# Travaux Pratiques - Série 4, TP° 1
# Séries de Fourier

## 1 Rappels

Un signal $u(t)$ périodique de période $T$ (de fréquence $f = \frac{1}{T}$ ou de pulsation $\omega = 2\pi f$) est décomposable en une somme de fonctions sinusoïdales :

$$u(t) = a_0 + \sum_{n=1}^{+\infty} [a_n \cos(n\omega t) + b_n \sin(n\omega t)] \tag{1}$$

Avec :
- $a_0$ : valeur moyenne du signal $u(t)$

$$a_0 = \frac{1}{T} \int_0^T u(t) \mathrm{d}t \tag{2}$$

- $a_n$ et $b_n$ : coefficients réels de la série de Fourier

$$a_n = \frac{2}{T} \int_0^T u(t) \cos(n\omega t) \mathrm{d}t \tag{3}$$

$$b_n = \frac{2}{T} \int_0^T u(t) \sin(n\omega t) \mathrm{d}t \tag{4}$$

Lorsque $n = 1$, on parle de la **composante fondamentale** (ou de l'**harmonique fondamental**) du signal ayant la pulsation $\omega$. Pour toute autre valeur de $n$ on parle de **l'harmonique de rang** $n$ ayant la pulsation $n\omega$.

La série de Fourier d'un signal periodique $u(t)$ peut être écrite également sous la forme suivante (cf. cours de maths, pour les détails) :

$$u(t) = a_0 + \sum_{n=1}^{+\infty} A_n \cos(n\omega t - \phi_n) \tag{5}$$

avec l'amplitude $A_n = \sqrt{a_n^2 + b_n^2}$ et la phase $\phi_n = \begin{cases} \arctan(b_n/a_n), & \text{si } a_n \geq 0 \\ \arctan(b_n/a_n) + \pi, & \text{si } a_n < 0, b_n \geq 0 \\ \arctan(b_n/a_n) - \pi, & \text{si } a_n < 0, b_n < 0. \end{cases}$

En Scilab, on peut obtenir directement la valeur de $\phi_n$ en utilisant la fonction `atan` avec deux arguments : `atan(bn, an)`.

## 2 Exercice

Nous nous intéresserons ici au signal carré représenté dans la Figure. 1.

On prendra les valeurs suivantes : $V = 5$ V et $f = 10$ Hz.

### 2.1 Représentation du signal

Ouvrir Scilab et créer un fichier SciNotes [1] permettant dans un premier temps de tracer le signal carré de la Figure 1 :

1. N'oubliez pas de faire des sauvegardes régulières de vos scripts sur une clé USB personnelle.

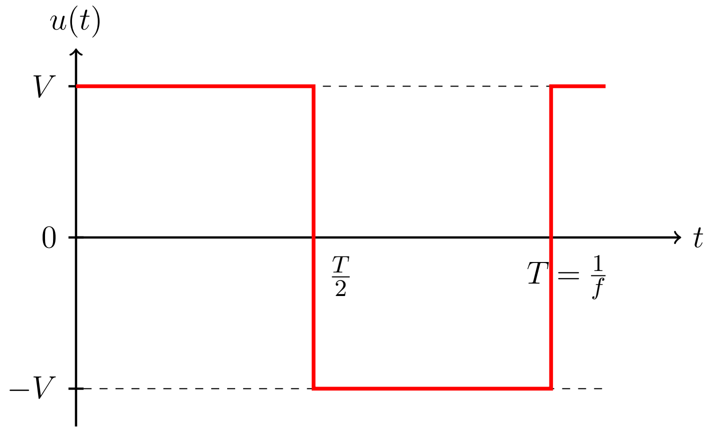


Figure 1 – Signal carré impair.

1. Déclarer dans votre fichier SciNotes les constantes que vous utiliserez par la suite : fréquence $f$ et amplitude $V$.
2. Définir le vecteur temps (on prendra par exemple un vecteur temps allant de 0 à $2 \times T$ en 400 points).
3. Définir les valeurs du signal $u(t)$ pour chaque point du vecteur temps. Vous pourrez utiliser soit une boucle *for* pour balayer chaque point du vecteur temps ou utiliser les fonctions dédiées (par exemple la fonction *ones()* ).
4. Tracer ensuite le signal $u(t)$ dans la sous-figure 1.
   *La figure qui devra être obtenue à la fin de ce TP sera divisée en 3 lignes et 3 colonnes, (en utilisant la fonction "subplot(a,b,c)").*

## 2.2 Calcul numérique des coefficients de Fourier

1. On commence par écrire les fonctions `an_num` et `bn_num` pour calculer numériquement les coefficients $a_n$ et $b_n$ en se basant sur les relations (2), (3) et (4). Ces fonctions doivent prendre en argument la fréquence, le temps, le signal, et le nombre $n$ d'harmoniques souhaitées et qui rendent en sortie le vecteur de $n$ premiers coefficients de Fourier. Attention, les vecteurs en Scilab commencent à 1 (et non pas à 0 comme en C).

   Voici le code pour calculer les coefficients $a_0$ et $a_n$.

```
function vectan = an_num(frequence, temps, signal, n)
    vectan = zeros(1,n+1)
    //On initialise a 0 tous les coefficients
    vectan(1) = inttrap(temps,signal)/(temps($)-temps(1))
    //On calcule a0, qui sera stocke dans la position 1 du vecteur
    for k = 1:n
vectan(k+1) = ...
2*inttrap(temps, signal.*cos(k*2*%pi*frequence*temps))/(temps($)-temps(1))
        //On calcule $a_k$, stocke dans la position k+1 du vecteur
    end
endfunction
```

   De manière analogue on écrit la fonction pour calculer les coefficients $b_n$.

```
function vectbn = bn_num(frequence, temps, signal, n)
 vectbn = zeros(1,n+1)
```

```
        for k = 1:n
                vectbn(k+1) = .......
    end
endfunction
```

2. Dans les sous-figures 7 et 8 tracer les histogrammes des coefficients $a_n$ et respectivement $b_n$ pour $n = 1, 2, 3, \ldots, 10$. On utilisera la fonction bar(x, y, largeur, 'couleur') où x est le vecteur des abscisses, y le vecteur des ordonnées, largeur (paramètre optionnel) est un nombre positif qui permet de choisir l'épaisseur des barres (un pourcentage de l'épaisseur maximum autorisé pour une barre) ; par défaut : épaisseur=0.8, i.e. 80%. Enfin, couleur (paramètre optionnel) spécifie la couleur de l'intérieur des barres (par défaut : 'blue').
3. Comparer les valeurs des coefficients $a_n$ et $b_n$ ? Qu'en déduisez vous ?
4. Quelle est l'amplitude du fondamental $A_1$ ?

Dans la sous-figure 9 tracer le diagramme spectral jusqu'à 10 harmoniques. On utilisera la fonction bar et on tracera pour chaque harmonique $A_n/A_1$.

## 2.3 Reconstruction du signal à l'aide de la série de Fourier

1. Avec Scilab, on peut programmer une fonction appelée *fourier* qui renvoie le signal reconstruit à partir de ses coefficients de Fourier. La fonction prend en argument le temps, la fréquence, et les vecteurs de coefficients $a_n$ et $b_n$.

```
function urecon = fourier(frequence, temps, an, bn)
    urecon=an(0+1)
    for i=1:size(an,2)-1
        urecon=urecon+an(i+1)*cos(i*2*%pi*frequence*temps)+......
endfunction
```

2. Calculer le signal reconstitué en utilisant les coefficients de Fourier calculés précédemment pour $n = 3$ harmoniques. Traces ce signal reconstitué dans la sous-figure 2. Tracer le signal reconstitué avec 5 harmoniques dans la sous-figure 3, avec 7 harmoniques dans la sous-figure 4, avec 10 harmoniques dans la sous-figure 5 et avec 100 harmoniques dans la sous-figure 6. Comparer visuellement les graphiques obtenus aux signal initial. Que remarquez-vous ?