

## TP1 : Représentation de signaux sous Matlab

**But de TP :** L'objectif de ce TP est de:

- générer,
- visualiser quelques signaux analogiques en utilisant MATLAB.

Rappel :

### Représentation de quelques signaux par MATLAB

Soit quelques notions de base de Matlab que vous allez utiliser :

*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.

*pi* : c'est la valeur 3.14

### Représentation des signaux

#### 1.1. Représentation de l'impulsion de Dirac : $\delta(t)$

Créer un fichier M-File nommé "Signal\_Dirac" contenant le programme ci-dessous.

```
%Génération d'une impulsion unité
%Génération d'un vecteur de -10 0 20
n=-10:20;
%Génération de l'impulsion unité
u=[zeros(1,10) 1 zeros(1,20)];
%Tracer le signal généré
stem(n,u);
xlabel('Temps indexé en n');
ylabel('Amplitude');
title('impulsion unité');
axis([-10 20 0 1.2]);
```

1- Exécuter ce programme.

2- Faire les changements nécessaires au programme ci-dessus pour représenter les impulsions suivantes :

$$\delta(t), 2\delta(t + 2), \delta(t + 3) \text{ et } \delta(t - 1).$$

#### 1.2 Génération d'une impulsion rectangulaire

Créer un fichier M-File nommé « signal\_rectangle » et l'enregistrer.

```
t = -1:0.00001:1 ;
x1 = rectpuls(t,0.05) ;
plot(t,x1) ; axis([-0.1 0.1 -0.2 1.2])
; grid ;
xlabel('Temps (sec)');
ylabel('Amplitude');
title('impulsion rectangulaire ');
```

1- Exécuter ce programme et déterminer les caractéristiques.

2- Modifier Ce programme pour construire une impulsion rectangulaire de durée  $T = 0.08$  s et d'amplitude  $A = 2V$ .

### 1.3 Génération du sinus cardinal : $\text{sinc}(x)$

Soit la fonction  $y(x) = \text{sinc}(x) = \sin(x)/x$ .

On utilise l'expression logique  $(x == 0)$  pour exprimer : que la  $\lim y(x) = 1$  lorsque  $x \rightarrow 0$ .

Créer un fichier M-File nommé «*signal\_sinc*» et l'enregistrer

```
%Tracage de la fonction sinus cardinal
%Domaine des valeurs de la variable x
x=-4*pi:pi/100:4*pi;
%valeurs de la fonction
y=(x==0)+sin(x)./(x+(x==0));
%Tracage de la fonction sinus cardinal
plot(x,y)
grid
title('sinus cardinal y=sin(x)/x')
```

### 1.4 Génération de quelques signaux :

Soit les programmes ci- dessous :

#### Programme1

```
fs = 10000 ;
t = 0:1/fs:1.5 ;
y = square(2*pi*50*t) ;
plot(t,y), axis([0 0.1 -1.2 1.2]),
grid
xlabel('Temps (sec)') ;
ylabel('Amplitude') ;
title('signal .....')
```

#### Programme2

```
t = -1:0.00001:1 ;
x2 = tripuls(t,0.04) ;
plot(t,x2),
axis([-0.1 0.1 -0.2 1.2]),
grid ;
xlabel('Temps (sec)') ;
ylabel('Amplitude') ;
title('impulsion .....')
```

#### Programme 3

```
fs = 10000 ;
t = 0:1/fs:1.5 ;
y1 = sawtooth(2*pi*50*t) ;
plot(t,y1); axis([0 0.1 -1.2 1.2]);
grid;
xlabel('Temps (sec)') ;
ylabel('Amplitude') ;
title('signal ..... ')
```

#### Programme 4

```
x=0:0.001:4*pi ;
y=sin(x) ;
plot(x, y) ;
axis([0 4*pi -1.2 1.2]) ;
grid ;
```

1- Exécuter, déterminer le nom, le type de chaque signal ainsi que sa forme générale et terminer ce qui manque dans les programmes (commentaires, titre...).

2- Faire les changements nécessaires au **programme 4** pour représenter le signal :  $y(t) = 2 \cdot \sin(x)$