



HAL
open science

Livret de Travaux Dirigés en Analyse des Signaux et des Images

David Boulinguez, Christophe Krzeminski

► **To cite this version:**

David Boulinguez, Christophe Krzeminski. Livret de Travaux Dirigés en Analyse des Signaux et des Images. École d'ingénieur. France. 2020. hal-03281332v2

HAL Id: hal-03281332

<https://hal.science/hal-03281332v2>

Submitted on 4 Sep 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

*Livret de Travaux Dirigés
en Analyse des Signaux et des Images*

Rédacteurs: David Boulinguez et Christophe Krzeminski

Année Scolaire Académique 2020-2021 CSI3

Sommaire

I	Introduction	v
II	Tutoriel	1
A	Produit de convolution	3
A.1	Produit de convolution de deux portes différentes	3
A.2	Réponse d'un système à un triangle	4
B	Propriétés de la Transformée de Fourier	7
B.1	Représentation temporelle d'un signal et propriétés de la transformée de Fourier	8
B.2	Calcul de la transformée de Fourier d'un signal évolué	9
B.3	Filtrage et transformée de Fourier d'un produit de convolution	10
C	Calcul de Transformée de Fourier	11
C.1	Phase et transformée de Fourier d'un signal sinusoidal	11
C.2	Linéarisation et spectre d'un signal cosinusoidal	12
C.3	Echantillonnage, spectre et recouvrement	13
D	Numérisation d'un signal	15
D.1	Echantillonnage d'un signal composé de deux sinusoides	15
D.2	Echantillonnage et filtrage	16
D.3	Conception d'un convertisseur analogique numérique	17
E	Transformée de Fourier Discrète d'un Signal	19
E.1	Représentation et analyse fréquentielle d'un signal	19
E.2	Notion de résolution fréquentielle et finesse d'analyse	20
E.3	Périodicité et analyse spectrale par transformée de Fourier Discrète	21
F	Transformée de Fourier Discrète	23
F.1	Spectre et calcul d'une transformée de Fourier d'ordre 3	23
F.2	Transformée de Fourier Discrète d'une séquence cosinusoidale	24
F.3	Spectre de transformée de Fourier Discrète et Analogique	25
G	Analyse et Applications des Signaux Aléatoires	27
G.1	Chaîne radar et probabilité de détection	28
III	Praticiel	31
H	Analyse de Signaux par Transformée de Fourier Discrète	33
H.1	Objectifs	33
H.2	Requis préalable	33

H.3	Rappels sur les notions de Transformée discrète	33
A)	Un peu de théorie	33
B)	Fenêtrage et TFD d'un signal périodique	35
a)	Fenêtrage	36
b)	Conditions d'échantillonnage et cas d'un signal périodique	36
H.4	Simulation numérique Matlab ou Octave	37
A)	Transformée de Fourier discrète d'un signal sinusoidal	37
B)	Modulation en fréquence d'un signal et repliement du spectre	37
I	Compression d'Image et Transformée en Cosinus Discret	39
I.1	Découverte de la transformée de Fourier en Cosinus Discret (2D)	40
I.2	Application à la compression d'une image	41
J	Etude et Analyse de Signaux Aléatoires	43
J.1	Séries aléatoires et statistiques simples	43
J.2	Analyse d'un bruit blanc gaussien	44
K	Introduction au langage Octave ou Scilab	47
K.1	Introduction	47
K.2	Notions de base	48
A)	Aide du logiciel Matlab/Octave/Scilab	48
B)	Environnement de travail	48
C)	Fonctions mathématiques	49
a)	Fonction standard mathématiques	49
D)	Définition de structure de données simple: vectorielle ou matricielle	50
E)	Usage d'une calculette	52
a)	Fonction mathématiques usuelles	52
b)	Calcul matriciel	53
F)	Notion d'opérateur	53
G)	Fonction statistique	53
a)	Génération de séquence aléatoire discrète	53
b)	Statistiques	54
H)	Structure de contrôle	54
K.3	Script de commande ou fonction&procédures	55
A)	Définition de script de commande	55
B)	Déclaration de fonctions ou procédures	55
C)	Structure d'affichage	55
a)	Notion de figure	55
b)	Différents tracés	56
D)	Boîte à outil de traitement du signal	56

Partie I

Introduction

Avant-propos

Ce livret est constitué des différents exercices de travaux dirigés en traitement du signal et des images au programme de la première année du cycle d'ingénieur de l'ISEN située à Lille.

En terme de cadre pour les connaissances requises, des bases mathématiques rudimentaires du cycle préparatoire les fonctions trigonométriques simples, la notion de distribution, quelques rudiments au niveau de l'espace et calcul complexe et le calcul intégral sont nécessaires. Toutefois, l'objectif n'est pas de faire des mathématiques mais d'illustrer les différentes notions en utilisant les outils mathématiques indispensables.

En terme de méthode de préparation, il est probable que l'usage d'une feuille de papier et d'un crayon en étant déconnecté de sollicitation digitale *In-fine* est la plus simple et probablement la meilleure méthode possible.

Il est recommandé de chercher à faire ces exercices de manière autonome, à son rythme. L'écrit et le papier demeure probablement le meilleur moyen de transmettre le savoir de manière *pérenne*. Par contre au niveau des explications, il ne faut pas craindre à solliciter de l'aide ou à communiquer oralement pour les points qui pourraient éventuellement bloquants.

Ce document a été écrit, dans le contexte de la pandémie de la Sars-Cov2, afin de tenter d'assurer une continuité d'enseignement.

Nomenclature

CAN *Convertisseur analogique/numérique*

Echantillonnage

TF *Transformée de Fourier*

TFD *Transformée de Fourier Discrète*

Q *Quantification*

SL *Système Linéaire*

Acronyme en l'anglais

ADC *Analog Digital Converter*

S *Sampling*

DFT *Discrete Fourier Transform*

FT *Fourier Transform*

LS *Linear System*

Q *Quantization*

Partie II
Tutoriel

Chapitre A

Produit de convolution

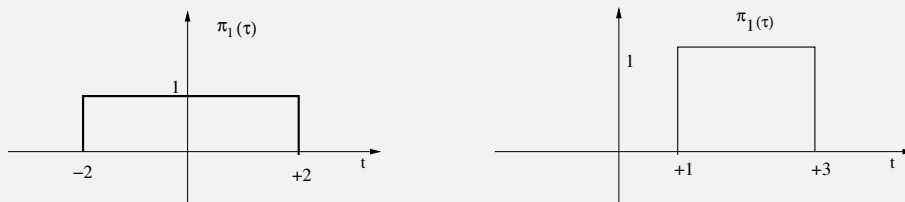
Objectifs

- ☒ Se remettre en mémoire les principes permettant de calculer un produit de convolution de deux fonctions simples et classiques en traitement du signal.

A.1 Produit de convolution de deux portes différentes

Enoncé:

Deux signaux $\pi_1(t)$ et $\pi_2(t)$ sont considérés ci-dessous:



On appelle $h(\tau)$, le produit de convolution de ces deux signaux.

- ✎ Ecrire la définition du produit de convolution sous forme intégrale.
- ✎ Calculer complètement le produit de convolution $h(\tau)$ en distinguant les différents cas.
- ✎ Représenter $h(\tau)$ le résultat du produit de convolution.

Aide

- Savoir: pour la résolution de l'exercice, il est nécessaire de connaître la définition mathématique du produit de convolution.
- Compréhension: il est important de distinguer les différents cas des produits de convolution $\pi_1(t)$ et $\pi_2(t)$ en fonction de la variable τ .
- Démarche: de manière à résoudre l'exercice, le tracé des fonctions $\pi_1(t)$ et $\pi_2(t)$ permet de distinguer les cas où le produit de convolution est non nul.

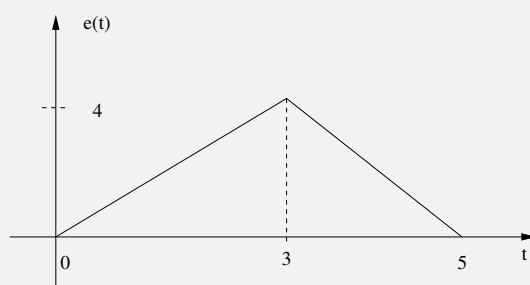
- Remarque: il faut faire attention au fait que le produit de convolution est défini par l'intégrale du produit des deux fonctions.

A.2 Réponse d'un système à un triangle

Enoncé:

Soit le système linéaire décrit par la réponse impulsionnelle:

(A.1)



➤ Calculer la réponse temporelle $y(t)$ du système pour l'entrée $e(t)$ suivante.

Seule la première intégrale non-nulle sera calculée entièrement. Les autres calculs seront posés, les bornes d'intégration définies ainsi que la fonction à intégrer.

Aide

- Savoir: définition du produit de convolution et définition d'une droite (définition de la pente et de l'ordonnée à l'origine).
- Compréhension: identifier au niveau du calcul de produit de convolution, quelle fonction est plus simple à déplacer en fonction de la variable τ ?
- Démarche:
 - il est nécessaire de poser de manière intelligente le produit de convolution.
 - Exprimer les fonctions $h(t-u)$ et $e(u)$.
 - Identifier les différents cas sans développer l'ensemble du calcul des intégrales.

Réponse d'un système à un signal porte

Énoncé:

Le signal suivant $e(t)$ est considéré :

$$e(t) = \begin{cases} 2 & \text{si } 0.5 < t < 3.5 \\ 0 & \text{sinon} \end{cases} \quad (\text{A.2})$$

Celui-ci est placé en entrée d'un filtre de réponse impulsionnelle $h(t)$:

$$h(t) = \sum_{k=-\infty}^{+\infty} \delta(t - 2k) \quad (\text{A.3})$$

↳ Calculer et représenter la sortie du filtre.

Si en entrée du filtre, le signal $e(t)$ est appliqué à la réponse impulsionnelle suivante:

$$h(t) = \sum_{k=-\infty}^{+\infty} e(t - 2k) \quad (\text{A.4})$$

↳ Dédurre simplement ce que serait la sortie du filtre de réponse impulsionnelle $h(t)$.

Aide

- Les notions indispensables préalables sont:
 - La notion de produit de convolution.
 - la notion de train de Dirac.
 - La périodisation temporelle d'une fonction.
 - La notion de recouvrement.
- En terme de démarche suggérée, il s'agit de calculer la réponse d'un filtre pour différentes entrées. La clef afin de résoudre le problème est d'utiliser le produit de convolution avec l'entrée en question afin d'estimer la sortie. La périodisation temporelle et les recouvrements sont à prendre en compte.

Chapitre B

Propriétés de la Transformée de Fourier

Objectifs

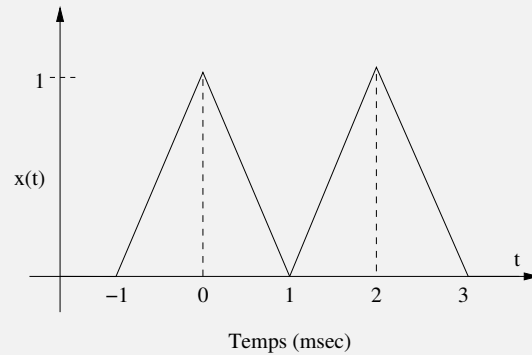
✘ Maîtriser le calcul de la transformée de Fourier de fonctions usuelles.

✘ Utilisation des propriétés de la transformée de Fourier.

B.1 Représentation temporelle d'un signal et propriétés de la transformée de Fourier

Enoncé:

Soit le signal $x(t)$ représenté par la figure suivante:



Sans calcul important, estimer au niveau de la transformée de Fourier de $x(t)$:

- ↪ La valeur en $X(0)$.
- ↪ Estimer sans calcul la valeur de $\int_{-\infty}^{+\infty} X(\nu) d\nu$.
- ↪ Estimer sans calcul la valeur de $\int_{-\infty}^{+\infty} X(\nu) e^{2i\pi\nu} d\nu$.
- ↪ Estimer sans calcul la valeur de $\int_{-\infty}^{+\infty} |X(\nu)|^2 d\nu$.

Aide

➤ Le savoir nécessaire:

- La transformée de Fourier d'une fonction $x(t)$ est définie par:

$$X(\nu) = \int_{-\infty}^{+\infty} x(t) e^{-2i\pi\nu t} dt \quad (\text{B.1})$$

- La transformée de Fourier inverse $X(\nu)$ est définie par:

$$x(t) = \int_{-\infty}^{+\infty} X(\nu) e^{2i\pi\nu t} d\nu \quad (\text{B.2})$$

- L'égalité de Parseval c'est-à-dire l'énergie d'une fonction ne dépend pas malheureusement de sa représentation temporelle, ni fréquentielle:

$$E = \int_{-\infty}^{+\infty} x^2(t) dt = \int_{-\infty}^{+\infty} |X(\nu)|^2 d\nu \quad (\text{B.3})$$

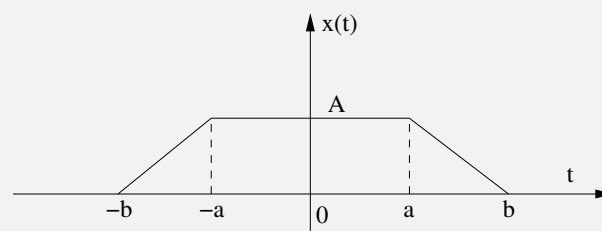
➤ Quelques questions à se poser:

- L'exercice comporte au premier abord, une forme de paradoxe: on demande de ne pas calculer la transformée de Fourier mais on demande des calculs ?
- Le calcul de $x(t)$ n'est pas impossible (la fonction peut être décomposée sous la forme de droite) mais demande en pratique beaucoup de calculs arithmétiques pour quelques valeurs seulement.
- En pratique, il est plus simple d'utiliser les définitions comme par exemple la valeur de la transformée de Fourier en $\nu = 0$, c'est à dire $X(\nu = 0)$.
- La notion de domaine en fréquence (ν) versus domaine en temps (t).

➤ La démarche suggérée est d'utiliser à bon escient les différentes définitions afin d'obtenir les valeurs demandées sur quelques points uniquement.

B.2 Calcul de la transformée de Fourier d'un signal évolué

Enoncé:



- ↯ Justifier l'existence de la transformée de Fourier de $x(t)$.
- ↯ Calculer la transformée de Fourier de $x(t)$ sans calculs d'intégrales compliquées.

Aide

➤ Le savoir nécessaire:

- *La différence entre l'espace des fonctions et des distributions.*
- La condition d'existence d'une transformée de Fourier.
- Les propriétés de la transformée de Fourier.

➤ La démarche de résolution:

- Pour le calcul de la transformée de Fourier, il est plus simple de dériver le signal $x(t)$ deux fois et de calculer la transformée de la dérivée simple puis de réintégrer.
- Il est également possible d'exprimer le signal $x(t)$ sous la forme du produit de convolution de deux portes comme au premier chapitre et décrire le produit des transformées de Fourier.

B.3 Filtrage et transformée de Fourier d'un produit de convolution

Énoncé:

Les signaux $x_1(t)$ et $x_2(t)$ suivant sont considérés:

$$x_1(t) = \text{sinc}(8\pi 10^3 t) \otimes \Pi_{[-3.75 \cdot 10^{-4}; -3.75 \cdot 10^{-4}]}(t) \quad (\text{B.4})$$

$$x_2(t) = \text{sinc}(76\pi 10^3 t) \otimes \Pi_{[-2.63 \cdot 10^{-5}; 2.63 \cdot 10^{-5}]}(t) \quad (\text{B.5})$$

↻ Déterminer les spectres des signaux $x_1(t)$ et de $x_2(t)$ et les représenter.

Les délais temporels suivant sont introduits:

$$x(t) = x_1(t + 3.75 \cdot 10^{-4}) \quad (\text{B.6})$$

et

$$x_2(t - 2.63 \cdot 10^{-5}t) \quad (\text{B.7})$$

ces signaux sont placés à l'entrée d'un filtre de réponse impulsionnelle:

$$h(t) = \sum_{k=-\infty}^{+\infty} \delta(t - 8.026 \cdot 10^{-4}k) \quad (\text{B.8})$$

↻ Calculer et représenter la sortie $y(t)$ du filtre.

Le signal $x_1(t)$ est placé en entrée du filtre $h(t)$.

↻ Déterminer et représenter le spectre d'amplitude de la sortie du filtre.

Aide

➤ Le savoir nécessaire:

- La transformée de Fourier d'un produit de convolution.
- La réponse d'un filtre en temporel.
- La transformée d'un produit.

➤ La démarche de résolution:

- Avant de se lancer sans réflexion dans des calculs complexes, il est intéressant de calculer la transformée de Fourier de la porte $\pi_{[-a; +a]}(t)$.
- et d'estimer également la transformée de Fourier de $\text{sinc}(2\pi t a)$.

Chapitre C

Calcul de Transformée de Fourier

Objectifs

- ⊗ Calcul de la transformée de Fourier de fonctions usuelles.
- ⊗ Utilisation des propriétés de la transformée de Fourier (influence du délai).
- ⊗ Echantillonnage et spectre d'un signal.

C.1 Phase et transformée de Fourier d'un signal sinusoïdal

Enoncé:

Soit le signal $x(t)$ défini par l'équation suivante:

$$x(t) = \sin(31.42t + 0.4) \quad (\text{C.1})$$

- ↯ Ce signal est-t-il d'énergie ou de puissance finie ?
- ↯ Quelle est la fréquence de $x(t)$?
- ↯ Quelle est la valeur du paramètre t_0 permettant de mettre le signal sous la forme d'un sinus retardé de t_0 ?

$$x(t) = \sin(2\pi\nu(t - t_0)) \quad (\text{C.2})$$

- ↯ Donner l'expression du spectre de $x(t)$?
- ↯ Déduire la densité spectrale de puissance du signal x puis la fonction d'autocorrélation.

Aide

➤ Le savoir nécessaire:

- Paramètres simples d'un cosinus, sinus (fréquence, phase, amplitude, retard).

- Notion d'énergie ou de puissance.
- Spectre élémentaire d'un cosinus ou d'un sinus.
- Théorème de Weiner-Kintchine pour la relation entre la notion de corrélation et de Densité Spectrale de Puissance (DSP).

➤ Démarche et question à se poser:

- Peut-on dans le cas présent directement utiliser la transformée de Fourier d'un signal sinusoïdal ?
- L'astuce consiste en utilisant la propriété du retard à se ramener à l'expression de la transformée d'un simple signal sinusoïdal.

C.2 Linéarisation et spectre d'un signal cosinusoidal

Énoncé

Soit le signal $x(t)$ suivant:

$$x(t) = A_0 + A_1 \cos^2(4\pi\nu_0 t + 2\phi_0) \quad (\text{C.3})$$

où A_1 est une constante réelle positive et A_0 une constante réelle négative.

- ☞ Déterminer l'expression du spectre de $x(t)$.
- ☞ Quelle est l'expression du spectre d'amplitude ? Le représenter.
- ☞ Quelle est l'expression du spectre de phase ? Le représenter.
- ☞ Déterminer de façon simple la puissance de $x(t)$.

Aide

➤ Le savoir nécessaire:

- Le spectre d'un cosinus simple.
- La linéarisation d'un cosinus et la prise en compte de la phase (retard) comme à l'exercice précédent.
- Spectre d'amplitude et notion de puissance du signal.

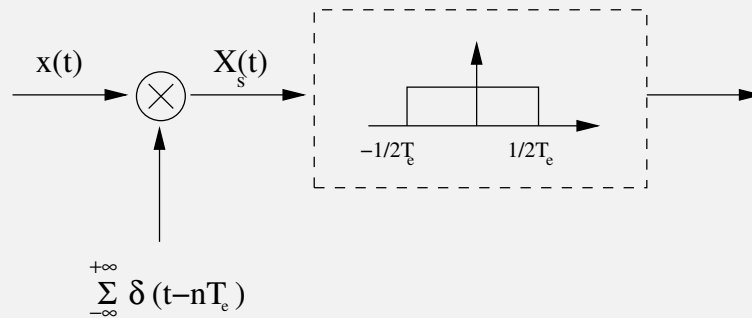
➤ La démarche de résolution:

- Se poser la question, quels sont les points bloquants par rapport au calcul de la transformée d'un cosinus simple ?
- Exprimer le signal $x(t)$ sous une forme plus simple permettant l'expression de sa transformée de Fourier sans calcul complexe.

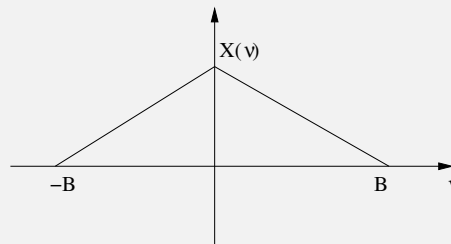
C.3 Echantillonnage, spectre et recouvrement

Enoncé:

Soit un signal continu $x(t)$:



- Donner l'expression de la transformée de Fourier du signal $x_s(t)$ en entrée du filtre.
L'hypothèse que la transformée de Fourier $x(t)$ est réalisée:



- Représenter $X_s(\nu)$ dans les cas où $T = \frac{1}{4B}$, $T = \frac{1}{2B}$ et $T = \frac{1}{B}$. Expliquer le phénomène.
➤ Pour les trois périodes d'échantillonnage différentes, représenter la transformée de Fourier du signal en sortie du filtre passe-bas.
➤ Pour quelle valeur maximale de la période d'échantillonnage T_e , retrouve-t-on le signal $x(t)$ en sortie ?

Aide

- Les connaissances nécessaires pour cet exercice sont les suivantes:
- La transformée de Fourier d'un train de dirac
 - La notion de recouvrement
 - Le théorème de "Shannon" sur l'échantillonnage
 - Le spectre d'un signal échantillonné
- La démarche de résolution est d'exprimer simplement le signal en entrée du filtre en identifiant dans le domaine temporel, l'opération réalisée (échantillonnage) et sa conséquence dans le domaine fréquentiel.

- L'aspect filtrage abordé étant idéal, le spectre restant sera facilement identifié et la fréquence d'échantillonnage maximale sera retrouvée afin d'éviter le recouvrement.

Chapitre D

Numérisation d'un signal

Objectifs

- ☒ Comprendre le principe de numérisation et d'échantillonnage d'un signal.
- ☒ Echantillonnage et transformée de Fourier d'un signal échantillonné.

D.1 Echantillonnage d'un signal composé de deux sinusoides

Enoncé:

Soit un signal continu $v(t)$ comportant deux sinusoides $V_1(t)$ et $V_2(t)$:

$$\boxed{V(t) = V_1(t) + V_2(t)} \quad (\text{D.1})$$

$V_1(t)$ est un signal sinusoidal de fréquence 1 Hz, d'amplitude entre 0 et 5 Volts alors que $V_2(t)$ est un signal cosinusoidal de fréquence 10 Hz d'amplitude plus faible entre -0.1 et 0.1 Volt.

- ↯ Donner une description mathématique de $V(t)$.
- ↯ Tracer de manière précise le spectre d'amplitude de $V(t)$.
Supposons que les deux signaux forment un signal porteur d'information et que le signal $V(t)$ doit être échantillonné.
- ↯ Quelle est la fréquence d'échantillonnage adéquate ?
- ↯ Reprendre la question précédente mais en considérant que $V_2(t)$ est du bruit perturbant le signal.

Aide

- La savoir préalable pour la résolution de l'exercice est le suivant:
 - L'expression d'un cosinus et d'un sinus.
 - Le spectre classique en transformée de Fourier.
 - Le théorème de Shannon au niveau de l'échantillonnage.

- La compréhension de différentes notions de base:
 - Les paramètres d'un cosinus, sinus c'est à dire principalement sa fréquence et son amplitude.
 - La différence entre un bruit et un signal.
- La démarche suggérée pour résoudre l'exercice est la suivante:

L'exercice ne comporte pas de difficultés techniques, il s'agit d'exprimer l'expression mathématique, calculer le spectre à partir des expressions du cours pour des signaux simples et, enfin, d'utiliser le théorème de Shannon à bon escient.

D.2 Echantillonnage et filtrage

Énoncé:

Un signal est échantillonné à 500 échantillons par seconde un signal réel continu qui est la somme de trois sinusoides de fréquences respectives 50 Hz, 100 Hz et 300 Hz.

- ✎ Esquisser le spectre d'amplitude du signal analogique.
- ✎ Esquisser le spectre d'amplitude du signal échantillonné en expliquant les phénomènes observés.
- ✎ Le signal analogique initial est appliqué en entrée d'un filtre de réponse impulsionnelle

$$h(t) = \text{sinc}(10\pi t) \quad (\text{D.2})$$

Quelle est l'allure du spectre d'amplitude du signal en sortie du filtre ?

Aide

- Savoir:
 - L'expression de la transformée de Fourier d'un sinus.
 - Le spectre d'un signal échantillonné.
 - La relation entre l'entrée et la sortie d'un filtre.
- En terme de compréhension:
 - La notion de recouvrement spectral.
 - La relation entre réponse impulsionnelle et réponse en fréquence.

En terme de démarche, le calcul du spectre du signal analogique ne pose pas de problèmes si l'on pose des amplitudes quelconques (A_0, A_1, A_2) . Il faut faire attention à la notion de recouvrement pour le spectre échantillonné

D.3 Conception d'un convertisseur analogique numérique

Enoncé:

En tant qu'ingénieur en traitement du signal, un client formule son besoin d'un système de numérisation spécifique basé sur un convertisseur analogique-numérique (CAN).

Trois signaux différents sont susceptibles d'être analysés :

- Une porte centrée d'amplitude 3V et de largeur 6 μ s.
- Un signal cosinus crête à crête d'amplitude 4 volts et de fréquence maximale de 200 Hz.
- Un signal en sinus cardinal d'amplitude maximale de 2 volts et de largeur de lobe principal maximal de 0.5 ms.

Le client souhaite une erreur de quantification maximale de 0.1 volt et que les signaux numériques soient numérisés sur le moins de bits possible.

➤ Proposer un système de numérisation complet.

Aide

- Il est nécessaire de connaître :
 - Les principes de la numérisation (échantillonnage, quantification).
 - Paramètre d'un ADC convertisseur analogique/numérique.
 - Fréquence d'échantillonnage et théorème de Shannon.
 - Erreur de quantification.
- Compréhension:
 - Les principes de la numérisation (échantillonnage, quantification).
- En terme de démarche de résolution, il est nécessaire d'estimer le spectre des trois signaux afin d'estimer une fréquence d'échantillonnage raisonnable. L'estimation de la dynamique maximale (en amplitude) est aussi nécessaire.

Chapitre E

Transformée de Fourier Discrète d'un Signal

Objectifs

- ⊗ Illustrer avec l'aide du logiciel Octave ou Matlab l'échantillonnage d'un signal.
- ⊗ Comprendre la notion de Transformée de Fourier Discrète (TFD) et de résolution fréquentielle.

E.1 Représentation et analyse fréquentielle d'un signal

Enoncé:

La génération d'un signal numérique à partir d'un ordinateur ou d'un processeur exige quelques précautions quant au choix de l'écart temporel entre la prise consécutive de deux échantillons. S'il est bien admis que le théorème de Shannon régit le choix optimal de la période d'échantillonnage lors de l'acquisition d'une information numérique représentative de l'évolution d'un phénomène physique analogique, il n'en demeure pas moins vrai que dans les problèmes de simulation ou de génération de signaux synthétiques, cette quantité mal choisie peut introduire de graves erreurs d'interprétation.

- ↻ Générer un signal de 0.5 s composé de la somme de deux sinusoides d'amplitude 1V échantillonnées à 256 Hz et dont la pulsation est aux fréquences respectives de 100 et 156 Hz. Représenter ce signal. Conclure.
- ↻ A partir du logiciel Matlab ou Octave générer et représenter une sinusoïde d'amplitude 1V dont la pulsation est à la fréquence de 356 Hz et échantillonnée à 256 Hz. Comparer ce signal à la sinusoïde dont la fréquence est situé à 100 Hz et échantillonnée à 256 Hz.

Aide

- Le savoir *théorique* nécessaire est le principe de l'échantillonnage temporel, c'est-à-dire:
 - La durée d'observation.

- La relation période/fréquence d'échantillonnage.
- Le nombre total d'échantillons.
- Le savoir pratique au niveau de Matlab&Octave.
 - Déclarer un tableau d'entier (fonction `[]`,:).
 - Afficher un signal numérique (fonction `stem`).
 - Calculer un sinus numérique (notion échantillon).

E.2 Notion de résolution fréquentielle et finesse d'analyse

Énoncé:

On souhaite calculer en utilisant la transformée de Fourier Discrète (TFD) avec une résolution d'au moins 5 Hz, le spectre d'un signal pour lequel, il est possible d'admettre que le spectre soit négligeable pour les fréquences au-delà de 1.25 kHz.

- ✎ Quelles sont la fréquence d'échantillonnage minimale du signal à traiter ?
- ✎ Quel est le nombre de points à considérer et la résolution spectrale obtenue.
- ✎ Est-il possible de compléter le signal par des échantillons de valeur nulle si on ne dispose que d'une observation de durée $T=150\text{ms}$?

Aide

- Le savoir nécessaire:
 - La relation entre la résolution d'une transformée de Fourier Discrète et la durée d'observation temporelle.
 - La différence entre la notion de finesse d'analyse ou également appelée résolution apparente et la résolution.
- En terme de compréhension:
 - La différence entre l'échantillonnage en temporel et échantillonnage en fréquentiel (notion de résolution apparente ou pas).
- En terme de démarche, le principe est d'estimer la fréquence minimale d'échantillonnage par le théorème de Shannon et la durée d'observation minimale également pour obtenir la résolution requise.

E.3 Périodicité et analyse spectrale par transformée de Fourier Discrète

Enoncé:

Un signal périodique $x(t)$ échantillonné à la fréquence d'échantillonnage ν_e . Un échantillon est noté $x(n)$. Nous avons la connaissance des échantillons qui sont de la forme:

$$x(n) = 1 + \sin\left(\frac{n\pi}{3}\right) + 2 \cos\left(\frac{n\pi}{4}\right) \quad (\text{E.1})$$

- ↻ Ecrire la fonction $x(t)$ dont l'échantillonnage a donné l'expression précédente de $x(n)$. L'expression mathématique par identification faire intervenir un signal sinusoidal de fréquence f_0 et un signal cosinusoidal de fréquence f_1 .
- ↻ Donner les expressions de f_0 et de f_1 en fonction de $T_e = \frac{1}{\nu_e}$.
- ↻ Quelle est la période du signal $x(t)$ en fonction de T_e .
- ↻ Combien $x(t)$ comporte-t-il d'échantillons par période ? Ce nombre est noté K .
- ↻ Réaliser une analyse spectrale du signal de x par une transformée de Fourier Discrète sur N points. Donner, en fonction de K ; le nombre de points N permettant d'obtenir un spectre de TFD ne faisant pas apparaître de raies non souhaitées, c'est-à-dire, autres que celles placées aux maxima des lobes principaux.
- ↻ Evaluer le spectre du signal analogique $x(t)$ à partir du spectre de la transformée de Fourier Discrète.

Aide

- En terme de savoir:
 - La période d'un signal continu.
 - La relation entre un signal continu et échantillonné.
 - La condition nécessaire pour la calcul de la transformée de Fourier discrète d'un signal période.
- La démarche de résolution:

Le principe est à partir de l'expression des échantillons de deviner la forme du signal continu à l'origine des échantillons et d'exprimer certains paramètres comme la fréquence, la période en fonction de ceux du signal échantillonné. La transformée de Fourier Discrète (TFD) d'un signal numérique exige d'utiliser la condition nécessaire pour ne pas avoir de raies non souhaitées et retrouver ainsi le spectre du signal analogique périodisé.

Chapitre F

Transformée de Fourier Discrète

Objectifs

- ✘ Calcul et représentation de la transformée discrète de fonctions classiques.
- ✘ Interprétation d'une transformée de Fourier Discrète.
- ✘ Illustration utilisant le logiciel Octave ou Matlab.

F.1 Spectre et calcul d'une transformée de Fourier d'ordre 3

Enoncé:

Un signal $x(t)$ échantillonné à $T_e = 0.1s$ est considéré à partir de $t = 0$. On cherche à calculer la transformée de Fourier discrète (TFD) à partir de trois échantillons temporels $x(n)$ $n \in \{0,1,2\}$:

$$\boxed{x(0) = 0, \quad x(1) = 1, \quad x(2) = 0} \quad (\text{F.1})$$

- ↻ Ecrire la matrice 3×3 permettant de calculer la TFD de x .
- ↻ En quelles fréquences la TFD va-t-elle évaluer le spectre de x ?
- ↻ Calculer les échantillons spectraux de TFD.
- ↻ Tracer les spectres d'amplitude et de phase de la TFD pour $\nu \in [-20; 20]$ Hz. L'axe des fréquences sera gradué en Hertz.

Aide

- Connaissances préalable indispensable:
 - Notion de résolution d'une transformée de Fourier Discrète
 - Définition matricielle d'une transformée de Fourier Discrète
 - Périodicité du spectre de transformée de Fourier Discrète (nécessaire pour la représentation du spectre)

- Cet exercice est une application simple de la transformée de Fourier Discrète. Il suffit d'écrire la matrice correspondant à la TFD et d'exprimer le produit de cette matrice par les échantillons temporels. Une fois, les échantillons spectraux obtenus, il faudra identifier le module et la phase pour le tracé du spectre.

F.2 Transformée de Fourier Discrète d'une séquence cosinusoidale

Enoncé:

- ✎ Calculer la TFD sur d'une séquence 1D cosinusoidale, de durée une seconde, de fréquence 2.6 Hz, d'amplitude 1 et échantillonnée à la fréquence 50 de Hz, en utilisant les fonctions `fft.m` et `fftshift.m` de Matlab ou Octave.
- ✎ Reprendre la même question en calculer la TFD sur 100 points. Expliquer le phénomène observé.
- ✎ Quelle fenêtre de pondération a été utilisée au niveau de la première question ?
- ✎ Reprendre le même signal, utiliser une autre fenêtre de pondération pour représenter le spectre. Conclure.
- ✎ Reprendre le signal précédent et ajouter une cosinusoide de durée 1 seconde, de fréquence 12.25 Hz, d'amplitude 0.05 V et échantillonnée à 50 Hz.
- ✎ Reprendre la question 1. Conclure.
- ✎ Reprendre la question 4. Comparer et conclure.

Aide

- Connaissances nécessaire:
 - La notion de transformée de Fourier Discrète
 - L'influence du nombre d'échantillons sur la résolution fréquentielle
 - L'influence du fenêtrage dans le domaine temporel.
- Savoir pratique Matlab&Octave:
 - La fonction `fft.m` implémentant la TFD n'est pas accompagnée de la définition d'un tableau de fréquences.
 - La fonction `fftshift` réorganise les échantillons spectraux afin d'obtenir le spectre analogique dans la gamme de fréquence $[-\frac{\nu_e}{2}, -\frac{\nu_e}{2}]$.
 - Le fenêtrage des échantillons s'effectue dans le domaine temporel et nécessite de réaliser un produit de vecteurs (opérateur `.*`).

F.3 Spectre de transformée de Fourier Discrète et Analogique

Enoncé:

Un signal $x(t)$ a été échantillonné, en respectant le théorème de Shannon, à la fréquence $F_e = 12$ kHz. Un bloc de cinq cent échantillons a été prélevé pour le traitement. Ces cinq cent échantillons forment un signal noté $x(n)$. Après calcul et représentation, il est constaté que les échantillons de transformée de Fourier discrète $X(k)$ calculés sur mille vingt quatre points contiennent uniquement des raies pour les échantillons spectraux $k \in \{40, 100, 924, 984\}$ ainsi qu'une phase nulle.

- ↻ Où sont situées les fréquences du signal analogique $x(t)$ si l'on considère un axe en Hertz ?
- ↻ Déduire le contenu du signal analogique $x(t)$ en supposant que ces raies ont une amplitude de 1.

Aide

- Les connaissances nécessaires sont:
 - La notion de résolution spectrale.
 - la notion de finesse d'analyse (c'est-à-dire l'écart entre deux raies fréquentielles).
 - La notion de spectre analogique. Ce dernier est situé entre $[-\frac{\nu_e}{2}, \frac{\nu_e}{2}]$.
- En terme de compréhension, la notion de finesse d'analyse est indispensable afin d'estimer les valeurs des différentes composantes fréquentielles.
- Il faut représenter le spectre du signal complet en particulier dans le domaine du spectre analogique $[-\frac{\nu_e}{2}, \frac{\nu_e}{2}]$. Le spectre analogique peut être raisonnablement assimilé à deux signaux cosinusoidaux de phase nulle et d'amplitude inconnue.

Chapitre G

Analyse et Applications des Signaux Aléatoires

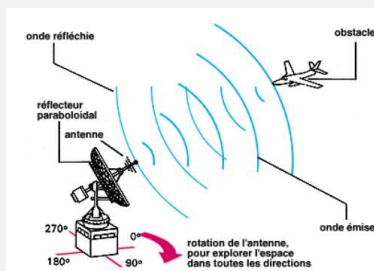
Objectifs

- ✘ Illustrations de l'utilisation d'un bruit gaussien.
- ✘ Application à la détection dans les domaines du radar/sonar/télécommunications.

G.1 Chaîne radar et probabilité de détection

Enoncé:

Un radar comportant une chaîne numérique d'émission et de réception peut se résumer simplement de la manière suivante:



Un signal est émis par l'émetteur radar. Ce signal est une porte de durée T et d'amplitude $+6V$. Si cette onde émise rencontre un obstacle aérien, une partie de l'onde va être réfléchi par ce dernier vers le récepteur Radar. Tout signal se propageant dans l'atmosphère subit des altérations dont la plus importante est l'ajout de bruit rendant ce signal aléatoire. Le récepteur radar reçoit donc :

- Le signal réfléchi auquel vient se superposer un bruit blanc additif gaussien d'écart-type et de moyenne m (avec $\sigma = 2V$ et $m = 1V$) en cas de présence d'obstacle.
- Un bruit blanc gaussien d'écart-type et de moyenne m (avec $\sigma = 2V$ et $m = 1V$) en cas d'absence d'obstacle.

Au niveau du récepteur, il s'agit ensuite de prendre une décision à instants réguliers pour savoir si oui ou non un obstacle est présent dans la zone aérienne étudiée. Pour cela, la règle de décision mise en place dans le cadre de cet exercice est la suivante : toutes les T secondes, on prélève un échantillon du signal reçu et :

- Si cet échantillon possède un niveau inférieur à $3V$, on décide qu'il n'y a pas d'obstacle.
- Si cet échantillon possède un niveau supérieur à $3V$, on décide qu'il y a présence d'obstacle.

Sachant que dans la zone étudiée, il y a autant de chance à priori qu'il y ait ou non un obstacle, quelle est la probabilité de se tromper sur la présence ou non d'un obstacle ?

Aide

➤ Le savoir nécessaire:

- * Savoir de base sur les variables et signaux aléatoires.
- * Notion de bruit blanc.
- * Probabilité erreur conditionnelle.

* La loi normale centrée ($m = 0$, $\sigma = 1$).

- Aspect compréhension: le verrou principal est probablement situé au niveau de l'analyse de ce qui est reçu au niveau de l'antenne en terme de signaux aléatoires.
- La démarche suggérée pour la résolution est la suivante: Il s'agit de faire le bilan des signaux aléatoires reçus dans les deux cas à savoir *l'absence* ou *la présence* d'un obstacle. Le tracé des deux densités de probabilité et d'identifier la probabilité d'erreur à intégrer. Calculer ensuite les deux intégrales en faisant particulièrement attention au changement de variables.

Partie III

Praticiel

Chapitre H

Analyse de Signaux par Transformée de Fourier Discrète

H.1 Objectifs

L'objectif de ce pratique est de:

- ⊗ Mettre en œuvre à l'aide du logiciel Matlab ou Octave des représentations de signaux discrets, d'échantillonnage, de calcul de leur transformée de Fourier discrète.
- ⊗ Illustrer de manière plus pratique les notions de signaux numériques, déchantillonnage, de transformée de Fourier Discrète, de fenêtrage et de résolution en fréquence enseignées dans le cadre de ce cours.

H.2 Requis préalable

La préparation nécessaire à ce pratique est la suivante :

- Se remémorer le cours en particulier les notions de signaux numériques et d'échantillonnage.
- Prendre connaissance des notions sur la transformée de Fourier discrète dans le cas où le cours n'a pas encore eu lieu.
- Disposer du logiciel Matlab ou Octave fonctionnel.

H.3 Rappels sur les notions de Transformée discrète

A) Un peu de théorie

La transformée de Fourier continue enseignée dans le cours de mathématiques n'est pas directement implémentable par un ordinateur. En effet, si l'on considère le signal temporel $x(t)$, sa transformée de Fourier continue $X(\nu)$ est :

$$X(\nu) = \int_{-\infty}^{+\infty} x(t)e^{-2i\pi\nu t} dt \quad (\text{H.1})$$

De manière à pouvoir réaliser une transformée de Fourier sur un ordinateur, différentes simplifications sont introduites. Ceci donne par simplification, l'expression finale de la transformée discrète :

$$X_{TFD}(k) = \sum_{n=0}^{N-1} x(nTe) e^{\frac{-2i\pi nk}{N}} \quad (\text{H.2})$$

avec k caractérise le numéro des échantillons spectraux $\in \{0,1,2,\dots,N-1\}$.

La résolution fréquentielle est donnée par:

$$\boxed{\nu_s = \frac{1}{NT_e}} \quad (\text{H.3})$$

La transformée de Fourier discrète se caractérise par une somme d'échantillons temporels qui sont chacun d'entre eux multipliés par une exponentielle complexe. La phase de cette exponentielle complexe dépend de différents paramètres comme l'indice de l'échantillon temporel (n) ou fréquentiel (k). Elle est parfaitement adaptée à un ordinateur car en pratique on ne va réaliser l'opération que pour une série de nombres. L'échantillon spectral extrait de la transformée de Fourier discrète est un nombre complexe caractérisé par un module et une phase.

Il est possible d'exprimer la transformée de Fourier discrète sous la forme d'un produit matriciel:

$$\begin{pmatrix} X(k=0) \\ X(k=1) \\ \vdots \\ X(k=N-2) \\ X(k=N-1) \end{pmatrix} = \begin{pmatrix} 1 & 1 & \dots & 1 & 1 \\ 1 & \dots & \dots & \dots & \dots \\ \vdots & \vdots & \exp \frac{-2ink}{N} & \vdots & \vdots \\ 1 & 1 & \dots & \dots & \dots \\ 1 & 1 & \dots & \dots & \dots \end{pmatrix} \times \begin{pmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-2) \\ x(N-1) \end{pmatrix} \quad (\text{H.4})$$

Les différents échantillons de TFD $X(k)$ sont obtenus par cette multiplication matricielle entre les échantillons temporels $x(n)$ et la matrice de TFD formée par les termes $e^{\frac{-2i\pi nk}{N}}$.

Exemple: transformée de Fourier discrète d'un signal sinusoidal

Soit le signal sinusoidal de fréquence unitaire échantillonné à une fréquence de 4 Hz. La transformée de Fourier Discrète est calculée sur 4 points. Les échantillons temporels sont alors :

$$x(nT_e) = \{x(0) = 0; \quad x(1) = 1; \quad x(2) = 0; \quad x(3) = -1\} \quad (H.5)$$

Si la formule précédente (H.6) est appliquée, l'équation suivante à résoudre est la suivante :

$$\begin{pmatrix} X(k=0) \\ X(k=1) \\ X(k=2) \\ X(k=3) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ -2i \\ 0 \\ 2i \end{pmatrix} \quad (H.6)$$

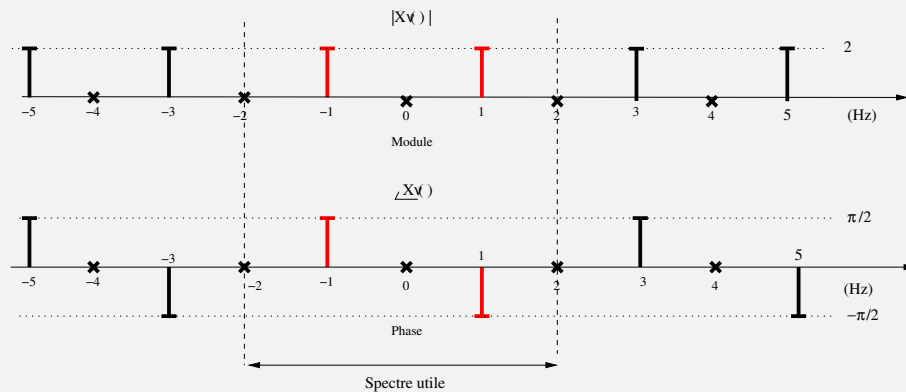
Les échantillons $X(k)$ de transformée discrète en fréquentiel sont respectivement:

$$X(k) = \{X(0) = 0; \quad X(1) = -2i; \quad X(2) = 0; \quad X(3) = 2i\} \quad (H.7)$$

Il reste maintenant à tracer le spectre en module et en phase des échantillons fréquentiels. La résolution fréquentielle est égale à:

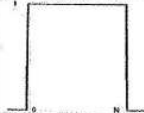
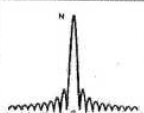
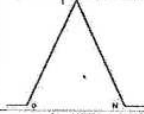
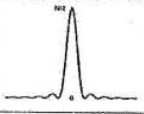
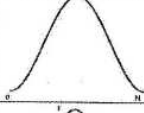
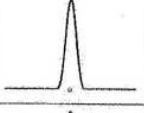
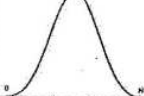
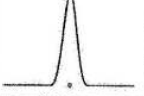
$$\Delta\nu = \frac{1}{NT_e} = 4 * \frac{1}{4} = 1Hz \quad (H.8)$$

Un échantillon sera tracé tous les Hertz. Notons que le signal de transformée de Fourier discrète est périodique par translation de période ν_e .



B) Fenêtrage et TFD d'un signal périodique

Deux aspects importants vont être étudiés dans le cadre du TP à savoir l'influence du fenêtrage et la condition à respecter afin de calculer la TFD d'un signal périodique. Ces deux aspects sont largement détaillés dans le cours de signaux numériques et dans les travaux dirigés. Il n'est pas question de refaire tout ce développement ici. Seuls les résultats principaux nécessaires au TP sont exposés. Ce dernier sera le cadre idéal afin de découvrir ces deux notions. La figure suivante présente les principales fenêtres disponibles ainsi que leurs principales propriétés.

Nom	Equation $\forall n \in \{0 \dots N-1\}$	Représentation temporelle	Représentation fréquentielle	Largeur lob.princ.	Amp. relative $\frac{\text{lob.princ}}{\text{lob.sec.}}$
Rectangulaire	$w[n] = 1$			$\frac{2}{N}$	-13 dB
Triangulaire	$w[n] = 1 - \frac{ n - \frac{N-1}{2} }{\frac{N-1}{2}}$			$\frac{4}{N}$	-25 dB
Hanning	$w[n] = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right)$			$\frac{4}{N}$	-41 dB
Blackman	$w[n] = 0.42 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right)$			$\frac{6}{N}$	-57 dB

Différents types de fenêtres et leurs caractéristiques

a) Fenêtrage

De manière à diminuer le nombre de raies parasites introduites par la transformée de Fourier Discrète, un fenêtrage est utilisé. En pratique, les échantillons temporels $x(nT_e)$ sont multipliés par les échantillons de la fenêtre $h(nT_e)$. La transformée de Fourier discrète est ensuite calculée:

$$X_{TFD}(k) = \sum_{n=0}^{N-1} [x(nT_e)h(nT_e)]e^{-\frac{2i\pi nk}{N}} \quad (\text{H.9})$$

b) Conditions d'échantillonnage et cas d'un signal périodique

Soit T_0 la période du signal considérée, T_e la période d'échantillonnage et N le nombre de points de la TFD. La condition nécessaire et suffisante pour calculer la TFD d'un signal périodique afin de ne pas faire apparaître de raies parasites est qu'il faut échantillonner un nombre entier de période L le signal. En pratique, N , T_0 , L , T_e vérifient donc la relation suivante:

$$NT_e = LT_0 \quad (\text{H.10})$$

H.4 Simulation numérique Matlab ou Octave

A) Transformée de Fourier discrète d'un signal sinusoïdal

Énoncé

La transformée de Fourier discrète (TFD) correspond à un échantillonnage dans le domaine fréquentiel de la transformée d'un signal discret. Un algorithme de calcul de transformée de Fourier discrète est implémenté dans Matlab ou Octave, il s'agit de la fonction *fft*. La fréquence d'échantillonnage vaut 40960 Hz. La résolution en fréquentielle correspond à l'aptitude à distinguer deux fréquences dans un signal. La résolution peut être quantifiée par l'écart maximal entre les fréquences de deux sinusoides afin d'observer le signal résultant avec un écart de plus de 3 dB entre les deux maxima. La résolution est de l'ordre de $\frac{F_e}{N}$, la valeur inverse de la durée d'observation NT_e . Du fait de la présence des lobes secondaires, la possibilité de séparer deux sinusoides de fréquences proches va dépendre de leurs amplitudes respectives.

- Quelle est la résolution fréquentielle sachant que les signaux comportent 2048 échantillons ?
- Générer effectivement un signal de 2048 points comportant deux sinusoides de fréquences f_0 , f_1 , A_0 et A_1 :
 - * $f_0 = 1000\text{Hz}$ et $f_1 = 15000\text{ Hz}$, $A_0 = 1$ et $A_1 = 1$.
 - * $f_0 = 2000\text{Hz}$ et $f_1 = 2040\text{ Hz}$, $A_0 = 1$ et $A_1 = 1$.
 - * $f_0 = 2412\text{Hz}$ et $f_1 = 2600\text{ Hz}$, $A_0 = 1$ et $A_1 = 0.025$.
 - * $f_0 = 2412\text{Hz}$ et $f_1 = 2568\text{ Hz}$, $A_0 = 1$ et $A_1 = 1$
- Recommencer les cas précédents en utilisant une fenêtre de Hamming.

B) Modulation en fréquence d'un signal et repliement du spectre

Énoncé

Générer un signal cosinusoidal $s(t)$ de fréquence instantanée $f(t)$:

$$s(t) = \cos(2\pi f(t)t) \quad (\text{H.11})$$

La fréquence instantanée $f(t)$ varie linéairement en temporel d'une fréquence f_0 à f_1 . Ce signal est échantillonné à une fréquence de 8 kHz. Deux cas peuvent être considérés: $f_0=100\text{ Hz}$ et $f_1= 1000\text{ Hz}$ ou $f_0=100\text{ Hz}$ et $f_1= 6000\text{ Hz}$.

Identifier le phénomène observé (nota-bene: on pourra représenter le signal temporel échantillonné à l'écran et écouter le signal obtenu à l'aide de la fonction *soundsc.m*).

Découvrir la notion de spectrogramme (fonction (spectrogram.m)) afin de représenter la variation temporelle de la fréquence du signal.

Chapitre I

Compression d'Image et Transformée en Cosinus Discret

Ce praticiel a pour objectif d'étudier l'effet de la compression par la transformation en Cosinus Discret pour différentes images modèles. Dans le cadre des logiciels Matlab ou Octave, une image en niveaux de gris n'est ni plus ni représentée par une matrice donc les valeurs appartiennent à l'intervalle $[0; 255]$ c'est-à-dire une décomposition en 256 niveaux de gris.

- ✘ Découvrir la notion d'image et la création numérique.

- ✘ Appréhender une autre transformée celle en Cosinus Discret.

- ✘ Observer l'influence par la pratique d'une compression d'image.

I.1 Découverte de la transformée de Fourier en Cosinus Discret (2D)

Énoncé:

Le premier temps consiste à créer une image:

- Créer à l'aide de Matlab ou Octave une image de 512 pixels x 512 pixels totalement noire avec en son centre un carré de pixels blancs de largeur 100 pixels. Pour rappel, la couleur noire correspond à un niveau de gris 0 et le blanc à un niveau de gris 255. Visualiser l'image en niveaux de gris grâce aux fonctions *colormap.m*, *imagesc.m*.
- Appliquer la transformée en Cosinus Discret sur l'image créée (en utilisant la fonction *dct2.m*)
- Visualiser le résultat de ce calcul.
- Grâce à la fonction *idct2.m*, reconstruire l'image initiale en ne retenant que :
 - * Les 80% plus grands coefficients de la transformée en cosinus discret (DCT).
 - * Les 50% plus grands coefficients de la transformée en cosinus en cosinus discret (DCT).
 - * Les 20% plus grands coefficients de la transformée en cosinus en cosinus discret (DCT).

Conclure sur l'effet de la compression par transformée en Cosinus Discret. Quel cas étudié ci-dessus vous semble le plus approprié pour compresser votre image ?

I.2 Application à la compression d'une image

Enoncé:

Le second temps nécessite de télécharger une image disponible sur le campus:

- Dans la rubrique du campus dédié au cours d'analyse des Signaux et des Images, télécharger et installer l'image 1 ou l'image 2 sur le disque de votre ordinateur. Ces images sont au format PGM (Portable GrayMap).
- Charger l'image sur Matlab ou Octave grâce à la fonction *imread.m*. Appliquer la compression d'image choisie à la quatrième question de la section précédente pour l'image que vous avez retenue (1 ou 2).

Conclure et ajuster votre seuil de coefficients si nécessaire.

Chapitre J

Etude et Analyse de Signaux Aléatoires

Objectifs

- ✘ Découvrir et illustrer la notion de signaux aléatoires et de bruit blanc gaussien
- ✘ Utiliser les outils de traitements de signal de Matlab ou Octave afin d'illustrer les différentes notions (moyenne, variance et stationnarité).

J.1 Séries aléatoires et statistiques simples

Enoncé de la première partie:

A l'aide de la fonction *randn*, générer mille échantillons d'une variable aléatoire de loi normale centrée (moyenne nulle) réduite (variance unité).

- Représenter l'histogramme de cette variable aléatoire grâce à la fonction *hist.m*. Commenter le résultat.
- Calculer la moyenne à l'aide de la fonction *mean* et commenter.
- Calculer la variance à l'aide de la fonction *std* et commenter.
- Reprendre ces trois questions à l'aide de vingt échantillons puis de dix mille échantillons d'une variable aléatoire avec les mêmes propriétés. Commenter.

Reprendre la première question a) avec mille échantillons et une variable aléatoire normale de moyenne deux et de variance unité.

Reprendre la première question a) avec mille échantillons et une variable aléatoire normale de moyenne deux et de variance quatre.

J.2 Analyse d'un bruit blanc gaussien

Énoncé de la seconde partie:

Le but de cette partie est d'analyser les différentes caractéristiques d'un bruit blanc gaussien. Les signaux sont supposés ergodiques. Une seconde de bruit blanc gaussien de moyenne nulle et de variance trois est enregistré. Ce bruit est échantillonné à la fréquence 20 kHz.

- Générer ce bruit sous Matlab ou Octave et le représenter en fonction du temps.
- Représenter l'histogramme de ce signal.
- Quelle est la fonction d'autocorrélation du bruit de la première question ?
A l'aide de la fonction *xcorr* représenter la fonction d'autocorrélation de ce bruit. Commenter.
- Quelle est la densité spectrale de puissance du bruit de la question une ? Représenter la densité spectrale de puissance de ce bruit en fonction de la fréquence et commenter.
- Ce signal est filtré par un filtre numérique RII (Réponse Impulsionnelle Infinie) de fonction de transfert en z :

$$H(z) = \frac{0.05z}{z - 0.95} \quad (\text{J.1})$$

A l'aide de la fonction *filter.m*, visualiser le bruit filtré temporellement et le comparer au bruit d'origine en superposant les deux courbes grâce à la fonction *hold on*.

- Générer une seconde d'un signal sinusoïdal de fréquence 200 Hz et d'amplitude 5 Volts échantillonné à 20 kHz. Représenter ce signal temporellement.
- En vous aidant des questions précédentes, générer une seconde d'une sinusoïde de fréquence 2 kHz, d'amplitude cinq Volts et perturbée par un bruit blanc gaussien de moyenne nulle et de variance trois Volts.
- Représenter temporellement le résultat et le comparer graphiquement au résultat de la question cinq.
- Représenter la densité spectrale de puissance de ce signal bruité.

Variables mathématiques (nécessaire)

A Amplitude

B Bande de fréquence

δ Distribution Dirac

E Energie

$e(t)$ Signal d'entrée dans le domaine temporel

f_0, f_1 Fréquence de signaux

ϕ Phase

ν Fréquence

h Réponse impulsionnelle

k Numéro des échantillons discrets en fréquence

n Numéro des échantillons discrets en temporel

$\pi_1(t), \pi_2(t)$ Distribution porte

t, τ Temps

t_0 Retard

T Période

T_e Période d'échantillonnage

TF Transformée de Fourier

$v(t)$ Signal temporel

$x(t)$ Signal temporel

$y(t)$ Signal de sortie

$X(\nu)$ Signal fréquentiel

Chapitre K

Introduction au langage Octave ou Scilab

K.1 Introduction

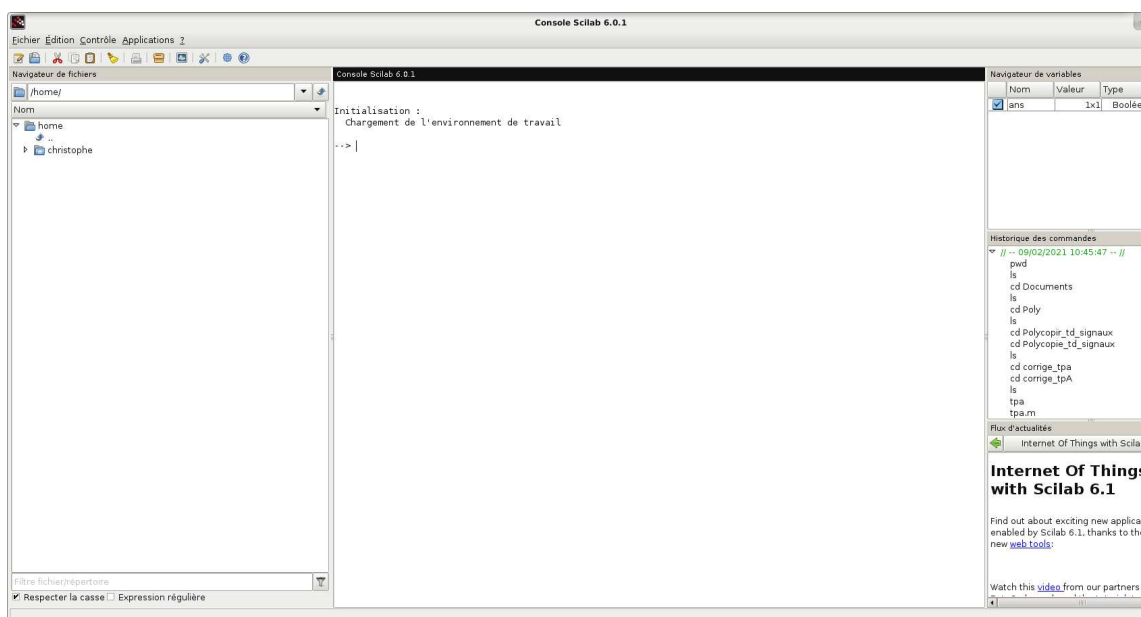
Le langage Matlab, commercialisé par la société Mathworks[®], et les dérivés libres Octave ou Scilab sont des logiciels de simulations numériques. Il s'agit de langage de programmation de haut niveau, interprété, ne nécessitant pas de compilation et permettent à l'aide de scripts et de commande simple d'interfacer des fonctions de bibliothèques mathématiques développées elles dans des langages plutôt bas niveau et pouvant demander un volume de code assez conséquent. Le logiciel de base est surtout apprécié dans le domaine de *l'ingénierie* car il permet de réaliser facilement des traitements mathématiques sur des tableaux d'entiers. Par contre, ces différents logiciels ne sont pas destinés à des calculs formels. Par rapport, à la version de base, ces logiciels se sont développés et proposent une longue liste de boîtes à outils et en particulier au niveau du traitement du signal. Par contre, il convient de se référer aux ouvrages techniques de base ou publication au niveau des définitions des conventions et des concepts pas forcément standard du point de vue de la littérature pour une utilisation avancée de ces fonctions. Ces différents langages sont très similaires dans leur écriture toutefois on peut noter qu'il existe quelques légères différences. Les fonctions présentées sont fonctionnelles pour les versions de Matlab[®], du logiciel Octave 4.4.1 ou encore de Scilab 6.0.1.

- Matlab[®] est un logiciel commercial de la société The Mathworks, Inc disponible sous la licence (éducation, recherche ou professionnelle) avec les droits et licences afférentes à l'adresse:
{<https://fr.mathworks.com/>}.
- GNU Octave est un logiciel libre disponible au téléchargement (sous la licence Creative Commons 4.0, licence internationale) au niveau du site:
{<https://www.gnu.org/software/octave/download.>}
- Scilab est disponible au téléchargement (sous licence GPL v2.0) à l'adresse Internet:
{<https://www.scilab.org/download/6.0.2>}.

Ce guide n'est pas une notice d'utilisation détaillée de ces logiciels mais peut-être vue comme une aide afin de débiter l'apprentissage de l'un ou l'autre de ces logiciels Mathworks[®], Octave ou Scilab.

K.2 Notions de base

Ces logiciels présentent une interface graphique permettant de réaliser de nombreuses opérations dans une session de travail (chargement, édition de fichiers, déclaration et lancement de fonctions. Un exemple est montré dans la figure suivante:



La fenêtre de lancement de commandes du programme permet ainsi de déclarer implicitement les variables. Toutefois, avec l'expérience, il est recommandé d'écrire un script, qui permet une initialisation mieux contrôlée des variables en mémoire. Ce script prend la forme d'un fichier texte dénommé souvent avec une extension `.m`. Sous Matlab ou Octave, le script se lance directement en l'appellant par le nom du fichier simplement dans l'interface de commande tandis que sous Scilab, celui-ci nécessite la fonction **exec(nom du fichier)**.

A) Aide du logiciel Matlab/Octave/Scilab

Il est possible d'accéder à la description d'une fonction en utilisant la commande **help** au niveau des différents logiciels dans l'interface de commande ce qui vous donnera rapidement l'information concernant la fonction ainsi que les arguments nécessaire.

B) Environnement de travail


La commande **pwd** permet d'afficher le répertoire courant tandis que **path** dans Matlab ou Octave permet d'afficher l'ensemble des chemins reconnus par défaut dans l'applications. Les quelques commandes en ligne permettent également de charger des fichiers de données (fonction **load**) ou de sauvegarder le résultat d'une simulation **save**:

Fonction	Description
pwd	Permet d'afficher le répertoire courant.
cd nom répertoire	Change de répertoire et permet d'aller dans le répertoire du nom spécifié.
clear all	Vide les variables en mémoire de l'espace de travail.
who	Affiche la liste des variables définies dans l'interface.
load	Charge un fichier de données au format adéquat.
save	Sauvegarde des données au format adéquat.

C) Fonctions mathématiques

a) Fonction standard mathématiques

Les fonctions courantes mathématiques permettent de réaliser des opérations dans l'espace des nombres entiers :

Fonction	Description
cos	fonction cosinus
sin	fonction sinus
log	fonction logarithme
log 10	fonction logarithme base 10
exp	fonction exponentielle
	fonction racine

Exemple d'initialisation simple de vecteurs ligne ou colonne ou matricielle

```
sqrt(-1)
ans = 0 + 1i
sin(pi/4)
ans = 0.70711
cos(pi/4)
ans = 0.70711
log(10)
ans = 2.3026
log10(10)
ans = 1
exp(1)
ans = 2.7183
sqrt(-1)
ans = 0 + 1i
```

Il est par contre nécessaire de nettoyer l'environnement de l'interface de commande où les programmes sont lancés. La fonction *clear all* permet d'effacer les données sauvegardées temporairement en mémoire tandis que l'autre fonction *close all* permet principalement de fermer les objets graphiques ou figures affichées. Par contre, malgré l'interactivité de l'interface de commande, il est recommandé de créer des scripts afin de lancer les différentes simulations souhaitées de manière à éviter des confusions entre les variables déclarées.

D) Définition de structure de données simple: vectorielle ou matricielle

L'un des avantages de ce logiciel est qu'il est possible de déclarer directement les variables au sein de l'interface à partir de symboles de ponctuation et de quelques nombres. En ce sens, il est retrouvé d'importantes similitudes avec le langage de programmation populaire python. Par exemple, la déclaration d'un vecteur composé de nombres différents se définit entre crochets []. Le symbole = est le symbole d'affectation qui permet d'affecter à une variable, des valeurs numériques. Comme, il est montré au niveau de l'exemple suivant, il est également possible à l'aide d'un symbole de ponctuation ; de définir non pas un vecteur mais une matrice en créant différents vecteurs. Le symbole : est un symbole d'itération, est aussi intéressant sur le plan de la syntaxe dans le sens où il permet aussi de créer facilement un vecteur de nombres entre une valeur minimale (min), un pas et une valeur maximale (max) ce qui donne la syntaxe suivante [*min* : *pas* : *max*]. Les valeurs numériques sont séparées par un pas de un, le plus courant comme le montre l'exemple ci après.

Fonction	Description
[]	symbole d'un vecteur ou d'une matrice
=	symbole d'affectation
:	symbole itération
;	symbole de fin d'opération
...	symbole permettant de continuer une commande sur plusieurs lignes

Exemple d'initialisation simple de vecteurs ligne ou colonne ou matricielle

```
A=[1 2 3 4 5 6 7 8 9 10 11 12];
A =
    1    2    3    4    5    6    7    8    9   10   11   12

A=[1 2 3 4; 5 6 7 8; 9 10 11 12];
A =
    1    2    3    4
    5    6    7    8
    9   10   11   12

A=[5:1:12];
A = 5    6    7    8    9   10   11   12
```

Les fonctions suivantes permettent de créer et d'initialiser facilement les tableaux de variable suivantes:

Fonction	Objet
ones	Créer un vecteur ou une matrice initialisée à la valeur unitaire.
zeros	Créer un vecteur ou une matrice initialisée à la valeur nulle.
eye	Créer une matrice dont les valeurs sur la diagonale sont unitaires.

Exemple d'initialisation simple de vecteurs ligne ou colonne ou matricielle

```
x=ones(1,5)
x =
```

```
1 1 1 1 1
```

```
x=zeros(5,1)
```

```
x =
```

```
0
0
0
0
0
```

```
x=zeros(5,5)
```

```
x =
```

```
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

```
x=eye(2,2)
```

```
x =
```

```
Diagonal Matrix
```

```
1 0
0 1
```

Bien évidemment, il est possible de combiner ces fonctions de base pour déclarer d'autres variables comme:

Exemple d'initialisation simple de vecteurs ligne ou colonne ou matricielle

```
(1/5)*ones(1,5)
```

```
ans =
```

```
0.20000 0.20000 0.20000 0.20000 0.20000
```

E) Usage d'une calculette

a) Fonction mathématiques usuelles

Croisons les doigts mais les différentes fonctions mathématiques sont implémentées et demeurent fonctionnelles. Les fonctions mathématiques trigonométriques ou circulaires usuelles c'est-à-dire \cos , \sin , \tan permettent de réaliser des opérations dans l'espace des nombres entiers \mathbb{N} ou l'espace complexe \mathbb{C} .¹ D'autres fonctions peut être un peu oubliées comme les fonctions logarithmes népérien (en base e) ou décimaux (logarithme dit vulgaire) peuvent être définies. Si l'on étend le domaine des fonctions trigonométriques, les fonctions hyperboliques sont également accessibles. Ainsi, le tableau ci-dessous résume les différentes fonctions mathématiques usuelles disponible dans le cadre du logiciel.

Fonction	Description
\cos	fonction cosinus
\sin	fonction sinus
\tan	fonction tangente
	fonction racine carrée
\log	logarithme népérien
\log_{10}	logarithme en base 10
\exp	fonction exponentielle
\cosh	fonction cosinus
\sinh	fonction sinus
\tanh	fonction tangente
atanh	fonction tangente

L'exemple suivant permet d'illustrer l'utilisation simple du logiciel comme une calculatrice standard mais en plus évoluée:

Exemple d'initialisation simple de vecteurs ligne ou colonne ou matricielle

```

sqrt(-1)
ans = 0 + 1i
sin(pi/4)
ans = 0.70711
cos(pi/4)
ans = 0.70711
log(10)
ans = 2.3026
log10(10)
ans = 1
exp(1)
ans = 2.7183
sqrt(-1)
ans = 0 + 1i

```

Dans le cas de l'utilisation du logiciel Scilab, il existe une différence notamment la variable π qui est déclarée à l'aide d'un (`%pi`):

¹Il faut faire attention et noter que les angles sont en radian par défaut.

```

nombre  $\pi$  Scilab
sin(%pi/4)
ans =

0.7071068

```

b) Calcul matriciel

L'écriture matricielle est importante dans le domaine de l'ingénierie et permet de d'écrire, de créer des modèles numériques dans les différents domaines des sciences appliquées de *l'ingénierie*. Un avantage est que le calcul vectoriel ou matriciel est simple pour des matrices de nombre. Les différentes fonctions de calcul matriciel sont ainsi résumées dans le tableau ci-dessous:

Fonction	Résultat
size	Dimension d'une matrice
length	Dimension vectorielle
det	Déterminant d'une matrice carrée
inv	Inverse d'une matrice
sum	Somme vectorielle

F) Notion d'opérateur

Ces logiciels permettent de réaliser les opérations suivantes sur des nombres, vecteurs (colonnes) ou ligne et matrice. Il faut noter qu'il est possible de réaliser ces opérations de manière vectorielle ou alors élément par élément:

Symbole	Opération	Symbole	Opération
+	Addition	-	Soustraction
*	Multiplication	/	Division
.*	Multiplication élément par élément	./	Division élément par élément
	Puissance		Division à gauche
.	Puissance	.\$	Division élément par élément
'	Transposition ou complexe conjugué	()	Permet de séparer une variable

G) Fonction statistique

a) Génération de séquence aléatoire discrète

La génération de séquence (pseudo) aléatoire à partir de lois de probabilité usuelle peut être réalisée par l'intermédiaire de différents générateurs de nombre aléatoire (GNA). Par exemple, la loi discrète uniforme est implémentée à l'aide de la fonction **rand** qui permet de générer une suite aléatoire composée de différents nombres ceux-ci étant distribués dans l'intervalle $[0; 1]$. Le tableau suivant résume les différentes séquences discrètes aléatoires possible:

Fonction	Description
rand	Génère une séquence aléatoire selon la loi discrète uniforme distribuée dans l'intervalle $[0 - 1]$.
randn	Génère une séquence aléatoire selon la loi discrète normale de moyenne nulle et de variance 1.

Exemple de variable aléatoire

```
%Prenons un nombre d\'echantillons (statistiques)
% compatible avec la loi des grands nombres
N=500000;
%
a =rand(1,N);
b =randn(1,N);
c= rande(1,N);
```

b) Statistiques

L'analyse de données ou de séquence ou de variable aléatoire sous forme de moment peut être réalisée par l'intermédiaire des fonctions suivantes:

Fonction	Résultat de la fonction
mean	Calcule l'espérance c'est-à-dire la moyenne de la séquence aléatoire.
mode	Cherche la valeur ayant la fréquence d'apparition la plus importante dans un jeu de données.
std	Calcule la déviation standard de la séquence aléatoire.
var	
cov	
moment	Permet d'estimer les différents moments statistiques.

La représentation graphique des variables aléatoires créés peut se faire à partir d'un histogramme. L'exemple suivant illustre les histogrammes des différentes loi de probabilité.

H) Structure de contrôle

Sans atteindre la complexité des langages objets, l'implémentation de structure d'algorithmie et de programmation de base sont accessibles de manière à automatiser des calculs répétitifs et de définir des blocs une série d'opérations à réaliser. Pour cela il est possible de définir des structures de compteur, de boucles, et de contrôle. Le tableau ci-dessous résume brièvement ces structures:

Instruction	Condition	Description
do	(until)	la commande définit une boucle itérative lancée au moins une fois.
for	(endfor)	la commande définit une boucle itérative avec une condition d'arrêt.
while	(endwhile)	la commande définit une boucle répétitive conditionnelle (tant que).
until	(condition)	la commande définit une boucle répétitive conditionnelle (jusqu'à).
break	(condition)	Définir une condition d'arrêt pour la fin de l'instruction.
continue	(condition)	Définir une condition pour la fin de l'instruction.

K.3 Script de commande ou fonction&procédures

A) Définition de script de commande

L'interface de lancement permet de lancer simplement des fonctions et des calculs simples. Toutefois de manière à préserver un historique des commandes dans l'interface de lancement, il est souvent nécessaire de regrouper l'ensemble des instructions dans un script de commande. Sous Matlab ou Octave, il prend la forme d'un fichier dénommé `.m` qui va regrouper l'ensemble des commandes tandis que sous Scilab il prends la forme d'un fichier `.sci`.

B) Déclaration de fonctions ou procédures

Ces logiciels permettent à l'utilisateur de créer ses propres fonctions en s'aidant des fonctions de base ou celles des boîtes à outils. Même si il existe une compatibilité avec les langages de bas niveau, il est par contre recommandé d'utiliser le langage et les instructions du logiciel en question. Il est possible de déclarer la fonction ou la procédure en ligne au niveau de l'environnement de travail.

Au niveau des logiciels Matlab ou Octave, le terme une fonction ou macro est utilisé tandis que sous Scilab c'est plutôt la dénomination procédure qui est utilisée. Par contre, le canevas de description de la fonction ou procédure pour ces différents logiciels qui nécessite un nom de fonction, des arguments en entrée et en sortie de la fonction.

Définition d'une fonction ou procédure

```
function <arguments_en_sortie>=<nom_de_la_fonction>(<arguments_en_entree>)
<instructions>
endfunction

function[res_{1},res_{2},...]=nom(var_{1},var_{2})
<instructions>
end
```

où var_1, var_2 sont les variables d'entrée et res_1, res_2 les variables de sortie de la fonction.

C) Structure d'affichage

L'un des avantages principaux de ces logiciels et qu'ils permettent d'afficher de manière très aisée les résultats de calcul.

a) Notion de figure

Une des grandes facilités apportées par ces logiciels par rapport au langage de programmation bas niveau et la facilité d'affichage des variables ou des résultats de calcul. Par l'intermédiaire de l'interface de commande, une fenêtre graphique peut être définie par la fonction **figure**. La figure suivante montre un exemple de création d'interface graphique pour les logiciels Octave ou Scilab.

Exemple de variable aléatoire

Le fait de pouvoir créer différentes figures peut-être réalisé par l'intermédiaire de ces logiciels. Pour cela, il suffit de créer de numéroter les figures par un numéro différent.

b) Différents tracés

Au niveau de l'analyse de données, différentes fonctions sont disponibles pour le tracé des données. La fonction **plot** permet de tracer une courbe dans un environnement à deux dimensions basée sur les valeurs données à la fonction. Cette fonction interpole les valeurs discrètes afin de présenter un affichage continu de ces valeurs discrètes. La fonction **stem** permet un affichage équivalent mais pour adaptées pour des valeurs discrètes.

Fonction	Argument	Description
plot	(x,y)	Réalise le tracé des valeurs continues selon l'axe x et y .
stem	(x,y)	Réalise le tracé des valeurs discrètes selon l'axe x et y .

Exemple de tracé pour un signal cosinusoidal

```
x=[0:1/20:5];
y=cos(2*pi*x);
figure(1)
plot(x,y)
figure(2)
stem(x,y)
```

La fonction **subplot** permet de réaliser des affichages avec des sous-figures.

D) Boîte à outil de traitement du signal

Les fonctions principales de traitement du signal sont disponibles par l'intermédiaire de boîte à outils complémentaire par rapport à la version de base des logiciels.

Au niveau d'octave, la commande **pkg list** permet d'afficher dans l'interface de travail, l'ensemble des boîtes à outils installées. Les fonctions de traitement de signal pour Octave sont partie d'un paquet, greffon d'Octave disponible sous Octave-Forge. Celle-ci en terme de dépendance logicielle demande une version d'Octave supérieure à la 3.8 ainsi qu'une version de la boîte à outils control de 2.4. Sans rentrer dans les détails, il existe différentes catégories de fonctions (GPLv3+, domaine publique):

- Génération des signaux complexes
- Réalisation de corrélations et des convolutions
- Implémentation de fonctions de filtrage et analyse de filtre
- Conception de filtre RIF et RII
- Transformée fréquentielles (fft, dct, dct2), de l'analyse spectrale

- Fenêtrage
- Rythme de l'échantillonnage et de l'interpolation

Au niveau de Scilab, la commande **getmodules** permet d'afficher les modules préinstallés de base. La version de base semble contenir le module de traitement du signal en standard et permet le calcul de corrélation, convolution, de filtre, d'identification, d'estimation spectrale ou de transformée fréquentielle.

Les fonctions nécessaires dont vous avez besoin pour les praticiens sont les suivants:

fft	(Variable, dimension)	Implémente une transformée de
dct		Implémente une transformée de
window (scilab) ou hamming (matlab, octave)		Génère une fenêtre de Hammin
xload (scilab) ou imread (matlab, octave)		Permet de charger une image

Index

- Affichage d'une image, 38
- Amplitude d'un signal, 15
- Analyse de signaux, 33
- Analyse spectrale, 19
- Autocorrélation, 11, 41

- Bruit, 15
- Bruit blanc gaussien, 41

- Compression d'image, 38
- Convertisseur numérique analogique, 15

- Densité spectrale de puissance, 11, 41
- Domaine fréquentiel, 7
- Domaine temporel, 7
- Dynamique, 15

- Echantillon fréquentiel, 23
- Echantillon temporel, 23
- Echantillonnage
 - Signal, 15
 - Temporel, 15
- Echantillonnage en fréquentiel, 19
- Echantillonnage en temporel, 19
- Echantillonnage numérique d'un signal, 19
- Egalité de Parseval, 7
- Energie d'un signal, 7, 11

- Filtrage, 11, 15, 41
- Filtre, 3
- Fréquence d'un signal, 15

- Générateur de nombres aléatoires, 51
- GNA, 51

- Image, 38

- Langage Octave, 45
- Language Matlab, 45
- Language Scilab, 45
- Linéarisation d'un signal cosinusoidal, 11
- Logiciel Matlab, 19, 33, 38, 41
- Logiciel Octave, 19, 33, 38, 41
- Loi normale, 27

- Notion de réponse impulsionnelle, 3
- Numérisation, 15

- Périodicité du spectre, 19
- Périodisation
 - fréquentielle, 3
 - temporelle, 3
- Phase, 11
- Probabilité de détection, 27
- Produit de convolution, 3, 7
 - Forme intégrale, 3
- Puissance d'un signal, 11

- Quantification, 15

- Réponse d'un système linéaire, 3
- Réponse impulsionnelle
 - Filtre, 7
- Réponse impulsionnelle d'un filtre, 3
- Résolution, 15
- Radar, 27
- Recouvrement, 15
- Recouvrement spectral, 3
- Repliement de spectre, 33
- Retard d'un signal, 11

- Signaux aléatoires, 27, 41
- Spectrogramme, 33
- Spectre analogique, 23
- Spectre d'amplitude, 15
- Spectre fréquentiel discret, 23
- Statistiques
 - Moyenne, 41
 - Variance, 41
- Système linéaire, 3

- Théorème de Shannon, 19
- Train de Dirac, 3
- Transformée de Fourier, 7, 11
- Transformée de Fourier Discrète
 - Définition matricielle, 23
 - Fenêtrage, 23
 - Finesse d'analyse, 23
 - Résolution fréquentielle, 23

- Transformée de Fourier discrète
 - Définition matricielle, 19
 - Exemple, 33
 - Signaux périodique, 33
- Transformée de Fourier rapide (FFT), 23
- Transformée en Cosinus discrète (2D), 38