ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
Τμήμα Επιστήμης Υπολογιστών

ΗΥ-215: Εφαρμοσμένα Μαθηματικά για Μηχανικούς
Εαρινό Εξάμηνο 2017-18
Διδάσκοντες: Γ. Στυλιανού, Γ. Καφεντζής

Τέταρτη Σειρά Ασκήσεων

Ημερομηνία Ανάθεσης: 19/3/2018 Ημερομηνία Παράδοσης: 30/3/2018, 16:00

Οι ασκήσεις με [$\star$] είναι **bonus**, $+10$ μονάδες η καθεμία στο βαθμό αυτής της σειράς ασκήσεων (δηλ. μπορείτε να πάρετε μέχρι $110/80$ σε αυτή τη σειρά.)

Άσκηση 1 - Μετασχηματισμός Fourier I - Ορισμός
Χρησιμοποιήστε τον ορισμό για να βρείτε το μετασχ. Fourier των παρακάτω σημάτων.

(α') $x(t) = e^{-2t}u(t-3)$

(β') $x(t) = e^{-4|t|}$

(γ') $x(t) = te^{-2t}u(t)$

Απ: (α) $X(f) = \frac{e^{-3(2+j2\pi f)}}{2+j2\pi f}$, (β) $X(f) = \frac{8}{16+4\pi^2 f^2}$, (γ) $X(f) = \frac{1}{(2+j2\pi f)^2}$

Άσκηση 2 - Μετασχηματισμός Fourier II - Ιδιότητες
Σας δίνεται ο μετασχ. Fourier ενός σήματος $x(t)$ ως

$$X(f) = \frac{4}{3+j2\pi f} \tag{1}$$

Βρείτε το μετασχ. Fourier των παρακάτω σημάτων με αποκλειστική χρήση ιδιοτήτων.

(α') $x(-2t)$

(β') $x(t-5)$

(γ') $x(8t-2)$

(δ') $tx(t)$

(ε') $e^{j6t}x(t)$

(ϛ') $\frac{dx(t)}{dt}$

Απ: (α) $\frac{2}{3-j\pi f}$, (β) $\frac{4e^{-j10\pi f}}{3+j2\pi f}$, (γ) $\frac{4}{24+j2\pi f}e^{-j\pi f/2}$,
(δ) $\frac{4}{(3+j2\pi f)^2}$, (ε) $\frac{4}{3+j(2\pi f-6)}$, (στ) $\frac{j8\pi f}{3+j2\pi f}$

Άσκηση 3 - Μετασχηματισμός Fourier III - Ιδιότητες
Υπολογίστε το ολοκλήρωμα

$$\int_{-\infty}^{+\infty} t^2 \left(\frac{\sin(t)}{\pi t}\right)^4 dt \tag{2}$$

Επιβεβαιώστε στο MATLAB την απάντηση χρησιμοποιώντας *συμβολική* ολοκλήρωση (συναρτήσεις `syms`, `int`).

Απ: $\frac{1}{2\pi^3}$

[$\star$] Άσκηση 4 - Μετασχηματισμός Fourier - Γρίφος :-)
Έστω ότι για ένα σήμα $x(t)$ μας δίνονται οι παρακάτω πληροφορίες:

- είναι πραγματικό

- $x(t) = 0$ για $t < 0$
- ισχύει
$$\int_{-\infty}^{+\infty} \Re\{X(f)\} e^{j2\pi f t} df = |t| e^{-|t|}$$
με $\Re\{X(f)\}$ το πραγματικό μέρος του μετασχ. Fourier του σήματος $x(t)$.

Βρείτε μια κλειστή έκφραση για το $x(t)$

Απ: $x(t) = 2te^{-t}u(t)$

Άσκηση 5 - Αντίστροφος Μετασχ. Fourier - I

Αν γνωρίζετε ότι για ένα μετασχ. Fourier $X(f)$ ισχύει ότι

$$|X(f)| = 2(u(f+3) - u(f-3)) \tag{3}$$

με $u(\cdot)$ τη βηματική συνάρτηση, και

$$\phi_x(f) = -3\pi f + \pi \tag{4}$$

τότε βρείτε

(α΄) την πολική αναπαράσταση του μετασχ. Fourier $X(f)$

(β΄) το σήμα $x(t)$

(γ΄) τα σημεία μηδενισμού του σήματος $x(t)$

Απ: (β) $x(t) = -12\mathrm{sinc}(9-6t)$, (γ) $t = \frac{9-k}{6},\ k \in \mathrm{Z}$

Άσκηση 6 - Αντίστροφος Μετασχ. Fourier - II

Έστω ο μετασχ. Fourier

$$X(f) = \frac{1}{2 - (2\pi f)^2 + j6\pi f} \tag{5}$$

Προσέξτε ότι

$$2 - (2\pi f)^2 + j6\pi f = 2 + (j2\pi f)^2 + j3(2\pi f) = (1 + j2\pi f)(2 + j2\pi f) \tag{6}$$

και εφαρμόστε ανάπτυγμα σε μερικά κλάσματα[1] για να βρείτε τον αντίστροφο Μετασχ. Fourier (δηλ. το σήμα στο χρόνο $x(t)$) χρησιμοποιώντας την ιδιότητα της γραμμικότητας και γνωστά ζεύγη μετασχηματισμών που γνωρίζετε ήδη από τους Πίνακές σας.

Απ.: $x(t) = (e^{-t} - e^{-2t})u(t)$

[$\star$] Άσκηση 7 - Συμβολικός Υπολογισμός Μετασχ. Fourier στο MATLAB[2]

Έχετε ήδη δει πως μπορούμε να κάνουμε υπολογισμούς στο MATLAB με συμβολικό τρόπο. Ο Μετασχ. Fourier είναι κι αυτός ένα ολοκλήρωμα και μπορεί να πραγματοποιηθεί με χρήση *συμβολικών* υπολογισμών. Δείτε τον παρακάτω κώδικα για το πως αναλύουμε και συνθέτουμε σήματα στο MATLAB με συμβολικούς υπολογισμούς.

(α΄) Ας ξεκινήσουμε με τον ευθύ μετασχηματισμό Fourier.

[1] Θυμηθείτε τη διαδικασία όπως περιγράφεται στο κεφάλαιο Υποβάθρου.

[2] Αυτή η άσκηση απαιτεί αποκλειστικά MATLAB - δεν είναι συμβατή με το Octave.

```
syms t;

% "Proper" FT definition
sympref('FourierParameters', [1, -2*sym(pi)]);

% Signals you already know :-)
x1(t) = dirac(t);             % Delta function
x2(t) = heaviside(t-2);       % Unit step function
x3(t) = sign(t);              % Sign function
x4(t) = rectangularPulse(t);  % Rect() function

% Fourier Transforms
X1 = fourier(x1, 'f');
X2 = fourier(x2, 'f');
X3 = fourier(x3, 'f');
X4 = fourier(x4, 'f');
```

Αν ζητήσετε από το MATLAB να σας τυπώσει τα αποτελέσματα, τότε πρέπει να δείτε τα ακόλουθα:

```
X1 = 1
X2 = exp(-pi*f*4i)*(dirac(f)/2 - 1i/(2*pi*f))
X3 = -1i/(pi*f)
X4 = (cos(pi*f)*1i + sin(pi*f))/(2*pi*f) - (cos(pi*f)*1i - sin(pi*f))/(2*pi*f)
```

Παρατηρήστε πως για τον τετραγωνικό παλμό (X4) το αποτέλεσμα που παίρνουμε δεν είναι αυτό που αναμένουμε θεωρητικά - είναι σίγουρα σωστό, αλλά δεν είναι όσο απλό έχουμε βρει εμείς στη θεωρία. Ας εφαρμόσουμε τη συνάρτηση απλοποίησης `simplify` σε αυτόν:

```
% Some more simplification...
X4s = simplify(X4)
```

παίρνουμε

```
X4s = sin(pi*f)/(pi*f)
```

Παρατηρήστε ότι μετά από την απλοποίηση, το αποτέλεσμα είναι το ίδιο με αυτό που θα βρίσκαμε υπολογίζοντας το μετασχ. Fourier στο χαρτί!

(β΄) Ας εξετάσουμε αν η σύνθεση, δηλ. ο αντίστροφος μετασχ. Fourier μας δίνει τα ίδια αποτελέσματα.

```
syms t;

% "Proper" FT definition
sympref('FourierParameters', [1, -2*sym(pi)]);

% Fourier Transforms
x1i = ifourier(X1, 't');
x2i = ifourier(X2, 't');
x3i = ifourier(X3, 't');
x4i = ifourier(X4, 't');
```

Αν ζητήσετε από το MATLAB να σας τυπώσει τα αποτελέσματα, τότε πρέπει να δείτε τα ακόλουθα:

```
x1i = dirac(t)
x2i = 1/2 - sign(4*pi + 2*pi*t)/2
x3i = sign(t)
x4i = (pi*heaviside((pi - 2*pi*t)/(2*pi)) - pi*heaviside(-(pi + 2*pi*t)/(2*pi)))/pi
```

Παρατηρήστε πως δυο από τις εκφράσεις δεν είναι σύμφωνες με τα αποτελέσματα που αναμένουμε θεωρητικά - είναι σίγουρα σωστές εκφράσεις, αλλά δεν είναι όσο απλές περιμένουμε. Ας εφαρμόσουμε ξανά τη συνάρτηση απλοποίησης `simplify` σε αυτές:

```
% Some more simplification...
x2i = simplify(x2i)
x4i = simplify(x4i)
```

παίρνουμε

```
x2i = 1/2 - sign(t + 2)/2
x4i = heaviside(t + 1/2) - heaviside(t - 1/2)
```

Παρατηρήστε ότι μετά από την απλοποίηση, οι εκφράσεις είναι πλέον πολύ κοντά με αυτές που έχουμε βρεί - ή θα βρίσκαμε - υπολογίζοντας τον αντίστροφο μετασχ. Fourier στο χαρτί!

Υπολογίστε συμβολικά τον ευθύ και τον αντίστροφο μετασχ. Fourier - όπως παραπάνω - των γνωστών σας σημάτων:

I. $x_5(t) = \mathrm{tri}(t)$

II. $x_6(t) = \frac{1}{\pi t}$

III. $x_7(t) = (e^{-2t} - e^{-4t})u(t)$

IV. $x_8(t) = 2\cos(2\pi 400 t)$

Οι συναρτήσεις `triangularPulse, exp, heaviside, cos` **θα σας φανούν χρήσιμες. Όπου χρειάζεται, χρησιμοποιήστε τη συνάρτηση** `simplify` **για να απλοποιήσετε το αποτέλεσμα. Παραδώστε τον κώδικά σας, φροντίζοντας να φαίνονται οι απαντήσεις που παίρνετε - βάλτε τις π.χ. σε σχόλια.**

Άσκηση 8 - Αριθμητικός Υπολογισμός Μετασχ. Fourier στο MATLAB

Συζητήσαμε αρκετά στις διαλέξεις για τον μετασχ. Fourier και βρήκαμε τη μαθηματική του έκφραση για αρκετά γνωστά μας σήματα. Για να υπολογίσουμε το ολοκλήρωμα του μετασχ. Fourier στο MATLAB με αριθμητικό τρόπο[3], θα πρεπει να πάρουμε δείγματα από τον άξονα του χρονου και τον άξονα της συχνότητας, ώστε να κατασκευάσουμε το γινόμενο $x(t)e^{-j2\pi ft}$ και να το ολοκληρώσουμε.

Το MATLAB φυσικά έχει δική του συνάρτηση που υπολογίζει τον μετ. Fourier ενός σήματος, και λέγεται `fft` . Παρ' όλα αυτά, εμείς θα φτιάξουμε τη δική μας, για να έχουμε απόλυτο έλεγχο και γιατι η `fft` απαιτεί κάποιες γνώσεις παραπάνω για να τη χρησιμοποιήσετε.

Θυμηθείτε, στο MATLAB, όλα είναι πίνακες, άρα έχουν διακριτές τιμές. Έχετε δει σε προηγούμενη σειρά ασκήσεων πώς υπολογίζουμε στο MATLAB ένα ολοκλήρωμα. Παρόμοια θα δουλέψουμε και εδώ, μόνο που το ολοκλήρωμά μας δε θα είναι τιμή, αλλά ένας πίνακας $[1 \times L]$, που θα περιέχει μερικές τιμές της συνάρτησης $X(f)$, δηλ. του μετασχ. Fourier που ψάχνουμε.

(α') Έστω ότι θέλουμε να δημιουργήσουμε ένα σήμα διάρκειας 10 δευτερολέπτων. Όπως γνωρίζετε ήδη, ο χρόνος των 10 δευτερολέπτων έχει άπειρες χρονικές στιγμές, οπότε θα πρέπει να διαλέξουμε κάποιες τιμές του σήματος. Έστω ότι θέλουμε να παίρνουμε τιμές ανά 0.01 δευτερόλεπτα. Ας δημιουργήσουμε πρώτα τον άξονα του χρόνου που θα χρησιμοποιηθεί για να πάρουμε τιμές από το σήμα μας. Θα είναι:

[3] Ο συμβολικός υπολογισμός της προηγούμενης άσκησης δεν εφαρμόζεται σε πραγματικά σήματα.

```
Dt = 1/100;      % Sampling step in time
D = 10;          % Signal duration in time
t = 0:Dt:D;      % Time axis (you've seen this before)
```

Έστω ότι θέλουμε να βρούμε το συχνοτικό περιεχόμενο ενός σήματος στο διάστημα $[-10, 10]$, με ίδια "ανάλυση" όπως και στο πεδιο του χρόνου, δηλ. 0.01 Hz. Κατασκευάζουμε τον άξονα των συχνοτήτων ως:

```
Df = 0.01;           % Sampling step in frequency
f = -10:Df:10;       % Frequency axis = [-10,...,10]
```

Θεωρούμε λοιπόν ότι μας ενδιαφέρουν μόνο οι παραπάνω συχνότητες $[-10, 10]$, με ανάλυση `Df`, και ότι σε αυτό το διάστημα θα υπολογίσουμε τον μετασχ. Fourier. Άρα ουσιαστικά θα βλέπουμε τον μετασχ. Fourier μόνο στο διάστημα $[-10, 10]$. Γνωριζοντας τα δείγματα του χρόνου και της συχνότητας μπορούμε να κατασκευάσουμε το λεγόμενο *πίνακα ανάλυσης* του μετασχ. Fourier:

```
M = exp(-j*2*pi*f'*t);
```

Δείτε τη διάστασή του στο MATLAB και προσπαθήστε να καταλάβετε τι περιέχει σε κάθε γραμμή και σε κάθε στήλη του. Αναφέρετε τα συμπεράσματά σας.

(β') Έστω ότι θέλουμε να βρούμε το φάσμα πλάτους του γνωστού σήματος

$$x(t) = e^{-at}u(t), \;\; a > 0 \tag{7}$$

του οποίου γνωρίζουμε από τη θεωρία ότι είναι

$$|X(f)| = \frac{1}{\sqrt{a^2 + 4\pi^2 f^2}} \tag{8}$$

Ας δημιουργήσουμε και σχεδιάσουμε το σημα μας, για $a = 1$:

```
a = 1;
x = exp(-a*t);
figure; plot(t,x);
xlabel('Time (s)');
title('Signal x(t) = exp(-at)');
```

Εδώ παραλείψαμε τη χρήση της βηματικής, αφού ο άξονας του χρόνου που δημιουργήσαμε πριν έχει θετικές τιμες.

Για να υπολογίσουμε τον μετασχ. Fourier, θα δουλέψουμε όπως για το ολοκλήρωμα Riemann σε προηγούμενη σειρά ασκήσεων, δηλ. θα δημιουργήσουμε την προσέγγιση του μετασχ. Fourier ως

$$X(f) = \lim_{\Delta t_i \to 0} \Delta t_i \sum_{i=-\infty}^{+\infty} \left(x(\Delta t_i) e^{-j2\pi f \Delta t_i} \right) \tag{9}$$

και όπως έχουμε δει σε προηγούμενες σειρές ασκήσεων, αυτό στο MATLAB υλοποιείται ως

```
X = Dt*x*exp(-j*2*pi*f'*t).';          % Prosoxh sto .' !
```

που μπορεί πιο σύντομα να γραφεί ως

```
X = Dt*x*M.';                          % Prosoxh sto .' !
```

Ο τελεστής `.'` υπολογίζει τον *ανάστροφο* ενός πίνακα (ενώ ο τελεστής `'` υπολογίζει τον *συζυγή ανάστροφο ή ερμητιανό* ενός πίνακα, και καλό θα ήταν να το αποφεύγετε). Πολλές φορές χρειάζεται αυτός ο τελεστής για να συμφωνούν οι διαστάσεις των πινάκων που εμπλέκονται ως γινόμενα.

(γ') Ας συγκρίνουμε το φάσμα πλάτους του παραπάνω με το θεωρητικό φάσμα πλάτους που ξέρουμε.

```
Xtheoretic = 1./(a + j*2*pi*f);
plot(f, abs(X)); grid;
hold on; plot(f, abs(Xtheoretic), 'r--');
hold off;
```

Μοιάζουν τα δυο φάσματα;

(δ') Ας προσπαθήσουμε τώρα να συνθέσουμε το σήμα στο χρόνο μέσω του αντίστροφου μετασχ. Fourier, δηλ. ας προσπαθήσουμε να βρούμε το σήμα στο χρόνο, $x(t)$! Ο μετασχ. Fourier που έχουμε βρει ορίζεται μόνο στο διάστημα $[-10, 10]$, αρα σίγουρα θα έχουμε κάποιο σφάλμα στον υπολογισμό του $x(t)$. Για να δούμε όμως...

Γνωριζοντας τα δείγματα του χρόνου και της συχνότητας μπορούμε να κατασκευάσουμε το λεγόμενο *πίνακα σύνθεσης* του μετασχ. Fourier:

```
Minv = exp(j*2*pi*t'*f);
```

Δείτε τη διάστασή του στο MATLAB και προσπαθήστε να καταλάβετε τι περιέχει σε κάθε γραμμή και σε κάθε στήλη του. Αναφέρετε τα συμπεράσματά σας.

(ε') Για να υπολογίσουμε τον αντίστροφο μετασχ. Fourier, θα δουλέψουμε όπως για το ολοκλήρωμα Riemann προηγουμένως, με μικρές διαφοροποιήσεις, δηλ. θα δημιουργήσουμε την προσέγγιση του αντίστροφου μετασχ. Fourier ως

$$x(t) = \lim_{\Delta f_i \to 0} \Delta f_i \sum_{i=-\infty}^{+\infty} \left(X(\Delta f_i) e^{j2\pi \Delta f_i t} \right) \tag{10}$$

Η σύνθεση του σήματος στο χρόνο γίνεται ως

```
x_est = Df*X*Minv;
```

(ϛ') Λόγω αριθμητικών σφαλμάτων, θα δείτε ότι το σήμα που λαμβάνετε είναι μιγαδικό (ενώ προφανώς δεν πρέπει να είναι). Επιβεβαιώστε το, ελέγχοντας τις τιμές του φανταστικού μέρους (συνάρτηση `imag`) του `x_est`. Ουσιαστικά λοιπόν, μόνο το πραγματικό μέρος έχει σημασία.

Τυπώστε στον ίδιο άξονα το γράφημα του πραγματικού μέρους (συνάρτηση `real`) και το αρχικό $x(t)$. Είναι τα ίδια; Αν όχι, γιατί; Τυπώστε επίσης και το φανταστικό μέρος - ξεχωριστά. Τι τάξη μεγέθους είναι το αριθμητικό σφάλμα που παράγει το MATLAB;

Παραδωστε κώδικα και γραφήματα για όλα τα παραπάνω ερωτήματα, και απαντήστε και σε όσα θεωρητικά ερωτήματα έχουν επισημανθεί παραπάνω.

Άσκηση 9 - Μετασχηματισμός Fourier και Παθολογία Φωνής

Ο Μετασχ. Fourier είναι ένα εξαιρετικά χρήσιμο εργαλείο σε πολλούς τομείς της Επεξεργασίας Σήματος. Ένας τομέας είναι η ανίχνευση παθολογίας φωνής.

(α') Ηχογραφήστε τη φωνή σας όταν εκφέρετε το φώνημα /α/, σταθερά, για περιπου 3 δευτερόλεπτα. Φροντίστε να βρίσκεστε σε ήσυχο περιβάλλον, και προσπαθήστε να χρησιμοποιήσετε κανονικό μικρόφωνο (όχι τα ενσωματωμένα των laptops). Χρησιμοποιήστε ένα προγραμμα ηχογράφησης της επιλογής σας (π.χ. το δωρεάν πρόγραμμα Wavesurfer ή το Audacity) και φροντίστε η ηχογράφηση να είναι μονοφωνική (δηλ. όχι στέρεο - δικαναλική) και να γίνει σε συχνότητα δειγματοληψίας 16000 Hz, και ακρίβεια αποθήκευσης 16 bits σε μορφη .WAV. Σας δίνεται και ένα ενδεικτικό σήμα στο site του μαθήματος.

(β') Χρησιμοποιήστε τον κώδικα ανάλυσης μετασχ. Fourier που σας δόθηκε στην προηγούμενη άσκηση για να αναλύσετε το σήμα όπως περιγράφεται παρακάτω:

- Επιλέξτε ένα τμήμα (ή, όπως λέμε στην ορολογία της Επεξεργασίας Σήματος, ένα *παράθυρο*) φωνής, με διάρκεια περίπου 50 ms, κατά προτίμηση από τη μέση περιπου της ηχογράφησης. Στο MATLAB, αυτό μπορεί να γίνει ως:

```
[s, fs] = wavread('myvoice.wav');  % Load the recorded speech signal
start = 1                          % Assume our speech segment starts
                                   % from time = 1 sec...
finish = 1.05 ;                    % ...and ends after 50 ms
start_s = round(start*fs);         % Convert time in samples
finish_s = round(finish*fs);       % -------- " ------------
segment = s(start_s:finish_s);     % Chop the desired speech segment
plot([start_s:finish_s]./fs, segment); % Visualization! :)
```

- Στη μεταβλητή `segment` έχετε ένα τμήμα φωνής σας διάρκειας 50 ms. Χρησιμοποιήστε κώδικά ανάλυσης Fourier για να αναλύσετε το σήμα σας στην περιοχή συχνοτήτων από 2000 ως 4000 Hz. Βρειτε το μετασχ. Fourier και απεικονίστε γραφικά το φάσμα πλάτους, με χρήση των εντολών `abs, plot`, όπως στην προηγούμενη άσκηση. Στον κώδικα της ανάλυσης, χρησιμοποιήστε μικρό βήμα στη συχνότητα, της τάξης του 1 Hz, δηλ.

```
Df = 1;
f = 2000:Df:4000;
```

- Αν το φάσμα πλάτους που θα δείτε, παρουσιάσει μια συμμετρία ως προς τη συχνότητα 3000 Hz, τότε υπάρχει μια πιθανότητα $20 - 30\%$ να αναπτύξετε ασθένεια στο φάρυγγά σας τα επόμενα 5 χρόνια. Προσέξτε λοιπόν να κάνετε σωστά την ανάλυση! :-)

Παραδωστε κώδικα που υπολογίζει το μετασχ. Fourier ενός τυχαίου παραθύρου φωνής, και εμφανίζει το φάσμα πλάτους του. Παραδώστε τυπωμένο στο χαρτί ένα γράφημα αυτού του φάσματος πλάτους. Γράψτε αν παρατηρείτε κάτι.

[$\star$] Άσκηση 10 - Επέκταση της προηγούμενης άσκησης

Ίσως να σκεφτήκατε ότι το να πάρουμε ένα τυχαίο κομμάτι απ' το σήμα φωνής μας και αφού το αναλύσουμε, να βγάλουμε απόφαση για κάτι τόσο σοβαρό όπως μια πιθανή παθολογία, είναι λίγο ριψοκίνδυνο και επιπόλαιο. Κάτι πιο ασφαλές θα ήταν το εξής:

(α') Χωρίστε όλο το σήμα σε παράθυρα διάρκειας 50 ms, με μια επικάλυψη γειτονικών παραθύρων της τάξης του 50%, δηλ. "προχωράτε" το παράθυρό σας πάνω στο σήμα της φωνής κάθε 25 ms, ωστε τα παράθυρά σας να επικαλύπτονται κατά μισό παράθυρο. Αν σας φαίνεται δύσκολο, μπορείτε να μη χρησιμοποιήσετε επικάλυψη. Αυτό μπορείτε να το κάνετε με χρήση βρόχων επανάληψης όπως τους γνωρίζετε από τη C (for, while) - δε διαφέρουν πολύ. Γράψτε help for, help while για να δείτε πως συντάσσονται. Σκεφτείτε οτι απλά πρέπει να διατρέχετε ένα πίνακα-γραμμή (που είναι το σήμα σας) ανά κάποιο αριθμό στοιχείων.

(β΄) Υπολογίστε το μετασχ. Fourier για τις συχνότητες $2000 - 4000$ Hz και βρείτε το φάσμα πλάτους του κάθε παραθύρου. Αποθηκεύστε το φάσμα πλάτους κάθε παραθύρου σε μια γραμμή ενός πίνακα X. Αυτό μπορεί να γίνει ως εξής:

```
% Let N be the whole signal duration in samples
for i = 1:N
    % Let seg be the variable holding
    % our current speech segment
    MF = dt*seg*M.';            % Compute Fourier Transform
    Fasma_platous = abs(MF);    % Compute Magnitude Spectrum
    Y(i, :) = Fasma_platous;    % Store it into the i-th row
                                % of matrix Y
end
```

(γ΄) Να υπολογίσετε το "μέσο φάσμα πλάτους ", δηλ. μια μέση τιμή όλων των φασμάτων πλάτους που έχετε βρει, έτσι ώστε στο τελος να έχουμε μόνο ένα φάσμα πλάτους, και να αποφασίσετε για την παθολογία βασει αυτού. Χρήσιμη θα σας φανεί η εντολή `mean` του MATLAB.

(δ΄) Ακολουθώντας μια τέτοια διαδικασία έχουμε πιο εύρωστα, με τη στατιστική έννοια, συμπεράσματα.

Παραδωστε κώδικα που υπολογίζει και εμφανίζει το μέσο φάσμα πλάτους του μετασχ. Fourier. Παραδώστε τυπωμένο στο χαρτί αυτό το μέσο φάσμα πλάτους. Γράψτε αν παρατηρείτε κάτι.

Άσκηση 11 - Μετασχηματισμός Fourier κι αφαίρεση θορύβου

Στις πρώτες γραμμές των σημειώσεών σας περί ανάλυσης στο χώρο της συχνότητας, συζητήσατε ένα πραγματικό πρόβλημα αποθορυβοποίησης φωνής, ως εφαρμογή-κίνητρο της Ανάλυσης Fourier. Ήρθε η ώρα να το δείτε να γίνεται στην πράξη! :)

Σας δίνεται στο site του μαθήματος ένα σήμα μουσικής sample-noise.wav. Πρόκειται για ένα γνωστό τραγούδι "μολυσμένο" με ένα ισχυρό σήμα ημιτόνου σε κάποια υψηλή, σταθερή, συχνότητα μεταξύ 1000 και 3000 Hz. Σκοπός της άσκησης είναι να αναλύσετε το σήμα και να αφαιρέσετε το θόρυβο. Ακολουθήστε τα παρακάτω βήματα.

(α΄) Αρχικά, ακούστε το σήμα.

```
[s, fs] = wavread('sample-noise.wav');  % Load the speech signal
soundsc(s, fs);                         % Listen to it!
t = 0:1/fs:(length(s)-1)/fs;            % Time axis in seconds
plot(t, s);                             % Visualize!
```

(β΄) Παρατηρήστε - και ακούστε - ότι η συνιστώσα του ημιτόνου είναι ισχυρή, και εύκολα διακρίνεται μέσα στον ήχο της ηχογράφησης. Γνωρίζετε όμως ότι λόγω της ισχύος της, θα πρέπει να "ξεχωρίζει" σχετικά στο φάσμα πλάτους του σήματος απο το υπόλοιπο σήμα. Επίσης, επειδή είναι σταθερής συχνότητας, μπορούμε να την εντοπίσουμε σε οποιοδήποτε σημείο (παράθυρο) του σήματος κι αν επιλέξουμε.

(γ΄) Διαλέξτε ένα τυχαίο παραθυρο σήματος, διάρκειας 30 ms και αναλύστε το στις παραπάνω συχνότητες ($1000 - 3000$ Hz) με τον μετασχ. Fourier, χρησιμοποιώντας φυσικά το φάσμα πλάτους[4]. Προσπαθήστε να εντοπίσετε το ημίτονο. Σκεφτείτε ότι ο μετασχ. Fourier του ημιτόνου πλησιάζει τη συνάρτηση Δέλτα που έχει γίνει συνέλιξη με το μετασχ. Fourier του παραθύρου σας (όπως ακριβώς είδατε στην Άσκηση 1).

[4] Σε σχέση με τις προηγούμενες δυο ασκήσεις, στην πραγματική ρέουσα ομιλία και στον ήχο, το σήμα αλλάζει πιο γρηγορα απ' ότι όταν λέμε ένα απλό /a/. Έτσι, χρησιμοποιούμε μικρότερο παράθυρο ανάλυσης για να είμαστε σχετικά ασφαλείς ότι το περιεχόμενό του δεν αλλάζει σημαντικά.

Πρακτικά, θα περιμένετε να δείτε κάποιο ισχυρό peak (κορυφή) στο φάσμα πλάτους. Όμως επειδη το περιεχόμενο του σήματος ειναι μουσική και φωνή, το φάσμα πλάτους θα περιέχει και άλλες συχνότητες. Οπότε η αναγνώριση του peak από ένα και μόνο παράθυρο δε θα είναι εύκολη, εκτός αν είστε τυχεροί/ες. :-) Στην ανάλυσή σας, χρησιμοποιήστε ενδεικτικά τον παρακάτω κώδικα:

```
T = 30;                             % Window duration of 30ms
Ts = T*10^(-3)*fs;                  % Window duration of 30ms in samples
x = s(45213:45213 + Ts);            % Choose randomly a 30 ms segment
                                    % from the speech signal that starts
                                    % from the sample number 45213
                                    % (also randomly chosen)
plot(x);                            % Visualize!
Dt = 1/fs;                          % Analysis step in time
Df = 1;                             % Analysis step in frequency
f = 1000:Df:3000;                   % Frequency axis that is of our
                                    % interest
t = 0:(1/fs):(Ts/fs);               % Time axis of 30ms duration
M = exp(-j*2*pi*f'*t);              % Matrix M
x = reshape(x, 1, length(x));       % Make sure x is a row vector

X = Dt*x*M.';                       % Fourier Transform
figure; plot(f, abs(X));            % Search for a strong peak in
                                    % [1000, 3000]
```

(δ') Επιλέξτε διάφορα παράθυρα μέσα στο σήμα $(4-5)$, όλα ίδιας διάρκειας, μέχρι να εντοπίσετε τη συχνότητα του ημιτόνου με κάποια βεβαιότητα. Προς διευκόλυνσή σας, δίνεται οτι η συχνότητα είναι *ακέραιος αριθμός*, πολλαπλάσιος του 100, στο διάστημα $[1000, 3000]$ Hz. Σε κάθε plot που κάνετε, στο πάνω μέρος υπάρχουν κάποια εικονίδια. Ένα από αυτά, ο Data Cursor, σας δίνει τις συντεταγμένες του σημείου του σήματος που θα κάνετε κλικ. Έτσι, μπορειτε να βρίσκετε εύκολα τη συχνότητα ενός σημείου στο φάσμα σας. Παραδωστε μερικά plots από τα παράθυρα που διαλέξατε, τόσο στο χρόνο όσο και στο φάσμα πλάτους.

(ε') Σας δίνουμε επιπλέον ότι το ισχυρό αυτό ημίτονο έχει πλάτος $A = 0.01$ και αρχική φάση $\phi = 0$, δηλ. είναι της μορφής

$$n(t) = A\cos(2\pi f_0 t)$$

Σε προηγούμενες ασκήσεις, έχετε δει πώς δημιουργούμε ενα απλό ημίτονο. Δημιουργήστε ένα ημίτονο στο MATLAB με πλάτος και φάση που σας δινεται παραπάνω, και με συχνότητα αυτήν που βρήκατε από την ανάλυσή σας στο προηγούμενο ερώτημα. Φροντίστε να έχει ίδια διάρκεια με ολόκληρο το σήμα s του τραγουδιού. Για να βρείτε τη διάρκεια αυτή, χρησιμοποιήστε τη συνάρτηση `length` του MATLAB. Για παραδειγμα, αν θέλετε να φτιάξετε ενα ημίτονο διάρκειας 100 δειγμάτων, δηλ. $1/160 = 0.00625$ sec (με συχνότητα δειγματοληψίας 16000 Hz), πλάτους 1 και συχνότητας 200 Hz, θα κάνετε το εξής:

```
A = 1;
f0 = 200;
fs = 16000;
n = A*cos(2*pi*f0*[0:99]/fs);      % Example
```

(ϛ') Αφαιρέστε το σήμα ημιτόνου που φτιάξατε παραπάνω από το σήμα της ηχογράφησης s, απλώς αφαιρώντας μεταξύ τους το διάνυσμα s και το διάνυσμα ημιτόνου που μόλις φτιάξατε, όπως παρακάτω. Ακούστε το αποτέλεσμα. Θα πρέπει να ακούγεται πλέον καθαρό το σήμα. :-)

```
clean_sig = s - n.';                % s = signal, n = cosine
soundsc(clean_sig, fs);             % Listen!
```

Αν η πράξη σας βγάζει σφάλμα διάστασης, αφαιρέστε το `.'`.

(ζ) Το παραπάνω παράδειγμα ήταν πολύ "εκπαιδευτικό" :-). Στην πράξη, το ημίτονο μπορεί να μην έχει σταθερό πλάτος ή μηδενική φάση, ή ακόμα κι αν έχει, δεν μπορούμε να γνωρίζουμε εκ των προτέρων τις τιμές τους. Έτσι, μια μέθοδος όπως η παραπάνω, στο πεδίο του χρόνου δηλαδή, δε θα δουλέψει. Συνήθως χρησιμοποιούμε μεθόδους στο χώρο της συχνότητας για να αφαιρέσουμε τον ενοχλητικό θόρυβο, εφαρμόζοντας τα λεγόμενα *notch* φίλτρα, τα οποία είναι συστήματα που μηδενίζουν το πλάτος μιας συγκεκριμένης συχνότητας από ένα σήμα που δέχονται ως είσοδο. Η εφαρμογή μιας τέτοιας τεχνικής ξεφεύγει από τα πλαίσια του μαθήματος[5], αν και ίσως προς το τέλος του μαθήματος να μπορέσετε να υλοποιήσετε ένα τέτοιο απλό φίλτρο! :)

Παραδωστε όσα plots ζητούνται στη διάρκεια της εκφώνησης, και κώδικα MATLAB που καθαρίζει το ηχητικό σήμα από το θόρυβο.

[5]Είναι αντικείμενο του μαθήματος ΗΥ370-Ψηφιακή Επεξεργασία Σήματος.