

## TP N°1 : Génération et affichage des signaux continus

### Objectif de TP:

Ce TP est destiné à la génération des signaux utiles sous l'environnement Matlab et de déterminer le produit de convolution et la corrélation.

### Préparation théorique

1. Générer un signal triangulaire et rectangle à l'aide des signaux (échelon et rampe).
2. Tracer les signaux suivants et déterminer s'il s'agit de signaux à énergie finie, à puissance finie ou n'appartenant à aucune de ces deux catégories. La fonction  $u(t)$  étant la fonction de Heaviside (échelon).

$$x(t) = A \cdot \sin(t) \text{ pour } -\infty < t < +\infty$$

$$x(t) = A \cdot [u(t + \tau) - u(t - \tau)], \text{ avec } \tau > 0$$

$$x(t) = tu(t)$$

$$x(t) = e^{-\alpha|t|}, \text{ avec } \alpha > 0$$

3. Calculer le produit de convolution en se servant de la représentation graphique

$$f(t) * \text{rect}\left(\frac{t}{T}\right), \text{ avec } f(t) = 1 - \frac{t}{T}, \quad \text{rect}\left(\frac{t}{T_1}\right) * \text{rect}\left(\frac{t}{T_2}\right) \text{ avec } T_1 > T_2$$

4. Calculer la fonction d'auto corrélation d'un signal sinusoïdal  $x(t) = \cos(\omega t + \varphi)$ .

### En utilisant les commandes Matlab suivantes pour faire votre travail

**Plot** : Trace une représentation graphique.

**Grid** : affiche une grille.

**Title** : attribue un titre au graphique.

**Xlabel** : attribue un texte à l'axe des abscisses.

**Ylabel** : attribue un texte à l'axe des ordonnées.

**tripuls(T,w,s)** : elle génère une impulsion triangulaire de largeur w et de centre s.

**rectpuls(t,w)** : elle génère un signal rectangulaire de largeur w.

**pulstrain(t,d, func)** : génère un train d'impulsions, ou « func » peut être la fonction gauss, tri ou rect, avec d est le temps du peigne de Dirac

**Conv(x,y)** : elle fait la convolution de vecteur x avec le vecteur y.

**Subplot** : représente graphiquement dans un graphique donnée

**Axis** : indique les échelles des axes de coordonnées.

**Clf** : efface le graphique courant.

**Hold on/off** : permet/interdit la superposition des représentations graphiques.

**N= length(x)** : renvoie dans n la dimension du vecteur X.

**[n,m]= size (A)** : renvoie la dimension de la matrice A dans deux variables séparées m et n.  
**A= zeros(n)** : génère une matrice nulle de dimension nxn

## Manipulation

### I- Génération des signaux utiles

#### Programme1

```
clear all;
%.....
t= - 5: 0.1 :5 ;
for i=1:length(t)
    if t(i)<0
        u(i)=zeros(1) ;
    else
        u(i)=ones(1) ;
    end
end
figure (1)
plot(t,u) ;
axis([-5 5 -0.04 1.2]) ;
title('u(t) ');
xlabel('t') ;
ylabel('Amplitude') ;
```

#### Programme2

```
clear all;
%.....
t=-5: 0.1 : 5 ;
for j=1:length(t);
    if t(j)>0
        r(j)=t(j);
    else
        r(j)=0;
    end
end
figure (2)
plot(t,r);
axis([-5 5 -0.04 5]) ;
xlabel('Temps (sec) ');
ylabel('Amplitude');
```

1- Soit le programme1 :

- Tapez et tester le programme1.

- Définir le type de signal après l'exécution de programme1.
- Utiliser ce programme pour déterminer la fonction rectangulaire (Rect(t/4)).
- Ecrire un programme qui génère le signal Rect (t/4) (utiliser la commande "rectpuls").

2- Soit le programme 2 :

- Tapez et tester le programme2:

- Définir le type de signal après l'exécution de programme2
- Utiliser ce programme pour déterminer la fonction triangulaire (tri(t/2))
- Ecrire un programme qui génère le signal Tri(t/4) (utiliser la commande "tripuls").

### 3- Représentation de l'impulsion de Dirac : $\delta(t)$ Soit le programme suivant :

```
% Génération d'une impulsion
unité
% Génération d'un vecteur de -20
0 30
n = -20:30;
% Génération de l'impulsion
unité
u = [zeros(1,20) 1 zeros(1,30)];
% Tracer le signal généré
stem(n,u);
xlabel('Temps indexé en n');
ylabel('Amplitude');
title('impulsion unité');
axis([-20 30 0 1.2]);
```

- Exécuter ce programme.
- Faire les changements nécessaires au programme ci-dessus pour représenter les impulsions suivantes :

$$5 \delta(t), 4 \delta(t + 4), \text{ et } \delta(t-3).$$

### II-Produit de convolution de deux signaux analogiques

Le programme Matlab suivant calcule le produit de convolution (La commande : **conv**) de deux signaux rectangulaires :

- Ecrire un programme pour avoir un produit de convolution :
  - o Deux signaux rectangulaires identiques (Rect (t/4))
  - o Deux signaux rectangulaires, l'un de durée 4s et amplitude 1V et l'autre de durée 7s et d'amplitude 1V.
  - o Vérifier théoriquement les résultats obtenus.

### III- Corrélation

- Ecrivez un programme pour avoir la représentation graphique d'un signal sinusoïdal ( $x(t)=\cos(\omega t)$ ) ayant une fréquence  $f=20\text{Hz}$  et un vecteur du temps  $t=0 :0.1\text{s}$  avec un pas de 0.001.

#### - Autocorrélation

- Ecrire un programme Matlab qui calcule et trace l'autocorrélation (commande **xcorr**) de deux signaux sinusoïdaux.
- Vérifier théoriquement le résultat obtenu.