TP3 - Traitement numérique du signal

Partie 1 : Diagrammes de Bode et Nyquist

Dans la première partie de ce TP, nous allons aborder le tracé des diagrammes de Bode et de Nyquist, qui permettent de représenter les fonctions de transfert complexes. Dans un premier temps, nous étudierons comment tracer ces deux diagrammes et comparerons leurs intérêts respectifs en les illustrant avec des exemples issus des différentes notions abordées en électronique et en automatique des systèmes linéaires.

1.1 – Diagramme de Bode

Dans ce chapitre, nous allons aborder brièvement le tracé des diagrammes de Bode de fonctions de transfert complexes. Nous écrivons d'abord l'expression de la fonction de transfert que l'on place dans une "function" qui prend des arguments en entrée (fréquence, valeurs des composants,...) et retourne le résultat.

```
function [H]=rc(f, R, C)
   H=1 ./(1+%i*2*%pi*R*C*f);
endfunction
```

Ensuite, on calcule son argument et son module, pour chaque valeur de la gamme de fréquence choisie.

Remarque : Le calcul du module d'un nombre complexe avec Scilab se fait grâce à la commande "abs(z)". Pour calculer l'argument d'un nombre complexe sous Scilab, il faut utiliser la commande "atan(imag(z),real(z))".

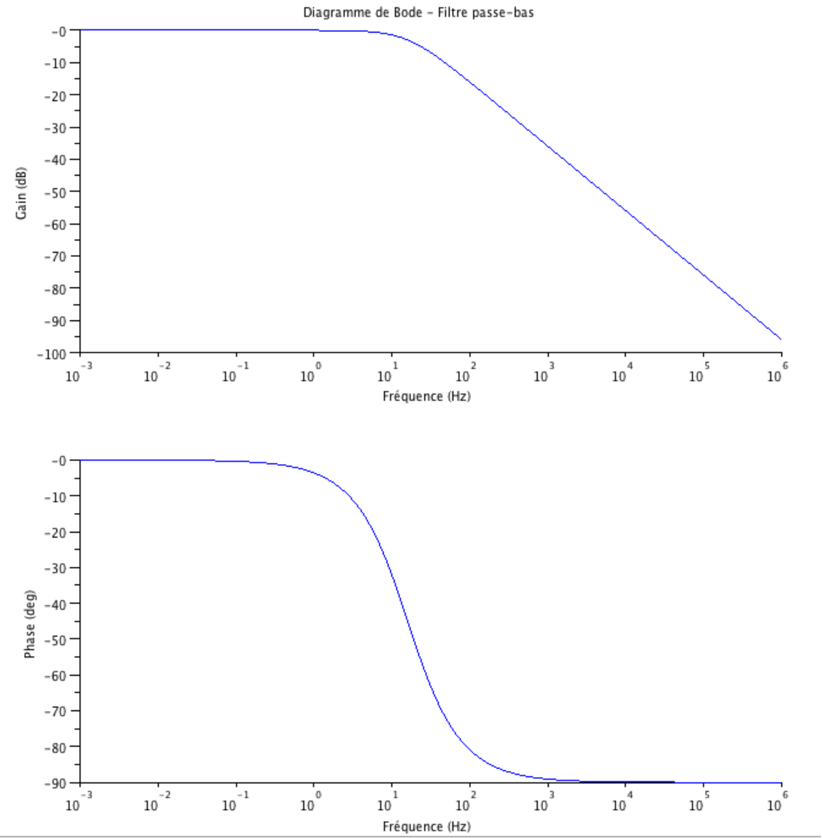
Dans un premier temps, nous allons nous intéresser uniquement à la variation du gain de la fonction de transfert en fonction de la fréquence. Nous choisirons comme valeur de la résistance et du condensateur 1kΩ et 10µF respectivement.

- **Tracer dans une même figure à l'aide de la fonction subplot les 3 tracés suivants :**
  - le gain en fonction de la fréquence en échelle linéaire pour les axes x et y.
    la fréquence variant de 0 à 1 kHz
  - le gain en fonction de la fréquence en échelle semi-logarithmique : gain en échelle linéaire, fréquence en échelle logarithmique.
    La fréquence variera dans ce cas entre 10^{-3} et 10^{6} Hz (utilisez la commande "logspace"). Pour un tracé en échelle logarithmique on doit utiliser la fonction plot2d (voir l'exemple ci-dessous).
  - le gain exprimé en dB en fonction de la fréquence en échelle semi-logarithmique avec la même gamme de fréquence que précédemment.
    Le gain en dB est calculé par la formule suivante : $20\log(|H|)$

Quelle représentation permet d'observer le plus clairement les variations du gain en fonction de la fréquence ?

Un **diagramme de Bode** est constitué de 2 figures en échelle semi-logarithmique représentant le gain en dB (le module) et la phase (l'argument) de la fonction de transfert en fonction de la fréquence. Le gain en dB et la phase peuvent être calculés simultanément grâce à la commande "**phasemag**".

```
fq=logspace(-3,6,1000);
[phase,gain]=phasemag(rc(fq,1e3,1e-5));
subplot(2,1,1);
plot2d(fq,gain,2,logflag="ln");
xtitle("Diagramme de Bode - Filtre passe-bas","Fréquence (Hz)","Gain (dB)");
subplot(2,1,2);
plot2d(fq,phase,2,logflag="ln");
xtitle("","Fréquence (Hz)","Phase (deg)");
```

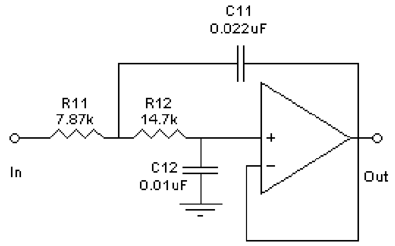


Questions de compréhension :

- Pour cette fonction de transfert, déterminer la fréquence de coupure.
- A cette fréquence de coupure, à quoi est égale cette fonction de transfert ?
- Quelle est la valeur du gain en linéaire et en dB ? Quelle est sa phase ?
- A quoi est égale la pente de la courbe de la fonction de transfert au-delà de cette fréquence de coupure ?
- A quoi correspond en linéaire une pente de 20dB par décade ?
- A quoi correspond en linéaire une pente de 6dB par octave ?

Tracez successivement les diagrammes de Bode de :

- un filtre passe-haut de fréquence de coupure fc=10 kHz
- un circuit RLC série dont la tension de sortie est prise aux bornes de R
  R=100 kΩ, C=1mF, L=1 H
- la cellule Sallen-Key représentée ci-dessous, de fonction de transfert :
  $H=Z_{C11}*Z_{C12} / [Z_{R11}*Z_{R12}+Z_{C12}*(Z_{R11}+Z_{R12})+Z_{C11}*Z_{C12}]$
  Quel est l'ordre de ce filtre ?



1.2 – Diagramme de Bode – fonction « bode » sous Scilab

Au-lieu de calculer nous-mêmes le module et l'argument, nous pouvons aussi utiliser la fonction « **bode** » déjà implémentée sous Scilab. Celle-ci trace le diagramme de Bode de tout système linéaire, entré sous la forme d'un polynôme en *p* (**transformée de Laplace**). Ceci s'écrit, pour un système du premier ordre :

```
K = 1; tau = 2;
p = poly(0, 'p');
h = syslin('c', K/(1+tau*p));
bode(h,1e-4,1e2) ;
```

Tracez les diagrammes de Bode de :

- un système du 1° ordre avec K=3 et τ=0.5 s
- un système du 2° ordre avec K=3, ξ=0.5, et ω_0=10
- deux systèmes du 1° ordre en série, avec K_1=4 / τ_1=0.5 s et K_2=1 / τ_2=2 s
- un système du 2° ordre avec K=15, ξ=0.5, et ω_0=10 en série avec un correcteur intégral 1/s

1.3 – Diagramme de Nyquist

Nous allons aborder maintenant le diagramme de Nyquist d'une fonction de transfert complexe, qui consiste à représenter la partie imaginaire de la fonction complexe en fonction de sa partie réelle pour chaque fréquence.

- Tracez successivement les diagrammes de Nyquist des fonctions de transfert précédentes :
  - un filtre passe-haut de fréquence de coupure fc=10 kHz

- un circuit RLC série dont la tension de sortie est prise aux bornes de R
  R=100 kΩ, C=1mF, L=1 H
- la cellule Sallen-Key précédente :

$$H=Z_{C11}*Z_{C12} / [Z_{R11}*Z_{R12}+Z_{C12}*(Z_{R11}+Z_{R12})+Z_{C11}*Z_{C12}]$$

Comme ci-dessus pour le diagramme de Bode, nous n'avons pas besoin de calculer nous-mêmes les parties réelles et imaginaires, car nous pouvons utiliser à nouveau une fonction déjà implémentée sous Scilab : « **nyquist** ». Celle-ci trace le diagramme de Nyquist de tout système linéaire, entré sous la forme d'un polynôme en *p* (t**ransformée de Laplace**). Ceci s'écrit, pour un système du premier ordre :

```
K = 1; tau = 2;
p = poly(0, 'p');
h = syslin('c', K/(1+tau*p));
nyquist(h,%f);// %t pour un diagramme de Nyquist symétrique (fréquences positives et négatives)
```

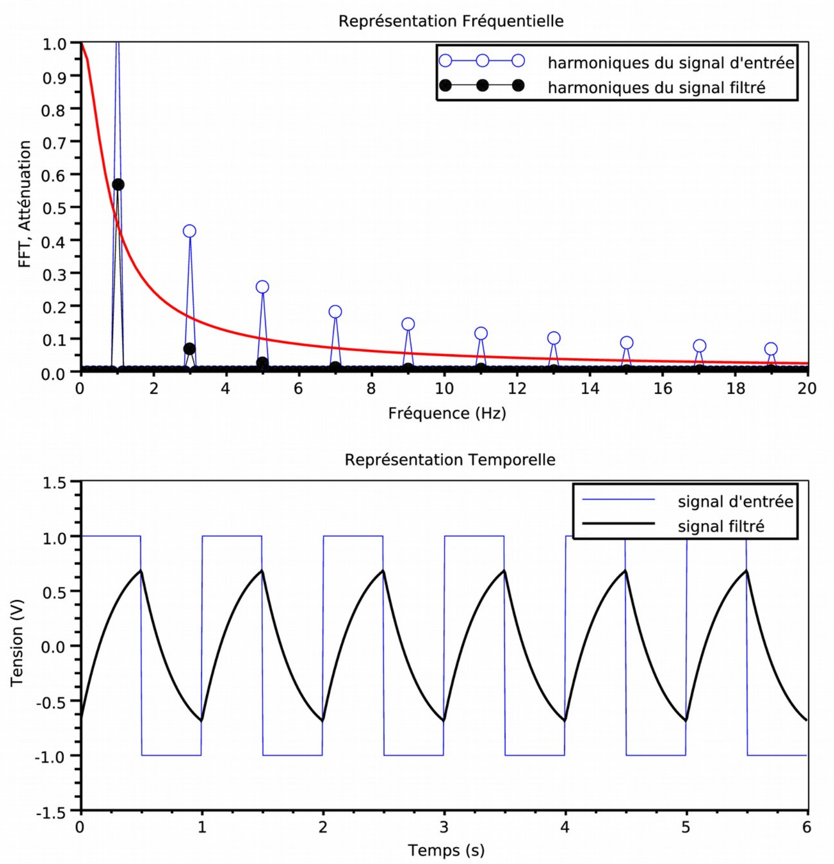
Tracez les diagrammes de Nyquist de :
- un système du 1° ordre avec K=3 et τ=0.5 s
- un système du 2° ordre avec K=3, ξ=0.5, et ω_0=10
- deux systèmes du 1° ordre en série, avec K_1=4 / τ_1=0.5 s et K_2=1 / τ_2=2 s
- un système du 2° ordre avec K=15, ξ=0.5, et ω_0=10 en série avec un correcteur intégral 1/s

Partie 2 : Notions de filtrage

Exercice de synthèse : Reconstruction par transformée de Fourier inverse d'un signal filtré

Cet exercice va vous permettre de faire la synthèse des différentes parties vues en TP OL3 depuis le début de l'année, à savoir :

- la création d'un signal périodique dans le domaine temporel
- le calcul de sa transformée de Fourier, c'est-à-dire sa représentation fréquentielle (spectre des fréquences)
- l'implémentation d'une fonction de transfert.
  Ici on se limitera à un filtre passe-bas et un filtre passe-haut du premier ordre. Vous tracerez son module en utilisant le même vecteur fréquence que la ***fft*** du signal ci-dessus. On règlera la fréquence de coupure de manière à ce qu'elle se situe à peu près à la fréquence fondamentale du signal carré.
- le calcul du signal de sortie filtré, obtenu par la multiplication du signal d'entrée par la fonction de transfert, pour chaque fréquence.
  On rappelle en effet que, dans le domaine fréquentiel, le signal de sortie après le filtre est égal au produit du signal d'entrée par la fonction de transfert : ***S(f)=H(f)*E(f)***.
- le calcul de la transformée de Fourier inverse du signal de sortie à l'aide de l'instruction ***fft(...,1)***. Cette étape permet de reconstruire le signal de sortie en représentation temporelle.
- enfin, la représentation de toutes ces données sur des graphes.



```
clear;clf;
// Filtre passe-bas
function [H]=lp(f, fc)
  H=1 ./(1+%i*f/fc);
endfunction
//
Npts=100;
Nmotifs=6;
N=Npts*Nmotifs;
Te=0.01;
t=Te*[0:N-1];
// 1 - Signal d'entrée temporel : Ve(t) //
// Signal 1 sinus
Ve=sin(2*%pi*1*t);
// 2 - TF du signal d'entrée : Ve(f)=TF(Ve(t)) //
TFVe=fft(Ve,-1);
// reconstruction du vecteur des fréquences
fr=1/(N*Te)*(0:N-1);
frt=[fr(1:N/2) flipdim(-fr(1:N/2),2)];
// 3 - Calcul du signal filtré : Vs(f)=Ve(f)*H(f) //
fc=0.5;
filt=lp(frt,fc);        // passe-bas
Vsfilt=TFVe.*filt;
// 4 - TF inverse du signal filtré : Vs(t)=TFI(Vs(f)) //
Vs=fft(Vsfilt,1);
// Affichage des graphiques //
subplot(2,1,1);
plot(fr(1:N/2),2/N*abs(TFVe(1:N/2)),'o-b');
plot(fr(1:N/2),2/N*abs(Vsfilt(1:N/2)),'.-k');
xset("thickness",3);
plot(fr(1:N/2),abs(filt(1:N/2)),'k');
xtitle("Représentation Fréquentielle","Fréquence (Hz)","FFT, Atténuation");
legend(['harmoniques du signal d''entrée';'harmoniques du signal filtré']);
a=gca();
a.data_bounds=[0 0;20 1];
subplot(2,1,2);
plot(t,Ve,'-b');
xset("thickness",3);
plot(t,Vs,'-k');
xtitle("Représentation Temporelle","Temps (s)","Tension (V)");
```

Questions :

- Pour un signal sinusoïdal de fréquence 1 Hz, filtré avec un passe-bas fc=0.1 Hz :
  Quelle est l'atténuation linéaire ?
  Quel est le déphasage ?
- Quelle fréquence de coupure fc faut-il pour atténuer le signal d'entrée d'un facteur 50 ? Et pour déphaser de $-\pi/4$?
- Dans le cas d'un signal constitué de 2 sinusoïdes de fréquence 1 et 16 Hz, quel type de filtre faut-il appliquer pour isoler la composante de plus haute fréquence ? Proposez une fréquence de coupure correcte.
- Pour un signal carré de fréquence 1 Hz, appliquez un filtre passe-bas de fréquences de coupure fc successives 0.1, 1 et 10 Hz, expliquez la forme du signal de sortie. On pourra calculer pour chaque fc, la constante de temps associée. Dans quel cas, observe-t-on un comportement intégrateur ?

Pour un signal triangulaire, appliquez un filtre passe-haut et réglez sa fc de manière à avoir un comportement dérivateur.