CENG 384 - Signals and Systems for Computer Engineers
Spring 2023
Homework 2

Şamlıoğlu, Erdem
e2448843@ceng.metu.edu.tr

Uludoğan, Mert
e2380996@ceng.metu.edu.tr

January 2, 2025

---

1. (a) $y^{'}(t) = x(t) - 5y(t)$

   (b) Our input is $x(t) = (e^{-t} + e^{-3t})u(t)$ and the system is initially at rest. To find the output $y(t)$ we can use the formula $y(t) = y_h(t) + y_p(t)$. For the homogenous part, we can use the formula: $y_h(t) = Ke^at$ From the part a, $y^{'} + 5y(t) = 0$, we can derive $(a + 5)Ke^at = 0$. So $a = -5$, and our $y_h(t) = Ce^{-5t}$.
   For the particular part, $y_p(t) = Ae^{-t} + Be^{-3t}$. From part a we know, $y^{'}(t) + 5y(t) = x(t)$. So we can calculate $y_p^{'}$ which is equal to $-Ae^{-t} + -3Be^{-3t}$. Putting the values we have found in the formula, we get:
   $-Ae^{-t} + -3Be^{-3t} + 5Ae^{-t} + 5Be^{-3t} = e^{-t} + e^{-3t}$. simplifying the equation:
   $4Ae^{-t} + 2Be^{-3t} = e^{-t} + e^{-3t}$. We can see that $4A = 1$ and $2B = 1$. So accordingly; $A = \frac{1}{4}$, $B = \frac{1}{2}$. Now we can put the values and find:
   $y_p(t) = \frac{1}{4}e^{-t} + \frac{1}{2}e^{-3t}$. so $y(t) = Ce^{-5t} + \frac{1}{4}e^{-t} + \frac{1}{2}e^{-3t}$. To find the value C, we know that the system is initially at rest, which means $y(0) = y^{'}(0) = y^{''}(0) = ... = 0$ For t=0, $y(0) = C + \frac{1}{4} + \frac{1}{2} = 0$. So C is equal to $\frac{-3}{4}$. Putting C in the equation we find finally:
   $y(t) = \frac{-3}{4}e^{-5t} + \frac{1}{4}e^{-t} + \frac{1}{2}e^{-3t}$

2. (a)
   $x[n] = 2\delta[n] + \delta[n+1]$
   Thanks to the "distributivity" property of Convolution, we can seperate the impulse function to use one-term impulse response functions.
   $h[n] = \delta[n-1] + 2\delta[n+1]$
   $h_0[n] = \delta[n-1]$, $h_1[n] = 2\delta[n+1]$
   for $h_0[n]$:
   $y_0[n] = x[n] * h_0[n] = 2\delta[n-1] + \delta[n]$, h_0 is clearly shift the current input
   for $h_1[n]$:
   $y_1[n] = x[n] * h_1[n] = 4\delta[n+1] + 2\delta[n+2]$
   $y[n] = y_0[n] + y_1[n]$
   $= 2\delta[n-1] + \delta[n] + 4\delta[n+1] + 2\delta[n+2]$

   (b)
   Firstly, let's take the derivative of $x(t)$, where $x(t) = u(t-1) + u(t+1)$
   $\frac{dx(t)}{t} = \delta(t-1) + \delta(t+1)$
   $h(t) = e^{-t} \cdot sin(t) \cdot u(t)$
   Using "commutative" property, we can change the position of $\frac{dx(t)}{t}$ and h(t) in convolution function:
   Let's say, $w(t) = e^{-t} \cdot sin(t) \cdot u(t)$ and $h(t) = \delta(t-1) + \delta(t+1)$
   Now, using "distributivity" property, $h_0(t) = \delta(t-1)$ and $h_1(t) = \delta(t+1)$
   $y_0(t) = w(t) * h_0(t) = e^{-t+1} \cdot sin(t-1) \cdot u(t-1)$
   $y_1(t) = w(t) * h_1(t) = e^{-t-1} \cdot sin(t+1) \cdot u(t+1)$
   $y(t) = y_0 + y_1$

3. (a) To find $y(t) = x(t)*h(t)$, we calculate: $\int_{\infty}^{-\infty} x(\tau)h(t-\tau)d\tau$. For the given values in part a, we get : $\int_0^t e^{-\tau}e^{-2t+2\tau}d\tau$ which is equal to $\int_0^t e^{-2t+\tau}d\tau$. Taking the constant e^{-2t} out, we get $e^{-2t}\int_0^t e^{\tau}d\tau$, Using the common integral and computing the boundaries, finally we get: $e^{-2t}(1 - e^t)u(t)$

   (b) To find $y(t) = x(t) * h(t)$ we need to calculate different regions for x(t) because it behaves differently for different values of t.

For $0 < t < 1$, we need to calculate: $\int_0^t e^{3t-3\tau}d\tau$. Taking the e^{3t} constant out, $e^{3t}\int_0^t e^{-3\tau}d\tau$. Applying u substitution and using common integral we get: $\frac{-1}{3}e^{3t}(e^{-3t}-1)(u(t)-u(t-1))$
For $t > 1$, we need to calculate $\int_0^1 e^{3t-3\tau}d\tau$. Taking the constant e^{3t} constant out, $e^{3t}\int_0^1 e^{-3\tau}d\tau$. Applying u substitution and using common integral we get: $\frac{(e^3-1)e^{3t-3}}{3}u(t-1)$

4. (a)
$y[n]-y[n-1]-y[n-2]=0$, $y[0]=1$ and $y[1]=1$
Firstly, we can write characteristic equation for this recurrence relation,
which is $r^2-r-1=0$
$r_1 = \dfrac{(1+\sqrt{5})}{2}$ and $r_2 = \dfrac{(1-\sqrt{5})}{2}$
so the general solution: $y[n] = A\cdot r_1^n + B\cdot r_2^n$
A and B can be found from the initial conditions.
$y[0]=1$ implies $A+B=1$
$y[1]=1$ implies $r_1\cdot A + r_2\cdot B = 1$
as results of calculations:
$A = \dfrac{(1+\sqrt{5})}{2\sqrt{5}}$
$B = \dfrac{(1-\sqrt{5})}{2\sqrt{5}}$
$y[n] = \dfrac{1}{\sqrt{5}}\cdot\Big((\dfrac{1+\sqrt{5}}{2})^n + (\dfrac{1-\sqrt{5}}{2})^n\Big)$

(b)
$y^(3)(t)-6y^{(2)}+13y^{(1)}-10y+0$
This is an homogenous differential
$(\alpha^3-\alpha^2+13\alpha-10)\cdot C\cdot e^{\alpha t}=0$
$(\alpha^3-\alpha^2+13\alpha-10)=0$
$r_1=2, r_2=(2+i), r_3=(2-i)$
$y = C_1e^{2t}+C_2e^{2t}cost+C_3e^{2t}sint$
$y(0)=1, y^{(1)}(0)=\dfrac{3}{2}, y^{(2)}(0)=3$
As a result of computations:
$y = 2e^2t - e^{2t}cost - \dfrac{1}{2}e^{2t}sint$

5. (a) To find the particular solution for $x(t)=cos(5t)$, guess a solution: $y(t)=Acos(5t)+Bsin(5t)$. Now we should find the first and second derivatives accordingly:
$y^{'}(t) = -5Asin(5t)+5Bcos(5t)$
$y^{''}(t) = -25Acos(5t)-25Bsin(5t)$
Putting what we have found in the equation: $(-25Acos(5t)-25Bsin(5t))+5(-5Asin(5t)+5Bcos(5t))+6(Acos(5t)+Bsin(5t)) = cos(5t)$
Now we can find $-25A+25B+6A=1$ and $-25B-25A+6B=0$. $A=\frac{-19}{986}$. $B=\frac{25}{986}$
$y(t) = \frac{-19}{986}cos(5t)+\frac{25}{986}sin(5t)$
(b) To find the homogeneous solution, we can use the characteristic equation which is:
r^2+5r+6 for this function. We can see from this equation, roots are $r_1=-2$, and $r_2=-3$.
So accordingly we can find $y_h(t) = Ce^{-2t}+De^{-3t}$ for some constants C,D.
(c) For general solution, we add up part a and b, we get :
$y(t) = Ce^{-2t}+De^{-3t}+\frac{-19}{986}cos(5t)+\frac{25}{986}sin(5t)$. and we also find it's derivative:
$y^{'}(t) = (-2Ce^{-2t}-3De^{-3t})+(-5\frac{-19}{986}sin(5t)+5\frac{25}{986}cos(5t))$ Since it's initially at rest. We can say that $y(0)=0$ and $y^{'}(0)=0$. So putting the values, $y(0)=C+D+\frac{-19}{986}=0$, $C+D=\frac{19}{986}$.
$y^{'}(0) = -2C-3D+\frac{125}{986}=0$, $2C+3D=\frac{125}{986}$
So we can conclude as, $C=\frac{-70}{986}$, $D=\frac{89}{986}$. applying it to the general solution we get:
$y(t) = \frac{-70}{986}e^{-2t}+\frac{89}{986}e^{-3t}+\frac{-19}{986}cos(5t)+\frac{25}{986}sin(5t)$

6. (a)
$w[n]-\frac{1}{2}w[n-1]=x[n]$

$2w[n]-w[n-1]=2x[n]$
$2w[n-1]-w[n-2]=2x[n-1]$
$2w[n-2]-w[n-3]=2x[n-2]$
...

(Multiplying and summing to eliminate left hand side except $w[n]$)

$2^n \cdot w[n] - w[0] = \sum_{k=0}^{n-1} x[n-k] \cdot 2^{n-k}$

$w[n] - w[0] = \sum_{k=0}^{n-1} x[n-k] \cdot 2^{-k}$

$w[0] = 0$ because the system is initially rest

$h_0[n] = w[n] = \sum_{k=0}^{n-1} \delta[n-k] \cdot 2^{-k}$

$h_0[n] = 2^{-n}u[n]$

(b)

Overall impulse is the convolution of the serial models of the system

$h[n] = h_0[n] * h_0[n]$

$\sum_{k=-\infty}^{\infty} 2^{-k}u[k] \cdot 2^{k-n}u[n-k]$

$\sum_{k=0}^{\infty} 2^{-k} \cdot 2^{k-n}u[n-k]$

$\sum_{k=0}^{\infty} 2^{-n}u[n-k]$

(c)

7.

```
import numpy as np
import matplotlib.pyplot as plt


input_data = np.genfromtxt('hw2_signal.csv', delimiter=',')
start_index = int(input_data[0])
x = input_data[1:]


h = np.zeros_like(x)
h[start_index - 5] = 1

# Perform discrete convolution of x[n] and h[n]
y = np.zeros_like(x)
for n in range(len(x)):
    for k in range(len(h)):
        if n - k >= 0:
            y[n] += x[k] * h[n - k]


n = np.arange(len(x))
plt.stem(n, x, linefmt='b-', markerfmt='bo', label='Input Signal x[n]')
plt.stem(n, y, linefmt='r-', markerfmt='ro', label='Output Signal y[n]')
plt.xlabel('n')
plt.ylabel('Amplitude')
plt.legend()
plt.show()
```

(a)

```
import matplotlib.pyplot as plt
import numpy as np

def convolution(x, h):

    M = len(x)
    N = len(h)

    x_padded = np.pad(x, (0, N - 1), mode='constant')
    h_padded = np.pad(h, (0, M - 1), mode='constant')

    # Initialize the output array
    y = np.zeros(M + N - 1)
```

```
    # Perform convolution
    for n in range(M + N - 1):
        for k in range(N):
            if n - k < 0 or n - k >= M:
                continue
            y[n] += x_padded[n - k] * h_padded[k]

    print(type(y))
    return y

def draw_seven_b(list, start_index, N):

    output_values = list
    # Create a time axis
    n = np.arange(start_index, start_index + output_values.size, 1)

    # Plot the signal
    plt.stem(n, output_values)
    plt.xlabel('Time (n)')
    plt.ylabel('Amplitude')
    plt.title('Discrete Signal')
    plt.suptitle(f"N = {N}")
    plt.show()



def seven_b():
    with open("hw2_signal.csv") as csv_file:
        x = csv_file.readline().strip()
        _x = x.split(',')
        almost_data = list(map(float, _x))
        start_index = int(almost_data[0])
        data_xn = almost_data[1:]

        # draw_seven_b(np.array(data_xn), start_index, 33)

        for N in [3,5,10,20]:
            start_index_output = 0 - (N - 1)
            data_hn = [1/N]*(N)
            # Define the input arrays
            graph = data_xn
            function = data_hn

            result = convolution(graph, function)

            draw_seven_b(result, start_index_output, N)

seven_b()
```