Signals and Systems

Lab1: Introduction to Matlab in Signals and Systems

Instructor: Prof. Xiaoming Chen
Zhejiang University

03/07/2024

Outline

Course Introduction

Assignments of computer exercises in parallel with traditional written problems can help to develop a stronger intuition and a deeper understanding of linear systems and signals.

- Reference: Computer Explorations in SIGNALS AND SYSTEMS USING MATLAB

- Tool: MATLAB
  (Download inside the campus:
  https://pan.zju.edu.cn/share/eacc667811fae1e16f122ed724)

Outline

Basic MATLAB Functions for Representing Signals

How to represent, manipulate, and analyze basic signals and systems in MATLAB.

- Signals will be represented by a row or column vector.
- All vectors represented in MATLAB are indexed starting with 1, i.e., $y(1)$ is the first element of the vector y.
- For example, to represent the discrete-time signal

$$\mathrm{x}\left[\mathrm{n}\right] = \begin{cases} 2\mathrm{n}, -3 \leq \mathrm{n} \leq 3, \\ 0, \text{otherwise} \end{cases}$$

$\gg \mathrm{n} = [-3 : 3];$
$\gg \mathrm{x} = 2 * \mathrm{n};$
To plot this signal: stem(n,x)

Basic MATLAB Functions for Representing Signals

- Examine the signal over a wider range of indices: extend both n and x.
  $\gg$ n $= [-5 : 5]$;
  $\gg$ x $= [0\ 0\ \mathrm{x}\ 0\ 0]$;
- Greatly extend the range of the signal: use the function **zeros**.
  $\gg$ n $= [-100 : 100]$;
  $\gg$ x $= [\mathrm{zeros}(1, 95)\ \mathrm{x}\ \mathrm{zeros}(1, 95)]$;

Basic MATLAB Functions for Representing Signals

- Define $x_1[n]$ to be the discrete-time unit impulse function and $x_2[n]$ to be a time-advanced version of $x_1[n]$, i.e., $x_1[n] = \delta[n]$ and $x_2[n] = \delta[n+2]$.
  $\gg$ nx1 = [0 : 10];
  $\gg$ x1 = [1 zeros(1, 10)];
  $\gg$ nx2 = [−5 : 5];
  $\gg$ x2 = [zeros(1, 3) 1 zeros(1, 7)];
- Without the index vectors, and simply typed stem(x1) and stem(x2), you would make plots of the signals $\delta[n-1]$ and $\delta[n-4]$ and not of the desired signals.

Representing a continuous-time signal by discrete-time samples

Representing a continuous-time signal by discrete-time samples

- Two simple methods are to use the colon operator with the optional step argument. Exp: create a vector that covered the interval $-5 \leq t \leq 5$ in steps of 0.1 seconds
t=[-5:0.1:5];t=linspace(-5,5,101).
- Sinusoids and complex exponentials are important signals for the study of linear systems. MATLAB provides several functions that are useful for defining such signals, like sin, cos, exp.
$\gg$ plot(t, sin(t))
- In general, use **stem** to plot short discrete-time sequences, and **plot** for sampled approximations of continuous-time signals or for very long discrete-time signals where the number of stems grows unwieldy.

Representing a continuous-time signal by discrete-time samples

To represent the discrete-time signal $x[n] = e^{j(\pi/8)n}$ for $0 \leq n \leq 32$

- $\gg$ n = [0 : 32];
  $\gg$ x = exp(j * (pi/8) * n);
- To plot complex signals, you must plot their real and imaginary parts, or magnitude and angle, separately. The MATLAB functions **real**, **imag**, **abs**, and **angle** compute these functions of a complex vector on an term-by-term basis.
- **angle**: phase of the complex numbers in radians.
- Add (+), subtract (−), multiply (.∗), divide (./), scale (2∗) and exponentiate (.^)signals.

Representing a continuous-time signal by discrete-time samples

- For multiplying, dividing and exponentiating on a term-by-term basis: $.*$
- Matrix multiplication operator: $*$
- Similar: $./$ and $.^\wedge$ when operating on vectors term-by-term, $/$ and $^\wedge$ are matrix operators.

Representing a continuous-time signal by discrete-time samples

Commands to label plots appropriately, and print them out

- **title**
- **xlabel**, **ylabel**
- **print**

Representing a continuous-time signal by discrete-time samples

Ability to write M-files.

- Two types of M-files: functions and command scripts. A command script is a text file of MATLAB commands whose filename ends in .m in the current working directory or elsewhere on your MATLABPATH.
- %prob1.m
  n = [0:16];
  xl = cos (pi*n/4) ;
  y1 = mean(x1);
  stem(n,xl)
  title('x1 = cos(pi*n/4)')
  xlabel('n (samples)')
  ylabel('x1[n] ')

Representing a continuous-time signal by discrete-time samples

An M-file implementing a function is a text file with a title ending in .m whose first word is function.

- The rest of the first line of the file specifies the names of the input and output arguments of the function.
- ```
  function [y,z] = foo(x)
  %[y,z] = foo(x) accepts a numerical argument x and returns two
  %arguments y and z, where y is 2*x and z is (5/9)*(x-32)
  y = 2*x;
  z = (5/9) * (x-32) ;
  ```
  ```

Discrete-Time Sinusoidal Signals

- Discrete-time complex exponentials play an important role in the analysis of discrete-time signals and systems. A discrete-time complex exponential has the form α^n, where α is a complex scalar.
- The discrete-time sine and cosine signals can be built from complex exponential signals by setting $\alpha = e^{\pm i\omega}$.

$$\cos(\omega \mathrm{n}) = \frac{1}{2}\left(\mathrm{e}^{\mathrm{i}\omega \mathrm{n}} + \mathrm{e}^{-\mathrm{i}\omega \mathrm{n}}\right)$$
$$\sin(\omega \mathrm{n}) = \frac{1}{2\mathrm{i}}\left(\mathrm{e}^{\mathrm{i}\omega \mathrm{n}} - \mathrm{e}^{-\mathrm{i}\omega \mathrm{n}}\right)$$

Basic Problems

Problem 1

Consider the signal

$$x_k[n] = \sin(\omega_k n)\,,$$

where $\omega_k = 2\pi k/5$. For $x_k[n]$ given by $k = 1$, 2, 4, and 6, use **stem** to plot each signal on the interval $0 \le n \le 9$. All of the signals should be plotted with separate axes in the same figure using **subplot**. How many unique signals have you plotted? If two signals are identical, explain how different values of ω_k can yield the same signal.

Problem 2

Now consider the following three signals

$$x_1[n] = \cos\left(\frac{2\pi n}{N}\right) + 2\cos\left(\frac{3\pi n}{N}\right)\,,$$

$$x_2[n] = 2\cos\left(\frac{2n}{N}\right) + \cos\left(\frac{3n}{N}\right)\,,$$

$$x_3[n] = \cos\left(\frac{2\pi n}{N}\right) + 3\sin\left(\frac{5\pi n}{2N}\right)\,.$$

Assume $N = 6$ for each signal. Determine whether or not each signal is periodic. If a signal is periodic, plot the signal for two periods, starting at $n = 0$. If the signal is not periodic, plot the signal for $0 \le n \le 4N$ and explain why it is not periodic. Remember to use **stem** and to appropriately label your axes.

Intermediate Problems

Problem 3

Plot each of the following signals on the interval $0 \leq n \leq 31$:

$$x_1[n] = \sin\left(\frac{\pi n}{4}\right)\cos\left(\frac{\pi n}{4}\right),$$
$$x_2[n] = \cos^2\left(\frac{\pi n}{4}\right),$$
$$x_3[n] = \sin\left(\frac{\pi n}{4}\right)\cos\left(\frac{\pi n}{8}\right).$$

What is the fundamental period of each signal? For each of these three signals, how could you have determined the fundamental period without relying upon MATLAB?

Problem 4

Consider the signals you plotted in Problem 2 and 3. Is the addition of two periodic signals necessarily periodic? Is the multiplication of two periodic signals necessarily periodic? Clearly explain your answers.

Linearity, time invariance, stability, causality, and invertibility

Problem 5

Basic Problems

For these problems, you are told which property a given system does not satisfy, and the input sequence or sequences that demonstrate clearly how the system violates the property. For each system, define MATLAB vectors representing the input(s) and output(s). Then, make plots of these signals, and construct a well reasoned argument explaining how these figures demonstrate that the system fails to satisfy the property in question.

(a). The system $y[n] = \sin((\pi/2)x[n])$ is not linear. Use the signals $x_1[n] = \delta[n]$ and $x_2[n] = 2\delta[n]$ to demonstrate how the system violates linearity.

(b). The system $y[n] = x[n] + x[n+1]$ is not causal. Use the signal $x[n] = u[n]$ to demonstrate this. Define the MATLAB vectors x and y to represent the input on the interval $-5 \leq n \leq 9$, and the output on the interval $-6 \leq n \leq 9$, respectively.

Implementing a First-Order Difference Equation

$$y[n] = ay[n-1] + x[n], \qquad (1.6)$$

Problem 6

(a). Write a function `y=diffeqn(a,x,yn1)` which computes the output $y[n]$ of the causal system determined by Eq. (1.6). The input vector `x` contains $x[n]$ for $0 \leq n \leq N-1$ and `yn1` supplies the value of $y[-1]$. The output vector `y` contains $y[n]$ for $0 \leq n \leq N-1$. The first line of your M-file should read

```
function y = diffeqn(a,x,yn1)
```

Hint: Note that $y[-1]$ is necessary for computing $y[0]$, which is the first step of the autoregression. Use a `for` loop in your M-file to compute $y[n]$ for successively larger values of n, starting with $n = 0$.

(b). Assume that $a = 1$, $y[-1] = 0$, and that we are only interested in the output over the interval $0 \leq n \leq 30$. Use your function to compute the response due to $x_1[n] = \delta[n]$ and $x_2[n] = u[n]$, the unit impulse and unit step, respectively. Plot each response using `stem`.

Continuous-Time Complex Exponential Signals

Problem 7

Consider the continuous-time sinusoid

$$x(t) = \sin(2\pi t/T) .$$

A symbolic expression can be created to represent $x(t)$ within MATLAB by executing

```
>>  x = sym('sin(2*pi*t/T)');
```

The variables of **x** are the single character strings **'t'** and **'T'**. The function **ezplot** can be used to plot a symbolic expression which has only one variable, so you must set the fundamental period of $x(t)$ to a particular value. If you desire $T = 5$, you can use **subs** as follows

```
>>  x5 = subs(x,5,'T');
```

Thus **x5** is a symbolic expression for $\sin(2\pi t/5)$. Create the symbolic expression for **x5** and use **ezplot** to plot two periods of $\sin(2\pi t/5)$, beginning at $t = 0$. If done correctly, your plot should be as shown in Figure 1.2.

Continuous-Time Complex Exponential Signals

Problem 8

In the following problems you will write M-files for extracting the real and imaginary components, or the magnitude and phase, of a symbolic expression for a complex signal.

(d). Store in `x` a symbolic expression for the signal

$$x(t) = e^{i2\pi t/16} + e^{i2\pi t/8}.$$

Remember to use `'i'` rather than `'j'` within symbolic expressions to represent $\sqrt{-1}$. The function `ezplot` cannot be used directly for plotting $x(t)$, since $x(t)$ is a complex signal. Instead, the real and imaginary components must be extracted and then plotted separately.

(e). Write a function `xr=sreal(x)` which returns a symbolic expression `xr` representing the real part of $x(t)$. If your function is working properly, `ezplot(xr)` will plot the real component of $x(t)$. Similarly, write a function `xi=simag(x)` which returns a symbolic expression `xi` representing the imaginary component of $x(t)$. The first line of the M-file `sreal.m` should be

```
function xr = sreal(x)
```

You can then use `compose('real(x)',x)` to create a symbolic expression for the real component of $x(t)$. Use `ezplot` and the functions you created to plot the real and imaginary components of $x(t)$ on the interval $0 \le t \le 32$. Use a separate plot for each component. What is the fundamental period of $x(t)$?

Outline

Assignments

Issued: March 7, 2024 Due: March 16, 2024

You should hand in the Matlab code (.m files), graphics, audio files and a brief description of your reasoning as well as comments if any. Please make sure that your Matlab code can be run on Matlab R2007b or higher version. You should pack all of your files into a .rar or .zip file, titled as xxxxxxx(your student ID) xxxx(your name) Lab pre, and then upload to blackboard before 11:59pm of the due day.