

Συστήματα Παράλληλης Επεξεργασίας

Εργαστηριακή αναφορά 1

Game Of Life

Γιάννος Παράνομος - 03118021

Νικήτας Τσίννας - 03118187

Οκτώβριος 2022

1 Εισαγωγή

Σε αυτή την εργαστηριακή άσκηση υλοποιούμε τον αυτοματισμό Conway's Game Of Life μέσω παράλληλου προγράμματος στο μοντέλου του κοινού χώρου διευθύνσεων με την χρήση του OpenMP module.

1.1 Υλοποίηση

Προσθέσαμε την ακόλουθη γραμμή κώδικα στο αρχείο που μας δόθηκε στο δεύτερο nested loop για να παραλληλοποιήσουμε τον υπολογισμό κελιών του πίνακα:

```
#pragma omp parallel for shared(previous, current, N) private(nbrs, i, j)
```

Χωρίζουμε τις μεταβλητές σε shared, δηλαδή σε αυτές που όλοι οι εργάτες θα έχουν πρόσβαση, και σε private, δηλαδή σε αυτές που οι εργάτες έχουν αποκλειστική πρόσβαση. Είναι σημαντικός ο διαχωρισμός ώστε να μην υπάρξουν ανεπιθύμητα race conditions.

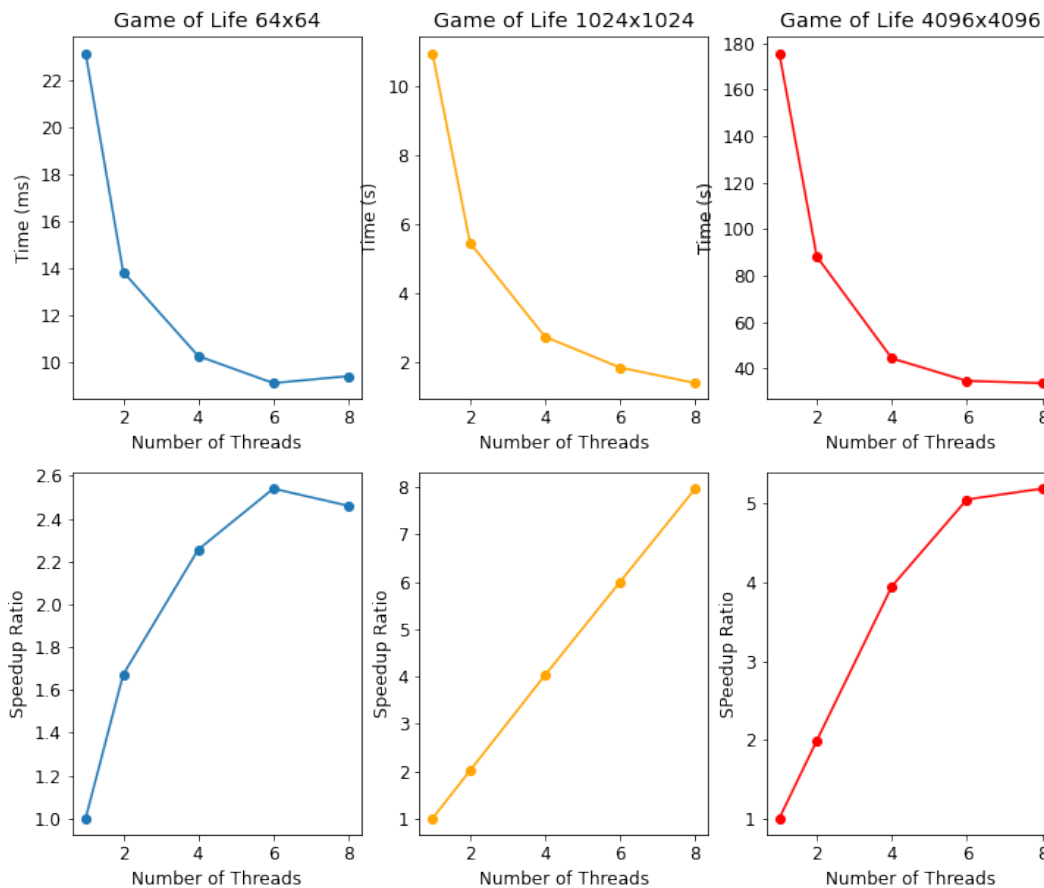
1.2 Εκτέλεση

Εκτελέσαμε το πρόγραμμα για 1000 χρονικά βήματα και για μεγέθη πίνακα 64x64, 1024x1024 και 4096x4096. Για κάθε διαφορετικό μέγεθος πίνακα εκτελέσαμε το πρόγραμμα επιλέγοντας διαφορετικό πλήθος threads, δηλαδή για 1, 2, 4, 6 και 8.

Άρα, οι συνολικές εκτελέσεις ήταν 15. Για κάθε μία από αυτές καταγράψαμε τους χρόνους εκτέλεσης και απεικονίσαμε τα αποτελέσματα σε διαγράμματα.

2 Αποτελέσματα

Threads Size	64x64	1024x1024	4096x4096
1	0.023136	10.969285	175.6115125
2	0.013838	5.457227	88.201773
4	0.010257	2.722889	44.463432
6	0.009107	1.828968	34.752681
8	0.009404	1.376893	33.832396



Το Speedup Ratio μας βοηθάει να αποφανθούμε για την αποτελεσματικότητα του παραλληλισμού και να εξάγουμε συμπεράσματα για το scalability. Ορίζεται ως:

$$S = \frac{T}{T_1}$$

Όπου T ο συνολικός χρόνος εκτέλεσης με όλους του εργάτες, ενώ T₁ συνολικός χρόνος εκτέλεσης για έναν εργάτη.

3 Σχόλια

Ενδιαφέρον παρουσιάζουν οι διαφορές στα διαγράμματα του δείκτη επιτάχυνσης (Speedup Ratio). Πιο συγκεκριμένα για κάθε μέγεθος πίνακα αντιστοιχα παρατηρούμε:

- 64x64: Έχουμε **typical speedup** με scalability break point στα 6 threads.
- 1024x1024: Έχουμε **perfect linear speedup**.
- 4096x4096: Έχουμε **linear speedup** για τα πρώτα 4 threads και έπειτα **typical speedup** με scalability break point στα 6 threads.

3.1 Μέγεθος πίνακα 64x64

Παρατηρούμε πως υπάρχει weak scalability. Αυτό υποθέτουμε πως συμβαίνει γιατί σπαταλιέται πολύς χρόνος στην πρόσβαση των threads στην κοινή μνήμη και υπάρχει συμφόρηση στο memory bus, χωρίς όμως να προσφέρεται σημαντική βελτίωση επίδοσης. Λόγω του μικρού μεγέθους του πίνακα, μπορούν να χωρέσουν και οι 2 πίνακες στην cache, οπότε η αρχική σειριακή εκτέλεση έχει αρκετά καλή επίδοση ήδη.

3.2 Μέγεθος πίνακα 1024x1024

Για αυτό το μέγεθος του πίνακα παρατηρούμε πως υπάρχει γραμμική επιτάχυνση μέχρι και τα 8 threads. Φαίνεται πως έχει παραλληλοποιηθεί κατάλληλα το πρόγραμμα. Υποθέτουμε πως μεγάλο ρόλο στην καλή κλιμάκωση έπαιξε η χρήση μεγαλύτερου μεγέθους cache εφόσον χρησιμοποιούνται περισσότεροι επεξεργαστές.

3.3 Μέγεθος πίνακα 4096x4096

Παρατηρούμε πως μέχρι και τα 4 threads έχουμε γραμμική επιτάχυνση για τον λόγο που περιγράψαμε προηγουμένως. Ωστόσο στα 6 threads παρατηρούμε πως σπάει η γραμμικότητα και στα 8 έχουμε scalability break. Μετά τα 6 threads φαίνεται πως η επικοινωνία για την πρόσβαση στην κοινή μνήμη απασχολεί πολύ περισσότερο το memory bus και έτσι δυσχεραίνει σε μεγαλύτερο βαθμό την επίδοση από ότι αυτή βελτιώνεται μέσω του παραλληλισμού.

4 GIF

Στην υποβολή έχουμε συμπεριλάβει ένα gif για μέγεθος πίνακα 100. Παρουσιάζονται μερικά ενδιαφέροντα μοτίβα. Ακολουθούν 2 στιγμιότυπα.

