

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

3ο Εξάμηνο

Σήματα και Συστήματα - Εργασία Matlab (2022-23)

Παναγιώτης Σταματόπουλος

Α.Μ: 03120096

Εξάμηνο: 5ο

**Άσκηση 1:**

1. Αρχικά καταγράφω το μικρό μου όνομα και το αποθηκεύω ως “name.wav” διάρκειας 2 δευτερολέπτων και συχνότητας δειγματοληψίας 8000, με χρήση των εντολών **audiorecorder()**, **recordblocking()** και **audiowrite()**. Ο κώδικας αυτός στο αρχείο “program.m” βρίσκεται εντός σχολίων.
2. Στη συνέχεια εισάγουμε το καταγεγραμμένο αρχείο ήχου με την εντολή **audioread()** και το αποθηκεύουμε στη μεταβλητή/διάνυσμα “nameData”. Το σχεδιάζουμε στο πεδίο του χρόνου με την εντολή **plot()**, με τον άξονα x βαθμονομημένο σε **(Figure 1)**. Για το διάστημα των 50 διαλέγουμε τις χρονικές στιγμές των και . Σχεδιάζουμε το τμήμα **(Figure 2)** και παράλληλα χρησιμοποιώντας την εντολή **sound()** (εντός σχολίων στο πρόγραμμα), ακούμε το γράμμα «α». Αν κάνουμε την ίδια διαδικασία για το ακριβώς επόμενο χρονικό παράθυρο (0.9 έως 0.95), ξεχωρίζουμε το γράμμα «ν». Επομένως συμπεραίνουμε ότι το απόσπασμα αντιστοιχεί στο πρώτο «α» του ονόματός μου. Από το διάγραμμα υπολογίζουμε την περίοδο με βάση τα επιλεγμένα σημεία.
3. Κανονικοποιούμε το σήμα στο διάστημα [-1,1] και φτιάχνουμε το παράθυρο Hamming (υλοποιήθηκε χωρίς τη χρήση της εντολής **hamming()**). Με την εντολή **conv()**, πραγματοποιούμε την συνέλιξη του τετραγώνου του κανονικοποιημένου σήματος με το παράθυρο Hamming η οποία ισούται με την ενέργεια του αρχικού σήματος και στη συνέχεια κανονικοποιούμε ξανά το σήμα με βάση του πλάτους της ενέργειας, προκειμένου να σχεδιαστούν στο ίδιο διάγραμμα και στην ίδια κλίμακα **(Figure 3)**. Φαίνεται ότι η ενέργεια παίρνει τις μεγαλύτερες τιμές της στην περιοχή όπου το σήμα είναι πιο πυκνό, ενώ μακριά από αυτό μηδενίζεται, όπως είναι λογικό.
4. Εφαρμόζουμε τον διακριτό μετασχηματισμό Fourier με την εντολή **fft()** στο τμήμα που απομονώσαμε στο ερώτημα β) και σχεδιάζουμε το συχνοτικό περιεχόμενό του σε γραμμική **(Figure 4)** και σε λογαριθμική **(Figure 5)** κλίμακα, με εύρος συχνοτήτων στον οριζόντιο άξονα 0-8000
5. Από το διάγραμμα σε γραμμική κλίμακα, παρατηρούμε ότι η θεμελιώδης συχνότητα του σήματος είναι , δηλαδή η οποία πράγματι συμπίπτει με τη θεμελιώδη συχνότητα που υπολογίσαμε στο ερώτημα β).

**Άσκηση 2:**

1. Φορτώνουμε στο Matlab την εικόνα “leaf.png” με την εντολή **imread()** και την απεικονίζουμε με την εντολή **imshow().** **(Figure 6)**
2. Απομονώνουμε τα μονοδιάστατα σήματα της παραμετροποιη-μένης καμπύλης με τις εντολές **find()** και **bwtraceboundary()** και τα σχεδιάζουμε σε κοινό διάγραμμα. **(Figure 7)**
3. Ορίζουμε το μιγαδικό σήμα , εκτελούμε τον μετασχηματισμό Fourier σε αυτό και σχεδιάζουμε το μέτρο του (**abs()**) με την εντολή **plot()**. **(Figure 8)**
4. Για τις διάφορες τιμές του Μ υπολογίζουμε το σήμα με τη χρήση της εντολής **Fr(N, M, n, Z[k])** που έχουμε ορίσει. Έχουμε και τα οποία όμως χρειάζεται να στρογγυλοποιήσουμε γιατί δεν έχουν ακέραιες τιμές. Στη συνέχεια τα ανακατασκευάζουμε με την εντολή **reconstruct()** και τα σχεδιάζουμε με την εντολή **imshow()**. Τελικά προκύπτει:
5. Μ = 10: **(Figure 9)**
6. M = 50: **(Figure 10)**
7. M = 200: **(Figure 11)**

Παρατηρούμε ότι το σχέδιο που προκύπτει δεν μοιάζει με το αρχικό, αλλά αποτελείται από υπέρθεση των κυκλικών τροχιών που αντιστοιχούν στους Μ πρώτους συντελεστές της διακριτής σειράς Fourier, με αποτέλεσμα να θυμίζει μια «μουτζούρα». Συγκρίνοντας τις εικόνες για τις διάφορες τιμές του Μ, φαίνεται ότι ο αριθμός των κυκλικών τροχιών δεν επηρεάζει τη βασική μορφή του σχεδίου, αλλά την ευκρίνειά του. Επομένως η διαφορά μεταξύ της αρχικής εικόνας και της ανακατασκευασμένης οφείλεται στην επιλογή των πρώτων μόνο συντελεστών της σειράς Fourier.

1. Για τις ίδιες τιμές του Μ με το ερώτημα δ) υπολογίζουμε το σήμα με χρήση των εντολών **FrK(N, M, n, Z[k])** και **Fr(N, M, n, Z[k])** που έχουμε ορίσει. Επαναλαμβάνουμε την ίδια διαδικασία με το προηγούμενο ερώτημα:
2. Μ = 10: (Figure 12)
3. Μ = 50: (Figure 13)
4. Μ = 200: (Figure 14)

Παρατηρούμε ότι το σχέδιο που προκύπτει πράγματι έχει τη μορφή της αρχικής εικόνας ήδη από την πρώτη τιμή του Μ, ενώ για Μ = 200 την έχουμε ανακατασκευάσει πλήρως. Παρατηρούμε πως για την ίδια τιμή του Μ οι εικόνες που προκύπτουν από τα ερωτήματα δ) και ε) έχουν σημαντική διαφορά, η οποία οφείλεται στους συντελεστές που διαλέγουμε στις δύο περιπτώσεις και όχι στο πλήθος τους, αφού για τον ίδιο αριθμό συντελεστών, στο ερώτημα ε) γίνεται υπέρθεση των Μ/2 πρώτων και των Μ/2 τελευταίων. Με αυτή τη μέθοδο παρατηρούμε ότι η αύξηση του πλήθους των συντελεστών συνεπάγεται την βελτίωση της ακρίβειας της απεικόνισης.

1. Στο “MSPaint” σχεδιάζω μια ασπρόμαυρη εικόνα σύμφωνα με την υπόδειξη του ερωτήματος και την αποθηκεύω ως “customImage.bmp”. **(Figure 15)** Επαναλαμβάνουμε τα προηγούμενα ερωτήματα:
2. Απομονώνουμε τα μονοδιάστατα σήματα x[n], y[n] και τα απεικονίζουμε σε κοινό διάγραμμα. **(Figure 16)**
3. Ορίζουμε το μιγαδικό σήμα , εκτελούμε τον διακριτό μετασχηματισμό Fourier και σχεδιάζουμε το μέτρο του. **(Figure 17)**
4. Ανακατασκευάζουμε την εικόνα σύμφωνα με το ερώτημα δ) για τις διάφορες τιμές του Μ:

* Μ = 10: **(Figure 18)**
* Μ = 50: **(Figure 19)**
* Μ = 200: **(Figure 20)**

1. Ανακατασκευάζουμε την εικόνα σύμφωνα με το ερώτημα ε) για τις διάφορες τιμές του Μ:

* Μ = 10: **(Figure 21)**
* Μ = 50: **(Figure 22)**
* Μ = 200: **(Figure 23)**

Παρατηρούμε ότι τα συμπεράσματα που προκύπτουν από τα ερωτήματα δ) και ε) επιβεβαιώνονται για την “customImage.bmp”.

**Άσκηση 3:**

**3.1:**

1. Αρχικά δημιουργούμε τα διανύσματα πόλων και μηδενικών του συστήματος και σχεδιάζουμε το διάγραμμα πόλων-μηδενικών. **(Figure 24)** Στη συνέχεια, υπολογίζουμε τα διανύσματα των συντελεστών a, b που αντιστοιχούν στο φίλτρο με χρήση της συνάρτησης **zp2tf()**.
2. Στη συνέχεια, σχεδιάζουμε την απόκριση πλάτους και φάσης του φίλτρου με τη χρήση της εντολής **freqz()**. **(Figure 25)** Παρατηρούμε ότι το φίλτρο εμφανίζει μέγιστο πλάτος στη συχνότητα με κέρδος 0 και στο διάγραμμα της φάσης η καμπύλη γίνεται σχεδόν κατακόρυφη στη συχνότητα αυτή. Αυτό σημαίνει ότι το φίλτρο είναι ζωνοπερατό, αποκόπτει δηλαδή τις συχνότητες μακριά από τη συχνότητα , διατηρώντας το ίδιο πλάτος αφού έχουμε μηδενικό λογαριθμικό κέρδος.
3. Σχεδιάζουμε την απόκριση του συστήματος με την εντολή **impz()** **(Figure 26)**, και την βηματική απόκριση με την εντολή **stepz()**. **(Figure 27)**
4. Διατηρώντας τα ίδια μηδενικά και το ίδιο κέρδος, μετακινούμε τους πόλους του συστήματος και σχεδιάζουμε την βηματική απόκριση και την απόκριση πλάτους:
5. Πόλοι:

Διάγραμμα Πόλων-Μηδενικών: **(Figure 28)**

Βηματική Απόκριση: **(Figure 31)**

Απόκριση Πλάτους-Φάσης: **(Figure 34)**

1. Πόλοι:

Διάγραμμα Πόλων-Μηδενικών: **(Figure 29)**

Βηματική Απόκριση: **(Figure 32)**

Απόκριση Πλάτους-Φάσης: **(Figure 35)**

1. Πόλοι:

Διάγραμμα Πόλων-Μηδενικών: **(Figure 30)**

Βηματική Απόκριση: **(Figure 33)**

Από τα διαγράμματα προκύπτουν τα εξής συμπεράσματα σχετικά με την απόσταση των πόλων από το μοναδιαίο κύκλο:

* Βηματική απόκριση: Διακρίνουμε τρεις περιπτώσεις των πόλων: εσωτερικά του μοναδιαίου κύκλου, πάνω του και εξωτερικά. Εσωτερικά η βηματική απόκριση έχει φθίνουσα ημιτονοειδή μορφή, εξωτερικά αύξουσα ενώ πάνω στον κύκλο αναλλοίωτη.
* Απόκριση Πλάτους: Όσο πιο κοντά βρίσκονται οι πόλοι στον μοναδιαίο κύκλο, τόσο πιο απότομη είναι η κορυφή του ζωνοπερατού φίλτρου.

1. Φτιάχνουμε τα επιμέρους ημιτονοειδή σήματα και στη συνέχεια τα υπερθέτουμε για να φτιάξουμε το σήμα x[n]. Χρησιμοποιώντας το φίλτρο που φτιάξαμε στην αρχή, διεγείρουμε το σύστημα με είσοδο την x[n] χρησιμοποιώντας την εντολή **filter()** και σχεδιάζουμε το σήμα εξόδου σε κοινό διάγραμμα με το σήμα εισόδου. **(Figure 36)**

Παρατηρούμε ότι το σήμα εξόδου έχει σημαντικά μικρότερο πλάτος (περίπου το μισό) από το σήμα εισόδου και διαφορετική μορφή. Οι κορυφές του x[n] φαίνονται να αποτελούνται από το άθροισμα 2-3 επιμέρους κορυφών, ενώ οι κορυφές του y[n] τείνουν περισσότερο στην τυπική ημιτονοειδή μορφή. Ωστόσο, και τα δύο σήματα έχουν μηδενική διαφορά φάσης.

1. Επαναλαμβάνουμε τα ερωτήματα α) και β) για διπλούς συζυγείς πόλους στις θέσεις , διατηρώντας τα ίδια μηδενικά. Σχεδιάζουμε το διάγραμμα πόλων-μηδενικών **(Figure 37)**, φτιάχνουμε το καινούριο φίλτρο και σχεδιάζουμε το διάγραμμα απόκρισης πλάτους. **(Figure 38)**

Παρατηρούμε ότι η κορυφή του φίλτρου σε σχέση με αυτό του πρώτου ερωτήματος μετατοπίζεται προς τα αριστερά και αυξάνεται το κέρδος σημαντικά, ενώ η φάση του είναι σχεδόν παντού αρνητική.

**3.2:**

1. Φορτώνουμε στο Matlab το αρχείο “viola\_series.wav” με συχνότητα δειγματοληψίας χρησιμοποιώντας την εντολή **audioread()** και το αποθηκεύουμε στη μεταβλητή “viola”. Στη συνέχεια σχεδιάζουμε και ακούμε το σήμα με τις εντολές **plot()** και **sound()** αντίστοιχα. **(Figure 39)**
2. Εφαρμόζουμε διακριτό μετασχηματισμό Fourier στο σήμα μας με την εντολή **fft()** και σχεδιάζουμε το μέτρο του φάσματος σε γραμμική κλίμακα, με εύρος συχνοτήτων . **(Figure 40)**

Στο διάγραμμα φαίνονται οι 5 πρώτες αρμονικές συχνότητες του σήματος και η επίδρασή τους στη δημιουργία του σήματος. Η θεμελιώδης και η πρώτη αρμονική φαίνεται να έχουν τη μεγαλύτερη επιρροή σε σχέση με τις υπόλοιπες αρμονικές, όπως φαίνεται από τα πλάτη τους.

1. Εκτελούμε την διαδικασία του ερωτήματος α) για το αρχείο “viola\_note.wav”. **(Figure 41)**

Ομοίως και για το β). **(Figure 42)**

Το διάγραμμα έχει παρόμοια μορφή με αυτή του ερωτήματος β) και διακρίνουμε τη θεμελιώδη συχνότητα με το μεγαλύτερο μέτρο και τις 3 πρώτες αρμονικές με σημαντικά μικρότερο μέτρο. Υπολογίζουμε εποπτικά τη θεμελιώδη συχνότητα .

1. Θέλουμε να υλοποιήσουμε ένα ζωνοπερατό φίλτρο για να απομονώσουμε την 2η αρμονική του σήματος, επομένως πρέπει να φτιάξουμε τους κατάλληλους πόλους, μηδενικά και κέρδος του φίλτρου. Θέτουμε μηδενικά στις θέσεις {-1,1}. Για τους πόλους, υπολογίζουμε την κανονικοποιημένη συχνότητα η οποία θα είναι η γωνία (σε rad) του φάσορα. Όσο πιο κοντά είναι το πλάτος του φάσορα στη μονάδα, τόσο περισσότερες συχνότητες αποκόπτονται γύρω από την επιθυμητή, επομένως μετά από δοκιμές καταλήγουμε σε πλάτος 0.9995. Μετατρέπουμε το φάσορα σε μορφή και θέτουμε αυτόν και το συζυγή του ως πόλους του συστήματος. Με δοκιμές βρίσκουμε ότι για Κ = 0.0005 έχουμε κέρδος 0 dB στη συχνότητα που θέλουμε. **(Figure 43)**

Διεγείρουμε το σύστημα με το σήμα “viola\_note” και εφαρμόζουμε διακριτό μετασχηματισμό Fourier πάνω στην έξοδο του συστήματος. Σχεδιάζουμε το μέτρο του φάσματος γύρω από το 0 και παρατηρούμε πως η κυρίαρχη συχνότητα είναι περίπου , πράγματι σχεδόν όσο η θεωρητική . **(Figure 44)**

Στη συνέχεια, παίρνουμε ένα τμήμα του σήματος εξόδου διάρκειας και το σχεδιάζουμε στο πεδίο του χρόνου. **(Figure 45)**

Πραγματοποιούμε την ίδια διαδικασία για τη δεύτερη αρμονική με διαφορά στο πλάτος του φάσορα το οποίο επιλέγουμε να είναι μονάδα και στο κέρδος το οποίο επιλέγουμε να είναι 0.003. **(Figure 46)**

Εφαρμόζουμε πάλι μετασχηματισμό Fourier στο σήμα εξόδου μετά τη διέγερση και το σχεδιάζουμε όπως προηγουμένως. Πράγματι, η κύρια συχνότητα είναι πολύ κοντά στη θεωρητική . **(Figure 47)**

Παίρνουμε πάλι ένα τμήμα του σήματος εξόδου διάρκειας και το σχεδιάζουμε στο πεδίο του χρόνου. **(Figure 48)**

**Άσκηση 4:**

1. Φορτώνουμε στο Matlab το σήμα “mixture.wav” ως “mixture” με την εντολή **audioread()** και θέτουμε την μεταβλητή “Fs” ίση με τη συχνότητα δειγματοληψίας. Ακούμε το σήμα με την εντολή **sound()** και πλοτάρουμε το συνολικό φάσμα γύρω από το 0 με τις εντολές **fft()**, **fftshift()** και **plot()**. **(Figure 49)**

Παρατηρούμε ότι εμφανίζονται και οι δύο θεμελιώδης συχνότητες των επιμέρους νοτών, καθώς και οι αρμονικές τους μέχρι και την 5η.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Harmonic\Freq |  | | Normalized [0,2π] | |
| Fundamental | 350 | 440 | 0,137445 | 0,172788 |
| 2 | 700 | 880 | 0,274889 | 0,345575 |
| 3 | 1050 | 1320 | 0,412334 | 0,518363 |
| 4 | 1400 | 1760 | 0,549779 | 0,69115 |
| 5 | 1750 | 2200 | 0,687223 | 0,863938 |

1. Υπολογίζουμε τις συχνότητες:
2. Συνθέτουμε κατάλληλα φίλτρα για να απομονώσουμε τις συχνότητες με την τρόπο που περιγράφεται στην άσκηση 3.2 ερώτημα δ) και προσθέτουμε τελικά τα σήματα εξόδου που προκύπτουν από τη διέγερση κάθε συστήματος για την αντίστοιχη συχνότητα. Έχουμε ανακατασκευάσει έτσι τις δύο νότες.
3. Ακούμε τα ηχητικά που μας δίνονται και τα τμήματά τους παράλληλα με τις ανακατασκευασμένες νότες και συγκρίνουμε τα διαγράμματα των τεσσάρων νοτών. Τελικά φαίνεται πως το σήμα θεμελιώδους συχνότητας 350Hz αντιστοιχεί στο “reed\_acoustic\_037-065-075.wav” και το σήμα των 440Hz στο “flute\_acoustic\_\_002-069-025.wav”.

**(Figure 50)** **(Figure 51)** **(Figure 52)** **(Figure 53)**

1. Φορτώνουμε στο Matlab το ηχητικό αρχείο “mixture2.wav” με την ίδια συχνότητα δειγματοληψίας. Εκτελούμε την ίδια διαδικασία με το ερώτημα δ) για να απομονώσουμε τις συχνότητες. Παρατηρούμε ότι η μορφή της ανακατασκευασμένης νότας συχνότητας 220Hz μοιάζει αρκετά με αυτή της ανακατασκευασμένης των 440Hz, το οποίο οφείλεται στο γεγονός ότι ορισμένες αρμονικές τους συμπίπτουν.

**(Figure 54)** **(Figure 55)** **(Figure 56)** **(Figure 57)**

Σημείωση:

Παραθέτω τον τίτλο των αντίστοιχων διαγραμμάτων στην αναφορά αντί για τα διαγράμματα αυτούσια για την μείωση του μεγέθους της αναφοράς.

Παραθέτω μαζί με τα στοιχεία του αρχείου zip και την εικόνα customImage.bmp για το τελευταίο μέρος της 2ης άσκησης.