Simulation and Modeling of Dynamic Systems

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

***Abstract*—This project focuses on the simulation and modeling of dynamic systems with unknown parameters, using real-time estimation techniques and nonlinear control strategies. The first part explores the implementation of gradient-based and Lyapunov-based estimators for a mass-spring-damper system, under various input conditions and noise levels. The second part deals with the control of a nonlinear roll angle system, where a smooth reference trajectory is tracked using a nonlinear feedback controller.**

***Index Terms*—real-time estimation, nonlinear control, Lyapunov method, gradient method, trajectory tracking, dynamic systems.**

I. Exercise 1

A. First Question: Real-Time Estimation of Unknown Parameters Using Gradient Method

In this section, we design a real-time estimator for the unknown parameters m, b, and k of a mass-spring-damper system using the gradient (steepest descent) method. The system is excited under two different input scenarios:

i) Constant input: $u(t) = 2.5$
ii) Sinusoidal input: $u(t) = 2.5\sin(t)$

For each case, we simulate the system response over a 20-second time interval and compute the real-time estimates $\hat{m}(t)$, $\hat{b}(t)$, and $\hat{k}(t)$. The state $x(t)$ is compared with the estimated state $\hat{x}(t)$, and the estimation error $e_x(t) = x(t) - \hat{x}(t)$ is analyzed. The performance of the estimator is evaluated based on the convergence of the parameter estimates and the accuracy of the state prediction. The problem is defined below:

Consider the classical mass-spring-damper system with an external force acting on the mass. The equation of motion describing the system is:

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = u(t), \tag{1}$$

where $x(t)$ [m] is the displacement of the mass from its equilibrium position, $m > 0$ [kg] is the mass, $b > 0$ [N·s/m] is the damping coefficient, $k > 0$ [N/m] is the spring constant, and $u(t)$ [N] is the external force input.

For the purposes of this problem, we assume the following parameter values:

$$m = 1.315, \quad b = 0.225, \quad k = 0.725.$$

We also assume that the states $x(t)$, $\dot{x}(t)$, and the input $u(t)$ are measurable.

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

The first step we need to take is to bring the system to it's linearly parameterized form. In order to do that i am using a second order filter so that I can minimize the error: $e = x(t) - \hat{x(t)}$ easier. So:

Given the system:

$$\ddot{x}(t) + \frac{b}{m}\dot{x}(t) + \frac{k}{m}x(t) = \frac{1}{m}u(t)$$

$$\Rightarrow \ddot{x} = -\frac{b}{m}\dot{x} - \frac{k}{m}x + \frac{1}{m}u$$

We define:

$$\theta_a^* = \begin{bmatrix} a_1 & a_2 \end{bmatrix}^T = \begin{bmatrix} \frac{b}{m} & \frac{k}{m} \end{bmatrix}^T \quad \text{and} \quad \theta_b^* = \begin{bmatrix} b_0 \end{bmatrix}^T = \frac{1}{m}$$

$$\text{Thus,} \quad \theta^* = \begin{bmatrix} \theta_a^* & \theta_b^* \end{bmatrix}^T = \begin{bmatrix} a_1 & a_2 & b_0 \end{bmatrix}^T = \begin{bmatrix} \frac{b}{m} & \frac{k}{m} & \frac{1}{m} \end{bmatrix}^T$$

Also,

$$\Delta = \begin{bmatrix} -\dot{x} & -x & u \end{bmatrix}^T$$

and so the initial system can be written as

$$\ddot{x} = \theta^{*T}\Delta$$

with

$$\theta^* = \begin{bmatrix} \theta_a^* & \theta_b^* \end{bmatrix}^T = \begin{bmatrix} a_1 & a_2 & b_0 \end{bmatrix}^T = \begin{bmatrix} \frac{b}{m} & \frac{k}{m} & \frac{1}{m} \end{bmatrix}^T$$

and

$$\Delta = \begin{bmatrix} -\dot{x} & -x & u \end{bmatrix}^T \tag{2}$$

Now all quantities of vector Δ are measurable. However, $\ddot{x}$ is not measurable, so we will apply linear parameterization again, by using a second order filter:

$$\Lambda(s) = s^2 + s(p_1 + p_2) + p_1 p_2 = (s + p_1)(s + p_2)$$

where p_1 and p_2 are poles placed in the left half-plane to ensure stability,

$$\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 \end{bmatrix}^T = \begin{bmatrix} p_1 + p_2 & p_1 p_2 \end{bmatrix}^T$$

Thus, by filtering both sides of the equation $\ddot{x} = \theta^{*T}\Delta$ with $\frac{1}{\Lambda(s)}$, after calculations, we end up with an expression of the form:

$$x = \Theta^T J, \text{where } J = \begin{bmatrix} -\frac{\Delta n{-}1^T(s)}{\Lambda(s)}x & \frac{\Delta m^T(s)}{\Lambda(s)}u \end{bmatrix}^T,$$

$$\Theta = \begin{bmatrix} \theta_a^{*T} - \lambda^T & \theta_b^{*T} \end{bmatrix},$$

$$\Delta_{n-1}(s) = \begin{bmatrix} s & 1 \end{bmatrix}^T \text{ and}$$

$$\Delta_m(s) = 1$$

so finally we have the linearly parameterized system:

$$x = \Theta^T J, \text{where} \tag{3}$$

$$J = \begin{bmatrix} -\frac{\begin{bmatrix} s & 1 \end{bmatrix}^T}{s^2+(p_1+p_2)s+p_1p_2} x & +\frac{1}{s^2+(p_1+p_2)s+p_1p_2} u \end{bmatrix}^T \text{ and}$$

$$\Theta = \begin{bmatrix} \frac{b}{m} - \lambda_1 & \frac{k}{m} - \lambda_2 & \frac{1}{m} \end{bmatrix}^T$$

GRADIENT METHOD

Next, we define the modeling error as $e = x - \hat{x} = (\Theta^* - \hat{\Theta})J = -\tilde{\theta}J$

$$\Leftrightarrow e = -\tilde{\theta}J$$

Where $\tilde{\theta} = \hat{\Theta} - \Theta^*$

We define the function we want to minimize, that is,

$$K(\hat{\Theta}) = \frac{e^2}{2} = \frac{(x - \hat{\Theta}^T J)^2}{2}$$

According to the gradient method:

$$\dot{\hat{\Theta}} = -\gamma \nabla K$$

The gradient of the function $K(\hat{\Theta})$ is:

$$\nabla K = -eJ$$

Therefore: $\dot{\hat{\Theta}} = \gamma e J$ with $\gamma > 0$ a constant, and

$$\hat{\Theta}(0) = \begin{bmatrix} 0.01 \\ 0.01 \\ 0.01 \end{bmatrix}, \text{(our choice)}$$

Thus, we have the signals:

$$J_1 = -\frac{s}{s^2 + (p_1 + p_2)s + p_1p_2} x(s)$$

$$J_2 = -\frac{1}{s^2 + (p_1 + p_2)s + p_1p_2} x(s)$$

$$J_3 = \frac{1}{s^2 + (p_1 + p_2)s + p_1p_2} u(s)$$

If we pass the three above equations in the time domain (with zero initial conditions) we finally get the following system:

$$(\Sigma) : \begin{cases} \hat{x}(t) = \hat{\Theta}^T J(t) \\ \ddot{x}(t) = -\frac{b}{m}\dot{x} - \frac{k}{m}x + \frac{1}{m}u \\ \ddot{J}_1(t) = -\lambda_1 \dot{J}_1(t) - \lambda_2 J_1 - \dot{x}(t) \\ \ddot{J}_2(t) = -\lambda_1 \dot{J}_2(t) - \lambda_2 J_2 - x(t) \\ \ddot{J}_3(t) = -\lambda_1 \dot{J}_3(t) - \lambda_2 J_3 - u(t) \\ \dot{\hat{\theta}}_1 = \gamma e J_1 \\ \dot{\hat{\theta}}_2 = \gamma e J_2 \\ \dot{\hat{\theta}}_3 = \gamma e J_3 \end{cases}$$

Thus, solving the above system, we obtain an estimate over time of the unknown parameters $\hat{m}$, $\hat{b}$ and $\hat{k}$, and we can then calculate them using the equation (15):

$$\Theta = \begin{bmatrix} \frac{b}{m} - \lambda_1 & \frac{k}{m} - \lambda_2 & \frac{1}{m} \end{bmatrix}^T$$

Although the real system is unknown, given that we have input-output measurement access, we can design the same adaptive system (3) and estimate the parameter vector $\hat{\Theta}(t)$ which may lead the system to stability. It is known from theory that the Lyapunov function $V = \frac{1}{2}\tilde{\theta}^2 \geq 0$ and its derivative (by Barbalat's lemma) will drive the system to a state where either the input u or the regressor vector J vanishes. We have practical convergence if the excitation signal does not vary slowly over time and the regressor vector leads the system to a persistently exciting state, i.e., the output x will be sufficiently rich to excite the adaptive system.

Next I will try to predict the results of the two different input scenarios:

PERSISTENT EXCITATION CONDITION (PEC)

Definition 4.1.1 (PEC): A piecewise continuous vector signal $\phi(t) : \mathbb{R}_+ \to \mathbb{R}^n$ satisfies PEC with excitation level $\alpha_0 > 0$ if there exist constants $\alpha_1, T_0 > 0$ such that:

$$\alpha_1 I \geq \frac{1}{T_0} \int_t^{t+T_0} \phi(\tau)\phi^T(\tau)\, d\tau \geq \alpha_0 I, \quad \forall t \geq 0. \tag{4}$$

The regressor vector $\phi(t)$ for parameter estimation is:

$$\phi(t) = [J_1(t), J_2(t), J_3(t)]^T, \tag{5}$$

where $J_i(t)$ are solutions to the filter equations:

$$\ddot{J}_1 + \lambda_1 \dot{J}_1 + \lambda_2 J_1 = -\dot{x}, \tag{6}$$

$$\ddot{J}_2 + \lambda_1 \dot{J}_2 + \lambda_2 J_2 = -x, \tag{7}$$

$$\ddot{J}_3 + \lambda_1 \dot{J}_3 + \lambda_2 J_3 = u \tag{8}$$

II. CASE 1: CONSTANT INPUT ($u = 2.5$)

At steady state ($\dot{x} = 0, \ddot{x} = 0$):

$$x = \frac{2.5}{k}, \quad J_1(t) \to 0, \quad J_2(t) \to -\frac{2.5}{k\lambda_2}, \quad J_3(t) \to \frac{2.5}{\lambda_2}. \tag{9}$$

The regressor matrix becomes:

$$\phi(t)\phi^T(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \left(\frac{2.5}{k\lambda_2}\right)^2 & -\frac{6.25}{k\lambda_2^2} \\ 0 & -\frac{6.25}{k\lambda_2^2} & \left(\frac{2.5}{\lambda_2}\right)^2 \end{bmatrix}. \tag{10}$$

Properties:

- **Rank** = 1 (linearly dependent columns)
- **Minimum eigenvalue** = 0

That means:

- PEC is **not satisfied**
- Only k can be estimated from steady-state response
- m and b are **unidentifiable**

III. Case 2: Sinusoidal Input ($u = 2.5\sin(t)$)

The input excites all system modes:

$$\phi(t) = \begin{bmatrix} A_1 \sin(t+\varphi_1) \\ A_2 \sin(t+\varphi_2) \\ A_3 \sin(t+\varphi_3) \end{bmatrix}, \quad \text{with distinct phases } \varphi_i. \tag{11}$$

For $T_0 = 2\pi$:

$$\frac{1}{2\pi}\int_t^{t+2\pi} \phi(\tau)\phi^T(\tau)d\tau =$$

$$\begin{bmatrix} \frac{A_1^2}{2} & \frac{A_1A_2}{2}\cos\Delta_{12} & \frac{A_1A_3}{2}\cos\Delta_{13} \\ \frac{A_1A_2}{2}\cos\Delta_{12} & \frac{A_2^2}{2} & \frac{A_2A_3}{2}\cos\Delta_{23} \\ \frac{A_1A_3}{2}\cos\Delta_{13} & \frac{A_2A_3}{2}\cos\Delta_{23} & \frac{A_3^2}{2} \end{bmatrix}, \tag{12}$$

where $\Delta_{ij} = \varphi_i - \varphi_j$
.

Properties:

- **Rank** = 3 (if $\varphi_i \neq \varphi_j$)
- **Minimum eigenvalue** > 0

That means:

- PEC is **satisfied**
- All parameters (m, b, k) are identifiable
- Convergence is guaranteed theoretically

Summary of Results

Input Signal	PEC Satisfied?	Identifiable Parameters
$u = 2.5$	No	Only k
$u = 2.5\sin(t)$	Yes	m, b, k

TABLE I: Comparison of input signal effects on parameter identifiability

Conclusion

The PEC explains why:

- Constant inputs fail to estimate mass (m) and damping (b)
- Sinusoidal inputs enable full parameter estimation

Detailed calculations as well as code can be found in the exercise1_a.m and GraDE_2nd_order.m files.

Plot Results

To make things easier, I decided to keep the same values for the λ_1 and λ_2 at 1.0 (in my code there is a snippet that calculates the $p1$ and $p2$ so that their real parts are positive, in order to ensure stability.)

With that being said, first i decided to use

$$\gamma = 1$$

. The results were:

```
=== Final Parameter Estimates for Set 1 (u = 2.5) ===
Mass:      m_true = 1.3150       m_est = 0.8189
Damping:   b_true = 0.2250       b_est = 0.2155
Stiffness: k_true = 0.7250       k_est = 0.7535

=== Final Parameter Estimates for Set 2 (u = 2.5sin(t)) ===
Mass:      m_true = 1.3150       m_est = 0.8404
Damping:   b_true = 0.2250       b_est = 0.1175
Stiffness: k_true = 0.7250       k_est = 0.1724
```

Fig. 1: Results with $\gamma = 1$

Increasing the γ did not help with the accuracy of the estimated parameters and only caused oscillation, so i tried to gradually decrease it. The final optimum value for γ that i found based on the λ_1 and λ_2 is

$$\gamma = 0.05$$

Finaly, the results for this run were:

For u = 2.5 (Input Set 1)

```
=== Final Parameter Estimates for Set 1 (u = 2.5) ===
Mass:      m_true = 1.3150       m_est = 1.2449
Damping:   b_true = 0.2250       b_est = 0.5696
Stiffness: k_true = 0.7250       k_est = 0.7199

=== Final Parameter Estimates for Set 2 (u = 2.5sin(t)) ===
Mass:      m_true = 1.3150       m_est = 1.3025
Damping:   b_true = 0.2250       b_est = 0.2233
Stiffness: k_true = 0.7250       k_est = 0.7131
```

Fig. 2: Results with $\gamma = 0.05$

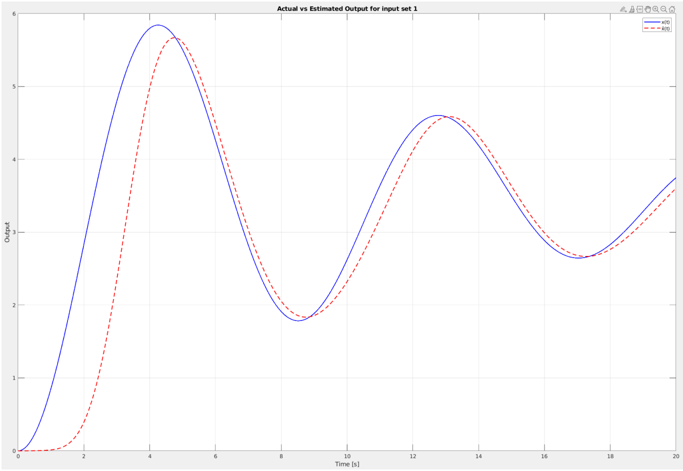

Fig. 3: $x(t)$ and $\hat{x}(t)$ for $\gamma = 0.05$

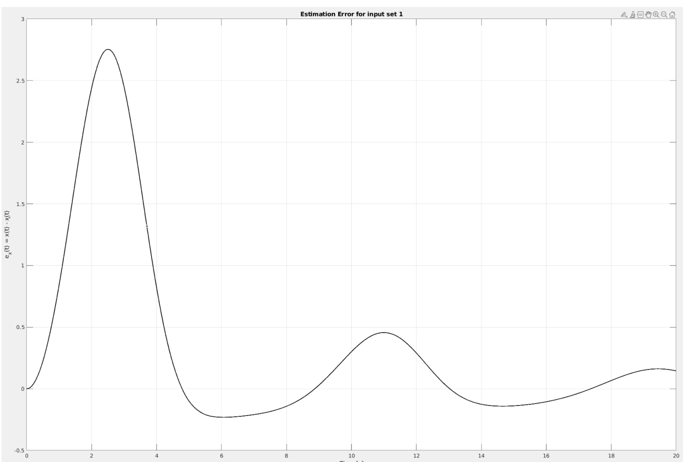

Fig. 4: Error e_x for $\gamma = 0.05$

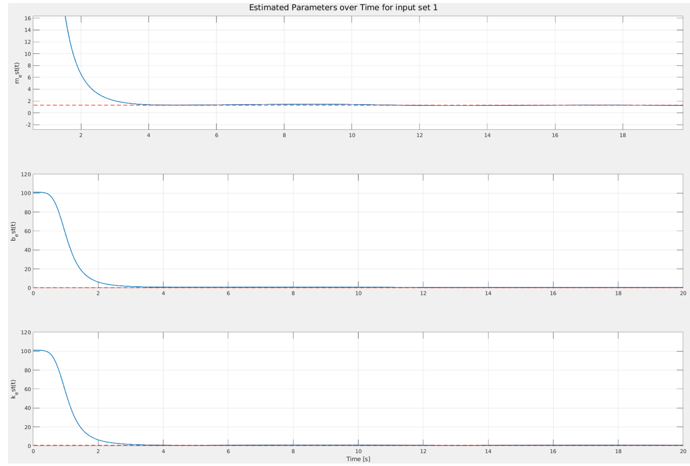

Fig. 5: Estimated parameters over time for $\gamma = 0.05$

For u = 2.5sin(t) (Input Set 2)

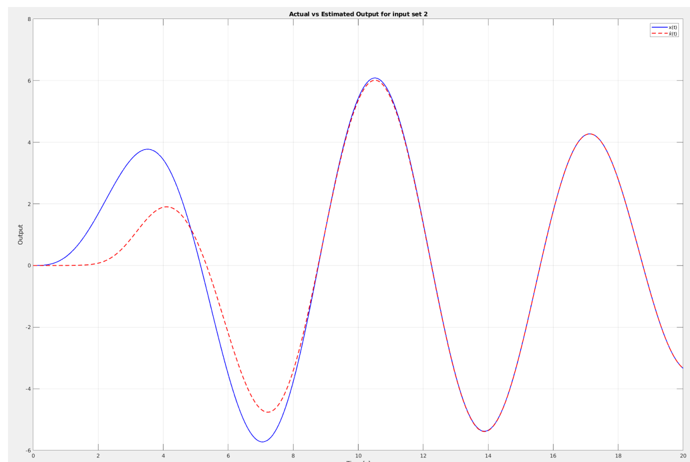

Fig. 6: $x(t)$ and $\hat{x}(t)$ for $\gamma = 0.05$

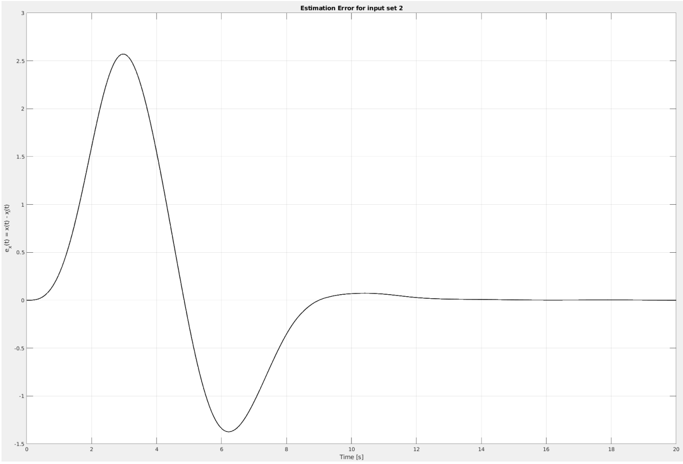

Fig. 7: Error e_x for $\gamma = 0.05$

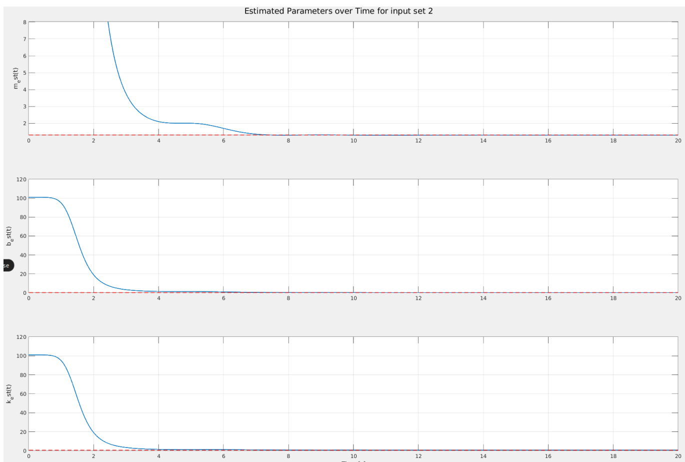

Fig. 8: Estimated parameters over time for $\gamma = 0.05$

As we can see, the accuracy of the estimated parameters was way better when we used as input the sine wave, since in that case the PEC was satisfied, as we already mentioned above.

A. Second Question: Real-Time Estimation of Unknown Parameters Using Lyapunov Method (Parallel and Series-Parallel)

I) 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:

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

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

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

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

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

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:

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

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:

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

In our case we have the system:

$$\begin{aligned} x_1 &= x(t) \\ x_2 &= \dot{x}(t) \end{aligned}$$

$$\Longrightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{b}{m}x_2 - \frac{k}{m}x_1 + \frac{1}{m}u \end{cases}$$

$$\Rightarrow \dot{\mathbf{x}} = \underbrace{\begin{pmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{pmatrix}}_{A} \mathbf{x} + \underbrace{\begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix}}_{B} u$$

So, we only want to estimate the terms a_{21}, a_{22} and b_2 of A and B matrices.

Finally we get the following system:

$$(\Sigma) : \begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{b}{m}x_2 - \frac{k}{m}x_1 + \frac{1}{m}u \\ \dot{\hat{x}}_1(t) = \hat{x}_2(t) \\ \dot{\hat{x}}_2(t) = \hat{a}_{21}\hat{x}_1(t) + \hat{a}_{22}\hat{x}_2(t) + \hat{b}_2 u(t) \\ \dot{\hat{a}}_{21}(t) = \gamma_1 e_2 \hat{x}_1(t) \\ \dot{\hat{a}}_{22}(t) = \gamma_1 e_2 \hat{x}_2(t) \\ \dot{\hat{b}}_2(t) = \gamma_2 e_2 u(t), \quad e_2 = x_2(t) - \hat{x}_2(t) \end{cases}$$

And by solving it we get our estimated parameters:

$$\hat{m} = 1/\hat{b}_2$$

$$\hat{b} = -\hat{a}_{22} \cdot \hat{m}$$

$$\hat{k} = -\hat{a}_{21} \cdot \hat{m}$$

By once again 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.

II) Series-Parallel Configuration

Given the system:

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

The estimated model system based on the Series-Parallel Config- uration is:

$$\dot{\hat{x}} = \hat{A}x + \hat{B}u + \Theta_m(x - \hat{x})$$

Next we define the errors:

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

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

$$= (A - \hat{A})x + (B - \hat{B})u - \Theta_m e_x = -e_A x - e_B u - \Theta_m e_x,$$

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

Lyapunov Function

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

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

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

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

But:

$$e_x^T e_A x = x^T e_A^T e_x = tr(e_A^T e_x x^T)$$
$$e_x^T e_B u = u^T e_B^T e_x = tr(e_B^T e_x u^T)$$

Thus:

$$\dot{V} = -e_x^T \Theta_m e_x + tr\left(-e_A^T e_x x^T - e_B^T e_x u^T + \frac{e_A^T \dot{\hat{A}}}{\gamma 1} + \frac{e_B^T \dot{\hat{B}}}{\gamma 2}\right)$$

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

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

With:

$$\Theta_m = \begin{pmatrix} \theta_{m1} & \theta_{m2} \\ \theta_{m3} & \theta_{m4} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad \hat{x} = \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix}$$

In our case we have the system:

$$x_1 = x(t)$$
$$x_2 = \dot{x}(t)$$

$$\Longrightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{b}{m}x_2 - \frac{k}{m}x_1 + \frac{1}{m}u \end{cases}$$

$$\Rightarrow \dot{\mathbf{x}} = \underbrace{\begin{pmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{pmatrix}}_{A} \mathbf{x} + \underbrace{\begin{pmatrix} 0 \\ \frac{1}{m} \end{pmatrix}}_{B} u$$

So, we only want to estimate the terms a_{21}, a_{22} and b_2 of *A* and *B* matrices.

Finally we get the following system:

$$(\Sigma): \begin{cases} \dot{x}_1(t) = x_2(t) \\ \dot{x}_2(t) = -\frac{b}{m}x_2 - \frac{k}{m}x_1 + \frac{1}{m}u \\ \dot{\hat{x}}_1(t) = x_2(t) + \theta_{m1}e_1 + \theta_{m2}e_2 \\ \dot{\hat{x}}_2(t) = \hat{a}_{21}x_1(t) + \hat{a}_{22}x_2(t) + \hat{b}_2 u(t) + \theta_{m3}e_1 + \theta_{m4}e_2 \\ \dot{\hat{a}}_{21}(t) = \gamma_1 e_2 x_1(t) \\ \dot{\hat{a}}_{22}(t) = \gamma_1 e_2 x_2(t) \\ \dot{\hat{b}}_2(t) = \gamma_2 e_2 u(t), \quad e_1 = x_1(t) - \hat{x}_1(t) \quad e_2 = x_2(t) - \hat{x}_2(t) \end{cases}$$

And by solving it we get our estimated parameters:

$$\hat{m} = 1/\hat{b}_2$$

$$\hat{b} = -\hat{a}_{22} \cdot \hat{m}$$

$$\hat{k} = -\hat{a}_{21} \cdot \hat{m}$$

By once again 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.

Detailed calculations as well as code can be found in the exercise1_b.m, LyapPar.m and LyapSP.m files.

Plot Results

Parallel Configuration

```
=== Final Parameter Estimates (Parallel Configuration) ===
Mass:      m_true = 1.3150       m_est = 1.3332
Damping:   b_true = 0.2250       b_est = 0.2061
Stiffness: k_true = 0.7250       k_est = 0.7287

=== Final Parameter Estimates (Series-Parallel Configuration) ===
Mass:      m_true = 1.3150       m_est = 1.3258
Damping:   b_true = 0.2250       b_est = 0.2261
Stiffness: k_true = 0.7250       k_est = 0.7163
```

Fig. 9: Results with $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$

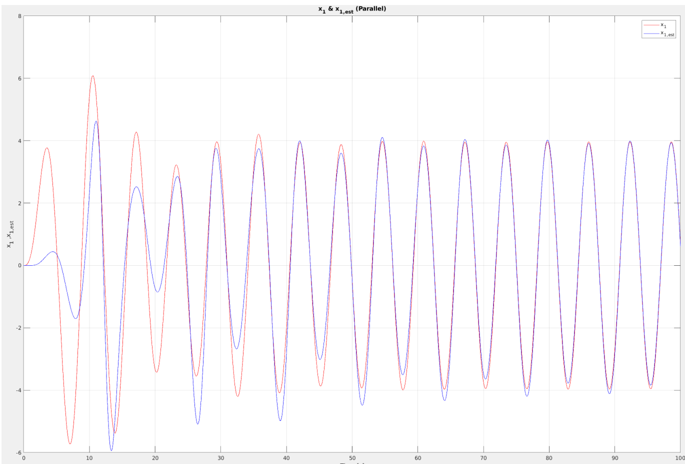

Fig. 10: $x(t)$ and $\hat{x}(t)$ for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$

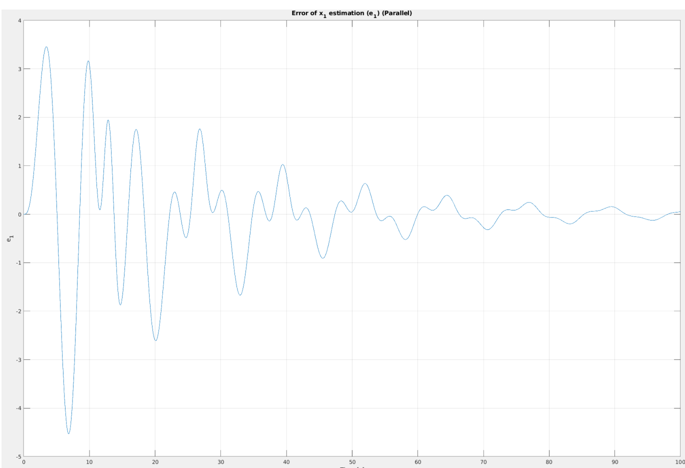

Fig. 11: Error e_x for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$

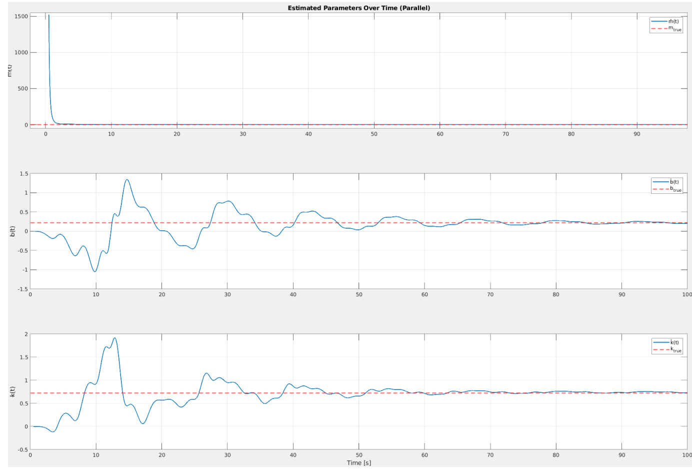

Fig. 12: Estimated parameters over time for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$

Series-Parallel Configuration

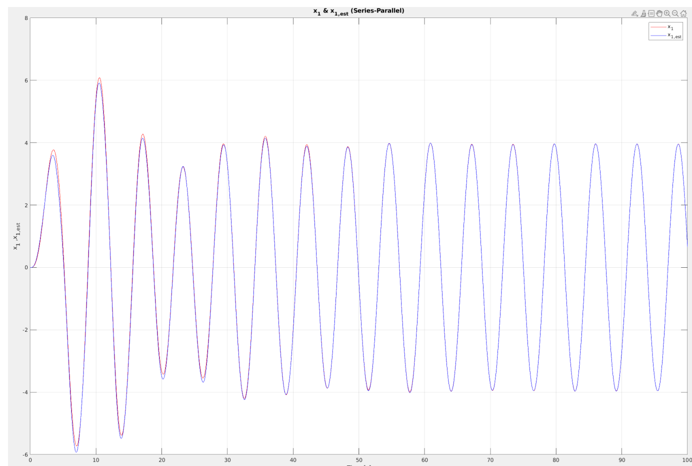

Fig. 13: $x(t)$ and $\hat{x}(t)$ for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$

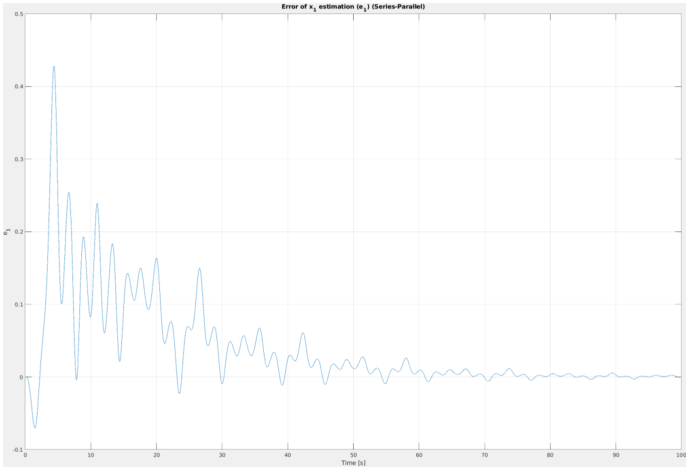

Fig. 14: Error e_x for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$

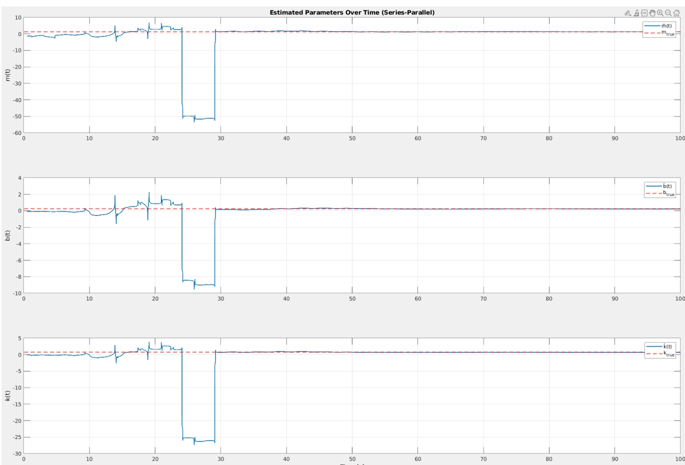

Fig. 15: Estimated parameters over time for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$

In the process of finding the optimal γ and θ_m values, I tested several combinations. The best results were obtained with $\gamma = [0.06, 0.019]$ for the parallel configuration, and $\gamma = [0.06, 1.5]$ with $\theta_m = [0.2, 0.2, 0.2, 0.2]$ for the series-parallel configuration.

As observed from the plots, when using the parallel configuration, the final estimated values $\hat{m}$, $\hat{b}$, and $\hat{k}$ were slightly less accurate compared to the series-parallel configuration. Moreover, in the series-parallel configuration, the error converges to a near steady-state more quickly and exhibits fewer oscillations. Lastly, for the estimated parameters in the series-parallel configuration, I applied a moving average filter to reduce large spikes and better illustrate the trend in the plots.

IV. Third Question: Real-Time Estimation of Unknown Parameters Using Lyapunov Method WITH NOISE(Parallel and Series-Parallel)

In the following section, we extend the procedure described in part (b) by introducing measurement noise to the system output $x(t)$. Specifically, we consider additive noise of the form $\eta(t) = \eta_0 \sin(2\pi f_0 t)$ for all $t \geq 0$, with $\eta_0 = 0.25$ and $f_0 = 20$. The goal is to compare the results obtained with and without the presence of noise, analyzing how the estimation accuracy of the parameters is affected. Furthermore, we investigate the impact of varying the noise amplitude η_0 on the accuracy of the estimated parameters. Finally, plots will be generated to illustrate the estimation error of the parameters as a function of the noise amplitude.

The noise that is added affects only the measurement of the output x, which is the state variable x_1 so we add it to our equations only after the actual solution for x_1 — that is, in the definition of the error. Thus, what we can easily observe is that the noise appears only after the actual value is computed and not during the estimation of x_1. Therefore, we can easily notice that:

$$\dot{\hat{\alpha}}_{21} = \gamma_1 e\hat{x_1} = \gamma_1(x_1 - \hat{x_1})\hat{x_1} = \gamma_1(\hat{x_1}^2 - x_1\hat{x_1}) \quad \text{(Parallel Config)}$$

$$\dot{\hat{\alpha}}_{21} = \gamma_1 e x_1 = \gamma_1(x_1 - \hat{x_1})x_1 = \gamma_1(x_1^2 - x_1\hat{x_1}) \quad \text{(Series-Parallel)}$$

Consequently, the noise that is added appears more significantly (both in the first and the second power term, affecting accuracy) in the series-parallel structure compared to the parallel one, where it appears only in the first power term. Therefore, we expect that in our simulations, the results under the presence of noise will be better for the parallel structure rather than for the series-parallel structure.

Detailed calculations as well as code can be found in the exercise1_c.m, LyapParNoise.m and LyapSPNoise.m files.

Plot Results

Parallel Configuration

```
=== Final Parameter Estimates (Parallel Configuration with NOISE) ===
Mass:      m_true = 1.3150       m_est = 1.3332
Damping:   b_true = 0.2250       b_est = 0.2061
Stiffness: k_true = 0.7250       k_est = 0.7287
Elapsed time is 1.331610 seconds.

=== Final Parameter Estimates (Series-Parallel Configuration with NOISE) ===
Mass:      m_true = 1.3150       m_est = 1.2707
Damping:   b_true = 0.2250       b_est = 0.2191
Stiffness: k_true = 0.7250       k_est = 0.6852
Elapsed time is 0.678750 seconds.
```

Fig. 16: Results with $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$ (NOISE)

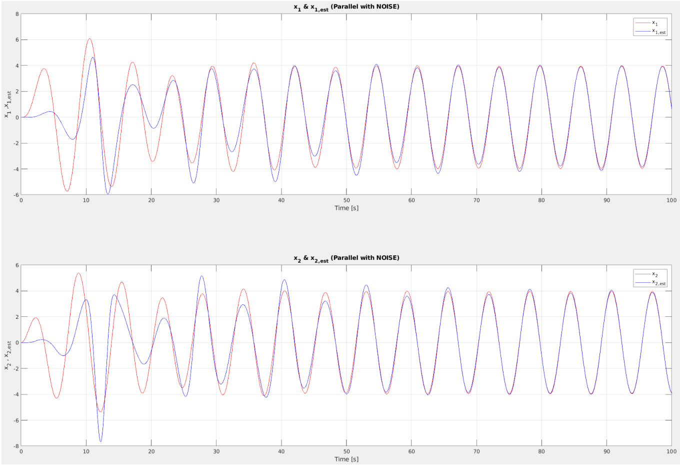

Fig. 17: $x(t)$ and $\hat{x}(t)$ for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$ (NOISE)

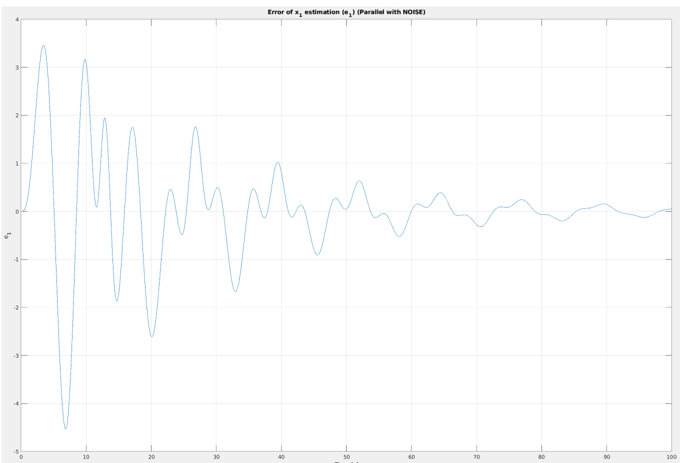

Fig. 18: Error e_x for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$ (NOISE)

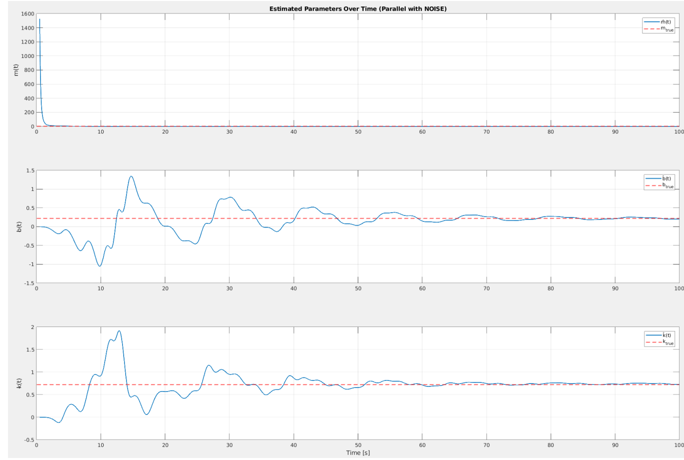

Fig. 19: Estimated parameters over time for $\gamma_1 = 0.06$ and $\gamma_2 = 0.019$ (NOISE)

SERIES-PARALLEL CONFIGURATION

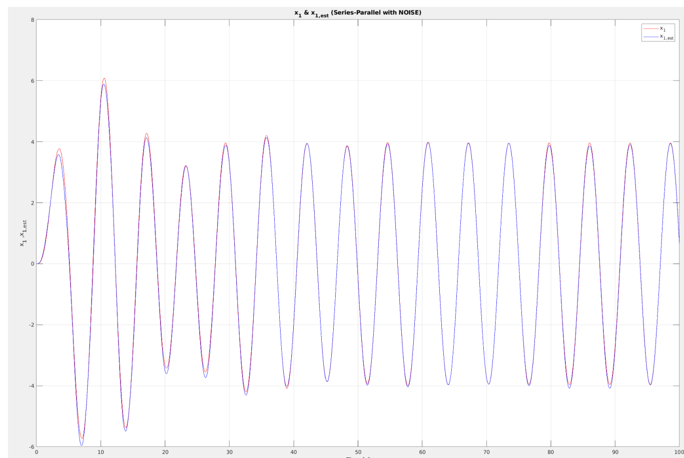

Fig. 20: $x(t)$ and $\hat{x}(t)$ for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$ (NOISE)

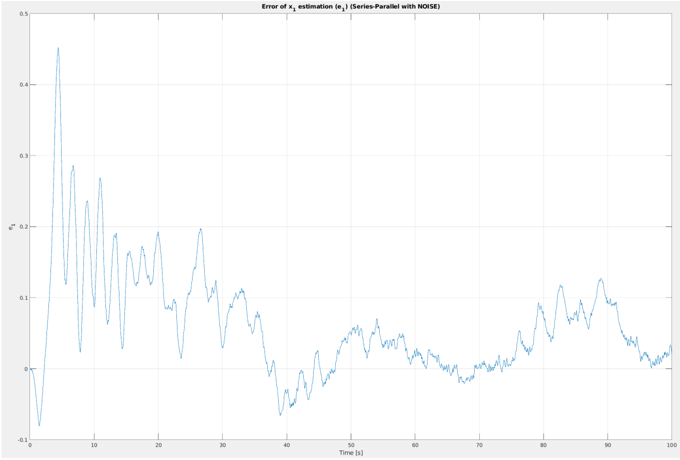

Fig. 21: Error e_x for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$ (NOISE)

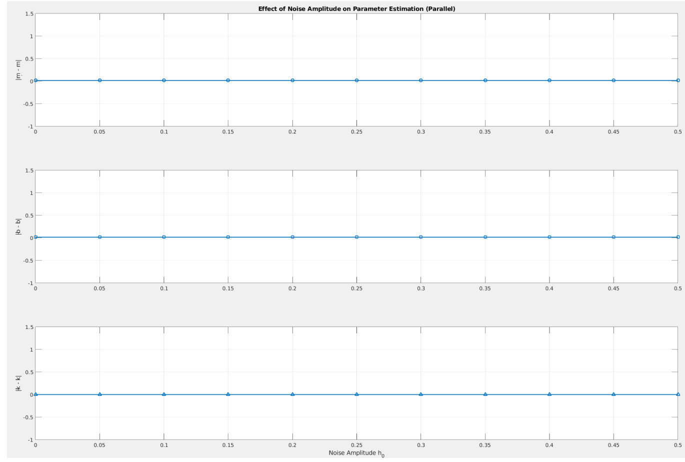

Fig. 23: Effects of noise amplitude on parameter estimation (Parallel Configuration)

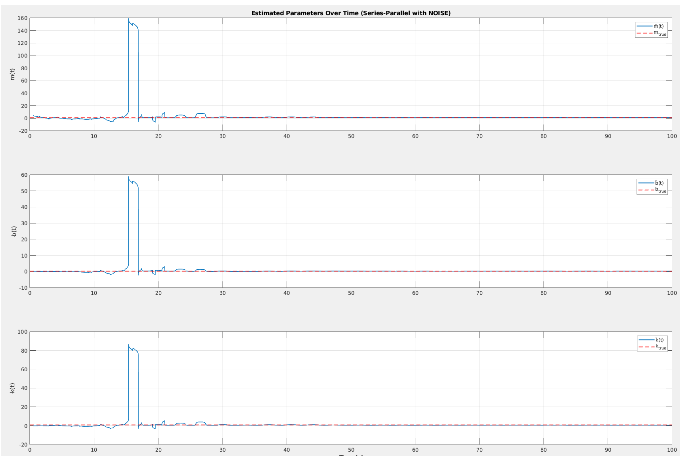

Fig. 22: Estimated parameters over time for $\gamma_1 = 0.06$ and $\gamma_2 = 1.5$ and $\theta_m = [0.2, 0.2, 0.2, 0.2]$ (NOISE)

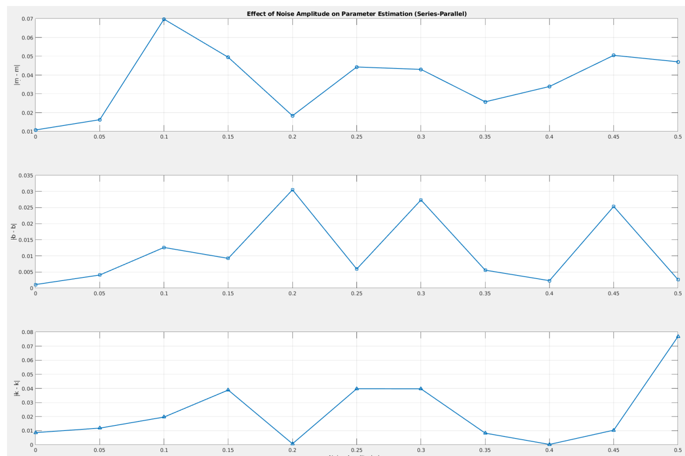

Fig. 24: Effects of noise amplitude on parameter estimation (Series-Parallel Configuration)

As observed, the results confirm the theoretical analysis. More specifically, it is clear that when noise is introduced, the estimated parameters in the parallel configuration remain essentially unchanged, whereas in the series-parallel configuration, significant deviations from the true values are observed.

To further investigate the effect of varying the noise amplitude η_0 on the accuracy of the parameter estimates, the following plots were generated:

In the series-parallel configuration, it was observed that the parameter estimation errors (for parameters $\hat{m}$, $\hat{b}$, $\hat{k}$) varied with the noise amplitude η_0. Specifically, as η_0 increased, the errors exhibited fluctuations, indicating that the series-parallel approach is sensitive to measurement noise. The magnitude of the error was not consistently increasing with noise, but larger noise levels tended to cause higher deviations in some cases. In contrast, the parallel configuration demonstrated remarkable robustness to noise. The estimation errors for all three parameters remained effectively zero across the entire range of noise amplitudes tested. This shows that, for this system, the parallel configuration is significantly more resilient to measurement noise compared to the series-parallel configuration, leading to more stable and accurate parameter estimation under noisy conditions.

V. EXERCISE 2

A. First Question: Design a suitable for the system controller

Consider the nonlinear system describing the roll angle of an aircraft with an input torque, given by the equation:

$$\ddot{r}(t) = -a_1\dot{r}(t) - a_2\sin(r(t)) + a_3\dot{r}^2(t)\sin(2r(t)) + bu(t) + d(t), \tag{13}$$

where $r(t)$ [rad] is the roll angle, $a_i > 0$, $i = 1, 2, 3$, and $b > 0$ are constant, unknown parameters, $u(t)$ is the control input, and $d(t)$ represents external disturbances. The control objective is to regulate the roll angle $r(t)$ from the initial condition $r(0) = 0$ to the desired value $\bar{r}_d = \frac{\pi}{10}$, and then return to zero.

To achieve this, a smooth reference trajectory $r_d(t)$ is proposed that moves from 0 to $\bar{r}_d$ and back to 0, over a time span of 20 [sec].

The chosen reference trajectory is defined as:

$$r_d(t) = \frac{\pi}{20}\left(1 - \cos\left(\frac{2\pi t}{T}\right)\right),$$

where $T = 20$ [sec] is the total duration of the maneuver.

This trajectory is well-suited for the control objective because:

- It starts at zero: $r_d(0) = 0$,
- Reaches the desired maximum value: $r_d(T/2) = \frac{\pi}{10}$,
- And returns smoothly back to zero: $r_d(T) = 0$.

The function is smooth (i.e., continuously differentiable), which ensures that both the reference position $r_d(t)$ and its derivatives (velocity and acceleration) are well-defined and continuous.

The non-linear controller is structured as follows:

- A time-varying normalization factor $\phi(t)$ is computed using the expression:
$$\phi(t) = (\phi_0 - \phi_\infty)e^{-\lambda t} + \phi_\infty,$$
which allows for aggressive control early in the transient phase and more conservative control later.
- The normalized position error is defined as:
$$z_1 = \frac{r(t) - r_d(t)}{\phi(t)}.$$
- A virtual control law α is introduced based on this normalized error:
$$\alpha = -k_1 \cdot T(z_1),$$
where $T(z) = \ln\left(\frac{1+z}{1-z}\right)$ is used for saturation-like smooth nonlinear feedback.
- The normalized velocity error is:
$$z_2 = \frac{\dot{r}(t) - \alpha}{\rho}.$$
- Finally, the control input $u(t)$ is given by:
$$u(t) = -k_2 \cdot T(z_2).$$

Detailed calculations as well as code can be found in the exercise2_a.m and u_nonlinear.m files.

PLOT RESULTS

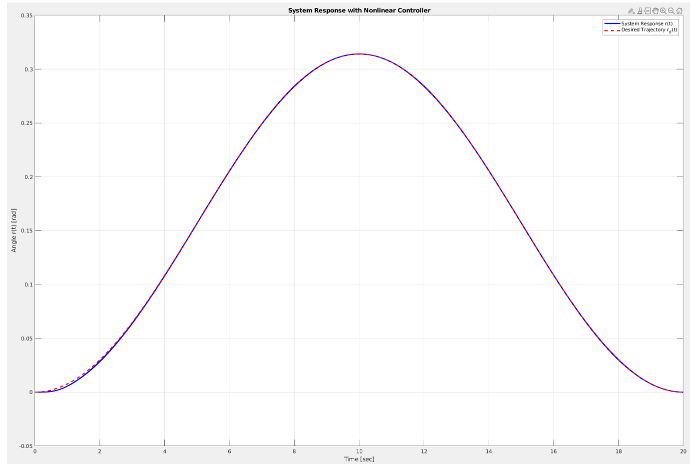

Fig. 25: System Response with Nonlinear Controller

The plot shows how the system behaves when using a nonlinear controller. The blue line is the actual system response $r(t)$, and the red dashed line is the desired trajectory $r_d(t)$. As we can see, the system follows the desired path very closely throughout the entire simulation time (0 to 20 seconds). This means the controller is doing a good job, keeping the system on track despite the nonlinear dynamics. The key idea behind the controller is the use of a time-varying gain $\phi(t)$ and a nonlinear transformation $T(z)$, which help smooth out the response and handle errors in a smart way. Overall, the results show that the controller works effectively, given the controller parameters that I gave as input:

- Initial value of $\phi(t)$: $\phi_0 = 1.0$
- Final value of $\phi(t)$: $\phi_\infty = 0.01$
- Exponential decay rate: $\lambda = 1$
- Normalization factor: $\rho = 1.5$
- Gain for position error: $k_1 = 2$
- Gain for velocity error: $k_2 = 4$

Observation: By making the ϕ_∞ value bigger we can see that the tracking accuracy of the desired trajectory worsens, as shown in the figure below.

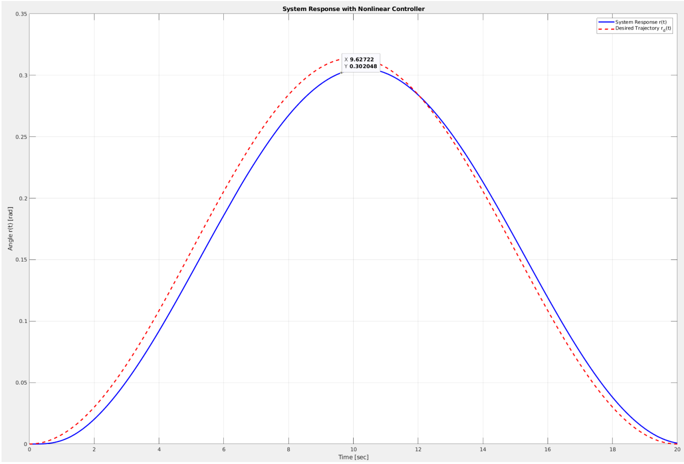

Fig. 26: System Response with Nonlinear Controller, bigger ϕ_∞

B. *Second Question: Lyapunov Method / Series-Parallel Configuration Estimator WITHOUT NOISE*

Now we will define the state variables of the system:

$$\begin{cases} x_1 = r(t) \\ x_2 = \dot{r}(t) \end{cases} \Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \ddot{r}(t) = -a_1 x_2 - a_2 \sin x_1 + a_3 x_2^2 \sin(2x_1) + bu \end{cases}$$

Next we will consider:

$$\sin x_1 = f(x)$$

and

$$x_2^2 \sin(2x_1) = g(x)$$

where $\quad x = [x_1, x_2]^T$

Thus:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -a_1 x_2 - a_2 f(x) + a_3 g(x) + bu$$

SERIES-PARALLEL CONFIGURATION LYAPUNOV

$$\dot{\hat{x}}_1 = x_2 + \theta_{m1}(x_1 - \hat{x}_1)$$
$$\dot{\hat{x}}_2 = -\hat{a}_1 x_2 - \hat{a}_2 f(x) + \hat{a}_3 g(x) + bu + \theta_{m2}(x_2 - \hat{x}_2), \quad \theta_{mi} > 0$$

Next we define the errors as such:

$$e_{x1} = x_1 - \hat{x}_1, \quad e_{x2} = x_2 - \hat{x}_2$$
$$e_{a1} = \hat{a}_1 - a_1, \quad e_{a2} = \hat{a}_2 - a_2, \quad e_{a3} = \hat{a}_3 - a_3, \quad e_b = \hat{b} - b$$

Then:

$$\dot{e}_{x1} = \dot{x}_1 - \dot{\hat{x}}_1, \quad \dot{e}_{x2} = \dot{x}_2 - \dot{\hat{x}}_2$$

$$\dot{e}_{a1} = \dot{\hat{a}}_1, \quad \dot{e}_{a2} = \dot{\hat{a}}_2, \quad \dot{e}_{a3} = \dot{\hat{a}}_3, \quad \dot{e}_b = \dot{\hat{b}}$$

Therefore:

$$\begin{aligned} \dot{e}_{x2} &= \dot{x}_2 - \dot{\hat{x}}_2 \\ &= -a_1 x_2 - a_2 f(x) + a_3 g(x) + bu \\ &\quad - \left(-\hat{a}_1 x_2 - \hat{a}_2 f(x) + \hat{a}_3 g(x) + \hat{b}u + \theta_{m2}(x_2 - \hat{x}_2)\right) \\ &= f(x)(a_2 - \hat{a}_2) + x_2(-a_1 + \hat{a}_1) - g(x)(-a_3 + \hat{a}_3) - \theta_{m2}(x_2 - \hat{x}_2 \\ &= x_2 e_{a1} + f(x) e_{a2} - g(x) e_{a3} - \theta_{m2} e_{x2} - u e_b \end{aligned}$$

Next we consider the Lyapunov function:

$$V_2(t) = \frac{1}{2} e_{x_2}^2 + \frac{1}{\gamma_1} e_{a1}^2 + \frac{1}{\gamma_2} e_{a2}^2 + \frac{1}{\gamma_3} e_{a3}^2 + \frac{1}{\gamma_4} e_b^2 \tag{14}$$

$$\begin{aligned} \Rightarrow \dot{V}_2(t) &= e_{x_2}\dot{e}_{x_2} + \frac{1}{\gamma_1} e_{a1}\dot{e}_{a1} + \frac{1}{\gamma_2} e_{a2}\dot{e}_{a2} + \frac{1}{\gamma_3} e_{a3}\dot{e}_{a3} + \frac{1}{\gamma_4} e_b\dot{e}_b \\ &= e_{x_2}\left(x_2 e_{a1} + f(x) e_{a2} - g(x) e_{a3} - u e_b - \theta_{m2} e_{x_2}\right) + \frac{1}{\gamma_1} e_{a1}\dot{\hat{a}}_1 \\ &\quad + \frac{1}{\gamma_2} e_{a2}\dot{\hat{a}}_2 + \frac{1}{\gamma_3} e_{a3}\dot{\hat{a}}_3 + \frac{1}{\gamma_4} e_b \dot{\hat{b}} \\ &= e_{x_2} x_2 e_{a1} + e_{x_2} f(x) e_{a2} - e_{x_2} g(x) e_{a3} - e_{x_2} u e_b \\ &\quad - e_{x_2}^2 \theta_{m2} + \frac{1}{\gamma_1} e_{a1}\dot{\hat{a}}_1 + \frac{1}{\gamma_2} e_{a2}\dot{\hat{a}}_2 + \frac{1}{\gamma_3} e_{a3}\dot{\hat{a}}_3 + \frac{1}{\gamma_4} e_b \dot{\hat{b}} \\ &= -\theta_{m2} \cdot e_{x_2}^2 + e_{a1}\left(e_{x_2} x_2 + \frac{\dot{\hat{a}}_1}{\gamma_1}\right) + e_{a2}\left(e_{x_2} f(x) + \frac{\dot{\hat{a}}_2}{\gamma_2}\right) \\ &\quad + e_{a3}\left(-e_{x_2} g(x) + \frac{\dot{\hat{a}}_3}{\gamma_3}\right) + e_b\left(-e_{x_2} u + \frac{\dot{\hat{b}}}{\gamma_4}\right) \end{aligned}$$

In order for the function to be negative semi-definite, we choose:

$$\begin{aligned} \dot{\hat{a}}_1 &= -\gamma_1 e_{x_2} x_2 \\ \dot{\hat{a}}_2 &= -\gamma_2 e_{x_2} f(x) \\ \dot{\hat{a}}_3 &= \gamma_3 e_{x_2} g(x) \\ \dot{\hat{b}} &= \gamma_4 e_{x_2} u, \quad \gamma_i > 0 \end{aligned}$$

and finally we are left with the following system:

$$(\Sigma) : \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -a_1 x_2 - a_2 f(x) + a_3 g(x) + bu \\ \dot{\hat{x}}_1 = x_2 + \theta_{m1}(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 = -\hat{a}_1 x_2 - \hat{a}_2 f(x) + \hat{a}_3 g(x) + \hat{b}u + \theta_{m2}(x_2 - \hat{x}_2) \\ \dot{\hat{a}}_1 = -\gamma_1 e_{x_2} x_2 \\ \dot{\hat{a}}_2 = -\gamma_2 e_{x_2} f(x) \\ \dot{\hat{a}}_3 = \gamma_3 e_{x_2} g(x) \\ \dot{\hat{b}} = \gamma_4 e_{x_2} u, \quad \gamma_i > 0, \quad \theta_{mi} > 0 \end{cases}$$

Detailed calculations as well as code can be found in the exercise2_b.m and roll_angle_dynamics.m files.

Plot Results

```
=== Final Parameter Estimates (Series-Parallel Configuration) ===
a1_true = 1.3150          a1_est = 1.3138
a2_true = 0.7250          a2_est = 0.7246
a3_true = 0.2250          a3_est = 0.5121
b_true = 1.1750           b_est = 1.1762
```

Fig. 27: Results with $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$

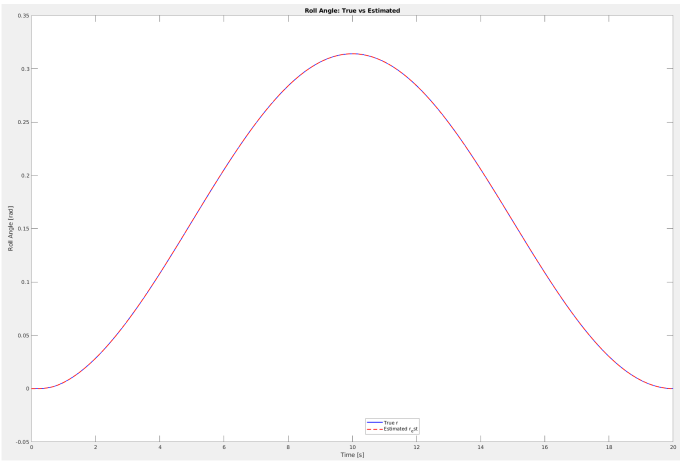

Fig. 28: $r(t)$ and $\hat{r}(t)$ for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$

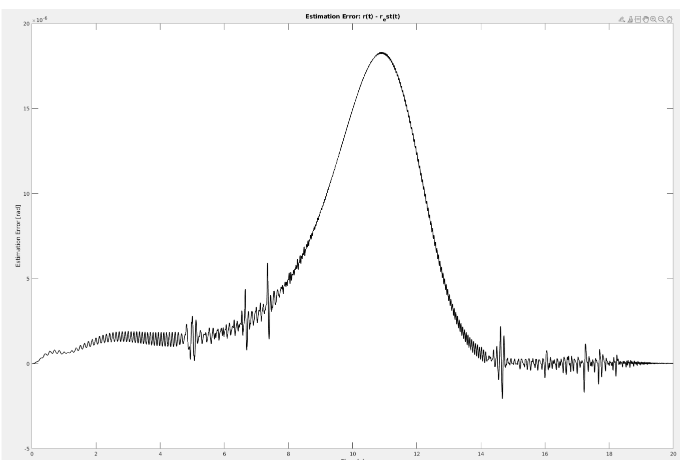

Fig. 29: Error $r - \hat{r}$ for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$

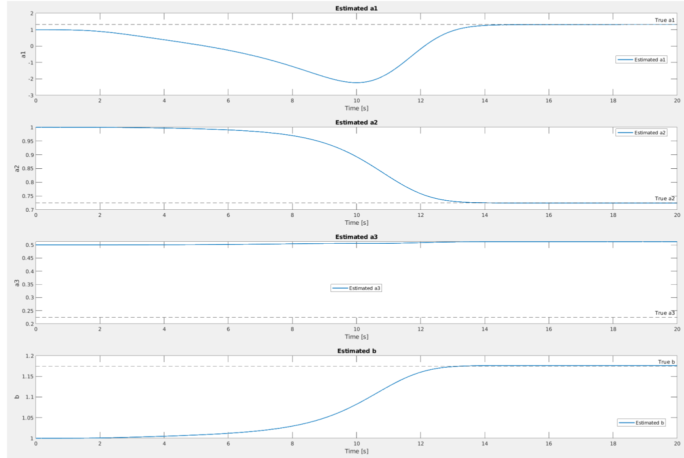

Fig. 30: Estimated parameters over time for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$

In the absence of external disturbances, the system's parameter estimation performance seems to yield the desired results. As seen from the figures above, the estimated parameters closely converge to their true values (except maybe for a_3), indicating that the estimator performs well under ideal conditions. This outcome is expected, as the lack of noise allows the Series-Parallel Configuration to operate without interference in the measured output. The tracking of the reference signal is also accurate, and the estimation error remains minimal throughout the experiment.

C. Third Question: Lyapunov Method / Series-Parallel Configuration Estimator WITH NOISE

In the next step, I introduced a non-negative disturbance into the system. Specifically, the disturbance term $d(t)$ was no longer set to zero. Instead, it was defined as

$$d(t) = 0.15 \sin(0.5t)$$

This means that a form of "noise" was added to the system's acceleration $\ddot{r}$, which corresponds to the state variable x_2. In the following section, we examine how this added noise affects the accuracy of the estimated parameters.

Detailed calculations as well as code can be found in the exercise2_c.m and roll_angle_dynamics_noise.m files.

Plot Results

```
=== Final Parameter Estimates (Series-Parallel Configuration) (NOISE) ===
a1_true = 1.3150          a1_est = 1.4180
a2_true = 0.7250          a2_est = 0.7163
a3_true = 0.2250          a3_est = 0.5147
b_true = 1.1750           b_est = 1.2517
```

Fig. 31: Results with $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$ (NOISE)

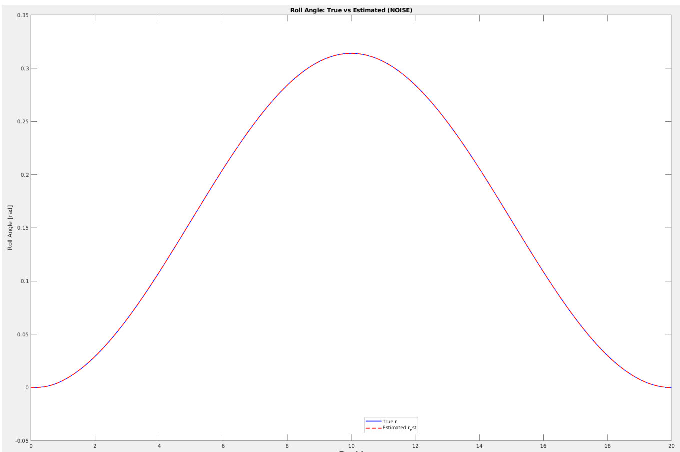

Fig. 32: $r(t)$ and $\hat{r}(t)$ for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$ (NOISE)

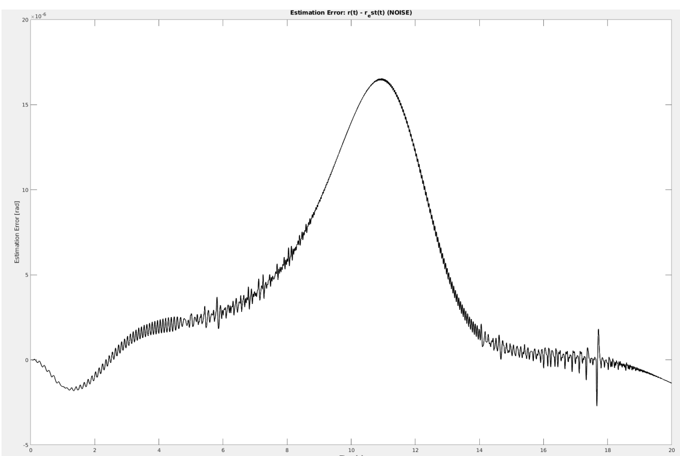

Fig. 33: Error $r - \hat{r}$ for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$ (NOISE)

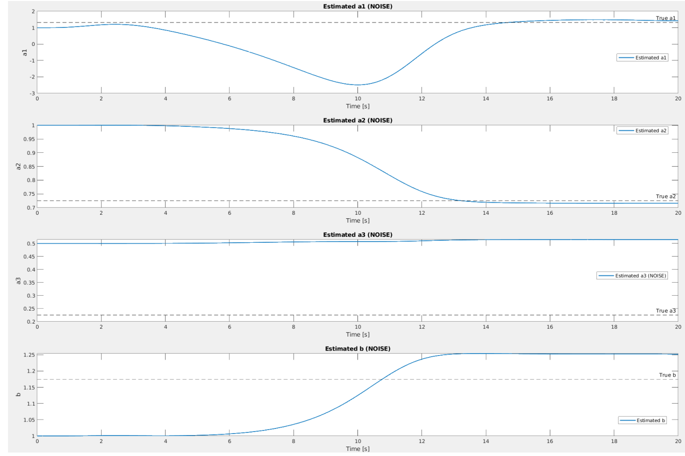

Fig. 34: Estimated parameters over time for $\gamma = [8.5, 0.027, 1.15, 0.03]$ and $\theta_m = [50, 90]$ (NOISE)

By analyzing the figures above, it is evident that the parameter estimation results do not converge to their true values, indicating that the presence of the disturbance affects the estimation performance. This behavior can be anticipated based on a similar reasoning to the analysis performed in Exercise 1c. Since the Series-Parallel Configuration relies on the actual output of the system, it is inherently more sensitive to external disturbances or measurement noise. The plots appear to validate this expectation, as they reflect the impact of the added noise on both the accuracy and stability of the estimated parameters.