

## ΠΕΡΙΕΧΟΜΕΝΑ

<b>1.Εισαγωγή.....</b>	<b>1</b>
1.1. Ετερογενή Δίκτυα	
<b>2. Πρακτικά Θέματα πάνω σε κυψελωτά Ε.Δ.....</b>	<b>2</b>
2.1 Κύριοι σταθμοί βάσης (Layer) κυψελοειδούς δικτύου	
2.2 Οι κυρίες δυσκολίες και τα προβλήματα που παρουσιάζονται	
2.3 Ο ρόλος του ετερογενούς δικτύου είναι να επιτυγχάνει	
<b>3. Προσομοίωση (Simulation).....</b>	<b>6</b>
3.1 Αναγκαιότητα για προσομοίωση Δικτύων	
3.2 Προσομοίωση Δικτύου	
3.3 Πλεονεκτήματα προσομοίωσης Δικτύου	
3.4 Μειονεκτήματα προσομοίωσης Δικτύου	
<b>4. Εξομοίωση Δικτύου (Emulation).....</b>	<b>7</b>
<b>5. Εισαγωγή στους Προσομοιωτές.....</b>	<b>8</b>
5.1 Εκτίμηση Επίδοσης	
5.2 Προγραμματίζοντας μια Προσομοίωση	
<b>6. Ανάλυση των κύριων εργαλείων προσομοίωσης.....</b>	<b>9</b>
6.1 NS-2 (Network Simulator version2)	
6.2 NS-3 (Network Simulator version3)	
6.3 OPNET Modeler (Optimized Network Engineering Tools)	
6.4 NETSIM (Network Based Environment for Modeling and Simulation)	
6.5 OMNeT++ (Optical Micro-Networks Plus Plus)	
6.6 J-Sim (Java-based simulation)	
6.7 REAL (Realistic and Large)	
6.8 Λοιποί Προσομοιωτές	
6.9 Αναφορά & Ταξινόμηση των βασικών εργαλείων προσομοίωσης με βάση τα διάφορα διακριτά τους χαρακτηριστικά	
<b>7. Προσομοίωση μεγάλης Κλίμακας.....</b>	<b>20</b>
7.1 NS2 to Web	
7.2 AKARI	
7.3 PlanetLab	
7.4 The Global Environment for Network Innovations (GENI)	
7.5 Future Internet Research and Experimentation(FIRE)	
<b>8. Εισαγωγή στο περιβάλλον του NS2 με προσομοίωση απλού κώδικα.....</b>	<b>24</b>
8.1 Κύρια μέρη στον κώδικα	
8.2 Ίχνος πακέτων της προσομοίωσης	
<b>9. Υλοποίηση προσομοίωσης - Throughput Δεδομένων στο 802.11.....</b>	<b>28</b>
9.1 Ορισμοί Bandwidth και Throughput	
9.2 Προσομοίωση 1ο Μέρος	
9.2.1 Περιγραφή κώδικα στο 1 <sup>ο</sup> μέρος προσομοίωσης	
9.2.2. Η Συνολική προσφερόμενη κίνηση	

9.2.3. Η Επιρροή του Data Rate πάνω στο Throughput

9.2.4. Διαδικασία μέτρησης αποτελεσμάτων

9.3 Προσομοίωση 2ο Μέρος

9.3.1. Διαδικασία μέτρησης αποτελεσμάτων

**Βιβλιογραφική Αναφορά.....39**

## **1.Εισαγωγή**

### **1.1 Ετερογενή Δίκτυα:**

Τα ετερογενή δίκτυα αποτελούν δίκτυα τα οποία μπορεί να περιλαμβάνουν τερματικές συσκευές αλλά και στοιχεία δικτύου διαφορετικού τύπου (π.χ. προηγούμενα κινητά τηλέφωνα, φορητοί υπολογιστές, δρομολογητές), να συνδυάζουν διαφορετικές τεχνολογίες επικοινωνίας και πρόσβασης (π.χ. 3G δίκτυα, 802.11 ασύρματα δίκτυα, δίκτυα οπτικών ινών) και να υποστηρίζουν διαφορετικές υπηρεσίες ανά τύπο χρηστών. Κατά την παροχή υπηρεσιών σε ετερογενή δίκτυα, είναι σημαντικό να υπάρχει δυνατότητα καθορισμού των υπηρεσιών που είναι διαθέσιμες σε κάθε χρήστη ανάλογα με τα χαρακτηριστικά του (π.χ. υποστήριξη κινητικότητας). Ανάλογα με τις επικρατούσες συνθήκες στο δίκτυο και τις προτιμήσεις του εκάστοτε χρήστη, η κατάλληλη υπηρεσία και το κατάλληλο μέσο πρόσβασης επιλέγεται με στόχο την καλύτερη παροχή ποιότητας υπηρεσίας.

Η ευρεία διάδοση του διαδικτύου, σε συνδυασμό με τις εξελίξεις που επιτρέπουν τη διασύνδεση ετερογενών συσκευών και δικτύων, διευκολύνουν το σχεδιασμό μίας νέας γενιάς δικτυακών υπηρεσιών. Συστατικό στοιχείο αυτών των υπηρεσιών αποτελεί η συνεργασία ανάμεσα στους κόμβους του δικτύου και η ανταλλαγή χρήσιμης πληροφορίας με απώτερο στόχο την αξιοποίηση παροχή υπηρεσιών. Κάθε κόμβος του δικτύου μπορεί να αποκτήσει πρόσβαση στο περιεχόμενο που τον ενδιαφέρει και τελικά να παρέχει κάποια υπηρεσία. Παράλληλα, η ευφυΐα μπορεί να διανεμηθεί σε όλο το δίκτυο και επομένως όλοι οι κόμβοι να είναι ικανοί να αισθάνονται τις αλλαγές που συμβαίνουν και να λαμβάνουν κατάλληλες αποφάσεις, σχετικές με την αποδοτικότερη παροχή υπηρεσιών ή τη βελτιστοποίηση των υποστηριζόμενων λειτουργιών στο δίκτυο.

### **Ορισμός :**

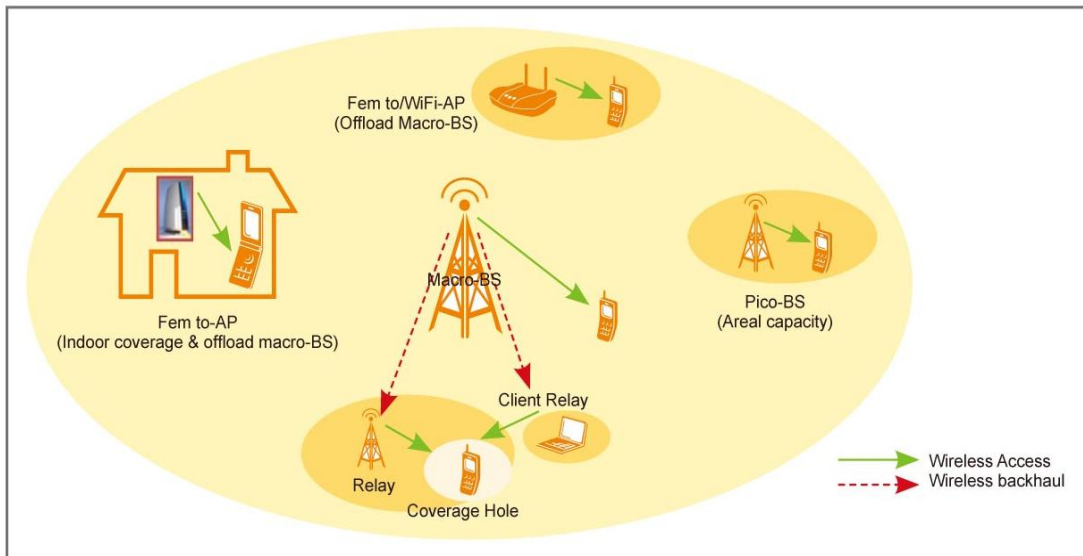
Ετερογενή δίκτυο, είναι το δίκτυο όπου υπάρχει η κατάλληλη συνδεσμολογία διαφορετικών τεχνολογιών δικτύων για την παροχή υπηρεσιών στις κινητές ή ηλεκτρονικές συσκευές (υπολογιστές) με διαφορετικά λειτουργικά συστήματα ή/και πρωτόκολλα επικοινωνίας μεταξύ τους.

### **Παράδειγμα:**

Σε ένα τοπικό δίκτυο (LANs) το οποίο συνδέει υπολογιστές με λειτουργικά συστήματα Windows, Linux και Mac ή αντίστοιχα να συνδέει Pc's με Apple Macintosh υπολογιστές, λέγεται ετερογενή Δίκτυο.

Η έννοια του ετερογενούς δικτύου, είναι επίσης ευρέως χρησιμοποιούμενη στην **ασύρματη δικτύωση**, όπου γίνεται χρήση διαφορετικών τεχνολογιών προσβασιμότητας. Παράδειγμα, ένα ασύρματο δίκτυο το οποίο παρέχει μια υπηρεσία και είναι σε θέση να διατηρήσει την υπηρεσία αυτήν κατά τη μετάβαση ενός κινητού σε ένα κυψελοειδές δίκτυο (σύστημα), ονομάζεται ασύρματο ετερογενές δίκτυο.

Εικόνα ετερογενούς δικτύου:



## 2. Πρακτικά Θέματα πάνω σε κυψελωτά Ε.Δ.

Ετερογενή δίκτυα επίσης, είναι ένας συνδυασμός μικρών και μεγάλων κυψελωτών συστημάτων (Macro / Pico / Femto ) με διαφορετικές τεχνολογίες μετάδοσης πληροφορίας (3G/LTE, LTE advanced), όπου όλα συνεργάζονται μαζί για να παρέχουν την καλύτερη δυνατή κάλυψη και χωρητικότητα χρηστών.

### 2.1 Κύριοι σταθμοί βάσης (Layer) κυψελωτού δικτύου:

- **Macro Cell:** Macro κυψέλες ή Macro Layer (επίπεδο) είναι ένα υπαίθριο σταθμό βάσης, η οποία είναι ανοικτή για το κοινό (public). Είναι το κεντρικό σταθμό βάσης στην κυψέλη, η εκπεμπόμενη ισχύς κυμαίνεται περίπου στα 43 dBm. Οι Macro κυψέλες συνδέονται μεταξύ τους χρησιμοποιώντας μια σύνδεση **\*backhaul**. Ωστόσο, η σύνδεση εξαρτάται από τη διαθέσιμη υποδομή. Οι macro κυψέλες χρησιμοποιούνται για να καλύψουν ευρύτερες εκτάσεις μεγάλων μεγεθών. Παράδειγμα, οι macrocells βρίσκονται σε αγροτικές περιοχές ή κατά μήκος των αυτοκινητοδρόμων.

( \* Σε ένα ιεραρχικό δίκτυο τηλεπικοινωνιών το τμήμα backhaul του δικτύου περιλαμβάνει τις ενδιάμεσες συνδέσεις μεταξύ του πυρήνα του δικτύου, ή του δικτύου κορμού και τα μικρά υποδίκτυα στην «άκρη» του συνόλου της ιεραρχίας του δικτύου.)

- **Pico Cell:** οι Pico κυψέλες είναι σίγουρα μικρότερο από τις Macro κυψέλες, η μεταδιδόμενη ισχύς είναι περίπου 23dBm με 30dBm. Είναι επίσης ένα υπαίθριο

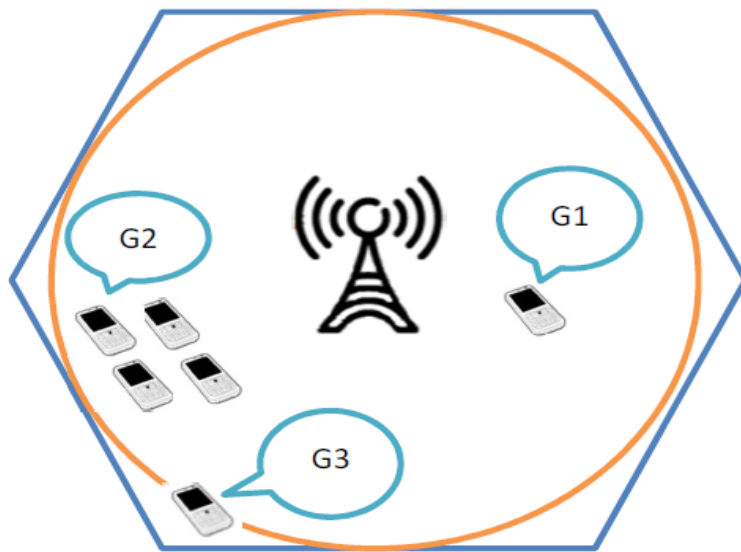
σταθμό βάσης, ανοικτή στο κοινό, και έχει μια μικρή κάλυψη. Συνήθως τα Picocell περιλαμβάνονται στο εσωτερικό του στρώματος Macro Cell, χρησιμοποιούνται για να ενισχύσουν τις αδύναμες περιοχές όπου υπάρχουν παρεμβολές ή να αυξήσουν την χωρητικότητα στις αστικές περιοχές. Η έννοια της backhauling ισχύει το ίδιο όπως στο Macro Cell επίπεδο.

- **Femto Cell:** οι Femto κυψέλες, σαφώς από το όνομα είναι μικρής εμβέλειας εδάφους σε σχέση με Macro και Pico Cells. Οι Femto Cell είναι εσωτερικές μονάδες ή σταθμοί βάσης σπιτιού ανοικτής ή περιορισμένης πρόσβασης. Η μεταδιδόμενη ισχύς είναι χαμηλή, μικρότερη από 23dBm. Ένα πλεονέκτημα αυτού του τύπου είναι ότι μπορεί να αυξήσει την κυκλοφορία δεδομένων επειδή η απόσταση μεταξύ των μονάδων (συσκευών) είναι μικρή. Παράδειγμα, στο ίδιο κτίριο υπάρχει σύνδεση υψηλής ταχύτητας, άλλωστε το μικρό εύρος αυξάνει την διάρκεια ζωής της μπαταριών σταθμών βάσης (στην περίπτωση όπου δεν έχουμε ηλεκτρική τροφοδοσία). Επίσης, αυτό το είδος κυψέλης μειώνει το συνολικό κόστος από την διαχειριστική πλευρά του φορέα, διότι ο δεν υπάρχει τεράστιο κόστος κατά την εγκατάσταση των σταθμών βάσης. Προβλήματα που δημιουργούνται στις οποίες υπάρχουν λύσεις αντιμετώπισής τους, είναι όπως η περιορισμένη πρόσβαση στο σταθμό βάσης, απώθηση, επαναχρησιμοποίηση συχνοτήτων σε μικρές αποστάσεις με συνέπεια να συμβαίνουν και παρεμβολές.

Τα Ετερογενή Δίκτυα αποτελούν συνδυασμό πολλών κυτταρικών κυψελών διαφορετικού τύπου και τεχνολογιών που υποστηρίζουν, προσπαθώντας να βελτιώσουν το στρώμα (Layer) των κυψελών, προκειμένου να ανταποκριθεί στην υψηλή κίνηση δικτύου και να διορθώσει τις περιοχές αδυναμίας μέσα στα ίδια στρώμα επιπέδου, αλλά και να αυξήσει η συνολική απόδοση.

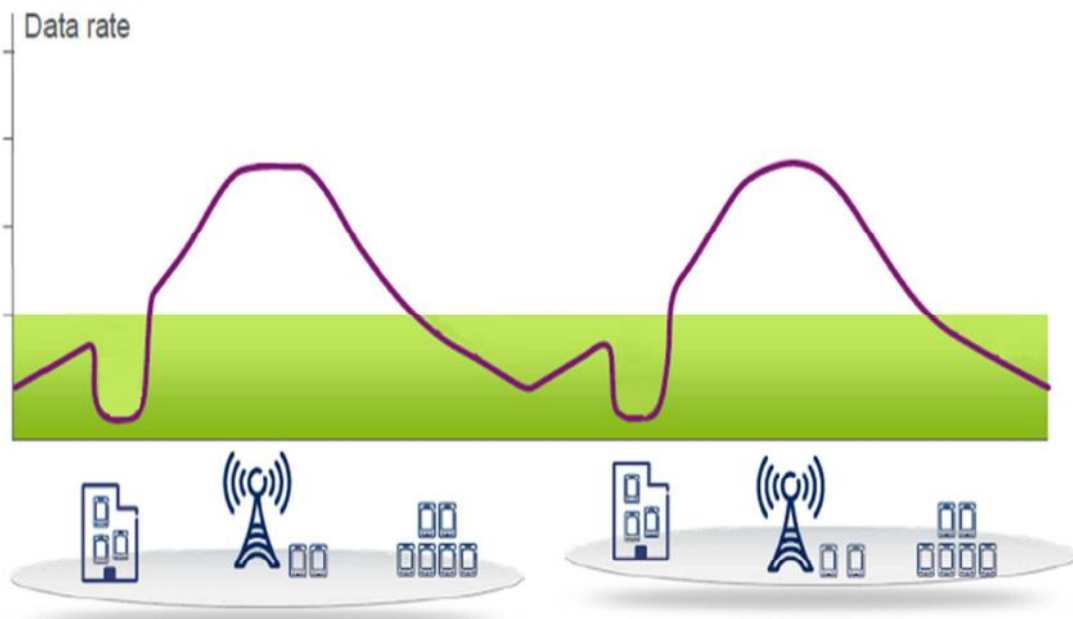
Μια ιδέα Ε.Δ. είναι να συνεργάζονται 'έξυπνα' στρώματα από Pico κυψέλες ή Femto κυψέλες πάνω στα πλαίσια του στρώματος Macro Cell. Δηλαδή, μιλάμε για υποδιαίρεση μιας μεγάλης κυψέλης σε υποκυψέλες. Δεν θα υπάρχει μόνο ένας σταθμό βάσης στην κυψέλη, αλλά πολλοί μικροί σταθμοί βάσης τύπου Pico ή Femto Cells, τα οποία προστίθενται στην υπάρχουσα μεγάλη κυψέλη. Όλοι αυτοί οι μικροί σταθμοί θα είναι συνδεδεμένα με τον μεγάλο σταθμό βάσης τύπου Macro cell χρησιμοποιώντας την κατάλληλη συνδεσμολογία.

Στην παρακάτω εικόνα υπάρχει μια απλοποίηση κυψελοειδούς συστήματος με έναν Macro σταθμό βάσης. Στην ομάδα 1 (G1) η συνδεσιμότητα του κινητού με τον σταθμό βάσης και η μεταφορά δεδομένων είναι πολύ καλή διότι υπάρχει επαρκής κάλυψη χωρίς εμπόδια και δεν υπάρχει φόρτος κίνησης στο δίκτυο. Στην ομάδα 2 (G2) δεν συμβαίνει αυτό λόγω του φόρτου κίνησης που υπάρχει στο δίκτυο σε μικρή περιοχή με αποτέλεσμα να μειώνεται ο ρυθμός μετάδοσης και η συνολική αποδοτικότητα του δικτύου για όλους τους χρήστες. Στην ομάδα 3 (G3) τα προβλήματα θα αυξηθούν ταχέως και η σύνδεση υποβαθμίζεται, λόγω της παρεμβολής από άλλες κυψέλες, αδύναμο σήμα και η κακή κάλυψη



**Fig.** Cell's Scenario

Εικόνα μεταβολής ρυθμού μετάδοσης από σταθμούς βάσης σε αστική περιοχή:



**Fig.** End user connection experience.

## **2.2 Οι κυρίες δυσκολίες και τα προβλήματα που παρουσιάζονται:**

- Αδυναμία κάλυψης στα άκρα της εμβέλειας του σταθμού βάσης και παρεμβολές συχνοτήτων από γειτονικές κυψέλες.
- Περιορισμένη χωρητικότητα για κάθε κυψέλη.
- Ο ρυθμός μετάδοσης μειώνεται ραγδαία στις αστικές περιοχές και στα κτήρια.
- Παρεμβολές λόγω φυσικών εμποδίων.

### 2.3 Ο ρόλος του ετερογενούς δικτύου είναι να επιτυγχάνει:

- Αύξηση στην συνολική απόδοση της κυψέλης.
- Αύξηση του ρυθμού μετάδοσης στα άκρα της κυψέλης.
- Αύξηση του ρυθμού μετάδοσης στα εσωτερικά κτήρια που βρίσκονται στην κυψέλη.

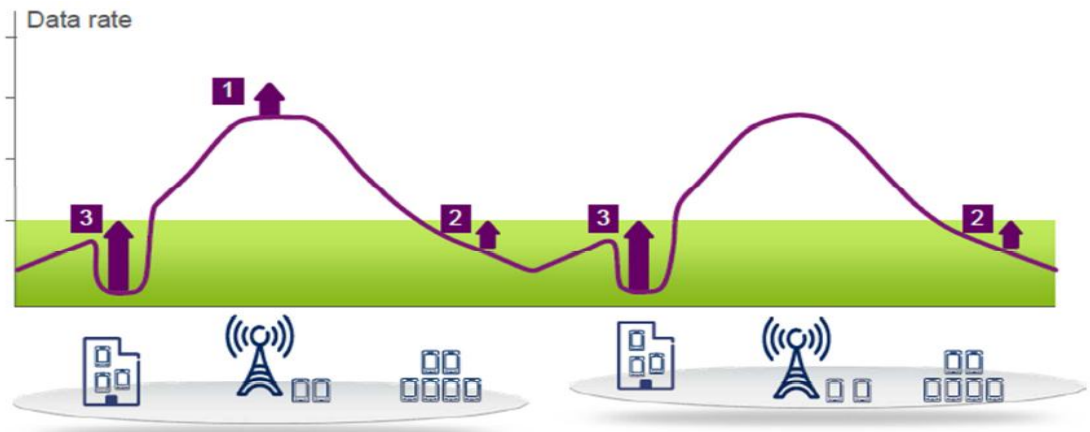


Fig. The mission of HetNets.

Για τον λόγο ότι θα εισαχθούν πολλοί μικροί σταθμοί βάσης, άρα και νέες κυψέλες, προφανώς θα πρέπει να ληφθούν υπ' όψιν παράγοντες όπως handoff, επαναχρησιμοποίηση συχνοτήτων και backhauling.

Εφαρμόζοντας την ετερογενή τεχνολογία με καινούργιες κυψέλες πάνω στο στρώμα του Macro cell επιπέδου, η συνολική απόδοση γίνεται καλύτερη όπως φαίνεται από την εικόνα. Το σύστημα έχει εισαγάγει νέες σούπερ Macro στάθμες βάσης που αυξάνουν τη συνολική απόδοση στη μέση της κυψέλης και στα άκρα της, ενώ προσθέτει νέες μικρές κυψέλες μέσα στο κτίριο και στους χώρους όπου απαιτούνται υψηλή κίνηση δεδομένων ή στις αστικές περιοχές για να λύσει τις απαιτήσεις της κυκλοφορίας τα ζητήματα χωρητικότητας.

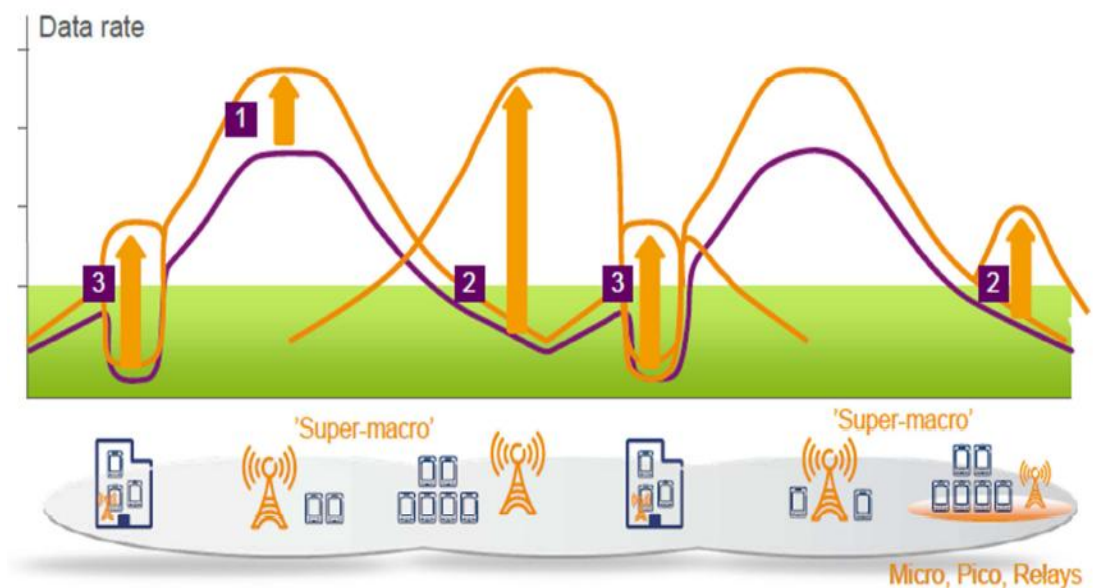


Fig. The improvements of HetNets.

### **3. Προσομοίωση (Simulation)**

Στον χώρο της έρευνας των ηλεκτρονικών υπολογιστών και των δικτύων επικοινωνιών , η προσομοίωση είναι μια χρήσιμη τεχνική δεδομένου του ότι η συμπεριφορά ενός δικτύου μπορεί να μοντελοποιηθεί υπολογίζοντας την αλληλεπίδραση μεταξύ των διαφόρων συνιστωσών του δικτύου (για παράδειγμα, δικτυακές οντότητες , όπως δρομολογητές , φυσικές συνδέσεις ή πακέτα ) με τη χρήση μαθηματικών τύπων . Αφού πάρουμε τα δεδομένα των παρατηρήσεων από τα πειράματα προσομοίωσης , μπορούμε στη συνέχεια να αναλύσουμε την συμπεριφορά του δικτύου και των υποστηριζόμενων πρωτόκολλων σε μια σειρά από offline πειράματα. Όλα τα είδη των περιβαλλοντικών χαρακτηριστικών μπορούν επίσης να τροποποιηθούν με έναν ελεγχόμενο τρόπο, ώστε να μπορούμε να εκτιμήσουμε το πώς μπορεί το δίκτυο να συμπεριφερθεί κάτω από διαφορετικούς συνδυασμούς παραμέτρων ή διαφορετικές συνθήκες διαμόρφωσης. Ένα άλλο χαρακτηριστικό της προσομοίωσης δικτύου που αξίζει να σημειωθεί είναι ότι το πρόγραμμα προσομοίωσης μπορεί να χρησιμοποιηθεί μαζί με διάφορες εφαρμογές και υπηρεσίες, προκειμένου να παρατηρηθούν end-to-end ή άλλες point-to-point επιδόσεις σε δίκτυα.

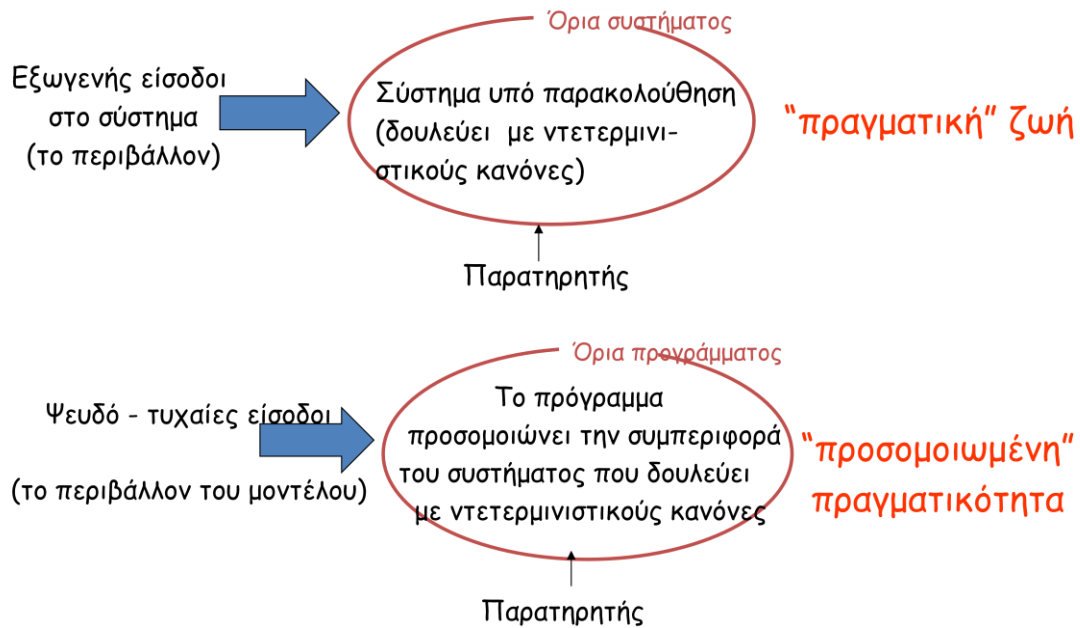
#### **3.1 Αναγκαιότητα για προσομοίωση Δικτύων:**

Η ανάπτυξη δικτύων είναι ακριβή και η επιλογή των κατάλληλων υλικών για την υλοποίηση δικτύων είναι αρκετά δύσκολη. Αυτά τα προβλήματα καθιστούν την προσομοίωση πολύ σημαντική κατά τη διάρκεια της έρευνας και της ανάπτυξης. Μια από τις μεγαλύτερες προκλήσεις για τον προσομοιωτή δικτύου είναι το ποσό της επεξεργαστικής ισχύς που απαιτείται για να πετύχει η προσομοίωση τις επιδόσεις του υλοποιημένου δικτύου.

#### **3.2 Προσομοίωση Δικτύου:**

Ένας τυπικός προσομοιωτής δικτύου επιτρέπει σε ένα χρήστη να αντιπροσωπεύει μια τοπολογία δικτύου καθορίζοντας τους κόμβους, τις συνδέσεις μεταξύ των κόμβων, το “traffic stream” μεταξύ των κόμβων στο δίκτυο. Υπάρχουν ανοιχτού κώδικα προσομοιωτές που οι ενδιαφερόμενοι έχουν ελεύθερη πρόσβαση σε αυτούς και υπάρχουν και προσομοιωτές που η χρήση τους γίνεται κατόπιν πληρωμής. Επίσης υπάρχουν πολλοί προσομοιωτές δικτύου, από πολύ απλοί μέχρι πολύ περίπλοκοι.





### 3.3 Πλεονεκτήματα προσομοίωσης Δικτύου:

- Μερικές φορές φθηνότερη.
- Βρίσκουμε τα bugs (στην σχεδίαση) εκ των προτέρων.
- Μπορεί να προσομοιώνει λεπτομέρειες του συστήματος .

### 3.4 Μειονεκτήματα προσομοίωσης Δικτύου:

- Το μοντέλο μπορεί να μην ανταποκρίνεται στην πραγματικότητα.
- Στα μεγάλα συστήματα χρειάζονται πολλοί υπολογιστικοί πόροι.
- Μπορεί να είναι αργή διαδικασία(ακριβή υπολογιστικά – 1 min πραγματικού χρόνου θα μπορούσε να διαρκεί πολλές ώρες προσομοίωσης)

## **4. Εξομοίωση Δικτύου (Emulation)**

Εξομοίωση δικτύου, σημαίνει ότι το υπό σχεδιασμό δίκτυο προσομοιώνεται, προκειμένου να αποτιμήσουμε την επίδοση του ή για να προβλέψουμε τις επιπτώσεις που μπορεί να προκύψουν μετά από πιθανές αλλαγές, ή βελτιστοποιήσεις. Η κύρια διαφορά μεταξύ τους, έγκειται στο ότι σε έναν εξομοιωτή δικτύου, τα τελικά συστήματα (end-systems) όπως οι υπολογιστές, μπορούν να συνδεθούν με τον εξομοιωτή και να λειτουργήσουν ακριβώς όπως αυτά θα συνδέονταν με ένα πραγματικό δίκτυο. Το κύριο σημείο είναι ότι στην πραγματικότητα η δουλειά ενός προσομοιωτή δικτύου είναι να μιμηθεί το δίκτυο που συνδέει τα end-systems, αλλά όχι τα end-systems αυτά καθ' αυτά. Τυπικά εργαλεία εξομοίωσης δικτύου περιλαμβάνουν τον NS2 , ο οποίος είναι ένας δημοφιλής προσομοιωτής δικτύου που

μπορεί επίσης να χρησιμοποιηθεί ως ένας (περιορισμένων δυνατοτήτων) εξομοιωτής. Σε αντίθεση, ένας τυπικός εξομοιωτής δικτύου, όπως ο WANSim, αποτελεί έναν απλό εξομοιωτή WAN (bridged) , και είναι βασισμένος σε Linux συστήματα.

## **5. Εισαγωγή στους Προσομοιωτές**

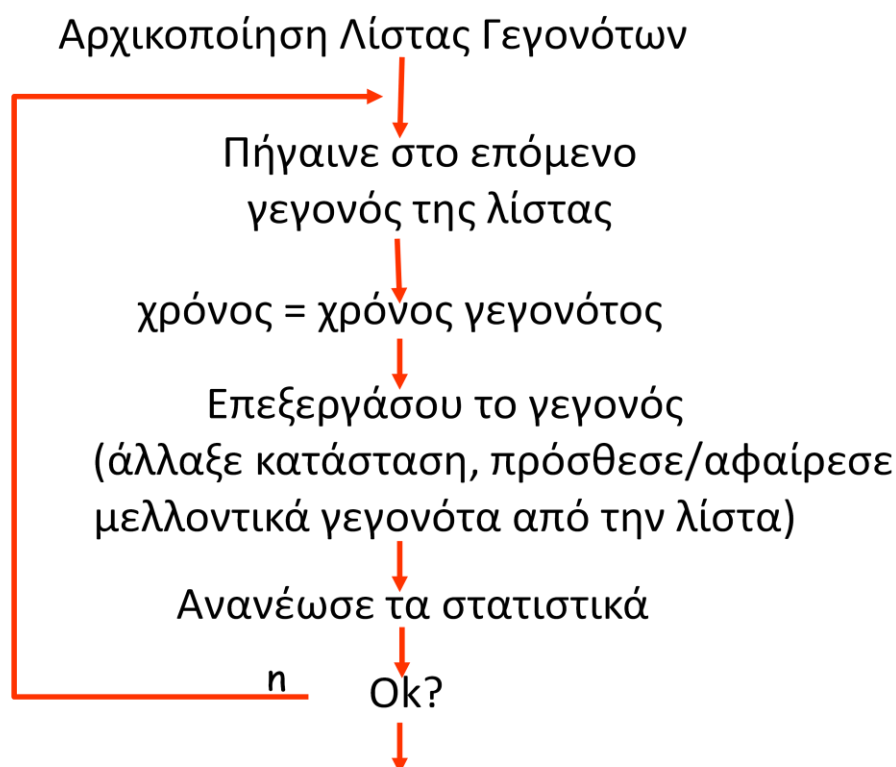
### **5.1 Εκτίμηση Επίδοσης:**

- Αριθμητικά Μοντέλα
- Προσομοίωση (Simulation)
- Εξομοίωση (Emulation)
- Πρωτότυπο

### **5.2 Προγραμματίζοντας μια Προσομοίωση:**

- **Χρόνος Προσομοίωσης:** εσωτερικά (στον προσομοιωτή) μεταβλητή που κρατεί τον χρόνο της προσομοίωσης.
- **Σύστημα “κατάσταση (state)”:** μεταβλητές που διατηρεί το πρόγραμμα προσομοίωσης που ορίζουν την κατάσταση του συστήματος. Π.χ. μπορεί να παρακολουθεί τα πακέτα στην ουρά, τρέχουσα τιμή του retransmission timer.
- **Γεγονότα (events):** σημεία στον χρόνο που το σύστημα αλλάζει κατάσταση. Κάθε event σχετίζεται με κάποιον χρόνο (του γεγονότος αυτού). Π.χ. άφιξη ενός πακέτου στην ουρά, αναχώρηση από την ουρά. Ακριβώς σε αυτά τα χρονικά σημεία η προσομοίωση πρέπει να λάβει δράση (αλλαγή κατάστασης και ενδεχομένως να προκαλέσει καινούρια γεγονότα).

Διάγραμμα Ροής Προσομοιωτή:



## 6. Ανάλυση των κύριων εργαλείων προσομοίωσης

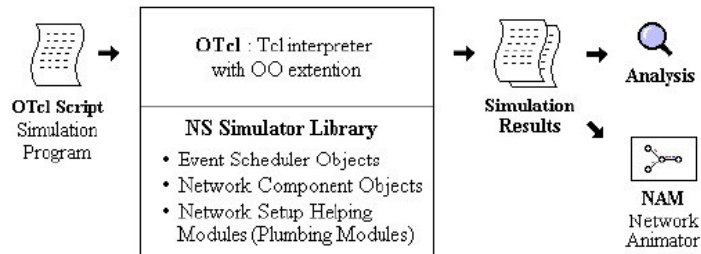
### 6.1 NS-2 (Network Simulator version2):

Το NS-2 είναι ένα εργαλείο προσομοίωσης διακριτών συμβάντων για την μοντελοποίηση του δικτύου, το οποίο χρησιμοποιείται συνήθως για εκπαιδευτικούς και ερευνητικούς σκοπούς. Υποστηρίζει προσομοίωση με IP και στοχεύει στη δικτύωση της έρευνας (networking research). Παρέχει υποστήριξη για την προσομοίωση του πρωτοκόλλου TCP, τη δρομολόγηση και multicast πρωτόκολλα πάνω σε όλα τα δίκτυα (ενσύρματα και ασύρματα). Αυτός ο προσομοιωτής είναι υλοποιημένος σε C++ και Object oriented Tcl (OTcl). Είναι ένα πακέτο λογισμικού ανοιχτού κώδικα διαθέσιμο για Windows 32 και Linux πλατφόρμες το οποίο πρωτοκυκλοφόρησε το 1996. Το NS -2 είναι η εξέλιξη του προσομοιωτή tcpsim που ήταν μια έκδοση του REAL προσομοιωτή που βασίζεται στο λογισμικό προσομοίωσης NEST(με το οποίο αναπτύσσονται και δοκιμάζονται καταναεμημένα συστήματα και πρωτόκολλα) . Παρέχει λειτουργίες εξομοίωσης NS-2 και μπορεί να χρησιμοποιείται για παράλληλη και καταναεμημένη προσομοίωση: PDNs

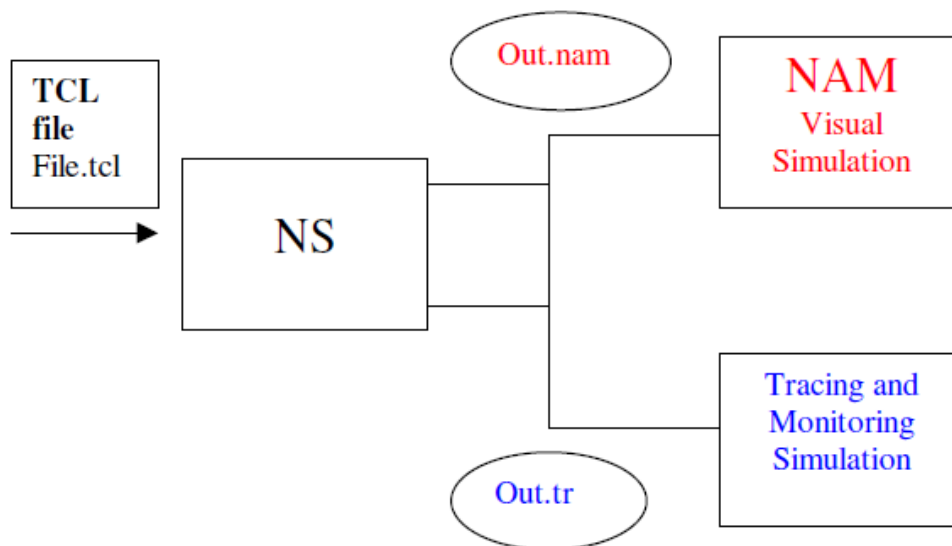
Το NS-2 χρησιμοποιείται για την αξιολόγηση της απόδοσης των υφιστάμενων πρωτόκολλων δικτύου, για την αξιολόγηση νέων πρωτόκολλων δικτύου πριν από τη

χρήση τους, για την εκτέλεση μεγάλης κλίμακας πειραμάτων που δεν είναι δυνατή σε πραγματικά πειράματα, καθώς επίσης και για την προσομοίωση μιας ποικιλίας δικτύων IP.

Η αρχιτεκτονική του NS-2 είναι:



Όπως φαίνεται στο παραπάνω σχήμα, ένα αντικειμενοστραφές Tcl script (OTel) περνάει από τον διερμηνέα και προστίθενται σε αυτό έτοιμες βιβλιοθήκες του προσομοιωτή. Μετά την διαδικασία αυτή προκύπτουν τα αποτελέσματα της προσομοίωσης σε αναλυτική μορφή, σε NAM (Network Animator) μορφή ή ακόμη και σε γραφήματα με τη βοήθεια των XGRAPH και TraceGraph εργαλείων που διαθέτει το NS-2.



**C ++ :** Είναι γρήγορη για να τρέξει μια υλοποίηση, αλλά πιο αργή στις αλλαγές και γι αυτό είναι κατάλληλη για τη λεπτομερή εφαρμογή του πρωτοκόλλου.

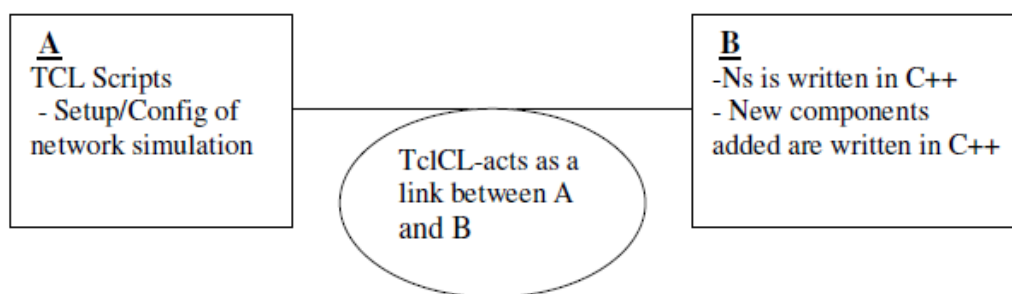
**Otel (Object Oriented Tool Command Language):** Τρέχει πολύ πιο αργά από τη C ++ , αλλά είναι πολύ γρήγορη (και διαδραστική) στις αλλαγές, γι αυτό είναι ιδανική

για τη διαμόρφωση προσομοιώσεων. Το NS παρέχει τη δυνατότητα τα αντικείμενα και οι μεταβλητές να εμφανίζονται και στις δύο γλώσσες (OTcl και C++).

Προγραμματισμός σε OTcl:

- Αρχικοποιούμε ένα γεγονός.
- Ρυθμίζουμε την τοπολογία του δικτύου χρησιμοποιώντας τα αντικείμενα του δικτύου που καλούμαστε να προσομοιώσουμε.
- Ορίζει πότε πρέπει να αρχίσει ή να σταματήσει η μετάδοση πακέτων από την πηγή.

Είναι, επίσης δυνατόν να προστεθούν συναρτήσεις και μεταβλητές στη C++ που συνδέεται με TclCL σύνδεσμο με τον TCL κώδικα.



Ένα άλλο σημαντικό συστατικό της NS, εκτός από τα αντικείμενα του δικτύου είναι το χρονοδιάγραμμα συμβάντων ή γεγονότων. Ένα συμβάν ή γεγονός στον NS είναι ένα αναγνωριστικό πακέτου που είναι μοναδικό για κάθε πακέτο και διαθέτει προγραμματισμένη ώρα και έναν δείκτη σε ένα αντικείμενο που χειρίζεται το συμβάν. Το χρονοδιάγραμμα εκδήλωσης στο NS-2 οργανώνει το χρονόμετρο προσομοίωσης, πετάει γεγονότα από την ουρά συμβάντων, ενεργοποιεί στοιχεία του δικτύου στην προσομοίωση.

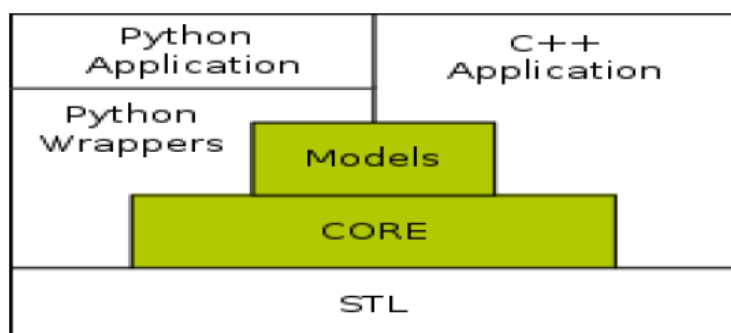
## **6.2 NS-3 (Network Simulator version3):**

Το NS-3 είναι ένας προσομοιωτής διακριτών συμβάντων επόμενης γενιάς που κυκλοφόρησε πρόσφατα και προορίζεται για αντικαταστάτης του NS-2. Διαθέτει πολλά χαρακτηριστικά, πολλά από τα οποία βρίσκονται ακόμα σε εξέλιξη και αναμένεται να γίνει η πρώτη επιλογή στις επιστημονικές κοινότητες σύντομα. Το NS-3 στοχεύει κατά κύριο λόγο στην έρευνα και στην εκπαιδευτική χρήση του και είναι υπό την άδεια GNU GPLv2 άδεια (ανοιχτού κώδικα), είναι διαθέσιμο για την έρευνα και την ανάπτυξη. Βασίζεται στις συνεχιζόμενες υλοποιήσεις της κοινότητας για την ανάπτυξη νέων μοντέλων, την επέμβαση ή τη διατήρηση των υφιστάμενων συστημάτων, και το μερίδιο αποτελεσμάτων. Είναι ένας προσομοιωτής ο οποίος βασίζεται σε Unix / Linux συστήματα καθώς και σε Windows (Cygwin)

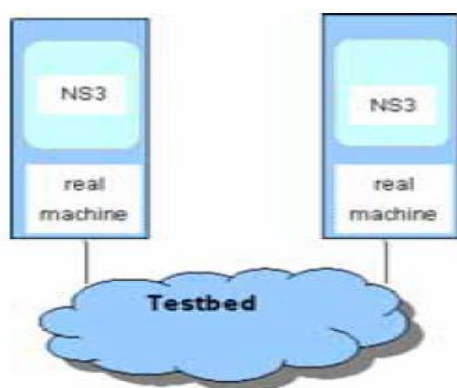
Είναι γραμμένος σε C ++, με Python scripting διεπαφές για τους χρήστες. Εστιάζει σε IPv4 και IPv6 δίκτυα, αλλά υποστηρίζει και άλλες αρχιτεκτονικές μη-IP, όπως αισθητήρες ή DTNs. Οι χρήστες του NS-3 θα είναι σε θέση να χρησιμοποιούν παραδείγματα έτοιμων σεναρίων που παρέχονται, αλλά αναμένεται ότι οι περισσότεροι χρήστες (στο πλαίσιο της έρευνας) θα χρειαστεί είτε να γράψουν νέα σενάρια ή να τροποποιήσουν ήδη υπάρχοντα ή ακόμα και τον τρόπο μοντελοποίησης του προσομοιωτή.

Η εφαρμογή της προσομοίωσης και των βασικών μοντέλων βασίζεται σε C++. Το Ns-3 είναι δομημένο ως βιβλιοθήκη η οποία μπορεί να είναι στατικά ή δυναμικά συνδεδεμένη με ένα κύριο πρόγραμμα σε C ++. Οι βιβλιοθήκες αυτές καθορίζουν την έναρξη της προσομοίωσης και την τοπολογία προσομοίωσης. Επίσης γίνεται χρήση προγραμμάτων σε Python.

Η αρχιτεκτονική του NS-3 είναι:



Οι οντότητες των πρωτοκόλλων είναι σχεδιασμένες για να είναι πιο κοντά στους πραγματικούς υπολογιστές. Χρησιμοποιεί ελαφριές εικονικές μηχανές για την εικονικοποίηση της πλατφόρμας δοκιμών και ολοκλήρωσης προσομοιώσεων.

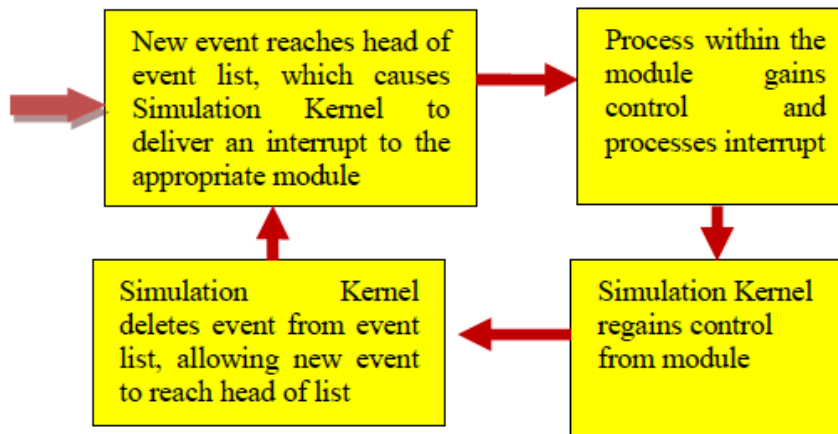


### **6.3 OPNET Modeler (Optimized Network Engineering Tools):**

Ο προσομοιωτής OPNET είναι το πρώτο εμπορικά διαθέσιμο εργαλείο προσομοιωτή δικτύου. Αναπτύχθηκε για πρώτη φορά στο MIT. Στο διαδίκτυο υπάρχουν αρκετά web-based demo. Αυτός ο προσομοιωτής παρέχει ένα καλό γραφικό περιβάλλον που καλύπτει τις ανάγκες ολοκλήρωσης της μοντελοποίησης ενός δικτύου. Είναι ένα αρκετά μεγάλο και ισχυρό λογισμικό προσομοίωσης με μεγάλη

ποικιλία δυνατοτήτων για την προσομοίωση για ολόκληρα τα ετερογενή δίκτυα που προσομοιώνει με διάφορα πρωτόκολλα. Τόσο η συμπεριφορά και όσο και οι επιδόσεις των συστημάτων μπορούν να αναλυθούν εκτελώντας προσομοιώσεις διακριτών συμβάντων.

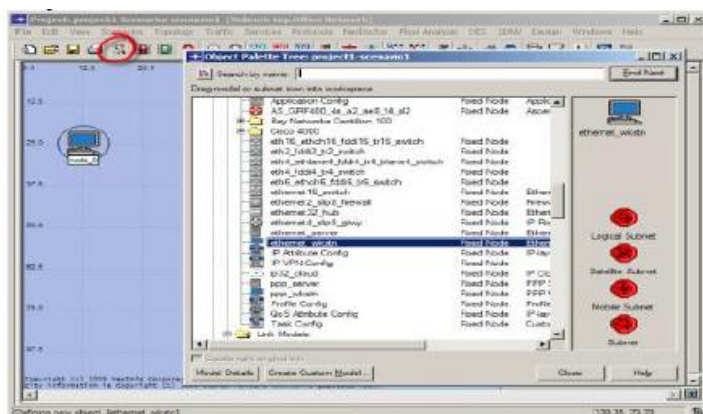
Η αρχιτεκτονική του OPNET είναι:

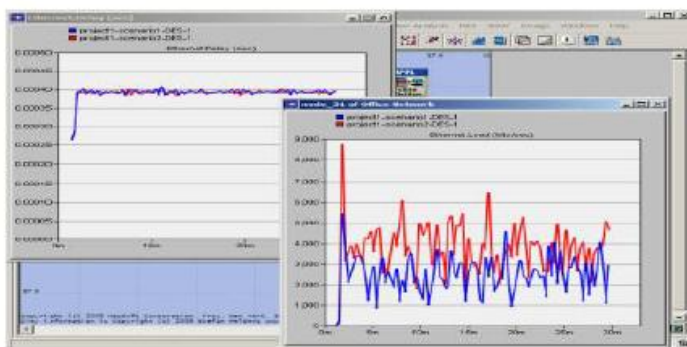
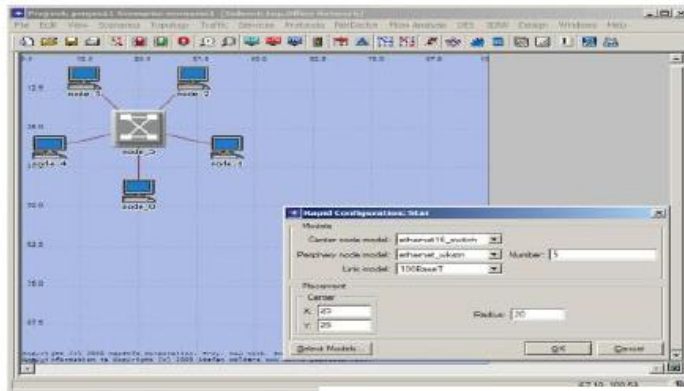


Η κύρια γλώσσα προγραμματισμού OPNET είναι η C (πρόσφατες κυκλοφορίες υποστηρίζουν και C++). Η αρχική διαμόρφωση (τοπολογία, ρύθμιση παραμέτρων) επιτυγχάνεται συνήθως με τη χρήση Graphical User Interface (GUI), ένα σύνολο από XML αρχεία ή μέσω βιβλιοθηκών σε C.

Στα Σενάρια προσομοίωσης καταστάσεων κρίσης (π.χ., αλλαγή των παραμέτρων μετά από κάποιο χρόνο, την ενημέρωση τοπολογίας, κλπ.) συνήθως απαιτείται γραπτώς C ή C++ κώδικας, αν και σε απλούστερες περιπτώσεις μπορεί κανείς να χρησιμοποιήσει ειδικό «σενάριο» παραμέτρους (π.χ., η σύνδεση αποτυγχάνει / επαναφορά του χρόνου).

Το OPNET παρέχει γραφική διεπαφή για τη συγγραφή του κώδικα:



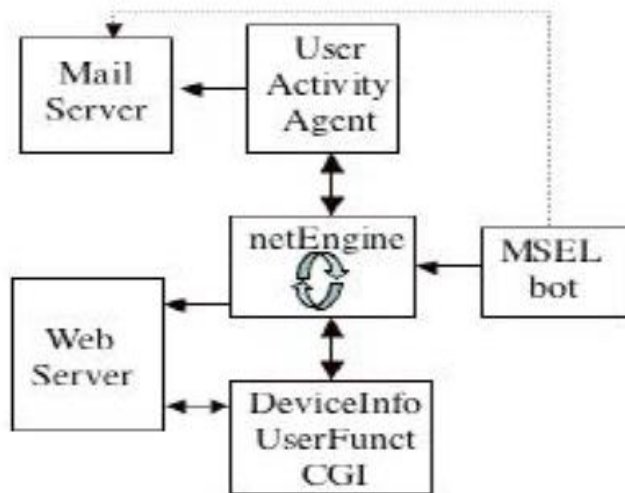


## 6.4 NETSIM (Network Based Environment for Modeling and Simulation)

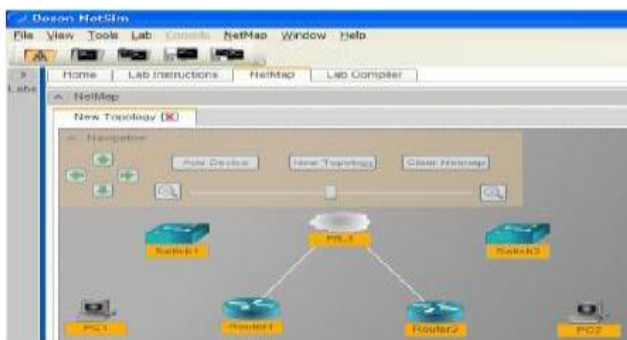
Πρόκειται για μια εφαρμογή που προσομοιώνει Cisco συστήματα δικτύωσης υλικού και λογισμικού. Έχει σχεδιαστεί για να βοηθήσει το χρήστη στην εκμάθηση της δομής εντολών Cisco IOS. Έχει αντικειμενοστραφή σύστημα για να υποστηρίξει περιβάλλον προσομοίωσης και ανάλυσης επικοινωνίας φωνής και δεδομένων για σενάρια Παγκόσμιων Συστημάτων Επικοινωνίας (High Frequency Global Communication Systems – HFGCS).



Η αρχιτεκτονική του NETSIM:



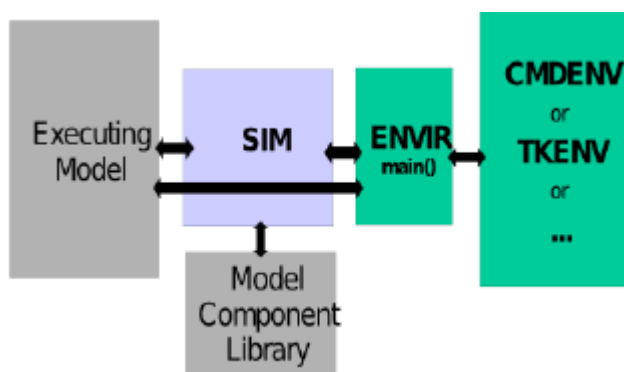
Προσφέρει δυνατότητα δημιουργίας γρήγορης και ανεξάρτητης πλατφόρμας λογισμικού σε Java που μπορεί εύκολα να χρησιμοποιηθεί για την δημιουργία WWW-based προγραμμάτων. Χρησιμοποιώντας Java μπορεί κανείς να δημιουργήσει μικρά προγράμματα που ονομάζονται μικροεφαρμογές- applets που είναι ενσωματωμένα σε ένα κώδικα HTML και έτσι η εφαρμογή είναι ορατή σε οποιονδήποτε browser συμβατό με java .



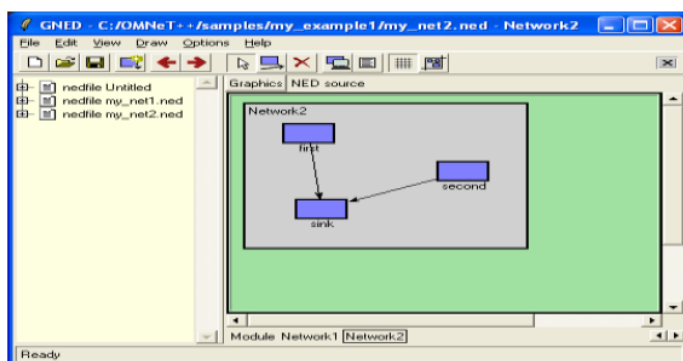
## 6.5 OMNeT++ (Optical Micro-Networks Plus Plus)

Το OMNeT++ είναι ένα επεκτάσιμο εργαλείο το οποίο βασίζεται σε βιβλιοθήκες προσομοίωσης διακριτών συμβάντων σε C++ και σε ένα επαναχρησιμοποιήσιμο σύνολο βιβλιοθηκών του. Υποστηρίζει προσομοίωση ενσύρματων και ασύρματων δικτύων επικοινωνίας, on-chip δίκτυα, και δίκτυα ουρών. Ακόμα, υποστηρίζει εξειδικευμένες λειτουργικότητες, όπως υποστήριξη για δίκτυα αισθητήρων, ασύρματων ad-hoc δικτύων, πρωτόκολλα διαδικτύου, μοντέλα επιδόσεων, φωτονικά δίκτυα, κλπ.

Η αρχιτεκτονική του OMNeT++ :



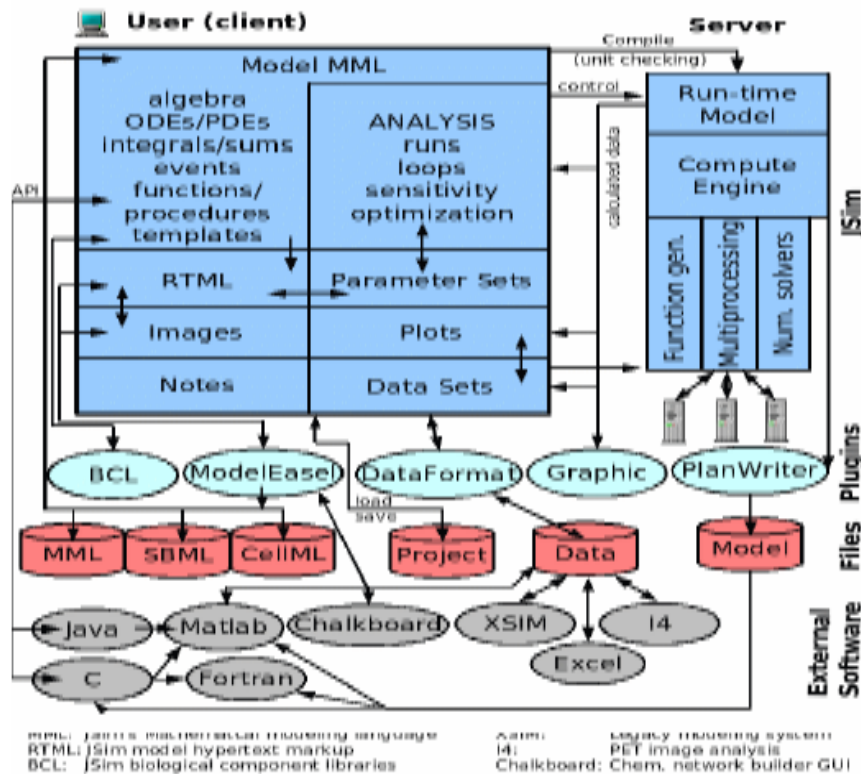
Αποτελείται από βιβλιοθήκες κλάσεων σε C++ οι οποίες αποτελούν τον πυρήνα των προσομοιώσεων και είναι πολύ χρήσιμες( παραγωγή τυχαίων αριθμών, συλλογές στατιστικών, ανακάλυψη τοπολογιών, κ.α). Αυτές χρειάζονται στην δημιουργία των συστατικών της προσομοίωσης(κανάλια, κ.α). Ο OMNeT++ παρέχει την κατάλληλη υποδομή για την συγκέντρωση της προσομοίωσης και των εργαλείων της(γλώσσα NED, ini αρχεία). Διαθέτει επίσης περιβάλλοντα για προσομοιώσεις (Tkenv, Cmdenv). Ένα από τα περιβάλλοντα αυτά βασίζεται στο Eclipse για να σχεδιάζει τις προσομοιώσεις, να τις τρέχει και να τις αξιολογεί. Επίσης διαθέτει διεπαφές για υλοποίηση προσομοιώσεων σε πραγματικό χρόνο (MRIP), για παράλληλα κατανομημένη προσομοίωση, για σύνδεσης με βάσεις δεδομένων.



## 6.6 J-Sim (Java-based simulation):

Ο J-Sim είναι ένας προσομοιωτής δικτύων, ο οποίος διαθέτει μηχανήμα προσομοίωσης σε java. Διαθέτει συστήματα σε java για τη δημιουργία ποσοτικών και αριθμητικών μοντέλων και για την ανάλυσή των μοντέλων αυτών σε σχέση με πειραματικά δεδομένα.

Η αρχιτεκτονική του J-Sim :

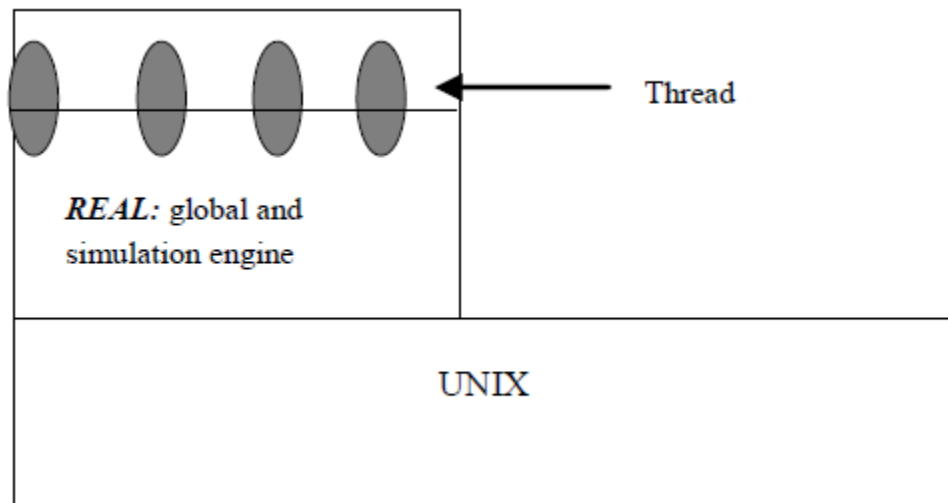


Ο JSIM παρέχει βασικές κλάσεις για τις προσομοιώσεις σε java. Αυτές οι κλάσεις μπορούν να χρησιμοποιηθούν ή να επεκταθούν άμεσα σύμφωνα με τις απαιτήσεις του χρήστη. Η Tcl γλώσσα αποτελεί βασικό τμήμα του JSIM προσομοιωτή. Χρησιμοποιείται σαν σύνδεσμος μεταξύ όλων των στοιχείων και καθορίζει τον τρόπο λειτουργίας του συστήματος. Ο JSIM παρέχει επίσης τη δυνατότητα χειρισμού Java αντικειμένων από περιβάλλον σε Tcl.

## 6.7 REAL (Realistic and Large):

Είναι ένας προσομοιωτής δικτύου αρχικά για τη μελέτη της δυναμικής συμπεριφοράς της ροής και των συστημάτων ελέγχου συμφόρησης σε δίκτυα δεδομένων με μεταγωγή πακέτων. Παρέχει στους χρήστες έναν τρόπο προσδιορισμού αυτών των δικτύων και την προσομοίωση της συμπεριφοράς τους.

Η αρχιτεκτονική του REAL:



Χρησιμοποιεί C κώδικα για τη δημιουργία δομών γραφημάτων. Επίσης διαθέτει λειτουργίες σε C οι οποίες εκτελούνται παράλληλα από το νήμα στο οποίο βασίζεται η προσομοίωση.

### **6.8 Λοιποί Προσομοιωτές:**

- SSF
- SSFNet
- DaSSF
- QualNet
- Simscript
- HEGONS
- GTNeTS
- ATM-TN
- x-Sim

**6.9 Αναφορά & Ταξινόμηση των βασικών εργαλείων προσομοίωσης με βάση τα διάφορα διακριτά τους χαρακτηριστικά:**

Name of Network Simulator	Availability(site)
NS2	Free for use <a href="http://www.isi.edu/nsnam/ns/ns-build.html">http://www.isi.edu/nsnam/ns/ns-build.html</a>
NS3	Free for use <a href="http://www.nsnam.org/ns-3-13/download/">http://www.nsnam.org/ns-3-13/download/</a>
OPNET	Commercial network simulator <a href="http://www.opnet.com/university_program/it_guru_academic_edition/">http://www.opnet.com/university_program/it_guru_academic_edition/</a>
NetSim	Commercial network simulator for use at the undergraduate level <a href="http://www.ssfnet.org/download/license.html">http://www.ssfnet.org/download/license.html</a>
OMNeT++	Free for academic and non-profit use <a href="http://www.omnetpp.org/component/docman/cat_view/17-downloads/1-omnet-releases">http://www.omnetpp.org/component/docman/cat_view/17-downloads/1-omnet-releases</a>
REAL	Free for use <a href="http://www.cs.cornell.edu/skeshav/real/overview.html">http://www.cs.cornell.edu/skeshav/real/overview.html</a>
J-Sim	Free for use <a href="https://sites.google.com/site/jsimofficial/downloads">https://sites.google.com/site/jsimofficial/downloads</a>
QualNet	Commercial simulator <a href="http://www.it.iitb.ac.in/~qualnet/">http://www.it.iitb.ac.in/~qualnet/</a>

	Network simulators name
Commercial	OPNET, QualNet
Open source	NS2, NS3, OMNeT++, SSFNet, J-Sim

## LANGUAGE USED BY SIMULATORS

Name of Network Simulator	Language
NS2	C++,Otel
NS3	C++, Python
OPNET	C (C++):
NetSim	Java
OMNeT++	C++
REAL	C
J-Sim	Java, Tcl

### 7. Προσομοίωση μεγάλης Κλίμακας

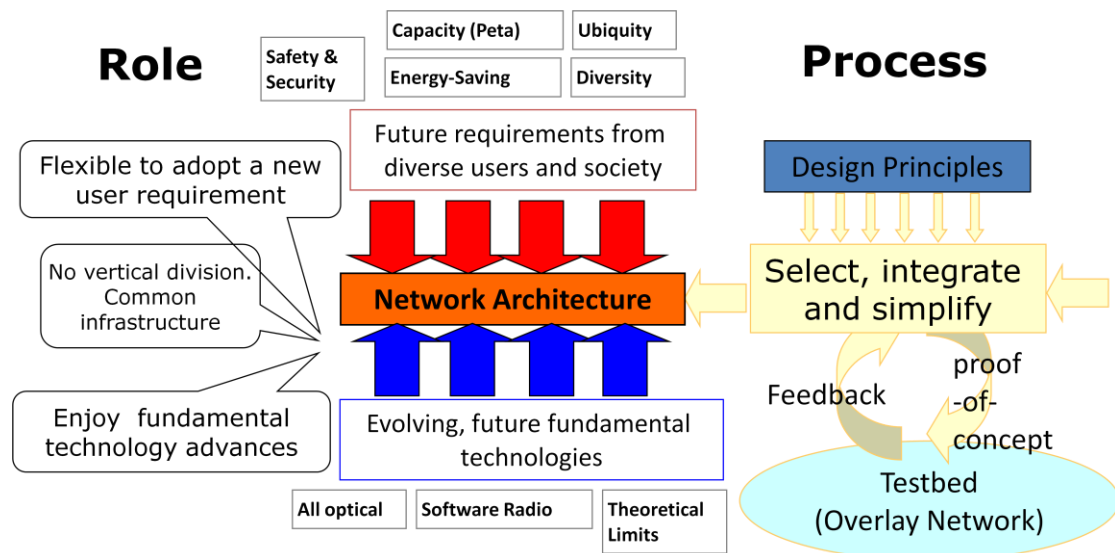
Με την προσομοίωση δικτύων μεγάλης κλίμακας γίνεται εφικτή η περιγραφή της συμπεριφοράς δικτύων μεγάλου μεγέθους . Παραδείγματα Δικτύων μεγάλης κλίμακας είναι τα Εταιρικά Δίκτυα (π.χ. Τράπεζες , Οργανισμοί κ.α.). Αυτά έχουν πολύ υψηλές υπολογίσιμες απαιτήσεις (απαιτούνται πολλοί υπολογιστικοί πόροι για την προσομοίωση τέτοιων μεγάλων δικτύων) και επομένως η διαδικασία της προσομοίωσης τους είναι πιο αργή σε σχέση με την προσομοίωση μικρότερης κλίμακας δικτύων.

#### 7.1 NS2 to Web

Στον προσομοιωτή NS2 στο Web η προσομοίωση (Web Based Network Simulation) παρέχεται ως υπηρεσία (as a service). Δεν είναι απαραίτητη προϋπόθεση να έχουμε ισχυρό υπολογιστή. Δουλεύουν σε όλες τις πλατφόρμες (Windows, Linux, Mac OS κτλ), άρα μειώνονται οι ασυμβατότητες. Δυνατότητα για προσομοίωση πολύ μεγάλου όγκου δεδομένων / πολύ μεγάλης έκτασης δίκτυα (WebNMS).

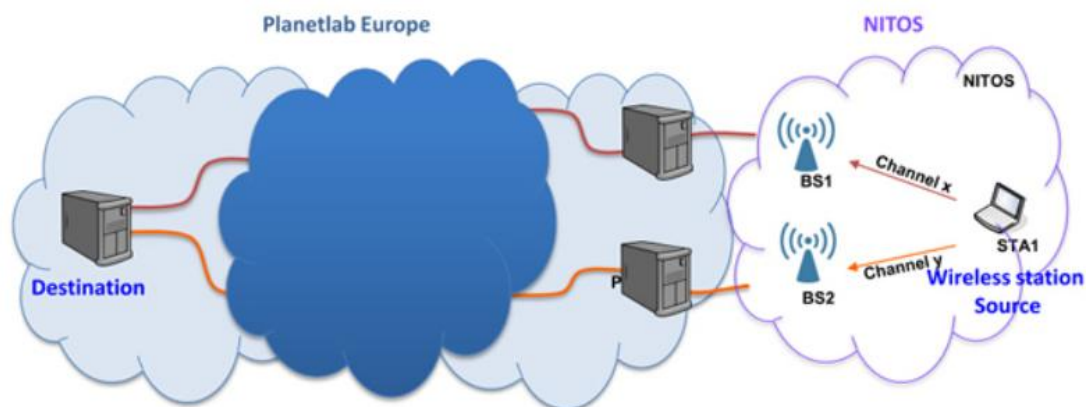
## 7.2 AKARI

Είναι μια Πλατφόρμα Προσομοίωσης Δικτύων και Υπηρεσιών η οποία αποτελεί Ερευνητικό Project για Δίκτυα επόμενης γενιάς και δίνει μια Καινούρια προσέγγιση σε προβλήματα.



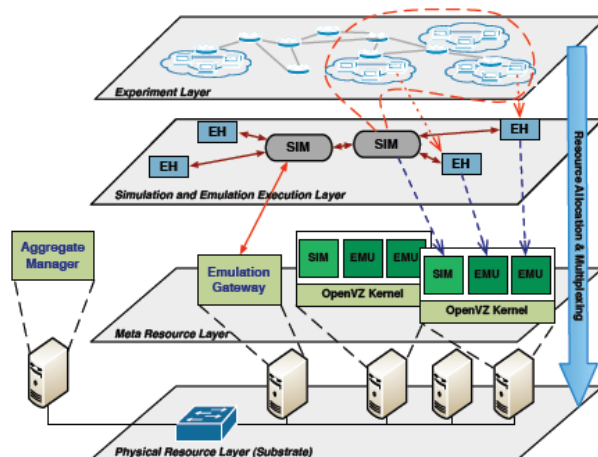
## 7.3 PlanetLab:

Είναι μια Πλατφόρμα Προσομοίωσης Δικτύων και Υπηρεσιών η οποία βασίζεται σε ένα Δίκτυο Κοινοπραξίας το οποίο αποτελείται από ένα σύνολο οργανισμών που συμμετέχουν στην κοινοπραξία (κυβερνητικοί, ακαδημαϊκοί, βιομηχανικοί). Έτσι οι ερευνητές μπορούν να πειραματιστούν σε νέες τεχνολογίες υπό πραγματικές συνθήκες και σε μεγάλη κλίμακα. Επίσης οι χρήστες μπορούν να έχουν πολλαπλές οπτικές γωνίες από τις οποίες μπορούν να παρατηρήσουν την αντίδραση και τη συμπεριφορά του δικτύου.



#### **7.4 The Global Environment for Network Innovations (GENI):**

Είναι ένα εικονικό εργαστήριο, δηλαδή μια Πλατφόρμα Προσομοίωσης Δικτύων και Υπηρεσιών για προσομοίωση δικτύων και για έρευνα μεγάλης κλίμακας. Το τμήμα του GENI το οποίο ονομάζεται PrimoGENI παρέχει ολοκληρωμένη λύση για προσομοίωση δικτύων σε πραγματικό χρόνο και για την κατασκευή, ανάπτυξη και λειτουργία δοκιμών. Επίσης παρέχει προσομοίωση σε πραγματικό χρόνο και έχει τη δυνατότητα να αλληλεπιδρά με πραγματικές εφαρμογές. Υποστηρίζει ακόμη πειράματα μεγάλης κλίμακας καθώς παρέχει εκατομμύρια προσομοιωμένες οντότητες (hosts, routers και links). Το PrimoGENI χρησιμοποιεί σε πραγματικό χρόνο μια επέκταση του προσομοιωτή SSFNet, που είναι ικανός να λειτουργεί σε παράλληλα και κατανεμημένα συστήματα με σκοπό την προσομοίωση δικτύων μεγάλης κλίμακας και μπορεί να συλλάβει λεπτομερή συναλλαγές σε επίπεδο μεταφοράς πακέτων, επομένως παρέχει ακριβή αποτελέσματα.



#### **7.5 Future Internet Research and Experimentation(FIRE)**

Παρέχει μια πλατφόρμα και τα εργαλεία για τη δοκιμή καινοτόμων ιδεών για το μέλλον του Διαδικτύου. Το FIRE προωθεί την έννοια του πειράματος με γνώμονα την έρευνα που διεξάγεται από διάφορα ερευνητικά προγράμματα, που συνδυάζουν το όραμα της ακαδημαϊκής έρευνας με πειράματα ευρείας κλίμακας. Μεταξύ άλλων, το FIRE αποσκοπεί στη δημιουργία μιας μεγάλης κλίμακας πειραμάτων στον ευρωπαϊκό χώρο που να είναι δυναμική και βιώσιμη. Ο χώρος αυτός κατασκευάζεται με τη σταδιακή σύνδεση και τη συνένωση των υφιστάμενων και επικείμενων πειραμάτων για τις μελλοντικές τεχνολογίες του Διαδικτύου.

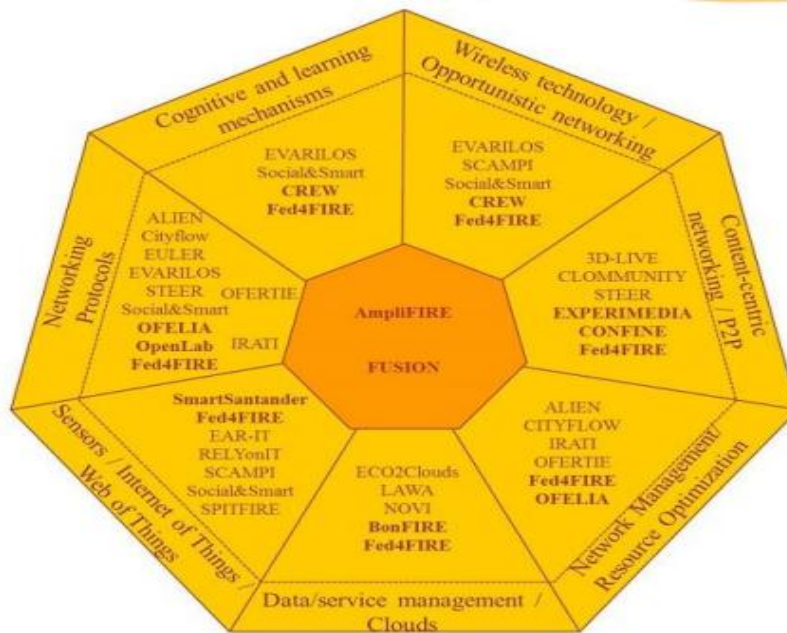


## FIRE Research



## FIRE Experimental Facility





## 8. Εισαγωγή στο περιβάλλον του NS2 με προσομοίωση απλού κώδικα

Το δίκτυο αυτό αποτελείται από 4 κόμβους (  $n_0$  ,  $n_1$  ,  $n_2$  ,  $n_3$  ) , όπως φαίνεται στο παραπάνω σχήμα . Τα duplex συνδέσεις μεταξύ  $n_0$  και  $n_2$  , και  $n_1$  και  $n_2$  έχουν 2 Mbps του εύρους ζώνης και 10 ms καθυστέρηση . Η σχέση αμφίδρομη επικοινωνία μεταξύ  $n_2$  και  $n_3$  έχει 1,7 Mbps του εύρους ζώνης και 20 ms καθυστέρηση . Κάθε κόμβος χρησιμοποιεί μια ουρά DropTail , της οποίας το μέγιστο μέγεθος είναι 10 . Το " tcp " agent συνδέεται με  $n_0$  , και η σύνδεση είναι εγκατεστημένος σε ένα tcp " sink " agent που συνδέονται με  $n_3$  . Ως προεπιλογή , το μέγιστο μέγεθος ενός πακέτου που ένα " tcp " agent μπορεί να δημιουργήσει είναι 1 Kbyte . Μια tcp " sink " agent δημιουργεί και στέλνει πακέτα ACK στον αποστολέα ( παράγοντας tcp ) και ελευθερώνει τα ληφθέντα πακέτα .

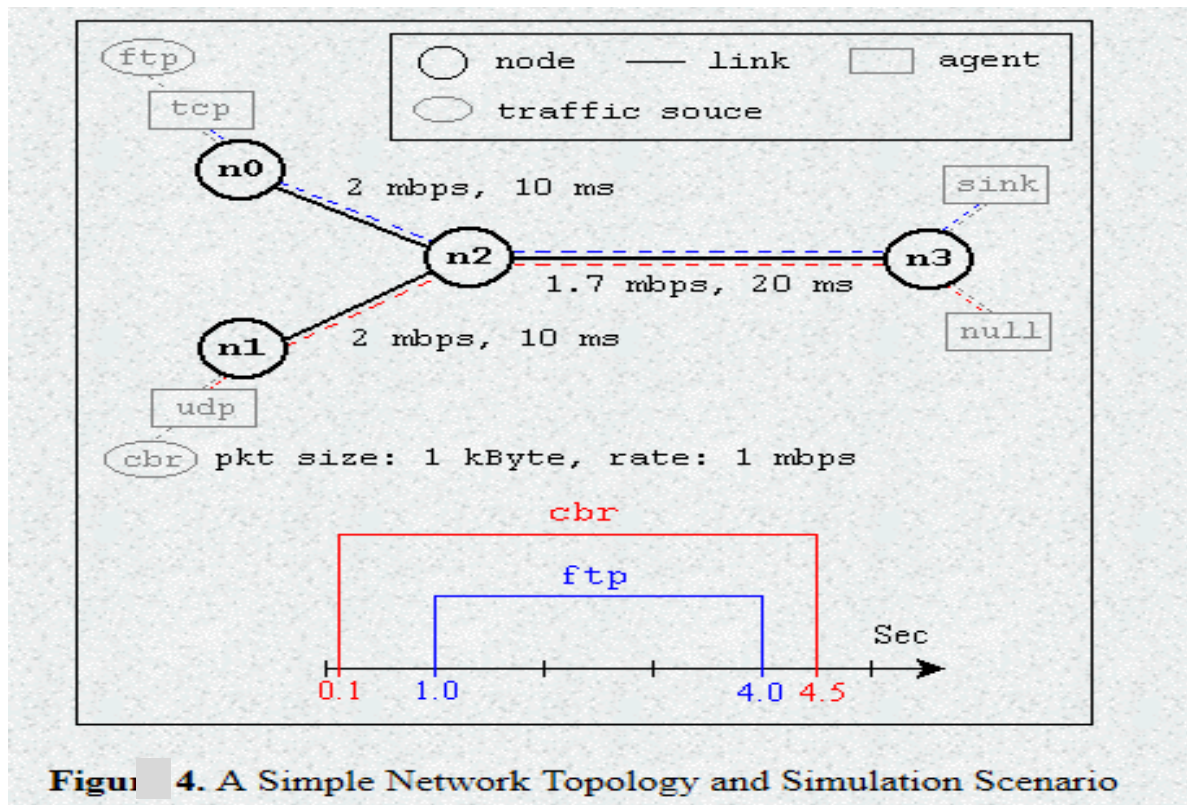
Μια 'UDP' agent που είναι συνδεδεμένη με  $n_1$  συνδέεται σε ένα " null " παράγοντα που συνδέονται με  $n_3$ . Ένα " null " agent ελευθερώνει μόνο τα πακέτα που λαμβάνονται .

Τα 'ftp' και 'cbr' γεννήτρια κυκλοφορίας του φόρτου κίνησης , επισυνάπτονται στην " tcp " και " udp " , αντίστοιχα , και το " cbr " έχει ρυθμιστεί να παράγει 1 KByte πακέτα με ρυθμό 1 Mbps .

Δηλ. το ποσό του διαστήματος που πρέπει να στέλνει κάθε φορά τα πακέτα το πρωτόκολλο 'cbr' είναι:

$$\text{Interval} = 1024 * 8 / 10^6 (1\text{Mbit}) = 0,008 \text{ ms}$$

Η 'cbr' έχει οριστεί να ξεκινήσει στο 0,1 sec και να σταματήσει στο 4,5 sec , και το 'ftp' έχει οριστεί να ξεκινήσει στο 1,0 sec και θα σταματήσει στο 4,0 sec .



**Figure 4. A Simple Network Topology and Simulation Scenario**

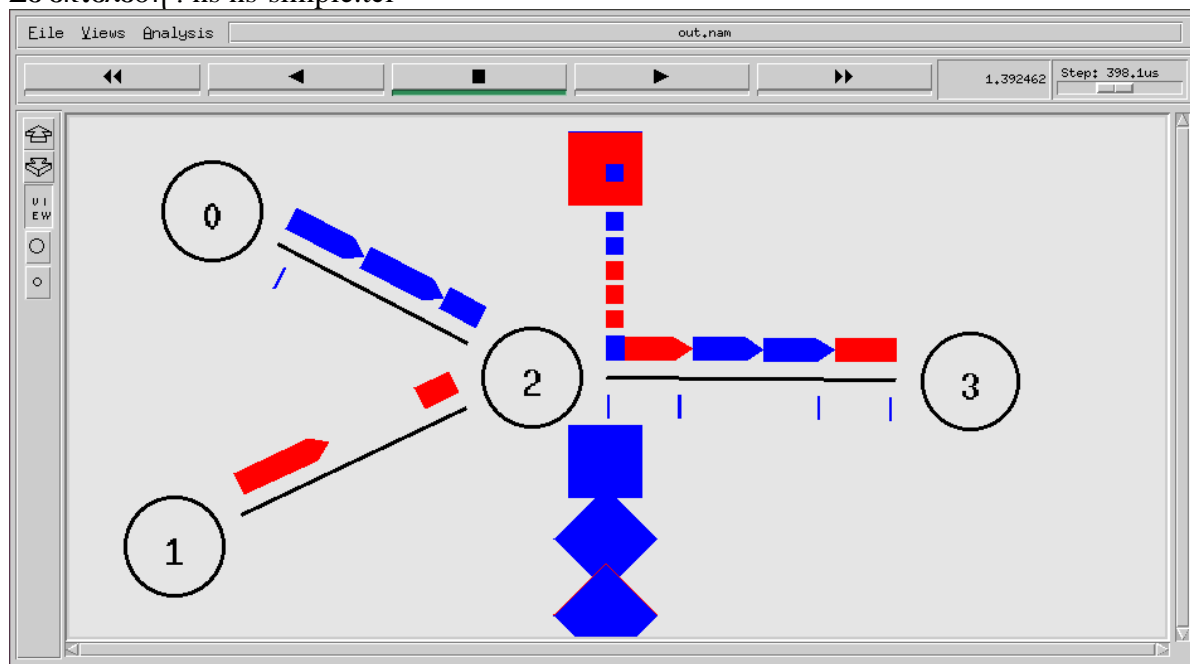
### 8.1 Κύρια μέρη στον κώδικα:

(ο Κώδικας βρίσκεται στον φάκελο "Εισαγωγή στο NS2 κώδικας")

- **set ns [new Simulator]:** δημιουργεί ένα αντικείμενο προσομοιωτή NS παράδειγμα, και την εκχωρεί στην μεταβλητή ns. Αυτή η γραμμή κάνει τα ακόλουθα:
  - Initialize the packet format
  - Create a scheduler (default is calendar scheduler)
  - Select the default address format
  - Create compound objects such as nodes and links (described later)
  - Connect network component objects created (ex. attach-agent)
  - Set network component parameters (mostly for compound objects)
  - Create connections between agents (ex. make connection between a "tcp" and "sink")
  - Specify NAM display options
- **\$ns namtrace-all file-descriptor:** η λειτουργία είναι μέλος του προσομοιωτή και καταγράφει το ίχνος της προσομοίωσης σε NAM μορφότυπο εισόδου. Δίνει, επίσης, το όνομα του αρχείου που το ίχνος θα γράψει αργότερα με την εντολή \$ ns flush-ίχνος. Παρομοίως, η συνάρτηση μέλος trace-all είναι για την καταγραφή του ίχνους προσομοίωσης σε μια γενική μορφή.
- **proc finish {}Q:** καλείται μετά στο τέλος της προσομοίωσης μετά από την εντολή \$ ns at 5.0 "finish". Σε αυτή τη λειτουργία, οι διαδικασίες προκαθορισμένες

- **set n0 [\$ns node]:** Δημιουργία κόμβου.
- **\$ns duplex-link node1 node2 bandwidth delay queue-type:** δημιουργεί δύο simplex συνδέσεις καθορισμένου εύρους ζώνης και καθυστέρησης, και συνδέει τους δύο κόμβους που ορίζεται. Στο NS, η ουρά εξόδου ενός κόμβου υλοποιείται ως μέρος ενός συνδέσμου, ως εκ τούτου, οι χρήστες θα πρέπει να καθορίσουν την επιλογή-τύπου ουρά κατά τη δημιουργία δεσμών. Στο παραπάνω σενάριο προσομοίωσης, DropTail ουρά χρησιμοποιείται.
- **\$ns queue-limit node1 node2 number:** αυτή η γραμμή καθορίζει το όριο ουράς των δύο simplex links που συνδέουν node1 και Node2 .
- **set tcp [new Agent/TCP or UDP]:** δημιουργία του παράγοντα (πρωτόκολλο) TCP ή UDP. Αποτελεί η πηγή επικοινωνίας για κίνησης πακέτων .
- **\$ns attach-agent node agent:** αποδίδει ένα αντικείμενο παράγοντα (πρωτόκολλο επιπέδου εφαρμογής) που δημιουργείται σε ένα αντικείμενο κόμβου. Το πρωτόκολλο επιπέδου εφαρμογής που της αποδίδεται είναι αυτό που δημιουργεί και παράγει την κίνηση πακέτων.
- **\$ns connect agent1 agent2:** όταν δύο παράγοντες - agents οι οποίοι θα επικοινωνούν μεταξύ τους, το επόμενο βήμα είναι να δημιουργηθεί μια λογική σύνδεση δικτύου μεταξύ τους. Η γραμμή αυτή δημιουργεί μια σύνδεση δικτύου, ρυθμίζοντας τη διεύθυνση και τον αριθμό port προορισμού στο δίκτυο του κάθε ζεύγος .

Σε εκτέλεση : ns ns-simple.tcl



Πρακτικά στο παράδειγμα αυτό, ο κόμβος 1 παράγει κίνηση πακέτων με το πρωτόκολλο cbr (κόκκινα πακέτα), τα οποία έχουν ως προορισμό τον κόμβο 3. Ο

Κόμβος 3 στέλνει μόνο μια φορά αίτηση στον κόμβο 1, ότι είναι ελεύθερος και ακούει. Τα μπλε πακέτα είναι η κίνηση των πακέτων που δημιουργούνται από το πρωτόκολλο ftp μέσω του κόμβου 0. Ο προορισμός των μπλε πακέτων είναι ο κόμβος 3, ο οποίος πρέπει να στέλνει κάθε φορά request (τα μικρά μπλε κομμάτια) αίτηση ώστε ο κόμβος 0 να ξαναστείλει επόμενα πακέτα. Ενώ στον κόμβο 2 υπάρχει καθορισμένο πλήθος ουράς . Οπότε εάν ξεπεραστεί αυτό το πλήθος στην ουρά αναμονής, τα πακέτα χάνονται όπως φαίνεται και από την εικόνα.

## **8.2 Ίχνος πακέτων της προσομοίωσης:**

Για να παράγουμε το ίχνος (trace file) των πακέτων όπου φαίνονται στην κίνηση του δικτύου προσθέτουμε το παρακάτω κώδικα:

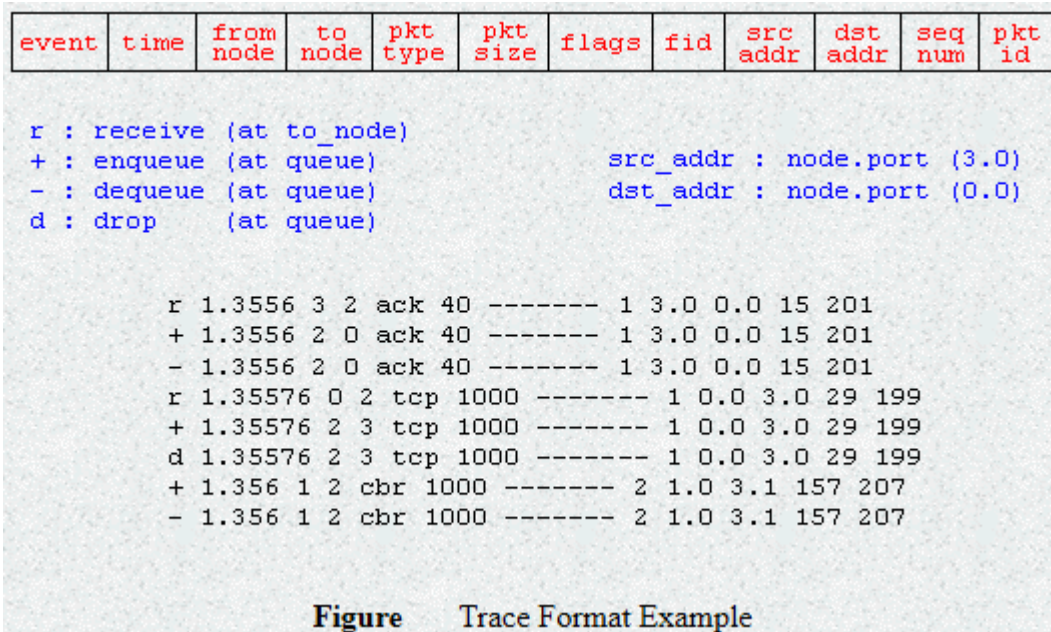
```
#Open the NAM trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Open the Trace file
set tf [open out.tr w]
$ns trace-all $tf

#Define a 'finish' procedure
proc finish {} {
    global ns nf tf
    $ns flush-trace
    #Close the NAM trace file
    close $nf
    #Close the Trace file
    close $tf
    #Execute NAM on the trace file
    exec nam out.nam &
    exit 0
}
```

Τρέχοντας το παραπάνω script δημιουργεί ένα αρχείο ίχνος NAM που πρόκειται να χρησιμοποιηθούν ως εισροή στη NAM (γραφικό περιβάλλον) και ένα αρχείο ανίχνευσης που ονομάζεται “out.tr” όπου χρησιμοποιείται για την ανάλυση προσομοίωσης.





Κάθε γραμμή ίχνους ξεκινά με ένα γεγονός (+, -, d, r) ακολουθούμενο από το χρόνο προσομοίωσης (σε δευτερόλεπτα) του γεγονότος, καθώς και από και προς τον κόμβο, τα οποία προσδιορίζουν τη σχέση κατά την οποία συνέβη το γεγονός. Η επόμενη πληροφορία στη γραμμή πριν από σημαίες (εμφανίστηκε ως "-----", δεδομένου ότι δεν έχει οριστεί η σημαία) είναι ο τύπος και το μέγεθος του πακέτου (σε Bytes). Το επόμενο πεδίο είναι η ταυτότητα της ροής (fid) του IPv6, ένας χρήστης μπορεί να ορίσει για κάθε ροή στην είσοδο OTcl του σεναρίου. Ακόμα κι αν το fid πεδίο δεν μπορεί να χρησιμοποιηθεί σε μια προσομοίωση, οι χρήστες μπορούν να χρησιμοποιήσουν αυτό τον τομέα για τους σκοπούς της ανάλυσης. Το πεδίο fid χρησιμοποιείται επίσης όταν ορίζετε το χρώμα ρεύμα για την οθόνη NAM. Τα επόμενα δύο πεδία είναι πηγή και τη διεύθυνση προορισμού σε μορφές "node.port". Το επόμενο πεδίο εμφανίζει πακέτο αύξων αριθμός του πρωτοκόλλου στρώματος δικτύου. Σημειώστε ότι ακόμη και αν UDP εφαρμογές δεν χρησιμοποιούν αύξοντα αριθμό, NS παρακολουθεί πακέτων UDP αύξοντα αριθμό για τους σκοπούς της ανάλυσης. Το τελευταίο πεδίο δείχνει το μοναδικό αναγνωριστικό του πακέτου.

## 9. Υλοποίηση προσομοίωσης - Throughput Δεδομένων στο 802.11

### 9.1 Ορισμοί Bandwidth και Throughput:

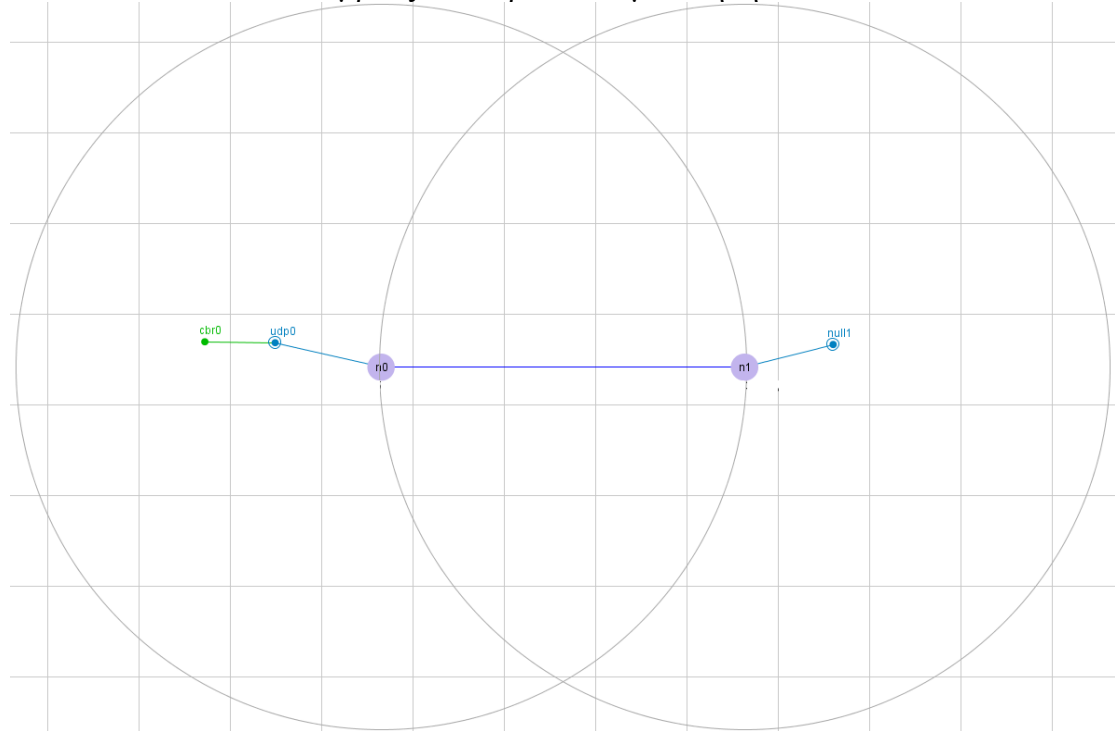
**Bandwidth :** Ο αριθμός των bits που μπορούν να μεταφερθούν πάνω από το δίκτυο σε ένα συγκεκριμένο χρονικό διάστημα. Για παράδειγμα, αν το εύρος ζώνης είναι 10Mbps περιμένουμε από το δίκτυο να μπορεί να μεταφέρει  $10 \cdot 10^6$  bits ανά δευτερόλεπτο. Σε αυτή την περίπτωση η μετάδοση ενός bit διαρκεί 0,1 msec.

**Throughput:** Συχνά χρησιμοποιείται για να αναφερθούμε στη μετρημένη (πραγματική) απόδοση ενός συστήματος. Για παράδειγμα, ενδέχεται λόγω ατελειών

στο σχεδιασμό μια ζεύξη ονομαστικής διαμετακομιστικής ικανότητας ( $\Delta I$ ) 10Mbps να έχει πραγματική  $\Delta I = 2\text{Mbps}$

## **9.2 Προσομοίωση 1ο Μέρους:**

Mac 802.11 πάνω σε 2 κόμβους όταν προσθέτουμε κίνηση



Στο 1<sup>ο</sup> μέρος, προσομοιώνεται η ασύρματη Mac/802.11 διάδοση πακέτων από τον κόμβο Source προς τον κόμβο Destination, όπου απέχουν 200m, σε διάστημα χρόνου 10 sec. Σε επίπεδο εφαρμογής, χρησιμοποιείται το πρωτόκολλο CBR (Constant Bit Rate) για την παραγωγή και αποστολή πακέτων πάνω στο πρότυπο UDP. Για την προσομοίωση, έχουμε ως σταθερά Bandwith (εύρος καναλιού) = 22Mbit (αν και στην πραγματικότητα το πρότυπο 802.11 μπορεί να υποστηρίξει bandwidth μέχρι 11Mbps). Ένα σημαντικό κομμάτι στο απλό hop ασύρματο δίκτυο που γίνεται, είναι η αλλαγή του ρυθμού μετάδοσης δεδομένων (Data rate) ώστε να παρατηρήσουμε πως μεταβάλλεται το throughput του δικτύου σε σχέση με το data rate.

### **9.2.1 Περιγραφή κώδικα στο 1<sup>ο</sup> μέρος προσομοίωσης:**

**Οι προκαθορισμένες τιμές της προσομοίωσης και των χαρακτηριστικών των κόμβων πάνω στο Mac/802.11**

```

#
=====
# Define options
#
=====
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(nn) 2 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(sc) "./scenario" ;# scenario file
set val(x) 3000.0 ;# X dimension of topography
set val(y) 400.0 ;# Y dimension of topography
set val(simtime) 10.0 ;#sim time
set val(drate) 2.0e6 ;#default datarate

```

**Παραμετροποίηση και ορισμός των χαρακτηριστικών των κόμβων όπου χρησιμοποιούν το πρότυπο Mac/802.11**

**Δηλαδή όλοι οι κόμβοι της προσομοίωσης κληρονομούν τις τιμές των προκαθορισμένων μεταβλητών.**

```
# configure node
```

```

$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) \
    -channelType $val(chan) \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace OFF

```

**Ορίζεται η μετάδοση δεδομένων (data rate) από τον κόμβο της Πηγής πάνω σε ένα υποτιθέμενο bandwidth=22 Mbit**

```

$val(mac) set dataRate_ $val(drate)
$val(mac) set bandwidth_ 22.0e6
$val(prop) set pathlossExp_ 4.0

```



**Πάνω στον κόμβο της Πηγής ιδρύεται το πρωτόκολλο επικοινωνίας UDP**

**#Attach a Udp Agent to Source**

```
set udp_(0) [new Agent/UDP]
$ns_ attach-agent $node_(0) $udp_(0)
```

**Ο κόμβος Προορισμού των πακέτων ορίζεται ως NULL διότι πρέπει να ακούει και να παραλαμβάνει τα πακέτα που στέλνει η Πηγή**

**#Attach a data-sink to destination**

```
set null_(0) [new Agent/Null]
$ns_ attach-agent $node_(1) $null_(0)
```

**Στο μέρος του παρακάτω κώδικα ιδρύεται το πρωτόκολλο CBR (Constant Bit Rate) πάνω στον κόμβο της Πηγής όπου χρησιμοποιεί το UDP protocol. Επίσης, αποτελεί το μέρος όπου στέλνεται και δημιουργείται η κίνηση (traffic) των πακέτων από τον κόμβο Source.**

**#add traffic...make src talk to dst**

```
set cbr_(0) [new Application/Traffic/CBR]
$cbr_(0) set packetSize_ 2048
$udp_(0) set packetSize_ 2048
$cbr_(0) set interval_ 0.0008
$cbr_(0) set random_ 0.96749
$cbr_(0) set maxpkts_ 1000000
$cbr_(0) attach-agent $udp_(0)
$ns_ connect $udp_(0) $null_(0)
```

### **9.2.2. Η Συνολική προσφερόμενη κίνηση:**

**Τονίζεται πως η Συνολική προσφερόμενη κίνηση των πακέτων που δημιουργείται από την Πηγή (S) και έχουν ως προορισμό τον κόμβο D , είναι:**

- **Η κίνηση πακέτων από το πρωτόκολλο CBR (στέλνονται 2KB ανά διάστημα = 0.0008 sec)**
- **Η κίνηση πακέτων 2KB από το πρωτόκολλο CBR ανά τυχαίο διάστημα που έχει εύρος περίπου 0.9 sec**
- **Η κίνηση πακέτων 2KB από το πρωτόκολλο UDP**
- **Ο ρυθμός μετάδοσης των δεδομένων (data rate bps) καθορίζεται από την παράμετρο που εισάγουμε στην εντολή γραμμών.**

### **9.2.3. Η Επιρροή του Data Rate πάνω στο Throughput:**

Για το 1<sup>ο</sup> μέρος της προσομοίωσης, θα παρατηρήσουμε τις αλλαγές που προκαλούνται στο throughput λόγω της συνολικής κίνησης που υπάρχει με διαφορετικό ρυθμό μετάδοσης δεδομένων πάνω στο δίκτυο.

Σημείωση:

Throughput είναι, το επιτυχές ποσοστό της παραλαμβανόμενης κίνησης δεδομένων σε σχέση με την συνολική κίνηση.

Δηλ. για την προσομοίωση αυτή έχουμε

$$\text{Throughput} = \frac{\text{αριθμός παραλαμβανόμενων πακετών}}{\text{αριθμός σταλθέντων πακετών}}$$

#### 9.2.4. Η Διαδικασία μέτρησης αποτελεσμάτων:

Με τις εντολές:

- ***ns single-hop.tcl -drate {data-rate}*** , τρέχουμε την προσομοίωση δίνοντας ως παράμετρο (argument) το datarate .

Μετά από κάθε προσομοίωση θα έχουμε ως έξοδο δύο αρχεία

- ***single-hop.tr*** , έξοδος αρχείου trace πακέτων προσομοίωσης.
  - ***single-hop.nam*** , έξοδος αρχείου γενικού προσομοίωσης.
- 
- ***grep "^r" {trace file name} | wc -l*** , επιστρέφει τον αριθμό παραλαμβανόμενων πακετών.
  - ***grep "^s" {trace file name} | wc -l*** , επιστρέφει τον αριθμό των πακέτων που στάλθηκαν.

Π.χ. για datarate = 1Mbps = 10<sup>6</sup> bps

```
a@a-VirtualBox:~/Desktop/Ex1$ ns single-hop.tcl -drate 1.0e6
num_nodes is set 2
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
SORTING LISTS ...DONE!
NS EXITING...
a@a-VirtualBox:~/Desktop/Ex1$ grep "^r" single-hop.tr | wc -l
15199
a@a-VirtualBox:~/Desktop/Ex1$ grep "^s" single-hop.tr | wc -l
27121
```

Π.χ. για datarate = 4Mbps = 4\*10<sup>6</sup> bps

```

a@a-VirtualBox:~/Desktop/Ex1$ ns single-hop.tcl -drate 4.0e6
num_nodes is set 2
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
NS EXITING...
a@a-VirtualBox:~/Desktop/Ex1$ grep "^r" single-hop.tr | wc -l
21225
a@a-VirtualBox:~/Desktop/Ex1$ grep "^s" single-hop.tr | wc -l
31981

```

Π.χ. για  $\text{datarate} = 9.5\text{Mbps} = 9.5 \cdot 10^6 \text{ bps}$

```

a@a-VirtualBox:~/Desktop/Ex1$ ns single-hop.tcl -drate 9.50e6
num_nodes is set 2
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
NS EXITING...
a@a-VirtualBox:~/Desktop/Ex1$ grep "^r" single-hop.tr | wc -l
27594
a@a-VirtualBox:~/Desktop/Ex1$ grep "^s" single-hop.tr | wc -l
37050

```

Π.χ. για  $\text{datarate} = 11\text{Mbps} = 11 \cdot 10^6 \text{ bps}$

```

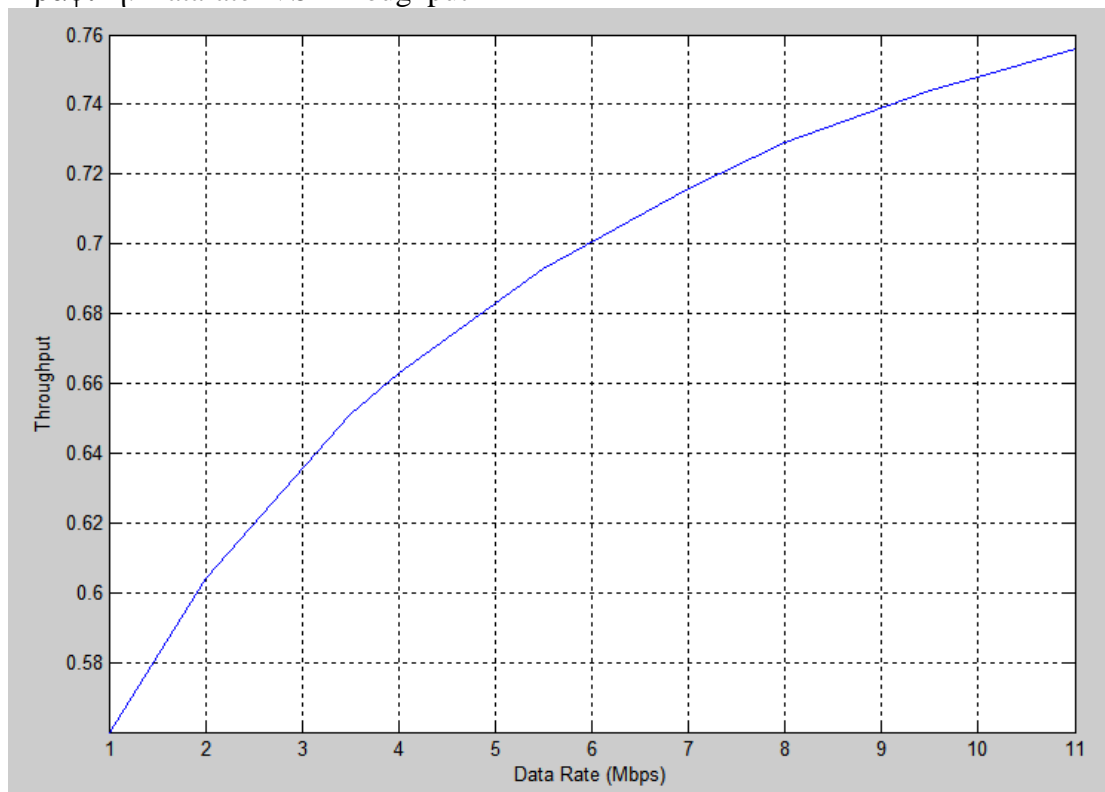
a@a-VirtualBox:~/Desktop/Ex1$ ns single-hop.tcl -drate 11.0e6
num_nodes is set 2
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
NS EXITING...
a@a-VirtualBox:~/Desktop/Ex1$ grep "^r" single-hop.tr | wc -l
28815
a@a-VirtualBox:~/Desktop/Ex1$ grep "^s" single-hop.tr | wc -l
38069

```

Πίνακας αποτελεσμάτων για διαφορετική παράμετρο του ρυθμού μετάδοσης data rate

Data Rate (Mbit)	Receive	Sent	Throughput R/S
1	15199	27121	0.560
2	17531	29049	0.604
3.5	20376	31289	0.651
4	21225	31981	0.663
5.5	23374	33691	0.693
7	25184	35146	0.716
8	26264	36022	0.729
9.5	27594	37050	0.744
10	28042	37441	0.748
11	28815	38069	0.756

Γραφική: Datarate VS Throughput

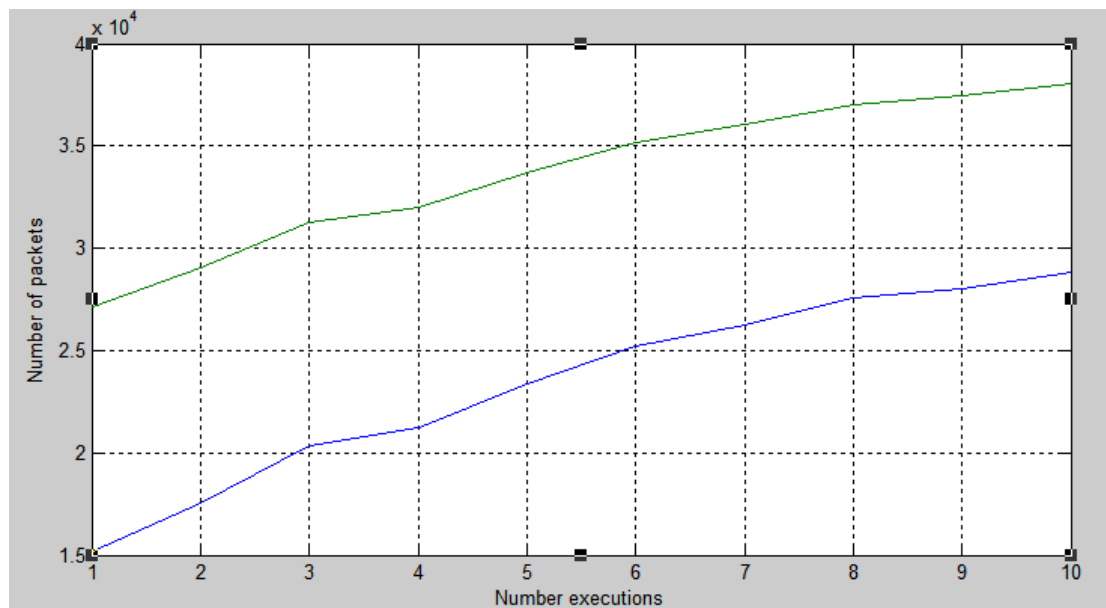


Με την αύξηση του ρυθμού μετάδοσης δεδομένων το throughput αυξάνεται διότι δεν περιορίζεται το κανάλι επικοινωνίας μεταξύ S και D και αξιοποιείται παραπάνω το εύρος ζώνης bandwidth.

Γραφική των μεταβαλλόμενων πακέτων για τις 10 εκτελέσεις της προσομοίωσης .

Η μπλε -> received packets

Η πράσινη -> sent packets

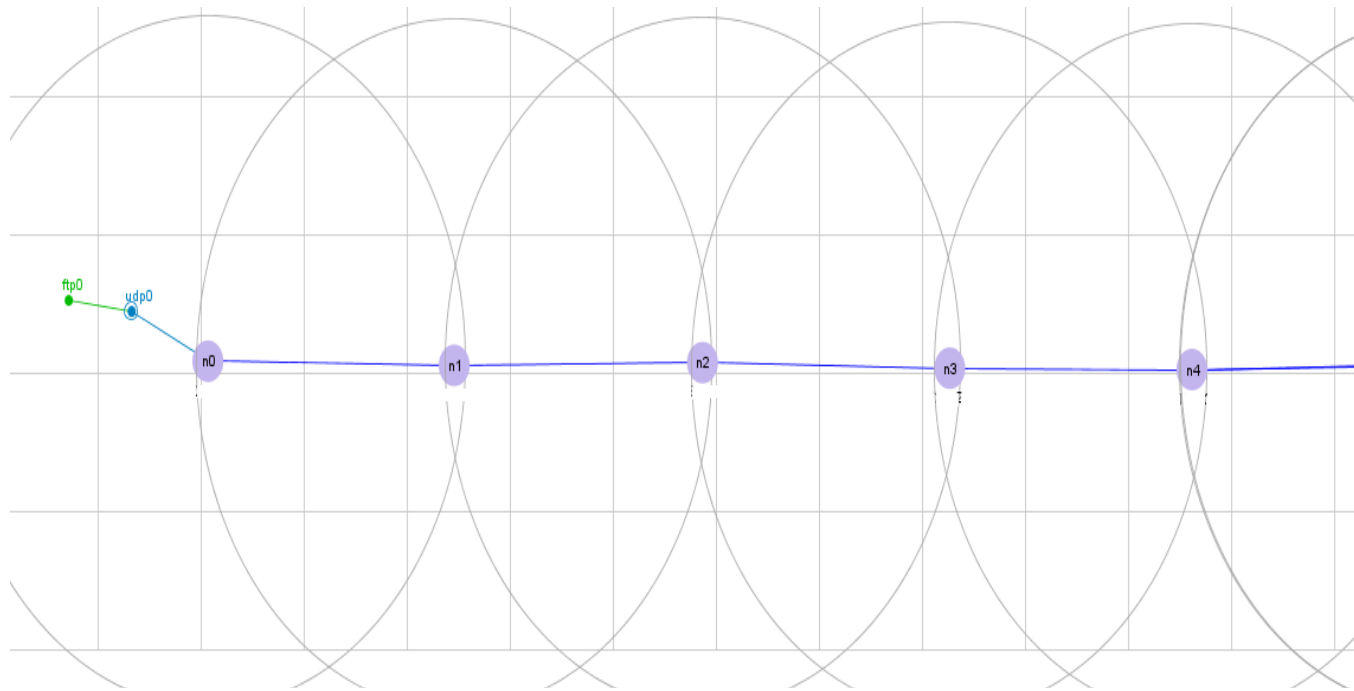


### 9.3 Προσομοίωση 2ο Μέρους:

Mac 802.11 πάνω σε η κόμβους όταν προσθέτουμε κίνηση

Στο 2<sup>ο</sup> μέρος, προσομοιώνεται η ασύρματη Mac/802.11 διάδοση πακέτων από τον κόμβο Source προς τον κόμβο Destination, όπου μεσολαβούν ενδιάμεσοι κόμβοι ανά απόσταση 200m, σε διάστημα χρόνου 10sec. Σε επίπεδο εφαρμογής, χρησιμοποιείται το πρωτόκολλο CBR (Constant Bit Rate) για την παραγωγή και αποστολή πακέτων πάνω στο πρότυπο UDP.

Στην προσομοίωση το Data Rate παραμένει σταθερό = 11Mbps, όπως και το Bandwidth = 22Mbit. Αυτό που αλλάζει στο multi hop δίκτυο, είναι ο αριθμός των κόμβων που μεσολαβούν ανάμεσα στους κόμβους Source και Destination. Παρουσιάζονται στην συνέχεια πως μεταβάλλεται το data rate σε σχέση με την αύξηση των κόμβων.



### 9.3.1. Η Διαδικασία μέτρησης αποτελεσμάτων:

Με τις εντολές:

- ***ns multi-hop.tcl -drate {data-rate}*** , τρέχουμε την προσομοίωση δίνοντας ως παράμετρο (argument) το datarate .

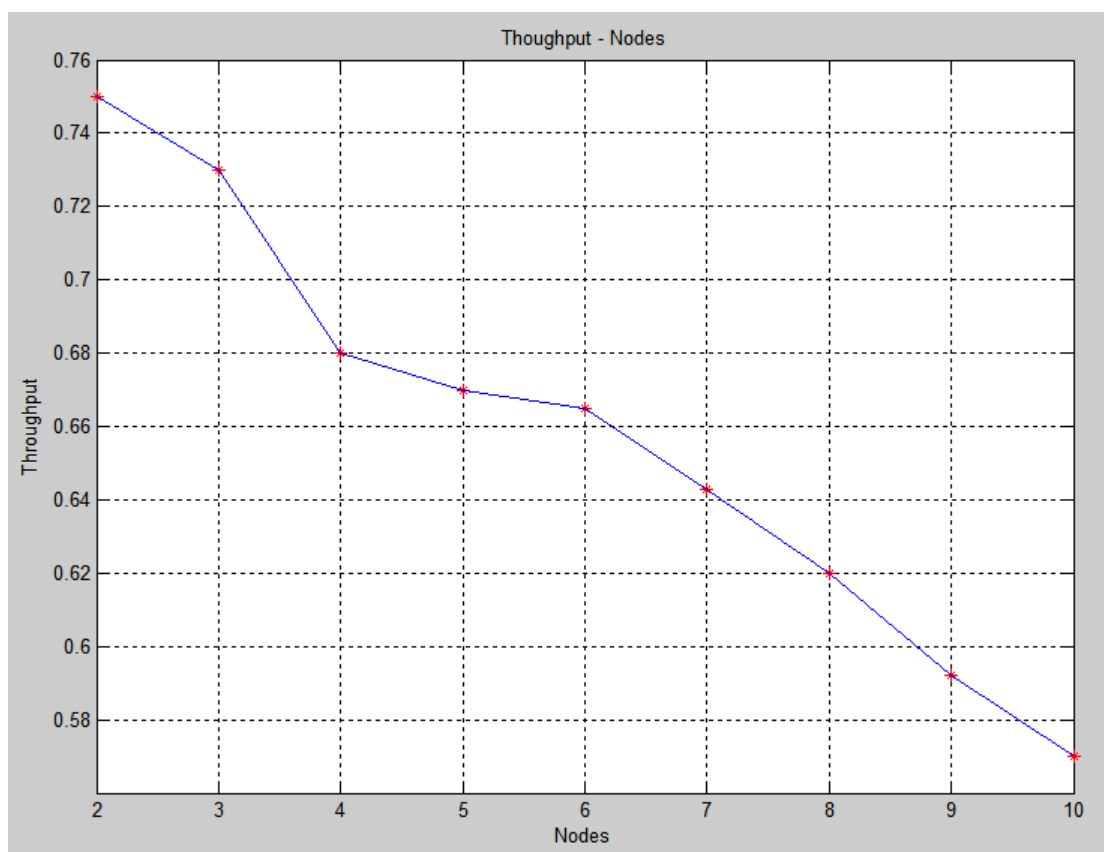
Μετά από κάθε προσομοίωση θα έχουμε ως έξοδο δύο αρχεία

- ***multi-hop.tr*** , έξοδος αρχείου trace πακέτων προσομοίωσης.
  - ***multi-hop.nam*** , έξοδος αρχείου γενικού προσομοίωσης.
- ***grep "^r" {trace file name} | wc -l*** , επιστρέφει τον αριθμό παραλαμβανόμενων πακέτων.
  - ***grep "^s" {trace file name} | wc -l*** , επιστρέφει τον αριθμό των πακέτων που στάλθηκαν.

Πίνακας αποτελεσμάτων για διαφορετικό πλήθος κόμβων

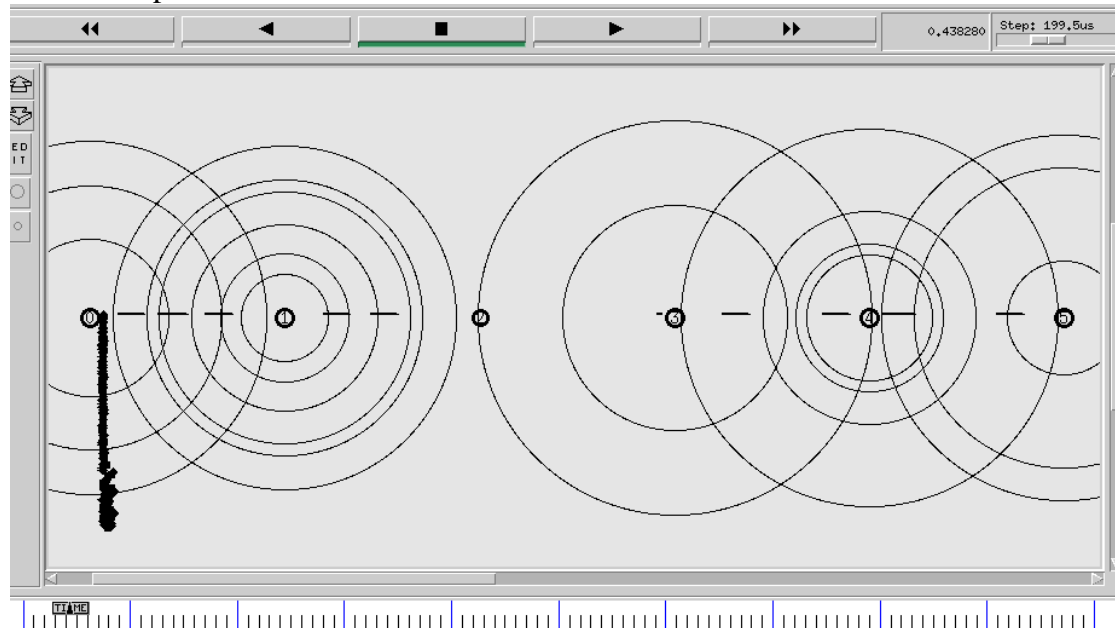
Κόμβοι	Receive	Sent	Throughput R/S
2	28839	38069	0.757
3	27733	37990	0.73
4	26249	38602	0.68
5	25987	38787	0.67
6	24842	37357	0.665
7	25022	38915	0.643
8	20962	33810	0.62
9	18939	31991	0.592
10	22272	39073	0.57

Γραφική Throughput - Nodes



Ο λόγος ο οποίος μειώνεται κατά ελάχιστο το throughput καθώς αυξάνουμε το πλήθος των κόμβων είναι, ότι η κίνηση μεταδίδεται και στους υπόλοιπους κόμβους με αποτέλεσμα να χάνουν επιπρόσθετα και αυτοί πακέτα. Αναφέρεται πως κυρίως η αποτυχία των πακέτων είναι συχνότερη από τον πρώτο κόμβο στον δεύτερο, διότι υπάρχει ο μεγαλύτερος φόρτος κίνησης καθώς ο κόμβος Source παράγει και στέλνει την κίνηση στο hop ασύρματο δίκτυο.

Προσομοίωση για 6 κόμβους:  
ns multi-hop.tcl -nn 6





## Βιβλιογραφική Αναφορά

Από επιστημονικές εργασίες (papers):

- An Introduction to NS, Nam and OTcl Scripting ,Paul Meenaghan and Declan Delaney
- Heterogeneous Cellular Networks: From Theory to Practice , Amitabha Ghosh, Nitin Mangalvedhe
- Cloud-based Network Simulator for Next Generation Wireless Technology (WiMAX and LTE) , Gerson Bacor
- Overhaul of IEEE 802.11 Modeling and Simulation in NS-2 , Qi Chen , Felix Schmidt-Eisenlohr , Daniel Jiang
- Advances in Network Simulation , Lee Breslau, Deborah Estrin, Kevin Fall, Sally Floyd, John Heidemann
- A Survey of Network Simulation Tools: Current Status and Future Developments, Julian Pan

Από Βιβλία – Διπλωματικές:

- Υλοποίηση αλγορίθμου και προσομοίωση για την βελτίωση διαχείρισης φορητών συσκευών σε ετερογενή ασύρματα δίκτυα - διπλωματική πανεπιστημίου Πατρών, Αυλωνίτης Αθανασίου και Λαμπροπούλου Αναστασίου.
- Παροχή προηγούμενων υπηρεσιών με επίγνωση περιβάλλοντος σε Δυναμικά Ετερογενή Δίκτυα – Διδακτορική Διατριβή , Αναστάσιος Α. Ζαφειρόπουλος
- ΕΤΕΡΟΓΕΝΗ ΣΥΣΤΗΜΑΤΑ ΚΙΝΗΤΗΣ ΤΗΛΕΦΩΝΙΑΣ GSM - UMTS , Διπλωματική, Μητρόπουλος Σ Γεώργιος
- HETEROGENEOUS NETWORKS AN INTRODUCTION, Prof. Dr.Ing. Armin Dekorsy, Mahmoud El-Arabied

Από το διαδίκτυο:

- <http://www.slideshare.net/engmahmoudELarabied/heterogeneous-networks-an-introduction>
- <http://www.crhc.illinois.edu/wireless/assignments/>
- <http://nile.wpi.edu/NS/>
- <http://www.himublog.com/2012/12/how-to-create-simple-wireless-scenario-on-ns2/>
- <http://www.isi.edu/nsnam/ns/tutorial/>
- <http://ns2-master.blogspot.gr/2011/04/sample-coding-in-wireless.html>