

Δίκτυα Επικοινωνιών

6η εργαστηριακή άσκηση

Επίδοση Τοπικών Δικτύων IEEE 802.3

Γεώργιος Δασούλας
Α.Μ: 03112010
6ο Εξάμηνο 2014-2015

12 Μαΐου 2015

Σε αυτή την άσκηση θα μελετηθεί η επίδοση του *MAC* πρωτοκόλλου *IEEE 802.3*. Για το σκοπό αυτό θα μετρηθεί το πλήθος των πακέτων και η ποσότητα των δεδομένων που μεταδόθηκαν επιτυχώς κατά τη διάρκεια της προσομοίωσης, καθώς και η μέση καθυστέρηση των πακέτων. Από αυτά τα δεδομένα θα γίνει υπολογισμός της χρησιμοποίησης του καναλιού και σύγκριση με τα θεωρητικά αναμενόμενα αποτελέσματα.

Μελέτη απόδοσης τοπικού δικτύου IEEE 802.3

Με βάση την προσομοίωση που εκτελέσαμε θα επαληθεύσουμε κατά πόσον ισχύει ή όχι η εξίσωση: $\eta = \frac{P}{P + 2 * \tau * e}$. Όπου:

- η είναι η απόδοση του διαύλου (πραγματικός ρυθμός μεταφοράς δεδομένων / ονομαστικό ρυθμό μετάδοσης ζεύξης)
- P είναι ο χρόνος μετάδοσης κάθε πλαισίου (Μήκος πλαισίου σε bit / ονομαστικό ρυθμό μετάδοσης ζεύξης)
- τ είναι η μέγιστη επιτρεπόμενη καθυστέρηση διάδοσης πάνω στο καλώδιο, την οποία μπορείτε να θεωρήσετε ίση με 25,6 μ s.
- $e = 2,718281828$

Για την επαλήθευση της θεωρητικής τιμής θα εκτελέσουμε μια σειρά προσομοιώσεων, η οποία πραγματοποιείται με μεταβαλλόμενο μέγεθος πακέτου από 64 Byte (ελάχιστο μέγεθος πλαισίου) μέχρι 1500 Byte (μέγιστο μέγεθος πλαισίου). Πραγματοποιήσαμε 10 προσομοιώσεις. Για την πειραματική τιμή χρησιμοποιήσαμε τον τύπο: $\eta = \frac{\text{numberpackets} * \text{packetsize} * 8}{(\text{endtime} - \text{starttime}) * 10000000}$.

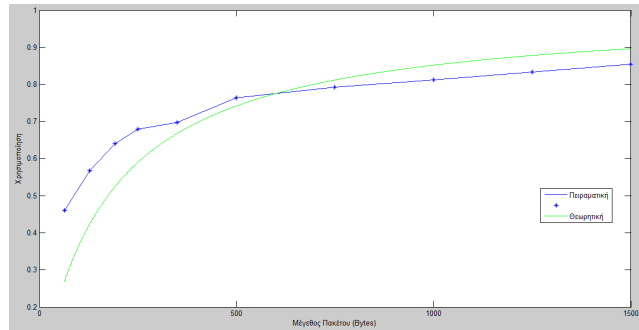
Μετά τον τροποποιημένο κώδικα ακολουθούν οι καμπύλες : Για τη χάραξη της γραφικής με το χρόνο τελευταίας επιτυχημένης λήψης τροποποιήσαμε τον κώδικα *awk* ως εξής :

```
BEGIN {
    sum_delay = 0;
    bufferspace = 100000;
    total_pkts_sent = 0;
    total_pkts_recv = 0;
    lastpackt_recv=0;
    total_pkts_dropped = 0;

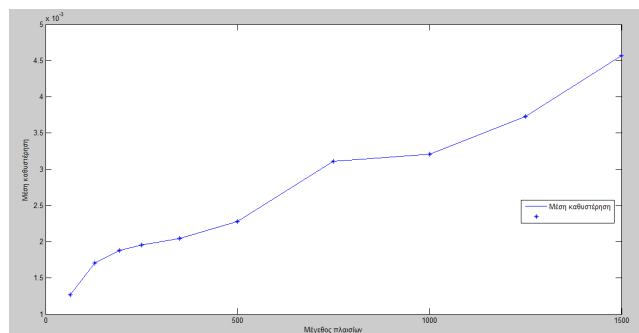
    if (sprintf(sqrt(2)) ~ /,/) dmfix = 1;
}

{
    if (dmfix) sub(/\. /, "", $0);
}
/^h/ && /cbr/ {
    total_pkts_sent++;
}
/^d/ && /cbr/ {
    total_pkts_dropped++;
}
8
/^-/ && /cbr/ {
    sendtimes[$12%bufferspace] = $2;
}
/^r/ && /cbr/ {
    pkts_recv++;
    sum_delay += $2 - sendtimes[$12%bufferspace]
    lastpackt_recv=$2;
}
END {

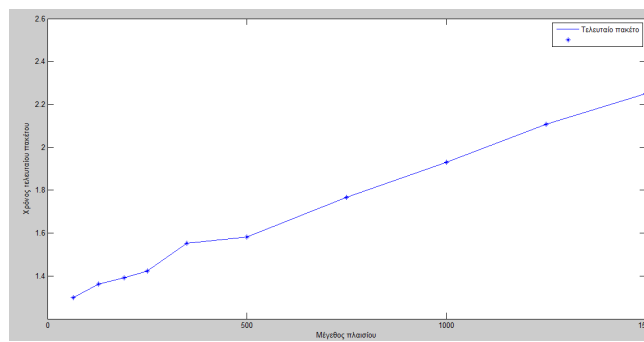
    printf("Last packet received \t: %f sec\n", 1.0*lastpackt_recv);
}
```



Σχήμα 1: Χρησιμοποίηση καναλιού



Σχήμα 2: Μέση καθυστέρηση



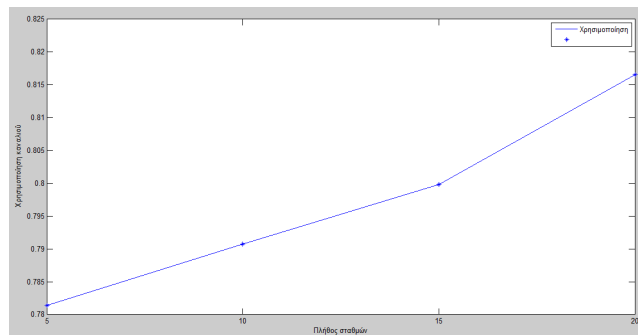
Σχήμα 3: Τελευταία λήψη

Από τις παραπάνω γραφικές βλέπουμε πως είναι προτιμότερο να χρησιμοποιούμε μεγάλα πλαίσια. Αυτό προκύπτει τόσο από την πειραματική καμπύλη που βλέπουμε να πλησιάζει προς το 1, όσο και από τη θεωρητική γραφική παράσταση, καθώς όσο το μέγεθος του πλαισίου μεγαλώνει, τόσο η χρησιμοποίηση του καναλιού προσεγγίζει το 1. Βέβαια, παρατηρώντας και τις τρεις γραφικές, συμπεραίνουμε πως όσο μεγαλώνουμε το πλαίσιο, τόσο αυξάνεται η μέση καθυστέρηση και ο χρόνος της τελευταίας επιτυχημένης λήψης. Γι' αυτό, αποσκοπώντας στη μεγαλύτερη χρησιμοποίηση του καναλιού έχουμε κάποιο κόστος στην καθυστέρηση και χρόνο λήψης.

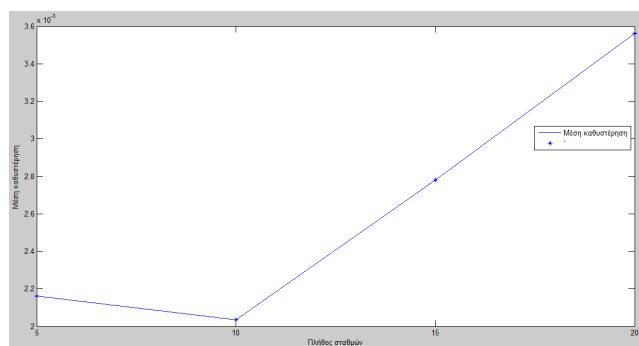
Επίδραση αριθμού σταθμών

Με την κατάλληλη τροποποίηση του *awk* και του *tcl* αρχείου προέκυψαν τα παρακάτω:

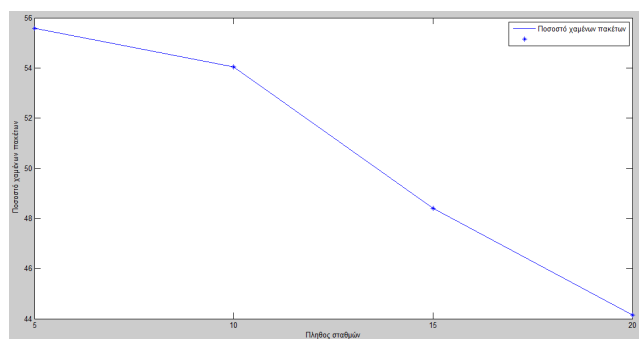
Χρησιμοποίηση	Καθυστέρηση	Ποσοστό απορριφθέντων	Πλήθος σταθμών
0.781441	0.002163	55.5783	5
0.790721	0.002034	54.0491	10
0.799827	0.002781	48.4032	15
0.816552	0.003562	44.1465	20



Σχήμα 4: Χρησιμοποίηση καναλιού



Σχήμα 5: Μέση καθυστέρηση



Σχήμα 6: Ποσοστό χαμένων πακέτων

(β) Καταρχάς , παρατηρούμε πως η χρησιμοποίηση του καναλιού αυξάνεται , όχι αισθητά, με την αύξηση των σταθμών, κάτι το οποίο οφείλεται στο ότι οι περισσότεροι κόμβοι διευκολύνουν τη μεταφορά δεδομένων. Επίσης, με την αύξηση των σταθμών έχουμε και μεγαλύτερη καθυστέρηση , κάτι το οποίο και πάλι οφείλεται στο ότι αυξάνεται η τοπολογία μας και στο ότι υπάρχουν περισσότεροι κόμβοι που πρέπει να δώσουν και να πάρουν πακέτα δεδομένων. Τέλος , τα απορριφθέντα πακέτα μειώνονται , κάτι το οποίο οφείλεται και πάλι στην αύξηση των κόμβων , μιας και διευκολύνεται έτσι η μετάδοση των δεδομένων χωρίς υπερχειλίση πληροφοριών.