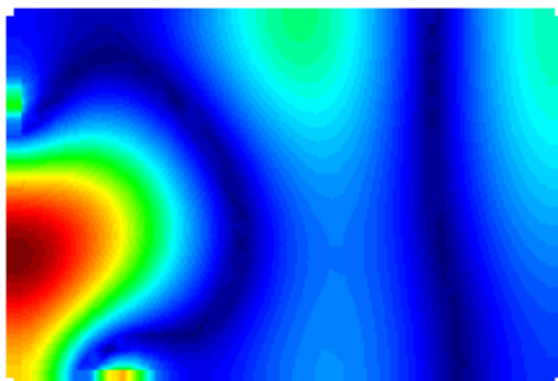


Τμήμα Μουσικής Τεχνολογίας και Ακουστικής
Παράρτημα Ρεθύμνου, Τ.Ε.Ι. Κρήτης

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

*Ανάπτυξη εκπαιδευτικών εφαρμογών Ακουστικής με τη
χρήση της μεθόδου των συνοριακών στοιχείων*



Ιωαννίδης Λεωνίδας
Επιβλέπων καθηγητής: Κουζούπης Σπύρος

Μάιος 2009

Ευχαριστώ για την πολύτιμη βοήθεια τους
τον καθηγητή μου, Σπύρο Κουζούπη
τους γονείς μου
την αδερφή μου, Άννα
τους φίλους

Περίληψη

Η εργασία αυτή πραγματεύεται την επίλυση προβλημάτων ακουστικής μέσω της υπολογιστικής μεθόδου των συνοριακών στοιχείων (Boundary Element Method). Παρουσιάζονται οι αρχές της μεθόδου και όλη η μεθοδολογία με την οποία μπορεί να επιλύσει προβλήματα, ο μηχανικός ή ο σπουδαστής του κλάδου της ακουστικής, με βασικό στόχο οι εφαρμογές που λύνονται εδώ να έχουν εκπαιδευτική σημασία. Κατ'αρχήν παρουσιάζεται το μαθηματικό υπόβαθρο που χρησιμοποιεί η μέθοδος, μετά αναφερόμαστε στην ολοκληρωτική εξίσωση, στις συνοριακές συνθήκες, στην διακριτοποίηση - δικτύωση συνόρων ώστε να μπορεί να γίνει κατανοητή η μέθοδος. Ο βασικός όμως στόχος της εργασίας είναι να αναπτυχθούν τα κατάλληλα εργαλεία ώστε να γίνει όσον το δυνατό πιο εύκολη η σύγκλιση διαφορετικών εφαρμογών, δηλαδή της σχεδίασης της γεωμετρίας ενός προβλήματος, της διακριτοποίησης της γεωμετρίας, της επίλυσης του αριθμητικού προβλήματος και της παρουσίασης των αποτελεσμάτων. Τα προβλήματα τα οποία έχουν επιλεγεί προς λύση, πιστεύουμε ότι έχουν διδακτική αποστολή. Αυτό επιτυγχάνεται με την σχετικά εύκολη εισαγωγή δεδομένων μέσω ενός σχεδιαστικού προγράμματος (GiD), το οποίο αναλαμβάνει πέραν από τη δημιουργία του δικτύωματος (προεπεξεργαστής) και την παρουσίαση των αποτελεσμάτων (μετά-επεξεργαστής). Παρουσιάζονται επίσης όλες οι αλλαγές που χρειάστηκαν να γίνουν σε ένα υπάρχοντα κώδικα βασικής υλοποίησης της αριθμητικής μεθόδου, με σκοπό την καλύτερη κατανόηση της ίδιας της μεθόδου των συνοριακών στοιχείων αλλά και την κατανόηση του τρόπου εργασίας με όλη την εφαρμογή που αναπτύχθηκε. Ακολούθως λύνεται, μία σειρά από ακουστικά προβλήματα διαφόρων τύπων (εκπομπής και σκέδασης), σε δύο και σε τρεις διαστάσεις, όπως π.χ. στάσιμα κύματα σε σωλήνες και σε δωμάτια, εκπομπή από διάφορα μοντέλα ηχείων και σκέδαση κυμάτων από τρισδιάστατα αντικείμενα. Για να παρουσιαστεί η αποτελεσματικότητα και οι δυνατότητες της μεθόδου καθώς και να εκπαιδευτεί ο αναγνώστης στο να λύνει τέτοια προβλήματα, στο τρίτο κεφάλαιο, υπάρχουν οδηγίες με τα βήματα που απαιτούνται για την επίλυση του κάθε προβλήματος, αλλά και τη χρήση της όλης εφαρμογής γενικότερα. Την λύση κάθε προβλήματος ακολουθεί σύντομη θεωρία καθώς και σύγκριση των αριθμητικών αποτελεσμάτων με την αντίστοιχη αναλυτική λύση, όπου αυτή υπάρχει. Τα παραρτήματα συμπληρώνουν το κυρίως κείμενο με έξτρα γραφήματα και με τους κώδικες που απαιτούνται για να τρέξουν οι εφαρμογές, καθώς και άλλα χρήσιμα στοιχεία.

Περιεχόμενα

Εισαγωγή.....	1
Κεφάλαιο 1^ο: Η μέθοδος των συνοριακών συνθηκών.....	2
1.1 Εισαγωγή.....	2
1.2 Βασικές έννοιες και τύποι.....	2
1.2.1 Το θεώρημα Gauss – Green.....	2
1.2.2 Η δεύτερη ταυτότητα του Green.....	4
1.2.3 Η συνάρτηση Δέλτα ή Dirac.....	5
1.3 Η θεμελιώδης εξίσωση του Helmholtz.....	5
1.4 Συνοριακές συνθήκες.....	6
1.5 Η ολοκληρωτική εξίσωση του Helmholtz για εσωτερικά προβλήματα.....	7
1.6 Η ολοκληρωτική εξίσωση του Helmholtz για εξωτερικά προβλήματα.....	9
1.7 Διακριτοποίηση συνόρων.....	10
1.7.1 Δισδιάστατα σύνορα.....	12
1.7.2 Τρισδιάστατα σύνορα.....	13
Κεφάλαιο 2^ο: Υλοποίηση εκπαιδευτικής εφαρμογής με χρήση λογισμικού σχεδίασης και δικτύωσης.....	16
2.1 Εισαγωγή.....	16
2.2 Το λογισμικό σχεδίασης και διακριτοποίησης: GiD.....	16
2.2.1 Το πρόγραμμα GiD ως προεπεξεργαστής (preprocessor).....	17
2.2.2 Σχεδιάζοντας στο GiD ένα παραλληλόγραμμο.....	18
2.2.3 Δικτύωση γεωμετριών.....	19
2.2.4 Εισάγοντας άλλα format αρχείων.....	21
2.2.5 Χρησιμοποιώντας το GiD για επίλυση προβλημάτων με εξωτερικό πρόγραμμα αριθμητικών υπολογισμών.....	22
2.3 Το πρόγραμμα Helm2D.....	23
2.4 Τα αρχεία διαμόρφωσης της εφαρμογής GiD – Helm2D.....	26
2.5 Οδηγίες εγκατάστασης της εφαρμογής.....	28
2.6 Το πρόγραμμα Helm3D.....	29
Κεφάλαιο 3^ο: Εφαρμογές.....	30
3.1 Εισαγωγή.....	30
3.2 Προβλήματα σε δύο διαστάσεις.....	31
3.2.1 Στάσιμο κύμα σε σωλήνα με κλειστό άκρο.....	32
3.2.2 Οδεύον κύμα σε σωλήνα.....	35
3.2.3 Ιδιορυθμοί σε ορθογώνιο δωμάτιο (δύο διαστάσεων).....	38
3.2.4 Κατανομή της πίεσης για διάφορες συχνότητες.....	42
3.2.5 Κατανομή πίεσης σε δωμάτιο δύο διαστάσεων με δύο πηγές.....	43
3.2.6 Κατανομή της πίεσης σε δωμάτιο με σχήμα ‘Γ’.....	43
3.3 Προβλήματα σε τρεις διαστάσεις.....	47

3.3.1	Εκπομπή κύματος από ακτινικά παλλόμενη σφαίρα.....	48
3.3.2	Μοντέλο ηχείου, εκπομπή από σφαιρικό πιστόνι σε σφαίρα.....	51
3.3.3	Εκπομπή από σφαιρικό δακτύλιο σε σφαίρα.....	55
3.3.4	Σκέδαση επίπεδου κύματος από μη ελαστική σφαίρα.....	62
3.3.5	Ιδιορυθμοί σε δωμάτιο τριών διαστάσεων.....	66
Βιβλιογραφία.....		71
Παράρτημα.....		72
A	Παράρτημα κεφαλαίου 3.....	72
B	Ρουτίνες για το πρόγραμμα Helm2D (Παράρτημα κεφ. 2).....	76
Γ	Αρχεία διαμόρφωσης για την εφαρμογή GiD – Helm2D.....	79
Δ	Ρουτίνες για το πρόγραμμα Helm3D (Παράρτημα κεφ. 2).....	85
E	Αρχεία διαμόρφωσης για την εφαρμογή GiD – Helm3D.....	90

Εισαγωγή

Η μέθοδος των συνοριακών στοιχείων (Boundary Element Method) είναι μια υπολογιστική τεχνική που εφαρμόζεται σε πολλά προβλήματα που περιγράφονται από φυσικούς νόμους και μπορούν να αναπαρασταθούν μαθηματικά με μερικές διαφορικές εξισώσεις. Οι αριθμητικές μέθοδοι σαν την Μέθοδο των Συνοριακών Στοιχείων (ΜΣΣ) είναι χρήσιμες και για την επίλυση ακουστικών προβλημάτων, ιδιαίτερα στις περιπτώσεις όπου δεν είναι εύκολη η αναλυτική τους λύση. Η τεχνική αυτή προέρχεται από την διακριτοποίηση της αντίστοιχης ολοκληρωματικής εξίσωσης που είναι ισοδύναμη με την μερική διαφορική εξίσωση που περιγράφει το πρόβλημα. Το πλεονέκτημα της μεθόδου αυτής εντοπίζεται στο γεγονός ότι μόνο οι συνοριακές επιφάνειες ενός χωρίου χρειάζονται να διαιρεθούν σε διακριτά στοιχεία, σε αντίθεση με άλλες μεθόδους όπως η FEM (Finite Element Method) όπου απαιτείται η διακριτοποίηση ολόκληρου του χωρίου. Γι' αυτήν την εργασία χρησιμοποιήθηκαν σαν βάση δύο ξεχωριστοί κώδικες για να λυθούν προβλήματα ακουστικής στις δύο αλλά και στις τρεις διαστάσεις. Με την μέθοδο των συνοριακών συνθηκών, χρησιμοποιώντας το δικτύωμα της επιφανείας ενός χωρίου (αντικειμένου) μπορούν να βρεθούν λύσεις της κυματικής εξίσωσης του Helmholtz για σημεία πάνω στα σύνορα αλλά και στη περιοχή μέσα ή έξω από το χωρίο. Σκοπός της εργασίας είναι μια ολοκληρωμένη εφαρμογή, που θα αναλαμβάνει την σχεδίαση και δικτύωση αντικειμένων, μέσω ενός γραφικού περιβάλλοντος (GUI), την λύση του προβλήματος και την γραφική απεικόνιση των αποτελεσμάτων. Οι εφαρμογές που θα παρουσιαστούν στην εργασία αυτή σχετίζονται με προβλήματα όπως η διάδοση, η εκπομπή και η σκέδαση ηχητικών κυμάτων.

Κεφάλαιο 1^ο

Η Μέθοδος των συνοριακών στοιχείων

1.1 Εισαγωγή

Οι αριθμητικές μέθοδοι βοηθούν τον μηχανικό στην μελέτη της συμπεριφοράς κατασκευών μέσω της χρήσης ηλεκτρονικού υπολογιστή. Τα προτερήματα είναι προφανή, χαμηλό κόστος της αριθμητικής έναντι της πειραματικής μεθόδου. Επικρατέστερες μέχρι σήμερα είναι, η μέθοδος πεπερασμένων στοιχείων (FEM) και η μέθοδος των συνοριακών στοιχείων (BEM). Σ' αυτό το κεφάλαιο παρουσιάζονται θεμελιώδεις αρχές της ακουστικής ξεκινώντας με την ολοκληρωτική εξίσωση του Helmholtz, καθώς και άλλα χρήσιμα μεγέθη τα οποία θα χρησιμοποιηθούν στην συνέχεια της εργασίας. Στην συνέχεια παρουσιάζονται οι αρχές της μεθόδου BEM για δισδιάστατα και τρισδιάστατα προβλήματα, όπου και αναλύονται οι απαραίτητες μαθηματικές σχέσεις για την κατανόηση της μεθόδου. Αναλύονται στάδια της μεθόδου, όπως ο μετασχηματισμός διαφορικών συναρτήσεων και η διακριτοποίηση χώρων, τα οποία είναι απαραίτητα για την υλοποίηση της μεθόδου. Σκοπός του κεφαλαίου αυτού, είναι η εξοικείωση του αναγνώστη με την μέθοδο αυτή, ή γνωριμία με τις βασικές έννοιες και εξισώσεις, ώστε να είναι σε θέση να χρησιμοποιήσει την μέθοδο.

1.2 Βασικές έννοιες και χρήσιμοι τύποι

Εδώ θα παρουσιαστούν μαθηματικές σχέσεις που θα χρησιμοποιηθούν παρακάτω και είναι απαραίτητες για την θεωρητική ανάπτυξη και κατανόηση της μεθόδου των συνοριακών στοιχείων (BEM).

1.2.1 Το θεώρημα Gauss – Green

Το θεώρημα αυτό αποτελεί θεμελιώδη σχέση που συνδέει το πεδιακό ολοκλήρωμα της παραγώγου μίας συναρτήσεως πάνω σ' ένα χωρίο, με το ολοκλήρωμα της συναρτήσεως στο σύνορο του πεδίου. Το πεδίο μπορεί να είναι δισδιάστατο ή τρισδιάστατο. Για την εξαγωγή της εν λόγω σχέσεως στις δύο διαστάσεις θεωρούμε το επίπεδο χωρίο του σχήματος 1.1. Θα διατυπώσουμε τη σχέση πρώτα για την παράγωγο της συναρτήσεως $f(x, y)$ ως προς x . Το πεδιακό ολοκλήρωμα μπορεί να γραφεί ως διπλό ολοκλήρωμα, πρώτα ως προς x και κατόπιν ως προς y . Θα έχουμε:

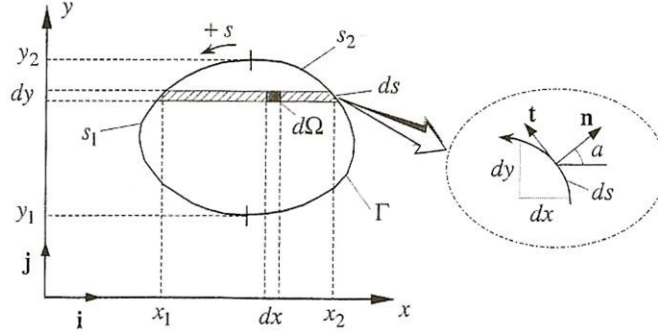
$$\int_{\Omega} f_x d\Omega = \int_{y_1}^{y_2} \left(\int_{x_1}^{x_2} \frac{\partial f}{\partial x} dx \right) dy = \int_{y_1}^{y_2} \{f(x_2, y) - f(x_1, y)\} dy \quad (1.1)$$

όπου $x_1 = x_1(y)$ και $x_2 = x_2(y)$

Επίσης έχουμε:

$$\frac{dy}{ds} = \cos \alpha = n_x \rightarrow dy = n_x ds$$

$$-\frac{dx}{ds} = \sin \alpha = n_y \rightarrow dx = -n_y ds$$



Σχήμα 1.1

όπου n_x και n_y είναι οι συνιστώσες του μοναδιαίου διανύσματος \mathbf{n} , καθέτου προς το σύνορο Γ . Το αρνητικό πρόσημο στην παραπάνω σχέση οφείλεται στο γεγονός ότι για $ds > 0$ συνεπάγεται $dx > 0$.

Επομένως μπορούμε να γράψουμε:

$$\int_{y_1}^{y_2} \{f(x_2, y) - f(x_1, y)\} dy = \int_{s_2} f(x_2, y) n_x ds - \int_{s_1} f(x_1, y) n_x ds \quad (1.2)$$

Στην παραπάνω σχέση το τόξο s_1 διαγράφεται κατά την αρνητική φορά (ωρολογιακή), όταν το y μεταβάλλεται από y_1 και το y_2 . Επομένως, θεωρώντας ενιαία φορά για το s , η σχέση (1.1), άρα και η (1.2), μπορεί να γραφεί ως

$$\int_{\Omega} \frac{\partial f}{\partial x} d\Omega = \int_{\Gamma} f n_x ds \quad (1.3)$$

Εναλλάσσοντας το ρόλο των x και y στην ανωτέρω σχέση λαμβάνουμε

$$\int_{\Omega} \frac{\partial f}{\partial y} d\Omega = \int_{\Gamma} f n_y ds \quad (1.4)$$

Εάν g είναι επίσης η συνάρτηση των x και y τότε οι σχέσεις (1.3) και (1.4) δίδουν

$$\int_{\Omega} g \frac{\partial f}{\partial x} d\Omega = - \int_{\Omega} f \frac{\partial g}{\partial x} d\Omega + \int_{\Gamma} f g n_x ds \quad (1.5)$$

$$\int_{\Omega} g \frac{\partial f}{\partial y} d\Omega = - \int_{\Omega} f \frac{\partial g}{\partial y} d\Omega + \int_{\Gamma} f g n_y ds \quad (1.6)$$

οι δύο αυτές σχέσεις εκφράζουν την παραγοντική ολοκλήρωση στις δύο διαστάσεις και είναι γνωστές ως *θεώρημα Gauss-Green*.

1.2.2 Η δεύτερη ταυτότητα του Green

Θεωρούμε τις συναρτήσεις $u = u(x, y)$ και $v = v(x, y)$. Εφαρμόζουμε τη σχέση (1.5) για $g = v$ και $f = \frac{\partial u}{\partial x}$ και την σχέση (1.6) για $g = v$ και $f = \frac{\partial u}{\partial y}$. Ακολουθώντας προσθέτοντας κατά μέλη λαμβάνουμε.

$$\int_{\Omega} v \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) d\Omega = - \int_{\Omega} \left(\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} \right) d\Omega + \int_{\Gamma} v \left(\frac{\partial u}{\partial x} n_x + \frac{\partial u}{\partial y} n_y \right) ds \quad (1.7)$$

Ομοίως εφαρμόζουμε τη σχέση (1.5) για $g = u$, $f = \frac{\partial v}{\partial x}$ και την (1.6) για $g = u$, $f = \frac{\partial v}{\partial y}$ και προσθέτοντας κατά μέλη λαμβάνουμε

$$\int_{\Omega} u \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) d\Omega = - \int_{\Omega} \left(\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} \right) d\Omega + \int_{\Gamma} u \left(\frac{\partial v}{\partial x} n_x + \frac{\partial v}{\partial y} n_y \right) ds \quad (1.8)$$

Αφαιρώντας τις σχέσεις (1.7) και (1.8) κατά μέλη έχουμε

$$\int_{\Omega} (v \nabla^2 u - u \nabla^2 v) d\Omega = \int_{\Gamma} \left(v \frac{\partial u}{\partial n} - u \frac{\partial v}{\partial n} \right) ds \quad (1.9)$$

όπου ∇^2 είναι ο τελεστής Laplace ή ο αρμονικός τελεστής

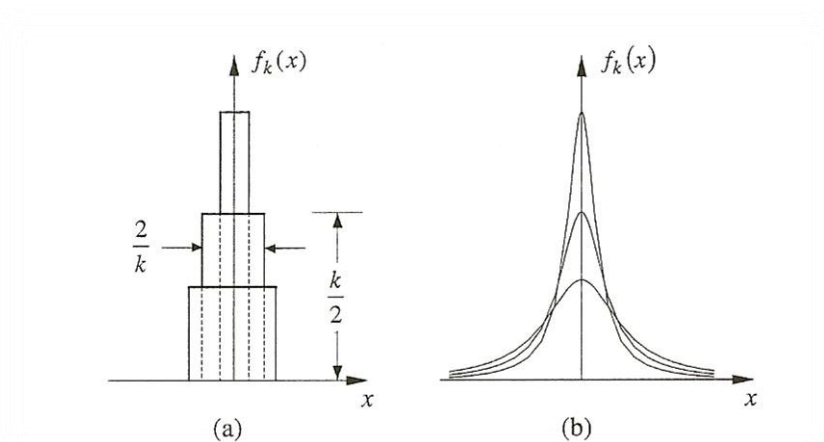
$$\nabla^2 \equiv \nabla \cdot \nabla = \left(\frac{\partial}{\partial x} i + \frac{\partial}{\partial y} j \right) \cdot \left(\frac{\partial}{\partial x} i + \frac{\partial}{\partial y} j \right) = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (1.10)$$

και $\frac{\partial}{\partial n}$ είναι ο τελεστής που παράγει την παράγωγο μίας συναρτήσεως κατά τη κατεύθυνση του διανύσματος \mathbf{n} .

Η σχέση (1.9) είναι γνωστή ως η δεύτερη ταυτότητα του Green για τον αρμονικό τελεστή ή ως ταυτότητα αμοιβαιότητας του Green.

1.2.3 Η συνάρτηση Δέλτα (δ) ή Dirac.

Η συνάρτηση δέλτα είναι μία μαθηματική έκφραση που παρουσιάστηκε από το θεωρητικό φυσικό Paul Dirac. Είναι μία συνάρτηση η οποία αναπαριστά μία απείρως στενή κορυφή με μοναδιαίο πλάτος. Μία συνάρτηση $\delta(x)$ που έχει την τιμή μηδέν παντού εκτός από $x = 0$ όπου η τιμή του είναι απείρως μεγάλη ώστε η συνολική ολοκλήρωση είναι 1.



Σχήμα 1.2: Η συνάρτηση Δέλτα

1.3 Η θεμελιώδης εξίσωση του Helmholtz

Η μέθοδος των συνοριακών συνθηκών παράγεται από την διακριτοποίηση της ολοκληρωτικής εξίσωσης του Helmholtz, που είναι μαθηματικά ισάξια της αυθεντικής μερικώς διαφορική εξίσωσης. Μια μεγάλη ποικιλία προβλημάτων στην ακουστική ανήκουν στη διάδοση χρονο-αρμονικών κυμάτων σε μία συνεχή συχνότητα ω . Όταν τέτοια κύματα προκύψουν σε ένα ομογενές μέσο διάδοσης, τότε η ακουστική πίεση P , σε όποιο σημείο, αλλάζει ημιτονοειδώς στην ίδια συχνότητα ω .

$$P = p e^{i\omega t}, \quad (1.11)$$

όπου p είναι το μιγαδικό πλάτος της πίεσης σε οποιοδήποτε σημείο. Τελικά η μερικώς διαφορική εξίσωση, γνωστή και ως η εξίσωση του Helmholtz, που μας δίνει λύσεις της πίεσης είναι:

$$\nabla^2 p + k^2 p = 0, \quad (1.12)$$

όπου ο $k=\omega/c$ είναι ο κυματάριθμος.

1.4 Συνοριακές συνθήκες

Οι συνθήκες για την επίλυση της εξίσωσης (1.12) κατηγοριοποιούνται σε ενεργές και παθητικές. Ενεργές συνοριακές συνθήκες είναι αυτές για τις οποίες είτε η κίνηση ή η ακουστική πίεση στα σύνορα, ή μέρος αυτών, είναι γνωστές. Παραδείγματος χάριν εάν θέλουμε να υπολογίσουμε την πίεση σ' ένα σημείο, μέσα σε ένα δωμάτιο, εξ' αιτίας μιας δονούμενης επιφάνειας, τότε η αρμονική κίνηση της επιφάνειας αυτής θα πρέπει να είναι γνωστή. Δηλαδή η συνθήκη που περιγράφει την ταχύτητα που κινείται μέρος ή το σύνολο των συνόρων είναι :

$$\frac{\partial p}{\partial n} = -i\omega\rho_0 v_n, \quad (1.13)$$

όπου n είναι το μοναδιαίο διάνυσμα της δονούμενης επιφάνειας. Η εξίσωση (1.13) αναφέρεται και ως *συνοριακή συνθήκη της ταχύτητας (velocity boundary condition)*.

Μια δεύτερη ενεργή συνοριακή συνθήκη προκύπτει όταν η ακουστική πίεση είναι ήδη γνωστή για μέρος των συνόρων. Τότε μπορούμε να πούμε πως η συνθήκη πίεσης, ή *συνοριακή συνθήκη της πίεσης (pressure boundary condition)* είναι:

$$p = p_e \quad (1.14)$$

Μια παθητική συνοριακή συνθήκη προκύπτει όταν ο ήχος ανακλάται από μία επιφάνεια (π.χ. από ένα απορροφητικό υλικό), που είναι μέρος των συνόρων. Όταν συμβεί αυτό, τότε το πλάτος και η φάση του ανακλώμενου κύματος εξαρτώνται από την ακουστική σύνθετη αντίσταση της απορροφητικής επιφάνειας η οποία είναι :

$$Z = \frac{p}{v_n} \quad (1.15)$$

όπου p και v είναι η πίεση και σωματιδιακή ταχύτητα, αντίστοιχα, πάνω στην επιφάνεια. Η παραπάνω συνθήκη λέγεται και *συνοριακή συνθήκη σύνθετης αντίστασης (impedance boundary condition)*.

1.5 Η ολοκληρωτική εξίσωση του Helmholtz για προβλήματα εσωτερικού τύπου.

Εδώ παρουσιάζεται η ανάπτυξη της μεθόδου των συνοριακών στοιχείων για προβλήματα που διέπονται από την εξίσωση του Helmholtz. Πρώτα θα αναζητήσουμε λύσεις σ' ένα κλειστό επίπεδο χωρίο V που περιγράφεται από ένα σύνορο S , όπως φαίνεται στο σχήμα 1.3

Ξεκινώντας από την διαφορική εξίσωση του Helmholtz:

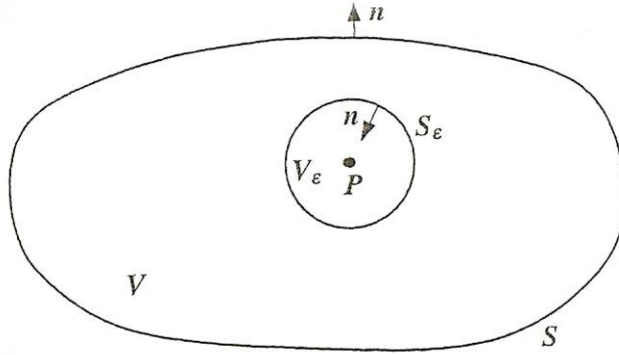
$$\nabla^2 p + k^2 p = 0 \quad (1.16)$$

όπου k είναι ο κυματάρριθμος και p η ακουστική πίεση.

$$\nabla^2 \psi + k^2 \psi = -\delta(Q - P) \quad (1.17)$$

όπου $\delta(Q - P)$ είναι η Dirac delta συνάρτηση σε οποιοδήποτε σημείο Q στον χώρο με μία σημειακή πηγή στο P , και ψ είναι η θεμελιώδης λύση στο Q . Για να λύσουμε την εξίσωση υιοθετούμε σύστημα σφαιρικών συντεταγμένων (r, θ, φ) με κέντρο την πηγή P . Ζητώντας μία συμμετρική θεμελιώδης λύση, η εξίσωση γίνεται

$$\frac{d^2 \psi}{dr^2} + \frac{2}{r} \frac{d\psi}{dr} + k^2 \psi = 0 \quad \text{οπουδήποτε εκτός από } P \quad (1.18)$$



Σχήμα 1.3: Ακουστικό πεδίο V με κλειστό σύνορο S .

Η γενική λύση της εξίσωσης (1.17) είναι

$$\psi = A \frac{e^{-ikr}}{r} + B \frac{e^{ikr}}{r}, \quad (1.19)$$

όπου A και B είναι απροσδιόριστοι συντελεστές. Αν υιοθετήσουμε την $e^{i\omega t}$ σύμβαση, ο πρώτος όρος αντιπροσωπεύει το εξερχόμενο κύμα και ο δεύτερος το εισερχόμενο. Αν $B = 0$ επειδή P είναι σημειακή πηγή. Ο συντελεστής A μπορεί να

προσδιοριστεί ολοκληρώνοντας την εξ.(1.19) πάνω σε ένα μικρό σφαιρικό όγκο Ω_ε που εμπεριέχει την πηγή P . Αυτό έχει σαν αποτέλεσμα :

$$\lim_{\varepsilon \rightarrow 0} \int_{\partial\Omega_\varepsilon} \frac{\partial y}{\partial n} \partial S = -1, \quad (1.20)$$

όπου $\partial\Omega_\varepsilon$ είναι η συνοριακή επιφάνεια του μικροσκοπικού σφαιρικού όγκου Ω_ε και n το μοναδιαίο διάνυσμα που είναι στραμμένο προς τα έξω και κάθετο στο $\partial\Omega_\varepsilon$. Διαφορίζοντας το ψ στην κατεύθυνση r και κάνοντας την ολοκλήρωση της 1.20 σε σφαιρικό σύστημα συντεταγμένων παίρνουμε $A = \frac{1}{4\pi}$. Τελικά, η θεμελιώδης λύση για τον τρισδιάστατο ελεύθερο χώρο είναι:

$$\psi = \frac{e^{-ikr}}{4\pi r}. \quad (1.21)$$

Για ένα εσωτερικό πρόβλημα, σκοπός είναι να λύσουμε την εξίσωση Helmholtz σε μία κοιλότητα V πεπερασμένων διαστάσεων. Οι συνοριακές συνθήκες (πίεση, ταχύτητα, ακουστική εμπέδηση) πάνω στην συνοριακή επιφάνεια S της κοιλότητας μπορεί να αναπαρασταθεί από την γενική έκφραση

$$\alpha p + \beta v_n = \gamma, \quad \text{πάνω στο } S \quad (1.22)$$

όπου v_n είναι η σωματιδιακή ταχύτητα, και α , β και γ είναι οι σταθερές που δηλώνονται από μας.

Για να εξάγουμε τις συνοριακές ολοκληρωτικές εξισώσεις, ξεκινάμε με την δεύτερη ταυτότητα του Green η οποία εφαρμόζεται στις συναρτήσεις p και ψ . Αν υποθέσουμε ότι το σημείο P , που συνδέεται με τη θεμελιώδη λύση ψ , βρίσκεται μέσα στον ακουστικό χώρο V , τότε ένας μικροσκοπικός σφαιρικός όγκος V_ε εξαιρείται από το χώρο V , επειδή εμπεριέχει το σημείο P που απειρίζεται το ψ .

Η εφαρμογή της δεύτερης ταυτότητας του Green έχει σαν αποτέλεσμα,

$$\int_{V-V_\varepsilon} (\psi \nabla^2 p - p \nabla^2 \psi) dV = \int_{S+S_\varepsilon} \left(\psi \frac{\partial p}{\partial n} - p \frac{\partial \psi}{\partial n} \right) dS \quad (1.23)$$

όπου το κάθετο μοναδιαίο διάνυσμα n είναι στραμμένο προς τα έξω του χώρου V . Αφού το μοναδιαίο σημείο P εξαιρείται από τον χώρο, τότε έχουμε $\nabla^2 p = -k^2 p$ και $\nabla^2 \psi = -k^2 \psi$. Έτσι το αριστερό μέρος της εξίσωσης (1.23) είναι ίσο με μηδέν. Το δεξιό μέρος είναι μία συνοριακή ολοκλήρωση που υπολογίζεται πάνω σε δύο συνοριακές επιφάνειες, ή μία είναι το τοίχωμα του χώρου V και η άλλη μικρότερη που εσωκλείει το σημείο P

Έτσι έχουμε:

$$\lim_{\varepsilon \rightarrow 0} \int_{S_\varepsilon} \psi \frac{\partial p}{\partial n} dS = 0 \quad (1.24)$$

Κάνοντας την ολοκλήρωση σε σφαιρικές συντεταγμένες και θεωρώντας το όριο $r \rightarrow 0$ τότε φαίνεται ότι

$$\lim_{\varepsilon \rightarrow 0} \int_{S_\varepsilon} p \frac{\partial \psi}{\partial n} dS = p(P) \lim_{\varepsilon \rightarrow 0} \int_{S_\varepsilon} p \frac{\partial \psi}{\partial n} dS = p(P) \quad (1.25)$$

Αντικαθιστώντας τις εξισώσεις (1.24) και (1.25) στην εξίσωση (1.23) μας δίνει:

$$p(P) = \int_S \left(\psi \frac{\partial p}{\partial n} - p \frac{\partial \psi}{\partial n} \right) dS, \quad P \text{ μέσα στο } S \quad (1.26)$$

Αν θυμηθούμε πως η ακουστική πίεση συνδέεται με την σωματιδιακή ταχύτητα εξ.(1.13):

$$\frac{\partial p}{\partial n} = -i\rho\omega v_n,$$

όπου ρ είναι η πυκνότητα του μέσου διάδοσης και ω η γωνιακή συχνότητα, η εξίσωση (1.26) θα γίνει

$$p(P) = - \int_S \left(i\rho\omega v_n \psi + p \frac{\partial \psi}{\partial n} \right) dS \quad (1.27)$$

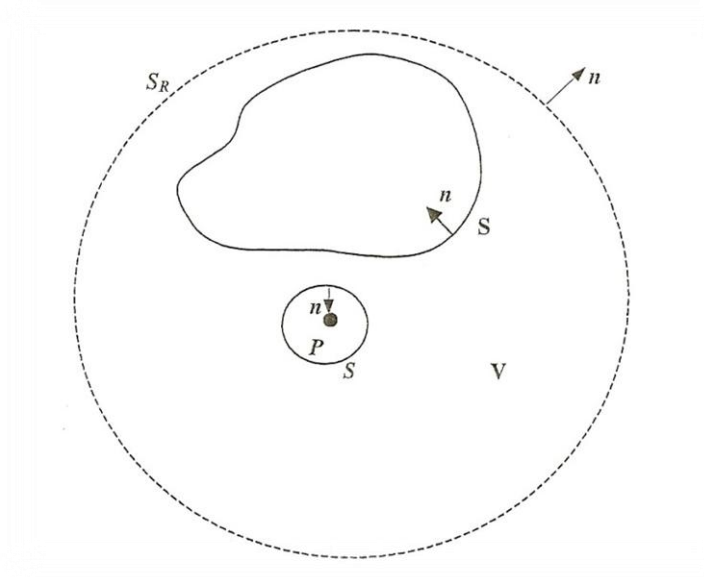
Η παραπάνω εξίσωση δηλώνει ότι η ακουστική πίεση p σε ένα οποιοδήποτε σημείο P μπορεί να προσδιοριστεί ολοκληρώνοντας την εξίσωση στο σύνορο. Αυτή είναι και η βασική ιδέα της μεθόδου των συνοριακών στοιχείων, επειδή μόνο η πληροφορία των συνόρων είναι αναγκαία για την εξαγωγή της λύσης.

1.6 Η ολοκληρωτική εξίσωση του Helmholtz για προβλήματα εξωτερικού τύπου.

Για ένα πρόβλημα εξωτερικού τύπου ο σκοπός είναι να λυθεί η θεμελιώδης εξίσωση σε ένα χωρίο V άνευ συνόρων, όπως φαίνεται στο σχήμα 1.4, εξ' αιτίας ακουστικής εκπομπής από ένα δονούμενο σώμα με συνοριακή επιφάνεια S . Το μοναδιαίο διάνυσμα n στην επιφάνεια S είναι στραμμένο προς τα μέσα. Όπως και στα εσωτερικά προβλήματα οι συνοριακές συνθήκες μπορούν να οριστούν από την γενική έκφραση της εξίσωσης (1.22).

Ακολουθώντας την ίδια πορεία όπως στα προβλήματα εσωτερικού τύπου, δηλαδή την εφαρμογή της δεύτερης ταυτότητας του Green στα p και ψ και μετά ψάχνοντας λύσεις του p με όριο $r \rightarrow \infty$ αυτή την φορά, εφ' όσον το ακουστικό χωρίο δεν έχει σύνορα. Τότε θα προκύψει η συνοριακή ολοκληρωτική εξίσωση που θα έχει την μορφή

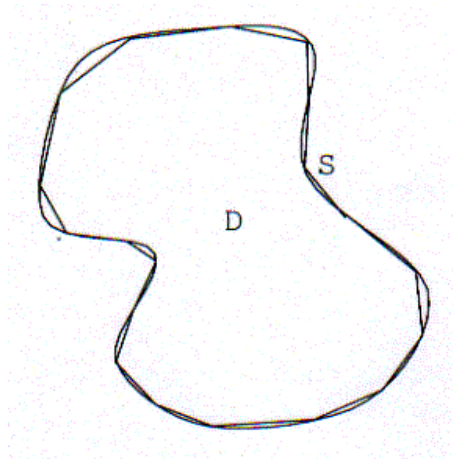
$$C(P)p(P) = - \int_S (i\rho\omega n \psi + p \frac{\partial \psi}{\partial n}) dS, \quad (1.28)$$



Σχήμα 1.4: Ακουστικό πεδίο V χωρίς σύνορα και δονούμενο σώμα με σύνορο S

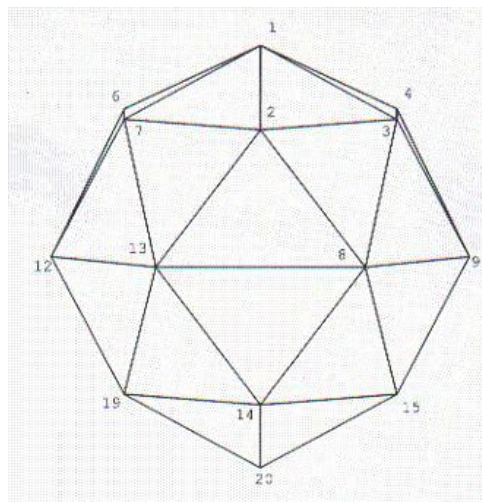
1.7 Διακριτοποίηση συνόρων

Στη για να λυθεί αριθμητικά η ολοκληρωτική εξίσωση του Helmholtz, είναι απαραίτητη η διακριτή αναπαράσταση του χώρου, είτε αυτός ανήκει στον δισδιάστατο ή στον τρισδιάστατο χώρο. Γενικά αυτό γίνεται εφικτό αν οριστεί μια επιφάνεια ως ένα σύνολο μικρότερων πλαισίων, ή μία ευθεία ως σύνολο μικρότερων ευθειών. Για παράδειγμα στο σχήμα 1.5 φαίνεται η αναπαράσταση ενός κλειστού δισδιάστατου συνόρου S ως ένα σύνολο από ευθείες γραμμές.



Σχήμα 1.5: Δισδιάστατο χωρίο ως άθροισμα γραμμών

Αν S είναι το αρχικό σύνορο τότε $\Delta S'_j$ (για $j = 1, 2, \dots, n$) είναι τα πλαίσια που αναπαριστούν μια προσέγγιση του S στην μέθοδο των συνοριακών στοιχείων. Δηλαδή όσο το n αυξάνει το $S' \approx S$. Αντίστοιχα στο σχήμα 1.6 βλέπουμε την αναπαράσταση μίας σφαίρας ως άθροισμα τριγωνικών πλαισίων. Αυτού του είδους η αναπαράσταση των συνόρων είναι το πρώτο βήμα στη διακριτοποίηση των τελεστών που προκύπτουν από την μορφοποίηση της κυματικής εξίσωσης του Helmholtz. Η ολοκλήρωση πάνω σε κάθε ένα απ' αυτά τα πλαίσια μπορεί να γενικευτεί, εφ' όσον όλα έχουν παρόμοια χαρακτηριστικά.



Σχήμα 1.6: Τρισδιάστατο χωρίο ως άθροισμα επιφανειών

1.7.1 Δισδιάστατα σύνορα

Η διακριτοποίηση δισδιάστατων αντικειμένων στο χώρο γίνεται διαιρώντας τις γραμμές που καθορίζουν το σύνορο σε ξ στοιχεία (elements). Τα στοιχεία μπορούν να είναι τριών ειδών α) γραμμικά, β) τετραγωνικά ή γ) κυβικά, κάτι που σημαίνει πως θα περιγράφονται από 2 ή 3 ή 4 κομβικά σημεία (nodes) αντίστοιχα. Η γεωμετρία του κάθε στοιχείου δίδεται από τη θέση των κομβικών σημείων, δηλαδή:

$$x = \sum_{i=1}^n x_i N_i$$

$$y = \sum_{i=1}^n y_i N_i$$

όπου x, y είναι οι συντεταγμένες στις κορυφές, $N_i(\xi)$ είναι οι *συναρτήσεις σχήματος*, που ορίζονται πάνω στο στοιχείο με τοπική συντεταγμένη $-1 \leq \xi \leq 1$ και n είναι ο αριθμός των κόμβων πάνω στο στοιχείο (element). Η σχεδίαση των στοιχείων φαίνεται στο σχήμα 1.5. Οι συναρτήσεις αυτές δίδονται παρακάτω:

Γραμμικά στοιχεία (linear elements):

$$N_1 = \frac{1}{2}(1 - \xi)$$

$$N_2 = \frac{1}{2}(1 + \xi)$$

Τετραγωνικά στοιχεία (Quadratic elements):

$$N_1 = \frac{1}{2}\xi(\xi - 1)$$

$$N_2 = \frac{1}{2}(1 - \xi)(1 + \xi)$$

$$N_3 = \frac{1}{2}\xi(\xi + 1)$$

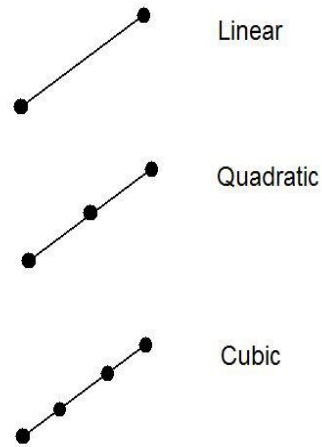
Κυβικά στοιχεία (Cubic elements):

$$N_1 = \frac{9}{16}\left(\frac{1}{9} - \xi^2\right)(\xi - 1)$$

$$N_2 = \frac{27}{16}(1 - \xi^2)\left(\frac{1}{3} - \xi\right)$$

$$N_3 = \frac{27}{16}(1 - \xi^2)\left(\frac{1}{3} + \xi\right)$$

$$N_4 = \frac{9}{16}\left(\frac{1}{9} - \xi^2\right)(\xi + 1)$$



Σχήμα 1.7 : Τα διαφορετικά είδη στοιχείων.

Έτσι όλη η συνοριακή γεωμετρία μπορεί να αναπαρασταθεί σαν σύνολο από τις παραπάνω συναρτήσεις :

$$d\Gamma = \sqrt{(dx)^2 + (dy)^2} = \sqrt{\left(\frac{dx}{d\xi}\right)^2 + \left(\frac{dy}{d\xi}\right)^2} d\xi = Jd\xi$$

όπου J είναι η Jacobian, και

$$\frac{dx}{d\xi} = \sum_{i=1}^n x_i \frac{dN_i}{d\xi}$$

$$\frac{dy}{d\xi} = \sum_{i=1}^n y_i \frac{dN_i}{d\xi}$$

Συνοριακές συνθήκες

Οι συνοριακές συνθήκες πίεσης και ταχύτητας μπορούν να αναπαρασταθούν με τις συναρτήσεις σχήματος που περιγράφουν την γεωμετρία δηλαδή:

$$p = \sum_{i=1}^n p_i N_i(\xi)$$

$$v_n = \sum_{i=1}^n v_{ni} N_i(\xi)$$

όπου p_i και v_i είναι οι τιμές της ακουστικής πίεσης και της σωματιδιακής ταχύτητας, αντίστοιχα, και N_i είναι οι ίδιες συναρτήσεις που χρησιμοποιούνται παραπάνω.

Η συνοριακή συνθήκη σε κάθε μία κορυφή του κάθε στοιχείου είναι της παρακάτω μορφής:

$$(CA)p + (CB)v_n = CC$$

όπου CA , CB , CC είναι σταθερές που δηλώνονται από τον χρήστη.

1.7.2 Τρισδιάστατα σύνορα

Είδαμε την ολοκληρωτική εξίσωση του Helmholtz παραπάνω. Για να την λύσουμε για χώρους τριών διαστάσεων πρέπει να τους κάνουμε διακριτούς με στοιχεία (elements) τεσσάρων (τετραγωνικά) ή τριών κορυφών (τριγωνικά). Η γεωμετρία των στοιχείων αυτών περιγράφεται:

$$x = \sum_{i=1}^n x_t N_t(\xi_1, \xi_2)$$

$$y = \sum_{i=1}^n y_t N_t(\xi_1, \xi_2)$$

$$z = \sum_{i=1}^n z_t N_t(\xi_1, \xi_2)$$

όπου x , y και z είναι οι συντεταγμένες στα σημεία κορυφών, $N(\xi, \xi)$ είναι οι γραμμικές εξισώσεις σχήματος που ορίζουν καθ' ένα στοιχείο, και n είναι ο αριθμός των κορυφών (3 ή 4) ανά στοιχείο. Οι εξισώσεις σχήματος φαίνονται παρακάτω καθώς και στο σχήμα 1.8:

Τετραγωνικά στοιχεία τεσσάρων κορυφών

$$N_1 = \frac{1}{4}(1 - \xi_1)(1 - \xi_2)$$

$$N_2 = \frac{1}{4}(1 + \xi_1)(1 - \xi_2)$$

$$N_3 = \frac{1}{4}(1 + \xi_1)(1 + \xi_2)$$

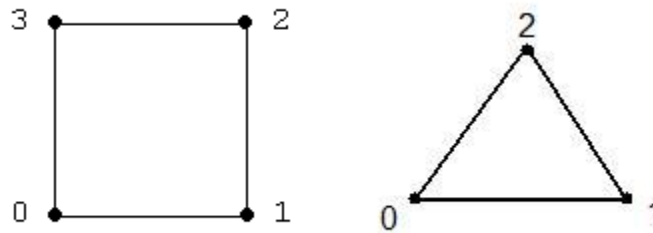
$$N_4 = \frac{1}{4}(1 - \xi_1)(1 + \xi_2)$$

Τριγωνικά στοιχεία τριών κορυφών

$$N_1 = \xi_1$$

$$N_2 = \xi_2$$

$$N_3 = 1 - \xi_1 - \xi_2$$



Σχήμα 1.8: Τετραγωνικό και τριγωνικό στοιχείο

Κεφάλαιο 2

Υλοποίηση εκπαιδευτικής εφαρμογής με χρήση λογισμικού σχεδίασης και δικτύωσης

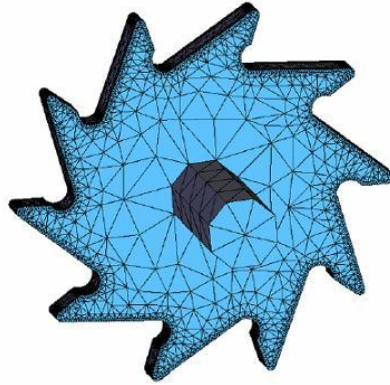
2.1 Εισαγωγή

Όπως είδαμε οι μέθοδοι πεπερασμένων και συνοριακών στοιχείων είναι ισχυρές υπολογιστικές μέθοδοι που μπορούν να βοηθήσουν τον μηχανικό στην επίλυση προβλημάτων. Πολλά λογισμικά έχουν αναπτυχθεί τα τελευταία τριάντα χρόνια, στα οποία αναπτύχθηκαν και οι μέθοδοι. Τα περισσότερα είναι εμπορικά και αρκετά εξειδικευμένα για να χρησιμοποιηθούν για εκπαιδευτικούς λόγους. Ένας από τους σκοπούς της εργασίας αυτής είναι και η ανάπτυξη εφαρμογών μέσω των οποίων οι σπουδαστές του κλάδου της ακουστικής θα έρθουν σε επαφή με την ΜΣΣ και τις δυνατότητες της και θα μπορούν να λύσουν ακουστικά προβλήματα. Υλοποιήθηκε μία εφαρμογή με την χρήση προγραμμάτων, που μπορεί ο σπουδαστής να προμηθευτεί μαζί με την αγορά του βιβλίου: 'Boundary Elements Acoustics Fundamentals and Computer Codes', T.W.Wu, [1]. Επίσης είναι αναγκαίο και το πρόγραμμα, GiD το οποίο μπορεί να βρεθεί από το διαδίκτυο στην τοποθεσία, gid.cimne.upc.es/ και να χρησιμοποιηθεί ελεύθερα η δοκιμαστική του έκδοση (η οποία έχει κάποιους περιορισμούς σε σχέση με την εμπορική). Απαιτούνται φυσικά και άλλα προγράμματα που αναπτύξαμε όπως κάποιες ρουτίνες που συμπληρώνουν την υλοποίηση της αριθμητικής μεθόδου (γραμμένα σε Fortran) και ειδικές ρουτίνες που χρησιμοποιούνται για την διασύνδεση των δύο προγραμμάτων.

2.2 Το λογισμικό σχεδίασης και διακριτοποίησης: GiD

Το GiD είναι ένα πρόγραμμα για γεωμετρική σχεδίαση, δικτύωση και διακριτοποίηση γεωμετριών, εισαγωγή και εξαγωγή δεδομένων για να γίνουν αριθμητικοί υπολογισμοί και τέλος απεικόνιση αποτελεσμάτων μέσα από το ίδιο το πρόγραμμα. Ο χρήστης μπορεί να σχεδιάζει ή να εισάγει έτοιμα γεωμετρικά μοντέλα από άλλα λογισμικά σχεδίασης, καθώς το GiD υποστηρίζει τους περισσότερους τύπους αρχείων άλλων εμπορικών προγραμμάτων (ACIS, IGES, DXF, κ.α.). Το πρόγραμμα GiD έχει τη δυνατότητα να παράγει και να εξάγει το δικτύωμα γεωμετριών σε υψηλή ανάλυση και με μία μεγάλη ποικιλία επιλογών. Με τη δημιουργία ειδικών αρχείων που συντάσσονται σε μια ειδική 'script' γλώσσα, το πρόγραμμα μπορεί να επικοινωνήσει με κώδικες επίλυσης. Μπορεί να γίνει εξαγωγή δεδομένων από την γεωμετρία και να γίνει εισαγωγή άλλων δεδομένων σε μεταβλητές που θα ορίζει ο χρήστης. Μέσα από απλά εργαλεία επικοινωνίας, που μπορεί να φτιάξει εύκολα ο χρήστης, μπορεί να γίνει εισαγωγή δεδομένων (όπως π.χ. συνοριακές συνθήκες). Διαθέτει επίσης το πρόγραμμα ένα εύχρηστο μετα-επεξεργαστή (postprocessor) για την επισκόπηση των αποτελεσμάτων, με πλήθος επιλογών, όπως γραφήματα, λίστες αποτελεσμάτων ή κινούμενες εικόνες (animation). Πριν αρχίσει η επίλυση των προβλημάτων ακουστικής, θα παρουσιαστούν απλά παραδείγματα σχεδιασμού και δικτύωσης γεωμετρικών

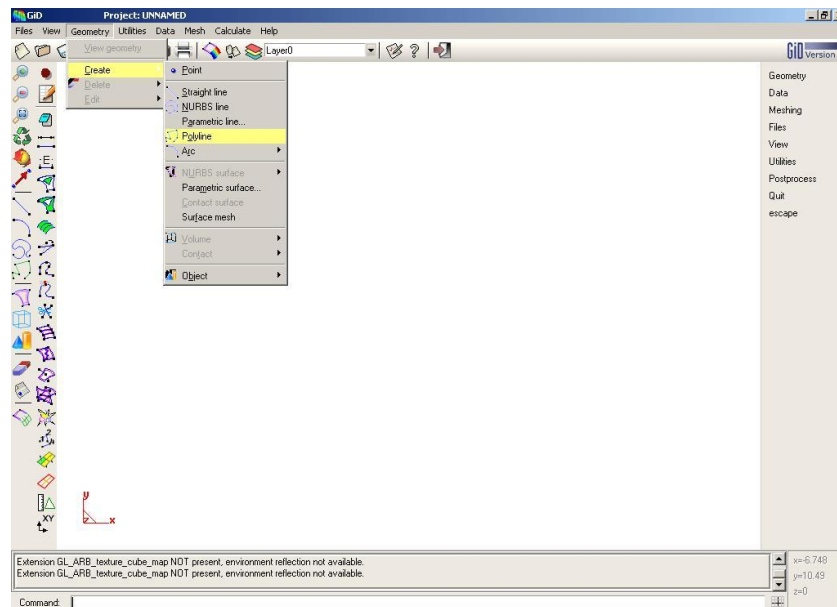
σχημάτων για την εξοικείωση του χρήστη με το πρόγραμμα. Για να πάρουμε μία εικόνα των δυνατοτήτων του προγράμματος παρουσιάζουμε στο σχήμα 2.1 το μοντέλο ενός γραναζιού όπως σχεδιάστηκε και δικτυώθηκε στο GiD.



Σχήμα 2.1 : Μοντέλο γραναζιού όπως σχεδιάστηκε και δικτυώθηκε στο GiD

2.2.1 Το GiD ως προεπεξεργαστής (preprocessor)

Το πρόγραμμα GiD όπως θα δούμε λειτουργεί σε δύο καταστάσεις. Στη πρώτη (προεπεξεργασία) μας δίνει την δυνατότητα σχεδιασμού γεωμετρικών σχημάτων και μοντέλων, την δικτύωση τους με διάφορες επιλογές και την εισαγωγή δεδομένων που αφορούν τους αριθμητικούς υπολογισμούς που σκοπεύουμε να εφαρμόσουμε στα μοντέλα. Στο παράδειγμα που ακολουθεί θα σχεδιάσουμε ένα επίπεδο παραλληλόγραμμο του οποίου μετ' έπειτα θα το δικτυώσουμε με τριγωνικά.

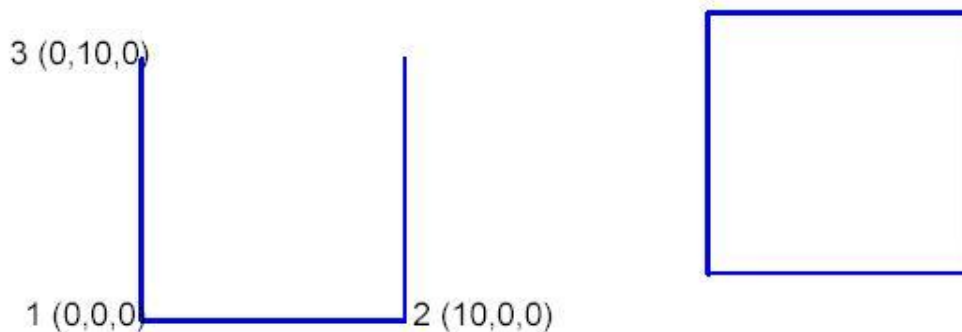


Σχήμα 2.2: Το πρόγραμμα GiD

Το GiD μπορεί να χειρισθεί τεσσάρων ειδών ‘αντικείμενα’, τα οποία ακολουθούν ιεραρχική προτεραιότητα, σημεία (points), γραμμές (lines), επιφάνειες (surfaces), όγκους (volumes). Η προτεραιότητα αναφέρεται στο ότι δεν μπορεί να υπάρξει μία επιφάνεια που δεν περιγράφεται από ένα σύνολο γραμμών ή π.χ. μία ευθεία γραμμή που να μην περιγράφεται από τουλάχιστον δύο σημεία. Από το παράθυρο **Geometry** μας δίνονται οι επιλογές δημιουργίας, επέμβασης και διαγραφής όλων των παραπάνω ‘αντικειμένων’. Θα δούμε με ένα απλό παράδειγμα τις βασικότερες επιλογές και δυνατότητες που μπορεί να μας προσφέρει το GiD.

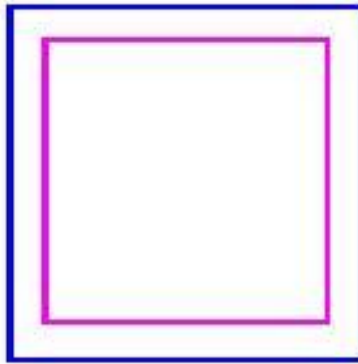
2.2.2 Σχεδιάζοντας στο GiD ένα παραλληλόγραμμα

Για να σχεδιάσουμε το παραλληλόγραμμα πρέπει πρώτα να εισάγουμε τα σημεία που περιγράφουν τις πλευρές του. Έτσι ανοίγουμε το **Utilities > Tools > Coordinates window** και μετά επιλέγουμε **Geometry > Create > Point**. Στο ‘coordinates window’ εισάγουμε τις συντεταγμένες $\langle 0,0,0 \rangle$ και επιλέγουμε **Apply**. Βλέπουμε πως το σημείο στην αρχή των αξόνων έχει ήδη σχεδιαστεί. Επαναλαμβάνουμε την διαδικασία αυτή για τα σημεία $\langle 0,10,0 \rangle$, $\langle 10,10,0 \rangle$, $\langle 10,0,0 \rangle$, εισάγοντάς τα στο ‘coordinates window’ και πατώντας **Apply** διαδοχικά. Τώρα μπορούμε να ενώσουμε τα σημεία αυτά με ευθείες γραμμές μέσω του **Geometry > Create > Straight line** και επιλέγοντας **Join** από την μπάρα επιλογών. Κατόπιν διαλέγουμε ένα – ένα τα σημεία για να τα ενώσουμε με γραμμές. Όταν τελειώσουμε με όλες τις τέσσερις γραμμές, πατάμε **Esc**.



Σχήμα 2.3: Σχεδιάζοντας ευθείες στο GiD

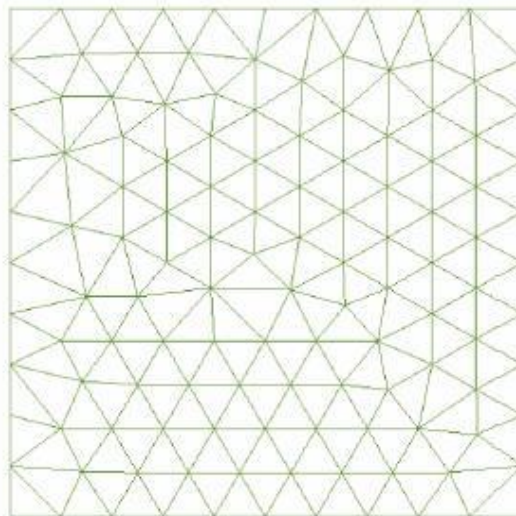
Τώρα θέλοντας να δημιουργήσουμε μία επιφάνεια από το παραλληλόγραμμα επιλέγουμε **Geometry > Create > Nurbs surface > By contour**. Με αυτή την τεχνική μας ζητείτε να επιλέξουμε τις γραμμές που ορίζουν την επιφάνεια. Διαλέγουμε με το ποντίκι όλες τις γραμμές και το αποτέλεσμα φαίνεται στο σχήμα 2.4 (η επιφάνεια υποδηλώνεται με πλαίσιο που πάντα έχει χρώμα ροζ). Πρέπει να πούμε πως εδώ, ότι το GiD παρέχει και άλλους τρόπους για την σχεδίαση αυτής της γεωμετρίας.



Σχήμα 2.4: Επιφάνεια με όρια ευθείες γραμμές

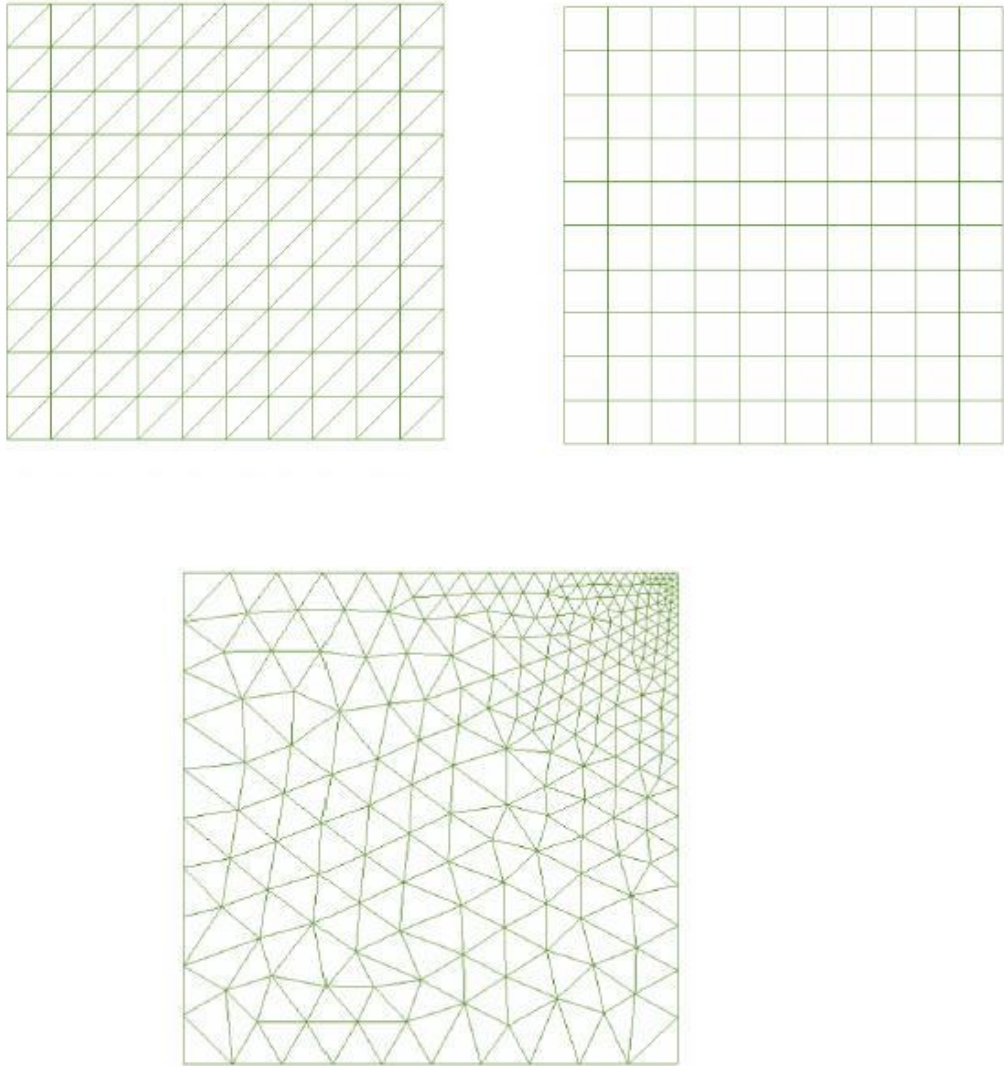
2.2.3 Δικτύωση γεωμετριών

Στον προεπεξεργαστή του GiD μπορούμε να φτιάξουμε και ένα δικτύωμα του παραλληλογράμμου που μόλις κατασκευάσαμε. Όλες οι επιλογές δικτύωσης βρίσκονται στο **Mesh**, στην μπάρα επιλογών. Μπορούμε να επιλέξουμε το είδος (Triangle, Quadrilateral, e.t.c.), το μέγεθος των στοιχείων και ποιά στοιχεία της γεωμετρίας (γραμμές, επιφάνειες, όγκοι) θα δικτυωθούν. Επιλέγουμε **Mesh>Generate mesh** και θα εμφανιστεί ένα παράθυρο που ζητάει να εισάγουμε το μέγεθος των στοιχείων (elements) του δικτύωματος. Αυτή η τιμή τελικά αναφέρεται στο εμβαδό που θα έχει κάθε στοιχείο, εφ' όσον θέλουμε να δικτυώσουμε μία επιφάνεια. Εισάγοντας την τιμή **<0.1>** και πατώντας **Ok** θα πάρουμε το παρακάτω αποτέλεσμα:



Σχήμα 2.5: Το δικτύωμα της επιφάνειας με τριγωνικά στοιχεία

Για την παρουσίαση των επιλογών που προσφέρονται για την δικτύωση, κατασκευάσαμε τα δικτύωματα που φαίνονται στο σχήμα 2.6. Αλλάξαμε τις παραμέτρους δικτύωσης, όπως το είδος των στοιχείων (Triangle, quadrilateral, Tetrahedral, Hexahedral, Prism), την συγκέντρωση των στοιχείων γύρω από ένα σημείο ή γραμμή και τον ισομερισμό των στοιχείων πάνω στην γεωμετρία.

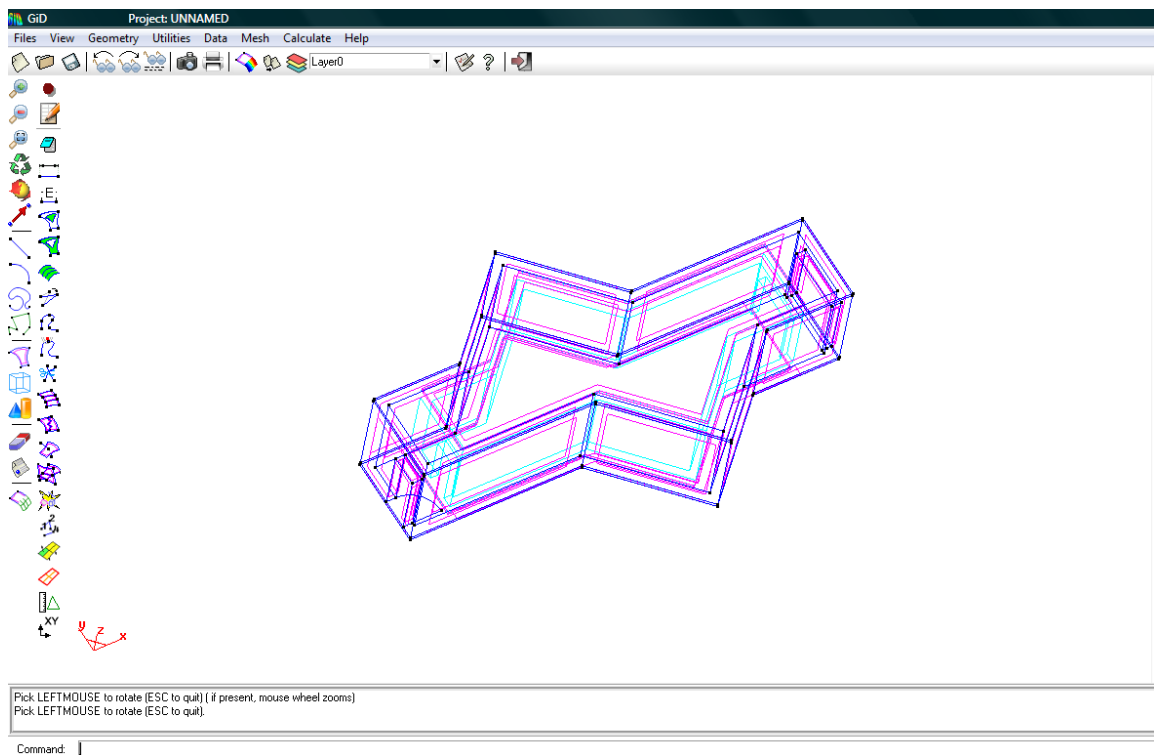


Σχήμα 2.6: Δικτύωματα με διάφορες επιλογές (τριγωνικά στοιχεία, τετραγωνικά, συγκέντρωση στοιχείων γύρω από ένα σημείο)

2.2.4 Εισάγοντας μοντέλα από άλλα format

Το GiD έχει την δυνατότητα να εισάγει και να επεξεργαστεί γεωμετρικά μοντέλα που έχουν σχεδιαστεί σε άλλα προγράμματα σχεδίασης όπως το AutoCAD ή το 3DstudioMax. Ας δούμε πως γίνεται αυτό με ένα απλό παράδειγμα:

Ξεκινάμε ένα καινούργιο 'project' στο GiD και το σώζουμε με το όνομα room.gid. Τώρα κάνουμε 'κλικ' στην επιλογή **Files > Import > DXF**. Θα μας ανοίξει ένας 'browser' από όπου και θα επιλέξουμε το αρχείο room.dxf που βρίσκεται στο CD-Rom που συνοδεύει την εργασία. Το GiD θα εισάγει την γεωμετρία και από την γραμμή εντολών θα μας ενημερώσει για τυχόν προβλήματα κατά την εισαγωγή καθώς και για τα μεγέθη που εισήχθησαν (αριθμός σημείων, γραμμών κτλ.).



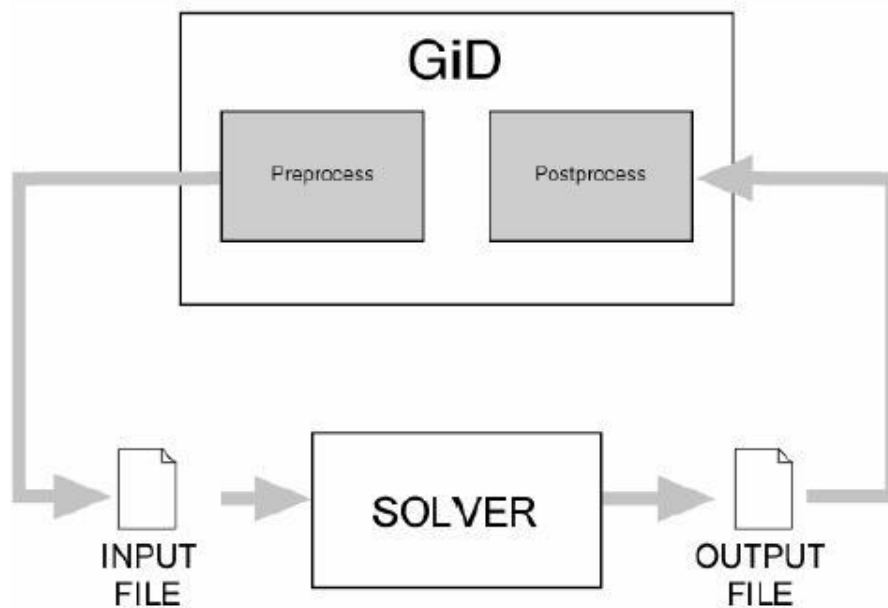
Σχήμα 2.7: Μοντέλο τρισδιάστατου χώρου όπως σχεδιάστηκε στο AutoCAD και εισήχθη στο GiD

Τώρα το GiD έχει εισάγει το μοντέλο και μπορούμε να παρέμβουμε σε αυτό όπως ακριβώς θα κάναμε και σε μία εφαρμογή που έγινε με το πρόγραμμα αυτό. Μπορούμε, επίσης μέσα από το GiD να καλέσουμε (ενεργοποιήσουμε) ένα πρόγραμμα αριθμητικής επίλυσης το οποίο να χρησιμοποιεί τα γεωμετρικά στοιχεία που θα του παρέχει το GiD. Να σημειωθεί πως ενδεχομένως μπορεί να υπάρξουν προβλήματα με την εισαγωγή αρχείων από άλλα προγράμματα, ανάλογα με το 'format' του αρχείου και την έκδοση του προγράμματος με το οποίο δημιουργήθηκε. Στο GiD θα βρούμε τις ανάλογες

επιλογές στο **Utilities > Preferences > Import** από όπου ο χρήστης μπορεί να ελέγξει τις παραμέτρους εισαγωγής, ώστε να έχει το καλύτερο δυνατό αποτέλεσμα και τις λιγότερες απώλειες.

2.2.5 Χρησιμοποιώντας το GiD για επίλυση προβλημάτων με εξωτερικό πρόγραμμα

Έχοντας κάποιο πρόγραμμα που μπορεί να κάνει αριθμητικούς υπολογισμούς μπορούμε να το προσαρμόσουμε εύκολα στο GiD. Αφού έχουμε σχεδιάσει μία γεωμετρία και έχουμε το δικτύωμα της, μπορούμε να ορίσουμε πάνω της συνθήκες ανάλογα με το είδος του προβλήματος. Το GiD έχει την δυνατότητα να χρησιμοποιήσει τις αυτές τις πληροφορίες (κορυφές, στοιχεία, συνδεσιμότητα δικτύωματος, συνοριακές συνθήκες, ορισμό διαφόρων υλικών της γεωμετρίας, κ.α.) για να τροφοδοτήσει το εξωτερικό πρόγραμμα με τα δεδομένα που χρειάζεται για την επίλυση και τελικά να επισκοπηθούν τα αποτελέσματα από τον μεταεπεξεργαστή. Στο παρακάτω διάγραμμα φαίνεται η διαδικασία.



Σχήμα 2.8: Η διαδικασία επίλυσης ενός προβλήματος

Η διαμόρφωση αυτή επιτυγχάνεται μέσω ειδικών τύπων αρχείων, όπως αναφέραμε και στην αρχή του κεφαλαίου. Τα ακόλουθα αρχεία υποστηρίζονται και μπορούν να διαβαστούν από το GiD. Δεν είναι αναγκαία η ανάπτυξη όλων παρά μόνον αυτών που χρειάζεται ο μηχανικός για τους υπολογισμούς του.

A) Αρχεία εισαγωγής δεδομένων στο πρόγραμμα:

.cnd : αρχείο διαμόρφωσης των συνθηκών του προβλήματος
.prb : αρχείο διαμόρφωσης των γενικών παραμέτρων
.mat : αρχείο διαμόρφωσης των υλικών της γεωμετρίας

B) Αρχείο συλλογής και εξαγωγής δεδομένων από το πρόγραμμα:

.bas : Το αρχείο αυτό μπορούμε να το ονομάσουμε αρχείο γενικής διαμόρφωσης και είναι το αρχείο που μεσολαβεί ανάμεσα στο GiD και στο εξωτερικό πρόγραμμα. Αναλαμβάνει να συγκεντρώσει όλα τα δεδομένα που αποτελούν το πρόβλημα και μετά να τα εγγράψει σε ένα αρχείο σε μορφή απλού κειμένου και με κατάληξη name.dat. Υπάρχει η δυνατότητα να φτιαχτούν πολλά .bas αρχεία εφ' όσον αυτό κρίνεται αναγκαίο.

Γ) Αρχείο ενεργοποίησης του εξωτερικού προγράμματος:

.bat : Το αρχείο αυτό είναι ένα απλό 'batch file' που εκτελείτε μέσω του 'command prompt' όταν ζητήσουμε την επίλυση από το GiD. Σκοπός του είναι να ενεργοποιήσει το εξωτερικό πρόγραμμα με το κατάλληλο αρχείο εισόδου. Μπορεί να γραφτεί είτε για Microsoft Windows είτε για Unix συστήματα ή και για τα δύο.

Το εξωτερικό πρόγραμμα (στην περίπτωση της εργασίας αυτής Helm2D.exe) τελικά θα διαβάσει το αρχείο .dat και θα γράψει τα αποτελέσματα σε ένα ξεχωριστό αρχείο εξόδου. Το εξωτερικό πρόγραμμα επίλυσης μπορεί να προγραμματιστεί σε γλώσσα της επιλογής μας.

Ο μετα-επεξεργαστής (postprocessor) του GiD μπορεί να διαβάσει αποτελέσματα που εξάγονται από το πρόγραμμα επίλυσης από αρχεία κειμένου με τα συγκεκριμένα format:

- A) .res : αρχείο αποτελεσμάτων. Το κάθε element θα αντιστοιχεί σε μία τιμή - λύση.
- B) .msh : το αρχείο της δικτύωσης. Αν λείπει αυτό το αρχείο τότε ο μεταεπεξεργαστής χρησιμοποιεί την υπάρχουσα δικτύωση από τον προεπεξεργαστή.

2.3 Το πρόγραμμα Helm2D

Το πρόγραμμα, το οποίο κάνει χρήση της ΜΣΣ για την επίλυση της ολοκληρωτικής εξίσωσης του Helmholtz στις δύο διαστάσεις είναι το Helm2D. Είναι σχεδιασμένο σε Fortran 77 και έχει την δυνατότητα να λύνει προβλήματα εσωτερικού και εξωτερικού τύπου με μια ποικιλία επιλογών και συνοριακών συνθηκών. Το πρόγραμμα μπορεί να διαβάσει την γεωμετρία, της συνοριακές συνθήκες και σημεία επίλυσης στο ακουστικό πεδίο από ένα αρχείο κειμένου. Στην έξοδο γράφει την ακουστική πίεση, σε μορφή πίνακα, για τις κορυφές της γεωμετρίας και για τα σημεία που ζήτησε ο χρήστης. Ο πηγαίος κώδικας του δίνεται μαζί με το πρόγραμμα και αποτελείται από εικοσιπέντε

ρουτίνες. Για να το χειριστεί κάποιος μέσω του GiD θα πρέπει να αλλάξει κάποιες από τις ρουτίνες με τις αντίστοιχες του Παραρτήματος Β καθώς και να προσθέσει μία ακόμα (Results.for) η οποία δίδεται επίσης στο Παράρτημα. Στην συνέχεια θα πρέπει να ξανακάνει 'compile' τον κώδικα, με οποιοδήποτε compiler της Fortran 77.

Μία σύντομη επισκόπηση στις ρουτίνες του προγράμματος όπως τελικά διαμορφώθηκαν:

HELM2D

Το αρχικό πρόγραμμα, καλεί τις ρουτίνες PREP και PROC

PREP

Αυτός είναι ο preprocessor που καλεί διαδοχικά τις ρουτίνες RGENE, RNODE, RELEM, BC, RBC, RFIELD, RCHIEF

RGENE

Αυτή η ρουτίνα διαβάζει τα γενικά στοιχεία του προβλήματος, [interior problem(INT=0), ή exterior(INT=1), radiation (ISC=0), scattering(ISC=1)], τις ιδιότητες του μέσου διάδοσης (δηλ. πυκνότητα, ταχύτητα ήχου) και την/ις συχνότητα/ες επίλυσης.

RNODE

Αυτή η ρουτίνα διαβάζει τις κορυφές (nodes) της γεωμετρίας.

RELEM

Αυτή η ρουτίνα διαβάζει τα στοιχεία καθώς, την συνδεσιμότητά τους (από ποιες κορυφές αποτελείτε κάθε στοιχείο) και τον τύπο των στοιχείων (linear, quadratic, ή cubic).

BC

Αυτή η ρουτίνα καλεί την RBC για να διαβάσει τις συνοριακές συνθήκες και μετά να ορίζει τον τύπο της κάθε συνθήκης. Η συνοριακή συνθήκη στην κάθε κορυφή του κάθε στοιχείου είναι της μορφής: (Βλέπε και ενότητα 1.7.1)

$$(CA) p + (CB) v_n = CC,$$

όπου CA, CB και CC είναι σταθερές που ορίζονται από τον χρήστη.

RBC

Βλέπε BC.

RFIELD

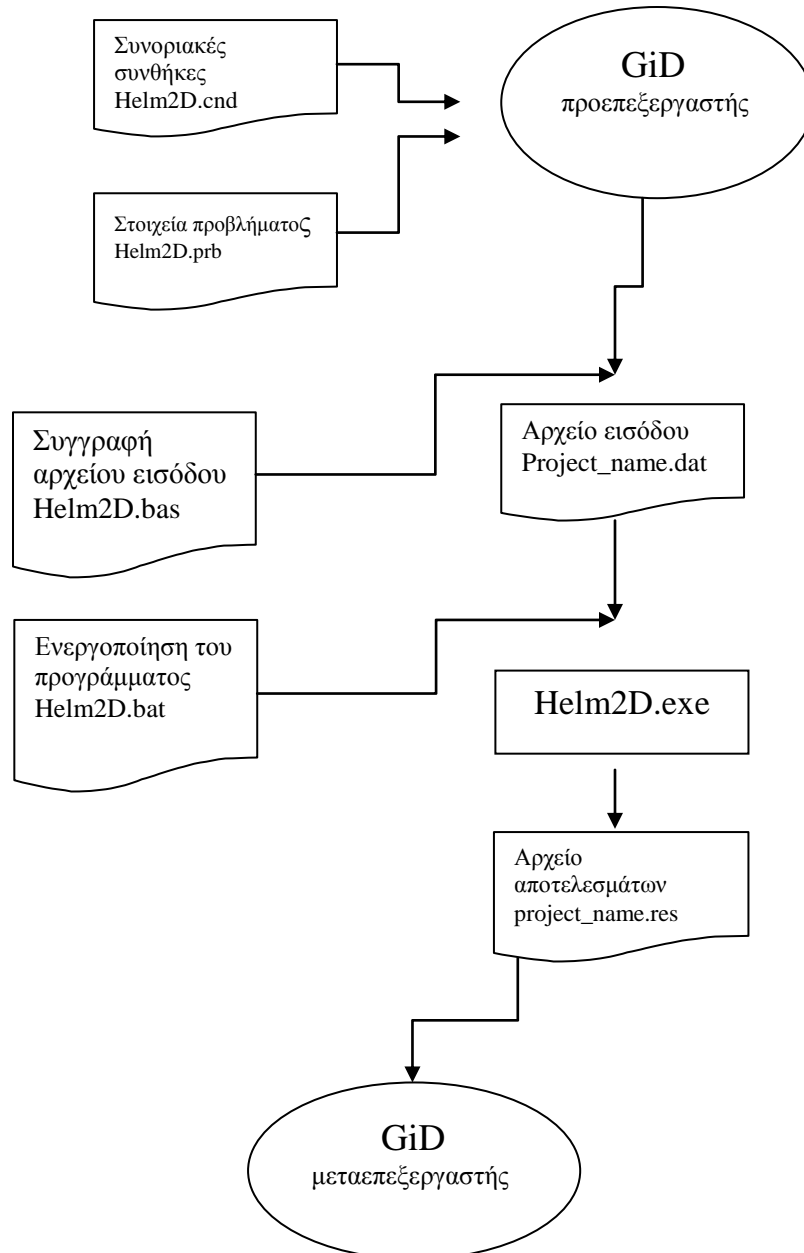
Αυτή η ρουτίνα διαβάζει τις συντεταγμένες των σημείων του ηχητικού πεδίου, εκεί δηλαδή όπου ο χρήστης ζητάει λύσεις του προβλήματος.

RGHIEF

Με αυτήν την ρουτίνα διαβάζονται τα Chief points (μόνο για προβλήματα εξωτερικού τύπου). (Βλέπε και ενότητα 3.3.1)

RESULTS

Με αυτή την ρουτίνα γράφεται σε ένα ξεχωριστό αρχείο οι λύσεις καθώς και η αντιστοίχησή τους με τα στοιχεία (elements) της γεωμετρίας, ώστε να διαβαστούν από τον μεταεπεξεργαστή. Η ρουτίνα αυτή καλείται από την PROC.for μετά από όλους τους υπολογισμούς.

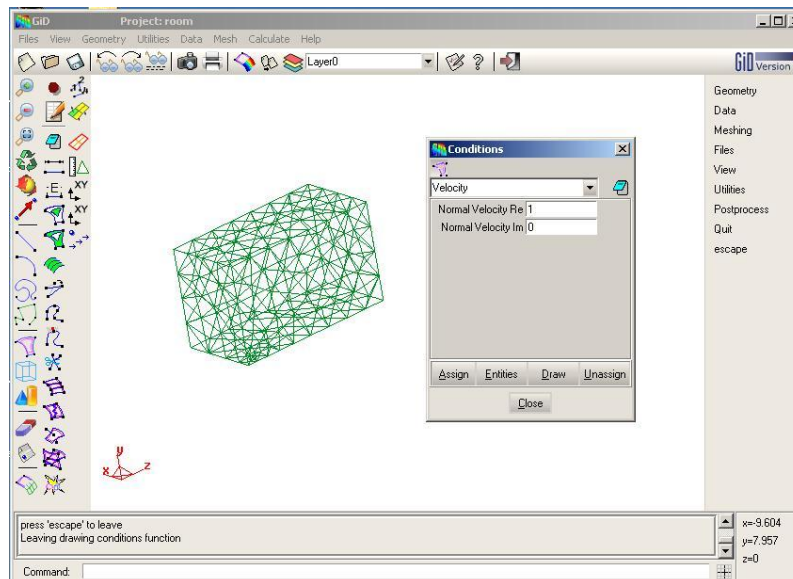


Σχήμα 2.9: Διάγραμμα ροής που δείχνει την διαδικασία και τα βήματα για την επίλυση ενός προβλήματος

Για να γίνει η διασύνδεση των δύο αυτών εφαρμογών πρέπει να γίνουν οι αλλαγές που αναφέρθηκαν και παραπάνω στο πηγαίο κώδικα του Helm2D, όσο αναφορά τον τρόπο που δέχεται τα δεδομένα για την επίλυση, όσο και στον τρόπο που εξάγει τις λύσεις. Επίσης πρέπει να αναπτυχθούν τα εξής αρχεία Helm2d.bas, Helm2d.cnd, Helm2d.prb, Helm2d.bat, που είναι τα αρχεία που θα αναλάβουν την μεταφορά των δεδομένων από το ένα πρόγραμμα στο άλλο όπως φαίνεται παραπάνω στο σχήμα 2.9. Αυτά τα αρχεία δίδονται στο Παράρτημα Γ και στο CD - Rom που συνοδεύει την εργασία αυτήν.

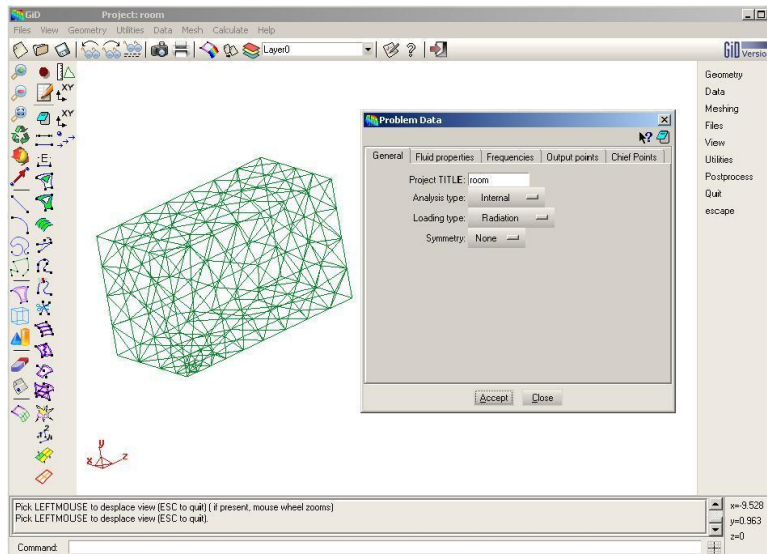
2.4 Τα αρχεία διαμόρφωσης της εφαρμογής GiD – Helm2D

Helm2d.cnd: Το αρχείο αυτό θα μας επιτρέπει να δηλώνουμε τις συνοριακές συνθήκες των προβλημάτων (ακουστική πίεση, σωματιδιακή ταχύτητα και ακουστική εμπέδηση) μέσα από τον προεπεξεργαστή του GiD.



Σχήμα 2.10: Το παράθυρο εισαγωγής συνοριακών συνθηκών

Helm2d.prb: Αντίστοιχα με αυτό το αρχείο δηλώνουμε από τον προεπεξεργαστή τα γενικά στοιχεία του προβλήματος όπως τον τύπο του προβλήματος, την ταχύτητα του ήχου, τα σημεία στον χώρο που θέλουμε λύσεις κ.α.



Σχήμα 2.11: Το παράθυρο εισαγωγής δεδομένων του προβλήματος (*Problem data*)

Helm2d.bas: Το αρχείο αυτό θα αναλάβει να γράψει σε ένα αρχείο κειμένου όλα τα δεδομένα με το κατάλληλο format, ώστε να μπορεί να διαβαστεί από το πρόγραμμα επίλυσης (Helm2D.exe) ως αρχείο εισόδου.

Helm2d.bat: Το αρχείο αυτό αναλαμβάνει την ενεργοποίηση του Helm2D.exe με το σωστό αρχείο εισόδου και να δώνει την σωστή ονομασία στο αρχείο εξόδου ώστε ο μετα-επεξεργαστής να σχεδιάσει σωστά τις λύσεις πάνω στα στοιχεία. Αυτή η διαδικασία φαίνεται παραπάνω στο διάγραμμα ροής του σχήματος 2.9.

Όπως είδαμε παραπάνω το Helm2D αποτελείται από εικοσιπέντε ρουτίνες, εκ των οποίων οι πρώτες έξι (Rgene.for, Rnode.for, Relem.for, RBC.for, Rfield.for, Rchief.for), ήταν αρχικά σχεδιασμένες να αποτελούν τον προεπεξεργαστή του προγράμματος. Τελικά εφ' όσον θέλουμε να χρησιμοποιήσουμε το GiD για την προεπεξεργασία, οι ρουτίνες αυτές ξανασχεδιάστηκαν για να αναλαμβάνουν μόνο την ανάγνωση των δεδομένων του προβλήματος και να τα σώζουν σε πίνακες ώστε να οδηγηθούν έπειτα προς επίλυση.

Έτσι σχεδιάστηκαν οι ρουτίνες εισόδου ώστε να δέχονται τα εξής δεδομένα:

- Γενικές πληροφορίες προβλήματος:
- Τύπος προβλήματος(εσωτερικό – εξωτερικό), είδος προβλήματος (εκπομπή – σκέδαση), πληροφορία συμμετρίας.
- Πυκνότητα μέσου διάδοσης, ταχύτητα του ήχου
- Συχνότητα/ες

- Συντεταγμένες κορυφών
- Είδος στοιχείων
- Συνδεσιμότητα στοιχείων
- Συνοριακές συνθήκες
- Σημεία επίλυσης στον ελεύθερο πεδίο

2.5 Οδηγίες για την εγκατάσταση της εφαρμογής

Αρχικά πρέπει να πούμε πως η εφαρμογή σχεδιάστηκε και μπορεί να ενεργοποιηθεί μόνο σε συστήματα με εγκατεστημένα τα Microsoft Windows (XP, Vista). Υπάρχει η δυνατότητα να λειτουργήσει και για Unix συστήματα μόνο αν προγραμματιστεί κατάλληλα το αρχείο Helm2D.bat για να ενεργοποιείτε το πρόγραμμα μέσω του ‘terminal’.

Οι ρουτίνες που αναλάμβαναν την προεπεξεργασία του προβλήματος πρέπει αντικατασταθούν με τις καινούργιες για να λειτουργήσει σωστά η εφαρμογή μέσα στο GiD. Στο Παράρτημα Β είναι όλες οι ρουτίνες που χρειάζονται για να γίνει ξανά ‘compile’ το πρόγραμμα. Εκτός από αυτή την αλλαγή στις ρουτίνες εισόδου πρέπει κατά την διαδικασία του ‘compile’ να προστεθεί και η νέα ρουτίνα εξόδου. Αυτή θα αναλαμβάνει, αφού το πρόβλημα έχει λυθεί, να γράψει στο αρχείο εξόδου (project_name.res) όλες τις λύσεις σε αντιστοίχιση με την πληροφορία της δικτύωσης, δηλαδή κάθε ένα στοιχείο με την αντίστοιχη λύση. Το αρχείο αυτό θα διαβάζεται από τον μετα-επεξεργαστή του GiD και θα μπορούμε να δούμε γραφικά πλέον τα αποτελέσματα, ζωγραφισμένα πάνω στην γεωμετρία. Στο Παράρτημα Β βρίσκεται ο κώδικας του (RESULTS.for).

Αφού το πρόγραμμα γίνει ‘compile’, το Helm2D είναι έτοιμο για χρήση. Για να λειτουργεί η εφαρμογή αυτή πρέπει το νέο εκτελέσιμο αρχείο (Helm2D.exe) μαζί με όλα τα αρχεία διαμόρφωσης, που περιγράφηκαν παραπάνω, να βρίσκονται σε ένα κοινό φάκελο που να ονομάζεται ‘Helm2D.gid’. Αυτός ο φάκελος πρέπει τελικά να βρίσκεται στο φάκελο εγκατάστασης του GiD και συγκεκριμένα μέσα στην τοποθεσία ‘Problem types’.

Μετά από τα παραπάνω βήματα ο χρήστης θα έχει την δυνατότητα να σχεδιάζει γεωμετρικά μοντέλα και έπειτα να καλεί την εφαρμογή Helm2D μέσα από τον προεπεξεργαστή του GiD. Θα μπορεί έπειτα να δηλώνει όλα τα στοιχεία ενός προβλήματος όπως οι συνοριακές συνθήκες (πίεση, ταχύτητα, ακουστ. εμπέδηση) και τα φυσικά μεγέθη που θα διέπουν το πρόβλημα (ταχύτητα ήχου, πυκνότητα, κ.τ.λ.). Στην συνέχεια να δικτυώνει την γεωμετρία και τέλος να επιλύει το πρόβλημα. Η διαδικασία αυτή θα φανεί αναλυτικότερα στις εφαρμογές που παρουσιάζονται στην επόμενη ενότητα.

2.4 Το πρόγραμμα Helm3D

Το πρόγραμμα αυτό παρέχεται επίσης με το ίδιο βιβλίο, [1] που αναφέρεται και στην εισαγωγή και μπορεί να λύσει τους ίδιους τύπους προβλημάτων με το Helm2D αλλά στις τρεις διαστάσεις. Μία αντίστοιχη εφαρμογή, για την αυτόματη ενεργοποίηση του μέσω του GiD, αναπτύχθηκε και παρουσιάστηκε σε συνέδριο ακουστικής, [7]. Με την εφαρμογή αυτή παρέχεται, το εκτελέσιμο πρόγραμμα (Helm3D.exe) και το πακέτο αρχείων διαμόρφωσης (Helm3D.gid) που είναι αναγκαίο για να τρέξει η εφαρμογή μέσω του GiD. Για την καλύτερη κατανόηση των προβλημάτων στις τρεις διαστάσεις έγιναν και εδώ αλλαγές στο κώδικα καθώς και σε κάποια από τα αρχεία διαμόρφωσης (Helm3D.prb, Helm3D.bas, Helm3D.bat). Οι αλλαγές αυτές θα μας βοηθήσουν στην καλύτερη επισκόπηση των αποτελεσμάτων για προβλήματα εσωτερικού τύπου. Θα μας δώσουν την δυνατότητα να σχεδιάζουμε επιπλέον επιφάνειες στο εσωτερικό της γεωμετρίας στις οποίες τελικά θα αναζητούμε λύσεις ελεύθερου πεδίου (field point). Έτσι ο χρήστης του προγράμματος μπορεί να έχει μεγαλύτερη ανάλυση στα αποτελέσματα και να επισκοπεί γραφικά την ακουστική πίεση και εντός των συνόρων. Οι αλλαγές στις ρουτίνες και στα νέα αρχεία διαμόρφωσης βρίσκονται στο Παράρτημα Δ και Ε καθώς και στο CD - Rom. Παραδείγματα με αυτή την εφαρμογή βρίσκονται στην ενότητα 3.3 όπου λύνονται προβλήματα στις τρεις διαστάσεις.

Για να τρέξει η νέα αυτή εφαρμογή θα πρέπει, παρόμοια με το Helm2D, να ξαναγίνει 'compile' το πρόγραμμα, με τις νέες ρουτίνες. Τελικά το εκτελέσιμο πρόγραμμα (Helm3D.exe) μαζί με τα νέα αρχεία διαμόρφωσης πρέπει να τοποθετηθούν σε φάκελο με το όνομα 'Helm3D.gid' και να τοποθετηθεί μέσα στον φάκελο 'Problem types', που βρίσκεται στην τοποθεσία εγκατάστασης του GiD.

Κεφάλαιο 3⁰

Εφαρμογές

3.1 Εισαγωγή

Όπως είδαμε και στο Κεφ. 1, η Μέθοδος των Συνοριακών Στοιχείων (ΜΣΣ ή ΒΕΜ) έχει χρησιμοποιηθεί για την επίλυση προβλημάτων και στην ακουστική, όπως π.χ. τον υπολογισμό της ηχητικής πίεσης που παράγεται από μηχανές ή τον υπολογισμό της ακουστικής πίεσης στο ελεύθερο πεδίο για κύματα που προσπίπτουν πάνω σε επιφάνειες και ανακλώνται.

Η εφαρμογή που σχεδιάστηκε εδώ είχε βασικά εκπαιδευτική αποστολή και περιέχει δύο βασικά μέρη: Α) τη γνωριμία και την επίδειξη στη πράξη ενός γραφικού περιβάλλοντος (GUI) που χρησιμοποιείται για την σχεδίαση του χώρου επίλυσης, την διακριτοποίηση των συνοριακών επιφανειών και την επίδοση όλων των απαραίτητων δεδομένων στο βασικό πρόγραμμα αριθμητικής επίλυσης, για το συγκεκριμένο πρόβλημα, καθώς και την μετέπειτα κατάλληλη παρουσίαση των αποτελεσμάτων Β) την συγγραφή ενός κειμένου οδηγού, που με απλά βήματα μπορεί να εκπαιδεύσει στην χρήση της μεθόδου ΒΕΜ ώστε μέσω των αποτελεσμάτων να κατανοηθούν καλύτερα μερικά βασικά προβλήματα ακουστικής. Στο συγκεκριμένο κεφάλαιο υλοποιείται ο δεύτερος αυτός στόχος, παρουσιάζοντας μια σειρά από ακουστικά προβλήματα, καθώς και τον τρόπο για να υλοποιηθούν μέσα από το συνδυασμό των εφαρμογών GiD και Helm2D ή GiD και Helm3D.

Το κεφάλαιο αυτό καθοδηγεί το χρήστη στην εφαρμογή της μεθόδου ΒΕΜ με έντεκα προβλήματα, όπου το κάθε ένα παρουσιάζει διαφορές ως προς α) τις εφαρμογές της ακουστικής β) τις χρήσεις της μεθόδου ΒΕΜ γ) της χρήσης των προγραμμάτων GiD, Helm2D και Helm3D.

Το κεφάλαιο αυτό έχει σχεδιαστεί με οδηγίες βήμα προς βήμα, που θα εξηγούν, πως να εισάγεις ένα γεωμετρικό μοντέλο, τον ορισμό συνοριακών συνθηκών και την ανάλυση των αποτελεσμάτων. Προτείνεται, για την καλύτερη κατανόηση, η ανάγνωση των εφαρμογών να γίνει με την σειρά που έχουν συνταχθεί, γιατί ορισμένα θέματα θα θεωρούνται γνωστά όσο προχωρούν οι εφαρμογές, άρα και δεν θα αναφέρονται ξανά και ξανά. Όσα γράφονται με έντονους χαρακτήρες δείχνουν εντολές και κινήσεις που πρέπει να γίνουν μέσω της εφαρμογής. Τέλος οι αριθμοί που περιβάλλονται από '<>' δείχνουν πως ο εν λόγω αριθμός πρέπει να εισαχθεί σε κάποιο παράθυρο της εφαρμογής, ή σε κάποια επιλογή του προγράμματος GiD.

Μετά την ανάγνωση του κεφαλαίου αυτού, ο χρήστης θα πρέπει να έχει κατανοήσει τις εξής αρχές της ακουστικής:

- Μονοδιάστατα στάσιμα κύματα
- Μονοδιάστατα οδεύοντα κύματα
- Ακουστική εμπέδηση (απορρόφηση)
- Στάσιμα κύματα σε ορθογώνιους χώρους
- Στάσιμα κύματα σε περίπλοκους χώρους
- Σφαιρικά κύματα

- Σκέδαση επιπέδων κυμάτων

Επίσης αρκετές από τις έννοιες της μεθόδου BEM, όπως:

- Εσωτερικά – εξωτερικά προβλήματα
- Τύπος συνοριακών στοιχείων (elements)
- Μέγεθος στοιχείων (6 elements ανά μήκος κύματος)
- Συμμετρία και συμμετρικά προβλήματα
- Σφάλματα της direct BEM και πως αντιμετωπίζονται (Chief points)

Στο τέλος ο χρήστης θα έχει λάβει γνώση για τα παρακάτω:

- Κατασκευή γεωμετρίας μέσα στο περιβάλλον GiD
- Εισαγωγή γεωμετρίας από γνωστά format (IGES)
- Δικτύωση της γεωμετρίας
- Εφαρμογή συνοριακών συνθηκών
- Επίλυση μέσω Helm2D και Helm3D
- Επισκόπηση αποτελεσμάτων μέσω του GiD

3.2 Προβλήματα σε δύο διαστάσεις

Τα παραδείγματα που ακολουθούν αναφέρονται σε χώρο δύο διαστάσεων. Ο χρήστης, αφού έχει κάνει τα βήματα που περιγράφονται στο κεφ. 2 για να λειτουργεί η εφαρμογή μέσω του GiD, θα πρέπει να αντιγράψει την εφαρμογή (το εκτελέσιμο πρόγραμμα) Helm2D, μαζί με τα αρχεία διαμόρφωσης, μέσα στο φάκελο problem types, που βρίσκεται στο φάκελο εγκατάστασης του GiD. Επίσης, για να λειτουργήσει σωστά η εφαρμογή, θα πρέπει όλα τα γεωμετρικά μοντέλα να σχεδιάζονται πάνω στους άξονες X – Y. Ας θυμηθούμε επίσης πως με την ΜΣΣ μπορούμε να αναζητήσουμε λύσεις για την ακουστική πίεση έχοντας μόνο διακριτοποιήσει την συνοριακή επιφάνεια και γνωρίζοντας τις συνοριακές συνθήκες, για μία από τις μεταβλητές (πίεση ή σωματιδιακή ταχύτητα) πάνω σε αυτή. Οι λύσεις (για τη περίπτωση εσωτερικών προβλημάτων, όπως στην ενότητα αυτή ζητούνται και μέσα από τα σύνορα. Έτσι αυτή η εφαρμογή έχει σχεδιαστεί με τέτοιο τρόπο ώστε να τροφοδοτεί με δύο δικτύωματα το πρόγραμμα. Το πρώτο δικτύωμα (συνοριακή γεωμετρία) είναι απαραίτητο για την επίλυση ενώ το δεύτερο δικτύωμα (στο εσωτερικό χωρίο) μας βοηθά μόνο στο να ζητήσουμε τις λύσεις πάνω στα σημεία αυτά, τα οποία άμα είναι αρκετά κοντά μπορούν να μας δώσουν μία πλήρη εικόνα του ηχητικού πεδίου στο εσωτερικό της γεωμετρίας. Αυτή η διαδικασία αναπτύσσεται αναλυτικά στο πρώτο παράδειγμα παρακάτω.

3.2.1: Στάσιμο κύμα σε σωλήνα με κλειστό άκρο

Ανοίγουμε το GiD και δημιουργούμε ένα καινούργιο project που το ονομάζουμε tube2d. Πρώτα θα σχεδιαστεί η γεωμετρία (σωλήνας). Ανοίγουμε το παράθυρο **Utilities > tools > coordinates** window το οποίο και αφήνουμε ανοιχτό, επιλέγουμε μετά **Geometry > Create > Straight line**. Εισάγουμε διαδοχικά τις παρακάτω συντεταγμένες: $\langle 0,0 \rangle$, $\langle 10,0 \rangle$, $\langle 10,1 \rangle$, $\langle 0,1 \rangle$, $\langle 0,0 \rangle$. Οι αριθμοί αυτοί αντιστοιχούν σε μέτρα (m).



Σχήμα 3.2.1: Γεωμετρική αναπαράσταση σωλήνα στις δύο διαστάσεις

Έπειτα πρέπει να φτιαχτεί η επιφάνεια η οποία αντιπροσωπεύει τις λύσεις στο εσωτερικό του σωλήνα. Για να λυθεί σωστά το πρόβλημα πρέπει η επιφάνεια να ανήκει σε ένα ξεχωριστό δικό της layer. Ανοίγουμε την επιλογή **Utilities > Layers** και δημιουργούμε ένα καινούργιο layer με το όνομα $\langle \text{field} \rangle$. Αν δεν έχει επιλεγεί από μόνο του, το επιλέγουμε με το ποντίκι. Αφού σιγουρευτούμε πως δουλεύουμε στο layer 'field' τότε για τη δημιουργία της επιφάνειας επιλέγουμε **Geometry > Create > Nurbs surface > By contour**. Με το ποντίκι επιλέγουμε και τις τέσσερις γραμμές του σωλήνα και πατάμε **Esc**. Αφού ορίσαμε τη γεωμετρία μας μένει να δηλώσουμε και τα συνοριακά στοιχεία του προβλήματός μας. Από την επιλογή **Data > Problem type** διαλέγουμε το **Helm2D**. Αν έχει φορτωθεί σωστά θα ενημερωθούμε αντίστοιχα από το GiD.

Δήλωση των συνοριακών συνθηκών

Κάνουμε click στην επιλογή **Data > Conditions**. Από το παράθυρο που θα ανοίξει επιλέγουμε το **Velocity** για να δηλώσουμε την σωματιδιακή ταχύτητα στα σύνορα του σωλήνα. Πρέπει πρώτα να δηλώσουμε μηδενική ταχύτητα σε όλες τις πλευρές του προβλήματος. Κατ' αρχήν θέτουμε την τιμή $\langle 0 \rangle$ στα **Normal Velocity real** και **imaginary** και επιλέγουμε **Assign**. Με drag & drop διαλέξτε όλες τις γραμμές και πατήστε **Esc**, μετά αλλάζουμε την τιμή του **Normal Velocity real** σε $\langle 1 \rangle$ και κάνουμε πάλι **Assign**, αυτή την φορά μόνο στην αριστερή κάθετη πλευρά του σωλήνα.

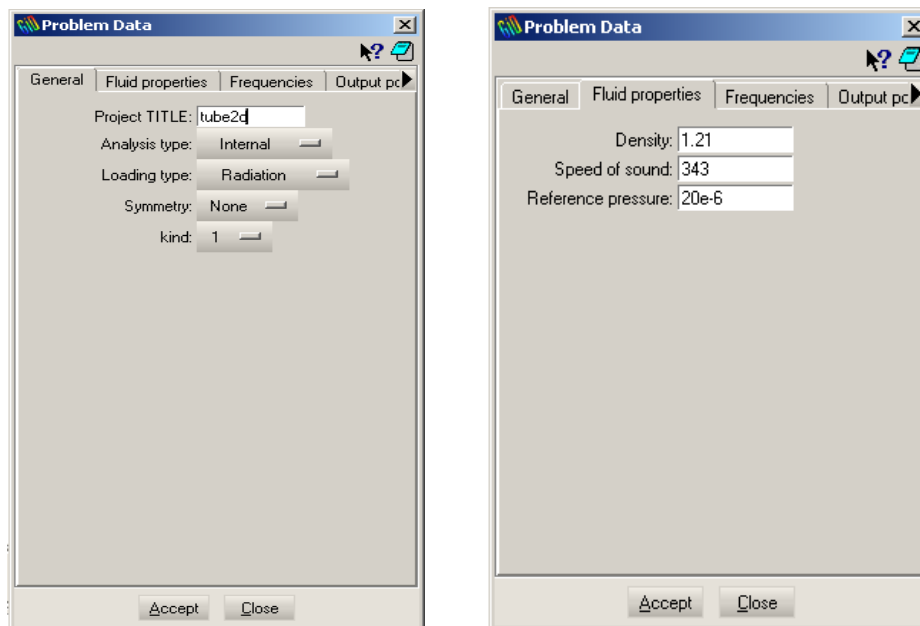
Δήλωση στοιχείων του προβλήματος

Ακολούθως θα δώσουμε τα υπόλοιπα δεδομένα για το πρόβλημα από το παράθυρο **Data > Problem data**. Στην επιλογή: *general*, δηλώνουμε ένα όνομα για το

πρόβλημα μας, π.χ. 'tube2d'. Ακολούθως στα πεδία (που φαίνονται και στο σχήμα 3.2.2), επιλέγουμε τα παρακάτω:

Analysis type: Internal
Loading type: Radiation
Symmetry: none
kind: 1

Το πρώτο πεδίο ('Analysis type') αναφέρεται σε ποιο χωρίο αναζητούμε λύσεις στο εξωτερικό ή εσωτερικό. Στο πρόβλημα μας εδώ, ζητάμε λύσεις εντός των συνόρων. Η επιλογή 'Loading type' μας επιτρέπει να διαλέξουμε το τύπο του προβλήματος, αν είναι δηλαδή εκπομπής (Radiation) ή σκέδασης (Scattering). Στο εν λόγω πρόβλημα έχουμε εκπομπή κύματος. Για τις επιλογές συμμετρίας θα μιλήσουμε παρακάτω στην παράγραφο 3.3.1 όπου θα παρουσιαστεί αντίστοιχη εφαρμογή. Η επιλογή kind αναφέρεται στο είδος των 'στοιχείων'. Επιλέγουμε: 1 για γραμμικά, 2 για τετραγωνικά και 3 για κυβικά στοιχεία. Σε αυτό και γενικά σε όλα τα προβλήματα που παρουσιάζονται στις δύο διαστάσεις έχουμε επιλέξει γραμμικά στοιχεία.



Σχήμα 3.2.2: Το παράθυρο εισαγωγής δεδομένων προβλήματος της εφαρμογής

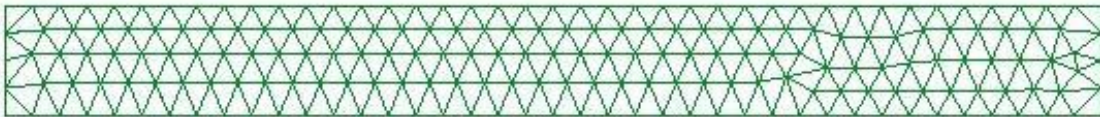
Στην επιλογή 'fluid properties' γράφουμε τα παρακάτω:

Density: 1.21 (πυκνότητα του αέρα, Kg/m^3),
Speed of sound: 343 (ταχύτητα του ήχου στον αέρα, m/s),
Reference pressure: 20e-6 (στάθμη αναφοράς, Pa).

Στην επιλογή '*frequencies*' δίνουμε τη τιμή <51.45> και στο **frequency start** και στο **frequency stop**. Η τιμή αυτή εκφράζει τη μία συχνότητα (Hz) για την οποία ζητάμε λύσεις στο πρόβλημα. Η παραπάνω τιμή επιλέχθηκε έτσι ώστε να δούμε ένα μέγιστο στο εσωτερικό του σωλήνα (βλ. θεωρία του προβλήματος παρακάτω). Από την επιλογή '*output points*' διαλέγουμε: **Field point type: mesh**. Πατάμε: **Accept** για να σωθούν τα δεδομένα.

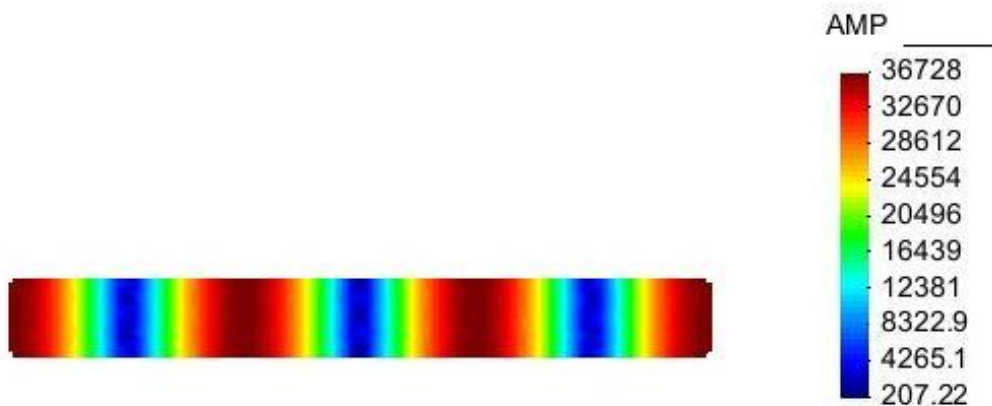
Δικτύωμα του σωλήνα

Η επόμενη ενέργεια είναι λογικά, να δώσουμε εντολή στο GiD να φτιάξει το δικτύωμα του σωλήνα. Πρέπει όμως να λάβουμε υπόψη πως λίγο παραπάνω στο Field point type επιλέξαμε <mesh>. Αυτό σημαίνει πως το πρόγραμμα επίλυσης περιμένει ένα επιπλέον δικτύωμα στα σημεία του οποίου θα βρεί λύσεις για το ακουστικό πεδίο (στο εσωτερικό του χωρίου). Όμως η επίλυση της ολοκληρωτικής εξίσωσης του Helmholtz με την ΜΣΣ απαιτεί το δικτύωμα των συνόρων του χωρίου. Για να λυθεί σωστά το πρόβλημα το πρόγραμμα επίλυσης απαιτεί σίγουρα και τη δικτύωση του συνόρου. Το GiD όμως αυτόματα δικτυώνει μόνο την ανώτερης τάξης επιφάνεια, οπότε εδώ θα δικτύωνε μόνο το εσωτερικό του σωλήνα. Για να το αναγκάσουμε να δημιουργήσει ξεχωριστό δικτύωμα για το σύνορο του προβλήματος (εδώ δηλαδή τις γραμμές που ορίζουν το ορθογώνιο), θα πρέπει να επιλέξουμε το **Mesh > Mesh criteria > Mesh > Lines**. Μετά πατάμε στην επιλογή **Mesh > Generate mesh**. Από το παράθυρο που θα ανοίξει εισάγουμε την τιμή <0.4>, που ορίζει το μέγεθος του κάθε στοιχείου και μετά πατάμε **OK**.



Σχήμα 3.2.3: Δικτύωμα του σωλήνα

Πλέον όλα είναι έτοιμα για την επίλυση η οποία θα ξεκινήσει όταν επιλέξουμε **Calculate > calculate**. Όταν η διαδικασία τελειώσει το GiD θα μας ενημερώσει και μπορούμε πλέον να δούμε τα αποτελέσματα στον μετα-επεξεργαστή από την επιλογή **View Results > Contour fill > AMP**.



Σχήμα 3.2.4: Γραφική αναπαράσταση στάσιμου κύματος στον σωλήνα

Θεωρία για το συγκεκριμένο πρόβλημα

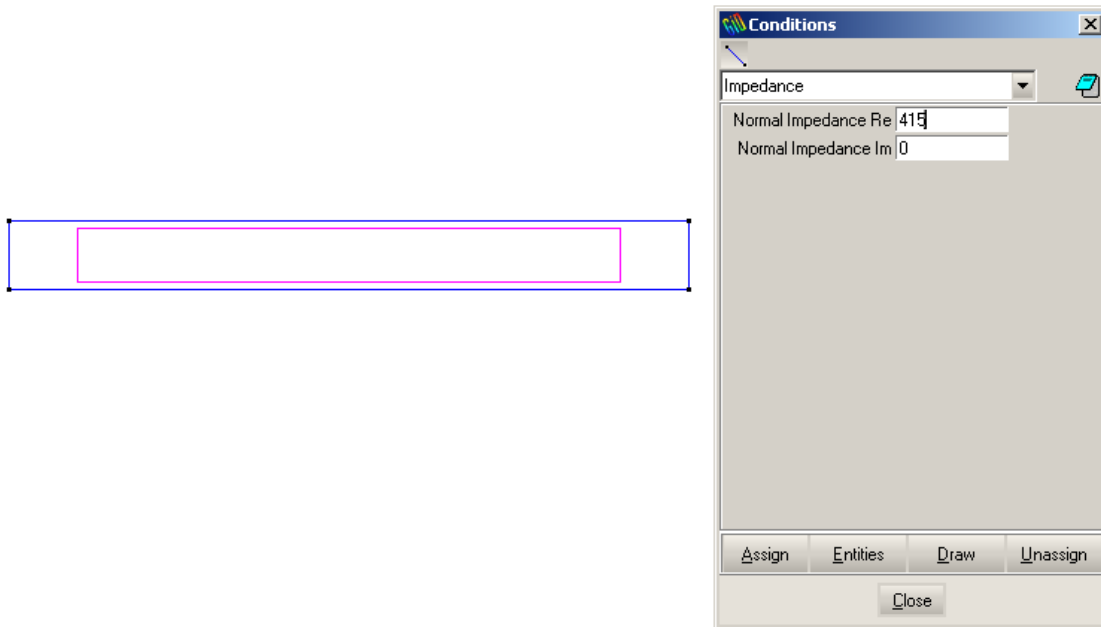
Όπως είναι γνωστό, οι συχνότητες συντονισμού ενός κλειστού σωλήνα μπορούν να βρεθούν θεωρητικά από την σχέση:

$$f = \frac{nc}{2l} \quad (3.2.1)$$

Όπου n είναι αριθμός του στάσιμου κύματος, c η ταχύτητα διάδοσης και l το μήκος του σωλήνα και f η συχνότητα συντονισμού. Στο συγκεκριμένο παράδειγμα όπου το μήκος $l = 10$ μέτρα και η συχνότητα $f = 51,45$ Hz το $n = 3$. Δηλαδή έχουμε $3\lambda/2$ μήκη κύματος μέσα στο μήκος του σωλήνα.

3.2.2: Οδεύον κύμα σε σωλήνα

Σε αυτό το παράδειγμα θα δούμε πως ορίζετε η ακουστική εμπέδηση σε κάποια περιοχή της γεωμετρίας που έχουμε σχεδιάσει. Επιστρέφουμε στον προεπεξεργαστή (preprocessor) του GiD **Files > Preprocess**. Επιλέγουμε να βλέπουμε την γεωμετρία μας με το **View > Mode > Geometry**. Τώρα μπορούμε να ορίσουμε την σύνθετη αντίσταση σε όλη την δεξιά πλευρά του σωλήνα με το **Data > Conditions**. Επιλέγουμε το **Impedance** και δίνουμε την τιμή **<0>** για το **Normal Impedance real** και **imaginary**, πατάμε **Assign** και επιλέγουμε όλο το σωλήνα. Μετά δίνουμε την τιμή **<415>** για το **real** και κάνουμε **Assign** μόνο στην δεξιά πλευρά του σωλήνα.



Σχήμα 3.2.5: Παράθυρο εισαγωγής συνοριακών συνθηκών

Τώρα πρέπει να ξανακάνουμε την δικτύωση, από το **Mesh > Generate mesh**, γιατί αλλάξανε οι συνοριακές συνθήκες. Αφού γίνει αυτό τότε μπορούμε να κάνουμε **Calculate > calculate** και να δούμε το αποτέλεσμα με τον μεταεπεξεργαστή.

Θεωρία του προβλήματος

Για την περίπτωση που έχουμε αντίσταση στο άκρο του σωλήνα, την οποία θέσαμε ίση με την αντίσταση του μέσου διάδοσης (αέρας), στο δεξί άκρο έχουμε μη ανάκλαση του ήχου. Άρα ο σωλήνας θεωρείται ότι έχει άπειρο μήκος. Για να βρούμε την αναλυτική λύση θα ξεκινήσουμε με το μονοδιάστατο κύμα:

$$p = A e^{-i k x} \quad (3.2.2)$$

Όπου A είναι το πλάτος της πίεσης στο σημείο που ορίζουμε την ταχύτητα (άκρο $x=0$).

$$\frac{\partial p}{\partial x} = -i \rho \omega v_x \quad (3.2.3)$$

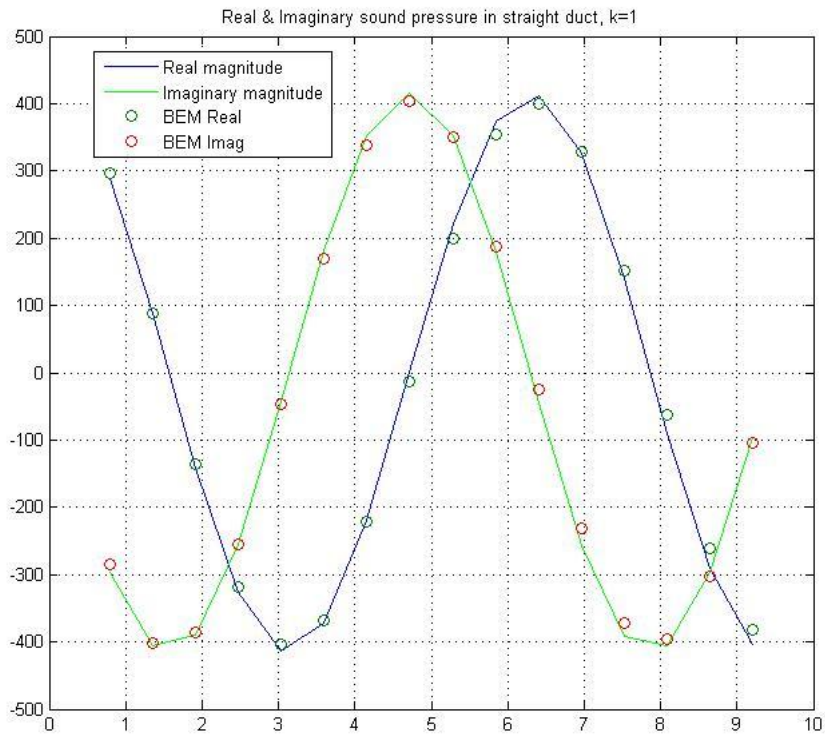
Διαφορίζοντας την εξίσωση χρησιμοποιώντας την συνθήκη στο ($x=0$).

$$-ikA = -i\rho\omega \quad (3.2.4)$$

Άρα, $A = \rho c$ και η αναλυτική λύση της ακουστικής πίεσης είναι:

$$p(x) = \rho c e^{-i k x} \quad (3.2.5)$$

Παρακάτω βλέπουμε το πραγματικό και φανταστικό μέρος της λύσης όπως υπολογίστηκε στο MATLAB και τη σύγκριση με την αριθμητική επίλυση. Παρατηρούμε πως υπάρχει ταύτιση των αποτελεσμάτων.

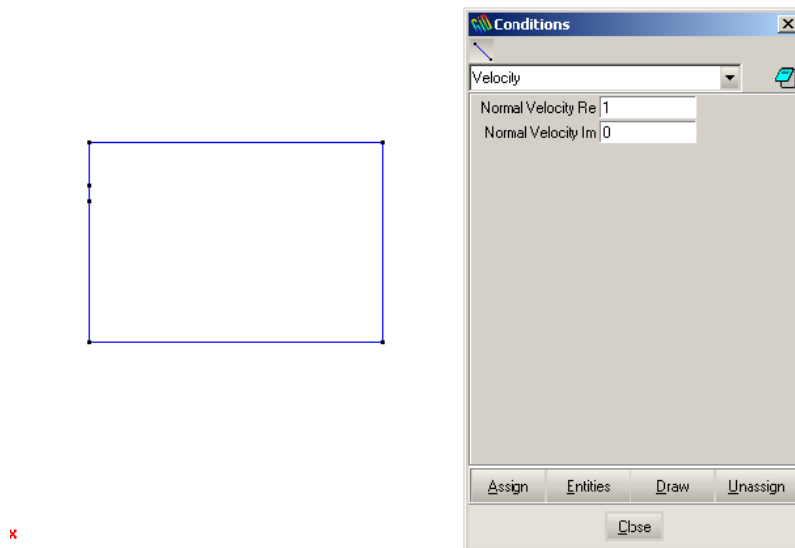


Γράφημα 3.2.6: Πραγματικό και φανταστικό πλάτος πίεσης σε σωλήνα απείρου μήκους

3.2.3: Ιδιορυθμοί σε ορθογώνιο δωμάτιο (δύο διαστάσεων)

Κατασκευή γεωμετρίας

Δημιουργούμε στο GiD ένα καινούργιο 'project' και το ονομάζουμε 'room2d'. Ανοίγουμε το παράθυρο **Utilities > Tools > Coordinates window**. Επιλέγουμε έπειτα **Geometry > Create > Straight line**. Εισάγουμε το περίγραμμα του δωματίου με τις παρακάτω συντεταγμένες: $\langle 0-0 \rangle$, $\langle 5.7-0 \rangle$, $\langle 5.7-3.8 \rangle$, $\langle 0-3.8 \rangle$, $\langle 0-0 \rangle$. Για να σχεδιάσουμε την πηγή του ήχου πάνω σε ένα τοίχωμα επιλέγουμε: **Geometry > Edit > Divide > Lines > Near point**. Το ποντίκι πρέπει να έχει αλλάξει σε μικρό σταυρό. Με τον σταυρό κάνουμε 'κλικ' πάνω στην αριστερή κάθετη γραμμή στο σημείο (0,2.7) και μετά επιλέγουμε την ίδια γραμμή και πατάμε **Esc**. Αν όλα τα βήματα έγιναν σωστά η αριστερή κάθετη γραμμή θα πρέπει να είναι πλέον χωρισμένη στα δύο. Επαναλαμβάνουμε τη διαδικασία για το σημείο (0,3) ώστε το αποτέλεσμα να είναι όπως αυτό του σχήματος 3.2.7.



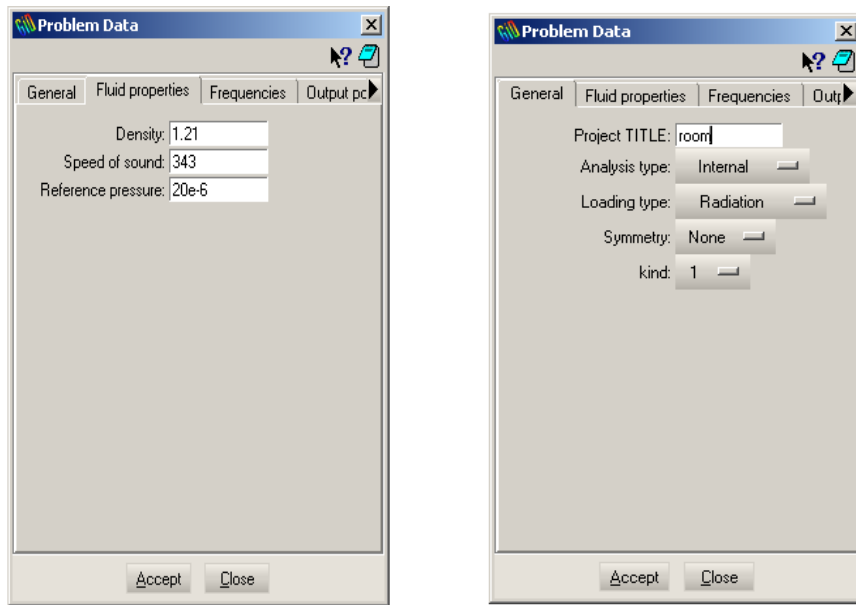
Σχήμα 3.2.7: Παράθυρο ορισμού των συνοριακών συνθηκών

Έπειτα πρέπει να φτιαχτεί η επιφάνεια πάνω στην οποία θα ζητήσουμε τις λύσεις (στο εσωτερικό του δωματίου). Όπως και στο προηγούμενο πρόβλημα πρέπει η επιφάνεια να ανήκει σε ένα ξεχωριστό δικό της επίπεδο εργασίας (layer). Ανοίγουμε την επιλογή **Utilities > Layers** και δημιουργούμε ένα καινούργιο επίπεδο με το όνομα 'field'. Αφού σιγουρευτούμε πως δουλεύουμε στο layer field τότε για την δημιουργία της επιφάνειας επιλέγουμε **Geometry > Create > Nurbs surface > By contour**. Με το ποντίκι διαλέγουμε όλες τις γραμμές του δωματίου και πατάμε **Esc**. Αφού ορίσαμε την γεωμετρία, μας μένει να δηλώσουμε και τις συνοριακές συνθήκες του προβλήματός. Από την επιλογή **Data > Problem type** διαλέγουμε το **Helm2D**. Αν έχει φορτωθεί σωστά θα ενημερωθούμε αντίστοιχα από το GiD.

Συνοριακές συνθήκες

Επιλέγουμε **Data > Conditions** και μετά **Velocity**. Αλλάζουμε την τιμή του **Normal Velocity Real** σε **<0>** και κάνουμε **Assign**, επιλέγοντας όλες τις πλευρές του δωματίου. Μετά αλλάζουμε την τιμή του **Normal Velocity Real** σε **<1>** και πατάμε πάλι **Assign** μόνο στην μικρή γραμμή πάνω αριστερά, που αναπαριστά τη πηγή του δωματίου.

Δήλωση των δεδομένων του προβλήματος

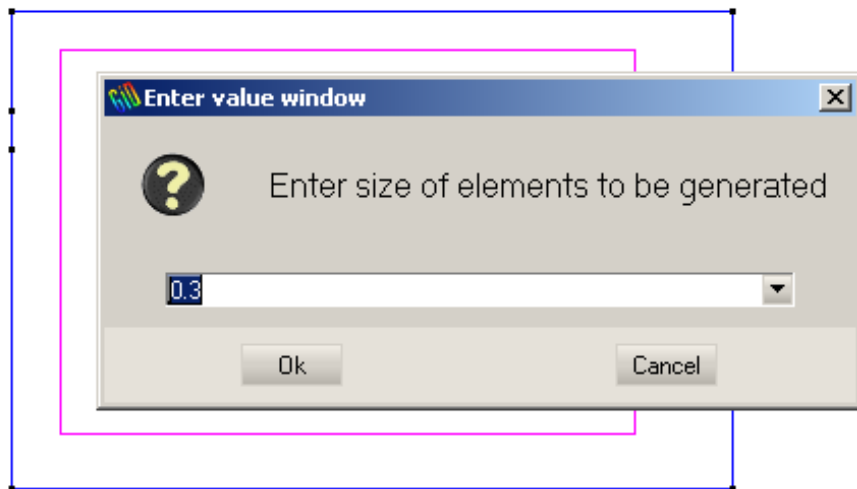


Σχήμα 3.2.8: Δεδομένα προβλήματος.

Επιλέγουμε **Data > Problem Data** και συμπληρώνουμε με τα δεδομένα του σχήματος 3.2.8 στις επιλογές **General** και **Fluid properties** αντίστοιχα. Για τις συχνότητες δηλώνουμε στο **Frequency start** και **Frequency stop** τη συχνότητα **<54.24>** που αντιπροσωπεύει τον ιδιορυθμό (1,1) του χώρου αυτού. Επιλέγουμε **Accept** για να σωθούν τα δεδομένα.

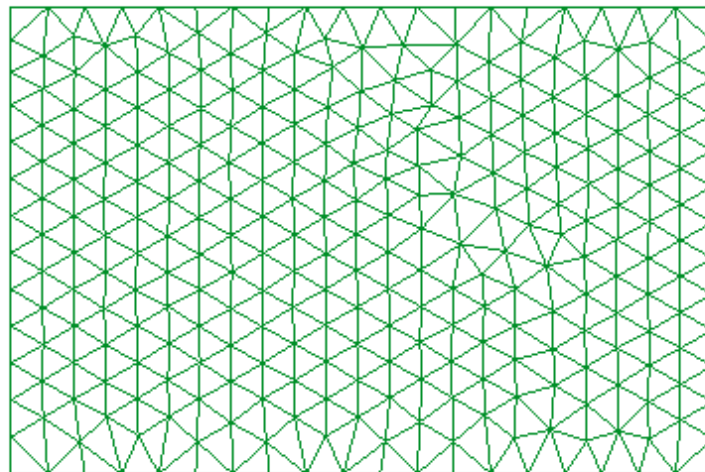
Δικτύωμα του δωματίου

Επιλέγουμε από το menu του GiD το **Mesh > Mesh criteria > Mesh > Lines** και φροντίζουμε να έχουμε επιλέξει την επιλογή **Mesh > Quadratic elements > Normal**, ώστε τα στοιχεία να είναι γραμμικά. Μετά την δήλωση των στοιχείων του προβλήματος, τέλος επιλέγουμε το **Mesh > Generate mesh**. Θα εμφανιστεί ένα παράθυρο που φαίνεται και στο σχήμα 3.2.9. Εκεί συμπληρώνουμε την τιμή **<0.3>** και πατάμε **Ok**.

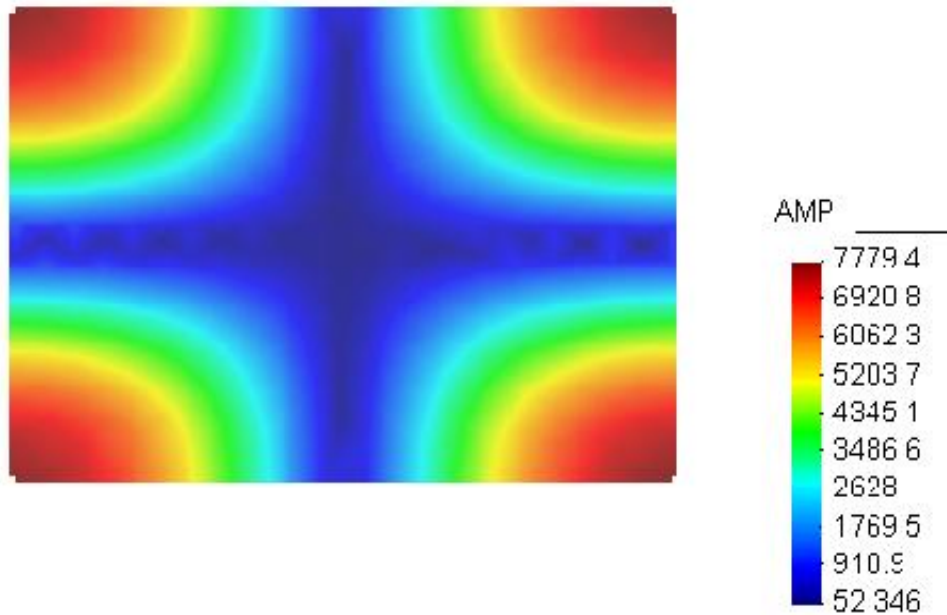


Σχήμα 3.2.9: Εισάγοντας το μέγεθος των στοιχείων

Σαν αποτέλεσμα παίρνουμε το σχήμα 3.2.10. Το Helm2D τώρα έχει όλα τα δεδομένα που χρειάζονται για να επιλύσει το πρόβλημα. Δίνοντας την εντολή **Calculate > calculate** γίνονται οι υπολογισμοί και ενημερώνεται ο χρήστης. Αν όλα πάνε καλά μπορούμε να δούμε τα αποτελέσματα με τον μετα-επεξεργαστή του GiD, μέσω της επιλογής **Files > Postprocess**. Μετά πατάμε **View Results > Contour Fill > AMP** και παίρνουμε τα αποτελέσματα όπως φαίνονται π.χ. στο σχήμα 3.2.11.

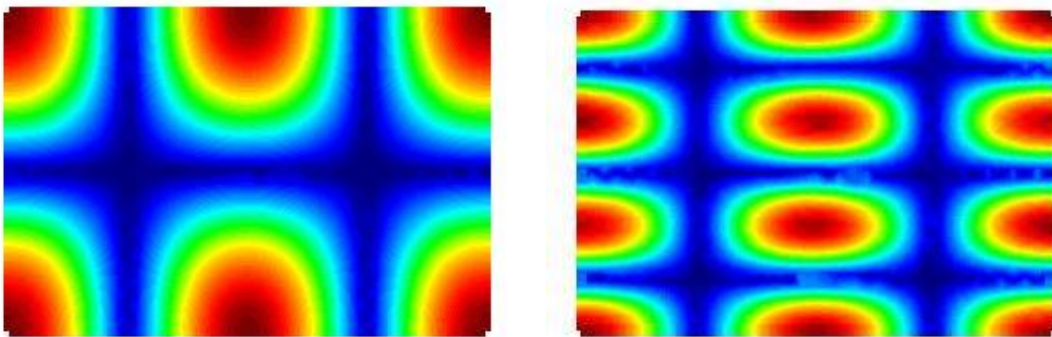


Σχήμα 3.2.10: Δικτύωμα του δωματίου



Σχήμα 3.2.11: Αναπαράσταση του μέτρου της πίεσης μέσα στο δωμάτιο για τον ιδιορυθμό (1,1).

Για να δούμε και άλλους ιδιορυθμούς επιστρέφουμε στον προ-επεξεργαστή και από την επιλογή **Data > Problem data**, στο παράθυρο **Frequency**, αλλάζουμε την συχνότητα. Όπως υπολογίζετε από την εξίσωση 3.2.6, οι ιδιορυθμοί (2,1) και (2,3) αντιστοιχούν στις συχνότητες 95.15 και 148.16. Αλλάζοντας τα δεδομένα και κάνοντας πάλι **calculate** θα πάρουμε τα αποτελέσματα του σχήματος 3.2.12



Σχήμα 3.2.12: Αναπαράσταση της πίεσης μέσα στο δωμάτιο όπως υπολογίστηκε για τους συντονισμούς (2,1), (2,3).

Θεωρία

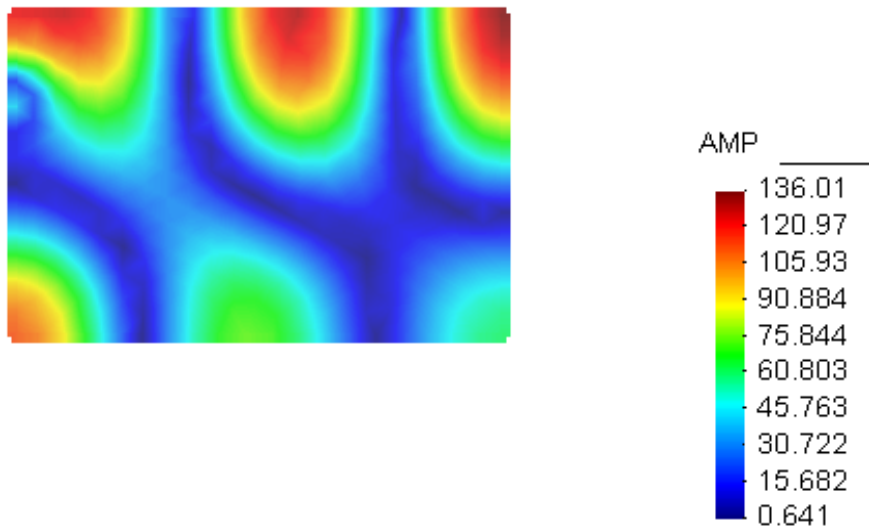
Σε δύο διαστάσεις ο υπολογισμός της συχνότητας κάθε ιδιοσυχνότητας δίδεται από την σχέση:

$$f_{n_x, n_y} = \frac{c}{2} \left[\left(\frac{n_x}{L_x} \right)^2 + \left(\frac{n_y}{L_y} \right)^2 \right]^{\frac{1}{2}} \quad (3.2.6)$$

όπου n_x , n_y ο αριθμός των δεσμών σε κάθε διάσταση και L_x , L_y οι διαστάσεις του δωματίου.

3.2.4: Κατανομή της πίεσης για διάφορες συχνότητες

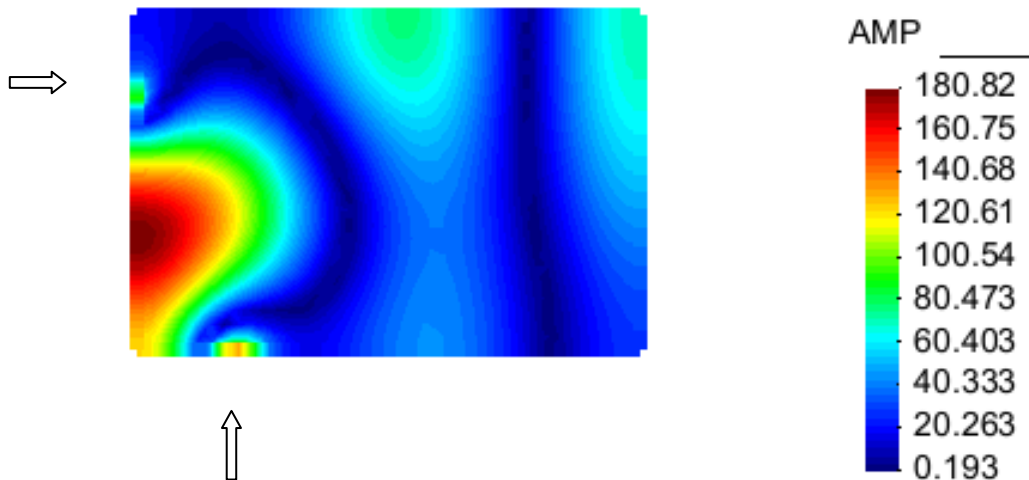
Για συχνότητες που είναι διαφορετικές από τις ιδιοσυχνότητες του παραπάνω προβλήματος μπορούμε τώρα να δούμε την κατανομή της πίεσης μέσα στο δωμάτιο. Επιστρέφουμε στον προ-επεξεργαστή από το **Files > Preprocess** και αλλάζουμε τα δεδομένα από το **Data > Problem data**. Στο παράθυρο **Frequencies** θέτουμε την τιμή <60> στη θέση **Frequency start**, και την τιμή <100> στη θέση **Frequency stop**. Τέλος αλλάζουμε την τιμή του **Frequency interval** από 0 σε <2>. Αυτή η αλλαγή θα λύσει το ίδιο πρόβλημα για όλες τις συχνότητες από 60 έως 100 με βήμα 2. Πατάμε **Accept** για να γίνει αποδοχή των νέων δεδομένων και ακολούθως **Calculate > calculate**. Τώρα από τον μετα-επεξεργαστή: **Files > Postprocess** μπορούμε να δούμε τις λύσεις με την επιλογή **View results > Contour fill > AMP** και να επιλέξουμε την συχνότητα π.χ. από το **View results > Default Analysis/step > frequency > 80**. Στο Παράρτημα A, στο τέλος της εργασίας παρατίθενται πολλά γραφήματα του ίδιου προβλήματος για διαφορετικές συχνότητες.



Σχήμα 3.2.13: Κατανομή της πίεσης στο εσωτερικό του δωματίου για την συχνότητα 80 Hz

3.2.5: Κατανομή πίεσης σε δωμάτιο δύο διαστάσεων με δύο πηγές

Επιστρέφουμε στον προ-επεξεργαστή και επιλέγουμε: **Geometry view** για να προσθέσουμε ακόμα μία πηγή, στο κάτω αριστερό άκρο. Η διαδικασία γίνεται με παρόμοιο τρόπο, επιλέγοντας: **Geometry > Edit > Divide > Line > Near point**. Προσθέτουμε τη νέα πηγή στην κάτω αριστερή γωνία του δωματίου. Αφού γίνει αυτό θα πρέπει να θέσουμε **Normal Velocity real <1>** στην περιοχή της δεύτερης πηγής. Τώρα επειδή άλλαξε η γεωμετρία θα πρέπει να ξαναγίνει το δικτύωμα, οπότε επιλέγουμε: **Mesh > Generate mesh**. Οι υπολογισμοί ενεργοποιούνται με την επιλογή **Calculate > calculate**. Στο σχήμα 3.2.14 φαίνεται η κατανομή της πίεσης για συχνότητα 70 Hertz. Περισσότερα γραφήματα του ίδιου προβλήματος για άλλες συχνότητες βρίσκονται στο Παράρτημα Α.

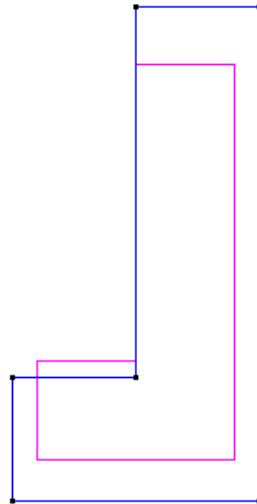


Σχήμα 3.2.14: Κατανομή της πίεσης σε δωμάτιο με δύο πηγές στις θέσεις που δείχνουν τα βέλη

3.2.6: Κατανομή της πίεσης σε δωμάτιο με σχήμα Γ

Όπως αναφέραμε και στην εισαγωγή αυτής της εργασίας, η ΜΣΣ γίνεται ιδιαίτερως χρήσιμη όταν έχουμε κάποιο πρόβλημα για το οποίο δεν υπάρχει η αναλυτική του λύση, είτε λόγω της γεωμετρίας του, είτε λόγω της πολυπλοκότητας του. Αυτή είναι και η ακόλουθη περίπτωση όπου έχουμε ένα χώρο μη συμμετρικό. Στον προ-επεξεργαστή του GiD δημιουργούμε ένα καινούργιο 'project' και το ονομάζουμε gamma.gid. Πρώτα θα γίνει η γεωμετρία του δωματίου. Ανοίγουμε το παράθυρο **Utilities > Tools > Coordinates window**, επιλέγουμε μετά **Geometry > Create > Straight line**. Εισάγουμε διαδοχικά τις παρακάτω συντεταγμένες : $\langle 0,0 \rangle$ $\langle 5,0 \rangle$ $\langle 5,10 \rangle$ $\langle 2.5,10 \rangle$ $\langle 2.5,2.5 \rangle$ $\langle 0,2.5 \rangle$ και τέλος $\langle 0,0 \rangle$ όπου και θα ερωτηθούμε αν θέλουμε να ενώσουμε ή όχι τα δύο τελευταία σημεία με τις επιλογές **Join** ή **No Join**. Διαλέγουμε την πρώτη και πατάμε

Esc. Με την επιλογή **Utilities > Layers** δημιουργούμε ένα layer με το όνομα **<field>**. Σιγουρευόμαστε πως δουλεύουμε σε αυτό το βοηθητικό layer και για την δημιουργία της επιφάνειας επιλέγουμε **Geometry > Create > Nurbs surface > By contour**. Με το ποντίκι διαλέγουμε και τις γραμμές του δωματίου και πατάμε **Esc**.

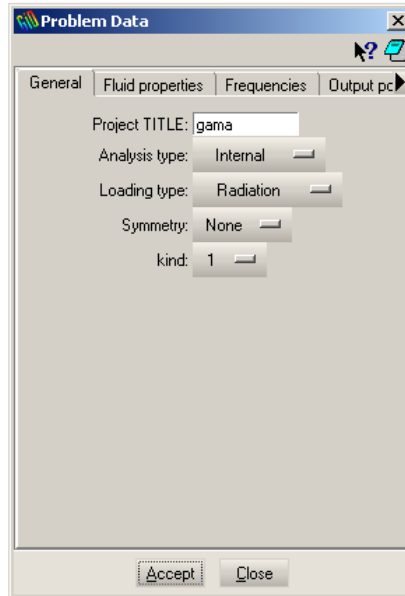


Σχήμα 3.2.15: Γεωμετρική αναπαράσταση δωματίου σχήματος 'Γ'

Συνοριακές συνθήκες

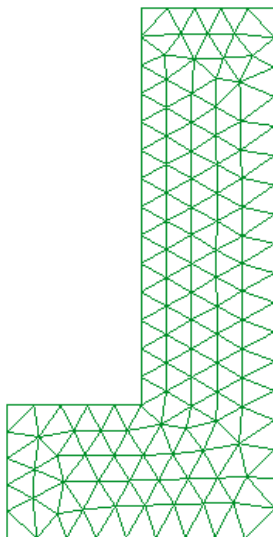
Επιλέγουμε **Data > Conditions**. Από το παράθυρο που θα ανοίξει επιλέγουμε **Velocity**. Πρώτα δηλώνουμε μηδενική ταχύτητα σε όλες τις πλευρές του δωματίου. Θέτουμε την τιμή **<0>** στα **Normal Velocity real** και **imaginary** και επιλέγουμε **Assign**. Διαλέγουμε όλες τις γραμμές του δωματίου και πατάμε **Esc**. Μετά αλλάζουμε την τιμή του **Normal Velocity real** σε **<1>** και πατάμε πάλι **Assign**, αυτή την φορά επιλέγουμε μόνο την πάνω οριζόντια πλευρά του δωματίου. Για να δηλώσουμε τα υπόλοιπα δεδομένα πηγαίνουμε στο **Data > Problem data** και συμπληρώνουμε τα παρακάτω στοιχεία όπως φαίνεται και στο σχήμα 3.2.16

Analysis type: Internal
Loading type: Radiation
Symmetry: None
kind: 1



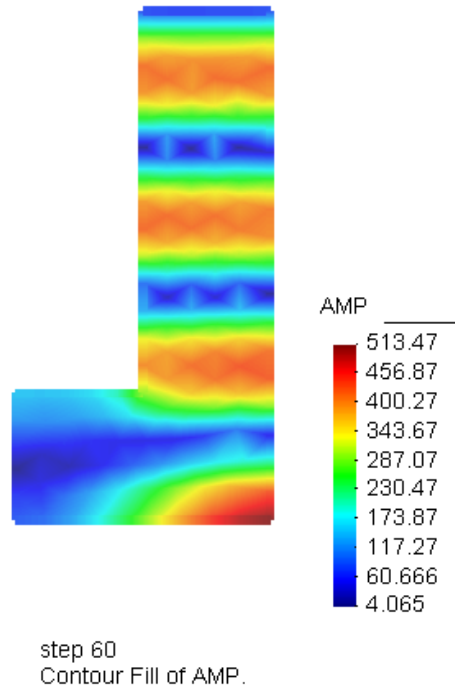
Σχήμα 3.2.16: Παράθυρο δεδομένων προβλήματος

Στην επιλογή **Fluid properties** αφήνουμε τις προεπιλεγμένες τιμές και στην επιλογή **Frequencies** δίνουμε την τιμή <60> για το **Frequency start**, <100> για το **Frequency stop** και <20> για το **Frequency interval**. Έτσι το Helm2D θα λύσει το πρόβλημα τρεις φορές για τις συχνότητες 60, 80 και 100. Στο παράθυρο **Output points** διαλέγουμε την επιλογή **Mesh**. Τέλος σώζουμε τα δεδομένα με το **Accept**. Στην συνέχεια θα εισάγουμε τις επιλογές δικτύωσης του δωματίου από το **Mesh > Mesh criteria > Mesh > Lines**. Επιλέγουμε όλες τις πλευρές και πατάμε **Esc**. Φροντίζουμε το **Mesh > Quadratic elements** να είναι **Normal**. Τέλος μπορούμε να κάνουμε το δικτύωμα επιλέγοντας **Mesh > Generate mesh**. Στο παράθυρο που θα εμφανιστεί συμπληρώνουμε την τιμή <0.5> και πατάμε **Ok**.



Σχήμα 3.2.17: Δικτύωμα του δωματίου 'Γ'

Τώρα μπορούμε πατώντας **Calculate > calculate** να δούμε την πίεση στο εσωτερικό του δωματίου από τον μετα-επεξεργαστή με την επιλογή **View results > Contour fill > AMP**. Επιπλέον γραφήματα παρατίθενται στο Παράρτημα Α, καθώς και στο CD υπάρχουν βίντεο με διαδοχικές λύσεις (animated) του προβλήματος.



Σχήμα 3.2.18: Στάσιμα κύματα στο εσωτερικό δωματίου 'Γ' για την συχνότητα 60 Hz

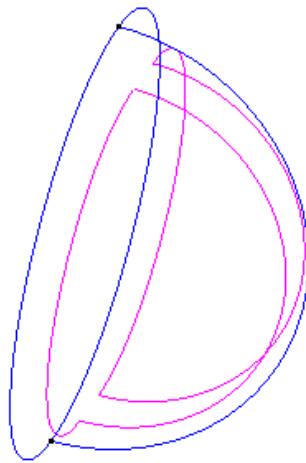
Προβλήματα σε τρεις διαστάσεις

Τα παρακάτω προβλήματα είναι τριών διαστάσεων συνεπώς χρειάζεται να έχει αντιγραφεί η εφαρμογή Helm3D στον φάκελο Problem types στον φάκελο εγκατάστασης του GiD. Πλέον οι field point λύσεις μπορούν να δίνονται σε ευθείες ή καμπύλες γραμμές. Με την επιλογή Mesh μπορεί ο χρήστης να αναζητά επιπλέον λύσεις πάνω σε επιφάνειες που θα σχεδιάζει εντός ή εκτός των γεωμετριών, ανάλογα με τον τύπο του προβλήματος. Αυτές οι επιφάνειες πρέπει να ανήκουν πάλι σε ένα ξεχωριστό επίπεδο (layer) σχεδίασης του οποίου το όνομα θα είναι 'field'. Πρέπει όμως να προσέξουμε το γεγονός ότι δεν μπορούμε να αναζητήσουμε field point λύσεις πάνω στα σύνορα, πράγμα που σημαίνει πως όταν σχεδιάζουμε επιπλέον επιφάνειες, θα πρέπει να προσέξουμε να μην έχουν όρια που να εφάπτονται ή να ξεπερνούν τα σύνορα. Αναλυτικότερα θα φανεί η διαδικασία στο παράδειγμα 3.3.5

3.3.1: Εκπομπή κύματος από ακτινικά παλλόμενη σφαίρα

Σχεδίαση γεωμετρίας

Το GiD έχει κάποια γεωμετρικά σχήματα και αντικείμενα έτοιμα προς σχεδίαση. Αυτά βρίσκονται στην επιλογή **Geometry > Create > Object** από όπου θα επιλέξουμε **Sphere**. Δίνουμε το κέντρο της σφαίρας $\langle 0 \ 0 \ 0 \rangle$ στη γραμμή εντολών (command) και έπειτα την ακτίνα του κύκλου $\langle 1 \rangle$ και πατάμε **Esc**. Σ' αυτήν την περίπτωση δεν μας χρειάζεται η πληροφορία της μισής σφαίρας αφού υπάρχει συμμετρία στο πρόβλημά μας, έτσι θα κάνουμε **Geometry > Delete > Surface** στις δύο επιφάνειες που βρίσκονται στον άξονα x , καθώς και τις γραμμές που τις περιγράφουν. Έτσι το σχήμα πρέπει να μοιάζει με το ημισφαίριο του σχήματος 3.3.1



Σχήμα 3.3.1 Γεωμετρική αναπαράσταση ενός ημισφαιρίου

Δήλωση συνοριακών συνθηκών

Επιλέγουμε το **Data > Problem type > Helm3D** από το 'menu' επιλογών. Εφ' όσον φορτωθεί το πρόβλημα μπορούμε να δηλώσουμε τις συνοριακές συνθήκες από το **Data > Conditions**. Από το παράθυρο που θα ανοίξει δηλώνουμε **Normal Velocity real** $\langle 1 \rangle$ και **Normal Velocity imaginary** $\langle 0 \rangle$ και πατάμε **Assign**. Επιλέγουμε με το ποντίκι τις δυο αυτές επιφάνειες και πατάμε **Esc**. Πηγαίνουμε μετά στο **Data > Problem data** για την δήλωση των λοιπών στοιχείων του προβλήματος. Δίνουμε ένα όνομα στο project π.χ. sphere_rad και επιλέγουμε:

Analysis type: Exterior
Loading type: Radiation
Symmetry: Y-Z

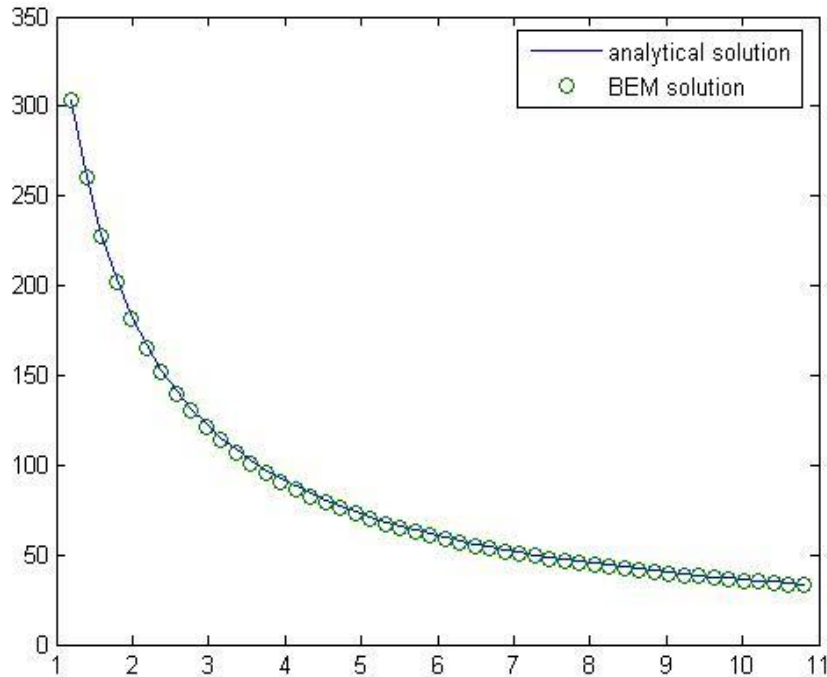
Η χρήση των επιλογών συμμετρίας μας εξοικονομεί υπολογιστική ισχύ και μειώνεται σημαντικά ο χρόνος υπολογισμού. Η γεωμετρία καθώς και οι συνοριακές συνθήκες στο πρόβλημα αυτό είναι συμμετρικές ως προς τους άξονες Y-Z, οπότε μπορούμε να χρησιμοποιήσουμε τις επιλογές συμμετρίας. Στην περιοχή των συχνοτήτων δηλώνουμε στα **Frequency start** και **Frequency stop** την τιμή <100>. Ζητάμε λύσεις στο πεδίο έξω από την σφαίρα στο **Output points**, και επιλέγουμε να είναι πάνω σε μία ευθεία από το **Field point type: Straight line**. Πάνω στον άξονα x από <1.2> με βήμα <0.2>, σε <50> σημεία.

Τέλος φτιάχνουμε το δικτύωμα της σφαίρας με την επιλογή **Mesh > Generate mesh**, και λύνουμε το πρόβλημα με το **Calculate > calculate**. Ένα αρχείο με το όνομα output.dat θα δημιουργηθεί στο φάκελο εργασίας του project, από όπου μπορούμε να δούμε τα αποτελέσματα. Στο γράφημα του σχήματος 3.2.2 μπορούμε να δούμε μία σύγκριση της αναλυτικής λύσης με την ΜΣΣ.

Θεωρία

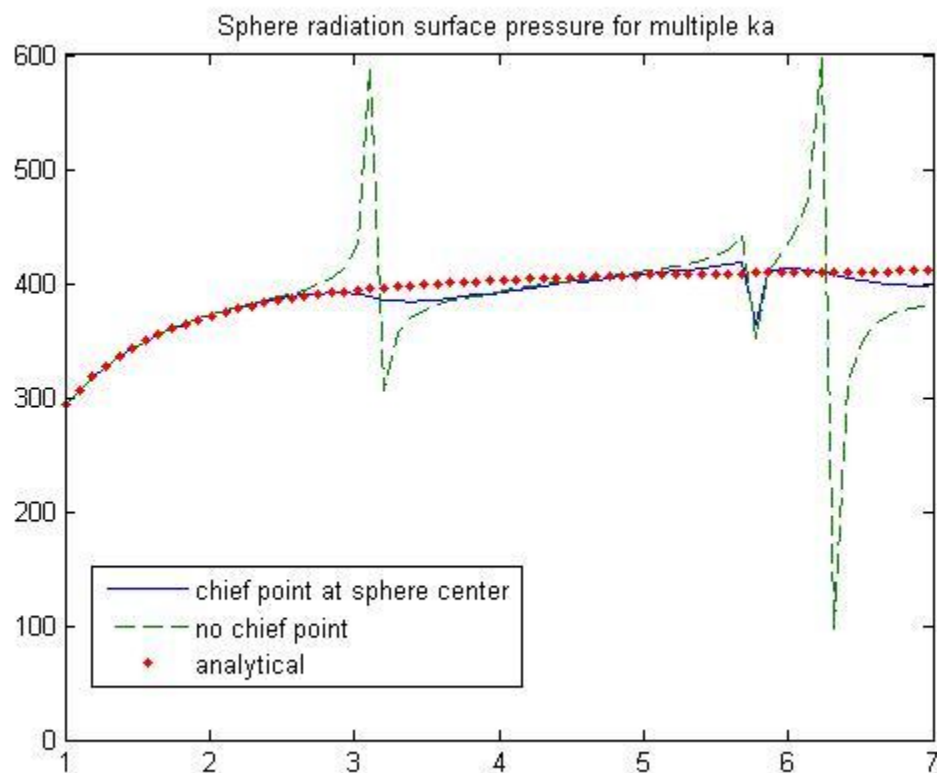
Η αναλυτική λύση για την ακουστική πίεση που εκπέμπεται από σφαίρα μπορεί να εξαχθεί από την σφαιρική κυματική εξίσωση, και είναι:

$$p(r) = \left(\frac{\alpha^2}{r}\right) \frac{i\rho\omega}{1+ika} e^{-ik(r-a)} \quad (3.3.1)$$



Γράφημα 3.3.2 Ακουστική πίεση έναντι απόστασης στο πρόβλημα της παλλόμενης σφαίρας, για σταθερή γωνία

Επιστρέφουμε στον προ-επεξεργαστή και ανοίγουμε το **Problem data** για να εξετάσουμε πως διαμορφώνεται η ακουστική πίεση για διάφορες ιδιοσυχνότητες της σφαίρας. Θα ζητήσουμε λύσεις για πολλαπλές συχνότητες με **Frequency start <55>**, **Frequency stop <380>** και **Frequency interval <5>**. Οι παραπάνω τιμές αντιστοιχούν σε $ka = 1$ ως $ka = 7$. Επίσης στην επιλογή **Chief points** προσθέτουμε ένα σημείο στο κέντρο της σφαίρας ($x_0 <0>$ $y_0 <0>$ $z_0 <0>$). Στο γράφημα του σχήματος 3.3.3 φαίνεται η ακουστική πίεση στην επιφάνεια της σφαίρας για πολλές ιδιοσυχνότητες της σφαίρας, με την χρήση ή μη της μεθόδου chief point, καθώς και η σύγκριση τους με την αναλυτική λύση. Η direct BEM παρουσιάζει λανθασμένες λύσεις στις φυσικές ιδιοσυχνότητες της γεωμετρίας, όταν οι λύσεις αυτές ζητηθούν έξω απ' αυτήν. Το πρόβλημα αυτό μπορεί να ξεπεραστεί με την χρήση της μεθόδου chief point. Στην περίπτωση της σφαίρας οι φυσικές ιδιοσυχνότητες αντιστοιχούν σε $ka = \pi$ και $ka=2\pi$.

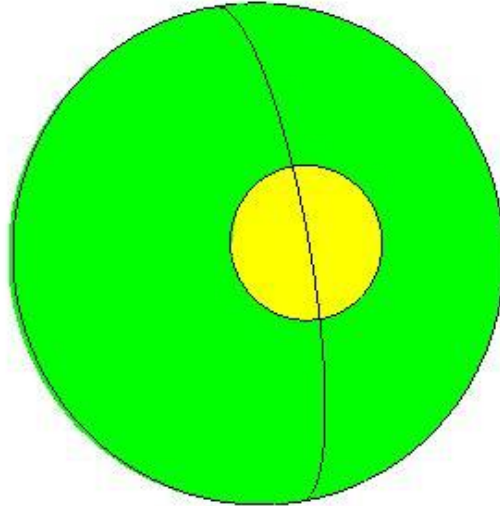


Γράφημα 3.3.3: Σύγκριση της chief point μεθόδου με την αναλυτική λύση για την παλλόμενη σφαίρα

3.3.2: Μοντέλο ηχείου, εκπομπή από σφαιρικό πιστόνι πάνω σε σφαίρα

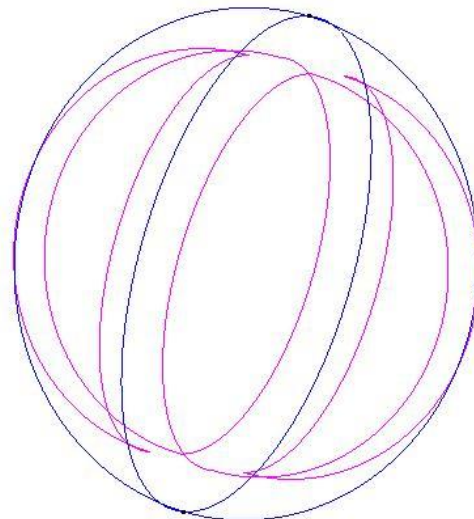
Κατασκευή γεωμετρίας

Το μοντέλο που φαίνεται στο σχήμα 3.3.5 χρειάζεται μία σειρά βημάτων για να σχεδιαστεί στο GiD. Δημιουργούμε ένα νέο project με τίτλο sphere_cap.gid.



Σχήμα 3.3.5: Μοντέλο ηχείου

Για τη σχεδίαση μίας ολόκληρης σφαίρας επιλέγουμε **Geometry > Create > Object > Sphere**. Θα μας ζητηθεί να εισάγουμε το κέντρο της σφαίρας. Στο command line εισάγουμε διαδοχικά **<0 0 0>** και πατάμε Enter. Έπειτα θα μας ζητηθεί να δώσουμε την ακτίνα της σφαίρας. Αντίστοιχα θα εισάγουμε την τιμή **<0.2>**. Η σφαίρα θα πρέπει να εμφανιστεί στην οθόνη. Μεγεθύνουμε με την επιλογή **View > Zoom > Frame**.

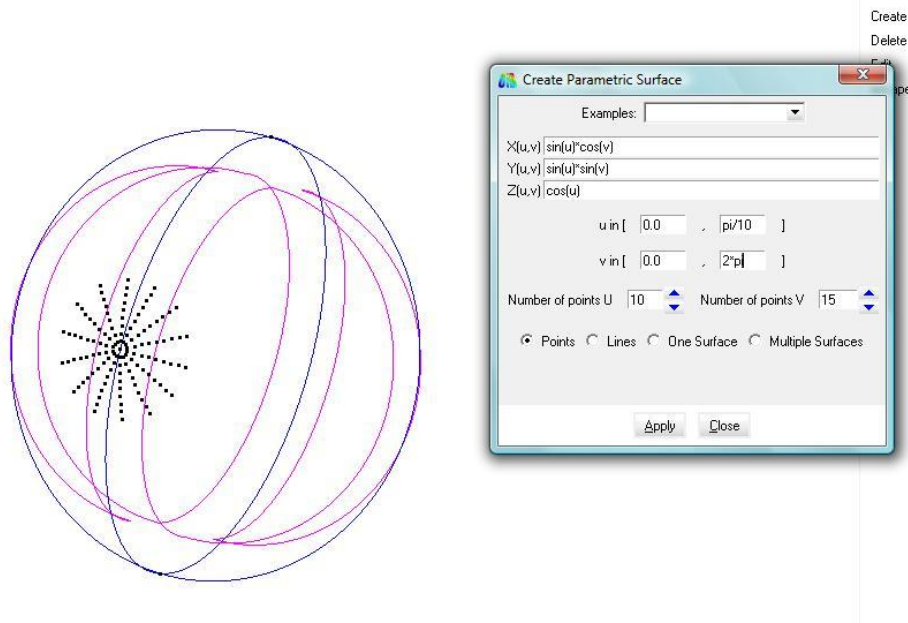


Σχήμα 3.3.6: Μοντέλο σφαίρας στο GiD

Τώρα για την δημιουργία του cap θα επιλέξουμε **Geometry > Create > Parametric surface**. Από το παράθυρο που θα ανοίξει θα συμπληρώσουμε σε καρτεσιανές συντεταγμένες τις εξισώσεις που ορίζουν αυτή την σφαιρική επιφάνεια.

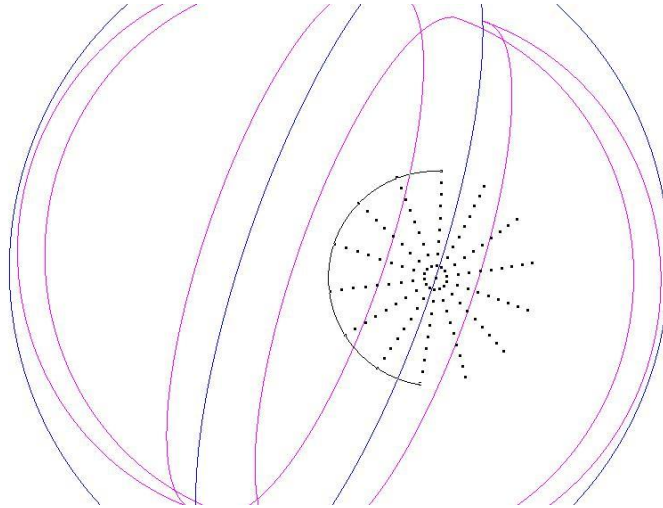
$$\begin{aligned}x(u, v) &= 0.2 * \sin(u) * \cos(v), \\y(u, v) &= 0.2 * \sin(u) * \sin(v), \\z(u, v) &= 0.2 * \cos(u), \\u &= [0, \frac{\pi}{10}] \\v &= [0, 2 * \pi]\end{aligned}$$

Επίσης επιλέγουμε **points** για την σχεδίαση αυτής της επιφάνειας. Το αποτέλεσμα θα πρέπει να είναι αυτό του σχήματος 3.3.7. Μετά ορίζουμε δύο καμπύλες γραμμές από **Geometry > Create > Arc > By 3 points** και επιλέγοντας **join**, για να διαλέξουμε τρία



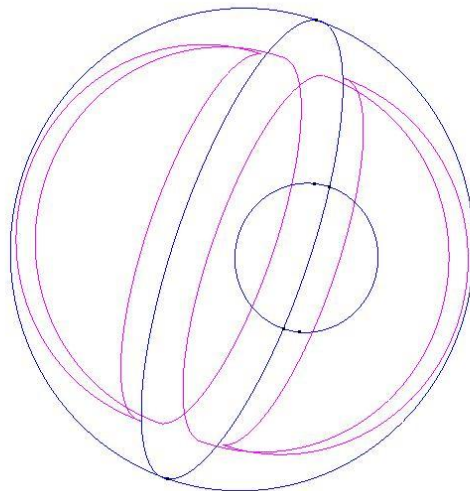
Σχήμα 3.3.7: Παράθυρο εισαγωγής παραμετροποιημένης επιφάνειας

διαδοχικά σημεία, που να ορίσουν την περιφέρεια του cap με δύο καμπύλες γραμμές όπως φαίνεται στο σχήμα 3.3.8.



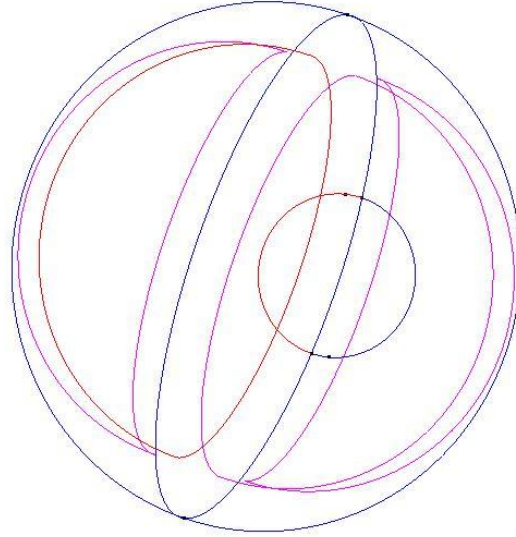
Σχήμα 3.3.8: Ορίζοντας την περιφέρεια του cap

Τώρα θα διαγράψουμε τα υπόλοιπα σημεία με **Geometry > Delete > Points** και επιλέγοντας με το ποντίκι όλα τα points και πατώντας **Esc**, θα διαγραφούν όλα όσα δεν ανήκουν σε κάποια γραμμή. Οι δύο αυτές καμπύλες γραμμές που δημιουργήθηκαν, τέμνονται με την γραμμή που χωρίζει τα δύο εμπρόσθια τεταρτημόρια της σφαίρας, επιλέγουμε **Geometry > Edit > Intersection > Line-line** και με το ποντίκι επιλέγουμε πρώτα την μία γραμμή του cap και μετά την γραμμή με την οποία τέμνεται, θα δημιουργηθεί ένα point στο σημείο εκεί όπου τέμνονται, και οι γραμμές θα χωριστούν. Επαναλαμβάνουμε την διαδικασία για την άλλη γραμμή του cap ώστε να καταλήξουμε στο αποτέλεσμα του σχήματος 3.3.9. Τώρα αυτό που μας έμεινε είναι να χωρίσουμε το κάθε ένα από τα δύο εμπρόσθια τεταρτημόρια από τις δύο μικρές επιφάνειες που θα ορίζουν το cap. Θα επιλέξουμε **Geometry > Edit > Divide > Surfaces > Split**, θα μας ζητηθεί να επιλέξουμε την επιφάνεια που θα χωριστεί, με το ποντίκι διαλέγουμε την επιφάνεια που ορίζει το αριστερό μπροστά τεταρτημόριο.

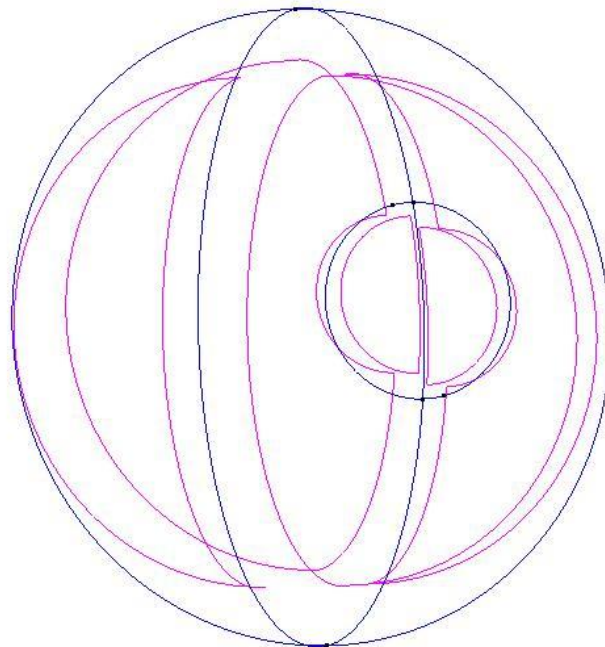


Σχήμα 3.3.9: Διχοτόμηση της περιφέρειας του πιστονιού με την περιφέρεια της σφαίρας

Μετά μας ζητείται να δώσουμε τις γραμμές που ορίζουν το χωρισμό της επιφάνειας, και επιλέγουμε τις δύο αυτές γραμμές όπως φαίνεται και στο σχήμα 3.3.10 και πατάμε **Esc**. Επαναλαμβάνουμε την διαδικασία και για το δεξιό τεταρτημόριο και η γεωμετρία μας είναι έτοιμη.



Σχήμα 3.3.10: Κάνοντας *split* στις επιφάνειες της σφαίρας



Σχήμα 3.3.11: Μοντέλο σφαίρας με σφαιρικό πιστόνι

Δήλωση συνοριακών συνθηκών

Τώρα θέλοντας να εξετάσουμε τον τρόπο που εκπέμπει το μοντέλο αυτό θα επιλέξουμε **Data > Problem type > Helm3D**, και αφού ενημερωθούμε από το command line ότι φορτώθηκε σωστά θα επιλέξουμε **Data > Conditions**. Από το παράθυρο που θα ανοίξει θα διαλέξουμε **Velocity**. Θα συμπληρώσουμε την τιμή <0> στα **Normal Velocity Re** και **Normal Velocity Im** και μετά πατάμε **Assign**. Με το ποντίκι επιλέγουμε όλες τις επιφάνειες του project και πατάμε **Esc**. Μετά αλλάζουμε την τιμή **Normal Velocity Re** σε <1> και πατάμε πάλι **Assign**, επιλέγοντας αυτή την φορά μόνο τις δύο επιφάνειες που ορίζουν το σφαιρικό πιστόνι.

Δήλωση λοιπών στοιχείων του προβλήματος

Ανοίγουμε το **Data>Problem data** και συμπληρώνουμε αντίστοιχα

Project title: cap

Analysis type: External

Loading type: Radiation

Symmetry: None

Στο παράθυρο **Fluid properties** αφήνουμε τις υπάρχουσες τιμές και στο **Frequencies** δίνουμε τις τιμές <175>, <175>, <0>. Θέλοντας να εξετάσουμε την πίεση από z έως -z σε μία σταθερή απόσταση από την σφαίρα, στην επιλογή **Output points** θα ζητήσουμε field point λύσεις σε καμπύλη γραμμή (**Arc**). Ζητάμε <99> σημεία, το κέντρο της καμπύλης θα συμπίπτει με της σφαίρας <0 0 0>, δίνουμε ακτίνα (**Arc radius**) <0.5>, **Start angle** <90> και **Angle step** <1.8>.

Δικτύωση και υπολογισμός πίεσης

Τέλος από την επιλογή **Mesh > Generate mesh** εξάγουμε το δικτύωμα του μοντέλου και μετά με την επιλογή **Calculate > calculate** οδηγούμε το πρόβλημα σε λύση. Το παρόν πρόβλημα αποτελεί υποπερίπτωση του επομένου, έτσι οι λύσεις παρουσιάζονται παρακάτω μετά την αντίστοιχη θεωρία.

3.3.3: Εκπομπή από σφαιρικό δακτύλιο πάνω σε σφαίρα

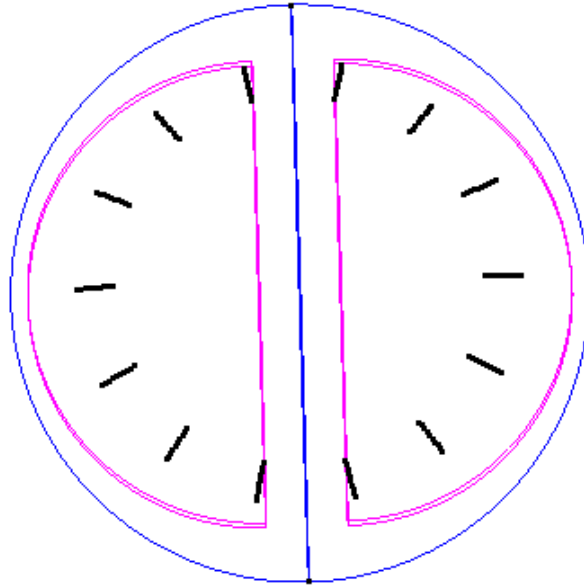
Κατασκευή γεωμετρίας

Με παρόμοιο τρόπο θα σχεδιαστεί και αυτό το μοντέλο. Το οποίο, όπως θα φανεί από την θεωρία, έχει την ίδια αναλυτική λύση ως πρόβλημα, με το προηγούμενο παράδειγμα να αποτελεί υποπερίπτωση αυτού.

Δημιουργούμε ένα καινούργιο 'project' με το όνομα sphere_strip.gid. Για τη σχεδίαση μίας ολόκληρης σφαίρας επιλέγουμε **Geometry > Create > Object > Sphere**. Θα μας ζητηθεί να εισάγουμε το κέντρο της σφαίρας. Στη γραμμή εντολών εισάγουμε <0 0 0> και πατάμε 'Enter'. Έπειτα θα μας ζητηθεί να δώσουμε την ακτίνα της σφαίρας.

Αντίστοιχα θα εισάγουμε την τιμή <0.2>. Η σφαίρα θα πρέπει να εμφανιστεί στην οθόνη. Μεγεθύνουμε με την επιλογή **View > Zoom > Frame**.

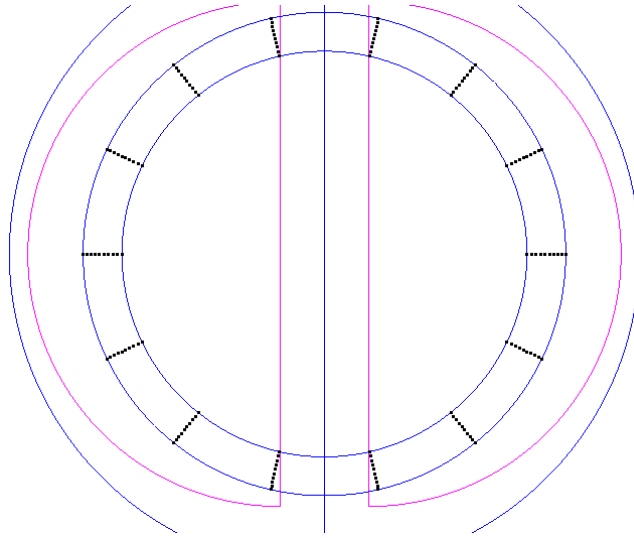
Τώρα για την δημιουργία του δακτύλιου θα επιλέξουμε **Geometry > Create > Parametric surface**. Από το παράθυρο που θα ανοίξει θα συμπληρώσουμε σε καρτεσιανές συντεταγμένες τις εξισώσεις που ορίζουν αυτή την σφαιρική επιφάνεια.



Σχήμα 3.3.12: Σημεία που ορίζουν τον δακτύλιο

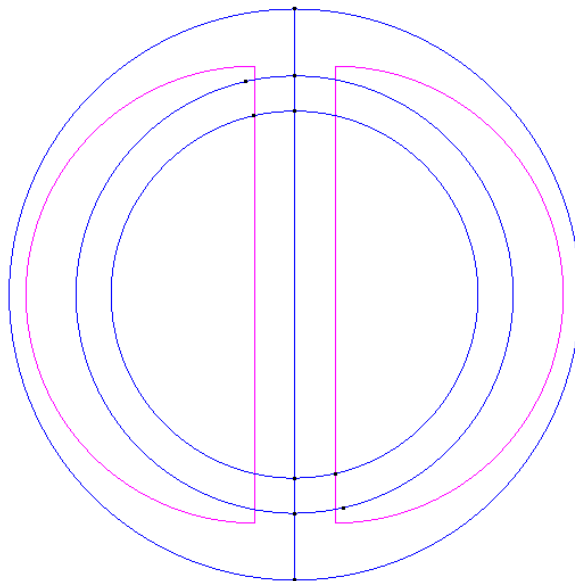
$$\begin{aligned}x(u, v) &= 0.2 * \sin(u) * \cos(v), \\y(u, v) &= 0.2 * \sin(u) * \sin(v), \\z(u, v) &= 0.2 * \cos(u), \\u &= [\pi/3.6, \pi/4.5] \\v &= [0, 2 * \pi]\end{aligned}$$

Παρατηρούμε πως αυτήν την φορά το μόνο που αλλάξαμε είναι η παράμετρος u η οποία κυμαίνεται από γωνία $\pi/3.6$ έως $\pi/4.5$, που αντιστοιχούν σε 40 και 50 μοίρες, σε αντίθεση με το πρώτο παράδειγμα όπου ήταν από 0 έως $\pi/10$, δηλαδή από 0 έως 18 μοίρες. Επίσης επιλέγουμε **points** για την σχεδίαση αυτής της επιφάνειας. Το αποτέλεσμα θα πρέπει να είναι αυτό του σχήματος (3.3.12). Μετά ορίζουμε δύο καμπύλες γραμμές από **Geometry > Create > Arc > By 3 points** και επιλέγοντας **join**, για να διαλέξουμε τρία διαδοχικά σημεία, που να ορίσουν την εξωτερική περιφέρεια του δακτυλίου, και επαναλαμβάνουμε την διαδικασία για την εσωτερική περιφέρεια του δακτυλίου, όπως βλέπουμε στο σχήμα (3.3.13).



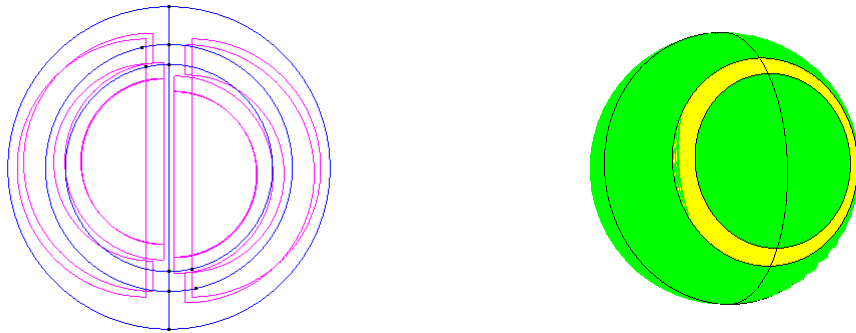
Σχήμα 3.3.13: Επιλέγοντας σημεία για τον ορισμό της περιμέτρου

Τώρα θα διαγράψουμε τα υπόλοιπα σημεία με **Geometry > Delete > Points** και επιλέγοντας με το ποντίκι όλα τα σημεία και πατώντας **Esc**. Θα διαγραφούν όλα εκείνα που δεν ανήκουν σε κάποια γραμμή. Οι τέσσερις αυτές καμπύλες γραμμές που δημιουργήθηκαν, τέμνονται με την γραμμή που χωρίζει τα δύο εμπρόσθια τεταρτημόρια της σφαίρας, και με την επιλογή **Geometry > Edit > Intersection > Line-line** θα διαχωρίσουμε μεταξύ τους αυτές τις γραμμές όπου αυτές τέμνονται. Έτσι με το ποντίκι επιλέγουμε πρώτα την μία γραμμή του δακτυλίου και μετά την γραμμή με την οποία τέμνεται, θα δημιουργηθεί ένα σημείο (point) στο σημείο εκεί που τέμνονται, και οι γραμμές θα χωριστούν. Επαναλαμβάνουμε την διαδικασία για όλες τις γραμμές του δακτυλίου ώστε να καταλήξουμε στο αποτέλεσμα του σχήματος (3.3.14). Παρατηρούμε πως εκεί που τέμνονται οι γραμμές, πλέον υπάρχουν σημεία (points).



Σχήμα 3.3.14: Ο διαχωρισμός των γραμμών (βλ. κείμενο)

Τώρα αυτό που μας έμεινε είναι να χωρίσουμε το κάθε ένα από τα δύο αυτά τεταρτημόρια από τις δύο μικρές επιφάνειες που θα ορίζουν τον δακτύλιο. Θα επιλέξουμε **Geometry > Edit > Divide > Surfaces > Split**, θα μας ζητηθεί να επιλέξουμε την επιφάνεια που θα χωριστεί, με το ποντίκι διαλέγουμε την επιφάνεια που ορίζει το αριστερό μπροστά τεταρτημόριο. Μετά μας ζητείται να δώσουμε τις γραμμές που ορίζουν το χωρισμό της επιφάνειας, και επιλέγουμε τις δύο γραμμές που ορίζουν την εσωτερική περίμετρο του δακτυλίου στο αριστερό τεταρτημόριο και πατάμε **Esc**. Επαναλαμβάνουμε την διαδικασία και για την εξωτερική περίμετρο. Αφού κάνουμε τα δύο τελευταία βήματα και για το δεξιό τεταρτημόριο η γεωμετρία είναι έτοιμη.



3.3.16: Γεωμετρικό μοντέλο της σφαίρας με δακτύλιο

Δήλωση συνοριακών συνθηκών

Δηλώνουμε ως problem type το Helm3D, εφ' όσον έχουμε ένα πρόβλημα στις τρεις διαστάσεις. Επιλέγουμε **Data > Conditions**. Στο παράθυρο **Velocity** ορίζουμε τα **Normal Velocity Re** και **Normal Velocity Im** ως $\langle 0 \rangle$ και πατάμε **Assign**, επιλέγουμε όλες τις επιφάνειες του project και πατάμε **Esc**. Έπειτα αλλάζουμε την τιμή του **Normal Velocity Re** σε $\langle 1 \rangle$ και κάνουμε πάλι **Assign**, αυτή την φορά μόνο στις δύο επιφάνειες που ορίζουν τον δακτύλιο.

Δήλωση λοιπών στοιχείων του προβλήματος

Ανοίγουμε το **Data > Problem data** και συμπληρώνουμε αντίστοιχα:

Project title: ring

Analysis type: External

Loading type: Radiation

Symmetry: None

Στο παράθυρο **Fluid properties** αφήνουμε τις υπάρχουσες τιμές και στο **Frequency** δίνουμε τις τιμές **Frequency start** $\langle 1500 \rangle$, **Frequency end** $\langle 3000 \rangle$, **Frequency interval** $\langle 1500 \rangle$. Αυτό σημαίνει πως το πρόβλημα θα λυθεί δύο φορές για τις συχνότητες 1500 και 3000. Θέλοντας να εξετάσουμε την πίεση από z έως $-z$ σε μία σταθερή απόσταση από την σφαίρα, στην επιλογή **Output points** θα ζητήσουμε field point λύσεις σε καμπύλη γραμμή (**Arc**). Ζητάμε $\langle 99 \rangle$ points, το κέντρο της καμπύλης θα

συμπίπτει με της σφαίρας <0 0 0>, δίνουμε ακτίνα (Arc radius) <0.5>, Start angle <90> και Angle step <1.8>.

Εξαγωγή δικτύωσης και υπολογισμός πίεσης

Τέλος από την επιλογή **Mesh > Generate mesh** εξάγουμε το δικτύωμα του μοντέλου και μετά με την επιλογή **Calculate > calculate** λύνουμε το πρόβλημα.

Θεωρία

Τα δύο παραπάνω προβλήματα υπάγονται στην ίδια αναλυτική λύση, με το πρώτο να αποτελεί υποπερίπτωση του δεύτερου. Για συμμετρική πηγή, η επιφανειακή ταχύτητα συνδέεται με την ακουστική πίεση ως εξής:

$$w(a, \beta, \theta) = \begin{cases} 0 & 0 \leq \theta \leq \alpha \\ \dot{w} & \alpha \leq \theta \leq \beta \\ 0 & \beta \leq \theta \leq \pi \end{cases}, \quad \beta < \alpha \quad (3.3.2)$$

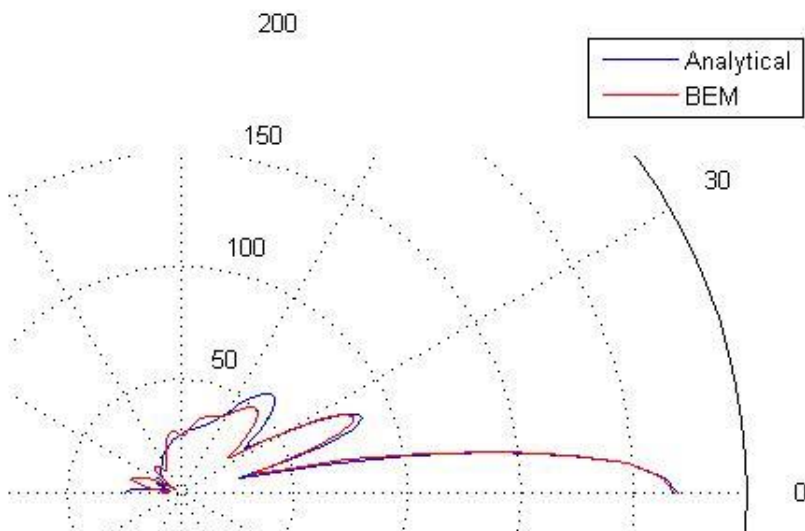
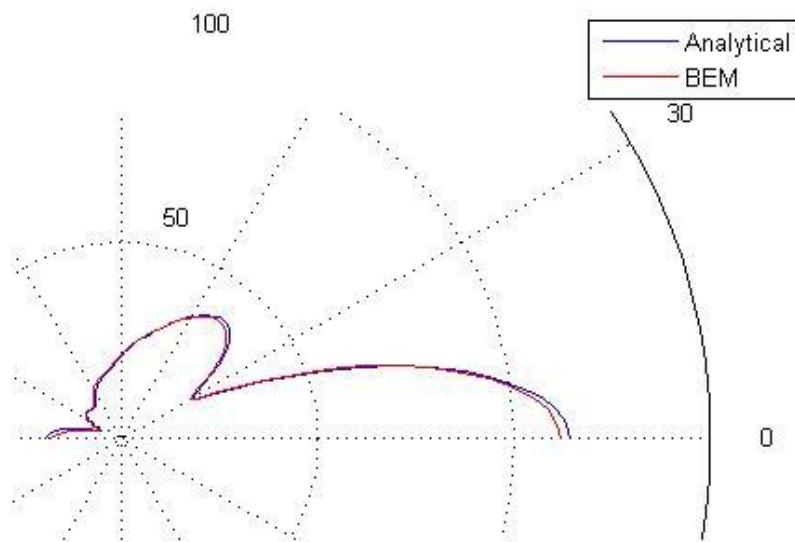
όπου α είναι η γωνία της εξωτερικής περιμέτρου του δακτυλίου και β η γωνία της εσωτερικής. Παρατηρούμε πώς αν $\beta = 0$ έχουμε την περίπτωση του συμμετρικού πιστονιού (cap), ενώ αν το β έχει τιμή (πάντα μικρότερη του α) τότε έχουμε την περίπτωση του συμμετρικού δακτυλίου (strip). Έτσι η ακουστική πίεση σε όποιο σημείο έξω από την σφαίρα με μέτρο r και γωνία θ μπορεί να υπολογιστεί από τον τύπο :

$$p(r, \theta) = \frac{i\rho_0 c \dot{w}}{2} [\cos\alpha - \cos\beta] \frac{h_0(kr)}{h'_0(kR)} + \frac{i\rho_0 c \dot{w}}{2} \sum_{n=1}^{\infty} [P_{n+1}(\cos\alpha) - P_{n+1}(\cos\beta) - P_{n-1}(\cos\alpha) + P_{n-1}(\cos\beta)] \frac{h_n(kr)}{h'_n(kR)} P_n(\cos\theta) \quad (3.3.3)$$

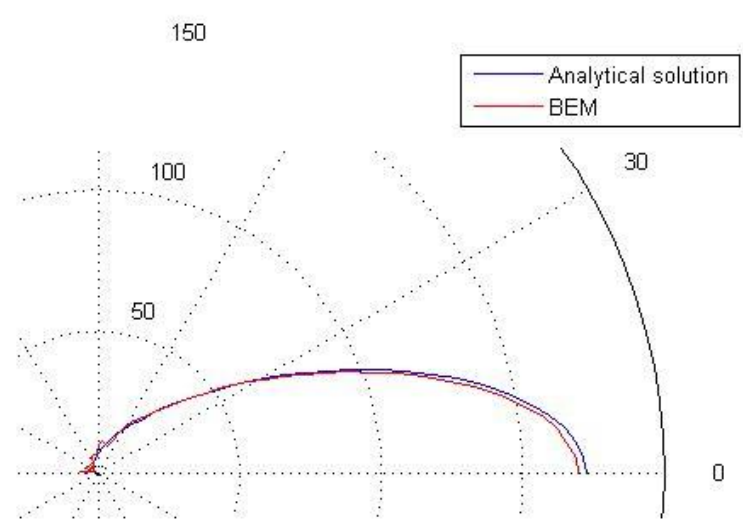
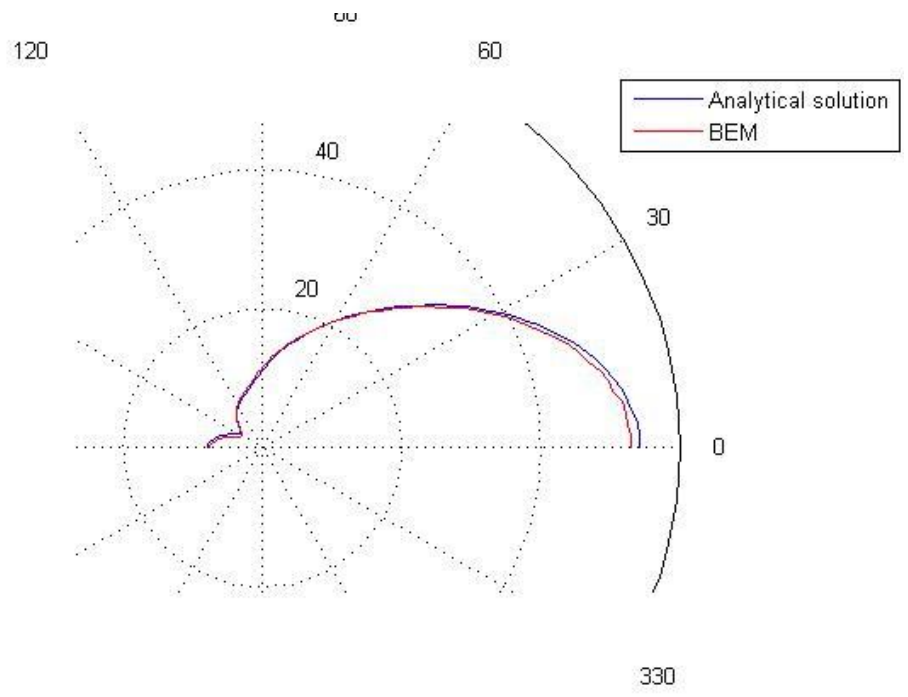
όπου P_n είναι το πολυώνυμο Legendre, h_0 η Hankel πρώτου βαθμού.

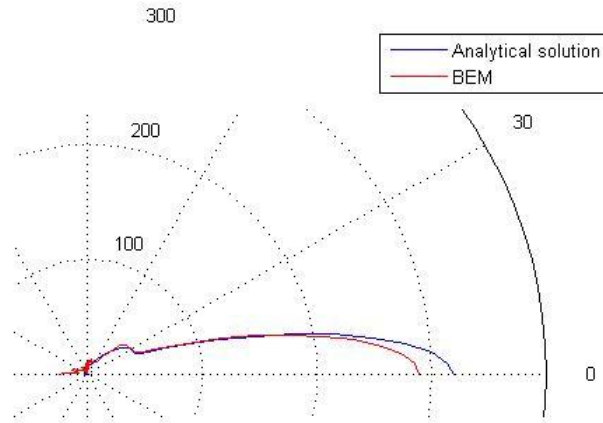
Αποτελέσματα

Όταν το GiD μας ενημερώσει πως η διαδικασία τελείωσε μπορούμε να αναζητήσουμε τις field point λύσεις στο αρχείο output.dat που βρίσκεται στο φάκελο του 'project'. Στο σχήμα 3.3.17 βλέπουμε τα αποτελέσματα για το συμμετρικό δακτύλιο σε σύγκριση με την αναλυτική λύση του προβλήματος όπως σχεδιάστηκε με την βοήθεια του MatLab, ενώ στο σχήμα 3.3.18 έχουμε τα συγκριτικά αποτελέσματα για το σφαιρικό πιστόνι.



Σχήμα 3.3.17: Ακουστική πίεση συναρτήσει της γωνίας θ για $kR = 5.5, 11$





Σχήμα 3.3.18: Ακουστική πίεση συναρτήσει της γωνίας θ για $kR = 1, 3, 6$ αντίστοιχα

3.3.4: Σκέδαση επίπεδου κύματος από μη ελαστική σφαίρα

Κατασκευή γεωμετρίας

Όμοια με τα προηγούμενα παραδείγματα θα φτιάξουμε μία σφαίρα με ακτίνα 1 μέτρο από την επιλογή **Geometry > Create > Object > Sphere** και θα αφαιρέσουμε την πληροφορία του όγκου με **Geometry > Delete > Volume**, επιλέγοντας τον όγκο και πατώντας **Esc**.

Δήλωση συνοριακών συνθηκών

Αφού δηλώσουμε το Helmh3d ως problem type μπορούμε να δηλώσουμε της συνθήκες και τα λοιπά στοιχεία του προβλήματος από το **Data > conditions** και **Data > Problem data**. Επιλέγοντας το πρώτο θα πρέπει να δηλώσουμε μηδενική ταχύτητα στα σύνορα της σφαίρας. Αυτό γίνεται κάνοντας **Assign velocity** και επιλέγοντας με το ποντίκι της δύο επιφάνειες της σφαίρας και πατώντας **Esc**, αφού πρώτα έχουμε συμπληρώσει την τιμή $\langle 0 \rangle$ στο **Normal Velocity Re** και **Normal velocity Im**.

Δήλωση λοιπών στοιχείων του προβλήματος

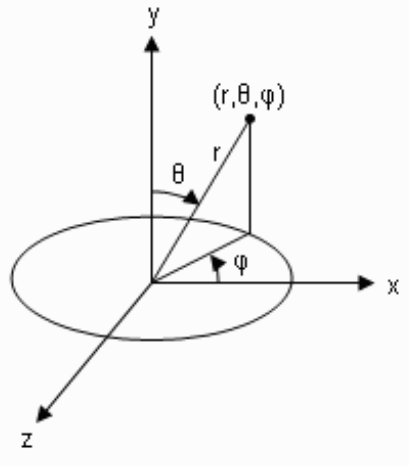
Επιλέγουμε το παράθυρο problem data για να δώσουμε τα στοιχεία του προβλήματος. Ονομάζουμε το 'project' sphere, και μετά εφ' όσον θέλουμε να ζητήσουμε λύσεις έξω από την σφαίρα.

Analysis type: External
Loading type: Scattering
Symmetry: None

Από τα νέα παράθυρα που θα εμφανιστούν για την επιλογή ‘scattering’ μπορούμε να αφήσουμε τις υπάρχουσες τιμές δηλαδή προσπίπτον κύμα έντασης <1> (Planewave amplitude), με κατεύθυνση από $-z$ σε z (Planewave Lz <-1>). Στην επιλογή **Fluid properties** συμπληρώνουμε στο **Density** <1.21>, στο **Speed of sound** <343>, και στο **Reference pressure** <20e-6>. Για την επιλογή συχνότητας συμπληρώνουμε την τιμή <171.5> στις θέσεις **Frequency start** και **Frequency stop**. Στην συνέχεια πηγαίνουμε στο παράθυρο **Output points** απ’ όπου και θα ζητήσουμε λύσεις σε ακτίνα δύο μέτρων και σε γωνίες από $-\pi$ μέχρι π για τον άξονα xz . Για να γίνει αυτό επιλέγουμε **Field point type** <Arc>, **Number of arc field points** <100>, **Arc center** <0,0,0>, **Axis direction** <Y>, **Arc radius** <2> και τέλος **Start angle** <0>, **Angle step** <1.8>. Αφού σώσουμε τα δεδομένα αυτά με το **Accept** μπορούμε να προχωρήσουμε στη λύση με το **Calculate** > **calculate**. Οι λύσεις μαζί με όλα τα στοιχεία του προβλήματος θα πρέπει να βρίσκονται στο φάκελο του ‘project’ με το όνομα output.dat.

Θεωρία

Πολλά προβλήματα σκέδασης εξετάζονται για επίπεδα κύματα που προσπίπτουν σε κάποιον όγκο. Εδώ επιλέξαμε τη σφαίρα γιατί υπάρχει αναλυτική λύση οπότε μπορούμε να διακριβώσουμε την αποτελεσματικότητα της μεθόδου. Για λόγους μαθηματικής απλότητας, επιλέγουμε την κατεύθυνση του κύματος να οδεύει πάνω σ’ έναν άξονα, στην περίπτωση μας στον άξονα z .



Σχήμα 3.3.6: Σύστημα σφαιρικών συντεταγμένων (r, θ, φ)

Πριν εξετάσουμε το πρόβλημα πρέπει να εντάξουμε τη σφαίρα σε ένα σύστημα σφαιρικών συντεταγμένων το οποίο και δίδεται στο σχήμα (3.3.6). Οι σχέσεις που το συνδέουν με το αντίστοιχο καρτεσιανό είναι:

$$\begin{aligned} x &= r \sin \theta \cos \varphi \\ y &= r \sin \theta \sin \varphi \\ z &= r \cos \theta \end{aligned} \tag{3.3.4}$$

όπου $r = \sqrt{x^2 + y^2 + z^2}$, $\theta = \tan^{-1} \left[\sqrt{x^2 + y^2} / z \right]$, $\varphi = \tan^{-1} [y / x]$.

Όπου r το μέτρο, θ η γωνία από τον άξονα z και φ η γωνία πάνω στο επίπεδο xy .
Το προσπίπτον κύμα p_i μπορεί να υπολογιστεί στο ελεύθερο πεδίο:

$$p_i(r, t) = P_0 e^{i(kr - \omega t)} \quad (3.3.5)$$

Η μαθηματική περιγραφή του επίπεδου κύματος σε σφαιρικές συντεταγμένες :

$$p_i(r, \theta, \varphi, \omega) = P_0 e^{ikr \cos \theta} \quad (3.3.6)$$

Το πρόβλημα σκέδασης περιλαμβάνει το διαχωρισμό της ολικής πίεσης σε δύο μέρη:

$$p_t = p_s + p_i \quad (3.3.7)$$

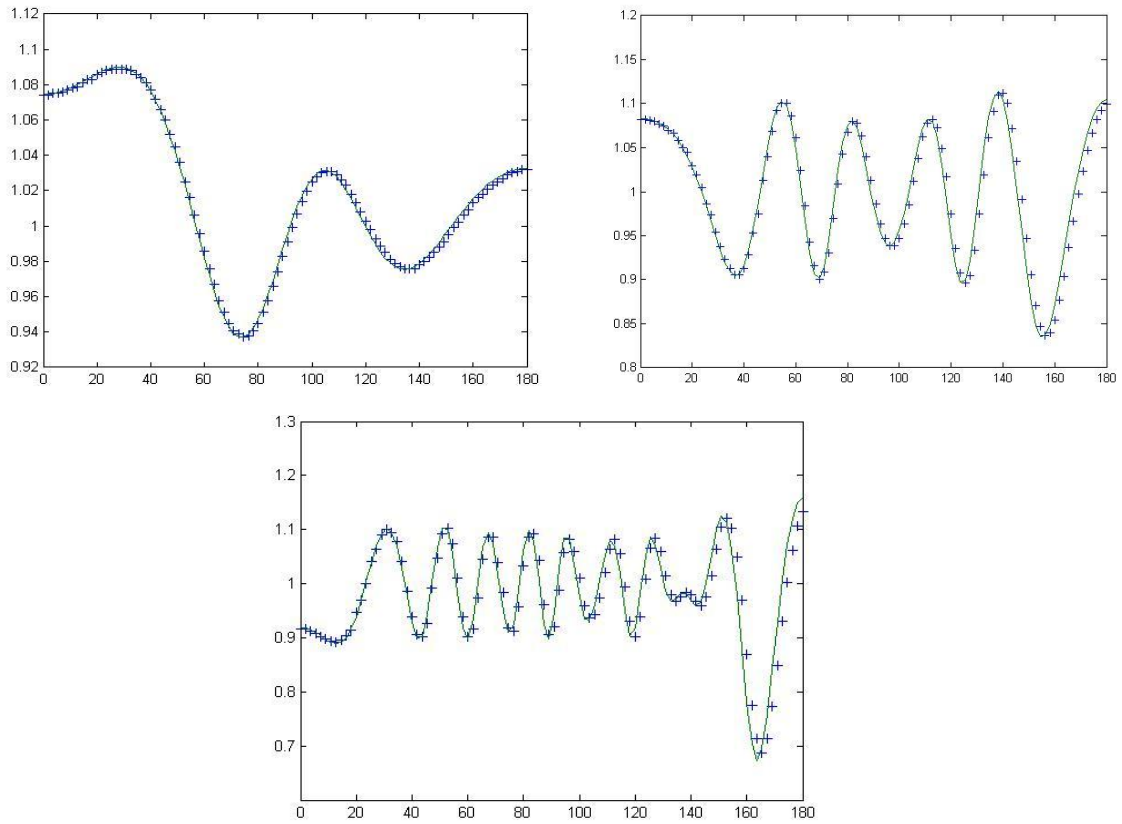
όπου p_i είναι το επίπεδο κύμα όπως περιγράφεται παραπάνω και p_s ένα καινούργιο μέγεθος που περιγράφει τις αλλαγές στο προσπίπτον κύμα όταν η σφαίρα βρεθεί στο ακουστικό πεδίο και p_t η συνολική πίεση. Για τη περίπτωση της σκληρής σφαίρας η σωματιδιακή ταχύτητα είναι μηδέν πάνω στην επιφάνεια της σφαίρας. Αυτή είναι και συνολική ακτινική ταχύτητα συμπεριλαμβανομένου του προσπίπτοντος και του σκεδαζόμενου κύματος. Η συνθήκη για τη σωματιδιακή ταχύτητα πάνω στη συνοριακή επιφάνεια θα είναι λοιπόν:

$$\dot{w}_t(\alpha, \theta) = \dot{w}_i(\alpha, \theta) + \dot{w}_s(\alpha, \theta) = 0 \quad (3.3.8)$$

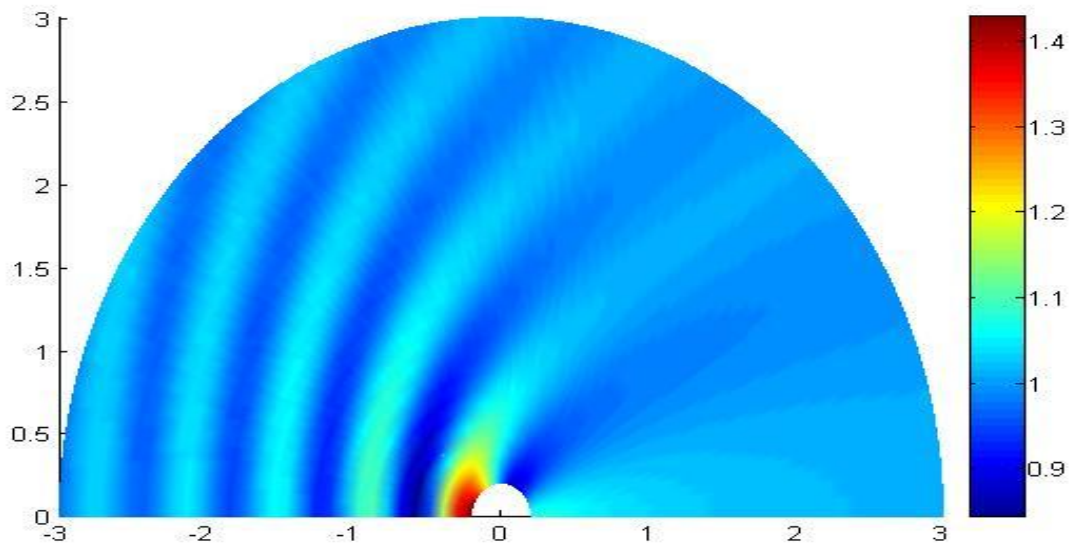
Τελικά το συνολικό πεδίο δίδεται από την σχέση :

$$p_t(r, \theta) = P_0 \sum_{n=0}^{\infty} (2n + 1) i^n \left(j_n(kr) - \frac{j'_n(ka)}{h'_n(ka)} h_n(kr) \right) P_n(\cos \theta) \quad (3.3.9)$$

Το πρόβλημα λύθηκε για διάφορες ιδιοσυχνότητες της σφαίρας και τα αποτελέσματα συγκρίθηκαν στο Mat Lab. Στο γράφημα 3.3.8 φαίνεται η σκέδαση του κύματος όπως υπολογίστηκε από την εξίσωση 3.3.9



Γράφημα 3.3.7: Πίεση στα σύνορα της σφαίρας προς γωνία (θ), για $ka=1, \pi, 6$.



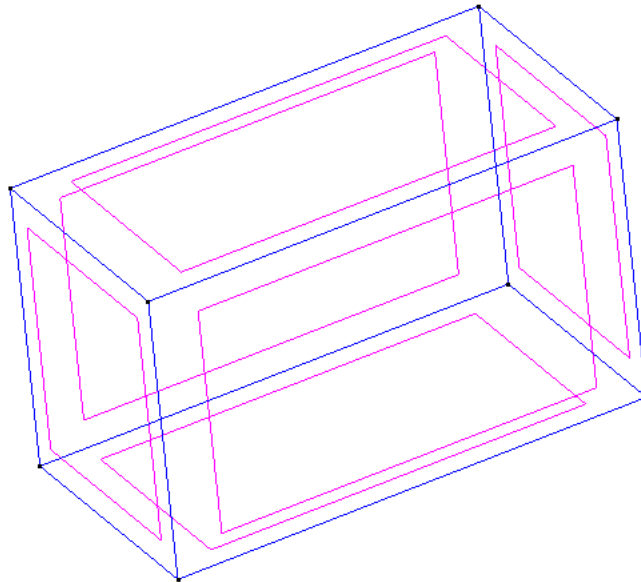
Γράφημα 3.3.8: Πίεση στο ακουστικό πεδίο για $ka = 6$

3.3.5 Ιδιορυθμοί σε δωμάτιο τριών διαστάσεων

Σε αυτό το παράδειγμα θα ασχοληθούμε πάλι με τους τρόπους δόνησης και την διέγερση στάσιμων κυμάτων, αλλά στις τρεις διαστάσεις. Οι διαστάσεις του δωματίου θα είναι 2.5 X 3 X 5.

Κατασκευή γεωμετρίας

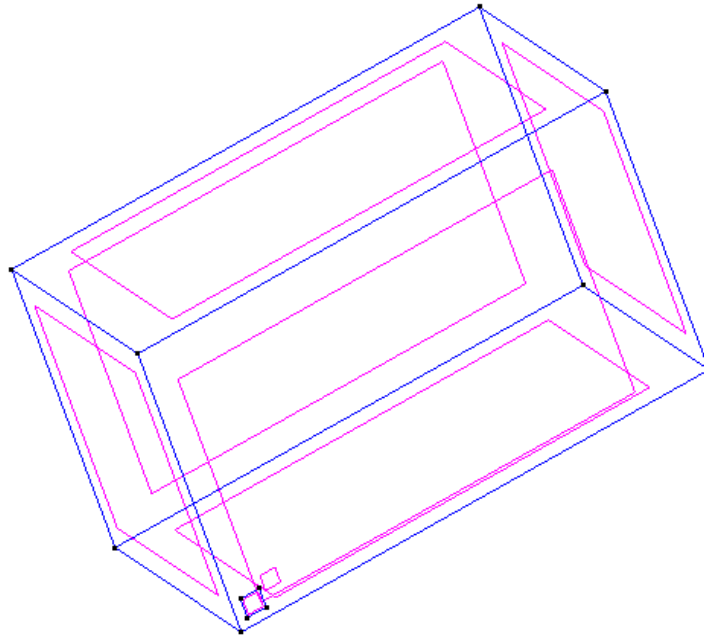
Ανοίγουμε ένα νέο 'project' στο GiD και το ονομάζουμε room.gid. Στην συνέχεια επιλέγουμε να φτιάξουμε ευθείες γραμμές με την επιλογή **Geometry > Create > Straight Line**. Πληκτρολογούμε την τιμή $\langle 0,0 \rangle$ στην γραμμή εντολών και πατάμε 'enter'. Μετά εισάγουμε με το ίδιο τρόπο την τιμή $\langle 2.5,0 \rangle$ και πάλι 'enter'. Διαδοχικά συνεχίζουμε με τις τιμές $\langle 2.5,3 \rangle$ και $\langle 3,0 \rangle$. Τέλος εισάγουμε πάλι $\langle 0,0 \rangle$ και πατάμε 'enter'. Θα ερωτηθούμε αν θέλουμε να ενώσουμε τα δύο αυτά σημεία. Με την επιλογή **join** επιλέγουμε να ενωθούν. Δημιουργούμε την επιφάνεια που ορίζουν οι πλευρές αυτές από την επιλογή **Geometry > Create > Nurbs Surface > By contour**. Με το ποντίκι επιλέγουμε όλες τις γραμμές και πατάμε **Esc**. Τώρα θα κατασκευάσουμε τις υπόλοιπες επιφάνειες του δωματίου μέσω των επιλογών αντιγραφής του GiD. Πατάμε στην επιλογή **Utilities > Copy**. Από το παράθυρο που θα εμφανιστεί επιλέγουμε στο **Entities : Surfaces** στο **Do Extrude : Surfaces** και στο **second point** στη διάσταση **z : 5**. Το αποτέλεσμα θα πρέπει να είναι σαν το παρακάτω.



Σχήμα 3.3.9 : Γεωμετρικό μοντέλο δωματίου

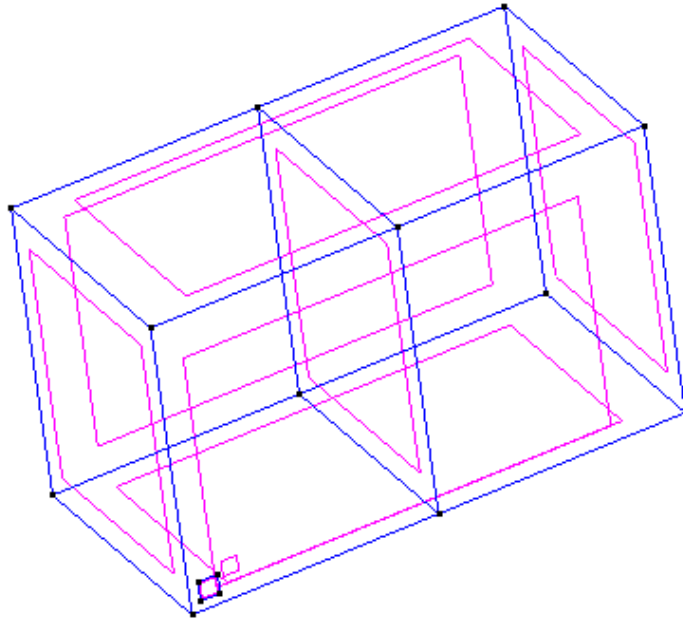
Για την κατασκευή του ηχείου θα κάνουμε ως εξής: παρόμοια με παραπάνω θα επιλέξουμε **Geometry > Create > Straight Line** και θα εισάγουμε στην γραμμή επιλογών διαδοχικά τις τιμές $\langle 0,0.1,0.1 \rangle$, $\langle 0,0.1,0.3 \rangle$, $\langle 0,0.3,0.3 \rangle$, $\langle 0,0.3,0.1 \rangle$ και πατώντας διαδοχικά 'enter' κάθε φορά. Τέλος θα εισάγουμε την πρώτη τιμή $\langle 0,0.1,0.1 \rangle$

και θα επιλέξουμε **join** όταν μας ζητηθεί. Έτσι φτιάξαμε τις γραμμές που ορίζουν το ηχείο μας. Για να δημιουργήσουμε την επιφάνεια που το ορίζει επιλέγουμε **Geometry > Create > Nurbs Surface > By Contour** επιλέγοντας και τις τέσσερις αυτές γραμμές και πατώντας **Esc**. Η πλευρά στην οποία βρίσκεται το ηχείο πρέπει τώρα να χωριστεί στα δύο με την επιλογή **Geometry > Edit > Divide > Surfaces > Split**. Επιλέγουμε την επιφάνεια που περιέχει το ηχείο και πατάμε **Esc**.



Σχήμα 3.3.10 : Μοντέλο δωματίου με ηχείο

Θέλοντας να εξετάσουμε την πίεση στο εσωτερικό του δωματίου, θα κατασκευάσουμε μία επιφάνεια η οποία θα αντιπροσωπεύει τις λύσεις αυτές. Για να θεωρηθεί ως μη συννοριακή αυτή η επιφάνεια θα πρέπει να ανήκει σε ένα ξεχωριστό επίπεδο (layer) σχεδίασης με το όνομα **field**. Πάμε στις επιλογές των επιπέδων από το **Utilities > Layers**. Από το παράθυρο που θα ανοίξει θα γράψουμε το όνομα **field** και θα πατήσουμε **New**. Αμέσως θα πρέπει να δημιουργηθεί ένα καινούριο επίπεδο και να έχει επιλεγεί ως το επίπεδο εργασίας. Τώρα θα κατασκευάσουμε την επιφάνεια κάθετα στο κέντρο του δωματίου, επιλέγοντας **Geometry > Create > Straight Line** και θα εισάγουμε στην γραμμή επιλογών διαδοχικά τις τιμές **<0.05,0.05,2.5>**, **<2.45,0.05,2.5>**, **<2.45,2.95,2.5>**, **<0.05,2.95,2.5>** και πατώντας διαδοχικά **enter** κάθε φορά. Τέλος θα εισάγουμε την πρώτη τιμή **<0.05,0.05,2.5>** και θα επιλέξουμε **join** όταν μας ζητηθεί. Παρατηρούμε πως αυτή η επιφάνεια δεν πρέπει να περιέχει σημεία πάνω στο σύνορο της γεωμετρίας, γι' αυτό και τα σημεία τοποθετήθηκαν ελάχιστα πιο μέσα από τις επιφάνειες του δωματίου, για τους λόγους που εξηγήθηκαν παραπάνω στην ενότητα 3.3. Έτσι δημιουργούμε την επιφάνεια με την επιλογή **Geometry > Create > Nurbs Surface > By Contour** και επιλέγοντας τις τέσσερις αυτές γραμμές και πατώντας **Esc**.



Σχήμα 3.3.11: Η επιπλέον επιφάνεια των λύσεων

Δήλωση συνοριακών συνθηκών

Δηλώνουμε το πρόβλημα με την επιλογή **Data > Problem Type > Helm3D**. Επιλέγουμε **Data > Conditions** και πηγαίνουμε στο **Velocity**. Συμπληρώνουμε την τιμή **<0>** στα **Normal Velocity Real** και **Imaginary**, για να δηλώσουμε μηδενική ταχύτητα στο σύνορο. Επιλέγουμε όλες τις πλευρές του project εκτός από την επιφάνεια που ανήκει στο επίπεδο field, και πατάμε **Esc**. Τώρα αλλάζουμε την τιμή **Real** σε **<1>** και επιλέγουμε μόνο την επιφάνεια που ορίζει το ηχείο, πατώντας έπειτα **Esc**.

Δήλωση λοιπόν στοιχείων του προβλήματος

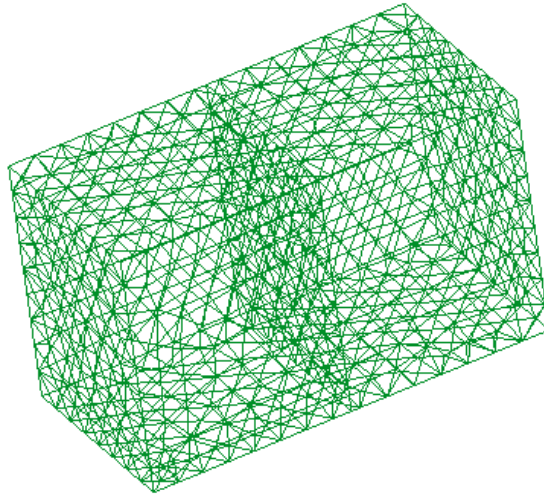
Ανοίγουμε το παράθυρο **Data > Problem data** και συμπληρώνουμε τα παρακάτω

Analysis type: Internal
Loading type: Radiation
Symmetry: None

Στο παράθυρο **Fluid Properties** αφήνουμε τις προεπιλεγμένες τιμές και στο παράθυρο **Frequencies** δηλώνουμε την τιμή **<68.5>** για τα **Frequency start** και **Frequency stop**. Για πάρουμε τις λύσεις που θέλουμε πάνω στην επιπλέον επιφάνεια που σχεδιάσαμε θα επιλέξουμε από το παράθυρο **Output points: Field point type: Mesh**. Πατάμε **Accept** και σώζουμε τα δεδομένα.

Εξαγωγή δικτύωσης και υπολογισμός προβλήματος

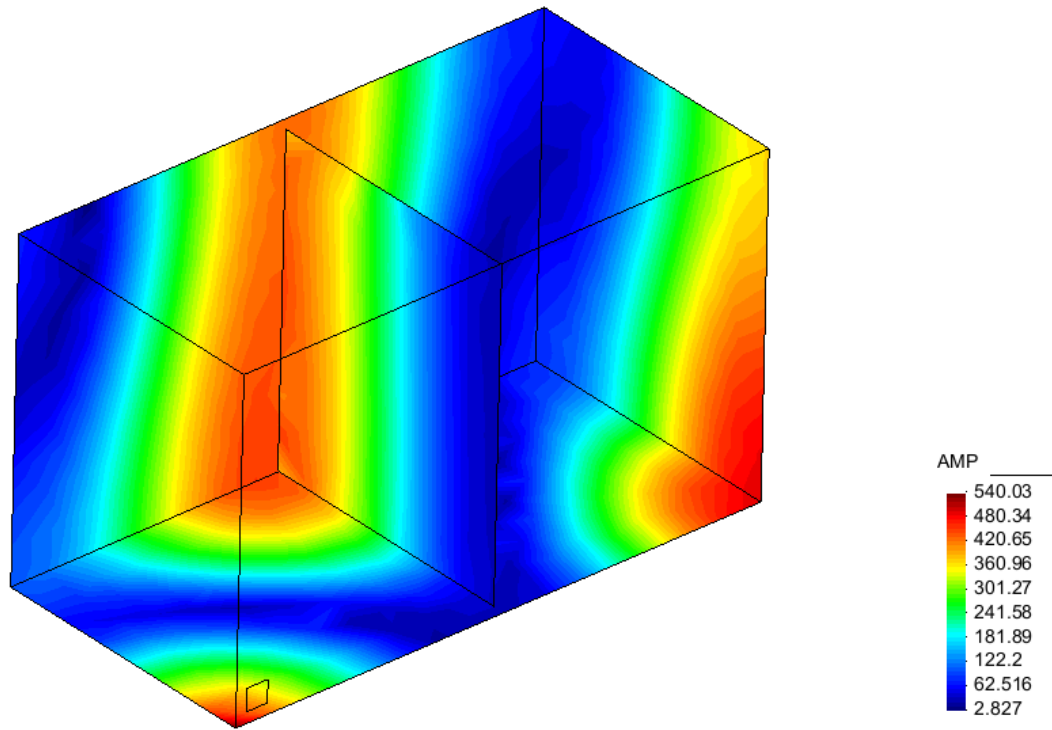
Θα δικτυώσουμε τα σύνορα και την επιφάνεια των λύσεων από το **Mesh > Generate Mesh** και εισάγοντας την τιμή **<0.35>** στο παράθυρο που θα μας ζητά το μέγεθος των στοιχείων. Πατάμε **Ok**. Οδηγούμε το πρόβλημα σε λύση από το **Calculate > calculate**.



Σχήμα 3.3.12: Δικτύωμα του δωματίου

Επισκόπηση αποτελεσμάτων

Αφού ολοκληρωθεί η επίλυση μπορούμε να μεταβούμε στον μετα-επεξεργαστή με το **Files > Postprocess**. Εκεί επιλέγοντας στο **View Results > Contour Fill > AMP** θα δούμε την πίεση στα σύνορα. Τώρα για να δούμε την πίεση και στην επιφάνεια που είναι στο εσωτερικό θα χρησιμοποιήσουμε τις επιλογές που βρίσκονται στο **Utilities > View Style**. Στο παράθυρο που θα ανοίξει θα διαλέξουμε **Front Faces** από την επιλογή **Culling**, και έτσι μπορούμε να δούμε στο εσωτερικό του χώρου και την επιφάνεια που είναι σχεδιασμένη μέσα του, όπως φαίνεται παρακάτω.



Σχήμα 3.3.12: Πίεση στα σύνορα και στο εσωτερικό του δωματίου

Αναφορές - Βιβλιογραφία

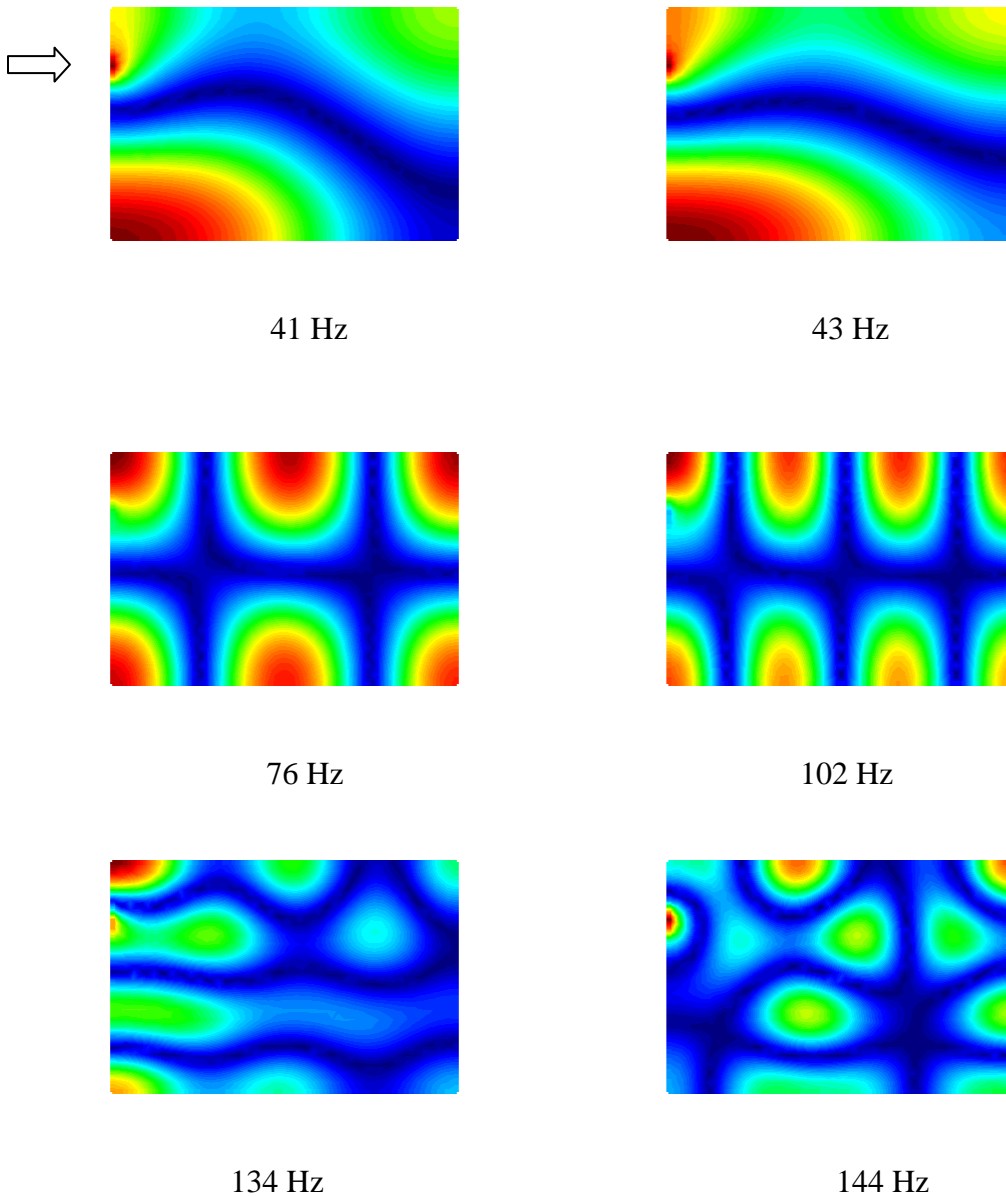
- [1] T.W. Wu. *Boundary Element Acoustics Fundamentals and Computer Codes*, WIT Press Southampton, Boston, 2000.
- [2] Stephen Kirkup, *The Boundary Element Method in Acoustics A development in Fortran*, Integrated Sound Software, West Yorkshire, 1998.
- [3] Earl G. Williams, *Fourier Acoustics - Sound Radiation and Nearfield Acoustical Holography*, Academic Press, Cambridge 1999.
- [4] C.J. Wu, X.J. Wang, H.B. Tang (2006) “Transmission loss prediction on a single-inlet/double-outlet cylindrical expansion-chamber muffler by using the modal meshing approach”, *Applied Acoustics* Vol. 69 pp. 173–178, (2008).
- [5] L. Shen, Y. J. Liu (2006) “An adaptive fast multipole boundary element method for three-dimensional acoustic wave problems based on the Burton-Miller formulation”, *Comput. Mech.* (2007) 40:461–472.
- [6] Ιωάννης Θ. Κατσικαδέλης, *Συνοριακά στοιχεία στην επιστήμη του μηχανικού*, Εκδόσεις Συμεών, Αθήνα 1999.
- [7] Laura A. Brooks, Rick C. Morgans, Colin H. Hansen (2005) “Learning Acoustics through the Boundary Element Method: An Inexpensive Graphical Interface and Associated Tutorials”, *Proceedings of ACOUSTICS 2005*, 9-11 November 2005, Busselton, Western Australia.
- [8] D. W. Herrin, T. W. Wu, and A. F. Seybert, ‘Boundary Element Modeling’, *Handbook of Noise and Vibration*, Chapter 8, Edited by Malcolm J. Crocker, Copyright © 2007 John Wiley & Sons, Inc.

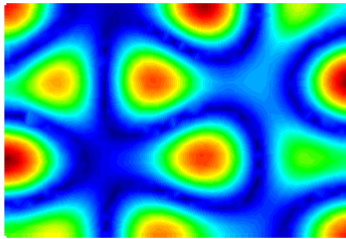
Παράρτημα Α

Εδώ παρατίθενται επιπλέον λύσεις και γραφήματα των εφαρμογών που λύθηκαν στο κεφάλαιο 3, για την καλύτερη κατανόηση των προβλημάτων.

Λύσεις και αποτελέσματα από την παράγραφο 3.2.3 – 3.2.4

Για το πρόβλημα του δισδιάστατου δωματίου με μία πηγή φαίνονται παρακάτω στάσιμα κύματα για τις αντίστοιχες συχνότητες που αναγράφονται κάτω από κάθε εικόνα. Η πηγή είναι τοποθετημένη εκεί όπου δείχνει το βέλος στην πρώτη εικόνα.

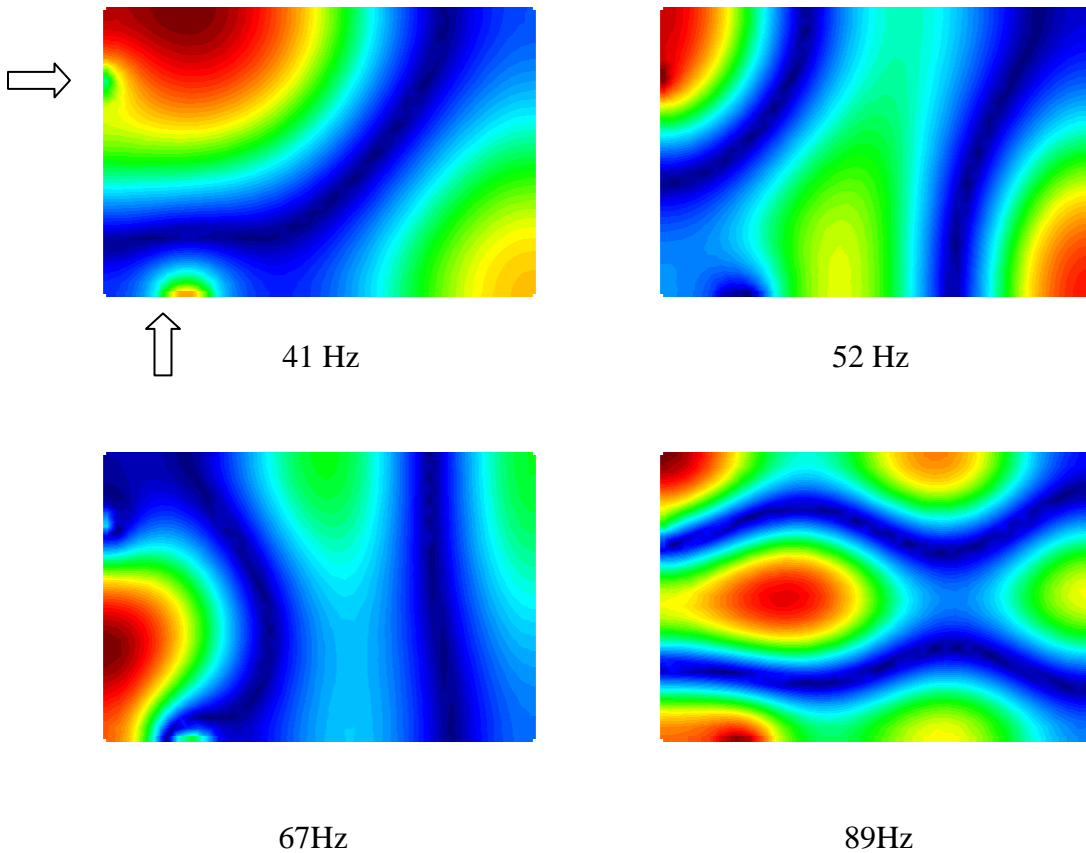


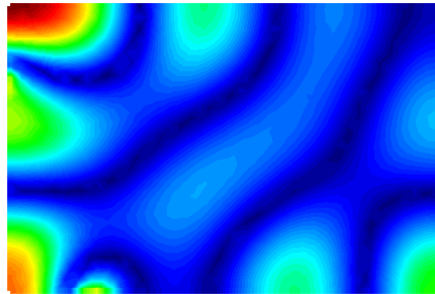


149 Hz

Αποτελέσματα και λύσεις για την παράγραφο 3.2.5

Για το πρόβλημα του δισδιάστατου δωματίου με δύο πηγές φαίνονται παρακάτω στάσιμα κύματα για τις αντίστοιχες συχνότητες που αναγράφονται κάτω από κάθε εικόνα. Οι δύο πηγές είναι τοποθετημένες εκεί όπου δείχνουν τα βέλη στην πρώτη εικόνα.

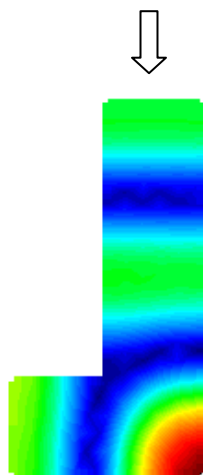




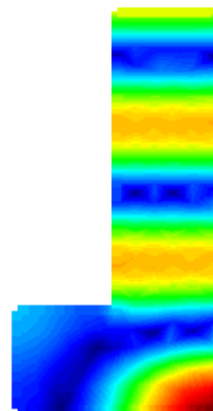
112 Hz

Αποτελέσματα και λύσεις για την παράγραφο 3.2.6

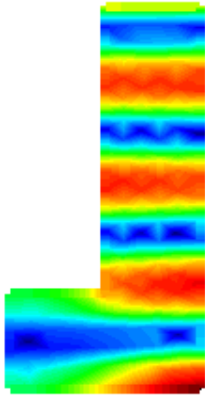
Για το πρόβλημα του δισδιάστατου δωματίου σε σχήμα 'Γ' φαίνονται παρακάτω στάσιμα κύματα για τις αντίστοιχες συχνότητες που αναγράφονται κάτω από κάθε εικόνα. Το βέλος δείχνει την πηγή.



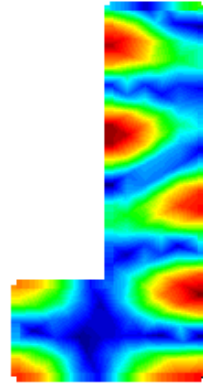
40Hz



50Hz



70Hz



80Hz

Παράρτημα Β

Εδώ παρατίθενται όλες οι ρουτίνες που είναι απαραίτητες και πρέπει να αλλάξουν για να δημιουργηθεί το εκτελέσιμο πρόγραμμα Helm2D.exe που αποτελεί το πρόγραμμα επίλυσης της εφαρμογής που περιγράφεται στο κεφάλαιο 2.

Αφού αντικατασταθούν από τις αρχικές ρουτίνες πρέπει να γίνουν compile με κάποιο compiler της FORTRAN 77, μαζί με τις υπόλοιπες ρουτίνες που αποτελούν το σύνολο της εφαρμογής.

Ρουτίνα preprocess του Helm2d Prep.for

```
      SUBROUTINE PREP
C
C----- PREPROCESSOR
C
      COMMON /CHI/NCH,XCH(20),YCH(20)
      COMMON /FPOINTS/NF,XF(1000),YF(1000),FS(1000)
      CHARACTER*60 TITLE
      CHARACTER*4 BLOCK
      READ(5,100) TITLE
      WRITE(6,100) TITLE

      CALL SETINT
      NCH=0
      NF=0

10    READ(5,200,END=99) BLOCK
      IF (BLOCK.EQ. 'GENE') CALL RGENE
      IF (BLOCK.EQ. 'NODE') CALL RNODEN
      IF (BLOCK.EQ. 'ELEM') CALL RELEMN
      IF (BLOCK.EQ. 'BOUN') CALL BC
      IF (BLOCK.EQ. 'FIEL') CALL RFIELD
      IF (BLOCK.EQ. 'CHIE') CALL RCHIEF
      IF (BLOCK.EQ. 'END ') GO TO 99
      IF (BLOCK.EQ. 'SOLV') RETURN
      GO TO 10
99    STOP
100   FORMAT(A60)
200   FORMAT(A4)
      END
```

Ρουτίνα ανάγνωσης κορυφών RNode.for

```
      SUBROUTINE RNODE
C
C----- READ IN NODAL POINT DATA           η ρουτίνα αυτή διαβάζει την πληροφορία
C                                           κορυφών
C
      COMMON /NAMES/RNN(2000),RFN(1000),ERN(1000)
      COMMON /CONTR/NNODE,NELEM
      COMMON /NODES/X(2000),Y(2000)
      READ(5,*) NNODE
      DO 30 NR=1,NNODE
      READ(5,*) RNN(NR),X(NR),Y(NR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NNODE =NUMBER OF NODES
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30    CONTINUE
C
C----- PRINT OUT NODAL POINT COORDINATES
C
      WRITE(6,100)
```

```

100  FORMAT (/2X,'NODAL POINT COORDINATES:')
DO 102 I=1,NNODE
    WRITE(6,101) I,X(I),Y(I)
101  FORMAT(5X,'NODE #',I4,5X,'X=',F11.3,3X,'Y=',F11.3)
102  CONTINUE
    RETURN
    END

```

Ρουτίνα ανάγνωσης στοιχείων RElem.for

```

SUBROUTINE RELEM
C
C---- READ IN ELEMENT CONNECTIVITY          ρουτίνα που σώζει την συνδεσιμότητα των
C                                           στοιχείων
C
    COMMON /PROB/INT,ISC
    COMMON /NAMES/RNN(2000),RFN(1000)
    COMMON /NODES/X(2000),Y(2000)
    COMMON /CONTR/NNODE,NELEM
    COMMON /ELEM/KIND(2000),NODE(4,2000)
    READ(5,*) NELEM,KINDI
    NE=1
C
    NUMBER=0
    DO 50 NR=1,NELEM
        READ(5,*) N,(NODE(J,NR),J=1,KINDI+1)
CCCCCCCCC-----THIS PART ADDED BY LEONIDAS-----CCCCCCCCC
        IF(INT.EQ.1) THEN
C-----LOOP FOR NODES-----
            DO 30 J=1,KINDI+1
                DO 20 I=1,NELEM
                    IF (NODE(J,NR).EQ.RNN(I)) THEN
                        NODE(J,NR)=I
                        GOTO 30
                    ENDIF
                CONTINUE
            20 CONTINUE
            30 CONTINUE
            ENDIF
CCCCCCCCC-----CCCCCCCCC
CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC
C    NELEM =NUMBER OF ELEMENTS C
C    KINDI =1    LINEAR ELEMENTS C
C           =2    QUADRATIC ELEMENTS C
C           =3    CUBIC ELEMENTS C
CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC-----CCCCCCCCC
        IF(KINDI.LT.1.OR.KINDI.GT.3) GO TO 99
        DO 40 I=NE,NE+NELEM-1
            KIND(I)=KINDI
    40 CONTINUE
    50 CONTINUE
C
C    PRINT OUT ELEMENT CONNECTIVITY
C
    WRITE(6,55)
    55  FORMAT (/2X,'ELEMENT CONNECTIVITY :'/)
    DO 60 I=1,NELEM
C
        KINDI=KIND(I)
        WRITE(6,200) I,KIND(I),(NODE(J,I),J=1,KINDI+1)
    200  FORMAT(5X,'ELEMENT #',I3,3X,'DEGREE=',I3,3X,
&      'NODES=',4I3)
    60  CONTINUE
    RETURN
    99  WRITE(6,*) '----ERROR IN ELEMENT DATA----'
        STOP
    END

```

Ρουτίνα ανάγνωσης σημείων επίλυσης στο ακουστικό πεδίο RField.for

```

SUBROUTINE RFIELD
C
C---- READ IN FIELD POINTS   ρουτίνα που σώζει σημεία επίλυσης στο ακουστικό πεδίο
C
COMMON /NAMES/RNN(2000),RFN(1000),ERN(1000)
COMMON /FPOINTS/NF,XF(1000),YF(1000),FS(1000)
READ(5,*) NF
DO 30 I=1,NF
  READ(5,*) RFN(I),XF(I),YF(I)
30 CONTINUE

RETURN
END

```

Ρουτίνα εξόδου αποτελεσμάτων Results.for

```

SUBROUTINE RESULTS
C
C PROGRAM FOR GID
C
C
C
COMMON /NAMES/RNN(2000),RFN(1000),ERN(1000)
COMMON /FPOINTS/NF,XF(1000),YF(1000),FS(1000)
COMMON /FRE/FREQ,WN,IFR
COMMON /CONTR/NNODE,NELEM
COMMON /SOL/SP(2000)
COMMON /NODES/X(2000),Y(2000)
COMMON /ELEM/KIND(2000),NODE(4,2000)
INTEGER YFN,XFN,RNN,RFN
COMPLEX SP,FS

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C-----OUTPUT RESULTS FILE-----C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C-----Print out pressure real on Nodes-----
C
WRITE(7,20) FREQ
20 FORMAT('Result "Pres_real" "frequency" ',F8.3,' Scalar OnNodes',//,'Values')

DO 40 I=1,NNODE
WRITE(7,30) RNN(I),REAL(SP(I))
30 FORMAT(2X,F6.0,TL1," ",2X,F10.3)
40 CONTINUE

C-----Print out pressure real at field points-----
C
DO 45 I=1,NF
WRITE(7,46) RFN(I),REAL(FS(I))
46 FORMAT(2X,F6.0,TL1," ",4X,F10.3)
45 CONTINUE
WRITE(7,50)
50 FORMAT(1X,'End Values'//)

C
C-----Print out pressure imaginary on Nodes-----
C
WRITE(7,60) FREQ
60 FORMAT('Result "Pres_imag" "frequency" ',F8.3,' Scalar OnNodes',//,'Values')
DO 80 I=1,NNODE
WRITE(7,70) RNN(I),AIMAG(SP(I))
70 FORMAT(2X,F6.0,TL1," ",2X,F10.3)
80 CONTINUE

C-----Print out pressure imaginary AT FIELD POINTS-----
DO 85 I=1,NF

```

```

      WRITE (7,86)      RFN(I),AIMAG(FS(I))
86      FORMAT(2X,F6.0,TL1," ",4X,F10.3)
85      CONTINUE

      WRITE (7,90)
90      FORMAT(1X,'End Values'/)
C
C-----Print out amplitude on Nodes-----
C
      WRITE (7,100)  FREQ
100     FORMAT('Result "AMP" "frequency" ',F8.3,' Scalar OnNodes'/,'Values')
      DO 120 I=1,NNODE
      WRITE (7,110)  RNN(I),CABS(SP(I))
110     FORMAT(2X,F6.0,TL1," ",2X,F10.3)
120     CONTINUE
C-----Print out PRESSURE AMPLITUDE AT FIELD POINTS-----
      DO 125 I=1,NF
      WRITE (7,126)  RFN(I),CABS(FS(I))
126     FORMAT(2X,F6.0,TL1," ",4X,F10.3)
125     CONTINUE

      WRITE (7,130)
130     FORMAT(1X,'End Values'/)
C
C-----Print out DECIBEL on Nodes-----
C
      WRITE (7,140)  FREQ
140     FORMAT('Result "db SPL" "frequency" ',F8.3,' Scalar OnNodes'/,'Values')
      DO 160 I=1,NNODE
      WRITE (7,150)  RNN(I),20*LOG10(CABS(SP(I))/0.00002)
150     FORMAT(2X,F6.0,TL1," ",2X,F10.3)
160     CONTINUE
C-----Print out DECIBELS AT FIELD POINTS-----
      DO 165 I=1,NF
      WRITE (7,166)  RFN(I),20*LOG10(CABS(FS(I))/0.00002)
166     FORMAT(2X,F6.0,TL1," ",4X,F10.3)
165     CONTINUE

      WRITE (7,170)
170     FORMAT(1X,'End Values'/)

      RETURN
      END

```

Παράρτημα Γ

Εδώ παρατίθενται τα αρχεία διαμόρφωσης της εφαρμογής Helm2D.gid. Τα αρχεία αυτά είναι αναγκαία για να λειτουργεί η εφαρμογή (Helm2D.exe) μέσω του προγράμματος GiD. Τελικά θα πρέπει να δημιουργηθεί ένας φάκελος με όνομα Helm2D.gid που θα περιέχει τα παρακάτω αρχεία καθώς και το εκτελέσιμο αρχείο (Helm2D.exe). Αυτός ο φάκελος πρέπει να τοποθετηθεί στον υποφάκελο 'Problem Types' που βρίσκεται στον φάκελο εγκατάστασης του GiD.

Αρχείο `template (Helm2D.bas)` (συλλογή στοιχείων προβλήματος από το GiD)

```

*GenData(Project_title)
GENERAL INFORMATION
*#
*# Analysis type
*#
*if(strcmp(GenData(Analysis_type),"Internal")==0)
1*\

```

```

*elseif(strcmp(GenData(Analysis_type),"External")==0)
0*\
*else
*MessageBox ERROR : Unknown analysis type.
*endif
*#
*# Scattering (0) or radiation (1)
*#
*if(strcmp(GenData>Loading_type),"Scattering")==0)
,1
*elseif(strcmp(GenData>Loading_type),"Radiation")==0)
,0
*else
*MessageBox ERROR : Unknown loading type.
*endif
*GenData(Density,real),*GenData(Speed_of_sound,real)
*GenData(Frequency_start,real),*GenData(Frequency_stop,real),*GenData(Frequency_interval,
real)
*#
NODES
*#
*Set elems(linear)
*set var NNODES(int)=0
*loop elems
*set var NNODES=LoopVar
*end elems
*NNODES
*Set elems(linear)
*loop elems
*format "%8i,%10.3g,%10.3g"
*elemsConec(1)*NodesCoord(1,real)
*end nodes
*#
ELEMENTS
*#
*set elems(linear)
*nelem,*GenData(kind)
*Set elems(linear)
*loop elems
*format "%8i,%8i,%8i"
*LoopVar*elemsConec(1)*elemsConec(2)
*end elems
*#
*# Calculate total number of BCs
*#
*#*Set var NBC=NNODES
*Set var ZERO=0.0
*Set var ONE=1.0
*Set var NEGONE=-1.0
*set elems(all)
*set Cond Pressure *elems
*add Cond Velocity *elems
*add Cond Impedance *elems
*set var NBC=CondNumEntities(int)
*#*Set Cond Pressure *elems
*#*Set var NBC=NBC+CondNumEntities(int)
*#*set elems(all)
*#*Set Cond Velocity *elems
*#*Set var NBC=NBC+CondNumEntities(int)
*#*set elems(all)
*#*Set Cond Impedance *elems
*#*Set var NBC=NBC+CondNumEntities(int)
*if(NBC==0)
*MessageBox ERROR : No boundary conditions defined.
*endif
BOUNDARY CONDITIONS
*operation(NNODES+NBC)
*#
*# Write out default boundary condition, V=0
*#
*set elems(linear)

```

```

*loop elems
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*LoopVar*LoopVar*ZERO*ZERO*ZERO*ONE*ZERO*ZERO*ZERO
*end elems
*#
*# Write out Pressure boundary condition
*#
*set elems(all)
*Set Cond Pressure *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInCond
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*LoopVar*LoopVar*ZERO*ONE*ZERO*ZERO*ZERO*cond(Pressure_Re,real)*cond(Pressure_Im,real)
*end elems
*endif
*set elems(all)
*Set Cond Velocity *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInCond
*if(ElemsNnode(int)==2)
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*LoopVar*LoopVar*ZERO*ZERO*ZERO*NEGONE*ZERO*cond(Normal_Velocity_Re,real)*cond(Normal_Vel
ocity_Im,real)
*endif
*end elems
*endif
*set elems(all)
*Set Cond Impedance *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInCond
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*LoopVar*LoopVar*ZERO*ZERO*NEGONE*ZERO*cond(Normal_Impedance_Re,real)*cond(Normal_Impedance_Im
,real)*ZERO*ZERO
*end elems
*endif
*#
*# Field points
*#
*if(strcmp(GenData(Field_point_type),"Straight")==0)
*# Field points are on a straight line
*if(GenData(Number_of_line_field_points,int)>0)
*if(GenData(Number_of_line_field_points,int)<200)
*Set var NFLDPT=GenData(Number_of_line_field_points,int)
*else
*Set var NFLDPT=100
*WarningBox WARNING: Only first 100 field points will be used
*endif
FIELD POINTS
*NFLDPT
*Set var XF=GenData(Start_point_x,real)
*Set var YF=GenData(Start_point_y,real)
*for (IField=1;IField<=NFLDPT;IField=IField+1)
*format "%10.3g,%10.3g,%10.3g"
*XF*YF
*Set var XF=XF+GenData(X_Step,real)
*Set var YF=YF+GenData(Y_Step,real)
*endfor
*endif
*#
*elseif(strcmp(GenData(Field_point_type),"Arc")==0)
*# Field points are on an arc
*#
*if(GenData(Number_of_arc_field_points,int)>0)
*if(GenData(Number_of_arc_field_points,int)<200)
*Set var NFLDPT=GenData(Number_of_arc_field_points,int)
*else
*Set var NFLDPT=100
*WarningBox WARNING: Only first 100 field points will be used
*endif
FIELD POINTS
*NFLDPT

```

```

*Set var PI=3.141592654
*Set var THETA=GenData(Start_angle,real)*PI/180.0
*Set var DTHETA=GenData(Angle_step,real)*PI/180.0
*Set var X0=GenData(Arc_centre_X,real)
*Set var Y0=GenData(Arc_centre_Y,real)
*for (IField=1;IField<=NFLDPT;IField=IField+1)
*Set var U=GenData(Arc_radius,real)*cos(THETA)
*Set var V=GenData(Arc_radius,real)*sin(THETA)
*if(strcmp(GenData(Axis_direction),"X")==0)
*format "%10.3g,%10.3g,%10.3g"
*X0*operation(Y0+U)*operation(Z0+V)
*elseif(strcmp(GenData(Axis_direction),"Y")==0)
*format "%10.3g,%10.3g,%10.3g"
*operation(X0+V)*Y0*operation(Z0+U)
*elseif(strcmp(GenData(Axis_direction),"Z")==0)
*format "%10.3g,%10.3g,%10.3g"
*operation(X0+U)*operation(Y0+V)*Z0
*endif
*Set var THETA=THETA+DTHETA
*endfor
*endif
*#
*elseif(strcmp(GenData(Field_point_type),"Mesh")==0)
*# Field points from mesh conditions
*#if(GenData(Number_of_field_points_on_x,int)>0)
*#if(GenData(Number_of_field_points_on_x,int)<100)
*#if(GenData(Number_of_field_points_on_y,int)>0)
*#if(GenData(Number_of_field_points_on_y,int)<100)
*#Set var NFLDPTx=GenData(Number_of_field_points_on_x,int)
*#Set var NFLDPTy=GenData(Number_of_field_points_on_y,int)
*#Set var NFLDPT=NFLDPTx*NFLDPTy
*#else
*#Set var NFLDPT=2500
*#WarningBox WARNING: Only the first 50 field points on x axis and the first 50 field
points on y axis will be used
*endif
FIELD POINTS
*Set var NFLDPT=2500
*set layer field *nodes
*loop nodes *OnlyInLayer
*Set var NFLDPT=loopvar
*end nodes
*NFLDPT
*set layer field *nodes
*loop nodes *OnlyInLayer
*format "%8i,%10.3g,%10.3g"
*nodesnum *NodesCoord(1,real) *NodesCoord(2,real)
*end nodes
*#Set var YF=GenData(Start_point_y,real)
*#for (IFieldy=1;IFieldy<=NFLDPTy;IFieldy=IFieldy+1)
*#Set var XF=GenData(Start_point_x,real)
*#for (IFieldx=1;IFieldx<=NFLDPTx;IFieldx=IFieldx+1)
*#format "%10.3g,%10.3g"
*#XF*YF
*#Set var XF=XF+GenData(X_Step,real)
*#endfor
*#Set var YF=YF+GenData(Y_Step,real)
*#endfor
*#endif
*#
*#endif
*endif
*# Chief points - modded A Duncan 3/1/06 to allow up to 5 Chief points
*#
*if(GenData(Number_of_Chief_points,int)>0)
CHIEF POINTS
*GenData(Number_of_Chief_points,int)
*format "%10.3g,%10.3g,%10.3g"
*GenData(Chief_point_x1,real)*GenData(Chief_point_y1,real)
*endif
*if(GenData(Number_of_Chief_points,int)>1)

```

```

*format "%10.3g,%10.3g,%10.3g"
*GenData(Chief_point_x2,real)*GenData(Chief_point_y2,real)
*endif
*if(GenData(Number_of_Chief_points,int)>2)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x3,real)*GenData(Chief_point_y3,real)
*endif
*if(GenData(Number_of_Chief_points,int)>3)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x4,real)*GenData(Chief_point_y4,real)
*endif
*if(GenData(Number_of_Chief_points,int)>4)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x5,real)*GenData(Chief_point_y5,real)
*endif
*if(GenData(Number_of_Chief_points,int)>5)
*WarningBox WARNING: Only a maximum of 5 Chief points will be used
*endif
SOLVE

```

Αρχείο εισαγωγής συνοριακών συνθηκών (Helm2D.cnd)

```

NUMBER: 1 CONDITION: Pressure
CONDTYPE: over lines
CONDMESHTYPE: over elements
QUESTION: Pressure_Re
VALUE: 0
QUESTION: Pressure_Im
VALUE: 0
END CONDITION
NUMBER: 2 CONDITION: Velocity
CONDTYPE: over lines
CONDMESHTYPE: over elements
QUESTION: Normal_Velocity_Re
VALUE: 1
QUESTION: Normal_Velocity_Im
VALUE: 0
END CONDITION
NUMBER: 3 CONDITION: Impedance
CONDTYPE: over lines
CONDMESHTYPE: over elements
QUESTION: Normal_Impedance_Re
VALUE: 415
QUESTION: Normal_Impedance_Im
VALUE: 0
END CONDITION

```

Αρχείο εισαγωγής δεδομένων προβλήματος (Helm2D.prb)

```

PROBLEM DATA

TITLE: General
QUESTION: Project_TITLE:
VALUE: Title_name
HELP: A descriptive title for the current project
QUESTION: Analysis_type:#CB#(Internal,External)
VALUE: Internal
HELP: Is the analysis of the interior or exterior of a closed volume?
QUESTION: Loading_type:#CB#(Radiation,Scattering)
VALUE: Radiation
DEPENDENCIES:
(Radiation,HIDE,Planewave_Amplitude,#CURRENT#,HIDE,Planewave_Lx,#CURRENT#,HIDE,Planewave_
Ly,#CURRENT#,HIDE,Planewave_Lz,#CURRENT#)
DEPENDENCIES:
(Scattering,RESTORE,Planewave_Amplitude,#CURRENT#,RESTORE,Planewave_Lx,#CURRENT#,RESTORE,
Planewave_Ly,#CURRENT#,RESTORE,Planewave_Lz,#CURRENT#)
HELP: Is it a radiation or scattering (incident plane wave) problem?

```


QUESTION: Planewave_amplitude:
 VALUE: 1
 HELP: Pressure amplitude of incident plane wave
 QUESTION: Planewave_Lx:
 VALUE: 0
 HELP: X direction cosine of plane wave wavevector
 QUESTION: Planewave_Ly:
 VALUE: 0
 HELP: Y direction cosine of plane wave wavevector
 QUESTION: Planewave_Lz:
 VALUE: -1
 HELP: Z direction cosine of plane wave wavevector
 QUESTION: Symmetry:#CB#(None,Y-Z,X-Z,X-Y)
 VALUE: None
 HELP: On which plane is the geometric symmetric about?
 QUESTION: kind:#CB#(1,2,3)
 VALUE: 2
 HELP: 1 for linear 2 for quadratic 3 for cubic elems

TITLE: Fluid properties
 QUESTION: Density:
 VALUE: 1.21
 HELP: Density of the fluid
 QUESTION: Speed_of_sound:
 VALUE: 343
 HELP: Speed of sound in the fluid
 QUESTION: Reference_pressure:
 VALUE: 20e-6
 HELP: Reference pressure for SPL calculations

TITLE: Frequencies
 QUESTION: Frequency_start:
 VALUE: 100
 HELP: Starting frequency
 QUESTION: Frequency_stop:
 VALUE: 100
 HELP: Stopping frequency
 QUESTION: Frequency_interval:
 VALUE: 0
 HELP: Frequency increment

TITLE: Output points
 QUESTION: Output_node:
 VALUE: 1
 HELP: Node number of the node data written to the output file
 QUESTION: Field_point_type:#CB#(Straight,Arc,Mesh)
 VALUE: Straight
 DEPENDENCIES:
 (Straight,RESTORE,Number_of_line_field_points,#CURRENT#,RESTORE,Start_point_x,#CURRENT#,RESTORE,Start_point_y,#CURRENT#,RESTORE,X_step,#CURRENT#,RESTORE,Y_step,#CURRENT#,HIDE,Number_of_arc_field_points,#CURRENT#,HIDE,Arc_centre_X,#CURRENT#,HIDE,Arc_centre_Y,#CURRENT#,HIDE,Axis_direction,#CURRENT#,HIDE,Arc_radius,#CURRENT#,HIDE,Start_angle,#CURRENT#,HIDE,Angle_step,#CURRENT#,HIDE,Number_of_field_points_on_x,#CURRENT#,HIDE,Number_of_field_points_on_y,#CURRENT#)
 DEPENDENCIES:
 (Arc,HIDE,Number_of_line_field_points,#CURRENT#,HIDE,Start_point_x,#CURRENT#,HIDE,Start_point_y,#CURRENT#,HIDE,X_step,#CURRENT#,HIDE,Y_step,#CURRENT#,RESTORE,Number_of_arc_field_points,#CURRENT#,RESTORE,Arc_centre_X,#CURRENT#,RESTORE,Arc_centre_Y,#CURRENT#,RESTORE,Axis_direction,#CURRENT#,RESTORE,Arc_radius,#CURRENT#,RESTORE,Start_angle,#CURRENT#,RESTORE,Angle_step,#CURRENT#,HIDE,Number_of_field_points_on_x,#CURRENT#,HIDE,Number_of_field_points_on_y,#CURRENT#)
 DEPENDENCIES:
 (Mesh,RESTORE,Number_of_field_points_on_x,#CURRENT#,RESTORE,Number_of_field_points_on_y,#CURRENT#,HIDE,Number_of_line_field_points,#CURRENT#,RESTORE,Start_point_x,#CURRENT#,RESTORE,Start_point_y,#CURRENT#,RESTORE,X_step,#CURRENT#,RESTORE,Y_step,#CURRENT#,HIDE,Number_of_arc_field_points,#CURRENT#,HIDE,Arc_centre_X,#CURRENT#,HIDE,Arc_centre_Y,#CURRENT#,HIDE,Axis_direction,#CURRENT#,HIDE,Arc_radius,#CURRENT#,HIDE,Start_angle,#CURRENT#,HIDE,Angle_step,#CURRENT#)
 QUESTION: Number_of_field_points_on_x
 VALUE: 1
 HELP: Number of field points on x axis (max 100)

QUESTION: Number_of_field_points_on_y
 VALUE: 1
 HELP: Number of field points on y axis (max 100)
 QUESTION: Number_of_line_field_points
 VALUE: 1
 HELP: Number of field points on line (max 100)
 QUESTION: Start_point_x:
 VALUE: 0
 HELP: X coordinate of first field point
 QUESTION: Start_point_y:
 VALUE: 0
 HELP: Y coordinate of first field point
 QUESTION: X_step:
 VALUE: 0.1
 HELP: X increment for consecutive field points
 QUESTION: Y_Step:
 VALUE: 0.1
 HELP: Y increment for consecutive field points
 QUESTION: Number_of_arc_field_points
 VALUE: 1
 HELP: Number of field points on arc (max 100)
 QUESTION: Arc_centre_X:
 VALUE: 0
 QUESTION: Arc_centre_Y:
 VALUE: 0
 QUESTION: Axis_direction:#CB#(X,Y)
 VALUE: Y
 HELP: Direction of arc axis in space
 QUESTION: Arc_radius:
 Value: 1
 QUESTION: Start_angle:
 Value: 0
 HELP: Start angle (deg)
 QUESTION: Angle_step
 Value: 10
 HELP: Angle increment (deg)

TITLE: Chief_Points
 QUESTION: Number_of_Chief_points:
 VALUE: 0
 HELP: Number of Chief points (0 for none, max 5)?
 QUESTION: Chief_point_x1:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y1:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_x2:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y2:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_x3:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y3:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_x4:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y4:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_x5:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y5:
 VALUE: 0
 HELP: Y coordinate of the Chief point

END GENERAL DATA

Batch file (ενεργοποίηση του Helm2D.exe)

```
rem OutputFile: output.dat
echo %1.dat > %2\%1.name
%3\helm2d.exe < %2\%1.name
del %1.post.res
del %1.post.msh
rename output.post.res %1.post.res
rename output.post.msh %1.post.msh
```

Παράρτημα Δ

Εδώ παρατίθενται οι νέες ρουτίνες που χρειάζονται για την κατασκευή του προγράμματος Helm3D.

Ρουτίνα αρχική του προγράμματος Helm3d.for

```
PROGRAM Helm3D
C
C 3-D BEM CODE FOR the Helmholtz Equation
C written by T. W. Wu timwu@engr.uky.edu
C
CHARACTER*15 input
write(*,*) 'Specify input file name.'
read(*,100) input
100 FORMAT(A15)
OPEN(5,FILE=input,STATUS='OLD')
OPEN(6,FILE='output.dat')
OPEN(7,FILE='output.post.res')
C
C-----Print out Header for results
C
WRITE(7,10)
10 FORMAT(1X,'GiD Post Results File 1.0 '/')
C
CALL PREP
CALL PROC
STOP
END
```

Ρουτίνα επεξεργασίας Proc.for

```
SUBROUTINE PROC
C
C---- PROCESSOR
C
COMMON /FRANGE/FREQ1,FREQ2,DF
COMMON /FLUID/RHO,C
COMMON /FRE/FREQ,WN,IFR
100 WN=2.*3.14159*FREQ/C
write(6,*) ' '
write(6,*) ' Frequency=',FREQ,' Hz'
write(6,*) ' Wavenumber=',WN
C
CALL FORM
CALL SOLVE
CALL SOLOUT
CALL FIELD
CALL POWER
CALL RESULTS
IF (DF.EQ.0.0.OR.FREQ2.EQ.0.0) RETURN
```

```

FREQ=FREQ+DF
IFR=IFR+1
IF (FREQ.GT.FREQ2) THEN
  RETURN
ELSE
  CALL BC
  GO TO 100
END IF

END

```

Ρουτίνα ανάγνωσης κορυφών RNode.for

```

SUBROUTINE RNODE
C
C----- READ IN NODAL POINT DATA, re-implemented by Leonidas
C
C-----Nea metabliti NodeName NNAME sto read kai sto write parakato
C
COMMON /NAMES/NNAME(1000),NAMEF(1000)
COMMON /CONTR/NNODE,NELEM
COMMON /NODES/X(3,2000)
READ(5,*) NNODE
DO 30 NR=1,NNODE
  READ(5,*) NNAME(NR),X(1,NR),X(2,NR),X(3,NR)
30 CONTINUE
C
C----- PRINT OUT NODAL POINT COORDINATES
C
WRITE(6,100)
100 FORMAT(/2X,'NODAL POINT COORDINATES:')
DO 102 I=1,NNODE
  WRITE(6,101) NNAME(I),X(1,I),X(2,I),X(3,I)
101 FORMAT(5X,'NODE #',I4,5X,'X=',F8.3,3X,'Y=',F8.3,3X,'Z=',F8.3)
102 CONTINUE
RETURN
END

```

Ρουτίνα ανάγνωσης στοιχείων RElem.for

```

SUBROUTINE RELEM
C
C----- READ IN ELEMENT CONNECTIVITY
C
COMMON /NAMES/NNAME(1000),NAMEF(1000)
COMMON /PROB/INT,ISC
COMMON /CONTR/NNODE,NELEM
COMMON /NODES/X(3,2000)
COMMON /ELEM/NODE(4,2000)

READ(5,*) NELEM
DO 50 NR=1,NELEM
  READ(5,*) ELNUM,NODE(1,NR),NODE(2,NR),NODE(3,NR),NODE(4,NR)
CCCCCCCC-----THIS PART ADDED BY LEONIDAS-----CCCCCCCC
  IF (INT.EQ.1) THEN
C-----LOOP FOR NODES-----
  DO 30 J=1,4
    DO 20 I=1,NELEM
      IF (NODE(J,NR).EQ.NNAME(I)) THEN
        NODE(J,NR)=I
        GOTO 30
      ENDIF
    CONTINUE
  30 CONTINUE
  ENDIF
CCCCCCCC-----
  IF (INT.EQ.0) THEN
C-----LOOP FOR NODES-----

```

```

        DO 35 J=1,4
          DO 25 I=1,NELEM
            IF (NODE(J,NR).EQ.NNAME(I)) THEN
              NODE(J,NR)=I
              GOTO 35
            ENDIF
25      CONTINUE
35      CONTINUE
        ENDIF

50     CONTINUE
C
C     PRINT OUT ELEMENT CONNECTIVITY
C
        WRITE(6,55)
55     FORMAT(/2X,'ELEMENT CONNECTIVITY :'/)
        DO 60 I=1,NELEM
          WRITE(6,200) I,(NODE(J,I),J=1,4)
200    FORMAT(5X,'ELEMENT #',I4,3X,'NODES=',4I5)
60     CONTINUE

        RETURN
        END

```

Ρουτίνα ανάγνωσης σημείων επίλυσης RField.for

```

        SUBROUTINE RFIELD
C
C----- READ IN FIELD POINTS
C
C-----re-implemented by Leonidas (nea metabliti NAMEF)
        COMMON /NAMES/NNAME(1000),NAMEF(1000)
        COMMON /FPOINTS/NF,XF(3,1000),FS(1000)
        COMPLEX FS

        READ(5,*) NF
        DO 30 I=1,NF
          READ(5,*) NAMEF(I),XF(1,I),XF(2,I),XF(3,I)
30      CONTINUE

        RETURN
        END

```

Ρουτίνα επίλυσης στα σημεία (field points) Field.for

```

        SUBROUTINE FIELD
C
C Evaluate Field-Point Solution
C
        COMMON /NAMES/NNAME(1000),NAMEF(1000)
        COMMON /PROB/INT,ISC,ISYM
        COMMON /FLUID/RHO,C
        COMMON /FPOINTS/NF,XF(3,1000),FS(1000)
        COMMON /CONTR/NNODE,NELEM
        COMMON /ELEM/NODE(4,2000)
        COMMON /SOL/SP(2000),VN(4,2000)
        COMMON /NODES/X(3,2000)
        COMMON /FRE/FREQ,WN,IFR
        COMMON /SCA/AM,AF,BE,GA
        DIMENSION XII(12),WT(12)
        DIMENSION QN(3),PSI(4),DPSI(2,4),P(3)
        REAL NN1,NN2,NN3
        COMPLEX IK,SP,VN,FS,PP,VV,GREEN,DGDN,IRHOW

        IF(NF.EQ.0) RETURN

        WRITE(6,*) ' '
        WRITE(6,*) ' Field-Point Solution:'
        WRITE(6,*) ' '

```

```

C1=1.0/12.56637061
IK=CMPLX(0.,1.)*WN
IRHOW=CMPLX(0.,1.)*RHO*6.283185*FREQ

NLOOP=1
IF (ISYM.NE.0) NLOOP=2

DO 900 IP=1,NF
  FS(IP)=0.0
  P(1)=XF(1,IP)
  P(2)=XF(2,IP)
  P(3)=XF(3,IP)
DO 840 LOOP=1,NLOOP
  IF (LOOP.EQ.2) P(ISYM)=-XF(ISYM,IP)
  XP=P(1)
  YP=P(2)
  ZP=P(3)
  DO 800 K=1,NELEM
    I34=4
    IF (NODE(3,K).EQ.NODE(4,K)) I34=3
    CALL GETINT(NINP,XII,WT)
    DO 450 INP1=1,NINP
      DO 400 INP2=1,NINP
        IF (I34.EQ.4) THEN
          XI1=XII(INP1)
          XI2=XII(INP2)
          CONST=1.0
        ELSE
          ETA1=XII(INP1)
          ETA2=XII(INP2)
          RHO1=0.5+0.5*ETA1
          THE=0.5+0.5*ETA2
          XI1=RHO1*(1.0-THE)
          XI2=RHO1*THE
          CONST=0.25*RHO1
        END IF
        CALL SHAPE(XI1,XI2,PSI,DPSI,I34)
        XX=0.0
        YY=0.0
        ZZ=0.0
        DXDS1=0.0
        DXDS2=0.0
        DYDS1=0.0
        DYDS2=0.0
        DZDS1=0.0
        DZDS2=0.0
        PP=0.0
        VV=0.0
      DO 200 I=1,I34
        J=NODE(I,K)
        XX=XX+X(1,J)*PSI(I)
        YY=YY+X(2,J)*PSI(I)
        ZZ=ZZ+X(3,J)*PSI(I)
        DXDS1=DXDS1+X(1,J)*DPSI(1,I)
        DYDS1=DYDS1+X(2,J)*DPSI(1,I)
        DZDS1=DZDS1+X(3,J)*DPSI(1,I)
        DXDS2=DXDS2+X(1,J)*DPSI(2,I)
        DYDS2=DYDS2+X(2,J)*DPSI(2,I)
        DZDS2=DZDS2+X(3,J)*DPSI(2,I)
        PP=PP+SP(J)*PSI(I)
        VV=VV+VN(I,K)*PSI(I)
      CONTINUE
      NN1=DYDS1*DZDS2-DZDS1*DYDS2
      NN2=-(DXDS1*DZDS2-DZDS1*DXDS2)
      NN3=DXDS1*DYDS2-DYDS1*DXDS2
      DETJ=SQRT(NN1**2+NN2**2+NN3**2)
      QN(1)=NN1/DETJ
      QN(2)=NN2/DETJ
      QN(3)=NN3/DETJ
      RX=XX-XP
      RY=YY-YP

```

200

```

RZ=ZZ-ZP
R=SQRT (RX*RX+RY*RY+RZ*RZ)
DRDN=(RX*QN(1)+RY*QN(2)+RZ*QN(3))/R

GREEN=C1*CEXP(-IK*R)/R
DGDN=-(1.0+IK*R)*GREEN/R*DRDN

FACTOR=DETJ*WT(INP1)*WT(INP2)*CONST

FS(IP)=FS(IP)+(-IRHOW*GREEN*VV-PP*DGDN)*FACTOR

400    CONTINUE
450    CONTINUE
800    CONTINUE
840    CONTINUE

XP=XF(1,IP)
YP=XF(2,IP)
ZP=XF(3,IP)

IF(ISC.EQ.1) THEN
  FS(IP)=FS(IP)+AM*CEXP(-IK*(AF*XP+BE*YP+GA*ZP))
END IF

AMP=CABS(FS(IP))
WRITE(6,850) XP,YP,ZP,FS(IP),AMP
850    FORMAT(2X,'X=',F6.2,'Y=',F6.2,'Z=',F6.2,'P=',2F9.3,'AMPL=',F9.3)

900    CONTINUE
RETURN
END

```

Ρουτίνα εγγραφής αποτελεσμάτων Results.for

```

SUBROUTINE RESULTS
C
C PROGRAM FOR GID-helm3dcccccccccc
C
COMMON /NAMES/NNAME(1000),NAMEF(1000)
COMMON /ELEM/NODE(4,2000)
COMMON /PROB/INT,ISC
COMMON /CONTR/NNODE,NELEM
COMMON /NODES/X(3,2000)
COMMON /SOL/SP(2000),VN(4,2000)
COMMON /FPOINTS/NF,XF(3,1000),FS(1000)
COMMON /FRE/FREQ,WN,IFR
INTEGER NNAME,NAMEF
COMPLEX SP,FS

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C-----OUTPUT RESULTS FILE-----C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C-----Print out pressure real on Nodes-----
C
WRITE(7,20) FREQ
20    FORMAT('Result "Pres_real" "frequency" ',F8.3,' Scalar OnNodes'/', 'Values')

DO 40 I=1,NNODE
WRITE(7,30) NNAME(I),REAL(SP(I))
30    FORMAT(2X,I6," ",2X,F10.3)
40    CONTINUE

C-----Print out pressure real at field points-----
C
DO 45 I=1,NF
WRITE(7,46) NAMEF(I),REAL(FS(I))
46    FORMAT(2X,I6," ",4X,F10.3)
45    CONTINUE
WRITE(7,50)

```

```

50     FORMAT(1X,'End Values'/)

C
C-----Print out pressure imaginary on Nodes-----
C
      WRITE(7,60)   FREQ
60     FORMAT('Result "Pres_imag" "frequency" ',F8.3,' Scalar OnNodes'/,'Values')
      DO 80 I=1,NNODE
      WRITE(7,70)   NNAME(I),AIMAG(SP(I))
70     FORMAT(2X,I6," ",2X,F10.3)
80     CONTINUE
C-----Print out pressure imaginary AT FIELD POINTS-----
      DO 85 I=1,NF
      WRITE(7,86)   NAMEF(I),AIMAG(FS(I))
86     FORMAT(2X,I6," ",4X,F10.3)
85     CONTINUE

      WRITE(7,90)
90     FORMAT(1X,'End Values'/)
C
C-----Print out amplitude on Nodes-----
C
      WRITE(7,100)  FREQ
100    FORMAT('Result "AMP" "frequency" ',F8.3,' Scalar OnNodes'/,'Values')
      DO 120 I=1,NNODE
      WRITE(7,110)  NNAME(I),CABS(SP(I))
110    FORMAT(2X,I6," ",2X,F10.3)
120    CONTINUE
C-----Print out PRESSURE AMPLITUDE AT FIELD POINTS-----
      DO 125 I=1,NF
      WRITE(7,126)  NAMEF(I),CABS(FS(I))
126    FORMAT(2X,I6," ",4X,F10.3)
125    CONTINUE

      WRITE(7,130)
130    FORMAT(1X,'End Values'/)
C
C-----Print out DECIBEL on Nodes-----
C
      WRITE(7,140)  FREQ
140    FORMAT('Result "dbspl" "frequency" ',F8.3,' Scalar OnNodes'/,'Values')
      DO 160 I=1,NNODE
      WRITE(7,150)  NNAME(I),20*LOG10(CABS(SP(I))/0.00002)
150    FORMAT(2X,I6," ",2X,F10.3)
160    CONTINUE
C-----Print out DECIBELS AT FIELD POINTS-----
      DO 165 I=1,NF
      WRITE(7,166)  NAMEF(I),20*LOG10(CABS(FS(I))/0.00002)
166    FORMAT(2X,I6," ",4X,F10.3)
165    CONTINUE

      WRITE(7,170)
170    FORMAT(1X,'End Values'/)

      RETURN
      END

```

Παράρτημα Ε

Εδώ παρατίθενται τα αρχεία διαμόρφωσης για την εφαρμογή GiD – Helm3D

Helm3D.cnd αρχείο εισαγωγής συνοριακών συνθηκών

```

NUMBER: 1 CONDITION: Surface_Pressure
CONDTYPE: over surfaces
CONDMESHTYPE: over elements
QUESTION: Pressure_Re

```


VALUE: 0
 QUESTION: Pressure_Im
 VALUE: 0
 END CONDITION
 NUMBER: 2 CONDITION: Surface_Velocity
 CONDTYPE: over surfaces
 CONDMESHTYPE: over elements
 QUESTION: Normal_Velocity_Re
 VALUE: 1
 QUESTION: Normal_Velocity_Im
 VALUE: 0
 END CONDITION
 NUMBER: 3 CONDITION: Surface_Impedance
 CONDTYPE: over surfaces
 CONDMESHTYPE: over elements
 QUESTION: Normal_Impedance_Re
 VALUE: 415
 QUESTION: Normal_Impedance_Im
 VALUE: 0
 END CONDITION

Helm3D.prb αρχείο εισαγωγής στοιχείων προβλήματος

PROBLEM DATA

TITLE: General
 QUESTION: Project_TITLE:
 VALUE: Title_name
 HELP: A descriptive title for the current project
 QUESTION: Analysis_type:#CB#(Internal,External)
 VALUE: Internal
 HELP: Is the analysis of the interior or exterior of a closed volume?
 QUESTION: Loading_type:#CB#(Radiation,Scattering)
 VALUE: Radiation
 DEPENDENCIES:
 (Radiation,HIDE,Planewave_Amplitude,#CURRENT#,HIDE,Planewave_Lx,#CURRENT#,HIDE,Planewave_Ly,#CURRENT#,HIDE,Planewave_Lz,#CURRENT#)
 DEPENDENCIES:
 (Scattering,RESTORE,Planewave_Amplitude,#CURRENT#,RESTORE,Planewave_Lx,#CURRENT#,RESTORE,Planewave_Ly,#CURRENT#,RESTORE,Planewave_Lz,#CURRENT#)
 HELP: Is it a radiation or scattering (incident plane wave) problem?
 QUESTION: Planewave_amplitude:
 VALUE: 1
 HELP: Pressure amplitude of incident plane wave
 QUESTION: Planewave_Lx:
 VALUE: 0
 HELP: X direction cosine of plane wave wavevector
 QUESTION: Planewave_Ly:
 VALUE: 0
 HELP: Y direction cosine of plane wave wavevector
 QUESTION: Planewave_Lz:
 VALUE: -1
 HELP: Z direction cosine of plane wave wavevector
 QUESTION: Symmetry:#CB#(None,Y-Z,X-Z,X-Y)
 VALUE: None
 HELP: On which plane is the geometric symmetric about?

 TITLE: Fluid properties
 QUESTION: Density:
 VALUE: 1.21
 HELP: Density of the fluid
 QUESTION: Speed_of_sound:
 VALUE: 343
 HELP: Speed of sound in the fluid
 QUESTION: Reference_pressure:
 VALUE: 20e-6
 HELP: Reference pressure for SPL calculations

 TITLE: Frequencies
 QUESTION: Frequency_start:

VALUE: 100
 HELP: Starting frequency
 QUESTION: Frequency_stop:
 VALUE: 100
 HELP: Stopping frequency
 QUESTION: Frequency_interval:
 VALUE: 0
 HELP: Frequency increment

TITLE: Output points
 QUESTION: Output_node:
 VALUE: 1
 HELP: Node number of the node data written to the output file
 QUESTION: Field_point_type:#CB#(Straight,Arc,Mesh)
 VALUE: Straight
 DEPENDENCIES:
 (Straight,RESTORE,Number_of_line_field_points,#CURRENT#,RESTORE,Start_point_x,#CURRENT#,RESTORE,Start_point_y,#CURRENT#,RESTORE,Start_point_z,#CURRENT#,RESTORE,X_step,#CURRENT#,RESTORE,Y_step,#CURRENT#,RESTORE,Z_step,#CURRENT#,HIDE,Number_of_arc_field_points,#CURRENT#,HIDE,Arc_centre_X,#CURRENT#,HIDE,Arc_centre_Y,#CURRENT#,HIDE,Arc_centre_Z,#CURRENT#,HIDE,Axis_direction,#CURRENT#,HIDE,Arc_radius,#CURRENT#,HIDE,Start_angle,#CURRENT#,HIDE,Angle_step,#CURRENT#,HIDE,Height_z,#CURRENT#,HIDE,Number_xpoints,#CURRENT#,HIDE,Number_ypoints,#CURRENT#)
 DEPENDENCIES:
 (Arc,HIDE,Number_of_line_field_points,#CURRENT#,HIDE,Start_point_x,#CURRENT#,HIDE,Start_point_y,#CURRENT#,HIDE,Start_point_z,#CURRENT#,HIDE,X_step,#CURRENT#,HIDE,Y_step,#CURRENT#,HIDE,Z_step,#CURRENT#,RESTORE,Number_of_arc_field_points,#CURRENT#,RESTORE,Arc_centre_X,#CURRENT#,RESTORE,Arc_centre_Y,#CURRENT#,RESTORE,Arc_centre_Z,#CURRENT#,RESTORE,Axis_direction,#CURRENT#,RESTORE,Arc_radius,#CURRENT#,RESTORE,Start_angle,#CURRENT#,RESTORE,Angle_step,#CURRENT#,HIDE,Height_z,#CURRENT#,HIDE,Number_xpoints,#CURRENT#,HIDE,Number_ypoints,#CURRENT#)
 DEPENDENCIES:
 (Mesh,HIDE,Number_of_line_field_points,#CURRENT#,HIDE,Start_point_x,#CURRENT#,HIDE,Start_point_y,#CURRENT#,HIDE,Start_point_z,#CURRENT#,HIDE,X_step,#CURRENT#,HIDE,Y_step,#CURRENT#,HIDE,Z_step,#CURRENT#,HIDE,Number_of_arc_field_points,#CURRENT#,HIDE,Arc_centre_X,#CURRENT#,HIDE,Arc_centre_Y,#CURRENT#,HIDE,Arc_centre_Z,#CURRENT#,HIDE,Axis_direction,#CURRENT#,HIDE,Arc_radius,#CURRENT#,HIDE,Start_angle,#CURRENT#,HIDE,Angle_step,#CURRENT#,HIDE,Height_z,#CURRENT#,HIDE,Number_xpoints,#CURRENT#,HIDE,Number_ypoints,#CURRENT#)
 QUESTION: Number_of_line_field_points
 VALUE: 1
 HELP: Number of field points on line (max 100)
 QUESTION: Start_point_x:
 VALUE: 0
 HELP: X coordinate of first field point
 QUESTION: Start_point_y:
 VALUE: 0
 HELP: Y coordinate of first field point
 QUESTION: Start_point_z:
 VALUE: 0
 HELP: Z coordinate of first field point
 QUESTION: X_step:
 VALUE: 0.1
 HELP: X increment for consecutive field points
 QUESTION: Y_Step:
 VALUE: 0.1
 HELP: Y increment for consecutive field points
 QUESTION: Z_Step:
 VALUE: 0.1
 HELP: Z increment for consecutive field points
 QUESTION: Number_of_arc_field_points
 VALUE: 1
 HELP: Number of field points on arc (max 100)
 QUESTION: Arc_centre_X:
 VALUE: 0
 QUESTION: Arc_centre_Y:
 VALUE: 0
 QUESTION: Arc_centre_Z:
 VALUE: 0
 QUESTION: Axis_direction:#CB#(X,Y,Z)
 VALUE: Z
 HELP: Direction of arc axis in space

QUESTION: Arc_radius:
 Value: 1
 QUESTION: Start_angle:
 Value: 0
 HELP: Start angle (deg)
 QUESTION: Angle_step
 Value: 10
 HELP: Angle increment (deg)
 QUESTION: Height_z
 Value: 1
 HELP: Height of mesh
 QUESTION: Number_xpoints
 Value: 10
 HELP: Number of points in x axis
 QUESTION: Number_ypoints
 Value: 10
 HELP: Number of points in y axis

TITLE: Chief_Points
 QUESTION: Number_of_Chief_points:
 VALUE: 0
 HELP: Number of Chief points (0 for none, max 5)?
 QUESTION: Chief_point_x1:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y1:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_z1:
 VALUE: 0
 HELP: Z coordinate of the Chief point
 QUESTION: Chief_point_x2:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y2:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_z2:
 VALUE: 0
 HELP: Z coordinate of the Chief point
 QUESTION: Chief_point_x3:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y3:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_z3:
 VALUE: 0
 HELP: Z coordinate of the Chief point
 QUESTION: Chief_point_x4:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y4:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_z4:
 VALUE: 0
 HELP: Z coordinate of the Chief point
 QUESTION: Chief_point_x5:
 VALUE: 0
 HELP: X coordinate of the Chief point
 QUESTION: Chief_point_y5:
 VALUE: 0
 HELP: Y coordinate of the Chief point
 QUESTION: Chief_point_z5:
 VALUE: 0
 HELP: Z coordinate of the Chief point
 END GENERAL DATA

Helm3D.bas αρχείο γενικής διαμόρφωσης και εγγραφής του προβλήματος

```
*GenData(Project_title)
GENERAL INFORMATION
*#
*# Analysis type
*#
*if (strcmp(GenData(Analysis_type),"Internal")==0)
1*\
*elseif (strcmp(GenData(Analysis_type),"External")==0)
0*\
*else
*MessageBox ERROR : Unknown analysis type.
*endif
*#
*# Scattering (0) or radiation (1)
*#
*if (strcmp(GenData>Loading_type),"Scattering")==0)
,1*\
*elseif (strcmp(GenData>Loading_type),"Radiation")==0)
,0*\
*else
*MessageBox ERROR : Unknown loading type.
*endif
*#
*# Symmetry
*#
*if (strcmp(GenData(Symmetry),"None")==0)
,0
*elseif (strcmp(GenData(Symmetry),"Y-Z")==0)
,1
*elseif (strcmp(GenData(Symmetry),"X-Z")==0)
,2
*elseif (strcmp(GenData(Symmetry),"X-Y")==0)
,3
*else
*MessageBox ERROR : Please define an symmetry type under Data -> Problem Type
*endif
*#
*#Extra info for scattering problems:
*if (strcmp(GenData>Loading_type),"Scattering")==0)
*GenData(Planewave_amplitude,real),*GenData(Planewave_Lx,real),*GenData(Planewave_Ly,real)
),*GenData(Planewave_Lz,real)
*endif
*#
*#
*GenData(Density,real),*GenData(Speed_of_sound,real)
*GenData(Frequency_start,real),*GenData(Frequency_stop,real),*GenData(Frequency_interval,
real)
*#
*#
*#
NODES
*#
*#implemented by ioannidis leonidas 19 -5 -09
*#
*set layer Layer0 *nodes
*loop nodes *OnlyInLayer
*Set var NPOINTS=loopvar
*end nodes
*NPOINTS
*set layer Layer0 *nodes
*loop nodes *OnlyInLayer
*format "%8i,%10.3g,%10.3g,%10.3g"
*NodesNum*NodesCoord(1,real)*NodesCoord(2,real)*NodesCoord(3,real)
*end nodes
*#
*#
*#
ELEMENTS
```

```

*#
*#implemented by ioannidis leonidas 19 -5 -09
*#
*set layer Layer0 *elems
*loop elems *OnlyInLayer
*Set var NELEMS=loopvar
*end elems
*NELEMS
*set layer Layer0 *elems
*loop elems *OnlyInLayer
*format "%8i,%8i,%8i,%8i,%8i"
*elemsnum*elemsConec*elemsConec(3)
*end elems
*#
*# Calculate total number of BCs
*#
**Set var NBC=NELEMS
*Set var ZERO=0.0
*Set var ONE=1.0
*Set var NEGONE=-1.0
*Set Cond Surface_Pressure *elems
*add Cond Surface_Velocity *elems
*add Cond Surface_Impedance *elems
*set var NBC=CondNumEntities(int)
**Set Cond Surface_Pressure *elems
**Set var NBC=NBC+CondNumEntities(int)
**set elems(all)
**Set Cond Surface_Velocity *elems
**Set var NBC=NBC+CondNumEntities(int)
**set elems(all)
**Set Cond Surface_Impedance *elems
**Set var NBC=NBC+CondNumEntities(int)
*if(NBC==0)
*MessageBox ERROR : No boundary conditions defined.
*endif
BOUNDARY CONDITIONS
*operation(NELEMS+NBC)
*#
*# Write out default boundary condition, V=0
*#
*set layer Layer0 *elems
*loop elems *OnlyInLayer
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*ElemsNum*ElemsNum*ZERO*ZERO*ZERO*ONE*ZERO*ZERO*ZERO
*end elems
*#
*# Write out Pressure boundary condition
*#
*set elems(all)
*Set Cond Surface_Pressure *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInCond
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*ElemsNum*ElemsNum*ZERO*ONE*ZERO*ZERO*ZERO*cond(Pressure_Re,real)*cond(Pressure_Im,real)
*end elems
*endif
*set layer Layer0 *elems
*Set Cond Surface_Velocity *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInLayer
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"
*ElemsNum*ElemsNum*ZERO*ZERO*ZERO*NEGONE*ZERO*cond(Normal_Velocity_Re,real)*cond(Normal_V
elocity_Im,real)
*end elems
*endif
*set elems(all)
*Set Cond Surface_Impedance *elems
*if(CondNumEntities(int)>0)
*loop elems *OnlyInCond
*format "%8i,%8i,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g,%10.3g"

```

```

*ElmsNum*ElmsNum*ZERO*NEGONE*ZERO*cond(Normal_Impedance_Re,real)*cond(Normal_Impedance_
Im,real)*ZERO*ZERO
*end elems
*endif
*#
*# Field points
*#
*if(strcmp(GenData(Field_point_type),"Straight")==0)
*# Field points are on a straight line
*if(GenData(Number_of_line_field_points,int)>0)
*if(GenData(Number_of_line_field_points,int)<100)
*Set var NFLDPT=GenData(Number_of_line_field_points,int)
*else
*Set var NFLDPT=100
*WarningBox WARNING: Only first 100 field points will be used
*endif
FIELD POINTS
*NFLDPT
*Set var XF=GenData(Start_point_x,real)
*Set var YF=GenData(Start_point_y,real)
*Set var ZF=GenData(Start_point_z,real)
*for(IField=1;IField<=NFLDPT;IField=IField+1)
*format "%10.3g,%10.3g,%10.3g"
*XF*YF*ZF
*Set var XF=XF+GenData(X_Step,real)
*Set var YF=YF+GenData(Y_Step,real)
*Set var ZF=ZF+GenData(Z_Step,real)
*endfor
*endif
*#
*elseif(strcmp(GenData(Field_point_type),"Arc")==0)
*# Field points are on an arc
*#
*if(GenData(Number_of_arc_field_points,int)>0)
*if(GenData(Number_of_arc_field_points,int)<100)
*Set var NFLDPT=GenData(Number_of_arc_field_points,int)
*else
*Set var NFLDPT=100
*WarningBox WARNING: Only first 100 field points will be used
*endif
FIELD POINTS
*NFLDPT
*Set var PI=3.141592654
*Set var THETA=GenData(Start_angle,real)*PI/180.0
*Set var DTHETA=GenData(Angle_step,real)*PI/180.0
*Set var X0=GenData(Arc_centre_X,real)
*Set var Y0=GenData(Arc_centre_Y,real)
*Set var Z0=GenData(Arc_centre_Z,real)
*for(IField=1;IField<=NFLDPT;IField=IField+1)
*Set var U=GenData(Arc_radius,real)*cos(THETA)
*Set var V=GenData(Arc_radius,real)*sin(THETA)
*if(strcmp(GenData(Axis_direction),"X")==0)
*format "%10.3g,%10.3g,%10.3g"
*X0*operation(Y0+U)*operation(Z0+V)
*elseif(strcmp(GenData(Axis_direction),"Y")==0)
*format "%10.3g,%10.3g,%10.3g"
*operation(X0+V)*Y0*operation(Z0+U)
*elseif(strcmp(GenData(Axis_direction),"Z")==0)
*format "%10.3g,%10.3g,%10.3g"
*operation(X0+U)*operation(Y0+V)*Z0
*endif
*Set var THETA=THETA+DTHETA
*endfor
*endif
*#
*elseif(strcmp(GenData(Field_point_type),"Mesh")==0)
*# Field points from mesh conditions
FIELD POINTS
*set layer field *nodes
*loop nodes *OnlyInLayer
*Set var NFLDPT=loopvar

```

```

*end nodes
*NFLDPT
*set layer field *nodes
*loop nodes *OnlyInLayer
*format "%8i,%10.3g,%10.3g,%10.3g"
*NodesNum*NodesCoord(1,real) *NodesCoord(2,real) *NodesCoord(3,real)
*end nodes
*endif
*#
*# Chief points - modded A Duncan 3/1/06 to allow up to 5 Chief points
*#
*if (GenData(Number_of_Chief_points,int)>0)
CHIEF POINTS
*GenData(Number_of_Chief_points,int)
*format "%10.3g,%10.3g,%10.3g"
*GenData(Chief_point_x1,real)*GenData(Chief_point_y1,real)*GenData(Chief_point_z1,real)
*endif
*if (GenData(Number_of_Chief_points,int)>1)
*format "%10.3g,%10.3g,%10.3g"
*GenData(Chief_point_x2,real)*GenData(Chief_point_y2,real)*GenData(Chief_point_z2,real)
*endif
*if (GenData(Number_of_Chief_points,int)>2)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x3,real)*GenData(Chief_point_y3,real)*GenData(Chief_point_z3,real)
*endif
*if (GenData(Number_of_Chief_points,int)>3)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x4,real)*GenData(Chief_point_y4,real)*GenData(Chief_point_z4,real)
*endif
*if (GenData(Number_of_Chief_points,int)>4)
*format "%10.3g%10.3g%10.3g"
*GenData(Chief_point_x5,real)*GenData(Chief_point_y5,real)*GenData(Chief_point_z5,real)
*endif
*if (GenData(Number_of_Chief_points,int)>5)
*WarningBox WARNING: Only a maximum of 5 Chief points will be used
*endif
SOLVE

```

Helm3D.bat (batch file) αρχείο ενεργοποίησης

```

rem OutputFile: output.dat
echo %1.dat > %2\%1.name
%3\helm3dn.exe < %2\%1.name
del %1.post.res
rename output.post.res %1.post.res

```