Signals and Systems

Laboratory Report 1B

Author: ***Jumna Alic***
Instructor: ***Tadeusz Stepinski***
Date: ***April, 2025***

Academic Year 2024/2025

Task 1

In the first task, we write a MATLAB script to generate the discrete-time function

$$y[n] = 1.97 \cdot y[n-1] - y[n-2],$$

with the initial conditions $y[1] = 1$ and $y[0] = 0$. The MATLAB code used to implement this function is shown below:

```
%Lab2_Task1

n = 0:50;
yn = zeros(1, length(n));
% Initial conditions
yn(1) = 0;  %  y[0]
yn(2) = 1;  %  y[1]

for k = 3:length(n) % MATLAB indexing starts at 1, n=2 is at index 3
    yn(k) = 1.97 * yn(k-1) - yn(k-2);
end

% Plot
stem(n, yn, 'filled');
title('Difference Equation');
xlabel('Time index n');
ylabel('$y[n]$', 'Interpreter', 'latex');
grid on;
```

The figure generated by the code above illustrates the plot of the response of the difference equation $y[n]$ over the discrete time index n. Figure is shown below.

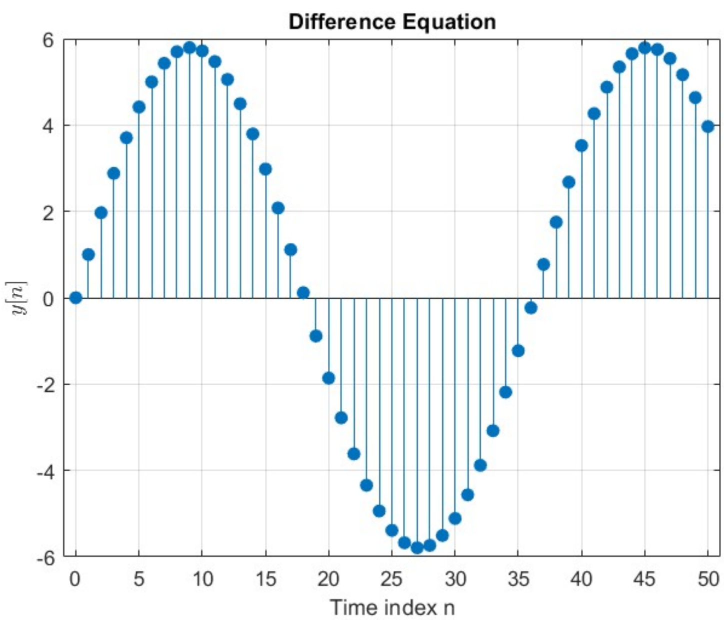

Figure 1: Response $y[n]$

From the plot, we can clearly see that the output starts with $y[0] = 0$ and $y[1] = 1$, as expected from the initial conditions. The rest of the values follow the difference equation and show how the response evolves over time based on those starting points.
After this we change the difference equation to

$$y[n] = 1.90 \cdot y[n-1] - y[n-2]$$

Initial conditions stay the same, $y[1] = 1$ and $y[0] = 0$.
The MATLAB code used to implement the changed function is shown below:

```
%Lab2_Task1_b

n = 0:50;
yn = zeros(1, length(n));
% Initial conditions
yn(1) = 0;  %  y[0]
yn(2) = 1;  %  y[1]


for k = 3:length(n) % MATLAB indexing starts at 1, n=2 is at index 3
    yn(k) = 1.90 * yn(k-1) - yn(k-2);
end

% Plot
stem(n, yn, 'filled');
title('Difference Equation');
xlabel('Time index n');
ylabel('$y[n]$', 'Interpreter', 'latex');
grid on;
```

The figure generated by this code is shown below.

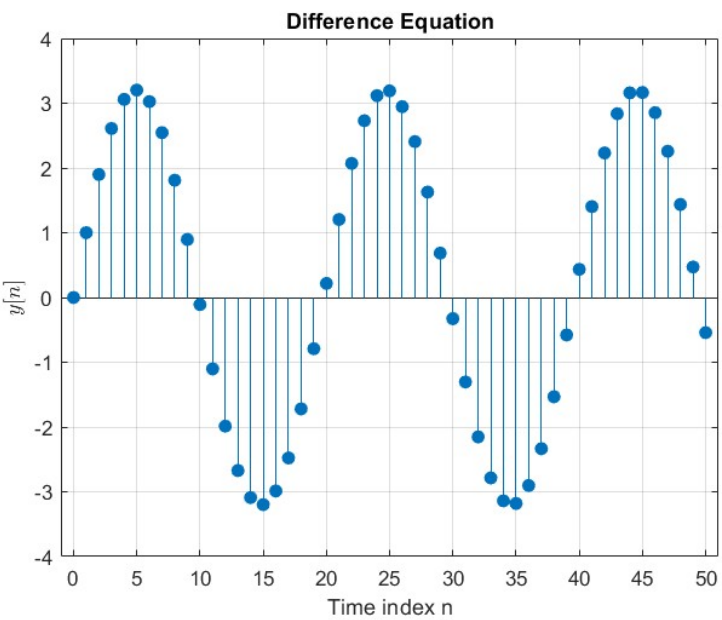

Figure 2: Response $y[n]$

When comparing the responses of two systems, we can see that for the first system, where $a = 1.97$, we observe a more pronounced oscillatory behavior. The larger value of a leads to larger oscillations, which persist for a longer period. Although the system remains stable with the given initial conditions, the oscillations are quite pronounced, and the system is close to the instability boundary. If the value of a were increased further, it could lead to instability.

In contrast, the second system, with $a = 1.90$, exhibits smaller oscillations that decay more rapidly. The lower value of a results in a system that is more stable and has less oscillatory behavior. The oscillations still occur, but their amplitude is smaller and reduces over time, leading to faster stabilization.

Thus, the main difference between the two systems is the amplitude and persistence of the oscillations. The system with $a = 1.97$ exhibits larger, more sustained oscillations, while the system with $a = 1.90$ has smaller, decaying oscillations. Both systems remain stable, but the one with $a = 1.90$ is more damped and stabilizes faster.

In summary, as the value of a decreases, the system becomes less oscillatory and more stable, whereas increasing a results in larger oscillations and a system that is closer to the instability boundary.

Task 2

In Task 2, we write a MATLAB script to generate the response $y[n]$ for the second-order system shown in the figure below.

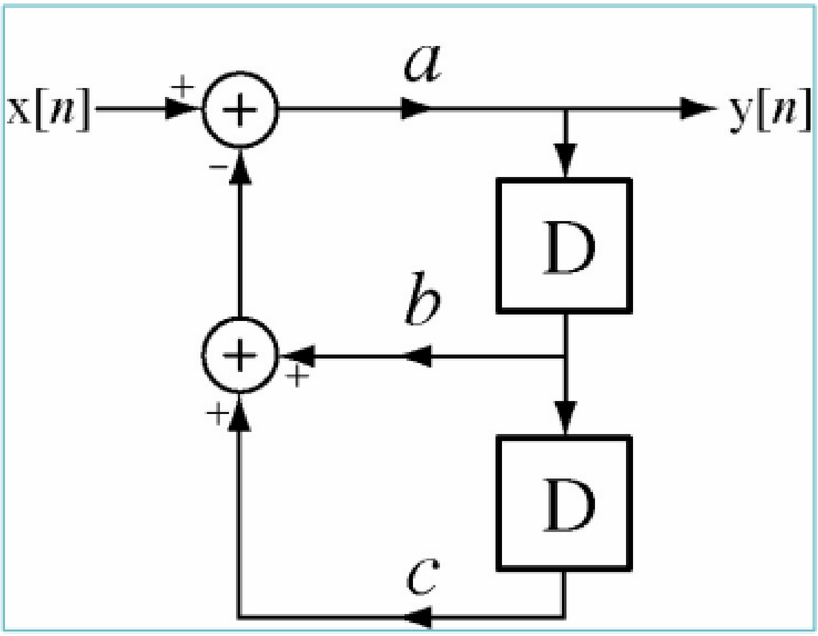

Figure 3: Second-order system

The system described in Figure 3 is modeled by the following difference equation:

$$y[n] = a \cdot x[n] - a \cdot b \cdot y[n-1] - a \cdot c \cdot y[n-2]$$

By changing the values of a, b, and c, we observe different responses for the system. In this task, the initial values for the coefficients are set to $a = 1$, $b = -1.5$, and $c = 0.8$. The MATLAB code used to generate this response is shown below:

```
%Lab2_Task2

n = 0:60;
y = [0 ones(1, length(n)-1)]; % First value is 0, rest are ones
x = ones(1, length(n));

a = 1;
b = -1.5;
c = 0.8;

for k = 3:length(n)
    y(k) = a*x(k) - a*b*y(k-1) - a*c*y(k-2);
end

% Plot
stem(n, y, 'filled');
title(['Difference Equation, a=' num2str(a) ', b=' num2str(b) ', c=' num2str(c)]);
xlabel('Time index n');
ylabel('$y[n]$', 'Interpreter', 'latex');
grid on;
```

The figure generated by the code above illustrates the plot of the response of the difference equation $y[n]$ over the discrete time index n. This plot is shown below.

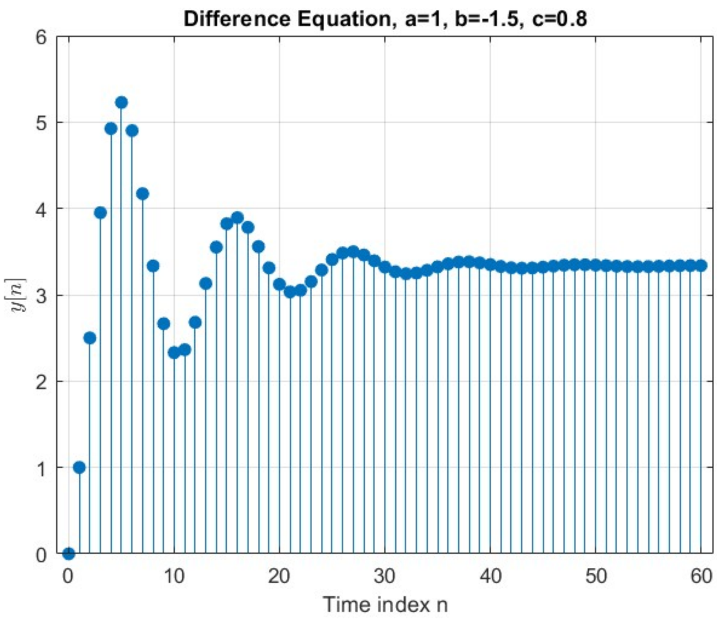

Figure 4: Response $y[n]$

Next, we change the value of c to 0.6. The response generated for these new values is shown in Figure below.

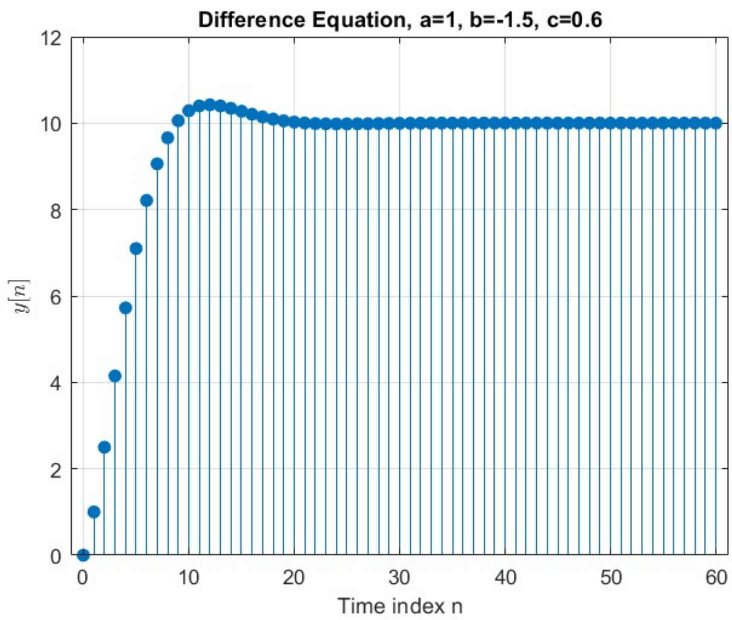

Figure 5: Response $y[n]$

Finally, we change the value of c again, setting it to 0.5. The response generated for this value of c is shown in Figure below.

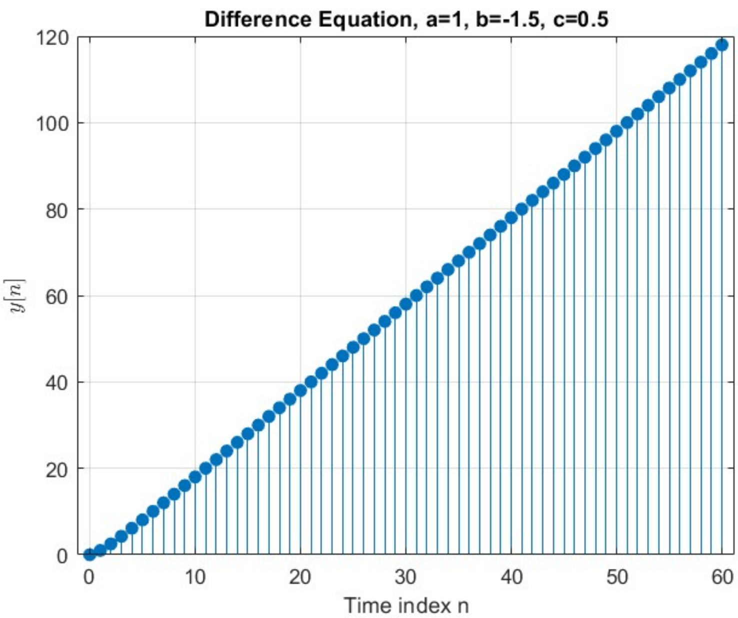

Figure 6: Response $y[n]$

In this task, we observed how changing the value of c affects the system's response. Initially, with $c = 0.8$, the system exhibited a certain level of oscillation and damping, which is evident in Figure 4.
As we reduced c to 0.6, shown in Figure 5, the system's response became more damped and less oscillatory, indicating that lower values of c reduce the impact of past outputs on the current output, thus slowing down the response.

Finally, when we further reduced c to 0.5, the system response, as shown in Figure 6, became even more subdued, exhibiting a further damped and stable behavior.
This suggests that as the value of c decreases, the system moves towards a more stable, less oscillatory response.

Overall, these results demonstrate the sensitivity of a second-order discrete system to the values of its coefficients. Tuning c plays a crucial role in controlling the damping behavior of the system, and the plots highlight how varying these parameters influences the system's dynamics and stability.

After this, we observed the effect of increasing the value of the coefficient a to a value greater than 1. First, we changed a to 1.2, and the resulting response is shown below.

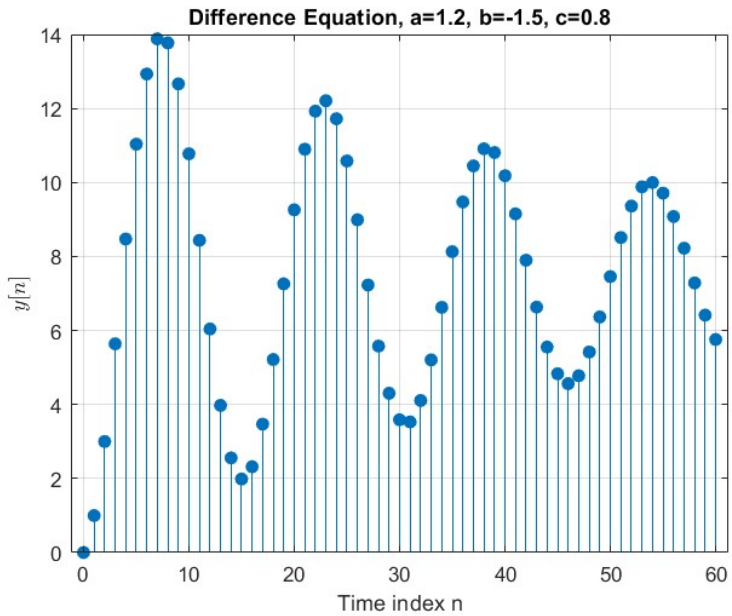

Figure 7: Response $y[n]$

Next, we increased the value of a to 1.5. The response generated for this value of a is shown in Figure below.

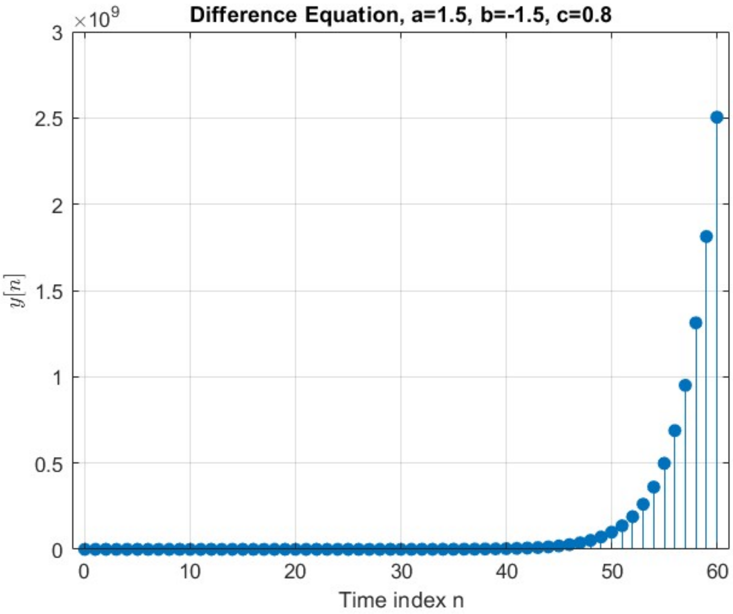

Figure 8: Response $y[n]$

When we increased the value of a, the system's response became more pronounced, with greater amplitude oscillations.
Since $x[n] = 1$ for all n, the changes in the output are purely due to the variations in the coefficient a. With $a = 1.2$, the system exhibited larger oscillations, as shown in Figure 7, compared to the behavior observed with smaller values of a.
Increasing a further to 1.5, as shown in Figure 8, led to even larger oscillations, emphasizing the system's sensitivity to changes in a.
A higher value of a increases the system's response amplitude, making the output more oscillatory.
For $a = 1.2$, the system remains stable, with oscillations decaying over time. However, when a is increased further to 1.5, the system becomes unstable, and the oscillations grow in amplitude. This behavior indicates that increasing a beyond a certain point leads to instability in the system.
Therefore, for $a = 1.2$, the system is stable, but for $a = 1.5$, the system becomes unstable, with the oscillations continuing to grow, signifying an unstable response.