



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών  
Τομέας Σημάτων, Ελέγχου και Ρομποτικής  
Ζωγράφου 15773, Αθήνα

## Σήματα και Συστήματα - Εργασία MATLAB (2019-20)

---

### ΟΔΗΓΙΕΣ

- Στόχος της εργασίας είναι η εμπέδωση σημαντικών εννοιών του μαθήματος αλλά και γενικότερα η εξοικείωση με τις εφαρμογές του περιβάλλοντος **MATLAB** στο μάθημα 'Σήματα και Συστήματα'.
- Η εργασία είναι ατομική.
- Η εργασία είναι προαιρετική. Μπορεί να μετρήσει προσθετικά στο βαθμό έως και 10% της συνολικής βαθμολογίας τους μαθήματος.
- Τρόπος παράδοσης: Ηλεκτρονική υποβολή μέσω της σελίδας του μαθήματος η οποία είναι διαθέσιμη στο [mycourses.ntua.gr](https://mycourses.ntua.gr) (λεπτομέρειες θα δοθούν μέσω ανακοίνωσης).
- Παραδοτέα: Θα πρέπει να υποβληθεί ένα αρχείο zip το οποίο να περιλαμβάνει όλα τα ακόλουθα αρχεία:
  1. Ένα αρχείο `.m` που να περιέχει τον κώδικα MATLAB που γράψατε. Το αρχείο να ονομαστεί program.m. Συμπεριλάβετε επεξηγηματικά σχόλια στον κώδικα όπου θεωρείτε απαραίτητο.
  2. Αναφορά σε pdf (μη χρησιμοποιήσετε άλλα formats όπως `.doc`, `.docx`, κτλ) η οποία να ονομαστεί report.pdf. Παρακαλώ μην αποστείλετε χειρόγραφο και κατόπιν σαρωμένη (scanned) αναφορά γιατί θα είναι δύσκολη η βαθμολόγησή της.
  3. Ένα αρχείο `.txt` με τα προσωπικά στοιχεία σας καταχωρημένα σε 3 γραμμές: η πρώτη γραμμή να περιλαμβάνει τον αριθμό μητρώου (ΑΜ) σας, η δεύτερη και τρίτη γραμμή να περιλαμβάνουν το επώνυμο και το όνομα σας, αντίστοιχα, γραμμένα με ελληνικούς κεφαλαίους χαρακτήρες (ή με λατινικούς κεφαλαίους χαρακτήρες σε περίπτωση αλλοδαπών ονοματεπωνυμικών στοιχείων). Τα στοιχεία πρέπει να ταυτίζονται με τα στοιχεία σας που είναι καταχωρημένα

στη γραμματεία της Σχολής, π.χ., τρόπος γραφής, ορθογραφία ονόματος και επωνύμου. Το αρχείο να ονομαστεί info.txt.

4. Ένα αρχείο .wav που να περιέχει το ανακατασκευασμένο σήμα του ερωτήματος 2.2 στ) (βλ. μέρος 2.2 της εκφώνησης παρακάτω). Το αρχείο να ονομαστεί reconstructed.wav.

Βεβαιωθείτε πως το ονομάτεπώνυμο και το AM σας περιλαμβάνονται τόσο στον κώδικα MATLAB (εντός σχολίων), όσο και στην αναφορά. Το όνομα του αρχείου zip που θα υποβάλετε πρέπει να ταυτίζεται με το AM σας, π.χ., 12345.zip. Υπενθυμίζεται πως η τήρηση όλων των ανωτέρω χαρακτηριστικών των παραδοτέων είναι άκρως απαραίτητη για τη διεξαγωγή της βαθμολόγησης.

- Ημερομηνία παράδοσης: **εως και μία εβδομάδα μετά τη λήξη των μαθημάτων της χειμερινής περιόδου**
- Σημείωση: Για ερωτήσεις επικοινωνήστε με την Νάνσυ Ζλατίντση (nzlat@cs.ntua.gr)

---

## 1 Σχεδίαση Φίλτρων

Στο πρώτο μέρος της άσκησης, θα κατασκευάσουμε, με χρήση του Matlab, ορισμένα φίλτρα, με σκοπό να μελετήσουμε τις ιδιότητές τους.

### 1.1 Σχεδίαση Βαθυπερατών Φίλτρων (Κινούμενου Μέσου)

Ένα αιτιατό φίλτρο διακριτού χρόνου ορίζεται μέσω της εξίσωσης διαφορών:

$$y[n] + \sum_{i=1}^M a_i y[n-i] = \sum_{i=0}^N b_i x[n-i] \quad (1)$$

Συχνά, για την αποθορυβοποίηση (denoising) και την ομαλοποίηση (smoothing) σημάτων, χρησιμοποιούμε φίλτρα κινούμενου μέσου. Αυτά ορίζονται μέσω της αιτιατής εξίσωσης διαφορών:

$$y[n] = \frac{1}{N+1} \sum_{i=0}^N x[n-i] \quad (2)$$

- α) Ορίστε στο Matlab τα διανύσματα  $b, a$ , για φίλτρα κινούμενου μέσου για τιμές του  $N = 2, 4, 10$ .
- β) Σχεδιάστε την απόκριση πλάτους και φάσης των φίλτρων αυτών, με χρήση της εντολής **freqz()**, η οποία δέχεται ως ορίσματα τα διανύσματα  $b, a$  που ορίζουν την εξίσωση διαφορών του φίλτρου. Τι παρατηρείτε;
- γ) Σχεδιάστε τα διαγράμματα πόλων και μηδενικών των φίλτρων, με χρήση της συνάρτησης **zpd.m** (η οποία σας δίνεται). Για να μεταφέρετε τα φίλτρα σας σε μορφή πόλων μηδενικών, χρησιμοποιήστε τη συνάρτηση **tf2zp()**. Τι παρατηρείτε;

## 1.2 Σχεδίαση Ζωνοπερατών Φίλτρων

Στο πρώτο ερώτημα, υλοποιήθηκαν βαθυπερατά φίλτρα στο πεδίο του χρόνου. Εδώ, για τη σχεδίαση ζωνοπερατών (bandpass) φίλτρων, θα εργαστούμε στο πεδίο της συχνότητας.

- α) Σχεδιάστε, με χρήση Matlab, ένα φίλτρο με πόλους στις θέσεις  $\{0.68 \pm 0.51i\}$  και μηδενικά στις θέσεις  $\{1.2, -0.6\}$ . Δώστε ένα διάγραμμα πόλων - μηδενικών του συστήματος, με χρήση της συνάρτησης `zpd.m`. Βρείτε τα διανύσματα συντελεστών  $a, b$  που αντιστοιχούν σε αυτό το φίλτρο με χρήση της συνάρτησης `zp2tf`.
- β) Σχεδιάστε την απόκριση πλάτους και φάσης του φίλτρου, με χρήση της συνάρτησης `freqz()` και σχολιάστε τα αποτελέσματα.
- γ) Σχεδιάστε την κρουστική απόκριση του συστήματος (με χρήση της συνάρτησης `impz()`), καθώς και τη βηματική απόκριση (με χρήση της `stepz()`), χρησιμοποιώντας την αναπαράσταση του φίλτρου που προήλθε από χρήση της `zp2tf()`.
- δ) Μετακινήστε τους πόλους του συστήματος στις θέσεις  $\{0.76 \pm 0.57i\}$ ,  $\{0.8 \pm 0.6i\}$ , και τέλος  $\{0.84 \pm 0.63i\}$  (κρατώντας τα μηδενικά στις ίδιες θέσεις με πριν, δηλαδή  $\{1.2, -0.6\}$ ), και σχεδιάστε, για κάθε περίπτωση, τη βηματική απόκριση του συστήματος, καθώς και την απόκριση πλάτους για την πρώτη περίπτωση. Τι παρατηρείτε; Αιτιολογήστε την απάντησή σας με χρήση των διαγραμμάτων πόλων-μηδενικών.
- ε) Διεγείρετε το σύστημα του ερωτήματος α) χρησιμοποιώντας, ως είσοδο, μία κρουστική παλμοσειρά. Για να δημιουργήσετε την παλμοσειρά, χρησιμοποιήστε τη συνάρτηση `gensig()`, η οποία δέχεται ως ορίσματα τον τύπο του παλμού (για κρουστική παλμοσειρά, `pulse`), την περίοδο του,  $T$ , τη διάρκεια του σήματος (σε sec), και τη χρονική απόσταση ανάμεσα σε δύο διαδοχικά δείγματα (επίσης σε sec). Συγκεκριμένα, χρησιμοποιήστε διάρκεια σήματος  $t = 100s$ , απόσταση διαδοχικών δειγμάτων  $t = 1s$ , ενώ για την περίοδο χρησιμοποιήστε τις τιμές  $T = 50s$  και  $T = 5s$ . Τι παρατηρείτε σε κάθε περίπτωση;
- στ) Επαναλάβετε το ερωτήματα α), β) για συζυγείς πόλους στις θέσεις  $\pm 0.8i$ , και διατηρώντας τα μηδενικά ως έχουν. Τι παρατηρείτε σχετικά με τη ζώνη διέλευσης του φίλτρου;

## 2 Ανάλυση και Ανασύνθεση Μουσικών Σημάτων

Στο δεύτερο μέρος αυτής της άσκησης, θα αναλύσουμε, με χρήση του Matlab, μουσικά σήματα (ηχογραφήσεις από συγκεκριμένες νότες) στο πεδίο της συχνότητας, με χρήση του Διακριτού Μετασχηματισμού Fourier (DFT) (ένας ταχύς αλγόριθμος υλοποίησής του είναι ο Fast Fourier Transform - FFT), και στη συνέχεια θα επιχειρήσουμε να τα ανασυνθέσουμε ως αθροίσματα ημιτονοειδών.

### 2.1 Ανάλυση Μουσικών Σημάτων

- α) Φορτώστε στο Matlab, με χρήση της εντολής `wavread()` (ή `audioread()`), τα αρχεία *flute\_note.wav*, *clarinet\_note.wav*, *cello\_note.wav* από το συμπληρωματικό υλικό της άσκησης, τα οποία αποτελούν ηχογραφήσεις από νότες φλάουτου, κλαρινέτου και τσέλου, με συχνότητα δειγματοληψίας 44.1 kHz. Ακούστε τα σήματα με χρήση της εντολής `sound()`.
- β) Για το κάθε σήμα, σχεδιάστε ένα απόσπασμά του στο πεδίο του χρόνου, με χρήση της εντολής `plot()`. Είναι τα σήματα περιοδικά; Αν ναι, υπολογίστε (εποπτικά) την περίοδό τους (σε δείγματα και σε ms).
- γ) Με χρήση της εντολής `fft()` για το κάθε σήμα σχεδιάστε το φάσμα του, και υπολογίστε τη θεμελιώδη συχνότητα (σε Hz). Επιβεβαιώστε τη σχέση μεταξύ θεμελιώδους συχνότητας και περιόδου. Τι παρατηρείτε σχετικά με τη μορφή του φάσματος, καθώς και την εμφάνιση αρμονικών υψηλότερης τάξης σε κάθε περίπτωση; Συνδέστε τα ηχητικά χαρακτηριστικά του κάθε σήματος (ηχόχρωμα) με το βαθμό εμφάνισης των αρμονικών.
- Υπόδειξη:** Για να υπολογίσετε απευθείας τη συχνότητα του σήματος από το φάσμα που σχεδιάσατε, μπορείτε να πάρετε αριθμό σημείων DFT ίσο με τη συχνότητα δειγματοληψίας των σημάτων.
- δ) Υπολογίστε την ενέργεια του κάθε σήματος, σε κυλιόμενα παράθυρα, αφού πρώτα κανονικοποιήσετε τα σήματα στο διάστημα  $[-1, 1]$ . Σημειώνεται ότι η ενέργεια ενός σήματος  $x[n]$  επικαλυπτόμενο από ένα παράθυρο  $w[n]$  δίνεται από τον εξής τύπο:

$$E[n] = \sum_{m=0}^M x^2[m]w^2[n-m] \quad (3)$$

Χρησιμοποιήστε τετραγωνικό παράθυρο, μήκους 1000 δειγμάτων. Για το κάθε σήμα, σχεδιάστε την ενέργεια του σήματος στο ίδιο διάγραμμα με το σήμα (κλιμακώστε το σήμα κατάλληλα). Τι παρατηρείτε;

**Υπόδειξη:** Παρατηρήστε ότι η παραπάνω εξίσωση μπορεί να εκφραστεί ως συνέλιξη, που μπορεί να υπολογιστεί στο Matlab μέσω της εντολής `conv()`.

- ε) Φορτώστε στο Matlab το αρχείο *cello\_note\_noisy.wav*, το οποίο έχει προκύψει μέσω της προσθήκης Γκαουσιανού θορύβου στο αρχικό σήμα νότας τσέλου, με παραμέτρους  $\{\mu, \sigma\} = \{0, 0.05\}$ . Υπολογίστε το φάσμα του με χρήση της εντολής `fft()`, και ακούστε το σήμα μέσω της εντολής `sound()`. Τι παρατηρείτε σε σχέση με το φάσμα του θορυβώδους σήματος;

στ) Χρησιμοποιώντας κατάλληλο βαθυπερατό φίλτρο (κινούμενου μέσου), επιχειρήστε να αποθρομβοποιήσετε το σήμα του ερωτήματος ε). Σχεδιάστε στο Matlab το φάσμα του φιλτραρισμένου σήματος, και ακούστε εκ νέου το φιλτραρισμένο σήμα. Σχολιάστε τα αποτελέσματα.

**Υπόδειξη:** Ρυθμίστε τη ζώνη διέλευσης του βαθυπερατού φίλτρου με κριτήριο το τμήμα του φάσματος του αρχικού σήματος που περιέχει σημαντικές αρμονικές.

ζ) Χρησιμοποιώντας το αρχείο *cello\_note.wav*, υλοποιήστε ένα ζωνοπερατό φίλτρο, με στόχο να απομονώσετε την 9η αρμονική του σήματος. Επαναλάβετε την ίδια διαδικασία για την 6η αρμονική. Σχεδιάστε τα φάσματα των προκύπτοντων σημάτων στο πεδίο της συχνότητας, καθώς και ένα απόσπασμά τους στο πεδίο του χρόνου. Σχολιάστε τα αποτελέσματα.

**Υπόδειξη:** Χρησιμοποιήστε όσα είδατε στο **Μέρος 1** τόσο σχετικά με τη σχέση της κεντρικής συχνότητας του φίλτρου με τη φάση των πόλων, όσο και τη σχέση του εύρους της ζώνης διέλευσης του φίλτρου με το μέτρο τους.

## 2.2 Σύνθεση Μουσικών Σημάτων ως Άθροισμα Ημιτονοειδών

Γνωρίζουμε, από την ανάλυση Fourier, ότι μπορούμε να εκφράσουμε ένα οποιοδήποτε σήμα ως άθροισμα άπειρων ημιτονοειδών (ημιτόνων ή συνημιτόνων):

$$x(t) = a_0 + \sum_{n=1}^{+\infty} a_n \cos(2\pi n f_0 t) + \sum_{n=1}^{+\infty} b_n \sin(2\pi n f_0 t) \quad (4)$$

Μπορούμε να τροποποιήσουμε την παραπάνω εξίσωση, ώστε να περιέχει μόνο έναν τύπο ημιτονοειδούς, ως:

$$x(t) = a_0 + \sum_{n=1}^{+\infty} c_n \cos(2\pi n f_0 t + \phi_n) \quad (5)$$

όπου οι  $c_n$ ,  $\phi_n$  είναι οι συντελεστές πλάτους και φάσης που αντιστοιχούν στη συχνότητα  $n f_0$ .

- α) Επιλέξτε ένα από τα τρία ηχητικά σήματα που αναλύσατε στο προηγούμενο ερώτημα. Απομονώστε από αυτό ένα τμήμα, με μήκος ίσο με ακέραιο πολλαπλάσιο της θεμελιώδους περιόδου (για παράδειγμα 10 φορές).
- β) Υπολογίστε, για το απόσπασμα που επιλέξατε, το διακριτό μετασχηματισμό Fourier, και πλοτάρτε το πλάτος του. Υποθέτοντας προσεγγιστικά ότι τα peaks του μετασχηματισμού αντιστοιχούν στις αρμονικές του σήματος, υπολογίστε τους συντελεστές πλάτους για κάθε σημαντική αρμονική (που ικανοποιεί κάποιο κατώφλι πλάτους) ως:

$$c_n = \frac{A_n}{A_1} \quad (6)$$

όπου  $A_n$  το πλάτος του μετασχηματισμού Fourier στην  $n$ -οστή αρμονική.

γ) Υπολογίστε τις παραμέτρους  $\phi_n$  για τις επιμέρους αρμονικές, χρησιμοποιώντας την εντολή **angle()** που επιστρέφει ως όρισμα τη φάση του Διακριτού Μετασχηματισμού Fourier στο διάστημα  $[-\pi, \pi]$ .

**Υπόδειξη:** Έχοντας βρει από το προηγούμενο ερώτημα τα σημεία (indexes) που αντιστοιχούν στις αρμονικές, αρκεί να πάρετε τις αντίστοιχες τιμές από το διάνυσμα φάσης.

δ) Προσθέστε τα επιμέρους ημιτονοειδή που προέκυψαν από τις παραμέτρους  $c_n, \phi_n$  που υπολογίσατε. Χρησιμοποιήστε την εντολή **linspace()** για να ορίσετε τη μεταβλητή  $t$ , με  $f_s$  δείγματα ανά δευτερόλεπτο.

ε) Ως έλεγχο ορθότητας, αντιπαραβάλλετε οπτικά το αρχικό απόσπασμα του σήματος με ένα απόσπασμα από το ανακατασκευασθέν, ίσου μήκους, καθώς και ακούστε τον τόνο που αντιστοιχεί σε αυτό - αφού κανονικοποιήσετε το σήμα σας στο διάστημα  $[-1, 1]$ . Είναι η ανακατασκευή τέλεια; Αν όχι, πόσες αρμονικές απαιτήθηκαν για να πάρετε ικανοποιητικό αποτέλεσμα; Σχολιάστε, συγκρίνοντας το αποτέλεσμα με το αντίστοιχο με χρήση μόνο της κύριας αρμονικής, καθώς και των 5 πρώτων αρμονικών.

**Υπόδειξη:** Επιλέξτε το απόσπασμα του ανακατασκευασμένου σήματος κατά τέτοιο τρόπο, ώστε να είναι συμφασικό με το αρχικό απόσπασμα.

στ) Αποθηκεύστε το ανακατασκευασμένο σήμα σε ένα αρχείο τύπου .wav, με χρήση της ρουτίνας **wavwrite()**.