

TP1 – GENERATION ET REPRESENTATION DES SIGNAUX

OBJECTIFS DU TP

- 1- Présenter les commandes de base permettant de générer et représenter graphiquement les signaux sous MATLAB.
- 2- Générer et représenter quelques signaux connus en traitement du signal (sinus, sinus cardinal, échelon...).
- 3- Il est demandé de tester les exemples de ce TP. Des exercices d'application sont donnés en fin de TP.

PARTIE I :

COMMANDES DE BASE POUR LE GRAPHISME EN 2 DIMENSIONS SOUS MATLAB

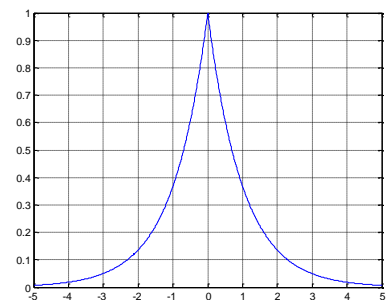
I-1- Représentation d'un seul graphe

Soit x et y deux vecteurs de même taille. **plot(x,y)** permet de tracer y en fonction de x .

Exemple :

Les commandes suivantes permettent de représenter le graphe du signal $s(t) = e^{-|t|}$ sur l'intervalle de temps $[-5,5]$ avec un pas de 0.01

```
>> t=-5:0.01:5;    % Définir l'intervalle de temps
>> s=exp(-abs(t)); % Définir les valeurs du signal
>> plot(t,s);grid   % Tracé du graphe avec une grille
```



I-2- Spécification des caractéristiques du graphe

Il est possible de modifier la couleur, le style de la ligne et le style des points utilisés pour tracer le graphe

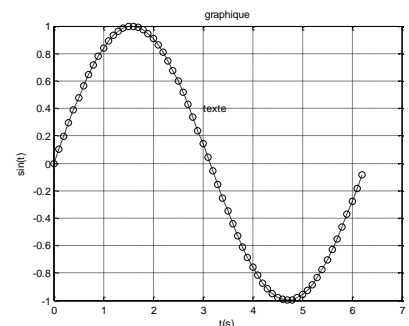
Exemple :

```
>> t=0:0.1:2*pi;
>> plot(t,sin(t),'ko--');grid
```

- **Couleur** : 'k' noir, 'b' bleu (couleur par défaut), 'r' rouge, 'g' vert, 'y' jaune
- **Style des points** : 'o' cercle, 's' carré, '*' étoile, '+' plus, 'x' croix
- **Style de la ligne** : '-' continue, '--' discontinue, '-.' tiret-point, ':' pointillé

Il est possible d'insérer des informations utiles sur le graphe.

```
>> title('graphique') % titre pour le graphe
>> xlabel('t(s)')      % titre pour l'axe des x
>> ylabel('sin(t)')    % titre pour l'axe des y
>> text(3,0.4,'texte') % texte à la position (3,0.4)
```



Il est possible de spécifier les limites des axes des x et des y

```
>> axis([-4 4 -2 2]) % fixer les limites des axes x : -4→4, y : -2→2
```

Cette commande est équivalente à :

```
>> xlim([-4 4])      % limites pour l'axe x : -4→4
>> ylim([-2 2])      % limites pour l'axe y : -2→2
```

Pour plus d'informations sur les commandes précédentes, taper dans la fenêtre de commande `>> doc plot`

I-3- Représentation de plusieurs graphes

I-3-1 Représentation sur des figures (fenêtres) différentes

Utiliser la commande **figure** à chaque fois qu'on veut tracer un graphe dans une fenêtre séparée.

Exemple :

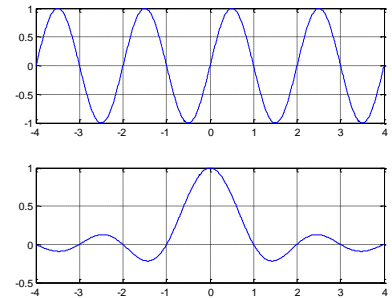
```
>> t=-4:0.01:4;
>> plot(t,sin(pi*t));grid % signal sinusoïdale de période  $T_o=2\pi/\omega=2\pi/\pi=2s$ 
>> figure
>> plot(t,sinc(t));grid % signal sinus cardinal  $\text{sinc}(t)=\sin(\pi t)/\pi t$  (pour  $t \neq 0$ )
```

I-3-2 Représentation sur la même figure mais sur des axes différents

La commande **subplot(n,m,p)** ou **subplot(nmp)** partage la figure en $n \times m$ parties, et se place dans la partie p (en comptant de gauche vers la droite, du haut vers le bas) pour représenter un graphe.

Exemple :

```
>> t=-4:0.01:4;
>> subplot(211);
>> plot(t,sin(pi*t));grid
>> subplot(212);
>> plot(t,sinc(t));grid
```

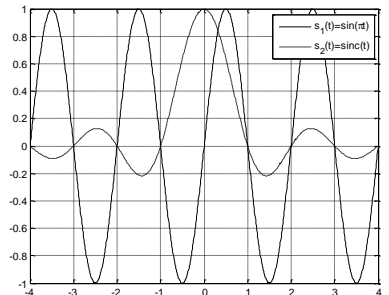


I-3-3 Représentation sur la même figure et sur les mêmes axes

La commande **plot(x1,y1, x2,y2,...)** permet de tracer **y1** en fonction de **x1**, **y2** en fonction de **x2** ... sur la même figure et sur les mêmes axes.

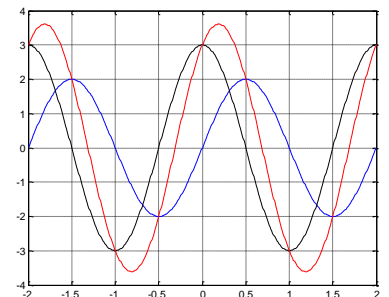
Exemple1 :

```
>> t=-4:0.01:4;
>> plot(t,sin(pi*t),'k', t,sinc(t),'k—');grid
>> legend('s_1(t)=sin(\pit)','s_2(t)=sinc(t)')
```



Exemple2 :

```
>> t=-2:0.01:2;
>> s1=2*sin(pi*t);
>> s2=3*cos(pi*t);
>> s3=s1+s2;
>> plot(t,s1,'b',t,s2,'k',t,s3,'r');grid
```



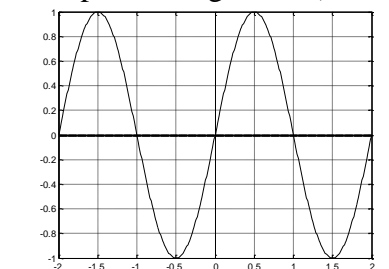
I-4- Quelques remarques

Remarque 1 :

Il est parfois utile de tracer l'axe des x (qui correspond à $y=0$) et l'axe des y (qui correspond à $x=0$) avec un trait continu. Ceci permet de distinguer facilement les parties positives des parties négatives (les 4 quadrants du plan).

Exemple :

```
>> t=-2:0.01:2;
>> plot(t,sin(pi*t),'k'); grid
>> hold on % garder le graphe précédent
>> plot(t,0,'k') % trace l'axe des x sur le graphe précédent
>> plot([0 0],[-1 1],'k') % trace l'axe des y à partir des 2 points à ses extrémités
```



Remarque 2 :

Dans certains cas, et afin de représenter un signal sur un intervalle $[a,b]$, le pas de temps n'est pas spécifié. Au lieu de cela, on fixe le nombre de valeurs linéairement espacées à utiliser sur cet intervalle. La commande **linspace(a,b,n)** permet de générer un ensemble de **n** valeurs équidistantes entre **a** et **b**, les valeurs **a** et **b** étant comprises.

Exemple :

```
>> t=linspace(-2,2,5)
t = -2   -1    0    1    2
```

PARTIE II :

GENERATION ET REPRESENTATION DE SIGNAUX CONNUS EN TRAITEMENT DU SIGNAL

II-1- Signaux disponibles sur MATLAB

Une liste de quelques signaux disponibles est obtenue à partir de la boîte à outils de traitement du signal (Signal Processing Toolbox) de MATLAB. Pour voir cette liste, taper **>> doc** dans la fenêtre de commande, la fenêtre de l'aide (Help) s'ouvre. Cliquer, dans la partie gauche de cette fenêtre, sur :

```
+ Signal Processing Toolbox
  + fx Functions
    + Waveform Generation
      - chirp
      - diric
      - gauspuls
      ...
      - vco
```

Liste de quelques signaux prédéfinis sur MATLAB
Cliquer pour avoir le détail de chaque signal

Nous présenterons dans la suite quelques uns des ces signaux.

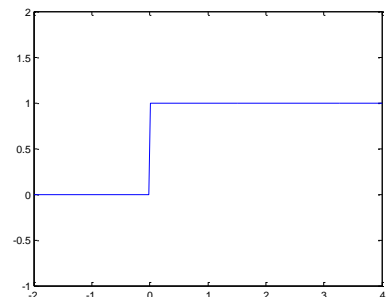
II-1-1 Signaux non périodiques

II-1-1-1- Echelon

La commande **heaviside(t)** génère un échelon unitaire.

Exemple :

```
>> t=-2:0.01:4;
>> plot(t,heaviside(t));ylim([-1 2])
```

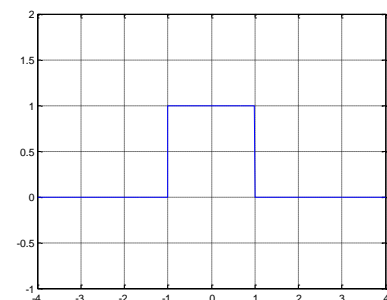


II-1-1-2- Impulsion rectangulaire ($rect_T$)

La commande **rectpuls(t,T)** génère une impulsion rectangulaire d'amplitude 1 et de largeur T centrée à 0.

Exemple :

```
>> t=-4:0.01:4;
>> plot(t, rectpuls(t,2));ylim([-1 2]);grid
```

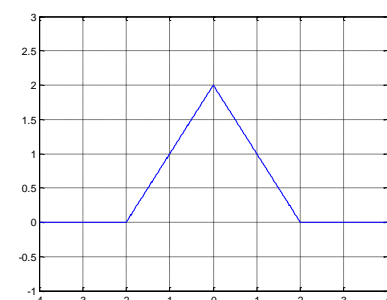


II-1-1-3- Impulsion triangulaire (tri_T)

La commande **tripuls(t,T)** génère une impulsion triangulaire de hauteur 1 et de base T centrée à 0.

Exemple :

```
>> t=-4:0.01:4;
>> plot(t, 2*tripuls(t,4));ylim([-1 3]);grid
```



II-1-2 Signaux périodiques

II-1-2-1- Signal carré

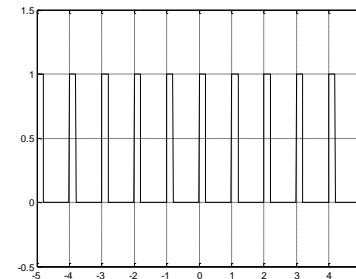
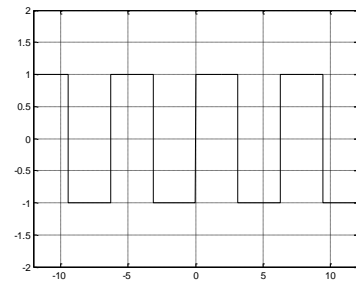
La commande **a*`square(w*t,d)`** permet de générer un signal carré d'amplitude **a** (variant entre **-a** et **a**), de période **$T_0=2\pi/w$** et la durée de la partie positive est **d%** de **T_0** .

Exemple 1 :

```
>> t=-12:0.01:12;  
>> s=square(t);  
>> plot(t,s,'k');axis([-12 12 -2 2]);grid
```

Exemple 2 :

```
>> t=-5:0.01:5;  
>> s=0.5*square(2*pi*t,20)+0.5;  
>> plot(t,s,'k');ylim([-0.5 1.5]);grid
```

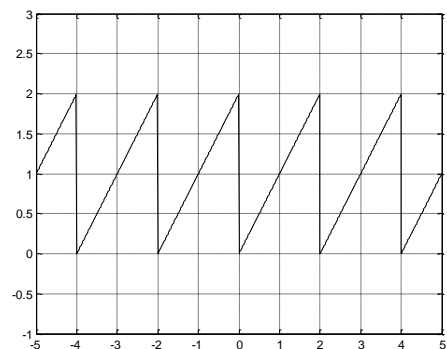


II-1-2-2- Signal en dents de scie

La commande **a*`sawtooth(w*t)`** permet de générer un signal en dents de scie d'amplitude **a** (variant entre **-a** et **a**) et de période **$T_0=2\pi/w$** .

Exemple 1 :

```
>> t=-5:0.01:5;  
>> s=sawtooth(pi*t)+1;  
>> plot(t,s,'k');ylim([-1 3]);grid
```



II-2- Signaux non disponibles sur MATLAB, à générer par l'utilisateur

II-2-1 Signaux non périodiques

Exemple :

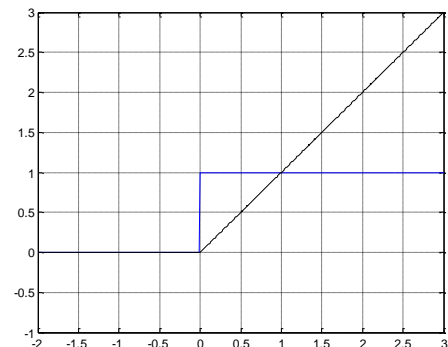
Le programme suivant permet de générer et représenter les signaux échelon et rampe unité :

```
t=-2:0.01:3;  
for i=1:length(t)  
    if t(i)<0  
        u(i)=0;r(i)=0;  
    else  
        u(i)=1;r(i)=t(i);  
    end  
end
```

```
plot(t,u,t,r,'k');ylim([-1 3]);grid
```

Les commandes suivantes permettent aussi de générer ces signaux d'une façon directe :

```
>> u=[zeros(1,200) ones(1,301)];  
>> r=[zeros(1,200) t(201:501)];
```



II-2-2 Signaux périodiques

Les étapes suivantes permettent de générer puis représenter un signal périodique donné sur un certain nombre de périodes. Ces étapes doivent être traduites en un programme MATLAB qui sera exécuté :

1- Définir le signal sur une période.

2- Définir le nombre de périodes sur lesquelles le signal doit être représenté.

- 3- Faire une périodisation du signal (créer des copies du signal, puis regrouper ces copies en un seul vecteur).
- 4- Définir l'intervalle de temps total de représentation.
- 5- Tracer le signal périodisé sur cet intervalle.

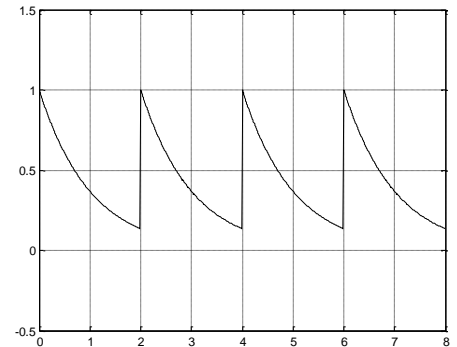
Exemple 1 :

On considère le signal suivant :

$$s(t) = e^{-t} \text{ pour } 0 \leq t < 2, \text{ périodique de période } T_0 = 2\text{s}$$

Le programme suivant permet de générer et représenter ce signal sur 4 périodes :

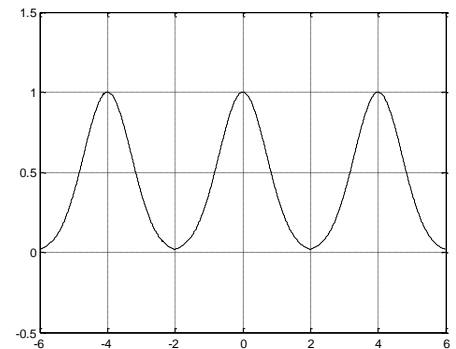
```
t1=0 : 0.01 : 1.99 ; % définir la 1ère période
y=exp(-t1) ; % définir les valeurs du signal sur la 1ère période
n=4 ; % définir le nombre de périodes
s=[ ] ; % vecteur vide [ ]
for i=1 : n } % périodisation du signal
s=[s y] ; % regrouper les copies du signal dans le vecteur s
end ;
t=0 : 0.01 : 2*n-0.01 ; % intervalle de temps total de représentation
plot(t,s,'k') ; ylim([-0.5 1.5]); grid % tracé du signal périodisé
```



Exemple 2 :

$$s(t) = e^{-t^2} \text{ pour } -2 \leq t < 2, \text{ périodique de période } T_0 = 4\text{s}, \text{ à représenter sur 3 périodes } [-6,6]$$

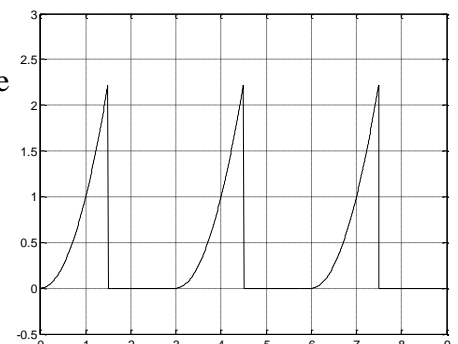
```
t1=-2 : 0.01 : 1.99 ; % 1ère période
y=exp(-t1.^2) ; % valeurs du signal sur la 1ère période (remarquer qu'on a utilisé .^, puissance élément par élément, au lieu de ^ car t1 est un vecteur)
n=3 ; % nombre de périodes
s=[ ] ; % vecteur vide [ ]
for i=1 : n } % périodisation du signal
s=[s y] ; % regrouper les copies du signal
end ; % dans le vecteur s
t=-6 : 0.01 : 5.99 ; % intervalle de temps total de représentation
plot(t,s,'k') ; ylim([-0.5 1.5]); grid % tracé du signal périodisé
```



Exemple 3 :

$$s(t) = \begin{cases} t^2 & \text{pour } 0 \leq t < 1.5 \\ 0 & \text{pour } 1.5 \leq t < 3 \end{cases}, \text{ périodique de période } T_0 = 3\text{s}, \text{ à représenter sur 3 périodes } [0,9]$$

```
t1=0 : 0.01 : 2.99 ; % 1ère période
y=[t1(1:150).^2, zeros(1,150)] ; % valeurs du signal sur la 1ère période
n=3 ; % nombre de périodes
s=[ ] ; % vecteur vide [ ]
for i=1 : n } % périodisation du signal
s=[s y] ; % regrouper les copies du signal dans le vecteur s
end ;
t=0 : 0.01 : 8.99 ; % intervalle de temps total de représentation
plot(t,s,'k') ; ylim([-0.5 3]); grid % tracé du signal périodisé
```



EXERCICES

EXERCICE 1

On considère le signal écrit sous MATLAB sous la forme suivante **sawtooth(4*pi*t/3) -1**

- 1- Ce signal est-il périodique ? Si oui, quelle est la sa période ?
- 2- Donner les commandes MATLAB permettant de représenter graphiquement ce signal sur l'intervalle de temps $[-3,3]$ avec un pas de 0.01. Mettre dans ce graphe :
 - Titre pour l'axe des x ' *temps (s)* '
 - Titre pour l'axe des y ' *s(t)* '
 - Titre pour le graphe ' *Impulsion triangulaire* '
 - Limites pour l'axe des y entre -2 et 2
 - Axe des x et axe des y

EXERCICE 2

1- On considère le signal écrit sous MATLAB sous la forme suivante :

$$2*\text{square}(\pi*t/2,25)+1$$

Représenter manuellement ce signal sur les 2 premières périodes positives

- 2- Donner les commandes MATLAB permettant de générer et représenter un signal en dents de scie de période $T_o = 3s$ et d'amplitude variant entre 0 et 3 sur les 2 premières périodes positives.

EXERCICE 3

Donner les commandes MATLAB permettant de générer puis représenter les graphes des signaux suivant sur l'intervalle de temps $[-4,4]$ avec un pas de 0.01 :

$$x(t) = 2\text{rect}_4(t) ; y(t) = 3\text{tri}_2(t) ; z(t) = e^{-|t|}.\sin(\pi t)$$

EXERCICE 4

On considère les signaux périodiques suivants :

$$x(t) = 2\sin(\pi t) ; y(t) = 3\cos\left(\frac{\pi}{2}t\right) ; z(t) = 0.5\sin(4\pi t)$$

- 1- Quelle est la période de chaque signal ?
- 2- Donner les commandes MATLAB permettant de faire la représentation graphique de ces signaux sur l'intervalle de temps $[0, 4]$ avec un pas de 0.01. Proposer 3 solutions :
 - Solution 1 : Tracer les graphes sur la même figure et sur les mêmes axes (graphes superposés)
 - Solution 2 : Tracer les graphes sur la même figure mais sur des axes différents
 - Solution 3 : Tracer les graphes sur des figures (fenêtres) différentes

EXERCICE 5

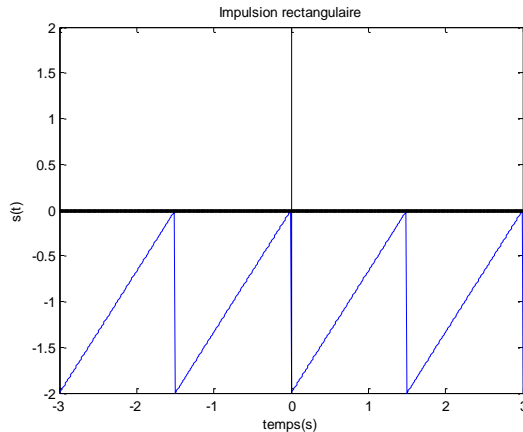
- 1- Donner les commandes MATLAB permettant de représenter graphiquement le signal périodique, de période $T_o = 2$, défini par : $s(t) = t$ pour $0 \leq t < 2$ sur 4 périodes (intervalle de temps $[-2, 6]$, pas de 0.01). Proposer 2 solutions.
- 2- Donner les commandes MATLAB permettant de représenter graphiquement le signal périodique, de période $T_o = 1$, défini par : $s(t) = \begin{cases} 1 & \text{pour } 0 \leq t < \frac{3}{4} \\ 0 & \text{pour } \frac{3}{4} \leq t < 1 \end{cases}$ sur les 3 premières périodes positives avec un pas de 0.01. Proposer 2 solutions.

SOLUTION DES EXERCICES

EXERCICE 1

1- Le signal **sawtooth(4*pi*t/3)-1** est périodique de période $T_0 = \frac{2\pi}{\omega} = \frac{2\pi}{\frac{4\pi}{3}} = \frac{3}{2} = 1.5 \text{ s}$

```
>> t=-3:0.01:3;
>> x=sawtooth(4*pi*t/3)-1;
>> plot(t,x)
>> xlabel('temps(s)')
>> ylabel('s(t)')
>> title('Impulsion rectangulaire')
>> ylim([-2 2])
>> hold on
>> plot(t,0,'k')
>> plot([0 0],[-2 2],'k')
```



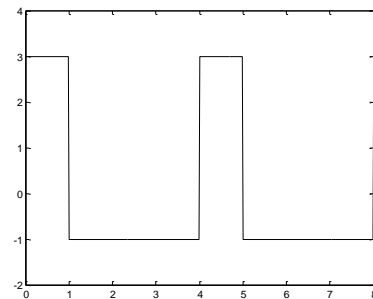
EXERCICE 2

1- Représentation du signal $2*\text{square}(\pi*t/2,25)+1$ sur les 2 premières périodes positives

Période du signal : $T_0 = 2\pi/\omega = 2\pi/(\pi/2) = 4 \text{ s}$

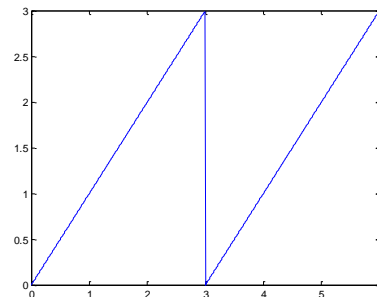
Amplitude du signal : $[-2 \ 2]+1 = [-1 \ 3]$

Durée de la partie positive : 25%



2- Génération et représentation d'un signal en dents de scie de période $T_0 = 3\text{s}$ et d'amplitude variant entre 0 et 3 sur les 2 premières périodes positives

```
>> t=0:0.01:6;
>> x=1.5*sawtooth(2*pi*t/3)+1.5;
>> plot(t,x)
```



EXERCICE 3

```
>> t=-4:0.01:4;
>> x=2*rectpuls(t,4);
>> y=3*tripuls(t,2);
>> z=exp(-abs(t)).*sin(pi*t);
>> subplot(311);plot(t,x);ylim([-1 3])
>> subplot(312);plot(t,y);ylim([-1 4])
>> subplot(313);plot(t,z);ylim([-1 1])
```

