



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Συστήματα Παράλληλης Επεξεργασίας 9ο Εξάμηνο

1η Εργαστηριακή Άσκηση

PARLAB13

Γεώργιος Πιττάκης el20605
Ηρακλής Νικολαΐδης el20607
Βίκτωρας Γιαννάκης el19707

Περιεχόμενα

1	Εισαγωγή	1
2	Περιγραφή Προβλήματος	1
3	Παράλληλη Υλοποίηση με OpenMP	1
4	Πειραματική Διάταξη	1
4.1	Διαστάσεις Πλέγματος και Νήματα	1
4.2	Μετρήσεις Απόδοσης	2
5	Αποτελέσματα	2
5.1	Πλέγμα Διαστάσεων 64×64	2
5.2	Πλέγμα Διαστάσεων 1024×1024	2
5.3	Πλέγμα Διαστάσεων 4096×4096	2
6	Ανάλυση	4
7	Συμπεράσματα	4

1 Εισαγωγή

Σε αυτήν την αναφορά, συζητάμε την υλοποίηση και την ανάλυση επιδόσεων του Game of Life του Conway, ενός κλασικού παραδείγματος cellular automaton, με τη χρήση παράλληλης επεξεργασίας. Ο κύριος στόχος ήταν ο παραλληλισμός του προγράμματος με χρήση OpenMP και η αξιολόγηση των επιδόσεων με διαφορετικά μεγέθη πινάκων και πλήθος νημάτων.

2 Περιγραφή Προβλήματος

Το Παιχνίδι της Ζωής διεξάγεται σε ένα ορθογώνιο πλέγμα όπου κάθε κελί μπορεί να είναι είτε "ζωντανό" είτε "νεκρό". Η κατάσταση κάθε κελιού στην επόμενη γενιά εξαρτάται από τους οκτώ γείτονές του. Οι κανόνες αλλαγής κατάστασης είναι οι εξής:

- Ένα ζωντανό κελί πεθαίνει από μοναξιά αν έχει λιγότερους από δύο γείτονες.
- Ένα ζωντανό κελί επιβιώνει αν έχει δύο ή τρεις γείτονες.
- Ένα ζωντανό κελί πεθαίνει από υπερπληθυσμό αν έχει περισσότερους από τρεις γείτονες.
- Ένα νεκρό κελί με ακριβώς τρεις γείτονες ζωντανεύει λόγω αναπαραγωγής.

Το πρόβλημα υλοποιήθηκε σε πίνακα διαστάσεων $N \times N$ με προσομοίωση T χρονικών βημάτων.

3 Παράλληλη Υλοποίηση με OpenMP

Χρησιμοποιήσαμε το OpenMP για να παραλληλοποιήσουμε τον αλγόριθμο, εκμεταλλευόμενοι την αρχιτεκτονική πολλαπλών πυρήνων των σύγχρονων επεξεργαστών. Η παράλληλη επεξεργασία εφαρμόστηκε στον διπλό βρόχο που επεξεργάζεται κάθε κελί του πλέγματος όπως φαίνεται πιο κάτω.

Κώδικας:

```
#pragma omp parallel for shared(N, previous, current) private(i, j, nbrs)
for ( i = 1 ; i < N-1 ; i++ )
    for ( j = 1 ; j < N-1 ; j++ ) {
        nbrs = previous[i+1][j+1] + previous[i+1][j] + previous[i+1][j-1] +
                previous[i][j-1] + previous[i][j+1] +
                previous[i-1][j-1] + previous[i-1][j] + previous[i-1][j+1];
        if ( nbrs == 3 || ( previous[i][j]+nbrs ==3 ) )
            current[i][j]=1;
        else
            current[i][j]=0;
    }
```

4 Πειραματική Διάταξη

4.1 Διαστάσεις Πλέγματος και Νήματα

Το πρόγραμμα δοκιμάστηκε σε τρεις διαστάσεις πλέγματος:

- 64×64
- 1024×1024
- 4096×4096

Για κάθε μέγεθος πλέγματος, το πρόγραμμα εκτελέστηκε με 1, 2, 4, 6 και 8 νήματα για να αξιολογηθούν οι επιδόσεις.

4.2 Μετρήσεις Απόδοσης

Οι χρόνοι εκτέλεσης μετρήθηκαν για κάθε συνδυασμό μεγέθους πλέγματος και αριθμού νημάτων, με τη χρήση 1000 γενεών κάθε φορά.

5 Αποτελέσματα

Τα αποτελέσματα του πειράματος παρουσιάζονται παρακάτω. Παρατηρείται σταθερή μείωση στον χρόνο εκτέλεσης καθώς αυξάνεται ο αριθμός των νημάτων, ιδιαίτερα για μεγαλύτερα μεγέθη πλέγματος.

5.1 Πλέγμα Διαστάσεων 64×64

Νήματα	Χρόνος (δευτερόλεπτα)
1	0.023094
2	0.013743
4	0.010129
6	0.009534
8	0.009883

5.2 Πλέγμα Διαστάσεων 1024×1024

Νήματα	Χρόνος (δευτερόλεπτα)
1	10.969119
2	5.459525
4	2.723377
6	1.830102
8	1.378207

5.3 Πλέγμα Διαστάσεων 4096×4096

Νήματα	Χρόνος (δευτερόλεπτα)
1	175.895132
2	88.222330
4	44.600186
6	35.545087
8	32.751054

Οι αντίστοιχες γραφικές παραστάσεις για τα παραπάνω αποτελέσματα εμφανίζονται παρακάτω:

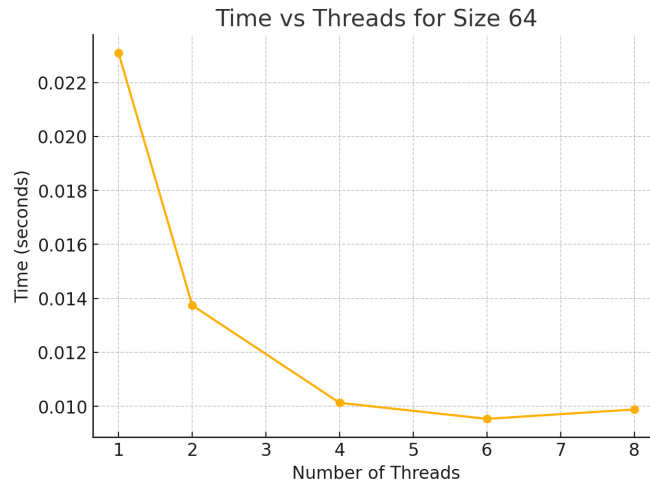


Figure 1: Χρόνος vs Νήματα για Πλέγμα 64×64

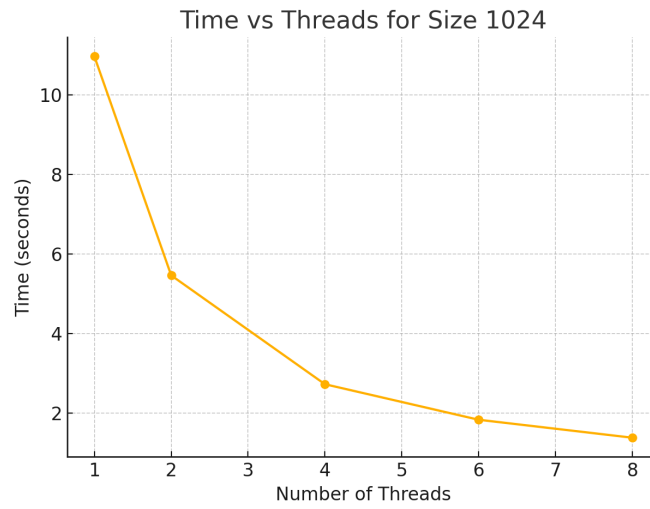


Figure 2: Χρόνος vs Νήματα για Πλέγμα 1024×1024

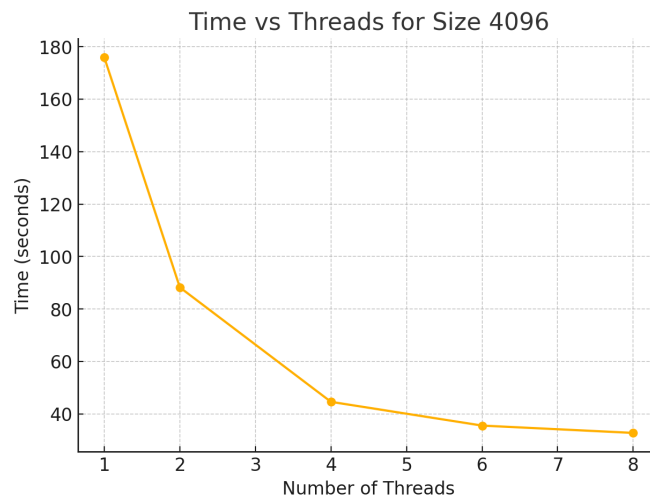


Figure 3: Χρόνος vs Νήματα για Πλέγμα 4096×4096

6 Ανάλυση

Από τα αποτελέσματα, παρατηρούμε σημαντικές βελτιώσεις στην απόδοση με την αύξηση των νημάτων. Ωστόσο, για το μικρότερο μέγεθος πλέγματος (64×64), τα κέρδη από την παράλληλη επεξεργασία μειώνονται μετά τα 4 νήματα λόγω του ότι το κόστος επικοινωνίας και ο χρόνος που απαιτείται για τη δημιουργία των νημάτων είναι κοντά στην τάξη μεγέθους επίλυσης του προβλήματος. Για τα μεγαλύτερα μεγέθη πλέγματος (1024×1024 και 4096×4096), η ταχύτητα συνεχίζει να αυξάνεται με την προσθήκη περισσότερων νημάτων. Παρατηρούμε ότι στο μέγεθος πλέγματος 4096×4096 δεν επιτυγχάνεται κλιμάκωση πιθανός λόγο συμφόρισης στο διάδρομο μνήμης και λόγω των cache miss.

Παρά τα σημαντικά κέρδη επιδόσεων που παρατηρούμε με την αύξηση του αριθμού των νημάτων, οι μετρήσεις μας δείχνουν ότι η παράλληλη υλοποίηση δεν κλιμακώνεται γραμμικά. Αυτό σημαίνει ότι η απόδοση δεν βελτιώνεται αναλογικά με τον αριθμό των νημάτων που χρησιμοποιούνται, ιδιαίτερα σε μικρότερα μεγέθη πινάκων όπως το 64×64 . Οι λόγοι που μπορεί να ευθύνονται για αυτό είναι οι εξής:

- Υπάρχει σειριακό μέρος
- Κόστος συγχρονισμού / επικοινωνίας
- Κόστος από την παραλληλοποίηση
- Συμφόρηση στο διάδρομο μνήμης

7 Συμπεράσματα

Η παράλληλη υλοποίηση του Game of Life με χρήση OpenMP έφερε σημαντικές βελτιώσεις στις επιδόσεις, ιδιαίτερα για μεγαλύτερα μεγέθη πλέγματος. Όπως αναμενόταν, τα οφέλη του παραλληλισμού γίνονται πιο εμφανή όσο αυξάνεται το μέγεθος του προβλήματος και τα νήματα αξιοποιούνται πιο αποτελεσματικά.