



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΧΗΜΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

ΤΟΜΕΑΣ ΙΙ: ΑΝΑΛΥΣΗΣ, ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΑΝΑΠΤΥΞΗΣ ΔΙΕΡΓΑΣΙΩΝ ΚΑΙ  
ΣΥΣΤΗΜΑΤΩΝ

Διπλωματική Εργασία  
του **ΔΙΟΝΥΣΙΟΥ Π. ΜΠΑΛΤΑ**

ΔΙΕΡΕΥΝΗΣΗ ΧΩΡΟΥ ΛΥΣΕΩΝ ΜΗ ΓΡΑΜΜΙΚΟΥ  
ΠΡΟΒΛΗΜΑΤΟΣ ΑΝΤΙΔΡΑΣΗΣ-ΔΙΑΧΥΣΗΣ

Επιβλέπων Καθηγητής  
**ΑΝΔΡΕΑΣ Γ. ΜΠΟΥΝΤΟΥΒΗΣ**

Αθήνα

Φεβρουάριος 2018

# Ευχαριστίες

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε στη σχολή Χημικών Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου, υπό την επίβλεψη του Καθηγητή κ. Ανδρέα Μπουντουβή. Πριν την παρουσίαση της εργασίας, θα ήθελα να ευχαριστήσω όλους όσους συνέβαλαν στην επιτυχή διεκπεραίωση αυτής. Αρχικά, θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον επιβλέποντα Καθηγητή κ. Μπουντουβή για την πρόταση που μου έκανε να αναλάβω ως διπλωματική εργασία, την εκπόνηση ενός ενδιαφέροντος θέματος. Επιπρόσθετα, επιθυμώ να τον ευχαριστήσω ιδιαίτερα για την πολύτιμη βοήθειά του καθ' όλη την διάρκεια της πραγματοποίησης της εργασίας, τις χρήσιμες παρατηρήσεις του και την καθοριστική καθοδήγησή του στην φάση της ολοκλήρωσής της. Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στην οικογένειά μου, και ιδιαίτερα στην μητέρα μου, για την πολύτιμη ηθική και υλική υποστήριξη που μου παρείχαν σε όλη μου την ζωή. Θα ήθελα επίσης να ευχαριστήσω τους φίλους μου, για την βοήθεια τους στις δύσκολες στιγμές των σπουδών μου.

# Περίληψη

Η παρούσα εργασία εστιάζει στην διερεύνηση του χώρου λύσεων προβλήματος συνοριακών τιμών το οποίο περιγράφει ένα σύστημα αντίδρασης-διάχυσης. Μελετάται η πολλαπλότητα των λύσεων και η αλλαγή τους ανάλογα με τις παραμέτρους του προβλήματος. Για την επίλυση χρησιμοποιούνται τρεις κύριες μέθοδοι: η μέθοδος Galerkin/πεπερασμένων στοιχείων για την διακριτοποίηση των μη γραμμικών διαφορικών εξισώσεων, η μέθοδος Newton για την επίλυση των μη γραμμικών αλγεβρικών εξισώσεων που προκύπτουν από αυτές και η μέθοδος παραμετρικού βηματισμού μήκους-τόξου για να ξεπεραστούν οι υπολογιστικές δυσκολίες που δημιουργούν αυτά τα προβλήματα, καθώς εμφανίζουν οριακά σημεία στα οποία η Ιακωβιανή του συστήματος είναι ιδιάζουσα.

# Abstract

This thesis focuses on the exploration of the solution space of a boundary value problem that describes a reaction-diffusion system. We study the multiplicity of solutions as well as their dependence on problem parameters. Three main methods are used to solve the problem: The Galerkin / finite elements method for the discretization of the nonlinear differential equations, the Newton method for solving the nonlinear algebraic equations derived from those and the arc-length parameter continuation method to overcome the computational difficulties that these problems create when limit points appear, at which the Jacobian matrix of the system becomes singular.

# Περιεχόμενα

<b>1</b>	<b>Εισαγωγή/Συστήματα Αντίδρασης-Διάχυσης</b>	<b>6</b>
<b>2</b>	<b>Μεθοδολογία</b>	<b>8</b>
2.1	Μέθοδος Πεπερασμένων Στοιχείων (Finite Element Method) . . .	8
2.1.1	Διακριτοποίηση . . . . .	8
2.1.2	Συναρτήσεις Βάσης . . . . .	9
2.1.3	Ισοπαραμετρική Απεικόνιση . . . . .	9
2.1.4	Υπολογισμός Παραγώγων Συναρτήσεων Βάσης . . . . .	11
2.1.5	Σταθμισμένα Υπόλοιπα Galerkin . . . . .	12
2.1.6	Εισαγωγή Συνοριακών Συνθηκών . . . . .	13
2.1.7	Υπολογισμός Ολοκληρωμάτων/Αριθμητική Ολοκλήρωση	13
2.1.8	Υπολογισμός του Πίνακα των Εξισώσεων Διακριτοποίησης	16
2.2	Μέθοδος Newton . . . . .	16
2.3	Μέθοδος Βηματισμού Μήκους-Τόξου (Arc-Length Continuation)	17
<b>3</b>	<b>Εφαρμογή σε Πρόβλημα Αντίδρασης-Διάχυσης και Αποτελέσματα</b>	<b>21</b>
3.1	Εισαγωγή . . . . .	21
3.2	Διακριτοποίηση . . . . .	21
3.3	Κατασκευή Υπολοίπων Galerkin . . . . .	23
3.4	Συναρτήσεις Βάσης και Αριθμητική Ολοκλήρωση . . . . .	24
3.5	Ροή Αλγορίθμου . . . . .	25
3.6	Παρουσίαση Αποτελεσμάτων . . . . .	26

<b>4 Μελλοντικά Βήματα</b>	<b>31</b>
<b>Βιβλιογραφία</b>	<b>32</b>
<b>Παράρτημα: Κώδικας Fortran</b>	<b>34</b>

# Κεφάλαιο 1

## Εισαγωγή/Συστήματα Αντίδρασης-Διάχυσης

Τα φαινόμενα αντίδρασης-διάχυσης περιγράφονται από μαθηματικά μοντέλα τα οποία αφορούν συνήθως τη χωροχρονική μεταβολή της συγκέντρωσης μίας ή περισσοτέρων χημικών ουσιών. Συμπεριλαμβάνουν δύο κύρια φαινόμενα: χημικές αντιδράσεις που μετατρέπουν την μία ουσία σε άλλη και φαινόμενα διάχυσης που τις κατανέμουν στον χώρο. Εκτός από τη χημεία [9], τα φαινόμενα αντίδρασης-διάχυσης εμφανίζονται στην βιολογία [8], στην γεωλογία [10], στην φυσική [11] και στην οικολογία [12]. Από μαθηματικής σκοπιάς, έχουν την μορφή μη γραμμικών παραβολικών μερικών διαφορικών εξισώσεων. Στην γενική περίπτωση τέτοια συστήματα περιγράφονται από την εξίσωση:

$$\partial_t \mathbf{u} = \underline{\underline{\mathbf{D}}} \nabla^2 \mathbf{u} + \mathbf{R}(\mathbf{u}) \quad (1.1)$$

Όπου:

- $\mathbf{u}(\mathbf{x}, t)$  η διανυσματική συνάρτηση που μας ενδιαφέρει,
- $\underline{\underline{\mathbf{D}}}$  διαγώνιος πίνακας σταθερών διάχυσης,
- $\mathbf{R}(\mathbf{u})$  όρος που συμπεριλαμβάνει όλες τις χημικές αντιδράσεις

Καθώς ο όρος της αντίδρασης είναι συνήθως πολύπλοκος, τα συστήματα αυτά είναι συνήθως μη γραμμικά. Αυτή η εργασία εστιάζει στην λύση του προβλήματος

συνοριακών τιμών (ΠΣΤ) σε δύο διαστάσεις σε τετράγωνο χωρίο  $D$  μοναδιαίου μήκους και πλάτους:

$$\nabla^2 u + \lambda e^{\frac{u}{1+au}} = 0 \quad (1.2)$$

με συνοριακές συνθήκες

$$u = 0 \text{ στο σύνορο } \partial D \quad (1.3)$$

Το ΠΣΤ αποτελεί την εξίσωση που ορίζει την συγκέντρωση  $u$  σε ένα αντιδραστήρα με χημική αντίδραση και διάχυση. Ο πρώτος όρος της εξίσωσης (1.2) αποτελεί τον όρο διάχυσης και ο δεύτερος της χημικής αντίδρασης. Το πρόβλημα αυτό αποτελεί παραλλαγή του προβλήματος Bratu [13]. Θέτοντας  $a=0$  το ΠΣΤ ταυτίζεται με το πρόβλημα αυτό.



# Κεφάλαιο 2

## Μεθοδολογία

### 2.1 Μέθοδος Πεπερασμένων Στοιχείων (Finite Element Method)

#### 2.1.1 Διακριτοποίηση

Στην γενική περίπτωση ένα πρόβλημα συνοριακών τιμών (ΠΣΤ) ορίζεται ως:

$$Lu = f \text{ στο } D$$

$$Bu = g \text{ στο } \partial D$$

Όπου:

$f, g$  συναρτήσεις

$L$  ένας διαφορικός τελεστής

$B$  ένας συνοριακός τελεστής ορισμένος στο σύνορο  $\partial D$  του χωρίου  $D$ .

Τα ΠΣΤ μπορούν να γίνουν περίπλοκα και η επίλυση τους με κλασικές αναλυτικές μεθόδους να μην είναι δυνατή. Για αυτό το λόγο σε αυτήν την εργασία το ΠΣΤ θα λυθεί αριθμητικά, υπολογίζοντας μία οσοδήποτε ακριβή επιθυμούμε προσέγγιση της τιμής της συνάρτησης σε πεπερασμένο αριθμό σημείων του χώρου

με ελεγχόμενη ακρίβεια. Αυτό γίνεται εφικτό με την διαδικασία που περιγράφεται παρακάτω.

Το χωρίο  $D$  υποδιαιρείται σε επάλληλα υποχωρία (subdomains) με το “άπλωμα” ενός πλέγματος (mesh). Κάθε υποχωρίο ονομάζεται στοιχείο (element). Στα όρια και στο εσωτερικό των στοιχείων ορίζονται οι κόμβοι (nodes) του πλέγματος. Το πλήθος των στοιχείων συμβολίζεται με  $NE$  και το πλήθος των κόμβων με  $N$ .

Η λύση  $u$  προσεγγίζεται από το άθροισμα συναρτήσεων  $\varphi^j$ , ίσου με τον αριθμό των κόμβων  $N$ , πολλαπλασιασμένων με συντελεστές  $u_j$ .

$$u(x) = \sum_{j=1} u_j \varphi^j(\underline{x}) \quad (2.1)$$

Ο συντελεστής  $u_j$ , όπως θα δούμε παρακάτω, είναι η (προσεγγιστική) τιμή της λύσης  $u$  στον κόμβο  $j$ .

### 2.1.2 Συναρτήσεις Βάσης

Οι συναρτήσεις  $\varphi^j$  ονομάζονται συναρτήσεις βάσης και αποτελούν αυθαίρετες συναρτήσεις του διανύσματος θέσης. Οι συναρτήσεις είναι ίσες στον αριθμό με τους κόμβους  $N$  και κάθε συνάρτηση είναι μονάδα στον κόμβο που αντιστοιχεί:

$$\begin{aligned} \varphi^j(\underline{x}_k) &= 1 \text{ εάν } j = k \\ \varphi^j(\underline{x}_k) &= 0 \text{ εάν } j \neq k \end{aligned} \quad (2.2)$$

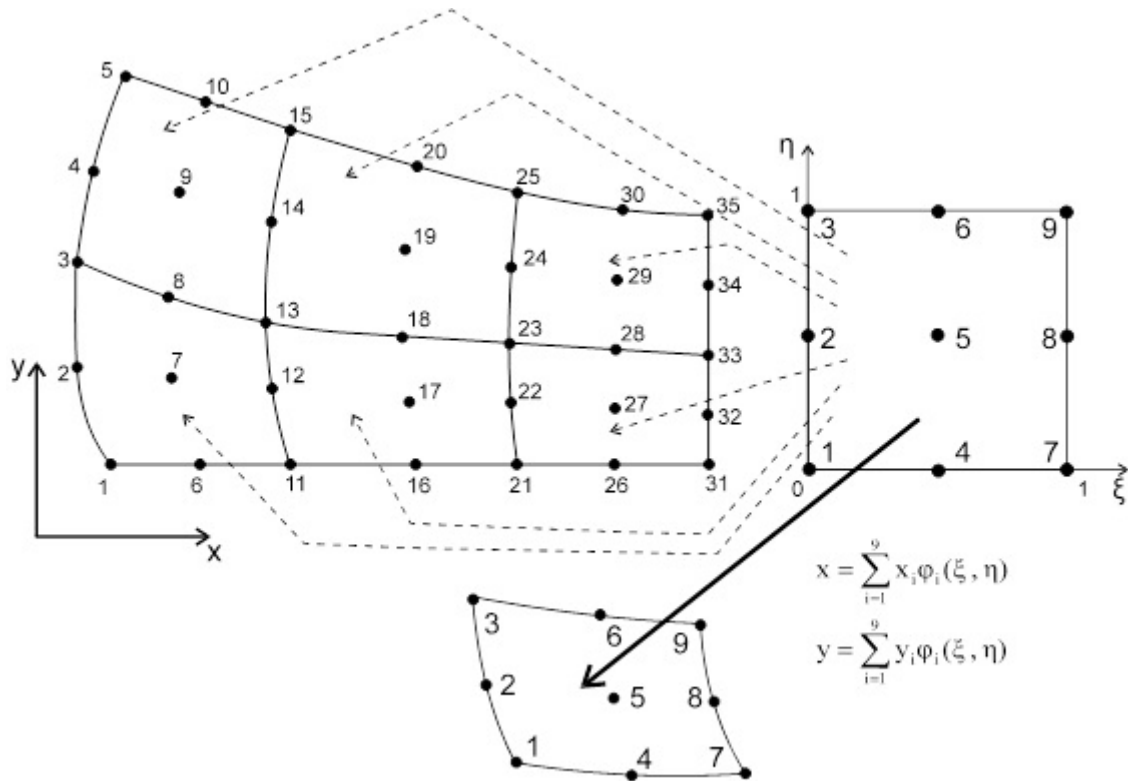
Επίσης κάθε συνάρτηση είναι μηδενική στα στοιχεία στα οποία δεν ανήκει ο κόμβος της.

Οι συναρτήσεις βάσης που χρησιμοποιούνται συνήθως είναι πολώνυμα 1ου ή 2ου βαθμού λόγω της απλότητάς τους.

### 2.1.3 Ισοπαραμετρική Απεικόνιση

Οι συναρτήσεις βάσης, όπως ορίστηκαν στην εξίσωση (2.1), είναι συναρτήσεις του διανύσματος θέσης  $\underline{x}_i$ . Ο υπολογισμός τους όμως είναι περίπλοκος σε τυχαίο

στοιχείο ενός πλέγματος. Μπορούμε όμως να τις ορίσουμε σε μοναδιαίο στοιχείο αναφοράς και να χρησιμοποιήσουμε ένα μετασχηματισμό που καλείται ισοπαραμετρική απεικόνιση (Σχήμα (2.1)). Το στοιχείο αναφοράς χρησιμοποιεί σύστημα συντεταγμένων  $\underline{\xi}$ . Η μετατροπή από το σύστημα  $\underline{x}$  στο σύστημα  $\underline{\xi}$  γίνεται εύκολα από υπολογιστική σκοπιά σύμφωνα με την παρακάτω εξίσωση:



Σχήμα 2.1: Ισοπαραμετρική απεικόνιση σε δύο διαστάσεις

$$\underline{x} = \sum_{j=1}^{n_k} \underline{x}_j \varphi_j(\underline{\xi}) \quad (2.3)$$

Όπου:

$\underline{x}$  το διάνυσμα θέσης στο στοιχείο του πλέγματος,

$\underline{x}_i$  το διάνυσμα θέσης του κόμβου  $i$ , σε τοπική αρίθμηση, του στοιχείου του πλέγματος

$\underline{\xi}$  το διάνυσμα θέσης στο στοιχείο αναφοράς

$n_k$  το πλήθος των κόμβων στο στοιχείο αναφοράς

$\varphi_j$  η συνάρτηση βάσης που αντιστοιχεί στον κόμβο  $j$

Ένα δείγμα συναρτήσεων βάσης αναφέρεται στο εδάφιο (3.4). Η αντίστροφη απεικόνιση από το  $\underline{x}$  στο  $\underline{\xi}$  :

$$\underline{\xi} \equiv \underline{\xi}(\underline{x})$$

υπολογίζεται με λύση των εξισώσεων (2.3), αν και στην πράξη δεν προκύπτει να είναι απαραίτητο.

## 2.1.4 Υπολογισμός Παραγώγων Συναρτήσεων Βάσης

Η ιακωβιανή του μετασχηματισμού ορίζεται ως:

$$\underline{J}_{=0} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \quad (2.4)$$

Σύμφωνα με τον κανόνα της αλυσίδας:

$$\frac{\partial \varphi_i}{\partial \xi} = \frac{\partial \varphi_i}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial \varphi_i}{\partial y} \frac{\partial y}{\partial \xi} \quad (2.5)$$

$$\frac{\partial \varphi_i}{\partial \eta} = \frac{\partial \varphi_i}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial \varphi_i}{\partial y} \frac{\partial y}{\partial \eta}$$

ή σε μορφή πινάκων:

$$\underline{J}_{=0} \begin{bmatrix} \frac{\partial \varphi_i}{\partial x} \\ \frac{\partial \varphi_i}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \varphi_i}{\partial \xi} \\ \frac{\partial \varphi_i}{\partial \eta} \end{bmatrix} \quad (2.6)$$

Λύνοντας το σύστημα (2.6) βρίσκουμε τις μερικές παραγώγους των συναρτήσεων ως προς  $x$  και  $y$ :

$$\frac{\partial \varphi_i}{\partial x} = \frac{1}{\det \underline{J}_{=0}} \left( \frac{\partial y}{\partial \eta} \frac{\partial \varphi_i}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial \varphi_i}{\partial \eta} \right) \quad (2.7)$$

$$\frac{\partial \varphi_i}{\partial y} = \frac{1}{\det \underline{J}_{=0}} \left( \frac{\partial x}{\partial \xi} \frac{\partial \varphi_i}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial \varphi_i}{\partial \xi} \right)$$

Όπου  $\det \underline{J}_{=0}$  η ορίζουσα της Ιακωβιανής:

$$\det \underline{J}_{=0} = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \quad (2.8)$$

Τα στοιχεία της Ιακωβιανής υπολογίζονται απευθείας από την εξίσωση (2.3):

$$\frac{\partial x}{\partial \xi} = \sum_{i=1}^{n_k} x_i \frac{\partial \varphi_i(\xi, \eta)}{\partial \xi}, \quad \frac{\partial x}{\partial \eta} = \sum_{i=1}^{n_k} x_i \frac{\partial \varphi_i(\xi, \eta)}{\partial \eta} \quad (2.9)$$

$$\frac{\partial y}{\partial \xi} = \sum_{i=1}^{n_k} y_i \frac{\partial \varphi_i(\xi, \eta)}{\partial \xi}, \quad \frac{\partial y}{\partial \eta} = \sum_{i=1}^{n_k} y_i \frac{\partial \varphi_i(\xi, \eta)}{\partial \eta}$$

Οι συναρτήσεις βάσης ορίζονται στο  $\underline{\xi}$ -σύστημα και είναι συνήθως απλής μορφής. Συνεπώς εύκολα υπολογίζουμε αναλυτικά και τις παραγώγους τους  $\frac{\partial \varphi_i}{\partial \xi}$  και  $\frac{\partial \varphi_i}{\partial \eta}$  ώστε να τις χρησιμοποιήσουμε στις προηγούμενες εξισώσεις.

## 2.1.5 Σταθμισμένα Υπόλοιπα Galerkin

Κάθε προσεγγιστική λύση  $u$  (εξίσωση 2.1) του ΠΣΤ δεν θα ικανοποιεί ακριβώς σε κάθε σημείο την διαφορική εξίσωση. Το υπόλοιπο (residual) της θα είναι  $Lu - f$ . Είναι προφανές ότι χρειάζεται κάποια μέθοδος για τον υπολογισμό των παραγόντων  $u_j$  οι οποίοι θα ελαχιστοποιούν το υπόλοιπο αυτό. Η μέθοδος Galerkin αναζητεί λύση που μηδενίζει τα σταθμισμένα με τις συναρτήσεις βάσης υπόλοιπα της ΔΕ (Galerkin weighted residuals):

$$R_i \equiv \int_D \varphi^i (Lu - f) dS \text{ και } R_i = 0, i = 1, 2, \dots, N \quad (2.10)$$

Με αντικατάσταση της  $u$  από την (2.1) προκύπτει

$$R_i \equiv \int_D \varphi^i (L(\sum_{j=1}^N u_j \varphi^j) - f) dS \quad (2.11)$$

Η παραπάνω εξίσωση αποτελεί συνάρτηση μόνο των κομβικών αγνώστων  $u_j$ . Επομένως έχουμε το σύστημα εξισώσεων:

$$\underline{R}(u) = \underline{0} \quad (2.12)$$

Στην περίπτωση που η διαφορική εξίσωση είναι γραμμική, το παραπάνω σύστημα είναι γραμμικό και εύκολο να λυθεί αν μπορούμε να υπολογίσουμε όλα τα ολοκληρώματά του, κατασκευάζοντας τους πίνακες και λύνοντάς το, για παράδειγμα, με απαλοιφή Gauss. Εάν το σύστημα είναι μη γραμμικό, όπως σε αυτήν την εργασία, λύνεται με την μέθοδο Newton, κάνοντας μία αρχική εκτίμηση της λύσης και βελτιώνοντάς την επαναληπτικά, λύνοντας ένα γραμμικό σύστημα κάθε φορά, όπως θα αναλυθεί παρακάτω.

### 2.1.6 Εισαγωγή Συνοριακών Συνθηκών

Τα ΠΣΤ μπορούν να έχουν δύο είδη συνοριακών συνθηκών:

Το πρώτο είδος είναι οι συνθήκες Dirichlet (ή essential) οι οποίες ορίζουν την τιμή της συνάρτησης σε τμήμα του συνόρου. Αυτές αντικαθιστούν τα υπόλοιπα Galerkin στους επηρεαζόμενους κόμβους. Εάν στον συνοριακό κόμβο  $N$ , στην θέση  $\underline{t}_n$  η λύση ορίζεται ως  $u = F(\underline{t})$ , τότε το υπόλοιπο που αντιστοιχεί στον κόμβο θα πρέπει να γίνει:

$$R_n \equiv u_n - F(\underline{t}_n) = 0 \quad (2.13)$$

Το δεύτερο είδος ονομάζεται συνθήκη Neumann (ή natural) και ορίζει την τιμή παραγώγου της λύσης σε τμήμα του συνόρου. Αυτές εισάγονται απευθείας στην εξίσωση των υπολοίπων Galerkin κατά περίπτωση, εφαρμόζοντας διάφορες μαθηματικές τεχνικές οι οποίες δεν θα αναλυθούν εδώ.

### 2.1.7 Υπολογισμός Ολοκληρωμάτων/Αριθμητική Ολοκλήρωση

Ένα τυπικό είδος ολοκλήρωματος που εμφανίζεται στις εξισώσεις διακριτοποίησης είναι

$$I_{ij} \equiv \int_D \varphi^i L \varphi^j d\underline{x} \quad (2.14)$$

Το ολοκλήρωμα  $I_{ij}$  αποτελεί άθροισμα NE (αριθμός στοιχείων) ολοκληρωμάτων:

$$I_{ij} = \sum_{k=1}^{NE} I_{ij}^k \quad (2.15)$$

$$I_{ij}^k \equiv \int_{E_k} \varphi_i L \varphi_j d\underline{x}$$

Όπου  $E_k (k = 1, 2, \dots, NE)$  στοιχείο του πλέγματος. Οι συναρτήσεις βάσης με πάνω δείκτη αναφέρονται σε “ολικό” συμβολισμό και αυτές με κάτω δείκτη σε “τοπικό” συμβολισμό.

Καθώς όπως είδαμε νωρίτερα οι συναρτήσεις βάσης έχουν οριστεί “τοπικά”, τα ολοκληρώματα  $I_{ij}^k$  υπολογίζονται μέσω της ισοπαραμετρικής απεικόνισης με εφαρμογή μετασχηματισμού ολοκληρώματος:

$$I_{ij}^k = \int_{E_0} \varphi_i(\underline{\xi}) L_{\underline{\xi}} \varphi_j(\underline{\xi}) \det \underline{J}_{\underline{0}} d\underline{\xi} \quad (2.16)$$

Όπου  $E_0$  το στοιχείο αναφοράς

$L_{\underline{\xi}}$  ο διαφορικός τελεστής εκφρασμένος στο  $\underline{\xi}$ -σύστημα

$\det \underline{J}_{\underline{0}}$  η ορίζουσα της Ιακωβιανής της ισοπαραμετρικής απεικόνισης του  $E_0$  στο  $E_k$  (εξίσωση 2.8) ([2] σελ.270-277).

Ένα δεύτερο είδος ολοκληρώματος που εμφανίζεται είναι συνοριακό:

$$I_{bi} = \oint \varphi^i g(t) dt \quad (2.17)$$

Όπως και πριν, το ολοκλήρωμα αποτελεί άθροισμα ολοκληρωμάτων επί των επιμέρους στοιχείων:

$$I_{bi} = \sum_k I_{bi}^k \quad (2.18)$$

$$I_{bi}^k \equiv \int_{\partial E_k \cap \partial D} \varphi_i g(t) dt$$

Ο δείκτης  $k$  στην (2.18) σαρώνει μόνο τα συνοριακά στοιχεία του πλέγματος δηλαδή όλα τα στοιχεία  $E_k$  τα οποία έχουν κοινά σημεία με το  $\partial D$ . Με  $\partial E_k$  συμβο-

λίζεται το σύνορο του στοιχείου  $E_k$ . Όπως και πριν, χρησιμοποιούμε την ισοπαραμετρική απεικόνιση:

$$I_{bi}^k = \int_{\partial E_{0k}} \varphi_i(\underline{\xi}(\lambda))g(t(\underline{\xi}(\lambda)))det\underline{J}_{\underline{0}}d\lambda \quad (2.19)$$

Το  $\partial E_{0k}$  αποτελεί το τμήμα του συνόρου του στοιχείου αναφοράς που απεικονίζεται στο  $\partial E_k \cap \partial D$ . Η μεταβλητή ολοκλήρωσης  $\lambda$  μεταβάλλεται πάνω στο  $\partial E_{0k}$ . Οι ποσότητες στο ολοκλήρωμα ( $\varphi_i, g(t), det\underline{J}_{\underline{0}}$ ) υπολογίζονται πάνω στο  $\partial E_{0k}$  (δηλαδή για  $\underline{\xi} = \underline{\xi}(\lambda)$ ).

Μετά από αυτή τη διαδικασία τα ολοκληρώματα υπολογίζονται αριθμητικά στο στοιχείο αναφοράς χρησιμοποιώντας την μέθοδο Gauss (Gauss Quadrature) ([3] κεφ.3 ,[4] σελ. 195-202).

$$\int_{E_0} f(\underline{\xi})d\underline{\xi} = \sum_{k=1}^{NGP} w_k f(\underline{\xi}_{gk}) \quad (2.20)$$

όπου:

- $E_0$  το στοιχείο αναφοράς
- $w_k$  είναι συντελεστές βάρους, τα λεγόμενα Gauss weights
- $\underline{\xi}_{gk}$  είναι σημεία μέσα στο στοιχείο αναφοράς στα οποία υπολογίζεται η προς ολοκλήρωση συνάρτηση, τα λεγόμενα σημεία Gauss (Gauss points)
- NGP είναι το πλήθος των σημείων Gauss στο στοιχείο αναφοράς.

Για δεδομένη συνάρτηση  $f(\underline{\xi})$  στην (2.20) η ακρίβεια της ολοκλήρωσης εξαρτάται από την επιλογή των σημείων Gauss και των συντελεστών βάρους. Για τις κοινά χρησιμοποιούμενες συναρτήσεις βάσης, οι οποίες είναι πολυώνυμα μικρού βαθμού, μπορούμε να έχουμε έως και 100% ακρίβεια.



### 2.1.8 Υπολογισμός του Πίνακα των Εξισώσεων Διακριτοποίησης

Όπως είδαμε νωρίτερα, τα ολοκληρώματα που εμφανίζονται στο υπόλοιπα Galerkin αποτελούν αθροίσματα ολοκληρωμάτων επί των στοιχείων του πλέγματος. Για αυτό τον λόγο, μπορούμε στους υπολογισμούς μας να σαρώνουμε τα στοιχεία και να υπολογίζουμε τις επιμέρους συνεισφορές τους σε κάθε ολοκλήρωμα.

## 2.2 Μέθοδος Newton

Εάν το ΠΣΤ που προσπαθούμε να επιλύσουμε είναι μη γραμμικό, το σύστημα των εξισώσεων που προκύπτει από την μέθοδο Galerkin θα είναι και αυτό μη γραμμικό. Για την επίλυση του συστήματος μη γραμμικών εξισώσεων

$$\underline{R}(\underline{u}) = 0 \quad (2.21)$$

χρησιμοποιείται η μέθοδος Newton (γνωστή και ως μέθοδος Newton-Raphson, [3] σελ. 197-214,[5],[6]). Η μέθοδος Newton αποτελεί επαναληπτική μέθοδο: με δεδομένη μια αρχική εκτίμηση  $\underline{u}^{(0)}$  της λύσης  $\underline{u}$ , η  $(k+1)$ -προσέγγιση  $\underline{u}^{(k+1)}$  προκύπτει από την  $k$ -προσέγγιση,  $\underline{u}^{(k)}$ , λύνοντας το σύστημα:

$$\underline{\underline{J}}^{(k)} \underline{\underline{\delta}}^{(k+1)} = -\underline{\underline{R}}^{(k)}$$

$$\text{όπου} \quad (2.22)$$

$$\underline{u}^{(k+1)} = \underline{u}^{(k)} + \underline{\delta}^{(k+1)}$$

Ο πίνακας  $\underline{\underline{J}}$  ονομάζεται Ιακωβιανή του συστήματος (2.21) και τα στοιχεία του είναι οι μερικές παραγώγοι των υπολοίπων Galerkin ως προς τους κομβικούς αγνώστους:

$$J_{ij} = \frac{\partial R_i}{\partial u_j} \quad (2.23)$$

Η διαδικασία επαναλαμβάνεται έως ότου η διαφορά του μέτρου μεταξύ δύο διαδοχικών προσεγγίσεων να γίνει αρκούντως μικρή:

$$\|\underline{\delta}^{(k+1)}\| = \sqrt{\sum_{i=1}^N (\underline{u}_i^{(k+1)} - \underline{u}_i^{(k)})^2} < \epsilon \quad (2.24)$$

Όπου  $\epsilon$  μία προκαθορισμένη μικρή τιμή (π.χ.  $\epsilon = 0.0001$ ). Όσο μικρότερο είναι το  $\epsilon$  τόσο μεγαλύτερη ακρίβεια θα έχει η λύση, αλλά παράλληλα θα αυξάνεται και το υπολογιστικό κόστος. Συνεπώς, επιλέγεται η μεγαλύτερη δυνατή τιμή που μας δίνει επαρκή ακρίβεια για τις ανάγκες μας.

## 2.3 Μέθοδος Βηματισμού Μήκους-Τόξου (Arc-Length Continuation)

Πολλά προβλήματα συνοριακών τιμών που παρουσιάζουν υπολογιστικό ενδιαφέρον είναι της μορφής:

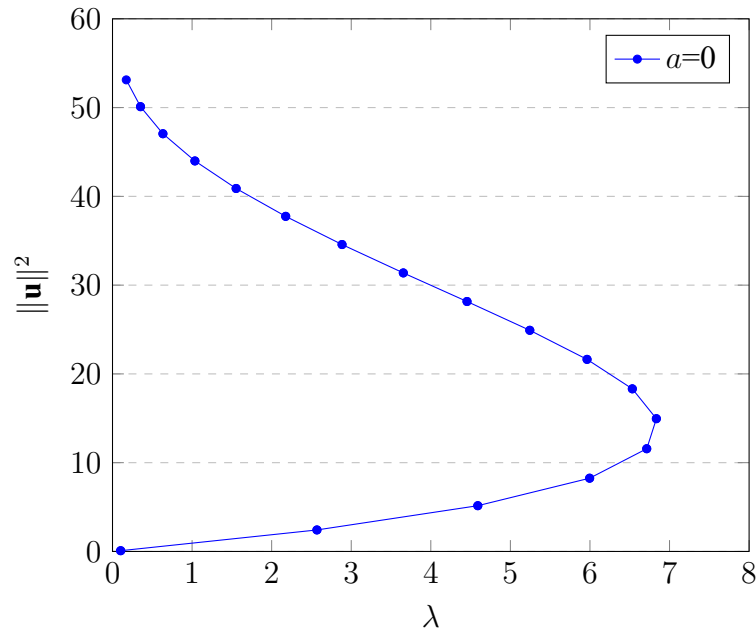
$$F(\underline{u}, \underline{\lambda}) = 0 \quad (2.25)$$

Όπου:

- $F$  είναι μία μη γραμμική συνάρτηση
- $\underline{u}$  ένα πεδίο λύσεων (π.χ. πεδίο ροών, συγκεντρώσεων, κ.α.)
- $\underline{\lambda}$  διάνυσμα παραμέτρων (αριθμοί Reynolds, κ.α.)

Σε αυτά τα προβλήματα είναι χρήσιμο να μπορούμε να υπολογίσουμε την μεταβολή της λύσης  $\underline{u}$  ως προς  $\underline{\lambda}$ , δηλαδή να υπολογίσουμε “μονοπάτια” λύσεων  $(\underline{u}(\lambda), \lambda)$ .

Μία ιδιαιτερότητα των μονοπατιών αυτών των προβλημάτων είναι η συχνή εμφάνιση οριακών σημείων-σημείων στροφής (limit points ή επίσης turning points ή folds) (ενδεικτικά στο Σχήμα (2.2) για  $\lambda = 7$ ). Ένα οριακό σημείο [7]  $P^0 = (\underline{u}^0, \underline{\lambda}^0)$  ορίζεται ως μία λύση της εξίσωσης (2.25) για την οποία η παράγωγος *Fréchet* ικανοποιεί:



Σχήμα 2.2: Χώρος Λύσεων για  $a = 0$

α)  $F_u^0$  έχει ορίζουσα 0

β)  $F_\lambda^0 \notin R(F_u^0)$ .

Το σημείο καλείται απλό αν επίσης:

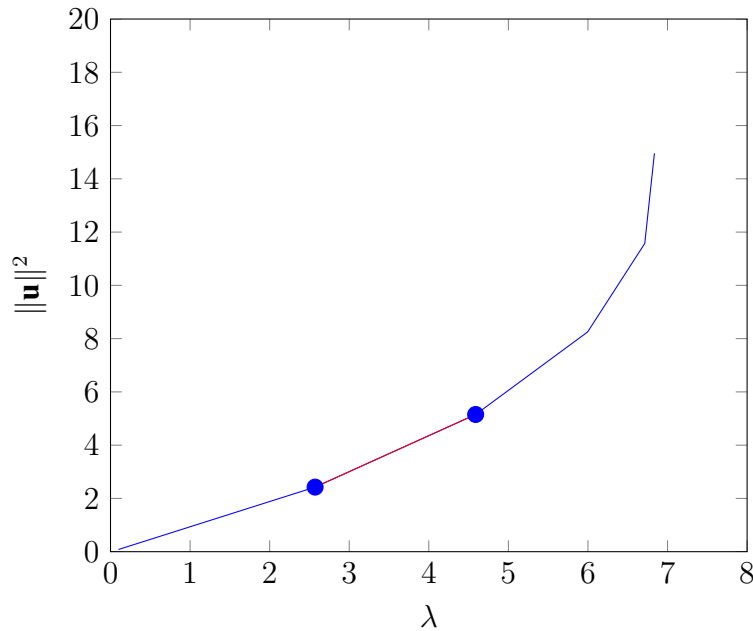
γ)  $\dim N(F_u^0) = \text{codim} R(F_u^0) = 1$

Εδώ το  $N$  και το  $R$  σημαίνουν null space και range αντίστοιχα. Οι δείκτες  $u$  και  $\lambda$  δείχνουν παραγωγή ως προς το  $u$  και  $\lambda$  αντίστοιχα.

Τα προβλήματα τα οποία εμφανίζουν αυτά τα σημεία έχουν την δυνατότητα για μη μοναδικές λύσεις, δηλαδή αντιστοιχία παραπάνω από μία λύσεων  $\underline{u}$  στο ίδιο  $\underline{\lambda}$ . Ο βηματισμός σε παράμετρο, όμως, με τη μέθοδο Netwon, αδυνατεί να προχωρήσει κοντά σε σημεία στροφής διότι εκεί η Ιακωβιανή γίνεται ιδιάζουσα. Αυτή η ιδιάζουσα κατάσταση γίνεται εύκολα αντιληπτή λόγω και του ότι η εφαπτομένη της καμπύλης της λύσης ως προς την παράμετρο απειρίζεται στο σημείο στροφής.

Η μέθοδος που χρησιμοποιούμε για να ξεπεράσουμε αυτήν την δυσκολία είναι ο βηματισμός μήκους-τόξου (arc-length continuation) [7]. Σε αυτήν την μέθοδο, οι  $\underline{u}$  και  $\underline{\lambda}$  εκφράζονται ως συναρτήσεις μίας παραμέτρου  $s$  που εκφράζει το μήκος του

τόξου της καμπύλης της λύσης, η οποία είναι μονότονα αύξουσα, σε αντιδιαστολή με τη μεταβολή της λύσης ως προς την παράμετρο, όπου η μονοτονία αλλάζει εκατέρωθεν του σημείου στροφής (Σχήμα 2.2). Για δύο διαδοχικές κοντινές λύσεις σε γράφημα ( $\|\underline{u}\|-\lambda$ ) η μεταβολή της  $s$  είναι το μήκος του ευθύγραμμου τμήματος που τις ενώνει (Σχήμα 2.3).



Σχήμα 2.3: Το μήκος της κόκκινης γραμμής είναι ίσο με την μεταβολή της παραμέτρου  $s$

Η  $s$  (μήκος τόκος) υπακούει στην εξίσωση (Πυθαγόρειο Θεώρημα):

$$N \equiv \left\| \frac{\partial \underline{u}}{\partial s} \right\|^2 + \left| \frac{\partial \lambda}{\partial s} \right|^2 - 1 = 0 \quad (2.26)$$

Ή, σε γραμμικοποιημένη μορφή:

$$N(\underline{u}(s), \lambda(s), s) \equiv \sum_i \left( \frac{\partial \underline{u}_0}{\partial s} (\underline{u}_i(s) - \underline{u}(s_0)) \right) + \frac{\partial \lambda_0}{\partial s} (\lambda(s) - \lambda(s_0)) - (s - s_0) = 0 \quad (2.27)$$

Για λόγους απλούστευσης της παρουσίασης της μεθόδου, θεωρούμε ότι η παράμετρος  $\lambda$  που μας ενδιαφέρει είναι μία. Αφού κατασκευάσουμε το σύστημα με τις εξισώσεις (2.25) και (2.27) εφαρμόζουμε την μέθοδο Newton για δεδομένο  $\Delta s$

το οποίο επιλέγουμε. Προκύπτει το επαυξημένο (augmented) σύστημα (N+1) εξισώσεων με (N+1) αγνώστους:

$$\begin{bmatrix} F_u & F_\lambda \\ N_u & N_\lambda \end{bmatrix} \begin{bmatrix} \delta \underline{u}^i \\ \delta \underline{\lambda}^i \end{bmatrix} = - \begin{bmatrix} F \\ N \end{bmatrix} \quad (2.28)$$

Όπου οι δείκτες  $u$  και  $\lambda$  δείχνουν την αντίστοιχη διαφόριση.

Για κάθε βήμα της μεθόδου ο παραπάνω πίνακας χρησιμοποιεί μία εκτίμηση της λύσης για να υπολογίσει την επόμενη εκτίμηση, έως ότου η επαναληπτική διαδικασία Newton στο επαυξημένο σύστημα (2.28) να συγκλίνει, όπως είδαμε στην παράγραφο (2.2).

Σημειώνεται ότι ο επαυξημένος πίνακας του συστήματος (2.27) είναι κανονικός (δηλ. μη ιδιάζων) [5] και συνεπώς η μέθοδος Newton δεν αντιμετωπίζει προβλήματα σύγκλισης.

## Κεφάλαιο 3

# Εφαρμογή σε Πρόβλημα Αντίδρασης-Διάχυσης και Αποτελέσματα

### 3.1 Εισαγωγή

Σε αυτό το κεφάλαιο εφαρμόζεται η παραπάνω μεθοδολογία σε πρόβλημα αντίδρασης-διάχυσης. Το πρόβλημα θα λυθεί σε τετράγωνο χωρίο  $D$ , με πλευρά μοναδιαίου μήκους. Ένα σύστημα αντίδρασης-διάχυσης μπορεί να εκφραστεί ως:

$$\nabla^2 u + \lambda e^{\frac{u}{1+au}} = 0 \quad (3.1)$$

με συνοριακές συνθήκες

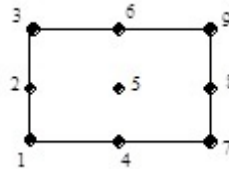
$$u = 0 \text{ στο σύνορο } \partial D \quad (3.2)$$

### 3.2 Διακριτοποίηση

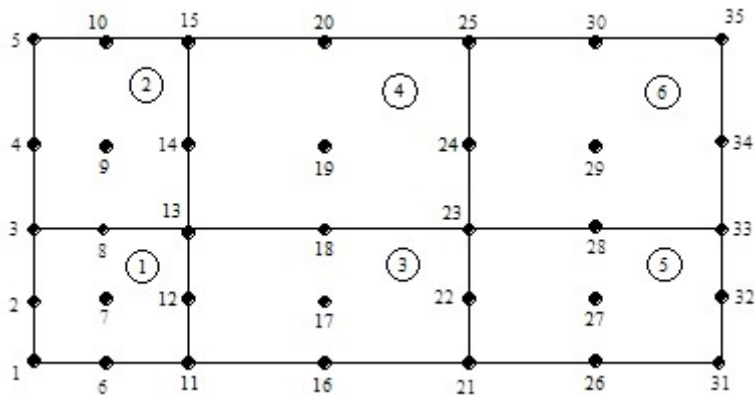
Στην εργασία αυτή χρησιμοποιήσαμε τετράγωνα στοιχεία με εννέα κόμβους το καθένα, οχτώ στην περιφέρεια και έναν στο κέντρο, όπως φαίνεται στο παρακάτω

σχήμα. Τα στοιχεία και οι κόμβοι αριθμούνται σε αυτή την εργασία από κάτω προς τα πάνω και από αριστερά προς τα δεξιά. Υπάρχουν δύο είδη αρίθμησης κάθε κόμβου, ένας τοπικός αριθμός που αναφέρεται στους κόμβους κάθε στοιχείου (Σχήμα 3.1) και ένας ολικός αριθμός που αναφέρεται σε όλους τους κόμβους του χωρίου (Σχήμα 3.2). Ως παράδειγμα, ο “τοπικός” κόμβος 3 του στοιχείου 2 αντιστοιχεί στον “ολικό” κόμβο 5.

Στο πρόβλημά μας επιλέξαμε 10 στοιχεία στην  $x$ -διεύθυνση και 10 στην  $y$ -διεύθυνση, συνολικά 100. Κάθε στοιχείο έχει 9 κόμβους, συνολικά για τα 100 στοιχεία έχουμε 441 κόμβους, καθώς κάποιοι είναι κοινοί.



Σχήμα 3.1: Τοπική Αρίθμηση Κόμβων Στοιχείου



Σχήμα 3.2: Ολική Αρίθμηση Κόμβων και Στοιχείων. Οι αριθμοί μέσα σε  $\bigcirc$  αναφέρονται σε στοιχεία.

### 3.3 Κατασκευή Υπολοίπων Galerkin

Οι τελεστές για αυτό το πρόβλημα, όπως είδαμε στην παράγραφο 2.1.1 είναι:

$$L = \nabla^2 + \lambda e^{\frac{u}{1+au}}, f = 0 \quad (3.3)$$

$$B = 1, g = 0 \text{ στο } \partial D$$

Όπου:

$\mathbb{1}$  ο ταυτοτικός τελεστής ( $\mathbb{1}u \equiv u$ )

$\partial D$  το σύνορο του χωρίου.

Γίνεται κατασκευή των υπολοίπων Galerkin:

$$R_i \equiv \iint \varphi^i (Lu - f) dx dy = 0 \quad (3.4)$$

Με αντικατάσταση από την εξίσωση (3.3):

$$R_i = \iint \varphi^i \nabla^2 u dx dy + \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \quad (3.5)$$

Με κατά παράγοντες ολοκλήρωση στο πρώτο κομμάτι προκύπτει:

$$R_i = \iint \nabla(\varphi^i \nabla u) dx dy - \iint \nabla \varphi^i \nabla u dx dy + \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \quad (3.6)$$

Αναλύοντας τον διαφορικό τελεστή  $\nabla$ :

$$\begin{aligned} R_i &= \iint \left( \frac{\partial}{\partial x} (\varphi^i \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y} (\varphi^i \frac{\partial u}{\partial y}) \right) dx dy \\ &\quad - \iint \frac{\partial \varphi^i}{\partial x} \frac{\partial u}{\partial x} dx dy - \iint \frac{\partial \varphi^i}{\partial y} \frac{\partial u}{\partial y} dx dy \\ &\quad + \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \end{aligned} \quad (3.7)$$

Εφαρμόζουμε το Θεώρημα Green στο πρώτο ολοκλήρωμα:

$$\begin{aligned} R_i &= \oint \varphi^i \left( \frac{\partial u}{\partial x} dy - \frac{\partial u}{\partial y} dx \right) \\ &\quad - \iint \frac{\partial \varphi^i}{\partial x} \frac{\partial u}{\partial x} dx dy - \iint \frac{\partial \varphi^i}{\partial y} \frac{\partial u}{\partial y} dx dy \\ &\quad + \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \end{aligned} \quad (3.8)$$



Εφόσον το ΠΣΤ έχει συνθήκη Dirichlet, όλοι οι όλα τα υπόλοιπα στους κόμβους του συνόρου οι οποίοι θα περιέχουν το συνοριακό ολοκλήρωμα δεν θα χρειαστεί να υπολογισθούν. Συνεπώς η τελική μορφή των σταθμισμένων υπολοίπων είναι:

$$R_i = - \iint \frac{\partial \varphi^i}{\partial x} \frac{\partial u}{\partial x} dx dy - \iint \frac{\partial \varphi^i}{\partial y} \frac{\partial u}{\partial y} dx dy + \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \quad (3.9)$$

Υπολογίζουμε την παράγωγο  $\frac{\partial R_i}{\partial \lambda}$  και την παράγωγο  $\frac{\partial R_i}{\partial u_j}$  οι οποίες θα μας χρειαστούν στην εφαρμογή της μεθόδου μήκους-τόξου:

$$\frac{\partial R_i}{\partial \lambda} = \iint \varphi^i \lambda e^{\frac{u}{1+au}} dx dy \quad (3.10)$$

(Υπενθυμίζεται ότι  $u = \sum_j u_j \varphi^j$  συνεπώς  $\frac{\partial u}{\partial u_j} = \varphi^j$ .)

$$\begin{aligned} \frac{\partial R_i}{\partial u_j} = & - \iint \frac{\partial \varphi^i}{\partial x} \frac{\partial \varphi^j}{\partial x} dx dy - \iint \frac{\partial \varphi^i}{\partial y} \frac{\partial \varphi^j}{\partial y} dx dy \\ & + \iint \varphi^i \varphi^j \lambda \frac{e^{\frac{u}{1+au}}}{(1+au)^2} dx dy \end{aligned} \quad (3.11)$$

### 3.4 Συναρτήσεις Βάσης και Αριθμητική Ολοκλήρωση

Οι συναρτήσεις βάσεις που χρησιμοποιούμε είναι πολυώνυμα 2ου βαθμού:

$$\begin{aligned} \varphi_1 &= (1 - 3\xi + 2\xi^2)(1 - 3\eta + 2\eta^2) \\ \varphi_2 &= (1 - 3\xi + 2\xi^2)(4\eta - 2\eta^2) \\ \varphi_3 &= (1 - 3\xi + 2\xi^2)(-\eta + 2\eta^2) \\ \varphi_4 &= (4\xi - 4\xi^2)(1 - 3\eta + 2\eta^2) \\ \varphi_5 &= (4\xi - 4\xi^2)(4\eta - 2\eta^2) \\ \varphi_6 &= (4\xi - 4\xi^2)(-\eta + 2\eta^2) \\ \varphi_7 &= (-\xi + 2\xi^2)(1 - 3\eta + 2\eta^2) \\ \varphi_8 &= (-\xi + 2\xi^2)(4\eta - 2\eta^2) \\ \varphi_9 &= (-\xi + 2\xi^2)(-\eta + 2\eta^2) \end{aligned} \quad (3.12)$$

Όπου  $\xi$  και  $\eta$  οι χωρικές μεταβλητές  $0 \leq \xi \leq 1$  και  $0 \leq \eta \leq 1$  στο στοιχείο αναφοράς.

Για την αριθμητική ολοκλήρωση αυτών των συναρτήσεων χρησιμοποιούμε 9 σημεία Gauss στο στοιχείο αναφοράς:

Πίνακας 3.1: Σημεία Gauss Στοιχείου Αναφοράς και τα Βάρη τους

Σημεία Gauss	Βάρη
(0.1127,0.1127)	0.077160494
(0.1127,0.5)	0.12345679
(0.1127,0.8872)	0.077160494
(0.5,0.1127)	0.12345679
(0.5,0.5)	0.197530864
(0.5,0.8873)	0.12345679
(0.8873,0.1127)	0.077160494
(0.8873,0.5)	0.12345679
(0.8873,0.8873)	0.077160494

### 3.5 Ροή Αλγορίθμου

Ο αλγόριθμος υλοποιείται σε κώδικα Fortran ο οποίος παρατίθεται στο Παράρτημα. Ακολουθούνται τα εξής βήματα:

1. Ορίζεται ο αριθμός των στοιχείων
2. Γίνεται αρίθμηση των κόμβων των στοιχείων τοπικά και ολικά.
3. Υπολογίζονται οι  $x, y$ -συντεταγμένες κάθε κόμβου.
4. Ορίζονται οι κόμβοι με συνθήκες Dirichlet
5. Υπολογίζεται μία λύση του προβλήματος σε εύκολη περιοχή (μακριά από οριακά σημεία) εκτελώντας μία απλή μέθοδο Newton-Raphson.

6. Η προηγούμενη λύση πολλαπλασιάζεται με έναν συντελεστή και χρησιμοποιείται ως αρχική εκτίμηση για την εύρεση δεύτερης λύσης με μέθοδο Newton. Οι δύο λύσεις χρησιμοποιούνται για την εύρεση των παραγώγων  $\frac{\partial \mathbf{u}}{\partial s}$  και  $\frac{\partial \lambda}{\partial s}$  που χρησιμοποιούνται στην μέθοδο βηματισμού μήκους-τόξου.
7. Εκτελείται η μέθοδος βηματισμού μήκους-τόξου για επιθυμητό αριθμό επαναλήψεων. Κάθε φορά αποθηκεύονται και χρησιμοποιούνται οι δύο προηγούμενες λύσεις για τον υπολογισμό των παραγώγων.

### 3.6 Παρουσίαση Αποτελεσμάτων

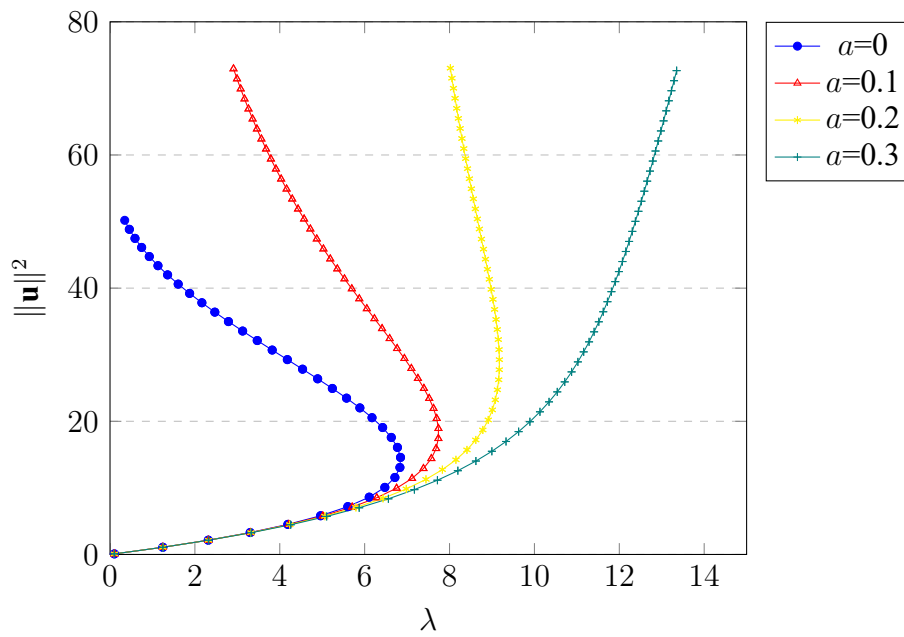
Το ΠΣΤ (εξισώσεις (3.1) και (3.2)) επιλύθηκε για διαφορετικές τιμές του  $a$ , ώστε να βρεθεί η μορφή του χώρου λύσεων (Σχήμα 3.3). Για την αναπαράσταση του χώρου λύσεων χρησιμοποιείται η Ευκλείδεια νόρμα του διανύσματος της λύσης:

$$\|\mathbf{u}\|^2 = \sqrt{\sum_{i=1}^N u_i^2} \quad (3.13)$$

Όπως βλέπουμε με μία πρώτη εκτίμηση (Σχήμα (3.3)), για  $a \leq 0.2$  το ΠΣΤ τιμών εμφανίζει πολλαπλές λύσεις για την ίδια τιμή  $\lambda$ . Πραγματοποιήθηκαν περαιτέρω υπολογισμοί για διαφορετικά  $a$ , με μια απλή μέθοδο διχοτόμησης, ώστε να βρεθεί μία καλύτερη εκτίμηση της κρίσιμης τιμής για την οποία εμφανίζεται οριακό σημείο (Σχήμα 3.4). Η τιμή αυτή του  $a$  είναι περίπου 0.243.

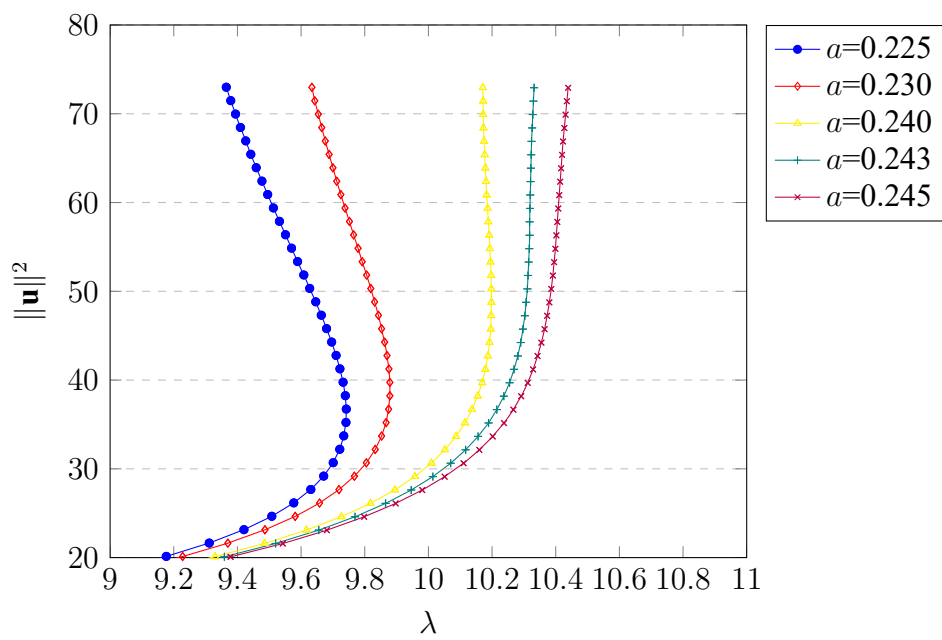
Παρατίθενται επίσης γραφήματα ισοϋψών της λύσης (Σχήματα (3.5) και (3.6)). Όπως εξάγεται από τα γραφήματα, παρότι ο κάτω κλάδος των λύσεων έχει συνεχείς τιμές για διαφορετικές τιμές της παραμέτρου  $a$ , ο άνω κλάδος διαφέρει πολύ. Αυτό σημαίνει πως ο πειραματικός καθορισμός της παραμέτρου  $a$  πρέπει να γίνεται προσεκτικά, λαμβάνοντας υπόψη την μη γραμμική φύση του προβλήματος και την πολλαπλότητα των λύσεων.

Η μέθοδος βηματισμού μήκους-τόξου ήταν ικανή να εξερευνήσει επαρκώς τον χώρο λύσεων και να ξεπεράσει την δυσκολία του οριακού σημείου. Αυτό οφείλε-

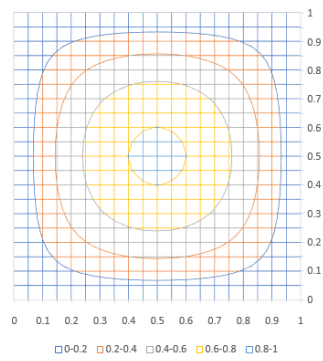
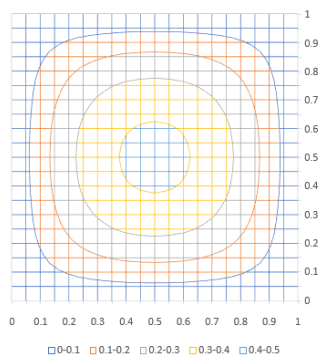


Σχήμα 3.3: Χώρος Λύσεων για εύρος τιμών των παραμέτρων  $a$  και  $\lambda$

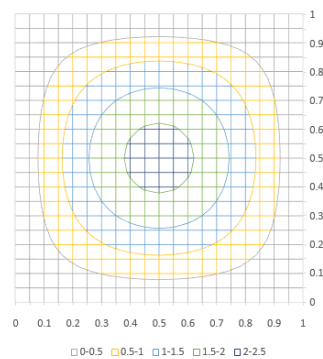
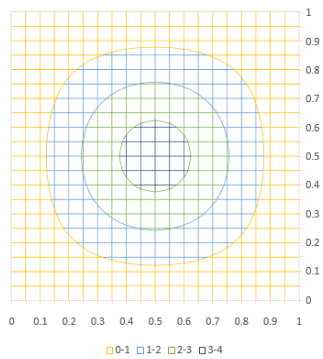
ται στο ότι ο επαυξημένος πίνακας που χρησιμοποιείται από την μέθοδο δεν είναι ιδιάζων κοντά στο οριακό σημείο, όπως θα ήταν εάν χρησιμοποιούσαμε μια απλή μέθοδο βηματισμού παραμέτρου.



Σχήμα 3.4: Μεγένθυση του τμήματος όπου εμφανίζεται το οριακό σημείο. Η κρίσιμη τιμή του  $a$  για την οποία εμφανίζεται οριακό σημείο εκτιμάται ανάμεσα σε 0.24 και 0.243

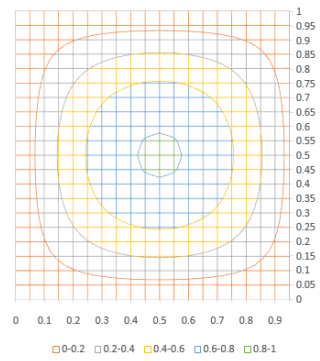
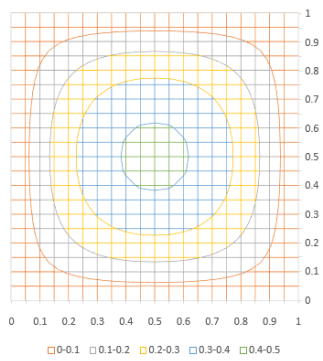


(α') Τιμές λύσης  $u$  για  $\lambda=4.26$ , κάτω κλάδος (β') Τιμές λύσης  $u$  για  $\lambda=6.17$ , κάτω κλάδος

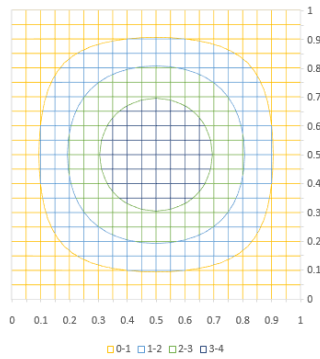


(γ') Τιμές λύσης  $u$  για  $\lambda=4$ , άνω κλάδος (δ') Τιμές λύσης  $u$  για  $\lambda=6.1$ , άνω κλάδος

Σχήμα 3.5: Οι τιμές της λύσης στην επιφάνεια  $x-y$  για  $a=0$



(α') Τιμές λύσης  $u$  για  $\lambda=4.28$ , κάτω κλάδος (β') Τιμές λύσης  $u$  για  $\lambda=6.35$ , κάτω κλάδος



(γ') Τιμές λύσης  $u$  για  $\lambda=4$ , άνω κλάδος (δ') Τιμές λύσης  $u$  για  $\lambda=6.1$ , άνω κλάδος

Σχήμα 3.6: Οι τιμές της λύσης στην επιφάνεια  $x-y$  για  $a=0.1$

## Κεφάλαιο 4

### Μελλοντικά Βήματα

Υπάρχει ενδιαφέρον για την εφαρμογή της μεθόδου σε άλλα προβλήματα συνοριακών τιμών που εμφανίζονται σε συστήματα αντίδρασης-διάχυσης. Είναι δυνατές βελτιώσεις στις δυνατότητες του αλγορίθμου. Η διαδικασία εύρεσης των κρίσιμων τιμών του  $a$  στην παρούσα εργασία έγινε χειροκίνητα, αλλά θα μπορούσε να αυτοματοποιηθεί. Ο αλγόριθμος χρησιμοποιεί απαλοιφή Gauss για την επίλυση των συστημάτων, η οποία δεν δημιουργεί πρόβλημα υπολογιστικού κόστους στις διαστάσεις αυτού του προβλήματος. Ωστόσο, για πολυπλοκότερες εφαρμογές υπάρχουν καλύτερες μέθοδοι όπως ο επιλύτης frontal [14]. Υπάρχουν επίσης μέθοδοι στην βιβλιογραφία οι οποίες χρησιμοποιούν πλέγματα διαφορετικού μεγέθους για αρχικές εκτιμήσεις της λύσης, μειώνοντας το υπολογιστικό κόστος [7]. Ένα ακόμα ενδιαφέρον θέμα σχετικά με την πολλαπλότητα των λύσεων των μη γραμμικών προβλημάτων τύπου αντίδρασης-διάχυσης είναι η μελέτη της αλλαγής της ευστάθειας των λύσεων. Αυτό επιτυγχάνεται με ανάλυση ιδιοτιμών και παρακολούθηση της αλλαγής προσήμου ενός ή περισσοτέρων ιδιοτιμών κατά τον βηματισμό σε παράμετρο.



# Βιβλιογραφία

- [1] Μπουντουβής, Ανδρέας Γ. *Υπολογιστική Ανάλυση με την Μέθοδο των Πεπερασμένων Στοιχείων-Εισαγωγικές Σημειώσεις* 1992.
- [2] Sokolnikoff, I.S. and Redheffer, R.M. 1958 *Mathematics of Physics and Modern Engineering*. McGraw-Hill, New York.
- [3] Μπακόπουλος, Α. και Χρυσοβέργης, Ι. 1986 *Αριθμητικές Μέθοδοι Μερικών Διαφορικών Εξισώσεων*. Αθήνα.
- [4] Zienkiewicz, O. C. 1977 *The Finite Element Method*. 3rd edition, McGraw-Hill, London.
- [5] Keller, H.B. 1977 *Numerical Solution of Bifurcation and Nonlinear Eigenvalue Problems*. In *Applications of Bifurcation Theory* (Ed. by P. Rabinowitz), Academic Press, New York, p.359.
- [6] Rheinboldt, W.C. 1978 *Numerical Methods for a Class of Finite Dimensional Bifurcation Problems*. *SIAM J. Num. Anal.* 15, 1.
- [7] Bolstad, John H. and Keller, Herbert B. *A Multigrid Continuation Method For Elliptic Problems with Folds* *SIAM J. Sci. Stat. Comput.* 1988
- [8] Britton, N.F. *Reaction-diffusion equations and their applications to biology*. London 1986
- [9] Lee, Kyoung-Jin et al. *Experimental observation of self-replicating spots in a reaction–diffusion system*. *Nature Journal*, 1994

- [10] Domenico, Patrick A. and Schwartz, Franklin W. *Physical and Chemical Hydrogeology* 1998
- [11] Stewart, J. D. *Neutron Diffusion Theory* Nuclear Science and Engineering, 1965
- [12] Cantrell, Robert Stephen, and Chris Cosner. *Spatial ecology via reaction-diffusion equations*. John Wiley & Sons, 2004.
- [13] Bratu, G. "Sur les équations intégrales non linéaires." Bulletin de la Société Mathématique de France 42 (1914): 113–142
- [14] Irons, B. M., 1970. *A frontal solution scheme for finite element analysis*. Int. J. Numer. Methods Eng. 2, 5–32

# Παράρτημα

```
program arc_length
!*****
!      2-dimensional nonlinear problem , arc-length method
!      biquadratic finite element basis functions
!*****
parameter (nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,ny,ne,np,xorigin,yorigin,xlast,ylast,deltax,deltay,nop(nemax,9),&
xpt(nnmax),ypt(nnmax),ncod(nnmax),bc(nnmax),&
sk(nnmax,nnmax),r1(nnmax),ud(nnmax),ntop(nemax),nlat(nemax)
common/ublock/u(nnmax),olamda,alpha
common/arclength/uprev(nnmax),uprev2(nnmax),ds,n,normu(nnmax)
common/gauss/w(3),gp(3)
data w,gp/0.2777777777777778,0.4444444444444444,0.2777777777777778,0.1127016654,0.5,0.8872983346/
integer mode
real normu

open(unit=1,file="normu-1.txt")

call xydiscr
call nodnumb
call xycoord
call boundaries
call initial_values
alpha=0
u(np+1)=0.1 λ! parameter
ds=0.1
factor=1.05 !factor that increments the initial solution slightly for arc-length initialization
write(1,*) "a      ",ds      ",factor      "
write(1,*) alpha,ds,factor
write(1,*) "1      ",u"

call nonlinear_loop(1)
do j=1,np
normu(j)=u(j)
```

```

end do
write(1,*) u(np+1),norm2(normu)
do i=1,np+1
uprev2(i)=uprev(i)
uprev(i)=u(i)
u(i)=u(i)*factor
end do

call nonlinear_loop(1)
do j=1,np
normu(j)=u(j)
end do
write(1,*) u(np+1),norm2(normu)
do i=1,np+1
uprev2(i)=uprev(i)
uprev(i)=u(i)
u(i)=u(i)*factor
end do

do i=1,100
call nonlinear_loop(2)
do j=1,np+1
uprev2(j)=uprev(j)
uprev(j)=u(j)
end do

do j=1,np
normu(j)=uprev(j)
end do

write(1,*) u(np+1),norm2(normu)
end do

end
subroutine nonlinear_loop(mode)
parameter(nemax=400,nnmax=1681,tol=0.00001)
common/block1/nex,ney,nnx,ny,ne,np,xorigin,yorigin,xlast,ylast,deltax,deltay,nop(nemax,9),&
xpt(nnmax),ypt(nnmax),ncod(nnmax),bc(nnmax),&
sk(nnmax,nnmax),r1(nnmax),ud(nnmax),ntop(nemax),nlat(nemax)
common/ublock/u(nnmax),olamda,alpha
common/gauss/w(3),gp(3)
common/arclength/uprev(nnmax),uprev2(nnmax),ds,normu(nnmax)
data w,gp/0.2777777777777778,0.4444444444444444,0.2777777777777778,0.1127016654,0.5,0.8872983346/
real normu
integer repetition_count

```

```

repetition_count=0
flag=0
do while (flag.ne.1)
do i=1,np+1
r1(i)=0.
do j=1,np+1
sk(i,j)=0.
ud(j)=0.
end do
end do
do nell=1,ne
call abfind(nell,mode)
end do
!secondary filling of the sk matrix
if (mode==2) then
!calculate dx/ds^T, dl/ds
do i=1,np+1
sk(np+1,i)=(uprev(i)-uprev2(i))/ds
end do

!calculate -n
do i=1,np+1
r1(np+1)=r1(np+1)-(u(i)-uprev(i))*(uprev(i)-uprev2(i))/ds
end do
r1(np+1)=r1(np+1)+ds
end if
!impose essential boundary conditions
do 1003 i=1,np
if(ncod(i).ne.1) go to 1003
r1(i)=bc(i)
do 1004 j=1,np
1004 sk(i,j)=0.
sk(i,i)=1.
1003 continue
call gelim(mode)
!check for final solution, calculate new u.
do i=1,np+1
u(i)=u(i)+ud(i)
enddo
!make a smaller array to use for norm function
do i=1,np
normu(i)=ud(i)
end do
if ((norm2(normu,1)<tol).and.(mode==1)) then
flag=1

```

```

end if
if ((norm2(ud,1)<tol).and.(mode==2)) then
flag=1
end if
repetition_count=repetition_count+1
if(repetition_count>100) then !stops the run if converging takes too long
stop
end if
enddo

end

subroutine xydiscr
!defines the number of elements and size of each
parameter(nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,nyy,ne,np,xorigin,yorigin,xlast,ylast,&
deltax,deltay,nop(nemax,9),xpt(nnmax),ypt(nnmax),&
ncod(nnmax),bc(nnmax),sk(nnmax,nnmax),rl(nnmax),ud(nnmax),&
ntop(nemax),nlat(nemax)
common/gauss/w(3),gp(3)

nex=10
ney=10
xorigin=0.
yorigin=0.
xlast=1.
ylast=1.
deltax=(xlast-xorigin)/nex
deltay=(ylast-yorigin)/ney

return
end

subroutine nodnumb
!creates local to global numbering matrix, number of elements and number of nodes

parameter(nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,nyy,ne,np,xorigin,yorigin,xlast,ylast,&
deltax,deltay,nop(nemax,9),xpt(nnmax),ypt(nnmax),&
ncod(nnmax),bc(nnmax),sk(nnmax,nnmax),rl(nnmax),ud(nnmax),&
ntop(nemax),nlat(nemax)

ne=nex*ney !nex and ney are the number of elements in x and y direction,
!as defined in subroutine xydiscr
nnx=2*nex+1 !nnx= number of nodes in x direction
nyy=2*ney+1 !nyy= number of nodes in y direction
np=nnx*nyy !np is total number of nodes

```

```

!nodal numbering
!nel is the global numbering of the element
!nop(nel,1) is designed to be the global numbering of node 1 (in local numbering) of element nel
nel=0
do i=1,nex
do j=1,ney
nel=nel+1
do k=1,3
l=3*k-2
nop(nel,1)=nny*(2*i+k-3)+2*j-1
nop(nel,1+1)=nop(nel,1)+1
nop(nel,1+2)=nop(nel,1)+2
enddo
enddo
enddo

return
end
subroutine xycoord
!creates (x,y) coordinates of each node
parameter(nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,nny,ne,np,xorigin,yorigin,xlast,ylast,&
deltax,deltay,nop(nemax,9),xpt(nnmax),ypt(nnmax),&
ncod(nnmax),bc(nnmax),sk(nnmax,nnmax),r1(nnmax),ud(nnmax),&
ntop(nemax),nlat(nemax)

xpt(1)=xorigin
ypt(1)=yorigin
!the order of operations is starting from the bottom left of the grid and defining one column at a time
do i=1,nnx
nnode=(i-1)*nny+1
xpt(nnode)=xpt(1)+(i-1)*deltax/2.
ypt(nnode)=ypt(1)
do j=2,nny
xpt(nnode+j-1)=xpt(nnode)
ypt(nnode+j-1)=ypt(nnode)+(j-1)*deltay/2.
enddo
enddo
return
end
subroutine tsfun(x,y)
!quadratic basis functions and their derivatives for given point (x,y) in local coordinates
common/tsfn/phi(9),phic(9),phie(9)
real l1,l2,l3

```

```

!this is the definition of the basis functions
l1(c)=2.*c**2-3.*c+1.
l2(c)=-4.*c**2+4.*c
l3(c)=2.*c**2-c
dl1(c)=4.*c-3.
dl2(c)=-8.*c+4.
dl3(c)=4.*c-1.
!values of combinations of multiplications between functions and derivatives
phi(1)=l1(x)*l1(y)
phi(2)=l1(x)*l2(y)
phi(3)=l1(x)*l3(y)
phi(4)=l2(x)*l1(y)
phi(5)=l2(x)*l2(y)
phi(6)=l2(x)*l3(y)
phi(7)=l3(x)*l1(y)
phi(8)=l3(x)*l2(y)
phi(9)=l3(x)*l3(y)
phic(1)=l1(y)*dl1(x)
phic(2)=l2(y)*dl1(x)
phic(3)=l3(y)*dl1(x)
phic(4)=l1(y)*dl2(x)
phic(5)=l2(y)*dl2(x)
phic(6)=l3(y)*dl2(x)
phic(7)=l1(y)*dl3(x)
phic(8)=l2(y)*dl3(x)
phic(9)=l3(y)*dl3(x)
phie(1)=l1(x)*dl1(y)
phie(2)=l1(x)*dl2(y)
phie(3)=l1(x)*dl3(y)
phie(4)=l2(x)*dl1(y)
phie(5)=l2(x)*dl2(y)
phie(6)=l2(x)*dl3(y)
phie(7)=l3(x)*dl1(y)
phie(8)=l3(x)*dl2(y)
phie(9)=l3(x)*dl3(y)
return
end
subroutine abfind(nell,mode)
!finds contribution of each node of element nell the the matrices A and B of the equation A*X=B
!mode refers to whether this is a run for the initial value estimation which uses a basic matrix(1) or
!of the arc-length method which require an expanded matrix
parameter(nemax=400,nnmax=1681)
common/ublock/u(nnmax),olamda,alpha
common/block1/nex,ney,nnx,nyy,ne,np,xorigin,yorigin,xlast,ylast,&
deltax,deltay,nop(nemax,9),xpt(nnmax),ypt(nnmax),&

```



```

ncod(nnmax),bc(nnmax),sk(nnmax,nnmax),r1(nnmax),ud(nnmax),&
ntop(nemax),nlat(nemax)
common/tsfn/phi(9),phic(9),phie(9)
common/gauss/w(3),gp(3)
data w,gp/0.277777777777778,0.4444444444444,0.277777777777778,0.1127016654,0.5,0.8872983346/
dimension ngl(9),tphx(9),tphy(9)
integer mode

do i=1,9
ngl(i)=nop(nell,i) !returns the global numbering of node i of element nell
enddo

do j=1,3 !j and k loop through all the nodes of element need
do k=1,3
call tsfun(gp(j),gp(k)) !calculate derivatives of basis functions
!at gauss point of the node in local numbering

!isoparametric transformation
x1=0.
x2=0.
y1=0.
y2=0.
do n=1,9
!x1 is the  $\xi$ -derivative of x, y1 is the  $\xi$ -derivative of y
!x2 is the  $\eta$ -derivative of x, y2 is the  $\eta$ -derivative of y
x1=x1+xpt(ngl(n))*phic(n)
x2=x2+xpt(ngl(n))*phie(n)
y1=y1+ypt(ngl(n))*phic(n)
y2=y2+ypt(ngl(n))*phie(n)
enddo

dett=x1*y2-x2*y1 !determinant of the isoparametric mapping matrix

do ia=1,9
!tphx(ia) is the x-derivative of the ia basis function.
!tphy(ia) is the y-derivative of the ia basis function.
tphx(ia)=(y2*phic(ia)-y1*phie(ia))/dett
tphy(ia)=(x1*phie(ia)-x2*phic(ia))/dett
enddo
!residuals section

!calculating du/dx,du/dy,u
usolx=0
usoly=0
usol=0
do l=1,9

```

```

usolx=usolx+u(ngl(1))*tphx(1)
usoly=usoly+u(ngl(1))*tphy(1)
usol=usol+u(ngl(1))*phi(1)
enddo
!calculating -r1
do l=1,9
  rl(ngl(1))=rl(ngl(1))&
  +w(j)*w(k)*dett*(usolx*tphx(1)+usoly*tphy(1))&
  -u(np+1)*w(j)*w(k)*phi(1)*dett*exp(usol/(1+alpha*usol))
enddo

!calculating sk(i,j)
do l=1,9
do m=1,9
sk(ngl(1),ngl(m))=sk(ngl(1),ngl(m))&
-w(j)*w(k)*dett*(tphx(1)*tphx(m)+tphy(1)*tphy(m))&
+u(np+1)*w(j)*w(k)*dett*phi(1)*phi(m)*exp(usol/(1+alpha*usol))*1/((alpha*usol+1)**2)
enddo
enddo

!when calculating the expanded matrix, this section defines  $\theta R \theta /$ 
if (mode==2) then
do l=1,9
sk(ngl(1),np+1)=sk(ngl(1),np+1)+w(j)*w(k)*phi(1)*dett*exp(usol/(1+alpha*usol))
end do
end if
enddo
enddo

return
end

subroutine gelim(mode)
!performs gauss elimination on the AX=B matrix
!mode defines whether the standard(1) or expanded(2) matrix is used
parameter(nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,ny,ne,np,xorigin,yorigin,xlast,ylast,&
deltax,deltay,nop(nemax,9),xpt(nnmax),ypt(nnmax),&
ncod(nnmax),bc(nnmax),sk(nnmax,nnmax),rl(nnmax),ud(nnmax),&
ntop(nemax),nlat(nemax)
integer mode
if (mode==1) then
!transformation to an upper triangular matrix
do i=1,np-1 !row number 1
do j=i+1,np !row number 2

```

```

p=sk(j,i)/sk(i,i)
r1(j)=r1(j)-p*r1(i)
do k=1,np !column number
sk(j,k)=sk(j,k)-p*sk(i,k)
enddo
enddo
enddo
!backward substitution
do l=np,1,-1 !row number 1
ud(l)=r1(l)/sk(l,l)
do j=l-1,1,-1 !row number 2
pl=sk(j,l)/sk(l,l)
sk(j,l)=0.
r1(j)=r1(j)-pl*r1(l)
enddo
enddo
endif
if (mode==2) then
!in mode 2 we use the expanded matrix which is one row and column bigger,
!so we just add 1 to any counter that uses np
do i=1,np
do j=i+1,np+1
p=sk(j,i)/sk(i,i)
r1(j)=r1(j)-p*r1(i)
do k=1,np+1
sk(j,k)=sk(j,k)-p*sk(i,k)
enddo
enddo
enddo

do l=np+1,1,-1
ud(l)=r1(l)/sk(l,l)
do j=l-1,1,-1
pl=sk(j,l)/sk(l,l)
sk(j,l)=0.
r1(j)=r1(j)-pl*r1(l)
enddo
enddo
endif
return
end

subroutine initial_values
parameter(nemax=400,nnmax=1681)
common/block1/nex,ney,nnx,nnny,ne,np,xorigin,yorigin,xlast,ylast,deltax,deltay,nop(nemax,9),&
xpt(nnmax),ypt(nnmax),ncod(nnmax),bc(nnmax),&

```

```

sk ( nnmax , nnmax ) , r1 ( nnmax ) , ud ( nnmax ) , ntop ( nemax ) , nlat ( nemax )
common/ ublock / u ( nnmax ) , olamda , alpha
do i=1,np
u(i)=0.
end do

do i=1, nny
u(i)=0.
enddo

do i= nny , np , nny
u(i)=0.
enddo

do i=1, np- nny , nny
u(i)=0.
enddo

do i=1+ nny *( nnx - 1 ) , np
u(i)=0.
enddo
return
end subroutine
subroutine boundaries

parameter ( nemax=400 , nnmax=1681 )
common/ block1 / nex , ney , nnx , nny , ne , np , xorigin , yorigin , xlast , ylast , deltax , deltay , nop ( nemax , 9 ) , &
xpt ( nnmax ) , ypt ( nnmax ) , ncod ( nnmax ) , bc ( nnmax ) , &
sk ( nnmax , nnmax ) , r1 ( nnmax ) , ud ( nnmax ) , ntop ( nemax ) , nlat ( nemax )
common/ ublock / u ( nnmax ) , olamda , alpha
! prepare for essential boundary conditions
do i=1,np
ncod(i)=0
bc(i)=0.
enddo

do i=1, nny
ncod(i)=1
bc(i)=0.
enddo

do i= nny , np , nny
ncod(i)=1
bc(i)=0.
enddo

```

```
do i=1,np-nny,nny
ncod(i)=1
bc(i)=0.
enddo
```

```
do i=1+nny*(nnx-1),np
ncod(i)=1
bc(i)=0.
enddo
return
end
```