Report: OPEN MP Problem Set

Δημήτριος Ψαρράς
AEM: 4407

Μάθημα: Εργαλεία Προγραμματισμού

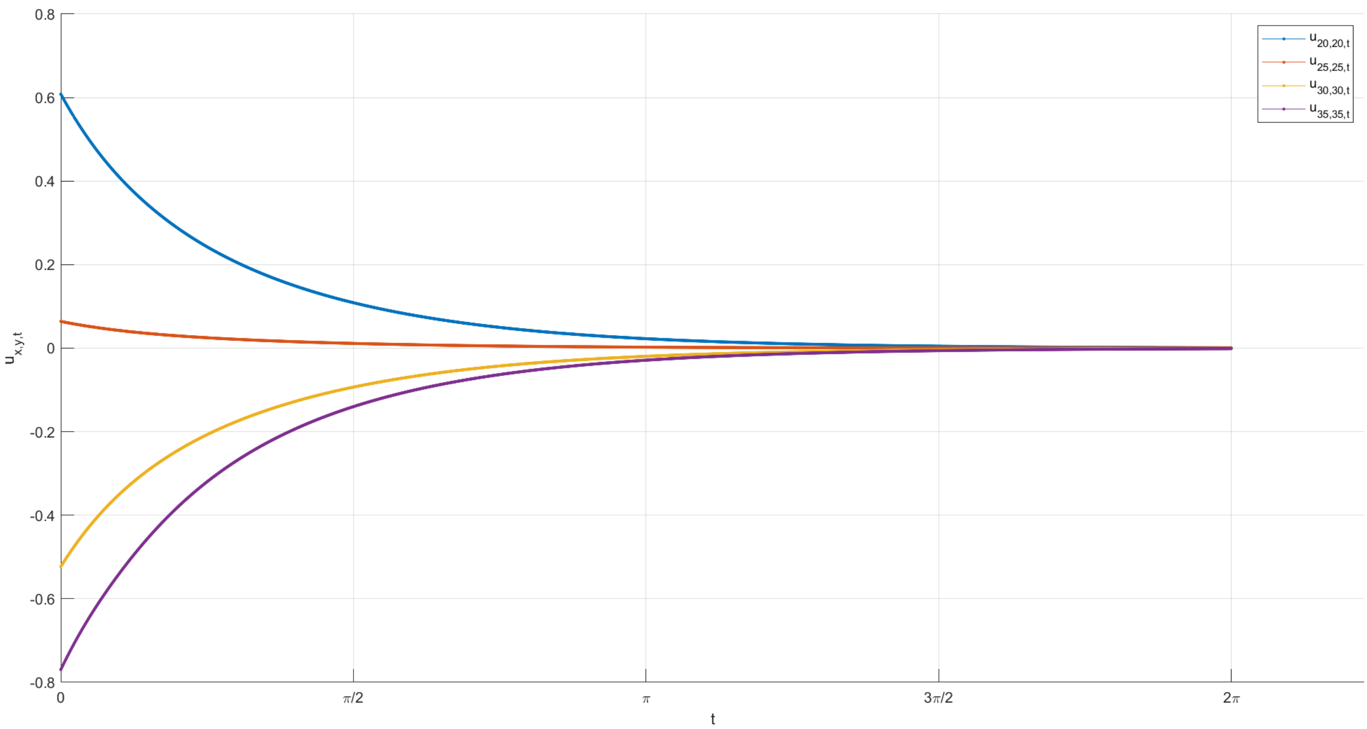
Άσκηση 1η

Στη πρώτη άσκηση ζητάτε να λυθεί η παραβολική εξίσωση $u_t = \varkappa(u_{xx} + u_{yy})$ με $\varkappa = 4/5$ με χρήση της μεθόδου Forward-Time Centered-Space (FTCS).

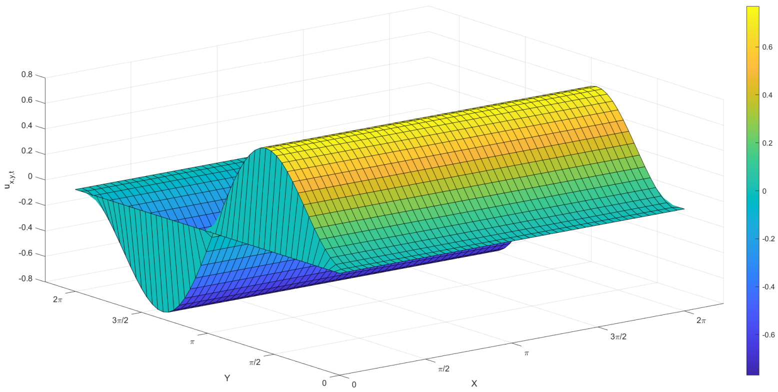
$$u_{i,j}^{n+1} = u_{i,j}^{n} + \varkappa\Delta t\left(\frac{u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}}{\Delta x^2} + \frac{u_{i,j+1}^{n} - 2u_{i,j}^{n} + u_{i,j-1}^{n}}{\Delta y^2}\right) \tag{1}$$

Τα διαστήματα των συντεταγμένων x και y είναι [0, 2π] και οι αρχικές και συνοριακές συνθήκες είναι $u(x, y, 0) = sinxsin\frac{y}{2}$ και $u(0, y, t) = u(2\pi, y, t) = u(x, 0, t) = u(x, 2\pi, t) = 0$, αντίστοιχα. Τέλος, το χρονικό βήμα για την μέθοδο λαμβάνεται ίσο με $dt = \frac{\pi}{1200}$.

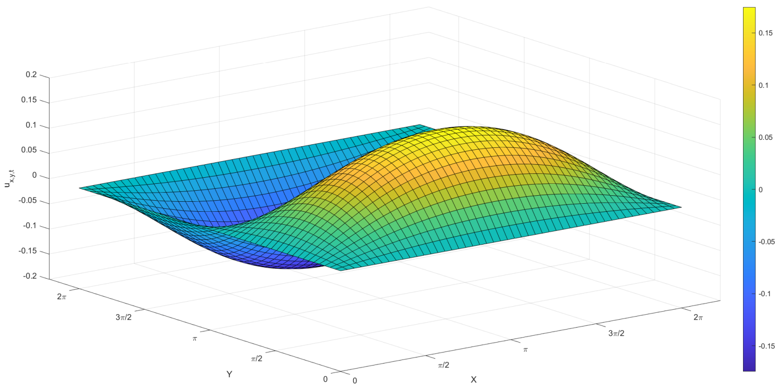
Αρχικά ο κώδικας για την επίλυση του προβλήματος αυτού υλοποιήθηκε ως σειριακό πρόγραμμα και βρίσκεται στο αρχείο **Exer1_Psarras_single.c**. Εκτελώντας το πρόγραμμα για $N \times M$ (N=50 και M=50) πλεγματικά σημεία επιστρέφεται ένα αρχείο .csv το οποίο περιέχει τα απαιτούμενα δεδομένα για τον σχεδιασμό των ζητούμενων διαγραμμάτων. Αρχικά παρουσιάζεται η λύση $u(x, y, t)$ τεσσάρων σημείων (20,20), (25,25), (30,30) και (35,35) συναρτήσει του χρόνου και μετά παρατίθεται το διάγραμμα της λύσης $u(x, y, t)$ για t=0,π/2,π,2π σε διαφορετικά σχήματα.



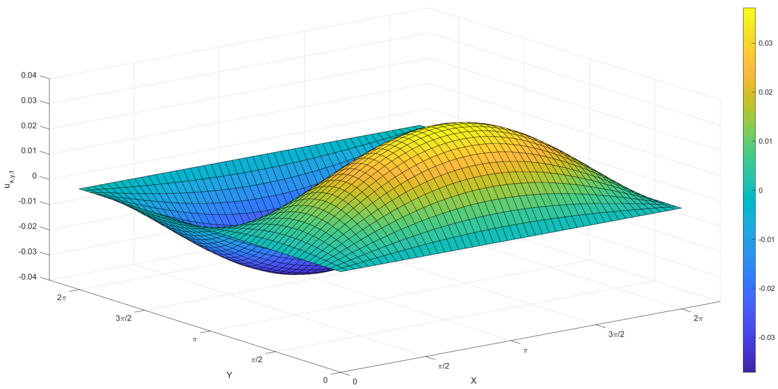
Σχήμα 1: Λύση $u(x, y, t)$ για τέσσερα διαφορετικά σημεία (20,20), (25,25), (30,30) και (35,35) συναρτήσει του χρόνου.



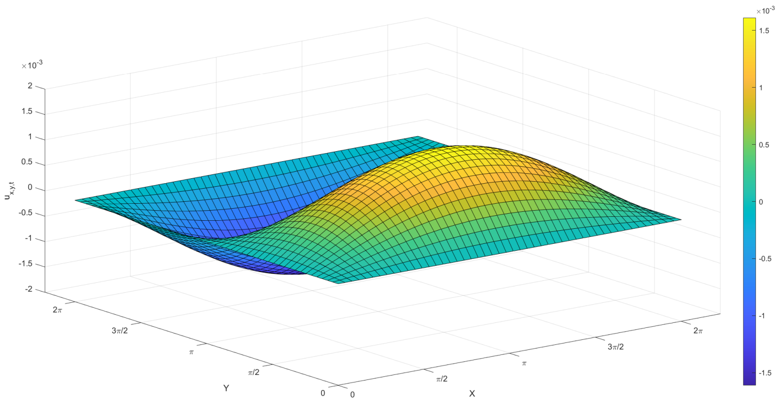
(α′)



(β′)



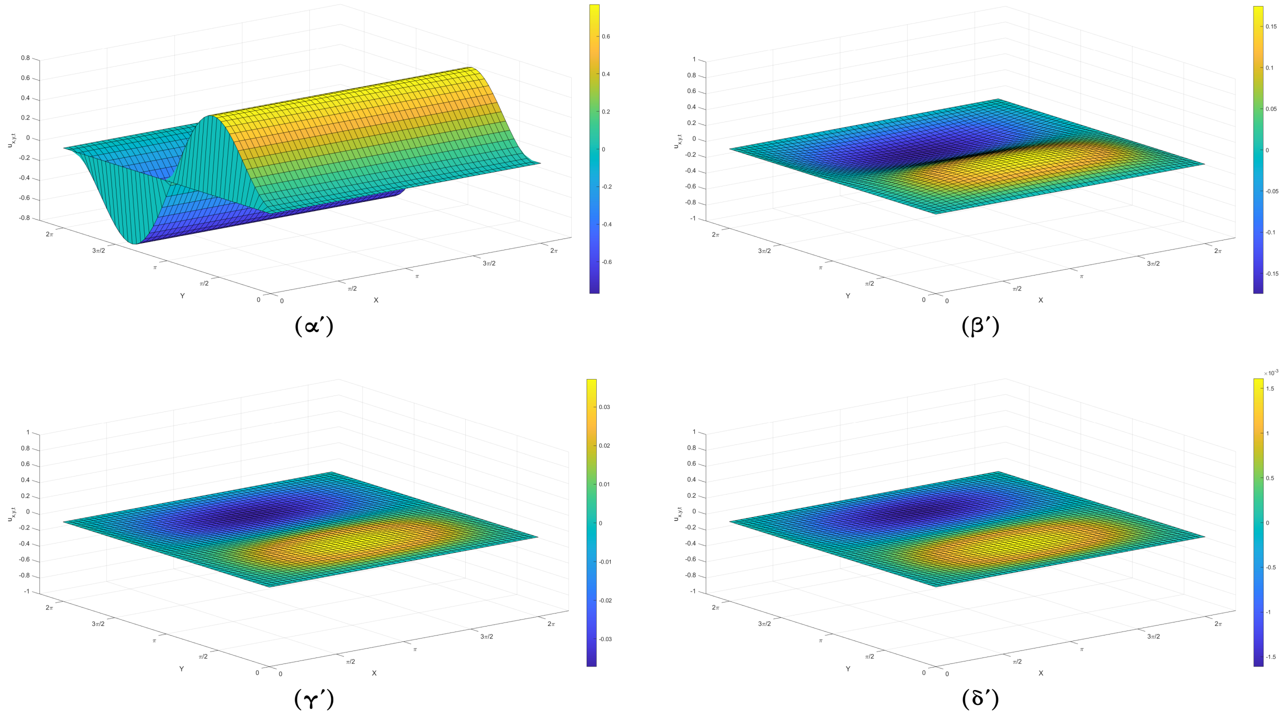
(γ′)



(δ′)

Σχήμα 2: Διάγραμμα της επιφάνειας της λύσης $u(x, y, t)$ (α′) Για την χρονική στιγμή t=0. (β′) Για την χρονική στιγμή t=π/2. (γ′) Για την χρονική στιγμή t=π. (δ′) Για την χρονική στιγμή t=2π.

Είναι σημαντικό να παρατηρηθεί ότι στα διαγράμματα του σχήματος 2, όπου το κίτρινο χρώμα αντιπροσωπεύει την υψηλή θερμοκρασία και το μπλε χρώμα την χαμηλή, η μέγιστη και η ελάχιστη τιμή της λύσης $u(x, y, t)$ μειώνεται καθώς αυξάνεται ο χρόνος. Επομένως, το σύστημα, όπου αρχικά η δεξιά περιοχή είναι κίτρινη και η αριστερή μπλε, με την πάροδο του χρόνου προσεγγίζει μια κατάσταση ισορροπίας όπου η θερμοκρασιακή διαφορά των δύο περιοχών μειώνεται σημαντικά. Αυτό το φαινόμενο δεν φαίνεται καθαρά στα διαγράμματα του σχήματος 2 καθώς το εύρος των τιμών του άξονα z δεν παραμένει σταθερό για τις διαφορετικές χρονικές στιγμές. Έτσι σχεδιάστηκαν και τα διαγράμματα του σχήματος 3 όπου το εύρος τιμών του άξονα z παραμένει σταθερό και ίσο με [-1,1].



Σχήμα 3: Διάγραμμα της επιφάνειας της λύσης $u(x, y, t)$ (εύρος τιμών του άξονα z είναι [-1,1] (α′) Για την χρονική στιγμή t=0. (β′) Για την χρονική στιγμή t=π/2. (γ′) Για την χρονική στιγμή t=π. (δ′) Για την χρονική στιγμή t=2π.

Επιπρόσθετα, προκειμένου να δειχθεί ότι ο ρυθμός σύγκλισης συμφωνεί με τον αναμενόμενο το πρόγραμμα του αρχείου **Exer1_Psarras_single.c** εκτελέστηκε για 25×25, 50×50 και 100×100 πλεγματικά σημεία. Έτσι, εξετάσθηκαν 3 περιπτώσεις όπου το βήμα για x και y είναι h για 100×100 πλεγματικά σημεία, 2h για 50×50 και 4h για 25×25. Ειδικότερα, για την μέθοδο που χρησιμοποιήθηκε η τάξη του σφάλματος είναι $\mathrm{O}(dt, h_x^2, h_y^2)$ και άρα τάξης 1. Επομένως, ο λόγος:

$$\left|\frac{u_{2h} - u_{4h}}{u_h - u_{2h}}\right| = 2 \tag{2}$$

για την μελέτη του ρυθμού σύγκλισης αναμμένεται να ισούται με 2. Έτσι υπολογίσθηκε για την περίπτωση 25×25, 50×50 και 100×100 πλεγματικών σημείων ότι η λύση ισούται με $u(10, 10, \frac{\pi}{2}) = 0.1179$, $u(20, 20, \frac{\pi}{2}) = 0.1090$ και $u(40, 40, \frac{\pi}{2}) = 0.1044$. Άρα το κλάσμα ισούται με:

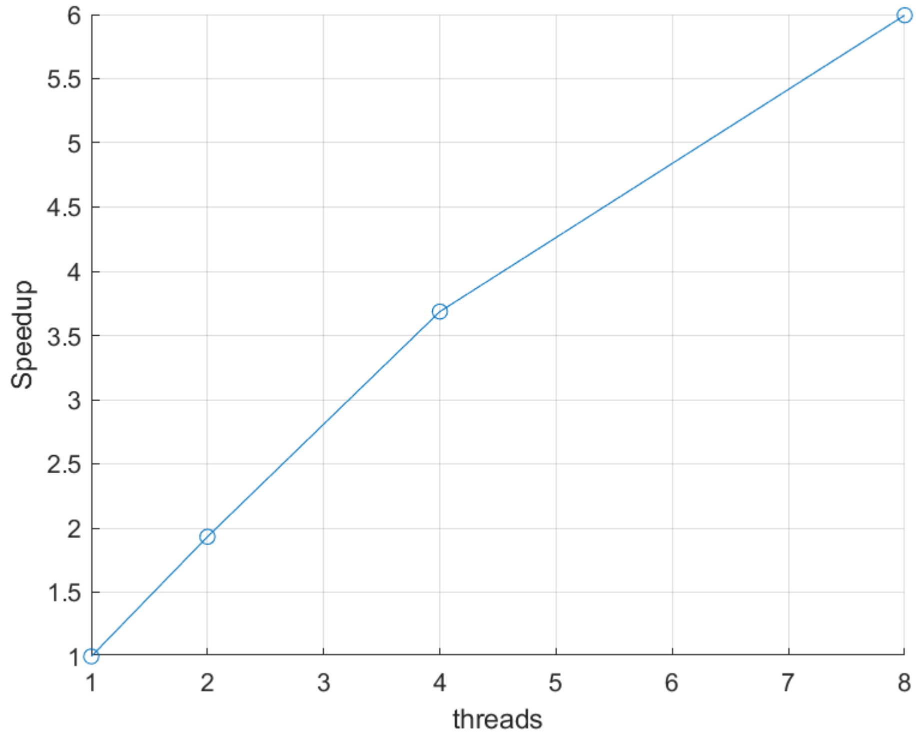
$$\left|\frac{u_{2h} - u_{4h}}{u_h - u_{2h}}\right| = \frac{0.1179 - 0.1090}{0.1090 - 0.1044} = \frac{0.0089}{0.0046} = 1.93 \tag{3}$$

Επομένως, ο ρυθμός σύγκλισης συμφωνεί με τον αναμενόμενο.

Ακολούθως, το πρόγραμμα αλλάχτηκε καταλλήλως με χρήση OpenMP προκειμένου να μειωθεί ο χρόνος εκτέλεσης του. Η λύση βρίσκεται στο αρχείο **Exer1_Psarras_parallel.c**. Αρχικά επιβεβαιώνεται η σωστή λειτουργία του παραλληλοποιημένου προγράμματος για $N \times M$ (N=50 και M=50) πλεγματικά σημεία και στη συνέχεια αυξάνεται ο αριθμός των πλεγματικών σημείων προκειμένου να συγκριθεί ο χρόνος εκτέλεσης για 1, 2, 4, 8 threads. Ειδικότερα, ο αριθμός των πλεγματικών σημείων αυξήθηκε στα $N \times M$ (N=500 και M=500) και το χρονικό βήμα έγινε dt=π/3600. Παρακάτω παρουσιάζονται τα αποτελέσματα στο Πίνακα 1 και σε ένα σχεδιάγραμμα, Σχήμα 4.

Πίνακας 1: Αποτελέσματα για $N \times M$(N=500 και M=500) πλεγματικά σημεία

Threads	Time (sec)	Speedup
1	246.114	1
2	127.362	1.932
4	66.764	3.686
8	41.059	5.994



Σχήμα 4: Επιτάχυνση (speedup) λόγο παράλληλης επεξεργασίας συναρτήσει του αριθμού των threads για $N \times M$(N=500 και M=500) πλεγματικά σημεία

Άσκηση 2η

Στην δεύτερη άσκηση δόθηκε στο φοιτητή το αρχείο **matmul.c** προκειμένου να παραλληλοποιηθεί. Η λύση βρίσκεται στο αρχείο **matmul_Psarras_parallel.c**. Προκειμένου να επιβεβαιωθεί η ορθή λειτουργία του παραλληλοποιημένου προγράμματος συγκρίθηκαν τα αποτελέσματα του με το σειριακό πρόγραμμα για ένα απλό παράδειγμα. Ειδικότερα θεωρήθηκε ο πολλαπλασιασμός δύο 3×3 πινάκων A και B:

$$A = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 3 & 3 \\ 3 & 3 & 3 \end{pmatrix} \tag{4}$$

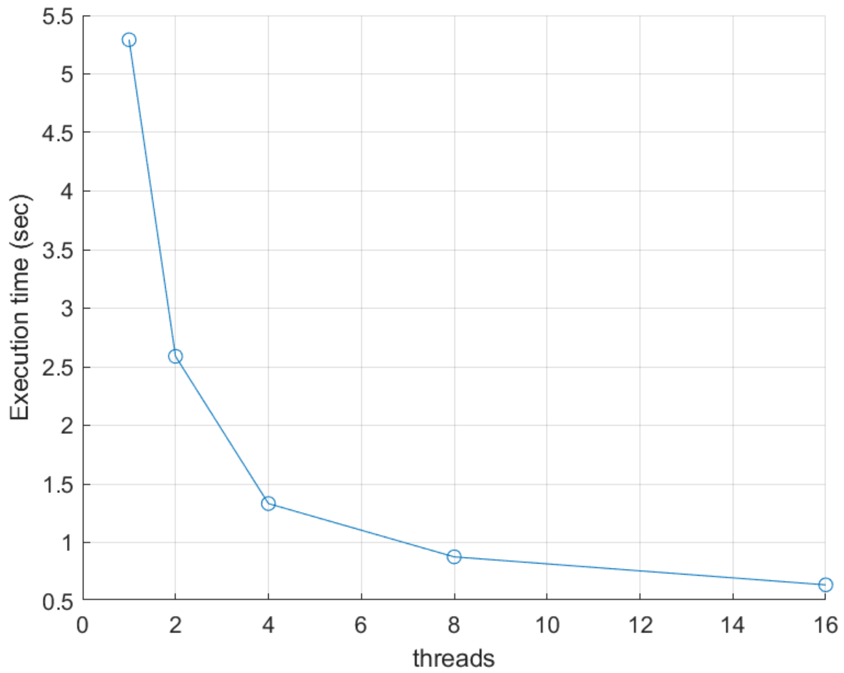
$$B = \begin{pmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{pmatrix} \tag{5}$$

$$A \cdot B = \begin{pmatrix} 18 & 18 & 18 \\ 18 & 18 & 18 \\ 18 & 18 & 18 \end{pmatrix} \tag{6}$$

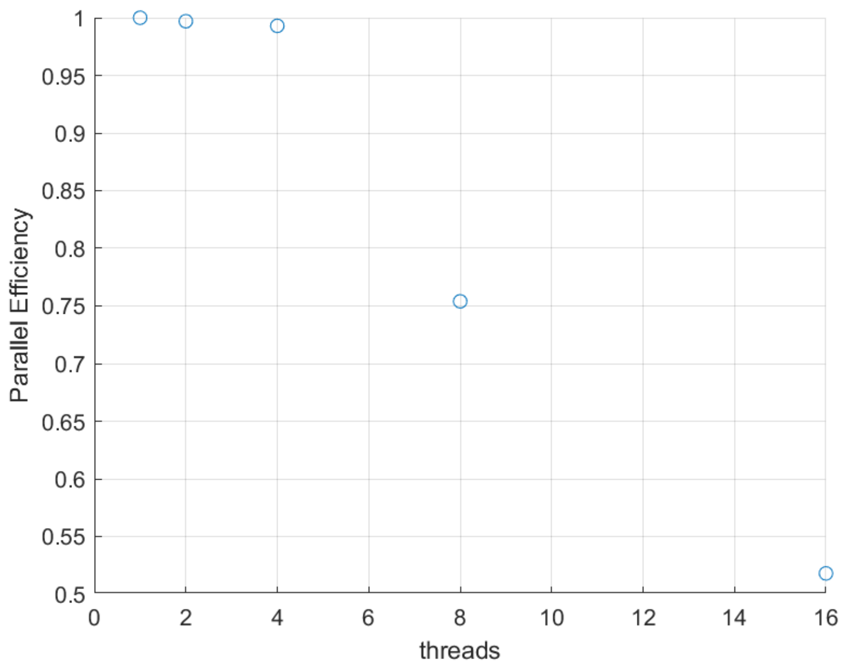
Με το παραπάνω παράδειγμα διασφαλίζεται ότι το νέο παραλληλοποιημένο πρόγραμμα εκτελείτε χωρίς να προκύπτει κάποιο σφάλμα. Έτσι, έχοντας επιβεβαιώσει την σωστή λειτουργία του παράλληλου προγράμματος, το πρόγραμμα εκτελείται για μεγαλύτερο αριθμό γραμμών και στηλών προκειμένου να χρονομετρηθεί η εκτέλεση του για διαφορετικό αριθμό threads. Ειδικότερα, χρησιμοποιήθηκαν πίνακας με χίλιες (1000) σειρές και στήλες.

Πίνακας 2: Αποτελέσματα για $N \times M$(N=1000 και M=1000) τετραγωνικούς πίνακες

Threads	Time (sec)	Speedup	Parallel Efficiency
1	5.293	1	1
2	2.655	1.993	0.997
4	1.332	3.973	0.993
8	0.877	6.034	0.754
16	0.638	8.295	0.518



Σχήμα 5: Χρόνος εκτέλεσης του προγράμματος συναρτήσει του αριθμού των threads για 1000×1000 τετραγωνικούς πίνακες



Σχήμα 6: Βαθμός παράλληλης απόδοσης συναρτήσει του αριθμού των threads για 1000×1000 τετραγωνικούς πίνακες

Άσκηση 3η

Στη τρίτη άσκηση γίνεται χρήση αριθμητικών μεθόδων με σκοπό τον υπολογισμό ενός ολοκληρώματος. Στην παρούσα άσκηση χρησιμοποιήθηκε η μέθοδος του Simpson για την υλοποίηση της αριθμητικής ολοκλήρωσης. Ειδικότερα, υλοποιήθηκε η μέθοδος Simpson 1/3 με πολλαπλά βήματα για τον υπολογισμό του ολοκληρώματος σε συγκεκριμένο εύρος a έως b. Έτσι, για n σημεία του x και f(x) εντός των ορίων [a,b], υπάρχουν n-1 τμήματα του συγκεκριμένου εύρους τιμών. Έτσι το ολοκλήρωμα είναι δυνατό να προσεγγισθεί από την σχέση:

$$I \approx (b-a)\frac{f(x_0) + 4\sum_{i=1,3,5,...}^{n-1} f(x_i) + 2\sum_{j=2,4,6,...}^{n-1} f(x_j) + f(x_n)}{3n} \tag{7}$$

Έτσι με την χρήση αυτή της σχέσης θα υπολογισθεί το ολοκλήρωμα της συνάρτησης:

$$f(x) = xe^{2x} \tag{8}$$

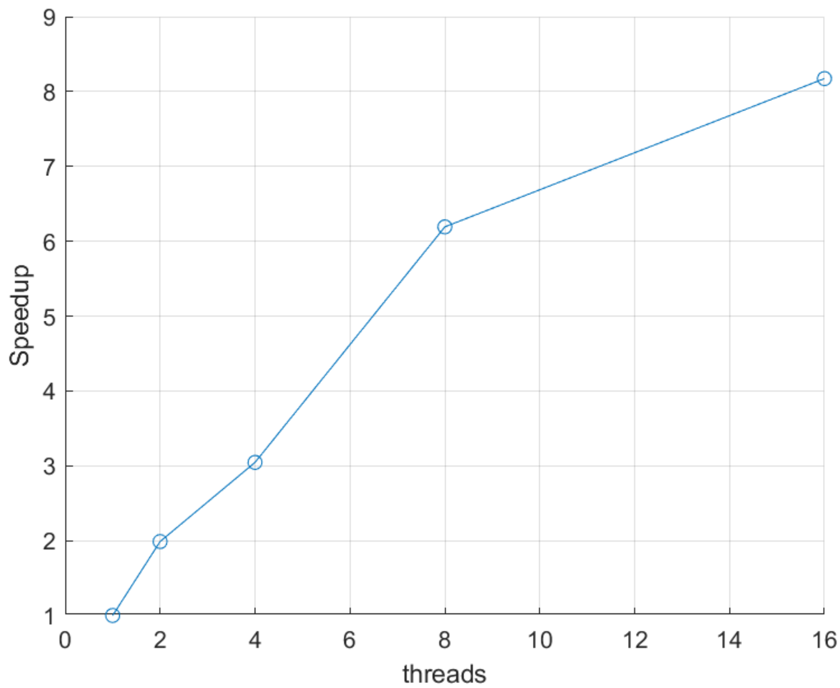
για το εύρος τιμών από 0 έως 3.

$$\int_0^3 xe^{2x}dx \tag{9}$$

Συγκεκριμένα, το εύρος [0,3] θα χωριστεί σε n=100000001 σημεία δεδομένων με αποτέλεσμα να χρειαστεί να υπολογισθούν 100000000 τμήματα για την μέθοδο του Simpson 1/3. Ο κώδικας που υλοποιεί τους υπολογισμούς αυτούς βρίσκεται στο αρχείο **Exer3_Psarras_simpsonrule.c**. Τα αποτελέσματα από την εκτέλεση του προγράμματος με την χρήση πολλών threads παρουσιάζονται στο πίνακα 3 και στο σχήμα 7.

Πίνακας 3: Αποτελέσματα για N=100000000 τμήματα

Threads	Time (sec)	Speedup	Parallel Efficiency
1	1.839	1	1
2	0.925	1.988	0.994
4	0.604	3.045	0.761
8	0.297	6.192	0.774
16	0.225	8.173	0.511



Σχήμα 7: Επιτάχυνση (speedup) λόγο παράλληλης επεξεργασίας συναρτήσει του αριθμού των threads.