



Linear Control Systems (EE-379)

DE-44 Mechatronics

Syndicate– C

Lab Report 2

Name of members:

NC M. Ahmed Hussain Babar (427781)

PC Abdul Haseeb Zahid (432219)

PC Muhammad Rizwan Aslam (432069)

NC Muhammad Saqib (411829)

Submitted to: LE M.Shahzad Alam

Abstract

This lab focused on time-domain analysis of linear control systems, emphasizing key performance metrics such as settling time, rise time, peak time, and overshoot. The exercise introduced first- and second-order systems and their transient responses. Using MATLAB, a second-order transfer function was simulated with a step input to evaluate system behavior. Practical applications were explored, including system stabilization in engineering contexts. The lab reinforced ethical practices and collaboration, building on foundational control system concepts for performance evaluation.

Introduction

This lab utilized **MATLAB** to perform a **time-domain analysis** of linear control systems. The core objective was to numerically characterize system behavior using key **time-domain specifications**, including **rise time**, **settling time**, **peak time**, and **overshoot**. By simulating system responses, we explored the characteristic behaviors of first-order (exponential) and second-order (damped) systems, using models like RLC circuits, to evaluate performance against standard engineering design requirements.

Objectives

- To understand key time-domain performance metrics in control systems, including settling time, rise time, peak time, and overshoot.
- To differentiate between the characteristics and responses of first-order and second-order systems.
- To apply MATLAB tools for modeling and simulating control systems, such as creating transfer functions with `tf`, simulating responses with `lsim`, and extracting performance data using `stepinfo`.
- To analyze step responses of systems and interpret transient behavior in practical engineering applications, like stability tuning in mechatronic systems.

Methodology

1. Clear workspace using `clc`, `clear`, and `clear all`.
2. Define numerator and denominator for the transfer function: `num = [4]`, `den = [1 5 4]`, representing $G(s) = 4 / (s^2 + 5s + 4)$.
3. Create transfer function object: `tf = tf(num, den)`.
4. Generate time vector: `time = linspace(0, 20, 1000)`.
5. Create a delayed unit step input (starting at $t=1$): Initialize step as zeros, then set to 1 for $time > 1$ using a for loop.
6. Simulate system response: `y = lsim(tf, step, time)`.
7. Plot response and input: `plot(time, y)`, `hold on`, `plot(time, step)`.
8. Extract performance metrics: `info = stepinfo(y, time)`.

Code:

Matlab Code

```
clc
clear
clear all
num = [4]
den = [1 5 4]
tf = tf(num, den)

time = linspace(0, 20, 1000);
step = zeros(size(time));
for i = 1 : length(step)
    if time(i) <= 1
        step(i) = 0
    else
        step(i) = 1
    end
end
y = lsim(tf, step, time)
```

```
plot(time, y)
hold on
plot(time, step)
info = stepinfo(y, time);
```

Results

The MATLAB simulation produced the following output:

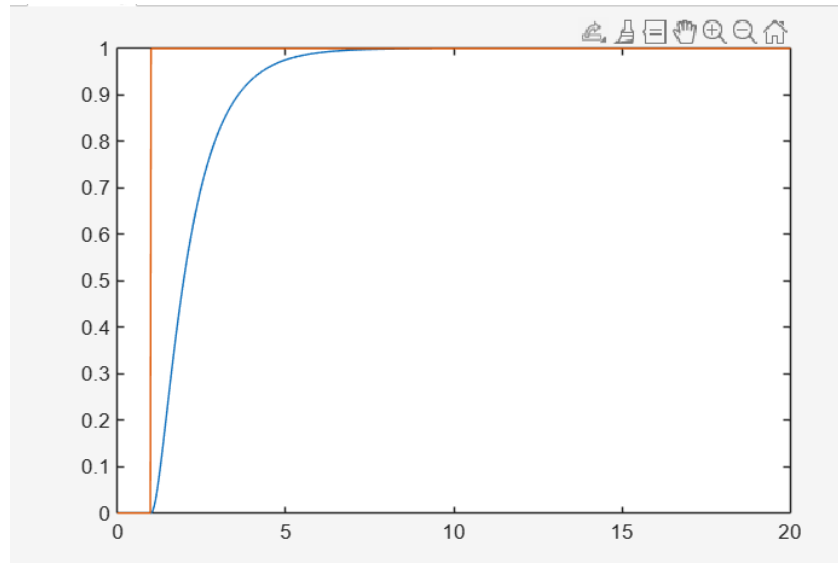


Figure 1: Output Graph

The transfer function $G(s) = 4 / (s^2 + 5s + 4)$ was simulated with a unit step input delayed to $t=1$. The response plot showed a smooth rise to the steady-state value of 1 without oscillation or overshoot, characteristic of an overdamped system.

Key performance metrics from stepinfo:

- RiseTime: 2.3121 seconds
- TransientTime: 5.2007 seconds
- SettlingTime: 5.2007 seconds
- SettlingMin: 0.9012
- SettlingMax: 1.0000
- Overshoot: 0
- Undershoot: 0
- Peak: 1.0000
- PeakTime: 20 seconds

Discussion

The lab integrated theoretical time-domain concepts with practical MATLAB simulation. The overdamped response (no overshoot, gradual settling) was expected given the system's poles, demonstrating how damping affects transients. Using `lsim` and `stepinfo` streamlined analysis, avoiding manual plotting or calculations. In engineering, such metrics are crucial for tuning; for instance, reducing settling time in robotics enhances efficiency, while minimizing overshoot prevents instability. Limitations included the simplified model (no noise or nonlinearities), but the exercise provided a solid foundation for advanced topics like controller design. Potential extensions involve comparing damping ratios or adding feedback loops.

Conclusion

This laboratory exercise developed understanding of time-domain performance metrics in linear control systems. Through MATLAB simulation of a second-order system, key parameters like rise time, settling time, peak time, and overshoot were analyzed, highlighting their role in system evaluation. Ethical collaboration ensured effective learning. The skills acquired support further exploration of control strategies and real-time implementations.