Εργασία στα πλαίσια του μαθήματος "Εργαλεία Προγραμματισμού"

Δημητρίου Ελευθέριος

1 Άσκηση - FCTS method for parabolic PDE

α) Στην άσκηση αυτή θα λύσουμε την παραβολική εξίσωση

$$u_t = k(u_{xx} + u_{yy}) \qquad (x, y) \in [0, 2\pi] \times [0, 2\pi] \qquad t \in [0, 2\pi]$$

με $k = 4/5$ και οριακές συνθήκες

$$u(x, y, t = 0) = sin(x)sin(\frac{y}{2})$$

και συνοριακές συνθήκες

$$u(x = 0, y, t) = 0$$

$$u(x = 2\pi, y, t) = 0$$

$$u(x, y = 0, t) = 0$$

$$u(x, y = 2\pi, t) = 0$$

με την μέθοδο FTCS. Η μέθοδος αυτή, μας δίνει την λύση ως αναδρομική σχέση

$$u_{i.j}^{n+1} = u_{i,j}^n + k\Delta t\Big(\frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta y^2}\Big)$$

με χρονικό βήμα που προκύπτει από την θεωρία μας ως

$$\frac{k\Delta t}{\Delta x^2 + \Delta y^2} \leq \frac{1}{8}$$

για την ευστάθεια της μεθόδου, από τις συνοριακές συνθήκες τύπου Dirichlet.

Για την άσκησή μας θα ισχύει $\Delta x \approx \Delta y$. Για την αριθμητική επίλυση, χωρίσαμε το πλέγμα μας σε NxM = 50x50 πλεγματικά σημεία. Έπειτα πήραμε χωρικά βήματα $\Delta x = \Delta y = \frac{2\pi - 0}{N-1}$ δηλαδή η χωρική μετατόπιση θα είναι $x = x_0 + i * dx$ και $y = y_0 = j * dy$ με $i, j = 0, 1, 2, ..., N$.

Το πρόγραμμά μας υλοποίησε την παραπάνω διαδικασία για την λύση της μ.δ.ε και βάση αυτής σχηματίσαμε τα ακόλουθα διαγράμματα, τα οποία είναι η λύση σε διαφορετικές χρονικές στιγμές και πιο συγκεκριμένα για $t = 0, \pi/2, \pi, 2\pi$.

Στην συνέχεια παρουσιάζουμε τις γραφικές παραστάσεις

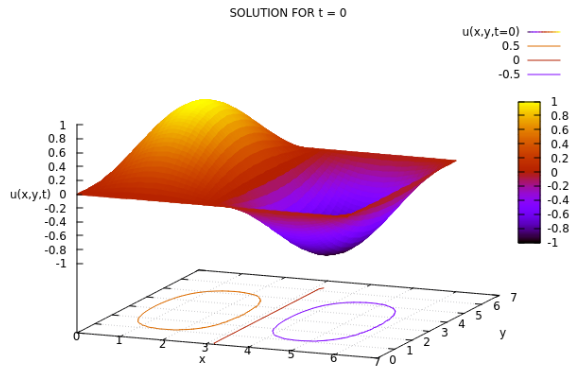

Figure 1: Γράφημα της $u(x, y)$ για δεδομένη χρονική στιγμή, $t = 0$

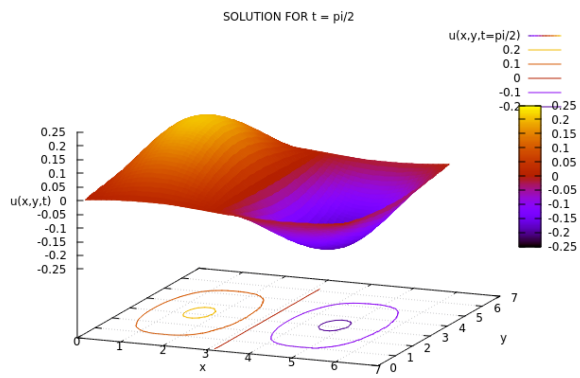

Figure 2: Γράφημα της $u(x, y)$ για δεδομένη χρονική στιγμή, $t = \pi/2$

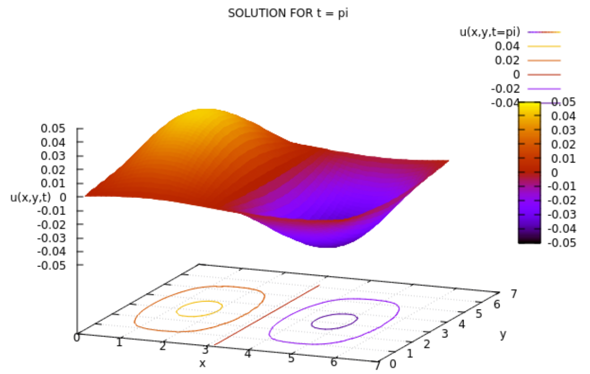

Figure 3: Γράφημα της $u(x, y)$ για δεδομένη χρονική στιγμή, $t = \pi$

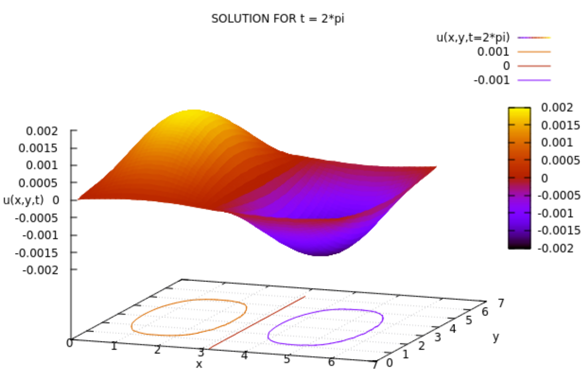

Figure 4: Γράφημα της $u(x, y)$ για δεδομένη χρονική στιγμή, $t = 2\pi$

Έπειτα σχεδιάσαμε την λύση για τέσσερα διαφορετικά ζεύγη σημείων (x, y) ως συνάρτηση του χρόνου.

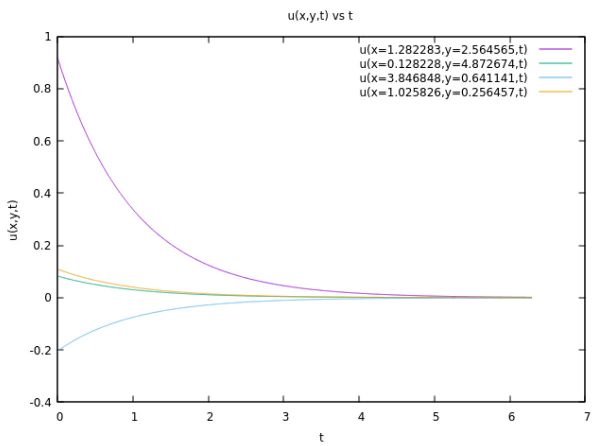

Figure 5: Γράφημα της συνάρτησης $u(x, y, t)$ για x = const, y = const, συναρτήσει του χρόνου.

Όπως βλέπουμε από το διάγραμμα, οι λύσεις για τα σταθερά σημεία συγκλίνουν προς το μηδέν καθώς $t \to 2\pi$ το οποίο και το περιμέναμε λόγω της φύσης της μ.δ.ε.

Για τον ρυθμό σύγκλισης

Στο ερώτημα αυτό μας ζητήθηκε να ελένξουμε αν ο ρυθμός σύγκλισης συμφωνεί με τον αναμενόμενο. Προσπαθώντας να καταλάβω το συγκεκριμένο ερώτημα, η προσοχή μου στράφηκε σε κάποιες ενδιαφέρουσες [1]σημειώσεις σύμφωνα με τις οποίες μπορούμε να κάνουμε έλεγχο για τον ρυθμό σύγκλισης μέσω της σχέσης

$$\frac{u_h - u_{h/2}}{u_{h/2} - u_{h/4}} = 2^P + O(h)$$

όπου $P =$ η τάξη του σφάλματος, ενώ η παραπάνω σχέση προκύπτει για διαφορετικές δοκιμές διαμέρισης του διαστήματος ολοκλήρωσης. Έκανα μερικές δοκιμές για διαφορετικά βήματα και πράγματι πήρα τιμές κοντά στο 2 αφού στην περίπτωσή μας έχουμε σφάλμα τάξης $O(h)$ άρα $P \approx 1$. Θα ήθελα να σημειώσω εδώ πως δεν είμαι απόλυτα σίγουρος για το αν κατάλαβα πλήρως το συγκεκριμένο ερώτημα.

β) Στην συνέχεια της άσκησης, χρησιμοποιήσαμε παράλληλο προγραμματισμό με χρήση της OPENMP, με σκοπό να βελτιστοποιήσουμε τον χρόνο εκτέλεσης του προγράμματος. Στο ακόλουθο διάγραμμα φαίνεται η χρονική διαφορά συναρτήσει του αριθμού των threads

[1]DN2255 - Numerical Solutions of Differential Equations Spring 2012 Olof Runborg

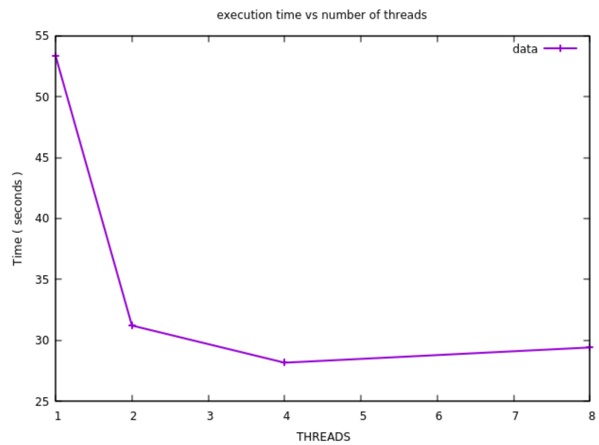

Figure 6: Ο χρόνος εκτέλεσης συναρτήσει του αριθμού των threads.

Όπως ήταν αναμενόμενο, όσο αυξάνουμε το πλήθος των threads τόσο γρηγορότερη είναι η εκτέλεση του προγράμματος.

Σημειώνουμε εδώ πως το γεγονός ότι για το πλήθος των 8 threads υπάρχει μία ελάχιστη αύξηση της χρονικής διάρκειας της εκτέλεσης. Αυτό πιστεύω οφείλεται στο γεγονός ότι το σύστημά μας έχει μόνο 4 threads.

2 Άσκηση - Matrix multiplication

Στην άσκηση αυτή χρησιμοποιήσαμε τον παράλληλο προγραμματισμό με χρήση της OPENMP για τον πολλαπλασιασμό δύο NxN πινάκων. Αρχικά, ελένξαμε, με ένα μικρό πλήθος στοιχείων των πινάκων, ότι το αποτέλεσμα συμφωνεί τόσο με την σειριακή όσο και με την παράλληλη εκτέλεση. Συγκεκριμένα πήραμε

$$\begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 8 & 8 \\ 8 & 8 \end{pmatrix}$$

και στην συνέχεια ορίσαμε το πλήθος των στοιχείων ως $N = 1000$ και μέσω της παράλληλης επεξεργασίας, σχεδιάσαμε την γραφική παράσταση του χρόνου εκτέλεσης, συναρτήσει του αριθμού των threads

Παρατηρούμε και πάλι πως με την αύξηση των threads, μειώνεται ο χρόνος εκτέλεσης του προγράμματος.

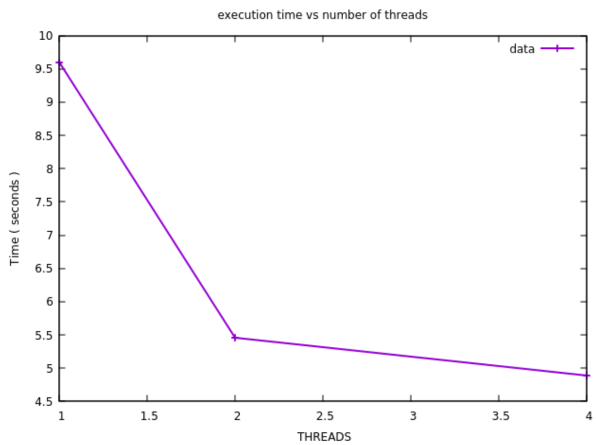

Figure 7: Ο χρόνος εκτέλεσης συναρτήσει του αριθμού των threads.

3 Άσκηση 3 - Numerical Integration from Computational Mathematics I

Στην άσκηση αυτή θα βρούμε την τιμή του ολοκληρώματος

$$\int_0^3 xe^{2x}\,dx$$

με την μέθοδο του Simpson - h/3 για διαμέριση του χώρου με $n = 10^9$ σημεία, χρησιμοποιώντας την παράλληλη επεξεργασία με την χρήση της OPENMP.

Σύμφωνα με την μέθοδο, το ολοκλήρωμα προσεγγίζεται ως

$$I \approx \int_{x_0}^{x_n} f(x)\,dx \approx \frac{h}{3}(f_0 + 4f_1 + 2f_2 + ... + 4f_{n-1} + f_n), \qquad h = \frac{b-a}{n}$$

όπου η μετατόπιση σε κάθε βήμα, του x εκφράζεται μέσω της αναδρομικής σχέσης

$$x_i = x_0 + i * h$$

όπου $i =$ ο αριθμός της κάθε επανάληψης.

Παρατηρούμε πως η πραγματική τιμή του ολοκληρώματος συμφωνεί απόλυτα με την αριθμητική τιμή, κάτι που το περιμέναμε λόγω του μεγάλου πλήθους της διαμέρισης του διαστήματος ολοκλήρωσης.

$$I = 504.535992$$

Παρακάτω, παρουσιάζουμε την γραφική παράσταση της χρονικής διάρκειας εκτέλεσης του προγράμματος, συναρτήσει του αριθμού των threads όπου φαίνεται πως με την αύξηση του αριθμού μειώνεται ο χρόνος εκτέλεσής του.

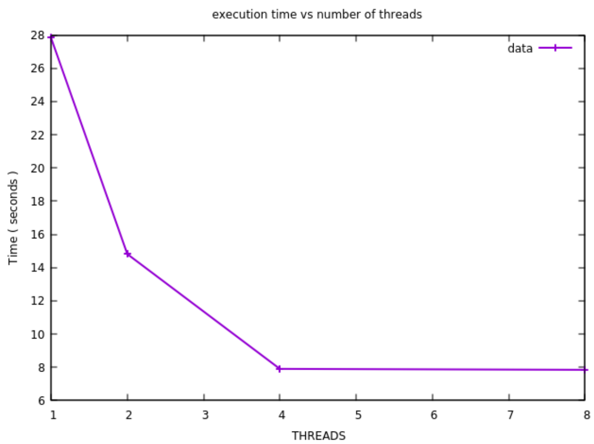

Figure 8: Ο χρόνος εκτέλεσης συναρτήσει του αριθμού των threads.