UNIVERSITY OF MARYLAND BALTIMORE COUNTY(UMBC)

LINEAR QUADRATIC STATE ESTIMATION

SREE BHARGAV BALUSU

May 9, 2024

1 MODEL DESCRIPTION

1.1 LINEAR DYNAMICAL SYSTEM

The focus of our study is on a linear dynamical system, which is a fundamental model for understanding and predicting the behavior of dynamic systems. The model is described by two primary equations:

1.2 NO INPUT CASE

1. **State Equation:**
$$x_{t+1} = A_t x_t + B_t w_t$$
2. **Measurement Equation:**
$$y_t = C_t x_t + v_t$$

where:

- $\mathbf{x}_t$ is the state at time t, represented as an n-vector.
- $\mathbf{y}_t$ is the measurement at time t, represented as a p-vector.
- $\mathbf{w}_t$ is the input or process noise at time t, represented as an m-vector, affecting the state evolution.
- $\mathbf{v}_t$ is the measurement noise or measurement residual at time t, represented as a p-vector, affecting the measurements.

- The system dynamics are governed by the matrices $\mathbf{A}_t$, $\mathbf{B}_t$, and $\mathbf{C}_t$, with measurements $\mathbf{y}_1, \ldots, \mathbf{y}_T$ being known.
- The noises $\mathbf{w}_t$ and $\mathbf{v}_t$ are unknown but assumed to be small.

1.3 INPUT CASE

1. **State Equation:**

$$x_{t+1} = A_t x_t + G_t u_t + B_t w_t$$

2. **Measurement Equation:**

$$y_t = C_t x_t + v_t$$

- $\mathbf{x}_t$ is the state at time t, represented as an n-vector.
- $\mathbf{y}_t$ is the measurement at time t, represented as a p-vector.
- $\mathbf{u}_t$ is the control input at time t,is part of the model where you can influence the system's state by adjusting .
- $\mathbf{w}_t$ is the input or process noise at time t, represented as an m-vector, affecting the state evolution.
- $\mathbf{v}_t$ is the measurement noise or measurement residual at time t, represented as a p-vector, affecting the measurements.
- The system dynamics are governed by the matrices $\mathbf{A}_t$, $\mathbf{B}_t$, and $\mathbf{C}_t$, with measurements $\mathbf{y}_1, \ldots, \mathbf{y}_T$ being known.
- The noises $\mathbf{w}_t$ and $\mathbf{v}_t$ are unknown but assumed to be small.

1.4 SIGNIFICANCE OF THE MODEL

This linear dynamical system model captures the essence of many real-world phenomena, allowing for the prediction and estimation of system states over time. The presence of noise, w_t and v_t, reflects the uncertainty inherent in real-world processes and observations. Estimating the system's states x_t given noisy observations y_t is crucial for various applications, from navigation and tracking to economic forecasting and beyond.

1.5 CHALLENGES IN STATE ESTIMATION

The primary challenge in state estimation lies in dealing with the uncertainties introduced by the process and measurement noise. Accurately estimating the states x_t from noisy measurements y_t requires sophisticated filtering techniques that can effectively distinguish the signal (true state) from the noise.

1.6 APPROACH

We made this in two phases; they are:

1. Data generation
2. Estimation

1.7 DATA GENERATION

- Initial state x_0 is a state vector with one sample from a multivariate normal distribution with zero mean and covariance matrix Q.
- v_t: Create a list of T vectors from a multivariate normal distribution with zero mean and covariance matrix R, represents observation noise over T (the steps).
- w_t: Generate a list of T vectors where each vector is drawn from a multivariate normal distribution with zero mean and covariance matrix P, it is process noise over T.
- u_t:is the control input at time t,is part of the model where you can influence the system's state by adjusting .

1.8 NO INPUT CASE

The matrices A_t, B_t, and C_t are taken as follow:

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Now the dynamic system model will be:

$$\begin{aligned} x_{t+1} &= Ax_t + Bw_t \\ y_t &= Cx_t + v_t, \quad \text{for } t = 1, 2, \ldots \end{aligned}$$

Calculate the next state x_t and calculate the measurements y_t

Now, we have x_t from $t = 0$ to $T - 1$ and y_t from $t = 0$ to $T - 1$.

1.9 INPUT CASE

Every other things are same only change is the control input and G. The control input u in our system is a sinusoidal input, specifically designed to simulate oscillatory behavior. Concurrently, G is defined as a random matrix with a shape of $(4, 1)$. This matrix is essential for introducing stochastic elements into the system dynamics, which helps in analyzing the system's robustness and response under varied conditions.

MATHEMATICAL REPRESENTATION The sinusoidal control input u and the random matrix G are defined as follows:

- The sinusoidal input u is mathematically formulated to represent periodic oscillations which are critical for certain testing and operational scenarios of the system.
- The matrix G, with its dimensions 4×1, consists of elements drawn from a normal distribution, reflecting the stochastic nature of the system's external influences.

$$u = A \sin(2\pi f t + \phi)$$

where:

- A, f, and ϕ represent the amplitude, frequency, and phase of the sinusoidal input, respectively.

1.10 ESTIMATING STATE(NO INPUT CASE)

The main goal of this is to minimize:

$$J_{\text{meas}} + \lambda J_{\text{proc}}$$

Subject to:

$$x_{t+1} = A_t x_t + B_t w_t, \quad t = 1, \ldots, T-1$$

Variables:

- States $x_1, \ldots, x_T$
- Input noise $w_1, \ldots, w_{T-1}$

Primary objective J_{meas} is the sum of squares of measurement residuals:

$$J_{\text{meas}} = \|C_1 x_1 - y_1\|^2 + \cdots + \|C_T x_T - y_T\|^2$$

Secondary objective J_{proc} is the sum of squares of process noise:

$$J_{\text{proc}} = \|w_1\|^2 + \cdots + \|w_{T-1}\|^2$$

Where $\lambda > 0$ is a parameter that trades off measurement and process errors.

1.11 ESTIMATING STATE(INPUT CASE)

The main goal of this is to minimize:

$$J_{\text{meas}} + \lambda J_{\text{proc}}$$

Subject to:

$$x_{t+1} = Ax_t + Gu_t + Bw_t, \quad t = 1, \dots, T-1$$

Variables:

- States $x_1, \dots, x_T$
- Input noise $w_1, \dots, w_{T-1}$

Primary objective J_{meas} is the sum of squares of measurement residuals:

$$J_{\text{meas}} = \|C_1 x_1 - y_1\|^2 + \cdots + \|C_T x_T - y_T\|^2$$

Secondary objective J_{proc} is the sum of squares of process noise:

$$J_{\text{proc}} = \|w_1\|^2 + \cdots + \|w_{T-1}\|^2$$

Where $\lambda > 0$ is a parameter that trades off measurement and process errors.

CONSTRAINED LEAST SQUARES FORMULATION(NO INPUT CASE)

Here comes the constrained least square formulation which helps to reduce both the process and the measurement noise. Minimize:

$$\|C_1 x_1 - y_1\|^2 + \cdots + \|C_T x_T - y_T\|^2 + \lambda(\|w_1\|^2 + \cdots + \|w_{T-1}\|^2)$$

Subject to:

$$x_{t+1} = A_t x_t + B_t w_t, \quad t = 1, \dots, T-1$$

- can be written as

$$\begin{aligned} &\text{minimize } \|\tilde{A}z - \tilde{b}\|^2 \\ &\text{subject to } \tilde{C}z = \tilde{d} \end{aligned}$$

- vector z contains the $Tn + (T-1)m$ variables:

$$z = (x_1, \dots, x_T, w_1, \dots, w_{T-1})$$

The matrices

$$at = \begin{pmatrix} ct \\ 0 \end{pmatrix}$$

$$\tilde{A} = \begin{bmatrix} a_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_T & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \sqrt{\lambda I} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \sqrt{\lambda I} \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\tilde{C} = \begin{bmatrix} A_1 & -I & 0 & \cdots & 0 & 0 & B_1 & 0 & \cdots & 0 \\ 0 & A_2 & -I & \cdots & 0 & 0 & 0 & B_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{T-1} & -I & 0 & 0 & \cdots & B_{T-1} \end{bmatrix}, \quad \tilde{d} = 0$$

2 CONSTRAINED LEAST SQUARES FORMULATION(INPUT CASE)

Here comes the constrained least square formulation which helps to reduce both the process and the measurement noise. Minimize:

$$\|C_1 x_1 - y_1\|^2 + \cdots + \|C_T x_T - y_T\|^2 + \lambda(\|w_1\|^2 + \cdots + \|w_{T-1}\|^2)$$

Subject to:

$$x_{t+1} = Ax_t + Gu_t + Bw_t, \quad t = 1, \ldots, T-1$$

- can be written as

$$\begin{aligned} &\text{minimize } \|\tilde{A}z - \tilde{b}\|^2 \\ &\text{subject to } \tilde{C}z = \tilde{d} \end{aligned}$$

- vector z contains the $Tn + (T-1)m$ variables:

$$z = (x_1, \ldots, x_T, w_1, \ldots, w_{T-1})$$

The matrices

$$\tilde{A} = \begin{bmatrix} a_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_T & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \sqrt{\lambda I} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \sqrt{\lambda I} \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$
\tilde{C} = \left[\begin{array}{cccccc|cccc|cccc}
A_1 & -I & 0 & \cdots & 0 & 0 & G_1 & 0 & \cdots & 0 & B_1 & 0 & \cdots & 0 \\
0 & A_2 & -I & \cdots & 0 & 0 & 0 & G_2 & \cdots & 0 & 0 & B_2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & A_{T-1} & -I & 0 & 0 & \cdots & G_{T-1} & 0 & 0 & \cdots & B_{T-1}
\end{array}\right]
$$

This time Ctilde consists of Three blocks everthing same as before except the block that consists G

2.1 FINDING ESTIMATES OF NEXT SATE AND THE MEASUREMENTS

Extract the $\hat{x}$ from z.

Now compute $\hat{y}$ which is the estimate of measurement.

$\hat{x}$ – It is the estimate of state.

$\hat{y}$ – Estimate of measurement.

$$\hat{y} = C\hat{x} + v$$

We have state and its estimates ,also measurements and its estimates

2.2 CROSS CHECKING

The matrix $\tilde{A}$ is a block matrix combining the matrices C_1 to C_T, and λI for the control inputs.

The vector $\tilde{b}$ is a vector containing the desired outputs y_1 to y_T, followed by zeros.

The vector $\tilde{z}$ is the vector containing the state and control variables $(x_1, \ldots, x_T, w_1, \ldots, w_{T-1})$.

The matrix-vector multiplication $\tilde{A}\tilde{z}$ involves multiplying each block of $\tilde{A}$ by the corresponding elements of $\tilde{z}$. Since $\tilde{A}$ has a block-diagonal structure with C_t blocks and λI blocks, and since $\tilde{b}$ is simply the vector of y_t values followed by zeros, the multiplication and subsequent subtraction will result in a vector of residuals as the image depicts.

The product $\tilde{A}\tilde{z}$ would multiply each block C_t with the corresponding state x_t, and each $\sqrt{\lambda I}$ with the corresponding control w_t. Subtracting $\tilde{b}$ from this product gives us the residuals:

$$
\tilde{A}\tilde{z} - \tilde{b} = \begin{bmatrix} C_1 x_1 \\ \vdots \\ C_T x_T \\ \sqrt{\lambda} w_1 \\ \vdots \\ \sqrt{\lambda} w_{T-1} \end{bmatrix} - \begin{bmatrix} y_1 \\ \vdots \\ y_T \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} C_1 x_1 - y_1 \\ \vdots \\ C_T x_T - y_T \\ \sqrt{\lambda} w_1 \\ \vdots \\ \sqrt{\lambda} w_{T-1} \end{bmatrix}
$$

This results in a vector where each element is the residual at each time step t, both for the output mismatch $v(t) = C_t x_t - y_t$ and the weighted control input $w(t)\sqrt{\lambda}$. This is exactly what's depicted in the image you've uploaded, confirming that $\tilde{A}\tilde{z} - \tilde{b}$ gives the residuals as a stack of the measurement residuals $v(t)$ followed by the process noise residuals $w(t)\sqrt{\lambda}$ for for each time step.

PLOTTING

1. Plot the graph which consists of both state (x) and its estimates ($\hat{x}$) from the stamp 0 to $T-1$.

2. Plot the graph for which consists of both measured (y) and the estimates ($\hat{y}$) with respect to the stamp 0 to $T-1$.

3. Plot the graph for $(\tilde{A}\tilde{z}-\tilde{b})$ half of the value of it with respect to t from 0 to $T-1$.

4. Plot the graph for $(\tilde{C}\tilde{z}-\tilde{d})=0$. It should be a horizontal line whose values are zero.

3 Methodology of K-Fold Cross-Validation and Hyperparameter Tuning

3.1 K-Fold Cross-Validation

K-fold cross-validation involves dividing the dataset into k equally sized subsets. In each iteration, one subset is used as the test set and the remaining $k-1$ subsets are used as the training set. This process helps in validating the model effectively across different subsets of the dataset.

3.2 Hyperparameter Tuning

Hyperparameter tuning is performed to optimize the lambda parameter of the LQE model. Different values are tested to determine which configuration yields the lowest reconstruction error.

4 Implementation Details

- **Data Partitioning:** The dataset is randomly partitioned into k subsets, ensuring that each subset is used once as the test set.

- **Shuffling and Random Seed:** Data shuffling is enabled with a random seed set to ensure reproducibility.

- **Loop Through Folds:** The process iterates over each fold, training the model on $k-1$ folds and testing it on the remaining fold.

- **Lambda Hyperparameter:** A range of lambda values are tested to find the optimal setting that minimizes the error.

5 Results

1. **Reconstruction Error:** The reconstruction error is calculated for each lambda setting across all folds.

2. **Optimal Lambda Selection:** The lambda that results in the lowest average error across all k-folds is selected as the optimal parameter.

6 CONCLUSION

The use of K-fold cross-validation combined with hyperparameter tuning provides a robust framework for optimizing and validating the LQE model, enhancing its performance and reliability on unseen data.

7 RESULTS(NO INPUT AND INPUT CASE)

7.1 NO INPUT CASE

- The state vector $\mathbf{x}_t$ represents the position and velocity of the mass at time t:

$$\mathbf{x}_t = \begin{bmatrix} p_t \\ z_t \end{bmatrix}$$

  where p_t is the 2-vector of position and z_t is the 2-vector of velocity.

- The state evolution is governed by the state matrix A_t, control matrix B_t, and the output matrix C_t, defined as:

$$A_t = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

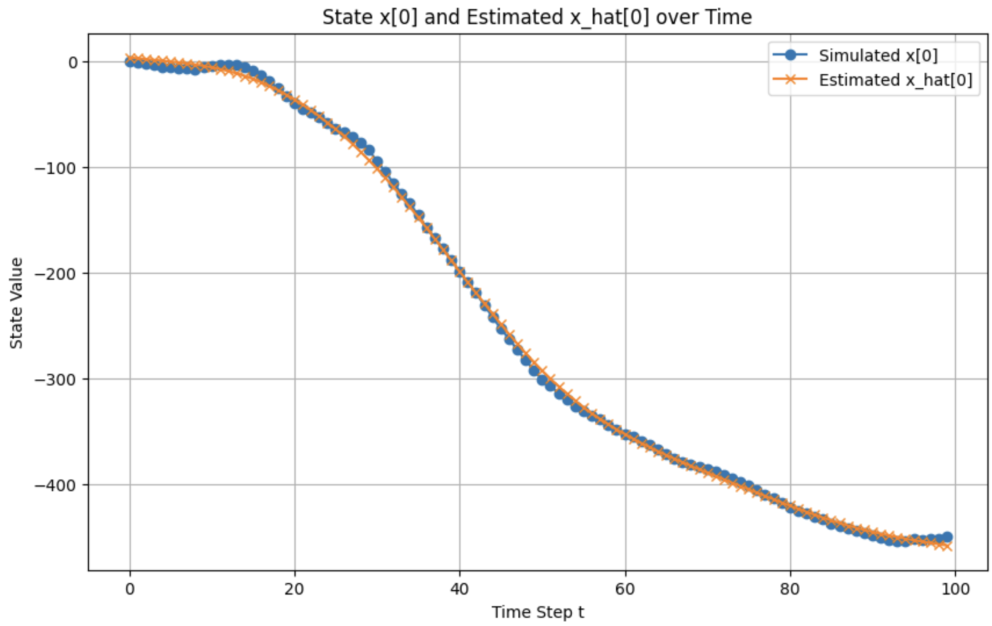
- The output $\mathbf{y}_t$ at time t is a noisy measurement of the position:

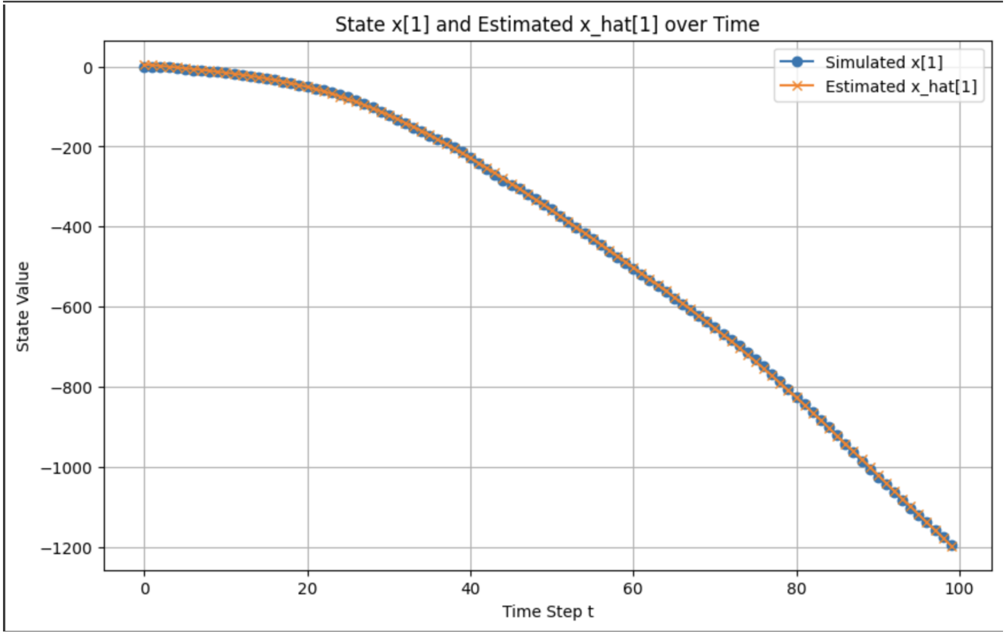
$$\mathbf{y}_t = C_t \mathbf{x}_t + \mathbf{w}_t$$

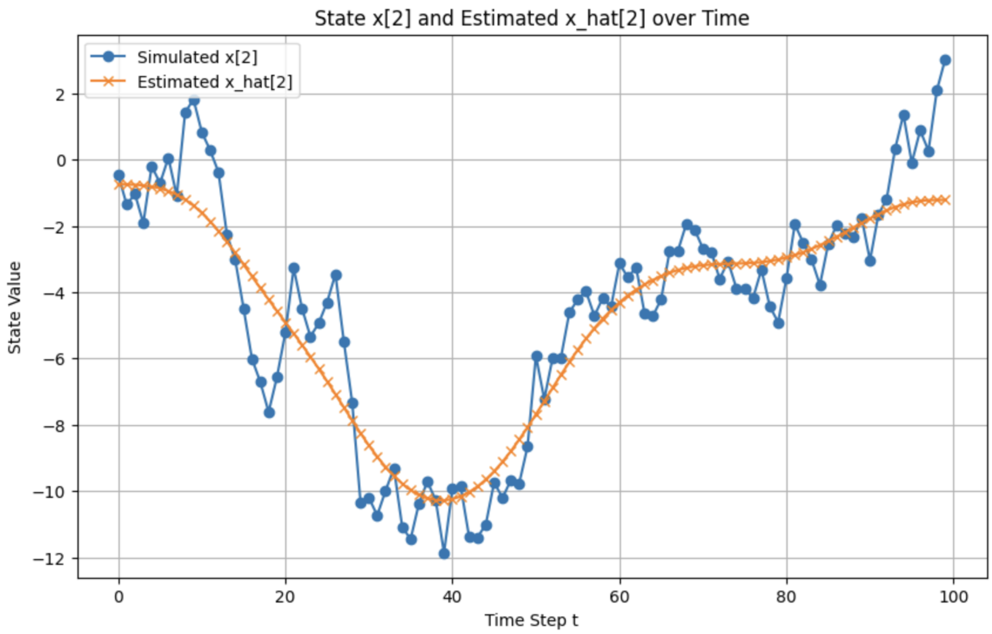
  where $\mathbf{w}_t$ is the measurement noise.

The simulation runs over a time horizon $T = 100$.

Figure 7.1: Below are the results of state and its estimates with respect to time series






State x[2] and Estimated x_hat[2] over Time
Simulated x[2]
Estimated x_hat[2]
State Value
Time Step t
2
0
−2
−4
−6
−8
−10
−12
0
20
40
60
80
100

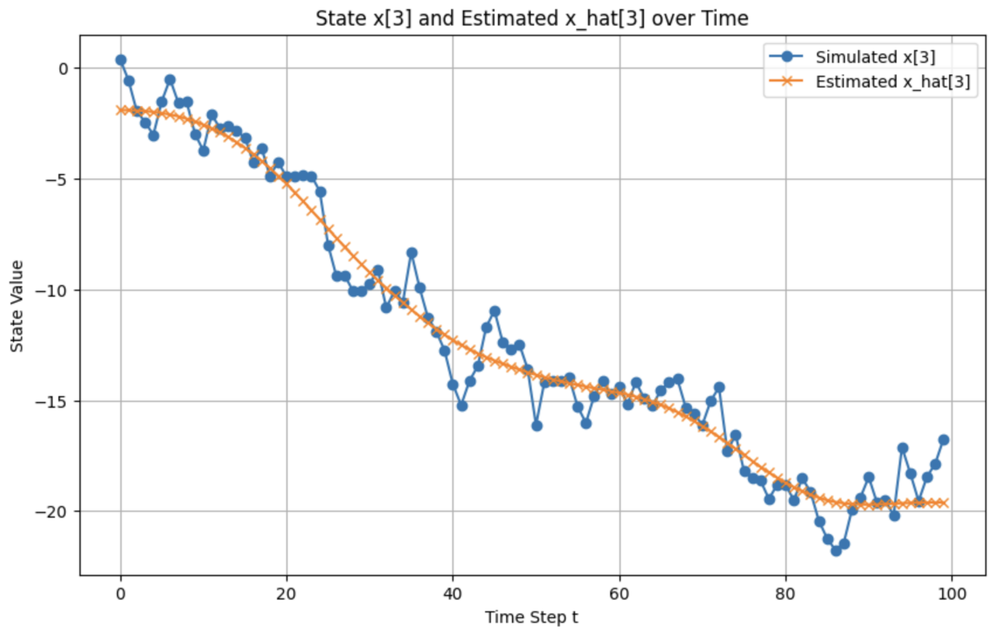
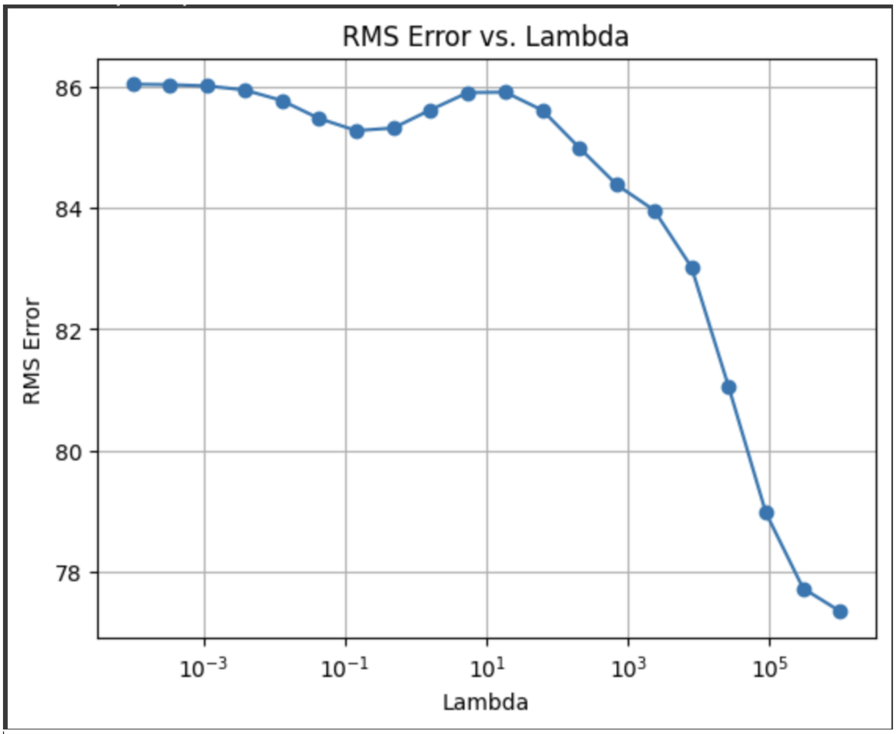

State x[3] and Estimated x_hat[3] over Time
Simulated x[3]
Estimated x_hat[3]
State Value
Time Step t
0
−5
−10
−15
−20
0
20
40
60
80
100

Figure 7.2: Below are the results of Cross validation with fixed split and a random split



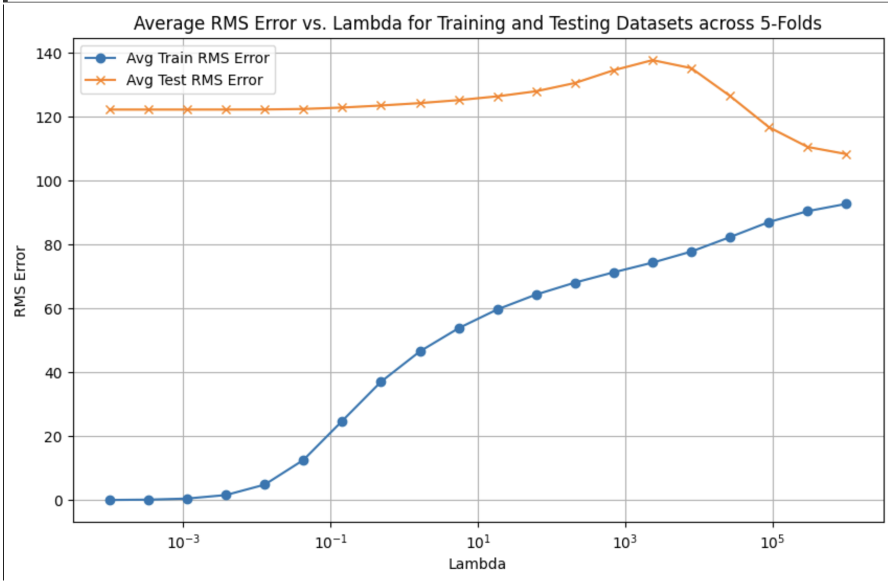
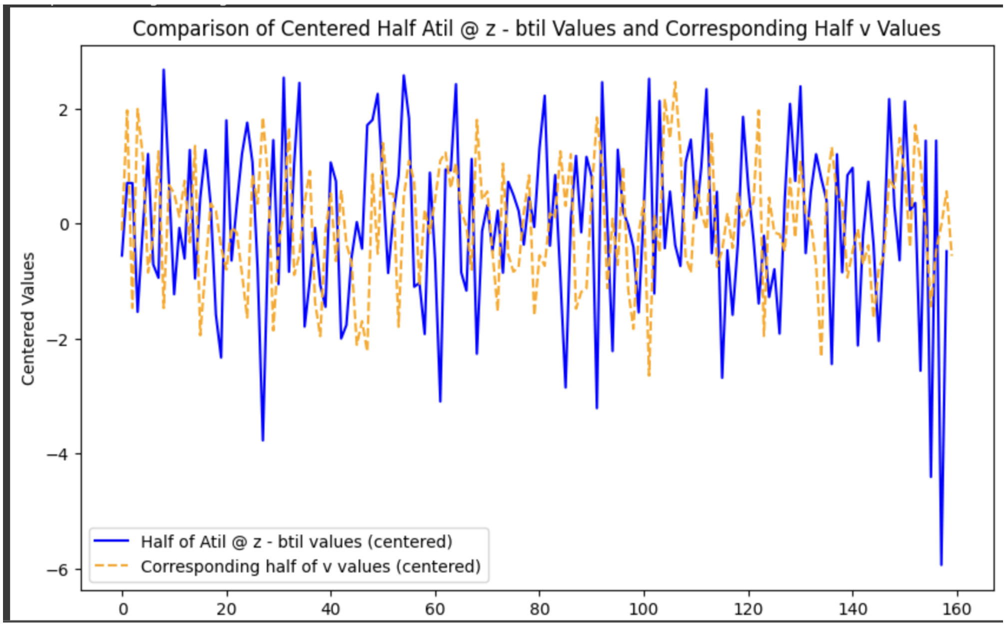


Figure 7.3: Based on preliminary simulations and theoretical considerations, it is suggested to set the value of λ to approximately 1000. This choice of λ ensures a good compromise between accuracy and the robustness of the model. Specifically, $\lambda \approx 10^3$ provides a suitable level of regularization to manage measurement noise effectively while preventing the over-amplification of control inputs.

Figure 7.4: Below are the results of cross checking



8 INPUT CASE

- The state vector $\mathbf{x}_t$ represents the position and velocity of the mass at time t:

$$\mathbf{x}_t = \begin{bmatrix} p_t \\ z_t \end{bmatrix}$$

  where p_t is the 2-vector of position and z_t is the 2-vector of velocity.

- The state evolution is governed by the state matrix A_t, control matrix B_t, and the output matrix C_t, defined as:

$$A_t = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad G_t = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix}$$

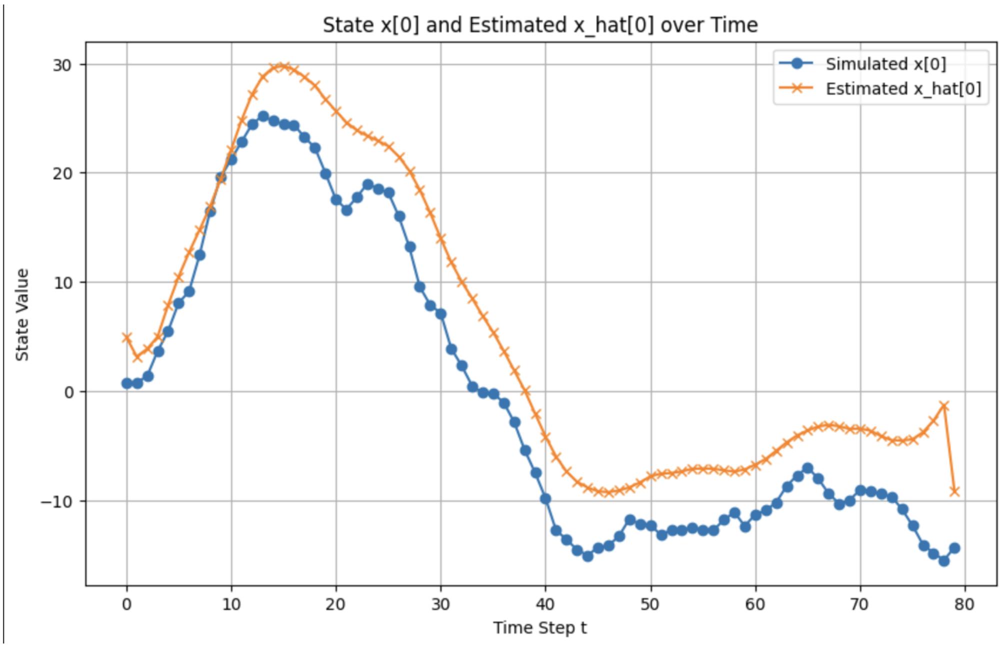
- The output $\mathbf{y}_t$ at time t is a noisy measurement of the position:

$$\mathbf{y}_t = C_t \mathbf{x}_t + \mathbf{w}_t$$

  where $\mathbf{w}_t$ is the measurement noise.

$\lambda = 10$ Regularization parameter. The simulation runs over a time horizon $T = 100$.
Cross Checking

Figure 8.1: Below are the results of state and its estimates with respect to time series



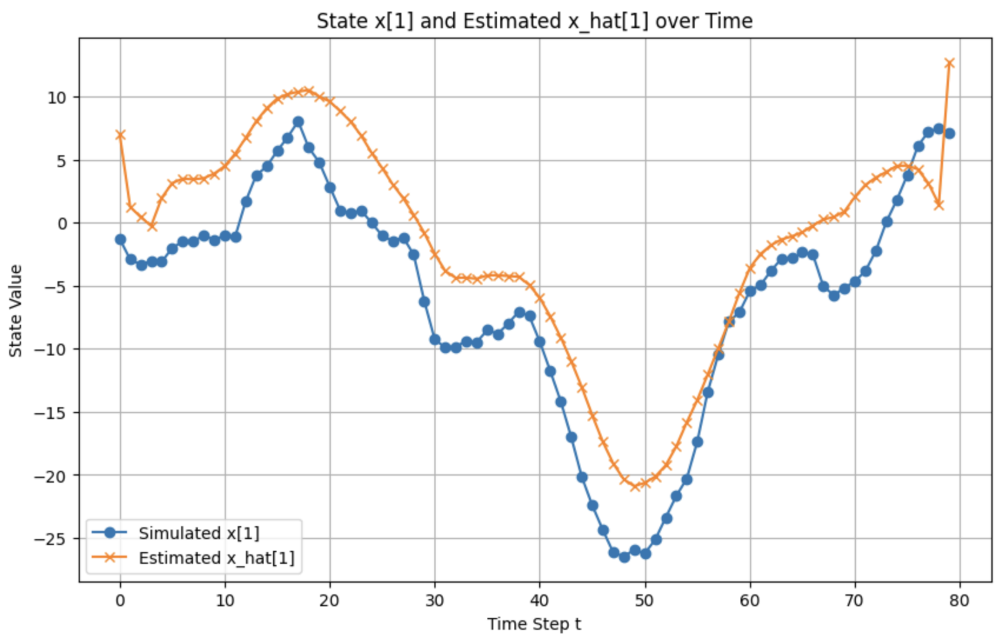
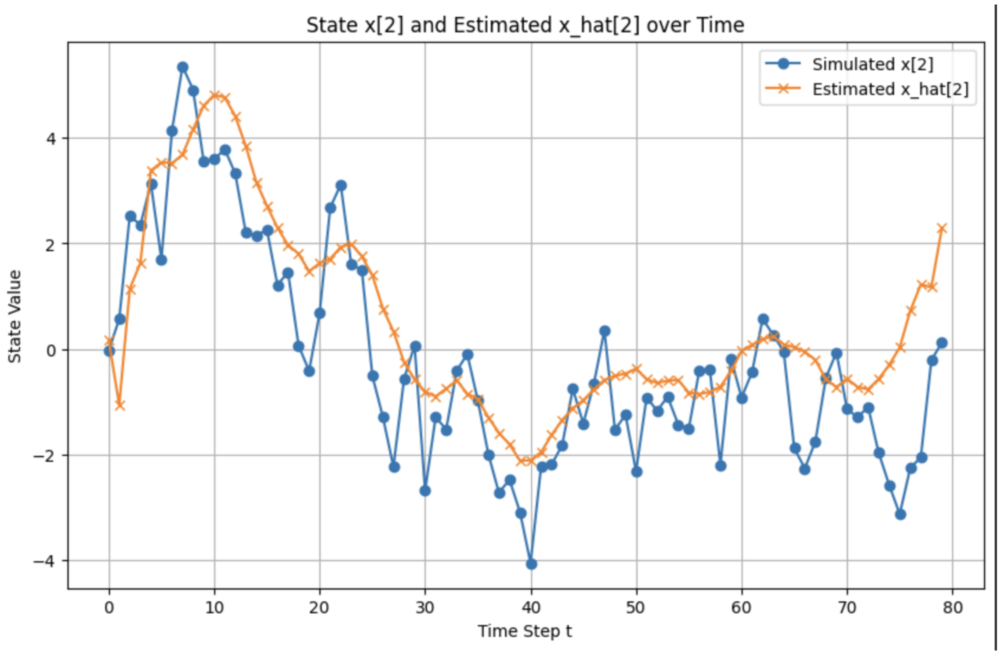


Figure 8.2: Below are the results of state and its estimates with respect to time series



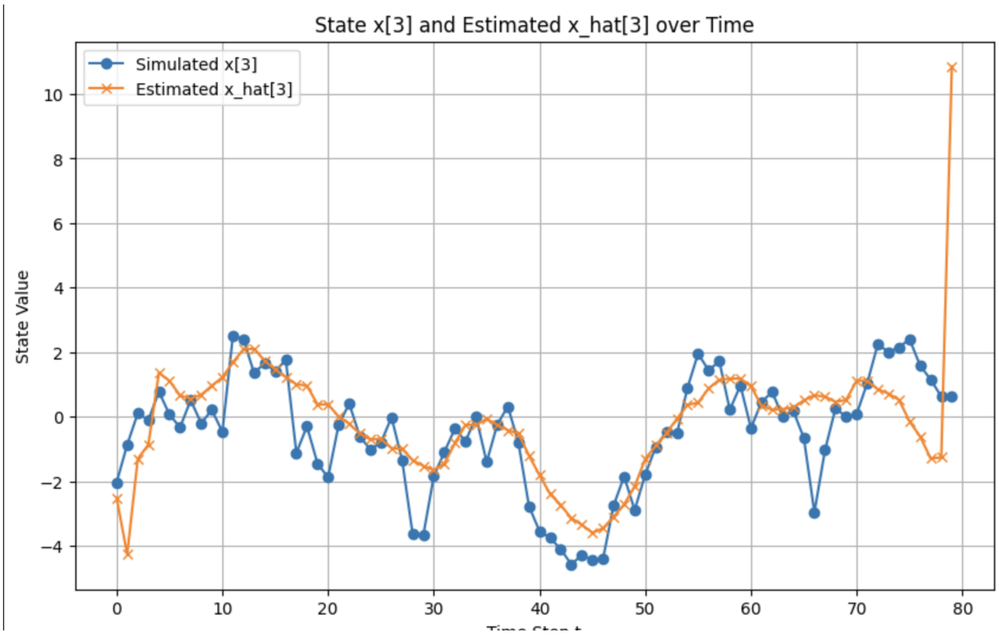


Figure 8.3: Below are the results of cross checking

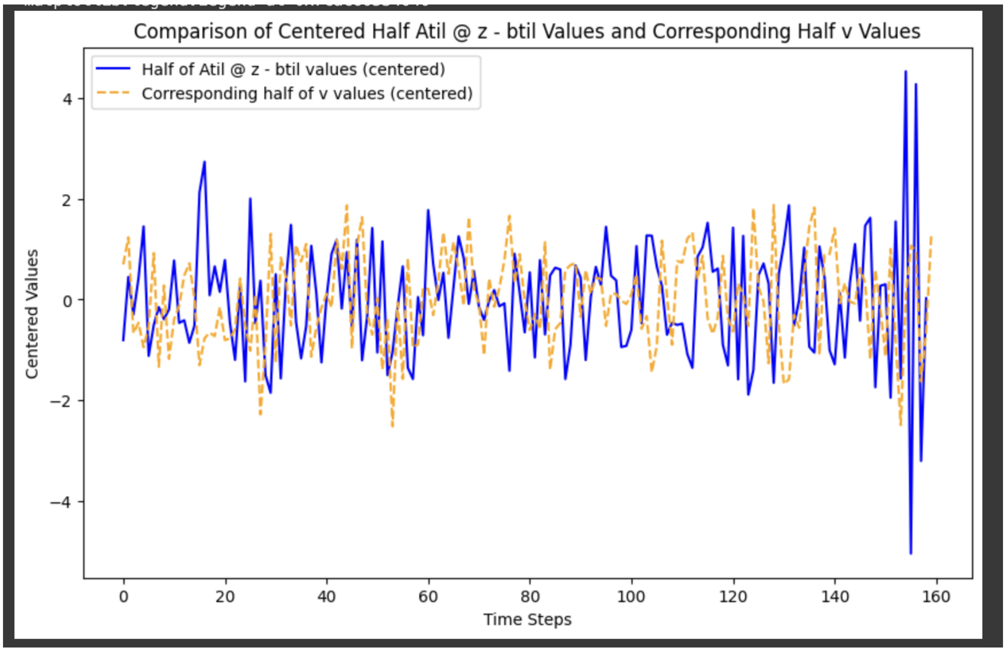
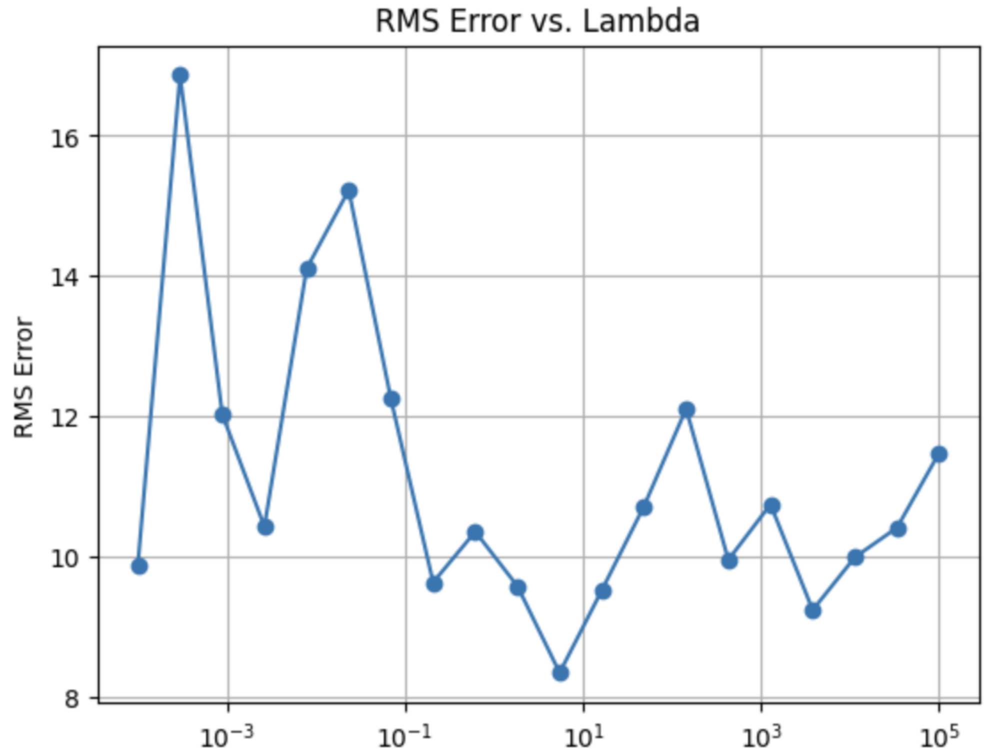


Figure 8.4: Below are the results of Cross validation with fixed split and a random split



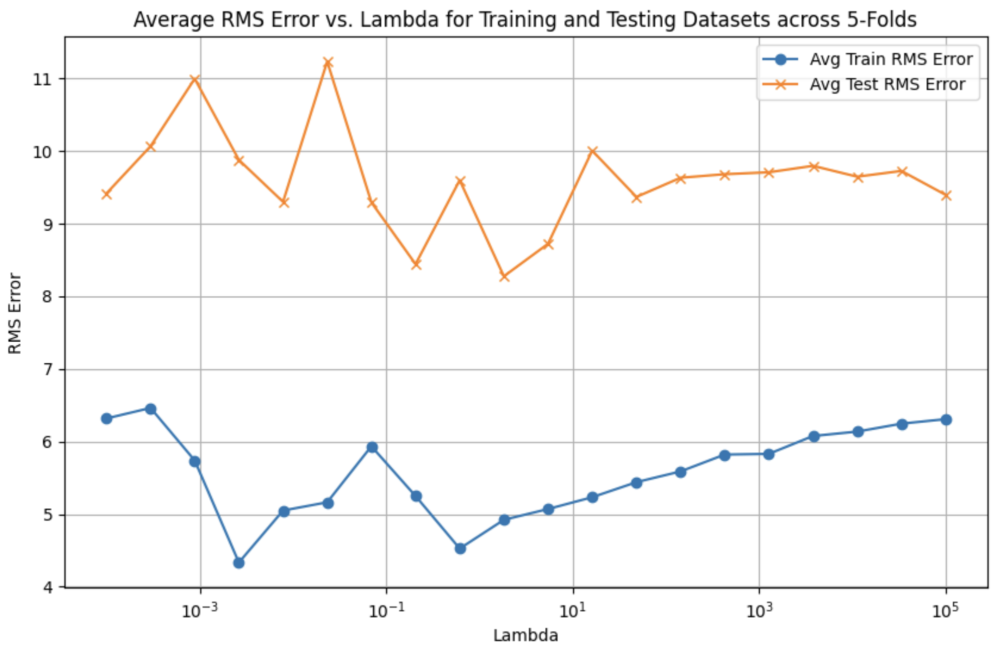


Figure 8.5: Based on preliminary simulations and theoretical considerations, it is suggested to set the value of λ to approximately 10. This choice of λ ensures a good compromise between accuracy and the robustness of the model. Specifically, $\lambda \approx 10$ provides a suitable level of regularization to manage measurement noise effectively while preventing the over-amplification of control inputs.

Figure 8.6: Below is the plot for control input u and its estimate over time

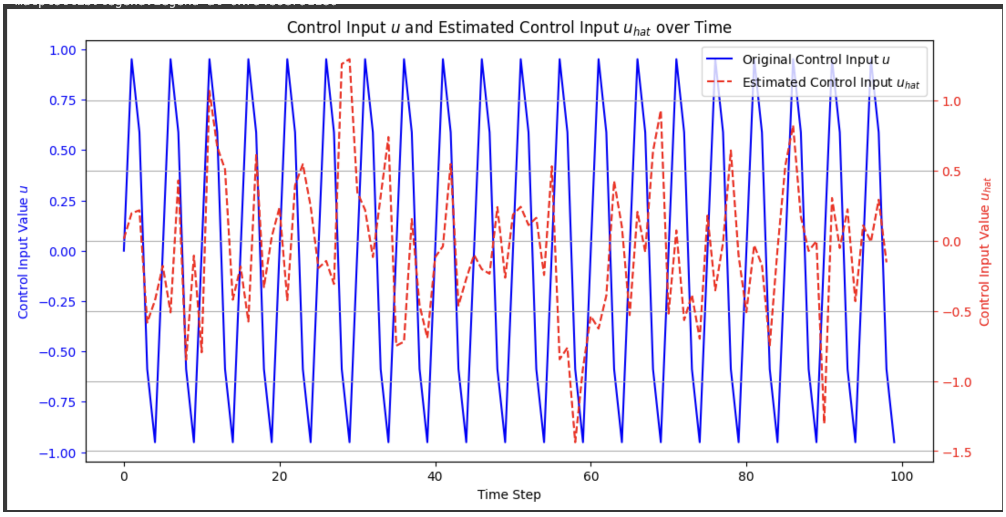


Figure 8.7: The plot contrasts the original control input u shown in solid blue with its estimated counterpart estimates in dashed red, highlighting the estimation's attempt to follow the sinusoidal pattern amidst visible deviations.

9 TWO INPUT CASE

- The state vector $\mathbf{x}_t$ represents the position and velocity of the mass at time t:

$$\mathbf{x}_t = \begin{bmatrix} p_t \\ z_t \end{bmatrix}$$

  where p_t is the 2-vector of position and z_t is the 2-vector of velocity.

- The state evolution is governed by the state matrix A_t, control matrix B_t, and the output matrix C_t, defined as:

$$A_t = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_t = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad G_t = \begin{bmatrix} g_1 & h_1 \\ g_2 & h_2 \\ g_3 & h_3 \\ g_4 & h_4 \end{bmatrix}$$

- The output $\mathbf{y}_t$ at time t is a noisy measurement of the position:

$$\mathbf{y}_t = C_t\mathbf{x}_t + \mathbf{w}_t$$

  $\lambda = 10^3$ Regularization parameter

The simulation runs over a time horizon $T = 100$.

Figure 9.1: Below are the results of state and its estimates with respect to time series

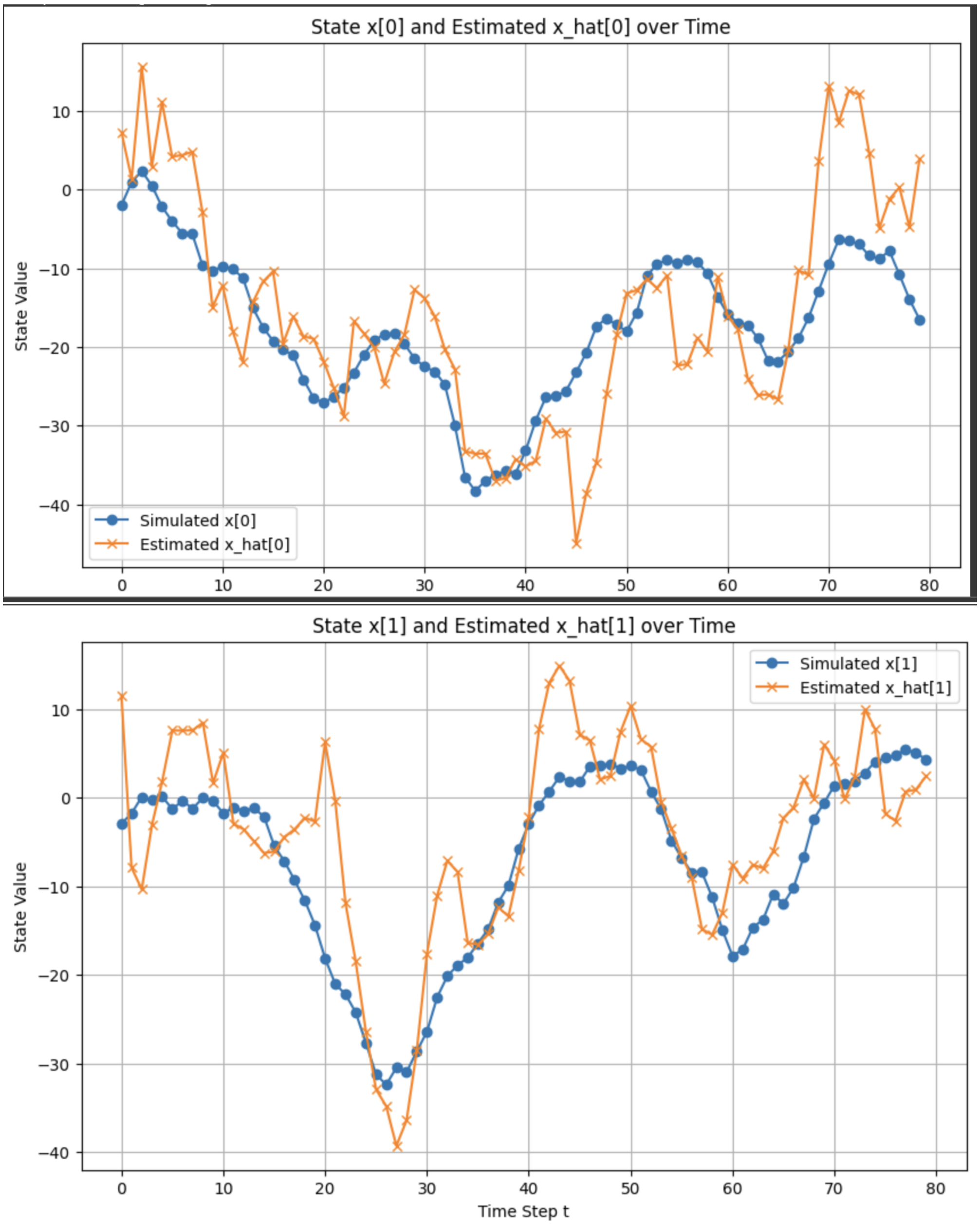


Figure 9.2: Below are the results of state and its estimates with respect to time series

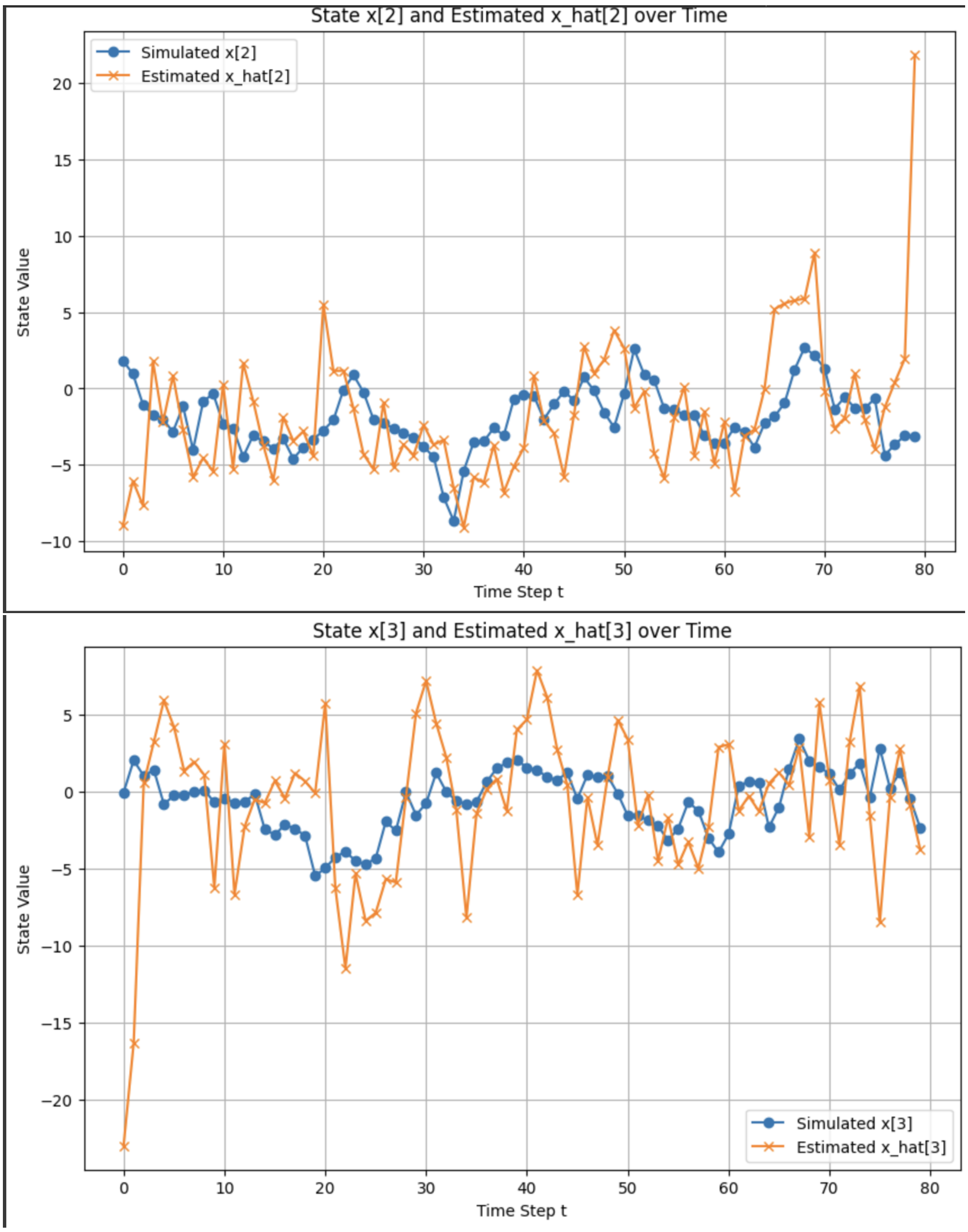


Figure 9.3: Below are the results of Cross validation with fixed split and a random split

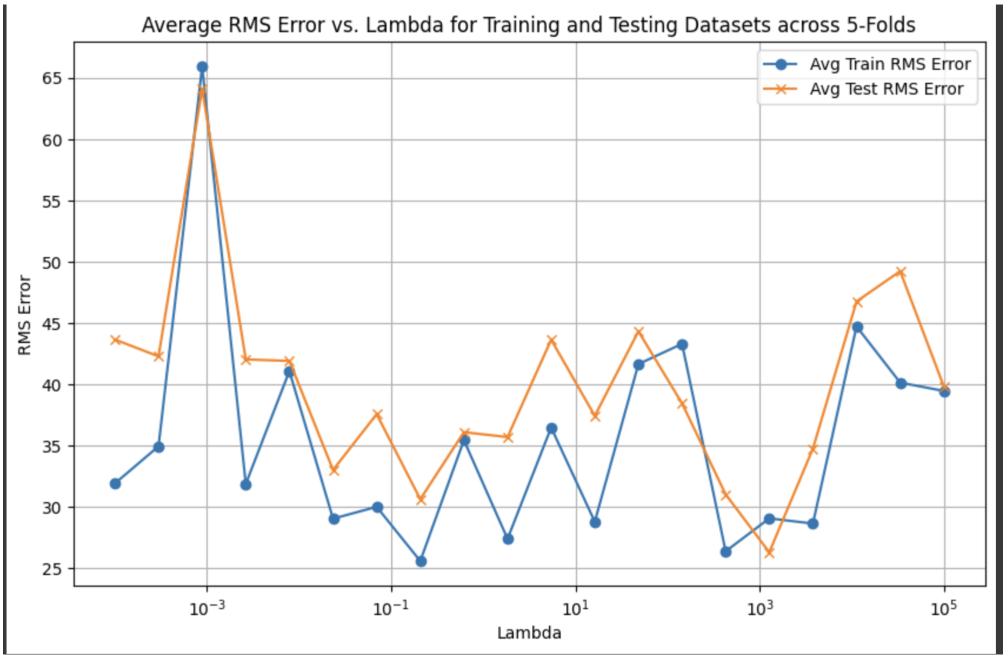


Figure 9.4: Based on preliminary simulations and theoretical considerations, it is suggested to set the value of λ to approximately 1000. This choice of λ ensures a good compromise between accuracy and the robustness of the model. Specifically, $\lambda \approx 10^3$ provides a suitable level of regularization to manage measurement noise effectively while preventing the over-amplification of control inputs.

Figure 9.5: Below is the plot for control input u and its estimate over time

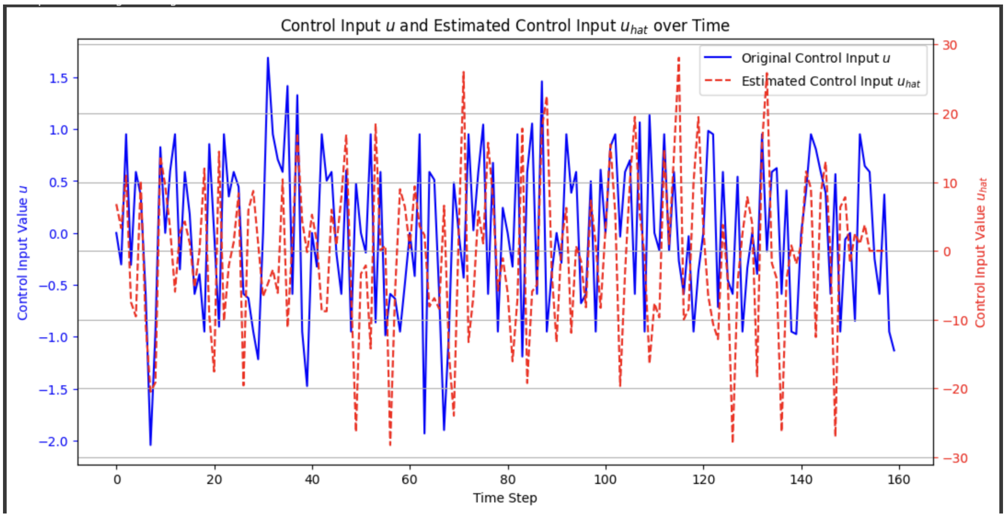


Figure 9.6: The plot contrasts the original control input u shown in solid blue with its estimated counterpart estimates in dashed red, highlighting the estimation's attempt to follow the sinusoidal pattern amidst visible deviations.