OpenMP

11/3/2019

Μπαγκής Ευάγγελος

Πίνακας 1 Αποτελέσματα της μεθόδου Liebmann

| **Number of threads** | **U(x/2, y/2)=U(0.5, 0.5)** | **Number of iterations** | **Time** |
|---|---|---|---|
| | | *while* | |
| 1 Thread | 4.332206 | 112605 | 6m12,432s |
| 2 Threads | 4.332130 | 112891 | 4m54,860s |
| 4 Threads | 4.332181 | 113089 | 3m27,791s |
| 8 Threads | 4.332204 | 113457 | 3m36,842s |

Πίνακας 1 Αποτελέσματα της μεθόδου Successive Over Relaxation για ω=1, ω1.95, ω=1.99

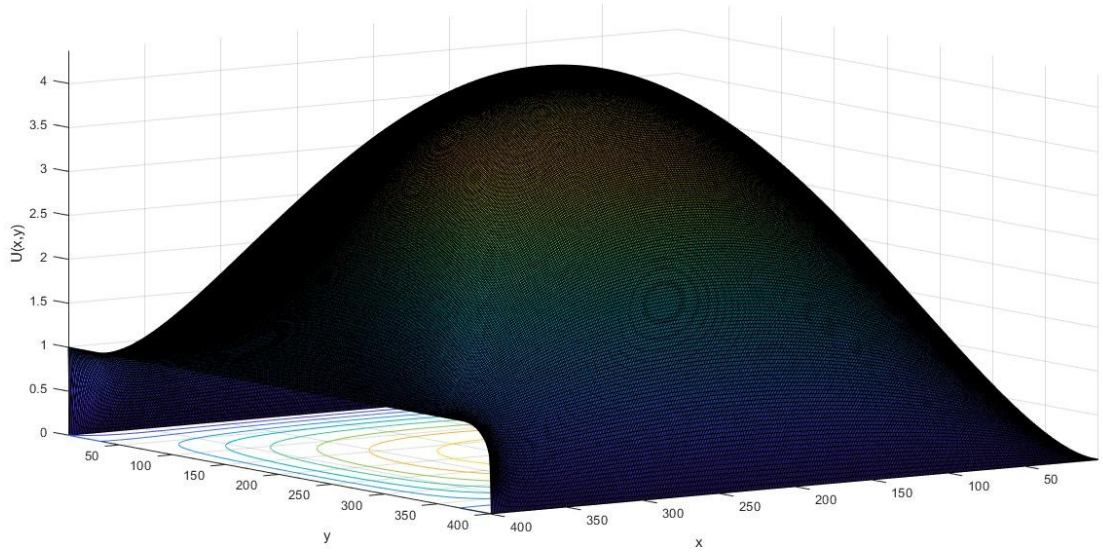
| **Number of threads** | **w** | **U(x/2, y/2)=U(0.5, 0.5)** | **Number of iterations** | **Time** |
|---|---|---|---|---|
| | | | *while* | |
| 1 Thread | 1 | 4.332203 | 112581 | 9m58,722s |
| 2 Threads | 1 | 4.33220 | 112581 | 6m11,897s |
| 4 Threads | 1 | 4.332203 | 112581 | 5m23,229s |
| 8 Threads | 1 | 4.332203 | 112581 | 5m13,679s |
| | | | | |
| 1 Thread | 1.95 | 4.337011 | 3904 | 0m20,639s |
| 2 Threads | 1.95 | 4.337011 | 3904 | 0m12,982s |
| 4 Threads | 1.95 | 4.337011 | 3904 | 0m11,123s |
| 8 Threads | 1.95 | 4.336987 | 3904 | 0m10,842s |
| | | | | |
| 1 Thread | 1.99 | 4.339143 | 785 | 0m4,181s |
| 2 Threads | 1.99 | 4.339143 | 785 | 0m2,621s |
| 4 Threads | 1.99 | 4.339143 | 785 | 0m2,293s |
| **8 Threads** | **1.99** | **4.339175** | **785** | **0m2,231s** |

Πίνακας 3 Παράλληλη επιτάχυνση

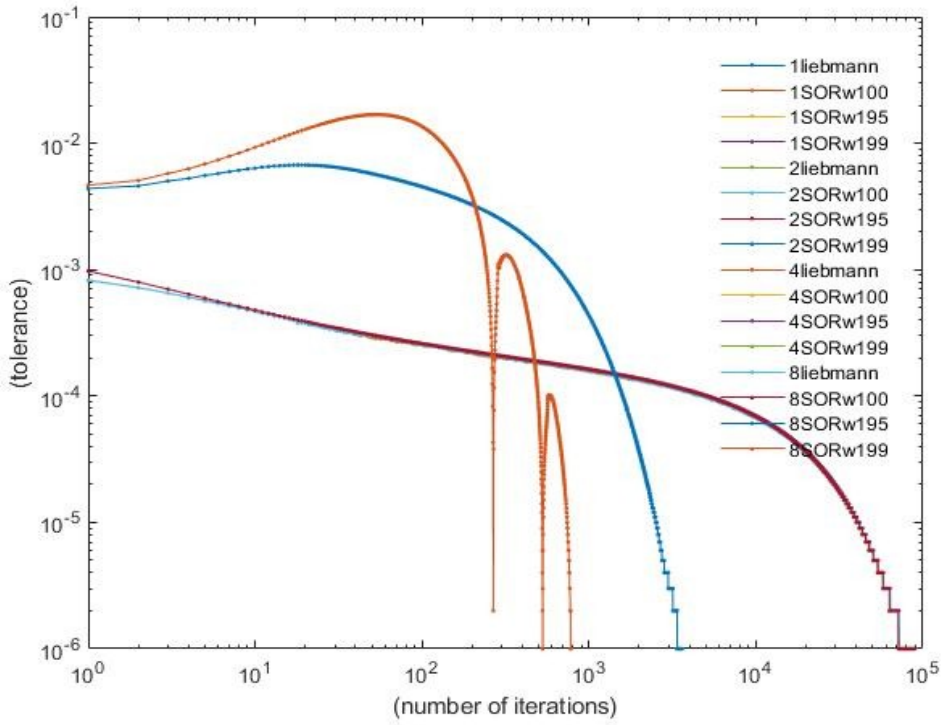
| **Number of threads** | **Parallel speedup Liebmann** | **Parallel speedup SOR w=1** | **Parallel speedup SOR w=1.95** | **Parallel speedup SOR w=1.99** |
|---|---|---|---|---|
| | | | | |
| 2 Threads | 1.2631 | 1.6099 | 1.5898 | 1.5952 |
| 4 Threads | 1.7923 | 1.8523 | 1.8555 | 1.8234 |
| 8 Threads | 1.7175 | 1.9087 | 1.9036 | 1.8740 |

Πίνακας 4 Απόδοση παράλληλης επεξεργασίας

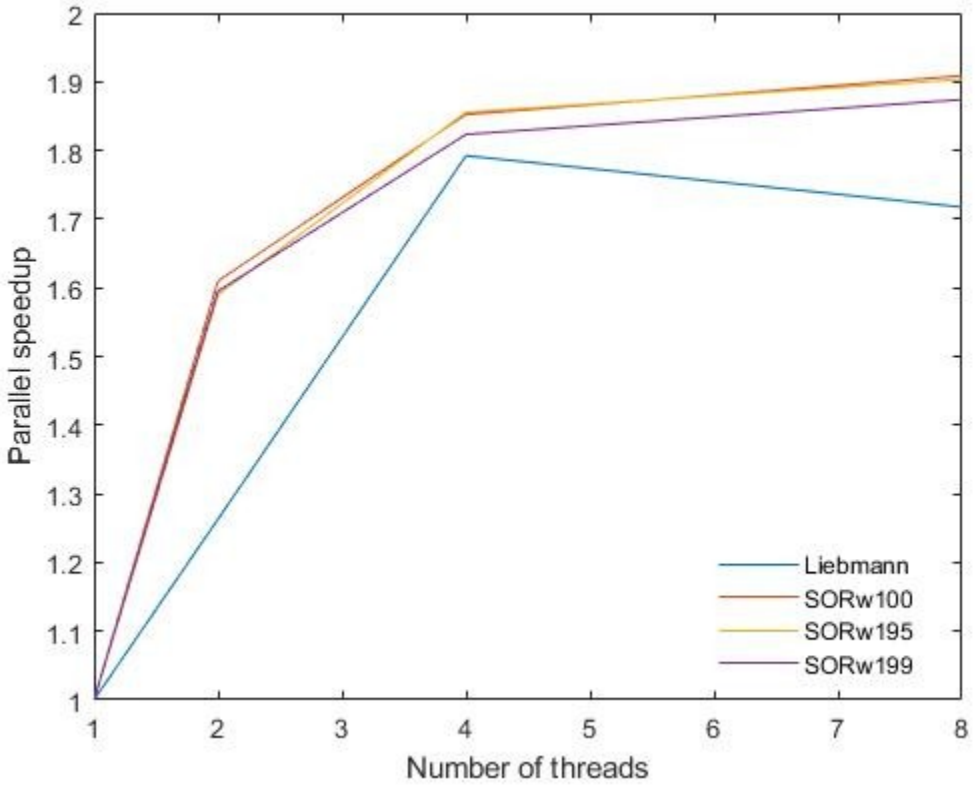
| **Number of threads** | **Parallel efficiency Liebmann** | **Parallel efficiency SOR w=1** | **Parallel efficiency SOR w=1.95** | **Parallel efficiency SOR w=1.99** |
|---|---|---|---|---|
| | % | % | % | % |
| 2 Threads | 63.15 | 80.5 | 79.49 | 79.76 |
| 4 Threads | 44.81 | 46.31 | 46.39 | 45.58 |
| 8 Threads | 21.47 | 23.86 | 23.8 | 23.43 |



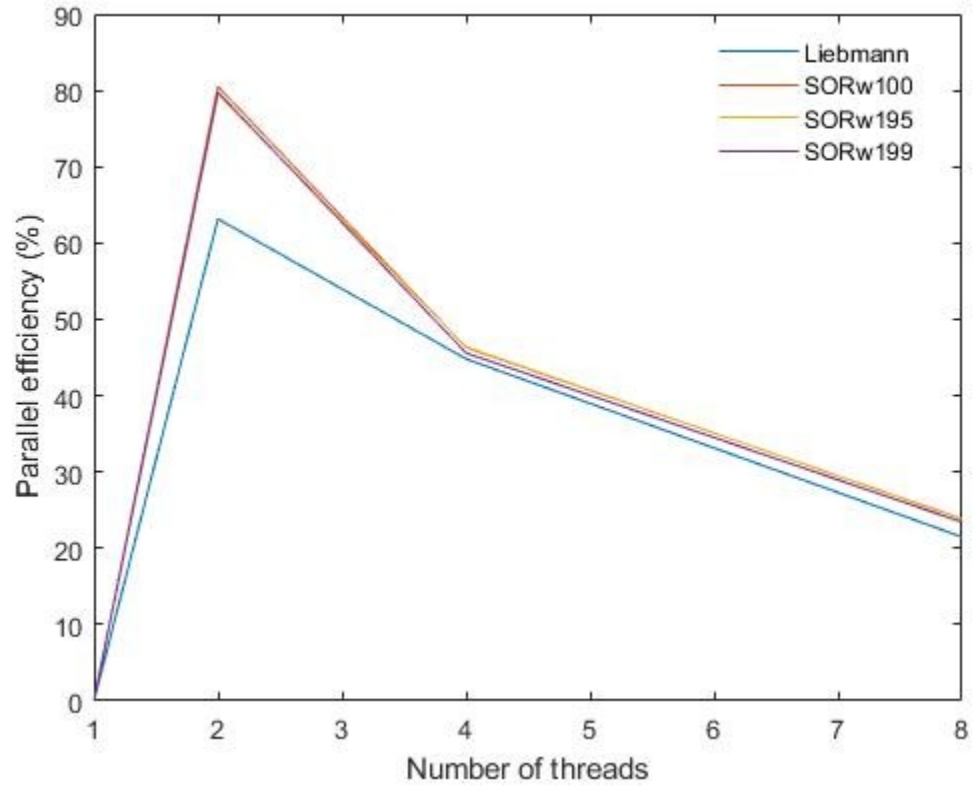

Σχήμα 1 Η λύση της ελλειπτικής διαφορικής εξίσωσης με την μέθοδο SOR ω=1.99.




Σχήμα 2 Λογαριθμικό διάγραμμα για την ακρίβεια των λύσεων σε σχέση με τον αριθμό των επαναλήψεων για τις 16 περιπτώσεις. Ο αριθμός μπροστά από τα ονόματα είναι τα νήματα.




Σχήμα 3 Παράλληλη επιτάχυνση σε σχέση με τον αριθμό των νημάτων




Σχήμα 4 Απόδοση παράλληλης επεξεργασίας

# Σχόλια

Χρησιμοποιήθηκε επεξεργαστής Intel core I - 3 5005U με 2 φυσικούς πυρήνες.

Στο σχήμα 1 απεικονίζεται η λύση της διαφορικής εξίσωσης $\nabla^2 U = S(x,y) = -10(x^2+y^2+5)$ με συνοριακές συνθήκες $U(0,y)=0, U(1,y)=0, U(x,0)=0, U(x,1)=1$ που σχεδιάστηκε με το πακέτο Matlab.

Από το σχήμα 2 παρατηρούμε ότι η tolerance για την μέθοδο Liebmann και για την SOR red-black με ω=1 (ουσιαστικά εφαρμόζουμε και πάλι την μέθοδο Liebmann σε 2 βήματα χρησιμοποιώντας τις νέες τιμές στα σημεία που έχουν ήδη υπολογιστεί) ακολουθεί την ίδια πορεία και ο αριθμός των επαναλήψεων είναι ο ίδιος (112000 περίπου). Ακόμα βλέπουμε ότι για ω=1.95 και ω=1.99 η tolerance ξεκινάει από υψηλότερες τιμές και αρχικά μοιάζουν να αποκλίνουν αλλά πολύ γρήγορα οι μέθοδοι συγκλίνουν και η tolerance τείνει πολύ γρηγορότερα στην ζητούμενη ακρίβεια (μετά από ~3900 και ~800 επαναλήψεις αντίστοιχα). Βλέπουμε λοιπόν ότι το ω μειώνει τον χρόνο εκτέλεσης κατά ένα τεράστιο ποσοστό. Ο αριθμός των νημάτων που χρησιμοποιούμε δεν παίζει κάποιο ρόλο στην tolerance και έτσι οι καμπύλες συμπίπτουν για την ίδια μέθοδο ανεξαρτήτως από τον αριθμό των νημάτων άρα φαίνονται μόνο 4 καμπύλες αντί για 16. Τέλος, αν η ζητούμενη ακρίβεια ήταν της τάξεως του $10^{-3}$ οι δύο πρώτες μέθοδοι θα συγκλίνουν γρηγορότερα από τις δύο τελευταίες.

Η παράλληλη επιτάχυνση για την μέθοδο Liebmann αρχικά αυξάνει για 2 και 4 νήματα όπου φτάνει και στο μέγιστο (σχήμα 3). Από εκεί και πέρα αρχίζει να φθίνει και πλέον ο συγχρονισμός των νημάτων επιβαρύνει τους υπολογισμούς περισσότερο από ότι τους βοηθάει. Για τις υπόλοιπες 3 μεθόδους οι καμπύλες ακολουθούν παρόμοια συμπεριφορά. Αρχικά παρατηρούμε μια ραγδαία αύξηση στην επιτάχυνση για 2 νήματα. Η αύξηση συνεχίζεται μέχρι να φτάσουμε στα 4 νήματα αλλά με μικρότερη κλίση. Από τα 4 νήματα και πάνω η επιτάχυνση συνεχίζει να έχει ανοδική πορεία αλλά η κλίση είναι πλέον πάρα πολύ μικρή και φτάνουμε σε κορεσμό.

Στο τελευταίο σχήμα (4) απεικονίζεται η απόδοση της παράλληλης επεξεργασίας που ακολουθεί παρόμοια συμπεριφορά και για τις 4 μεθόδους. Αρχικά για 1 νήμα η απόδοση είναι μηδέν αφού ο αλγόριθμος τρέχει σειριακά. Η απόδοση για 2 νήματα φτάνει το μέγιστο και αυτό είναι αποτέλεσμα του διπύρηνου επεξεργαστή. Από εκεί και πέρα η απόδοση μειώνεται αλλά παραμένει θετική. Αυτή η μείωση οφείλεται και πάλι στον χρόνο που σπαταλάει ο επεξεργαστής για να συγχρονίσει τα νήματα.

Συμπερασματικά η μέθοδος SOR red – black συγκλίνει γρηγορότερα από την μέθοδο Liebmann με όσα νήματα κι αν τρέξουμε τον κώδικα. Επίσης, η μέγιστη απόδοση επιτυγχάνεται για 2 νήματα στον συγκεκριμένο επεξεργαστή. Η επιτάχυνση είναι μονότονα αύξουσα για τον SOR αλλά για την Liebmann υπάρχει μέγιστο στα 4 νήματα. Τέλος, η παράμετρος (ω) μεταβάλει τους υπολογισμούς έτσι ώστε να υπάρχουν μεγάλες μεταβολές αρχικά αλλά τελικά η tolerance συγκλίνει πολύ πιο γρήγορα.