EECS 120 Signals & Systems — University of California, Berkeley: Spring 2004
Gastpar — January 30, 2004

Homework 2
Due: February 9, 2004, at 4pm

Reading OWN Chapter 2.

Practice Problems *(Suggestions.)* OWN 2.7, 2.8, 2.16(c)

Problem 1 *(Convolution via matlab.)* 15 Points

Matlab has a built-in function `conv` that can evaluate convolution operations for you. To get started, define a time vector, for example:

```
n = [ -100:1000 ];
```

Of course, the length of the time vector has to be chosen to accommodate the signal. The next step is to initialize a signal vector of the same length:

```
x = zeros(1, length(n));
```

Next, the signal has to get its values. If you simply want to set $x[n=0]=1$, this can be done as follows:

```
x(n==0) = 1;
```

In other cases, you are given a signal such as $x[n]=\cos(\frac{\pi}{64}n)$. In your matlab program, you can do this as follows:

```
x = cos ( pi*n/64 );
```

To look at your signal, you can now use

```
plot(n, x);
```

which automatically also gives you the correct time axis.

Remark: Clearly, the signal $x[n]=\cos(\frac{\pi}{64}n)$ is infinitely long, so in matlab, you can only approximate it. For periodic signals, a rule of thumb is to at least include ten or twenty periods of the signal, but of course, the more the better (but also: the slower the program execution).

Hence, you can define two signals, `x1` and `x2`, in this fashion, and find their convolution using the matlab command `conv`. Use this approach to redo the examples that we did in class, i.e., evaluate the convolution of

$$x[n] = \delta[n+1]+\delta[n]-\delta[n-1], \quad (1)$$
$$h[n] = \left(\frac{1}{2}\right)^n u[n]. \quad (2)$$

and of

$$x[n] = u[n-1], \quad (3)$$
$$h[n] = \left(-\frac{1}{2}\right)^n u[n]. \quad (4)$$

For each case, furnish a .m-file containing the way you construct the signals, and how you call the function `conv`, as well as three plots showing, showing $x[n]$, $h[n]$, and $(x*h)[n]$, respectively. *Hint:* To do a nice and aligned triple plot, you can use the matlab function `subplot`.

Problem 2 *(Your new job at BT&T.)* 25 Points

You start your new job at BT&T (Berkeley Telephone & Telegraph, Inc.). Customers have complained about high-frequency disturbances in their phone conversations, caused by the squeaking BART trains (while using their cell phones on BART). Your team must come up with a solution for the problem by applying a suitable system to the signal $x[n]$ coming into the base station (and thus, potentially corrupted by BART squeaking). Your system should filter $x[n]$ into a signal $y[n]$ with *less* BART squeaking, but still containing (of course!) the actual phone conversation.

A senior engineer in your team suggests to use the following system, shown in Figure 1. You decide to verify whether this system does the trick.

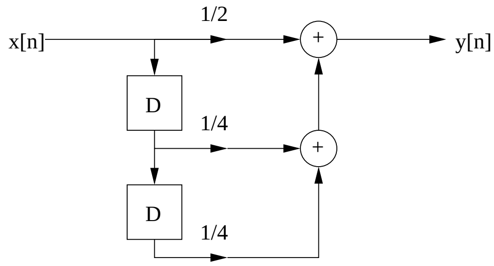

Figure 1: The suggested squeaking-suppression system.

(a) (5 Points) Show that the system is linear and time-invariant, and find its impulse response $h[n]$.

(b) (5 Points) Is the system causal? Memoryless? Stable? Why is stability an interesting issue for BT&T?

(c) (10 Points) Now comes the test: Suppose that the actual conversation signal, without the disturbing noise, is

$$x_c[n] \quad = \quad \cos(\frac{\pi}{32}n), \tag{5}$$

and the high-frequency squeaking is

$$x_s[n] \quad = \quad \cos(\frac{\pi}{3}n). \tag{6}$$

So, the noise-corrupted input signal to your system is $x[n] = x_c[n] + x_s[n]$. Use matlab to give plots of *both* the input and output signals for the following three cases: *(i),* the input to the system is $x_c[n]$, *(ii),* the input to the system is $x_s[n]$, and *(iii),* the input to the system is $x[n] = x_c[n] + x_s[n]$. Does the system attenuate the squeaking? What's your opinion on your colleague's suggestion?

(d)(5 Points) Derive the formula for the output signal $y[n]$ when the input is $x[n] = x_c[n] + x_s[n]$. *Hint:* $\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$.

(e) (Extra credit 5 Points) Can you improve your colleague's system?

Problem 3 *(Review of Linear Algebra.)* 20 Points

In this class, we will use boldface symbols to denote vectors $\mathbf{x} \in C^N$. This notation means that $\mathbf{x}$ is a vector of length N with complex entries. Matrices will be denoted by upper case letters $H \in C^{M\times N}$, which means that H is a matrix of dimensions $M \times N$ (i.e., with M rows and N columns).

(a) (5 Points) Let $N = M = 2$, and consider

$$H = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{x} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{7}$$

Evaluate $H\mathbf{x}$.

(b) (10 Points) For the matrix H of part *(a)*, find the eigenvalues λ_1, λ_2 and the corresponding eigenvectors $\mathbf{u}_1, \mathbf{u}_2$, i.e., that satisfy

$$H\mathbf{u}_k \quad = \quad \lambda_k \mathbf{u}_k, \text{ for } k = 1, 2. \tag{8}$$

Remark: You may use matlab to find λ_1, λ_2 and $\mathbf{u}_1, \mathbf{u}_2$, but if you do so, then you must verify in paper and pencil that your numbers are indeed the eigenvalues and eigenvectors. No credit is given for a purely numerical answer.

(c) (5 Points) Using your results from part *(b)*, write the matrices

$$U = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 \end{pmatrix} \quad \text{and} \quad \Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}. \tag{9}$$

Show that

$$H \quad = \quad U \Lambda U^H, \tag{10}$$

where ^{H} is called the "Hermitian transpose", defined as the regular transpose with the additional property the every element is complex conjugated.

(d) Extra Credit (5 Points) Repeat parts *(b)* and *(c)* for the following matrix:

$$A = \begin{pmatrix} 1 & 2 \\ -2 & 1 \end{pmatrix}. \tag{11}$$

Problem 4 *(Convolution.)* 10 Points

(a) OWN Problem 2.23 (all parts, in particular the sketches).

(b) OWN Problem 2.24 (a).

Problem 5 *(Impulse response and system properties.)* 15 Points

OWN Problem 2.48 (a), (c), (e).

Problem 6 *(Associativity of convolution.)* 15 Points

(a) Use the definition of convolution to establish the property of associativity, i.e.,

$$(x * (y * z))[n] \quad = \quad ((x * y) * z)[n], \tag{12}$$

by showing that both sides of this equation can be written as $\sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} x[k]y[m]z[n-k-m]$.

(b) Consider the following three signals, defined for $-\infty < n < \infty$:

$$d[n] \quad = \quad \begin{cases} -1, & n = 0 \\ 1, & n = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

$$e[n] \quad = \quad -1, \forall n \tag{14}$$

$$u[n] \quad = \quad \begin{cases} 1, & n \geq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Compute

$$(u * (d * e))[n] \quad \text{and} \quad ((u * d) * e)[n]. \tag{16}$$

What do you observe? (Compare with Part *(a)*.)

(c) Which step of your proof in Part *(a)* does not hold for the signals considered in Part *(b)*?