
Signal
Fourier
Problems
MATLAB
systems
data
z-Transform
Transform
signal
book
continuous-time
discrete-time
Control
FFT
Haar
course
wavelet
Systems
analysis
edition
Discrete-Time
system
text
DTFT
Analysis
Solutions
students
website
Sampling
State
Web
time
speech
transform
function
demos
coeficients
used
use
Representation
representation
various
material
digital
version
Section
covered
shown
downloaded
Models
Filters
electrical
third

Contents

Chapter 1

Fourier Series

The frequency spectrum is a complex-valued function of the frequency variable, and thus it is usually specified in terms of an amplitude spectrum and a phase spectrum [1]. The complex exponential form is given by:

$$\begin{aligned} x(t) &= \sum_{-\infty}^{\infty} X_n e^{jn\omega_0 t} \\ X_n &= \int_{-T/2}^{T/2} x(t) e^{-jn\omega_0 t}\, dt \end{aligned} \tag{1.1}$$

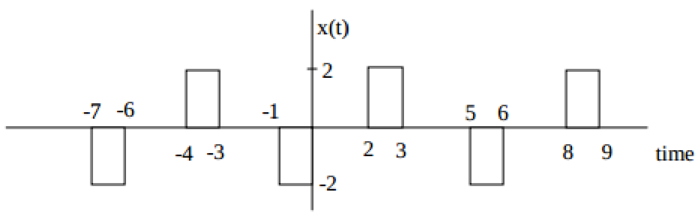
In the next exercises we will also be using the first and/or the second derivative of the previous expressions (**??**), and therefore we write them here explicitly:

$$\begin{aligned} \dot{x}(t) &= \sum_{-\infty}^{\infty} jn\omega_0 X_n e^{jn\omega_0 t} \\ jn\omega_0 X_n &= \int_{-T/2}^{T/2} \dot{x}(t) e^{-jn\omega_0 t}\, dt \end{aligned} \tag{1.2}$$

$$\ddot{x}(t) = \sum_{-\infty}^{\infty} -n^2\omega_0^2 X_n e^{jn\omega_0 t}$$
$$-n^2\omega_0^2 X_n = \int_{-T/2}^{T/2} \dot{x}(t) e^{-jn\omega_0 t}\, dt \tag{1.3}$$

Problem 1

For the following signal:



a) Find the Fourier series.

b) Plot the spectra versus frequency, $\omega = n\omega_0$.

Solution

The period of the shown signal is $T = 6$ and therefore $\omega_0 = \frac{2\pi}{T} = \frac{\pi}{3}$.

Taking the derivative of $x(t)$ we get:

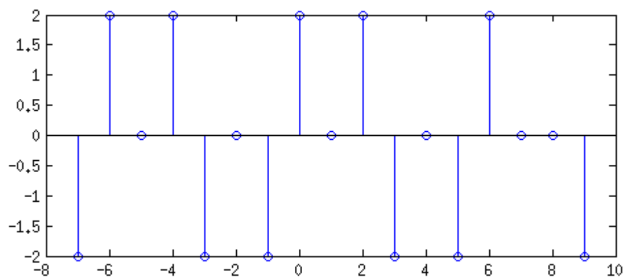


Figure 1.1: Derivative $\dot{x}$

The range t = [-3,3] contains one complete period of the signal. Usign (1.2) we have:

$$\dot{x}(t) = 2(-\delta(t+1) + \delta(t) + \delta(t-2) - \delta(t-3))$$

The Fourier coefficients are obtained with:

$$\begin{aligned}
jn\omega_0 X_n &= \frac{2}{6}\int_{-3}^{3}(-\delta(t+1) + \delta(t) + \delta(t-2) - \delta(t-3))e^{-jn\omega_0 t}\,dt \\
&= \frac{1}{3}(-e^{jn\omega_0} + 1 + e^{-2jn\omega_0} - e^{-3jn\omega_0}) \\
&= \frac{1}{3}[e^{\frac{jn\omega_0}{2}}(e^{\frac{-jn\omega_0}{2}} - e^{\frac{jn\omega_0}{2}}) - e^{\frac{-5jn\omega_0}{2}}(e^{\frac{-jn\omega_0}{2}} - e^{\frac{jn\omega_0}{2}})] \\
&= \frac{1}{3}[(e^{\frac{-jn\omega_0}{2}} - e^{\frac{jn\omega_0}{2}})(e^{-jn\omega_0}(e^{\frac{3jn\omega_0}{2}} - e^{\frac{-3jn\omega_0}{2}}))] \\
&= \frac{4j}{3}[\sin\frac{n\omega_0}{2}\sin\frac{3n\omega_0}{2}e^{-jn\omega_0}] \\
X_n &= \frac{-4j}{n\pi}[\sin(n\frac{\pi}{6})\sin(n\frac{\pi}{2})e^{-jn\frac{\pi}{3}}]
\end{aligned}$$

Next we use Matlab to plot the magnitude and phase of the spectra using the script given in [2]

Listing 1.1: Calculate and plot magnitude and phase of Xn

```
n=1:15;
Xn=-4*j./n/pi.*sin(pi*n/6).*sin(n*pi/2).*exp(-j*n*pi/3);
n=-15:-1;
X_n=-4*j./n/pi.*sin(pi*n/6).*sin(n*pi/2).*exp(-j*n*pi/3);
Xn=[X_n 0 Xn];
n=-15:15;
```

```
subplot(211),stem(n,abs(Xn));
 title('|X_n|')
subplot(212),stem(n,angle(Xn))
 title('angle(X_n) in rad')
```

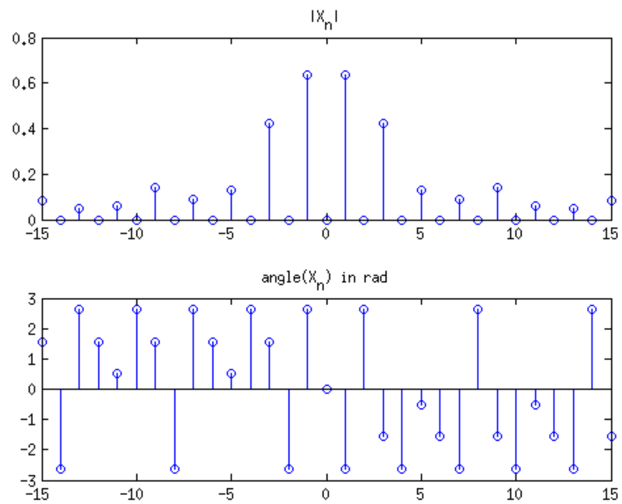


Figure 1.2: Magnitude and Angle X_n

We then plot the approximation of the function using its Fourier coefficients [2].

Listing 1.2: Approximation of x(t) with Fourier coefficients

```
function [x,t] = fapprox(N,T)
    t = -1.5*T:T/1000:1.5*T;
    w0 = 2*pi/T;
    X0 = 0;
    x = X0*ones(1,length(t)); % dc component
    for n=1:N,
     Xn = -4*j/n/pi*sin(pi*n/6)*sin(n*pi/2)*exp(-j*n*pi/3);
     X_n = conj(Xn);
     x = x + Xn*exp(j*n*w0*t) + X_n*exp(-j*n*w0*t);
    end
end
```

We do this for N=5 and N=50:

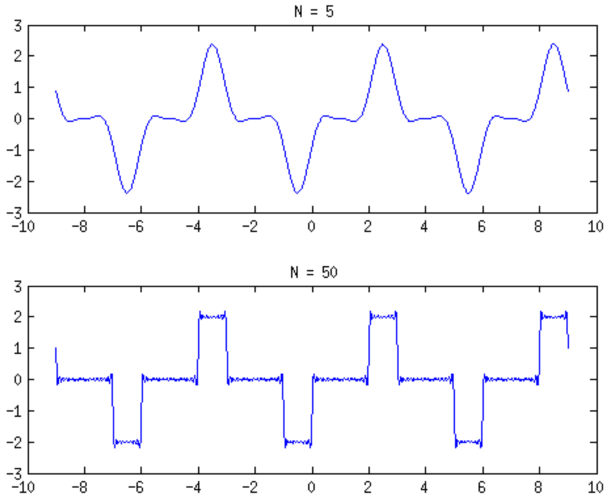
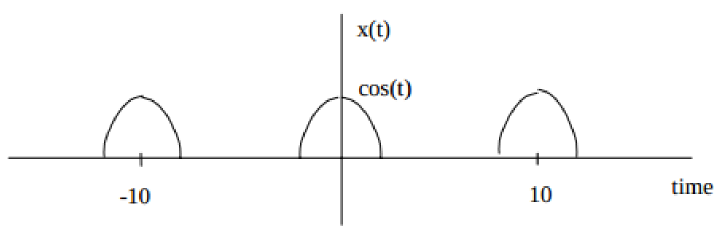


Figure 1.3: Approximation of x(t) by Xn

As we can see, the larger the N the more close to the original function we get. However, as a consequence of the Gibbs effect, we can't say that it'll be equal.

Problem 2

Repeat problem 1 for the following signal:



Solution

The period of the shown signal is $T = 10$ and therefore $\omega_0 = \frac{2\pi}{T} = \frac{\pi}{5}$.

If we take the first and second derivative of $x(t)$ we get:

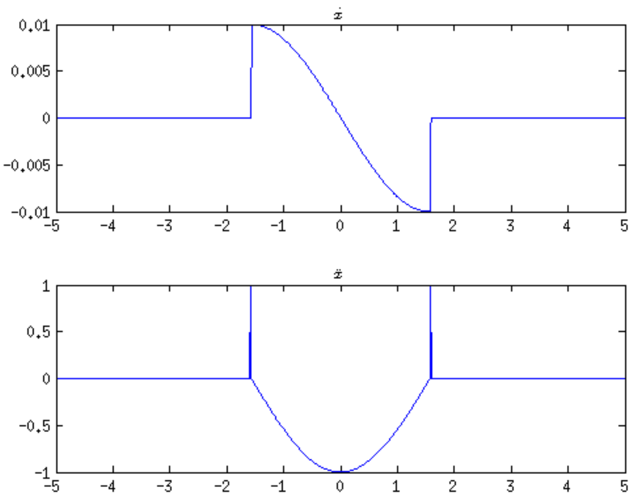


Figure 1.4: Derivative $\dot{x}$

The range t = [-5,5] contains one complete period of the signal. Applying (1.2) we have:

$$\ddot{x}(t) = -x(t) + \delta(t + \pi/2) + \delta(t - \pi/2)$$
$$\sum_{-\infty}^{\infty} -n^2\omega_0^2 X_n e^{jn\omega_0 t} = \sum_{-\infty}^{\infty} X_n e^{jn\omega_0 t} + \delta(t + \pi/2) + \delta(t - \pi/2)$$
$$\sum_{-\infty}^{\infty} (1 - n^2\omega_0^2) X_n e^{jn\omega_0 t} = \delta(t + \pi/2) + \delta(t - \pi/2)$$

Whe can now obtain X_n with:

$$
\begin{aligned}
(1 - n^2\omega_0^2)X_n &= \frac{1}{T}\int_{-5}^{5} \delta(t + \pi/2) + \delta(t - \pi/2)e^{-jn\omega_0 t}\, dt \\
&= \frac{1}{T}(e^{jn\frac{\omega_0}{2}} + e^{-jn\frac{\omega_0}{2}}) \\
X_n &= \frac{1}{5(1 - \frac{n^2\pi^2}{25})}\cos(\frac{n\pi^2}{10})
\end{aligned}
$$

Next we use Matlab to plot the magnitude and phase of the spectra using the script given in [2]

Listing 1.3: Calculate and plot magnitude and phase of Xn

```
n=-10:10;
Xn=cos(pi/2*n*w0)/5./(1-(n*w0).^2);
subplot(121),stem(n,abs(Xn))
 title('|X_n|')
subplot(122),stem(n,angle(Xn))
 title('angle(X_n) in rad')
```

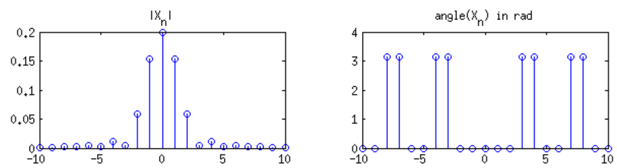


Figure 1.5: Magnitude and Angle X_n

We then plot the approximation of the function using its Fourier coefficients [2].

Listing 1.4: Approximation of x(t) with Fourier coefficients

```
function [x,t] = fapprox2(N,T)
    w0 = 2*pi/T;
    t = -1.5*T:T/1000:1.5*T;
    c0 = 1/5;
    x = c0*ones(1,length(t)); % dc component
    for n=1:N,
     cn = cos(pi/2*n*w0)/5/(1-(n*w0)^2);
     c_n = cn;
     x = x + cn*exp(j*n*w0*t) + c_n*exp(-j*n*w0*t);
    end
    plot(t,x)
    title([' N = ',num2str(N)])
end
```

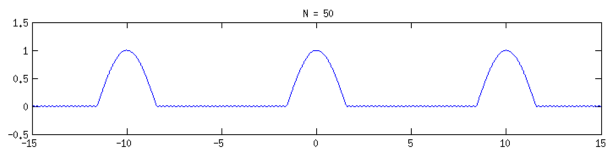


Figure 1.6: Approximation of x(t) by Xn

Problem 3

Compute the Fourier series for the following signals:

1. $x(t) = 2 + 4\cos(50t + \pi/2) + 12\cos(100t - \pi/3)$

Solution

Lets recall from [1] that:

$$x(t) = a_0 + \sum_{k=1}^{\infty} A_k \cos(k\omega_0 t + \theta_k) \quad -\infty < t < \infty$$

And the equivalent coefficients between the trigonometric series and the exponential form are:

$$\begin{aligned} X_0 &= a_0 \\ |X_n| &= \frac{1}{2}A_k \ , k = 1, 2, .. \\ \angle X_n &= \theta_k \ , k = 1, 2, .. \end{aligned} \tag{1.4}$$

From (1.4) we can immediately calculate the X_n coefficients:

Listing 1.5: Plot Magnitude and Angle of X_n

```
fe = [-100 -50 0 50 100];
Ae = [6 2 2 2 6];
pe = [pi/3 -pi/2 0 pi/2 -pi/3];

figure(1);
subplot(2,1,1), stem(fe,Ae);
grid on;
```

```
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|X_n|', 'fontweight', 'bold');

subplot(2,1,2), stem(fe,pe);
grid on;
xlabel('Frec [rad/s]');
ylabel('Phase');
title('$\angle x$','interpreter','latex')
```

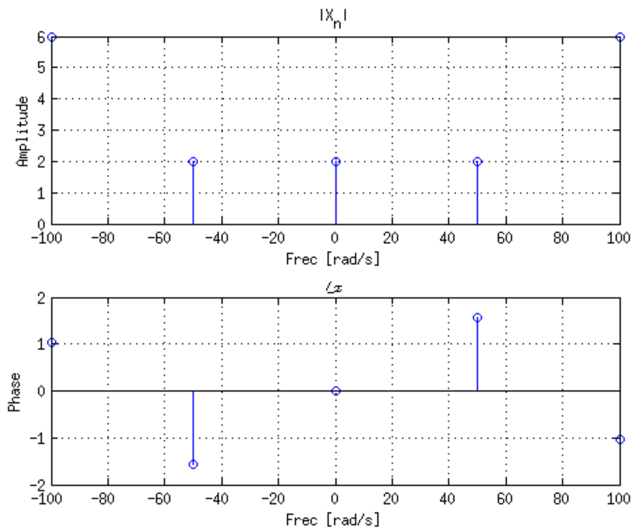


Figure 1.7: Magnitude and Angle of X_n

2. $x(t) = 4\cos(2\pi(1000)t)\cos(2\pi(750000)t)$

Solution

$$
\begin{aligned}
x(t) &= 4\left(\frac{e^{j2\pi(1000)t} + e^{-j2\pi(1000)t}}{2}\right)\left(\frac{e^{j2\pi(750000)t} + e^{-j2\pi(750000)t}}{2}\right) \\
&= e^{j2\pi(751000)t} + e^{j2\pi(-749000)t} + e^{j2\pi(749000)t} + e^{j2\pi(-751000)t}
\end{aligned}
$$

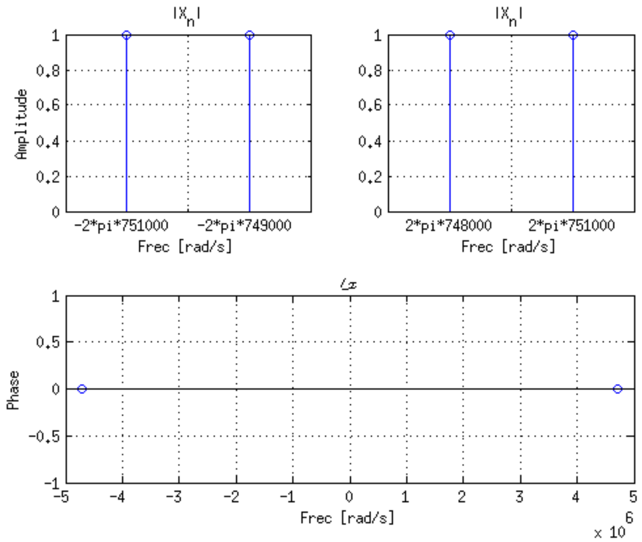
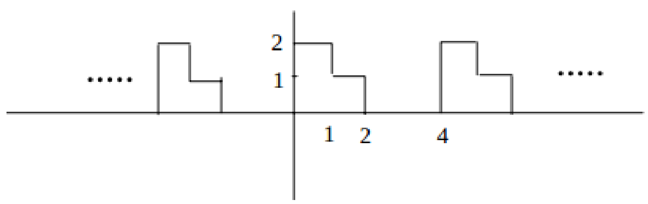


Figure 1.8: Magnitude and Angle of X_n

3. The function:



Solution

The period of the shown signal is $T = 4$ and therefore $\omega_0 = \frac{2\pi}{T} = \frac{\pi}{2}$.

Taking the derivative of the function we get:

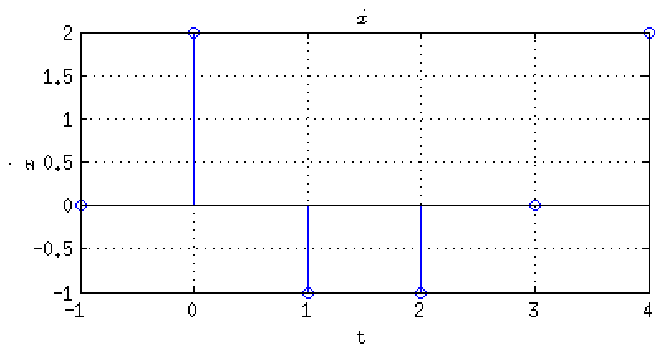


Figure 1.9: Derivative $\dot{x}$

In the range $[-1, 3]$ we have:

$$\dot{x}(t) = 2\delta(t) - \delta(t-1) - \delta(t-2)$$

Applying (1.2) we have:

$$\begin{aligned} jn\omega_0 X_n &= \frac{1}{T}\int_{-T/2}^{T/2} [2\delta(t) - \delta(t-1) - \delta(t-2)] e^{-jn\omega_0 t}\, dt \\ &= \frac{1}{4}[2 - e^{-jn\frac{\pi}{2}} - e^{nj\pi}] \end{aligned}$$

We can use the following properties:

$$\begin{aligned} e^{-jn\pi} &= (e^{-j\pi})^n = [\cos(\pi) - j\sin(\pi)]^n = (-1)^n \\ e^{-jn\frac{\pi}{2}} &= (e^{-j\frac{\pi}{2}})^n = [\cos(\frac{\pi}{2}) - j\sin(\frac{\pi}{2})]^n = (-j)^n \end{aligned}$$

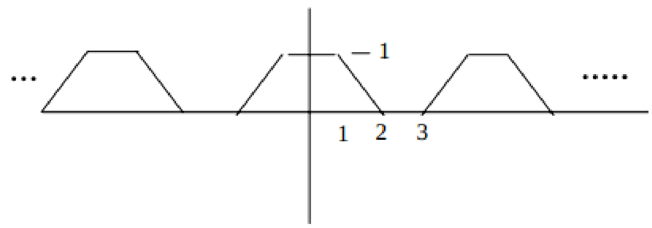
And reduce the equation to:

$$X_n = \frac{1}{2jn\pi}[2 - (-1)^n - (-j)^n]$$

where

$$X_0 = \frac{1}{4}\int_{-1}^{3} x(t)\, dt = \frac{3}{4}$$

4. The function:



Solution

The period of the shown signal is $T = 5$ and therefore $\omega_0 = \frac{2\pi}{T} = \frac{2\pi}{5}$.

Taking the derivative of the function we get:

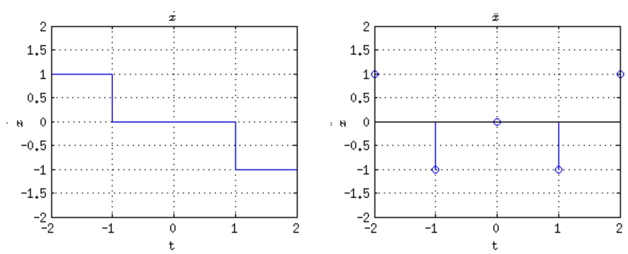


Figure 1.10: First and Second Derivatives $\dot{x}$ $\ddot{x}$

In the range $[-2, 2]$ we have:

$$\ddot{x}(t) = \delta(t+2) - \delta(t+1) - \delta(t-1) + \delta(t-2)$$

Applying (1.3) we have:

$$
\begin{aligned}
-n^2\omega_0^2 X_n &= \frac{1}{T}\int_{-T/2}^{T/2} [\delta(t+2) - \delta(t+1) - \delta(t-1) + \delta(t-2)] e^{-jn\omega_0 t}\, dt \\
&= \frac{1}{5}[(e^{2jn\omega_0} + e^{-2jn\omega_0}) - (e^{jn\omega_0} + e^{-jn\omega_0})] \\
X_n &= \frac{2}{5n^2\omega_0^2}[\cos(n\omega_0) - \cos(2n\omega_0)]
\end{aligned}
$$

We can calculate the dc component by finding the area of the trapezoid:

$$
\begin{aligned}
X_0 &= \frac{1}{T}\int_{-T/2}^{T/2} x(t)\, dt \\
&= \frac{1}{5}\frac{B.b}{2}h = \frac{3}{5}
\end{aligned}
$$

Problem 4

For the signals given in Problem 3c) and 3d), use Matlab to plot the truncated Fourier series for N = 3, N = 10 and N= 40. (Use subplot to save paper).

Solution

1. For problem 3c:

Listing 1.6: Approximation of x(t) with Fourier coefficients

```
function [x,t] = fapprox3(N,T)
    t = -1.5*T:T/1000:1.5*T;
    w0 = 2*pi/T;
    X0 = 3/4;

    n_p = [1:N];
    n_n = [-N:-1];
    Xn = (2 - (-1).^n_p - (-j).^n_p) ./ (2*j*pi.*n_p);
    X_n = (2 - (-1).^n_n - (-j).^n_n) ./ (2*j*pi.*n_n);

    Xn = [X_n X0 Xn];
    n = [n_n 0 n_p];
```

```

    x = Xn*exp(j*w0*n'*t);
    x = real(x);
end
```

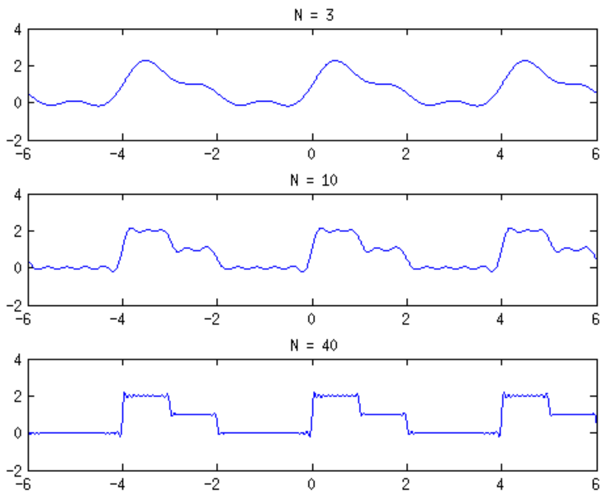


Figure 1.11: Approximation of x(t) by Xn for N=[3,10,30]

2. For problem 3d:

Listing 1.7: Approximation of x(t) with Fourier coefficients

```
function [x,t] = fapprox4(N,T)
    t = -1.5*T:T/1000:1.5*T;
    w0 = 2*pi/T;
    X0 = 3/5;

    n_p = [1:N];
    n_n = [-N:-1];
    Xn = (cos(n_p * w0) - cos(2*n_p * w0) ) * 2 ./ (5*w0
        ^2*n_p.^2);
    X_n = (cos(n_n * w0) - cos(2*n_n * w0) ) * 2 ./ (5*w0
        ^2*n_n.^2);


    Xn = [X_n X0 Xn];
    n = [n_n 0 n_p];

    x = Xn*exp(j*w0*n'*t);
    x = real(x);
end
```

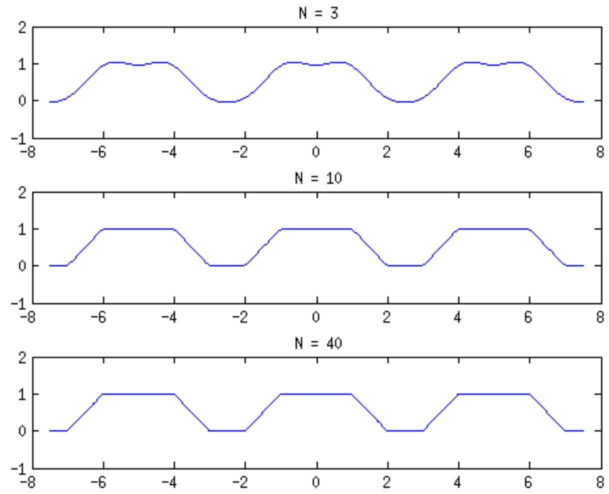
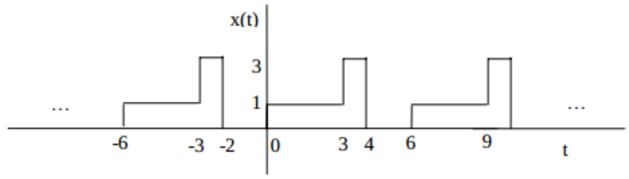


Figure 1.12: Approximation of x(t) by Xn for N=[3,10,30]

Problem 5

Find the Fourier series for the following signal.



Also, sketch the approximation if a large number of terms are kept in the series (say N=30).

Solution

The period of the shown signal is $T = 6$ and therefore $\omega_0 = \frac{2\pi}{T} = \frac{\pi}{3}$.

Taking the derivative of the function we get:

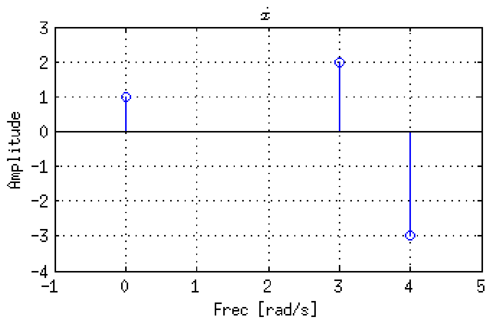


Figure 1.13: Derivative $\dot{x}$

In the range $[-1, 5]$ we have:

$$\dot{x}(t) = \delta(t) + 2\delta(t-3) - 3\delta(t-4)$$

Applying (1.2) we have:

$$\begin{aligned}
jn\omega_0 X_n &= \frac{1}{T}\int_{-T/2}^{T/2} [\delta(t) + 2\delta(t-3) - 3\delta(t-4)] e^{-jn\omega_0 t}\, dt \\
&= \frac{1}{6}[1 + 2e^{-2jn\omega_0} - 3e^{-4nj\omega_0}] \\
&= \frac{1}{6}[1 + 2(-1)^n - 3e^{-\frac{4}{3}jn\pi}] \\
X_n &= \frac{1}{2jn\pi}[1 + 2(-1)^n - 3e^{-\frac{4}{3}jn\pi}]
\end{aligned}$$

where

$$X_0 = \frac{1}{6}\int_{-1}^{3} x(t)\, dt = 1$$

The plot for approximating the function using its Fourier coefficients is:

Listing 1.8: Approximation of x(t) with Fourier coefficients

```
function [x,t] = fapprox5(N,T)
    t = -1.5*T:T/1000:1.5*T;
    w0 = 2*pi/T;
    X0 = 1;
```

```
    n_p = [1:N];
    n_n = [-N:-1];
    Xn = (1 + 2*(-1).^n_p - 3*exp(-4*j*n_p*w0) ) ./ (6*j*n_p*w0);
    X_n = (1 + 2*(-1).^n_n - 3*exp(-4*j*n_n*w0) ) ./ (6*j*n_n*w0);

    Xn = [X_n X0 Xn];
    n = [n_n 0 n_p];

    x = Xn*exp(j*w0*n'*t);
    x = real(x);
end
```

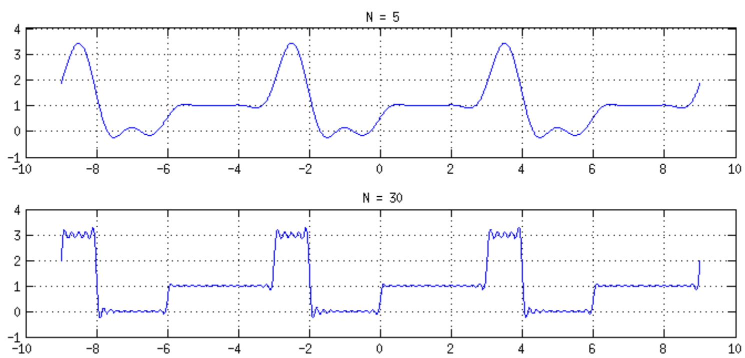


Figure 1.14: Approximation of x(t) by Xn

Chapter 2

Fourier Transform

When we want to know the description of aperiodic signals in terms of the frecuency content we need to use the Fourier Transform. The frecuency components of this non-periodic signals are defined for all reall values of the frecuency variable of ω and not just for discrete values as in the case of periodic ones in which we used the Fourier Series [1].

The Fourier Transform and its inverse of an aperiodic signal are defined as:

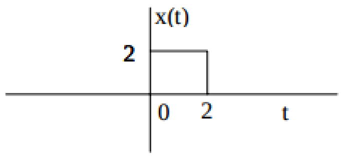
$$\begin{aligned} X(\omega) &= \int_{-\infty}^{\infty} x(t)e^{-j\omega t}\, dt \\ x(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}\, d\omega \end{aligned} \tag{2.1}$$

In the next exercises we will also be using the following properties:

$$x(t) \Leftrightarrow X(\omega) \tag{2.2a}$$
$$x(t-t_0) \Leftrightarrow e^{-j\omega t_0}X(\omega) \tag{2.2b}$$
$$e^{j\omega_0 t}x(t) \Leftrightarrow X(\omega-\omega_0) \tag{2.2c}$$
$$\frac{dx(t)}{dt} \Leftrightarrow j\omega X(\omega) \tag{2.2d}$$
$$x(\alpha t) \Leftrightarrow \frac{1}{|\alpha|}X\left(\frac{\omega}{a}\right) \tag{2.2e}$$

For each signal, find the Fourier transform, $X(\omega)$, and then plot $|X(\omega)|$ (note, you may want to use MATLAB for the plot in 3.)

Problem 1



Solution

Taking the derivative of $x(t)$ we get:

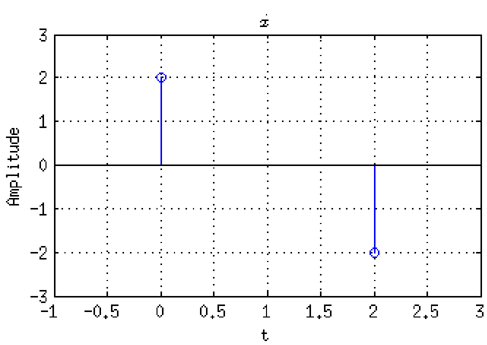


Figure 2.1: Derivative $\dot{x}$

Applying 2.1 and 2.2c to $\dot{x}$ we have:

$$
\begin{aligned}
j\omega X(\omega) &= 2\int_{-\infty}^{\infty} (\delta(t) - \delta(t-2))e^{-j\omega t}\, dt \\
&= 2[1 - e^{-2j\omega}] \\
&= 2e^{-j\omega}[e^{-j\omega} - e^{-j\omega}] \\
&= 4je^{-j\omega}\sin(\omega) \\
X(\omega) &= 4\frac{\sin(\omega)}{\omega}e^{-j\omega} \\
&= 4Sa(\omega)e^{-j\omega}
\end{aligned}
$$

The plot of the magintude and angle of $X(\omega)$ is:

Listing 2.1: Plot of Magnitude and Angle

```
LIM = 3*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 4 * sin(w)./w .* exp(-j * w);

subplot(1,2,1), plot(w,real(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');

subplot(1,2,2), plot(w,angle(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

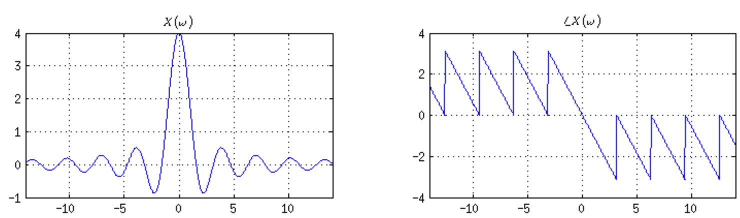
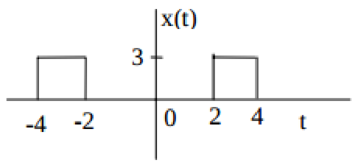


Figure 2.2: Magnitude $|X(\omega)|$ and Angle

Problem 2



Solution

Lets call $x_{p1}(t)$ and $X_{p1}(\omega)$ the function in time domain of the first point and its Fourier Transform respecively. We can express our function in terms of such function as:

$$x(t) = \frac{3}{2}[x_{p1}(t-2) + x_{p1}(t+4)]$$

As we can see from (2.2b) the displacement in time is reflected in the frecuency domain as a multiplication by an exponential.

$$\begin{aligned} X(\omega) &= \frac{3}{2}X_{p1}(\omega)[e^{-2j\omega} + e^{4j\omega}] \\ &= \frac{3}{2}X_{p1}(\omega)e^{j\omega}[e^{-3j\omega} + e^{3j\omega}] \\ &= 12Sa(\omega)\cos(3\omega) \end{aligned}$$

The plot of the magintude and angle of $X(\omega)$ is:

Listing 2.2: Plot of Magnitude and Angle

```
LIM = 1.5*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 12 * sin(w)./w .* cos(3*w);

subplot(1,2,1), plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');

subplot(1,2,2), plot(w,angle(Xw));
```

```
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

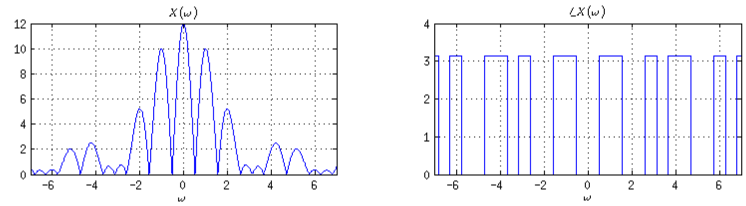
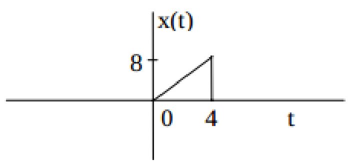


Figure 2.3: Magnitude $|X(\omega)|$ and Angle

Problem 3



Solution

Taking the derivative of $x(t)$ we get:

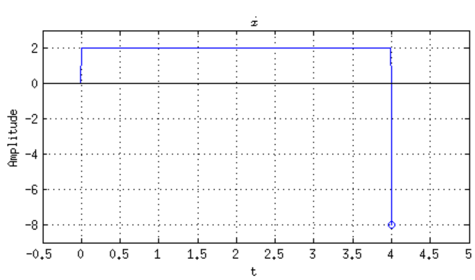


Figure 2.4: Derivative $\dot{x}$

Lets call $x_{p1}(t)$ and $X_{p1}(\omega)$ the function in time domain of the first point and its Fourier Transform respecively. We can express the first derivative of our function in terms of such function as:

$$\dot{x}(t) = x_{p1}(t/2) - 8\delta(t-4)$$

As we can see from (2.2e) the scale in time is reflected inversely in the frecuency domain.

$$\begin{aligned} j\omega X(\omega) &= X_{p1}(2\omega) - 8e^{-4j\omega} \\ &= 4Sa(2\omega)e^{-2j\omega} - 8e^{-4j\omega} \\ X(\omega) &= \frac{4j}{\omega}e^{-2j\omega}[2e^{-2j\omega} - Sa(2\omega)] \end{aligned}$$

The plot of the magintude and angle of $X(\omega)$ is:

Listing 2.3: Plot of Magnitude and Angle

```
LIM = 2*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 4*j./w .* exp(-2*j*w) .* (2*exp(-2*j*w) -  sin(2*w)./(2*w)
     );

subplot(1,2,1), plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');

subplot(1,2,2), plot(w,angle(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

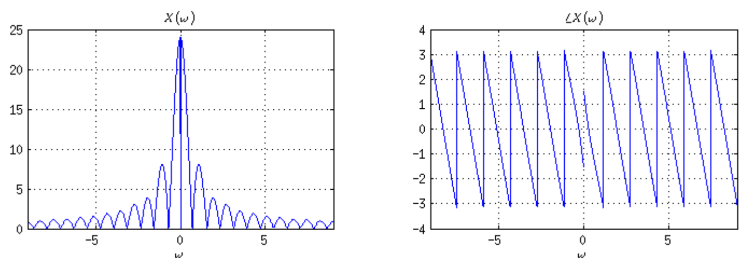


Figure 2.5: Magnitude $|X(\omega)|$ and Angle

Problem 4

$x(t) = cos(200t)G_4(t)$

Solution

The generic Fourier Transform for the gate function $G_\tau(t)$ is:

$$\begin{aligned} \mathfrak{F}\{G_\tau(t)\} &= \tau Sa\left(\frac{\omega\tau}{2}\right) \\ \mathfrak{F}\{G_4(t)\} &= 4Sa(2\omega) \end{aligned} \tag{2.3}$$

We can express our function as:

$$x(t) = \frac{1}{2}[e^{200t} + e^{-200t}]G_4(t)$$

As we can see from (2.2c) the displacement in frecuency is reflected in the time domain as a multiplication by an exponential.

$$X(\omega) = 2[Sa(2(\omega - 200)) + Sa(2(\omega + 200))]$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.4: Plot of Magnitude

```
Sa=@(x) sin(x)./x;

LIM = 100*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 2 * ( Sa(2*(w-200)) + Sa(2*(w+200)) );

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

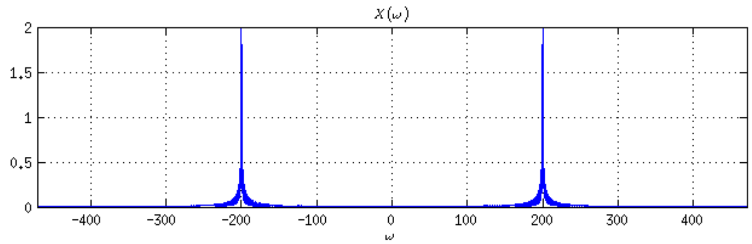


Figure 2.6: Magnitude $|X(\omega)|$

Problem 5

$x(t) = e^{-3t}cos(10t)u(t)$

Solution

From the 1st result of the Fourier Transforms table we have:

$$\mathfrak{F}\{e^{-\alpha t}u(t)\} = \frac{1}{\alpha + j\omega} \tag{2.4}$$

Rearranging the equation for $x(t)$ we have:

$$\begin{aligned} x(t) &= \frac{1}{2}[e^{-3t}e^{10jt} + e^{-3t}e^{-10jt}]u(t) \\ &= \frac{1}{2}[e^{-t(3-10j)} + e^{-t(3+10j)}]u(t) \end{aligned}$$

We can now apply directly the result from (2.5):

$$\begin{aligned} X(\omega) &= \frac{1}{2}\left[\frac{1}{(3-10j)+j\omega} + \frac{1}{(3+10j)+j\omega}\right] \\ &= \frac{1}{2}\left[\frac{1}{3+(\omega-10)j} + \frac{1}{3+(\omega+10)j}\right] \end{aligned}$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.5: Plot of Magnitude

```
LIM = 10*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 0.5 * ( 1./(3+(w-10)*j) + 1./(3+(w+10)*j));

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\|X(\omega)\|$','interpreter','latex');
```

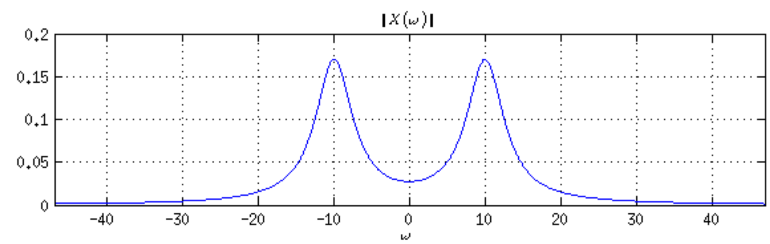
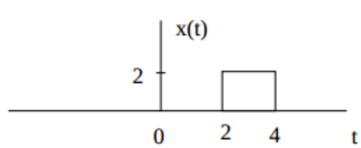


Figure 2.7: Magnitude $|X(\omega)|$

Problem 6

Find the Fourier transform of the following signals. Sketch $|X(\omega)|$ in each case.

1. :



Solution

Lets call $x_{p1}(t)$ and $X_{p1}(\omega)$ the function in time domain of the first point and its Fourier Transform respecively. We can express our function in terms of such function as:

$$x(t) = x_{p1}(t-2)$$

As we can see from (2.2b) the displacement in time is reflected in the frecuency domain as a multiplication by an exponential.

$$\begin{aligned} X(\omega) &= X_{p1}(\omega)e^{-2j\omega} \\ &= 4Sa(\omega)e^{-3j\omega} \end{aligned}$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.6: Plot of Magnitude

```
Sa=@(x) sin(x)./x;

LIM = -0.8*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 4 * Sa(w) .* exp(-3*j*w);

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

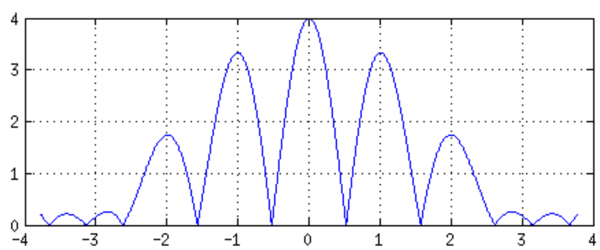


Figure 2.8: Magnitude $|X(\omega)|$

2. $x(t) = 2e^{-2t}u(t)$

Solution

Directly from (2.5) we have:

$$X(\omega) = \frac{2}{2 + j\omega}$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.7: Plot of Magnitude

```
LIM = 2*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 2./(2 + j*w);

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

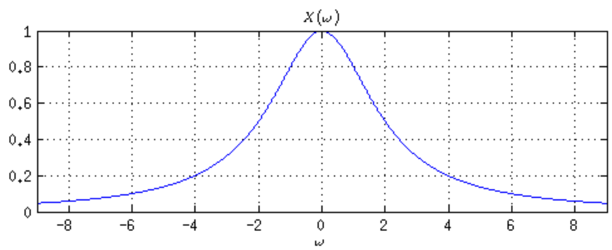


Figure 2.9: Magnitude $|X(\omega)|$

3. $x(t) = 5e^{-5t}u(t)$

Solution

Directly from (2.5) we have:

$$X(\omega) = \frac{5}{5 + j\omega}$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.8: Plot of Magnitude

```
LIM = 3*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 5./(5 + j*w);

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

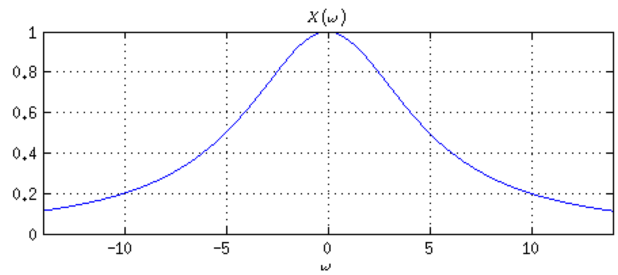


Figure 2.10: Magnitude $|X(\omega)|$

4. $x(t) = e^{-2t}\cos(4t)u(t)$

Solution

$$x(t) = \frac{1}{2}\left[e^{-2t}e^{4jt}u(t) + e^{-2t}e^{-4jt}u(t)\right]$$

Directly from (2.5) we have:

$$\begin{aligned} X(\omega) &= \frac{1}{2}\left[\frac{1}{(2-4j)+j\omega} + \frac{1}{(2+4j)+j\omega}\right] \\ &= \frac{1}{2}\left[\frac{1}{2+(\omega-4)j} + \frac{1}{2+(\omega+4)j}\right] \end{aligned}$$

The plot of the magintude of $X(\omega)$ is:

Listing 2.9: Plot of Magnitude

```
LIM = 2*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
```

```
Xw = 0.5* (1./(2 + (w-4)*j) + 1./(2 + (w+4)*j));

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

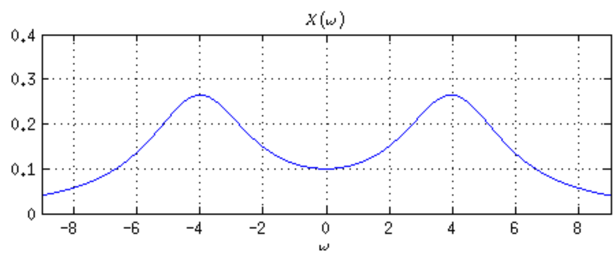
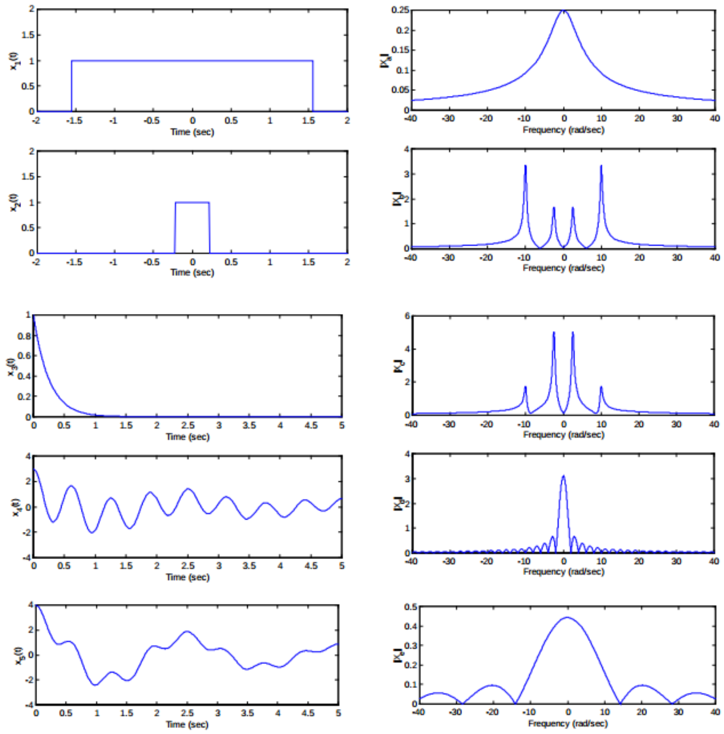


Figure 2.11: Magnitude $|X(\omega)|$

Problem 7

Match the time responses with the corresponding frequency responses.



Solution

As stated in (2.3) the Fourier Transform of a Gate function is a Sa. The 1st gate is more spread in time that the 2nd and therefore its corresponding transform must be more narrow.

$1 \Leftrightarrow d$

$2 \Leftrightarrow e$

The 3rd plot corresponds to a negative exponential truncated by a step function. We know for (2.5) and for Points 6b and 6c that the corresponding transform corresponds to a figure like (a).

$3 \Leftrightarrow a$

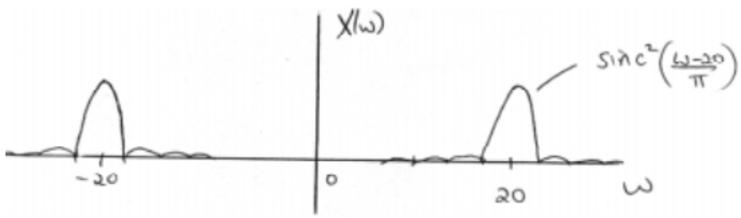
From the following plots we see that the frecuency of 5 (rad/s) is more dominant in the last one and the 10 (rad/s) is more dominant in the previous one.

$4 \Leftrightarrow b$

$5 \Leftrightarrow c$

Problem 8

Compute the inverse Fourier transform of the following signal



Solution

From the 14th result of the Fourier Transforms table we have:

$$Tr_\tau(t) = \begin{cases} 1 - \frac{|t|}{\tau}, & \text{if } |t| < \tau \\ 0, & \text{if } |t| > \tau \end{cases}$$

$$\begin{aligned} \mathfrak{F}\{Tr_\tau(t)\} = \tau \left[Sa\left(\frac{\omega\tau}{2}\right)\right]^2 \\ \mathfrak{F}\{Tr_2(t)\} = 2Sa(\omega)^2 \end{aligned} \tag{2.5}$$

First, we transform sinc into a known function like Sa:

$$sinc^2\left(\frac{\omega}{\pi}\right) = Sa^2(\omega)$$

As we saw in Point 4, when we multiply by a $\cos$ in time we are adding 2 components of the function displaced simetrically with half of its amplitude. The displacement in the plot is of 20, therefore we need to multiply by a $\cos(20t)$. We end up having:

$$x(t) = Tr_2(t)\cos(20t)$$

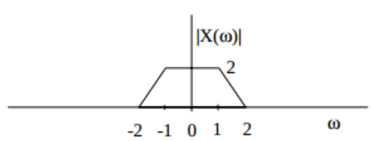
Chapter 3

Sampling and Reconstruction

Problem 1

Draw $|Xs(\omega)|$ for the following cases if $x_s(t) = x(t)p(t)$ with sampling period T.

$$p(t) = \sum_{n=-\infty}^{n=\infty} \delta(t - nT).$$



Solution

The minimum sampling period of the signal according to the Nyquist theorem is:

$$\omega_m = 2$$
$$\omega_{Ns} >= 2\omega_m >= 4$$
$$T_{Ns} = \frac{2\pi}{\omega_s} <= \frac{\pi}{2}$$

- $T_s = \pi/4sec$

This sampling period is lower than the minimum required and therefore no aliasing will occur as can be seen in (3.1).

$$\omega_s = \frac{2\pi}{T_s} = 8$$

Listing 3.1: Plot of $|X_s(\omega)|$

```
t1=[-2:0.01:-1];
x1=[2*t1+4];
t2=[-1:0.01:1];
x2=[0*t2+2];
t3=[1:0.01:2];
x3=[-2*t3+4];

t=[t1 t2 t3];
x=[x1 x2 x3];

ws = 8;
t = [t-ws t t+ws];
x = [x x x];

plot(t,x);
grid on;
ylim([-0.5 2.5]);
xlabel('$\omega$','interpreter','latex');
title('$|X_s(\omega)|$','interpreter','latex');
```

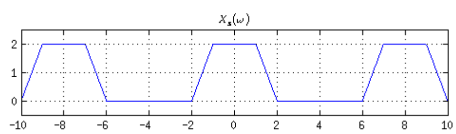


Figure 3.1: Sampling $|X_s(\omega)|$

- $T_s = \pi/2sec$

This sampling period is equal than the minimum required and therefore is in the limit of no aliasing as can be seen in (3.2).

$$\omega_s = \frac{2\pi}{T_s} = 4$$

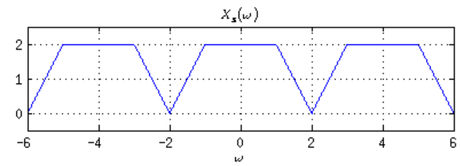


Figure 3.2: Sampling $|X_s(\omega)|$

- $T_s = 2\pi/3sec$

This sampling period is lower than the minimum required and therefore aliasing will occur as can be seen in (3.3).

$$\omega_s = \frac{2\pi}{T_s} = 4$$

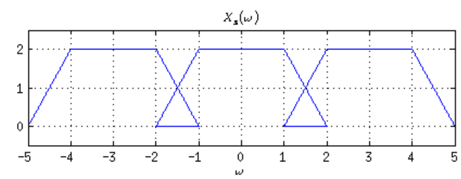


Figure 3.3: Sampling $|X_s(\omega)|$

Problem 2

Repeat Problem 1 where $x(t) = e^{-t/4}cos(t)u(t)$

In order to examine the effects of aliasing in the time domain, plot x(t) for each of the sampling times for t=0 to 15 sec. In MATLAB, this is done by defining your time vector with the time increment set to the desired sampling period. MATLAB then ”reconstructs” the signal by connecting the sampled points with straight lines (this is known as a linear interpolation). Compare your sampled/reconstructed signals with a signal that is more accurate, one that is created by using a very small sampling period (such as T = 0.05 sec) by plotting them on the same graph.

Solution

First we calculate $X(\omega)$.

$$x(t) = \frac{1}{2}\left[e^{-t/4}e^{jt}u(t) + e^{-t/4}e^{-jt}u(t)\right]$$

Directly from (2.5) we have:

$$\begin{aligned} X(\omega) &= \frac{1}{2}\left[\frac{1}{(1/4+j)+j\omega} + \frac{1}{(1/4-j)+j\omega}\right] \\ &= \frac{1}{2}\left[\frac{1}{1/4+(\omega+1)j} + \frac{1}{1/4+(\omega-1)j}\right] \end{aligned}$$

The plot of the magintude of $X(\omega)$ is:

Listing 3.2: Plot of Magnitude

```
LIM = 0.5*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 0.5* (1./(0.25 + (w-1)*j) + 1./(0.25 + (w+1)*j));

plot(w,abs(real(Xw)));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');
```

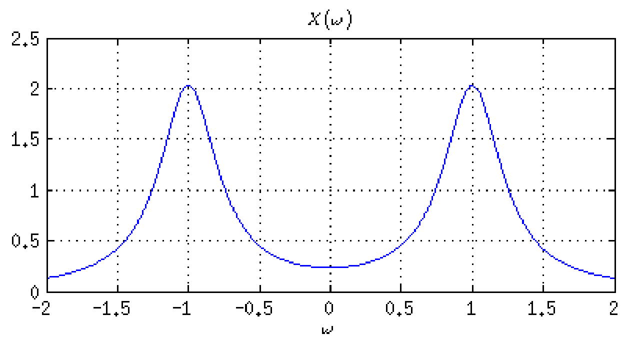


Figure 3.4: Magnitude $|X(\omega)|$

- $T_s = \pi/4sec$

This sampling period is lower than the minimum required and therefore no aliasing will occur as can be seen in (3.5).

$$\omega_s = \frac{2\pi}{T_s} = 8$$

Listing 3.3: Plot of $|X_s(\omega)|$

```
LIM = 2.5*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 0.5* (1./(0.25 + (w-1)*j) + 1./(0.25 + (w+1)*j));
Xw = abs(real(Xw));

ws = 8;
w = [w-ws w w+ws];
Xw = [Xw Xw Xw];

plot(w,Xw);
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
ylim([-0.5 2.5]);
xlabel('$\omega$','interpreter','latex');
title('$|X_s(\omega)|$','interpreter','latex');
```

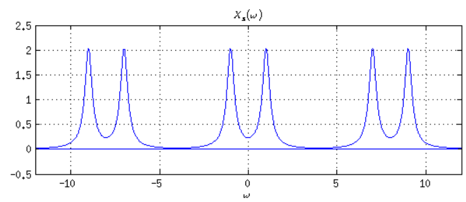


Figure 3.5: Sampling $|X_s(\omega)|$

- $T_s = \pi/2sec$

This sampling period is equal than the minimum required and therefore is in the limit of no aliasing as can be seen in (3.6).

$$\omega_s = \frac{2\pi}{T_s} = 4$$

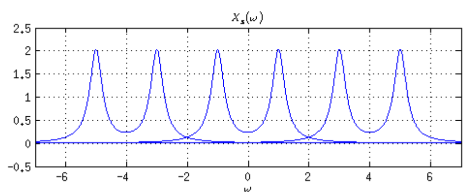


Figure 3.6: Sampling $|X_s(\omega)|$

- $T_s = 2\pi/3 sec$

This sampling period is lower than the minimum required and therefore aliasing will occur as can be seen in (3.7).

$$\omega_s = \frac{2\pi}{T_s} = 4$$

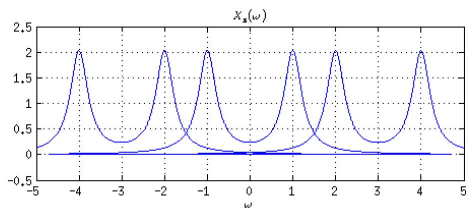


Figure 3.7: Sampling $|X_s(\omega)|$

Plot of ”reconstructed” x(t) with MATLAB:

Listing 3.4: Plot of x(t) for different T

```
u = @(x)(x>=0);

i=1;
for T=[0.05, pi/4, pi/2, 2*pi/3]
    t=[0:T:15];
    x=exp(-t/4).*cos(t).*u(t);

    subplot(4,1,i), plot(t,x);
    grid on;
    xlabel('t');
    title(['Reconstructed x(t) with T=' num2str(T)]);
    i = i + 1;
end
```

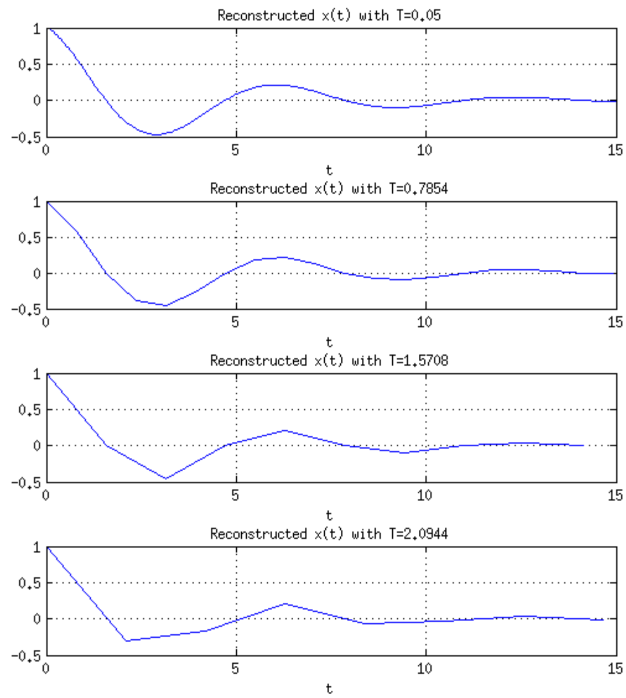
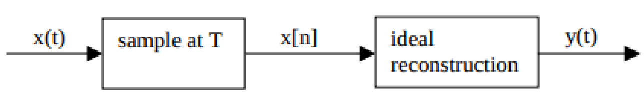


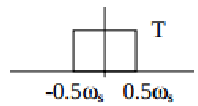
Figure 3.8: Reconstructed x(t)

Problem 3

Consider the following sampling and reconstruction configuration:



The output y(t) of the ideal reconstruction can be found by sending the sampled signal $x_s(t) = x(t)p(t)$ through an ideal lowpass filter:



Let $x(t) = 2 + cos(50\pi t)$ and T = 0.01 sec.

- Draw $|X_s(\omega)|$ where $x_s(t) = x(t)p(t)$. Determine if aliasing occurs.

Solution

First we calculate $X(\omega)$.

$$x(t) = 2(1) + \frac{1}{2}\left[e^{50\pi jt}(1) + e^{-50\pi jt}(1)\right]$$

Using this known result $\mathfrak{F}\{1\} = 2\pi\delta(\omega)$ and from (2.2c) we have:

$$X(\omega) = 4\pi\delta(\omega) + \pi\left[\delta(\omega - 50\pi) + \delta(\omega + 50\pi)\right]$$

The plot of the magintude of $X(\omega)$ is:

Listing 3.5: Plot of Magnitude

```
fe = [-50*pi 0 50*pi];
Ae = [pi 4*pi pi];

figure(1);
stem(fe,Ae);
grid on;
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|X_\omega|', 'fontweight', 'bold');
```

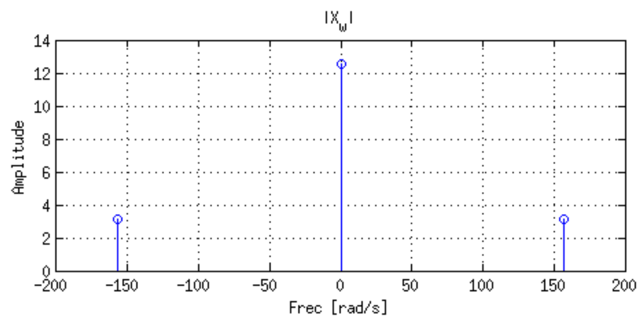


Figure 3.9: Magnitude $|X(\omega)|$

The minimum sampling period of the signal according to the Nyquist theorem is:

$$\begin{aligned} \omega_m &= 50\pi \\ \omega_{Ns} &>= 2\omega_m >= 100\pi \\ T_{Ns} &= \frac{2\pi}{\omega_s} <= \frac{1}{50} <= 0.02 \end{aligned}$$

Since $T_s = 0.01sec$ is lower than the minimum required no aliasing will occur as can be seen in (3.10).

$$\omega_s = \frac{2\pi}{T_s} = 200\pi$$

Listing 3.6: Plot of $|X_s(\omega)|$

```
w = [-50*pi 0 50*pi];
Xw = [pi 4*pi pi];

ws = 200*pi;
w = [w-ws w w+ws];
Xw = [Xw Xw Xw];

stem(w,Xw);
grid on;
%xlim([-50*pi-2 50*pi+2]);
xlabel('$\omega$','interpreter','latex');
title('$|X_s(\omega)|$','interpreter','latex');
```

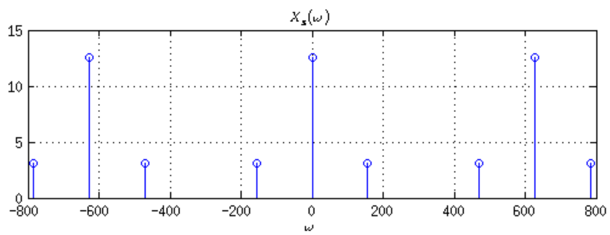


Figure 3.10: Sampling $|X_s(\omega)|$

- Determine the expression for y(t).

Solution

The limits of the lowpass filter are $-0.5\omega_s = -100\pi$ to $0.5\omega_s = 100\pi$. The Fourier transform of $Y(\omega) = H(\omega)X(\omega)$ is:

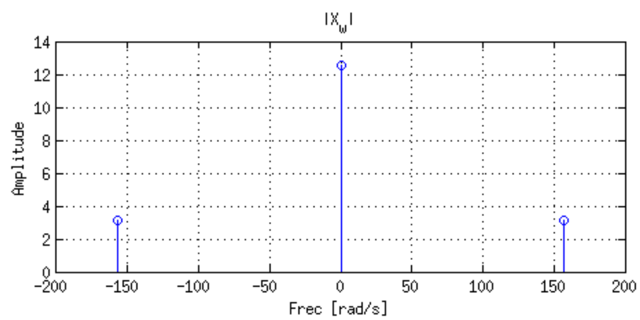


Figure 3.11: Magnitude $|X(\omega)|$

And therefore:

$$y(t) = 2 + cos(50\pi t)$$

- Determine an expression for x[n].

Solution

$$x(n) = 2 + cos(50\pi n T_s)$$
$$= 2 + cos(0.5\pi n)$$

Problem 4

Repeat Problem 3 for $x(t) = 2 + \cos(50\pi t)$ and T = 0.025 sec.

- Draw $|X_s(\omega)|$ where $x_s(t) = x(t)p(t)$. Determine if aliasing occurs.

Solution

Since $T_s = 0.025sec$ is greater than the minimum required, aliasing will occur as can be seen in (3.12).

$$\omega_s = \frac{2\pi}{T_s} = 80\pi$$

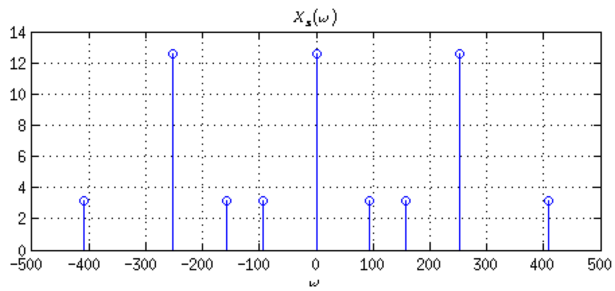


Figure 3.12: Sampling $|X_s(\omega)|$

- Determine the expression for y(t).

Solution

The limits of the lowpass filter are $-0.5\omega_s = -40\pi$ to $0.5\omega_s = 40\pi$. The Fourier transform of $Y(\omega) = H(\omega)X(\omega)$ is:

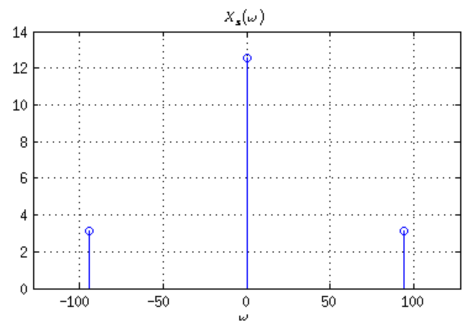


Figure 3.13: Magnitude $|X(\omega)|$

And therefore:

$$y(t) = 2 + \cos(30\pi t)$$

- Determine an expression for x[n].

Solution

$$\begin{aligned} x(n) &= 2 + \cos(50\pi n T_s) \\ &= 2 + \cos(1.25\pi n) \end{aligned}$$

Problem 5

Repeat Problem 3 for $x(t) = 1+\cos(20\pi t)+\cos(60\pi t)$ and T = 0.01 sec.

- Draw $|X_s(\omega)|$ where $x_s(t) = x(t)p(t)$. Determine if aliasing occurs.

Solution

First we calculate $X(\omega)$.

$$X(\omega) = \pi[2\delta(\omega) + \delta(\omega - 20\pi) + \delta(\omega + 20\pi) + \delta(\omega - 60\pi) + \delta(\omega + 60\pi)]$$

The plot of the magintude of $X(\omega)$ is:

Listing 3.7: Plot of Magnitude

```
w = [-60*pi -20*pi 0 20*pi 60*pi];
Xw = [pi pi 2*pi pi pi];

figure(1);
stem(w,Xw);
grid on;
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|X_\omega|', 'fontweight', 'bold');
```

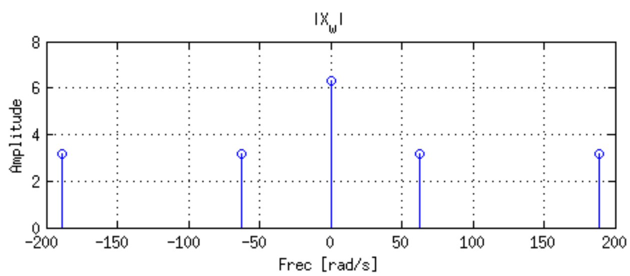


Figure 3.14: Magnitude $|X(\omega)|$

The minimum sampling period of the signal according to the Nyquist theorem is:

$$\begin{aligned} \omega_m &= 60\pi \\ \omega_{Ns} &>= 2\omega_m >= 120\pi \\ T_{Ns} &= \frac{2\pi}{\omega_s} <= \frac{1}{50} <= 0.01666 \end{aligned}$$

Since $T_s = 0.01sec$ is lower than the minimum required no aliasing will occur as can be seen in (3.15).

$$\omega_s = \frac{2\pi}{T_s} = 200\pi$$

Listing 3.8: Plot of $|X_s(\omega)|$

```
w = [-60*pi -20*pi 0 20*pi 60*pi];
Xw = [pi pi 2*pi pi pi];

ws = 200*pi;
w = [w-ws w w+ws];
Xw = [Xw Xw Xw];

figure(1);
stem(w,Xw);
grid on;
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|X_\omega|', 'fontweight', 'bold');
```

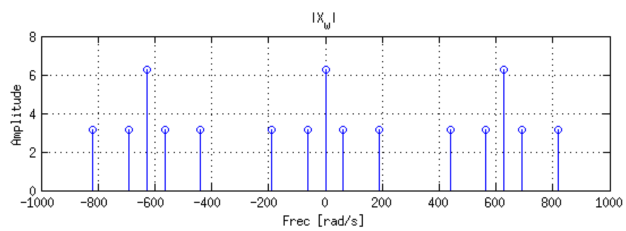


Figure 3.15: Sampling $|X_s(\omega)|$

- Determine the expression for y(t).

Solution

The limits of the lowpass filter are $-0.5\omega_s = -100\pi$ to $0.5\omega_s = 100\pi$. The Fourier transform of $Y(\omega) = H(\omega)X(\omega)$ is:

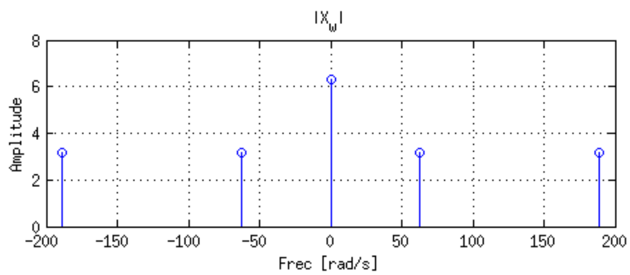


Figure 3.16: Magnitude $|X(\omega)|$

And therefore:

$$y(t) = x(t) = 1 + \cos(20\pi t) + \cos(60\pi t)$$

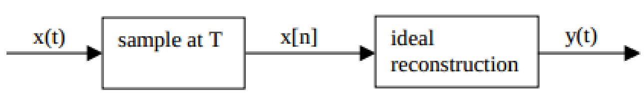
- Determine an expression for x[n].

Solution

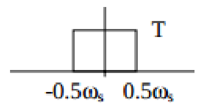
$$\begin{aligned} x(n) &= 1 + \cos(20\pi n T_s) + \cos(60\pi n T_s) \\ &= 2 + \cos(0.2\pi n) + \cos(0.6n) \end{aligned}$$

Problem 6

Consider the following sampling and reconstruction configuration:



The output y(t) of the ideal reconstruction can be found by sending the sampled signal $x_s(t) = x(t)p(t)$ through an ideal lowpass filter:



- Let $x(t) = 1 + cos(15\pi t)$ and T = 0.1 sec. Draw $|X_s(\omega)|$ where $x_s(t) = x(t)p(t)$. Determine the expression for y(t).

Solution

First we calculate $X(\omega)$.

$$X(\omega) = \pi[2\delta(\omega) + \delta(\omega - 15\pi) + \delta(\omega + 15\pi)]$$

The plot of the magintude of $X(\omega)$ is:

Listing 3.9: Plot of Magnitude

```
w = [-15*pi 0 15*pi];
Xw = [pi 2*pi pi];

figure(1);
stem(w,Xw);
grid on;
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|X_\omega|', 'fontweight', 'bold');
```

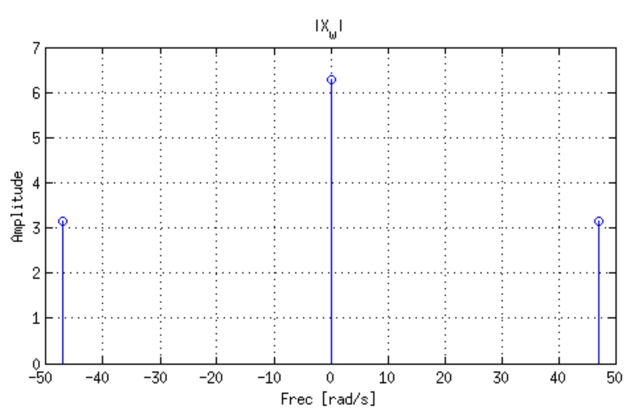


Figure 3.17: Magnitude $|X(\omega)|$

The minimum sampling period of the signal according to the Nyquist theorem is:

$$
\begin{aligned}
\omega_m &= 15\pi \\
\omega_{Ns} &>= 2\omega_m >= 30\pi \\
T_{Ns} &= \frac{2\pi}{\omega_s} <= \frac{1}{15} <= 0.066
\end{aligned}
$$

Since $T_s = 0.1 sec$ is greater than the minimum required, aliasing will occur as can be seen in (3.18).

$$\omega_s = \frac{2\pi}{T_s} = 20\pi$$

Listing 3.10: Plot of $|X_s(\omega)|$

```
w = [-15*pi 0 15*pi];
Xw = [pi 2*pi pi];

ws = 20*pi;
w = [w-ws w w+ws];
Xw = [Xw Xw Xw];

stem(w,Xw);
grid on;
%xlim([-10*pi-2 10*pi+2]);  %passband filter
xlabel('Frec [rad/s]');
ylabel('Amplitude');
title('|x_\omega|', 'fontweight', 'bold');
```

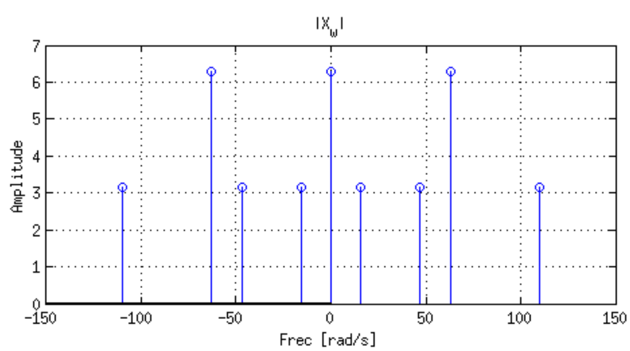


Figure 3.18: Sampling $|X_s(\omega)|$

The limits of the lowpass filter are $-0.5\omega_s = -10\pi$ to $0.5\omega_s = 10\pi$. The

Fourier transform of $Y(\omega) = H(\omega)X(\omega)$ is:

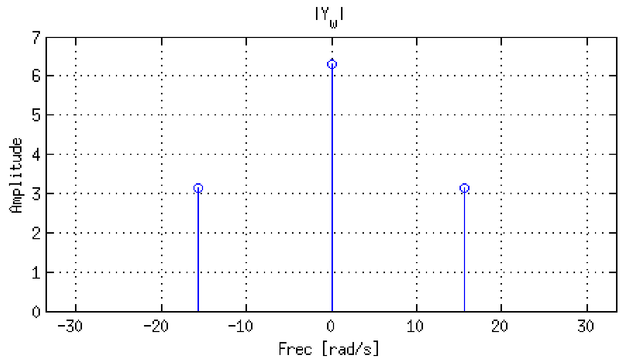


Figure 3.19: Magnitude $|X(\omega)|$

And therefore only the $\omega = 5\pi$ passes through it:

$$y(t) = 1 + cos(5\pi t)$$

- Let $X(\omega) = \frac{1}{(j\omega+1)}$ and T = 1 sec. Draw $|X_s(\omega)|$ where $x_s(t) = x(t)p(t)$. Does aliasing occur? (Justify your answer).

Solution

The magnitude of $X(\omega)$ is:

$$|X(\omega)| = \frac{1}{\omega^2 + 1}$$

From the previous equation we can see that the function never intesects the w axis, and therefore there will always be aliasing. However, setting w_m to its FWHM we have:

$$\frac{1}{\omega_m^2 + 1} = \frac{1}{2}$$
$$\omega_m = \sqrt{3}$$

Hipoteticaly taking this value as w_m we see that $w_s = 2\pi > 2w_m = 2\sqrt{3}$ and in this case there will not be aliasing.

The plot of the magintude of $X(\omega)$ and $X_s(\omega)$ is:

Listing 3.11: Plot of Magnitude

```
LIM = 1.5*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 1./(j*w + 1);

subplot(2,1,1), plot(w,abs(real(Xw)));
grid on;
xlabel('$\omega$','interpreter','latex');
title('$|X(\omega)|$','interpreter','latex');

%FWHM
hold on
stem(sqrt(3),1./(j*sqrt(3) + 1))

ws = 2*pi;
w = [w-ws w w+ws];
Xw = [Xw Xw Xw];

subplot(2,1,2), plot(w,abs(real(Xw)));
grid on;
%xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$|X_s(\omega)|$','interpreter','latex');

%FWHM
hold on
stem(sqrt(3),1./(j*sqrt(3) + 1))
```

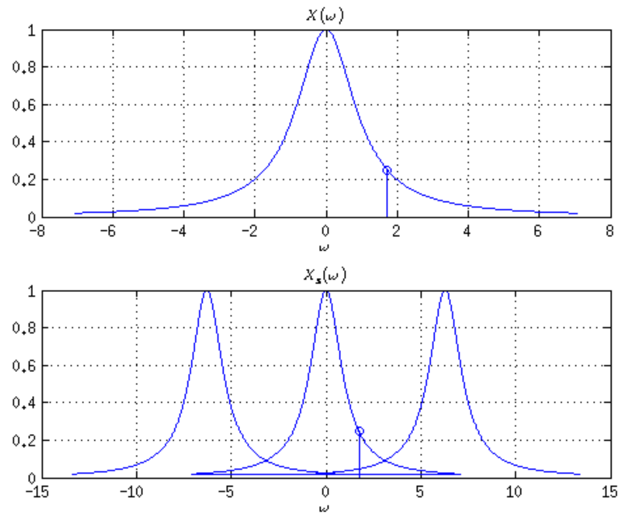


Figure 3.20: Magnitude

Chapter 4

DTFT and DFT

When we move into a discrete time domain, the Fourier Transform equivalent in this domain is called the DTFT or Discrete Time Fourier Transform which is defined as [1]:

$$\mathfrak{F}\{x[n]\} = \sum_{-\infty}^{\infty} x[n]e^{-j\omega n} = X(\omega) \tag{4.1}$$

Problem 1

Compute the DTFT of the following signals and sketch $X(\omega)$.

- $x[n] = \left[\begin{array}{cccc} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{array}\right]$

Solution

Aplying the definition of (4.1) we have:

$$
\begin{aligned}
X(\omega) &= \sum_{0}^{3} \frac{1}{4} e^{-j\omega n} \\
&= \frac{1}{4}[1 + e^{-jw} + e^{-2jw} + e^{-3jw}] \\
&= \frac{1}{4} e^{-j\frac{3\omega}{2}} [e^{j\frac{3\omega}{2}} + e^{-j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} + e^{-j\frac{3\omega}{2}}] \\
&= \frac{1}{2} e^{-j\frac{3\omega}{2}} [\cos(\frac{3\omega}{2}) \cos(\frac{\omega}{2})]
\end{aligned}
$$

The plot of the magintude $X(\omega)$ is:

Listing 4.1: Plot of Magnitude

```
LIM = 2.3;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 0.5 * exp(-j * 1.5 * w) .* ( cos(w*1.5) + cos(w*0.5) );

subplot(2,1,1), plot(w,abs(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\|X(\omega)\|$','interpreter','latex');

subplot(2,1,2), plot(w,angle(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

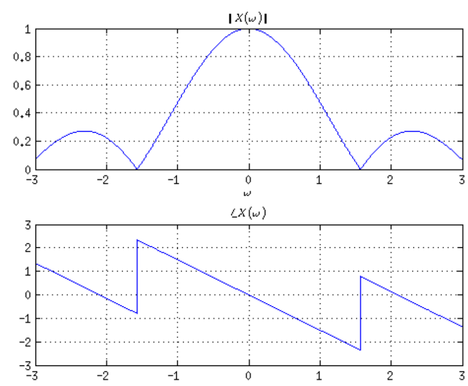


Figure 4.1: Magnitude $|X(\omega)|$

- $x[n] = \left[\begin{array}{ccc} 1 & -2 & 1 \end{array}\right]$

Solution

Aplying the definition of (4.1) we have:

$$
\begin{aligned}
X(\omega) &= \sum_{0}^{2} x[n]e^{-j\omega n} \\
&= [1 - 2e^{-jw} + e^{-2jw}] \\
&= e^{-jw}[e^{jw} - 2 + e^{-jw}] \\
&= e^{-jw})2[\cos(\omega) - 1] \\
&= 2e^{-jw}[\cos(\omega) - 1]
\end{aligned}
$$

The plot of the magintude $X(\omega)$ is:

Listing 4.2: Plot of Magnitude

```
LIM = 0.7*pi;
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 2 * exp(-j*w) .* (cos(w)-1);

subplot(2,1,1), plot(w,abs(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\|X(\omega)\|$','interpreter','latex');

subplot(2,1,2), plot(w,angle(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

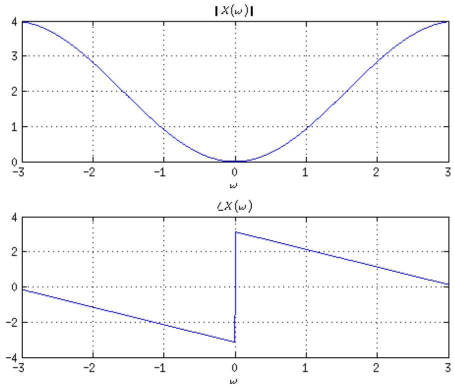


Figure 4.2: Magnitude $|X(\omega)|$

- $x[n] = 2\left(\frac{3}{4}\right)^n u[n]$

Solution

$$X(\omega) = \sum_{0}^{\infty} 2\left(\frac{3}{4}\right)^n e^{-j\omega n}$$

Recall that the geometric series solution is given by:

$$\sum ar^n = \frac{a}{1-r} \text{ if } |r| < 1 \tag{4.2}$$

Then, aplying (4.1) and (4.2) we have:

$$\begin{aligned} X(\omega) &= \sum_{0}^{\infty} 2\left(\frac{3}{4}e^{-j\omega}\right)^n \\ &= \frac{2}{1 - \frac{3}{4}e^{-j\omega}} \end{aligned}$$

The plot of the magintude $X(\omega)$ is:

Listing 4.3: Plot of Magnitude

```
LIM = 0.7*pi;
```

```
w = -1.5*LIM:LIM/1000:1.5*LIM;
Xw = 2 ./(1 - 3/4*exp(-j*w) );

subplot(2,1,1), plot(w,abs(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\|X(\omega)\|$','interpreter','latex');

subplot(2,1,2), plot(w,angle(Xw));
grid on;
xlim([-round(1.5*LIM) round(1.5*LIM)]);
xlabel('$\omega$','interpreter','latex');
title('$\angle X(\omega)$','interpreter','latex');
```

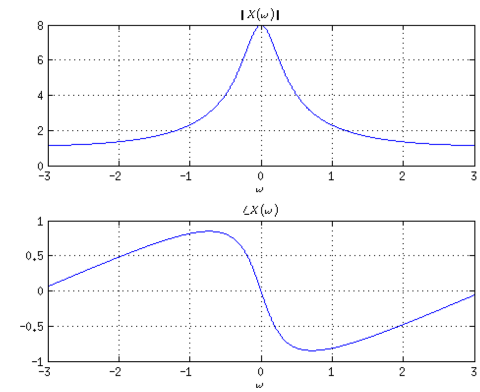


Figure 4.3: Magnitude $|X(\omega)|$

Chapter 5

Z-Transform

Chapter 6

Haar Base

Problem 1

Create a GUI that plots the approximation of the function $x(t) = t^2$ using Haar Base. The values for

$$\phi(2^j - t) = \begin{cases} 1, & \text{if } \frac{k}{2^j} <= t <= \frac{k+1}{2^j} \\ 0, & \text{in any other case} \end{cases}$$

With

$$C_k = 2^j \int_{\frac{k}{2^j}}^{\frac{k+1}{2^j}} f(t)\phi(t-k)\, dt$$

Solution

For j=0 we have:

$$\begin{aligned}
C_k &= \int_k^{k+1} t^2\phi(t-k)\;dt \\
&= \frac{1}{3}t^3\bigg|_k^{k+1} \\
&= \frac{1}{3}[3k^2+3k+1] \text{ with } \; k=-3,-2,-1,0,1,2 \\
f(t) &= \sum_{k=-3}^{2}\frac{1}{3}[3k^2+3k+1]\phi(t-k)
\end{aligned}$$

For j¿=0 we use the following code to plot the GUI in matlab:

Listing 6.1: Approximation Function

```
function [ tk, fs ] = approximate_t2( j, t )
%APPROXIMATE Aproximacin de la funcin t^2 mediante steps

j2=2^j;
k=[j2*(-t) : j2*(t)-1]
fs = 1/(j2^2) * (k.^2 + k + 1/3)
tk = k/j2
%plot(tk,tk.^2,'r')
%hold on
%stairs(tk,fs,'b')

end
```

Listing 6.2: Plot of the Approximation Function

```
function [ output_args ] = plotAprox(tk,f,fs)
%PLOTAPROX Summary of this function goes here
%   Detailed explanation goes here

x = linspace(tk(1),-tk(1),int32(20));
plot(x,f(x),'r')
hold on
stairs(tk,fs,'b')

end
```

Listing 6.3: GUI

```
function varargout = haar(varargin)
% HAAR MATLAB code for haar.fig
```

```
%      HAAR, by itself, creates a new HAAR or raises the existing
%      singleton*.
%
%      H = HAAR returns the handle to a new HAAR or the handle to
%      the existing singleton*.
%
%      HAAR('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in HAAR.M with the given input arguments.
%
%      HAAR('Property','Value',...) creates a new HAAR or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before haar_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to haar_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help haar

% Last Modified by GUIDE v2.5 04-Apr-2013 15:42:43

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @haar_OpeningFcn, ...
                   'gui_OutputFcn',  @haar_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


function calcAndPlot(handles)
cla
```

```
tmax = str2num(get(handles.tmax,'String'));
j = str2num(get(handles.j,'String'));
[tk,fs]=approximate_t2(j,tmax);
plotAprox(tk,handles.ft2,fs);

% --- Executes just before haar is made visible.
function haar_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to haar (see VARARGIN)

handles.ft2 = @(t) t.^2;

calcAndPlot(handles)


% Set the current data value.
%handles.current_data = handles.peaks;
%plot(handles.current_data)

% Choose default command line output for haar
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes haar wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = haar_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pbCalcular.
function pbCalcular_Callback(hObject, eventdata, handles)
% hObject    handle to pbCalcular (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
calcAndPlot(handles)

% --- Executes on selection change in popupmenu2.
```

```matlab
function popupmenu2_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu2 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from popupmenu2


% --- Executes during object creation, after setting all properties.
function popupmenu2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider


% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

```

```
function tmax_Callback(hObject, eventdata, handles)
% hObject    handle to tmax (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of tmax as text
%        str2double(get(hObject,'String')) returns contents of tmax as a double


% --- Executes during object creation, after setting all properties.
function tmax_CreateFcn(hObject, eventdata, handles)
% hObject    handle to tmax (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function j_Callback(hObject, eventdata, handles)
% hObject    handle to j (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of j as text
%        str2double(get(hObject,'String')) returns contents of j as a double


% --- Executes during object creation, after setting all properties.
function j_CreateFcn(hObject, eventdata, handles)
% hObject    handle to j (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
```

```
    set(hObject,'BackgroundColor','white');
end
```

Chapter 7

Haar Transform

Bibliography

[1] Edward Kamen and Bonnie Heck. *Fundamentals of Signals and Systems: With MATLAB Examples*. Prentice Hall PTR, 2000.

[2] Georgia Tech School of Electrical and Computer Engineering. Worked Problems, Chapter 4. http://users.ece.gatech.edu/~bonnie/book/worked_problems/Chap4_Fseries_sol.pdf. [Online; accessed Apr-2013].