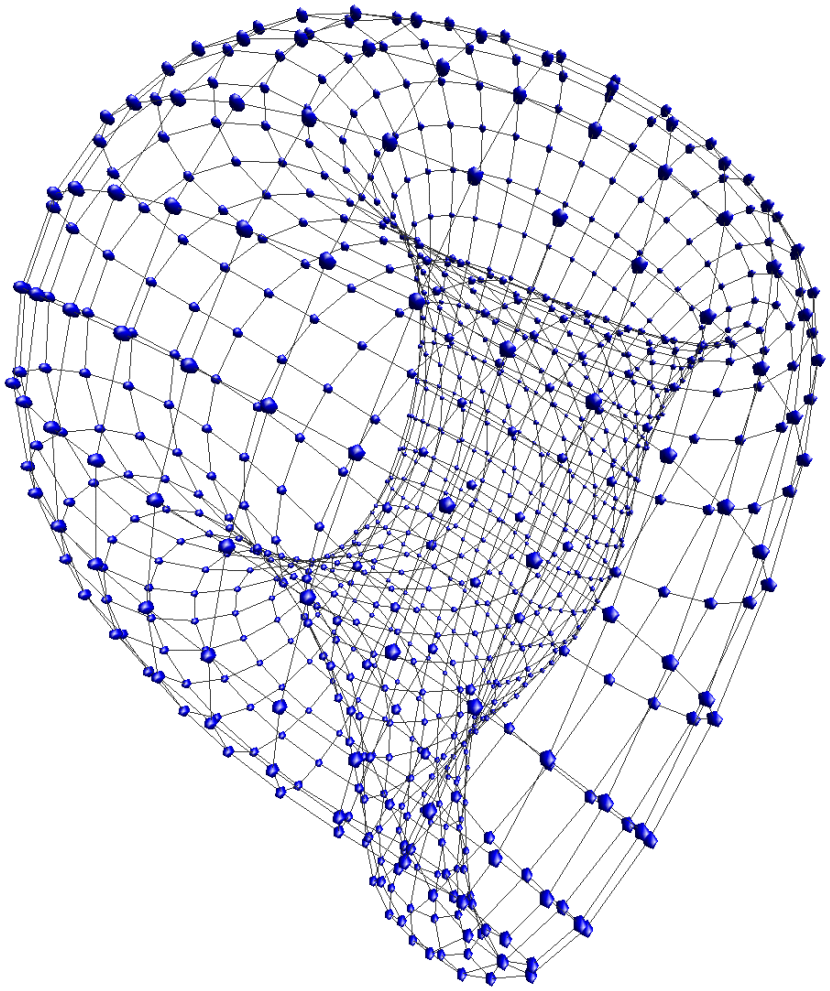
Παράλληλη Επίλυση Γραμμικού Συστήματος

Υλοποίηση της παράλληλης μεθόδου επίλυσης γραμμικών συστημάτων Gauss-Jordan κατά στήλες, μορφή kji.



Παπαπαναγιωτάκης-Μπουσύ Ιάσων

Μάθημα: Παράλληλοι Αλγόριθμοι.

Contents

Πληροφορίες για το πρόγραμμα

- Το πρόγραμμα αναπτύχθηκε σε Gnu Linux σε C++ με την χρήση του Message Passing Interface (MPI).
- Υλοποιήθηκαν οι δύο ζητούμενοι τρόποι μοιράσματος, κατά ομάδες και round-robin.
- Το master process δημιουργεί ένα τυχαίο γραμμικό σύστημα και μοιράζει στα υπόλοιπα τις στήλες που τους αντιστοιχούν ανάλογα με τον ζητούμενο τρόπο μοιράσματος.
- Το master process χειρίζεται πάντα το διάνυσμα B.
- Τα ορίσματα του προγράμματος είναι η διάσταση του γραμμικού συστήματος και ο τρόπος διαμοιρασμού των στηλών.

  mpiexec -n <number of processes> ./parallel-gauss-jordan <dimention> <distribution mode (0/1)>

  distribution = 1, ανά ομάδες και distribution = 0, round-robin.

- Η διάσταση πρέπει να είναι πολλαπλάσιο του αριθμού διεργασιών.
- Για compile χρησιμοποιείται το makefile που συνοδεύει τον πηγαίο κώδικα.

Μετρήσεις

Οι μετρήσεις έγιναν στο εργαστήριο Linux της σχολής καθώς ο υπολογιστής Ουρανός HP-superdome είχε τεχνικά προβλήματα και δεν βρέθηκε τρόπος να εκτελέσει κώδικα C++. Να σημειωθεί ότι το περιβάλλον στο οποίο έγιναν οι μετρήσεις είναι ανομοιογενές, με πολλούς άλλους φοιτητές να το χρησιμοποιούν ταυτόχρονα. Έγινε μεγάλη προσπάθεια για την όσο καλύτερη παραμετροποίηση της εκτέλεσης.

Επιλέχθηκαν μεγέθη πινάκων που να αναδεικνύουν την υπεροχή της παράλληλης έκδοσης τους προγράμματος. Παρατηρήθηκε ότι για μικρά μεγέθη οι εκτελέσεις με λιγότερες διεργασίες τερμάτιζαν πιο γρήγορα από αυτές με πολλές.

Τα επιλεγμένα μεγέθη είναι 1024, 2048, 3200, 4096, 6400.

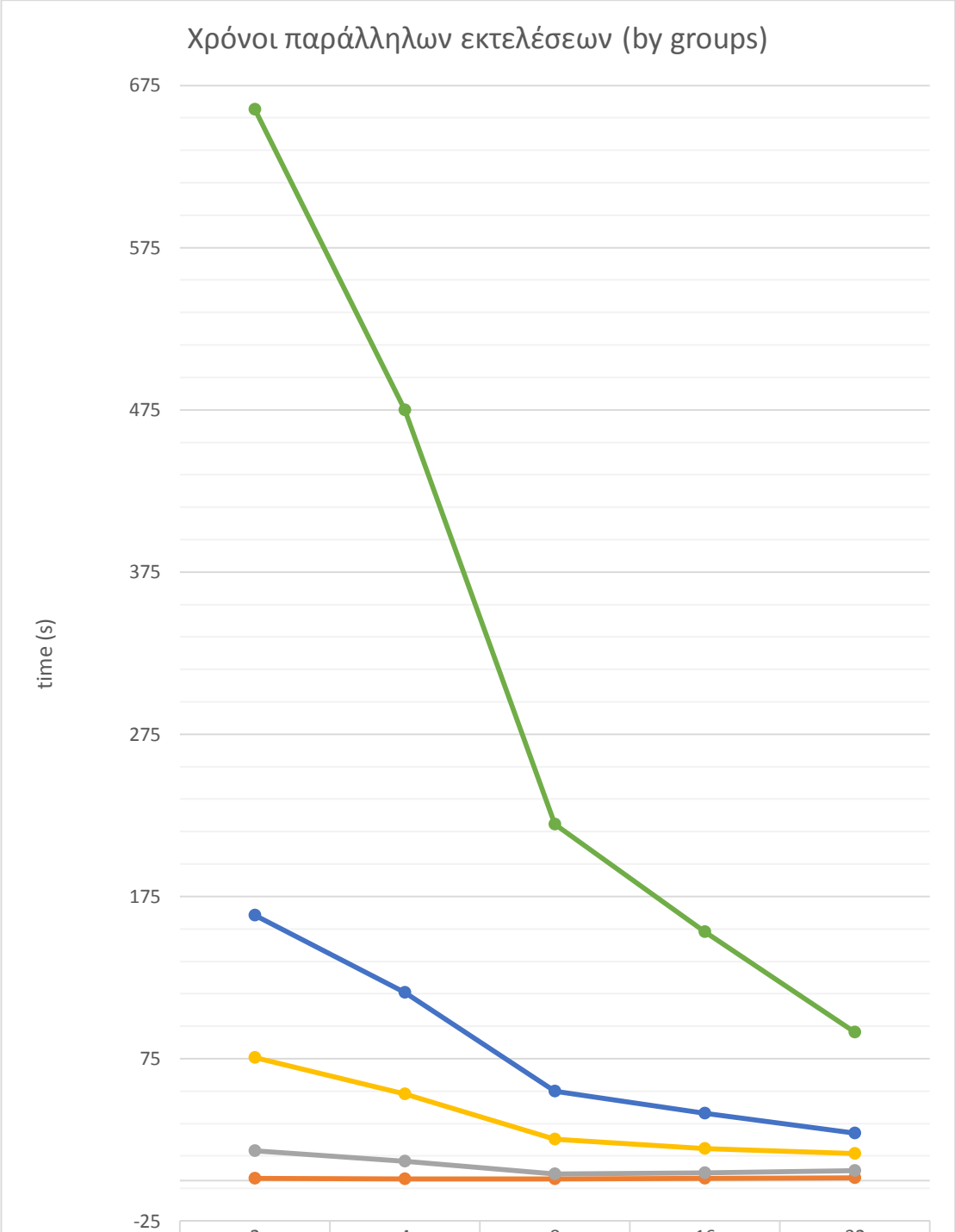
Σειριακή εκτέλεση

Για την μέτρηση του σειριακού χρόνου χρησιμοποιήθηκε το ίδιο πρόγραμμα με μόνο μία διεργασία. Παρακάτω παρουσιάζονται οι χρόνοι εκτέλεσης για τα επιλεγμένα μεγέθη.

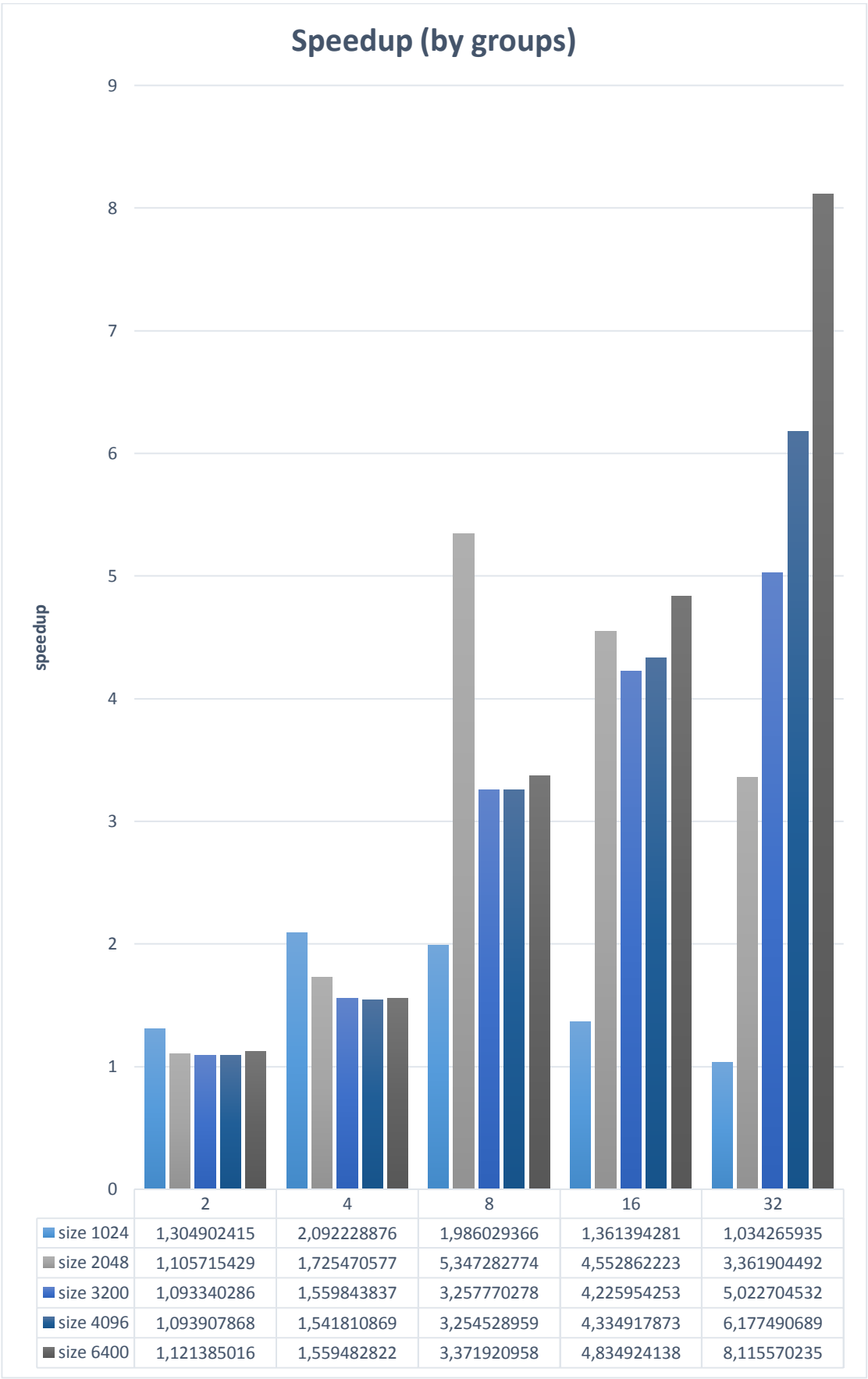
Παράλληλες εκτελέσεις

Πρώτα παρουσιάζονται τα αποτελέσματα για το μοίρασμα ανά ομάδες και έπειτα τα αποτελέσματα με το κυκλικό μοίρασμα. Κάθε ενότητα περιέχει πρώτα τους χρόνους και μετά τα speedup και efficiency.

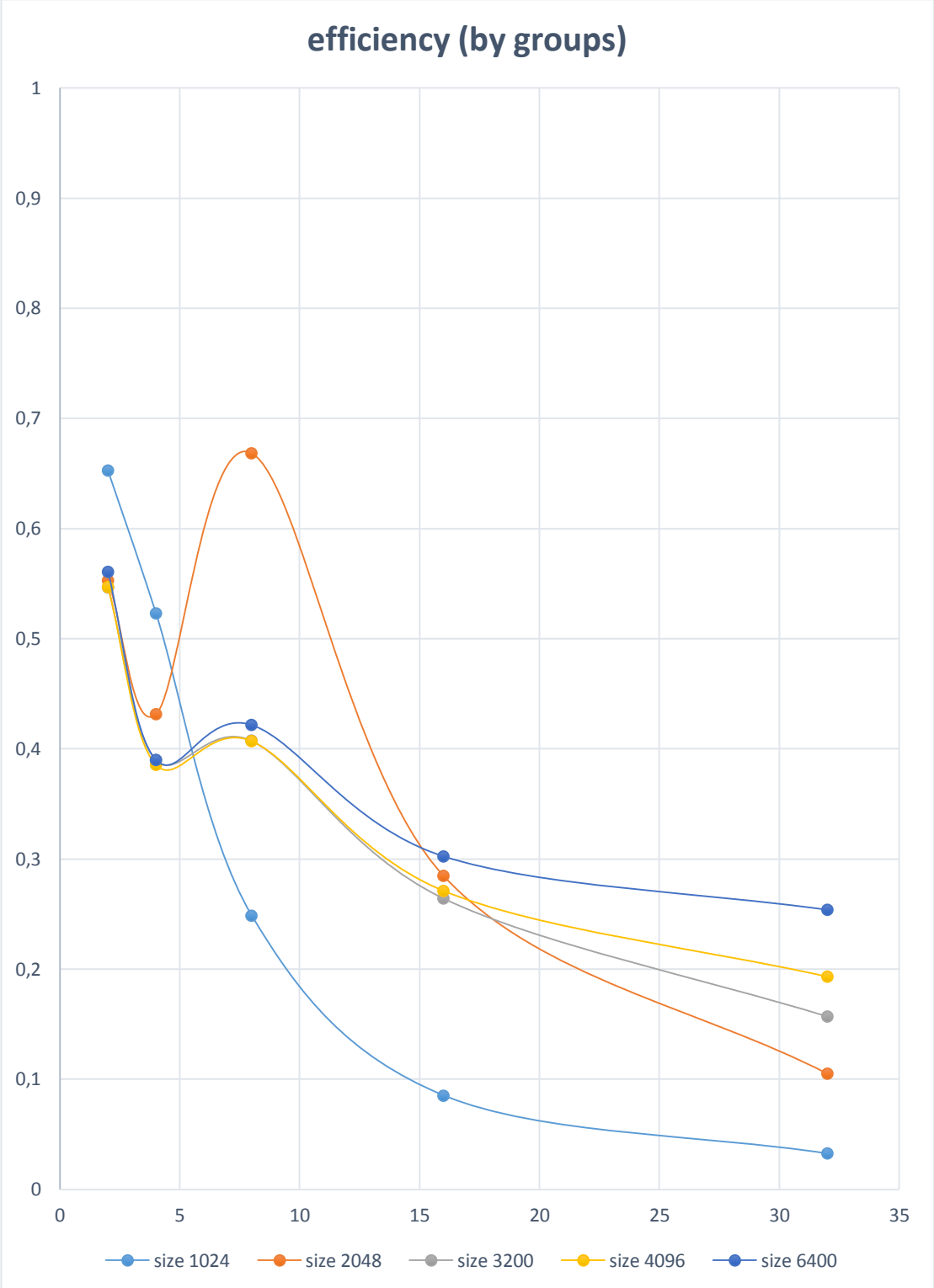
Μοίρασμα ανά ομάδες



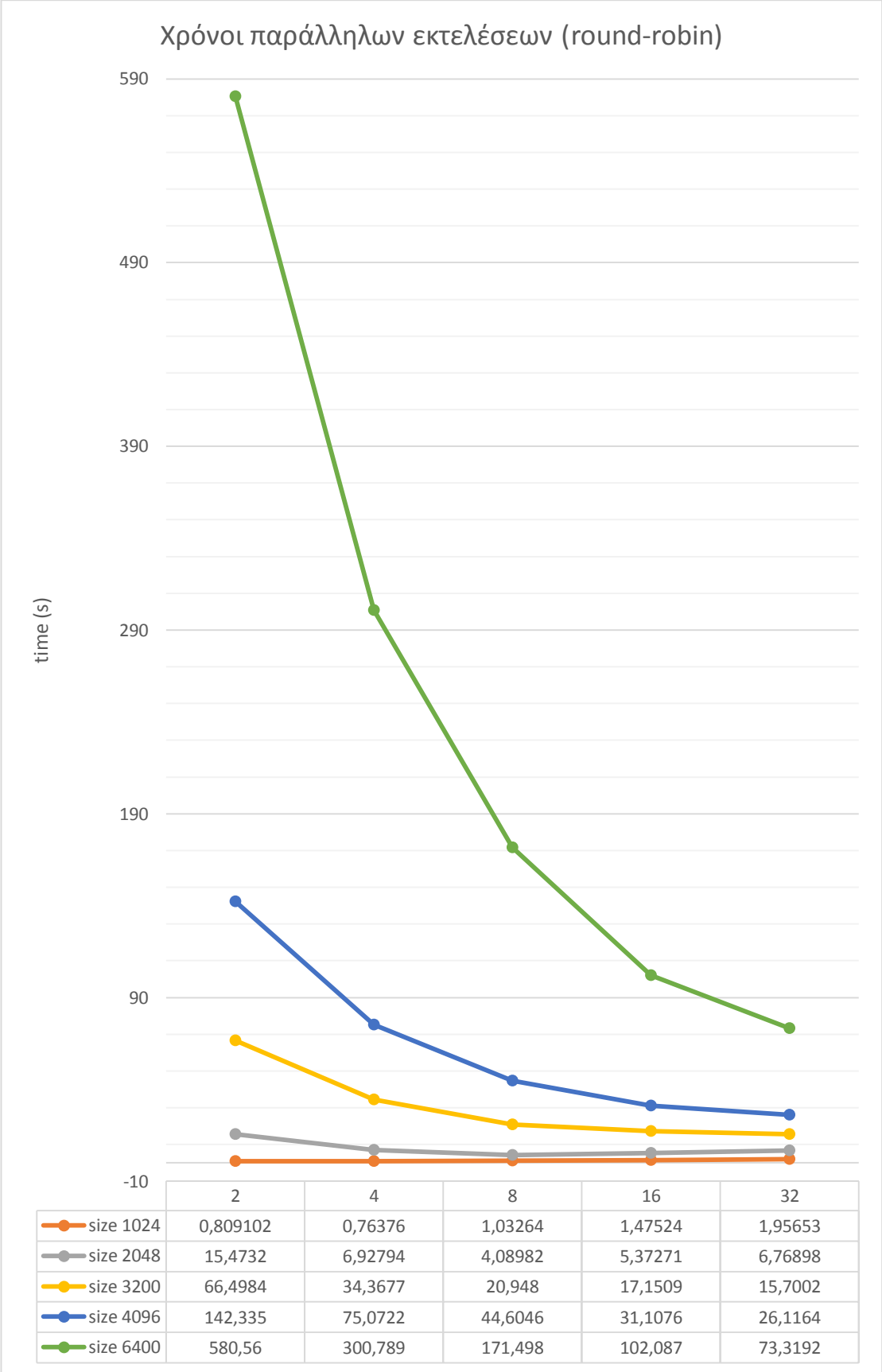
	2	4	8	16	32
size 1024	1,18615	0,73979	0,779349	1,13693	1,49653
size 2048	18,2471	11,6931	3,77315	4,43152	6,00139
size 3200	75,754	53,0982	25,4238	19,5991	16,4901
size 4096	163,309	115,867	54,8912	41,2107	28,9187
size 6400	660,411	474,885	219,63	153,172	91,2536



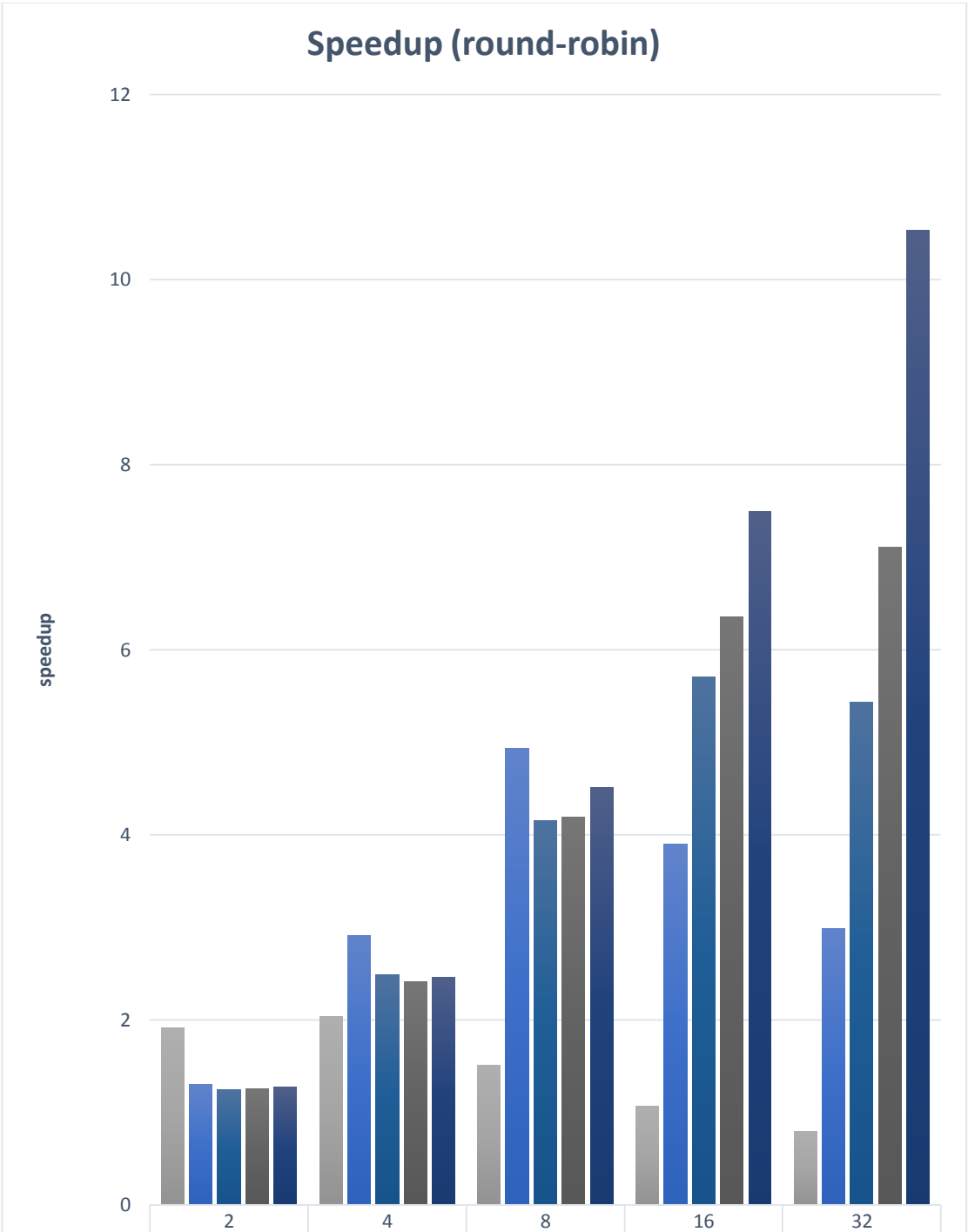
	2	4	8	16	32
size 1024	1,304902415	2,092228876	1,986029366	1,361394281	1,034265935
size 2048	1,105715429	1,725470577	5,347282774	4,552862223	3,361904492
size 3200	1,093340286	1,559843837	3,257770278	4,225954253	5,022704532
size 4096	1,093907868	1,541810869	3,254528959	4,334917873	6,177490689
size 6400	1,121385016	1,559482822	3,371920958	4,834924138	8,115570235


efficiency (by groups)
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
0
5
10
15
20
25
30
35
size 1024
size 2048
size 3200
size 4096
size 6400

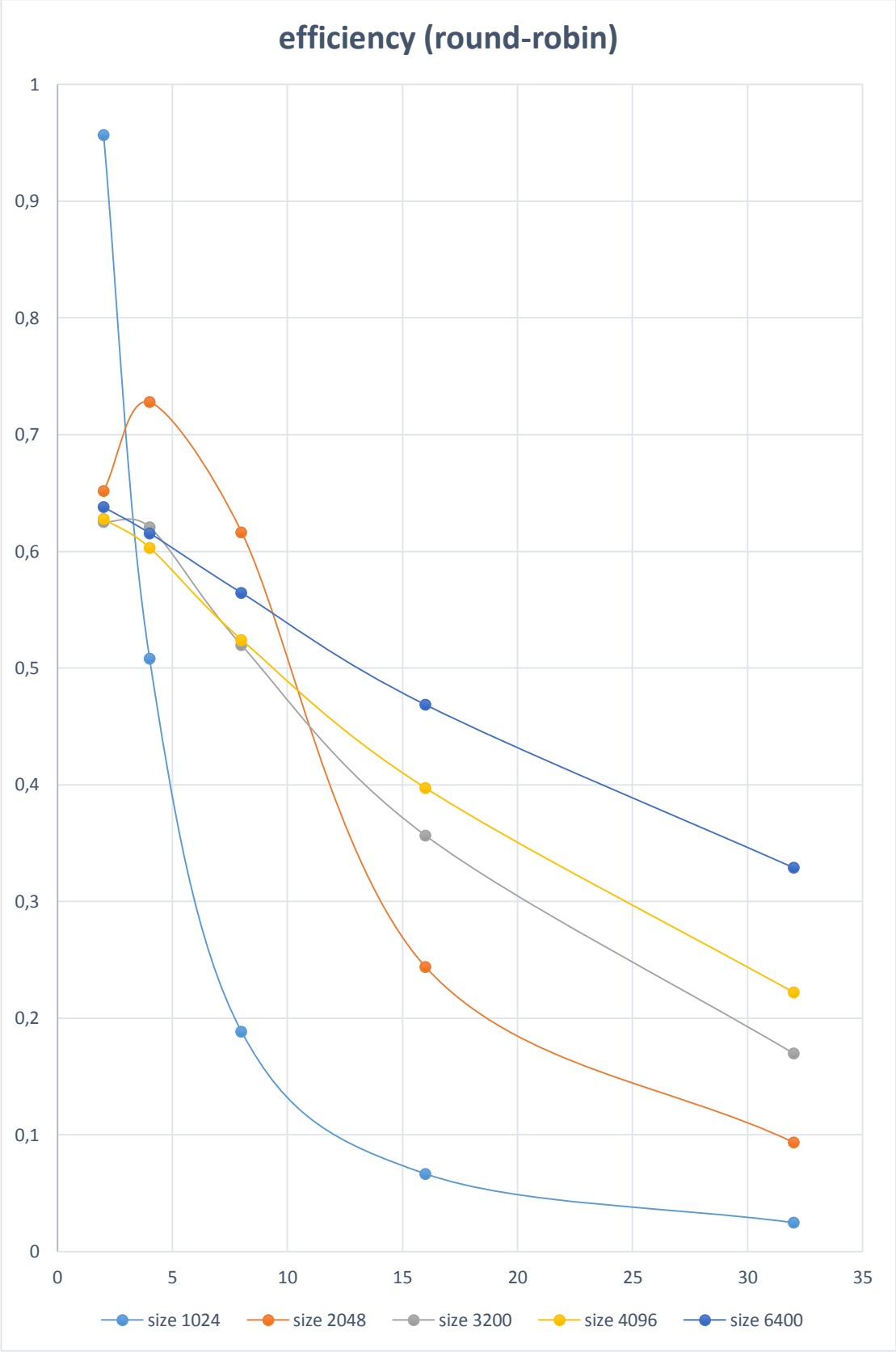
Κυκλικό μοίρασμα



	2	4	8	16	32
size 1024	0,809102	0,76376	1,03264	1,47524	1,95653
size 2048	15,4732	6,92794	4,08982	5,37271	6,76898
size 3200	66,4984	34,3677	20,948	17,1509	15,7002
size 4096	142,335	75,0722	44,6046	31,1076	26,1164
size 6400	580,56	300,789	171,498	102,087	73,3192



	2	4	8	16	32
size 1024	1,912997372	2,03235091	1,50628181	1,064342887	0,791099549
size 2048	1,303408474	2,911096228	4,931243918	3,898904056	2,986960862
size 3200	1,249936239	2,482187864	4,154998044	5,702466195	5,434633404
size 4096	1,255102399	2,411768518	4,193091826	6,355754316	7,112683346
size 6400	1,275621813	2,462107989	4,514710186	7,496935223	10,53161868


efficiency (round-robin)
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
0
5
10
15
20
25
30
35
size 1024
size 2048
size 3200
size 4096
size 6400

Σχόλια

Παρατηρούμε ότι καθώς για μικρές διαστάσεις το overhead της παραλληλίας επιβαρύνει σημαντικά τον συνολικό χρόνο με ακραίο παράδειγμα η εκτέλεση του προγράμματος για μέγεθος 1024 με 32 διεργασίες να είναι πιο αργή από την σειριακή εκτέλεση. Από την άλλη, όσο η διάσταση αυξάνετε εμφανίζεται όλο και περισσότερο το πλεονέκτημα της παράλληλης εκτέλεσης του προγράμματος. Αυτό συμβαίνει γιατί ο χρόνος επικοινωνίας αποτελεί όλο και μικρότερο ποσοστό του συνολικού χρόνου καθώς κάθε διεργασία έχει όλο και μεγαλύτερο όγκο δεδομένων να επεξεργαστεί.

Σχετικά με την διαφορά μεταξύ των δύο τρόπων μοιράσματος των στηλών, παρατηρούμε ότι το κυκλικό μοίρασμα υπερτερεί του μοιράσματος ανά ομάδες όπως ήταν αναμενόμενο. Στο κυκλικό μοίρασμα απασχολούνται όλοι οι επεξεργαστές για μεγαλύτερη χρονική διάρκεια με αποτέλεσμα χαμηλότερους χρόνους εκτέλεσης. Σε κάθε περίπτωση έχουμε μερική οδήγηση η οποία επιβραδύνει την εκτέλεση αλλά είναι απαραίτητη αν δεν θέλουμε να πέσει το πρόγραμμα σε αριθμητική αστάθεια.

Όσων αφορά την αποδοτικότητα (efficiency) του προγράμματος, παρατηρούμε ότι αυτή πέφτει όσο αυξάνονται ο αριθμός των διεργασιών. Όσο μικρότερη είναι η διάσταση προβλήματος τόσο πιο γρήγορα πέφτει η αποδοτικότητα. Αυτό οφείλετε στην κλιμακούμενη απαιτούμενη επικοινωνία που γίνετε πάνω από ένα δίκτυο που δεν σχεδιαστικέ για κάτι τέτοιο (Ethernet δίκτυο του Linux lab).

Τέλος, η θεωρητική αποδοτικότητα της μεθόδου κατά στήλες με μερική οδήγηση για O(n) επεξεργαστές είναι 0.286. Αν και στις παρούσες δοκιμές δεν είχαμε O(n) επεξεργαστές, παρατηρούμε ότι θα ήταν μάλλον απίθανο να επιτευχθεί αποδοτικότητα 1. Ένας από τους λόγους για τον οποίο δεν μπορούμε να το πετύχουμε θεωρητικά είναι ότι κατά την διάρκεια της οδήγησης ένας επεξεργαστής δουλεύει και οι υπόλοιποι περιμένουν ενώ για να είχαμε αποδοτικότητα 1 θα έπρεπε να δουλεύουν συνέχεια όλοι οι επεξεργαστές.