

Homework 2 Solutions

GSI : Pulkit Grover

Problem 1 (*Complex numbers*)

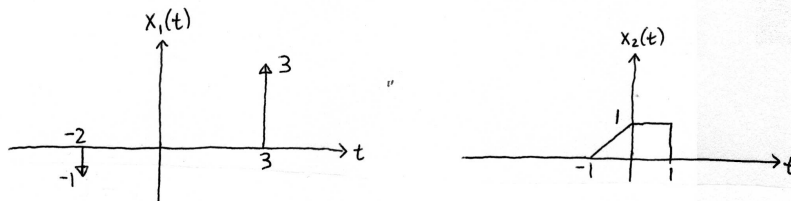
The polar representation is more convenient for finding roots. The N^{th} root of a complex number $c = re^{j\theta}$ can be found by simply raising c to the $\frac{1}{N}$ power: $c^{\frac{1}{N}} = r^{\frac{1}{N}}e^{j\frac{\theta}{N}}$. Notice however, $c = re^{j\theta} = re^{j(\theta+2\pi)} = re^{j(\theta+2n\pi)}$ for all $n \in \mathbb{Z}$. Therefore, $r^{\frac{1}{N}}e^{j\frac{(\theta+2n\pi)}{N}}$ is an N^{th} root for all $n \in \mathbb{Z}$. Does that mean that there is an infinite number of N^{th} roots? Not quite. If you look closely at the expression, you will notice that it repeats every N integers. Therefore, there are only N unique roots (also remember that every polynomial of order N has exactly N complex roots). So we only need to consider $0 \leq n < N$.

Now back to our example, $c_2 = -16 = 16e^{j\pi}$. The four fourth roots of c_2 are $2e^{j\frac{\pi}{4}}, 2e^{j\frac{3\pi}{4}}, 2e^{j\frac{5\pi}{4}}, 2e^{j\frac{7\pi}{4}}$.

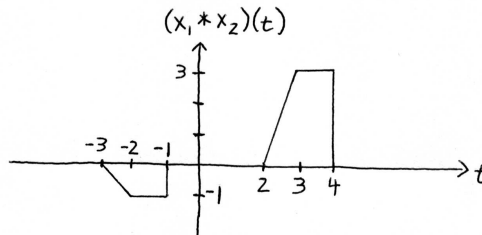
Problem 2 (*Graphical Convolution.*)

① (d) $x_1(t) = -\delta(t+2) + 3\delta(t-3)$

$$x_2(t) = \begin{cases} 1+t & -1 \leq t \leq 0 \\ 1 & 0 \leq t \leq 1 \\ 0 & |t| > 1 \end{cases}$$

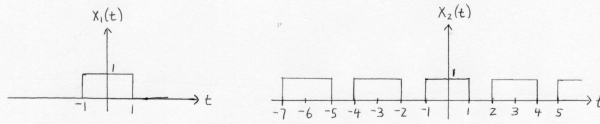


Convolution with delta functions is easy.
 Just remember that $x(t) * \delta(t-t_0) = x(t-t_0)$



$$(b) \quad x_1(t) = \begin{cases} 1 & |t| \leq 1 \\ 0 & |t| > 1 \end{cases}$$

$$x_2(t) = \sum_{k=-\infty}^{\infty} x_1(t+3k)$$



We can see that
$$x_2(t) = \sum_{k=-\infty}^{\infty} x_1(t) * \delta(t+3k)$$

$$= x_1(t) * \sum_{k=-\infty}^{\infty} \delta(t+3k)$$

So,
$$x_1(t) * x_2(t) = x_1(t) * \left(x_1(t) * \sum_{k=-\infty}^{\infty} \delta(t+3k) \right)$$

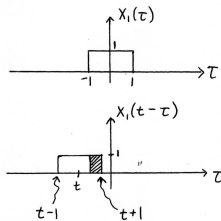
$$= (x_1(t) * x_1(t)) * \sum_{k=-\infty}^{\infty} \delta(t+3k)$$

$$= \tilde{x}(t) * \sum_{k=-\infty}^{\infty} \delta(t+3k)$$

$$= \sum_{k=-\infty}^{\infty} \tilde{x}(t+3k)$$

where $\tilde{x}(t) = x_1(t) * x_1(t)$

Now, let's graphically convolve $x_1(t)$ with $x_1(t)$



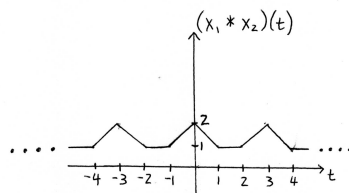
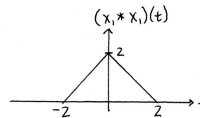
If $t+1 < -1$ then $(x_1 * x_1)(t) = 0$

If $-1 \leq t+1 \leq 1$ then $(x_1 * x_1)(t) = (t+1) - (-1) = t+2$

If $-1 \leq t-1 \leq 1$ then $(x_1 * x_1)(t) = 1 - (t-1) = 2-t$

If $t-1 > 1$ then $(x_1 * x_1)(t) = 0$

$$\Rightarrow (x_1 * x_1)(t) = \begin{cases} 2 - |t| & \text{if } |t| \leq 2 \\ 0 & \text{if } |t| > 2 \end{cases}$$



(c) We will be encountering box and triangle functions very often in this class; so we might as well give them special

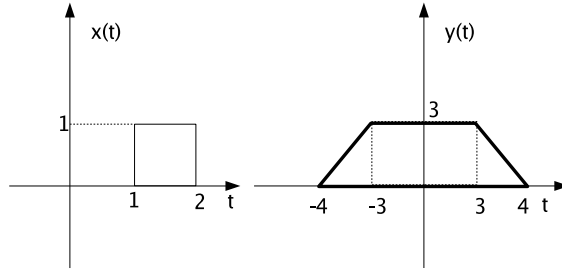


Figure 1: Problem 2(c)

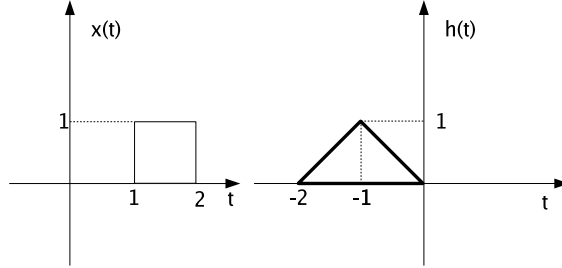


Figure 2: Problem 2(d)

names. So let us define:

$$\Pi(t) = u(t + \frac{1}{2}) - u(t - \frac{1}{2})$$

$$\Lambda(t) = \Pi(t) * \Pi(t)$$

$\Pi(t)$ is basically a box centered around $t = 0$ and has a base and height of 1. $\Lambda(t)$ is a triangle centered around $t = 0$ and has a base of 2 and height of 1. You will encounter this notation in many text books.

In this problem we are given $y(t), x(t)$ and are asked to find $h(t)$ such that $y(t) = x(t) * h(t)$. The first thing to notice is that $x(t) = \Pi(t - \frac{3}{2})$. Second, from our experience in doing graphical convolutions with box functions, we know that $h(t)$ has to be a rectangle of the form $a\Pi(\frac{t-t_0}{b})$. Therefore, we need to determine its length b and height a and the center point on the t -axis t_0 . We also know that the length of the interval in which $y(t) > 0$ cannot exceed the sum of the two intervals in which $x(t)$ and $h(t)$ are nonzero respectively. Therefore, $b = 8 - 1 = 7$. The maximum value of $y(t)$ (happens to be 3) is the area of the box $x(t)$ (happens to be 1) scaled by a . Therefore, $a = 3$. Also, notice that $y(t)$ is centered around $t = 0$, while $x(t)$ is delayed by $\tau = \frac{3}{2}$. Therefore, in order to compensate of the delay of $x(t)$, $h(t)$ has to be delayed by $t_0 = -3/2$.

$$\Rightarrow h(t) = 3\Pi(\frac{t + \frac{3}{2}}{7})$$

We can also approach this problem graphically, by going back to the definition of convolution $y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau$. We first flip one of the signals ($x(t)$ in this case, since we are trying to determine $h(t)$). Figure 3 shows a flipped version of $x(t)$. How do we position $h(t)$ in order to maximize the value of the convolution at $t = 0$? we need complete overlap between $x(-\tau)$ and $h(\tau)$. Since $y(t)$ is symmetric around the axis, we know that if we drag $x(-\tau)$ in either direction along the axis by the same amount, we should get the same result. Therefore, $h(\tau)$ has to be symmetric around the axis of $x(-\tau)$, which is why we get the above result.

(d) Let $z(t) = \Pi(t) * \Lambda(t)$. Clearly, $z(t)$ has its maximum value at $t = 0$. This maximum value is the area of overlap (the sum of the two trapezoids). Therefore, $z_{max} = 0.75$.

$$y(t) = x(t) * h(t) = \Pi(t - \frac{3}{2}) * \Lambda(t + 1) = \Pi(t) * \delta(t - \frac{3}{2}) * \Lambda(t) * \delta(t + 1)$$

$$\Rightarrow y(t) = \Pi(t) * \Lambda(t) * \delta(t - \frac{3}{2}) * \delta(t + 1) = z(t) * \delta(t - \frac{1}{2})$$

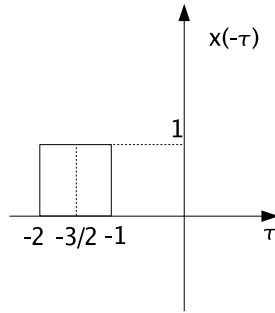


Figure 3: $x(-t)$ the time reversed (flipped) version of $x(t)$ in problems 2cd

Therefore, as we can see $y(t)$ is just a delayed version of $z(t)$. Therefore, $y_{max} = z_{max} = 0.75, y(\frac{1}{2}) = z(0)$.

Once again we can also look at this problem graphically. Going back to the definition of convolution $y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau$. In order to perform this convolution graphically, we first flip one of the signals (doesn't matter which one) along the t-axis. Figure 3 shows a flipped version of $x(t)$. The next step is the *drag* step. How far do we have to drag $x(-\tau)$ before its center completely overlaps with the center of $h(\tau)$? when they overlap, what is the area?

Problem 3 (*Convolution.*)

- 2.21(a)

$$x[n] = \alpha^n u[n]$$

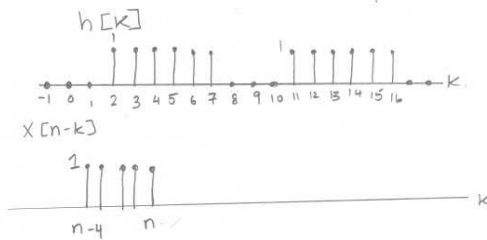
$$h[n] = \beta^n u[n]$$

$$\alpha \neq \beta$$

$$\begin{aligned} y[n] &= (x \star h)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n - k] \\ &= \sum_{k=-\infty}^{\infty} \alpha^k u[k] \beta^{n-k} u[n - k] = \sum_{k=0}^{\infty} \alpha^k \beta^{n-k} u[n - k] \\ &= \beta^n \cdot \sum_{k=0}^{\infty} \left(\frac{\alpha}{\beta}\right)^k u[n - k] = \beta^n u[n] \cdot \sum_{k=0}^n \left(\frac{\alpha}{\beta}\right)^k \\ &= \beta^n u[n] \frac{1 - \left(\frac{\alpha}{\beta}\right)^{n+1}}{1 - \left(\frac{\alpha}{\beta}\right)} = \frac{\beta^{n+1} - \alpha^{n+1}}{\beta - \alpha} u[n] \end{aligned}$$

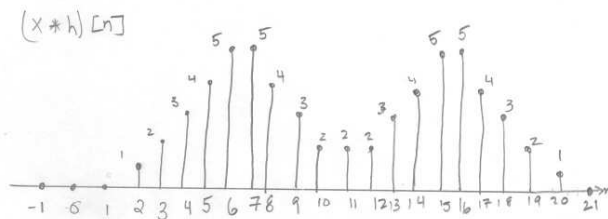
- 2.21(d)

d) OWN 2.21d $(x * h)[n] = \sum_{k=-\infty}^{\infty} h[k] x[n-k]$



$(x * h)[n] = \#$ of overlapping non-zero samples.

$$= \begin{cases} 0, & n \leq 1 \\ n-1, & 2 \leq n \leq 6 \\ 12-n, & 7 \leq n \leq 10 \\ 2, & n = 11 \\ n-10, & 12 \leq n \leq 15 \\ 21-n, & 16 \leq n \leq 21 \\ 0, & \text{else} \end{cases}$$



• 2.22(a)

$$x(t) = e^{-\alpha t} u(t)$$

$$h(t) = e^{-\beta t} u(t)$$

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau = \int_{-\infty}^{\infty} e^{-\alpha \tau} u(\tau) e^{-\beta(t - \tau)} u(t - \tau) d\tau \\ &= \begin{cases} \int_0^t e^{-\alpha \tau} e^{-\beta(t - \tau)} d\tau & t \geq 0 \\ 0 & t < 0 \end{cases} \end{aligned}$$

If $\alpha \neq \beta$, then:

$$y(t) = e^{-\beta t} \int_0^t e^{-(\alpha - \beta)\tau} d\tau \quad u(t) = e^{-\beta t} \frac{e^{-(\alpha - \beta)t} - 1}{\beta - \alpha} \quad u(t)$$

If $\alpha = \beta$, then:

$$y(t) = e^{-\beta t} \int_0^t 1 d\tau \quad u(t) = t e^{-\beta t} u(t)$$

See plot below.

• 2.22(e)

First, we observe that $h(t)$ can be written as:

$$h(t) = \begin{cases} 1-t & 0 \leq t \leq 1 \\ 0 & \text{else} \end{cases}$$

Because $x(t)$ is periodic, $y(t)$ will also be periodic. We will compute $y(t)$ for one period.

For $-\frac{1}{2} < t < \frac{1}{2}$, we have

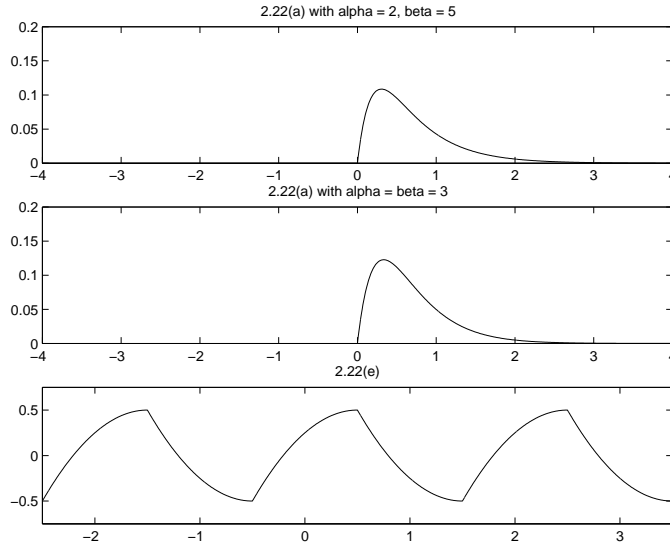
$$\begin{aligned} y(t) &= x(t) \star h(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \\ &= \int_0^{t+1/2} (1-\tau)d\tau + \int_{t+1/2}^1 (-1)(1-\tau)d\tau \\ &= \left[\tau - \frac{1}{2}\tau^2\right]_{\tau=0}^{t+1/2} + \left[\frac{1}{2}\tau^2 - \tau\right]_{\tau=t+1/2}^1 \\ &= \left((t + \frac{1}{2}) - \frac{1}{2}(t + \frac{1}{2})^2\right) + \left(\frac{1}{2} - 1\right) - \left(\frac{1}{2}(t + \frac{1}{2})^2 - (t + \frac{1}{2})\right) \\ &= \frac{1}{4} + t - t^2 \end{aligned}$$

For $\frac{1}{2} < t < \frac{3}{2}$, we have

$$\begin{aligned} y(t) &= \int_0^{t-1/2} (-1)(1-\tau)d\tau + \int_{t-1/2}^1 (1-\tau)d\tau \\ &= \left[\frac{1}{2}\tau^2 - \tau\right]_{\tau=0}^{t-1/2} + \left[\tau - \frac{1}{2}\tau^2\right]_{\tau=t-1/2}^1 \\ &= \left(\frac{1}{2}(t - \frac{1}{2})^2\right) + \left(1 - \frac{1}{2}\right) - \left((t - \frac{1}{2}) - \frac{1}{2}(t - \frac{1}{2})^2\right) \\ &= t^2 - 3t + \frac{7}{4} \end{aligned}$$

$y(t)$ will have a period of 2.

The plots of the solutions to problems 2.22(a) and (e) are shown in the following figure.



Problem 4 (*Impulse response and system properties.*)

- 2.29(d)

Not causal, because $h(-2) = e^{-4} \neq 0$

Stable, because $\int_{-\infty}^{\infty} |h(t)|dt = \int_{-\infty}^{-1} e^{2t}dt = \frac{1}{2}e^{-2t} < \infty$

- 2.29(g)

Causal, because $h(t) = 0$ for all $t < 0$

Unstable, because $\int_{-\infty}^{\infty} |h(t)| dt = \infty$

(Observe that $-e^{(t-100)/100}$ approaches $-\infty$ as t goes to ∞)

Problem 5 (*Noise suppression system for airplanes.*)

- (a)

Assume that the two delay elements both store the value 0 at the beginning of time.

Linearity First, assume that an input signal $x_1[n]$ produces an output signal

$$y_1[n] = \frac{2}{3}x_1[n] + \frac{1}{3}x_1[n-1] + \frac{1}{3}x_1[n-2]$$

Similarly, assume that an input signal; $x_2[n]$ produces an output signal

$$y_2[n] = \frac{2}{3}x_2[n] + \frac{1}{3}x_2[n-1] + \frac{1}{3}x_2[n-2]$$

If we use the signal $x[n] = \alpha x_1[n]$ as the input to the system, the output will be

$$y[n] = \frac{2}{3}\alpha x_1[n] + \frac{1}{3}\alpha x_1[n-1] + \frac{1}{3}\alpha x_1[n-2] = \alpha y_1[n]$$

If we use the signal $x[n] = x_1[n] + x_2[n]$ as the input to the system, the output will be

$$y_1[n] = \frac{2}{3}(x_1[n] + x_2[n]) + \frac{1}{3}(x_1[n-1] + x_2[n-1]) + \frac{1}{3}(x_1[n-2] + x_2[n-2]) = y_1[n] + y_2[n]$$

Therefore, the system is linear.

Time-invariance Again assume that an input signal $x_1[n]$ produces an output signal

$$y_1[n] = \frac{2}{3}x_1[n] + \frac{1}{3}x_1[n-1] + \frac{1}{3}x_1[n-2]$$

If we use the signal $x[n] = x_1[n - n_0]$ as the input to the system, the output will be

$$y[n] = \frac{2}{3}x_1[n - n_0] + \frac{1}{3}x_1[n - n_0 - 1] + \frac{1}{3}x_1[n - n_0 - 2] = y_1[n - n_0]$$

Therefore, the system is time-invariant.

Impulse response We can see from the figure that the impulse response of the system is

$$h[n] = \frac{2}{3}\delta[n] + \frac{1}{3}\delta[n-1] + \frac{1}{3}\delta[n-2]$$

- (b)

Causality Yes, the system is causal, because the current output $y[n]$ depends only on current and past inputs.

Memorylessness No, the system is not memoryless, because $y[n]$ depends on $x[n-1]$ and $x[n-2]$.

Stability Yes, the system is stable, because

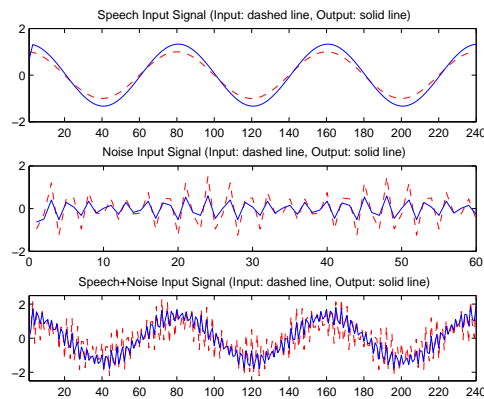
$$\sum_{n=-\infty}^{\infty} |h[n]| = \frac{4}{3} < \infty$$

Stability is important because we want all bounded input signals to produce bounded outputs. If we attempted to implement a noise suppression system for which a bounded input produced an unbounded output, then speaking into the system could cause it to break, short circuit, catch fire, etc.

- (c)

The following figures show the input and output signals for the three cases. We see that the system significantly attenuates the noise. When the input to the system is the combined signal, the output is much less noisy than the input. The following Matlab script was used to generate the figures.

```
h = [2/3 1/3 1/3];
index = 1:1:240;
noise_input = cos(3*pi/4*index) + 0.5*cos(2*pi/3*index);
speech_input = cos(pi/40*index);
total_input = cos(3*pi/4*index) + 0.5*cos(2*pi/3*index)+cos(pi/40*index);
noise_output = conv(noise_input,h);
speech_output = conv(speech_input,h);
total_output = conv(total_input,h);
subplot(3,1,1),plot(index,speech_input,'--r',index,speech_output(1:240),'-b')
axis([1 240 -2 2])
title('Speech Input Signal (Input: dashed line, Output: solid line)')
subplot(3,1,2),plot(index(1:60),noise_input(1:60),'--r',index(1:60),noise_output(1:60),'-b')
title('Noise Input Signal (Input: dashed line, Output: solid line)')
subplot(3,1,3),plot(index,total_input,'--r',index,total_output(1:240),'-b')
axis([1 240 -2.5 2.5])
title('Speech+Noise Input Signal (Input: dashed line, Output: solid line)')
```



- (d)

We can use Matlab to compute the SNR. At the input, the SNR is 0.8000. At the output, the SNR is 7.7331. We see that the filter has increased the SNR dramatically.

```
signal_input_power = (sum(speech_input.^2))/240;
noise_input_power = (sum(noise_input.^2))/240;
snr_input = signal_input_power/noise_input_power;
signal_output_power = (sum((speech_output(1:240)).^2))/240;
noise_output_power = (sum((noise_output(1:240)).^2))/240;
snr_output = signal_output_power/noise_output_power;
```

Problem 6 (Matlab, vector bases)

- (a)

$$\|\vec{v}_i\| = \sqrt{\vec{v}_i^T \vec{v}_i} = \sqrt{\vec{v}_i \cdot \vec{v}_i}$$

$$\theta_{ij} = \arccos((\vec{v}_i \cdot \vec{v}_j) / (\|\vec{v}_i\| \|\vec{v}_j\|))$$

Using the Matlab script shown below, we compute that:

$$\vec{v}_i \cdot \vec{v}_j = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

We find that the norm of each of the four vectors is equal to 1.0000, and that the angle between each pair of unique vectors is $\frac{\pi}{2}$. These four vectors form an orthonormal basis set.

```
load('basis.mat');
dot(v1,v1)
dot(v2,v2)
dot(v3,v3)
dot(v4,v4)
dot(v1,v2)
dot(v1,v3)
dot(v1,v4)
dot(v2,v3)
dot(v2,v4)
dot(v3,v4)
acos(dot(v1,v2))
acos(dot(v1,v3))
acos(dot(v1,v4))
acos(dot(v2,v3))
acos(dot(v2,v4))
acos(dot(v3,v4))
```

- (b)

$$\begin{aligned} \vec{y}^T \vec{v}_1 &= \alpha_1 \vec{v}_1^T \vec{v}_1 + \alpha_2 \vec{v}_2^T \vec{v}_1 + \alpha_3 \vec{v}_3^T \vec{v}_1 + \alpha_4 \vec{v}_4^T \vec{v}_1 \\ &= \alpha_1 * 1 + 0 + 0 + 0 \\ &= \alpha_1 \\ \vec{y}^T \vec{v}_2 &= \alpha_2 \\ \vec{y}^T \vec{v}_3 &= \alpha_3 \\ \vec{y}^T \vec{v}_4 &= \alpha_4 \end{aligned}$$

- (c)

```
x1_alpha1 = dot(x1,v1);
x1_alpha2 = dot(x1,v2);
x1_alpha3 = dot(x1,v3);
x1_alpha4 = dot(x1,v4);
x2_alpha1 = dot(x2,v1);
x2_alpha2 = dot(x2,v2);
x2_alpha3 = dot(x2,v3);
x2_alpha4 = dot(x2,v4);
x3_alpha1 = dot(x3,v1);
x3_alpha2 = dot(x3,v2);
x3_alpha3 = dot(x3,v3);
x3_alpha4 = dot(x3,v4);
```

$$\begin{aligned} \vec{x}_1 &= 5.2630 \vec{v}_1 + 0.0367 \vec{v}_2 + 1.3194 \vec{v}_3 - 4.6446 \vec{v}_4 \\ \vec{x}_2 &= 1.5750 \vec{v}_1 - 1.9564 \vec{v}_2 + 3.9976 \vec{v}_3 + 4.2687 \vec{v}_4 \\ \vec{x}_3 &= 0.2789 \vec{v}_1 + 1.2448 \vec{v}_2 - 1.7953 \vec{v}_3 - 3.7327 \vec{v}_4 \end{aligned}$$

Problem 7

Let us use year 2001 as our reference year (i.e. $n = 0$).

$$y[n] = 1.06y[n-1] + x[n]$$

$$x[n] = 1500u[n]$$

$$y[n] = 0 \quad \forall n < 0$$

$x[n]$ is the input to the system (a scaled step function). $y[n]$ is the output (account balance). In order to find the impulse response $h[n]$, we solve the difference equation when the input is $\delta[n]$.

$$h[n] = 1.06h[n-1] + \delta[n]$$

$$\Rightarrow h[0] = 1.06h[-1] + \delta[0] = 1$$

$$h[1] = 1.06h[0] + \delta[1] = 1.06$$

$$h[2] = 1.06h[1] + \delta[2] = 1.06^2$$

$$h[3] = 1.06h[2] = 1.06^3$$

$$\Rightarrow h[n] = 1.06^n \forall n \geq 0$$

$$\Rightarrow h[n] = 1.06^n u[n]$$

As we can see from the impulse response, the system is causal, but NOT stable or memoryless. Although not required, we can compute the output:

$$y[n] = x[n] * h[n] = 1500 \sum_{k=-\infty}^n h[k] = 1500u[n] \sum_{k=0}^n 1.06^k$$

$$y[n] = 1500u[n] \frac{1 - 1.06^{n+1}}{1 - 1.06} = 25000(1.06^{n+1} - 1)u[n]$$

Optional Problem

a) The difference equation is given by

$$S[n+1] = S[n] + X[n] - Y[n] \tag{1}$$

The reasoning is easy : If $X[n]$ enters the box, and $Y[n]$ leaves, the state increases by $X[n] - Y[n]$.

b) The equation becomes

$$S[n+1] = S[n] + S[n-1] \tag{2}$$

which is the Fibonacci sequence. Dividing both sides by $S[n]$, and assuming in the limit, $\frac{S[n+1]}{S[n]} = \frac{S[n]}{S[n-1]} = x$ (say), we get a quadratic equation in x . The positive solution is the golden ratio : $\frac{1+\sqrt{5}}{2}$.