Université d'Ottawa Faculté de génie

École d'ingénierie et de technologie de l'information



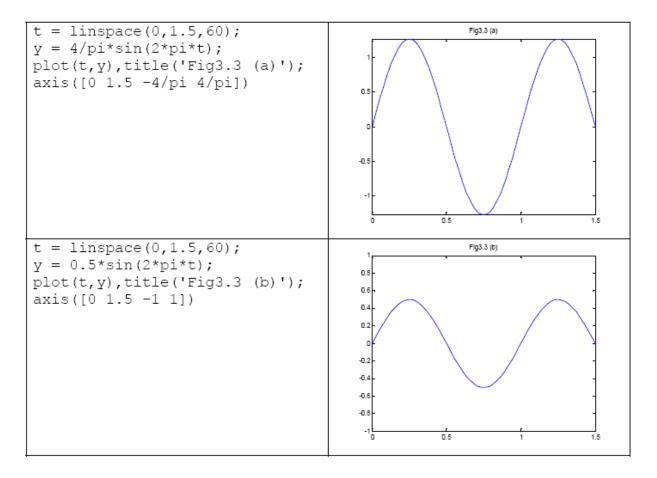
University of Ottawa Faculty of Engineering

School of Information Technology and Engineering

RESEAUX - LAB1

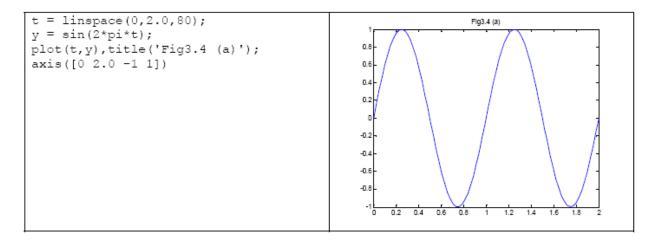
Matlab est un outil très important de simulation pour les étudiants en ingénierie. Durant cette expérimentation, les étudiants utiliserons Matlab pour générer, afficher et analyser des signaux à l'aide des transformés de Fourier.

1) Les commandes suivantes de MATLAB régénèrent la Figure 3.3 du livre de Stallings. Exécutez les scripts et affichez les ondes sinusoïdales résultantes.



```
t = linspace(0, 1.5, 60);
                                                            Flg3.3 (c)
y = \sin(2*pi*2*t);
plot(t,y),title('Fig3.3 (c)');
axis([0 1.5 -1 1])
                                             0.2
                                             -0.2
                                             -0.4
                                             -0.6
                                             -0.8
t = linspace(0, 1.5, 60);
                                                            Flg3.3 (d)
y = 0.5*sin(2*pi*2*t);
plot(t,y), title('Fig3.3 (d)');
axis([0 1.5 -1 1])
                                             0.2
                                             -0.2
                                             -0.4
                                             -0.6
```

2) Utiliser MATLAB pour régénérer la Figure 3.4 du livre de Stallings et affichez les figures résultantes.



- 3) Utiliser MATLAB pour régénérer 3.6 (a) et affichez les figures résultantes.
- 4) Testez les scripts de MATLAB suivants et affichez les figures résultantes. Expliquez la représentation en domaine fréquentiel résultant (remarquez que la résolution de la figure est basse).

```
t = 0:0.001:1.0;
\sin(2*pi*50*t)+1/3*\sin(2*pi*150*t)+1/5
*sin(2*pi*250*t)+1/7*sin(2*pi*350*t)+
1/9*sin(2*pi*450*t);
figure(1);
plot(1000*t(1:50),x(1:50))
title ('Signal containing frequencies
50, 150, 250, 350, and 450 Hz')
xlabel('time (milliseconds)')
% the following Matlab scripts
displays the frequency spectrum of a
signal.
Y = fft(x, 512);
Pyy = Y.* conj(Y) / 512;
f = 1000*(0: 256) / 512;
figure(2);
plot(f, Pyy(1: 257))
title('Frequency content of x')
xlabel('frequency (Hz)')
```

5) Les scripts de MATLAB suivants peuvent générer les représentations en domaine fréquentiel des fichiers *tarzan.wav* and *FIDDLE_B.wav*. Gênerez et soumettez la représentation en domaine fréquentiel de *CEG3185_lab1_high.wav* et *CEG3185_lab1_low.wav*.

```
[y, Fs, bits] = wavread('TARZAN.wav');
                                                                 Frequency content of x
wavplay(y);
                                                   60
Y = fft(y, 32768);
Pyy = Y.* conj(Y) / 32768;
                                                   50
f = Fs*(0: 16384)/32768;
plot(f, Pyy(1: 16385))
title('Frequency content of x')
xlabel('frequency (Hz)')
                                                   20
                                                                   frequency (Hz)
[y, Fs, bits] =
                                                                 Frequency content of x
wavread('FIDDLE_B.wav');
wavplay(y);
Y = fft(y, 4096);
Pyy = Y.* conj(Y) / 4096;
f = Fs*(0: 2048) / 4096;
plot(f, Pyy(1: 2049))
title('Frequency content of x')
xlabel('frequency (Hz)')
                                                                                10000
                                                                           8000
                                                                     6000
                                                                   frequency (Hz)
```

6) Les scripts de MATLAB suivants enlèvent un certain spectre de fréquences du fichier *tarzan.wav*. Voir la figure de droite. Veuillez enlever les fréquences du fichier *CEG3185_lab1_high.wav* qui sont plus hautes qu'un certain niveau que vous aurez choisi, transcrivez le fichier audio traité dans un nouveau fichier puis l'écouter et le comparer avec le fichier *CEG3185_lab1_high.wav*. Y a-t-il des différences?

```
SIZ=WAVREAD('TARZAN.wav', 'size');
                                                             Frequency content of x
[y, Fs, bits] = wavread('TARZAN.wav');
wavplay(y);
                                                30-
                                                25
Y = fft(y, 65536);
Pyy = Y.* conj(Y) / 65536;
                                                20
f=(1:65536);
plot(f, Pyy(1:65536));
                                                15
title('Frequency content of x');
                                                10
figure(2);
for n=8192:32769
    Y(n) = 0;
end
for n=32769:65536-8191
    Y(n) = 0;
Pyy = Y.* conj(Y) / 65536;
f=(1:65536);
plot(f, Pyy(1:65536));
title('Frequency content of x');
y=ifft(Y);
wavwrite(y,Fs,bits,'TARZAN2.wav');
```

