**Department of Electronic and Telecommunication Engineering**
**University of Moratuwa**


# Assignment 1
## System Identification and PID Control

## EN2143 - Electronic Control Systems

Aazir M.A.M. - 220005R

March 20, 2025

# Contents

# 1. Introduction

In this MATLAB based activity, our task is to use our own transfer function G of order greater than 2 with at least one zero. Then use it to generate input and output data (u,y). After that, we should design a PID controller for the estimated plant Gs (tf1). Finally visualizing the step response of the controlled plant.

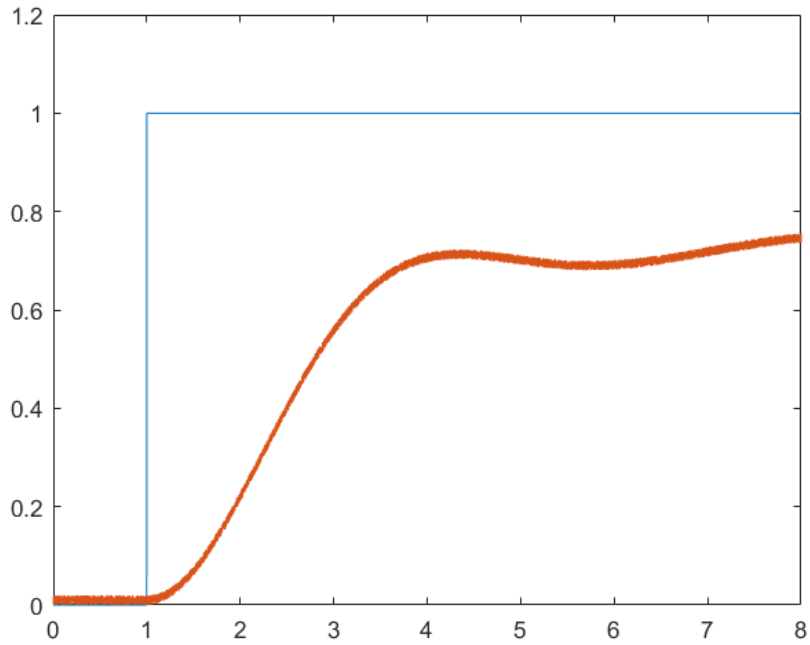
# 2. Transfer Function

Here, I have used a third-order transfer function with one zero at -1.5. The transfer function of the plant:

$$G_{(s)} = \frac{s + 1.5}{2s^3 + 3s^2 + 5s + 2}$$

The following MATLAB code is used to define the transfer function and simulate the step-response of the plant.

```
clc; close all; clear;

s = tf('s');
G = (s + 1.5) / (2*s^3 + 3*s^2 + 5*s + 2);  % My transfer Function for data generation
dt = 0.001;                                  % Sampling Interval
t = 0:dt:8;
u = ones(length(t),1);
u(1:1/dt)=0;                                 % this sets the first 10 samples to zero
y = lsim(G,u,t);                             % this generates the plant response for input u
y = y + rand(length(t),1)*0.02;              % add random noise to the response
plot(t,[u,y]); axis([0 8 0 1.2]);
```

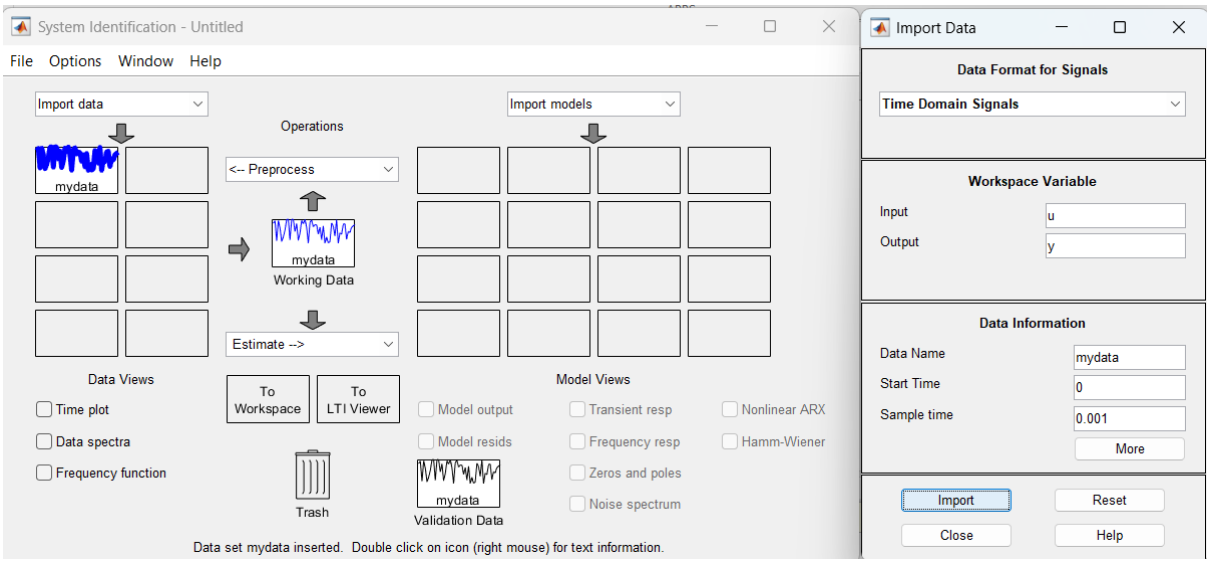



The plotted step-response of the plant is shown above. The response settles around 0.8 after a few oscillations.

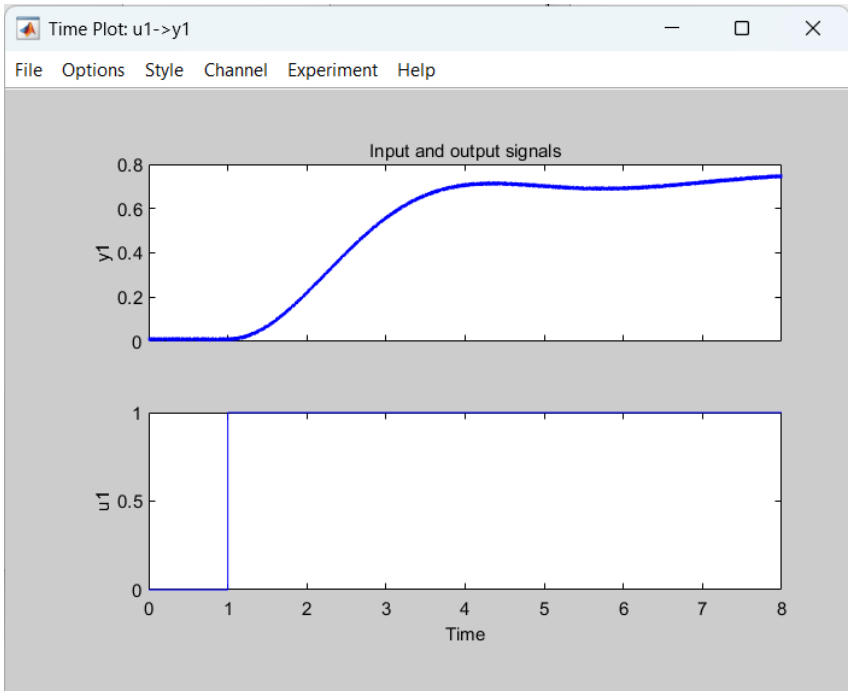
# 3. System Identification

Next, we import the input and output data into System Identification App which is used to estimate the transfer function of the plant.

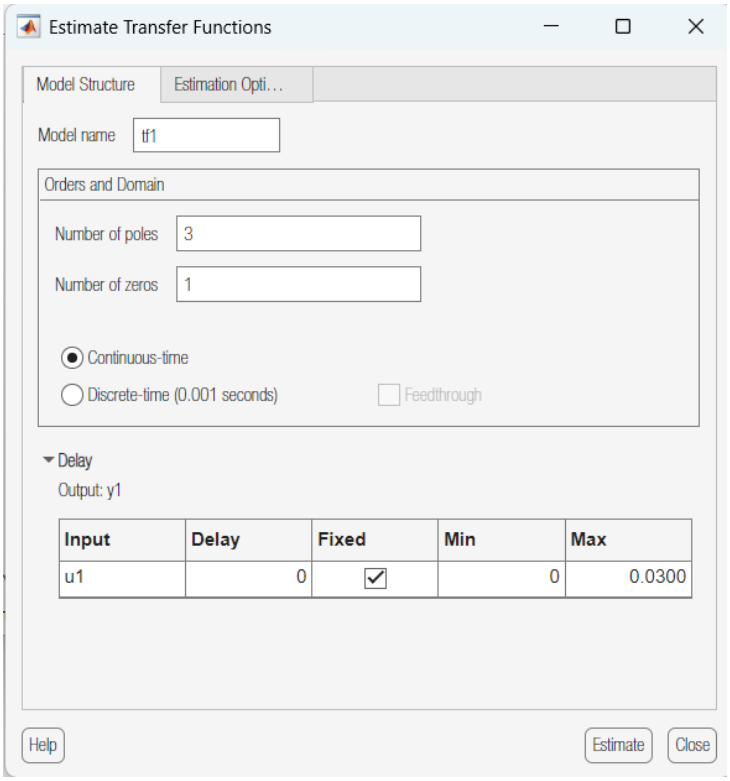
Importing the Input (u) and Output (y) data of the plant.



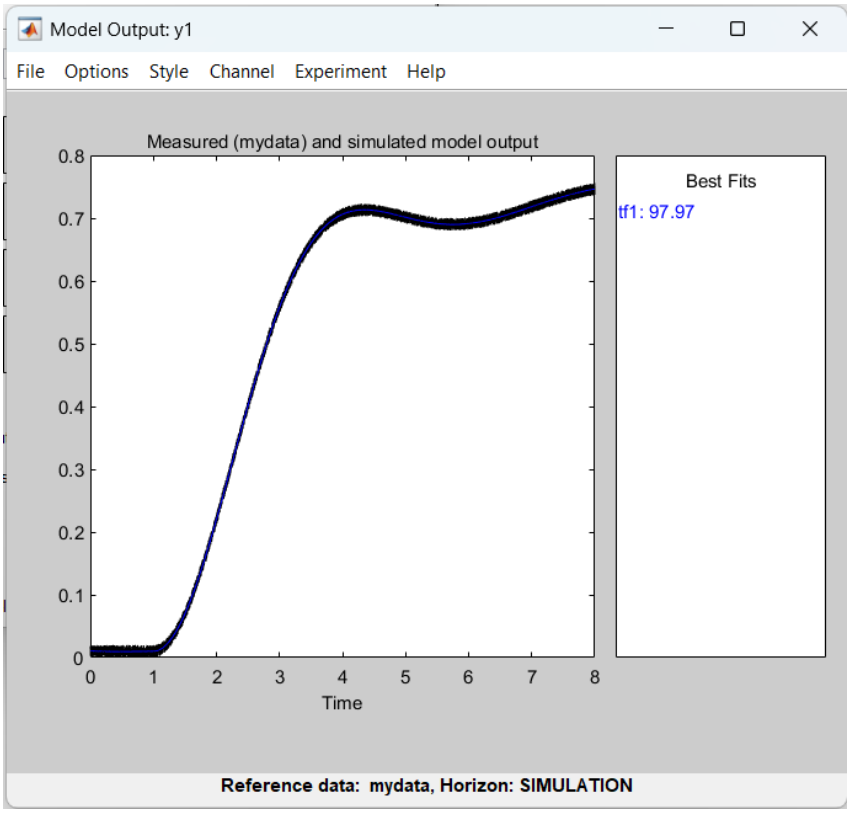

Check time plot, visualize data, and confirm the generated data has been properly imported.



Estimate the Transfer Function by choosing 3 poles and 1 zeros.




After the estimation let's see how closely the model output fits with the actual data.



# 4. Comparison of the Models

The transfer function of the estimated model from the System Identification App is:

$$\mathrm{tf1}_{(s)} = \frac{0.4972s + 0.7673}{s^3 + 1.502s^2 + 2.51s + 1.01}$$

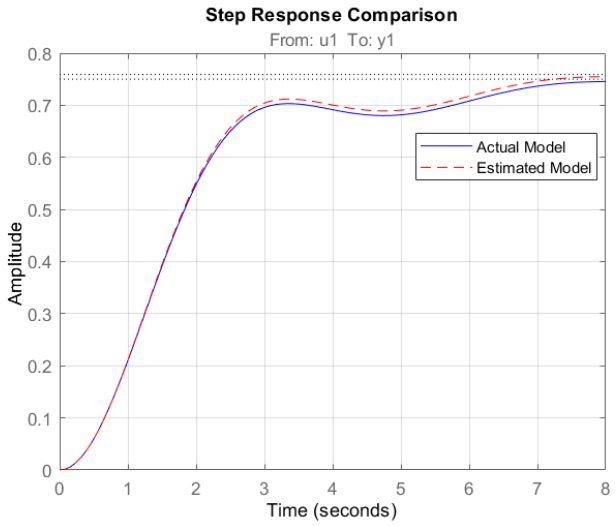
The accuracy of the estimation from the app is approx. **97.97%**.

```
>> G

G =

          s + 1.5
  ----------------------
  2 s^3 + 3 s^2 + 5 s + 2

Continuous-time transfer function.
Model Properties
>> tf1

tf1 =
  From input "u1" to output "y1":
         0.4972 s + 0.7673
  -------------------------------
  s^3 + 1.502 s^2 + 2.51 s + 1.01

Name: tf1
Continuous-time identified transfer function.

Parameterization:
   Number of poles: 3   Number of zeros: 1
   Number of free coefficients: 5
   Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.

Status:
Estimated using TFEST on time domain data "mydata".
Fit to estimation data: 97.97% (stability enforced)
FPE: 3.299e-05, MSE: 3.293e-05

Model Properties
>> zero(G), zero(tf1)

ans =

   -1.5000


ans =

   -1.5431

>> pole(G), pole(tf1)

ans =
```

```
  -0.5000 + 1.3229i
  -0.5000 - 1.3229i
  -0.5000 + 0.0000i


ans =

  -0.4992 + 1.3260i
  -0.4992 - 1.3260i
  -0.5033 + 0.0000i

>>
```

Figure 1: Useful Information of Both Models

As we can see above, the placements of zeros and poles don't differ much in the estimated model from the actual plant.
Below is the comparison of the step-response between the actual model and estimated model.



## 5. PID Controller Design

Here, We use PID Tuner App of MATLAB to build a PID controller for our plant. The use of a PID feedback will improve the system's performance. The initial parameters of our PID Controller given by the app is:

$$K_p = 2.615, K_i = 1.515, K_d = 1.129$$

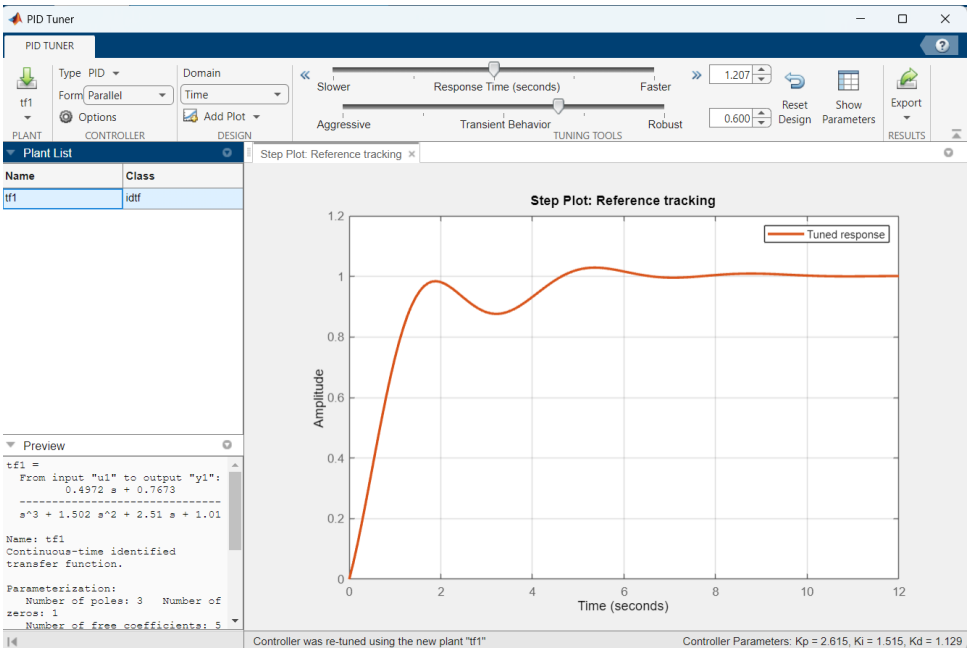
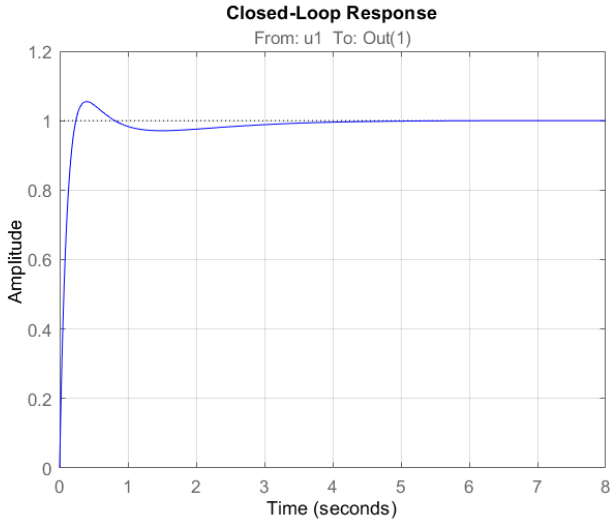


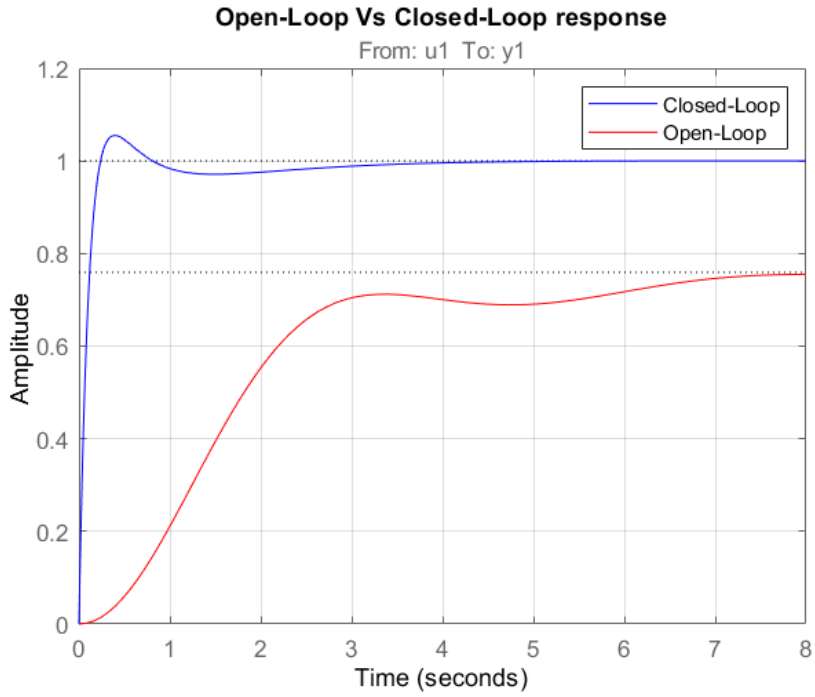

Figure 2: PID Tuner

The closed-loop step response after further tuning the PID controller manually is:

$$K_p = 31.618, K_i = 11.6535, K_d = 21.4464$$



A comparison of the Open Loop Vs Closed Loop Step response can be seen below.




Even though there is a initial overshoot, the system was able to settle much quickly and keep up the stability.