

<p style="text-align: center;">TP de Traitement du signal. Séance 1 : Premiers contacts avec MatLab et les TF IN41 : Traitement du signal</p>

<p style="text-align: center;">Introduction</p>

a) Contexte

Cet énoncé constitue le premier volet des séances de TP de l'unité de valeurs IN41 (traitement du signal). Les TP de IN41 sont orientés comme des « mini » projets et par conséquent seront relativement longs. La vocation de cette première séance est de permettre à chaque élève de prendre en main le logiciel *MatLab* (connaissance de l'environnement, commandes usuelles, programmation de base) et de réaliser ses premiers programmes qui constitueront les briques de bases nécessaires à la réalisation des TP suivant. Le premier TP sera assez théorique mais les bases des signaux et du traitements doit être traité avant de s'attaquer à des problèmes informatiques.

Le TP est en conséquence structuré comme suit :

1. Présentation de *MatLab* et du cadre de développement des applications du TP.
2. Réalisation des premiers programmes.

b) Objectifs pratiques

Les objectifs sont triples : rafraîchir et consolider les connaissances acquises en cours, découvrir et assimiler les commandes usuelles de *MatLab* et le fonctionnement de *MatLab Help*, et mettre en application l'ensemble de ces savoirs et savoir-faire via la réalisation de premiers programmes en vue des futures TP.

En conséquence, il sera demandé aux élèves, pour chaque séance :

- de rédiger en temps réel un court rapport illustré retraçant le TP en suivant la trame des questions posées dans l'énoncé. La clarté des graphiques, la pertinence des remarques, et la correction linguistique seront prises en compte dans l'évaluation. La concision des réponses sera d'autant plus appréciée qu'elle ira de pair avec leur précision,
- de remplir au fil du TP les pages qui ont été laissées vides sur le présent document.
- d'écrire soigneusement les programmes demandés sous formes de M-File commentées,
- et enfin, le rapport ainsi que l'ensemble des codes à l'adresse laurent.moalic@utbm.fr avec pour objet « IN41_TP'n'_'votrenom'_'prenom' » dans les 15 jours suivant.

1. Présentation de MatLab et du cadre de développement des applications du TP

Principales caractéristiques de MatLab 7 :

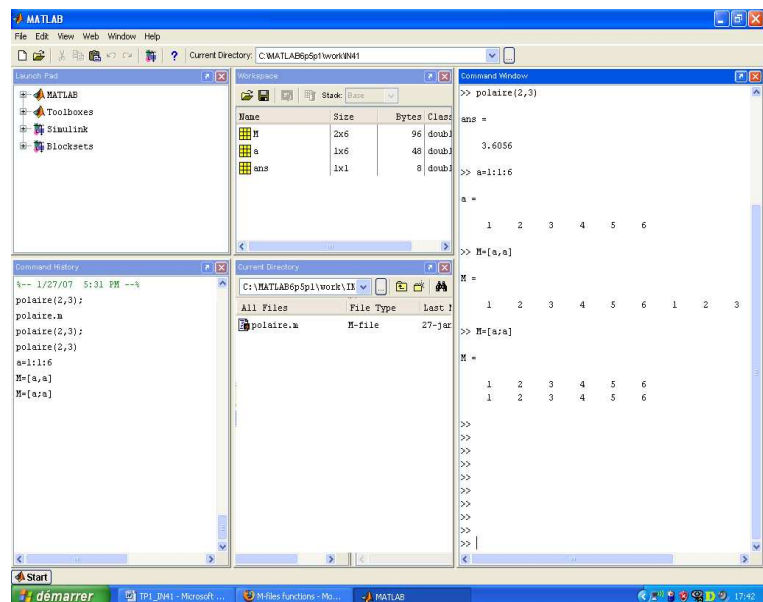
MatLab est un langage mathématique interprété et un environnement de développement interactif. Il fait appel à des fonctions et routines qui ont été développées dans des langages de plus bas niveau et sont interprétées par le logiciel. C'est pourquoi nous ne devons pas compiler les programmes réalisés.

Calcul matriciel et non calcul formel : inutile de demander à MatLab de calculer une dérivée ! MatLab est un outil performant de calcul numérique, qui permet de résoudre des problèmes d'analyse matricielle, d'approximation de solutions d'équations différentielles, de statistiques et de fouille de données, de traitement du signal numérique, ou encore d'optimisation numérique. Pour des calculs mathématiques formels (comme la résolution exacte d'ED, le calcul d'intégrales, etc...), l'utilisateur devra se reporter à d'autres logiciels...ou encore au services d'un mathématicien !

Principales fenêtres de MatLab:

- Command window:
- Command History:
- Current directory:
- Workspace:
- Help:
- Profile:

Figure 1: vue des 6 fenêtres de MatLab
Pour réunir les 6 fenêtres en une avec onglets de sélection : Menu Desktop, commande « Desktop Layout », option « All Tabbed ».



Insistons sur Help : le menu et la commande help : l'aide de *MatLab* est munie d'un moteur de recherche très pratique, et s'avère bien souvent être un recours précieux pour se sortir des mauvaises passes, mais aussi pour découvrir de nouvelles fonctionnalités. Une utilisation intelligente de cet outil peut permettre de renforcer très rapidement ses connaissances en *MatLab* et mener à la réalisation de programmes au-delà de ses espérances.

Remarques : Penser aussi à utiliser la commande help suivie du nom de la fonction à étudier dans la fenêtre de commande (plus rapide, mais nécessite de savoir à l'avance ce que l'on recherche !).

M-files : les m-files sont les fichiers dans lesquels sont écrits les programmes et les fonctions. Ils portent l'extension « .m », et sont analogues (par exemple) aux fichiers « .c » du langage C. Toute fonction devra être déclarée à la première ligne du fichier en respectant la typologie suivante : « 'function' » suivi du vecteur de sortie « [sortie1, sortie2] » « '=' » « nomdelafonction » « (entrée1, entrée2, entrée3) ».

Exemple : La fonction qui renvoie le module et l'argument d'un nombre complexe $x+iy$.

Structures : Les principales structures à maîtriser avec Matlab sont les « matrices » ou « vecteurs ».

Exemples : Créer un vecteur $a = 123456$, puis une matrice $M=[a,a]$.

Les commandes qu'il faut absolument connaître avant de se lancer :

Tirages aléatoires :
Boucles :
a) For :
b) While :
Taille d'une matrice :

Représentations graphiques :

a) plot : $A=1:1:10$; $B=A.*2$; plot(A,B);
b) Options graphiques (à chercher soi-même) :
Nouvelle figure ou superposition de graphiques:
Couleurs :
Formes des graphes :
Réglage de la taille et du nom des axes :
Légendes :
Titres :

Remarque : bien penser au « ; » en fin de ligne de commande ;

2. Exercices

a) Soit la fonction $x(t) = \sin(2\pi \cdot 50 \cdot t) + \sin(2\pi \cdot 120 \cdot t)$ définie sur $[0, 0.6]$. On définit le vecteur des temps de la manière suivante : $t = 0:0.001:0.6$; Représenter x en fonction de t en apportant tout le soin nécessaire à la présentation du graphique (axes, titre, etc....).

b) Réaliser les signaux suivants :

- Une impulsion unité (Dirac) défini par $f(n) = 1$ si $n = 0$ et 0 sinon.
- Un échelon unité par défini par $f(n) = 1$ si $n \geq 0$ et 0 sinon.
- La fonction signe.
- La fonction porte ou fonction rectangle. (Essayer d'utiliser les fonctions ones et zeros)

c) On veut analyser les propriétés de certaines séquences élémentaires. Générer un signal sinusoïdal sur 10000 points et le visualiser sur 200 échantillons. Puis déterminer son minimum, maximum, moyenne, sa médiane et sa dispersion.

d) Visualiser les parties du code sous Matlab et répondre aux questions.

Remarques :

```
clear          % effacer mémoire
close all      % fermer toutes les fenêtres (figures)
```

- Révision de fonction $\exp(j \cdot 2 \cdot n \cdot \text{freq})$; ici j est égal à $i = \sqrt{-1}$

```
Np=128;
n=linspace(0,Np-1,Np); % creation d'un vecteur qui varie lineairement
freq=0.05;
expo=exp(j*n*2*pi*freq); % = sin(n*2*freq) + i * cos(n*2*freq)
fs=1;
```

```
figure;
subplot(1,3,1);
plot(expo);          % l'axis x est les reels et y les imaginaires
axis([-1 1 -1 1]);
title('exp(j*n*2*freq)');
```

```
subplot(1,3,2);
plot(real(expo));
title('real exp(j*n*2*freq)');
```

```
subplot(1,3,3);
plot(imag(expo));
title('imag exp(j*n*2*freq)');
```

Expliquer ce que représentent les trois figures ?

Que représente le décalage entre la partie réel et la partie imaginaire ?

- TF de fonctions sinusoïdales

```
freq=0.2;
fx=linspace(-0.5,0.5,Np);
%Etude du cosinus et du cosinus^2
cosinus=cos(2*pi*n*freq/fs);
fcosinus=fft(cosinus);
cosinus2=cos(2*pi*n*freq/fs).*cos(2*pi*n*freq/fs);
fcosinus2=fft(cosinus2);

%On trace les resultats
subplot(2,2,1)      % Definit le nombre de fenetres verticales/horizontales
plot(cosinus)        % Trace la courbe "en continue"
title('cosinus');
subplot(2,2,2)
plot(fx,fftshift(real(fcosinus)));
title('fft cosinus');
axis([-0.5 0.5 0 60])

subplot(2,2,3)
plot(cosinus2)
title('cosinus2');
subplot(2,2,4)
plot(fx,fftshift(real(fcosinus2)));
title('fft cosinus2');
axis([-0.5 0.5 0 60])
```

Que fait la fonction FFT ?

Que fait la fonction fftshift ?

Quel rapport y a-t-il entre le fft cosinus et fft cosinus2 ?

Faites varier la frequence (0.2,0.4,0.49), Qu'observez vous ?

- construction d'une Gaussienne et sa transformee de Fourier

```
Np=128;
n=linspace(0,Np-1,Np); %Construction de la Gaussienne
sigma=2;
nzero=Np/2-1;
x=n-nzero;
x2sig2=(x.*x)/sigma^2;
gauss=(exp(-x2sig2));
fgauss=fftshift(fft(fftshift(gauss)));

% Visualisez la Gaussienne et sa FFT
figure
subplot(2,1,1)
plot(gauss)
title('gauss');
subplot(2,1,2)
plot(real(fgauss))
title('fgauss');
```

Quelle est la forme de la FFT ?

Faites varier sigma. Qu'observez-vous ?
Conclusions sur la TF d'une gaussienne ?

e) Soit un signal carré quelconque

Calculer la TF de $\text{rec}(t)$ et tracer la.
En déduire la TF de tout signal rectangulaire de durée T et d'amplitude A : $A \cdot \text{rec}(t/T)$
Tracer cette TF.

f) On définit la convolution de deux signaux $x(t)$ et $y(t)$ par :

$$x * y(t) = \int_{-\infty}^{+\infty} x(u) y(t-u) du$$

Définir une fonction permettant de calculer le produit de convolution
Montrer que la TF de $x * y(t)$ égale le produit des TF de x et y.

g) Convolution de deux portes

Tracer une fonction carrée, et le produit de convolution de cette fonction.
Pourquoi obtient-on des triangles ?

Bonus :

Lien intéressant pour tester visualiser les séries de Fourier :
<http://users.ece.gatech.edu/mccllella/matlabGUIs/ZipFiles/fseriesdemo-v120.zip>
Faire un petit bilan de l'application