέτο.
- Propagation delay: ο χρόνος που χρειάζεται το σήμα να μεταδοθεί στο μέσο επικοινωνίας.

Υπάρχει ένα ελάχιστο επίπεδο καθυστέρησης που θα οφείλεται στο χρόνο που χρειάζεται για να μεταδοθεί ένα πακέτο σειριακά μέσω ενός link . Σε αυτό προστίθεται μια πιο μεταβλητού επιπέδου καθυστέρηση που οφείλεται στην συμφόρηση του δικτύου (network congestion) .

## 5.3 Εξαγωγή Στατιστικών

Στατιστικά στοιχεία για τα σενάρια μπορούμε να βγάλουμε με τη χρήση κάποιου script .Το script που χρησιμοποιούμε είναι γραμμένο σε γλώσσα awk και υπάρχει σε αρκετά forum σχετικά με NS-2 . Επίσης θα χρησιμοποιήσουμε την εντολή xgraph και Microsoft Excel για να εμφανίσουμε τα διαγράμματα .

Παρακάτω θα παραθέσω 3 κώδικες γραμμένες σε awk , το πρώτο για την ρυθμοαπόδοση το δεύτερο για την καθυστέρηση και το τρίτο για τα υπόλοιπα αποτελέσματα (throughput.awk , delay.awk , results.awk) .

### 5.3.1 Throughput.awk

```
BEGIN {
    recvdSize = 0
    startTime = 400
    stopTime = 0
}
{
    event = $1
    time = $2
    node_id = $3
    pkt_size = $8
    level = $4
```



```

# Store start time
if (level == "MAC" && event == "s" && pkt_size >= 1023) {
    if (time < startTime) {
        startTime = time
    }
}
# Update total received packets' size and store packets arrival time
if (level == "MAC" && event == "r" && pkt_size >= 1023) {
    if (time > stopTime) {
        stopTime = time
    }
    # Rip off the header
    hdr_size = pkt_size % 1023
    pkt_size -= hdr_size
    # Store received packet's size
    recvdSize += pkt_size
}
}
END {
    printf("Average Throughput[kbps] = %.4f\t\t
StartTime=%.4f\tStopTime=%.4f\n", (recvdSize/(stopTime-
startTime))*(8/1000000), startTime, stopTime)
}

```

### 5.3.2 Delay.awk

```

BEGIN {
    seqno = -1;
    count = 0;
}
{
    if($4 == "MAC" && $1 == "s" && seqno < $6) {
        seqno = $6;
    }
    if($4 == "MAC" && $1 == "s") {
        start_time[$6] = $2;
    } else if(($7 == "tcp") && ($1 == "r")) {

```

```

        end_time[$6] = $2;
    } else if($1 == "D" && $7 == "tcp") {
        end_time[$6] = -1;
    }
}
END {
    for(i=0; i<=seqno; i++) {
        if(end_time[i] > 0) {
            delay[i] = end_time[i] - start_time[i];
            count++;
        } else
        {
            delay[i] = -1;
        }
    }
    for(i=0; i<=seqno; i++) {
        if(delay[i] > 0) {
            n_to_n_delay = n_to_n_delay + delay[i];
        }
    }
    n_to_n_delay = n_to_n_delay/count;
    print "\n";
    print "Average End-to-End Delay  = " n_to_n_delay * 1000 " ms";
    print "\n";
}

```

### 5.3.3 Results.awk

```

BEGIN {
    packet_collision=0;
    packet_drop=0;
    packet_sent=0;
    receivedPackets = 0;
}
{
    if($4 == "MAC" && $1 == "s" && $8>=1023) {
        packet_sent++;
    }
}

```

```

}else if(($4 == "MAC") && ($1 == "r")&&($8>=1023)) {
    receivedPackets++;
}
if($4 == "MAC" && $1 == "D" && $5 == "COL") {
    packet_collision++;
} else if($4 == "MAC" && $1 == "D" && $5 == "RET"){
    packet_drop++;
}else if($4 == "MAC" && $1 == "D" && $5 == "CBK"){
    packet_drop++;
}
}
}
END {
print "SentPackets          = " packet_sent"\n";
print "ReceivedPackets      = " receivedPackets"\n";
print "Packet Delivery Ratio  = "      receivedPackets/packet_sent*100 "%\n";
print "Packet Collision = " packet_collision "\n";
print "Packet Drop = " packet_drop "\n";
print "\n";
}

```

## 5.4 Αποτελέσματα Προσομοίωσης

Τα αποτελέσματα που θα ακολουθήσουν βγήκαν από μετρήσεις που έγιναν σε σενάρια ασύρματων τοπικών δικτύων με μέγεθος από δύο έως δέκα κόμβους , ανά δύο κόμβους , με συγκεκριμένες συνδέσεις μεταξύ κόμβων . Στον υπολογισμό του delay δεν υπολογίζονται τα drop πακέτα .Επίσης στον υπολογισμό του throughput αφαιρείτε ο header και μένει καθαρή πληροφορία. Ο κάθε σταθμός είναι σε κατάσταση κορεσμού και δεν υπάρχουν παράσιτα και κάθε σταθμός .

<b>ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΠΡΟΣΟΜΟΙΩΣΗΣ</b>	
<b>Πρωτόκολλο</b>	<b>802.11</b>
<b>Τοπολογία</b>	<b>ad-hoc</b>
<b>Movement</b>	<b>Ακίνητοι</b>
<b>Έκταση περιοχής</b>	<b>1000x1000</b>
<b>Τύπος πακέτου</b>	<b>TCP</b>
<b>Μέγεθος πακέτου</b>	<b>1023 byte</b>
<b>CWmax</b>	<b>1023</b>
<b>CWmin</b>	<b>31</b>
<b>Basic Rate</b>	<b>1 Mbps</b>
<b>Data Rate</b>	<b>11 Mbps</b>
<b>Χρόνος προσομοίωσης</b>	<b>60 sec</b>
<b>R (Επανεκπομπές)</b>	<b>6</b>

Πίνακας 5.1 Χαρακτηριστικά Προσομοίωσης

<b>Node</b>	<b>Throughput</b>		<b>Delay</b>	
	<b>RTS/CTS</b>	<b>Βασική Μέθοδος</b>	<b>RTS/CTS</b>	<b>Βασική Μέθοδος</b>
<b>2</b>	3,2061	3,3986	0,989988	0,989933
<b>4</b>	3,2200	3,4187	0,990021	0,98999
<b>6</b>	3,1962	3,3560	0,990135	0,990086
<b>8</b>	3,1786	3,2974	0,990158	0,990121
<b>10</b>	3,1674	3,2799	0,990288	0,990191

Πίνακας 5.2 Τιμές για την μέση ρυθμοαπόδοση και μέση καθυστέρηση

Node	Drop		Collision	
	RTS/CTS	Βασική Μέθοδος	RTS/CTS	Βασική Μέθοδος
<b>2</b>	0	0	0	0
<b>4</b>	0	0	380	458
<b>6</b>	4	0	409	612
<b>8</b>	4	1	678	725
<b>10</b>	0	2	910	1098

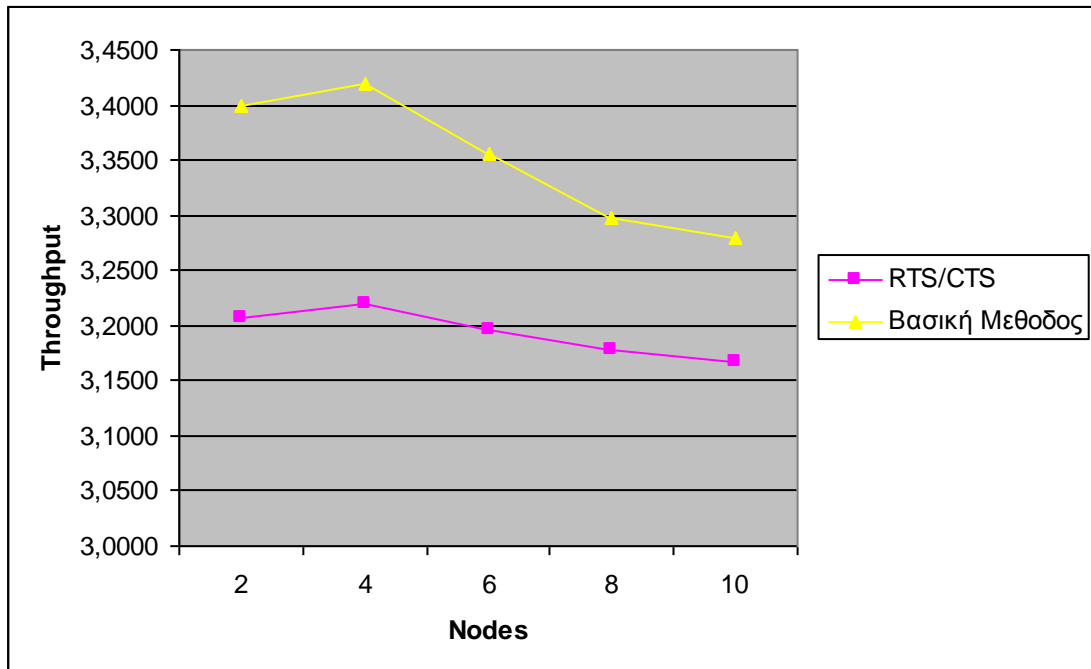
Πίνακας 5.3 Τιμές για το σύνολο συγκρούσεων και απορριφθέντων πακέτων

Node	Sent Packet		Receive Packet	
	RTS/CTS	Βασική Μέθοδος	RTS/CTS	Βασική Μέθοδος
<b>2</b>	25097	26487	25097	26487
<b>4</b>	25700	27107	25220	26700
<b>6</b>	26140	27618	25050	26954
<b>8</b>	25890	28351	25340	27162
<b>10</b>	26330	28654	25420	27033

Πίνακας 5.4 Τιμές για το σύνολο των σταλμένων και ληφθέντων πακέτων.

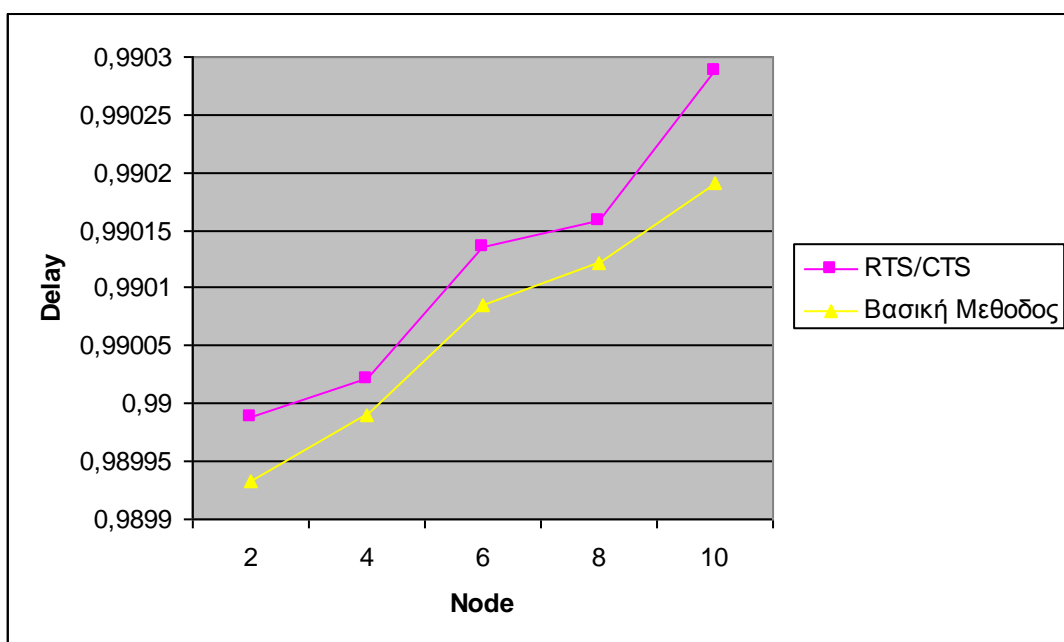
Μπορούμε να δούμε και με γραφική απεικόνιση , έτσι θα μπορέσουμε να βγάλουμε συμπεράσματα ευκολότερα .

### 5.4.1 Διάγραμμα 1 . Μέση Ρυθμοαπόδοση (Average Throughput).



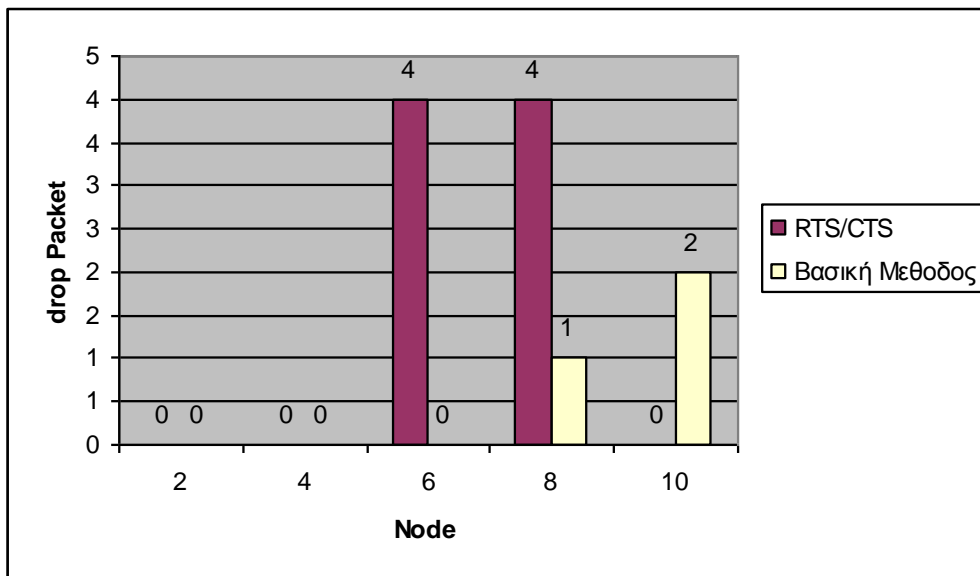
Στο διάγραμμα αυτό βλέπουμε την ρυθμοαπόδοση για την RTS/CTS μέθοδο και για την Βασική μέθοδο . Οι τιμές και στις δυο περιπτώσεις κυμαίνονται από 3,15 έως 3,25 για την RTS/CTS μέθοδο και από 3,25 έως 3,45 για την Βασική μέθοδο .

### 5.4.2 Διάγραμμα 2 . Μέση Καθυστέρηση (Average Delay).



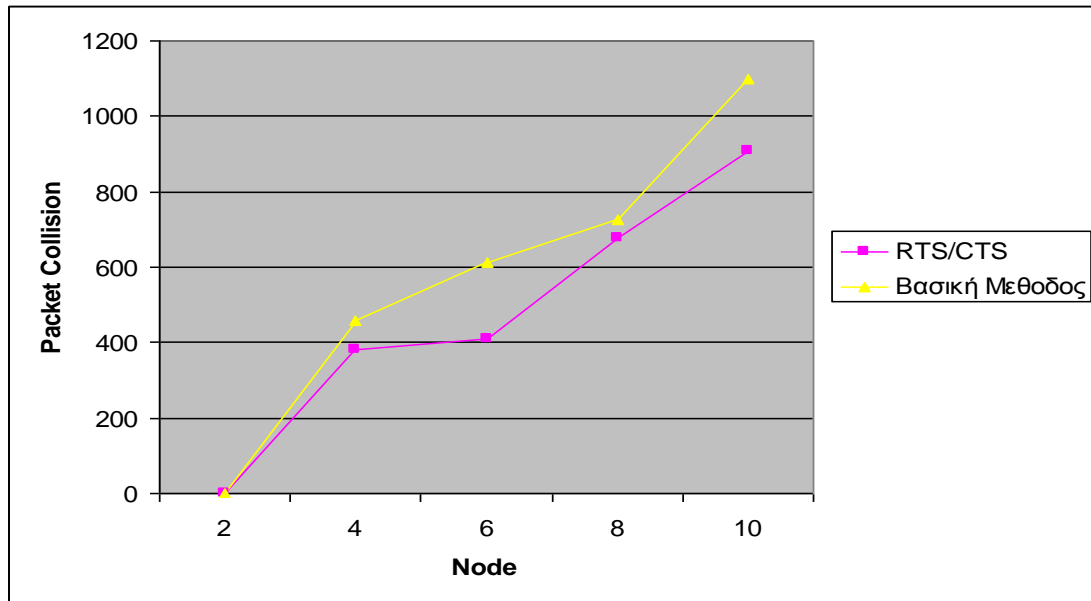
Στο διάγραμμα αυτό βλέπουμε πως κυμαίνεται η μέση καθυστέρηση για καθένα από τα σενάρια που τρέξαμε. Η καθυστέρηση στην RTS/CTS μέθοδο είναι μεγαλύτερη από την Βασική μέθοδο και αυτό λόγω του ότι η ρυθμοαπόδοση στην Βασική μέθοδο ήταν μεγαλύτερη από την μέθοδο RTS/CTS.

### 5.4.3 Διάγραμμα 3 . Σύνολο Απορριφθέντων Πακέτων (Drop Packet).



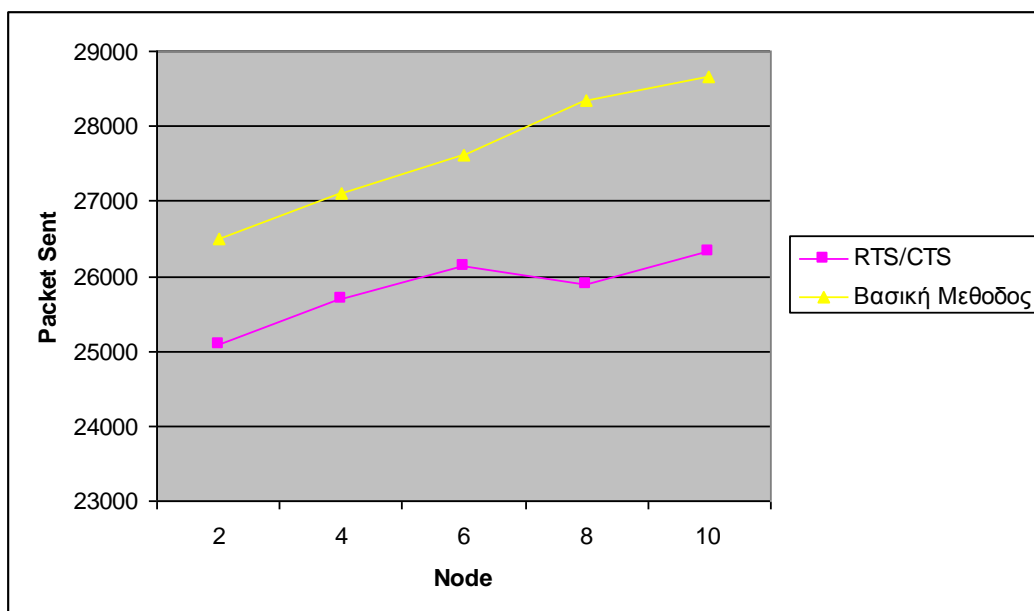
Στο διάγραμμα αυτό βλέπουμε πως κυμαίνεται ο αριθμός απορριφθέντων πακέτων για καθένα από τα σενάρια που τρέξαμε. Στην μέθοδο RTS/CTS απόρριψη μπορεί να γίνει μόνο σε RTS πακέτο , ενώ στην Βασική μέθοδο απόρριψη γίνεται σε TCP πακέτα.

#### 5.4.4 Διάγραμμα 4 . Σύνολο Συγκρούμενων Πακέτων(Collision Packet) .



Στο διάγραμμα αυτό βλέπουμε το αναμενόμενο .Όσο αυξάνεται ο αριθμός των κόμβων ανά σενάριο αυξάνεται και ο αριθμός των συγκρούσεων .Στην περίπτωση των 2 κόμβων όπως είναι προφανές δεν έχουμε καθόλου συγκρούσεις , γιατί δεν έχουμε καμία εξωτερική παρεμβολή , ούτε παρεμβολή από άλλους σταθμούς .

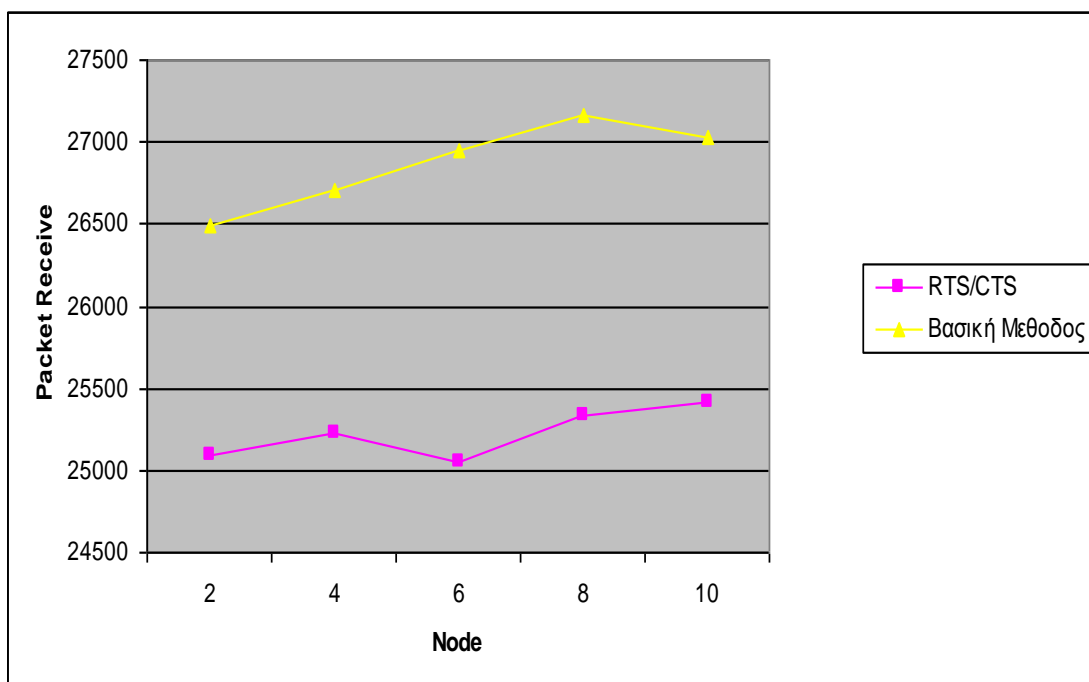
#### 5.4.5 Διάγραμμα 5 . Σύνολο Σταλμένων Πακέτων(Sent Packet).





Στο διάγραμμα αυτό βλέπουμε το αναμενόμενο .Τα αποσταλμένα πακέτα στην Βασική μέθοδο είναι πάντα περισσότερα από την RTS/CTS μέθοδο για κάθε σενάριο που τρέξαμε. Στην RTS/CTS μέθοδο παρατηρούμε ότι οι τιμές κινούνται στα ίδια πλαίσια για όλα τα σενάρια , αντίθετα στην Βασική μέθοδο έχουμε ανοδική πορεία των τιμών .

#### 5.4.6 Διάγραμμα 6 . Σύνολο Ληφθέντων Πακέτων(Receive Packet) .



Στο διάγραμμα αυτό βλέπουμε το αναμενόμενο .Τα ληφθέντα πακέτα στην Βασική μέθοδο είναι πάντα περισσότερα από την RTS/CTS μέθοδο για κάθε σενάριο που τρέξαμε. Στην RTS/CTS μέθοδο παρατηρούμε ότι οι τιμές είναι ανωδικές για όλα τα σενάρια με εξαίρεση το σενάριο 6 κόμβων. Βασική μέθοδο επίσης έχουμε ανωδική πορεία τιμών στα ληφθέντα πακέτα με εξαίρεση των 10 κόμβων που έχουμε μεγάλο αριθμό συγκρούσεων .

# Παράρτημα

Στο παράρτημα θα παρατεθούν οι κώδικες των 5 αρχείων προσομοίωσης αλλά και ενδεικτικά από ένα αρχείο με τις συνδέσεις και την θέση κάθε κόμβου .

## Σενάριο 2 κόμβων

```
# Define options
# ===== Print simulation start time =====
puts "\t<< time : [clock format [clock seconds]] >>"

# ===== Node parameters =====
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(rp) AODV ;# routing protocol

# ===== Topology parameter =====
set val(x) 1000 ;# X dimension of topology
set val(y) 1000 ;# Y dimension of topology
set val(nn) 2 ;# number of mobilenodes
set val(seed) 0.0 ;# move speed of node
set val(stop) 60 ;# simulation time

# Initialize the Link Layer Model
LL set mindelay_ 0us ;# 50us (default value)
LL set delay_ 0us ;# 25us (default value)
LL set bandwidth_ 0 ;# not used
LL set debug_ false
LL set avoidReordering_ false

# Initialize the MAC Layer Model
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023
Mac/802_11 set SlotTime_ 0.000020
Mac/802_11 set SIFS_ 0.000010
Mac/802_11 set PreambleLength_ 144
Mac/802_11 set PLCPHeaderLength_ 48
Mac/802_11 set PLCPDataRate_ 1.0e6 ;# 1Mbps
Mac/802_11 set RTSThreshold_ 0
Mac/802_11 set rtxLimit_ 6
Mac/802_11 set ShortRetryLimit_ 7
Mac/802_11 set LongRetryLimit_ 1
Mac/802_11 set basicRate_ 1Mb ;# 802.11 basic rate
```

```

Mac/802_11 set dataRate_          11Mb    ;# 802.11 data rate

# Other default settings
Agent/Null set sport_             0
Agent/Null set dport_             0

Agent/CBR set sport_              0
Agent/CBR set dport_              0

Agent/TCPSink set sport_          0
Agent/TCPSink set dport_          0

Agent/TCP set sport_              0
Agent/TCP set dport_              0
Agent/TCP set packetSize_        1023

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1

# unity gain, omni-directional antennas
# set up the antennas to be centered in the node and 1.5 meters above it
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

# Initialize the PHY Layer Model

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface
Phy/WirelessPhy set CPTthresh_ 10.0           ;# capture threshold (db)
Phy/WirelessPhy set CSTthresh_ 1.559e-11     ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10      ;# receive power threshold
Phy/WirelessPhy set bandwidth_ 2e6           ;# 802.11b = 2 Mbps
Phy/WirelessPhy set Pt_ 0.28183815          ;# transmitted signal power
Phy/WirelessPhy set freq_ 914e+6             ;# frequency
Phy/WirelessPhy set L_ 1.0                   ;# system loss factor

# Initialize Global Variables
set ns_ [new Simulator]

# **** Open trace file ****

set namfd [open 2-node.nam w]
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)
set tracefd [open 2-node.tr w]
$ns_ trace-all $tracefd

# **** Build a topology object ****
set topo [new Topography]

# **** topology size : 1000m x 1000m ****
$topo load_flatgrid $val(x) $val(y)

```

```

# **** create channel ***set udp($i) [new Agent/UDP]*
set chan [new $val(chan)]

# **** Create God ****
set god_ [create-god $val(nn)]

# **** Random Mode ****
$defaultRNG seed 1 ;# ON/OFF the ranodm mode. (seed 0 is ON)

# **** set node parameter ****
puts "ns_ node-config"

$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channel $chan

# Δημιουργία κόμβων

for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0 ;# disable random motion
}

# Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes
$node_(0) set X_ 250.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 400.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0

# **** set node size in nam ****

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 80
}

#Δημιουργία ροής δεδομένων από τον κόμβο 0 στον 1

set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]

```

```

$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 0.0 "$ftp start"

# Stop procedure

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}

$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 2-node.nam &

    exit 0
}

puts "Starting Simulation..."
$ns_ run

# ===== Print simulation stop time =====
puts "\t<< time : [clock format [clock seconds]] >>"

```

## Σενάριο 4 κόμβων

```

# Define options
# ===== Print simulation start time =====
puts "\t<< time : [clock format [clock seconds]] >>"

# ===== Node parameters =====
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(rp) AODV ;# routing protocol

# ===== Topology parameter =====
set val(x) 1000 ;# X dimension of topology
set val(y) 1000 ;# Y dimension of topology

```

```

set val(nn)          4          ;# number of mobilenodes
set val(seed)       0.0        ;# move speed of node
set val(stop)       60        ;# simulation time

# Initialize the Link Layer Model
LL set mindelay_    0us        ;# 50us (default value)
LL set delay_       0us        ;# 25us (default value)
LL set bandwidth_   0          ;# not used
LL set debug_       false
LL set avoidReordering_ false

# Initialize the MAC Layer Model
Mac/802_11 set CWMin_      31
Mac/802_11 set CWMax_     1023
Mac/802_11 set SlotTime_  0.000020
Mac/802_11 set SIFS_      0.000010
Mac/802_11 set PreambleLength_ 144
Mac/802_11 set PLCPHeaderLength_ 48
Mac/802_11 set PLCPDataRate_ 1.0e6 ;# 1Mbps
Mac/802_11 set RTSThreshold_ 0
Mac/802_11 set rtxLimit_  6
Mac/802_11 set ShortRetryLimit_ 7
Mac/802_11 set LongRetryLimit_ 1
Mac/802_11 set basicRate_ 1Mb ;# 802.11 basic rate
Mac/802_11 set dataRate_  11Mb ;# 802.11 data rate

# Other default settings
Agent/Null set sport_    0
Agent/Null set dport_    0

Agent/CBR set sport_     0
Agent/CBR set dport_     0

Agent/TCPSink set sport_  0
Agent/TCPSink set dport_  0

Agent/TCP set sport_     0
Agent/TCP set dport_     0
Agent/TCP set packetSize_ 1023

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1

# unity gain, omni-directional antennas
# set up the antennas to be centered in the node and 1.5 meters above it
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

# Initialize the PHY Layer Model

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface
Phy/WirelessPhy set CPThresh_ 10.0 ;# capture threshold (db)

```

```

Phy/WirelessPhy set CStresh_ 1.559e-11      ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10     ;# receive power threshold
Phy/WirelessPhy set bandwidth_ 2e6          ;# 802.11b = 2 Mbps
Phy/WirelessPhy set Pt_ 0.28183815         ;# transmitted signal power
Phy/WirelessPhy set freq_ 914e+6           ;# frequency
Phy/WirelessPhy set L_ 1.0                 ;# system loss factor

# Initialize Global Variables
set ns_ [new Simulator]

# **** Open trace file ****
set namfd [open 4-node.nam w]
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)
set tracefd [open 4-node.tr w]
$ns_ trace-all $tracefd

# **** Build a topology object ****
set topo [new Topography]

# **** topology size : 1000m x 1000m ****
$topo load_flatgrid $val(x) $val(y)

# **** create channel ***set udp($i) [new Agent/UDP]*
set chan [new $val(chan)]

# **** Create God ****
set god_ [create-god $val(nn)]

# **** Random Mode ****
$defaultRNG seed 1 ;# ON/OFF the ranodm mode. (seed 0 is ON)

# **** set node parameter ****
puts "ns_ node-config"
$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channel $chan

# Δημιουργία κόμβων

for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]

```

```

    $node_($i) random-motion 0 ;#  disable random motion
}

#  Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes
$node_(0) set X_ 250.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 400.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0

$node_(2) set X_ 275.0
$node_(2) set Y_ 425.0
$node_(2) set Z_ 0.0

$node_(3) set X_ 325.0
$node_(3) set Y_ 500.0
$node_(3) set Z_ 0.0

# **** set node size in nam ****

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 80
}

#Δημιουργία ροής δεδομένων από τον κόμβο 0 στον 1

set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 0.0 "$ftp start"

#Δημιουργία ροής δεδομένων από τον κόμβο 2 στον 3

set tcp1 [new Agent/TCP]
set sink1 [new Agent/TCPSink]
$ns_ attach-agent $node_(2) $tcp1
$ns_ attach-agent $node_(3) $sink1
$ns_ connect $tcp1 $sink1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns_ at 0.0 "$ftp1 start"

#  Stop procedure

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}

```



```

$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 4-node.nam &

        exit 0
}

```

```

puts "Starting Simulation..."
$ns_ run

```

```

# ===== Print simulation stop time =====
puts "\t<< time : [clock format [clock seconds]] >>"

```

## Σενάριο 6 κόμβων

```

# Define options
# ===== Print simulation start time =====
puts "\t<< time : [clock format [clock seconds]] >>"

# ===== Node parameters =====
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(rp) AODV ;# routing protocol

# ===== Topology parameter =====
set val(x) 1000 ;# X dimension of topology
set val(y) 1000 ;# Y dimension of topology
set val(nn) 6 ;# number of mobilenodes
set val(seed) 0.0 ;# move speed of node
set val(stop) 60 ;# simulation time

# Initialize the Link Layer Model
LL set mindelay_ 0us ;# 50us (default value)
LL set delay_ 0us ;# 25us (default value)
LL set bandwidth_ 0 ;# not used
LL set debug_ false
LL set avoidReordering_ false

# Initialize the MAC Layer Model
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023

```

```

Mac/802_11 set SlotTime_          0.000020
Mac/802_11 set SIFS_              0.000010
Mac/802_11 set PreambleLength_    144
Mac/802_11 set PLCPHeaderLength_  48
Mac/802_11 set PLCPDataRate_      1.0e6      ;# 1Mbps
Mac/802_11 set RTSThreshold_      0
Mac/802_11 set rtxLimit_          6
Mac/802_11 set ShortRetryLimit_   7
Mac/802_11 set LongRetryLimit_    1
Mac/802_11 set basicRate_         1Mb       ;# 802.11 basic rate
Mac/802_11 set dataRate_          11Mb      ;# 802.11 data rate

# Other default settings
Agent/Null set sport_             0
Agent/Null set dport_             0

Agent/CBR set sport_              0
Agent/CBR set dport_              0

Agent/TCPSink set sport_          0
Agent/TCPSink set dport_          0

Agent/TCP set sport_              0
Agent/TCP set dport_              0
Agent/TCP set packetSize_         1023

Queue/DropTail/PriQueue set Prefer_Routing_Protocols  1

# unity gain, omni-directional antennas
# set up the antennas to be centered in the node and 1.5 meters above it
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

# Initialize the PHY Layer Model

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface

Phy/WirelessPhy set CPTresh_ 10.0           ;# capture threshold (db)
Phy/WirelessPhy set CSTresh_ 1.559e-11     ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10    ;# receive power threshold
Phy/WirelessPhy set bandwidth_ 2e6         ;# 802.11b = 2 Mbps
Phy/WirelessPhy set Pt_ 0.28183815        ;# transmitted signal power
Phy/WirelessPhy set freq_ 914e+6          ;# frequency
Phy/WirelessPhy set L_ 1.0                ;# system loss factor

# Initialize Global Variables
set ns_ [new Simulator]

# **** Open trace file ****
set namfd [open 6-node.nam w]
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)

```

```

set tracefd [open 6-node.tr w]
$ns_ trace-all $tracefd

# **** Build a topology object ****
set topo [new Topography]

# **** topology size : 1000m x 1000m ****
$topo load_flatgrid $val(x) $val(y)

# **** create channel ***set udp($i) [new Agent/UDP]*
set chan [new $val(chan)]

# **** Create God ****
set god_ [create-god $val(nn)]

# **** Random Mode ****
$defaultRNG seed 1 ;# ON/OFF the random mode. (seed 0 is ON)

# **** set node parameter ****
puts "ns_ node-config"

$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channel $chan

# Δημιουργία κόμβων

for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0 ;# disable random motion
}

# Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes
$node_(0) set X_ 250.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 400.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0

$node_(2) set X_ 275.0
$node_(2) set Y_ 425.0

```

```
$node_(2) set Z_ 0.0
```

```
$node_(3) set X_ 325.0  
$node_(3) set Y_ 500.0  
$node_(3) set Z_ 0.0
```

```
$node_(4) set X_ 350.0  
$node_(4) set Y_ 450.0  
$node_(4) set Z_ 0.0
```

```
$node_(5) set X_ 475.0  
$node_(5) set Y_ 475.0  
$node_(5) set Z_ 0.0
```

```
# **** set node size in nam ****
```

```
for {set i 0} {$i < $val(nn)} {incr i} {  
    $ns_ initial_node_pos $node_($i) 80  
}
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 0 στον 1
```

```
set tcp [new Agent/TCP]  
set sink [new Agent/TCPSink]  
$ns_ attach-agent $node_(0) $tcp  
$ns_ attach-agent $node_(1) $sink  
$ns_ connect $tcp $sink  
set ftp [new Application/FTP]  
$ftp attach-agent $tcp  
$ns_ at 0.0 "$ftp start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 2 στον 3
```

```
set tcp1 [new Agent/TCP]  
set sink1 [new Agent/TCPSink]  
$ns_ attach-agent $node_(2) $tcp1  
$ns_ attach-agent $node_(3) $sink1  
$ns_ connect $tcp1 $sink1  
set ftp1 [new Application/FTP]  
$ftp1 attach-agent $tcp1  
$ns_ at 0.0 "$ftp1 start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 4 στον 5
```

```
set tcp2 [new Agent/TCP]  
set sink2 [new Agent/TCPSink]  
$ns_ attach-agent $node_(4) $tcp2  
$ns_ attach-agent $node_(5) $sink2  
$ns_ connect $tcp2 $sink2  
set ftp2 [new Application/FTP]  
$ftp2 attach-agent $tcp2  
$ns_ at 0.0 "$ftp2 start"
```

```

# Stop procedure

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}

$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 6-node.nam &

    exit 0
}
puts "Starting Simulation..."
$ns_ run

# ===== Print simulation stop time =====
puts "\t<< time : [clock format [clock seconds]] >>"

```

## Σενάριο 8 κόμβων

```

# Define options
# ===== Print simulation start time =====
puts "\t<< time : [clock format [clock seconds]] >>"

# ===== Node parameters =====
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(rp) AODV ;# routing protocol

# ===== Topology parameter =====
set val(x) 1000 ;# X dimension of topology
set val(y) 1000 ;# Y dimension of topology
set val(nn) 8 ;# number of mobilenodes
set val(seed) 0.0 ;# move speed of node
set val(stop) 60 ;# simulation time

# Initialize the Link Layer Model
LL set mindelay_ 0us ;# 50us (default value)
LL set delay_ 0us ;# 25us (default value)
LL set bandwidth_ 0 ;# not used
LL set debug_ false

```

```

LL set avoidReordering_ false

# Initialize the MAC Layer Model
Mac/802_11 set CWMin_          31
Mac/802_11 set CWMax_          1023
Mac/802_11 set SlotTime_       0.000020
Mac/802_11 set SIFS_           0.000010
Mac/802_11 set PreambleLength_ 144
Mac/802_11 set PLCPHeaderLength_ 48
Mac/802_11 set PLCPDataRate_   1.0e6      ;# 1Mbps
Mac/802_11 set RTSThreshold_    0
Mac/802_11 set rtxLimit_        6
Mac/802_11 set ShortRetryLimit_ 7
Mac/802_11 set LongRetryLimit_  1
Mac/802_11 set basicRate_       1Mb       ;# 802.11 basic rate
Mac/802_11 set dataRate_        11Mb      ;# 802.11 data rate

# Other default settings
Agent/Null set sport_          0
Agent/Null set dport_          0

Agent/CBR set sport_           0
Agent/CBR set dport_           0

Agent/TCPSink set sport_        0
Agent/TCPSink set dport_        0

Agent/TCP set sport_           0
Agent/TCP set dport_           0
Agent/TCP set packetSize_      1023

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1

# unity gain, omni-directional antennas
# set up the antennas to be centered in the node and 1.5 meters above it
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

# Initialize the PHY Layer Model

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface

Phy/WirelessPhy set CPTresh_ 10.0           ;# capture threshold (db)
Phy/WirelessPhy set CSTresh_ 1.559e-11     ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10    ;# receive power threshold
Phy/WirelessPhy set bandwidth_ 2e6         ;# 802.11b = 2 Mbps
Phy/WirelessPhy set Pt_ 0.28183815        ;# transmitted signal power
Phy/WirelessPhy set freq_ 914e+6          ;# frequency
Phy/WirelessPhy set L_ 1.0                 ;# system loss factor

```

```

# Initialize Global Variables
set ns_ [new Simulator]

# **** Open trace file ****
set namfd [open 8-node.nam w]
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)
set tracefd [open 8-node.tr w]
$ns_ trace-all $tracefd

# **** Build a topology object ****
set topo [new Topography]

# **** topology size : 1000m x 1000m ****
$topo load_flatgrid $val(x) $val(y)

# **** create channel ***set udp($i) [new Agent/UDP]*
set chan [new $val(chan)]

# **** Create God ****
set god_ [create-god $val(nn)]

# **** Random Mode ****
$defaultRNG seed 1 ;# ON/OFF the ranodm mode. (seed 0 is ON)

# **** set node parameter ****
puts "ns_ node-config"

$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -topoInstance $topo \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -movementTrace OFF \
    -channel $chan

# Δημιουργία κόμβων

for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0 ;# disable random motion
}

# Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes

```

```
$node_(0) set X_ 250.0
$node_(0) set Y_ 500.0
$node_(0) set Z_ 0.0
```

```
$node_(1) set X_ 400.0
$node_(1) set Y_ 500.0
$node_(1) set Z_ 0.0
```

```
$node_(2) set X_ 275.0
$node_(2) set Y_ 425.0
$node_(2) set Z_ 0.0
```

```
$node_(3) set X_ 325.0
$node_(3) set Y_ 500.0
$node_(3) set Z_ 0.0
```

```
$node_(4) set X_ 350.0
$node_(4) set Y_ 450.0
$node_(4) set Z_ 0.0
```

```
$node_(5) set X_ 475.0
$node_(5) set Y_ 475.0
$node_(5) set Z_ 0.0
```

```
$node_(6) set X_ 275.0
$node_(6) set Y_ 580.0
$node_(6) set Z_ 0.0
```

```
$node_(7) set X_ 375.0
$node_(7) set Y_ 565.0
$node_(7) set Z_ 0.0
```

```
# **** set node size in nam ****
```

```
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 80
}
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 0 στον 1
```

```
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 0.0 "$ftp start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 2 στον 3
```

```
set tcp1 [new Agent/TCP]
set sink1 [new Agent/TCPSink]
$ns_ attach-agent $node_(2) $tcp1
```



```

$ns_ attach-agent $node_(3) $sink1
$ns_ connect $tcp1 $sink1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns_ at 0.0 "$ftp1 start"

```

#Δημιουργία ροής δεδομένων από τον κόμβο 4 στον 5

```

set tcp2 [new Agent/TCP]
set sink2 [new Agent/TCPSink]
$ns_ attach-agent $node_(4) $tcp2
$ns_ attach-agent $node_(5) $sink2
$ns_ connect $tcp2 $sink2
set ftp2 [new Application/FTP]
$ftp2 attach-agent $tcp2
$ns_ at 0.0 "$ftp2 start"

```

#Δημιουργία ροής δεδομένων από τον κόμβο 6 στον 7

```

set tcp3 [new Agent/TCP]
set sink3 [new Agent/TCPSink]
$ns_ attach-agent $node_(7) $tcp3
$ns_ attach-agent $node_(6) $sink3
$ns_ connect $tcp3 $sink3
set ftp3 [new Application/FTP]
$ftp3 attach-agent $tcp3
$ns_ at 0.0 "$ftp3 start"

```

# Stop procedure

```

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}

```

```

$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"

```

```

proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 8-node.nam &

    exit 0
}

```

```

puts "Starting Simulation..."
$ns_ run

```

```

# ===== Print simulation stop time =====
puts "\t<< time : [clock format [clock seconds]] >>"

```

## Σενάριο 10 κόμβων

```
# Define options
# ===== Print simulation start time =====
puts "\t<< time : [clock format [clock seconds]] >>"

# ===== Node parameters =====
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(rp) AODV ;# routing protocol

# ===== Topology parameter =====
set val(x) 1000 ;# X dimension of topology
set val(y) 1000 ;# Y dimension of topology
set val(nn) 10 ;# number of mobilenodes
set val(seed) 0.0 ;# move speed of node
set val(stop) 60 ;# simulation time

# Initialize the Link Layer Model
LL set mindelay_ 0us ;# 50us (default value)
LL set delay_ 0us ;# 25us (default value)
LL set bandwidth_ 0 ;# not used
LL set debug_ false
LL set avoidReordering_ false

# Initialize the MAC Layer Model
Mac/802_11 set CWMin_ 31
Mac/802_11 set CWMax_ 1023
Mac/802_11 set SlotTime_ 0.000020
Mac/802_11 set SIFS_ 0.000010
Mac/802_11 set PreambleLength_ 144
Mac/802_11 set PLCPHeaderLength_ 48
Mac/802_11 set PLCPDataRate_ 1.0e6 ;# 1Mbps
Mac/802_11 set RTSThreshold_ 0
Mac/802_11 set rtxLimit_ 6
Mac/802_11 set ShortRetryLimit_ 7
Mac/802_11 set LongRetryLimit_ 1
Mac/802_11 set basicRate_ 1Mb ;# 802.11 basic rate
Mac/802_11 set dataRate_ 11Mb ;# 802.11 data rate

# Other default settings
Agent/Null set sport_ 0
Agent/Null set dport_ 0

Agent/CBR set sport_ 0
Agent/CBR set dport_ 0

Agent/TCPSink set sport_ 0
```

```

Agent/TCP sink set dport_ 0

Agent/TCP set sport_ 0
Agent/TCP set dport_ 0
Agent/TCP set packetSize_ 1023

Queue/DropTail/PriQueue set Prefer_Routing_Protocols 1

# unity gain, omni-directional antennas
# set up the antennas to be centered in the node and 1.5 meters above it
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

# Initialize the PHY Layer Model

# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface

Phy/WirelessPhy set CPTresh_ 10.0 ;# capture threshold (db)
Phy/WirelessPhy set CSTresh_ 1.559e-11 ;# carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10 ;# receive power threshold
Phy/WirelessPhy set bandwidth_ 2e6 ;# 802.11b = 2 Mbps
Phy/WirelessPhy set Pt_ 0.28183815 ;# transmitted signal power
Phy/WirelessPhy set freq_ 914e+6 ;# frequency
Phy/WirelessPhy set L_ 1.0 ;# system loss factor

# Initialize Global Variables
set ns_ [new Simulator]

# **** Open trace file ****
set namfd [open 10-node.nam w]
$ns_ namtrace-all-wireless $namfd $val(x) $val(y)
set tracefd [open 10-node.tr w]
$ns_ trace-all $tracefd

# **** Build a topology object ****
set topo [new Topography]

# **** topology size : 1000m x 1000m ****
$topo load_flatgrid $val(x) $val(y)

# **** create channel ***set udp($i) [new Agent/UDP]*
set chan [new $val(chan)]

# **** Create God ****
set god_ [create-god $val(nn)]

# **** Random Mode ****
$defaultRNG seed 1 ;# ON/OFF the random mode. (seed 0 is ON)

# **** set node parameter ****

```

```
puts "ns_ node-config"
```

```
$ns_ node-config -adhocRouting $val(rp) \  
    -llType $val(ll) \  
    -macType $val(mac) \  
    -ifqType $val(ifq) \  
    -ifqLen $val(ifqlen) \  
    -antType $val(ant) \  
    -propType $val(prop) \  
    -phyType $val(netif) \  
    -topoInstance $topo \  
    -agentTrace OFF \  
    -routerTrace OFF \  
    -macTrace ON \  
    -movementTrace OFF \  
    -channel $chan
```

```
# Δημιουργία κόμβων
```

```
for {set i 0} {$i < $val(nn)} {incr i} {  
    set node_($i) [$ns_ node]  
    $node_($i) random-motion 0 ;#    disable random motion  
}
```

```
# Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes
```

```
$node_(0) set X_ 250.0  
$node_(0) set Y_ 500.0  
$node_(0) set Z_ 0.0
```

```
$node_(1) set X_ 400.0  
$node_(1) set Y_ 500.0  
$node_(1) set Z_ 0.0
```

```
$node_(2) set X_ 275.0  
$node_(2) set Y_ 425.0  
$node_(2) set Z_ 0.0
```

```
$node_(3) set X_ 325.0  
$node_(3) set Y_ 500.0  
$node_(3) set Z_ 0.0
```

```
$node_(4) set X_ 350.0  
$node_(4) set Y_ 450.0  
$node_(4) set Z_ 0.0
```

```
$node_(5) set X_ 475.0  
$node_(5) set Y_ 475.0  
$node_(5) set Z_ 0.0
```

```
$node_(6) set X_ 275.0  
$node_(6) set Y_ 580.0  
$node_(6) set Z_ 0.0
```

```
$node_(7) set X_ 375.0
$node_(7) set Y_ 565.0
$node_(7) set Z_ 0.0
```

```
$node_(8) set X_ 425.0
$node_(8) set Y_ 425.0
$node_(8) set Z_ 0.0
```

```
$node_(9) set X_ 450.0
$node_(9) set Y_ 550.0
$node_(9) set Z_ 0.0
```

```
# **** set node size in nam ****
```

```
for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 80
}
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 0 στον 1
```

```
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 0.0 "$ftp start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 2 στον 3
```

```
set tcp1 [new Agent/TCP]
set sink1 [new Agent/TCPSink]
$ns_ attach-agent $node_(2) $tcp1
$ns_ attach-agent $node_(3) $sink1
$ns_ connect $tcp1 $sink1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns_ at 0.0 "$ftp1 start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 4 στον 5
```

```
set tcp2 [new Agent/TCP]
set sink2 [new Agent/TCPSink]
$ns_ attach-agent $node_(4) $tcp2
$ns_ attach-agent $node_(5) $sink2
$ns_ connect $tcp2 $sink2
set ftp2 [new Application/FTP]
$ftp2 attach-agent $tcp2
$ns_ at 0.0 "$ftp2 start"
```

```
#Δημιουργία ροής δεδομένων από τον κόμβο 7 στον 6
```

```

set tcp3 [new Agent/TCP]
set sink3 [new Agent/TCPSink]
$ns_ attach-agent $node_(7) $tcp3
$ns_ attach-agent $node_(6) $sink3
$ns_ connect $tcp3 $sink3
set ftp3 [new Application/FTP]
$ftp3 attach-agent $tcp3
$ns_ at 0.0 "$ftp3 start"

```

#Δημιουργία ροής δεδομένων από τον κόμβο 8 στον 9

```

set tcp4 [new Agent/TCP]
set sink4 [new Agent/TCPSink]
$ns_ attach-agent $node_(8) $tcp4
$ns_ attach-agent $node_(9) $sink4
$ns_ connect $tcp4 $sink4
set ftp4 [new Application/FTP]
$ftp4 attach-agent $tcp4
$ns_ at 0.0 "$ftp4 start"

```

# Stop procedure

```

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "$node_($i) reset";
}

$ns_ at $val(stop) "stop"
$ns_ at $val(stop) "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd namfd
    $ns_ flush-trace
    close $tracefd
    close $namfd
    exec nam 10-node.nam &

    exit 0
}

```

puts "Starting Simulation..."

\$ns\_ run

```

# ===== Print simulation stop time =====
puts "\t<< time : [clock format [clock seconds]] >>"

```

## Βιβλιογραφία

- [1] Π. Ράππης ,(Φεβρουάριος 2010) .Διδακτορική διατριβή .
- [2] G. Bianchi, "IEEE 802.11—Saturation throughput analysis," *IEEE Commun. Lett.*, vol. 2, pp. 318–320.
- [3] G. Bianchi, L. Fratta, and M. Oliveri, "Performance analysis of IEEE 802.11 CSMA/CA medium access control protocol," in *Proc. IEEE PIMRC*, Taipei, Taiwan, Oct. 1996, pp. 407–411.
- [4] G Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Comm.* V 18. N3, March 2000
- [5] Wu, H., Cheng , S., Peng., Y., Long, K., Ma, J. (2002b) IEEE 802.11 Distributed Coordination Function (DCF) : Analysis and Enhancement . In : Proceedings of the IEEE International Conference on Communications (ICC).
- [6] H. Wu, Y. Peng, K. Long, S. Cheng, "A simple model of IEEE 802.11 wireless LAN", *Proceedings of ICII 2001*, pp.514-519, Oct. 2001
- [7] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - ANSI/IEEE Std 802.11", 1999 Edition [ISO/IEC 8802-11: 1999]
- [8] Douglas E. Comer, «Δίκτυα και Διαδίκτυα υπολογιστών », 3<sup>η</sup> έκδοση, Κλειδάριθμος, 2003
- [9] Α. Αλεξόπουλος, Γ. Λαγογιαννης, «Τηλεπικοινωνίες και Δίκτυα Υπολογιστών», 5<sup>η</sup> έκδοση, 1999
- [10] "IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", Nov. 1997. P802.11
- [11] IEEE 802.15 TG6 (2007) WPAN Task Group 6. Body Area Networks (BAN). Ανάκτηση 15/11/2010 από <http://www.ieee802.org/15/pub/TG6.html>
- [12] IEEE 802.15.1 TG1a (2007) WPAN Task Group 1a. Body Area Networks (BAN). Ανάκτηση 15/11/2010 από <http://www.ieee802.org/15/pub/TG1a.html>

- [13] Mouhamad IBRAHIM and Giovanni NEGLIA “Introduction to Network Simulator”
- [14] <http://www.isi.edu/nsnam/ns/>
- [15] <http://www.isi.edu/nsnam/ns/tutorial/index.html>
- [16] <http://allaboutns2.wordpress.com/2009/07/24/%CE%B5%CE%B3%CE%BA%CE%B1%CF%84%CE%AC%CF%83%CF%84%CE%B1%CF%83%CE%B7-%CF%84%CE%BF%CF%85-ns-2/#more-34>
- [17] <http://www.wifi-doc.com/O.Reilly-802.11.Wireless.Netwo/0596100523/wireless802dot112-chp-11-sect-3.html>
- [18] <http://ckp.made-it.com/ieee80211.html>
- [19] <http://www.intel.com/support/wireless/wlan/sb/CS-025317.htm>
- [20] [http://www.joshuarobinson.net/docs/ns-802\\_11b.html](http://www.joshuarobinson.net/docs/ns-802_11b.html)
- [21] [http://read.pudn.com/downloads71/sourcecode/unix\\_linux/network/254894/wireless/PerferencelIssues-fig4.tcl\\_\\_.htm](http://read.pudn.com/downloads71/sourcecode/unix_linux/network/254894/wireless/PerferencelIssues-fig4.tcl__.htm)
- [22] <http://www.scalable-networks.com/boards/viewtopic.php?f=27&t=1653>
- [23] <http://www.cs.binghamton.edu/~kliu/research/ns2code/note.pdf>
- [24] <http://code.google.com/p/hsbt/source/browse/trunk/HSBT/ns-2.29/ns-mac.tcl?r=117>
- [25] <http://www.cs.binghamton.edu/~kliu/research/ns2code/#trace>
- [26] [http://dcalab.csie.thu.edu.tw/wiki/ns-2.32/class\\_mac\\_csma.html](http://dcalab.csie.thu.edu.tw/wiki/ns-2.32/class_mac_csma.html)
- [27] <http://lovefei.com/?p=64>
- [28] [http://dcalab.csie.thu.edu.tw/wiki/ns-2.32/class\\_mac\\_csma.html#0086bc55ca4efcf6a14daec30c29e711](http://dcalab.csie.thu.edu.tw/wiki/ns-2.32/class_mac_csma.html#0086bc55ca4efcf6a14daec30c29e711)