Lab 2 Report

1. Introduction

In this experiment, we perform system identification on a dynamic system using time-series data. The goal is to identify a mathematical model that represents the system's behavior by fitting it to input-output data. Specifically, we focus on building a transfer function and a state-space model based on the observed data, and validating the model using simulation.

2. Objectives

The objectives of the experiment are:

To preprocess the input-output data by removing offsets and noise.

To identify the system parameters and derive a transfer function.

To verify the model by comparing simulated outputs with the actual system behavior.

3. Materials and Methods

Data Collection

The data for this experiment was obtained from a dataset called SysIdenData_1.mat. This data includes time-series measurements of water level (y_act) and pump voltage (u_act) over time. The dataset is loaded into MATLAB using the following command:

```
load('SysIdenData_1.mat');
```

Data Preprocessing

The raw data contains offsets and noise, which need to be removed for accurate modeling. The preprocessing steps involve the following:

1. Time Synchronization and Cropping: The data is cropped to remove irrelevant initial periods where the system is in a steady state. A cutoff time (t_cutoff) is chosen to remove this section. In this system, V_max we set is 3V, V_min = 2.1V. Hence, the code can calculate:

```
u_offset = (V_max + V_min)/2;
t_cutoff = 132/0.75;
```

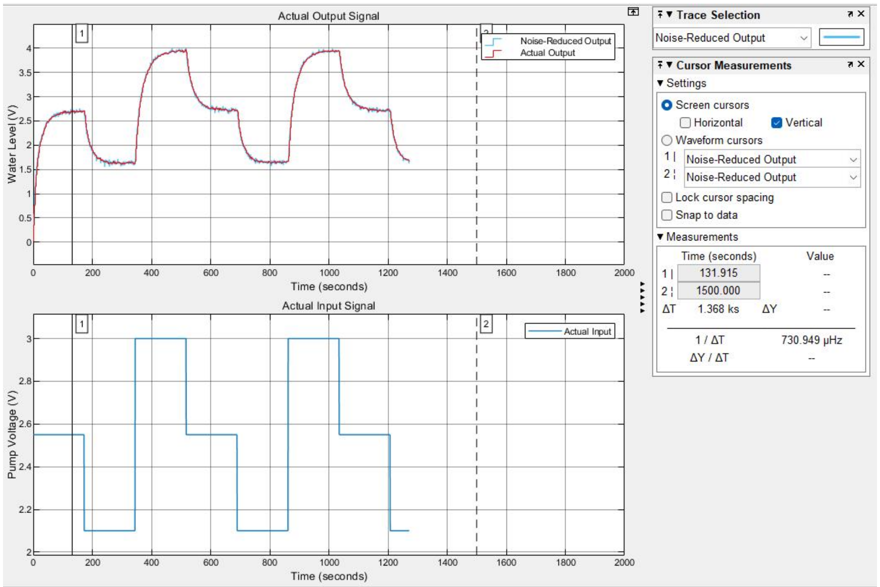

Figure 1: Data cropping at the t_cutoff point, removing the initial steady-state portion.

2. Offset Removal: The input signal (u_act) and output signal (y_act) are both offset by their respective mean values to ensure that the system's zero-state response is considered. This is done by subtracting the average value over the first steady-state period.

```
t = LogData.time(t_cutoff:end)-LogData.time(t_cutoff);
y_act = LogData.signals(1).values(t_cutoff:end,2);
y_actm = LogData.signals(1).values(t_cutoff:end,1);
u_act = LogData.signals(2).values(t_cutoff:end,1);
```

- **Offset-Free Output Signal (Figure 1):**
  This plot compares the noise-reduced output signal (y_act) with the measured output (y_actm).

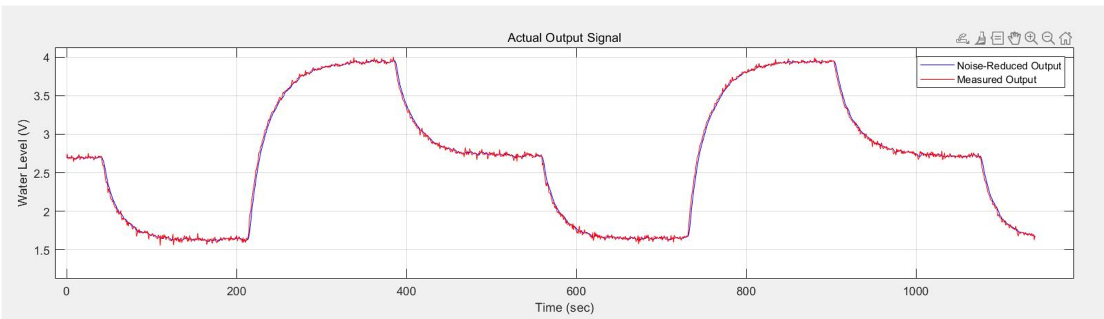

*Figure 2: Comparison of noise-reduced output signal (*y_act*) and measured output signal (*y_actm*).*

- **Offset-Free Input Signal (Figure 2):**
  This plot shows the offset-free input signal (u_act) over time.

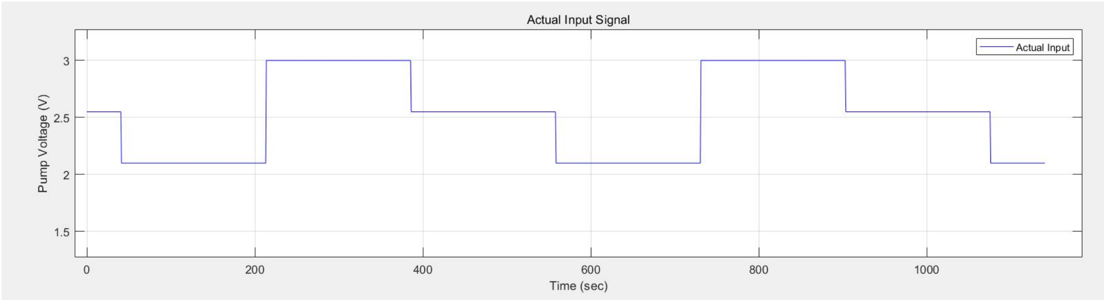

*Figure 3: Offset-free actual input signal (*u_act*).*

System Identification

Once the data is preprocessed, we proceed with system identification using the following steps:

1. Data Division: The data is divided into two parts, with the first half used for identifying the system parameters, and the second half used for model validation.

```
% Part 2 step a
N = floor(length(y) / 2); % the first half of the offset-free input-output data
```

2. Model Structure: The system is modeled as a discrete-time Linear Time-Invariant (LTI) system. We use a state-space form and transfer function to describe the system. The model is based on a second-order difference equation with parameters a_1, a_2, b_1 and b_2 which are estimated using the least squares method.

```
k_start = 3;
Y = y(k_start:N);
Phi = [y(k_start-1:N-1), y(k_start-2:N-2), u(k_start-1:N-1), u(k_start-2:N-2)];
theta = inv((Phi' * Phi)) * (Phi' * Y);
```

3. Transfer Function: A transfer function is derived from the identified parameters:

```
% transfer function
h = t(2)-t(1);
numerator = [b1 b2];
denominator = [1 a1 a2];
Gz = tf(numerator, denominator, h);% Eq1
```

4. State-Space Representation: The state-space model is formulated as follows:

```
% state space
G = [0 1; -a2 -a1];  % matrix G
H = [0; 1];  % matrix H
C = [b2 b1];  % matrix C
D = 0;  % matrix D
sys_ss = ss(G, H, C, D, h);
```

Model Verification

The identified model is validated by comparing its simulated output with the actual output. The following steps are involved in simulation:

1. **Simulation for the Second Half of Data:**

   The second half of the dataset is used for model validation.

   (1) Model Verification (Second Half) (Figure 3):

   This plot shows the comparison of the actual output with the simulated output for the second half of the data. The actual output is plotted in red, and the simulated output is plotted in blue.

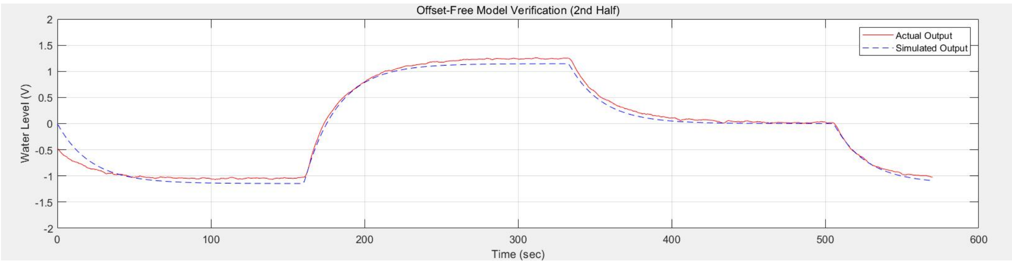

Figure 4: Actual output vs. simulated output for the second half of the dataset.

2. **Simulation for the Entire Dataset:**

   Simulation is also performed over the entire dataset to compare the actual output with the predicted output.

   (2) Model Verification (Entire Dataset) (Figure 4):

   This plot compares the actual output and the simulated output over the entire dataset.

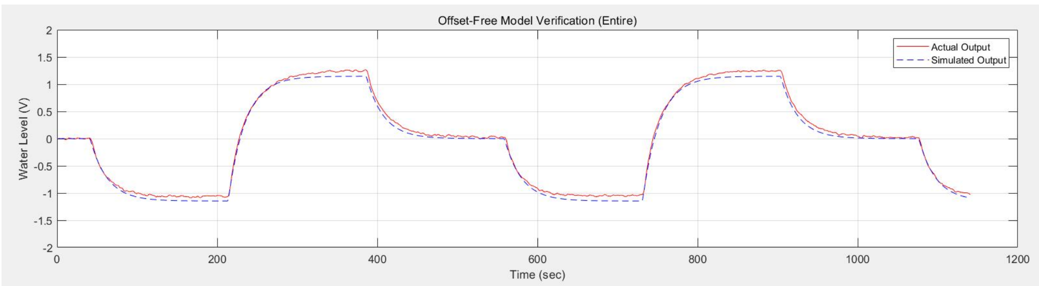

Figure 5: Actual output vs. simulated output for the entire dataset.

4. Results

4.1 Data Preprocessing

The preprocessing steps successfully removed noise and offsets from the original signals. The following plots show the raw and offset-free signals:

Actual Output Signal (Noise-Reduced):
As shown in Figure 2, the noise-reduced output signal (y_act) is compared with the measured output signal (y_actm), demonstrating the effectiveness of noise reduction.

Actual Input Signal:
As shown in Figure 3, the actual input voltage (u_act) over time is displayed.

4.2 System Identification

The identified transfer function and state-space model were successfully derived. The transfer function representing the system is:

$$G(z) = \frac{b_1 z + b_2}{z^2 + a_1 z + a_2}$$

where a_1, a_2, b_1 and b_2 are the estimated parameters.

4.3 Model Verification

The simulated output based on the identified model was compared with the actual output. The results indicate that the model's response closely matches the actual system behavior, confirming the validity of the identified model.

Figures 4 and 5 show the comparison between actual and simulated outputs for both the second half and the entire dataset, verifying the model's accuracy.

5. Discussion

Since the **Repeating Sequence Stair** block outputs zero during its first period, the initial input voltage in the simulation is zero. However, the actual system's initial voltage is u_{offset}, which is a segment we extracted from the actual waveform and is not starting from zero. Therefore, as shown in **Figure 4**, the simulated and actual outputs do not match at the initial moment.

6. Conclusion

This experiment demonstrated the process of system identification using input-output data. We successfully preprocessed the data, identified the system parameters, and derived both a transfer function and a state-space model. The model was validated through simulation, showing good agreement with the actual system response.