數位訊號處理導論

Introduction to Digital Signal Processing

(10620EE 366000)

Homework #2: Room acoustics and 2D spatial filtering

Due: 24:00, 04/15/2018

Part 1.

This homework is an extension of Example 2.8 “Echo generation and reverberation” in the textbook and give you a chance to practice the modeling and implementation of a digital room acoustic system.

The modeling in Example 2.8 only considers one echo path. Here please build the system model which takes two echo paths into consideration. D_1 in number of sampling intervals and a_1 are the single-echo round trip delay and reflection coefficient for the 1^{st} echo path, respectively, and D_2 and a_2 are the same parameters for the 2^{nd} echo path.

(a) Derive the non-recursive LCCDE of the digital room acoustic system, similar to textbook Eq. (2.101) and the corresponding impulse response.

(b) Derive the recursive LCCDE of the digital room acoustic system, find out the impulse response using the derived LCCDE and compare with the one derived in (a). Please use the MATLAB function **filter()** to compute and plot the impulse response and to verify your derivation , where $a_1 = 0.7$, $a_2 = 0.5$, $D_1 = 350$ and $D_2 = 560$.

(c) Please comment whether your modeled digital room acoustic system can be potentially implemented in real time and under what condition of a_1, a_2, D_1, and D_2 the modeled digital room acoustic system is stable.

(d) Given the input $x[n]$, which is a unit step sequence, i.e., $u[n]$, find out the zero-input and zero-state responses of the system

(e) Following (d), find out the steady state and transient step responses of the system. Please use the MATLAB function **filter()** to compute and plot the step response of the system and to verify your derivation, where $a_1 = 0.7$, $a_2 = 0.5$, $D_1 = 350$ and $D_2 = 560$. Please show your computational results in the form as shown in textbook Fig. 2.26, and tell how much time (or how long in time) the transient step response and zero input response last in theory and in your computational results, respectively.

(f) Approximate your modeled system as a FIR system and implement it using the MATLAB function **filter()** (or **conv()**)with $a_1 = 0.7$, $a_2 = 0.5$, and D_1 and D_2 correspond to 50 ms and 80 ms delays (note that $D = \tau Fs$ where τ is the delay and

Fs is the sampling rate). Given the sound file **Halleluyah.wav**, you can load the file, play the sound and save the output sound by using the following MATLAB codes:

```
[x, Fs] = wavread(' Halleluyah.wav');
sound(x, Fs);
% Your system implementation
% …
sound(y, Fs);
wavwrite(y,Fs, ' Halleluyah_FIRecho.wav');
```

where x is the input sound (a column vector), y is the output sound (a column vector, too), and Fs is the sampling rate in samples/sec (i.e., in Hz).

Please elaborate how you approximate the reverberator as a FIR filter. That is, elaborate how you determine the order of the FIR filter (i.e., the M in the general LCCDE in Eq. 2.94). Also plot x and y together as a function of time and check the changes in x done by your FIR filter. You may need to zoom in the plot to see the changes in details.

(g) Write **your own MATLAB codes** to implement your modeled IIR system using the recursive LCCDE derived in (b) with a_1 = 0.7, a_2 = 0.5, and D_1 and D_2 correspond to 50 ms and 80 ms delays (note that $D = \tau$Fs where τ is the delay and Fs is the sampling rate). To generate digital reverberation, we will use the same sound file in (f), and save your output sound as **Halleluyah_IIRecho.wav**. You may verify your results using MATLAB function **filter()**. Note that you have to provide the **initial condition** (i.e., y[n] and x[n], n < 0) used in your implementation. Also plot x and y together as a function of time, check the changes in x done by your IIR system, and comment the output differences between the two different implementations in (f) and (g). Note that you may need to zoom in the plot to see the changes in details.

(h) Change the D_1 and D_2 in (g) correspond to 100 ms and 160 ms delays, and to 500 ms and 800 ms delay, respectively. Please tell which implementation of the three sets {D_1, D_2} sounds natural, which means it sounds like in a real room, e.g., in a bathroom and elaborate the reason why such an implementation makes the output sound natural?

(i) Change the a_1 and a_2 in (g) to the values resulting in an unstable system, and grasp the feeling about an unstable digital room-acoustic system sounds like.

(j) Following (g), change the initial condition (e.g., y[n] = x[n] = 1, n < 0, or y[n] and x[n] are random numbers (try MATLAB function rand()), n < 0. Note that you need to scale y[n] or x[n] to the values close to your input) which you used in (g) and see how the initial condition affect the response. Combining the derivation in

(d) and (e) together with the current results, please tell that will the initial condition affect the transient response? Plot the outputs with different initial condition together as a function of time and tell the difference.

(k) Following (j), verify whether the system output y[n] is equal to the zero-input response plus zero-state response.

Part 2 (optional, you do not need to hand in this part)

Given the image lena.jpg, you can load the file and show the sound by using the following MATLAB codes:

x = double(imread('lena.jpg'));

imshow(x,[]);

Please try to answer the following problems in your textbook.

53. The second derivative operation $y = \frac{d^2x}{dt^2}$ is approximated by the difference equation

$$y[n] = x[n+1] - 2x[n] + x[n-1], \tag{2.124}$$

which is a noncausal LTI system and is used as an edge detector in image processing.

(a) Determine the impulse response of this edge detector.

(b) Load the Lena image in MATLAB and process it row-by-row using the above impulse response. Display the resulting image and comment on its appearance.

(c) Now process the Lena image column-by-column using the impulse response in (a). Display the resulting image and comment on its appearance.

54. The 2D second derivative or *Laplacian* can be approximated by the noncausal impulse response

$$h[m,n] = \boxed{\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}} \tag{2.125}$$

in which the $[0,0]$ term is in the center. It is also used as an edge detector in image processing.

(a) Load the Lena image in MATLAB and process it using the impulse response (2.125). Display the resulting image and comment on its appearance.

(b) An edge-enhanced image is obtained by subtracting the Laplacian of the image from the original. Determine the impulse response of this operation.

(c) Now process the Lena image using the impulse response in (b). Display the resulting image and comment on its appearance.

Please tell how you can obtain similar results in 54(b) using the system provided in 53? Also tell if there is any benefit if we use the system provided in 53 to generate similar results in 54(b). (Hint: You can use MATLAB functions conv() and conv2() to implement 1D and 2D convolution sum, respectively).

Notice:

1. Please hand in your solution files to the LMS elearning system, including your word file of the detailed solutions, the associated Matlab codes, and all the related materials. It would be nice that you can put your "KEY" code segment with comments side by side along with your answer in the word or pdf file if needed.
2. Name your solution files "EE3660_HW2_StudentID.doc" and "EE3660_HW2_StudentID.m", and archive them as a single zip file: EE3660_HW2_StudentID.zip.
3. The first line of your word/pdf or Matlab file should contain your name and some brief description, e.g., % EE 3660 王小明 u9612345 HW2 MM/DD/2018
4. If you need raw data file instead of MATLAB data, please let me know ASAP.

.