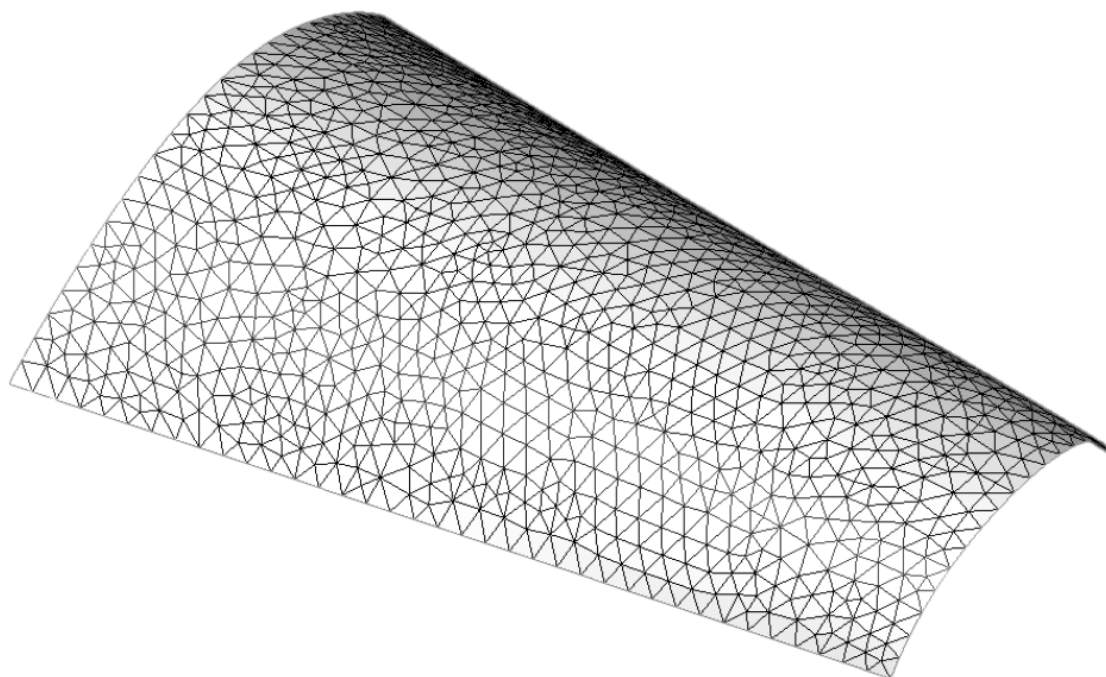




ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΠΟΛΙΤΙΚΩΝ ΜΗΧΑΝΙΚΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΣΤΑΤΙΚΗΣ ΚΑΙ ΑΝΤΙΣΕΙΣΜΙΚΩΝ ΕΡΕΥΝΩΝ

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα



Διπλωματική Εργασία

Φιλυππίδης Αχιλλέας

Επιβλέπων Καθηγητής: Λαγαρός Δ. Νικόλαος

Αθήνα, Μάρτιος 2019

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Διπλωματική Εργασία

Φιλυπίδης Αχιλλέας

Επιβλέπων Καθηγητής: Λαγαρός Δ. Νικόλαος

Αθήνα, Μάρτιος 2019

Copyright © Αχιλλέας Φιλίππιδης, 2019
Με επιφύλαξη παντός δικαιώματος

Απαγορεύεται η αντιγραφή, αποθήκευση σε αρχείο πληροφοριών, διανομή, αναπαραγωγή, μετάφραση ή μετάδοση της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό, υπό οποιαδήποτε μορφή και με οποιοδήποτε μέσο επικοινωνίας, ηλεκτρονικό ή μηχανικό, χωρίς την προηγούμενη έγγραφη άδεια του συγγραφέα. Επιτρέπεται η αναπαραγωγή, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν στη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Η έγκριση της διπλωματικής εργασίας από τη Σχολή Πολιτικών Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου δεν υποδηλώνει αποδοχή των απόψεων του συγγραφέα (Ν. 5343/1932, Άρθρο 202).

Copyright © Achilleas Filippidis, 2019
All Rights Reserved

Neither the whole nor any part of this diploma thesis may be copied, stored in a retrieval system, distributed, reproduced, translated, or transmitted for commercial purposes, in any form or by any means now or hereafter known, electronic or mechanical, without the written permission from the author. Reproducing, storing and distributing this thesis for non-profitable, educational or research purposes is allowed, without prejudice to reference to its source and to inclusion of the present text. Any queries in relation to the use of the present thesis for commercial purposes must be addressed to its author.

Approval of this diploma thesis by the School of Civil Engineering of the National Technical University of Athens (NTUA) does not constitute in any way an acceptance of the views of the author contained herein by the said academic organization (L. 5343/1932, art. 202).

ΕΥΧΑΡΙΣΤΙΕΣ

Με την παράδοση και παρουσίαση της παρούσας διπλωματικής εργασίας ολοκληρώνεται για μένα και ο μεγαλύτερος κύκλος και σημαντικότερος κύκλος της έως τώρα ζωής μου, αυτός του προπτυχιακού φοιτητή πολιτικού μηχανικού. Ένας κύκλος που μου πρόσφερε πολύ περισσότερα απ' όσα περίμενα την στιγμή που έκανα την επιλογή μου, μεταλλάσσοντας με σε έναν τελείως διαφορετικό άνθρωπο από τον δεκαοχτάχρονο νέο που περνούσε τις πύλες του Εθνικού Μετσόβιου Πολυτεχνείου πριν από μερικά χρόνια. Νιώθω την ανάγκη λοιπόν να ευχαριστήσω όλους όσους συνεισέφεραν σε αυτό το ταξίδι, αλλά ακόμα περισσότερο όσους βοήθησαν έμπρακτα στην ολοκλήρωσή του.

Πρώτον απ όλους θα ήθελα να ευχαριστήσω τον καθηγητή κ. Λαγαρό Δ. Νικόλαο, επιβλέπων της διπλωματικής εργασίας μου, για την εξαιρετη συνεργασία που μου προσέφερε, κατανοώντας κάποιες από τις προσωπικές δυσκολίες μου και κάνοντας ότι μπορούσε για να με διευκολύνει. Τον ευχαριστώ επίσης για την ελευθερία που μου άφησε ως προς τον τρόπο διαχείρισης των προβλημάτων που μου έδινε η οποία θεωρώ πως με έκανε αρκετά παραγωγικότερο, αλλά και για την κατεύθυνση που μου έδωσε σε αυτό το τόσο ενδιαφέρον αντικείμενο της τοπολογικής βελτιστοποίησης. Παράλληλα θα ήθελα να ευχαριστήσω και τους υποψήφιους διδάκτορες υπό την επίβλεψή του, κυρίου Σωτηρόπουλο Στέφανο και Καζάκη Γεώργιο για την βοήθεια και συνεργασία που είχαμε σε διάφορα θέματα της διπλωματικής εργασίας. Ακόμα, θα ήθελα να ευχαριστήσω και τον υποψήφιο διδάκτορα κ. Μπακαλάκο Σεραφείμ για τη πολύτιμη βοήθειά του στην κατανόηση της δομής του MSolve, στην ανάπτυξη του οποίου έχει συνεισφέρει.

Θερμά θα ήθελα να ευχαριστήσω και όλους μου τους φίλους που μου στάθηκαν στην διάρκεια των φοιτητικών μου χρόνων, ήτανε δίπλα μου σε στιγμές ευχάριστες και δυσάρεστες για να με συντροφεύσουν κατά το πέρασμά μου από το πρώτο έως και το τελευταίο έτος, την τελευταία ώρα μαθήματος στη Σχολή Πολιτικών Μηχανικών. Δεν θα κάνω την προσπάθεια να τους αναφέρω ονομαστικά έναν έναν, αλλά δεν νομίζω πως χρειάζεται μιας και αυτοί γνωρίζουν ποιοί είναι.

Τέλος, το μεγαλύτερο ευχαριστώ νιώθω πως το οφείλω στην οικογένεια μου. Στον αδερφό μου Φίλιππο για τις υπέροχες συζητήσεις, άλλοτε με ένταση και άλλοτε όχι, αντικείμενο των οποίων πολλές φορές ήταν ο κοινωνικός ρόλος του μηχανικού, ένα αντικείμενο που διαμόρφωσε τις συνειδήσεις μας. Στον πατέρα μου Σπύρο, ο οποίος αποτέλεσε την πρώτη εικόνα μηχανικού που αντίκρισα ήδη από μικρή ηλικία. Και εννοείται στην μητέρα μου η οποία έχει καταβάλει κάθε προσπάθεια ανθρωπίνως δυνατή για να φροντίσει να μη μου λείψει τίποτα όλα αυτά τα χρόνια των σπουδών μου.

Στους γονείς μου

που ήταν πάντα εκεί

και στους φίλους μου

προπτυχιακούς ή μη



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΠΟΛΙΤΙΚΩΝ ΜΗΧΑΝΙΚΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΔΟΜΟΣΤΑΤΙΚΗΣ ΚΑΙ ΑΝΤΙΣΕΙΣΜΙΚΩΝ ΕΡΕΥΝΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΞΙΟΛΟΓΗΣΗ ΜΕΘΟΔΩΝ ΤΟΠΟΛΟΓΙΚΗΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΚΑΤΑΣΚΕΥΩΝ ΣΕ ΜΗ
ΔΟΜΗΜΕΝΑ ΠΛΕΓΜΑΤΑ

ΦΙΛΙΠΠΙΔΗΣ ΑΧΙΛΛΕΑΣ
ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΛΑΓΑΡΟΣ Δ. ΝΙΚΟΛΑΟΣ

ΠΕΡΙΛΗΨΗ

Ποιος είναι ο βέλτιστος φορέας που θα μπορέσει να αναλάβει με τον οικονομικότερο και ασφαλέστερο τρόπο έναν δεδομένο συνδυασμό φορτίσεων και να τον οδηγήσει στις στηρίξεις που του διατίθενται; Ένα ερώτημα που ο εκάστοτε μηχανικός καλείται να απαντήσει με κάθε έργο που αναλαμβάνει να μελετήσει και να σχεδιάσει, έχοντας όμως πλέον το σύγχρονο εργαλείο των αριθμητικών μεθόδων επίλυσης με χρήση της θεωρίας των πεπερασμένων στοιχείων στα χέρια του. Στην χρήση αυτού του εργαλείου έρχονται να πατήσουν οι διάφορες μέθοδοι βελτιστοποίησης που έχουν αναπτυχθεί τα τελευταία χρόνια, αντιμετωπίζοντας έναν φορέα μηχανικής συμπεριφοράς ως μια συνάρτηση, η παραγωγή και επεξεργασία της οποίας μπορεί να δώσει τοπικά η ακόμη και τα πολυπλόκτα ολικά ακρότατα της. Μια αξιόπιστη εφαρμογή των εν λόγω μεθόδων βελτιστοποίησης με χρήση πεπερασμένων στοιχείων ευελπιστεί να αποτελέσει και η παρούσα διπλωματική. Συγκεκριμένα στο πρώτο μέρος της παρατίθεται η θεωρία για την ανάπτυξη ενός τριγωνικού στοιχείου παχιάς πλάκας έξι κόμβων, καθένας εκ των οποίων διαθέτει πέντε βαθμούς ελευθερίας. Χρησιμοποιώντας το τριγωνικό αυτό στοιχείο για την διακριτοποίηση κελυφωτών φορέων επιχειρείται η βελτιστοποίηση της τοπολογίας αυτών με εφαρμογή τριών διαφορετικών μεθόδων, με την μέθοδο SIMP, με την μέθοδο BESO και τέλος με την μέθοδο Level Set. Το διαφορετικό και ενδιαφέρον κομμάτι που έρχεται να προσθέσει η διπλωματική αυτή στην χρήση των παραπάνω μεθόδων βελτιστοποίησης τοπολογίας είναι η εφαρμογή τους σε εντελώς μη δομημένα πλέγματα πεπερασμένων, όπως αυτά που προκύπτουν με την διακριτοποίηση που παρέχει ο αλγόριθμος Delaunay σε τριγωνικά στοιχεία, μια διακριτοποίηση που δύσκολα θα συναντήσει κανείς στις έως τώρα δημοσιεύσεις επί του θέματος. Πέραν της μαθηματικής πλευράς των μεθόδων προβάλλεται και η αλγοριθμική τους εφαρμογή, ώστε να κατανοηθούν σε πρακτικό επίπεδο και να και να παρουσιαστούν οι ιδιαιτερότητες και τα προβλήματα που προέκυψαν από την εφαρμογή της καθεμιάς με χρήση τριγωνικών στοιχείων, έναντι των τετραγωνικών που συναντάει συνήθως κανείς σε ανάλογες εργασίες. Για την διεκπεραίωση των αναλύσεων χρησιμοποιήθηκε η προγραμματιστική δομή MSolve που αναπτύχθηκε στο εργαστήριο στατικής και αντισεισμικών ερευνών του Εθνικού Μετσόβιου Πολυτεχνείου γραμμένη σε

γλώσσα C#, στην οποία εισήχθη ο κώδικας για το εξακομβικό τριγωνικό στοιχείο παχιάς πλάκας καθώς και οι κώδικες για την εφαρμογή των μεθόδων βελτιστοποίησης που αναφέρθηκαν. Παρόλα αυτά, επειδή αφενός δεν πρόκειται για μια από τις πλέον διαδεδομένες γλώσσες προγραμματισμού, και αφετέρου για την κατανόηση των κωδίκων χρειάζεται πρώτα γνώση και κατανόηση της λογικής με την οποία διαχειρίζεται συγκεκριμένα πράγματα η δομή MSolve, ο κώδικας που χρησιμοποιήθηκε δεν παρατίθεται αλλά παρατίθεται ο αντίστοιχος σε matlab, ο οποίος και επεξηγείται βήμα προς βήμα. Τελικώς, παρουσιάζονται εφαρμογές των μεθόδων σε διαφορετικούς φορείς, από καθαρά επίπεδους με χρήση μονάχα της μεμβρανικής λειτουργίας των τριγώνων έως και κελυφωτούς φορείς στον χώρο για τους οποίους η ανάλυση απαιτεί την ενεργοποίηση της συνολικότερης λειτουργίας των τριγωνικών στοιχείων ως πεπερασμένα στοιχεία παχιάς πλάκας. Παράλληλα με την παρουσίαση των εφαρμογών αναλύεται η επιρροή των παραμέτρων που εισάγονται κατά την εκκίνηση κάθε μορφής προβλήματος, όπως η πυκνότητα της διακριτοποίησης στην μορφή της προκύπτουσας λύσης ή ακόμα και στην ύπαρξη της, και γίνεται αποτίμηση των αποτελεσμάτων στα οποία καταλήγει η κάθε μέθοδος. Οι συγκρίσεις των τριών μεθόδων που γίνονται στην παρούσα εργασία αποσκοπούν να συμβάλλουν στην δημιουργία υποβάθρου για περαιτέρω έρευνες και συζήτηση, οι οποίες ίσως κάποια μέρα καταλήξουν στην εύρεση της βέλτιστης μεθόδου για την βελτιστοποίηση της τοπολογίας κελυφωτών φορέων.



NATIONAL TECHNICAL UNIVERSITY OF ATHENS
SCHOOL OF CIVIL ENGINEERING
INSTITUTE OF STRUCTURAL ANALYSIS AND ANTISEISMIC RESEARCH

DIPLOMA THESIS

ASSESSMENT OF TOPOLOGY OPTIMIZATION METHODS IN UNSTRUCTURED MESHES

FILIPPIDIS ACHILLEAS
SUPERVISING PROFESSOR: LAGAROS D. NIKOLAOS

ABSTRACT

Which is the optimal structural model that could receive a given loadcase and steer it towards its supports while undergoing the least deformation? A question that any engineer is called to answer with each project he decides to take on, having though in his possession the modern tool of numerical analyses with the use of the finite element theory. The advantages of this numerical approach are exploited by the purely mathematical methods for the optimization of structures developed in the recent years, confronting a structural model as a function which could give after differentiation local, or even the deeply desired global, maxima. A reliable application of these optimization methods with the use of finite elements hopefully is represented by this thesis. In particular, in the first part the theory for the construction of a six node triangular finite element for thick plates is laid out, with each of the nodes having five degrees of freedom. Using this triangular element for the discretization of shell structural models the topology optimization of them is attempted with three different methods, the method of SIMP, the BESO method and finally the Level Set method. The unique and noteworthy part that this thesis will try to add in the work revolving these methods is their implementation on completely unstructured grids such as those provided by the Delaunay triangulation algorithm, a discretization that seldom one encounters on relevant research. Apart from the mathematical side of the methods, their algorithmic realization is also presented in order to further understand them in a practical level and display their distinctive details and the problems occurring from their implementation with the use of triangular finite elements, in comparison with the usually encountered quadrilateral elements in similar projects. For the execution of the analyses the programming structure MSolve was used, which was developed in the Institute of Structural Analysis and Antiseismic Research of National Technical University of Athens. MSolve is written in the programming language C#, in which the code for the triangular six node finite element was added, as well as those parts responsible for the implementation of the topology optimization methods mentioned above. However, C# isn't one of the most popular programming languages, additionally to the fact that in order to understand the used in this thesis one should first familiarize themselves with the logic behind the

operations of MSolve. For this reason equivalent code written in matlab is exhibited and laid out step by step. In the final parts of this work applications of the above methods on different structural models are presented, from ones that rest in a 2D plane surface and only require the membrane behavior of the elements to shells belonging in the 3D space and require the full structural behavior of our triangular thick plate finite elements. Alongside the demonstration of the examples, the influence of the parameters used at the instantiation of each kind of problem is also discussed, such as the mesh density of the domains impact on the type, or even on the existence, of solution. The results from every approach are assessed and a thoughtful comparison between them is carried out. Ultimate goal of this thesis is to hopefully contribute in the foundations for the discussion about the optimal method for topology optimization problems of shell based structural models.

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1 ^ο	
ΒΑΣΙΚΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΗΣ ΘΕΩΡΙΑΣ ΠΕΠΕΡΑΣΜΕΝΩΝ ΣΤΟΙΧΕΙΩΝ	1
1.1 Γενικά.....	1
1.2 Ισοπαραμετρικές απεικονίσεις-Αριθμητική ολοκλήρωση	3
1.2.1 Ισοπαραμετρικές Απεικονίσεις Τετραπλευρικών Στοιχείων	3
1.2.2 Ισοπαραμετρικές Απεικονίσεις Τριγωνικών Στοιχείων.....	9
1.3 Θεωρία Λεπτών πλακών-κελυφών	11
1.4 Θεωρία παχίων πλακών-κελυφών	13
ΚΕΦΑΛΑΙΟ 2 ^ο	
ΕΞΑΚΟΜΒΙΚΟ ΤΡΙΓΩΝΙΚΟ ΣΤΟΙΧΕΙΟ ΠΛΑΚΑΣ ΒΑΣΙΣΜΕΝΟ ΣΤΗΝ ΘΕΩΡΙΑ ΠΑΧΙΩΝ ΠΛΑΚΩΝ-ΚΕΛΥΦΩΝ	16
2.1 Γενικά.....	16
2.2 Μembranική λειτουργία	16
2.3 Καμπτική λειτουργία	18
2.4 Διατμητική λειτουργία.....	21
2.5 Ανάλυση κώδικα σε matlab.....	24
ΚΕΦΑΛΑΙΟ 3 ^ο	
ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΤΟΠΟΛΟΓΙΑΣ	28
3.1 Γενικά.....	28
3.2 Βελτιστοποίηση τοπολογίας με χρήση της μεθόδου SIMP	31
3.2.1 Μαθηματική Διατύπωση της SIMP.....	31
3.2.2 Εφαρμογή της μεθόδου SIMP σε μη δομημένα πλέγματα	34
3.2.3 Κώδικας matlab για βελτιστοποίηση με την μέθοδο SIMP	35
3.3 Βελτιστοποίηση τοπολογίας με την μέθοδο BESO.....	40
3.3.1 Μαθηματική Διατύπωση της BESO	40
3.3.2 Εφαρμογή της BESO μεθόδου σε μη δομημένα πλέγματα	42
3.3.3 Παρουσίαση κώδικα matlab για την βελτιστοποίηση τοπολογίας με την BESO μέθοδο	43
3.4 Βελτιστοποίηση τοπολογίας με την μέθοδο Level Set	46
3.4.2 Εφαρμογή της μεθόδου Level Set σε μη δομημένα πλέγματα	51
3.4.3 Επεξήγηση κώδικα Matlab για Level Set	53

ΚΕΦΑΛΑΙΟ 4^ο

ΕΦΑΡΜΟΓΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΤΟΠΟΛΟΓΙΑΣ.....	65
4.1 Γενικά.....	65
4.2 Εφαρμογή 1 ^η : Βελτιστοποίηση Τοπολογίας Προβόλου.....	66
4.3 Εφαρμογή 2 ^η : Βελτιστοποίηση Τοπολογίας Γέφυρας	78
4.4 Εφαρμογή 3 ^η : Βελτιστοποίηση τετραγωνικής πλάκας εσωτερικά πακτωμένης	85
4.5 Εφαρμογή 4 ^η : Βελτιστοποίηση πλάκας εγκάρσιως φορτισμένης	91
4.6 Εφαρμογή 5 ^η : Βελτιστοποίηση προβόλου μορφής κελύφους στο χώρο	96

ΚΕΦΑΛΑΙΟ 5^ο

ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΡΟΤΑΣΕΙΣ ΓΙΑ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	103
5.1 Συμπεράσματα	103
5.2 Προτάσεις για μελλοντική έρευνα	107
Βιβλιογραφία.....	108
ΠΑΡΑΡΤΗΜΑ Α	109
ΠΑΡΑΡΤΗΜΑ Β.....	113
ΠΑΡΑΡΤΗΜΑ Γ	116
ΠΑΡΑΡΤΗΜΑ Δ	118

ΚΕΦΑΛΑΙΟ 1^ο

ΒΑΣΙΚΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΗΣ ΘΕΩΡΙΑΣ ΠΕΠΕΡΑΣΜΕΝΩΝ ΣΤΟΙΧΕΙΩΝ

1.1 Γενικά

Από αρχαιοτάτων ήδη χρόνων, η σχεδίαση κατασκευών ικανοποιητικής δυσκαμψίας που θα μπορούσαν να φέρουν με ασφάλεια τα φορτία που προβλεπόταν να τις καταπονήσουν, καταφέροντας συγχρόνως να είναι λειτουργικές και να ικανοποιούν τον σκοπό κατασκευής τους, αποτελούσε την βασική επιδίωξη των πολιτικών μηχανικών. Είτε επρόκειτο για ένα μονώροφο κτίριο κλασσικής κατόψεως είτε για μια γέφυρα ανοίγματος πολλών μέτρων ο μηχανικός έπρεπε, και πρέπει, να εγγυηθεί για την σωματική ακεραιότητα όσων χρησιμοποιούν το έργο του, καθώς και για την αντοχή του έργου αυτού μέχρι το πέρας ζωής που του έχει οριστεί. Οι στόχοι αυτοί αν και εύκολοι ως προς την διατύπωσή τους κάθε άλλο παρά εύκολοι γίνονται κατά την υλοποίησή τους. Για να επιτευχθούν λοιπόν κατά το δυνατόν μεγαλύτερο βαθμό οι απαιτήσεις που συνιστούν ένα σύνηθες πρόβλημα δομοστατικού σχεδιασμού, προϋποτίθεται η εις βάθος κατανόηση από πλευράς του μηχανικού της μηχανικής αντοχής των υλικών που έχουν επιλεγεί για την ενσάρκωση της κατασκευής, όπως επίσης και του στατικού μοντέλου που έχει σχηματιστεί για την ανάλυση της. Ως εκ τούτου, ο υπολογισμός της απόκρισης φορέων μηχανικής συμπεριφοράς υποβαλλόμενων σε στατική φόρτιση αποτέλεσε αντικείμενο που προσέλκυσε μεγάλο ενδιαφέρον από πλευράς επιστημόνων και ερευνητών.

Η αρχική προσέγγιση του προβλήματος, η οποία μάλιστα μονοπώλησε το ενδιαφέρον όσων καταπιάστηκαν με το πρόβλημα της στατικής ανάλυσης φορέων για αρκετό καιρό, αφορά στον σχηματισμό σχέσεων για την περιγραφή του υλικού ως συνεχούς μέσου, κατάλληλων για την περιγραφή των τμημάτων που συνθέτουν τον φορέα. Εν συνεχεία, με εφαρμογή των συνοριακών συνθηκών φορτίσεως και δεσμεύσεως και επίλυση του συστήματος διαφορικών εξισώσεων που σχηματίζονται, μπορεί να ευρεθεί τελικώς η απόκριση του στατικού μοντέλου για το οποίο λαμβάνει χώρα η όλη διαδικασία. Είναι σαφές πως μια τέτοια πορεία αντιμετώπισης μόνο σε απλούς ισοστατικούς έως και υπερστατικούς μερικών μονάχα βαθμών υπερστατικότητας φορείς μπορεί να έχει εφαρμογή, καθώς σε συνθετότερα μοντέλα η αναζήτηση αναλυτικών λύσεων για το στατικό πρόβλημα αποτελεί αντικείμενο αρκετά επίπονο, ακόμα και ανθρωπίνως αδύνατο όταν οι βαθμοί υπερστατικότητας και η πολυπλοκότητα του μοντέλου αυξηθούν παραπάνω από κάποια σημείο.

Όταν λοιπόν το αρχικό στατικό μοντέλο είναι συνθετότερο απ' όσο είναι δυνατόν να περιγραφεί με καλώς διατυπωμένες μαθηματικές εξισώσεις, και συνεπώς να μπορεί να επιλυθεί, μια διαφορετική προσέγγιση στο πρόβλημα απαιτείται. Η συνηθέστερη αντιμετώπιση σε τέτοιες περιπτώσεις είναι η απλοποίηση του γενικότερου στατικού μοντέλου που έχει επιλεγεί για την μαθηματική προσομοίωση της κατασκευής κάνοντας κάποιες παραδοχές για τον τρόπο συμπεριφοράς των μελών του, για τις μεταξύ τους συνθήκες διεπαφής, για τον τρόπο με τον οποίο αναλαμβάνουν την φόρτιση που τους ανατίθεται, όπως επίσης και τον τρόπο με τον οποίο μεταφέρουν την φόρτιση αυτή στις θέσεις στηρίξεώς τους. Έτσι, ακόμα και το συνθετότερο εκ των στατικών προβλημάτων μπορεί μέσω της απλοποίησης του να γίνει επιλύσιμο. Το ζήτημα που τίθεται όμως με την υιοθέτηση αυτής της θεώρησης είναι αυτό της ακρίβειας, καθώς οι απλοποιητικές παραδοχές συνοδεύονται με το τίμημα των κατά προσέγγιση αποτελεσμάτων. Η μειωμένη

ακρίβεια εντείνεται όσο αυξάνονται οι απλοποιήσεις και παραδοχές για το φορέα, οδηγώντας πολλές φορές σε αποτελέσματα που μικρή σχέση έχουν με την πραγματική συμπεριφορά του φορέα.

Και κάπως έτσι φτάνουμε στον σύγχρονο τρόπο αντιμετώπισης της ανάλυσης στατικών φορέων με την εφαρμογή της θεωρίας των πεπερασμένων στοιχείων. Όντας ουσιαστικά μια μέθοδος αριθμητικής ανάλυσης της απόκρισης, η ακρίβεια της είναι εξ ορισμού περιορισμένη σε σύγκριση με τις αναλυτικές μορφές λύσεων που προκύπτουν από την επίλυση των διαφορικών εξισώσεων του μοντέλου, όταν βεβαίως αυτές μπορούν να διατυπωθούν. Παρόλα αυτά η μέθοδος δεν υπόσχεται απόλυτη σύγκλιση στην ακριβή λύση σε τάξη πολλών δεκαδικών ψηφίων, αλλά αντ' αυτού αναμένεται ικανοποιητική σύγκλιση ταυτόχρονα με την ικανότητα να αναλυθεί και το πλέον σύνθετο στατικά μοντέλο φορέα, χωρίς να γίνουν απλοποιήσεις που να αφορούν τον τρόπο συμπεριφοράς του.

Βασικός εφευρέτης και θεμελιωτής της μεθόδου δεν ήταν άλλος από τον John Argyris, απόφοιτο πολιτικό μηχανικό του Εθνικού Μετσόβιου Πολυτεχνείου. Αποτελώντας μέλος του Τμήματος Μηχανικών της Βασιλικής Κοινωνίας Αεροναυτικής του Λονδίνου (Royal Aeronautical Society of London), και συγκεκριμένα κατά την διάρκεια του δευτέρου παγκοσμίου πολέμου, καταπιάστηκε με την προσπάθεια σχεδιασμού καλύτερων μαχητικών αεροσκαφών με σκοπό την ενίσχυση της πολεμικής ισχύος των Συμμαχικών δυνάμεων, σκοπό στον οποίο είχε στρατολογηθεί το μεγαλύτερο μέρος της επιστημονικής κοινότητας της τότε εποχής. Εκείνη την περίοδο οι αεροναυπηγοί μηχανικοί προσπαθούσαν να αναπτύξουν μια μέθοδο ώστε να καταφέρουν να αναλύσουν τις τάσεις στις οπισθοκλινείς πτέρυγες των μαχητικών αεροσκαφών, ένα έργο στο οποίο οι αντίστοιχοι μηχανικοί των δυνάμεων του Άξονα είχαν αποτύχει όπως μπορούσε να διακρίνει κανείς στο προβληματικό μαχητικό ME262. Η μη ορθογωνική γεωμετρία των πτερυγίων δημιουργούσε πρόβλημα στον υπολογισμό των τάσεων που αναπτυσσόταν στο πτερύγιο και άρα στην σωστή διαστασολόγηση τους. Τον Αύγουστο του 1943, ο John Argyris αφοσιώθηκε για τρία μερόνυχτα στην επίλυση αυτού του δύσκολου γρίφου με μοναδική του βοήθεια ένα από τα αρχικά υπολογιστικά συστήματα που τότε αναπτυσσόταν και ήταν δυνατόν να διαχειριστεί συστήματα έως 64 αγνώστων, και τελικώς κατέληξε στο συμπέρασμα πως η λύση θα μπορούσε να βρεθεί με την εφαρμογή τριγωνικών στοιχείων για την δημιουργία ενός πλέγματος. Η μέθοδος αυτή έδωσε αμέσως τα πολυπόθητα αποτελέσματα με μικρό σε σύγκριση με τα προηγούμενα σφάλμα, με αποτέλεσμα την γένεση της έννοιας της μητρικής ανάλυσης μέσω του μητρώου στιβαρότητας για έναν φορέα και του υπολογισμού του διανύσματος επικόμβιων μετατοπίσεων. Αμέσως, όλες οι δημοσιεύσεις που έγιναν επί του θέματος ανακλήθηκαν και χαρακτηρίστηκαν ως κρατικά μυστικά. Με την πάροδο του χρόνου, η θεωρία που εφαρμόστηκε από τον John Argyris για την επίλυση του σύνθετου αυτού προβλήματος ονομάστηκε 'Η Θεωρία των Πεπερασμένων Στοιχείων'.

Η βασική λογική της μεθόδου έγκειται στην δημιουργία ενός μητρώου δυσκαμψίας της κατασκευής μέσω του οποίου οι επικόμβιες δράσεις αντιστοιχίζονται σε επικόμβιες μετατοπίσεις και αντιστρόφως. Ειδικότερα, στην περίπτωση που ο φορέας βρίσκεται ακόμα στην ελαστική περιοχή και η θεώρηση μικρών μετακινήσεων έχει νόημα, η αντιστοίχιση μεταξύ δράσεων και μετακινήσεων μπορεί να είναι άμεση, ενώ στην περίπτωση που υπάρχουν λόγοι για να θεωρήσουμε μη γραμμική ανάλυση για την κατασκευή οι επικόμβιοι άγνωστοι θα ευρεθούν μέσω επαναληπτικής διαδικασίας, όπως είναι για παράδειγμα η μέθοδος Newton-Raphson. Στην περίπτωση που το μητρώο δυσκαμψίας για το κάθε στοιχείο δε χρειάζεται αριθμητικό υπολογισμό αλλά είναι έτοιμο και χρειάζεται απλώς πολλαπλασιασμό με τις ελαστικές ιδιότητες του, έχουμε να κάνουμε με την μέθοδο της άμεσης στιβαρότητας, μια ειδικότερη περίπτωση της μεθόδου των πεπερασμένων στοιχείων. Από την άλλη, όταν είναι γνωστή μονάχα η γεωμετρία των στοιχείων, δηλαδή οι θέσεις των κόμβων που το αποτελούν από το πλέγμα, καθώς και οι ελαστικές του ιδιότητες αλλά τίποτε πέραν αυτών των στοιχείων, τότε είναι που πρέπει να εφαρμοστεί η μέθοδος

των πεπερασμένων στοιχείων στο πλήρες εύρος της. Για την πλήρη εφαρμογή της απαιτείται η εύρεση των μητρώων ακαμψίας των στοιχείων έχοντας μονάχα τα παραπάνω δεδομένα, και η οποία επιτυγχάνεται μέσω ολοκληρώσεως, αριθμητικής ή μη.

Η σχέση μέσω της οποίας υπολογίζεται το μητρώο δυσκαμψίας του κάθε στοιχείου δίνεται από την παρακάτω γενική σχέση, η οποία ισχύει για κάθε είδους πεπερασμένο στοιχείο

$$[K] = \sum_m [t^{(m)}]^T \left[\int_{V_e} [B^{(m)}]^T [E] [B^{(m)}] dV_e \right] [t^{(m)}] \quad (1.1)$$

όπου το άθροισμα εκτελείται για όλα τα m στοιχεία του πλέγματος πεπερασμένων στοιχείων, ενώ ο πολλαπλασιασμός με το μητρώο $[t^{(m)}]$ εξυπηρετεί τον μετασχηματισμό του μητρώου δυσκαμψίας του στοιχείου από το τοπικό Καρτεσιανό του σύστημα στο καθολικό σύστημα ως προς το οποίο εκφράζεται το καθολικό μητρώο δυσκαμψίας του φορέα. Οι όροι $[B^{(m)}]$ και $[E]$ συμβολίζουν το μητρώο παραμορφώσεως του στοιχείου και το μητρώο ελαστικότητας αυτού αντίστοιχα, τον τρόπο δημιουργίας και την έννοια των οποίων θα την δούμε αναλυτικότερα στην συνέχεια. Ο τρόπος εξαγωγής της παραπάνω σχέσης μπορεί να ευρεθεί σε όλα σχεδόν τα βιβλία περί της θεωρίας πεπερασμένων στοιχείων, αν και στην συγκεκριμένη ακριβώς μορφή ευρέθη στο βιβλίο του κ. Παπαδρακάκη 'Ανάλυση Φορέων με την Μέθοδο των Πεπερασμένων Στοιχείων'[12]. Οι όροι όμως που παρουσιάζονται στην παραπάνω σχέση θα παρουσιαστούν ένας προς ένας, καθώς θα τους συναντήσουμε κατά την διαμόρφωση του μητρώου δυσκαμψίας του εξακομβικού τριγωνικού στοιχείου που χρησιμοποιείται στην παρούσα διπλωματική

1.2 Ισοπαραμετρικές απεικονίσεις-Αριθμητική ολοκλήρωση

1.2.1 Ισοπαραμετρικές Απεικονίσεις Τετραπλευρικών Στοιχείων

Επινόηση των Taig και Irons, οι ισοπαραμετρικές απεικονίσεις αποτελούν στην ουσία τους μια αναγωγή της γεωμετρίας και των συντεταγμένων ενός πεπερασμένου στοιχείου σε ένα ισοδύναμο σύστημα συντεταγμένων. Παρόλα αυτά, επειδή η ονομασία φυσικό σύστημα έχει δοθεί από τον John Argyris και για το σύστημα του τριγώνου με άξονες αναφοράς τους άξονες παράλληλους στις πλευρές του, στη συνέχεια δε θα γίνεται χρήση του όρου αυτού και θα αποκαλούμε το σύστημα απλώς ισοπαραμετρικό. Η ισοπαραμετρική θεώρηση δίνει την δυνατότητα της εφαρμογής της θεωρίας των πεπερασμένων στοιχείων ακόμα και σε μερικώς παραμορφωμένα στοιχεία, υπό την έννοια της καμπυλότητας των πλευρών τους ή μη ορθογωνικότητας των γωνιών τους, σε περιπτώσεις που αυτές θα έπρεπε να είναι ορθογώνιες. Για τον υπολογισμό του μητρώου στιβαρότητας εφαρμόζεται αριθμητική ολοκλήρωση, ενώ χρειάζεται και ο υπολογισμός ενός μητρώου μετασχηματισμού $[J]$ το οποίο εκφράζει την Ιακωβιανή της απεικόνισης, μιας και παρεμβάλλεται κατά τους υπολογισμούς ο μετασχηματισμός στο ισοπαραμετρικό σύστημα.

Ο όρος ισοπαραμετρικός εξηγείται από το γεγονός ότι τόσο οι συνιστώσες της μετατόπισης u, v, w όσο και οι Καρτεσιανές συντεταγμένες x, y, z εκφράζονται, ως προς τις αντίστοιχες επικόμβιες ποσότητες, με τις ίδιες συναρτήσεις παρεμβολής ή συναρτήσεις σχήματος. Εάν ο βαθμός του πολυωνύμου που ορίζει τις συντεταγμένες είναι μικρότερος του βαθμού του πολυωνύμου των μετατοπίσεων τότε έχουμε την λεγόμενη υποπαραμετρική θεώρηση, ενώ αν συμβαίνει το αντίστροφο τότε έχουμε την υπερπαραμετρική θεώρηση. Τα ισοπαραμετρικά στοιχεία που παρουσιάζονται ακολουθούν τον ίδιο τρόπο παρεμβολής των επικόμβιων μετατοπίσεων για τον σχηματισμό των πεδίων

μετατοπίσεων εντός του στοιχείου με αυτόν που παρουσιάστηκε στο προηγούμενο κεφάλαιο για τα συνήθη πεπερασμένα, είναι δηλαδή και αυτά σχηματισμένα ως στοιχεία συνεχούς μέσου.

Μιας και η ανάλυση γενικότερα του ισοπαραμετρικού τρόπου απεικόνισης για τα πεπερασμένα στοιχεία αποτελεί θέμα που θα χρειαστεί αρκετές σελίδες για να ολοκληρωθεί στο πλήρες εύρος του, εδώ παρουσιάζονται μονάχα τα απαραίτητα για την ισοπαραμετρική απεικόνιση στοιχείων πλάκας και κελύφους, δηλαδή επιφανειακών στοιχείων στον χώρο. Μιας και πρόκειται περί επιφανειακών στοιχείων, αρκεί η κατανόηση της απεικόνισης στοιχείων του επιπέδου. Αρχικώς παρουσιάζεται η απεικόνιση ενός τετραπλευρικού στοιχείου τεσσάρων κόμβων, και στη συνέχεια παρουσιάζεται η απεικόνιση ενός τριγωνικού, όντας πιο συνθετότερη και περιλαμβάνοντας την χρήση ενός επιπλέον συστήματος συντεταγμένων που θα παρουσιαστεί στην πορεία.

Για τον υπολογισμό των μητρώων στιβαρότητας απαιτείται η απεικόνιση του Καρτεσιανού συστήματος στο ισοπαραμετρικό. Η ορθή απεικόνιση, δηλαδή η απεικόνιση των Καρτεσιανών συντεταγμένων x, y ως προς τις ισοπαραμετρικές συντεταγμένες r, s είναι

$$\begin{aligned} x &= x(r, s) \\ y &= y(r, s) \end{aligned} \quad (1.2)$$

ενώ η αντίστροφη απεικόνιση των ισοπαραμετρικών συντεταγμένων συναρτήσει των Καρτεσιανών δίνεται ως

$$\begin{aligned} r &= r(x, y) \\ s &= s(x, y) \end{aligned} \quad (1.3)$$

Κάποιος που έχει ασχοληθεί με τις ισοπαραμετρικές απεικονίσεις κατά πάσα πιθανότητα θα έχει συνηθίσει την χρήση των ελληνικών γραμμάτων ξ, η (και ζ για την καθ ύψος διάσταση) έναντι των λατινικών r, s (και t για την καθ ύψος), ακόμα και σε διεθνή συγγράμματα. Ο λόγος που αυτά δε χρησιμοποιούνται στο παρόν είναι πως παρακάτω παρουσιάζεται και ο κώδικας σε matlab υπεύθυνος για την υλοποίηση του πεπερασμένου στοιχείου (Παράρτημα Α), συνεπώς είναι καλύτερα για τον αναγνώστη να εξοικειωθεί με τον τρόπο συμβολισμού που χρησιμοποιείται στον κώδικα, δηλαδή τους λατινικούς χαρακτήρες για το ισοπαραμετρικό σύστημα.

Σημαντικό στοιχείο κατά την εφαρμογή της ισοπαραμετρικής απεικόνισης είναι πως αυτή πρέπει να είναι αμφιμονοσήμαντη, δηλαδή κάθε Καρτεσιανό σημείο να αντιστοιχεί σε ένα και μόνον ισοπαραμετρικό, και το αντίστροφο.

Ο μετασχηματισμός των συντεταγμένων εξαρτάται από τον αριθμό των κόμβων σε κάθε πλευρά του στοιχείου. Από την στιγμή που έχουμε δυο κόμβους σε κάθε πλευρά ο μετασχηματισμός αυτός θα είναι γραμμικός, ενώ στην περίπτωση που υπήρχε και τρίτος ενδιαμέσος, ο μετασχηματισμός θα είχε την μορφή πολυωνύμου δευτέρας τάξεως. Άρα θα έχουμε

$$x = a_1 + a_2 r + a_3 s + a_4 r s \quad (1.4)$$

$$y = a_5 + a_6 r + a_7 s + a_8 r s \quad (1.5)$$

ή σε μορφή διανύσματος

$$x = [1 \quad r \quad s \quad rs] \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \quad (1.6)$$

$$y = [1 \quad r \quad s \quad rs] \begin{bmatrix} a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} \quad (1.7)$$

Με ανάλογες σχέσεις λόγω της ισοπαραμετρικής θεώρησης εκφράζεται και το πεδίο των μετατοπίσεων στο Καρτεσιανό σύστημα

$$u = [1 \quad r \quad s \quad rs] \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \end{bmatrix} \quad (1.8)$$

$$v = [1 \quad r \quad s \quad rs] \begin{bmatrix} \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \end{bmatrix} \quad (1.9)$$

Όπως θα δούμε παρακάτω, το στοιχείο που χρησιμοποιείται στην παρούσα διπλωματική είναι ένα τριγωνικό στοιχείο έξι κόμβων, και έτσι σε αναλογία με τα τετραπλευρικά στοιχεία οι συναρτήσεις σχήματος για τον μετασχηματισμό των συντεταγμένων θα έχει την μορφή πολωνύμου με όρους υψωμένους στο τετράγωνο.

Οι γενικευμένες συντεταγμένες $\{\alpha\}$ ή $\{\beta\}$ μπορούν να υπολογιστούν μέσω των παρακάτω σχέσεων

$$\begin{aligned} x_1 &= a_1 - a_2 - a_3 + a_4 \quad (r = -1, s = -1) \\ x_2 &= a_1 + a_2 - a_3 - a_4 \quad (r = 1, s = -1) \\ x_3 &= a_1 + a_2 + a_3 + a_4 \quad (r = 1, s = 1) \\ x_4 &= a_1 - a_2 + a_3 - a_4 \quad (r = -1, s = 1) \end{aligned} \quad (1.10)$$

ή

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \quad (1.11)$$

Τελικώς, οι γενικευμένες συντεταγμένες $\{\alpha\}$ δίνονται υπό μορφή

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad (1.12)$$

Με την αντικατάσταση της σχέσης (1.12) στην (1.6) παίρνουμε την έκφραση της συντεταγμένης x ως προς τις συντεταγμένες x_i των τεσσάρων κόμβων του στοιχείου. Προκύπτει

$$[x] = \begin{bmatrix} \frac{1}{4}(1-r)(1-s) & \frac{1}{4}(1+r)(1-s) & \frac{1}{4}(1-r)(1+s) & \frac{1}{4}(1+r)(1+s) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad (1.13)$$

ή σε πιο συνεπτυγμένη μορφή

$$[x] = [N_1 \quad N_2 \quad N_3 \quad N_4] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad (1.14)$$

όπου με N_i δίνονται οι συναρτήσεις σχήματος για τον κάθε κόμβο του στοιχείου, οι εκφράσεις των οποίων είναι

$$\begin{aligned} N_1 &= \frac{1}{4}(1-r)(1-s) \\ N_2 &= \frac{1}{4}(1+r)(1-s) \\ N_3 &= \frac{1}{4}(1+r)(1+s) \\ N_4 &= \frac{1}{4}(1-r)(1+s) \end{aligned} \quad (1.15)$$

Με αντίστοιχο τρόπο ορίζεται και ο μετασχηματισμός για την y συντεταγμένη των στοιχείων

$$[y] = [N_1 \quad N_2 \quad N_3 \quad N_4] \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \quad (1.16)$$

για να καταλήξουμε τελικώς στον μετασχηματισμό της σχέσης (1.17) απ' όπου προκύπτει το πεδίο μετατοπίσεων εντός του εκάστοτε τετραγωνικού στοιχείου έχοντας γνώση μονάχα των μετατοπίσεων στους κόμβους αυτού.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{bmatrix} \quad (1.17)$$

ή

$$\{u\} = [N]\{d\} \quad (1.18)$$

Το μητρώο $[N]$ της παραπάνω σχέσης αποτελεί το μητρώο συναρτήσεων σχήματος για τον ισοπαραμετρικό μετασχηματισμό.

Υπολογίζοντας το μητρώο παραμορφώσεως μέσω της απαραίτητης διαφορίσης των συναρτήσεων σχήματος θα ολοκληρωνόντουσαν και οι απαραίτητες για τον υπολογισμό του μητρώου στιβαρότητας εργασίες, εφόσον βέβαια το σύστημα στο οποίο έγινε ο υπολογισμός των συναρτήσεων σχήματος είναι και το κατεξοχήν σύστημα που είναι εκφρασμένες οι συντεταγμένες των κόμβων του πεπερασμένου στοιχείου. Επειδή όμως έχει παρεμβληθεί η μετατροπή στο ισοπαραμετρικό σύστημα, δε μπορεί να γίνει με άμεσο τρόπο η διαφορίση των συναρτήσεων σχήματος ως προς x και y . Έστω λοιπόν πως έχουμε μια δισδιάστατη συνάρτηση $\phi_{(x,y)}$ ορισμένη εντός του στοιχείου, όπως είναι άλλωστε και οι δισδιάστατες συναρτήσεις του πεδίου μετατοπίσεων u και v εντός του στοιχείου. Αφού εκφράσουμε την εν λόγω συνάρτηση ως προς r και s με χρήση των σχέσεων (1.4) και (1.5), θα κάνουμε χρήση του κανόνα της αλυσίδας για να υπολογίσουμε τις ζητούμενες παραγώγους ως προς x και y , όπως φαίνεται από τις παρακάτω σχέσεις

$$\frac{\partial \phi}{\partial x} = \frac{\partial \phi}{\partial r} \frac{\partial r}{\partial x} + \frac{\partial \phi}{\partial s} \frac{\partial s}{\partial x} \quad (1.19)$$

$$\frac{\partial \phi}{\partial y} = \frac{\partial \phi}{\partial r} \frac{\partial r}{\partial y} + \frac{\partial \phi}{\partial s} \frac{\partial s}{\partial y}$$

ή και αντιστρόφως μπορούμε να εκφράσουμε τις παραγώγους ως προς r και s της συναρτήσεως βάσει των παραγώγων της ως προς x και y αυτής με χρήση του ανωτέρω κανόνα, δηλαδή

$$\frac{\partial \phi}{\partial r} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial r} \quad (1.20)$$

$$\frac{\partial \phi}{\partial s} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial s} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial s}$$

Η σχέση (1.20) μπορεί να γραφεί υπό συνεπτυγμένη μητρική μορφή ως

$$\begin{bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial y} \end{bmatrix} \quad (1.21)$$

ή

$$\begin{bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{bmatrix} = [J] \begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial y} \end{bmatrix} \quad (1.22)$$

όπου $[J]$ είναι το Ιακωβιανό μητρώο που προκύπτει από την παραγωγή των εκφράσεων των x και y των σχέσεων (1.4) και (1.5) ως προς τις ισοπαραμετρικές συντεταγμένες r και s . Το μητρώο του Ιακωβιανού μετασχηματισμού δίνεται λοιπόν από την σχέση

$$[J] = \begin{bmatrix} N_{1,r} & N_{2,r} & N_{3,r} & N_{4,r} \\ N_{1,s} & N_{2,s} & N_{3,s} & N_{4,s} \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{bmatrix} \quad (1.23)$$

Άρα τελικώς για να υπολογίσουμε τις απαιτούμενες παραγώγους προς x και y δεν έχουμε παρά να λύσουμε ως προς αυτές την σχέση (1.22), κάνοντας χρήση του Ιακωβιανού μητρώου του μετασχηματισμού όπως αυτό υπολογίστηκε από την σχέση (1.23) για την ισοπαραμετρική απεικόνιση, δηλαδή:

$$\begin{bmatrix} \frac{\partial \varphi}{\partial x} \\ \frac{\partial \varphi}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \frac{\partial \varphi}{\partial r} \\ \frac{\partial \varphi}{\partial s} \end{bmatrix} \quad (1.25)$$

Η ολοκληρωτική σχέση από την οποία θα προκύψει το μητρώο στιβαρότητας του στοιχείου είναι

$$[k] = \int_{A_e} [B]^T [E] [B] t dA_e \quad (1.26)$$

όπου $[B]$ είναι το μητρώο παραμορφώσεως του στοιχείου, το οποίο περιέχει τις παραγώγους των συναρτήσεων σχήματος ως προς τις Καρτεσιανές συντεταγμένες x και y , $[E]$ είναι το μητρώο ελαστικότητας του στοιχείου και t είναι το πάχος του. Επειδή όμως το μητρώο παραμορφώσεως $[B]$ έχει υπολογιστεί βάσει των παραγώγων ως προς τις ισοπαραμετρικές συντεταγμένες r και s , η παραπάνω σχέση λαμβάνει την μορφή

$$[k] = \int_{-1}^1 \int_{-1}^1 [B(r,s)]^T [E] [B(r,s)] t \det[J] dr ds \quad (1.27)$$

όπου παρατηρούμε την συμπερίληψη της ορίζουσας $\det[J]$ του Ιακωβιανού μητρώου μετασχηματισμού λόγω της ισοπαραμετρικής απεικόνισης.

Ο αναλυτικός υπολογισμός του ολοκληρώματος της σχέσης (1.27), απ' όπου και θα προκύψει το μητρώο στιβαρότητας για το εκάστοτε πεπερασμένο στοιχείο αποτελεί εξαιρετικά δύσκολο εγχείρημα (σε ορισμένες μάλιστα περιπτώσεις ακατόρθωτο). Προκειμένου λοιπόν να γίνει με κάποιον τρόπο ο απαραίτητος υπολογισμός του ολοκληρώματος ώστε να μπορέσει να σχηματισθεί το μητρώο καθολικής στιβαρότητας του φορέα και να επιλυθεί η εξίσωση ισορροπίας, καταφεύγουμε στην αριθμητική ολοκλήρωση αυτού. Υπάρχουν αρκετοί τρόποι να εφαρμοστεί η αριθμητική ολοκλήρωση για να υπολογιστεί το ολοκλήρωμα. Ο συνηθέστερος στην περίπτωση των πεπερασμένων στοιχείων είναι η αριθμητική ολοκλήρωση Gauss, και αυτή θα εφαρμοστεί και στο παρών έργο.

Για δισδιάστατη αριθμητική ολοκλήρωση, όπως αυτή που απαιτείται για επιφανειακά στοιχεία πλακών-κελυφών, ο τύπος λαμβάνει την μορφή

$$\int [F(r,s)] dr ds = \sum_{i,j} w_{ij} [F(r_i, s_j)] + [e] \quad (1.28)$$

όπου το άθροισμα περιλαμβάνει όλα τα i, j ισοπαραμετρικά σημεία του στοιχείου, και ο συντελεστής $w_{i,j}$ αποτελεί τον συντελεστή βάρους των σημείων αυτών. Στην περίπτωση που η ισοπαραμετρική απεικόνιση επιτυγχάνεται με αναγωγή τόσο της κατά x όσο και της κατά

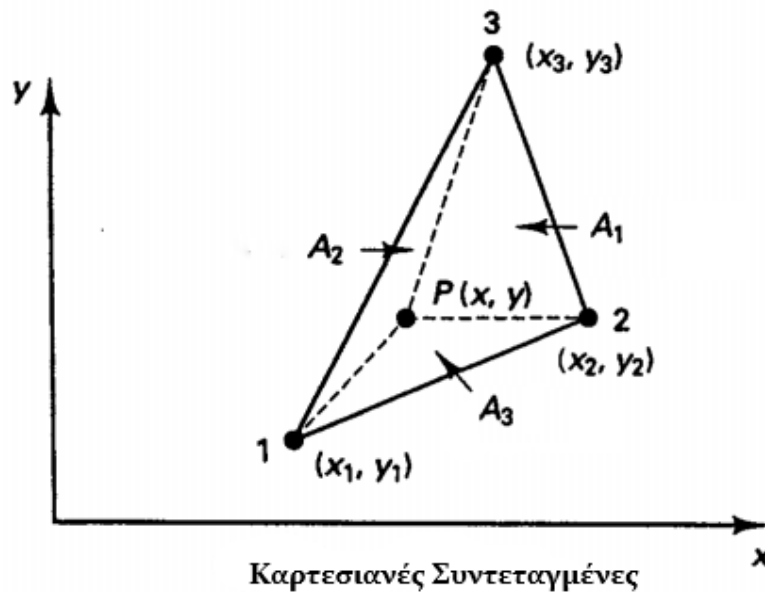
γ διάστασης ξεχωριστά, το βάρος w_{ij} προκύπτει από το γινόμενο των κατά x και κατά y βαρών. Τέτοιες διαφορετικές για κάθε διάσταση ισοπαραμετρικές απεικονίσεις είναι αρκετά συνήθεις σε ορθογωνικές γεωμετρίες στοιχείων, καθώς μπορεί ο μηχανικός να επιλέξει μεγαλύτερη ή μικρότερη πυκνότητα σημείων κατά μια διάσταση ανάλογα με το τι εξυπηρετεί τους σκοπούς του ή τι πιστεύει πως θα δώσει πιο ακριβή αποτελέσματα. Σε μη ορθογωνικές όμως γεωμετρίες, όπως αυτές των τριγωνικών πεπερασμένων στοιχείων, κάθε σημείο αντιστοιχίζεται με ένα βάρος, όπως επίσης η πυκνότητα των σημείων είναι καθολική για όλη την επιφάνεια και δε γίνεται μεγαλύτερη πύκνωση της μιας διάστασης σε σχέση με τις υπόλοιπες. Ο όρος [e] της παραπάνω σχέσης συμβολίζει το σφάλμα κατά τον αριθμητικό υπολογισμό, με την συμπερίληψη του οποίου είναι πλήρης ο υπολογισμός του ολοκληρώματος. Παρόλα αυτά ο όρος αυτός δεν υπολογίζεται κατά την εκτέλεση του αριθμητικού υπολογισμού, εξ ου και το προσεγγιστικό της λύσεως. Πως άλλωστε θα μπορούσε βέβαια να υπολογιστεί, αφού για να γίνει η εκτίμηση του απαιτείται η γνώση της αναλυτικής λύσεως, που αν ήταν γνωστή δεν θα υπήρχε λόγος για την εκτέλεση της αριθμητικής ολοκληρώσεως. Έτσι τα ολοκληρώματα υπολογίζονται προσεγγιστικώς από τα αθροίσματα

$$\int [F(r,s)] dr ds \approx \sum_{i,j} w_{ij} [F(r_i, s_j)] \quad (1.29)$$

όπου ο όρος $F(r_i, s_j)$ συμβολίζει την τιμή της προς ολοκλήρωση συναρτήσεως στην θέση i, j . Τα σημεία ολοκλήρωσης Gauss καθώς και τα βάρη που αντιστοιχούν σε αυτά για ισοπαραμετρικές απεικονίσεις ορθογωνικών γεωμετριών μπορούν να ευρεθούν σχεδόν σε όλα τα συγγράμματα περί πεπερασμένων στοιχείων, όπως και στο βιβλίο του καθηγητή κ. Παπαδρακάκη Μανώλη που έχει βασιστεί μεγάλο μέρος της παρουσίασης των πεπερασμένων στοιχείων. Επειδή στο παρόν δεν ερχόμαστε αντιμέτωποι με την ισοπαραμετρική απεικόνιση τέτοιων στοιχείων, αλλά η απεικόνιση που θα μας απασχολήσει είναι αυτή των τριγωνικών στοιχείων, δε θα επεκταθούμε παραπάνω στην κατανόηση και εκτέλεση της.

1.2.2 Ισοπαραμετρικές Απεικονίσεις Τριγωνικών Στοιχείων

Για την εφαρμογή της τριγωνικής ισοπαραμετρικής απεικόνισης θα χρησιμοποιήσουμε ένα νέο για τα μέχρι τώρα δεδομένα σύστημα συντεταγμένων, αυτό των επιφανειακών συντεταγμένων. Προκειμένου να γίνει ευκολότερα αντιληπτή η έννοια των επιφανειακών συντεταγμένων θα χρησιμοποιήσουμε το παρακάτω σκαρίφημα (το οποίο μπορεί να βρει κανείς, μαζί με πολλά άλλα επί του θέματος, στο βιβλίο 'Finite Element Procedures' του Klaus-Jürgen Bathe [...])



Σχήμα 1.1

Για το τρίγωνο του σχήματος 1.1, η θέση ενός τυχαίου σημείου P εντός του τριγώνου ορίζεται με τις επιφανειακές του συντεταγμένες ως

$$L_1 = \frac{A_1}{A}, \quad L_2 = \frac{A_2}{A}, \quad L_3 = \frac{A_3}{A} \quad (1.30)$$

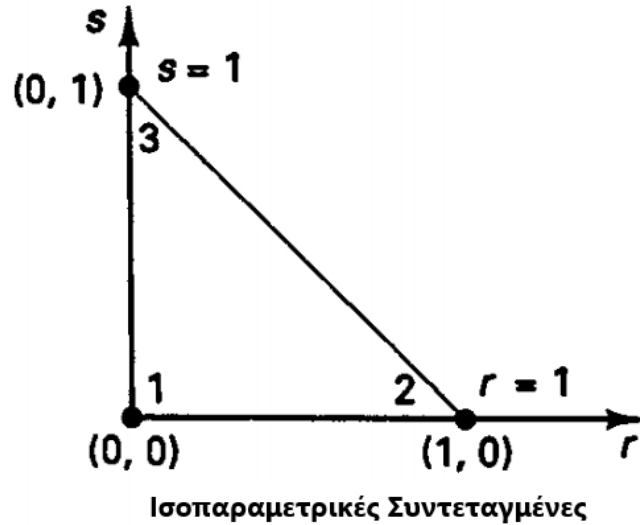
όπου τα εμβαδά A_i , $i=1,2,3$ ορίζονται όπως στο παραπάνω σκαρίφημα, και με A δίνεται το συνολικό εμβαδό του τριγώνου. Συγχρόνως, για τις επιφανειακές συντεταγμένες ισχύει η ιδιότητα

$$L_1 + L_2 + L_3 = 1 \quad (1.31)$$

όπως εύκολα μπορεί να αποδειχθεί αντικαθιστώντας τις επιφανειακές συντεταγμένες από την σχέση (1.30). Παρατηρεί κανείς πως για τον ορισμό της θέσης ενός σημείου P στην επιφάνεια του τριγώνου αρκούν οι δύο εκ των επιφανειακών συντεταγμένων, ήτοι οι L_1 και L_2 , αφού λόγω της ιδιότητας της σχέσεως (1.31) η τρίτη επιφανειακή συντεταγμένη θα δίνεται ως $L_3=1-L_1-L_2$. Χρήση της ιδιότητας αυτής θα γίνει στην πορεία κατά τον σχηματισμό και την διαφορίση των συναρτήσεων σχήματος του εξακομβικού στοιχείου πλάκας που θα χρησιμοποιηθεί προκειμένου να απλοποιηθούν οι προκύπτουσες εξισώσεις, αλλά προς το παρόν θα αγνοήσουμε την ιδιότητα αυτή στους υπόλοιπους ορισμούς των συντεταγμένων που θα δοθούν. Έτσι, οι Καρτεσιανές συντεταγμένες x, y του τυχαίου σημείου P εντός του τριγώνου δίνονται ως

$$x = L_1x_1 + L_2x_2 + L_3x_3 \quad (1.32)$$

$$y = L_1y_1 + L_2y_2 + L_3y_3 \quad (1.33)$$

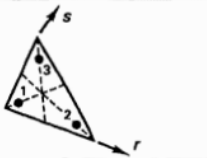

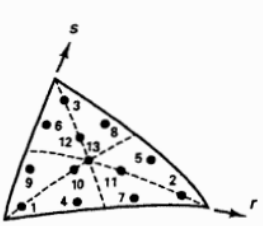


Σχήμα 1.2

Χρησιμοποιώντας το ισοπαραμετρικό σύστημα απεικόνισης, δηλαδή ανάγοντας την επιφάνεια του υπό μελέτη τριγωνικού στοιχείου σε αυτήν ενός ισοσκελούς ορθογώνιου τριγώνου με κορυφές τα σημεία 1(0,0), 2(1,0) και 3(0,1) όπως φαίνεται στο παραπάνω σκαρίφημα(Σχήμα 1.2), οι επιφανειακές συντεταγμένες θα δίνονται ως

$$L_1 = 1 - r - s, \quad L_2 = r, \quad L_3 = s \quad (1.34)$$

Έτσι, μπορούμε εύκολα και γρήγορα να βρούμε για κάθε ισοπαραμετρικό σημείο τις επιφανειακές του συντεταγμένες. Αντίστοιχοί πίνακες με τα σημεία και τα βάρη για την εκτέλεση της Gauss αριθμητικής ολοκλήρωσης σε συνήθεις ορθογωνικές ή παραπλήσιες γεωμετρίες υπάρχουν και για τριγωνικά στοιχεία. Επιλέγοντας την τάξη της αριθμητικής ολοκλήρωσης προκύπτει από πίνακα ο αριθμός των σημείων και τα αντίστοιχά τους βάρη. Οι σχέσεις που δίνουν τις θέσεις των ισοπαραμετρικών σημείων καθώς και τα βάρη των σημείων αυτών δεν παρατίθενται εδώ, μιας και δεν αποτελούν αντικείμενο της διπλωματικής αυτής. Παρόλα αυτά στην περίπτωση που επιθυμεί κανείς να εντρυφήσει περαιτέρω επί της αριθμητικής ολοκλήρωσης Gauss σε τριγωνικές γεωμετρίες μπορεί να τις βρει στο βιβλίο του κ.Παπαδρακάκη 'Ανάλυση Φορέων με την Μέθοδο των Πεπερασμένων Στοιχείων' ή και στο βιβλίο του καθηγητή Klaus-Jürgen Bathe 'Finite Element Procedures,' το βιβλίο στο οποίο ανήκει και ο παρακάτω πίνακας των ισοπαραμετρικών τριγωνικών σημείων και αντίστοιχών τους συντελεστών βάρους.

Σημεία ολοκλήρωσης	Βαθμός ακρίβειας	Διάταξη σημείων	Συντεταγμένη r	Συντεταγμένη s	Βάρη w
3	2		$r_1 = 0.16666\ 66666\ 667$ $r_2 = 0.66666\ 66666\ 667$ $r_3 = r_1$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$	$w_1 = 0.33333\ 33333\ 333$ $w_2 = w_1$ $w_3 = w_1$
7	5		$r_1 = 0.10128\ 65073\ 235$ $r_2 = 0.79742\ 69853\ 531$ $r_3 = r_1$ $r_4 = 0.47014\ 20641\ 051$ $r_5 = r_4$ $r_6 = 0.05971\ 58717\ 898$ $r_7 = 0.33333\ 33333\ 333$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$ $s_4 = r_6$ $s_5 = r_4$ $s_6 = r_4$ $s_7 = r_7$	$w_1 = 0.12593\ 91805\ 448$ $w_2 = w_1$ $w_3 = w_1$ $w_4 = 0.13239\ 41527\ 885$ $w_5 = w_4$ $w_6 = w_4$ $w_7 = 0.225$
13	7		$r_1 = 0.06513\ 01029\ 022$ $r_2 = 0.86973\ 97941\ 956$ $r_3 = r_1$ $r_4 = 0.31286\ 54960\ 049$ $r_5 = 0.63844\ 41885\ 698$ $r_6 = 0.04869\ 03154\ 253$ $r_7 = r_5$ $r_8 = r_4$ $r_9 = r_6$ $r_{10} = 0.26034\ 59660\ 790$ $r_{11} = 0.47930\ 80678\ 419$ $r_{12} = r_{10}$ $r_{13} = 0.33333\ 33333\ 333$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$ $s_4 = r_6$ $s_5 = r_4$ $s_6 = r_5$ $s_7 = r_6$ $s_8 = r_5$ $s_9 = r_4$ $s_{10} = r_{10}$ $s_{11} = r_{10}$ $s_{12} = r_{11}$ $s_{13} = r_{13}$	$w_1 = 0.05334\ 72356\ 088$ $w_2 = w_1$ $w_3 = w_1$ $w_4 = 0.07711\ 37608\ 903$ $w_5 = w_4$ $w_6 = w_4$ $w_7 = w_4$ $w_8 = w_4$ $w_9 = w_4$ $w_{10} = 0.17561\ 52574\ 332$ $w_{11} = w_{10}$ $w_{12} = w_{10}$ $w_{13} = -0.14957\ 00444\ 677$

Πίνακας 1.1: Σημεία αριθμητικής ολοκλήρωσης Gauss τριγωνικής επιφάνειας

Τα στοιχεία του παραπάνω πίνακα για την εκτέλεση της αριθμητικής ολοκλήρωσης του μητρώου δυσκαμψίας για το τριγωνικό στοιχείο που χρησιμοποιείται διατίθενται για τον κώδικα μέσω της συναρτήσεως TriGauss, που παρατίθεται στο Παράρτημα Α μαζί με τις υπόλοιπες συναρτήσεις υπεύθυνες για την υλοποίηση του πεπερασμένου στοιχείου.

1.3 Θεωρία Λεπτών πλακών-κελυφών

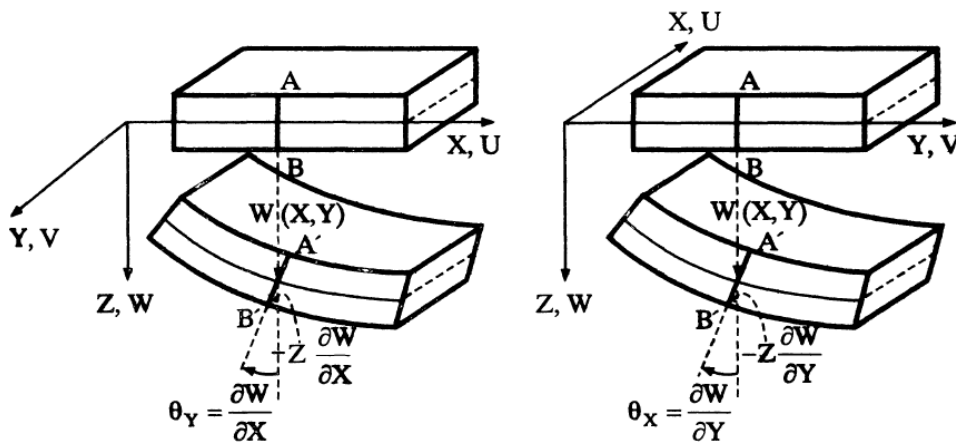
Η βασική θεωρία για τα στοιχεία πλάκας ή κελύφους με την οποία έρχεται αντιμέτωπος κανείς κατά την ενασχόληση του με τέτοιου είδους στοιχεία είναι η θεωρία Kirchhoff, η οποία αναφέρεται στην συμπεριφορά πεπερασμένων στοιχείων πλάκας ή κελύφους αγνοώντας την διατμητική παραμόρφωση στην οποία μπορεί αυτά να υπόκεινται. Η υπόθεση αυτή έχει νόημα στην περίπτωση που το πάχος των στοιχείων αυτών είναι επαρκώς μικρό ώστε η συνεισφορά της διατμητικής δυσκαμψίας να είναι αμελητέα σε σύγκριση με την συνεισφορά της μεμβρανικής και της καμπτικής συμπεριφοράς των στοιχείων. Κάπως έτσι προκύπτει και το όνομα της θεωρίας, ως θεωρία λεπτών πλακών-κελυφών, ή εναλλακτικά ως θεωρία Kirchhoff από το όνομα του θεμελιωτή αυτής.

Για να ισχύει η θεωρία αυτή, επιπλέον του περιορισμού του πάχους των στοιχείων σε σύγκριση με τις άλλες διαστάσεις, απαιτείται να μην εμφανίζονται μεγάλες εγκάρσιες παραμορφώσεις για τα στοιχεία, γεγονός που θα αναιρούσε την υπόθεση αμελητέας συνεισφορά της διατμητικής λειτουργίας στην ακαμψία του στοιχείου, πέραν του τεράστιου σφάλματος κατά την θεώρηση γραμμικότητας γεωμετρίας για το πρόβλημα. Ακόμα, με την υιοθέτηση της θεωρίας αυτής υιοθετείται και η υπόθεση πως επιφάνειες κάθετες στην μέση επιφάνεια της πλάκας παραμένουν κάθετες σε αυτήν και μετά την παραμόρφωση αυτής, υπόθεση που απορρέει ουσιαστικά από την θεώρηση αμελητέων διατμητικών παραμορφώσεων που ήδη αναφέρθηκε. Τέλος, στην μέση επιφάνεια της πλάκας οι τάσεις σ_x , σ_y και τ_{xy} θεωρούνται μηδενικές, πέραν της τάσεως σ_z που θεωρείται μηδενική ως μέρος του ορισμού των στοιχείων ως στοιχεία πλάκας-κελύφους, αφού δε μελετάται η παραμόρφωση αυτών κατά την διάσταση του πάχους των.

Από τα παραπάνω είναι εμφανές πως η θεωρία λεπτών πλακών-κελυφών Kirchhoff είναι σε μεγάλο βαθμό ανάλογη με την κλασσική θεωρία Euler-Bernoulli των δοκών, κατά την οποία επίσης αμελούνται οι καθύψος διατμητικές παραμορφώσεις των στοιχείων, με όλα τα παρελκόμενα που μπορεί να έχει μια τέτοια υπόθεση για την συμπεριφορά των στοιχείων. Η παραδοχή των διατήρησης της καθετότητας επιφανειών ως προς την μέση επιφάνεια των στοιχείων ακόμα και μετά την παραμόρφωση, σε συνδυασμό με την παραδοχή μηδενικής τάσεως σ_z επιτρέπει την θεώρηση σταθερής βύθισης για κάθε σημείο μιας γραμμής κάθετης στην μέση επιφάνεια της πλάκας(ή του κελύφους). Έτσι, μπορούμε να γράψουμε

$$W(X,Y,Z)=W(X,Y) \quad (1.35)$$

όπου $W(X,Y)$ είναι η βύθιση ενός σημείου με συντεταγμένες X και Y της μέσης επιφάνειας.



Σχήμα 1.3: Κάμψη πλάκας κατά την κλασσική θεωρία Kirchhoff

Η παραδοχή περί καθετότητας επιφανειών μετά την παραμόρφωση των στοιχείων, ή οποία φαίνεται και σχηματικώς στο παραπάνω σκαρίφημα(βιβλίο 'Ανάλυση Φορέων με την Μέθοδο των Πεπερασμένων Στοιχείων' του καθηγητή κ.Παπαδρακάκη), επιτρέπει την διατύπωση των μετατοπίσεων τυχαίας θέσεως εντός του στοιχείου ως

$$U(X,Y,Z) = -Z \frac{\partial W}{\partial X}(X,Y) \quad (1.36)$$

$$V(X,Y,Z) = -Z \frac{\partial W}{\partial Y}(X,Y) \quad (1.37)$$

Από τις σχέσεις (1.35),(1.36) και (1.37) μπορεί πλέον να προσδιοριστεί η μετατόπιση οποιοδήποτε σημείου του στοιχείου. Βάσει αυτών οι ανηγμένες παραμορφώσεις διατυπώνονται ως

$$\epsilon_x = \frac{\partial U}{\partial X} = -Z \frac{\partial^2 W}{\partial X^2}$$

$$\epsilon_y = \frac{\partial V}{\partial Y} = -Z \frac{\partial^2 W}{\partial Y^2}$$

$$\varepsilon_z = \frac{\partial W}{\partial Z} = 0 \tag{1.38}$$

$$\gamma_{xy} = \frac{\partial U}{\partial Y} + \frac{\partial V}{\partial X} = -2Z \frac{\partial^2 W}{\partial X \partial Y}$$

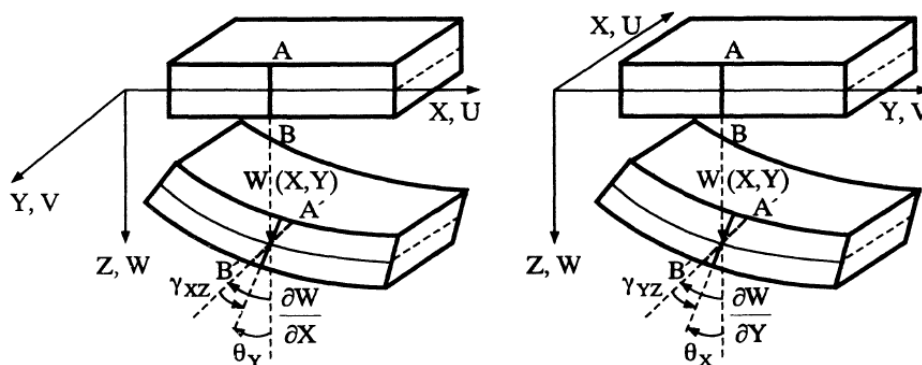
$$\gamma_{yz} = \frac{\partial V}{\partial Z} + \frac{\partial W}{\partial Y} = 0$$

$$\gamma_{xz} = \frac{\partial U}{\partial Z} + \frac{\partial W}{\partial X} = 0$$

Οι παραπάνω ανηγμένες παραμορφώσεις αποτελούν τις διέπουσες εξισώσεις για ένα στοιχείο που υπακούει στην θεωρία λεπτών πλακών κελυφών Kirchhoff.

1.4 Θεωρία παχιών πλακών-κελυφών

Στις περιπτώσεις που το πάχος των στοιχείων ξεπερνάει κάποιο όριο, πέραν του οποίου δεν μπορούν να ισχύουν πλέον οι παραδοχές της θεωρίας Kirchhoff, εφαρμόζεται η θεωρία παχιών πλακών-κελυφών Reissner-Mindlin. Κατά την θεωρία αυτή συνυπολογίζεται η διατμητική δράση του πεπερασμένου κατά την παραμόρφωση του στην δυσκαμψία που αυτό προσδίδει στο συνολικό μητρώο στιβαρότητας. Πλέον, επίπεδες επιφάνειες που είναι κάθετες στην μέση επιφάνεια της πλάκας ή του κελύφους πριν την παραμόρφωση δε παραμένουν κάθετες αναγκαστικά και μετά από αυτήν, αλλά διατηρούν την επιπεδότητά τους χωρίς να λαμβάνεται υπόψη καμπύλωση τους λόγω της παραμόρφωσης. Οι διατμητικές παραμορφώσεις γ_{xz} και γ_{yz} θεωρούνται ομοιόμορφα κατανεμημένες κατά την διεύθυνση του ύψους της πλάκας, και εκφράζονται με την απόκλιση του παραμορφωμένου κατακόρυφου επιπέδου της πλάκας από το κάθετο επίπεδο στην μέση επιφάνεια της μετά την παραμόρφωση, το οποίο αποτελεί ουσιαστικά και το επίπεδο μετά την παραμόρφωση κατά την θεωρία Kirchhoff. Παραστατικά τα επίπεδα αυτά πριν και μετά την παραμόρφωση φαίνονται στο παρακάτω σκαρίφημα(βιβλίο 'Ανάλυση Φορέων με την Μέθοδο των Πεπερασμένων Στοιχείων' του καθηγητή κ.Παπαδρακάκη).



Σχήμα 1.4: Κάμψη πλακών κατά την θεωρία Reissner-Mindlin

Για την βύθιση W σε μια θέση X, Y της μέσης επιφάνειας της πλάκας ισχύει και πάλι η σχέση (1.35) σύμφωνα με την οποία η βύθιση είναι σταθερή καθ ύψος, γεγονός το οποίο οφείλεται στην υπόθεση επιπεδότητας των διατομών, ασχέτως του αν αυτές παραμένουν κάθετες ή όχι στην μέση επιφάνεια. Άρα μπορούμε και πάλι να γράψουμε

$$W(X, Y, Z) = W(X, Y) \quad (1.39)$$

για την βύθιση W οποιοδήποτε X, Y, Z σημείου του στοιχείου. Οι μετατοπίσεις U και V του στοιχείου θα δίνονται ως

$$\begin{aligned} U(X, Y, Z) &= -Z\theta_Y(X, Y) \\ V(X, Y, Z) &= -Z\theta_X(X, Y) \end{aligned} \quad (1.40)$$

Λόγω της μη αμέλησης της διατμητικής συμπεριφοράς κατά τον υπολογισμό των ανηγμένων παραμορφώσεων του στοιχείου, αυτές θα δίνονται ως

$$\begin{aligned} \varepsilon_x &= -Z \frac{\partial \theta_Y}{\partial X} \\ \varepsilon_Y &= -Z \frac{\partial \theta_X}{\partial Y} \\ \varepsilon_Z &= \frac{\partial W}{\partial Z} = 0 \\ \gamma_{XY} &= -Z \left(\frac{\partial \theta_Y}{\partial Y} + \frac{\partial \theta_X}{\partial X} \right) \\ \gamma_{YZ} &= -\theta_X + \frac{\partial W}{\partial Y} \\ \gamma_{XZ} &= -\theta_Y + \frac{\partial W}{\partial X} \end{aligned} \quad (1.41)$$

Οι παραπάνω ανηγμένες παραμορφώσεις αποτελούν τις διέπουσες εξισώσεις για ένα στοιχείο πλάκας ή κελύφους που υπακούει στην θεωρία Reissner-Mindlin. Όπως στην περίπτωση της θεωρίας Kirchhoff έγινε ο παραλληλισμός της ως αντίστοιχη της θεωρίας δοκού Euler-Bernoulli, έτσι και η θεωρία Reissner-Mindlin παχίων πλακών-κελυφών μπορεί να παραλληλιστεί με την θεωρία δοκού Timoshenko. Στοιχεία όπως ο συνυπολογισμός των διατμητικών παραμορφώσεων και η μη διατήρηση της καθετότητας των διατομών έπειτα από την παραμόρφωση των στοιχείων αποτελούν παραδοχές που συναντώνται και στην υιοθέτηση της θεωρίας Timoshenko για δοκούς, σε αντιδιαστολή με την θεωρία Euler-Bernoulli που αγνοεί διατμητικά φαινόμενα κατά την παραμόρφωση.

Βάσει των παραπάνω σχέσεων ανηγμένων παραμορφώσεων και μετατοπίσεων συντίθεται το μητρώο παραμορφώσεως του στοιχείου, ο πολλαπλασιασμός του οποίου με τις επικόμβιες μετατοπίσεις μας επιτρέπει τον υπολογισμό των παραπάνω παραμορφώσεων. Αναλυτικότερα η μεθοδολογία αυτή θα παρουσιαστεί στο επόμενο κεφάλαιο, όπου και θα γίνει η σύνθεση των μητρώων παραμορφώσεως για κάθε μια από τις τρεις λειτουργίες του εξακομβικού τριγωνικού στοιχείου πλάκας.

ΚΕΦΑΛΑΙΟ 2^ο

ΕΞΑΚΟΜΒΙΚΟ ΤΡΙΓΩΝΙΚΟ ΣΤΟΙΧΕΙΟ ΠΛΑΚΑΣ ΒΑΣΙΣΜΕΝΟ ΣΤΗΝ ΘΕΩΡΙΑ ΠΑΧΙΩΝ ΠΛΑΚΩΝ-ΚΕΛΥΦΩΝ

2.1 Γενικά

Προκειμένου να γίνουν οι στατικές αναλύσεις των φορέων και να μπορέσουν να εφαρμοσθούν επ' αυτών οι μέθοδοι βελτιστοποίησης τοπολογίας που περιγράφονται παρακάτω δημιουργήθηκε ο κώδικας του Παραρτήματος Α. Στο παρών κεφάλαιο επεξηγούνται οι εξισώσεις και τα καταστατικά μοντέλα στα οποία έχει βασισθεί ο εν λόγω κώδικας, καθώς επίσης παρουσιάζεται βήμα βήμα ο κώδικας του στοιχείου. Μπορεί να βρει κανείς τις εξισώσεις που χρησιμοποιούνται, όπως επίσης και συναφή θεωρητικά στοιχεία, στο βιβλίο *Finite Elements for Solids, Fluids and Optimization* του G. A. Mohr []. Το τριγωνικό αυτό στοιχείο βασίζεται στην θεωρία παχιών πλακών και κελυφών Reissner-Mindlin που έχει περιγραφεί συνοπτικώς στο προηγούμενο κεφάλαιο. Η μηχανική συμπεριφορά ενός τριγωνικού πεπερασμένου στοιχείου κατά Reissner-Mindlin μπορεί να αναλυθεί σε τρεις επιμέρους συνιστώσες, αυτήν που οφείλεται στην μεμβρανική λειτουργία, αυτήν που οφείλεται στην καμπτική λειτουργία και τέλος αυτήν που οφείλεται στην διατμητική λειτουργία του στοιχείου. Η επαλληλία των παραπάνω συνιστωσών και η αριθμητική τους ολοκλήρωση στην επιφάνεια του τριγώνου επιτρέπει τον υπολογισμό του μητρώου ακαμψίας του τριγωνικού στοιχείου, που είναι άλλωστε και το ζητούμενο ώστε να μπορέσουμε στην συνέχεια να δημιουργήσουμε το μητρώο ακαμψίας του φορέα στο καθολικό του σύστημα εφαρμόζοντας τους απαραίτητους μετασχηματισμούς. Στη συνέχεια του παρόντος κεφαλαίου θα δούμε τον σχηματισμό των συνιστωσών που θα συνθέσουν το μητρώο δυσκαμψίας ενός εξακομβικού τριγωνικού πεπερασμένου στοιχείου με την θεωρία παχιών πλακών κελυφών.

2.2 Μεμβρανική λειτουργία

Η μεμβρανική λειτουργία του τριγώνου αφορά την συμπεριφορά του ως ενός επίπεδης έντασης παραμόρφωσης στοιχείου. Σε κάθε έναν από τους κόμβους αντιστοιχούν δυο εντός επιπέδου βαθμοί ελευθερίας, κατά τους τοπικούς x και y , απαριθμώντας έτσι συνολικά δώδεκα εντός επιπέδου βαθμούς ελευθερίας για κάθε στοιχείο. Το μοντέλο που χρησιμοποιείται είναι αυτό του γραμμικής παραμορφώσεως τριγώνου που πρώτο παρουσιάστηκε από τον Argyris [] και Fraeijns de Veubeke και το οποίο στο εξής θα αναφέρεται με την συντομογραφία LST (Linear Strain Triangle).

Οι συναρτήσεις σχήματος με χρήση των επιφανειακών συντεταγμένων που παρουσιάστηκαν στο κεφάλαιο 1.2.2 είναι:

$$\begin{aligned}
 N_1 &= L_1(2L_1 - 1) \\
 N_2 &= L_2(2L_2 - 1) \\
 N_3 &= L_3(2L_3 - 1) \\
 N_4 &= 4L_1L_2
 \end{aligned}
 \tag{2.1}$$

$$N_5 = 4L_2L_3$$

$$N_6 = 4L_1L_3$$

Παρατηρώντας όμως πως $L_1 + L_2 + L_3 = 1$ οι παραπάνω σχέσεις μπορούν να γραφούν και μόνο συναρτήσει των L_1 και L_2 εφαρμόζοντας την αντικατάσταση $L_3 = 1 - L_1 - L_2$

Χρησιμοποιώντας τον κανόνα της αλυσίδας η παραμόρφωση ε_x παραδείγματος χάριν μπορεί να υπολογιστεί ως εξής:

$$\varepsilon_x = \frac{\partial u}{\partial x} = \frac{\partial u}{\partial L_1} \frac{\partial L_1}{\partial x} + \frac{\partial u}{\partial L_2} \frac{\partial L_2}{\partial x} \quad (2.2)$$

Και αντικαθιστώντας μέσω του μετασχηματισμού $u = [N]\{d\}$ έχουμε:

$$\varepsilon_x = \left[\left\{ \frac{\partial N}{\partial L_1} \right\}^T \frac{\partial L_1}{\partial x} + \left\{ \frac{\partial N}{\partial L_2} \right\}^T \frac{\partial L_2}{\partial x} \right] \{d\} \quad (2.3)$$

Όπου ο όρος $\left\{ \frac{\partial N}{\partial L_1} \right\}$ συμβολίζει το $\left\{ \frac{\partial N_1}{\partial L_1}, \frac{\partial N_1}{\partial L_1}, \dots, \frac{\partial N_6}{\partial L_1} \right\}$.

Οι απαιτούμενες τοπικές παράγωγοι των συναρτήσεων σχήματος ως προς τις επιφανειακές συντεταγμένες L_1 και L_2 μπορούν εύκολα να εξαχθούν από τις σχέσεις (2.1) και να γραφούν υπό μορφή μητρώου S , εφαρμόζοντας την αντικατάσταση $L_3 = 1 - L_1 - L_2$. Στη συνέχεια μπορούμε να σχηματίσουμε το μητρώο μετασχηματισμού S των τοπικών παραγώγων ως προς τις συντεταγμένες L_1 και L_2 υπό την μορφή

$$S = \begin{bmatrix} \left\{ \frac{\partial N}{\partial L_1} \right\}^T \\ \left\{ \frac{\partial N}{\partial L_2} \right\}^T \end{bmatrix}$$

$$= \begin{bmatrix} 4L_1 - 1 & 0 & 4L_1 + 4L_2 - 3 & 4L_2 & -4L_2 & 4 - 8L_1 - 4L_2 \\ 0 & 4L_2 - 1 & 4L_1 + 4L_2 - 3 & 4L_1 & 4 - 4L_1 - 8L_2 & -4L_1 \end{bmatrix} \quad (2.4)$$

Γράφοντας τους απαραίτητους μετασχηματισμούς σε Καρτεσιανές παραγώγους ως προς τα τοπικά x και y , και χρησιμοποιώντας και πάλι τον κανόνα της αλυσίδας κατά την παραγωγή, προκύπτει

$$\begin{bmatrix} \left\{ \frac{\partial N}{\partial x} \right\}^T \\ \left\{ \frac{\partial N}{\partial y} \right\}^T \end{bmatrix} = \begin{bmatrix} \frac{\partial L_1}{\partial x} & \frac{\partial L_2}{\partial x} \\ \frac{\partial L_1}{\partial y} & \frac{\partial L_2}{\partial y} \end{bmatrix} \begin{bmatrix} \left\{ \frac{\partial N}{\partial L_1} \right\}^T \\ \left\{ \frac{\partial N}{\partial L_2} \right\}^T \end{bmatrix} = J^{-1} S \quad (2.5)$$

Από τους ορισμούς των επιφανειακών συντεταγμένων των σχέσεων (1.32) και (1.33) το αντίστροφο μητρώο J^{-1} του Ιακωβιανού μετασχηματισμού προκύπτει ως

$$J^{-1} = \frac{1}{2\Delta} \begin{bmatrix} -y_{32} & -y_{13} \\ x_{32} & x_{13} \end{bmatrix} \quad (2.6)$$

Όπου το Δ συμβολίζει το εμβαδόν της επιφάνειας του τριγώνου και οι υπόλοιποι όροι του μητρώου δίνονται από τις σχέσεις $y_{32} = y_3 - y_2$, $y_{13} = y_1 - y_3$, $x_{32} = x_3 - x_2$ και $x_{13} = x_1 - x_3$.

Επομένως αποκτούμε ένα μητρώο μετασχηματισμού των Καρτεσιανών παραγώγων υπό την μορφή

$$G = J^{-1}S = \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & g_{16} \\ g_{21} & g_{22} & g_{23} & g_{24} & g_{25} & g_{26} \end{bmatrix} \quad (2.7)$$

το οποίο μπορεί να υπολογιστεί αριθμητικώς σε κάθε σημείο της επιφάνειας του τριγώνου με χρήση των συντεταγμένων (L_1, L_2) . Χρησιμοποιώντας τους όρους του παραπάνω μητρώου (σχέση 2.7) το μητρώο παραμορφώσεως για την μεμβρανική λειτουργία υπολογίζεται ως

$$B_m = \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{pmatrix} = \begin{bmatrix} g_{11} & 0 & g_{12} & 0 & g_{13} & 0 & g_{14} & 0 & g_{15} & 0 & g_{16} & 0 \\ 0 & g_{21} & 0 & g_{22} & 0 & g_{23} & 0 & g_{24} & 0 & g_{25} & 0 & g_{26} \\ g_{21} & g_{11} & g_{22} & g_{12} & g_{23} & g_{13} & g_{24} & g_{14} & g_{25} & g_{15} & g_{26} & g_{16} \end{bmatrix} \{d\} \quad (2.8)$$

Το παραπάνω μητρώο υπολογίζεται για κάθε θέση αριθμητικής ολοκλήρωσης και πολλαπλασιάζεται με το παρακάτω μητρώο ελαστικότητας που αντιστοιχεί στην μεμβρανική συμπεριφορά

$$D_m = \frac{E \cdot t}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix} \quad (2.9)$$

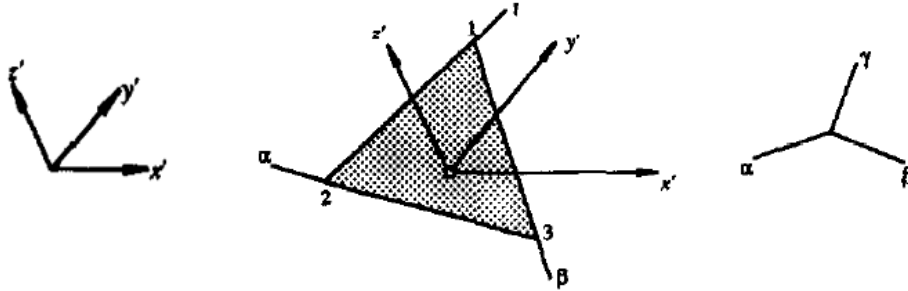
Όπου με E συμβολίζεται το μέτρο ελαστικότητας κατά Young, με t το πάχος του στοιχείου πλάκας και με ν ο συντελεστής Poisson.

2.3 Καμπτική λειτουργία

Έχοντας συμπεριλάβει την ακαμψία των βαθμών ελευθερίας u και v μέσω της μεμβρανικής συμπεριφοράς του στοιχείου, απομένει να ληφθούν υπόψη οι ακαμψίες του στοιχείου κατά τους τοπικούς βαθμούς ελευθερίας w, ϕ_x και ϕ_y . Με επαλληλία της καμπτικής και διατμητικής συμπεριφοράς του στοιχείου υπολογίζεται και η εν λόγω ακαμψία στις περιπτώσεις παχιών πλακών-κελυφών. Κατά τον σχηματισμό του μητρώου παραμορφώσεως που αντιστοιχεί στην καμπτική λειτουργία, αντί των κλασικών παραμορφώσεων $\epsilon_x, \epsilon_y, \epsilon_{xy}$ που συνθέτουν τα συνήθη μητρώα παραμορφώσεως πεπερασμένων στοιχείων, γίνεται χρήση των καμπυλοτήτων της επιφάνειας

$$\kappa_x = \frac{\partial \varphi_x}{\partial x}, \quad \kappa_y = \frac{\partial \varphi_y}{\partial y}, \quad \kappa_{xy} = \frac{\partial \varphi_x}{\partial y} + \frac{\partial \varphi_y}{\partial x} \quad (2.10)$$

Προκειμένου να διευκολυνθούν οι υπολογισμοί μεταφέρουμε τις κλίσεις και τις καμπυλότητες από το τοπικό Καρτεσιανό σύστημα του τριγώνου στο φυσικό του σύστημα. Το σύστημα αυτό ορίζεται από τρεις άξονες, καθένας εκ των οποίων είναι παράλληλος σε μια από τις πλευρές του τριγώνου.



Σχήμα 2.1: Τοπικό Καρτεσιανό(αριστερά) και φυσικό(δεξιά) σύστημα τριγώνου

Προσδιορίζοντας τις φυσικές καμπυλότητες του παράλληλα σε κάθε πλευρά του τριγώνου έχουμε

$$\begin{aligned} \kappa_\alpha &= \frac{\partial \varphi_\alpha}{\partial s_\alpha} = \frac{\partial \varphi_\alpha / \partial L_2 - \partial \varphi_\alpha / \partial L_1}{L_{21}} \\ \kappa_b &= \frac{\partial \varphi_b}{\partial s_b} = \frac{\partial \varphi_b / \partial L_3 - \partial \varphi_b / \partial L_2}{L_{32}} \\ \kappa_c &= \frac{\partial \varphi_c}{\partial s_c} = \frac{\partial \varphi_c / \partial L_1 - \partial \varphi_c / \partial L_3}{L_{13}} \end{aligned} \quad (2.11)$$

όπου $\varphi_\alpha, \varphi_b, \varphi_c$ οι κλίσεις εκφρασμένες επί των πλευρών του τριγώνου, s_α, s_b, s_c συντεταγμένες που 'τρέχουν' επί των πλευρών του τριγώνου και L_{21}, L_{32}, L_{13} τα μήκη των πλευρών 1-2, 2-3 και 3-1 αντίστοιχα.

Οι κλίσεις στο φυσικό σύστημα δίνονται από τις σχέσεις

$$\varphi_\alpha = c_{ax}\varphi_x + c_{ay}\varphi_y, \quad \varphi_b = c_{bx}\varphi_x + c_{by}\varphi_y, \quad \varphi_c = c_{cx}\varphi_x + c_{cy}\varphi_y \quad (2.12)$$

όπου τα συνημίτονα διεύθυνσης είναι

$$\begin{aligned}
 c_{ax} &= \frac{x_{21}}{L_{21}} = \frac{x_2 - x_1}{L_{21}}, & c_{ay} &= \frac{y_{21}}{L_{21}} = \frac{y_2 - y_1}{L_{21}} \\
 c_{bx} &= \frac{x_{32}}{L_{32}} = \frac{x_3 - x_2}{L_{32}}, & c_{by} &= \frac{y_{32}}{L_{32}} = \frac{y_3 - y_2}{L_{32}} \\
 c_{cx} &= \frac{x_{13}}{L_{13}} = \frac{x_1 - x_3}{L_{13}}, & c_{cy} &= \frac{y_{13}}{L_{13}} = \frac{y_1 - y_3}{L_{13}}
 \end{aligned} \quad (2.13)$$

Οι παραμορφώσεις που οφείλονται στην καμπτική συμπεριφορά δίνονται ως

$$\{\varepsilon\} = \{\kappa_a, \kappa_b, \kappa_c\} = B_b\{d\} \quad (2.14)$$

Όπου το μητρώο παραμορφώσεως $B_{b,n}$ μπορεί να δοθεί σε συνοπτική μορφή

$$B_{b,n} = \begin{bmatrix} \{0\}^T & c_{ax}\{\dot{f}_a\}^T & c_{ay}\{\dot{f}_a\}^T \\ \{0\}^T & c_{bx}\{\dot{f}_b\}^T & c_{by}\{\dot{f}_b\}^T \\ \{0\}^T & c_{cx}\{\dot{f}_c\}^T & c_{cy}\{\dot{f}_c\}^T \end{bmatrix} \quad (2.15)$$

και

$$\begin{aligned}
 \{\dot{f}_a\} &= \frac{\partial\{f\}}{\partial s_a} = \frac{\{1 - 4L_1, 4L_2 - 1, 0, 4L_1 - 4L_2, 4L_3, -4L_3\}}{L_{21}} \\
 \{\dot{f}_b\} &= \frac{\partial\{f\}}{\partial s_b} = \frac{\{0, 1 - 4L_2, 4L_3 - 1, -4L_1, 4L_2 - 4L_3, 4L_1\}}{L_{32}} \\
 \{\dot{f}_c\} &= \frac{\partial\{f\}}{\partial s_c} = \frac{\{4L_1 - 1, 0, 1 - 4L_3, 4L_2, -4L_2, 4L_3 - 4L_1\}}{L_{13}}
 \end{aligned} \quad (2.16)$$

Οι καμπυλότητες που υπολογίζονται στο φυσικό σύστημα δεν μπορούν να χρησιμοποιηθούν απευθείας για τον υπολογισμό της ενέργειας παραμορφώσεως του στοιχείου καθώς επειδή υπερκαλύπτονται μεταξύ τους (για παράδειγμα τουλάχιστον δυο από τις πλευρές θα περιέχουν τον όρο κ_x). Προκειμένου να αντιμετωπιστεί αυτό το πρόβλημα οι καμπυλότητες $\kappa_a, \kappa_b, \kappa_c$ μετατρέπονται στις τεχνητές καμπυλότητες χ_a, χ_b, χ_c , οι οποίες δεν έχουν κανένα φυσικό νόημα, μέσω της σχέσης

$$\{\kappa_n\} = \begin{Bmatrix} \kappa_a \\ \kappa_b \\ \kappa_c \end{Bmatrix} = \begin{bmatrix} 1 & \cos^2\gamma & \cos^2\beta \\ \cos^2\gamma & 1 & \cos^2\alpha \\ \cos^2\beta & \cos^2\alpha & 1 \end{bmatrix} \begin{Bmatrix} \chi_a \\ \chi_b \\ \chi_c \end{Bmatrix} = W\{\chi_n\} \quad (2.17)$$

Οι γωνίες α, β, γ του τριγώνου διακρίνονται στο σχήμα 2.1 ως οι αντικριστές των πλευρών a, b και c αντίστοιχα και ο υπολογισμός τους μπορεί εύκολα να γίνει με χρήση του νόμου των συνημίτονων. Στη συνέχεια μετασχηματίζουμε και πάλι τις καμπυλότητες από το φυσικό σύστημα του τριγώνου πίσω στο τοπικό Καρτεσιανό σύστημα με χρήση της σχέσης

$$\{\kappa_n\} = \begin{bmatrix} c_{ax}^2 & c_{ay}^2 & 2c_{ax}c_{ay} \\ c_{bx}^2 & c_{by}^2 & 2c_{bx}c_{by} \\ c_{cx}^2 & c_{cy}^2 & 2c_{cx}c_{cy} \end{bmatrix} \begin{Bmatrix} \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{Bmatrix} = C\{\kappa_c\} \quad (2.18)$$

Με χρήση των παραπάνω μητρώων συνθέτουμε το μητρώο μετασχηματισμού από το φυσικό στο Καρτεσιανό σύστημα για να εκτελεσθεί η αριθμητική ολοκλήρωση ως

$$H = C^T W^{-1} \quad (2.19)$$

Άρα τελικώς το μητρώο παραμορφώσεως της καμπτικής λειτουργίας του στοιχείου που θα χρησιμοποιηθεί κατά την ολοκλήρωση είναι

$$B_b = HB_{b,n} \quad (2.20)$$

Το μητρώο ελαστικότητας που θα χρησιμοποιηθεί στους υπολογισμούς είναι το ίδιο με το αντίστοιχο της μεμβρανικής λειτουργίας, πολλαπλασιασμένο όμως επί $t^2/12$, δηλαδή

$$D_b = \frac{E t^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (2.21)$$

2.4 Διατμητική λειτουργία

Μέχρι αυτό το σημείο η διαδικασία και ο σχηματισμός των μητρώων παραμορφώσεως που ακολουθήθηκε καλύπτει την περίπτωση που αντιμετωπίζουμε λεπτές πλάκες ή κελύφη κατά την θεωρία Kirchhoff. Προκειμένου να δημιουργηθεί ένα στοιχείο παχιάς πλάκας απαιτείται και ο συνυπολογισμός των διατμητικών παραμορφώσεων που αναπτύσσονται κατά την διάσταση του πάχους, όπως επιτάσσει η θεωρία Reissner-Mindlin παχίων πλακών-κελυφών. Οι διατμητικές αυτές παραμορφώσεις υπολογίζονται ως

$$\gamma_x = \frac{\partial w}{\partial x} - \varphi_x, \quad \gamma_y = \frac{\partial w}{\partial y} - \varphi_y \quad (2.23)$$

Αντίστοιχα με την περίπτωση της καμπτικής συμπεριφοράς, οι φυσικές διατμητικές παραμορφώσεις παράλληλα στις πλευρές του τριγώνου ορίζονται ως

$$\gamma_a = \frac{\partial w}{\partial s_a} - \varphi_a = \frac{\partial w / \partial L_2 - \partial w / \partial L_1}{L_{21}} - \varphi_a$$

$$\gamma_b = \frac{\partial w}{\partial s_b} - \varphi_b = \frac{\partial w / \partial L_3 - \partial w / \partial L_2}{L_{32}} - \varphi_b \quad (2.24)$$

$$\gamma_c = \frac{\partial w}{\partial s_c} - \varphi_c = \frac{\partial w / \partial L_1 - \partial w / \partial L_3}{L_{13}} - \varphi_c$$

Οι διατμητικές παραμορφώσεις στο φυσικό σύστημα είναι

$$\{\varepsilon\} = \{\gamma_a, \gamma_b, \gamma_c\} = B_{s,n}\{d\} \quad (2.25)$$

όπου το μητρώο παραμορφώσεως δίνεται ως

$$B_{s,n} = \begin{bmatrix} \left\{ \dot{f}_a \right\}^T & -c_{ax}\{f\}^T & -c_{ay}\{f\}^T \\ \left\{ \dot{f}_b \right\}^T & -c_{bx}\{f\}^T & -c_{by}\{f\}^T \\ \left\{ \dot{f}_c \right\}^T & -c_{cx}\{f\}^T & -c_{cy}\{f\}^T \end{bmatrix} \quad (2.26)$$

και

$$\begin{aligned} \left\{ \dot{f}_a \right\} &= \frac{\partial \{f\}}{\partial s_a} = \frac{\{1 - 4L_1, 4L_2 - 1, 0, 4L_1 - 4L_2, 4L_3, -4L_3\}}{L_{21}} \\ \left\{ \dot{f}_b \right\} &= \frac{\partial \{f\}}{\partial s_b} = \frac{\{0, 1 - 4L_2, 4L_3 - 1, -4L_1, 4L_2 - 4L_3, 4L_1\}}{L_{32}} \\ \left\{ \dot{f}_c \right\} &= \frac{\partial \{f\}}{\partial s_c} = \frac{\{4L_1 - 1, 0, 1 - 4L_3, 4L_2, -4L_2, 4L_3 - 4L_1\}}{L_{13}} \end{aligned} \quad (2.27)$$

όπου

$$\{f\} = \{L_1(2L_1 - 1), L_2(2L_2 - 1), L_3(2L_3 - 1), 4L_1L_2, 4L_2L_3, 4L_3L_1\} \quad (2.28)$$

Στη συνέχεια, εκφράζουμε τις διατμητικές παραμορφώσεις του φυσικού συστήματος του τριγώνου συναρτήσει αυτών του Καρτεσιανού, με χρήση των συνημίτονων διεύθυνσης των πλευρών όπως φαίνεται στις σχέσεις

$$\gamma_a = c_{ax}\gamma_x + c_{ay}\gamma_y$$

$$\gamma_b = c_{bx}\gamma_x + c_{by}\gamma_y \quad (2.29)$$

$$\gamma_c = c_{cx}\gamma_x + c_{cy}\gamma_y$$

Το εμβαδόν της επιφάνειας του τριγώνου υπολογίζεται ως

$$\Delta = \frac{x_{21}y_{32} - x_{32}y_{21}}{2} = \frac{L_a L_b (c_{ax}c_{by} - c_{bx}c_{ay})}{2} \quad (2.30)$$

όπου με L_a, L_b, L_c συμβολίζονται τα μήκη των πλευρών, είναι δηλαδή ένας άλλος συμβολισμός για τα L_{21}, L_{32}, L_{13} αντίστοιχα. Συνεπώς οι εξισώσεις (2.29) μπορούν να γραφούν στη μορφή

$$\begin{aligned}\gamma_x &= \frac{L_a L_b (c_{by} \gamma_a - c_{ay} \gamma_b)}{2\Delta} \\ \gamma_y &= \frac{L_a L_b (-c_{bx} \gamma_a + c_{ax} \gamma_b)}{2\Delta}\end{aligned}\quad (2.31)$$

με κυκλική εναλλαγή των δεικτών a, b, c οι Καρτεσιανές διατμητικές παραμορφώσεις μπορούν να γραφούν

$$\begin{aligned}\gamma_x &= \frac{L_b L_c (c_{cy} \gamma_b - c_{by} \gamma_c)}{2\Delta} \\ \gamma_y &= \frac{L_b L_c (-c_{cx} \gamma_b + c_{bx} \gamma_c)}{2\Delta}\end{aligned}\quad (2.32)$$

και

$$\begin{aligned}\gamma_x &= \frac{L_c L_a (c_{ay} \gamma_c - c_{cy} \gamma_a)}{2\Delta} \\ \gamma_y &= \frac{L_c L_a (-c_{ax} \gamma_c + c_{cx} \gamma_a)}{2\Delta}\end{aligned}\quad (2.33)$$

Λαμβάνοντας τον μέσο όρο για τις προκύπτουσες διατμητικές παραμορφώσεις σύμφωνα με τις παραπάνω εξισώσεις έχουμε

$$\{\gamma_x, \gamma_y\} = T_G \{\gamma_a, \gamma_b, \gamma_c\} \quad (2.34)$$

όπου

$$T_G^T = \frac{1}{6\Delta} \begin{bmatrix} c_{by} L_a L_b - c_{cy} L_c L_a & c_{cx} L_c L_a - c_{bx} L_a L_b \\ c_{cy} L_b L_c - c_{ay} L_a L_b & c_{ax} L_a L_b - c_{cx} L_b L_c \\ c_{ay} L_c L_a - c_{by} L_b L_c & c_{bx} L_b L_c - c_{ax} L_c L_a \end{bmatrix} \quad (2.35)$$

Άρα το μετασχηματισμένο μητρώο παραμορφώσεως που οφείλεται στην διατμητική συμπεριφορά του στοιχείου θα δίνεται ως

$$B_s = [T_G B_{s,n}] \quad (2.36)$$

Το μητρώο ελαστικότητας που αντιστοιχεί στην διατμητική συμπεριφορά του στοιχείου είναι

$$D_s = c_s \frac{Et}{2(1-\nu^2)} \begin{bmatrix} (1-\nu) & 0 \\ 0 & (1-\nu) \end{bmatrix} \quad (2.37)$$

Όπου c_s ο συντελεστής για την αποφυγή του φαινομένου της διατμητικής υπερακαμψίας.

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Λαμβάνοντας υπόψη και τις παραπάνω σχέσεις έχουν διατυπωθεί τα απαραίτητα για τον σχηματισμό ενός πεπερασμένου στοιχείου έξι κόμβων με την θεωρία παχιών πλακών κελυφών. Την πρακτική αλγοριθμική εφαρμογή τους δούμε στο αμέσως επόμενο κεφάλαιο, όπου θα αναλυθεί ο κώδικας υπεύθυνος για την δημιουργία ενός τέτοιου πεπερασμένου.

2.5 Ανάλυση κώδικα σε matlab

Για τις ανάγκες εκπόνησης της διπλωματικής δημιουργήθηκε ο κώδικας για το τριγωνικό στοιχείο στην αντικειμενοστραφή γλώσσα προγραμματισμού C# προκειμένου να ενσωματωθεί στην γενικότερη δομή κώδικα MSolve που αφορά αναλύσεις πεπερασμένων στοιχείων. Παρόλα αυτά μιας και ο εν λόγω κώδικας απαιτεί γνώση της γλώσσας C# καθώς και μελέτη των κλάσεων που περιέχονται στη δομή MSolve, παρατίθεται στο Παράρτημα Α ο αντίστοιχος κώδικας για το τριγωνικό στοιχείο σε matlab, όντας πιο απλός στην κατανόηση και ανεξάρτητος στο μεγαλύτερο μέρος του από άλλα μέρη κώδικα. Οι συναρτήσεις που χρησιμοποιούνται από την συνάρτηση που είναι υπεύθυνη για την δημιουργία του μητρώου ακαμψίας επίσης παρατίθενται παρακάτω στο Παράρτημα Α.

Η συνάρτηση για την δημιουργία ενός στοιχείου καλείται με την εντολή `ElementMohrTri6(elgc,El,v,thick)`. Οι παράμετροι που χρειάζονται κατά την κλήση της είναι:

`elgc` : μητρώο 6x3 με τις συντεταγμένες των κόμβων του στοιχείου στο καθολικό σύστημα
`El` : μέτρο ελαστικότητας υλικού κατά Young
`v` : συντελεστής Poisson
`thick` : πάχος του τριγωνικού στοιχείου πλάκας

Γραμμές 8-9

Εκκίνηση του μητρώου ακαμψίας `Kelement` του στοιχείου διαστάσεων 30x30, έχοντας 5 βαθμούς ελευθερίας σε καθέναν από τους 6 κόμβους. Η μεταβλητή `hnel` συμβολίζει τους κόμβους που υπάρχουν στο στοιχείο, και τίθεται ίση με 6.

Γραμμές 11-20

Κλήση της συνάρτησης `dirvecT3` υπεύθυνης για την μετατροπή των συντεταγμένων των κόμβων από το καθολικό(μεταβλητή `elgc`) στο τοπικό σύστημα του τριγώνου(μεταβλητή `elc`) καθώς και για τον υπολογισμό των μοναδιαίων διανυσμάτων που προσδιορίζουν το τοπικό σύστημα του τριγώνου. Ο τρόπος που υπολογίζονται και η κατεύθυνση που έχουν αυτά τα διανύσματα επεξηγείται παρακάτω, στην παράγραφο σχετικά με την συνάρτηση `dirvecT3`. Επίσης μορφώνεται το μητρώο `Transform` με το οποίο θα πολλαπλασιάσουμε στο τελευταίο βήμα το μητρώο ακαμψίας που υπολογίστηκε στο τοπικό σύστημα για να το μεταφέρουμε στο καθολικό.

Γραμμές 22-24

Υπολογισμός των μηκών των πλευρών του τριγώνου στις μεταβλητές L21,L32,L13. Η συνάρτηση distance που χρησιμοποιείται για αυτό το σκοπό απλά παρατίθεται στο τέλος του Παραρτήματος Α και δεν θα αναλυθεί περαιτέρω.

Γραμμές 26-31

Υπολογισμός των συνημίτονων διεύθυνσης των πλευρών στο τοπικό σύστημα, όπως ορίζεται από τις σχέσεις (2.13). Η πρώτη γραμμή των μητρών c_x και c_y αναφέρεται στην $a(1-2)$ πλευρά, η δεύτερη στην $b(2-1)$ και η τρίτη στην $c(3-1)$.

Γραμμές 33-35

Υπολογισμός των γωνιών α , β , γ της επιφάνειας του τριγώνου με χρήση του νόμου των συνημίτονων.

Γραμμές 37-43

Υπολογισμός των μητρών W και C όπως αυτά ορίζονται από τις σχέσεις (2.17) και (2.18) αντίστοιχα. Βάσει των μόλις υπολογισμένων μητρών, από την σχέση (2.19) υπολογίζουμε το μητρώο H που θα μας χρειαστεί για να φέρουμε το μητρώο παραμορφώσεως καμπτικής συμπεριφοράς στην μορφή που θέλουμε.

Γραμμή 45

Υπολογισμός του εμβαδού της επιφάνειας του τριγώνου Delta βάσει της σχέσης (2.30)

Γραμμές 46-48

Υπολογισμός του μητρώου T_g βάσει της σχέσης (2.35)

Γραμμή 50

Δημιουργία των ισοπαραμετρικών σημείων (r,s) της τριγωνικής επιφάνειας στα οποία θα γίνει η αριθμητική ολοκλήρωση για να προκύψει το μητρώο ακαμψίας του στοιχείου. Το βάρος κάθε στοιχείου αποθηκεύεται στον πίνακα w . Επιλέγεται αυτομάτως από τον κώδικα η επιλογή τέταρτης τάξης αριθμητικής ολοκλήρωσης κατά Gauss, αν και θα μπορούσε να οριστεί ως μεταβλητή και να περνάει ως παράμετρος έναρξης στην συνάρτηση δημιουργίας του στοιχείου. Η συνάρτηση TriGauss που είναι υπεύθυνη για την δημιουργία των σημείων και των αντίστοιχων βαρών παρατίθεται επίσης στο Παράρτημα Α.

Γραμμή 52

Εκκίνηση της επανάληψης που υπολογίζει την ακαμψία σε κάθε σημείο (r,s) της επιφάνειας. Η επανάληψη κλείνει στην Γραμμή 100.

Γραμμές 53-55

Υπολογισμός των επιφανειακών συντεταγμένων L1,L2 και L3 του τριγώνου που αντιστοιχούν στο ισοπαραμετρικό σημείο (r,s) .

Γραμμές 57-62

Υπολογισμός των συναρτήσεων σχήματος N από την σχέση(2.1) καθώς και των παραγώγων τους στο φυσικό σύστημα όπως προκύπτουν από τις σχέσεις (2.27)και των παραγώγων τους ως προς τις επιφανειακές συντεταγμένες μέσω των σχέσεων (2.5)

Γραμμές 64-68

Υπολογισμός των μητρών S και του αντιστροφου J^{-1} του Ιακωβιανού μετασχηματισμού που απαιτούνται για τον υπολογισμό του μητρώου G , το οποίο με την σειρά του περιέχει τους όρους που συνθέτουν το μητρώο παραμορφώσεως για την μεμβρανική λειτουργία του στοιχείου.

Γραμμές 70-72

Σχηματισμός του μητρώου παραμορφώσεως B_m για την μεμβρανική συμπεριφορά του στοιχείου χρησιμοποιώντας τους όρους του μητρώου G , και τοποθετώντας τους στις θέσεις που υποδεικνύονται από την σχέση (2.8)

Γραμμές 74-82

Στις γραμμές 80 με 82 υπολογίζεται το μητρώο ελαστικότητας που αντιστοιχεί στην μεμβρανική συμπεριφορά του στοιχείου, όπως αυτό ορίζεται από την σχέση(2.9) Από την άλλη στις γραμμές 74-79 συντίθεται το μητρώο που αντιστοιχεί στις ελαστικές ιδιότητες του υλικού, με τις τρεις πρώτες γραμμές να αφορούν την καμπτική συμπεριφορά, και τις δυο επόμενες του πίνακα να αφορούν την διατμητική. Επίσης σε έναν πίνακα συντίθεται και το μητρώο παραμορφώσεως της καμπτικής και διατμητικής συμπεριφοράς, με τις τρεις πρώτες γραμμές να αφορούν και πάλι την καμπτική λειτουργία και τις δυο επόμενες την διατμητική.

Γραμμές 84- 93

Ουσιαστικά οι γραμμές αυτές αφορούν τον σχηματισμό των μητρώων παραμορφώσεως καμπτικής και διατμητικής συμπεριφοράς, και στην συνέχεια την σύνθεση των επιμέρους αυτών μητρώων σε ένα που θα είναι συμβατό με το μητρώο που υπολογίζεται στις γραμμές 74-79 ώστε να συμπεριληφθεί η καμπτική και η διατμητική ακαμψία του στοιχείου. Τέλος στην γραμμή 93 γίνεται ο μετασχηματισμός από το φυσικό σύστημα που χρησιμοποιήθηκε για τον υπολογισμό των καμπυλοτήτων και των παραμορφώσεων του στοιχείου στο τοπικό καρτεσιανό σύστημα ώστε να υπολογισθεί και στη συνέχεια η ακαμψία του στοιχείου στο σύστημα αυτό. Για τον μετασχηματισμό αυτό χρησιμοποιούνται οι σχέσεις (2.20) και (2.36) για την καμπτική και την διατμητική λειτουργία αντίστοιχα.

Γραμμές 95-98

Σε αυτές τις γραμμές εισάγεται τελικώς η ακαμψία που αντιστοιχεί στο σημείο αριθμητικής ολοκλήρωσης το οποίο αφορά η τρέχουσα επανάληψη(Γραμμή 52). Δημιουργούνται καταρχάς δυο μητρώα μιας διάστασης στα οποία απαριθμούνται οι βαθμοί ελευθερίας της μεμβρανικής λειτουργίας(μητρώο $indexm$) και της καμπτικής-διατμητικής λειτουργίας (μητρώο $indexbs$), ούτως ώστε η ακαμψία που αφορά την εκάστοτε επιμέρους λειτουργία του στοιχείου να προστίθεται στους βαθμούς ελευθερίας τους οποίους επηρεάζει. Έπειτα, εφαρμόζεται τελικώς η αριθμητική ολοκλήρωση εισάγοντας στους επηρεαζόμενους βαθμούς ελευθερίας την ακαμψία που προκύπτει από τον πολλαπλασιασμό μητρώου παραμορφώσεως και μητρώου υλικού με το βάρος του σημείου ολοκλήρωσης Gauss και το εμβαδόν της τριγωνικής επιφάνειας.

Γραμμή 101

Το μητρώο ακαμψίας που υπολογίσθηκε στο τοπικό καρτεσιανό σύστημα του στοιχείου μετατρέπεται με χρήση του μητρώου Transform των γραμμών 15-20 στο καθολικό σύστημα του φορέα, ώστε να συμπεριληφθεί η επίδραση του στους αρμόζοντες καθολικούς βαθμούς ελευθερίας και να μπορέσει εν τέλει να λυθεί το στατικό πρόβλημα πεπερασμένων στοιχείων που αντιμετωπίζουμε.

Η συνάρτηση για τον καθορισμό του τοπικού συστήματος x-y-z του στοιχείου, καθώς και για την μετατροπή των συντεταγμένων του από το καθολικό στο τοπικό του σύστημα, είναι η $dirvecT3(eIncoor)$. Η μόνη παράμετρος που χρειάζεται κατά την κλήση της είναι η $eIncoor$, ένα 6×3 μητρώο που αποτελείται από τις X-Y-Z καθολικές συντεταγμένες του στοιχείου.

Γραμμή 9

Εκκίνηση του 3×3 μητρώου dir , το οποίο προορίζεται για την αποθήκευση κατά στήλη των μοναδιαίων διανυσμάτων που συνθέτουν το τοπικό καρτεσιανό σύστημα x - y - z

Γραμμές 10-11

Δημιουργία του κατά x μοναδιαίου διανύσματος, το οποίο ορίζεται να έχει κατεύθυνση από τον πρώτο προς τον δεύτερο κόμβο του στοιχείου. Το μοναδιαίο μέτρο του διανύσματος επιτυγχάνεται με την διαίρεση των συνιστωσών του με το συνολικό μήκος του διανύσματος που αρχικά προκύπτει, στην Γραμμή 11.

Γραμμές 12-13

Δημιουργία του κατά z μοναδιαίου διανύσματος, κάθετου στο επίπεδο του τριγωνικού στοιχείου πλάκας. Η καθετότητα του διανύσματος αυτού επιτυγχάνεται μέσω ενός εξωτερικού γινομένου μεταξύ δύο οποιονδήποτε διανυσμάτων που να ανήκουν στο επίπεδο του τριγώνου. Και πάλι το μοναδιαίο μέτρο του διανύσματος επιτυγχάνεται με την διαίρεση του με το αρχικώς προκύπτον μέτρο του.

Γραμμή 14

Δημιουργία του κατά y μοναδιαίου διανύσματος, το οποίο προκύπτει πλέον εύκολα ως ένα διάνυσμα κάθετο στο επίπεδο που ορίζουν τα δυο προηγούμενα, ικανοποιώντας όμως και το δεξιόστροφο του τρισσορθογώνιου καρτεσιανού συστήματος που θέλουμε να δημιουργήσουμε. Έτσι, το ζητούμενο διάνυσμα σχηματίζεται από το εξωτερικό γινόμενο των μοναδιαίων διανυσμάτων z και x , με αυτή την φορά, μιας και η αντίστροφη σειρά των διανυσμάτων κατά τον υπολογισμό του εξωτερικού γινομένου θα δώσει ένα αριστερόστροφο τρισσορθογώνιο Καρτεσιανό σύστημα. Στην περίπτωση του κατά y μοναδιαίου διανύσματος δεν υπάρχει λόγος να γίνει διαίρεση με το μήκος του για να σχηματισθεί το μοναδιαίο, αφού το εξωτερικό γινόμενο δυο μοναδιαίων διανυσμάτων δίνει με την σειρά του ένα μοναδιαίο κάθετο σε αυτά διάνυσμα.

Γραμμή 15

Αναστροφή του μητρώου dir έτσι ώστε η κάθε στήλη να περιέχει ένα εκ των μοναδιαίων διανυσμάτων x - y - z των τοπικών Καρτεσιανών αξόνων.

Γραμμές 17-18

Μετατροπή των συντεταγμένων των κόμβων από το X - Y - Z καθολικό σύστημα στο τοπικό σύστημα του τριγώνου που ορίζεται από τα μόλις υπολογισθέντα μοναδιαία διανύσματα του μητρώου dir .

Η συνάρτηση για την δημιουργία των σημείων αριθμητικής ολοκλήρωσης κατά Gauss και των αντίστοιχών τους συντελεστών βάρους είναι η `TriGauss(option)`. Μοναδική παράμετρος εισόδου που απαιτείται είναι η τάξη της αριθμητικής ολοκλήρωσης που επιθυμεί ο χρήστης, προκειμένου να κατασκευασθεί το απαραίτητο πλήθος σημείων. Οι (r,s) ισοπαραμετρικές συντεταγμένες του τριγώνου και τα βάρη τους για κάθε τάξη αριθμητικής ολοκλήρωσης μπορούν να ευρεθούν σε διάφορα βιβλία και επιστημονικές δημοσιεύσεις περί αριθμητικών υπολογιστικών μεθόδων, τα χρησιμοποιούμενα στον κώδικα του Παραρτήματος Α βρέθηκαν στην ιστοσελίδα http://math2.unc.edu/~shaodeng/TEACHING/math5172/Lectures/Lect_15.PDF, σε έγγραφο που συνέθεσε ο κ. Shaozhong Deng.

Οι υπόλοιπες συναρτήσεις που χρησιμοποιούνται είτε άμεσα είτε έμμεσα από την συνάρτηση δημιουργίας του εξακομβικού τριγωνικού στοιχείου απλά παρατίθενται στο

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

τέλος του Παραρτήματος Α και δε επεξηγούνται, όντας προφανείς ως προς την λειτουργία τους και τις εσωτερικές τους διαδικασίες

ΚΕΦΑΛΑΙΟ 3^ο

ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΤΟΠΟΛΟΓΙΑΣ

3.1 Γενικά

Για την κατασκευή ενός έργου υπάρχει πληθώρα επιλογών που να ικανοποιεί τις απαιτήσεις και τις προδιαγραφές που αυτό πρέπει να πληροί. Το πρόβλημα λοιπόν που καλείται να λύσει ο μηχανικός κατά τον σχεδιασμό δεν ολοκληρώνεται με την εύρεση ενός φορέα που υπακούει στον κανονισμό και επιτυγχάνει στον λόγο δημιουργίας του, αλλά παρατείνεται στην επιλογή εκείνη η οποία αποτελεί την καλύτερη από τις λύσεις που μπορεί να καταλήξει. Το πλήθος, η λειτουργικότητα και η αποδοτικότητα των κατασκευαστικών σεναρίων εξαρτώνται άμεσα με την εμπειρία και τις ικανότητες του μηχανικού, οδηγώντας στο προφανές συμπέρασμα πως ένας πιο έμπειρος και με μεγαλύτερο γνωστικό υπόβαθρο μηχανικός θα καταλήξει σε καλύτερη σχεδίαση φορέα από έναν άπειρο όχι και τόσο καλοδιαβασμένο ανταγωνιστή του. Αυτήν την ικανότητα ενός καλού μηχανικού να καταλήγει σε μια βέλτιστη μορφή φορέα από ένα σύνολο πιθανών λύσεων είναι που προσπαθούν να επιτύχουν οι μέθοδοι βελτιστοποίησης.

Οι μέθοδοι βελτιστοποίησης κατά την εφαρμογή τους στοχεύουν στην εύρεση του καλύτερου δυνατού αποτελέσματος από ένα σύνολο δυνατών λύσεων σε ένα πρόβλημα σαφώς ορισμένο. Συνήθως σε κατασκευαστικού είδους τέτοια προβλήματα το σύνολο των λύσεων είναι αρκετά μεγάλο, και δεν είναι δυνατόν να ελεγχθούν όλες οι πιθανές λύσεις ώστε να επιλεγεί τελικώς με απόλυτη σιγουριά η καλύτερη δυνατή. Αντ' αυτού, η κάθε μέθοδος εφαρμόζει κάποιου είδους επεξεργασία στα δεδομένα του προβλήματος προκειμένου να κατευθυνθεί με έναν αυτοδιδαχθή τρόπο γρήγορα στην μορφή της λύσης που ικανοποιεί στο μεγαλύτερο δυνατό βαθμό τα κριτήρια που έχουν τεθεί. Τα κριτήρια αυτά ποικίλουν όχι μόνο ανάλογα με τον αντικειμενικό σκοπό της μεθόδου βελτιστοποίησης, αλλά εξαρτώνται και από τον τρόπο που η μέθοδος αντιλαμβάνεται και διαχειρίζεται το πρόβλημα.

Οι μέθοδοι βελτιστοποίησης στατικών φορέων χωρίζονται σε τρεις βασικές κατηγορίες. Στην βελτιστοποίηση σχήματος, στην βελτιστοποίηση μεγέθους και τέλος στην βελτιστοποίηση της τοπολογίας αυτών. Στο παρών έργο ασχολούμαστε καθαρά με την βελτιστοποίηση της τοπολογίας των φορέων, και για την ακρίβεια με τρεις διαφορετικές μεθόδους εφαρμογής της, παρόλα αυτά γίνεται μια σύντομη αναφορά και στις άλλες δυο μεθόδους για λόγους πληρότητας.

Η βελτιστοποίηση μεγέθους αφορά την εύρεση των διατομών εκείνων για δεδομένη γεωμετρία φορέα που θα ελαχιστοποιήσουν την αντικειμενική συνάρτηση του προβλήματος. Κλασικό παράδειγμα της βελτιστοποίησης μεγέθους ενός φορέα είναι ο καθορισμός του πάχους σε κάθε σημείο ενός κελυφωτού φορέα δεδομένης γεωμετρίας. Υπό μια έννοια αποτελεί την πιο εύκολη στην πρακτική της εφαρμογή μέθοδο, καθώς ένα συνήθες πρόβλημα κατά τον σχεδιασμό φορέων είναι η εύρεση των κατάλληλων διατομών για κάθε θέση, αφού όμως η συνολικότερη γεωμετρία του φορέα έχει επιλεγεί βάσει των λειτουργικών απαιτήσεων της κατασκευής. Από την άλλη, η βελτιστοποίηση σχήματος σκοπό έχει την εύρεση του βέλτιστου σχήματος ενός φορέα, με στόχο και πάλι την ελαχιστοποίηση της αντικειμενικής συνάρτησης. Η διαδικασία της βελτιστοποίησης ξεκινάει με την εισαγωγή(με κάποιο τρόπο) της γεωμετρίας ενός φορέα στον υπολογιστή και των

απαιτήσεων που θα πρέπει να ικανοποιεί, και επαφίεται στον αλγόριθμο η επεξεργασία της γεωμετρίας αυτής έως ώτου να καταλήξει στην καλύτερη δυνατή μορφή για την ικανοποίηση των απαιτήσεων που του δόθηκαν. Τέλος, η βελτιστοποίηση της τοπολογίας ενός φορέα αντιμετωπίζει το πρόβλημα κατά το οποίο της διατίθεται ένα πλέγμα πεπερασμένων στοιχείων, οι συνοριακές του συνθήκες και ο περιορισμός υλικού που μπορεί να χρησιμοποιήσει, και εναπόκειται σε αυτήν ο καθορισμός των περιοχών υλικού και κενού εντός του πλέγματος για την δημιουργία της μορφής εκείνης που θα ικανοποιεί με τον καλύτερο τρόπο την αντικειμενική συνάρτηση. Όπως βλέπουμε η βελτιστοποίηση σχήματος έχει αρκετά κοινά με την βελτιστοποίηση τοπολογίας ενός φορέα. Η βασική τους όμως διαφορά είναι πως για την μέθοδο βελτιστοποίησης σχήματος το ίδιο το πλέγμα πεπερασμένων αποτελεί παράμετρο για την μέθοδο, καθώς η μέθοδος μεταβάλλει συνεχώς τα όρια του φορέα χωρίς να περιορίζεται από κάποιο χώρο κίνησης της. Από την άλλη, η βελτιστοποίηση τοπολογίας δέχεται ως δεδομένο το πλέγμα πεπερασμένων στοιχείων και προσπαθεί να κάνει την καλύτερη επιλογή στοιχείων που θα χρησιμοποιήσει προκειμένου να επιτύχει την βελτιστοποίηση του φορέα, χωρίς όμως να μπορεί να επέμβει επί της μορφής του πλέγματος.

Σκοπός λοιπόν βελτιστοποίησης της τοπολογίας ενός φορέα είναι η επιλογή των στοιχείων εκείνων από το πλέγμα του τα οποία θα συνθέσουν την τελική γεωμετρία της κατασκευής, όπου ως επιλογή νοείται είτε η πλήρης επιλογή με την έννοια της ύπαρξης υλικού ή κενού είτε η μερική επιλογή με την έννοια της ύπαρξης πυκνότητας μεγαλύτερης από την ελάχιστη επιτρεπόμενη. Παρακάτω ο τρόπος επιλογής ανάλογα με την μέθοδο θα γίνει ευκολότερα κατανοητός κατά την παρουσίαση των τριών μεθόδων ξεχωριστά, όπως κατανοητός θα γίνει και ο τρόπος εφαρμογής τους σε μη δομημένα πλέγματα πεπερασμένων στοιχείων, που αποτελεί άλλωστε και το αντικείμενο της παρούσας διπλωματικής.

Κατά τον ορισμό των μεθόδων βελτιστοποίησης έγινε αναφορά στον σκοπό που έχει η κάθε μια εξ αυτών, ο οποίος είναι η επίτευξη της καλύτερης δυνατής τιμής για την αντικειμενική συνάρτηση. Η έννοια της αντικειμενικής συνάρτησης, την οποία και συναντάμε σε όλες τις μεθόδους βελτιστοποίησης, αποτελεί στην ουσία το φυσικό μέγεθος που είναι το πλέον σημαντικό για τον φορέα. Αν και δεν υπάρχει κάποιος γενικός κανόνας, συνήθως ως αντικειμενική συνάρτηση επιλέγεται η συνολική ελαστική παραμόρφωση του φορέα, που εξαρτάται από την μεταβλητή κατανομής του υλικού στα στοιχεία του πλέγματος. Σε πολλές περιπτώσεις όμως εάν κάποιος κάνει μια πιο ενδελεχή έρευνα των μεθόδων βελτιστοποίησης, κι όχι απαραίτητα μόνον της τοπολογίας, θα συναντήσει πολλές εφαρμογές κατά τις οποίες ως αντικειμενική συνάρτηση δεν επιλέγεται η συνολική ελαστική παραμόρφωση του φορέα αλλά φυσικά μεγέθη όπως μια χαρακτηριστική μετακίνηση σε κάποιο βαθμό ελευθερίας του φορέα ή ο περιορισμός της μέγιστης αναπτυσσόμενης τάσης στα στοιχεία. Για την ακρίβεια, ως αντικειμενική συνάρτηση μπορεί να επιλεγεί οποιοδήποτε φυσικό μέγεθος που εμφανίζεται κατά την ανάλυση ενός φορέα, εφόσον βεβαίως μπορεί να διατυπωθεί μαθηματικώς και να συνδεθεί με τις υπόλοιπες παραμέτρους που εμφανίζονται κατά την επίλυση του προβλήματος πεπερασμένων στοιχείων. Ο λόγος που συνήθως επιλέγεται η συνολική ελαστική παραμόρφωση του φορέα έναντι των άλλων χαρακτηριστικών του μπορούν να αποτελέσουν κριτήριο βέλτιστης μορφής είναι πως η ελαστική παραμόρφωση δίνει μια καλύτερη συνολική εικόνα για την κατασκευή έναντι για παράδειγμα του περιορισμού της μετακινήσεως ενός μονάχα βαθμού ελευθερίας. Παρόλα αυτά έχει υπάρξει έρευνα για την υλοποίηση βελτιστοποίησης ως προς την μέγιστη εμφανιζόμενη τάση ή κάποια χαρακτηριστική μετακίνηση κυρίως επειδή η βελτιστοποίηση τέτοιων μεγεθών έχει μεγάλη σημασία σε πρακτικά κατασκευαστικά προβλήματα, όπου για παράδειγμα ενδιαφέρει η ένα σημείο μιας κατασκευής παρά την όλη του φόρτιση να μην εμφανίζει μεγάλες μετακινήσεις, ή σε έναν

φορέα να θέλουμε να περιορίσουμε τις αναπτυσσόμενες τάσεις προκειμένου να είναι εφικτή η χρήση οικονομικότερου υλικού χαμηλότερης ποιότητας.

Επιπλέον των παραπάνω, οι βελτιστοποιήσεις μιας κατασκευής μπορούν να εφαρμοστούν για περισσότερες από μια φορτίσεις, έχοντας βεβαίως όμως πρώτα ορίσει την σημασία της κάθε φόρτισης για τον προκύπτων βέλτιστο φορέα. Η σημασία της κάθε φορτίσεως καθορίζεται μέσω ενός συντελεστή βαρύτητας για την κάθε φόρτιση. Έτσι η μέθοδος της βελτιστοποίησης εφαρμόζεται ξεχωριστά για κάθε φόρτιση και τα αποτελέσματα που προκύπτουν επαλληλίζονται πολλαπλασιασμένα με τον συντελεστή βαρύτητας της φορτίσεως. Στα πλαίσια της παρούσας εργασίας δεν γίνεται κάποια εφαρμογή με πολλαπλά σενάρια φορτίσεως, αλλά η επεξεργασία του κώδικα για την υλοποίηση τέτοιων εφαρμογών είναι αρκετά απλή δεδομένου του πως είναι στημένος ο αλγόριθμος. Το σημείο στο οποίο όμως πρέπει να δοθεί ιδιαίτερη προσοχή είναι η θέση στον κώδικα που πρέπει να γίνει η επαλληλία με τους συντελεστές βαρύτητας, κι αυτό γιατί ενώ είναι αρκετά απλό στην μέθοδο SIMP όπου υπάρχει η έννοια της πυκνότητας των στοιχείων, γίνεται λιγάκι πιο σύνθετο στις μεθόδους BESO και Level Set, όπου και υπάρχουν δύο καταστάσεις για τα στοιχεία, αυτή της ύπαρξης υλικού ή αυτή της ύπαρξης κενού. Σε αυτές τις μεθόδους λοιπόν η επαλληλία πρέπει να γίνει σε όρους ευαισθησιών, ώστε να γίνει στην συνέχεια η επιλογή των στοιχείων που θα παραμείνουν ή όχι στην κατασκευή. Παραδείγματα με πολλαπλά σενάρια φορτίσεως κατά την βελτιστοποίηση τοπολογίας μπορεί κανείς να βρει στις περισσότερες δημοσιεύσεις επί της μεθόδου, με ένα αρκετά χαρακτηριστικό αυτό που παρουσιάζεται την δημοσίευση "A 99 line topology optimization code written in Matlab" του O. Sigmund.

Στην συνέχεια του παρόντος κεφαλαίου θα δούμε ξεχωριστά τις τρεις μεθόδους βελτιστοποίησης τοπολογίας επί των οποίων γίνεται το παρών έργο, τόσο ως προς την θεωρητική τους πλευρά όσο και ως προς την πρακτική τους εφαρμογή σε πλέγματα πεπερασμένων στοιχείων. Συγχρόνως, παρουσιάζονται τα σημεία στα οποία ο κώδικας που αναπτύχθηκε αναγκαστικά διαχειρίστηκε διαφορά το πρόβλημα από τον τρόπο που το διαχειρίζονται οι συγγραφείς χαρακτηριστικών δημοσιεύσεων επί της κάθε μεθόδου, κι αυτό λόγω του ότι ο αλγόριθμος έπρεπε να αντιμετωπίσει τα μη δομημένα πλέγματα πεπερασμένων στοιχείων που εξάγονται από το GMSH, πλέγματα τα οποία σπανίως συναντά κανείς ψάχνοντας την βιβλιογραφία των μεθόδων βελτιστοποίησης τοπολογίας. Υπάρχουν όμως περιπτώσεις φορέων όπου λόγω της μορφής τους η διακριτοποίηση που θα προκύψει είναι αναπόφευκτα μη δομημένη, συνεπώς είναι αναγκαία η ανάπτυξη μεθοδολογίας για την αντιμετώπιση του προβλήματος της βελτιστοποίησης και σε μη δομημένα πλέγματα. Έτσι, η μορφή της διακριτοποίησης δεν θα αποτελεί πρόβλημα στην εφαρμογή των μεθόδων και βέλτιστες τοπολογίες φορέων θα μπορούν να βρεθούν για κάθε είδους γεωμετρίας και συνθήκες φορτίσεως.

3.2 Βελτιστοποίηση τοπολογίας με χρήση της μεθόδου SIMP

3.2.1 Μαθηματική Διατύπωση της SIMP

Η μέθοδος βελτιστοποίησης φορέων Solid Isotropic Material with Penalization, ή σε συντομογραφία SIMP, είναι μια από τις πρώτες μεθόδους βελτιστοποίησης που αναπτύχθηκαν σε ικανοποιητικό επίπεδο και συνέβαλε σημαντικά στην περαιτέρω διάδοση τους και συγκέντρωση πολλών ερευνών επί του θέματος. Η διάδοση αυτή οφείλεται και σε μεγάλο βαθμό στη δημοσίευση "A 99 line topology optimization code written in Matlab" του O. Sigmund, πάνω στην οποία έχει βασισθεί και ο κώδικας βελτιστοποίησης με την μέθοδο SIMP του παρόντος έργου. Στην αρχή αυτού του κεφαλαίου παρουσιάζονται οι μαθηματικές σχέσεις και η λογική που ακολουθείται για την εύρεση του βέλτιστου φορέα σύμφωνα με την μέθοδο SIMP, ή ακριβέστερα σύμφωνα με την υλοποίηση της στην δημοσίευση του O. Sigmund, ενώ στην συνέχεια παρουσιάζεται βήμα βήμα ο κώδικας σε matlab του Παραρτήματος Β, αντίστοιχος αυτού που εισήχθη στην δομή MSolve και χρησιμοποιήθηκε για τις αναλύσεις.

Κατά την μέθοδο SIMP η ακαμψία ενός στοιχείου εξαρτάται από την πυκνότητα του σε υλικό. Οι ιδιότητες του υλικού θεωρούνται σταθερές στα πλαίσια κάθε στοιχείου και πολλαπλασιάζονται με την πυκνότητα του στοιχείου, υψωμένη σε κάποιο βαθμό, ώστε να αγνοηθεί η επιρροή των στοιχείων με πολύ μικρή πυκνότητα στην συνολική δυσκαμψία του φορέα. Στο παρών έργο, ο συγκεκριμένος τρόπος αντιμετώπισης για την αφαίρεση της δυσκαμψίας των κενών στοιχείων αλλοιώνεται μερικώς λόγω κάποιων αριθμητικών προβλημάτων που προκύπτουν κατά την χρήση του MSolve, και τα οποία θα εξηγηθούν στην πορεία, όπως άλλωστε και ο τρόπος αντιμετώπισης τους.

Η βασική λογική της μεθόδου έγκειται στην ελαχιστοποίηση της αντικειμενικής συναρτήσεως, η οποία στην δική μας περίπτωση είναι η ελαστική παραμόρφωση της κατασκευής. Υπάρχουν κι άλλες προσεγγίσεις της μεθόδου με διαφορετικές αντικειμενικές συναρτήσεις, μια αρκετά συχνή παραλλαγή των οποίων είναι ο περιορισμός της μετακινήσεως σε έναν συγκεκριμένο βαθμό ελευθερίας.

Οι βασικές σχέσεις για την διατύπωση ενός προβλήματος βελτιστοποίησης με την μέθοδο SIMP είναι

$$\begin{aligned} \min_x: c(x) &= U^T K U = \sum_{e=1}^N (x_e)^p u_e^T k_0 u_e \\ \text{υποκείμενο σε: } & \frac{V(x)}{V_0} = f \\ & : K U = F \\ & : 0 < x_{min} \leq x \leq 1 \end{aligned} \tag{3.1}$$

όπου με U και F συμβολίζουμε τα διανύσματα καθολικών επικομβίων μετατοπίσεων και δράσεων αντίστοιχα, με K το μητρώο ακαμψίας του φορέα στο καθολικό σύστημα, u_e και k_e είναι το διάνυσμα επικομβίων μετατοπίσεων και το μητρώο ακαμψίας για τα διάφορα τριγωνικά στοιχεία, x είναι το διάνυσμα μεταβλητών σχεδιασμού που στην

προκειμένη περίπτωση περιλαμβάνει τις πυκνότητες των στοιχείων, x_{min} είναι το διάλυμα ελάχιστων πυκνοτήτων για κάθε στοιχείο έτσι ώστε να μην τίθεται η τιμή μηδέν ως πυκνότητα και να αποφεύγεται ο σχηματισμός ιδιόμορφων μητρώων καθολικής δυσκαμψίας, N είναι ο αριθμός των πεπερασμένων στοιχείων που χρησιμοποιήθηκαν για την διακριτοποίηση του φορέα, p (penalization power) είναι η δύναμη στην οποία υψώνεται η πυκνότητα, $V(x)$ και V_0 είναι ο όγκος του υλικού και ο συνολικός όγκος του διατιθέμενου χώρου για τον φορέα αντίστοιχα, και τέλος f είναι το ποσοστό του διατιθέμενου όγκου που θέλουμε να καταλαμβάνει ο βέλτιστος φορέας που θα δημιουργηθεί.

Το πρόβλημα βελτιστοποίησης (3.1) μπορεί να λυθεί με διάφορες προσεγγίσεις όπως η μέθοδος των Optimality Criteria(OC), η μέθοδος Sequential Linear Programming(SLP), και η Method of Moving Asymptotes(MMA). Για απλότητα, στη δημοσίευση του O. Sigmund χρησιμοποιείται η μέθοδος των Optimality Criteria, συνεπώς και στο παρόν, που αποτελεί εφαρμογή σε μη δομημένα πλέγματα με χρήση τριγωνικών στοιχείων της παραπάνω δημοσίευσης όπως προαναφέρθηκε, χρησιμοποιείται η ίδια προσέγγιση.

Κατά Bendsøe(1995)[8], μια ευρετική στρατηγική ενημέρωσης των μεταβλητών σχεδιασμού μπορεί να διαμορφωθεί ως

$$x_e^{new} = \begin{cases} \max(x_{min}, x_e - m) & \text{if } x_e B_e^n \leq \max(x_{min}, x_e - m) \\ x_e B_e^n & \text{if } \max(x_{min}, x_e - m) < x_e B_e^n < \min(1, x_e + m) \\ \min(1, x_e + m) & \text{if } \min(1, x_e + m) \leq x_e B_e^n \end{cases} \quad (3.2)$$

όπου m (move) είναι το θετικό όριο βήματος κατά την ενημέρωση των μεταβλητών σχεδιασμού, $n(=1/2)$ είναι ένας συντελεστής αριθμητικής απόσβεσης και το B_e μπορεί να βρεθεί από την συνθήκη βελτιστοποίησης ως

$$B_e = \frac{-\frac{\partial c}{\partial x_e}}{\lambda \frac{\partial V}{\partial x_e}} \quad (3.3)$$

όπου το λ στην παραπάνω είναι ένας πολλαπλασιαστής Lagrange που μπορεί να υπολογιστεί αλγοριθμικά μέσω επαναλήψεων.

Η ευαισθησία της αντικειμενικής συνάρτησης μπορεί να βρεθεί ως

$$\frac{\partial c}{\partial x_e} = -p(x_e)^{p-1} u_e^T k_0 u_e \quad (3.4)$$

Έχει παρατηρηθεί πως μπορεί να επιτευχθεί ελαχιστοποίηση της ελαστικής παραμόρφωσης μιας κατασκευής δημιουργώντας εναλλάξ τμήματα υλικού και κενών σε όλο τον διατιθέμενο χώρο για την υλοποίηση του φορέα. Τα μοτίβα αυτά ονομάζονται checkerboard patterns, ακριβώς λόγω της μορφής τους που είναι παραπλήσια ενός σκακιστικού ταμπλό με τα εναλλάξ ασπρόμαυρα τετράγωνα. Μια τέτοια κατασκευή όμως παρουσιάζει πρακτικές δυσκολίες κατά την υλοποίηση της στην πραγματικότητα, αφού δεν

πρέπει να ξεχνάμε τον λόγο που εξυπηρετούν οι μέθοδοι βελτιστοποίησης. Προκειμένου να αποφευχθεί η σύγκλιση του αλγορίθμου βελτιστοποίησης σε έναν τέτοιας μορφής φορέα εφαρμόζεται φίλτρο ανεξαρτητοποίησης από το πλέγμα, με το οποίο η ευαισθησία της αντικειμενικής συνάρτησης για ο κάθε στοιχείο τροποποιείται ως

$$\widehat{\frac{\partial c}{\partial x_e}} = \frac{1}{x_e \sum_{f=1}^N \widehat{H}_f} \sum_{f=1}^N \widehat{H}_f x_f \frac{\partial c}{\partial x_f} \quad (3.5)$$

Ο συντελεστής \widehat{H}_f , ο οποίος στην ουσία αποτελεί έναν συντελεστή βάρους, μπορεί να γραφεί ως

$$\widehat{H}_f = r_{min} - dist(e, f), \quad (3.6)$$

$$\{f \in N \mid dist(e, f) \leq r_{min}\}, \quad e = 1, \dots, N,$$

όπου ο όρος $dist(e, f)$ ορίζεται ως η απόσταση από το κέντρο του τριγωνικού στοιχείου e έως το κέντρο του στοιχείου f . Στην περίπτωση που η απόσταση αυτή είναι μεγαλύτερη της ακτίνας επιρροής του φίλτρου r_{min} ο συντελεστής \widehat{H}_f τίθεται ίσος με μηδέν. Έτσι, αντί των ευαισθησιών που διατυπώθηκαν στην σχέση (3.5) για τις αντικειμενικές συναρτήσεις των στοιχείων χρησιμοποιούνται αυτές της (3.6), ώστε να αποφευχθούν τα μη υλοποιήσιμα μοτίβα μορφής σκακιστικού ταμπλό, κατά την εφαρμογή των Optimality Criteria.

Το φίλτρο όμως αυτό δημιουργεί και ένα πρόβλημα. Επειδή οι ευαισθησίες των στοιχείων επηρεάζονται από αυτές των γύρω στοιχείων, ή τουλάχιστον όσων βρίσκονται εντός της ακτίνας επιρροής του φίλτρου, στοιχεία τα οποία δεν θα έπρεπε να συμμετέχουν στην κατασκευή όντας όχι και τόσο χρήσιμα σε σχέση με κάποια άλλα αποκτούν πυκνότητα. Η παρασιτική αυτή επιβίωση ορισμένων από τα πεπερασμένα του πλέγματος οφείλεται στην επαφή τους με σημαντικά για την βέλτιστη μορφή που αναζητείται στοιχεία. Οι ευαισθησίες τους μεγαλώνουν λόγω της μικρής τους απόστασης από τα σημαντικά αυτά στοιχεία, ενώ μπορεί αρχικώς η καθαρά δικά τους ευαισθησία χωρίς την εφαρμογή του φίλτρου να ήταν αμελητέα. Αποτέλεσμα αυτού είναι μέρος του διατιθέμενου υλικού να προσδίδεται σε αυτά εις βάρος κάποιων άλλων, αφού το συνολικό ποσοστό υλικού που μπορεί να χρησιμοποιηθεί είναι δεδομένο. Σε δυσμενείς περιπτώσεις μπορεί η ευαισθησία τους να ξεπεράσει αυτήν που έχουν κάποια στοιχεία σε ένα άλλο τμήμα του φορέα και να 'κλέψουν' την δική τους πυκνότητα, αποδυναμώνοντας μέλη του φορέα και οδηγώντας σε μεγαλύτερη συνολική ελαστική παραμόρφωση από αυτήν που θα μπορούσε να επιτευχθεί. Το πρόβλημα αυτό γίνεται ακόμα πιο σοβαρό κατά την εφαρμογή της μεθόδου BESO, όπως θα εξηγηθεί παρακάτω. Η παρασιτική αυτή επιβίωση στοιχείων παρατηρείται κυρίως σε οξείες γωνίες μεταξύ ήδη σχηματισμένων μελών του φορέα. Για να επιλυθεί έστω ως ένα βαθμό αυτό το πρόβλημα ακολουθείται μια στρατηγική μείωσης της ακτίνας επιρροής του φίλτρου όσο πλησιάζουμε προς την λύση. Η μείωση αυτή όμως δε πρέπει να είναι απότομη, αφού αν μειωθεί πολύ η ακτίνα επιρροής θα δημιουργηθεί αστάθεια στον αλγόριθμο, συνέπεια της ξαφνικής απώλειας της επαφής μεταξύ πολλών στοιχείων κατά τον υπολογισμό των ευαισθησιών. Επιλέγουμε λοιπόν να συνδέσουμε την ακτίνα επιρροής με την μεταβλητή *change* που δηλώνει την μέγιστη μεταβολή από την μια επανάληψη του αλγορίθμου βελτιστοποίησης στην επόμενη, μέσω της παρακάτω σχέσης

$$r_{min} = \frac{r_{min,0}}{1.5 - 5change} \quad (3.7)$$

η λογική της οποίας είναι πως το $r_{min,0}$ που εισάγεται κατά την έναρξη του κώδικα είναι περίπου ίσο με 1.25 επί του μεγέθους των στοιχείων (με το μέγεθος των στοιχείων να είναι ίσο με τον μέσο όρο των πλευρών των τριγώνων) θα ξεκινήσει να μειώνεται σταδιακά από την στιγμή που το change θα πέσει κάτω από την τιμή 0.1. Προφανώς και οι παραπάνω τιμές (π.χ. το 0.1 κάτω από το οποίο αρχίζει η μεταβολή της ακτίνας επιρροής) είναι ενδεικτικές και θα μπορούσαν να έχουν επιλεγθεί διαφορετικές ανάλογα με το εκάστοτε πρόβλημα.

3.2.2 Εφαρμογή της μεθόδου SIMP σε μη δομημένα πλέγματα

Προκειμένου να μπορεί να εφαρμοστεί η μέθοδος βελτιστοποίησης τοπολογίας SIMP στα μη δομημένα πλέγματα πεπερασμένων στοιχείων που εξάγονται από την διακριτοποίηση του GMSH, πρέπει να γίνουν ορισμένες αλλαγές στον τρόπο που η μέθοδος αντιλαμβάνεται το πλέγμα.

Το βασικότερο πρόβλημα που δημιουργείται από την εφαρμογή σε ένα μη δομημένο πλέγμα είναι αυτό της δυσκολίας ορισμού για το εκάστοτε στοιχείο αυτών που βρίσκονται στην γύρω του περιοχή, κάτι που χρειάζεται κατά το φιλτράρισμα των ευαισθησιών. Στον κώδικα της δημοσίευσης παρατηρεί κανείς πως η ακτίνα εφαρμογής του φίλτρου εκφράζεται σε αριθμό στοιχείων και όχι σε κάποια μονάδα μήκους, όπως θα περίμενε κανείς αντικρίζοντας τον όρο ακτίνα. Αυτό βολεύει στην περίπτωση ενός δομημένου πλέγματος καθώς και μόνο από τον αύξοντα αριθμό του στοιχείου μπορεί γρήγορα να υπολογιστεί η θέση του μέσα στο πλέγμα, δηλαδή σε ποια σειρά και στήλη βρίσκεται. Έτσι μπορούν με προσαύξηση του αύξοντα αριθμού του πεπερασμένου κατά την ακτίνα επιρροής προς κάθε διάσταση να προσδιοριστούν εύκολα τα περιμετρικά στοιχεία που θα επηρεάσουν την ευαισθησία. Στην περίπτωση όμως ενός μη δομημένου πλέγματος ο αύξων αριθμός καμία πληροφορία δεν μπορεί να μας δώσει για την θέση του στοιχείου μέσα στο πλέγμα.

Στην περίπτωση λοιπόν ενός μη δομημένου πλέγματος πρέπει να ευρεθεί μια μεθοδολογία με την οποία θα είναι εφικτός ο προσδιορισμός των στοιχείων εκείνων που επηρεάζουν το υπό εξέταση πεπερασμένο. Η ακτίνα πλέον δεν έχει νόημα να εκφραστεί ως αριθμό στοιχείων μιας και όπως είπαμε δε μπορούμε να εξάγουμε κανένα συμπέρασμα από την σύγκριση του αύξοντα αριθμού των στοιχείων, αλλά ορίζεται όπως θα περίμενε κανείς σε όρους μονάδας μήκους. Συγχρόνως, δημιουργούμε ένα νέο μητρώο στο οποίο αποθηκεύεται για κάθε στοιχείο το σημείο κέντρου βάρους, το οποίο υπολογίζεται ως ο μέσος όρος των συντεταγμένων των κορυφών του. Έχοντας πλέον τα κεντροβαρικά σημεία των τριγώνων εφαρμόζουμε μια επαναληπτική διαδικασία μέσω της οποίας για κάθε i στοιχείο ελέγχεται αν η απόσταση του κέντρου βάρους του από το κέντρο βάρους ενός j στοιχείου είναι μικρότερη της ακτίνας r_{min} , οπότε και ανήκει στην σφαίρα επιρροής της. Οι αύξοντες αριθμοί των στοιχείων j που επηρεάζουν την ευαισθησία του στοιχείου i αποθηκεύονται σε ένα μητρώο 'συνδέσεων', στην γραμμή ή την στήλη που αφορά το στοιχείο i . Από προγραμματιστικής σκοπιάς, το σημείο αυτό είναι από τα λιγα εκείνα σημεία που ευκολότερα υλοποιούνται στην C# παρά στην Matlab. Κι αυτό γιατί η C# δίνει την δυνατότητα χρήσης δυναμικών πεδίων αποθήκευσης όπως οι λίστες, και συγχρόνως επιτρέπει την δημιουργία μητρώων που να περιέχουν τέτοιες λίστες. Με τις λίστες γίνεται αρκετά πιο ελαφρύς και γρήγορος ο κώδικας μιας και δε χρειάζεται να δημιουργηθεί στήλη με όλα τα στοιχεία και να αποθηκευτεί μια Boolean τιμή ύπαρξης ή όχι στη σφαίρα επιρροής, αλλά αρκεί η αποθήκευση της ταυτότητας των στοιχείων που ανήκουν στην εν λόγω σφαίρα σε μια λίστα που αντιστοιχεί στο εκάστοτε στοιχείο.

Το επόμενο θέμα που τίθεται είναι σε ποια σημεία του αλγορίθμου πρέπει να υπολογίζονται οι συνδέσεις μεταξύ των στοιχείων. Στην περίπτωση που η ακτίνα επιρροής παρέμενε σταθερή καθ' όλη την διάρκεια των επαναλήψεων θα μπορούσαμε να υπολογίζουμε μονάχα μια φορά τις συνδέσεις και να ανατρέχουμε στον πίνακα που τις αποθηκεύτηκαν κάθε φορά που καλείται η συνάρτηση για το φίλτρο. Επειδή όμως η ακτίνα επιρροής του φίλτρου δεν μένει η ίδια σε όλη την διάρκεια εκτέλεσης της μεθόδου, στην προσπάθεια αντιμετώπισης του φαινομένου της παρασιτικής επιβίωσης πυκνότητας που εξηγείται παραπάνω, υπάρχει ανάγκη δημιουργίας εκ νέου του πεδίου που αποθηκεύονται οι συνδέσεις κάθε φορά που η ακτίνα επιρροής αλλάζει. Ο τρόπος και η λογική με την οποία αλλάζει η ακτίνα παρουσιάζεται στο κεφάλαιο 3.2.1, ενώ για την εκ νέου δημιουργία του μητρώου των συνδέσεων απαιτείται απλώς η σύγκριση με την ακτίνα που υπήρχε στην προηγούμενη επανάληψη η οποία και θα πρέπει να έχει αποθηκευτεί σε μια μεταβλητή για να διατηρηθεί.

Στον κώδικα της δημοσίευσης ο υπολογισμός των σχέσεων μεταξύ των στοιχείων γίνεται σε κάθε επανάληψη της μεθόδου, χωρίς μάλιστα την ανάγκη αποθήκευσης σε κάποιο μητρώο των σχέσεων των στοιχείων, οπότε ο αλγόριθμος δεν επηρεάζεται από πλευράς ταχύτητας με την αλλαγή της ακτίνας επιρροής του φίλτρου.

3.2.3 Κώδικας matlab για βελτιστοποίηση με την μέθοδο SIMP

Όπως και στην περίπτωση του τριγωνικού εξακομβικού στοιχείου, γράφτηκε κώδικας σε matlab για την μέθοδο SIMP βελτιστοποίησης τοπολογίας προκειμένου να παρουσιαστεί και η αλγοριθμική της προσέγγιση. Ο κώδικας αυτός παρατίθεται συνολικώς, μαζί με τις διάφορες συναρτήσεις που καλεί κατά την εκτέλεση του, στο Παράρτημα Β. Σε αυτό το κεφάλαιο θα παρουσιαστεί βήμα προς βήμα ο κώδικας του Παραρτήματος Β και θα εξηγηθούν τα σημεία στα οποία διαφοροποιείται από αυτόν της δημοσίευσης του O. Sigmund.

Η συνάρτηση που καλείται για την εκτέλεση της βελτιστοποίησης με την μέθοδο SIMP είναι η `TopologySIMP(nodes,elemConn,volfrac,rmin,penal,E,v,thickness, fixedDOFs,F)`. Οι παράμετροι που απαιτούνται κατά την κλήση της είναι

`nodes`: οι κόμβοι στο καθολικό σύστημα

`elemConn`: οι συνδέσεις των κόμβων για το κάθε στοιχείο

`volfrac`: ο λόγος του όγκου που θα πρέπει να έχει ο βέλτιστος φορέας ως προς τον συνολικό διατιθέμενο όγκο

`rmin`: η ακτίνα επιρροής του φίλτρου

`penal`: η δύναμη στην οποία υψώνεται η πυκνότητα των στοιχείων

`E`: το μέτρο ελαστικότητας κατά Young του υλικού

`v`: ο λόγος Poisson του υλικού

`thickness`: το πάχος των τριγωνικών στοιχείων

`fixedDOFs`: το διάνυσμα των δεσμευμένων βαθμών ελευθερίας του φορέα

`F`: το διάνυσμα των επικομβίων δράσεων του φορέα

Γραμμές 3-9

Εκκίνηση διάφορων μεταβλητών που θα χρειαστούν στην συνέχεια του αλγορίθμου, όπως ο συνολικός αριθμός τριγωνικών πεπερασμένων `nel`, το διάνυσμα των πυκνοτήτων των στοιχείων που ξεκινάει έχοντας την τιμή `volfrac` για

το κάθε ένα στοιχείο, ο αριθμός της τρέχουσας επανάληψης loop, η αποθήκευση του rmin που δόθηκε από τον χρήστη στην μεταβλητή rmin0 ώστε να μην χαθεί μεταβάλλοντας το rmin στην συνέχεια και τέλος το μητρώο centerPoints για την αποθήκευση των κεντροβαρικών σημείων των τριγώνων.

Γραμμές 11-14

Υπολογισμός των μητρώων ακαμψίας των τριγωνικών στοιχείων με κλήση της συνάρτησης ElementMohrTri6 του παραρτήματος Α, η οποία παρουσιάστηκε στο κεφάλαιο 2. Επίσης υπολογίζονται τα κεντροβαρικά σημεία των τριγώνων ως μέσος όρος των συντεταγμένων των τριών κορυφών τους.

Γραμμή 16

Εκκίνηση των επαναλήψεων για την εκτέλεση της βελτιστοποίησης στον φορέα. Το όριο το οποίο θέλουμε να ικανοποιείται ώστε να σταματήσουν είναι η μέγιστη διαφορά μεταξύ δυο συνεχόμενων επαναλήψεων να είναι μικρότερη από 0.01(μεταβλητή change), αν και θα μπορούσαμε να έχουμε επιλέξει ένα διαφορετικό όριο ανάλογα με την ακρίβεια που επιζητούμε από την λύση.

Γραμμές 17-20

Προσαύξηση του αριθμού επαναλήψεων κατά ένα αφού ξεκινάει μια καινούργια επανάληψη, αποθήκευση στο διάνυσμα cold των πυκνοτήτων που προέκυψαν από την προηγούμενη και εκκίνηση της μεταβλητής c στο μηδέν για την αποθήκευση της ελαστικής παραμόρφωσης του φορέα.

Γραμμή 21

Κλήση της συναρτήσεως FEA για την εκτέλεση της ανάλυσης πεπερασμένων στοιχείων και τον υπολογισμό των επικόμβιων μετατοπίσεων του φορέα. Η συνάρτηση FEA παρουσιάζεται ξεχωριστά παρακάτω.

Γραμμές 22-24

Εφόσον έχει γίνει κάποια αλλαγή στην ακτίνα επιρροής του φίλτρου, πρέπει να ξαναδημιουργηθεί το μητρώο των συνδέσεων μεταξύ των στοιχείων που αλληλοεπηρεάζονται. Στην περίπτωση του κλασσικού κώδικα 99 line topology optimization, επειδή έχουμε ένα δομημένο πλέγμα τετραγωνικών στοιχείων δεν υπάρχει λόγος για τον σχηματισμό ενός τέτοιου πίνακα, μιας και είναι γνωστές οι σχετικές θέσεις των σημείων. Γι αυτόν τον λόγο άλλωστε η ακτίνα ορίζεται σε αριθμό στοιχείων και δεν γίνεται σύγκριση με την απόσταση που έχουν τα μεταξύ τους κέντρα βάρους. Ο υπολογισμός αποστάσεων μεταξύ των στοιχείων και η δημιουργία μητρώου συνδέσεων γίνονται σε βάρος της ταχύτητας του κώδικα, αλλά από την άλλη είναι αναγκαίες ενέργειες για την εφαρμογή της βελτιστοποίησης σε μη δομημένο πλέγμα, όπως αυτό που προκύπτει από την διακριτοποίηση με τριγωνικά στοιχεία. Η προσπάθεια για σύγκλιση σε καλύτερη λύση χωρίς παρασιτικά στοιχεία επιβαρύνει περαιτέρω τον κώδικα με την επανάληψη του υπολογισμού του μητρώου συνδέσεων, υπολογισμός που γίνεται με κλήση της συναρτήσεως Connections.

Γραμμές 25-30

Στις γραμμές αυτές επανάληψη τρέχει όλα τα στοιχεία και υπολογίζει την ευαισθησία τους μέσω της σχέσεως (3.4), όπως επίσης και την συνεισφορά τους στην ελαστική παραμόρφωση του φορέα c. Προκειμένου να γίνουν όλα αυτά πρέπει να βρεθούν οι επικόμβιες μετατοπίσεις του κάθε τριγωνικού στοιχείου. Αρχικά λοιπόν αποθηκεύονται σε μητρώο στήλη οι βαθμοί ελευθερίας του κάθε στοιχείου μέσω της συνάρτησης ElementDOFs και εξάγονται από το διάνυσμα των επικόμβιων μετατοπίσεων ολόκληρου του φορέα αυτές που αντιστοιχούν στο τρέχων στοιχείο. Οι επικόμβιες αυτές μετατοπίσεις του στοιχείου αποθηκεύονται στην μεταβλητή Ue, η οποία χρησιμοποιείται στους υπολογισμούς για την

προσαύξηση της ελαστικής παραμόρφωσης του φορέα στην γραμμή 28, όπως επίσης και στον υπολογισμό της ευαισθησίας του στοιχείου στην γραμμή 29.

Γραμμή 31

Εφαρμογή του φίλτρου της σχέσεως (3.6) με κλήση της συναρτήσεως checkSIMP και απευθείας ενημέρωση των ευαισθησιών των στοιχείων.

Γραμμή 32

Υπολογισμός των πυκνοτήτων των στοιχείων στο διάνυσμα x με εφαρμογή των Optimality Criteria μέσω της συνάρτησης OC.

Γραμμή 33

Υπολογισμός του change ως η μέγιστη διαφορά μεταξύ των πυκνοτήτων της προηγούμενης και της τρέχουσας επανάληψης.

Γραμμές 34-36

Στην περίπτωση που το change είναι μικρότερο του 0.1 η ακτίνα επιρροής r_{min} μειώνεται μέσω της σχέσης (3.7)

Γραμμές 37-39

Υποχρεωτικός τερματισμός των επαναλήψεων για την εκτέλεση της βελτιστοποίησης στην περίπτωση που ο κώδικας φτάσει έναν συγκεκριμένο αριθμό επαναλήψεων, προκειμένου να περιορίσουμε χρονικά την όλη διαδικασία. Αυθαίρετα έχει επιλεγεί στο παρών η τιμή των εκατό επαναλήψεων, μετά τις οποίες το change τίθεται μικρότερο του ορίου ώστε να μην ικανοποιείται η συνθήκη της γραμμής 16.

Η συνάρτηση για την εκτέλεση της στατικής ανάλυσης με τα τριγωνικά πεπερασμένα στοιχεία είναι η FEA(Kelements,nodes,elemConn,x,penal,fixedDOFs,F) , για την κλήση της οποίας χρειάζεται η εισαγωγή σε αυτήν των παρακάτω μεταβλητών

Kelements : τρισδιάστατο μητρώο το οποίο περιέχει τα 30 x 30 μητρώα δυσκαμψίας όλων των στοιχείων, είναι δηλαδή διαστάσεως 30 x 30 x nel, θεωρώντας πως έχουμε πυκνότητα 1.

nodes : οι κόμβοι στο καθολικό σύστημα

elemConn : οι συνδέσεις των κόμβων για το κάθε στοιχείο

x : οι πυκνότητες των στοιχείων

penal : η δύναμη στην οποία υψώνεται η πυκνότητα των στοιχείων

fixedDOFs : το διάνυσμα των δεσμευμένων βαθμών ελευθερίας του φορέα

F : το διάνυσμα των επικομβίων δράσεων του φορέα

Γραμμές 2-4

Υπολογισμός κάποιων χρήσιμων μεταβλητών για μετέπειτα ευκολία και εκκίνηση του συνολικού μητρώου ακαμψίας.

Γραμμές 5-12

Μέσω μιας επανάληψης που τρέχει για όλα τα στοιχεία του φορέα εισάγεται η ακαμψία καθενός στους επηρεαζόμενους βαθμούς ελευθερίας του καθολικού μητρώου ακαμψίας του φορέα. Προκειμένου να ληφθεί υπόψη η πυκνότητα του καθενός στην δυσκαμψία που παρέχει πολλαπλασιάζονται τα εκάστοτε μητρώα δυσκαμψίας με την πυκνότητα του τρέχοντος στοιχείου υψωμένη στη δύναμη penal. Η τακτική αυτή σκοπό έχει να 'τιμωρήσει' τα στοιχεία με μικρή πυκνότητα και να εκμηδενίσει σχεδόν την επίδραση τους στην ακαμψία του φορέα, χωρίς όμως να δημιουργεί σειρές ή στήλες με μηδενικά που θα οδηγήσουν σε μηδενική ορίζουσα και ιδιόμορφο μητρώο ακαμψίας φορέα. Μεταξύ των γραμμών 7 έως 11

λαμβάνει χώρα ένας έλεγχος της πυκνότητας για την αποφυγή πολύ μικρών δυσκαμψιών των στοιχείων που θα προκαλέσουν αριθμητικό πρόβλημα. Αν και η στρατηγική αυτή δεν έχει και τόσο νόημα στην matlab γιατί δεν παρουσιάζονται τα προβλήματα για τα οποία αυτή ακολουθείται, παρόλα αυτά έχει ενσωματωθεί στον κώδικα προκειμένου να παρουσιαστεί το σημείο και ο τρόπος εφαρμογής της όπως γίνεται στον κώδικα του MSolve.

Γραμμές 13-14

Προκειμένου να βρεθούν οι αδέσμευτοι βαθμοί ελευθερίας για τους οποίους θα επιλυθεί η εξίσωση ισορροπίας του φορέα δημιουργείται ένα διάνυσμα με όλους τους βαθμούς ελευθερίας και από αυτούς αφαιρούνται όσοι βρίσκονται στο διάνυσμα fixedDOFs, δημιουργώντας το ζητούμενο διάνυσμα freeDOFs.

Γραμμές 15-16

Επιλύεται τελικώς η εξίσωση ισορροπίας και υπολογίζονται οι επικόμβιες μετατοπίσεις για τους ελεύθερους κόμβους του φορέα. Το καλό με την εκκίνηση του διανύσματος U ως στήλη μηδενικών στην γραμμή 15 είναι πως ήδη είναι έτοιμες οι μετατοπίσεις των δεσμευμένων βαθμών ελευθερίας, και στην γραμμή 16 εισάγονται και οι υπόλοιποι στις θέσεις τους.

Η συνάρτηση για την εφαρμογή του φίλτρου στις ευαισθησίες των στοιχείων είναι η checkSIMP(connect,centerPoints,rmin,x,dc). Στην ουσία, το φίλτρο για το οποίο γίνεται λόγος είναι μια μέθοδος ομαλοποίησης των ευαισθησιών, λαμβάνοντας υπόψη για το εκάστοτε στοιχείο τα γύρω του με έναν συντελεστή βαρύτητας για το καθένα. Οι μεταβλητές που απαιτούνται κατά την κλήση της είναι

connect: το μητρώο που περιέχει για κάθε στοιχείο τα ID όσων από τα υπόλοιπα βρίσκονται εντός της ακτίνας επιρροής του
 centerPoints: το μητρώο που περιέχει τα κέντρα βάρους των τριγωνικών πεπερασμένων
 rmin: η ακτίνα επιρροής του φίλτρου
 x: το διάνυσμα με τις πυκνότητες των στοιχείων
 dc: το διάνυσμα με τις ευαισθησίες των στοιχείων

Γραμμές 2-3

Υπολογισμός του συνολικού αριθμού στοιχείων nel και εκκίνηση του διανύσματος για τον υπολογισμό των φιλτραρισμένων ευαισθησιών

Γραμμές 4-5

Έναρξη των επαναλήψεων για όλα τα στοιχεία και δημιουργία της μεταβλητής sum.

Γραμμές 6-12

Εδώ ξεκινάει μια δεύτερη επανάληψη για την εύρεση των j στοιχείων εκείνων που βρίσκονται εντός της ακτίνας επιρροής του τρέχοντος i στοιχείου από την προηγούμενη επανάληψη. Για τον λόγο αυτό ελέγχονται οι θέσεις του μητρώου connect αν είναι μη μηδενικές, αν έχει δηλαδή αποθηκευθεί σε αυτές το ID του συνδεδεμένου στοιχείου. Για τα j στοιχεία που επηρεάζουν το i εκτελούνται οι σχέσεις (3.5) και (3.6) για τον υπολογισμό του συντελεστή βάρους και των φιλτραρισμένων ευαισθησιών τους.

Η συνάρτηση για την εφαρμογή των κριτηρίων βελτιστοποίησης (Optimality Criteria) είναι η $OC(x, \nu, dc)$, για την κλήση της οποίας χρειάζονται οι παρακάτω μεταβλητές

x : το διάνυσμα με τις πυκνότητες των στοιχείων

ν : ο λόγος του όγκου που θα πρέπει να έχει ο βέλτιστος φορέας ως προς τον συνολικό διατιθέμενο όγκο

dc : το διάνυσμα με τις ευαισθησίες των στοιχείων

Γραμμές 2-5

Υπολογισμός κάποιων αρχικών παραμέτρων όπως ο αριθμός των στοιχείων, οι μεταβλητές $I1$ και $I2$ για τον αλγόριθμο εύρεσης του λ , και το όριο βήματος για την ενημέρωση των πυκνοτήτων $move$.

Γραμμές 6-13

Εφαρμογή των κριτηρίων βελτιστοποίησης των σχέσεων (3.2) έως ώτου να προκύψει λύση η οποία όμως θα ικανοποιεί και τον περιορισμό όγκου ν της κατασκευής. Το όριο για την ολοκλήρωση των επαναλήψεων στην γραμμή 6 είναι πιο αυστηρό από το αντίστοιχο της δημοσίευσης στο οποίο βασίζεται. Ο σκοπός της μείωσης του ορίου είναι η αύξηση των απαιτούμενων επαναλήψεων μέχρι την εύρεση της ενημερωμένης λύσης πυκνοτήτων, κάτι που παρατηρήθηκε πως απαιτείται στην περίπτωση μη δομημένου πλέγματος πολλών στοιχείων.

Η συνάρτηση για τον υπολογισμό των συνδέσεων μεταξύ των στοιχείων ανάλογα με την ακτίνα επιρροής $Connections(centerPoints, rmin)$, όπως και η συνάρτηση για την εύρεση των βαθμών ελευθερίας κάθε στοιχείου $ElementDOFs(elemConn)$ είναι αρκετά σαφείς ως προς την λειτουργία τους και τις εσωτερικές τους διαδικασίες, συνεπώς απλά παρατίθενται στο τέλος του Παραρτήματος Β χωρίς να χρειάζεται να γίνει κάποια περαιτέρω αναφορά εδώ.

3.3 Βελτιστοποίηση τοπολογίας με την μέθοδο BESO

3.3.1 Μαθηματική Διατύπωση της BESO

Η δεύτερη μέθοδος βελτιστοποίησης τοπολογίας που εξετάζεται στα πλαίσια της παρούσας διπλωματικής εργασίας είναι η μέθοδος BESO, το ακρωνύμιο της οποίας σημαίνει Bi-directional Evolutionary Structural Optimization. Όπως γρήγορα θα παρατηρήσει κανείς, η συγκεκριμένη μέθοδος έχει πάρα πολλές ομοιότητες με την μέθοδο SIMP που εξετάσαμε προηγουμένως, κυρίως ως προς τον τρόπο που προσεγγίζει το πρόβλημα και στις διαδικασίες που χρησιμοποιεί για την επίλυση του. Η βασική διαφορά της είναι πως δεν αντιλαμβάνεται παρά δυο καταστάσεις του υλικού, την ύπαρξη συμπαγούς υλικού και την μη ύπαρξη υλικού, δηλαδή το κενό. Αυτή η δυαδική προσέγγιση της κατάστασης υλικού των στοιχείων βρίσκεται σε πλήρη αντιδιαστολή με την προσέγγιση τους από την SIMP μέθοδο, σύμφωνα με την οποία όπως είδαμε μπορεί έχουν οποιαδήποτε πυκνότητα μεταξύ της κατάστασης πλήρους υλικού, ισοδύναμης με μοναδιαία πυκνότητα, και του κενού. Η εύρεση λύσεων αποτελούμενων αποκλειστικά από περιοχές συμπαγούς υλικού και κενών, και κατά συνέπεια λύσεων υλοποιήσιμων και στην πραγματικότητα, είναι που δίνει ένα μεγάλο πλεονέκτημα στην BESO μέθοδο έναντι της SIMP όταν πρόκειται για προβλήματα βελτιστοποίησης σε πρακτικές, αληθινές συνθήκες, για φορείς που θέλουμε να κατασκευασθούν. Το πλεονέκτημα ετούτο συνοδεύεται βέβαια και με κάποια μειονεκτήματα και εξαρτήσεις από τις αρχικές συνθήκες του προβλήματος, όπως θα δούμε στην πορεία.

Στη συνέχεια του παρόντος κεφαλαίου παρουσιάζονται οι μαθηματικές σχέσεις στις οποίες βασίζεται η μέθοδος BESO, ενώ στο τέλος θα παρουσιαστεί και πάλι ο κώδικας matlab που γράφηκε για την κατανόηση και της αλγοριθμικής της πλευράς. Όπως και στην περίπτωση της SIMP μεθόδου, επιλέχθηκε μια από τις πιο διαδεδομένες δημοσιεύσεις επί του θέματος και έγινε προσπάθεια να εφαρμοστεί η μεθοδολογία της σε μη δομημένα πλέγματα αποτελούμενα από τα τριγωνικά στοιχεία που αναπτύχθηκαν. Προκειμένου να είναι πιο ευδιάκριτα τα κοινά σημεία με την δημοσίευση, αλλά και τα σημεία που επιλέχθηκε διαφορετική πορεία αντιμετώπισης του προβλήματος, ο κώδικας που γράφηκε σε matlab έχει κατά το δυνατόν όμοια μορφή με αυτό της δημοσίευσης. Η δημοσίευση που επιλέχθηκε ως πρότυπη για την μέθοδο είναι η 'A further review of ESO type methods for topology optimization' των Xiaodong Huang και Yi-Min Xie.

Η μέθοδος Evolutionary Structural Optimization(ESO) αρχικώς αναπτύχθηκε από τους Xie και Steven(1992,1993,1997), βασισμένη στην εμπειρική ιδέα πως μια κατασκευή κινείται προς την βελτιστοποίηση της αφαιρώντας σταδιακά κάποια από τα στοιχεία της. Η Bi-directional Evolutionary Structural Optimization(BESO) είναι μια επέκταση της ιδέας αυτής, και που επιτρέπει την εισαγωγή νέων στοιχείων σε θέσεις δίπλα σε εκείνα τα στοιχεία με τις μεγαλύτερες τάσεις. Σημειωτέον πως όταν γίνεται λόγος, τόσο εδώ όσο και παρακάτω, για προσθήκη ή αφαίρεση στοιχείων δεν εννοούμε την αλλαγή του πλέγματος των στοιχείων αλλά την εξίσωση της πυκνότητας με ένα για τα στοιχεία που θέλουμε να προσθέσουμε και με μηδέν για αυτά που θέλουμε να αφαιρέσουμε.

Οι ESO/BESO μέθοδοι έχουν χρησιμοποιηθεί σε πλήθος εφαρμογών και πολλές δημοσιεύσεις έχουν γίνει επί του θέματος. Παρόλα αυτά έχουν σημειωθεί κάποια μειονεκτήματα της μεθόδου στην υπάρχουσα βιβλιογραφία. Πρώτο εξ αυτών είναι η αποτυχία των αυθεντικών ESO/BESO μεθόδων στο να συγκλίνουν σε μια βέλτιστη λύση. Αντ' αυτού μετά το τέλος των επαναλήψεων πρέπει να ελεγχθεί το σύνολο τους και να επιλεγεί η βέλτιστη εξ αυτών. Επιπλέον, δεν είναι εύκολο να εφαρμοστούν οι εν λόγω μέθοδοι σε προβλήματα βελτιστοποίησης με στόχο την ελαχιστοποίηση της μετακίνησης

ενός συγκεκριμένου βαθμού ελευθερίας, πέραν δηλαδή της κλασσικής διατύπωσης περί ελαχιστοποίησης της συνολικής ελαστικής παραμόρφωσης.

Κατά την ανωτέρω δημοσίευση, ένα πρόβλημα βελτιστοποίησης με την BESO μέθοδο διατυπώνεται ως ελαχιστοποίηση του όρου της ελαστικής παραμόρφωσης

$$C = \frac{1}{2} u^T K u$$

(3.8)

$$\text{Υποκείμενη σε: } V^* - \sum_{i=1}^N V_i x_i = 0$$

$$x_i = x_{min} \text{ or } 1$$

όπου η ελαστική παραμόρφωση είναι η μισή από αυτήν που χρησιμοποιείται στην μέθοδο SIMP. V_i είναι ο όγκος (ή η επιφάνεια στην περίπτωση μας) ενός μεμονωμένου στοιχείου και V^* είναι ο όγκος που θέλουμε να καταλαμβάνει ο βέλτιστος φορέας. Όπως και πριν, με N συμβολίζεται ο αριθμός των πεπερασμένων στοιχείων που χρησιμοποιούνται και με x_i η πυκνότητα του κάθε στοιχείου, η οποία στην προκειμένη περίπτωση μπορεί να πάρει μονάχα τις τιμές x_{min} και 1. Το x_{min} συνήθως είναι περίπου 0.001, ή κάποιος άλλος επίσης πολύ μικρός αριθμός, ώστε να αγνοείται στην ουσία η επίδραση όσων στοιχείων έχουν τέτοια πυκνότητα αλλά να μην δημιουργείται και αριθμητικό πρόβλημα κατά την επίλυση λόγω της ιδιομορφικότητας του καθολικού μητρώου στιβαρότητας.

Η διαχείριση των μητρώων δυσκαμψίας και του υλικού των στοιχείων είναι ανάλογη αυτής της μεθόδου SIMP, μιας και οι ιδιότητες των στοιχείων με μικρή πυκνότητα (για την ακρίβεια με x_{min} στην BESO) 'τιμωρούνται' με τον πολλαπλασιασμό επί την πυκνότητα τους υψωμένη σε μια δύναμη p (penalization) για την απαλοιφή τους πρακτικά από το μητρώο στιβαρότητας. Σε μαθηματική μορφή η εν λόγω τακτική διαχείρισης είναι

$$E(x_i) = E x_i^p \quad (3.9)$$

Για την επιλογή των στοιχείων εκείνων που θα αποτελέσουν τελικώς τον φορέα λαμβάνεται υπόψη η ευαισθησία της αντικειμενικής συνάρτησης σε σχέση με την αλλαγή του εκάστοτε στοιχείου, η οποία μπορεί να βρεθεί ως

$$\frac{\partial C}{\partial x_e} = -\frac{p x_e^{p-1}}{2} u_e^T K u_e \quad (3.10)$$

Η BESO μέθοδος, όντας τόσο κοινή με την SIMP στον τρόπο διαχείρισης του προβλήματος βελτιστοποίησης, αντιμετωπίζει και αυτή προβλήματα ανάπτυξης λύσεων με μοτίβα σκακιέρας, με εναλλάξ περιοχές δηλαδή υλικού και κενών. Προκειμένου να αντιμετωπιστεί αυτό το πρόβλημα, κατά την επίλυση εφαρμόζεται και εδώ ένα φίλτρο στις ευαισθησίες των στοιχείων το οποίο αποσκοπεί στην μεταξύ τους ομαλοποίηση. Μάλιστα, αυτό το φίλτρο είναι αρκετά παρόμοιο με το αντίστοιχο που χρησιμοποιήθηκε από την SIMP μέθοδο στην δημοσίευση του O.Sigmund, όπως φαίνεται από την παρακάτω σχέση

$$\widehat{\frac{\partial C}{\partial x_e}} = \frac{1}{\sum_{f=1}^N \widehat{H}_f} \sum_{f=1}^N \widehat{H}_f \frac{\partial C}{\partial x_e} \quad (3.11)$$

με τον συντελεστή \widehat{H}_f να βρίσκεται και πάλι στον ρόλο ενός συντελεστή βάρους ο οποίος υπολογίζεται ως

$$\widehat{H}_f = r_{min} - dist(e, f), \quad (3.12)$$

$$\{f \in N \mid dist(e, f) \leq r_{min}\}, \quad e = 1, \dots, N,$$

Βέβαια, αν και το φίλτρο μοιάζει σχεδόν ολόιδιο στις δυο περιπτώσεις, παρατηρεί κανείς πως στην εφαρμογή του στην BESO μέθοδο δεν υπάρχει ο πολλαπλασιασμός με την πυκνότητα στα αθροίσματα του αριθμητή και του παρονομαστή. Ο λόγος γι αυτό δεν έχει να κάνει με την υιοθέτηση κάποιας διαφορετικής τακτικής για το φιλτράρισμα των ευαισθησιών, αλλά με το γεγονός πως ο όρος αυτός δεν συνεισφέρει σε κάτι στην προκειμένη περίπτωση αφού παίρνει τις διακριτές τιμές μηδέν και ένα. Εάν είναι ένα προφανώς δεν υπάρχει κάποια ουσία στην εύρεση γινομένου μαζί του, ενώ αν είναι μηδέν θα έχει ήδη μηδενιστεί ο όρος της ευαισθησίας πριν την εφαρμογή του φίλτρου από την σχέση (3.10), οπότε και πάλι δεν έχει νόημα ο πολλαπλασιασμός μαζί του.

3.3.2 Εφαρμογή της BESO μεθόδου σε μη δομημένα πλέγματα

Η αλήθεια είναι πως εδώ πέρα δεν υπάρχουν και πολλά πράγματα να ειπωθούν. Κι αυτό όχι γιατί η μέθοδος δεν αντιμετωπίζει κανένα πρόβλημα κατά την εφαρμογή της σε μη δομημένα πλέγματα πεπερασμένων στοιχείων, αλλά επειδή η μεθοδολογία που παρουσιάστηκε στην εφαρμογή της μεθόδου SIMP σε μη δομημένα πλέγματα εφαρμόζεται σχεδόν επακριβώς και στην περίπτωση της BESO μεθόδου. Πρόκειται για ακόμα μια απόδειξη της ομοιότητας των δυο μεθόδων, αφού διαχειρίζονται το πρόβλημα με τόσο κοινές διαδικασίες που δεν χρειάστηκε να γραφεί νέος κώδικας για την BESO στα σημεία που αφορούν την εφαρμογή επί του μη δομημένου πλέγματος πεπερασμένων στοιχείων. Το μοναδικό σημείο διαφοροποίησης μεταξύ τους είναι οι θέσεις στον κώδικα στις οποίες υπάρχει η ανάγκη εκ νέου υπολογισμού των συνδέσεων των στοιχείων του φορέα. Για την BESO αρκεί αυτό να γίνει στις αρχές του αλγορίθμου μια φορά, και στη συνέχεια τα αποτελέσματα διατηρούνται σε πίνακα και χρησιμοποιούνται τα ίδια κάθε φορά που χρειάζονται οι σχετικές θέσεις των στοιχείων στους υπολογισμούς. Αυτό γίνεται επειδή ο αλγόριθμος δε διαχειρίζεται πυκνότητες για τα στοιχεία, και ως εκ τούτου δεν εμφανίζεται σε τόσο μεγάλο βαθμό το πρόβλημα της παρασιτικής επιβίωσης υλικού σε θέσεις που δεν θα έπρεπε. Έτσι δεν υπάρχει λόγος η ακτίνα να μεταβάλλεται με το πέρασμα των επαναλήψεων, επομένως δε χρειάζεται να ξαναγίνει υπολογισμός ούτε και των σχετικών θέσεων των στοιχείων.

3.3.3 Παρουσίαση κώδικα matlab για την βελτιστοποίηση τοπολογίας με την BESO μέθοδο

Για την κατανόηση της αλγοριθμικής πλευράς της μεθόδου γράφτηκε κώδικας σε matlab αντίστοιχος με εκείνον που χρησιμοποιήθηκε για τις αναλύσεις και εισήχθη στην γενικότερη προγραμματιστική δομή MSolve. Όπως και στην παρουσίαση του θεωρητικού υποβάθρου της μεθόδου, έχει γίνει προσπάθεια να είναι κατά το δυνατόν όμοιος με αυτόν της δημοσίευσης στην οποία στηρίχθηκε, αλλά και στην μέθοδο SIMP με την οποία τόσο πολύ μοιάζει. Άλλωστε και οι συγγραφείς της δημοσίευσης ‘A further review of ESO type methods for topology optimization’ αναφέρουν πως και στο δικό τους κώδικα που παρουσιάζουν έχει γίνει προσπάθεια να είναι στο μεγαλύτερο δυνατό βαθμό όμοιος με αυτόν της δημοσίευσης ‘‘A 99 line topology optimization code written in Matlab’’ του O. Sigmund ώστε να είναι εμφανή τόσο τα σημεία που διαχειρίζονται το πρόβλημα με διαφορετική μεθοδολογία όσο και αυτά που οι δυο μέθοδοι ταυτίζονται.

Η συνάρτηση με την οποία εκτελείται η βελτιστοποίηση τοπολογίας με εφαρμογή της BESO μεθόδου είναι η `TopologyBESO(nodes,elemConn,er,volfrac,rmin,penal,E,v,thickness,fixedDOFs,F)`. Οι μεταβλητές που εισάγονται κατά την κλήση της είναι οι

`nodes`: οι κόμβοι στο καθολικό σύστημα

`elemConn`: οι συνδέσεις των κόμβων για το κάθε στοιχείο

`er`: ρυθμός μεταβολής του όγκου του φορέα για κάθε επανάληψη της διαδικασίας

`volfrac`: ο λόγος του όγκου που θα πρέπει να έχει ο βέλτιστος φορέας ως προς τον συνολικό διατιθέμενο όγκο

`rmin`: η ακτίνα επιρροής του φίλτρου

`penal`: η δύναμη στην οποία υψώνεται η πυκνότητα των στοιχείων

`E`: το μέτρο ελαστικότητας κατά Young του υλικού

`v`: ο λόγος Poisson του υλικού

`thickness`: το πάχος των τριγωνικών στοιχείων

`fixedDOFs`: το διάνυσμα των δεσμευμένων βαθμών ελευθερίας του φορέα

`F`: το διάνυσμα των επικομβίων δράσεων του φορέα

Γραμμές 3-9

Αρχικοποίηση μερικών από τις απαραίτητες για το πρόβλημα μεταβλητές όπως ο αριθμός των στοιχείων `nel`, η μεταβολή `change` μεταξύ των επαναλήψεων, το διάνυσμα `x` ύπαρξης ή όχι υλικού στα στοιχεία του πλέγματος, το διάνυσμα `dc` των ευαισθησιών των στοιχείων, ο όγκος `v` του φορέα, η τρέχουσα επανάληψη `loop` της μεθόδου και το διάνυσμα `c` στο οποίο θα αποθηκεύεται η ελαστική παραμόρφωση κάθε επανάληψης. Η ιδιαιτερότητα με αυτό το διάνυσμα `c` είναι πως από πριν πρέπει να οριστεί το συνολικό του μέγεθος μιας και η συνεχής αλλαγή του μεγέθους πινάκων οδηγεί σε πιο αργό κώδικα(ή μηνύματα σφάλματος σε πιο αυστηρές γλώσσες), μια ιδιαιτερότητα όμως που μπορεί να παρακαμφθεί στην περίπτωση γραφής σε C# με χρήση λίστας για την αποθήκευση της ελαστικής παραμόρφωσης κάθε επανάληψης. Το μέγεθος ορίζεται να είναι 200, μιας και συνήθως μέσα στις 150 θα έχουμε λύση.

Γραμμές 11-17

Υπολογισμός του 30x30 μητρώου δυσκαμψίας κάθε στοιχείου με χρήση της συνάρτησης `ElementMohrTri6` του Παραρτήματος A, με θεώρηση πλήρους

υλικού(πυκνότητα 1), και το αποθηκεύουμε στο τρισδιάστατο μητρώο Kelements. Επίσης δημιουργείται το μητρώο των κεντροβαρικών σημείων για τα τριγωνικά στοιχεία centerPoints, το οποίο εισάγεται στην συνάρτηση Connections μαζί με την ακτίνα επιρροής του φίλτρου rmin για να δημιουργηθεί το μητρώο connect, το οποίο με την σειρά του περιέχει για κάθε στοιχείο εκείνα τα γύρω του στοιχεία που θα επηρεάσουν την ευαισθησία του με την δικιά τους.

Γραμμές 19-20

Εκκίνηση των επαναλήψεων για την εφαρμογή της μεθόδου BESO στην βελτιστοποίηση τοπολογίας του φορέα και δημιουργία του διανύσματος dc στο οποίο θα αποθηκευτούν οι ευαισθησίες dc των στοιχείων κατά την τρέχουσα επανάληψη.

Γραμμή 21

Υπολογισμός του όγκου που θα πρέπει να επιτευχθεί για τον φορέα από την τρέχουσα επανάληψη, ο οποίος μειώνεται κατά ποσοστό er σε κάθε καινούργια επανάληψη ξεκινώντας από τον συνολικό όγκο του φορέα, δηλαδή τιμή 1 για τον όγκο vol, και καταλήγοντας στην τιμή volfrac που έχουμε εισάγει ως απαιτούμενο όγκο για τον βέλτιστο φορέα.

Γραμμές 22-24

Αποθήκευση των ευαισθησιών που προέκυψαν από την προηγούμενη επανάληψη, εφόσον υπάρχει προηγούμενη, ώστε να ληφθεί ο μέσος όρος μεταξύ της τρέχουσας και της προηγούμενης ευαισθησίας των στοιχείων.

Γραμμή 26

Εκτέλεση της ανάλυσης πεπερασμένων στοιχείων με χρήση της συναρτήσεως FEA, η οποία παρουσιάστηκε και στην επεξήγηση του κώδικα για την μέθοδο SIMP(κεφάλαιο 3.2.3) και συνεπώς δε αναλυθεί στο παρών κεφάλαιο(αν και παρατίθεται στο Παράρτημα Β για πληρότητα).

Γραμμές 27-32

Εύρεση για κάθε στοιχείο των καθολικών επικόμβιων βαθμών ελευθερίας του και αποθήκευση τους στο διάνυσμα elDOFs. Το διάνυσμα αυτό χρησιμοποιείται έπειτα για την εξαγωγή από το διάνυσμα U των επικόμβιων μετατοπίσεων που αντιστοιχούν στους κόμβους του εκάστοτε στοιχείου, και την αποθήκευσή τους στο διάνυσμα Ue. Με χρήση του διανύσματος Ue υπολογίζεται η συνεισφορά της ελαστικής παραμόρφωσης του στοιχείου στην συνολική ελαστική παραμόρφωση c(loop) του φορέα της τρέχουσας επανάληψης και η ευαισθησία dc(i) για το κάθε στοιχείο.

Γραμμές 34-38

Εφαρμογή του φίλτρου για την ομαλοποίηση των ευαισθησιών με σκοπό την αποφυγή μοτίβων σκακιάρας στον φορέα. Το φίλτρο εφαρμόζεται μέσω της συνάρτησης checkBESO(connect,centerPoints,rmin,dc) που αποτελεί αλγοριθμική εφαρμογή της σχέσης(3.6), η οποία στο μεγαλύτερο της μέρος είναι ίδια με την checkSIMP αν εξαιρέσουμε την απαλοιφή του όρου x. Στην περίπτωση που υπάρχει προηγούμενη επανάληψη της μεθόδου λαμβάνεται ως ευαισθησία των στοιχείων ο μέσος όρος μεταξύ των τρεχουσών ευαισθησιών και των προηγουμένως υπολογιζόμενων.

Γραμμή 40

Εφαρμογή των Optimality Criteria με την μορφή που παίρνουν για την περίπτωση της BESO μεθόδου, με χρήση της συνάρτησης ADDDEL(dc,vol). Υπάρχουν κάποιες διαφορές ως προς τον τρόπο που βρίσκεται η τιμή του x διανύσματος που

αντιστοιχεί σε κάθε στοιχείο, οι οποίες όμως θα συζητηθούν παρακάτω κατά την παρουσίαση της συνάρτησης ADDDEL.

Γραμμές 41-43

Έλεγχος για το αν έχει επιτευχθεί ή όχι σύγκλιση της μεθόδου. Ο έλεγχος γίνεται μόνο στην περίπτωση που έχουν προηγηθεί δέκα επαναλήψεις, αφού το κριτήριο σύγκλισης λαμβάνει υπόψη τις ελαστικές παραμορφώσεις του φορέα για τις τελευταίες δέκα επαναλήψεις που έχουν γίνει. Παρατηρούμε εδώ μια τελείως διαφορετική προσέγγιση του ελέγχου συγκλίσεως μεταξύ της SIMP και της BESO, αφού η πρώτη ελέγχει αν η μέγιστη διαφορά που υπάρχει μεταξύ των πυκνοτήτων των στοιχείων, από βήμα σε βήμα της μεθόδου, βρίσκεται κάτω από κάποιο όριο. Από την άλλη, μιας και η BESO δεν διαχειρίζεται πυκνότητες για τα στοιχεία αλλά μπορεί να αντιληφθεί μόνο την δυαδική κατάσταση ύπαρξης υλικού ή κενού, δεν μπορεί να εφαρμοστεί ένα ανάλογο κριτήριο σύγκλισης. Αντί αυτού, γίνεται έλεγχος για το αν η ποσοστιαία μέση αύξηση της ελαστικής παραμόρφωσης κατά τις τελευταίες πέντε επαναλήψεις σε σύγκριση με την μέση ελαστική παραμόρφωση των προτελευταίων πέντε βρίσκεται κάτω από κάποιο όριο. Συνοπτικώς δηλαδή, η SIMP ολοκληρώνεται όταν δεν υπάρχει μεγάλη αλλαγή στις πυκνότητες των στοιχείων, ενώ η BESO ολοκληρώνεται όταν η ελαστική παραμόρφωση του φορέα είναι περίπου σταθερή για δέκα συνεχόμενες επαναλήψεις.

Η συνάρτηση για την εφαρμογή του φίλτρου στις ευαισθησίες των στοιχείων καλείται με την εντολή `checkBESO(connect,centerPoints,rmin,dc)`. Κατά την κλήση της απαιτείται η εισαγωγή των παρακάτω μεταβλητών
`connect`: το μητρώο που περιέχει για κάθε στοιχείο τα ID όσων από τα υπόλοιπα βρίσκονται εντός της ακτίνας επιρροής του
`centerPoints`: το μητρώο που περιέχει τα κέντρα βάρους των τριγωνικών πεπερασμένων
`rmin`: η ακτίνα επιρροής του φίλτρου
`dc`: το διάστημα με τις ευαισθησίες των στοιχείων

Γραμμές 2-3

Υπολογισμός του συνολικού αριθμού στοιχείων `nel` του πλέγματος, και δημιουργία του διάνυσματος `dcp` όπου θα αποθηκευτούν οι ευαισθησίες μετά την ομαλοποίησή τους από το φίλτρο.

Γραμμές 4-5

Εκκίνηση της επανάληψης ώστε να γίνουν οι απαραίτητοι υπολογισμοί για όλα τα στοιχεία `i`, και αρχικοποίηση της μεταβλητής `sum` για τον υπολογισμό του αθροίσματος που χρειάζεται για την σχέση (3.12)

Γραμμές 6-12

Εκκίνηση δεύτερης επανάληψης η οποία τρέχει για όλα τα στοιχεία `j` με σκοπό την εύρεση όσων βρίσκονται εντός της ακτίνας `rmin` επιρροής του στοιχείου `i` από τον προηγούμενο επαναληπτικό βρόχο. Εφόσον το στοιχείο `j` όντως βρίσκεται εντός της ακτίνας επιρροής του `i`, εφαρμόζεται η σχέση (3.11) και υπολογίζεται η συνεισφορά του στο άθροισμα `sum`.

Γραμμή 13

Ουσιαστικά εδώ είναι που ολοκληρώνεται η εφαρμογή του φίλτρου, με την διαίρεση του κάθε όρου ευαισθησίας με το άθροισμα `sum`, όπως φαίνεται και στην σχέση (3.11)

3.4 Βελτιστοποίηση τοπολογίας με την μέθοδο Level Set

Η τελευταία μέθοδος βελτιστοποίησης τοπολογίας που θα εφαρμοστεί και θα αποτιμηθεί στ παρών έργο είναι η Level Set. Πρόκειται για μια εντελώς διαφορετική και αρκετά πιο σύνθετη, ιδίως κατά την εφαρμογή της σε μη δομημένα πλέγματα, προσέγγιση της εύρεσης μιας μορφής φορέα για την ανάληψη δεδομένης στατικής φορτίσεως. Ο διακριτοποιημένος φορέας μοντελοποιείται με μια συνάρτηση επιφάνειας, από την διαφόριση και επεξεργασία της οποίας θα προκύψει στο τέλος η επιλογή εκείνων των στοιχείων που θα συνθέσουν την λύση του προβλήματος βελτιστοποίησης. Η επιφάνεια αυτή μπορεί να αναπαρασταθεί στον χώρο λαμβάνοντας ως ύψος των σημείων της για κάθε θέση την τιμή της συναρτήσεως μοντελοποίησης για το εν λόγω σημείο. Η λύση για κάθε επανάληψη της μεθόδου αποτελείται από το μέρος εκείνο της επιφάνειας που κατά την τομή με ένα οριζόντιο επίπεδο σε σταθερό, δεδομένο για κάθε επανάληψη, ύψος βρίσκεται άνω (ή κάτω, ανάλογα με την διατύπωση των μαθηματικών σχέσεων για την εφαρμογή της μεθόδου) της τομής αυτής. Εξ ου άλλωστε και το όνομα της μεθόδου Level Set, αφού προσπαθεί να βρει την λύση στο πρόβλημα της βελτιστοποίησης τοπολογίας με την σύγκριση της επιφάνειας με σταθερό δεδομένο επίπεδο. Παρακάτω θα εξετάσουμε τον τρόπο μοντελοποίησης και αντιμετώπισης ενός στατικού φορέα, καθώς και την επεξεργασία στην οποία θα τον υποβάλλουμε κατά τις επαναλήψεις της μεθόδου.

Όπως και η μέθοδος BESO, έτσι και η Level Set αντιλαμβάνεται με δυαδικό τρόπο την κατάσταση του υλικού των στοιχείων, μεταβάλλοντάς την μεταξύ της ύπαρξης υλικού ή κενού. Καθώς όπως είπαμε η επιλογή ύπαρξης υλικού γίνεται με τον έλεγχο της θέσης των σημείων σχετικώς ως προς ένα οριζόντιο επίπεδο, και επειδή πέραν του ότι πρόκειται για μια συνεχή επιφάνεια έχουν εφαρμοστεί επ' αυτής και φίλτρα ομαλοποίησης των τιμών της, δεν συναντώνται στοιχεία μιας εκ των δύο φύσεων σε μικρού πλήθους συστάδες, αλλά αντ' αυτού έχουμε υπολογίσιμες περιοχές όπου συναντάται μόνο μια από τις δυο καταστάσεις υλικού. Ο λόγος γι αυτό είναι η ομαλότητα της επιφάνειας που σχηματίζεται από την Level Set Function, αφού δε μπορεί να δημιουργήσει απότομους λοφίσκους τοπικών ακρότατων άνω ή κάτω από το επίπεδο ελέγχου αλλά είναι αναγκασμένη να μεταβάλλεται με έως συγκεκριμένου μεγέθους κλίση.

Γι άλλη μια φορά, βασικό ζητούμενο του προβλήματος βελτιστοποίησης είναι η ελαχιστοποίηση τη ελαστικής παραμόρφωσης του φορέα υπό δεδομένη στατική φόρτιση, απαιτώντας συγχρόνως την χρήση συγκεκριμένου ποσοστού υλικού από το διαθέσιμο χώρο για την εφαρμογή της μεθόδου. Η μαθηματική διατύπωση της παραπάνω συνθήκης για την εφαρμογή της μεθόδου, λαμβάνοντας συγχρόνως υπόψη την δυαδική κατάσταση ύπαρξης ή μη υλικού, είναι

$$\min: c(x) = U^T K U = \sum_{e=1}^N u_e^T k_e u_e$$

$$\text{Υποκείμενη σε : } V(x) = V_{req} \quad (3.13)$$

$$: K U = F$$

$$: x_e = 0 \text{ or } x_e = 1 \forall e = 1, \dots, N$$

όπου $x=(x_1, \dots, x_N)$ είναι το διάνυσμα των 'πυκνοτήτων' των στοιχείων λαμβάνοντας για κάθε πεπερασμένο e την τιμή $x_e=1$ για την προσομοίωση συμπαγούς υλικού ή την τιμή $x_e=0$ για την προσομοίωση κενού, $c(x)$ είναι η αντικειμενική συνάρτηση ελαστικής παραμόρφωσης του φορέα, F και U είναι τα διανύσματα καθολικής επικόμβιας φόρτισης και καθολικών επικόμβιων μετακινήσεων του φορέα και K είναι το μητρώο δυσκαμψίας εκφρασμένο στο καθολικό σύστημα του. Η εξίσωση $KU=F$ είναι η κλασική εξίσωση ελαστικής ισορροπίας που συναντήσαμε και στις προηγούμενες μεθόδους

βελτιστοποίησης, με μέγεθος βαθμών ελευθερίας ανάλογο με την πυκνότητα διακριτοποίησης του πλέγματος. Αντίστοιχα, με u_e και k_e συμβολίζονται το διάνυσμα επικόμβιων μετακινήσεων και το μητρώο δυσκαμψίας για το εκάστοτε πεπερασμένο στοιχείο e . Με $c(x)$ συμβολίζεται και πάλι η αντικειμενική συνάρτηση του προβλήματος βελτιστοποίησης, που όπως και στις προηγούμενες περιπτώσεις είναι η ελαστική παραμόρφωση του φορέα για την οποία γίνεται προσπάθεια ελαχιστοποίησης της.

Η Level Set μέθοδος χρησιμοποιείται για την εύρεση τοπικών ακροτάτων(ελαχίστων) στο πρόβλημα βελτιστοποίησης που τις διατίθεται κάθε φορά, όπως αναφέρεται και για την SIMP όπως θα δούμε παρακάτω. Επιβαρύνοντας το γεγονός πως πρόκειται περί τοπικών ακροτάτων έχουμε την εφαρμογή της μεθόδου σε μη δομημένα πλέγματα, τα οποία ακριβώς λόγω της μη 'ισότιμης' διακριτοποίησης τους ως προς όλες τις δυνατές κατευθύνσεις εύρεσης λύσης οδηγούν πολλές φορές σε αμφιλεγόμενου βέλτιστου μορφές. Η συνάρτηση επιπέδου level set με την οποία γίνεται η κρίση για τα στοιχεία που θα συνθέσουν την λύση καταλαμβάνει έναν επίπεδο τομέα A , μονάχα εντός του οποίου μπορεί να κινηθεί και να λάβει τιμές. Όπως θα δούμε παρακάτω ο τομέας αυτός μπορεί να είναι και μεγαλύτερος του χώρου που καταλαμβάνει το πλέγμα πεπερασμένων που έχει δοθεί στο πρόβλημα, αν και συνήθως καταλαμβάνει ίσα ίσα μια παραπάνω σειρά εικονικών στοιχείων περιμετρικώς του πλέγματος, για λόγους που θα εξηγηθούν παρακάτω. Εάν λοιπόν η κατασκευή σε κάθε βήμα καταλαμβάνει έναν χώρο Ω , υποσύνολο του χώρου A , η συνάρτηση Level Set εντός αυτού του χώρου έχει την ακόλουθη ιδιότητα:

$$\psi(p) \begin{cases} < 0 & \text{εάν } p \in \Omega \\ = 0 & \text{εάν } p \in \partial\Omega \\ > 0 & \text{εάν } p \notin \Omega \end{cases} \quad (3.14)$$

όπου p μπορεί να είναι οποιοδήποτε σημείο του χώρου που καταλαμβάνει το πλέγμα πεπερασμένων, και με $\partial\Omega$ συμβολίζεται το όριο του Ω .

Η ακόλουθη εξίσωση εξέλιξης της Level Set συνάρτησης χρησιμοποιείται για την ανάπτυξη αυτής, και κατ' επέκταση και της κατασκευής:

$$\frac{\partial\psi}{\partial t} = -v|\nabla\psi| - \omega g \quad (3.15)$$

όπου με t συμβολίζεται ο χρόνος, $v(p)$ και $g(p)$ είναι βαθμωτά μεγέθη επί του χώρου κίνησης της συνάρτησης, και ω είναι μια θετική σταθερά με την οποία προσδιορίζεται η επιρροή του όρου g . Το πεδίο v καθορίζει την γεωμετρική κίνηση του ορίου $\partial\Omega$ της κατασκευής και επιλέγεται βάσει της παραγώγου σχήματος του αντικειμένου βελτιστοποίησης. Ο όρος g καθορίζει την σύμπτυξη νέων κενών εντός του χώρου της κατασκευής και επιλέγεται βάσει της τοπολογικής παραγώγου του αντικειμένου της βελτιστοποίησης.

Εάν η παράμετρος ω λάβει την τιμή μηδέν, τότε η σχέση (3.15) παίρνει την μορφή της καθιερωμένης Hamilton-Jacob εξελικτικής εξίσωσης για μια Level Set συνάρτηση ψ με ταχύτητα $v(p)$ κινήσεως του ορίου της. Η κατεύθυνση της ταχύτητας v κινήσεως της συνάρτησης είναι κάθετη ως προς το όριο $\partial\Omega$ του χώρου Ω της κατασκευής, παίρνοντας ως θετική φορά αυτήν που δείχνει από την πλευρά υλικού προς αυτήν του κενού. Η απλούστερη αυτή εξίσωση που δεν περιλαμβάνει τον όρο g συναντάται συνήθως σε προβλήματα βελτιστοποίησης τοπολογίας με εφαρμογή της Level Set μεθόδου(βλέπε Allaire et al.2004 [5], Wang et al.2003 [11]). Ωστόσο, σε επιφανειακά προβλήματα, όπως αυτά που προσπαθούν να αναλυθούν με την χρήση του τριγωνικού στοιχείου πλάκας, η Hamilton-Jacobi εξελικτική εξίσωση έχει το μεγάλο μειονέκτημα πως δε μπορεί να

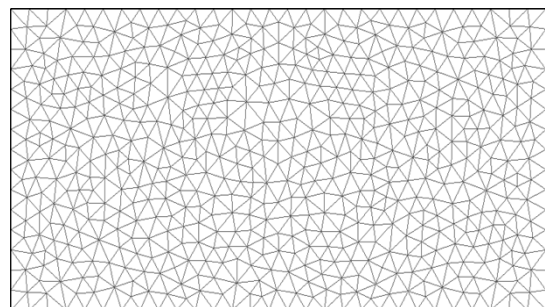
συμπύζει τις νέες περιοχές κενού που δημιουργούνται εντός της κατασκευής κατά την πρόοδο της μεθόδου (Allaire et al. 2004). Στην υλοποίηση της μεθόδου που υιοθετείται στη δημοσίευση της V.J.Challis συμπεριλαμβάνεται και ο πρόσθετος όρος g όπως συστήνεται από Burger et al. (2004), ούτως ώστε οι νέες περιοχές κενού που σχηματίζονται όσο προχωράνε οι επαναλήψεις της μεθόδου, εντός περιοχών υλικού της κατασκευής, να μπορούν να συμπυκνωθούν. Ο λόγος περί συμπύκνωσης νέων περιοχών που περιβάλλονται από της αντίθετης φύσης στοιχεία γίνεται μονάχα για την περίπτωση που αυτές οι νέες περιοχές είναι περιοχές κενού. Το αντίστροφο σενάριο δεν έχει κανένα απολύτως φυσικό νόημα καθώς περιοχές υλικού περιβαλλόμενες από κενό δεν μπορούν να αναλάβουν φορτίο και να συμβάλλουν στην ακαμψία του φορέα, και άρα δε γίνεται προσπάθεια για την αντιμετώπιση του.

Για την δημιουργία ενός αλγόριθμου επίλυσης προβλημάτων βελτιστοποίησης τοπολογίας με την μέθοδο Level Set πρέπει τα απαραίτητα στοιχεία από το πλέγμα πεπερασμένων στοιχείων να μεταφερθούν στην επιφάνεια για να γίνει επ' αυτής η απαραίτητη επεξεργασία. Μιας και η δημιουργία αναλυτικής δισδιάστατης εξίσωσης που να περιγράφει την ζητούμενη επιφάνεια είναι αρκετά δύσκολη υπόθεση και προϋποθέτει την ύπαρξη κατάλληλου αλγορίθμου για την εφαρμογή συνάρτησης επί σημείων (όπως είναι η κλασική fit συνάρτηση της matlab για την εφαρμογή πολυωνυμικών συναρτήσεων σε σημειακά δεδομένα), αλγόριθμοι που δεν συμπεριλαμβάνεται στη δομή MSolve που χρησιμοποιήθηκε, η επιφάνεια προσομοιώνεται από ένα πλέγμα σημείων καθένα από τα οποία πέραν των συντεταγμένων του διαθέτει και την τιμή της Level Set συναρτήσεως στην θέση που βρίσκεται.

Απαραίτητη προϋπόθεση για να μπορέσει να λειτουργήσει η Level Set είναι το πλέγμα των σημείων με τα οποία η μέθοδος αντιλαμβάνεται την επιφάνεια να είναι δομημένο, για λόγους που έχουν να κάνουν με την μαθηματική επεξεργασία της επιφάνειας και θα εξηγηθούν στην πορεία. Όταν το πλέγμα πεπερασμένων στοιχείων αποτελείται από τετραγωνικά στοιχεία, διακριτοποιημένα με τρόπο τέτοιο ώστε για το καθένα να είναι ξεκάθαρο ποιο στοιχείο βρίσκεται από πάνω του, ποιο αριστερά, ποιο δεξιά και ποιο κάτω του (εκτός βεβαίως της περιπτώσεως που ένα στοιχείο βρίσκεται στην περιμετρική συστάδα στοιχείων), η υλοποίηση του αναγκαίου για την μέθοδο δομημένου πλέγματος σημείων είναι εφικτή χωρίς ιδιαίτερη δυσκολία. Ουσιαστικά στην περίπτωση αυτή αρκεί η εύρεση για κάθε στοιχείο του κεντροβαρικού του σημείου το οποίο στην συνέχεια χρησιμοποιείται για την περιγραφή της επιφάνειας, αποτελώντας έτσι έναν άμεσο διάυλο επικοινωνίας μεταξύ της συνάρτησης επιφάνειας Level Set και του εκάστοτε πεπερασμένου στοιχείου για την μεταφορά δεδομένων. Μια τέτοια διακριτοποίηση χρησιμοποιείται σε όλες τις εφαρμογές της μεθόδου που συνάντησε ο γραφών, και γι αυτό η εφαρμογή της μεθόδου Level Set στο μη δομημένο πλέγμα τριγωνικών στοιχείων χρειάστηκε να προσεγγιστεί με διαφορετικό τρόπο από αυτόν που συναντάται στην υπάρχουσα βιβλιογραφία. Στις δυο παρακάτω εικόνες παρατίθενται προς σύγκριση δυο πλέγματα τετραγωνικών στοιχείων, το πρώτο εκ των οποίων είναι δομημένο ενώ το δεύτερο όχι.



Εικόνα 3.1: Δομημένο πλέγμα



Εικόνα 3.2: Μη δομημένο πλέγμα

Προς το παρών θα ασχοληθούμε με την εφαρμογή της μεθόδου σε καθαρά δομημένα πλέγματα πεπερασμένων, όντας απλούστερη στην κατανόηση της και ειδικά όταν πρόκειται για μια πρώτη επαφή με την Level Set, ενώ παρακάτω στο κεφάλαιο .. θα παρουσιαστεί και η στρατηγική που ακολουθήθηκε ώστε να μπορέσει να γίνει βελτιστοποίηση τοπολογίας μη δομημένα διακριτοποιημένων πλεγμάτων με την μέθοδο. Συγχρόνως, στην δημοσίευση 'A discrete level-set topology optimization code written in Matlab', όπου και βασίστηκε όπως προαναφέρθηκε ο σχηματισμός του αλγορίθμου, τα πλέγματα που χρησιμοποιούνται είναι όλα δομημένα, οπότε και βολεύει να παρουσιαστεί αρχικώς εξ' ολοκλήρου η εν λόγω μεθοδολογία.

Κατά κάποιο τρόπο, από την στιγμή που αντιλαμβανόμαστε την επιφάνεια ως ένα πλέγμα σημείων, έχουμε εφαρμόσει μια διακριτοποίηση αυτής με πυκνότητα ανάλογη με την πυκνότητα τοποθέτησης των σημείων. Προφανώς στην περίπτωση που τα σημεία αυτά βρίσκονται στα κέντρα βάρους των στοιχείων που συνθέτουν το πλέγμα πεπερασμένων η πυκνότητα αυτή θα είναι η ίδια με την πυκνότητα διακριτοποίησης των πεπερασμένων. Αντ' αυτού, όπως θα δούμε παρακάτω, στην εφαρμογή της μεθόδου σε μη δομημένο πλέγμα πεπερασμένων η διακριτοποίηση του είναι ανεξάρτητη της πυκνότητας που θα έχει το πλέγμα σημείων για την προσομοίωση της συναρτήσεως επιφανείας Level Set, και για την ακρίβεια η πυκνότητα αυτή δίνεται ως ξεχωριστή παράμετρος από τον χρήστη σε κάθε εφαρμογή της μεθόδου.

Έτσι λοιπόν εάν με c_e συμβολίσουμε το σημείο που αντιστοιχεί στο κέντρο βάρους ενός τετραγωνικού στοιχείου e τότε η συνάρτηση της Level Set ικανοποιεί την παρακάτω σχέση

$$\psi(c_e) \begin{cases} < 0 & \text{εάν } x_e = 1 \\ > 0 & \text{εάν } x_e = 0 \end{cases} \quad (3.16)$$

Η διακριτοποιημένη Level Set συνάρτηση μπορεί να ενημερώνεται και να εξελίσσεται λύνοντας αριθμητικώς την σχέση (3.13) για να βρεθεί έτσι ένας νέος φορέας για το τρέχων βήμα της μεθόδου. Η συνάρτηση ψ αρχικοποιείται ως μια συνάρτηση αποστάσεως των στοιχείων από το όριο του τομέα Ω ύπαρξης υλικού, και με την χρήση μιας αριθμητικής τακτικής υπολογισμού των απαραίτητων παραγώγων επιτυγχάνεται η εξέλιξη της εντός κάθε επαναληπτικού βρόχου της μεθόδου. Επιπροσθέτως, το χρονικό βήμα για τον αριθμητικό υπολογισμό των διαφορικών επιλέγεται έτσι ώστε να ικανοποιεί την συνθήκη Courant-Friedrichs-Lewy (CFL), η οποία είναι απαραίτητη για την σύγκλιση όταν πρόκειται για αριθμητική επίλυση μερικώς διαφορικών εξισώσεων. Η συνθήκη αυτή περιορίζει το χρονικό βήμα να είναι

$$\Delta t \leq \frac{h}{\max|v|} \quad (3.17)$$

όπου h είναι η ελάχιστη απόσταση μεταξύ γειτονικών σημείων του πλέγματος διακριτοποίησης της επιφάνειας, και λαμβάνεται η μέγιστη τιμή που προκύπτει από τον έλεγχο όλων των σημείων (βλέπε Sethian 1999, Osher και Fedkiw 2002).

Όπως αναφέρθηκε και προηγουμένως, τα δυο βαθμωτά πεδία v και g επιλέγονται τυπικώς βάσει της σχηματικής και τοπολογικής ευαισθησίας του στόχου βελτιστοποίησης αντιστοίχως, με τον στόχο βελτιστοποίησης να είναι η ελαχιστοποίηση της ελαστικής παραμόρφωσης για τον φορέα. Προκειμένου να ικανοποιείται και ο περιορισμός υλικού που έχει τεθεί για την επιδιωκόμενη λύση, τα πεδία v και g επιλέγονται με εφαρμογή των ευαισθησιών σχήματος και τοπολογίας στην παρακάτω σχέση Lagrange

$$L = c(x) + \lambda^k (V(x) - V_{req}) + \frac{1}{2\lambda^k} [V(x) - V_{req}]^2 \quad (3.18)$$

όπου λ^k και Λ^k είναι παράμετροι οι οποίοι μεταβάλλονται σε κάθε επανάληψη k του αλγορίθμου βελτιστοποίησης. Εξελίσσονται με εφαρμογή των σχέσεων

$$\lambda^{k+1} = \lambda^k + \frac{1}{\Lambda^k}(V(x) - V_{req}), \quad \Lambda^{k+1} = a \Lambda^k \quad (3.19)$$

όπου $a \in (0,1)$ είναι μια σταθερή παράμετρος που εισάγεται εξ αρχής στον κώδικα και δεν μεταβάλλεται με τις επαναλήψεις, η οποία μάλιστα στα πλαίσια των αλγορίθμων που αναπτύχθηκαν για την Level Set είναι εντελώς σταθερή και ίση με 0.9 (όπως και στην πρότυπη για την εφαρμογή της μεθόδου δημοσίευση της V.J.Challis) χωρίς να αλλάζει τιμή ανά πρόβλημα.

Στην περίπτωση συνοριακών συνθηκών του φορέα δίχως την επίδραση τριβών για το συνεχώς μεταβαλλόμενο σύνορο του, η ευαισθησία σχήματος για την ελαστική παραμόρφωση $c(x)$ δίνεται ως (κατά Allaire 2004 [5]):

$$\frac{\delta c}{\delta \Omega}|_e = -u_e^T k_e u_e \quad (3.20)$$

Συγχρόνως, η ευαισθησία σχήματος για τον όγκο $V(x)$ δίνεται

$$\frac{\delta V}{\delta \Omega}|_e = 1 \quad (3.21)$$

Χρησιμοποιώντας τις παραπάνω προκύπτουσες ευαισθησίες σχήματος, η κάθετη στο σύνορο ταχύτητα v για κάθε στοιχείο e στην επανάληψη k του αλγορίθμου είναι

$$v|_e = -\frac{\delta L}{\delta \Omega}|_e = u_e^T k_e u_e - \lambda^k - \frac{1}{\Lambda^k}(V(x) - V_{req}) \quad (3.22)$$

Κατά Burger et al.(2004) ο όρος g δίνεται ως

$$g = -\text{sign}(\psi) \delta_T L \quad (3.23)$$

όπου $\delta_T L$ είναι η ευαισθησία τοπολογίας της σχέσης (3.13). Μιας και μονάχα περιοχές κενού συμπύσσονται εντός περιοχών υλικού και όχι το αντίθετο, για τους λόγους που παρουσιάστηκαν παραπάνω, η σχέση (3.23) παίρνει την μορφή

$$g = \begin{cases} \delta_T L & \text{εάν } \psi < 0 \\ 0 & \text{εάν } \psi > 0 \end{cases} \quad (3.24)$$

Η ευαισθησία τοπολογίας για την ελαστική παραμόρφωση $c(x)$ στις δυο διαστάσεις είναι (Allaire et al. 2004 [5])

$$\delta_T c|_e = \frac{\pi(\lambda + 2\mu)}{2\mu(\lambda + \mu)} (4\mu u_e^T k_e u_e + (\lambda - \mu) u_e^T (k_{Tr})_e u_e) \quad (3.25)$$

όπου $u_e^T (k_{Tr})_e u_e$ είναι η προσέγγιση με χρήση πεπερασμένων στοιχείων για το γινόμενο $\text{tr}(\sigma)\text{tr}(\epsilon)$, στο οποίο με σ συμβολίζεται ο τανυστής των τάσεων και με ϵ ο τανυστής (και με tr συμβολίζεται το ίχνος του μητρώου). Με λ και μ συμβολίζονται οι σταθερές Lamé για συμπαγές υλικό. Η ευαισθησία τοπολογίας για τον όγκο $V(x)$ δίνεται ως

$$\delta_T V|_e = -\pi \quad (3.26)$$

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Χρησιμοποιώντας τις ευαισθησίες αυτές στον ορισμό που δόθηκε για την σχέση Lagrange που διατυπώθηκε και λαμβάνοντας υπόψη την σχέση ... μπορεί να ευρεθεί ο όρος g .

Για εμβάθυνση στο θεωρητικό υπόβαθρο της μεθόδου μπορεί κανείς να ανατρέξει στο έργο των Allaire et al.(2004), όντας βασικό και ένα από τα πρώτα έργα περί της μεθόδου, καθώς και στις υπόλοιπες δημοσιεύσεις που αναφέρθηκαν στο παρών κεφάλαιο για τις διάφορες μεθοδολογίες που υιοθετήθηκαν, πέραν βεβαίως της δημοσίευσης της V.J.Challis 'A discrete level-set topology optimization code written in Matlab' στην οποία έχει βασιστεί άλλωστε η υλοποίηση της Level Set μεθόδου.

3.4.2 Εφαρμογή της μεθόδου Level Set σε μη δομημένα πλέγματα

Ένα από τα πιο δύσκολα εγχειρήματα που είχε να αντιμετωπίσει η παρούσα διπλωματική, αν όχι το πιο δύσκολο, ήταν η εφαρμογή της μεθόδου Level Set σε μη δομημένα πλέγματα πεπερασμένων στοιχείων. Το μη δομημένο της μορφής του πλέγματος δημιούργησε μεγάλα προβλήματα στις διαδικασίες επεξεργασίας της επιφάνειας, και συγκεκριμένα στην παραγωγή της ως προς x και y . Ο λόγος που η παραγωγή δημιουργεί το πρόβλημα είναι πως κατά την εκτέλεση αυτής υπάρχουν πράξεις που απαιτούν την τιμή της Level Set συνάρτησης στις θέσεις κατά x θετικά ή y αρνητικά ενός τυχαίου σημείου του πλέγματος. Όταν όμως το πλέγμα πεπερασμένων είναι μη δομημένο, αυτές οι σχέσεις μεταξύ στοιχείων όπως αναφέρθηκε και παραπάνω δεν είναι σαφώς ορισμένες.

Για την επίλυση του προβλήματος αυτού, μιας και η απαίτηση γνώσης της τιμής της Level Set σε σχετικές θέσεις κάθε σημείου δεν είναι ελαστική ως προς την εφαρμογή της, η ανάπτυξη ενός δομημένου πλέγματος σημείων για να εκτελεσθεί η παραγωγή αποτέλεσε μονόδρομο. Έτσι δημιουργήθηκε ως πρόσθετο πεδίο στον αλγόριθμο της πρότυπης δημοσίευσης ένας δομημένος κάνναβος σημείων όπου με την χρήση συντελεστών βαρύτητας θα μπορούσε να γίνει η μεταφορά της Level Set από το πλέγμα πεπερασμένων όπου αρχικώς θα είχε υπολογισθεί στον κάνναβο σημείων, και αντίστροφα. Το πρόβλημα που μένει να λυθεί λοιπόν είναι το ποιά σημεία του καννάβου της επιφάνειας επηρεάζουν ποια κεντροβαρικά σημεία του πλέγματος τριγωνικών στοιχείων και αντίστροφα, καθώς επίσης και ποιός είναι ο βαθμός αυτής της επιρροής, δηλαδή με τι συντελεστή βαρύτητας επηρεάζει το ένα σημείο του ενός συστήματος τα σημεία του άλλου.

Το πρώτο από τα ζητήματα αυτά λύνεται με την χρήση μιας ακτίνας επιρροής για την κρίση του ποιά σημεία θα πρέπει να επηρεάζουν ποιά. Για το εκάστοτε σημείο που ανήκει στο ένα από το δυο συστήματα εξετάζονται όλα τα σημεία του άλλου συστήματος ως προς την μεταξύ τους απόσταση, και εφόσον βρεθεί ότι η απόσταση αυτή είναι μικρότερη της ακτίνας επιρροής που έχει ορισθεί θεωρείται πως τα δυο σημεία αυτά αλληλοεπηρεάζονται. Για να γίνει βέβαια αυτός ο έλεγχος θα πρέπει πρώτα τα κεντροβαρικά σημεία των τριγωνικών στοιχείων να προβληθούν στο επίπεδο $x-y$, όπου και βρίσκεται το πλέγμα σημείων για την προσομοίωση της επιφάνειας. Αυτό υλοποιείται πολύ εύκολα με τον μηδενισμό της z συντεταγμένης των κεντροβαρικών σημείων των πεπερασμένων στοιχείων. Ο έλεγχος αυτός αποθηκεύεται με την μορφή λογικών τιμών σε μητρώα ώστε να μην χρειάζεται να επαναλαμβάνεται σε κάθε επανάληψη της μεθόδου, όπου με την έννοια λογική τιμή εννοούμε για παράδειγμα την τιμή 0 για καμία σχέση και έναν θετικό αριθμό εφόσον υπάρχει επιρροή.

Αφού επιλυθεί το πρώτο ζήτημα σειρά έχει ο υπολογισμός του βαθμού επιρροής του κάθε σημείου κατά την μεταφορά ενός πεδίου από το ένα σύστημα στο άλλο. Μιας και χρησιμοποιήθηκε η ίδια λογική με αυτήν που χρησιμοποιήθηκε στο φίλτρο της δημοσίευσης για την SIMP, δηλαδή ο έλεγχος με την χρήση μιας ακτίνας επιρροής, ακολουθείται παρόμοια λογική και για τον υπολογισμό των συντελεστών βάρους με τον

οποίο επηρεάζουν τα σημεία που θα έπρεπε να επηρεάζουν. Έτσι, για το κάθε σημείο που βρίσκεται εντός της ακτίνας επιρροής ενός άλλου αφαιρείται από την ακτίνα η μεταξύ τους απόσταση, και το αποτέλεσμα υψώνεται εις τον κύβο, που αποτελεί και το σημείο που ο αλγόριθμος αποκλίνει από αυτόν της δημοσίευσης. Αυτό εξυπηρετεί την στρατηγική που θέλουμε να εφαρμόσουμε κατά την μεταφορά πεδίων μεταξύ των δυο συστημάτων, κατά την οποία σημεία που βρίσκονται μεν εντός της ακτίνας επιρροής αλλά δεν είναι και ακριβώς δίπλα στον κατ' εξέταση σημείο έχουν πολύ μικρότερη επιρροή από αυτά που βρίσκονται όντως δίπλα κατά την προβολή του στο επίπεδο x-y.

Το επόμενο πρόβλημα που δημιουργείται λόγω της εφαρμογής της μεθόδου σε μη δομημένο πλέγμα πεπερασμένων στοιχείων είναι αυτό της επέκτασης του πλέγματος με εικονικά ψευδό-στοιχεία τα οποία αν και δεν συμμετέχουν προσφέροντας ακαμψία κατά την ανάλυση πεπερασμένων στοιχείων είναι πολύ σημαντικά κατά την εκ νέου δημιουργία της Level Set συναρτήσεως. Κατά τον αλγόριθμο της δημοσίευσής, όταν πρόκειται να παραγωγισθεί η επιφάνεια, για την σωστή παραγωγή των σημείων που αρχικώς είναι στα όρια του πλέγματος θα πρέπει να γίνει επέκταση αυτού, ώστε να υπάρχουν σημεία σε όλες τις σχετικές θέσεις για τα οριακά αυτά σημεία. Για να δημιουργηθούν τα σημεία αυτά, κατά την εκ νέου δημιουργία της Level Set μέσω της Reinit συναρτήσεως γίνεται επέκταση του πλέγματος σημείων με χρήση της εντολής `padarray` της `matlab`. Η συνάρτηση αυτή, που αν και υπάρχει ούτως ή άλλως στην `matlab` παρουσιάζεται εκδοχή της και στην πρότυπη για το παρόν δημοσίευση, προσθέτει σειρές και στήλες περιμετρικώς ενός διδιάστατου μητρώου (ίσως και περισσότερων διαστάσεων αλλά για τα προβλήματα που προσπαθούμε να λύσουμε αρκούν οι δυο) για την επέκταση αυτού. Κάτι τέτοιο όμως δεν μπορεί να γίνει στα πλέγματα που προκύπτουν από το GMSH, γιατί όντας μη δομημένα δε μπορούν να στηθούν σε διδιάστατα μητρώα. Αντ' αυτού αποθηκεύονται σε μητρώα λίστες με την ίδια σειρά που εξάγονται από το GMSH, οπότε δεν έχει νόημα αυτής της λίστας. Για να είμαστε ακόμα πιο ακριβείς, και η ίδια η λίστα δεν λέει τίποτα για τις σχετικές θέσεις των πεπερασμένων στο χώρο, μιας και δυο στοιχεία που μπορεί να είναι το ένα μετά το άλλο στην λίστα αυτή μπορεί να βρίσκονται αντιδιαμετρικά στο πλέγμα.

Ο τρόπος που το πρόβλημα αυτό αντιμετωπίζεται στην παρούσα διπλωματική ίσως να μην είναι ο βέλτιστος από πλευράς ταχύτητας του κώδικα, αλλά είναι αποτελεσματικός. Καταρχάς πρέπει να προσδιοριστεί το ποιά στοιχεία βρίσκονται στα όρια του πλέγματος, κάτι το οποίο γίνεται με τον έλεγχο για όλα τα στοιχεία του αν υπάρχουν άλλα στοιχεία στις σχετικές τους θέσεις, δηλαδή αν έχουν ένα στοιχείο κατά x θετικά τους, κατά x αρνητικά τους και ούτω καθεξής. Για τα στοιχεία που βρίσκεται πως κάποια σχετική τους θέση είναι κενή, δημιουργείται ένα νέο ψευδό-στοιχείο για την χρήση του μετέπειτα στον αλγόριθμο, σε όσες θέσεις απαιτείται να υπάρχει η περιμετρική ζώνη εικονικών στοιχείων για την εκτέλεση κάποιας διαδικασίας. Όλα αυτά τα ψευδό-στοιχεία ουσιαστικά προσομοιώνονται μονάχα από το κεντροβαρικό τους σημείο, χωρίς να έχουν κόμβους με τρόπο ανάλογο των πεπερασμένων του πλέγματος.

Ο τρόπος με τον οποίο δημιουργείται ένα νέο εικονικό στοιχείο θα γίνει καλύτερα κατανοητός με την μελέτη και επεξήγηση του κώδικα, δουλειά που εκτελείται στην αμέσως επόμενη παράγραφο του παρόντος κεφαλαίου.

3.4.3 Επεξήγηση κώδικα Matlab για Level Set

Όπως και στις δυο προηγούμενες μεθόδους, έτσι και εδώ αναπτύχθηκε κώδικας σε Matlab ευθέως αντίστοιχος αυτού που γράφθηκε για την δομή `MSolve` μέσω της οποίας εκτελέστηκαν οι αναλύσεις πεπερασμένων στοιχείων. Η δημοσίευση στην οποία βασίστηκε η ανάπτυξή του, όπως ήδη έχει αναφερθεί, είναι η 'A discrete level-set topology optimization code written in Matlab' της V.J.Challis. Αν και ο σκοπός κατά την ανάπτυξη του ήταν να είναι κατά το δυνατόν όμοιος με αυτόν της δημοσίευσης που στηρίχθηκε η όλη

υλοποίηση της μεθόδου, αυτό στην συγκεκριμένη μέθοδο κατέστη αδύνατο λόγω των εμποδίων που δημιουργήθηκαν από το μη δομημένο της δοθείσας στον αλγόριθμο διακριτοποίησης. Εμπόδια που επέβαλλαν την ανάπτυξη διαφορετικών μεταβλητών και μεθοδολογίας σε ορισμένα σημεία προκειμένου τελικώς το πρόβλημα που προέκυπτε να είναι επιλύσιμο, έστω και αν σε συγκεκριμένες περιπτώσεις όπως θα δούμε τα αποτελέσματα είναι αμφιβόλου ποιότητας.

Για την εκτέλεση της βελτιστοποίησης τοπολογίας με την μέθοδο Level Set καλείται η συνάρτηση `TopologyLS(nodes elemConn, elemSize, volfrac, stepLength, numReinit, topWeight, E, ν, thickness, fixedDOFs, F, ForcedNodes, gridx, gridy, point1, point2)`, η οποία παρατίθεται στο παράρτημα Δ, μαζί με όλες τις συναρτήσεις που καλεί αυτή με την σειρά της.

Οι παράμετροι που απαιτούνται κατά την κλήση της είναι οι

- `nodes`: οι κόμβοι στο καθολικό σύστημα συντεταγμένων
- `elemConn`: οι συνδέσεις των κόμβων για το κάθε στοιχείο
- `elemSize`: το μέγεθος των στοιχείων, το οποίο έχει οριστεί από τον χρήστη κατά την διακριτοποίηση στο GMSH
- `volfrac`: ο λόγος του όγκου που θα πρέπει να έχει ο βέλτιστος φορέας ως προς τον συνολικό διατιθέμενο όγκο
- `stepLength`: μεταβλητή που προσδιορίζει το χρονικό εύρος για την εξέλιξη του σχεδιασμού του φορέα, εκφρασμένη σε συνάρτηση με την τιμή CFL
- `topWeight`: ο συντελεστής βάρους ω της σχέσεως (3.15)
- `numReinit`: ο αριθμός των επαναλήψεων ανά τις οποίες θα εφαρμοστεί η εκ νέου δημιουργία της Level Set συναρτήσεως στην επιφάνεια
- `E`: το μέτρο ελαστικότητας κατά Young του υλικού
- `ν`: ο λόγος Poisson του υλικού
- `thickness`: το πάχος των τριγωνικών στοιχείων
- `fixedDOFs`: το διάνυσμα των δεσμευμένων βαθμών ελευθερίας του φορέα
- `F`: το διάνυσμα των επικομβίων δράσεων του φορέα
- `ForcedNodes`:
- `gridx`: Ο αριθμός των κατά x σημείων που συνθέτουν το πλέγμα για την προσομοίωση της επιφάνειας
- `gridy`: Ο αριθμός των κατά y σημείων που συνθέτουν το πλέγμα για την προσομοίωση της επιφάνειας
- `point1`: το κάτω αριστερά σημείο σε κάτοψη του ορθογωνίου που θα αποτελέσει την επιφάνεια της μεθόδου
- `point2`: το πάνω δεξιά σημείο σε κάτοψη του ορθογωνίου που θα αποτελέσει την επιφάνεια της μεθόδου

Γραμμές 2-12

Αρχικοποίηση μερικών από τις απαραίτητες για την εκτέλεση της μεθόδου παραμέτρους, όπως τον συνολικό αριθμό στοιχείων `nel`, το διάνυσμα `stuc` όπου για το κάθε στοιχείο αποθηκεύεται το αν υπάρχει ή όχι υλικό, το διάνυσμα `shapeSens` όπου αποθηκεύονται οι ευαισθησίες σχήματος των στοιχείων του φορέα σε κάθε επανάληψη, το διάνυσμα `topSens` όπου αποθηκεύονται οι ευαισθησίες τοπολογίας των στοιχείων, οι μεταβλητές λ και μ που αποτελούν τις σταθερές Lamé του υλικού, τις μεταβλητές I_a , I_b και α της σχέσεως Langrange (3.18), και τέλος το διάνυσμα `objective` όπου και θα αποθηκεύονται οι συνολικές ελαστικές παραμορφώσεις από κάθε επανάληψη της μεθόδου. Το ζήτημα που παρουσιάζεται και εδώ είναι πως δε γνωρίζουμε εκ των προτέρων πόσες επαναλήψεις απαιτούνται προτού η μέθοδος συγκλίνει, οπότε το διάνυσμα `objective` αρχικοποιείται με έναν ικανοποιητικό αριθμό στοιχείων.

Γραμμές 13-18

Στις γραμμές αυτές εκτελούνται δυο πολύ σημαντικές εργασίες για την συνέχεια του αλγορίθμου. Η πρώτη, που εκτελείται μεταξύ των γραμμών 14 έως και 17 αφορά τον υπολογισμό και την αποθήκευση στο μητρώο `centerPoints` των κεντροβαρικών σημείων για το κάθε τριγωνικό πεπερασμένο στοιχείο. Η δεύτερη, που εκτελείται στην γραμμή 28 με κλήση της συναρτήσεως `Connect` αφορά τον προσδιορισμό των σχετικών θέσεων μεταξύ των στοιχείων, με τον τρόπο που αναφέρθηκε στο κεφάλαιο 3.4.2(τον δεύτερο, μιας και ο πρώτος απέτυχε) και που θα αναλυθεί πρακτικώς παρακάτω όπου και θα παρουσιαστεί η εν λόγω συνάρτηση. Αυτό που ο αναγνώστης πρέπει να έχει στο μυαλό του προς το παρών είναι πως το προκύπτον μητρώο `connections` από την συνάρτηση `Connect` nel στον αριθμό σειρές και τέσσερις στήλες. Κάθε σειρά αναφέρεται και σε ένα στοιχείο, ενώ σε κάθε στήλη αποθηκεύεται ο αύξων αριθμός του στοιχείου έχει την σχετική θέση που έχει οριστεί να αντιστοιχεί στην συγκεκριμένη στήλη. Για την ακρίβεια, στην πρώτη στήλη αποθηκεύεται το στοιχείο j που βρίσκεται κατά x θετικά του τρέχοντος i στοιχείου, στην δεύτερη στήλη αποθηκεύεται το στοιχείο j που βρίσκεται κατά x αρνητικά του i , στην τρίτη στήλη αποθηκεύεται το στοιχείο j που βρίσκεται κατά y θετικά του i και τέλος στην τέταρτη στήλη αποθηκεύεται το στοιχείο j που βρίσκεται κατά y αρνητικά του i . Στην περίπτωση που κανένα j στοιχείο δεν ικανοποιεί τα κριτήρια για κάποια από τις σχετικές θέσεις του i στοιχείου, στην αντίστοιχη στήλη θα βρίσκεται η τιμή μηδέν που δεν αντιστοιχεί σε κανέναν αύξων αριθμού πεπερασμένου(ή η τιμή -1 για τον κώδικα σε C# μιας και εκεί το 0 αποτελεί έγκυρο δείκτη σε μητρώα).

Γραμμές 19-21

Εδώ υλοποιούνται άλλα δυο πολύ σημαντικά για την εκτέλεση του αλγορίθμου σημεία. Καταρχάς με την χρήση της συνάρτησης `GridCreator` δημιουργείται το μητρώο `grid` που περιέχει τις συντεταγμένες των στοιχείων του πλέγματος σημείων για την προσομοίωση της επιφάνειας της μεθόδου. Στην επόμενη γραμμή καλείται η συνάρτηση `CreateFakeElements`, μέσω της οποίας δημιουργούνται τα ψευδό-στοιχεία για τις θέσεις που δεν υπάρχουν στοιχεία που να ικανοποιούν τα κριτήρια της συνάρτησης `Connect`. Για όσες λοιπόν θέσεις του μητρώου `connections` περιέχουν την τιμή 0, και άρα δεν έχει ευρεθεί το κατάλληλο στοιχείο για την συγκεκριμένη σχετική θέση, δημιουργείται και ένα νέο σημείο στην θέση περίπου που θα έπρεπε να είναι το κέντρο βάρους του στοιχείου που λείπει από την σχετική αυτή θέση. Ο αύξων αριθμός του αποθηκεύεται στην ανάλογη στήλη του μητρώου `connectionsFake`, το οποίο έχει την ίδια δομή με το μητρώο `connections` που είδαμε, αλλά έχει την διαφορά πως έχει την τιμή 0 στις θέσεις που υπάρχει πραγματικό στοιχείο για να καλύψει την σχετική θέση και τον θετικό αύξων αριθμό στις θέσεις που δημιουργήθηκε ψευδό-στοιχείο.

Γραμμές 22-31

Πρόκειται για ακόμα μια σειρά πεδίων που χρειάζεται να δημιουργηθούν μονάχα μια φορά στην αρχή της μεθόδου αλλά είναι απαραίτητα σε κάθε της επανάληψη. Δημιουργούνται τα μητρώα που περιέχουν τις συνδέσεις των πεπερασμένων στοιχείων, καθώς και των εικονικών ψευδό-στοιχείων, με τα σημεία του πλέγματος που αντιστοιχούν στην επιφάνεια. Οι συνδέσεις καθορίζονται βάσει ακτινών επιρροής για το κάθε κέντρο βάρους στοιχείου αλλά και το κάθε σημείο, χωρίς όμως την απαίτηση η ακτίνα των συνδέσεων από το πλέγμα πεπερασμένων στο πλέγμα σημείων να είναι η ίδια με την ακτίνα της αντίστροφης διαδικασίας. Συγχρόνως υπολογίζονται οι συντελεστές βαρύτητας με τους οποίους θα πολλαπλασιάζονται τα μεγέθη που μεταφέρονται από το πλέγμα πεπερασμένων στο πλέγμα σημείων της επιφάνειας ή αντιστρόφως. Αναλυτικότερα ο τρόπος

καθορισμού των συνδέσεων ή ο υπολογισμός των συντελεστών βαρύτητας παρουσιάζονται παρακάτω κατά την παρουσίαση των υπεύθυνων για την δημιουργία των εν λόγω μητρώων συναρτήσεων.

Γραμμές 32-36

Το σημαντικότερο μητρώο όλων, αυτό χωρίς το οποίο η όλη διαδικασία δεν είχε κανέναν απολύτως νόημα, είναι το μητρώο όπου και αποθηκεύονται τα επιμέρους μητρώα δυσκαμψίας των στοιχείων στο καθολικό σύστημα. Πρόκειται για ένα μητρώο διαστάσεων $30 \times 30 \times n_{el}$, η τρίτη διάσταση του οποίου τρέχει για όλα τα πεπερασμένα στοιχεία του πλέγματος. Τα μητρώα δυσκαμψίας υπολογίζονται με την χρήση της συναρτήσεως `ElementMohrTri6` που παρουσιάστηκε στο κεφάλαιο 2.3

Γραμμές 37-39

Εκκίνηση των επαναλήψεων της μεθόδου για την βελτιστοποίηση της τοπολογίας με την μέθοδο `Level Set`. Προτού όμως ο αλγόριθμος εισέλθει στον βρόγχο των επαναλήψεων, καλείται η συνάρτηση `Reinit` για να δημιουργηθεί το μητρώο `Isf`, που περιέχει τις τιμές της `Level Set Function` στο πλέγμα των σημείων που προσομοιώνουν την επιφάνεια. Την συνάρτηση αυτή θα την συναντήσουμε και παρακάτω καθώς ανά μερικές επαναλήψεις της μεθόδου η `Isf` θα πρέπει να υπολογίζεται εκ νέου, ενώ την μεθοδολογία με την οποία γίνεται η επανεκκίνησή της θα παρουσιαστεί στην πορεία κατά την ανάλυση της συναρτήσεως `Reinit`. Στην συνέχεια, στην γραμμή 39 ξεκινάνε οι επαναλήψεις της μεθόδου, ο συνολικός αριθμός των οποίων καθορίζεται εκ των προτέρων όπως στην περίπτωση του διανύσματος `objective`, αν και πολύ πιθανόν να έχουμε σύγκλιση του κώδικα προτού ολοκληρωθεί το σύνολο των επαναλήψεων, όπου τον έλεγχο της συγκλίσεως θα τον δούμε σε επόμενες γραμμές του κώδικα. Επίσης δημιουργείται το διάνυσμα `struc`, πρακτικής σημασίας, που αντικαθιστά τις μηδενικές τιμές του διανύσματος `struc` με μια ελάχιστη πυκνότητα υλικού (0.001 στην προκειμένη), για την αποφυγή αριθμητικών προβλημάτων του κώδικα.

Γραμμή 41

Επίλυση της εξισώσεως ισορροπίας του φορέα για την εύρεση των επικόμβιων μετατοπίσεων του με χρήση της συναρτήσεως `FEA(Finite Element Analysis)`, η οποία στο μεγαλύτερο της μέρος είναι ολόγρια με την αντίστοιχη που χρησιμοποιούνταν στις μεθόδους `BESO` και `SIMP`. Όπως και οι υπόλοιπες συναρτήσεις που χρησιμοποιούνται από την `Level Set` έτσι και αυτή παρατίθεται στο Παράρτημα Δ παρά την μεγάλη της ομοιότητα με τις αντίστοιχες συναρτήσεις που παρατίθενται στα παραρτήματα Β και Γ, χωρίς όμως να γίνεται κάποια περαιτέρω αναφορά στον τρόπο λειτουργίας της.

Γραμμές 42-47

Υπολογισμός των ευαισθησιών σχήματος και τοπολογίας των πεπερασμένων στοιχείων, όπως καθορίζεται από τις σχέσεις (3.25) και (3.26) αντίστοιχα. Η επανάληψη των γραμμών αυτών τρέχει για όλα τα `nel` πεπερασμένα στοιχεία του πλέγματος, βρίσκοντας αρχικώς τους βαθμούς ελευθερίας του καθολικού συστήματος που αντιστοιχούν στο στοιχείο. Με χρήση του διανύσματος `eIDOFs` που δημιουργήθηκε εξάγουμε από το διάνυσμα `U` που περιέχει τις επικόμβιες μετατοπίσεις όλων των στοιχείων αυτές που αντιστοιχούν στους κόμβους του τρέχοντος στοιχείου, τις οποίες και αποθηκεύουμε στο διάνυσμα `Ue`. Έχοντας λοιπόν τις επικόμβιες μετατοπίσεις του στοιχείου υπολογίζονται η ευαισθησία σχήματος και τοπολογίας αυτού, στις γραμμές 45 και 46.

Γραμμές 48-49

Εδώ γίνεται ο υπολογισμός του όγκου της κατασκευής `volCurr` και της ελαστικής παραμόρφωσης `objective(iterNum)` που αντιστοιχούν στην τρέχουσα επανάληψη

της μεθόδου. Σημαντικό στοιχείο που πρέπει να κρατήσει κανείς από τους υπολογισμούς αυτών των γραμμών είναι πως η ελαστική παραμόρφωση υπολογίζεται ως το αρνητικό άθροισμα του διανύσματος των ευαισθησιών σχήματος, που είναι και λογικό εάν εξετάσουμε προσεκτικότερα τον τρόπο υπολογισμού των στοιχείων του συγκεκριμένου διανύσματος.

Γραμμές 50-52

Έλεγχος σύγκλισης της μεθόδου προκειμένου να σταματήσει προτού ολοκληρωθούν όλες οι επαναλήψεις αυτής. Ο έλεγχος απαιτεί να έχουν προηγηθεί τουλάχιστον 5 επαναλήψεις της μεθόδου, όχι μονάχα για λόγους αξιοπιστίας των αποτελεσμάτων μέσω επαρκούς επεξεργασίας της μεθόδου, αλλά και επειδή επικαλούνται στοιχεία έως και πριν από πέντε επαναλήψεις σε επόμενο μέρος του ελέγχου σύγκλισης. Επίσης θα πρέπει ο όγκος που χρησιμοποιείται από τον φορέα που έχει επιτευχθεί να είναι ποσοστιαίως κοντά σε αυτόν που απαιτείται κατά την εκκίνηση του αλγορίθμου, όπου το κοντά ορίζεται και πάλι από τον χρήστη του κώδικα χωρίς κάποιον γενικό κανόνα αλλά αναλόγως με το ποιό ακρίβεια τον ικανοποιεί. Τέλος, θα πρέπει η ελαστική παραμόρφωση των φορέων των τελευταίων πέντε επαναλήψεων να είναι ποσοστιαίως παραπλήσια με αυτήν που επιτεύχθηκε στην τελευταία αυτή επανάληψη για την οποία γίνεται ο έλεγχος σύγκλισης. Στην περίπτωση που ικανοποιούνται όλα τα παραπάνω κριτήρια η γραμμή 51 οδηγεί εκτός του βασικού βρόγχου επαναλήψεων της μεθόδου και ο αλγόριθμος φτάνει στο τέλος του.

Γραμμές 53-58

Αφού πρώτα οι παράμετροι l_a και L_a της συναρτήσεως Langrange ανανεωθούν για το τρέχον βήμα της επανάληψης στις γραμμές 54 και 55, γίνεται χρήση αυτών ώστε να ληφθεί υπόψη στις ευαισθησίες των στοιχείων η απόκλιση του φορέα του τρέχοντος βήματος ως προς τον όγκο του σε σχέση με τον όγκο φορέα που πρέπει να επιτευχθεί από τον αλγόριθμο. Η σημασία της επιρροής αυτής στις ευαισθησίες είναι τεράστια για την ορθή λειτουργία του αλγορίθμου, μιας και αν αυτή δεν υπήρχε σε κανένα άλλο σημείο δεν θα υπήρχε κάτι που να πιέζει τον φορέα στην εύρεση λύσεως με όγκο πιο κοντά σε αυτόν που απαιτείται από τον χρήστη. Αν όμως κανείς παρατηρήσει με μεγαλύτερη προσοχή τον κώδικα θα δει πως η λειτουργία αυτών των γραμμών είναι ακόμα σημαντικότερη, και μπορεί μάλιστα να είναι το μοναδικό σημείο που θα απελευθερώσει τελικά την μέθοδο από μια επαναλαμβανόμενη λούπα που θα δίνει συνεχώς το ίδιο αποτέλεσμα. Η συνεχής επανάληψη χωρίς ουσία θα οφείλεται στο μη ανηγμένο των ευαισθησιών ως προς τα ελαστικά χαρακτηριστικά του φορέα, με αποτέλεσμα στην περίπτωση που τα ελαστικά χαρακτηριστικά κάνουν τις ευαισθησίες πολύ μεγάλες σε σύγκριση με την Level Set συνάρτηση(η οποία θα πρέπει να θυμηθούμε πως από την κλήση της Reinit ορίζεται ανάλογα με το μέγεθος συγκεκριμένων αποστάσεων, που κατά πάσα πιθανότητα δεν είναι και πολύ μεγάλες) να είναι πολύ δύσκολο να περάσει αυτή πάνω από την τιμή μηδέν και να δημιουργηθούν περιοχές κενού. Εάν κάτι τέτοιο συμβεί κάθε επανάληψη θα δίνει τον ίδιο τον αρχικό φορέα, δηλαδή παντού υλικό, και δε θα υπάρχει κανένας τρόπος να αλλάξει η Level Set προς την κατεύθυνση αφαίρεσης υλικού.

Γραμμή 59

Κλήση της συναρτήσεως UpdateStep για την εξέλιξη της συναρτήσεως επιφανείας Level Set.

Γραμμή 60-62

Κλήση της συναρτήσεως Reinit, που συναντήσαμε και στην αρχή του κώδικα, προκειμένου να δημιουργηθεί εκ νέου η συνάρτηση επιφανείας Level Set. Η κλήση αυτή πραγματοποιείται κάθε έναν συγκεκριμένο αριθμό επαναλήψεων, ο οποίος

δίνεται εξ αρχής από τον χρήστη μέσω της μεταβλητής numReinit. Έτσι, αν η επανάληψη είναι ακέραιο πολλαπλάσιο του numReinit πραγματοποιείται η κλήση της συναρτήσεως, ενώ σε κάθε άλλη περίπτωση ο κώδικας συνεχίζει κανονικά, σε μια ακόμα επανάληψη της μεθόδου.

Αφού παρουσιάστηκε η βασική συνάρτηση που καλείται για την εφαρμογή της μεθόδου, θα προχωρήσουμε και στην παρουσίαση των υπολοίπων συναρτήσεων που καλούνται από την TopologyLS για την υλοποίηση των επιμέρους πεδίων και μητρώων που απαιτούνται για την εκτέλεση του αλγορίθμου. Παρουσιάζονται ανάλογα με την σειρά που παρατίθενται στο Παράρτημα Δ, η οποία είναι η ίδια με την σειρά που εμφανίζονται στη λογική σειρά του αλγορίθμου.

Η επόμενη συνάρτηση που θα παρουσιαστεί είναι η Connect(centerPoints, elemSize) η οποία είναι υπεύθυνη για τον ορισμό των σχετικών θέσεων μεταξύ των στοιχείων. Η συνάρτηση αυτή καλεί με την σειρά της άλλες τέσσερις συναρτήσεις για την εύρεση του κατάλληλου στοιχείου για κάθε μια από τις σχετικές θέσεις, την κατά x θετικά, την κατά x αρνητικά, την κατά y θετικά και την κατά y αρνητικά. Επειδή οι επιμέρους αυτές συναρτήσεις είναι αρκετά απλές στην λογική που ακολουθούν, και η εκτενής παρουσίαση και των τεσσάρων θα κούραζε παραπάνω απ' όσο απαιτείται για την κατανόηση τους, θα παρουσιαστεί αμέσως μετά μονάχα η πρώτη, δηλαδή η υπεύθυνη για τις κατά x θετικά συνδέσεις.

Οι μεταβλητές που απαιτούνται κατά την κλήση της είναι οι:

centerPoints: τα σημεία των κέντρων βάρους των τριγωνικών στοιχείων

elemSize: το μέγεθος των στοιχείων, το οποίο έχει οριστεί από τον χρήστη κατά την διακριτοποίηση στο GMSH

Η συνάρτηση αποτελείται από όλο όλο 5 γραμμές (αν εξαιρέσουμε φυσικά το όνομα και το τέλος της), κατά τις οποίες αρχικά δημιουργείται το μητρώο connect, και στην συνέχεια κάθε μία από τις επόμενες γραμμές είναι υπεύθυνη για τον καθορισμό των στοιχείων που ανήκουν στην σχετική θέση του εκάστοτε τριγωνικού πεπερασμένου, και άρα αποθηκεύουν αντίστοιχη στήλη του μητρώου connections. Η συνάρτηση ConnectionsXplus(centerPoints, elemSize) αποθηκεύει στην πρώτη στήλη του εν λόγω μητρώου βρίσκοντας για κάθε στοιχείο το κατά x θετικά κοντινότερό του.

Για την κλήση της συναρτήσεως ConnectionsXplus απαιτούνται οι ίδιες μεταβλητές που απαιτούνται κατά την κλήση της συναρτήσεως Connect, οι οποίες όπως βλέπουμε περνάνε αυτούσιες μέσω της Connect χωρίς να υποστούν οιαδήποτε αλλαγή. Η λογική της είναι η εξής:

Γραμμές 2-5

Αρχικοποίηση του διανύσματος όπου θα αποθηκεύονται οι κατά x σχετικές θέσεις, και έναρξη της επανάληψης η οποία τρέχει για όλα τα i πεπερασμένα στοιχεία του πλέγματος. Επίσης δημιουργούνται οι μεταβλητές maxDist και minDist που θα χρειαστούν παρακάτω κατά την εκτέλεση των ελέγχων για το αν ένα στοιχείο ανήκει όντως στην κατά x θετικά σχετική θέση ή όχι.

Γραμμές 6-17

Εισαγωγή στον δεύτερο βρόγχο επαναλήψεων ο οποίος επίσης τρέχει σε όλα τα πεπερασμένα στοιχεία του πλέγματος. Στην συνέχεια για κάθε j στοιχείο του εσωτερικού βρόχου ελέγχεται αν η κατά x συνιστώσα του είναι μεγαλύτερη της x συνιστώσας του i στοιχείου της προηγούμενης επανάληψης. Έπειτα ελέγχεται αν η απόσταση των κέντρων βάρους τους είναι η μικρότερη απόσταση που έχει βρεθεί

για j στοιχείο έως τώρα (σύγκριση με το `minDist`), καθώς και το αν η απόσταση αυτή είναι μικρότερη από μια τυπική απόσταση για να μην βρεθεί κάποιο στοιχείο που να βρίσκεται πολύ μακριά κατά x θετικά, το οποίο όμως να ικανοποιεί όλους τους άλλους ελέγχους (σύγκριση με το `maxDist`, το οποίο αυθαίρετα έχει επιλεγεί ίσο με 3 επί το μέγεθος των στοιχείων). Τέλος, ελέγχεται αν η διαφορά των κατά x συνιστωσών των στοιχείων i και j που κρίνονται είναι μεγαλύτερη της διαφοράς των κατά y συνιστωσών τους, ένας έλεγχος που γίνεται ώστε να είναι κάπως ξεκάθαρη η σχέση και να μην την ικανοποιήσει για παράδειγμα κάποιο στοιχείο που βρίσκεται λίγο μόνο κατά x θετικά αλλά περισσότερο ανήκει στην θέση του κατά y αρνητικά στοιχείου. Το j στοιχείο που ικανοποιεί όλους τους παραπάνω ελέγχους αποθηκεύεται τελικώς στην σειρά που αντιστοιχεί στο i στοιχείο του διανύσματος `connectionsXplus`, το οποίο με την σειρά του αποθηκεύεται στην πρώτη στήλη του μητρώου `connections`. Στην περίπτωση που κανένα στοιχείο j δεν ικανοποιεί όλους τους ελέγχους τότε όπως είπαμε αποθηκεύεται η τιμή 0 για την θέση αυτή.

Ανάλογες είναι και οι διαδικασίες που εφαρμόζονται από τις συναρτήσεις `connectionsXminus`, `connectionsYplus` και `connectionsYminus`, μετά την κλήση των οποίων και την συμπερίληψη των διανυσμάτων τους στο μητρώο `connections` αυτό είναι έτοιμο για περαιτέρω χρήση.

Η επόμενη συνάρτηση που θα μελετηθεί είναι η `GridCreator(gridx,gridy,point1,point2)`, η οποία για την ακρίβεια είναι αρκετά απλή ως προς τις εσωτερικές τις διαδικασίες. Για την κλήση απαιτούνται οι μεταβλητές:

`gridx`: Ο αριθμός των κατά x σημείων που συνθέτουν το πλέγμα για την προσομοίωση της επιφάνειας

`gridy`: Ο αριθμός των κατά y σημείων που συνθέτουν το πλέγμα για την προσομοίωση της επιφάνειας

`point1`: το κάτω αριστερά σημείο σε κάτοψη του ορθογωνίου που θα αποτελέσει την επιφάνεια της μεθόδου

`point2`: το πάνω δεξιά σημείο σε κάτοψη του ορθογωνίου που θα αποτελέσει την επιφάνεια της μεθόδου

Γραμμές 2-4

Δημιουργία των μεταβλητών `sizeX` και `sizeY` που αποτελούν το βήμα κατά την δημιουργία των στοιχείων της επιφάνειας κατά x και y αντίστοιχα. Επίσης δημιουργείται το μητρώο `grid` αποθήκευσης των συντεταγμένων των σημείων της επιφάνειας, η οποία αποτελείται από κλάσμα `gridx` επί `gridy` σημείων.

Γραμμές 5-10

Δυο επαναλήψεις οι οποίες τρέχουν για όλα τα σημεία κατά x και κατά y της επιφάνειας για τον υπολογισμό των συντεταγμένων των σημείων, ανάλογα με την θέση του στο μητρώο `grid`.

Στη συνέχεια του κώδικα, στην γραμμή 21 αυτού, περνάμε στο δεύτερο πολύ σημαντικό στοιχείο διαφοροποίησης του από αυτόν της δημοσιεύσεως, το σημείο της δημιουργίας των ψευδό-στοιχείων στις θέσεις όπου δεν υπάρχει κάποιο πεπερασμένο που να καλύπτει τα κριτήρια συνδέσεως σε κάποια σχετική οποιοδήποτε στοιχείου. Τα ψευδό-στοιχεία αυτά δημιουργούνται με την κλήση της συνάρτησης

```
CreateFakeElements(centerPoints,connections,elemSize)
```

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Οι μεταβλητές που απαιτούνται κατά την κλήση της είναι οι
 centerPoints: τα σημεία των κέντρων βάρους των τριγωνικών στοιχείων
 connections: το μητρώο που περιέχει τις σχετικές θέσεις των στοιχείων
 elemSize: το μέγεθος των στοιχείων, το οποίο έχει οριστεί από τον χρήστη κατά την
 διακριτοποίηση στο GMSH

Γραμμές 2-4

Δημιουργείται το μητρώο connectionsFake, το οποίο περιέχει για κάθε πεπερασμένο το ψευδό στοιχείο που υπάρχει στην σχετική θέση που αντιστοιχεί στην κάθε του στήλη, όπου οι στήλες έχουν την ίδια λογική που έχουν και στο μητρώο connections που ήδη είδαμε. Στην περίπτωση που δεν υπάρχει ψευδό-στοιχείο σε κάποια σχετική θέση τότε αποθηκεύεται η τιμή 0 στην αντίστοιχη στήλη. Επίσης δημιουργείται το μητρώο tempCenterPointsFake, στο οποίο θα αποθηκευτούν προσωρινά οι συντεταγμένες των ψευδό-στοιχείων που θα δημιουργηθούν, και το οποίο αρχικοποιείται στις μέγιστες διαστάσεις που θα μπορούσε να έχει μιας και δεν είναι γνωστό εκ των προτέρων πόσα ψευδό-στοιχεία θα χρειαστεί να δημιουργηθούν. Επίσης δημιουργείται η μεταβλητή sum για τον υπολογισμό των σημείων που θα δημιουργηθούν και εκκινείται στην τιμή 0.

Γραμμές 5-26

Ξεκινάει ο επαναληπτικός βρόγχος που θα περάσει από όλα τα πεπερασμένα του πλέγματος στην γραμμή 5. Στην συνέχεια, στις γραμμές 6 έως και 10, ελέγχεται το αν υπάρχει πραγματικό στοιχείο στην σχετική θέση κατά x θετικά του τρέχοντος στοιχείου που εξετάζεται, ή αν είναι αποθηκευμένη η τιμή 0 το οποίο σημαίνει πως κανένα στοιχείο δεν ικανοποιούσε τα κριτήρια. Στην περίπτωση λοιπόν που είναι αποθηκευμένη η τιμή 0, πρέπει να δημιουργηθεί ένα ψευδό-στοιχείο. Αυξάνεται η μεταβλητή sum κατά ένα λόγω του νέου σημείου που θα υλοποιηθεί, και έπειτα δημιουργείται ένα νέο σημείο με συντεταγμένες τις ίδιες με αυτές του κέντρου βάρους του τρέχοντος στοιχείου προσαυξάνοντας όμως την ως προς x συνιστώσα του κατά elemSize, αφού η σχετική θέση αφορά την πρώτη στήλη και την κατά x θετικά σύνδεση. Το νέο αυτό σημείο που δημιουργήθηκε αναφέρεται στο κέντρο βάρους του ψευδό-στοιχείου που εικονικά πλέον υπάρχει στην συγκεκριμένη θέση. Ο αύξων αριθμός του αποθηκεύεται στην αντίστοιχη θέση του μητρώου connectionsFake, η οποία εν προκειμένω είναι η πρώτη, και οι συντεταγμένες του στο προσωρινό μητρώο tempCenterPointsFake. Η ίδια εργασία εκτελείται και για τις υπόλοιπες σχετικές θέσεις μεταξύ των γραμμών 11-25.

Γραμμές 27-30

Δημιουργείται το μητρώο centerPointsFake μιας και πλέον είναι γνωστό το πόσα ψευδό-στοιχεία δημιουργήθηκαν, και περνάνε σε αυτό οι συντεταγμένες που είχαν προσωρινώς αποθηκευτεί στο μητρώο tempCenterPointsFake.

Αφού πλέον δημιουργήθηκαν και τα εικονικά στοιχεία του πλέγματος σειρά έχει ο καθορισμός των συνδέσεων μεταξύ του καννάβου σημείων για την προσομοίωση της επιφάνειας της Level Set και των κεντροβαρικών σημείων των πεπερασμένων στοιχείων, είτε αυτά είναι πραγματικά είτε εικονικά. Για τον καθορισμό των συνδέσεων καλούνται οι αντίστοιχες συναρτήσεις για κάθε είδος συνδέσεως, διαφορετικές για την περίπτωση που τα στοιχεία είναι πραγματικά και για την περίπτωση που έχουμε να κάνουμε με ψευδό-στοιχεία. Μιας και οι συναρτήσεις αυτές είναι αρκετά παρόμοιες ως προς τον τρόπο λειτουργίας τους, θα παρουσιαστεί και θα αναλυθεί μονάχα η πρώτη εξ αυτών καθώς οι υπόλοιπες μπορούν να γίνουν εύκολα κατανοητές με μελέτη της πρώτης και διάβασμα του κώδικα τους. Η πρώτη εξ αυτών που καλείται είναι η MeshToGrid(grid,centerPoints,rmin), υπεύθυνη για τον καθορισμό των σημείων του καννάβου της επιφάνειας που θα

επηρεάσουν το κάθε στοιχείο κατά την μεταφορά πεδίων, όπως για παράδειγμα η Level Set συνάρτηση.

Οι μεταβλητές που απαιτούνται κατά την κλήση της είναι οι
 grid: το πλέγμα σημείων για την προσομοίωση της επιφάνειας
 centerPoints: τα σημεία των κέντρων βάρους των τριγωνικών στοιχείων
 rmin: η ακτίνα επιρροής εντός της οποίας ένα σημείο του πλέγματος θεωρείται πως έχει επίδραση στο εκάστοτε πεπερασμένο στοιχείο

Η λογική του είναι αρκετά απλή, γι αυτό και δε θα γίνει παρουσίαση γραμμή γραμμή. Ουσιαστικά ελέγχεται για το κάθε κεντροβαρικό σημείο των τριγωνικών πεπερασμένων το ποιά σημεία του πλέγματος έχουν από αυτό απόσταση μικρότερη από την ακτίνα επιρροής rmin. Τα σημεία αυτά θεωρείται πως το επηρεάζουν, και ως εκ τούτου αποθηκεύεται ένας μη μηδενικός δείκτης στην αντίστοιχη θέση του μητρώου(μονάδα εδώ, ο αύξων αριθμός του στοιχείου σε άλλα), μιας και έχουμε ορίσει το 0 να υποδεικνύει πως δεν υπάρχει καμία μεταξύ τους σχέση. Με την ίδια πρακτική υπολογίζονται και οι συνδέσεις πεπερασμένων, πραγματικών η μη, και σημείων καννάβου στις υπόλοιπες συναρτήσεις, ώστε να γίνει τελικώς εφικτός ο μετασχηματισμός πεδίων από το ένα σύστημα προσομοίωσης στο άλλο.

Οι επόμενες συναρτήσεις που συναντάμε στην πορεία του αλγορίθμου είναι αυτές με τις οποίες θα καθοριστούν τα βάρη για τους μετασχηματισμούς από το ένα σύστημα σημείων στο άλλο. Και πάλι καλούνται τέσσερις διαφορετικές συναρτήσεις για τον υπολογισμό των βαρών κατά των μετασχηματισμό από το πλέγμα πεπερασμένων σημείων, πραγματικών ή μη, στο πλέγμα σημείων για την προσομοίωση της επιφάνειας, και αντίστροφα. Για άλλη μια φορά λοιπόν θα γίνει η παρουσίαση της μιας μονάχα εξ αυτών των συναρτήσεων, για λόγους συντομίας, και οι υπόλοιπες μπορούν να γίνουν εύκολα κατανοητές στη συνέχεια με απλό διάβασμα των εντολών τους. Η συνάρτηση που είναι υπεύθυνη για τον υπολογισμό των βαρών για τον μετασχηματισμό που ήδη εξετάσαμε είναι η MeshWeights(meshToGrid,centerPoints,grid,rmin), για την κλήση της οποίας απαιτούνται οι παρακάτω μεταβλητές:

meshToGrid: οι σχέσεις των σημείων του καννάβου της επιφάνειας με τα κέντρα βάρους των πεπερασμένων
 centerPoints: τα σημεία των κέντρων βάρους των τριγωνικών στοιχείων
 grid: το πλέγμα σημείων για την προσομοίωση της επιφάνειας
 rmin: η ακτίνα επιρροής εντός της οποίας ένα σημείο του πλέγματος θεωρείται πως έχει επίδραση στο εκάστοτε πεπερασμένο στοιχείο

Στην αρχή της συναρτήσεως ξεκινάνε οι επαναλήψεις που τρέχουν για όλα τα κεντροβαρικά σημεία των πεπερασμένων στοιχείων, για όλα τα σημεία του καννάβου, αφού πρώτα όμως έχει δημιουργηθεί το μητρώο meshWeights για την αποθήκευση των συντελεστών βάρους για κάθε τιμή του καννάβου της επιφάνειας που θα μεταφέρεται στο πλέγμα πεπερασμένων. Έπειτα, στις γραμμές 7-10, αφού καταρχάς ελεγχθεί αν το τρέχον σημείο του καννάβου θα έπρεπε να επηρεάζει το τρέχων σημείο του πλέγματος πεπερασμένων μέσω της εξέτασης της αντίστοιχης τιμής στο μητρώο meshToGrid, υπολογίζεται ο συντελεστής βάρους με τον οποίο θα επηρεάζει. Ο συντελεστής αυτός υπολογίζεται με τον ίδιο περίπου τρόπο που συναντήσαμε κατά την εξέταση του φίλτρου της SIMP, με την διαφορά πως υψώνεται κίολας εις τον κύβο. Ο λόγος που γίνεται αυτό είναι πως τα σημεία του καννάβου που βρίσκονται πιο κοντά στην προβολή ενός σημείου του πλέγματος στο x-y επίπεδο θα το επηρεάσουν σε πολύ μεγαλύτερο βαθμό απ ότι τα σημεία εκείνα του καννάβου που ενώ βρίσκονται εντός της ακτίνας rmin και το επηρεάζουν θεωρητικώς, πρακτικά βρίσκονται κάπως πιο μακριά. Η ίδια ακριβώς λογική ακολουθείται και στις υπόλοιπες τρεις συναρτήσεις για τον υπολογισμό των συντελεστών βάρους και για τις

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

υπόλοιπες μετατροπές μεταξύ σημείων για την προσομοίωση της επιφάνειας και σημείων του πλέγματος πεπερασμένων, και ως εκ τούτου δεν παρουσιάζονται ξεχωριστά μια μια.

Αφού ολοκληρώθηκαν όλες οι παραπάνω συνθέσεις των βοηθητικών μητρώων για τα διάφορα στάδια του αλγορίθμου μπορούν πλέον να ξεκινήσουν οι επαναλήψεις της μεθόδου. Για να ξεκινήσουν όμως πρέπει να έχει σχηματισθεί η Level Set συνάρτηση, οπότε καλείται η συνάρτηση Reinit η οποία είναι υπεύθυνη για την εκ νέου δημιουργία της Level Set ανά καθορισμένο αριθμό επαναλήψεων. Έτσι, ακριβώς προτού ο αλγόριθμος προχωρήσει μέσα στον κύριο επαναληπτικό του βρόγχο γίνεται κλήση της συνάρτησης Reinit(struc,centerPoints,centerPointsFake,elemSize,gridToMesh, gridToFakeMesh, gridWeights,fakeGridWeights), την οποία και θα δούμε αμέσως τώρα. Για την κλήση της απαιτούνται οι μεταβλητές:

struc: το διάνυσμα που περιέχει την κατάσταση υλικού για το κάθε πεπερασμένο στοιχείο
 centerPoints: τα σημεία των κέντρων βάρους των τριγωνικών στοιχείων
 centerPointsFake: τα σημεία των κέντρων βάρους των ψευδό-στοιχείων που δημιουργήθηκαν
 elemSize: το μέγεθος των στοιχείων, το οποίο έχει οριστεί από τον χρήστη κατά την διακριτοποίηση στο GMSH
 gridToMesh: μητρώο που καθορίζει το ποια στοιχεία του πλέγματος πεπερασμένων επηρεάζουν τα εκάστοτε σημεία του καννάβου.
 gridToFakeMesh: μητρώο που καθορίζει το ποιά εκ των ψευδό-στοιχείων που δημιουργήθηκαν επηρεάζουν
 gridWeights: μητρώο που περιέχει τους συντελεστές βάρους που αντιστοιχούν στον μετασχηματισμό που καθορίζεται από το gridToMesh μητρώο
 fakeGridWeights: μητρώο που περιέχει τους συντελεστές βάρους που αντιστοιχούν στον μετασχηματισμό που καθορίζεται από το gridToFakeMesh μητρώο

Επειδή ο στόχος είναι ο σχηματισμός της Level Set σε επίπεδο καννάβου σημείων(αφού με αυτά έχουμε επιλέξει να λάβουμε υπόψη την επιφάνεια) παρόλο που ο φορέας είναι εκφρασμένος μέσω πεπερασμένων και άρα με μη δομημένο πλέγμα κεντροβαρικών σημείων, θα πρέπει να βρεθεί τρόπος να υπολογιστούν πρώτα οι τιμές της Level Set συναρτήσεως στο επίπεδο αυτού του μη δομημένου πλέγματος σημείων. Αυτό επιτυγχάνεται με την κλήση της συναρτήσεως ReinitLevelSetInMesh(struc,centerPoints, centerPointsFake,elemSize), η οποία και κάνει αυτό ακριβώς που λέει και το όνομά της. Για την κλήση αυτής απαιτούνται μερικές από τις μεταβλητές με τις οποίες ο αλγόριθμος τροφοδότησε την συνάρτηση Reinit, όπως φαίνεται ακριβώς παραπάνω.

Η ReinitLevelSetInMesh με την σειρά της καλεί την συνάρτηση Bwdist η οποία λειτουργεί με ανάλογο τρόπο με αυτόν της συνάρτησης που παρουσιάζεται στην δημοσίευση της V. J. Challis. Πρόκειται για μια συνάρτηση εύρεσης του κοντινότερου στοιχείου που να είναι σε κατάσταση ύπαρξης υλικού, είτε αυτό είναι πραγματικό στοιχείο, είτε είναι ένα από το ψευδό-στοιχεία που δημιουργήθηκαν. Η διαφορά με τον κώδικα της δημοσίευσης είναι πως εδώ οι αποστάσεις υπολογίζονται σε όρους μονάδας μέτρησης μήκους σε αντίθεση με τους όρους στοιχείων που συναντά κανείς εκεί. Επίσης στον αλγόριθμο της παρούσας εργασίας, ο έλεγχος γίνεται τόσο για τα στοιχεία του πλέγματος πεπερασμένων όσο και για τα περιμετρικά αυτού ψευδό-στοιχεία που δημιουργήθηκαν. Σε κάθε στοιχείο αποθηκεύεται η ελάχιστη απόσταση από το κέντρο βάρους του μέχρι κάποιο άλλο κέντρο βάρους στοιχείου που να περιέχει υλικό. Η συνάρτηση Bwdist παρατίθεται στο Παράρτημα Δ μαζί με τις υπόλοιπες, και όντας πολύ απλή στην κατανόηση της, αλλά και επειδή επεξηγείται και στον κώδικα της πρότυπης δημοσίευσής, δεν θα αναλυθεί περαιτέρω.

Αφού λοιπόν υπολογιστούν οι τιμές της Level Set συνάρτησης στο πλέγμα πεπερασμένων στοιχείων με χρήση της ReinitLevelSetInMesh, θα κάνουμε χρήση των μητρώων που

δημιουργήσαμε για την υλοποίηση των μετασχηματισμών από το πλέγμα πεπερασμένων στο πλέγμα σημείων του καννάβου καθώς και τον συνοδευτικών τους που περιέχουν τους συντελεστές βάρους για κάθε μετασχηματισμό. Μεταξύ λοιπόν των γραμμών 4-17 της συναρτήσεως Reinit ελέγχονται όλες οι πιθανές σχέσεις επιρροών, είτε μεταξύ πραγματικών στοιχείων και καννάβου είτε μεταξύ ψευδό-στοιχείων και καννάβου, και για όσα στοιχεία πεπερασμένων βρεθεί πως έχουν επιρροή στο τρέχων σημείο της επιφάνειας προσαυξάνουν την τιμή της Level Set σε αυτό το σημείο με την δικιά τους τιμή πολλαπλασιασμένη με τον αντίστοιχο συντελεστή βαρύτητας(γραμμή 8 για πραγματικά στοιχεία και 13 για ψευδό-στοιχεία). Μετά την επιστροφή και από την Reinit ο αλγόριθμος έχει πλέον και την Level Set εκφρασμένη στο επίπεδο του καννάβου που έχει δημιουργηθεί.

Προχωρώντας παρακάτω στον κώδικα φτάνουμε στην γραμμή 59 όπου με κλήση της συναρτήσεως UpdateStep(Isf,shapeSens,topSens,gridToMesh,gridWeights,meshToGrid,meshWeights,stepLength,topWeight) εφαρμόζεται η απαραίτητη εξέλιξη στην συνάρτηση της Level Set ώστε να μπορέσει να κινηθεί προς την εύρεση βέλτιστου. Πρόκειται για μια συνάρτηση που είναι ευθέως ανάλογη της αντίστοιχης που υπάρχει στην δημοσίευση της V. J. Challis, με την μόνη προφανή διαφορά πως απαιτείται ο μετασχηματισμός από το πλέγμα πεπερασμένων στοιχείων σε έναν κάρναβο σημείων. Οι απαιτούμενες κατά την κλήση της μεταβλητές είναι

Isf: η Level Set συνάρτηση εκφρασμένη στο σύστημα του καννάβου των σημείων

shapeSens: οι ευαισθησίες σχήματος των στοιχείων στο σύστημα των πεπερασμένων στοιχείων

topSens: οι ευαισθησίες τοπολογίας των στοιχείων στο σύστημα των πεπερασμένων στοιχείων

gridToMesh: μητρώο που καθορίζει το ποια στοιχεία του πλέγματος πεπερασμένων επηρεάζουν τα εκάστοτε σημεία του καννάβου.

gridWeights: μητρώο που περιέχει τους συντελεστές βάρους που αντιστοιχούν στον μετασχηματισμό που καθορίζεται από το gridToMesh μητρώο

meshToGrid μητρώο που καθορίζει το ποια σημεία του καννάβου επηρεάζουν τα εκάστοτε στοιχεία του πλέγματος πεπερασμένων.

meshWeights: μητρώο που περιέχει τους συντελεστές βάρους που αντιστοιχούν στον μετασχηματισμό που καθορίζεται από το meshToGrid μητρώο

stepLength: μεταβλητή που προσδιορίζει το χρονικό εύρος για την εξέλιξη του σχεδιασμού του φορέα, εκφρασμένη σε συνάρτηση με την τιμή CFL

topWeight: ο συντελεστής βάρους ω της σχέσεως(3.15)

Γραμμές 2-3

Δημιουργία των μητρώων shapeSensGrid και topSensGrid για την αποθήκευση των ευαισθησιών στο σύστημα των σημείων της επιφάνειας, ώστε να μπορεί να γίνει επεξεργασία και διαφόριση αυτών.

Γραμμές 4-13

Επαναλήψεις οι οποίες τρέχουν για όλα τα πεπερασμένα στοιχεία και όλα τα σημεία του καννάβου προκειμένου να γίνει η μεταφορά των στοιχείων της Level Set συναρτήσεως από το σύστημα των κεντροβαρικών σημείων των πεπερασμένων που είναι αρχικώς εκφρασμένη στο σύστημα του καννάβου των σημείων για την προσομοίωση της επιφάνειας.

Γραμμές 14-15

Εφαρμόζεται μια ομαλοποίηση των ευαισθησιών με εργαλείο ανάλογο αυτών που χρησιμοποιούνται κατά την επεξεργασία εικόνας. Η con2 συνάρτηση κατά κάποιο τρόπο λειαίνει τις διαφορές των τιμών από σημείο σε σημείο ώστε να υπάρχει μια πιο ομαλή κατανομή αυτών. Η ομαλοποίηση αυτή βέβαια δεν έχει και πολύ μεγάλο

νόημα στην περίπτωση που το πλέγμα των σημείων για την προσομοίωση της επιφάνειας είναι πολύ πυκνό και έτσι πολλά σημεία αυτού εξάγουν την τιμή τους από το ίδιο στοιχείο του πλέγματος.

Γραμμή 17

Κλήση της συναρτήσεως `Evolve` η οποία είναι υπεύθυνη για την διαφόριση της συναρτήσεως `Level Set` και την τελική της επεξεργασία ώστε να επιστρέψει στην κυρίως συνάρτηση τοπολογίας και να γίνει η κρίση των στοιχείων που πρέπει να μείνουν στον φορέα και αυτών που πρέπει να φύγουν.

Η επόμενη και τελευταία συνάρτηση που θα πρέπει να εξετάσουμε είναι η `Evolve(v,g,lsf,stepLength,w,meshToGrid,meshWeights)` η οποία όπως είπαμε είναι υπεύθυνη για την παραγωγή της επιφάνειας και την συμπερίληψη των ευαισθησιών τοπολογίας στη `Level Set` συνάρτηση ώστε να μπορέσει να βρεθεί μια νέα λύση για το πρόβλημα. Κατά την κλήση της απαιτούνται οι μεταβλητές:

`v`: η κάθετη στο σύνορο ταχύτητα για την εξέλιξη της συναρτήσεως

`g`: ο όρος της σχέσεως (3.23)

`lsf`: η `Level Set` συνάρτηση εκφρασμένη στο σύστημα του καννάβου των σημείων

`stepLength`: μεταβλητή που προσδιορίζει το χρονικό εύρος για την εξέλιξη του σχεδιασμού του φορέα, εκφρασμένη σε συνάρτηση με την τιμή `CFL`

`w`: ο συντελεστής βάρους ω της σχέσεως (3.15)

`meshToGrid` μητρώο που καθορίζει το ποια σημεία του καννάβου επηρεάζουν τα εκάστοτε στοιχεία του πλέγματος πεπερασμένων.

`meshWeights`: μητρώο που περιέχει τους συντελεστές βάρους που αντιστοιχούν στον μετασχηματισμό που καθορίζεται από το `meshToGrid` μητρώο

Γραμμή 2

Υπολογισμός του χρονικού βήματος `dt` όπως αυτό ορίζεται από την σχέση (3.17)

Γραμμές 3-12

Αφού βρούμε τις παραγώγους προς κάθε διεύθυνση για το τρέχον βήμα του χρονικού εύρους στις γραμμές 4-7 εφαρμόζουμε την επεξεργασία της `Level Set` συναρτήσεως με χρήση των παραγώγων που υπολογίστηκαν και των ευαισθησιών τοπολογίας (που περιέχονται στην μεταβλητή `g`). Η σχέση για την επεξεργασία και εξέλιξη της `Level Set` είναι αρκετά μεγάλη και σύνθετη, όπως μπορεί κανείς εύκολα να διαπιστώσει με μια μελέτη των γραμμών 9-11 της συναρτήσεως.

Γραμμές 13-22

Πλέον η επεξεργασία της `Level Set` συναρτήσεως εκφρασμένης στον κάνναβο των σημείων έχει ολοκληρωθεί και σειρά έχει η μεταφορά αυτής στο σύστημα των κεντροβαρικών σημείων για τα πεπερασμένα στοιχεία. Αυτό γίνεται και πάλι με χρήση των μητρώων για τις συνδέσεις και των αντίστοιχών τους βαρών που σχηματίστηκαν κατά την αρχή του αλγορίθμου.

Γραμμή 23

Φτάνουμε λοιπόν στο σημείο της κρίσης για τα πεπερασμένα στοιχεία, αφού με την χρήση της `Level Set` συναρτήσεως εκφρασμένης στο σύστημα τριγωνικών στοιχείων μπορεί να γίνει η επιλογή για τα στοιχεία που θα συνθέσουν τον φορέα. Η επιλογή γίνεται με εφαρμογή ουσιαστικά της σχέσης (3.14), οπότε όσα στοιχεία έχουν στην `Level Set` συνάρτηση τιμή μικρότερη ή ίση του μηδενός παραμένουν ή προστίθενται στο φορέα και δίνεται στο αντίστοιχό τους κελί του διανύσματος `struc` η τιμή 1, ενώ για όσα στοιχεία η `Level Set` έχει τιμή μεγαλύτερη του μηδενός τους δίνεται η τιμή

Ο στο διάνυσμα `struc` και δεν συμμετέχουν πλέον ενεργά στην ακαμψία του φορέα. Το προκύπτον διάνυσμα `struc` αποτελεί ουσιαστικά τον φορέα της τρέχουσας επανάληψης της `Level Set` μεθόδου, που πρακτικά είναι και ο σκοπός του αλγορίθμου.

ΚΕΦΑΛΑΙΟ 4^ο

ΕΦΑΡΜΟΓΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΤΟΠΟΛΟΓΙΑΣ

4.1 Γενικά

Αφού παρουσιάστηκε το θεωρητικό υπόβαθρο του τριγωνικού πεπερασμένου στοιχείου καθώς και των μεθόδων βελτιστοποίησης τοπολογίας που θα συγκριθούν, αλλά και η αλγοριθμική υλοποίηση όλων των παραπάνω, θα περάσουμε στην εξέταση κάποιων χαρακτηριστικών εφαρμογών ανάλυσης. Οι αναλύσεις έγιναν όπως έχει ήδη ειπωθεί με χρήση της δομής MSolve, που περιέχει πλήθος πεπερασμένων στοιχείων και τρόπων ανάλυσης, όπως γραμμικές ή μη γραμμικές, στατικές δυναμικές κ.α. Στην δομή αυτή προστέθηκαν τα κομμάτια κώδικα που αφορούν την βελτιστοποίηση τοπολογίας με καθεμία από τις τρεις μεθόδους, καθώς και ο κώδικας για τον σχηματισμό ενός τριγωνικού πεπερασμένου στοιχείου.

Για την διακριτοποίηση των προς ανάλυση φορέων χρησιμοποιήθηκε το πρόγραμμα GMSH [3], το οποίο με χρήση του αλγορίθμου Delaunay διακριτοποιεί με μη δομημένο τρόπο την εκάστοτε γεωμετρία σε τριγωνικά πεπερασμένα στοιχεία. Το μέγεθος των στοιχείων αυτών δίνεται από τον χρήστη και αποτελεί στην ουσία την παράμετρο πυκνότητας του πλέγματος, μιας και η γεωμετρία του φορέα δεν επηρεάζεται.

Για το 'διάβασμα' του πλέγματος πεπερασμένων που προκύπτει από το GMSH γράφτηκε επίσης ένα κομμάτι κώδικα το οποίο εντάχθηκε στην δομή MSolve, το οποίο όμως δεν παρουσιάζεται εδώ πέρα καθώς δεν αποτελεί αντικείμενο των πλαισίων της διπλωματικής.

Στις περισσότερες δημοσιεύσεις της μεθόδου συναντά κανείς το μέτρο ελαστικότητας να είναι ίσο με 1.0, προκειμένου το πρόβλημα να είναι κάπως πιο γενικό σε όρους υλικού. Παρόλα αυτά ένα τόσο μικρό μέτρο ελαστικότητας οδηγεί σε πολύ εύκαμπτους φορείς, που με την σειρά τους δημιουργούν αριθμητικά προβλήματα κατά την επίλυση του στατικού συστήματος με το MSolve. Προκειμένου αυτά τα προβλήματα να αντιμετωπιστούν και να γίνει εφικτή η επίλυση της εξίσωσης ισορροπίας, έπρεπε να δοθεί ένα μεγαλύτερο μέτρο ελαστικότητας το οποίο θα μείωνε κιόλας τις αναπτυσσόμενες μετατοπίσεις. Επιλέχθηκε λοιπόν να δοθεί ως μέτρο ελαστικότητας για όλες τις παρακάτω εφαρμογές το μέτρο ελαστικότητας του δομικού χάλυβα ίσο με $E=210$ GPa. Μιας και ούτως ή άλλως λοιπόν είναι αναπόφευκτη η χρήση ενός σχετικά υπολογίσιμου μέτρου ελαστικότητας, καλύτερα να χρησιμοποιηθεί κάποιο που να συναντάται στη φύση, αν και όπως είδαμε μελετώντας τους αλγορίθμους των μεθόδων το μέτρο ελαστικότητας δεν παίζει κάποιο ρόλο στις μορφές που προκύπτουν (εκτός ίσως από την Level Set που δεν έχει σχεδιαστεί ως αδιάστατη προς τα ελαστικά χαρακτηριστικά). Το μέτρο Poisson έχει επιλεγεί ίσο με 0.3.

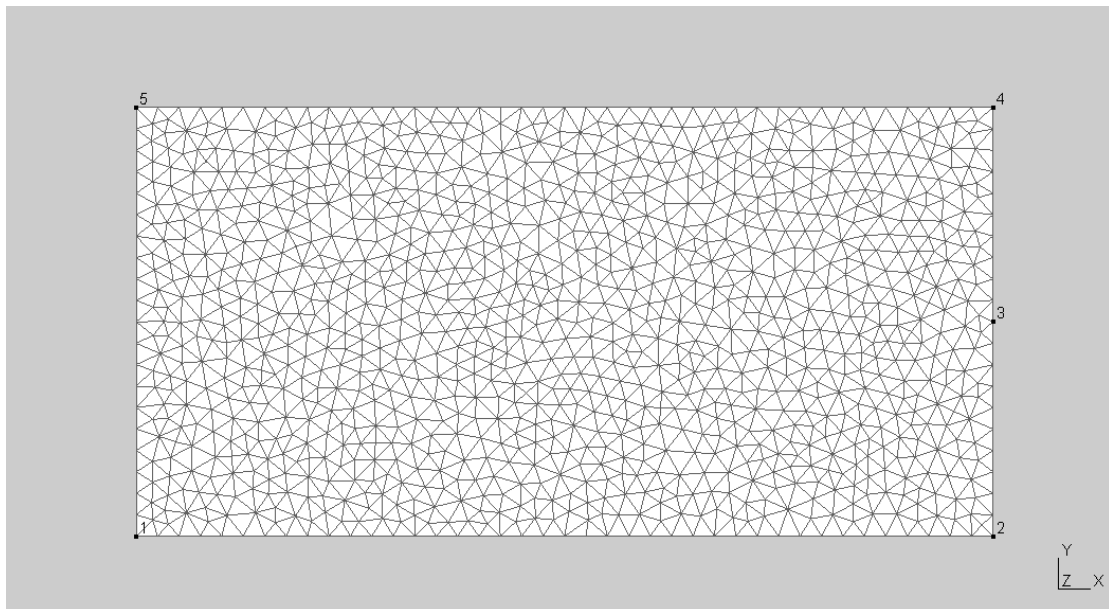
Το επόμενο στοιχείο των φορέων που πρέπει να επιλεγεί είναι το πάχος των πεπερασμένων πλάκας που θα χρησιμοποιηθούν, δηλαδή το γενικότερο πάχος της πλάκας ή του κελύφους που αποτελεί τον φορέα. Μιας και χρησιμοποιούνται στοιχεία με την θεωρία παχιών πλακών-κελυφών καλό θα είναι το πάχος να είναι με τέτοιο τρόπο ανάλογο των υπολοίπων διαστάσεων του πεπερασμένου ώστε να ικανοποιείται η επιλογή της συγκεκριμένης θεωρίας για την προσομοίωση της συμπεριφοράς των στοιχείων. Αν και δεν υπάρχει κάποιος τεκμηριωμένος κανόνας για τις αναλογίες, υπάρχει η σύσταση από ερευνητές να επιλέγεται ως πάχος το ένα πέμπτο περίπου τις μικρότερης από τις άλλες διαστάσεις, ενώ για στοιχεία λεπτών πλακών κελυφών προτείνεται να τηρείται μια αναλογία πάχους και μικρότερης εκ των λοιπών διαστάσεων ίση με ένα προς δέκα. Έτσι σε

κάθε εφαρμογή το πάχος των στοιχείων εξαρτάται από το μέγεθος των στοιχείων, το οποίο καθορίστηκε κατά την διακριτοποίηση του GMSH.

4.2 Εφαρμογή 1^η : Βελτιστοποίηση Τοπολογίας Προβόλου

Ο πρώτος φορέας με την βελτιστοποίηση του οποίου θα ασχοληθούμε είναι στην ουσία ένας απλός πρόβολος. Πρόκειται για μια μονόπακτη πλάκα επί του επιπέδου x-y με ένα μοναδιαίο φορτίο κατά την y διεύθυνση, το οποίο μάλιστα συναντάται σε αρκετές από τις δημοσιεύσεις περί βελτιστοποίησης τοπολογίας. Είναι κιάλας μια εκ των εφαρμογών που παρουσιάζονται στην κλασικότερη όλων των δημοσιεύσεων περί τοπολογικής βελτιστοποίησης, την δημοσίευση ‘A 99 line topology optimization code written in Matlab’ του O. Sigmund.

Η γεωμετρία του καθώς και η διακριτοποίηση του, όπως αυτή προέκυψε από το πρόγραμμα GMSH φαίνεται στην παρακάτω εικόνα.



Εικόνα 4.1: Πλέγμα πεπερασμένων στοιχείων για τον φορέα πρόβολου

Το πλέγμα αποτελείται από 2074 τριγωνικά εξακομβικά πεπερασμένα στοιχεία. Οι κόμβοι κατά μήκος της πλευράς 1-5 έχουν δεσμευμένους όλους τους βαθμούς ελευθερίας μιας και βρίσκονται στην πακτωμένη πλευρά της πλάκας. Το μοναδιαίο φορτίο ασκείται στην θέση του σημείου 3, στην κατά y διεύθυνση και φορά προς τα αρνητικά του άξονα. Η πλάκα έχει διαστάσεις 2.0x1.0. Το μέγεθος των στοιχείων επιλέχθηκε ίσο με 0.04, και επομένως το πάχος αυτών επιλέχθηκε ίσο με 0.01 ώστε να τηρείται στο περίπου η αναλογία που ειπώθηκε παραπάνω για την ‘εγκυρότητα’ χρήσης της θεωρίας Reissner-Mindlin.

Ο απαιτούμενος όγκος που ζητείται από τον βέλτιστο φορέα να έχει είναι 0.5, που αν και πρόκειται για μεγάλο σχετικά ποσοστό όγκου για τις συνήθεις εφαρμογές SIMP και BESO, είναι απαραίτητος προκειμένου να λάβουμε ικανοποιητικά αποτελέσματα και από την Level Set μέθοδο.

Πέραν της ελαστικής παραμόρφωσης κατά τις αναλύσεις όπως είπαμε εξάγεται και μια χαρακτηριστική μετατόπιση, η οποία για το παράδειγμα αυτό είναι η ίδια με τον βαθμό ελευθερίας στον οποίο ασκείται το μοναδιαίο φορτίο.

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Οι αναλύσεις ξεκίνησαν με την SIMP μέθοδο, η οποία περιορίστηκε στις 100 επαναλήψεις.

Η μορφή φορέα που μας έδωσε η μέθοδος SIMP είναι αυτή της παρακάτω εικόνας, όπου τα πιο θολά σημεία του υποδηλώνουν περιοχές με ενδιάμεση πυκνότητα, δηλαδή ούτε υπάρχει πλήρες υλικό αλλά ούτε και κενό. Παρόλα αυτά όπως βλέπουμε οι περιοχές αυτές είναι αρκετά περιορισμένες, δείγμα του ότι η μέθοδος κατάφερε να συγκλίνει σε μια ξεκάθαρη για τα δεδομένα της μορφή φορέα.



Εικόνα 4.2: Φορέας της SIMP μεθόδου για την 1^η εφαρμογή

Τα αποτελέσματα που δίνει η παραπάνω λύση κατά την SIMP μέθοδο είναι τα παρακάτω:

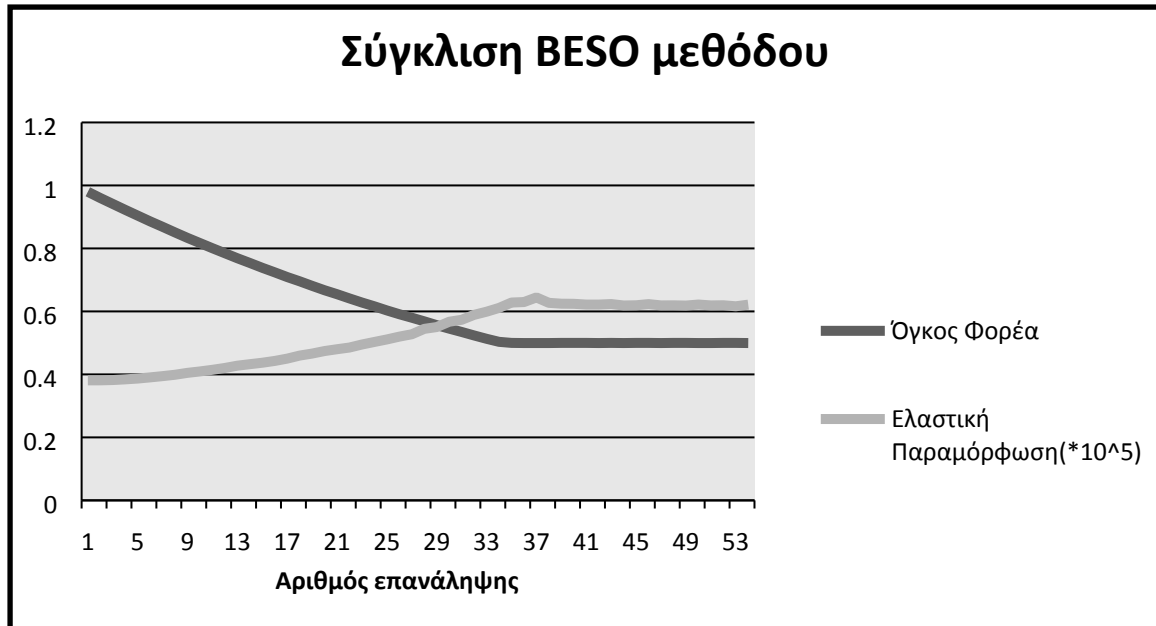
Επανάληψη: 100

Όγκος: 0.500000001229545

Ελαστική Παραμόρφωση: 6.35871540207467E-05

Χαρακτηριστική μετατόπιση: -6.36916192311433E-05

Η επόμενη στην σειρά μέθοδος τοπολογικής βελτιστοποίησης είναι η BESO μέθοδος. Εδώ αν και περιορίσαμε την μέθοδο στις εκατό επαναλήψεις αυτή βρήκε την λύση της ήδη από την πεντηκοστή τέταρτη. Παρόλα αυτά, όπως αναφέρθηκε και κατά την ανάλυση της BESO μεθόδου, η λύση δεν είναι αυτή στην οποία συγκλίνει ο αλγόριθμος, αλλά αυτή που εμφανίζει την μικρότερη ελαστική παραμόρφωση από όσες έχουν τον κατάλληλο όγκο. Πρόκειται για μια ιδιομορφία της μεθόδου η οποία επισημαίνεται και στην δημοσίευση στην οποία στηρίχθηκε ο αλγόριθμος. Για να δούμε και γραφικά αυτό το λεπτό σημείο αρκεί να κάνουμε το διάγραμμα των αποτελεσμάτων που επιτυγχάνονται σε κάθε επανάληψη της μεθόδου, το οποίο θα μας βοηθήσει κιάλας στην εύρεση της βέλτιστης λύσης. Όπως επεξηγήθηκε, γίνεται αναζήτηση της λύσης εκείνης που θα έχει τον απαραίτητο όγκο(ίσο με 0.5) και θα παρουσιάζει την μικρότερη ελαστική παραμόρφωση. Το διάγραμμα για να γίνει η σωστή επιλογή είναι το παρακάτω:



Διάγραμμα 4.1

Στην προκειμένη περίπτωση η λύση αυτή επιτυγχάνεται στην επανάληψη 53, λίγο δηλαδή πριν συγκλίνει ο κώδικας. Η μορφή που προέκυψε από την ανάλυση με την μέθοδο BESO είναι αυτή της παρακάτω εικόνας, όπου όπως βλέπουμε δεν υπάρχουν θολά σημεία μιας και η μέθοδος αντιλαμβάνεται δυαδικώς την κατάσταση του υλικού των πεπερασμένων.



Εικόνα 4.3: Φορέας της BESO μεθόδου για την 1^η εφαρμογή

Τα αποτελέσματα που δίνει η παραπάνω λύση της BESO μεθόδου είναι τα παρακάτω:

Επανάληψη: 53

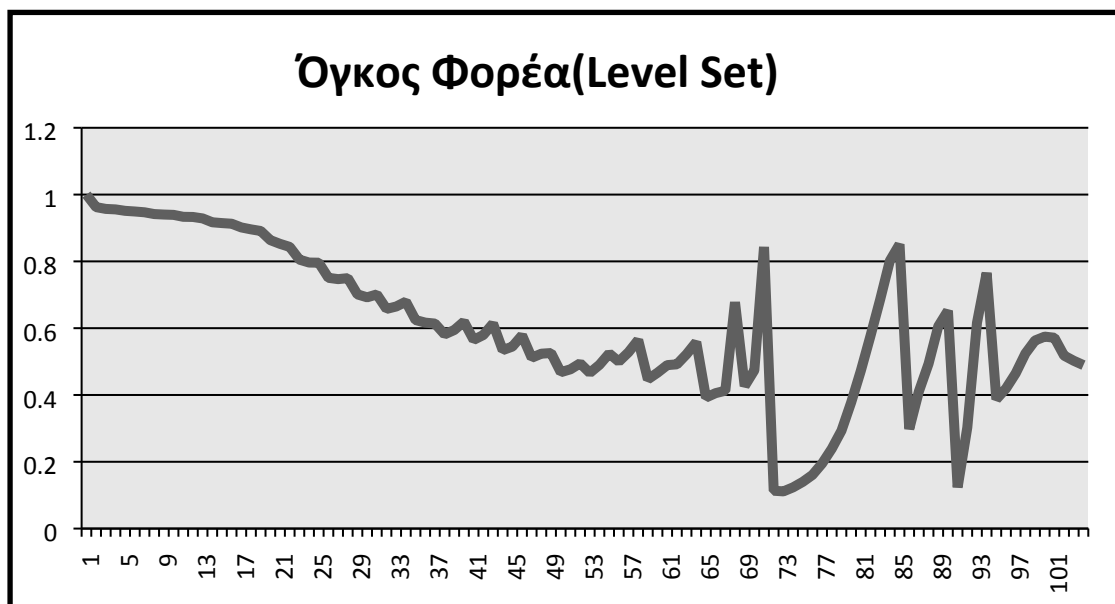
Όγκος: 0.500018322082923

Ελαστική Παραμόρφωση: 6.16232799323805E-05

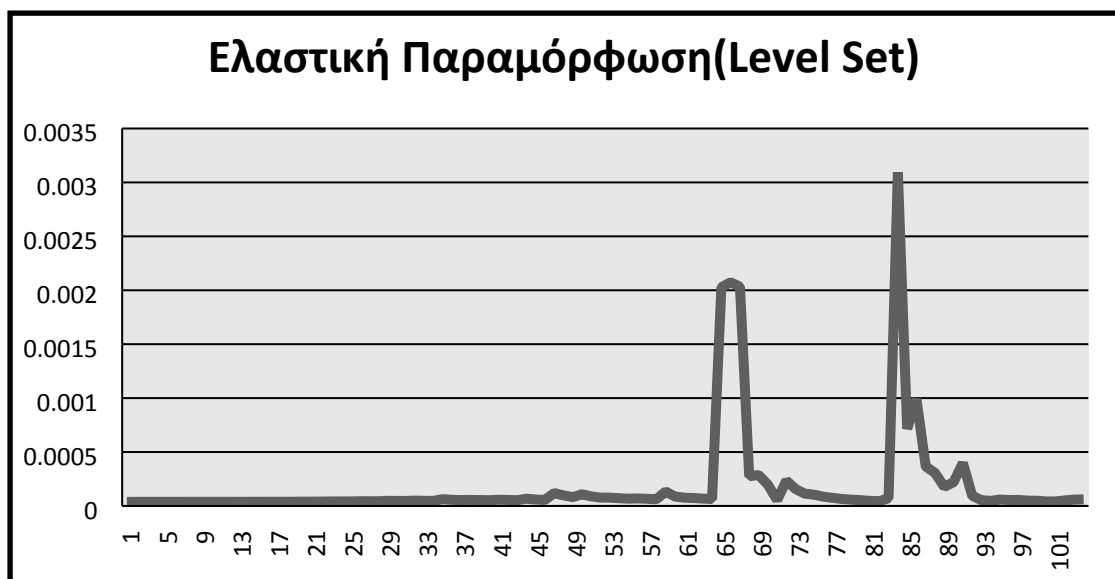
Χαρακτηριστική μετατόπιση: -6.16316599929623E-05

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Η τελευταία μέθοδος με την οποία θα γίνει προσπάθεια εύρεσης της βέλτιστης τοπολογίας του φορέα είναι η Level Set. Η μέθοδος αυτή, όπως θα δούμε και από τα αποτελέσματα είναι η πιο ασταθής από πλευράς προσέγγισης του προβλήματος και σύγκλισης του αλγορίθμου. Όπως και στην περίπτωση της BESO, η λύση επιλέγεται με τον έλεγχο των αποτελεσμάτων που ικανοποιούν την απαίτηση του όγκου, και την επιλογή αυτού που έχει την μικρότερη ελαστική παραμόρφωση, που αποτελεί και το κριτήριο βελτιστοποίησης. Επειδή στην περίπτωση αυτή λόγω των ασταθειών του αλγορίθμου βρίσκονται φορείς με μεγάλες ελαστικές παραμορφώσεις σε σύγκριση με τις υπόλοιπες και δεν είναι εφικτό να τις ανάγουμε με τέτοιο τρόπο ώστε να έχουμε στο ίδιο γράφημα και τις ελαστικές παραμορφώσεις κάθε επανάληψης αλλά και τον όγκο που επιτευχτεί. Γι αυτό τον λόγο δημιουργήθηκαν δυο ξεχωριστά διαγράμματα με την παράλληλη μελέτη των οποίων μπορούμε να καταλήξουμε στον βέλτιστο φορέα κατά την Level Set. Τα διαγράμματα είναι τα παρακάτω:



Διάγραμμα 4.2



Διάγραμμα 4.3

Όπως βλέπουμε από τα παραπάνω διαγράμματα, επιτυγχάνεται ικανοποιητικός όγκος από την τεσσαρακοστή τέταρτη επανάληψη, προς σύγκριση με την μέθοδο BESO που το είχε καταφέρει από την τριακοστή τέταρτη, δέκα επαναλήψεις τη μεθόδου δηλαδή πριν. Το άλλο πολύ σημαντικό σημείο που παρατηρούμε από το διάγραμμα ελαστικής παραμορφώσεως των φορέων που προκύπτουν σε κάθε επανάληψη είναι τα απότομα άλματα που παρουσιάζει από ένα σημείο και μετά.

Για να κατανοήσουμε καλύτερα το γιατί αυτά συμβαίνουν θα εξετάσουμε τους φορείς που έχουν σχηματισθεί αμέσως πριν και αμέσως μετά το πρώτο άλμα. Για την ακρίβεια, ο φορέας πριν το πρώτο άλμα θα τον παρουσιαζόταν ούτως ή άλλως, καθώς αποτελεί την λύση της μεθόδου που έχει ικανοποιητικό όγκο και την μικρότερη ελαστική παραμόρφωση, ή τουλάχιστον την μικρότερη ελαστική παραμόρφωση πριν το πρώτο άλμα. Όπως θα δούμε ευθύς αμέσως, άλματα τέτοιας μορφής κατά τις επαναλήψεις της μεθόδου επιφέρουν σημαντικό πλήγμα στην αξιοπιστία των αποτελεσμάτων που προκύπτουν στη συνέχεια αυτού. Το άλμα συνέβη προχωρώντας από την επανάληψη 64 στην επανάληψη 65 της μεθόδου, οι προκύπτοντες φορείς των οποίων φαίνονται στις παρακάτω εικόνες.



Εικόνα 4.4: Φορέας της BESO μεθόδου για την 1^η εφαρμογή



Εικόνα 4.5: Φορέας 65^{ης} επανάληψης Level Set

Βλέπουμε λοιπόν πως υπάρχει αρκετά σοβαρός λόγος στον οποίο οφείλεται η εμφάνιση αυτού του απότομου άλματος στις ελαστικές παραμορφώσεις των επαναλήψεων. Η μεταβολή της Level Set συνάρτησης στην 65^η επανάληψη οδηγεί στο κόψιμο κάποιων από τις 'ράβδους' υλικού που δημιουργούνται και συνθέτουν τον φορέα. Αυτό το κόψιμο οδηγεί σε τέτοιες απότομες μεταβολές της ελαστικής παραμορφώσεως, μιας και ξαφνικά χάνεται μέρος του δρόμου του φορτίου προς τις στηρίξεις το οποίο συμβάλει σημαντικά, μιας και ήδη έχει μειωθεί αρκετά ο όγκος με τον οποίο κινείται ο φορέας.

Παρόλα αυτά υπάρχουν περιπτώσεις που η μέθοδος μπορεί να ανακάμψει από τέτοιες απώλειες σημαντικών μελών του φορέα. Οι πιθανότητες για να γίνει κάτι τέτοιο αυξάνονται όσοι περισσότεροι δρόμοι υλικού υπάρχουν που να οδηγούν το φορτίο στις στηρίξεις. Το πραγματικά μεγάλο πρόβλημα όμως, αυτό από το οποίο η μέθοδος είναι αδύνατον να επανέλθει και να συνεχίσει την προσπάθεια για την εύρεση του βέλτιστου φορέα, προκαλείται όταν κοπούν όλοι οι δρόμοι υλικού μεταξύ φορτίου και στηρίξεων. Εάν κάτι τέτοιο συμβεί εμφανίζονται απότομα πάρα πολύ μεγάλες ελαστικές παραμορφώσεις και η συνέχιση του αλγορίθμου δεν έχει απολύτως κανένα νόημα. Στην περίπτωση του παραπάνω φορέα, όπως αυτός προκύπτει μετά την επανάληψη 65, παρατηρούμε ότι παραμένει ένας δρόμος που μπορεί να ακολουθήσει το φορτίο μέχρι τις στηρίξεις, αλλά συγχρόνως έχουν κοπεί όλα τα υπόλοιπα μέλη του. Το αποτέλεσμα είναι ενδιάμεσο των σεναρίων επιβίωσης ή όχι του φορέα που μόλις αναφέρθηκαν, οπότε δεν είναι ξεκάθαρο εξαρχής το τι θα ακολουθήσει.

Όπως θα δούμε από την επόμενη εικόνα, στην οποία παρουσιάζεται ένας από τους φορείς που έχουν επιτευχθεί μετά το άλμα και ο οποίος έχει αρκετά μικρή ελαστική παραμόρφωση, στα ίδια επίπεδα με αυτόν που αποτελεί τελικώς την βέλτιστη λύση της Level Set, η μέθοδος έχει αποτύχει στο να ανακάμψει από τις απότομες απώλειες της.



Εικόνα 4.6: Φορέας 75^{ης} επανάληψης Level Set μεθόδου

Η παραπάνω εικόνα αποδεικνύει πως τα αποτελέσματα δεν μπορούν πλέον να θεωρηθούν αξιόπιστα, μιας και η παραπάνω κατανομή υλικού κανένα φυσικό νόημα δεν έχει. Έτσι, ως γενικό συμπέρασμα από τα παραπάνω πρέπει να κρατήσει κανείς πως κατά την εμφάνιση αλμάτων είτε στην ελαστική παραμόρφωση είτε στον όγκο του φορέα που επιτυγχάνεται (αφού αν παρατηρήσουμε το διάγραμμα όγκου και εκεί εμφανίζονται έντονα

άλματα) χρειάζεται ιδιαίτερη προσοχή στην αξιοποίηση των αποτελεσμάτων από εκεί και πέρα.

Τα αποτελέσματα από την βέλτιστη λύση που επετεύχθη με την Level Set στην επανάληψη 64(εικόνα 4.4) είναι:

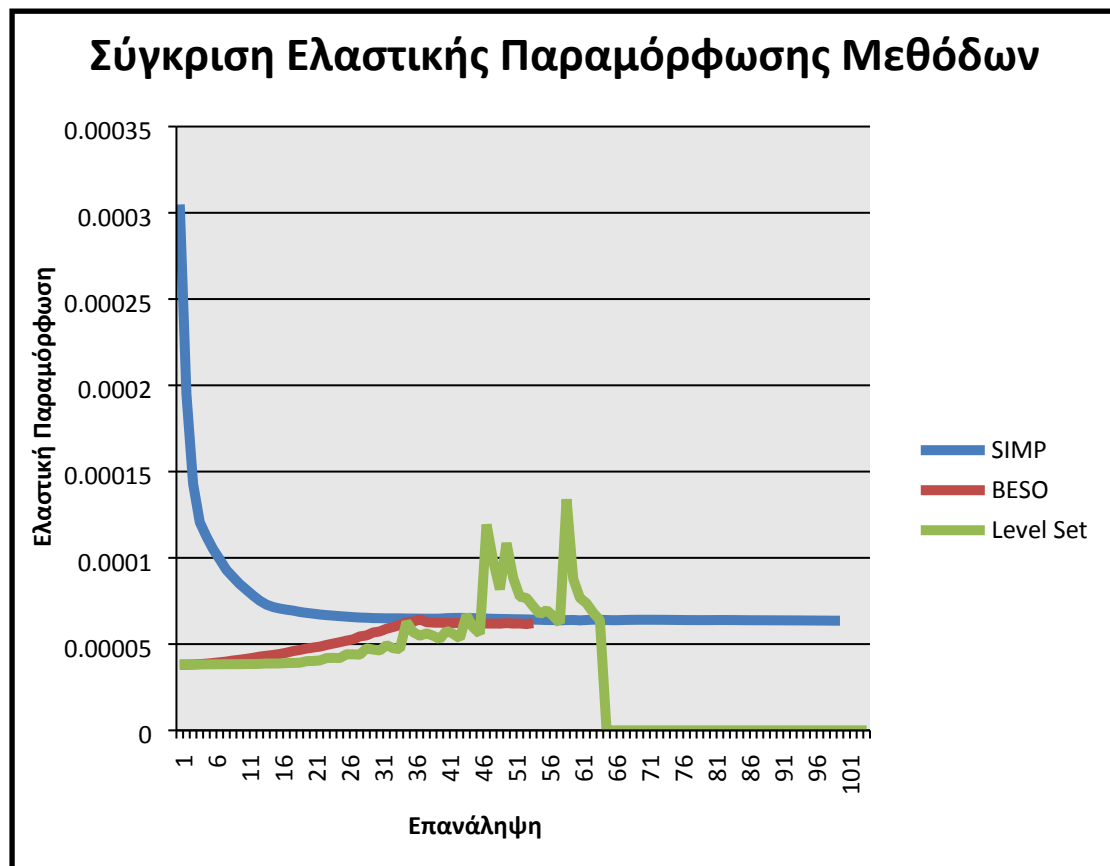
Επανάληψη: 64

Όγκος: 0.528341369334619

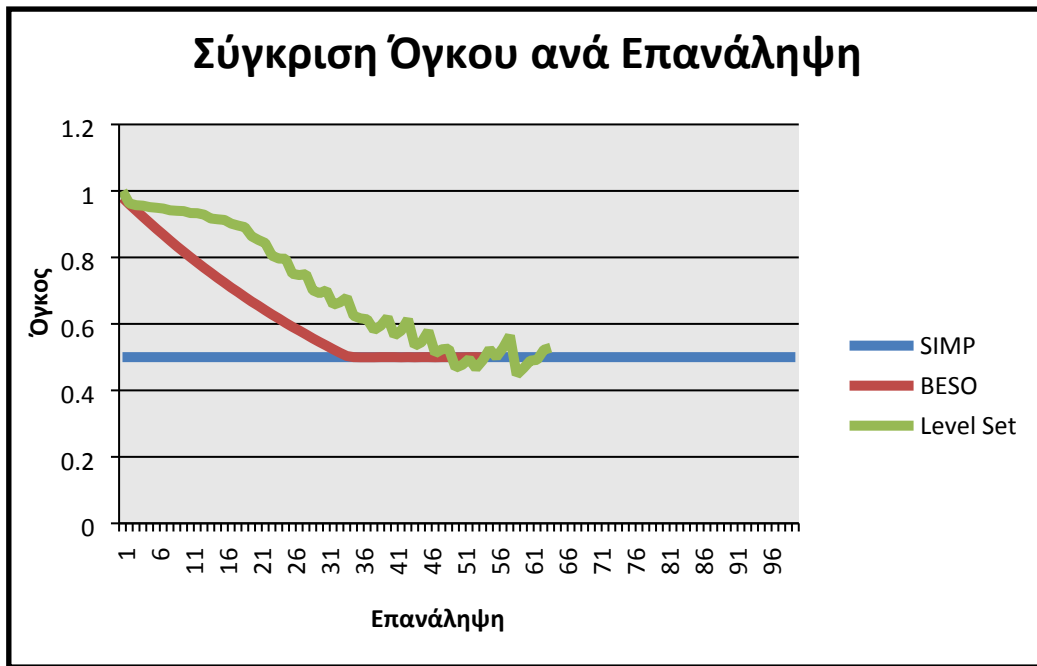
Ελαστική Παραμόρφωση: 6.16232799323805E-05

Χαρακτηριστική μετατόπιση: -6.3899187158266E-05

Αφού πλέον έχουν εξαχθεί αποτελέσματα από όλες τις μεθόδους, σειρά έχει η μεταξύ τους σύγκριση. Η γραφική απεικόνιση της ελαστικής παραμόρφωσης των τριών φορέων σε κάθε βήμα της μεθόδου θα μας δώσει μια καλή εικόνα για τον τρόπο κίνησης και την ταχύτητα σύγκλισης του κάθε αλγόριθμου ώστε να κατανοήσουμε καλύτερα τα θετικά και τα αρνητικά της κάθε μεθόδου και να καταλήξουμε σε μια τεκμηριωμένη αποτίμηση αυτών. Παράλληλα με την ελαστική παραμόρφωση θα αποτυπώσουμε σε διάγραμμα και τον όγκο που επιτυγχάνει η κάθε μέθοδος σε κάθε της βήμα ώστε να δούμε την ταχύτητα με την οποία η καθεμιά πλησιάζει τον απαιτούμενο όγκο που της ζητείται, αν και κάτι τέτοιο δεν έχει και πολύ νόημα στην περίπτωση της SIMP μεθόδου που εξαρχής έχει τον απαιτούμενο όγκο.



Διάγραμμα 4.4



Διάγραμμα 4.5

Το πρώτο πράγμα που παρατηρεί κανείς από τα παραπάνω διαγράμματα είναι το πόσο ομαλές είναι οι μέθοδοι SIMP και BESO ακόμα και κατά την εφαρμογή τους σε μη δομημένα πλέγματα, σε αντίθεση με την Level Set η οποία παρουσιάζει απότομες μεταβολές τόσο στην ελαστική της παραμόρφωση όσο και στον όγκο των φορέων που δημιουργεί από επανάληψη σε επανάληψη. Επίσης, σε σύγκριση με την BESO μέθοδο η Level Set φαίνεται να είναι και πιο αργή, καθώς επιτυγχάνει φορείς που να έχουν τον απαιτούμενο όγκο σε μεταγενέστερη επανάληψη από αυτήν που το καταφέρνει η BESO, η οποία κινείται με σταθερό ρυθμό προς την λύση, ρυθμό ο οποίος της έχει επιβληθεί εντός του κώδικά της μέσω της μεταβλητής er που εξηγήθηκε παραπάνω. Από την άλλη η Level Set δεν διαθέτει κάποιου τέτοιου είδους μεταβλητή και το μοναδικό της εργαλείο για την μεταβολή του όγκου προς το επιθυμητό ποσοστό είναι η επίδραση των μεταβλητών la και lb στις ευαισθησίες, επίδραση η οποία δεν είναι καθόλου σταθερή ως προς τον ρυθμό της.

Αν όμως παραβλέψουμε αυτήν την αστάθεια του αλγορίθμου και κοιτάξουμε τα αποτελέσματα της Level Set μεθόδου θα δούμε πως παρόλο που έχει καταλήξει σε φορέα αρκετά διαφορετικής μορφής από αυτόν των μεθόδων BESO και SIMP, οι οποίοι είναι σχετικά κοντά στις μορφές που εξάγονται, δεν έχει πολύ μεγάλη απόκλιση στην ελαστική παραμόρφωση που επιτυγχάνει όταν έρχεται σε σύγκριση με τις άλλες δυο. Κι αυτό παρά τις απότομες μεταβολές που εμφανίζονται στην περιοχή κοντά στην λύση, χωρίς καν να γίνεται αναφορά στο άλμα αμέσως μετά την επανάληψη που δίνει την λύση και που έχει παραληφθεί για λόγους ευκρίνειας του διαγράμματος.

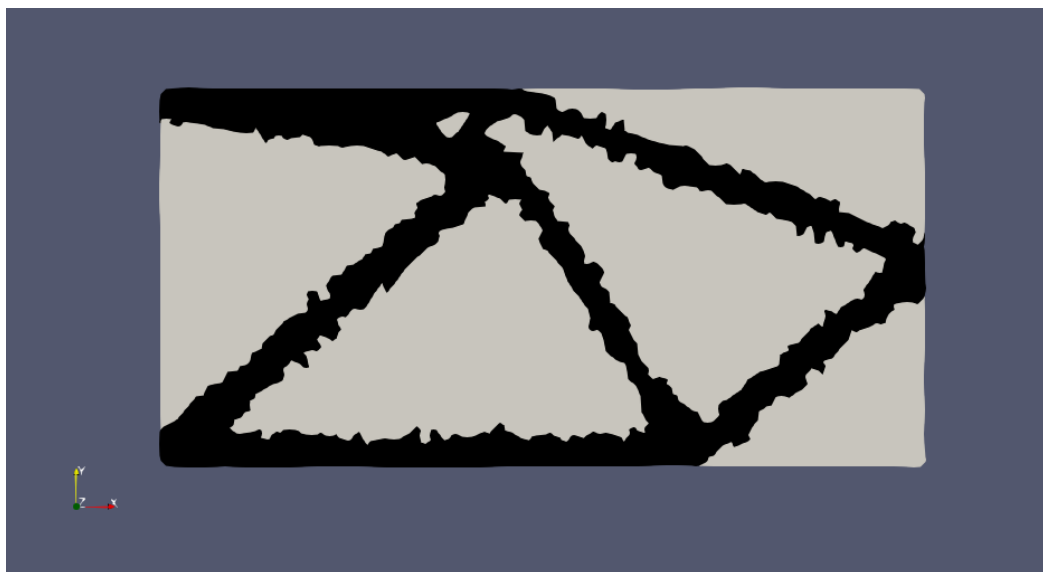
Μια ακόμα παρατήρηση που έχει νόημα να γίνει, ειδικά μιας και βασικό στοιχείο της διπλωματικής αυτής είναι το μη δομημένο των πλεγμάτων πεπερασμένων που έχει να διαχειριστεί, είναι η μη συμμετρία των φορέων που προκύπτουν. Είναι λογικό να περίμενε κανείς για ένα πλήρως ισότροπο υλικό όπως αυτό που χρησιμοποιείται με τις ίδιες ελαστικές ιδιότητες προς κάθε κατεύθυνση και για συνοριακές συνθήκες που μπορούν να θεωρηθούν συμμετρικές να περίμενε μια ανάλογη συμμετρία και στην λύση, ειδικότερα όταν έχει μελετήσει διάφορες δημοσιεύσεις επί των μεθόδων όπου για συμμετρικά προβλήματα οι λύσεις είναι πάντοτε συμμετρικές. Η μη συμμετρία των πλεγμάτων που προκύπτουν από το GMSH όμως δεν επιτρέπει κάτι τέτοιο. Τα αποτελέσματα της μη συμμετρίας αυτής γίνονται ακόμα πιο εμφανή όταν ο όγκος μειωθεί στο 0.3, όπως θα

δούμε παρακάτω, και για την ακρίβεια θα υπάρξει και επίδραση και στις μορφές των φορέων από τις SIMP και BESO που πρότινος ήταν σχεδόν ολίδιες.

Έχοντας πλέον τελειώσει με την πρώτη εφαρμογή και την σύγκριση των αποτελεσμάτων με την κάθε μέθοδο ήρθε η ώρα να προχωρήσουμε σε παραμετρικές αναλύσεις για την καλύτερη κατανόηση των θετικών και των αρνητικών σημείων καθεμιάς. Η πρώτη εξ αυτών των παραμετρικών δεν προοριζόταν αρχικώς ώστε να είναι παραμετρική αλλά αποτελούσε μια αρκετά φιλόδοξη προσπάθεια εύρεσης λύσης στον ίδιο φορέα, με την ίδια διακριτοποίηση και συνοριακές συνθήκες αλλά με την διαφορά πως ο απαιτούμενος όγκος που θα έπρεπε να είχε η λύση ήταν 0.3 έναντι του 0.5 που είδαμε. Το 0.3 ως ποσοστό όγκου είναι κλασικότερο σε προβλήματα SIMP και BESO, αλλά σπανίως παρατηρείται σε προβλήματα με την Level Set, για τον λόγο που θα δούμε σε λίγο. Οι φορείς που προκύπτουν για τις μεθόδους SIMP και BESO είναι αυτοί των δυο παρακάτω εικόνων.



Εικόνα 4.7: Φορέας SIMP μεθόδου για την 1^η εφαρμογή με 0.3 όγκο



Εικόνα 4.8: Φορέας BESO μεθόδου για την 1^η εφαρμογή με 0.3 όγκο

Για το πρόβλημα της βελτιστοποίησης στην περίπτωση μείωσης του όγκου στο 0.3 δεν θα γίνει η ίδια εκτενής ανάλυση που έγινε παραπάνω για την κυρίως εφαρμογή, κι αυτό όχι μονάχα επειδή πρόκειται περί παραμετρικής ανάλυσης, αλλά και επειδή δεν υπάρχουν αποτελέσματα και από τις τρεις μεθόδους προς σύγκριση. Ας ξεχάσουμε όμως προς στιγμήν τα προβλήματα που θα μας δημιουργήσει στην συνέχεια για εξετάσουμε λιγάκι καλύτερα τα αποτελέσματα των SIMP και BESO που έχουμε παραπάνω. Βλέπουμε πως οι δύο μέθοδοι καταλήγουν σε σχεδόν αντισυμμετρικές λύσεις για το ίδιο ακριβώς πρόβλημα, ένα αρκετά ενδιαφέρον αποτέλεσμα. Η διαφωνία μεταξύ των δυο μεθόδων ως προς το ποιός είναι ο βέλτιστος φορέας είναι πλασματική, όπως βλέπουμε και από τα παρακάτω αποτελέσματα, και οφείλεται στον τρόπο αντίληψης του πλέγματος από τις μεθόδους.

Αποτελέσματα SIMP για 0.3 όγκο

Επανάληψη: 100

Όγκος: 0.300000000836247

Ελαστική Παραμόρφωση: 0.00012069840495785

Χαρακτηριστική μετατόπιση: -0.000121472222094499

Αποτελέσματα BESO για 0.3 όγκο

Επανάληψη: 85

Όγκος: 0.300121986499508

Ελαστική Παραμόρφωση: 0.000109750452794252

Χαρακτηριστική μετατόπιση :-0.000109809408366675

Βλέπουμε λοιπόν πως τα αποτελέσματα είναι σχεδόν τα ίδια με μια ποσοστιαία απόκλιση της τάξης του 9%, η οποία αν και μοιάζει μεγάλη σε πρώτο βαθμό δεν οφείλεται τόσο στην διαφορετική κατεύθυνση που πήραν οι δυο μέθοδοι αλλά στην αποτελεσματικότητα ως προς την βελτιστοποίηση των μεθόδων, όπως θα δούμε και στην τελική αποτίμηση αυτών. Όπως αναφέρθηκε και προηγουμένως, η Level Set στην περίπτωση αυτή δε μπόρεσε να συγκλίνει σε κάποιον φορέα ο οποίος να έχει όγκο περίπου ίσο με 0.3 και να έχει συγχρόνως νόημα από πλευρά στατικής. Ο καλύτερος φορέας στον οποίο η Level Set μπόρεσε να καταλήξει για το πρόβλημα της βελτιστοποίησης είναι ο παρακάτω



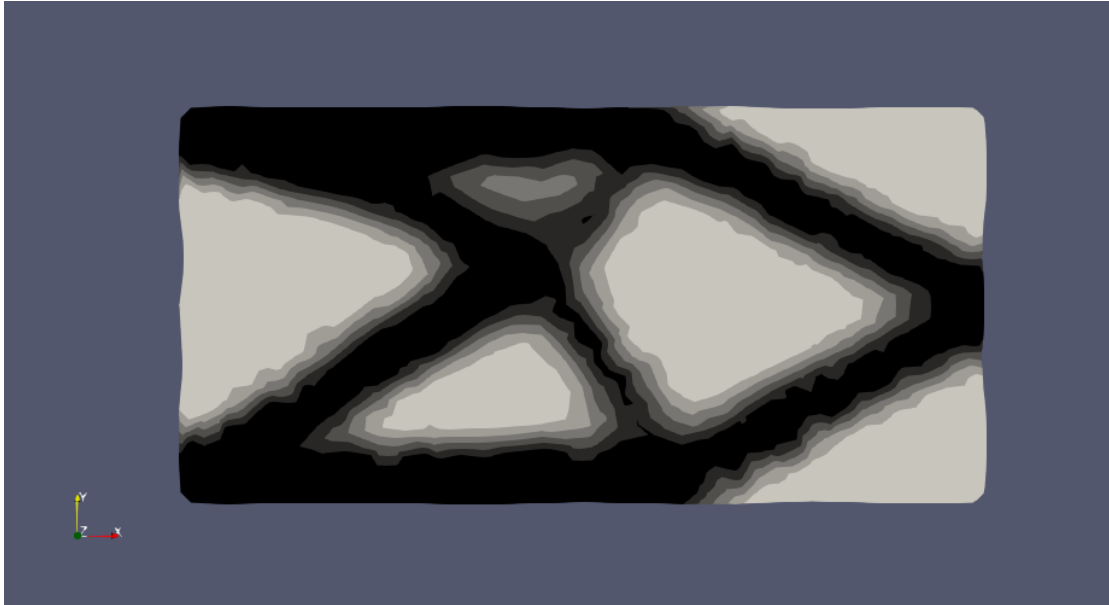
Εικόνα 4.9: Φορέας Level Set μεθόδου για την 1^η εφαρμογή με 0.3 όγκο

Όπως βλέπουμε έχουν κοπεί όλοι οι δρόμοι υλικού που οδηγούν από το φορτίο προς τις στηρίξεις και η παραπάνω λύση δε μπορεί ούτε καν να μπει προς σύγκριση δίπλα στις υπόλοιπες. Η εφαρμογή αυτή αποτελεί απόδειξη του παραπάνω κανόνα σε ότι αφορά τα αποτελέσματα της Level Set μεθόδου, πως δηλαδή δε μπορεί να υπάρξει λύση από την στιγμή που κοπούν όλοι οι δρόμοι υλικού.

Λόγω λοιπόν των παραπάνω αποτελεσμάτων ο απαιτούμενος όγκος του βέλτιστου φορέα τόσο για αυτήν την εφαρμογή όσο και για όλες τις υπόλοιπες άλλαξε σε 0.5 προκειμένου να εξαχθούν αποτελέσματα και να μπορέσει να γίνει η σύγκριση. Παρόλα αυτά υπάρχει τρόπος ώστε να μπορέσει και η Level Set να δώσει αποτελέσματα με μικρότερα ποσοστά όγκου. Η Level Set αντιμετωπίζει το πρόβλημα πως όντας επιφάνεια στην βάση της δεν είναι πολύ εύκολο δημιουργήσει τις ίδιες απότομες γεωμετρίες που κατασκευάζονται με τις άλλες δυο μεθόδους, ειδικά όταν εφαρμόζεται και φίλτρο ομαλοποίησης των τιμών της. Παρόλα αυτά η λύση δεν είναι η παραμέληση του φίλτρου και η μη εφαρμογή του, καθώς κάτι τέτοιο θα επηρέαζε με αποτελέσματα δημιουργώντας απότομες και παράξενες γεωμετρίες χωρίς πρακτικό νόημα. Το πρόβλημα μπορεί να λυθεί με την εφαρμογή σε πυκνότερα πλέγματα πεπερασμένων στοιχείων, τα οποία θα δώσουν την δυνατότητα στην επιφάνεια να έχει παραπάνω σημεία και έτσι να μπορεί να κινηθεί και να σχηματίσει πιο σύνθετες γεωμετρίες με περισσότερα ακρότατα, και άρα περισσότερες περιοχές υλικού και κενών. Τα πυκνότερα αυτά πλέγματα όμως απαιτούν και αρκετά περισσότερο χρόνο για την ανάλυση τους, οπότε προκειμένου να γίνουν αρκετές εφαρμογές των μεθόδων και να εξαχθούν αρκετά αποτελέσματα προς καλύτερη κατανόηση και σύγκριση των μεθόδων σε όσα σημεία η λύση για την Level Set θα μπορούσε να βρεθεί με πυκνότερα πλέγματα δεν παρουσιάζεται λύση. Στα σημεία αυτά απλώς παρουσιάζονται τα αποτυχημένα της αποτελέσματα και εξηγούνται οι λόγοι για τους οποίους δεν μπορέσει να συγκλίνει σε αξιόλογη λύση, λόγος ο οποίος συνήθως είναι η ύπαρξη πολλών μελών όπως θα βλέπουμε από τις άλλες δυο μεθόδους. Δεν είναι όμως μονάχα αυτά τα προβλήματα που αντιμετωπίζει η Level Set, όπως θα δούμε παρακάτω.

Η δεύτερη παραμετρική ανάλυση που θα γίνει για την πρώτη αυτή εφαρμογή είναι η τοπολογική βελτιστοποίηση με τις ίδιες ακριβώς αρχικές συνθήκες, με την μοναδική διαφορά όμως πως το πλέγμα πεπερασμένων στοιχείων θα είναι αρκετά πιο αραιό, θα αποτελείται δηλαδή από 908 εξακομβικά τριγωνικά πεπερασμένα. Δεν θα κοιτάξουμε τόσο

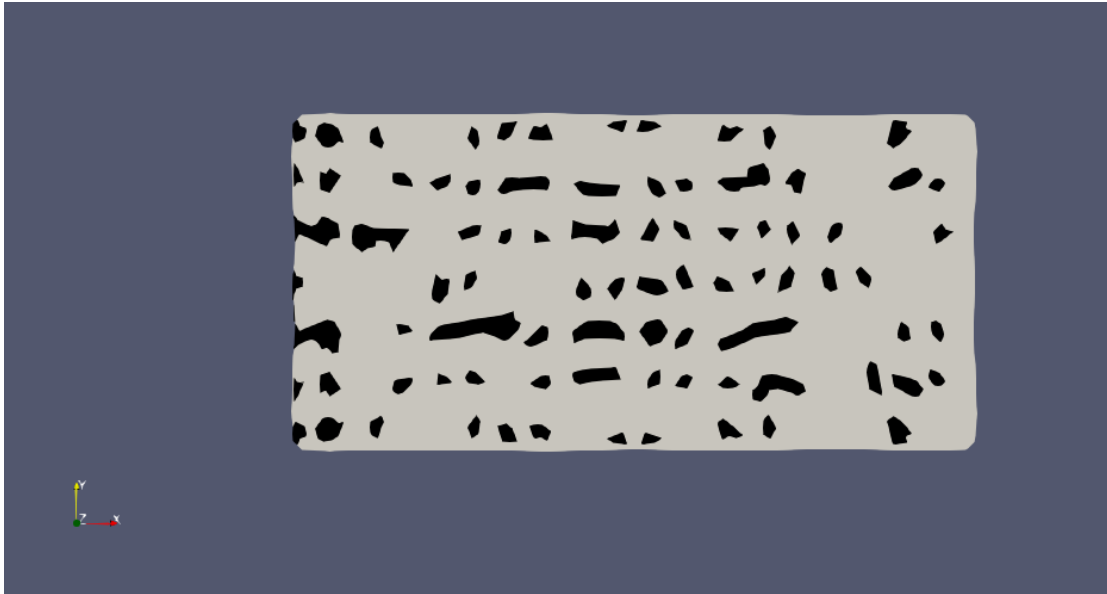
τα συγκριτικά αποτελέσματα ελαστικής παραμόρφωσης όπως έγινε για την κυρίως εφαρμογή, αλλά θα επικεντρωθούμε περισσότερο στην επίδραση που έχει η αραίωση της διακριτοποίησης στις μορφές που συγκλίνουν οι μέθοδοι, μορφές τις οποίες βλέπουμε στις τρεις παρακάτω εικόνες.



Εικόνα 4.10: Φορέας SIMP μεθόδου για την 1^η εφαρμογή με αραιό πλέγμα



Εικόνα 4.1:: Φορέας BESO μεθόδου για την 1^η εφαρμογή με αραιό πλέγμα

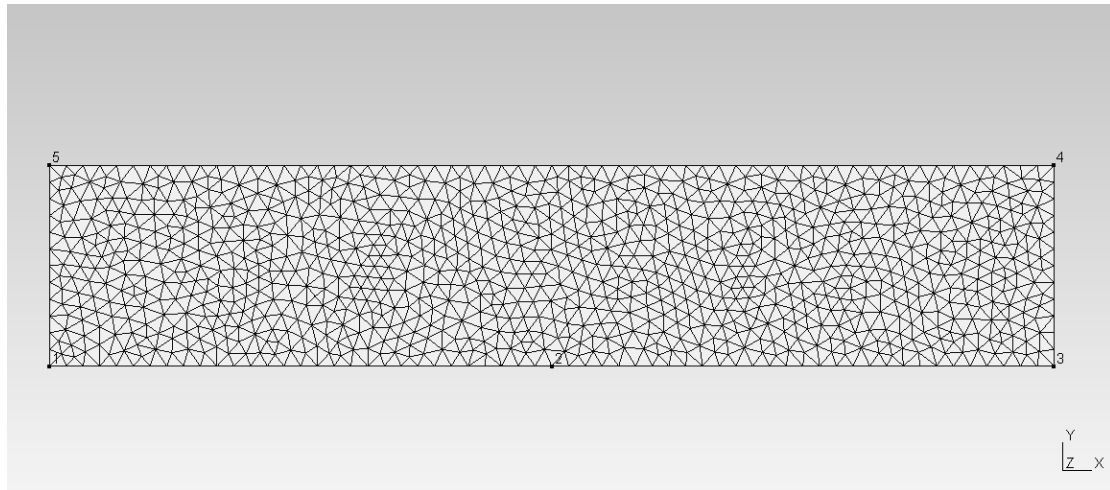


Εικόνα 4.12: Φορέας Level Set μεθόδου για την 1^η εφαρμογή με αραιό πλέγμα

Βλέπουμε εντελώς διαφορετικές εικόνες για τους φορείς της κάθε μεθόδου σε σύγκριση με αυτούς που συναντήσαμε στην εφαρμογή του ίδιου προβλήματος με πυκνότερα διακριτοποιημένο πλέγμα. Ακόμα και για την SIMP μέθοδο η οποία γενικώς δεν παρουσιάζει μεγάλη εξάρτηση από την διακριτοποίηση βλέπουμε πως έχει καταλήξει σε διαφορετική μορφή από αυτήν που κατέληξε παραπάνω. Όσον αφορά την BESO μέθοδο η διαφορετικότητα ήταν αναμενόμενη, αφού όπως είπαμε η διακριτοποίηση παίζει μεγάλο ρόλο για αυτήν, οδηγώντας την μάλιστα στην μοναδική μορφή που συναντήσαμε στην παρούσα εφαρμογή όπου υπάρχουν δυο εφελκυσμένα μέλη για την στήριξη του θλιβόμενου ενός που παραλαμβάνει το φορτίο. Τέλος η Level Set όπως ακριβώς περιμέναμε δεν μπόρεσε να συγκλίνει σε κάποια λύση, και μάλιστα η απόκλιση της έγινε ολοφάνερη από την δεύτερη ήδη επανάληψη του προβλήματος όπου και παρουσίασε τεράστια απότομη μεταβολή του ποσοστού όγκου που διέθετε, πέφτοντας μάλιστα κάτω από το 0.2. Αυτό αποτελεί απόδειξη πως η Level Set είναι ευθέως εξαρτώμενη από την διακριτοποίηση του πλέγματος τόσο ως προς την μορφή της προκύπτουσας αλλά και ως προς την ικανότητα εύρεσης οποιασδήποτε αξιόλογης μορφής βέλτιστου φορέα.

4.3 Εφαρμογή 2^η : Βελτιστοποίηση Τοπολογίας Γέφυρας

Ο επόμενος φορέας του οποίου επιχειρείται η τοπολογική βελτιστοποίηση είναι ένας φορέας μορφής γέφυρας. Για την ακρίβεια πρόκειται για μια αμφιαρθρωτή πλάκα με ένα φορτίο στο μέσον της, το οποίο ασκείται στο κάτω σύνορό της. Η γεωμετρία που τροφοδοτείται στο πρόβλημα μαζί με την διακριτοποίηση της, όπως προέκυψε με την χρήση του GMSH, είναι αυτή της παρακάτω εικόνας:



Εικόνα 4.13: Διακριτοποίηση φορέα τύπου γέφυρας 2^{ης} εφαρμογής

Το πλέγμα αποτελείται από 2031 τριγωνικά εξακομβικά πεπερασμένα στοιχεία. Ο κόμβος 2 έχει δεσμευτεί ως άρθρωση ενώ ο κόμβος 4 αποτελεί κύλιση με ελεύθερη την κατά x κίνηση. Το μοναδιαίο φορτίο ασκείται στην θέση του σημείου 3, στην κατά y διεύθυνση και φορά προς τα αρνητικά του άξονα. Η πλάκα έχει διαστάσεις 7.5×1.5 , υπάρχει δηλαδή ένας λόγος ύψους προς μήκος της γέφυρας ίσος με ένα προς πέντε. Το μέγεθος των στοιχείων επιλέχθηκε ίσο με 0.08, και επομένως το πάχος αυτών επιλέχθηκε ίσο με 0.02 ώστε να τηρείται και πάλι έστω στο περίπου η αναλογία που ειπώθηκε παραπάνω. Ο απαιτούμενος όγκος έχει ορισθεί ίσος με 0.5, ενώ η χαρακτηριστική μετατόπιση για τον φορά είναι η κατακόρυφη μετατόπιση του κόμβου 3, δηλαδή η μετατόπιση στην θέση όπου ασκείται το μοναδιαίο φορτίο.

Η πρώτη μέθοδος με την οποία έγινε προσπάθεια βελτιστοποίησης του φορέα είναι και πάλι η SIMP. Οι επαναλήψεις περιορίζονται και πάλι στις 100, μιας και αν αφήσουμε ελεύθερο τον αλγόριθμο ώσπου να συγκλίνει θα έχουμε αρκετές παραπάνω, χωρίς όμως κάποια ιδιαίτερη ουσία μιας και η μέθοδος έχει σχεδόν καταλήξει στον βέλτιστό της φορέα.



Εικόνα 4.14: Φορέας SIMP μεθόδου για την 2^η εφαρμογή

Βλέπουμε μια μορφή φορέα που θυμίζει σε μεγάλο βαθμό τις γεωμετρίες που συναντά κανείς σε γέφυρες, αποτελώντας μια μίξη γεφυρών τύπου ελκυστήρα τόξου και δικτυωτής.

Τα αποτελέσματα της SIMP μεθόδου για την βελτιστοποίηση της γέφυρας είναι

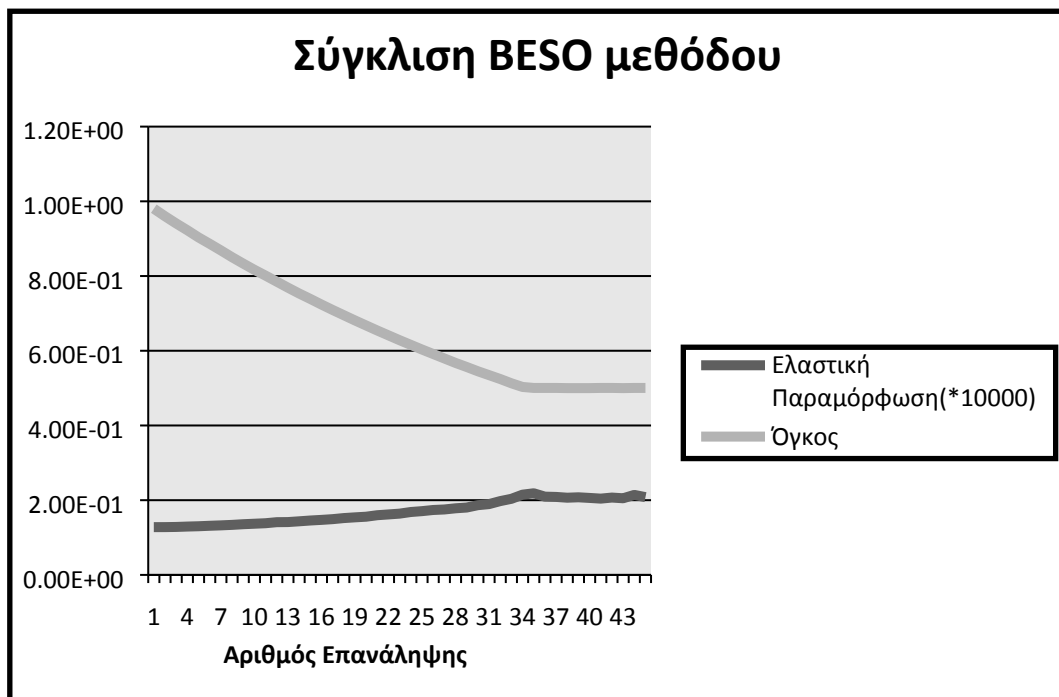
Επανάληψη: 100

Όγκος: 0.500000005530782

Ελαστική Παραμόρφωση: 2.22865692183502E-05

Χαρακτηριστική μετατόπιση: -2.23599837551807E-05

Ακολουθεί η βελτιστοποίηση με την μέθοδο BESO, για την οποία επετεύχθη και εδώ παρόμοιας μορφής φορέας με αυτήν της SIMP μεθόδου, αποδεικνύοντας για άλλη μια φορά την ομοιότητα μεταξύ των δυο. Πρώτα όμως παρουσιάζεται και πάλι σε διάγραμμα όγκος και η ελαστική παραμόρφωση που προέκυπτε σε κάθε επανάληψη της μεθόδου, ώστε να αποφασιστεί ποιά είναι εκείνη που θα μας δώσει την βέλτιστη λύση, καθώς και από ποιά επανάληψη ο κώδικας αρχίζει να συγκλίνει.



Διάγραμμα 4.6

Βλέπουμε λοιπόν πως από τις επαναλήψεις από τις οποίες έχουμε τον απαιτούμενο όγκο, αυτή που παρουσιάζει την ελάχιστη ελαστική παραμόρφωση είναι η τεσσαρακοστή πρώτη, η γεωμετρία της οποίας είναι η παρακάτω.



Εικόνα 4.15: Φορέας BESO μεθόδου για την 2^η εφαρμογή με αραιό πλέγμα

Όπως αναφέρεται και παραπάνω λοιπόν, καταλήγουμε σε μια μορφή φορέα αρκετά παρόμοια με την αντίστοιχη της μεθόδου SIMP.

Τα αποτελέσματα του παραπάνω φορέα της BESO μεθόδου είναι:

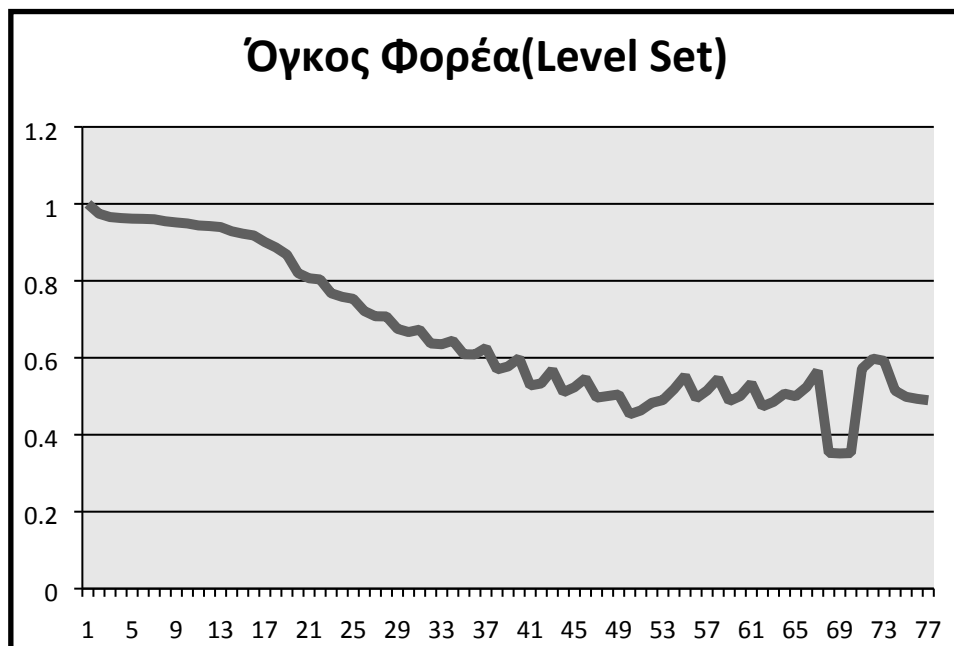
Επανάληψη: 41

Όγκος: 0.500499999999991

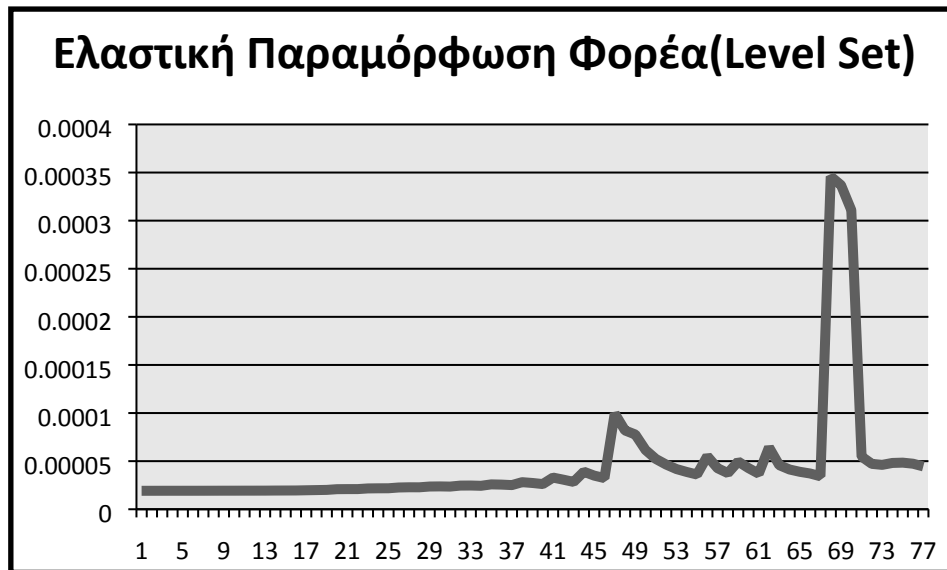
Ελαστική Παραμόρφωση: 2.04199667173826E-05

Χαρακτηριστική μετατόπιση: -2.04199667173826E-05

Προχωρούμε λοιπόν στην μέθοδο Level Set για την εύρεση της βέλτιστης μορφής του φορέα. Χαράσσουμε καταρχάς και πάλι τα διαγράμματα που θα βοηθήσουν στην απόφαση περί του ποιά είναι η καλύτερη εκ των λύσεων που προτείνονται από την Level Set μέθοδο. Επειδή και πάλι παρουσιάζονται έντονα άλματα δεν είναι δυνατόν τα χαραχτούν στο ίδιο γράφημα, για άλλη μια φορά λοιπόν χαράσσονται δυο ξεχωριστά γραφήματα.



Διάγραμμα 4.7



Διάγραμμα 4.8

Όπως βλέπουμε από τα παραπάνω διαγράμματα, η βέλτιστη λύση επιτυγχάνεται για την τεσσαρακοστή τέταρτη επανάληψη, η οποία για άλλη μια φορά βρίσκεται ακριβώς πριν από άλμα της μεθόδου στην επόμενη. Η μορφή της φαίνεται στην παρακάτω εικόνα.

Εικόνα 4.16: Φορέας Level Set μεθόδου 44^{ης} επανάληψης για την 2^η εφαρμογή

Παρόλο που τα αποτελέσματά της δείχνουν να είναι τα καλύτερα δυνατά που μπορεί η μέθοδος να προσφέρει, δεν είναι δυνατόν να επιλεγεί η συγκεκριμένη γεωμετρία για την κατασκευή του φορέα. Κι αυτό για τον προφανή λόγο της εμφάνισης δυο κομμένων μελών στο φορέα, γεγονός που δεν επιτρέπει την αξιοποίηση αυτού ως σοβαρού αποτελέσματος. Το φαινόμενο αυτό δημιουργείται επειδή η Level Set όπως έχει εξηγηθεί και παραπάνω δε μπορεί να συντηρήσει γεωμετρίες φορέων με πολλά μέλη, τρανταχτή απόδειξη του οποίου είναι ακριβώς αυτό το παράδειγμα. Δεν είναι ότι δε μπορεί να φτάσει σε τέτοιες γεωμετρίες, αλλά το ότι δε μπορεί να τις συντηρήσει και να προχωρήσει επ' αυτών.

Επομένως δε μπορεί να είναι αυτή η λύση που θα πάρουμε από την Level Set, και αναγκαστικά θα πρέπει να προχωρήσουμε σε μεταγενέστερες επαναλήψεις για να επιλέξουμε μια καλύτερη λύση, με την απαραίτητη όμως προσοχή καθώς έχουμε την απότομη μεταβολή της ελαστικής παραμόρφωσης στην αμέσως επόμενη επανάληψη.

Η επόμενη επανάληψη που παρουσιάζει ικανοποιητική ελαστική παραμόρφωση για όγκο κοντά στο απαιτούμενο ποσοστό είναι η εξηκοστή πέμπτη επανάληψη, την γεωμετρία της οποίας βλέπουμε στην επόμενη εικόνα.



Εικόνα 4.16: Φορέας Level Set μεθόδου 65^{ης} επανάληψης για την 2^η εφαρμογή

Η λύση αυτή είναι αρκετά πιο απότομη και 'άγαρμπη' από τις λύσεις των άλλων δυο μεθόδων καθώς και από την λύση που σχεδόν συνέκλινε η Level Set. Παρόλα αυτά είναι υλοποιήσιμη, οπότε και αυτήν θα θεωρήσουμε ως αποτέλεσμα της Level Set. Τα χαρακτηριστικά της λύσης αυτής είναι

Επανάληψη: 65

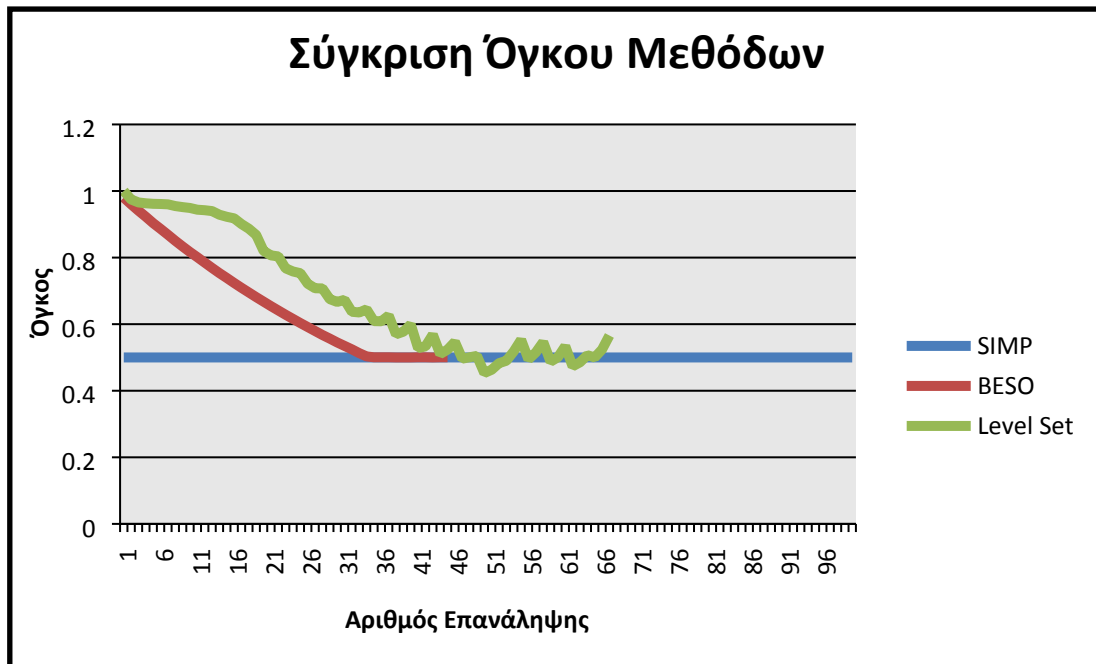
Όγκος: 0.5

Ελαστική Παραμόρφωση: 3.89089E-05

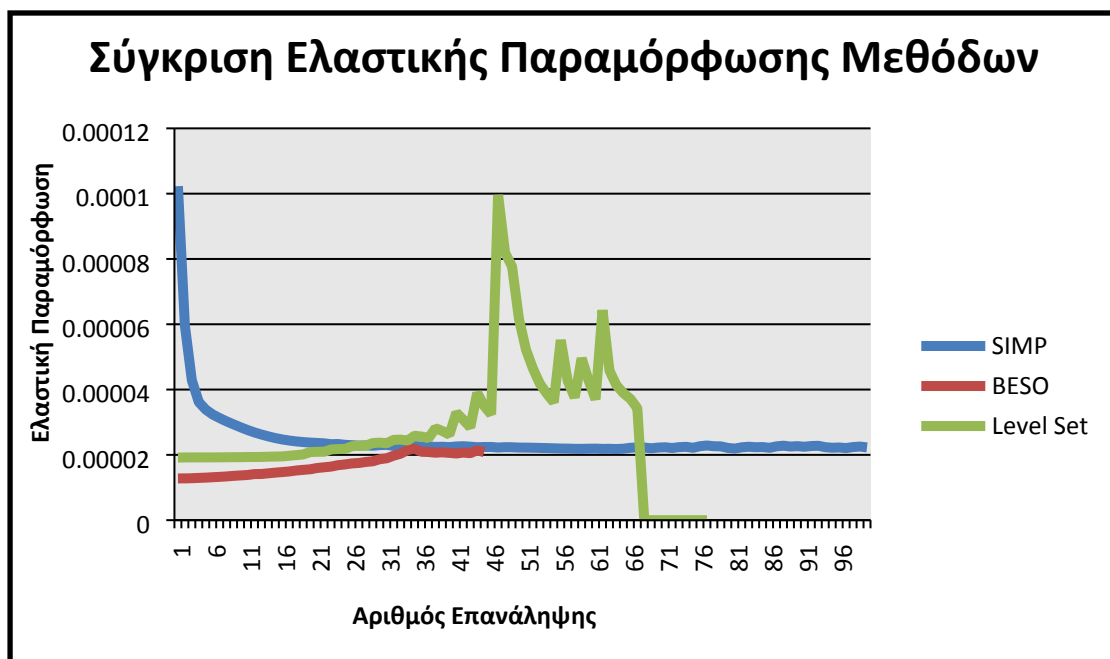
Χαρακτηριστική μετατόπιση : -2.59637346297666E-05

Παρόλο που βλέπουμε έναν όχι και τόσο βέλτιστο φορέα, ο οποίος πετυχαίνει χειρότερα αποτελέσματα όταν βρίσκεται σε σύγκριση με τις άλλες δυο μεθόδους, ένα αρκετά σημαντικό συμπέρασμα εξάγεται από αυτήν την εφαρμογή. Η εμφάνιση άλματος στην ελαστική παραμόρφωση κατά το πέρασμα από επανάληψη σε επανάληψη δεν σημαίνει αυτόματα μη σύγκλιση της μεθόδου. Αποδεικνύεται στην ουσία αυτό που ειπώθηκε και παραπάνω, πως αν κοπούν απότομα μερικά μέλη του φορέα αλλά υπάρχουν ακόμα αρκετοί δρόμοι για να καταλήξει το φορτίο στις στηρίξεις, ακριβώς όπως στην γεωμετρία της επαναλήψεως 44, η μέθοδος είναι δυνατόν να ανακάμψει και να μπορέσει να συγκλίνει σε διαφορετική λύση.

Τα συγκριτικά διαγράμματα όγκου και ελαστικής παραμόρφωσης για τις τρεις μεθόδους, όπου για την Level Set έχουν παραληφθεί όλα μετά το δεύτερο και αρκετό πιο απότομο άλμα της, είναι τα εξής:



Διάγραμμα 4.9

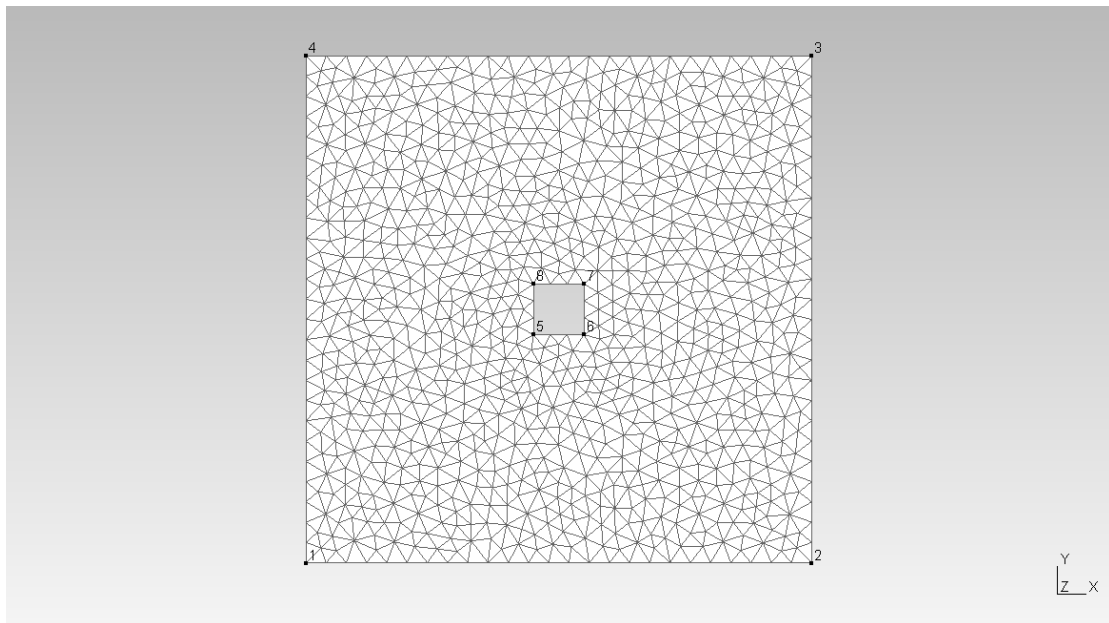


Διάγραμμα 4.10

Το πρόβλημα αυτό ήταν λιγάκι καλύτερο για να ξεχωρίσουμε τις μεθόδους, όπως εύκολα παρατηρεί κανείς από τα αποτελέσματα. Η Level Set εδώ δεν κατάφερε να πετύχει εξίσου καλά αποτελέσματα σε σχέση με τις άλλες δυο μεθόδους, ενώ η BESO ξεχωρίζει έχοντας βρει τον φορέα με την μικρότερη ελαστική παραμόρφωση.

4.4 Εφαρμογή 3^η : Βελτιστοποίηση τετραγωνικής πλάκας εσωτερικά πακτωμένης

Η επόμενη γεωμετρία της οποίας η τοπολογική βελτιστοποίηση θα επιχειρηθεί είναι αρκετά πιο ενδιαφέρουσα, τόσο ως προς τις συνοριακές τις συνθήκες όσο και προς τις μορφές φορέων που προτείνονται ως βέλτιστες. Πρόκειται περί μιας τετράγωνης πλάκας με ένα εσωτερικό κενό, στο όριο του οποίου θεωρείται πακτωμένη. Η πλάκα φορτίζεται με τέσσερα σημειακά φορτία, ένα σε κάθε της γωνία, οι φορές των οποίων είναι τέτοιες ώστε να δημιουργείται μια αντιωρολογιακή περιστροφή αυτής περί την στήριξη της, η οποία όπως είπαμε βρίσκεται στο κενό στο μέσον της. Η διακριτοποιημένη γεωμετρία της πλάκας έπειτα όπως εξάγεται από το GMSH είναι αυτή της παρακάτω εικόνας.

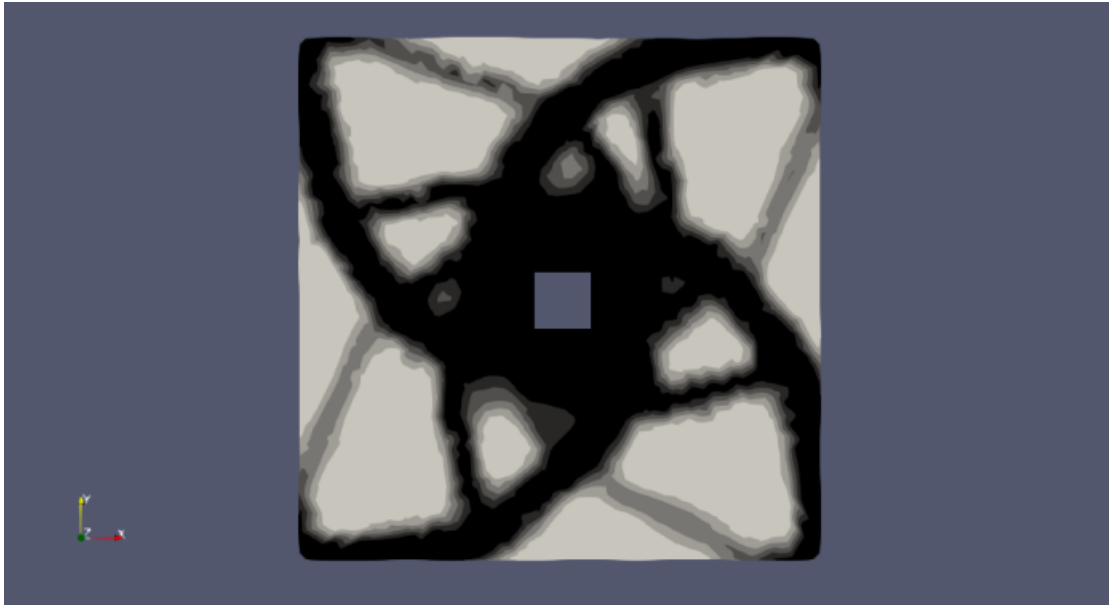


Εικόνα 4.17: Διακριτοποιημένος φορέας τετραγωνικής πλάκας εσωτερικά πακτωμένης

Το πλέγμα αποτελείται από 1993 τριγωνικά εξακομβικά πεπερασμένα στοιχεία. Οι κόμβοι κατά μήκος των πλευρών 5-6, 6-, 7-8 και 8-5 έχουν δεσμευμένους όλους τους βαθμούς ελευθερίας τους ώστε να μπορεί να θεωρηθεί η πλάκα πακτωμένη στο εσωτερικό της άνοιγμα. Από πλευράς φόρτισης εφαρμόζονται τέσσερα μοναδιαία φορτία στις τέσσερις γωνίες του φορέα, ένα στον κόμβο 1 κατά x θετικά, ένα στον κόμβο 2 κατά y θετικά, ένα στον κόμβο 3 κατά x αρνητικά και τέλος ένα στον κόμβο 4 κατά y αρνητικά. Το μέγεθος των στοιχείων επιλέχθηκε και πάλι ίσο με 0.08, οπότε και το πάχος αυτών επιλέχθηκε ίσο με 0.02.

Ο απαιτούμενος όγκος για τις λύσεις δίνεται στο πρόβλημα ίσος με 0.5 για άλλη μια φορά. Ως χαρακτηριστικές μετατοπίσεις τίθενται αυτές που αντιστοιχούν στους βαθμούς ελευθερίας κατά τους οποίους ασκείται το κάθε φορτίο.

Οι αναλύσεις ξεκινούν με την SIMP μέθοδο, η οποία εξαντλεί της 100 επαναλήψεις που τις διατίθενται μέχρι να καταλήξει στην μορφή που προτείνει. Παρακάτω παρατίθενται η γεωμετρία της λύσης της μεθόδου καθώς και τα αποτελέσματα αυτής.



Εικόνα 4.18: Φορέας SIMP μεθόδου για την 3^η εφαρμογή

Βλέπουμε μια πλήρως λειτουργική μορφή φορέα, καθώς η μέθοδος εκμεταλλεύεται όλο τον χώρο που της διατίθεται για να αναπτύξει την γεωμετρία της, σε αυτό το τόσο εντυπωσιακό αποτέλεσμα που δίνεται στην εικόνα. Τα αποτελέσματα της λύσης αυτής είναι :

Επανάληψη: 100

Όγκος: 0.5000000037234

Ελαστική Παραμόρφωση: 0.000149572600768093

Χαρακτηριστική μετατοπίσεις

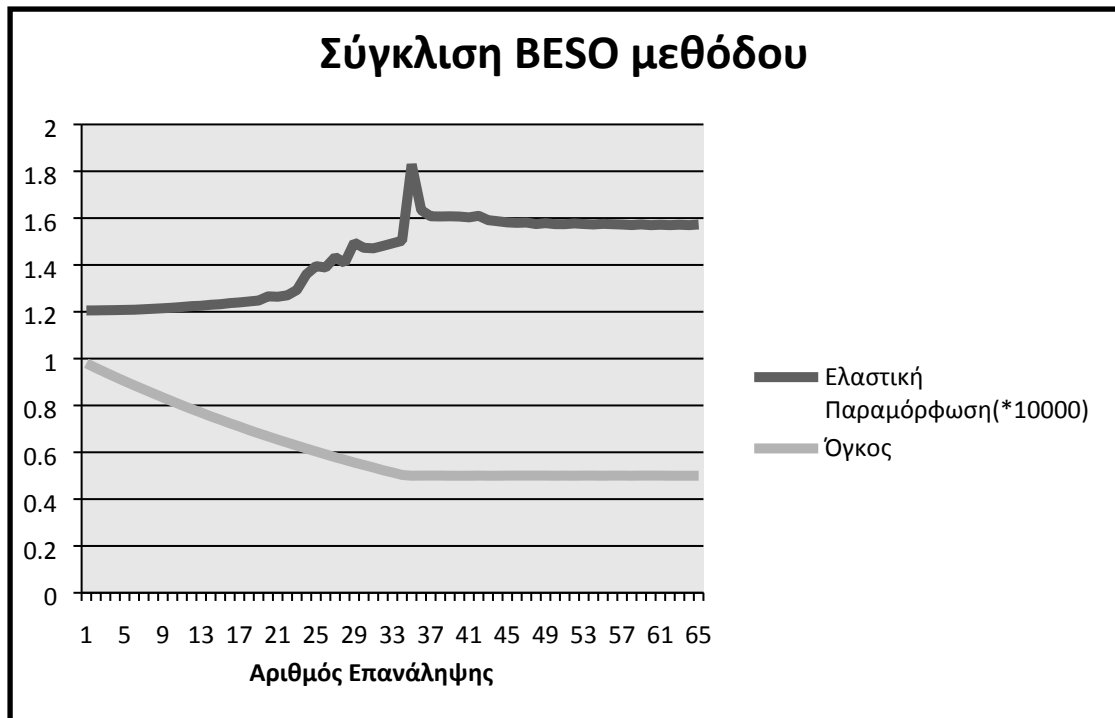
κατά x θετικά: 3.77925588802086E-05

κατά y θετικά: 3.76316715259359E-05

κατά x αρνητικά: -3.74769928488651E-05

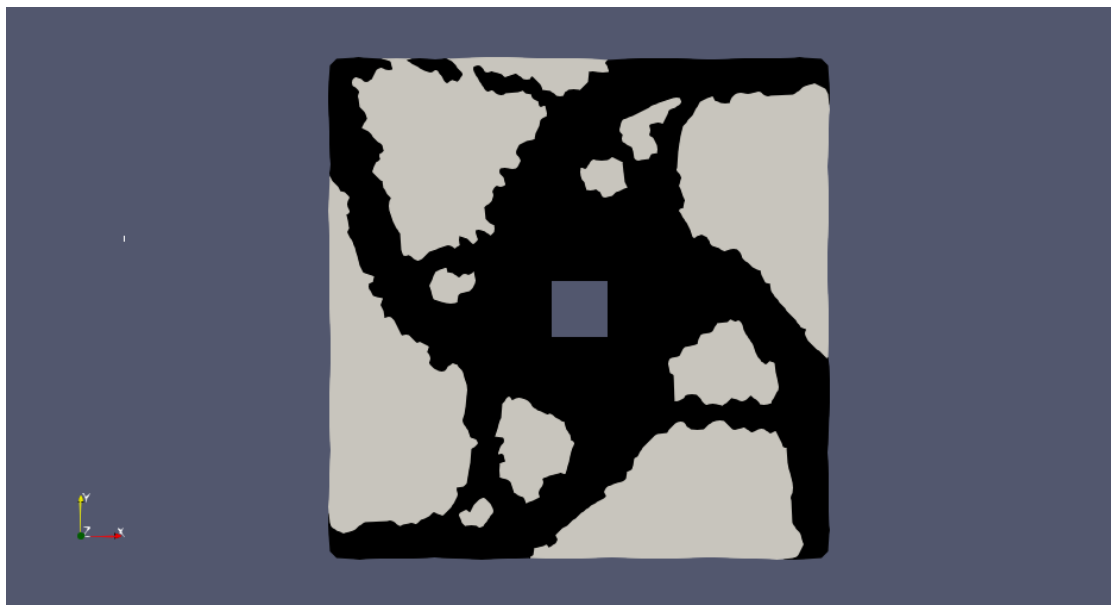
κατά y αρνητικά: -3.72196221327131E-05

Όπως και στις προηγούμενες περιπτώσεις, οι αναλύσεις συνεχίζονται με την BESO μέθοδο, για την οποία και πάλι προκειμένου να αποφασίσουμε την βέλτιστη μορφή φορέα που προτείνεται από την μέθοδο καταφεύγουμε στον σχηματισμό του παρακάτω διαγράμματος:



Διάγραμμα 4.11

Από το παραπάνω διάγραμμα φαίνεται πως η λύση με την ελάχιστη ελαστική παραμόρφωση εμφανίζεται για την τριακοστή τέταρτη επανάληψη, αμέσως μετά την οποία όμως παρουσιάζεται ένα άλμα που εγείρει αμφιβολίες καθώς δεν συναντώνται συχνά απότομες μεταβολές στην BESO, ενώ στην Level Set που τις είδαμε σήμαιναν την απώλεια κάποιου από τα μέλη του φορέα. Κι όντως οι αμφιβολίες αποδεικνύονται βάσιμες, προβάλλοντας την μορφή του φορέα κατά την επανάληψη αυτή στην παρακάτω εικόνα.

Εικόνα 4.19: Φορέας BESO μεθόδου 34ης επανάληψης για την 3^η εφαρμογή

Όπως βλέπουμε λοιπόν όντως ένα από τα μέλη του φορέα έχει κοπεί, οδηγώντας στην επόμενη στην απότομη μεταβολή που είδαμε στο διάγραμμα 4.11, συνεπώς δεν είναι δυνατόν να αποτελέσει αυτός την τοπολογικά βέλτιστη επιλογή για την κατασκευή. Προχωρώντας λοιπόν στις μετά το άλμα επαναλήψεις βλέπουμε πως η ελαστική παραμόρφωση συνεχώς μειώνεται, και καταλήγει να είναι η ελάχιστη στην τελευταία επανάληψη της μεθόδου, η οποία είναι η εξηκοστή πέμπτη επανάληψη και η μορφή του φορέα αυτής φαίνεται στην παρακάτω εικόνα.



Εικόνα 4.20: Φορέας BESO μεθόδου 65ης επανάληψης για την 3^η εφαρμογή

Η μέθοδος έχει καταφέρει να ανακάμψει πλήρως, και να καταλήξει σε μια βέλτιστη τοπολογία αρκετά παρόμοια αυτής της SIMP μεθόδου, με την διαφορά όμως πως της λείπουν τα δευτερεύοντα μέλη που παρουσιάζονται με χαμηλότερη πυκνότητα. Η BESO όμως θα τα ήθελε αυτά τα μέλη για την ανάπτυξη της βέλτιστης τοπολογίας της, όπως είδαμε από την λύση που στην οποία οριακά κόπηκε ένα από τα μέλη της, το οποίο ήταν στην ίδια θέση με αυτήν των δευτερευόντων μελών της SIMP μεθόδου. Το γεγονός αυτό μας προϊδεάζει για το ότι σε αυτήν την εφαρμογή η SIMP θα εμφανίσει καλύτερη συμπεριφορά από την BESO και θα καταλήξει σε χαμηλότερη ελαστική παραμόρφωση, μιας και η BESO έχει προσπαθήσει να ακολουθήσει την γεωμετρία της αλλά δεν έχει το απαραίτητο υλικό για να το κάνει. Τα αποτελέσματα της παραπάνω μορφής φορέα για την εξηκοστή πέμπτη επανάληψη της BESO είναι:

Επανάληψη: 65

Όγκος: 0.499943143812701

Ελαστική Παραμόρφωση: 0.000157291048572728

Χαρακτηριστική μετατοπίσεις

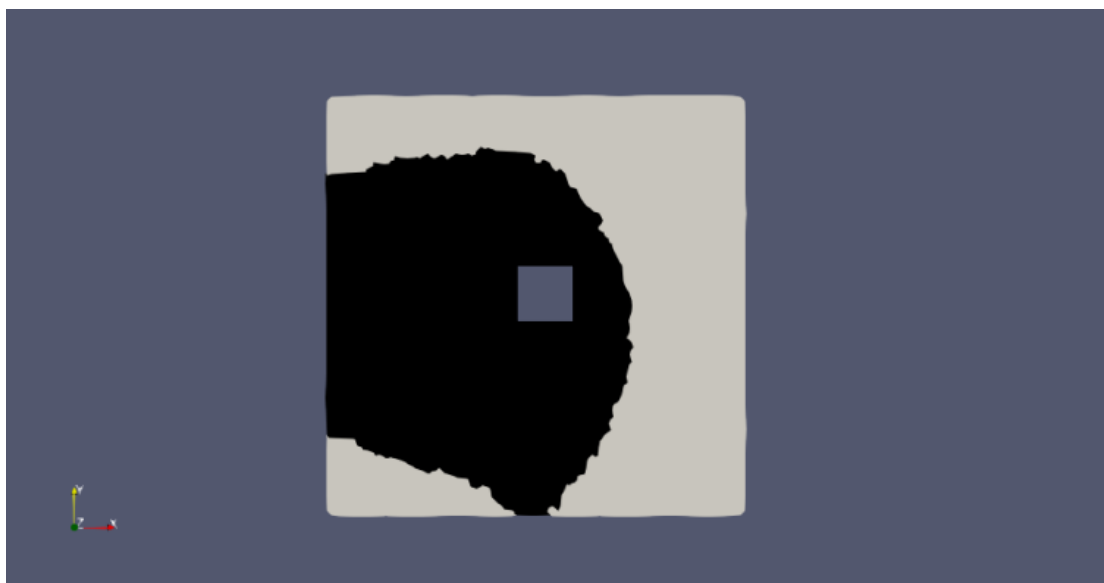
κατά x θετικά: 3.90908620551005E-05

κατά y θετικά: 3.8679835767863E-05

κατά x αρνητικά: -3.98542368807812E-05

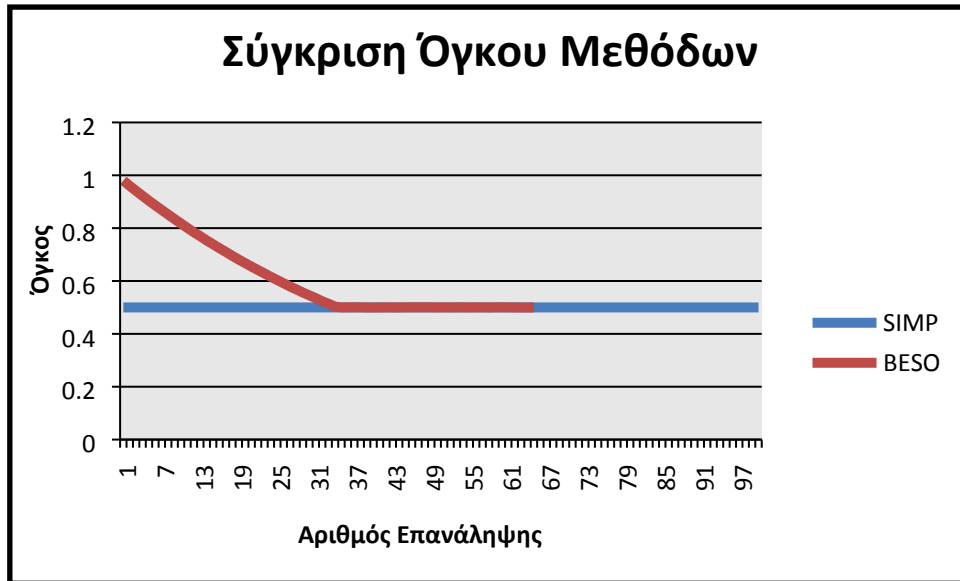
κατά y αρνητικά: -3.97772668273674E-05

Στην συνέχεια επιχειρείται η τοπολογική βελτιστοποίηση με την Level Set μέθοδο. Όπως είπαμε και πριν όμως, η Level Set δεν μπορεί να ακολουθήσει πιο σύνθετες γεωμετρίες με πολλά μέλη, για λόγους οι οποίοι θα εξηγηθούν ξανά και συγκεντρωτικά στο επόμενο κεφάλαιο. Αυτή η εφαρμογή λοιπόν θα αποτελέσει την πρώτη εφαρμογή κατά την οποία η Level Set δεν μπόρεσε να καταλήξει σε φορέα. Ο μόνος τρόπος με τον οποίο θα μπορούσε να ευρεθεί λύση για τον φορέα είναι η πύκνωση του πλέγματος πεπερασμένων στοιχείων, κάτι το οποίο θα έδινε την δυνατότητα στην επιφάνεια της Level Set να ακολουθήσει τις γεωμετρίες περισσότερων μελών που εμφανίζονται. Ενδεικτικά παρουσιάζεται ο καλύτερος φορέας που η μέθοδος μπορεί να βρει, με τον οποίο κίόλας γίνεται εμφανές πως τα αποτελέσματά της στην παρούσα εφαρμογή δεν μπορούν να ληφθούν υπόψη.

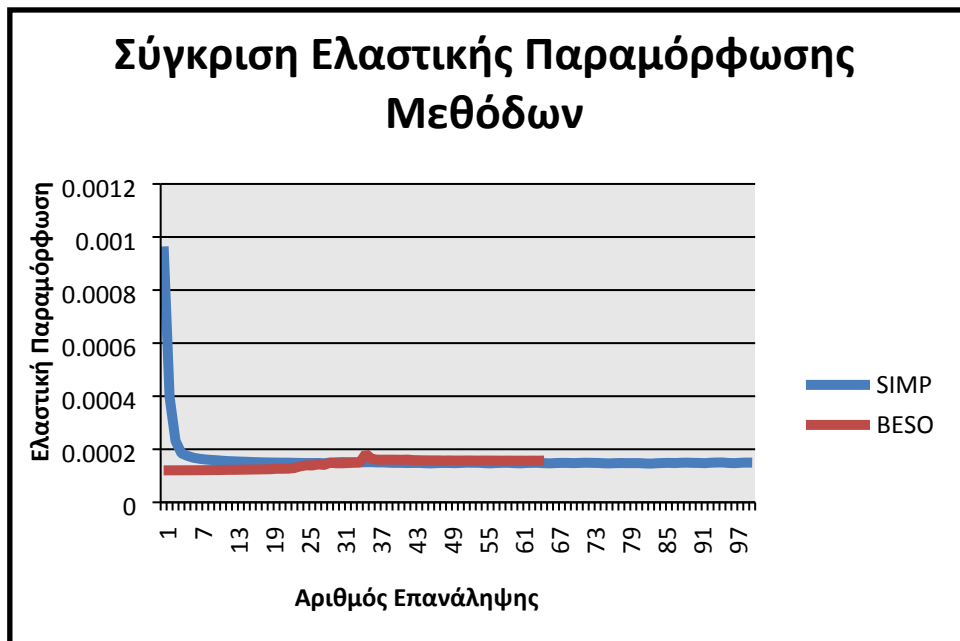


Εικόνα 4.21: Φορέας Level Set μεθόδου για την 3^η εφαρμογή

Η σύγκριση λοιπόν εφαρμόζεται μονάχα μεταξύ των μεθόδων SIMP και BESO, καθώς τα αποτελέσματα της Level απλώς θα κατέστρεφαν την κλίμακα του διαγράμματος χωρίς να προσφέρουν κάτι στην σύγκριση. Έτσι, τα συγκριτικά διαγράμματα όγκου και ελαστικής παραμόρφωσης θα είναι:



Διάγραμμα 4.12

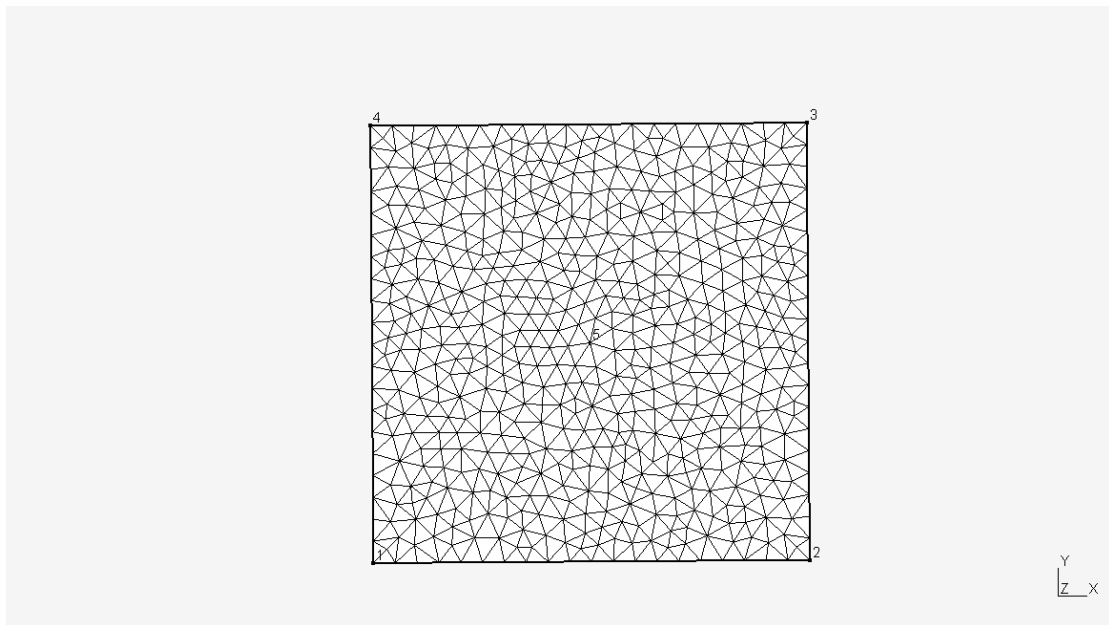


Διάγραμμα 4.13

Παρατηρούμε λοιπόν πως όντως η BESO μέθοδος συγκλίνει σε μορφή για την οποία η ελαστική παραμόρφωση είναι μεγαλύτερη από αυτήν της SIMP μεθόδου. Αυτό όμως δεν χρεώνεται απαραίτητα ως ανωτερότητα της SIMP στην εύρεση καλύτερης μορφής λύσεως, αλλά στην αδυναμία της BESO να κινηθεί στο υπάρχον πλέγμα, καθώς απαιτείται πυκνότερο πλέγμα ώστε να μπορέσει να αναπτύξει και να συντηρήσει όλα τα μέλη που θέλει χωρίς να κοπούν. Παρόλο λοιπόν που η SIMP αναδεικνύεται ως καλύτερη για την συγκεκριμένη εφαρμογή, σκεπτικισμός απαιτείται για την γενίκευση του αποτελέσματος αυτού.

4.5 Εφαρμογή 4^η : Βελτιστοποίηση πλάκας εγκάρσιως φορτισμένης

Η επόμενη εφαρμογή αποτελεί στην ουσία μια εισαγωγή για την εξέταση φορέων αποτελούμενων από στοιχεία πλακών που ζούνε στον χώρο, με την γεωμετρία και τις συνοριακές τους συνθήκες να μην περιορίζονται στις δυο διαστάσεις. Στις έως τώρα εφαρμογές γινόταν χρήση μονάχα της μεμβρανικής λειτουργίας του στοιχείου για τον καθορισμό της ακαμψίας αυτού, κάλλιστα λοιπόν θα μπορούσαν να εκτελεστούν και με την χρήση ενός πεπερασμένου στοιχείου επίπεδης έντασης παραμόρφωσης. Για να εκμεταλλευτούμε λοιπόν το πλήρες εύρος δυνατοτήτων που μας δίνει το τριγωνικό στοιχείο πλάκας που αναπτύξαμε θα εξετάσουμε και εφαρμογές στις οποίες θα συνεισφέρει στην ακαμψία του στοιχείου και η καμπτική και διατμητική συνιστώσα. Το μη δομημένο πλέγμα πεπερασμένων στοιχείων για την πλάκα όπως αυτό προέκυψε από το GMSH φαίνεται στην επόμενη εικόνα.

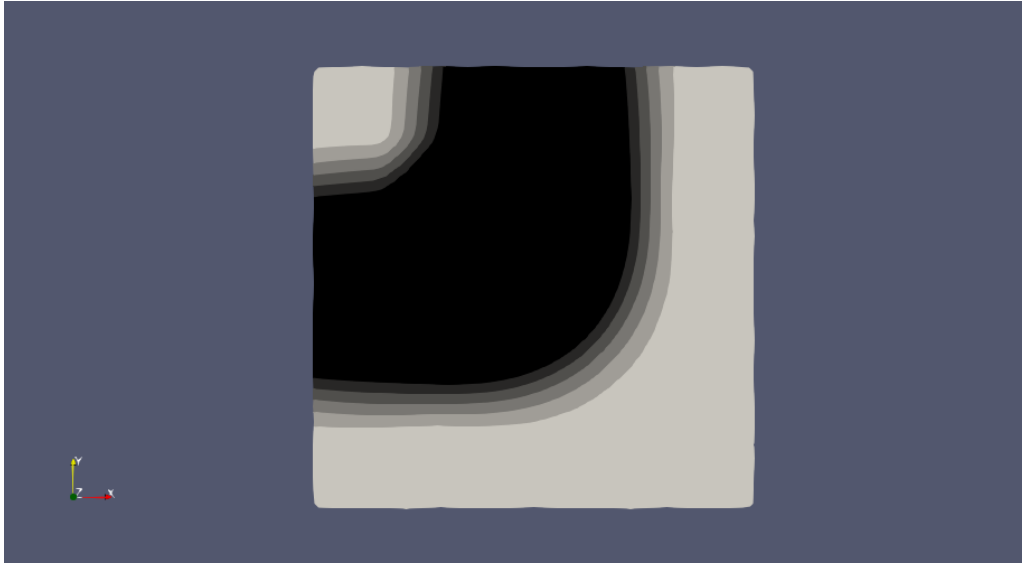


Εικόνα 4.22: Διακριτοποιημένος φορέας τετραγωνικής πλάκας 4^{ης} εφαρμογής

Πρόκειται για μια πλάκα διαστάσεων 2.0x2.0 η οποία είναι πακτωμένη σε όλες τις τις πλευρές και στο μέσον της(κόμβος 5) ασκείται ένα εγκάρσιο μοναδιαίο φορτίο. Για την διακριτοποίηση του πλέγματος έχει επιλεγεί μέγεθος στοιχείων ίσο με 0.08, το οποίο έχει οδηγήσει σε 2328 εξακομβικά στοιχεία.

Ο απαιτούμενος όγκος δίνεται για άλλη μια φορά ίσος με 0.5, ενώ ως χαρακτηριστική μετατόπιση ορίζεται η κατακόρυφη μετακίνηση του κόμβου 5, όπου δηλαδή ασκείται και το φορτίο.

Ο φορέας που προέκυψε από την εφαρμογή της SIMP μεθόδου στο πρόβλημα δίνεται στην παρακάτω εικόνα.



Εικόνα 4.23: Φορέας SIMP μεθόδου για την 4^η εφαρμογή

Η παραπάνω μορφή φορέα, η οποία λόγω της απλότητας του προβλήματος έχει ευρεθεί από την 58 ήδη επανάληψη αποτελώντας την μοναδική φορά που η SIMP μέθοδος συγκλίνει πριν τις 100 επαναλήψεις, δίνει τα εξής αποτελέσματα ελαστικής παραμόρφωσης:

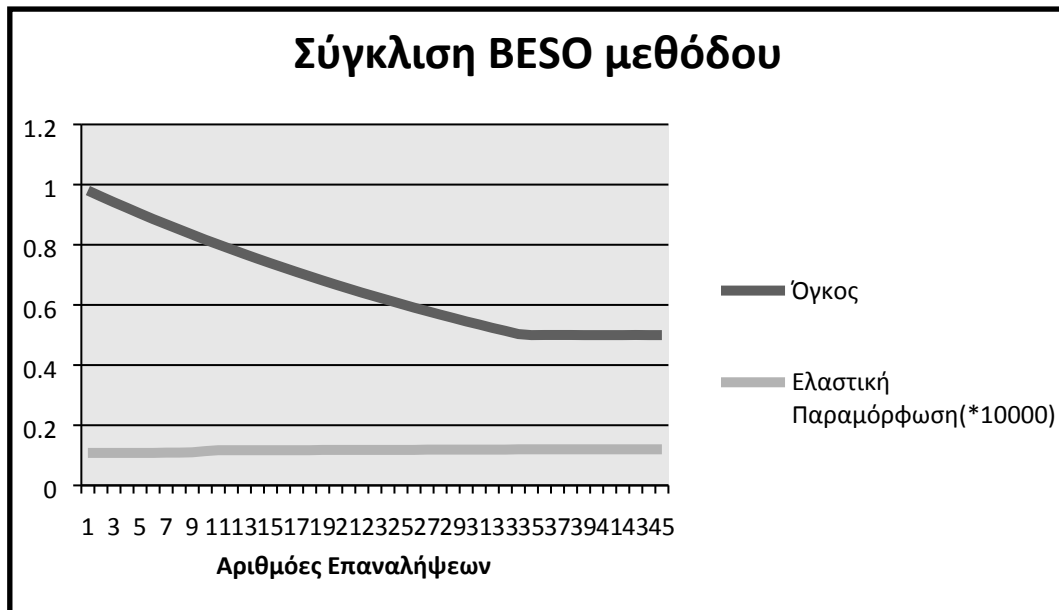
Επανάληψη: 58

Όγκος: 0,500000987634229

Ελαστική Παραμόρφωση: 2,40584120057008E-05

Χαρακτηριστική μετατόπιση: -2,40800421244351E-05

Στη συνέχεια εφαρμόστηκε η BESO μέθοδος η οποία μας δίνει την παρακάτω γεωμετρία βέλτιστου φορέα μέσω της εξέτασης όπως είπαμε του διαγράμματος όγκου-ελαστικής παραμόρφωσης των επαναλήψεων.



Διάγραμμα 4.14

Βλέπουμε από το παραπάνω διάγραμμα πως η BESO από την στιγμή που πετύχει των απαιτούμενο όγκο και πέρα διατηρεί σταθερή σε μεγάλο βαθμό και την ελαστική της παραμόρφωση, χωρίς να κάνει αλλαγές και απλώς να περιμένει να περάσουν οι επαναλήψεις ώστε να συγκλίνει τελικώς ο αλγόριθμος. Αυτό αποδεικνύει το ίδιο πράγμα που αναφέρθηκε στην SIMP ως προς την απλότητα του προβλήματος που τίθεται.

Εικόνα 4.23: Φορέας BESO μεθόδου για την 4^η εφαρμογή

Τα αποτελέσματα ελαστικής παραμόρφωσης που η BESO μέθοδος προτείνει θα είναι:

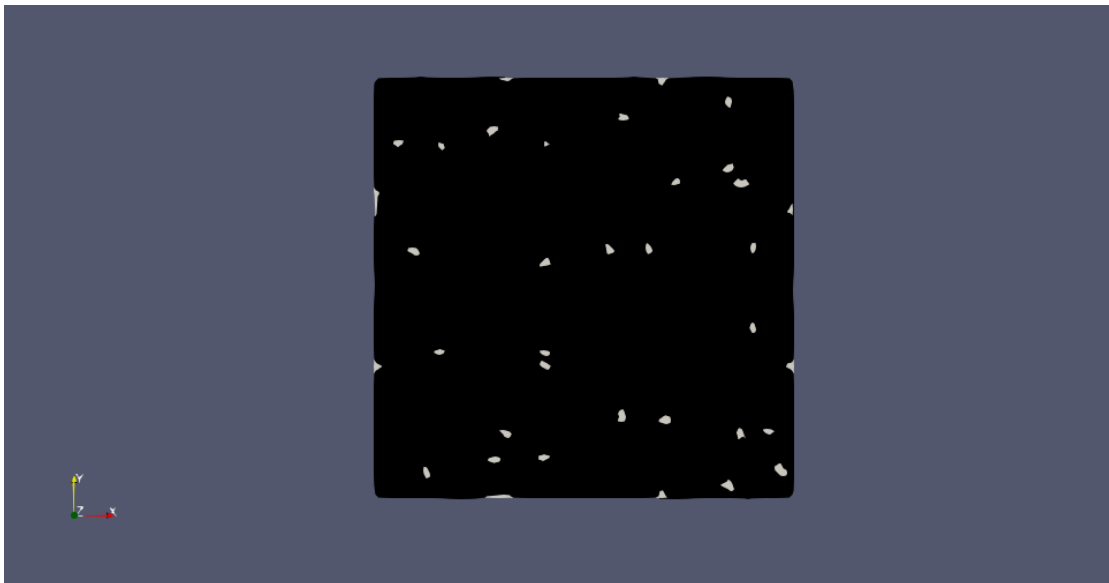
Επανάληψη: 44

Όγκος: 0,499641752577315

Ελαστική Παραμόρφωση: 1,19725255281057E-05

Χαρακτηριστική μετατόπιση: -2,39462394685486E-05

Τέλος, επιχειρήθηκε η βελτιστοποίηση με την Level Set μέθοδο για την εφαρμογή, κάτι που ανέδειξε το δεύτερο πρόβλημα στον αλγόριθμο αυτής.



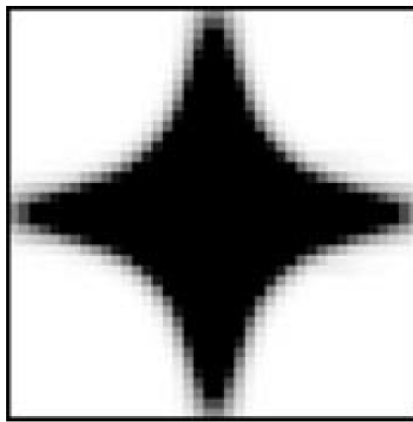
Εικόνα 4.24: Φορέας Level Set μεθόδου για την 4^η εφαρμογή

Από την παραπάνω εικόνα καταλαβαίνουμε πως για ακόμα μια φορά η Level Set μέθοδος δε μπορεί να καταλήξει σε κάποιας μορφής λύση. Το ερώτημα όμως που εγείρεται είναι το γιατί, καθώς οι άλλες δυο μέθοδοι βρήκαν πολύ γρήγορα τις βέλτιστες μορφές τους για την εφαρμογή αυτή. Η απάντηση λοιπόν βρίσκεται στον ίδιο τον αλγόριθμο της Level Set, ή τουλάχιστον στην αλγοριθμική υλοποίηση της Level Set από την V. J. Challis. Ο αλγόριθμος αυτός λοιπόν δεν είναι φτιαγμένος έτσι ώστε να είναι ανηγμένος ως προς τα ελαστικά χαρακτηριστικά του προβλήματος. Και ενώ αυτό μπορεί να λυθεί εύκολα για το μέτρο ελαστικότητας κατά Young πολλαπλασιάζοντας με αυτό τις προκύπτουσες μετατοπίσεις, δεν συμβαίνει το ίδιο και για το πάχος του στοιχείου. Στην περίπτωση της εντός επιπέδου έντασης-παραμόρφωσης των στοιχείων το πάχος μπορεί να ληφθεί υπόψη ως ακόμα ένας συντελεστής όπως είναι το μέτρο ελαστικότητας και να πολλαπλασιαστεί με τις μετατοπίσεις λύνοντας το πρόβλημα. Όμως στη δυσμενή περίπτωση που το στοιχείο αναπτύσσει και εκτός επιπέδου ένταση, συμμετέχουν δηλαδή στην ακαμψία του η καμπτική και διατμητική του λειτουργία που συναντήσαμε στο 2^ο Κεφάλαιο, ένας απλός πολλαπλασιασμός δε μπορεί να λύσει τα πράγματα. Κι αυτό γιατί το μητρώο ελαστικότητας που αντιστοιχεί στην καμπτική συμπεριφορά του στοιχείου περιέχει το πάχος εις την τρίτη, ενώ τα άλλα δυο μητρώα ελαστικότητας έχουν τον όρο του πάχους εις την πρώτη. Ετσι δεν είναι δυνατόν οι μετατοπίσεις να πολλαπλασιαστούν με έναν συντελεστή που θα τις φέρει

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

όλες στο ίδιο επίπεδο προκειμένου να μπορέσει η μέθοδος να λειτουργήσει. Γενικώς λοιπόν η Level Set υπό την παρούσα της αλγοριθμική υλοποίηση δε μπορεί να λειτουργήσει όταν εμφανίζονται εγκάρσιες στο επίπεδο του στοιχείου μετατοπίσεις.

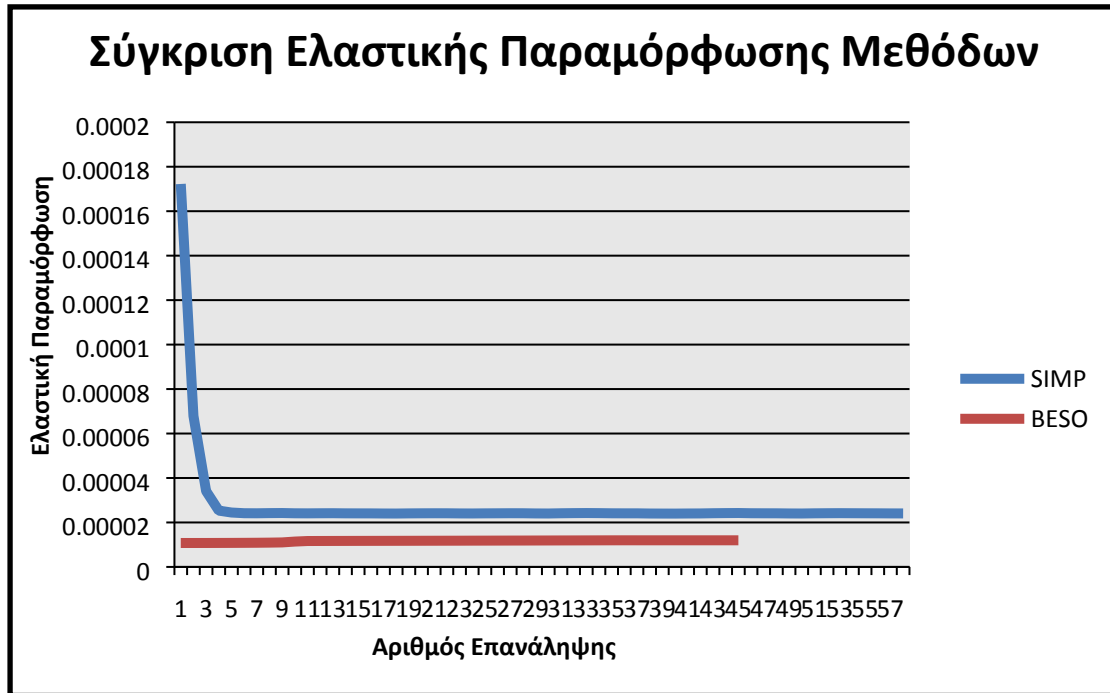
Η σύγκριση λοιπόν θα γίνει μονάχα για τις άλλες δυο μεθόδους και πάλι. Και οι δυο μέθοδοι συνέκλιναν γρήγορα στις λύσεις που προκύπτουν κάνοντας το πρόβλημα να φαίνεται εύκολο. Το θέμα είναι όμως πως και οι δυο μέθοδοι έκαναν λάθος στην βέλτιστη προκύπτουσα μορφή τους, καθώς το πρόβλημα αυτό έχει συναντηθεί σε δημοσίευση με την χρήση δομημένου πλέγματος και η μορφή που δίνεται είναι τελείως διαφορετική (η δημοσίευση αφορά μη γραμμικές αναλύσεις αλλά συμπεριλαμβάνει και μερικές γραμμικές του προβλήματος προς σύγκριση, τις οποίες και θα επικαλεστούμε). Πρόκειται για την δημοσίευση 'On topology optimization of linear and nonlinear plate problems' των B. Boroomand και A. R. Barekatein [2], η προκύπτουσα μορφή της οποίας είναι η παρακάτω:



Εικόνα 4.25: Φορέας SIMP μεθόδου για πρόβλημα παρόμοιο της 4^{ης} εφαρμογής αλλά σε δομημένο πλέγμα

Πρόκειται για αρκετά πιο λογική μορφή φορέα για το πρόβλημα της βελτιστοποίησης, κάτι που μας οδηγεί στο συμπέρασμα πως υπάρχει κάποιο σοβαρό εμπόδιο που ξεγέλασε τις μεθόδους και τις οδήγησε στην σύγκλιση μη βέλτιστης λύσης. Και αυτό το εμπόδιο δεν είναι άλλο από το ίδιο το πλέγμα πεπερασμένων στοιχείων, που όντας μη δομημένο οδήγησε τις μεθόδους προς τις δυο πλευρές, αυτές που τα στοιχεία του κέντρου και της γύρω περιοχής είχανε πιο κοντά τα κεντροβαρικά σημεία τους προς. Ίσως το πρόβλημα αυτό να μην εμφανιζόταν με πυκνότερη διακριτοποίηση, αλλά προς το παρόν οδηγεί στο συμπέρασμα πως το μη δομημένο του πλέγματος μπορεί να επηρεάσει σε μεγάλο βαθμό την προκύπτουσα λύση, και με βάση τα έως τώρα δεδομένα όχι προς το καλύτερο.

Χαράζεται λοιπόν μονάχα το διάγραμμα της ελαστικής παραμόρφωσης των δυο μεθόδων για την μετέπειτα χρήση του στην μεταξύ τους σύγκριση, αλλά σίγουρα όχι για την επιλογή του βέλτιστου φορέα.

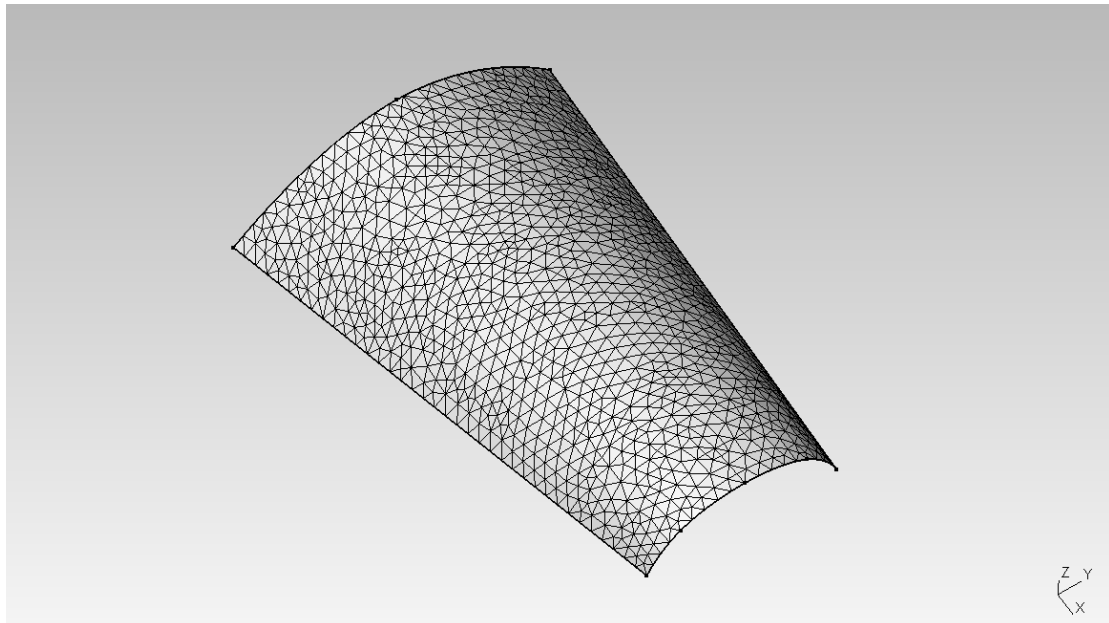


Διάγραμμα 4.15

Από το παραπάνω διάγραμμα βλέπουμε για άλλη μια φορά την ικανότητα της BESO μεθόδου να βρίσκει καλύτερες μορφές λύσεων από την σχεδόν όμοια της SIMP.

4.6 Εφαρμογή 5^η : Βελτιστοποίηση προβόλου μορφής κελύφους στο χώρο

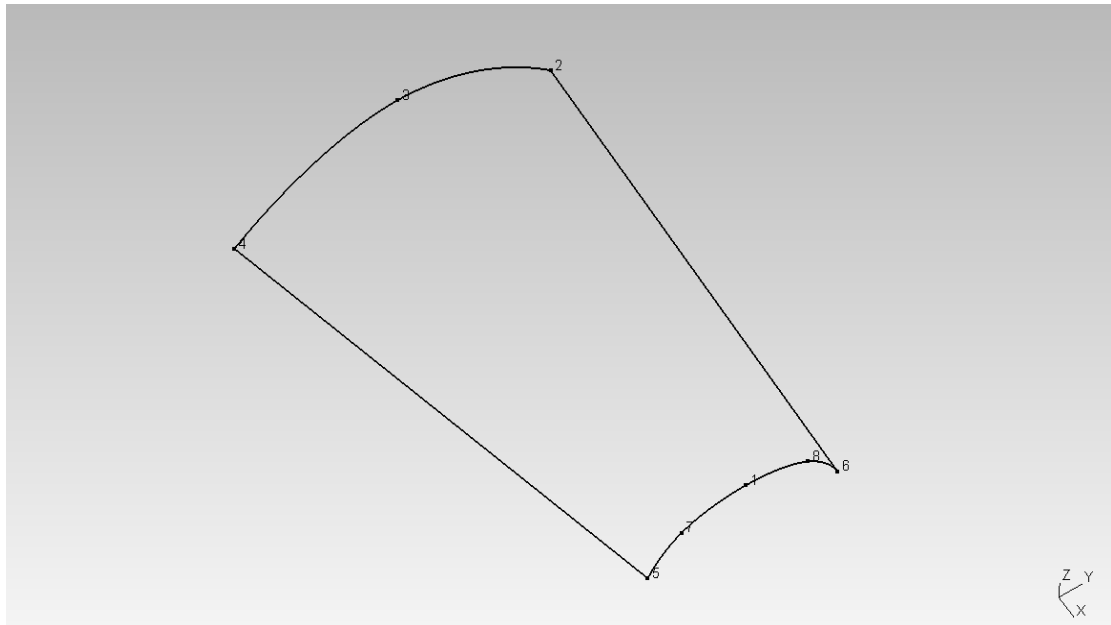
Η τελευταία εφαρμογή αφορά την τοπολογική βελτιστοποίηση ενός καμπυλωμένου προβόλου κελυφωτής μορφής στον χώρο. Πρόκειται για το πιο πρακτικό πρόβλημα απ όσα έχουν παρουσιαστεί έως τώρα, καθώς στέγαστρα παρόμοιας μορφής συναντώνται πολύ συχνά ως κατασκευαστικές λύσεις για την προστασία από καιρικά φαινόμενα. Βέβαια η φόρτιση στην οποία θα υποβληθεί είναι αρκετά απλοϊκή και δεν πλησιάζει τα καιρικά αυτά φαινόμενα, καθώς αποτελείται από ένα σημειακό φορτίο στην άκρη του προβόλου και όχι από ένα κατανεμημένο σε όλη του την επιφάνεια. Παρόλα αυτά ενδιαφέρον έχει να δούμε τις προκύπτουσες μορφές με την κάθε μέθοδο βελτιστοποίησης, έστω και αν δεν τύχουν πραγματικής υλοποίησης. Η γεωμετρία του φορέα διακριτοποιημένη μέσω του GMSH είναι αυτή της παρακάτω εικόνας.



Εικόνα 4.25: Διακριτοποιημένος φορέας προβόλου μορφής κελύφους στον χώρο

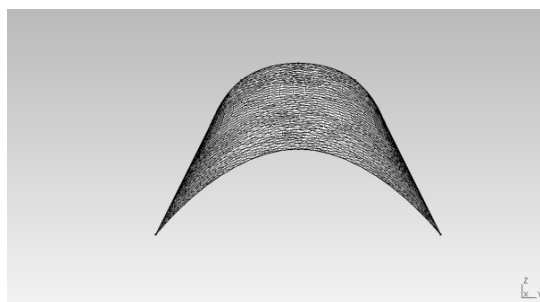
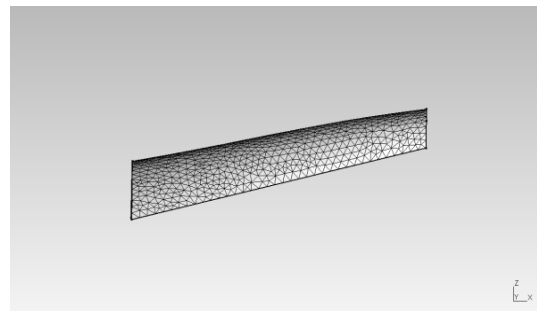
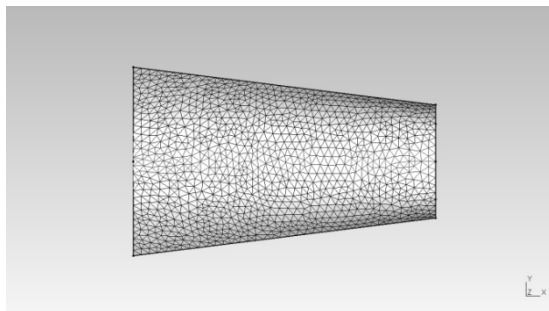
Το πλέγμα αποτελείται από 2144 τριγωνικά εξακομβικά πεπερασμένα στοιχεία. Επειδή κατά την εφαρμογή του πλέγματος χαλάει η αρίθμηση των κόμβων για το συγκεκριμένο παράδειγμα, μια ιδιοτροπία του GMSH, παρουσιάζεται ένα δεύτερο σχήμα(εικόνα 4.26) για την παρουσίαση των συνοριακών συνθηκών του φορέα. Κατά μήκος της πλευράς 2-3-4 οι κόμβοι έχουν δεσμευμένους όλου τους βαθμούς ελευθερίας του προκειμένου να υλοποιηθεί η πάκτωση της πλευράς. Το μοναδιαίο φορτίο ασκείται στην θέση του σημείου 1, στην κατά z διεύθυνση και φορά προς τα αρνητικά του άξονα. Το μέγεθος των στοιχείων επιλέχθηκε ίσο με 0.23, και επομένως το πάχος αυτών επιλέχθηκε ίσο με 0.05 για να τηρηθεί προσεγγιστικά η απαραίτητη αναλογία.

Ο απαιτούμενος όγκος έχει ορισθεί ίσος με 0.5, ενώ η χαρακτηριστική μετατόπιση για τον φορέα είναι η κατακόρυφη μετατόπιση του κόμβου 1, δηλαδή η μετατόπιση στην θέση όπου ασκείται το μοναδιαίο φορτίο.



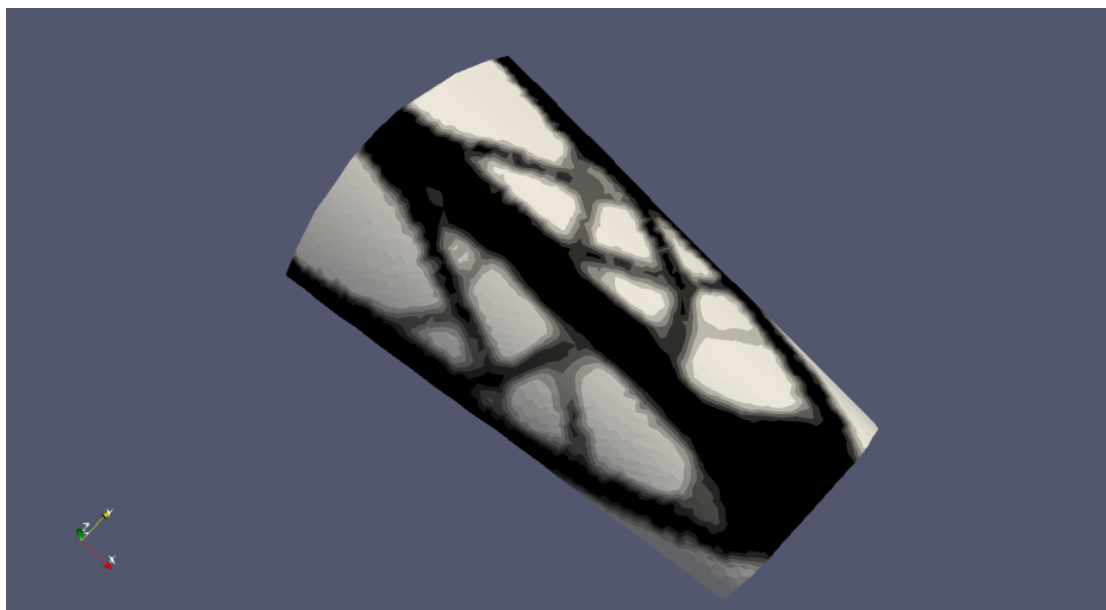
Εικόνα 4.26: Γεωμετρία φορέα προβόλου μορφής κελύφους στον χώρο

Επίσης παρατίθενται και τρεις όψεις του φορέα, για την καλύτερη κατανόηση της γεωμετρίας του στον χώρο.



Εικόνα 4.27: Όψεις του φορέα της 5^{ης} εφαρμογής

Οι αναλύσεις ξεκινάνε με την SIMP μέθοδο, στην οποία επιτρέπονται έως 100 επαναλήψεις για την εύρεση λύσης. Η μέθοδος για άλλη μια φορά εκμεταλλεύεται και τις 100, καταλήγοντας στην παρακάτω μορφή.



Εικόνα 4.28: Φορέας SIMP μεθόδου για την 5^η εφαρμογή

Βλέπουμε έναν έντονα αραχνοειδές φορέα, ο οποίος εκμεταλλεύεται όλο τον χώρο που του δίνεται για να αναπτυχθεί με τον καλύτερο για τα δεδομένα του τρόπο. Τα αποτελέσματα της μεθόδου είναι:

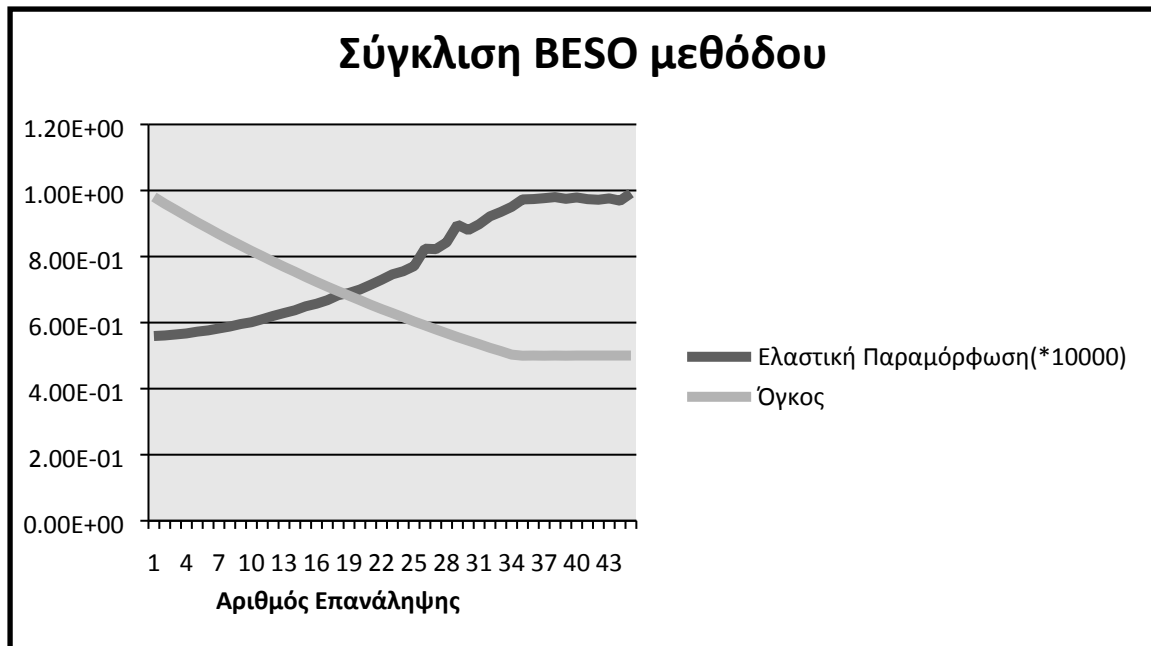
Επανάληψη: 100

Όγκος: 0.500000001393415

Ελαστική Παραμόρφωση: 0.000104895021351401

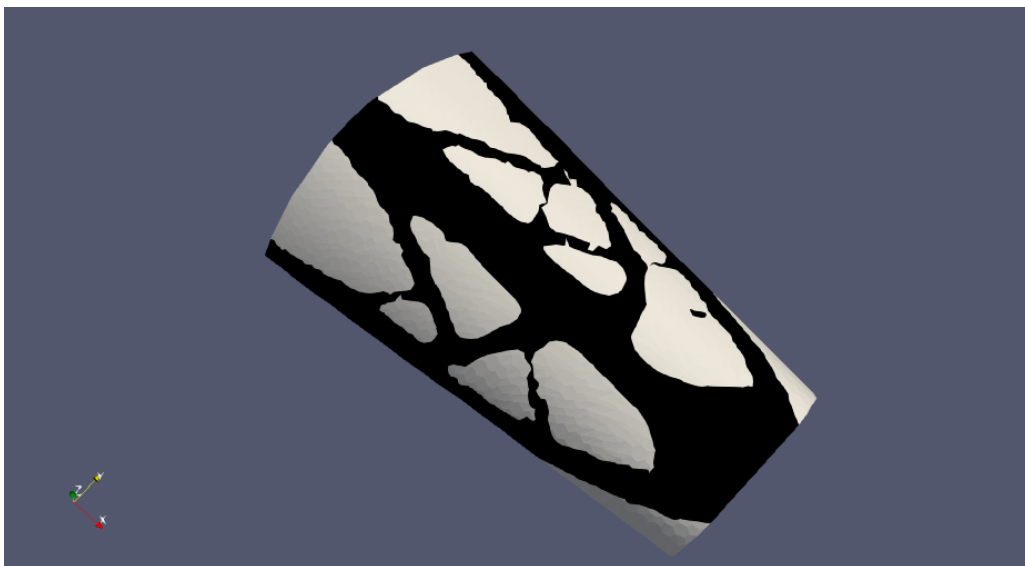
Χαρακτηριστική μετατόπιση: -0.00010565288020304

Πέραν της αρκετά εντυπωσιακής μορφής του φορέα, δεν υπάρχει κάτι άλλο να σχολιαστεί, οπότε περνάμε στην εφαρμογή της BESO μεθόδου, χαράζοντας την μεταβολή όγκου και ελαστικής της παραμόρφωσης με το πέρασμα των επαναλήψεων στο παρακάτω διάγραμμα.



Διάγραμμα 4.16

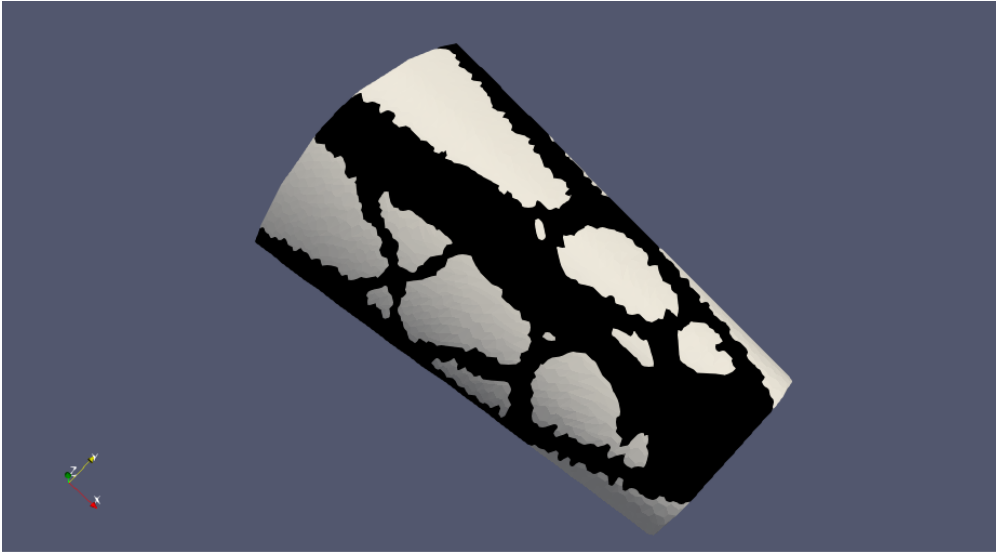
Από το διάγραμμα βλέπουμε πως η καλύτερη δυνατή λύση που μπορεί να επιλεγεί απ' όσες έχουν πετύχει τον απαιτούμενο όγκο είναι η τριακοστή τέταρτη. Προβάλλοντας όμως τα αποτελέσματα της μεθόδου για την εν λόγω επανάληψη βλέπουμε πως δεν είναι υλοποιήσιμα, λόγω των μελών που έχουν κοπεί η οριακά πρόκειται να κοπούν με την παραμικρή πραγματική ένταση.



Εικόνα 4.29: Φορέας BESO μεθόδου 34^{ης} επανάληψης για την 5^η εφαρμογή

Αναγκαζόμαστε λοιπόν να προχωρήσουμε, ώστε να βρεθεί μια λύση η οποία να είναι και υλοποιήσιμη πέραν από ικανοποιητική ως προς τα αποτελέσματά της. Μερικές επαναλήψεις παρακάτω βρίσκεται η τεσσαρακοστή πρώτη επανάληψη, τα αποτελέσματα της οποίας είναι αρκετά ικανοποιητικά σε σύγκριση με τα υπόλοιπα που να είναι όμως και κατασκευάσιμα. Η μορφή φορέα που προκύπτει από την επανάληψη αυτή φαίνεται στην παρακάτω εικόνα.

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα



Εικόνα 4.29: Φορέας BESO μεθόδου 41^{ης} επανάληψης για την 5^η εφαρμογή

Η μέθοδος καταλήγει λοιπόν σε άλλον έναν φορέα που επεκτείνεται κατά το μεγαλύτερο δυνατό ποσοστό στα άκρα του διατιθέμενου του χώρου, σε μια αραχνοειδή μορφή παρόμοια με αυτή που συναντήσαμε στην SIMP μέθοδο. Τα αποτελέσματα της μεθόδου για την τεσσαρακοστή πρώτη επανάληψη της είναι:

Επανάληψη: 41

Όγκος: 0.500267132867124

Ελαστική Παραμόρφωση: 9.73734663436477E-05

Χαρακτηριστική μετατόπιση: 9.74187320474802E-05

Τα αποτελέσματα των δυο μεθόδων βρίσκονται αρκετά κοντά, και ενώ ο φορέας της BESO παρουσίασε μικρό-άλματα στην προσπάθειά του να συντηρήσει διαφορετικές γεωμετρίες δεν αντιμετώπισε τελικώς το πρόβλημα της 3^{ης} εφαρμογής όπου η απότομη μεταβολή τον οδήγησε σε μεγαλύτερη ελαστική παραμόρφωση από αυτήν της SIMP μεθόδου.

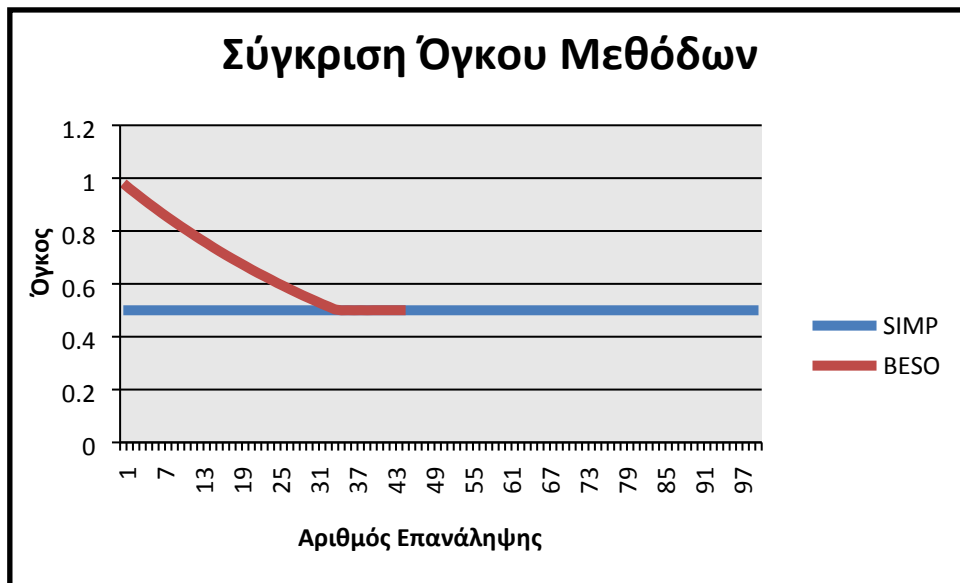
Για άλλη μια φορά έρχεται η σειρά της Level Set μεθόδου να παρουσιάσει την πρότασή της για την βέλτιστη μορφή του φορέα, και για άλλη μια φορά τα αποτελέσματα της δεν είναι ικανοποιητικά και δε μπορούν να συμπεριληφθούν στη σύγκριση με τις άλλες δυο μεθόδους.

Ο καλύτερος φορέας στον οποίο μπόρεσε να συγκλίνει η Level Set μέθοδος παρατίθεται στην παρακάτω εικόνα, για εξακρίβωση της αδυναμίας αυτής στην εύρεση βέλτιστης μορφής.

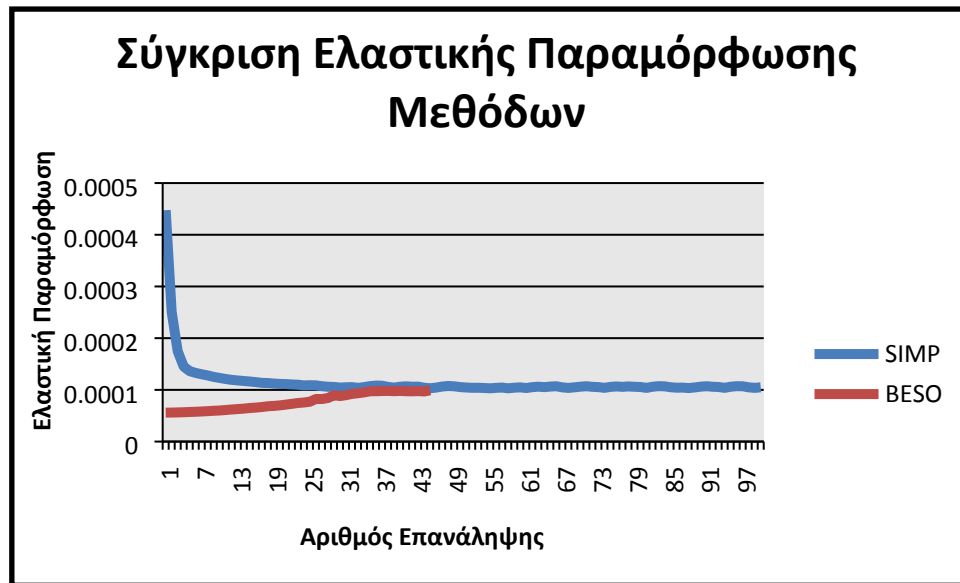


Εικόνα 4.30: Φορέας Level Set0 μεθόδου για την 5^η εφαρμογή

Αφού λοιπόν τα αποτελέσματα της Level Set και πάλι δεν μπορούν συμπεριληφθούν στην σύγκριση των μεθόδων, αυτή θα γίνει μονάχα μεταξύ της SIMP και της BESO. Παρατίθενται λοιπόν παρακάτω τα συγκριτικά τους διαγράμματα ως προς τον όγκο και την ελαστική παραμόρφωση των φορέων τους για την εκάστοτε επανάληψη του αλγορίθμου τους.



Διάγραμμα 4.17



Διάγραμμα 4.18

Βλέπουμε για άλλη μια φορά την BESO μέθοδο να υπερτερεί οριακά έναντι της SIMP και να καταφέρνει την εύρεση φορέα με μικρότερη ελαστική παραμόρφωση, σ αυτήν την τελευταία εφαρμογή της παρούσας διπλωματικής.

ΚΕΦΑΛΑΙΟ 5^ο

ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΡΟΤΑΣΕΙΣ ΓΙΑ ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

5.1 Συμπεράσματα

Έχοντας ολοκληρώσει επαρκή αριθμό εφαρμογών, μπορούμε πλέον να αποτιμήσουμε την ικανότητα των μεθόδων στην επίτευξη βέλτιστων μορφών φορέων που θα παρουσιάζουν την ελάχιστη ελαστική παραμόρφωση για δεδομένες συντοριακές συνθήκες στηρίξεως και φόρτισης. Μέσω των παραδειγμάτων επισημάνθηκαν τα υπέρ και τα κατά της εφαρμογής καθεμιάς σε κάθε είδους πρόβλημα τοπολογικής βελτιστοποίησης σε μη δομημένα πλέγματα. Θα ξεκινήσουμε λοιπόν με μια σύντομη συγκεντρωτική αναφορά των σημείων αυτών, μέσω της οποίας θα κατέστη ξεκάθαρο ποιά είναι η βέλτιστη μέθοδος τοπολογικής βελτιστοποίησης φορέων, στα πλαίσια τουλάχιστον της παρούσας διπλωματικής.

Η SIMP μέθοδος όπως είδαμε ήταν σταθερή και γρήγορη στην εύρεση ενός βέλτιστου φορέα, όπου ως εύρεση δεν εννοούμε την σύγκλιση του αλγορίθμου αλλά την κατάληξη σε έναν φορέα ήδη από μικρό αριθμό επαναλήψεων και στην συνέχεια εφαρμογή μικρό-αλλαγών επ' αυτού στις πυκνότητες των στοιχείων χωρίς όμως αυτές να έχουν κάποια ουσιαστική επίδραση στην ακαμψία του φορέα, όπως φαίνεται άλλωστε και από τα αποτελέσματα. Η δυσκολία ως προς την σύγκλιση της μεθόδου οφείλεται κατά κύριο λόγο στο ότι αυτή διαχειρίζεται πυκνότητες για τα στοιχεία και δεν αντιλαμβάνεται με δυαδικό τρόπο την κατάσταση υλικού αυτών. Έτσι, υπάρχουν κυριολεκτικά άπειροι συνδυασμοί για την δημιουργία ενός φορέα, σε αντίθεση με την περίπτωση ανάγνωσης μόνο ύπαρξης ή όχι υλικού όπου το πλήθος των λύσεων εξαρτάται από την πλήθος των πεπερασμένων στοιχείων για την διακριτοποίηση της κατασκευής. Το πλήθος αυτό μπορεί να είναι σε αρκετές περιπτώσεις πολύ μεγάλο όταν το πρόβλημα βασίζεται σε πλέγματα πολλών στοιχείων, αλλά σίγουρα είναι πεπερασμένο. Ουσιαστικά λοιπόν η ίδια η αντίληψη του υλικού από την SIMP που της δίνει πού μεγαλύτερη ελευθερία στην δημιουργία διαφορετικών φορέων είναι και που της στερεί την δυνατότητα γρήγορης σύγκλισης σε βέλτιστη λύση.

Δεν έχει μονάχα αρνητικά όμως η χρήση πυκνοτήτων για την συμπερίληψη στοιχείων στο φορέα που θα αποτελέσει την λύση, καθώς είναι ο λόγος που της δίνει την δυνατότητα να είναι σχετικώς ανεξάρτητη από την διακριτοποίηση του φορέα. Όπως είδαμε παραπάνω οι άλλες δυο μέθοδοι είναι εξαρτημένες από την πυκνότητα του πλέγματος, με την εξάρτηση αυτή να κυμαίνεται από την μορφή της προκύπτουσας λύσης όταν εφαρμόζεται η BESO, έως και την ικανότητα εύρεσης λύσης όπως είδαμε επανειλημμένα για τις εφαρμογές της Level Set. Η δυνατότητα αυτή την καθιστά ως την καταλληλότερη μεταξύ των μεθόδων για την γρήγορη εύρεση μιας βέλτιστης τοπολογίας σε πραγματικά κατασκευαστικά προβλήματα ο σχεδιασμός των οποίων έχει χρονικούς περιορισμούς, ελέγχοντας βεβαίως συνεχώς την μέθοδο μιας και αν την αφήσει κανείς ελεύθερη όπως είδαμε μπορεί να φτάσει σε μεγάλο αριθμό επαναλήψεων.

Το μεγάλο όμως μειονέκτημα της SIMP μεθόδου είναι πως δεν είναι πρακτική η χρήση πυκνοτήτων για τα στοιχεία που πρόκειται να συνθέσουν τελικώς τον φορέα. Τέτοιου είδους υλικά στα οποία να είναι εφικτός ο καθορισμός της πυκνότητας τους σε κάθε θέση δεν είναι συναντώνται στην φύση, ακόμα όμως και στην περίπτωση που γίνει κάποια μέρα εφικτή η κατασκευή ενός τέτοιου υλικού φαίνεται οικονομικά ασύμφορα όταν

στέκεται σε σύγκριση με την χρήση ή όχι υλικού σε κάθε θέση ενός φορέα. Έτσι η μέθοδος μπορεί μόνον να χρησιμοποιηθεί για να δώσει μια γενική εικόνα των στοιχείων από τα οποία θα πρέπει να αποτελείται ο φορέας, ή στην περίπτωση που θέλει κανείς να επιλέξει αυτούσια την λύση που του προτείνεται από την μέθοδο θα πρέπει να καθορίσει από ποιά πυκνότητα και άνω ένα στοιχείο θα ανήκει στον φορέα, για την τιμή της οποίας συνίστανται τιμές περί το 0.6 με 0.7.

Το πλέον αξιοπερίεργο της παρούσας διπλωματικής παρουσιάζεται κατά την σύγκριση της ελαστικής παραμόρφωσης που επιτυγχάνουν οι φορείς των μεθόδων SIMP και BESO. Η μέθοδος BESO λοιπόν καταφέρνει, προς μεγάλη έκπληξη του γράφοντος, να συγκλίνει σε λύσεις που εμφανίζουν μικρότερη ελαστική παραμόρφωση από αυτήν των λύσεων της SIMP μεθόδου. Το πραγματικά παράξενο σ' αυτό είναι πως η SIMP μέθοδος έχει την δυνατότητα να αναπτύξει ακριβώς τους ίδιους φορείς με την BESO μέθοδο αλλά και πολλές ακόμα λύσεις παραπλήσιας γεωμετρίας δίνοντας πυκνότητα σε στοιχεία που ίσως να ήθελε η BESO αλλά είχε εξαντλήσει τον διατιθέμενο της όγκο, ή αφαιρώντας πυκνότητα από στοιχεία που η BESO οριακά επέλεξε έναντι κάποιων άλλων. Παρόλα αυτά στην πρακτική εφαρμογή τους η BESO συγκλίνει σε καλύτερες λύσεις, κάτι το οποίο δικαιολογείται από τους συγγραφείς της δημοσίευσης της BESO ο αλγόριθμος της οποίας υιοθετήθηκε με την επιχειρηματολογία πως η BESO μέθοδος μπορεί να συγκλίνει σε ολικά ακρότατα του προβλήματος βελτιστοποίησης σε αντίθεση με την SIMP μέθοδο η οποία δυσκολεύεται στην εύρεση αυτού του ολικού ακρότατου και συγκλίνει σε τοπικά ακρότατα της περιοχής γύρω από το ολικό ακρότατο. Αν και κάτι τέτοιο αντιμετωπίστηκε με αμφιβολία και δυσπιστία ως προς την ορθότητά του από τον γραφών, αποδείχθηκε από τα αποτελέσματα πως όντως υφίσταται κάτι τέτοιο παρά την τεράστια ομοιότητα των δυο μεθόδων σε επίπεδο κώδικα και προκυπτουσών λύσεων.

Πέραν του καθαρού πλεονεκτήματος της BESO έναντι της SIMP μεθόδου λόγω των παραπάνω, άλλο ένα πλεονέκτημα πρέπει να της χρεωθεί λόγω της αντίληψης του υλικού από πλευράς της. Η εύρεση λύσεων καθαρά αποτελούμενων από περιοχές υλικού και περιοχές κενού τις κάνουν εφαρμόσιμες σε πραγματικά δομοστατικά προβλήματα ως φορείς που είναι δυνατόν να κατασκευαστούν. Επιπλέον αυτού έχει και την ικανότητα ορισμού του ρυθμού συγκλίσεως της με τον καθορισμό της μεταβλητής er , η οποία βέβαια συνίσταται να μην λαμβάνει μεγάλες τιμές καθώς κάτι τέτοιο θα οδηγήσει σε απότομες μεταβολές του αλγορίθμου που μπορεί να οδηγήσουν με την σειρά τους σε όχι και τόσο βέλτιστες μορφές ή ακόμα και σε μη σύγκλιση του αλγορίθμου.

Παράλληλα όμως το ίδιο πλεονέκτημα οφείλει να καταλογισθεί και στην Level Set μέθοδο για τις μορφές λύσεων που προτείνει, έναντι της SIMP μονάχα βεβαίως μιας και η BESO όπως είπαμε λειτουργεί με τον ίδιο τρόπο. Αυτό όμως είναι και το τελευταίο θετικό έναντι των άλλων μεθόδων που θα τις καταλογισθεί.

Το κυριότερο μειονέκτημα της Level Set είναι η άμεση εξάρτησή της από την πυκνότητα του πλέγματος. Πρόκειται για μια εξάρτηση που συναντήθηκε και κατά την μελέτη της BESO μεθόδου, αλλά σίγουρα όχι στον ίδιο βαθμό. Η Level Set είναι εξαρτώμενη από την διακριτοποίηση όχι μονάχα ως προς της μορφή της λύσεως στην οποία θα συγκλίνει αλλά και ως προς την ίδια την ικανότητα της να μπορέσει να συγκλίνει σε κάποια λύση, έστω και οποιασδήποτε μορφής. Στις εφαρμογές που είδαμε παραπάνω όπου τα πλέγματα είχαν περί τα 2000 πεπερασμένα στοιχεία η μόνη μέθοδος που δε μπορούσε να βρει λύση σε κάποια από αυτά ήταν η Level. Αυτό αν παρατηρήσουμε καλύτερα τις εφαρμογές θα δούμε πως συνέβαινε στα πλέγματα που η λύση απαιτούσε αναγκαστικά αρκετά μέλη, με χαρακτηριστικό παράδειγμα τέτοιας εφαρμογής να είναι η τρίτη στην σειρά, στο κεφάλαιο 4.4. Το πρόβλημα της Level Set είναι πως δε μπορεί να συντηρήσει τις ίδιες φινετσάτες λύσεις που συντηρούν οι άλλες δυο μέθοδοι με την ύπαρξη λεπτών μελών λόγω της ίδιας της της φύσης. Όντας μια επιφάνεια, και ειδικότερα όντας μια επιφάνεια οι τιμές της οποίες έχουν περάσει από φίλτρο ομαλοποίησης, της είναι δύσκολο να αναπτύξει

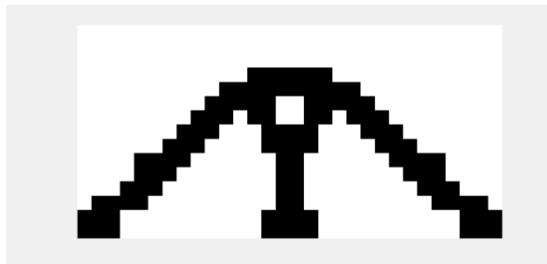
απότομες μεταβολές στην μορφή της ώστε να προκύψουν και να συντηρηθούν τα λεπτά μέλη που συναντώνται στις άλλες δυο μεθόδους.

Προκύπτει όμως το ερώτημα κατά πόσον αυτή είναι όντως μια ιδιαιτερότητα της μεθόδου ή αποτελεί μια αδυναμία της διπλωματικής αυτής να αναπτύξει ορθώς τον αλγόριθμο της Level Set σε μη δομημένα πλέγματα. Για την εξέταση του ερωτήματος αυτού ανατρέχουμε στους κώδικες σε Matlab που παρατίθενται στις δημοσιεύσεις πάνω στις οποίες βασίστηκε η ανάπτυξη του αλγορίθμου της BESO και της Level Set. Έχοντας αντιγράψει τους κώδικες αυτούς καθαυτούς στην Matlab θα επιχειρήσουμε να τους τροφοδοτήσουμε με ένα πρόβλημα αραιής διακριτοποίησης και να παρατηρήσουμε τον τρόπο απόκρισής τους.

Το πρόβλημα είναι σχεδόν ίδιο με αυτό της γέφυρας της δεύτερης εφαρμογής. Δίνεται ένα δομημένο πλέγμα 450 τετρακομβικών πεπερασμένων στοιχείων το οποίο αποτελείται από 30 στοιχεία στην κατά x διεύθυνση και 15 στοιχεία στην κατά y, ενώ ο απαιτούμενος όγκος δίνεται ίσος με 0.25 . Οι εντολές για την κλήση της BESO και για την Level Set θα είναι

```
BESO :      SBESO (30, 15, 0.25, 0.02, 2) ;
Level Set :  top_levelset (30, 15, 0.25, 3, 2, 2) ;
```

Στις δυο παρακάτω εικόνες βλέπουμε τα αποτελέσματα των δυο μεθόδων όταν εφαρμόζονται στο τόσο αραιά διακριτοποιημένο πλέγμα με τόσο λίγο διατιθέμενο όγκο.



Εικόνα 5.1: Λύση BESO μεθόδου για 30x15



Εικόνα 5.2: Λύση Level Set μεθόδου για 30x15

Βλέπουμε λοιπόν πως η Level Set γενικότερα ως μέθοδος δεν μπορεί να συντηρήσει μορφές με λεπτά μέλη, όπου με τον όρο λεπτά δεν εννοείται λεπτά υπό την έννοια κάποιας μονάδος μέτρησης μήκους αλλά λεπτά υπό την έννοια λίγων στοιχείων που να συνθέτουν τις εγκάρσιες διατομές τους. Το επόμενο βήμα είναι να αποδείξουμε πως το πρόβλημα αυτό όντως διορθώνεται με την εφαρμογή πυκνότερη διακριτοποίησης όπως ισχυριζόμαστε παραπάνω, και παράλληλα να δείξουμε πως με την πύκνωση του πλέγματος η BESO μπορεί να πετύχει ακόμα πιο σύνθετες και λεπτεπίλεπτες γεωμετρίες, τις οποίες όμως δεν μπορεί να ακολουθήσει η Level Set. Δεκαεξαπλασιάζουμε λοιπόν τον αριθμό των πεπερασμένων στοιχείων φτάνοντας τα 7200 τετρακομβικά πεπερασμένα στοιχεία, τετραπλασιάζοντας την διακριτοποίηση ως προς τις δυο διευθύνσεις. Οι εντολές για την κλήση των μεθόδων πλέον γίνονται:

```
BESO :      SBESO (120, 60, 0.25, 0.02, 2) ;
Level Set :  top_levelset (120, 60, 0.25, 3, 2, 2) ;
```

Αξιολόγηση μεθόδων τοπολογικής βελτιστοποίησης κατασκευών σε μη δομημένα πλέγματα

Στις παρακάτω εικόνες βλέπουμε τα νέα αποτελέσματα που προκύπτουν με την πύκνωση του πλέγματος



Εικόνα 5.3: Λύση BESO μεθόδου για 120x60



Εικόνα 5.4: Λύση Level Set μεθόδου για 120x60

Όντως λοιπόν η Level εξαρτάται από την πυκνότητα του πλέγματος ως προς την ικανότητά της να μπορεί να συγκλίνει σε λύσεις ανάλογα με τον διατιθέμενο σε αυτήν όγκο για τον φορέα. Συγχρόνως η BESO καταλήγει σε πιο σύνθετες γεωμετρίες, οι οποίες της επιτρέπουν και περαιτέρω μείωση της ελαστικής παραμόρφωσης που εμφανίζει.

Λαμβάνοντας όλα τα παραπάνω υπόψη, και προκειμένου η διπλωματική να κλείσει εκπληρώνοντας την υπόσχεσή της για την υπόδειξη της βέλτιστης μεθόδου για την τοπολογική βελτιστοποίηση φορέων σε μη δομημένα πλέγματα πεπερασμένων στοιχείων πρέπει να καταλήξουμε στην καλύτερη εκ των τριών παραπάνω μεθόδων. Προτού όμως προχωρήσουμε σε κάτι τέτοιο ο γραφών οφείλει να δηλώσει η επιλογή αυτή της καλύτερης μεθόδου τοπολογικής βελτιστοποίησης φορέων περιορίζεται στα στοιχεία που παρέχονται από τις εφαρμογές που εκτελέστηκαν, και σε κάθε περίπτωση περαιτέρω έρευνα πρέπει να γίνει πριν από την εκλογή κάποιας μεθόδου ως ανώτερης κάποιας άλλης. Στα πλαίσια λοιπόν της διπλωματικής αυτής εργασίας ως καλύτερης μεταξύ των μεθόδων για την τοπολογική βελτιστοποίηση σε μη δομημένα πλέγματα αναδεικνύεται η BESO μέθοδος, τα αποτελέσματα της οποίας είχαν στις περισσότερες περιπτώσεις την μικρότερη ελαστική παραμόρφωση για τον φορέα και παράλληλα θα μπορούσαν να εφαρμοστούν και σε έναν πραγματικό σχεδιασμό, χάρη στην ικανότητα της BESO να ορίζει μονάχα καταστάσεις υλικού ή κενού για τα πεπερασμένα στοιχεία που διαχειρίζεται.

5.2 Προτάσεις για μελλοντική έρευνα

Έχοντας ολοκληρώσει την τεκμηριωμένη αποτίμηση των μεθόδων μέσω κατάλληλων εφαρμογών, και έχοντας προτείνει την BESO ως την καλύτερη για την τοπολογική βελτιστοποίηση κατασκευών όταν αυτή είναι ανάγκη να εφαρμοσθεί σε μη δομημένα πλέγματα πεπερασμένων στοιχείων, το έδαφος είναι κατάλληλο για να προχωρήσουμε σε προτάσεις για μελλοντική έρευνα επί του θέματος.

Ίσως η βασικότερη πρόταση για έρευνα είναι η περαιτέρω μελέτη της τοπολογικής βελτιστοποίησης σε μη δομημένα πλέγματα. Αν και η παρούσα διπλωματική θα ήθελε να λάβει τα εύσημα του απόλυτου οδηγού για την επιλογή μεθόδου σε ανάλογες περιπτώσεις οι πέντε εφαρμογές που παρουσιάστηκαν δεν επαρκούν και υπάρχουν πολλά ακόμα σημεία που θα πρέπει να εξετασθούν. Έτσι αν υπάρχει η θέληση για την ικανότητα εφαρμογής τοπολογικής βελτιστοποίησης σε γεωμετρικές των οποίων η μη δομημένη διακριτοποίηση είναι μονόδρομος θα πρέπει να στραφεί το ενδιαφέρον της κοινότητας των μηχανικών στην ανάπτυξη πλήρως αξιόπιστων εξαντλητικά τεκμηριωμένων μεθόδων για την αξιόπιστη εκτέλεση της βελτιστοποίησης τοπολογίας σε τέτοιου είδους πλέγματα.

Επιπλέον, ένα θέμα που θα μπορούσε να αποβεί όχι μονάχα πολύ ενδιαφέρον αλλά και πολύ χρήσιμο, ιδιαίτερα μάλιστα για τον ελλαδικό χώρο, είναι βελτιστοποίηση τοπολογίας σε φορείς υποβαλλόμενους σε δυναμικές διεγέρσεις. Πέραν των εντυπωσιακών μορφών φορέων που μπορεί να προκύψουν από μια τέτοιου είδους τοπολογική βελτιστοποίηση, άξια διαλόγου θα είναι και η διαδικασία με την οποία θα εφαρμοσθεί ο αλγόριθμος της βελτιστοποίησης στις αναλύσεις πεπερασμένων στοιχείων, καθώς δεν υπάρχει ένα αποτέλεσμα αλλά μια χρονοϊστορία μετατοπίσεων και ελαστικής παραμορφώσεως του φορέα.

Βιβλιογραφία

- [1] ADINA R&D Inc., ADINA System Release Notes, 2017
- [2] B. Boroomand, A.R.Barekatein: 'On topology optimization of linear and nonlinear plate problems', *Struct Multidisc Optim* (2009) 39:17–27
- [3] C. Geuzaine and J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, Volume 79, Issue 11, pages 1309-1331, 2009
- [4] G. A. Mohr: 'Finite Elements for Solids, Fluids, and Optimization', Oxford University Press, 1992
- [5] Gregoire Allaire, Francois Jouve, Anca-Maria Toader: 'Structural optimization using sensitivity analysis and a level-set method', *Journal of Computational Physics* 194 (2004) 363–393, 2004
- [6] John Argyris, Lazarus Tenekb, Lars Olofsson: 'TRIC: a simple but sophisticated 3-node triangular element based on 6 rigid-body and 12 straining modes for fast computational simulations of arbitrary isotropic and laminated composite shells', *Comput. Methods Appl. Mech. Engrg.* 145 (1997) 11-85
- [7] Klaus-Jürgen Bathe: 'Finite element procedures', Prentice Hall, 1996
- [8] Martin P. Bendsøe: 'Optimization of structural topology, shape, and material', Springer, 1995
- [9] O.Sigmund: 'A 99 line topology optimization code written in Matlab' *Struct Multidisc Optim* 21, 120–127, 2001
- [10] Vivien J. Challis: 'A discrete level-set topology optimization code written in Matlab', *Struct Multidisc Optim* (2010) 41:453–464, 2009
- [11] Wang, Wei & Scali, Monica & Vignani, Rita & Spadafora, Antonia & Sensi, Elisabetta & Mazzuca, Silvia & Cresti, Mauro. (2013). Wang et al 2003.
- [12] Xiaodong Huang, Yi Min Xie: 'A further review of ESO type methods for topology optimization', *Struct Multidisc Optim* (2010) 41:671–683, 2001
- [13] Μ.Παπαδρακάκης: 'Ανάλυση Φορέων με τη Μέθοδο των Πεπερασμένων Στοιχείων', Εκδόσεις Παπασωτηρίου, Αθήνα 2001
- [14] Μελισσιανός Β., Βερνάρδος Σ., Θανάσουλας Η., Οδηγίες Χρήσης Λογισμικού ADINA, Έκδοση 2.1, 2016.

ΠΑΡΑΡΤΗΜΑ Α

```

1 function [ Kelement ] = ElementMohrTri6( elgc,El,v,thick )
2 %ElementMohrTri6 Calculation of element's stiffness matrix
3 % Calculates the stiffness matrix of a triangular 6 node thick plate
4 % element in the global coordinate system according to the functions
5 % provided by Mohr G.A. in the book
6 % "Finite Elements for Solids, Fluids and Optimization"
7
8 Kelement=zeros(30,30);
9 nnel=6;
10
11 [dir,elc]=dirvecT3(elgc);
12 temp=[dir' zeros(3,2); zeros(2,3) dir(1:2,1:2)'];
13 zero=zeros(size(temp,1),size(temp,2));
14
15 Transform=[ temp , zero , zero , zero , zero , zero
16             zero , temp , zero , zero , zero , zero
17             zero , zero , temp , zero , zero , zero
18             zero , zero , zero , temp , zero , zero
19             zero , zero , zero , zero , temp , zero
20             zero , zero , zero , zero , zero , temp];
21
22 L21=distance(elc(2,:),elc(1,:));
23 L32=distance(elc(3,:),elc(2,:));
24 L13=distance(elc(1,:),elc(3,:));
25
26 cx=[(elc(2,1)-elc(1,1))/L21
27     (elc(3,1)-elc(2,1))/L32
28     (elc(1,1)-elc(3,1))/L13];
29 cy=[(elc(2,2)-elc(1,2))/L21
30     (elc(3,2)-elc(2,2))/L32
31     (elc(1,2)-elc(3,2))/L13];
32
33 ath=acos(((L32^2)+(L13^2)-(L21^2))/(2*L32*L13));
34 bth=acos(((L21^2)+(L13^2)-(L32^2))/(2*L21*L13));
35 cth=acos(((L21^2)+(L32^2)-(L13^2))/(2*L21*L32));
36
37 W=[1 (cos(cth))^2 (cos(bth))^2
38    (cos(cth))^2 1 (cos(ath))^2
39    (cos(bth))^2 (cos(ath))^2 1];
40 C=[cx(1)^2 cy(1)^2 2*cx(1)*cy(1)
41    cx(2)^2 cy(2)^2 2*cx(2)*cy(2)
42    cx(3)^2 cy(3)^2 2*cx(3)*cy(3)];
43 H=(C')/W;
44
45 Delta=L21*L32*(cx(1)*cy(2)-cx(2)*cy(1));
46 Tg=(1/(6*Delta))*[cy(2)*L21*L32-cy(3)*L13*L21 cx(3)*L21*L13-cx(2)*L32*L21
47                  cy(3)*L13*L32-cy(1)*L32*L21 cx(1)*L21*L32-cx(3)*L32*L13
48                  cy(1)*L21*L13-cy(2)*L13*L32 cx(2)*L32*L13-cx(1)*L13*L21]';
49
50 [r,s,w]=TriGauss(4);
51
52 for ir=1:size(r,1)

```

```

53     L1=1-r(ir)-s(ir);
54     L2=r(ir);
55     L3=s(ir);
56
57     N=[L1*(2*L1-1); L2*(2*L2-1); L3*(2*L3-1); 4*L1*L2; 4*L2*L3; 4*L3*L1];
58     Na=(1/L21)*[1-4*L1; 4*L2-1; 0; 4*L1-4*L2; 4*L3; -4*L3];
59     Nb=(1/L32)*[0; 1-4*L2; 4*L3-1; -4*L1; 4*L2-4*L3; 4*L1];
60     Nc=(1/L13)*[4*L1-1; 0; 1-4*L3; 4*L2; -4*L2; 4*L3-4*L1];
61     N1=[4*L1-1; 0; 4*L1+4*L2-3; 4*L2; -4*L2; 4-8*L1-4*L2];
62     N2=[0; 4*L2-1; 4*L1+4*L2-3; 4*L1; 4-4*L1-8*L2; -4*L1];
63
64     S=[N1'
65         N2'];
66     Jinv=(1/(2*Delta))*[-(elc(3,2)-elc(2,2)) -(elc(1,2)-elc(3,2))
67                          (elc(3,1)-elc(2,1)) (elc(1,1)-elc(3,1))];
68     G=Jinv*S;
69
70     Bm=[G(1,1) 0 G(1,2) 0 G(1,3) 0 G(1,4) 0 G(1,5) 0 G(1,6) 0
71          0 G(2,1) 0 G(2,2) 0 G(2,3) 0 G(2,4) 0 G(2,5) 0 G(2,6)
72          G(2,1) G(1,1) G(2,2) G(1,2) G(2,3) G(1,3) G(2,4) G(1,4) G(2,5) G(1,5) G(2,6) G(1,6)];
73
74     D=[thick^2 v*(thick^2) 0 0 0
75         v*(thick^2) thick^2 0 0 0
76         0 0 0.5*(1-v)*(thick^2) 0 0
77         0 0 0 5*(1-v) 0
78         0 0 0 0 5*(1-v)];
79     D=(El*thick/(12*(1-(v^2))))*D;
80     Dm=(thick*El/(1-(v^2)))*[1 v 0
81                               v 1 0
82                               0 0 (1-v)/2];
83
84     Bb=zeros(3,3*nnel); Bs=Bb;
85     for in=1:nnel
86         Bb(:,3*(in-1)+1:3*(in-1)+3)=[ 0 cx(1)*Na(in) cy(1)*Na(in)
87                                         0 cx(2)*Nb(in) cy(2)*Nb(in)
88                                         0 cx(3)*Nc(in) cy(3)*Nc(in)];
89         Bs(:,3*(in-1)+1:3*(in-1)+3)=[ Na(in) -cx(1)*N(in) -cy(1)*N(in)
90                                         Nb(in) -cx(2)*N(in) -cy(2)*N(in)
91                                         Nc(in) -cx(3)*N(in) -cy(3)*N(in)];
92     end
93     B=[H*Bb; Tg*Bs];
94
95     indexm=[1 2 6 7 11 12 16 17 21 22 26 27];
96     indexbs=[3 4 5 8 9 10 13 14 15 18 19 20 23 24 25 28 29 30];
97     Kelement(indexbs,indexbs)=Kelement(indexbs,indexbs)+Delta*w(ir)*(B'*D*B);
98     Kelement(indexm,indexm)=Kelement(indexm,indexm)+Delta*w(ir)*(Bm'*Dm*Bm);
99
100    end
101    Kelement=Transform'*Kelement*Transform;
102 end

```

```

1  function D = distance( a,b )
2  %distance between two 3D points
3  D=sqrt(((a(1)-b(1))^2)+((a(2)-b(2))^2)+((a(3)-b(3))^2));
4  end

```

```

1  function c = cross_product( a,b )

```

```

2 %cross product of two vectors
3 c=[a(2)*b(3)-a(3)*b(2) b(1)*a(3)-a(1)*b(3) a(1)*b(2)-a(2)*b(1)];
4 end
1 function [ dir,elc] = dirvecT3(elncoor)
2 %dirvecT3 Calculation of Direction Vectors and Local Coordinates for a
3 %triangularr element
4 % The Local Coordinate System is constructed so that the x axis is
5 % parallel to the 1-2 side of the triangle, the z axis is vertical to
6 % the plane surface of the triangle and the y axis is orthogonal to
7 % the other two while maintaining a counter-clockwise order of the x-y-z
8 % axes
9 dir=zeros(3,3);
10 dir(1,:)=elncoor(2,:)-elncoor(1,:);
11 dir(1,:)=dir(1,:)/distance(dir(1,:),[0 0 0]);
12 dir(3,:)=cross_product(elncoor(2,:)-elncoor(1,:),elncoor(3,:)-elncoor(1,:));
13 dir(3,:)=dir(3,:)/(distance(dir(3,:),[0 0 0]));
14 dir(2,:)=cross_product(dir(3,:),dir(1,:));
15 dir=dir';
16
17 elc=((dir')*(elncoor'));
18 end

1 function [r,s,w] = TriGauss ( option )
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4 % Function TriGauss provides the Gaussian points and weights %
5 % for the Gaussian quadrature of order 'option' for the standard triangles. %
6 % %
7 % Input: option - the order of the Gaussian quadrature (option<=8) %
8 % %
9 % Output: [r,s,w] %
10 % r gives the x-coordinates of triangular isoparametric points %
11 % r gives the y-coordinates of triangular isoparametric points %
12 % w gives the weights %
13 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
14 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15 if (option == 1)
16 Gauss=[0.33333333333333 0.33333333333333 1.00000000000000];
17 elseif (option == 2)
18 Gauss=[0.16666666666667 0.16666666666667 0.33333333333333
19 0.16666666666667 0.66666666666667 0.33333333333333
20 0.66666666666667 0.16666666666667 0.33333333333333];
21 elseif (option == 3)
22 Gauss=[0.33333333333333 0.33333333333333 -0.56250000000000
23 0.20000000000000 0.20000000000000 0.52083333333333
24 0.20000000000000 0.60000000000000 0.52083333333333
25 0.60000000000000 0.20000000000000 0.52083333333333];
26 elseif (option == 4)
27 Gauss=[0.44594849091597 0.44594849091597 0.22338158967801
28 0.44594849091597 0.10810301816807 0.22338158967801
29 0.10810301816807 0.44594849091597 0.22338158967801
30 0.09157621350977 0.09157621350977 0.10995174365532
31 0.09157621350977 0.81684757298046 0.10995174365532
32 0.81684757298046 0.09157621350977 0.10995174365532];
33 elseif (option == 5)
34 Gauss=[0.33333333333333 0.33333333333333 0.22500000000000

```



```
35 0.47014206410511 0.47014206410511 0.13239415278851
36 0.47014206410511 0.05971587178977 0.13239415278851
37 0.05971587178977 0.47014206410511 0.13239415278851
38 0.10128650732346 0.10128650732346 0.12593918054483
39 0.10128650732346 0.79742698535309 0.12593918054483
40 0.79742698535309 0.10128650732346 0.12593918054483];
41 elseif (option == 6)
42 Gauss=[0.24928674517091 0.24928674517091 0.11678627572638
43 0.24928674517091 0.50142650965818 0.11678627572638
44 0.50142650965818 0.24928674517091 0.11678627572638
45 0.06308901449150 0.06308901449150 0.05084490637021
46 0.06308901449150 0.87382197101700 0.05084490637021
47 0.87382197101700 0.06308901449150 0.05084490637021
48 0.31035245103378 0.63650249912140 0.08285107561837
49 0.63650249912140 0.05314504984482 0.08285107561837
50 0.05314504984482 0.31035245103378 0.08285107561837
51 0.63650249912140 0.31035245103378 0.08285107561837
52 0.31035245103378 0.05314504984482 0.08285107561837
53 0.05314504984482 0.63650249912140 0.08285107561837];
54 elseif (option == 7)
55 Gauss=[0.33333333333333 0.33333333333333 -0.14957004446768
56 0.26034596607904 0.26034596607904 0.17561525743321
57 0.26034596607904 0.47930806784192 0.17561525743321
58 0.47930806784192 0.26034596607904 0.17561525743321
59 0.06513010290222 0.06513010290222 0.05334723560884
60 0.06513010290222 0.86973979419557 0.05334723560884
61 0.86973979419557 0.06513010290222 0.05334723560884
62 0.31286549600487 0.63844418856981 0.07711376089026
63 0.63844418856981 0.04869031542532 0.07711376089026
64 0.04869031542532 0.31286549600487 0.07711376089026
65 0.63844418856981 0.31286549600487 0.07711376089026
66 0.31286549600487 0.04869031542532 0.07711376089026
67 0.04869031542532 0.63844418856981 0.07711376089026];
68 elseif (option == 8)
69 Gauss=[0.33333333333333 0.33333333333333 0.14431560767779
70 0.45929258829272 0.45929258829272 0.09509163426728
71 0.45929258829272 0.08141482341455 0.09509163426728
72 0.08141482341455 0.45929258829272 0.09509163426728
73 0.17056930775176 0.17056930775176 0.10321737053472
74 0.17056930775176 0.65886138449648 0.10321737053472
75 0.65886138449648 0.17056930775176 0.10321737053472
76 0.05054722831703 0.05054722831703 0.03245849762320
77 0.05054722831703 0.89890554336594 0.03245849762320
78 0.89890554336594 0.05054722831703 0.03245849762320
79 0.26311282963464 0.72849239295540 0.02723031417443
80 0.72849239295540 0.00839477740996 0.02723031417443
81 0.00839477740996 0.26311282963464 0.02723031417443
82 0.72849239295540 0.26311282963464 0.02723031417443
83 0.26311282963464 0.00839477740996 0.02723031417443
84 0.00839477740996 0.72849239295540 0.02723031417443];
85
86
87 end
88 r=Gauss(:,1);
89 s=Gauss(:,2);
90 w=Gauss(:,3);
91 end
```

ΠΑΡΑΡΤΗΜΑ Β

```

1 function [ x ] = TopologySIMP( nodes,elemConn,volfrac,rmin,penal,E,v,thickness,fixedDOFs,F )
2 % INITILIAZE
3 nel=size(elemConn,1);
4 change=1;
5 x(1:nel)=volfrac;
6 loop=0;
7 rmin0=rmin;
8 rminold=rmin;
9 centerPoints=zeros(nel,3);
10 % CALCULATION OF STIFFNESSES FOR THE TRIANGULAR ELEMENTS
11 Kelements=zeros(30,30,nel);
12 for i=1:nel
13     centerPoints(i,:)=(nodes(elemConn(i,1,:))+nodes(elemConn(i,2,:))+nodes(elemConn(i,3,:)))/3;
14     Kelements(:,i)=ElementMohrTri6(nodes(elemConn(i,:)),E,v,thickness);
15 end
16 % START OF ITERATIONS
17 while change>0.01
18     loop=loop+1;
19     xold=x;
20     c=0;
21     dc=zeros(nel,1);
22     U=FEA(Kelements,nodes,elemConn,x,penal,fixedDOFs,F);
23     if (rmin~=rminold)
24         connect=Connections(centerPoints,rmin);
25         rminold=rmin;
26     end
27 %OBJECTIVE FUNCTION AND SENSITIVITY ANALYSIS
28     for i=1:nel
29         elemDOFs=ElementDOFs(elemConn(i,:));
30         Ue=U(elemDOFs,1);
31         c=c+x(i)^penal*Ue'*Kelements(:,i)*Ue;
32         dc(i,1)=-penal*x(i)^(penal-1)*Ue'*Kelements(:,i)*Ue;
33     end
34 %FILTERING SENSITIVITIES
35     dc=checkSIMP(connect,nel,rmin,x,dc);
36 %DESIGN UPDATE BY THE OPTIMALITY CRITERIA METHOD
37     x=OC(nel,x,volfrac,dc);
38 %CHECK FOR CONVERGENCE
39     change=max(abs(x-xold));
40     if (change < 0.1)
41         rmin = rmin0 / (1.5 - 5 * change);
42     end
43     if loop==200
44         change=0;
45     end
46 end

1 function [ U ]=FEA( Kelements,nodes,elemConn,x,penal,fixedDOFs,F )
2 nnodes=size(nodes,1);
3 nel=size(elemConn,1);
4 K=zeros(nnodes*5,nnodes*5);
5 for i=1:nel

```

```

6   elDOFs=ElementDOFs(elemConn(i,:));
7   if (x(i)>0.25)
8       K(elDOFs,elDOFs)=K(elDOFs,elDOFs)+(x(i)^penal)*Kelements(:,i);
9   else
10      K(elDOFs,elDOFs)=K(elDOFs,elDOFs)+(x(i)*sqrt(1/250))*Kelements(:,i);
11  end
12  end
13  allDOFs=[1:5*nnodes];
14  freeDOFs=setdiff(allDOFs,fixedDOFs);
15  U=zeros(5*nnodes,1);
16  U(freeDOFs,1)=K(freeDOFs,freeDOFs)\F(freeDOFs);
17  end

```

```

1  function [ dcn ]= checkSIMP( connect,centerPoints,rmin,x,dc )
2  nel=length(x);
3  dcn(1:nel)=0;
4  for i=1:nel
5      sum=0;
6      for j=1:nel
7          if (connect(i,j)>0)
8              fac=rmin-distance(centerPoints(i,:),centerPoints(j,:));
9              sum=sum+max(0,fac);
10             dcn(i)=dcn(i)+max(0,fac)*x(j)*dc(j);
11         end
12     end
13     dcn(i)=dcn(i)/(x(i)*sum);
14 end
15 end

```

```

1  function [ x ]=OC(x,volfrac,dc)
2  nel=length(x);
3  l1=0;
4  l2=10000000;
5  move=0.2;
6  while (l2-l1>1e-10)
7      lmid=0.5*(l1+l2);
8      xnew=max(0.001,max(x-move,min(1,min(x+move,x.*sqrt(-dc./lmid)))));
9      if sum(xnew)>volfrac*nel;
10         l1=lmid;
11     else
12         l2=lmid;
13     end
14 end
15 end

```

```

1  function [connect] = Connections( centerPoints,rmin )
2  connect=zeros(nel,nel);
3  for i=1:nel
4      for j=1:nel
5          if (distance(centerPoints(i,:),centerPoints(j,:))<rmin)
6              connect(i,j)=j;

```

```
7     end
8     end
9  end
10 end
```

```
1  function [ ElementDOFs ] = ElementDOFs(elemConn)
2  ElementDOFs=zeros(size(elemConn,1),30);
3  for i=1:size(elemConn,1)
4      for j=1:6
5          for k=1:5
6              ElementDOFs(i,(j-1)*5+k)=(elemConn(i,j)-1)*5+k;
7          end
8      end
9  end
10 end
```

ΠΑΡΑΡΤΗΜΑ Γ

```
1 function [ ] = TopologyBESO( nodes,elemConn,er,volfrac,rmin,penal,E,v,thickness,fixedDOFs,F )
2 %INITIALIZATION
3 nel=size(elemConn,1);
4 change=1;
5 x(1:nel)=1;
6 dc(1:nel)=0;
7 vol=1;
8 loop=0;
9 c=zeros(200);
10 % CALCULATION OF STIFFNESSES FOR THE TRIANGULAR ELEMENTS
11 centerPoints=zeros(nel,3);
12 Kelements=zeros(30,30,nel);
13 for i=1:nel
14     centerPoints(i,:)=(nodes(elemConn(i,1,:))+nodes(elemConn(i,2,:))+nodes(elemConn(i,3,:)))/3;
15     Kelements(:,i)=ElementMohrTri6(nodes(elemConn(i,:)),E,v,thickness);
16 end
17 connect=Connections(centerPoints,rmin);
18 % START OF ITERATIONS
19 while change>0.001
20     loop=loop+1;
21     vol=max(vol*(1-er),volfrac);
22     if (loop>1)
23         olddc=dc;
24     end
25     %FINITE ELEMENT ANALYSIS
26     U=FEA(Kelements,nodes,elemConn,x,penal,fixedDOFs,F);
27     for i=1:nel
28         elemDOFs=ElementDOFs(elemConn(i,:));
29         Ue=U(elemDOFs,1);
30         c(loop)=c(loop)+0.5*x(i)^penal*Ue'*Kelements(:,i)*Ue;
31         dc(i)=0.5*x(i)^(penal-1)*Ue'*Kelements(:,i)*Ue;
32     end
33     %FILTERING OF SENSITIVITIES
34     dc=checkBESO(connect,centerPoints,nel,rmin,dc);
35     %STABILIZATION OF EVOLUTIONARY PROCESS
36     if (loop>1)
37         dc=0.5*(dc+olddc);
38     end
39     %BESO DESIGN UPDATE
40     x=ADDDEL(dc,vol);
41     if loop>10
42         change=abs(sum(c(loop-9:loop-5))-sum(c(loop-4:loop)))/sum(c(loop-4:loop));
43     end
44 end
```

```
1 function [ U ]=FEA( Kelements,nodes,elemConn,x,penal,fixedDOFs,F )
2 nnodes=size(nodes,1);
3 nel=size(elemConn,1);
4 K=zeros(nnodes*5,nnodes*5);
5 for i=1:nel
6     eIDOFs=ElementDOFs(elemConn(i,:));
```

```

7   if (x(i)>0.25)
8       K(eIDOFs,eIDOFs)=K(eIDOFs,eIDOFs)+(x(i)^penal)*Kelements(:,i);
9   else
10      K(eIDOFs,eIDOFs)=K(eIDOFs,eIDOFs)+(x(i)*sqrt(1/250))*Kelements(:,i);
11  end
12  end
13  allDOFs=[1:5*nnodes];
14  freeDOFs=setdiff(allDOFs,fixedDOFs);
15  U=zeros(5*nnodes,1);
16  U(freeDOFs,1)=K(freeDOFs,freeDOFs)\F(freeDOFs);
17  end

```

```

1  function [ dc ] = checkBESO( connect,centerPoints,rmin,dc )
2  nel=size(centerPoints,1);
3  dcn(1:nel)=0;
4  for i=1:nel
5      sum=0;
6      for j=1:nel
7          if (connect(i,j)>0)
8              fac=rmin-distance(centerPoints(i,:),centerPoints(j,:));
9              sum=sum+max(0,fac);
10             dcn(i)=dcn(i)+max(0,fac)*dc(j);
11         end
12     end
13     dcn(i)=dcn(i)/sum;
14 end
15 end

```

```

1  function [ x ] = ADDDEL( dc,vol )
2  l1=min(dc);
3  l2=max(dc);
4  nel=length(dc);
5  while((l2-l1)/l2>1e-5)
6      lmid=0.5*(l1+l2);
7      x=max(0.001,sign(dc-lmid));
8      if (sum(x)>vol*nel)
9          l1=lmid;
10     else
11         l2=lmid;
12     end
13 end
14 end

```

```

1  function [connect] = Connections( centerPoints,rmin )
2  connect=zeros(nel,nel);
3  for i=1:nel
4      for j=1:nel
5          if (distance(centerPoints(i,:),centerPoints(j,:))<rmin)
6              connect(i,j)=j;
7          end
8      end
9  end
10 end

```

ΠΑΡΑΡΤΗΜΑ Δ

```
1 function [ struc ] = TopologyLS( nodes,elemConn,elemSize, volfrac, stepLength, numReinit,
topWeight, E,v,thickness,fixedDOFs,F,ForcedNodes,gridx,gridy,point1,point2 )
2 %INITIALIZATION
3 nel=size(elemConn,1);
4 struc=ones(nel,1);
5 shapeSens=zeros(nel,1);
6 topSens=zeros(nel,1);
7 lambda=E*v/((1+v)*(1-v));
8 mu=E/(2*(1+v));
9 la=-0.01;
10 La=1000;
11 alpha=0.9;
12 objective=zeros(200,1);
13 %CENTER POINTS CREATION-CONNECTIONS BETWEEN ELEMENTS
14 centerPoints=zeros(nel,3);
15 for i=1:nel
16     centerPoints(i,:)=(nodes(elemConn(i,1,:),:)+nodes(elemConn(i,2,:),:)+nodes(elemConn(i,3,:),:))/3;
17 end
18 [ connections ] = Connect( centerPoints,elemSize );
19 %GRID CREATION-FAKE ELEMENTS CREATION
20 [ grid ] = GridCreator( point1,point2,gridx,gridy );
21 [ centerPointsFake,connectionsFake ] = CreateFakeElements(centerPoints,connections,elemSize);
22 % TRANSFORMATIONS BETWEEN GRID AND MESH
23 [ meshToGrid ] = MeshToGrid( grid,centerPoints,rmin );
24 [ fakeMeshToGrid ] = FakeMeshToGrid( centerPointsFake,grid,rmin );
25 [ gridToMesh ] = GridToMesh( grid,centerPoints,rmin );
26 [ gridToFakeMesh ] = GridToFakeMesh( grid,centerPointsFake,rmin );
27 % CALCULATION OF WEIGHTS FOR THE TRANSFORMATIONS BETWEEN GRID AND MESH
28 [ meshWeights ] = MeshWeights( meshToGrid,centerPoints,grid,rmin );
29 [ fakeMeshWeights ] = FakeMeshWeights( fakeMeshToGrid,centerPointsFake,grid,rmin );
30 [ gridWeights ] = GridWeights( gridToMesh,centerPoints,grid,rmin );
31 [ fakeGridWeights ] = FakeGridWeights( gridToFakeMesh,centerPointsFake,grid,rmin );
32 % CALCULATION OF STIFFNESSES FOR THE TRIANGULAR ELEMENTS
33 Kelements=zeros(30,30,nel);
34 for i=1:nel
35     Kelements(:, :, i)=ElementMohrTri6(nodes(elemConn(i, :, :)),E,v,thickness);
36 end
37 % START OF ITERATIONS
38 [ Isf ] = Reinit(
struc,centerPoints,centerPointsFake,elemSize,gridToMesh,gridToFakeMesh,gridWeights,fakeGridWei
ghts );
39 for iterNum=1:200
40     strucx=max(struc,0.001);
41     [ U ]=FEALS( Kelements,nodes,elemConn,strucx,fixedDOFs,F );
42     for i=1:nel
43         eIDOFs=ElementDOFs(elemConn(i,:));
44         Ue=U(eIDOFs,1);
45         shapeSens(i)=max(struc(i),0.001)*Ue'*Kelements(:, :, i)*Ue;
46     end
47     topSens(i)=struc(i)*pi/2*(lambda+2*mu)/mu/(lambda+mu)*(4*mu*Ue'*Kelements(:, :, i)*Ue);
48     volCurr=sum(struc)/nel;
```

```

49 objective(iterNum)=-sum(shapeSens);
50 if (iterNum>5)&&(abs(volCurr-volfrac)<0.02)&&(all(abs(objective(iterNum)-objective(iterNum-
5:iterNum-1))<0.01*abs(objective(iterNum)))));
51     return;
52 end
53 if iterNum>1
54     la=la-1/La*(volCurr-volfrac);
55     La=alpha*La;
56     shapeSens=shapeSens-la+1/La*(volCurr-volfrac);
57     topSens=topSens+pi*(la-1/La*(volCurr-volfrac));
58 end
59 [ struc,lsf ] = UpdateStep(
lsf,shapeSens,topSens,gridToMesh,gridWeights,meshToGrid,meshWeights,stepLength,topWeight );
60 if (~mod(iterNum,numReinit))
61     [ lsf ] = Reinit(
struc,centerPoints,centerPointsFake,elemSize,gridToMesh,gridToFakeMesh,gridWeights,fakeGridWei
ghts );
62 end
63 end
64 end

```

```

1 function [ grid ] = GridCreator( point1,point2,gridx,gridy )
2 sizeX=abs((point1(1)-point2(1))/gridx-1);
3 sizeY=abs((point1(2)-point2(2))/gridy-1);
4 grid=zeros(gridx,grdy,2);
5 for i=1:gridx
6     for j=1:gridy
7         grid(i,j,1)=point1(0)+i*sizeX;
8         grid(i,j,2)=point1(1)+j*sizeY;
9     end
10 end
11 end

```

```

1 function [ gridToMesh ] = GridToMesh( grid,centerPoints,rmin )
2 gridToMesh=zeros(size(grid,1),size(grid,2),size(centerPoints,1));
3 for i=1:size(grid,1)
4     for j=1:size(grid,2)
5         for iel=size(centerPoints,1)
6             if (distance2D(grid(i,j,:),centerPoints(iel,[1 2]))<rmin);
7                 gridToMesh(i,j,iel)=iel;
8             end
9         end
10     end
11 end
12 end

```

```

1 function [ meshToGrid ] = MeshToGrid( grid,centerPoints,rmin )
2 meshToGrid=zeros(size(centerPoints,1),size(grid,1),size(grid,2));
3 for iel=1:size(centerPoints,1);
4     for i=1:size(grid,1)
5         for j=1:size(grid,2)
6             if (distance2D(grid(i,j,:),centerPoints(iel,[1 2]))<rmin)
7                 meshToGrid(iel,i,j)=1;
8             end

```



```

9     end
10    end
11   end
12  end

```

```

1  function [ connectionsYminus ] = ConnectionsYminus( centerPoints,elemSize )
2  connectionsYminus=zeros(size(centerPoints,1),1);
3  for i=1:size(centerPoints,1)
4      maxDist=3*elemSize;
5      minDist=10000*elemSize;
6      for j=1:size(centerPoints,1)
7          if (centerPoints(j,2)<centerPoints(i,2))
8              if distance(centerPoints(i,:),centerPoints(j,:))<maxDist
9                  if distance(centerPoints(i,:),centerPoints(j,:))<minDist
10                     if (abs(centerPoints(i,1)-centerPoints(j,1))<abs(centerPoints(i,2)-centerPoints(j,2)))
11                         connectionsYminus(i)=j;
12                         minDist=distance(centerPoints(i,:),centerPoints(j,:));
13                     end
14                 end
15             end
16         end
17     end
18 end
19 end

```

```

1  function [ connectionsYplus ] = ConnectionsYplus( centerPoints,elemSize )
2  connectionsYplus=zeros(size(centerPoints,1),1);
3  for i=1:size(centerPoints,1)
4      maxDist=3*elemSize;
5      minDist=10000*elemSize;
6      for j=1:size(centerPoints,1)
7          if (centerPoints(j,2)>centerPoints(i,2))
8              if distance(centerPoints(i,:),centerPoints(j,:))<maxDist
9                  if distance(centerPoints(i,:),centerPoints(j,:))<minDist
10                     if (abs(centerPoints(i,1)-centerPoints(j,1))<abs(centerPoints(i,2)-centerPoints(j,2)))
11                         connectionsYplus(i)=j;
12                         minDist=distance(centerPoints(i,:),centerPoints(j,:));
13                     end
14                 end
15             end
16         end
17     end
18 end
19 end

```

```

1  function [ connectionsXminus ] = ConnectionsXminus( centerPoints,elemSize )
2  connectionsXminus=zeros(size(centerPoints,1),1);
3  for i=1:size(centerPoints,1)
4      maxDist=3*elemSize;
5      minDist=10000*elemSize;
6      for j=1:size(centerPoints,1)
7          if (centerPoints(j,1)<centerPoints(i,1))
8              if distance(centerPoints(i,:),centerPoints(j,:))<maxDist

```

```

9         if distance(centerPoints(i,:),centerPoints(j,:))<minDist
10             if (abs(centerPoints(i,1)-centerPoints(j,1))>abs(centerPoints(i,2)-centerPoints(j,2)))
11                 connectionsXminus(i)=j;
12                 minDist=distance(centerPoints(i,:),centerPoints(j,:));
13             end
14         end
15     end
16 end
17 end
18 end
19 end

```

```

1 function [ connectionsXplus ] = ConnectionsXplus( centerPoints,elemSize )
2 connectionsXplus=zeros(size(centerPoints,1),1);
3 for i=1:size(centerPoints,1)
4     maxDist=3*elemSize;
5     minDist=10000*elemSize;
6     for j=1:size(centerPoints,1)
7         if (centerPoints(j,1)>centerPoints(i,1))
8             if distance(centerPoints(i,:),centerPoints(j,:))<maxDist
9                 if distance(centerPoints(i,:),centerPoints(j,:))<minDist
10                     if (abs(centerPoints(i,1)-centerPoints(j,1))>abs(centerPoints(i,2)-centerPoints(j,2)))
11                         connectionsXplus(i)=j;
12                         minDist=distance(centerPoints(i,:),centerPoints(j,:));
13                     end
14                 end
15             end
16         end
17     end
18 end
19 end

```

```

1 function [ centerPointsFake,connectionsFake ] = CreateFakeElements(
centerPoints,connections,elemSize )
2 connectionsFake=zeros(size(centerPoints,1),4);
3 tempCenterPointsFake=zeros(size(centerPoints,1)*4,3);
4 sum=0;
5 for i=1:size(centerPoints,1)
6     if connections(i,1)==0
7         sum=sum+1;
8         connectionsFake(i,1)=sum;
9         tempCenterPointsFake(sum,:)=centerPoints(i,:)+[elemSize 0 0];
10    end
11    if connections(i,2)==0
12        sum=sum+1;
13        connectionsFake(i,2)=sum;
14        tempCenterPointsFake(sum,:)=centerPoints(i,:)+[-elemSize 0 0];
15    end
16    if connections(i,3)==0
17        sum=sum+1;
18        connectionsFake(i,3)=sum;
19        tempCenterPointsFake(sum,:)=centerPoints(i,:)+[0 elemSize 0];
20    end

```

```

21   if connections(i,4)==0
22       sum=sum+1;
23       connectionsFake(i,4)=sum;
24       tempCenterPointsFake(sum,:)=centerPoints(i,:)+[0 -elemSize 0];
25   end
26 end
27 centerPointsFake=zeros(sum,1);
28 for i=1:sum
29     centerPointsFake(i,:)=tempCenterPointsFake(i,:);
30 end
31 end

```

```

1   function [ bwdist ] = Bwdist( image,imageSupport,centerPoints,centerPointsSupport,elemSize )
2   bwdist=zeros(length(image),1);
3   for i=1:length(image)
4       mindist=100000*elemSize;
5       for j=1:length(image)
6           if (image(j)~=0)
7               if (distance(centerPoints(i,:),centerPoints(j,:))<mindist)
8                   mindist=distance(centerPoints(i,:),centerPoints(j,:));
9               end
10          end
11      end
12      for j=1:length(imageSupport)
13          if (imageSupport(j)~=0)
14              if (distance(centerPoints(i,:),centerPointsSupport(j,:))<mindist)
15                  mindist=distance(centerPoints(i,:),centerPointsSupport(j,:));
16              end
17          end
18      end
19      bwdist(i)=mindist;
20  end
21  end

```

```

1   function [ fakeMeshToGrid ] = FakeMeshToGrid( centerPointsFake,grid,rmin )
2   fakeMeshToGrid=zeros(length(centerPointsFake),size(grid,1),size(grid,2));
3   for iel=1:length(centerPointsFake);
4       for i=1:size(grid,1)
5           for j=1:size(grid,2)
6               if (distance2D(centerPointsFake(iel,[1 2]),grid(i,j,:))<rmin)
7                   fakeMeshToGrid(iel,i,j)=1;
8               end
9           end
10      end
11  end
12  end

```

```

1   function [ gridToFakeMesh ] = GridToFakeMesh( grid,centerPointsFake,rmin )
2   gridToFakeMesh=zeros(size(grid,1),zeros(grid,2),length(centerPointsFake));
3   for i=1:size(grid,1)
4       for j=1:size(grid,2)
5           for iel=1:length(centerPointsFake)
6               if (distance2D(centerPointsFake(iel,[1 2]),grid(i,j,:))<rmin);
7                   gridToFakeMesh(i,j,iel)=iel;

```

```

8     end
9     end
10    end
11   end
12  end

```

```

1  function [ connections ] = Connect( centerPoints,elemSize )
2  connections=zeros(length(centerPoints),4);
3  connections(:,1)=ConnectionsXplus(centerPoints,elemSize);
4  connections(:,2)=ConnectionsXminus(centerPoints,elemSize);
5  connections(:,3)=ConnectionsYplus(centerPoints,elemSize);
6  connections(:,4)=ConnectionsYminus(centerPoints,elemSize);
7  end

```

```

1  function [ meshWeights ] = MeshWeights( meshToGrid,centerPoints,grid,rmin )
2  meshWeights=zeros(length(centerPoints),size(grid,1),size(grid,2));
3  for iel=1:length(centerPoints)
4    sum=0;
5    for i=1:size(grid,1)
6      for j=1:size(grid,2)
7        if (meshToGrid(iel,i,j)>0)
8          sum=sum+max(0,(rmin-distance2D(centerPoints(iel,[1 2]),grid(i,j))))^3);
9          meshWeights(iel,i,j)=max(0,(rmin-distance2D(centerPoints(iel,[1 2]),grid(i,j))))^3);
10         end
11       end
12     end
13     meshWeights(iel,:,:) = meshWeights(iel,:,+)/sum;
14   end
15 end

```

```

1  function [ fakeMeshWeights ] = FakeMeshWeights( fakeMeshToGrid,centerPointsFake,grid,rmin )
2  fakeMeshWeights=zeros(length(centerPointsFake),size(grid,1),size(grid,2));
3  for iel=1:length(centerPointsFake)
4    sum=0;
5    for i=1:size(grid,1)
6      for j=1:size(grid,2)
7        if (fakeMeshToGrid(iel,i,j)>0)
8          sum=sum+max(0,(rmin-distance2D(centerPointsFake(iel,[1 2]),grid(i,j))))^3);
9          fakeMeshWeights(iel,i,j)=max(0,(rmin-
distance2D(centerPointsFake(iel,[1 2]),grid(i,j))))^3);
10         end
11       end
12     end
13     fakeMeshWeights(iel,:,:) = fakeMeshWeights(iel,:,+)/sum;
14   end
15 end

```

```

1  function [ gridWeights ] = GridWeights( gridToMesh,centerPoints,grid,rmin )
2  gridWeights=zeros(size(grid,1),size(grid,2),length(centerPoints));
3  for i=1:size(grid,1)
4    for j=1:size(grid,2)
5      sum=0;

```

```

6     for iel=1:length(centerPoints)
7         if (gridToMesh(i,j,iel)>0)
8             sum=sum+max(0,(rmin-distance2D(centerPoints(iel,[1 2]),grid(i,j)))^3);
9             gridWeights(i,j,iel)=max(0,(rmin-distance2D(centerPoints(iel,[1 2]),grid(i,j)))^3);
10            end
11        end
12        gridWeights(i,j,:)=gridWeights(i,j,+)/sum;
13    end
14 end
15 end

1 function [ fakeGridWeights ] = FakeGridWeights( gridToFakeMesh,centerPointsFake,grid,rmin )
2 fakeGridWeights=zeros(size(grid,1),size(grid,2),length(centerPointsFake));
3 for i=1:size(grid,1)
4     for j=1:size(grid,2)
5         sum=0;
6         for iel=1:length(centerPointsFake)
7             if (gridToFakeMesh(i,j,iel)>0)
8                 sum=sum+max(0,(rmin-distance2D(centerPointsFake(iel,[1 2]),grid(i,j)))^3);
9                 fakeGridWeights(i,j,iel)=max(0,(rmin-distance2D(centerPointsFake(iel,[1 2]),grid(i,j)))^3);
10            end
11        end
12        fakeGridWeights(i,j,:)=fakeGridWeights(i,j,+)/sum;
13    end
14 end
15 end

1 function [ Isf,IsfFake ] = ReinitLevelSetInMesh( struc,centerPoints,centerPointsFake,elemSize )
2 strucFake=zeros(length(centerPointsFake));
3 Isf=(~struc.*(Bwdist(struc,strucFake,centerPoints,centerPointsFake,elemSize)-0.5*elemSize))...
4 -(struc.*(Bwdist(struc-1,strucFake-1,centerPoints,centerPointsFake,elemSize)-0.5*elemSize));
5 IsfFake=(~strucFake.*(Bwdist(strucFake,struc,centerPointsFake,centerPoints,elemSize)
-0.5*elemSize))...
6 -(strucFake.*(Bwdist(strucFake-1,struc-1,centerPointsFake,centerPoints,elemSize)-0.5*elemSize));
7 end

1 function [ Isf ] = Reinit(
struc,centerPoints,centerPointsFake,elemSize,gridToMesh,gridToFakeMesh,gridWeights,
fakeGridWeights )
2 [IsfTemp,IsfFake] = ReinitLevelSetInMesh( struc,centerPoints,centerPointsFake,elemSize );
3 Isf=zeros(size(gridToMesh,1),size(gridToMesh,2));
4 for i=1:size(gridToMesh,1)
5     for j=1:size(gridToMesh,2)
6         for iel=1:length(struc)
7             if gridToMesh(i,j,iel)>0
8                 Isf(i,j)=Isf(i,j)+IsfTemp(gridToMesh(i,j,iel))*gridWeights(i,j,iel);
9             end
10        end
11    end
12    for iel=1:length(IsfFake)
13        if gridToFakeMesh(i,j,iel)>0
14            Isf(i,j)=Isf(i,j)+IsfFake(gridToFakeMesh(i,j,iel))*fakeGridWeights(i,j,iel);
15        end
16    end

```

```

17 end
18 end

```

```

1 function [ struc,lsf ] = UpdateStep(
lsf,shapeSens,topSens,gridToMesh,gridWeights,meshToGrid,meshWeights,stepLength,topWeight )
2 shapeSensGrid=zeros(size(gridToMesh,1),size(gridToMesh,2));
3 topSensGrid=zeros(size(gridToMesh,1),size(gridToMesh,2));
4 for i=1:size(gridToMesh,1)
5     for j=1:size(gridToMesh,2)
6         for iel=1:length(shapeSens)
7             if (gridToMesh(i,j,iel)>0)
8                 shapeSensGrid(i,j)=shapeSensGrid(i,j)+shapeSens(gridToMesh(i,j,iel))*gridWeights(i,j,iel);
9                 topSensGrid(i,j)=topSensGrid(i,j)+topSens(gridToMesh(i,j,iel))*gridWeights(i,j,iel);
10            end
11        end
12    end
13 end
14 shapeSensGrid=conv2(shapeSensGrid,1/6*[0 1 0; 1 2 1; 0 1 0],'valid');
15 topSensGrid=conv2(topSensGrid,1/6*[0 1 0; 1 2 1; 0 1 0],'valid');
16
17 [struc,lsf]=Evolve(shapeSensGrid,topSensGrid.*(lsf<0),lsf,stepLength,topWeight,meshToGrid,
meshWeights);
18 end

```

```

1 function [ struc,lsf ] = Evolve( v,g,lsf,stepLength,w,meshToGrid,meshWeights )
2 dt=0.05/max(abs(v(:)));
3 for i=1:20*stepLength
4     dpx=circshift(lsf,[-1,0])-lsf;
5     dmx=lsf-circshift(lsf,[1,0]);
6     dpy=circshift(lsf,[0,-1])-lsf;
7     dmy=lsf-circshift(lsf,[0,1]);
8     %Update Level Set Function using an upwind scheme
9     lsf=lsf-dt*min(v,0).*sqrt(min(dmx,0).^2+max(dpx,0).^2+min(dmy,0).^2+max(dpy,0).^2)...
10        -dt*max(v,0).*sqrt(max(dmx,0).^2+min(dpx,0).^2+max(dmy,0).^2+min(dpy,0).^2)...
11        -dt*w*g;
12 end
13 lsfToStruc=zeros(size(meshToGrid,1),1);
14 for iel=1:size(meshToGrid,1)
15     for i=1:size(meshToGrid,2)
16         for j=1:size(meshToGrid,3)
17             if (meshToGrid(iel,i,j)>0)
18                 lsfToStruc(iel)=lsfToStruc(iel)+lsf(i,j)*meshWeights(iel,i,j);
19             end
20         end
21     end
22 end
23 struc=(lsfToStruc<0);
24 end

```

```

1 function [ U ]=FEALS( Kelements,nodes,elemConn,strucx,fixedDOFs,F )
2 nnodes=size(nodes,1);
3 nel=size(elemConn,1);
4 K=zeros(nnodes*5,nnodes*5);
5 for i=1:nel

```

```
6   eDOFs=ElementDOFs(elemConn(i,:));
7   K(eDOFs,eDOFs)=K(eDOFs,eDOFs)+strucx(i)*Kelements(:,i);
8
9   end
10  allDOFs=[1:5*nnodes];
11  freeDOFs=setdiff(allDOFs,fixedDOFs);
12  U=zeros(5*nnodes,1);
13  U(freeDOFs,1)=K(freeDOFs,freeDOFs)\F(freeDOFs);
14
15  end
```