GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL of ELECTRICAL and COMPUTER ENGINEERING

EE 2200 Fall 1998

Lab #2: Introduction to Complex Exponentials

Date: week of 5 Oct 1998

Lab is held in College of Computing Building, room 309.

This is *the official* Lab#2 description; it is nearly *identical* to the one in Appendix C.2 of the text, but there are some small differences in Sections 4 and 5.

The Warm-up section of each lab must be completed in Lab and the steps marked *Instructor Verification* must also be signed off **during the lab time**. One of the laboratory instructors must verify the appropriate steps by initialing on the **Instructor Verification** line. When you have completed a step that requires verification, simply raise your hand and demonstrate the step to the instructor.

Lab Report: It is only necessary to turn in Sections 4 and 5 with explanations as this week's lab report. Staple the **Instructor Verification** sheet to the end of your lab report as evidence that the appropriate steps were witnessed by the instructor.

The report will **due during the week of 12-Oct at the start of your lab**.

---

1 Introduction

The goal of this laboratory is to gain familiarity with complex numbers and their use in representing sinusoidal signals such as $x(t) = A\cos(\omega t + \phi)$ as complex exponentials $z(t) = Ae^{j\phi}e^{j\omega t}$. The real part operator is the key:

$$x(t) = A\cos(\omega t + \phi) = \Re e\{Ae^{j\phi}e^{j\omega t}\}$$

2 Overview

Manipulating sinusoid functions using complex exponentials turns trigonometric problems into simple arithmetic and algebra. In this lab, we first review the complex exponential signal and the phasor addition property needed for adding cosine waves. Then we will use MATLAB to make plots of phasor diagrams that show the vector addition needed when combining sinusoids.

2.1 Complex Numbers in MATLAB

MATLAB can be used to compute complex-valued formulas and also to display the results as vector or "phasor" diagrams. For this purpose several new functions have been written and are available on the *DSP First CD-ROM*. Make sure that this toolbox has been installed by doing `help` on the new M-files: `zvect`, `zcat`, `ucplot`, `zcoords`, and `zprint`. Each of these functions can plot (or print) several complex numbers at once, if the input is formed into a vector of complex numbers. The following example will plot five vectors all on one graph:

CD-ROM
DSP First MATLAB *Toolbox*

```
zvect( [ 1+j, j, 3-4*j, exp(j*pi), exp(2i*pi/3) ] )
```

Here are some of MATLAB's complex number operators:

`conj`	Complex conjugate
`abs`	Magnitude
`angle`	Angle (or phase) in radians
`real`	Real part
`imag`	Imaginary part
`i,j`	pre-defined as $\sqrt{-1}$
`x = 3 + 4i`	`i` suffix defines imaginary constant
`exp(j*theta)`	Function for the complex exponential $e^{j\theta}$

Each of these functions takes a vector (or matrix) as its input argument and operates on each element of the vector.

CD-ROM

Z Drill

Finally, there is a complex numbers drill program called:

```
zdrill
```

which generates complex number problems and tests your answers. *Please spend some time with this drill since it is very useful in helping you to get a feel for complex arithmetic.*

When unsure about a command, use `help`.

2.2 Sinusoid Addition Using Complex Exponentials

Recall that sinusoids may be expressed in the form:

$$x(t) = A\cos(2\pi f_0 t + \phi) = \Re e\left\{Ae^{j\phi}e^{j2\pi f_0 t}\right\} \tag{1}$$

Consider the sum of cosine waves given by (2)

$$x(t) = \sum_{k=1}^{N} A_k \cos(2\pi f_0 t + \phi_k) \tag{2}$$

where each cosine waves in the sum has the same frequency, f_0. This sum is difficult to simplify using trigonometric identities, but it reduces to an algebraic sum of complex numbers when solved using complex exponentials. This is the *Phasor Addition Rule* presented in the text. A summary of the phasor addition rule using the complex exponential representation of the cosines (1) is:

$$\begin{aligned}
x(t) &= \Re e\left\{\sum_{k=1}^{N} Z_k e^{j2\pi f_0 t}\right\} && (3)\\
&= \Re e\left\{\left(\sum_{k=1}^{N} Z_k\right) e^{j2\pi f_0 t}\right\} && (4)\\
&= \Re e\left\{Z_s e^{j2\pi f_0 t}\right\} && (5)\\
&= A_s \cos(2\pi f_0 t + \phi_s) && (6)
\end{aligned}$$

where

$$Z_k = A_k e^{j\phi_k} \tag{7}$$

and

$$Z_s = \sum_{k=1}^{N} Z_k = A_s e^{j\phi_s} \tag{8}$$

We see that the sum signal $x(t)$ is a single sinusoid and it is periodic with period $T_0 = 1/f_0$.

2.3 Harmonic Sinusoids

There is an important extension where $x(t)$ is the sum of N cosine waves whose frequencies (f_k) are all multiples of one basic frequency f_0.

$$f_k = k f_0 \qquad \text{(HARMONIC FREQUENCIES)}$$

The sum of N cosine waves given by (2) becomes

$$x(t) = \sum_{k=1}^{N} A_k \cos(2\pi k f_0 t + \phi_k) = \Re e \left\{ \sum_{k=1}^{N} Z_k \, e^{j2\pi k f_0 t} \right\} \tag{9}$$

This particular signal $x(t)$ is also periodic with period $T_0 = 1/f_0$. The frequency f_0 is called the *fundamental frequency*, and T_0 is called the *fundamental period*.

3 Warm-up

The instructor verification sheet is at the end of this lab.

3.1 Complex Numbers

To exercise your understanding of complex numbers, do the following:

(a) Define $z_1 = -1 + j0.3$ and $z_2 = 0.8 + j0.7$. Enter these in MATLAB and plot them with `zvect`, and print them with `zprint`.

(b) Compute the conjugate z^* and the inverse $1/z$ for both z_1 and z_2 and plot the results. In MATLAB, see `help conj`. Display the results numerically with `zprint`.

(c) Compute $z_1 + z_2$ and plot. Use `zcat` to show the sum as vectors head-to-tail. Use `zprint` to display the results numerically.

(d) Compute $z_1 z_2$ and z_1/z_2 and plot. Use the `zvect` plot function to show how the angles of z_1 and z_2 determine the angles of the product and quotient. Use `zprint` to display the results numerically.

(e) Work a few problems on the complex number drill program. To start the program simply type `zdrill`.

Instructor Verification (separate page)

3.2 Sinusoidal Synthesis with an M-File

Write an M-file that will synthesize a waveform in the form of (2). Write the function without using `for` loops. Take advantage of the fact that matrix-vector multiplication computes a sum of products. For example,

$$\mathbf{c} = \mathbf{A}\mathbf{b} \qquad \Longrightarrow \qquad c_n = \sum_{k=1}^{L} a_{nk} b_k \tag{10}$$

where c_n represents the n^{th} element of the vector $\mathbf{c}$, a_{nk} is the element in the n^{th} row and k^{th} column of the matrix $\mathbf{A}$, b_k is the k^{th} element of the vector $\mathbf{b}$, and L is the number of columns in $\mathbf{A}$. The first few statements of the M-file should look like

```
function        xx = sumcos(f, Z, fs, dur)
%SUMCOS   Function to synthesize a sum of cosine waves
%  usage:
%      xx = sumcos(f, Z, fs, dur)
%       f = vector of frequencies
%              (these could be negative or positive)
%       Z = vector of complex exponentials: Amp*eˆ(j*phase)
%      fs = the sampling rate in Hz
%     dur = total time duration of signal
%
%    Note: f and Z must be the same length.
%             Z(1) corresponds to frequency f(1),
%             Z(2) corresponds to frequency f(2), etc.
```

The MATLAB syntax `length(f)` returns the number of elements in the vector `f`, so we do not need a separate input argument for number of frequencies. On the other hand, the programmer should provide error checking to make sure that the lengths of `f` and `Z` are the same. It is possible (although not required) to complete this function in a single line. For some hints, please refer to the review of matrix multiplication in the *Using* MATLAB Appendix of the text.

In order to use this M-file to synthesize periodic waveforms, you would simply choose the entries in the frequency vector to be integer multiples of the desired fundamental frequency. Try the following test and plot the result.

```
xx = sumcos([20],   [1],    200, 0.25);
xx = sumcos([20 40], [1 1/2], 200, 0.25);
xx = sumcos([20 40 60 80], [1 -1 1 -1], 200, 0.25);
```

Instructor Verification (separate page)

4 Exercises: Complex Exponentials

4.1 Representation of Sinusoids with Complex Exponentials

In MATLAB consult `help` on `exp`, `real` and `imag`.

(a) Generate the signal $x(t) = Ae^{j(\omega_0 t + \phi)}$ for $A = 3$, $\phi = 0.1\pi$, and $\omega_0 = 2\pi(1250)$. Take a range for t that will cover 2 or 3 periods.

(b) Plot the real part versus t and the imaginary part versus t. Use `subplot(2,1,i)` to put both plots in the same window.

(c) Verify that the real and imaginary parts are sinusoids and that they have the correct frequency, phase and amplitude.

4.2 Verify Addition of Sinusoids Using Complex Exponentials

Generate four sinusoids with the following amplitudes and phases:

$$\begin{aligned} x_1(t) &= 7\cos(2\pi(1250)t + 0.1\pi) \\ x_2(t) &= 7\cos(2\pi(1250)t + 0.4\pi) \\ x_3(t) &= 7\cos(2\pi(1250)t + 0.6\pi) \\ x_4(t) &= 7\cos(2\pi(1250)t + 0.9\pi) \end{aligned}$$

(a) Make a plot of all four signals over a range of t that will exhibit approximately 3 cycles. Make sure the plot includes negative time so that the phase at $t = 0$ can be measured. *In order to get a smooth plot make sure that your have at least 20 samples per period of the wave.*[1]

(b) Verify that the phase of all four signals is correct at $t = 0$, and also verify that each one has the correct maximum amplitude. Use `subplot(3,2,i)` to make a six-panel subplot that puts all of these plots on the same page.

(c) Create the sum sinusoid via: $x_5(t) = x_1(t) + x_2(t) + x_3(t) + x_4(t)$. Make a plot of $x_5(t)$ over the same range of time as used in the last plot. Include this as the lower panel in the plot by using `subplot(3,1,3)`.

(d) Measure the magnitude and phase of $x_5(t)$ directly from the plot. In your lab report, include this plot with sufficient annotation to show how the magnitude and phase were measured.

(e) Now do some complex arithmetic; create the complex amplitudes corresponding to the sinusoids $x_i(t)$:

$$z_i = A_i e^{j\phi_i} \qquad\qquad i = 1, 2, 3, 4, 5$$

Give the numerical values of z_i in polar *and* Cartesian form.

(f) Verify that $z_5 = z_1 + z_2 + z_3 + z_4$. Show a plot of these five complex numbers as vectors. Use the MATLAB functions `zvect`, `zcat` and `zprint` discussed in the Warm-up.

(g) Relate the magnitude and phase of z_5 to the plot of $x_5(t)$.

When unsure about a command, use `help`.

5 Periodic Waveforms

Each of the following waveforms can be synthesized with a simple call to the function `sumcos`. Plot a short section of the signal to observe its characteristic shape.

Note: It is important to have a sampling rate that is at least *twice as high as the highest frequency component in your signal*, a fact that will be explained when sampling is discussed in Chapter 4. However, for this lab you should just choose a very large number for f_s to get a smooth plot; at least ten times the highest frequency.

[1] If you have already studied sampling in Chapter 4, then this requirement of 20 samples per period amounts to considerable oversampling.

(a) Try your `sumcos` M-file with the fundamental $f_0 = 20$ Hz, $f_k = kf_0$, and

$$Z_k = \begin{cases} \dfrac{j}{k} & k = 1, 3, 5, 7, \ldots \quad \text{(i.e., an odd integer)} \\ 0 & k = 0, 2, 4, 6, 8, \ldots \end{cases} \tag{11}$$

Specify the duration to get three periods of the waveform.

Make plots for three different cases: $N = 5$, 10, and 25 (where N is the number of cosines). Use a three-panel subplot to show all three signals together. Explain how the period of synthesized waveform is related to the fundamental frequency.

Explain what happens as $N \to \infty$. What wave shape do the plots converge to? Although the waveshape is converging to a simple form, it is not perfect. Describe any unusual features in the converging waveform as $N \to \infty$.

(b) It is informative to listen to these signals as a different number of coefficients are added. Repeat the synthesis from part (a) with $f_0 = 1$ kHz and listen to the cases where $N = 1, 2, 3, 4, 5$ and 10. You will need about 1 second of the signal to hear differences. Comment on how the sound changes as the number of coefficients N increases.

Note: when using `sound(x, fs)` or `soundsc(x, fs)`,[2] the sampling frequency should be very high to avoid aliasing effects (discussed in Chapter 4).

(c) Now try the coefficients

$$Z_k = \begin{cases} \dfrac{j(-1)^k}{2\pi k} & k = 1, 2, 3, 4, \ldots \\ \pi & k = 0 \end{cases} \tag{12}$$

Choose the fundamental frequency to be $f_0 = 20$ Hz. Compute the signal for three cases: $N = 5$, 10, and 50, and plot all three functions together with a three-panel subplot. What waveshape is approximated with this sum of cosines as $N \to \infty$? Explain how the period of synthesized waveshape is related to the fundamental.

[2]The MATLAB function called `soundsc` will automatically scale the signal to the maximum allowed by the sound card.

Lab #2
EE-2200
Fall-1998
Instructor Verification Sheet

Staple this page to the end of your Lab Report.

Name: ____ Date of Lab: __________

Part 3.1 Complex number exercises and drill:

Verified: _________________

Part 3.2 Complete `sumcos.m` and plot test waveforms:

Verified: _________________