Previous Exam solution

Michel Hardenberg

January 5, 2024

Problem 1

Periodicity

i) $x(t) = 3\cos(3t + \pi/3)$ is periodic as it is a simple sinusoid with $T = \frac{2\pi}{3}$.

ii) $x(t) = 3\cos^2(3t+\pi/3)$ is also periodic. To see this note that $y(t) = 3\cos(3t+\pi/3)$ is periodic as before. Then that $y(t+\frac{\pi}{3}) = -y(t)$. but since we take the square $x(t) = y^2(t) = y^2(t+\frac{\pi}{3})$ it now is periodic with half the period of $y(t)$, namely $T = \frac{\pi}{3}$.

iii) $x(t) = \cos(\frac{\pi}{2}t) + \cos(\frac{1}{2}t)$ is a sum of sinusoids with periods of $T_1 = 4$ and $T_2 = 4\pi$. $x(t)$ is periodic if there exist two integers n, m such that $n \cdot T_1 = m \cdot T_2$. This is not possible as the fraction of $n/m \propto \pi$, which is irrational. Hence, no such integer numbers exist.

iv) $x(t) = e^{j\pi t}$ is trivially periodic, as it is a simple complex exponential and thus by Eulers rule, a sum of sinusoids of equal periods $T = 2$.

System properties

$$y(t) = x(t)x(t+2). \tag{1}$$

memoryless? No, as the system depends on future values.

causal? No, as the system depends on future values.

stable? Yes. Let $x(t)$ be bounded by $|x(t)| < M < \infty$. Then $|y(t)| = |x(t) \cdot x(t+2)| \leq M^2 < \infty$. Hence, $y(t)$ is bounded by M^2 for M bounded input and thus stable.

TI Yes. To be time invarient we need

$$y(t+t_0) = x(t+t_0)x(t+2+t_0), \tag{2}$$

which can be seen by insertion.

linear No. Let $x(t) = a \cdot x_0(t) + b \cdot x_1(t)$. Then

$$\begin{aligned} x(t)x(t+2) &= (a \cdot x_0(t) + b \cdot x_1(t))(a \cdot x_0(t+2) + b \cdot x_1(t+2)) && (3) \\ &= a^2 \cdot x_0(t)x_0(t+2) + b^2 \cdot x_1(t)x_1(t+2) \\ &+ ab \cdot (x_0(t)x_1(t+2) + x_0(t+2)x_1(t)) && (4) \\ &\neq a \cdot \underbrace{x_0(t)x_0(t+2)}_{y(t)|_{x(t)=x_0(t)}} + b \cdot \underbrace{x_1(t)x_1(t+2)}_{y(t)|_{x(t)=x_1(t)}}. && (5) \end{aligned}$$

$$y[n] = max\{x[n], x[n+1]\}. \tag{6}$$

memoryless? No, as the system depends on future values.

causal? No, as the system depends on future values.

stable? Yes. Let $x(t)$ be bounded by M as before. Then by definition $y(t)$ is also bounded by M and thus it is stable.

TI? Yes. Any values we input will shift the output by the same. As we compare to the one-setp-ahead value, we will always cover the same pairings. Different integer jumps could produce different pairing depending on time delays of the imput and could be time variant.

Linear No. Generally[1]

$$\begin{aligned} & a \cdot max\{x_0[n], x_0[n+1]\} + b \cdot max\{x_1[n], x_1[n+1]\} \\ & \neq max\{ax_0[n] + bx_1[n], ax_0[n+1] + bx_1[n+1]\} && (7) \end{aligned}$$

Consider e.g.

$$\begin{aligned} x_0[n] &= [\dots, 1, 2, 1, 2, \cdots] && (8) \\ x_1[n] &= [\dots, 1, -1, 1, -1, \cdots] && (9) \\ \Rightarrow x_0[n] + x_1[n] &= [\dots, 2, 1, 2, 1, \dots] \text{ and} && (10) \\ a = b &= 1. && (11) \end{aligned}$$

Then

$$\begin{aligned} y_0[n] &= [\dots, 2, 2, 2, \dots] && (12) \\ y_1[n] &= [\dots, 1, 1, 1, \dots] && (13) \\ \Rightarrow y_0[n] + y_1[n] &= [\dots, 3, 3, 3, \dots], && (14) \end{aligned}$$

but

$$\begin{aligned} y_{x_0+x_1}[n] &= [\dots, 2, 2, 2, \dots] && (15) \\ &\neq y_0[n] + y_1[n]. && (16) \end{aligned}$$

[1] There are exceptions, but that's not the point.

Problem 2 - Fourier series

This exercise would usually require integration by parts, which we havn't used thorughout the course and will not appear in the final exam. I have sligtly changed the signal to make this problem more representative of the coming exam. Let

$$x(t) = \begin{cases} -1, & \text{if } 0 \leq t < 1 \\ 1, & \text{if } 1 \leq t < 2 \\ 0, & \text{if } 2 \leq t < 4 \end{cases} \tag{17}$$

and $T = 4$. For now, assume $k \neq 0$, then we know

$$\begin{aligned}
a_{k\neq 0} &= \frac{1}{T}\int_T dt \quad x(t)e^{-jk\frac{2\pi}{T}t} && (18)\\
&= \frac{1}{4}\left(\int_0^1 dt \quad -e^{-jk\frac{\pi}{2}t} + \int_1^2 dt \quad e^{-jk\frac{\pi}{2}t}\right) && (19)\\
&= \frac{1}{4}\left(\left[\frac{2}{jk\pi}e^{-jk\frac{\pi}{2}t}\right]_0^1 - \left[\frac{2}{jk\pi}e^{-jk\frac{\pi}{2}t}\right]_1^2\right) && (20)\\
&= \frac{1}{jk2\pi}\left(-e^{-jk\frac{\pi}{2}} + e^{-jk\frac{\pi}{2}} - e^{-jk\pi} - 1\right) && (21)\\
&= \frac{1}{jk2\pi}\left(2\cdot e^{-jk\frac{\pi}{2}} - (1 + e^{-jk\pi})\right) && (22)\\
&= \frac{1}{jk2\pi}\left(2\cdot e^{-jk\frac{\pi}{2}} - e^{-jk\pi/2}(e^{jk\pi/2} + e^{-jk\pi/2})\right) && (23)\\
&= \frac{e^{-jk\pi/2}}{jk2\pi}\left(2 - 2\cos(k\pi/2)\right) && (24)\\
&= \frac{e^{-jk\pi/2}}{jk\pi}\left(1 - \cos(k\pi/2)\right) && (25)
\end{aligned}$$

For $k = 0$

$$\begin{aligned}
a_0 &= \frac{1}{4}\int_T dt \quad x(t)e^0 && (26)\\
&= \frac{1}{4}\left(-\int_0^1 dt \quad + \int_1^2 dt\right) && (27)\\
&= 0. && (28)
\end{aligned}$$

AS the problem has been simplified, it would be reasonable to plot the synthesiyed function using different numbers of n and the orignal to check.

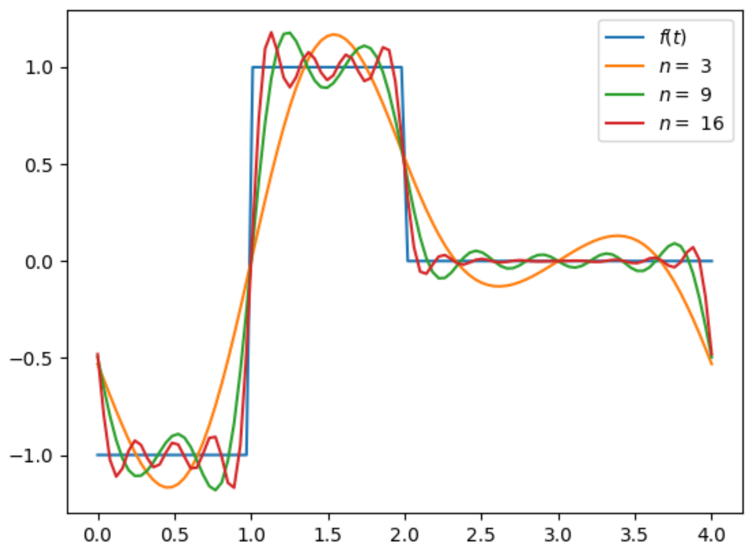

Figure 1: Orignal and FOurier series representation of $f(t)$.

Here is some code that does it.

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


# original function
def func(t):
    out = np.zeros_like(t)
    out[t < 2] = 1.
    out[t < 1] = -1.

    return out

# our a_k series
def ak_fn(k):
    if k == 0:
        return 0.
    factor = np.exp(-1j * k * np.pi / 2) / (1j * np.pi * k)
    return factor * (1 - np.cos(k*np.pi / 2))

# synthesis eqn. for c.t. signals
def synth(t, n, ak_fn, period):
    out = np.zeros_like(t)
    for k in range(-n, n):
        out += np.real(ak_fn(k)*np.exp(1j * k * (2*np.pi / period)
    * t))
    return out

# create samples of times and functions values
ts = np.linspace(0, 4, 100)
fs = func(ts)

# make approximations for ifferent values of N
synths = {}
for k in [3, 9, 16]:
    synths[k] = synth(ts, k, ak_fn, 4)

# plot
fig, ax = plt.subplots()
ax.plot(ts, fs, label=r"$f(t)$")
for key in synths:
    ax.plot(ts, synths[key], label=f"$n = $ {key}")

ax.legend()
plt.savefig("figures/p2_synth.png")
plt.show()
```

Problem 3 LTI

Frequency response

Let $\dot{y}(t) + 3y(t) = x(t)$ be some LTI system.

We can solve the difference equation by using the Fourier transform.

$$\dot{y}(t) + 3y(t) = x(t) \tag{29}$$
$$\Rightarrow j\omega Y(j\omega) + 3Y(j\omega) = X(j\omega). \tag{30}$$

We now have eliminated the differential and can simply rearange as such

$$Y(j\omega)(j\omega + 3) = X(j\omega) \tag{31}$$
$$\Rightarrow \frac{Y(j\omega)}{X(j\omega)} \equiv H(j\omega) = \frac{1}{j\omega + 3} \tag{32}$$

Bode plot

You could just compute a few values and then sketch it by hand. Otherwise, here is some code that does it.

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


# frequencies
ws = np.linspace(0, np.pi, 100)

# analytical frequency response
def f_rsp(w):
    return 1 / (1j * w + 3)

# convert to dB
def dB(s):
    return 20*np.log10(s)

# plot
fig, ax = plt.subplots()
ax.plot(ws, np.abs(f_rsp(ws)))
ax.set_title("Bode plot")
ax.set_xlabel(r"$\omega$")

plt.savefig("figures/p3_bode.png")
plt.show()
```

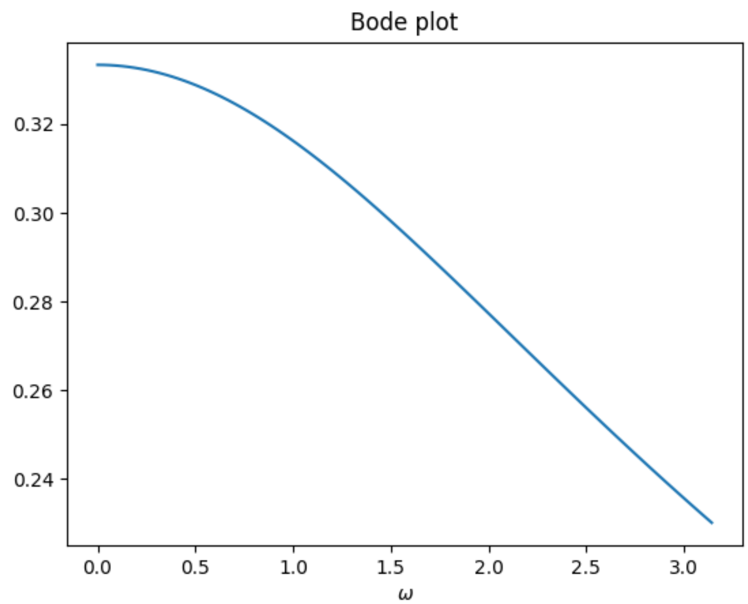

Figure 2: Bode plot of system.

System output

Let some input be defined as

$$x(t) = \begin{cases} e^{-t}, & \text{if } t \geq 0 \\ 0 \end{cases} \tag{33}$$

$$= e^{-t}u(t). \tag{34}$$

From earlier we know

$$H(j\omega) = \frac{1}{3 + j\omega} \tag{35}$$

Then using table 4.2 from the book:

$$\begin{align}
h(t) &= e^{-3t}u(t) \tag{36}\\
\Rightarrow y(t) &= h(t) * x(t) \tag{37}\\
&= \int_{\mathbb{R}} d\lambda \;\; h(\lambda)x(t-\lambda) \tag{38}\\
&= \int_{\mathbb{R}} d\lambda \;\; e^{-3\lambda}u(\lambda)e^{-(t-\lambda)}u(t-\lambda) \tag{39}\\
&= \int_0^{\infty} d\lambda \;\; e^{-3\lambda}e^{-(t-\lambda)} \quad \underbrace{u(t-\lambda)}_{t \text{ must be } \geq \lambda} \tag{40}\\
& \tag{41}
\end{align}$$

Note that $t \geq 0$, otherwise would this also give 0, as λ needs to be greater or equal to 0, and t must be greater than or equal to λ due to the step functions. Hence

$$\begin{align}
y(t) &= u(t)\int_0^t d\lambda \;\; e^{-3\lambda}e^{-(t-\lambda)} \tag{42}\\
&= u(t)\int_0^t d\lambda \;\; e^{-2\lambda-t} \tag{43}\\
&= u(t)e^{-t}\int_0^t d\lambda \;\; e^{-2\lambda} \tag{44}\\
&= -u(t)\left[\frac{e^{-2(t+\lambda)}}{2}\right]_0^t \tag{45}\\
&= \frac{e^{-t}-e^{-3t}}{2}u(t). \tag{46}
\end{align}$$

Problem 4 LTI convolution

Let

$$h[n] = 2(u[n] - u[n-5]), \tag{47}$$

be an impulse response of some system and

$$x[n] = u[n] - u[n-3] \tag{48}$$

the input.

Discrete case

Scetch both

```python
# imports
import numpy as np
import matplotlib.pyplot as plt


# define series
def h_fn(n):
    return 2*(np.heaviside(n, 1) - np.heaviside(n-5, 1))

def x_fn(n):
    return np.heaviside(n, 1) - np.heaviside(n-3, 1)

# define ns
ns = np.arange(-3,10)


#plot
fig, ax = plt.subplots()
ax.stem(ns, h_fn(ns), linefmt='tab:blue', label='h[n]')
ax.stem(ns, x_fn(ns), linefmt='tab:orange', label='x{n]')

ax.legend()
plt.savefig("figures/p4_stems.png")
plt.show()
```

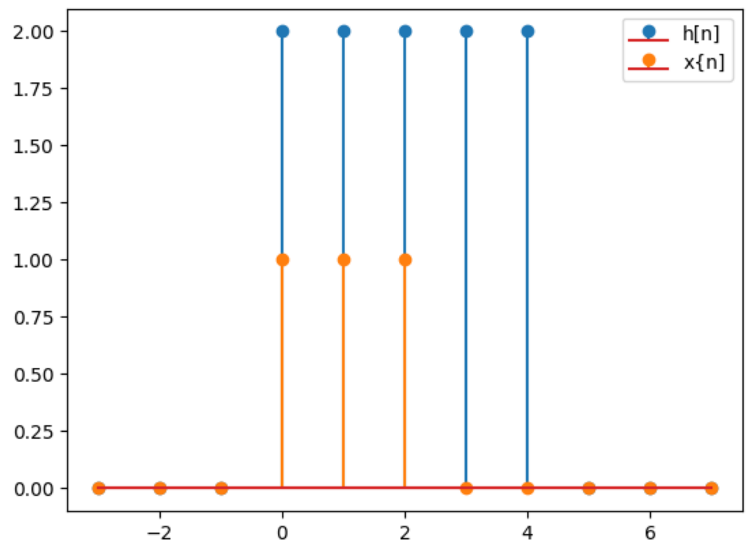

Figure 3: Input and impulse response.

Sketch the convolution

```
ys = np.convolve(xs, hs)
ks = np.arange(-(n + n), m + m - 1)

#plot
fig, ax = plt.subplots()
ax.stem(ks, ys, linefmt='tab:green', label='y[n]')
ax.set_xlim([-3, 10])
ax.legend()
plt.savefig("figures/p4_conv.png")
plt.show()
```

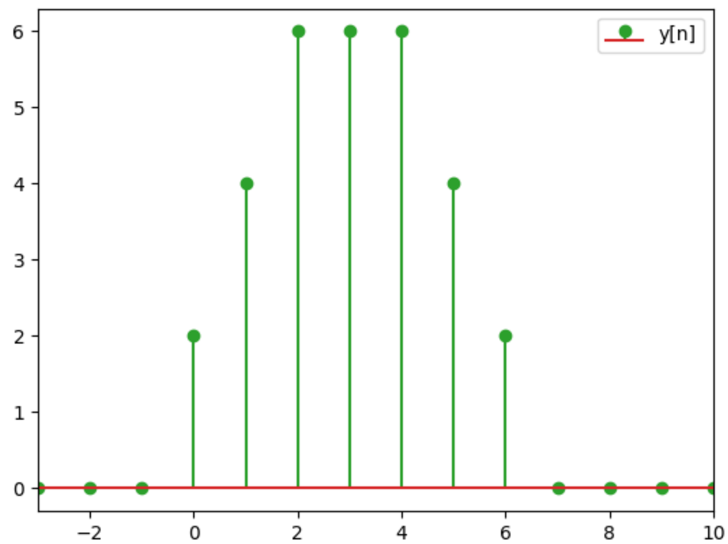

Figure 4: Output.

Continuous case

Let

$$h(t) = (u(t) - u(t)), \tag{49}$$

be an impulse response of some system and

$$x(t) = u(t) - u(t-3) \tag{50}$$

the input.

Scetch both

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


# define series
def h_fn(t):
    return np.heaviside(t, 1) - np.heaviside(t-5, 1)

```

```python
def x_fn(t):
    return np.heaviside(t, 1) - np.heaviside(t-3, 1)

# define ns
n, m = 3, 8
ts = np.linspace(-n, m, 100)
xs, hs = x_fn(ts), h_fn(ts)

#plot
fig, ax = plt.subplots()
ax.plot(ts, hs, label='h[n]')
ax.plot(ts, xs, label='x{n]')

ax.legend()
plt.savefig("figures/p4_stems_ct.png")
plt.show()
```

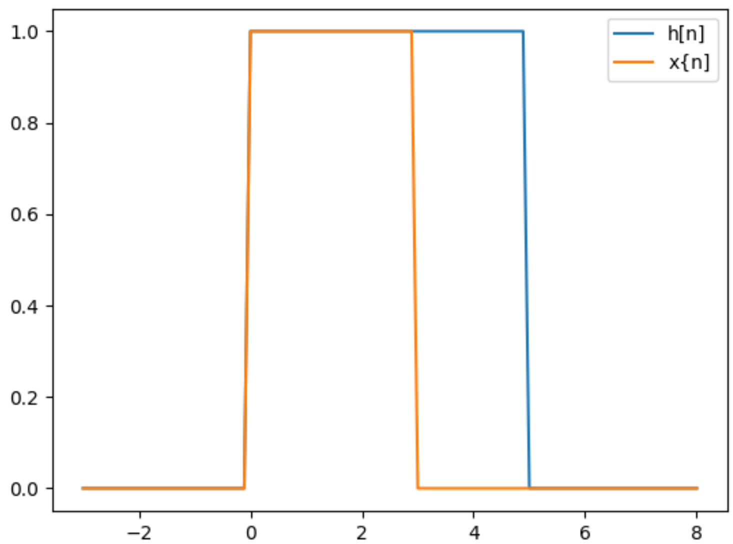

Figure 5: Input and impulse response.

Sketch convolution

We find this by integration

$$
\begin{aligned}
y(t) &= h(t) * x(t) && (51)\\
&= \int_{\mathbb{R}} d\lambda \;\; h(\lambda)x(t-\lambda) && (52)\\
&= \int_{\mathbb{R}} d\lambda \;\; (u(\lambda) - u(\lambda-5))(u(t-\lambda) - u(t-\lambda-3)) && (53)\\
&= \int_0^5 d\lambda \;\; u(t-\lambda) - u(t-\lambda-3) && (54)\\
&= \int_0^5 d\lambda \;\; u(t-\lambda) - \int_0^5 d\lambda \;\; u(t-\lambda-3). && (55)
\end{aligned}
$$

Then

$$y(t) = \begin{cases} 0, & \text{if } t < 0 \\ \int_0^t d\lambda = t, & \text{if } 0 \leq t < 3 \\ \int_0^t d\lambda - \int_0^{t-3} d\lambda \quad = 3, & \text{if } 3 \leq t < 5 \\ \int_0^5 d\lambda - \int_0^{t-3} d\lambda \quad = 8 - t, & \text{if } t \leq 8 \\ 0, & \text{if } t > 8 \end{cases} \tag{56}$$

```
def ys_fn(t):
    out = np.zeros_like(t)
    out[t < 8] = 8. -t[t<8]
    out[t < 5] = 3.
    out[t < 3] = t[t<3]
    out[t<0] = 0.

    return out

ts = np.linspace(-(n + n), m + m - 1, 100)
ys = ys_fn(ts)

#plot
fig, ax = plt.subplots()
ax.plot(ts, ys, 'tab:green', label='y[n]')
ax.set_xlim([-3, 10])
ax.legend()
plt.savefig("figures/p4_conv_ct.png")
plt.show()
```

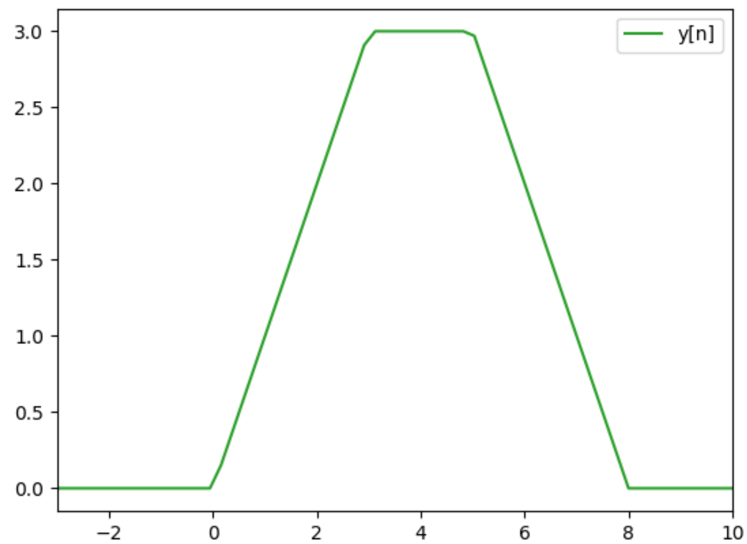

Figure 6: Output.

Unsurpirisngly this looks alot like the d.t. case.

Problem 5 Laplace

These types of question are part of the curriculum but not of the qualification description or "Kvalifikationsbeskrivelse" and will not be assessed in the final exam.