



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

---

Άσκηση 1 - Προηγμένα Θέματα  
Αρχιτεκτονικής Υπολογιστών

---

Γρηγόριος Θανάσουλας

gregthanasoulas@gmail.com

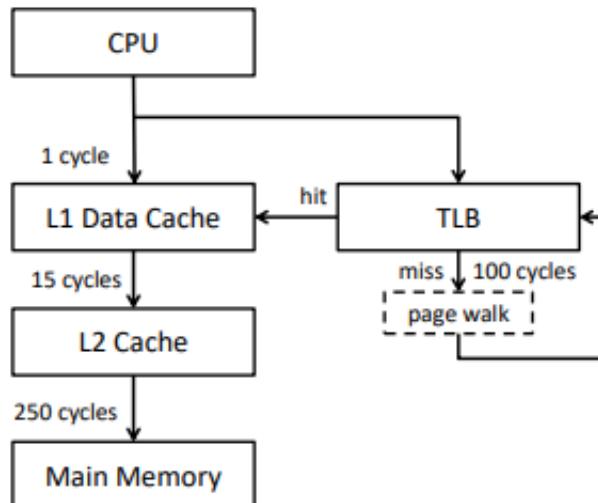
A.M: 03114131

14 Απριλίου 2020

# 1 Σκοπός

Στην άσκηση αυτή γίνεται μελέτη της επίδρασης διαφόρων παραμέτρων της ιεραρχίας μνήμης στην απόδοση ενός συνόλου εφαρμογών. Για το σκοπό αυτό διεξάγεται μία σειρά μετρήσεων, με τη χρήση του εργαλείου PIN, το οποίο εισάγει δυναμικά κώδικα ανάμεσα στις εντολές της εφαρμογής προκειμένου να συλλέξει πληροφορίες σχετικά με την εκτέλεση τους. Στη μελέτη της απόδοσης της μνήμης, χρησιμοποιούνται τα ακόλουθα 10 μετροπρογράμματα: blackscholes, bodytrack, canneal, facesim, ferret, fluidanimate, freqmine, rtview, swaptions, streamcluster.

Η ιεραρχία μνήμης που εξετάζεται είναι η παρακάτω:



# 2 Πειραματική Αξιολόγηση

Σαν βασική μετρική επίδοσης θα χρησιμοποιηθεί το IPC (Instructions Per Cycle). Θα μελετήσουμε την απόδοση σε δύο διαφορετικές περιπτώσες, θεωρώντας στην πρώτη περίπτωση ότι ο κύκλος του ρολογιού παραμένει σταθερός ανεξαρτήτως των παραμέτρων της μνήμης, ενώ στη δεύτερη περίπτωση ο κύκλος μεταβάλλεται καθώς αλλάζουν τα χαρακτηριστικά της μνήμης.

## 2.1 Μελέτη επίδρασης παραμέτρων ιεραρχίας μνήμης στην απόδοση της εφαρμογής θεωρώντας σταθερό κύκλο ρολογιού

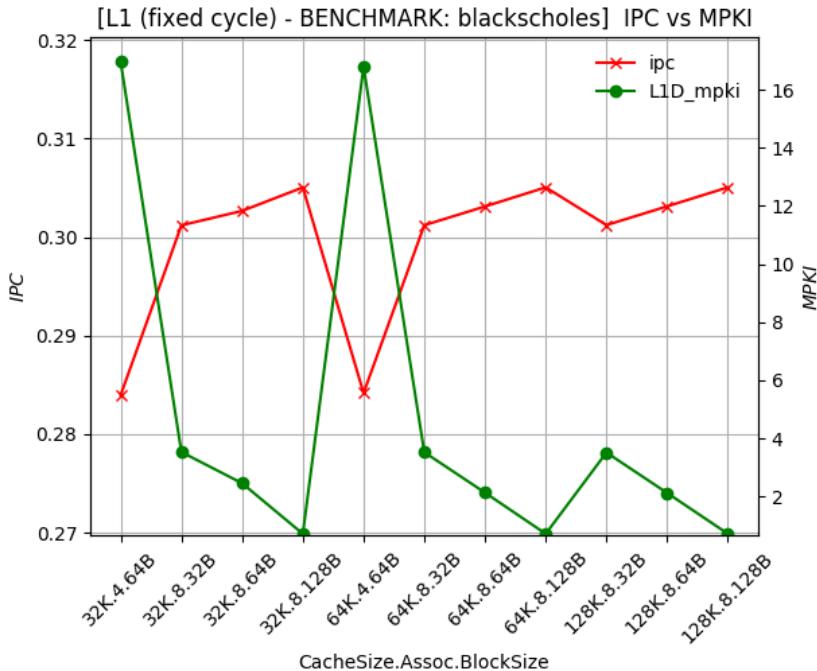
### 2.1.1 L1 Cache

Οι παράμετροι της L2 cache και του TLB θα διατηρηθούν σταθερές και συγκεκριμένα ίσες με:

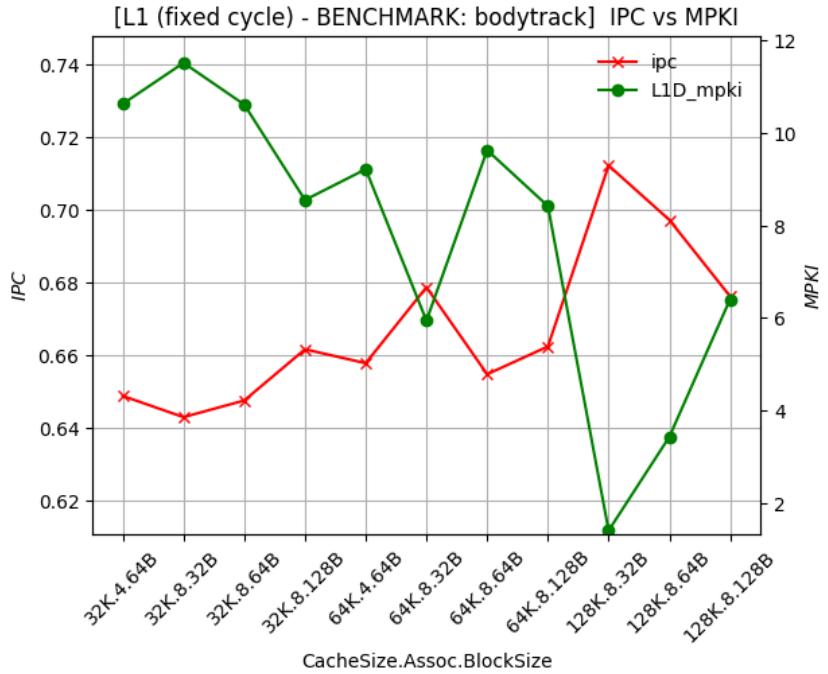
- L2 size = 1024 KB
- L2 associativity = 8
- L2 block size = 128 B
- TLB size = 64 entr.
- TLB associativity = 4
- TLB page size = 4096 B

Ακολουθούν τα διαγράμματα που προέκυψαν και ο σχετικός σχολιασμός τους:

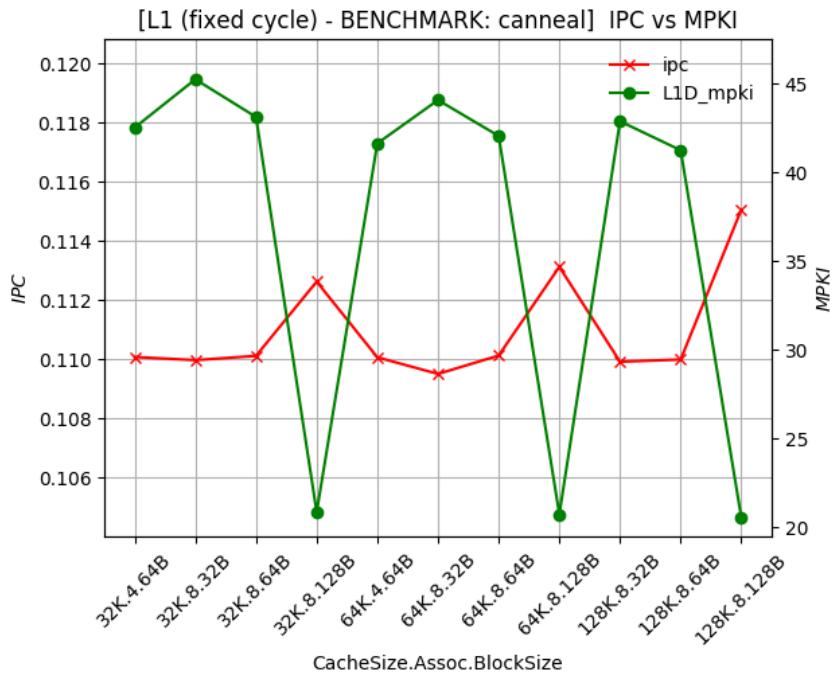
*blackscholes*



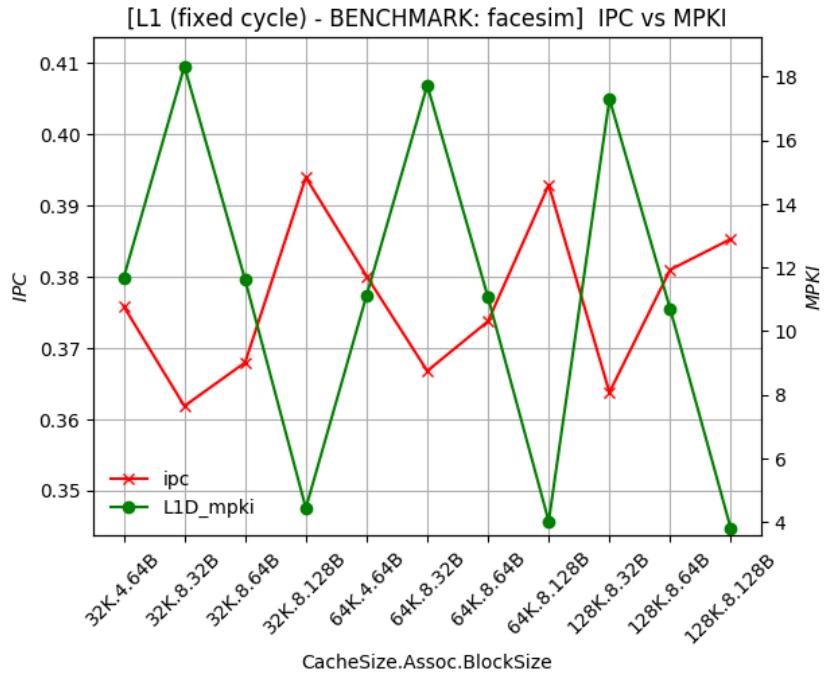
### *bodytrack*



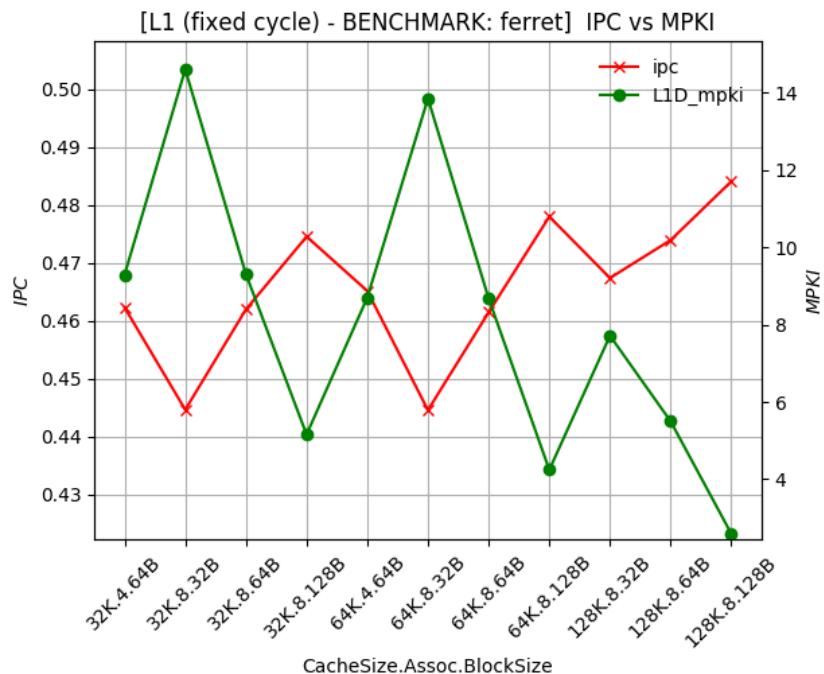
### *canneal*



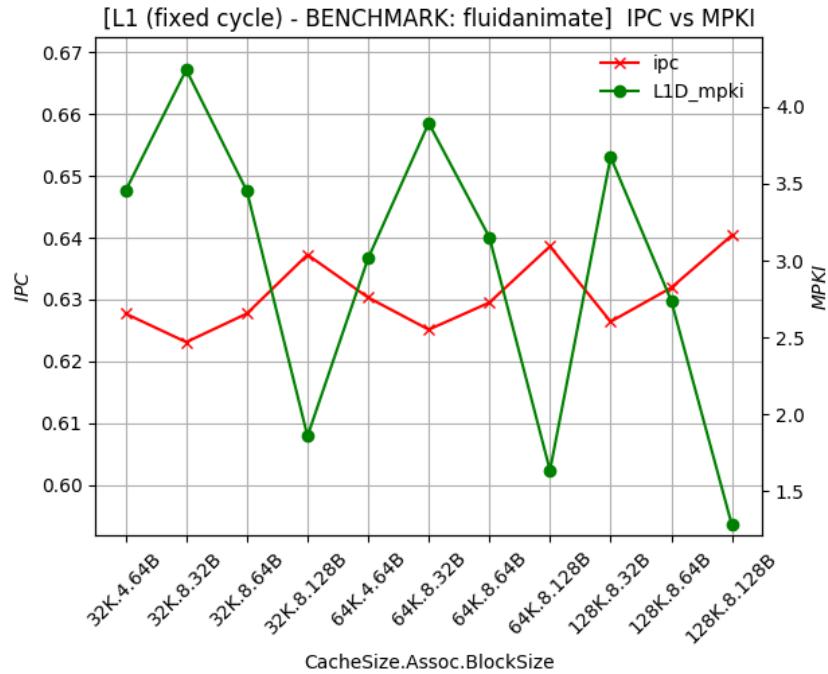
## *facesim*



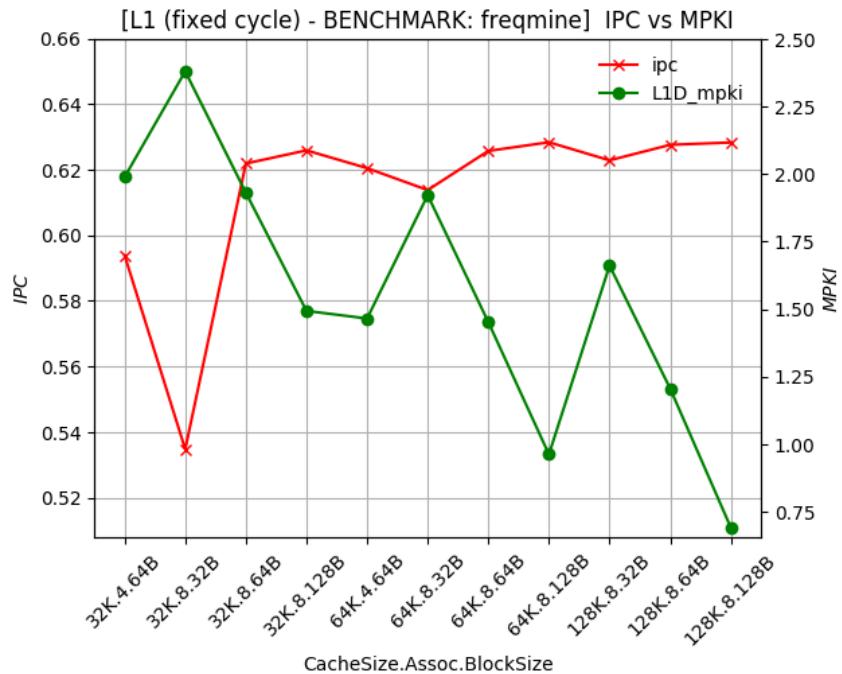
## *ferret*



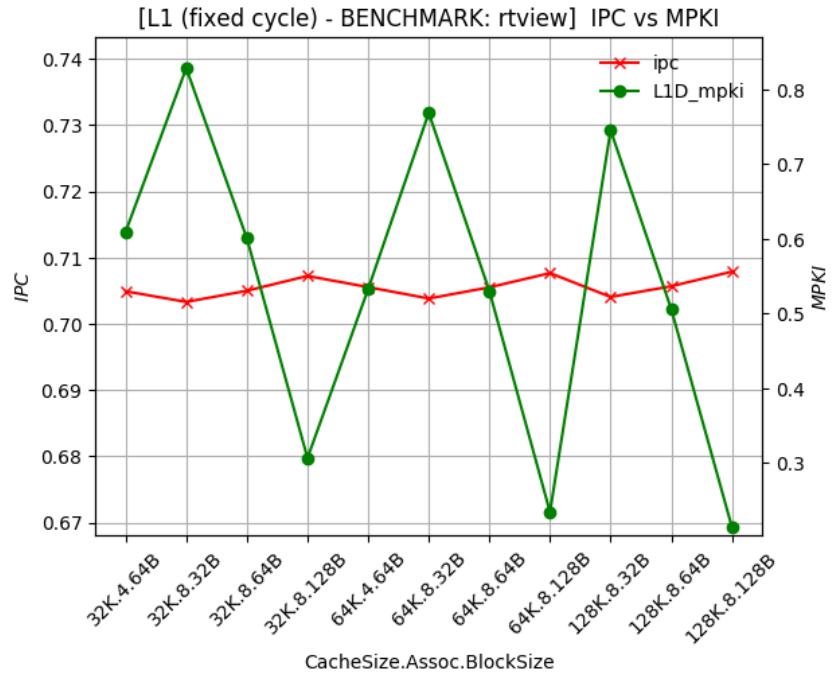
## *fluidanimate*



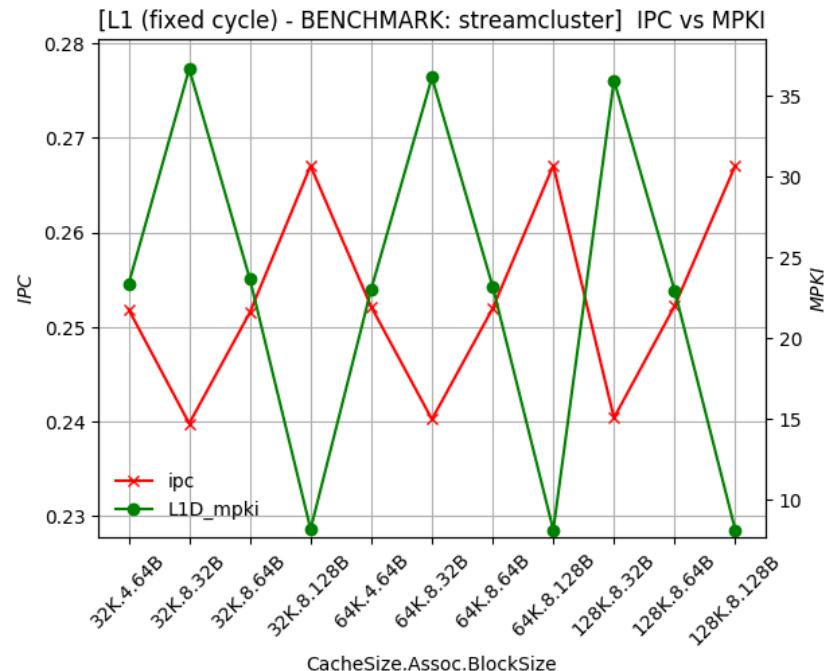
## *freqmine*



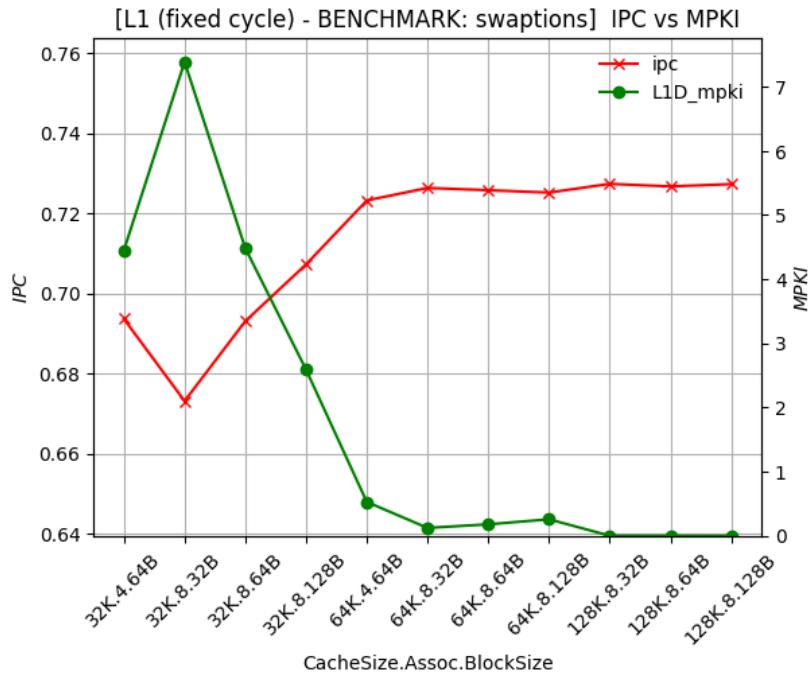
*rtview*



*streamcluster*

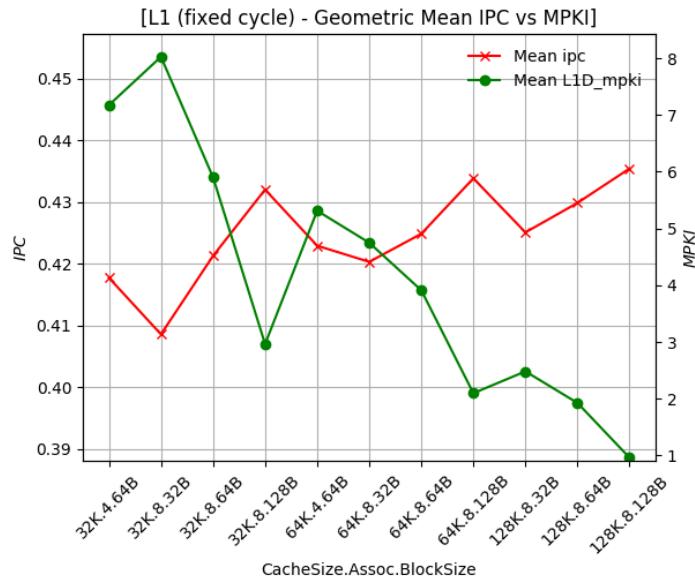


### *swaptions*



Επίσης, για λόγους συνολικής εποπτείας παρατίθεται ένα διάγραμμα των γεωμετρικών μέσων των προηγούμενων:

### *Geometric Average*



## Συμπεράσματα για την L1 Cache

Παρατηρούμε πως τα μεγέθη Instructions Per Cycle και Misses Per KiloInstructions μεταβάλλονται αντιστρόφως ανάλογα. Αυτό σαφώς είναι αναμενόμενο, αφού όσο περισσότερες αστοχίες υπάρχουν (μεγαλύτερο MPKI), τόσο περισσότεροι κύκλοι χρειάζονται για να εκτελεστεί η εντολή και άρα η επίδοση θα είναι χειρότερη, δηλαδή το IPC θα είναι μικρότερο, και αντίστροφα.

1. Παρατηρούμε πως σχεδόν σε όλα τα benchmarks με εξαίρεση το swaptions, η μεταβολή του μεγέθους της cache δεν επηρεάζει αρκετά το IPC. Αυτό θα μπορούσε να ερμηνευθεί από το ότι ενδεχομένως στα benchmarks αυτά, τα misses οφείλονται κυρίως σε compulsory misses και όχι σε capacity misses.
2. Στο benchmark blackscholes, παρατηρούμε πως η αύξηση του associativity αύξησε το IPC και μείωσε το MPKI (μειώθηκαν τα conflict misses).
3. Στα benchmarks bodytrack, canneal, facesim, ferret, fluidanimate, streamcluster, swaptions (σε άλλα περισσότερο και σε άλλα λιγότερο, σε άλλα κυρίως για μικρότερες τιμές μεγέθους της cache), παρατηρούμε ότι η αύξηση του block size οδηγεί στην αύξηση του IPC και επομένως στη μείωση των MPKI (compulsory misses κατά κύριο λόγο).
4. Το benchmark rtview είναι εκείνο που επηρεάζεται λιγότερο από τις αλλαγές των παραμέτρων. Συγκεκριμένα το IPC κυμαίνεται μεταξύ 0.70 – 0.71, ενώ ταυτόχρονα τα MPKI μεταξύ 0.2 – 0.75 περίπου.
5. Το benchmark swaptions φαίνεται να επηρεάζεται καθοριστικά από το μέγεθος της L1 Cache. Η επιλογή L1 Cache Size μεγαλύτερο ή ίσο των 64K βελτιώνει δραματικά την επίδοση.

Σαν γενικό συμπέρασμα, βλέπουμε ότι το block size και το associativity επηρεάζουν σε βαθμό σε σχέση με το Cache Size.

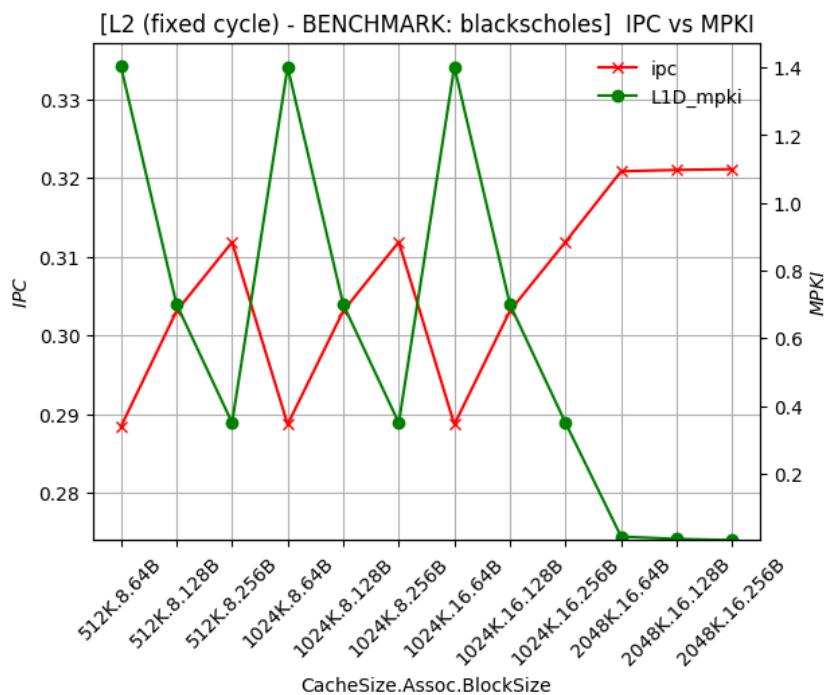
Θα μπορούσαμε να πούμε με βάση τις παραπάνω μετρήσεις, ότι οι συνδυασμοί 64K.8.128B και 128K.8.128B αποτελούν τις καλύτερες επιλογές μας αναφορικά με την L1 Cache.

### 2.1.2 L2 Cache

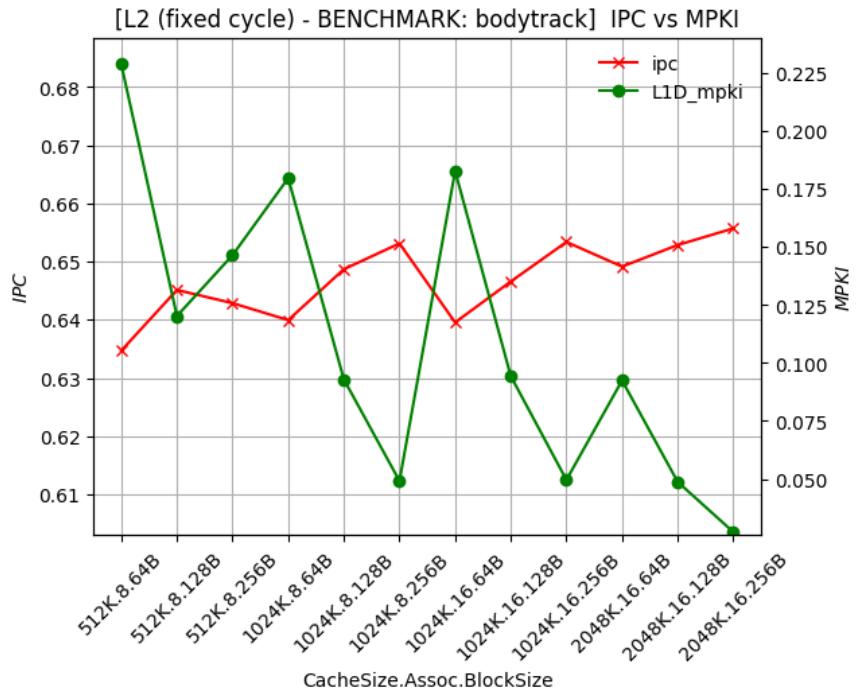
Οι παράμετροι της L1 cache και του TLB θα διατηρηθούν σταθερές και συγκεκριμένα ίσες με:

- L1 size = 32 KB
- L1 associativity = 8
- L1 block size = 64 B
- TLB size = 64 entr.
- TLB associativity = 4
- TLB page size = 4096 B

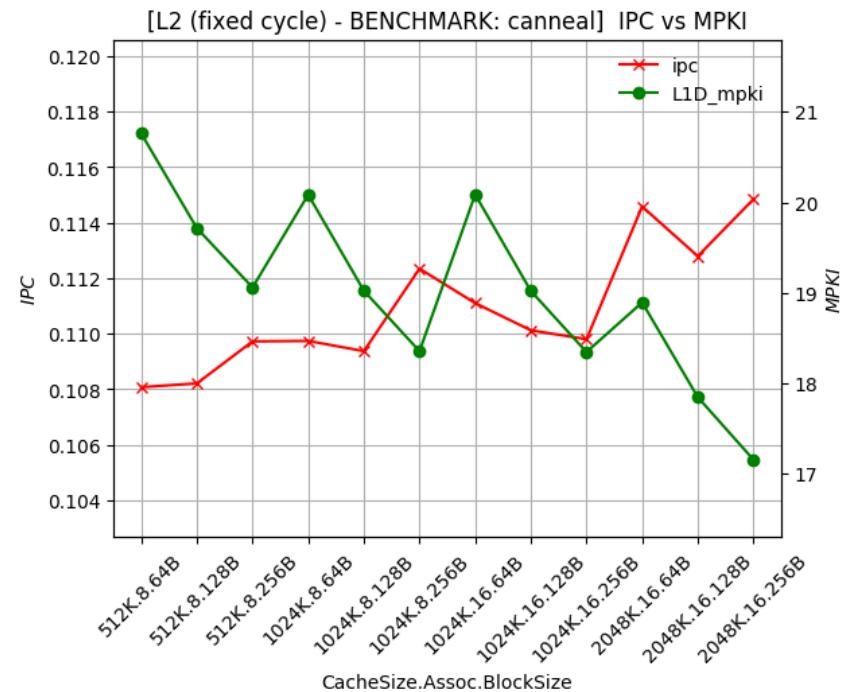
#### *blackscholes*



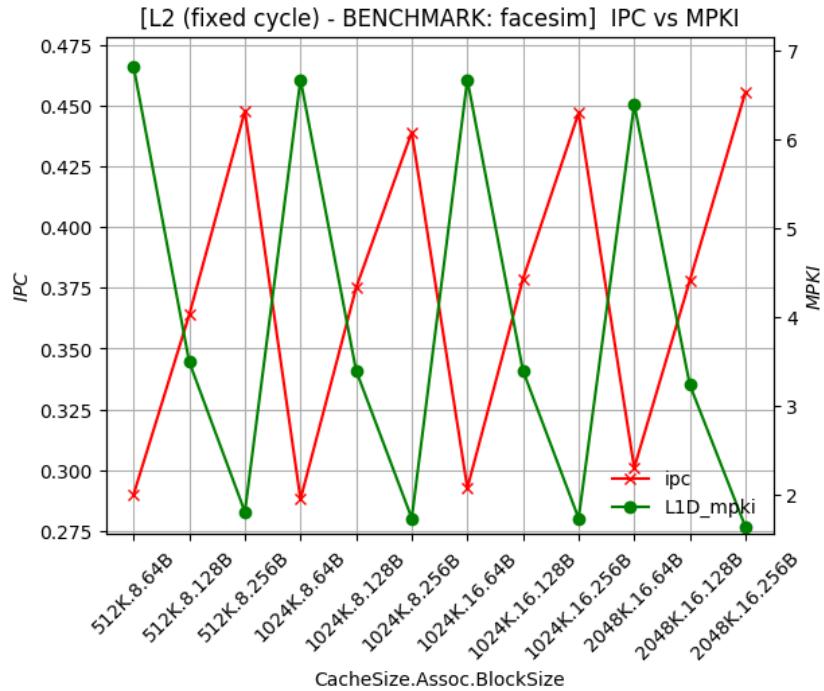
***bodytrack***



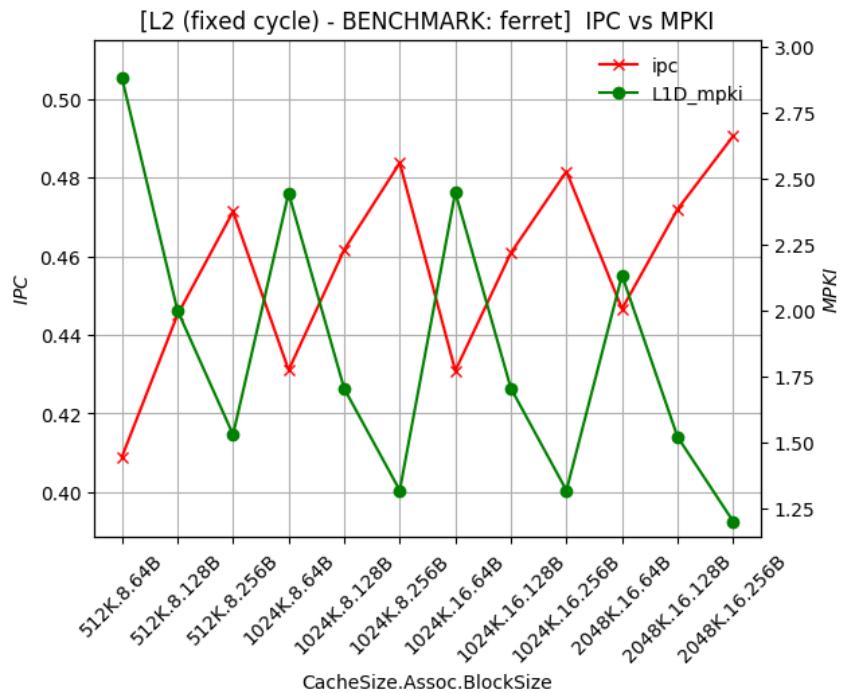
***canneal***



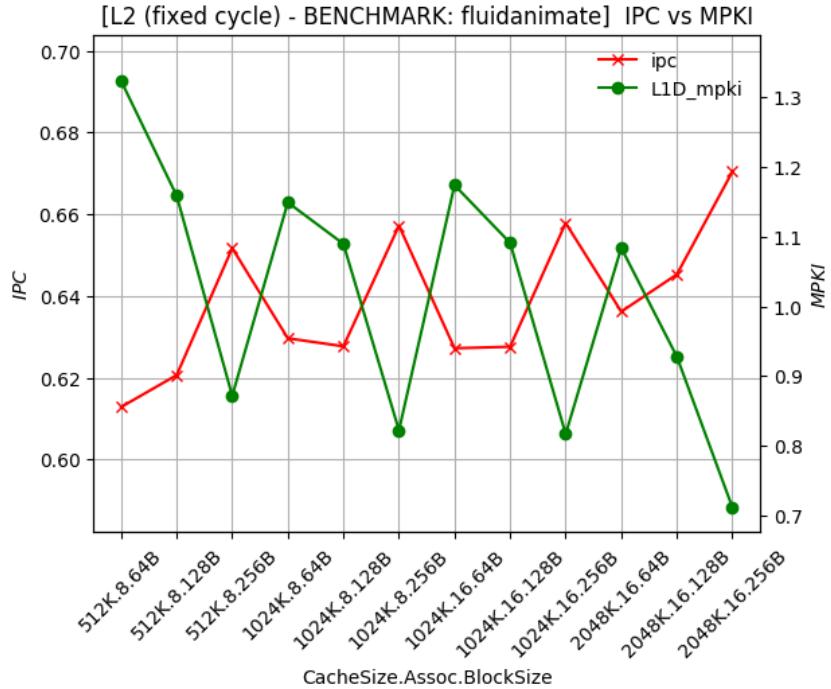
*facesim*



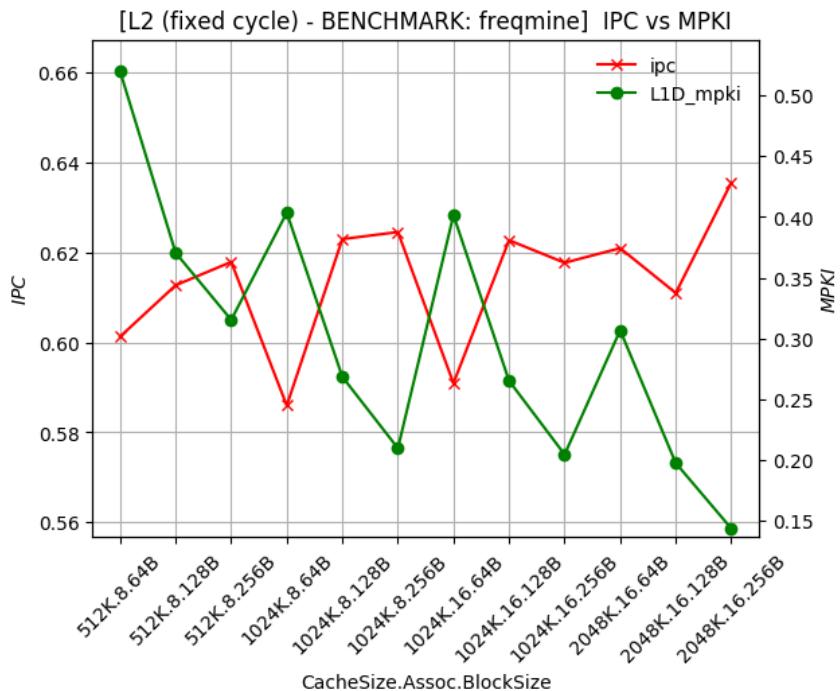
*ferret*



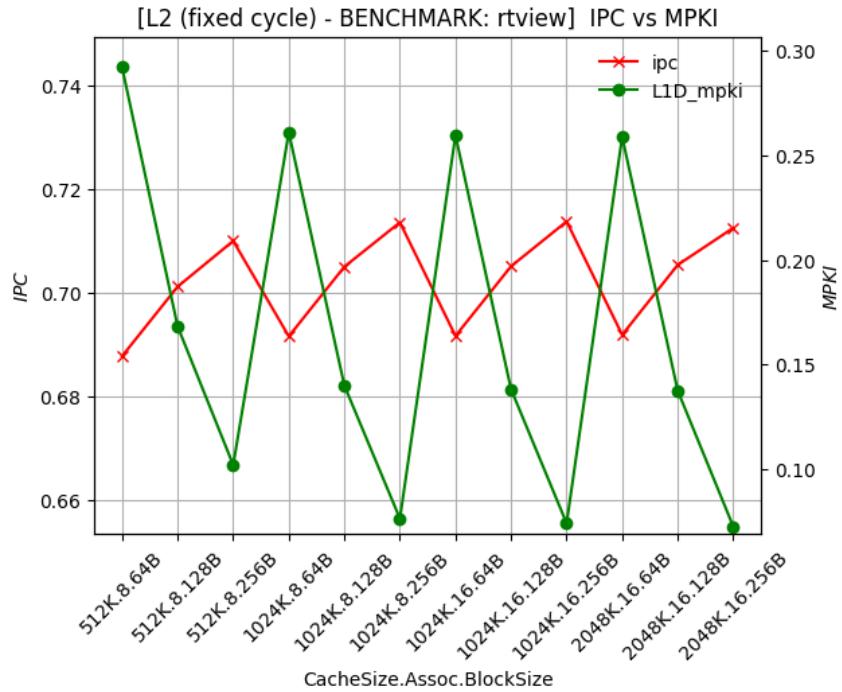
### *fluidanimate*



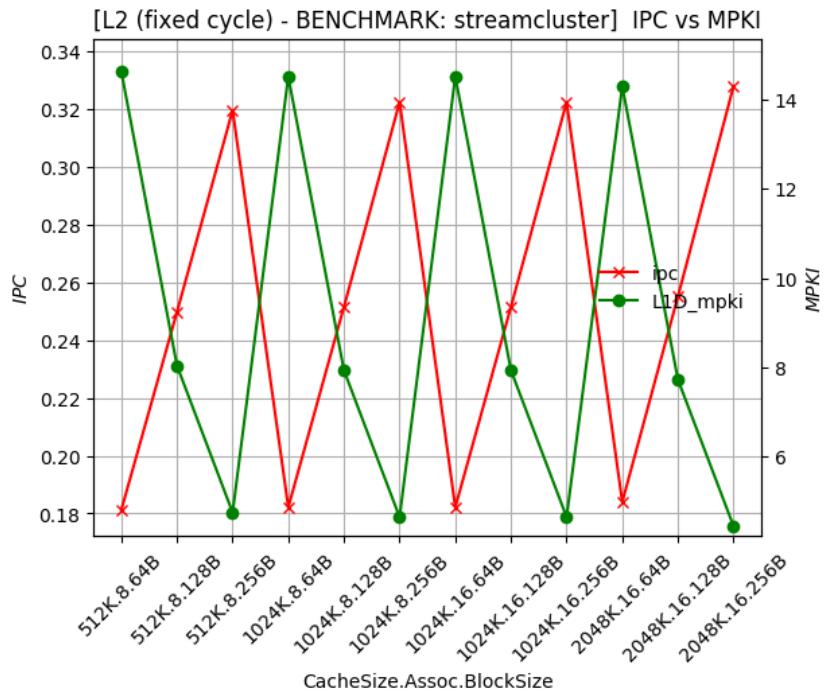
### *freqmine*



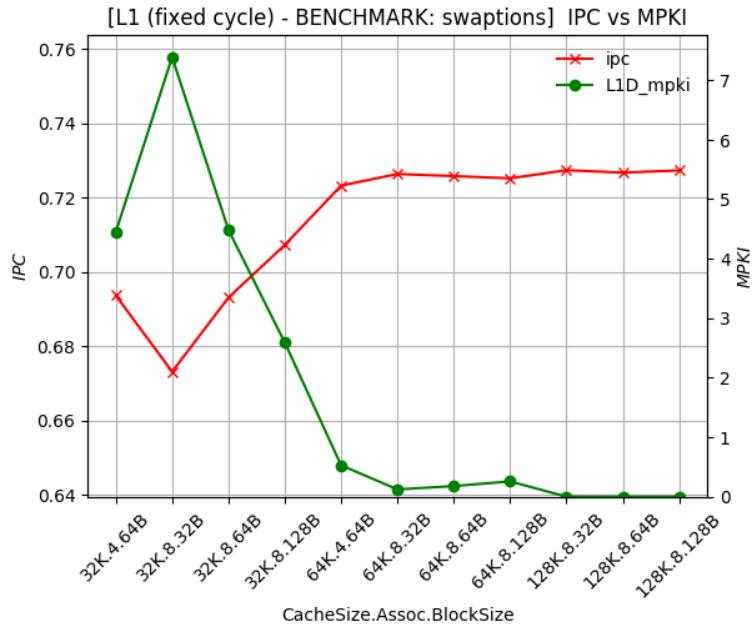
### *rtview*



### *streamcluster*

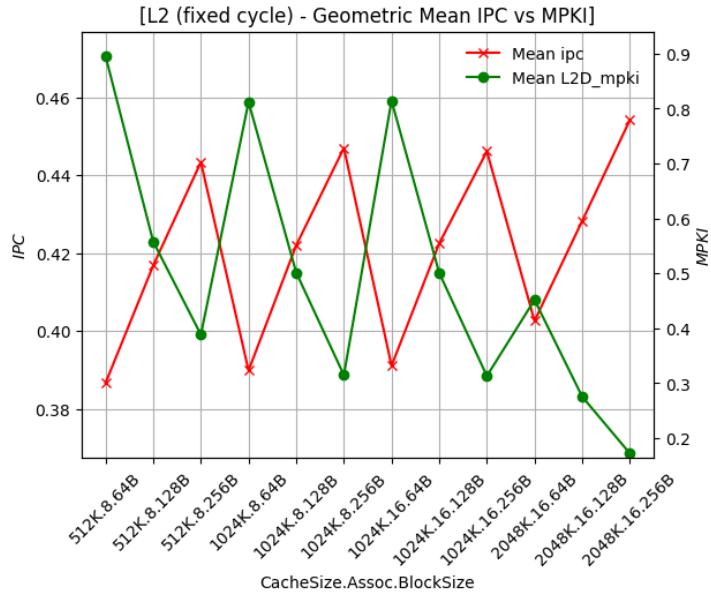


### swaptions



Επίσης, για λόγους συνολικής εποπτείας παρατίθεται ένα διάγραμμα των γεωμετρικών μέσων των προηγούμενων:

### Geometric Average



## Συμπεράσματα για την L2 Cache

Και σε αυτή την περίπτωση παρατηρούμε πως τα μεγέθη Instructions Per Cycle και Misses Per KiloInstructions μεταβάλλονται αντιστρόφως ανάλογα. Αυτό σαφώς είναι αναμενόμενο, αφού όσο περισσότερες αστοχίες υπάρχουν (μεγαλύτερο MPKI), τόσο περισσότεροι κύκλοι χρειάζονται για να εκτελεστεί η εντολή και άρα η επίδοση θα είναι χειρότερη, δηλαδή το IPC θα είναι μικρότερο, και αντίστροφα.

1. Στα benchmarks canneal, ferret, fluidanimate παρατηρούμε ότι η αύξηση του μεγέθους της cache προκαλεί τη βελτίωση της επίδοσης, οπότε και μειώνονται τα capacity misses.
2. Στο benchmark blackscholes, παρατηρούμε πως η επιλογή L2 Cache Size 2048K επιφέρει δραστική βελτίωση της επίδοσης. Μάλιστα για την επιλογή αυτή, τα υπόλοιπα χαρακτηριστικά φαίνεται πως δεν επηρεάζουν πλέον την απόδοση περεταίρω.
3. Τα benchmark blackscholes και ferret επηρεάζονται περισσότερο από το associativity η οποία λογικά οφείλεται σε μείωση των conflict misses. Σε γενικές γραμμές, η αύξηση του associativity δεν επηρεάζει αρνητικά.
4. Στα benchmarks blackscholes, bodytrack, facesim, ferret, fluidanimate, rtview, streamcluster είναι εμφανές ότι η αύξηση του block size προκαλεί και αύξηση στο IPC, ενώ τα compulsory misses μειώνονται.
5. Το benchmark swaptions δείχνει να επηρεάζεται λιγότερο από τις αλλαγές των παραμέτρων και διατηρεί σχεδόν σταθερό IPC. Αυτό ίσως οφείλεται στο μέγεθος των δεδομένων που επεξεργάζεται και είτε χρησιμοποιεί συνεχώς τα ίδια blocks, είτε όποια blocks διώχνονται δεν απαιτούνται ξανά.

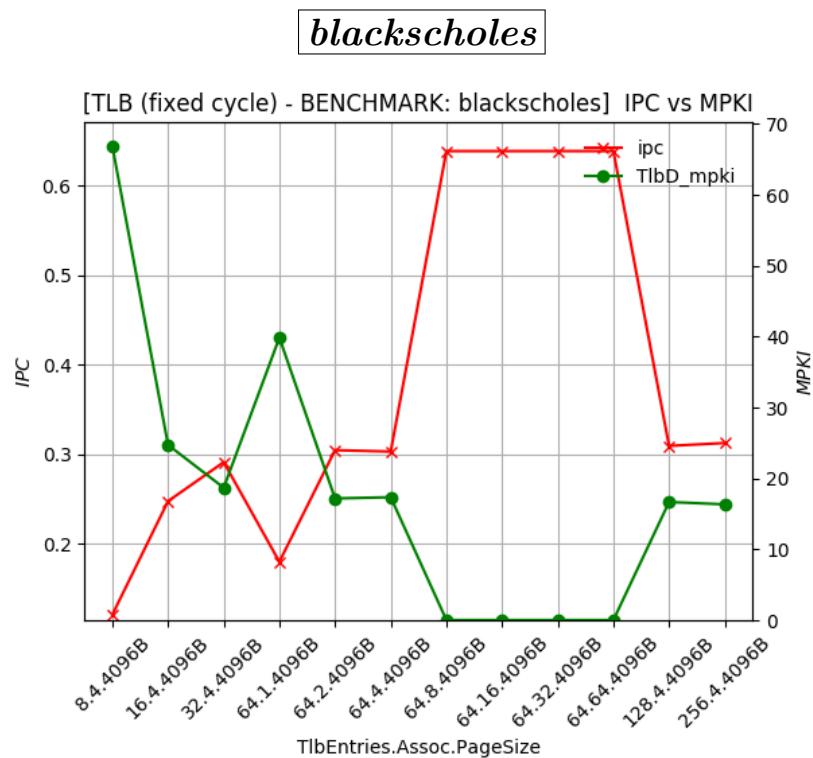
Σε γενικές γραμμές λοιπόν, βλέπουμε και πάλι ότι τη μεγαλύτερη επίδραση στην απόδοση έχει η μεταβολή του block size, χωρίς να είναι αμελητέες οι αλλαγές που προκύπτουν από την αύξηση-μείωση των cache size και associativity.

Θα μπορούσαμε λοιπόν με τις παραπάνω μετρήσεις να πούμε πως οι συνδυασμοί  $2048K.16.256B$  και  $2048K.16.256B$  αποτελούν αρκετά καλές επιλογές για την *L2 Cache*.

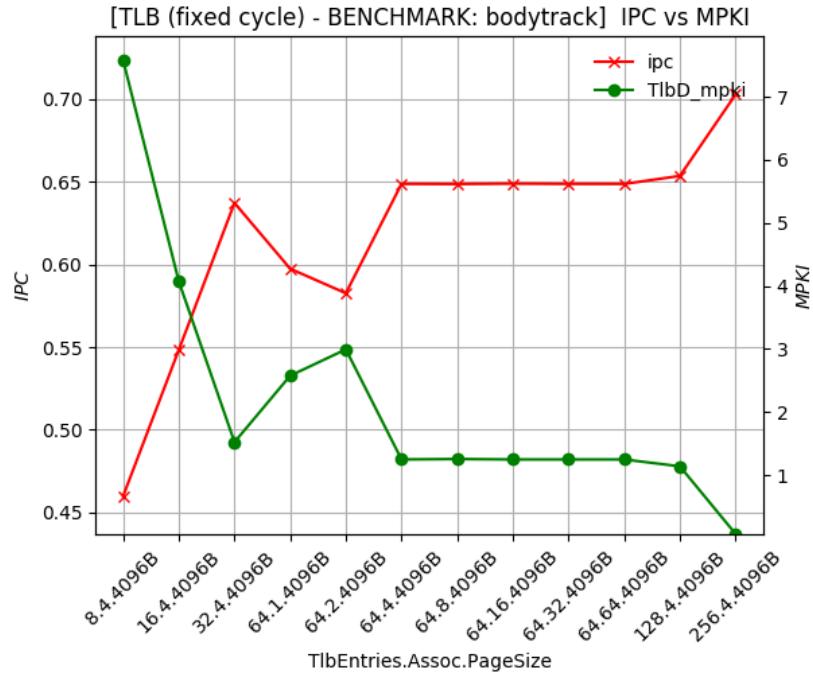
### 2.1.3 TLB

Οι παράμετροι της L1 cache και της L2 cache θα διατηρηθούν σταθερές και συγκεκριμένα ίσες με:

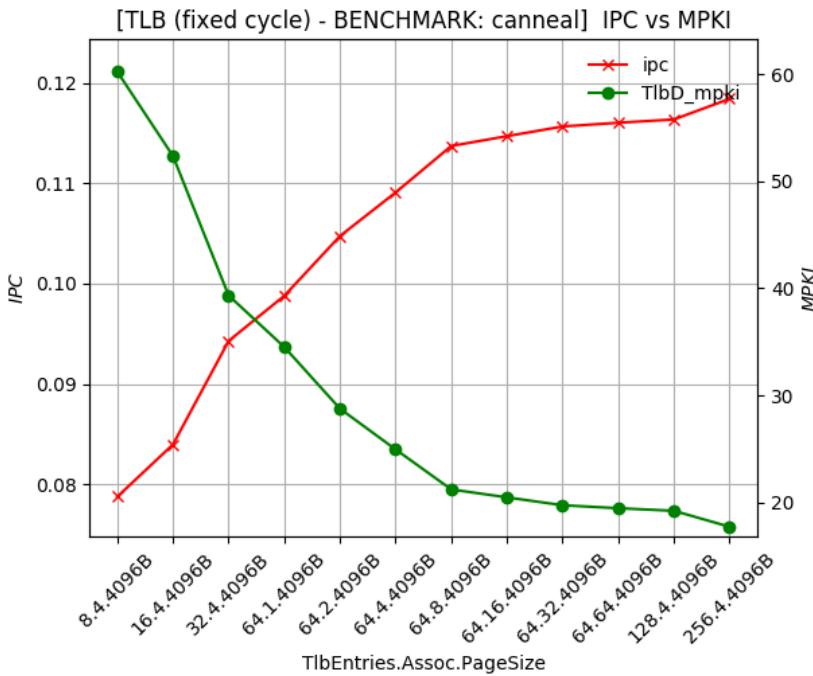
- L1 size = 32 KB
- L1 associativity = 8
- L1 block size = 64 B
- L2 size = 1024 KB
- L2 associativity = 8
- L2 block size = 128 B



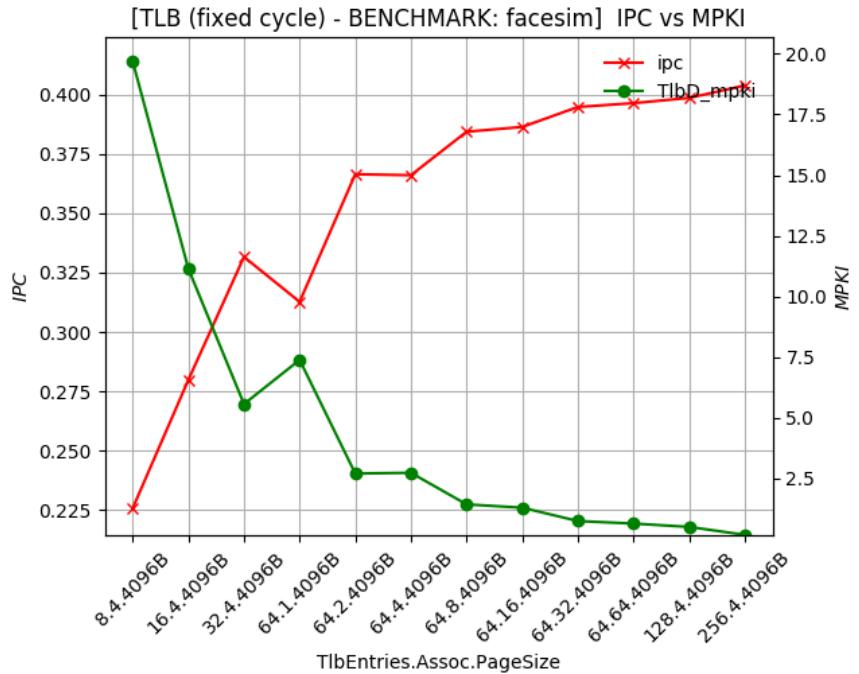
### *bodytrack*



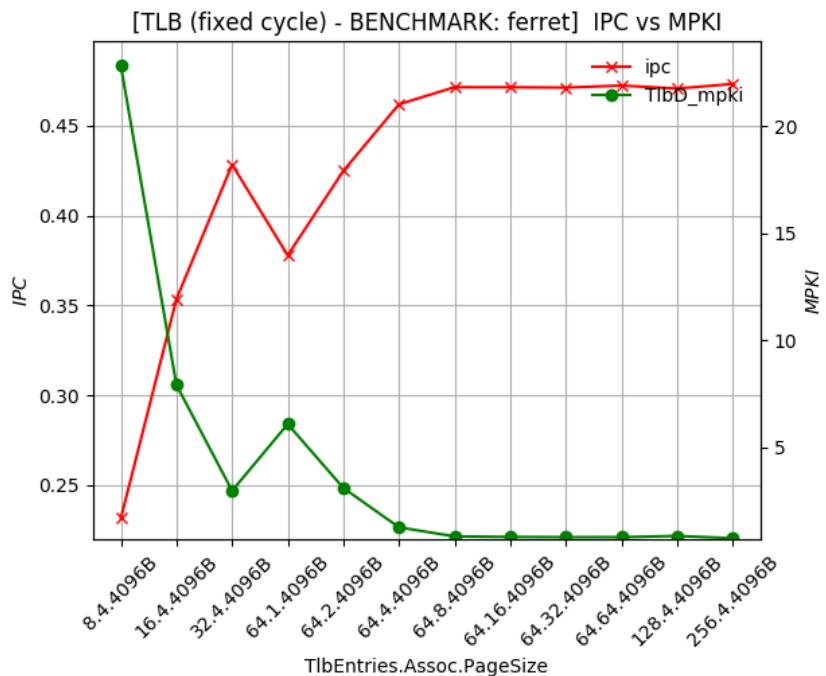
### *canneal*



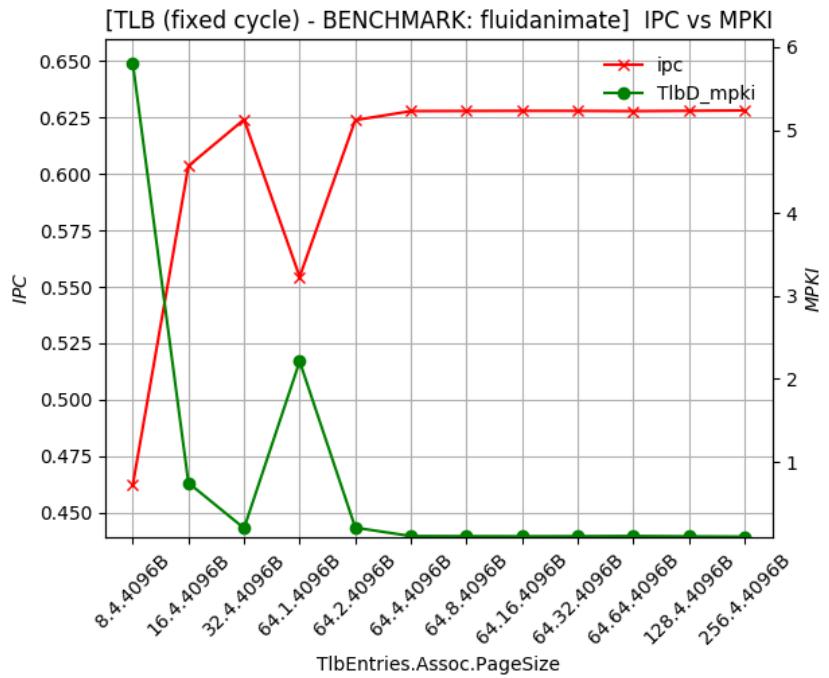
*facesim*



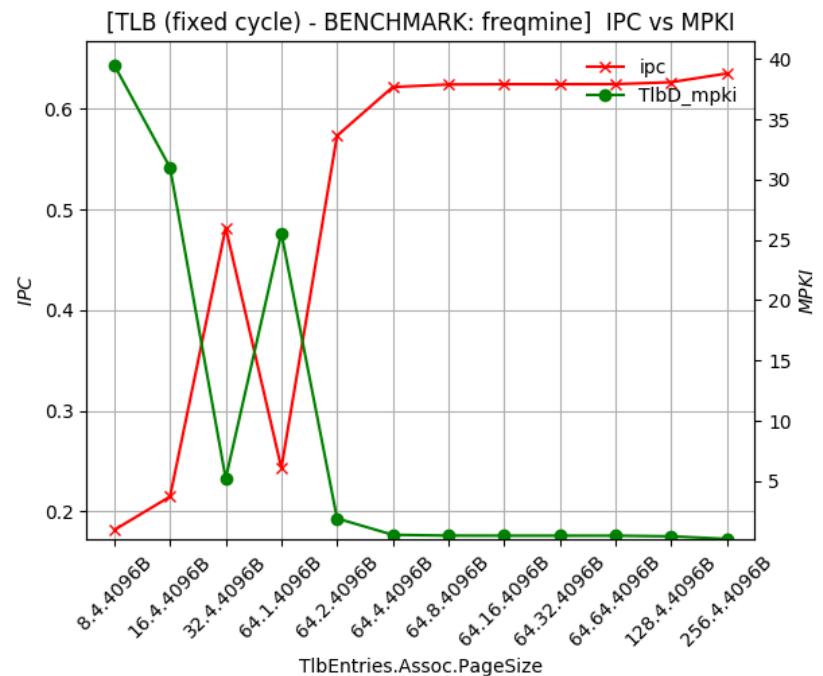
*ferret*



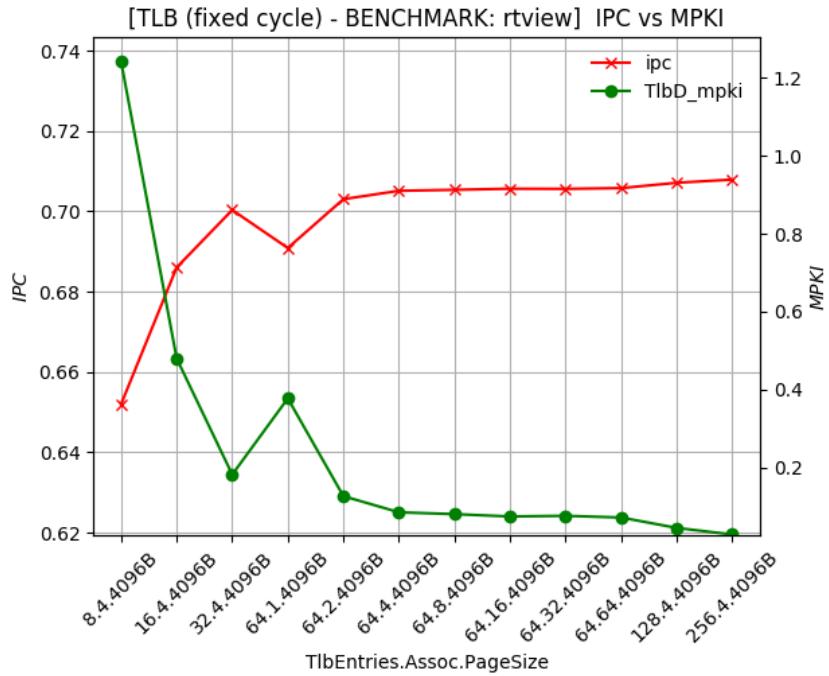
## ***fluidanimate***



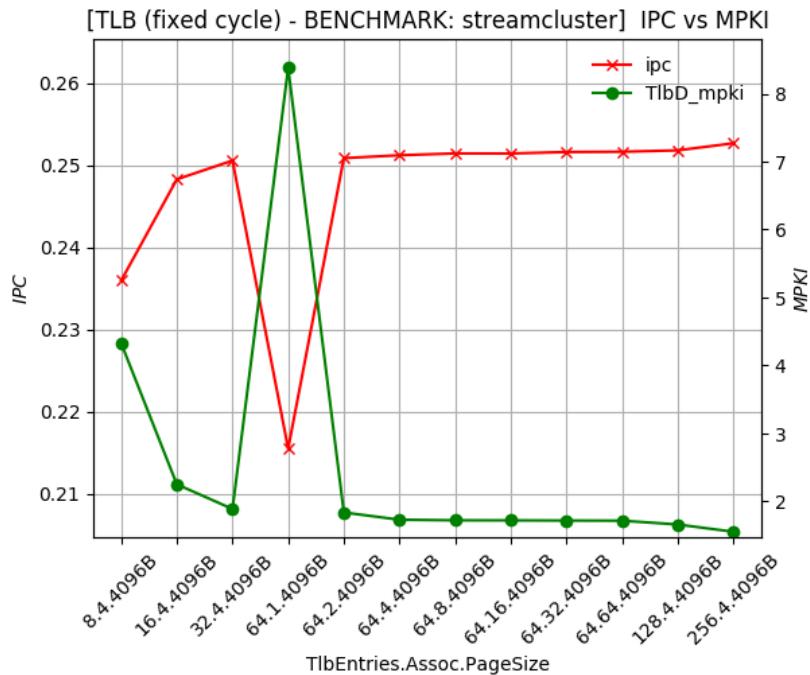
## ***freqmine***



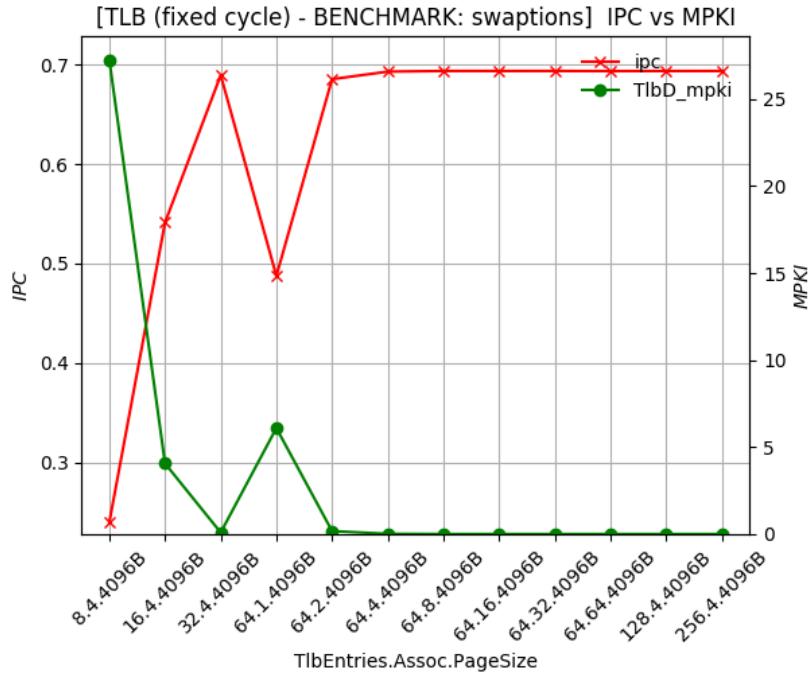
### rtview



### streamcluster

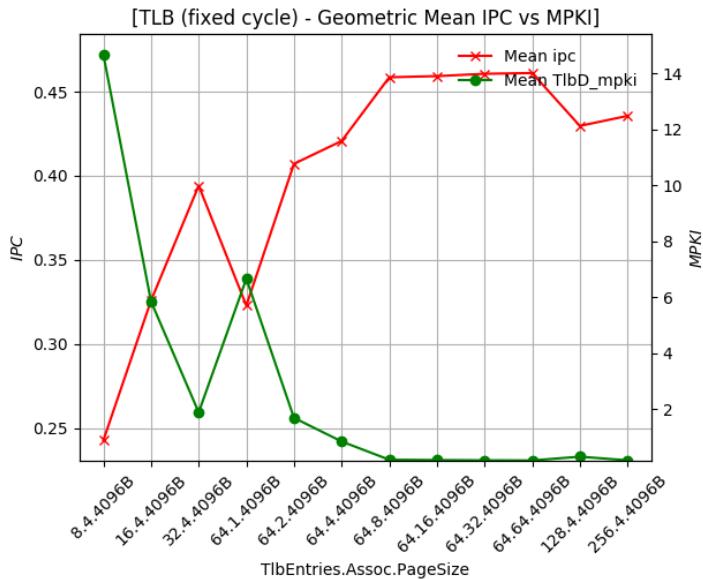


### swaptions



Επίσης, για λόγους συνολικής εποπτείας παρατίθεται ένα διάγραμμα των γεωμετρικών μέσων των προηγούμενων:

### Geometric Average



## Συμπεράσματα για το TLB

Αναμενόμενα παρατηρούμε να μεταβάλλονται αντιστρόφως ανάλογα τα μεγέθη IPC, MPKI. Αυτό είναι λογικό αφού όσο περισσότερες αστοχίες γίνονται, τόσο χειρότερη επίδοση θα έχουμε (άρα μικρότερο IPC) και αντίστροφα.

1. Σε όλα τα benchmarks, παρατηρούμε ότι η αύξηση του TLB size έχει ως αποτέλεσμα την αύξηση του IPC, οπότε και τη μείωση των misses. Ωστόσο, σε όλα τα μετροπρογράμματα με εξαίρεση ίσως των blackscholes, bodytrack, η συνεχής αύξηση του TLB size πάνω από 64B ή 128B δεν προσέφερε κάποια ουσιαστική αλλαγή στην απόδοση.
2. Όσον αφορά το associativity, παρατηρούμε σε όλα τα benchmarks ότι συμβάλει θετικά στην αύξηση της απόδοσης, ωστόσο μέχρι ένα όριο. Στα περισσότερα, η απόδοση είχε σημαντική βελτίωση μέχρι το associativity να πάρει τιμές από 4 έως 8, αλλά από εκεί και πάνω, δεν υπήρχε κάποια αξιοσημείωτη μεταβολή.
3. Για το TLB page δεν μπορούμε να εξάγουμε κάποιο συμπέρασμα, δίοτι μας δίνεται σε όλες τις μετρήσεις ίσο με 4096B.
4. Δεν υπήρχε κάποιο benchmark του οποίου δεν επηρεάστηκε η απόδοση και οι αστοχίες μετά τη μεταβολή των παραμέτρων.

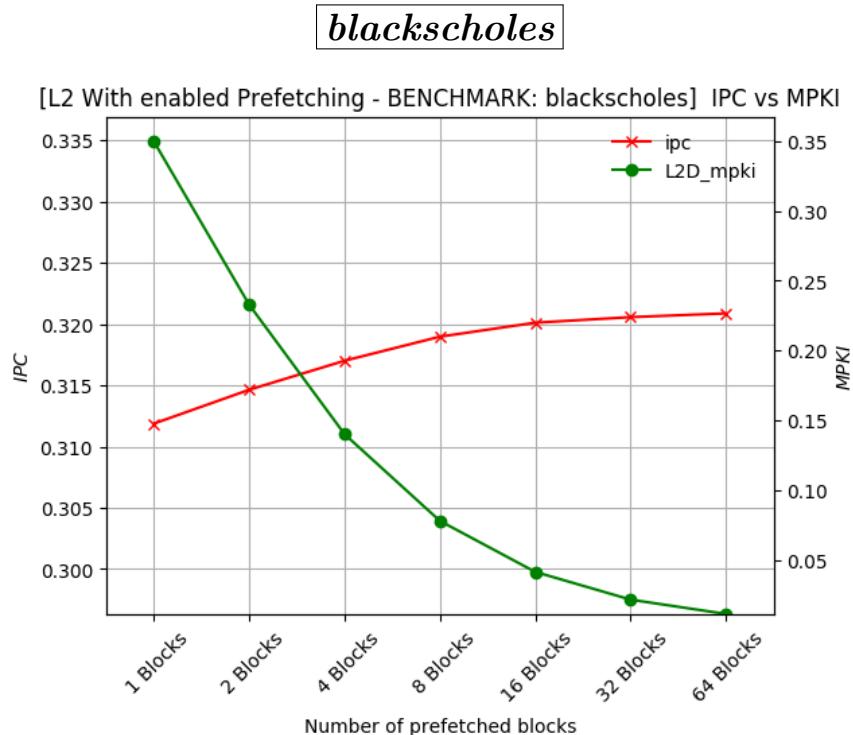
Γενικά λοιπόν, λαμβάνοντας υπόψιν και το διάγραμμα των γεωμετρικών μέσων τιμών, διαπιστώνουμε ότι τόσο το μέγεθος του TLB, όσο και η το associativity παίζουν σημαντικό ρόλο στην επίδοση, μέχρι ενός ορίου. Από ένα σημείο και πέρα η αύξησή τους δεν επηρεάζει σημαντικά, ούτε στο IPC, ούτε στα misses.

*Μπορούμε με τις παραπάνω μετρήσεις και διαγράμματα για το TLB, να διαλέγουμε ως αρκετά καλή επιλογή χαρακτηριστών το συνδυασμό 64.8.4096B.*

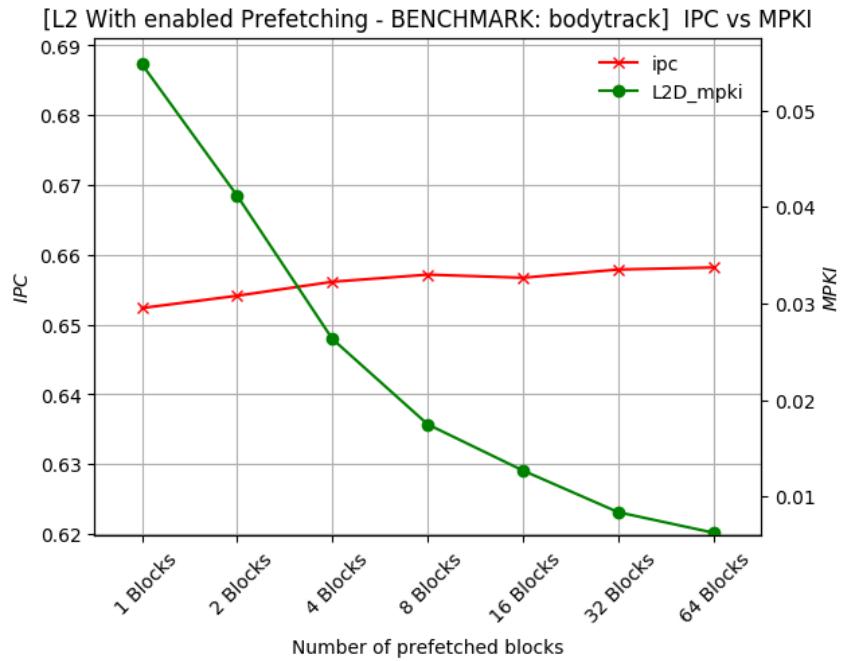
#### 2.1.4 Prefetching

Οι παράμετροι της L1 cache και της L2 cache και TLB θα διατηρηθούν σταθερές και συγκεκριμένα ίσες με:

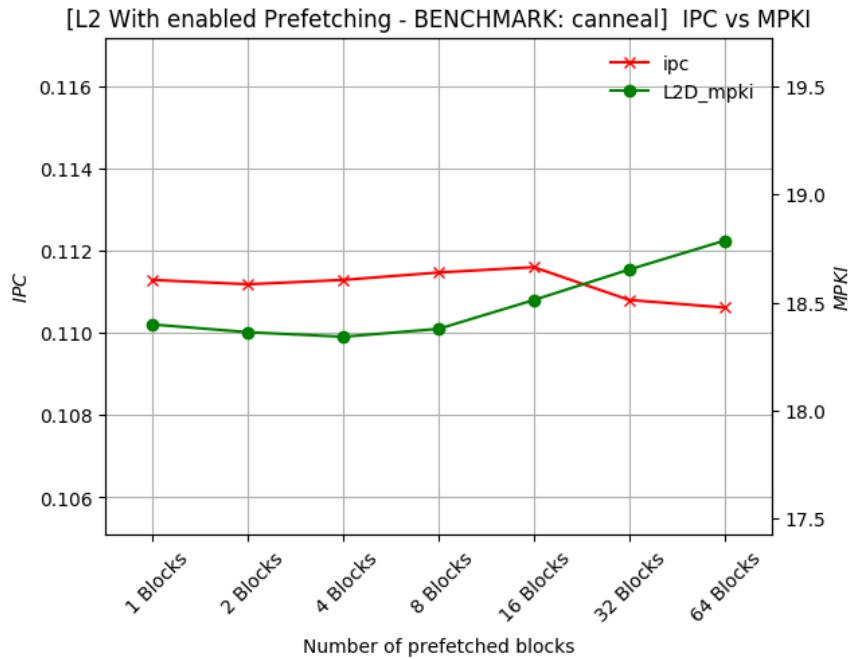
- L1 size = 32 KB
- L1 block size = 64 B
- L1 associativity = 8
- L2 size = 1024 KB
- L2 block size = 128 B
- L2 associativity = 8
- TLB size = 64 entries
- TLB associativity = 4
- TLB page size = 4096 B



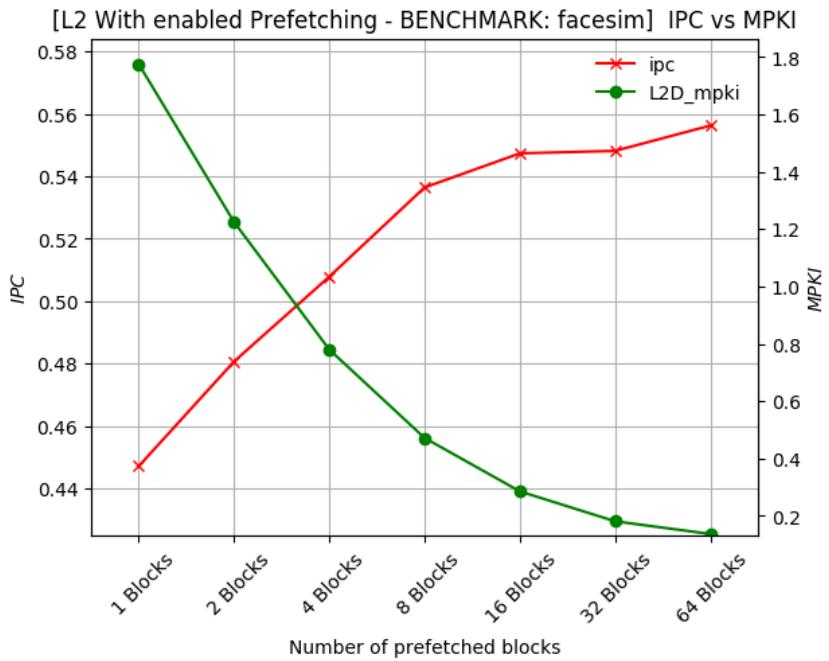
### *bodytrack*



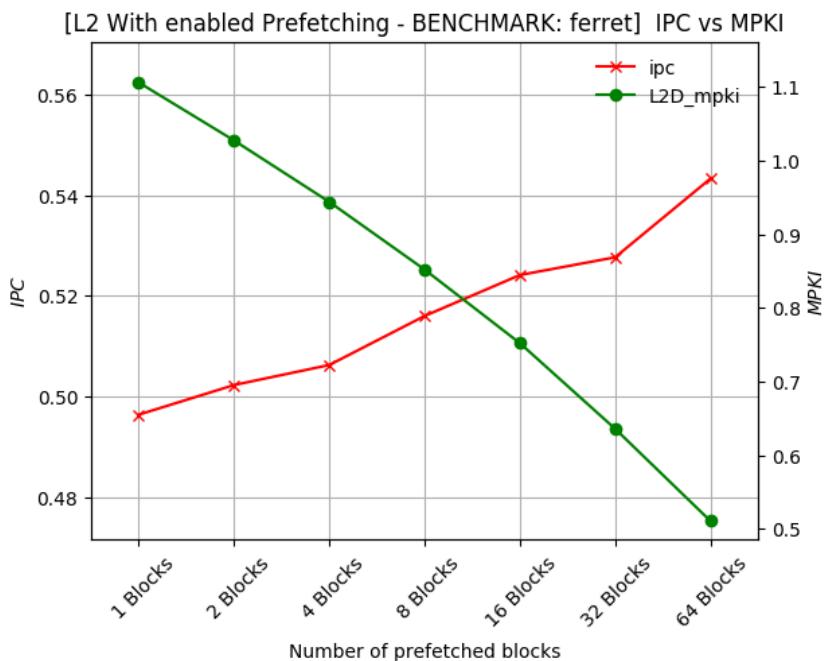
### *canneal*



*facesim*

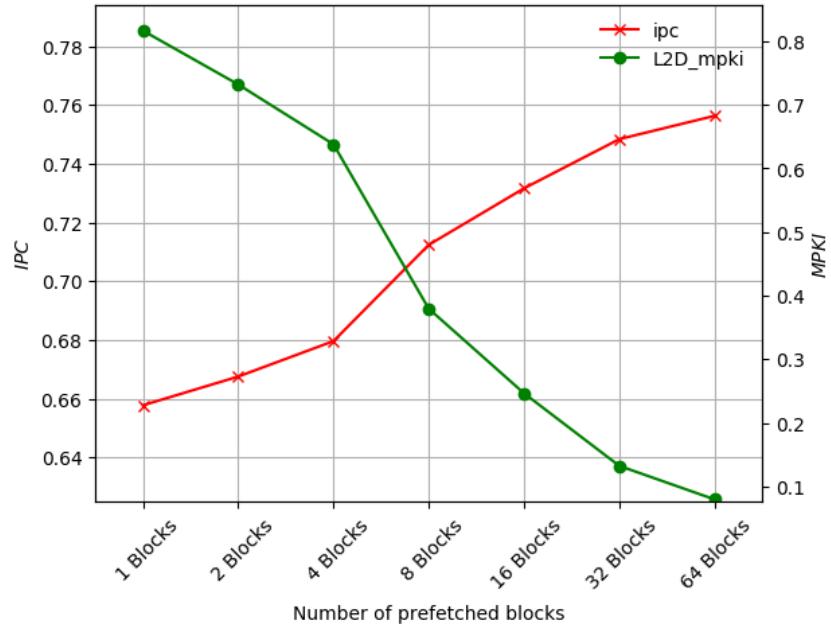


*ferret*



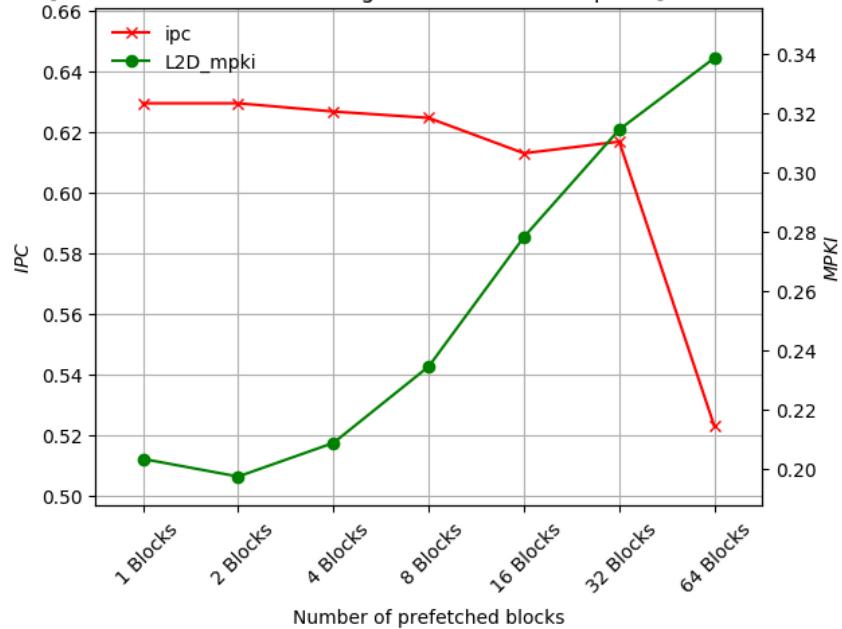
## *fluidanimate*

[L2 With enabled Prefetching - BENCHMARK: *fluidanimate*] IPC vs MPKI

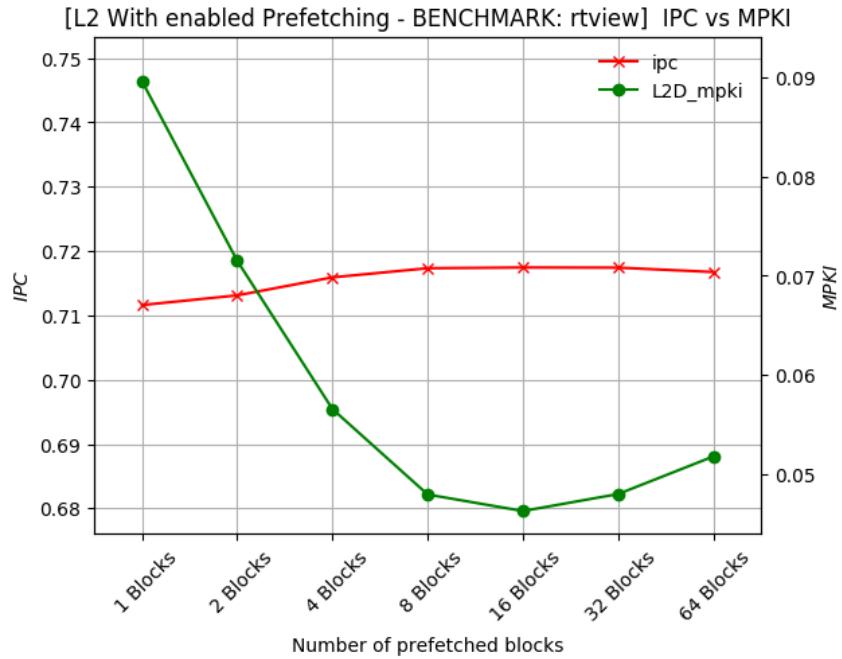


## *freqmine*

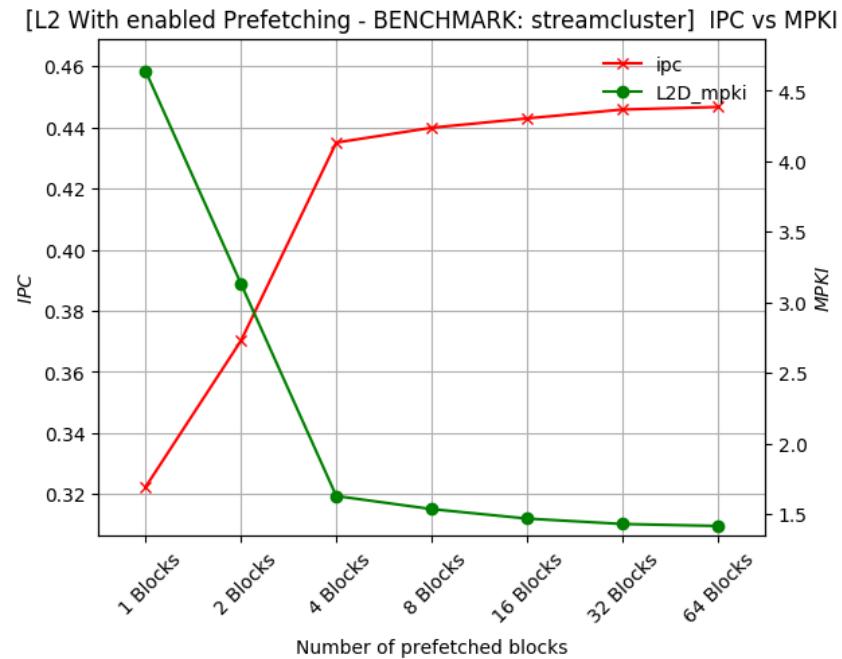
[L2 With enabled Prefetching - BENCHMARK: *freqmine*] IPC vs MPKI

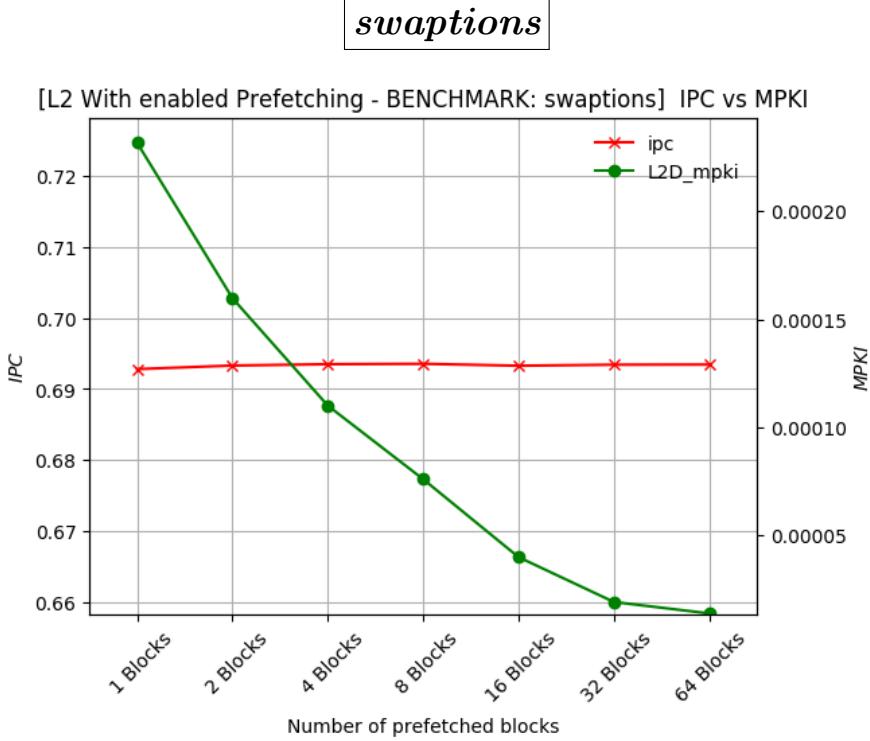


### *rtview*



### *streamcluster*





### Συμπεράσματα για το Prefetching

- Παρατηρούμε πως υπάρχουν benchmarks, στα οποία δεν επηρεάζεται σχεδόν καθόλου το IPC τους από το prefetching. Αυτά είναι τα εξής: blackscholes (IPC στο διάστημα 0.31 – 0.32 περίπου), bodytrack (IPC στο διάστημα 0.65 – 0.66 περίπου), canneal (IPC στο διάστημα 0.110 – 0.112 περίπου), rtview (IPC στο διάστημα 0.71 – 0.72 περίπου), swaptions (IPC στο διάστημα 0.690 – 0.695 περίπου). Λόγω αυτής της συμπεριφοράς, μπορούμε να συμπεράνουμε πως οι προσπελάσεις δεν γίνονται σε διαδοχικά blocks.
- Στα benchmarks facesim, ferret, fluidanimate, streamcluster παρατηρούμε ότι παρουσιάζουν σημαντική βελτίωση στο IPC τους για αυξανόμενο πλήθος prefetched blocks.
- Στο benchmark freqmine βλέπουμε ότι παρουσιάζει κατά κύριο λόγο μείωση του IPC με την αύξηση του πλήθους των prefetched blocks. Αυτό μπορεί να συμβαίνει διότι κατά το prefetching, κάποια block στην cache αντικαθίστανται υποχρεωτικά από άλλα, και αν φεύγουν συνεχώς αυτά που χρειάζεται το πρόγραμμα, θα αυξάνονται τα misses.

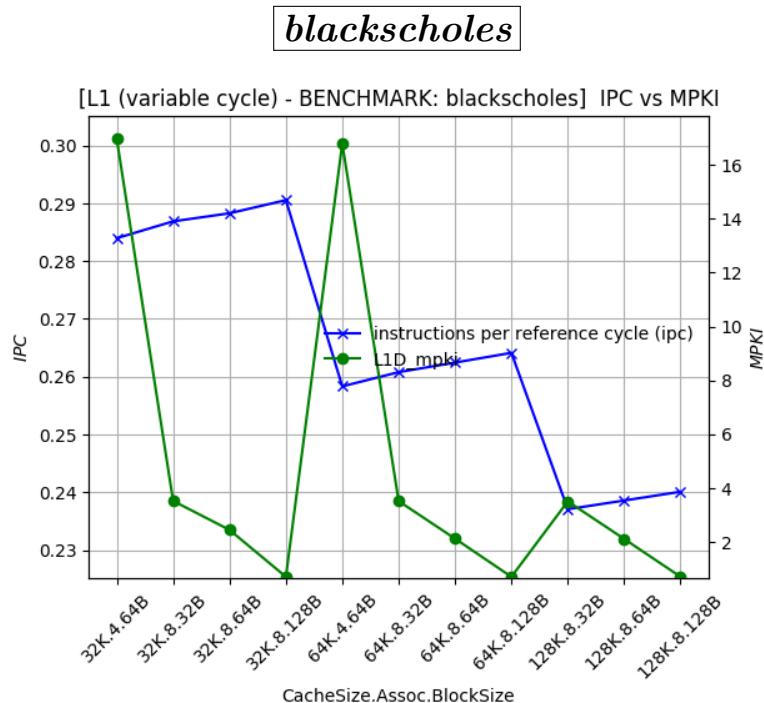
Σε γενικές γραμμές λοιπόν, βλέπουμε ότι το prefetching είναι επιθυμητό και επιδρά θετικά στην αύξηση της επίδοσης των προγραμμάτων μέχρι ενός ορίου.

## 2.2 Μελέτη επίδρασης παραμέτρων ιεραρχίας μνήμης στην απόδοση της εφαρμογής για μεταβλητό κύκλο ρολογιού

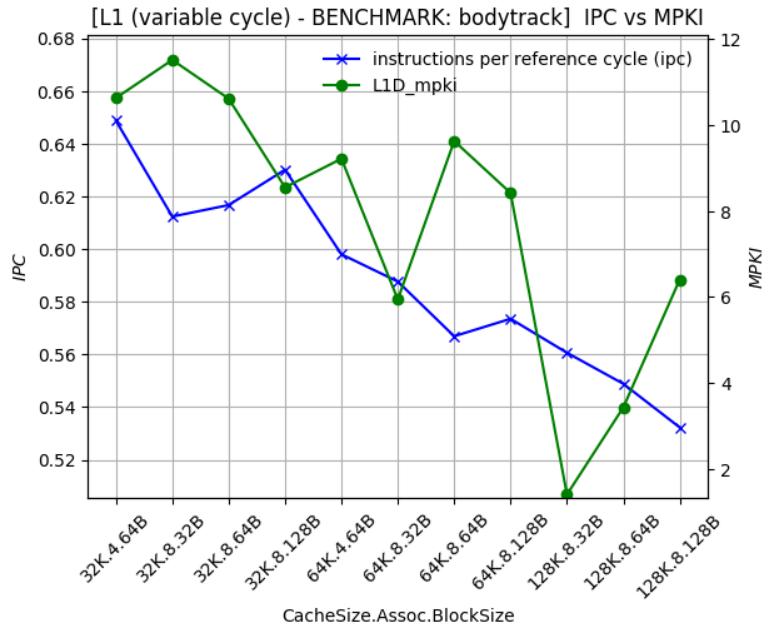
Στο κομμάτι αυτό της εργασίας γίνεται σχολιασμός των αποτελεσμάτων των προηγούμενων πειραμάτων, θεωρώντας όμως ότι ο κύκλος του ρολογιού μεταβάλλεται καθώς αλλάζουν τα χαρακτηριστικά των L1 Cache, L2 Cache, TLB. Σχολιάζουμε την επίδοση των διαφορετικών benchmarks θεωρώντας κάθε φορά ως αρχικό σημείο αναφοράς τη πρώτη προσομοίωση για το καθένα από αυτά και πως διπλασιασμός του associativity ή του μεγέθους (size) προκαλεί αύξηση του κύκλου κατά 5 % ή 10% αντίστοιχα.

Συγκεκριμένα, θεωρώντας ως αρχικό κύκλο αναφοράς τον κύκλος της πρώτης προσομοίωσης, σχεδιάζουμε διαγράμματα στα οποία **το ipc αντιπροσωπεύει το πλήθος των εντολών που εκτελούνται σε χρνικό διάστημα ίσο με τον αρχικό κύκλο αναφοράς.**

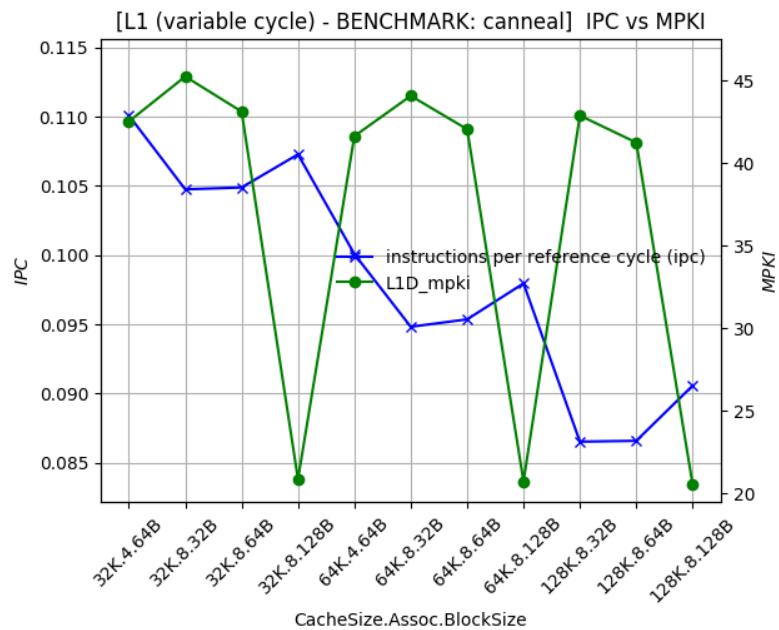
### 2.2.1 L1 Cache



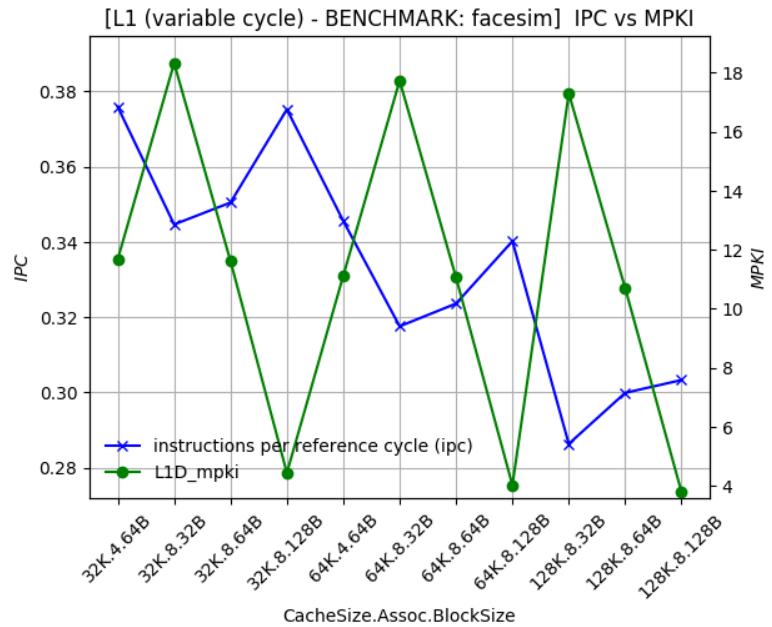
### *bodytrack*



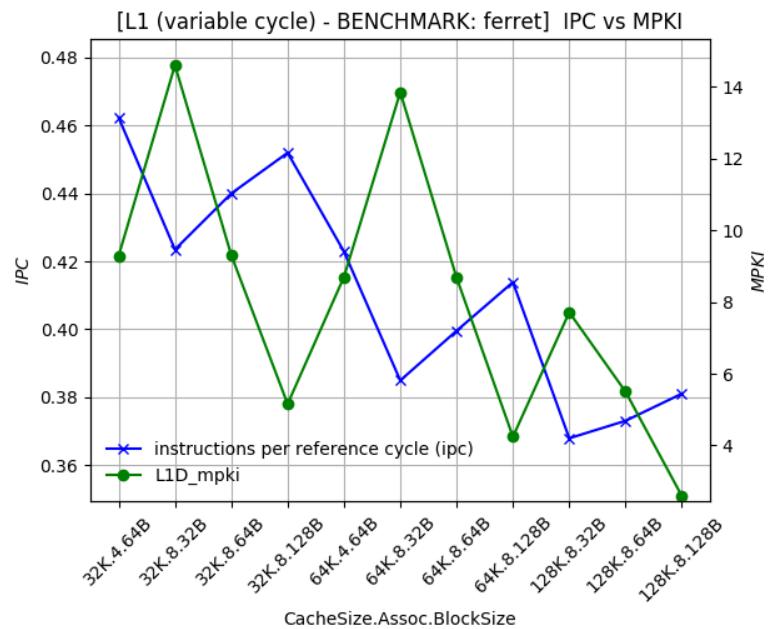
### *canneal*



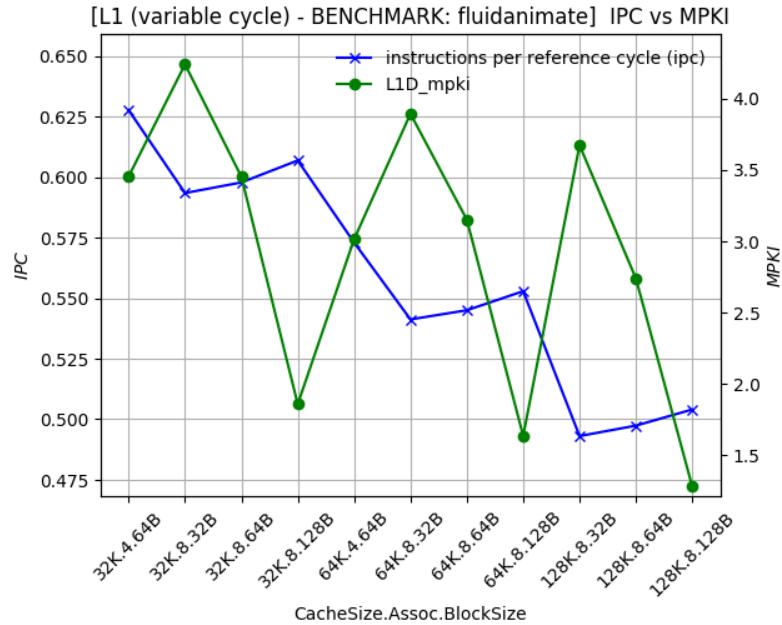
***facesim***



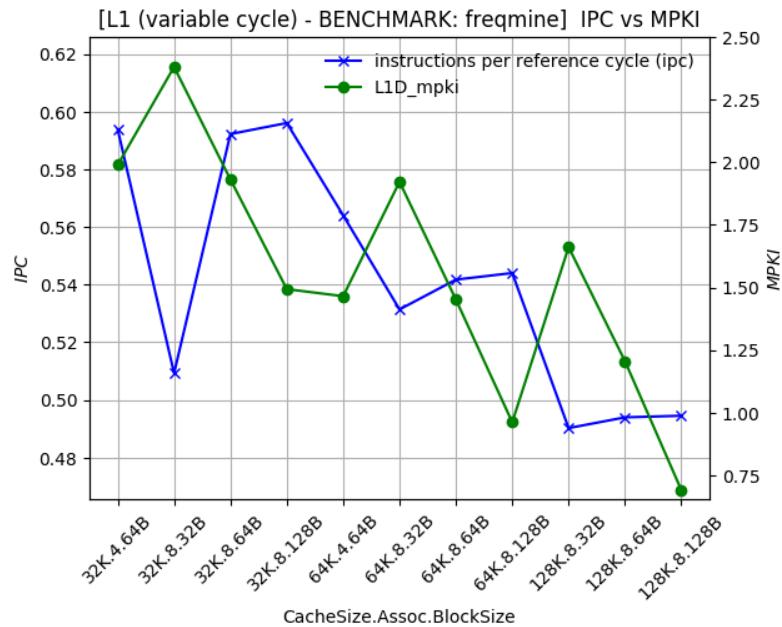
***ferret***



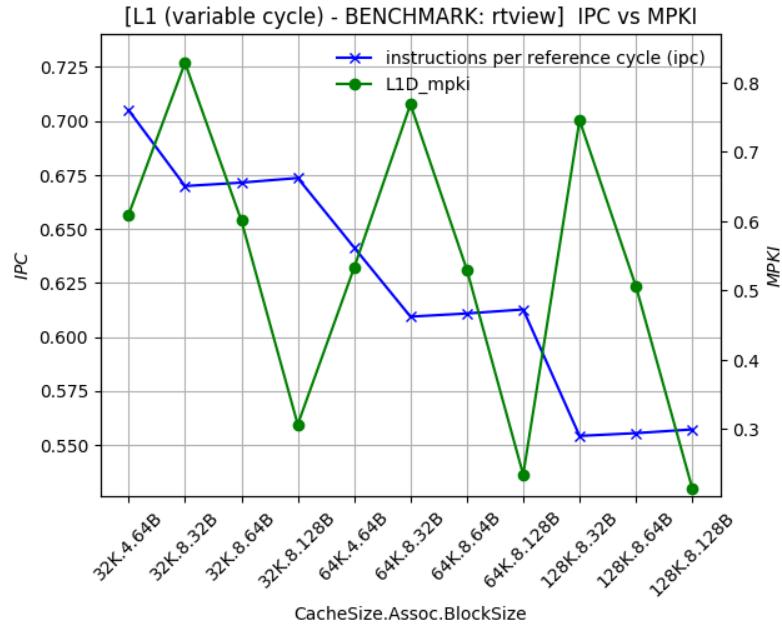
### ***fluidanimate***



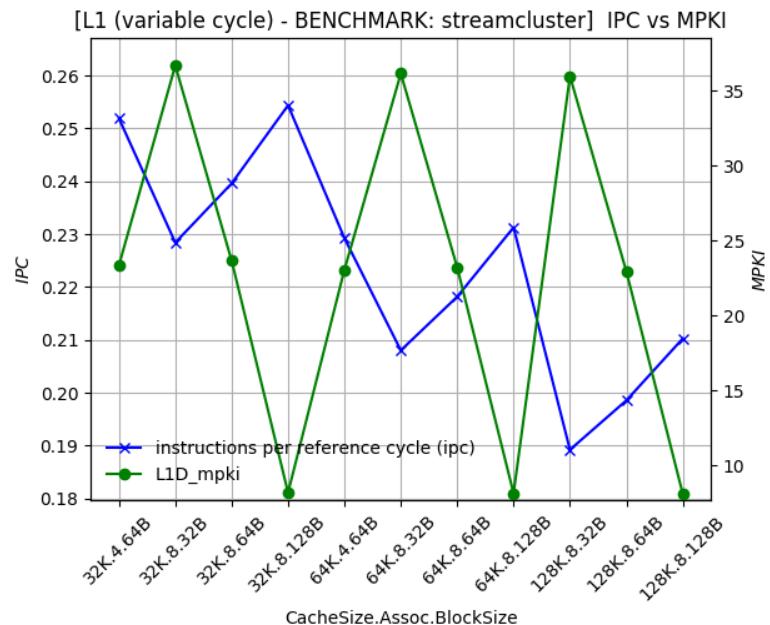
### ***freqmine***



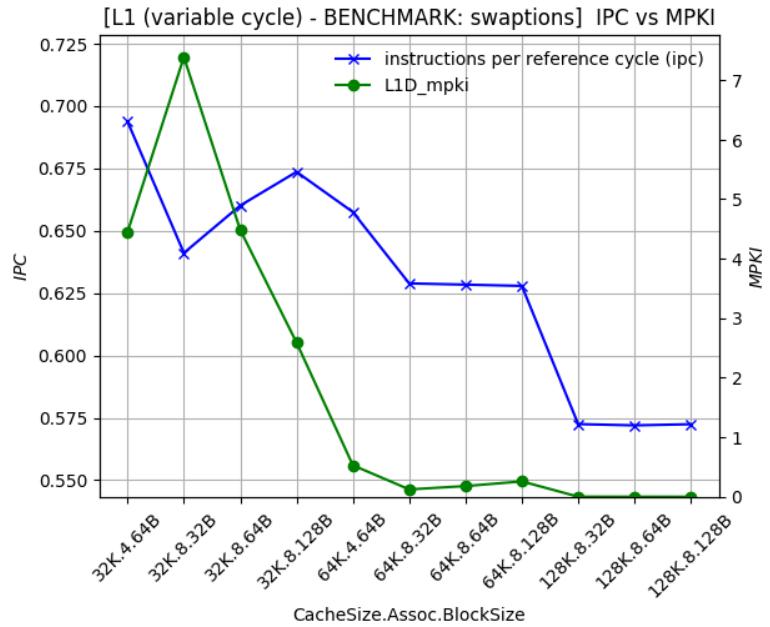
### *rtview*



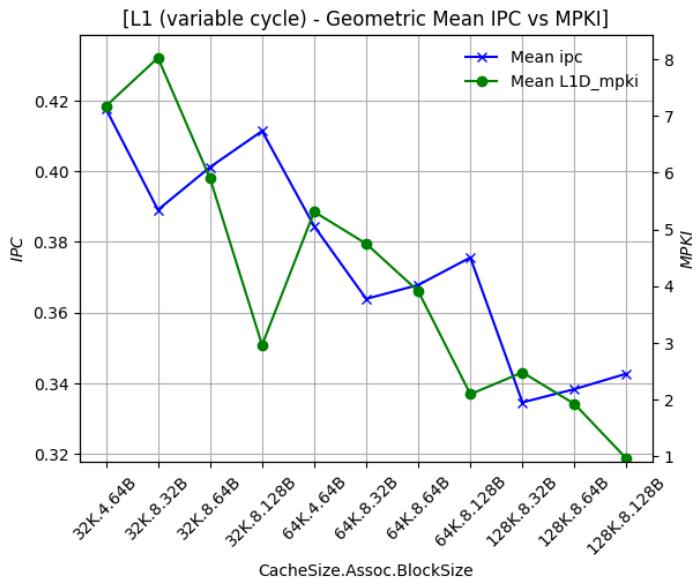
### *streamcluster*



### *swaptions*



### *Geometric Average*



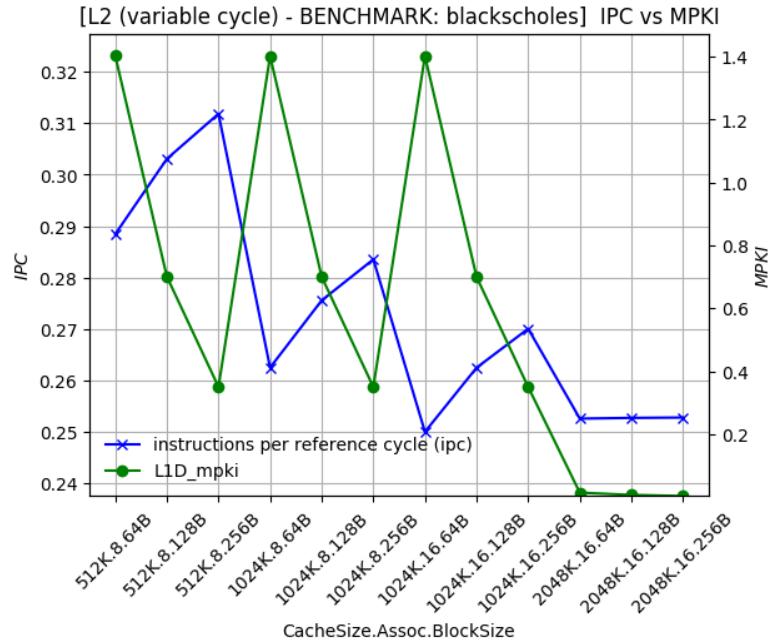
## Συμπεράσματα για την L1 Cache

Παρατηρούμε πως πλέον δεν είναι απαραίτητα αντιστρόφως ανάλογα τα μεγέθη IPC, MPKI. Καθώς, όπως αναφέραμε, πλέον το IPC αναφέρεται στον αρχικό κύκλο του πειράματος αναφοράς, πλέον ενώ μπορεί να αυξάνονται οι εντολές που εκτελούνται στον νέο κύκλο, η διάρκεια του κύκλου λόγω μεταβολής ενδέχεται να είναι αρκετά μεγαλύτερη σε σχέση με τον κύκλο αναφοράς οπότε και η επίδοση, δηλαδή οι εντολές ανα χρονικό διάστημα να μην είναι καλή.

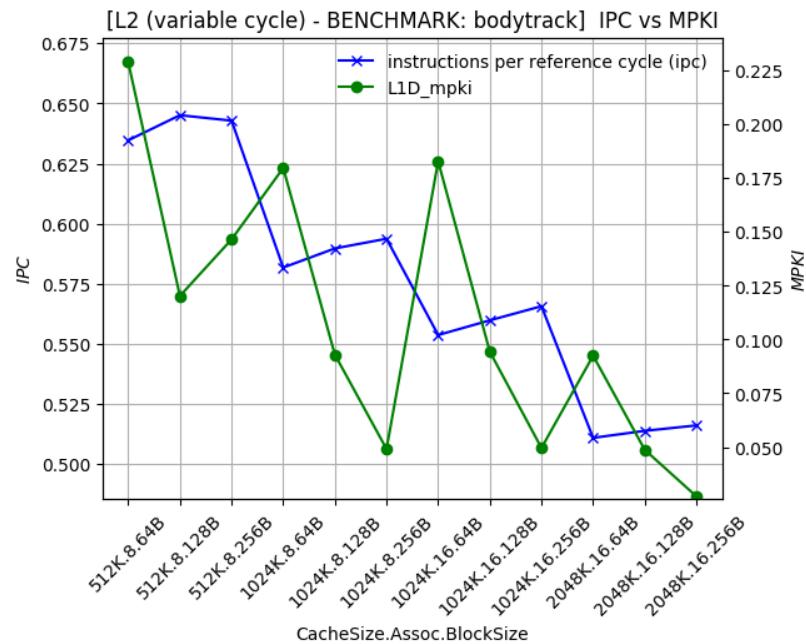
Είναι εμφανές ότι η μεταβολή των χαρακτηριστικών της L1 Cache επιδρά ή αρνητικά ή χωρίς καμία μεταβολή στην επίδοση, καθώς οι εντολές που εκτελούνται στον αρχικό κύκλο αναφοράς ή μειώνονται ή παραμένουν σταθερές. Επομένως σε κάθε περίπτωση, η βέλιτση επιλογή είναι η 32K.4.64B L1 Cache.

### 2.2.2 L2 Cache

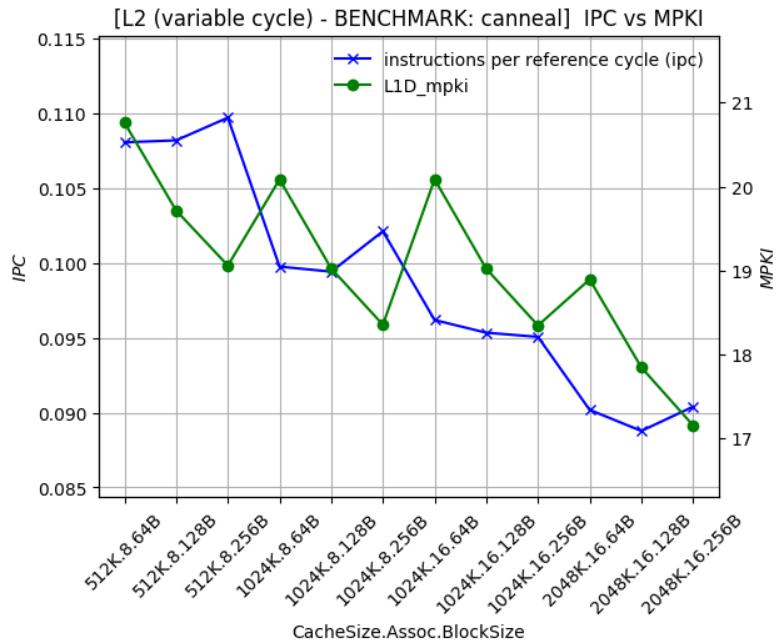
*blackscholes*



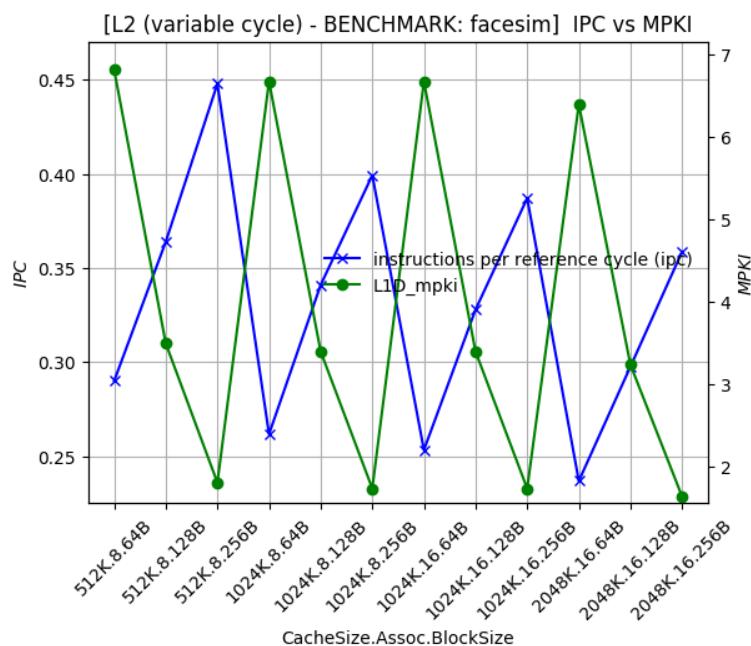
*bodytrack*



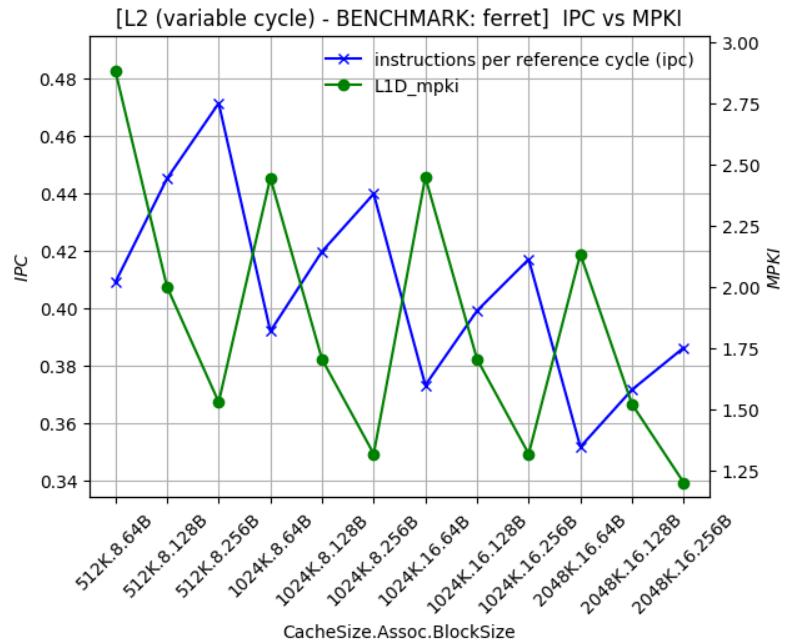
***canneal***



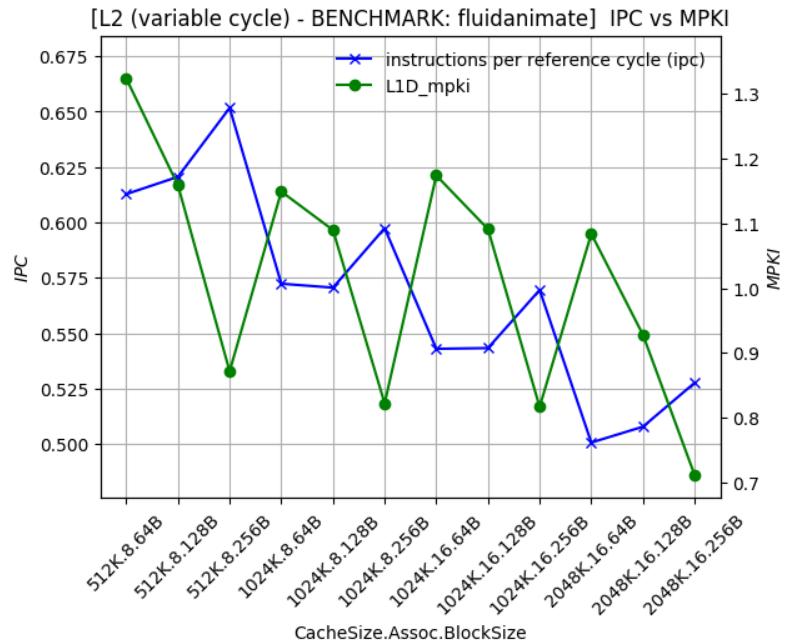
***facesim***



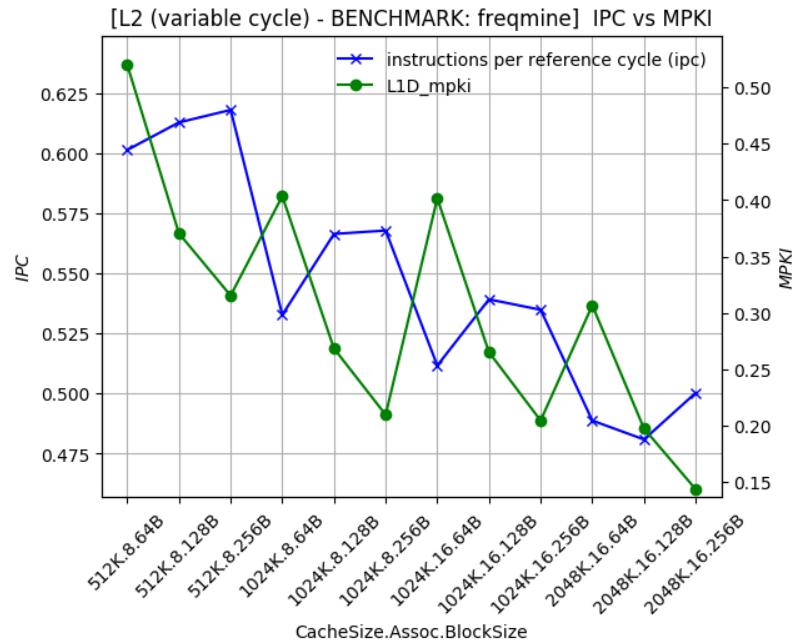
## *ferret*



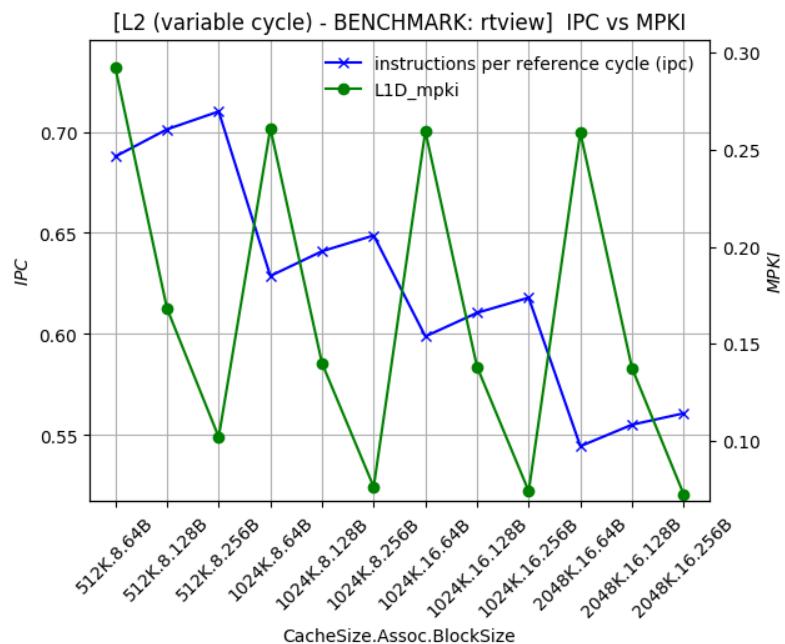
## *fluidanimate*



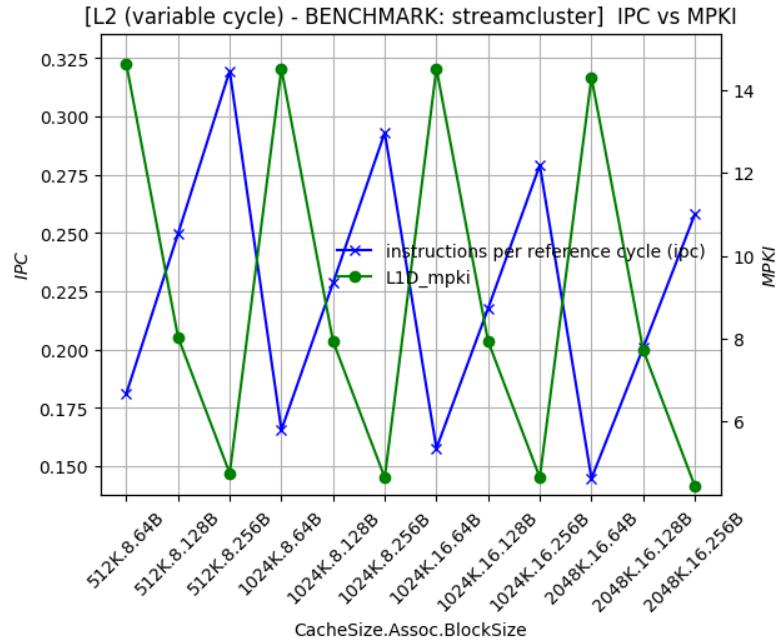
## ***freqmine***



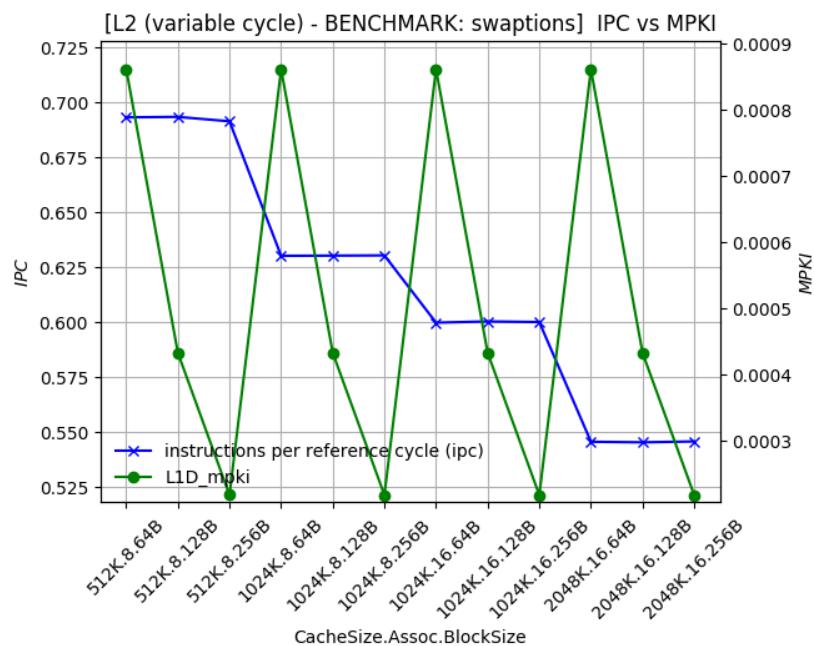
## ***rtview***



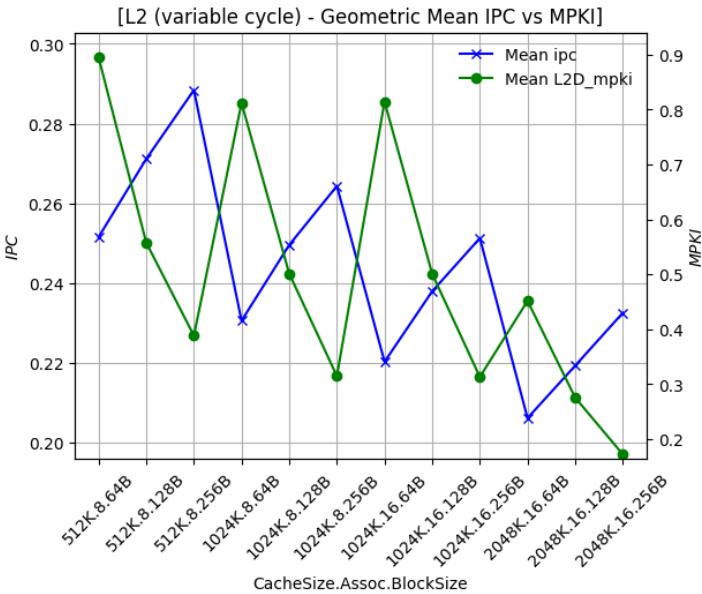
### ***streamcluster***



### ***swaptions***



## Geometric Average



### Συμπεράσματα για την L2 Cache

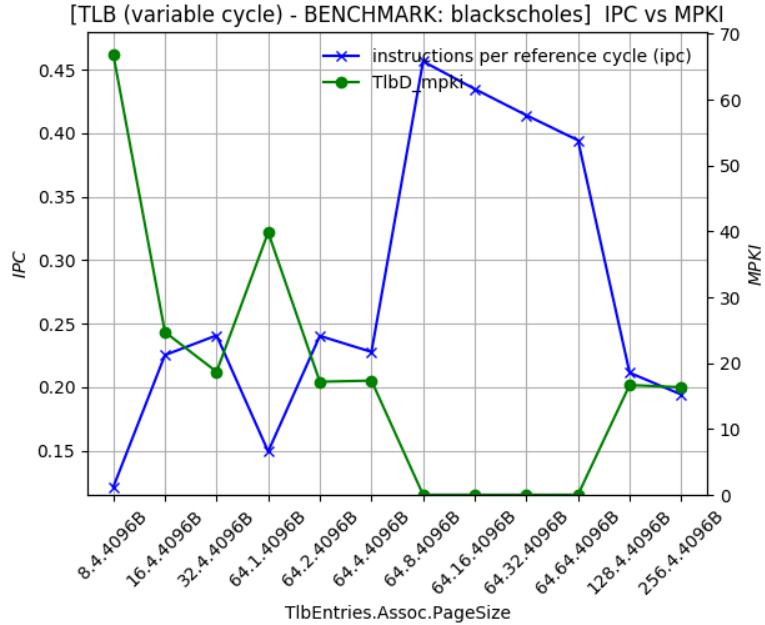
Όπως και για την L1, έτσι και για την L2 δεν είναι απαραίτητα αντιστρόφως ανάλογα τα μεγέθη IPC, MPKI. Καθώς, όπως αναφέραμε, πλέον το IPC αναφέρεται στον αρχικό κύκλο του πειράματος αναφοράς, πλέον ενώ μπορεί μεν να αυξάνονται οι εντολές που εκτελούνται στον νέο κύκλο, η διάρκεια του κύκλου λόγω μεταβολής ενδέχεται να είναι αρκετά μεγαλύτερη σε σχέση με τον κύκλο αναφοράς οπότε και η επίδοση, δηλαδή οι εντολές ανα χρονικό διάστημα να μην είναι καλή.

1. Σε όλα τα benchmarks βλέπουμε πως η επίδοση μειώνεται καθώς αυξάνεται το associativity ή το μέγεθος της L2. Αυτό συμβαίνει καθώς ο κύκλος του ρολογιού αυξάνεται πιο γρήγορα σε σχέση με το πόσες περισσότερες εντολές εκτελούνται στον νέο κύκλο.
2. Για σταθερά τα υπόλοιπα χαρακτηριστικά, η αύξηση του block size επιφέρει σε όλα τα benchmarks σημαντική βελτίωση της επίδοσης (εκτός από το benchmark swaptions που δεν επηρεάζεται).

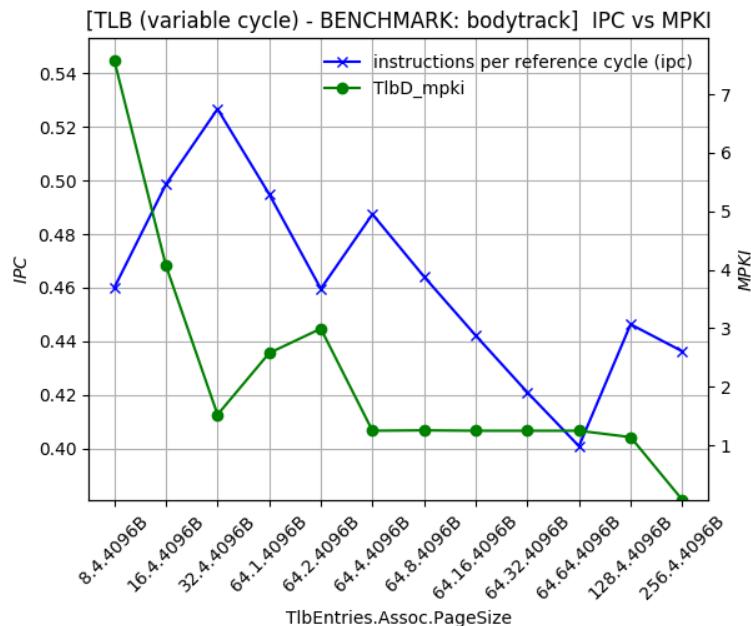
Από τα επιμέρους διαγράμματα του κάθε benchmark αλλά και από τον γεωμετρικό τους μέσο, είναι προφανές πως ο 3ος συνδυασμός, 512K.8.256B αποτελεί τη βέλτιστη δυνατή επιλογή για την L2 Cache.

### 2.2.3 TLB

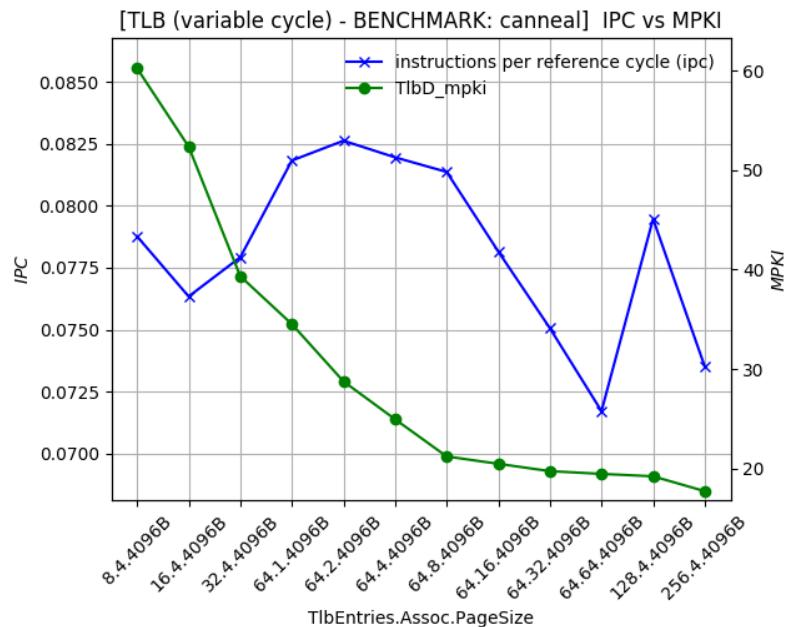
*blackscholes*



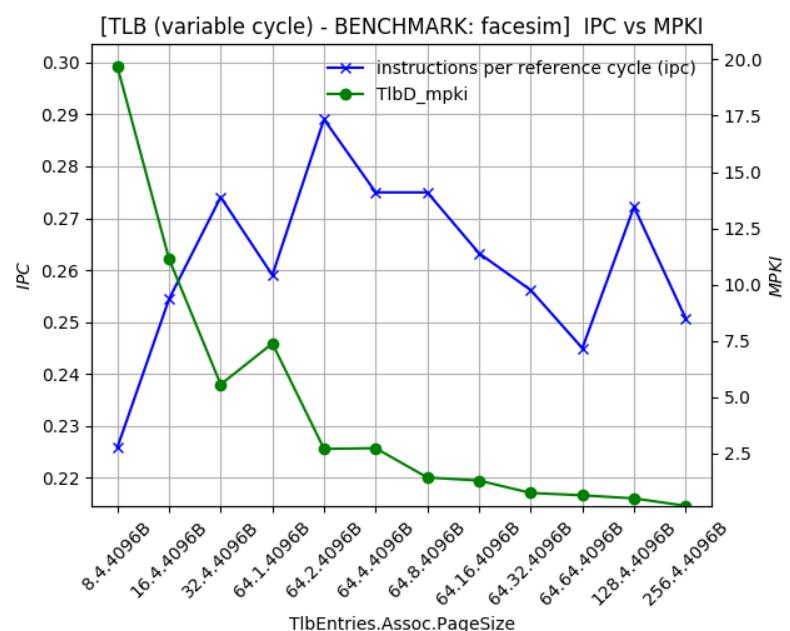
*bodytrack*



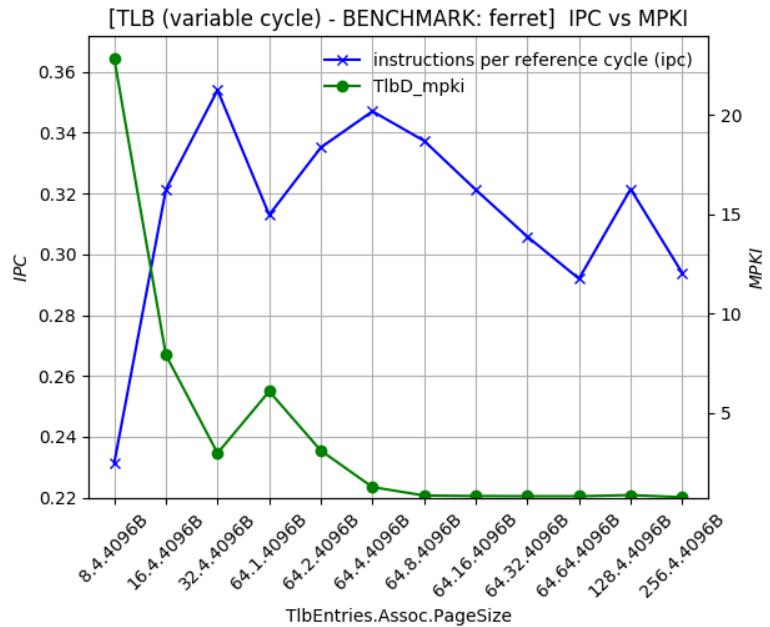
*canneal*



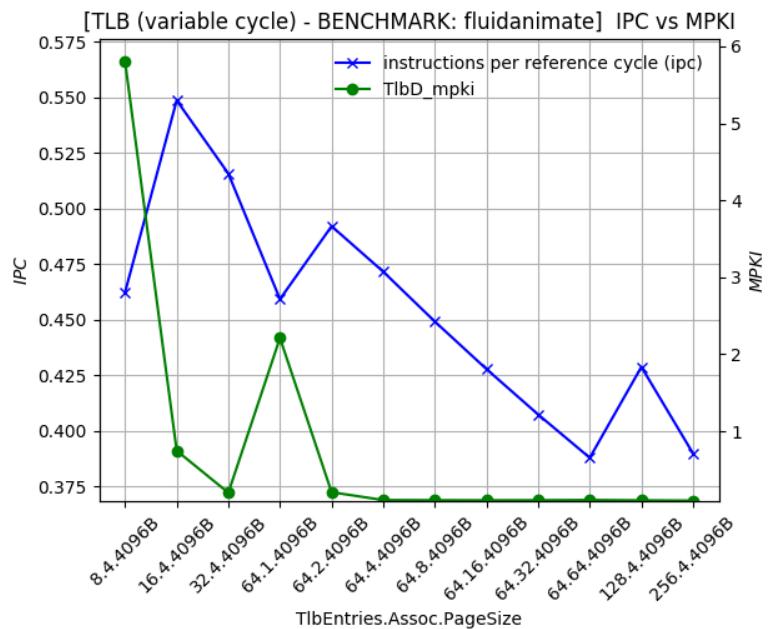
*facesim*



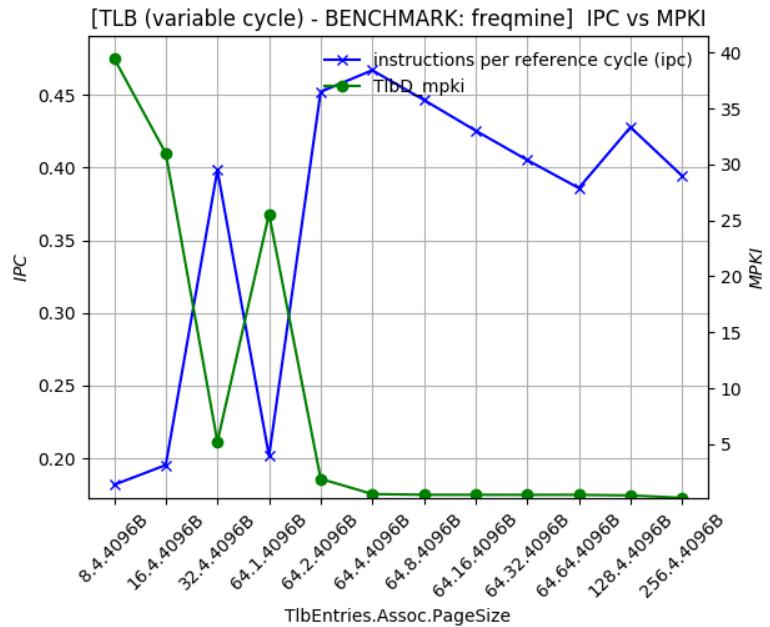
*ferret*



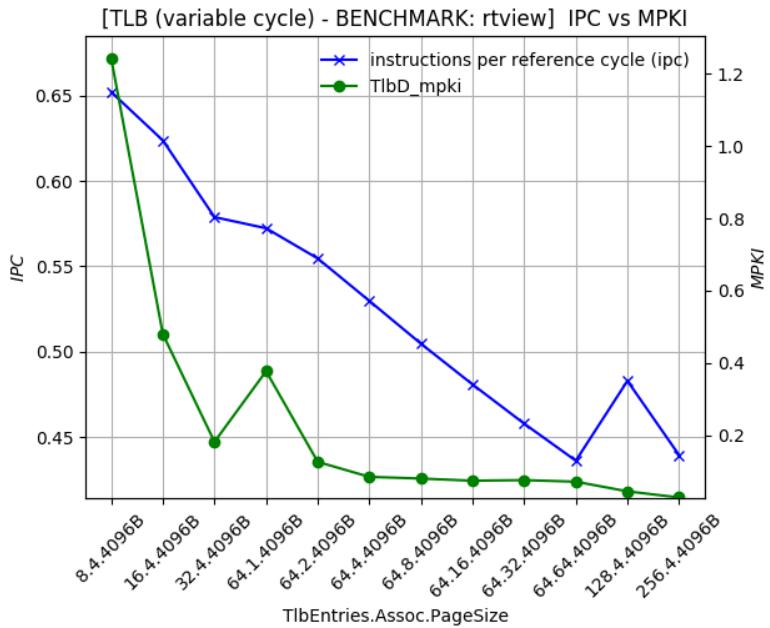
*fluidanimate*



***freqmine***

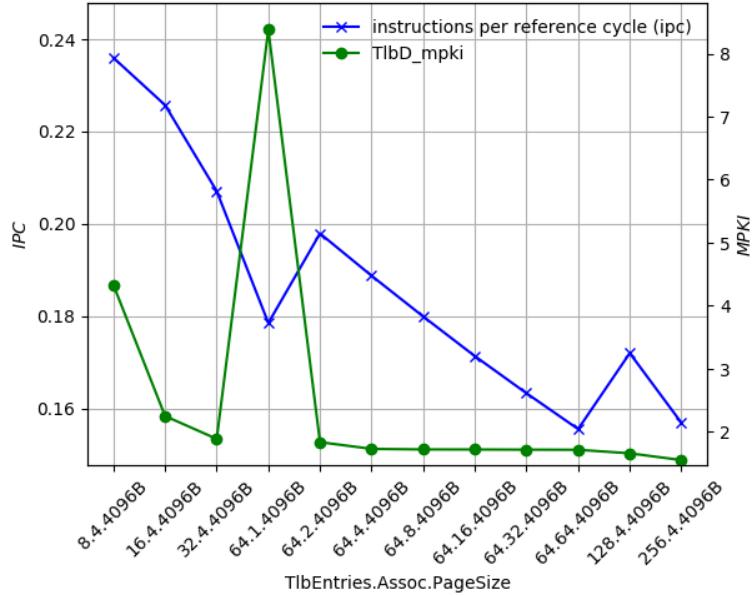


***rtview***



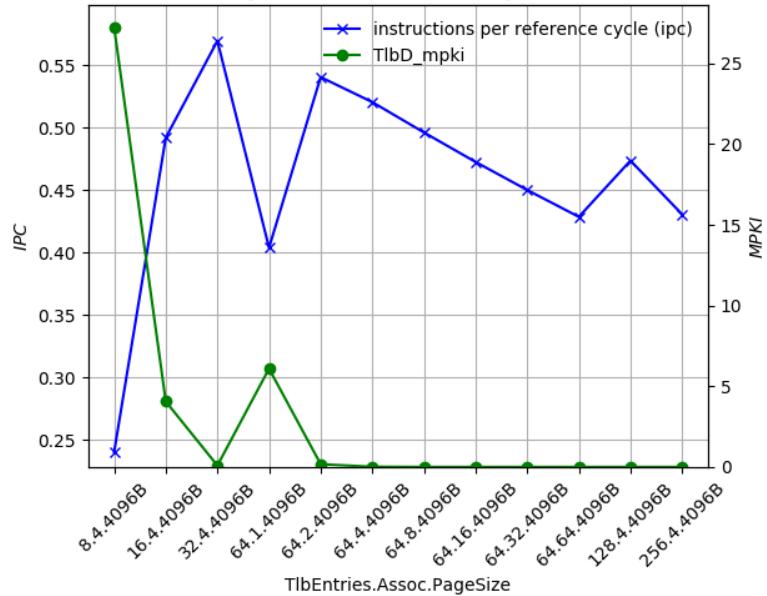
## *streamcluster*

[TLB (variable cycle) - BENCHMARK: streamcluster] IPC vs MPKI

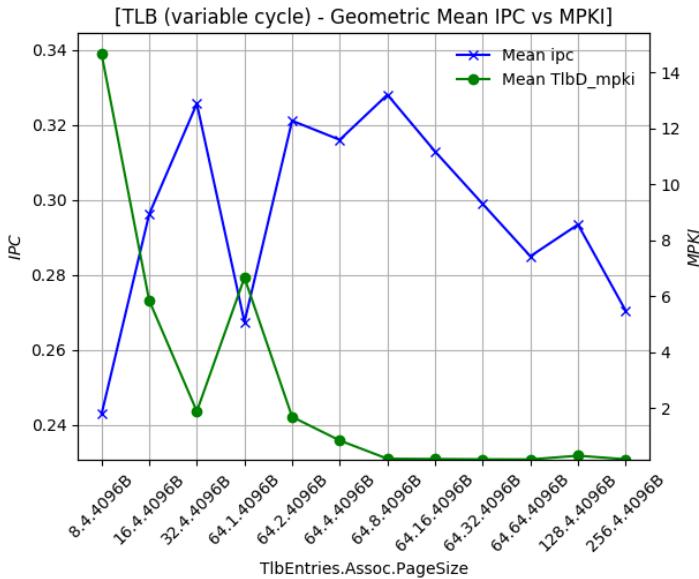


## *swaptions*

[TLB (variable cycle) - BENCHMARK: swaptions] IPC vs MPKI



## *Geometric Average*



## Συμπεράσματα για το TLB

Σε γενικές γραμμές, παρατηρούμε να μεταβάλλονται αντιστρόφως ανάλογα τα μεγέθη IPC, MPKI. Αυτό είναι λογικό αφού όσο περισσότερες αστοχίες γίνονται, τόσο χειρότερη επίδοση θα έχουμε (άρα και μικρότερο IPC) και αντίστροφα.

1. Σε όλα τα παραπάνω benchmarks, για μέγεθος TLB 64 η αλλαγή του associativity από 1 σε 2 ή 4 οδηγεί σε βελτίωση της επίδοσης κατά μέσο όρο. Από το σημείο αυτό και πέρα ωστόσο, για μέγεθος 64 κάθε διπλασιασμός του associativity οδηγεί σε μείωση της επίδοσης με γραμμικό τρόπο.
2. Στο swaptions η καλύτερη επίδοση επιτυγχάνεται για 32.4.4096B. Συγκεκριμένα για σταθερό το associativity = 4 και το Page Size, η αυξηση των TLB Entries από 8 μέχρι 32 οδηγεί σε βελτίωση της επίδοσης, ενώ από 32 σε 64 μειώνει την επίδοση.
3. Στα benchmark swaptions και rtview η καλύτερη επίδοση επιτυγχάνεται για το αρχικό configuration 8.4.4096B.

*Γενικότερα, κάθε benchmark επηρεάζεται με διαφορετικό τρόπο από τα χαρακτηριστικά της TLB. Με βάσει τα διαγράμματα των γεωμετρικών μέσων ωστόσο, μπορούμε να πούμε πως 2 σχετικά καλοί και προτιμητέοι συνδυασμοί είναι οι 32.4.4096, 64.2.4096 και 64.8.4096.*