

**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ**  
**Σχολή Μηχανικών Παραγωγής και Διοίκησης**



**ΕΠΙΣΤΗΜΟΝΙΚΟΙ ΥΠΟΛΟΓΙΣΜΟΙ ΤΕΧΝΙΚΩΝ  
CYCLIC REDUCTION ΓΙΑ HERMITE  
COLLOCATION ΠΕΠΕΡΑΣΜΕΝΑ ΣΤΟΙΧΕΙΑ**

Διπλωματική Εργασία Μ.Δ.Ε  
**ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΣΤΙΣ ΕΠΙΣΤΗΜΕΣ ΜΗΧΑΝΙΚΩΝ**

**ΝΙΚΟΛΑΟΣ Γ. ΕΛΕΥΘΕΡΙΟΥ**

Επιβλέποντες: Καθηγητής **Ιωάννης Σαριδάκης**  
Επ. Καθηγητής **Εμμανουήλ Μαθιουδάκης**

ΧΑΝΙΑ , 2017

Η διατριβή αυτή εξετάστηκε με επιτυχία στις 2 Ιουνίου 2017 από την παρακάτω Τριμελή Επιτροπή :

- Καθηγητή Ιωάννη Σαριδάκη ως επιβλέπων
- Επικ. Καθηγητή Εμμανουήλ Μαθιουδάκη ως συνεπιβλέπων
- Καθηγήτρια Έλενα Παπαδοπούλου

η οποία ορίστηκε κατά τη διάρκεια της 1<sup>ης</sup>/6-9-2016 ΓΣΕΣ της Σχολής Μηχανικών Παραγωγής και Διοίκησης του Πολυτεχνείου Κρήτης.

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω τους Επιβλέποντες της διατριβής μου Καθηγητή Ιωάννη Σαριδάκη και Επικ. Καθηγητή Εμμανουήλ Μαθιουδάκη για την επιστημονική καθοδήγηση που μου παρείχαν σε όλα τα στάδια των μεταπτυχιακών μου σπουδών.

Επίσης, ευχαριστώ το τρίτο μέλος της συμβουλευτικής μου επιτροπής Καθηγήτρια Έλενα Παπαδοπούλου, καθώς και τον Αναπληρωτή Καθηγητή Ανάργυρο Δελή για την πολύτιμη προσφορά τους στην απόκτηση επιστημονικών γνώσεων.

Ιδιαίτερα θα ήθελα να ευχαριστήσω την Ειρήνη για τη συμπαράστασή της στην ολοκλήρωση των σπουδών μου.

Τέλος ευχαριστώ τα μέλη της οικογένειάς μου για την υποστήριξη που μου παρείχαν σε όλη τη διάρκεια των σπουδών μου.



# Περίληψη

Η αριθμητική μέθοδος Πεπερασμένων Στοιχείων Collocation, για την επίλυση Προβλημάτων Συνοριακών Τιμών με τη χρήση ως συναρτήσεις βάσης των κυβικών πολυωνύμων Hermite, αποτελεί ένα προσεγγιστικό σχήμα με υψηλή τάξη ακρίβειας. Αντικείμενο της παρούσας διατριβής είναι η εφαρμογή της μεθόδου για ρεαλιστικές εφαρμογές, στις οποίες απαιτούνται υλοποιήσεις σε υπολογιστικές αρχιτεκτονικές υψηλών επιδόσεων. Αυτό συμβαίνει εξαιτίας της δομής και της διάστασης του παραγόμενου αλγεβρικού συστήματος εξισώσεων, του οποίου η αποδοτική διαδικασία της επίλυσης χρειάζεται να λάβει υπόψη τις παράλληλες ιδιότητες του πίνακα συντελεστών. Σε ένα σημαντικό μέρος των υπολογισμών εμπλέκονται επιλύσεις βασικών γραμμικών συστημάτων, των οποίων οι πίνακες συντελεστών έχουν block τριδιαγώνια μορφή. Στα σύγχρονα υπολογιστικά περιβάλλοντα πολλαπλών πυρήνων, τα οποία διαθέτουν και επιταχυντές υπολογισμών, οι άμεσες κλασικές σειριακές διαδικασίες επίλυσης χρειάζεται να αντικατασταθούν με τεχνικές Cyclic Reduction, ώστε να είναι εφικτή η ταυτόχρονη πραγματοποίηση υπολογισμών από μεγαλύτερο αριθμό υπολογιστικών πυρήνων. Με αυτό τον τρόπο η επαναληπτική διαδικασία επίλυσης των Collocation γραμμικών συστημάτων μπορεί να πραγματοποιηθεί αποδοτικά, βασισμένη σε υλοποιήσεις κατάλληλων αλγορίθμων με υψηλό βαθμό παράλληλων διαδικασιών. Αυτό έχει ως άμεση συνέπεια την επίλυση συστημάτων μεγάλης διάστασης τα οποία αντιστοιχούν σε προβλήματα υψηλών διακριτοποιήσεων, τριών διαστάσεων με τη χρήση των σύγχρονων υπολογιστικών αρχιτεκτονικών πολλαπλών πυρήνων.



# Περιεχόμενα

<b>Ευχαριστίες</b>	<b>iii</b>
<b>Περίληψη</b>	<b>v</b>
<b>1 Αριθμητική Μέθοδος Hermite Collocation σε τρεις διαστάσεις</b>	<b>1</b>
1.1 Εισαγωγή . . . . .	1
1.2 Πεπερασμένα Στοιχεία Hermite Collocation σε τρεις διαστάσεις . . . . .	3
<b>2 Επίλυση του Collocation Γραμμικού Συστήματος</b>	<b>17</b>
2.1 Εισαγωγή . . . . .	17
2.2 Διαμέριση του Collocation Πίνακα . . . . .	18
2.3 Η Επαναληπτική Μέθοδος Επίλυσης . . . . .	25
2.3.1 Μέθοδος Cyclic Reduction . . . . .	26
2.3.2 Μέθοδος Τεχνικής Πολυπλέγματος . . . . .	32
2.3.3 Επαναληπτική Μέθοδος BiCGSTAB . . . . .	40
2.4 Παράλληλος Επιλυτής για Collocation Γραμμικά Συστήματα σε Τρεις Διαστάσεις . . . . .	43
2.5 Παράλληλος αλγόριθμος για υπολογιστικούς πυρήνες CPU αρχιτεκτονικών	52
2.6 Παράλληλος αλγόριθμος για υπολογιστικούς πυρήνες αρχιτεκτονικών με επιταχυντές . . . . .	55
<b>3 Υλοποίηση Παράλληλου Αλγορίθμου</b>	<b>57</b>
3.1 Πρόβλημα δοκιμής 1 . . . . .	60

3.2 Πρόβλημα δοκιμής 2 . . . . .	61
3.3 Πρόβλημα δοκιμής 3 . . . . .	62
3.4 Αποτελέσματα παράλληλης επίλυσης . . . . .	64
<b>4 Συμπεράσματα</b>	<b>67</b>



# Κεφάλαιο 1

## Αριθμητική Μέθοδος Hermite Collocation σε τρεις διαστάσεις

### 1.1 Εισαγωγή

Ένα Πρόβλημα Συνοριακών Τιμών (ΠΣΤ), το οποίο ορίζεται σε ένα χωρίο  $\Omega$  μπορεί να περιγραφεί μέσω των παρακάτω σχέσεων

$$(ΠΣΤ) \quad \begin{cases} \mathcal{L}u(\mathbf{x}) = f(\mathbf{x}) , & \mathbf{x} \in \Omega \\ \mathcal{B}u(\mathbf{x}) = g(\mathbf{x}) , & \mathbf{x} \in \partial\Omega \end{cases} . \quad (1.1)$$

όπου ο τελεστής  $\mathcal{L}$  εφαρμόζεται εντός του χωρίου  $\Omega$ , ενώ ο τελεστής  $\mathcal{B}$  περιγράφει τις συνοριακές συνθήκες του προβλήματος και εφαρμόζεται στο σύνορο του χωρίου  $\partial\Omega$ . Μία από τις πιο διαδεδομένες μεθόδους αριθμητικής επίλυσης ΠΣΤ είναι αυτή των Πεπερασμένων Στοιχείων, η υλοποίηση της οποίας μπορεί να περιγραφεί στα παρακάτω βήματα :

- **Βήμα 1:** Γεωμετρική Διαμέριση του πεδίου  $\Omega$  σε πεπερασμένο πλήθος στοιχείων.

- **Βήμα 2:** Επιλογή  $n$  γραμμικά ανεξάρτητων, κατά τμήματα συνεχών πολυωνυμικών συναρτήσεων  $\Phi_1, \dots, \Phi_n$ , οι οποίες ονομάζονται συναρτήσεις βάσης και χρησιμοποιούνται για την προσέγγιση της πραγματικής λύσης  $u(\mathbf{x})$  του ΠΣΤ ως

$$u(\mathbf{x}) \simeq u_n(\mathbf{x}) = a_1\Phi_1(\mathbf{x}) + \dots + a_n\Phi_n(\mathbf{x}) = \sum_{k=1}^n a_k\Phi_k(\mathbf{x}). \quad (1.2)$$

- **Βήμα 3:** Επιλογή μιας μεθόδου διακριτοποίησης για μετάβαση από το συνεχή χώρο στο διακριτό, δηλαδή μετατροπή του ΠΣΤ σε ένα πρόβλημα επίλυσης ενός γραμμικού συστήματος, το οποίο θα προσδιορίσει τους συντελεστές του παραπάνω γραμμικού συνδιασμού των συναρτήσεων βάσης. Το γραμμικό σύστημα θα έχει τη μορφή

$$C\vec{\mathbf{a}} = \vec{\mathbf{b}} \quad (1.3)$$

όπου  $C \in \mathbb{R}^{n,n}$ ,  $\vec{\mathbf{a}} = [a_1 \ a_2 \ \dots \ a_n]^T$  και  $\vec{\mathbf{b}} = [b_1 \ b_2 \ \dots \ b_n]^T$ .

- **Βήμα 4:** Επιλογή κατάλληλης μεθόδου για την επίλυση του γραμμικού συστήματος και προσδιορισμός των αγνώστων  $a_1, \dots, a_n$ . Η επιλογή αυτή βασίζεται στο μέγεθος και στις ιδιότητες του παραγόμενου γραμμικού συστήματος.

Διαφορετικές επιλογές σε κάθε ένα από τα παραπάνω βήματα συνεπάγονται και διαφορετικές μεθόδους [1] για την επίλυση ΠΣΤ. Η πιο συνηθισμένη μέθοδος πεπερασμένου στοιχείου είναι η μέθοδος Galerkin, η οποία είναι άμεσα συνδεδεμένη με διαδικασίες ολοκλήρωσης. Για την αποφυγή χρήσης μεθόδων αριθμητικής ολοκλήρωσης, μπορεί να εφαρμοστεί η μέθοδος **Collocation**, η οποία αποτελεί αντικείμενο της παρούσας εργασίας και παρουσιάζεται στην επόμενη ενότητα.

## 1.2 Πεπερασμένα Στοιχεία Hermite Collocation σε τρεις διαστάσεις

Στην ενότητα αυτή θα γίνει σύντομη περιγραφή υλοποίησης της αριθμητικής μεθόδου επίλυσης ΠΣΤ Hermite Collocation σε τρεις διαστάσεις. Για το σκοπό αυτό θα χρησιμοποιηθεί το ελλειπτικό πρόβλημα μοντέλο τύπου **Poisson** ορισμένο στον μοναδιαίο κύβο. Το πρόβλημα αυτό σε τρισδιάστατες καρτεσιανές συντεταγμένες έχει τη μορφή :

$$\begin{cases} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) u(x, y, z) = f(x, y, z) , & (x, y, z) \in \Omega \\ u(x, y, z) = 0 , & (x, y, z) \in \partial\Omega \end{cases} \quad (1.4)$$

όπου  $\Omega \equiv [0, 1] \times [0, 1] \times [0, 1]$ .

Για την αριθμητική επίλυση με τη μέθοδο πεπερασμένων στοιχείων Collocation, ακολουθούμε τα παρακάτω βήματα.

- **Βήμα 1:** Θα θεωρήσουμε ομοιόμορφη διαμέριση του χωρίου  $\Omega$  σε  $n_s$  ανα κατεύθυνση υποδιαστήματα  $I_k^x = I_j^y = I_k^z$  ,  $i = 1, \dots, n_s$  με βήμα διακριτοποίησης  $h = \frac{1}{n_s}$ . Οι συντεταγμένες των κόμβων  $(x_i, y_j, z_k)$  του πλέγματος διακριτοποίησης θα είναι :

$$\begin{cases} x_i = (i - 1) h, & i = 1, \dots, n_s + 1 \\ y_j = (j - 1) h, & j = 1, \dots, n_s + 1 \\ z_k = (k - 1) h, & k = 1, \dots, n_s + 1 \end{cases} \quad (1.5)$$

- **Βήμα 2:** Ως συναρτήσεις βάσης επιλέγονται τα **tricubic** πολυώνυμα **Hermite** της μορφής :

$\Phi_k(x, y, z) = \Phi_i(z)\Phi_j(y)\Phi_k(x)$ , οπότε η συνάρτηση  $u(x, y, z)$  προσεγγίζεται από τη σχέση

$$u(x, y, z) \simeq u_n(x, y, z) = \sum_{i=1}^{2n_s+2} \sum_{j=1}^{2n_s+2} \sum_{k=1}^{2n_s+2} a_{i,j,k} \Phi_i(z) \Phi_j(y) \Phi_k(x) \quad (1.6)$$

με  $\Phi_i$  τα πολυώνυμα βάσης Hermite που αντιστοιχούν σε μία διάσταση.

Σε κάθε κόμβο διακριτοποίησης  $x_m$  για κάθε διάσταση, αντιστοιχούν δύο συναρτήσεις βάσης, οι οποίες ορίζονται ως εξής :

$$\Phi_{2m-1}(x) = \begin{cases} \Phi\left(\frac{x-x_m}{h}\right) & , \quad x \in I_{m-1} \cup I_m \\ 0 & , \quad \text{διαφορετικά} \end{cases}, \quad (1.7)$$

$$\Phi_{2m}(x) = \begin{cases} \Psi\left(\frac{x-x_m}{h}\right) & , \quad x \in I_m \cup I_{m-1} \\ 0 & , \quad \text{διαφορετικά} \end{cases}, \quad (1.8)$$

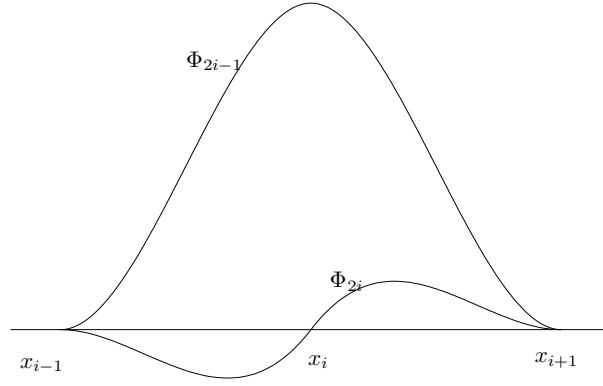
όπου  $m = 1, \dots, (n_s + 1)$ ,  $I_i \equiv [x_i, x_{i+1}]$ ,  $i = 1, \dots, n_s$  1.1

Για να ισχύουν οι παραπάνω ορισμοί για  $m = 1$  και  $m = n_s + 1$ , θεωρούμε δύο εικονικούς κόμβους  $x_0 := -h$  και  $x_{n_s+2} := 1 + h$ , οι οποίοι θα βρίσκονται εντός του χωρίου  $\Omega$ .

Παρατηρούμε ότι ισχύουν οι εξής ιδιότητες για τα μονοδιάστατα πολυώνυμα βάσης

$$\begin{cases} \Phi_{2m-1}(x_i) & = h \frac{d}{dt} \Phi_{2m}(x_i) = \delta_m^i \\ \Phi_{2m}(x_i) & = \frac{d}{dt} \Phi_{2m-1}(x_i) = 0 \end{cases} \quad (1.9)$$

για όλα τα  $m = 1, \dots, (n_s + 1)$  και όπου  $\delta_m^i$  : Δέλτα του Kronecker.



**Σχήμα 1.1:** Πολυώνυμα Hermite [24] σε μια διάσταση ορισμένα στον κόμβο  $x_i$ .

Τα τμηματικά κυβικά πολυώνυμα Hermite [2] όπως αυτά ορίζονται στο διάστημα  $[-1, 1]$  είναι τα εξής :

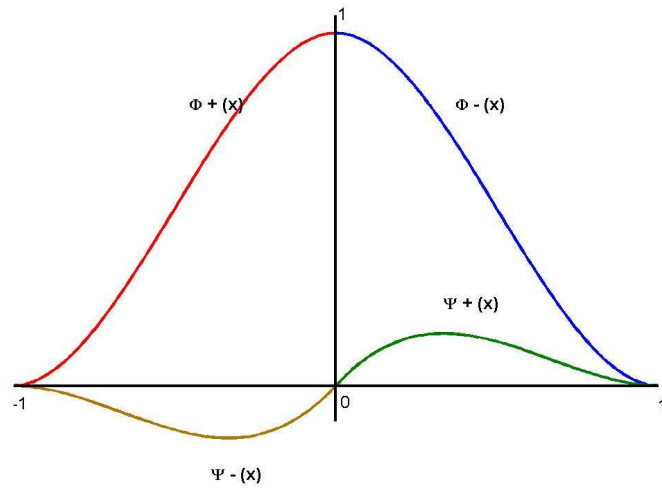
$$\Phi(x) = \begin{cases} \Phi_+(x) & , \quad x \in [0, 1] \\ \Phi_-(x) & , \quad x \in [-1, 0] \\ 0 & , \quad x \notin [-1, 1] \end{cases} \quad , \quad (1.10)$$

$$\Psi(x) = \begin{cases} \Psi_+(x) & , \quad x \in [0, 1] \\ \Psi_-(x) & , \quad x \in [-1, 0] \\ 0 & , \quad x \notin [-1, 1] \end{cases} \quad , \quad (1.11)$$

όπου

$$\Phi_+(x) = \begin{cases} (1-x)^2(1+2x) & , \quad x \in [0, 1] \\ 0 & , \quad x \notin [0, 1] \end{cases} \quad , \quad (1.12)$$

$$\Phi_-(x) = \Phi_+(-x) = \begin{cases} (1+x)^2(1-2x) & , \quad x \in [-1, 0] \\ 0 & , \quad x \notin [-1, 0] \end{cases} \quad , \quad (1.13)$$



**Σχήμα 1.2:** Κυβικά πολυώνυμα Hermite σε μία διάσταση.

$$\Psi_+(x) = \begin{cases} x(1-x)^2 & , \quad x \in [0, 1] \\ 0 & , \quad x \notin [0, 1] \end{cases} , \quad (1.14)$$

$$\Psi_-(x) = -\Psi_+(-x) = \begin{cases} x(1+x)^2 & , \quad x \in [-1, 0] \\ 0 & , \quad x \notin [-1, 0] \end{cases} . \quad (1.15)$$

Το Σχήμα 1.2 εμφανίζει τα κυβικά πολυώνυμα Hermite, όπως αυτά ορίζονται στο  $[-1, 1]$ .

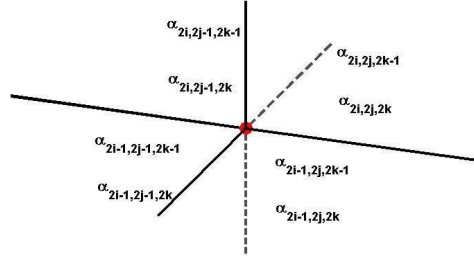
Βασισμένοι στις προηγούμενες ιδιότητες των πολυωνύμων Hermite στη μία διάσταση, προκύπτουν οι παρακάτω ιδιότητες των **tricubic Hermite** πολυωνύμων για κάθε κόμβο  $(x_i, y_j, z_k)$  του τρισδιάστατου πλέγματος διακριτοποίησης :

$$\left\{ \begin{array}{lcl} \Phi_{2i-1,2j-1,2k-1}(x,y,z) & = & \Phi_{2i-1}(z)\Phi_{2j-1}(y)\Phi_{2k-1}(x) \\ \Phi_{2i-1,2j-1,2k}(x,y,z) & = & \Phi_{2i-1}(z)\Phi_{2j-1}(y)\Phi_{2k}(x) \\ \Phi_{2i-1,2j,2k-1}(x,y,z) & = & \Phi_{2i-1}(z)\Phi_{2j}(y)\Phi_{2k-1}(x) \\ \Phi_{2i-1,2j,2k}(x,y,z) & = & \Phi_{2i-1}(z)\Phi_{2j}(y)\Phi_{2k}(x) \\ \Phi_{2i,2j-1,2k-1}(x,y,z) & = & \Phi_{2i}(z)\Phi_{2j-1}(y)\Phi_{2k-1}(x) \\ \Phi_{2i,2j-1,2k}(x,y,z) & = & \Phi_{2i}(z)\Phi_{2j-1}(y)\Phi_{2k}(x) \\ \Phi_{2i,2j,2k-1}(x,y,z) & = & \Phi_{2i}(z)\Phi_{2j}(y)\Phi_{2k-1}(x) \\ \Phi_{2i,2j,2k}(x,y,z) & = & \Phi_{2i}(z)\Phi_{2j}(y)\Phi_{2k}(x) \end{array} \right. \quad (1.16)$$

με τις εξής ιδιότητες :

$$\left\{ \begin{array}{lcl} \Phi_{2i-1,2j-1,2k-1}(x,y,z) & = & 1 \\ h \frac{\partial}{\partial x} \Phi_{2i-1,2j-1,2k}(x,y,z) & = & 1 \\ h \frac{\partial}{\partial y} \Phi_{2i-1,2j,2k-1}(x,y,z) & = & 1 \\ h^2 \frac{\partial^2}{\partial y \partial x} \Phi_{2i-1,2j,2k}(x,y,z) & = & 1 \\ h \frac{\partial}{\partial z} \Phi_{2i,2j-1,2k-1}(x,y,z) & = & 1 \\ h^2 \frac{\partial^2}{\partial z \partial x} \Phi_{2i,2j-1,2k}(x,y,z) & = & 1 \\ h^2 \frac{\partial^2}{\partial z \partial y} \Phi_{2i,2j,2k-1}(x,y,z) & = & 1 \\ h^3 \frac{\partial^3}{\partial z \partial y \partial x} \Phi_{2i,2j,2k}(x,y,z) & = & 1 \end{array} \right. \quad (1.17)$$

Σαν άμεση συνέπεια των προηγούμενων σχέσεων προκύπτει :



**Σχήμα 1.3:** Η διάταξη των 8 αγνώστων στον κόμβο πλέγματος  $(x_i, y_j, z_k)$ .

$$\left\{ \begin{array}{ll} u_{ns}(x_i, y_j, z_k) & = a_{2i-1,2j-1,2k-1} \\ h \frac{\partial}{\partial x} u_{ns}(x_i, y_j, z_k) & = a_{2i-1,2j-1,2k} \\ h \frac{\partial}{\partial y} u_{ns}(x_i, y_j, z_k) & = a_{2i-1,2j,2k-1} \\ h^2 \frac{\partial^2}{\partial y \partial x} u_{ns}(x_i, y_j, z_k) & = a_{2i-1,2j,2k} \\ h \frac{\partial}{\partial z} u_{ns}(x_i, y_j, z_k) & = a_{2i,2j-1,2k-1} \\ h^2 \frac{\partial^2}{\partial z \partial x} u_{ns}(x_i, y_j, z_k) & = a_{2i,2j-1,2k} \\ h^2 \frac{\partial^2}{\partial z \partial y} u_{ns}(x_i, y_j, z_k) & = a_{2i,2j,2k-1} \\ h^3 \frac{\partial^3}{\partial z \partial y \partial x} u_{ns}(x_i, y_j, z_k) & = a_{2i,2j,2k} \end{array} \right. \quad (1.18)$$

όπου  $i, j, k = 1, \dots, (n_s + 1)$ .



- **Βήμα 3:** Επιλέγεται η αριθμητική μέθοδος **Collocation** [3, 11, 12, 13, 6, 14] ως μέθοδος διακριτοποίησης η οποία κατασκευάζει το γραμμικό σύστημα  $\mathbf{Ct} = \mathbf{b}$ . Η κατασκευή αυτή βασίζεται στις απαιτούμενες συνθήκες μηδενικού σφάλματος στα εσωτερικά Collocation σημεία  $(\sigma_i^x, \sigma_j^y, \sigma_k^z)$  του  $\Omega$  και στα  $(\xi_i^x, \xi_j^y, \xi_k^z)$  του συνόρου

$$\begin{cases} \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) u(\sigma_i^x, \sigma_j^y, \sigma_k^z) - f(\sigma_i^x, \sigma_j^y, \sigma_k^z) = 0, & (\sigma_i^x, \sigma_j^y, \sigma_k^z) \in \Omega \\ u(\xi_i^x, \xi_j^y, \xi_k^z) = 0, & (\xi_i^x, \xi_j^y, \xi_k^z) \in \partial\Omega \end{cases} \quad (1.19)$$

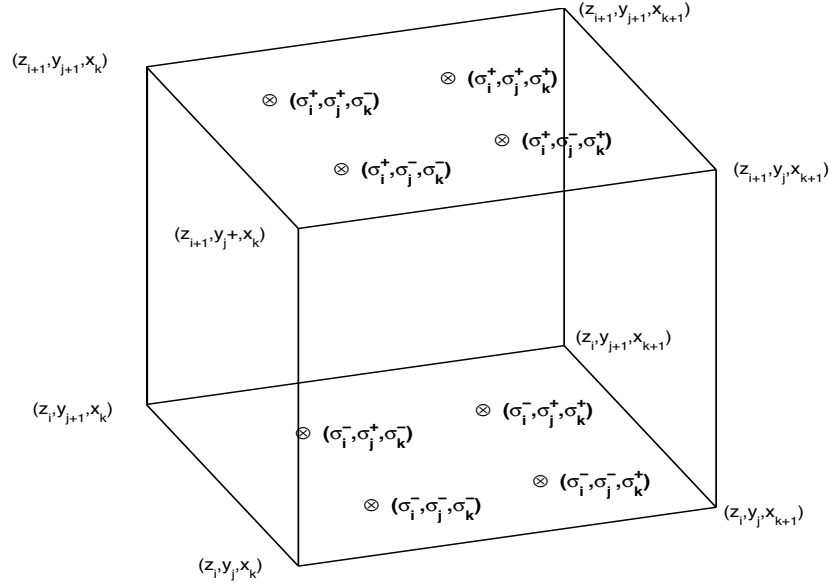
Η εφαρμογή των παραπάνω συνθηκών σε κάθε πεπερασμένο στοιχείο  $I_{ijk}^{xyz}$  στο οποίο αντιστοιχούν 64 μη μηδενικές συναρτήσεις βάσης. Επομένως η λύση του ΠΣΤ στο σημείο  $(x, y, z) \in I_{ijk}^{xyz}$  θα προσεγγίζεται από τη σχέση :

$$u_{ns}(x, y, z) = \sum_{p=2i-1}^{2i+2} \sum_{q=2j-1}^{2j+2} \sum_{r=2k-1}^{2k+2} a_{p,q,r} \Phi_p(z) \Phi_q(y) \Phi_r(x) . \quad (1.20)$$

Έτσι σε κάθε κόμβο πλέγματος  $(x_i, y_j, z_k)$  θα αντιστοιχούν 8 άγνωστοι συντελεστές  $a_{p,q,r}$  η διάταξη των οποίων εμφανίζεται στο Σχήμα 1.3. Γι'αυτό το λόγο κάθε πεπερασμένο στοιχείο  $I_{ijk}^{xyz}$  διαθέτει 64 βαθμούς ελευθερίας.

Στο χωρίο  $\Omega$  υπάρχουν  $8n_s^3$  **εσωτερικά** Collocation σημεία  $(\sigma_i^x, \sigma_j^y, \sigma_k^z)$ , ενώ το πλήθος των συνολικών αγνώστων είναι  $8(n_s + 1)^3$ . Όμως υπάρχουν  $8[3(n_s + 1) + 1]$  άγνωστοι, οι οποίοι αντιστοιχούν στο **σύνορο** και μπορούν να απαλειφθούν από τη χρήση των συνοριακών συνθηκών. Έτσι το πλήθος των αγνώστων που απαιτούν υπολογισμό μετά από αυτή τη διαδικασία απαλοιφής είναι  $8n_s^3$ .

Όπως είναι γνωστό [3, 24, 19] στην περίπτωση ελλειπτικών Προβλημάτων Συνοριακών Συνθηκών, η κλασσική επιλογή των Collocation σημείων είναι αυτή των



**Σχήμα 1.4:** Τα οκτώ Collocation σημεία στο πεπερασμένο στοιχείο  $I_{ijk}^{zyx}$ .

x x x 8	x x 16 24	x x 32 40	x x 48 56	x x x 64
x x x x	x x x x	x x x x	x x x x	x x x x
x 6 x 7	14 15 22 23	30 31 38 39	46 47 54 55	x 62 x 63
x x x x	x x x x	x x x x	x x x x	x x x x
x 4 x 5	12 13 20 21	28 29 36 37	44 45 52 53	x 60 x 61
x x x x	x x x x	x x x x	x x x x	x x x x
x 2 x 3	10 11 18 19	26 27 34 35	42 43 50 51	x 58 x 59
x x x x	x x x x	x x x x	x x x x	x x x x
x x x 1	x x 9 17	x x 25 33	x x 41 49	x x x 57
x x x x	x x x x	x x x x	x x x x	x x x x

**Σχήμα 1.5:** Αρίθμηση αγνώστων γειτονικών του συνόρου της  $Z$  διάστασης για  $n_s = 4$ .

σημείων Gauss. Οι συντεταγμένες τους θα είναι  $(\sigma_i^\pm, \sigma_j^\pm, \sigma_k^\pm)$  με  $i, j, k = 1, \dots, n_s$  όπου  $\sigma_i^\pm = \frac{h}{2}(2i - 1 \pm \frac{\sqrt{3}}{3})$ .

Τα οκτώ εσωτερικά Collocation σημεία, κάθε πεπερασμένου στοιχείου  $I_{ijk}^{zyx}$  εμφανίζονται στο Σχήμα 1.4 και είναι :

$$\left\{ \begin{array}{ll} (\sigma_i^-, \sigma_j^-, \sigma_k^-) & , \quad (\sigma_i^+, \sigma_j^-, \sigma_k^-) \\ (\sigma_i^-, \sigma_j^-, \sigma_k^+) & , \quad (\sigma_i^+, \sigma_j^-, \sigma_k^+) \\ (\sigma_i^-, \sigma_j^+, \sigma_k^-) & , \quad (\sigma_i^+, \sigma_j^+, \sigma_k^-) \\ (\sigma_i^-, \sigma_j^+, \sigma_k^+) & , \quad (\sigma_i^+, \sigma_j^+, \sigma_k^+) \end{array} \right. \quad (1.21)$$

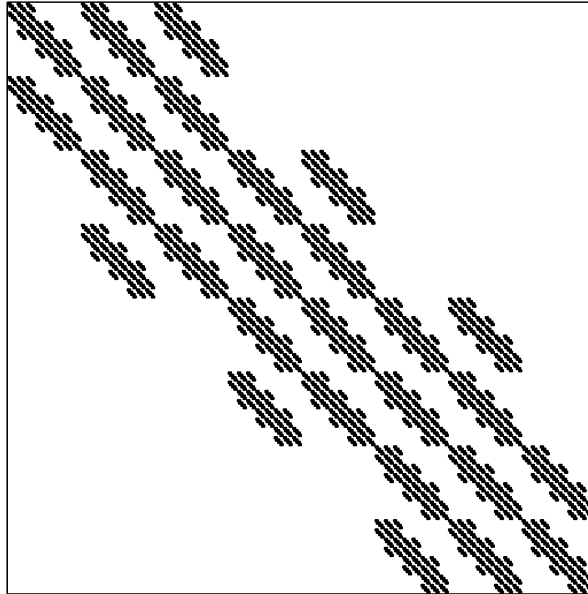
x x x 136	x x 144 152	x x 160 168	x x 176 184	x x x 192
x x x 72	x x 80 88	x x 96 104	x x 112 120	x x x 128
x 134 x 135 142 143	150 151 158 159	166 167 174 175	182 183	x 190 x 191
x 70 x 71	78 79 86 87	94 95 102 103	110 111 118 119	x 126 x 127
x 132 x 133 140 141	148 149 156 157	164 165 172 173	180 181	x 188 x 189
x 68 x 69	76 77 84 85	92 93 100 101	108 109 116 117	x 124 x 125
x 130 x 131 138 139	146 147 154 155	162 163 170 171	178 179	x 186 x 187
x 66 x 67	74 75 82 83	90 91 98 99	106 107 114 115	x 122 x 123
x x x 129	x x 137 145	x x 153 161	x x 169 177	x x x 185
x x x 65	x x 73 81	x x 89 97	x x 105 113	x x x 121

**Σχήμα 1.6:** Αρίθμηση εσωτερικών αγνώστων ως προς την  $Z$  διάσταση για  $n_s = 4$ .

- **Βήμα 4:** Επίλυση του παραγόμενου **Collocation** γραμμικού συστήματος με τη χρήση κάποιας αριθμητικής μεθόδου [26, 28]. Επειδή η διάσταση του γραμμικού συστήματος είναι μεγάλη, απαιτείται η χρήση μιας επαναληπτικής διαδικασίας

8	16	24	32	40	48	56	64
7	15	23	31	39	47	55	63
6	14	22	30	38	46	54	62
5	13	21	29	37	45	53	61
4	12	20	28	36	44	52	60
3	11	19	27	35	43	51	59
2	10	18	26	34	42	50	58
1	9	17	25	33	41	49	57

**Σχήμα 1.7:** Block Τριδιαγώνια αρίθμηση εξισώσεων ως προς τη  $Z$  διάσταση για  $n_s = 4$ .



**Σχήμα 1.8:** Block Τριδιαγώνια δομή του Collocation Πίνακα.

επίλυσης, ανάλογα με τη δομή του πίνακα συντελεστών  $C$ . Η αρίθμηση αγνώστων και εξισώσεων καθορίζει τη δομή του Collocation πίνακα και κατά συνέπεια την επιλογή της μεθόδου επίλυσής του.

Η αρίθμηση που θα επιλέξουμε, παρουσιάζεται ενδεικτικά στο Σχήμα 1.5 για τους αγνώστους που αναφέρονται σε γειονικά του συνόρου ως προς τη  $Z$  διάσταση πεπερασμένα στοιχεία και στο Σχήμα 1.6 για τους αγνώστους των εσωτερικών. Οι άγνωστοι, οι οποίοι έχουν απαλειφθεί λόγω των συνοριακών συνθηκών, σημειώνονται με "x". Ο Collocation πίνακας θα έχει Block τριδιαγώνια μορφή σύμφωνα με αυτή την αρίθμηση, όπως φαίνεται στο Σχήμα 1.8.

Είναι προφανές ότι ο πίνακας συντελεστών έχει αραιή δομή, οπότε η χρήση επαναληπτικής αριθμητικής μεθόδου επίλυσης είναι αναγκαία.

Ειδικότερα ο Collocation Πίνακας  $C \in \mathbb{R}^{8n_s^3 \times 8n_s^3}$  στη block τριδιαγώνια μορφή του θα έχει τη δομή

$$C = \begin{bmatrix} A_2 & A_3 & -A_4 & O & O & \cdots & O & O & O & O & O \\ A_4 & A_1 & -A_2 & O & O & \cdots & O & O & O & O & O \\ O & A_1 & A_2 & A_3 & -A_4 & \cdots & O & O & O & O & O \\ O & A_3 & A_4 & A_1 & -A_2 & \cdots & O & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & \cdots & A_1 & A_2 & A_3 & -A_4 & O \\ O & O & O & O & O & \cdots & A_3 & A_4 & A_1 & -A_2 & O \\ O & O & O & O & O & \cdots & O & O & A_1 & A_2 & -A_4 \\ O & O & O & O & O & \cdots & O & O & A_3 & A_4 & -A_2 \end{bmatrix}, \quad (1.22)$$

όπου τα  $A_i \in \mathbb{R}^{4n_s^2 \times 4n_s^2}$  είναι υποπίνακες της μορφής :

$$A_i = \begin{bmatrix} K_2 & K_3 & -K_4 & O & O & \cdots & O & O & O & O & O \\ K_4 & K_1 & -K_2 & O & O & \cdots & O & O & O & O & O \\ O & K_1 & K_2 & K_3 & -K_4 & \cdots & O & O & O & O & O \\ O & K_3 & K_4 & K_1 & -K_2 & \cdots & O & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & \cdots & K_1 & K_2 & K_3 & -K_4 & O \\ O & O & O & O & O & \cdots & K_3 & K_4 & K_1 & -K_2 & O \\ O & O & O & O & O & \cdots & O & O & K_1 & K_2 & -K_4 \\ O & O & O & O & O & \cdots & O & O & K_3 & K_4 & -K_2 \end{bmatrix}, \quad (1.23)$$

με  $K_i \in \mathbb{R}^{2n_s \times 2n_s}$  είναι οι βασικοί πίνακες μορφής :

$$K_i = \begin{bmatrix} \alpha_2 & \alpha_3 & -\alpha_4 & O & O & \cdots & O & O & O & O & O \\ \alpha_4 & \alpha_1 & -\alpha_2 & O & O & \cdots & O & O & O & O & O \\ O & \alpha_1 & \alpha_2 & \alpha_3 & -\alpha_4 & \cdots & O & O & O & O & O \\ O & \alpha_3 & \alpha_4 & \alpha_1 & -\alpha_2 & \cdots & O & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & \cdots & \alpha_1 & \alpha_2 & \alpha_3 & -\alpha_4 & O \\ O & O & O & O & O & \cdots & \alpha_3 & \alpha_4 & \alpha_1 & -\alpha_2 & O \\ O & O & O & O & O & \cdots & O & O & \alpha_1 & \alpha_2 & -\alpha_4 \\ O & O & O & O & O & \cdots & O & O & \alpha_3 & \alpha_4 & -\alpha_2 \end{bmatrix}, \quad (1.24)$$

ενώ οι τιμές των  $\alpha_i$  θα είναι :

$A_1$	$K_1$	$K_2$	$K_3$	$K_4$
$\alpha_1$	$-\kappa^+$	$-\beta^+$	$\zeta^+$	$-\theta^+$
$\alpha_2$	$-\beta^+$	$-\gamma^+$	$\epsilon^-$	$\nu$
$\alpha_3$	$\zeta^+$	$\epsilon^-$	$\zeta^-$	$-\epsilon^+$
$\alpha_4$	$-\theta^+$	$\nu$	$-\epsilon^+$	$\delta^+$

(1.25)

$A_2$	$K_1$	$K_2$	$K_3$	$K_4$
$\alpha_1$	$-\beta^+$	$-\gamma^+$	$\epsilon^-$	$\nu$
$\alpha_2$	$-\gamma^+$	$-\lambda^+$	$-\delta^-$	$\mu^+$
$\alpha_3$	$\epsilon^-$	$-\delta^-$	$\theta^-$	$-\nu$
$\alpha_4$	$\xi$	$\mu^+$	$-\xi$	$\mu^-$

(1.26)

$A_3$	$K_1$	$K_2$	$K_3$	$K_4$
$\alpha_1$	$\zeta^+$	$\epsilon^-$	$\zeta^-$	$-\epsilon^+$
$\alpha_2$	$\epsilon^-$	$-\delta^-$	$\theta^-$	$-\nu$
$\alpha_3$	$\zeta^-$	$\theta^-$	$\kappa^-$	$-\beta^-$
$\alpha_4$	$-\epsilon^+$	$-\nu$	$-\beta^-$	$\gamma^-$

(1.27)

$A_4$	$K_1$	$K_2$	$K_3$	$K_4$
$\alpha_1$	$-\theta^+$	$\nu$	$-\epsilon^+$	$\delta^+$
$\alpha_2$	$\nu$	$\mu^+$	$-\nu$	$\mu^-$
$\alpha_3$	$-\epsilon^+$	$-\nu$	$-\beta^-$	$\gamma^-$
$\alpha_4$	$\delta^+$	$\mu^-$	$\gamma^-$	$-\lambda^-$

(1.28)

$$\begin{aligned} \mu\epsilon \beta^\pm &= \frac{1}{h} \frac{93\sqrt{3}\pm 157}{108}, \quad \gamma^\pm = \frac{66\sqrt{3}\pm 111}{324}, \quad \delta^\pm = \frac{2\sqrt{3}\pm 1}{108}, \quad \epsilon^\pm = \frac{1}{h} \frac{13\pm 3\sqrt{3}}{108}, \quad \zeta^\pm = \frac{1}{h^2} \frac{24\pm 7\sqrt{3}}{18}, \\ \theta^\pm &= \frac{1}{h} \frac{23\pm 9\sqrt{3}}{108}, \quad \kappa^\pm = \frac{1}{h^2} \frac{43\sqrt{3}\pm 72}{18}, \quad \lambda^\pm = h \frac{3\sqrt{3}\pm 5}{72}, \quad \mu^\pm = h \frac{\sqrt{3}\pm 1}{216}, \quad \nu^\pm = \frac{\sqrt{3}}{108}. \end{aligned}$$

Για την αποδοτική επίλυση του Collocation γραμμικού συστήματος σε σύγχρονες υπολογιστικές αρχιτεκτονικές είναι απαραίτητη η σχεδίαση κατάλληλου παράλληλου αλγορίθμου επίλυσης του. Η δομή του πίνακα συντελεστών επιτρέπει την διεξαγωγή υπολογισμών σε παράλληλα περιβάλλοντα. Όμως είναι δυνατή η αύξηση των παράλληλων ιδιοτήτων του γραμμικού συστήματος, ώστε να είναι δυνατή η υλοποίηση του παράλληλου αλγορίθμου σε αρχιτεκτονικές κοινής μνήμης πολλαπλών υπολογιστικών πηρύνων. Η διαδικασία αυτή μαζί με την κατασκευή και υλοποίηση του παράλληλου αλγορίθμου παρουσιάζονται στο επόμενο κεφάλαιο.





## **Κεφάλαιο 2**

# **Επίλυση του Collocation Γραμμικού Συστήματος**

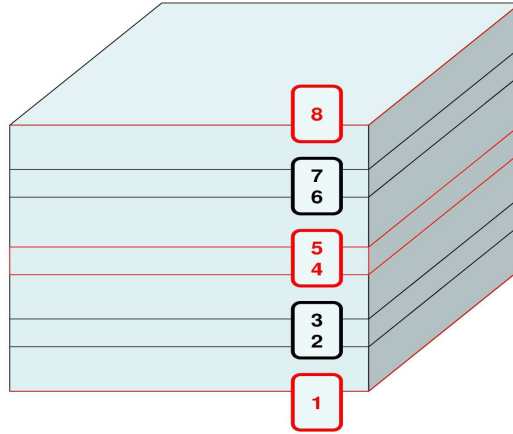
### **2.1 Εισαγωγή**

Στην ενότητα αυτή θα κατασκευαστεί παράλληλος αλγόριθμος της επαναληπτικής διαδικασίας επίλυσης του Collocation γραμμικού συστήματος. Η δομή του πίνακα συντελεστών επιτρέπει τη διεξαγωγή βασικών πράξεων γραμμικής άλγεβρας την ίδια χρονική στιγμή. Όμως είναι δυνατή η περαιτέρω αύξηση των παράλληλων ιδιοτήτων με την ομαδοποίηση και την απεξάρτηση αγώστων μεταξύ τους. Έτσι, αρχικά θα εφαρμοστούν τεχνικές στο γραμμικό σύστημα με σκοπό την αύξηση της ανεξαρτητοποίησης των αγνώστων, ενώ στη συνέχεια θα επιλεγεί μία κατάλληλη αποδοτική επαναληπτική διαδικασία επίλυσης του. Στη συνέχεια θα παρουσιαστεί η κατάλληλη οργάνωση των πράξεων για την υλοποίηση του παράλληλου αλγορίθμου σε αρχιτεκτονικές κοινής μνήμης πολλαπλών πυρήνων.

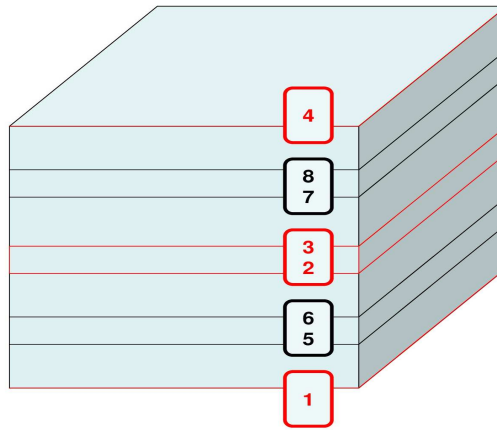
## 2.2 Διαμέριση του Collocation Πίνακα

Όπως παρουσιάστηκε στο προηγούμενο κεφάλαιο, ο πίνακας συντελεστών των αγνώστων του Collocation γραμμικού συστήματος, έχει block τριδιαγώνια μορφή. Η χρήση μίας μεθόδου χρωματισμού και ομαδοποίησης των αγνώστων μπορεί να οδηγήσει στην ανεξαρτιτοποίηση τους με αποτέλεσμα την αλλαγή της δομής του Collocation πίνακα [17, 18, 19, 20]. Έτσι θα είναι δυνατός ο ταυτόχρονος υπολογισμός αυτών των ομάδων των αγνώστων. Η τεχνική χρωματισμού που θα χρησιμοποιήσουμε θα βασιστεί σε δύο χρώματα. Στην πρώτη φάση χρησιμοποιούμε την ιδέα της επαναρίθμησης αγνώστων και εξισώσεων και στη συνέχεια εφαρμόζουμε ένα μετασχηματισμό για τη βελτίωση του βαθμού παραλληλοποίησης του γραμμικού συστήματος. Για την επαναρίθμηση των αγνώστων και των εξισώσεων θα χρησιμοποιηθεί η ιδέα της **Red-Black** διάταξης. Σύμφωνα με αυτή, άγνωστοι/εξισώσεις χωρίζονται σε δύο υποομάδες αγνώστων/εξισώσεων ανάλογα με το χρωματισμό τους με την παρακάτω αρχή: *Τα μέλη των Red υποομάδων είναι γειτονικά μόνο με μέλη των Black υποομάδων.* Η παραπάνω ιδέα επαναρίθμησης είναι ισοδύναμη με την εφαρμογή ενός μετασχηματισμού ομοιότητας στο γραμμικό σύστημα μέσω του μεταθετικού πίνακα  $P \in \mathbb{R}^{8n_s^3 \times 8n_s^3}$  της μορφής :

$$P = \begin{bmatrix} P_1 & O & O & O & O & O & \dots & O & O & O & O \\ O & O & P_3 & O & O & O & \dots & O & O & O & O \\ O & O & O & O & P_5 & O & \dots & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & O & \dots & O & O & O & O \\ O & O & O & O & O & O & \dots & O & P_{n_s-1} & O & O \\ O & O & O & O & O & O & \dots & O & O & O & P_{n_s+1} \\ O & P_2 & O & O & O & O & \dots & O & O & O & O \\ O & O & O & P_4 & O & O & \dots & O & O & O & O \\ O & O & O & O & O & P_6 & \dots & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & O & \dots & P_{n_s-2} & O & O & O \\ O & O & O & O & O & O & \dots & O & O & P_{n_s} & O \end{bmatrix}, \quad (2.1)$$



**Σχήμα 2.1:** Ομαδοποίηση αγνώστων για  $n_s = 4$



**Σχήμα 2.2:** Επαναρίθμηση αγνώστων για  $n_s = 4$

$$\text{όπου } P_j \in \mathbb{R}^{v,v} \text{ με } j = \begin{cases} 4n_s^2 & \text{όταν } j = 1 \text{ και } j = n_s + 1 \\ 8n_s^2 & \text{όταν } j = 2, \dots, n_s \end{cases}.$$

R		B		R		B		R	
x x x 8		x 16	x 24	x 32	x 40	x 48	x 56	x x	x 64
	8	16	24	32	40	48	56		64
	7	15	23	31	39	47	55		63
x x 6 7		14 15	22 23	30 31	38 39	46 47	54 55	x x	62 63
	6	14	22	30	38	46	54		62
	5	13	21	29	37	45	53		61
x x 4 5		12 13	20 21	28 29	36 37	44 45	52 53	x x	60 61
	4	12	20	28	36	44	52		60
	3	11	19	27	35	43	51		59
x x 2 3		10 11	18 19	26 27	34 35	42 43	50 51	x x	58 59
	2	10	18	26	34	42	50		58
	1	9	17	25	33	41	49		57
x x x 1		x 9	x 17	x 25	x 33	x 41	x 49	x x	x 57

**Σχήμα 2.3:** Red - Black ομαδοποίηση αγνώστων και εξισώσεων

x x x 8	x 40	x 48	x 16	x 24	x 56	x 64	x x	x 32
	8	40	48	16	24	56	64	32
	7	39	47	15	23	55	63	31
x x 6 7	38 39	46 47	14 15	22 23	54 55	62 63	x x	30 31
	6	38	46	14	22	54	62	30
	5	37	45	13	21	53	61	29
x x 4 5	36 37	44 45	12 13	20 21	52 53	60 61	x x	28 29
	4	36	44	12	20	52	60	28
	3	35	43	11	19	51	59	27
x x 2 3	34 35	42 43	10 11	18 19	50 51	58 59	x x	26 27
	2	34	42	10	18	50	58	26
	1	33	41	9	17	49	57	25
x x x 1	x 33	x 41	x 9	x 17	x 49	x 57	x x	x 25

**Σχήμα 2.4:** Red - Black αρίθμηση αγνώστων και εξισώσεων

με

$$P_j = \begin{bmatrix} I_1 & O & O & O & O & O & \cdots & O & O & O & O \\ O & O & I_3 & O & O & O & \cdots & O & O & O & O \\ O & O & O & O & I_5 & O & \cdots & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & O & \cdots & O & O & O & O \\ O & O & O & O & O & O & \cdots & O & I_{n_s-1} & O & O \\ O & O & O & O & O & O & \cdots & O & O & O & I_{n_s+1} \\ O & I_2 & O & O & O & O & \cdots & O & O & O & O \\ O & O & O & I_4 & O & O & \cdots & O & O & O & O \\ O & O & O & O & O & I_6 & \cdots & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & O & O & \cdots & I_{n_s-2} & O & O & O \\ O & O & O & O & O & O & \cdots & O & O & I_{n_s} & O \end{bmatrix}, \quad (2.2)$$

για  $I_j \in \mathbb{R}^{v,v}$  μοναδιαίο πίνακα με διάσταση  $j = \begin{cases} 2n_s & \text{όταν } j = 1 \text{ και } j = n_s + 1 \\ 4n_s & \text{όταν } j = 2, \dots, n_s \end{cases}$ .

Ο χρωματισμός και η ομαδοποίηση αγνώστων και εξισώσεων πραγματοποιείται σε δύο στάδια. Στο πρώτο στάδιο γίνεται ομαδοποίηση και χρωματισμός αγνώστων και εξισώσεων ως προς την  $Z$  κατεύθυνση. Δηλαδή ομαδοποιούνται άγνωστοι/εξισώσεις οι οποίοι αντιστοιχούν στην ίδια κάθετη γραμμή του πλέγματος διακριτοποίησης. Το σχήμα 2.1 εμφανίζει αυτή την ομαδοποίηση για  $n_s = 4$ , ενώ το σχήμα 2.2 την επαναρίθμυσή τους. Στο δεύτερο στάδιο αυτής της διαδικασίας εφαρμόζεται η ίδια τεχνική στους αγνώστους/εξισώσεις που αναφέρονται στην ίδια κάθετη γραμμή του πλέγματος διακριτοποίησης. Το Σχήμα 2.3 εμφανίζει την ομαδοποίηση αγνώστων/εξισώσεων στο ίδιο οριζόντιο επίπεδο του πλέγματος διακριτοποίησης, ενώ το Σχήμα 2.4 αναφέρεται στην επαναρίθμυσή τους για  $n_s = 4$ . Όπως αναφέρθηκε η παραπάνω διαδικασία είναι ισοδύναμη με την εφαρμογή του μετασχηματισμού ομοιότητας στο γραμμικό σύστημα μέσω

του πίνακα  $P$ . Ειδικότερα, για τον πίνακα συντελεστών των αγνώστων θα ισχύει :

$$PCP^T = \begin{bmatrix} D_R & -H_B \\ -H_R & D_B \end{bmatrix}, \quad (2.3)$$

όπου

$$D_R = \text{diag}[A_2 \underbrace{\tilde{A} \cdots \tilde{A}}_{p-1} - A_2] \quad (2.4)$$

$$H_R = -\text{diag}[A_4 \underbrace{\hat{A} \cdots \hat{A}}_{p-1} - A_4] \quad (2.5)$$

$$D_B = \text{diag}[\underbrace{\tilde{A} \cdots \tilde{A}}_p] \quad (2.6)$$

$$H_B = -\text{diag}[\underbrace{\hat{A} \cdots \hat{A}}_p] \quad (2.7)$$

$$\tilde{A} = \begin{bmatrix} A_1 & -A_2 \\ A_1 & A_2 \end{bmatrix} \quad \text{και} \quad \hat{A} = \begin{bmatrix} A_3 & -A_4 \\ A_3 & A_4 \end{bmatrix}. \quad (2.8)$$

Η εφαρμογή του μετασχηματισμού σε κάθε πίνακα  $A$  για  $j = 1, 2$  θα οδηγήσει σε :

$$P_j A_j P_j^T = \begin{bmatrix} D_{A_R} & -H_{A_B} \\ -H_{A_R} & D_{A_B} \end{bmatrix}, \quad (2.9)$$

όπου

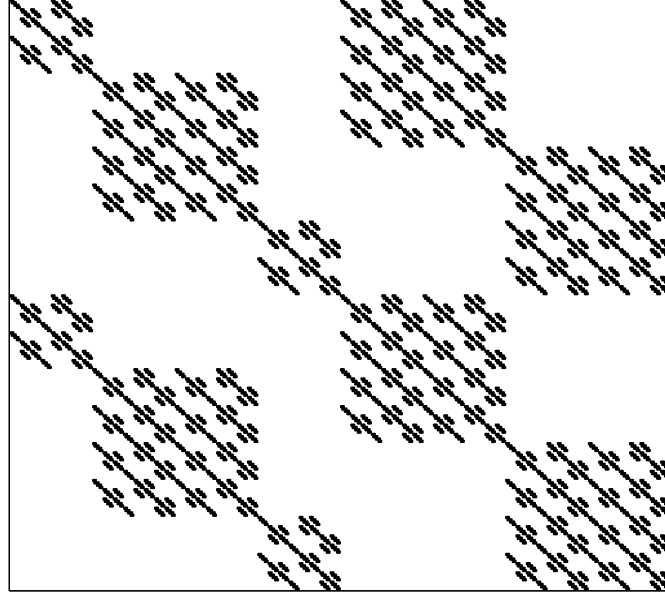
$$D_{A_R} = \text{diag}[K_2 \underbrace{\tilde{K} \cdots \tilde{K}}_{p-1} - K_2] \quad (2.10)$$

$$H_{A_R} = -\text{diag}[K_4 \underbrace{\hat{K} \cdots \hat{K}}_{p-1} - K_4] \quad (2.11)$$

$$D_{A_B} = \text{diag}[\underbrace{\tilde{K} \cdots \tilde{K}}_p] \quad (2.12)$$

$$H_{A_B} = -\text{diag}[\underbrace{\hat{K} \cdots \hat{K}}_p] \quad (2.13)$$

$$\tilde{K} = \begin{bmatrix} K_1 & -K_2 \\ K_1 & K_2 \end{bmatrix} \quad \text{και} \quad \hat{K} = \begin{bmatrix} K_3 & -K_4 \\ K_3 & K_4 \end{bmatrix}. \quad (2.14)$$



**Σχήμα 2.5:** Δομή του Red-Black Collocation Πίνακα.

Η εικόνα 2.5 εμφανίζει τη δομή του Collocation πίνακα στις τρεις διαστάσεις, σύμφωνα με την επαναρίθμηση αγνώστων/εξισώσεων.

Έτσι το Collocation γραμμικό σύστημα θα έχει την ισοδύναμη μορφή

$$C_P = PCP^T(Pt) = P\mathbf{b} \ , \quad (2.15)$$

όπου

$$C_P = PCP^T = \begin{bmatrix} A'_2 & O & O & \dots & O & O & O & A'_3 & -A'_4 & \dots & O & O \\ O & A'_1 & -A'_2 & \dots & O & O & O & A'_3 & A'_4 & \dots & O & O \\ O & A'_1 & A'_2 & \dots & O & O & O & O & O & \dots & O & O \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & \dots & A'_1 & -A'_2 & O & O & O & \dots & O & O \\ O & O & O & \dots & A'_1 & A'_2 & O & O & O & \dots & A'_3 & -A'_4 \\ O & O & O & \dots & O & O & -A'_2 & O & O & \dots & A'_3 & A'_4 \\ A'_4 & O & O & \dots & O & O & O & A'_1 & -A'_2 & \dots & O & O \\ O & A'_3 & -A'_4 & \dots & O & O & O & A'_1 & A'_2 & \dots & O & O \\ O & A'_3 & A'_4 & \dots & O & O & O & O & O & \dots & O & O \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & \dots & A'_3 & -A'_4 & O & O & O & \dots & O & O \\ O & O & O & \dots & A'_3 & A'_4 & O & O & O & \dots & A'_1 & -A'_2 \\ O & O & O & \dots & O & O & -A'_4 & O & O & \dots & A'_1 & A'_2 \end{bmatrix} \quad (2.16)$$

$$A'_i = PA_iP^T = \begin{bmatrix} K_2 & O & O & \dots & O & O & O & K_3 & -K_4 & \dots & O & O \\ O & K_1 & -K_2 & \dots & O & O & O & K_3 & K_4 & \dots & O & O \\ O & K_1 & K_2 & \dots & O & O & O & O & O & \dots & O & O \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & \dots & K_1 & -K_2 & O & O & O & \dots & O & O \\ O & O & O & \dots & K_1 & K_2 & O & O & O & \dots & K_3 & -K_4 \\ O & O & O & \dots & O & O & -K_2 & O & O & \dots & K_3 & K_4 \\ K_4 & O & O & \dots & O & O & O & K_1 & -K_2 & \dots & O & O \\ O & K_3 & -K_4 & \dots & O & O & O & K_1 & K_2 & \dots & O & O \\ O & K_3 & K_4 & \dots & O & O & O & O & O & \dots & O & O \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & \dots & K_3 & -K_4 & O & O & O & \dots & O & O \\ O & O & O & \dots & K_3 & K_4 & O & O & O & \dots & K_1 & -K_2 \\ O & O & O & \dots & O & O & -K_4 & O & O & \dots & K_1 & K_2 \end{bmatrix} \quad (2.17)$$



## 2.3 Η Επαναληπτική Μέθοδος Επίλυσης

Στην ενότητα αυτή θα παρουσιαστεί η επαναληπτική διαδικασία επίλυσης του Collocation γραμμικού συστήματος. Η διαδικασία αυτή θα είναι ένας συνδυασμός τριών επαναληπτικών τεχνικών επίλυσης γραμμικών συστημάτων. Σύμφωνα με τη βιβλιογραφία, για κάθε διάσταση μπορεί να χρησιμοποιηθεί μία κατάλληλη αριθμητική μέθοδος επίλυσης.

Ειδικότερα για την επίλυση των γραμμικών συστημάτων που αναφέρονται σε :

- Μονοδιάστατο πρόβλημα θα γίνει χρήση της άμεσης μεθόδου επίλυσης Cyclic Reduction [28, 26]
- Διδιάστατο πρόβλημα θα γίνει χρήση της επαναληπτικής μεθόδου επίλυσης Τεχνικής Πολυπλέγματος [4, 5, 7, 10, 15, 16, 27]
- Τριδιάστατο πρόβλημα θα γίνει χρήση της μη στατικής επαναληπτικής μεθόδου BiCGSTAB με Gauss-Seidel προρύθμιση [28, 29, 18].

Ακολουθεί μία σύντομη περιγραφή για κάθε μέθοδο επίλυσης, όπως αυτές θα εφαρμοστούν σε κάθε περίπτωση.

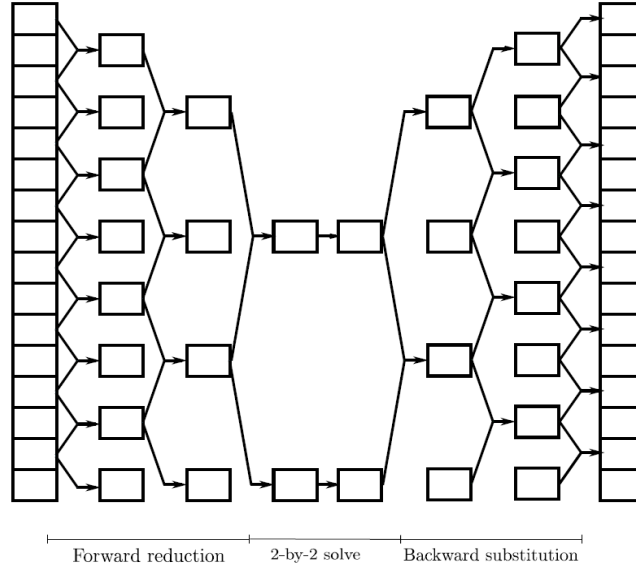
### 2.3.1 Μέθοδος Cyclic Reduction

Η μέθοδος Cyclic Reduction επιλέγεται για την επίλυση του γραμμικού προβλήματος, το οποίο αναφέρεται σε γραμμικό σύστημα της μίας διάστασης. Η μέθοδος αυτή χρησιμοποιείται για την επίλυση τριδιαγώνιων γραμμικών συστημάτων και block τριδιαγώνιων γραμμικών συστημάτων. Συγκεκριμένα, το Collocation γραμμικό σύστημα της μίας διάστασης έχει block τριδιαγώνια μορφή, επομένως η μέθοδος που μπορεί να επιλεγεί είναι η **block** έκδοσή της. Η βασική ιδέα της άμεσης αυτής μεθόδου είναι να εξαλειφθούν οι μισοί από τους αγνώστους, να ανασυνταχθούν οι εξισώσεις και να εξαλειφθούν ξανά οι μισοί άγνωστοι. Η διαδικασία αυτή επαναλαμβάνεται ώσπου το γραμμικό σύστημα να φτάσει σε διάσταση  $2 \times 2$ . Ο αλγόριθμος αυτός μπορεί εύκολα να χρησιμοποιηθεί σε μια μεγάλη ποικιλία παράλληλων αρχιτεκτονικών.

Ο αλγόριθμος αποτελείται από δύο φάσεις:

**Forward Reduction:** στην οποία κάθε άρτια εξίσωση  $i$  ενός γραμμικού συστήματος διάστασης  $n$  εκφράζεται ως γραμμικός συνδυασμός των  $i, i - 1, i + 1$  εξισώσεων, με αποτέλεσμα να παράγεται ένα καινούριο τριδιαγώνιο σύστημα με  $\frac{n}{2}$  εξισώσεις και  $\frac{n}{2}$  αγνώστους. Η διαδικασία αυτή μπορεί να εκτελεστεί ταυτόχρονα για όλες τις άρτιες εξισώσεις και επαναλαμβάνεται ώσπου να έχει παραμείνει ένα γραμμικό σύστημα 2 εξισώσεων με 2 αγνώστους. Το  $2 \times 2$  σύστημα λύνεται τετριμμένα δίνοντας τις τιμές των  $x_n$  και  $x_{\frac{n}{2}}$  αγνώστων.

**Backward Substitution:** στην οποία κάθε περιττός άγνωστος  $x_i$  λύνεται από την αντί-κατάσταση των γνωστών τιμών των  $x_{i-1}$  και  $x_{i+1}$ , τα οποία είναι γνωστά από το σύστημα που έχει λυθεί στο παραπάνω βήμα. Ο προσδιορισμός των περιττών αγνώστων μπορεί να πραγματοποιηθεί ταυτόχρονα για όλους τους αγνώστους. Κάθε φάση του αλγορίθμου αποτελείται από  $\log_2(n) - 1$  βήματα.



**Σχήμα 2.6:** Σχηματική αναπαράσταση αλγορίθμου Cyclic Reduction.

### Αλγόριθμος της μεθόδου Cyclic Reduction

Υποθέτουμε ότι το σύστημα  $Qx = u$  διάστασης  $n$ , έχει πίνακα συντελεστών  $Q$  της μορφής :

$$Q = \begin{bmatrix} a_1 & c_1 & & & \\ b_2 & a_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{n-1} & a_{n-1} & c_{n-1} \\ & & & b_n & a_n \end{bmatrix} \quad (2.18)$$

Η πρώτη βασική διαδικασία του αλγορίθμου είναι η εξάλειψη των περιττών αγνώστων.

Η διαδικασία αυτή περιγράφεται ως Block LU παραγοντοποίηση με την εφαρμογή μεταθέσεων των γραμμών και των στηλών του πίνακα  $Q$ . Έστω  $S$  ο μεταθετικός πίνακας :

$$S[1, 2, \dots, n]^T = [1, 3, \dots, |2, 4, \dots]^T \quad (2.19)$$

Ο πίνακας συντελεστών γίνεται :

$$SQS^T = \left[ \begin{array}{ccc|ccc} a_1 & & & c_1 & & \\ & a_3 & & b_3 & c_3 & \\ & & \ddots & & \ddots & \ddots \\ \hline b_2 & c_2 & & a_2 & & \\ & & b_4 & & a_4 & \\ & & & & & \ddots \end{array} \right] \quad (2.20)$$

Η εξάλειψη των περιττών αγνώστων είναι ισοδύναμη με τη μερική παραγοντοποίηση του πίνακα στη μορφή :

$$SQS^T = \left[ \begin{array}{ccc|ccc} 1 & & & O & & \\ & \ddots & & & \ddots & \\ & & 1 & & & O \\ \hline l_1 & m_1 & & 1 & & \\ & & l_2 & & \ddots & \\ & & & & \ddots & 1 \end{array} \right] \left[ \begin{array}{ccc|ccc} a_1 & & & c_1 & & \\ & a_3 & & b_3 & c_3 & \\ & & \ddots & & \ddots & \ddots \\ \hline O & & & a_1^{(1)} & c_1^{(1)} & \\ & \ddots & & b_2^{(1)} & a_2^{(1)} & \ddots \\ & & O & & \ddots & \ddots \end{array} \right] \quad (2.21)$$

Για  $n$  άρτιο , αυτή η διάσπαση υπολογίζεται για  $i = 1, \dots, \frac{n}{2} - 1$  βήματα :

$$\begin{aligned} m_i &= \frac{c_{2i}}{a_{2i+1}}, & l_i &= \frac{b_{2i}}{a_{2i-1}}, & l_{\frac{n}{2}} &= \frac{b_n}{a_{n-1}} \\ c_i^{(1)} &= -m_i c_{2i+1}, & b_{i+1}^{(1)} &= -l_{i+1} b_{2i+1} & . & \\ a_i^{(1)} &= a_{2i} - l_i c_{2i-1} - m_i b_{2i+1}, & a_{\frac{n}{2}}^{(1)} &= a_n - l_n c_{n-1} & & \end{aligned} \quad (2.22)$$

Η διαδικασία αυτή κατασκευάζει το Shur Complement του πίνακα σε τριδιαγώνια μορφή και μπορούμε να επαναλάβουμε την εξάλειψη των περιττών αγνώστων του νέου μειωμένου συστήματος για άλλη μία φορά.

### Αλγόριθμος της μεθόδου Block Cyclic Reduction

Η μέθοδος που θα χρησιμοποιηθεί για τον αλγόριθμο επίλυσης του Collocation γραμμικού συστήματος είναι η **Block Cyclic Reduction**. Αυτή είναι η ίδια διαδικασία με τη διαφορά ότι τα στοιχεία του πίνακα, είναι υποπίνακες. Επομένως θα υπάρχουν πράξεις γραμμικής άλγεβρας μεταξύ πινάκων στη θέση των αριθμητικών.

Υποθέτουμε το block τριδιαγώνιο πίνακα  $T(B, A, C)$  τάξης  $n = 2^q - 1$ . Ο πίνακας  $T$  είναι πίνακας συντελεστών του γραμμικού συστήματος :

$$\begin{bmatrix} A & C & & O \\ B & A & C & \\ & \ddots & \ddots & \ddots \\ & & B & A & C \\ O & & & B & A \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}, \quad (2.23)$$

όπου  $A, B, C \in \mathbb{R}^{m \times m}$ . Αρχικά εφαρμόζεται η διαδικασία μετάθεσης στις γραμμές και τις στήλες των περιπτών θέσεων με αποτέλεσμα :

$$\left[ \begin{array}{ccc|ccc} A & & & O & C & & O \\ & \ddots & & & B & \ddots & \\ & & \ddots & & & \ddots & C \\ O & & & A & O & & B \\ \hline B & C & & O & A & & O \\ & \ddots & \ddots & & & \ddots & \\ O & & B & C & O & & A \end{array} \right] \begin{bmatrix} u_1 \\ u_3 \\ \vdots \\ \frac{u_{2(2^{q-1})-1}}{u_2} \\ u_4 \\ \vdots \\ u_{2(2^{q-1})-2} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_3 \\ \vdots \\ \frac{b_{2(2^{q-1})-1}}{b_2} \\ b_4 \\ \vdots \\ b_{2(2^{q-1})-2} \end{bmatrix}, \quad (2.24)$$

το οποίο μπορεί να γραφεί στη μορφή :

$$\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{bmatrix} u_{odd} \\ u_{even} \end{bmatrix} = \begin{bmatrix} b_{odd} \\ b_{even} \end{bmatrix}. \quad (2.25)$$

Υποθέτουμε ότι ο παραπάνω πίνακας συντελεστών είναι ομαλός και έπειτα εφαρμόζουμε τη μέθοδο κατασκευής Schur complement για τον  $H_{11}$  υποπίνακα και καταλήγουμε σε ένα μικρότερο σύστημα διάστασης  $2^{q-1} - 1$  :

$$(H_{22} - H_{21}H_{11}^{-1}H_{12})u_{even} = b^{(1)}, b^{(1)} = b_{even} - H_{21}H_{11}^{-1}b_{odd}. \quad (2.26)$$

Με την παραπάνω διαδικασία, το σύστημα που προκύπτει έχει την ίδια δομή με τον αρχικό πίνακα  $T$ , δηλαδή θα είναι της μορφής

$$\begin{bmatrix} A^{(1)} & C^{(1)} & & & O \\ B^{(1)} & A^{(1)} & C^{(1)} & & \\ & \ddots & \ddots & & \\ & & B^{(1)} & A^{(1)} & C^{(1)} \\ O & & & B^{(1)} & A^{(1)} \end{bmatrix} \begin{bmatrix} u_2 \\ u_4 \\ u_6 \\ \vdots \\ u_{2^q-2} \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \\ b_3^{(1)} \\ \vdots \\ b_{2^{q-1}-1}^{(1)} \end{bmatrix}, \quad (2.27)$$

όπου :

$$b_i^{(1)} = b_{2i} - BA^1 b_{2i-1} - CA^{-1} b_{2i+1}, \quad i = 1, \dots, 2^{q-1} - 1. \quad (2.28)$$

και

$$\begin{aligned} A^{(1)} &= A - BA^{-1}C - CA^{-1}B \\ B^{(1)} &= -BA^{-1}B \\ C^{(1)} &= -CA^{-1}C \end{aligned} \quad (2.29)$$

Αυτή η διαδικασία μπορεί να επαναλαμβάνεται κυκλικά και να παράγει μία ακολουθία block υποσυστημάτων μεγέθους  $2^{q-k} - 1$  :

$$\begin{bmatrix} A^{(k)} & C^{(k)} & & & O \\ B^{(k)} & A^{(k)} & C^{(k)} & & \\ & \ddots & \ddots & & \\ & & B^{(k)} & A^{(k)} & C^{(k)} \\ O & & & B^{(k)} & A^{(k)} \end{bmatrix} \begin{bmatrix} u_{1 \cdot 2^k} \\ u_{2 \cdot 2^k} \\ \vdots \\ u_{2^{q-k}-1 \cdot 2^k} \end{bmatrix} = \begin{bmatrix} b_1^{(k)} \\ b_2^{(k)} \\ b_3^{(k)} \\ \vdots \\ b_{2^{q-k}-1}^{(k)} \end{bmatrix}, \quad (2.30)$$

για  $k = 0, 1, \dots, q-1$ . Οι πίνακες  $A^k, B^k, C^k$ , και τα διανύσματα  $b_i^k$  υπολογίζονται σύμφωνα με τις παρακάτω σχέσεις :

$$b_i^{(k+1)} = b_{2i}^{(k)} - B^{(k)}(A^{(k)})^{-1}b_{2i-1}^{(k)} - C^{(k)}(A^{(k)})^{-1}b_{2i+1}^{(k)}, \quad i = 1, \dots, 2^{q-k} - 1 \quad (2.31)$$

και

$$\begin{aligned} A^{(k+1)} &= A^{(k)} - B^{(k)}(A^{(k)})^{-1}C^{(k)} - C^{(k)}(A^{(k)})^{-1}B^{(k)} \\ B^{(k+1)} &= -B^{(k)}(A^{(k)})^{-1}B^{(k)} \\ C^{(k+1)} &= -C^{(k)}(A^{(k)})^{-1}C^{(k)} \end{aligned} \quad (2.32)$$

για  $k = 0, 1, \dots, q-2$ , και  $A^{(0)} = A, B^{(0)} = B, C^{(0)} = C$ , υπό την προϋπόθεση ότι η  $\det A^{(k)} \neq 0$  για κάθε  $k$ .

Στο τέλος αυτής της διαδικασίας έχει παραχθεί το  $m \times m$  σύστημα  $A^{(q-1)}u_{2^{q-1}} = b_1^{(q-1)}$ , από το οποίο υπολογίζονται οι τιμές του  $u_{2^{q-1}}$ .

Η δεύτερη φάση (Backward Substitution:) εκτελείται λύνοντας τα υποσυστήματα

$$A^{(k)}u_{(2i-1)2^k} = b_{2i-1}^{(k)} - B^{(k)}u_{(2i-2)2^k} - C^{(k)}u_{(2i)2^k}, \quad (2.33)$$

για  $i = 1, 2, \dots, 2^{q-k-1}$ ,  $k = q-2, \dots, 0$  αρχίζοντας με το  $u_{2^{q-1}}$ . Με αυτή την επαναληπτική διαδικασία προσδιορίζονται όλοι οι περιττοί άγνωστοι του γραμμικού συστήματος και είναι ουσιαστικά μία μορφή προσδιορισμού αγνώστων σύμφωνα με τη διαδικασία της κλασσικής πίσω-αντικατάστασης.

### 2.3.2 Μέθοδος Τεχνικής Πολυπλέγματος

Η Τεχνική Πολυπλέγματος αποτελεί ένα σύνολο επαναληπτικών αλγορίθμων στην αριθμητική επίλυση ενός γραμμικού συστήματος χρησιμοποιώντας μία ακολουθία από κλιμακούμενα μεγέθη τους. Η κύρια ιδέα της τεχνικής πολυπλέγματος είναι η χρήση μεθόδων εξομάλυνσης, όπως π.χ. η Gauss-Seidel, ώστε να επταχύνεται η μείωση των όρων των σφαλμάτων στις κατευθύνσεις των υψίσυχων ιδιοδιανυσμάτων του επαναληπτικού πίνακα. Ορισμένοι όροι σφαλμάτων εμφανίζονται λιγότερο ομαλοί για ένα σύστημα που αναφέρεται σε ένα πιο αραιό πλέγμα διακριτοποίησης, με αποτέλεσμα να γίνεται αναγκαία η χρήση ενός πιο αραιού πλέγματος, ώστε να μειωθούν οι αντίστοιχοι όροι σφάλματος. Η διαδικασία αυτή επαναλαμβάνεται χρησιμοποιώντας μια σειρά κατάλληλων μεταβάσεων σε διαφορετικού μεγέθους πλεγμάτων, ώστε να κατασκευαστούν τα συστήματα που αναφέρονται σε αυτά.

Θα θεωρήσουμε ένα πρόβλημα συνωριακών τιμών. Επίσης θεωρούμε ομοιόμορφο διαμερισμό του διαστήματος στο οποίο ορίζεται το πρόβλημα σε  $n$  διαστήματα με βήμα διακριτοποίησης  $h = \frac{1}{n}$ . Η διακριτοποίηση αυτή οδηγεί στην κατασκευή του συστήματος

$$A\mathbf{u} = \mathbf{f} \quad , \quad (2.34)$$

όπου  $\mathbf{u}$  : η μοναδική ακριβής λύση του γραμμικού συστήματος.

Επίσης θα χρησιμοποιηθεί ο συμβολισμός  $\mathbf{v}$ , ο οποίος αναφέρεται σε κάποια προσέγγιση της λύσης του γραμμικού συστήματος η οποία προέκυψε από τη χρήση μίας επαναληπτικής μεθόδου. Υπάρχουν δύο σημαντικά μέτρα για τη σύγκριση της προσεγγιστικής λύσης  $\mathbf{v}$  με την πραγματική λύση  $\mathbf{u}$ , ώστε να έχουμε τη δυνατότητα εκτίμησης της προσέγγισης της λύσης από την επαναληπτική μέθοδο. Το πρώτο είναι το αλγεβρικό σφάλμα  $\mathbf{e}$  και δίνεται από τη σχέση

$$\mathbf{e} = \mathbf{u} - \mathbf{v} \quad . \quad (2.35)$$



Ένα ακόμη υπολογιστικό μέτρο που δείχνει πόσο καλά η  $\mathbf{v}$  προσέγγισε την  $\mathbf{u}$  αποτελεί το υπόλοιπο (residual), που ορίζεται ως

$$\mathbf{r} = \mathbf{f} - A\mathbf{v} \quad . \quad (2.36)$$

Το υπόλοιπο εκφράζει κατά πόσο η προσεγγιστική λύση δεν κατάφερε να ικανοποιήσει το πραγματικό πρόβλημα  $A\mathbf{u} = \mathbf{f}$  και χρησιμοποιείται συχνότερα στην πράξη σε σχέση με το σφάλμα. Αυτό συμβαίνει διότι το σφάλμα προϋποθέτει τη γνώση της πραγματικής λύσης  $\mathbf{u}$ . Λόγω της μοναδικότητας της λύσης έχουμε  $\mathbf{r} = \mathbf{0}$  αν και μόνο αν  $\mathbf{e} = \mathbf{0}$ . Ωστόσο, δεν ισχύει πάντοτε ότι αν  $\mathbf{r}$  είναι μικρό σε νόρμα, το  $\mathbf{e}$  είναι επίσης μικρό σε νόρμα. Με την βοήθεια των ορισμών των  $\mathbf{r}$  και  $\mathbf{e}$  προκύπτει μια σχέση μεταξύ τους

$$A\mathbf{u} = \mathbf{f} \Leftrightarrow A(\mathbf{v} + \mathbf{e}) = \mathbf{f} \Leftrightarrow A\mathbf{e} = \mathbf{f} - A\mathbf{v} \Leftrightarrow \underline{A\mathbf{e} = \mathbf{r}} \quad . \quad (2.37)$$

Η σχέση αυτή είναι η εξίσωση υπολοίπου ή το γραμμικό σύστημα υπολοίπου. Παρατηρούμε ότι το σφάλμα ικανοποιεί το ίδιο σύνολο εξισώσεων, όπως η άγνωστη  $\mathbf{u}$ , όταν την θέση της  $\mathbf{f}$  πάρει το υπόλοιπο  $\mathbf{r}$ . Υποθέτουμε ότι η προσέγγιση  $\mathbf{v}$  υπολογίστηκε με κάποια μέθοδο. Έτσι μπορούμε να υπολογίσουμε το υπόλοιπο  $\mathbf{r} = \mathbf{f} - A\mathbf{v}$ . Για τη βελτίωση της προσέγγισης  $\mathbf{v}$ , μπορούμε να λύσουμε την εξίσωση υπολοίπου ως προς  $\mathbf{e}$  και η νέα προσέγγιση που προκύπτει από τον ορισμό του σφάλματος είναι

$$\mathbf{u} = \mathbf{r} + \mathbf{e} \quad . \quad (2.38)$$

Μία από τις επαναληπτικές μεθόδους επίλυσης του γραμμικού συστήματος  $A\mathbf{u} = \mathbf{f}$  είναι η μέθοδος Gauss-Seidel, η οποία βασίζεται στην παρακάτω διάσπαση του πίνακα συντελεστών

$$A = D - L - U \quad (2.39)$$

όπου  $D$  είναι το διαγώνιο τμήμα του  $A$ , και  $-L$ ,  $-U$  αποτελούν αντίστοιχα τα αυστηρά κάτω και πάνω τριγωνικά μέρη του. Το σύστημα  $A\mathbf{u} = \mathbf{f}$  θα έχει τη μορφή

$$(D - L - U)\mathbf{u} = \mathbf{f} \quad . \quad (2.40)$$

Απομονώνοντας το κάτω τριγωνικό μέρος του  $A$  έχουμε

$$(D - L)\mathbf{u} = U\mathbf{u} + \mathbf{f} \quad (2.41)$$

ή

$$\mathbf{u} = (D - L)^{-1}U\mathbf{u} + (D - L)^{-1}\mathbf{f} \quad . \quad (2.42)$$

Ο επαναληπτικός Gauss-Seidel πίνακας ορίζεται από τη σχέση

$$R_{GS} = (D - L)^{-1}U \quad (2.43)$$

και η μέθοδος Gauss-Seidel μπορεί να γραφεί στη μορφή

$$\mathbf{v}^{(m+1)} = R_{GS}\mathbf{v}^{(m)} + (D - L)^{-1}\mathbf{f} \quad . \quad (2.44)$$

Η μέθοδος αυτή ανήκει στην κατηγορία των Στατικών Επαναληπτικών Μεθόδων, δηλαδή ο επαναληπτικός πίνακας παραμένει σταθερός για κάθε επαναληπτικό βήμα. Η κατηγορία αυτή σχεδιάστηκε έτσι ώστε η ακριβής λύση  $\mathbf{u}$ , να αποτελεί σταθερό σημείο (fixed point) του επαναληπτικού τύπου. Αυτό σημαίνει ότι οι επαναλήψεις δεν αλλάζουν την ακριβή λύση

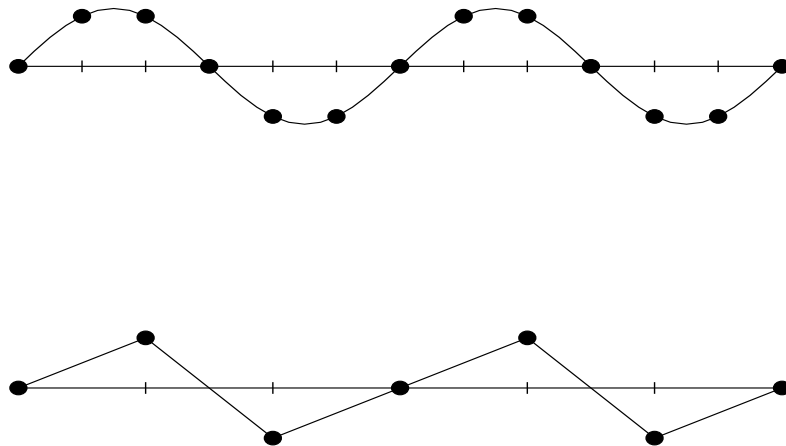
$$\mathbf{u} = R_{GS}\mathbf{u} + (D - L)^{-1}\mathbf{f} \quad . \quad (2.45)$$

Αφαιρώντας τις δύο τελευταίες σχέσεις, λαμβάνουμε

$$\mathbf{e}^{(m+1)} = R_{GS}\mathbf{e}^{(m)} \quad (2.46)$$

ή

$$(D - L)\mathbf{e}^{(m+1)} - U\mathbf{e}^{(m)} = 0 \quad . \quad (2.47)$$



**Σχήμα 2.7:** Το αραιό πλέγμα (κάτω) αναγνωρίζει μια συνάρτηση ως λιγότερο ομαλή από ότι το πιο πυκνό πλέγμα(πάνω).

Η δεύτερη στρατηγική ενσωματώνει την ιδέα να χρησιμοποιήσουμε το σχήμα εξομάλυνσης στην εξίσωση υπολοίπου. Η διαδικασία αυτή περιγράφεται με τον αλγόριθμο

---

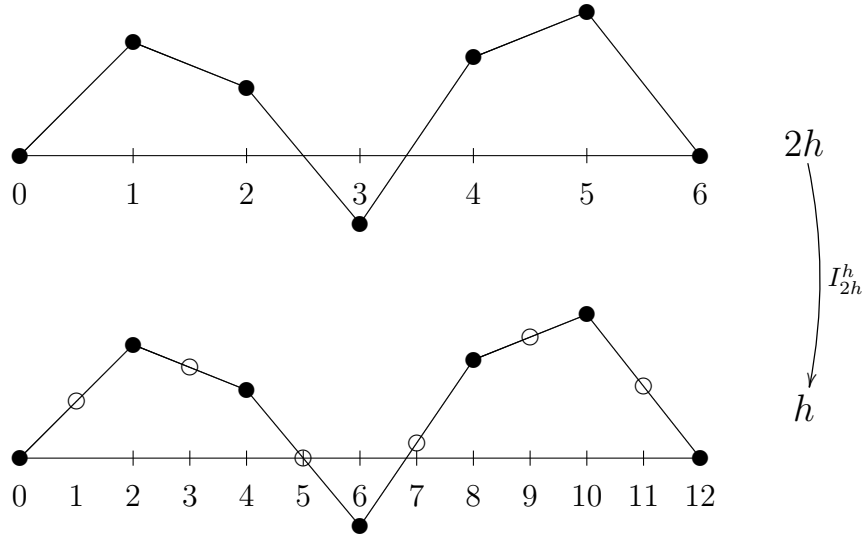
### Αλγόριθμος Τεχνικής Πολυπλέγματος δύο πλεγμάτων

---

- Εξομαλύνεται η  $Au = f$  για  $\Omega^h$  και λαμβάνουμε την προσέγγιση  $v^h$ .
- Υπολογίζουμε το υπόλοιπο  $r = f - Av$ .

Εξομαλύνεται η εξίσωση υπολοίπου  $Ae = r$  για  $\Omega^{2h}$  και λαμβάνουμε μια προσέγγιση για το  $e^{2h}$ .

- Διορθώνουμε την προσέγγιση που είχαμε λάβει στο  $\Omega^h$  με την εκτίμηση του σφάλματος στο  $\Omega^{2h} : v^h \leftarrow v^h + e^{2h}$ .
-



**Σχήμα 2.8:** Γραμμική παρεμβολή των ποσοτήτων στους κόμβους του πυκνού μονοδιάστατου πλέγματος συναρτήσει των τιμών στο αραιό.

### Παρεμβολή

Η παρεμβολή είναι η διαδικασία στην οποία ο τελεστής της οποίας μετασχηματίζει διανύσματα τιμών που αναφέρονται από ένα αραιό πλέγμα σε ένα πυκνό.

Χρησιμοποιείται ο συμβολισμός

$$I_{2h}^h : \Omega^{2h} \longrightarrow \Omega^h . \quad (2.48)$$

Ο τελεστής επέκτασης μπορεί να είναι παρεμβολή ανώτερης τάξης ή να λαμβάνει υπόψη την κατεύθυνση διάδοσης της πληροφορίας κατά την επίλυση παραβολικών ή υπερβολικών προβλημάτων. Οι πρακτικές αυτές, μολοντί επιταχύνουν τη σύγκλιση σε σχέση με μία γραμμική ή μηδενικού βαθμού παρεμβολή, έχουν μεγαλύτερο υπολογιστικό κόστος κατά την εφαρμογή τους που αντισταθμίζει το προηγούμενο πλεονέκτημα. Γι'αυτό το λόγο, κατά κανόνα χρησιμοποιείται η γραμμική παρεμβολή.

Στις δύο διαστάσεις ο τελεστής επέκτασης μπορεί να οριστεί με όμοιο τρόπο. Αν θεωρήσουμε ότι  $I_{2h}^h \mathbf{v}^{2h} = \mathbf{v}^h$ , τότε οι συνιστώσες του  $\mathbf{v}^h$  δίνονται από τις σχέσεις

$$\left\{ \begin{array}{lcl} v_{2i,2j}^h & = & v_{ij}^{2h} \\ v_{2i+1,2j}^h & = & \frac{1}{2}(v_{ij}^{2h} + v_{i+1,j}^{2h}) \\ v_{2i,2j+1}^h & = & \frac{1}{2}(v_{ij}^{2h} + v_{i,j+1}^{2h}) \\ v_{2i+1,2j+1}^h & = & \frac{1}{4}(v_{ij}^{2h} + v_{i+1,j}^{2h} + v_{i,j+1}^{2h} + v_{i+1,j+1}^{2h}), \quad 0 \leq i, j \leq \frac{n}{2} - 1 \end{array} \right. . \quad (2.49)$$

### Παρεκβολή

Η παρεκβολή είναι η αντίστροφη διαδικασία της παρεμβολής. Ο τελεστής παρεκβολής (περιορισμού) μετασχηματίζει διανύσματα τιμών από ένα πυκνό πλέγμα σε ένα αραιό. Χρησιμοποιείται ο συμβολισμός

$$I_h^{2h} : \Omega^h \longrightarrow \Omega^{2h} . \quad (2.50)$$

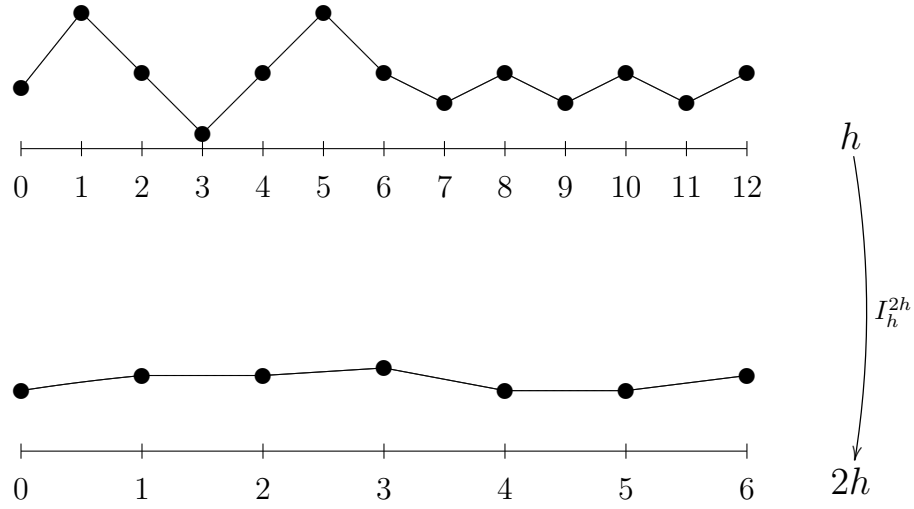
Η πιο απλή μορφή παρεκβολής είναι η προβολή (*injection*). Ορίζεται από την σχέση

$$v_j^{2h} = v_{2j}^h . \quad (2.51)$$

Οι συνιστώσες του διανύσματος  $(v_j^{2h})$  στο αραιό πλέγμα παίρνουν τις τιμές τους απευθείας από τις αντίστοιχες των συνιστώσεων του διανύσματος  $(v_{2j}^h)$  στο πυκνό πλέγμα. Ορίζεται ο τελεστής προβολής εντελώς ανάλογα και για τις δύο διαστάσεις από την σχέση

$$v_{i,j}^{2h} = v_{2i,2j}^h . \quad (2.52)$$

Παρατηρούμε ότι ο τελεστής αυτός για τον υπολογισμό της  $v^{2h}$  σε ένα σημείο εκμεταλλεύεται μόνο την πληροφορία που μας δίνει η τιμή της  $v^h$  στο σημείο αυτό. Σε κάποιες περιπτώσεις αυτό δεν είναι αποδοτικό. Στο Σχήμα 2.9 παρουσιάζεται ένας εναλλακτικός τελεστής περιορισμού μεγαλύτερης ακρίβειας που εκμεταλλεύεται και την πληροφορία από τα *γειονικά* σημεία, ο περιοριστής πλήρους στάθμισης (*full weighting*), ο οποίος ορίζεται



**Σχήμα 2.9:** Ο τελεστής περιορισμού πλήρους στάθμισης για τη μεταφορά των ποσοτήτων από το πυκνό στο αραιό πλέγμα.

$$\begin{aligned}
 v_{ij}^{2h} = \frac{1}{16} [ & v_{2i-1,2j-1}^h + v_{2i-1,2j+1}^h + v_{2i+1,2j-1}^h + v_{2i+1,2j+1}^h \\
 & + 2(v_{2i,2j-1}^h + v_{2i,2j+1}^h + v_{2i-1,2j}^h + v_{2i+1,2j}^h) \\
 & + 4v_{2i,2j}^h ], \quad 1 \leq j \leq \frac{n}{2} - 1 \quad .
 \end{aligned} \tag{2.53}$$

Η στρατηγική συνολικής μετάβασης μεταξύ προβλημάτων διαφορετικών πλεγμάτων περιγράφεται από ένα αριθμητικό σχήμα με την ονομασία κύκλος τεχνικής πολυπλέγματος. Υπάρχουν αρκετά τέτοια σχήματα με το πιο συνηθισμένο να αναφέρεται σαν V-Κύκλος. Σε αυτό η διαδικασία ξεκινά από το πιο πυκνό πλέγμα και με διαδοχικές μεταβάσεις σε αραιότερα πλέγματα επιλύεται το πρόβλημα που αντιστοιχεί στο πιο αραιό της επιλογής. Στη συνέχεια με διαδοχικές μεταβάσεις σε πυκνότερα πλέγματα μέχρι την επιστροφή στο

αρχικό πυκνό πλέγμα. Εκεί επιτυγχάνεται η διόρθωση της προσεγγιστικής λύσης και αν αυτή αποδεκτή η διαδικασία ολοκληρώνεται, διαφορετικά επαναλαμβάνεται με τον ίδιο τρόπο. Στη συνέχεια παρουσιάζεται ο αλγόριθμος τεχνικής πολυπλέγματος για την περίπτωση της στρατηγικής μετάβασης των πλεγμάτων σύμφωνα με τον αλγόριθμο του V-Κύκλου για  $L$  εναλλαγές πλεγμάτων.

---

### Αλγόριθμος V-Κύκλου $L$ επιπέδων

---

- Εξομαλύνεται η  $A\mathbf{u} = \mathbf{f}$   $v_1$  φορές οπότε προκύπτει μια εκτίμηση  $\mathbf{v}^h$ .
  - Υπολογίζεται η ποσότητα  $\mathbf{f}^{2h} = I_h^{2h}\mathbf{r}^h$ .
    - Εξομαλύνεται  $A^{2h}\mathbf{u}^{2h} = \mathbf{f}^{2h}$   $v_1$  φορές αρχική εκτίμηση  $\mathbf{v}^{2h}$ .
    - Υπολογίζεται η ποσότητα  $\mathbf{f}^{4h} = I_{2h}^{4h}\mathbf{r}^{2h}$ .
      - Εξομαλύνεται  $A^{4h}\mathbf{u}^{4h} = \mathbf{f}^{4h}$   $v_1$  φορές αρχική εκτίμηση  $\mathbf{v}^{4h}$ .
      - Υπολογίζεται η ποσότητα  $\mathbf{f}^{8h} = I_{4h}^{8h}\mathbf{r}^{4h}$ .
      - .
      - .
      - .
      - Επιλύεται η εξίσωση  $A^{Lh}\mathbf{u}^{Lh} = \mathbf{f}^{Lh}$  στο αραιότερο πλέγμα.
      - .
      - .
      - .
      - Διορθώνεται η προσέγγιση  $\mathbf{v}^{4h}$  σύμφωνα με τη σχέση:  $\mathbf{v}^{4h} \leftarrow \mathbf{v}^{4h} + I_{8h}^{4h}\mathbf{v}^{8h}$ .
      - Εξομαλύνεται η  $A^{4h}\mathbf{u}^{4h} = \mathbf{f}^{4h}$   $v_2$  φορές με αρχική εκτίμηση  $\mathbf{v}^{4h}$ .
      - Διορθώνεται η προσέγγιση  $\mathbf{v}^{2h}$  σύμφωνα με τη σχέση:  $\mathbf{v}^{2h} \leftarrow \mathbf{v}^{2h} + I_{4h}^{2h}\mathbf{v}^{4h}$ .
      - Εξομαλύνεται η  $A^{2h}\mathbf{u}^{2h} = \mathbf{f}^{2h}$   $v_2$  φορές με αρχική εκτίμηση  $\mathbf{v}^{2h}$ .
      - Διορθώνεται η προσέγγιση  $\mathbf{v}^h$  σύμφωνα με τη σχέση:  $\mathbf{v}^h \leftarrow \mathbf{v}^h + I_{2h}^h\mathbf{v}^{2h}$ .
      - Εξομαλύνεται η  $A^h\mathbf{u}^h = \mathbf{f}^h$   $v_2$  φορές με αρχική εκτίμηση  $\mathbf{v}^h$ .
-

### 2.3.3 Επαναληπτική Μέθοδος BiCGSTAB

Για την επίλυση του γραμμικού προβλήματος, το οποίο αναφέρεται στο τρισδιάστατο πρόβλημα, θα επιλεγεί μια Krylov επαναληπτική μέθοδος. Πιο συγκεκριμένα στην εργασία [18] επιλύονται τα Collocation γραμμικά συστήματα δύο διαστάσεων κάνοντας χρήση της BiCGSTAB μεθόδου με Gauss-Seidel προρύθμιση. Η μέθοδος αυτή εμφανίζεται να είναι αποδοτική και σε παράλληλα υπολογιστικά περιβάλλοντα. Όπως προκύπτει από τον αλγόριθμο της μεθόδου που εμφανίζεται παρακάτω, οι πράξεις της μεθόδου με το μεγαλύτερο υπολογιστικό κόστος είναι αυτές της επίλυσης του γραμμικού συστήματος προρύθμισης και του πολλαπλασιασμού διανύσματος με τον πίνακα συντελεστών των αγνώστων. Η προρύθμιση θα είναι τύπου Gauss-Seidel, βασισμένη στην παρακάτω block διάσπαση του πίνακα συντελεστών

$$C = D - L - U \quad (2.54)$$

όπου

$$D = \begin{bmatrix} D_R & O \\ O & D_B \end{bmatrix} \quad L = \begin{bmatrix} O & O \\ -H_R & O \end{bmatrix} , \quad (2.55)$$

και

$$U = \begin{bmatrix} O & -H_B \\ O & O \end{bmatrix} . \quad (2.56)$$



---

## Αλγόριθμος της προορυθμισμένης BiCGSTAB μεθόδου

---

Επιλογή αρχικής προσέγγισης  $\mathbf{x}^{(0)}$  της λύσης  $\mathbf{x}$

$$\mathbf{r}^{(0)} = \mathbf{b} - C\mathbf{x}^{(0)}$$

Επιλογή αρχικής τιμής  $\hat{r}$  (συνήθως  $\hat{r} = r^{(0)}$ )

**for**  $i = 1, 2, \dots$

$$\rho_{i-1} = \hat{r}^T \mathbf{r}^{(i-1)}$$

**if**  $\rho_{i-1} = 0$  η μέθοδος αποτυγχάνει

**if**  $i = 1$

$$\mathbf{p}^{(1)} = \mathbf{r}^{(0)}$$

**else**

$$\beta_{i-1} = \frac{\rho_{i-1}}{\rho_{i-2}} \frac{\alpha_{i-1}}{\omega_{i-1}}$$

$$\mathbf{p}^{(i)} = \mathbf{r}^{(i-1)} + \beta_{i-1} (\mathbf{p}^{(i-1)} - \omega_{i-1} \mathbf{v}^{(i-1)})$$

**endif**

Επίλυση του  $M \hat{\mathbf{p}} = \mathbf{p}^{(i)}$

$$\mathbf{v}^{(i)} = C \hat{\mathbf{p}}$$

$$\alpha_i = \frac{\rho_{i-1}}{\hat{r}^T \mathbf{v}^{(i)}}$$

$$\mathbf{s} = \mathbf{r}^{(i-1)} - \alpha_i \mathbf{v}^{(i)}$$

**if**  $\|\mathbf{s}\|$  αρκετά μικρό **then**

$$\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} + \alpha_i \hat{\mathbf{p}} \quad \text{έξοδος}$$

Επίλυση του  $M \mathbf{z} = \mathbf{s} \quad ; \quad \mathbf{t} = C \mathbf{z}$

$$\omega_i = \frac{\mathbf{s}^T \mathbf{t}}{\mathbf{t}^T \mathbf{t}}$$

$$\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} + \alpha_i \hat{\mathbf{p}} + \omega_i \mathbf{z}$$

Έλεγχος σύγκλισης

**if**  $\omega_i = 0$  έξοδος

$$\mathbf{r}^{(i)} = \mathbf{s} - \omega_i \mathbf{t}$$

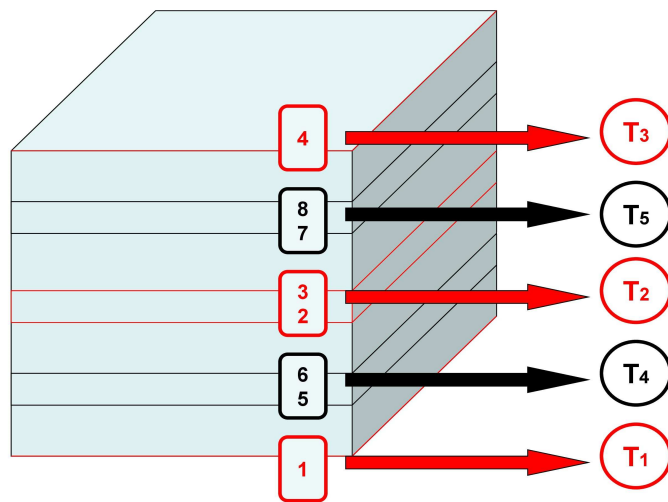
**end**

---

Είναι φανερό ότι η επίλυση αυτή έχει αυξημένες παράλληλες ιδιότητες, αφού σε αυτές εμπλέκονται μεγάλα τμήματα του Collocation πίνακα  $C$ . Αυτός κατασκευάστηκε με τέτοιο τρόπο ώστε να διαθέτει αυξημένο βαθμό παράλληλων χαρακτηριστικών και η δεύτερη βασική πράξη της μεθόδου, αυτή του πολλαπλασιασμού πίνακα με διάνυσμα, διαθέτει υψηλό βαθμό παράλληλων ιδιοτήτων. Το σύνολο των πράξεων της μεθόδου συμπληρώνεται με βασικές πράξεις γραμμικής άλγεβρας μεταξύ διανυσμάτων, όπως εσωτερικό γινόμενο, γραμμικός συνδυασμός και υπολογισμός νόρμας. Οι πράξεις αυτές είναι πλήρως παραλληλοποιήσιμες. Για το λόγο αυτό η χρήση της BiCGSTAB μεθόδου αναμένεται να επιλύσει αποδοτικά το Collocation γραμμικό σύστημα σε παράλληλες αρχιτεκτονικές.

## 2.4 Παράλληλος Επίλυτής για Collocation Γραμμικά Συστήματα σε Τρεις Διαστάσεις

Στην ενότητα αυτή θα κατασκευαστεί ο παράλληλος αλγόριθμος για την αποδοτική επίλυση του Collocation γραμμικού συστήματος σύμφωνα με την αριθμητική μέθοδο που παρουσιάστηκε στις προηγούμενες ενότητες. Η επαναληπτική μέθοδος BiCGSTAB θα χρησιμοποιηθεί ως τη βασική διαδικασία για την επίλυση του γραμμικού συστήματος. Στην επίλυση του συστήματος σε αρχιτεκτονικές πολλαπλών υπολογιστικών πυρήνων θα υπολογιστούν ταυτόχρονα ομάδες αγνώστων ανεξάρτητα μεταξύ τους. Για το λόγο αυτό θα χρειαστεί να γίνει η αντιστοίχιση των ανεξάρτητων ομάδων των αγνώστων σε υπολογιστικά νήματα (Threads) με σκοπό τον προσδιορισμό τους. Έτσι θα υπάρξουν αντίστοιχα Red και Black υπολογιστικά νήματα σύμφωνα με το χρωματισμό που έχει γίνει στις ομάδες των αγνώστων ως προς την τρίτη διάσταση του προβλήματος. Η εικόνα 2.10 εμφανίζει το είδος της αντιστοίχισης αυτής για  $n_s = 4$ .



**Σχήμα 2.10:** Αντιστοίχιση των ομάδων αγνώστων με τα υπολογιστικά νήματα.

Σύμφωνα με αυτή την αντιστοίχιση, σε κάθε υπολογιστικό νήμα θα προσδιοριστούν οι άγνωστοι που αντιστοιχούν σε κάθε γραμμή πλέγματος της τρίτης διάστασης της διακριτοποίησης του χωρίου. Η αντιστοίχιση αυτή αναφέρεται στο πρώτο επίπεδο παραλληλοποίησης, το οποίο εμφανίζει το γραμμικό μας σύστημα και αφορά διαδικασίες της επαναληπτικής μεθόδου BiCGSTAB. Για τον προσδιορισμό των αγνώστων είναι απαραίτητη η ίδιου τύπου αντιστοίχιση σε κάθε υπολογιστικό νήμα των δεξιών μελών κάθε εξίσωσης. Στη συνέχεια εμφανίζεται ο παράλληλος αλγόριθμος της μεθόδου BiCGSTAB για το Collocation γραμμικό σύστημα.

---

### Παράλληλος Αλγόριθμος της προρυθμισμένης BiCGSTAB μεθόδου

---

Επιλογή αρχικής προσέγγισης  $x^{(0)}$  της λύσης  $x$

Παράλληλος υπολογισμός  $r^{(0)} = b - C_P x^{(0)}$

Επιλογή αρχικής τιμής  $\hat{r}$  (συνήθως  $\hat{r} = r^{(0)}$ )

**for**  $i = 1, 2, \dots$

Παράλληλος υπολογισμός  $\rho_{i-1} = \hat{r}^T r^{(i-1)}$

**if**  $\rho_{i-1} = 0$  η μέθοδος αποτυγχάνει

**if**  $i = 1$

Παράλληλος υπολογισμός  $p^{(1)} = r^{(0)}$

**else**

Παράλληλος υπολογισμός  $\beta_{i-1} = \frac{\rho_{i-1}}{\rho_{i-2}} \frac{\alpha_{i-1}}{\omega_{i-1}}$

Παράλληλος υπολογισμός  $p^{(i)} = r^{(i-1)} + \beta_{i-1} (p^{(i-1)} - \omega_{i-1} v^{(i-1)})$

**endif**

Παράλληλη Επίλυση του  $M \hat{p} = p^{(i)}$

Παράλληλος υπολογισμός  $v^{(i)} = C \hat{p}$

Παράλληλος υπολογισμός  $\alpha_i = \frac{\rho_{i-1}}{\hat{r}^T v^{(i)}}$

Παράλληλος υπολογισμός  $s = r^{(i-1)} - \alpha_i v^{(i)}$

**if**  $\|s\|$  αρκετά μικρό **then**

Παράλληλος υπολογισμός  $x^{(i)} = x^{(i-1)} + \alpha_i \hat{p}$  έξοδος

Παράλληλη Επίλυση του  $Mz = s$

Παράλληλος υπολογισμός του  $t = C_P z$

Παράλληλος υπολογισμός  $\omega_i = \frac{s^T t}{t^T t}$

Παράλληλος υπολογισμός  $x^{(i)} = x^{(i-1)} + \alpha_i \hat{p} + \omega_i z$

Έλεγχος σύγκλισης

**if**  $\omega_i = 0$  έξοδος

Παράλληλος υπολογισμός  $r^{(i)} = s - \omega_i t$

**end**

---

Ο παράλληλος αλγόριθμος της μεθόδου περιλαμβάνει βασικές διαδικασίες γραμμικής άλγεβρας, όπως εσωτερικά γινόμενα, πρόσθεση και νόρμες διανυσμάτων, διαδικασίες που είναι πλήρως παραλληλοποιήσιμες και μπορούν να εκτελεστούν ταυτόχρονα. Αυτές βασίζονται στην ίδια διαμέριση όλων των βοηθητικών διανυσμάτων της μεθόδου σύμφωνα με τη διαμέριση των διανυσμάτων αγνώστου και δεξιού μέλους που περιγράφηκε παραπάνω. Ο παράλληλος αλγόριθμος της BiCGSTAB περιλαμβάνει επίσης τις διαδικασίες πολλαπλασιασμού διανύσματος με τον πίνακα Collocation  $C_P$  καθώς και της επίλυσης γραμμικού συστήματος με τον πίνακα προρύθμισης  $M$ . Οι διαδικασίες αυτές χαρακτηρίζονται από υψηλό βαθμό παραλληλοποίησης και μπορούν να αναφερθούν ως ένα δεύτερο επίπεδο παραλληλοποίησης. Ειδικότερα, η παράλληλη διαδικασία του πολλαπλασιασμού διανύσματος με τον Collocation πίνακα  $C_P$  περιγράφεται από τον παρακάτω αλγόριθμο .

---

Παράλληλος αλγόριθμος υπολογισμού  $t = C \cdot z$

---

- Βήμα 1 : Παράλληλος υπολογισμός  $t_{1:4n_s^3} = D_R \cdot z_{1:4n_s^3}$
  - Βήμα 2 : Παράλληλος υπολογισμός  $t_{1:4n_s^3} = t_{1:4n_s^3} - H_B \cdot z_{4n_s^3+1:8n_s^3}$
  - Βήμα 3 : Παράλληλος υπολογισμός  $t_{4n_s^3+1:8n_s^3} = D_B \cdot z_{4n_s^3+1:8n_s^3}$
  - Βήμα 4 : Παράλληλος υπολογισμός  $t_{4n_s^3+1:8n_s^3} = t_{4n_s^3+1:8n_s^3} - H_R \cdot z_{1:4n_s^3}$
- 

Ο παραπάνω αλγόριθμος περιλαμβάνει τέσσερις παράλληλες διαδικασίες, οι οποίες μπορούν να περιγραφούν από τους παρακάτω τέσσερις αλγορίθμους. Κάθε αλγόριθμος αναφέρεται στην πράξη του πολλαπλασιασμού με τον κατάλληλο υποπίνακα της διαμέρισης του Collocation πίνακα  $C_P$ .

---

Παράλληλος αλγόριθμος υπολογισμού  $t = D_R \cdot z$

---

Παράλληλος υπολογισμός  $t_1 = A'_2 \cdot z_1$

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Παράλληλος υπολογισμός  $t_i = A'_1 \cdot z_i - A'_2 \cdot z_{i+1}$

Παράλληλος υπολογισμός  $t_{i+1} = A'_1 \cdot z_i + A'_2 \cdot z_{i+1}$

**end**

Παράλληλος υπολογισμός  $t_{n_s} = -A'_2 \cdot z_{n_s}$

---

---

Παράλληλος αλγόριθμος υπολογισμού  $t = H_R \cdot z$

---

Παράλληλος υπολογισμός  $t_1 = A'_4 \cdot z_1$

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Παράλληλος υπολογισμός  $t_i = A'_3 \cdot z_i - A'_4 \cdot z_{i+1}$

Παράλληλος υπολογισμός  $t_{i+1} = A'_3 \cdot z_i + A'_4 \cdot z_{i+1}$

**end**

Παράλληλος υπολογισμός  $t_{n_s} = -A'_4 \cdot z_{n_s}$

---

---

Παράλληλος αλγόριθμος υπολογισμού  $t = D_B \cdot z$

---

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Παράλληλος υπολογισμός  $t_i = A'_1 \cdot z_i - A'_2 \cdot z_{i+1}$

Παράλληλος υπολογισμός  $t_{i+1} = A'_1 \cdot z_i + A'_2 \cdot z_{i+1}$

**end**

---

---

Παράλληλος αλγόριθμος υπολογισμού  $t = H_B \cdot z$

---

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Παράλληλος υπολογισμός  $t_i = A'_3 \cdot z_i - A'_4 \cdot z_{i+1}$

Παράλληλος υπολογισμός  $t_{i+1} = A'_3 \cdot z_i + A'_4 \cdot z_{i+1}$

**end**

---

Στους παραπάνω αλγορίθμους έχει θεωρηθεί ομοιόμορφη διαμέριση μήκους  $4n_s^2$  στα διανύσματα  $z$  και  $t$ .

Η παράλληλη διαδικασία της μεθόδου BiCGSTAB, περιλαμβάνει την επίλυση του γραμμικού συστήματος με τον πίνακα προρύθμισης

$$M = \begin{bmatrix} D_R & O \\ -H_R & D_B \end{bmatrix}, \quad (2.57)$$

της μεθόδου Gauss-Seidel σύμφωνα με την Red-Black διαμέριση του Collocation πίνακα  $C_P$ . Ο παρακάτω αλγόριθμος περιγράφει αυτή τη παράλληλη διαδικασία.

---

Παράλληλος αλγόριθμος επίλυσης  $M \cdot z = s$

---

- Βήμα 1 : Παράλληλη επίλυση  $D_R \cdot z_{1:4n_s^3} = s_{1:4n_s^3}$
  - Βήμα 2 : Παράλληλος υπολογισμός  $s_{4n_s^3+1:8n_s^3} = s_{4n_s^3+1:8n_s^3} - H_R \cdot s_{1:4n_s^3}$
  - Βήμα 3 : Παράλληλη επίλυση  $D_B \cdot z_{4n_s^3+1:8n_s^3} = s_{4n_s^3+1:8n_s^3}$
- 

Το βήμα 2 του παραπάνω αλγορίθμου μπορεί να περιγραφεί από τον αλγόριθμο παράλληλου υπολογισμού πολλαπλασιασμού του υποπίνακα  $H_R$  με διάνυσμα σύμφωνα με τον

αλγόριθμο που παρουσιάστηκε παραπάνω. Οι παράλληλες διαδικασίες του βήματος 1 και 3 μπορούν να περιγραφούν από τους παράλληλους αλγορίθμους που ακολουθούν.

---

### Παράλληλος αλγόριθμος επίλυσης $D_R \cdot z = s$

---

Παράλληλη επίλυση  $A'_2 \cdot z_1 = s_1$

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Παράλληλη επίλυση  $A'_1 \cdot z_i = \frac{1}{2} \cdot (s_i + s_{i+1})$

Παράλληλη επίλυση  $A'_2 \cdot z_{i+1} = \frac{1}{2} \cdot (s_{i+1} - s_i)$

**end**

Παράλληλη επίλυση  $-A'_2 \cdot z_{n_s} = s_{n_s}$

---

### Παράλληλος αλγόριθμος επίλυσης $D_B \cdot z = s$

---

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Παράλληλη επίλυση  $A'_1 \cdot z_i = \frac{1}{2} \cdot (s_i + s_{i+1})$

Παράλληλη επίλυση  $A'_2 \cdot z_{i+1} = \frac{1}{2} \cdot (s_{i+1} - s_i)$

**end**

---

Τα διανύσματα  $s, z$  των παραπάνω αλγορίθμων έχει θεωρηθεί ότι έχουν διαμεριστεί ομοιόμορφα κατά  $4n_s^2$ . Οι παράλληλοι αλγόριθμοι επίλυσης  $D_R \cdot z = s$  και  $D_B \cdot z = s$  περιλαμβάνουν τις παράλληλες διαδικασίες επίλυσης γραμμικών συστημάτων με πίνακες συντελεστών των αγνώστων τους πίνακες  $A'_1$  και  $A'_2$ . Τα συστήματα αυτά επιλύονται με την εφαρμογή του παράλληλου αλγορίθμου της Τεχνικής Πολυπλέγματος που ακολουθεί.

---

### Παράλληλος Αλγόριθμος επίλυσης V-Κύκλου

---

- Παράλληλη Εξομάλυνση η  $Au = f \quad v_1$ .
- Παράλληλος Υπολογισμός  $f^{2h} = I_h^{2h} r^h$ .
  - Παράλληλη Εξομάλυνση  $A^{2h} u^{2h} = f^{2h} \quad v_1$ .
  - Παράλληλος Υπολογισμός  $f^{4h} = I_{2h}^{4h} r^{2h}$ .
    - Παράλληλη Εξομάλυνση  $A^{4h} u^{4h} = f^{4h} \quad v_1$ .
    - Παράλληλος Υπολογισμός  $f^{8h} = I_{4h}^{8h} r^{4h}$ .



- Παράλληλη Επίλυση  $A^{Lh}\mathbf{u}^{Lh} = \mathbf{f}^{Lh}$ .
- Παράλληλη Διόρθωση  $\mathbf{v}^{4h} \leftarrow \mathbf{v}^{4h} + I_{8h}^{4h}\mathbf{v}^{8h}$ .
- Παράλληλη Εξομάλυνση  $A^{4h}\mathbf{u}^{4h} = \mathbf{f}^{4h}$ .
- Παράλληλη Διόρθωση  $\mathbf{v}^{2h} \leftarrow \mathbf{v}^{2h} + I_{4h}^{2h}\mathbf{v}^{4h}$ .
- Παράλληλη Εξομάλυνση  $A^{2h}\mathbf{u}^{2h} = \mathbf{f}^{2h}$ .
- Παράλληλη Διόρθωση  $\mathbf{v}^h \leftarrow \mathbf{v}^h + I_{2h}^h\mathbf{v}^{2h}$ .
- Παράλληλη Εξομάλυνση  $A^h\mathbf{u}^h = \mathbf{f}^h$ .

Οι επί μέρους διαδικασίες του αλγορίθμου της τεχνικής πολυπλέγματος περιλαμβάνουν πράξεις οι οποίες μπορούν να εκτελεστούν ταυτόχρονα με πολύ απλούς παράλληλους αλγορίθμους και έτσι δεν περιγράφονται. Στους παραπάνω αλγορίθμους εμφανίζονται οι πράξεις πολλαπλασιασμού πίνακα με διάνυσμα, η επίλυση γραμμικού συστήματος με πίνακα διάστασης  $4n_s^2$ . Πίνακες αυτού του μεγέθους αναφέρονται σε προβλήματα πεπερασμένων στοιχείων δυο διαστάσεων.

Ο πολλαπλασιασμός μπορεί να περιγραφεί με τον παρακάτω αλγόριθμο.

---

Παράλληλος αλγόριθμος υπολογισμού  $A_i \cdot \mathbf{z} = \mathbf{t}$  για  $i = 1, \dots, 4$ .

---

- *RED διαδικασία:*

$$\text{Παράλληλος υπολογισμός } t_1^R = K_2 \cdot z_1^R + K_3 \cdot z_1^B - K_4 \cdot z_2^B$$

**for**  $i = 1, 3, \dots, n_s - 3$  **με βήμα 2**

$$\text{Παράλληλος υπολογισμός } t_{i+1}^R = K_1 \cdot z_{i+1}^R - K_2 \cdot z_{i+2}^R + K_3 \cdot z_i^B + K_4 \cdot z_{i+1}^B$$

$$\text{Παράλληλος υπολογισμός } t_{i+2}^R = K_1 \cdot z_{i+1}^R + K_2 \cdot z_{i+2}^R + K_3 \cdot z_{i+2}^B - K_4 \cdot z_{i+3}^B$$

**end**

$$\text{Παράλληλος υπολογισμός } t_{n_s}^R = -K_2 \cdot z_{n_s}^R + K_3 \cdot z_{n_s-1}^B + K_4 \cdot z_{n_s}^B$$

- *BLACK διαδικασία:*

Παράλληλος υπολογισμός  $t_1^B = K_4 \cdot z_1^R + K_1 \cdot z_1^B - K_2 \cdot z_2^B$

**for**  $i = 1, 3, \dots, n_s - 3$  **με βήμα 2**

Παράλληλος υπολογισμός  $t_{i+1}^B = K_3 \cdot z_{i+1}^R - K_4 \cdot z_{i+2}^R + K_1 \cdot z_i^B + K_2 \cdot z_{i+1}^B$

Παράλληλος υπολογισμός  $t_{i+2}^B = K_3 \cdot z_{i+1}^R + K_4 \cdot z_{i+2}^R + K_1 \cdot z_{i+2}^B - K_2 \cdot z_{i+3}^B$

**end**

Παράλληλος υπολογισμός  $t_{n_s}^B = -K_4 \cdot z_{n_s}^R + K_1 \cdot z_{n_s-1}^B + K_2 \cdot z_{n_s}^B$

Για την επίλυση του γραμμικού συστήματος  $A'_i z = s$  θα χρησιμοποιηθεί μία επέκταση του παράλληλου αλγορίθμου, ο οποίος παρουσιάστηκε στην εργασία [15]. Η τροποποίηση του παράλληλου αλγορίθμου βασίζεται σε δύο παράλληλες διαδικασίες. Η πρώτη αναφέρεται στην εφαρμογή της παράλληλης διαδικασίας της μεθόδου Cyclic Reduction για την επίλυση των βασικών γραμμικών συστημάτων  $K_i z = s$ ,  $i = 1, 2$ . Η δεύτερη αναφέρεται στη διαδικασία του πολλαπλασιασμού  $K_i z = t$ , η οποία περιγράφεται με τον παρακάτω αλγόριθμο.

Παράλληλος αλγόριθμος υπολογισμού του  $K_i \cdot z = t$  για  $i = 1, \dots, 4$ .

Υπολογισμός  $t_1 = \alpha_2 \cdot z_1 + \alpha_3 \cdot z_2 - \alpha_4 \cdot z_3$

Υπολογισμός  $t_2 = \alpha_4 \cdot z_1 + \alpha_1 \cdot z_2 - \alpha_2 \cdot z_3$

**for**  $i = 3, 5, \dots, n_s - 2$  **με βήμα 2 do in parallel**

Υπολογισμός  $t_i = \alpha_1 \cdot z_{i-1} + \alpha_2 \cdot z_i + \alpha_3 \cdot z_{i+1} - \alpha_4 \cdot z_{i+2}$

Υπολογισμός  $t_{i+1} = \alpha_3 \cdot z_{i-1} + \alpha_4 \cdot z_i + \alpha_1 \cdot z_{i+1} - \alpha_2 \cdot z_{i+2}$

**enddo**

Υπολογισμός  $t_{n_s-1} = \alpha_1 \cdot z_{n_s-2} + \alpha_2 \cdot z_{n_s-1} - \alpha_4 \cdot z_{n_s}$

Υπολογισμός  $t_{n_s} = \alpha_3 \cdot z_{n_s-2} + \alpha_4 \cdot z_{n_s-1} - \alpha_2 \cdot z_{n_s}$

Είναι φανερό ότι οι παραπάνω παράλληλοι αλγόριθμοι μαζί με τη διαδικασία του Cyclic Reduction προσφέρουν ένα ακόμη επίπεδο παραλληλοποίησης στη μέθοδό μας. Σε αυτούς τους δύο παράλληλους αλγορίθμους, η διεξαγωγή των πράξεων με την εμπλοκή των βασικών πινάκων  $K_i$  μπορεί να πραγματοποιηθεί με τη χρήση των 16 στοιχείων τους. Έτσι κατά την υλοποίηση των αλγορίθμων, δεν είναι απαραίτητη η χρήση δομών δεδομένων δύο διαστάσεων στη διεξαγωγή των πράξεων, αλλά μόνο των 16 στοιχείων τους, τα οποία συνθέτουν τους πίνακες αυτούς.

## 2.5 Παράλληλος αλγόριθμος για υπολογιστικούς πυρήνες CPU αρχιτεκτονικών

Ο παράλληλος αλγόριθμος της μεθόδου, που περιγράφηκε στην προηγούμενη ενότητα, κατά την αποδοτική υλοποίηση του σε αρχιτεκτονικές κοινής μνήμης με σχετικά μικρό (ως προς το μέγεθος του προβλήματος) πλήθος υπολογιστικών πυρήνων, μπορεί να εκμεταλλευτεί μόνο το τρίτο επίπεδο της παραλληλοποίησης. Δηλαδή να γίνεται ταυτόχρονος προσδιορισμός αγνώστων που αναφέρονται στο ίδιο επίπεδο διακριτοποίησης ως προς την τρίτη διάσταση. Οπότε οι παράλληλες διαδικασίες του επιλυτή θα περιγράφονται από τους παρακάτω αλγορίθμους.

---

Παράλληλος αλγόριθμος υπολογισμού  $t = D_R \cdot z$

---

Υπολογισμός  $t_1 = A'_2 \cdot z_1$

*!\$OMP PARALLEL DO*

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Υπολογισμός  $t_i = A'_1 \cdot z_i - A'_2 \cdot z_{i+1}$

Υπολογισμός  $t_{i+1} = A'_1 \cdot z_i + A'_2 \cdot z_{i+1}$

**end**

*!\$OMP END PARALLEL DO*

Υπολογισμός  $t_{n_s} = -A'_2 \cdot z_{n_s}$

---

---

Παράλληλος αλγόριθμος υπολογισμού  $t = H_B \cdot z$

---

*!\$OMP PARALLEL DO*

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Υπολογισμός  $t_i = A'_3 \cdot z_i - A'_4 \cdot z_{i+1}$

Υπολογισμός  $t_{i+1} = A'_3 \cdot z_i + A'_4 \cdot z_{i+1}$

**end**

*!\$OMP END PARALLEL DO*

---

---

Παράλληλος αλγόριθμος υπολογισμού  $t = D_B \cdot z$

---

*!\$OMP PARALLEL DO*

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Υπολογισμός  $t_i = A'_1 \cdot z_i - A'_2 \cdot z_{i+1}$

Υπολογισμός  $t_{i+1} = A'_1 \cdot z_i + A'_2 \cdot z_{i+1}$

**end**

*!\$OMP END PARALLEL DO*

---

Παράλληλος αλγόριθμος υπολογισμού  $t = H_R \cdot z$

---

Υπολογισμός  $t_1 = A'_4 \cdot z_1$

*!\$OMP PARALLEL DO*

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Υπολογισμός  $t_i = A'_3 \cdot z_i - A'_4 \cdot z_{i+1}$

Υπολογισμός  $t_{i+1} = A'_3 \cdot z_i + A'_4 \cdot z_{i+1}$

**end**

*!\$OMP END PARALLEL DO*

Υπολογισμός  $t_{n_s} = -A'_4 \cdot z_{n_s}$

---

Παράλληλος αλγόριθμος επίλυσης  $D_R \cdot z = s$

---

Επίλυση  $A'_2 \cdot z_1 = s_1$

*!\$OMP PARALLEL DO*

**for**  $i = 2, 4, \dots, n_s - 2$  με βήμα 2

Επίλυση  $A'_1 \cdot z_i = \frac{1}{2} \cdot (s_i + s_{i+1})$

Επίλυση  $A'_2 \cdot z_{i+1} = \frac{1}{2} \cdot (s_{i+1} - s_i)$

**end**

*!\$OMP END PARALLEL DO*

Επίλυση  $-A'_2 \cdot z_{n_s} = s_{n_s}$

---

---

Παράλληλος αλγόριθμος επίλυσης  $D_B \cdot z = s$

---

*!\$OMP PARALLEL DO*

**for**  $i = 1, 3, \dots, n_s - 1$  με βήμα 2

Επίλυση  $A'_1 \cdot z_i = \frac{1}{2} \cdot (s_i + s_{i+1})$

Επίλυση  $A'_2 \cdot z_{i+1} = \frac{1}{2} \cdot (s_{i+1} - s_i)$

**end**

*!\$OMP END PARALLEL DO*

---

## 2.6 Παράλληλος αλγόριθμος για υπολογιστικούς πυρήνες αρχιτεκτονικών με επιταχυντές

Σε περίπτωση όπου η αρχιτεκτονική υπολογισμών περιλαμβάνει μεγάλο πλήθος από σχετικά αδύναμους σε υπολογιστική ισχύ υπολογιστικούς πυρήνες, όπως η περίπτωση χρήσης συσκευών επιτάχυνσης, τότε η αποδοτική υλοποίηση του παράλληλου αλγορίθμου επίλυσης θα εκμεταλλευτεί το τρίτο ή και το δεύτερο επίπεδο της παραλληλοποίησης του. Οπότε θα εκτελεστούν ταυτόχρονα οι πράξεις που συμπεριλαμβάνει η διαδικασία προσδιορισμού αγνώστων οι οποίοι αναφέρονται στο ίδιο επίπεδο διακριτοποίησης ως προς την δεύτερη και την πρώτη διάσταση. Έτσι θα υλοποιηθεί ο παράλληλος αλγόριθμος της επίλυσης σύμφωνα με την τεχνική πολυπλέγματος και την διαδικασία του Cyclic reduction καθώς και των παρακάτω αλγορίθμων πολλαπλασιασμού πινάκων με διανύσματα.

---

Παράλληλος αλγόριθμος υπολογισμού  $A_i \cdot z = t$  για  $i = 1, \dots, 4$ .

---

- *RED διαδικασία:*

$$\text{Υπολογισμός } t_1^R = K_2 \cdot z_1^R + K_3 \cdot z_1^B - K_4 \cdot z_2^B$$

*!\$ACC PARALLEL LOOP*

**for**  $i = 1, 3, \dots, n_s - 3$  **με βήμα 2**

$$\text{Υπολογισμός } t_{i+1}^R = K_1 \cdot z_{i+1}^R - K_2 \cdot z_{i+2}^R + K_3 \cdot z_i^B + K_4 \cdot z_{i+1}^B$$

$$\text{Υπολογισμός } t_{i+2}^R = K_1 \cdot z_{i+1}^R + K_2 \cdot z_{i+2}^R + K_3 \cdot z_{i+2}^B - K_4 \cdot z_{i+3}^B$$

**end**

*!\$ACC END PARALLEL*

$$\text{Υπολογισμός } t_{n_s}^R = -K_2 \cdot z_{n_s}^R + K_3 \cdot z_{n_s-1}^B + K_4 \cdot z_{n_s}^B$$

- *BLACK διαδικασία:*

$$\text{Υπολογισμός } t_1^B = K_4 \cdot z_1^R + K_1 \cdot z_1^B - K_2 \cdot z_2^B$$

*!\$ACC PARALLEL LOOP*

**for**  $i = 1, 3, \dots, n_s - 3$  **με βήμα 2**

$$\text{Υπολογισμός } t_{i+1}^B = K_3 \cdot z_{i+1}^R - K_4 \cdot z_{i+2}^R + K_1 \cdot z_i^B + K_2 \cdot z_{i+1}^B$$

$$\text{Υπολογισμός } t_{i+2}^B = K_3 \cdot z_{i+1}^R + K_4 \cdot z_{i+2}^R + K_1 \cdot z_{i+2}^B - K_2 \cdot z_{i+3}^B$$

**end**

*!\$ACC END PARALLEL*

$$\text{Υπολογισμός } t_{n_s}^B = -K_4 \cdot z_{n_s}^R + K_1 \cdot z_{n_s-1}^B + K_2 \cdot z_{n_s}^B$$

---

Παράλληλος αλγόριθμος υπολογισμού του  $K_i \cdot z = t$  για  $i = 1, \dots, 4$ .

---

$$\text{Υπολογισμός } t_1 = \alpha_2 \cdot z_1 + \alpha_3 \cdot z_2 - \alpha_4 \cdot z_3$$

$$\text{Υπολογισμός } t_2 = \alpha_4 \cdot z_1 + \alpha_1 \cdot z_2 - \alpha_2 \cdot z_3$$

*!\$ACC PARALLEL LOOP*

**for**  $i = 3, 5, \dots, n_s - 2$  **με βήμα 2**

$$\text{Υπολογισμός } t_i = \alpha_1 \cdot z_{i-1} + \alpha_2 \cdot z_i + \alpha_3 \cdot z_{i+1} - \alpha_4 \cdot z_{i+2}$$

$$\text{Υπολογισμός } t_{i+1} = \alpha_3 \cdot z_{i-1} + \alpha_4 \cdot z_i + \alpha_1 \cdot z_{i+1} - \alpha_2 \cdot z_{i+2}$$

**end**

*!\$ACC END PARALLEL*

$$\text{Υπολογισμός } t_{n_s-1} = \alpha_1 \cdot z_{n_s-2} + \alpha_2 \cdot z_{n_s-1} - \alpha_4 \cdot z_{n_s}$$

$$\text{Υπολογισμός } t_{n_s} = \alpha_3 \cdot z_{n_s-2} + \alpha_4 \cdot z_{n_s-1} - \alpha_2 \cdot z_{n_s}$$

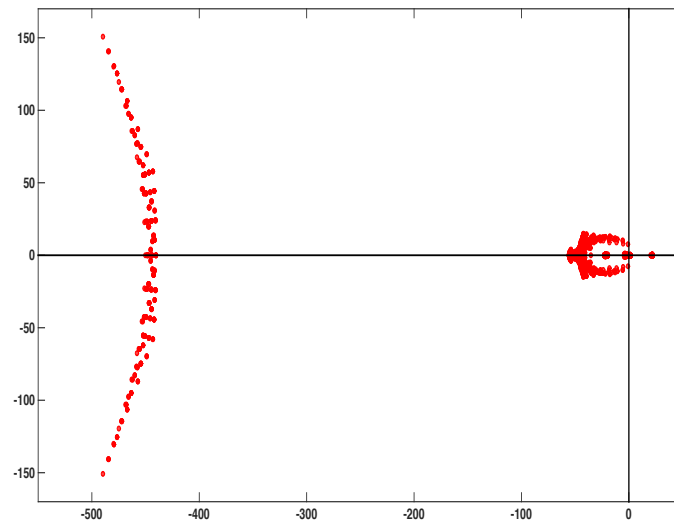
---



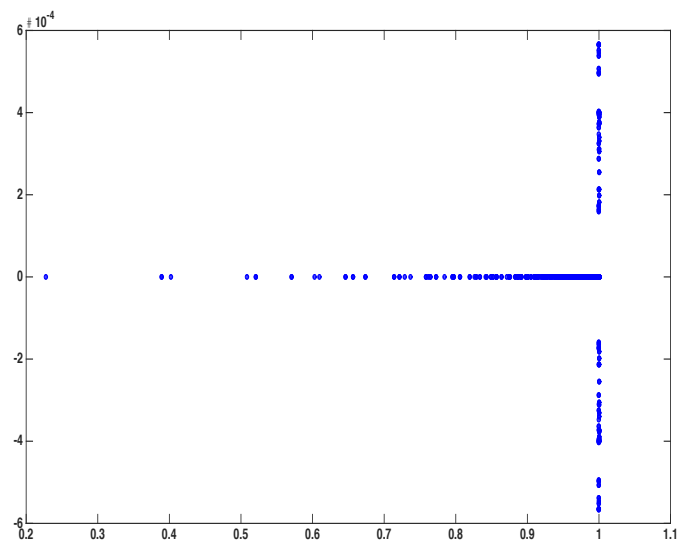
## Κεφάλαιο 3

# Υλοποίηση Παράλληλου Αλγορίθμου

Το κεφάλαιο αυτό παρουσιάζει την υλοποίηση του παράλληλου αλγορίθμου που παρουσιάστηκε στην προηγούμενη ενότητα και τα αποτελέσματα από τη διερεύνηση της συμπεριφοράς κατά την εκτέλεσή του. Αναπτύχθηκαν δύο εφαρμογές υλοποίησης του αλγορίθμου για παράλληλες αρχιτεκτονικές κοινής μνήμης με τη διεξαγωγή CPU ή CPUGPU υπολογισμών. Οι εφαρμογές αναπτύχθηκαν με τη χρήση της γλώσσας προγραμματισμού FORTRAN για υπολογισμούς διπλής ακρίβειας, καθώς και με τη χρήση των προτύπων `open_MP` [23] και `open_ACC` [22]. Οι δοκιμές πραγματοποιήθηκαν σε ένα μηχάνημα τύπου Dell R730, το οποίο διαθέτει δύο επεξεργαστές τύπου Xeon E5-2630 των 8 πυρήνων, συγχρονισμένους στα 2.4 GHz. Η συνολική μνήμη του υπολογιστικού συστήματος είναι 144 GB, ενώ το λειτουργικό του σύστημα είναι τύπου Linux έκδοσης Ubuntu 14.04. Το μηχάνημα διαθέτει έναν επιταχυντή υπολογισμών τύπου Nvidia Tesla K40 [21] συνδεδεμένο μέσω διαύλου τύπου PCI Express 3.0. Η συγκεκριμένη κάρτα γραφικών (GPU) διαθέτει 12 GB αποκλειστικής μνήμης για 2880 υπολογιστικούς πυρήνες αρχιτεκτονικής Kepler. Για την ανάπτυξη των εφαρμογών χρησιμοποιήθηκε ο μεταγλωττιστής PGI-CDK [25] έκδοσης 17.2, καθώς και οι βιβλιοθήκες cuBLAS [8] από το λογισμικό CUDA [9] έκδοσης 7.5.



**Σχήμα 3.1:** Οι ιδιοτιμές του πίνακα  $C$ .



**Σχήμα 3.2:** Οι ιδιοτιμές του προρυθμισμένου πίνακα.

Σαν πρόβλημα δοκιμής χρησιμοποιήθηκε το πρόβλημα συνοριακών τιμών το οποίο ορίζεται στον μοναδιαίο κύβο με συνορικές συνθήκες τύπου Dirichlet και επιδέχεται τη λύση :

$$u(x, y, z) = 10 \phi(x) \phi(y) \phi(z) , \quad (3.1)$$

όπου

$$\phi(x) = e^{-100(x-0.1)^2} (x^2 - x) \quad (3.2)$$

για την περίπτωση του πρώτου προβλήματος δοκιμής,

$$\phi(x) = x^{\frac{9}{2}} (x - 1)^2 \quad (3.3)$$

για την περίπτωση του δεύτερου προβλήματος δοκιμής και

$$u(x, y, z) = 10 \sin(x/10) \sin(10y) \sin(z) (x - 1)(y - 1)(z - 1) \quad (3.4)$$

το τρίτο πρόβλημα δοκιμής.

Η μελέτη των ιδιοτιμών του πίνακα  $C$ , οι οποίες εμφανίζονται στο διάγραμμα του σχήματος 3.1, μας υποδεικνύει ότι χρειάζεται προρύθμιση του Collocation πίνακα, πριν την επίλυση του γραμμικού συστήματος με κάποια επαναήπτική μέθοδο. Αυτό συμβαίνει διότι βρίσκονται και στα δύο μιγαδικά ημιεπίπεδα με μεγάλα φανταστικά μέρη, ενώ κάποιες από αυτές βρίσκονται πολύ κοντά στο μηδέν. Οπότε η διαδικασία προρύθμισης θα πρέπει να παράξει πίνακα συντελεστών των αγνώστων με ιδιοτιμές πολύ κοντά στη μονάδα. Η εφαρμογή της Gauss-Seidel προρύθμισης στον πίνακα παράγει τον προρυθμισμένο Collocation πίνακα, ο οποίος έχει τις ιδιοτιμές που εμφανίζονται στο διάγραμμα του σχήματος 3.2. Όπως παρατηρούμε, οι ιδιοτιμές έχουν μεταφερθεί σε μία περιοχή,

η οποία βρίσκεται πολύ κοντά στη μονάδα, ενώ ταυτόχρονα τα μιγαδικά τους μέρη είναι πολύ μικρά σε μέτρο. Αυτό μας οδηγεί στο συμπέρασμα σύγκλισης της μεθόδου BiCGSTAB, ενώ ο Gauss-Seidel πίνακας προρύθμισης οδήγησε σε μία αποτελεσματική διαδικασία προρύθμισης του συστήματος.

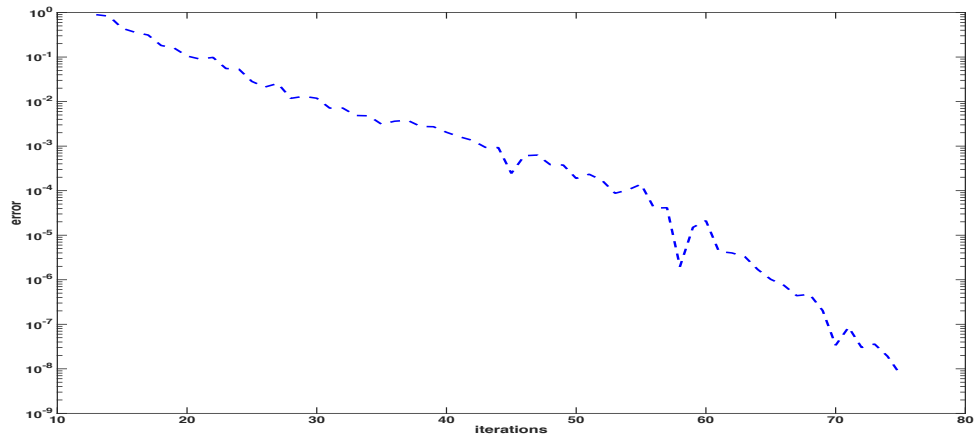
### 3.1 Πρόβλημα δοκιμής 1

Στη συνέχεια έγινε μελέτη της συμπεριφοράς σύγκλισης της μεθόδου BiCGSTAB με Gauss-Seidel προρύθμιση για την επίλυση του Collocation γραμμικού συστήματος. Επιλύθηκαν προβλήματα ομοιόμορφης διακριτοποίησης από 8 έως 256 πεπερασμένα στοιχεία ανά κατεύθυνση. Ο πίνακας παρακάτω εμφανίζει τους βαθμούς ελευθερίας του κάθε προβλήματος και τα βήματα σύγκλισης της επαναληπτικής μεθόδου που απαιτήθηκαν για την επίλυση του κάθε προβλήματος. Το κριτήριο σύγκλισης της μεθόδου για όλα τα προβλήματα είναι το

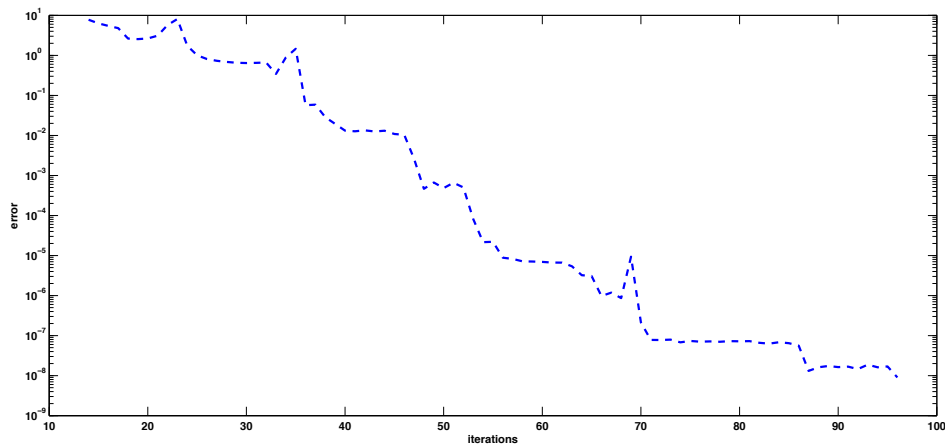
$$\frac{\|r^{(n)}\|_2}{\|b\|_2} < 10^{-8}, \quad (3.5)$$

όπου  $r^{(n)}$  το υπόλοιπο προσέγγισης της λύσης του γραμμικού συστήματος στο βήμα  $n$  της επαναληπτικής μεθόδου. Ο μέγιστος αριθμός βημάτων είναι 1000, ενώ αρχική προσέγγιση της λύσης του γραμμικού συστήματος είναι το δεξί του μέλος σε κάθε περίπτωση. Στον ίδιο πίνακα εμφανίζονται οι νόρμες 2 και άπειρο του υπολοίπου προσέγγισης του γραμμικού συστήματος καθώς και του σφάλματος προσέγγισης της Collocation μεθόδου από τη θεωρητική λύση του ΠΣΤ. Οι τελευταίες δύο στήλες εμφανίζουν την τάξη σύγκλισης της μεθόδου ως προς τη νόρμα του απείρου και την  $L2$  για το σφάλμα της Collocation μεθόδου.

$n_s$	dof	Iter	$\ b - Ct\ _\infty$	$\ b - Ct\ _2$	$\ u - t\ _\infty$	$\ u - t\ _2$	ofc $\ \cdot\ _\infty$	ofc $\ \cdot\ _2$
8	5832	11	7.143e-9	6.368e-8	1.491e-3	1.633e-3	-	-
16	39304	20	9.605e-10	3.736e-8	9.193e-5	1.205e-4	4.02	4.90
32	287496	38	7.929e-9	7.730e-7	5.290e-6	1.898e-5	4.12	4.16
64	2197000	75	6.407e-9	2.226e-6	3.225e-7	3.284e-6	4.04	4.03
128	17173512	152	7.521e-9	5.498e-6	2.003e-8	5.771e-7	4.01	4.01
256	135796744	325	1.964e-8	5.105e-6	1.250e-9	1.045e-7	4.00	3.97



**Σχήμα 3.3:** Διάγραμμα σύγκλισης της μεθόδου BiCGSTAB για το πρώτο πρόβλημα.



**Σχήμα 3.4:** Διάγραμμα σύγκλισης της μεθόδου BiCGSTAB για το δεύτερο πρόβλημα.

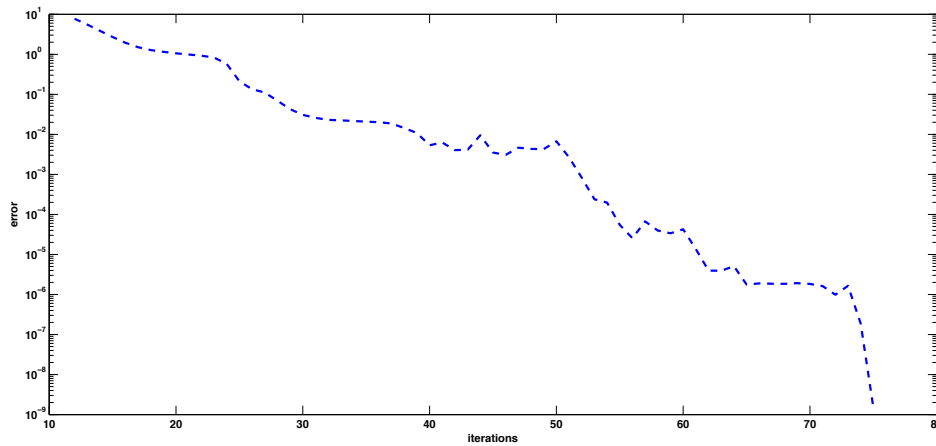
Το διάγραμμα σύγκλισης της επαναληπτικής μεθόδου BiCGSTAB του σχήματος 3.3 εμφανίζει τη λογαριθμική μείωση του σφάλματος επίλυσης του γραμμικού συστήματος για πρόβλημα διάστασης  $n_s = 64$ .

## 3.2 Πρόβλημα δοκιμής 2

Η συμπεριφορά σύγκλισης της μεθόδου για το δεύτερο πρόβλημα δοκιμής παρουσιάζεται στον επόμενο πίνακα, ενώ το γράφημα του σχήματος 3.4 εμφανίζει την λογαριθμική μείωση του σφάλματος για πρόβλημα μεγέθους με διακριτοποίηση  $n_s = 64$ .

$n_s$	Iterations	$\ \mathbf{b} - C\mathbf{t}\ _\infty$	$\ \mathbf{b} - C\mathbf{t}\ _2$	$\ \mathbf{u} - \mathbf{t}\ _\infty$	$\ \mathbf{u} - \mathbf{t}\ _2$	ofc
8	11	1.700e-12	2.098e-11	4.884e-8	2.184e-7	-
16	22	2.418e-11	6.925e-10	3.110e-9	3.835e-8	3.97
32	49	4.465e-11	3.420e-9	2.021e-10	6.748e-9	3.94
64	96	5.363e-11	1.203e-8	1.255e-11	1.212e-9	4.01
128	427	1.152e-10	1.215e-7	2.426e-12	1.260e-9	-
256	1000	3.847e-8	7.120e-5	2.005e-10	2.893e-7	-

Παρατηρήθηκε ότι στο πρόβλημα διάστασης  $n_s = 128$  η επαναληπτική μέθοδος BiCG-STAB με αρχική προσέγγιση της λύσης του συστήματος το δεξί μέλος του δεν καταφέρνει να επεκτείνει τη βάση του υποχώρου πάνω από 427 στοιχεία. Έτσι η επαναληπτική διαδικασία διακόπηκε με το σφάλμα προσέγγισης που εμφανίζεται στον παραπάνω πίνακα. Επίσης για το πρόβλημα διάστασης  $n_s = 256$  η επαναληπτική διαδικασία διακόπηκε ξανά από τον προκαθορισμένο μέγιστο αριθμό των 1000 επαναλήψεων χωρίς να έχει επιτευχθεί η επιθυμητή ακρίβεια προσέγγισης. Για το λόγο αυτό δεν υπολογίστηκε και η τάξη ακρίβειας της μεθόδου για αυτές τις δύο περιπτώσεις.



**Σχήμα 3.5:** Διάγραμμα σύγκλισης της μεθόδου BiCGSTAB για το τρίτο πρόβλημα.

### 3.3 Πρόβλημα δοκιμής 3

Ο παρακάτω πίνακας εμφανίζει την συμπεριφορά σύγκλισης της μεθόδου για το τρίτο πρόβλημα δοκιμής. Παρατηρούμε μείωση της τάξης ακρίβειας της Collocation μεθόδου για πυκνές διακριτοποιήσεις εξαιτίας της μεγάλης τιμής του κριτηρίου σύγκλισης για

αυτό το παράδειγμα. Δηλαδή η μέθοδος χρειάζεται περισσότερα βήματα για να συγκλίνει, το οποίο μπορεί να πραγματοποιηθεί αν μειωθεί η τιμή του τελεστή κριτηρίου σύγκλισης από το δεδομένο  $10^{-8}$ .

$n_s$	Iterations	$\ \mathbf{b} - C\mathbf{t}\ _\infty$	$\ \mathbf{b} - C\mathbf{t}\ _2$	$\ \mathbf{u} - \mathbf{t}\ _\infty$	$\ \mathbf{u} - \mathbf{t}\ _2$	ofc
8	10	8.108e-8	1.007e-6	8.848e-5	6.498e-4	-
16	18	2.730e-8	1.019e-6	5.192e-6	1.095e-4	4.10
32	36	5.524e-8	4.197e-6	3.286e-7	1.910e-5	3.98
64	75	1.528e-8	3.585e-6	2.032e-8	3.366e-6	4.02
128	153	3.459e-8	3.395e-5	1.459e-9	6.540e-7	3.80
256	300	6.058e-8	1.670e-4	9.624e-10	1.410e-6	-

Το γράφημα της λογαριθμικής μείωσης του σφάλματος της επαναληπτικής μεθόδου για το πρόβλημα μεγέθους  $n_s = 64$  περιλαμβάνεται στην εικόνα 3.5. Παρατηρούμε την διαρκή αυξομείωση του σε σχέση με τα επαναληπτικά βήματα. Η συμπεριφορά αυτή αποτελεί χαρακτηριστικό της BiCGSTAB μεθόδου.

### 3.4 Αποτελέσματα παράλληλης επίλυσης

Η μελέτη της συμπεριφοράς του παράλληλου αλγορίθμου πραγματοποιήθηκε σε δύο αρχιτεκτονικές κοινής μνήμης για το πρώτο πρόβλημα δοκιμής. Στην πρώτη χρησιμοποιήθηκαν αποκλειστικά CPU υπολογιστικοί πυρήνες, ενώ στη δεύτερη έγινε χρήση των υπολογιστικών πυρήνων του επιταχυντή Tesla K40. Ο παρακάτω πίνακας εμφανίζει το χρόνο εκτέλεσης σε δευτερόλεπτα για κάθε ένα πρόβλημα επίλυσης από τη σειριακή έκδοση έως και τη χρήση 16 CPU πυρήνων.

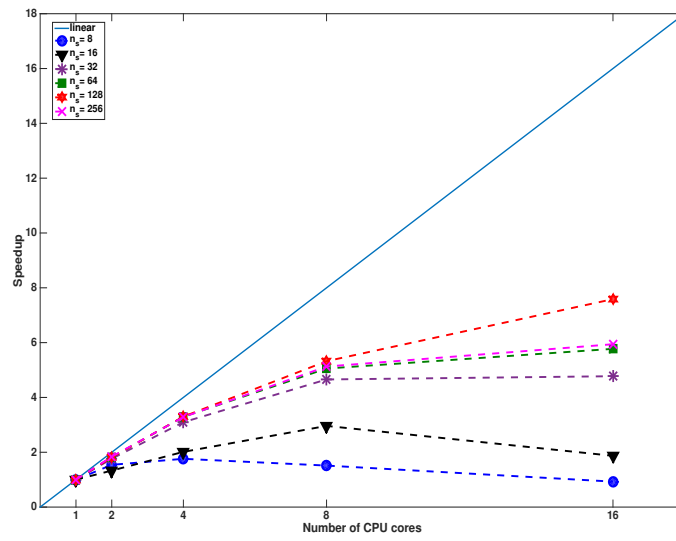
$n_s$	CPU 1	CPU 2	CPU 4	CPU 8	CPU 16
8	9.995e-2	6.482e-2	5.675e-2	6.600e-2	0.107
16	0.5937	0.4459	0.2947	0.204	0.317
32	7.5874	4.4509	2.5370	1.687	1.645
64	118.98	65.463	36.122	23.53	20.61
128	2033.6	1121.1	616.25	382.0	268.5
256	34808	18914	10599	6787	5858

Το γράφημα του σχήματος 3.6 εμφανίζει την επιτάχυνση που σημειώθηκε σε κάθε περίπτωση.

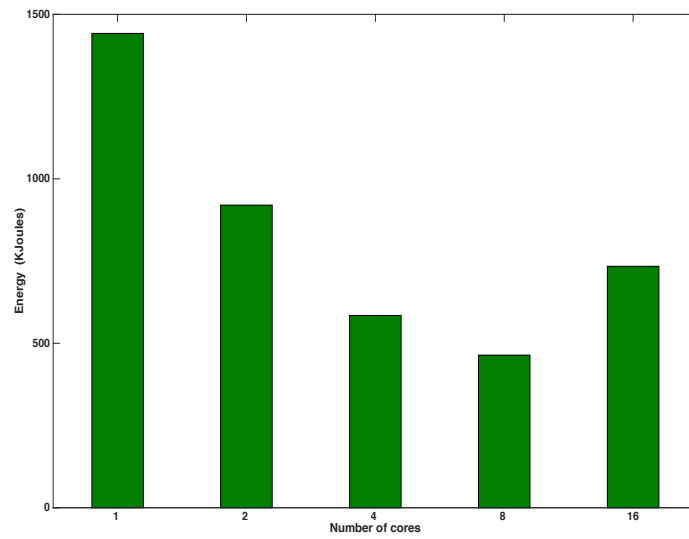
Η κατανάλωση ενέργειας κατά την υλοποίηση της εφαρμογής μετρήθηκε με τη χρήση της εφαρμογής **likwid**. Με αυτό το λογισμικό παρακολούθησης κατανάλωσης της ενέργειας των CPU υπολογιστικών πυρήνων και της μνήμης, μετρήθηκαν οι τιμές της ενέργειας σε KJoules που απαιτήθηκαν κατά την επίλυση του προβλήματος διάστασης  $n_s = 256$ . Η κατανάλωση ενέργειας απεικονίζεται στο γράφημα του σχήματος 3.7. Όπως παρατηρούμε, η χρήση 8 υπολογιστικών πυρήνων από έναν επεξεργαστή αποτελεί την ενεργειακά αποδοτικότερη επιλογή.

Ο παράλληλος αλγόριθμος υλοποιήθηκε και σε μία εφαρμογή κατάλληλη για υπολογιστικά συστήματα με επιταχυντές. Η ανάπτυξη της εφαρμογής έγινε με τη χρήση της γλώσσας προγραμματισμού FORTRAN και του προτύπου open\_ACC. Στο υπολογιστικό σύστημα, το οποίο χρησιμοποιήθηκε για τη μελέτη της συμπεριφοράς του αλγορίθμου,





**Σχήμα 3.6:** Η επιτάχυνση σε κάθε περίπτωση.



**Σχήμα 3.7:** Η κατανάλωση ενέργειας των CPU υπολογιστικών πυρήνων.

δεν παρατηρήθηκε επιτάχυνση. Αυτό οφείλεται στο γεγονός αδυναμίας επίλυσης μεγάλων προβλημάτων εξαιτίας του περιορισμένου μεγέθους μνήμης στα 12GB του επιταχυντή. Τα προβλήματα που μπορούσαν να επιλυθούν με τη χρήση του ήταν μικρότερα από  $n_s = 128$ , οπότε στα δύο πρώτα επίπεδα παραλληλοποίησης υπήρξαν προβλήματα μεγέθους  $4 \cdot 128^2$ , τα οποία θεωρούνται μικρής διάστασης. Αν και όλες οι πράξεις γραμμικής άλγεβρας πραγματοποιήθηκαν χωρίς την ύπαρξη στη μνήμη των υπολογιστικών συστημάτων ή τμημάτων του, παρ'όλα αυτά για το συγκεκριμένο επιταχυντή, ο οποίος διαθέτει περίπου 3000 υπολογιστικούς πυρήνες και 12GB κοινής μνήμης, η συγκεκριμένη παράλληλη υλοποίηση δεν υπήρξε αποδοτική. Όπως παρουσιάζεται στην εργασία [15] στο συγκεκριμένο τύπο επιταχυντών αποδοτική επίλυση μπορεί να υπάρξει για προβλήματα μεγέθους άνω των 512 πεπερασμένων στοιχείων ανά κατεύθυνση.

Τμηματικές μετρήσεις απόδοσης του παράλληλου αλγορίθμου έδειξαν ενθαρρυντικά αποτελέσματα. Έτσι κατά την υλοποίηση της εφαρμογής για το πρόβλημα με 128 πεπερασμένα στοιχεία ανά κατεύθυνση υπήρξε επιτάχυνση της τάξης  $1.3\times$  έναντι της σειριακής υλοποίησης. Οι μετρήσεις αυτές δεν συμπεριλαμβάνουν την εφαρμογή της προρύθμισης του γραμμικού συστήματος, διότι δεν ήταν αρκετή η μνήμη του επιταχυντή για την ολοκλήρωση της διαδικασίας αυτής στο συγκεκριμένο πρόβλημα. Οπότε αναμένεται να υπάρξει αποδοτική υλοποίηση του αλγορίθμου σε αρχιτεκτονικές με πολλαπλούς επιταχυντές, οι οποίοι συνολικά θα διαθέτουν μεγαλύτερα μεγέθη μνήμης κι έτσι θα είναι εφικτή η επίλυση μεγαλύτερου μεγέθους προβλημάτων.

## Κεφάλαιο 4

### Συμπεράσματα

Το αντικείμενο αυτής της διατριβής είναι η αποδοτική εφαρμογή της μεθόδου Πεπερασμένων Στοιχείων Hermite Collocation σε ρεαλιστικά προβλήματα τριών διαστάσεων. Τα βασικότερα συμπεράσματα από την εφαρμογή της μεθόδου, την κατασκευή παράλληλου αλγορίθμου επίλυσης και την υλοποίηση του σε παράλληλες αρχιτεκτονικές κοινής μνήμης, μπορούν να περιγραφούν συνοπτικά με τις παρακάτω διαπιστώσεις.

- Το παραγόμενο γραμμικό σύστημα έχει συγκεκριμένη δομή, οπότε είναι δυνατή η διεξαγωγή των πράξεων χωρίς την αποθήκευση του πίνακα συντελεστών των αγνώστων στη μνήμη του υπολογιστικού συστήματος, αλλά μόνο των βασικών τιμών που τον συνθέτουν. Αυτό έχει σαν αποτέλεσμα τη δραστική μείωση της απαιτούμενης μνήμης κατά την αριθμητική επίλυση.
- Η τάξη ακρίβειας προσέγγισης της μεθόδου παρατηρήθηκε να είναι κοντά στο τέσσερα για τρία προβλήματα δοκιμών που χρησιμοποιήθηκαν. Αυτό είναι πλήρως συμβατό με τη θεωρία της μεθόδου.
- Η χρήση του υβριδικού επιλυτή από τρεις διαφορετικές μεθόδους επίλυσης οδήγησε στην κατασκευή αποδοτικού παράλληλου αλγορίθμου για αρχιτεκτονικές υπολογισμών κοινής μνήμης.

- Η εφαρμογή της διαδικασίας ομαδοποίησης, χρωματισμού και επαναρίθμησης αγνώστων βοήθησε στην αύξηση των παράλληλων ιδιοτήτων του αλγεβρικού συστήματος.
- Η ύπαρξη τριών επιπέδων παράλληλων ιδιοτήτων στο αριθμητικό σχήμα επίλυσης επιτρέπει την κατάλληλη αξιοποίηση τους ανάλογα με την αρχιτεκτονική υλοποίησης του αλγορίθμου. Έτσι είναι εφικτή η αποδοτική επίλυση προβλημάτων για κάθε είδος παράλληλης αρχιτεκτονικής υπολογισμών. Σε περίπτωση ύπαρξης σχετικά μικρού αριθμού ισχυρών υπολογιστικών πυρήνων χρειάζεται να εκμεταλλευτούν οι παράλληλες ιδιότητες που προέκυψαν από την διακριτοποίηση της τρίτης διάστασης του προβλήματος. Εάν όμως είναι διαθέσιμο μεγάλο πλήθος υπολογιστικών πυρήνων μπορεί να γίνει εκμετάλλευση και των υπόλοιπων δυο επιπέδων παραλληλοποίησης, τα οποία αναφέρονται στην πρώτη και δεύτερη διάσταση της διακριτοποίησης.
- Η χρήση της τεχνικής cyclic reduction σε συνδυασμό με την τεχνική πολυπλέγματος διατηρούν σταθερό το πλήθος των επαναληπτικών τους βημάτων ως προς το μέγεθος του προβλήματος, σε αντίθεση με την BiCGSTAB μέθοδο.
- Η Gauss-Seidel προρύθμιση, όπως εφαρμόστηκε στο γραμμικό σύστημα, αποτέλεσε μια αποδοτική διαδικασία προρύθμισης.

Η μελέτη της συμπεριφοράς σύγκλισης της προοριθμισμένης επαναληπτικής μεθόδου BiCGSTAB οδήγησε στο συμπέρασμα της μη καταλληλότητας μιας Krylov μεθόδου για την επίλυση μεγάλων προβλημάτων ή ύπαρξης πυκνής διακριτοποίησης. Αυτό συμβαίνει, διότι το μέγεθος του γραμμικού συστήματος που προκύπτει είναι  $8n_x n_y n_z$ , όπου  $n_x, n_y$  και  $n_z$  το πλήθος των πεπερασμένων στοιχείων ανά κατεύθυνση διακριτοποίησης, με συνέπεια τα επαναληπτικά βήματα να αυξάνονται εκθετικά ως προς τη διάσταση του προβλήματος. Επίσης, όλες οι Krylov μέθοδοι υπολογίζουν εσωτερικά γινόμενα διανυσμάτων διάστασης μεγέθους του προβλήματος, με αποτέλεσμα τη δημιουργία πολύ

μεγάλων ή πολύ μικρών αριθμητικών τιμών. Έτσι σε πολύ πυκνές διακριτοποιήσεις εμφανίζονται φαινόμενα overflow αριθμών κινητής υποδιαστολής. Για παράδειγμα, αυτό το φαινόμενο εμφανίστηκε στα προβλήματα δοκιμής πλήθους  $n_s = 512$  πεπερασμένων στοιχείων ανά κατευθυνση. Όλα αυτά μπορούν να αποφευχθούν με την επέκταση της τεχνικής πολυπλέγματος στην τρίτη διάσταση της διακριτοποίησης και αυτό προϋποθέτει την κατασκευή κατάλληλων διδιάστατων τελεστών μετάβασης πλεγμάτων. Οπότε θα υπάρξει ανεξαρτησία των επαναληπτικών βημάτων της τεχνικής πολυπλέγματος ως προς το μέγεθος του προβλήματος, ενώ τα φαινόμενα overflow αριθμών κινητής υποδιαστολής θα περιοριστούν. Αυτό μπορεί να αποτελέσει αντικείμενο εξέλιξης του αριθμητικού σχήματος που παρουσιάστηκε σε αυτή τη διατριβή.



# Βιβλιογραφία

- [1] O. Axelsson and A. Barker, "Finite element solution of boundary value problems".  
*Theory and computation*, Academic Press, Orlando, Fl., 1984.
- [2] G. Birkhoff, M. H. Schultz and R. S. Varga, "Piecewise Hermite in one and two  
Variables with Applications to Partial Differential Equations", *Numer. Math.* 11,  
232-256, 1968.
- [3] C. de Boor and Swartz, "Collocation at Gaussian Points", *SIAM J. Numer. Anal.*  
10, 582-606, 1973.
- [4] A. Brandt, "A guide to multigrid development", in *Multigrid Methods*,  
W.Hackbusch and U. Trottenberg, eds., Springer Verlag, Berlin, 1982, pp. 220-  
312.
- [5] W. L. Briggs, V. E. Henson, and S. F. McCormick, "A Multigrid Tutorial", SIAM,  
Philadelphia, 2000 . Second edition.
- [6] C. C. Christara, "Parallel solvers for spline collocation equations", *Advances in*  
*Eng. Software*, vol. 27, pp.71-89, 1996.
- [7] C. C. Christara and B. Smith, "Multigrid and Multilevel Methods for Quadratic  
Spline Collocation", *BIT*, vol. 37 (4), pp.781-803, 1997.
- [8] cuBLAS Library web page, <https://developer.nvidia.com/cublas>
- [9] CUDA Software web page, <https://developer.nvidia.com/cuda-toolkit>

- [10] W. Hackbusch and U. Trottenberg, "Multigrid Methods", vol. 960, *Lecture Notes in Mathematics*, Springer-Verlag, Berlin, 1982.
- [11] E. N. Houstis, "Application of the method of Collocation on Lines for Solving Nonlinear Hyperbolic Problems", *Math. Comp.* 31, 443 - 456, 1977.
- [12] E. N. Houstis, R. E. Lynch, T. S. Papatheodorou and J. R. Rice, "Evaluation of Numerical Methods for Elliptic Partial Differential Equations", *J Comp. Phys.* 27, 323 - 350, 1978.
- [13] E. N. Houstis, W. Mitchell, J. R. Rice, "Collocation Software for Second Order Elliptic Partial Differential Equations", *ACM Trans. Math. Software* 11, 379-412, 1985.
- [14] C. E. Houstis, E. N. Houstis and J.R. Rice, "Partitioning PDE Computations: Methods and Performance Evaluation", *Parallel Computing*, vol. 5, pp. 141-163, 1997.
- [15] V. G. Mandikas, E. N. Mathioudakis, "A parallel multigrid solver for incompressible flows on computing architectures with accelerators", *The journal of Supercomputing*, DOI: 10.1007/s11227-017-2066-y, Springer, 2017.
- [16] E. N. Mathioudakis, V. G. Mandikas, G. Kozyrakis, N. Kampanis, and J. Ekarinaris, "Multigrid cell-centered techniques for high-order incompressible flow numerical solutions", *Aerospace Science and Technology - AESTE*, vol. 64, pp. 85-101, Elsevier, 2017.
- [17] E. N. Mathioudakis, E. P. Papadopoulou and Y. G. Saridakis, " Mapping Parallel Iterative Algorithms for PDE Computations on a Distributed Memory Computers", *Parallel Algorithms and Applications* , 8, pp. 141-154, 1996.



- [18] E. N. Mathioudakis, E. P. Papadopoulou and Y. G. Saridakis, "Bi-CGSTAB for collocation equations on distributed memory parallel computers" , *Numerical Mathematics and advanced applications - ENUMATH 2001*, pp. 957-966, 2003.
- [19] E. N. Mathioudakis, E. P. Papadopoulou and Y. G. Saridakis, "Iterative Solution of Elliptic Collocation Systems on a Cognitive Parallel Computer", *Computers and Mathematics with applications*, vol. 48, pp. 951-970, 2004.
- [20] E. N. Mathioudakis and E. P. Papadopoulou, "Grid computing for Bi-CGSTAB applied to the solution of modified Helmholtz equation", *Int. J. Applied Maths and comp. sciences*, vol. 4, no. 3, pp. 179-184, 2007.
- [21] NVidia Tesla GPU, <https://www.nvidia.com/en-us/data-center/?ncid=van-datacenter>.
- [22] OpenACC web page, <http://www.openacc.org>.
- [23] OpenMP web page, <http://www.openmp.org>.
- [24] T.S. Papatheodorou, "Inverses for a Class of Banded Matrices and Applications to Piecewise Cubic Approximation", *J. Comp. Appl. Math.* 8 (4), 285-288, 1982.
- [25] PGI software, <http://www.pgroup.com>.
- [26] Y. Saad, "Iterative Methods for sparse linear systems", SIAM, 2003.
- [27] U. Trottenberg, C. Oosterlee, and A. Schüller, "Multigrid", Academic Press, New-York, 2001.
- [28] R.S. Varga, "Matrix Iterative Analysis", *Prentice Hall*, Englewood Cliffs, NJ, 1962
- [29] H. A. van der Vorst, "Bi-CGSTAB : A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems", *SIAM J. Sci. Statist. Comput.*, vol. 13, pp. 631-644, 1992.