The University of Texas at Austin
Dept. of Electrical and Computer Engineering
Midterm #1 ***Solutions 1.0***

Date: October 22, 2025 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 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	24		IIR Filter Analysis
2	27		Predistortion
3	27		Sinusoidal Amplitude Modulation
4	22		Design Tradeoffs
Total	100		

Lectures 3 & 6 | Lab #3 | HW 0.4, 1.1, 2.1, 3.1, 3.3 | JSK Ch. 7

Midterm 1 Prob 1: Sp 25, Sp23, Sp18, Sp17, F16, F12, Sp07

Problem 1.1 *IIR Filter Analysis*. 24 points.

Consider a causal linear time-invariant (LTI) discrete-time infinite impulse response (IIR) filter with input $x[n]$ and output $y[n]$ observed for $n \geq 0$ is described by

Weighted average of previous output and current input value

$$y[n] = \alpha\, y[n-1] + (1-\alpha)\, x[n]$$

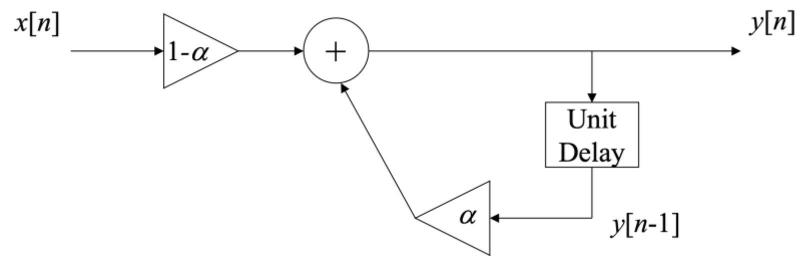
where α is a real-valued constant $0 < \alpha < 1$.

Very similar to Spring 2025 Problem 1.1 for $K = 1$

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

Consider $n = 0$: $y[0] = x[0] + \alpha\, y[-1]$ **where $y[-1]$ is an initial condition**
Consider $n = 1$: $y[1] = x[1] + \alpha\, y[0]$ **no initial conditions present**
The initial condition must equal zero as a necessary condition for LTI to hold: $y[-1] = 0$

(b) Draw a block diagram. Be sure to use arrows to indicate the order of operations. *3 points*.



(c) Compute the transfer function in the z-domain including the region of convergence. 6 *points*.

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

$$Y(z) = \alpha\, z^{-1}\, Y(z) + (1-\alpha)\, X(z)$$
$$Y(z) - \alpha\, z^{-1}\, Y(z) = (1-\alpha)\, X(z)$$
$$Y(z)(1 - \alpha\, z^{-1}) = (1-\alpha)\, X(z)$$
$$H(z) = \frac{Y(z)}{X(z)} = \frac{1-\alpha}{1 - \alpha\, z^{-1}}$$

The poles are the roots of $1 - \alpha\, z^{-1} = 0$ which means $\alpha\, z^{-1} = 1$ or $z = \alpha$

The pole is at $z = \alpha$.

For a causal system, the region of convergence would be $|z| > \alpha$

(d) Is the filter bounded-input bounded-output (BIBO) stable for all allowable values of α which are $0 < \alpha < 1$? Why or why not? *6 points.*

Answer #1: **For a causal system, the poles must be inside the unit circle for BIBO stability.**

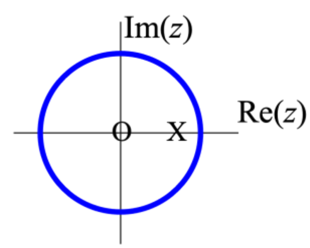
Since $0 < \alpha < 1$, the pole $z = \alpha$ will always be inside the unit circle.

Answer #2: **For a BIBO stable system, the region of convergence for the transfer function must include the unit circle. This is the case because the ROC is $|z| > \alpha$ and $0 < \alpha < 1$.**

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

Since the LTI system is BIBO stable, we can substitute $z = e^{j\,\omega}$ into $H(z)$ to find the discrete-time frequency response of the filter:

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = \frac{1}{1 - \alpha\, e^{-j\,\omega}} \quad \text{for } 0 < \alpha < 1$$



(f) For what values for α over $0 < \alpha < 1$ will the filter be lowpass? *3 points*.

The angle of the pole (0 rad/sample) indicates the center of the passband.
The closer the pole is to the unit circle, the more selective the response.
As α tends to 0 but not equal to 0, the filter tends to an allpass filter $y[n] = x[n]$.
As α tends to 1 but not equal to 1, the filter tends to a more selective lowpass filter.

JSK Sec. 2.5, 2.12, 3.3, 4.2, 7.1 & 7.2

Lecture Slide 6-15 | Lectures 5 & 6 | Lab #3 | HW 2.1, 2.3, 3.1, 3.3

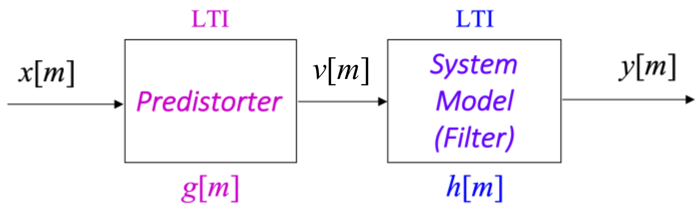
Midterm 1: Prob 1.2 F24, 1.2 Sp24, 1.3 Sp23, 1.2 F19

Problem 1.2 *Predistortion.* 27 points.

Predistortion compensates the distortion that occurs in later processing of the signal.

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

A predistorter is bounded-input bounded-output (BIBO) stable.

Distortion. The input-output relationship for the LTI distortion (reverberation) is

$$y[m] = v[m] - \alpha\, v[m-K]$$

where α is any non-zero real number and K is a positive finite integer.

(a) Give the zeros of the LTI distortion $h[m]$ as expressions involving α and K. *9 points.*

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

$$\boldsymbol{Y(z) = V(z) + \alpha\, z^{-K}\, V(z)}$$

$$\boldsymbol{Y(z) = (1 - \alpha\, z^{-K}) V(z)}$$

$$\boldsymbol{H(z) = \frac{Y(z)}{V(z)} = 1 - \alpha\, z^{-K}}$$

$\boldsymbol{H(z)}$ **is a finite impulse response (FIR) filter.**

The zeros are the $\boldsymbol{K}$ roots of $\boldsymbol{1 - \alpha\, z^{-K} = 0}$ which means $\boldsymbol{\alpha\, z^{-K} = 1}$ or $\boldsymbol{z^K = \alpha}$

Because $\boldsymbol{z}$ is complex-valued, we get complex-valued roots:

$$\boldsymbol{\sqrt[K]{\alpha}\, e^{j\frac{2\pi}{K}n} \quad \text{if } \alpha > 0}$$

$$\boldsymbol{\sqrt[K]{-\alpha}\, e^{-j\frac{\pi}{K}}\, e^{j\frac{2\pi}{K}n} \quad \text{if } \alpha < 0}$$

for $\boldsymbol{n = 0, 1, \ldots, K-1}$.

There are $\boldsymbol{K}$ poles at the origin for $\boldsymbol{H(z)}$.

(b) Give the poles of the predistorter $g[m]$ as expressions involving α and K to guarantee bounded-input bounded-output stability. *18 points.*

We'd like to design the LTI predistorter $\boldsymbol{g[m]}$ so that the cascade of LTI predistorter $\boldsymbol{g[m]}$ and the LTI distortion $\boldsymbol{h[m]}$ is allpass.

***Pole-zero cancellation*: When $\boldsymbol{|\alpha| < 1}$, $\boldsymbol{G(z) = \frac{1}{H(z)}}$. The zeros of $\boldsymbol{H(z)}$ become the poles of $\boldsymbol{G(z)}$. $\boldsymbol{G(z)}$ would be BIBO stable because all poles would be inside the unit circle.**

***Notch configuration:* When $\boldsymbol{|\alpha| = 1}$, the zeros of $\boldsymbol{H(z)}$ are on the unit circle. The poles of $\boldsymbol{G(z)}$ would be the zeros of $\boldsymbol{H(z)}$ scaled by 0.9 so that the poles of $\boldsymbol{G(z)}$ would be inside the unit circle for BIBO stability.**

***Allpass configuration:* When $\boldsymbol{|\alpha| > 1}$, the zeros of $\boldsymbol{H(z)}$ are outside the unit circle. The poles of $\boldsymbol{G(z)}$ would be the zeros of $\boldsymbol{H(z)}$ reflected inside the unit circle by keeping the angle of each zero the same and inverting its magnitude.**

There are $\boldsymbol{K}$ zeros at the origin for $\boldsymbol{G(z)}$. $\boldsymbol{G(z)}$ is also known as an all-pole IIR filter.

$G(z)$ when $|\alpha| < 1$ is an IIR comb filter. You'll implement it in lab #7.

See https://en.wikipedia.org/wiki/Comb_filter

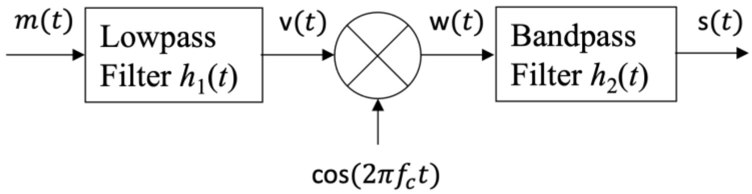
JSK Ch. 2, 3, 5 & 7 | Lectures 4 & 6 | Handout E Convolving Rectangular Pulses

Midterm Problems 1.3 Sp25, 1.4 F20, 1.2 Sp20, 1.2 Sp19, 1.3 F18, 1.2 Sp18, 1.2 F17

Labs 2 & 3 | HW 0.1, 0.2, 0.3, 1.3, 3.1

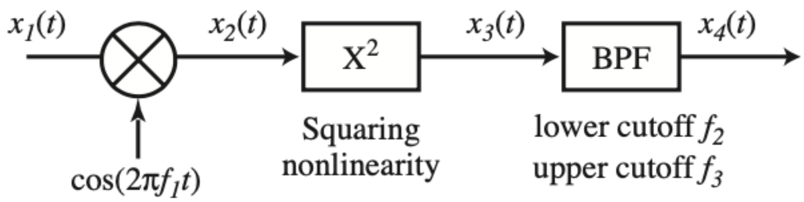
Problem 1.3. *Sinusoidal Amplitude Modulation*. 27 points.

Sinusoidal amplitude modulation can be used to convey a baseband message signal $m(t)$ wirelessly as a bandpass RF signal $s(t)$ that can propagate further. Here's a block diagram representation:

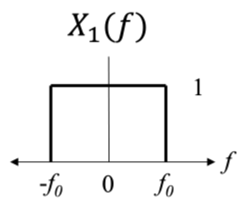


Handout H Amplitude Modulation

An alternate modulation approach uses a squaring block and is shown below:



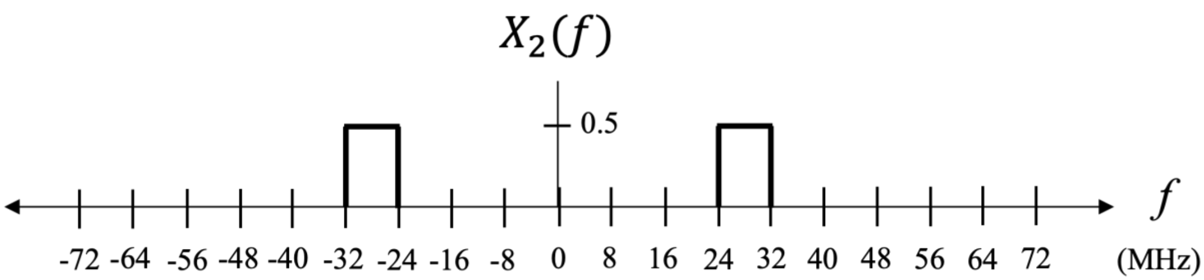
Using the baseband signal $x_1(t)$ whose spectrum is plotted on the right and using the parameters for f_0 and f_1 shown on the right,



$f_0 = 4$ MHz

$f_1 = 28$ MHz

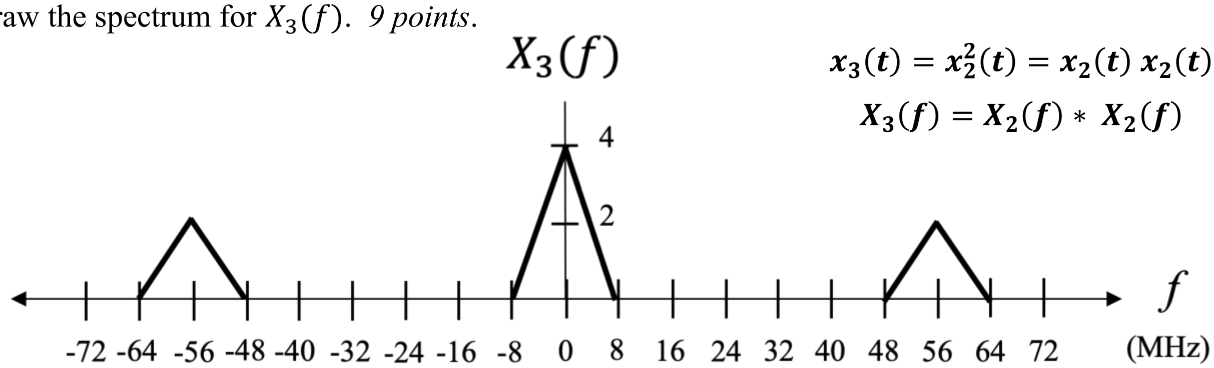
(a) Draw the spectrum for $X_2(f)$. *6 points*.



(b) Draw the spectrum for $X_3(f)$. *9 points*.

$$\boldsymbol{x_3(t) = x_2^2(t) = x_2(t)\, x_2(t)}$$

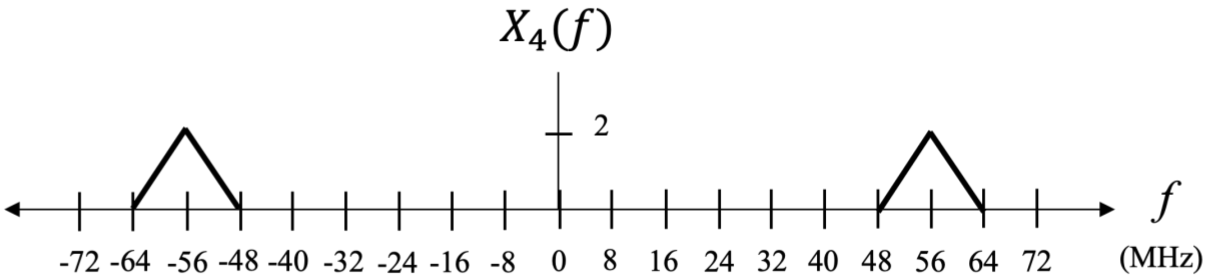
$$\boldsymbol{X_3(f) = X_2(f) * X_2(f)}$$



(c) Give values of f_2 and f_3 to keep only bandpass part of the spectrum of $X_3(f)$ in positive frequencies. The bandpass filter will also keep negative frequencies from $-f_3$ to $-f_2$. *6 points*.

$\boldsymbol{f_2 = 48}$ **MHz and** $\boldsymbol{f_3 = 64}$ **MHz**

(d) Draw the spectrum for $X_4(f)$. *6 points*.



JSK Sec. 2.5, 2.8, 3.3, 3.4, 4.2, 6.1, 7.1 & 7.2 | Lectures 4 & 5

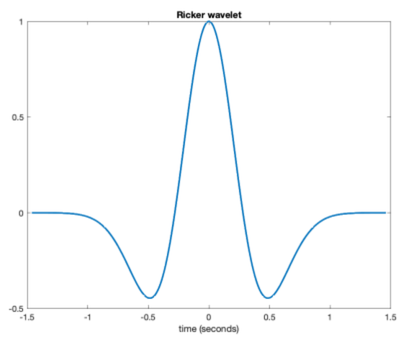
Labs 2 & 3 | HW 0.3, 1.1, 1.3, 2.1, 2.2, 2.3

Problem 1.4. *Design Tradeoffs*. 22 points.

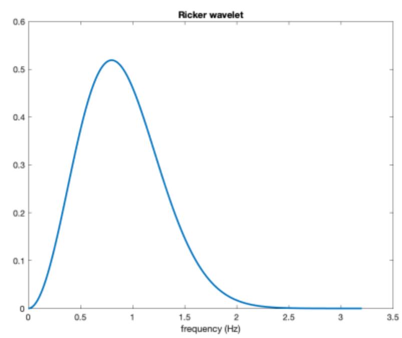
The Laplace of Gaussian filter is used to detect edges in images. It is the equivalent to a cascade of a Laplace highpass filter (second derivative) and a Gaussian lowpass filter to smooth the result.

An example continuous-time impulse response of a 1-D Laplace of Gaussian filter is the Ricker wavelet. The Richer wavelet is infinite in extent, and is plotted in the time and frequency domains below:

Continuous Time (CT) Domain



CT Frequency Domain

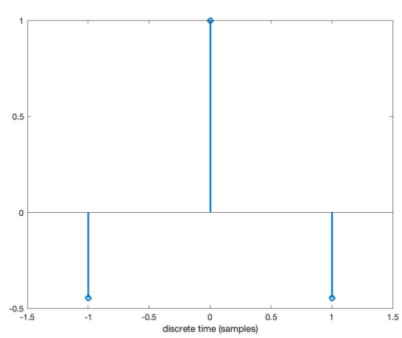


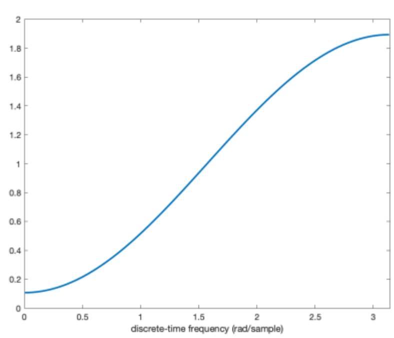
Below are two sampled versions of the Ricker wavelet (left) to be used as finite impulse response (FIR) impulse responses. The associated frequency response is plotted on the right. For each sampled version,

(a) Describe the signal quality in the time and frequency domains vs. the CT Ricker wavelet
(b) Give a formula for run-time computational complexity with each filter running at sampling rate f_s

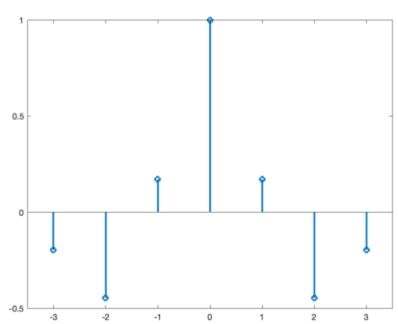
Discrete-Time (DT) Domain

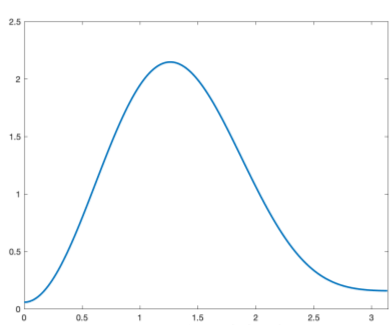
DT Frequency Domain





3-coefficient Ricker filter (*6 points*)
(a) DT impulse response captures the peak and both valleys of CT Ricker wavelet. However, DT frequency response is highpass vs. bandpass CT frequency response. Linear phase. Low signal quality.
(b) Run-time implementation complexity for an FIR filter with 3 coefficients is $3f_s$ multiplications/s





7-coefficient Ricker filter (*6 points*)
(a) DT impulse response captures the peak and both valleys of CT Ricker wavelet, and DT frequency response has a similar bandpass shape as CT frequency response. Linear phase. High signal quality.
(b) Run-time implementation complexity for an FIR filter with 7 coefficients is $7f_s$ multiplications/s

(c) Under what conditions would you advocate using the 3-point vs. the 7-point Ricker filter? *10 points*.

Use the 3-coefficient Ricker filter if severely limited in run-time computational resources; otherwise, use the 7-coefficient Ricker filter because it has higher signal quality. Signal quality increases with more coefficients.

```matlab
% Matlab code to generate the figures for problem 1.4

%
https://wiki.seg.org/wiki/Dictionary:Ricker_wavelet?__cf_chl_tk=9nmYoFeme7Fdjn5iSF
fT2n3NEtS4oyBglXC72Cd8ML0-1761102764-1.0.1.1-
JpFzK6UIJNwkexHr3qqHwrgnIZoSSNznLFSA5pjq_TY

% Continuous-Time Ricker Wavelet
fM = 0.8;
TD = sqrt(6) / (pi * fM);
t = -1.5*TD : (TD/1000) : 1.5*TD;
exp1 = (pi^2)*(fM^2)*(t.^2);
y = (1 - 2*exp1) .* exp(-exp1);
figure;
plot(t, y, 'LineWidth', 2);
xlabel('continuous time (seconds)');
title('Ricker wavelet');

% Continuous-Time Frequency response dies out
f = 0 : (4*fM)/1000 : 4*fM;
Y1 = 2/sqrt(pi);
Y2 = (f.^2) / (fM^3);
Y3 = exp(-(f.^2) / (fM^2) );
Y = Y1 .* Y2 .* Y3;
figure;
plot(f, Y, 'LineWidth', 2);
xlabel('frequency (Hz)');
title('Ricker wavelet');

% Sample to pick the peak and valleys
% (a) Three coefficients
tvalley = TD / 2;
Ts = tvalley;
n = -1 : 1;
t = n * Ts;
exp1 = (pi^2)*(fM^2)*(t.^2);
h = (1 - 2*exp1) .* exp(-exp1);
figure;
stem(n, h, 'LineWidth', 2);
xlabel('discrete time (samples)');
xlim( [-1.5 1.5] );
figure;
[H, W] = freqz(h);
plot(W, abs(H), 'LineWidth', 2);
xlabel('discrete-time frequency (rad/sample)');
xlim( [0, pi] );

% (b) Seven coefficients
tvalley = TD / 2;
Ts = tvalley / 2;
nmax = round(1.5*TD/Ts);
n = -3 : 3;
t = n * Ts;
exp1 = (pi^2)*(fM^2)*(t.^2);
h = (1 - 2*exp1) .* exp(-exp1);
figure;
stem(n, h, 'LineWidth', 2);
```

```
xlabel('discrete time (samples)');
xlim( [-3.5 3.5] );
figure;
[H, W] = freqz(h);
plot(W, abs(H), 'LineWidth', 2);
xlim( [0, pi] );
xlabel('discrete-time frequency (rad/sample)');
```