Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 1, Part D**

Chart, scatter chart

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 1, Part D

import numpy as np

import matplotlib.pyplot as plt

# Nonzero portions of h and x

h = [3, 2, 1];

x = [0, 1, -2, 2, -1, 0];

# Bounds that equal zero

# These are known from Problem 1, Part C

n = np.arange(-2,6);

y = np.convolve(x,h);

fig, ax = plt.subplots(1, figsize=(12,3))

ax.stem(n, y)

ax.set\_ylim([-4, 4])

ax.set\_title("Output from the NumPy function convolve")

ax.set\_xlabel("n")

ax.set\_ylabel("conv(x,h)")

ax.grid(True)

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 2, Part E**

Chart, line chart

Description automatically generated

Chart

Description automatically generated

**(Not sure if phase is needed since the equation isn’t complex, so I just plotted it anyways)**

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 2, Part E

import numpy as np

import matplotlib.pyplot as plt

r1 = -1/4

r2 = 1/2

c1 = -1/24

c2 = -1/12

nmin = -1;

nmax = 12;

n = np.arange(nmin, nmax + 1)

h = np.zeros(np.size(n))

zeroIdx = -nmin

for m in np.arange(0, nmax):

if m < 1:

h[zeroIdx + m] = -2.0\*(c1\*r1\*\*m + c2\*r2\*\*m)

elif m < 2:

h[zeroIdx + m] = -2.0\*(c1\*r1\*\*m + c2\*r2\*\*m) \

+ 3.0\*(c1\*r1\*\*(m-1) + c2\*r2\*\*(m-1))

else:

h[zeroIdx + m] = -2.0\*(c1\*r1\*\*m + c2\*r2\*\*m) \

+ 3.0\*(c1\*r1\*\*(m-1) + c2\*r2\*\*(m-1))

phaseh = np.angle(h)

fig, ax = plt.subplots(1, figsize=(12,3))

ax.stem(n, h)

ax.set\_title("Impulse Response h[n]")

ax.set\_xlabel("Time n")

ax.set\_ylabel("{h[n]}")

plt.show()

fig, ax = plt.subplots(1, figsize=(12,3))

ax.stem(n, phaseh)

ax.set\_title("The Phase of the Impulse Response h[n]")

ax.set\_xlabel("Time n")

ax.set\_ylabel("Phase {h[n]}")

plt.show()

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 3, Part B**

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 3, Part B

import numpy as np

import matplotlib.pyplot as plt

Omega = np.linspace(0, np.pi, 1024)

z = np.exp(1j\*Omega)

oneVec = np.ones(np.size(z))

H = np.divide(z \* (-2.0\*z + 3.0\*oneVec), -8.0\*z\*z + 2.0\*z + 1.0\*oneVec)

magH = np.abs(H);

phaseH = np.angle(H);

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(Omega, magH)

ax.set\_title("Magnitude of the Frequency Response $H(e^{j\Omega})$")

ax.set\_ylabel("$|H(e^{j\Omega})|$")

ax.set\_xlabel("Angular frequency $\Omega$")

plt.show()

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(Omega, phaseH)

ax.set\_title("Phase of the Frequency Response $H(e^{j\Omega})$")

ax.set\_ylabel("Phase of $H(e^{j\Omega})$")

ax.set\_xlabel("Angular frequency $\Omega$")

plt.show()

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 3, Part C**

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 3, Part C

import numpy as np

import matplotlib.pyplot as plt

from scipy import signal

a = [-8.0, 2.0, 1.0]

b = [-2.0, 3.0]

w, H = signal.freqz(b, a, 1024)

magH = np.abs(H);

phaseH = np.angle(H);

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(w, magH)

ax.set\_title("Magnitude of the Frequency Response $H(e^{j\Omega})$")

ax.set\_ylabel("$|H(e^{j\Omega})|$")

ax.set\_xlabel("Angular frequency $\Omega$")

plt.show()

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(w, phaseH)

ax.set\_title("Phase of the Frequency Response $H(e^{j\Omega})$")

ax.set\_ylabel("Phase of $H(e^{j\Omega})$")

ax.set\_xlabel("Angular frequency $\Omega$")

plt.show()

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 4, Part E**

Chart, line chart

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 4, Part E

import numpy as np

import matplotlib.pyplot as plt

numpts = 2000

t = np.linspace(-2.0, 3.0, numpts)

y = np.zeros(np.size(t))

for m in np.arange(0, numpts+1):

if t[m] <= -1.0:

continue

elif t[m] <= 0.0:

y[m] = (pow(t[m], 4.0) / 12.0) - pow(t[m], 2.0) - ((2.0 \* t[m]) / 3.0) + (1.0 / 4.0)

elif t[m] <= 1.0:

y[m] = (1.0 / 4.0) - ((2.0 \* t[m]) / 3.0)

elif t[m] <= 2.0:

y[m] = (-(pow(t[m], 4.0) / 12.0)) + pow(t[m], 2.0) - ((4.0 \* t[m]) / 3.0)

else:

break

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(t,y)

ax.set\_xlabel("Time t")

ax.set\_ylabel("y(t)")

ax.set\_title("The output of the system y(t) = x(t)\*h(t)")

plt.show()

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 5, Part B**

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 5, Part B

import numpy as np

import matplotlib.pyplot as plt

omega = np.linspace(-10.0, 10.0, 2000)

jomega = 1j\*omega

omega2 = np.multiply(jomega, jomega)

oneVec = np.ones(np.size(omega))

H = np.divide(-3.0\*omega2 + 11.0\*oneVec, 2.0\*omega2 - 5.0\*jomega - 7.0\*oneVec)

magH = np.abs(H);

phaseH = np.angle(H);

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(omega, magH)

ax.set\_title("Magnitude of the Frequency Response $H(j\omega)$")

ax.set\_ylabel("$|H(j\omega)|$")

ax.set\_xlabel("Angular frequency $\omega$")

plt.show()

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(omega, phaseH)

ax.set\_title("Phase of the Frequency Response $H(j\omega)$")

ax.set\_ylabel("Phase of $H(j\omega)$")

ax.set\_xlabel("Angular frequency $\omega$")

plt.show()

Phil Nevins  
ECE 315 Signals and Systems  
11/8/2022  
  
**Problem 5, Part C**

Chart, line chart

Description automatically generated

Chart, line chart, histogram

Description automatically generated

#Phil Nevins

#ECE 315 Signals and Systems

#11/8/2022

#Problem 5, Part C

import numpy as np

import matplotlib.pyplot as plt

from scipy import signal

a = [2.0, -5.0, -7]

b = [-3.0, 0.0, 11.0]

H = signal.freqs(b, a, worN=np.linspace(-10.0, 10.0, 1024))

magH = np.abs(H);

phaseH = np.angle(H);

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(w, magH)

ax.set\_title("Magnitude of the Frequency Response $H(j\omega)$ from freqz")

ax.set\_ylabel("$|H(j\omega)|$")

ax.set\_xlabel("Angular frequency $\omega$")

plt.show()

fig, ax = plt.subplots(1, figsize=(12,3))

ax.plot(w, phaseH)

ax.set\_title("Phase of the Frequency Response $H(j\omega)$ from freqz")

ax.set\_ylabel("Phase of $H(j\omega)$")

ax.set\_xlabel("Angular frequency $\omega$")

plt.show()