

Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών

Άσκηση 1η

Αναστάσιος Στέφανος Αναγνώστου

19 Απριλίου 2023

Περιεχόμενα

1	L1 Cache	3
1.1	Blackscholes	3
1.2	Bodytrack	4
1.3	Canneal	5
1.4	Fluidanimate	5
1.5	Freqmine	6
1.6	Rtview	6
1.7	streamcluster	7
1.8	swaptions	8
2	L2 Cache	9
2.1	Blackscholes	9
2.2	Bodytrack	10
2.3	Canneal	11
2.4	Fluidanimate	11
2.5	Freqmine	12
2.6	Rtview και steamcluster	13
2.7	Swaptions	14
3	TLB	15
3.1	Blackscholes	15
3.2	Bodytrack	16
3.3	Canneal, fluidanimate, freqmine, rtview, streamcluster και swap- tions	16
4	Προανάκληση	18
4.1	Εκτελέσιμα	18
5	Προσαρμογή Κύκλου Ρολογιού	21
5.1	L1 Cache και L2 Cache	21
5.2	TLB	24
5.3	Προανάκληση	26
6	Παράρτημα	27
6.1	Κώδικας εκτέλεσης πειραμάτων	27
6.2	Κώδικας εξαγωγής IPC	35
6.3	Κώδικας παραγωγής γραφημάτων	36

Σημειώνεται, ότι πριν την έναρξη των πειραμάτων, πρέπει να πλοηγηθεί κανείς στο αρχείο `cache.h` και να εντοπίσει τους κατασκευαστές των αντικειμένων `cache`. Εκεί, πρέπει να αλλαχθούν οι προκαθορισμένες τιμές ποινής αστοχίας ώστε να ταυτίζονται με τις ζητούμενες από την εκφώνηση της άσκησης.

1 L1 Cache

Στο πρώτο πείραμα δοκιμάζονται διάφορες τιμές των παραμέτρων του πρώτου επιπέδου της κρυφής μνήμης, διατηρώντας σταθερές τις παραμέτρους του δευτέρου επιπέδου και του TLB. Συγκεκριμένα, δοκιμάζονται οι τιμές:

L1 size (KB)	L1 associativity	L1 block size (KB)
32	4	64
32	8	32, 64, 128
64	4	64
64, 128	8	32, 64, 128

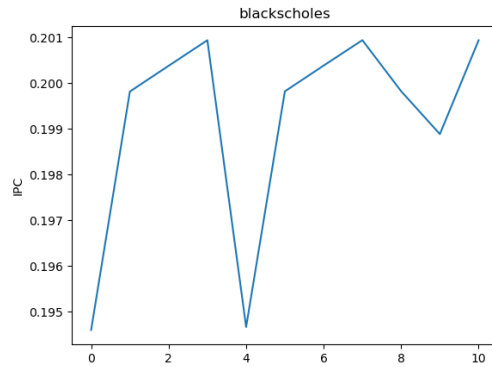
και διατηρούνται σταθερές οι τιμές:

L2 size (KB)	L2 associativity	L2 block size (KB)
1024	8	128
TLB size (entries)	TLB associativity	TLB page size (B)
64	4	4096

Παρακάτω, φαίνονται οι εντολές ανά κύκλο που επιτεύχθηκαν στα διάφορα μετροπρογράμματα με τις παραπάνω διατάξεις. Σημειώνεται ότι οι διατάξεις δοκιμάστηκαν και διατάσσονται στα γραφήματα όπως παρατίθενται παραπάνω.

1.1 Blackscholes

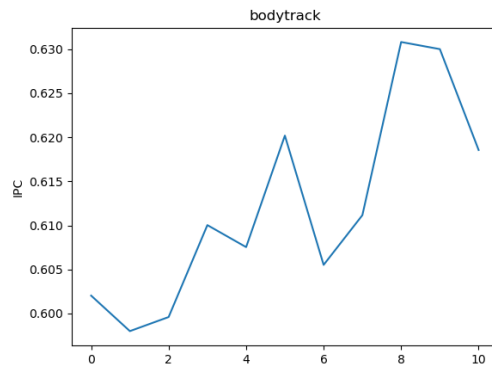
Επιθεωρώντας το γράφημα του εκτελέσιμου `blackscholes 1`, φαίνεται ότι η αύξηση της συσχετιστικότητας $4 \rightarrow 8$ συνεισφέρει στην αύξηση του IPC. Στην συνέχεια, η αύξηση του μεγέθους του μπλοκ συνεισφέρει επίσης στην αύξηση του IPC, αλλά όχι στον ίδιο βαθμό. Όταν επαναφέρεται η συσχετιστικότητα $8 \rightarrow 4$ και δοκιμάζεται διπλάσιο μέγεθος L1, η απόδοση πέφτει και αυξάνεται εκ νέου με την αύξηση της συσχετιστικότητας. Σε αυτό το σημείο είναι σαφές ότι, τουλάχιστον για αυτό το εκτελέσιμο, η συσχετιστικότητα 8 αποδίδει καλύτερα. Διπλασιάζοντας εκ νέου το μέγεθος L1, σταθερής της συσχετιστικότητας, δεν επιτυγχάνεται το ίδιο καλή απόδοση όπως προηγουμένως αλλά παρατηρείται η ίδια αύξηση του IPC μεγάλωνοντας το μέγεθος του μπλοκ. Σημειώνεται, βέβαια, ότι η απόδοση είναι εν πάση περιπτώσει μικρή, στο 20%.



Σχήμα 1: IPC στο εκτελέσιμο blackscholes συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.2 Bodytrack

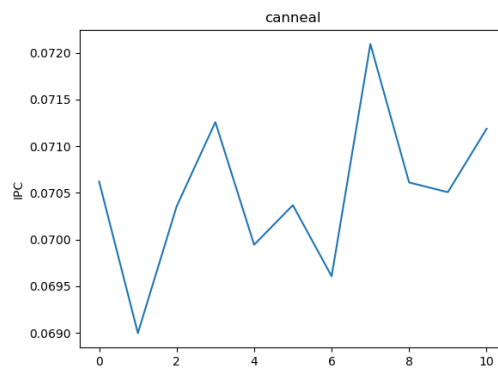
Στο εκτελέσιμο bodytrack παρατηρείται σαφώς καλύτερη απόδοση, τάξεως 60%. Εν προκειμένω, φαίνεται, ότι η αύξηση του μεγέθους L1 προκαλεί αύξηση στην απόδοση, η οποία ενισχύεται και από την αύξηση του μεγέθους μπλοκ, ίσως με εξαίρεση το μέγεθος cache 128 KB, όπου σημειώνεται μεν η μέγιστη απόδοση για συσχιστικότητα 8 αλλά σημειώνεται μείωση με την αύξηση του μεγέθους μπλοκ.



Σχήμα 2: IPC στο εκτελέσιμο bodytrack συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.3 Canneal

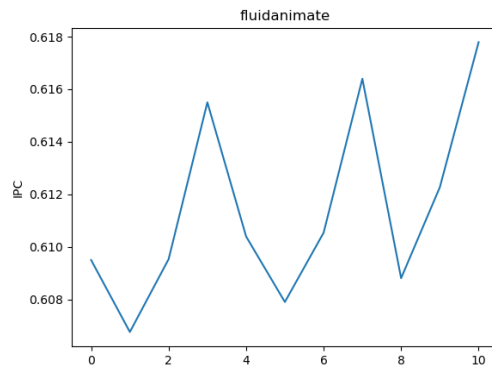
Στο γράφημα 3 διακρίνεται αρχικά εξαιρετικά χαμηλό IPC, τάξεως 1%. Στην συνέχεια, παρατηρείται πτώση με την αύξηση της συσχετιστικότητας. Τα ακραία μεγέθη της μνήμης L1 (32 KB, 128 KB) σημειώνουν καλύτερη απόδοση, ευνοούμενη περαιτέρω από την αύξηση του μεγέθους μπλοκ. Τέλος, παρατηρείται μία ιδιομορφία στα 64 KB χωρητικότητας μνήμης και 128B μπλοκ, όπου σημειώνεται ξαφνική αύξηση και η βέλτιστη απόδοση εκ των δοκιμασμένων παραμέτρων, παρότι προηγουμένως σημειώνονταν κακές αποδόσεις.



Σχήμα 3: IPC στο εκτελέσιμο canneal συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.4 Fluidanimate

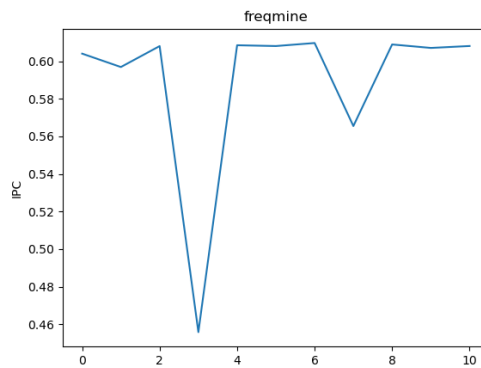
Στο γράφημα 4 διακρίνεται αρχικά σχετικά υψηλό IPC, τάξεως 60%. Παρατηρείται επαναλαμβανόμενη συμπεριφορά στην απόδοση. Συγκεκριμένα, η περιοδική αυτή συμπεριφορά φαίνεται να προκαλεί άμεση απεικόνιση της επίδρασης των μεταβολών του μεγέθους του μπλοκ. Οι τιμές του μεγέθους του μπλοκ μεταβάλλονται ως $64 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 64 \rightarrow 32 \dots$ και οι τιμές της απόδοσης ακολουθούν την μονοτονία του μεγέθους του μπλοκ, δηλαδή αύξηση οδηγεί σε αύξηση και μείωση σε μείωση. Η αύξηση ενισχύεται περαιτέρω από την αύξηση του μεγέθους της L1, για αυτό και παρατηρείται μέγιστη απόδοση στις μέγιστες τιμές 128 KB, 128 B.



Σχήμα 4: IPC στο εκτελέσιμο fluidanimate συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.5 Freqmine

Στο γράφημα 5 διακρίνεται αρχικά σχετικά υψηλό IPC, τάξεως 60%. Η απόδοση είναι αρκετά σταθερή παρόλες τις μεταβολές των παραμέτρων. Αξιοσημειώτες είαι δύο ιδιομορφίες, οι οποίες σημειώνονται αμφότερες όταν αυξάνεται το μέγεθος του μπλοκ σε 128B. Κατά τ'άλλα, επικρατεί σταθερότητα.

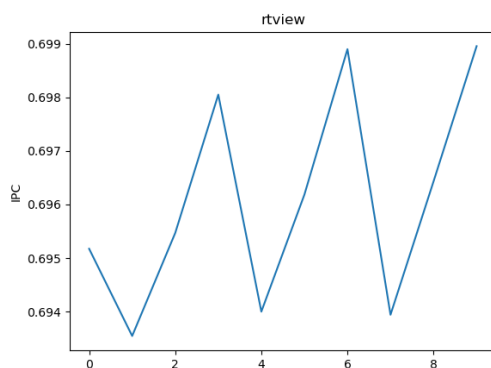


Σχήμα 5: IPC στο εκτελέσιμο freqmine συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.6 Rtview

Στο γράφημα 6 παρατηρείται παρόμοια συμπεριφορά με το γράφημα 4, τόσο στην περιοδικότητα όσο και στην τάξη μεγέθους, η οποία είναι ελαφρώς μεγαλύτερο στο

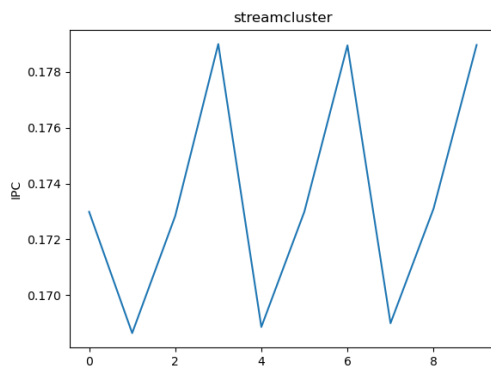
70%. Παρατηρείται, πάλι, μέγιστη απόδοση με την αύξηση του μεγέθους της L1 και του μεγέθους του μπλοκ.



Σχήμα 6: IPC στο εκτελέσιμο rtview συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.7 streamcluster

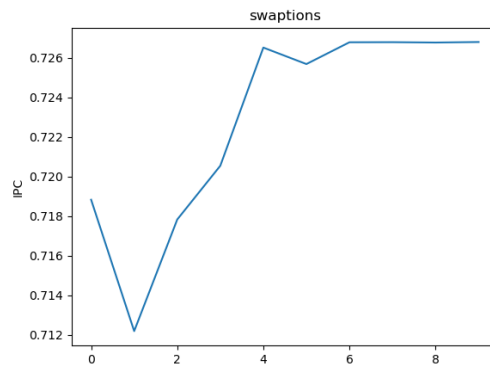
Στο γράφημα 7 παρατηρείται παρόμοια συμπεριφορά με τα γραφήματα 4 και 6, μόνο στην περιοδικότητα, όμως, και όχι στην υάξη μεγέθους, η οποία είναι χαμηλή στο 20%. Παρατηρείται, πάλι, μέγιστη απόδοση με την αύξηση του μεγέθους της L1 και του μεγέθους του μπλοκ.



Σχήμα 7: IPC στο εκτελέσιμο streamcluster συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

1.8 swaptions

Στο γράφημα 8 σημειώνεται απόδοση τάξεως 70%. Σαφής είναι η εξάρτηση από το μέγεθος της L1 και το μέγεθος μπλοκ, τα οποία προκαλούν αύξηση καθώς τα ίδια αυξάνονται. Η αύξηση της συσχετιστικότητας φαίνεται να προκαλεί μείωση όταν το μέγεθος της μνήμης είναι μικρό, φαινόμενο ακόμα ασθενέστερο για μεγαλύτερα μεγέθη μνήμης και μπλοκ. Παρατηρείται, πάλι, μέγιστη απόδοση με την αύξηση του μεγέθους της L1 και του μεγέθους του μπλοκ.



Σχήμα 8: IPC στο εκτελέσιμο swaptions συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

Κάτι αξιοσημείωτο είναι, ότι παρόλες τις διάφορες διακυμάνσεις, δεν προκαλείται αλλαγή στην τάξη μεγέθους της απόδοσης, καθώς διατηρείται η πρώτη σχετικά σταθερή. Επίσης, η τάση να αυξημένης απόδοσης με συσχετιστικότητα 8 πιθανώς οφείλεται και στην ταύτιση με την συσχετιστικότητα του δευτέρου επιπέδου cache.

2 L2 Cache

Στο δεύτερο πείραμα, δοκιμάζονται διάφορες τιμές των παραμέτρων του δευτέρου επιπέδου της κρυφής μνήμης, διατηρώντας σταθερές τις παραμέτρους του πρώτου επιπέδου της κρυφής μνήμης και του TLB. Συγκεκριμένα, δοκιμάζονται οι τιμές:

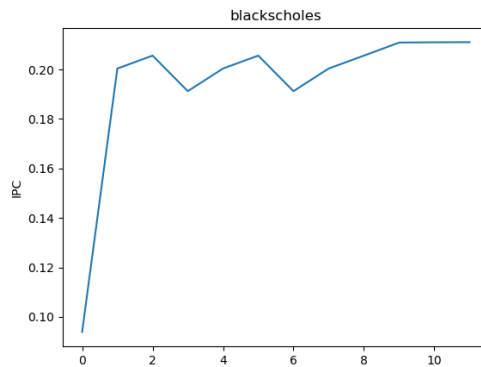
L2 size (KB)	L2 associativity	L2 block size (KB)
512	8	64, 128, 256
1024	8, 16	64, 128, 256
2048	16	64, 128, 256

και διατηρούνται σταθερές οι τιμές:

L1 size (KB)	L1 associativity	L1 block size (KB)
32	8	64
TLB size (entries)	TLB associativity	TLB page size (B)
64	4	4096

2.1 Blacksholes

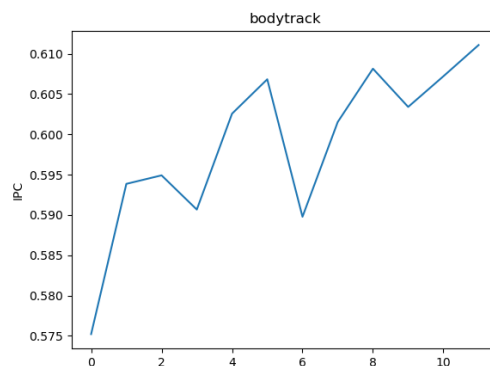
Επιθεωρώντας το γράφημα του εκτελέσιμου blacksholes 9, φαίνεται ότι η αύξηση της απόδοσης εξαρτάται σε μεγάλο βαθμό από το μέγεθος του μπλοκ. Συγκεκριμένα, παρότι η απόδοση με το πρώτο σύνολο παραμέτρων (μέγεθος μπλοκ 64 B) ήταν άξεως 10%, αμέσως η αύξηση του μεγέθους μπλοκ οδήγησε σε διπλασιασμό της και η περεταίρω αύξησή του ενέτεινε ακόμα, όχι στον ίδιο βαθμό, την απόδοση. Η αλλαγή του μεγέθους της μνήμης (και η επαναφορά του μικρού μεγέθους μπλοκ) έριξε την απόδοση αλλά οι ακόλουθες αυξήσεις την επανέφεραν. Το ίδιο μοτίβο παρατηρείται και στην συνέχεια, με μέγιστη απόδοση να επιτυγχάνεται τελικά για τις μέγιστες τιμές χωρητικότητας και μεγέθους μπλοκ.



Σχήμα 9: IPC στο εκτελέσιμο blackscholes συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.2 Bodytrack

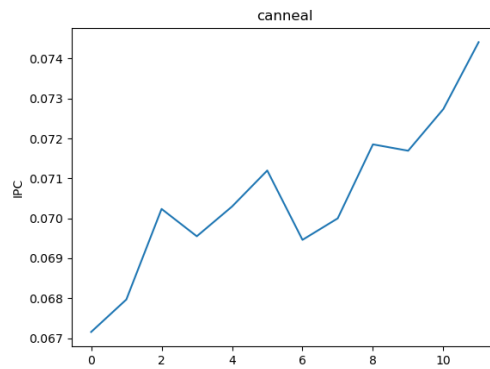
Στο εκτελέσιμο bodytrack παρατηρείται σαφώς καλύτερη απόδοση, τάξεως 60%. Η αύξηση του μεγέθους του μπλοκ πάλι φαίνεται να ευνοεί την απόδοση, αφού παρατηρούνται αυξήσεις της όταν αυτό αυξάνεται αλλά και απότομες πτώσεις όταν αυτό λαμβάνει την ελάχιστη τιμή του. Συνολικά, η απόδοση φαίνεται να ευνοείται και από της χωρητικότητας και της συσχετιστικότητας, αφού έχει γενικώς αυξητική τάση. Η μέγιστη απόδοση λαμβάνεται για μέγιστες τιμές συσχετιστικότητας, χωρητικότητας και μεγέθους μπλοκ, με εμφανή διαφορά.



Σχήμα 10: IPC στο εκτελέσιμο bodytrack συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.3 Canneal

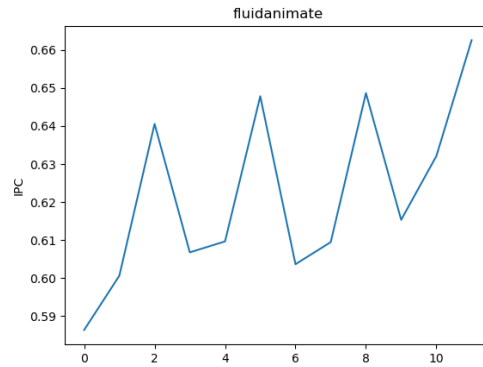
Στο γράφημα 11 διακρίνεται αρχικά εξαιρετικά χαμηλό IPC, τάξεως μικρότερης του 1%. Ακόμα μια φορά, παρατηρείται γενικώς αυξητική τάση. Η αύξηση του μεγέθους μπλοκ του δευτέρου επιπέδου ευνοεί, γεγονός φανερό από τις πτώσεις της απόδοσης στην επαναφορά του μεγέθους του μπλοκ στην ελάχιστη τιμή. Η γενικώς αυξητική τάση δείχνει ότι η απόδοση ευνοείται επίσης από την αύξηση του μεγέθους του δευτέρου επιπέδου μνήμης. Η αύξηση της συσχετιστικότητας ενδεχομένως να έχει θετική επίδραση, αλλά η επίδραση του μεγέθους της μνήμης είναι μεγαλύτερη. Αυτό φαίνεται από την απότομη αύξηση της απόδοσης στο τέλος, όπου η συσχετιστικότητα παραμένει σταθερή στο 16 ενώ το μέγεθος της μνήμης διπλασιάζεται.



Σχήμα 11: IPC στο εκτελέσιμο canneal συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.4 Fluidanimate

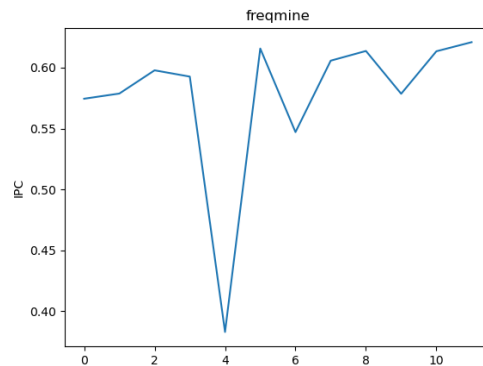
Στο γράφημα 12 διακρίνεται αρχικά σχετικά υψηλό IPC, τάξεως 60%. Παρατηρείται επαναλαμβανόμενη συμπεριφορά στην απόδοση. Συγκεκριμένα, η περιοδικότητα φαίνεται να οφείλεται, πάλι, στις τιμές του μεγέθους μπλοκ. Αυτήν την φορά, η επίδραση της χωρητικότητας της μνήμης δεν είναι τόσο ισχυρή, αφού η απόδοση λαμβάνει τις ίδιες τιμές παρόλες τις αυξήσεις της χωρητικότητας. Η αύξηση της συσχετιστικότητας φαίνεται να συνεισφέρει σε έναν μικρό βαθμό. Η μέγιστη τιμή απόδοσης λαμβάνεται για μέγιστες τιμές μεγέθους μπλοκ, συσχετιστικότητας και χωρητικότητας.



Σχήμα 12: IPC στο εκτελέσιμο fluidanimate συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.5 Freqmine

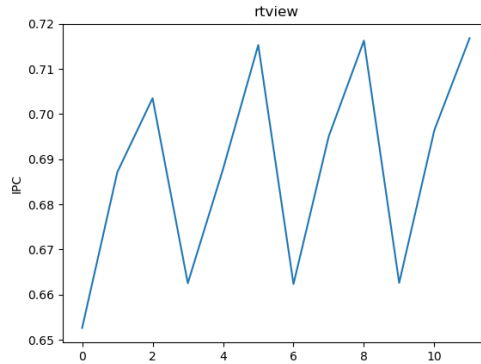
Στο γράφημα 13 διακρίνεται αρχικά σχετικά υψηλό IPC, τάξεως 50-60%. Στην αρχή παρουσιάζεται μία ελαφρώς αυξητική τάση προκαλούμενη από την αύξηση του μεγέθους του μπλοκ δευτέρου επιπέδου. Παρατηρείται μία μικρή μείωση όταν αυξάνεται η χωρητικότητα και επαναφέρεται στο ελάχιστο μέγεθος το μπλοκ, αλλά όχι μεγαλύτερη από την προηγούμενη συνολική αύξηση, πράγμα που σημαίνει ότι και η αύξηση της χωρητικότητας συμβάλει στην αύξηση της απόδοσης. Ωστόσο, στις τιμές παραμέτρων 1024 KB, 8, 128 B, παρατηρείται μία ιδιομορφία και μειώνεται πολύ απότομα η απόδοση, επαναφερόμενη στην αναμενόμενη τιμή αμέσως. Η μέγιστη απόδοση επιτυγχάνεται, για ακόμα μία φορά, στις μέγιστες τιμές χωρητικότητας, συσχέτισης και μεγέθους μπλοκ.



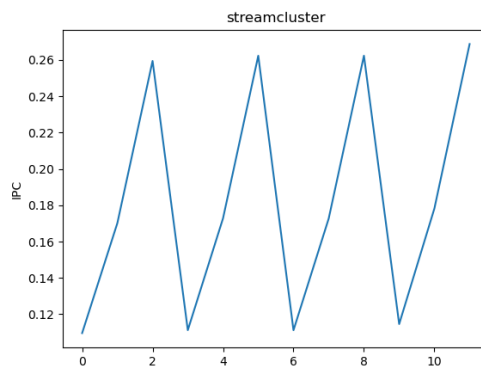
Σχήμα 13: IPC στο εκτελέσιμο freqmine συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.6 Rtview και steamcluster

Τα εκτελέσιμα rtview και steamcluster παρουσιάζονται μαζί γιατί έχουν όμοια συμπεριφορά. Συγκεκριμένα, με εξαίρεση την τάξη μεγέθους του IPC, αντιστοίχως 70% και 20%, η συμπεριφορά τους στις μεταβολές των παραμέτρων είναι σχεδόν πανομοιότυπη. Συγκεκριμένα, αμφότερα φαίνονται να εξαρτώνται σε μεγάλο βαθμό από το μέγεθος του μπλοκ. Αξιοσημείωτη είναι η συμπεριφορά του steamcluster, το οποίο φαίνεται να είναι εντελώς ανεπηρέαστο από τις μεταβολές στην χωρητικότητα και την συσχέτιστικότητα της μνήμης. Το δε rtview φαίνεται να εξαρτάται σε κάποιον βαθμό από την χωρητικότητα και την συσχέτιστικότητά, αλλά όχι τόσο όσο από το μέγεθος μπλοκ, αφού γρήγορα σταθεροποιείται το εύρος των τιμών του IPC. Το μέγιστη IPC επιτυγχάνεται για μέγιστο μέγεθος μπλοκ, δηλαδή 256 B, και χωρητικότητα μνήμης διάφορη των 512 KB.



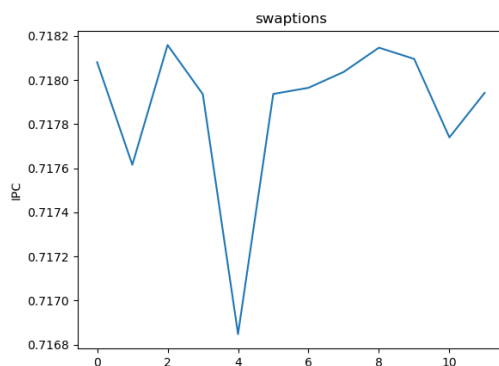
(α') IPC στο εκτελέσιμο rtview συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης



(β') IPC στο εκτελέσιμο steamcluster συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

2.7 Swaptions

Στο γράφημα 15 σημειώνεται απόδοση τάξεως 70%, με πάρα πολύ μικρές διακυμάνσεις. Οι διακυμάνσεις είναι τόσο μικρές ώστε να μην έχει και τόσο νόημα να μιλήσει κανείς για εξάρτηση από κάποια παράμετρο. Ακόμα και η ιδιομορφία, αν και οπτικά φανερή, προκαλεί στο IPC μία μεταβολή τάξεως 0.1%. Δεν φαίνεται να υπάρχει σαφής εξάρτηση.



Σχήμα 15: IPC στο εκτελέσιμο swaptions συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

3 TLB

Στο τρίτο πείραμα, δοκιμάζονται διάφορες τιμές των παραμέτρων του TLB, διατηρώντας σταθερές τις τιμές των παραμέτρων του πρώτου και δευτέρου επιπέδου της κρυφής μνήμης. Συγκεκριμένα, δοκιμάζονται οι τιμές:

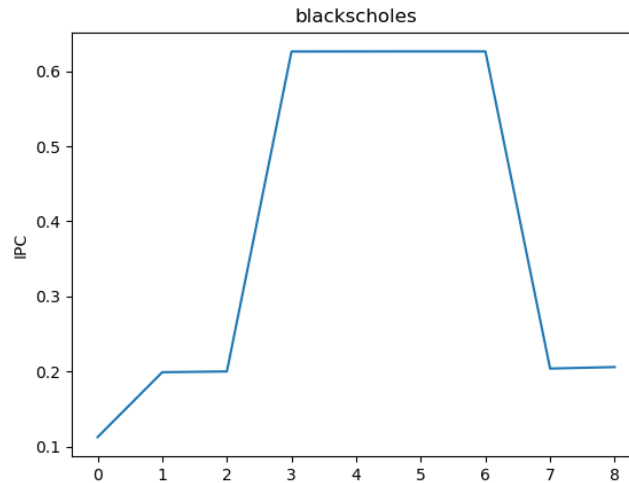
TLB size (entries)	TLB associativity	TLB page size (B)
64	1, 2, 4, 8, 16, 32, 64	4096
128, 256	4	4096

και διατηρούνται σταθερές οι τιμές:

L1 size (KB)	L1 associativity	L1 block size (KB)
32	8	64
L2 size (KB)	L2 associativity	L2 block size (KB)
1024	8	128

3.1 Blackscholes

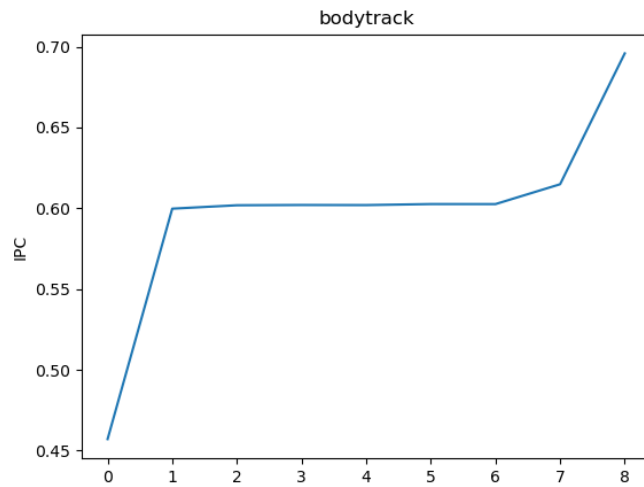
Επιθεωρώντας το γράφημα του εκτελέσιμου blackscholes 16, φαίνεται ότι το IPC αυξάνεται με την αύξηση της συσχετιστικότητας μέχρι και 4. Μετά μένει υψηλό και σταθερό μέχρι και την τιμή 64. Στην συνέχεια, η αύξηση του πλήθους των καταχωρήσεων το ρίχνει κατά πολύ.



Σχήμα 16: IPC στο εκτελέσιμο blackscholes συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

3.2 Bodytrack

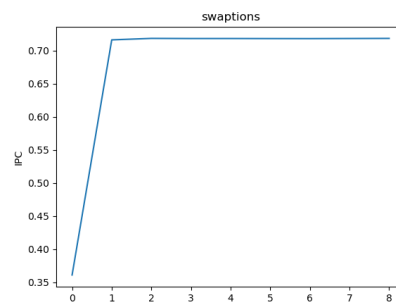
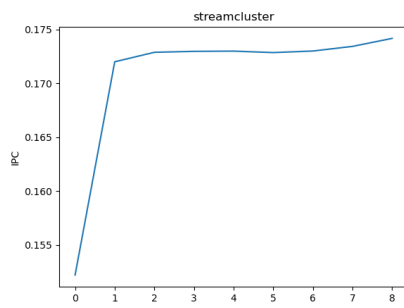
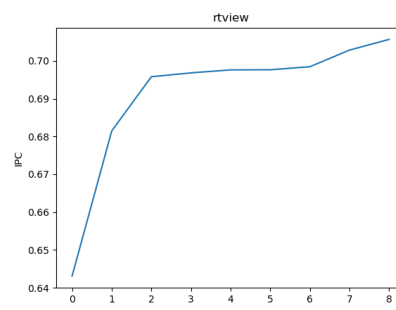
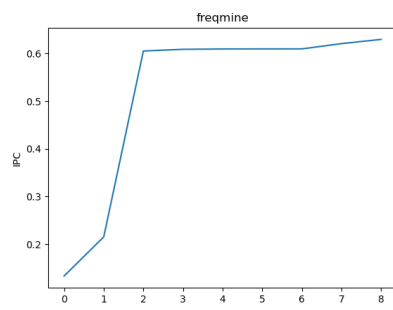
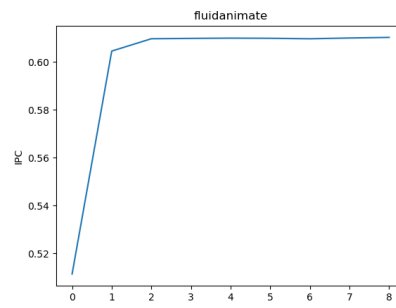
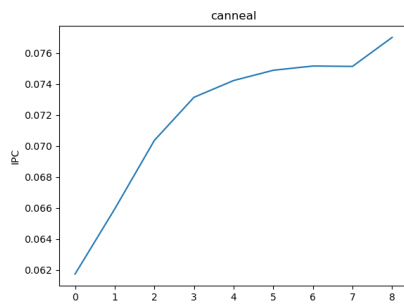
Στο εκτελέσιμο bodytrack παρατηρείται αυξητική τάση του IPC με την συσχέτιστικότητα, μέχρι και την τιμή 2. Μετά, παραμένει σταθερό σχεδόν μέχρι και την τιμή 64. Τέλος, η αύξηση του πλήθους των καταχωρήσεων προκαλεί την σημαντική του αύξηση.



Σχήμα 17: IPC στο εκτελέσιμο bodytrack συναρτήσει διαφόρων παραμέτρων της ιεραρχίας μνήμης

3.3 Canneal, fluidanimate, freqmine, rtview, streamcluster και swaptions

Τα εκτελέσιμα παρουσιάζονται μαζί γιατί έχουν πανομοιότυπη συμπεριφορά. Γενικώς, το IPC εξαρτάται έντονα από την συσχέτιστικότητα, καθώς παρατηρείται έντονη αυξητική μεταβολή του ακόμα και σε μικρές μεταβολές της. Συγκεκριμένα, σχεδόν σε όλες τις περιπτώσεις, μέχρι και την συσχέτιστικότητα 4 έχει σημειωθεί ήδη πολύ μεγάλη αύξηση του IPC και οι υπόλοιπες μεταβολές προκαλούν μικροδιακυμάνσεις. Ελαφρώς διαφορετική συμπεριφορά έχει το εκτελέσιμο canneal, το οποίο παρουσιάζει αύξουσα συμπεριφορά μέχρι και την συσχέτιστικότητα 32, πέφτει ελαφρώς στην 64 και μετά αυξάνεται εκ νέου λόγω της αύξησης των καταχωρήσεων.



4 Προανάκληση

Στο τέταρτο πείραμα, διατηρούνται σταθερές οι παράμετροι όλης της ιεραρχίας μνήμης και υλοποιείται προανάκληση NEXT-N-LINE. Συγκεκριμένα, οι παράμετροι είναι οι ακόλουθες:

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

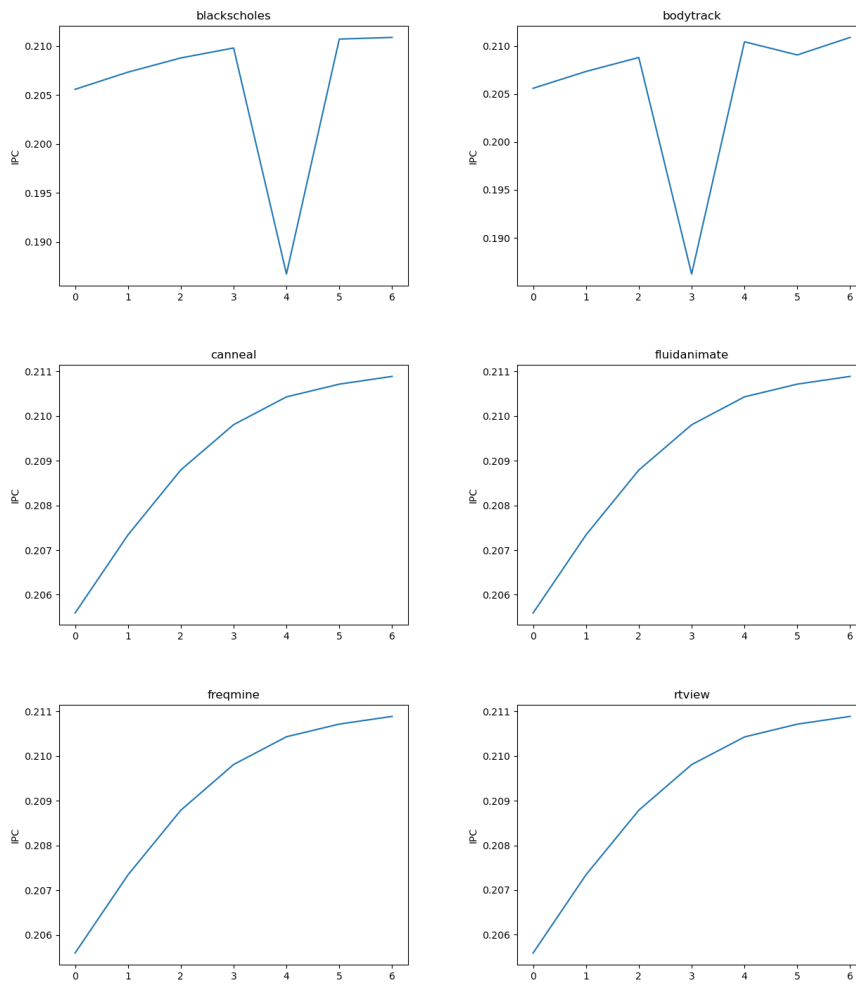
και συμπληρώνεται ο ακόλουθος κώδικας στο αρχείο `cache.h`.

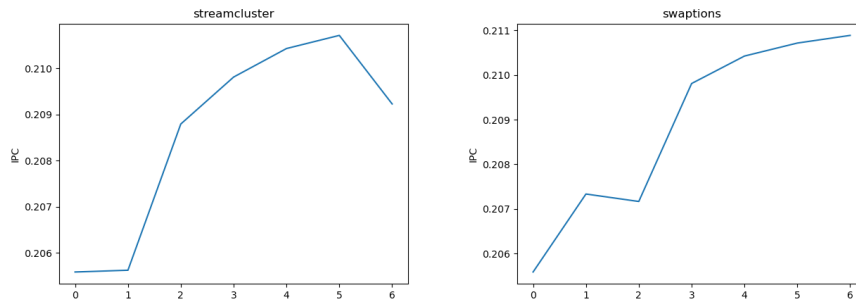
```
1 // PREFETCHING
2 ADDRINT prefetch_addr = addr;
3 for (UINT32 i=0; i < _l2_prefetch_lines; i++) {
4     prefetch_addr += L2BlockSize();
5     /* ..... */
6     /* Add here prefetching code. */
7     /* ..... */
8     CACHETAG prefetch_tag;
9     UINT32 prefetch_set_index;
10    SplitAddress(prefetch_addr, L2LineShift(), L2SetIndexMask(),
11                prefetch_tag, prefetch_set_index);
12    SET& prefetch_set = _l2_sets[prefetch_set_index];
13    bool prefetch_hit = prefetch_set.Find(prefetch_tag);
14    if (!prefetch_hit) {
15        CACHETAG l2_replaced = prefetch_set.Replace(prefetch_tag);
16        // If L2 is inclusive and a TAG has been replaced we need
17        // to remove
18        // all evicted blocks from L1.
19        if ((L2_INCLUSIVE == 1) && !(l2_replaced == INVALID_TAG)) {
20            ADDRINT replacedAddr = ADDRINT(l2_replaced) << FloorLog2(
21                L2NumSets());
22            replacedAddr = replacedAddr | prefetch_set_index;
23            replacedAddr = replacedAddr << L2LineShift();
24            for (UINT32 i=0; i < L2BlockSize(); i+=L1BlockSize()) {
25                ADDRINT newAddr = replacedAddr | i;
26                SplitAddress(newAddr, L1LineShift(), L1SetIndexMask(),
27                            l1Tag, l1SetIndex);
28                l1Set = _l1_sets[l1SetIndex];
29                l1Set.DeleteIfPresent(l1Tag);
30            }
31        }
32    }
33 }
```

4.1 Εκτελέσιμα

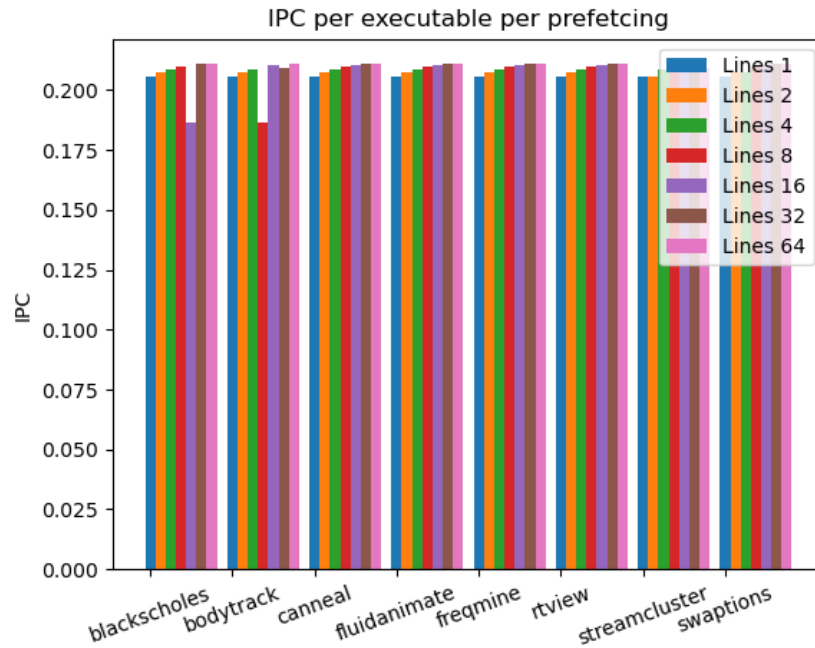
Αρχικά, παρατηρείται ότι η τάξη μεγέθους του IPC, σε όλα τα εκτελέσιμα, είναι περίπου 20%. Γενικώς, η αύξηση του πλήθους προανακλησόμενων γραμμών

προκαλεί αύξηση στο IPC. Μόνο στα εκτελέσιμα blackscholes, bodytrack και swaptions παρατηρούνται λίγες ανωμαλίες, όπως οι απότομες πτώσεις και η άμεση ανάκαμψη στα δύο πρώτα εκτελέσιμα και η ηπιότερη πτώση και άμεση ανάκαμψη στο τελευταίο.





Τα συμπεράσματα συνοψίζονται και στο παρακάτω γράφημα:



Σχήμα 18: Μέση τιμή IPC ανά πλήθος προανακληνόμενων γραμμών ανά πρόγραμμα

Στο γράφημα 18 φαίνεται η μέση τιμή του IPC ανά πλήθος προανακληνόμενων γραμμών ανά πρόγραμμα. Αποτυπώνεται η παραπάνω διαπίστωση, ότι δηλαδή όσο περισσότερες γραμμές προανακαλούνται τόσο μεγαλύτερο το IPC.

5 Προσαρμογή Κύκλου Ρολογιού

Προηγούμενος υποτέθηκε ότι οι αλλαγές στην αρχιτεκτονική της ιεραρχίας μνήμης δεν επιφέρουν αλλαγές στην διάρκεια του κύκλου του επεξεργαστή. Αυτό, όμως, δεν ισχύει. Για αυτόν τον λόγο, για καθέναν διπλασιασμό χωρητικότητας θα υποτεθεί 15% αύξηση της διάρκειας του κύκλου, ενώ για κάθε διπλασιασμό της συσχετιστικότητας 5% αύξηση.

Έστω, D, C, T η χρονική διάρκεια του προγράμματος, το πλήθος των κύκλων και η διάρκεια του κύκλου αντιστοίχως:

$$\begin{aligned} D &= C * T \\ IPC &= \frac{I}{C} = \frac{I * T}{D} = T * \frac{I}{D} \end{aligned} \quad (5.1)$$

Επιχειρείται ο υπολογισμός του προσαρμοσμένου IPC, για έναν διπλασιασμό χωρητικότητας.

$$\begin{aligned} D_{adj} &= C * adj(T) = C * (T + \frac{15}{100} * T) \Leftrightarrow \\ D_{adj} &= C * T * (1 + \frac{15}{100}) = T * (C + C * \frac{15}{100}) \Leftrightarrow \\ D_{adj} &= T * adj(C) \Rightarrow \\ IPC_{adj} &= T * \frac{I}{D_{adj}} = T * \frac{I}{T * adj(C)} \Leftrightarrow \\ IPC_{adj} &= \frac{I}{adj(C)} = \frac{I}{C * (1 + \frac{15}{100})} \Leftrightarrow \\ IPC_{adj} &= \frac{100}{115} * IPC \end{aligned} \quad (5.2)$$

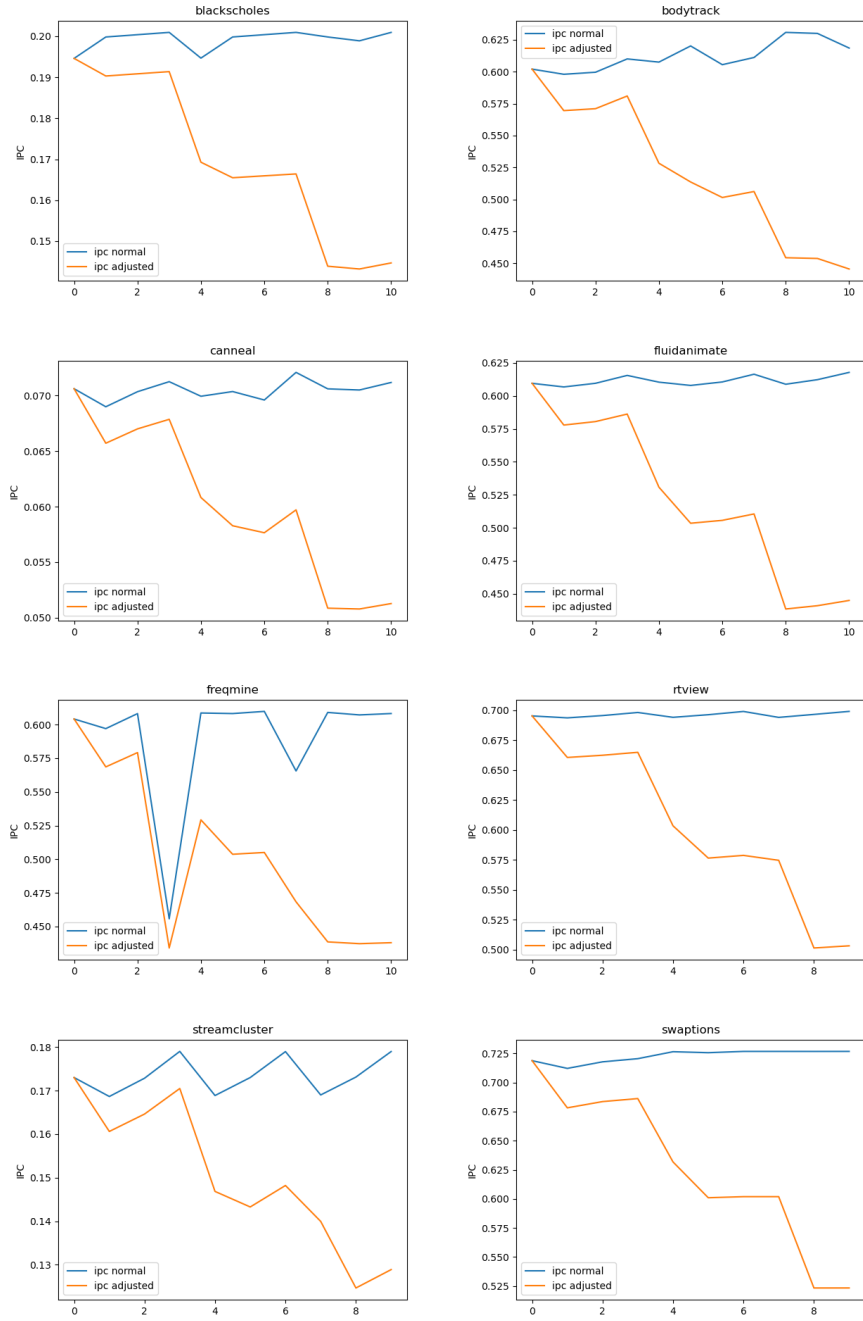
Άρα, αν n είναι το πλήθος των διπλασιασμών χωρητικότητας και m το πλήθος διπλασιασμών συσχετιστικότητας, το προσαρμοσμένο IPC είναι:

$$IPC_{adj} = \left(\frac{100}{105}\right)^n \left(\frac{100}{115}\right)^m IPC \quad (5.3)$$

Επομένως, είναι δυνατόν να προσαρμοστούν οι προηγούμενες μετρήσεις απλώς κλιμακώνοντάς τες καταλλήλως.

5.1 L1 Cache και L2 Cache

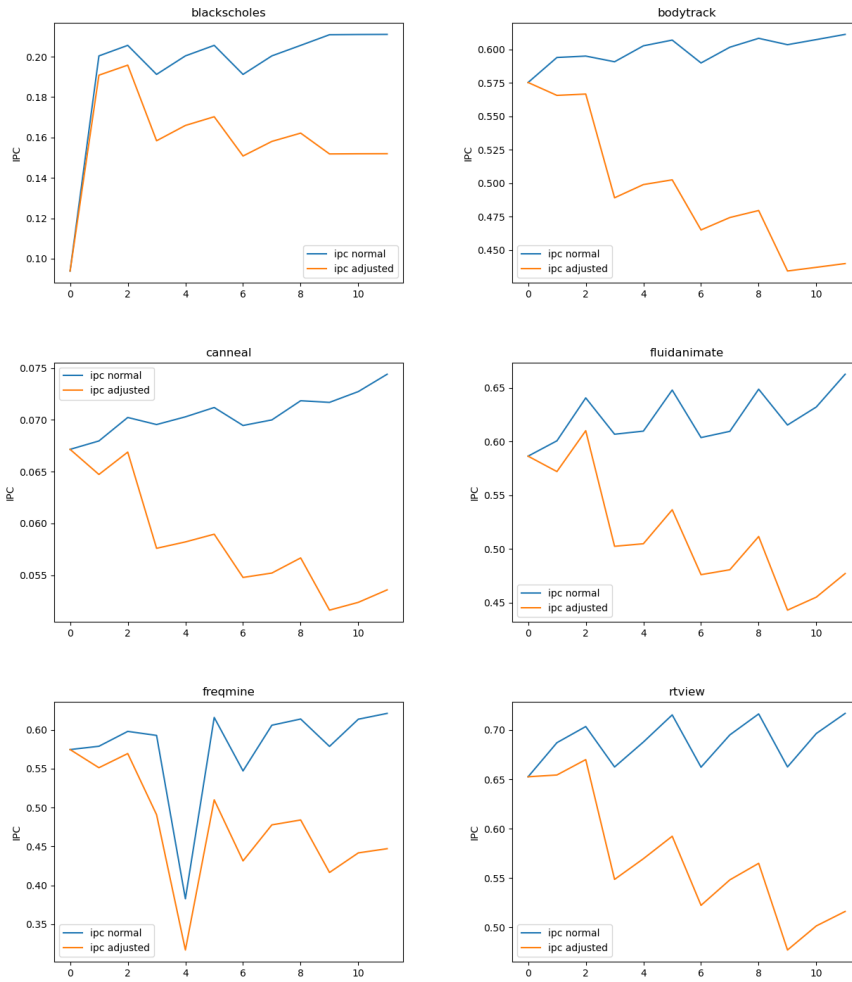
Στα πειράματα σχετικά με τα δύο επίπεδα της κρυφής μνήμης γίνονταν διαφοροποιήσεις κυρίως στο μέγεθος του μπλοκ και στην χωρητικότητα. Η συσχετιστικότητα είτε διπλασιαζόταν είτε έμενε σταθερή. Επομένως, δεδομένου μάλιστα ότι ο διπλασιασμός της χωρητικότητας προκαλεί μεγαλύτερη αύξηση του κύκλου, αναμένεται πτώση των καταγεγραμμένων IPC. Ωστόσο, αναμένεται να είναι φανερό η θετική επίδραση της αύξησης του μεγέθους μπλοκ.

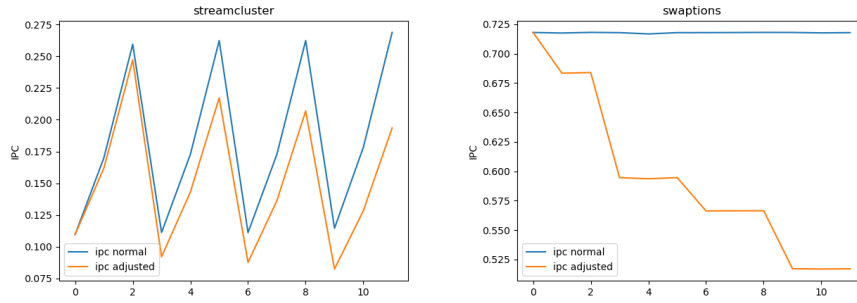


Πράγματι, κατά τη διάρκεια του πρώτου πειράματος παρατηρείται πτώση στην επίδοση του συστήματος, πράγμα λογικό αφού αυξάνεται συνεχώς η χωρητικότητα

του πρώτου επιπέδου μνήμης. Παρόλα αυτά, όταν η χωρητικότητα και η συσχέτιστικότητα παραμένουν σταθερές, η αύξηση του μεγέθους μπλοκ συνεχίζει να προκαλεί αύξηση στην επίδοση.

Στο δεύτερο πείραμα, είχε παρατηρηθεί μεγαλύτερη εξάρτηση από το μέγεθος μπλοκ από,τι στο πρώτο πείραμα. Επομένως, αναμένωνται ποιοτικά όμοιες γραφικές παραστάσεις, με κλιμακωμένες τις τιμές IPC.

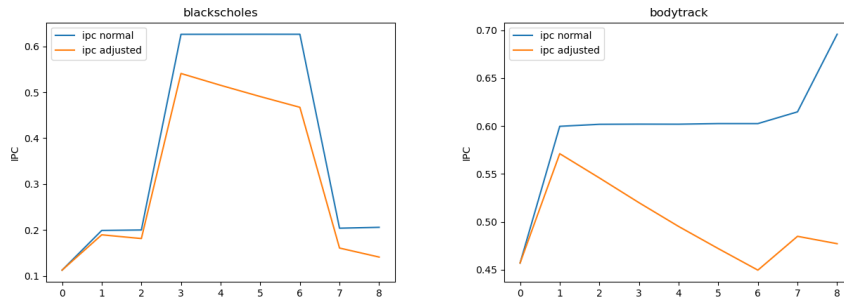


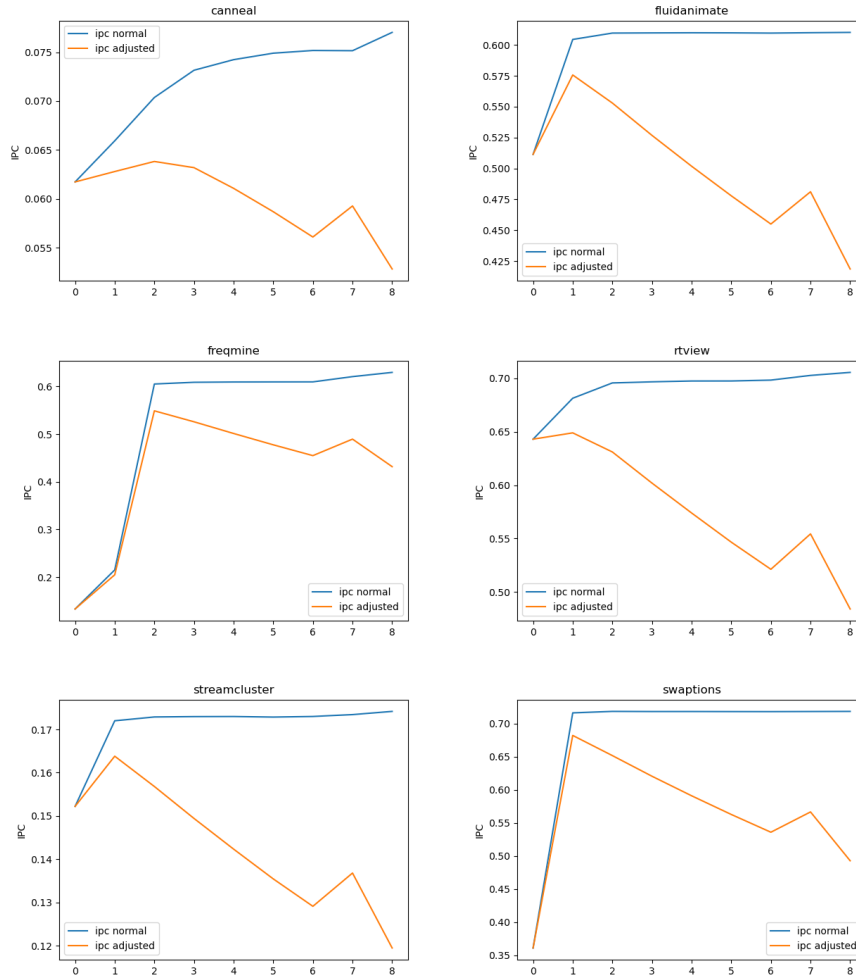


Πράγματι, στα εκτελέσιμα στα οποία το IPC δεν έδειχνε μεγάλες μεταβολές, η κλιμάκωση είναι έντονη, όπως για παράδειγμα στο εκτελέσιμο swaptions ή στο can-peal. Αντιθέτως, σε όσα έδειχναν έντονες διακυμάνσεις συναρτήσει του μεγέθους μπλοκ, όπως το blackscholes και το streamcluster, η κλιμάκωση δεν επιφέρει τόσο μεγάλη αλλαγή.

5.2 TLB

Στο πείραμα σχετικό με το TLB, είχε φανεί ότι η συσχετιστικότητα επιδρούσε άμεσα και σε μεγάλο βαθμό στην επίδοση του συστήματος. Μάλιστα, στα περισσότερα εκτελέσιμα, η συσχετιστικότητα είχε ήδη μεγιστοποιήσει την επίδρασή της στον πρώτο ή στον δεύτερο διπλασιασμό. Μικρότερη αλλά υπαρκτή επιρροή είχε επίσης το πλήθος των καταχωρήσεων στο TLB. Επομένως, αναμένεται η κατά πολύ ελάττωση της αυξητικής επίδρασης της συσχετιστικότητας στην επίδοση του συστήματος.





Πράγματι, παρατηρείται ότι τα σημεία των γραφικών παραστάσεων στα οποία είχε πλέον διπλασιαστεί πολλαπλές φορές η συσχετιστικότητα παρουσιάζουν χαμηλές τιμές IPC. Πολλά εκτελέσιμα απέδιδαν καλύτερα για χαμηλές τιμές συσχετιστικότητας, όπως 2 ή 4, και δεν βελτιώνονταν για υψηλές τιμές της. Αυτό αποτυπώνεται και στα ανανεωμένα γραφήματα, όπου παρουσιάζονται, κλιμακωμένα πλέον, τοπικά μέγιστα στην αρχή, μετά φθίνουν λόγω των αλληπάληλων διπλασιασμών και στο τέλος προκαλείται πάλι μέγιστο λόγω της επαναφοράς της συσχετιστικότητας σε μικρή τιμή και τον διπλασιασμό των καταχωρήσεων, ο οποίος επιφέρει μικρότερη κλιμάκωση.

5.3 Προανάκληση

Στο πείραμα της προανάκλησης δεν πραγματοποιήθηκαν μεταβολές στην χωρητικότητα της μνήμης ή στην συσχετιστικότητα, επομένως τα αποτελέσματα δεν χρειάζονται προσαρμογή.

6 Παράρτημα

6.1 Κώδικας εκτέλεσης πειραμάτων

Για την εκτέλεση των πειραμάτων, συνεγράφησαν κατάλληλα προγράμματα φλοιού bash, τα οποία σε μεγάλο βαθμό είναι όμοια μεταξύ τους και διαφέρουν σε λεπτομέρειες. Παρατίθενται και τα τέσσερα, ένα για το κάθε πείραμα, παρακάτω.

```

#!/bin/bash

# bash script to execute the first experiment

PIN_ROOT="/adv-comp-arch/lab1/pin-3.27-98718-gbeaa5d51e-gcc
-linux/pin"
SIM_ROOT="/adv-comp-arch/lab1/advcomparch-ex1-helptestcode/
pintool/obj-intel64/simulator.so"
EXE_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
executables"
INP_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
inputs"
OUT_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
outputs/exp1"

L1_FLAGS=(
"-L1c 32 -L1a 4 -L1b 64"
"-L1c 32 -L1a 8 -L1b 32"
"-L1c 32 -L1a 8 -L1b 64"
"-L1c 32 -L1a 8 -L1b 128"
"-L1c 64 -L1a 4 -L1b 64"
"-L1c 64 -L1a 8 -L1b 32"
"-L1c 64 -L1a 8 -L1b 64"
"-L1c 64 -L1a 8 -L1b 128"
"-L1c 128 -L1a 8 -L1b 32"
"-L1c 128 -L1a 8 -L1b 64"
"-L1c 128 -L1a 8 -L1b 128"
)
L2_FLAGS="-L2c 1024 -L2a 8 -L2b 128"
TLB_FLAGS="-TLBe 64 -TLBa 4 -TLBp 4096"

COMMANDS=(
"${EXE_ROOT}/blackscholes 1 ${INP_ROOT}/in_64K.txt prices.
txt"
"${EXE_ROOT}/bodytrack ${INP_ROOT}/sequenceB_4 4 4 4000 5 0
1"
"${EXE_ROOT}/canneal 1 15000 2000 ${INP_ROOT}/400000.nets
128"
"${EXE_ROOT}/fluidanimate 1 5 ${INP_ROOT}/in_300K.fluid out.
fluid"
"${EXE_ROOT}/freqmine ${INP_ROOT}/kosarak_990k.dat 790"
"${EXE_ROOT}/rtview ${INP_ROOT}/happy_buddha.obj -automove -
nthreads 1 -frames 3 -res 1920 1080"
"${EXE_ROOT}/streamcluster 10 20 128 16384 16384 1000 none
output.txt 1"
"${EXE_ROOT}/swaptions -ns 64 -sm 40000 -nt 1"
)

OUTPUT_FILE=(
"${OUT_ROOT}/blacksholes/exp1-blackscholes"

```

```

"${OUT_ROOT}/bodytrack/exp1-bodytrack"
"${OUT_ROOT}/canneal/exp1-canneal"
"${OUT_ROOT}/fluidanimate/exp1-fluidanimate"
"${OUT_ROOT}/freqmine/exp1-freqmine"
"${OUT_ROOT}/rtview/exp1-rtview"
"${OUT_ROOT}/streamcluster/exp1-streamcluster"
"${OUT_ROOT}/swaptions/exp1-swaptions"
)

export LD_LIBRARY_PATH=~/.adv-comp-arch/lab1/parsec-3.0/pkgs/
libs/hooks/inst/amd64-linux.gcc-serial/lib

for i in "${!L1_FLAGS[@]}";
do
    for j in "${!COMMANDS[@]}";
    do
        ${PIN_ROOT} -t ${SIM_ROOT} -o "${OUTPUT_FILE[$j]}${i}
            }.txt" ${L1_FLAGS[$i]} ${L2_FLAGS} ${TLB_FLAGS}
            -- ${COMMANDS[$j]}
    done
done

```

```

#!/bin/bash
# bash script to execute the second experiment

PIN_ROOT="/adv-comp-arch/lab1/pin-3.27-98718-gbeaa5d51e-gcc
-linux/pin"
SIM_ROOT="/adv-comp-arch/lab1/advcomparch-ex1-helpcode/
pintool/obj-intel64/simulator.so"
EXE_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
executables"
INP_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
inputs"
OUT_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
outputs/exp2"

L1_FLAGS="-L1c 32 -L1a 8 -L1b 64"
L2_FLAGS=(
"-L2c 512 -L2a 4 -L2b 64"
"-L2c 512 -L2a 8 -L2b 128"
"-L2c 512 -L2a 8 -L2b 256"
"-L2c 1024 -L2a 8 -L2b 64"
"-L2c 1024 -L2a 8 -L2b 128"
"-L2c 1024 -L2a 8 -L2b 256"
"-L2c 1024 -L2a 16 -L2b 64"
"-L2c 1024 -L2a 16 -L2b 128"
"-L2c 1024 -L2a 16 -L2b 256"
"-L2c 2048 -L2a 8 -L2b 64"
"-L2c 2048 -L2a 8 -L2b 128"
"-L2c 2048 -L2a 8 -L2b 256"
)
TLB_FLAGS="-TLBe 64 -TLBa 4 -TLBp 4096"

COMMANDS=(
"${EXE_ROOT}/blackscholes 1 ${INP_ROOT}/in_64K.txt prices.
txt"
"${EXE_ROOT}/bodytrack ${INP_ROOT}/sequenceB_4 4 4 4000 5 0
1"
"${EXE_ROOT}/canneal 1 15000 2000 ${INP_ROOT}/400000.nets
128"
"${EXE_ROOT}/fluidanimate 1 5 ${INP_ROOT}/in_300K.fluid out.
fluid"
"${EXE_ROOT}/freqmine ${INP_ROOT}/kosarak_990k.dat 790"
"${EXE_ROOT}/rtview ${INP_ROOT}/happy_buddha.obj -automove -
nthreads 1 -frames 3 -res 1920 1080"
"${EXE_ROOT}/streamcluster 10 20 128 16384 16384 1000 none
output.txt 1"
"${EXE_ROOT}/swaptions -ns 64 -sm 40000 -nt 1"
)

OUTPUT_FILE=(
"${OUT_ROOT}/blacksholes/exp2-blackscholes"

```

```

"${OUT_ROOT}/bodytrack/exp2-bodytrack"
"${OUT_ROOT}/canneal/exp2-canneal"
"${OUT_ROOT}/fluidanimate/exp2-fluidanimate"
"${OUT_ROOT}/freqmine/exp2-freqmine"
"${OUT_ROOT}/rtview/exp2-rtview"
"${OUT_ROOT}/streamcluster/exp2-streamcluster"
"${OUT_ROOT}/swaptions/exp2-swaptions"
)

export LD_LIBRARY_PATH=~/.adv-comp-arch/lab1/parsec-3.0/pkgs/
libs/hooks/inst/amd64-linux.gcc-serial/lib

for i in "${!L2_FLAGS[@]}";
do
    for j in "${!COMMANDS[@]}";
    do
        ${PIN_ROOT} -t ${SIM_ROOT} -o "${OUTPUT_FILE[$j]}${i}
            }.txt" ${L1_FLAGS} ${L2_FLAGS[$i]} ${TLB_FLAGS}
            -- ${COMMANDS[$j]}
    done
done

```

```

#!/bin/bash
# bash script to execute the third experiment
PIN_ROOT="/adv-comp-arch/lab1/pin-3.27-98718-gbeaa5d51e-gcc
-linux/pin"
SIM_ROOT="/adv-comp-arch/lab1/advcomparch-ex1-helppcode/
pintool/obj-intel64/simulator.so"
EXE_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
executables"
INP_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
inputs"
OUT_ROOT="/adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
outputs/exp3"

L1_FLAGS="-L1c 32 -L1a 8 -L1b 64"
L2_FLAGS="-L2c 1024 -L2a 8 -L2b 128"
TLB_FLAGS=(
"-TLBe 64 -TLBa 1 -TLBp 4096"
"-TLBe 64 -TLBa 2 -TLBp 4096"
"-TLBe 64 -TLBa 4 -TLBp 4096"
"-TLBe 64 -TLBa 8 -TLBp 4096"
"-TLBe 64 -TLBa 16 -TLBp 4096"
"-TLBe 64 -TLBa 32 -TLBp 4096"
"-TLBe 64 -TLBa 64 -TLBp 4096"
"-TLBe 128 -TLBa 4 -TLBp 4096"
"-TLBe 256 -TLBa 4 -TLBp 4096"
)

COMMANDS=(
"${EXE_ROOT}/blackscholes 1 ${INP_ROOT}/in_64K.txt prices.
txt"
"${EXE_ROOT}/bodytrack ${INP_ROOT}/sequenceB_4 4 4 4000 5 0
1"
"${EXE_ROOT}/canneal 1 15000 2000 ${INP_ROOT}/400000.nets
128"
"${EXE_ROOT}/fluidanimate 1 5 ${INP_ROOT}/in_300K.fluid out.
fluid"
"${EXE_ROOT}/freqmine ${INP_ROOT}/kosarak_990k.dat 790"
"${EXE_ROOT}/rtview ${INP_ROOT}/happy_buddha.obj -automove -
nthreads 1 -frames 3 -res 1920 1080"
"${EXE_ROOT}/streamcluster 10 20 128 16384 16384 1000 none
output.txt 1"
"${EXE_ROOT}/swaptions -ns 64 -sm 40000 -nt 1"
)

OUTPUT_FILE=(
"${OUT_ROOT}/blackscholes/exp3-blackscholes"
"${OUT_ROOT}/bodytrack/exp3-bodytrack"
"${OUT_ROOT}/canneal/exp3-canneal"
"${OUT_ROOT}/fluidanimate/exp3-fluidanimate"
"${OUT_ROOT}/freqmine/exp3-freqmine"
"${OUT_ROOT}/rtview/exp3-rtview"

```



```

    "${OUT_ROOT}/streamcluster/exp3-streamcluster"
    "${OUT_ROOT}/swaptions/exp3-swaptions"
)

export LD_LIBRARY_PATH=~/.adv-comp-arch/lab1/parsec-3.0/pkgs/
    libs/hooks/inst/amd64-linux.gcc-serial/lib

for i in "${!TLB_FLAGS[@]}";
do
    for j in "${!COMMANDS[@]}";
    do
        ${PIN_ROOT} -t ${SIM_ROOT} -o "${OUTPUT_FILE[$j]}${i}
            }.txt" ${L1_FLAGS} ${L2_FLAGS} ${TLB_FLAGS[$i]}
            -- ${COMMANDS[$j]}
    done
done

```

```

#!/bin/bash
# bash script to execute the fourth experiment

PIN_ROOT=~/.adv-comp-arch/lab1/pin-3.27-98718-gbeaa5d51e-gcc
-linux/pin"
SIM_ROOT=~/.adv-comp-arch/lab1/advcomparch-ex1-helpcode/
pintool/obj-intel64/simulator.so"
EXE_ROOT=~/.adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
executables"
INP_ROOT=~/.adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
inputs"
OUT_ROOT=~/.adv-comp-arch/lab1/parsec-3.0/parsec_workspace/
outputs/exp4"

L1_FLAGS="-L1c 32 -L1a 8 -L1b 64"
L2_FLAGS="-L2c 1024 -L2a 8 -L2b 128 -L2prf "
TLB_FLAGS="-TLBe 64 -TLBa 4 -TLBp 4096"

COMMANDS=(
    "${EXE_ROOT}/blackscholes 1 ${INP_ROOT}/in_64K.txt
    prices.txt"
    "${EXE_ROOT}/bodytrack ${INP_ROOT}/sequenceB_4 4 4 4000
    5 0 1"
    "${EXE_ROOT}/canneal 1 15000 2000 ${INP_ROOT}/400000.
    nets 128"
    "${EXE_ROOT}/fluidanimate 1 5 ${INP_ROOT}/in_300K.fluid
    out.fluid"
    "${EXE_ROOT}/freqmine ${INP_ROOT}/kosarak_990k.dat 790"
    "${EXE_ROOT}/rtview ${INP_ROOT}/happy_buddha.obj -
    automove -nthreads 1 -frames 3 -res 1920 1080"
    "${EXE_ROOT}/streamcluster 10 20 128 16384 16384 1000
    none output.txt 1"
    "${EXE_ROOT}/swaptions -ns 64 -sm 40000 -nt 1"
)
OUTPUT_FILE=(
    "${OUT_ROOT}/blackscholes/exp4-blackscholes"
    "${OUT_ROOT}/bodytrack/exp4-bodytrack"
    "${OUT_ROOT}/canneal/exp4-canneal"
    "${OUT_ROOT}/fluidanimate/exp4-fluidanimate"
    "${OUT_ROOT}/freqmine/exp4-freqmine"
    "${OUT_ROOT}/rtview/exp4-rtview"
    "${OUT_ROOT}/streamcluster/exp4-streamcluster"
    "${OUT_ROOT}/swaptions/exp4-swaptions"
)

PREFETCH=(1 2 4 8 16 32 64)

export LD_LIBRARY_PATH=~/.adv-comp-arch/lab1/parsec-3.0/pkgs/
libs/hooks/inst/amd64-linux.gcc-serial/lib

```

```

for i in "${!PREFETCH[@]}";
do
    echo ${L2_FLAGS}${PREFETCH[$i]}
    for j in "${!COMMANDS[@]}";
    do
        $PIN_ROOT -t $SIM_ROOT -o "${OUTPUT_FILE[$j]}${i}.
            txt" $L1_FLAGS ${L2_FLAGS}${PREFETCH[$i]}
            $TLB_FLAGS -- $COMMANDS[$j]
    done
done

```

6.2 Κώδικας εξαγωγής IPC

Ο παρακάτω κώδικας χρησιμοποιήθηκε για την εξαγωγή του IPC από τις εξόδους των εκτελέσιμων και την εγγραφή του σε ένα αρχείο `ipc.txt`. Πρέπει να εκτελεστεί σε έναν κατάλογο όπου βρίσκονται υποκατάλογοι με τα αποτελέσματα των προσομοιώσεων ανά εκτελέσιμο.

```

#!/bin/bash

# specify the output file name
output_file="ipc.txt"

for subdir in */; do
    cd $subdir
    sorted_files=$(ls *.txt | sort -V)
    for file in $sorted_files; do
        # extracts the sixth line of the file
        ipc_line=$(sed '6q;d' $file)
        # extracts the floating point number from the sixth
        line
        number=$(echo $ipc_line | grep -oE '0\.[0-9]+')
        # appends the number to the output file
        echo $number >> $output_file
    done
    cd ..
done

```

6.3 Κώδικας παραγωγής γραφημάτων

Ο παρακάτω κώδικας χρησιμοποιήθηκε για την παραγωγή των γραφημάτων επιδόσεως καθεμίας προσομοίωσης ανά εκτελέσιμο ανά πείραμα. Επομένως, εκτελείται από έναν κατάλογο ο οποίος εμπεριέχει υποκαταλόγους, ένας για κάθε πείραμα, στους οποίους εμπεριέχονται τα αποτελέσματα των προσομοιώσεων ανά εκτελέσιμο.

```

import os
import matplotlib.pyplot as plt

# define the parent directory as the current directory
parent_dir = os.getcwd()

# loop through all experiment directories
for exp_dir in os.listdir(parent_dir):
    if not os.path.isdir(os.path.join(parent_dir, exp_dir)):
        continue # skip non-directory files

    # loop through all subdirectories within the experiment
    # directory
    for sub_dir in os.listdir(os.path.join(parent_dir,
exp_dir)):
        if not os.path.isdir(os.path.join(parent_dir,
exp_dir, sub_dir)):
            continue # skip non-directory files

        # find the ipc.txt file in the subdirectory
        data_file = os.path.join(parent_dir, exp_dir,
sub_dir, 'ipc.txt')
        if not os.path.isfile(data_file):
            continue # skip if ipc.txt file not found

        # read the numbers from the ipc.txt file
        with open(data_file, 'r') as f:
            numbers = []
            for line in f:
                line = line.strip()
                if line:
                    try:
                        number = float(line)
                    except ValueError:
                        continue # skip non-numeric lines
                    numbers.append(number)

        # create a plot of the numbers
        fig, ax = plt.subplots()
        ax.plot(numbers)
        ax.set_title(sub_dir)
        ax.set_ylabel("IPC")

        # save the plot in the same directory as the ipc.txt
        # file
        plot_file = os.path.join(parent_dir, exp_dir,
sub_dir, '{}_{}_ipc.png'.format(exp_dir, sub_dir)
)
        fig.savefig(plot_file)
        plt.close(fig)

```

Για την παραγωγή των προσαρμοσμένων γραφημάτων, εκτελείται ο παρακάτω κώδικας. Πρέπει να εκτελεστεί από τον ίδιο κατάλογο με το εκτελέσιμο `bash` για την εξαγωγή του `ipc`. Σημειώνεται, ότι ο συγκεκριμένος κώδικας αφορά στο πρώτο εκτελέσιμο. Ο πίνακας `adjustments` θα πρέπει να προσαρμοστεί στο κάθε πείραμα.

```

import os
import matplotlib.pyplot as plt

parent_dir = os.getcwd()
subdirs = sorted(os.listdir(parent_dir))

assos = (100.0/105.0)
storg = (100.0/115.0)
adjustments = [1, assos, assos, assos,
                storg, assos*storg, assos*storg, assos*storg,
                assos*(storg**2), assos*(storg**2), assos*(storg**2)
                ]

for sub_dir in subdirs:
    # find the ipc.txt file in the subdirectory
    data_file = os.path.join(parent_dir, sub_dir, 'ipc.txt')
    if not os.path.isfile(data_file):
        continue # skip if ipc.txt file not found

    # read the numbers from the ipc.txt file
    with open(data_file, 'r') as f:
        numbers = []
        for line in f:
            line = line.strip()
            if line:
                try:
                    number = float(line)
                except ValueError:
                    continue # skip non-numeric lines
                numbers.append(number)

    adjusted = [ a * b for a, b in zip(numbers, adjustments)
                ]

    # create a plot of the numbers
    fig, ax = plt.subplots()
    ax.plot(numbers, label="ipc normal")
    ax.plot(adjusted, label="ipc adjusted")
    ax.set_title(sub_dir)
    ax.set_ylabel("IPC")
    plt.legend()

    # save the plot in the same directory as the ipc.txt
    # file
    plot_file = os.path.join(parent_dir, sub_dir, '
                             ipc_adjusted.png')
    print(plot_file)
    fig.savefig(plot_file)
    plt.close(fig)

```