EE 301 Lab 6 – Fourier Series

In this lab we will gain experience with the Fourier Series.

1 What you will learn

This lab will be focused on approximating signals as sum of sinusoids and becoming more familiar with the Fourier Series.

2 Background Information and Notes

2.1 Introduction

As mentioned previously, sinusoids are important for signal analysis. Many real world signals are sinusoidal. For example, musical signals and speech sounds can be approximately described as sums of sinusoids. We have studied in class that any periodic signal can be written as a sum of amplitude-scaled and phase-shifted sinusoids. Equivalently, using Euler's inverse formulas we could write the periodic signal as sums of complex exponentials.

The need for representing a signal in the frequency domain may be because it may be simpler than a time-domain representation. Also, understanding the signal effects through a system may be simpler using a signal's spectrum. Suppose that we have a signal that is the sum of two different signals. If we would like to separate one signal from the other, but the signals overlap in time, it may be possible to separate the two signals if their frequency-domain representations do not overlap.

In this laboratory, we will examine a tool, *Fourier Series*, that allows us to work with spectral representations of periodic continuous-time signals.

2.2 Frequency-domain representation

In general we have thought of signals as time-varying quantities, like s(t). To plot s(t), we plot time along the horizontal axis and signal values along the vertical axis. For the frequency domain analysis, we plot a spectral value versus frequency. This process involves *transforming* the signal.

The frequency domain representation of a signal (i.e., its *spectrum*) is simple to construct when the signal is composed of a sum of simple complex exponential signals. In such a case, the spectrum consists of a few isolated spectral lines ("spikes") on the frequency axis (at the frequencies of those complex exponentials). Alternatively, we can plot the magnitude and phase separately.

The more complex exponentials added to our signal, the more we add spectral lines to its frequency-domain representation. If more and more complex exponentials are added we can basically represent any signal., even signals do not look very sinusoidal, like square waves and sawtooth waves.

2.3 Periodic Continuous-Time Signals – The Fourier Series

Suppose that we have a periodic continuous-time signal s(t) with period T seconds. The complex exponentials have harmonically related frequencies, and form a harmonic series

$$\ldots,\ -3\omega_0,\ -2\omega_0,\ -\omega_0,\ 0,\ \omega_0,\ 2\omega_0,\ 3\omega_0,\ \ldots \tag{2.1}$$

where,

$$\omega_0 = \frac{2\pi}{T} \tag{2.2}$$

is the fundamental frequency. The frequency $k\omega_0$, for $k \geq 2$, is called the k-th harmonic of the fundamental frequency, or the k-th harmonic frequency.

The representation of s(t) in terms of complex exponentials with these frequencies is given by the Fourier Series synthesis formula:

$$\begin{aligned} s(t) &= \ldots C_{-2}e^{j\frac{2\pi(-2)}{T}t} + C_{-1}e^{j\frac{2\pi(-1)}{T}t} + C_0 e^{j\frac{2\pi(0)}{T}t} + C_1 e^{j\frac{2\pi(1)}{T}t} + C_2 e^{j\frac{2\pi(2)}{T}t} + \ \ldots \\ &= \sum_{k=-\infty}^{\infty} C_k e^{j\frac{2\pi k}{T}t} \end{aligned} \tag{2.3}$$

where the C_k's, which are called Fourier coefficients. The Fourier coefficients are determined by the Fourier series analysis formula

$$C_k = \frac{1}{T}\int_{\langle T\rangle} s(t)e^{-j\frac{2\pi k}{T}t}dt \tag{2.4}$$

where $\int_{\langle T\rangle}$ indicates an integral over a T second interval. The Fourier synthesis formula shows that the complex exponential component of s(t) at frequency $\frac{2\pi k}{T}t$ is

$$C_k e^{j\frac{2\pi k}{T}t} \tag{2.5}$$

In general, the Fourier coefficients (C_k's), are complex, and thus they have a magnitude $|C_k|$ and a phase $\angle C_k$. The magnitude $|C_k|$ is the strength of the exponential component at frequency $k\omega_0 = 2\pi k / T$ while the angle $\angle C_k$ provides the phase of that component. The coefficient C_0 is the DC term and it measures the average value of the signal over one period.

Once the C_k values are determined, the spectrum of s(t) is simply a plot consisting of spectral lines at frequencies

$$\ldots,\ -2\omega_0,\ -\omega_0,\ 0,\ \omega_0,\ 2\omega_0,\ \ldots$$

The Fourier synthesis formula is very similar to the formula given in Lab 5 for the correlation between a sinusoid and a complex exponential. The interpretation is the same. When we compute for C_k values, we are computing the correlation between the signal s(t) and a complex exponential with frequency $2\pi k / T$. This correlation portrays how much of a particular complex exponential is contained in the signal s(t).

Partial Series

The synthesis formula of (2.3) has infinite limits. Thus we need an infinite number of complex exponentials to represent our signal. In practical situations, however, we can only include a finite number of terms in the sum. Thus if we use only the first N positive and negative frequencies, plus the DC term (at k = 0), our approximate synthesis equation becomes

$$s(t) \approx \sum_{k=-N}^{N} C_k e^{j\frac{2\pi k}{T}t} \tag{2.6}$$

Fourier series theory shows that this approximation becomes better as $N \rightarrow \infty$. Alternatively, we can say that the mean-squared value of the difference between s(t) and the approximation tends to zero as $N \rightarrow \infty$. It can be shown that

$$MS\left(s(t) - \sum_{k=-N}^{N} C_k e^{j\frac{2\pi k}{T}t} \right) = MS\left(s(t)\right) - \sum_{k=-N}^{N} \left|C_k\right|^2 \tag{2.7}$$
$$\rightarrow 0 \text{ as } N \rightarrow \infty$$

***T*-Second Fourier Series**

If a signal s(t) is periodic with period T , then it is also periodic with period 2T , and period 3T, etc. Thus when using the Fourier series, we are at liberty to choose the value of T. Typically, we choose T to be the smallest period (or the fundamental period of s(t)). However, there are also situations where one may not. A good example would be if one wishes to perform spectral analysis/synthesis of two or more periodic signals that have different fundamental periods.

One solution would be to write a separate Fourier series for each signal. Then each Fourier series would be based on a different harmonic series of frequencies. Another solution would be to choose T to be a multiple of the fundamental periods of both signals.

When a T value is explicitly specified in a Fourier series, we will call this the T -second Fourier series. In determining the T values let us compare a T -second Fourier series to a 2T - second Fourier series. The T -second Fourier series has components at the frequencies

$$\ldots,\ -2\omega_0,\ -\omega_0,\ 0,\ \omega_0,\ 2\omega_0,\ \ldots \tag{2.8}$$

where,

$$\omega_0 = \frac{2\pi}{T} \tag{2.9}$$

and the 2T- second Fourier series has components at the frequencies.

$$\ldots,\ -2\omega_0',\ -\omega_0',\ 0,\ \omega_0',\ 2\omega_0',\ \ldots = \ldots,\ -\omega_0,\ -\frac{\omega_0}{2},\ 0,\ \frac{\omega_0}{2},\ \omega_0,\ \ldots \tag{2.10}$$

where

$$\omega_0' = \frac{2\pi}{2T} = \frac{\omega_0}{2} \tag{2.11}$$

We can see that the 2T -second Fourier series decomposes s(t) into frequency components with half the separation of that of the T -second Fourier series. The additional coefficients in the 2T -Fourier series are zero, and it turns out that the nonzero coefficients are the same as for the T -second Fourier series. It can be shown that with C_k and C_k' denoting the T -second and 2T -second Fourier coefficients, respectively, then

$$\alpha_k = \begin{cases} \alpha_{k/2} & k \text{ even} \\ 0 & k \text{ odd} \end{cases} \tag{2.12}$$

Fourier series analysis/synthesis can be performed over one fundamental period or over any number of fundamental periods.

Aperiodic Continuous-Time Signals

The Fourier series can also be applied when the signal s(t) is not periodic. We will then be determining the spectrum of the signal over a finite segment, (for example from time t_1 to time t_2), and performing Fourier series analysis/synthesis over the finite segment only. The Fourier coefficients are then,

$$C_k = \frac{1}{T}\int_{t1}^{t2} s(t)e^{-j\frac{2\pi k}{T}t}dt \tag{2.13}$$

where $T = t_2 - t_1$, and then we have

$$s(t) = \sum_{k=-\infty}^{\infty} C_k e^{j\frac{2\pi k}{T}t},\ \text{for } t_1 \le t \le t_2 \tag{2.14}$$

This gives us an idea of the frequency content of the signal during the given time interval.

2.4 Some MATLAB commands for this lab

- **Fourier Series Synthesis in** MATLAB**:** The function fourier_synthesis is a function that is provided to compute the approximate T -second Fourier series synthesis formula, equation. Its inputs are the period T and a set of 2N + 1 Fourier coefficients. Its output is the synthesized signal. The calling command is

  ```
  >> [ss,tt] = fourier_synthesis(CC, T, periods, Ns);
  ```

  where CC is a vector containing the Fourier coefficients, T is the interval (in seconds) over which the Fourier series is applied. periods is the (integer) number of periods to include in the resynthesis; periods defaults to a value of 1 if not provided. The optional parameter Ns specifies how many samples per period to include in the output signal.

It is assumed that CC contains the coefficients C_{-N} ...C_N. (N is implicitly determined from the length of CC.) Thus, CC has length 2N +1, the CC(n) element contains the Fourier series coefficient C_{n-N-1}. Further, note that the C_0 coefficient falls at CC(N+1).

The two returned parameters are the signal vector ss and the corresponding signal support vector tt.

- **Fourier Series Analysis in** MATLAB**:** The function fourier_analysis is the complement to the function fourier_synthesis. It performs T -second Fourier series analysis on an input signal. The calling command is

  ```
  >> [CC,ww] = fourier_analysis(ss,T,N);
  ```

  where ss is a vector containing the signal samples, T is the interval T in seconds over which the Fourier series is to be computed, and N is the number of positive harmonics to include in the analysis. (2N+1 is the total number of harmonics.) It is assumed that ss contains samples of the signal to be analyzed over the interval [0,T].

  The outputs are the vectors CC, which contains the 2N + 1 Fourier coefficients, and ww, which contains the frequencies (in Hertz) associated with each Fourier coefficient.

3 Guided Exercises

1. In this problem, you will "hand tune" the amplitudes and phases of three sinusoids so that their sum matches a "target" periodic signal as well as possible. The signals are considered to be continuous-time. One could do this task analytically or numerically using the Fourier series analysis formula, but we want you to gain the insight that results from doing it manually. A graphical MATLAB program has been written to facilitate this procedure.

   Download the files sinsum.m and sinsum.fig and execute sinsum. MATLAB will bring up a GUI window with three sinusoids (colored, dotted lines), the sum of these three sinusoids (the black, dashed line), and a target periodic signal (the black, solid line). The frequencies of the sinusoids are ω_0, $2\omega_0$, $3\omega_0$, where ω_0 is the fundamental frequency of the target signal.

   As stated earlier, the goal of this problem is to adjust the amplitudes and phases of the three sinusoids to approximate the target signal as closely as possible. You can enter the amplitude and phase for each sinusoid in the spaces provide in the GUI window, or using the mouse, you can click-and-drag each sinusoid to change its amplitude and phase. In addition to displaying the three sinusoids, their sum, and the target signal, the GUI window also shows the mean-squared error between the sum and the target signals.

Use sinsum.m to hand tune the amplitudes and phases of the three sinusoids to make the mean squared error as small as you can.
(Hint: You should be able achieve an MSE less than 0.24. You will receive +2 bonus points if you can achieve an MSE less than 0.231.)
(Hint: In attempting to minimize the MSE you might try to adjust one sinusoid to minimize the MSE, then another, then another. After doing all three, go back and see if readjusting them in a "second round" has any benefits.)

- Include the resulting figure window in your report. (On Windows systems, use the "Copy to Clipboard" button to copy the figure, then you can simply paste it into a Word or similar document. There is also a "Print Figure" button for other systems if you can't get access to a PC.)

2. In this problem you will simply apply fourier_synthesis to a given set of Fourier coefficients and find the resulting continuous-time signal. Download the file fourier_synthesis.m. Use it to generate an approximation to the signal with the following Fourier coefficients:

$$C_k = \begin{cases} -\left(\dfrac{2}{\pi k}\right)^2 & k = \pm 1,\ \pm 3,\ \pm 5,\ \ldots \\ 0 & k = 0,\ \pm 2,\ \pm 4,\ \ldots \end{cases}$$

Let T = 0.1 seconds, and generate 5 periods of the signal. Use N = 20, giving you 41 Fourier series coefficients. (Hint: First, define a frequency support vector, kk = -20:20. Then, generate CC from kk and set all even harmonics to zero.)

- Use stem to plot the magnitude of the Fourier coefficients. Use your kk vector as the x-axis.
- Use plot to plot samples of the continuous-time signal that fourier_synthesis returns versus time in seconds.
- What kind of signal is this?

3. In this problem you will use the Fourier series analysis and synthesis formula to see how the accuracy of the approximate synthesis formula (2.6) depends on N.

Download the files lab6_data.mat and fourier_analysis.m. lab6_data.mat contains the variables step_signal and step_time, which are the signal and support vectors for the samples of a periodic continuous-time signal with fundamental period T_0 = 1 second. Note that there are N_s = 16384 samples in one fundamental period. (step_signal and step_time include several fundamental periods, but you'll be dealing with only one period in several parts of this problem. As such, you might find it useful to create a one-period version of step_signal.)

(a) First, let us examine step_signal.
- Use plot to plot step_signal versus its support vector.
- Compute the mean-squared value of step_signal.

(b) Use fourier_analysis to perform a T_0 second Fourier series analysis over *a single period* of step_signal with N = 50.

- Use subplot and stem to plot the magnitude and phase of the resulting Fourier series coefficients. Make sure that your x-axis is given in frequency.

(c) (Resynthesize FS approximations) Use fourier_analysis and fourier_synthesis to generate an approximations of step_signal with N = 25, 50, 100, and 200. (Perform T_0 - second Fourier analysis and synthesis over a single period of the signal for each N. Be sure to resynthesize a single period with N_s = 16384 samples.)

- Use plot and subplot to plot your resynthesized signals for each N in separate panels of a subplot array.
- Calculate the mean-squared error of the resynthesis for each value of N.
- Compute the sum of the squared magnitudes of CC for each value of N.
- Find and document a relationship between the mean-squared errors and the sum of squared magnitudes of CC you have computed. (Hint: Consider the mean-squared value that you computed for step_signal.)

(d) (Meet an MSE target) Find the smallest value of N for which the mean-squared error of the resynthesis is less than 0.5% of the mean-squared value of step_signal.

- Include this value in your report

4 Review

1. None.

5 Lab Report

1. The first page of your Lab report should be a cover sheet with your name, USC ID and Lab #. Please note that all reports should be typed.
2. Answer all the questions which were asked in the lab report. Kindly display the code lines you executed to arrive at your answer along with figures to support them. Please give written explanation or put comment lines where necessary. Please note that each figure should have proper labels for the x and y axis and should have a suitable title.
3. Answer the review questions.
4. Submit a printout of your completed M-file documenting all the lab exercises.

This lab document and the figures contained were adapted from a University of Michigan Signals and Systems course lab handout (2002).