The University of Texas at Austin
Dept. of Electrical and Computer Engineering
Midterm #1 ***Solutions 2.0***

Date: October 16, 2024 Course: EE 445S Evans

Name: ____
Last, First

- **Exam duration**. The exam is scheduled to last 75 minutes.
- **Materials allowed**. You may use books, notes, your laptop/tablet, and a calculator.
- **Disable all networks**. Please disable all network connections on all computer systems. You may not access the Internet or other networks during the exam.
- **No AI tools allowed**. As mentioned on the course syllabus, you may not use GPT or other AI tools during the exam.
- **Electronics.** Power down phones. No headphones. Mute your computer systems.
- **Fully justify your answers**. When justifying your answers, reference your source and page number as well as quote the particular content in the source for your justification. You could reference homework solutions, test solutions, etc.
- **Matlab**. No question on the test requires you to write or interpret Matlab code. If you base an answer on Matlab code, then please provide the code as part of the justification.
- **Put all work on the test**. All work should be performed on the quiz itself. If more space is needed, then use the backs of the pages.
- **Academic integrity.** By submitting this exam, you affirm that you have not received help directly or indirectly on this test from another human except the proctor for the test, and that you did not provide help, directly or indirectly, to another student taking this exam.

Problem	*Point Value*	*Your score*	*Topic*
1	25		FIR Filter Analysis
2	27		Predistortion
3	24		Discrete-time Feedback System
4	24		Mystery Systems
Total	100		

Lectures 3 5 & 6	Lab #3	HW 0.1, 1.1, 1.3, 2.1, 2.2, 2.3, 3.2

JSK Ch. 7	Midterm 1 Problem 1: Sp10, F16, F18, F20, F23

Problem 1.1 *FIR Filter Analysis*. 25 points.

Consider a causal linear time-invariant (LTI) discrete-time finite impulse response (FIR) filter with input $x[n]$ and output $y[n]$ observed for $n \geq 0$. The transfer function in the z-domain is

$$H(z) = a + b\, z^{-100} \quad \text{for } z \neq 0$$

where a and b are real-valued, non-zero constants.

(a) Give the equation for output $y[n]$ in terms of the input $x[n]$ in the discrete-time domain for $n \geq 0$. *6 points*.

$H(z) = \frac{Y(z)}{X(z)} = a + b\, z^{-100}$ which means $Y(z) = a\, X(z) + \; b\, z^{-100}\, X(z)$

So, $y[n] = a\, x[n] + b\, x[n-100]$ for $n \geq 0$.

(b) What are the initial condition(s) and their value(s)? Why? *3 points*.

We can see the initial conditions by starting to compute the first few values of $y[n]$.

$$y[0] = \; a\, x[0] + b\, x[-100]$$

$$y[1] = \; a\, x[1] + b\, x[-99]$$

The initial conditions are

$$x[-1] = x[-2] = \cdots = x[-100] = 0$$

This will satisfy the necessary (but not sufficient) conditions for linearity and time-invariance properties to hold.

(c) In managing the memory for storing the previous input values, would you advocate for a linear buffer or a circular buffer? Why? *3 points*

To update the storage of the current and previous 100 input samples each time a new input sample arrives, a linear buffer would require 100 reads and 100 writes

$$x[-100] = x[-99];\; x[-99] = x[-98]; \ldots ; x[-1] = x[0]$$

whereas a circular buffer would require one read and two writes. The circular buffer update would need to read the pointer to the oldest value, write the current input value into the memory location for the oldest sample, and then update the pointer to the oldest value. I would use a circular buffer due to its much lower complexity.

(d) Derive a formula for the discrete-time frequency response of the filter. *3 points*.

Since the transfer function $H(z)$ includes the unit circle in the region of convergence, we can substitute $z = e^{j\,\omega}$ to convert the transfer function into a frequency response:

$$H(e^{j\,\omega}) = \; a + b\, e^{-j\,100\,\omega}$$

(e) Give all possible conditions on the constants a and b so that the FIR filter has constant group delay. Compute the constant group delay. *10 points*.

$$H(e^{j\,\omega}) = \; a + b\, e^{-j\,100\,\omega} = e^{-j\,50\,\omega}\left(a\, e^{j\,50\,\omega} + b\, e^{-j\,50\,\omega}\right)$$

Constant group delay means the phase response has constant slope: $GD(\omega) = -\frac{d}{d\omega} \angle H(e^{j\,\omega})$

Only possible if there's symmetry in impulse response about its midpoint (lecture slide 5-17):

Even symmetry: $a = b$. $H(e^{j\,\omega}) = e^{-j\,50\,\omega}(2\, a \cos(50\,\omega))$ and $GD(\omega) = 50$ samples

Odd symmetry: $a = -b$. $H(e^{j\,\omega}) = e^{-j\,50\,\omega}(2\, j\, a \sin(50\,\omega))$ and $GD(\omega) = 50$ samples

Lectures 5 & 6	Lab #3	HW 2.1, 2.3, 3.1, 3.3	JSK Ch. 7	Handout O

Lecture Slide 6-15	Midterm 1: Prob 1.2 Sp24, 1.3 Sp23 & 1.2 F19

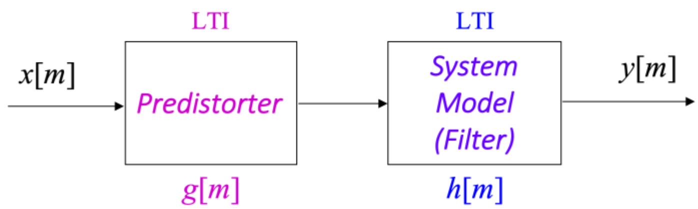
Note: If the predistorter were placed after the system, then it would be called an equalizer

Problem 1.2 *Predistortion.* 27 points.

Predistortion is a technique used to compensate the distortion in another system.

An example is applying predistortion to an audio signal before being played by an audio system.

The block diagram illustrates the use of predistortion when the predistorter is a linear time invariant (LTI) system and the distortion is modeled as LTI.

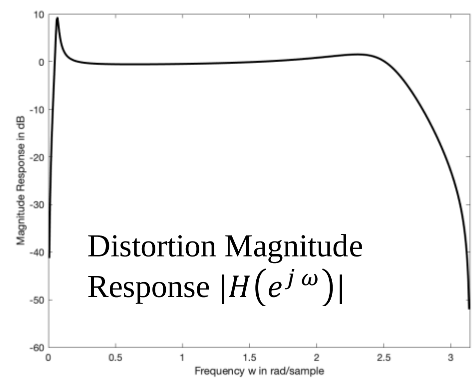


- sampling rate f_s is 44100 Hz.
- $g[m]$ is the impulse response of the discrete-time LTI predistorter
- $h[m]$ is the impulse response of a discrete-time LTI model of the distortion in a playback system.

A predistorter is bounded-input bounded-output stable.

Distortion. Its poles and zeros are given below and its magnitude response $|H(e^{j\,\omega})|$ is on given on the right.

Poles	Zeros
$p_0 = 0.75\, e^{j\,0.8\,\pi}$	$z_0 = 1$
$p_1 = 0.75\, e^{-j\,0.8\,\pi}$	$z_1 = -1$
$p_2 = 0.99\, e^{j\,0.02\,\pi}$	$z_2 = 1$
$p_3 = 0.99\, e^{-j\,0.02\,\pi}$	$z_3 = -1.25$



An all-pass cascade would mean $\left|G_{freq}(\omega)\, H_{freq}(\omega)\right| = g$ where g is a positive constant. Per lecture slide 6-15, a pole-zero pair in an all-pass configuration has two cases: (1) if the zero is inside the unit circle, then the pole and zero have the same value and cancel or (2) if the zero is outside the unit circle, the pole would have the same angle and reciprocal magnitude. If the zero is on the unit circle, place pole inside the unit circle at the zero angle for a notch configuration.

(a) What continuous-time frequency corresponds to pole location p_2? *3 points.*

From sampling a continuous-time sinusoidal signal $x(t) = \cos(2\,\pi\, f_0\, t)$ at sampling rate f_s ,

$x[n] = x(n\, T_s) = \cos\left(2\,\pi\, \frac{f_0}{f_s} n\right)$ where $\hat{\omega}_0 = 2\,\pi\, \frac{f_0}{f_s}$.

The discrete-time frequency $\hat{\omega}_2$ association with pole location p_2 is $0.02\,\pi$, so

$\hat{\omega}_2 = 2\,\pi\, \frac{f_2}{f_s}$ which means $f_2 = \frac{\hat{\omega}_2}{2\,\pi} f_s = \frac{0.02\,\pi}{2\,\pi}(44100 \text{ Hz}) = 441$ Hz. Close to note 'A4'.

(b) Give the zeros of the predistorter. Explain how you determined each one. *12 points.*

***Answer #1*: The zeros of the predistorter would be the values of the poles of the distortion. This will lead to pole-zero cancellations. Zeros do not cause BIBO stability.**

$z_0 = p_0 = 0.75\, e^{j\,0.8\,\pi}$ and $z_1 = p_1 = 0.75\, e^{-j\,0.8\,\pi}$

$z_2 = p_2 = 0.99\, e^{j\,0.02\,\pi}$ and $z_3 = p_3 = 0.99\, e^{-j\,0.02\,\pi}$

***Answer #2*: Put the zeros in an all-pass configuration where each zero is at the same angle as the pole but the magnitude is inverted. Lecture Slide 6-15 and Handout O on all-pass filters.**

$z_0 = \frac{1}{0.75}\, e^{j\,0.8\,\pi} = \frac{4}{3}\, e^{j\,0.8\,\pi}$ **and** $z_1 = \frac{1}{0.75}\, e^{-j\,0.8\,\pi} = \frac{4}{3}\, e^{-j\,0.8\,\pi}$

$z_2 = \frac{1}{0.99}\, e^{j\,0.02\,\pi}$ **and** $z_3 = \frac{1}{0.99}\, e^{-j\,0.02\,\pi}$

(c) Give the poles of the predistorter. Explain how you determined each one. *12 points.*

Notch configuration: $p_0 = 0.9$ **and** $p_1 = -0.9$ **and** $p_2 = 0.9$.

Reciprocal magnitudes for an all-pass configuration: $p_3 = -0.8$.

```
%% Matlab code to generate the magnitude response in Problem 1.2
clear all;

fs = 44100;

z0 = 1;
z1 = -1;
numer1 = [1 -(z0+z1) z0*z1];
poleAngle = 0.8*pi;
r0 = 0.75;
p0 = r0 * exp(j*poleAngle);
p1 = r0 * exp(-j*poleAngle);
denom1 = [1 -(p0+p1) p0*p1];

z2 = 1;
z3 = -1.25;
numer2 = [1 -(z2+z3) z2*z3];
r = 0.99;
fpole = 441;
poleAngle = 2*pi*fpole/fs;
p2 = r * exp(j*poleAngle);
p3 = r * exp(-j*poleAngle);
denom2 = [1 -(p2+p3) p2*p3];

%%% Normalize the DC response to 1 in linear units by
%%% setting H(z) evaluated at z = exp(j pi/2) to be 1
numer = conv(numer1, numer2);  % polynomial multiplication
denom = conv(denom1, denom2);  % polynomial multiplication
zval = exp(j*pi/2);
zvec = zval .^ [0 -1 -2 -3 -4];
C = (denom * zvec') / (numer * zvec');

figure;
[h,w] = freqz(C*numer, denom);
p = plot(w,20*log10(abs(h)),'k');
p(1).LineWidth = 2;
xlim( [0 pi] );
xlabel('Frequency w in rad/sample');
ylabel('Magnitude Response in dB');
```

Lectures 5 & 6	HW 0.4, 1.1, 2.1, 2.2, 3.1, 3.3
Lab #3	Midterm 1.1 Sp21

Problem 1.3. *Discrete-Time Feedback System*. 24 points.

Consider a discrete-time linear time-invariant (LTI) system with input signal $x[n]$ and output signal $y[n]$ that is governed by the following second-order difference equation for $n \geq 0$:

$$y[n] = 1.6\, y[n-1] - K\, y[n-2] + x[n]$$

where K is a real-valued constant.

(a) What are the initial conditions of the system and what values should they have? *6 points*.

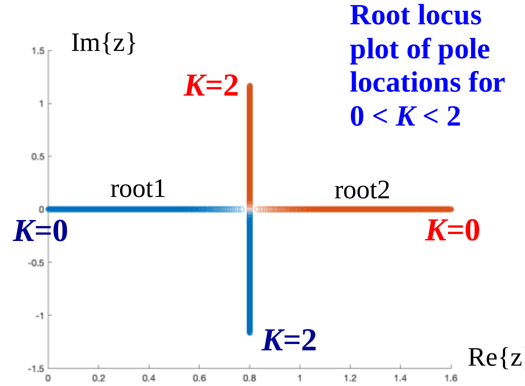
A necessary condition for a system to have LTI properties is that it must be "at rest". That is, the initial conditions of the system must be zero.
We can find the initial conditions of the system by computing the first output values:

$$\boldsymbol{y[0] = 1.6\, y[-1] - K\, y[-2] + x[0]}$$

$x[0]$ and $y[0]$ are initial values of input signal $x[n]$ and output signal $y[n]$, respectively. Neither is an initial condition of the system. Hence $y[-1] = 0$ and $y[-2] = 0$.

(b) Derive the transfer function $H(z)$ for the system, which will depend on K. *6 points*.

Root locus plot of pole locations for $0 < K < 2$



Take the z-transform of both sides of the equation:

$$\boldsymbol{Y(z) = 1.6\, z^{-1} Y(z) - K\, z^{-2}\, Y(z) + X(z)}$$

Next, collect $Y(z)$ terms on left-hand side of the equation:

$$\boldsymbol{Y(z) - 1.6\, z^{-1}\, Y(z) + K\, z^{-2}\, Y(z) = X(z)}$$

$$\boldsymbol{(1 - 1.6\, z^{-1} + K\, z^{-2})\, Y(z) = X(z)}$$

$$\boldsymbol{H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - 1.6\, z^{-1} + K\, z^{-2}} = \frac{z^2}{z^2 - 1.6\, z + K}}$$

$H(z)$ has two zeros at the origin ($z = 0$) and poles at

$$\boldsymbol{\frac{-1.6 \pm \sqrt{(-1.6)^2 - 4K}}{2} = \frac{-1.6 \pm \sqrt{(-2 \times 0.8)^2 - 4K}}{2} = 0.8 \pm \sqrt{0.64 - K}}$$

Region of convergence is complex z plane outside a circle of the larger pole radius: $|z| > \max\{\, |p_0|, |p_1|\, \}$

```
K = 0 : 0.001 : 2;
root1 = 0.8 - sqrt(0.64 - K);
root2 = 0.8 + sqrt(0.64 - K);
hold on;
scatter(real(root1), imag(root1));
scatter(real(root2), imag(root2));
hold off;
```

(c) Give the range of values for K for which the system is bounded-input bounded-output (BIBO) stable. *6 points*.

Both poles must be inside unit circle for BIBO stability. For $K \leq 0.64$, poles are real-valued:
$-1 < 0.8 - \sqrt{0.64 - K}$ and $0.8 + \sqrt{0.64 - K} < 1$
The left inequality gives $K > -2.6$ and the right one gives $K > 0.6$. Hence, $K > 0.6$.
For $K > 0.64$, the poles are complex-valued: $0.8 \pm j\sqrt{K - 0.64}$.
$\left|0.8 + j\sqrt{K - 0.64}\right| < 1$ which means $\sqrt{(0.8)^2 + (K - 0.64)} < 1$
By squaring both sides, $(0.8)^2 + (K - 0.64) < 1$ which means $K < 1$. So, $0.64 < K < 1$.

The full range of K for BIBO stability is $0.6 < K < 1$.

(d) Describe the possible frequency selectivity (lowpass, highpass, bandpass, bandstop, allpass or notch) that the system could exhibit for different values of K for which the system is BIBO stable. *6 points*.

For $0.6 < K < 0.64$, poles are real-valued between 0.8 and 1.0, not inclusive. Lowpass. At $K = 0.64$, there is double pole at $z = 0.8$, which is also means a lowpass response. As K increases from 0.64 to 1, the pole separation increases and the response becomes bandpass.

Midterm 1: 1.4 Sp11, 1.3 Sp15, 1.5 Sp19, 1.4 F19, 1.4 Sp20, 1.4 F21, 1.4 Sp20, 1.4 F F23

Lecture 4	Handout Common Signals in Matlab
HW 1.2 1.3 & 2.2	In-Lecture #1 Assignment

Problem 1.4. *Mystery Systems*. 24 points.

You're trying to identify unknown discrete-time systems.

You input a discrete-time chirp signal $x[n]$ and look at the output to figure out what the system is.

The discrete-time chirp is formed by sampling a chirp signal that sweeps 0 to 8000 Hz over 0 to 5s

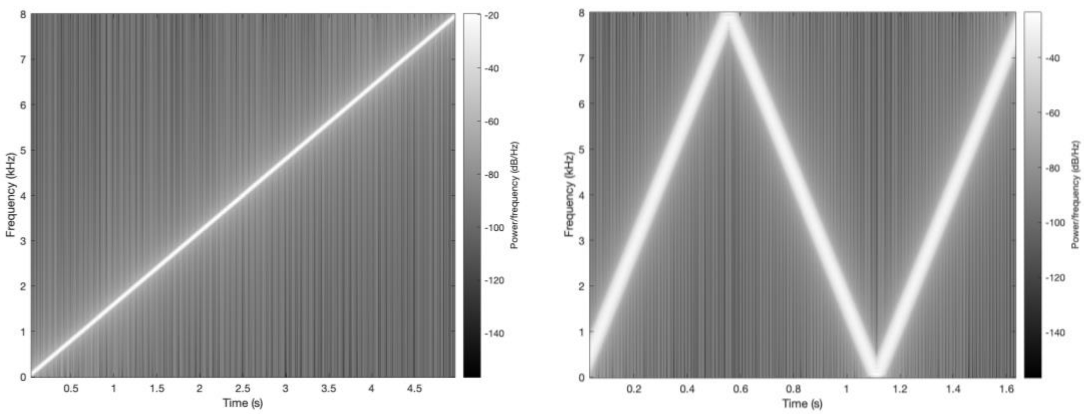
$$x(t) = \cos(2\pi f_1 t + 2\pi \mu t^2)$$

where $f_1 = 0$ Hz, $f_2 = 8000$ Hz, and $\mu = \frac{f_2 - f_1}{2\, t_{\max}} = \frac{8000 \text{ Hz}}{10\, s} = 800 \text{ Hz}^2$. Sampling rate f_s is 16000 Hz.

In each part below, identify the unknown system as one of the following **with justification**:

1. filter – give selectivity (lowpass, highpass, bandpass, bandstop) and passband/stopband frequencies
2. upsampler – give upsampling factor
3. downsampler – give downsampling factor
4. pointwise nonlinearity – give the integer exponent k to produce the output $y[n] = x^k[n]$

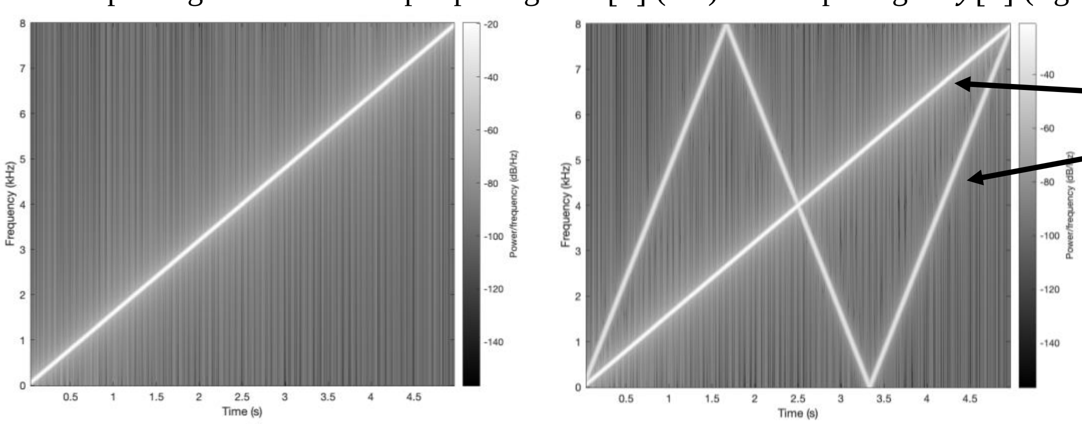
(a) Given spectrograms of the chirp input signal $x[n]$ (left) and output signal $y[n]$ (right). *12 points*.



Midterm Problems
1.4 Sp 22
1.2 F 18
1.2(d) Sp 18

When compared to the input spectrogram, the output spectrogram has the same range of frequencies along the vertical axis but one-third the duration in time along the horizontal axis. The principal frequency in the output spectrogram is a chirp pattern that is wider and has aliasing. Downsampling by 3 per homework problem 2.2(d).

(b) Given spectrograms of the chirp input signal $x[n]$ (left) and output signal $y[n]$ (right). *12 points*.



Same as input frequency $\boldsymbol{f_0}$

$\boldsymbol{3f_0}$

Lecture 4

Cubic nonlinearity gives frequencies $+f_0$ and $+3f_0$ as well as their negative counterparts:

$$\cos^3(2\pi f_0 t) = \cos^2(2\pi f_0 t)\cos(2\pi f_0 t) = \left(\frac{1}{2} + \frac{1}{2}\cos(2\pi(2f_0)t)\right)\cos(2\pi f_0 t)$$

$$\cos^3(2\pi f_0 t) = \frac{1}{2}\cos(2\pi f_0 t) + \frac{1}{2}\cos(2\pi(2f_0)t)\cos(2\pi f_0 t)$$

The product of two cosines gives a sum and difference of the frequencies, scaled by (1/2):

$$\cos^3(2\pi f_0 t) = \frac{3}{4}\cos(2\pi f_0 t) + \frac{1}{4}\cos(2\pi(3f_0)t)$$

The frequency at $+3f_0$ will alias once $f_0 \geq \frac{1}{6} f_s$. Folding for $\frac{1}{6} f_s \leq f_0 < \frac{1}{3} f_s$.

```
%% Matlab code to generate the spectrograms for Problem 1.4
fs = 16000;
Ts = 1 / fs;
tmax = 5;
t = 0 : Ts : tmax;

%% Create chirp signal
f1 = 0;
f2 = fs/2;
mu = (f2 - f1) / (2*tmax);
x = cos(2*pi*f1*t + 2*pi*mu*(t.^2));

%% (a) Downsample by 3
downsamplingFactor = 3;
xLength = length(x);
y = x(1:downsamplingFactor:xLength);

%%% Spectrogram parameters
blockSize = 1024;
overlap = 1023;

%%% Plot spectrogram of input signal
figure;
spectrogram(x, blockSize, overlap, blockSize, fs, 'yaxis');

%%% Plot spectrogram of output signal
figure;
spectrogram(y, blockSize, overlap, blockSize, fs, 'yaxis');

%% (b) Cubic nonlinearity
y = x .^ 3;

%%% Spectrogram parameters
blockSize = 1024;
overlap = 1023;

%%% Plot spectrogram of input signal
figure;
spectrogram(x, blockSize, overlap, blockSize, fs, 'yaxis');

%%% Plot spectrogram of output signal
figure;
spectrogram(y, blockSize, overlap, blockSize, fs, 'yaxis');
```