ECE 300
Signals and Systems
Homework 5

Due Date: Thursday April 6 at the beginning of class

Problems:

1. Read the **Appendix-A** and then do the following:

a) Show that the functions $v_k(t) = e^{jk\omega_o t}$ are orthogonal, that is, show that

$$< v_n(t), v_m(t) > = \begin{cases} 0 & n \neq m \\ T & n = m \end{cases}$$

In addition, show that for these functions $c_k = \frac{1}{T}\int_T x(t)e^{-jk\omega_o t}dt$

Write an m-file that will do all of the following (parts b-h), and turn it in with your homework.

For parts **b-f** use the following functions:

$$v_1(t) = 0.7071$$
$$v_2(t) = 1.0754e^t - 1.2638$$
$$v_3(t) = 4.9632t + 4.9632 - 4.2232e^t$$

b) Using Matlab, show that these functions are (approximately) orthogonal over the interval [-1,1].

c) Using Matlab, show that these functions have unit length.

d) Assume we want to approximate $x(t) = t^4 - t$ in this interval using only $v_1(t)$, so that $x(t) \approx c_1 v_1(t)$. Determine c_1.

e) Assume we want to approximate $x(t)$ using the first two functions, so that $x(t) \approx c_1 v_1(t) + c_2 v_2(t)$. Determine c_1 and c_2.

f) Assume we want to approximate $x(t)$ using all three functions. Determine c_1, c_2, and c_3.

g) Plot the original function $x(t) = t^4 - t$ and the three approximations (from parts **d**, **e**, and **f**) on the same graph. Be sure to use different line types and a legend.

h) Using the *principle of orthogonality*, with the vectors

$$w_1(t) = 1$$
$$w_2(t) = t$$
$$w_3(t) = e^t$$

determine an approximation to $x(t)$ using all three functions. Plot the approximation and the real function on the same graph. Compare your results to those in part **g. It may be useful to plot the error between the approximation made using the principle of orthogonality and the estimate from part f.**

Hint: To solve the system of equations Ac =b in Matlab, just type c = A\b. To enter the matrix

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

into Matlab, type

A = [a b c; d e f; g h i];

Special Note: We will be using the code you write in the next part for the next few homeworks and labs, so be sure you do this and understand what is going on!

2. Read the **Appendix-B** and then do the following:

a) Copy the file **Trigonometric_Fourier_Series.m** (you wrote this for homework 4**)** to file **Complex_Fourier_Series.m**.

b) Modify **Complex_Fourier_Series.m** so it computes the average value c_o

c) Modify **Complex_Fourier_Series.m** so it also computes c_k for $k = 1$ to $k = N$

d) Modify **Complex_Fourier_Series.m** so it also computes the Fourier series estimate using the formula

$$x(t) \approx c_o + \sum_{k=1}^{N} 2\,|\,c_k\,|\cos(k\omega_o t + \measuredangle c_k)$$

You will probably need to use the Matlab functions **abs** and **angle** for this.

e) Using the code you wrote in part **d,** find the complex Fourier series representation for the following functions (defined over a single period)

$$f_1(t) = e^{-t}u(t) \quad 0 \le t < 3$$

$$f_2(t) = \begin{cases} t & 0 \le t < 2 \\ 3 & 2 \le t < 3 \\ 0 & 3 \le t < 4 \end{cases}$$

$$f_3(t) = \begin{cases} 0 & -2 \le t < -1 \\ 1 & -1 \le t < 2 \\ 3 & 2 \le t < 3 \\ 0 & 3 \le t < 4 \end{cases}$$

These are the same functions you used for the trigonometric Fourier series. Use N = 10 and turn in your plots for each of these functions. Also, turn in your Matlab program for one of these. *Note that the values of **low** and **high** will be different for each of these functions!*

Appendix-A

The Fourier series is just one possible way of representing one function in terms of other functions. In order to determine a Fourier series representation for a function the function must be periodic. However, it is not necessary for a function to be periodic in order to represent it in terms of other functions. In fact, the Fourier series is really just a very special case of a general idea. In order to understand the general procedure a bit, we need some definitions first.

Inner Products An inner product can be thought of as a projection of one vector onto another vector. The "dot product" of two ordinary (Euclidean) vectors is an example of an inner product. We need to expand this type of definition to include functions. We will denote the inner product between the two functions $x(t)$ and $v(t)$ as $< x(t), v(t) >$.There are mathematical rules for what constitutes a valid inner product, but we will use the following

$$< x(t), v(t) > = \int_T x(t)v^*(t)dt$$

Here the * means the complex conjugate. Note that the inner product of a function with itself gives us the length squared of the function, just as the dot product of a vector with itself gives us the length squared of the vector.

Orthogonal Functions Two functions $x(t)$ and $v(t)$ are orthogonal if their inner product is zero, i.e., if $< x(t), v(t) > = 0$. This is similar to saying to vectors are orthogonal if their dot product is zero.

Orthonormal Functions Two functions $x(t)$ and $v(t)$ are orthonormal if they are both orthogonal and have unit length, i.e. if

$$
\begin{aligned}
&< x(t), v(t) >= 0 \\
&< x(t), x(t) >= 1 \\
&< v(t), v(t) >= 1
\end{aligned}
$$

Orthogonal Function Representation If functions $v_1(t), v_2(t), ..., v_n(t)$ are orthogonal, then the estimate of a function $x(t)$ that minimizes the squared error is

$$x(t) \approx c_1 v_1(t) + c_2 v_2(t) + ... + c_n v_n(t)$$

where

$$c_k = \frac{< x(t), v_k(t) >}{< v_k(t), v_k(t) >}$$

If the $v_i(t)$ are orthonormal, then $c_k =< x(t), v_k(t) >$ since the denominator is clearly equal to 1. Here the squared error is defined as $< e(t), e(t) >$ where the error signal $e(t)$ is defined as

$$e(t) = x(t) - [c_1 v_1(t) + c_2 v_2(t) + ... + c_n v_n(t)]$$

Principle of Orthogonality This principle states that to produce the estimate that minimizes the squared error, the error signal should be orthogonal to each of the functions used in creating the estimate. Mathematically, this means that

$$< e(t), v_i(t) > = 0 \text{ for } i = 1...N$$

Note that this does not require the $v_i(t)$ to be orthogonal or orthonormal. However, there is a set of simultaneous equations that must be solved.

Note: If the function is a constant, say 0.7071, and you want to square it, you may need to use the following function definition:

*v1 = @(t) 0.7071*ones(1,length(t))*

Appendix-B

In the majority of this course we will be using the complex (or exponential) form of the Fourier series, since it is really easier to do various mathematical things with it once you get used to it.

<u>Exponential Fourier Series</u> If $x(t)$ is a periodic function with fundamental period T, then we can represent $x(t)$ as a Fourier series

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\omega_o t}$$

where $\omega_o = \frac{2\pi}{T}$ is the fundamental period, $\mathrm{c_o}$ is the average (or DC, i.e. zero frequency) value, and

$$\mathrm{c_o} = \frac{1}{T}\int_0^T x(t)dt$$

$$c_k = \frac{1}{T}\int_0^T x(t)e^{-jk\omega_o t}dt$$

If $x(t)$ is a real function, then we have the relationships $|c_k| = |c_{-k}|$ (the magnitude is even) and $\measuredangle c_{-k} = -\measuredangle c_k$ (the phase is odd). Using these relationships we can then write

$$x(t) = c_o + \sum_{k=1}^{\infty} 2|c_k|\cos(k\omega_o t + \measuredangle c_k)$$

This is usually a much easier form to deal with, since it lends itself easily to thinking of a phasor representation of $x(t)$. This will be particularly useful when we starting filtering periodic signals.