Πολυτεχνείο Κρήτης

Σχολή Μηχανικών Παραγωγής και Διοίκησης

Παραδείγματα επίλυσης με την μέθοδο των πεπερασμένων στοιχείων σε περιβάλλον Matlab

~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

Διπλωματική Εργασία

Μιχαλέλη Γεωργία - Ασπασία

Επιβλέπων Καθηγητής:
Σταυρουλάκης Γεώργιος, Καθηγητής - Πολυτεχνείου Κρήτης

Χανιά, Ιούλιος 2020

ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική εργασία αναλύονται οι μηχανικές ιδιότητες τριών διαφορετικών παραδειγμάτων με τη χρήση της μεθόδου των πεπερασμένων στοιχείων.. Ενός δικτυώματος ράβδων, ενός πλαισίου δοκών και ενός δίσκου. Σκοπός είναι να δημιουργήσουμε και να παρουσιάσουμε τρία διαφορετικά προγράμματα σε περιβάλλον Matlab που θα αναλύουν τις δομές αυτές με την μέθοδο των πεπερασμένων στοιχείων (FEM analysis). Επιπλέον τα προγράμματα αυτά μπορούν να αποτελέσουν την βάση για την ανάλυση κάθε τέτοιας παρόμοιας δομής αλλάζοντας τα δεδομένα εισόδου κατάλληλα. Το περιβάλλον Matlab είναι εύχρηστο και βολικό για ανάλυση τέτοιων κατασκευών. Σημειώνεται επίσης ότι τα προγράμματα μπορούν να εκτελεστούν και με το ελεύθερα διαθέσιμο λογισμικό octave που είναι συμβατό με το Matlab. Η δημιουργία των προγραμμάτων έχει γίνει με τη βοήθεια της θεωρίας που περιβάλλει τέτοιου είδους κατασκευές αλλά και με την βοήθεια υπάρχουσας βιβλιογραφίας.

Τα παραδείγματα μας είναι τα εξείς:
Α) Ένα δικτύωμα αποτελούμενο από 6 στοιχεία και 5 κόμβους πακτωμένο σε δύο σημεία,
Β) Ενα πλαίσιο αποτελούμενο από 8 στοιχεία και 8 κόμβους και
Γ) Ένα δίσκο όπου στο κέντρο του υπάρχει μία παραλληλόγραμμη οπή.

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία υλοποιήθηκε με τη στήριξη, την βοήθεια και την υπομονή ορισμένων ανθρώπων στους οποίους θα ήθελα να εκφράσω τις θερμότερες ευχαριστίες μου.

Αρχικά, θα ήθελα να ευχαριστήσω τον κ. Γεώργιο Σταυρουλάκη, καθηγητή της σχολής Μηχανικών Παραγωγής και Διοίκησης Πολυτεχνείου Κρήτης, για την υπομονή & την εποπτεία της πορείας της εργασίας όλους αυτούς τους μήνες παρόλο τις δυσκολίες της από απόστασης εγγραφής αλλά και το χρόνο που μου αφιέρωνε το διαστημα που βρισκόμουν στα Χανιά.

Επίσης θα ήθελα να ευχαριστήσω τον φίλο και συνεργάτη Μελανδινό για την στήριξη του και την υπομονή του τους μήνες που χρειάστηκε να απουσιάσω από την εταιρία εργασίας μου για να μεταβώ στα Χανιά προκειμένου να ολοκληρωσω το προγραμματιστικό κομμάτι της διπλωματικής.

Τέλος, η παρούσα διπλωματική εργασία είναι αποκλιστικά αφιερωμένη στους γονείς μου Γιάννη και Δήμητρα, για όλη την αγάπη, τις θυσίες και την υπομονή που έκαναν όλα αυτά τα χρόνια μέχρι να καταφέρω να ολοκληρωσω τις σπουδές μου.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΔΟΜΗ ΕΡΓΑΣΙΑΣ

Η δομή της διπλωματικής εργασίας που περιγράφεται παρακάτω απαρτίζεται από τέσσερα βασικά κεφάλαια. Το πρώτο κεφάλαιο αναφέρεται στις βασικές έννοιες των πεπερασμένων στοιχείων, στα χαρακτηριστικά γνωρίσματα της μεθόδου, στην ευρεία χρήση της στον επιστημονικό κόσμο αλλά και στα πλεονεκτήματα και μειονεκτήματα της. Στο δεύτερο κεφάλαιο περιγράφεται η γλώσσα προγραμματισμού Matlab. Υπάρχει μία μικρή ιστορική αναδρομή των γλωσσών προγραμματισμού και στη συνέχεια επισημαίνεται η σημαντικότητα της στην επίλυση πολύπλοκων προβλημάτων με μικρό υπολογιστικό κόστος. Ενώ τέλος δεν παραλείπουμε να αναφέρουμε τα πλεονεκτήματα και τα μειονεκτήματα της έναντι άλλων γλωσσών προγραμματισμού. Στο τρίτο και προτελευταίο κεφάλαιο περιγράφουμε κάποια ακόμα θεωρητικά στοιχεία όσον αφορά τα στατικά και τα δυναμικά προβλήματα στην μηχανική αλλά και τη θεωρία για τις ιδιοτιμές και ιδιομορφές. Το τέταρτο και τελευταίο κεφάλαιο το έχουμε χωρίσει σε τρεις ενότητες. Κάθε ενότητα αφορά και ένα από τα τρία προγραμματιστικά προβλήματα που καλούμαστε να λύσουμε και να δημιουργήσουμε κώδικα στο υπολογιστικό πρόγραμμα Matlab για την ανάλυση τους. Πρώτα αναλύουμε το παράδειγμα δομής (κατασκευής) πλαισίου δοκών, στην συνέχεια το παράδειγμα δομής ράβδων και τέλος το παράδειγμα δομής δίσκου επίπεδης έντασης με οπή. Σε κάθε ενότητα αρχικά αναφέρουμε κάποια εισαγωγικά στοιχεία που έχουν σχέση με το εκάστοτε παράδειγμα και στην συνέχεια αναγράφουμε των κώδικα που έχουμε δημιουργήσει. Στο τέλος παρουσιάζουμε τα αποτελέσματα της ανάλυσης των προβλημάτων μας αλλά και ένα μικρό συμπέρασμα.

Στο τέλος αυτής της διπλωματικής και μετά την ενότητα της βιβλιογραφιας παραθέτουμε σε παράρτημα τα επιπλέον κομμάτια κώδικα από κάθε παράδειγμα, τις συναντήσεις δηλαδή που συμπληρώνουν τους κωδικές μας.

ΚΕΦΑΛΑΙΟ 1

ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΤΩΝ ΠΕΠΕΡΑΣΜΕΝΩΝ ΣΤΟΙΧΕΙΩΝ

1.1 Γενικά

Η μέθοδος των πεπερασμένων στοιχείων (Finite Elements Method, FEM), αποτελεί ένα εξαιρετικά σημαντικό εργαλείο για την αριθμητική επίλυση πολλών προβλημάτων της μηχανικής και της μαθηματικής φυσικής. Διάφορα τεχνικά προβλήματα περιγράφονται από διαφορικές εξισώσεις και συστήματα, για τις οποίες τις περισσότερες φορές είναι αδύνατο να δοθούν λύσεις εκτός αν πρόκειται για προβλήματα με πολύ απλή μορφή. Έτσι λοιπόν, η ανάγκη επίλυσης προβλημάτων σύνθετης μορφής οδήγησε στην ανάπτυξη προσεγγιστικών μεθόδων επίλυσης, όπως η Μέθοδος Πεπερασμένων Στοιχείων. Πρόκειται για μία προσεγγιστική μέθοδο με αρκετά αξιόπιστα αποτελέσματα με κύριο πλεονέκτημα την δυνατότητα ευρείας εφαρμογής της μεθόδου σε διαφορετικά είδη προβλημάτων όπως: Στερεά μηχανικά, Δυναμικά, Θερμικά, Υδροδυναμικά και Ηλεκτροστατικά προβλήματα τα οποία μπορούν να περιγραφούν από διαφορικές εξισώσεις. Μερικά τέτοια παραδείγματα της καθημερινότητας είναι οι κατασκευές και οι μελέτες αεροσκαφών, πλοίων, αυτοκινήτων, παντός είδους μηχανημάτων, γεφυρών, σηράγγων, πολυκατοικιών, θεμελιώσεων & οδοστρωμάτων. [13]

Τα πεπερασμένα στοιχεία εισάγουν την νέα τεχνολογία και την επανάσταση στην επιστήμη και στην βιομηχανία, υποστηρίζουν την διεξαγωγή εικονικών πειραμάτων και για αυτό τον λόγο διδάσκονται ως βασικό μάθημα σε όλα τα Ανώτατα ιδρύματα, των Ηνωμένων Πολιτειών της Αμερικής, της Ευρώπης και της Ελλάδας. [6]

Το χαρακτηριστικό της μεθόδου των πεπερασμένων στοιχείων είναι η χρήση διδιάστατων και τρισδιάστατων στοιχείων για την προσομοίωση συνεχών μέσων. Μια από τις πρώτες δημοσιεύσεις στις οποίες παρουσιάσθηκε η ιδέα αυτή είναι των Turner, Clough, Martin, και Topp (1956), ορισμένα όμως χαρακτηριστικά της είχαν ήδη περιγραφεί από τους Courant (1943), Hrenikoff (1941), McHenry (1943) και άλλους, μεταξύ αυτών και τον Έλληνα καθηγητή Αργύρη. Ακολούθησαν πολλές δημοσιεύσεις την περίοδο 1954-60. Τα πρώτα πεπερασμένα στοιχεία χρησιμοποιήθηκαν σε προβλήματα επίπεδης εντατικής κατάστασης, αργότερα όμως διατυπώθηκαν στοιχεία και για τρισδιάστατα στερεά, ελάσματα υπό κάμψη, παχιά κελύφη, και άλλες μορφές κατασκευών. Μετά την καθιέρωσή τους στη γραμμική ελαστική περιοχή εφαρμόσθηκαν και σε δυσκολότερα προβλήματα όπως η δυναμική συμπεριφορά, ο λυγισμός και η μη-γραμμική απόκριση και συμπεριφορά του υλικού. Για να επιλυθούν δε προβλήματα με μη-γραμμική συμπεριφορά του υλικού απαιτείται επαναληπτική διαδικασία. Στις αρχές της δεκαετίας του 1960 αναγνωρίσθηκε ότι η μέθοδος αποτελεί συγκεκριμένη μορφή της μεθόδου Ritz, και το 1964 οι Zienkiewicz και Cheung έδειξαν ότι μπορεί να εφαρμοσθεί σε όλα τα προβλήματα πεδίου που έχουν μεταβολική διατύπωση. Για προβλήματα κατασκευών υπάρχει τώρα ένας ικανός αριθμός προγραμμάτων γενικής χρήσης και το γεγονός αυτό σε συνάρτηση με τις δυνατότητες της μεθόδου έχει οδηγήσει στην ραγδαία εξέλιξη και χρήση της τα τελευταία χρόνια.

1.2 Η μέθοδος των πεπερασμένων στοιχείων

Η μέθοδος πεπερασμένων στοιχείων είναι μία προσεγγιστική μέθοδος επίλυσης διαφορικών εξισώσεων που χρησιμοποιούνται κανόνες μητρωικού λογισμικού.

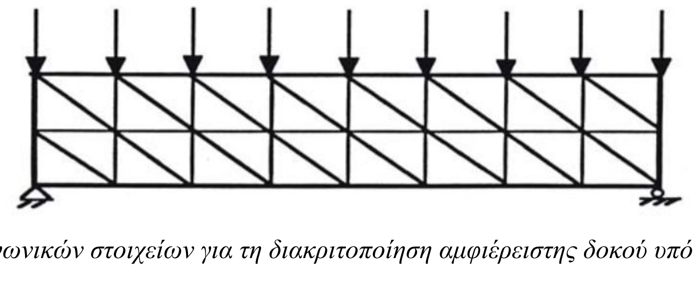
Η αναλυτική λύση των εξισώσεων με τις οποίες περιγράφονται τα διάφορα τεχνικά προβλήματα είναι δυνατή μόνο σε ειδικές περιπτώσεις, όπου οι καταπονήσεις και τα γεωμετρικά σχήματα είναι πάρα πολύ απλά. Ομως, υπήρχε η ανάγκη να λυθούν και πιο σύνθετα προβλήματα και γι'αυτό το λόγο αναπτύχθηκαν διάφορες προσεγγιστικές μέθοδοι.

Για να εφαρμοστεί η μέθοδος των πεπερασμένων στοιχείων απαιτούνται τα εξής στάδια:

A. Εισάγεται η γεωμετρία της κατασκευής σε ένα πρόγραμμα στο οποίο δημιουργείται το μοντέλο.
B. Γίνεται η διακριτοποίηση του μοντέλου σε πεπερασμένα στοιχεία και αφού ετοιμαστεί το πλέγμα επιλέγεται το είδος της επίλυσης και εισάγονται τα επιπλέον δεδομένα που απαιτούνται στο πρόγραμμα. Για παράδειγμα, αν επιλεγεί να λυθεί το μοντέλο σε στατική καταπόνηση θα πρέπει να δοθούν τα δεδομένα για τις δυνάμεις και τις στηρίξεις.
C. Οταν ετοιμαστούν τα δεδομένα για την επίλυση, το πρόγραμμα θα κάνει την επίλυση του προβλήματος.
D. Όταν τελειώσει η επίλυση το πρόγραμμα θα προβάλει τα αποτελέσματα στον μελετητή.

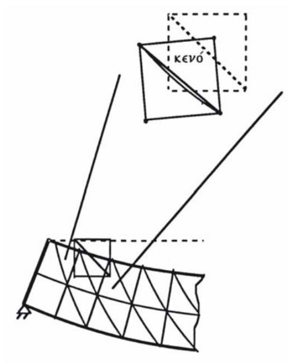
1.3 Βασικές έννοιες της μεθόδου

Η βασική έννοια της μεθόδου των πεπερασμένων στοιχείων είναι, όπως και στη μητρωική ανάλυση, η δυνατότητα προσομοίωσης της πραγματικής κατασκευής με συστατικά στοιχεία τα οποία συνδέονται σε ένα πεπερασμένο αριθμό κόμβων. Η μεθοδολογία αυτή αποτελεί φυσιολογική προσομοίωση των πλαισίων, καθώς αυτά αποτελούνται από δοκούς που είναι συνδεδεμένες στα άκρα τους. Σε μία συνεχή όμως κατασκευή δεν υπάρχουν φυσικοί διαχωρισμοί και συνεπώς απαιτείται να γίνει τεχνητός διαχωρισμός σε στοιχεία, τα οποία να συνδέονται κατά μήκος των άκρων (πλευρών) τους. Τα τεχνητά αυτά στοιχεία, ή πεπερασμένα στοιχεία είναι συνήθως τετράπλευρα ή τριγωνικά και οι κόμβοι συνήθως βρίσκονται στα άκρα. Το Σχήμα 1 δείχνει τον κορμό δοκού υποδιαιρεμένο σε τριγωνικά στοιχεία. Για να γίνει χρήση μητρωικών μεθόδων απαιτείται να προσομοιωθεί η συνεχής κατασκευή με ένα πεπερασμένο αριθμό διακριτών μεταβλητών. Οι μεταβλητές αυτές, για τα προβλήματα της μηχανικής, είναι οι μετατοπίσεις των κόμβων και σε ορισμένες περιπτώσεις και οι παράγωγοί τους. Εάν περιλαμβάνονται και οι παράγωγοι γίνεται λόγος για βαθμούς ελευθερίας αντί για μετατοπίσεις κόμβων. Οι μετατοπίσεις στο εσωτερικό των στοιχείων πρέπει να είναι συμβατές με τις μετατοπίσεις των κόμβων και όλες οι αλληλεπιδράσεις των στοιχείων εκφράζονται σε σχέση με τις κομβικές μετατοπίσεις.



Σχήμα 1 *Χρήση τριγωνικών στοιχείων για τη διακριτοποίηση αμφιέρειστης δοκού υπό καμπτική φόρτιση*

Με αυτό τον τρόπο οι μόνοι άγνωστοι είναι οι μετατοπίσεις στους κόμβους και το πρόβλημα μετατρέπεται από συνεχές σε διακριτό. Παρ' όλο που μπορεί να υπάρχει μεγάλος αριθμός κομβικών μετατοπίσεων ο *αριθμός τους είναι πεπερασμένος. Το πρόβλημα εκφράζεται τότε ως ένα σύνολο (σύστημα) γραμμικών εξισώσεων οι οποίες επιλύονται με αριθμητικές (μητρωικές) μεθόδους.*



Σχήμα 2 *Παραμόρφωση στοιχείων*

Για να επιτευχθεί ακριβής λύση ενός συγκεκριμένου προβλήματος στη διακριτοποιημένη μορφή του, θα πρέπει να ικανοποιούνται οι συνθήκες ισορροπίας και γεωμετρικής συμβιβαστότητας στο εσωτερικό των στοιχείων αλλά και στα σύνορά τους. Οι απαιτήσεις αυτές ανάγονται στην ικανοποίηση τεσσάρων συνθηκών. Ας θεωρηθεί, για παράδειγμα, η συμβιβαστότητα μεταξύ των στοιχείων. Σε μία συνεχή κατασκευή όπως το έλασμα του Σχήματος 2 ισχύει συνέχεια των μετατοπίσεων στα κοινά όρια των στοιχείων. Συνεπώς στο αριθμητικό μοντέλο (πεπερασμένα στοιχεία) δεν επαρκεί να ικανοποιείται η συνθήκη της συνέχειας των μετατοπίσεων στους κόμβους και μόνο. Εάν δηλαδή δεν διατυπωθούν περιορισμοί στις μετατοπίσεις κατά μήκος των ορίων των στοιχείων το θεωρητικό μοντέλο της κατασκευής θα είναι περισσότερο εύκαμπτο επειδή θα δημιουργηθούν κενά, όπως δείχνει το Σχήμα 2. Ένας τρόπος να περιορισθεί το σφάλμα είναι να χρησιμοποιηθούν μικρότερα και περισσότερα στοιχεία διότι έτσι θα δημιουργηθούν περισσότεροι κόμβοι και συνεπώς περισσότερα σημεία στα οποία θα ικανοποιείται η συμβιβαστότητα.

Μία διακριτή προσομοίωση δεν μπορεί όμως να αποδώσει με απόλυτη ακρίβεια την συμπεριφορά ενός συνεχούς μέσου, ανεξαρτήτως του αριθμού των διακριτών μεταβλητών που χρησιμοποιούνται. Υπάρχει δηλαδή πάντοτε ένα σφάλμα, το οποίο όμως μπορεί να περιορισθεί και να γίνει αμελητέο και τοπικό. Δεν είναι συνεπώς δυνατόν να ικανοποιηθούν όλες οι προαναφερθείσες συνθήκες με απόλυτη ακρίβεια, έστω και αν γίνει χρήση μεγάλου αριθμού στοιχείων. Είναι όμως δυνατό, με σωστή επιλογή των ιδιοτήτων των στοιχείων και κατάλληλη διακριτοποίηση, να περιορισθεί το αριθμητικό σφάλμα. Ο προσδιορισμός των ιδιοτήτων των στοιχείων αποτελεί ένα από τα βασικότερα στάδια διατύπωσης μιας λύσης. Θα πρέπει τότε να γίνεται αυτό έτσι ώστε να ικανοποιούνται επαρκώς οι συνθήκες συμβιβαστότητας χωρίς να χρειασθεί να γίνει χρήση υπερβολικά μικρών στοιχείων.

Η συμπεριφορά των στοιχείων καθορίζεται από συναρτήσεις οι οποίες ορίζουν τον τρόπο μεταβολής των τάσεων ή των μετατοπίσεων στο εσωτερικό τους. Με άλλα λόγια, προκαθορίζεται ο τρόπος συμπεριφοράς των διαφόρων μεταβλητών. Το αποτέλεσμα είναι ότι, παρ’όλο που οι συνθήκες ισορροπίας και συμβιβαστότητας ικανοποιούνται μόνο στους κόμβους, η προδιαγεγραμμένη συμπεριφορά στο εσωτερικό κάθε στοιχείου εξασφαλίζει ότι η συμβιβαστότητα ικανοποιείται επαρκώς στο εσωτερικό και στα σύνορά τους.

Συμπεραίνεται λοιπόν ότι απαιτείται προσοχή κατά την υποδιαίρεση (διακριτοποίηση) της κατασκευής, καθώς επίσης και κατά την επιλογή της συνάρτησης που περιγράφει τη συμπεριφορά στο εσωτερικό του κάθε στοιχείου.

Γιατί πρέπει να γνωρίζουμε τα πεπερασμένα στοιχεία;

Η θεωρητική γνώση των πεπερασμένων στοιχείων βοηθά να αναπτύξει ο φοιτητής, ο ερευνητής ή ο μηχανικός της πράξης την αυτοπεποίθηση και την ικανότητα να λύνει, να λειτουργεί και να ερμηνεύει σωστά τα αποτελέσματα που λαμβάνει από ένα πρόγραμμα, και αν χρειασθεί να γράψει το δικό του πρόγραμμα.[14]

1.4 Πλεονεκτήματα Μεθόδου

- Μπορεί εύκολα να χειριστεί πολύ σύνθετη γεωμετρία
- Μπορεί να χειριστεί μια μεγάλη ποικιλία μηχανικών προβλημάτων
- Μπορεί να χειριστεί σύνθετους περιορισμούς
- Μπορεί να χειρίζεται πολύπλοκη φόρτωση

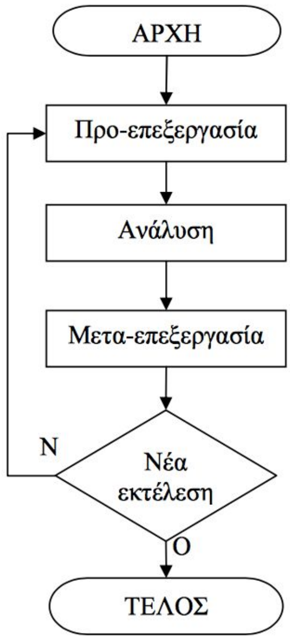
1.5 Μειονεκτήματα Μεθόδου

- Μια γενική λύση κλειστής μορφής, η οποία θα επέτρεπε να εξεταστεί η ανταπόκριση του συστήματος στις αλλαγές σε διάφορες παραμέτρους, δεν παράγεται.
- Αποκτά μόνο "κατά προσέγγιση" λύσεις.
- Εμπεριέχει "εγγενή" σφάλματα.
- Τα λάθη των χρηστών μπορεί να είναι κρίσιμα στο αποτέλεσμα.
- Ως βασικό μειονέκτημα της μεθόδου ,όμως, θεωρούνται οι αυξημένες απαιτήσεις σε υπολογιστική ισχύ, ιδίως όταν αυτή εφαρμόζεται σε πιο σύνθετα προβλήματα με πολύπλοκη γεωμετρία και ιδίως για μη γραμμικά προβλήματα. [13]

1.6 Περι Λογισμικού και Μεθόδου Πεπερασμένων Στοιχείων

Ένα πλήρες υπολογιστικό περιβάλλον για την ανάλυση μίας μηχανολογικής κατασκευής με τη Μέθοδο των Πεπερασμένων Στοιχείων (ΜΠΣ) εμφανίζει τρία διακριτά μέρη (βλ. Σχήμα).

‘Προ-επεξεργασία’ (pre-processing)



Αφορά στην εισαγωγή όλων εκείνων των δεδομένων που απαιτούνται για τη μελέτη της κατασκευής, δηλαδή:

- Γεωμετρική περιγραφή της κατασκευής
- Διακριτοποίηση της κατασκευής
- Δήλωση ιδιοτήτων υλικού της κατασκευής
- Δήλωση στηρίξεων
- Δήλωση φορτίσεων

Η γεωμετρική περιγραφή της κατασκευής επιτυγχάνεται μέσω της σχεδίασης αυτής. Με τον όρο ‘διακριτοποίηση της κατασκευής’, ισοδύναμα με τον όρο ‘δημιουργία πλέγματος’, εννοούμε τη διαίρεση της κατασκευής σε πλήθος στοιχείων με πεπερασμένες γεωμετρικές διαστάσεις (Πεπερασμένα Στοιχεία - ΠΣ). Ο τύπος των ΠΣ αποτελεί επιλογή του χρήστη. Επομένως, η προ-επεξεργασία είναι ουσιαστικά ένα σχεδιαστικό περιβάλλον, στο οποίο προσομοιώνεται η προς μελέτη κατασκευή. Το προϊόν αυτής της προσομοίωσης καλείται ‘μοντέλο’

‘Ανάλυση’ (Analysis)

Σε αυτό το τμήμα λαμβάνει χώρα ο υπολογισμός όλων των ποσοτήτων ενδιαφέροντος, όπως κομβικές μετατοπίσεις, τάσεις, παραμορφώσεις, ιδιοσυχνότητες, κοκ.

‘**Μετά-επεξεργασία’ (post-processing**)

Σε αυτό το τμήμα, παρουσιάζονται τα αποτελέσματα της ανάλυσης. Ο πλέον συνήθης τρόπος παροσυσίασης είναι μέσω χρωματικής απεικόνισης. Αυτό σημαίνει ότι η κατανομή της ποσότητας ενδιαφέροντος, π.χ. τάση, εμφανίζεται ως κατανομή χρωμάτων σε όλη της έκταση της κατασκευής σύμφωνα με μία χρωματική κλίμακα. Επιπροσθέτως, είναι δυνατή η παρουσίαση αποτελεσμάτων είτε με τη μορφή γραφημάτων, είτε με τη μορφή πινάκων είτε ως περιεχόμενο κάποιου αρχείου δεδομένων. Διευκρινίζεται ότι η χρωματική απεικόνιση, αν και δίδει μία γρήγορη και εποπτική εικόνα της κατανομής ενός μεγέθους, είναι δυνατόν να παραπλανήσει. Αυτό οφείλεται στο γεγονός ότι η κατασκευή μίας χρωματικής κατανομής στηρίζεται σε διαδικασίες παρεμβολής (interpolation) μεταξύ αριθμητικών τιμών σε συγκεκριμένα σημεία, οπότε υπάρχει ο κίνδυνος αυτό που απεικονίζεται χρωματικά να διαφέρει σημαντικά από αυτό που περιγράφεται αριθμητικά. [12]

ΚΕΦΑΛΑΙΟ 2

ΠΡΟΓΡΑΜΜΑΤΙΖΟΝΤΑΣ ΣΕ MATLAB

2.1 Η γλώσσα προγραμματισμού MATLAB

Η MATLAB είναι μία υψηλού επιπέδου γλώσσα προγραμματισμού που επιτρέπει την ολοκλήρωση υπολογιστικά απαιτητικών εργασιών όπως C, C++ και Fortran. Η γλώσσα προγραμματισμού MATLAB (το όνομα προήλθε από τις λέξεις Matrix Laboratory - εργαστήρι πινάκων) λειτουργεί ως διερμηνέας εντολών (command interpreter), οι οποίες δίνονται μέσω του παραθύρου εντολών της (MATLAB command window). Η βασική δομή της γλώσσας είναι ο πίνακας. Ολες οι μεταβλητές αποθηκεύονται ως πίνακες. Ακόμα και ο αριθμός θεωρείται ότι είναι ένας πίνακας 1x1.

Οι εντολές αυτές μπορεί να είναι:

1. ορισμοί μεταβλητών και πράξεις
2. κλήση ενσωματωμένων συναρτήσεων της MATLAB και των εγκατεστημένων εργαλειοθηκών της (toolboxes)
3. κλήση συναρτήσεων (functions) ή αρχείων εντολών MATLAB (scripts) που κατασκευάζονται από τους χρήστες με τη μορφή m-file

Η ευρεία χρήση της MATLAB οφείλεται σε μεγάλο βαθμό στην επεκτασιμότητα της μέσω των διάφορων εργαλειοθηκών, κάθε μια από τις οποίες περιέχει ένα αριθμό συναρτήσεων για ένα συγκεκριμένο αντικείμενο.

2.2 Ιστορία της Matlab

Η Matlab αποτελεί την φυσική εξέλιξη προγραμμάτων σε Fortran για την επίλυση γραμμικών προβλημάτων (προγραμμα Linpack) και προβλημάτων εύρεσης ιδιοτιμών (Eispack).

Αναπτύχθηκε αρχικά από τον Cleve Moler τη δεκαετία του 1970 για την επίλυση των παραπάνω προβλημάτων χωρίς την γνώση Fortran από τον χρήστη.

Επίσης διατυπώθηκε η ανάγκη ύπαρξης περιβάλλοντος και σχεδιασμού διαγραμμάτων. Ως αποτέλεσμα, την δεκαετία του 1980 το πρόγραμμα γράφτηκε ξανά σε γλώσσα C με περισσότερες λειτουργίες, βιβλιοθήκες, δυνατότητες γραφικών παραστάσεων και απεικονίσεων.

Το 1984 ιδρύεται η εταιρία MathWorks η οποία είναι υπεύθυνη για την διάθεση στο εμπόριο, την εξέλιξη και την υποστήριξη του λογισμικού μέχρι σήμερα. [7]

2.3 Περιβάλλον Matlab

Το περιβάλλον του προορίζεται για την ανάπτυξη αλγορίθμων, επεξεργασία, ανάλυση και απεικόνιση δεδομένων, αριθμητικές μεθόδους επίλυσης προβλημάτων και μοντελοποίησης συστημάτων. Τα βασικότερα μέρη του περιβάλλοντος της Matlab είναι τα εξής.

- **Command Window** (Παράθυρο Εντολών): Στο παράθυρο αυτό εισάγονται οι εντολές της Matlab, καλούνται συναρτήσεις και M-files, καθώς τυπώνονται και τα αποτελέσματα.
- **Command History** (Ιστορικό Εντολών): Στο Command History αποθηκεύονται οι εντολές που εισάγουμε στο Command Window.
- **Workspace** (Χώρος Εργασίας): Στο Workspace αποθηκεύονται οι μεταβλητές και οι πίνακες που έχουμε δημιουργήσει. Ο χώρος αυτός βρίσκεται στην μνήμη του υπολογιστή, και μέσα σ'αυτόν εκτελούνται οι πράξεις και ταυτόχρονα διατηρούνται όλα τα αποτελέσματα τους, με σκοπό να μπορούν να χρησιμοποιηθούν σε επόμενες πράξεις.
- **Current Folder** (τρέχων κατάλογος): Εδώ είναι το σημείο αναφοράς των αρχείων της Matlab, αν ο χρήστης θέλει να καλέσει μία συνάρτηση ή ένα M-files θα πρέπει να βρίσκεται στον τρέχοντα κατάλογο.

2.4 Πλεονεκτήματα γλώσσας προγραμματισμού Matlab

1. Η εκμάθηση της γλώσσας προγραμματισμού Matlab είναι σχετικά εύκολη σε σύγκριση με άλλες γλώσσες προγραμματισμού.
2. Ο κώδικας της Matlab είναι σχεδιασμένος να δίνει γρήγορα αποτελέσματα όταν πραγματοποιούνται πράξεις πινάκων.
3. Είναι γλώσσα προγραμματισμού για ανάπτυξη εφαρμογών και ταυτόχρονα λογισμικό υλοποίησης επιστημονικών υπολογισμών.
4. Εύκολος εντοπισμός και διόρθωση λαθών.
5. Φιλικό περιβάλλον επικοινωνίας με τον χρήστη.

2.5 Μειονεκτήματα γλώσσας προγραμματισμού Matlab

1. Η Matlab δεν αποτελεί μία γλώσσα προγραμματισμού γενικής χρήσης.
2. Η Matlab είναι σχεδιασμένη για επιστημονικούς υπολογισμούς και δεν είναι κατάλληλη για κάποιες άλλες λειτουργίες (π.χ ανάγνωση κειμένου)
3. Οι αναπτυσσόμενες εφαρμογές υστερούν σε απόδοση από την άποψη του χρόνου εκτέλεσης σε σχέση με τις αντίστοιχες του αναπτύσσονται στις κλασικές γλώσσες προγραμματισμού (C, C++).
4. Η Matlab είναι μία μεταφραζόμενη προγραμματιστική γλώσσα που απαιτεί τον μεταφραστή(τη γλώσσα Matlab δηλαδή) για να εκτελέσει κάποιο πρόγραμμα. Δεν παράγει εκτελέσιμα αρχεία. (π.χ .exe).

ΚΕΦΑΛΑΙΟ 3

ΔΥΝΑΜΙΚΑ - ΣΤΑΤΙΚΑ ΠΡΟΒΛΗΜΑΤΑ

ΕΙΣΑΓΩΓΗ

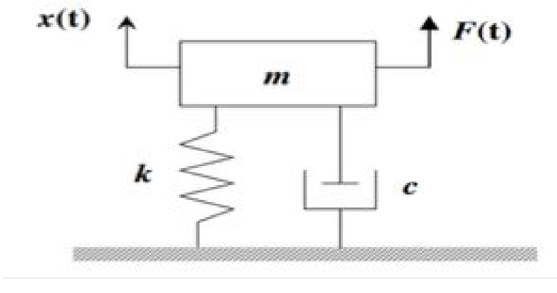
Διάφοροι τύποι στατικών ή δυναμικών προβλημάτων επαφής αντιμετωπίζονται ευρέως σε πολλούς τομείς, όπως μηχανική, πολιτική και υδραυλική μηχανική. Η κύρια δυσκολία προσομοίωσης αυτών των προβλημάτων έγκειται στην εγγενή και ισχυρή μη γραμμικότητα λόγω του γεγονότος ότι τόσο η περιοχή επαφής όσο και η κατανομή των δυνάμεων επαφής είναι άγνωστες πριν από την ανάλυση.
Όπως είναι γνωστό, τα φορτία που δέχονται οι κατασκευές χωρίζονται σε δύο κυρίες κατηγορίες, στατικά και δυναμικά ή αλλιώς σταθερά ή μεταβαλλόμενα με το χρόνο.[5]

3.1 Τι είναι ένα Δυναμικό Σύστημα

Ως δυναμικό σύστημα ορίζεται κάθε σύστημα φυσικό, χημικό, βιολογικό, οικονομικό, οικολογικό, κλπ, που εξελίσσεται με την πάροδο του χρόνου. Η κατάσταση ενός δυναμικού συστήματος χαρακτηρίζεται, κάθε χρονική στιγμή, από ένα πεπερασμένο ή άπειρο πλήθος παραμέτρων και το σύνολο των εφικτών καταστάσεων ορίζει τον πεπερασμένης ή άπειρης διάστασης χώρο καταστάσεων[1].

3.2 Εξισώσεις Κίνησης

Ο τρόπος με τον οποίο ένα σύστημα εκτελεί ανεξάρτητες κινήσεις ορίζεται από τους βαθμούς ελευθερίας του [2]. Σε αυτή την παράγραφο θα αναλύσουμε ένα δυναμικό σύστημα ταλάντωσης, ενός βαθμού ελευθερίας, το οποίο αποτελείται από ένα ελατήριο, μία μάζα και έναν αποσβεστήρα (Σχήμα 3).



Σχήμα 3: Δυναμικό Σύστημα ενός Βαθμού Ελευθερίας

Στο παραπάνω σχήμα χρησιμοποιήθηκε ο εξής συμβολισμός:
m: μάζα
k: ελατήριο
c: αποσβεστήρας
x(t): απόκριση συστήματος (μετατόπιση), χρονικά μεταβαλλόμενη
F(t): εξωτερική διέγερση συστήματος (δύναμη), χρονικά μεταβαλλόμενη
Θεωρούμε πως το κάτω άκρο των στοιχείων ελαστικότητας και απόσβεσης (k, cαντίστοιχα) είναι προσδεδεμένο σε ακίνητη και απαραμόρφωτη οριζόντια επιφάνεια και το άνω άκρο τους συνδεδεμένο με τη μάζα m.

Το παραπάνω παράδειγμα δυναμικού συστήματος χαρακτηρίζεται ως σύστημα ενός Βαθμού Ελευθερίας, καθώς το μοναδικό εμφανιζόμενο κινηματικό μέγεθος του συστήματος είναι η κοινή μετατόπιση του κέντρου της μάζας m και των άνω άκρων των στοιχείων k και c .

Η βαρύτητα αποτελεί μία μόνιμη, στατική φόρτιση που δεν λαμβάνεται υπόψη, γιατί ως αρχή της μετατόπισης θέτουμε την, παραμορφωμένη από το βάρος, κατάσταση ισορροπίας του συστήματος.

Οι φυσικές ιδιότητες ενός δυναμικού συστήματος χαρακτηρίζονται από τα φυσικά του στοιχεία. Στο συγκεκριμένο σύστημα ορίζονται τρία φυσικά στοιχεία (μάζα, απόσβεση, δυσκαμψία).

Η μάζα m χαρακτηρίζει την αδράνεια του συστήματος και έχει ως αποτέλεσμα την ανάπτυξη αδρανειακών δυνάμεων, σύμφωνα με την εξίσωση:

$$Fm = m\ddot{x} \qquad (1)$$

Σύμφωνα με την παραπάνω εξίσωση, η αδρανειακή δύναμη μίας μάζας m είναι ανάλογη της επιτάχυνσης της, με σταθερά αναλογίας την ίδια τη μάζα m .

Απόσβεση ονομάζεται η μείωση του πλάτους ταλάντωσης, λόγω τριβής. Το φαινόμενο αυτό οφείλεται στην απώλεια ενέργειας από το ταλαντευόμενο σύστημα προς το περιβάλλον. Ορίζεται από την εξίσωση:

$$Fc = c\dot{x} \qquad (2)$$

Όπου c η σταθερά απόσβεσης. Η εξίσωση (2) δηλώνει ότι η δύναμη απόσβεσης είναι ανάλογη της σχετικής ταχύτητας $\dot{x}$, με σταθερά αναλογίας την ποσότητα c.

Η σταθερά ελατηρίου k εκφράζει την σκληρότητα ενός ελατηρίου (στοιχείο ελαστικότητας – παραμορφωσιμότητας) και εξαρτάται από τα φυσικά χαρακτηριστικά του (μήκος, πάχος σύρματος, διάμετρο σπειρών, υλικό και θερμοκρασία, απόσταση μεταξύ σπειρών).

Μεταβολή του μήκους του ελατηρίου κατά απόσταση x οδηγεί στην εμφάνιση ελαστικής δύναμης, η οποία είναι ανάλογη της μετατόπισης x με σταθερά αναλογίας την ποσότητα k, και περιγράφεται από την εξίσωση 3.

$$Fk = kx \qquad (3)$$

Στον πίνακα που ακολουθεί συνοψίζονται τα στοιχεία που μελετήσαμε μέχρι τώρα.

Στοιχείο	**Φυσική Σημασία**	**Αναπτυσσόμενες Δυνάμεις**	**Εξίσωση**
m	Μάζα Σώματος	Αδρανειακές	$F_m = m\ddot{x}$
c	Απόσβεση Ενέργειας	Απόσβεσης	$F_c = c\dot{x}$
k	Παραμορφωσιμότητα	Ελαστικές	$F_k = kx$

Πίνακας 1: Φυσικά στοιχεία συστήματος

Τα τρία στοιχεία του πίνακα είναι συνδεδεμένα μεταξύ τους από τη δυναμική ισορροπία του συστήματος και το σύνολο των εσωτερικών δυνάμεων θα είναι αλγεβρικά ίσο προς την εξωτερική διέγερση (ασκούμενη δύναμη), σύμφωνα με την εξίσωση ισορροπίας:

$$Fk+Fc+Fm = f(t) \qquad (4)$$

Αντικαθιστώντας στην εξίσωση (4) τις εσωτερικές δυνάμεις με τις εξισώσεις (1), (2), (3) προκύπτει:

$$m\ddot{x}+c\dot{x}+kx= f(t) \qquad (5)$$

Η παραπάνω σχέση είναι μία Γραμμική Διαφορική Εξίσωση 2ης τάξης, η οποία περιγράφει την κίνηση του συστήματος. Επιλύοντας την βλέπουμε πώς θα αποκριθεί το σύστημά μας.

3.3 Ιδιομορφική Ανάλυση

Στην ενότητα που ακολουθεί θα γίνει αναλυτική επισκόπηση των Ιδιομορφών και Ιδιοσυχνοτήτων ενός δυναμικού συστήματος. Πρόκειται για δύο έννοιες με αρκετά σημαντικό ρόλο στην δυναμική ανάλυση των κατασκευών, δηλαδή στην εσωτερική τους ταλάντωση ή συμπεριφορά, που πραγματοποιείται χωρίς την επιβολή εξωτερικών δυνάμεων. Σε ένα δυναμικό σύστημα, ο αριθμός των ιδιομορφών και των ιδιοσυχνοτήτων είναι ίσος με τον βαθμό ελευθερίας του. Ο υπολογισμός τους γίνεται με τη βοήθεια των μητρώων μάζας και δυσκαμψίας της κατασκευής [3].

3.3.1 Ιδιοσυχνότητες – Ιδιοτιμές

Με τον όρο Ιδιοσυχνότητα εννοούμε τη συχνότητα εκείνη για την οποία το σύστημα απορροφά τη μέγιστη ενέργεια και βρίσκεται σε κατάσταση συντονισμού. Συμβολίζεται με ω και χαρακτηρίζει την ταλάντωση ενός συστήματος στο οποίο ασκούνται μόνο συντηρητικές δυνάμεις (δυνάμεις των οποίων το έργο είναι μηδενικό).[3]

Η μονάδα μέτρησης της κυκλικής ιδιοσυχνότητας είναι (rad/sec) και για μονοβάθμια συστήματα ή αλλιώς ταλαντωτές δίνεται από την παρακάτω σχέση:

$$\omega 2 = \lambda = k/m \qquad (6)$$

Επίσης, η συχνότητα του συστήματος μετρούμενη σε (Hz) δίνεται από τη σχέση:

$$f = \omega/2\pi$$

Οι ιδιοτιμές υπολογίζονται από τα μητρώα μάζας m και δυσκαμψίας k, όπως παρουσιάζεται παρακάτω (σε μορφή πίνακα, matrix equation):

$$\mathrm{M}\ddot{u}(t) - \mathrm{K}u(t) = 0 \qquad (7)$$

Παρατηρούμε ότι η εξίσωση είναι της μορφής *u(t) = af(t)*, όπου:

a: το άγνωστο και ανεξαρτήτου χρόνου διάνυσμα μετακινήσεων,*f(t)*: η κοινή – άγνωστη χρονική συνάρτηση όλων των μετακινήσεων

Από τις εξισώσεις (6), (7) προκύπτει:

$$K - \lambda M = 0 \qquad (8)$$

Και η προφανής λύση αυτής η ορίζουσα:

$$det(K - \lambda M) = 0$$

Οι λύσεις της ορίζουσας είναι οι ζητούμενες ιδιοτιμές λ.

Σε κάθε μία ιδιοτιμή που υπολογίσαμε αντιστοιχεί ένας πίνακας μετακινήσεων ai,j . Το a καλείται ιδιοδιάνυσμα του προβλήματος της ιδιοτιμής. Για τον υπολογισμό του a γίνεται κανονικοποίηση από την οποία προκύπτουν τα ιδιοδιανύσματα φi .

$\varphi i = ciai$ όπου ci : αυθαίρετη σταθερά.

Υποθέτουμε ότι το σύστημα εκτελεί απλή αρμονική ταλάντωση με διάνυσμα μετακινήσεων, με ημιτονοειδή μορφή, σε σχέση με το χρόνο:

$$U = \varphi sin\omega t$$

Όπου ω: η γωνιακή συχνότητα, φ: τα ιδιοδιανύσματα

Από τις σχέσεις (6), (8) προκύπτει:

$$(K - \lambda M)\varphi = 0 \qquad (9)$$

Λύση της παραπάνω εξίσωσης είναι η προφανής φ = 0 ή η ορίζουσα ίση με το μηδέν:

$$det(K - \lambda M) = 0$$

Με λύσεις τις $\omega i2 = \lambda i$, όπου i ο αριθμός των ιδιοτιμών και των βαθμών ελευθερίας του συστήματος.[3]

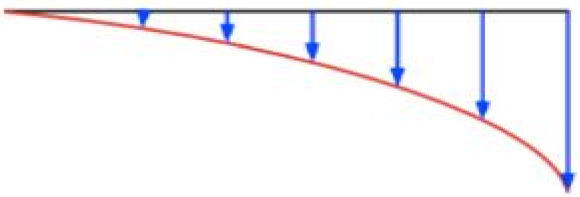
3.3.2 Ιδιοδιανύσματα – Ιδιομορφές

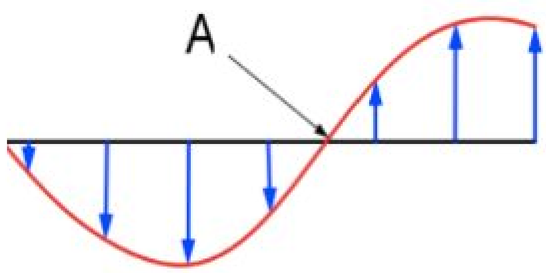
Τα ιδιοδιανύσμαα είναι μία έννοια με την οποία βλέπουμε τον τρόπο ταλάντωσης ενός μηχανικού συστήματος. Μέσω αυτών λαμβάνουμε χρήσιμες πληροφορίες σχετικά με την παραμόρφωση και την μετατόπιση του συστήματος. Συμβολίζονται με φi, ενώ ο υπολογισμός τους γίνεται με τη βοήθεια της εξίσωσης (9):

$$(-\omega 2M+K)\varphi = 0$$

Όπου:
M: το μητρώο μάζας,
K: το μητρώο δυσκαμψίας, ω: η κυκλική ιδιοσυχνότητα, φ: τα ιδιοδιανύσματα

Στις τεχνολογικές εφαρμογές συνήθως υπολογίζεται ένας μικρός αριθμός πρώτων ιδιομορφών (5-20) [3]. Αυτό συμβαίνει, πρώτον γιατί οι δυνάμεις διέγερσης στη φύση διεγείρουν μόνο τις πρώτες ιδιοσυχνότητες της κατασκευής (χαμηλές) και δεύτερον επειδή οι αδρανειακές δυνάμεις έχουν μικρότερη επιρροή στη δυναμική του συστήματος όσο αυξάνεται η συχνότητα διέγερσης. Στο ακόλουθο σχήμα παρουσιάζονται, για τυχαίο παράδειγμα, οι πρώτες δύο (2) ιδιομορφές μίας απλής δοκού.





Σχήμα 4: *Πρώτη ιδιομορφή δοκού*

Σχήμα 5: *Δεύτερη Ιδιομορφή δοκού*

Η ιδιομορφική ανάλυση είναι ιδιαίτερα χρήσιμη καθώς μας δίνει χρήσιμες πληροφορίες για την συμπεριφορά του δυναμικού συστήματος. Εκτός από τη μορφή της ταλάντωσης μας δείχνει και κάποια κρίσιμα σημεία (δεσμούς) της κατασκευής. Για παράδειγμα αν διεγείρουμε στο σημείο Α σύμφωνα με τον δεύτερο τρόπο ταλάντωσης, δηλαδή σύμφωνα με την δεύτερη ιδιομορφή, δεν θα πάρουμε καμία πληροφορία για την ταλάντωση του συστήματος καθώς το σημείο αυτό είναι σημείο δεσμού [4].

ΚΕΦΑΛΑΙΟ 4

ΑΝΑΛΥΣΗ ΣΥΣΤΗΜΑΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ MATLAB

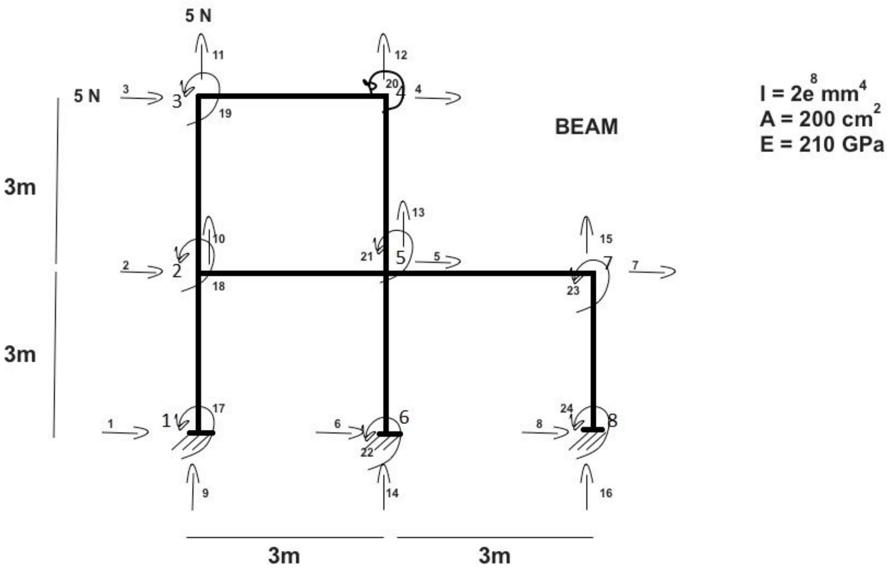
4.1 Δομές Πλαισίου με Δοκούς

Με την ταχεία ανάπτυξη του αστικού πληθυσμού τόσο στις αναπτυσσόμενες όσο και στις βιομηχανικές χώρες, οι κατασκευές πλαισίου έχουν γίνει ένα υλικό επιλογής για κατοικίες και εμπορικές κατασκευές λόγω της περιορισμένης διαθέσιμης γης. Οι δομές πλαισίων είναι οι δομές που έχουν το συνδυασμό δέσμης, κολώνας και πλάκας για να αντιστέκονται στα πλευρικά και στα φορτία βαρύτητας. Αυτές οι δομές συνήθως χρησιμοποιούνται για να υπερνικήσουν τα μεγάλα φορτία τις στιγμές που αναπτύσσονται λόγω της εφαρμοζόμενης φόρτωσης. Μια δομή πλαισίου έχει τρεις βαθμούς ελευθερίας σε κάθε κόμβο. Επομένως, ο συνολικός αριθμός βαθμών ελευθερίας είναι τριπλάσιος του αριθμού των κόμβων. Όταν πρόκειται να αναλύσουμε το πλαίσιο για τις εφαρμοζόμενες φορτίσεις, η αναλυτική επίλυση του, μπορεί να είναι κουραστική και ανακριβής, ειδικά όταν πρόκειται για μεγάλο αριθμός βαθμών ελευθερίας που δρουν στη δομή του πλαισίου. Για την επίλυση ενός τέτοιου πλαισίου πολλών βαθμών ελευθερίας, η μέθοδος ανάλυσης πεπερασμένων στοιχείων μπορεί να είναι μία από τις πλέον κατάλληλες μεθόδους που χρησιμοποιούνται ευρέως. Η ευκολία υπολογισμού περίπλοκων βημάτων για την εύρεση της λύσης, είναι ο λόγος που χρησιμοποιείται τόσο πολύ στις μεθόδους ανάλυσης αλλά και σε διαφορετικές γλώσσες προγραμματισμού. Η μέθοδος των πεπερασμένων στοιχείων είναι ένα εργαλείο για την επίλυση μονοδιάστατων, δισδιάστατων και τρισδιάστατων δομών με προσέγγιση, αντί για επίλυση πολύπλοκων διαφορικών εξισώσεων.

4.1.1 Παράδειγμα Πλαισίου Δοκών - Κώδικας

Εμείς στο πρώτο μας παράδειγμα καλούμαστε να επιλύσουμε ένα πλαίσιο δοκών όπως αυτό εμφανίζεται σχηματικά παρακάτω μέσω δημιουργίας κώδικα στο υπολογιστικό σύστημα της MATLAB. Η πρώτη διαδικασία που πρέπει να λάβει χώρα είναι η διακριτοποίηση, δηλαδή ο διαχωρισμός της δομής του πλαισιου σε αρκετά μικρά στοιχεία που συνδέονται με κόμβους. Όλα τα στοιχεία και οι κόμβοι αριθμούνται για να δημιουργήσουν μια μήτρα συνδεσιμότητας. Είναι σημαντικό να θυμόμαστε ότι η σειρά των κόμβων και των στοιχείων αριθμείται με ακρίβεια, καθώς επηρεάζει τον χρόνο υπολογισμού. Πρόκειται για δύο διαστάσεων πλαίσιο υπό στατική φόρτιση. Κάθε κόμβος έχει τρεις βαθμούς ελευθερίας, δύο ως προς τους αξονες x'x & y'y και ένα περιστροφικό βαθμό ελευθερίας. Προκειμένου να λυθεί το πρόβλημα μας έπρεπε να γίνει αριθμοποίηση όλων των βαθμών ελευθερίας του πλαισίου. Η αριθμοποίηση ξεκίνησε από τις οριζόντιες αντιδράσεις , στη συνέχεια αριθμοποιήσαμε όλες τις κάθετες και τέλος όλες τις περιστροφικές οπως ακριβώς εμφανίζονται παρακάτω στο σχήμα αναπαράστασης.

Αρχικά καλούμαστε να δηλώσουμε όλες τις μεταβλητές μας: E,I,A,L,numberElements, nodeCoordinates, elementNodes, numberNodes σύμφωνα με το παρακάνω σχήμα. Το υλικό των δοκών υποθέτουμε ότι είναι χάλυβας με συντελεστή ελαστικότητας E 210GPa. Η δομή μας έχει συνολικό ύψος 6m και συνολικό πλάτος 6m. Ενώ ασκούνται δυνάμεις στις αντιδράσεις στις θέσεις 3 & 11 από 5N η κάθε μία όπως φαίνεται και στο παρακάτω σχήμα. Συνολικά έχουμε 8 στοιχεία (numberElements).
Οι βαθμοί ελευθερίας στους κόμβους συγκράτησης καταχωρούνται στην μεταβλητή "prescribedDof", ενώ οι υπόλοιποι δηλώνονται ως "activeDof." Προετοιμάζουμε όλα τα στοιχεία εισόδου που είναι απαραίτητα για την ανάλυση του πλαισίου δοκών όπως τα έχουμε ήδη αναφέρει σε ένα αρχείο m.file το οποίο είναι απαραίτητο για την επεξεργασία και την ανάλυση.



Σχήμα 6 *: Πρότυπο πλαίσιο δοκών - Παράδειγμα προς επίλυση*

Κώδικας - Matlab

```
% example 2D beam structure
clc;
clear all;
%E: modulus of elasticity - Μεταβλητή Ελαστικότητας
%I:  second moment of area - Στιγμιαία Αδράνεια
%L: length of bar - Μήκος δοκού
E=210000;
A=200;
I=2e8;
EA=E*A;
EI=E*I;
%generation of coordinates and connectivities - Ορισμός συντεταγμένων & συνδεσιμότητας
numberElements=8;
nodeCoordinates=[0 0;0 3000;0 6000;3000 6000;3000 3000;3000 0;6000 3000;6000 0];
xx=nodeCoordinates;
elementNodes(1,1)=1;
elementNodes(1,2)=2;
elementNodes(2,1)=2;
elementNodes(2,2)=3;
elementNodes(3,1)=3;
elementNodes(3,2)=4;
elementNodes(4,1)=4;
elementNodes(4,2)=5;
elementNodes(5,1)=5;
elementNodes(5,2)=2;
elementNodes(6,1)=6;
elementNodes(6,2)=5;
```

```
elementNodes(7,1)=7;
elementNodes(7,2)=5;
elementNodes(8,1)=8;
elementNodes(8,2)=7;
numberNodes=size(nodeCoordinates,1);
xx=nodeCoordinates(:,1);
yy=nodeCoordinates(:,2);
%for structure:
%displacements: displacement vector
%force: force vector
%stiffness: stiffness matrix
%GDof: global number of degrees of freedom
GDof=3*numberNodes;
U=zeros(GDof,1);
force=zeros(GDof,1);

%force vector
force(3)=5000;
force(11)=-5000;

%stiffness matrix
[stiffness]=formStiffness2Dframe(GDof,numberElements,elementNodes,numberNodes,xx,yy,EI,EA);

%boundary conditions and solutions
prescribedDof=[1 6 8 9 14 16 17 22 24]';

%solution
displacements=solution(GDof,prescribedDof,stiffness,force);

%output displacements/reactions
outputDisplacementsReactions(displacements,stiffness,GDof,prescribedDof)

%drawing mesh and deformed shape
 clf
 U=displacements;
 drawingMesh(nodeCoordinates+500*[U(1:numberNodes)...
    U(numberNodes+1:2*numberNodes)],elementNodes,'L2','k.-');
 drawingMesh(nodeCoordinates,elementNodes,'L2','k--');
```

4.1.2 Αποτελέσματα πλαισίου δοκού

Αφού έγινε η διαδικασία της επεξεργασίας από το πρόγραμμα, υπολογίστηκαν όλες οι μήτρες ακαμψίας των μελών που οδήγησαν στον υπολογισμό της συνολικής ακαμψίας.(Όλα τα δεδομένα εισόδου λαμβάνονται σε μονάδες N και mm για επεξεργασία.

Displacements =

Node Number	Displacements (mm)	
1	**X**	0
	Y	0
2	**X**	0.2582
	Y	-0.0292
3	**X**	0.7182
	Y	-0.2105
4	**X**	0.5368
	Y	-0.3541
5	**X**	0.1903
	Y	-0.1782
6	**X**	0
	Y	0
7	**X**	0.0805
	Y	-0.1497
8	**X**	0
	Y	0

TABLE I: OF DISPLACEMENTS

Node Number	Rotational displacements (radian)
1	0
2	-0.0001
3	-0.0001
4	-0.0001
5	-0.0001
6	0

7	0.000
8	0

TABLE II: OF ROTATIONAL MOMENTS

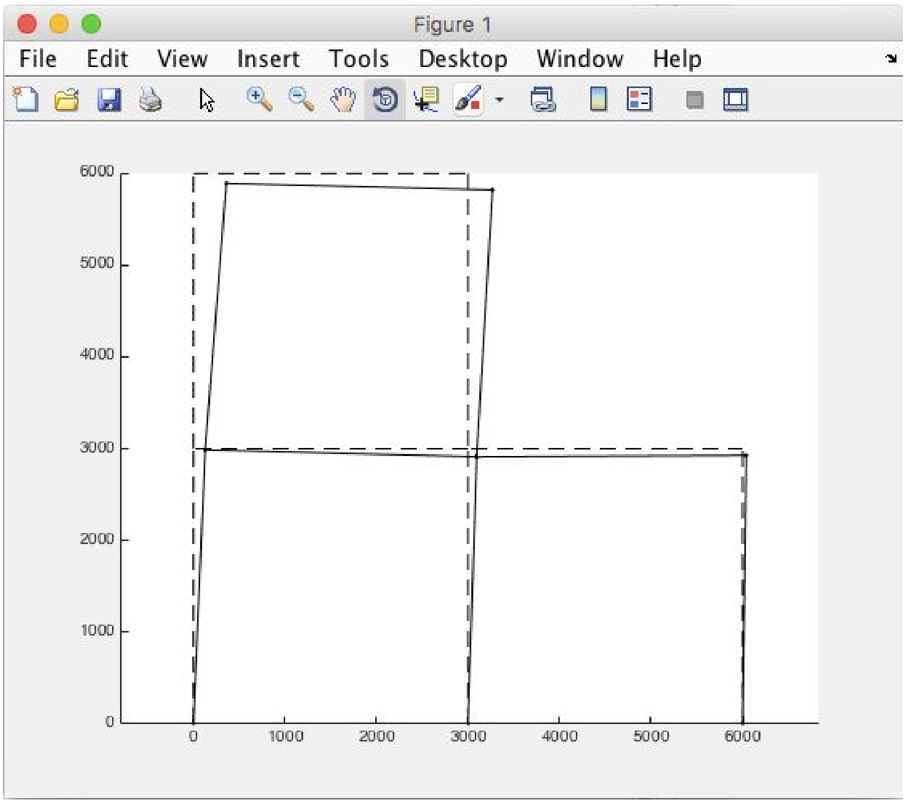
Ο παρακάτω πίνακας δείχνει τις αντιδράσεις από το αναπτυγμένο πρόγραμμα Matlab. Όλες οι οριζόντιες και κάθετες αντιδράσεις βρίσκονται σε Newton (N) και οι τιμές περιστροφής είναι σε ακτίνια.

reactions ans =

Node Number		Reactions
1	Horizontal	15000
	Vertical	4000
	Rotational	39193000
2	Horizontal	-20000
	Vertical	25000
	Rotational	37290000
3	Horizontal	-15000
	Vertical	21000
	Rotational	22893000

TABLE III: OF NODAL REACTIONS

Απεικόνιση ανάλυσης σχηματικά:



Σχήμα 7: *Παραμόρφωση πλαισίου δοκών - Σχήμα επίλυσης*

4.1.3 Συμπεράσματα

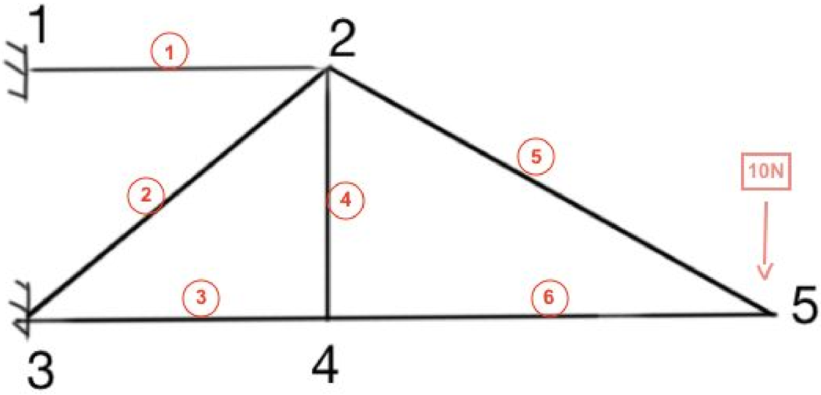
Το πρόγραμμα Matlab είναι μια βολική πλατφόρμα για την ανάπτυξη υπολογιστικών αλγορίθμων, που όμως βοηθάει και στην εικονική/σχηματική απεικόνισή τους. Σκοπός μας ήταν να αναπτύξουμε ένα πρόγραμμα ανάλυσης της δομής πλαισίου χρησιμοποιώντας FEM στο Matlab. Μελετώντας την θεωρία αλλά και την υπάρχουσα βιβλιογραφία υλοποιήθηκε ο παραπάνω κώδικας ο οποίος είναι ευέλικτος και προσαρμόζεται ανάλογα με τα δεδομένα που του δίνονται κάθε φορά για να αναλύει τέτοιου είδους δομές πλαισίου δοκών. Αλλάζοντας τα δεδομένα εισόδου μπορούμε να συγκρίνουμε διαφορετικές δομές και να έχουμε καλύτερη εικόνα για την μηχανικη δράση και αντοχή τέτοιων κατασκευών.

4.2 Δομές Δικτυωμάτων Ράβδου

Τα Δικτυώματα όταν ξεκίνησαν να εφαρμόζονται, κατασκευάζονταν από Ξυλεία, αφού ήταν το μόνο φτηνό υλικό με ικανοποιητική Εφελκυστική Αντοχή, τουλάχιστον μέχρι που εμφανίστηκαν νέες μεταλλουργικές μέθοδοι και έπεσε η τιμή του Σιδήρου. Κατασκευαζόταν με Ξύλινα διαγώνια Μέλη και Μεταλλικά κατακόρυφα. Σε αρχαία κείμενα και τιμητικές στήλες, αναφέρεται η κατασκευή Ξύλινων Δικτυωτών Γεφυρών στη Ρωμαϊκή Αυτοκρατορία, συνήθως σαν Στρατιωτικές Γέφυρες, χτιζόμενες εύκολα και γρήγορα, κατα την προέλαση των λεγεώνων στις εκστρατείες, αν και δεν επιτεύχθηκαν ιδαιτέρα μεγάλα ανοίγματα. Ένα από τα καλύτερα σχέδια Δικτυώματος με μεγάλο άνοιγμα, αναπτύχθηκε από τον **Theodore Burr**, από το Torringtonστο Connecticut, και βασίστηκε σε σχέδια του Palladio. Ένα Δικτύωμα που ενισχύεται από ένα Τόξο, σχέδιο που χρησιμοποιήθηκε ευρέως στις Σκεπαατές Γέφυρες των Ηνωμένων Πολιτειών. [9]

4.2.1 Παράδειγμα Δικτυώματος Ράβδου - Κώδικας

Το δικτύωμα αποτελείται από ράβδους οι οποίοι συνδέονται μεταξύ τους με αρθρώσεις (κόμβους). Οι εξωτερικές δυνάμεις ασκούνται στους κόμβους του δικτυώματος. Το δικτύωμα του προβλήματος μας αποτελείται από 6 ράβδους και 5 κόμβους όπως αναπαρίσταται παρακάτω. Κατά την μοντελοποίηση του δικτυώματος, η ράβδος αντικαθίσταται με ένα πεπερασμένο στοιχείο. Στην ράβδο θεωρούμε ότι ασκούνται μόνο αξονικές δυνάμεις, οι οποίες είναι εφεκλιστικές ή θλιπτικές. Ενώ τέλος πρέπει να αναφέρουμε ότι όσον αφορά τον άξονα της ράβδου και την διεύθυνση των αξονικών δυνάμεων συμπίπτουν.[6]



Σχήμα 8: *Πρότυπο Δικτύωμα Ράβδων - Παράδειγμα προς επίλυση*

Το πρόβλημα το οποίο καλούμαστε να δημιουργήσουμε το κώδικα και να το επιλύσουμε εμφανίζεται παρακάτω. Να πούμε ότι όταν λύνουμε δικτυώματα καλούμαστε να θέσουμε κάποιες αρχικές μεταβλητές και να δηλώσουμε κάποια στοιχεία σύμφωνα με το κάθε παράδειγμα. Αυτές είναι: nnodes, xnode, ynode, nelements, cnct, E, A, rho, load, nubcs.

Κώδικας - Matlab - **παραπομπή συναρτήσεων*****

```
%% example of trussd2d element
% solving a two-dimensional truss
% Copyright: G.E. Stavroulakis (2004)
%% number of nodes - Συνολικός Αριθμός
nnodes = 5; % Συνολικός αριθμός κόμβων
%% coordinates of nodes - Θέσεις για x,y συντεταγμένες κόμβων
xnode=zeros(nnodes,1);
ynode=zeros(nnodes,1);
xnode(1) = 1;
ynode(1) = 1;
xnode(2) = 5;
ynode(2) = 1;
xnode(3) = 0;
ynode(3) = 0;
xnode(4) = 5;
ynode(4) = 0 ;
xnode(5) = 10;
ynode(5) = 0;

%% number of elements - αριθμός πεπερασμένων στοιχείων δομήματος
nelements = 6;

%% connectivity of elements - συνδεσμολογία
cnct = zeros(nelements,2); %κόμβος αρχής, κόμβος τέλους
cnct(1,1) = 1;
cnct(1,2) = 2;
cnct(2,1) = 2;
cnct(2,2) = 3;
cnct(3,1) = 3;
cnct(3,2) = 4;
cnct(4,1) = 2;
cnct(4,2) = 4;
cnct(5,1) = 2;
cnct(5,2) = 5;
cnct(6,1) = 4;
cnct(6,2) = 5;

%% material constants - μέτρο Ελαστικότητας E, και εμβαδόν διατομής A
E = 100000;
A = 0.1;
rho = 2400.0;

%% loading
% number of loads - αριθμός φορτίσεων
nuload = 1;
```

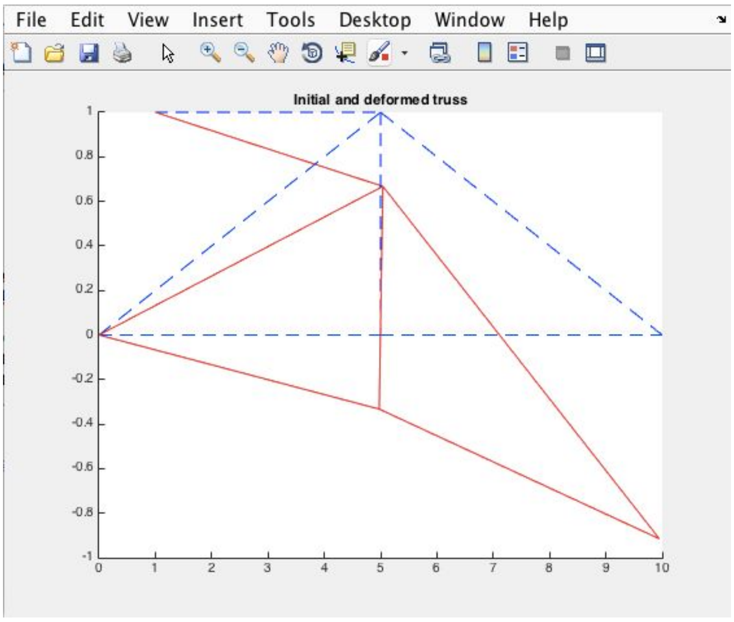
```
% number of node, x - y loadings - για κάθε φόρτιση , αριθμός κόμβου και χ,ψ συνιστώσες της.
loads = zeros(nuload,3);
loads(1,1) = 5;
loads(1,2) = 0;
loads(1,3) = -10;

%% boundary conditions - συνοριακές συνθήκες στήριξης
% number of boundary conditions
nubcs = 2;
% number of node, x - y displs (code 1 = 0, code 0 = free) για κάθε στήριξη , στη διεύθυνση χ,ψ δίνω 1 για
στήριξη , 0 για κύλιση.
bcs = zeros(nubcs,3);
bcs(1,1) = 1;
bcs(1,2) = 1;
bcs(1,3) = 1;
bcs(2,1) = 3;
bcs(2,2) = 1;
bcs(2,3) = 1;

%% preparation
% stiffnes and mass matrix - μητρώο δυσκαμψίας
Ktotal = zeros(2*nnodes,2*nnodes);
Mtotal = zeros(2*nnodes,2*nnodes);

% loading vector - φορτιστικός όρος
Ftotal = zeros(2*nnodes,1);

% space for the solution - λύση μετακινήσεις κόμβων
Utotal = zeros(2*nnodes,1);

% space for local stiffness  and mass matrix - χώρος για τοπικό μητρώο δυσκαμψίας
Kelm = zeros(4,4);
Melm = zeros(4,4) ;

% assemply of the stiffness and mass matrix - δημιουργία μητρώο δυσκαμψίας
for i=1:nelements  %κύκλος για όλα τα πεπερασμένα στοιχεία

   % for element i
   % first node number cnct(i,1)
   % first node coordinates xnode(cnct(i,1)), ynode(cnct(i,1))
   P1 = [ xnode(cnct(i,1)), ynode(cnct(i,1)) ];
   P2 = [ xnode(cnct(i,2)), ynode(cnct(i,2)) ];
   % local 4x4 stiffness matrix - δημιουργία τοπικού μητρωου δυσκαμψίας
   Kelm = truss2d(A,E,P1,P2);
 % assemply in global Ktotal stiffness matrix - τοποθέτηση στοιχείων από τοπικό μητρωο  (πεπερασμενου
στοιχείου) στο συνολικό μητρώο του δομήματος
   Ktotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)= ...
```

```
    Ktotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)+Kelm(1:2,1:2);
    Ktotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)= ...
    Ktotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)+Kelm(1:2,3:4);
    Ktotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)= ...
    Ktotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)+Kelm(3:4,1:2);
    Ktotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)= ...
    Ktotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)+Kelm(3:4,3:4);

    % local 4x4 mass matrix
    Melm = mtruss2d(A,rho,P1,P2)

    % assemply in global Mtotal mass matrix
    Mtotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)= ...
    Mtotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)+Melm(1:2,1:2);
    Mtotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)= ...
    Mtotal((cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)+Melm(1:2,3:4);
    Mtotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)= ...
    Mtotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,1)-1)*2+1:(cnct(i,1)-1)*2+2)+Melm(3:4,1:2);
    Mtotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)= ...
    Mtotal((cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2,(cnct(i,2)-1)*2+1:(cnct(i,2)-1)*2+2)+Melm(3:4,3:4);
end

% loading vector - δημιουργία διανυσματος φορτίσεων
for i=1:nuload
    % load at node loads(i,1) with x-y contribution equal to loads(i,2), loads(i,3)
    Ftotal(2*(loads(i,1)-1)+1)=loads(i,2);
    Ftotal(2*(loads(i,1)-1)+2)=loads(i,3);
end
%% imposing the boundary conditions = εισαγωγή συνοριακών συνθηκων
for i=1:nubcs
    % for node bcs(i,1) check x-y supports
    % if bcs(i,2)=1 then x displacement is fixed equal to zero
    % if bcs(i,3)=1 then y displacement is fixed equal to zero
    if bcs(i,2)==1
        Ktotal(2*(bcs(i,1)-1)+1,:)=0;
        Ktotal(:,2*(bcs(i,1)-1)+1)=0;
        Ktotal(2*(bcs(i,1)-1)+1,2*(bcs(i,1)-1)+1)=0;
    end
    if bcs(i,3)==1
        Ktotal(2*(bcs(i,1)-1)+2,:)=0;
        Ktotal(:,2*(bcs(i,1)-1)+2)=0;
        Ktotal(2*(bcs(i,1)-1)+2,2*(bcs(i,1)-1)+2)=0;
    end
end
% for mass matrix
for i=1:nubcs
    % for node bcs(i,1) check x-y supports
```

```
    % if bcs(i,2)=1 then x displacement is fixed equal to zero
    % if bcs(i,3)=1 then y displacement is fixed equal to zero
    if bcs(i,2)==1
        Mtotal(2*(bcs(i,1)-1)+1,:)=0;
        Mtotal(:,2*(bcs(i,1)-1)+1)=0;
        Mtotal(2*(bcs(i,1)-1)+1,2*(bcs(i,1)-1)+1)=1;
    end
    if bcs(i,3)==1
        Mtotal(2*(bcs(i,1)-1)+2,:)=0;
        Mtotal(:,2*(bcs(i,1)-1)+2)=0;
        Mtotal(2*(bcs(i,1)-1)+2,2*(bcs(i,1)-1)+2)=1;
    end
end
%
%% solving the system of equations - επίλυση συστήματος εξισώσεων
Utotal=Ktotal\Ftotal;
%% calculating eigenvalues
for i=1,2*nnodes
    Mtotal(i,i)=Mtotal(i,i);
end
V=eig(Ktotal,Mtotal);   % solve the matrix equation and print
V=sqrt(V);
[D,V1]=eig(Ktotal,Mtotal);
%% printing the solution
%
for ieig=1:size(V)
    ieig
    V(ieig)
newplot
hold on

for i=1:nelements
x = [xnode(cnct(i,1)) xnode(cnct(i,2)) ];
y=[ynode(cnct(i,1)) ynode(cnct(i,2)) ];
    plot(x,y,'b--')

    % displacements of nodes
    xdispl = [D(2*(cnct(i,1)-1)+1,ieig) D(2*(cnct(i,2)-1)+1,ieig) ];
    ydispl = [D(2*(cnct(i,1)-1)+2,ieig) D(2*(cnct(i,2)-1)+2,ieig) ];
    plot(x+0.5*xdispl,y+0.5*ydispl,'r')
end
eigen=num2str(V(ieig));
titl = ['eigenvector for eigenvalue equal to =' eigen];
title(titl)
hold off
pause
end
```

4.2.2 Αποτελέσματα Δικτυώματος Ράβδου

Με βάση το πρότυπο σχήμα μας (Σχήμα 8) η επίλυση του κώδικα για στατικό πρόβλημα μας δίνει το παρακάτω σχήμα παραμόρφωσης όταν υποστεί την δύναμη φορτίου στον κόμβο 5.



Σχήμα 9: Παραμόρφωση δικτυώματος μετά την επίλυση

Τα αμέσως επόμενα αποτελέσματα “Melm” αποτελούν μητρώα μάζας για κάθε στοιχείο/ράβδο χωριστά.

Melm =

Columns 1 Column 2 Column 3 Column 4

320	0	160	0
0	320	0	160
160	0	320	0
0	160	0	320

Melm =

Columns 1 Columns 2 Columns 3 Columns 4

407.9216	0	203.9608	0
0	407.9216	0	203.9608
203.9608	0	407.9216	0
0	203.9608	0	407.9216

Melm =

Columns 1 Columns 2 Columns 3 Columns 4

400	0	200	0
0	400	0	200
200	0	400	0
0	200	0	400

Melm =

Columns 1	Columns 2	Columns 3	Columns 4
80	0	40	0
0	80	0	40
40	0	80	0
0	40	0	80

Melm =

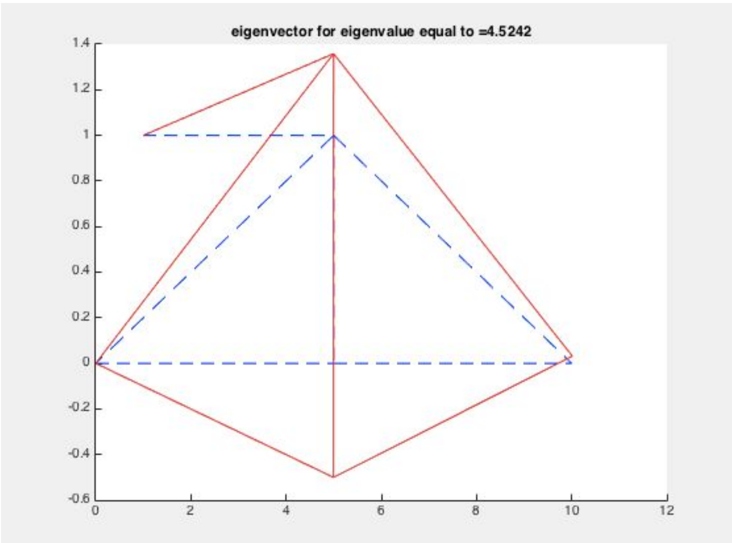
Columns 1	Columns 2	Columns 3	Columns 4
407.9216	0	203.9608	0
0	407.9216	0	203.9608
203.9608	0	407.9216	0
0	203.9608	0	407.9216

Melm =

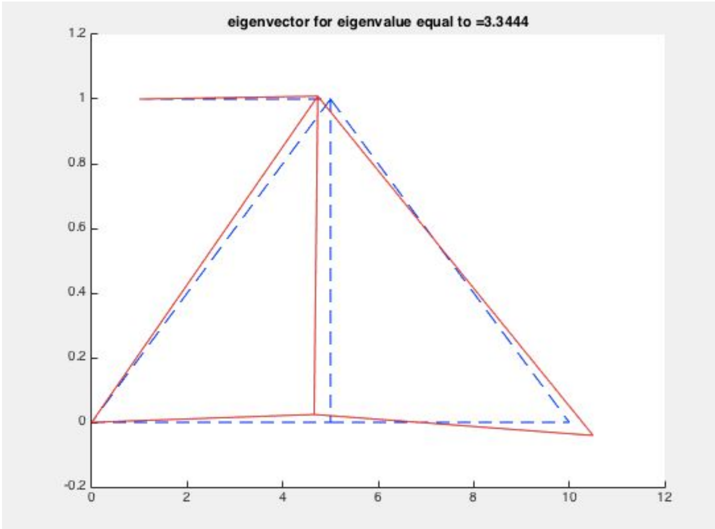
Columns 1	Columns 2	Columns 3	Columns 4
400	0	200	0
0	400	0	200
200	0	400	0
0	200	0	400

Όσον αφορά τα αποτελέσματα που προκύπτουν από την επίλυση του προβλήματος, οι ιδιοτιμές και οι ιδιομορφές αναπαρίστανται στα παρακάτω σχήματα. Εμφανίζονται σχηματικά οι πρώτες 10 ιδιομορφές με τις αντίστοιχες ιδιοτιμές τους.

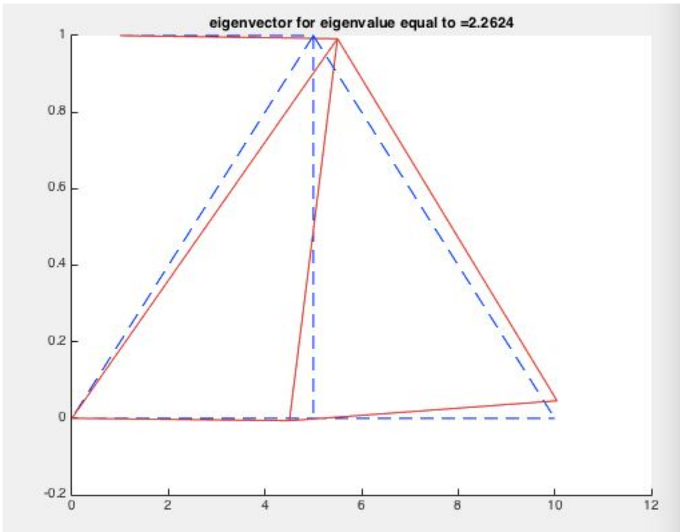
ieig = 1 ans = 4.5242



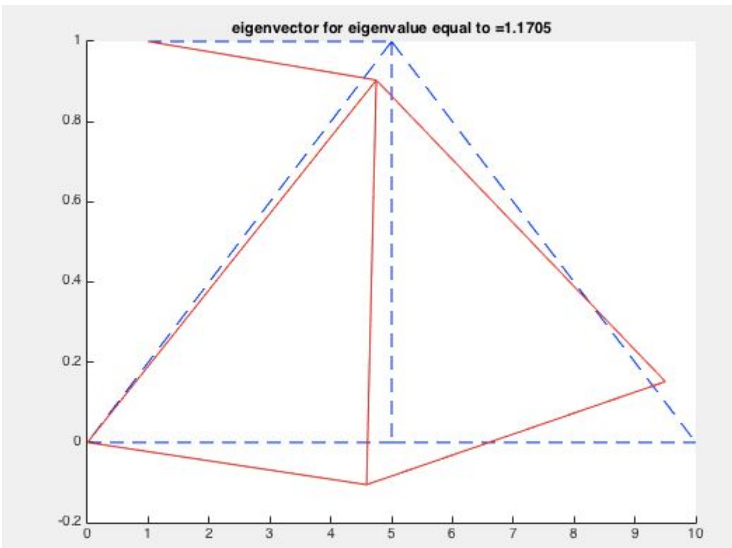
ieig = 2   ans = 3.3444



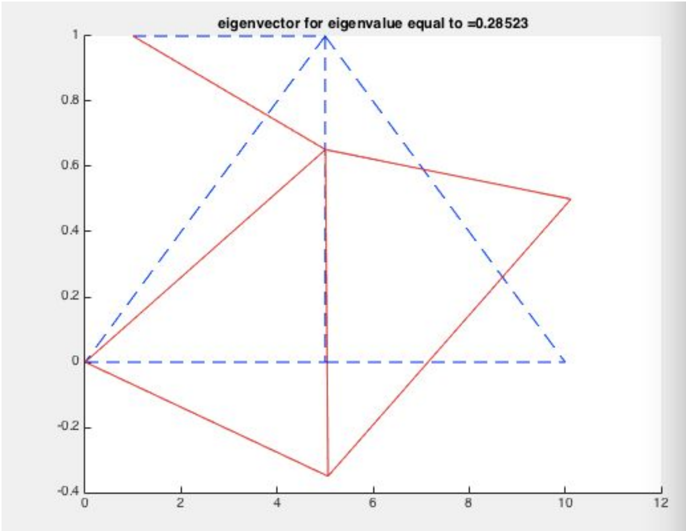
ieig = 3   ans = 2.2624



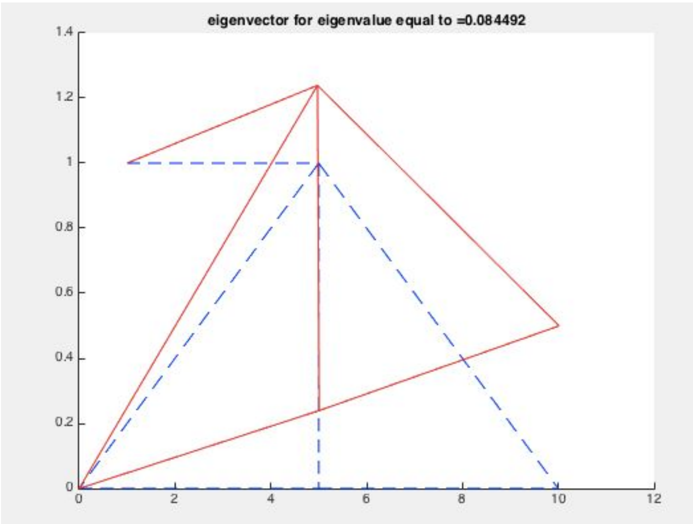
ieig = 4   ans =1.1705



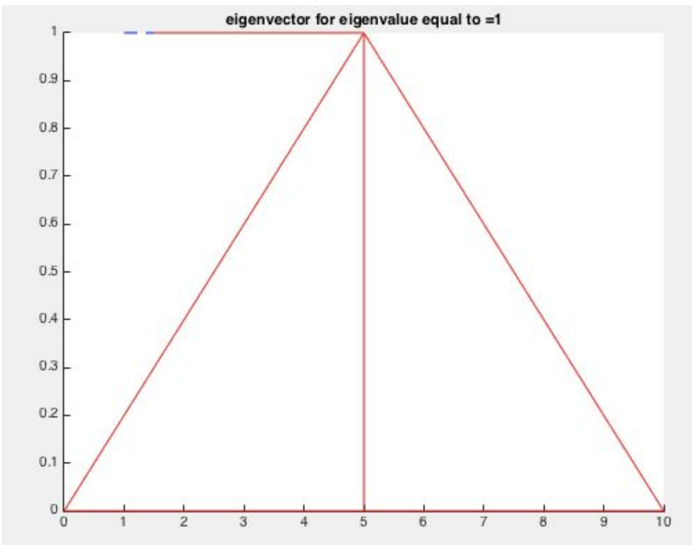
ieig = 5   ans =0.2852



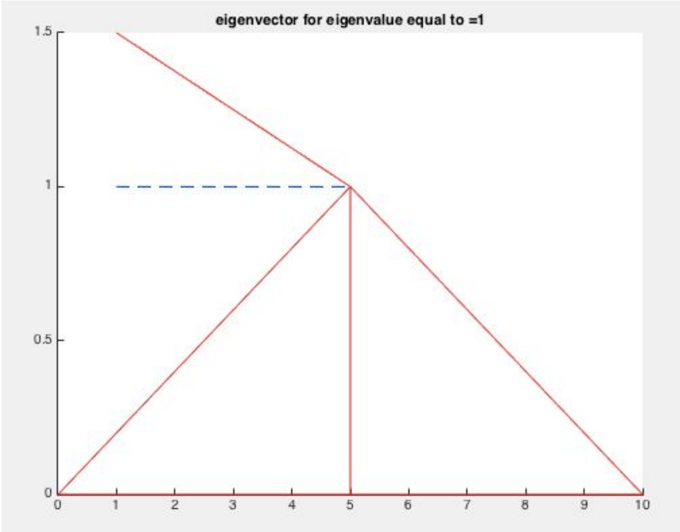
ieig = 6   ans = 0.0845



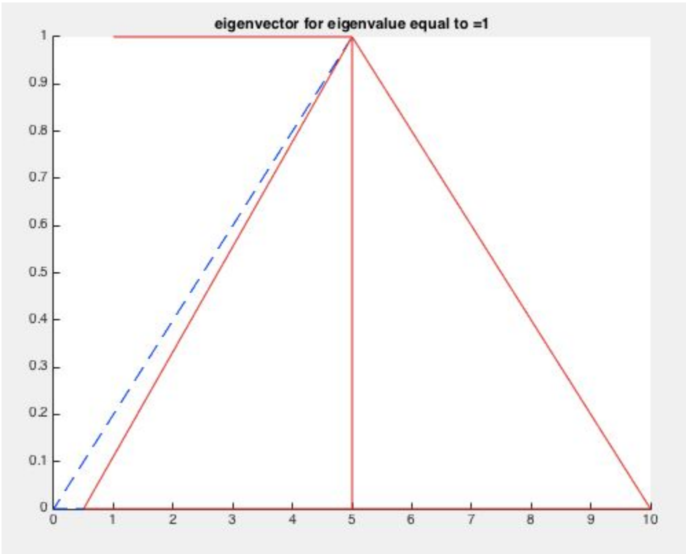
ieig =7   ans =1



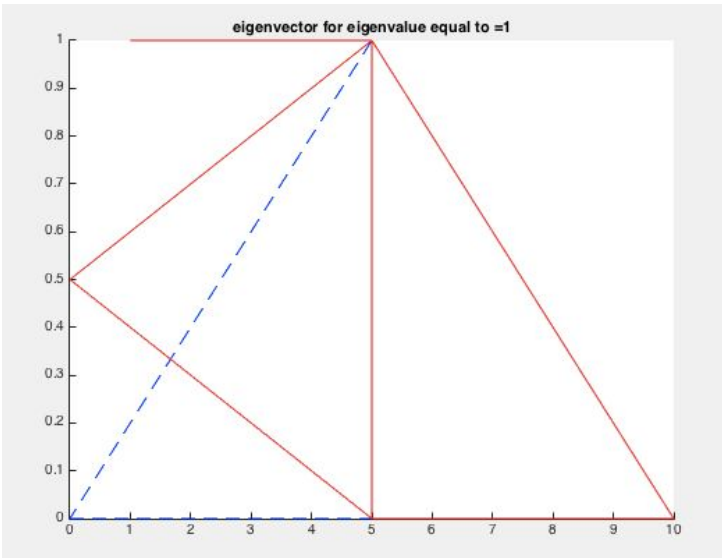
ieig = 8   ans = 1



ieig = 9   ans =1



ieig = 10   ans =1



4.2.3 Συμπεράσματα

Σκοπός μας ήταν να δημιουργηθεί ένας αλγόριθμος προσαρμόσιμος σε δομές ράβδου, όπου ανάλογα τα δεδομένα που εισάγουμε να μας δίνει αποτελέσματα για την παραμόρφωση της δομής αλλά και τις ιδιοτιμές και ιδιομορφές. Τέτοιου είδους κώδικές μπορούν να είναι η βάση για τη μελέτη πολύπλοκων μηχανικών κατασκευών αλλά και συγκρισης τους. Ανάλογα με τον αριθμό κόμβων και στοιχείων που δηλώνονται αλλά και της διαταξης του μπορούμε να πάρουμε διαφορετικά αποτελέσματα αντοχής και παραμορφωσης για τις δομές ράβδου.

4.3 Δομές Πλαισίου Δίσκου σε Έλξη

Στη μηχανική, ένα υλικό βρίσκεται σε κατάσταση πίεσης (Plane Stress) όταν όλες οι τάσεις δρουν σε ένα μόνο επίπεδο, δηλαδή οι άμεσες και οι διατμητικές τάσεις που είναι κάθετες στο επίπεδο είναι μηδέν, $\sigma_z=\tau_{xz}=\tau_{yz}=0$. [11] Όταν η κατάσταση αυτή συμβαίνει σε μία ολόκληρη δομή, όπως συμβαίνει συχνά με τις λεπτές πλάκες, η ανάλυση της τάσης απλοποιείται σημαντικά, καθώς η κατάσταση καταπόνησης μπορεί να αναπαρασταθεί ως μήτρα 2×2 από 3×3.
Ανάλυση τάσης στο επίπεδο εκτελείται για δομές με φορτία στο επίπεδο τους όπως πλάκες με οπές κλπ.
Η καταπόνηση στο επίπεδο (plane stress) συμβαίνει συνήθως σε λεπτές επίπεδες πλάκες στις οποίες ασκούνται μόνο δυνάμεις φορτίου οι οποίες είναι παράλληλες προς αυτές. Σε ορισμένες περιπτώσεις, μια ελαφρά καμπυλωμένη λεπτή πλάκα μπορεί επίσης να θεωρηθεί ότι έχει τάση στο επίπεδο για τον σκοπό ανάλυσης τάσεων. Αυτή είναι η περίπτωση, για παράδειγμα, ενός κυλινδρικού λεπτού τοιχώματος γεμάτου με ρευστό υπό πίεση. Σε τέτοιες περιπτώσεις, εξαρτήματα τάσης κάθετα προς την πλάκα είναι αμελητέα σε σύγκριση με εκείνα που είναι παράλληλα προς αυτήν. [10]

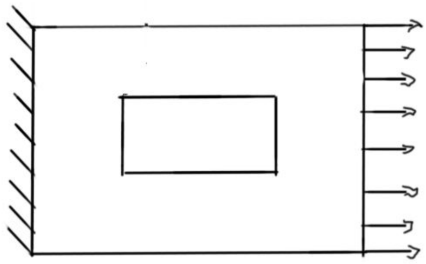
Οι δομές πλάκας ή κελύφους με ένα πάχος που είναι σχετικά μικρό σε σύγκριση με τις διαστάσεις του υπόλοιπου πλαισίου και με φορτίσεις έξω από το επίπεδο μπορούν να οριστούν με στοιχεία κάμψης πλάκας ή κελύφους. Αυτά τα στοιχεία μπορεί να βρίσκονται οπουδήποτε στον τρισδιάστατο χώρο. Μπορούν να διακριθούν τρεις διαφορετικοί τύποι στοιχείων πλάκας και κελύφους:

1. Στοιχεία κάμψης στο επίπεδο
2. Στοιχεία επίπεδου κελύφους
3. Καμπύλα στοιχεία κελύφους

Τα στοιχεία κάμψης πλακών πρέπει να πληρούν τις ακόλουθες συνθήκες όσον αφορά το σχήμα και τη φόρτωση:

- Πρέπει να είναι επίπεδα, δηλαδή οι συντεταγμένες των κόμβων των στοιχείων πρέπει να είναι σε ένα επίπεδο, το xy επίπεδο του στοιχείου.
- Το πάχος t πρέπει να είναι μικρό σε σχέση με τις διαστάσεις b στο επίπεδο του στοιχείου.
- Η φόρτωση δύναμης F πρέπει να ενεργεί κάθετα στο επίπεδο του στοιχείου,
- η φόρτωση της ροπής M πρέπει να ενεργεί γύρω από έναν άξονα ο οποίος ευρίσκεται στο επίπεδο του στοιχείου.

Σκοπός αυτής της εργασίας είναι να αναλύσει μια πλάκα με μία οπή που υπόκειται σε καταπόνηση όπως δείχνει και το πρότυπο σχήμα παρακάτω. Για να θεωρήσουμε την ύπαρξη της οπής αυτής, υπολογιστικά υποθέσαμε ότι η ελαστικότητα όλων των σημείων που περιλαμβάνει η οπή έχουν οριακά μηδενική ελαστικότητα. Το σχήμα είναι ένας δίσκος δύο διαστάσεων. Το ένα του άκρο (αριστερό) είναι πακτωμένο ενώ στο άλλο ακρο του (δεξί) ασκείτε ομοιόμορφα κατανεμημένη δύναμη $1*10^6$ N. Εχει μήκος 5m, πλάτος 1m και μέτρο ελαστικότητας $E=10e^7$.



Σχήμα 10: πρότυπο σχήμα που αναλύουμε

4.3.1 Παράδειγμα Πλαισίου Δίσκου με οπή - Κώδικας

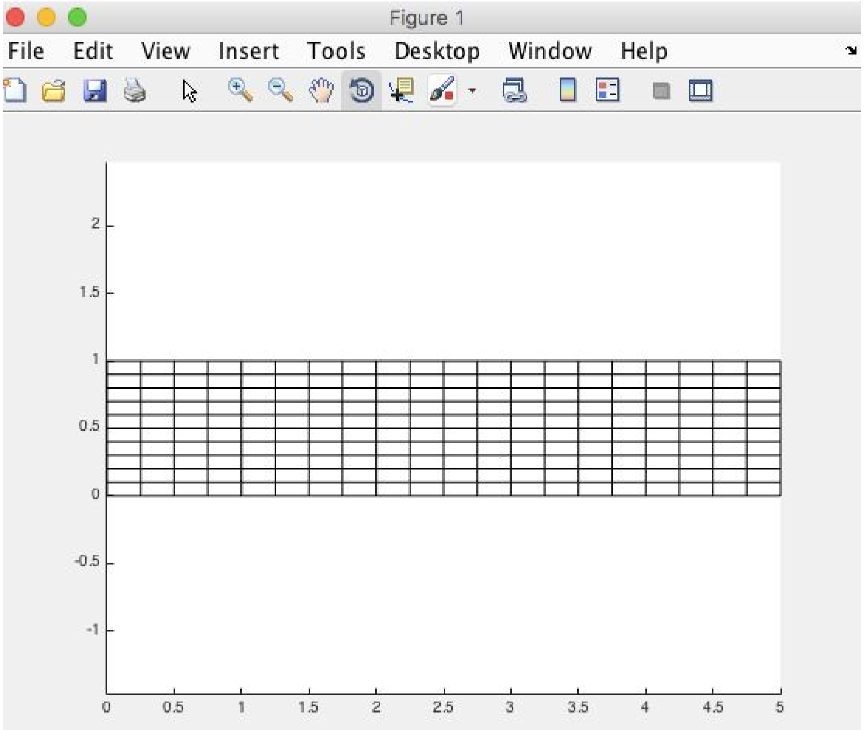
Το πρόβλημα το οποίο καλούμαστε να δημιουργήσουμε το κώδικα και να το επιλύσουμε εμφανίζεται παρακάτω. Να πούμε ότι όταν λύνουμε οποιαδήποτε δομή καλούμαστε να θέσουμε κάποιες αρχικές μεταβλητές και να δηλώσουμε κάποια στοιχεία σύμφωνα με το κάθε παράδειγμα. Αυτές είναι: elasticity (Poisson), Load (P), Mesh generation (Lx, Ly), numberElementsX, numberElementsY, nodeCoordinates, elementNodes, numberNodes, GDof.

```
%2D plate problem
%clear memory
clear all;
colordef white;
clf
%materials
%elasticity ?
poisson=0.30;
%load
P=1e6;
%Mesh generation
Lx=5;
Ly=1;
numberElementsX=20;
numberElementsY=10;
numberElements=numberElementsX*numberElementsY;
[nodeCoordinates,elementNodes]=rectangularMesh(Lx,Ly,numberElementsX,numberElementsY);
xx=nodeCoordinates(:,1);
yy=nodeCoordinates(:,2);
%Change the elasticity in specific area based on numberElements number
for i=1:numberElements
   E(i)=10e7;
```

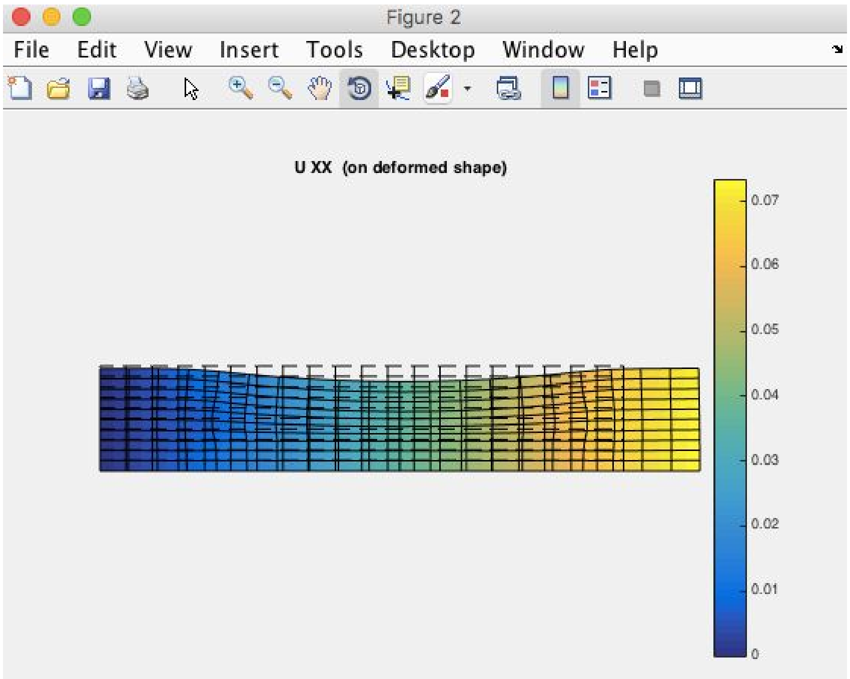
```
    if i>=65 && i<=76
       E(i)=10e7/10000;
    elseif i>=85 && i<=96
       E(i)=10e7/10000;
    elseif i>=105 && i<=116
      E(i)=10e7/10000;
    elseif i>=125 && i<=136
      E(i)=10e7/10000;
    end
end
drawingMesh(nodeCoordinates,elementNodes,'Q4','k-');
numberNodes=size(xx,1);

%GDof: global number of degrees of freedom
GDof=2*numberNodes;
%computation of the system stiffness matrix
stiffness=formStiffness2D(GDof,numberElements,elementNodes,numberNodes,nodeCoordinates,E,poisson,1,1);
% deform shape
figure
drawingField(nodeCoordinates+scaleFactor*[UX UY],elementNodes,'Q4',UX);   %U XX
hold on
drawingMesh(nodeCoordinates+scaleFactor*[UX UY],elementNodes,'Q4','k-');
drawingMesh(nodeCoordinates,elementNodes,'Q4','k--');
colorbar
title('U XX  (on deformed shape)')
axis off
% stresses at nodes
stresses2D(GDof,numberElements,elementNodes,numberNodes,nodeCoordinates,displacements,UX,UY,E,poisson,sc
aleFactor);
```

4.3.2 Αποτελέσματα Ανάλυσης Δίσκου

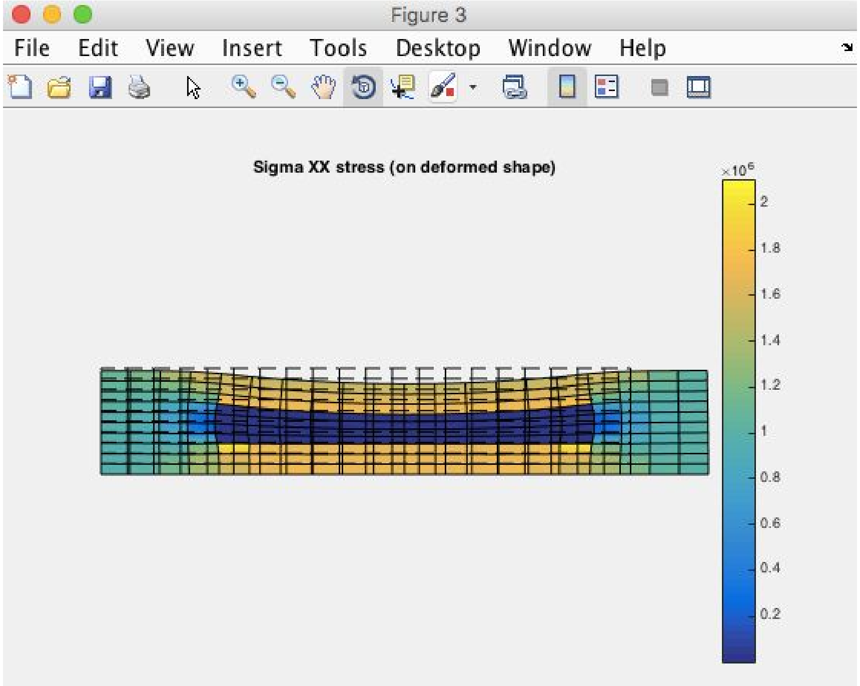
Παρακάτω απεικονίζονται τα αποτελέσματα της ανάλυσης σε διαγράμματα που αφορουν την παραμόρφωση του στερεού τόσο σε σχέση με τις τάσεις όσο και με την παραμόρφωση μετατόπισης που δέχεται από τις δυνάμεις που του ασκούνται.



Σχήμα 11: Αρχική μορφή στερεού



Σχήμα 12: Παραμόρφωση μετατόπισης



Σχήμα 13: Παραμόρφωση τάσεων

4.3.3 Συμπεράσματα

Στην ανάλυση του στερεού μου υλοποιήσαμε με τον κώδικα matlab βρήκαμε τις καταπονήσεις που δέχεται ο δίσκος ανάλογα με το συνολο των δυνάμεων που του ασκούνται αλλα και του υλικού. Για ανάθεση διαφορετικών δεδομένων τα αποτελέσματα θα διαφέρουν. Επίσης σε κάθε περίπτωση θα μπορούμε να βλέπουμε την καταπόνηση και τις αντιδράσεις του στερεού στο σημείο που έχει την οπή. Ο κώδικας που έχει δημιουργηθεί μπορεί να εφαρμοστεί σε κάθε παράδειγμα παρόμοιας δομής με διαφορετικά δεδομένα εισόδου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Σπύρος Πνευματικός, Δυναμικά Συστήματα, Πανεπιστήμιο Πάτρας, Τμήμα Μαθηματικών, 2010-2011

[2] Δ. Βενετσάνος, Ι. Αντωνιάδης, Εργαστήριο Δυναμικής και Κατασκευών, ΕΜΠ, Σχολή Μηχανολόγων Μηχανικών, 2010-2011

[3] Αμφιθέα Τσιγούρη, Δυναμική απόκριση πλακών και επιρροή αποκόλλησης, Διπλωματική εργασία, Πολυτεχνείο Κρήτης, 2016

[4] Γεώργιος Ταϊρίδης, Optimal design of smart structures with intelligent control, Διδακτορική Διατριβή, Πολυτεχνείο Κρήτης, 2016

[5] Γεώργιος Ε. Σταυρουλάκης, Μαρία Ε. Σταυρουλάκη, Αλίκη Δ. Μουραντόβα, Υπολογιστική Μηχανική. Θεωρία και Εφαρμογές Πεπερασμένων Στοιχείων, Ηλεκτρονικό βιβλίο, Εκδόσεις Κάλλιπος, 2015 https://repository.kallipos.gr/handle/11419/4557

[6] Δρ. Πασχάλης Κ. Γκότσης, Πεπερασμένα Στοιχεία, Εκδόσεις Ζήτη, 2013

[7] http://www.mathworks.com

[8] Finite Element Analysis Program, International Journal For Technological Research In Engineering Volume 3, Issue 9, May-2016, ISSN (Online): 2347 - 4718

[9] Μεταλλικές Δικτυωτές Γέφυρες , Διπλωματική, Δημήτριος Κεκές, ΑΤΕΙ Αθηνων, 2009

[10] Plane stress, wikipedia: https://en.wikipedia.org/wiki/Plane_stress

[11] Finite Element Analysis, University Malaysia Pahang, Dr. Gul Ahmed Jokhio, http://ocw.ump.edu.my/pluginfile.php/9857/mod_resource/content/2/17_plane_strain.pdf

[12] Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Μηχανολόγων Μηχανικών, Σημειώσεις Εργαστηρίου Δυναμικής και κατασκευών Μαρτιος 2005. http://users.ntua.gr/cprovat/yliko/AMKI_2D_truss.pdf

[13]Berdekou Andriana, Διπλωματική Εργασία, Πολυτεχνείο Κρήτης 2018

[14] Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Ναυπηγών Μηχανολόγων Μηχανικών, Υπολογιστικές μέθοδοι και εφαρμογές σε λεπτότοιχες κατασκευές, Πέτρος Α. Καρύδης. (http://users.ntua.gr/caridis/)

ΠΑΡΑΠΟΜΠΗ - ΣΥΝΑΡΤΗΣΕΩΝ ΠΡΟΓΡΑΜΜΑΤΩΝ

1. ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΠΛΑΙΣΙΟΥ ΔΟΚΩΝ

```
function [stiffness]=formStiffness2Dframe(GDof,numberElements,elementNodes,numberNodes,xx,yy,EI,EA);
stiffness=zeros(GDof);
%In matrix elementNodes we define the connections (left and right nodes) at each element
%computatio of the system stiffness matrix
for e=1:numberElements;
   %elementDof:element degree of freedom (Dof)
   indice=elementNodes(e,:);
   elementDof=[indice indice+numberNodes indice+2*numberNodes];
   nn=length(indice);
   xa=xx(indice(2))-xx(indice(1));
   ya=yy(indice(2))-yy(indice(1));
   length_element=sqrt(xa*xa+ya*ya);
   ll=length_element;
   cosa=xa/length_element;
   sena=ya/length_element;

   L=[cosa*eye(2) sena*eye(2) zeros(2);
      -sena*eye(2) cosa*eye(2) zeros(2);
      zeros(2,4) eye(2)];

   oneu=[1 -1; -1 1];
   oneu2=[1 -1; 1 -1];
   oneu3=[1 1; -1 -1];
   oneu4=[4 2; 2 4];

   k1=[EA/ll*oneu zeros(2,4);
      zeros(2) 12*EI/ll^3*oneu 6*EI/ll^2*oneu3;
      zeros(2) 6*EI/ll^2*oneu2 EI/ll*oneu4];
   stiffness(elementDof,elementDof)=stiffness(elementDof,elementDof)+L'*k1*L;
end

%...............................................................


function outputDisplacementsReactions...
   (displacements,stiffness,GDof,prescribedDof)

% output of displacements and reactions in
% tabular form

% GDof: total number of degrees of freedom of
% the problem

% displacements
%disp('Displacements')
%displacements=displacements1;
jj=1:GDof; format;
Displacements=[jj' displacements]

% reactions
F=stiffness*displacements;
reactions=F(prescribedDof);
disp('reactions')
[prescribedDof reactions]
end

%...............................................................

function displacements=solution(GDof,prescribedDof,stiffness,force)
% function to find solution in terms of global displacements
activeDof=setdiff([1:GDof]', ...
   [prescribedDof]);
```

```
U=stiffness(activeDof,activeDof)\force(activeDof);
displacements=zeros(GDof,1);
displacements(activeDof)=U;
```

2. ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΔΙΚΤΥΩΜΑΤΟΣ ΡΑΒΔΟΥ

```
function K = truss2d(A,E,P1,P2)
%   TRUSS2D
%
%   This command generates the 4 x 4 stiffness matrix for a planar
%   truss element in global coordinates. The syntax is:
%            K = truss2d(A,E,P1,P2)
%   where: A is the cross-sectional area;
%          E is the Young's modulus;
%          P1 and P2 are vectors of the {x,y} coordinates of the
%            endpoints.

l = norm(P2-P1);
alpha = atan2(P2(2)-P1(2),P2(1)-P1(1));
lamb = [cos(alpha) sin(alpha) -cos(alpha) -sin(alpha)]';
K = A*E/l*lamb*lamb';

%................................................................


function M = mtruss2d(A,rho,P1,P2)

%   MTRUSS2D
%
%   This command generates the 4 x 4 mass matrix for a planar truss
%   element in global coordinates. The syntax is:
%            M = mtruss2d(A,rho,P1,P2)
%   where: A is the cross-sectional area;
%          rho is the material density;
%          P1 and P2 are vectors of the {x,y} coordinates of the
%            end points.

L = norm(P2-P1);
I2 = [1 0; 0 1];
mm = [2*I2 I2; I2 2*I2];
M = rho*A*L/6*mm;
end
```

3. ΠΑΡΑΔΕΙΓΜΑΤΟΣ ΔΙΣΚΟΥ ΜΕ ΟΠΗ

```
function [stiffness,mass]=formStiffness2D(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,E,poisson,rho,thickness)

% compute stiffness matrix (and mass matrix)
% for plane stress Q4 elements

stiffness=zeros(GDof);
mass=zeros(GDof);

% 2 by 2 quadrature
[gaussWeights,gaussLocations]=gaussQuadrature('complete');

for e=1:numberElements
  indice=elementNodes(e,:);
  elementDof=[ indice indice+numberNodes ];
  ndof=length(indice);

  % cycle for Gauss point
  for q=1:size(gaussWeights,1)
    GaussPoint=gaussLocations(q,:);
```

```
    xi=GaussPoint(1);
    eta=GaussPoint(2);

% shape functions and derivatives
    [shapeFunction,naturalDerivatives]=shapeFunctionQ4(xi,eta)

% Jacobian matrix, inverse of Jacobian,
% derivatives w.r.t. x,y
    [Jacob,invJacobian,XYderivatives]=...
        Jacobian(nodeCoordinates(indice,:),naturalDerivatives);

%  B matrix
    B=zeros(3,2*ndof);
    B(1,1:ndof)       = XYderivatives(:,1)';
    B(2,ndof+1:2*ndof)  = XYderivatives(:,2)';
    B(3,1:ndof)       = XYderivatives(:,2)';
    B(3,ndof+1:2*ndof)  = XYderivatives(:,1)';

% stiffness matrix
C=E(e)/(1-poisson^2)*[1 poisson 0;poisson 1 0;0 0 (1-poisson)/2];
    stiffness(elementDof,elementDof)=...
        stiffness(elementDof,elementDof)+...
        B'*C*thickness*B*gaussWeights(q)*det(Jacob);
% mass matrix
    mass(indice,indice)=mass(indice,indice)+...
        shapeFunction*shapeFunction'*...
        rho*thickness*gaussWeights(q)*det(Jacob);
    mass(indice+numberNodes,indice+numberNodes)=...
        mass(indice+numberNodes,indice+numberNodes)+...
        shapeFunction*shapeFunction'*...
        rho*thickness*gaussWeights(q)*det(Jacob);

  end
end

%..............................................................

function displacements=solution(GDof,prescribedDof,stiffness,force)
% function to find solution in terms of global displacements
activeDof=setdiff([1:GDof]', ...
    [prescribedDof]);
U=stiffness(activeDof,activeDof)\force(activeDof);
displacements=zeros(GDof,1);
displacements(activeDof)=U;


%..............................................................

function stresses2D(GDof,numberElements,...
    elementNodes,numberNodes,nodeCoordinates,...
    displacements,UX,UY,E,poisson,scaleFactor)

% 2 by 2 quadrature
[gaussWeights,gaussLocations]=gaussQuadrature('complete');

% stresses at nodes
stress=zeros(numberElements,size(elementNodes,2),3);
stressPoints=[-1 -1;1 -1;1 1;-1 1];

for e=1:numberElements
  indice=elementNodes(e,:);
  elementDof=[ indice indice+numberNodes ];
  nn=length(indice);
  for q=1:size(gaussWeights,1)
    pt=gaussLocations(q,:);
    wt=gaussWeights(q);
    xi=pt(1);
    eta=pt(2);
```

```
% shape functions and derivatives
    [shapeFunction,naturalDerivatives]=shapeFunctionQ4(xi,eta)

% Jacobian matrix, inverse of Jacobian,
% derivatives w.r.t. x,y
    [Jacob,invJacobian,XYderivatives]=...
        Jacobian(nodeCoordinates(indice,:),naturalDerivatives);

%  B matrix
    B=zeros(3,2*nn);
    B(1,1:nn)       = XYderivatives(:,1)';
    B(2,nn+1:2*nn)  = XYderivatives(:,2)';
    B(3,1:nn)       = XYderivatives(:,2)';
    B(3,nn+1:2*nn)  = XYderivatives(:,1)';

% element deformation
C=E(e)/(1-poisson^2)*[1 poisson 0;poisson 1 0;0 0 (1-poisson)/2];
    strain=B*displacements(elementDof);
    stress(e,q,:)=C*strain;
  end
end

% drawing stress fields
% on top of the deformed shape
figure
drawingField(nodeCoordinates+scaleFactor*[UX UY],...
    elementNodes,'Q4',stress(:,:,1));%sigma XX
hold on
drawingMesh(nodeCoordinates+scaleFactor*[UX UY],...
    elementNodes,'Q4','k-');
drawingMesh(nodeCoordinates,elementNodes,'Q4','k--');
colorbar
title('Sigma XX stress (on deformed shape)')
axis off

%..............................................................

function [nodeCoordinates,elementNodes] = rectangularMesh(Lx,Ly,...
                                  numberElementsX,numberElementsY)
          xx = 0:Lx/numberElementsX:Lx;
          yy = 0:Ly/numberElementsY:Ly;
          [XX YY] = meshgrid(yy,xx);
          nodeCoordinates = [YY(:),XX(:)];
          elementNodes = zeros(numberElementsX*numberElementsY,4);
          j = 1;
          i =1;
          i1 =0;
          counter = 0;
          for j = 1:numberElementsY
              for i =1: numberElementsX
                  counter = counter +1;
                  if i ==1 && j==1
                      i1 =1;
                  else
                      i1 = i1 +1;
                  end

                  i2 = i1 + 1;
                  i4 = i1 + numberElementsX + 1;
                  i3 = i2 + numberElementsX + 1;
                  elementNodes(counter,:) = [i1 i2 i3 i4];
              end
               i1 = i1+1;i2 = i2+1;
          end
 end
```