Solution Set for Homework #7

By Prof. Brian Evans

November 12, 2021 ***Version 2.0 with 2.0 Comments in Blue for 7.2, 7.3 and 7.4***

PROBLEM 1: DISCRETE-TIME AVERAGING FILTERS. ***34 points.***

For a discrete-time LTI system with input signal $x[n]$ and impulse response $h[n]$, the output signal $y[n]$ is the convolution of $h[n]$ and $x[n]$:

$$y[n] = h[n] * x[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k]$$

(a) Compute the output $y[n]$ when the input $x[n]$ is a rectangular pulse of amplitude 1 for $n = 0, 1, \ldots N_x - 1$ and amplitude 0 otherwise, and $x[n]$ is filtered by an LTI unnormalized averaging filter whose impulse response $h[n]$ is a rectangular pulse of amplitude 1 for $n = 0, 1, \ldots N_h - 1$ and amplitude 0 otherwise. Assume $N_x \neq N_h$.

i. Write the difference equation relating output $y[n]$ and input $x[n]$. *3 points.*

***Solution*: The impulse response $h[n]$ has extent $n = 0, 1, \ldots N_h - 1$:**

$$\boldsymbol{y[n] = h[n] * x[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k] = \sum_{k=0}^{N_h-1} h[k]\, x[n-k]}$$

$$\boldsymbol{y[n] = h[0]\, x[n] + h[1]\, x[n-1] + \cdots + h[N_h - 1]\, x[n-(N_h-1)]}$$

Since $h[n] = 1$ for $n = 0, 1, \ldots, N_h - 1$,

$$\boldsymbol{y[n] = x[n] + x[n-1] + x[n-2] + \cdots + x[n-(N_h-1)]}$$

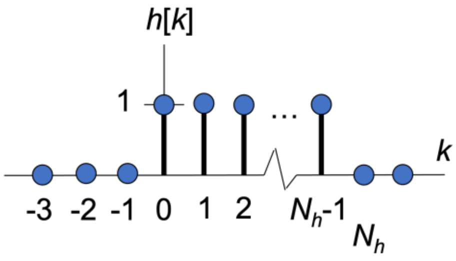
ii. What are the initial conditions and what values should they be set to? *3 points.*

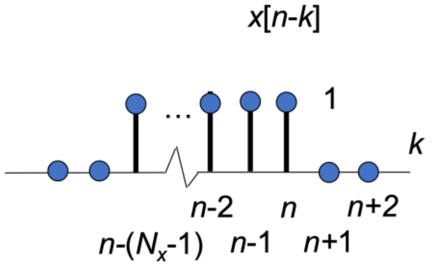
***Solution:* Initial conditions can be discovered by computing the first several values of $y[n]$ for n ≥ 0: $y[0] = x[0] + x[-1] + x[-2] + \cdots + x[-(N_h-1)]$.**

The initial conditions are $x[-1], x[-2], \ldots, x[-(N_h-1)]$. They must be set to zero as necessary conditions for LTI system properties to hold.

iii. Develop a formula for $y[n] = h[n] * x[n]$ using the convolution definition in terms of N_x and N_h. Show the intermediate steps in computing the convolution. *6 points.*

***Solution:* First, we'll define $N_{min} = \min(N_h, N_x)$ and $N_{max} = \max(N_h, N_x)$ and the length of the convolution result $N_y = N_h + N_x - 1$. As we flip and slide $x[n-k]$ across $h[k]$, where the shift n is with respect to the convolution variable k, the extent of $h[k]$ is $0 \le k \le N_h - 1$ and the extent of $x[n-k]$ is $n - (N_x - 1) \le k \le n$.**





There are five cases to consider:

1. ***No overlap*. $n < 0$. Amplitude is 0.**
2. ***Partial overlap*. $0 \le n \le N_{min} - 1$. Amplitude is $(n+1)$.**
   Initial overlap of one sample at $n = 0$ with a product of one. Each shift by one in n adds one more overlapping sample with product of one.

$$y[n] = \sum_{k=0}^{n} h[k]\, x[n-k] = \sum_{k=0}^{n} 1 = (n+1)$$

3. ***Complete overlap*. $N_{min} - 1 \le n \le N_{max} - 1$. Amplitude is N_{min}.**
   Here, N_{min} samples overlap, and each sample has a value of one.
4. ***Partial overlap*. $N_{max} \le n \le N_y - 1$. Amplitude is $N_y - n$.**
   Amplitude reduces by one each time *n* is incremented.

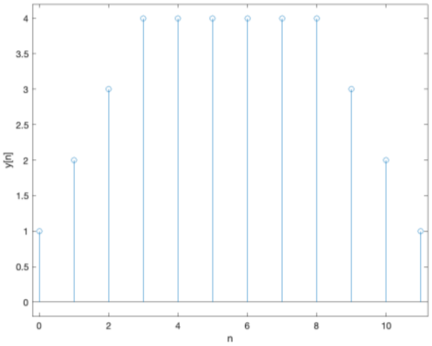
$$y[n] = \sum_{k=n-(N_x-1)}^{N_h-1} 1 = (N_h - 1) + (N_x - 1) + 1 - n = N_y - n$$

5. **No overlap. $n \ge N_y$. Amplitude is 0.**

iv. Validate the formula for $y[n]$ by using Matlab to compute the convolution for $N_x =$ 9 samples and $N_h = 4$ samples. *3 points.*

***Solution:* Using Matlab:**

```
h = ones(1, 4);
x = ones(1, 9);
y = conv(h, x);
n = 0 : 11;
stem(n, y);
xlim([-0.2 11.2]);
ylim([-0.2, 4.2]);
xlabel('n');
ylabel('y[n]');
```



v. Use the *z*-transform to find $y[n] = h[n] * x[n]$. Track region of convergence. *3 points.*

***Solution:* Convolution in the discrete-time domain becomes a product in the z-transform domain: $Y(z) = H(z)X(z)$. Here,**

$$H(z) = \sum_{n=0}^{N_h-1} h[n]\, z^{-n} = 1 + z^{-1} + z^{-2} + \cdots + z^{-(N_h-1)} \text{ if } z \ne 0$$

$$X(z) = \sum_{n=0}^{N_x-1} x[n]\, z^{-n} = 1 + z^{-1} + z^{-2} + \cdots + z^{-(N_x-1)} \text{ if } z \ne 0$$

Convolution in the time domain has become polynomial multiplication in the z-transform domain. The polynomial multiplication will produce a polynomial whose coefficients will fit a trapezoidal pattern because $N_x \ne N_h$.

$$X(z) = 1 + 2\, z^{-1} + 3\, z^{-2} + \cdots + 2\, z^{-(N_y-2)} + z^{-(N_y-1)} \text{ if } z \ne 0$$

(a) An LTI system outputs the weighted average of the previous output value $y[n-1]$ and current input value $x[n]$ using difference equation $y[n] = 0.9\, y[n-1] + 0.1\, x[n]$ for $n \geq 0$

i. What are the initial conditions and what value should they be set to? *3 points.*

***Solution*: Initial conditions can be discovered by computing the first values of $y[n]$ for $n \geq 0$: $y[0] = 0.9\, y[-1] + 0.1\, x[0]$. Initial condition $y[-1] = 0$ for LTI to hold.**

ii. Compute a formula for the impulse response $h[n]$ for the system. *3 points.*

***Solution:* To compute the impulse response $h[n]$, we let the input be an impulse $\delta[n]$: $h[n] = 0.9\, h[n-1] + 0.1\, \delta[n]$ for $n \geq 0$ with $h[-1] = 0$. We'll compute the output values and infer the impulse response as in LTI Example #2 on Lecture Slide 11-4:**

$$h[0] = 0.9\, h[-1] + 0.1\, \delta[0] = 0.1$$
$$h[1] = 0.9\, h[0] + 0.1\, \delta[1] = 0.1\, (0.9)$$
$$h[2] = 0.9\, h[1] = 0.1\, (0.9)^2 \ldots$$

Inferring the pattern gives $h[n] = 0.1\, (0.9)^n\, u[n]$.

iii. Develop a formula for $y[n] = h[n] * x[n]$ using the convolution definition when the input signal is $x[n] = 0.8^n\, u[n]$. *6 points.*

***Solution:* With $h[n] = 0.1\, (0.9)^n\, u[n]$ and $x[n] = 0.8^n\, u[n]$.**

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k] = \sum_{k=-\infty}^{\infty} \left(0.1\, (0.9)^k\, u[k]\right) \left((0.8)^{n-k}\, u[n-k]\right)$$

$u[k]$ is 1 for $k \geq 0$ and 0 otherwise. $u[n-k]$ is 1 when $n-k \geq 0$ or $k \leq n$, and 0 otherwise. Limits of summation become $k = 0$ and $k = n$ and $n \geq 0$ because $k \geq 0$:

$$y[n] = u[n] \sum_{k=0}^{n} 0.1\, (0.9)^k\, (0.8)^{n-k} = 0.1\, (0.8)^n\, u[n] \sum_{k=0}^{n} (0.9)^k\, (0.8)^{-k}$$

$$y[n] = 0.1\, (0.8)^n\, u[n] \sum_{k=0}^{n} \left(\frac{0.9}{0.8}\right)^k = 0.1\, (0.8)^n \frac{1 - \left(\frac{0.9}{0.8}\right)^{n+1}}{1 - \left(\frac{0.9}{0.8}\right)} u[n]$$

$$y[n] = 0.1 \frac{(0.8)^{n+1} - (0.9)^{n+1}}{0.8 - 0.9} u[n] = -(0.8)^{n+1}\, u[n] + (0.9)^{n+1}\, u[n]$$

For more info, see Handout F Convolution of Two Causal Exponential Sequences.

iv. Use the *z*-transform to find $y[n] = h[n] * x[n]$. Track region of convergence. *4 points.*

***Solution:* Convolution in the discrete-time domain becomes a product in the z-transform domain: $Y(z) = H(z)X(z)$. Here,**

$H(z) = \frac{0.1}{1-0.9\, z^{-1}}$ for $|z| > 0.9$ and $X(z) = \frac{1}{1-0.8\, z^{-1}}$ for $|z| > 0.8$

$Y(z) = \left(\frac{0.1}{1-0.9\, z^{-1}}\right) \left(\frac{1}{1-0.8\, z^{-1}}\right)$ for $|z| > \max(0.8, 0.9)$

We'll use partial fractions decomposition to express the transfer function as a sum of two first-order terms and apply the inverse z-transform. See Ex. 8-10 on p. 219.

$Y(z) = 0.1 \left(\frac{A}{1-0.9\, z^{-1}} + \frac{B}{1-0.8\, z^{-1}}\right)$ which gives $y[n] = A\, (0.9)^n\, u[n] + B(0.8)^n\, u[n]$

Using the method from Ex. 8-10 on p. 219, $A = \frac{1}{1-\frac{0.8}{0.9}} = 0.9$ and $B = \frac{1}{1-\frac{0.9}{0.8}} = -0.8$.

PROBLEM 2: TRANSFER FUNCTION & FREQUENCY RESPONSE CONNECTIONS. ***16 points.***

Signal Processing First, problem P-8.16, page 242.

In addition, for each of the four filters,

i. give a formula for the transfer function in the z-domain including the region of convergence,

ii. give a formula for the frequency response from the transfer function in the z-domain in part i. why can we convert from the transfer function in the z-domain to a frequency response directly?

iii. plot the magnitude response in the frequency domain

iv. indicate the frequency selectivity as lowpass, highpass, bandpass, or bandstop**.**

Connecting filter poles/zeros to its frequency selectivity: Please see lecture slide 11-7 and watch a YouTube video in spring 2014 for the Real-Time DSP Lab course from the 1:29 to the 22:25 mark and from 43:01 to the end (50:51 mark). When poles and zeros are separated in angle, the angles of the poles near the unit circle indicate the frequencies in the passband(s), and the angles of the zeros near or on the unit circle indicate the frequencies in the stopband(s). Poles must be inside the unit circle for bounded-input bounded-output (BIBO) stability. Please see lecture slides 11-8 and 11-9.

***Solution:* Pole-Zero Plot #1**

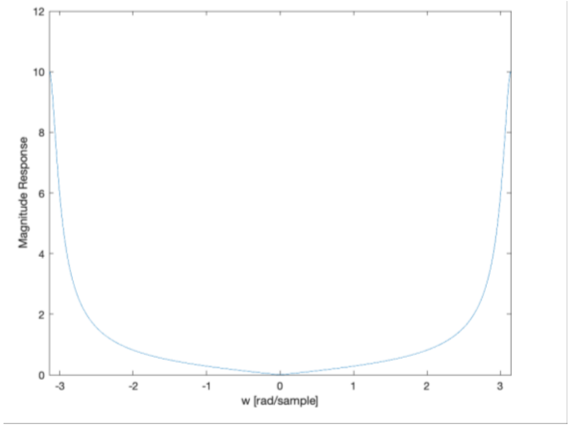
$\boldsymbol{H(z) = C\,\frac{z-z_0}{z-p_0}}$ **for** $|z| > |p_0|$ **where** $z_0 = 1$ **and** $p_0 = -0.9$ **and *C* is given in Matlab code.**

Because the region of convergence $|z| > 0.9$ **includes the unit circle** $e^{j\omega}$**,**

$\boldsymbol{H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0}}$ **for** $|z| > |p_0|$

```
%%% Specify the filter
z0 = 1;
p0 = -0.9;
C = 0.5;    %% to match (D)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a highpass selectivity and matches (D).

***Solution:* Pole-Zero Plot #2**

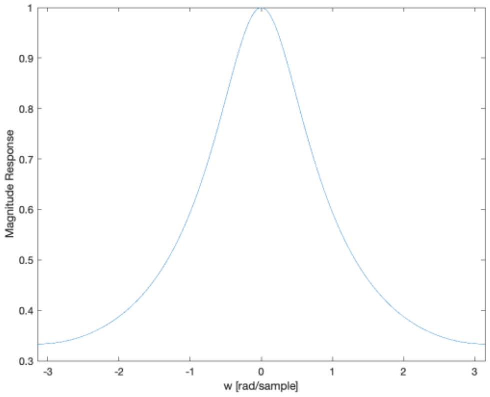
$\boldsymbol{H(z) = C\,\frac{z-z_0}{z-p_0}}$ **for** $|z| > |p_0|$ **where** $z_0 = 0$ **and** $p_0 = 0.5$ **and *C* is given in Matlab code.**

Because the region of convergence $|z| > 0.5$ **includes the unit circle** $e^{j\omega}$**,**

$\boldsymbol{H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0}}$ **for** $|z| > |p_0|$

```
%%% Specify the filter
z0 = 0;
p0 = 0.5;
C = 0.5;    %% to match (B)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a lowpass selectivity and matches (B).

Solution: **Pole-Zero Plot #3**

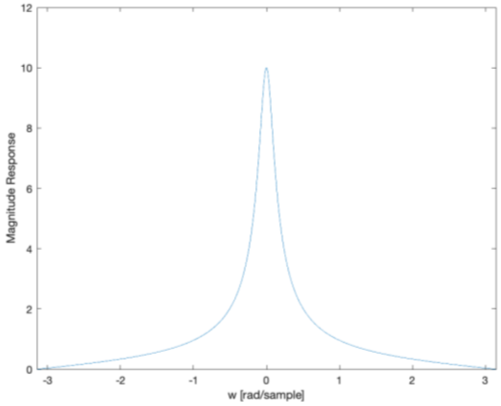
$H(z) = C\,\frac{z-z_0}{z-p_0}$ for $|z| > |p_0|$ where $z_0 = -1$ and $p_0 = 0.9$ and C is given in Matlab code.

Because the region of convergence $|z| > 0.9$ includes the unit circle $e^{j\omega}$,

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0} \text{ for } |z| > |p_0|$$

```
%%% Specify the filter
z0 = -1;
p0 = 0.9;
C = 0.5;    %% to match (A)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a lowpass selectivity and matches (A).

Solution: **Pole-Zero Plot #4**

$H(z) = C\,\frac{(z-z_0)(z-z_1)}{(z-p_0)(z-p_1)}$ for $|z| > \max\{\,|p_0|, |p_1|\,\}$ where $z_0 = 0$, $z_1 = 0$, $p_0 = 0.9\,e^{j\frac{\pi}{6}}$, $p_1 = 0.9\,e^{-j\frac{\pi}{6}}$ and C is given in the Matlab code below.

Because the region of convergence $|z| > 0.9$ includes the unit circle $e^{j\omega}$,

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{(e^{j\omega}-z_0)(e^{j\omega}-z_1)}{(e^{j\omega}-p_0)(e^{j\omega}-p_1)} \text{ for } |z| > |p_0|$$

We can expand the numerator and denominator of the transfer function $H(z)$:

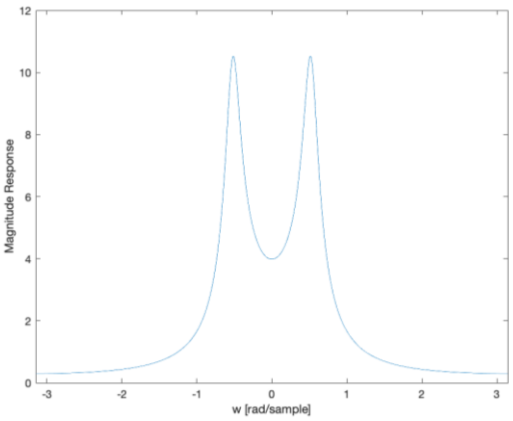
$$\boldsymbol{H(z) = C\,\frac{(z-z_0)(z-z_1)}{(z-p_0)(z-p_1)} = C\,\frac{z^2-(z_0+z_1)\,z+z_0z_1}{z^2-(p_0+p_1)\,z+p_0p_1} = C\,\frac{b_0z^2+b_1z+b_2}{a_0z^2+a_1z+a_2}}$$

```
%%% Specify the filter
%%% Zeros and numerator coefficients
z0 = 0;
z1 = 0;
b0 = 1;
b1 = -(z0+z1);
b2 = z0*z1;
numerCoeffs = [b0 b1 b2];

%%% Poles and denominator coefficients
p0 = 0.9*exp(j*pi/6);
p1 = 0.9*exp(-j*pi/6);
a0 = 1;
a1 = -(p0+p1);
a2 = p0*p1;
denomCoeffs = [a0 a1 a2];

%%% Gain for the filter
C = 1;

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (b0 + b1*exp(-j*w) + b2*exp(-j*2*w));
Hdenom = (a0 + a1*exp(-j*w) + a2*exp(-j*2*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a bandpass selectivity and matches (E).

<u>PROBLEM 3: CONTINUOUS-TIME SYSTEM PROPERTIES</u>. *16 points.*

Signal Processing First, problem P-9.2, page 279.

In each of the following cases, state whether or not the continuous time system is (i) linear, (ii) time-invariant, (iii) stable, and (iv) causal. In each case, x(t) he represents the input and y(t) represents the corresponding output of the system. Provide a brief justification, either in the form of mathematical equations or statements in the form of complete, crack, push sentences. Remember, in order to show the system does not have the property, all you have to do is give an example and put up with is not satisfy the condition of the property.

(a) An exponential system: $y(t) = e^{x(t+2)}$. **Used in speech denoising and machine learning**.

(b) A phase modulator: $y(t) = \cos(\omega_c t + x(t))$. **<u>Phase modulation</u> is used for low-power transmission in IoT systems. The digital version, <u>Phase Shift Keying</u>, is used in RFID and Bluetooth and higher transmit power systems, such as Wi-Fi and cellular communications.**

(c) An amplitude modulator: $y(t) = (A + x(t))\cos(\omega_c t)$. **Used in AM radio. Amplitude modulation (without the offset of A) is used in Wi-Fi, cellular and cable modems.**

(d) Take the even part of the input signal: $y(t) = \frac{x(t)+x(-t)}{2}$. **Primarily for theoretical analysis.**

***Solution:* Linearity**

When checking each system for linearity, we can use the quick test of input signal of 0 for all time, which is a by-product of the homogeneity property when the input signal is scaled by a = 0. If the output signal is not zero for all time, then the system is not linear. If the output is zero for all time, then we'll have to apply the mathematical definitions for homogeneity and additivity.

(a) $y(t) = e^{x(t+2)} = e^0 = 1$. Fails all-zero input test. Not linear.

(b) $y(t) = \cos(\omega_c t + x(t)) = \cos(\omega_c t)$. Fails all-zero input test. Not linear.

(c) $y(t) = (A + x(t))\cos(\omega_c t) = A\cos(\omega_c t)$. Fails all-zero input test. Not linear.

(d) $y(t) = \frac{x(t)+x(-t)}{2} = 0$. Passes all-zero input test. Check for homogeneity and additivity.

- ***Homogeneity*. Input $a\,x(t)$. Output $y_{scaled}(t) = \frac{(a\,x(t))+(a\,x(-t))}{2} = a\,\frac{x(t)+x(-t)}{2} = a\,y(t)$**
- ***Additivity*. Input $x_1(t) + x_2(t)$. Output**

$$y_{additive}(t) = \frac{(x_1(t)+x_2(t))+(x_1(-t)+x_2(-t))}{2} = \frac{x_1(t)+x_1(-t)}{2} + \frac{x_2(t)+x_2(-t)}{2} = y_1(t) + y_2(t)$$

Yes, system (d) is linear.

***Solution:* Time-Invariance**

For time-invariant system, shift of the input signal by any real-valued τ causes the same shift in output signal, i.e. $x(t$ - $\tau)$ means $y(t$ - $\tau)$ for all τ.

(a) $y(t) = e^{x(t+2)}$. Input $x(t-\tau)$. Output $y_{shifted}(t) = e^{x((t-\tau)+2)} = e^{x(t-\tau+2)} = y(t-\tau)$. *Time-invariant.*

(b) $y(t) = \cos(\omega_c t + x(t))$. The signal $\cos(\omega_c t)$ is part of the system and does not shift in time when the input shifts in time. *Time-varying.*

(c) $y(t) = (A + x(t))\cos(\omega_c t)$. The signal $\cos(\omega_c t)$ is part of the system and does not shift in time when the input shifts in time. *Time-varying.*

(d) $y(t) = \frac{x(t)+x(-t)}{2}$. The copy of the input signal shifts in the same way that the input signal shifts. The copy that is reversed in time gives the negated shift. *Time-varying.*

***Solution:* Stability**

A stable system will always produce a bounded amplitude output signal when given a bounded amplitude input signal. Let $|x(t)| < B < \infty$

(a) $|y(t)| = \left|e^{x(t+2)}\right| \le |e^B|$. Bounded output. *Stable.*

(b) $y(t) = \cos(\omega_c t + x(t))$. Output will always be in range [-1, 1] regardless of the value of $x(t)$. Bounded output. *Stable.*

(c) $|y(t)| = |(A + x(t))\cos(\omega_c t)| \le |A + x(t)|\,|\cos(\omega_c t)| \le |A + x(t)| \le |A| + |x(t)| \le |A| + B$. Bounded output. *Stable.*

(d) $|y(t)| = \left|\frac{x(t)+x(-t)}{2}\right| \le \frac{|x(t)|}{2} + \frac{|x(-t)|}{2} \le B$. **Bounded output.** ***Stable.***

***Solution:* Causality**

A causal system depends only on current and previous input values and/or previous output value to compute an output value.

(a) $y(t) = e^{x(t+2)}$. **Depends on input 2 seconds in the future.** ***Not Causal.***

(b) $y(t) = \cos(\omega_c t + x(t))$. **Only depends on the current input value** $x(t)$. ***Causal.***

(c) $y(t) = (A + x(t)) \cos(\omega_c t)$. **Only depends on the current input value** $x(t)$. ***Causal.***

(d) $y(t) = \frac{x(t)+x(-t)}{2}$. **Check specific values of time *t*.**

- **When *t* = 0, *y*(0) = (*x*(0) + *x*(0)) / 2 = *x*(0). Causal.**
- **When *t* = 2, *y*(2) = (*x*(2) + *x*(-2)) / 2. Causal.**
- **When *t* = -2, *y*(-2) = (*x*(-2) + *x*(2)) / 2. Depends on future input *x*(2).** ***Not Causal.***

PROBLEM 4: CONTINUOUS-TIME AVERAGING FILTERS. ***34 points.***

For a **continuous**-time LTI system with input signal $x(t)$ and impulse response $h(t)$, the output signal $y(t)$ is the convolution of $h(t)$ and $x(t)$:

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\lambda)\, x(t-\lambda)\, d\lambda$$

(a) Compute the output $y(t)$ when the input $x(t)$ is a rectangular pulse of amplitude 1 for $t \in [0, T_x]$ and amplitude 0 otherwise and $x[n]$ is filtered by an LTI unnormalized averaging filter whose impulse response $h(t)$ is a rectangular pulse of amplitude 1 for $t \in [0, T_h]$ and amplitude 0 otherwise. Assume $T_x \neq T_h$.

i. Write an equation relating output $y(t)$ and input $x(t)$. *4 points*

Solution: **With** $h(t) = 1$ **for** $t \in [0, T_h]$,

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\lambda)\, x(t-\lambda)\, d\lambda = \int_{0}^{T_h} x(t-\lambda)\, d\lambda$$

We can apply the change of variables $u = t - \lambda$. $du = -d\lambda$. **As** $\lambda \to 0$, $u \to t$. **As** $\lambda \to T_h$, $u \to t - T_h$. **This gives**

$$y(t) = \int_{0}^{T_h} x(t-\lambda)\, d\lambda = \int_{t-T_h}^{t} x(u)\, du$$

The averaging filter integrates the input signal over the previous T_h **seconds.**

Although not asked, this filter is stable unlike the integrator over all time in part (b).

ii. What is(are) the initial condition(s) and what value should it(they) be set to? *3 points*

Solution: **The initial conditions are the memory of the previous T_h seconds of the input signal. This memory (signal buffer) would have to be initially zeroed out.**

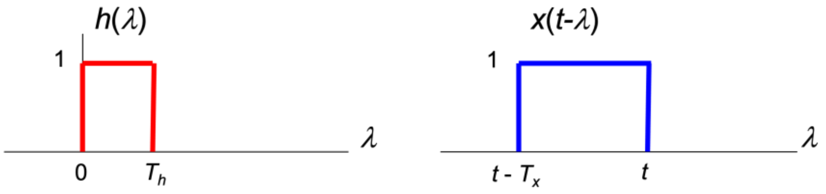
iii. Develop a formula for $y(t) = h(t) * x(t)$ using the convolution definition in terms of T_x and T_h. Show the intermediate steps in computing the convolution. *6 points*

Solution: **Trapezoid of duration $T_y = T_h + T_x$. Let $T_{min} = \min(T_h, T_x)$ and $T_{max} = \max(T_h, T_x)$. As we flip and slide one rectangular pulse against the other, partial overlap occurs from 0 to T_{min} seconds, complete overlap from T_{min} to T_{max} seconds, and partial overlap from T_{max} to T_y seconds.**

$$y(t) = \begin{cases} 0 & \text{for } t < 0 \\ t & \text{for } 0 \le t < T_{min} \\ T_{min} & \text{for } T_{min} \le t < T_{max} \\ T_y - t & \text{for } T_{max} \le t < T_y \\ 0 & \text{for } t > T_y \end{cases}$$

The details of the flip-and-slide are analogous to problem 1(a)iii for the discrete-time convolution of two rectangular pulses, and explained next.

We will hold $h(t)$ in place and flip and slide $x(t)$about $h(t)$:



There are five cases to consider:

1. **No overlap. $t < 0$. Amplitude is 0.**
2. **Partial overlap. $0 \le t < T_{min}$. Amplitude is t.**
   Initial overlap at the origin, and integration of a point is zero. Each shift by a time unit adds that much to the area.

$$y(t) = \int_{-\infty}^{\infty} h(\lambda)\, x(t-\lambda)\, d\lambda = \int_{0}^{t} d\lambda = t$$

3. **Complete overlap. $T_{min} \le t < T_{max}$. Amplitude is T_{min}.**
   Here, T_{min} seconds overlap, and each amplitude has a value of one.
4. **Partial overlap. $T_{max} \le n < T_y$. Amplitude is $T_y - t$.**
   Amplitude reduces by the same amount that t is shifted.

$$y(t) = \int_{-\infty}^{\infty} h(\lambda)\, x(t-\lambda)\, d\lambda = \int_{t-T_x}^{T_h} d\lambda = T_h - (t - T_x) = T_y - t$$

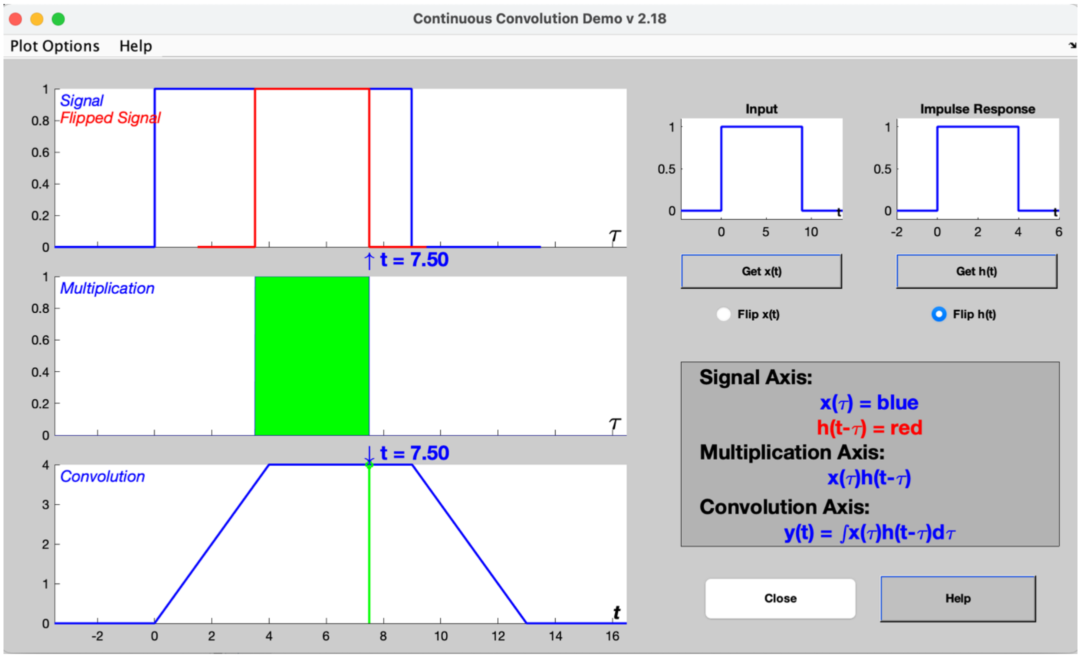
5. **No overlap. $t \ge T_y$. Amplitude is 0.**

iv. Validate the formula for $y(t)$ to compute the convolution for $T_x = 9$ seconds and $T_h = 4$ seconds. *3 points.*

***Solution*: Trapezoid of duration $T_y = 4 + 9 = 13$ seconds. Let $T_{min} = \min(4, 9) = 4$ seconds and $T_{max} = \max(4, 9) = 9$ seconds. As we flip and slide one rectangular pulse against the other, partial overlap occurs from 0 to T_{min} seconds, complete overlap from T_{min} to T_{max} seconds, and partial overlap from T_{max} to T_y seconds.**

$$y(t) = \begin{cases} 0 & \text{for } t < 0 \\ t & \text{for } 0 \leq t < 4 \\ 4 & \text{for } 4 \leq t < 9 \\ 13 - t & \text{for } 9 \leq t < 13 \\ 0 & \text{for } t > 13 \end{cases}$$

Using the cconvdemo from *Signal Processing First*,



(b) When an input signal has an average value of zero, i.e. the DC component is zero, an LTI integrator can be used as an averaging filter. The differential equation governing the input-output relationship is

$$y(t) = \int_{0^-}^{t} x(\tau)\, d\tau \quad \text{for } t \geq 0$$

The integrator was operating before *t* = 0 seconds, but we aren't able to observe that. We are observing the system starting at *t* = 0 seconds.

Since we are starting the integration at *t* = 0 seconds, there is ambiguity as to whether Dirac delta signal would be included at time 0. We can use 0$^-$ as the lower limit to indicate that integration starts at time 0 before the impulse occurs.

i. What is(are) the initial condition(s) and what value should it(they) be set to? *3 points*

***Solution*: The initial condition is the initial integration value *y*(0). It should be set to zero as a necessary condition for LTI properties to hold.**

ii. What is the impulse response? *3 points*

***Solution:* For input $x(t) = \delta(t)$, the output is the impulse response**

$$\boldsymbol{h(t) = \int_{0^-}^{t} \delta(\tau)\, d\tau = u(t)}$$

iii. Develop a formula for $y(t) = h(t) * x(t)$ using the convolution definition when the input signal is $x(t) = u(t)$. Note that $x(t)$ has bounded amplitude. *9 points*

***Solution:* Using the convolution definition,**

$$\boldsymbol{y(t) = h(t) * x(t) = u(t) * u(t) = \int_{-\infty}^{\infty} u(\lambda)\, u(t-\lambda)\, d\lambda}$$

$u(\lambda)$ is 1 for $\lambda \geq 0$ and 0 otherwise, whereas $u(t-\lambda)$ is 1 for $t - \lambda \geq 0$ or $\lambda \leq t$. Also, note that $t \geq 0$ because $\lambda \geq 0$:

$$\boldsymbol{y(t) = \int_{0}^{t} d\lambda = t\, u(t)}$$

iv. Is the LTI integrator bounded-input bounded-output (BIBO) stable? Your work in part iii might be helpful. *3 points*

***Solution:* For input signal $x(t) = u(t)$, whose amplitude is bounded in [0, 1], the output $y(t) = t\, u(t)$ grows without bound as $t \to \infty$. LTI integrator is not stable.**