

**Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**  
**Παράρτημα Ρεθύμνου**  
**Τμήμα Μουσικής Τεχνολογίας και Ακουστικής**

**Τίτλος Πτυχιακής Εργασίας**

Δημιουργία εφαρμογής για χρήση με ακουστικά με σκοπό προσομοίωση του ακουστικού πεδίου που δημιουργείται στο ακουστικό κανάλι του αυτιού από τη χρήση ηχείων σε φυσικό ακουστικό χώρο.

**Σπουδαστής:**

Λελεδάκης Ιωάννης (Α.Μ. 562)

**Επιβλέπων καθηγητής:**

κ. Παπαδάκης Νικόλαος

# Περιεχόμενα

Περίληψη.....	4
<b>ΚΕΦΑΛΑΙΟ 1ο : ΨΥΧΟΑΚΟΥΣΤΙΚΗ.....</b>	<b>6</b>
1.1 Ανατομία του αυτιού.....	6
1.1.1 Εξωτερικό αυτί.....	6
1.1.1.1 Πτερύγιο (pinna).....	6
1.1.1.2 Ακουστικό κανάλι (auditory canal).....	7
1.1.2 Μέσο αυτί.....	7
1.1.2.1 Σφύρα, άκμονας, αναβολέας.....	7
1.1.2.2 Μύες του αναβολέα και της σφύρας.....	9
1.1.2.3 Aural reflex.....	9
1.1.2.4 Ευσταχιανή σάλπιγγα.....	9
1.1.3 Εσωτερικό αυτί.....	10
1.1.3.1 Αιθουσαίο.....	10
1.1.3.2 Κοχλίας.....	11
1.1.3.2.1 Εσωτερικό του κοχλία.....	11
1.1.3.2.2 Βασική μεμβράνη (basilar membrane).....	12
1.1.3.2.3 ORGAN OF CORTI – Τριχοειδή κύτταρα – Stereocilia.....	13
1.2 Αντίληψη Του Χώρου (Sound Localization).....	14
1.2.1 Εισαγωγή.....	14
1.2.2 Εντοπισμός στο οριζόντιο επίπεδο.....	15
1.2.2.1 Πρότυπα για τον εντοπισμό.....	15
1.2.2.2 Επιδόσεις σε πειράματα ηχητικού εντοπισμού και πλευρικής τοποθέτησης.....	18
1.2.3 Πλευρική Τοποθέτηση.....	19
1.2.4 Εντοπισμός στο Κάθετο Επίπεδο (median plain).....	23
1.2.5 Εντοπισμός της Απόστασης.....	25
<b>ΚΕΦΑΛΑΙΟ 2ο : ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΟΣ.....</b>	<b>26</b>
2.Συνέλιξη (Convolution).....	26
2.1 Γενικά.....	26
2.2 Σχέση μεταξύ συνέλιξης και φιλτραρίσματος.....	27
2.3 Θεώρημα συνέλιξης.....	27
2.4 Διακριτός μετασχηματισμός Fourier (DFT).....	27

2.5 FFT (Γρήγορος μετασχηματισμός Fourier).....	29
2.6 Γρήγορη συνέλιξη.....	29
2.7 Ψηφιακό Φιλτράρισμα με τη χρήση της γρήγορης συνέλιξης.....	30
2.7.1 Τμηματοποίηση του σήματος.....	34
<b>ΚΕΦΑΛΑΙΟ 3ο : ΜΕΘΟΔΟΣ ΥΠΟΛΟΓΙΣΜΟΥ ΚΡΟΥΣΤΙΚΗΣ ΑΠΟΚΡΙΣΗΣ</b>	
<b>MLS</b> .....	39
3. Ακουστικές Μετρήσεις.....	39
3.1 Κρουστική απόκριση.....	39
3.1.1 Μέτρηση της κρουστικής απόκρισης με τη μέθοδο MLS.....	40
3.2 Εφαρμογή Αντήχησης μέσω συνέλιξης (Reverberation via Convolution).....	41
<b>ΚΕΦΑΛΑΙΟ 4ο : ΜΕΘΟΔΟΣ ΠΡΟΣΟΜΟΙΩΣΗΣ ΤΟΥ ΦΥΣΙΚΟΥ</b>	
<b>ΑΚΟΥΣΤΙΚΟΥ ΠΕΔΙΟΥ ΣΕ ΑΚΟΥΣΤΙΚΑ</b> .....	43
4.1 Μέθοδος.....	43
<b>ΚΕΦΑΛΑΙΟ 5ο : ΔΙΑΔΙΚΑΣΙΑ ΜΕΤΡΗΣΕΩΝ ΜΕΘΟΔΟΛΟΓΙΑ</b> .....	46
5.1 Διάταξη.....	46
5.2 Διαδικασία.....	47
<b>ΚΕΦΑΛΑΙΟ 6ο : ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΜΕ ΤΗ ΧΡΗΣΗ ΤΟΥ</b>	
<b>MATLAB</b> .....	49
6.1 Παρουσίαση/Ανάλυση Κώδικα και Εφαρμογής.....	49
6.2 Γραφική διεπαφή χρήστη και περιγραφή χρήσης .....	50
6.3 Περιγραφή του τρόπου δημιουργίας της εφαρμογής μέσω του MATLAB.....	51
6.3.1 Δημιουργία της διεπαφής χρήστη μέσω του guide.....	51
6.3.2 Συγγραφή κώδικα για τη διεπαφή που δημιουργήθηκε μέσω του guide.....	54
6.3.2.1 Ptyxiaki_Leledakis_562_OpeningFcn.....	54
6.3.2.2 view_menu_Callback.....	59
6.3.2.3 load_button_Callback.....	59
6.3.2.4 process_button_Callback.....	60
6.3.2.5 Callbacks για τα κουμπιά που αναπαράγουν τα σήματα στην έξοδο.....	60
6.3.2.6 Callbacks για το κουμπί Draw All και τα υπόλοιπα κουμπιά που αφορούν τη σχεδίαση των γραφημάτων.....	61
6.4 Τρόπος εγκατάστασης της εφαρμογής.....	63
6.5 Τρόπος χρήσης της εφαρμογής.....	63

<b>ΚΕΦΑΛΑΙΟ 7ο : ΣΥΜΠΕΡΑΣΜΑΤΑ – ΠΕΡΙΟΡΙΣΜΟΙ ΤΗΣ ΕΡΕΥΝΑΣ-ΠΡΟ- ΤΑΣΕΙΣ.....</b>	<b>65</b>
7.1 Συμπεράσματα - Προτάσεις.....	65
<b>Βιβλιογραφία.....</b>	<b>66</b>
Ελληνική.....	66
Ξενόγλωσση.....	66
Σχήματα.....	67
<b>ΠΑΡΑΡΤΗΜΑ Α – ΠΛΗΡΗΣ ΛΙΣΤΑ ΚΩΔΙΚΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ.....</b>	<b>68</b>

## Περίληψη

Κατά την ακρόαση με ακουστικά το ακουστικό πεδίο που δημιουργείται στο ακουστικό κανάλι είναι διαφορετικό από όταν η ακρόαση γίνεται σε χώρο. Τρεις είναι οι κύριοι λόγοι γι' αυτό. Ο πρώτος είναι ότι κατά την ακρόαση με ηχεία το ακουστικό πεδίο που δημιουργείται από το ένα ηχείο καταφθάνει και στα δύο αυτιά, πράγμα το οποίο δεν συμβαίνει με την χρήση ακουστικών. Ο δεύτερος λόγος είναι ότι οι συναρτήσεις μεταφοράς του κεφαλιού (Head Related Transfer Functions) επηρεάζουν το ακουστικό πεδίο το οποίο δημιουργείται στο ακουστικό κανάλι εξαιτίας της ακρόασης σε χώρο. Ο τρίτος λόγος είναι ότι κατά την ακρόαση σε χώρο με ηχεία επηρεάζεται η ηχητική συμπεριφορά του χώρου κάτι το οποίο δεν συμβαίνει κατά την ακρόαση με ακουστικά.

Στην εργασία αυτή, με εύρεση των κρουστικών αποκρίσεων που δημιουργούνται από την ακρόαση με ηχεία στο ακουστικό κανάλι του κάθε αυτιού και χρήση της συνέλιξης, θα γίνει προσομείωση του ακουστικού πεδίου που δημιουργείται στο ακουστικό κανάλι του αυτιού από τη χρήση ηχείων σε φυσικό ακουστικό χώρο. Πρακτικά χρειάζονται τέσσερις κρουστικές αποκρίσεις. Δύο για κάθε αυτί εξαιτίας του κάθε ηχείου.

Στην συνέχεια θα γίνει αξιοποίηση των παραπάνω δεδομένων για την συγγραφή κώδικα σε γλώσσα προγραμματισμού MATLAB.

Στο πρώτο μέρος (θεωρητικό) θα περιλαμβάνονται οι εξής ενότητες:

- Σύντομη περιγραφή της ανατομίας του αυτιού
- Αντίληψη του χώρου
- Εντοπισμός πηγής και binaural αντίληψη
- Συνέλιξη
- Μετασχηματισμός Fourier
- Θεώρημα συνέλιξης
- Υπολογισμός Συνέλιξης με χρήση του FFT
- Ακουστικές Μετρήσεις\

Στο δεύτερο, και τελευταίο, μέρος (μετρήσεις/διαδικασία) θα περιλαμβάνονται οι εξής ενότητες:

- Παρουσίαση του τρόπου λήψης των μετρήσεων και των εργαλείων που χρησιμοποιήθηκαν
- Παρουσίαση του τρόπου με τον οποίο θα χρησιμοποιηθούν τα δεδομένα από το πρόγραμμα.
- Κώδικας και ανάλυσή του.
- Περιγραφή του τρόπου εγκατάστασης της εφαρμογής
- Περιγραφή του τρόπου χρήσης της εφαρμογής.
- Συμπεράσματα

Τέλος, το παράρτημα θα περιλαμβάνει τη βιβλιογραφία, την πλήρη λίστα του κώδικα, λίστα των εργαλείων και οργάνων μετρήσεων.

## Θεωρητικό Μέρος

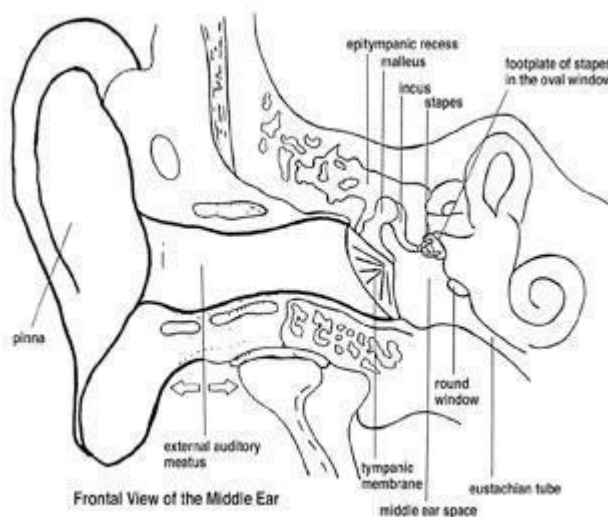
# ΚΕΦΑΛΑΙΟ 1<sup>ο</sup>: ΨΥΧΟΑΚΟΥΣΤΙΚΗ

## Εισαγωγή 1ου Κεφαλαίου

Στο κεφάλαιο αυτό γίνεται αναφορά σε βασικές έννοιες της ψυχοακουστικής, ξεκινώντας από μία σύντομη αναφορά στην ανατομία του ανθρώπινου συστήματος ακοής, και συνεχίζοντας σε έννοιες όπως ο ηχητικός εντοπισμός και η αντίληψη του χώρου. Σκοπός του κεφαλαίου αυτού είναι να γίνει κατανοητός ο τρόπος με τον οποίο μέσω του ακουστικού μας συστήματος εντοπίζουμε ηχητικές πηγές στο χώρο.

## 1.1 Ανατομία του αυτιού

Τα τρία κύρια τμήματα του ανθρώπινου ακουστικού συστήματος, που φαίνονται στο παρακάτω σχήμα (1.1) είναι το έξω αυτί, το μέσο αυτί και το εσωτερικό αυτί.



Σχήμα 1.1: Δομή του αυτιού

### 1.1.1 Εξωτερικό αυτί

Το εξωτερικό αυτί αποτελείται από το περύγιο (pinna) και το ακουστικό κανάλι (auditory canal) το οποίο τελειώνει στην ακουστική μεμβράνη ή τύμπανο από όπου αρχίζει το μέσο αυτί.

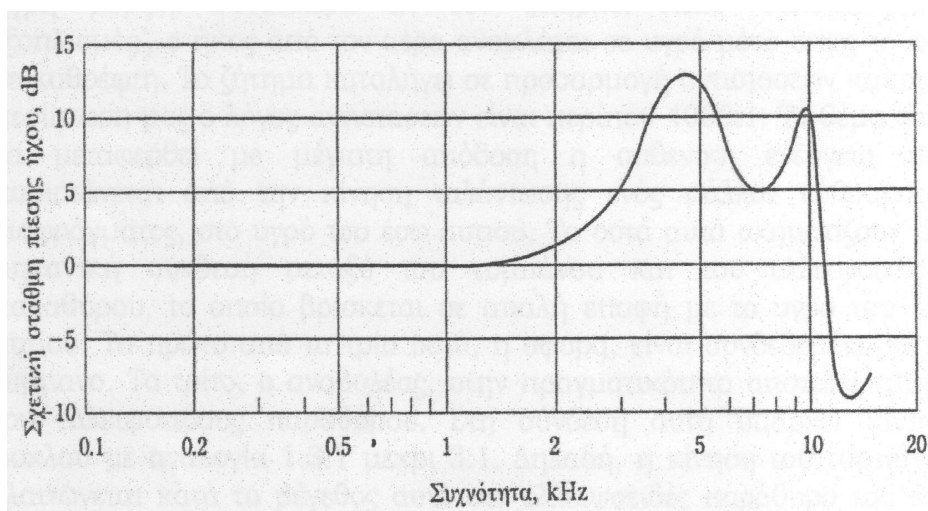
#### 1.1.1.1 Πτερύγιο (pinna)

Το περύγιο είναι μία συσκευή συλλογής ήχων. Το περύγιο προσφέρει διαφοροποίηση των ήχων που προέρχονται από εμπρός σε σύγκριση με αυτούς που προέρχονται από πίσω. Τα ηχητικά ερεθίσματα είναι καλύτερα ακουστά όταν το περύγιο σχηματίζει γωνία 45 μοιρών σε σχέση με τη διεύθυνση μετάδοσής τους.

Η κατασκευή του προσφέρει αύξηση της πίεσης του ήχου κατά περίπου 5dB στις σημαντικές συχνότητες ομιλίας. Οι νυχτερίδες έχουν πολύ εξελιγμένα πτερύγια τα οποία έχουν πολύ μεγάλη ευαισθησία στην αντί-γνευση της κατεύθυνσης μίας ηχητικής πηγής.

#### 1.1.1.2 Ακουστικό κανάλι (auditory canal)

Το ακουστικό κανάλι αυξάνει και αυτό την ηχηρότητα των ήχων που το διαπερνούν. Έχει μέση διάμετρο 0.7cm και το μήκος του περίπου 3cm. Το φαινόμενο του συντονισμού στο κανάλι του αυτιού αυξάνει την πίεση του ήχου στο τύμπανο σε ορισμένες συχνότητες. Το μέγιστο βρίσκεται κοντά στην συχνότητα στην οποία ο σωλήνας των 3 cm έχει ένα τέταρτο μήκους κύματος, περίπου 3000Hz. Η ενίσχυση που παρέχει είναι γύρω στα 10 με 15 dB. Η απόκριση συχνότητας φαίνεται στο σχήμα που ακολουθεί.



Σχήμα 1.2: Απόκριση συχνότητας του καναλιού του αυτιού.

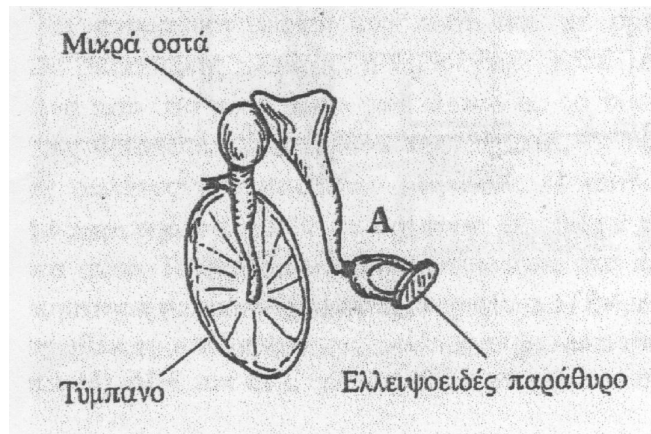
#### 1.1.2 Μέσο αυτί

Το μέσο αυτί αποτελείται από το τύμπανο, τα οστά και τους μύες του μέσου αυτιού, την κοιλότητα του μέσου αυτιού και την ευσταχιανή σάλπιγγα.

##### 1.1.2.1 Σφύρα, άκμονας, αναβολέας

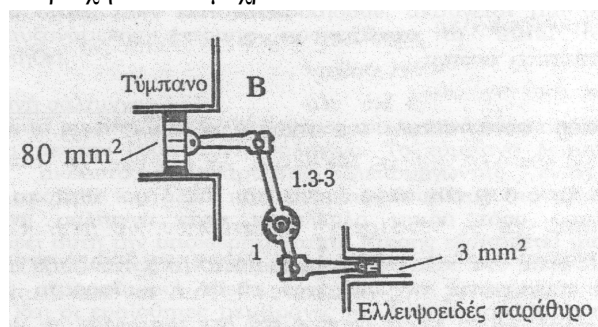
Στο μέσο αυτί υπάρχουν 3 οστά τα οποία είναι ενωμένα μεταξύ τους με μύες όπως θα δούμε παρακάτω. Τα οστά αυτά είναι η σφύρα, ο άκμονας, και ο αναβολέας. Η σφύρα είναι ενωμένη με το τύμπανο, ενώ ο άκμονας είναι συνδεδεμένος με το ελλειψοειδές παράθυρο όπως φαίνεται και στο παρακάτω σχήμα.





Σχήμα 1.3: Οστά στο μέσο αυτί

Τα οστά αυτά είναι τα μικρότερα οστά που υπάρχουν στο ανθρώπινο σώμα. Η λειτουργία τους είναι η μετάδοση της ενέργειας από την ταλάντωση του τυμπάνου και η μεταφορά της στο ελλειψοειδές παράθυρο. Αυτό μεταφράζεται στο πρόβλημα της μεταφοράς ενέργειας από ένα αραιό μέσο όπως είναι ο αέρας σε ένα πυκνό μέσο που έχει υγρή μορφή. Το ζήτημα καταλήγει σε προσαρμογή των αντιστάσεων και στην περίπτωση μας ο λόγος των αντιστάσεων είναι 4000:1. Το θέμα είναι να μεταφερθεί με τη μέγιστη απόδοση η ασθενική ενέργεια που παριστάνεται από την κίνηση ταλάντωσης ενός μάλλον εύθραυστου διαφράγματος, στο υγρό του έσω αυτιού. Τα οστά αυτά σχηματίζουν μία μηχανική σύνδεση μεταξύ του τυμπάνου και του ελλειψοειδούς παραθύρου, το οποίο βρίσκεται σε απαλή επαφή με το υγρό του έσω αυτιού. Το πρώτο από τα τρία οστά, η σφύρα, είναι συνδεδεμένη με το τύμπανο. Το τρίτο, ο αναβολέας, στην πραγματικότητα αποτελεί τμήμα του ελλειψοειδούς παραθύρου. Στη σύνδεση αυτή υπάρχει κίνηση μοχλού με αναλογία 1.3:1 μέχρι 3:1. Δηλαδή, η κίνηση του τυμπάνου ελαττώνεται κατά το μέγεθος αυτό στο ελλειψοειδές παράθυρο του έσω αυτιού. Επίσης η επιφάνεια του τυμπάνου είναι περίπου 80 τετραγωνικά χιλιοστά ενώ του ελλειψοειδούς παραθύρου είναι μόνο 3 τετραγωνικά χιλιοστά. Έτσι μία δεδομένη δύναμη στο τύμπανο αυξάνεται κατά ένα λόγο  $80/3$  ή περίπου 27 φορές. Η συνολική αύξηση δύναμης των παραπάνω είναι μεταξύ 35 και 80 φορές. Αφού ο λόγος ακουστικών αντιστάσεων μεταξύ αέρα και νερού είναι της τάξης 4000:1 ο λόγος πιέσεων που θα χρειαζόταν για την προσαρμογή των δύο μέσων θα ήταν περίπου 63:1 και παρατηρούμε ότι αυτός ο αριθμός βρίσκεται μέσα στην περιοχή από 35 μέχρι 80.



Σχήμα 1.4 Μηχανικό ανάλογο των οστών στο εσωτερικό αυτί.

### 1.1.2.2 Μύες του αναβολέα και της σφύρας

Δύο μικροί μύες είναι συνδεδεμένοι ανάμεσα στα οστά του μέσου αυτιού. Οι μύες αυτοί συστέλλονται σε δυνατούς θορύβους, κατά την φώνηση και στην σωματική κίνηση. Όταν συστέλλονται, ελαττώνουν την ένταση των ήχων που διέρχονται στο έσω αυτί με έμφαση στις συχνότητες κάτω από 2kHz. Αυτή η ανακλαστική κίνηση χρειάζεται περίπου 10 ms για να συμβεί για ήχους υψηλής έντασης. Οι μύες αυτοί θεωρείται ότι επιτελούν διάφορες λειτουργίες όπως:

α) Προστατεύουν το εσωτερικό αυτί

β) Παρέχουν ένα έλεγχο της έντασης για δυνατούς ήχους χαμηλής συχνότητας

γ) Επειδή συστέλλονται εξαιτίας της φώνησης και της σωματικής κίνησης, ελαττώνουν την αντίληψη των ήχων που παράγει ο κάθε άνθρωπος

δ) Επειδή ήχοι χαμηλής συχνότητας μπορεί να καλύψουν ήχους υψηλότερης συχνότητας (masking), η επιλεκτική ελάττωση της έντασης των ήχων αυτών μπορεί να βελτιώνει την κατανόηση πιο περίπλοκων ήχων, όπως η ομιλία.

### 1.1.2.3 Aural reflex

Συμπληρωματική προστασία στο αυτί, πέρα από την λειτουργία των μυών του αναβολέα και της σφύρας, παρέχει και το τύμπανο του αυτιού το οποίο για δυνατούς ήχους αυξάνει την καμπυλότητά του, με αποτέλεσμα να αυξάνει ταυτόχρονα και την σκληράδα του. Ο συνδυασμός αυτών των προστατευτικών μηχανισμών του αυτιού καλείται 'aural reflex'. Οι μηχανισμοί αυτοί έχουν μικρή αποτελεσματικότητα για απότομους ήχους υψηλής έντασης. Το aural reflex' όταν ενεργοποιηθεί, μπορεί να διαρκέσει 2-3 δευτερόλεπτα, μέχρι το αυτί να επανέλθει σε φυσιολογική λειτουργία. Οι μηχανισμοί αυτοί του αυτιού αξιοποιούνται από τις ένοπλες δυνάμεις σε ορισμένες περιπτώσεις. Για διάφορα πυροβόλα όπλα, πριν εκπυρσοκροτήσουν, ακούγεται μία σειρά από ήχους αυξανόμενης έντασης προκειμένου το aural reflex να ενεργοποιηθεί.

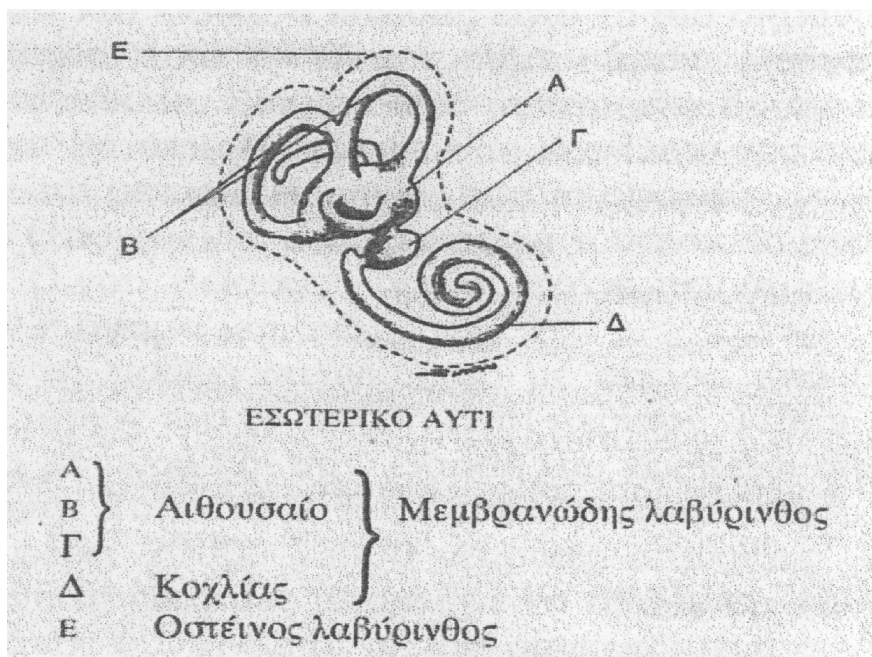
### 1.1.2.4 Ευσταχιανή σάλπιγγα

Το μέσο αυτί συνδέεται με το άνω μέρος του λαιμού πίσω από τη ρινική κοιλότητα με την ευσταχιανή σάλπιγγα. Το τύμπανο λειτουργεί σαν σύστημα ακουστικής ανάρτησης, λειτουργώντας ενάντια στην υποχωρητικότητα του αέρα που είναι παγιδευμένος στο μέσο αυτί. Η ευσταχιανή σάλπιγγα είναι κατάλληλα μικρή τόσο ώστε να μη καταστρέφει αυτή την υποχωρητικότητα. Επίσης εξυπηρετεί μία δεύτερη λειτουργία εξισώνοντας την στατική πίεση του αέρα στο μέσο αυτί με την εξωτερική ατμοσφαιρική πίεση έτσι ώστε να λειτουργούν σωστά το τύμπανο και οι ευαίσθητες μεμβράνες του έσω αυτιού. Κάθε φορά που καταπίνουμε, οι ευσταχιανές σάλπιγγες ανοίγουν, ισορροπώντας την πίεση στο μέσο αυτί. Όταν ένα αεροπλάνο αλλάζει

απότομα ύψος, υπάρχει περίπτωση να χρειάζεται κάποιος χρόνος μέχρι η πίεση στο μέσο αυτό να ισορροπήσει με κατάποση. Η ευσταχιανή σάλπιγγα έχει και μία άλλη λειτουργία ανάγκης, την απαγωγή των υγρών σε περίπτωση μόλυνσης του αυτιού.

### 1.1.3 Εσωτερικό αυτί

Τοποθετημένο βαθιά στο πλάγιο μέρος του κρανίου, το εσωτερικό αυτί περιέχεται μέσα σε ένα οστό όχι συμπαγές, σε μορφή πυραμίδας της οποίας η κορυφή είναι στραμμένη προς τα μέσα και μπροστά. Το εσωτερικό αυτί παρουσιάζεται ως μία οστέινη θήκη η οποία αποκαλείται λαβυρινθική κύστη ή οστέινος λαβύρινθος. Η δομή του κελύφους, πυκνή σαν από ελεφαντοστόν, περιέχει τον μεμβρανώδη λαβύρινθο, ο οποίος αποτελείται από το αιθουσαίο και από τον κοχλία. Πρόκειται για ένα σύνολο ιδιαίτερα περίπλοκο, εξαιτίας της ανατομικής κατασκευής των διαφόρων στοιχείων που το απαρτίζουν. Η διάκριση δε των κοινών δεσμών που συνδέουν τα διάφορα μέρη του εσωτερικού αυτιού είναι πολύ δύσκολη, σχεδόν αδύνατη. Οι δυνατότητες του εσωτερικού αυτιού εκτείνονται από την αντίληψη των σωματικών κινήσεων μέχρι την ικανότητα αποκρυπτογράφησης και ανάλυσης των ελάχιστων κυματικών μετακινήσεων των ηχητικών πιέσεων. Ο μεμβρανώδης λαβύρινθος περιλαμβάνει δύο όργανα: το ένα ονομάζεται αιθουσαίο και το άλλο κοχλίας.

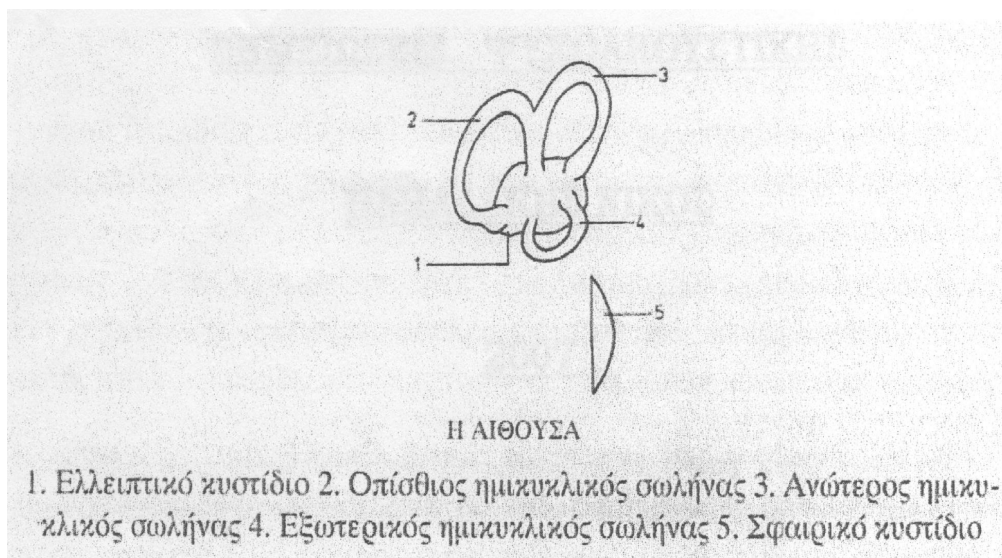


Σχήμα 1.5: Εσωτερικό αυτί

#### 1.1.3.1 Αιθουσαίο

Το αιθουσαίο αποτελείται από το ελλειπτικό κυστίδιο στο οποίο ακουμπούν οι ημικυκλικοί σωλήνες και το σφαιρικό κυστίδιο. Περιλαμβάνει:

- το ελλειπτικό κυστίδιο που οργανώνει τις κινήσεις κυρίως στον τομέα του οριζώντιου και ιδιαίτερα στο επίπεδο του κεφαλιού.
- Το σφαιρικό κυστίδιο που συμπεριφέρεται με ανάλογο τρόπο αλλά επικεντρώνεται περισσότερο στην καθετότητα και ιδιαίτερα στον τομέα της σπονδυλικής στήλης.



*Σχήμα 1.6 : Η Αίθουσα*

- Τους ημικύκλιους σωλήνες που τελειοποιούν τις πλάγιες γωνιώσεις ή τις γωνιώσεις σε επίπεδο εξωτερικό-οπίσθιο και ενισχύουν την αντίληψη της στάσης του σώματος στους τρεις άξονες στο χώρο.

### 1.1.3.2 Κοχλίας

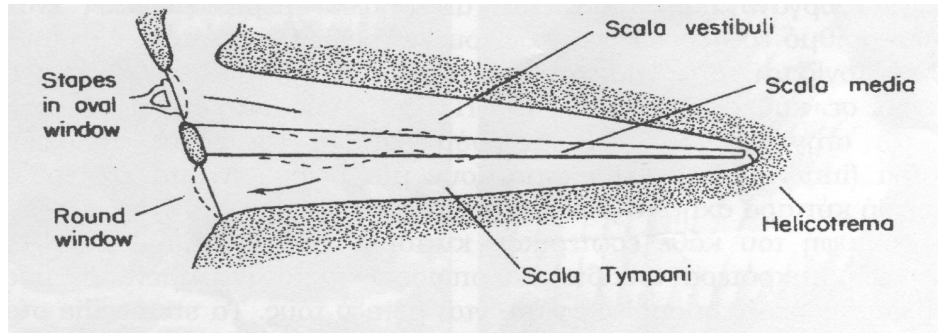
Το εσωτερικό αυτί και ειδικά ο κοχλίας είναι το πιο σημαντικό μέρος της ακοής. Πολλά από όσα ξέρουμε για την λειτουργία του κοχλία προέρχονται από μελέτες του G. von Bekesy ο οποίος έκανε πειράματα αμέσως μετά το θάνατο ζώων ή ακόμα και ανθρώπων.

Ο κοχλίας έχει τη μορφή σαλιγκαριού με σπείρα από δυόμισι μέχρι τρεις έλικες. Το μήκος του σε έκταση είναι περίπου 34mm. Το εσωτερικό του είναι γεμάτο με υγρό και στην έκτασή του χωρίζεται στα δύο από την βασική μεμβράνη. Το όργανο του Corti βρίσκεται πάνω στη βασική μεμβράνη και φέρει 20000-30000 νευρικές απολήξεις με τη μορφή τριχιδίων. Σε γενικές γραμμές μπορούμε να πούμε ότι κάθε τριχίδιο της βασικής μεμβράνης συντονίζεται σε μία συγκεκριμένη συχνότητα και το αποτέλεσμα της ταλάντωσης της μεμβράνης που είναι μία ηλεκτρική τάση μεταφέρεται μέσω νευρών στον εγκέφαλο που γίνεται η αντίληψη του ήχου.

#### 1.1.3.2.1 Εσωτερικό του κοχλία

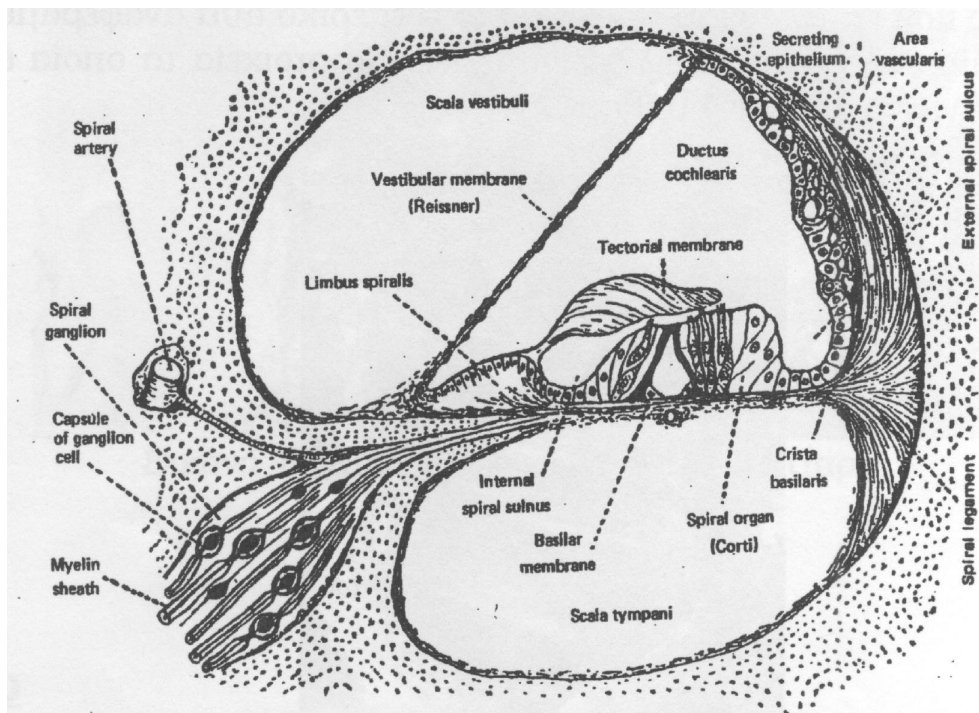
Όπως αναφέραμε και παραπάνω το εσωτερικό του κοχλία χωρίζεται σε τρία μέρη. Στο σχήμα 1.7

βλέπουμε τον κοχλία σε έκταση. Παρατηρούμε ότι αποτελείται από την scala vestibule, την scala media και την scala tympani.



Σχήμα 1.7: Εικόνα του κοχλία σε έκταση

Το ελλειψοειδές παράθυρο (oval window) επικοινωνεί με την scala vestibule, ενώ το στρογγυλό παράθυρο επικοινωνεί με την scala tympani όπως φαίνεται στην παραπάνω εικόνα. Στο σχήμα 1.8 βλέπουμε μία διατομή του κοχλία στην οποία φαίνονται καθαρά τα τρία μέρη του. Η scala vestibule και η scala tympani περιέχουν το υγρό περίλεμφος, ενώ η scala media το υγρό ενδόλεμφος. Παρατηρούμε ότι η βασική μεμβράνη χωρίζει την scala media από τη scala tympani.

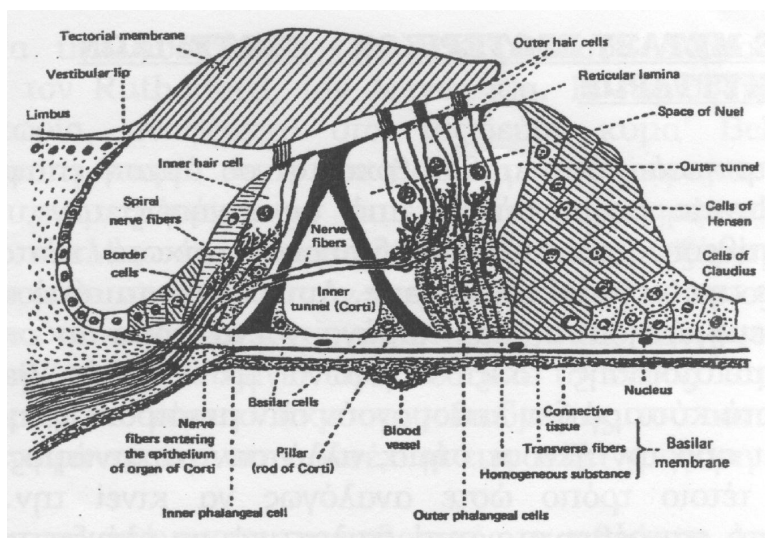


Σχήμα 1.8: Ο κοχλίας σε διατομή

#### 1.1.3.2.2 Βασική μεμβράνη (basilar membrane)

Η scala media χωρίζεται από την scala tympani από μία μεμβράνη που λέγεται βασική μεμβράνη (basilar

membrane). Από την εργασία του Von Bekesy έχει βρεθεί ότι το εύρος της μεμβράνης είναι περίπου 0.1 mm στην βάση του κοχλία και 0.5mm περίπου στο ελικότρημα. Η δυσκαμψία επίσης της μεμβράνης είναι διαφορετική κατα την έκτασή της. Η δυσκαμψία της αυξάνεται κατα 100 περίπου φορές από την κορυφή προς τη βάση του κοχλία. Στην επιφάνεια της βασικής μεμβράνης στην scala media βρίσκεται το organ of corti.



Σχήμα 1.9: Organ of Corti

#### 1.1.3.2.3 ORGAN OF CORTI – Τριχοειδή κύτταρα – Stereocilia

Το organ of corti όπως φαίνεται και στο σχήμα 1.9, είναι ένα σύνθετο βιολογικό όργανο που αποτελείται από πολλά μέρη. Περιέχει έναν μεγάλο αριθμό εσωτερικών τριχοειδών κυττάρων. Υπολογίζεται ότι υπάρχουν περίπου 20000 – 30000 τριχοειδή κύτταρα σε κάθε ανθρώπινο αυτί. Μέσα στο organ of corti τα εσωτερικά τριχοειδή κύτταρα σχηματίζουν μία σειρά, ενώ τα εξωτερικά τριχοειδή κύτταρα σχηματίζουν τρεις σειρές.

Στην κορυφή του κάθε εσωτερικού κυττάρου βρίσκονται περίπου 40 stereocilia (μικρότερα τριχίδια) τα οποία είναι κατανομημένα σε δύο παράλληλες σειρές οι οποίες εφάπτονται μεταξύ τους. Τα stereocilia στα εσωτερικά τριχοειδή κύτταρα σχηματίζουν ένα ανοικτό U. Στην κορυφή των εξωτερικών τριχοειδών κυττάρων βρίσκονται περίπου 150 stereocilia τα οποία είναι κατανομημένα σε τρεις σειρές σχήματος V. Οι κινήσεις της βασικής μεμβράνης προκαλούν τριβή μεταξύ των τριχιδίων και της tectorial membrane. Κάθε stereocilia είναι συνδεδεμένο με νευρικά κύτταρα, τα οποία μετατρέπουν την μηχανική κίνηση σε ηλεκτρικό σήμα το οποίο μεταφέρεται στον εγκέφαλο.

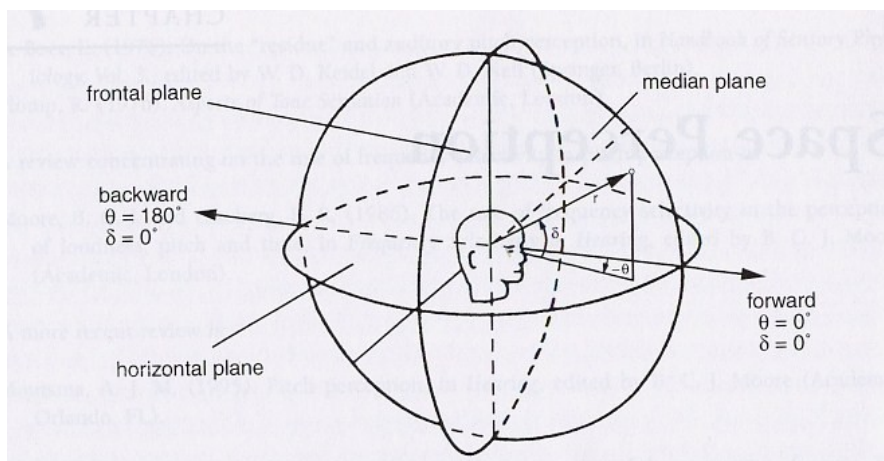
## 1.2 Αντίληψη Του Χώρου (Sound Localization)

### 1.2.1 Εισαγωγή

Η ικανότητα εντοπισμού ηχητικών πηγών είναι μεγάλης σημασίας τόσο για τους ανθρώπους όσο και για τα υπόλοιπα έμβια όντα. Καθορίζει την διεύθυνση που βρίσκονται αντικείμενα προς αναζήτηση ή αποφυγή και υποδεικνύει την κατάλληλη διεύθυνση έτσι ώστε να στραφεί εκεί η οπτική προσοχή. Ενώ τα πιο αξιόπιστα στοιχεία που χρησιμοποιούνται για τον εντοπισμό ήχων εξαρτώνται από την σύγκριση των σημάτων που φτάνουν και στα δύο αυτιά, ο εντοπισμός μπορεί επίσης να βασίζεται εν μέρει και στο σήμα ενός αυτιού μόνο.

Ο όρος “εντοπισμός” αναφέρεται στον καθορισμό της κατεύθυνσης και της απόστασης από μία ηχητική πηγή. Κάποιες φορές, όταν φοράμε ακουστικά, το ηχητικό είδωλο της πηγής εντοπίζεται μέσα στο κεφάλι. Τα ακουστικά επιτρέπουν τον ακριβή έλεγχο των διαφορών του σήματος μεταξύ των αυτιών και την εξάλειψη των επιδράσεων από τις ανακλάσεις του χώρου. Επομένως, η πλευρική τοποθέτηση (lateralization) μπορεί να θεωρηθεί ως μία εργαστηριακή έκδοση του εντοπισμού, που μας παρέχει ικανά μέσα για τη μελέτη της αντίληψης της κατεύθυνσης. Το εύρος των πιθανών πλευρικών θέσεων είναι περιορισμένο: Οι πιο πλευρικοί ήχοι εντοπίζονται σε κάποιο σημείο πάνω στη νοητή γραμμή που ενώνει τα δύο αυτιά, αλλά ένας εντοπισμένος ήχος μπορεί να θεωρηθεί ότι έρχεται από οποιαδήποτε κατεύθυνση.

Είναι χρήσιμο να ορίσουμε μερικούς κοινούς όρους που χρησιμοποιούνται σε μελέτες ηχητικού εντοπισμού. Ο όρος monaural (μονόωτος) αναφέρεται στις καταστάσεις όπου ο ήχος φτάνει στο ένα αυτί μόνο. Ο όρος binaural (δίωτος) αναφέρεται σε καταστάσεις όπου ο ήχος φτάνει και στα δύο αυτιά. Όταν τα ερεθίσματα που φτάνουν και στα δύο αυτιά είναι πανομοιότυπα, λέγεται diotic (διωτικός). Όταν ο ήχος είναι διαφορετικός στα δύο αυτιά, λέγεται dichotic (διχωτικός). Οι διευθύνσεις των ηχητικών πηγών στο χώρο ορίζονται συνήθως σχετικά με το κεφάλι. Για το σκοπό αυτό, τρία επίπεδα ορίζονται, όπως αυτά φαίνονται στο παρακάτω σχήμα.



Σχήμα 1.10: Σύστημα συντεταγμένων για τον καθορισμό της θέσης μίας ηχητικής πηγής σε σχέση με το κεφάλι

Το οριζόντιο επίπεδο περνάει από τα άνω όρια των εισόδων των ακουστικών καναλιών και τα κάτω όρια των (eye sockets)οπτικών οπών. Το πρόσθιο επίπεδο βρίσκεται σε ορθή γωνία με το οριζόντιο επίπεδο και το τέμνει στα άνω όρια των εισόδων των ακουστικών καναλιών. Το διάμεσο (κάθετο) επίπεδο, που βρίσκεται σε ορθή γωνία με το οριζόντιο και το πρόσθιο επίπεδο. Το σημείο όπου συναντιούνται και τα τρία επίπεδα βρίσκεται περίπου στο κέντρο του κεφαλιού. Ορίζει το σημείο αναφοράς ενός συστήματος συντεταγμένων για τον καθορισμό των γωνιών των ήχων σε σχέση με το κεφάλι. Η διεύθυνση ενός ήχου μπορεί να καθοριστεί από το αζιμούθιο του και το ύψος του. Όλοι οι ήχοι οι οποίοι βρίσκονται στο διάμεσο επίπεδο έχουν  $0^\circ$  αζιμούθιο. Όλοι οι ήχοι που βρίσκονται στο οριζόντιο επίπεδο, έχουν  $0^\circ$  ύψος. Ένας ήχος με  $0^\circ$  αζιμούθιο και  $0^\circ$  ύψος, βρίσκεται ακριβώς μπροστά από το κεφάλι. Ένας ήχος με  $90^\circ$  αζιμούθιο και  $0^\circ$  ύψος, βρίσκεται ακριβώς απέναντι από το αριστερό αυτί. Ένας ήχος με  $180^\circ$  αζιμούθιο και  $0^\circ$  ύψος βρίσκεται ακριβώς πίσω από το κεφάλι. Ένας ήχος με  $0^\circ$  αζιμούθιο και  $90^\circ$  ύψος βρίσκεται ακριβώς πάνω από το κεφάλι, ενώ ένας ήχος με  $0^\circ$  αζιμούθιο και  $270^\circ$  ύψος βρίσκεται ακριβώς κάτω από το κεφάλι. Γενικά το αζιμούθιο είναι η γωνία που προκύπτει από την προβολή στο οριζόντιο επίπεδο ( $\theta$  στο σχήμα), ενώ το ύψος είναι η γωνία που προκύπτει από την προβολή στο διάμεσο(κάθετο) επίπεδο.

Σχετικά με τον εντοπισμό των ήχων υπάρχουν δύο προσεγγίσεις. Η πρώτη ασχολείται με το κατά πόσο καλά γίνεται αντιληπτή η διεύθυνση μιας ηχητικής πηγής σε σχέση με την ακριβή της διεύθυνση. Η αντίληψη της διεύθυνσης γενικώς ανταποκρίνεται σχετικά καλά σε σχέση με την πραγματική διεύθυνση, παρόλο που σφάλματα με ημιτονικά ερεθίσματα συχνά συμβαίνουν, σχετικά με το αν ο ήχος ερχεται από μπροστά ή από πίσω, ή από πάνω ή από κάτω από το οριζόντιο επίπεδο. Η δεύτερη προσέγγιση του εντοπισμού ασχολείται με το πόσο καλά ο παρατηρητής μπορεί να ανιχνεύσει μία μικρή μεταβολή στην θέση μιας ηχητικής πηγής. Η προσέγγιση αυτή μετράει την ανάλυση του ακουστικού συστήματος. Όταν η ανάλυση μετράται χρησιμοποιώντας ερεθίσματα που προέρχονται από ηχεία, η μικρότερη ανιχνεύσιμη μεταβολή στη γωνιακή θέση σχετικά με τον παρατηρητή, ονομάζεται Ελάχιστη Ακουστή Γωνία (Minimum Audible Angle - MAA).

## 1.2.2 Εντοπισμός στο οριζόντιο επίπεδο

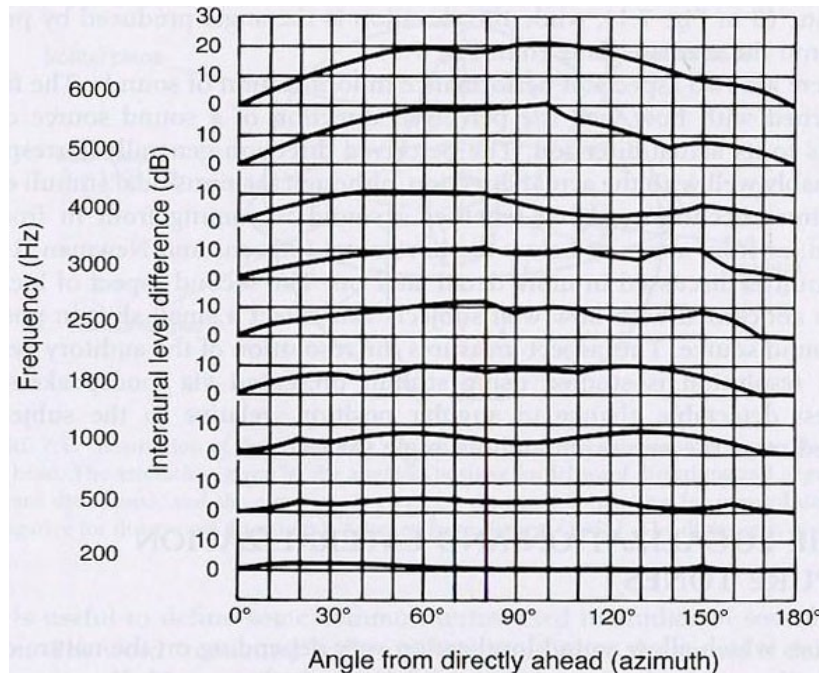
Τα πρότυπα(cues) που επιτρέπουν τον εντοπισμό ήχων ποικίλουν ανάλογα με τη φύση των ήχων.

### 1.2.2.1 Πρότυπα για τον εντοπισμό

Στην περίπτωση που έχουμε μία ημιτονική πηγή ήχου τοποθετημένη στη μία πλευρά του κεφαλιού στο οριζόντιο επίπεδο, για παράδειγμα, με αζιμούθιο  $45^\circ$  και ύψος  $0^\circ$ . Ο ήχος που φτάνει το απομακρυσμένο αυτί καθυστερεί χρονικά και είναι πιο εξασθενημένος από αυτόν που φτάνει στο κοντινό αυτί. Επομένως υπάρχουν δύο πιθανά πρότυπα για τον εντοπισμό μίας ηχητικής πηγής: η χρονική μεταξύ των αυτιών διαφορά (ITD – interaural time difference : ενδοακουστική χρονική διαφορά) και η διαφορά έντασης μεταξύ των δύο αυτιών(IID – interaural intensity difference : ενδοακουστική διαφορά έντασης). Όταν

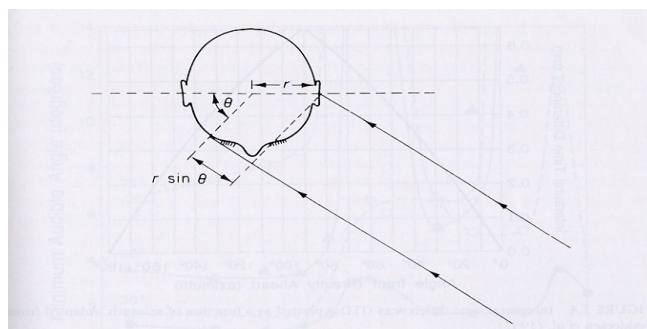


παρουσιάζεται σε dB, η IID αναφέρεται και ως ILD (interaural level difference : ενδοακουστική διαφορά στάθμης). Λόγω της φύσης των ήχων, οι ITDs και ILDs δεν έχουν την ίδια επίδραση για όλες τις συχνότητες.



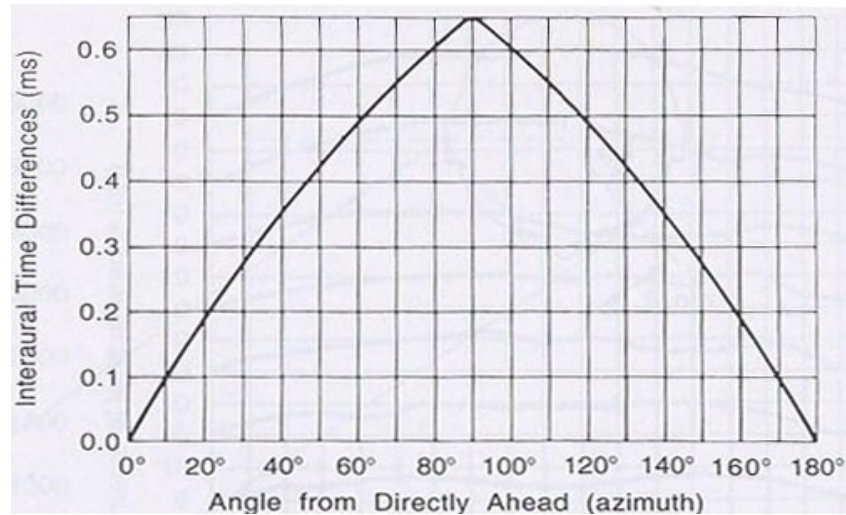
Σχήμα 1.11: Ενδοακουστικές διαφορές στάθμης (ILDs) για ημιτονικά ερεθίσματα, σχεδιασμένο συναρτήσει του αζιμούθιου. κάθε καμπύλη αφορά και διαφορετική συχνότητα.

Χαμηλής συχνότητας ήχοι έχουν μήκος κύματος το οποίο είναι μεγάλο συγκρινόμενο με το μέγεθος του κεφαλιού, με αποτέλεσμα ο ήχος να “κάμπτεται” πολύ εύκολα γύρω από το κεφάλι. Το αποτέλεσμα είναι ότι το κεφάλι ρίχνει λίγη ή και καθόλου ηχητική σκιά. Ενώ αντιθέτως, στις υψηλές συχνότητες όπου το μήκος κύματος είναι μικρό συγκρινόμενο με το μέγεθος του κεφαλιού, ελάχιστη διάθλαση συμβαίνει. Για ηχητικές πηγές οι οποίες είναι απομακρυσμένες από τον ακροατή, οι ILDs είναι αμελητέες περίπου κάτω από τα 500 Hz, αλλά μπορεί να είναι και αρκετά μεγάλες μέχρι και 20dB στις υψηλές συχνότητες όπως φαίνεται στο παρακάτω σχήμα (Feddersen et. al. 1957).



Σχήμα 1.12: Απεικόνιση της μεθόδου για τον υπολογισμό της διαφοράς στο χρόνο άφιξης στα δύο αυτιά για απομακρυσμένη ηχητική πηγή υπο γωνία (αζιμούθιο)  $\theta$  radians σε σχέση με τον παρατηρητή.

Οι ITDs έχουν εύρος τιμών απο 0ms (για ήχο ο οποίος είναι ευθεία μπροστά) μέχρι και 690ms για ένα ήχο με 90° αζιμούθιο ( ακριβώς απέναντι απο το αυτί). Η χρονική διαφορά μπορεί να υπολογιστεί απο τη διαφορά δρόμου μεταξύ των δύο αυτιών, όπως φαίνεται στο παρακάτω σχήμα το οποίο προέρχεται απο τον Feddersen et. al. (1957). Πρακτικά, η ITD για δεδομένο αζιμούθιο ποικίλει ελάχιστα σε σχέση με τη συχνότητα, οπότε η συνάρτηση που φαίνεται στο παρακάτω σχήμα είναι μόνο μία προσέγγιση.



Σχήμα 1.13: Ενδοακουστικές διαφορές χρόνου (ITDs) σχεδιασμένες συναρτήσει του αζιμούθιου.

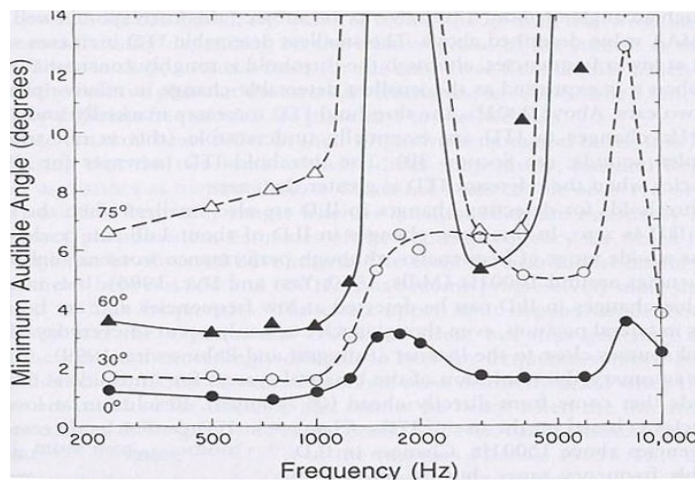
Για ένα ημιτονοειδή τόνο, η χρονική διαφορά είναι ισοδύναμη με μία διαφορά φάσης μεταξύ των δύο αυτιών, η οποία ονομάζεται IPD (interaural phase difference : ενδοακουστική διαφορά φάσης). Για παράδειγμα, για ένα τόνο 200Hz με 5000ms περίοδο, μια ITD 500ms είναι ισοδύναμη με μία IPD 36° (1/10 ενός κύκλου). Για χαμηλής συχνότητας τόνους η IPD μας παρέχει μία αποτελεσματική και σαφή πληροφορία σχετικά με τη θέση ενός ήχου. Παρόλα αυτά, για υψηλές συχνότητες η IPD μας παρέχει μόνο ασαφή πρότυπα (cues). Για παράδειγμα: για ένα ημίτονο 4kHz η περίοδος είναι μόλις 250ms και μία ITD 500ms θα είχε ως αποτέλεσμα δύο ολόκληρους κύκλους IPD. Για υψηλών συχνών ήχους, το ακουστικό σύστημα δεν έχει τρόπο να καθορίσει ποιός κύκλος στο αριστερό αυτί αντιστοιχεί σε ποιό κύκλο στο δεξί αυτί. Οι ασάφειες ξεκινάνε να συμβαίνουν όταν η περίοδος του ήχου είναι περίπου διπλάσια της μέγιστης πιθανής ITD, π.χ. Όταν η περίοδος είναι περίπου 1380ms και η συχνότητα του ήχου είναι περίπου 725Hz. Ένα ημίτονο τέτοιας συχνότητας προσπίπτοντας στη μία πλευρά του κεφαλιού (αζιμούθιο 90°) παράγει κυματομορφές στα δύο αυτιά που είναι αντίθετης φάσης (IPD = 180°). Η θέση της ηχητικής πηγής είναι τώρα ασαφής, απο τη στιγμή που η κυματομορφή στο δεξί αυτί μπορεί να είναι είτε μισό κύκλο πίσω απο το αριστερό ή μισό κύκλο μπροστά, ένας τόνος 725Hz σε -90° αζιμούθιο θα δώσει το ίδιο αποτέλεσμα σε κυματομορφές στα δύο αυτιά όπως και ένας τόνος με 90° αζιμούθιο. Οι κινήσεις της κεφαλής ή της ηχητικής πηγής μπορεί να έχουν ως αποτέλεσμα το ξεκαθάρισμα αυτής της ασάφειας, οπότε δεν υπάρχει κάποιο απόλυτο άνω όριο στην ικανότητα να χρησιμοποιούμε διαφορές φάσης μεταξύ των δύο αυτιών. Παρόλα αυτά, οι διαφορές φάσης γίνονται ιδιέταιρα ασαφείς για συχνότητες άνω των 1500Hz περίπου.

Συνοψίζοντας, για ημιτονοηδείς τόνους, το φυσικό πρότυπο της ILD πρέπει να είναι το πιο χρήσιμο

για υψηλές συχνότητες, ενώ το πρότυπο της ITD πρέπει να είναι το πιο χρήσιμο στις χαμηλές συχνότητες. Η ιδέα που λέει ότι ο ηχητικός εντοπισμός βασίζεται σε ILDs για υψηλές συχνότητες και ITDs για χαμηλές συχνότητες ονομάζεται duplex theory και προέρχεται από τον Lord Rayleigh(1907).

### 1.2.2.2 Επιδόσεις σε πειράματα ηχητικού εντοπισμού και πλευρικής τοποθέτησης

Σε μελέτες για τον εντοπισμό με τη χρήση ημιτονοειδών, χρησιμοποιούνται συνήθως εξάρσεις τόνων με σταδιακή έναρξη και λήξη έτσι ώστε να ελαχιστοποιηθούν τα πρότυπα(cues) σχετικά με την περιβάλλουσα του χρόνου μεταξύ των δύο αυτιών.



Σχήμα 1.14: Η ελάχιστη ακουστή γωνία(MAA) για ημιτονικά σήματα σχεδιασμένη ως συνάρτηση της συχνότητας. κάθε καμπύλη δείχνει τα αποτελέσματα για διαφορετική διεύθυνση αναφοράς. Δεδομένα από Mills(1972).

Η παραπάνω εικόνα δείχνει την MAA (Minimum Audible Angle – Ελάχιστη Ακουστή γωνία) για ημιτονοειδή σήματα που παρουσιάζονται στο οριζόντιο επίπεδο σχεδιασμένα συναρτήσει της συχνότητας. Κάθε καμπύλη δέχνει τα αποτελέσματα για ένα διαφορετικό αξιμούθιο αναφοράς. Το ζητούμενο ήταν να ανιχνευθεί μία μεταβολή στη θέση γύρω από αυτό το αξιμούθιο. Η MAA είναι ελάχιστη για ένα αξιμούθιο αναφοράς 0°, π.χ. Για ήχους που προέρχονται από ευθεία μπροστά από τον παρατηρητή. Μία μεταβολή μίας μόνο μοίρας 1° είναι αντιληπτή για συχνότητες μικρότερες του 1kHz. Οι επιδόσεις πέφτουν για συχνότητες 1500-1600Hz. Αυτό είναι συνεπές με τη διπλή θεωρία, πάνω από τα 1500Hz οι διαφορές φάσης μεταξύ των δύο αυτιών είναι ασαφή πρότυπα(cues) για τον εντοπισμό, αλλά μέχρι τα 1800 Hz οι ILDs είναι μικρές και δεν αλλάζουν ιδιόαιμα με το αξιμούθιο. Οι επιδόσεις πέφτουν αξιοσημείωτα όταν το αξιμούθιο αναφοράς μετακινείται από τις 0°. Πράγματι, για αξιμούθιο αναφοράς 60° και 75° η MAA ήταν τόσο μεγάλη που δεν μπορούσε να καθοριστεί όταν η συχνότητα ήταν γύρω στα 1800Hz. Αυτό μπορεί να γίνει κατανοητό λαμβάνοντας υπόψη το σχήμα 2. Για μια συχνότητα 1800Hz, υπάρχει ILD περίπου 10dB για αξιμούθια από 55° μέχρι και 140°, αλλά η ILD μεταβίαις αλλάζει όταν το αξιμούθιο μεταβάλλεται μέσα σε αυτό το εύρος.

Αυτό οδηγεί σε πολύ μεγάλη ΜΑΑ.

Η ΜΑΑ μετράται χρησιμοποιώντας ήχους οι οποίοι παράγονται από μεγάφωνα, και παράγουν και ITDs και ILDs. Όταν μελετάται η ανάλυση με τη χρήση ακουστικών, είναι εφικτό να μελετηθεί η επίδραση των ILDs ανεξάρτητα από αυτή των ITDs και το αντίστροφο. Στη μία πειραματική μέθοδο, δύο ερεθίσματα, παρουσιάζονται διαδοχικά, ένα εκ των οποίων είναι πανομοιότυπο και στα δύο αυτιά ενώ το άλλο μπορεί να έχει μία ITD ή ILD. Η εργασία που ανατίθεται στον παρατηρητή είναι συνήθως να υποδείξει αν το πρώτο ερέθισμα ήταν προς τ' αριστερά ή τα δεξιά του δεύτερου. Μ' άλλα λόγια, η εργασία είναι να αναγνωρίσει μία μεταβολή στη θέση η οποία σχετίζεται με την ILD ή την ITD.

Τα κατώφλια για την ανίχνευση αλλαγών στην ITD είναι ελάχιστα όταν η ITD αναφοράς είναι μηδέν, ανταποκρινόμενα σε ένα ήχο που ακούστηκε στο κέντρο του κεφαλιού (Yost 1974). Σε αυτή την περίπτωση, μια αλλαγή στην ITD της τάξης των 10μs μπορεί να ανιχνευθεί για συχνότητα 900Hz. Μια τέτοια αλλαγή θα παραχθεί μετακινώντας μια ηχητική πηγή κατά  $1^\circ$  σε σχέση με τον παρατηρητή, αυτό ανταποκρίνεται καλά με την ΜΑΑ τιμή που περιγράψαμε παραπάνω. Η ελάχιστη ανιχνεύσιμη ITD αυξάνεται κάπως στις χαμηλότερες συχνότητες, παρόλο που το κατώφλι είναι περίπου σταθερό στις  $3^\circ$ , όταν εκφράζεται ως η μικρότερη ανιχνεύσιμη αλλαγή στην σχετική φάση στα δύο αυτιά. Πάνω από τα 900Hz, το κατώφλι της ITD αυξάνεται ελάχιστα και πάνω από τα 1500Hz οι αλλαγές στην ITD είναι ουσιαστικά μη ανιχνεύσιμες. Το κατώφλι ITD αυξάνεται για όλες τις συχνότητες όταν το ITD αναφοράς είναι μεγαλύτερο του 0.

Τα κατώφλια για την ανίχνευση αλλαγών στην ILD είναι επίσης ελάχιστα όταν η ILD αναφοράς είναι 0. Σε αυτή την περίπτωση, αλλαγές στην ILD της τάξης του 1dB μπορούν να ανιχνευθούν κατά μήκος ενός μεγάλου εύρους συχνοτήτων, παρόλο που οι επιδόσεις πέφτουν ελαφρά για συχνότητες γύρω στα 1000 Hz (Mills 1960; Yost and Dye 1988). Είναι ενδιαφέρον το ότι οι αλλαγές στην ILD είναι ανιχνεύσιμες σε χαμηλές συχνότητες και ακούγονται ως μεταβολές στην πλευρική θέση, παρόλο που κάτι τέτοιο συμβαίνει στην καθημερινή ζωή για ηχητικές πηγές κοντά στον παρατηρητή (Brungart and Rabinowitz 1999).

Συνοψίζοντας, η ανάλυση του ακουστικού - binaural - συστήματος για ημιτονοειδή είναι καλύτερη για ήχους που προέρχονται από κατευθείαν μπροστά ( $0^\circ$  αζιμούθιο). Η ανάλυση για χαμηλές συχνότητες βασίζεται στη χρήση των ITDs. Αλλαγές στην ITD δεν είναι ανιχνεύσιμες για συχνότητες πάνω από 1500Hz. Αλλαγές στην ILD μπορούν να ανιχνευθούν για όλες τις ακουστές συχνότητες αλλά στην πράξη, ILDs ικανά μεγάλες για να μας παρέχουν χρήσιμα στοιχεία για ηχητικό εντοπισμό υπάρχουν συνήθως μόνο στις υψηλές συχνότητες.

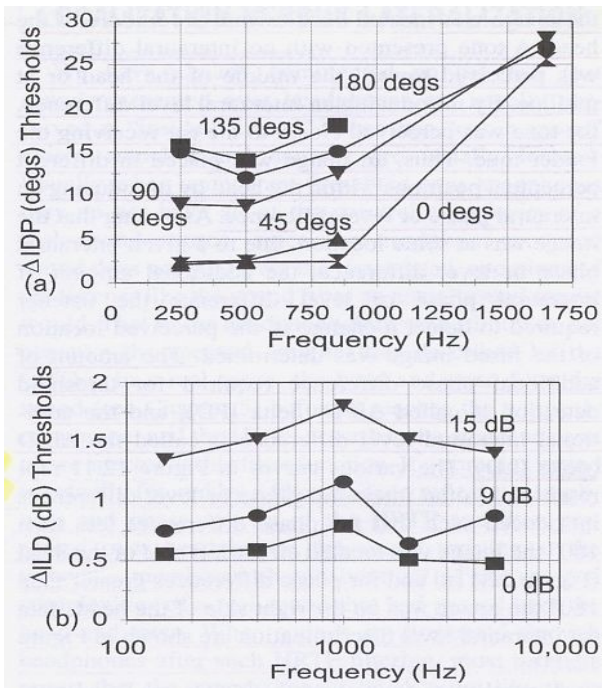
### 1.2.3 Πλευρική Τοποθέτηση

Μελετώντας τον εντοπισμό πραγματικών πηγών ήχου, παρατηρούμε πως είναι αδύνατο να διαχωριστούν οι μεταβλητές του ενδοακουστικού χρόνου από αυτές της ενδοακουστικής στάθμης γιατί και οι δύο

αυτές διαφορές συνυπάρχουν πάντα σε πειράματα εντοπισμού. Επιπλέον, οι διαφορές που προκύπτουν από τις HRTFs στο φάσμα δεν μπορούν να ελεγχθούν με ακρίβεια σε μια μελέτη που γίνεται σε ελεύθερο ηχητικό πεδίο, και το φαινόμενο προβαδίσματος (βλ. Παρακάτω) είναι σχεδόν πάντοτε παρόν. Ένας απλός τρόπος να ελέγξουμε τα ερεθίσματα με μεγαλύτερη ακρίβεια, είναι αυτά να παρουσιαστούν μέσω ακουστικών. Με αυτό τον τρόπο μπορεί αυτός που εκτελεί το πείραμα να μεταχειριστεί απευθείας την χρονική διαφορά ή τη διαφορά στάθμης, ή κάποια φασματική διαφορά μέσω των ακουστικών και έτσι μπορεί να ελέγχει τις μεταβλητές. Όταν ένας τόνος παρουσιάζεται σε ένα ακροατή μέσω ακουστικών, ο ακροατής στις περισσότερες περιπτώσεις θα αντιληφθεί το είδωλο της πηγής μέσα στο κεφάλι με θέση που αλλάζει συναρτήσει των ILDs και ITDs. Για να διαφοροποιήσουμε την αντίληψη του εσωτερικού (ή ενδοκρανιακού) ειδώλου της πηγής που προκύπτει από την παρουσίαση με ακουστικά, από αυτή του εξωτερικού που προκύπτει από εξωτερικές πηγές ήχου, ο όρος lateralization (πλευρική τοποθέτηση) χρησιμοποιείται για να περιγράψει εσωτερική αντίληψη ενώ ο όρος localization (τοποθέτηση/εντοπισμός) για την εξωτερική αντίληψη. Το είδωλο που σχηματίζεται από binaural παρουσιάσεις μέσω ακουστικών κάποιες φορές αναφέρεται και ως συγχωνευμένο είδωλο (fused image) λόγω του ότι ο ακροατής αναφέρει πως ακούει ένα είδωλο σαν ο ήχος που έφτανε από δύο πηγές στα δύο αφτιά ήταν αντιληπτικά συγχωνευμένος. Ένας ακροατής δεν θα αντιληφθεί ένα συγχωνευμένο είδωλο αν η χρονική διαφορά ή η διαφορά στη συχνότητα είναι πολύ μεγάλες (παραπάνω από κάποια milliseconds ανάλογα με το ερέθισμα), ο ακροατής θα αναφέρει πως άκουσε δύο είδωλα, ένα σε κάθε αφτί. Επίσης, αν τα δύο αφτιά λάβουν ανεξάρτητα ημιτονοειδή σήματα με σημαντική διαφορά στη συχνότητα, ο ακροατής αντιλαμβάνεται δύο είδωλα, ένα σε κάθε αυτί, και οι δύο συχνότητες γίνονται αντιληπτές.

Έχει δειχθεί ότι ένα συγχωνευμένο είδωλο σε ένα πείραμα πλευρικής τοποθέτησης εμφανίζεται πρώτα στο αυτί που λαμβάνει πρώτο το ερέθισμα ή σε αυτό που έλαβε το πιο έντονο ερέθισμα, περίπου με τον ίδιο τρόπο που ένα εξωτερικό είδωλο γίνεται αντιληπτό περισσότερο προς το αυτί που έλαβε τον ήχο πρώτο ( οπότε και πιο έντονο ήχο). Επομένως η διαδικασία της πλευρικής τοποθέτησης παρουσιάζεται κατάλληλη για τη μελέτη των επιδράσεων της χρονικής διαφοράς και της διαφοράς έντασης πάνω στην ικανότητα του ακουστικού μας συστήματος στο να εντοπίζει ηχητικές πηγές.





Σχήμα 1.15: Η τιμή της πρόσθετης IPD (ενδοακουστικής διαφοράς φάσης) και της πρόσθετης ILD (ενδοακουστικής διαφοράς στάθμης) που απαιτείται για να γίνει αντιληπτή μια μεταβολή στην θέση μίας πηγής συναρτήσει της συχνότητας. Οι διάφορες καμπύλες αναπαριστούν τις τιμές των standard IPD ή ILD.

Το παραπάνω σχήμα δείχνει αποτελέσματα πειραμάτων στα οποία τόνοι παρουσιάστηκαν σε διαφορετικές συχνότητες και οι διαφορές χρόνου ( φάσης για την ακρίβεια ) και στάθμης διατηρήθηκαν σταθερές. Αυτά και άλλα αποτελέσματα δείχνουν ότι όσο η ενδοακουστική διαφορά φάσης αυξάνεται προς τις 180°, το συγχωνευμένο είδωλο εντοπίζεται πιο κοντά στο αυτί που έλαβε τον τόνο πρώτο. Καθώς η ενδοακουστική διαφορά φάσης ξεπέρασε τις 180°, το είδωλο εντοπίστηκε στην άλλη πλευρά του κεφαλιού (προς το αυτί που καθυστέρουσε το σήμα) και όταν η ενδοακουστική διαφορά φάσης έφτασε τις 360°, το είδωλο εντοπίστηκε πάλι κοντά στο μέσο του κεφαλιού. Ένας τόνος που παρουσιάστηκε χωρίς ενδοακουστικές διαφορές εντοπίστηκε στο μέσο του κεφαλιού ή πάνω στην ευθεία που ενώνει τα δύο αυτιά. Εισάγοντας ενδοακουστική διαφορά στάθμης, ο τόνος έγινε αντιληπτός πιο κοντά στο αυτί που λάμβανε τον πιο έντονο τόνο. Επομένως ένα είδωλο εντοπίζεται σε διαφορετικές αντιληπτικές θέσεις μέσα στο κεφάλι αν εισάγουμε μία ενδοακουστική διαφορά στάθμης ή φάσης. Υποθέτοντας ότι το είδωλο βρίσκεται σε κάποια θέση λόγω κάποιας ενδοακουστικής διαφοράς φάσης ή στάθμης, και η επιπρόσθετη ενδοακουστική διαφορά φάσης ή στάθμης που θα χρειαστεί ο ακροατής για να ανιχνεύσει την μεταβολή στην αντιληπτή θέση της πηγής ήταν καθορισμένη. Το ποσό της πρόσθετης διαφοράς φάσης που απαιτείται για να γίνει οριακά αντιληπτό αυτό, λέγεται  $\Delta$ IPD (δέλτα IPD), και η πρόσθετη διαφορά στάθμης  $\Delta$ ILD (δέλτα ILD).

Διάφορες πλευρές αυτών των δεδομένων είναι ενδιαφέρουσες. Πρώτα, παρατηρούμε πως για οποιαδήποτε αρχική διαφορά φάσης, το ποσό της  $\Delta$ IPD που απαιτείται για να γίνει ανίχνευση παραμένει σταθερό

για συχνότητες μέχρι τα 900Hz και μετά αυξάνεται. Αυτό δείχνει ότι σε συχνότητες μεγαλύτερες των 900Hz, οι ενδοακουστικές διαφορές φάσης (χρόνου) δεν αρκούν για να επεξεργαστούμε μια ενδοακουστική διαφορά. Αυτό συμφωνεί με την πρόβλεψη των Stevens και Newman που προέβλεπε ότι η ενδοακουστική χρονική διαφορά δεν είναι χρήσιμο στοιχείο για τον εντοπισμό ημιτονοειδών υψηλής συχνότητας. Επομένως το binaural σύστημα δεν χρησιμοποιεί την ενδοακουστική χρονική διαφορά στις υψηλές ημιτονοειδείς συχνότητες για τον εντοπισμό ηχητικών πηγών.

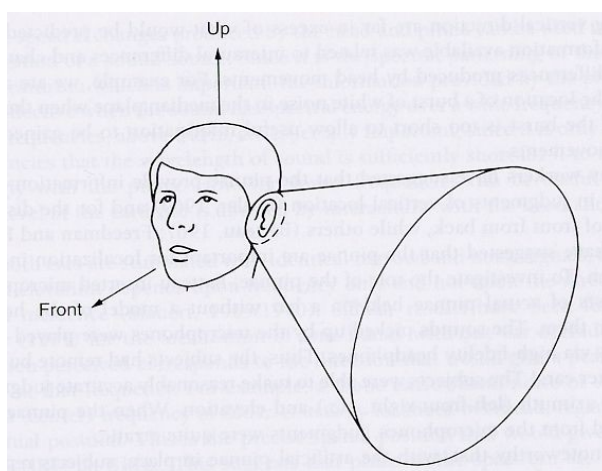
Η δεύτερη ενδιαφέρουσα πλευρά αυτών των δεδομένων είναι ότι καθώς το είδωλο μετακινείται προς τη μία πλευρά του κεφαλιού με την εισαγωγή ενδοακουστικής διαφοράς φάσης κοντά στις 180° ή μία μεγάλη ενδοακουστική διαφορά στάθμης, το ποσό της πρόσθετης ενδοακουστικής διαφοράς φάσης ή στάθμης που απαιτείται για να γίνει αντιληπτή μια μεταβολή στη θέση, αυξάνεται επίσης.

Το γεγονός ότι μέσω ακουστικών η ενδοακουστική διαφορά στάθμης είναι προσεγγιστικά η ίδια για όλες τις συχνότητες (παρατηρούμε εδώ πως υπάρχει μία μικρή αύξηση της ΔILD για συχνότητες γύρω από τα 1000Hz) δεν σημαίνει πως για τον εντοπισμό πηγών σε ελεύθερο πεδίο ο ακροατής θα μπορούσε να χρησιμοποιήσει ενδοακουστικές διαφορές στάθμης στις χαμηλές συχνότητες. Αυτό ισχύει γιατί ακόμα και αν μπορεί ο ακροατής να αντιληφθεί διαφορές 2dB ενδοακουστικής διαφοράς στάθμης στα 200 Hz με ακουστικά, αυτή η 2dB ενδοακουστική διαφορά στάθμης δεν θα συμβεί σε ελεύθερο πεδίο στα 200Hz. Η φυσική ενδοακουστική διαφορά στάθμης στα 200Hz είναι μικρότερη από 2dB. Επομένως, στα 200 Hz στον ακροατή δεν θα παρουσιαστούν αρκετά μεγάλες ενδοακουστικές διαφορές στάθμης στο ελεύθερο πεδίο ώστε να τις χρησιμοποιήσει για εντοπισμό.

Τα δεδομένα που εξετάστηκαν μέχρι τώρα δείχνουν ότι το binaural σύστημα μπορεί να επεξεργαστεί ενδοακουστικές διαφορές χρόνου μόνο για χαμηλής συχνότητας ερεθίσματα. Αυτό δεν ισχύει όμως και για σύνθετους ήχους. Αν ένας υψηλής συχνότητας σύνθετος ήχος παρουσιαστεί, τέτοιος ώστε υπάρχει μία χαμηλής συχνότητας επανάληψη στη χρονική περιβάλλουσα της κυματομορφής, τότε το binaural σύστημα φαίνεται να είναι σχεδόν το ίδιο ευαίσθητο σε διαφορές ενδοακουστικού χρόνου όσο και για ημιτονοειδή χαμηλής συχνότητας. Πολλά ερεθίσματα με συχνотικό περιεχόμενο μόνο υψηλών συχνοτήτων που έχουν όμως στη κυματομορφή μία χαμηλής συχνότητας επανάληψη μπορούν διακριθούν με βάση τις ενδοακουστικές χρονικές διαφορές. Παρατηρήσεις από πειράματα πλευροποίησης (lateralization) οδήγησαν στην επαναδιατύπωση της διπλής θεωρίας (duplex theory): Οι ενδοακουστικές διαφορές στάθμης είναι πρότυπα (cues) που χρησιμοποιούνται για τον εντοπισμό οποιουδήποτε ήχου υψηλής συχνότητας. Οι ενδοακουστικές διαφορές χρόνου είναι πρότυπα που χρησιμοποιούνται για τον εντοπισμό ήχων με χαμηλές συχνότητες ή για ήχους με υψηλές συχνότητες αλλά που περιέχουν χαμηλής συχνότητας επαναλήψεις στην κυματομορφή τους.

### 1.2.4 Εντοπισμός στο Κάθετο Επίπεδο (median plain)

Παραπάνω είδαμε ότι η ενδοακουστική χρονική διαφορά ITD και η ενδοακουστική διαφορά έντασης IID είναι μεγάλης σημασίας για τον εντοπισμό μιας ηχητικής πηγής στο οριζόντιο επίπεδο. Αν το κεφάλι ενός ακροατή παρέμενε σταθερό, τότε υπάρχει ένας αριθμός διαφορετικών θέσεων που θα μπορούσαν να έχουν τις ίδιες τιμές διαφορών έντασης και χρόνου. Για παράδειγμα ένας ήχος μπροστά από τον ακροατή θα είχε τις ίδιες διαφορές με ένα ήχο που είναι ακριβώς πίσω από τον ακροατή, τις ίδιες με ένα που είναι ακριβώς από πάνω και τις ίδιες που είναι ακριβώς από κάτω από τον ακροατή. Το επίπεδο αυτό είναι μια περιοχή σύγκυσης όπου οι ενδοακουστικές διαφορές θα είναι οι ίδιες. Οι περιοχές σύγκυσης υπάρχουν εάν το κεφάλι



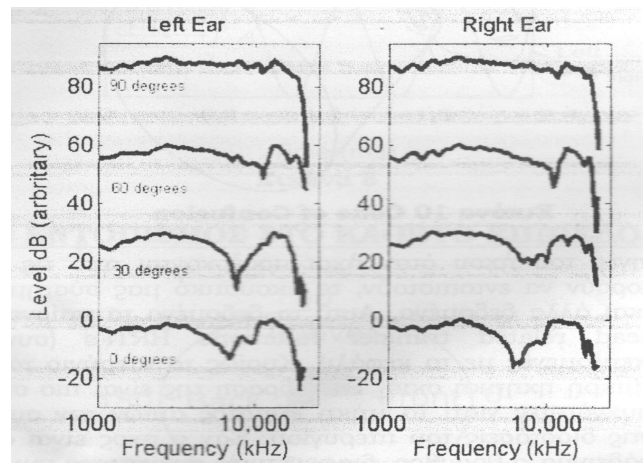
*Σχήμα 1.16: Ένας κώνος σύγκυσης για ένα σφαιρικό κεφάλι και καθορισμένη ITD. Όλες οι ηχητικές πηγές στην επιφάνεια του κώνου θα προκαλέσουν την καθορισμένη αυτή ITD.*

είναι σταθερό. Μικρές κινήσεις του κεφαλιού είναι ικανές για ένα ακροατή να εντοπίσει την ηχητική πηγή. Ακόμα και όταν δεν κινούμε το κεφάλι μας μπορούμε να εντοπίσουμε ήχους σε περιοχές σύγκυσης. Δηλαδή, μπορούμε να κρίνουμε ότι ένας ήχος έρχεται από εμπρός ή από πίσω, ή από πάνω ή από κάτω μας. Ισχύει πάντως ότι όταν κάνουμε λάθη σχετικά με τον εντοπισμό μίας πηγής, αυτά συνήθως είναι στις περιοχές σύγκυσης.

Επειδή οι πηγές του ήχου όταν οι ήχοι προέρχονται από τις περιοχές σύγκυσης μπορούν να εντοπιστούν, το ακουστικό μας σύστημα πρέπει να αξιοποιεί και άλλα δεδομένα. Αυτά τα δεδομένα τα παίρνει από τις λεγόμενες Head Related Transfer Functions HRTFs (συναρτήσεις μεταφοράς σχετιζόμενες με το κεφάλι). Κυρίως το περύγιο του αυτιού δρα σαν μία μικρή ηχητική σκιά. Η επίδραση της είναι πιο σημαντική για υψηλές συχνότητες γιατί τα μήκη κύματος αυτών των συχνοτήτων είναι κοντά στις διαστάσεις του περυγίου. Εάν ο ήχος είναι σύνθετος, όπως για παράδειγμα ο θόρυβος, διαφορετικές συχνότητες του ήχου θα ελαττωθούν και θα καθυστερήσουν χρονικά. Αυτή η καθυστέρηση θα οδηγήσει σε διαφορά φάσης και σαν αποτέλεσμα θα έχουμε ένα φάσμα φάσεων για τις HRTFs. Το ποσό της ελάττωσης και της καθυστέρησης θα

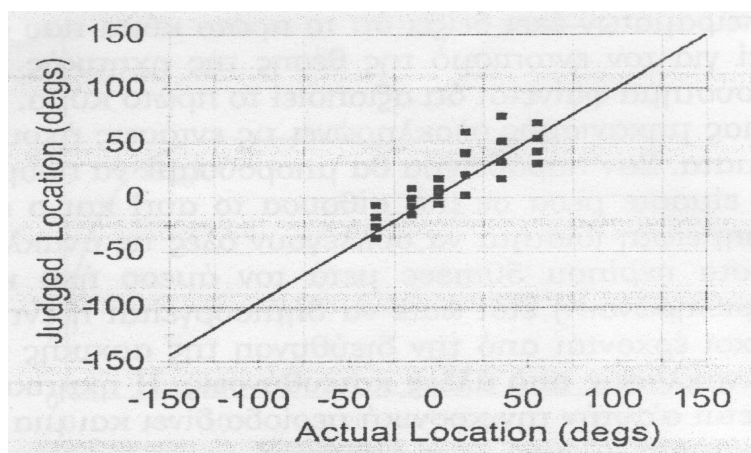


εξαρτάται από την κατεύθυνση από την οποία προέρχεται ο ήχος. Για παράδειγμα, η επίδραση του περυγίου είναι πιο μεγάλη για ήχους που προέρχονται από πίσω σε σχέση με ήχους που έρχονται από μπροστά. Κατά συνέπεια, οι HRTFs για ένα σύνθετο ήχο θα διαφέρουν ανάλογα με τη σχέση που έχει η πηγή του ήχου με το ανθρώπινο σώμα. Επειδή οι μεγαλύτερες αλλαγές στις HRTFs συμβαίνουν για μεγάλες συχνότητες εξαιτίας της αλληλεπίδρασης μεταξύ του μήκους κύματος και του μεγέθους του εμποδίου, δεν είναι περίεργο που οι πιο σημαντικές ενδείξεις για τον εντοπισμό στο κάθετο επίπεδο συμβαίνουν για τις υψηλές συχνότητες. Η παρακάτω εικόνα δείχνει τις HRTFs για τέσσερις γωνίες στον κάθετο άξονα (0°, 30°, 60° και 90°).



Σχήμα 1.17: Δύο set από HRTFs παρουσιάζονται, αυτές που αφορούν το αριστερό αυτί στα αριστερά και αυτές που αφορούν το δεξί αυτί στα δεξιά. Οι μετρήσεις των HRTFs έγιναν στο μεσαίο ακτινικό πεδίο στις 0ο ( ακριβώς μπροστά), 30ο, 60ο και 90ο ( ακριβώς πάνω από το κεφάλι) ύψους. Αξιοσημείωτη είναι η βαθιά φασματική κοιλάδα, και το πώς αυτή μετακινείται φασματικά με τη μεταβολή της γωνίας (ύψους).

Επειδή η ηχητική πηγή είναι στο κάθετο επίπεδο (ανάμεσα στα μάτια) η ένταση που φτάνει σε κάθε αυτί είναι περίπου η ίδια. Παρατηρούμε ότι υπάρχουν κάποια φασματικά κενά κοντά στα 10000 Hz και κυρίως στις 30° και 60°. Τα κενά αυτά φαίνεται ότι μετακινούνται με την αύξηση της γωνίας. Οπότε το φάσμα των HRTFs παρέχει πληροφορίες για τον κάθετο εντοπισμό μιας ηχητικής πηγής. Οι HRTFs χρησιμοποιούνται επίσης και για τον διαχωρισμό εάν η ηχητική πηγή είναι πίσω ή μπροστά από τον ακροατή. Η παρακάτω εικόνα δείχνει την ικανότητα ενός ακροατή να κρίνει την κάθετη τοποθέτηση μίας ηχητικής πηγής με ευρύ φάσμα. Το γεγονός ότι τα δεδομένα είναι διασκορπισμένα μας δείχνει ότι το ακουστικό σύστημα δεν είναι τόσο ικανό να διακρίνει την κάθετη τοποθέτηση ενός ήχου τόσο καλά σε σχέση με την οριζόντια τοποθέτηση.



Σχήμα 1.18: Αντιληπτή θέση στο κάθετο επίπεδο μιας πηγής θορύβου.

Συμπερασματικά μπορούμε να πούμε ότι ο εντοπισμός μίας ηχητικής πηγής στον κάθετο άξονα είναι το αποτέλεσμα της αλληλεπίδρασης του ήχου με τα πτερύγια του αυτιού, το κεφάλι και το σώμα.

### 1.2.5 Εντοπισμός της Απόστασης

Για τον εντοπισμό μίας ηχητικής πηγής σε σχέση με την απόστασή της λιγότερα είναι γνωστά σε σχέση με τις προηγούμενες περιπτώσεις. Μια πιθανή ένδειξη για την απόσταση είναι η ένταση. Γενικά οι πιο δυνατοί ήχοι είναι πιο κοντά σε σχέση με τους μακρινούς. Μπορεί όμως οι δυνατοί ήχοι να είναι μακριά και οι ήχοι χαμηλής έντασης να είναι κοντά. Κατά συνέπεια η ένταση του ήχου μπορεί να είναι ένδειξη για την απόσταση της πηγής, μόνο εάν ο ακροατής έχει πληροφορίες για την αναμενόμενη έντασή της. Για παράδειγμα, εάν κάποιος μιλάει, τότε από εμπειρία ξέρουμε πως αυτό πρέπει να ακούγεται και αυτό μπορεί να δώσει τις πληροφορίες σε κάποιον για την απόσταση της ηχητικής πηγής.

Οι πρώτες ανακλάσεις από μία ηχητική πηγή παρέχουν ενδείξεις για τον εντοπισμό της απόστασης της πηγής. Η αναλογία του άμεσου ήχου που φθάνει στον ακροατή σε σχέση με τις πρώτες ανακλάσεις μειώνεται όσο πιο μακριά είναι η ηχητική πηγή. Κοντά στον ακροατή η ένταση του ανακλώμενου ήχου είναι μικρότερη σε σχέση με τον άμεσο ήχο που φθάνει σε αυτόν. Γενικά όμως μπορούμε να πούμε ότι οι ακροατές έχουν πολύ μικρότερη ικανότητα να κρίνουν την απόσταση μίας ηχητικής πηγής σε σχέση με την ικανότητα να κρίνουν τη θέση της στον οριζόντιο ή κάθετο άξονα.

# ΚΕΦΑΛΑΙΟ 2ο : ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΟΣ

## Εισαγωγή 2ου Κεφαλαίου

Στο κεφάλαιο αυτό γίνεται αναφορά σε βασικές έννοιες της Ψηφιακής Επεξεργασίας Σήματος όπως η συνέλιξη, ο διακριτός μετασχηματισμός Fourier (DFT) και ο γρήγορος μετασχηματισμός Fourier (FFT). Σκοπός του κεφαλαίου αυτού είναι να παρουσιαστούν οι έννοιες της Ψηφιακής Επεξεργασίας Σήματος που θα χρησιμοποιηθούν για τη δημιουργία της εφαρμογής.

## 2.Συνέλιξη (Convolution)

### 2.1 Γενικά

Η Συνέλιξη είναι μία από τις βασικότερες διαδικασίες στην ψηφιακή επεξεργασία ηχητικού σήματος. Η συνέλιξη ενός ήχου με μία κρουστική απόκριση μπορεί να έχει ως αποτέλεσμα μία μεγάλη ποικιλία από μουσικά εφέ. Χαρακτηριστικό παράδειγμα αποτελούν τα Convolution reverbs που υπάρχουν σε μορφή plugin. Στη συνέχεια ακολουθεί μία πιο αναλυτική περιγραφή της συνέλιξης.

Η διαδικασία της συνέλιξης για δύο πεπερασμένες ακολουθίες εκφράζεται μαθηματικά ως:

$$y[k] = \sum_{n=0}^{N-1} x[n] h[k-n]$$

όπου  $N$  είναι το μήκος της ακολουθίας  $x$  και το  $k$  παίρνει τιμές εντός του μήκους του  $h$ . Κάθε δείγμα του  $x[n]$  εξυπηρετεί ως συνάρτηση στάθμισης για ένα μετατοπισμένο χρονικά αντίγραφο του  $h[n]$ . Όλα αυτά τα σταθμισμένα και χρονικά μετατοπισμένα αντίγραφα προστίθενται μεταξύ τους. Ο συμβατικός τρόπος για να υπολογίσουμε αυτή την εξίσωση, είναι να υπολογίσουμε το άθροισμα για κάθε τιμή του  $k$ . Αυτή είναι η απ' ευθείας συνέλιξη ( direct convolution – απ' ευθείας υπολογισμός της συνέλιξης στο πεδίο του χρόνου ). Το μήκος της ακολουθίας εξόδου είναι:

$$\text{μήκος}(output) = \text{μήκος}(x) + \text{μήκος}(h) - 1$$

Στην περίπτωση ενός φίλτρου,  $h$  είναι η κρουστική απόκριση η οποία είναι σύντομη σε σχέση με το σήμα εισόδου. Για παράδειγμα, η κρουστική απόκριση ενός φίλτρου εξομάλυνσης (smoothing filter) διαρκεί μόνο λίγα milliseconds.

## 2.2 Σχέση μεταξύ συνέλιξης και φιλτραρίσματος

Η συνέλιξη είναι άμεσα συνδεδεμένη με το φιλτράρισμα. Ανακαλώντας την γενική εξίσωση ενός FIR φίλτρου:

$$y[n] = (ax[n]) \pm (bx[n-1]) \pm \dots (ix[n-j])$$

μπορούμε να θεωρήσουμε τους παράγοντες  $a, b, \dots, i$  ως στοιχεία ενός πίνακα  $h(i)$ , όπου κάθε στοιχείο του πίνακα  $h(i)$  πολλαπλασιάζεται με το αντίστοιχο στοιχείο ενός πίνακα  $x[j]$ . Με αυτά υπόψιν, η γενική εξίσωση ενός FIR φίλτρου μπορεί να γραφτεί και ως συνέλιξη:

$$y[n] = \sum_{m=0}^{N-1} h[m]x[n-m]$$

όπου  $N$  είναι το μήκος της ακολουθίας  $h$  σε δείγματα και το  $n$  παίρνει τιμές σε όλο το εύρος του  $x$ . Παρατηρούμε πως οι παράγοντες  $h$  παίζουν το ρόλο της κρουστικής απόκρισης στην εξίσωση της συνέλιξης. Πράγματι, μπορούμε να πάρουμε την κρουστική απόκριση ενός FIR φίλτρου από τις τιμές των παραγόντων του. Επομένως κάθε FIR φίλτρο μπορεί να αναπαρασταθεί και ως μία συνέλιξη και αντιστρόφως.

## 2.3 Θεώρημα συνέλιξης

Ένα από τα πιο σημαντικά στοιχεία στην θεωρία των LTI (Linear Time Invariant – Γραμμικά Χρονικά Αμετάβλητα) συστημάτων και γενικά στην ψηφιακή επεξεργασία σήματος, είναι το θεώρημα της συνέλιξης. Αυτό μας λέει ότι η συνέλιξη στο πεδίο του χρόνου ισοδυναμεί με πολλαπλασιασμό στο πεδίο των συχνοτήτων. Ειδικότερα, το θεώρημα αυτό λέει ότι ο διακριτός μετασχηματισμός Fourier (DFT) που σχηματίζεται από τη συνέλιξη δύο ακολουθιών  $x[n]$  και  $h[n]$ , είναι το γινόμενο των DFT των ακολουθιών  $X[n]$  και  $H[n]$ . Δηλαδή μπορούμε εναλλακτικά να υπολογίσουμε την συνέλιξη με τη χρήση του θεωρήματος της συνέλιξης. Αν για ένα σύστημα γνωρίζουμε την κρουστική του απόκριση και θέλουμε να βρούμε την έξοδο του για συγκεκριμένη είσοδο, μπορούμε να βρούμε τον διακριτό μετασχηματισμό Fourier της κρουστικής απόκρισης και της εισόδου και στην συνέχεια να τους πολλαπλασιάσουμε. Αυτό που θα βρούμε θα είναι η έξοδος του συστήματος για τη δεδομένη είσοδο.

## 2.4 Διακριτός μετασχηματισμός Fourier (DFT)

Για να υπολογίσουμε το φάσμα ενός ψηφιακού σήματος χρησιμοποιούμε τον διακριτό μετασχηματισμό Fourier (DFT). Ο διακριτός μετασχηματισμός Fourier είναι ένας τρόπος υπολογισμού του συχνοτικού περιεχομένου μίας πεπερασμένης ακολουθίας αριθμών. Έστω μία ακολουθία  $N$  δειγμάτων:  $x[0], x[1], x[2], \dots, x[N-1]$ . Ονομάζουμε διακριτό μετασχηματισμό Fourier της ακολουθίας αυτής, την ακολουθία:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \quad 0 \leq n < N-1$$

Η ποσότητα  $X[k]$  δίνει το φάσμα σε  $n$  διαφορετικές συχνότητες, από 0 μέχρι την συχνότητα δειγματοληψίας  $f_s$ . Η απόσταση των δειγμάτων είναι  $\Delta f = f_s / N$ . Για να υπολογίσουμε  $N$  συνιστώσες συχνότητας από  $N$  δεδομένα, δηλαδή χρειάζονται  $N^2$  υπολογισμοί.

Εκτός του μετασχηματισμού Fourier υπάρχει και ο αντίστροφος μετασχηματισμός Fourier. Ο αντίστροφος μετασχηματισμός Fourier κάνει ακριβώς το αντίθετο. Μετασχηματίζει τα  $X[k]$  δείγματα τα οποία βρίσκονται στο πεδίο των συχνοτήτων, σε  $x[k]$  δείγματα τα οποία βρίσκονται στο πεδίο του χρόνου.

$$x[k] = \frac{1}{N} \sum_{n=0}^{N-1} X[n] e^{j2\pi kn/N} \quad 0 \leq n < N-1$$

Ο DFT έχει ορισμένες σημαντικές ιδιότητες οι οποίες είναι σημαντικές για διάφορες εφαρμογές. Μερικές από αυτές είναι

1. Γραμμικότητα

Εάν το  $X_1[k]$  είναι ο DFT<sub>N</sub> του  $x_1[n]$  και το  $X_2[k]$  είναι ο DFT<sub>N</sub> του  $x_2[n]$  τότε  $a_1 X_1[k] + a_2 X_2[k]$  είναι ο DFT<sub>N</sub> του  $a_1 x_1[n] + a_2 x_2[n]$  όπου  $a_1$  και  $a_2$  πραγματικές ή μιγαδικές σταθερές.

2. Κυκλική ολίσθηση στη συχνότητα

Εάν το  $X_1[k]$  είναι ο DFT<sub>N</sub> του  $x_1[n]$  και το  $X_2[k]$  είναι ο DFT<sub>N</sub> του  $x_2[n]$  τότε  $X(k-k_0)$  είναι ο DFT<sub>N</sub> του  $x[n] W_N^{k_0 n}$

με άλλα λόγια, ο πολλαπλασιασμός της ακολουθίας  $x[n]$  με την εκθετική ακολουθία  $e^{-j2\pi k_0 n/N}$ , ισοδυναμεί με την κυκλική ολίσθηση στη συχνότητα κατά  $k_0$  μονάδες του DFT αυτής.

3. Χρονική Μετατόπιση

Αν  $X[n]$  είναι ο DFT<sub>N</sub> του  $x[n]$  τότε  $x[n-n_0] \Leftrightarrow X[n] e^{-j2\pi k n_0/N} = X[n] W_n^{kn_0}$

4. Κυκλική συνέλιξη

Εάν το  $X_1[k]$  είναι ο DFT<sub>N</sub> του  $x_1[n]$  και το  $X_2[k]$  είναι ο DFT<sub>N</sub> του  $x_2[n]$  τότε η κυκλική συνέλιξη του  $x_1[n]$  με το  $x_2[n]$  είναι ο DFT<sub>N</sub> του  $X_1[k]$  επί  $X_2[k]$

Η κυκλική συνέλιξη δύο ακολουθιών στο πεδίο του χρόνου, ισοδυναμεί με τον πολλαπλασιασμό των αντίστοιχων DFT αυτών.

5. Πολλαπλασιασμός δύο ακολουθιών

Εάν το  $X_1[k]$  είναι ο DFT<sub>N</sub> του  $x_1[n]$  και το  $X_2[k]$  είναι ο DFT<sub>N</sub> του  $x_2[n]$

τότε  $(1/N)$  επί την κυκλική συνέλιξη του  $X_1[k]$  με το  $X_2[k]$  είναι ο DFT<sub>N</sub> του  $x_1[n]$  επί το  $x_2[n]$

Πρόκειται ουσιαστικά για τη διττή έκφραση της προηγούμενης ιδιότητας, και επαληθεύεται έτσι το γεγονός ότι ο πολλαπλασιασμός δύο ακολουθιών στο πεδίο του χρόνου αντιστοιχεί στην κυκλική συνέλιξη των DFT αυτών.

6. Θεώρημα Parseval

Εάν το  $X_1[k]$  είναι ο  $DFT_N$  του  $x_1[n]$  και το  $X_2[k]$  είναι ο  $DFT_N$  του  $x_2[n]$  τότε

$$\sum_{n=0}^{N-1} |x[n]|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X[k]|^2$$

Το θεώρημα του Parseval εκφράζει τη διατήρηση της ενέργειας κατά τη μετάβαση από το πεδίο του χρόνου στο πεδίο της συχνότητας. Με άλλα λόγια, το άθροισμα των τετραγώνων των τιμών των δειγμάτων ισούται με τη μέση τιμή των τετραγώνων των φασματικών γραμμών. Η αναπαράσταση των  $|X[k]|^2$  ονομάζεται φάσμα ισχύος της  $x[n]$ . Παρατηρείται ότι το φάσμα ισχύος εξαρτάται μόνο από το μέτρο του φάσματος και όχι από τη φάση του.

## 2.5 FFT (Γρήγορος μετασχηματισμός Fourier)

Οι πρώτοι υψηλής απόδοσης αλγόριθμοι για τον υπολογισμό του DFT πρωτοαναπτύχθηκαν στη δεκαετία του 1960. Στο σύνολό τους είναι γνωστοί ως γρήγοροι μετασχηματισμοί Fourier (FFTs), όλοι βασίζονται στο γεγονός ότι ο απλός DFT περιλαμβάνει περιττούς υπολογισμούς. Ο DFT ενός N-μήκους σήματος περιγράφεται από:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} = \sum_{n=0}^{N-1} x[n] W_N^{kn}$$

όπου  $W_N = e^{-j2\pi/N}$  και το  $X[k]$  ισχύει για  $0 \leq k \leq (N-1)$ . Τελικά φαίνεται ότι οι ίδιες τιμές του  $W_N^{kn}$  υπολογίζονται πολλές φορές καθώς γίνονται οι υπολογισμοί και ιδιαίτερα αν οι μετασχηματισμοί έχουν μεγάλο μήκος. Αυτό συμβαίνει επειδή το  $W_N^{kn}$  είναι περιοδική συνάρτηση με περιορισμένο αριθμό διαφορετικών τιμών. Το ίδιο συμβαίνει και για τον αντίστροφο μετασχηματισμό (IDFT). Αποτελεί στόχο του αλγόριθμου FFT να εξαλείψει τους περιττούς υπολογισμούς.

Αυστηρά μιλώντας, ο FFT δεν είναι ένας μετασχηματισμός από μόνος του αλλά ένα σύνολο από αλγόριθμους για τον υπολογισμό του DFT και ο καθένας έχει διαφορετικά χαρακτηριστικά, πλεονεκτήματα και περιορισμούς. Ένας αλγόριθμος ο οποίος είναι κατάλληλος για μία υλοποίηση σε μία υψηλού επιπέδου γλώσσα προγραμματισμού ίσως να μην κάνει για υλοποίηση σε εξειδικευμένο hardware. Αυτό το οποίο έχουν κοινό οι διαφορετικοί αλγόριθμοι FFT είναι η γενική τους προσέγγιση στον υπολογισμό του DFT: την αποσύνθεση ενός DFT σε έναν αριθμό από διαδοχικά μικρότερους και απλούστερους DFTs.

## 2.6 Γρήγορη συνέλιξη

Η απ' ευθείας συνέλιξη (direct convolution) είναι υπολογιστικά πολύ δαπανηρή, απαιτώντας  $N^2$  υπολογισμούς, όπου N το μήκος της μεγαλύτερης ακολουθίας εισόδου. Οπότε η απ' ευθείας συνέλιξη σπάνια

χρησιμοποιείται για την υλοποίηση φίλτρων στενού εύρους ή αντηχητών (και τα δύο έχουν μεγάλες κρουστικές αποκρίσεις) ενώ υπάρχουν πιά απλές μέθοδοι.

Πολλές πρακτικές εφαρμογές της συνέλιξης χρησιμοποιούν μία μέθοδο που ονομάζεται γρήγορη συνέλιξη (Stockham 1969). Η γρήγορη συνέλιξη μεγάλων ακολουθιών εκμεταλεύεται το γεγονός ότι το γινόμενο δύο N-σημείων διακριτών μετασχηματισμών Fourier (DFT) είναι ίσο με τον DFT της συνέλιξης των δύο N-σημείων ακολουθιών. Απο τη στιγμή που ο DFT μπορεί να υπολογιστεί πολύ γρήγορα χρησιμοποιώντας έναν αλγόριθμο FFT (γρήγορος μετασχηματισμός Fourier), αυτό οδηγεί σε δραματική αύξηση της ταχύτητας με την οποία μπορεί να γίνει η συνέλιξη. Πρίν γίνουν οι FFTs και οι δύο ακολουθίες επιμηκύνονται προσθέτοντας μηδενικά μέχρι και οι δύο ακολουθίες να έχουν μήκος ίσο με την έξοδο της συνέλιξης. Η διαδικασία αυτή ονομάζεται zero padding (προσθήκη μηδενικών στοιχείων). Τα αποτελέσματα της συνέλιξης μπορούν να ανασυντεθούν χρησιμοποιώντας τον αντίστροφο FFT (IFFT).

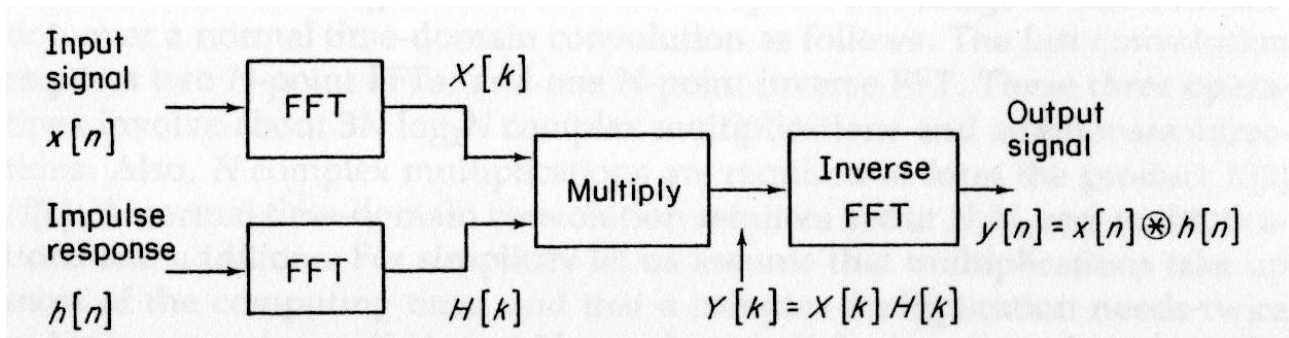
Αυτο σημαίνει πως μπορούμε να αντικαταστήσουμε την απ' ευθείας συνέλιξη με FFTs, οι οποίοι είναι πολύ πιο γρήγοροι για μεγάλες τιμές του N. Συγκεκριμένα, η γρήγορη συνέλιξη κάνει  $N \log_2(N)$  υπολογισμούς. Για παράδειγμα ας σκεφτούμε την περίπτωση απ' ευθείας συνέλιξης δύο ήχων διάρκειας δύο δευτερολέπτων με δειγματοληψία στα 48 kHz. Η διαδικασία αυτή απαιτεί  $96.000^2$  υπολογισμούς. Με τη γρήγορη συνέλιξη απαιτούνται περίπου 1500000 υπολογισμοί, 6144 φορές πιο γρήγορα.

Για πραγματικού χρόνου εφαρμογές όπου λίγο πολύ απαιτείται άμεση έξοδος, είναι δυνατόν να κάνουμε συνέλιξη ανα τμήματα, δηλαδή με λίγα δείγματα σε κάθε τμήμα. Η συνέλιξη ανα τμήματα και η κανονική συνέλιξη παράγουν παρόμοια αποτελέσματα.

## 2.7 Ψηφιακό Φιλτράρισμα με τη χρήση της γρήγορης συνέλιξης

Κατά γενικό κανόνα το ψηφιακό φιλτράρισμα μπορεί να γίνει στο πεδίο των συχνοτήτων. Αντί λοιπόν να κάνουμε συνέλιξη του επιθυμητού σήματος στο πεδίο του χρόνου με την κρουστική απόκριση του επιθυμητού φίλτρου, μετασχηματίζουμε το σήμα και το πολλαπλασιάζουμε με την αντίστοιχη συχνοτική απόκριση. Στο τέλος ένας αντίστροφος μετασχηματισμός μας δίνει το φιλτραρισμένο σήμα. Παρόλο που η μέθοδος αυτή φαίνεται επίπονη, συνήθως αποδεικνύεται γρηγορότερη από την συνέλιξη στο πεδίο του χρόνου. Αυτό συμβαίνει λόγω της σχετικής απλότητας του πολλαπλασιασμού, και της ταχύτητας του FFT. Η προσέγγιση αυτή είναι ιδιαίτερα πολύτιμη για το ψηφιακό φιλτράρισμα μεγάλων ακολουθιών εισόδου.

Συχνά ξεκινάμε με μία κρουστική απόκριση, παρά με μία συχνοτική απόκριση. Είναι αναγκαίο τότε να κάνουμε τον μετασχηματισμό και της κρουστικής απόκρισης στο πεδίο των συχνοτήτων. Στο παρακάτω σχήμα φαίνεται ολόκληρη η διαδικασία φιλτραρίσματος και είναι γνωστή ως γρήγορη συνέλιξη. Στο σημείο αυτό σημειώνουμε πως ο πολλαπλασιασμός του  $X[k]$  με  $H[k]$  είναι μιγαδικός, περιλαμβάνοντας τη φάση και το πλάτος.



Σχήμα 2.1: Γρήγορη Συνέλιξη

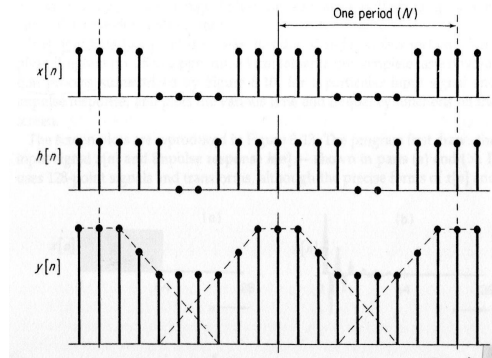
Οι FFT αλγόριθμοι προϋποθέτουν ότι οι συναρτήσεις που μετασχηματίζονται είναι περιοδικές, με περίοδο  $N$ . Επόμενο είναι λοιπόν ότι η γρήγορη συνέλιξη που παρουσιάζεται στο παραπάνω σχήμα παράγει ένα σήμα εξόδου  $y[n]$  το οποίο είναι η περιοδική, ή κυκλική, συνέλιξη των  $x[n]$  με  $h[n]$ . Την κυκλική συνέλιξη μπορούμε να τη φανταστούμε ως την τοποθέτηση των  $N$  δειγμάτων της μίας συνάρτησης πάνω στην περιφέρεια ενός κυλίνδρου, και τα  $N$  δείγματα της άλλης συνάρτησης με αντίστροφη σειρά πάνω σε ένα δεύτερο ομόκεντρο κύλινδρο. Ο ένας κύλινδρος περιστρέφεται, και οι συμπίπτουσες τιμές δειγμάτων πολλαπλασιάζονται και έπειτα προστίθενται. Το αποτέλεσμα είναι ότι η έξοδος που προκύπτει είναι επίσης περιοδική.

Στο ψηφιακό φιλτράρισμα χρειαζόμαστε γραμμική και όχι κυκλική συνέλιξη. Αυτό συμβαίνει γιατί θέλουμε να συνεχίσουμε ένα μη περιοδικό σήμα εισόδου και κρουστική απόκριση με αποτέλεσμα μία μη περιοδική έξοδο. Ευτυχώς φαίνεται πως μία περίοδος του σήματος εξόδου το οποίο προκύπτει με κυκλική συνέλιξη δίνει το σωστό αποτέλεσμα, με την προϋπόθεση ότι το μήκος των μετασχηματισμών έχει αυξηθεί με αρκετό zero filling.

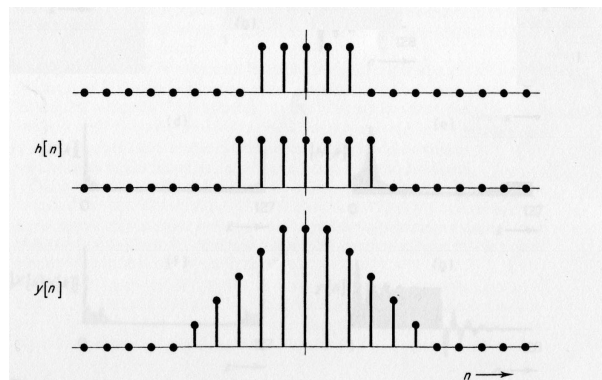
Το πρόβλημα και η λύση του φαίνονται στο επόμενο σχήμα. Το πρώτο σχήμα δείχνει δύο συναρτήσεις  $x[n]$  και  $h[n]$ , και την κυκλική τους συνέλιξη  $y[n] = x[n] * h[n]$  ( όπου  $*$  είναι το σύμβολο της συνέλιξης ). Υποθέτουμε πως όλες οι συναρτήσεις είναι περιοδικές. Δεν έχει ιδιαίτερη σημασία το σχήμα των  $x[n]$  και  $h[n]$ . Έχουν επιλεγεί τετραγωνικοί παλμοί γιατί είναι πιο βολικοί στην απεικόνιση. Ένας από αυτούς έχει πέντε πεπερασμένες τιμές ανά περίοδο, ο άλλος επτά, και η περίοδος είναι εννέα περιόδους δειγματοληψίας. Το  $y[n]$  μπορεί να βρεθεί αντιστρέφοντας είτε το  $x[n]$  είτε το  $h[n]$  – γεγονός το οποίο δεν κάνει διαφορά στην περίπτωση αυτή – στη συνέχεια πολλαπλασιάζοντας και τέλος αθροίζοντας τα δείγματα μίας περιόδου. Επομένως:

$$y[n] = x[n] * h[n] = \sum_{i=0}^{N-1} x[i]h[n-i], \quad 0 \leq n \leq (N-1)$$





Σχήμα 2.2: Κυκλική (περιοδική) Συνέλιξη



Σχήμα 2.3: Γραμμική (μη περιοδική) συνέλιξη

Παρατηρούμε πως σε αυτή την περίπτωση το  $y[n]$  αποτελείται από μία σειρά αλληλεπικαλυπτόμενων τριγώνων με επίπεδη κορυφή τα οποία φαίνονται με διακεκομμένη γραμμή στο σχήμα. Η αλληλεπικάλυψη συμβαίνει επειδή οι επαναλήψεις των  $x[n]$  και  $h[n]$  είναι κοντά με λίγες μηδενικές τιμές μεταξύ τους. Παρόλα αυτά, αν η περίοδος αυξηθεί με την εισαγωγή περισσότερων μηδενικών, η αλληλεπικάλυψη χάνεται. Κάθε περίοδος του  $y[n]$  τότε έχει την ίδια μορφή με αυτή που απαιτείται από τη γραμμική συνέλιξη μη επαναλαμβανόμενων εκδοχών του  $x[n]$  και του  $h[n]$ .

Μπορούμε εύκολα να δούμε ότι η αλληλεπικάλυψη στο παραπάνω σχήμα θα εξαφανιστεί αν η περίοδος αυξηθεί κατά τουλάχιστον  $(5 + 7 - 1) = 11$  περιόδους δειγματοληψίας μέσω zero-filling. Γενικά μπορούμε να αποφύγουμε την αλληλεπικάλυψη στην κυκλική συνέλιξη δύο συναρτήσεων με  $N_1$  και  $N_2$  δείγματα ανά περίοδο αντίστοιχα αν κάνουμε την περίοδο  $N \geq (N_1 + N_2 - 1)$ . Η μέθοδος της γρήγορης συνέλιξης που είδαμε στο σχήμα (... το πρώτο σχήμα της ενότητας ...) μας δίνει τότε το ζητούμενο αποτέλεσμα για το ψηφιακό φιλτράρισμα. (Αν θέλουμε να χρησιμοποιήσουμε έναν radix-2 FFT αλγόριθμο πρέπει τότε να κάνουμε zero-filling στις ακολουθίες  $x[n]$  και  $h[n]$  μέχρι το μήκος τους  $N$  να είναι ίσο με τη μικρότερη δύναμη του 2 η οποία εκπληρώνει τις παραπάνω προϋποθέσεις.) Ο τελευταίος αντίστροφος μετασχηματισμός δίνει μία ακολουθία δειγμάτων  $y[n]$  η οποία περιέχει  $(N_1 + N_2 - 1)$  αριθμό δειγμάτων ίσο με αυτό της γραμμικής συνέλιξης των  $x[n]$  και  $h[n]$ , ακολουθούμενη από  $N - (N_1 + N_2 - 1)$  μηδενικά.

Μπορούμε να κάνουμε μία εκτίμηση του πλεονεκτήματος ταχύτητας που έχει η γρήγορη συνέλιξη σε σύγκριση με τη συνέλιξη στο πεδίο του χρόνου ως εξής. Η γρήγορη συνέλιξη απαιτεί δύο N-σημείων FFTs, και ένα N-σημείων αντίστροφο FFT. Αυτές οι πράξεις περιλαμβάνουν περίπου  $3N\log_2 N$  μιγαδικούς πολλαπλασιασμούς και προσθέσεις/αφαιρέσεις. Επίσης, απαιτείται N αριθμός μιγαδικών πολλαπλασιασμών για να σχηματιστεί το γινόμενο των  $X[k]$  και  $H[k]$ . Μια απλή συνέλιξη στο πεδίο του χρόνου απαιτεί  $N_1 N_2$  πραγματικούς πολλαπλασιασμούς και προσθέσεις. Για λόγους απλότητας ας υποθέσουμε ότι οι πολλαπλασιασμοί καταναλώνουν τον περισσότερο από το χρόνο επεξεργασίας, και ότι ένας μιγαδικός πολλαπλασιασμός χρειάζεται τον διπλάσιο χρόνο για να γίνει από ένα πραγματικό. Αν οι  $N_1$  και  $N_2$  έχουν το μισό από το μήκος του μετασχηματισμού N, το πλεονέκτημα ταχύτητας της γρήγορης συνέλιξης θα είναι περίπου:

$$\frac{N_1 N_2}{2(3N\log_2 N + N)} = \frac{N^2/4}{2(3N\log_2 N + N)}$$

Τιμές αυτής της συνάρτησης δίνονται στον παρακάτω πίνακα για διαφορετικά μήκη μετασχηματισμών. Παρατηρούμε, για παράδειγμα, ότι η γρήγορη συνέλιξη είναι περίπου 4 φορές γρηγορότερη από την συνέλιξη στο πεδίο του χρόνου για σήματα 1024 σημείων.

Μήκος των μετασχηματισμών N	Αναμενόμενο πλεονέκτημα ταχύτητας για τη γρήγορη συνέλιξη
64	0,42
128	0,73
256	1,28
512	2,29
1024	4,13
2048	7,53
4096	13,8

*Πίνακας 2.1 Παράθεση τιμών του μήκους της κρουστικής απόκρισης με τις τιμές του μήκους του μετασχηματισμού.*

Στην πράξη συνήθως θέλουμε να επεξεργαστούμε πραγματικά σήματα και κρουστικές αποκρίσεις, στην οποία περίπτωση μόνο ο μισός από τον πλήρη μετασχηματισμό χρειάζεται να υπολογιστεί. Επιπλέον, το zero-filling που θα γίνει στις ακολουθίες  $x[n]$  και  $h[n]$  σημαίνει ότι μεγάλο μέρος των πολλαπλασιασμών του μετασχηματισμού δεν χρειάζεται να γίνει. Ιδιαίτερα αποδοτικοί αλγόριθμοι γρήγορης συνέλιξης λαμβάνουν αυτούς τους παράγοντες υπόψη και δίνουν πλεονεκτήματα ταχύτητας πάνω από διπλάσια αυτών που παρουσιάζονται στον παραπάνω πίνακα.

Για την εκτίμηση του πλεονεκτήματος ταχύτητας, υποθέσαμε ότι η γρήγορη συνέλιξη αντικαθίσταται

από μη αναδρομικό( non recursive ) φιλτράρισμα στο πεδίο του χρόνου. Φυσικά, αν είχαμε την επιλογή να χρησιμοποιήσουμε ένα αναδρομικό (recursive) ψηφιακό φίλτρο, το πλεονέκτημα θα το είχε η επεξεργασία στο πεδίο του χρόνου.

### 2.7.1 Τμηματοποίηση του σήματος

Σε αυτό το σημείο θα δούμε την πρακτική δυσκολία που προκύπτει όταν κάνουμε ψηφιακό φιλτράρισμα με τη γρήγορη συνέλιξη. Μέχρι τώρα υποθέσαμε ότι οι  $x[n]$  και  $h[n]$  έχουν περίπου τον ίδιο αριθμό δειγμάτων και ότι έχουν επιμηκυνθεί με μηδενικά μέχρι να έχουν το ίδιο μήκος μετασχηματισμού. Παρόλα αυτά, στην πράξη χρειαζόμαστε να συνελίξουμε ένα μακροσκελές σήμα εισόδου ή ακολουθία δεδομένων με μία σχετικά σύντομη κρουστική απόκριση. Για παράδειγμα, μπορεί να έχουμε ένα σήμα εισόδου με αρκετές χιλιάδες δείγματα εισόδου, αλλά μία κρουστική απόκριση με λιγότερα από εκατό δείγματα. Σε αυτή την περίπτωση δεν είναι οικονομικό από την άποψη χρόνου υπολογισμών να χρησιμοποιήσουμε το ίδιο μήκος μετασχηματισμού και για τις δύο συναρτήσεις. Επιπλέον, σε εφαρμογές πραγματικού χρόνου η χρήση ενός ιδιαίτερα μακροσκελούς μετασχηματισμού για το  $x[n]$  μπορεί να έχει ως συνέπεια μη αποδεκτές τιμές καθυστέρησης λόγω επεξεργασίας.

Το πρόβλημα αυτό μπορεί να λυθεί χωρίζοντας το σήμα εισόδου σε τμήματα μήκους που το καθιστούν εύχρηστο, πραγματοποιώντας γρήγορη συνέλιξη σε κάθε τμήμα και συνδυάζοντας τις εξόδους. Δύο προσεγγίσεις χρησιμοποιούνται πιο συχνά: Η μέθοδος επικάλυψης-πρόσθεσης, και η μέθοδος επικάλυψης-κράτησης. Θα αναπτύξουμε και τις δύο με τη σειρά.

Η μέθοδος επικάλυψης-πρόσθεσης φαίνεται στο παρακάτω σχήμα. Για λόγους απλότητας χρησιμοποιούμε πολύ σύντομους μετασχηματισμούς  $N = 8$ . Επίσης έχουμε επιλέξει ένα σήμα εισόδου και μία κρουστική απόκριση με μικρές ακέραιες τιμές δειγμάτων πράγμα που καθιστά εύκολο να ελεγχθούν οι τιμές της συνέλιξης.

Το σήμα εισόδου αρχικά χωρίζεται σε μη αλληλεπικαλυπτόμενα τμήματα  $x_0[n]$ ,  $x_1[n]$ ,  $x_2[n]$ , ..., περιέχοντας πέντε τιμές το καθένα. Επειδή η  $h[n]$  έχει μόλις τέσσερα δείγματα, η συνέλιξή του με καθένα από τα τμήματα παράγει μία ακολουθία εξόδου  $(5 + 4 - 1) = 8$  τιμών – το απαιτούμενο μήκος μετασχηματισμού. Οι διάφορες ακολουθίες εξόδου ή τμήματα, ονομάζονται  $y_0[n]$ ,  $y_1[n]$ ,  $y_2[n]$ , ... πάνω στο σχήμα. Σημειώνουμε ότι υπάρχει αλληλεπικαλύπτονται τριών δειγμάτων λόγω της επιμήκυνσης που προκαλεί η συνέλιξη. Η συνολική ακολουθία  $y[n]$  βρίσκεται από την πρόσθεση ή υπέρθεση των τμημάτων εξόδου.

Παρουσιάσαμε αυτή την διαδικασία σαν μία πράξη στο πεδίο του χρόνου. Στην πράξη όμως τα τμήματα εξόδου θα υπολογιστούν από την γρήγορη συνέλιξη με zero-filling στα τμήματα εισόδου και την κρουστική απόκριση μέχρι το επιθυμητό μήκος μετασχηματισμού.

Γενικά, η μέθοδος επικάλυψης-πρόσθεσης μπορεί να συνοψιστεί ως εξής. Έστω μία κρουστική απόκριση που περιέχει  $N_2$  μη μηδενικές τιμές, και ένα σήμα εισόδου που διαιρείται σε μη επικαλυπτόμενα

τιμήματα μήκους  $N_1$ , όπου  $N_1 < N_2$ . Το  $i$ -οστό τμήμα του  $x[n]$  δίνεται από:

$$x_i[n] = x[n], \quad i N_1 \leq n \leq (i + N_1) N_1 - 1$$

$$i = 0, \quad \text{αλλού}$$

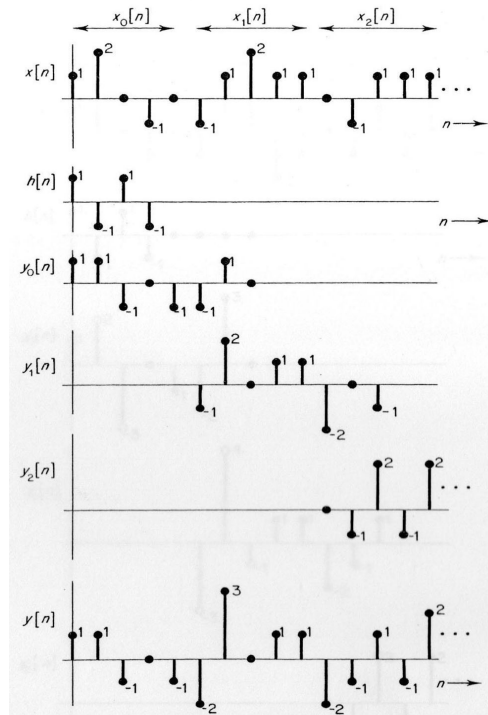
Τα ξεχωριστά τμήματα εξόδου είναι:

$$y_i[n] = x[n] * h[n] = \sum_{k=0}^{N_2-1} h[k] x_i[n-k]$$

Παρόλο που βρίσκονται από τον αντίστοιχο πολλαπλασιασμό στο πεδίο των συχνοτήτων. Η συνολική έξοδος δίνεται από:

$$y[n] = x[n] * h[n] = \sum_{i=0}^{N_2-1} h[i] x[n-i]$$

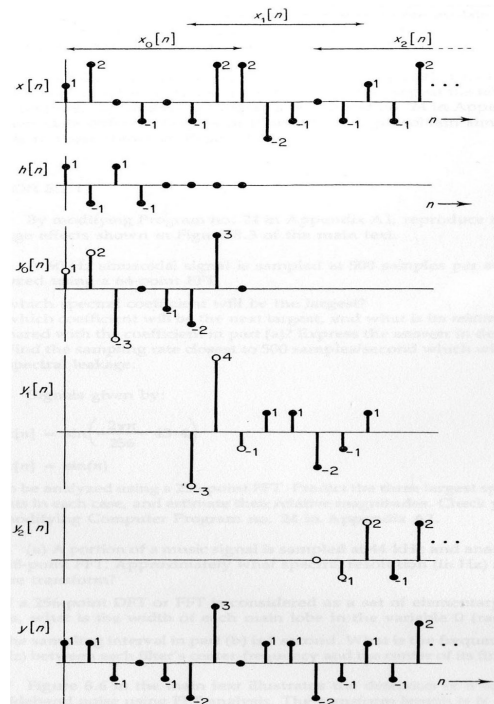
Όπως σημειώσαμε ήδη, το κάθε τμήμα της εξόδου  $y_i[n]$  υπολογίζεται με την γρήγορη συνέλιξη, δίνοντας μία ακολουθία με  $(N_1 + N_2 - 1)$  τιμές δειγμάτων. Η συνολική έξοδος βρίσκεται με την υπέρθεση, χρησιμοποιώντας μία επικάλυψη  $(N_2 - 1)$  σημείων μεταξύ των γειτονικών τμημάτων. Η εναλλακτική μέθοδος επικάλυψης-κράτησης, είναι μία παραλλαγή του ίδιου θέματος.



Σχήμα 2.4: Συνέλιξη με τη μέθοδο επικάλυψης-πρόσθεσης

Στην περίπτωση αυτή το σήμα εισόδου υποδιαιρείται σε αλληλεπικαλυπτόμενα τμήματα μήκους ίσου με αυτό του μετασχηματισμού ( $N$ ). Η αλληλεπικάλυψη είναι  $(N_2 - 1)$  σημείων, όπου  $N_2$  το μήκος της κρουστικής απόκρισης  $h[n]$ . Η τελευταία γεμίζεται με μηδενικά μέχρι το μήκος  $N$ . Η γρήγορη συνέλιξη του κάθε

τμήματος με  $h[n]$  δίνει τμήματα εξόδου στα οποία τα πρώτα  $(N_2 - 1)$  σημεία είναι λάθος, και αποβάλλονται. Οι τιμές που απομένουν συνδέονται με τα άλλα τμήματα της εξόδου για να δώσουν τη συνολική έξοδο  $y[n]$ . Στο σχήμα που ακολουθεί παρουσιάζεται η διαδικασία για το ίδιο σήμα εισόδου και την ίδια κρουστική απόκριση όπως και στο προηγούμενο σχήμα. Τα τμήματα της εισόδου έχουν μήκος 8 δειγμάτων, και αλληλεπικαλύπτονται σε  $(N_2 - 1) = 3$  τιμές δειγμάτων. Οι πρώτες τρεις τιμές δειγμάτων κάθε τμήματος εξόδου απορρίπτονται, και αυτά που απομένουν συνδέονται για να σχηματίσουν το σήμα εξόδου  $y[n]$ . Στην πράξη το κάθε τμήμα εξόδου υπολογίζεται με την τεχνική της γρήγορης συνέλιξης, και αντιπροσωπεύει μία περίοδο κυκλικής συνέλιξης.



Σχήμα 2.5: Συνέλιξη με τη μέθοδο overlap-save

Σημειώνουμε πως οι τρεις πρώτες τιμές της  $y[n]$  στο σχήμα δεν προκύπτουν από την  $y_0[n]$ . Θα τα βρούμε μέσω κυκλικής συνέλιξης μεταξύ του  $h[n]$  και προηγούμενου τμήματος εισόδου αυτού που ξεκινάει πριν το  $n = 0$  και αλληλεπικαλύπτεται με το  $x_0[n]$ .

Ποιό μήκος  $N$  μετασχηματισμού πρέπει να επιλέξουμε για γρήγορη συνέλιξη με τη μέθοδο επικάλυψης-πρόσθεσης ή τη μέθοδο επικάλυψης-κράτησης; Πρώτα, πρέπει να σιγουρευτούμε ότι το  $N$  είναι αρκετά μεγάλο για να δώσει το σωστό αποτέλεσμα η κυκλική συνέλιξη. Τι θα συμβεί αν

$$N \geq (N_1 + N_2 - 1) \text{ και } N_1 > N_2$$

Αν βάλουμε  $N_1 = (N_2 + 1)$ , τότε  $N = N \geq 2N_2$ . Οπότε το μήκος του μετασχηματισμού θα πρέπει να είναι τουλάχιστον διπλάσιο από αυτό της κρουστικής απόκρισης  $h[n]$ . Επιλογή του  $N = 2N_2$  σημαίνει πως πολλές σχετικά μικρές συνελίξεις θα πρέπει να πραγματοποιηθούν. Αντιθέτως αν το  $N$  είναι πολύ μεγαλύτερο

ρο, θα πρέπει να εκτελέσουμε ένα μικρό αριθμό από μακροσκελείς συνελίξεις. Έχει βρεθεί ότι ο καλύτερος συμβιβασμός εξαρτάται από το μήκος της κρουστικής απόκρισης, και ως κάποιο σημείο από τον αλγόριθμο γρήγορης συνέλιξης που χρησιμοποιείται. Προσεγγιστικές τιμές βρίσκονται στον παρακάτω πίνακα.

Μήκος της κρουστικής απόκρισης ( $N_2$ )	Μήκος μετασχηματισμού (N)
11 – 17	64
18 – 29	128
30 – 52	256
53 – 94	512
95 – 171	1024

*Πίνακας 2.2 Παράθεση τιμών του μήκους της κρουστικής απόκρισης με τις τιμές του μήκους του μετασχηματισμού.*

# ΚΕΦΑΛΑΙΟ 3<sup>ο</sup> : ΜΕΘΟΔΟΣ ΥΠΟΛΟΓΙΣΜΟΥ ΚΡΟΥΣΤΙΚΗΣ ΑΠΟΚΡΙΣΗΣ MLS

## Εισαγωγή 3ου Κεφαλαίου

Στο κεφάλαιο αυτό γίνεται αναφορά στις βασικές έννοιες της ακουστικής που αφορούν τη μέθοδο υπολογισμού της κρουστικής απόκρισης ενός χώρου με τη μέθοδο MLS. Σκοπός του κεφαλαίου αυτού είναι να περιγραφεί το θεωρητικό υπόβαθρο της διαδικασίας που θα ακολουθηθεί στη συνέχεια για τη λήψη των μετρήσεων.

## 3. Ακουστικές Μετρήσεις

### 3.1 Κρουστική απόκριση

Η κρουστική απόκριση ενός συστήματος ονομάζεται η απόκριση εξόδου του συστήματος όταν στην είσοδό του εφαρμοστεί ένας παλμός απειροστής διάρκειας και μοναδιαίου ύψους (με μαθηματικούς όρους μία συνάρτηση δέλτα  $\delta(t)$ ). Γνωρίζοντας την κρουστική απόκριση ενός γραμμικού και χρονικά αμετάβλητου συστήματος μπορούμε να υπολογίσουμε την απόκρισή του σε οποιοδήποτε σήμα εισόδου.

- Η απαίτηση της γραμμικότητας συνεπάγεται ότι η απόκριση του συστήματος όταν στην είσοδό του εφαρμοστούν ταυτόχρονα δύο σήματα ισούται με το άθροισμα των αποκρίσεων που θα προέκυπταν εάν τα δύο σήματα εφαρμόζονταν ξεχωριστά.
- Η απαίτηση της χρονικής χρονικής μη μεταβλητότητας του συστήματος συνεπάγεται ότι κάθε φορά που στην είσοδο του συστήματος εφαρμοστεί το ίδιο σήμα η έξοδος του θα είναι ταυτόσημη.

Υπό τις παραπάνω προϋποθέσεις η απόκριση του συστήματος,  $y(t)$ , σε ένα οποιοδήποτε σήμα εισόδου,  $x(t)$ , μπορεί να προκύψει με τα παρακάτω βήματα:

1. Διαιρούμε το σήμα εισόδου σε απειροστά τμήματα που απέχουν μεταξύ τους απόσταση  $\Delta T$  (εκτελούμε δηλαδή δειγματοληψία με συχνότητα δειγματοληψίας  $f_s = 1/\Delta T$ )
2. Θεωρώντας κάθε μεμονωμένο δείγμα ως ένα απειροστό παλμό, η απόκριση του συστήματος σε κάθε τέτοιο παλμό θα ισούται με την κρουστική του απόκριση πολλαπλασιασμένη με το πλάτος του παλμού.
3. Αθροίζοντας σε κάθε χρονική στιγμή την επίδραση από όλους τους προηγούμενους παλμούς (δείγματα) προκύπτει η συνολική απόκριση του συστήματος στη σύνθετη κυματομορφή εισόδου του συστήματος.

Η παραπάνω διαδικασία είναι η συνέλιξη, η οποία αναπτύχθηκε σε προηγούμενη ενότητα. Στη συνέχεια θα περιγραφεί ο τρόπος μέτρησης της κρουστικής απόκρισης ενός χώρου με τη μέθοδο MLS.

### 3.1.1 Μέτρηση της κρουστικής απόκρισης με τη μέθοδο MLS

Η μέθοδος της ακολουθίας μεγίστου μήκους (Maximum Length Sequence – MLS) αναπτύχθηκε τα τελευταία 15 χρόνια και έφερε κυριολεκτικά επανάσταση στον τρόπο πραγματοποίησης των ακουστικών μετρήσεων. Εφαρμόζεται σε κάθε γραμμικό και χρονικά αμετάβλητο σύστημα, ακουστικό, μηχανικό ή ηλεκτρονικό και βασίζεται στην διέγερση του συστήματος με μία εξειδικευμένη κατηγορία σημάτων εισόδου που ονομάζονται ακολουθίες μεγίστου μήκους και αναπτύχθηκαν στα πλαίσια της θεωρίας αριθμών. Στην αρχική της μορφή η μέθοδος αναπτύχθηκε για τη μέτρηση των χαρακτηριστικών των μεγαφώνων.

Μία ακολουθία μεγίστου μήκους αποτελείται από μία σειρά μοναδιαίων παλμών και παύσεων (μηδενικών) που εναλλάσσονται κατά τέτοιο τρόπο ώστε, σε κάθε χρονική στιγμή, η πιθανότητα να εμφανιστεί ένας θετικός παλμός ή παύση να είναι ίση (δηλαδή 50%) και ανεξάρτητη από το προηγούμενο δείγμα.

Τα βασικά χαρακτηριστικά της ακολουθίας μεγίστου μήκους είναι τα ακόλουθα:

- Παράγεται με αυστηρά καθορισμένο, ντετερμινιστικό τρόπο.(1)
- Ο αριθμός των μονάδων είναι ίσος με τον αριθμό των μηδενικών πλέον μίας (της αρχικής μονάδας).(2)
- Η συνάρτηση της αυτοσυσχέτισης (autocorrelation function) είναι μία συνάρτηση δέλτα.(3)
- Ο μετασχηματισμός Fourier της ακολουθίας περιέχει σταθερή ενέργεια ανά συχνότητα, όπως δηλαδή και μία δέλτα συνάρτηση ή ο λευκός θόρυβος (για το λόγο αυτό οι ακολουθίες μεγίστου μήκους ονομάζονται και ψευδοτυχαίος λευκός θόρυβος).(4)
- Είναι περιοδικό σήμα με περίοδο που εξαρτάται από ένα ακέραιο  $N$  ο οποίος ονομάζεται τάξη της ακολουθίας. Το μήκος (περίοδος) μίας ακολουθίας τάξης  $N$  ισούται προς:  $L = 2^N - 1$  τιμές. Αν ονομάσουμε  $x[n]$  την ακολουθία μεγίστου μήκους που εισάγεται στο σύστημα,  $y[n]$  το σήμα εξόδου του συστήματος,  $h[n]$  την κρουστική απόκριση του συστήματος  $R_{xx}$  την αυτοσυσχέτιση και με  $*$  την πράξη της συνέλιξης, τότε από τις ιδιότητες 3 και 4 προκύπτουν τα παρακάτω:

$$R_{xx}[T=kt_0] = \sum x[nt_0 - kt_0]x[nt_0] = \delta(t - kt_0)$$

$$y[n] = (h * x)[n] \Leftrightarrow R_{yx}[n] = (h * R_{xx})[n] \Leftrightarrow R_{yx}[n] = h[n] \text{ για } R_{xx} = \delta[n]$$

Δηλαδή η κρουστική απόκριση του συστήματος ισούται με την ετεροσυσχέτιση (cross-correlation) της εισόδου με την έξοδο του συστήματος. Αυτή ακριβώς η ιδιότητα των ακολουθιών μεγίστου μήκους τις κατέστησε το βασικό σήμα εισόδου για τη μέτρηση της απόκρισης οποιουδήποτε γραμμικού και χρονικά αμετάβλητου συστήματος.



Η βασική διαφοροποίηση της μεθόδου MLS με τη διέγερση του συστήματος με λευκό θόρυβο είναι ότι σε αυτή την περίπτωση όλοι οι υπολογισμοί λαμβάνουν χώρα στο πεδίο του χρόνου και δεν παρεμβάλλεται κανένας μετασχηματισμός Fourier περιορίζοντας έτσι σημαντικά τα πιθανά σφάλματα. Επίσης, επειδή το σήμα εισόδου είναι 100% ντετερμινιστικό (δηλαδή σε κάθε χρονική στιγμή γνωρίζουμε ακριβώς την τιμή που εισάγεται στο σύστημα), με τη διαδικασία της ετεροσυσχέτισης αποκλείονται από την μέτρηση όλα τα συμβάντα που δε συνδέονται γραμμικά με την είσοδο του συστήματος (λ.χ. τυχαίος θόρυβος). Κατ' αυτό τον τρόπο επιτυγχάνεται μία πολύ σημαντική αύξηση του λόγου σήματος προς θόρυβο και είναι δυνατή η πραγματοποίηση αξιόπιστων μετρήσεων σε περιβάλλον με υψηλό θόρυβο βάθους.

Βασικός περιορισμός που πρέπει πάντα να λαμβάνεται υπόψη σε μετρήσεις με ακολουθία μεγίστου μήκους είναι ότι το μήκος της ακολουθίας που θα χρησιμοποιηθεί θα πρέπει να είναι μεγαλύτερο από το μήκος της προς μέτρηση απόκρισης. Όπως ισχύει και για μετρήσεις με λευκό θόρυβο, διπλασιασμός του αριθμού των μετρήσεων (επαναλήψεων) συνεπάγεται διπλασιασμό του λόγου σήματος προς θόρυβο (3dB).

### 3.2 Εφαρμογή Αντήχησης μέσω συνέλιξης (Reverberation via Convolution)

Ένας ακριβής αλλά υπολογιστικά απαιτητικός τρόπος εξομοίωσης της αντήχησης σε ένα δεδομένο χώρο είναι να γίνει συνέλιξη της κρουστικής απόκρισης του χώρου με το σήμα που θέλουμε να ακουστεί μέσα στο χώρο αυτό. Μπορούμε να σκεφτούμε την αντήχηση σαν ένα τύπο φίλτρου όπου το μήκος (σε δείγματα) της κρουστικής απόκρισης αντιστοιχεί στο χρόνο αντήχησης (σε δείγματα) του υπό εξομοίωση χώρου. Η κρουστική απόκριση ενός χώρου μπορεί να μετρηθεί ηχογραφώντας την απόκριση του χώρου όταν αυτός διεγείρεται από ένα πολύ σύντομο και δυνατό ήχο. Αυτό το σύνολο δειγμάτων στη συνέχεια γίνεται συνέλιξη με το σήμα που θέλουμε να τοποθετήσουμε στο χώρο αυτό.

Παραπάνω έχουμε κάνει τη διάκριση μεταξύ απ' ευθείας και γρήγορης συνέλιξης. Η απ' ευθείας συνέλιξη δεν είναι πρακτική για την εφαρμογή αντήχησης λόγω του τεράστιου όγκου υπολογισμών που απαιτεί. Για παράδειγμα, με συχνότητα δειγματοληψίας 48kHz και μήκος κρουστικής απόκρισης 3 δευτερόλεπτα, κάθε δείγμα σε κάθε κανάλι του σήματος εισόδου πρέπει να πολλαπλασιαστεί και να αθροιστεί 48000 x 3 φορές. Για ένα δευτερόλεπτο σήματος εισόδου τα παραπάνω σημαίνουν

144,000	x	48,000	=	6,912,000,000
Πολλαπλασιασμοί/ προσθέσεις ανά δείγμα (κρουστική απόκριση)		Δείγματα ανά δεύτ.		Πολλαπλασιασμοί/ προσθέσεις ανά δείγμα ανά κανάλι

Επομένως για να εξομοιώσουμε την αντήχηση ενός δευτερολέπτου ενός stereo σήματος με απ' ευθείας

συνέλιξη απαιτούνται 13.824 δισεκατομμύρια πολλαπλασιασμοί/προσθέσεις. Ο υπολογισμός αυτός σε πραγματικό χρόνο απαιτεί επίπεδα απόδοσης που ξεπερνούν αυτά των κοινών προσωπικών υπολογιστών. Με ένα επεξεργαστή σήματος ο οποίος εκτελεί 100 εκατομμύρια πολλαπλασιασμούς/προσθέσεις το δευτερόλεπτο σε μία πρακτική εφαρμογή, όπως σε ένα plugin board για ένα προσωπικό υπολογιστή, αυτός ο υπολογισμός θα πάρει περίπου δύο λεπτά και δεκαοκτώ δευτερόλεπτα για να γίνει, ένας λόγος 138:1 συγκρινόμενο με μία real time υλοποίηση.

Οπότε ο μόνος πρακτικός τρόπος εφαρμογής αντήχησης με συνέλιξη είναι μέσω της γρήγορης συνέλιξης, αξιοποιώντας το πλεονέκτημα ταχύτητας που προσφέρει ο γρήγορος μετασχηματισμός Fourier (FFT).

# ΚΕΦΑΛΑΙΟ 4<sup>ο</sup> : ΜΕΘΟΔΟΣ ΠΡΟΣΟΜΟΙΩΣΗΣ ΤΟΥ ΦΥΣΙΚΟΥ ΑΚΟΥΣΤΙΚΟΥ ΠΕΔΙΟΥ ΣΕ ΑΚΟΥΣΤΙΚΑ

## Εισαγωγή 4ου Κεφαλαίου

Στο κεφάλαιο αυτό, αναλύεται η μέθοδος που θα χρησιμοποιηθεί για την προσομοίωση της ακρόασης από ηχεία σε φυσικό ακουστικό χώρο σε ακουστικά. Σκοπός αυτού του κεφαλαίου είναι να παρουσιαστεί ο τρόπος με τον οποίο συνδυάζονται τα περιεχόμενα των προηγούμενων κεφαλαίων για την επίτευξη του σκοπού της εργασίας αυτής.

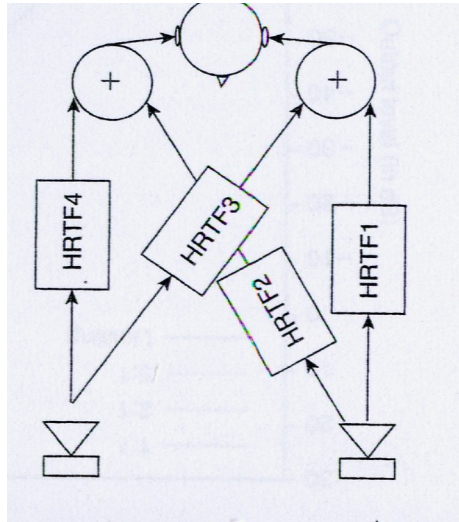
### 4.1 Μέθοδος

Με το σχήμα στο τέλος της παραγράφου σαν οδηγό, στη συνέχεια θα αναλυθεί η μέθοδος προσομοίωσης του φυσικού ακουστικού πεδίου σε ακουστικά. Στο παραπάνω σχήμα φαίνεται τι ακριβώς φτάνει στο κάθε αυτί από το κάθε ηχείο υπό συνθήκες ακρόασης με ηχεία. Όπως φαίνεται στο σχήμα, το σήμα του αριστερού αυτιού είναι η υπέρθεση του σήματος από το αριστερό ηχείο διαμορφωμένο από τη συνάρτηση μεταφοράς της κεφαλής για ακρόαση με το αριστερό αυτί από το αριστερό ηχείο, με αυτό του σήματος από το δεξί ηχείο διαμορφωμένο από τη συνάρτηση μεταφοράς της κεφαλής για ακρόαση με το αριστερό αυτί από το δεξί ηχείο. Αντίστοιχα, το σήμα του δεξιού αυτιού είναι η υπέρθεση του σήματος από το αριστερό ηχείο διαμορφωμένο από τη συνάρτηση μεταφοράς της κεφαλής για ακρόαση με το δεξί αυτί από το αριστερό ηχείο, με αυτό του σήματος από το δεξί ηχείο διαμορφωμένο από τη συνάρτηση μεταφοράς της κεφαλής για ακρόαση με το δεξί αυτί από το δεξί ηχείο.

Σύμφωνα με τα παραπάνω για να επιτευχθεί η προσομοίωση θα πρέπει αρχικά να γίνουν τέσσερις μετρήσεις κρουστικής απόκρισης, δύο για κάθε αυτί. Η κάθε μία από τις κρουστικές αποκρίσεις που θα μετρηθούν θα περιέχει τη συνάρτηση μεταφοράς της κεφαλής για ακρόαση από τη συγκεκριμένη θέση με το συγκεκριμένο αυτί από το συγκεκριμένο ηχείο, καθώς και την απόκριση του συγκεκριμένου χώρου ακρόασης στη συγκεκριμένη θέση. Στη συνέχεια αφού πραγματοποιηθούν οι μετρήσεις, γίνεται συνέλιξη του σήματος του κάθε ηχείου για κάθε διαδρομή με την αντίστοιχη κρουστική απόκριση και το αποτέλεσμα της παραπάνω διαδικασίας όταν αναπαραχθεί από ακουστικά θα πρέπει να δίνει την εντύπωση στον ακροατή ότι ακούει το υλικό που αναπαράγεται όχι από τα ακουστικά αλλά από τα ηχεία μέσα στον χώρο που μετρήθηκε από τη συγκεκριμένη θέση ακρόασης.

Οι μετρήσεις θα γίνουν με τη μέθοδο μέτρησης της κρουστικής απόκρισης MLS όπως αυτή περιγράφεται

στην αντίστοιχη ενότητα του προηγούμενου κεφαλαίου. Για τη συνέλιξη των σημάτων με τις αποκρίσεις θα χρησιμοποιηθεί η μέθοδος που περιγράφεται στην ενότητα 3.5 “Ψηφιακό φιλτράρισμα με τη χρήση της γρήγορης συνέλιξης” λόγω του πλεονεκτήματος ταχύτητας που παρουσιάζει στην επεξεργασία των δεδομένων σε σύγκριση με την απλή συνέλιξη στο πεδίο του χρόνου.



Σχημα 4.1 : Απεικόνιση του τρόπου επίδρασης των συναρτήσεων μεταφοράς της κεφαλής στην ακρόαση

## Πειραματικό Μέρος

# ΚΕΦΑΛΑΙΟ 5<sup>ο</sup> : ΔΙΑΔΙΚΑΣΙΑ ΜΕΤΡΗΣΕΩΝ

## ΜΕΘΟΔΟΛΟΓΙΑ

### Εισαγωγή 5ου Κεφαλαίου

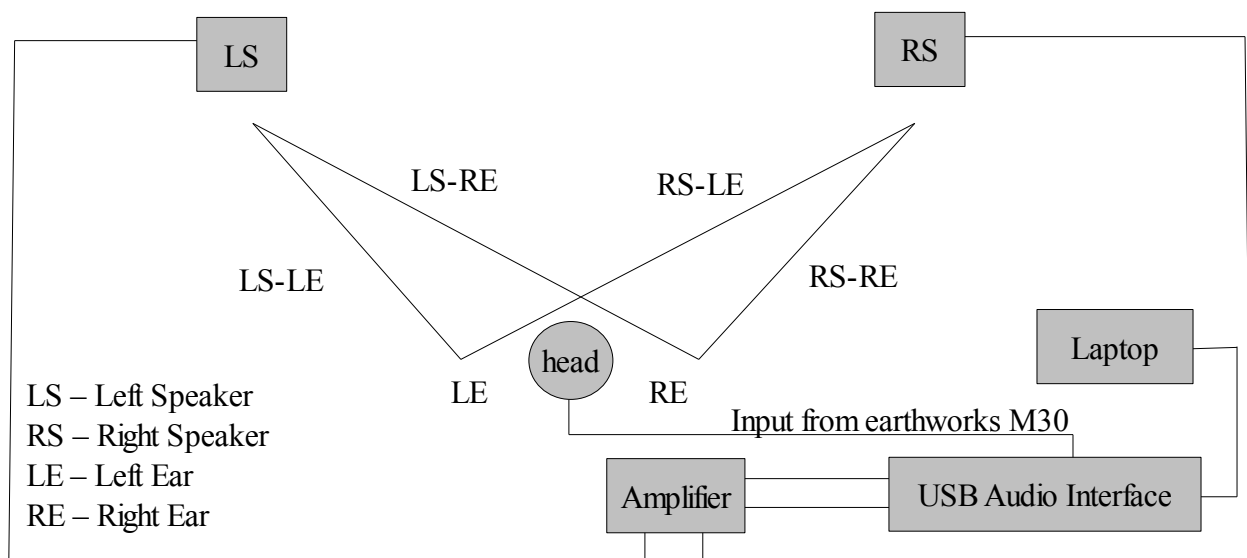
Στο κεφάλαιο αυτό, αναλύεται ο τρόπος με τον οποίο σχεδιάστηκε και διεξήχθη η πειραματική διαδικασία. Σκοπός αυτού του κεφαλαίου είναι η κατανόηση της διαδικασίας λήψης των μετρήσεων σύμφωνα με όσα έχουν αναφερθεί στο θεωρητικό μέρος.

### 5.1 Διάταξη

Για τη μέτρηση των τεσσάρων κρουστικών αποκρίσεων χρησιμοποιήθηκαν τα εξής παρακάτω:

- Μικρόφωνο Earthworks M30
- Κεφάλι από κούκλα
- Ενισχυτής
- Laptop ( Pentium(R) Dual-Core CPU T4200 @ 2.00 GHz 2.00 GHz  
4.00GB RAM)
- Audio Interface Tascam US-144
- Ηχεία ηχοσυστήματος εργαστηρίου εικόνα/ήχος
- Λογισμικό Smart acoustic tools της Rational Acoustics LLC

Η διάταξη είχε ως εξής:



## 5.2 Διαδικασία

Όπως φαίνεται στο παραπάνω σχήμα, το ηχείο που χρειάζεται κάθε φορά, τροφοδοτήθηκε με MLS σήμα για να κάνουμε την μέτρηση της κρουστικής απόκρισης από το αυτί που χρειάζεται κάθε φορά. Το MLS σήμα που χρησιμοποιήθηκε παρήχθη από την κάρτα με συχνότητα δειγματοληψίας 44100Hz. Στη συνέχεια παρουσιάζουμε τη διαδικασία αναλυτικά για κάθε μία από τις τέσσερις αποκρίσεις.

### LSLE ( Left Speaker Left Ear – Αριστερό Ηχείο Αριστερό Αυτί )

Για να λάβουμε την LSLE απόκριση, τοποθετούμε το μικρόφωνο στην τρύπα που έχει το κεφάλι για το αριστερό αυτί, από μέσα, τροφοδοτούμε το αριστερό ηχείο με MLS σήμα, ρυθμίζουμε τη στάθμη του σήματος από την κάρτα ήχου έτσι ώστε να έχουμε όσο το δυνατόν υψηλότερης στάθμης σήμα χωρίς να παραμορφώνει η είσοδος και αφήνουμε σταθερό το gain της προενίσχυσης καθ' όλη τη διάρκεια των μετρήσεων. Ρυθμίζουμε το λογισμικό να κάνει αρκετά περάσματα του MLS σήματος και πραγματοποιούμε τη μέτρηση. Αφού ολοκληρωθεί η μέτρηση σώζουμε την κρουστική απόκριση σε ένα wav αρχείο με συχνότητα δειγματοληψίας 44100 Hz και εύρος bit 16 και με το όνομα LSLE.wav στον ίδιο φάκελο με το m file που περιέχει τον κώδικα που θα χρησιμοποιήσει τις αποκρίσεις.

### RSLE ( Right Speaker Left Ear – Δεξί Ηχείο Αριστερό Αυτί )

Για να λάβουμε την RSLE απόκριση, τοποθετούμε το μικρόφωνο στην τρύπα που έχει το κεφάλι για το αριστερό αυτί, από μέσα, τροφοδοτούμε το δεξί ηχείο με MLS σήμα. Ρυθμίζουμε το λογισμικό να κάνει αρκετά περάσματα του MLS σήματος για να πετύχουμε καλό λόγο σήματος προς θόρυβο αλλά και για να αποφύγουμε να εισέλθουν στη μέτρηση εξωγενείς ήχοι που τυχόν ακούστηκαν την ώρα της μέτρησης, και πραγματοποιούμε τη μέτρηση. Αφού ολοκληρωθεί η μέτρηση σώζουμε την κρουστική απόκριση σε ένα wav αρχείο με συχνότητα δειγματοληψίας 44100 Hz και εύρος bit 16 και με το όνομα RSLE.wav στον ίδιο φάκελο με το m file που περιέχει τον κώδικα που θα χρησιμοποιήσει τις αποκρίσεις.

### LSRE ( Left Speaker Right Ear – Αριστερό Ηχείο Δεξί Αυτί )

Για να λάβουμε την LSRE απόκριση, τοποθετούμε το μικρόφωνο στην τρύπα που έχει το κεφάλι για το δεξί αυτί, από μέσα, τροφοδοτούμε το αριστερό ηχείο με MLS σήμα. Ρυθμίζουμε το λογισμικό να κάνει αρκετά περάσματα του MLS σήματος για να πετύχουμε καλό λόγο σήματος προς θόρυβο αλλά και για να αποφύγουμε να εισέλθουν στη μέτρηση εξωγενείς ήχοι που τυχόν ακούστηκαν την ώρα της μέτρησης, και πραγματοποιούμε τη μέτρηση. Αφού ολοκληρωθεί η μέτρηση σώζουμε την κρουστική απόκριση σε ένα wav αρχείο με συχνότητα δειγματοληψίας 44100 Hz και εύρος bit 16 και με το όνομα LSRE.wav στον ίδιο φάκελο με το m file που περιέχει τον κώδικα που θα χρησιμοποιήσει τις αποκρίσεις.

### RSRE ( Right Speaker Right Ear – Δεξί Ηχείο Δεξί Αυτί )

Για να λάβουμε την RSRE απόκριση, τοποθετούμε το μικρόφωνο στην τρύπα που έχει το κεφάλι για το

δεξί αυτί, από μέσα, τροφοδοτούμε το δεξί ηχείο με MLS σήμα. Ρυθμίζουμε το λογισμικό να κάνει αρκετά περάσματα του MLS σήματος για να πετύχουμε καλό λόγο σήματος προς θόρυβο αλλά και για να αποφύγουμε να εισέλθουν στη μέτρηση εξωγενείς ήχοι που τυχόν ακούστηκαν την ώρα της μέτρησης, και πραγματοποιούμε τη μέτρηση. Αφού ολοκληρωθεί η μέτρηση σώζουμε την κρουστική απόκριση σε ένα wav αρχείο με συχνότητα δειγματοληψίας 44100 Hz και εύρος bit 16 και με το όνομα RSRE.wav στον ίδιο φάκελο με το m file που περιέχει τον κώδικα που θα χρησιμοποιήσει τις αποκρίσεις.

Τα wav αρχεία που προέκυψαν από τις παραπάνω μετρήσεις θα διαβαστούν από το πρόγραμμα που θα αναλυθεί παρακάτω μέσω της συνάρτησης του MATLAB wavread.



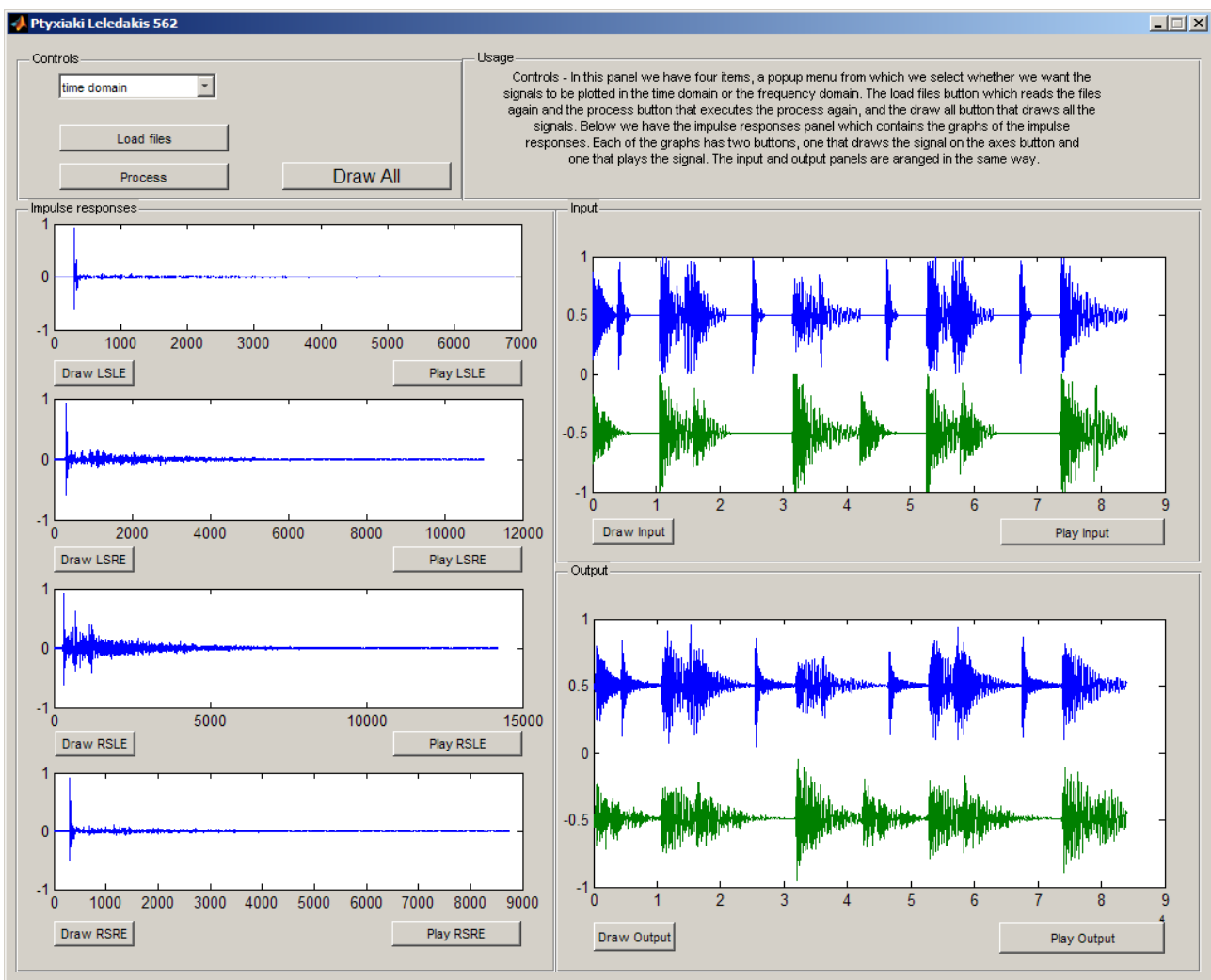
# ΚΕΦΑΛΑΙΟ 6<sup>ο</sup>: ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΜΕ ΤΗ ΧΡΗΣΗ ΤΟΥ MATLAB

## Εισαγωγή 6ου Κεφαλαίου

Στο κεφάλαιο αυτό παρουσιάζεται και αναλύεται ο τρόπος δημιουργίας της εφαρμογής και ο πηγαίος κώδικάς της. Σκοπός του κεφαλαίου αυτού είναι να γίνουν κατανοητά τα βήματα που ακολουθήθηκαν για τη δημιουργία της εφαρμογής καθώς επίσης και ο πηγαίος κώδικας της εφαρμογής.

## 6.1 Παρουσίαση/Ανάλυση Κώδικα και Εφαρμογής

Η εφαρμογή που θα αναλυθεί παρακάτω, όταν ξεκινήσει εμφανίζεται όπως φαίνεται παρακάτω



Εικόνα 5.1: Γραφική Διεπαφή Χρήστη της εφαρμογής

Την ανάλυσή μας θα την ξεκινήσουμε από την περιγραφή της διεπαφής χρήστη.

## 6.2 Γραφική διεπαφή χρήστη και περιγραφή χρήσης

Με μία πρώτη ματιά στην διεπαφή μπορούμε να δούμε ότι αποτελείται από τέσσερα πάνελ. Το πρώτο πάνω δεξιά είναι το πάνελ με τα χειριστήρια ( Controls ). Αυτό περιέχει τα εξής τρία κουμπιά και ένα πτυσσόμενο μενού( pop-up menu )

**pop-up** – Περιέχει δύο στοιχεία, το ένα είναι το “**time domain**” και το άλλο είναι το “**frequency domain**”. Η επιλογή αυτή αφορά το πώς θα απεικονίζονται τα σήματα στα γραφήματα, στο πεδίο του χρόνου ή στο πεδίο των συχνοτήτων

**load files**– φορτώνει τα wav αρχεία ( αρχεία με τις αποκρίσεις και το αρχείο με το σήμα εισόδου )

**process** – το κουμπί αυτό, όπως δηλώνει και το όνομά του, πατώντας το εκτελούμε τη συνέλιξη των κρουστικών αποκρίσεων με το σήμα εισόδου όπως θα δούμε παρακάτω πιο αναλυτικά.

**Draw All** – πατώντας το κουμπί αυτό ξαναζωγραφίζουμε όλα τα διαγράμματα.

Στο διπλανό πάνελ ( Usage ) περιγράφεται σύντομα η χρήση της εφαρμογής μέσω του παρακάτω κειμένου. “Controls - In this panel we have four items, a popup menu from which we select whether we want the signals to be plotted in the time domain or the frequency domain. The load files button which reads the files again and the process button that executes the process again, and the draw all button that draws all the signals. Below we have the impulse responses panel which contains the graphs of the impulse responses. Each of the graphs has two buttons, one that draws the signal on the axes button and one that plays the signal. The input and output panels are arranged in the same way.” . Χρησιμοποιήθηκε η αγγλική γλώσσα, επειδή το guide δεν έπαιρνε καλά τα ελληνικά( δεν υποστηρίζονται οι τόνοι ), και για το λόγο αυτό, για να είναι το αποτέλεσμα πιο πλήρες, χρησιμοποιήθηκαν τα αγγλικά.

Στο κάτω αριστερά πάνελ ( Impulse Responses ) βλέπουμε τέσσερα γραφήματα με το καθένα να έχει από κάτω του δύο κουμπιά. Σε αυτά παρουσιάζονται οι κρουστικές αποκρίσεις, η κάθε μία στο γράφημά της με δύο κουμπιά κάτω από την κάθε μία. Το πρώτο κουμπί Draw LSLE για παράδειγμα ξαναζωγραφίζει το γράφημα LSLE. Το δεύτερο κουμπί Play LSLE αναπαράγει στην έξοδο της κάρτας ήχου του υπολογιστή την κρουστική LSLE.

Στο τέταρτο και τελευταίο πάνελ, βλέπουμε όπως και πριν δύο γραφήματα. Στα γραφήματα αυτά παρουσιάζονται το σήμα εισόδου και το σήμα εξόδου αντίστοιχα. Τα δύο κουμπιά που βλέπουμε κάτω από το κάθε

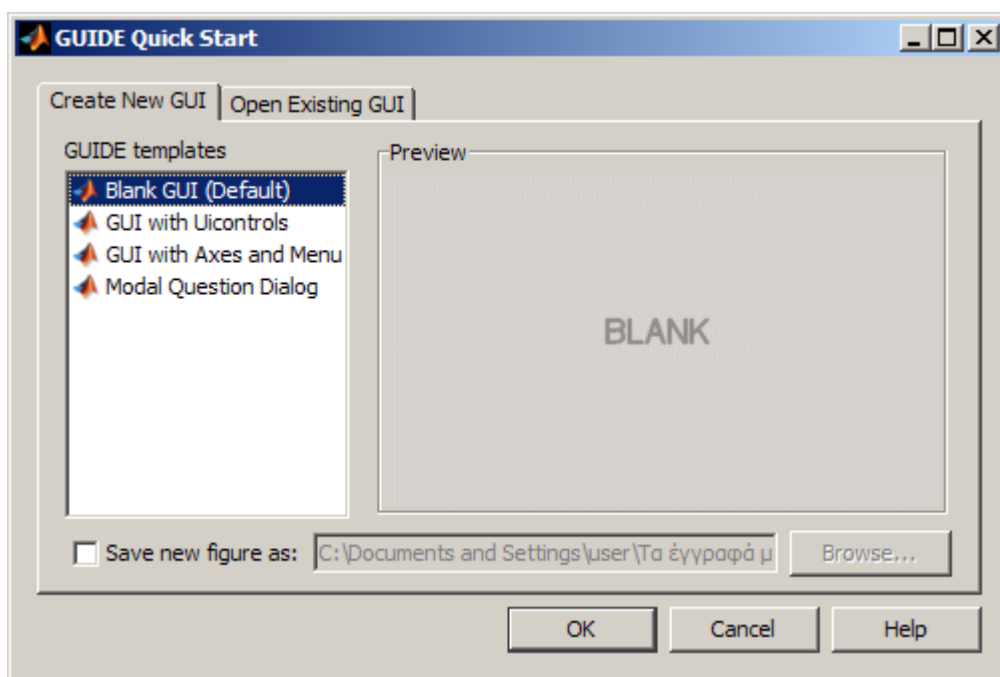
πάνελ εκτελούν την ίδια αντίστοιχη λειτουργία με αυτή που εκτελούν τα κουμπιά στο Impulse Responses πάνελ.

### 6.3 Περιγραφή του τρόπου δημιουργίας της εφαρμογής μέσω του MATLAB

Για να δημιουργήσουμε μία αυτόνομη εφαρμογή με το MATLAB πρέπει αρχικά να εγκαταστήσουμε τον MCR ( MATLAB Compiler Runtime ), η διαδικασία εγκατάστασης του οποίου περιγράφεται αναλυτικά στα αρχεία βοήθειας του MATLAB. Στη συνέχεια εκτελούμε την εντολή **deploytool** από το command window του MATLAB. Εμφανίζεται ένα παράθυρο διαλόγου στο όπου επιλέγουμε windows standalone application. Στην εφαρμογή που θα δημιουργήσουμε δίνουμε όνομα ίδιο με αυτό του m-file ( και της function ). Σε περίπτωση που δεν έχει επιλεγεί κάποιος compiler, τρέχουμε από το command window του MATLAB την εντολή **mbuild -setup** και επιλέγουμε κάποιον από αυτούς που εμφανίζονται στη λίστα που μας δίνει η παραπάνω εντολή. Τέλος κάνουμε build.

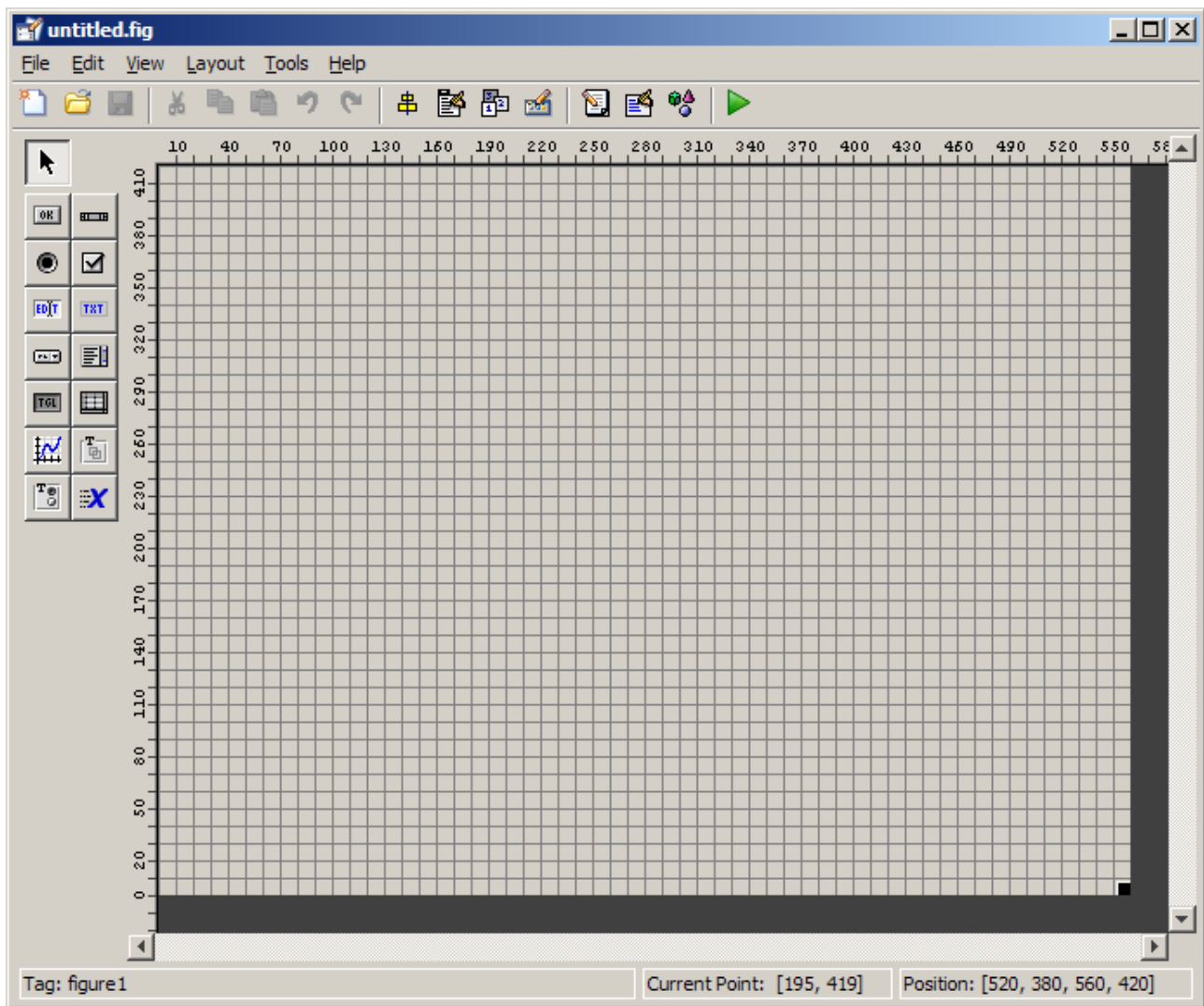
#### 6.3.1 Δημιουργία της διεπαφής χρήστη μέσω του guide

Για να δημιουργήσουμε τη διεπαφή χρήστη τρέχουμε στο command window του MATLAB την εντολή **guide**. Όταν πατήσουμε enter εμφανίζεται το παρακάτω παράθυρο διαλόγου.



*Εικόνα 5.4: GUIDE Quick start wizard*

Επιλέγουμε Blank GUI και πατάμε OK. Όταν πατήσουμε OK εμφανίζεται το παράθυρο του guide όπου θα δουλέψουμε.

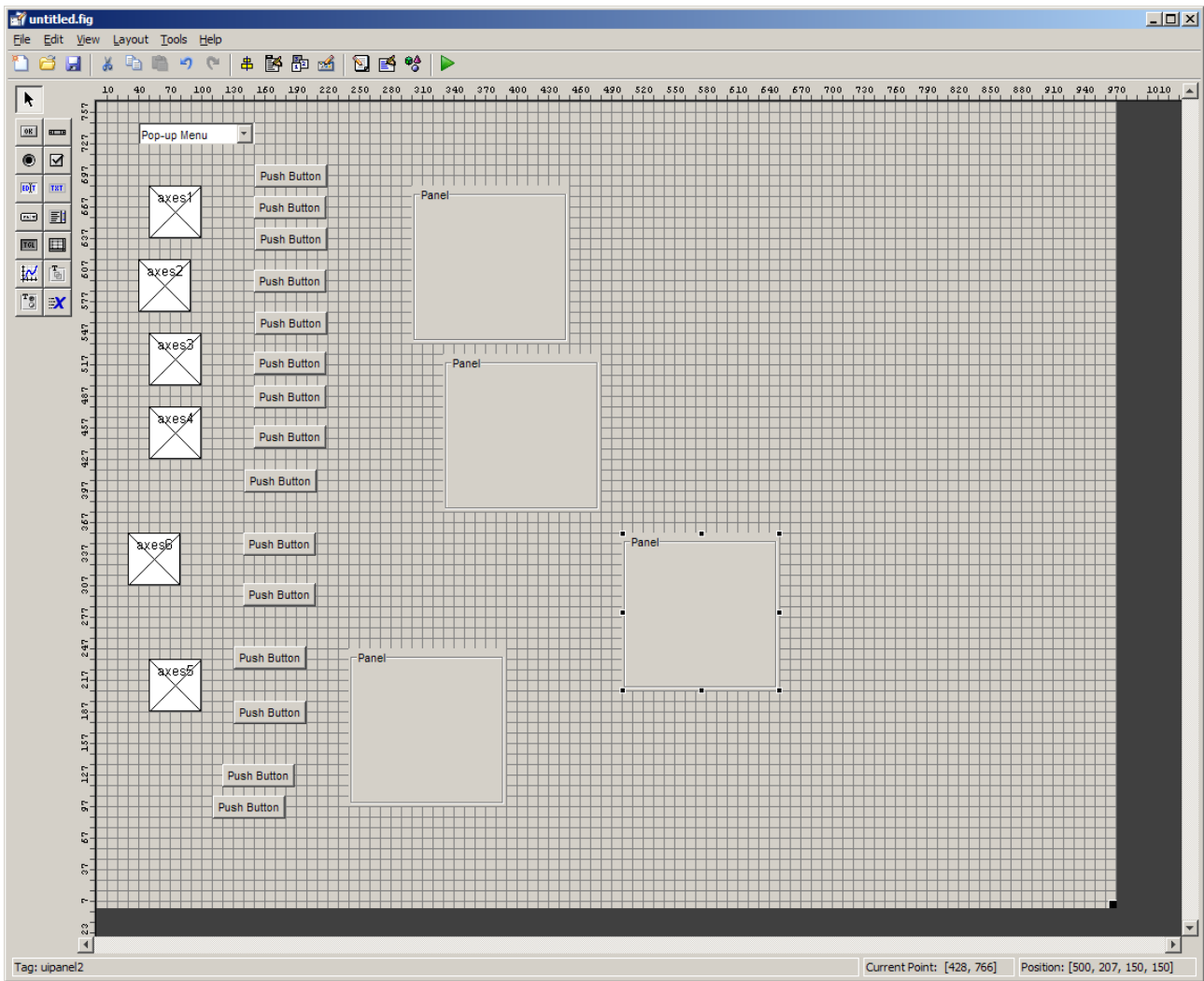


Εικόνα 5.5: Επιφάνεια εργασίας του GUIDE

Στο παραπάνω σχήμα βλέπουμε την επιφάνεια εργασίας του GUIDE. Στα αριστερά βλέπουμε μία γραμμή εργαλείων με αντικείμενα που μπορούμε να σύρουμε μέσα στην επιφάνεια εργασίας και να αποτελέσουν έτσι τα στοιχεία της γραφικής διεπαφής χρήστη της εφαρμογής μας.

Για να δημιουργήσουμε την εφαρμογή μας, θα εισάγουμε στην επιφάνεια εργασίας 6 αντικείμενα **axes**, 15 **Push Buttons**, 4 **panels**, 1 **pop-up menu** και τέλος ένα αντικείμενο **static text** όπως φαίνονται στην παρακάτω εικόνα.

Τα τοποθετούμε σε μία διάταξη όπως αυτή του σχήματος 5.1 και ρυθμίζουμε τα χαρακτηριστικά τους μέσω του property inspector. Τέλος όταν ρυθμίσουμε όλα τα στοιχεία της διεπαφής έτσι όπως τα θέλουμε, μπορούμε να προχωρήσουμε στην συγγραφή κώδικα για να γίνουν λειτουργικά τα στοιχεία του GUI.



Εικόνα 5.6: Η επιφάνεια εργασίας του guide με τα στοιχεία που θα χρησιμοποιήσουμε για τη δημιουργία της διεπαφής χρήστη

### 6.3.2 Συγγραφή κώδικα για τη διεπαφή που δημιουργήθηκε μέσω του guide

Στη συνέχεια θα δημιουργηθεί μέσω του guide το m-file που θα είναι το κύριο αρχείο της εφαρμογής. Αφού **δημιουργηθεί** το m-file, ανοίγει ο editor με το αρχείο που περιέχει τον κώδικα που έχει **παράγει αυτόματα** το guide σύμφωνα με **ότι έχουμε** σχεδιάσει **νωρίτερα**.

#### 6.3.2.1 Ptychiaki\_Leledakis\_562\_OpeningFcn

Η Παρακάτω συνάρτηση είναι η συνάρτηση που τρέχει όταν ανοίγει το πρόγραμμα και περιέχει τις αρχικοποιήσεις μεταβλητών και διαδικασίες που γίνονται κατά την εκκίνηση. Στην περίπτωση του προγράμματός αυτού, όταν αυτό ανοίγει όπως φαίνεται και αναλυτικά παρακάτω, αρχικοποιούνται όλες οι μεταβλητές που χρησιμοποιούνται και εκτελείται η επεξεργασία στα σήματα που υπάρχουν ήδη μέσα στον φάκελο με το εκτελέσιμο.

Ξεκινώντας, φτιάχνουμε κάποιες μεταβλητές που αφορούν τα γενικά χαρακτηριστικά του προγράμματός. Το αναγνωριστικό handles μπαίνει σε κάθε μεταβλητή γιατί έτσι σχετίζεται κάθε μία από αυτές με το GUI( Γραφική Διεπαφή Χρήστη) εξηγείται αναλυτικά στα σχόλια παραπάνω. Εν ολίγοις οποιαδήποτε μεταβλητή για να συσχετιστεί με δεδομένα από το GUI πρέπει να ανοίκει στη δομή ( structure όπως στη γλώσσα προγραμματισμού C ) handles.

```
handles.isTimeDomainSelected = 1;
```

αυτή η μεταβλητή χρησιμοποιείται από την draw\_all\_button\_Callback και χρησιμοποιείται στον καθορισμό του τι θα σχεδιαστεί στα γραφήματα.

```
handles.outputGain = 0.9;
```

Εδώ ρυθμίζουμε το gain της εξόδου που γράφεται στο αρχείο output.wav και χρησιμοποιείται εφόσον κλιπάρουν τα δεδομένα που γράφονται (έχουν τιμές πάνω από 1 και κάτω από -1)

```
handles.NBITS = 16;
```

Bit Depth του wav αρχείου που θα γραφτούν τα δεδομένα

```
handles.FS = 44100;
```

Συχνότητα Δειγματοληψίας του wav αρχείου που θα γραφτούν τα δεδομένα.

```
handles.nyquist = handles.FS/2;
```

Συχνότητα Nyquist

```
handles.inputSig = wavread('inputSig');
```

Εισαγωγή του αρχείου με το σήμα εισόδου μέσω της wavread

```
handles.N = length(handles.inputSig);
```

Δημιουργία μεταβλητής N που θα χρησιμοποιηθεί ως μήκος του FFT

```
handles.inputL = handles.inputSig(:,1);
```

Δημιουργία πίνακα με τα δεδομένα του αριστερού καναλιού του στερεοφωνικού αρχείου εισόδου.

```
handles.inputR = handles.inputSig(:,2);
```

Δημιουργία πίνακα με τα δεδομένα του δεξιού καναλιού του στερεοφωνικού αρχείου εισόδου.

```
disp('loaded input signal file: inputSig.wav');
```

Με την εντολή disp εμφανίζουμε μηνύματα στο παράθυρο γραμμής εντολών (MCR) που τρέχει μαζί με την εφαρμογή. Η ίδια διαδικασία ακολουθείται και για τα υπόλοιπα αρχεία

```
handles.impulseLSLE = wavread('LSLE');
```

```
handles.impulseLSRE = wavread('LSRE');
```

```
handles.impulseRSLE = wavread('RSLE');
```

```
handles.impulseRSRE = wavread('RSRE');
```

Στο σημείο αυτό αφού έχουν φορτωθεί τα αρχεία στο πρόγραμμα, εκτελούμε τον FFT σε κάθε ένα από τα αρχεία.

Για το αριστερό κανάλι του σήματος εισόδου

```
handles.XL = fft(handles.inputL, handles.N);
```

Για το δεξί κανάλι του σήματος εισόδου

```
handles.XR = fft(handles.inputR, handles.N);
```

Για κάθε ένα από τα σήματα των αποκρίσεων

```
handles.HLSLE = fft(handles.impulseLSLE, handles.N);
```

```
handles.HLSRE = fft(handles.impulseLSRE, handles.N);
```

```
handles.HRSLE = fft(handles.impulseRSLE, handles.N);
```

```
handles.HRSRE = fft(handles.impulseRSRE, handles.N);
```

Στη συνέχεια αφού έχει γίνει μετασχηματισμός όλων των σημάτων, γίνονται και οι υπολογισμοί για τη δημιουργία του σήματος εξόδου

```
handles.YL = handles.XL.*handles.HLSLE + handles.XL.*handles.HRSLE;
```

```
handles.YR = handles.XR.*handles.HRSRE + handles.XR.*handles.HLSRE;
```

Όπως απεικονίζεται και στο σχήμα 5.1, για το αριστερό αντί θα αθροιστούν, το φάσμα του σήματος από το αριστερό ηχείο το οποίο έχει γίνει συνελιχθεί με το φάσμα της απόκρισης  $LSLE$  με το φάσμα του σήματος από το δεξί ηχείο το οποίο έχει συνελιχθεί με το φάσμα της απόκρισης  $RSLE$ . Για το δεξί αντί θα αθροιστούν, το φάσμα του σήματος από το δεξί ηχείο το οποίο έχει γίνει συνελιχθεί με το φάσμα της απόκρισης  $RSRE$  με το φάσμα του σήματος από το αριστερό ηχείο το οποίο έχει συνελιχθεί με το φάσμα της απόκρισης  $LSRE$ .

Αφού γίνουν οι υπολογισμοί για το σήμα εξόδου, είναι όλα έτοιμα για να δημιουργηθεί και το αρχείο εξόδου `output.wav`. Πρώτο βήμα είναι η δημιουργία ενός πίνακα για την αποθήκευση των τιμών των του σήματος εξόδου.

```
handles.au_output = zeros(handles.N, 2);
```

Με την εντολή αυτή δημιουργείται ένας πίνακας με όνομα `au_output` 2 στήλες και  $N$  γραμμές.



```
handles.au_output(:,1) = ifft(handles.YL, handles.N);
handles.au_output(:,2) = ifft(handles.YR, handles.N);
```

Με τις δύο παραπάνω εντολές οι τιμές του αντίστροφου FFT του πίνακα YL αποθηκεύονται στην πρώτη στήλη του πίνακα `au_output` και με τη δεύτερη οι τιμές του αντίστροφου FFT του πίνακα YR αποθηκεύονται στην δεύτερη στήλη του πίνακα `au_output`.

Επόμενη διαδικασία είναι η κανονικοποίηση (normalization) των δεδομένων στον πίνακα εξόδου έτσι ώστε να μην ψαλιδίζεται η κυματομορφή του σήματος εξόδου.

Βρίσκουμε το μέγιστο και στα δύο κανάλια

```
handles.outMaxL = max(handles.au_output(:,1));
handles.outMaxR = max(handles.au_output(:,2));
```

Βρίσκουμε το ελάχιστο και στα δύο κανάλια

```
handles.outMinL = min(handles.au_output(:,1));
handles.outMinR = min(handles.au_output(:,2));
```

Εκτελείται η κανονικοποίηση σε κάθε κανάλι ξεχωριστά

```
handles.au_output(:,1) = ((handles.au_output(:,1) - handles.outMinL) ./ ...
(handles.outMaxL - handles.outMinL) - 0.5) .* 2;

handles.au_output(:,2) = ((handles.au_output(:,2) - handles.outMinR) ./ ...
(handles.outMaxR - handles.outMinR) - 0.5) .* 2;

handles.au_output = handles.au_output .* handles.outputGain;
```

Και τέλος τα δεδομένα γράφονται στο αρχείο εξόδου.

```
wavwrite(handles.au_output, handles.FS, handles.NBITS, 'output.wav');
```

Τα ορίσματα της `wavwrite` είναι με τη σειρά, ο πίνακας που περιέχει τα δεδομένα που θα γραφτούν στο αρχείο, η συχνότητα δειγματοληψίας του αρχείου, το βάθος bit του αρχείου και τέλος το όνομα του αρχείου.

Το επόμενο μέρος που είναι και το τελευταίο αφορά την δημιουργία των γραφημάτων.

Αριθμός των στοιχείων στο πεδίο του χρόνου

```
handles.nfreq = floor(handles.N/2)+1;
```

Συχνότητα που θα χρησιμοποιήσουμε στην plot

```
handles.frequency = linspace(0, handles.nyquist, handles.nfreq);
```

Μετατροπή σε dB – αφορά την απεικόνιση των σημάτων στο πεδίο των συχνοτήτων

```
handles.dBhLSLE = 20*log(abs(handles.HLSLE(1:handles.nfreq)));  
handles.dBhLSRE = 20*log(abs(handles.HLSRE(1:handles.nfreq)));  
handles.dBhRSLE = 20*log(abs(handles.HRSLE(1:handles.nfreq)));  
handles.dBhRSRE = 20*log(abs(handles.HRSRE(1:handles.nfreq)));  
handles.dBXL = 20*log(abs(handles.XL(1:handles.nfreq)));  
handles.dBXR = 20*log(abs(handles.XR(1:handles.nfreq)));  
handles.dBYL = 20*log(abs(handles.YL(1:handles.nfreq)));  
handles.dBYR = 20*log(abs(handles.YR(1:handles.nfreq)));
```

Σχεδίαση των γραφημάτων των κρουστικών αποκρίσεων στο πεδίο του χρόνου.

```
plot(handles.lsle_resp_graph, handles.impulseLSLE);
```

Στην περίπτωση που η plot χρησιμοποιείται όπως εδώ, με γραφική διεπαφή χρήστη και πολλαπλά αντικείμενα axes η plot παίρνει σαν πρώτο όρισμα το αντικείμενο πάνω στο οποίο θα σχεδιάσει.

```
plot(handles.lsre_resp_graph, handles.impulseLSRE);  
plot(handles.rsle_resp_graph, handles.impulseRSLE);  
plot(handles.rsre_resp_graph, handles.impulseRSRE);
```

Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα. Εδώ το πρώτο όρισμα είναι το αντικείμενο axes στο οποίο θα σχεδιαστούν οι κυματομορφές και το δεύτερο είναι ένας πίνακας του οποίου τα δύο στοιχεία είναι τα δύο σήματα που θα σχεδιαστούν στο γράφημα. Για να παρουσιαστούν και τα δύο σήματα στο ίδιο γράφημα, το αριστερό κανάλι έχει μεταφερθεί πάνω από τον άξονα  $x$  και το δεξί κάτω από τον άξονα  $x$  παίρνοντας τιμές στα διαστήματα  $0 - 0.5$  και  $-0.5 - 0$  αντίστοιχα.

```
plot(handles.input_signal_graph, [handles.inputSig(:,1).*0.5+0.5 ...  
handles.inputSig(:,2).*0.5-0.5]);
```

```
plot(handles.output_signal_graph, [handles.au_output(:,1).*0.5+0.5 ...
handles.au_output(:,2).*0.5-0.5]);
```

### 6.3.2.2 view\_menu Callback

Η συνάρτηση αυτή είναι η συνάρτηση που καλείται όταν συμβεί κάποια αλλαγή στο αντικείμενο της διεπαφής view\_menu. Το αντικείμενο αυτό είναι το pop-up menu, και η λειτουργία που κάνει είναι να καθορίζει αν στα γραφήματα τα σήματα θα απεικονίζονται στο πεδίο του χρόνου ή στο πεδίο των συχνοτήτων. Αυτό πετυχαίνεται μέσω μίας μεταβλητής την οποία αντιμετωπίζουμε σαν μία boolean μεταβλητή, μπορεί να πάρει δηλαδή δύο μόνο διακριτές τιμές. Στο παρακάτω τμήμα κώδικα αυτό που συμβαίνει είναι ότι όταν καλείται η συνάρτηση αυτή αποθηκεύεται στην μεταβλητή str αυτό που είναι επιλεγμένο στο μενού. Μέσω της get() η οποία επιστρέφει την τιμή της καθορισμένης ιδιότητας του αντικειμένου hObjectc. Στην περίπτωση αυτή το String ( αλφαριθμητικό ) που εμφανίζει το αντικείμενο view\_menu την εν λόγω στιγμή. Στην επόμενη γραμμή συμβαίνει το ίδιο αλλά αυτή την φορά αποθηκεύεται στην μεταβλητή val η τιμή του αντικειμένου που είναι επιλεγμένο στο view menu. Στη συνέχεια δίνεται, ανάλογα με την τιμή val τι είναι επιλεγμένο στο view menu και αντίστοιχα δίνουμε την τιμή 1 στην μεταβλητή isTimeDomainSelected αν η επιλογή είναι time domain, το πρώτο στοιχείο του μενού δηλαδή ή 0 αν είναι επιλεγμένο το δεύτερο στοιχείο του μενού.

Το όνομα που έχει δοθεί στο μενού ( view\_menu ) στην περίπτωση μας

```
str = get(hObject, 'String');
```

Η επιλογή που έκανε ο χρήστης στο μενού

```
val = get(hObject, 'Value');
```

Μέσω της δομής ελέγχου switch επιλέγουμε την τιμή που θα πάρει η μεταβλητή

isTimeDomainSelected 0 αν η επιλογή (val) που έχει γίνει στο αντικείμενο ( str ) είναι

'frequency domain' ή 1 αν είναι 'time domain'.

```
switch str{val};
    case 'time domain'
        handles.isTimeDomainSelected = 1;
    case 'frequency domain'
        handles.isTimeDomainSelected = 0;
end
```

### 6.3.2.3 load\_button\_Callback

Αυτή η συνάρτηση είναι αυτή που καλείται όταν πατιέται το κουμπί load. Οι συναρτήσεις των οποίων το όνομα τελειώνει με Callback είναι αυτές οι συναρτήσεις που καλούνται όταν συμβαίνει κάτι στο αντικείμενο που έχει αντιστοιχηθεί με αυτές. Με συνάρτηση αυτή ξαναφορτώνονται μέσω της wavread αρχεία με τα σήματα των αποκρίσεων και του σήματος εισόδου, ακριβώς όπως είχαν φορτωθεί και στην \_OpeningFcn.

```
handles.inputSig = wavread('inputSig');
handles.N = length(handles.inputSig);
handles.inputL = handles.inputSig(:,1);
handles.inputR = handles.inputSig(:,2);
handles.impulseLSLE = wavread('LSLE');
handles.impulseLSRE = wavread('LSRE');
handles.impulseRSLE = wavread('RSLE');
handles.impulseRSRE = wavread('RSRE');
```

### 6.3.2.4 process\_button\_Callback

Η συνάρτηση αυτή είναι αυτή που καλείται όταν πατηθεί το κουμπί process και επανεκτελεί τους υπολογισμούς και παράγει ένα νέο αρχείο εξόδου. Αυτό το κουμπί σε συνδυασμό με το load μπορούμε να τα χρησιμοποιούμε σε περίπτωση που αντικαταστήσουμε κάποιο από τα αρχεία στο φάκελο με το εκτελέσιμο, για να ακούσουμε το καινούριο αποτέλεσμα. Η διαδικασία που γίνεται είναι πανομοιότυπη με αυτή που γίνεται κατά την αρχικοποίηση της εφαρμογής.

### 6.3.2.5 Callbacks για τα κουμπιά που αναπαράγουν τα σήματα στην έξοδο.

Οι εν λόγω συναρτήσεις καλούνται όταν πατιούνται τα αντίστοιχα κουμπιά στη διεπαφή. Σε κάθε μία γίνεται αναπαραγωγή στην έξοδο της κάρτας ήχου το αντίστοιχο σήμα με τη χρήση της wavplay. Το συντακτικό της wavplay έχει ως εξής: το πρώτο όρισμα είναι ο πίνακας που περιέχει το σήμα που θα αναπαράγουμε στην έξοδο, το δεύτερο είναι η συχνότητα δειγματοληψίας με την οποία θα στέλνονται τα δείγματα στην έξοδο και το τελευταίο όρισμα είναι ένα αλφαριθμητικό το οποίο χρησιμοποιούμε έτσι ώστε όταν αναπαράγεται το σήμα στην έξοδο να μην παγώνει η εφαρμογή. Η εντολή έχει ως εξής για κάθε ένα από τα σήματα ξεχωριστά:

Για το σήμα εξόδου:

```
wavplay(handles.au_output, handles.FS, 'async');
```

Για το σήμα εισόδου:

```
wavplay(handles.inputSig, handles.FS, 'async');
```

Για το σήμα για την LSLE απόκριση:

```
wavplay(handles.impulseLSLE, handles.FS, 'async');
```

Για το σήμα για την LSRE απόκριση:

```
wavplay(handles.impulseLSRE, handles.FS, 'async');
```

Για το σήμα για την RSLE απόκριση:

```
wavplay(handles.impulseRSLE, handles.FS, 'async');
```

Για το σήμα για την RSRE απόκριση:

```
wavplay(handles.impulseRSRE, handles.FS, 'async');
```

### 6.3.2.6 Callbacks για το κουμπί Draw All και τα υπόλοιπα κουμπιά που αφορούν τη σχεδίαση των γραφημάτων

Οι συναρτήσεις αυτές καλούνται όταν πατιούνται τα κουμπιά που αφορούν τα γραφήματα όπως το Draw All και τα υπόλοιπα αποκλειστικά για κάθε γράφημα Draw. Στην draw\_all Callback σχεδιάζουμε όλα τα γραφήματα της εφαρμογής στο πεδίο του χρόνου αν η μεταβλητή `isTimeDomainSelected` έχει τιμή 1 αλλιώς τα σχεδιάζουμε στο πεδίο των συχνοτήτων. Η διαδικασία σχεδίασης είναι ίδια με αυτή που γίνεται κατά την αρχικοποίηση της εφαρμογής με τη διαφορά ότι χρησιμοποιείται η δομή ελέγχου ροής του προγράμματος `if` έτσι ώστε να είναι εφικτή η σχεδίαση των κυματομορφών όταν `isTimeDomainSelected == 1` αλλιώς να σχεδιάζονται τα φάσματα.

```
if(handles.isTimeDomainSelected == 1)
    plot(handles.lsle_resp_graph, handles.impulseLSLE);
    plot(handles.lsre_resp_graph, handles.impulseLSRE);
    plot(handles.rsle_resp_graph, handles.impulseRSLE);
    plot(handles.rsre_resp_graph, handles.impulseRSRE);
    % Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα
    plot(handles.input_signal_graph, [handles.inputSig(:,1).*0.5+0.5 ...
        handles.inputSig(:,2).*0.5-0.5]);

    % Παρουσίαση και των δύο καναλιών εξόδου στο ίδιο γράφημα
```

```

plot(handles.output_signal_graph,[handles.au_output(:,1).*0.5+0.5 ...
    handles.au_output(:,2).*0.5-0.5]);
else

semilogx(handles.lsle_resp_graph, handles.frequency, handles.dBhLSLE);
axis(handles.lsle_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.lsre_resp_graph, handles.frequency, handles.dBhLSRE);
axis(handles.lsre_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.rsle_resp_graph, handles.frequency, handles.dBhRSLE);
axis(handles.rsle_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.rsre_resp_graph, handles.frequency, handles.dBhRSRE);
axis(handles.rsre_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.input_signal_graph, handles.frequency, handles.dBXL +
handles.dBXR);
axis(handles.input_signal_graph, [0 handles.nfreq/2 min(handles.dBXL +
handles.dBXR) ...
max(handles.dBXL + handles.dBXR)]);
semilogx(handles.output_signal_graph, handles.frequency, handles.dBYL +
handles.dBYR);
axis(handles.output_signal_graph, [0 handles.nfreq/2 min(handles.dBYL
+ handles.dBYR) ...
max(handles.dBYL + handles.dBYR)]);
end

% Update handles structure
guidata(hObject, handles);

```

## 6.4 Τρόπος εγκατάστασης της εφαρμογής

Στο σημείο αυτό θα περιγραφεί ο τρόπος εγκατάστασης της εφαρμογής. Αρχικά η εφαρμογή βρίσκεται στη μορφή συμπιεσμένου αρχείου .rar απ' όπου πρέπει να αποσυμπιεστεί στην επιθυμητή τοποθεσία στον σκληρό δίσκο του υπολογιστή όπου θα εγκατασταθεί.

Τα περιεχόμενα του συμπιεσμένου αυτού αρχείου είναι οι τέσσερις αποκρίσεις σε μορφή .wav που φορτώνονται στο πρόγραμμα κατά την εκκίνηση, όπως και το αρχείο εισόδου. Ένα συμπιεσμένο αρχείο .rar το οποίο περιέχει φάκελο με αποκρίσεις, οργανωμένες σε φακέλους, από τον ίδιο χώρο με άλλη κάρτα ήχου και άλλο ένα σετ μετρήσεων από ένα άλλο χώρο με άλλο εξοπλισμό ο οποίος αναγράφεται στο όνομα του αντίστοιχου φακέλου. Ένα αρχείο κειμένου το οποίο περιέχει οδηγίες για την εγκατάσταση και τη χρήση της εφαρμογής. Ένα συμπιεσμένο αρχείο με τον πηγαίο κώδικα της εφαρμογής σε γλώσσα MATLAB, και τέλος το αρχείο <όνομα\_εφαρμογής>\_pkg.exe το οποίο προκύπτει όπως περιγράφηκε στην παράγραφο 5.4.2.

Αρχικά αποσυμπιέζουμε το συμπιεσμένο αρχείο στον φάκελο της επιλογής μας. Στη συνέχεια εκτελούμε το αρχείο Ptyxiaki\_Leledakis\_562\_pkg.exe. Όταν εκτελεστεί το συγκεκριμένο αρχείο, μέσα στον ίδιο φάκελο εμφανίζονται άλλα τέσσερα αρχεία, ένα readme.txt το οποίο αφορά τον MCR στον οποίο έγινε αναφορά στην παράγραφο 5.4.2, το πρόγραμμα εγκατάστασης του MCR MCRInstaller.exe, ένα αρχείο \_install.bat που αφορά την εγκατάσταση του MCR και το ίδιο το αρχείο της εφαρμογής Ptyxiaki\_Leledakis\_562.exe.

Κατά την εκτέλεση του Ptyxiaki\_Leledakis\_562\_pkg.exe ερωτάται ο χρήστης αν θέλει να εγκαταστήσει τον MCR στον υπολογιστή του. Σε υπολογιστή ο οποίος δεν έχει εγκατεστημένο το MATLAB είναι αναγκαίο να εγκατασταθεί ο MCR για να μπορέσει να τρέξει η εφαρμογή μέσω αυτού. Μόλις ολοκληρωθεί η εγκατάσταση του MCR η εφαρμογή είναι έτοιμη να τρέξει.

## 6.5 Τρόπος χρήσης της εφαρμογής

Η χρήση της εφαρμογής έχει δύο σκέλη. Την ακρόαση των αρχείων ήχου, και την παρουσίαση των αρχείων αυτών σε γραφήματα στο πεδίο του χρόνου ή στο πεδίο των συχνοτήτων.

Το πάνελ Controls περιέχει τα γενικά χειριστήρια που αφορούν συνολικά τα αρχεία ήχου που χειρίζεται η εφαρμογή. Από το drop down μενού όπου αρχικά είναι επιλεγμένη η επιλογή time domain, επιλέγοντας οποιαδήποτε από τις δύο επιλογές και στη συνέχεια πατώντας το κουμπί Draw All ξανασχεδιάζονται τα γραφήματα απεικονίζοντας τα σήματα είτε στο πεδίο του χρόνου είτε στο πεδίο των συχνοτήτων.

Σε περίπτωση που αντικατασταθούν τα αρχεία των αποκρίσεων με άλλα ή το αρχείο εισόδου, θα πρέπει να πατηθεί το κουμπί Load Files για να ξαναφορτωθούν τα αρχεία στην εφαρμογή και έπειτα να πατηθεί το κουμπί process για να γίνει η επεξεργασία πάνω στα νέα αρχεία.

Για να γίνει ακρόαση οποιουδήποτε από τα αρχεία αρκεί να πατηθεί το αντίστοιχο κουμπί που βρίσκεται κάτω από το αντίστοιχο γράφημα.



# ΚΕΦΑΛΑΙΟ 7ο : ΣΥΜΠΕΡΑΣΜΑΤΑ-ΠΕΡΙΟΡΙΣΜΟΙ ΤΗΣ ΕΡΕΥΝΑΣ-ΠΡΟΤΑΣΕΙΣ

## 7.1 Συμπεράσματα - Προτάσεις

Κατά την ακρόαση του αρχείου εξόδου output.wav με ακουστικά παρατηρούμε πώς η προσομοίωση έχει επιτευχθεί σε ικανοποιητικό βαθμό. Αυτό που παρατηρούμε είναι πώς έχει μετακινηθεί η στερεοφωνική εικόνα έξω από το κεφάλι και σε θέση που προσεγγίζει αυτή που έχει στην κανονική ακρόαση από τα ηχεία.

Παρατηρείται κάποια απόκλιση από την πραγματική ακρόαση πρώτον λόγω του ότι οι μετρήσεις πραγματοποιήθηκαν με κεφάλι κούκλας, και όχι με dummy head που είναι ειδικό για τέτοιες μετρήσεις γιατί δεν υπήρχε διαθέσιμο, το οποίο δεν είχε κατασκευαστεί με προδιαγραφές που να προσεγγίζουν τα ακουστικά χαρακτηριστικά του ανθρώπινου κεφαλιού πέρα από το σχήμα και τη θέση των αυτιών καθώς και τη γενική τους μορφολογία χωρίς ιδιαίτερη λεπτομέρεια στο σχήμα και το ανάγλυφο του περυγίου. Επίσης είναι γνωστό ότι στις HRTFs συμμετέχει σε ένα μικρό βαθμό και το άνω μέρος του κορμού το οποίο επίσης δεν υπήρχε.

Οι αποκλίσεις αυτές λόγω των ατελειών των ομοιωμάτων που χρησιμοποιήθηκαν ήταν αναμενόμενες επίσης, λόγω του ότι ο κάθε άνθρωπος έχει μάθει να εντοπίζει τις ηχητικές πηγές στο χώρο χρησιμοποιώντας τις δικές του συναρτήσεις μεταφοράς του κεφαλιού οι οποίες διαμορφώνονται σημαντικά από σχήμα και το ανάγλυφο του περυγίου του αυτιού. Οπότε πέρα από τις ατέλειες και τις ελλείψεις των ομοιωμάτων, περιοριστικός παράγοντας στη ρεαλιστικότητα του αποτελέσματος ήταν και η άγνωστη στον ακροατή απόκριση της κεφαλής που χρησιμοποιήθηκε περιορίζοντας με τον τρόπο αυτό την ικανότητά του να αντιληφθεί με σαφήνεια τη θέση των ηχείων στην προσομοίωση. Η προσομοίωση της ακρόασης από ηχεία σε ακουστικά πραγματοποιήθηκε σε ικανοποιητικό βαθμό τελικώς μέσω της εφαρμογής και των μετρήσεων που έγιναν με τα συγκεκριμένα υλικά και μέσα.

Σχετικά με την αποτελεσματικότητα της εργασίας θα μπορούσε να πραγματοποιηθεί υποκειμενικός έλεγχος με τη χρήση listening tests για να καθοριστεί ο βαθμός στον οποίο το αποτέλεσμά ήταν πετυχημένο. Στη συνέχεια θα μπορούσε να πραγματοποιηθεί και αντικειμενικός έλεγχος της αποτελεσματικότητας με μέτρηση και σύγκριση των κρουστικών αποκρίσεων αφού παρθούν νέες με τη χρήση dummy head. Τέλος θα μπορούσε να γίνει και έλεγχος αποτελεσματικότητας της μεθόδου και σε άλλους χώρους με πιο πολύπλοκο ακουστικό πεδίο

# Βιβλιογραφία

## Ελληνική

Κουζούπης Σπύρος, Σηφάκης Μηνάς, (2007) ΑΤΕΙ Κρήτης, παράρτημα Ρεθύμνου, τμήμα Μουσικής Τεχνολογίας Και Ακουστικής, «Εφαρμοσμένη Ακουστική II Σημειώσεις Εργαστηριακών Ασκήσεων»

Παπαδάκης, Ν. (2009) ΑΤΕΙ Κρήτης, παράρτημα Ρεθύμνου, τμήμα Μουσικής Τεχνολογίας Και Ακουστικής, Πανεπιστημιακές σημειώσεις για το μάθημα «Εισαγωγή στην Ψυχοακουστική». ΑΤΕΙ Κρήτης, Παράρτημα Ρεθύμνου, Τμήμα Μουσικής Τεχνολογίας και Ακουστικής.

Παπαδάκης Νίκος Ποταμίτης Ηλίας, (2009) ΑΤΕΙ Κρήτης, παράρτημα Ρεθύμνου, τμήμα Μουσικής Τεχνολογίας Και Ακουστικής,, 'Σημειώσεις Εργαστηρίου Ψηφιακή Επεξεργασία Σήματος'

Ποταμίτης Ηλίας, (2009), ΑΤΕΙ Κρήτης, παράρτημα Ρεθύμνου, τμήμα Μουσικής Τεχνολογίας Και Ακουστικής,, 'Σημειώσεις Θεωρίας Ψηφιακή Επεξεργασία Σήματος'

Κατσάνος, Ευάγγελος, “Διδακτικές σημειώσεις: Βασικά Στοιχεία για τη χρήση του MATLAB & εφαρμογή σε προβλήματα κατασκευών”

Χατζίκος, Ευάγγελος Β. (2010) “Matlab για επιστήμονες και μηχανικούς “, Εκδόσεις Τζιόλα

Σκαρλάτος Δημήτρης (2003), “Εφαρμοσμένη Ακουστική”, Φιλομάθεια

## Ξενόγλωσση

Roads, Curtis (1996). The Computer Music Tutorial, Massachusetts Institute of Technology, Pt. 3 Mixing and Signal processing, pp.486

Moore, Brian C. J. (2003). An Introduction to the Psychology of Hearing, Fifth Edition, 233, 238

Yost, William A. (2006). Fundamentals of Hearing, Fifth Edition

Lynn, Paul A., Fuerst, Wolfgang (1994) 'Introductory Digital Signal Processing', John Wiley & sons

Zolzer Udo, (1995) 'Digital Audio Signal Processing', John Wiley & sons

Ingle, Vinay K., Proakis, John G., (2007) 'Digital Signal Processing Using MATLAB', CENGAGE Learning

Lyons, Richard G., (2001) 'Understanding Digital Audio Signal Processing', Prentice Hall

Hanselman, Duane and Bruce Littlefield. (2001) 'Mastering MATLAB 6: A Comprehensive Tutorial and Reference' Prentice Hall

Smith, Steven W. (1997) 'The Scientists and Engineer's Guide to Digital Signal Processing', California Technical Publishing

Howard, David M., Angus, Jamie, (1996) "Acoustics and Psychoacoustics", Third Edition, Focal Press

Cook, Perry R, (2001) "Music, Cognition, and Computerized Sound", An Introduction to Psychoacoustics", MIT Press

Hartmann, William M. (1998) "Signals, Sound, and Sensation" Springer Science & Business Media, Inc.

Crocker, Malcom J. (1998) "Handbook of Acoustics", John Wiley & sons

Halmrast T, Buen A, A paper presented by [www.akutek.info](http://www.akutek.info), "Simplified Room Acoustic Measurements"

Griesinger, David, "Beyond MLS - Occupied hall measurement with FFT techniques"

Ahnert, Wolfgang, Schmidt, Wolfgang "Fundamentals to perform acoustical measurements – Appendix to

EASERA”

F. Rund, D. Štorek, O. Glaser, M. Bard ”ORIENTATION IN SIMPLE VIRTUAL AUDITORY SPACE CREATED WITH MEASURED HRTF", Faculty of Electrical Engineering  
Czech Technical University in Prague, Prague, Czech Republic

Fausti, Patrizio, Farina, Angelo, “ACOUSTIC MEASUREMENTS IN OPERA HOUSES: COMPAR- ISON BETWEEN DIFFERENT TECHNIQUES AND EQUIPMENT”,  
Department of Engineering, University of Ferrara, I-44100 Ferrara, Italy, Department of Industrial Engineering, University of Parma, I-43100 Parma, Italy

“FFT Tutorial”, University of Rhode Island Department of Electrical and Computer Engineering  
ELE 436: Communication Systems

Swarztrauber, Paul N.(1986), “Symmetric FFTs”, Math. Comp., 47, pp. 323-346

## Σχήματα

### ΚΕΦΑΛΑΙΟ 1

Σχήμα 1.1 από <http://webschoolsolutions.com/patts/systems/ear.htm>

Σχήμα 1.2 – 1.9 από Σημειώσεις μαθήματος Ψυχοακουστικής, Παπαδάκης Νίκος

Σχήμα 1.10 – 1.14, 1.16 από Moore, Brian C. J. (2003). An Introduction to the Psychology of Hearing, Fifth Edition,

Σχήμα 1.15, 1.17, 1.18 από Yost, William A. (2006). Fundamentals of Hearing, Fifth Edition

### ΚΕΦΑΛΑΙΟ 2

Σχήμα 2.1 – 2.5 από Lynn, Paul A., Fuerst, Wolfgang(1994) 'Introductory Digital Signal Processing', John Wiley & sons

## ΚΕΦΑΛΑΙΟ 4

Σχήμα 4.1 από Howard, David M., Angus, Jamie, “Acoustics and Psychoacoustics”, Third Edition, Focal Press

# ΠΑΡΑΡΤΗΜΑΑ – ΠΛΗΡΗΣ ΛΙΣΤΑ ΚΩΔΙΚΑ ΤΗΣ ΕΦΑΡ- ΜΟΓΗΣ

Παρακάτω παρουσιάζεται η πλήρης λίστα με τον κώδικα της εφαρμογής όπως είναι στο αρχείο Ptyxiaki\_Leledakis\_562.m.

```
function varargout = Ptyxiaki_Leledakis_562(varargin)
% Ptyxiaki_Leledakis_562 M-file for Ptyxiaki_Leledakis_562.fig
%     Ptyxiaki_Leledakis_562, by itself, creates a new
Ptyxiaki_Leledakis_562 or raises the existing
%     singleton*.
%
%     H = Ptyxiaki_Leledakis_562 returns the handle to a new
Ptyxiaki_Leledakis_562 or the handle to
%     the existing singleton*.
%
%     Ptyxiaki_Leledakis_562('CALLBACK',hObject,eventData,handles,...)
calls the local
%     function named CALLBACK in Ptyxiaki_Leledakis_562.M with the given
input arguments.
%
%     Ptyxiaki_Leledakis_562('Property','Value',...) creates a new
Ptyxiaki_Leledakis_562 or raises the
%     existing singleton*. Starting from the left, property value pairs
are
%     applied to the GUI before Ptyxiaki_Leledakis_562_OpeningFcn gets
called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to Ptyxiaki_Leledakis_562_OpeningFcn
via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
```

```

% Edit the above text to modify the response to help Ptyxiaki_Leledakis_562

% Last Modified by GUIDE v2.5 04-Jan-2012 23:02:13

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @Ptyxiaki_Leledakis_562_OpeningFcn, ...
                  'gui_OutputFcn',  @Ptyxiaki_Leledakis_562_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Ptyxiaki_Leledakis_562 is made visible.
function Ptyxiaki_Leledakis_562_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Ptyxiaki_Leledakis_562 (see
VARARGIN)

% Ο κώδικας από το σημείο αυτό και πάνω δημιουργείται αυτόματα
% από το guide και δεν θα αναλυθεί αφόσον αναλύεται επαρκώς στο
% σχετικό help του MATLAB.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΑΡΧΙΚΟΠΟΙΗΣΕΙΣ%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Ξεκινώντας, φτιάχνουμε κάποιες μεταβλητές που αφορούν τα γενικά
% χαρακτηριστικά του προγράμματός. Το αναγνωριστικό handles μπαίνει σε κάθε

```

```

% μεταβλητή γιατί έτσι σχετίζεται κάθε μία από αυτές με το GUI ( Γραφική
% Διεπαφή Χρήστη) εξηγείται αναλυτικά στα σχόλια παραπάνω. Εν ολίγοις
% οποιαδήποτε μεταβλητή για να συσχετιστεί με δεδομένα από το GUI πρέπει να
% ανοίκει στη δομή ( structure όπως στη γλώσσα προγραμματισμού C ) handles.

```

```

handles.isTimeDomainSelected = 1; % αυτή η μεταβλητή χρησιμοποιείται από
% την draw_all_button_Callback
% και χρησιμοποιείται στον καθορισμό
% του τι θα σχεδιαστεί στα γραφήματα.

```

```

handles.outputGain = 0.9; % Εδώ ρυθμίζουμε το gain της εξόδου
που
% γράφεται στο αρχείο output.wav και
% χρησιμοποιείται εφόσον κλιπάρουν τα
% δεδομένα που γράφονται (έχουν τιμές
% πάνω από 1 και κάτω από -1)

```

```

handles.NBITS = 16; %% Bit Depth του wav αρχείου που θα γρα-
% φτούν τα δεδομένα

```

```

handles.FS = 44100; %% Συχνότητα Δειγματοληψίας του wav
%αρχείου που θα γραφτούν τα δεδομένα.

```

```

handles.nyquist = handles.FS/2; % Συχνότητα Nyquist

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΕΙΣΑΓΩΓΗ WAV ΑΡΧΕΙΩΝ %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

handles.inputSig = wavread('inputSig'); % Εισαγωγή του αρχείου με το
% σήμα εισόδου μέσω της wavread

```

```

handles.N = length(handles.inputSig); % Δημιουργία μεταβλητής N που
% θα χρησιμοποιηθεί ως μήκος
% του FFT

```



```

handles.inputL = handles.inputSig(:,1);      Δημιουργία πίνακα με τα δεδο-
% μένα του αριστερού καναλιού
% του στερεοφωνικού αρχείου
% εισόδου.

handles.inputR = handles.inputSig(:,2);      % Δημιουργία πίνακα με τα δεδο-
% μένα του δεξιού καναλιού
% του στερεοφωνικού αρχείου
% εισόδου.

disp ('loaded input signal file: inputSig.wav'); %% Με την εντολή disp('')
% εμφανίζουμε μηνύματα
% στο παράθυρο γραμμής
% εντολών (MCR) που τρέχει μαζί
% με την εφαρμογή.

handles.impulseLSLE = wavread('LSLE');
disp ('loaded impulse response file: LSLE.wav');

handles.impulseLSRE = wavread('LSRE');
disp ('loaded impulse response file: LSRE.wav');

handles.impulseRSLE = wavread('RSLE');
disp ('loaded impulse response file: RSLE.wav');

handles.impulseRSRE = wavread('RSRE');
disp ('loaded impulse response file: RSRE.wav');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΕΦΑΡΜΟΓΗ ΤΟΥ FFT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

handles.XL = fft(handles.inputL, handles.N);
disp ('FFT performed on data of left channel of input file');

handles.XR = fft(handles.inputR, handles.N);
disp ('FFT performed on data of right channel of input file');

```

```

handles.HLSLE = fft(handles.impulseLSLE, handles.N);
disp ('FFT performed on data of LSLE impulse response file');

handles.HLSRE = fft(handles.impulseLSRE, handles.N);
disp ('FFT performed on data of LSRE impulse response file');

handles.HRSLE = fft(handles.impulseRSLE, handles.N);
disp ('FFT performed on data of RSLE impulse response file');

handles.HRSRE = fft(handles.impulseRSRE, handles.N);
disp ('FFT performed on data of RSRE impulse response file');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΠΡΑΓΜΑΤΟΠΟΙΗΣΗ ΥΠΟΛΟΓΙΣΜΩΝ %%%%%%%%%%%%%%%
disp ('Calculating the output data');
tic
handles.YL = handles.XL.*handles.HLSLE + handles.XL.*handles.HRSLE;
handles.YR = handles.XR.*handles.HRSRE + handles.XR.*handles.HLSRE;
toc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ ΕΞΟΔΟΥ output.wav %%%%%%%%%%%%%%%
disp ('Creating table to store the output data');
handles.au_output = zeros(handles.N, 2);
disp ('Performing inverse FFT on the processed data');
tic
handles.au_output(:,1) = ifft(handles.YL, handles.N);
handles.au_output(:,2) = ifft(handles.YR, handles.N);
toc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% NORMALIZATION ΣΤΑ ΔΕΔΟΜΕΝΑ ΕΞΟΔΟΥ %%%%%%%%%%%%%%%
disp ('Normalizing the output data');
tic
handles.outMaxL = max(handles.au_output(:,1));
handles.outMaxR = max(handles.au_output(:,2));

handles.outMinL = min(handles.au_output(:,1));
handles.outMinR = min(handles.au_output(:,2));

handles.au_output(:,1) = ((handles.au_output(:,1)- handles.outMinL)./ ...
(handles.outMaxL - handles.outMinL) - 0.5).*2;

handles.au_output(:,2) = ((handles.au_output(:,2)- handles.outMinR)./ ...

```

```

(handles.outMaxR - handles.outMinR) - 0.5).*2;
handles.au_output = handles.au_output .* handles.outputGain;
toc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% WRITING THE OUTPUT FILE %%%%%%%%%
disp ('Creating output file output.wav');
wavwrite(handles.au_output, handles.FS, handles.NBITS, 'output.wav' );
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΣΧΕΤΙΚΑ ΜΕ ΤΑ ΓΡΑΦΗΜΑΤΑ ΣΤΟ ΠΕΔΙΟ ΣΥΧΝΟΤΗΤΩΝ %%%%%%%%%
handles.nfreq = floor(handles.N/2)+1; % Αριθμός των στοιχείων στο πεδίο
                                     % του χρόνου

handles.frequency = linspace(0, handles.nyquist, handles.nfreq); %
Συχνότητα

                                     % που θα
                                     % χρησιμοπ
                                     % οίήσουμε
                                     % στην
                                     % plot

handles.dBhLSLE = 20*log(abs(handles.HLSLE(1:handles.nfreq))); % Μετατροπή
                                     % σε dB

handles.dBhLSRE = 20*log(abs(handles.HLSRE(1:handles.nfreq)));
handles.dBhRSLE = 20*log(abs(handles.HRSLE(1:handles.nfreq)));
handles.dBhRSRE = 20*log(abs(handles.HRSRE(1:handles.nfreq)));
handles.dBXL = 20*log(abs(handles.XL(1:handles.nfreq)));
handles.dBXR = 20*log(abs(handles.XR(1:handles.nfreq)));
handles.dBYL = 20*log(abs(handles.YL(1:handles.nfreq)));
handles.dBYR = 20*log(abs(handles.YR(1:handles.nfreq)));

plot(handles.lsle_resp_graph, handles.impulseLSLE);
plot(handles.lsre_resp_graph, handles.impulseLSRE);
plot(handles.rsle_resp_graph, handles.impulseRSLE);
plot(handles.rsre_resp_graph, handles.impulseRSRE);

% Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα
plot(handles.input_signal_graph, [handles.inputSig(:,1)].*0.5+0.5 ...

```

```

handles.inputSig(:,2).*0.5-0.5]);

% Παρουσίαση και των δύο καναλιών εξόδου στο ίδιο γράφημα
plot(handles.output_signal_graph,[handles.au_output(:,1).*0.5+0.5 ...
handles.au_output(:,2).*0.5-0.5]);

% Choose default command line output for Ptyxiaki_Leledakis_562
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Ptyxiaki_Leledakis_562 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Ptyxiaki_Leledakis_562_OutputFcn(hObject, eventdata,
handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on selection change in view_menu.
function view_menu_Callback(hObject, eventdata, handles)
% hObject handle to view_menu (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns view_menu
% contents as cell array
% contents{get(hObject,'Value')} returns selected item from
view_menu

% Determine the selected data set.
str = get(hObject, 'String');
val = get(hObject, 'Value');

```

```

% Set current data to the selected data set.
switch str{val};
    case 'time domain' % User selects peaks.
        handles.isTimeDomainSelected = 1;

    case 'frequency domain' % User selects membrane.
        handles.isTimeDomainSelected = 0;

end

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function view_menu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to view_menu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), ...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in load_button.
function load_button_Callback(hObject, eventdata, handles)
% hObject    handle to load_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΕΙΣΑΓΩΓΗ WAV ΑΡΧΕΙΩΝ %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
handles.inputSig = wavread('inputSig');      % Εισαγωγή του αρχείου με το
% σήμα εισόδου μέσω της wavread

handles.N = length(handles.inputSig);      % Δημιουργία μεταβλητής N που

```

```

% Θα χρησιμοποιηθεί ως μήκος
% του FFT

handles.inputL = handles.inputSig(:,1);      Δημιουργία πίνακα με τα δεδο-
% μένα του αριστερού καναλιού
% του στερεοφωνικού αρχείου
% εισόδου.

handles.inputR = handles.inputSig(:,2);      % Δημιουργία πίνακα με τα δεδο-
% μένα του δεξιού καναλιού
% του στερεοφωνικού αρχείου
% εισόδου.

disp ('loaded input signal file: inputSig.wav'); %% Με την εντολή disp('')
% εμφανίζουμε μηνύματα
% στο παράθυρο γραμμής
% εντολών (MCR) που τρέχει μαζί
% με την εφαρμογή.

handles.impulseLSLE = wavread('LSLE');
disp ('loaded impulse response file: LSLE.wav');

handles.impulseLSRE = wavread('LSRE');
disp ('loaded impulse response file: LSRE.wav');

handles.impulseRSLE = wavread('RSLE');
disp ('loaded impulse response file: RSLE.wav');

handles.impulseRSRE = wavread('RSRE');
disp ('loaded impulse response file: RSRE.wav');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in process_button.

```

```

function process_button_Callback(hObject, eventdata, handles)
    % hObject    handle to process_button (see GCBO)
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΕΦΑΡΜΟΓΗ ΤΟΥ FFT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    handles.XL = fft(handles.inputL, handles.N);
    disp ('FFT performed on data of left channel of input file');

    handles.XR = fft(handles.inputR, handles.N);
    disp ('FFT performed on data of right channel of input file');

    handles.HLSLE = fft(handles.impulseLSLE, handles.N);
    disp ('FFT performed on data of LSLE impulse response file');

    handles.HLSRE = fft(handles.impulseLSRE, handles.N);
    disp ('FFT performed on data of LSRE impulse response file');

    handles.HRSLE = fft(handles.impulseRSLE, handles.N);
    disp ('FFT performed on data of RSLE impulse response file');

    handles.HRSRE = fft(handles.impulseRSRE, handles.N);
    disp ('FFT performed on data of RSRE impulse response file');

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΠΡΑΓΜΑΤΟΠΟΙΗΣΗ ΥΠΟΛΟΓΙΣΜΩΝ %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    disp ('Calculating the output data');
    tic
    handles.YL = handles.XL.*handles.HLSLE + handles.XL.*handles.HRSLE;
    handles.YR = handles.XR.*handles.HRSRE + handles.XR.*handles.HLSRE;
    toc

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ΔΗΜΙΟΥΡΓΙΑ ΑΡΧΕΙΟΥ ΕΞΟΔΟΥ output.wav %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    disp ('Creating table to store the output data');
    handles.au_output = zeros(handles.N, 2);
    disp ('Performing inverse FFT on the processed data');
    tic
    handles.au_output(:,1) = ifft(handles.YL, handles.N);
    handles.au_output(:,2) = ifft(handles.YR, handles.N);
    toc

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% NORMALIZATION ΣΤΑ ΔΕΔΟΜΕΝΑ ΕΞΟΔΟΥ %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

disp ('Normalizing the output data');
tic
handles.outMaxL = max(handles.au_output(:,1));
handles.outMaxR = max(handles.au_output(:,2));

handles.outMinL = min(handles.au_output(:,1));
handles.outMinR = min(handles.au_output(:,2));

handles.au_output(:,1) = ((handles.au_output(:,1)- handles.outMinL)./ ...
    (handles.outMaxL - handles.outMinL) - 0.5).*2;

handles.au_output(:,2) = ((handles.au_output(:,2)- handles.outMinR)./ ...
    (handles.outMaxR - handles.outMinR) - 0.5).*2;
handles.au_output = handles.au_output .* handles.outputGain;
toc

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% WRITING THE OPUTPUT FILE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
disp ('Creating output file output.wav');
wavwrite(handles.au_output, handles.FS, handles.NBITS, 'output.wav' );

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in play_output_button.
function play_output_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_output_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

wavplay(handles.au_output, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in play_input_button.
function play_input_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_input_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```



```

wavplay(handles.inputSig, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in play_lsle_button.
function play_lsle_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_lsle_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

wavplay(handles.impulseLSLE, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in play_lsre_button.
function play_lsre_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_lsre_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

wavplay(handles.impulseLSRE, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in play_rsle_button.
function play_rsle_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_rsle_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

wavplay(handles.impulseRSLE, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

```

```

% --- Executes on button press in play_rsre_button.
function play_rsre_button_Callback(hObject, eventdata, handles)
% hObject    handle to play_rsre_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

wavplay(handles.impulseRSRE, handles.FS, 'async');

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_all_button.
function draw_all_button_Callback(hObject, eventdata, handles)
% hObject    handle to draw_all_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(handles.isTimeDomainSelected == 1)
    plot(handles.lsle_resp_graph, handles.impulseLSLE);
    plot(handles.lsre_resp_graph, handles.impulseLSRE);
    plot(handles.rsle_resp_graph, handles.impulseRSLE);
    plot(handles.rsre_resp_graph, handles.impulseRSRE);
    % Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα
    plot(handles.input_signal_graph, [handles.inputSig(:,1).*0.5+0.5 ...
        handles.inputSig(:,2).*0.5-0.5]);

    % Παρουσίαση και των δύο καναλιών εξόδου στο ίδιο γράφημα
    plot(handles.output_signal_graph, [handles.au_output(:,1).*0.5+0.5 ...
        handles.au_output(:,2).*0.5-0.5]);
else

    semilogx(handles.lsle_resp_graph, handles.frequency, handles.dBhLSLE);
    axis(handles.lsle_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
    max(handles.dBhLSLE)]);

    semilogx(handles.lsre_resp_graph, handles.frequency, handles.dBhLSRE);
    axis(handles.lsre_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
    max(handles.dBhLSLE)]);

```

```

semilogx(handles.rsle_resp_graph, handles.frequency, handles.dBhRSLE);
axis(handles.rsle_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.rsre_resp_graph, handles.frequency, handles.dBhRSRE);
axis(handles.rsre_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);

semilogx(handles.input_signal_graph, handles.frequency, handles.dBXL +
handles.dBXR);
axis(handles.input_signal_graph, [0 handles.nfreq/2 min(handles.dBXL +
handles.dBXR) ...
max(handles.dBXL + handles.dBXR)]);
semilogx(handles.output_signal_graph, handles.frequency, handles.dBYL +
handles.dBYR);
axis(handles.output_signal_graph, [0 handles.nfreq/2 min(handles.dBYL
+ handles.dBYR) ...
max(handles.dBYL + handles.dBYR)]);
end

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function lsle_resp_graph_CreateFcn(hObject, eventdata, handles)
% hObject    handle to lsle_resp_graph (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate lsle_resp_graph

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function lsre_resp_graph_CreateFcn(hObject, eventdata, handles)
% hObject    handle to lsre_resp_graph (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```

```

% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate lsre_resp_graph

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function rsle_resp_graph_CreateFcn(hObject, eventdata, handles)
% hObject      handle to rsle_resp_graph (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate rsle_resp_graph

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function rsre_resp_graph_CreateFcn(hObject, eventdata, handles)
% hObject      handle to rsre_resp_graph (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate rsre_resp_graph

% Update handles structure
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function input_signal_graph_CreateFcn(hObject, eventdata, handles)
% hObject      handle to input_signal_graph (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate input_signal_graph

% Update handles structure

```

```

guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function output_signal_graph_CreateFcn(hObject, ~, handles)
% hObject    handle to output_signal_graph (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate output_signal_graph

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_output.
function draw_output_Callback(hObject, ~, handles)
% hObject    handle to draw_output (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if(handles.isTimeDomainSelected == 1)
    % Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα
    plot(handles.output_signal_graph,[handles.au_output(:,1).*0.5+0.5 ...
        handles.au_output(:,2).*0.5-0.5]);
else
    semilogx(handles.output_signal_graph, handles.frequency, handles.dBYL +
handles.dBYR);
    axis(handles.output_signal_graph, [0 handles.nfreq/2 min(handles.dBYL
+ handles.dBYR) ...
        max(handles.dBYL + handles.dBYR)]);
end
% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_input.
function draw_input_Callback(hObject, ~, handles)
% hObject    handle to draw_input (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(handles.isTimeDomainSelected == 1)

```

```

    % Παρουσίαση και των δύο καναλιών εισόδου στο ίδιο γράφημα
    plot(handles.input_signal_graph, [handles.inputSig(:,1).*0.5+0.5 ...
        handles.inputSig(:,2).*0.5-0.5]);
else
    semilogx(handles.input_signal_graph, handles.frequency, handles.dBXL +
handles.dBXR);
    axis(handles.input_signal_graph, [0 handles.nfreq/2 min(handles.dBXL +
handles.dBXR) ...
        max(handles.dBXL + handles.dBXR)]);
end

% --- Executes on button press in draw_lsle_button.
function draw_lsle_button_Callback(hObject, ~, handles)
% hObject    handle to draw_lsle_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(handles.isTimeDomainSelected == 1)
    plot(handles.lsle_resp_graph, handles.impulseLSLE);
else
    semilogx(handles.lsle_resp_graph, handles.frequency, handles.dBhLSLE);
    axis(handles.lsle_resp_graph, [0 handles.nfreq min(handles.dBhLSLE)
max(handles.dBhLSLE)]);
end

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_lsre_button.
function draw_lsre_button_Callback(hObject, ~, handles)
% hObject    handle to draw_lsre_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(handles.isTimeDomainSelected == 1)
    plot(handles.lsre_resp_graph, handles.impulseLSRE);
else
    semilogx(handles.lsre_resp_graph, handles.frequency, handles.dBhLSRE);
    axis(handles.lsre_resp_graph, [0 handles.nfreq min(handles.dBhLSRE)
max(handles.dBhLSRE)]);
end

```

```

end

% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_rsle_button.
function draw_rsle_button_Callback(hObject, ~, handles)
% hObject    handle to draw_rsle_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if(handles.isTimeDomainSelected == 1)
    plot(handles.rsle_resp_graph, handles.impulseRSLE);
else
    semilogx(handles.rsle_resp_graph, handles.frequency, handles.dBhRSLE);
    axis(handles.rsle_resp_graph, [0 handles.nfreq min(handles.dBhRSLE)
max(handles.dBhRSLE)]);
end
% Update handles structure
guidata(hObject, handles);

% --- Executes on button press in draw_rsre_button.
function draw_rsre_button_Callback(hObject, ~, handles)
% hObject    handle to draw_rsre_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

if(handles.isTimeDomainSelected == 1)
    plot(handles.rsre_resp_graph, handles.impulseRSRE);
else
    semilogx(handles.rsre_resp_graph, handles.frequency, handles.dBhRSRE);
    axis(handles.rsre_resp_graph, [0 handles.nfreq min(handles.dBhRSRE)
max(handles.dBhRSRE)]);
end
% Update handles structure
guidata(hObject, handles);

```