TTT4120 Digital Signal Processing

Problem Set 1

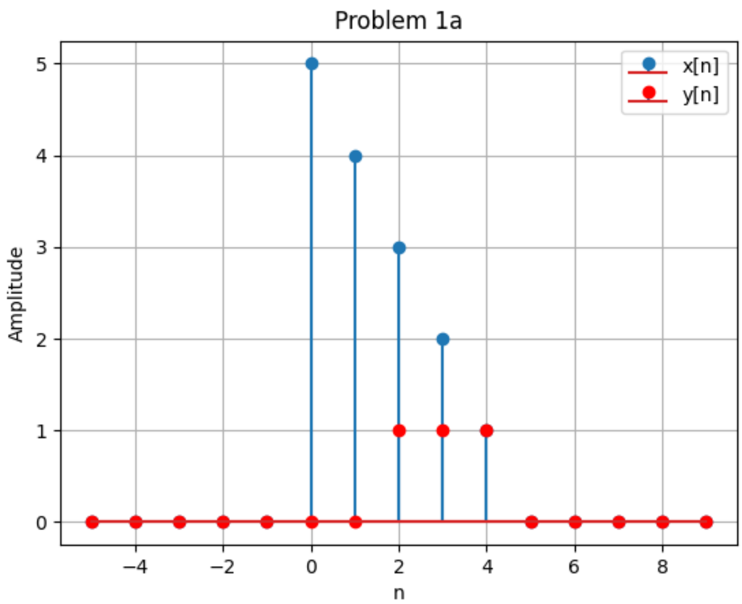
Peter Pham

2023-09-06

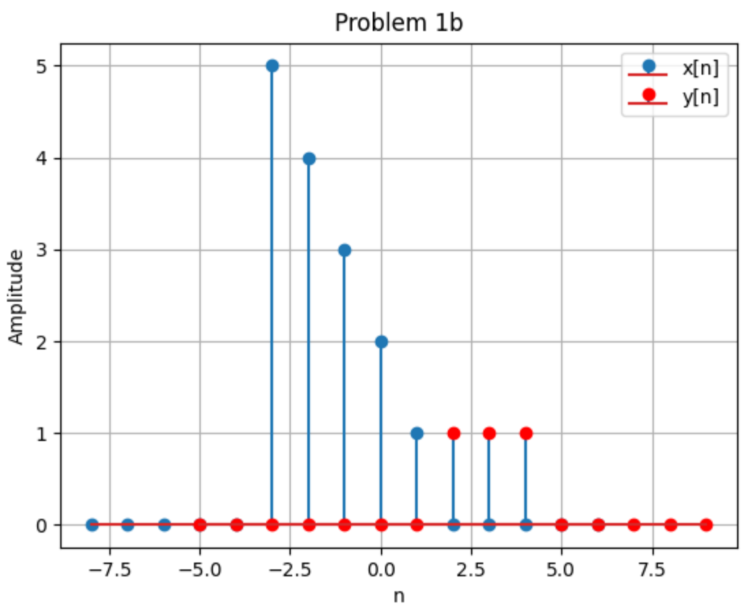
Contents

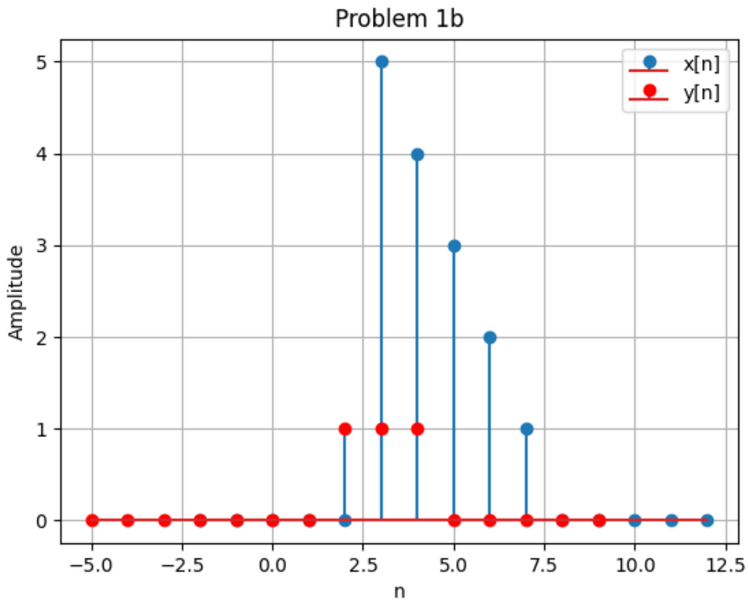
Problem 1 (2 points)

(a) Sketch $x[n]$ and $y[n]$

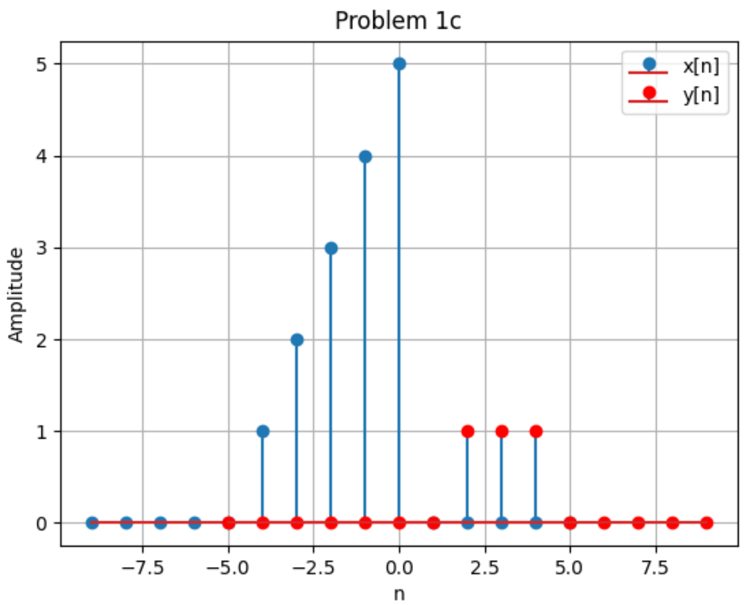


(b) Sketch $x[n-k]$ for $k = 3$ and $k = -3$.

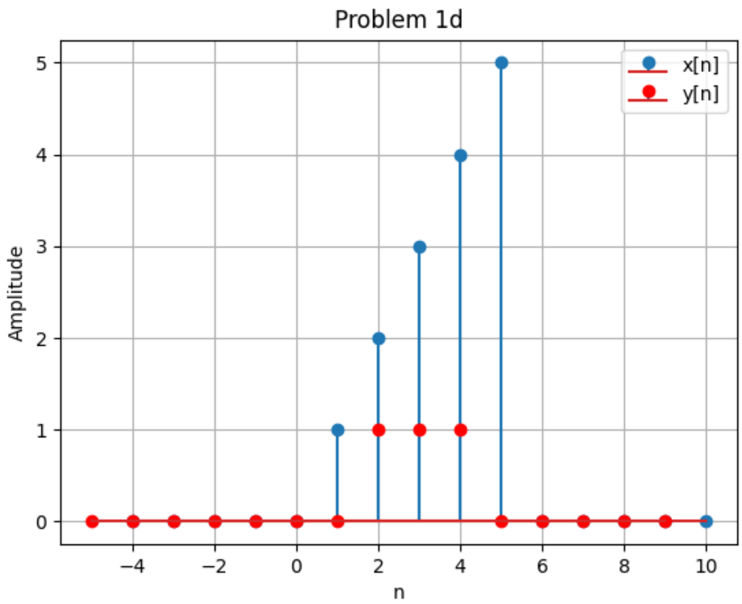



Problem 1b
x[n]
y[n]
Amplitude
n
5
4
3
2
1
0
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
12.5

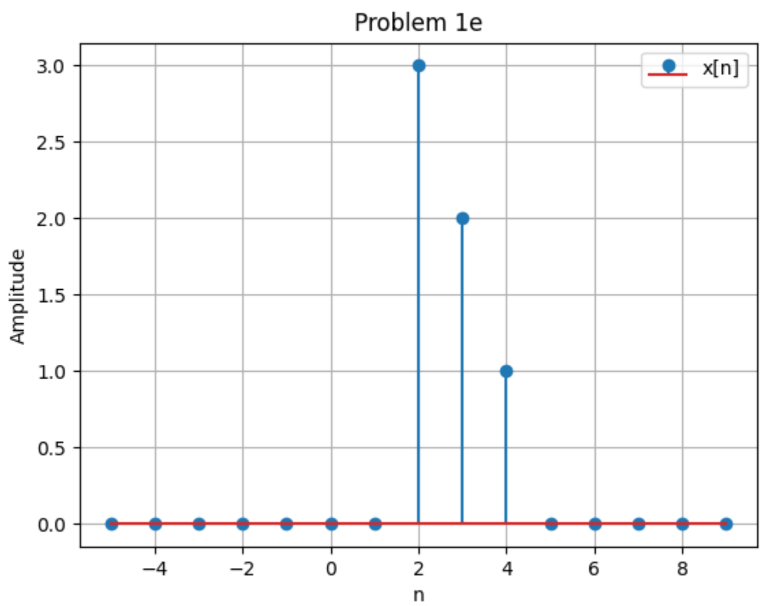
(c) Sketch $x[-n]$.


Problem 1c
x[n]
y[n]
Amplitude
n
5
4
3
2
1
0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5

(d) Sketch $x[5-n]$.



(e) Sketch $x[n] \cdot y[n]$.



(f) Express the signal $x[n]$ by using the unit sample sequence $\delta[n]$.

$$x[n] = 5\delta[n] + 4\delta[n-1] + 3\delta[n-2] + 2\delta[n-3] + 1\delta[n-4]$$

(g) Express the signal $y[n]$ by using the unit step signal $u[n]$.

$$y[n] = u[n-2] - u[n-5]$$

(h) Compute the energy of the signal $x[n]$.

The energy og the signal $x[n] = 55$ Solved in python

Python Code

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

# Problem 1:
def x_n(n):
    if 0 <= n <= 4:
        return 5 - n
    else:
        return 0

def y_n(n):
    if 2 <= n <= 4:
        return 1
    else:
        return 0

n_values = np.arange(-5, 10)  # A range of n values
x_values = [x_n(n) for n in n_values]
y_values = [y_n(n) for n in n_values]
plt.figure()


# (a) Sketch x[n] and y[n]
```

```
def problem_1a():
    plt.stem(n_values, x_values, label="x[n]")
    plt.stem(n_values, y_values, label="y[n]", markerfmt='ro')
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem␣1a')
    plt.grid(True)
    plt.show()
problem_1a()

# (b) Sketch x[n-k] for k=3 and k=-3
def problem_1b1():
    plt.stem(n_values-3, x_values, label="x[n]")
    plt.stem(n_values, y_values, label="y[n]", markerfmt='ro')
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem␣1b')
    plt.grid(True)
    plt.show()
problem_1b1()
def problem_1b2():
    plt.stem(n_values+3, x_values, label="x[n]")
    plt.stem(n_values, y_values, label="y[n]", markerfmt='ro')
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem␣1b')
    plt.grid(True)
    plt.show()
problem_1b2()

# (c) Sketch x[-n]
def problem_1c():
    plt.stem(-1*n_values, x_values, label="x[n]")
    plt.stem(n_values, y_values, label="y[n]", markerfmt='ro')
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem␣1c')
```

```
    plt.grid(True)
    plt.show()
problem_1c()

# (d) Sketch x[5-n]
def problem_1d():
    plt.stem(-1*n_values+5, x_values, label="x[n]")
    plt.stem(n_values, y_values, label="y[n]", markerfmt='ro')
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem 1d')
    plt.grid(True)
    plt.show()
problem_1d()

# (e) Sketch x[n] * y[n]
def problem_1e():
    def x_n_y_n(n):
        return x_n(n)*y_n(n)
    xy_values = [x_n_y_n(n) for n in n_values]

    plt.stem(n_values, xy_values, label="x[n]")
    plt.xlabel('n')
    plt.ylabel('Amplitude')
    plt.legend()
    plt.title('Problem 1e')
    plt.grid(True)
    plt.show()
problem_1e()



# (h) Compute the energy of the signal x[n]

def problem_1h():
    e=0
    for n in n_values:
        e+=x_n(n)*x_n(n)

    print("The energy og the signal x[n] = ",e)
```

```
problem_1h()

# Energy = sum of (x[n])^2 for all n
```

Problem 2 (2 points)

(a) Which physical frequencies F_1 can f_1 correspond to if $F_s = 6000Hz$?

As the sampling frequency need to be twice as high as the frequency after sampling we get from the equation:

$$-\frac{F_s}{2} \leq f \leq \frac{F_s}{2}$$

this gives us:

$$-3000Hz \leq f \leq 3000Hz$$

(b) Use Matlab or Python to generate a sequence of length 4 seconds of $x[n]$.

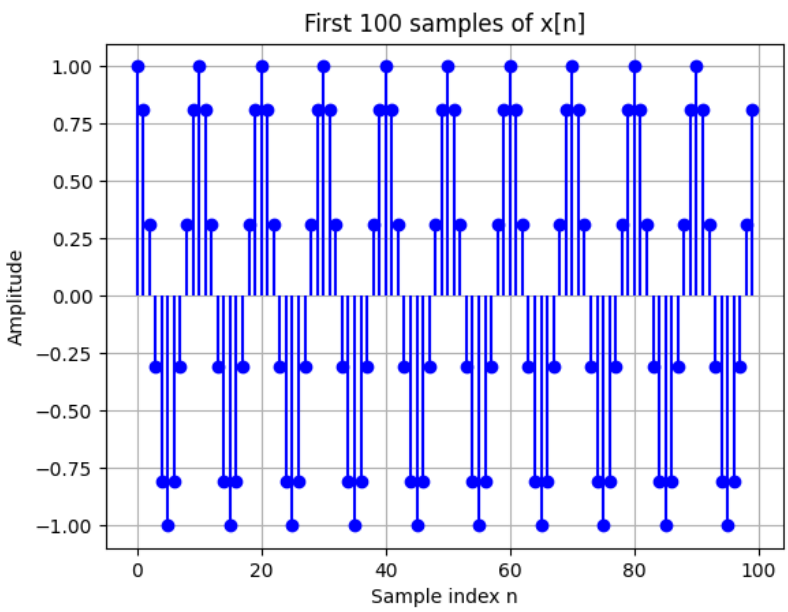


Figure 1: *
Generated sequence of lenght 4 seconds of $x[n]$ where $A = 1$, $f_1 =$, $F_s = 6000Hz$ and $T = 4s$

(c) Use the Matlab command soundsc or the Python command sounddevice.play to listen to the harmonic when the normalized frequency $f_1 = 0.3$and the sampling rate F_s is given by respectively 1000Hz, 3000Hz and 12000Hz. Comment on what you hear.

When listening to a higher normalized frequencies the pitch did also sound higher.

(d) Now assume a fixed sampling rate $F_s = 8000Hz$ while the physical frequency F_1 is respectively 1000Hz, 3000Hz and 6000Hz. Comment on what you hear. Relate it to the corresponding normalized frequency f_1.

In this case the pitch was diffrent where the highest pitch was 3000 Hz, but then i have difficulty deciding whenever if 1000Hz og 6000Hz are the highest pitch. When i played the tones again with the corresponding normalized frequency f_1 the pitch sounded accordingly to the frequencies 6000Hz being the highest and 1000Hz being the lowest.

Problem 3 (2 points)

I will write a bit more on problem (3(a)) for future usage, the rest will use the same methods only shorter.

Linearity

To decide whenever a system is linear or not we have to check if the principle of superposition is satisfied.

Additivity: $T(x_1[n] + x_2[n]) = T(x_1[n]) + T(x_2[n])$

Time-Invariance

A system is time-invariant if a time shift in the input signal results in an identical time shift in the output signal. Mathematically, this property can be expressed as:

$$T(x[n-k]) = y[n-k] \tag{1}$$

where $T(x[n]) = y[n]$.

Causality

A system is causal if the output at any time n depends only on the present and past input values but not on future input values. Mathematically, this can be expressed as:

$$y[n] = T(x[n], x[n-1], x[n-2], \ldots)$$

(a) $y[n] = x[n] - x^2[n-1]$

Linearity

Additivity: If we have two signals $x_1[n]$ and $x_2[n]$. The system response of $x_1[n] + x_2[n]$ would be:

$$\begin{aligned} y[n] &= (x_1[n] + x_2[n]) - (x_1[n-1] + x_2[n-1])^2 & (2) \\ &= x_1[n] + x_2[n] - (x_1^2[n-1] + 2x_1[n-1]x_2[n-1] + x_2^2[n-1]) & (3) \end{aligned}$$

This is not equal to $y_1[n] + y_2[n]$ where $y_1[n] = x_1[n] - x_1^2[n-1]$ and $y_2[n] = x_2[n] - x_2^2[n-1]$

Therefore This system is not linear

Time-Invariance

If we consider an input $x[n-k]$, the system response would be:

$$y[n] = x[n-k] - (x[n-k-1])^2$$

This is exactly the output $y[n]$ shifted by k samples, assuming $T(x[n]) = y[n]$

Causality

In this system, the outpyt $y[n]$ at any time demepds on the current $x[n]$ and the past input $x[n-1]$, this means that the system is causal.

Summary

- The system is not linear.
- The system is time-invariant.
- The system is causal.

(b) $y[n] = nx[n] + 2x[n-2]$

- **Linearity**, two signals $x_1[n]$ and $x_2[n]$:

  $$y[n] = nx_1[n] + nx_2[n] + 2x_1[n-2] + 2x_2[n-2]$$

  This is equal to $y_1[n]+y_2[n]$ where $y_1[n] = nx_1[n]+2x_1[n-2]$ and $y_2[n] = nx_2[n]+2x_2[n-2]$

- **Time-Invariance**, consider an input $x[n-k]$

  $$y[n] = nx[n-k] - 2x[n-k-2]$$

  This is not the same as $y[n]$ shifted by k samples as it would be:

  $$y[n-k] = (n-k) \cdot x[n-k] + 2 \cdot x[n-k-2]$$

- **Causality**: The same as before, the output depends of a past input $x[n-2]$

Summary

- The system is linear.
- The system is not time-invariant.
- The system is causal.

(c) $y[n] = x[n] - x[n-1]$

- **Linearity**, two signals $x_1[n]$ and $x_2[n]$:

$$y[n] = x_1[n] + x_2[n] - x_1[n-1] - x_2[n-1]$$

  This is equal to $y_1[n] + y_2[n]$ where $y_1[n] = x_1[n] - x_1[n-1]$ and $y_2[n] = x_2[n] - x_2[n-1]$

- **Time-Invariance**, consider an input $x[n-k]$

$$y[n] = x[n-k] - x[n-k-1]$$

  This is the same as $y[n]$ shifted by k samples as it would be:

$$y[n-k] = x[n-k] - x[n-k-1]$$

- **Causality**: The same as before, the output depends of a past input $x[n-1]$

Summary

- The system is linear.
- The system is time-invariant.
- The system is causal.

(d) $y[n] = x[n] + 3x[n+4]$

- **Linearity**, two signals $x_1[n]$ and $x_2[n]$:

$$y[n] = x_1[n] + x_2[n] + 3x_1[n+4] + 3x_2[n+4])$$

  This is equal to $y_1[n]+y_2[n]$ where $y_1[n] = x_1[n]+3x_1[n+4]$ and $y_2[n] = x_2[n]+3x_2[n+4]$

- **Time-Invariance**, consider an input $x[n-k]$

$$y[n] = x[n-k] + 3x[n-k+4]$$

  this is the same as $y[n]$ shifted by k samples.

- **Causality**: The system is not casual because the output at time n depends on a future input $x[n+4]$.

Summary

- The system is linear.
- The system is time-invariant.
- The system is not causal.

Problem 4 (2 points)

$$y[n] = x[n] + 2x[n-1] + x[n-2]$$
$$y[n] = -0.9y[n-1] + x[n].$$

(a)Find the unit pulse responses $h_1[n]$ and $h_2[n]$ of the systems.

The unit pulse response $h[n]$ of a system is the output whenever ethe input is a unit pulse $\delta[n]$, which is 1 at n=0 and 0 everywhere else.

$$\delta[n] = \begin{cases} 1, & \text{if } n = 0 \\ 0, & \text{otherwise} \end{cases}$$

by substituting $x[n]$ with $\delta[n]$ we get:

$$h_1[n] = \delta[n] + 2\delta[n-1] + \delta[n-2]$$
$$h_2[n] = -0.9h_2[n-1] + \delta[n].$$

(b) State whether each system is FIR or IIR. Justify your answer.

The first system unit pulse response will eventually become zero as the impulse response only depends on the current and past inputs, therefore it is a FIR system. The second system on the other hand will have a unit pulse system that continues indefinitly as the you have to consider the previous outputs $y_2[n-1]$ to find the impulse response $h_2[n]$. This means that this system is IIR.

(c) Determine whether each system is stable.

A system is bounded-input bounded-output stable (BIBO) if

$$|x[n]| \leq M_x < \infty \Rightarrow |y[n]| \leq M_y < \infty \Rightarrow, \forall n$$

The first system is an FIR system and has finite coefficients, therefore the system wil produce a bounded output for any bounded input.

The second system is a IIR system, therefore its stability is determined by the roots of the characteristic equation assosiated with the system,We can find that by setting $x[n] = 0$ and solving the resulting homogenous equation. In this case it would be:

$$y_2[n] + 0.9y_2[n-1] = 0$$

The roots of this equation would be determined by the charasteristic polymial:

$$1 + 0.9x^{-1} = 0$$

This gives us that the root would be $z = -\frac{1}{0.1} = -1.11111$. Since the magnitude is greater than 1, this system is stable.

(d) Represent the filters graphically by a filter structure.

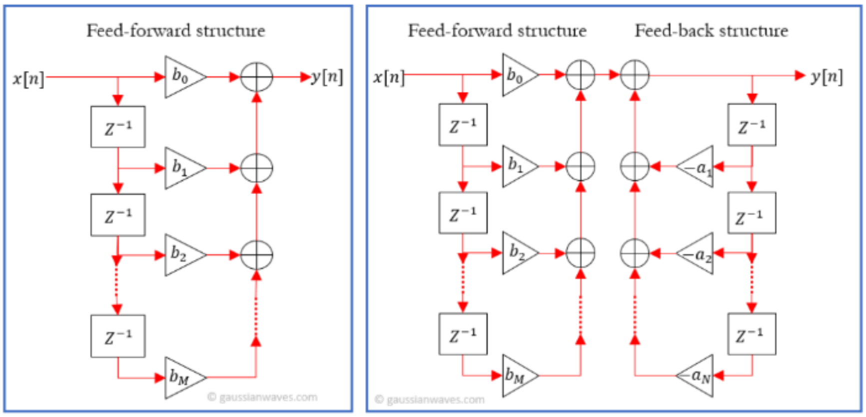


Figure 2: