Signals and Systems

Lecture2: Linear Time-Invariant Systems

Instructor: Prof. Xiaoming Chen
Zhejiang University

03/21/2024

Outline

Tutorial: filter

The **filter** command computes the output of a causal, LTI system for a given input when the system is specified by a linear constant -coefficient difference equation,

$$\sum_{k=0}^{K} a_k y[n-k] = \sum_{m=0}^{M} b_m x[n-m]. \tag{1}$$

If **x** is a MATLAB vector containing the input x[n] on the interval $n_x \leq n \leq n_x + N_x - 1$ and the vectors **a** and **b** contain the coefficients a_k and b_k, then **y=filter(b,a,x)** returns the output of the causal LTI system satisfying

$$\sum_{k=0}^{K} a(k+1)y(n-k) = \sum_{m=0}^{M} b(m+1)x(n-m). \tag{2}$$

Tutorial: filter

To specify the system described by the difference equation $y[n] + 2y[n-1] = x[n] + 3x[n-1]$, define these vectors as **a**=[1 2] and **b**=[1 3].

- Define coefficient vectors **a1** and **b1** to describe the causal LTI system specified by $y[n] = 0.5x[n] + x[n-1] + 2x[n-2]$.
- Define coefficient vectors **a2** and **b2** to describe the causal LTI system specified by $y[n] = 0.8y[n-1] + 2x[n]$.
- Define coefficient vectors **a3** and **b3** to describe the causal LTI system specified by $y[n] - 0.8y[n-1] = 2x[n-1]$.
- For each of these three systems, use **filter** to compute the response $y[n]$ on the interval $1 \leq n \leq 4$ to the input signal $x[n] = nu[n]$. You should begin by defining the vector x=[1 2 3 4], which contains x[n] on the $1 \leq n \leq 4$ interval.

Tutorial: lsim with Differential Equations

The function **lsim** can be used to simulate the output of continuous-time, causal LTI systems described by linear constant-coefficient differential equations of the form

$$\sum_{k=0}^{N} a_k \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b_m \frac{d^m x(t)}{dt^m}. \tag{3}$$

To use **lsim**, the coefficients a_k and b_m must be stored in MATLAB vectors **a** and **b**, respectively, in descending order of the indices k and m.

$$\sum_{k=0}^{N} a(N+1-k) \frac{d^k y(t)}{dt^k} = \sum_{m=0}^{M} b(M+1-m) \frac{d^m x(t)}{dt^m}. \tag{4}$$

Tutorial: lsim with Differential Equations

Note that **a** must contain N+1 elements, which might require appending zeros to **a** to account for coefficients a_k that equal zeros. Similarly, the vector **b** must contain M+1 elements.

» y=lsim(b,a,x,t);

Consider the causal LTI system described by the first-order differential equation

$$\frac{dy(t)}{dt} = -\frac{1}{2}y(t) + x(t), \tag{5}$$

compute and plot the step response of this system.

The actual step response is

$$s(t) = 2(1 - e^{-t/2})u(t). \tag{6}$$

Tutorial: lsim with Differential Equations

- Use **lsim** to compute the response of the causal LTI system described by

$$\frac{dy(t)}{dt} = -2y(t) + x(t), \tag{7}$$

  to the input $x(t) = u(t-2)$, using t=[0:0.5:10].

- The functions **impulse** and **step** can be used to compute the impulse and step response of such systems. Use them to compute the step and impulse responses of the causal LTI system characterized by Eq. (5). Compare the step response computed by **step** with that computed by **lsim**. Compare the signal returned by **impulse** with the exact impulse response given by the derivative of $s(t)$ in Eq. (6).

Outline

Linearity and Time-Invariance

Problem 1

Consider the following three systems:

$$
\begin{array}{ll}
\text{System 1:} & w[n] = x[n] - x[n-1] - x[n-2]\,, \\
\text{System 2:} & y[n] = \cos(x[n])\,, \\
\text{System 3:} & z[n] = n\,x[n]\,,
\end{array}
$$

where $x[n]$ is the input to each system, and $w[n]$, $y[n]$, and $z[n]$ are the corresponding outputs.

(a). Consider the three inputs signals $x_1[n] = \delta[n]$, $x_2[n] = \delta[n-1]$, and $x_3[n] = \big(\delta[n] + 2\,\delta[n-1]\big)$. For System 1, store in `w1`, `w2`, and `w3` the responses to the three inputs. The vectors `w1`, `w2`, and `w3` need to contain the values of $w[n]$ only on the interval $0 \le n \le 5$. Use `subplot` and `stem` to plot the four functions represented by `w1`, `w2`, `w3`, and `w1+2*w2` within a single figure. Make analogous plots for Systems 2 and 3.

(b). State whether or not each system is linear. If it is linear, justify your answer. If it is not linear, use the signals plotted in Part (a) to supply a counter-example.

(c). State whether or not each system is time-invariant. If it is time-invariant, justify your answer. If it is not time-invariant, use the signals plotted in Part (a) to supply a counter-example.

Outline

The pulse Response of Continuous-Time LTI Systems

- An important property of an LTI system is that it is completely defined by its impulse response. If $h(t)$ is the response of a continuous-time LTI system to the unit impulse $\delta(t)$, then

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)x(\tau)d\tau \tag{8}$$

  is the system response to any input $x(t)$.
- LTI systems also can be characterized effectively by their response to a pulse of finite duration as long as the duration of the pulse is chosen to be short enough. Namely, define $\delta^{\Delta}(t)$ to be the pulse

$$\delta^{\Delta}(t) = \begin{cases} \frac{1}{\Delta}, 0 \leq t < \Delta \\ 0, \text{otherwise} \end{cases} \tag{9}$$

The pulse Response of Continuous-Time LTI Systems

Problem 2

Consider the causal LTI system whose input $x(t)$ and output $y(t)$ satisfy the following first-order linear differential equation

$$\frac{dy(t)}{dt} + 3y(t) = x(t). \quad \text{Eq.}(2.20)$$

(a). Analytically derive the unit step response $s(t)$ and the unit impulse response $h(t)$ of the causal LTI system defined by Eq. (2.20). Hint: The impulse response is equal to $ds(t)/dt$. Store in `h` and `s` the values of $h(t)$ and $s(t)$ at each time sample in the vector `t=[-1:0.05:4]`. Plot both `h` and `s` versus `t`.

(b). Use `step` and `impulse` to verify the functions that you computed in Part (a). The functions `step(b,a,t)` and `impulse(b,a,t)` are explained in Tutorial 2.3, the tutorial for `lsim`. The vectors `b` and `a` are determined by the coefficients in Eq. (2.20). For the time vector `t`, use `t=[0:0.05:4]`.

The pulse Response of Continuous-Time LTI Systems

Problem 2

(c). Use `lsim` to simulate the response to $\delta^{\Delta}(t)$ for $\Delta = 0.1$, 0.2, and 0.4. For the time vector, use `t=[-1:0.05:4]`, and store in `d1`, `d2`, and `d3` the corresponding values of $\delta^{\Delta}(t)$ for $\Delta = 0.1$, 0.2, and 0.4, respectively. Remember that $\delta^{\Delta}(\Delta) = 0$, not $1/\Delta$. Store in `h1`, `h2`, and `h3` the simulated response for $\Delta = 0.1$, 0.2, and 0.4, respectively. These vectors contain the simulated values of $h^{\Delta}(t)$ at the time samples given in `t`.

On three separate figures, plot the simulated values of $h^{\Delta}(t)$. On each figure, also include a plot of $h(t)$, which was stored in `h` in Part (a). How does $h^{\Delta}(t)$ compare to $h(t)$ as Δ decreases?

(d). Why is it necessary to ensure that the pulse $\delta^{\Delta}(t)$ has unit area? Namely, what if the response to

$$D^{\Delta}(t) = \begin{cases} 1, & 0 \leq t < \Delta, \\ 0, & \text{otherwise}, \end{cases}$$

The pulse Response of Continuous-Time LTI Systems

Problem 2

(e). Now consider the following function

$$d_a(t) = a\,e^{-at}\,u(t)\,.$$

for $a > 0$. While $d_a(t)$ has infinite duration, most of the signal energy is contained in the interval $0 \leq t \leq 4/a$. For sufficiently large values of a, the response of the LTI system to $d_a(t)$ will for all practical purposes be identical to $h(t)$. What is the area of $d_a(t)$ as a function of $a > 0$?

Use `t=[-1:0.05:4]`, and store in `da1`, `da2`, and `da3` the corresponding values of $d_a(t)$ for $a = 4$, 8, and 16, respectively. For each of the three values of a, use `lsim` to simulate the response of the LTI system to $d_a(t)$. Plot each function on a separate figure using `plot`. On each figure, also include a plot of the impulse response computed in Part (a). How does the response to $d_a(t)$ compare to $h(t)$ for large values of a?

Outline

Problem 3

In this exercise, you will consider the problem of removing an echo from a recording of a speech signal. This project will use the audio capabilities of MATLAB to play recordings of both the original speech and the result of your processing. To begin this exercise you will need to load the speech file `lineup.mat`, which is contained in the Computer Explorations Toolbox. The Computer Explorations Toolbox can be obtained from The MathWorks at the address provided in the Preface. If this speech file is already somewhere in your MATLABPATH, then you can load the data into MATLAB by typing

```
>> load lineup.mat
```

You can check your MATLABPATH, which is a list of all the directories which are currently accessible by MATLAB, by typing `path`. The file `lineup.mat` must be in one of these directories.

Once you have loaded the data into MATLAB, the speech waveform will be stored in the variable y. Since the speech was recorded with a sampling rate of 8192 Hz, you can hear the speech by typing

```
>> sound(y,8192)
```

Problem 3

You should hear the phrase "line up" with an echo. The signal $y[n]$, represented by the vector y, is of the form

$$y[n] = x[n] + \alpha x[n-N], \tag{2.21}$$

where $x[n]$ is the uncorrupted speech signal, which has been delayed by N samples and added back in with its amplitude decreased by $\alpha < 1$. This is a reasonable model for an echo resulting from the signal reflecting off of an absorbing surface like a wall. If a microphone is placed in the center of a room, and a person is speaking at one end of the room, the recording will contain the speech which travels directly to the microphone, as well as an echo which traveled across the room, reflected off of the far wall, and then into the microphone. Since the echo must travel further, it will be delayed in time. Also, since the speech is partially absorbed by the wall, it will be decreased in amplitude. For simplicity ignore any further reflections or other sources of echo.

For the problems in this exercise, you will use the value of the echo time, $N = 1000$, and the echo amplitude, $\alpha = 0.5$.

Basic Problems

(a). In this exercise you will remove the echo by linear filtering. Since the echo can be represented by a linear system of the form Eq. (2.21), determine and plot the impulse

Problem 3

response of the echo system Eq. (2.21). Store this impulse response in the vector **he** for $0 \leq n \leq 1000$.

(b). Consider an echo removal system described by the difference equation

$$z[n] + \alpha z[n - N] = y[n], \tag{2.22}$$

where $y[n]$ is the input and $z[n]$ is the output which has the echo removed. Show that Eq. (2.22) is indeed an inverse of Eq. (2.21) by deriving the overall difference equation relating $z[n]$ to $x[n]$. Is $z[n] = x[n]$ a valid solution to the overall difference equation?

Intermediate Problems

(c). The echo removal system Eq. (2.22) will have an infinite-length impulse response. Assuming that $N = 1000$, and $\alpha = 0.5$, compute the impulse response using `filter` with an input that is an impulse given by `d=[1 zeros(1,4000)]`. Store this 4001 sample approximation to the impulse response in the vector `her`.

(d). Implement the echo removal system using `z=filter(1,a,y)`, where `a` is the appropriate coefficient vector derived from Eq. (2.22). Plot the output using `plot`. Also, listen to the output using `sound`. You should no longer hear the echo.

Outline

Assignments

Issued: March 21, 2024 Due: March 30, 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.