Simulation and Modeling of Dynamic Systems

Dimitrios Karatis 10775, *Electrical and Computer Engineering, AUTH*

***Abstract*—This final project addresses two major topics in real-time system identification and nonlinear modeling. In the first part, we estimate the parameters of a linear time-invariant system using a real-time estimation algorithm and analyze the effect of bounded bias disturbances. In the second part, we study structure selection and model evaluation techniques for an unknown nonlinear dynamic system, and implement real-time parameter estimation.**

***Index Terms*—real-time estimation, bias disturbance, structure selection, nonlinear modeling, model evaluation.**

I. Problem 1: Real-Time Estimation of LTI System

A. Part (a): Estimation without Bias

We consider a linear time-invariant system of the form:

$$\dot{x}(t) = Ax(t) + Bu(t)$$

where $x(t) \in \mathbb{R}^2$ is the system state, $u(t) \in \mathbb{R}$ is the control input, and $A \in \mathbb{R}^{2\times 2}, B \in \mathbb{R}^{2\times 1}$ are constant but unknown matrices.

The true system is given by:

$$A = \begin{bmatrix} -2.15 & 0.25 \\ -0.75 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1.5 \end{bmatrix}$$

We design a real-time parameter estimation algorithm to estimate $\hat{A}(t)$ and $\hat{B}(t)$, using an input $u(t)$ of our choice and measurable $x(t)$, $u(t)$. The algorithm is analyzed for stability, and simulation results are provided including:

- Estimated state $\hat{x}(t)$
- Estimation error $e_x(t) = x(t) - \hat{x}(t)$
- Time evolution of $\hat{A}(t), \hat{B}(t)$

For this part, I used the Lyapunov parallel estimator configuration, as shown below. The input signal was selected as:

$$u(t) = c\sin(d\cdot t) + f\sin(h\cdot t) + i\sin(j\cdot t),$$

in order to satisfy the condition of ***persistent excitation***, in accordance with Theorem 6.1.1 from the course's material. This theorem ensures that, under persistent excitation, the time-varying parameter estimates converge correctly. Specifically, the excitation signal takes the general form:

$$u(t) = \sum_{i=1}^{m} \alpha_i \sin(\omega_i t)$$

where the following conditions hold:

- $m \geq \frac{n}{2}$, with n being the number of parameters to be estimated,
- $\alpha_i \neq 0$ for all i,
- $\omega_i \neq \omega_j$ for all $i \neq j$.

These constraints guarantee that the regressor matrix is sufficiently rich over time to enable accurate estimation.

Parallel Configuration

Given the system:

$$\dot{x} = Ax + Bu$$

The estimated model system based on the Parallel Configuration is:

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u$$

Next we define the errors:

$$e_x = x - \hat{x}, \quad e_A = \hat{A} - A, \quad e_B = \hat{B} - B$$

$$\Rightarrow \dot{e}_x = \dot{x} - \dot{\hat{x}} = Ax + Bu - (\hat{A}\hat{x} + \hat{B}u) = (A - \hat{A})x - (\hat{B} - B)u, \quad (1)$$

$$\dot{e}_A = \dot{\hat{A}}, \quad \dot{e}_B = \dot{\hat{B}}$$

From equation (1) we add and subtract the term $A\hat{x}(t)$, so:

$$\dot{e}_x = A(x - \hat{x}) + (A - \hat{A})\hat{x} + (B - \hat{B})u = Ae_x - e_A\hat{x} - e_Bu$$

Lyapunov Function

$$V = \frac{1}{2}e_x^T e_x + \frac{1}{2\gamma_1}\text{tr}(e_A^T e_A) + \frac{1}{2\gamma_2}\text{tr}(e_B^T e_B)$$

$$\Rightarrow \dot{V} = e_x^T \dot{e}_x + \frac{1}{\gamma_1}\text{tr}(e_A^T \dot{e}_A) + \frac{1}{\gamma_2}\text{tr}(e_B^T \dot{e}_B)$$

By substituting the above expressions we have:

$$\begin{aligned}\dot{V} &= e_x^T(Ae_x - e_A\hat{x} - e_Bu) + \frac{1}{\gamma_1}\text{tr}(e_A^T\dot{\hat{A}}) + \frac{1}{\gamma_2}\text{tr}(e_B^T\dot{\hat{B}}) \\ &= e_x^T Ae_x - e_x^T e_A\hat{x} - e_x^T e_B u + \frac{1}{\gamma_1}\text{tr}(e_A^T\dot{\hat{A}}) + \frac{1}{\gamma_2}\text{tr}(e_B^T\dot{\hat{B}})\end{aligned}$$

But:

$$e_x^T e_A\hat{x} = \hat{x}^T e_A^T e_x = \text{tr}(e_A^T e_x \hat{x}^T), \quad e_x^T e_B u = u^T e_B^T e_x = \text{tr}(e_B^T e_x u^T)$$

Thus:

$$\dot{V} = e_x^T A e_x + \text{tr}\left(-e_A^T e_x\hat{x}^T - e_B^T e_x u^T + \frac{1}{\gamma_1}e_A^T\dot{\hat{A}} + \frac{1}{\gamma_2}e_B^T\dot{\hat{B}}\right)$$

Lastly, to ensure $\dot{V}$ is negative semi-definite, we choose:

$$\dot{\hat{A}} = \gamma_1 e_x\hat{x}^T, \quad \dot{\hat{B}} = \gamma_2 e_x u^T$$

Finally we get the following system:

This document was prepared by Dimitrios Karatis as part of the third assignment for the Simulation and Modeling of Dynamic Systems course at Aristotle University of Thessaloniki.

$$(\Sigma):\begin{cases}\dot{x}_1(t)=a_{11}x_1(t)+a_{12}x_2(t)+b_1u(t)\\ \dot{x}_2(t)=a_{21}x_1(t)+a_{22}x_2(t)+b_2u(t)\\ \dot{\hat{x}}_1(t)=\hat{a}_{11}\hat{x}_1(t)+\hat{a}_{12}\hat{x}_2(t)+\hat{b}_1u(t)\\ \dot{\hat{x}}_2(t)=\hat{a}_{21}\hat{x}_1(t)+\hat{a}_{22}\hat{x}_2(t)+\hat{b}_2u(t)\\ \dot{\hat{a}}_{11}(t)=\gamma_1e_1(t)\hat{x}_1(t)\\ \dot{\hat{a}}_{12}(t)=\gamma_1e_1(t)\hat{x}_2(t)\\ \dot{\hat{a}}_{21}(t)=\gamma_1e_2(t)\hat{x}_1(t)\\ \dot{\hat{a}}_{22}(t)=\gamma_1e_2(t)\hat{x}_2(t)\\ \dot{\hat{b}}_1(t)=\gamma_2e_1(t)u(t)\\ \dot{\hat{b}}_2(t)=\gamma_2e_2(t)u(t)\end{cases}$$

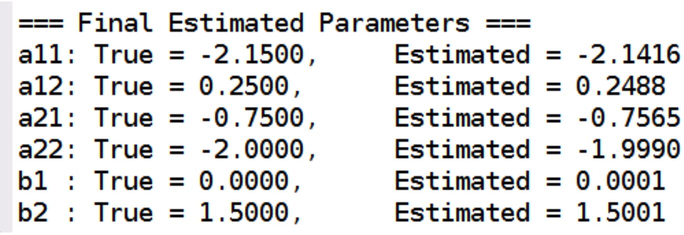
Subject to the projection constraints:

$$\begin{cases}\hat{a}_{11}(t)\in[-3,\,-1]\\ \hat{b}_2(t)\geq 1\end{cases}$$

By invoking Barbalat's Lemma, we ensure both the stability of the system and the convergence of the estimation to the true output, under the assumption of a uniformly bounded input, which we already have.

Detailed code can be found in the exercise1_a.m file.

Plot Results



```
=== Final Estimated Parameters ===
a11: True = -2.1500,     Estimated = -2.1416
a12: True = 0.2500,      Estimated = 0.2488
a21: True = -0.7500,     Estimated = -0.7565
a22: True = -2.0000,     Estimated = -1.9990
b1 : True = 0.0000,      Estimated = 0.0001
b2 : True = 1.5000,      Estimated = 1.5001
```

Fig. 1: Results with $\gamma_1 = 16$ and $\gamma_2 = 3$

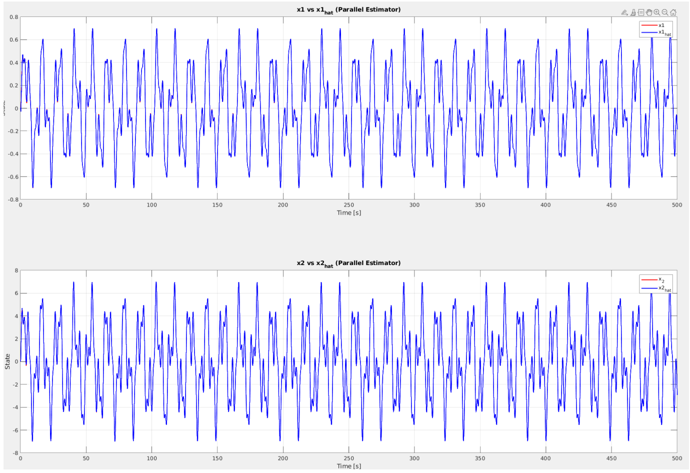


Fig. 2: $x(t)$ and $\hat{x}(t)$ for $\gamma_1 = 16$ and $\gamma_2 = 3$

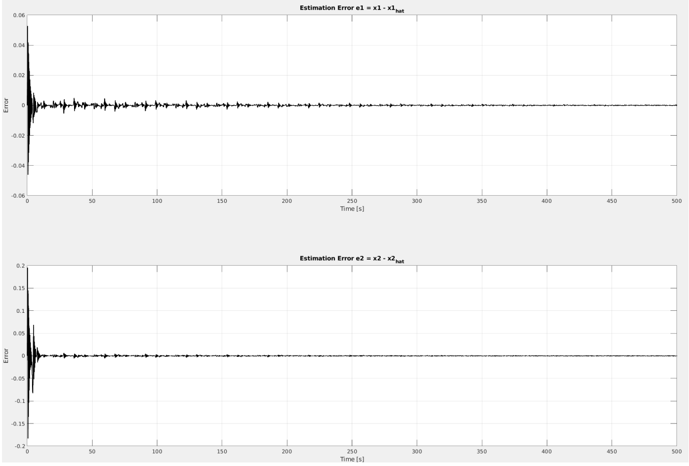


Fig. 3: Error e_x for $\gamma_1 = 16$ and $\gamma_2 = 3$

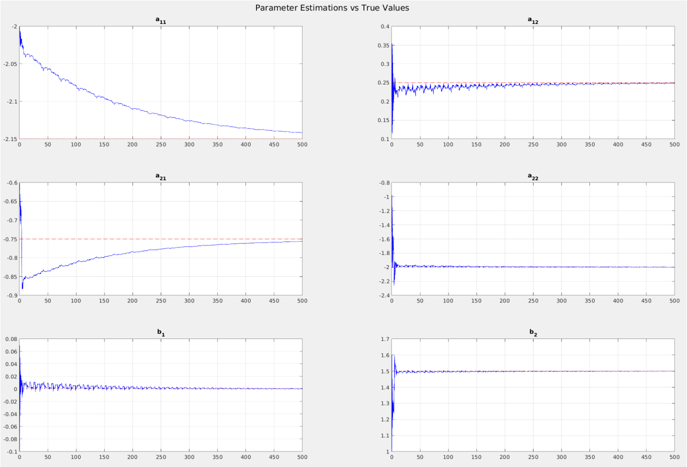


Fig. 4: Estimated parameters over time for $\gamma_1 = 16$ and $\gamma_2 = 3$

In the process of finding the optimal γ values, I tested several combinations. The best results were obtained with $\gamma = [16, 3]$ for the parallel configuration, reaching a steady state at 500 seconds.

The plots illustrate the performance of the real-time estimation algorithm for the unknown matrices A and B in the linear system $\dot{x}(t) = Ax(t) + Bu(t)$. The following observations can be made:

State Estimation ($x(t)$ vs. $\hat{x}(t)$)

- The plots of $x_1(t)$ and $\hat{x}_1(t)$ show that the estimated state $\hat{x}_1(t)$ closely tracks the true state $x_1(t)$ after an initial transient phase. Similarly, $\hat{x}_2(t)$ converges to $x_2(t)$, indicating the estimator's effectiveness.

Estimation Error ($e_x(t) = x(t) - \hat{x}(t)$)

- The error plots demonstrate that the estimation errors $e_1(t)$ and $e_2(t)$ decay to near zero over time.
- The error $e_1(t)$ exhibits faster convergence compared to $e_2(t)$, which shows larger initial deviations but stabilizes as the algorithm progresses.

Parameter Estimation ($\hat{A}(t)$ and $\hat{B}(t)$)

- The parameter estimation plot reveals that all estimates converge toward their true values (dashed lines).

Stability Analysis

- The boundedness of errors and the convergence of both states and parameters confirm the stability of the designed estimation algorithm.

B. Part (b): Parameter Estimation under Bounded Bias

In this part, we extend the estimation framework to account for the presence of a bounded unknown disturbance (bias) $\omega(t) \in \mathbb{R}^2$, such that:

$$\|\omega(t)\| \leq \bar{\omega}, \quad \forall t \geq 0$$

This disturbance acts as an additive uncertainty in the system dynamics:

$$\dot{x}(t) = Ax(t) + Bu(t) + \omega(t)$$

To ensure robust estimation in the presence of this bounded bias, I modifed the Lyapunov-based real-time estimator using a σ-modification technique. The goal is to stabilize the parameter estimation even when $\omega(t)$ causes persistent excitation to degrade over time.

Modified Adaptation Law: To mitigate parameter drift and ensure robustness, we introduce damping terms in the adaptation laws:

$$\dot{\hat{A}} = \gamma_1 e_x \hat{x}^T - \gamma_1 \sigma_\delta \hat{A}, \quad \dot{\hat{B}} = \gamma_2 e_x u^T - \gamma_2 \sigma_\delta \hat{B}$$

where Δ_σ represents the σ-modification term. This term penalizes large parameter deviations based on a saturation-like function. For each parameter θ, it is defined as:

$$\sigma_\delta(\theta) = \begin{cases} 0, & |\hat{\theta}| < M \\ \sigma\left(\frac{|\hat{\theta}|}{M} - 1\right), & M \leq |\hat{\theta}| \leq 2M \\ \sigma, & |\hat{\theta}| > 2M \end{cases}$$

Here, M is a design threshold for each parameter, and σ is a small positive constant controlling the strength of the damping, that also changes with each parameter.

Disturbance Model: The disturbance $\omega(t)$ is modeled as a smooth, bounded signal such as:

$$\omega(t) = \begin{bmatrix} \bar{\omega}_A \\ \bar{\omega}_B \cdot \sin(ht) \end{bmatrix}$$

with $\bar{\omega}_A, \bar{\omega}_B$ set to known bounds during experimentation.

System Overview: The full Lyapunov Parallel estimator with σ-modification and bounded bias is given as:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + \omega(t) \\ \dot{\hat{x}}(t) = \hat{A}\hat{x}(t) + \hat{B}u(t) \\ \dot{\hat{A}} = \gamma_1 e_x \hat{x}^T - \gamma_1 \sigma_\delta \hat{A} \\ \dot{\hat{B}} = \gamma_2 e_x u^T - \gamma_2 \sigma_\delta \hat{B} \end{cases}$$

with $e_x = x - \hat{x}$. Projection constraints from Part (a) should still be enforced.

Simulation Setup

In the MATLAB implementation, I defined:

- Adaptation gains: $\gamma_1 = 7.3, \gamma_2 = 0.3$
- Bias bounds: $\bar{\omega}_A = 0.02$, $\bar{\omega}_B = 0.05$
- Input signal: Same persistently exciting multi-sine as Part (a)
- Simulation time: 200 seconds

All state variables and estimation errors are initialized to zero. σ-modification damping parameters were chosen individually for each estimated parameter based on empirical tuning.

Detailed code can be found in the exercise1_b.m file.

PLOT RESULTS

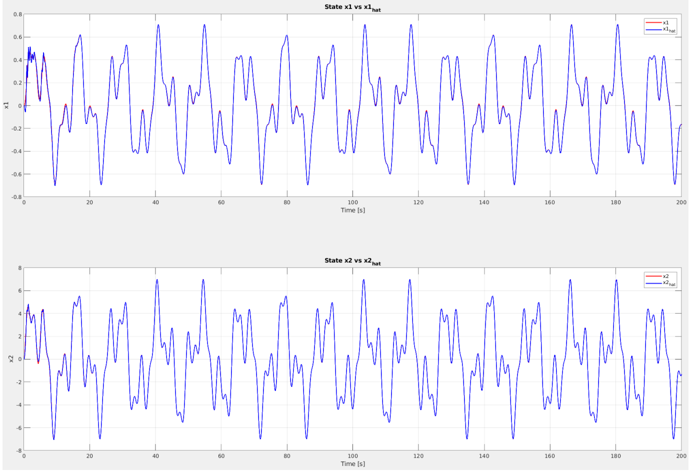


Fig. 5: True vs Estimated States under bias disturbance

Figure 5 shows the evolution of the actual states $x_1(t)$ and $x_2(t)$ versus their corresponding estimates $\hat{x}_1(t)$ and $\hat{x}_2(t)$, when the system is subject to an unknown, bounded bias disturbance $\omega(t)$. Despite the injected bias, the estimated trajectories successfully follow the true dynamics after an initial transient phase. This performance demonstrates the effectiveness of the σ-modified estimator in maintaining accurate state prediction even in the presence of persistent disturbances.

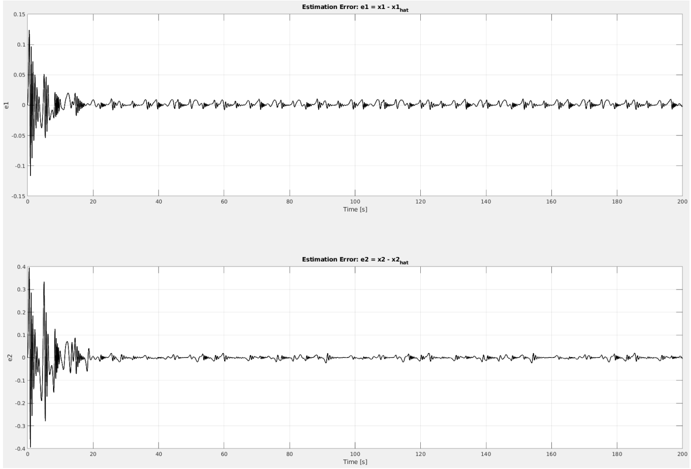


Fig. 6: Estimation Errors $e_1(t)$ and $e_2(t)$ under bounded bias

Figure 6 depicts the time evolution of the estimation errors $e_1(t)$ and $e_2(t)$ for the two system states. Both error signals exhibit an initial transient phase due to mismatched initial conditions and the effect of the unknown disturbance. However, they quickly stabilize and remain bounded close to zero throughout the entire simulation. The error $e_1(t)$ demonstrates particularly fast convergence with very low residual oscillations, while $e_2(t)$ shows slightly higher peak deviations during transients but ultimately converges similarly.

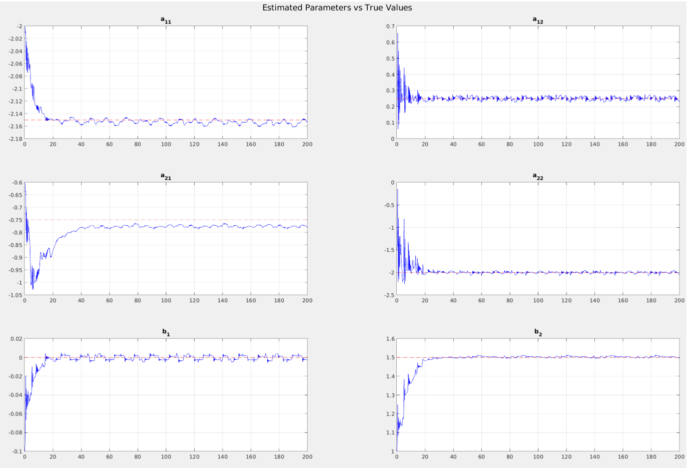


Fig. 7: Parameter estimation $\hat{A}(t), \hat{B}(t)$ vs. true values with bias

Figure 7 illustrates the convergence behavior of the estimated system parameters $\hat{A}(t)$ and $\hat{B}(t)$ in the presence of a bounded bias disturbance $\omega(t)$. Each subplot shows the evolution of a single parameter estimate over time (blue), along with its true value (red dashed line) for reference. It is evident that, despite the influence of the bias, the estimates gradually converge toward their true values. Notably, parameters such as $\hat{a}_{11}$ and $\hat{b}_2$ exhibit rapid convergence with minimal overshoot, while others like $\hat{a}_{21}$ converge more slowly, indicating potential sensitivity to initialization or input excitation. Overall, the estimator successfully tracks all parameters within acceptable margins.

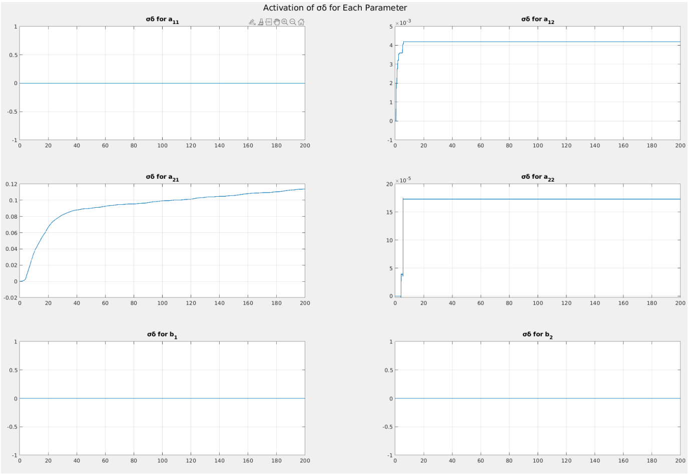


Fig. 8: Activation of δ_σ terms for all parameters

Figure 8 presents the activation behavior of the $\sigma\delta$ modification term for each estimated parameter over time. The plots reveal that for certain parameters, such as $\hat{a}_{11}$, $\hat{b}_1$, and $\hat{b}_2$, the $\sigma\delta$ term remains inactive (i.e., zero) throughout the simulation, indicating that their estimates stayed well within the defined bounds and did not require correction. In contrast, parameters like $\hat{a}_{21}$, $\hat{a}_{12}$, and $\hat{a}_{22}$ exhibit clear activation of the $\sigma\delta$ term, particularly in the initial stages of adaptation. This activation signifies that their estimates approached or exceeded the thresholds defined by the projection bounds. The gradual increase observed for $\hat{a}_{21}$'s $\sigma\delta$ term also suggests a slower convergence, possibly due to less persistent excitation or sensitivity to estimation errors.

Analysis of Bias Impact on Estimation Accuracy

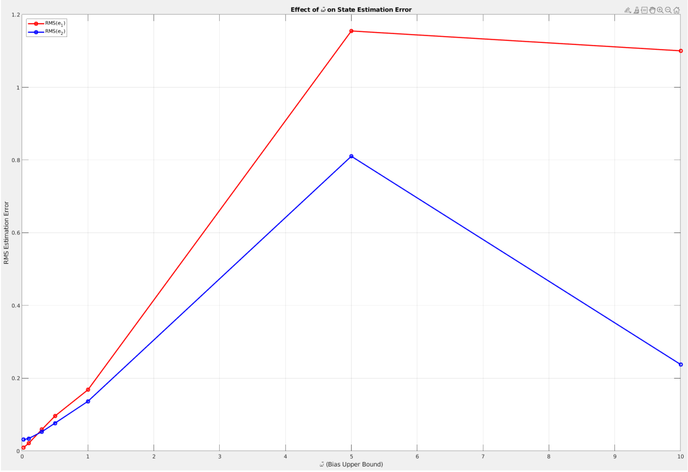


Fig. 9: Bias Impact on Estimation errors e1 and e2

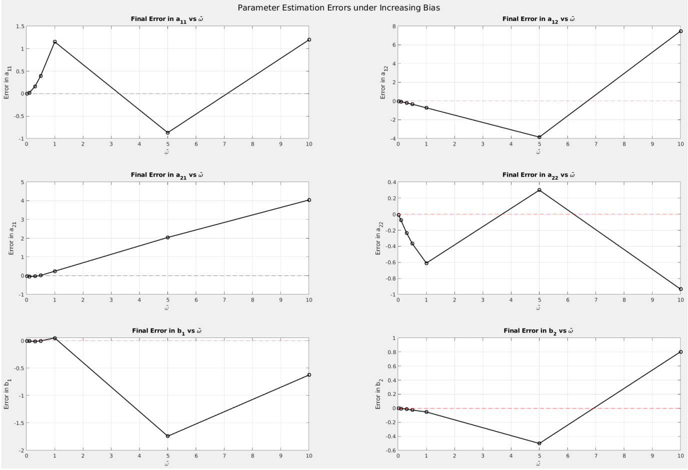


Fig. 10: Bias Impact on Estimation parameters

Next I also investigated the effect of increasing the bias upper bound $\bar{\omega}$ on the accuracy of both state and parameter estimation. The results, as shown in the RMS error plot, indicate that as $\bar{\omega}$ increases from 0.02 to 1.0, both RMS(e_1) and RMS(e_2) increase gradually, signifying a deterioration in estimation performance due to the stronger bias. Notably, when $\bar{\omega} = 5.0$, the RMS error peaks, particularly for e_1, indicating that the estimator struggles to remain robust under large disturbances, and after that it starts to decrease gradually.

The parameter error plots reveal that the estimation accuracy of a_{ij} and b_i elements deteriorates as the bias increases. For moderate bias values ($\bar{\omega} \leq 1.0$), parameter estimates remain close to their true values. However, for $\bar{\omega} = 5.0$ and $\bar{\omega} = 10.0$, large deviations are observed, especially in a_{12}, a_{21}, and b_1, where estimation errors exceed 3–7 units in magnitude. This confirms that the presence of stronger unknown bias severely affects the adaptation dynamics. Despite this, the use of σ-modification and projection methods manages to constrain certain parameters (e.g., a_{11} and b_2) within their feasible bounds, demonstrating partial robustness of the estimation method.

Conclusion

The robust real-time estimator using σ-modification demonstrates reliable performance under bounded disturbances. While higher disturbances degrade accuracy, stability and convergence of estimates are preserved at some degree.

II. Problem 2: Structure Selection and Model Evaluation of Nonlinear system

A. Part (a): Nonlinear System Modeling

We consider the nonlinear system:

$$\dot{x}(t) = f(x(t), u(t), \theta)$$

where the unknown nonlinear function is given by:

$$f(x, u, \theta) = -x^3 + \theta_1 \tanh(x) + \theta_2 \frac{1}{1 + x^2} + u$$

with parameter vector $\theta = [\theta_1, \theta_2]^T \in [0.5, 2]^2$. The objective is to approximate this system using basis function expansions. We experiment with different function classes (e.g., polynomials, Gaussians) and implement real-time parameter estimation.

We also apply cross-validation techniques to evaluate and compare model candidates based on modeling error and complexity.

Methodology

To approximate the unknown system, five different model structures were proposed, each defined by a distinct set of basis functions. These basis functions include both polynomial and nonlinear types such as hyperbolic tangent and Gaussian basis functions. Each model estimates the system output using a linear combination of these basis functions, with parameters updated in real time via a Lyapunov-based adaptation law.

The system states and estimated parameters evolve according to a set of ordinary differential equations, where the parameter update law is derived using the Lyapunov Series-Parallel approach.

The five models considered are summarized as follows:

- **Model 1:** Uses two nonlinear basis functions including $\tanh(x)$ and $\frac{1}{1+x^2}$.
- **Model 2:** Incorporates x^2, $\sin(x)$, and the term $\frac{x^2}{x^2+1}$.
- **Model 3:** Matches the true system structure (included for benchmarking only), with x^3, $\tanh(x)$, and $\frac{1}{1+x^2}$.
- **Model 4:** A pure polynomial model using x, x^2, and x^3.
- **Model 5:** Employs Gaussian radial basis functions centered at different points: e^{-x^2}, $e^{-(x-1)^2}$, and $e^{-(x+1)^2}$.

To evaluate model performance, two types of analysis were conducted:

- **Full-data simulation:** Each model was simulated over the full time horizon, and the energy of the tracking error (i.e., the integral of the squared output error) was computed.
- **Cross-validation:** A 10-fold cross-validation procedure was implemented. For each fold, the model was trained on 90% of the data and tested on the remaining 10%. The average validation error over all folds was used as a metric of generalization performance.

Detailed code can be found in the exercise2_a.m file.

Model Comparison Results

The results of the full data simulations and the cross-validation procedure are summarized below. Each model was simulated over a time horizon of $T = 200$ seconds with a fixed time step of $\Delta t = 0.01$ seconds. The system was excited by the input signal

$$u(t) = c\sin(dt) + f + 0.1\sin(0.2t) + 0.5\sin(0.3t),$$

with parameters $c = 2.5$, $d = 1$, and $f = 0$. This input was designed to ensure sufficient excitation of the system dynamics.

The true system parameters were set as $\theta^* = [0.8, 1.2]$. Estimation was performed using a Lyapunov-Series-Parallel law with estimator gain $\theta_m = 41.2$ and adaptation gains $\gamma = [39.9, 41.7, 43.3]$, which came up after some tuning. Identical initial conditions and gain settings were used for all models to maintain a fair comparison.

Model	**Total Error Energy (E)**	**Average CV Error**
1	0.07223	0.01445
2	0.04901	0.02160
3	0.00090	0.00607
4	**0.03548**	**0.01154**
5	0.09094	0.01878

As we can see, model 3 clearly performs the best across both evaluation metrics. That's not too surprising, since its basis functions (x^3, $\tanh(x)$, and $\frac{1}{1+x^2}$) are identical to those in the actual system. Because of this exact match, it's able to approximate the system accurately and quickly adapt the parameter estimates, which explains the low tracking and validation errors.

That said, if we exclude Model 3 due to its structural similarity to the true system (and the fact that it was essentially designed to match it), then **Model 4** stands out as the next best option. Even though it only uses basic polynomial terms, it still does a great job capturing the system's dynamics and shows the second-best performance overall.

Plot Results

The figures generated for each model include:

- **State trajectories:** Plots of the true state $x_1(t)$ versus the estimated state $\hat{x}_1(t)$ over time, illustrating how well the estimator tracks the system.
- **Tracking error:** The evolution of the error $e_1(t) = x_1(t) - \hat{x}_1(t)$ over time, providing insight into convergence speed and accuracy.
- **Parameter estimates:** Time histories of the estimated parameters $\hat{\theta}_1(t)$, $\hat{\theta}_2(t)$, and $\hat{\theta}_3(t)$, showing how the adaptive law adjusts these values in real-time.

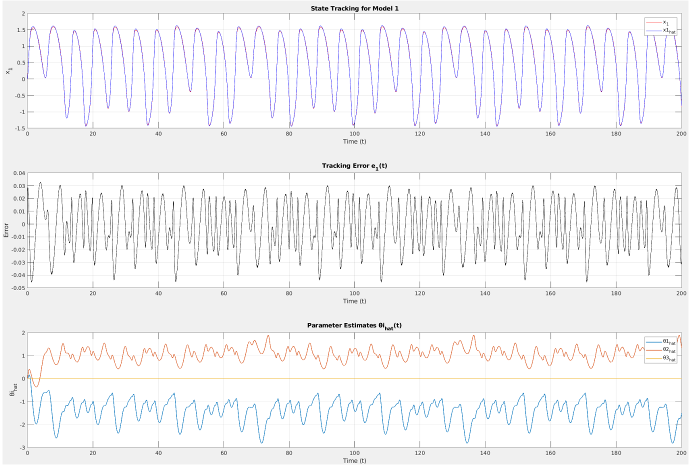


Fig. 11: Results for Model 1

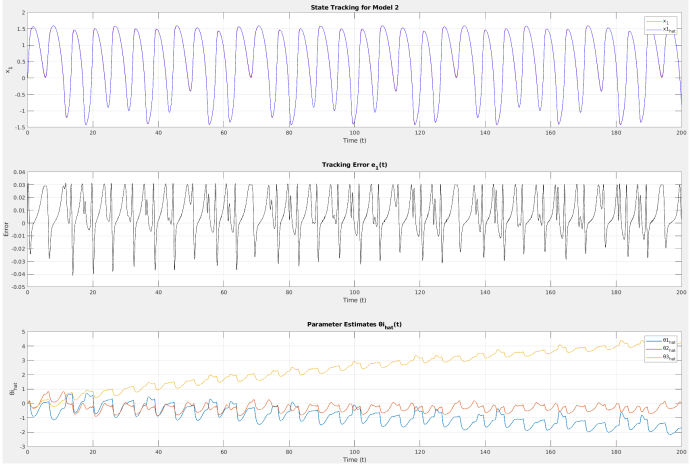


Fig. 12: Results for Model 2

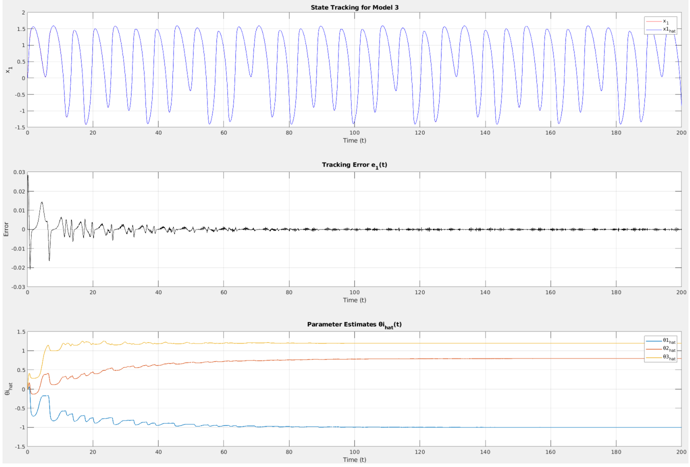


Fig. 13: Results for Model 3

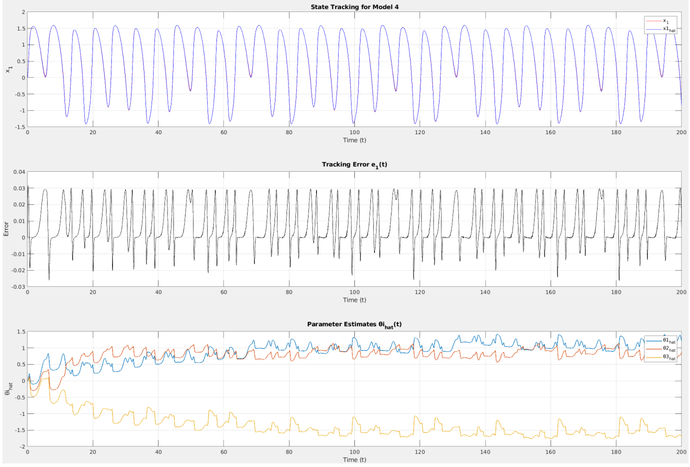


Fig. 14: Results for Model 4

From the simulation results, it is evident that all five models achieve low tracking error (but not zero, except of course for the model 3), with $\hat{x}_1$ closely matching the true state x_1 across the entire time horizon. This indicates that each model is capable of approximating the system dynamics in terms of state estimation.

However, the models differ significantly in the behavior of their parameter estimates. In particular, we observe that some models exhibit notable oscillations and lack of convergence in the estimated parameters $\hat{\theta}_i$. Among all models, **Model 4** demonstrates the most stable convergence, with minimal fluctuations in the parameter estimates. This suggests that **Model 4** provides not only accurate state tracking but also more reliable parameter learning, making it the most robust choice in terms of both estimation accuracy and convergence behavior.

B. Part (b): Final Model Selection and Stability Evaluation

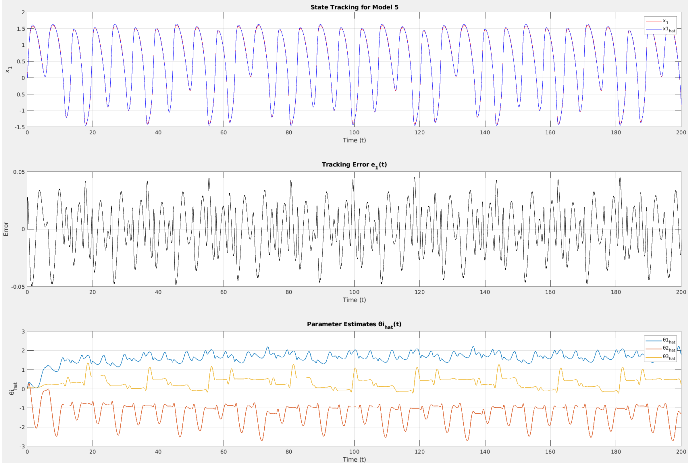


Fig. 15: Results for Model 5

To identify the most suitable model for approximating the nonlinear dynamic system, the five candidate models from above were evaluated. Each model was simulated using the same initial conditions and input signal, and the modeling performance was assessed based on the tracking error:

$$e_1(t) = x_1(t) - \hat{x}_1(t),$$

with the total modeling error energy computed as:

$$E = \int_0^T e_1^2(t)\,dt.$$

To balance modeling accuracy and complexity, the Bayesian Information Criterion (BIC) was calculated for each model:

$$\text{BIC} = k\log(N) + N\log\left(\frac{E}{N}\right),$$

where k is the number of model parameters and N is the number of time steps. The BIC penalizes more complex models while rewarding lower error, making it a suitable metric for model selection.

Detailed code can be found in the exercise2_b.m file.

Model Comparison Results

The table below summarizes the modeling error and BIC score for each model:

Model	Total Error Energy (E)	BIC
1	0.07223	−250620.41
2	0.04901	−258370.04
3	0.00090	−338377.95
4	**0.03548**	**-264831.22**
5	0.09094	−246004.36

Note: Model 3 corresponds to the true system (used for simulation), and thus was excluded from the model selection process to avoid bias. Among the remaining models, Model 4 achieved the lowest BIC score and error, and was therefore selected as the best model.

Stability Evaluation Using Alternative Input Signals

The stability of the final model was tested using input signals that differ from those used during training in part (a). Specifically, I employed a low-frequency sinusoid $u(t) = 2.5\sin(0.05t)$, a high-frequency sinusoid $u(t) = 2.5\sin(6t)$, and a composite input $u(t) = 1.5\sin(0.6t) + 0.8\sin(6.5t)$ that combines both low and high-frequency components. The results show that the model remains stable across all test cases. In the low-frequency case, although the state and its estimate $\hat{x}_1$ closely follow the reference, the parameter estimates $\hat{\theta}_i(t)$ exhibit slow and inconsistent convergence, indicating insufficient excitation. On the contrary, the high-frequency input improves parameter convergence significantly, with all $\hat{\theta}_i(t)$ stabilizing quickly and the estimation error decreasing over time. The most favorable performance is observed with the mixed-frequency input: it provides persistent excitation that ensures both fast and stable convergence of parameter estimates and minimal steady-state tracking error. These results confirm that the final model is stable and generalizes well to previously unseen input conditions, particularly when the inputs are sufficiently rich in frequency. The respective plots are shown below.

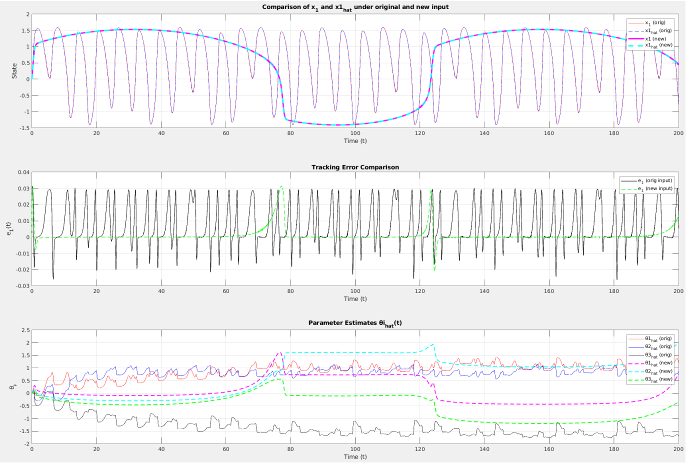


Fig. 16: Results for a low frequency input $(u(t) = 2.5\sin(0.05t))$

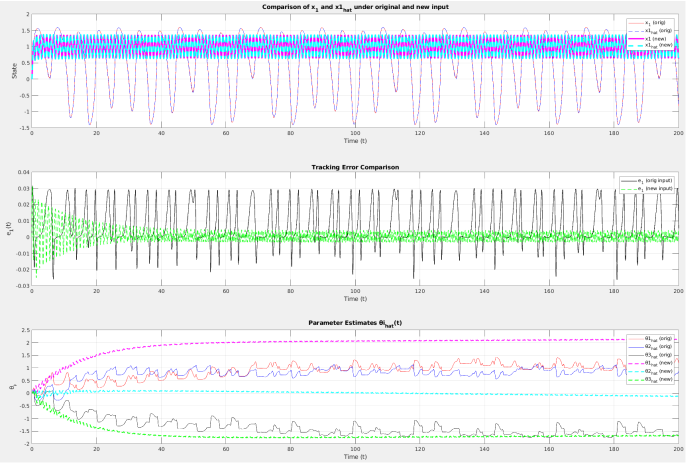


Fig. 17: Results for a high frequency input $(u(t) = 2.5\sin(6t))$

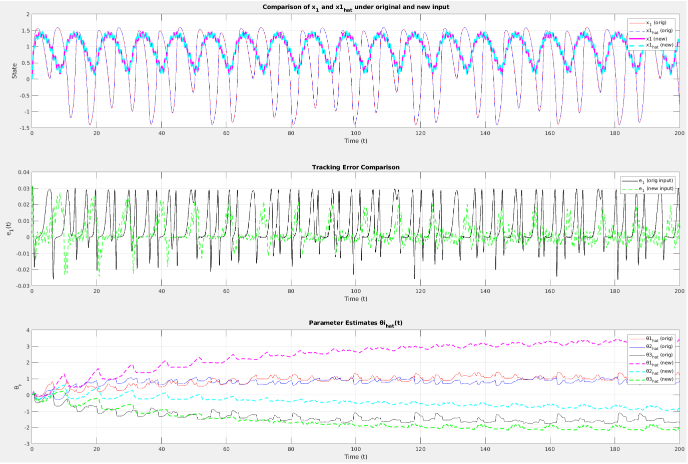


Fig. 18: Results for a mixed frequency input $(u(t) = 1.5\sin(0.6t) + 0.8\sin(6.5t))$

III. Conclusion

Model 4 was selected as the final model due to its optimal balance between modeling accuracy and complexity, as quantified by the BIC. The model exhibited excellent stability and generalization capability, demonstrating low error even when excited by a novel input signal. This confirms its suitability for approximating the dynamics of the original nonlinear system.