

Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών

24/4/2016

2^η Σειρά Ασκήσεων

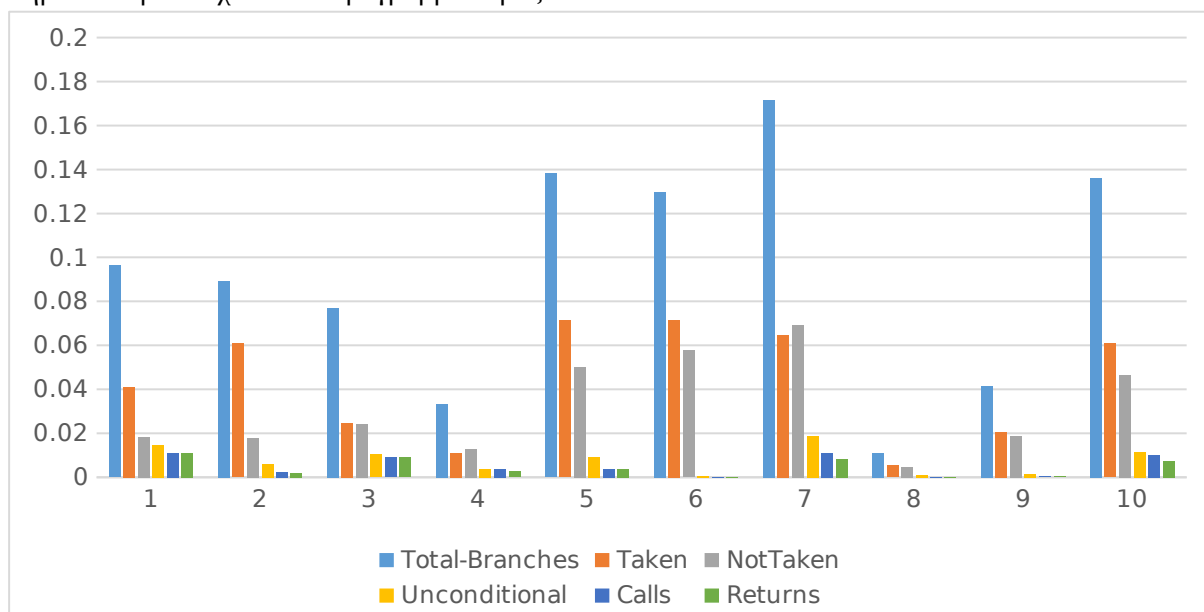
Φανούριος Αραπίδης

A.M.: 031 12 558

H.M.M.Y. 8^ο Εξάμηνο

➤ Μελέτη εντολών άλματος

Σκοπός αυτού του ερωτήματος είναι η συλλογή στατιστικών για τις εντολές άλματος και το ποσό σημαντικό ρόλο έχουν στα προγράμματα μας.



| | |
|----|---------------|
| 1 | Blacksholes |
| 2 | Bodytrack |
| 3 | Canneal |
| 4 | Facesim |
| 5 | Ferret |
| 6 | Fluidanimate |
| 7 | Freqmine |
| 8 | Rtview |
| 9 | Streamcluster |
| 10 | Swaptions |

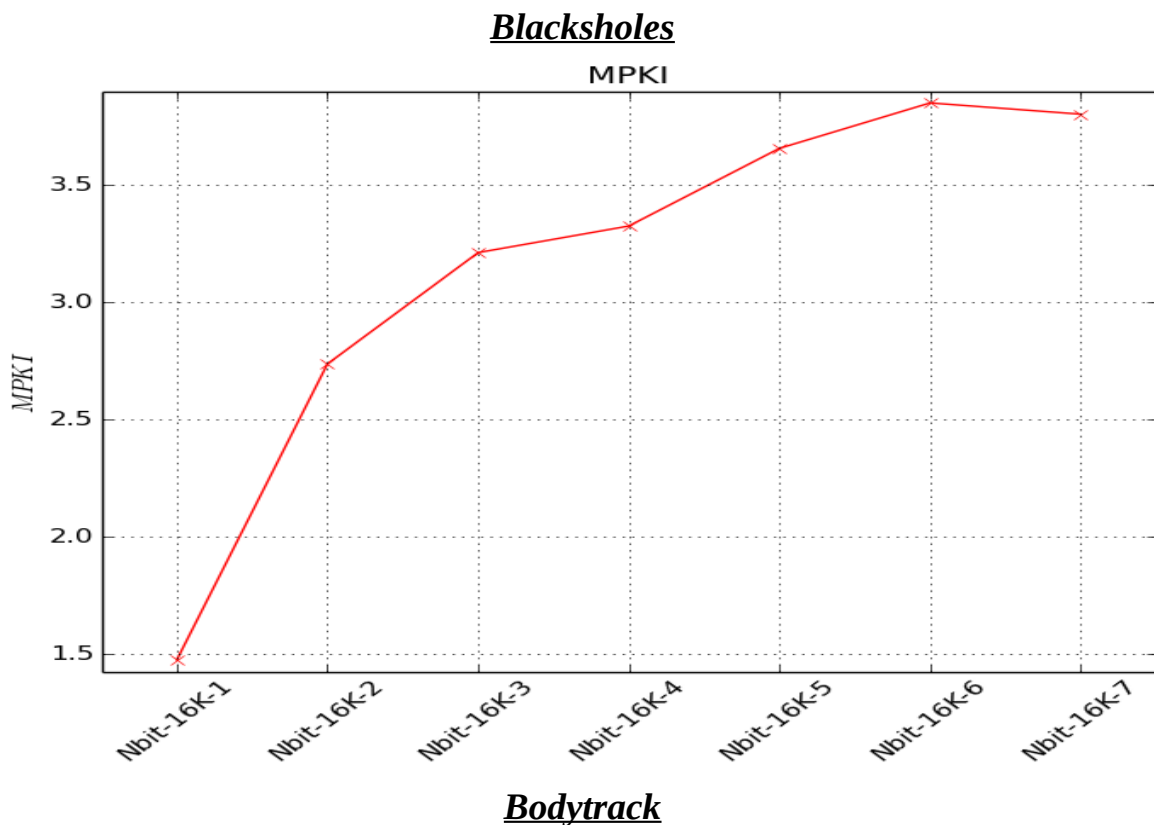
Συμπεράσματα

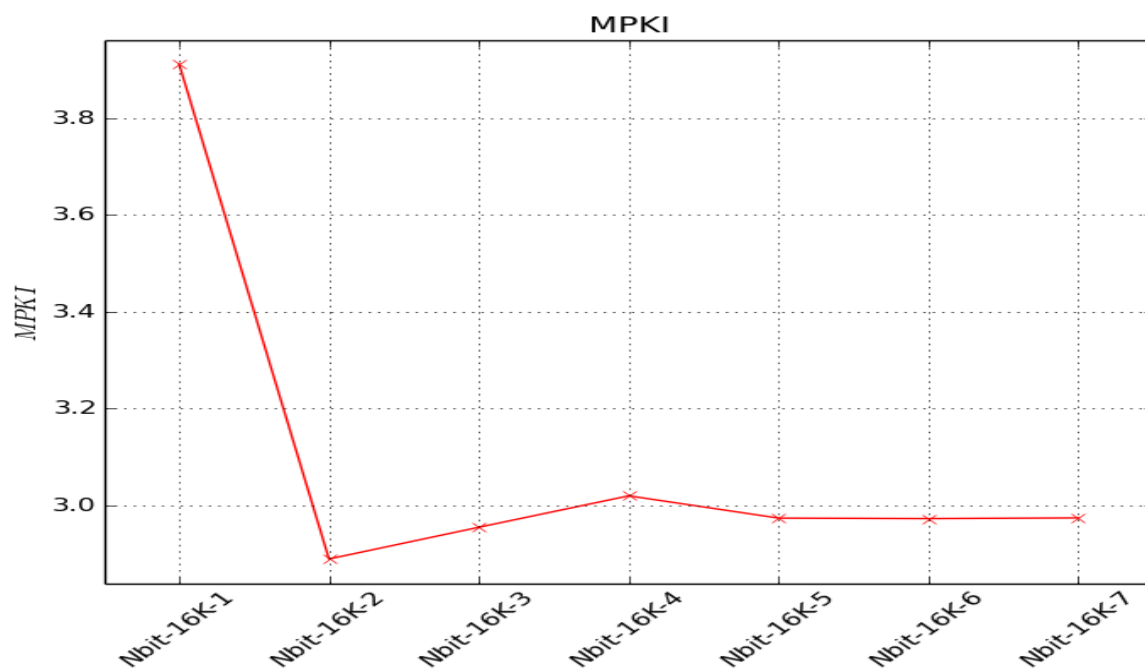
Συνοψίζοντας καταλήγουμε στα εξής συμπεράσματα, περίπου 1/10 είναι εντολή άλματος άρα μπορούμε να συμπεράνουμε πως η εντολής άλματος παίζουν αρκετά σημαντικό ρόλο στην επίδοση των προγραμμάτων μας με σοβαρή εξαίρεση την Rtnview που δεν έχει σχεδόν καθόλου διακλαδώσεις. Αρά μας ενδιαφέρει αρκετά η συμπεριφορά και βελτίωση των προβλέψεων των αλμάτων μέσα στα προγράμματα μας.

➤ Μελέτη των N-bit predictors

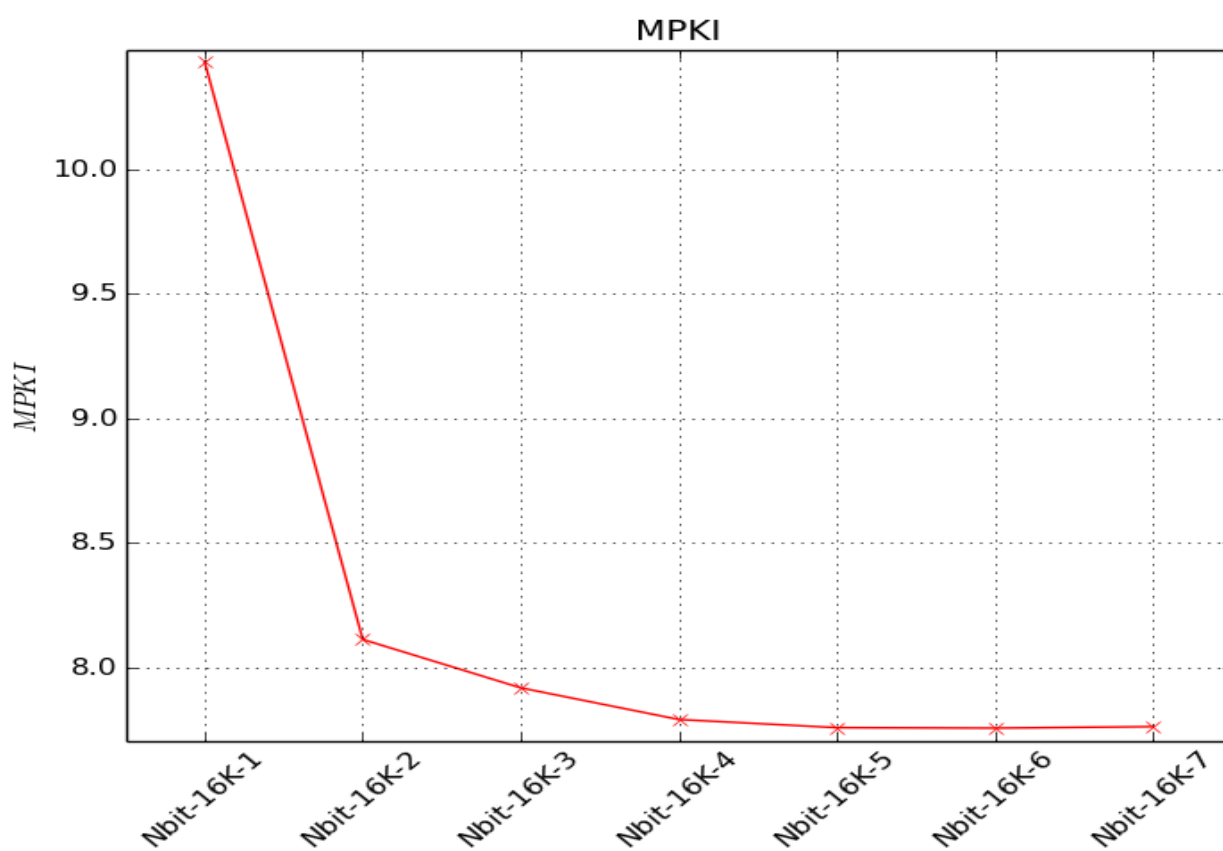
Σε αυτό το κομμάτι της άσκησης θα μελετήσουμε πως μπορεί ένα N-bit predictors να βελτιώσει την απόδοσή των προγραμμάτων μας. Αρχικά θα διατηρήσουμε σταθερό τον αριθμό των BHT entries και θα αυξάνουμε τα bit του predictor και στην συνέχεια θα επαναλάβουμε το ίδιο πείραμα όμως διατηρώντας αυτήν την φορά σταθερό το hardware και ίσο με 32K bits.

- Σταθερό BHT entries

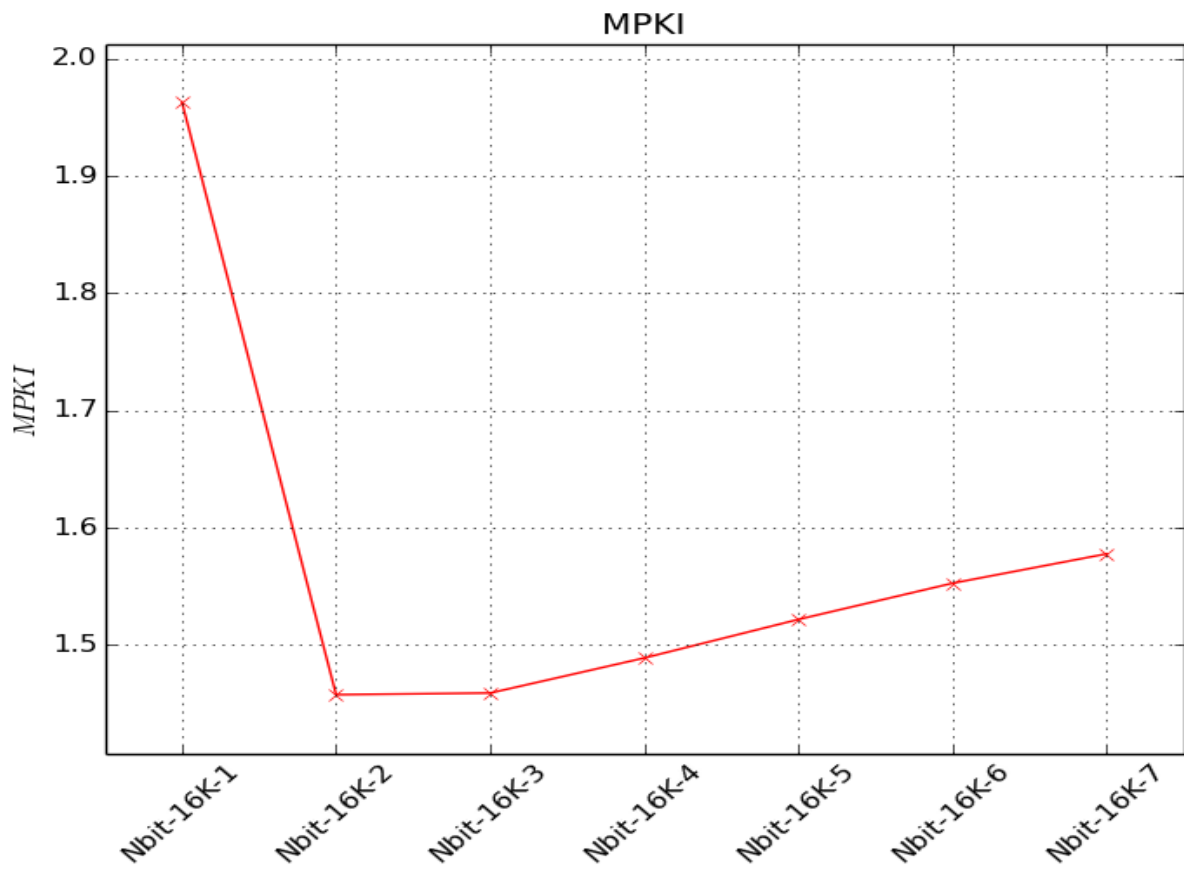




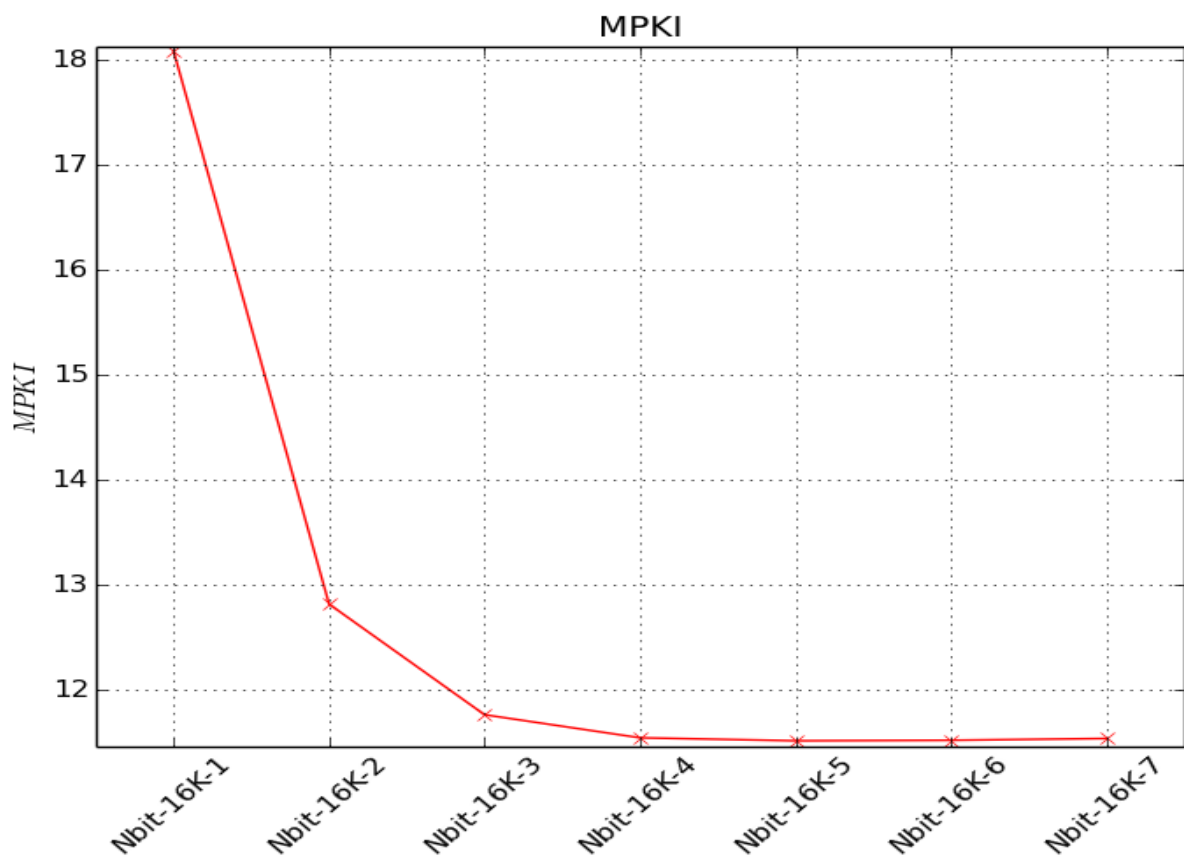
Canneal



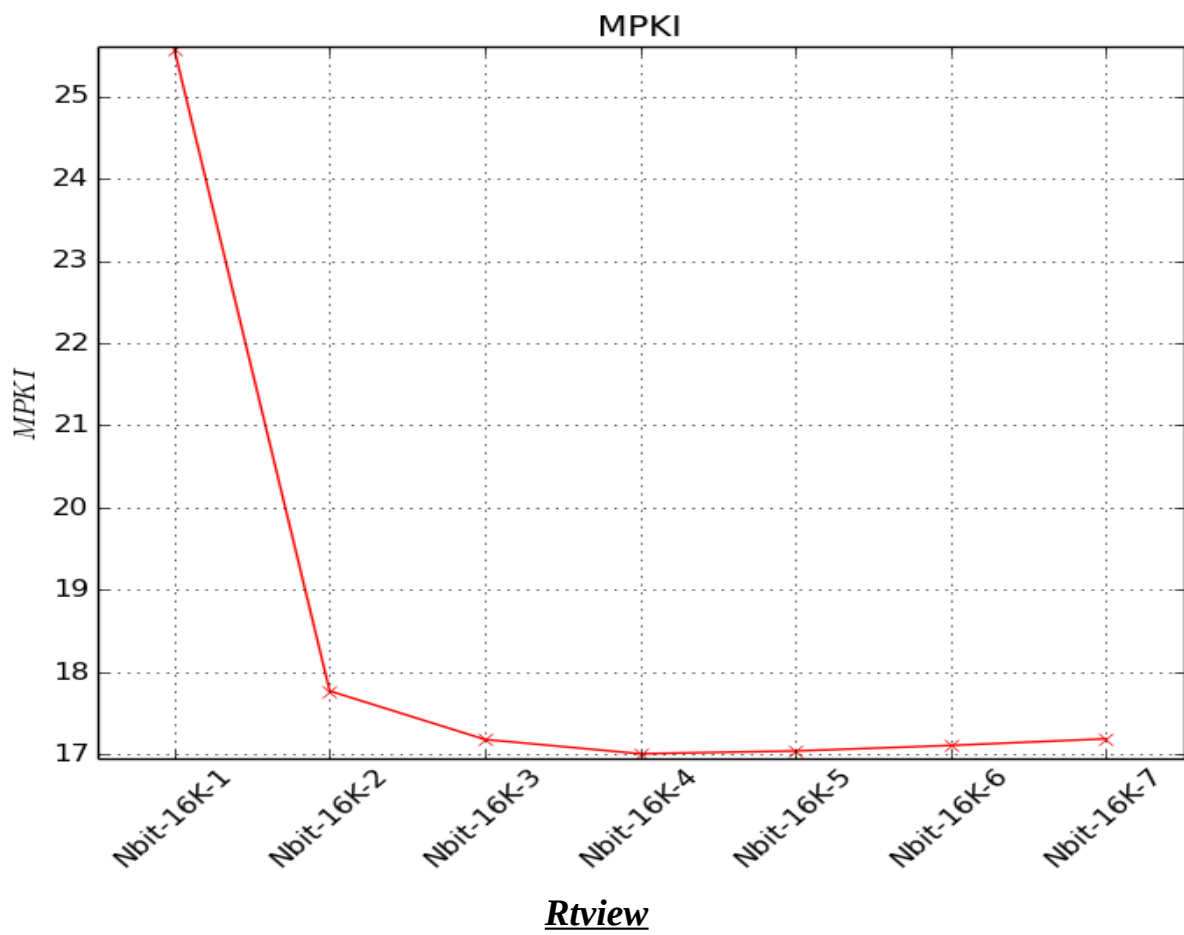
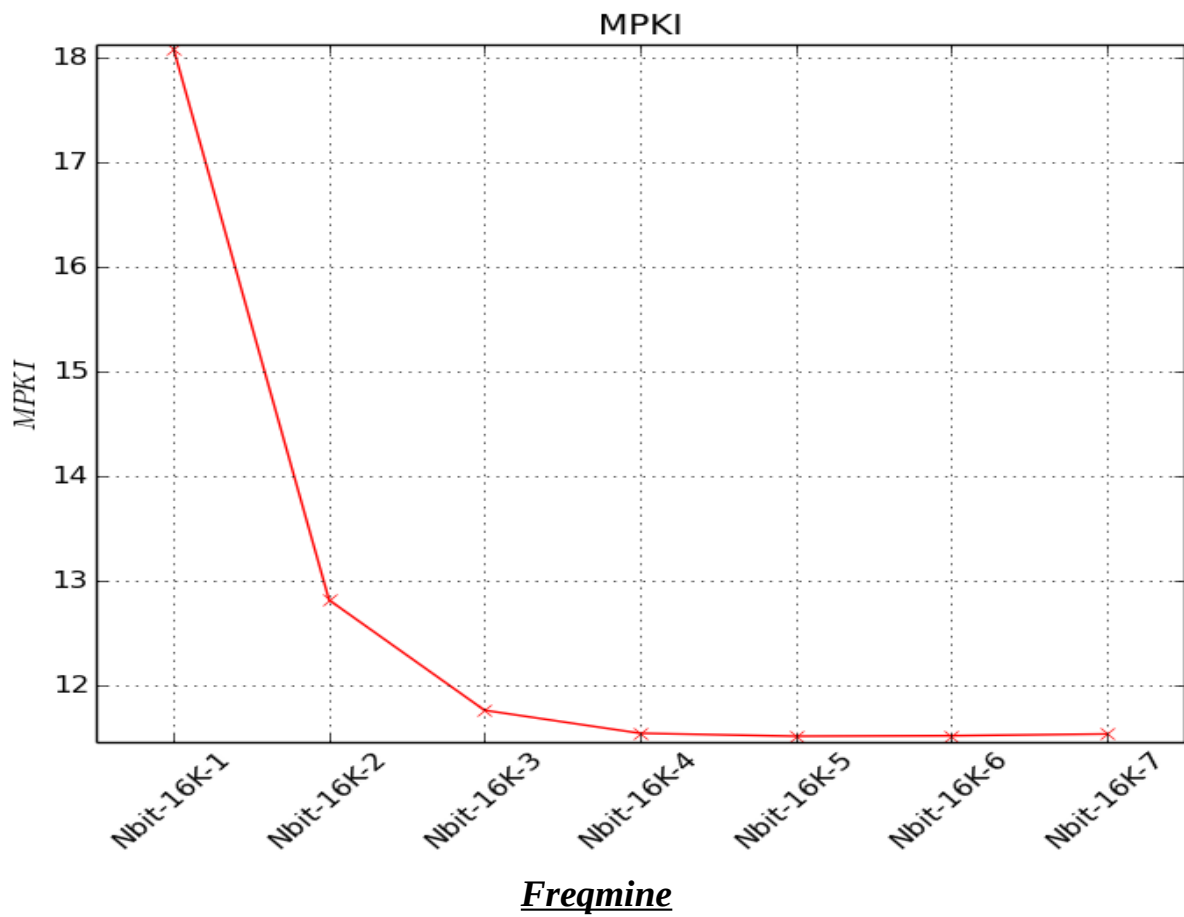
Facesim

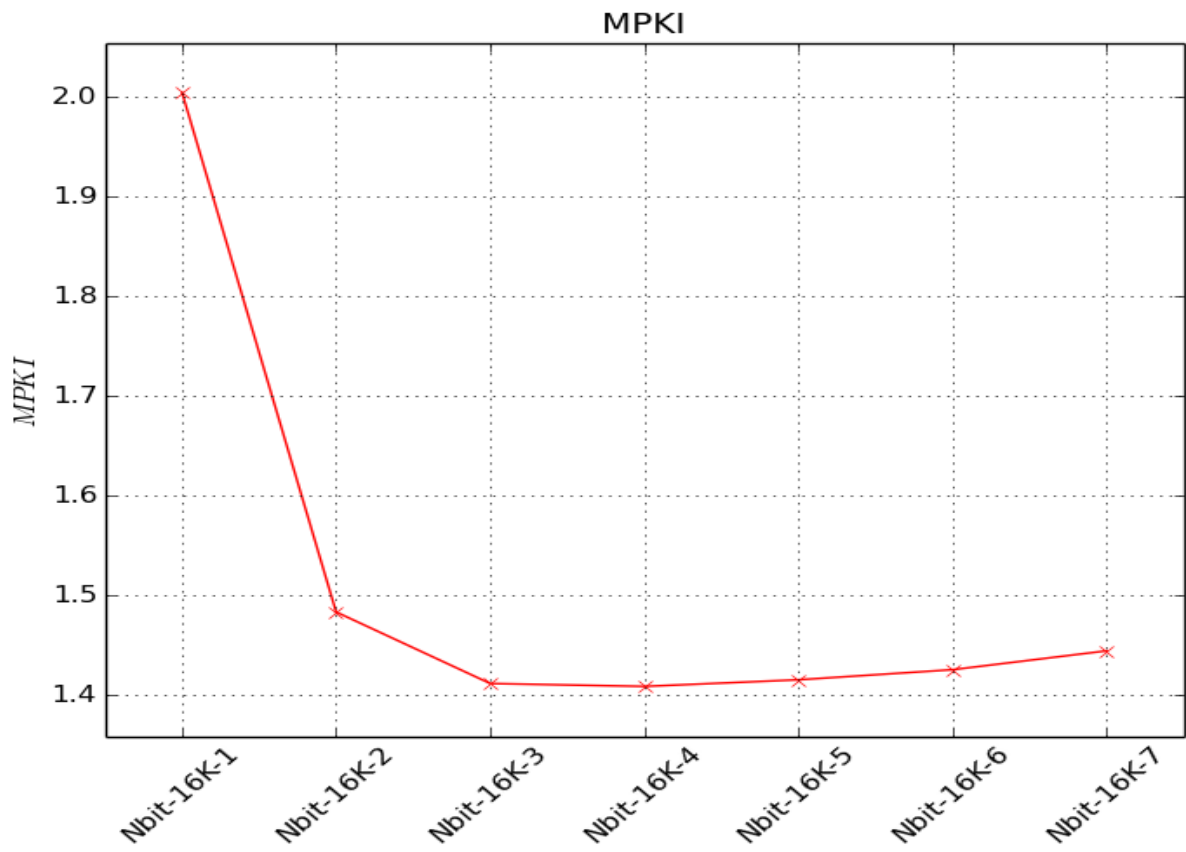


Ferret

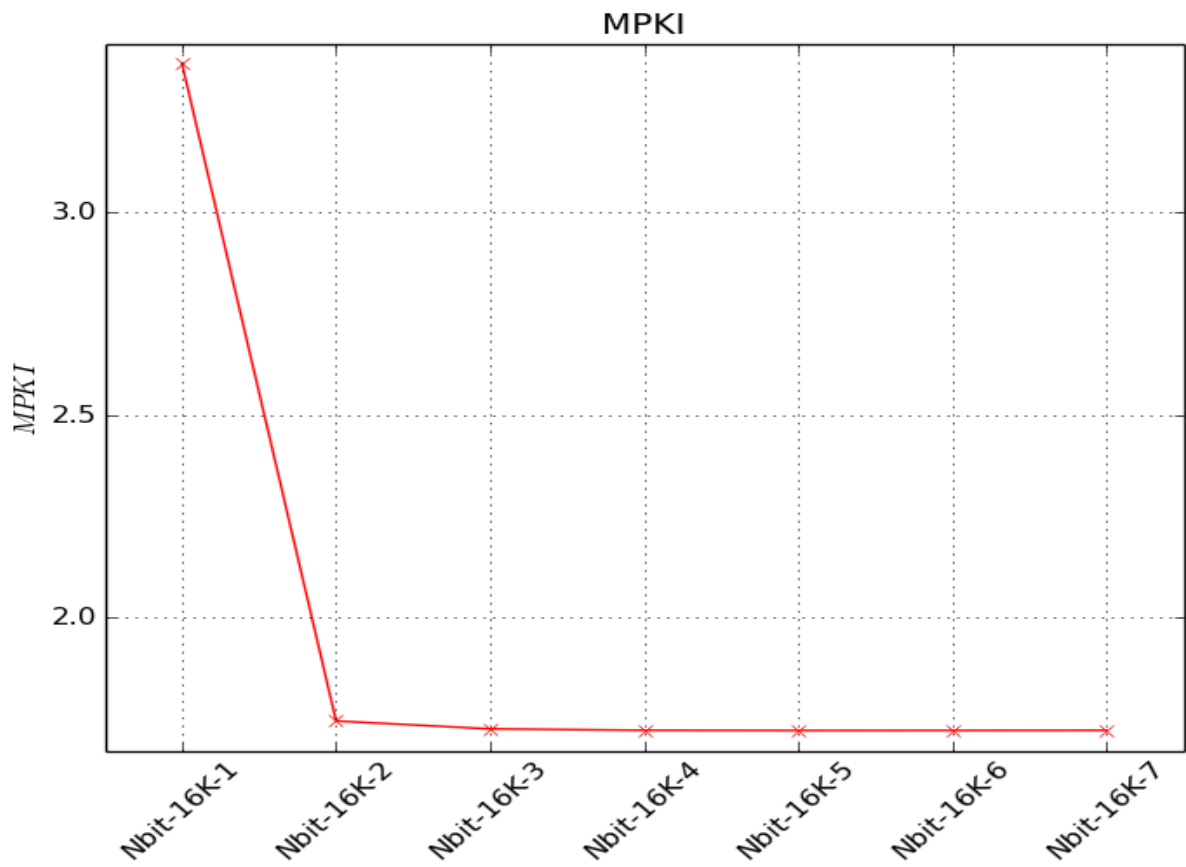


Fluidanimate

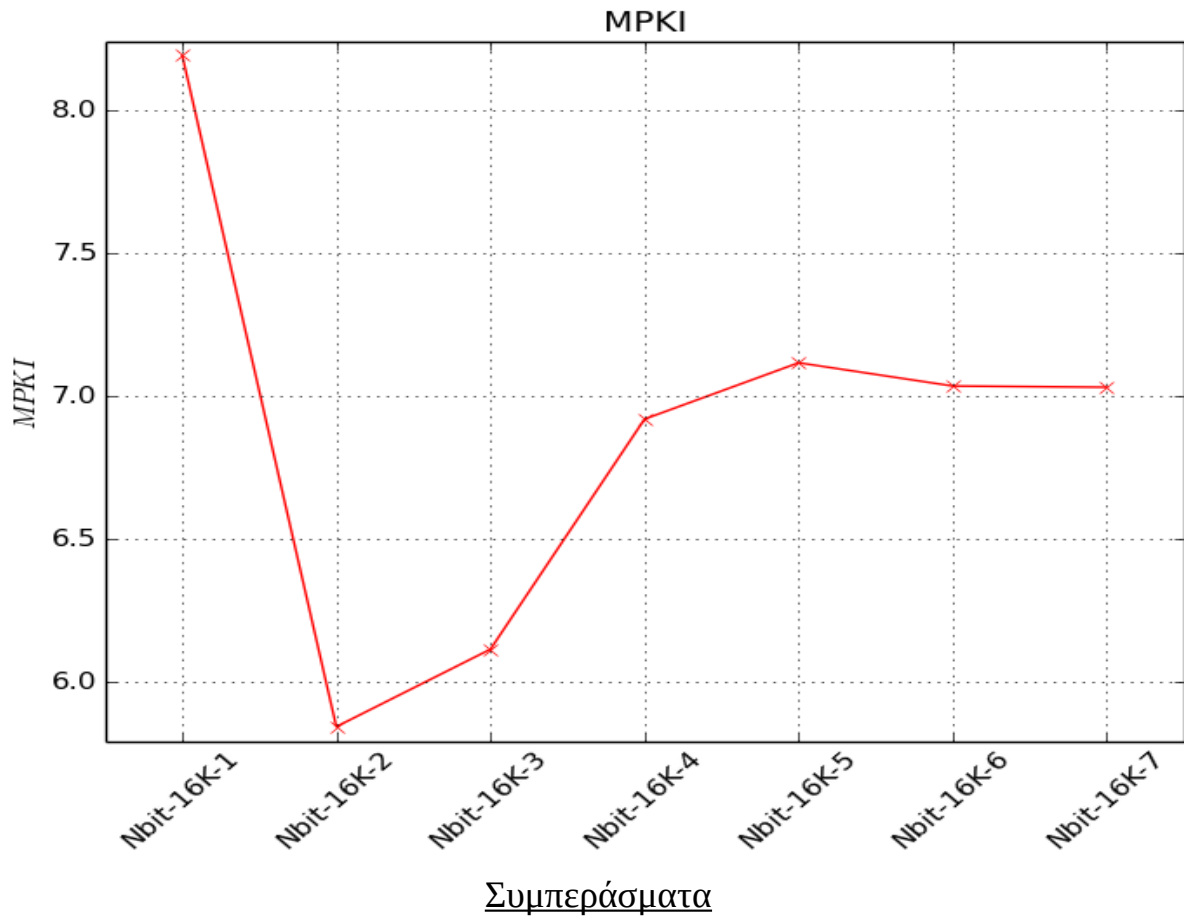




Streamcluster

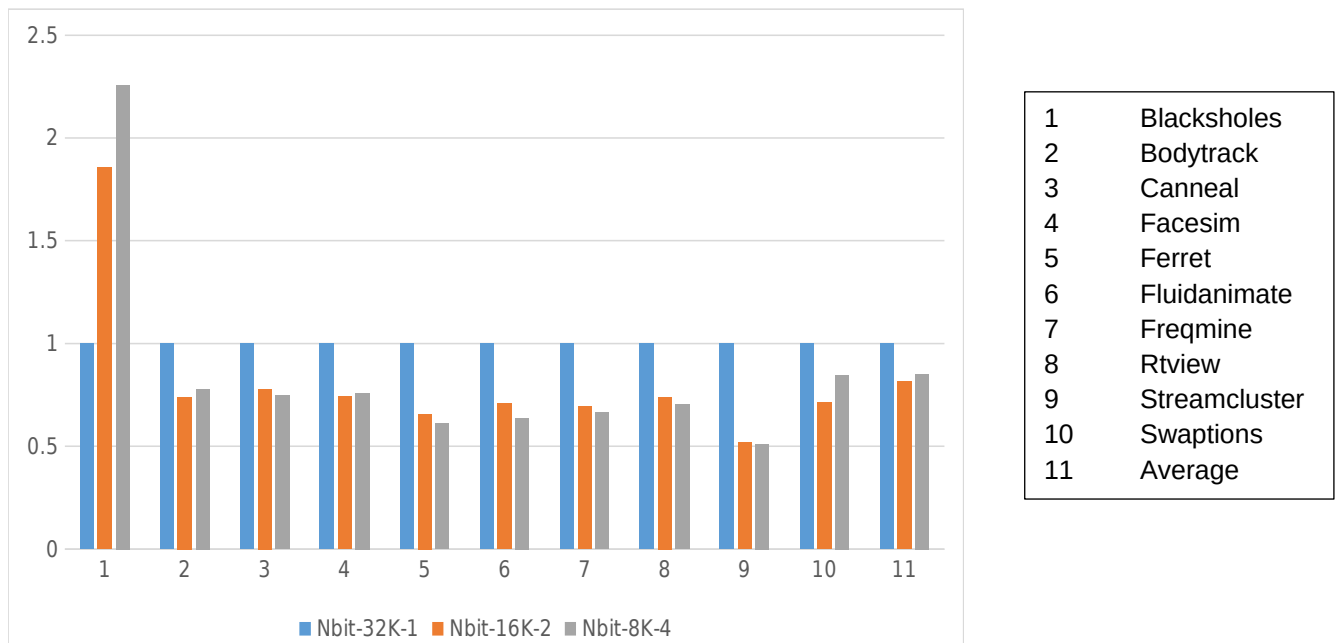


Swaptions



Συνορίζοντας καταλήγουμε στα εξής συμπεράσματα, αρχικά με την αύξηση των bits παρατηρούμε γενικά μια βελτίωση στην απόδοση των προγραμμάτων μας με χαρακτηριστικότερη την βελτίωση από το 1 → 2 bits και με μοναδική εξαίρεση το Blacksholes το οποίο χειροτέρεψε με την αλλαγή. Στη συνέχεια παρατηρούμε ότι με την αύξηση του ενός με δυο bits ακόμα σε γενικές γραμμές τα προγράμματα μας έχουν κάποια βελτίωση όμως είναι αρκετά ελάχιστη σε σχέση με την αρχική βελτίωση. Τέλος με την περεταίρω αύξηση παρατηρούμε πτώση της απόδοσης.

- Σταθερό hardware

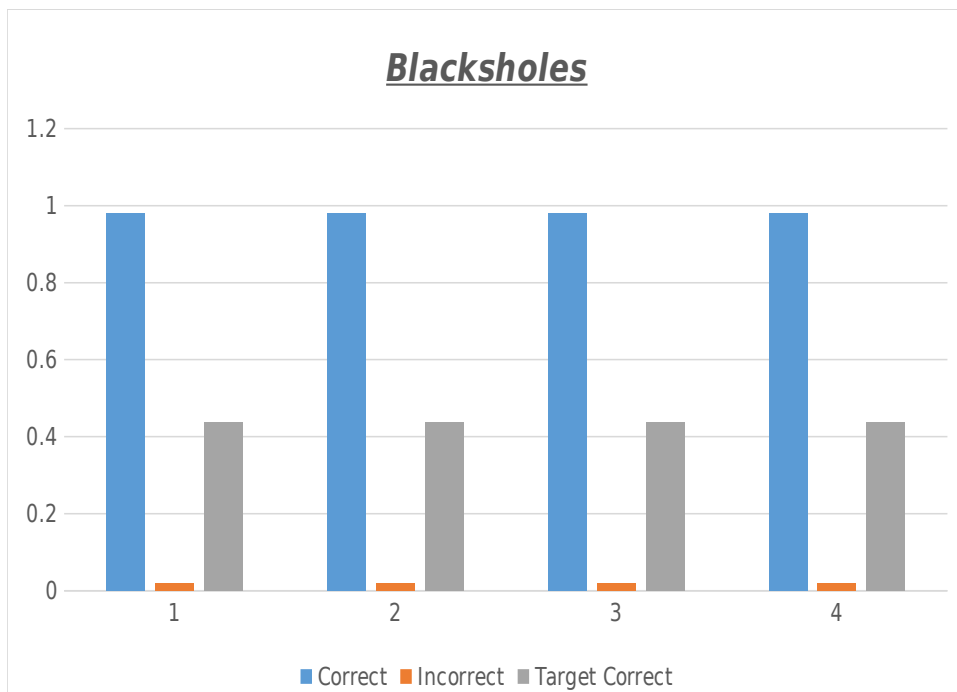


Συμπεράσματα

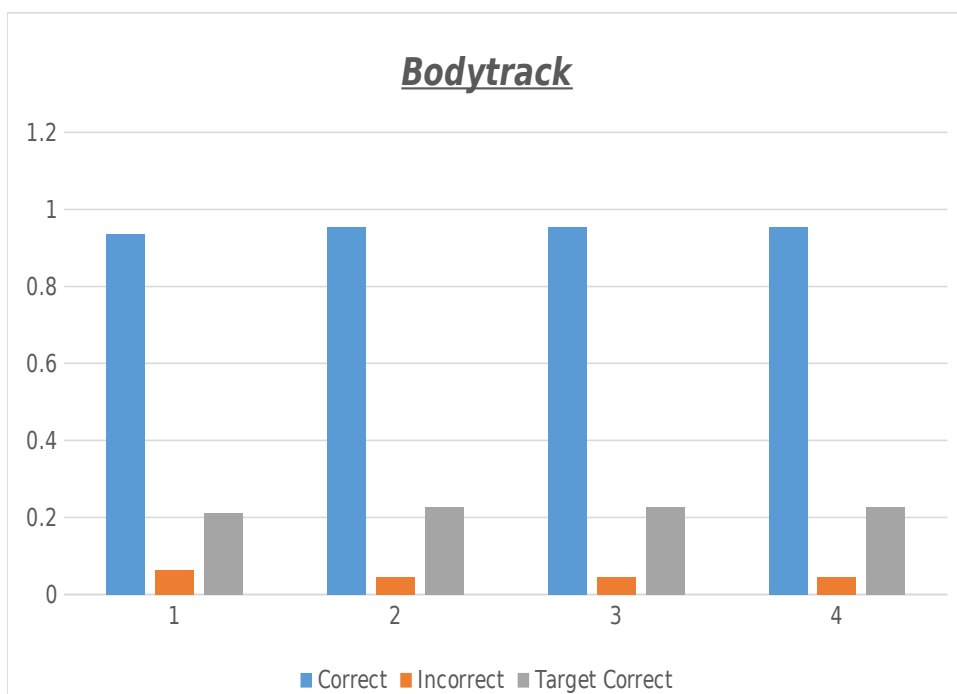
Συνορίζοντας καταλήγουμε στα εξής συμπεράσματα, αρχικά παρατηρούμε ότι όπως και πριν το Blacksholes μειώνετε η απόδοση του όταν αυξάνουν τα bits. Στην συνέχεια παρατηρούμε ότι όπως και πριν για ένα bit έχουν την χειρότερη απόδοση και πως με την αύξηση τους βελτιώνονται γενικά τα προγράμματα μας. Την μέγιστοί βελτίωση την βλέπουμε στο N-16K-2 το οποί είναι αναμενόμενο αποτέλεσμα.

➤ Μελέτη του BTB

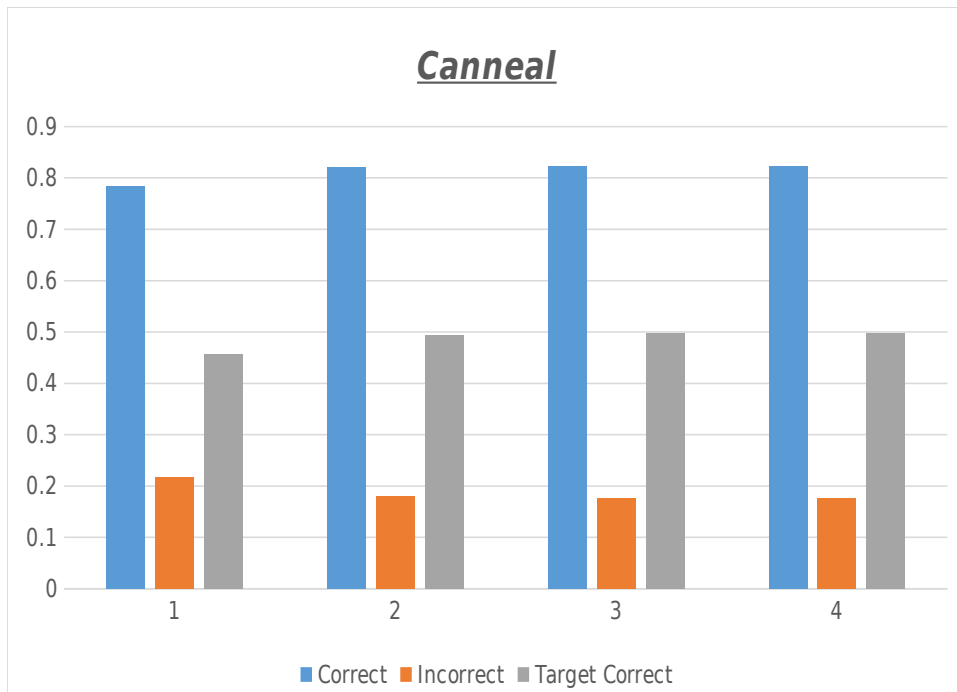
Σε αυτό το κομμάτι της άσκησης θα δούμε πως επηρεάζει ο πίνακας BTB τις προβλέψεις μας. Ο πίνακας BTB περιέχει τις διευθύνσεις από τα taken branches. Αρά εδώ μπορούμε να έχουμε 2 τύπους από λάθη πρώτον να μην κάνουμε σωστή πρόβλεψη για το αν ένα branche είναι taken και δεύτερον αν κάνουμε την σωστή πρόβλεψη να καταλήξουμε σε λάθος διεύθυνση αφού ανάλογα με τις προηγούμενες προβλέψεις αλλάζουν οι τιμές των διευθύνσεων. Τέλος για associativity μεγαλύτερα από 1 χριζόμαστε και μια πολιτική αντικατάστασης, εδώ χρησιμοποιήσαμε την τεχνική FIFO.



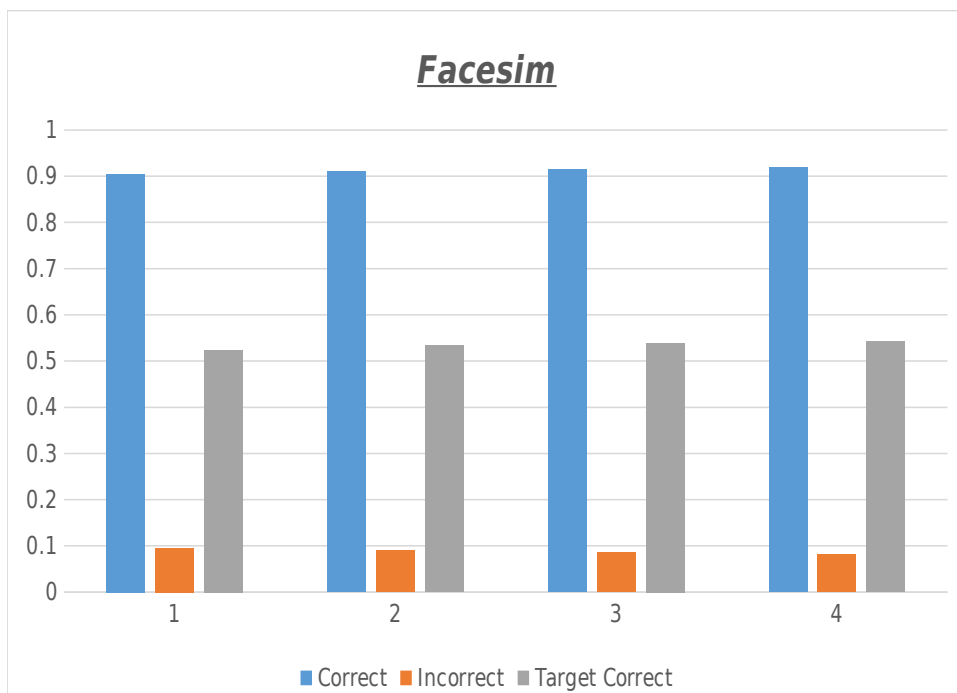
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



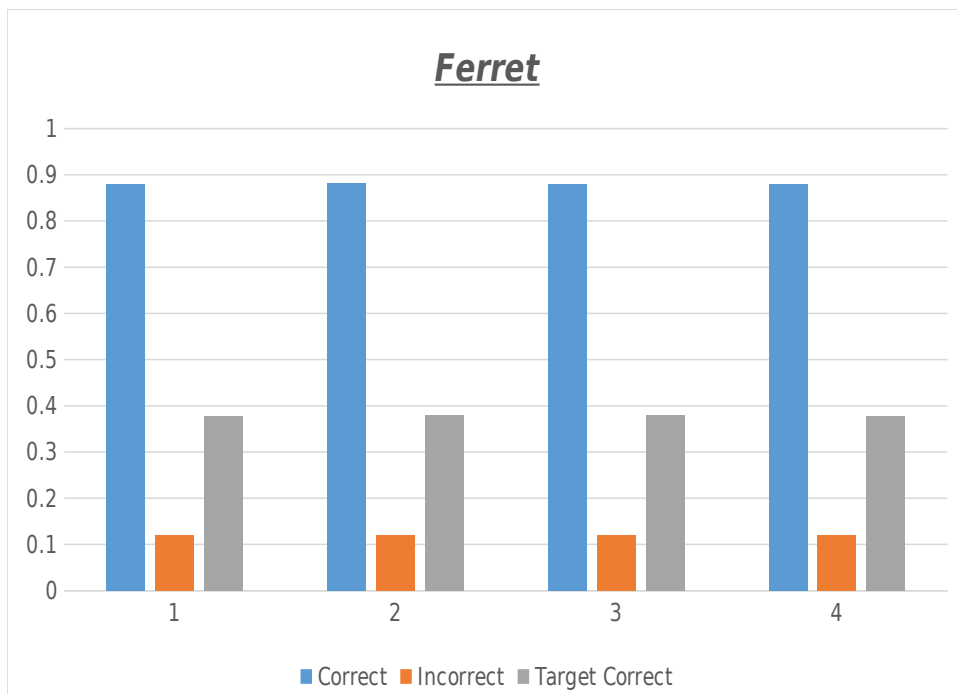
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



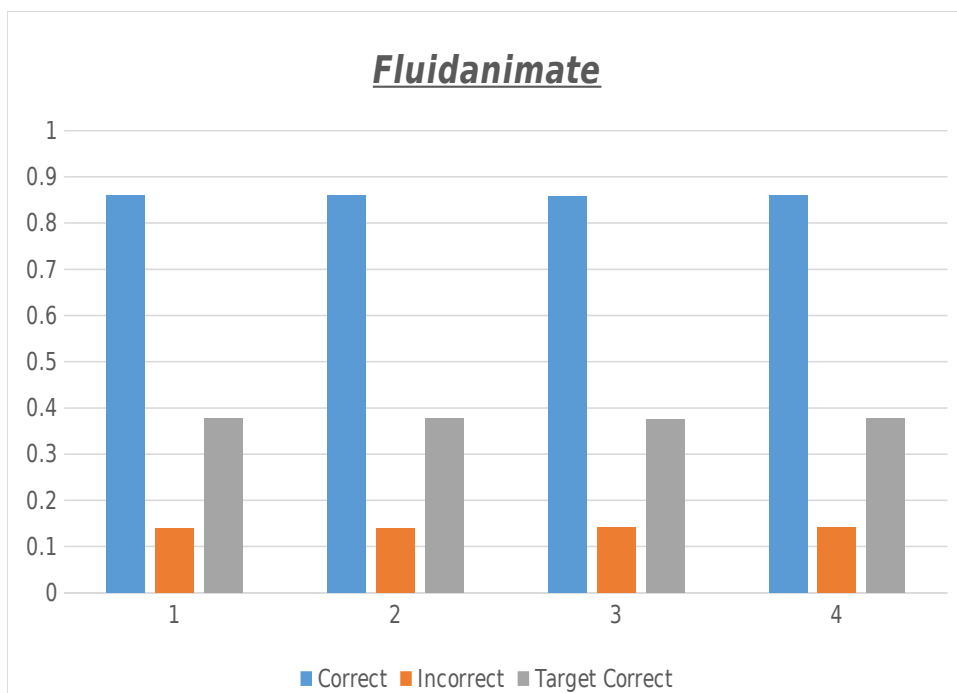
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



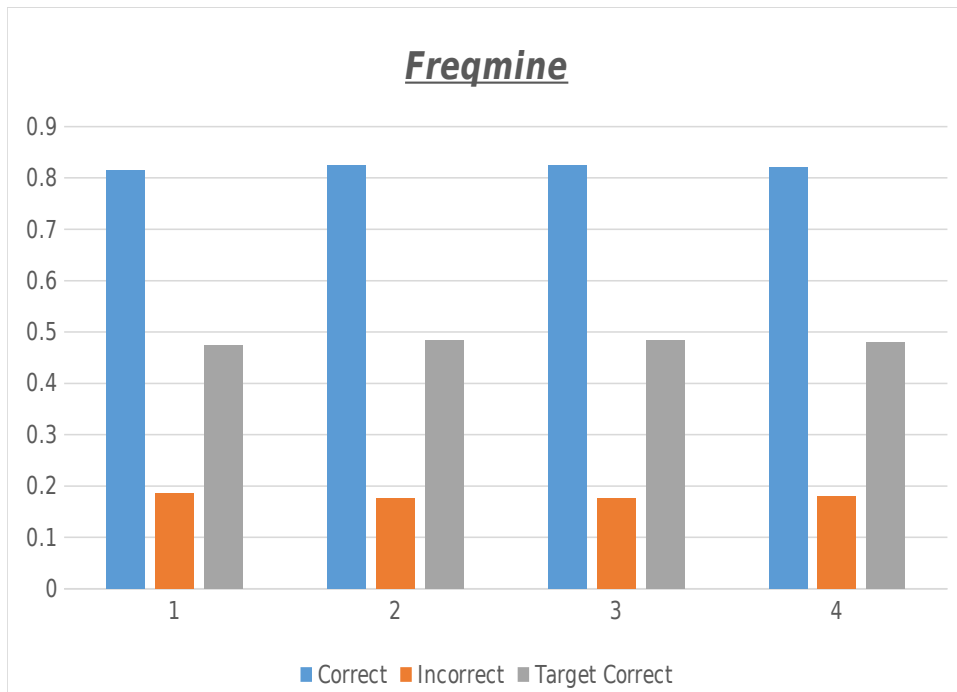
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



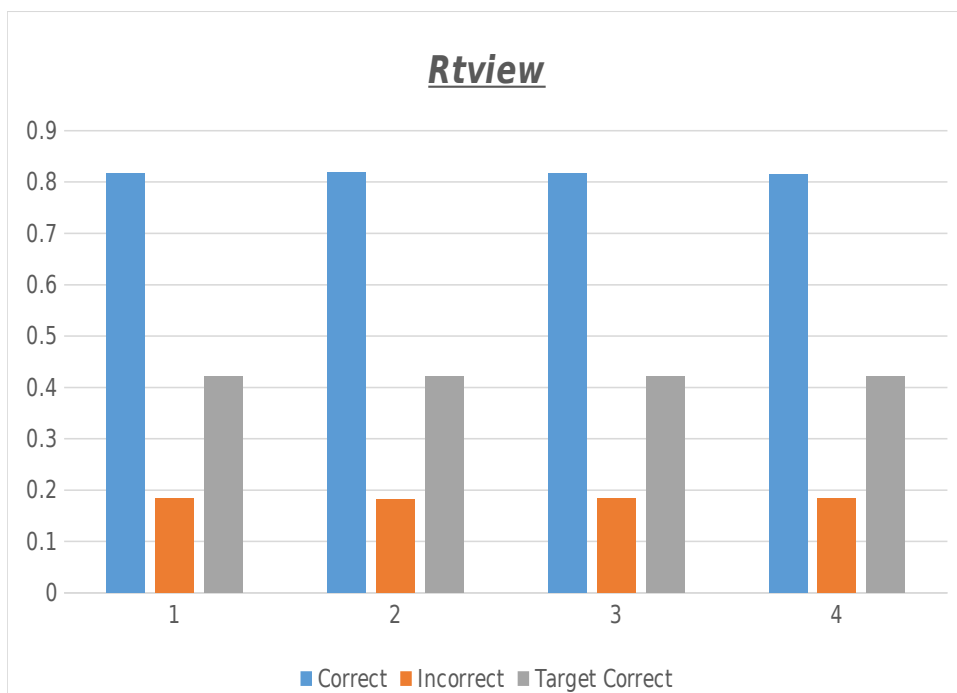
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



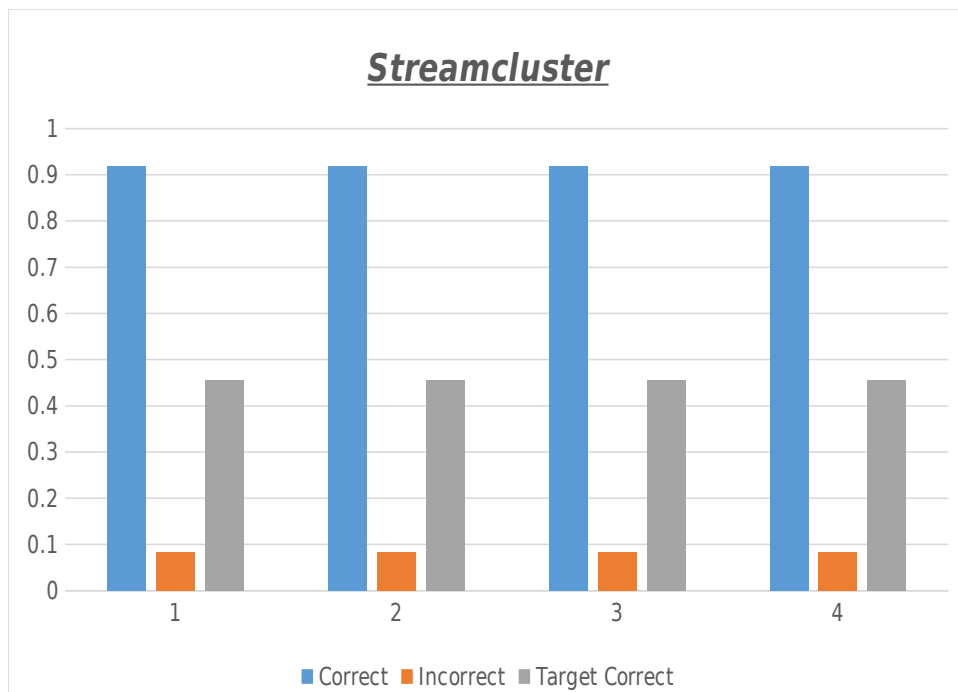
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



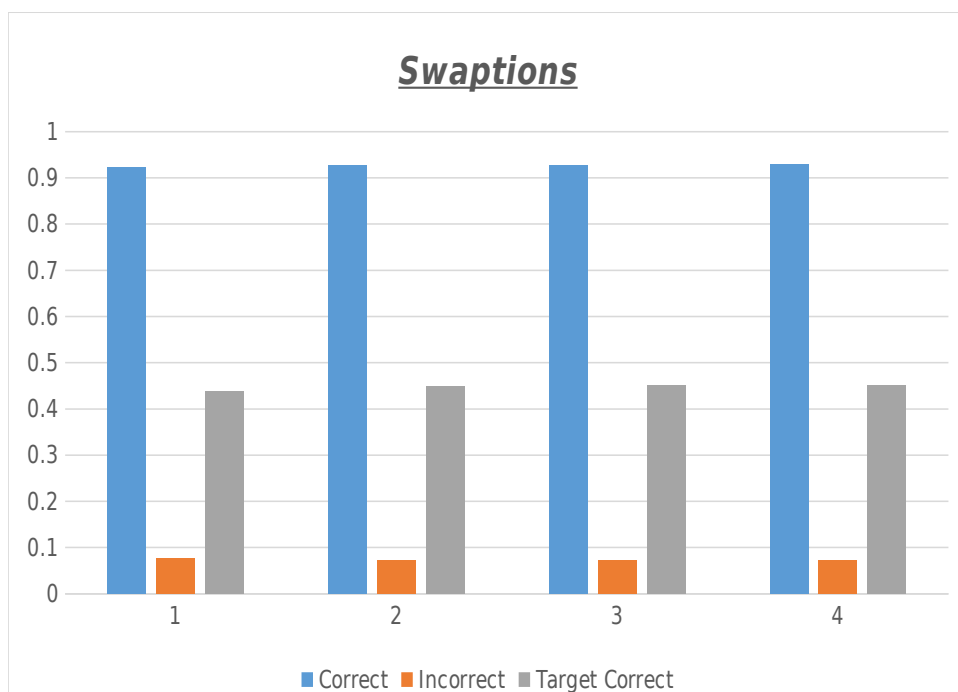
| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |



| | |
|---|-----------|
| 1 | BTB-512-1 |
| 2 | BTB-256-2 |
| 3 | BTB-128-4 |
| 4 | BTB-64-8 |

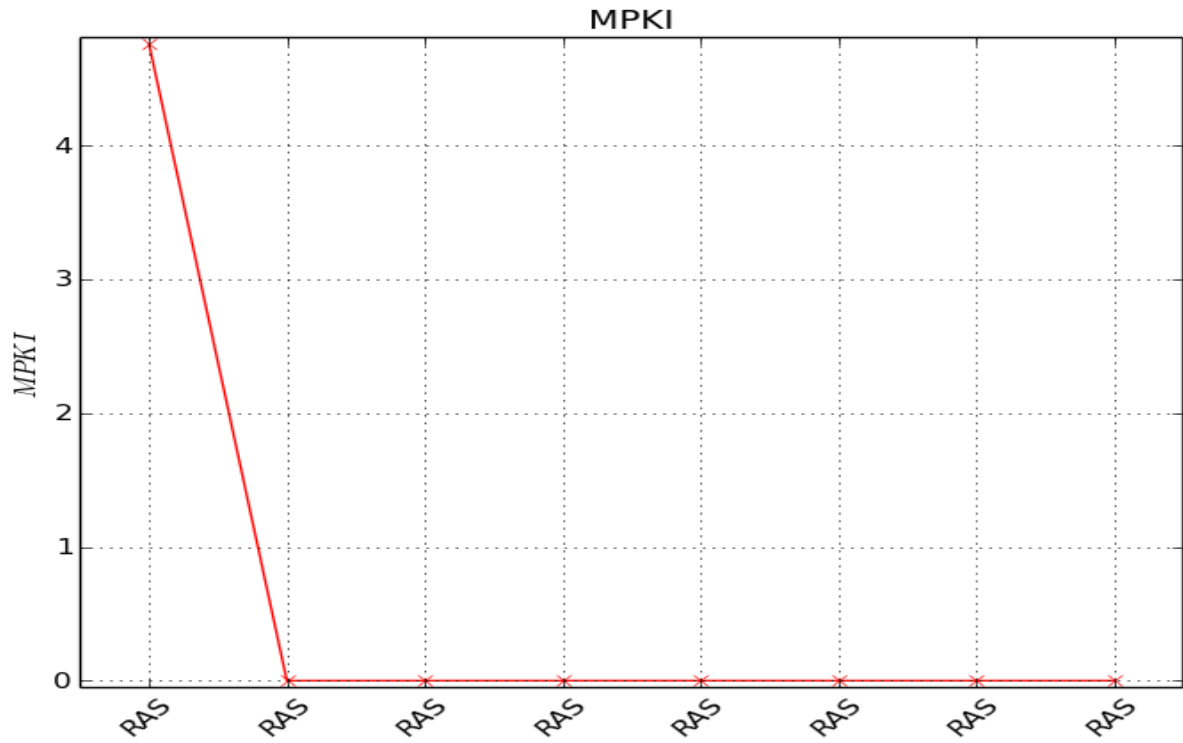
Συμπεράσματα

Συνορίζοντας καταλήγουμε στα εξής συμπεράσματα, παρατηρούμε ότι προβλέπουμε, σε αρκετά ικανοποιητικό βαθμό, την πορεία της σωστής διακλάδωσης, όμως με λανθασμένη διεύθυνση προορισμού σε αρκετές περιπτώσεις. Τέλος δεν βλέπουμε κάποια ουσιαστική διαφορά μεταξύ των τεσσάρων υλοποιήσεων. Τα παραπάνω αποτελέσματα είναι κανονικοποιημένα.

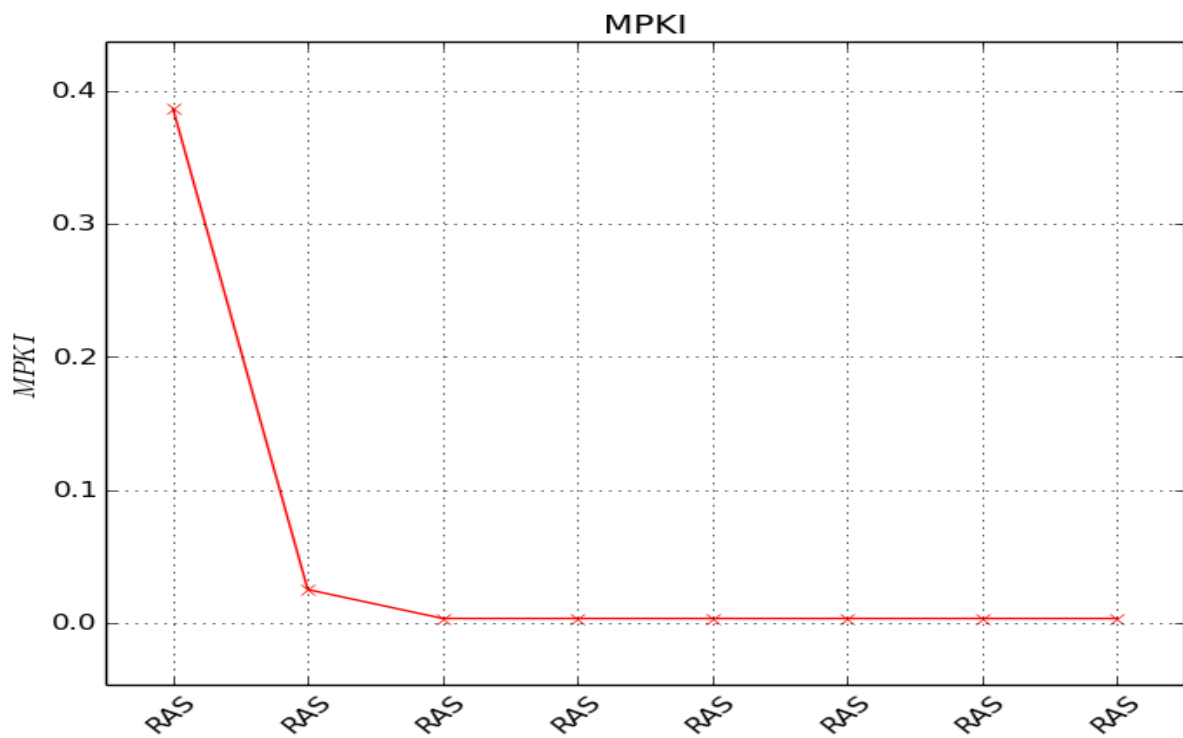
➤ Μελέτη του RAS

Εδώ θέλουμε αν μελετήσουμε την συμπεριφορά των εντολών επιστροφής (return). Δεδομένου ότι γνωρίζουμε την διεύθυνση επιστροφής, είναι ‘πάντα’ μια εντολή μετά από αυτήν που κάλεσε την συνάρτηση, είναι αρκετά εύκολο να προλέξουμε την διεύθυνση επιστροφής της.

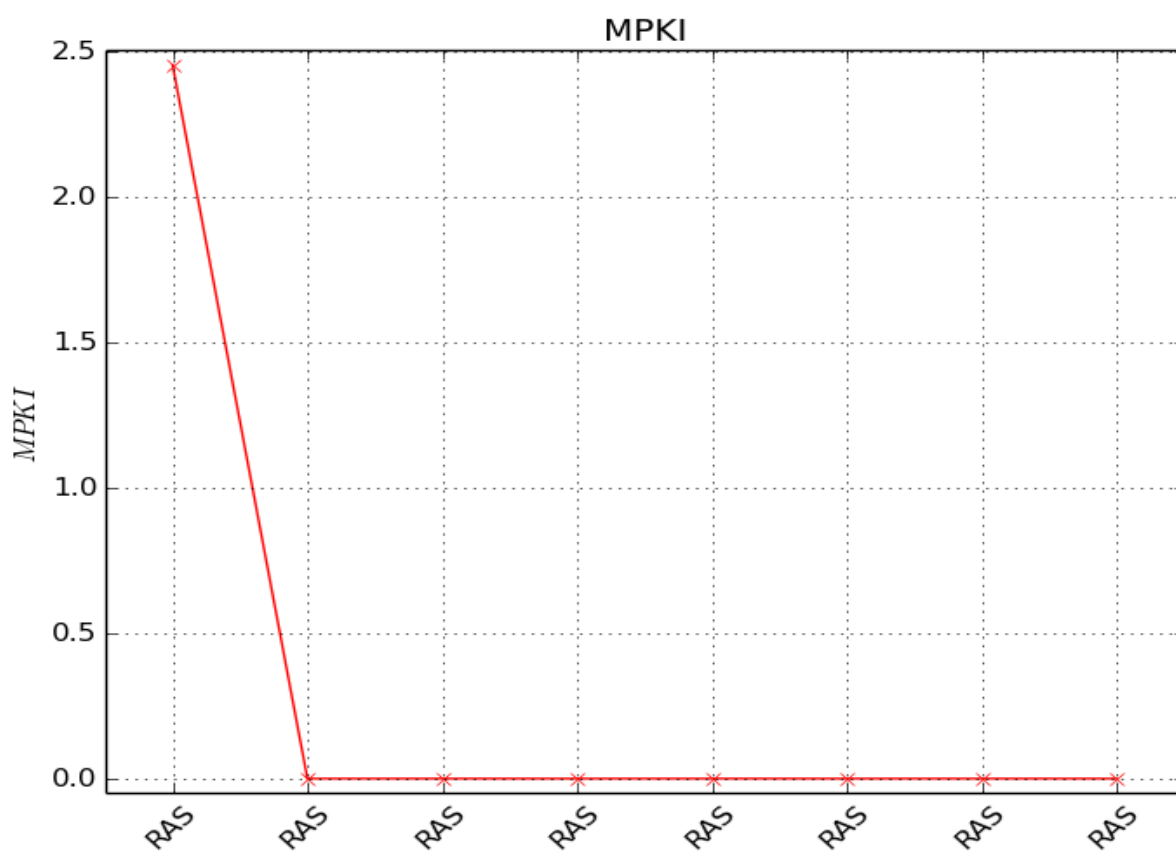
Blacksholes



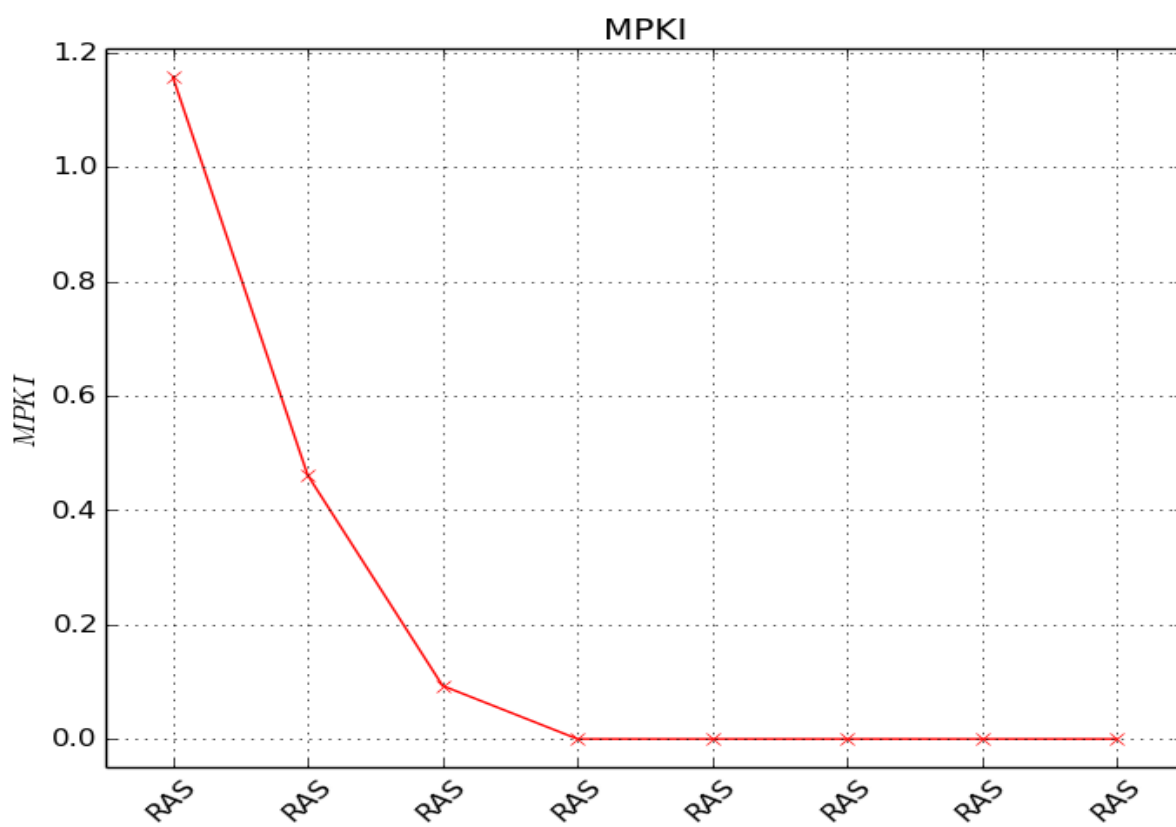
Bodytrack



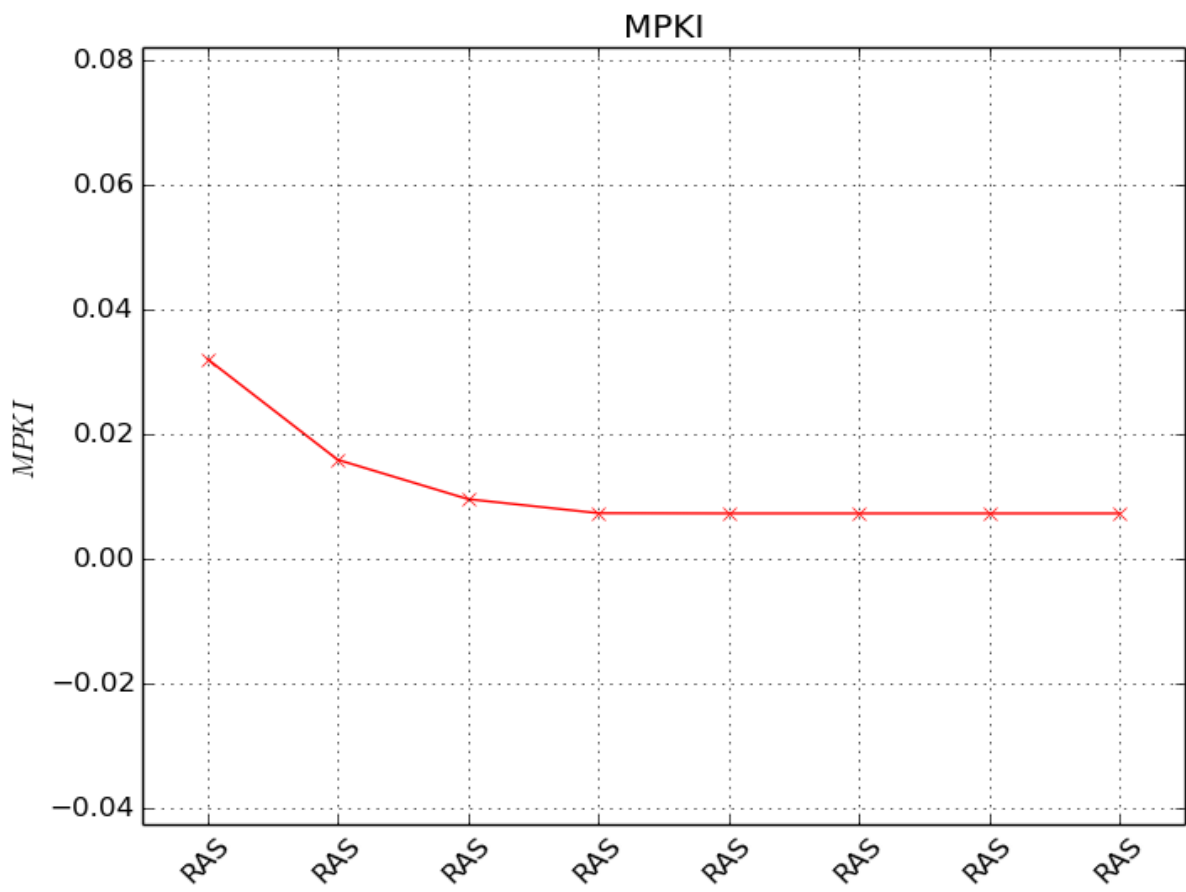
Canneal



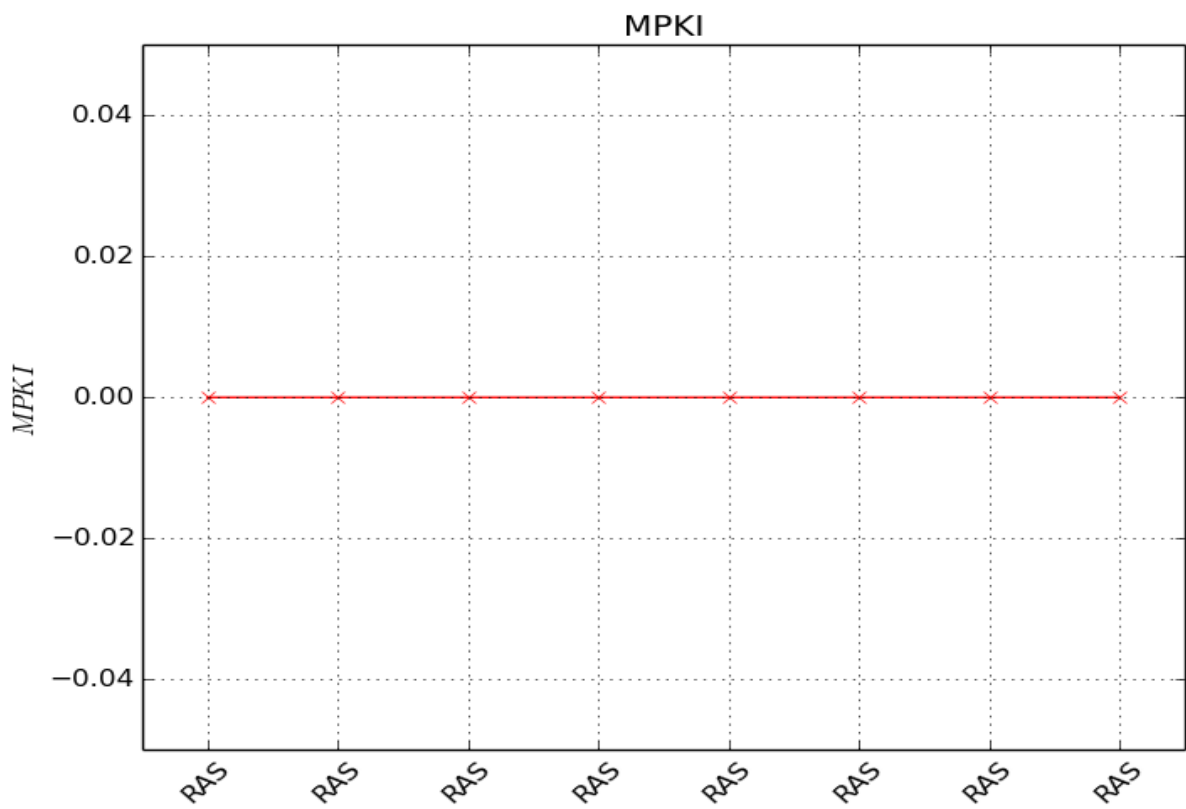
Facesim



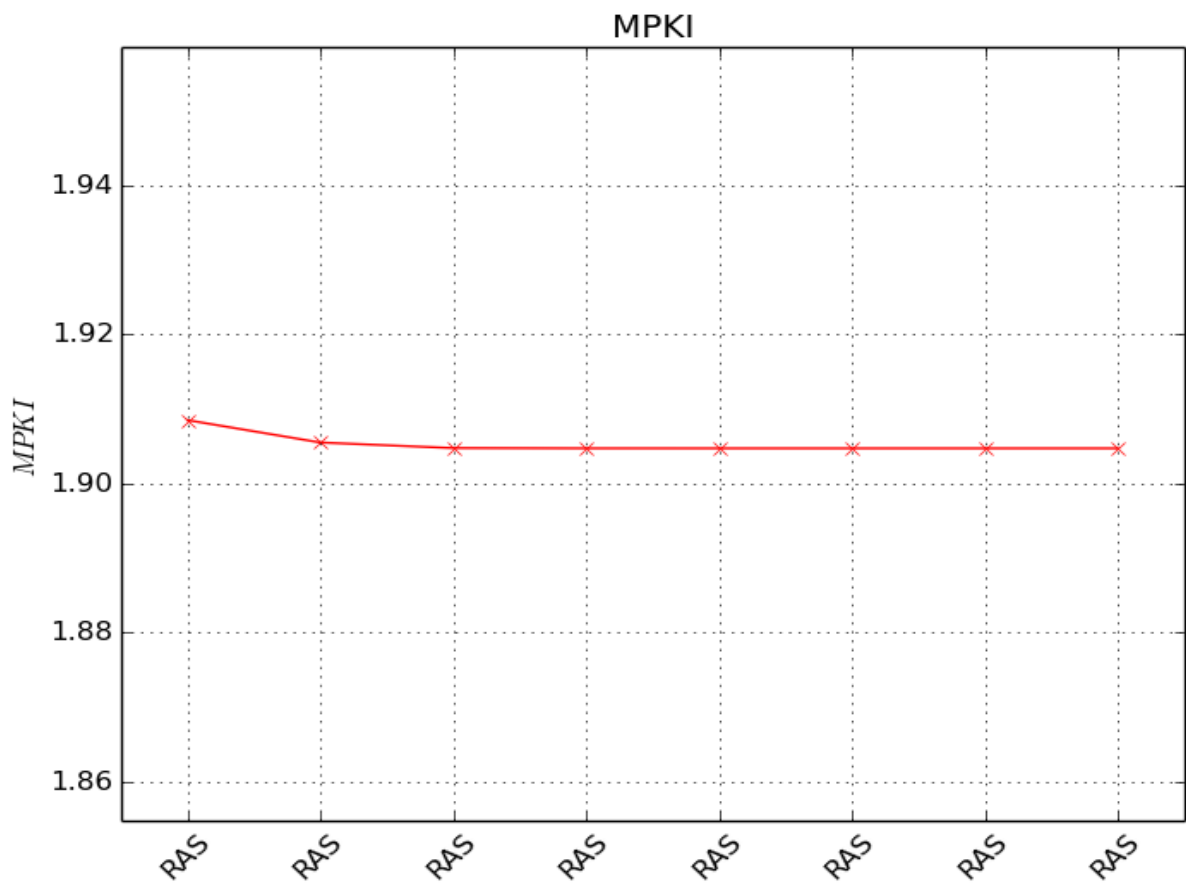
Ferret



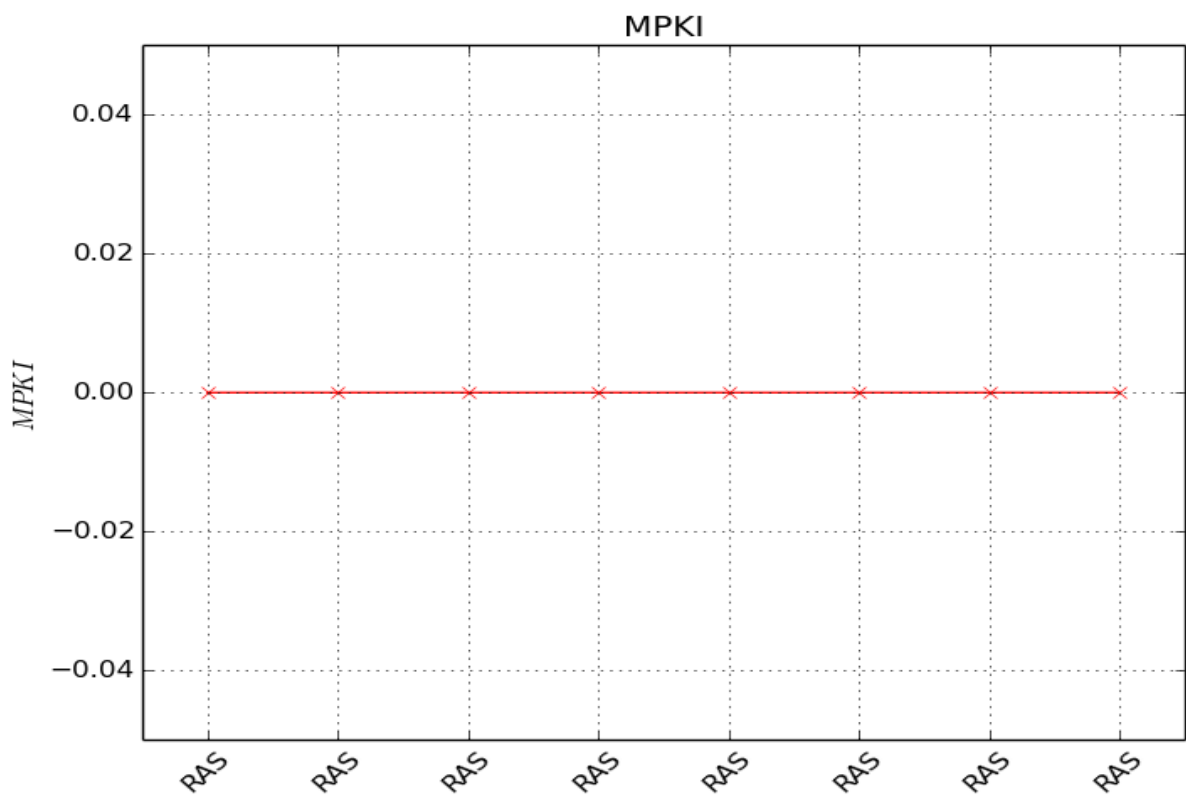
Fluidanimate



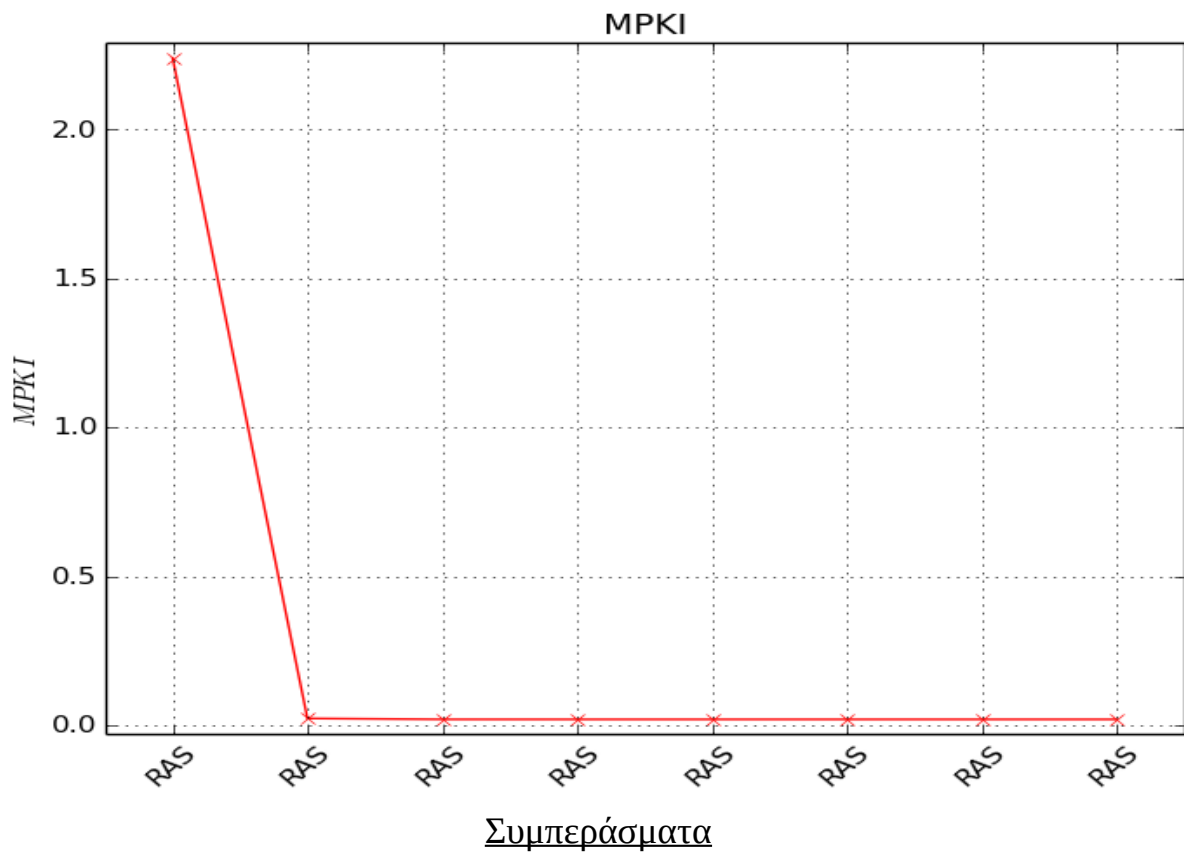
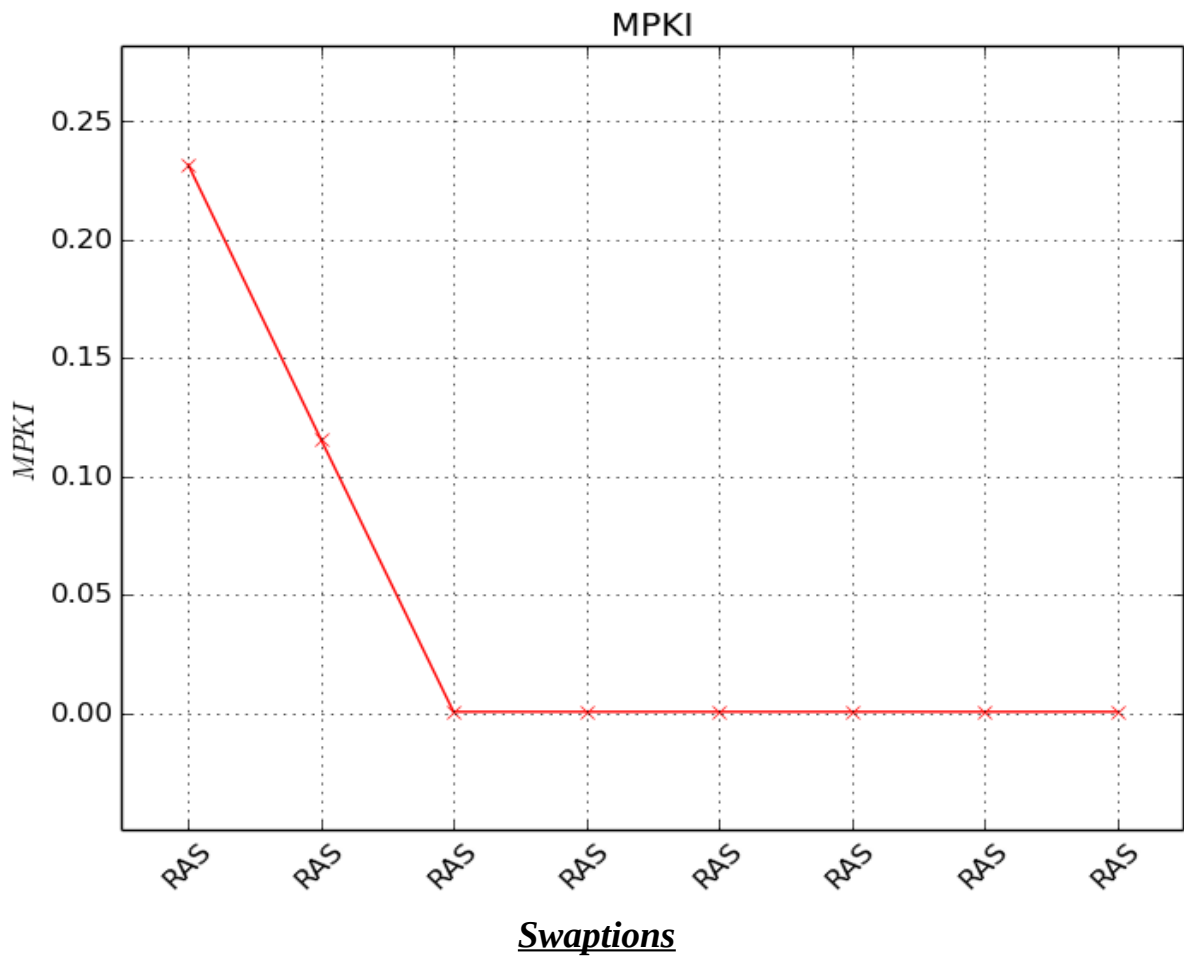
Freqmine



Rtview



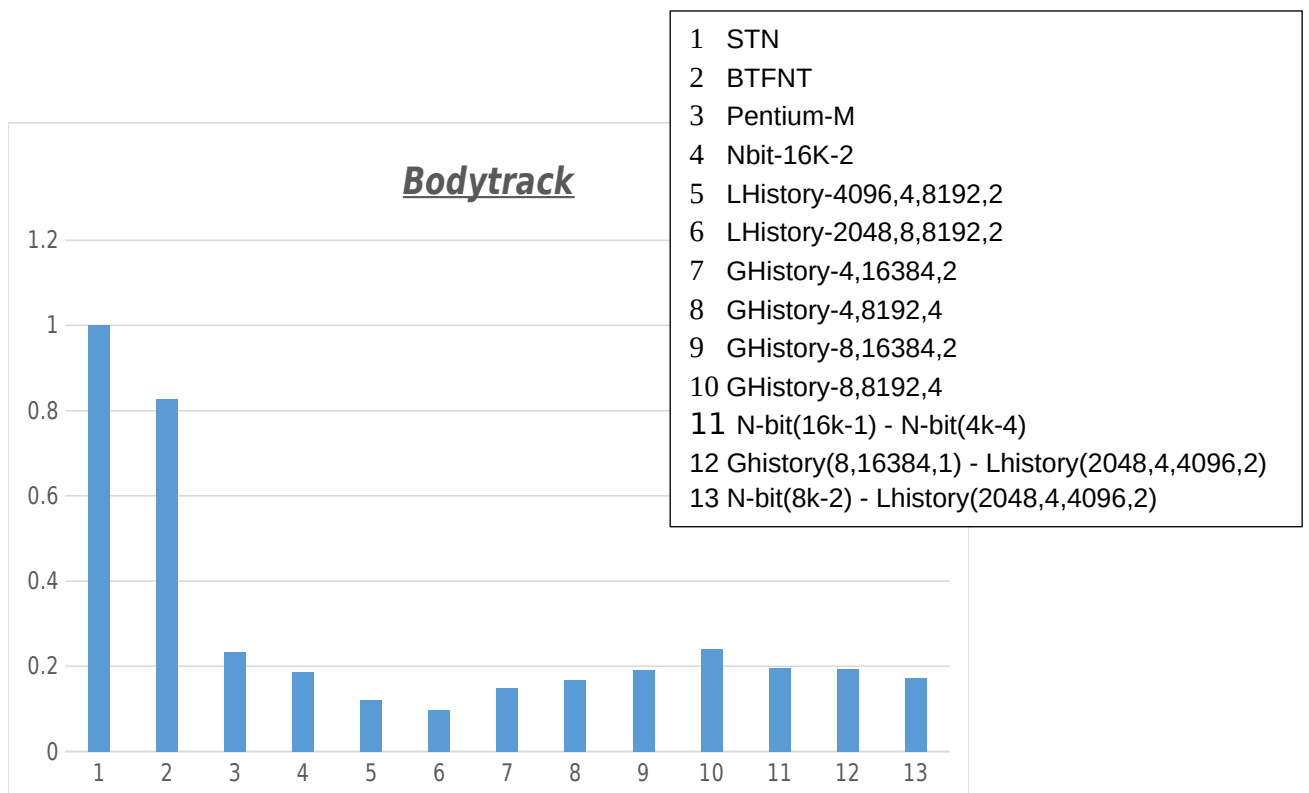
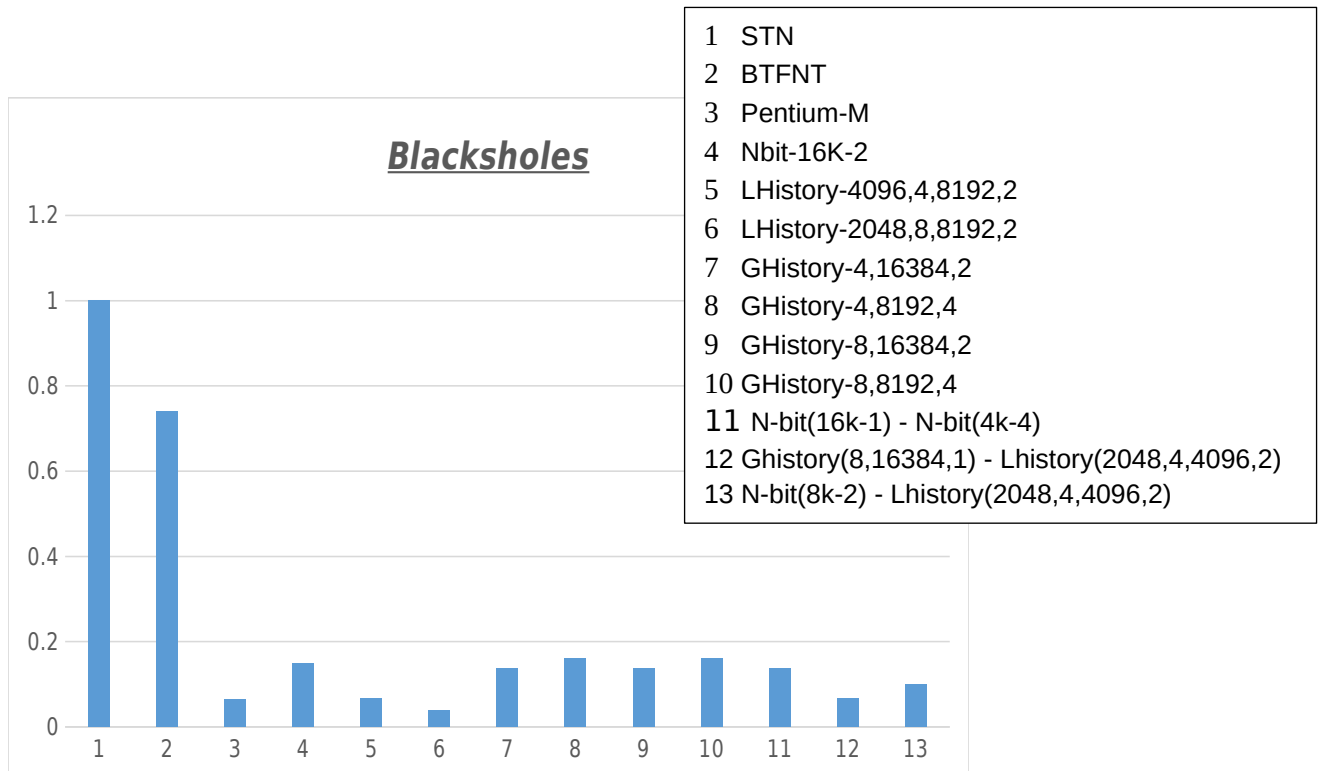
Streamcluster

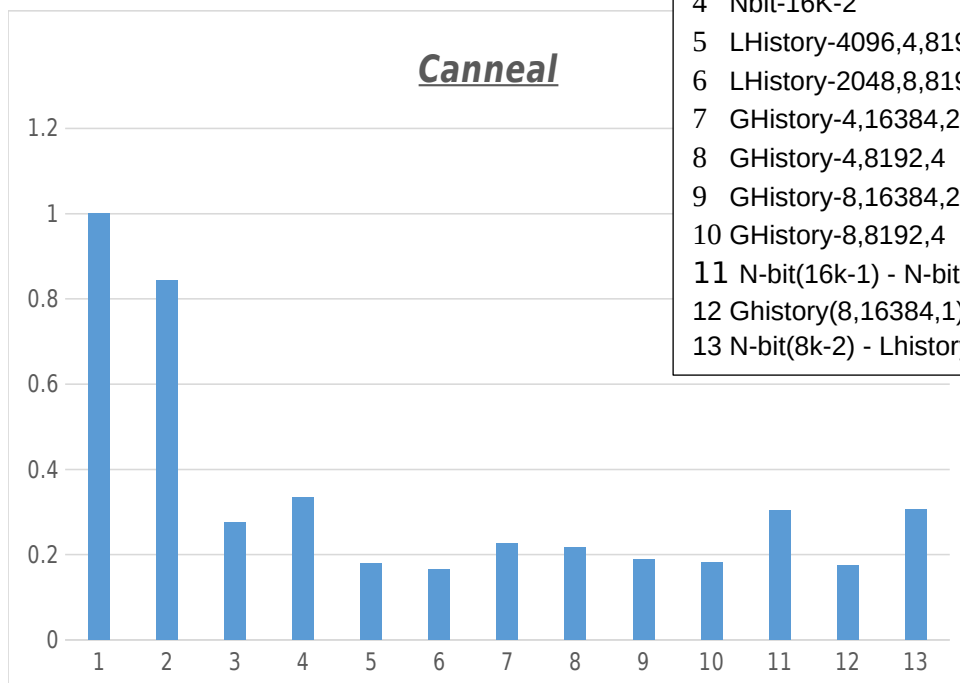


Συνορίζοντας καταλήγουμε στα εξής συμπεράσματα, αρχικά παρατηρούμε ότι με την αύξηση του σε καμία περίπτωση δεν μειώνετε η απόδοση των προγραμμάτων μας. Όταν αυξάνουμε τις εγγραφές $1 \rightarrow 2$ βλέπουμε ότι έχουν μια ραγδαία αύξηση της απόδοσης, όμως υπάρχουν και χαρακτηριστικές περιπτώσεις που δεν αλλάζουν καθόλου με την αύξηση. Σε εγγραφές μεγαλύτερες από 8 δεν βλέπουμε σχεδόν πουθενά αύξηση στην απόδοση των προγραμμάτων μας. Έτσι μπορούμε να συμπεράνουμε πως η καλύτερες τιμές είναι μεταξύ 2 και 4.

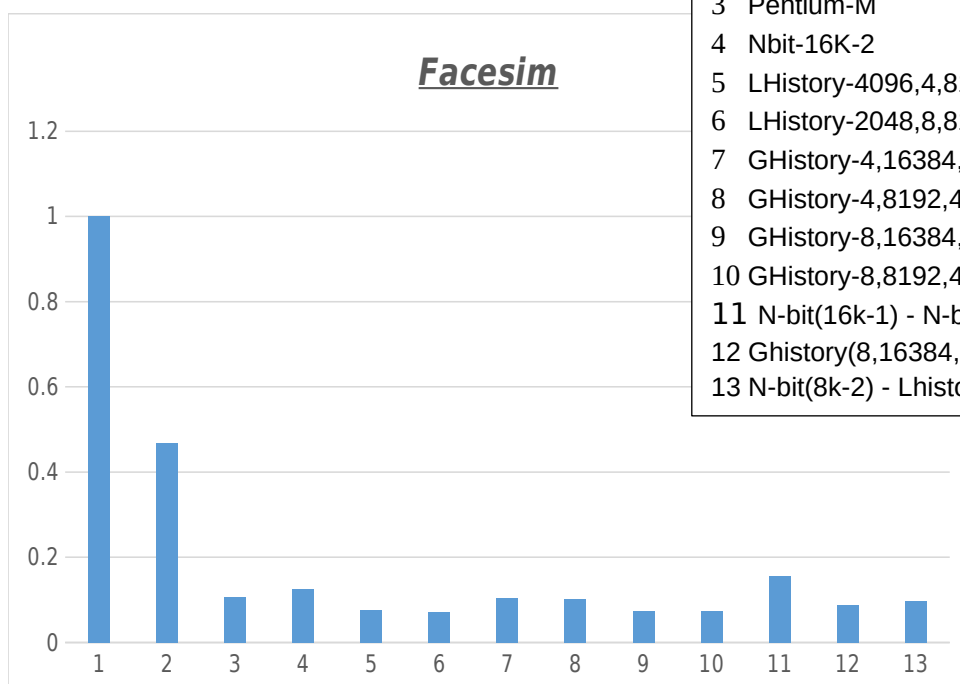
➤ Σύγκριση διαφορετικών predictors

Στο τελευταίο κομμάτι της άσκησης θέλουμε να συγκρίνουμε διάφορες τεχνικές πρόβλεψης μεταξύ τους, για να καταλείψουμε πια θα είναι καλύτερη στην υλοποίηση.

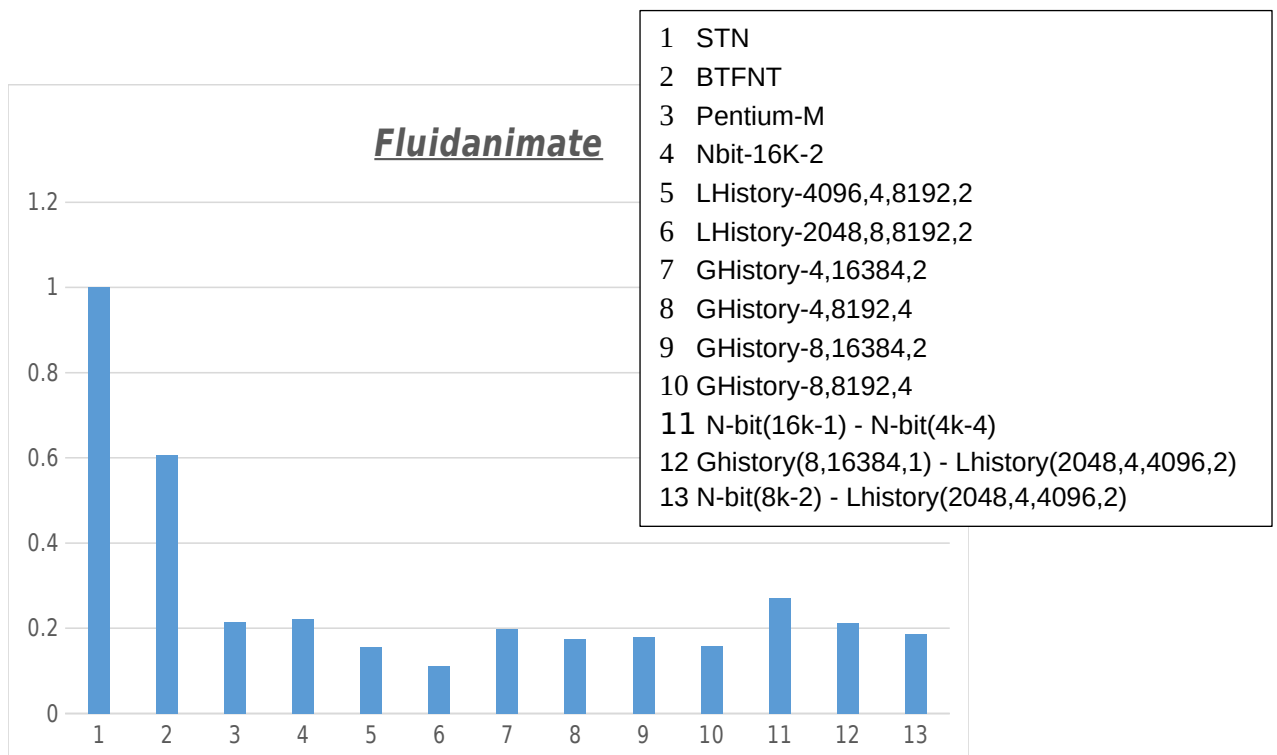
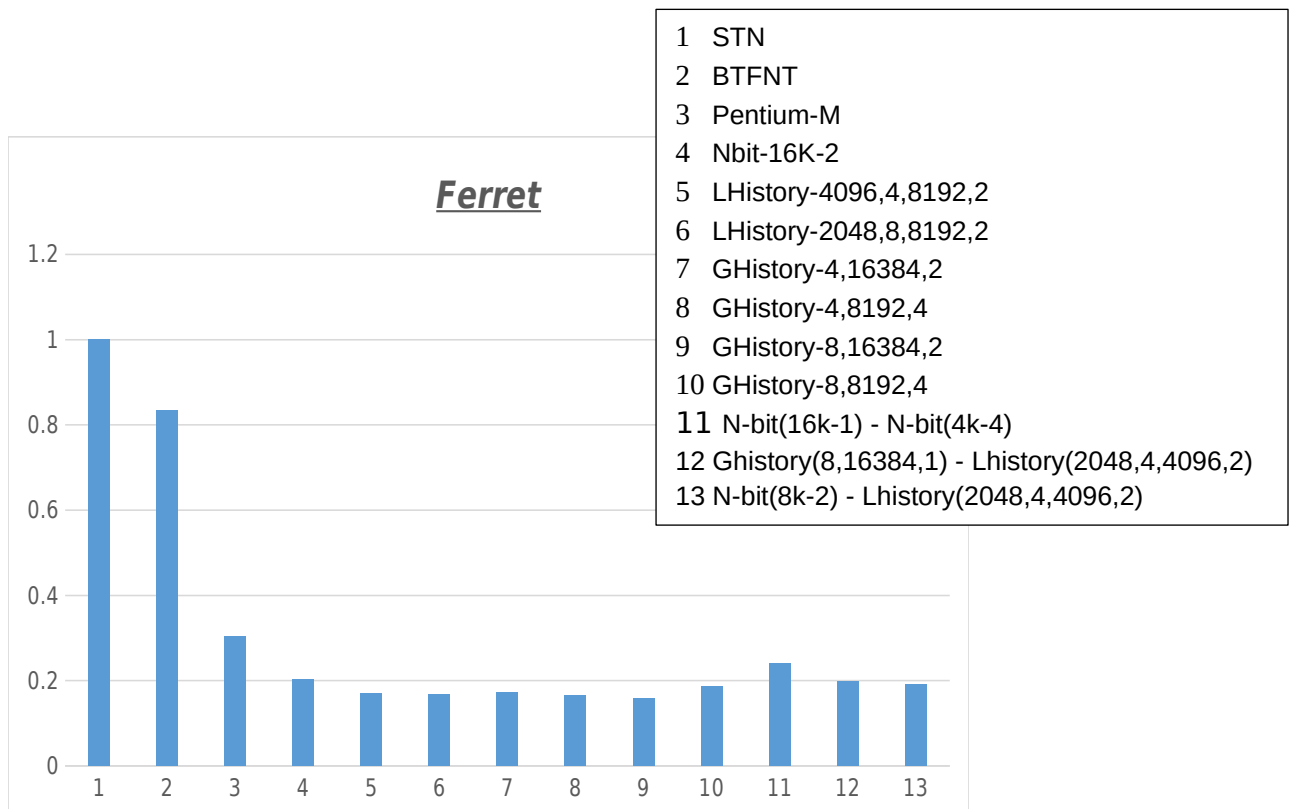


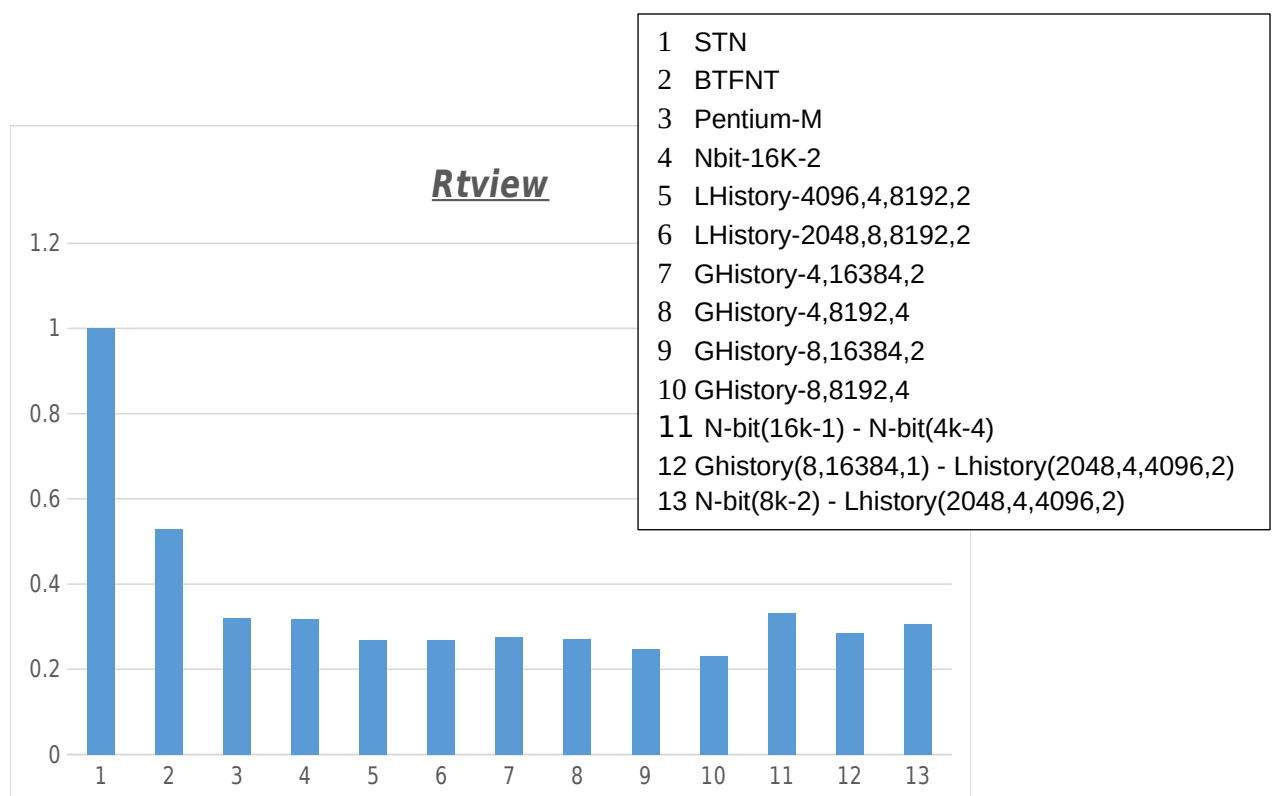
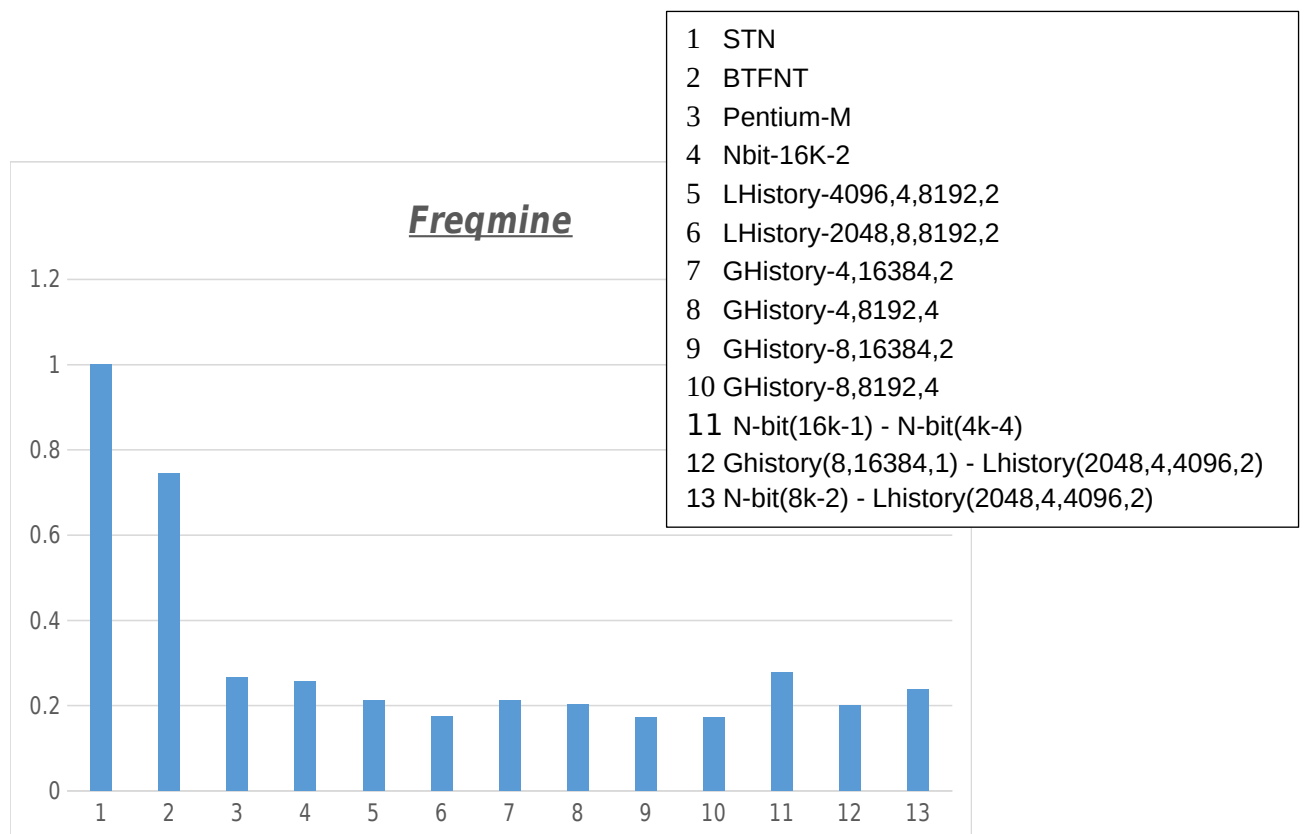


- 1 STN
- 2 BTFNT
- 3 Pentium-M
- 4 Nbit-16K-2
- 5 LHistory-4096,4,8192,2
- 6 LHistory-2048,8,8192,2
- 7 GHistory-4,16384,2
- 8 GHistory-4,8192,4
- 9 GHistory-8,16384,2
- 10 GHistory-8,8192,4
- 11 N-bit(16k-1) - N-bit(4k-4)
- 12 Ghistory(8,16384,1) - Lhistory(2048,4,4096,2)
- 13 N-bit(8k-2) - Lhistory(2048,4,4096,2)

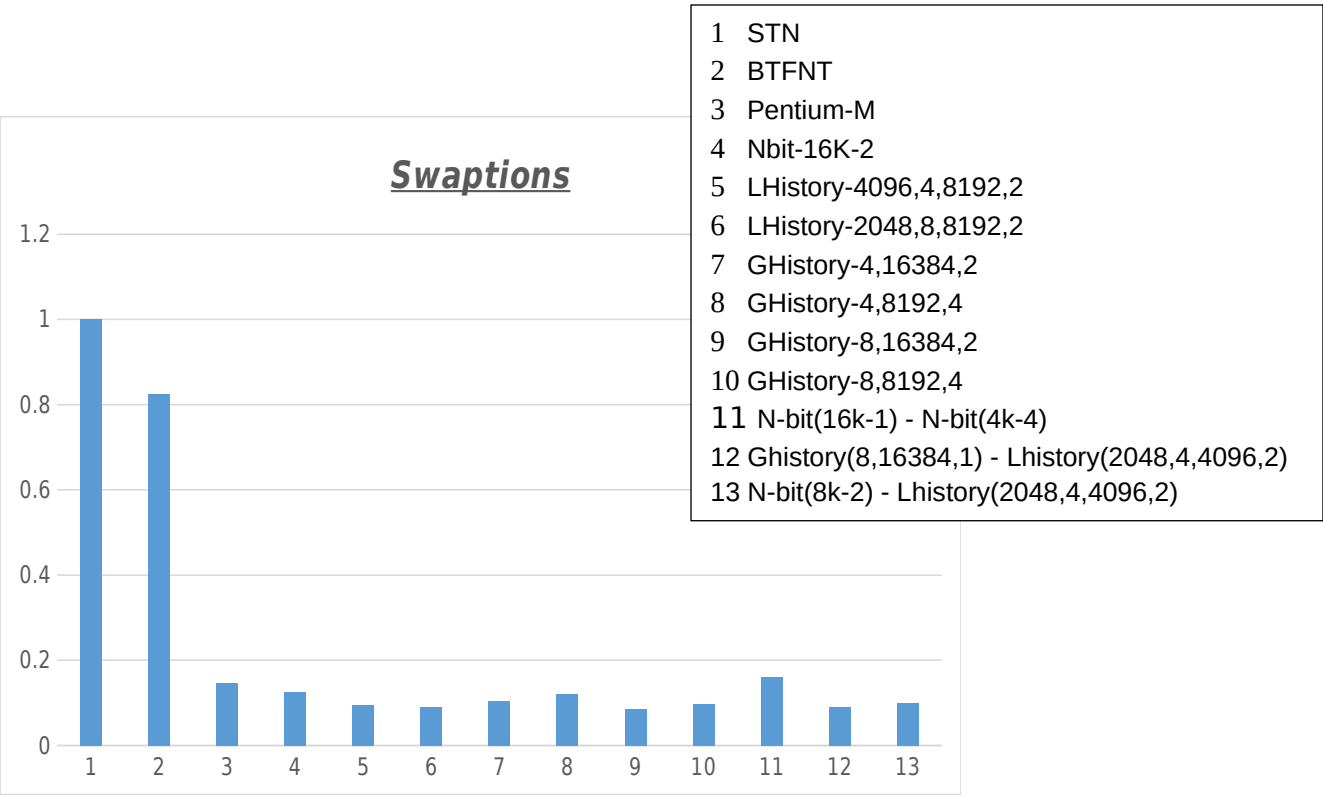
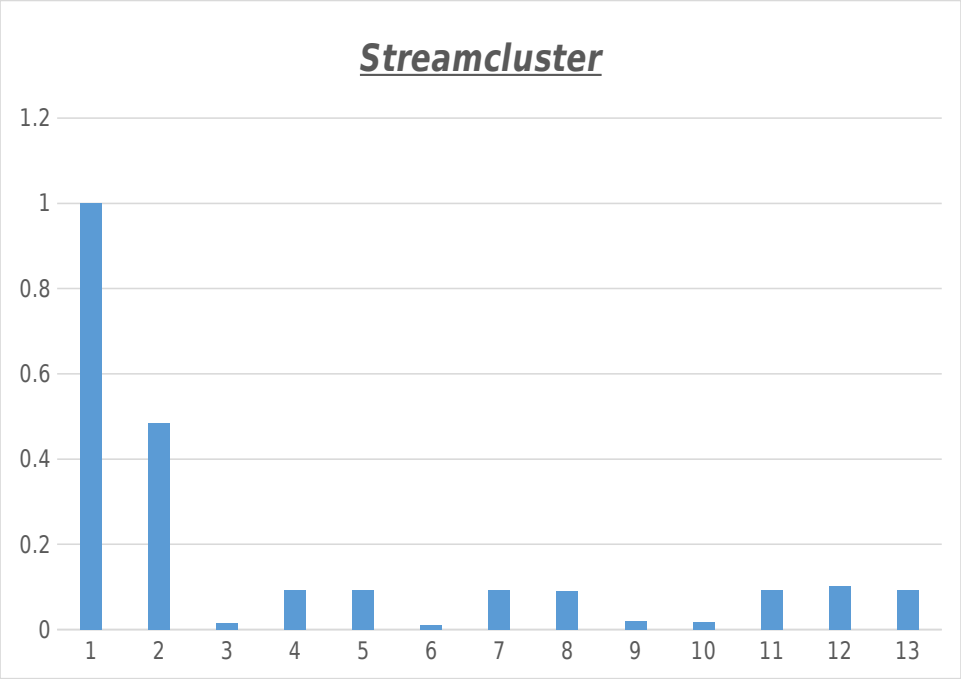
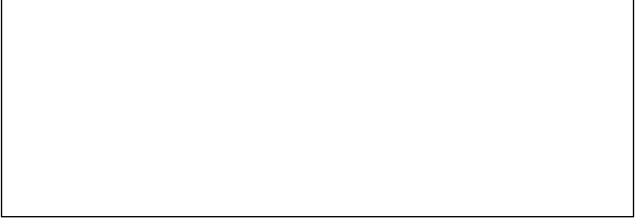


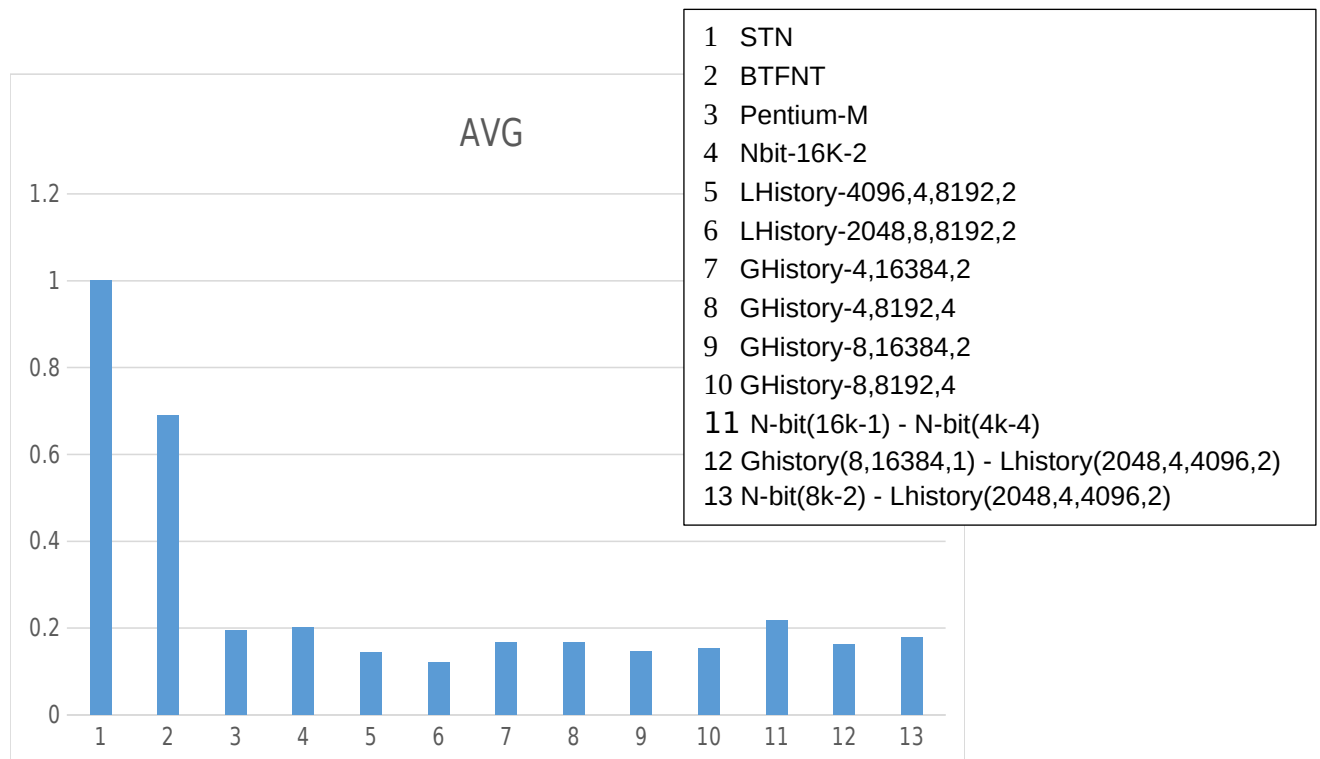
- 1 STN
- 2 BTFNT
- 3 Pentium-M
- 4 Nbit-16K-2
- 5 LHistory-4096,4,8192,2
- 6 LHistory-2048,8,8192,2
- 7 GHistory-4,16384,2
- 8 GHistory-4,8192,4
- 9 GHistory-8,16384,2
- 10 GHistory-8,8192,4
- 11 N-bit(16k-1) - N-bit(4k-4)
- 12 Ghistory(8,16384,1) - Lhistory(2048,4,4096,2)
- 13 N-bit(8k-2) - Lhistory(2048,4,4096,2)





- 1 STN
- 2 BTFNT
- 3 Pentium-M
- 4 Nbit-16K-2
- 5 LHistory-4096,4,8192,2
- 6 LHistory-2048,8,8192,2





Συμπεράσματα

Συνοψίζοντας καταλήγουμε στα εξής συμπεράσματα, αρχικά παρατηρούμε ότι την χειρότερη επίδοση την δίνει η STN και στην συνέχεια η BTFNT. Όλοι οι υπόλοιποι predictors δεν παρουσιάζουν σημαντικές διαφορές όπως φαίνεται και από το συνοπτικό διάγραμμα μέσης τιμής στο τέλος. Την βέλτιστη επίδοση έχουμε στο LHistory-2K-8-8K-2. Τα παραπάνω αποτελέσματα είναι κανονικοποιημένα.

Υ.Γ.: Τα ορθογραφικά και συντακτικά λάθη που μπορεί να έχω κάνει οφείλονται στην δυσλεξία που έχω. Επίσης, στον ίδιο οφείλονται και κάποια εκφραστικά λάθη που μπορεί να έχω κάνει διότι μου είναι αρκετά δύσκολα να εκφράσω την σκέψη μου μέσω του γραπτού λόγου. Σε περίπτωση που υπάρχει κάποια ασάφεια ή δυσκολία κατανόησης παρακαλώ να επικοινωνήσετε μαζί μου στο email: zoroxan@gmail.com.