



Σχολή Θετικών Επιστημών & Τεχνολογίας

Μεταπτυχιακές Σπουδές στα Μαθηματικά

Διπλωματική Εργασία

Μαθηματικά μοντέλα κρουστών οργάνων και υπολογιστικές  
προσομοιώσεις

Κωνσταντίνος Λιόλιος

Επιβλέπων καθηγητής: Νικόλαος Τσίτσας

Θεσσαλονίκη, Μάιος 2026

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Κωνσταντίνου Λιόλιου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης ο συγγραφέας/δημιουργός εκχωρεί στο ΕΑΠ, μη αποκλειστική άδεια χρήσης του δικαιώματος αναπαραγωγής, προσαρμογής, δημόσιου δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσής τους διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος και για όλο το χρόνο διάρκειας των δικαιωμάτων πνευματικής ιδιοκτησίας. Η ανοικτή πρόσβαση στο πλήρες κείμενο για μελέτη και ανάγνωση δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, αποθήκευση, πώληση, εμπορική χρήση, μετάδοση, διανομή, έκδοση, εκτέλεση, «μεταφόρτωση» (downloading), «ανάρτηση» (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού. Ο συγγραφέας/δημιουργός διατηρεί το σύνολο των ηθικών και περιουσιακών του δικαιωμάτων.



Μαθηματικά μοντέλα κρουστών οργάνων και υπολογιστικές  
προσομοιώσεις

Κωνσταντίνος Λιόλιος

Επιτροπή Επίβλεψης Διπλωματικής Εργασίας

Επιβλέπων Καθηγητής:

Νικόλαος Τσίτσας

Τμήμα Πληροφορικής

Αριστοτέλειο Πανεπιστήμιο  
Θεσσαλονίκης

Συν-Επιβλέπων Καθηγητής:

Νικόλαος Καραχάλιος

Τμήμα Μαθηματικών

Πανεπιστήμιο Θεσσαλίας

Θεσσαλονίκη, Μάιος 2026

## Περίληψη

Στην παρούσα εργασία ασχοληθήκαμε με την μελέτη και μοντελοποίηση κρουστών οργάνων. Συγκεκριμένα διερευνήσαμε τα φυσικά φαινόμενα μέσω των οποίων παράγεται ήχος από διάφορα κρουστά όργανα, κατασκευάσαμε μαθηματικά μοντέλα βάσει αυτών των φαινομένων και τα υλοποιήσαμε σαν προγράμματα σε υπολογιστή με τα οποία παραγάγαμε ήχο ο οποίος έμοιαζε άλλοτε λιγότερο και άλλοτε περισσότερο με αυτόν των πραγματικών οργάνων. Συγκρίναμε τον παραγόμενο ήχο με τον ήχο των πραγματικών οργάνων και προσπαθήσαμε να κατανοήσουμε που οφείλονται οι διαφορές που παρατηρήθηκαν και τι μπορεί να βελτιωθεί έτσι ώστε ο παραγόμενος ήχος να γίνει πιο ρεαλιστικός.

Στο πρώτο κεφάλαιο της εργασίας κάναμε μια γενική εισαγωγή στο θέμα, παρουσιάσαμε στοιχειωδώς τη λειτουργία του ήχου σε υπολογιστή και παρουσιάσαμε στοιχειωδώς κάποιες μαθηματικές μεθόδους που χρησιμοποιήσαμε κατά την εκπόνηση της εργασίας.

Στο επόμενο κεφάλαιο (2 – 6) ασχοληθήκαμε με διάφορα όργανα. Συγκεκριμένα στο κεφάλαιο 2 ασχοληθήκαμε με το μεταλλόφωνο, στο κεφάλαιο 3 με το ξυλόφωνο, στο κεφάλαιο 4 με τη μαρίμπα, στο κεφάλαιο 5 με το βιμπράφωνο και στο κεφάλαιο 6 με το πιάνο. Η μεθοδολογία ήταν πάντα η ίδια. Αρχικά ασχοληθήκαμε με τη Φυσική που διέπει τη λειτουργία του κάθε οργάνου. Πρώτα με το πως παίρνει ενέργεια το σύστημα (για τα τέσσερα πρώτα όργανα αυτό γίνεται με το χτύπημα κάποιας μπαγκέτας ενώ για το πιάνο με το χτύπημα ενός «σφυριού»). Στην συνέχεια υπολογίσαμε τις ιδιοσυχνότητες κάθε οργάνου, δηλαδή τις δυνατές συχνότητες δόνησης του μέσου παραγωγής ήχου κάθε οργάνου (ξύλινη / μεταλλική μπάρα ή χορδή για το πιάνο). Και τέλος είδαμε τους τρόπους με τους οποίους το σύστημα χάνει ενέργεια. Με βάση όλα τα παραπάνω μπορέσαμε να κατασκευάσουμε ένα μοντέλο της λειτουργίας για κάθε όργανο το οποίο υλοποιήσαμε στον υπολογιστή. Στο τέλος κάθε κεφαλαίου κάναμε και μια αποτίμηση των αποτελεσμάτων.

Στο τελευταίο κεφάλαιο βγάλαμε κάποια γενικά συμπεράσματα από όλη την προηγούμενη εργασία.

### Λέξεις – Κλειδιά

Μοντελοποίηση οργάνων, κρουστά, ήχος, μέθοδος χωριζομένων μεταβλητών, μαθηματικό μοντέλο

# Mathematical models of percussion instruments and computational simulations

Konstantinos Liolios

## Abstract

In this thesis, we studied and modeled percussion instruments. Specifically, we investigated the physical phenomena through which sound is produced by various percussion instruments, constructed mathematical models based on these phenomena, and implemented them as computer programs that generated sounds which sometimes resembled the real instruments more closely and sometimes less so.

We compared the generated sound with that of the real instruments and attempted to understand the causes of the observed differences and what could be improved in order to make the produced sound more realistic.

In the first chapter of the thesis, we provided a general introduction to the topic, presented a basic overview of how sound functions on a computer, and briefly introduced some of the mathematical methods used in carrying out the thesis.

In the subsequent chapters (2–6), we examined various instruments. Specifically, in Chapter 2 we studied the metallophone, in Chapter 3 the xylophone, in Chapter 4 the marimba, in Chapter 5 the vibraphone, and in Chapter 6 the piano. The methodology was always the same. First, we examined the physics governing the operation of each instrument—starting with how the system receives energy (for the first four instruments this occurs through striking with a mallet, while for the piano it occurs through the impact of a “hammer”). Next, we calculated the natural frequencies of each instrument, that is, the possible vibration frequencies of the sound-producing medium (a wooden or metal bar, or a string in the case of the piano). Finally, we examined the ways in which the system loses energy.

Based on all the above, we were able to construct a functional model for each instrument, which we implemented on a computer. At the end of each chapter, we also provided an evaluation of the results.

In the final chapter, we presented some general conclusions drawn from the entire study.

### **Keywords**

Instrument modeling, percussion, sound, method of separation of variables, mathematical model.

## Περιεχόμενα

Περίληψη.....	iv
Abstract .....	v
Κατάλογος Εικόνων / Σχημάτων .....	ix
Κατάλογος Πινάκων .....	xi
Συνοτομογραφίες & Ακρωνύμια.....	xii
1. Εισαγωγή.....	1
1.1 Γενικά.....	1
1.2 Δειγματοληψία με υπολογιστή.....	3
1.2.1 Το μικρόφωνο .....	3
1.2.2 Analog to Digital Converter (ADC).....	3
1.2.3 Η διαδικασία στον υπολογιστή .....	4
1.2.4 Θεώρημα δειγματοληψίας.....	5
1.3 Το αρχείο ήχου .wav .....	6
1.4 Μαθηματική εισαγωγή.....	7
1.4.1 Σειρές Fourier.....	7
1.4.2 Μέθοδος χωριζομένων μεταβλητών .....	8
1.4.3 Μέθοδος διχοτόμησης.....	9
1.4.4 Μέθοδος του Νεύτωνα (Newton – Raphson).....	9
1.4.5 Μέθοδος πεπερασμένων στοιχείων.....	10
2. Μοντελοποίηση του μεταλλόφωνου .....	11
2.1 Το όργανο.....	11
2.2 Το μοντέλο .....	12
2.2.1 Γενικά.....	12
2.2.2 Διέγερση του συστήματος.....	13
2.2.3 Οι ιδιοσυχνότητες του οργάνου .....	16
2.2.4 Απώλεια ενέργειας .....	21
2.2.5 Το μοντέλο συνολικά .....	23
2.3 Υλοποίηση στον υπολογιστή .....	24
2.4 Αποτίμηση μοντέλου.....	25
3. Μοντελοποίηση του ξυλόφωνου.....	29
3.1 Το όργανο.....	29
3.2 Το μοντέλο .....	30
3.2.1 Διέγερση συστήματος .....	30
3.2.2 Κυματική εξίσωση για αντηχείο .....	31
3.2.3 Ενεργό μήκος αντηχείου .....	33
3.2.4 Απώλειες αντηχείου .....	33
3.2.5 Το μοντέλο συνολικά .....	34
3.3 Υλοποίηση στον υπολογιστή .....	35
3.4 Αποτίμηση μοντέλου.....	35
4. Μοντελοποίηση της μαρίμπας .....	39
4.1 Το όργανο.....	39
4.2 Το μοντέλο .....	39
4.2.1 Συντονισμός Helmholtz .....	39
4.2.2 Απώλειες αντηχείου .....	42

4.2.3 Αντηχεία μαρίμπας.....	42
4.2.4 Ασθενής σύζευξη .....	42
4.2.5 Το μοντέλο συνολικά .....	44
4.3 Υλοποίηση στον υπολογιστή .....	44
4.4 Αποτίμηση μοντέλου.....	45
5. Μοντελοποίηση του βιμπράφωνου .....	48
5.1 Το όργανο.....	48
5.2 Το μοντέλο .....	49
5.2.1 Ιδιοσυχνότητες .....	49
5.2.2 Βιμπράτο .....	49
5.3 Υλοποίηση στον υπολογιστή .....	50
5.4 Αποτίμηση μοντέλου.....	50
6. Μοντελοποίηση του γούντμπλοκ (woodblock) .....	53
6.1 Το όργανο.....	53
6.2 Το μοντέλο .....	53
6.3 Υλοποίηση στον υπολογιστή .....	55
6.4 Αποτίμηση μοντέλου.....	55
7. Μοντελοποίηση του πιάνου .....	58
7.1 Το όργανο.....	58
7.2 Το μοντέλο .....	59
7.2.1 Κυματική εξίσωση σε μια διάσταση .....	59
7.2.2 Κυματική εξίσωση σε πραγματική χορδή.....	60
7.2.3 Αρχικές συνθήκες.....	63
7.2.4 Inharmonicity .....	64
7.2.5 Διαμήκη κύματα.....	65
7.2.6 Αντηχείο (soundboard).....	65
7.2.7 Συντονισμός Helmholtz .....	68
7.2.8 Το μοντέλο συνολικά .....	69
7.3 Υλοποίηση στον υπολογιστή .....	69
7.4 Αποτίμηση μοντέλου.....	69
8. Συμπεράσματα .....	76
Βιβλιογραφία.....	78
Παράρτημα Α: Κλάση CWaveFile - αρχείο .wav .....	79
Παράρτημα Β: Κλάσεις βάσης.....	81
Παράρτημα Γ: Ο κώδικας μεταλλόφωνου .....	86
Παράρτημα Δ: Ο κώδικας του ξυλόφωνου.....	89
Παράρτημα Ε: Ο κώδικας της μαρίμπας.....	93
Παράρτημα ΣΤ: Ο κώδικας του βιμπράφωνου .....	96
Παράρτημα Ζ: Ο κώδικας του γούντμπλοκ .....	99
Παράρτημα Η: Ο κώδικας του πιάνου .....	102

## Κατάλογος Εικόνων / Σχημάτων

Εικόνα 1: Παράδειγμα δειγματοληψίας.....	4
Εικόνα 2: Παράδειγμα παρεμβολής.....	5
Εικόνα 3: Aliasing.....	5
Εικόνα 4: Το μεταλλόφωνο.....	11
Εικόνα 5: Μπαγκέτες.....	12
Εικόνα 6: Δύναμη κατά την επαφή.....	14
Εικόνα 7: Γραφική παράσταση $(\omega, F(\omega))$ .....	16
Εικόνα 8: Γραφική παράσταση $\cos(x)$ , $1/\cosh(x)$ .....	18
Εικόνα 9: Γενικές κυματομορφές πραγματικού - μοντέλου μεταλλόφωνου.....	25
Εικόνα 10: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου μεταλλόφωνου.....	26
Εικόνα 11: Φάσμα ντο πραγματικού μεταλλόφωνου.....	26
Εικόνα 12: Φάσμα ντο μοντέλου μεταλλόφωνου.....	27
Εικόνα 13: Το ξυλόφωνο.....	29
Εικόνα 14: Ξυλόφωνο αριστερά, μαρίμπα δεξιά.....	30
Εικόνα 15: Γενικές κυματομορφές πραγματικού - μοντέλου ξυλόφωνου.....	36
Εικόνα 16: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου ξυλόφωνου.....	36
Εικόνα 17: Φάσμα ντο πραγματικού ξυλόφωνου.....	37
Εικόνα 18: Φάσμα ντο μοντελοποιημένου ξυλόφωνου.....	38
Εικόνα 19: Η μαρίμπα.....	39
Εικόνα 20: Αντηχείο Helmholtz.....	40
Εικόνα 21: Σύστημα μαζών - ελατηρίων.....	43
Εικόνα 22: Γενικές κυματομορφές πραγματικής - μοντέλου μαρίμπας.....	45
Εικόνα 23: Λεπτομερείς κυματομορφές πραγματικής - μοντέλου μαρίμπας.....	46
Εικόνα 24: Φάσμα ντο πραγματικής μαρίμπας.....	47
Εικόνα 25: Φάσμα ντο μοντελοποιημένης μαρίμπας.....	47
Εικόνα 26: Το βιμπράφωνο.....	48
Εικόνα 27: Ο μηχανισμός του βιμπράφωνου.....	49
Εικόνα 28: Γενικές κυματομορφές πραγματικού - μοντέλου βιμπράφωνου.....	51
Εικόνα 29: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου βιμπράφωνου.....	51
Εικόνα 30: Φάσμα ντο πραγματικού βιμπράφωνου.....	52
Εικόνα 31: Φάσμα ντο μοντελοποιημένου βιμπράφωνου.....	52
Εικόνα 32: Το γούντμπλοκ.....	53
Εικόνα 33: Γενικές κυματομορφές πραγματικού - μοντέλου γούντμπλοκ.....	56
Εικόνα 34: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου γούντμπλοκ.....	56
Εικόνα 35: Φάσμα πραγματικού γούντμπλοκ.....	56
Εικόνα 36: Φάσμα μοντελοποιημένου γούντμπλοκ.....	57
Εικόνα 37: Το πιάνο.....	58
Εικόνα 38: Χτύπημα χορδής πιάνου.....	63
Εικόνα 39: Αντηχείο πιάνου (soundboard).....	66
Εικόνα 40: Γενικές κυματομορφές ντο2 πραγματικού / μοντελοποιημένου πιάνου.....	70
Εικόνα 41: Λεπτομερείς κυματομορφές ντο2 πραγματικού / μοντελοποιημένου πιάνου.....	70
Εικόνα 42: Φάσμα ντο2 πραγματικού πιάνου.....	71
Εικόνα 43: Φάσμα ντο2 μοντελοποιημένου πιάνου.....	71

Εικόνα 44: Γενικές κυματομορφές ντο4 πραγματικού / μοντελοποιημένου πιάνου .....	72
Εικόνα 45: Λεπτομερείς κυματομορφές ντο4 πραγματικού / μοντελοποιημένου πιάνου ...	72
Εικόνα 46: Φάσμα ντο4 πραγματικού πιάνου.....	72
Εικόνα 47: Φάσμα ντο4 μοντελοποιημένου πιάνου .....	73
Εικόνα 48: Γενικές κυματομορφές ντο6 πραγματικού / μοντελοποιημένου πιάνου .....	73
Εικόνα 49: Λεπτομερείς κυματομορφές ντο6 πραγματικού / μοντελοποιημένου πιάνου ..	73
Εικόνα 50: Φάσμα ντο6 πραγματικού πιάνου.....	74
Εικόνα 51: Φάσμα ντο6 μοντελοποιημένου πιάνου .....	74

## Κατάλογος Πινάκων

Πίνακας 1: Οι 16 πρώτες τιμές της $\beta_n$ .....	19
Πίνακας 2: Οι πρώτες 15 τιμές της σχετικής συχνότητας.....	20
Πίνακας 3: Τα συνοδευτικά αρχεία του μεταλλόφωνου.....	25
Πίνακας 4: Τα συνοδευτικά αρχεία του ξυλόφωνου.....	36
Πίνακας 5: Τα συνοδευτικά αρχεία της μαρίμπας.....	45
Πίνακας 6: Τα συνοδευτικά αρχεία του βιμπράφωνου.....	51
Πίνακας 7: Τα συνοδευτικά αρχεία του γούντμπλοκ.....	55
Πίνακας 8: Συχνοτική απόκριση αντηχείου.....	68
Πίνακας 9: Τα συνοδευτικά αρχεία του πιάνου.....	70

## **Συντομογραφίες & Ακρωνύμια**

ΔΕ	Διαφορική εξίσωση
ΜΔΕ	Μερική διαφορική εξίσωση
ΜΠΣ	Μέθοδος πεπερασμένων στοιχείων

## 1. Εισαγωγή

### 1.1 Γενικά

Αν και δεν γνωρίζουμε ακριβώς από πότε ασχολείται ο άνθρωπος με την μουσική, γνωρίζουμε πως αυτή είναι συνυφασμένη με την πορεία του. Τα παλαιότερα λείψανα οργάνων που έχουν βρεθεί χρονολογούνται στο 80.000 π.Χ., υπάρχει τοιχογραφία του 40.000 π.Χ. η οποία απεικονίζει κάποιον να παίζει ένα μουσικό όργανο και πλήθος νεότερων ευρημάτων, όμως η πεποίθηση των ειδικών είναι πως η μουσική είναι πολύ παλιότερη αυτών. Παρόλα αυτά η μελέτη της λειτουργίας των μουσικών οργάνων έχει μόλις έναν με δύο αιώνες ιστορία. Ο λόγος είναι πως για να παραχθεί ήχος από ένα μουσικό όργανο πολλά και διαφορετικά φαινόμενα συμβαίνουν, τα όργανα δηλαδή είναι πολύπλοκα φυσικά συστήματα τα οποία για να μελετηθούν χρειάζεται η κατανόηση πολλών και διαφορετικών φυσικών φαινομένων η οποία δεν θα μπορούσε να γίνει πριν την πρόοδο των φυσικών επιστημών σε έναν μεγάλο βαθμό.

Γιατί να ασχοληθούμε με την μελέτη αυτή; Αφενός από θεωρητικό ενδιαφέρον, ο άνθρωπος έχει έμφυτη την ανάγκη να ανακαλύψει και να κατανοήσει τον κόσμο γύρω του. Αφετέρου για πρακτικούς λόγους. Μέσα από την μελέτη και την κατανόηση της λειτουργίας ενός οργάνου μπορούμε να βρούμε τρόπους να το βελτιώσουμε. Υπάρχει ακόμα ένας λόγος.

Αν και οι ηλεκτρονικοί υπολογιστές έχουν μια μικρή σχετικά ιστορία (~ 80 χρόνια) έχουν μπει σε κάθε πτυχή της ζωής μας. Προφανώς χρησιμοποιούνται και στην μουσική. Οι εφαρμογές, όσον αφορά τη μουσική, είναι πολλές. Σήμερα με έναν ηλεκτρονικό υπολογιστή και ελάχιστο εξοπλισμό μπορεί κανείς να κάνει την δουλειά που πριν 20-30 χρόνια γινόντανε μόνο σε μεγάλα στούντιο. Μπορεί δηλαδή κανείς να ηχογραφήσει (με υψηλή ποιότητα) και να επεξεργαστεί τον ήχο, αλλά και να δημιουργήσει μουσική από το μηδέν, δηλαδή χωρίς ανθρώπους να παίζουν κάποια μουσικά όργανα και δεν αναφερόμαστε στην ηλεκτρονική μουσική μόνο αλλά και στην «παραδοσιακή» μουσική, δηλαδή σε μουσική με όργανα. Για να «παίξει» ο υπολογιστής ένα μουσικό, η διαδικασία αρχικά είχε ως εξής: ηχογραφούσανε μια νότα ενός μουσικού οργάνου και στην συνέχεια παίζανε αυτήν τη νότα με διαφορετικούς ρυθμούς αναπαραγωγής. Όταν ο ρυθμός αναπαραγωγής είναι ο ίδιος με τον ρυθμό ηχογράφησης τότε το αποτέλεσμα είναι άψογο, ακούγεται ακριβώς αυτό που ηχογραφήθηκε. Αν παίζουμε τη νότα με υψηλότερο ρυθμό αναπαραγωγής (δηλαδή αν επιταχύνουμε) τότε φαινομενικά το ύψος ανεβαίνει (δηλαδή μπορούμε να ηχογραφήσουμε

μια νότα ντο και να την παίζουμε σαν ρε). Το πρόβλημα με αυτήν τη διαδικασία είναι πως όσο απομακρυνόμαστε από τον ρυθμό ηχογράφησης, τόσο λιγότερο ρεαλιστικό είναι το αποτέλεσμα. Αρχικά λοιπόν, εξ' αιτίας των δυνατοτήτων των υπολογιστών, για να προσομοιώσει κανείς ένα όργανο μπορούσε να ηχογραφήσει μερικές νότες και να δουλέψει με την παραπάνω διαδικασία. Όσο βελτιωνόταν οι υπολογιστές (από άποψη μνήμης / ταχύτητας) τόσο βελτιωνόταν και η παραπάνω διαδικασία. Δηλαδή κάποια στιγμή φτάσαμε στο σημείο όπου κάθε νότα ενός οργάνου μπορούσε να ηχογραφηθεί και να παίζεται κανονικά. Ούτε αυτό όμως ήταν αρκετό αφού το ηχόχρωμα εξαρτάται και από άλλες παραμέτρους π.χ. την ένταση. Έτσι στη συνέχεια κάθε νότα ηχογραφούνταν πολλές φορές (π.χ. μια σε χαμηλή ένταση, μια σε μέτρια και μία σε υψηλή) πράγμα που βελτιώνει το αποτέλεσμα αλλά αύξανε το κόστος. Ενώ η παραπάνω διαδικασία δίνει αποτελέσματα έχει δύο μειονεκτήματα. Το πρώτο είναι το κόστος σε αποθηκευτικό χώρο (σήμερα φτάνουμε σε terabyte για ένα όργανο). Το δεύτερο είναι πως ακόμα και έτσι δεν μπορούμε να πιάσουμε όλες τις δυνατότητες ενός οργάνου. Για παράδειγμα ένα βιολί μπορεί να παιχτεί με τα δάχτυλα ή και με ένα δοξάρι. Αναλόγως το που αγγίζει το δοξάρι την χορδή αλλάζει και το ηχόχρωμα (αλλιώς ακούγεται κοντά στον καβαλάρη και αλλιώς κοντά στην ταστιέρα). Επίσης ο βιολιστής μπορεί να αλλάζει την πίεση ασκεί στις χορδές, την γωνία με την οποία ακουμπά το δοξάρι τις χορδές κ.ο.κ. Αν αναλογιστεί κανείς πως όλα τα παραπάνω συνδυάζονται αλλά και πως ο βιολιστής μπορεί να αλλάζει ανάμεσά τους σε πραγματικό χρόνο είναι προφανές πως η διαχείριση όλων των παραπάνω είναι κατ' ελάχιστο πολύ δύσκολη, στην πραγματικότητα αδύνατη.

Τα τελευταία λοιπόν χρόνια η μεθοδολογία τείνει να αλλάξει. Σήμερα η μέθοδος που κερδίζει συνεχώς έδαφος είναι η φυσική μοντελοποίηση των οργάνων (Physical modelling synthesis). Αντί να ηχογραφούνται τα όργανα δημιουργούνται μοντέλα των οργάνων στηριγμένα στα φυσικά φαινόμενα που εμφανίζονται σε αυτά. Γράφονται δηλαδή προγράμματα που προσομοιώνουν την παραγωγή του ήχου από τα όργανα. Η μέθοδος αυτή (προς το παρόν) δεν δίνει απολύτως ρεαλιστικά αποτελέσματα αλλά έχει δύο μεγάλα πλεονεκτήματα. Δεν έχει ιδιαίτερες απαιτήσεις σε μνήμη – συνήθως είναι της τάξης των kilobytes. Μπορεί κανείς με λίγες παραμέτρους, οι οποίες μπορούν να αλλάζουν σε πραγματικό χρόνο, να προσομοιώσει καλύτερα τον τρόπο παιχνιδιού του οργάνου (π.χ. στο παράδειγμα με το βιολί παραπάνω μπορεί να έχουμε σαν μια παράμετρο την θέση που

ακουμπάει το δοξάρι στην χορδή και να την μεταβάλλουμε σε πραγματικό χρόνο κ.ο.κ.). Σε κάθε περίπτωση είναι προφανές πως η μέθοδος αυτή ήρθε για να μείνει.

Στην παρούσα εργασία θα ασχοληθούμε με την μελέτη και μοντελοποίηση κάποιων κρουστών οργάνων. Θα μελετήσουμε τα βασικά φυσικά φαινόμενα που αφορούν τη λειτουργία τους και θα προσπαθήσουμε να κατασκευάσουμε μοντέλα που την περιγράφουν. Στη συνέχεια θα υλοποιήσουμε αυτά τα μοντέλα σαν προγράμματα και θα παράξουμε ήχο με τη βοήθεια υπολογιστή. Τέλος θα προσπαθήσουμε να κάνουμε μια αποτίμηση των μοντέλων συγκρίνοντας τα αποτελέσματα με τον ήχο πραγματικών οργάνων.

## **1.2 Δειγματοληψία με υπολογιστή**

Δεδομένου του ότι ο σκοπός της εργασίας είναι η κατασκευή μοντέλων αλλά και η υλοποίησή τους σε υπολογιστή θα χρειαστούμε κάποια στοιχειώδη κατανόηση του πως λειτουργεί η δειγματοληψία με έναν ηλεκτρονικό υπολογιστή.

### **1.2.1 Το μικρόφωνο**

Η διαδικασία ξεκινά με το μικρόφωνο. Το μικρόφωνο είναι μια συσκευή με την οποία ο ήχος μετατρέπεται σε ηλεκτρικό σήμα και συνοπτικά έχει ως εξής: στην είσοδό του μικροφώνου υπάρχει ένα διάφραγμα, από κατάλληλο υλικό, το οποίο μπορεί να πάλλεται εξ' αιτίας των ηχητικών κυμάτων. Στην συνέχεια υπάρχει κάποια διάταξη η οποία μετατρέπει την κίνηση του διαφράγματος σε ηλεκτρικό ρεύμα. Συνήθως τα μικρόφωνα τα χαρακτηρίζουμε βάσει του τρόπου της μετατροπής. Έτσι έχουμε τα δυναμικά μικρόφωνα, στα οποία η μετατροπή γίνεται με τη βοήθεια ενός πηνίου και ενός μαγνήτη (το πηνίο πάλλεται μαζί με το διάφραγμα στο πεδίο του μαγνήτη και στα άκρα του εμφανίζεται επαγωγική τάση), τα πυκνωτικά μικρόφωνα, στα οποία η μετατροπή γίνεται με τη βοήθεια ενός πυκνωτή κ.α.

Σε κάθε περίπτωση στην έξοδο του μικροφώνου έχουμε σήμα ανάλογου της παραμόρφωσης του διαφράγματος κάθε χρονική στιγμή.

### **1.2.2 Analog to Digital Converter (ADC)**

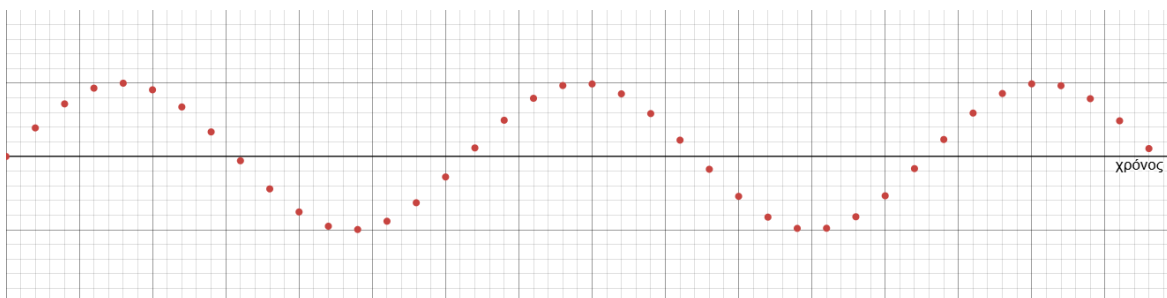
Δεδομένου ότι οι υπολογιστές διαχειρίζονται διακριτές ποσότητες ενώ το σήμα στην έξοδο του μικροφώνου είναι μια συνεχής ποσότητα (π.χ. τάση), ανάμεσά τους μεσολαβεί ένας

μετατροπέας που μετατρέπει την έξοδο του μικροφώνου σε ψηφιακό σήμα (μετατροπέας αναλογικό σε ψηφιακό).

### 1.2.3 Η διαδικασία στον υπολογιστή

Έχοντας λοιπόν ένα μικρόφωνο και έναν μετατροπέα (ADC) στη διάθεσή μας η διαδικασία στον υπολογιστή έχει ως εξής: σε τακτά χρονικά διαστήματα κρατάμε την τιμή του μετατροπέα (δεν καταγράφουμε δηλαδή συνεχώς αλλά παίρνουμε δείγματα, εξ' ου και ο όρος δειγματοληψία). Ο ρυθμός της καταγραφής ονομάζεται ρυθμός δειγματοληψίας (sample rate) και είναι βασικό χαρακτηριστικό της διαδικασίας διότι όπως θα δούμε παρακάτω καθορίζει την μέγιστη συχνότητα που μπορεί να καταγραφεί. Τον μετράμε σε Hz (Hertz). Όταν λοιπόν αναφερόμαστε σε δειγματοληψία 44,1 KHz εννοούμε πως λαμβάνουμε 44.100 δείγματα κάθε δευτερόλεπτο.

Ένα άλλο χαρακτηριστικό της δειγματοληψίας είναι το bit βάθος (bit depth). Οι υπολογιστές έχουν πεπερασμένη ακρίβεια, ως εκ τούτου το δυαδικό σήμα που παίρνουμε από τον μετατροπέα έχει πεπερασμένο αριθμό (δυαδικών) ψηφίων. Συνήθως το πλήθος των ψηφίων είναι 8/16/24 ή 32 και αναφερόμαστε σε δειγματοληψία 8/16/24 ή 32 bit αντίστοιχα. Ο αριθμός των bit καθορίζει την ακρίβεια της καταγραφής, όσο μεγαλύτερος τόσο μεγαλύτερη η ακρίβεια. Για παράδειγμα με 8 bit μπορούμε να αναπαραστήσουμε τους ακεραίους από το 0 έως το 255 (συνολικά 256 διαφορετικές τιμές) ενώ με 16, από το 0 έως το 65.635 (συνολικά 65.536 διαφορετικές τιμές), Όπως βλέπουμε η ακρίβεια αυξάνει εκθετικά. Στις μέρες μας ένα βάθος 32 bit είναι το σύνηθες.

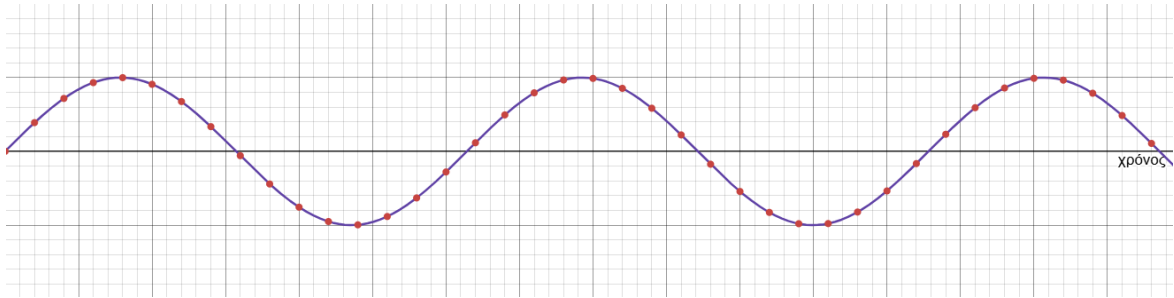


Εικόνα 1: Παράδειγμα δειγματοληψίας

Στην παραπάνω εικόνα βλέπουμε το αποτέλεσμα της δειγματοληψίας ενός απλού ημιτονικού ήχου. Στον οριζόντιο άξονα έχουμε τον χρόνο ενώ στον κατακόρυφο την τιμή του δείγματος. Σύμφωνα με όσα είπαμε παραπάνω όσο αυξάνει ο ρυθμός δειγματοληψίας

τόσο πιο πυκνά θα γίνονται τα δείγματα, ενώ όσο αυξάνεται το βάθος τόσο πιο πολλές στάθμες θα έχουμε στη διάθεσή μας για να την τιμή του κάθε δείγματος.

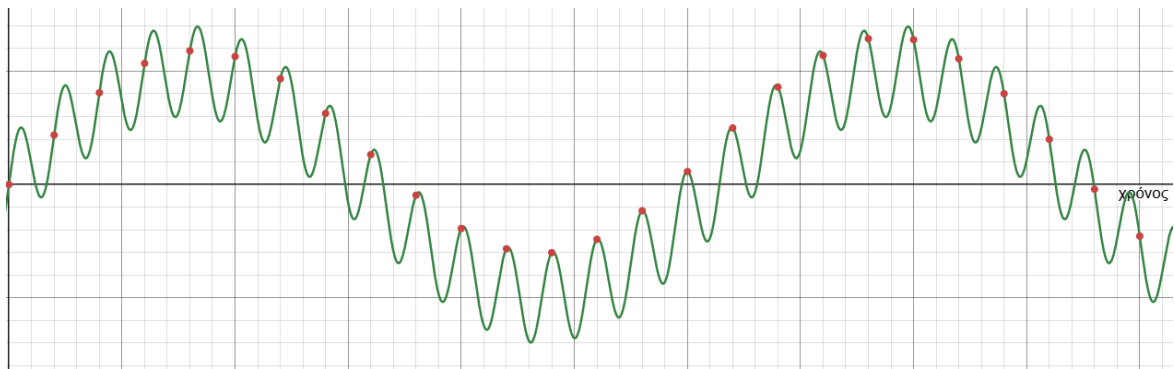
Για την αναπαραγωγή του ήχου γίνεται παρεμβολή στα δείγματα και έτσι μπορούμε να ξαναγυρίζουμε στα αρχικά – συνεχή μεγέθη. Ένα παράδειγμα παρεμβολής μπορούμε να δούμε στην παρακάτω εικόνα.



Εικόνα 2: Παράδειγμα παρεμβολής

#### 1.2.4 Θεώρημα δειγματοληψίας

Όπως αναφέραμε παραπάνω ο ρυθμός δειγματοληψίας καθορίζει την μέγιστη συχνότητα που μπορούμε να δειγματοληπτήσουμε. Έστω το παράδειγμα της παρακάτω εικόνας, όπου τα δείγματά είναι οι κόκκινες τελείες ενώ το πραγματικό σήμα η πράσινη καμπύλη.



Εικόνα 3: Aliasing

Είναι προφανές πως αν επιχειρήσουμε παρεμβολή ανάμεσα στα δείγματα θα καταλήξουμε σε ένα απλό ημίτονο όπως στις προηγούμενες περιπτώσεις και όχι στο πραγματικό σήμα και αυτό διότι τα δείγματα δεν είναι αρκετά ώστε να περιγράψουν το αρχικό σήμα. Σε αυτό ακριβώς το πρόβλημα απαντά το θεώρημα της δειγματοληψίας σύμφωνα με το οποίο:

Ένα συνεχές σήμα περιορισμένου εύρους ζώνης  $x(t)$ , με μέγιστη συχνότητα  $f_x$ , μπορεί να ανακτηθεί πλήρως από τα δείγματά του  $x(nT)$ ,  $n = 1, 2, 3, \dots$  εάν αυτά λαμβάνονται περιοδικά και η περίοδος δειγματοληψίας ικανοποιεί την σχέση  $T \leq \frac{1}{2f_x}$ .

[1] Η μέγιστη περίοδος δειγματοληψίας  $T = \frac{1}{2f_x}$  ονομάζεται διάστημα ή περίοδος Nyquist. Η ελάχιστη συχνότητα δειγματοληψίας  $f_s = 2f_x$  ονομάζεται ρυθμός ή συχνότητα Nyquist.

(Από Ψηφιακές επικοινωνίες I, Φούσκας, Δρ. Γιώργος)

Σύμφωνα με τα παραπάνω εάν δειγματοληπτούμε με ρυθμό  $f_{SR}$  τότε η μέγιστη συχνότητα που μπορούμε να ανακτήσουμε θα είναι ίση με:

$$f_{max} = \frac{f_{SR}}{2} \quad \text{Εξ. 1}$$

Το τελευταίο συμπέρασμα είναι πολύ βασικό για την υλοποίηση μοντέλων στον υπολογιστή και θα επανέλθουμε σε αυτό. Στην προσπάθειά μας να κατασκευάσουμε μοντέλα πραγματικών οργάνων θα χρησιμοποιήσουμε ανάλυση Fourier, δηλαδή θα φτάσουμε σε αποτελέσματα όπου θα χρειάζεται, θεωρητικώς, να υπολογίσουμε άπειρους όρους κάποιου αθροίσματος. Στην πράξη όμως κάτι τέτοιο δεν είναι αναγκαίο αφού η παραπάνω σχέση μας δίνει μια μέγιστη συχνότητα, δηλαδή ένα περιορισμένο πλήθος όρων που είναι αναγκαίο να χρησιμοποιήσουμε.

Να σημειώσουμε εδώ πως ο άνθρωπος μπορεί να ακούσει συχνότητες από περίπου 20 Hz έως περίπου 20 KHz. Σύμφωνα με την Εξ. 1 με ρυθμό δειγματοληψίας τα 44,1 KHz μπορούμε να καταγράψουμε συχνότητες έως και 22.050 Hz. Καλύπτουμε δηλαδή το ακουστικό φάσμα του ανθρώπου. Έτσι, αυτή θα είναι και η συχνότητα με την οποία θα δουλέψουμε.

### 1.3 Το αρχείο ήχου .wav

Όπως είπαμε και παραπάνω ο τελικός σκοπός της εργασίας είναι με τα μοντέλα που θα υλοποιήσουμε να παράξουμε ήχο. Για τον λόγο αυτό θα δημιουργήσουμε αρχεία τύπου wav. Το wav είναι ένα από τα πολλά αρχεία ήχου που χρησιμοποιούνται στις μέρες μας και οι λόγοι της επιλογής αυτής είναι αφενός πως είναι ανοιχτά αρχεία, δηλαδή υπάρχει δημόσια η πληροφορία που χρειάζεται για την υλοποίησή τους και αφετέρου πως δεν είναι συμπίεσμένα (τουλάχιστον στην βασική τους μορφή) δηλαδή περιέχουν τα δείγματα (samples) όπως τα περιγράψαμε παραπάνω ασυμπίεστα και χωρίς κάποια περεταίρω επεξεργασία.

Δεδομένου ότι δεν θέλουμε να ανοίγουμε αρχεία .wav αλλά να δημιουργούμε δεν χρειάζεται να δούμε κάθε λεπτομέρειά τους, παρά μόνο ότι χρειάζεται για την αποθήκευση.

Η δομή τους λοιπόν, από τη δική μας σκοπιά, είναι σχετικά απλή. Στην αρχή μπαίνει μια κεφαλίδα η οποία περιγράφει τα βασικά χαρακτηριστικά του αρχείου (sample rate, bit depth, αριθμός καναλιών κτλ.) και στην συνέχεια μια ή περισσότερες λίστες από δείγματα. Οι διαφορετικές λίστες ονομάζονται κανάλια και αντιστοιχούν σε διαφορετικές ταυτόχρονες ηχογραφήσεις. Όταν έχουμε δυο κανάλια τότε μιλάμε για στερεοφωνικό ήχο. Εμείς θα δουλέψουμε με μονοφωνικό ήχο.

Αναλυτικά ο κώδικας υλοποίησης αναλυτικά παρουσιάζεται στο Παράρτημα Α: Κλάση CWaveFile - αρχείο .wav.

## 1.4 Μαθηματική εισαγωγή

Παρακάτω θα παρουσιάσουμε κάποιες βασικές έννοιες και μεθόδους που χρησιμοποιήσαμε για την κατασκευή των μοντέλων μας.

### 1.4.1 Σειρές Fourier

Έστω πραγματική και ολοκληρώσιμη συνάρτηση  $f(x)$  σε διάστημα  $[a, b]$  όπου  $L = b - a$ . Υπό κατάλληλες προϋποθέσεις μπορούμε να προσεγγίσουμε τη συνάρτηση  $f(x)$  με τη βοήθεια του αθροίσματος:

$$f_N(x) = \frac{a_0}{2} + \sum_{n=1}^N \left( a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right) \right) \quad \text{Εξ. 2}$$

όπου:

$$\left\{ \begin{array}{l} a_n = \frac{2}{L} \int_a^b f(x) \cos\left(\frac{2\pi nx}{L}\right) dx \\ \text{και} \\ b_n = \frac{2}{L} \int_a^b f(x) \sin\left(\frac{2\pi nx}{L}\right) dx \end{array} \right. \quad \text{Εξ. 3}$$

Η προσέγγιση βελτιώνεται όσο αυξάνει το  $N$ , δηλαδή  $f_N \rightarrow f$  όταν  $N \rightarrow \infty$  και όταν  $N \rightarrow \infty$  η παραπάνω σειρά καλείται σειρά Fourier.

Η ανάλυση Fourier αποτελεί βασικό εργαλείο για τη μελέτη των μουσικών οργάνων, αφού οποιαδήποτε κυματομορφή που παράγεται από κάποιο όργανο, μπορεί να προσεγγιστεί με κάποια εξίσωση αντίστοιχη της Εξ. 2. (και μάλιστα με πεπερασμένο αριθμό όρων). Μπορούμε δηλαδή να προσεγγίσουμε τον ήχο του οργάνου με ένα πεπερασμένο άθροισμα ημιτόνων και συνημιτόνων.

#### 1.4.2 Μέθοδος χωριζομένων μεταβλητών

Έστω πως προσπαθούμε να υπολογίσουμε τη λύση Δ.Ε. συνάρτησης πολλών μεταβλητών (π.χ.  $u(x, t)$ ). Υπό προϋποθέσεις μπορούμε να θεωρήσουμε πως:

$$u(x, t) = X(x)T(t) \quad \text{Εξ. 4}$$

Από όπου προκύπτει πως:

$$\begin{cases} u_x(x, t) = X'(x)T(t) \\ u_t(x, t) = X(x)T'(t) \\ u_{xx}(x, t) = X''(x)T(t) \\ u_{tt}(x, t) = X(x)T''(t) \end{cases} \quad \text{Εξ. 5}$$

Αν αντικαταστήσουμε στην αρχική εξίσωση και κάνουμε τις πράξεις έτσι ώστε να καταλήξουμε σε μία εξίσωση όπου το πρώτο μέλος να είναι συνάρτηση μόνο τις μιας εξ' αυτών ενώ το δεύτερο μέλος μόνο της άλλης, να καταλήξουμε δηλαδή σε σχέση της μορφής:

$$F(X(x)) = G(T(t)) \quad \text{Εξ. 6}$$

τότε προφανώς μπορούμε να θεωρήσουμε πως και τα δύο μέλη ισούνται με κάποια σταθερά (έστω  $\lambda$ ) και να συνεχίσουμε με δύο Δ.Ε. μιας μεταβλητής.

$$\begin{cases} F(X(x)) = \lambda \\ \text{και} \\ G(T(t)) = \lambda \end{cases} \quad \text{Εξ. 7}$$

Από τις λύσεις των δύο αυτών εξισώσεων μπορούμε να υπολογίσουμε τη ζητούμενη συνάρτηση με τη βοήθεια της Εξ. 4.

Την μέθοδο χωριζομένων μεταβλητών θα την χρησιμοποιήσουμε κατά κόρον για την ανάλυση των μοντέλων μας όπως θα δούμε παρακάτω.

### 1.4.3 Μέθοδος διχοτόμησης

Έστω πως ψάχνουμε ρίζα συνεχούς συνάρτησης  $f(x)$  σε διάστημα  $[a, b]$  όπου ισχύει  $f(a)f(b) < 0$ . Τότε για να την βρούμε μπορούμε να εφαρμόσουμε τον παρακάτω αλγόριθμο:

1. Υπολογίζουμε το μέσο του διαστήματος  $c = \frac{a+b}{2}$ .
2. Αν  $f(c) \approx 0$  (αναλόγως της ακρίβειας που επιθυμούμε) τότε το  $c$  είναι η ζητούμενη ρίζα.
3. Αν  $f(a)f(c) < 0$  τότε η ρίζα βρίσκεται στο διάστημα  $[a, c]$ . Θέτουμε  $b = c$  και επαναλαμβάνουμε από το βήμα 1.
4. Αν  $f(b)f(c) < 0$  τότε η ρίζα βρίσκεται στο διάστημα  $[c, b]$ . Θέτουμε  $a = c$  και επαναλαμβάνουμε από το βήμα 1.

Η μέθοδος της διχοτόμησης μπορεί να υλοποιηθεί σχετικά εύκολα και εγγυάται την εύρεση της λύσης. Το μειονέκτημά της είναι πως η σύγκλιση μπορεί να είναι αργή.

### 1.4.4 Μέθοδος του Νεύτωνα (Newton – Raphson)

Έστω πως ψάχνουμε ρίζα παραγωγίσιμης συνάρτησης  $f(x)$  κοντά σε σημείο  $x_0$ . Τότε μπορούμε να εφαρμόσουμε τον παρακάτω επαναληπτικό αλγόριθμο:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Δηλαδή ξεκινώντας από το  $x = x_0$  υπολογίζουμε τις τιμές  $f(x_0)$  και  $f'(x_0)$  και εφαρμόζουμε τον παραπάνω τύπο για να υπολογίσουμε το  $x_1$ . Επαναλαμβάνουμε τα ίδια για τον υπολογισμό του  $x_2$  κ.ο.κ. Συνεχίζουμε έως ότου βρούμε την λύση ή φτάσουμε στη επιθυμητή ακρίβεια.

Η μέθοδος του Νεύτωνα μπορεί να συγκλίνει γρήγορα προς τη λύση επειδή κινούμαστε με βάση την εφαπτομένη. Δεν εγγυάται την εύρεση της ρίζας, μπορεί και να μην συγκλίνει. Η επιτυχία της εξαρτάται εν γένει από την επιλογή του αρχικού σημείου  $x_0$  αλλά και τη μορφή της συνάρτησης.

#### 1.4.5 Μέθοδος πεπερασμένων στοιχείων

Οι ΔΕ εκτός από αναλυτικά μπορούν να επιλυθούν και με αριθμητικές μεθόδους. Μια τέτοια μέθοδος είναι και η μέθοδος πεπερασμένων στοιχείων (ΜΠΣ) με την οποία μπορούμε να υπολογίσουμε προσεγγιστικές λύσεις ΜΔΕ. Η βασική ιδέα είναι πως χωρίζουμε το μεγάλο πρόβλημα σε πολλά μικρά κομμάτια (τα πεπερασμένα στοιχεία), λύνουμε προσεγγιστικά σε αυτά και συνδυάζουμε για να υπολογίσουμε ολόκληρη την λύση.

Αναλυτικά, εργαζόμαστε ως εξής:

- Διακριτοποίηση. Είναι η διαδικασία κατά την οποία χωρίζουμε το πεδίο μας στα πεπερασμένα στοιχεία που αναφέραμε παραπάνω.
- Τοπική προσέγγιση. Επιλογή συναρτήσεων με τις οποίες προσεγγίζουμε την λύση σε καθένα από τα στοιχεία. Διατύπωση «τοπικού» συστήματος εξισώσεων.
- Επίλυση / συναρμολόγηση.

## 2. Μοντελοποίηση του μεταλλόφωνου

### 2.1 Το όργανο

Το μεταλλόφωνο (Glockenspiel) είναι ένα κρουστό όργανο το οποίο αποτελείται από μεταλλικές μπάρες διαφορετικού μήκους τοποθετημένες δίπλα – δίπλα (συνήθως ακολουθούν τη διάταξη των πλήκτρων του πιάνου). Ο κρουστός για να παράξει ήχο, χτυπάει τις μπάρες με μπαγκέτες (Εικόνα 5) - ράβδους με κεφαλές διαφορετικών υλικών.



Εικόνα 4: Το μεταλλόφωνο

Ανήκει στα ιδιόφωνα όργανα, δηλαδή στα όργανα στα οποία η παραγωγή του ήχου οφείλεται στη δόνηση του σώματος του οργάνου καθαυτού (εν προκειμένω της κάθε μεταλλικής μπάρας). Το ύψος της παραγόμενης νότας εξαρτάται από τις φυσικές ιδιότητες του μετάλλου του, αλλά και από τις διαστάσεις της κάθε μπάρας. Το κούρδισμα του οργάνου γίνεται με την διαχείριση αυτών των δύο. Για το ίδιο μέταλλο κατασκευής όσο μικραίνουν οι διαστάσεις τόσο ανεβαίνει το τονικό ύψος. Όπως μπορούμε να δούμε και στην Εικόνα 4 οι μπάρες διατάσσονται από την μεγαλύτερη στην μικρότερη από τα αριστερά προς τα δεξιά, πράγμα που σημαίνει πως το τονικό ύψος αυξάνει από τα αριστερά προς τα δεξιά. Όπως αναφέραμε και παραπάνω οι μπαγκέτες έχουν κεφαλές διαφορετικών υλικών. Αναλόγως του υλικού ο παραγόμενος ήχος διαφέρει. Οι μαλακές μπαγκέτες παράγουν πιο γλυκό ήχο ενώ οι σκληρές πιο οξύ.



Εικόνα 5: Μπαγκέτες

Αν και το μεταλλόφωνο έχει χρησιμοποιηθεί πολύ σαν παιδικό όργανο / παιχνίδι, σαν όργανο έχει αρκετές δυνατότητες και χρησιμοποιείται και στην ορχήστρα, στην ενότητα των κρουστών αλλά και στις μάντες παρέλασης.

## 2.2 Το μοντέλο

### 2.2.1 Γενικά

Ένα απλό μοντέλο της λειτουργίας ενός μεταλλόφωνου έχει ως εξής:

- Χτυπάμε με μια μπαγκέτα κάποια από τις μπάρες του και το σύστημα διεγείρεται (κερδίζει ενέργεια).
- Η μπάρα αυτή πάλλεται σε συγκεκριμένες συχνότητες και παράγει ήχο.
- Ο ήχος σιγά σιγά εξασθενεί επειδή το σύστημα χάνει σταδιακά την αρχική του ενέργεια.

Για να το μοντελοποιήσουμε λοιπόν θα χρειαστεί να μελετήσουμε αυτά τα τρία κομμάτια ξεχωριστά.

### 2.2.2 Διέγερση του συστήματος

Έστω πως η κεφαλή μιας μπαγκέτας με μάζα  $m$  χτυπάει μια από τις μπάρες του οργάνου. Όταν βρεθεί σε επαφή με την μπάρα θα εμφανιστεί μια δύναμη (παρόμοια με αυτή ενός ελατηρίου) προς την αντίθετη κατεύθυνση της κίνησής της. Την δύναμη αυτή μπορούμε να την μοντελοποιήσουμε σαν ελατήριο δυσκαμψίας (σκληρότητας)  $k$ , η οποία μπορεί να θεωρηθεί σαν το «άθροισμα» της δυσκαμψίας της μπάρας και της δυσκαμψίας της κεφαλής της μπαγκέτας. Όσο διαρκεί η επαφή η μπάρα θα συμπιεστεί κατά ποσότητα  $x$ . Τα παραπάνω περιγράφονται με την παρακάτω εξίσωση:

$$m\ddot{x}(t) + kx(t) = 0 \quad \text{Εξ. 8}$$

Αν θεωρήσουμε πως:

$$\Omega^2 = \frac{k}{m} \text{ και } x = e^{rt} \quad \text{Εξ. 9}$$

και αντικαταστήσουμε στην Εξ. 8 προκύπτει:

$$r^2 e^{rt} + \Omega^2 e^{rt} = 0 \Rightarrow r^2 + \Omega^2 = 0 \quad \text{Εξ. 10}$$

με λύσεις τις:

$$r = \pm i\Omega \quad \text{Εξ. 11}$$

Επομένως η λύση της Εξ. 8 θα είναι:

$$x(t) = c_1 \cos(\Omega t) + c_2 \sin(\Omega t) \quad \text{Εξ. 12}$$

Η τελευταία μπορεί να γραφεί και ως:

$$x(t) = A \cos(\Omega t + \varphi) \quad \text{Εξ. 13}$$

με  $A = \sqrt{c_1^2 + c_2^2}$  και  $\varphi = \arctan\left(\frac{c_2}{c_1}\right)$  σταθερές οι οποίες εξαρτώνται από τις αρχικές συνθήκες του προβλήματος. Αν θεωρήσουμε πως την χρονική στιγμή  $t = 0$  είναι  $x(0) = 0$  και  $\dot{x}(t) = v_0$ , δηλαδή η μπαγκέτα έχει αρχική ταχύτητα  $v_0$ , τότε προκύπτει πως:

$$x(0) = A \cos(\varphi) \Rightarrow A = 0 \text{ ή } \cos(\varphi) = 0 \quad \text{Εξ. 14}$$

Στην περίπτωση μας το  $A$  δεν μπορεί να είναι μηδέν (δεν έχει φυσική σημασία) ενώ  $\varphi = \frac{\pi}{2} + k\pi, k \in \mathbb{Z}$ . Επίσης:

$$\dot{x}(t) = -A\Omega \sin(\Omega t + \varphi) = v_0 \quad \text{Εξ. 15}$$

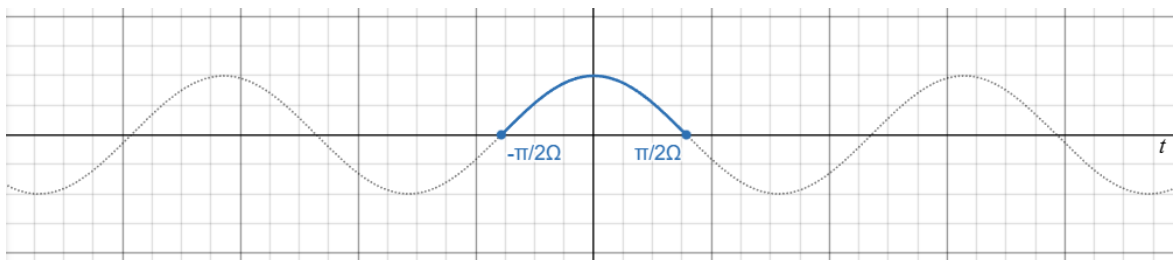
Αν θέσουμε  $\varphi = -\frac{\pi}{2}$  τότε από την τελευταία προκύπτει πως:

$$A = \frac{v_0}{\Omega} \quad \text{Εξ. 16}$$

Η δύναμη που θα εμφανιστεί λοιπόν θα ισούται με:

$$f(t) = kx(t) = kA \cos(\Omega t + \varphi) \quad \text{Εξ. 17}$$

Να σημειώσουμε σε αυτό το σημείο πως τα παραπάνω ισχύουν για τη διάρκεια της επαφής της κεφαλής της μπαγκέτας με την μπάρα. Η κεφαλή της μπαγκέτας θα πλησιάσει με κάποια ταχύτητα την μπάρα, κάποια στιγμή θα την αγγίξει, τότε θα πρωτοεμφανιστεί η παραπάνω δύναμη η οποία εν τέλει θα την κάνει να αναπηδήσει, να χάσει επαφή.



Εικόνα 6: Δύναμη κατά την επαφή

Στην Εικόνα 6 μπορούμε να δούμε την γραφική παράσταση της δύναμης ως προς το χρόνο (μπλε καμπύλη). Διαλέξαμε την χρονική στιγμή μηδέν να βρίσκεται στο μέσο του διαστήματος της επαφής και  $\varphi = 0$ . Ο συνολικός χρόνος επαφής λοιπόν θα είναι:

$$t_0 = \frac{\pi}{\Omega} = \pi \sqrt{\frac{m}{k}} \quad \text{Εξ. 18}$$

που σημαίνει πως όσο πιο σκληρή η κεφαλή τόσο μικρότερος ο χρόνος επαφής. Ρεαλιστικές τιμές της  $t_0$  είναι από 0,1 ms για σκληρές κεφαλές έως 8 ms για πολύ μαλακές.

Θέλουμε να βρούμε πως διεγείρονται οι διάφορες συχνότητες με το χτύπημα της μπαγκέτας. Για τον σκοπό αυτό θα πάρουμε τον μετασχηματισμό Fourier της δύναμης που υπολογίσαμε παραπάνω.

$$F(\omega) = \frac{kA}{2\pi} \int_{-\pi/2\Omega}^{\pi/2\Omega} \cos(\Omega t) e^{-i\omega t} dt \quad \text{Εξ. 19}$$

Με τη βοήθεια του τύπου του Euler προκύπτει:

$$\cos(\Omega t) = \text{Re} \left\{ \frac{1}{2} (e^{i\Omega t} + e^{-i\Omega t}) \right\} \quad \text{Εξ. 20}$$

Αν αντικαταστήσουμε στην παραπάνω και κάνουμε τις πράξεις προκύπτει πως:

$$F(\omega) = \frac{kA}{4\pi} \int_{-\frac{\pi}{2\Omega}}^{\frac{\pi}{2\Omega}} (e^{i(\Omega-\omega)t} + e^{-i(\Omega+\omega)t}) dt \quad \text{Εξ. 21}$$

Από την οποία καταλήγουμε στην:

$$F(\omega) = \frac{kA}{4\pi} \left[ \frac{e^{i(\Omega-\omega)t}}{i(\Omega-\omega)} - \frac{e^{-i(\Omega+\omega)t}}{i(\Omega+\omega)} \right]_{-\frac{\pi}{2\Omega}}^{\frac{\pi}{2\Omega}} = \frac{kA}{2\pi} \left( \frac{\sin \frac{\pi}{2} (1 - \frac{\omega}{\Omega})}{\Omega - \omega} + \frac{\sin \frac{\pi}{2} (1 + \frac{\omega}{\Omega})}{\Omega + \omega} \right) \Rightarrow$$

$$F(\omega) = \frac{kA}{2\pi} \cos \frac{\pi\omega}{2\Omega} \left( \frac{1}{\Omega - \omega} + \frac{1}{\Omega + \omega} \right) \quad \text{Εξ. 22}$$

Και τελικά:

$$F(\omega) = \frac{kA\Omega}{\pi} \frac{\cos \frac{\pi\omega}{2\Omega}}{\Omega^2 - \omega^2} \quad \text{Εξ. 23}$$

Αν αντικαταστήσουμε και το  $k$  από την Εξ. 9 και το  $A$  από την Εξ. 16 η τελευταία γίνεται:

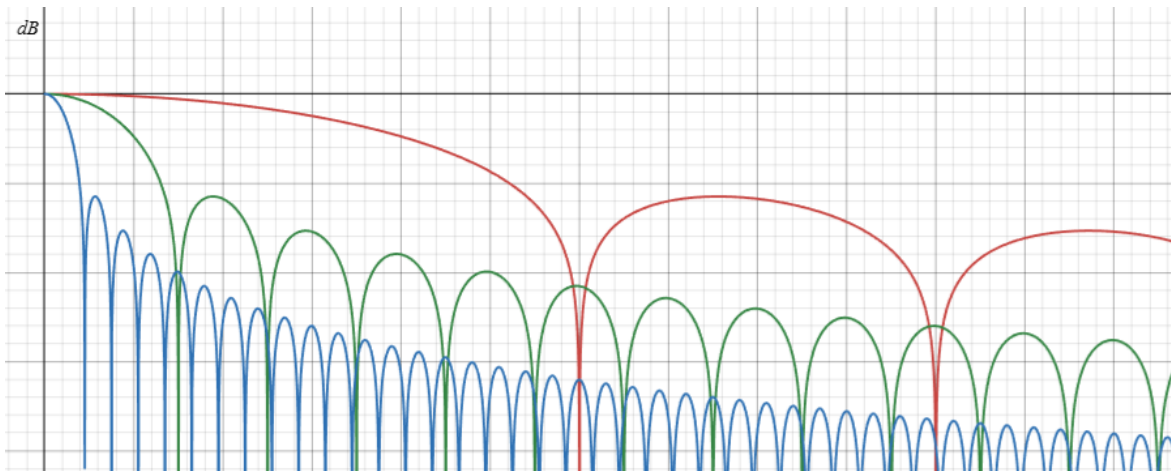
$$F(\omega) = \frac{mv_0}{\pi} \frac{\cos \left( \frac{\pi\omega}{2\Omega} \right)}{1 - \frac{\omega^2}{\Omega^2}}, \omega \neq \Omega \quad \text{Εξ. 24}$$

Και:

$$F(\Omega) = \lim_{\omega \rightarrow \pm\Omega} \frac{mv_0}{\pi} \frac{\cos \left( \frac{\pi\omega}{2\Omega} \right)}{1 - \frac{\omega^2}{\Omega^2}} = \frac{mv_0}{4} \quad \text{Εξ. 25}$$

Από την εξίσωση προκύπτει πως η  $F(\omega)$  είναι ανάλογη της ορμής της μπαγκέτας κατά την πρόσκρουση, πράγμα αναμενόμενο.

Στην παρακάτω εικόνα μπορούμε να δούμε την γραφική παράσταση σε κλίμακα dB ( $20 \log_{10} |F|$ ) για τρεις τιμές του  $\Omega$ , όπου  $\Omega_{\text{κόκκινο}} > \Omega_{\text{πράσινο}} > \Omega_{\text{μπλε}}$  (που σημαίνει πως  $k_{\text{κόκκινο}} > k_{\text{πράσινο}} > k_{\text{μπλε}}$ ). Παρατηρούμε πως όσο πιο σκληρή η μπαγκέτα τόσο περισσότερες συχνότητες διεγείρει.



Εικόνα 7: Γραφική παράσταση  $(\omega, F(\omega))$ .

### 2.2.3 Οι ιδιοσυχνότητες του οργάνου

Δεδομένου του ότι το μεταλλόφωνο αποτελείται από μπάρες τοποθετημένες η μια δίπλα στην άλλη, για να υπολογίσουμε τις ιδιοσυχνότητες του θα ξεκινήσουμε από την δυναμική εξίσωση Euler – Bernoulli δοκού, χωρίς φορτίο.

$$EI \frac{\partial^4 w(x, t)}{\partial x^4} + \rho A \frac{\partial^2 w(x, t)}{\partial t^2} = 0 \quad \text{Εξ. 26}$$

Όπου  $E$  το μέτρο ελαστικότητας,  $I$  η ροπή αδράνειας της διατομής,  $\rho A$  η γραμμική μάζα και  $w(x, t)$  η εγκάρσια μετατόπιση.

Θα προχωρήσουμε στην επίλυση με την μέθοδο των χωριζομένων μεταβλητών. Έστω πως:

$$w(x, t) = X(x)T(t) \quad \text{Εξ. 27}$$

Τότε αν αντικαταστήσουμε στην Εξ. 26 θα καταλήξουμε στην:

$$EI \frac{X^{(4)}(x)}{X(x)} = -\rho A \frac{T^{(2)}(t)}{T(t)} = \lambda \quad \text{Εξ. 28}$$

Αν θέσουμε:

$$\lambda = \rho A \omega^2 \quad \text{Εξ. 29}$$

Τότε από το πρώτο μέλος της Εξ. 28 και την Εξ. 29 προκύπτει:

$$X^{(4)}(x) - \frac{\rho A \omega^2}{EI} X(x) = 0 \quad \text{Εξ. 30}$$

Και αν ορίσουμε:

$$k^4 = \frac{\rho A \omega^2}{EI} \quad \text{Εξ. 31}$$

Καταλήγουμε στην:

$$X^{(4)}(x) - k^4 X(x) = 0 \quad \text{Εξ. 32}$$

Έστω πως:

$$X(x) = e^{rx} \quad \text{Εξ. 33}$$

Με αντικατάσταση στην Εξ. 32 προκύπτουν οι τέσσερις λύσεις:

$$r = \pm k, \pm ik \quad \text{Εξ. 34}$$

Επομένως η γενική λύση της Εξ. 32 θα είναι η:

$$X(x) = Ae^{kx} + Be^{-kx} + Ce^{ikx} + De^{-ikx} \quad \text{Εξ. 35}$$

Όπου  $A, B, C, D$  σταθερές. Η παραπάνω μπορεί να γραφεί και ως:

$$X(x) = A \cosh(kx) + B \sinh(kx) + C \cos(kx) + D \sin(kx) \quad \text{Εξ. 36}$$

Όπου τα  $A, B, C, D$  είναι σταθερές (διαφορετικές από αυτές της εξίσωσης Εξ. 35).

Από το δεύτερο μέλος της Εξ. 28 και την Εξ. 29 έχουμε:

$$-\rho A \frac{T^{(2)}(t)}{T(t)} = \rho A \omega^2 \Rightarrow T''(t) + \omega^2 T(t) = 0 \quad \text{Εξ. 37}$$

Η τελευταία έχει λύση την:

$$T(t) = A \cos(\omega t) + B \sin(\omega t) \quad \text{Εξ. 38}$$

Για δοκό της οποίας τα άκρα είναι ελεύθερα, σε αυτά δεν θα υπάρχει ροπή και διάτμηση δηλαδή αν η δοκός έχει μήκος  $L$  θα ισχύει:

$$X''(0) = 0, X'''(0) = 0, X''(L) = 0, X'''(L) = 0 \quad \text{Εξ. 39}$$

Επομένως από τις Εξ. 36 και Εξ. 39 προκύπτουν:

$$X''(0) = 0 \Rightarrow A = C \quad \text{Εξ. 40}$$

και:

$$X'''(0) = 0 \Rightarrow B = D \quad \text{Εξ. 41}$$

Με τη βοήθεια των οποίων η Εξ. 36 γίνεται:

$$X(x) = A(\cosh(kx) + \cos(kx)) + B(\sinh(kx) + \sin(kx)) \quad \text{Εξ. 42}$$

Επομένως από τις Εξ. 42 και Εξ. 39 προκύπτουν:

$$X''(L) = 0 \Rightarrow$$

$$A(\cosh(kL) - \cos(kL)) + B(\sinh(kL) - \sin(kL)) = 0$$

Εξ. 43

και:

$$X'''(L) = 0 \Rightarrow$$

$$A(\sinh(kL) + \sin(kL)) + B(\cosh(kL) - \cos(kL)) = 0$$

Εξ. 44

Αν θέσουμε  $\beta = kL$  από τις Εξ. 43 και Εξ. 44 προκύπτει:

$$\begin{cases} A(\cosh(\beta) - \cos(\beta)) + B(\sinh(\beta) - \sin(\beta)) = 0 \\ \text{και} \\ A(\sinh(\beta) + \sin(\beta)) + B(\cosh(\beta) - \cos(\beta)) = 0 \end{cases}$$

Εξ. 45

Για το τελευταίο σύστημα έχουμε:

$$\Delta(\beta) = \begin{vmatrix} \cosh(\beta) - \cos(\beta) & \sinh(\beta) - \sin(\beta) \\ \sinh(\beta) + \sin(\beta) & \cosh(\beta) - \cos(\beta) \end{vmatrix}$$

Εξ. 46

Αν κάνουμε τις πράξεις προκύπτει πως:

$$\Delta(\beta) = 1 - 2\cosh(\beta)\cos(\beta) + 1 = 2(1 - \cosh(\beta)\cos(\beta))$$

Εξ. 47

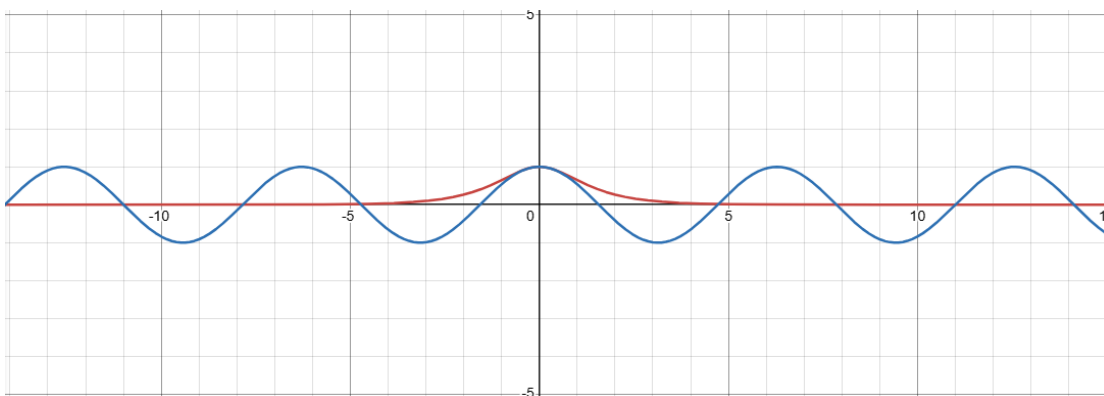
Η  $\Delta(\beta) = 0$  δίνει την χαρακτηριστική εξίσωση:

$$\cosh(\beta)\cos(\beta) = 1$$

Εξ. 48

Μια προφανή λύση της παραπάνω είναι η  $\beta = 0$ . Πέραν αυτής η τελευταία μπορεί να επιλυθεί αριθμητικά.

Στην Εικόνα 8 μπορούμε να δούμε την γραφική παράσταση των  $\cos(\beta)$  (μπλε) και  $1/\cosh(\beta)$  (κόκκινο) στην περιοχή του μηδενός.



Εικόνα 8: Γραφική παράσταση  $\cos(x)$ ,  $1/\cosh(x)$

Όπως μπορούμε να δούμε και από την γραφική παράσταση (Εικόνα 8), οι λύσεις της Εξ. 48 βρίσκονται κοντά στα σημεία όπου μηδενίζεται το συνημίτονο, δηλαδή μια πρώτη προσέγγιση μπορεί να είναι η:

$$\beta_n = \frac{(2n + 1)\pi}{2} \text{ για } n = 1, 2, 3, \dots \quad \text{Εξ. 49}$$

Εμείς θα εκμεταλλευτούμε αυτή την πληροφορία για να επιλύσουμε αριθμητικά το πρόβλημα με την μέθοδο της διχοτόμησης. Εφόσον οι λύσεις βρίσκονται κοντά στα σημεία  $n\pi + \frac{\pi}{2}$  μπορώ να τις φράξω σε διαστήματα  $[n\pi, n\pi + \pi]$ . Το υπερβολικό συνημίτονο είναι πάντα θετικό (στο  $[0, +\infty)$ ) ενώ το συνημίτονο έχει τιμή  $(-1)^n$  στα σημεία  $n\pi$ , δηλαδή στα σημεία  $n\pi$  και  $n\pi + \pi$  αλλάζει πρόσημο. Επομένως οι τιμές της  $\cosh(\beta)\cos(\beta) - 1$  είναι ετερόσημες στα άκρα αυτών των διαστημάτων και μπορούμε να εφαρμόσουμε την μέθοδο της διχοτόμησης.

Στον παρακάτω πίνακα ( Πίνακας 1) μπορούμε να δούμε τις 16 πρώτες τιμές του  $\beta$  που ικανοποιούν την Εξ. 48.

$\beta_0$	0	$\beta_8$	26.703537555508
$\beta_1$	4.730040744862	$\beta_9$	29.845130209103
$\beta_2$	7.853204624096	$\beta_{10}$	32.986722862693
$\beta_3$	10.995607838002	$\beta_{11}$	36.128315516283
$\beta_4$	14.137165491257	$\beta_{12}$	39.269908169872
$\beta_5$	17.278759657399	$\beta_{13}$	42.411500823462
$\beta_6$	20.420352245626	$\beta_{14}$	45.553093477052
$\beta_7$	23.561944902040	$\beta_{15}$	48.694686130642

**Πίνακας 1: Οι 16 πρώτες τιμές της  $\beta_n$**

Προφανώς για  $\beta_0 = 0$  δεν έχουμε ταλάντωση αλλά μεταφορική / περιστροφική κίνηση.

Έστω  $\beta_n$  ρίζα της Εξ. 48, τότε προκύπτει:

$$\beta_n = k_n L \Rightarrow \dots \Rightarrow \omega_n = \frac{\beta_n^2}{L^2} \sqrt{\frac{EI}{\rho A}} \quad \text{Εξ. 50}$$

Η τελευταία μας δίνει τις ιδιοσυχνότητες του προβλήματος. Για την συχνότητα και την γωνιακή συχνότητα ισχύει:

$$f = 2\pi\omega \quad \text{Εξ. 51}$$

Επομένως:

$$\frac{f_n}{f_1} = \frac{\omega_n}{\omega_1} \Rightarrow f_n = f_1 \frac{\omega_n}{\omega_1} \quad \text{Εξ. 52}$$

Άρα με την βοήθεια της Εξ. 50 έχουμε:

$$f_n = f_1 \frac{\beta_n^2}{\beta_1^2} \quad \text{Εξ. 53}$$

Από την τελευταία μπορούμε να υπολογίσουμε την n-οστή ιδιοσυχνότητα (overtone) αν γνωρίζουμε την θεμέλιο. Από τον Πίνακα 1 μπορούμε να αντικαταστήσουμε και να υπολογίσουμε τις 15 πρώτες ιδιοσυχνότητες. Τα αποτελέσματα παρουσιάζονται στον παρακάτω πίνακα.

$f_1$	$f_1$	$f_9$	$39.8122f_1$
$f_2$	$2.7565f_1$	$f_{10}$	$48.6349f_1$
$f_3$	$5.4039f_1$	$f_{11}$	$58.3398f_1$
$f_4$	$8.9329f_1$	$f_{12}$	$68.9271f_1$
$f_5$	$13.3442f_1$	$f_{13}$	$80.3965f_1$
$f_6$	$18.6378f_1$	$f_{14}$	$92.7483f_1$
$f_7$	$24.8137f_1$	$f_{15}$	$105.98230f_1$
$f_8$	$31.8718f_1$		

Πίνακας 2: Οι πρώτες 15 τιμές της σχετικής συχνότητας

Αν χρησιμοποιήσουμε την προσέγγιση Εξ. 49 από την Εξ. 53 προκύπτει:

$$f_n = f_1 \frac{(2n + 1)^2}{9} \quad \text{Εξ. 54}$$

Από την οποία μπορούμε υπολογίσουμε μια προσέγγιση της  $f_n$  συναρτήσεως της  $f_1$ .

### 2.2.4 Απώλεια ενέργειας

Η απώλεια ενέργειας μιας δονούμενης μπάρας γίνεται με τέσσερις τρόπους.

- Εσωτερική απόσβεση. Λόγω τριβών του ίδιου του υλικού της, μέρος της ενέργειάς της μετατρέπεται σε θερμική ενέργεια (material damping).
- Εκπομπή ήχου. Μέρος της ενέργειας της δόνησης μεταφέρεται από τα ηχητικά κύματα που δημιουργούνται (radiation damping).
- Απώλειες στα σημεία στήριξης. Και εδώ μιλάμε για απώλειες λόγω τριβών. Πολύ σημαντικό ρόλο παίζει η συγκεκριμένη κάθε φορά κατασκευή.
- Τριβή με τον αέρα (aerodynamic damping).

Οι παραπάνω μηχανισμοί δεν συμβάλλουν ισοδύναμα στην απώλεια ενέργειας. Στο πόσο συμβάλει ο καθένας παίζει σημαντικό ρόλο το υλικό κατασκευής της μπάρας. Εάν μιλάμε για ξύλο (άρα για το ξυλόφωνο) τότε ο πιο σημαντικός μηχανισμός είναι η εσωτερική απόσβεση και ακολουθούν οι απώλειες στα σημεία στήριξης, η εκπομπή του ήχου και η αεροδυναμική αντίσταση, η οποία είναι ελάχιστη. Αν μιλάμε για μέταλλο (δηλαδή για το μεταλλόφωνο) τότε ο πιο σημαντικός είναι η εκπομπή του ήχου και ακολουθούν η αεροδυναμική αντίσταση, η εσωτερική απόσβεση και οι απώλειες στα σημεία στήριξης.

Θα μοντελοποιήσουμε την εσωτερική (ιξώδη) απόσβεση (viscous damping). Έστω πως ξεκινάμε από την Δ.Ε. ταλάντωσης και προσθέτουμε μια αντίσταση ανάλογη της ταχύτητας:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = 0 \quad \text{Εξ. 55}$$

Θα αναζητήσουμε λύσεις της μορφής:

$$x(t) \propto e^{\lambda t} \quad \text{Εξ. 56}$$

Το χαρακτηριστικό πολυώνυμο είναι:

$$m\lambda^2 + c\lambda + k = 0 \quad \text{Εξ. 57}$$

Με ρίζες τις:

$$\lambda_{1,2} = \frac{-c \pm \sqrt{c^2 - 4mk}}{2m} \quad \text{Εξ. 58}$$

Σε αυτό το σημείο θα εισάγουμε κάποιους ορισμούς. Θα ορίσουμε την φυσική συχνότητα ταλάντωσης (natural angular frequency - Το  $\Omega$  της Εξ. 9) ως:

$$\omega_0 = \sqrt{\frac{k}{m}} \quad \text{Εξ. 59}$$

Τον λόγο απόσβεσης (damping ratio) ως:

$$\zeta = \frac{c}{2m\omega_0} \quad \text{Εξ. 60}$$

Την συχνότητα αποσβεσμένων ταλαντώσεων (damped angular frequency) ως:

$$\omega_d = \omega_0 \sqrt{1 - \zeta^2} \quad \text{Εξ. 61}$$

Τον συντελεστή ποιότητας  $Q$  (Quality factor) ως:

$$Q = 2\pi \frac{\text{αποθηκευμένη ενέργεια ανά κύκλο}}{\text{χαμένη ενέργεια ανά κύκλο}} \quad \text{Εξ. 62}$$

Σχέση συντελεστή ποιότητας - λόγου απόσβεσης για μικρό  $\zeta$ :

$$Q \approx \frac{1}{2\zeta} \quad \text{Εξ. 63}$$

Επανερχόμαστε στη λύση της εξίσωσης Εξ. 57. Διακρίνουμε τρεις περιπτώσεις.

**i. Υποαπόσβεση.** Αν  $\Delta = c^2 - 4mk < 0$  ή  $\zeta < 1$ . Τότε έχουμε δύο μιγαδικές ρίζες για το  $\lambda$  τις:

$$\lambda_{1,2} = -\zeta\omega_0 \pm i\omega_d \quad \text{Εξ. 64}$$

Με λύση της Εξ. 57 την:

$$x(t) = Ae^{-\zeta\omega_0 t} \cos(\omega_d t + \varphi) \quad \text{Εξ. 65}$$

**ii. Κρίσιμη απόσβεση.** Αν  $\Delta = c^2 - 4mk = 0$  ή  $\zeta = 0$ . Τότε έχουμε μια διπλή ρίζα για το  $\lambda$  την:

$$\lambda = -\omega_0 \quad \text{Εξ. 66}$$

Με λύση της Εξ. 57 την:

$$x(t) = (A + Bt)e^{-\omega_0 t} \quad \text{Εξ. 67}$$

iii. Υπεραπόσβεση. Αν  $\Delta = c^2 - 4mk > 0$  ή  $\zeta > 1$ . Σε αυτή την περίπτωση δεν έχουμε ταλάντωση αλλά μια αργή επιστροφή στο μηδέν.

Εμάς μας ενδιαφέρει η πρώτη περίπτωση όπου όπως μπορούμε να δούμε από την Εξ. 65 το πλάτος της ταλάντωσης φθίνει εκθετικά με τον χρόνο κατά έναν συντελεστή  $e^{-\zeta\omega_0 t}$ .

### 2.2.5 Το μοντέλο συνολικά

Με βάση όλα όσα είπαμε στις τρεις προηγούμενες παραγράφους μπορούμε να κατασκευάσουμε μοντέλο για το μεταλλόφωνο, το οποίο συνοπτικά περιγράφεται ως εξής:

$$f(t) = \sum_{n=1}^N A(t) a_n e^{-\frac{\pi f_n t}{Q_n}} \cos(2\pi f_n t + \varphi_n) \quad \text{Εξ. 68}$$

Αρχικά το σύστημα διεγείρεται με το χτύπημα της μπαγκέτας όπως είδαμε στην παράγραφο 2.2.2. Η ενέργεια δεν μεταφέρεται αστραπιαία αλλά χρειάζεται κάποιος χρόνος έστω  $t_0$ . Ο χρόνος αυτός εξαρτάται από την μπαγκέτα και είναι μικρός για σκληρές μπαγκέτες (της τάξης του 0,1 ms) και μεγαλύτερος για μαλακές (της τάξης των 8 ms) και μοντελοποιείται ως εξής:

$$A(t) = \begin{cases} \frac{t}{t_0}, & t \leq t_0 \\ \text{και} \\ 1, & t > t_0 \end{cases} \quad \text{Εξ. 69}$$

Επίσης και τα σχετικά πλάτη  $a_n$  εξαρτώνται από το χτύπημα της μπαγκέτας όπως είδαμε στις Εξ. 24 και Εξ. 25. Επομένως το  $a_n$  θα είναι της μορφής:

$$a_n = g(f_n) \frac{\cos\left(\frac{\pi^2 f_n}{\Omega}\right)}{1 - \frac{4\pi^2 f_n^2}{\Omega^2}} \quad \text{Εξ. 70}$$

Η απώλεια ενέργειας μοντελοποιείται με βάση την Εξ. 65 με τον όρο  $e^{-\frac{\pi f_n t}{Q_n}}$ . Όπως βλέπουμε οι ψηλότερες συχνότητες θα εξασθενίσουν πιο γρήγορα από ότι οι χαμηλότερες, όπως και περιμέναμε.

Τέλος ο όρος  $\cos(2\pi f_n t + \varphi_n)$  αναφέρεται στις ιδιοσυχνότητες του οργάνου όπως αυτές υπολογίστηκαν στην παράγραφο 2.2.3.

## 2.3 Υλοποίηση στον υπολογιστή

Η υλοποίηση στον υπολογιστή έγινε σε C++ με βάση την εξίσωση Εξ. 68. Να σημειωθεί πως οι τιμές των σταθερών, όπου χρειάστηκαν, υπολογίστηκαν για να παραχθούν ρεαλιστικά αποτελέσματα. Για παράδειγμα οι τιμές της δυσκαμψίας επιλέχθηκαν με τέτοιο τρόπο ώστε η  $A(t)$  να έχει ρεαλιστικές τιμές για σκληρές-μαλακές μπαγκέτες ( $t \in [0.1, 8], A(t) \in [0, 1]$ ) και δεν χρησιμοποιήθηκαν πραγματικές τιμές δυσκαμψίας διαφόρων υλικών.

Ο αλγόριθμος παραγωγής των δειγμάτων έχει ως εξής:

- Υπολογίζουμε το χρονικό διάστημα ενός δείγματος. Αφού γνωρίζουμε τον ρυθμό δειγματοληψίας σε  $Hz$  (δηλαδή τα δείγματα ανά δευτερόλεπτο) έστω  $sr$ , τότε το κάθε δείγμα αντιστοιχεί στο  $t_s = 1/sr$  του δευτερολέπτου.
- Υπολογίζουμε την συχνότητα της νότας την οποία θέλουμε να παράξουμε έστω  $f$  (σε  $Hz$ ).
- Από τον Πίνακα 2 μπορούμε να υπολογίσουμε τις ιδιοσυχνότητες βάση της  $f$ . Θα κρατήσουμε εκείνες για τις οποίες ισχύει  $f_i \leq sr$ , έστω  $N$  το πλήθος τους.
- Υπολογίζουμε τα  $a_n$  της Εξ. 68 για τις  $N$  συχνότητες  $f_i$  που υπολογίσαμε παραπάνω.
- Υπολογίζουμε τους αντίστοιχους συντελεστές ποιότητας  $Q_n$ .
- Υπολογίζουμε τους αντίστοιχους συντελεστές απωλειών  $\frac{\pi f_n}{Q_n}$ .
- Υπολογίζουμε τον χρόνο που χρειάζεται κάθε μια από τις ιδιοσυχνότητες ώστε το πλάτος να πέσει κάτω από ένα συγκεκριμένο όριο από την  $e^{-\frac{\pi f_n t}{Q_n}} < t_{threshold}$ . Η χρονική διάρκεια της νότας είναι το μέγιστο αυτών των διαστημάτων, έστω  $t_{tot}$ .
- Υπολογίζουμε τα αντίστοιχα  $\varphi_n$  – τα θέσαμε όλα ίσα με  $-\frac{\pi}{2}$ .
- Υπολογίζουμε την Εξ. 68 για τόσα βήματα  $n$ , για όσα ισχύει  $nt_s \leq t_{tot}$ .
- Κανονικοποιούμε το αποτέλεσμα.

Ο πλήρης κώδικας παρουσιάζεται στα παραρτήματα Β και Γ (Παράρτημα Β: Κλάσεις βάσης, Παράρτημα Γ: Ο κώδικας μεταλλόφωνου). Στο παράρτημα Β υπάρχουν οι βασικές κλάσεις (γονείς) της κλάσης CVIGlockenspiel, η οποία υλοποιεί το μεταλλόφωνο (παράρτημα Γ).

## 2.4 Αποτίμηση μοντέλου

Χρησιμοποιήσαμε το μοντέλο μας για να παράγουμε τον ήχο του μεταλλόφωνου. Δημιουργήσαμε τρία αρχεία .wav, ένα αρχείο με τη νότα ντο6, ένα με τη μείζονα κλίμακα από το ντο6 έως το ντο7 και ένα με μια μικρή μελωδία (Πίνακας 3).

Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
Glockenspiel Real C.wav	Glockenspiel Modeled C.wav	Ντο6 (C6) 1046,50 Hz
Glockenspiel Real Scale.wav	Glockenspiel Modeled Scale.wav	Ντο6 (C6) – Ντο7 (C7)
	Glockenspiel Modeled Greensleeves.mp3	Γνωστή μελωδία με το μοντελοποιημένο μεταλλόφωνο

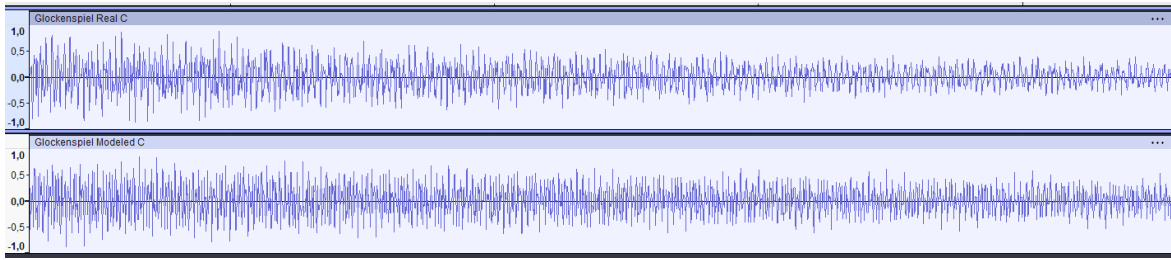
**Πίνακας 3: Τα συνοδευτικά αρχεία του μεταλλόφωνου**

Κατ' αρχάς μπορούμε να πούμε πως με το άκουσμα και μόνο το αποτέλεσμα φαίνεται αρκετά ρεαλιστικό. Με τη βοήθεια προγράμματος επεξεργασίας και εγγραφής ήχου (digital audio editor) προσπαθήσαμε να συγκρίνουμε με τρόπο πιο αντικειμενικό το μοντέλο μας με ένα πραγματικό μεταλλόφωνο. Συγκρίναμε λοιπόν την νότα ντο που παραγάγαμε με αυτή ενός πραγματικού μεταλλόφωνου. Τα αποτελέσματα της σύγκρισης παρουσιάζονται παρακάτω.



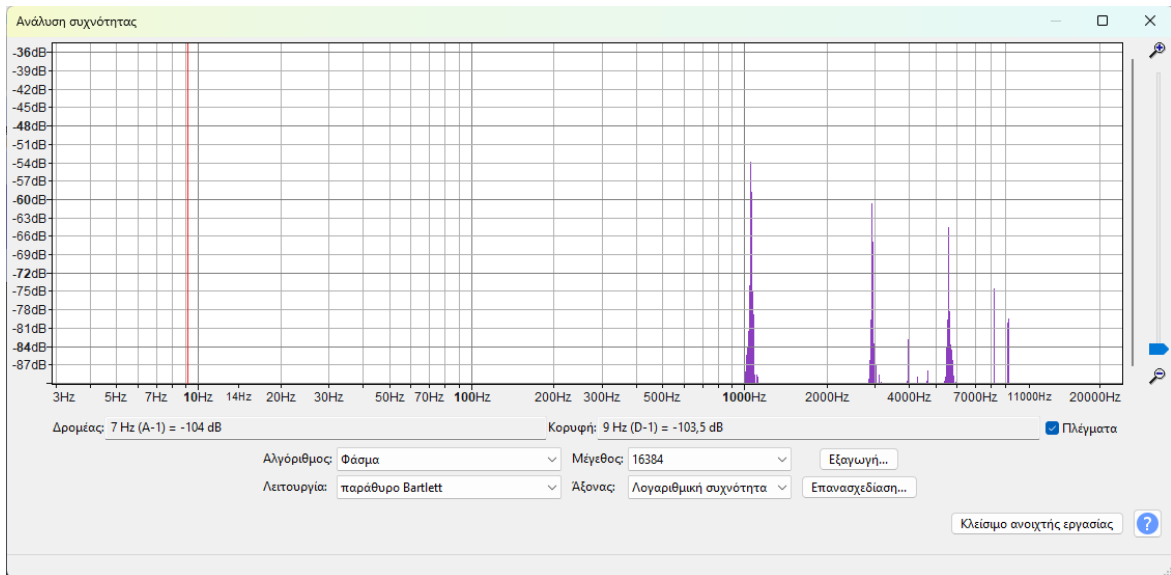
**Εικόνα 9: Γενικές κυματομορφές πραγματικού - μοντέλου μεταλλόφωνου**

Στην παραπάνω εικόνα (Εικόνα 9) βλέπουμε τη συνολική κυματομορφή πραγματικού μεταλλόφωνου (επάνω) και την αντίστοιχη του μοντέλου μας (κάτω). Γενικά η μορφή των δύο είναι αντίστοιχη, η ένταση και των δυο πέφτει εκθετικά - στο πραγματικό η πτώση είναι λίγο πιο απότομη αλλά οι δυο κυματομορφές μοιάζουν αρκετά.

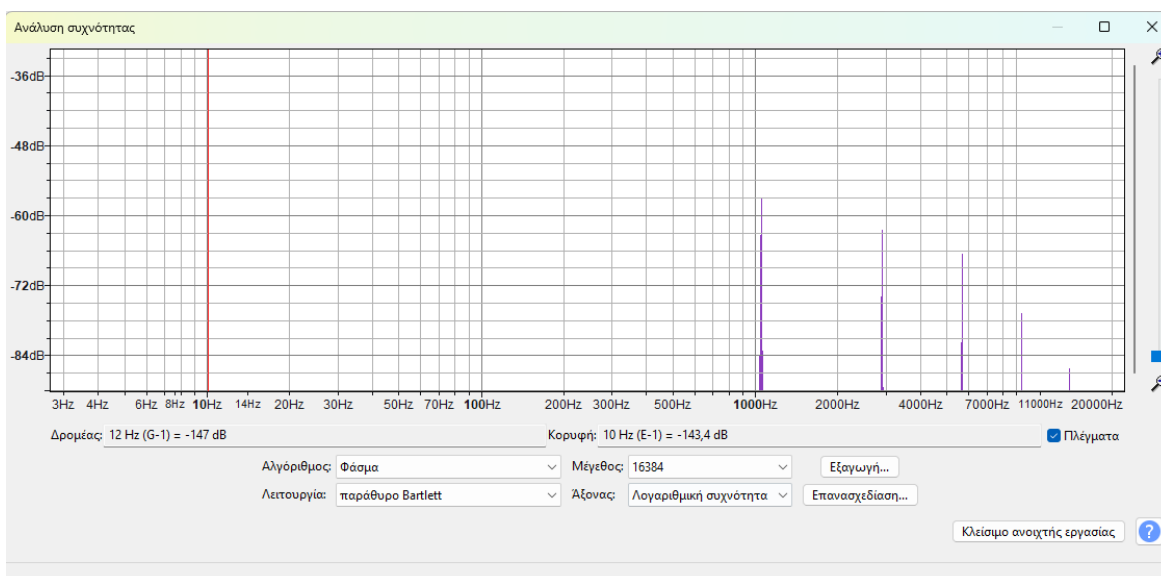


**Εικόνα 10: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου μεταλλόφωνου**

Στην Εικόνα 10 βλέπουμε ένα κομμάτι των κυματομορφών μεγεθυμένο. Και εδώ η ομοιότητα είναι εμφανής.



**Εικόνα 11: Φάσμα ντο πραγματικού μεταλλόφωνου**



Εικόνα 12: Φάσμα ντο μοντέλου μεταλλόφωνου

Στις εικόνες 11 και 12 βλέπουμε τα αντίστοιχα φάσματα. Όπως μπορούμε να δούμε και στα δύο φάσματα υπάρχουν οι τέσσερις πρώτες συχνότητες που υπολογίζουμε θεωρητικά για το ντο (1046,5 Hz, 2284,7 Hz, 5655,2 Hz και 9384,3 Hz) και οι εντάσεις τους πέφτουν αντίστοιχα. Στο πραγματικό μεταλλόφωνο εμφανίζονται και επί πλέον συχνότητες με μικρότερη ένταση.

Αν λάβουμε όλα τα παραπάνω υπόψη μπορούμε να πούμε πως το μοντέλο μας πλησιάζει αρκετά την πραγματικότητα. Τα βασικά σημεία που το διαφοροποιούν είναι τα εξής:

- Στην ανάλυση που κάναμε για τον υπολογισμό των ιδιοσυχνοτήτων θεωρήσαμε πως έχουμε ιδανική δοκό με τέλειο σχήμα. Κάτι τέτοιο δεν συμβαίνει στην πραγματικότητα. Πέραν του ότι ιδανική δοκός δεν υπάρχει, στην πράξη οι κατασκευαστές τέτοιων οργάνων χαράσσουν ή τρυπούν τις μπάρες ώστε να «βελτιώσουν» τον ήχο τους. Αυτό είναι μια εμπειρική δουλειά και το πως ακριβώς γίνεται διαφέρει από κατασκευαστή σε κατασκευαστή. Το αποτέλεσμα είναι πως στα πραγματικά όργανα, όπως μπορούμε να δούμε και στην Εικόνα 11, εμφανίζονται περισσότερες συχνότητες από όσες περιμέναμε.
- Όπως είδαμε στην παράγραφο 2.2.4 ο ακριβής υπολογισμός της απώλειας ενέργειας κάθε άλλο παρά απλός είναι και εξαρτάται από χαρακτηριστικά του εκάστοτε οργάνου, εμείς το μοντελοποιήσαμε κάπως πιο γενικά.

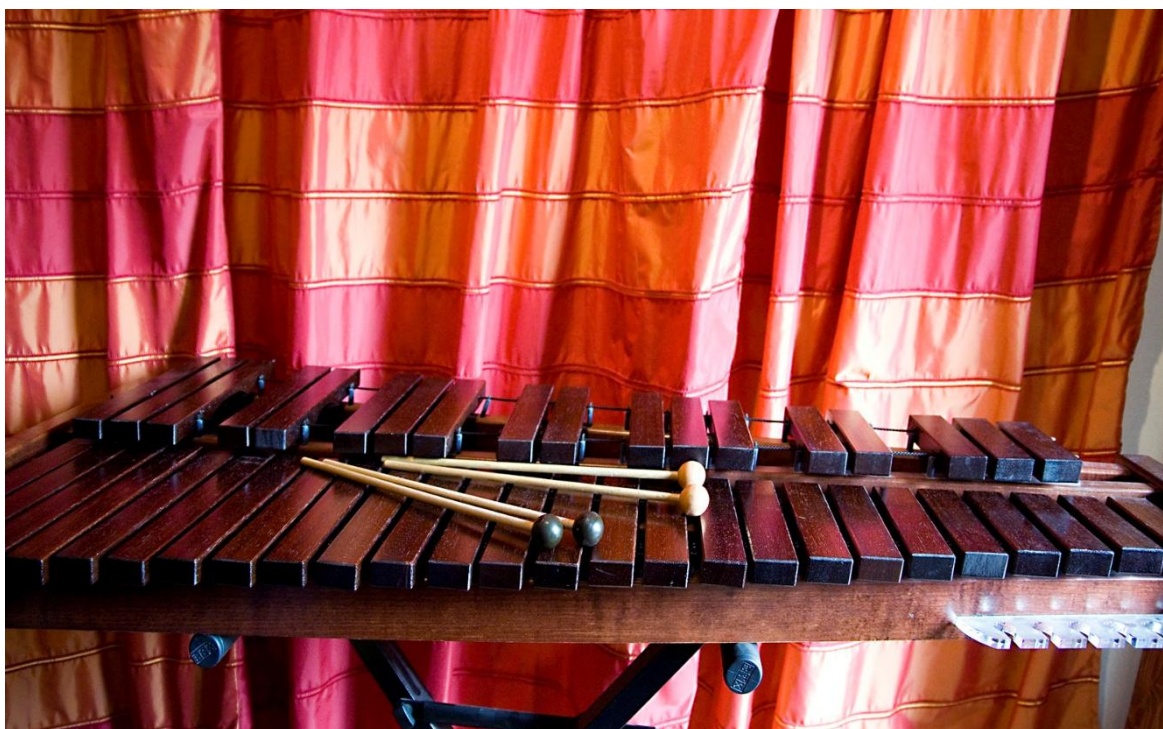
- Το άκουσμα του οργάνου εξαρτάται και από το παίξιμό του. Όπως είδαμε και στην παράγραφο 2.1 Το όργανο η κάθε μπαγκέτα διεγείρει με διαφορετικό τρόπο τις διάφορες ιδιοσυχνότητες του οργάνου αλλά και το που ακριβώς θα χτυπήσει η μπαγκέτα την μπάρα έχει το ίδιο αποτέλεσμα. Εμείς δεν λάβαμε καθόλου υπόψη το παίξιμο του οργάνου στο μοντέλο μας.
- Τέλος ο ήχος του κάθε οργάνου εξαρτάται και από τον χώρο γύρω του. Ο ήχος που φτάνει στον ακροατή δεν είναι μόνο ο ήχος που εκπέμπεται άμεσα από το όργανο αλλά και ο ήχος που ανακλάται στις επιφάνειες του χώρου γύρω του. Αυτό δίνει μια αίσθηση βάθους και έναν άλλο χαρακτήρα στο άκουσμα. Επίσης δεν ασχοληθήκαμε καθόλου με αυτά τα φαινόμενα αλλά στα ηχογραφημένα όργανα με τα οποία συγκρίναμε αυτό είναι εμφανές.

Να υπενθυμίσουμε σε αυτό το σημείο πως με το μοντέλο μας προσπαθήσαμε να πιάσουμε τα γενικά χαρακτηριστικά του ήχου του οργάνου χωρίς να μπορούμε σε πολλές λεπτομέρειες, πράγμα που ξεπερνάει τους στόχους της παρούσας εργασίας.

### 3. Μοντελοποίηση του ξυλόφωνου

#### 3.1 Το όργανο

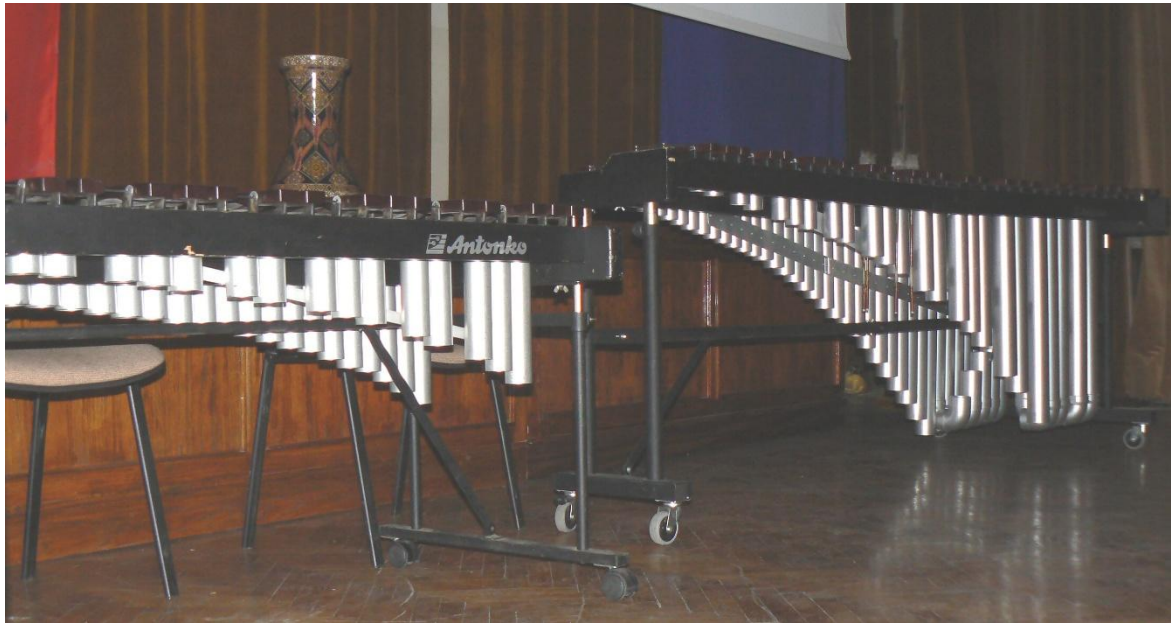
Το ξυλόφωνο (Xylophone - Εικόνα 13) είναι όργανο παρόμοιο με το μεταλλόφωνο με βασική διαφορά πως οι μπάρες του κατασκευάζονται από ξύλο και όχι από μέταλλο. Τα βασικά του χαρακτηριστικά είναι αντίστοιχα με αυτά του μεταλλόφωνου (η διάταξη των πλήκτρων, ο τρόπος παιξίματος, ο τρόπος με τον οποίο οι μπαγκέτες επιδρούν στον ήχο κτλ.) αλλά έχει δύο σημαντικές διαφορές.



Εικόνα 13: Το ξυλόφωνο

Κατ' αρχάς οι μπάρες του ξυλόφωνου «κουρδίζονται». Στην παράγραφο 2.2.3 Οι ιδιοσυχνότητες του οργάνου υπολογίσαμε τις ιδιοσυχνότητες δονούμενης μπάρας με ελεύθερα άκρα. Οι ιδιοσυχνότητες αυτές δεν είναι αρμονικές (δεν είναι ακέραια πολλαπλάσια κάποιας βασικής συχνότητας) πράγμα που δίνει αυτόν τον μεταλλικό ήχο στο όργανο. Για αισθητικούς λόγους προτιμώνται τουλάχιστον οι 2-3 πρώτες συχνότητες να είναι ακέραια πολλαπλάσια της θεμέλιας συχνότητας. Αυτό επιτυγχάνεται αφαιρώντας υλικό από τις μπάρες με κατάλληλο τρόπο.

Επί πλέον στα ξυλόφωνα συνήθως συναντάμε και αντηχεία. Σωλήνες με κλειστό το ένα άκρο κάτω από κάθε μπάρα οι οποίοι γενικά δίνουν ένταση στον ήχο, αλλά και αλλάζουν κάπως την χροιά του οργάνου αφού δεν ενισχύουν όλες τις συχνότητες με τον ίδιο τρόπο.



Εικόνα 14: Ξυλόφωνο αριστερά, μαρίμπα δεξιά

Στην παραπάνω εικόνα φαίνονται τα αντηχεία για το ξυλόφωνο αλλά και την μαρίμπα.

## 3.2 Το μοντέλο

### 3.2.1 Διέγερση συστήματος

Στην παράγραφο 2.2.2 Διέγερση του συστήματος είχαμε μελετήσει την επίδραση του χτυπήματος της μπαγκέτας στο μεταλλόφωνο. Για το ξυλόφωνο θα το ξαναδούμε με έναν εναλλακτικό τρόπο. Για το αρχικό πλάτος καθεμιάς από τις ιδιοσυχνότητες ισχύει:

$$A_n \propto \int_0^L F(x)\varphi_n(x)dx \quad \text{Εξ. 71}$$

Όπου  $F(x)$  είναι η κατανομή της δύναμης του χτυπήματος της μπαγκέτας και  $\varphi_n(x)$  η μορφή της ταλάντωσης της n-οστής ιδιοσυχνότητας. Επειδή η κεφαλή της μπαγκέτας είναι πολύ μικρή σε σχέση με τις διαστάσεις τις μπάρας, μπορούμε να θεωρήσουμε πως:

$$F(x) \approx F_0\delta(x - x_0) \quad \text{Εξ. 72}$$

με  $x_0$  την ακριβή θέση της επαφής. Επομένως αν αντικαταστήσουμε προκύπτει πως:

$$A_n \propto F_0 \int_0^L \delta(x - x_0) \varphi_n(x) dx \Rightarrow A_n \propto |\varphi_n(x_0)| \quad \text{Εξ. 73}$$

Και αν προσεγγίσουμε την  $\varphi_n(x_0)$  με το αντίστοιχο ημίτονο προκύπτει:

$$A_n \propto |\sin(n\pi x_0)| \quad \text{Εξ. 74}$$

Επίσης ισχύουν:

$$|F(\omega)| \propto \omega^{-p} \text{ και } \omega_n \propto n \quad \text{Εξ. 75}$$

Επομένως αν αντικαταστήσουμε θα έχουμε:

$$A_n \propto \frac{|\sin(n\pi x_0)|}{n^p} \quad \text{Εξ. 76}$$

### 3.2.2 Κυματική εξίσωση για αντηχείο

Έστω πως έχουμε κυλινδρικό σωλήνα μήκους  $L$  με κλειστό το ένα άκρο του. Αν ηχητικά κύματα πέσουν στο ανοιχτό άκρο του θα ισχύει η μονοδιάστατη ακουστική κυματική εξίσωση:

$$\frac{\partial^2 p}{\partial t^2} = v_0^2 \frac{\partial^2 p}{\partial x^2} \quad \text{Εξ. 77}$$

Όπου  $p$  η πίεση του αέρα,  $x \in [0, L]$  η θέση στον σωλήνα,  $t$  ο χρόνος και  $v_0 \approx 343 \text{ m/s}$  (σε κανονικές συνθήκες) η ταχύτητα του ήχου στον αέρα. Επίσης αν ξεκινήσουμε από τον ορισμό της πίεσης για την ταχύτητα, μετά από πράξεις προκύπτει πως:

$$u = -\frac{1}{i\omega\rho_0} \frac{\partial p}{\partial x} \Rightarrow u \propto -\frac{\partial p}{\partial x} \quad \text{Εξ. 78}$$

Θα λύσουμε την Εξ. 77 με την μέθοδο των χωριζομένων μεταβλητών. Θέτουμε:

$$p(x, t) = X(x)T(t) \quad \text{Εξ. 79}$$

Αν αντικαταστήσουμε στην Εξ. 77 μετά από λίγες πράξεις φθάνουμε στην:

$$\frac{T''(t)}{T(t)} = v_0^2 \frac{X''(t)}{X(t)} = -\omega^2 \quad \text{Εξ. 80}$$

Από την τελευταία προκύπτουν δυο εξισώσεις. Η χρονική εξίσωση:

$$T''(t) + \omega^2 T(t) = 0 \quad \text{Εξ. 81}$$

Η οποία έχει ως λύση την:

$$T(t) = A \cos(\omega t) + B \sin(\omega t) \quad \text{Εξ. 82}$$

με  $A, B$  σταθερές. Και την χωρική εξίσωση:

$$X''(x) + k^2 X(x) = 0 \text{ όπου } k = \frac{\omega}{v_0} \quad \text{Εξ. 83}$$

Με λύση την:

$$X(x) = C \cos(kx) + D \sin(kx) \quad \text{Εξ. 84}$$

όπου  $C, D$  σταθερές. Θεωρούμε πως για το κλειστό άκρο ισχύει  $x = 0$  ενώ για το ανοιχτό  $x = L$ . Στο κλειστό άκρο η ταχύτητα του αέρα προφανώς μηδενίζεται. Επομένως και με τη βοήθεια της Εξ. 78 έχουμε:

$$u(0, t) = 0 \Rightarrow \frac{\partial p}{\partial x}(0, t) = 0 \quad \text{Εξ. 85}$$

Και με την Εξ. 79 έχουμε:

$$\frac{\partial p}{\partial x}(0, t) = X'(0)T(t) \Rightarrow X'(0) = 0 \quad \text{Εξ. 86}$$

Επομένως αν αντικαταστήσουμε στην Εξ. 84 προκύπτει:

$$X'(0) = -Ck \sin(0) + Dk \cos(0) = 0 \Rightarrow D = 0 \quad \text{Εξ. 87}$$

Και:

$$X(x) = C \cos(kx) \quad \text{Εξ. 88}$$

Στο ανοιχτό άκρο η πίεση ισούται με την ατμοσφαιρική, επομένως:

$$p(L, t) = 0 \Rightarrow X(L) = 0 \Rightarrow C \cos(kL) = 0 \Rightarrow kL = \frac{\pi}{2}, \frac{3\pi}{2}, \frac{5\pi}{2} \dots \quad \text{Εξ. 89}$$

Δηλαδή:

$$k_n = \frac{(2n-1)\pi}{2L}, n = 1, 2, 3 \dots \quad \text{Εξ. 90}$$

Αφού  $k = \frac{\omega}{v_0}$  και  $\omega = 2\pi f$  προκύπτει πως:

$$f_n = \frac{(2n-1)v_0}{4L} = (2n-1)f_1, n = 1, 2, 3 \dots \quad \text{Εξ. 91}$$

Σύμφωνα με την παραπάνω εμφανίζονται μόνο περιττές αρμονικές. Αν συνοψίσουμε όλα τα παραπάνω καταλήγουμε στην:

$$p_n(x, t) = P_n \cos\left(\frac{(2n-1)\pi x}{2L}\right) \cos(\omega_n t + \varphi), n = 1, 2, 3 \dots \quad \text{Εξ. 92}$$

### 3.2.3 Ενεργό μήκος αντηχείου

Εμείς θεωρήσαμε πως στο  $x = L$  ισχύει  $p = 0$ , δηλαδή πως η ταλάντωση του αέρα σταματάει άμεσα. Κάτι τέτοιο όμως δεν συμβαίνει στην πραγματικότητα. Η πίεση δεν είναι ακριβώς μηδέν εμπρός στο στόμιο του σωλήνα. Στην πραγματικότητα το φαινόμενο εξελίσσεται σαν ο σωλήνας να είχε λίγο μεγαλύτερο μήκος από αυτό που πραγματικά έχει, έστω  $L_{eff}$ . Είναι δηλαδή:

$$L_{eff} = L + \Delta f \quad \text{Εξ. 93}$$

Αν αντικαταστήσουμε στην Εξ. 91 έχουμε:

$$f_n = \frac{(2n - 1)v_0}{4(L_{eff} + \Delta f)}, n = 1, 2, 3 \dots \quad \text{Εξ. 94}$$

Όπως βλέπουμε οι συχνότητες  $f_n$  είναι λίγο πιο χαμηλές από αυτές που υπολογίσαμε αρχικά. Για τους σωλήνες στα ξυλόφωνα συνήθως είναι  $\Delta L \approx 0.6r$  όπου  $r$  η ακτίνα του σωλήνα. Προφανώς και οι κατασκευαστές λαμβάνουν υπόψη την παραπάνω διαφορά και πράττουν αναλόγως.

### 3.2.4 Απώλειες αντηχείου

Σε αυτό το σημείο θα πρέπει να λάβουμε υπόψη και τις απώλειες ενέργειας. Στα αντηχεία οι απώλειες εμφανίζονται με τους ακόλουθους τρόπους:

- Εκπομπή ήχου (sound radiation). Είναι ο κυρίαρχος μηχανισμός. Τα ηχητικά κύματα που εκπέμπονται μεταφέρουν ενέργεια. Οι απώλειες αυξάνονται με την συχνότητα.
- Τριβή (friction). Κατά την ταλάντωση του αέρα μέσα στον σωλήνα, εμφανίζεται τριβή με τα τοιχώματά του, η οποία μετατρέπει μέρος της κινητικής ενέργειας των μορίων του αέρα σε θερμική. Τα τοιχώματα του σωλήνα θερμαίνονται.
- Θέρμανση αέρα (thermal losses). Η συμπίεση του αέρα του ανεβάζει την θερμοκρασία ενώ η αποσυμπίεση την κατεβάζει. Μέρος της θερμικής ενέργειας του αέρα μεταφέρεται στα τοιχώματα του σωλήνα.
- Δομικές απώλειες (structural losses). Είναι οι απώλειες που εμφανίζονται στα σημεία στήριξης των αντηχείων πάνω στο όργανο.

Θα ορίσουμε τον παράγοντα ποιότητας  $Q$ , όπως ορίστηκε στην Εξ. 62 βάσει της ενέργειας ή βάσει της συχνότητας ως:

$$Q = \frac{f_r}{\Delta f} \quad \text{Εξ. 95}$$

Όπου  $f_r$  η συχνότητα συντονισμού και  $\Delta f$  είναι το πλήρες εύρος στο ήμισυ του μεγίστου. Δηλαδή το εύρος συχνοτήτων μέσα στο οποίο η ισχύς θα πέσει στο μισό σε σχέση με το μέγιστο. Προφανώς όσο μεγαλύτερο το  $Q$  τόσο μικρότερες οι απώλειες. Αν διεγείρουμε ένα αντηχείο και το αφήσουμε ελεύθερο, τότε θα ισχύει:

$$p(t) = p_0 e^{-\frac{t}{\tau}} \cos(2\pi f_r t) \quad \text{Εξ. 96}$$

Όπου:

$$\tau = \frac{1}{\Delta f \pi} \quad \text{Εξ. 97}$$

η χρονική σταθερά απόσβεσης και:

$$Q = \pi f_r \tau \quad \text{Εξ. 98}$$

Όπως είδαμε στην εξίσωση Εξ. 91 σε ένα θεωρητικό αντηχείο διεγείρονται συγκεκριμένες και μόνο συχνότητες. Από την Εξ. 95 καταλήγουμε στην:

$$\Delta f = \frac{f_r}{Q} \quad \text{Εξ. 99}$$

Από όπου με δεδομένο ότι το  $Q$  εξαρτάται από τη συχνότητα και  $f_r \approx f_n$  προκύπτει πως:

$$\Delta f_n = \frac{f_n}{Q_n} \quad \text{Εξ. 100}$$

Δεν μιλάμε δηλαδή για μία συχνότητα αλλά για ένα εύρος συχνοτήτων με κέντρο αυτή τη συχνότητα. Το αντηχείο λοιπόν δρα λίγο – πολύ σαν φίλτρο ζώνης για καθένα από τους αρμονικούς της Εξ. 91.

### 3.2.5 Το μοντέλο συνολικά

Και το μοντέλο για το ξυλόφωνο θα έχει μορφή αντίστοιχη αυτή της Εξ. 68, με ελαφρά διαφορετικό τρόπο υπολογισμού των σταθερών, έτσι ώστε να συμπεριλάβουμε όσα αναφέραμε στις προηγούμενες παραγράφους.

$$f(t) = \sum_{n=1}^N A(t) a_n e^{-\frac{\pi f_n t}{Q_n}} \cos(2\pi f_n t + \varphi_n) \quad \text{Εξ. 101}$$

Όπως και για το μεταλλόφωνο, αρχικά το σύστημα διεγείρεται με το χτύπημα της μπαγκέτας εδώ θα χρησιμοποιήσουμε όμως την Εξ. 76 για τον υπολογισμό των σταθερών  $A_n$ . Για τον όρο  $A(t)$  ισχύουν όσα είπαμε για το μεταλλόφωνο. Και αυτός θα υπολογιστεί με την Εξ. 69. Η απώλεια ενέργειας μοντελοποιείται και εδώ με βάση την Εξ. 65 με τον όρο  $e^{-\frac{\pi f_n t}{Q_n}}$ . Τέλος ο όρος  $\cos(2\pi f_n t + \varphi_n)$  αναφέρεται στις ιδιοσυχνότητες του οργάνου, αυτή τη φορά όμως θα χρησιμοποιήσουμε τις συχνότητες που υπολογίστηκαν στην παράγραφο 2.2.3 αλλά τις τρεις πρώτες θα τις στρογγυλοποιήσουμε σε ακέραια πολλαπλάσια της θεμελίου.

### 3.3 Υλοποίηση στον υπολογιστή

Για την υλοποίηση στον υπολογιστή επεκτείναμε τον κώδικα του μεταλλόφωνου. Συγκεκριμένα δημιουργήσαμε νέα κλάση (CVIXylophone) η οποία κληρονομεί από την κλάση του μεταλλόφωνου (CVIGlockenspiel) – δηλαδή κάνει ότι και εκείνη, με μερικές διαφορές.

- Συγκεκριμένα «κουρδίζουμε» τις τρεις πρώτες ιδιοσυχνότητες σε ακέραια πολλαπλάσια της θεμελίας συχνότητας (αφού υπολογίσουμε τους πολλαπλασιαστές όπως στο μεταλλόφωνο, θέτουμε στις τρεις πρώτες τις τιμές 3, 6 και 10).
- Αντικαθιστούμε τη συνάρτηση η οποία υπολογίζει τα αρχικά πλάτη για τις ιδιοσυχνότητες του οργάνου με νέα η οποία τα υπολογίζει σύμφωνα με την Εξ. 76, δηλαδή ως  $A_n = |\sin(n\pi t)|n^p$ , με  $m$  και  $p$  σταθερές.
- Αλλάζουμε τις τιμές διάφορων σταθερών (π.χ.  $Q_n$ ) έτσι ώστε να ταιριάζουν καλύτερα στο μοντέλο του ξυλόφωνου όπως περιγράφηκε αναλυτικά στις προηγούμενες παραγράφους.

Ο πλήρης κώδικας του μοντέλου παρουσιάζεται στο παράρτημα Δ (Παράρτημα Δ: Ο κώδικας του ξυλόφωνου).

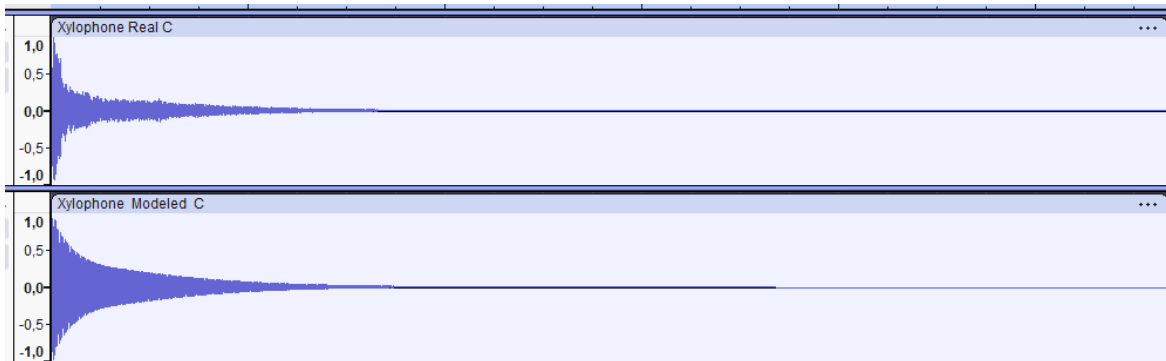
### 3.4 Αποτίμηση μοντέλου

Χρησιμοποιήσαμε το μοντέλο μας για να παράγουμε τον ήχο του ξυλόφωνου. Δημιουργήσαμε τρία αρχεία .wav, ένα αρχείο με τη νότα ντο5, ένα με τη μείζονα κλίμακα από το ντο5 έως το ντο6 και ένα με ένα άρπισμα (Πίνακας 4).

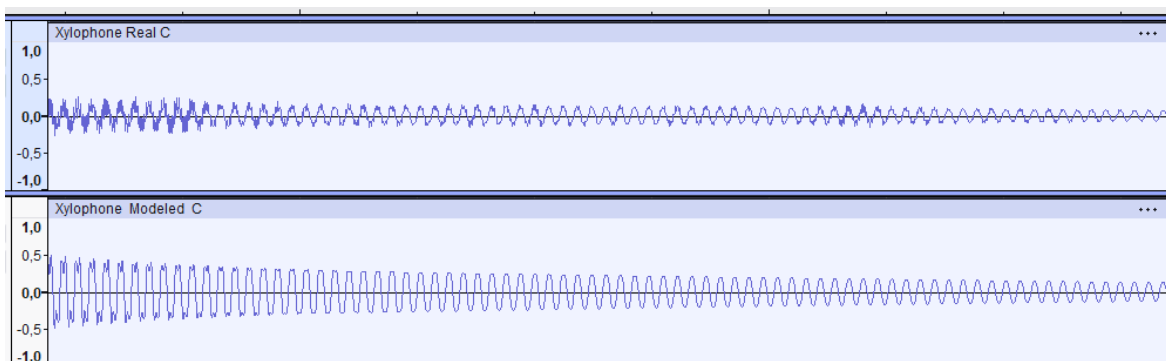
Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
Xylophone Real C.wav	Xylophone Modeled C.wav	Ντο6 (C5) 523,25 Hz
Xylophone Real Scale.wav	Xylophone Modeled Scale.wav	Ντο6 (C5) – Ντο7 (C6)
	Xylophone Modeled Arpeggio.wav	Άρπισμα με το ξυλόφωνο

**Πίνακας 4: Τα συνοδευτικά αρχεία του ξυλόφωνου**

Με τη βοήθεια προγράμματος επεξεργασίας και εγγραφής ήχου (digital audio editor) συγκρίναμε την νότα ντο που παραγάγαμε με αυτή ενός πραγματικού ξυλόφωνου. Τα αποτελέσματα της σύγκρισης παρουσιάζονται παρακάτω.



**Εικόνα 15: Γενικές κυματομορφές πραγματικού - μοντέλου ξυλόφωνου**

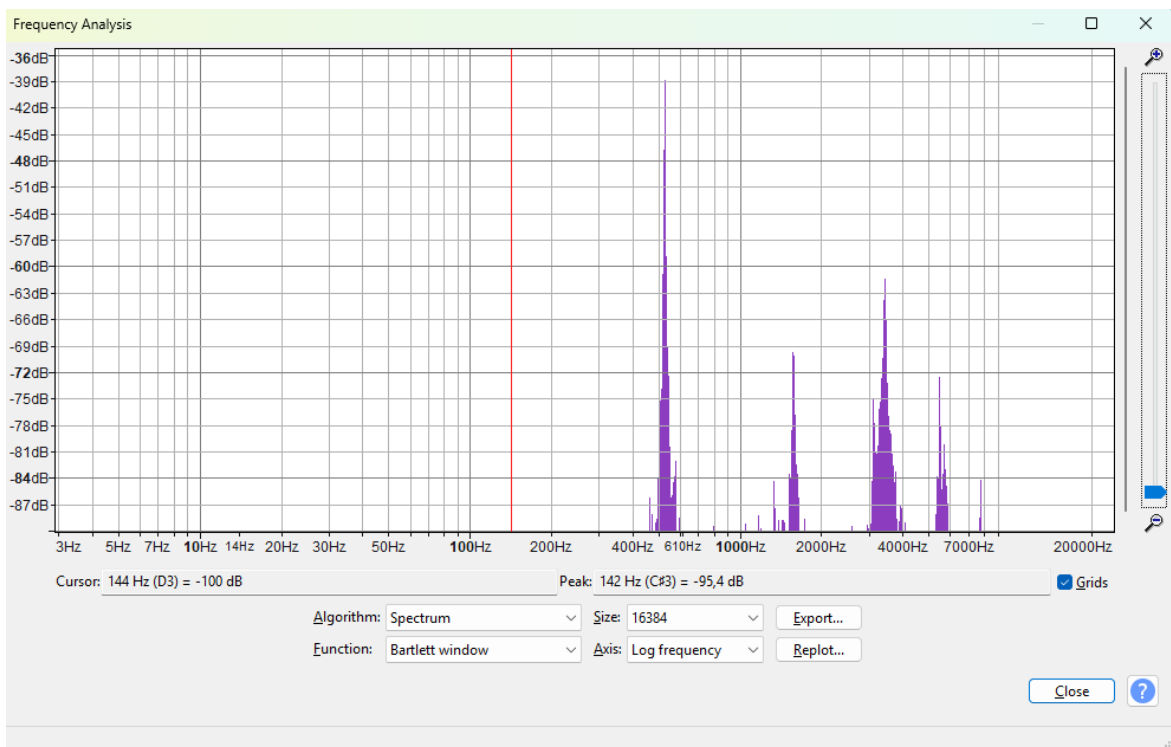


**Εικόνα 16: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου ξυλόφωνου**

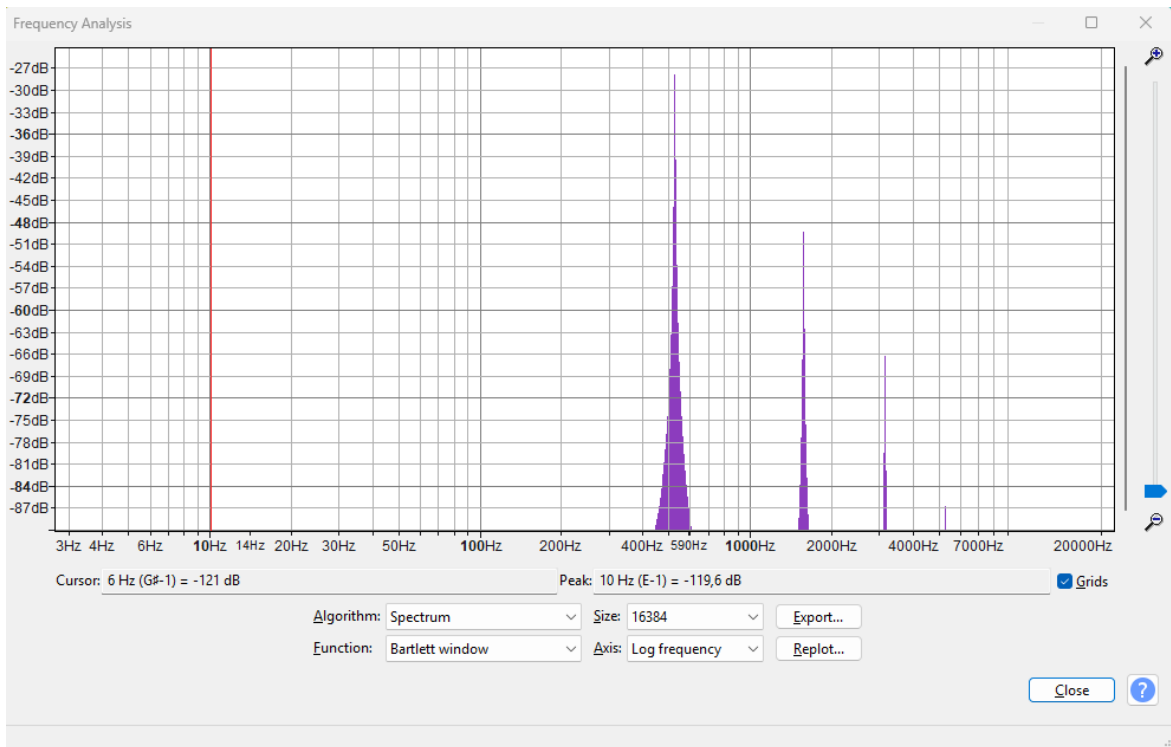
Στην Εικόνα 15 βλέπουμε την συνολική κυματομορφή πραγματικού ξυλόφωνου (επάνω) και την αντίστοιχη του μοντέλου μας (κάτω). Υπάρχει κάποια ομοιότητα, η ένταση και των δυο πέφτει εκθετικά, αλλά και εδώ στο πραγματικό όργανο η πτώση είναι λίγο πιο απότομη.

Στην Εικόνα 16 βλέπουμε ένα κομμάτι και των δύο κυματομορφών μεγεθυμένο. Και εδώ υπάρχει κάποια ομοιότητα αλλά από το μοντέλο φαίνεται πως λείπουν οι ψηλότερες συχνότητες.

Στις εικόνες Εικόνα 17 και Εικόνα 18 βλέπουμε τα αντίστοιχα φάσματα των δύο οργάνων. Οι βασικές συχνότητες είναι ίδιες. Στο πραγματικό όργανο όμως εμφανίζονται και επί πλέον συχνότητες αφού όπως ήδη έχουμε πει οι μπάρες του οργάνου δεν είναι ιδεατές κατασκευές αλλά πραγματικές, με ατέλειες αλλά και επεξεργασία (για λόγους αισθητικής) έτσι ώστε ο ήχος τους να είναι «βελτιωμένος» σε σχέση με αυτών ενός ιδεατού οργάνου.



Εικόνα 17: Φάσμα ντο πραγματικού ξυλόφωνου



Εικόνα 18: Φάσμα ντο μοντελοποιημένου ξυλόφωνου

## 4. Μοντελοποίηση της μαρίμπας

### 4.1 Το όργανο

Η μαρίμπα ανήκει στην ίδια οικογένεια με το μεταλλόφωνο και το ξυλόφωνο. Είναι όργανο κρουστό και παίζεται με μπαγκέτες. Τα βασικά του χαρακτηριστικά είναι παρόμοια με του ξυλοφώνου, αλλά έχει και κάποιες διαφορές σε σχέση με αυτό. Αν και τα πλήκτρα της είναι κατασκευασμένα επίσης από ξύλο, είναι αρκετά μεγαλύτερα από αυτά του ξυλοφώνου. Είναι επίσης περισσότερα με αποτέλεσμα να έχει μεγαλύτερο εύρος από το ξυλόφωνο (5 οκτάβες). Και τα αντηχεία του είναι μεγαλύτερα και του δίνουν έναν πιο ζεστό - πλούσιο ήχο (σύγκριση στην Εικόνα 14).



Εικόνα 19: Η μαρίμπα

Η μαρίμπα έχει μεγάλο σόλο ρεπερτόριο, σε αντίθεση με το ξυλόφωνο το οποίο έχει κυρίως συνοδευτικό χαρακτήρα. Παίζεται συνήθως με μαλακές μπαγκέτες.

### 4.2 Το μοντέλο

#### 4.2.1 Συντονισμός Helmholtz

Ο συντονισμός Helmholtz στην ακουστική (πήρε το όνομά του από τον Γερμανό φυσικό Hermann von Helmholtz), είναι το φαινόμενο κατά το οποίο ο αέρας εντός κοιλότητας πάλλεται σε συγκεκριμένη συχνότητα αν διεγερθεί. Π.χ. αν κανείς φυσήξει στο στόμιο ενός μπουκαλιού θα ακούσει ήχο κάποιας συχνότητας και όχι λευκό θόρυβο. Αυτό συμβαίνει

επειδή ο αέρας εντός της κοιλότητας λειτουργεί σαν ένα ελατήριο ενώ ο αέρας στο στόμιο της κοιλότητας λειτουργεί σαν μάζα. Έχουμε δηλαδή ένα σύστημα μάζας – ελατηρίου. Στην παρακάτω εικόνα βλέπουμε ένα αντηχείο Helmholtz όπως το σχεδίασε ο ίδιος.



Εικόνα 20: Αντηχείο Helmholtz

Έστω πως στο στόμιο του αντηχείου ο αέρας έχει συνολική μάζα:

$$m = \rho A L_{eff} \quad \text{Εξ. 102}$$

Όπου  $\rho$  η πυκνότητα του αέρα,  $A$  η διατομή του στομίου (έστω πως είναι κυλινδρικό) και  $L_{eff}$  το ενεργό μήκος του στομίου. Για μικρές αδιαβατικές μεταβολές για την πίεση στο εσωτερικό του αντηχείου ισχύει:

$$\Delta p = -\gamma \frac{p_0}{V} \Delta V \quad \text{Εξ. 103}$$

Όπου  $p$  η πίεση,  $p_0$  η ατμοσφαιρική πίεση,  $\gamma$  λόγος γραμμομοριακών ειδικών θερμοτήτων του αέρα ( $\gamma \approx 1,4$ ) και  $V$  ο όγκος του αέρα. Η δύναμη που εφαρμόζεται στην μάζα του αέρα θα είναι:

$$F = A \Delta p \quad \text{Εξ. 104}$$

Και επειδή  $\Delta V = Ax$  όπου  $x$  η μετατόπιση:

$$F = -\gamma \frac{A^2 p_0}{V} x \quad \text{Εξ. 105}$$

Ο δεύτερος νόμος του Νεύτωνα λέει πως:

$$F = m \ddot{x} \quad \text{Εξ. 106}$$

Και αν αντικαταστήσουμε την  $m$  από την Εξ. 102 και την  $F$  από την Εξ. 105 προκύπτει:

$$\begin{aligned} -\gamma \frac{A^2 p_0}{V} x &= \rho A L_{eff} \ddot{x} \Rightarrow \\ \ddot{x} + \frac{\gamma A p_0}{\rho L_{eff} V} x &= 0 \quad \text{Εξ. 107} \end{aligned}$$

Η τελευταία μπορεί να γραφτεί και ως:

$$\ddot{x} + \omega^2 x = 0, \omega = \frac{\gamma A p_0}{\rho L_{eff} V} \quad \text{Εξ. 108}$$

Επειδή  $v_s^2 = \frac{\gamma p_0}{\rho}$  ( $v_s$  η ταχύτητα του ήχου) προκύπτει πως  $\omega^2 = v_s^2 \frac{A}{L_{eff} V}$ . Αλλά και  $\omega =$

$2\pi f$  επομένως:  $f^2 = \frac{v_s^2}{4\pi^2} \frac{A}{L_{eff} V}$ , δηλαδή:

$$f_H = \frac{v_s}{2\pi} \sqrt{\frac{A}{L_{eff} V}} \quad \text{Εξ. 109}$$

Η  $f_H$  μας δίνει τη συχνότητα δόνησης του αντηχείου Helmholtz. Σε αντίθεση με το αντηχείο της παραγράφου 3.2.2 όπου εμφανίζονται πολλές συχνότητες (Εξ. 91) εδώ βλέπουμε μόνο μια. Επίσης και εδώ χρησιμοποιούμε το ενεργό μήκος  $L_{eff}$  και όχι το μήκος για τους λόγους που περιγράψαμε στην παράγραφο 3.2.3.

#### 4.2.2 Απώλειες αντηχείου

Οι απώλειες στα αντηχεία Helmholtz εμφανίζονται με τους ίδιους μηχανισμούς που εμφανίζονται στα αντηχεία σωλήνα (3.2.4 Απώλειες αντηχείου). Αν συγκεντρώσουμε όλες τις απώλειες σε έναν συντελεστή  $\alpha$ , τότε θα έχουμε:

$$\ddot{x} + \alpha\dot{x} + \omega_H^2 x = 0 \quad \text{Εξ. 110}$$

Και για τον συντελεστή ποιότητας θα ισχύει:

$$Q = \frac{\omega_H}{2\alpha} \quad \text{Εξ. 111}$$

Πειραματικά υπολογίζεται πως για την μαρίμπα το  $Q$  παίρνει τιμές από 80 για χαμηλές συχνότητες έως 15 για υψηλές. Η εξίσωση της μεταβολής του πλάτους θα είναι της μορφής:

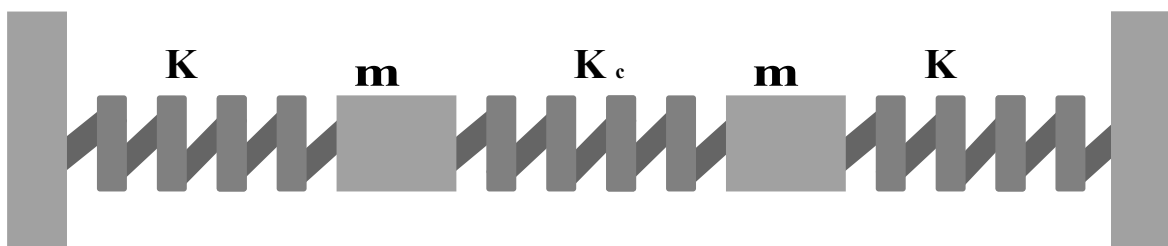
$$A(t) = A_0 e^{-\frac{\omega_H t}{2Q}} \quad \text{Εξ. 112}$$

#### 4.2.3 Αντηχεία μαρίμπας

Στο προηγούμενο κεφάλαιο μελετήσαμε τα αντηχεία σωλήνα ενώ παραπάνω είδαμε τα αντηχεία Helmholtz. Στην μοντελοποίηση της μαρίμπας χρειάζονται και οι δύο αναλύσεις αφού εμφανίζονται χαρακτηριστικά και των δύο. Συγκεκριμένα στους μακρύτερους σωλήνες κυριαρχούν τα χαρακτηριστικά των αντηχείων σωλήνα του προηγούμενου κεφαλαίου (στις χαμηλές νότες) ενώ στους πιο κοντούς σωλήνες τα χαρακτηριστικά των αντηχείων Helmholtz (στις μεσαίες και ψηλές νότες). Σε κάθε περίπτωση τα αντηχεία της μαρίμπας είναι φτιαγμένα με τέτοιο τρόπο ώστε να ενισχύουν την θεμέλιο συχνότητα κάθε νότας και όχι να παράγουν πλούσιες αρμονικές σειρές.

#### 4.2.4 Ασθενής σύζευξη

Έστω πως έχουμε δυο ίδιες μάζες ( $m$ ) συνδεδεμένες με ίδια ελατήρια ( $K$ ) σε τοίχο και με ελατήριο ( $K_c$ ) μεταξύ τους (Εικόνα 21).



**Εικόνα 21: Σύστημα μαζών - ελατηρίων**

Κάθε υποσύστημα μάζας - ελατηρίου περιγράφεται με την:

$$m\ddot{x} - kx = 0 \quad \text{Εξ. 113}$$

Η οποία μας δίνει σαν φυσική συχνότητα ταλάντωσης την:

$$\omega_0 = 2\pi f_0 = \sqrt{\frac{k}{m}} \quad \text{Εξ. 114}$$

Για ολόκληρο το σύστημα ισχύει:

$$\left\{ \begin{array}{l} m\ddot{x}_1 - kx_1 - k_c(x_1 - x_2) = 0 \\ \text{και} \\ m\ddot{x}_2 - kx_2 - k_c(x_2 - x_1) = 0 \end{array} \right\} \quad \text{Εξ. 115}$$

Όπου  $x_1, x_2$  οι αντίστοιχες μετατοπίσεις των μαζών. Αν  $k_c \ll k$  τότε μιλάμε για ασθενή σύζευξη, δηλαδή οι καθεμιά από τις μάζες κινούνται σχετικά ανεξάρτητα – το ελατήριο  $k_c$  διαταράσσει λίγο την κίνηση της κάθε μάζας.

Αν οι δύο μάζες κινούνται σε φάση, δηλαδή αν  $x_1 = x_2$ , τότε από το σύστημα Εξ. 115 προκύπτει πως:

$$\omega_1 = \sqrt{\frac{k}{m}} \quad \text{Εξ. 116}$$

Αν οι μάζες είναι εντελώς εκτός φάσης, δηλαδή αν  $x_1 = -x_2$ , τότε από το σύστημα Εξ. 115 προκύπτει πως:

$$\left\{ \begin{array}{l} m\ddot{x}_1 - (k + 2k_c)x_1 = 0 \\ \text{και} \\ m\ddot{x}_2 - (k + 2k_c)x_2 = 0 \end{array} \right\} \quad \text{Εξ. 117}$$

Και κατ' επέκταση:

$$\omega_2 = \sqrt{\frac{k + 2k_c}{m}} \quad \text{Εξ. 118}$$

Από όπου αν διαιρέσουμε κατά μέλη με την Εξ. 116 θα καταλήξουμε στην:

$$\omega_2 = \omega_1 \sqrt{1 + \frac{2k_c}{k}} \quad \text{Εξ. 119}$$

Και επειδή όπως είπαμε  $k_c \ll k$  θα είναι και  $\omega_2 \approx \omega_1$ , οι συχνότητες θα είναι δηλαδή πολύ κοντά μεταξύ τους.

Στην μαρίμπα μπορούμε να πούμε πως το σύστημα μπάρας (πλήκτρου) – αντηχείου είναι ένα ασθενώς συζευγμένο σύστημα. Οι πρωτεύοντες ταλαντωτές είναι οι μπάρες οι οποίες δονούνται εξ' αιτίας της μεταφοράς ενέργειας με τον χτύπο της μπαγκέτας. Οι ταλαντώσεις τους διεγείρουν τον αέρα στα αντηχεία όπου εμφανίζεται δευτερεύουσα ταλάντωση η οποία ανατροφοδοτεί (πολύ ασθενώς) την ταλάντωση των μπαρών. Στην πράξη (όπως είπαμε και παραπάνω) τα αντηχεία είναι κατασκευασμένα με τέτοιο τρόπο ώστε να ενισχύουν την θεμέλιο συχνότητα της κάθε μπάρας.

#### 4.2.5 Το μοντέλο συνολικά

Το μοντέλο της μαρίμπας θα στηριχθεί στο μοντέλο του ξυλόφωνου του προηγούμενου κεφαλαίου. Και εδώ θα εφαρμόσουμε την Εξ. 101 αλλά θα προσθέσουμε και έναν ακόμη όρο που θα αντιστοιχεί στον ήχο του αντηχείου κάθε νότας:

$$R(t) = A(t)a_0 e^{-\frac{\pi f_0 t}{Q}} \cos(2\pi f_0 t + \varphi_0) \quad \text{Εξ. 120}$$

Όπως βλέπουμε θα χρησιμοποιήσουμε τα στοιχεία (αρχικό πλάτος  $a_0$ , συχνότητα  $f_0$ , φάση  $\varphi_0$ ) της θεμέλιας συχνότητας – αφού αυτή ενισχύει το αντηχείο, αλλά ο συντελεστής ποιότητας  $Q$  θα είναι διαφορετικός έτσι ώστε η ένταση του να φθίνει αργότερα σε σχέση με αυτή των πλήκτρων.

### 4.3 Υλοποίηση στον υπολογιστή

Για την υλοποίηση στον υπολογιστή και για την μαρίμπα βασιστήκαμε στον κώδικα του μεταλλόφωνου. Δημιουργήσαμε μια νέα κλάση (CVIMarimba) η οποία κληρονομεί από την κλάση του μεταλλόφωνου (CVIGlockenspiel). Και εδώ «κουρδίσαμε» τις πρώτες ιδιοσυχνότητες του οργάνου, ξαναυπολογίσαμε τα αρχικά πλάτη όπως και στο ξυλόφωνο

και αλλάξαμε τις τιμές των σταθερών ώστε να ταιριάζουν καλύτερα στο όργανο. Επί πλέον όμως για τον υπολογισμό των δειγμάτων προσθέσαμε και έναν όρο που αντιστοιχεί στον ήχο του αντηχείου (Εξ. 120).

Ο πλήρης κώδικας του μοντέλου παρουσιάζεται στο παράστημα Ε (Παράρτημα Ε: Ο κώδικας της μαρίμπας).

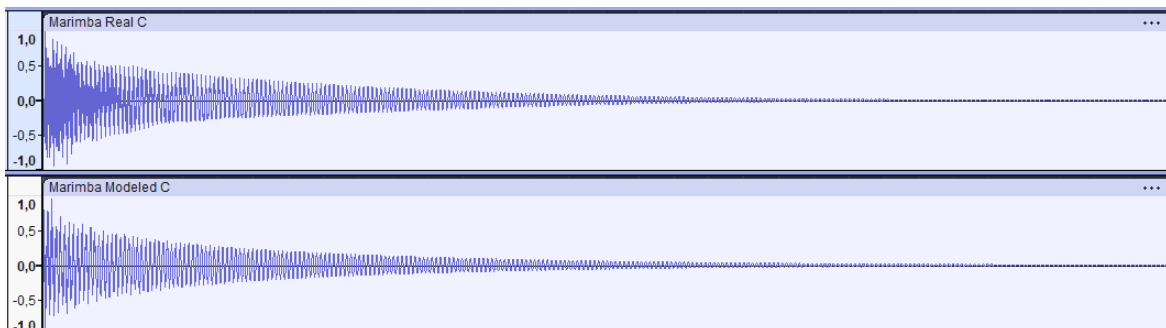
#### 4.4 Αποτίμηση μοντέλου

Με τη χρήση του μοντέλου μας αναπαρήγαμε τον ήχο της μαρίμπας. Συγκεκριμένα δημιουργήσαμε τρία αρχεία, ένα αρχείο με τη νότα ντο3, ένα με τη μείζονα κλίμακα από το ντο3 έως το ντο4 και ένα με μια μικρή μελωδία.

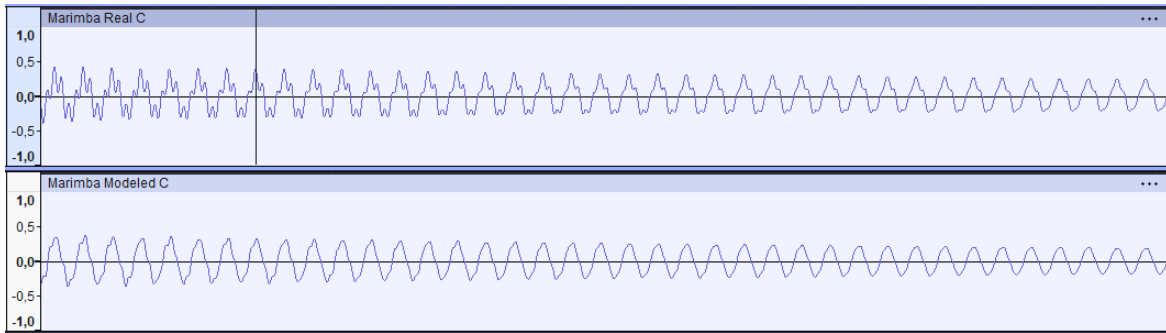
Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
Marimba Real C.mp3	Marimba Modeled C.mp3	Ντο3 (C3) 130,81 Hz
Marimba Real Scale.mp3	Marimba Modeled Scale.mp3	Ντο3 (C3) – Ντο7 (C4)
	Marimba Modeled Arpeggio.mp3	Άρπισμα

**Πίνακας 5: Τα συνοδευτικά αρχεία της μαρίμπας**

Πάντα με τη βοήθεια προγράμματος επεξεργασίας και εγγραφής ήχου συγκρίναμε τις δύο νότες ντο (πραγματική – μοντελοποιημένη). Τα αποτελέσματα της σύγκρισης μας παρουσιάζονται παρακάτω.



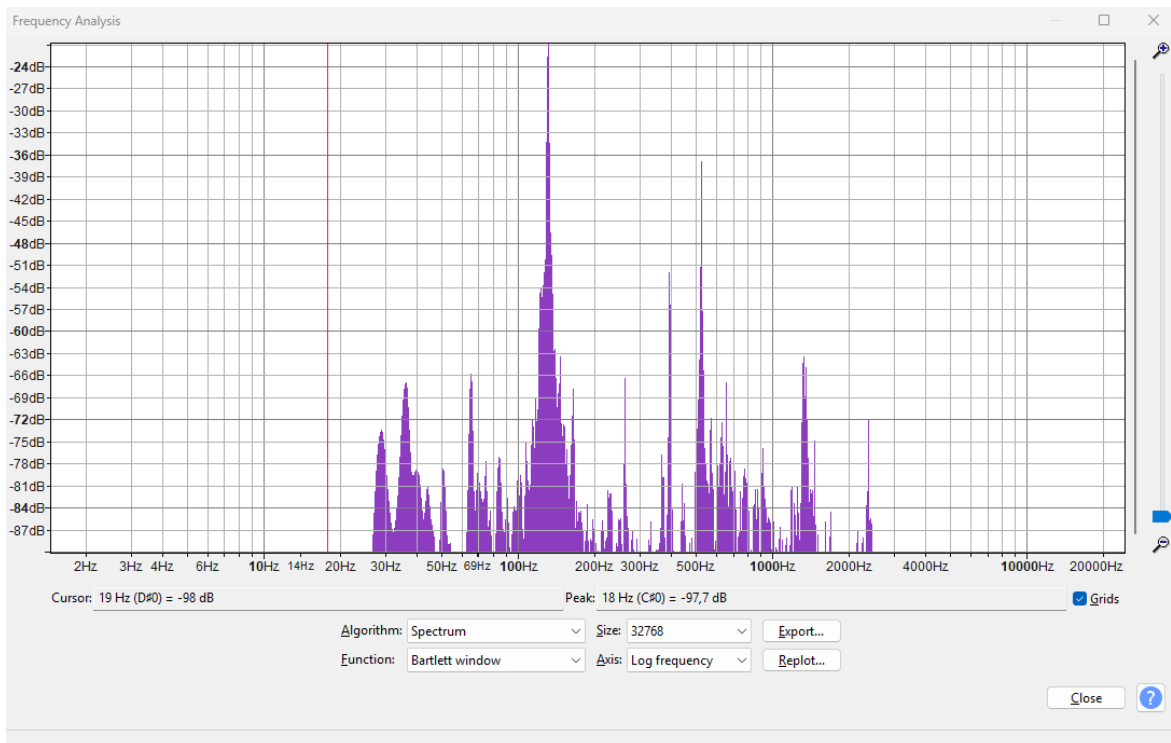
**Εικόνα 22: Γενικές κυματομορφές πραγματικής - μοντέλου μαρίμπας**



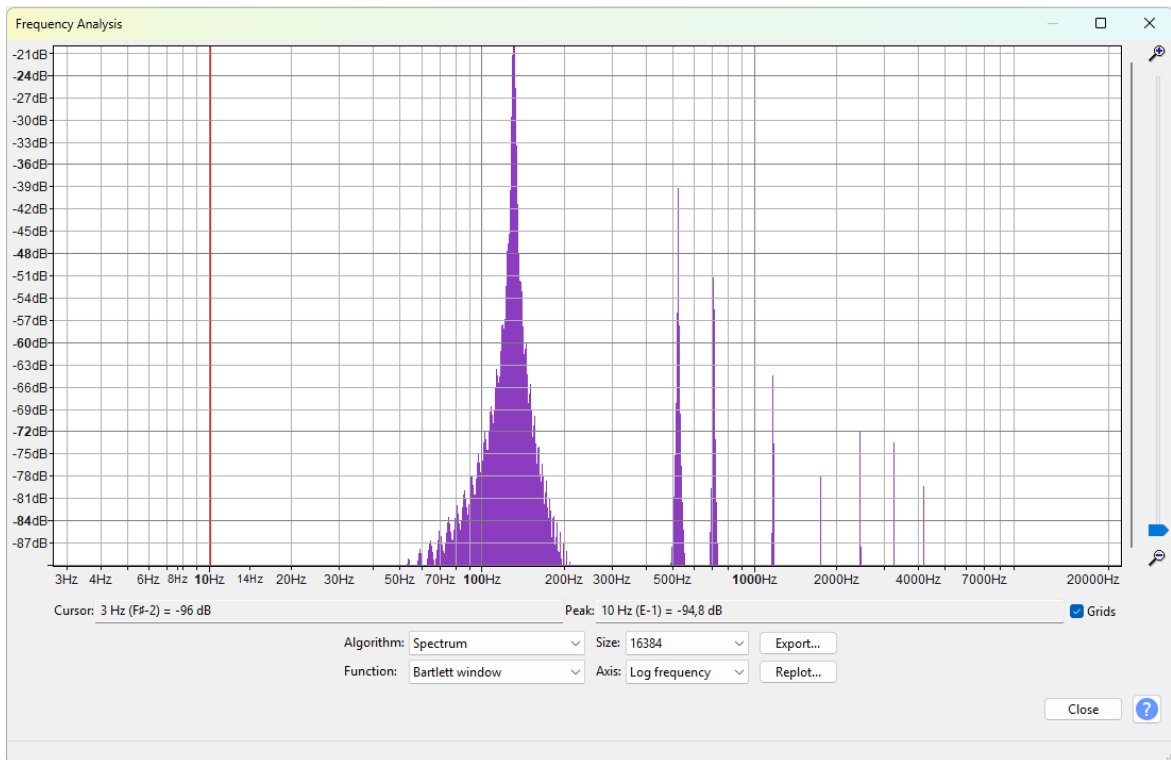
**Εικόνα 23: Λεπτομερείς κυματομορφές πραγματικής - μοντέλου μαρίμπας**

Στην Εικόνα 22 βλέπουμε την συνολική κυματομορφή μιας πραγματικής μαρίμπας (επάνω) και την αντίστοιχη του μοντέλου μας (κάτω). Στην Εικόνα 23 βλέπουμε ένα κομμάτι και των δύο κυματομορφών μεγεθυμένο. Γενικά υπάρχει κάποια ομοιότητα αλλά και διαφορές, κυρίως στις ψηλότερες συχνότητες.

Στις εικόνες Εικόνα 24 και Εικόνα 25 βλέπουμε τα αντίστοιχα φάσματα των δύο οργάνων. Το φάσμα του πραγματικού οργάνου είναι πιο πλούσιο σε αρμονικούς σε σχέση με αυτό του μοντελοποιημένου. Να σημειωθεί πως στο φάσμα του πραγματικού οργάνου βλέπουμε και συχνότητες μικρότερες της θεμελίου. Οι συχνότητες αυτές προφανώς και δεν προκύπτουν από την δόνηση της μπάρας, κάτι τέτοιο δεν θα ήταν δυνατό σύμφωνα και με την έως τώρα μελέτη μας. Οι χαμηλές αυτές συχνότητες μπορούν να προέλθουν από εξωτερικές πηγές, από τη δόνηση του σκελετού του οργάνου, από τα αντηχεία λόγω κίνησης του αέρα, μπορεί να είναι θόρυβος που προέρχεται από το χτύπημα της μπαγκέτας κ.α. Μπορεί ακόμα να εμφανίζονται λόγω της ηχογράφησης, από τη δόνηση του ίδιου του μικροφώνου ή του γερανού στο οποίο συνήθως στερεώνεται, από σύζευξη με το δάπεδο και τους τοίχους, να είναι φαινόμενο εγγύτητας κ.α.



Εικόνα 24: Φάσμα στο πραγματικής μαρίμπας



Εικόνα 25: Φάσμα στο μοντελοποιημένης μαρίμπας

## 5. Μοντελοποίηση του βιμπράφωνου

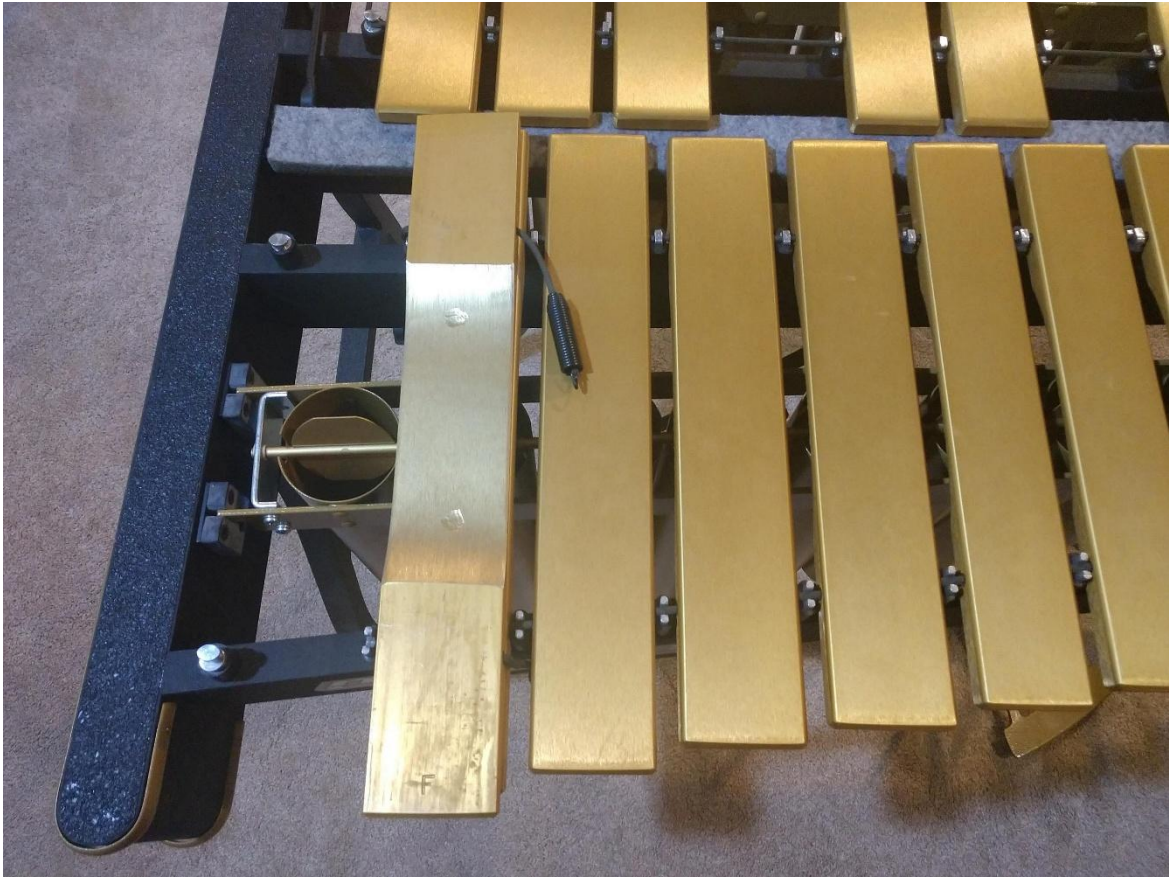
### 5.1 Το όργανο

Το βιμπράφωνο ανήκει στα κρουστά όργανα με μπαγκέτες, στην ίδια οικογένεια με τα μεταλλόφωνο, ξυλόφωνο και την μαρίμπα. Τα πλήκτρα του είναι μεταλλικά όπως του μεταλλόφωνου αλλά συνήθως είναι φτιαγμένα από αλουμίνιο. Έχει μεγαλύτερο εύρος σε σχέση με το μεταλλόφωνο (τρεις οκτάβες ή και περισσότερο F3 – F6). Έχει επίσης αντηχία, για κάθε νότα, παρόμοια με αυτά της μαρίμπας. Το εντελώς διαφορετικό που έχει είναι ένας ηλεκτροκίνητος μηχανισμός (κάτι σαν ανεμιστήρας) στο πάνω μέρος κάθε αντηχείου το οποίο μπορεί να ανοιγοκλείνει και να αυξομειώνει την ένταση κάνοντας ένα είδος βιμπράτου (εξ' ου και το όνομα). Στην παρακάτω εικόνα μπορούμε να δούμε το όργανο. Ένα κομμάτι του μηχανισμού φαίνεται στην πάνω δεξιά γωνία του οργάνου. Στην Εικόνα 27 μπορούμε να δούμε το άλλο μισό του μηχανισμού πάνω στο αντηχείο.



Εικόνα 26: Το βιμπράφωνο

Ο ήχος του είναι γλυκός, γεμάτος και χαρακτηριστικός λόγω του βιμπράτου. Είναι όργανο της ορχήστρας αλλά χρησιμοποιείται πολύ και στην τζαζ μουσική.



Εικόνα 27: Ο μηχανισμός του βιμπράφωνου

## 5.2 Το μοντέλο

### 5.2.1 Ιδιοσυχνότητες

Οι μπάρες του βιμπράφωνου «κουρδίζονται» για λόγους αισθητικής. Έχει υπολογιστεί πειραματικά πως οι ιδιοσυχνότητες έχουν συνήθως τιμές  $\times 3$ ,  $\times 6$ ,  $\times 9$ ,  $\times 14$ ,  $\times 20 \dots$  της θεμελίου. Αν δούμε τον Πίνακα 2 όπου υπολογίσαμε τις ιδιοσυχνότητες δονούμενης μπάρας με ελεύθερα άκρα, βλέπουμε πως οι ιδιοσυχνότητες του βιμπράφωνου είναι οι ίδιοι αριθμοί στρογγυλοποιημένοι προς τα πάνω στην κοντινότερη μονάδα. Το «κούρδισμα» γίνεται αφαιρώντας ή προσθέτοντας υλικό στην κάθε μπάρα.

### 5.2.2 Βιμπράτο

Όπως είπαμε το βιμπράφονο έχει έναν μηχανισμό που ανοιγοκλείνει τα αντηχεία με αποτέλεσμα να αυξομειώνεται η ένταση. Μιλάμε δηλαδή για διαμόρφωση πλάτους και όχι

συχνότητας όπως στα περισσότερα όργανα. Να σημειωθεί πως τα αντηχεία ποτέ δεν κλείνουν εντελώς. Η διαδικασία μπορεί να μοντελοποιηθεί σύμφωνα με την παρακάτω εξίσωση:

$$y_m(t) = y(t)(1 + m \sin(\omega_v t)) \quad \text{Εξ. 121}$$

Οι τιμές της συχνότητάς κυμαίνονται συνήθως ανάμεσα σε 3 – 7 Hz.

### 5.3 Υλοποίηση στον υπολογιστή

Για την υλοποίηση στον υπολογιστή και για το βιμπράφωνο βασιστήκαμε στον κώδικα του μεταλλόφωνου. Δημιουργήσαμε μια νέα κλάση (CVIVibraphone) η οποία κληρονομεί από την κλάση του μεταλλόφωνου (CVIGlockenspiel). Αρχικά κουρδίζουμε τις ιδιοσυχνότητες στρογγυλοποιώντας στην μονάδα πάνω. Επιλέγουμε άλλη τιμή για τον συντελεστή ποιότητας  $Q$  αλλά και τα αρχικά πλάτη των επί μέρους συχνοτήτων, έτσι ώστε να ταιριάζουν καλύτερα στο όργανο. Κατά των υπολογισμών των δειγμάτων προσθέτουμε ημιτονικό όρο που αντιστοιχεί στο αντηχείο, όπως κάναμε και για την μαρίμπα. Τέλος πολλαπλασιάζουμε βάση της Εξ. 121 για να προσομοιώσουμε το βιμπράτο.

Ο πλήρης κώδικας του μοντέλου παρουσιάζεται στο παράστημα ΣΤ (Παράρτημα ΣΤ: Ο κώδικας του βιμπράφωνου).

### 5.4 Αποτίμηση μοντέλου

Με τη χρήση του μοντέλου μας αναπαρήγαμε τον ήχο του βιμπράφωνου. Συγκεκριμένα δημιουργήσαμε τρία αρχεία, ένα αρχείο με τη νότα ντο4, ένα με τη μείζονα κλίμακα από το ντο4 έως το ντο5 και ένα με μια μικρή μελωδία.

Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
Vibraphone Real C.mp3	Vibraphone Modeled C.mp3	Ντο4 (C4) 261,262 Hz
Vibraphone Real Scale.mp3	Vibraphone Modeled Scale.mp3	Ντο4 (C4) – Ντο5 (C5)

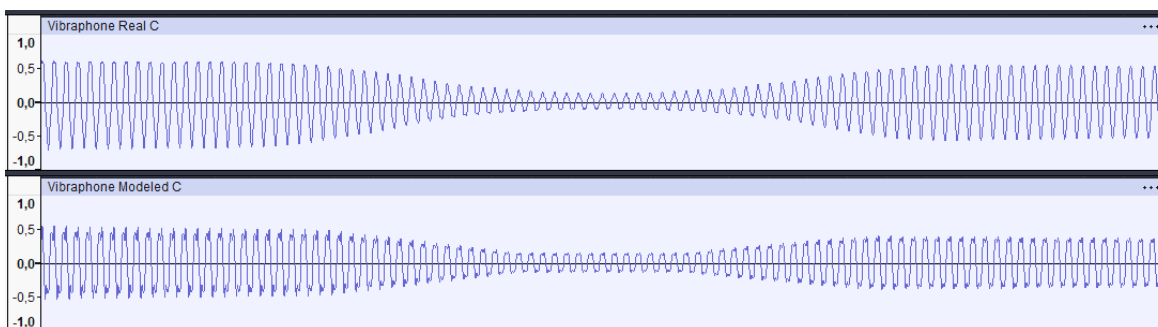
	Vibraphone Modeled Arpeggio.mp3	Άρπισμα
--	---------------------------------	---------

Πίνακας 6: Τα συνοδευτικά αρχεία του βιμπράφωνου

Συγκρίναμε τις δύο νότες ντο (πραγματική – μοντελοποιημένη). Τα αποτελέσματα της σύγκρισης μας παρουσιάζονται παρακάτω.



Εικόνα 28: Γενικές κυματομορφές πραγματικού - μοντέλου βιμπράφωνου

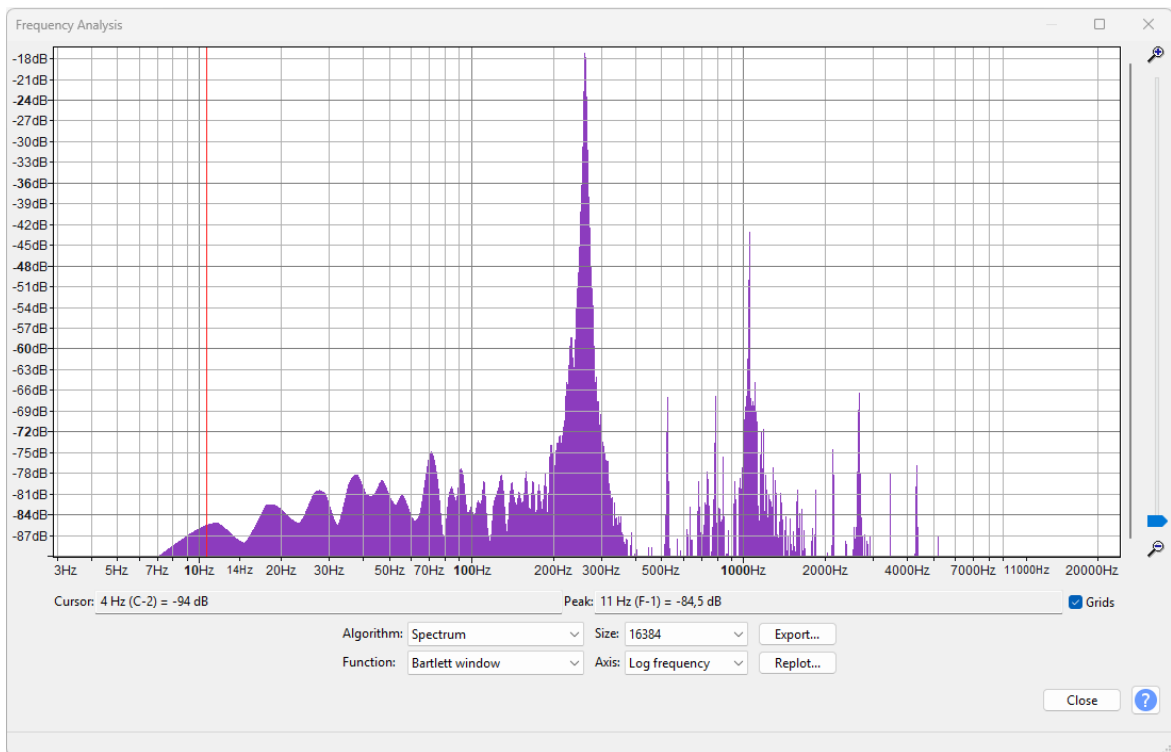


Εικόνα 29: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου βιμπράφωνου

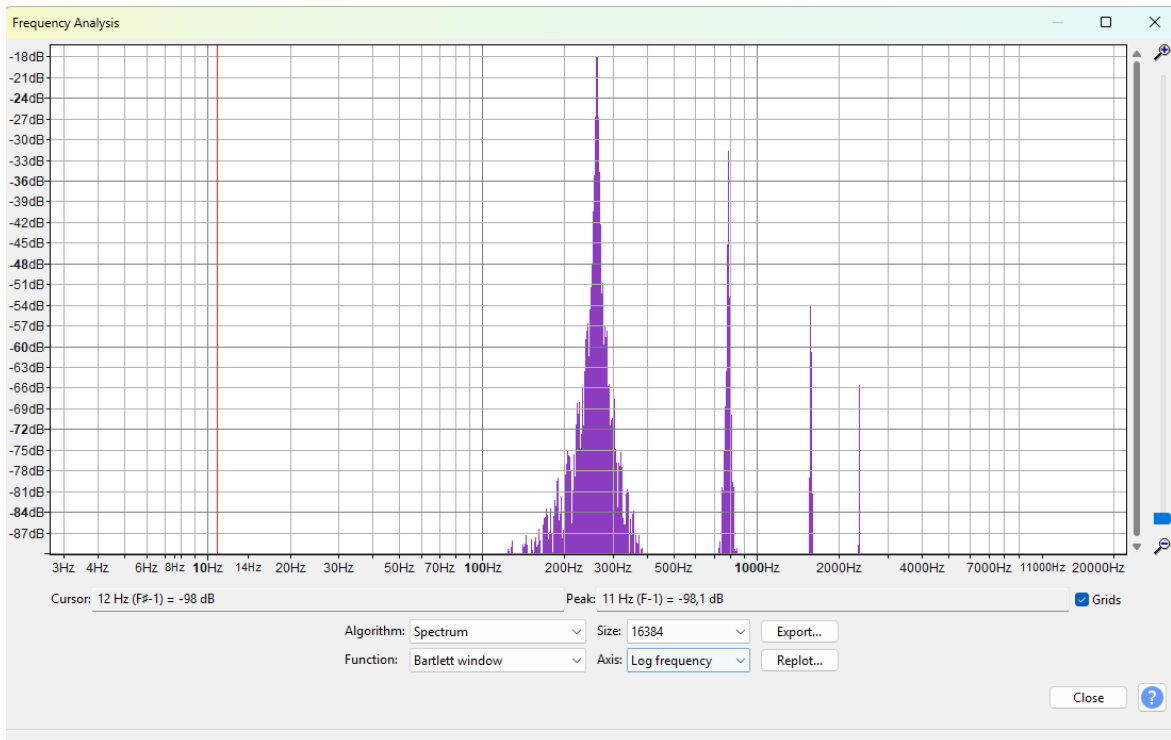
Στην Εικόνα 28 βλέπουμε την συνολική κυματομορφή πραγματικού βιμπράφωνου (επάνω) και την αντίστοιχη του μοντέλου μας (κάτω). Υπάρχει κάποια ομοιότητα, και στις δύο μπορούμε να δούμε την χαρακτηριστική μορφή του βιμπράτου με το μεταβαλλόμενο πλάτος, η ένταση και των δυο πέφτει εκθετικά, αλλά προφανώς και οι δύο κυματομορφές δεν ταυτίζονται.

Στην Εικόνα 29 μπορούμε να δούμε ένα κομμάτι και των δύο κυματομορφών μεγεθυμένο. Και εδώ υπάρχει κάποια ομοιότητα αλλά μπορούμε να δούμε διαφορά στις υψηλές ψηλές συχνότητες.

Στις εικόνες Εικόνα 30 και Εικόνα 31 βλέπουμε τα αντίστοιχα φάσματα. Το πραγματικό φάσμα είναι σίγουρα πιο πλούσιο σε συχνότητες σε σχέση με το μοντέλο μας, αλλά και οι Στο πραγματικό φάσμα βλέπουμε και συχνότητες μικρότερες της θεμελίου;



Εικόνα 30: Φάσμα ντο πραγματικού βιμπράφονου



Εικόνα 31: Φάσμα ντο μοντελοποιημένου βιμπράφονου

## 6. Μοντελοποίηση του γούντμπλοκ (woodblock)

### 6.1 Το όργανο

Το γούντμπλοκ (Εικόνα 32) είναι ένα ξύλινο ιδιόφωνο μουσικό όργανο. Αποτελείται από ένα κομμάτι ξύλου το οποίο μπορεί να είναι διαφόρων σχημάτων, αλλά συνήθως ορθογώνιο με μια ή δύο σχισμές, το οποίο κρούεται με μπαγκέτες. Υπάρχουν δύο ειδών γούντμπλοκ, το κινέζικο που παράγει ακαθόριστο τονικό ύψος και το τεμπλ μπλοκ (temple block) το οποίο κουρδίζεται. Το γούντμπλοκ κατασκευάζεται σε διάφορα μεγέθη και ο παραγόμενος ήχος εξαρτάται και από το μέγεθος. Χρησιμοποιείται και στην ορχήστρα στο τμήμα των κρουστών.



Εικόνα 32: Το γούντμπλοκ

### 6.2 Το μοντέλο

Για την μοντελοποίηση του γούντμπλοκ θα εφαρμόσουμε την θεωρία που είδαμε ως αυτό το σημείο.

Για τον υπολογισμό των ιδιοσυχνοτήτων του οργάνου θα θεωρήσουμε πως είναι μια ελαστική δοκός με ελεύθερα άκρα (το οποίο είναι σωστό σε πρώτη προσέγγιση) και θα

ακολουθήσουμε την ανάλυση που κάναμε στην παράγραφο 2.2.3 Οι ιδιοσυχνότητες του οργάνου. Θα χρησιμοποιήσουμε τις ιδιοσυχνότητες του Πίνακα 2.

Για την διέγερση του συστήματος θα ακολουθήσουμε όσα είδαμε στις παραγράφους 2.2.2 Διέγερση του συστήματος και 3.2.1 Διέγερση συστήματος. Στη Εξ. 76 είχαμε υπολογίσει το αρχικό πλάτος αντιστρόφως ανάλογο προς την συχνότητα υψωμένη σε κάποια δύναμη  $p$ . Πειραματικά θα την επιλέξουμε να έχει την τιμή  $\frac{1}{3}$ , δηλαδή:

$$A_n \propto \frac{1}{f_n^{1/3}} \quad \text{Εξ. 122}$$

Για την απώλεια ενέργειας ακολουθούμε την ανάλυση της παραγράφου 2.2.4 Απώλεια ενέργειας. με την ενέργεια να φθίνει με τον όρο:

$$e^{-\frac{\pi f_n t}{Q_n}} \quad \text{Εξ. 123}$$

όπου το  $Q_n$  θα το ορίσουμε (επίσης πειραματικά) ως:

$$Q_n = \frac{Q_0}{1 + 0.4n} \quad \text{Εξ. 124}$$

με  $Q_0 \in [40,80]$ .

Επίσης μπορούμε να δούμε την σχισμή του οργάνου σαν ένα αντηχείο Helmholtz (4.2.1 Συντονισμός Helmholtz). Με βάση αυτό θα προσθέσουμε στο μοντέλο μας έναν ακόμη όρο:

$$y_H = A_H e^{-\frac{\pi f_H t}{Q_H}} \sin(2\pi f_H t) \quad \text{Εξ. 125}$$

Τέλος αν και έχουμε δει αναλυτικά πως η μπαγκέτα διεγείρει το σύστημα, να σημειώσουμε πως το αποτέλεσμα δεν είναι αρκετά ρεαλιστικό. Αυτό οφείλεται στο ότι θεωρούμε πως μιλάμε για μια ομογενή ελαστική κατασκευή πράγμα που δεν συμβαίνει στην πραγματικότητα. Σε μικροσκοπικό επίπεδο η επιφάνεια του ξύλου είναι ασύμμετρη, έχει ίνες, πόρους που παγιδεύουν τον αέρα, η ίδια η μπαγκέτα με την κίνησή της στροβιλίζει τον αέρα κ.ο.κ. Όλα αυτά (καθώς και άλλα φαινόμενα που δεν αναφέραμε) έχουν σαν αποτέλεσμα να εμφανίζονται «τυχαίες» υψηλές συχνότητες κατά την διέγερση του συστήματος. Αυτές θα προσπαθήσουμε να μοντελοποιήσουμε με έναν όρο της μορφής:

$$N(t)e^{-t/\tau_N} \quad \text{Εξ. 126}$$

όπου  $N(t)$  θα είναι μια συνάρτηση θορύβου.

Συνολικά λοιπόν το μοντέλο μας θα περιγράφεται από την:

$$y(t) = N(t)e^{-t/\tau_N} + \sum_{n=1}^N A_n e^{-\frac{\pi f_n t}{Q_n}} \sin(2\pi f_n t) + A_H e^{-\frac{\pi f_H t}{Q_H}} \sin(2\pi f_H t) \quad \text{Εξ. 127}$$

### 6.3 Υλοποίηση στον υπολογιστή

Για την υλοποίηση του μοντέλου δημιουργήσαμε την κλάση CVIWoodblock η οποία κληρονομεί από την CVIMallet, τη βάση για τα όργανα με μπαγκέτα. Η κλάση στην ουσία υλοποιεί την Εξ. 127.

Ο πλήρης κώδικας παρουσιάζεται αναλυτικά στο παράρτημα Z (Παράρτημα Z: Ο κώδικας του γούντμπλοκ).

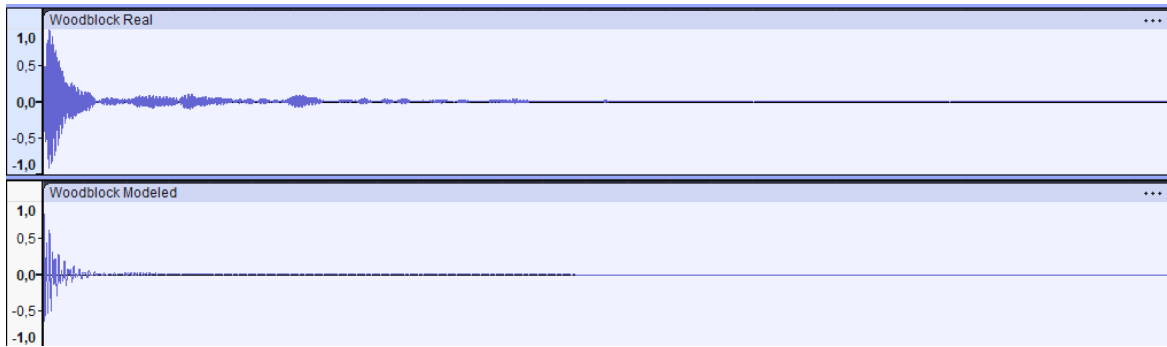
### 6.4 Αποτίμηση μοντέλου

Με το πρόγραμμά μας αναπαρήγαμε τον ήχο του γούντμπλοκ. Δημιουργήσαμε ένα αρχείο με έναν απλό χτύπο και το συγκρίναμε με τον ήχο ενός πραγματικού οργάνου. Δεδομένου ότι αφενός το όργανο δεν παράγει ήχο συγκεκριμένης συχνότητας και αφετέρου οι διαφορές ανάμεσα στον ήχο διαφορετικών οργάνων είναι αρκετά μεγάλες η σύγκριση δεν ήταν και τόσο απλή.

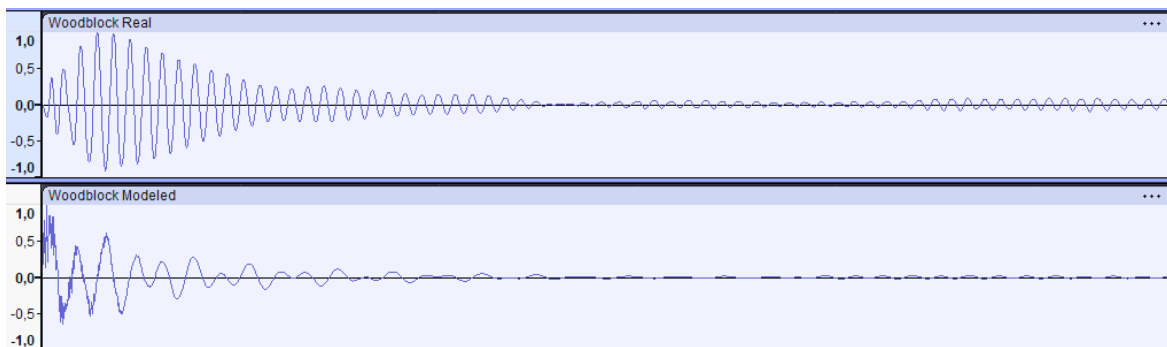
Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
Woodblock Real.mp3	Woodblock Modeled.mp3	Απλός χτύπος

Πίνακας 7: Τα συνοδευτικά αρχεία του γούντμπλοκ

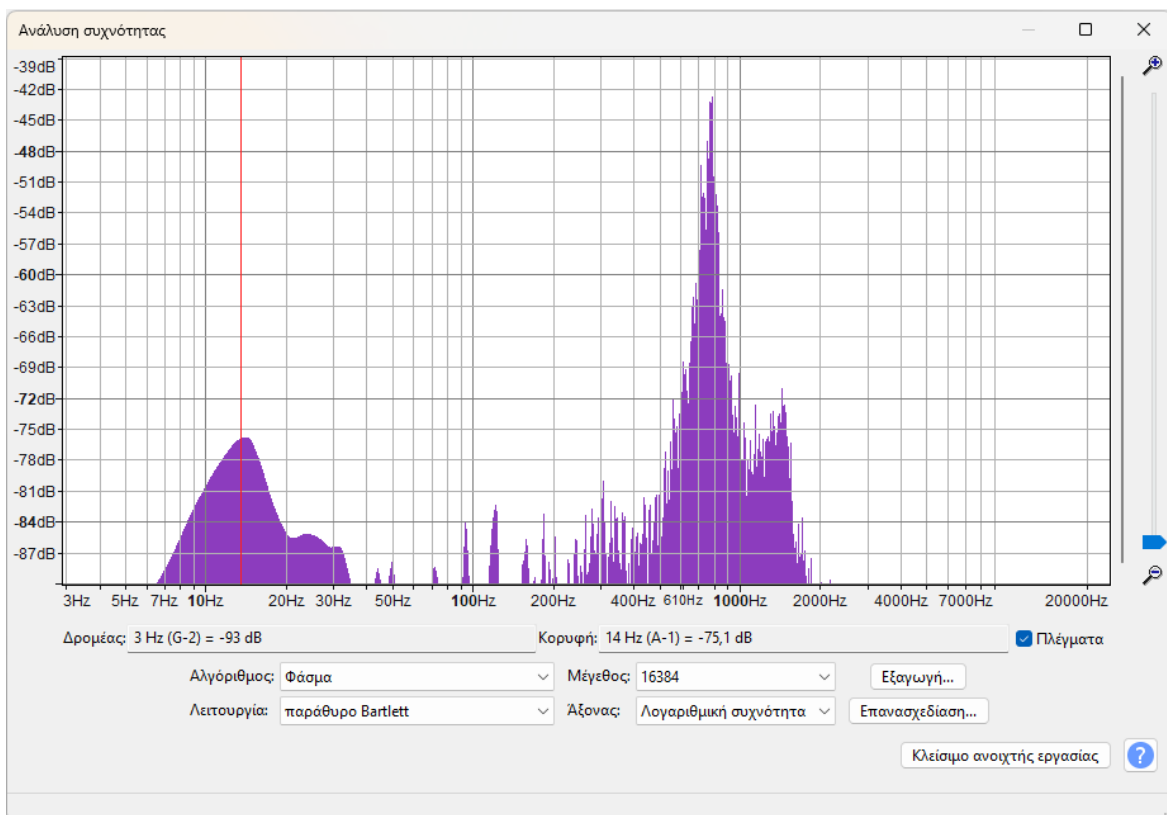
Με τη βοήθεια προγράμματος επεξεργασίας και εγγραφής ήχου συγκρίναμε τα αρχεία. Τα αποτελέσματα της σύγκρισης μας παρουσιάζονται παρακάτω.



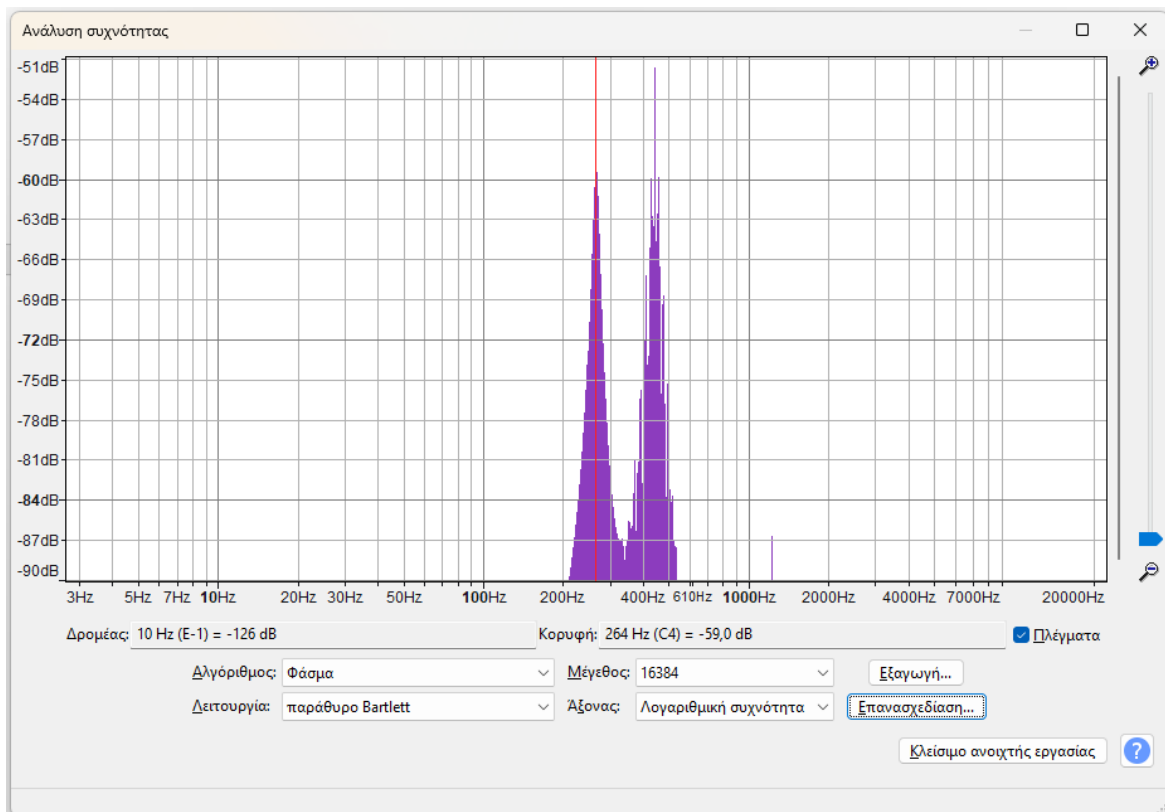
Εικόνα 33: Γενικές κυματομορφές πραγματικού - μοντέλου γούντμπλοκ



Εικόνα 34: Λεπτομερείς κυματομορφές πραγματικού - μοντέλου γούντμπλοκ



Εικόνα 35: Φάσμα πραγματικού γούντμπλοκ



**Εικόνα 36: Φάσμα μοντελοποιημένου γούντμπλοκ**

Η ομοιότητα σε αυτή την περίπτωση είναι κάθε άλλο παρά εμφανής. Οι γενικές κυματομορφές μοιάζουν σε κάποια γενικά χαρακτηριστικά αλλά όχι αρκετά. Οι λεπτομερείς κυματομορφές αλλά και τα φάσματα δεν μοιάζουν ιδιαίτερα. Παρόλα αυτά ο ήχος του μοντέλου θυμίζει τον ήχο της μπαγκέτας όταν αυτή χτυπάει ξύλο.

## 7. Μοντελοποίηση του πιάνου

### 7.1 Το όργανο

Το πιάνο (Εικόνα 37) είναι ένα από τα πιο δημοφιλή όργανα της δυτικής μουσικής.



Εικόνα 37: Το πιάνο

Κοιτάζοντάς το εξωτερικά μπορούμε να δούμε ένα ξύλινο σώμα (το οποίο λειτουργεί και ως ηχείο) το οποίο στέκεται πάνω σε ξύλινα πόδια και στην μια πλευρά έχει ένα σύνολο ασπρόμαυρων πλήκτρων. Όταν ο πιανίστας πιέσει κάποιο πλήκτρο τότε, με τη βοήθεια ενός όχι και τόσο απλού μηχανισμού, στο εσωτερικό του ένα μικρό σφυράκι χτυπάει την αντίστοιχη χορδή (έχει μια χορδή για κάθε πλήκτρο στην μπάσα περιοχή, δύο χορδές ανά πλήκτρο στη μεσαία και τρεις ανά πλήκτρο στην ψηλή) η οποία αρχίζει να πάλλεται και δημιουργεί τον ήχο ο οποίος ενισχύεται από το ηχείο. Όταν σταματάει να πιέζει το πλήκτρο τότε η χορδή μπλοκάρει και ο ήχος σταματάει. Το πιάνο λοιπόν ανήκει στα κρουστά – πληκτροφόρα όργανα. Έχει πολλές δυνατότητες παιξίματος και είναι πολυφωνικό όργανο. Αυτό που το έκανε να ξεχωρίζει από τα αντίστοιχα όργανα που προϋπήρχαν (π.χ. το τσέμπαλο) ήταν η δυνατότητά του να παίζει και χαμηλά (piano) και δυνατά (forte) από όπου πήρε και το όνομά του. Το μεγαλύτερο μέρος της δυτικής μουσικής των περασμένων αιώνων έχει γραφτεί με τη βοήθεια ενός πιάνου. Δεν είναι όργανο της ορχήστρας αλλά παίζει σε αυτή σαν φιλοξενούμενο όργανο.

## 7.2 Το μοντέλο

### 7.2.1 Κυματική εξίσωση σε μια διάσταση

Αν θεωρήσουμε μια ιδεατή χορδή μήκους  $L$  τότε η κυματική εξίσωση για αυτή θα είναι:

$$\frac{\partial^2 y}{\partial t^2} = v_s^2 \frac{\partial^2 y}{\partial x^2} \quad \text{Εξ. 128}$$

όπου  $y = y(x, t)$  η εγκάρσια μετατόπιση της χορδής και  $v_s = \sqrt{\frac{T}{\mu}}$  η ταχύτητα διάδοσης του κύματος με  $T$  η τάση της χορδής και  $\mu$  η γραμμική πυκνότητά της. Η παραπάνω μπορεί να λυθεί με την μέθοδο χωριζομένων μεταβλητών. Δηλαδή αν θέσουμε:

$$y(x, t) = X(x)T(t) \quad \text{Εξ. 129}$$

Τότε από την Εξ. 128 προκύπτει:

$$\frac{T''(t)}{T(t)} = v_s^2 \frac{X''(x)}{X(x)} = -\omega^2, \omega = \text{σταθ} \quad \text{Εξ. 130}$$

Έχουμε δηλαδή δύο εξισώσεις μιας μεταβλητής. Θα ακολουθήσουμε την ίδια διαδικασία όπως και στην παράγραφο 3.2.2 Κυματική εξίσωση για αντηχείο όπου επιλύσαμε για την κυματική εξίσωση του αντηχείου και θα καταλήξουμε στις:

$$\left\{ \begin{array}{l} T(t) = A\cos(\omega t) + B\sin(\omega t) \\ \text{και} \\ X(x) = C\cos(kx) + D\sin(kx), k = \frac{\omega}{v_s} \end{array} \right\} \quad \text{Εξ. 131}$$

Δεδομένου ότι και τα δυο άκρα της χορδής δεν μπορούν να κινηθούν, δηλαδή:

$$X(0) = 0 \text{ και } X(L) = 0 \quad \text{Εξ. 132}$$

Προκύπτει πως:

$$X_n(x) = \sin\left(\frac{n\pi x}{L}\right) \quad \text{Εξ. 133}$$

Και η λύση συνολικά θα είναι:

$$y(x, t) = \sum_{n=1}^{\infty} \left[ A_n \cos\left(\frac{n\pi x}{L}t\right) + B_n \sin\left(\frac{n\pi x}{L}t\right) \right] \sin\left(\frac{n\pi x}{L}\right) \quad \text{Εξ. 134}$$

### 7.2.2 Κυματική εξίσωση σε πραγματική χορδή

Παραπάνω είδαμε την κυματική εξίσωση για ιδεατή χορδή. Σε μια πραγματική χορδή όμως έχουμε και απόσβεση στην ταλάντωση αλλά η χορδή εμφανίζει και δυσκαμψία (σκληρότητα) τα οποία πρέπει να συνεκτιμηθούν. Η κυματική εξίσωση λοιπόν για πραγματική χορδή θα διαφέρει από αυτή της Εξ. 128 και θα είναι της μορφής:

$$\frac{\partial^2 y}{\partial t^2} + \alpha \frac{\partial y}{\partial t} = v_s^2 \frac{\partial^2 y}{\partial x^2} - \kappa \frac{\partial^4 y}{\partial x^4} \quad \text{Εξ. 135}$$

όπου  $\alpha$  ο συντελεστής απόσβεσης και  $\kappa$  η δυσκαμψία της χορδής. Και εδώ μπορούμε να προχωρήσουμε με την μέθοδο χωρισμού μεταβλητών. Θεωρούμε δηλαδή ότι:

$$y(x, t) = X(x)T(t) \quad \text{Εξ. 136}$$

Επομένως αν αντικαταστήσουμε στην Εξ. 136 μετά από κάποιες πράξεις θα έχουμε:

$$\frac{T''(t) + aT'(t)}{T(t)} = \frac{v_s^2 X''(x) - \kappa X^{(4)}(x)}{X(x)} = -\lambda \quad \text{Εξ. 137}$$

Από όπου προκύπτουν η χωρική και η χρονική εξίσωση:

$$\left\{ \begin{array}{l} T''(t) + aT'(t) + \lambda T(t) = 0, (\alpha) \\ \text{και} \\ \kappa X^{(4)}(x) - v_s^2 X''(x) + \lambda X(x) = 0, (\beta) \end{array} \right\} \quad \text{Εξ. 138}$$

Θα ξεκινήσουμε με την χωρική εξίσωση. Αν θέσουμε  $X(x) = e^{rx}$  και αντικαταστήσουμε στην Εξ. 138β θα καταλήξουμε στην:

$$\kappa r^4 - v_s^2 r^2 + \lambda = 0 \quad \text{Εξ. 139}$$

Αν θέσουμε  $m = r^2$  και λύσουμε ως προς  $m$  προκύπτει:

$$m_{1,2} = \frac{v_s^2 \pm \sqrt{v_s^4 - 4\kappa\lambda}}{2\kappa} \quad \text{Εξ. 140}$$

Και:

$$r = \pm \sqrt{m_{1,2}} \quad \text{Εξ. 141}$$

Επομένως:

$$X(x) = e^{\pm \sqrt{m_{1,2}} x} \quad \text{Εξ. 142}$$

Έχουμε όμως και τις οριακές συνθήκες  $X(0) = X(L) = 0$  οι οποίες για να ικανοποιηθούν πρέπει  $m_{1,2} < 0 \Rightarrow m_{1,2} = -k_{1,2}^2$ , δηλαδή:

$$r_{1,2} = \pm i k_{1,2} \quad \text{Εξ. 143}$$

Επομένως η γενική λύση της Εξ. 138β θα είναι η:

$$X(x) = A \cos(k_1 x) + B \sin(k_1 x) + C \cos(k_2 x) + D \sin(k_2 x) \quad \text{Εξ. 144}$$

Αν εφαρμόσουμε και τις οριακές συνθήκες  $X(0) = X(L) = 0$  προκύπτει  $C = -A$  επομένως ο όρος των συνημίτονων φεύγει και επί πλέον τα ημίτονα θα μηδενίζονται για  $x = L$  οπότε καταλήγουμε στην:

$$X_n(x) = B \sin(k_n x), k_n = \frac{\pi n}{L}, B = \text{σταθ} \quad \text{Εξ. 145}$$

Επίσης από τις Εξ. 139, Εξ. 143 και Εξ. 145:

$$\lambda_n = -(v_s^2 k_n^2 + \kappa k_n^4) = -\left(v_s^2 \left(\frac{\pi n}{L}\right)^2 + \kappa \left(\frac{\pi n}{L}\right)^4\right) \quad \text{Εξ. 146}$$

Επομένως η χρονική εξίσωση Εξ. 138α με  $\lambda_n = \omega_n^2$  γίνεται:

$$T_n''(t) + aT_n'(t) + \omega_n^2 T_n(t) = 0 \quad \text{Εξ. 147}$$

Όπως και πριν αν θέσουμε  $T(t) = e^{st}$  και αντικαταστήσουμε θα προκύψει:

$$s^2 + as + \omega_n^2 = 0 \quad \text{Εξ. 148}$$

Και:

$$s_{1,2} = \frac{-a \pm \sqrt{a^2 - 4\omega_n^2}}{2} \quad \text{Εξ. 149}$$

Από εδώ θα διακρίνουμε τρεις περιπτώσεις:

i. Υποαπόσβεση (underdamping) αν  $a^2 < 4\omega_n^2$ . Τότε προκύπτει πως:

$$\left\{ \begin{array}{l} T_n(t) = e^{-\frac{at}{2}} (C_n \cos(\Omega_n t) + D_n \sin(\Omega_n t)) \\ \mu\epsilon \\ \Omega_n = \sqrt{\omega_n^2 - \frac{a^2}{4}} \end{array} \right\} \quad \text{Εξ. 150}$$

ii. Κρίσιμη απόσβεση (critical damping) αν  $a^2 = 4\omega_n^2$ , επομένως:

$$T_n(t) = (C_n + D_n t) e^{-\frac{at}{2}} \quad \text{Εξ. 151}$$

iii. Υπεραπόσβεση (overdamping) αν  $\alpha^2 > 4\omega_n^2$ , οπότε:

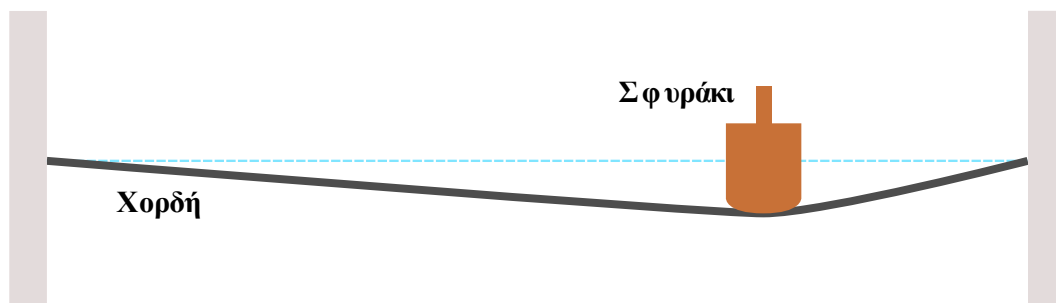
$$\left\{ \begin{array}{l} T_n(t) = C_n e^{s_1 t} + D_n e^{s_2 t} \\ \mu\epsilon \\ s_{1,2} = \frac{-\alpha \pm \sqrt{\alpha^2 - 4\omega_n^2}}{2} \end{array} \right\} \quad \text{Εξ. 152}$$

Στην περίπτωση μας, μας ενδιαφέρει η υποαπόσβεση αφού αυτή περιγράφει την ταλάντωση χορδής (η οποία φθίνει). Με αυτήν και το αποτέλεσμα της λύσης της χωρικής συνάρτησης καταλήγουμε στην γενική λύση της Εξ. 135 η οποία είναι η:

$$\left\{ \begin{array}{l} y(x, t) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right) e^{-\frac{at}{2}} (C_n \cos(\Omega_n t) + D_n \sin(\Omega_n t)) \\ \Omega_n = \sqrt{v_s^2 \left(\frac{\pi n}{L}\right)^2 + \kappa \left(\frac{\pi n}{L}\right)^4 - \frac{\alpha^2}{4}} \\ C_n, D_n \text{ σταθερές} \end{array} \right\} \quad \text{Εξ. 153}$$

### 7.2.3 Αρχικές συνθήκες

Όπως είπαμε παραπάνω όταν ο πιανίστας πιέσει ένα πλήκτρο στο πιάνο τότε ένας μηχανισμός αναγκάζει ένα σφυράκι να χτυπήσει την αντίστοιχη χορδή. Εκείνη τη στιγμή ακριβώς η χορδή είναι ακίνητη δηλαδή  $y_t(x, t) = 0$  και παίρνει ένα σχήμα  $y(x, 0) = f(x)$  το οποίο σε πρώτη προσέγγιση μοιάζει με δύο ευθείες που τέμνονται υπό γωνία κοντά στις  $180^\circ$  (Εικόνα 38).



Εικόνα 38: Χτύπημα χορδής πιάνου

Οι δύο παραπάνω εξισώσεις είναι αρκετές για να υπολογίσουμε τους συντελεστές  $C_n$  και  $D_n$  της Εξ. 153. Από ανάλυση Fourier προκύπτει πως:

$$C_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx \quad \text{Εξ. 154}$$

Και από την εξίσωση της ταχύτητας:

$$y_t(x, 0) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right) \left(-\frac{a}{2}C_n + D_n\Omega_n\right) = 0 \Rightarrow$$

$$D_n = \frac{aC_n}{2\Omega_n} \quad \text{Εξ. 155}$$

Επομένως μπορούμε να συνοψίσουμε:

$$\left\{ \begin{array}{l} y(x, t) = \sum_{n=1}^{\infty} C_n \sin\left(\frac{n\pi x}{L}\right) e^{-\frac{at}{2}} \left( \cos(\Omega_n t) + \frac{a}{2\Omega_n} \sin(\Omega_n t) \right), (\alpha) \\ C_n = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx, (\beta) \\ \Omega_n = \sqrt{v_s^2 \left(\frac{\pi n}{L}\right)^2 + \kappa \left(\frac{\pi n}{L}\right)^4 - \frac{\alpha^2}{4}}, (\gamma) \end{array} \right. \quad \text{Εξ. 156}$$

#### 7.2.4 Inharmonicity

Αν επιλύσουμε την κυματική εξίσωση σε ιδεατή χορδή για τις ιδιοσυχνότητες καταλήγουμε σε σχέση της μορφής:

$$f_n = nf_1 \quad \text{Εξ. 157}$$

Όλες οι συχνότητες που εμφανίζονται είναι ακέραια πολλαπλάσια της θεμελίου συχνότητας και τις καλούμε αρμονικές (harmonics). Σε πραγματικές όμως χορδές (π.χ. σε χορδές πιάνου ή κιθάρας) δεν συμβαίνει αυτό. Όπως είδαμε και παραπάνω (Εξ. 156) οι πραγματικές συχνότητες διαφέρουν από τον θεωρητικό υπολογισμό. Το φαινόμενο έχει μελετηθεί ήδη

από την δεκαετία του '60 και καλείται inharmonicity. Υπάρχει και ο ακόλουθος τύπος με τον οποίο μπορεί κανείς να υπολογίσει τις συχνότητες:

$$f_n = nf_0\sqrt{1 + Bn^2} \quad \text{Εξ. 158}$$

Όπου  $f_0$  η θεμέλια συχνότητα,  $f_n$  η ζητούμενη συχνότητα,  $n$  ο  $a/a$  της συχνότητας αυτής ( $n \geq 1$ ) και  $B$  συντελεστής ο οποίος εξαρτάται από χαρακτηριστικά της χορδής, όπως την τάση, το μήκος, τη διάμετρο, την δυσκαμψία κ.α. Όπως βλέπουμε και από την παραπάνω εξίσωση η πραγματική συχνότητα είναι ψηλότερη από το ακέραιο πολλαπλάσιο, πράγμα που συμφωνεί και με τους υπολογισμούς της προηγούμενης παραγράφου (Εξ. 156).

### 7.2.5 Διαμήκη κύματα

Στις χορδές του πιάνου δεν εμφανίζονται μόνο εγκάρσια κύματα αλλά και διαμήκη. Καθώς το σφυράκι χτυπάει μια χορδή αυτή επιμηκύνεται ελάχιστα το οποίο όμως είναι αρκετό για να δημιουργήσει τα διαμήκη κύματα. Τα κύματα αυτά έχουν υψηλότερες συχνότητες από ότι τα εγκάρσια τυπικά ανάμεσα στα 1250Hz – 50KHz και δίνουν χαρακτήρα ιδιαίτερα στις μπάσες χορδές. Και σε αυτή την περίπτωση ισχύει η κυματική εξίσωση (Εξ. 128). Στα τα διαμήκη κύματα όμως δεν εμφανίζεται inharmonicity και οι λύσεις της εξίσωσης κύματος σε αυτή την περίπτωση είναι ακεραία πολλαπλάσια μιας βασικής συχνότητας, δηλαδή ακολουθούν την Εξ. 157.

### 7.2.6 Αντηχείο (soundboard)

Το αντηχείο του πιάνου είναι μια ξύλινη πλάκα (Εικόνα 39) η οποία βρίσκεται κάτω από τις χορδές στα πιάνο με ουρά (ή πίσω στα κάθετα). Καθώς οι χορδές πάλλονται οι παλμοί μεταδίδονται μέσω της γέφυρας στο αντηχείο το οποίο πάλλεται με τη σειρά του. Το αντηχείο λόγω της μεγάλης επιφάνειάς του πιέζει πολύ περισσότερο αέρα και έτσι ο ήχος γενικά ενισχύεται. Μέσω του αντηχείου όμως αλλάζει και η χροιά του οργάνου καθώς εξ' αιτίας της κατασκευής του (όπως θα δούμε και παρακάτω) ενισχύει περισσότερο κάποιες συχνότητες ενώ άλλες όχι, αλλά καταστέλλει και συχνότητες. Για την μελέτη της απόκρισης ενός αντηχείου θα ξεκινήσουμε από την εξίσωση Kirchhoff – Love:

$$D\nabla^4 w(x, y, t) + \rho h \frac{\partial^2 w(x, y, t)}{\partial t^2} = F(x, y, t) \quad \text{Εξ. 159}$$

Όπου  $w(x, y, t)$  η κάθετη μετατόπιση της πλάκας,  $\rho$  η πυκνότητα του ξύλου,  $h$  το πάχος του,  $D = \frac{Eh^3}{12(1-\nu^2)}$  η δυσκαμψία,  $F(x, y, t)$  η δύναμη που ασκείται από τις χορδές. Εύκολα μπορεί κανείς να διαπιστώσει πως η παραπάνω εξίσωση μοιάζει αρκετά με την Εξ. 26 των Euler – Bernoulli για τις δοκούς αφού στην ουσία είναι η γενίκευσή της σε 2 διαστάσεις. Θα προσπαθήσουμε να λύσουμε την αντίστοιχη ομογενή εξίσωση, σε τετράπλευρη πλάκα.



Εικόνα 39: Αντηχείο πιάνου (soundboard)

$$\left\{ \begin{array}{l} D\nabla^4 w(x, y, t) + \rho h \frac{\partial^2 w(x, y, t)}{\partial t^2} = 0 \\ \text{όπου} \\ x \in [0, a], y \in [0, b] \end{array} \right\} \quad \text{Εξ. 160}$$

Έστω πως η λύση της είναι της μορφής:

$$\left\{ \begin{array}{l} w(x, y, t) = W(x, y)T(t), (\alpha) \\ \mu\epsilon \\ W(x, y) = X(x)Y(y), (\beta) \end{array} \right\} \quad \text{Εξ. 161}$$

Αν αντικαταστήσουμε προκύπτει:

$$\frac{T''(t)}{T(t)} = -\frac{D}{ph} \frac{\nabla^4 W(x, y)}{W(x, y)} = -\omega^2 \quad \text{Εξ. 162}$$

Από την παραπάνω προκύπτουν δυο εξισώσεις. Η χρονική:

$$T''(t) + \omega^2 T(t) = 0 \quad \text{Εξ. 163}$$

Και η χωρική:

$$\left\{ \begin{array}{l} \nabla^4 W(x, y) - \beta^4 W(x, y) = 0 \\ \mu\epsilon \\ \beta^4 = \frac{ph\omega^2}{D} \end{array} \right\} \quad \text{Εξ. 164}$$

Αν θεωρήσουμε ότι στα άκρα ( $x = 0, x = a, y = 0, y = b$ ) οι  $W$  και  $\nabla^2 W$  μηδενίζονται τότε για τις ιδιοσυναρτήσεις προκύπτει πως:

$$W_{mn}(x, y) = \sin\left(\frac{m\pi x}{a}\right) \sin\left(\frac{n\pi y}{b}\right), m, n \geq 1 \quad \text{Εξ. 165}$$

Και αν αντικαταστήσουμε:

$$\beta_{mn}^2 = \left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2 \Rightarrow$$

$$\omega_{mn} = \pi^2 \sqrt{\frac{D}{ph} \left[ \left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2 \right]} \Rightarrow$$

$$f_{mn} = \frac{\omega_{mn}}{2\pi} = \frac{\pi}{2} \sqrt{\frac{D}{ph} \left[ \left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2 \right]} \Rightarrow \quad \text{Εξ. 166}$$

Και η γενική λύση της Εξ. 160 θα είναι η:

$$w(x, y, t) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} A_{mn} \sin\left(\frac{m\pi x}{a}\right) \sin\left(\frac{n\pi y}{b}\right) \cos(\omega_{mn}t + \varphi_{mn}) \quad \text{Εξ. 167}$$

Φυσικά η Εξ. 159 δεν μπορεί να λυθεί αναλυτικά αφού το σχήμα του αντηχείου όπως μπορούμε να δούμε και στην Εικόνα 39 είναι αρκετά πολύπλοκο, αλλά και η πυκνότητά του δεν είναι σταθερή, επί πλέον πάνω του κολλάνε ξύλινες δοκούς (πλευρά) τα οποία πέρα από την υποστήριξη που προσφέρουν αλλάζουν την κατανομή των δονήσεων και τέλος αν και το αντηχείο είναι κολλημένο, τα άκρα του δεν είναι εντελώς σταθερά αλλά μπορούν να δονούνται ελαφρώς, δηλαδή το πρόβλημα δεν ανήκει στην περίπτωση των σταθερών άκρων αλλά ούτε και στον ελεύθερον παρά κάπου ενδιάμεσα. Στην πράξη η Εξ. 159 λύνεται με τη μέθοδο των πεπερασμένων στοιχείων (ΜΠΣ).

Εκείνο που κρατάμε από την παραπάνω ανάλυση είναι πως το αντηχείο στο πιάνο ενισχύει - φιλτράρει τις συχνότητες που παράγονται από τις δονούμενες χορδές. Πειραματικά έχει βρεθεί η απόκριση στις διάφορες συχνότητες. Τα αποτελέσματα παρουσιάζονται συνοπτικά στον παρακάτω πίνακα. Όπως βλέπουμε λειτουργεί λίγο – πολύ σαν φίλτρο ζώνης (Band Pass Filter).

Συχνοτικό εύρος	Αποτέλεσμα
20-80 Hz	Κατά βάση εξασθενούν
80-400 Hz	Ενισχύονται αρκετά
400-1500 Hz	Η περιοχή με την μεγαλύτερη ενίσχυση
1,5 – 5 KHz	Μερική εξασθένιση
5 – 8 KHz	Σχεδόν κόβονται

Πίνακας 8: Συχνοτική απόκριση αντηχείου

### 7.2.7 Συντονισμός Helmholtz

Το κουτί του πιάνου λειτουργεί ασθενώς και σαν αντηχείο Helmholtz (4.2.1 Συντονισμός Helmholtz). Η Εξ. 109 μας δίνει την συχνότητα Helmholtz. Στην περίπτωση του πιάνου αυτή συνήθως είναι ανάμεσα στα 70 – 120 Hz. Το φαινόμενο αυτό ενισχύει λίγο τις χαμηλές συχνότητες του πιάνου χωρίς να επηρεάζει πολύ τον ήχο του.

### 7.2.8 Το μοντέλο συνολικά

Με βάση όσα είπαμε σχεδιάζουμε το μοντέλο μας ως εξής:

- Αναλόγως της νότας που πατήθηκε επιλέγουμε αν θα χρειαστούμε μία, δύο ή τρεις χορδές. Αν χρειάζεται παραπάνω από μια εκτελούμε τον ίδιο κώδικα για ελαφρώς διαφορετική συχνότητα.
- Για την νότα που πατήθηκε υπολογίζουμε την θεμέλια συχνότητα. Με βάση αυτήν υπολογίζουμε τις ιδιοσυχνότητες λαμβάνοντας υπόψη την inharmonicity όπως είδαμε παραπάνω.
- Υπολογίζουμε τα αρχικά πλάτη για κάθε μια από τις συχνότητες αυτές.
- Θεωρώντας πως αυτές φθίνουν εκθετικά, υπολογίζουμε την χρονική σταθερά  $\tau_n$  για καθένα τους.
- Κάνουμε τους ίδιους υπολογισμούς (συχνότητες, πλάτη, σταθερές) για τα διαμήκη κύματα που εμφανίζονται στις χορδές.
- Αθροίζουμε για όλες τις συχνότητες.
- Για να μιμηθούμε το αντηχείο εφαρμόζουμε μερικά ζωνοπερατά φίλτρα στο τελικό σήμα.

## 7.3 Υλοποίηση στον υπολογιστή

Για την υλοποίηση του μοντέλου του πιάνου δημιουργήσαμε την κλάση `CVIPiano` η οποία κληρονόμησε από την `CVInstr`, την βασική κλάση των μοντέλων μας και εργαστήκαμε όπως ακριβώς περιγράψαμε στην προηγούμενη παράγραφο (7.2.8 Το μοντέλο συνολικά).

Ο πλήρης κώδικας του μοντέλου παρουσιάζεται στο παράρτημα Η (Παράρτημα Η: Ο κώδικας του πιάνου).

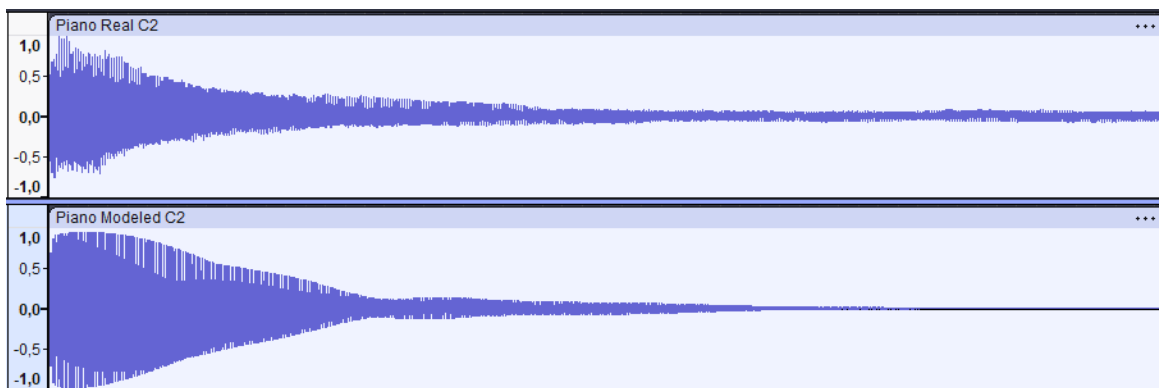
## 7.4 Αποτίμηση μοντέλου

Με τη χρήση του μοντέλου μας αναπαρήγαμε τον ήχο του πιάνου. Συγκεκριμένα δημιουργήσαμε αρκετά αρχεία με την νότα ντο σε διαφορετικά ύψη, δεδομένου ότι το πιάνο έχει μεγάλο εύρος και το ηχόχρωμα διαφέρει για κάθε περιοχή, τα οποία και συγκρίναμε με αντίστοιχα πραγματικού πιάνου. Τα αρχεία παρουσιάζονται στον παρακάτω πίνακα (Πίνακας 9: Τα συνοδευτικά αρχεία του πιάνου).

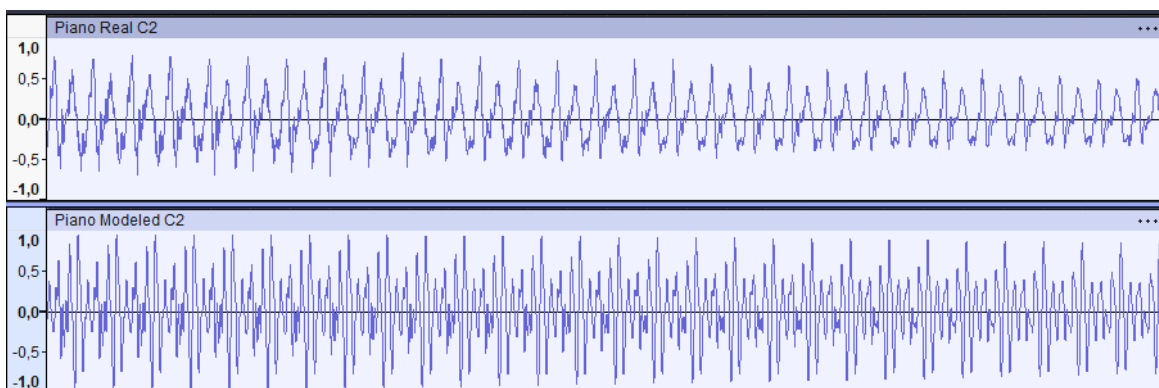
Όνομα αρχείου (πραγματικό όργανο)	Όνομα αρχείου (μοντελοποιημένο όργανο)	Περιγραφή
	Piano Modeled C1.mp3	Ντο1 (C1) 32,70 Hz
Piano Real C2.mp3	Piano Modeled C2.mp3	Ντο2 (C2) 65,41 Hz
Piano Real C4.mp3	Piano Modeled C4.mp3	Ντο4 (C4) 261,262 Hz
Piano Real C6.mp3	Piano Modeled C6.mp3	Ντο6 (C6) 1046,50 Hz
	Piano Modeled Scale C1 - C6. mp3	Ντο1 (C1) – Ντο6 (C1)
	Piano Modeled Minuett.mp3	Γνωστή μελωδία

**Πίνακας 9: Τα συνοδευτικά αρχεία του πιάνου**

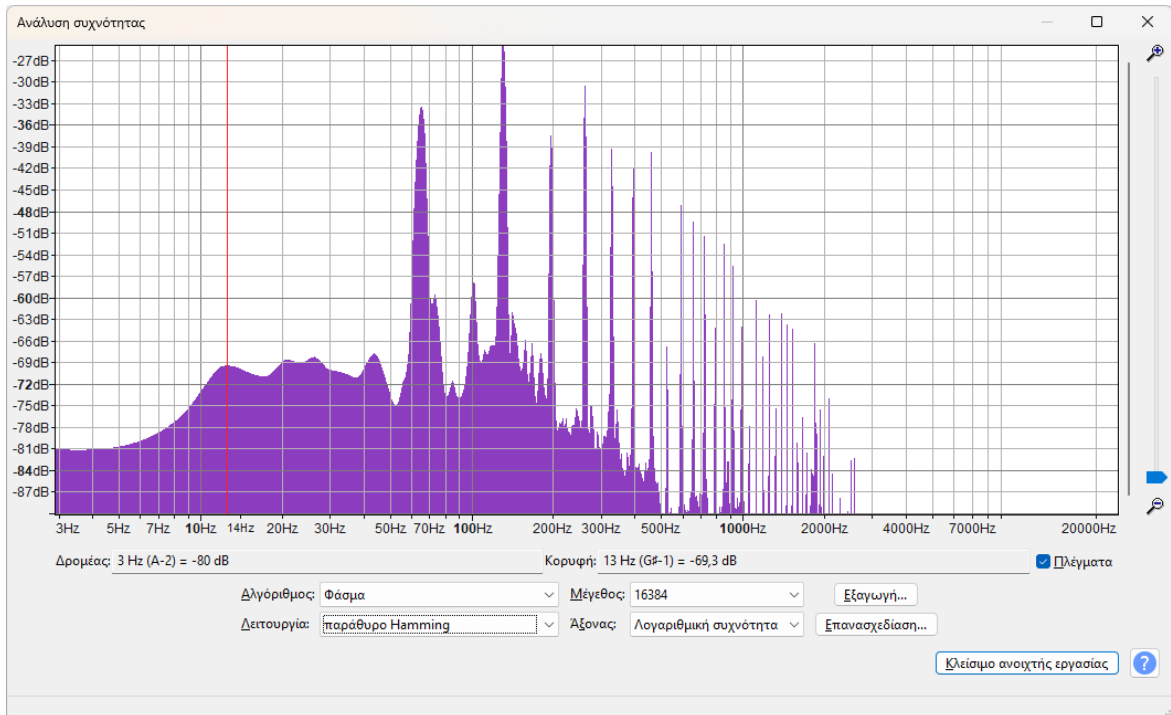
Πάντα με τη βοήθεια προγράμματος επεξεργασίας και εγγραφής ήχου συγκρίναμε τα αρχεία. Τα αποτελέσματα της σύγκρισης μας παρουσιάζονται παρακάτω.



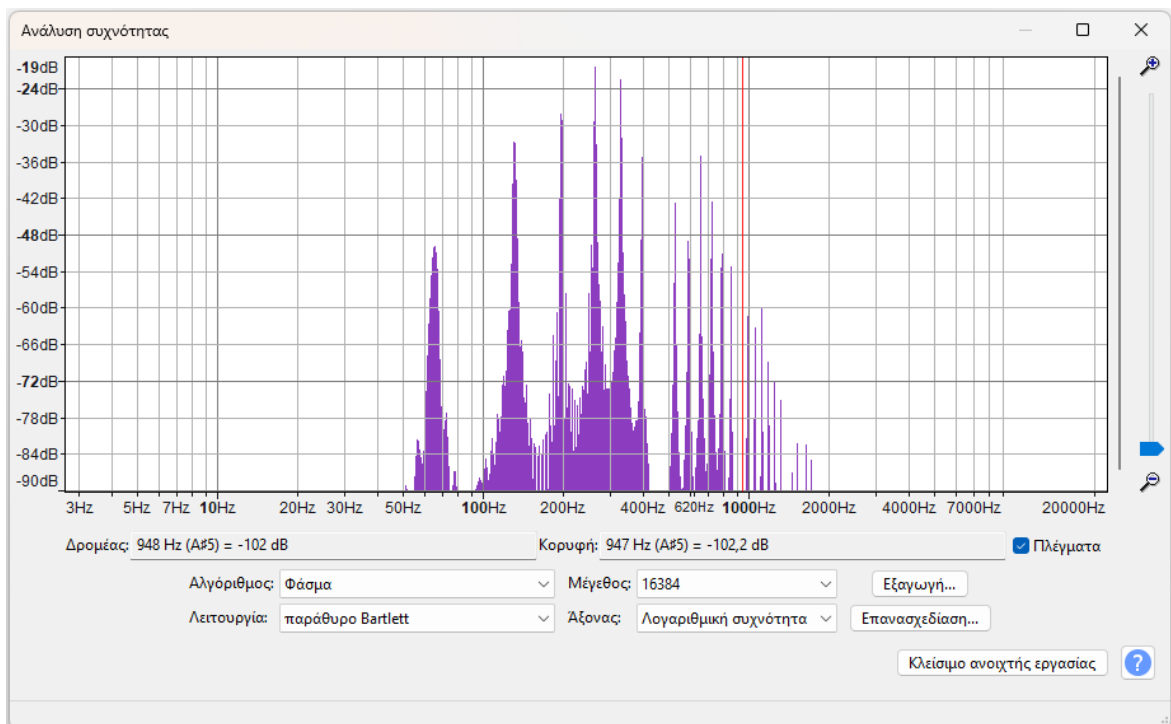
**Εικόνα 40: Γενικές κυματομορφές ντο2 πραγματικού / μοντελοποιημένου πιάνου**



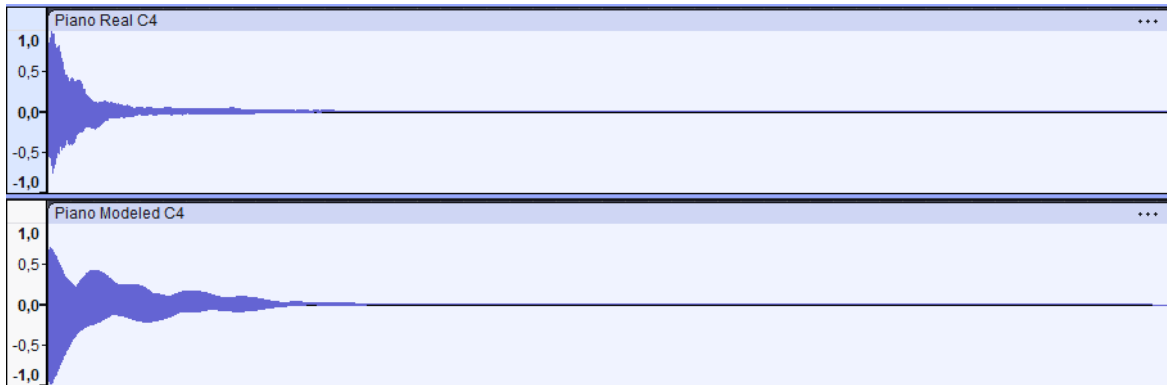
**Εικόνα 41: Λεπτομερείς κυματομορφές ντο2 πραγματικού / μοντελοποιημένου πιάνου**



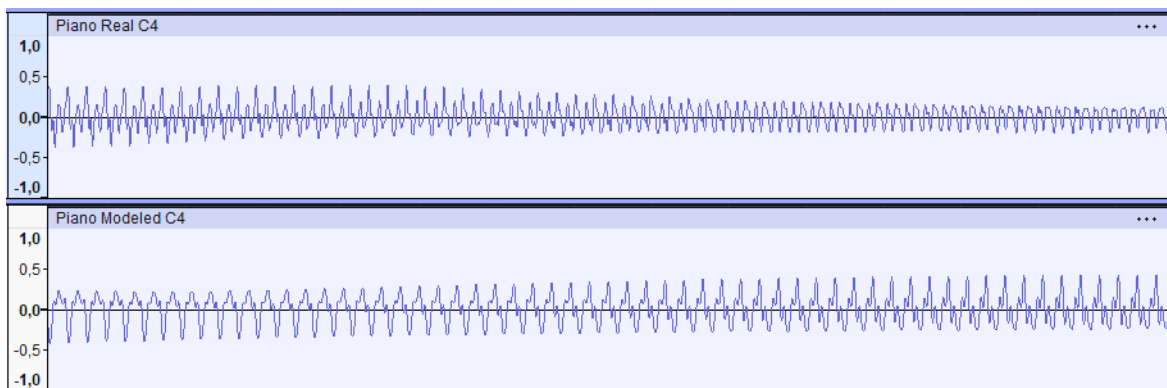
Εικόνα 42: Φάσμα ντο2 πραγματικού πιάνου



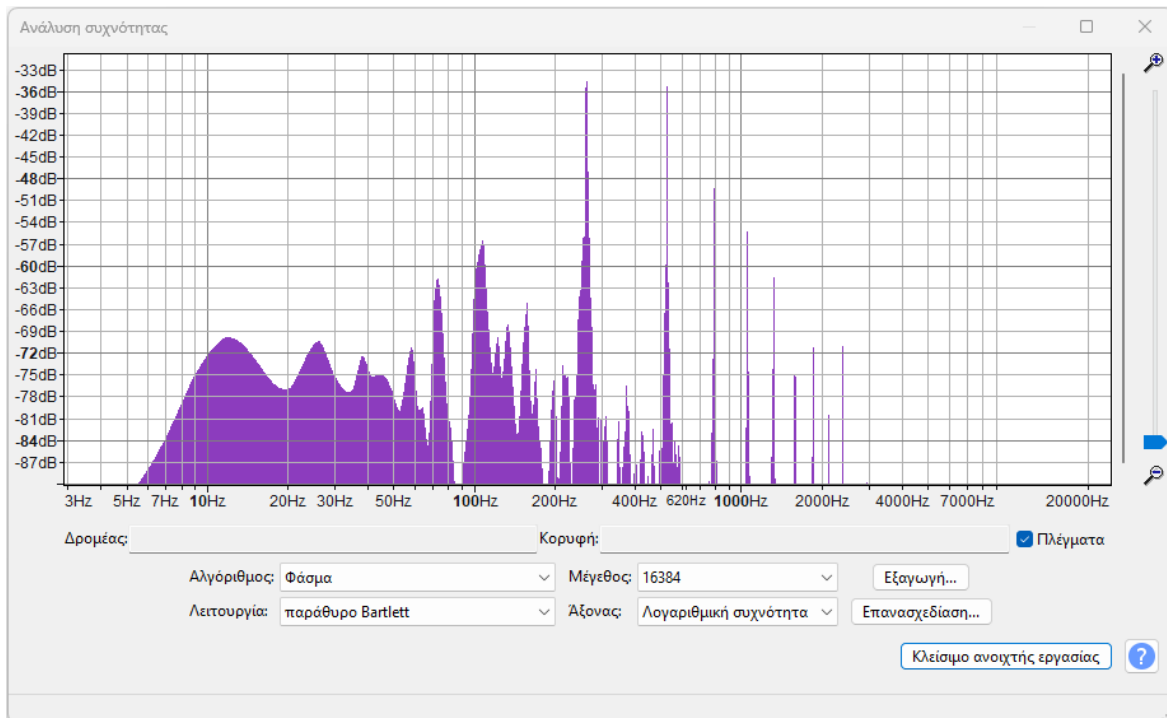
Εικόνα 43: Φάσμα ντο2 μοντελοποιημένου πιάνου



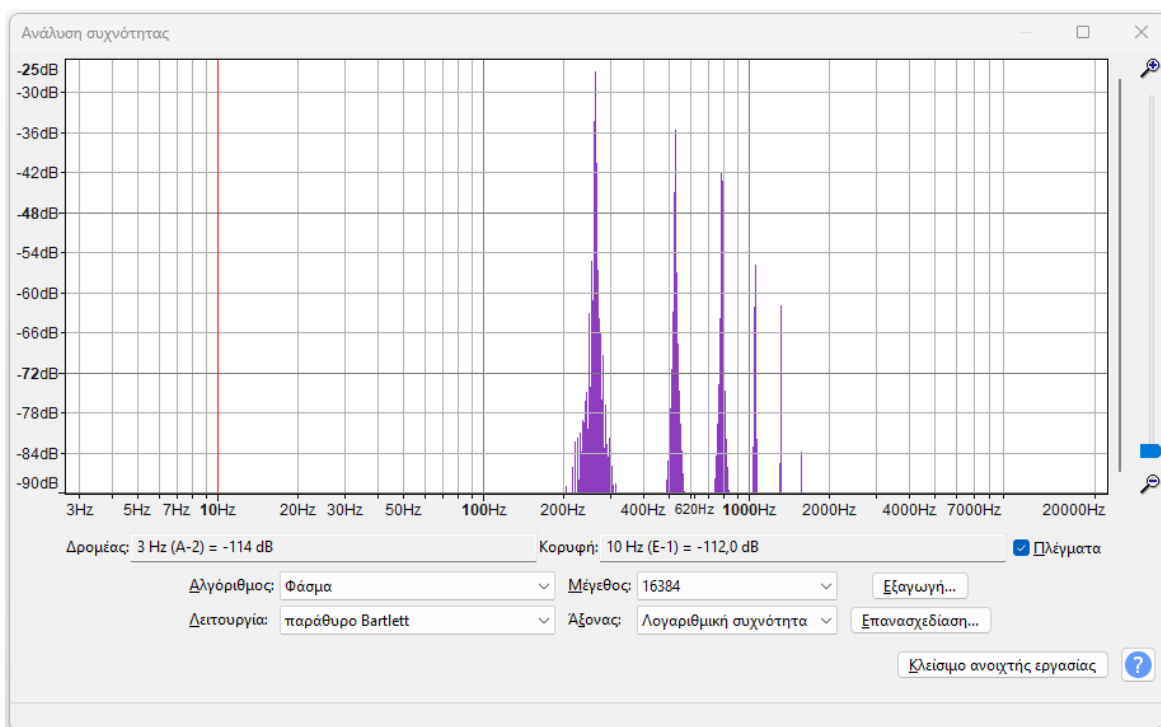
Εικόνα 44: Γενικές κυματομορφές ντο4 πραγματικού / μοντελοποιημένου πιάνου



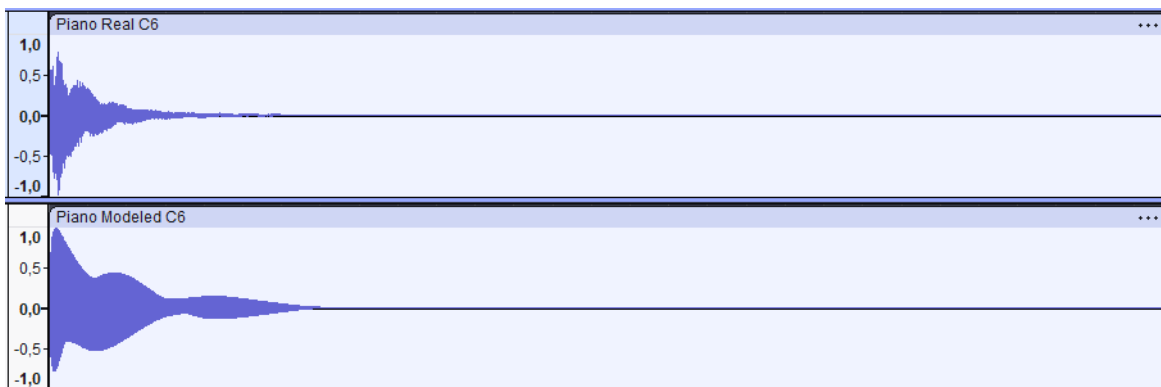
Εικόνα 45: Λεπτομερείς κυματομορφές ντο4 πραγματικού / μοντελοποιημένου πιάνου



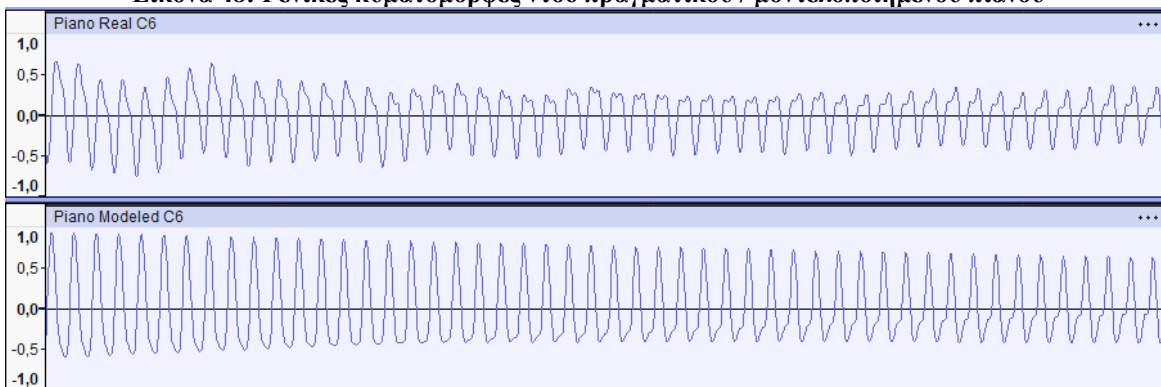
Εικόνα 46: Φάσμα ντο4 πραγματικού πιάνου



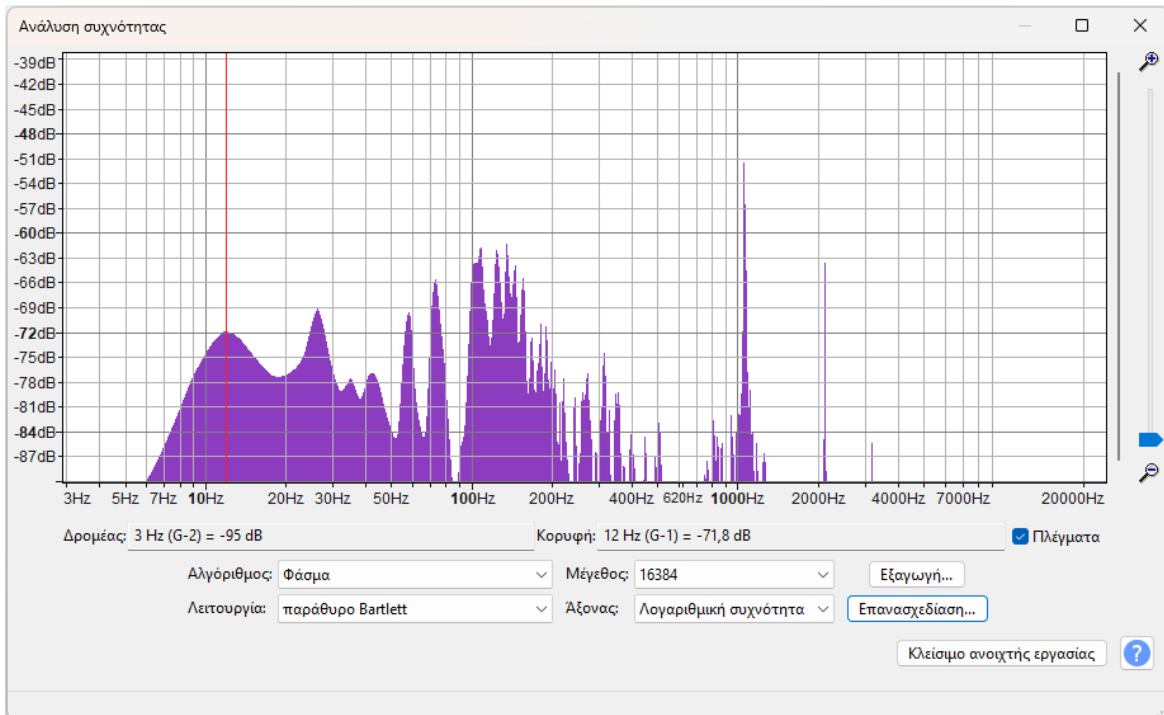
**Εικόνα 47: Φάσμα ντο4 μοντελοποιημένου πιάνου**



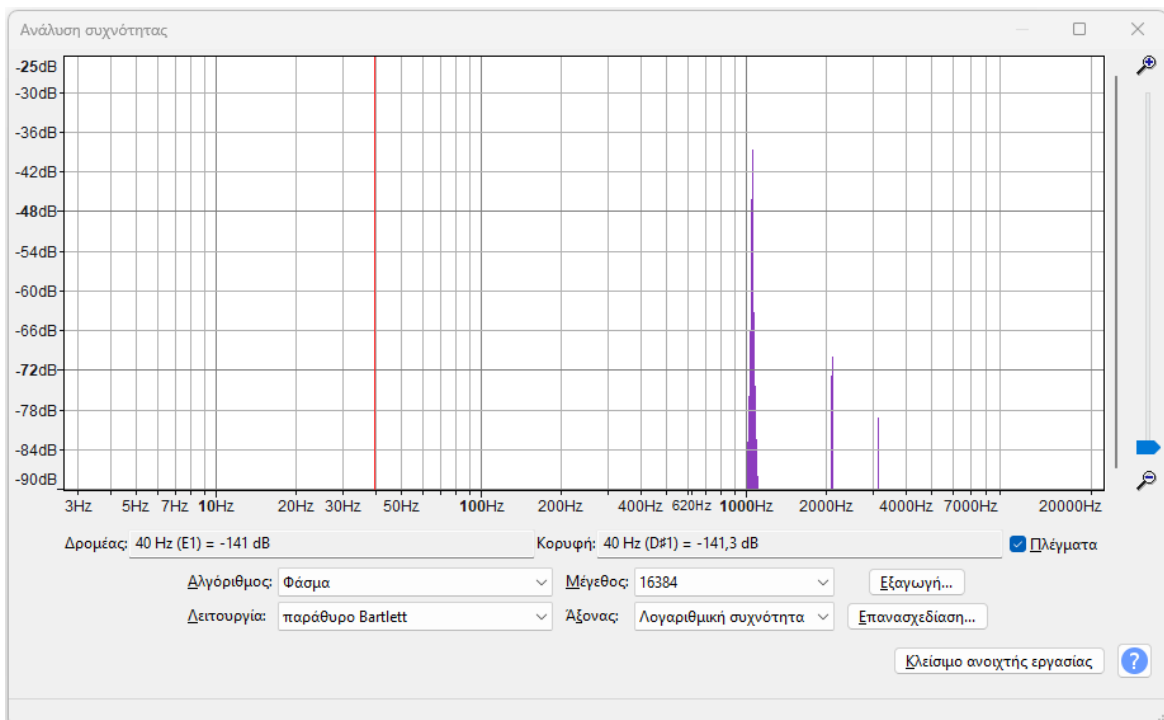
**Εικόνα 48: Γενικές κυματομορφές ντο6 πραγματικού / μοντελοποιημένου πιάνου**



**Εικόνα 49: Λεπτομερείς κυματομορφές ντο6 πραγματικού / μοντελοποιημένου πιάνου**



Εικόνα 50: Φάσμα ντο6 πραγματικού πιάνου



Εικόνα 51: Φάσμα ντο6 μοντελοποιημένου πιάνου

Ακούγοντας απλά τον ήχο του μοντέλου μας βλέπουμε πως δεν είναι αρκετά ρεαλιστικός, ιδιαίτερα στις ψηλές νότες όπου ακούγεται πολύ συνθετικός. Οι κυματομορφές παρουσιάζουν κάποιες ομοιότητες αλλά και αυτές με τη σειρά τους διαφέρουν αρκετά σε σχέση με αυτές του πραγματικού οργάνου. Τέλος αν προσέξουμε τα φάσματα θα διαπιστώσουμε πως οι συχνότητες που εμφανίζονται στο μοντελοποιημένο όργανο εμφανίζονται και στο πραγματικό αλλά δεν είναι οι μόνες. Στο πραγματικό όργανο εμφανίζονται πολλές περισσότερες συχνότητες ιδιαίτερα στην χαμηλή περιοχή του φάσματος. Επιγραμματικά το μοντελοποιημένο πιάνο απέχει αρκετά, όσον αφορά τον ήχο, από το πραγματικό. Ο λόγος που συμβαίνει αυτό είναι πως αν και τα φαινόμενα που μελετήσαμε περιγράφουν κάπως τη λειτουργία του πιάνου δεν είναι αρκετά για να δώσουν την πλήρη εικόνα. Εμείς ακολουθήσαμε μια στατική – γραμμική προσέγγιση, κάτι που δεν συμβαίνει στην πραγματικότητα όπου μιλάμε για ένα ασθενώς συζευγμένο, μη γραμμικό σύστημα. Έχουμε δηλαδή μοντελοποιήσει μερικά από τα βασικά χαρακτηριστικά του οργάνου, αλλά απέχουμε από την ακριβή μοντελοποίηση της λειτουργίας του.

## 8. Συμπεράσματα

Στην παρούσα εργασία προσπαθήσαμε να προσεγγίσουμε έστω και επιφανειακά την μοντελοποίηση κάποιων μουσικών οργάνων μελετώντας τα βασικά φυσικά φαινόμενα που διέπουν την λειτουργία τους. Πρόκειται για μια κάθε άλλο παρά απλή διαδικασία αφού ακόμα και για τα πιο απλά όργανα οι φυσικές διεργασίες που συμβαίνουν ώστε να παραχθεί ήχος είναι και πολλές και μάλλον αρκετά πολύπλοκες. Γενικά τα φαινόμενα που διέπουν τη λειτουργία των οργάνων είναι κατά βάση:

- Μη γραμμικά. Οι βασικές εξισώσεις λειτουργίας των οργάνων που κατασκευάσαμε σαν αθροίσματα ορισμένου αριθμού ημιτονικών όρων... μάλλον δεν ισχύουν.
- Συζευγμένα συστήματα. Δηλαδή η παλμική κίνηση μιας χορδής καθώς και του αντίστοιχου αντηχείου είναι ένα συζευγμένο σύστημα (το ένα επηρεάζει το άλλο) και πρέπει να τα προσεγγίσει κανείς σαν ένα ενιαίο σύστημα και όχι ξεχωριστά.
- Πολυδιάστατα. Έχουν δηλαδή μεγάλο αριθμό παραμέτρων που επηρεάζουν την λειτουργία τους (Π. χ. για να μοντελοποιήσει κανείς ακριβώς το χτύπημα μιας μπαγκέτας θα πρέπει να συνυπολογίσει τη μάζα της, την ταχύτητά της, το σχήμα της, το υλικό κατασκευής, το σημείο που χτυπάει πάνω στο όργανο, το υλικό κατασκευής του οργάνου...).
- Χαοτικά. Πολλά όργανα εμφανίζουν χαοτική συμπεριφορά, όπως για παράδειγμα τα χάλκινα πνευστά τα οποία μπορούν να δώσουν και εντελώς χαοτικές ταλαντώσεις.
- Έχουν φαινόμενα διαφορετικών χρονικών κλιμάκων. Π.χ. Κάποια φαινόμενα τα επηρεάζουν για σύντομο χρονικό διάστημα (π.χ. το χτύπημα της μπαγκέτας διαρκεί ms) ενώ κάποια άλλα για μεγάλα διαστήματα (π.χ. η ταλάντωση της μπάρας μπορεί να διαρκέσει λεπτά) τα οποία πρέπει να μελετηθούν χωριστά.
- Εμφανίζουν απώλειες. Δεν είναι δηλαδή τα ιδεατά – χωρίς απώλειες συστήματα που μελετώνται συνήθως.

Επί πλέον για την σωστή (ρεαλιστική) παραγωγή ήχου από ένα μοντέλο οργάνου, το όργανο πρέπει να μοντελοποιηθεί πλήρως, δεν γίνεται δηλαδή να φτάσουμε στον ρεαλισμό επιλέγοντας κάποια βασικά μόνο φαινόμενα (αν μοντελοποιηθεί το 90% των φυσικών φαινομένων ο ήχος δεν θα είναι 90% σωστός).

Τέλος εκτός από τα φαινόμενα καθαυτά για τους πραγματικούς υπολογισμούς χρειάζονται και ακριβή αριθμητικά δεδομένα. Για παράδειγμα χρειάζονται οι ακριβείς τιμές των σταθερών που έχουν να κάνουν με τα υλικά που χρησιμοποιεί το όργανο (π.χ. δυσκαμψία για το συγκεκριμένο υλικό). Αλλά και οι ίδιοι οι υπολογισμοί μπορεί να μην είναι και τόσο απλοί.

Αν και δύσκολη λοιπόν η μοντελοποίηση της λειτουργίας των μουσικών οργάνων, παρουσιάζει αρκετό ενδιαφέρον, όχι μόνο για την επίλυση του θεωρητικού προβλήματος αλλά και για την πρακτική εφαρμογή τους. Η χρήση του υπολογιστή στην μελέτη, διδασκαλία αλλά και παραγωγή της μουσικής στις μέρες μας θεωρείται αυτονόητη. Η φυσική μοντελοποίηση των οργάνων σήμερα, κερδίζει συνεχώς έδαφος σε σχέση πιο παραδοσιακές τεχνικές με χρήση ηχογραφημένου υλικού. Υπάρχει λοιπόν πεδίο εφαρμογής του προβλήματος.

## Βιβλιογραφία

- [1] Δ. Γ. Φούσκας, «Ψηφιακές επικοινωνίες I,» σε *Ψηφιακές επικοινωνίες I*, Ελληνικό Ανοικτό Πανεπιστήμιο, p. 226.
- [2] T. D. R. Neville H. Fletcher, «The Physics of Musical Instruments,» 1988, p. 756.
- [3] «Euphonics,» [Ηλεκτρονικό]. Available: <https://euphonics.org/homepage/>.
- [4] Έ. Αβέρωφ, Εισαγωγή στην οργανογνωσία, Αθήνα: Φίλιππος Νάκας, 1992.
- [5] Σ. Τραχανάς, Μερικές διαφορικές εξισώσεις, Ηράκλειο: Πανεπιστημιακές εκδόσεις Κρήτης, 2018.
- [6] Ν. Δ. Α. Γεώργιος Δ. Ακρίβης, Μερικές Διαφορικές Εξισώσεις, Αθήνα: Σύγχρονη Εκδοτική, 2017.
- [7] C. Roads, The computer music tutorial, The MIT Press, p. 1234.
- [8] H. E. W. a. D. H. White, Physics And Music - The science of musical sound, Dover books on physics, 2014, p. 432.
- [9] Δ. Αθανασιάδης, Ιστορία της μουσικής, Θεσσαλονίκη: Μακεδονικό ωδείο, 2000.

## Παράρτημα Α: Κλάση CWaveFile - αρχείο .wav

```
#pragma once

#include <vector>

class CWaveFile
{
public:
    struct WAVEFORMATEX
    {
        unsigned short wFormatTag; /* format type */
        unsigned short nChannels; /* number of channels (i.e. mono, stereo...) */
        unsigned long nSamplesPerSec; /* sample rate */
        unsigned long nAvgBytesPerSec; /* for buffer estimation */
        unsigned short nBlockAlign; /* block size of data */
        unsigned short wBitsPerSample; /* number of bits per sample of mono data */
        unsigned short cbSize; /* the count in bytes of the size of */
        /* extra information (after cbSize) */
    };

public:
    CWaveFile();

    virtual ~CWaveFile();

    static bool Save(const wchar_t * strFilePath, const WAVEFORMATEX &wfx, void
    *pData, int nDataSize);

protected:
    std::vector<unsigned char> m_arData;
};
```

### Κώδικας 1: Η δήλωση της κλάσης CWaveFile

```
#include "WaveFile.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#endif

#define MakeFourCC(ch0, ch1, ch2, ch3) ((unsigned long)(unsigned char)(ch0) | ((unsigned
long)(unsigned char)(ch1) << 8) | ((unsigned long)(unsigned char)(ch2) << 16) | ((unsigned
long)(unsigned char)(ch3) << 24 ))

CWaveFile::CWaveFile()
{
}

CWaveFile::~CWaveFile()
{
}

bool CWaveFile::Save(const wchar_t*strFilePath, const WAVEFORMATEX &wfx, void *pData, int
nDataSize)
{
    if (!strFilePath || !pData || nDataSize<=0)
        return false;

    struct WAVEHEADER
    {
        unsigned long dwRiff;
        unsigned long dwBytes;
        unsigned long dwWave;
        unsigned long dwFmt;
        int nSizeW;
        unsigned short wFormatTag;
        unsigned short nChannels;
    };
```

```
        unsigned long nSamplesPerSec;
        unsigned long nAvgBytesPerSec;
        unsigned short nBlockAlign;
        unsigned short wBitsPerSample;
        unsigned long dwData;
        unsigned long dwPcmBytes;
    };

    WAVEHEADER hdr;
    hdr.dwRiff = MakeFourCC('R','I','F','F');
    hdr.dwBytes = nDataSize+36;
    hdr.dwWave = MakeFourCC('W','A','V','E');
    hdr.dwFmt = MakeFourCC('f','m','t',' ');
    hdr.nSizeW = 16;
    hdr.wFormatTag = wfx.wFormatTag;
    hdr.nChannels = wfx.nChannels;
    hdr.nSamplesPerSec = wfx.nSamplesPerSec;
    hdr.nAvgBytesPerSec = wfx.nAvgBytesPerSec;
    hdr.nBlockAlign = wfx.nBlockAlign;
    hdr.wBitsPerSample = wfx.wBitsPerSample;
    hdr.dwData = MakeFourCC('d','a','t','a');
    hdr.dwPcmBytes = nDataSize;

    FILE* pFile = nullptr;

    errno_t e = _wfopen_s(&pFile, strFilePath, L"wb");

    if (!pFile)
        return false;

    size_t sz = fwrite(&hdr, 1, sizeof(hdr), pFile);

    if (sz != sizeof(hdr))
    {
        fclose(pFile);
        return false;
    }

    sz = fwrite(pData, 1, nDataSize, pFile);

    if (sz != nDataSize)
    {
        fclose(pFile);
        return false;
    }

    fclose(pFile);

    return true;
}
```

**Κώδικας 2: Υλοποίηση της κλάσης CWaveFile**

## Παράρτημα Β: Κλάσεις βάσης

```
#pragma once

#include "WaveFile.h"

class CVInstr
{
public:
    CVInstr();

    virtual ~CVInstr();

    bool SaveTestSound(const wchar_t* strFilePath, double freq, double dur);

    bool Save(const wchar_t* strFilePath, double dur, float* pData, int nSamples);

    virtual bool Play(std::vector < std::pair<int, double> >& piece, std::vector<float>
&arSamples);

    void SetGain(double gain);

    static double NoteFrequency(int nNote);

    inline int GetSampleRate() const
    {return m_nSampleRate;}

    inline int GetBitsPerSec() const
    {return m_nBps;}

    inline double GetMaxFrequency() const
    {return (double)m_nSampleRate / 2.; }

    inline double GetGain() const
    {return m_gain;}

    inline double Pi() const
    {return m_pi;}

protected:
    static double Clamp(double v, double v1, double v2);

protected:
    const static double m_pi;
    double m_gain;
    int m_nSampleRate;
    int m_nBps;
};
```

### Κώδικας 3: Δήλωση κλάσης CVInstr

```
#include "VInstr.h"
#include <vector>
#include <cassert>

const double CVInstr::m_pi = (double)3.1415926535897932384626433832795;

CVInstr::CVInstr()
{
    m_nSampleRate = 44100; // 44100 Hz
    m_nBps = 32; // 32 bits - float
    m_gain = 0.5;
}
```

```
CVInstr::~CVInstr()
{
}

bool CVInstr::SaveTestSound(const wchar_t* strFilePath, double freq, double dur)
{
    int nSamples = (int)ceil(dur * m_nSampleRate);

    std::vector<float> vcSamples;

    double pi2 = 2. * m_pi;
    double angle = 0.;
    double da = (pi2 * freq) / (double)m_nSampleRate;

    for (int i = 0; i < nSamples; ++i)
    {
        double v = sin(angle);

        angle = remainder(angle + da, pi2);

        vcSamples.push_back((float)v);
    }

    float* pData = (float*)vcSamples.data();

    return Save(strFilePath, dur, pData, nSamples);
}

bool CVInstr::Save(const wchar_t* strFilePath, double dur, float* pData, int nSamples)
{
    CWaveFile::WAVEFORMATEX wfx;
    wfx.cbSize = sizeof(wfx); // = 0
    wfx.nChannels = 1;
    wfx.nSamplesPerSec = m_nSampleRate;
    wfx.wBitsPerSample = m_nBps;
    wfx.wFormatTag = 0x0003; // WAVE_FORMAT_IEEE_FLOAT
    wfx.nBlockAlign = wfx.nChannels * wfx.wBitsPerSample / 8;
    wfx.nAvgBytesPerSec = wfx.nSamplesPerSec * wfx.nBlockAlign;

    int nByteSize = nSamples * sizeof(float);

    return CWaveFile::Save(strFilePath, wfx, (void*)pData, nByteSize);
}

double CVInstr::NoteFrequency(int nNote)
{
    return 440. * pow(2., ((double)nNote - 69.) / 12.);
}
```

```
}

double CVInstr::Clamp(double v, double v1, double v2)
{
    assert(v1 <= v2);

    if (v < v1)
        v = v1;

    if (v > v2)
        v = v2;

    return v;
}

void CVInstr::SetGain(double gain)
{
    m_gain = Clamp(gain, 0., 1.);
}

bool CVInstr::Play(std::vector < std::pair<int, double> >& piece, std::vector<float>&
arSamples)
{
    return false;
}

```

#### Κώδικας 4: Υλοποίηση κλάσης CVInstr

```
#pragma once

#include "VInstr.h"
#include <vector>

class CVIMallet : public CVInstr
{
public:
    CVIMallet();

    virtual ~CVIMallet();

    virtual bool Strike(int nNote, std::vector<float>& arSF);

    bool Play(std::vector < std::pair<int, double> >& piece, std::vector<float>&
arSamples) override;

    inline const double* BValues() const
    {return m_bn;}

    inline const int BCount() const
    {return m_nBs;}

protected:
    void CalcBs();

protected:
    static const int m_nBs = 64;
    double m_bn[m_nBs];
};

```

#### Κώδικας 5: Δήλωση κλάσης CVIMallet

```
#include "VIMallet.h"
#include <cassert>

CVIMallet::CVIMallet()
{
    CalcBs();
}

CVIMallet::~CVIMallet()
{
}

bool CVIMallet::Strike(int nNote, std::vector<float>& arSF)
{
    return false;
}

bool CVIMallet::Play(std::vector < std::pair<int, double> >& piece, std::vector<float>&
arSamples)
{
    for (int i = 0; i < (int)piece.size(); ++i)
    {
        std::vector<float> arSF;

        Strike(piece[i].first, arSF);

        int nStart = (int)(piece[i].second * GetSampleRate());
        int nEnd = nStart + (int)arSF.size();

        if (nStart > arSamples.size())
        {
            for (int j = (int)arSamples.size(); j < nStart; ++j)
                arSamples.push_back(0);
        }

        for (int j = nStart; j < nEnd; ++j)
        {
            if (j < arSamples.size())
                arSamples[j] += arSF[j - nStart];
            else
                arSamples.push_back(arSF[j - nStart]);
        }
    }

    // Μισό δευτερόλεπτο σιωπής...
    for (int i = 0; i < GetSampleRate() / 2; ++i)
        arSamples.push_back(0);

    return true;
}

void CVIMallet::CalcBs()
{
    const double errf = 0.0000001;
    const double errx = 0.0000001;
    const double interval = Pi(); // Pi() / 2;
    const double offset = Pi(); // 0; // Pi() / 4;
    const int nMaxSteps = 256;
    const int nMaxI = BCount() * 16;

    memset(m_bn, 0, sizeof(m_bn));

    int nRoot = 0;

    for (int i = 0; i < nMaxI; ++i)
    {
        double x1 = offset + i * interval;
        double x2 = x1 + interval;

        double fx1 = cosh(x1) * cos(x1) - 1;
        double fx2 = cosh(x2) * cos(x2) - 1;

        if (abs(fx1) < errf)
```

```
{
    m_bn[nRoot] = x1;

    ++nRoot;

    continue;
}
else if (abs(fx2) < errf)
{
    m_bn[nRoot] = x2;

    ++nRoot;

    continue;
}

if (fx1 * fx2 < 0)
{
    int nSteps = 0;

    while (true)
    {
        double dx = abs(x1 - x2);

        double xm = (x1 + x2) / 2.;

        double fxm = cosh(xm) * cos(xm) - 1;

        ++nSteps;

        if (dx < errx || nSteps >= nMaxSteps)
        {
            m_bn[nRoot] = xm;

            ++nRoot;

            break;
        }

        if (fxm * fx1 < 0)
            x2 = xm;
        else
            x1 = xm;

        fx1 = cosh(x1) * cos(x1) - 1;
        fx2 = cosh(x2) * cos(x2) - 1;
    }

    if (nRoot >= m_nBs)
        break;
}
}
```

### **Κώδικας 6: Υλοποίηση κλάσης CVIMallet**

## Παράρτημα Γ: Ο κώδικας μεταλλόφωνου

```
#pragma once

#include "VIMallet.h"

class CVIGlockenspiel : public CVIMallet
{
public:
    CVIGlockenspiel();

    virtual ~CVIGlockenspiel();

    bool Strike(int nNote, std::vector<float>& arSF) override;

protected:
    void GeneratePartFreqs();
    int CalcPartials(double f1, double* freqs) const;
    void CalcAmplitudes(int nPartials, const double* freqs, double* An0) const;

protected:
    double m_partFreq[m_nBs];
    double m_Q1;
    double m_a;
    double m_k;
    double m_omg;
};
```

### Κώδικας 7: Δήλωση κλάσης CVIGlockenspiel

```
#include "VIGlockenspiel.h"
#include <cassert>

CVIGlockenspiel::CVIGlockenspiel()
{
    GeneratePartFreqs();

    m_Q1 = 1000;
    m_a = 0.7;
    m_k = 0.5;
}

CVIGlockenspiel::~~CVIGlockenspiel()
{
}

bool CVIGlockenspiel::Strike(int nNote, std::vector<float>& arSF)
{
    const double dt = 1 / (double)GetSampleRate();
    double t = 0.;
    double f1 = NoteFrequency(nNote);

    // Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
    double freqs[m_nBs];
    memset(freqs, 0, sizeof(freqs));
    int nPartials = CalcPartials(f1, freqs);

    // Υπολογισμός πλατιών
    double An0[m_nBs];
    memset(An0, 0, sizeof(An0));
    CalcAmplitudes(nPartials, freqs, An0);

    // Υπολογισμός συντελεστών ποιότητας
    double Qn[m_nBs];
    for (int i = 0; i < nPartials; ++i)
        Qn[i] = m_Q1 * pow(freqs[i] / f1, m_k / 5);

    // Υπολογισμός συντελεστή απωλειών
    double dn[m_nBs];
    for (int i = 0; i < nPartials; ++i)
```

```

        dn[i] = m_pi* freqs[i] / (2. * Qn[i]);

// Υπολογισμός χρονικής διάρκειας
const double thres = 0.0005;
double t_end = 0.;

for (int i = 0; i < nPartials; ++i)
    t_end = std::max(-log(thres / abs(An0[i])) / dn[i], t_end);

// Γωνία φ
double phi[m_nBs];
for (int i = 0; i < nPartials; ++i)
    phi[i] = -Pi() / 2.;

// Χρόνος διέγερσης
double t0 = (0.1 + 7.9 * (1 - m_k)) / 1000;

// Εφαρμογή τύπου
std::vector<double> arSD;

while (t <= t_end)
{
    double y = 0;

    for (int j = 0; j < nPartials; ++j)
        y += An0[j] * exp(-dn[j] * t) * cos(2 * m_pi * freqs[j] * t +
phi[j]);

    if (t < t0)
        y = y * t / t0;

    arSD.push_back(y);

    t += dt;
}

// Κανονικοποίηση & εφαρμογή έντασης
double mx = 0.;
for (int i = 0; i < arSD.size(); ++i)
    mx = std::max(mx, abs(arSD[i]));

for (int i = 0; i < arSD.size(); ++i)
    arSF.push_back((float)(GetGain() * arSD[i] / mx));

return true;
}

void CVIGlockenspiel::GeneratePartFreqs()
{
    for (int i = 0; i < BCount(); ++i)
        m_partFreq[i] = pow(BValues()[i] / BValues()[0], 2.);
}

// Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
int CVIGlockenspiel::CalcPartials(double f1, double* freqs) const
{
    assert(freqs);

    int nPartials = 0;

    for (int i = 0; i < BCount(); ++i)
    {
        double fi = f1 * m_partFreq[i];

        if (fi > GetMaxFrequency())
        {
            nPartials = i - 1;

            break;
        }

        freqs[i] = fi;
    }
}

```

```
    }

    if (nPartials == 0)
        nPartials = BCount() - 1;

    return nPartials;
}

// Υπολογισμός πλατών
void CVIGlockenspiel::CalcAmplitudes(int nPartials, const double* freqs, double* An0)
const
{
    assert(freqs && An0);

    // Υπολογισμός

    double pi2 = Pi() * Pi();
    double omg = 2. * Pi() * (600 + 600 * m_k);

    for (int i = 0; i < nPartials; ++i)
        An0[i] = pow(abs(2. * Pi() * freqs[i] - omg), m_a) * cos(pi2 * freqs[i] /
omg) / (1 - pow(4*pi2*freqs[i]/ omg,2));

    double AnMax = abs(An0[0]);

    for (int i = 1; i < nPartials; ++i)
    {
        double av = abs(An0[i]);

        if (AnMax < av)
            AnMax = av;
    }

    for (int i = 0; i < nPartials; ++i)
        An0[i] /= AnMax;
}
```

### Κώδικας 8: Υλοποίηση κλάσης CVIGlockenspiel

## Παράρτημα Δ: Ο κώδικας του ξυλόφωνου

```
#pragma once

#include "VIGlockenspiel.h"

class CVIXylophone : public CVIGlockenspiel
{
public:
    CVIXylophone();

    virtual ~CVIXylophone();

    bool Strike(int nNote, std::vector<float>& arSF) override;

protected:
    void RetuneFreqs();

    void CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
    override;

protected:
    double m_qf;
};
```

### Κώδικας 9: Η δήλωση της κλάσης CVIXylophone

```
#include "VIXylophone.h"
#include <cassert>

CVIXylophone::CVIXylophone()
{
    RetuneFreqs();

    m_Q1 = 200;
    m_a = 0.9;
    m_k = 0.01;
    m_qf = -0.2;
}

CVIXylophone::~CVIXylophone()
{
}

bool CVIXylophone::Strike(int nNote, std::vector<float>& arSF)
{
    const double dt = 1 / (double)GetSampleRate();
```

```

double t = 0.;
double f1 = NoteFrequency(nNote);

// Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
double freqs[m_nBs];
memset(freqs, 0, sizeof(freqs));
int nPartials = CalcPartials(f1, freqs);

// Υπολογισμός πλατών
double An0[m_nBs];
memset(An0, 0, sizeof(An0));
CalcAmplitudes(nPartials, freqs, An0);

// Υπολογισμός συντελεστών ποιότητας
double Qn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    Qn[i] = m_Q1 * pow(freqs[i] / f1, m_qf);

// Υπολογισμός συντελεστή απωλειών
double dn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    dn[i] = m_pi * freqs[i] / (1. * Qn[i]);

// Υπολογισμός χρονικής διάρκειας
const double thres = 0.0005;
double t_end = 0.;

for (int i = 0; i < nPartials; ++i)
    t_end = std::max(-log(thres / abs(An0[i])) / dn[i], t_end);

// Γωνία φ
double phi[m_nBs];
for (int i = 0; i < nPartials; ++i)
    phi[i] = -Pi() / 2.;

// Χρόνος διεγερσης
double t0 = 0.002; // (0.1 + 7.9 * (1 - m_k)) / 1000;

// Εφαρμογή τύπου
std::vector<double> arSD;

while (t <= t_end)
{
    double y = 0;

    for (int j = 0; j < nPartials; ++j)
        y += An0[j] * exp(-dn[j] * t) * cos(2 * m_pi * freqs[j] * t +
phi[j]);
}

```

```
        if (t < t0)
            y = y * t / t0;

        arSD.push_back(y);

        t += dt;
    }

    // Κανονικοποίηση & εφαρμογή έντασης

    double mx = 0.;
    for (int i = 0; i < arSD.size(); ++i)
        mx = std::max(mx, abs(arSD[i]));

    for (int i = 0; i < arSD.size(); ++i)
        arSF.push_back((float)(GetGain() * arSD[i] / mx));

    return true;
}

void CVIXylophone::RetuneFreqs()
{
    //int nFreqs = BCount();
    //for (int i = 0; i < nFreqs; ++i)
    //{
    //    int nFr = (int)(m_partFreq[i] + 0.5);

    //    m_partFreq[i] = (double)nFr;
    //}

    m_partFreq[1] = 3;
    m_partFreq[2] = 6;
    m_partFreq[3] = 10;
}

// Υπολογισμός πλατιών
void CVIXylophone::CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
{
    assert(freqs && An0);

    // Υπολογισμός
    const double p = -0.9;
    const double m = 0.21;

    for (int i = 0; i < nPartials; ++i)
    {
        double n = i + 1;
```

```
        An0[i] = abs(sin(n * Pi() * m)) * pow(n, p);
    }

    double AnMax = abs(An0[0]);

    for (int i = 1; i < nPartials; ++i)
    {
        double av = abs(An0[i]);

        if (AnMax < av)
            AnMax = av;
    }

    for (int i = 0; i < nPartials; ++i)
        An0[i] /= AnMax;
}
```

**Κώδικας 10: Υλοποίηση της κλάσης CVIXylophone**

## Παράρτημα Ε: Ο κώδικας της μαρίμπας

```
#pragma once

#include "VIGlockenspiel.h"

class CVIMarimba : public CVIGlockenspiel
{
public:
    CVIMarimba();

    virtual ~CVIMarimba();

    bool Strike(int nNote, std::vector<float>& arSF) override;

protected:
    void RetuneFreqs();

    void CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
    override;

protected:
    double m_qMS;
    double m_qME;
    double m_helmFac;
};
```

### Κώδικας 11: Δήλωση της κλάσης CVIMarimba

```
#include "VIMarimba.h"
#include <cassert>

CVIMarimba::CVIMarimba()
{
    RetuneFreqs();

    m_Q1 = 80;

    m_qMS = 70.;
    m_qME = 15.;
    m_helmFac = 2.;
}

CVIMarimba::~CVIMarimba()
{
}

bool CVIMarimba::Strike(int nNote, std::vector<float>& arSF)
{
    const double dt = 1 / (double)GetSampleRate();
    double t = 0.;
    double f1 = NoteFrequency(nNote);

    // Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
    double freqs[m_nBs];
    memset(freqs, 0, sizeof(freqs));
    int nPartials = CalcPartials(f1, freqs);

    // Υπολογισμός Qmar
    double umar = (nNote - 45.) / (96. - 45.);
    umar = 1. - (umar < 0 ? 0. : (umar > 1. ? 1 : umar));
    double Qmar = m_qME + umar * (m_qMS - m_qME);
    double dMar = -2 * m_pi * freqs[0] / (2 * Pi() * Qmar);

    // Υπολογισμός πλατών
    double An0[m_nBs];
    memset(An0, 0, sizeof(An0));
    CalcAmplitudes(nPartials, freqs, An0);
}
```

```

// Υπολογισμός συντελεστών ποιότητας
double Qn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    Qn[i] = m_Q1 * pow(freqs[i] / f1, 0.19); //0.275

// Υπολογισμός συντελεστή απωλειών
double dn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    dn[i] = m_pi * freqs[i] / (2.4 * Qn[i]);

// Υπολογισμός χρονικής διάρκειας
const double thres = 0.0005;
double t_end = 0.;

for (int i = 0; i < nPartials; ++i)
    t_end = std::max(-log(thres / abs(An0[i])) / dn[i], t_end);

// Γωνία φ
double phi[m_nBs];
for (int i = 0; i < nPartials; ++i)
    phi[i] = -Pi() / 2.;

// Χρόνος διεγερσης
double t0 = 0.002; // (0.1 + 7.9 * (1 - m_k)) / 1000;

// Εφαρμογή τύπου
std::vector<double> arSD;

while (t <= t_end)
{
    double y = 0;

    for (int j = 0; j < nPartials; ++j)
        y += An0[j] * exp(-dn[j] * t) * cos(2 * m_pi * freqs[j] * t +
phi[j]);

    y += m_helmFac * exp(dMar * t) * cos(2 * m_pi * freqs[0] * t + phi[0]);

    if (t < t0)
        y = y * t / t0;

    arSD.push_back(y);

    t += dt;
}

// Κανονικοποίηση & εφαρμογή έντασης
double mx = 0.;
for (int i = 0; i < arSD.size(); ++i)
    mx = std::max(mx, abs(arSD[i]));

for (int i = 0; i < arSD.size(); ++i)
    arSF.push_back((float)(GetGain() * arSD[i] / mx));

return true;
}

void CVIMarimba::RetuneFreqs()
{
    //int nFreqs = BCount();
    //for (int i = 0; i < nFreqs; ++i)
    //{
    //    int nFr = (int)(m_partFreq[i] + 0.5);

    //    m_partFreq[i] = (double)nFr;
    //}

    m_partFreq[1] = 4;
    //m_partFreq[2] = 10;
    //m_partFreq[3] = 10;
}

```

```
void CVIMarimba::CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
{
    assert(freqs && An0);

    // Υπολογισμός
    const double p = -.5;// -0.9;
    const double m = 0.21;

    for (int i = 0; i < nPartials; ++i)
    {
        double n = i + 1;

        An0[i] = abs(sin(n * Pi() * m)) * pow(n, p);
    }

    double AnMax = abs(An0[0]);

    for (int i = 1; i < nPartials; ++i)
    {
        double av = abs(An0[i]);

        if (AnMax < av)
            AnMax = av;
    }

    for (int i = 0; i < nPartials; ++i)
        An0[i] /= AnMax;
}
```

**Κώδικας 12: Υλοποίηση της κλάσης CVIMarimba**

## Παράρτημα ΣΤ: Ο κώδικας του βιμπράφωνου

```
#pragma once

#include "VIGlockenspiel.h"

class CVIVibraphone : public CVIGlockenspiel
{
public:
    CVIVibraphone();

    virtual ~CVIVibraphone();

    bool Strike(int nNote, std::vector<float>& arSF) override;

protected:
    void RetuneFreqs();

    void CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
    override;

protected:
    double m_vibRate;
    double m_vibStep;
    double m_qMS;
    double m_qME;
    double m_helmFac;
};
```

### Κώδικας 13: Δήλωση της κλάσης CVIVibraphone

```
#include "VIVibraphone.h"
#include <cassert>

CVIVibraphone::CVIVibraphone()
{
    RetuneFreqs();

    m_Q1 = 500;
    m_a = 0.9;
    m_k = 0.01;

    m_vibRate = 1.53;
    m_vibStep = 0.25;

    m_helmFac = 1.;
    m_qMS = 70.;
    m_qME = 15.;
}

CVIVibraphone::~CVIVibraphone()
{
}

bool CVIVibraphone::Strike(int nNote, std::vector<float>& arSF)
{
    const double dt = 1 / (double)GetSampleRate();
    double t = 0.;
    double f1 = NoteFrequency(nNote);

    const double vibra = 2. * Pi() * m_vibRate;

    // Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
    double freqs[m_nBs];
    memset(freqs, 0, sizeof(freqs));
    int nPartials = CalcPartials(f1, freqs);

    // Υπολογισμός Qmar
    double umar = (nNote - 45.) / (96. - 45.);
```

```

umar = 1. - (umar < 0 ? 0. : (umar > 1. ? 1 : umar));
double Qmar = m_qME + umar * (m_qMS - m_qME);
double dMar = -2 * m_pi * freqs[0] / (Qmar);

// Υπολογισμός πλατών
double An0[m_nBs];
memset(An0, 0, sizeof(An0));
CalcAmplitudes(nPartials, freqs, An0);

// Υπολογισμός συντελεστών ποιότητας
double Qn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    Qn[i] = m_Q1 * pow(freqs[i] / f1, 0.32);

// Υπολογισμός συντελεστή απωλειών
double dn[m_nBs];
for (int i = 0; i < nPartials; ++i)
    dn[i] = m_pi * freqs[i] / (2. * Qn[i]);

// Υπολογισμός χρονικής διάρκειας
const double thres = 0.0005;
double t_end = 0.;

for (int i = 0; i < nPartials; ++i)
    t_end = std::max(-log(thres / abs(An0[i])) / dn[i], t_end);

// Γωνία φ
double phi[m_nBs];
for (int i = 0; i < nPartials; ++i)
    phi[i] = -Pi() / 2.;

// Χρόνος διέγερσης
double t0 = 0.002; // (0.1 + 7.9 * (1 - m_k)) / 1000;

// Εφαρμογή τύπου
std::vector<double> arSD;

while (t <= t_end)
{
    double y = 0;

    for (int j = 0; j < nPartials; ++j)
        y += An0[j] * exp(-dn[j] * t) * cos(2 * m_pi * freqs[j] * t +
phi[j]);

    y += m_helmFac * exp(dMar * t) * cos(2 * m_pi * freqs[0] * t + phi[0]);

    if (t < t0)
        y = y * t / t0;

    // vibration
    //if (t > 0.1)
    {
        double v = abs(sin(viba * t));
        v = v < m_vibStep ? m_vibStep : (v > (1 - m_vibStep) ? (1 -
m_vibStep) : v);
        y *= v;
    }

    arSD.push_back(y);

    t += dt;
}

// Κανονικοποίηση & εφαρμογή έντασης
double mx = 0.;
for (int i = 0; i < arSD.size(); ++i)
    mx = std::max(mx, abs(arSD[i]));

for (int i = 0; i < arSD.size(); ++i)
    arSF.push_back((float) (GetGain() * arSD[i] / mx));

```

```
        return true;
    }

void CVIVibraphone::RetuneFreqs()
{
    int nFreqs = BCount();
    for (int i = 1; i < nFreqs; ++i)
    {
        int nFr = (int)(m_partFreq[i] + 1.);

        m_partFreq[i] = (double)nFr;
    }
}

// Υπολογισμός πλατών
void CVIVibraphone::CalcAmplitudes(int nPartials, const double* freqs, double* An0)
const
{
    assert(freqs && An0);

    // Υπολογισμός
    const double p = -1.7;
    const double m = 0.21;

    for (int i = 0; i < nPartials; ++i)
    {
        double n = i + 1;

        An0[i] = abs(sin(n * Pi() * m)) * pow(n, p);
    }

    double AnMax = abs(An0[0]);

    for (int i = 1; i < nPartials; ++i)
    {
        double av = abs(An0[i]);

        if (AnMax < av)
            AnMax = av;
    }

    for (int i = 0; i < nPartials; ++i)
        An0[i] /= AnMax;
}
}
```

#### Κώδικας 14: Υλοποίηση της κλάσης CVIVibraphone

## Παράρτημα Z: Ο κώδικας του γούντμπλοκ

```
#pragma once

#include "VIMallet.h"

class CVIWoodblock : public CVIMallet
{
public:
    CVIWoodblock();

    virtual ~CVIWoodblock();

    bool Strike(int nNote, std::vector<float>& arSF) override;

protected:
    void GeneratePartFreqs();
    int CalcPartials(double f1, double* freqs) const;
    void CalcAmplitudes(int nPartials, const double* freqs, double* An0) const;
    void CalcQs(int nPartials, double* Qn, double Q0) const;
    void CalcTaun(int nPartials, const double* freqs, const double* Qn, double* Tn)
const;

protected:
    double m_partFreq[m_nBs];
};
```

### Κώδικας 15: Δήλωση κλάσης CVIWoodblock

```
#include "VIWoodblock.h"
#include <cassert>

CVIWoodblock::CVIWoodblock()
{
    GeneratePartFreqs();
}

CVIWoodblock::~~CVIWoodblock()
{
}

bool CVIWoodblock::Strike(int nNote, std::vector<float>& arSF)
{
    const double dt = 1 / (double)GetSampleRate();
    const double Q0 = 80;
    double t = 0.;
    double f1 = NoteFrequency(nNote);

    // Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
    double freqs[m_nBs];
    memset(freqs, 0, sizeof(freqs));
    int nPartials = CalcPartials(f1, freqs);

    // Υπολογισμός αρχικών πλατών
    double An0[m_nBs];
    memset(An0, 0, sizeof(An0));
    CalcAmplitudes(nPartials, freqs, An0);

    // Υπολογισμός συντελεστών ποιότητας
    double Qn[m_nBs];
    memset(Qn, 0, sizeof(Qn));
    CalcQs(nPartials, Qn, Q0);

    // Υπολογισμός συντελεστή απωλειών
    double dn[m_nBs];
    memset(dn, 0, sizeof(Qn));
    CalcTaun(nPartials, freqs, Qn, dn);

    // Συντελεστές helmholtz
```

```

double fH = 0.6 * freqs[0]; // 100;
double QH = 15;
double dH = QH / (m_pi * fH);
double AH = 0.5 * An0[0];

// Υπολογισμός χρονικής διάρκειας
const double thres = 0.001;
double t_end = 0.;

for (int i = 0; i < nPartials; ++i)
    t_end = std::max(-log(thres) * dn[i], t_end);

t_end = std::max(-log(thres) * dH, t_end);

// Εφαρμογή τύπου
std::vector<double> arSD;

while (t <= t_end)
{
    double y = 0;

    for (int j = 0; j < nPartials; ++j)
    {
        for (int k = 0; k < 25; ++k)
        {
            double dsp = 0.01;

            double f = freqs[j] * (1. + dsp * k);
            double a = An0[j] * exp(-t / dn[j]) * exp(-k/4);

            y += a * sin(2 * m_pi * f * t);

            f = freqs[j] * (1. - dsp * k);
            a = An0[j] * exp(-t / dn[j]) * exp(-k/4);

            y += a * sin(2 * m_pi * f * t);
        }

        y += AH * exp(-t / dH) * sin(2 * m_pi * fH * t);
    }

    arSD.push_back(y);

    t += dt;
}

// Κανονικοποίηση
double mx = 0.;
for (int i = 0; i < arSD.size(); ++i)
    mx = std::max(mx, abs(arSD[i]));

for (int i = 0; i < arSD.size(); ++i)
    arSD[i] = arSD[i] / mx;

// Θόρυβος
std::vector<double> arNoise;
WhiteNoise(arNoise, (int)arSD.size());
Bandpass(arNoise, 3600, 0.35, (double)GetSampleRate());

for (int i = 0; i < arSD.size(); ++i)
{
    double t = dt * i;

    arSD[i] += arNoise[i] * exp(-t / 0.0025);
}

mx = 0.;
for (int i = 0; i < arSD.size(); ++i)
    mx = std::max(mx, abs(arSD[i]));

// Κανονικοποίηση & εφαρμογή έντασης
for (int i = 0; i < arSD.size(); ++i)
    arSF.push_back((float)(GetGain() * arSD[i] / mx));

```

```
        return true;
    }

void CVIWoodblock::GeneratePartFreqs()
{
    for (int i = 0; i < BCount(); ++i)
        m_partFreq[i] = pow(BValues()[i] / BValues()[0], 2.);
}

// Υπολογισμός αποδεκτών ιδιοσυχνοτήτων
int CVIWoodblock::CalcPartials(double f1, double* freqs) const
{
    assert(freqs);

    int nPartials = 0;

    for (int i = 0; i < BCount(); ++i)
    {
        double fi = f1 * m_partFreq[i];

        if (fi > GetMaxFrequency())
        {
            nPartials = i - 1;

            break;
        }

        freqs[i] = fi;
    }

    if (nPartials == 0)
        nPartials = BCount() - 1;

    return nPartials;
}

// Υπολογισμός πλατιών
void CVIWoodblock::CalcAmplitudes(int nPartials, const double* freqs, double* An0) const
{
    assert(freqs && An0);

    // Υπολογισμός
    for (int i = 0; i < nPartials; ++i)
        An0[i] = 1. / pow(freqs[i], 1.3);
}

void CVIWoodblock::CalcQn(int nPartials, double* Qn, double Q0) const
{
    assert(Qn);

    Qn[0] = Q0;

    for (int i = 1; i < nPartials; ++i)
        Qn[i] = Q0 / (1. + 0.4 * i);
}

void CVIWoodblock::CalcTn(int nPartials, const double* freqs, const double* Qn, double* Tn) const
{
    assert(Qn && Tn);

    for (int i = 0; i < nPartials; ++i)
        Tn[i] = Qn[i] / (m_pi * freqs[i]);
}
```

**Κώδικας 16: Υλοποίηση κλάσης CVIWoodblock**

## Παράρτημα Η: Ο κώδικας του πιάνου

```
#pragma once

#include "VInstr.h"
#include <vector>

class CVIPiano : public CVInstr
{
public:
    struct NoteData
    {
        NoteData(int nNote = 0, double start = 0., double length = 0.)
        {
            this->nNote = nNote;
            this->start = start;
            this->length = length;
        }

        int nNote;
        double start;
        double length;
    };

protected:
    struct Partial
    {
        double freq;
        double phase;
        double amp;
        double decay;
    };

public:
    CVIPiano();

    virtual ~CVIPiano();

    bool Press(int nNote, double length, std::vector<float>& arSF);

    bool Play(std::vector < NoteData >& piece, std::vector<float>& arSamples);

protected:
    void ApplySoundboardFilters(std::vector<float>& arSamples) const;
    void ApplyResonator(std::vector<float>& arSamples, double fc, double Q, double g)
const;
    void Normalize(std::vector<float>& arSamples, float gain) const;
    bool Press(int nNote, double detune, double length, std::vector<float>& arSF);
    int NoteStrings(int nNote) const;
    void MergeArrays(std::vector<float>& ar1, const std::vector<float>& ar2) const;
    void MergeArrays(std::vector<float>& ar1, const std::vector<float>& ar2, const
std::vector<float>& ar3) const;
    void CalcInitialAmpls(std::vector<Partial> &partials) const;
    void CalcDecays(std::vector<Partial>& partials, double f0) const;
    void CalcPartials(std::vector<Partial>& partials, double f0, double B) const;
    void CalcLongParts(std::vector<Partial>& partials, double f0) const;

protected:
    double m_a, m_b, m_c;
    double m_tAttack;
};
```

**Κώδικας 17: Η δήλωση της κλάσης CVIPiano**

```
#include "VIPiano.h"

CVIPiano::CVIPiano()
{
    m_a = -9.5;
    m_b = -1.20;
    m_c = 0.085;

    m_tAttack = 0.01;

    srand(0);
}

CVIPiano::~CVIPiano()
{
}

bool CVIPiano::Play(std::vector < NoteData >& piece, std::vector<float>& arSamples)
{
    for (int i = 0; i < (int)piece.size(); ++i)
    {
        std::vector<float> arSF;

        Press(piece[i].nNote, piece[i].length, arSF);

        int nStart = (int)(piece[i].start * GetSampleRate());
        int nEnd = nStart + (int)arSF.size();

        if (nStart > arSamples.size())
        {
            for (int j = (int)arSamples.size(); j < nStart; ++j)
                arSamples.push_back(0);
        }

        for (int j = nStart; j < nEnd; ++j)
        {
            if (j < arSamples.size())
                arSamples[j] += arSF[j - nStart];
            else
                arSamples.push_back(arSF[j - nStart]);
        }
    }

    // Μισό δευτερόλεπτο σιωπής...
    for (int i = 0; i < GetSampleRate() / 2; ++i)
        arSamples.push_back(0);

    return true;
}

bool CVIPiano::Press(int nNote, double length, std::vector<float>& arSF)
{
    switch (NoteStrings(nNote))
    {
    {
    default:
    case 1:
    {
        Press(nNote, 0., length, arSF);
    }
    break;

    case 2:
    {
        std::vector<float> artmp;

        Press(nNote, 0., length, arSF);
        Press(nNote, 0.02, length, artmp);

        MergeArrays(arSF, artmp);
    }
    break;
    }
    }
}
```

```

case 3:
{
    std::vector<float> artmp1, artmp2;

    Press(nNote, 0., length, arSF);
    Press(nNote, 0.02, length, artmp1);
    Press(nNote, -0.02, length, artmp2);

    MergeArrays(arSF, artmp1);
    MergeArrays(arSF, artmp2);
}
break;

}

ApplySoundboardFilters(arSF);

Normalize(arSF, (float)GetGain());

return true;
}

bool CVIPiano::Press(int nNote, double detune, double length, std::vector<float>& arSF)
{
    const double pi2 = 2. * m_pi;
    const double dt = 1 / (double)GetSampleRate();

    double f0 = NoteFrequencyEx((double)nNote + detune);
    double t = 0.;

    // υπολογισμός B για την συγκεκριμένη νότα
    double B = exp(m_a + m_b * log(f0) + m_c * pow(log(f0),2));

    // Υπολογισμός αρμονικών
    std::vector<Partial> partials;
    CalcPartials(partials, f0, B);

    // Υπολογισμός αρχικού πλάτους
    CalcInitialAmpls(partials);

    // Υπολογισμός τ
    CalcDecays(partials, f0);

    // Διαμήκη κύματα
    CalcLongParts(partials, f0);

    int nSamples = (int)(length * GetSampleRate()) + 1;

    for (int nSample = 0; nSample < nSamples; ++nSample)
    {
        double y = 0;

        for (int i = 0; i < partials.size(); ++i)
        {
            double amp = partials[i].amp * exp(-t * partials[i].decay);

            double spr = sin(pi2 * partials[i].freq * t + partials[i].phase);

            y += amp * spr;
        }

        double attack = 1. - exp(-t / m_tAttack);

        y *= attack;

        //////////////////////////////////////

        arSF.push_back((float)y);

        t += dt;
    }

    //////////////////////////////////////
    // 25ms release time

```

```

int nTail = (int)(0.025 * GetSampleRate());
for (int i = 0; i < nTail; ++i)
{
    float u = 1.f - (float)i / (float)(nTail - 1);

    int nIndex = (int)arSF.size() - nTail + i;

    arSF[nIndex] *= u;
}

return true;
}

void CVIPiano::CalcPartials(std::vector<Partial>& partials, double f0, double B) const
{
    const double pi2 = 2. * m_pi;
    const double maxFreq = GetMaxFrequency();

    double fn = f0;

    int n = 1;

    while (fn <= maxFreq)
    {
        Partial partial;
        partial.freq = fn;
        partial.phase = pi2 * (double)rand() / (double)RAND_MAX; // Τυχαία φάση
        partial.amp = 0;
        partial.decay = 0;

        partials.push_back(partial);

        ++n;

        fn = n * f0 * sqrt(1 + B * pow((double)n, 2.));
    }

    //int np = std::min(4, (int)partials.size());
    //for (int i=0;i<np;++i)
    //    partials[i].phase = 0;
}

void CVIPiano::CalcInitialAmpls(std::vector<Partial>& partials) const
{
    const double b = 1. / 7.;
    const double a = 0.05;

    for (int i = 0; i < partials.size(); ++i)
    {
        double n = (double)(i + 1);

        double sn = sin(n * m_pi * b);
        double ex = exp(-a * n);

        partials[i].amp = sn * ex;
    }
}

void CVIPiano::CalcDecays(std::vector<Partial>& partials, double f0) const
{
    double tau1 = 0.8;
    double b = 0.0015;

    double u = (Clamp(f0, 55., 1046.5) - 55.) / (1046.5 - 55.);
    double glb = exp(-log(6) * u);

    for (int i = 0; i < partials.size(); ++i)
    {
        double n = (double)(i + 1);

        double freq = partials[i].freq;

```

```
        partials[i].decay = tau1 / (glb * (1 + b * pow(n, 2.)));
    }
}

void CVIPiano::Normalize(std::vector<float>& arSamples, float gain) const
{
    float mx = 0;

    for (int i = 0; i < arSamples.size(); ++i)
        mx = abs(arSamples[i]) > mx ? abs(arSamples[i]) : mx;

    if (mx <= 0)
        return;

    float fac = (float)(gain / mx);

    for (int i = 0; i < arSamples.size(); ++i)
        arSamples[i] *= fac;
}

void CVIPiano::ApplySoundboardFilters(std::vector<float>& arSamples) const
{
    ApplyResonator(arSamples, 100., 4, 0.7);
    ApplyResonator(arSamples, 300., 4, 0.5);
    ApplyResonator(arSamples, 800, 3, 0.3);

    //ApplyResonator(arSamples, 2500., 2, 0.1);
    //ApplyResonator(arSamples, 4000., 6, 0.15);
}

void CVIPiano::ApplyResonator(std::vector<float>& arSamples, double fc, double Q, double
g) const
{
    double fs = (double)GetSampleRate();
    double thetac = 2. * m_pi * fc / fs;
    double alpha = sin(thetac) / (2. * Q);

    double a0 = 1. + alpha;

    double a[3] = { a0, (-2. * cos(thetac)) / a0, (1. - alpha) / a0 };
    double b[3] = { g * alpha / a0, 0. / a0, -g * alpha / a0 };

    double x1 = 0.f;
    double x2 = 0.f;
    double y1 = 0.f;
    double y2 = 0.f;

    for (int i = 0; i < arSamples.size(); ++i)
    {
        double x = arSamples[i];

        double y = b[0] * x + b[1] * x1 + b[2] * x2 - a[1] * y1 - a[2] * y2;

        x2 = x1;
        x1 = x;
        y2 = y1;
        y1 = y;

        arSamples[i] = (float)y;
        //arSamples[i] += (float)y;
    }
}

int CVIPiano::NoteStrings(int nNote) const
{
    if (nNote < 36) // C2
        return 1;

    if (nNote < 53) // F3
        return 2;

    return 3;
}
```

```
void CVIPiano::MergeArrays(std::vector<float>& ar1, const std::vector<float>& ar2) const
{
    int nSz = std::max((int)ar1.size(), (int)ar2.size());

    for (int i = (int)ar1.size(); i < nSz; ++i)
        ar1.push_back(0);

    for (int i = 0; i < (int)ar2.size(); ++i)
        ar1[i] += ar2[i];
}

void CVIPiano::MergeArrays(std::vector<float>& ar1, const std::vector<float>& ar2, const
std::vector<float>& ar3) const
{
    int nSz = std::max((int)ar1.size(), (int)ar2.size());

    nSz = std::max(nSz, (int)ar3.size());

    for (int i = (int)ar1.size(); i < nSz; ++i)
        ar1.push_back(0);

    for (int i = 0; i < (int)ar2.size(); ++i)
        ar1[i] += ar2[i];

    for (int i = 0; i < (int)ar3.size(); ++i)
        ar1[i] += ar3[i];
}

void CVIPiano::CalcLongParts(std::vector<Partial>& partials, double f0) const
{
    const double maxFreq = GetMaxFrequency();
    const double vL = 5100.0;

    for (int i=1; ++i)
    {
        Partial partial;

        // συχνότητα
        partial.freq = i * vL / 2.;

        // πλάτος κύματος
        partial.amp = 0.04 * exp(-0.8 * i);

        // τ
        partial.decay = 80. + 40. * i;

        // Τυχαία φάση
        partial.phase = 2 * m_pi * (double)rand() / (double)RAND_MAX;

        if (partial.freq >= maxFreq)
            break;

        partials.push_back(partial);
    }
}
```

**Κώδικας 18: Υλοποίηση της κλάσης CVIPiano**

Υπεύθυνη Δήλωση Συγγραφέα:

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν.1599/1986, η παρούσα εργασία αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον και πληρούν τους κανόνες της επιστημονικής παράθεσης.