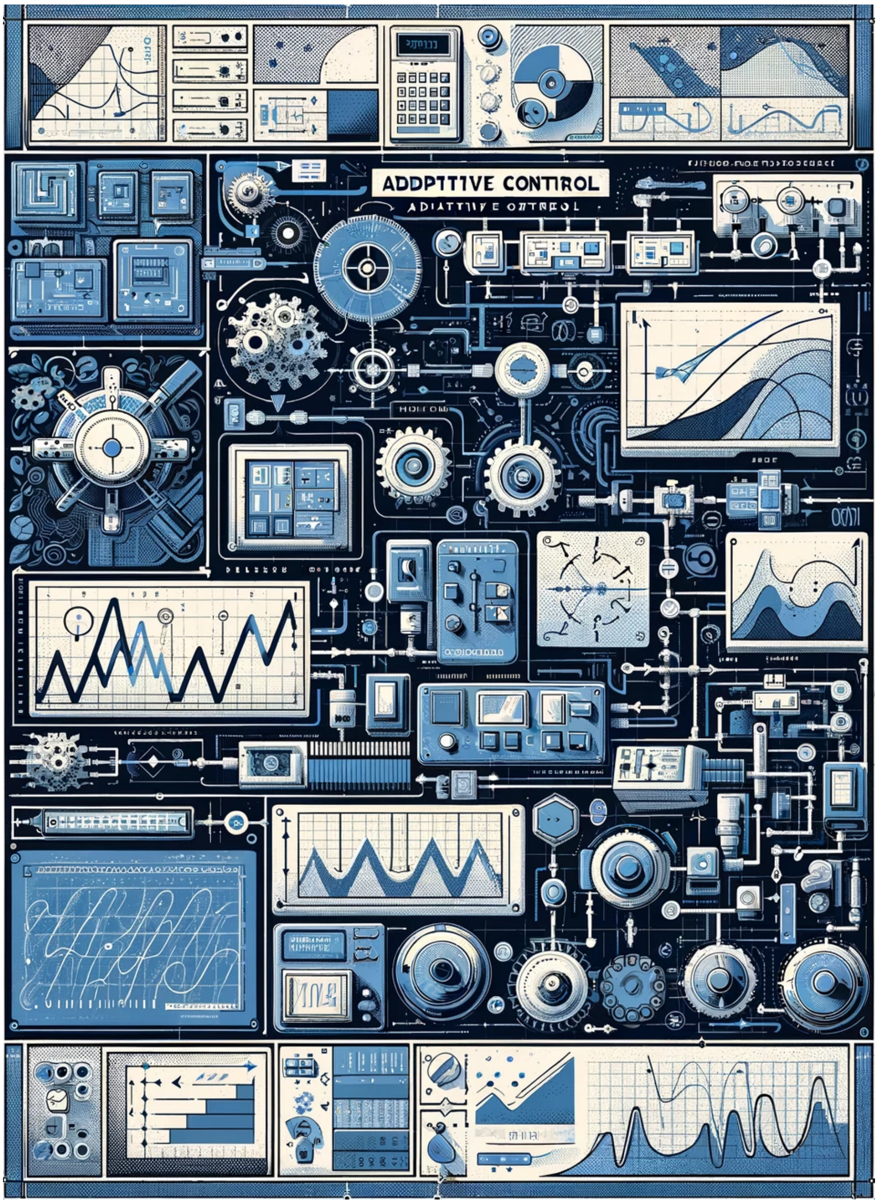
Simulation 5 Adaptive Control

Student Name: Amirhossein Ehsani
Student ID: 810602159
Instructor: Dr. Ayati



July 7, 2024

Contents

List of Figures

List of Tables

1 System

The student number : 810602159.

$$\rightarrow G(s) = \frac{1.24 \times 10^{-5}}{s^2 + 0.01663s + 1.494e \times 10^{-5}}$$

2 MIT Rule Derivation

First, it is necessary to consider the desired model. Here, we take the desired model in the following form:

$$G(s) = \frac{b_m}{s^2 + a_{m1}s + a_{m2}}$$

Considering the error in the form:

$$e = y - y_m$$

and defining the cost function as:

$$J = 0.5 \times e^2$$

we can obtain the natural law for the parameters as:

$$\frac{d\theta}{dt} = -\gamma \frac{\partial J}{\partial \theta} = -\gamma e \frac{\partial e}{\partial \theta}$$

To derive the closed-loop equation, we first rewrite the system in the form:

$$\ddot{y} = -a_1\dot{y} - a_2 y + bu$$

$$\dot{y}_m = -a_{m1}\dot{y}_m - a_{m2}y_m + b_m u_c$$

Therefore, the control form should be taken as:

$$u = \theta_1 u_c - \theta_2 y - \theta_3 \dot{y}$$

Substituting into the main equation, the closed-loop system relationship is obtained in the form:

$$\ddot{y} = -(b\theta_3 + a_1)\dot{y} - (b\theta_2 + a_2)y + b(\theta_1 u_c)$$

Now, we can convert this relationship into the following form:

$$y = \frac{(b\theta_1)}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} u_c$$

Now, by differentiating e with respect to the parameters only, we can calculate the value of $\frac{\partial e}{\partial \theta_i}$. The value of y_m only includes constant parameters and is removed.

The value of the derivative of the error with respect to the parameters is calculated as follows:

$$\frac{\partial e}{\partial \theta_1} = \frac{b}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} u_c$$

$$\frac{\partial e}{\partial \theta_2} = \frac{-b\theta_1}{(s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2))^2} \frac{b}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} u_c$$

$$= \frac{-b\theta_1 s}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} y$$

$$\frac{\partial e}{\partial \theta_3} = \frac{-b\theta_1 s}{(s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2))^2} \frac{b}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} u_c$$

$$= \frac{-b\theta_1 s}{s^2 + (b\theta_3 + a_1)s + (b\theta_2 + a_2)} \dot{y}$$

We take the desired system such as $\zeta = 0.6$ and $\omega_n = 0.04$. Therefore the desired system will be:

$$G_d es(s) = \frac{0.0016}{s^2 + 0.048s + 0.0016}$$

Since the derivative values in the adaptation mechanism cannot be calculated, we use their estimates. Thus, the correction laws can be rewritten in the form:

$$\frac{d\theta_1}{dt} = -\gamma \frac{a_{m2}}{s^2 + a_{m1}s + a_{m2}} e u_c$$

$$\frac{d\theta_2}{dt} = \gamma \frac{a_{m2}}{s^2 + a_{m1}s + a_{m2}} e y$$

$$\frac{d\theta_3}{dt} = \gamma \frac{a_{m2}}{s^2 + a_{m1}s + a_{m2}} e \dot{y}$$

In this relation, the value of γ is obtained as follows:

$$\gamma = \frac{b\gamma'}{a_{m2}}$$

We will take $b\gamma' = \lambda$

$$\gamma = \frac{\lambda}{a_{m2}}$$

3 MIT Gradient Descent Implementation

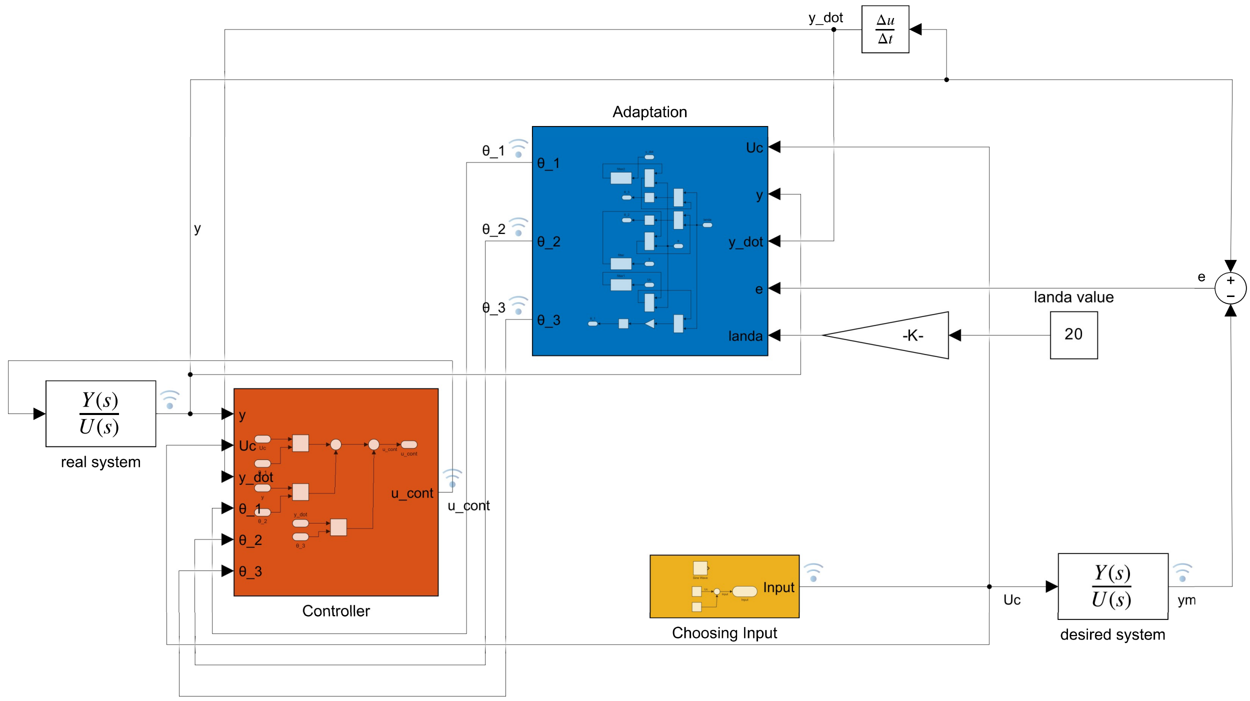


Figure 1: Simulink model of the MIT gradient descent.

1. **Input Signal** (u_c):

   - Represents the control input or the desired reference signal for the system.

2. **Transfer Function Block** ($G(s)$):

   - Models the dynamics of the system being controlled. The transfer function is typically represented as:

$$G(s) = \frac{b_m}{s^2 + a_{m1}s + a_{m2}}$$

3. **Controller** (u):

   - Generates the control signal based on the error (e) and possibly its derivatives. The control law is given by:

$$u = \theta_1 u_c - \theta_2 y - \theta_3 \dot{y}$$

4. **Adaptation Mechanism**:

- Adjusts the parameters $(\theta_1, \theta_2, \theta_3)$ to minimize the error (e) over time.

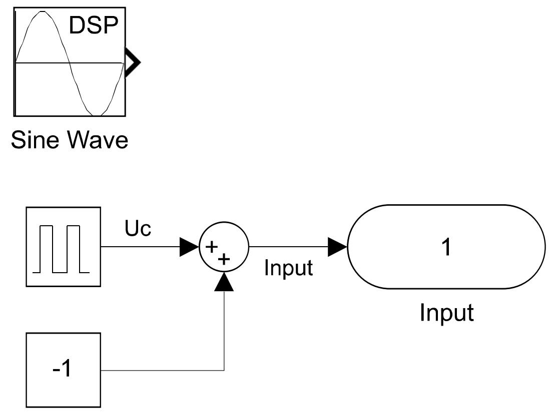


Figure 2: Choosing Input Subsystem (MIT)

The "Choosing Input" subsystem is responsible for selecting the control input signal (u_c) for the system.

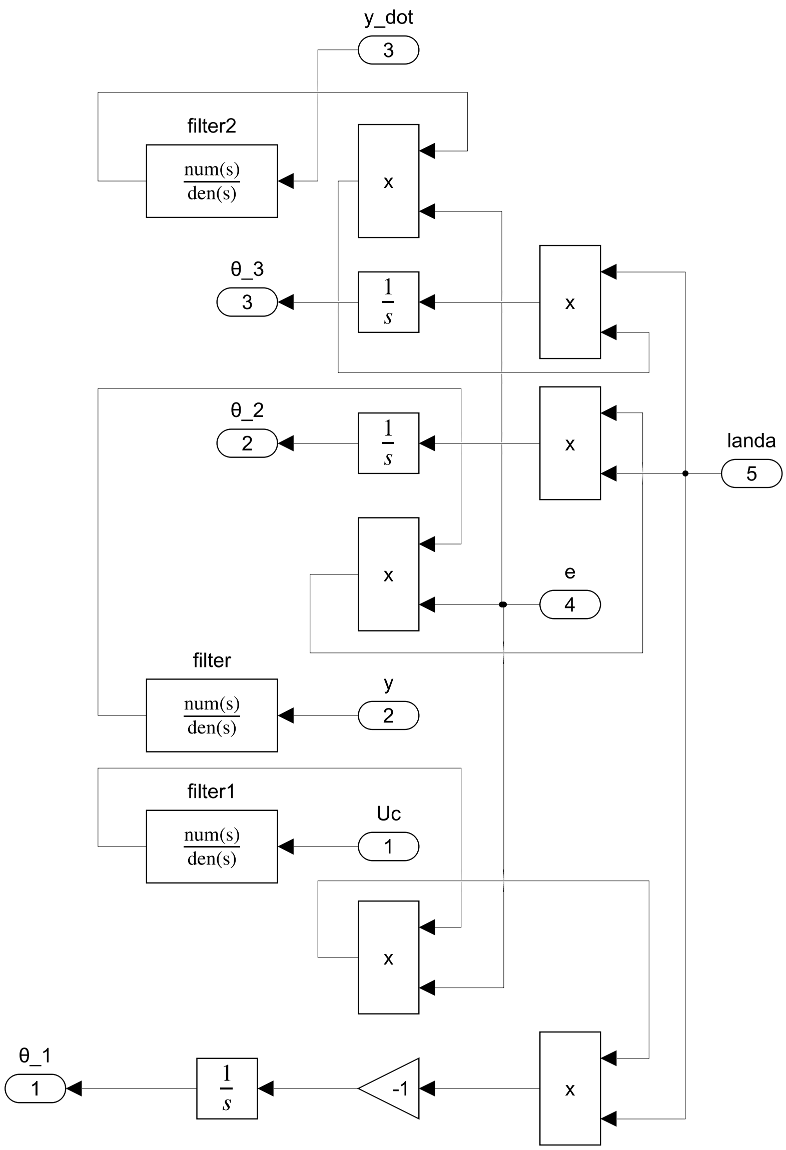


Figure 3: Adaptation Subsystem (MIT)

The "Adaptation" subsystem dynamically adjusts the parameters of the control system to minimize the error between the desired output (y_m) and the actual output (y). This is done by continuously updating the parameters ($\theta_1, \theta_2, \theta_3$) based on the adaptation laws.

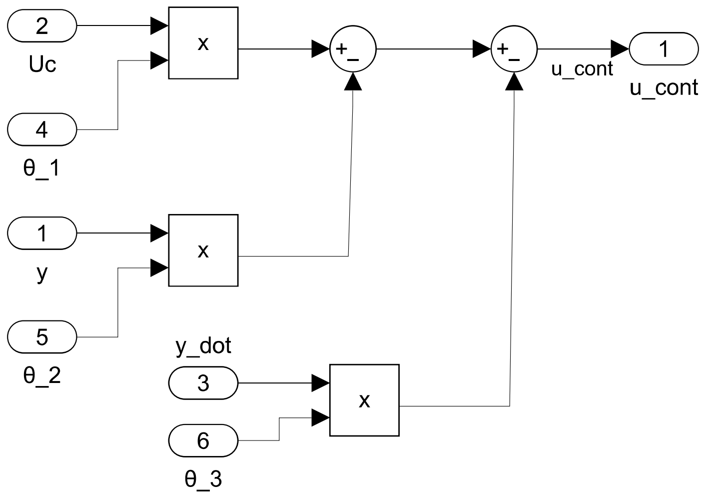


Figure 4: Controller Subsystem (MIT)

The "Controller" subsystem generates the control signal (u) based on the current system states and the error signal (e). The control law implemented in this subsystem takes the form $u = \theta_1 u_c - \theta_2 y - \theta_3 \dot{y}$.

4 MIT Normalized Gradient Descent Implementation

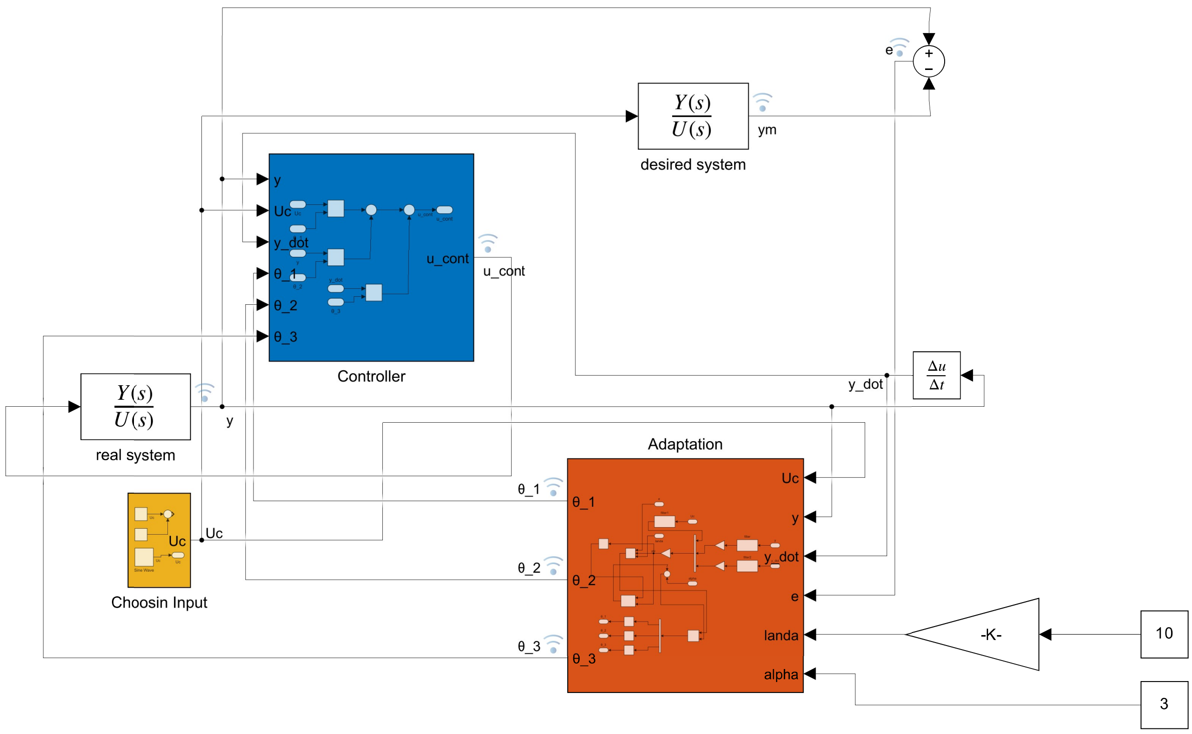


Figure 5: Simulink Model for MIT Normalized Gradient Descent Method

The whole Simulink model for the MIT normalized gradient descent method demonstrates the entire control system, including all subsystems working together to achieve the desired performance.

The "Choosing Input" subsystem is responsible for selecting the appropriate control input signal (u_c) for the system.

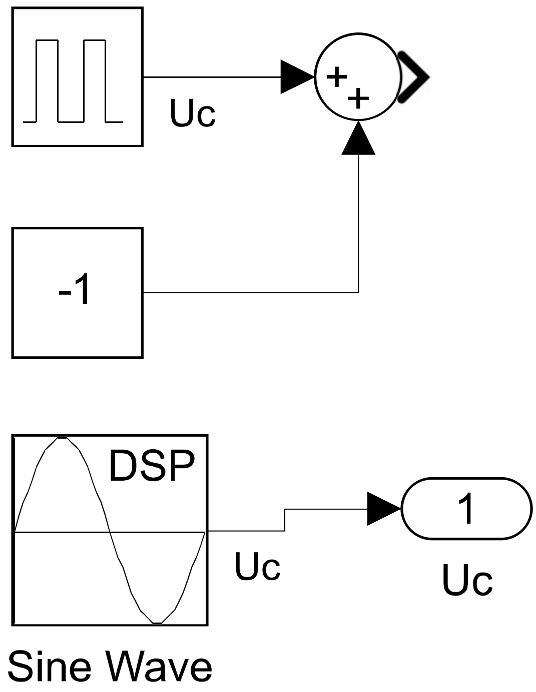


Figure 6: Choosing Input Subsystem (Normalized MIT)

The "Adaptation" subsystem dynamically adjusts the parameters of the control system to minimize the error between the desired output (y_m) and the actual output (y). This is done by continuously updating the parameters ($\theta_1, \theta_2, \theta_3$) based on the adaptation laws. In this section, it is necessary to derive the adaptation law in the proper format. Therefore, for calculation purposes, we need to modify the relationship in the adaptation block in the Simulink model:

$$\frac{d\theta}{dt} = \frac{\gamma \varphi e}{\alpha + \varphi^T \varphi}$$

where φ is a vector. Hence, the operation should be performed as follows. The modified block should be designed as shown below:

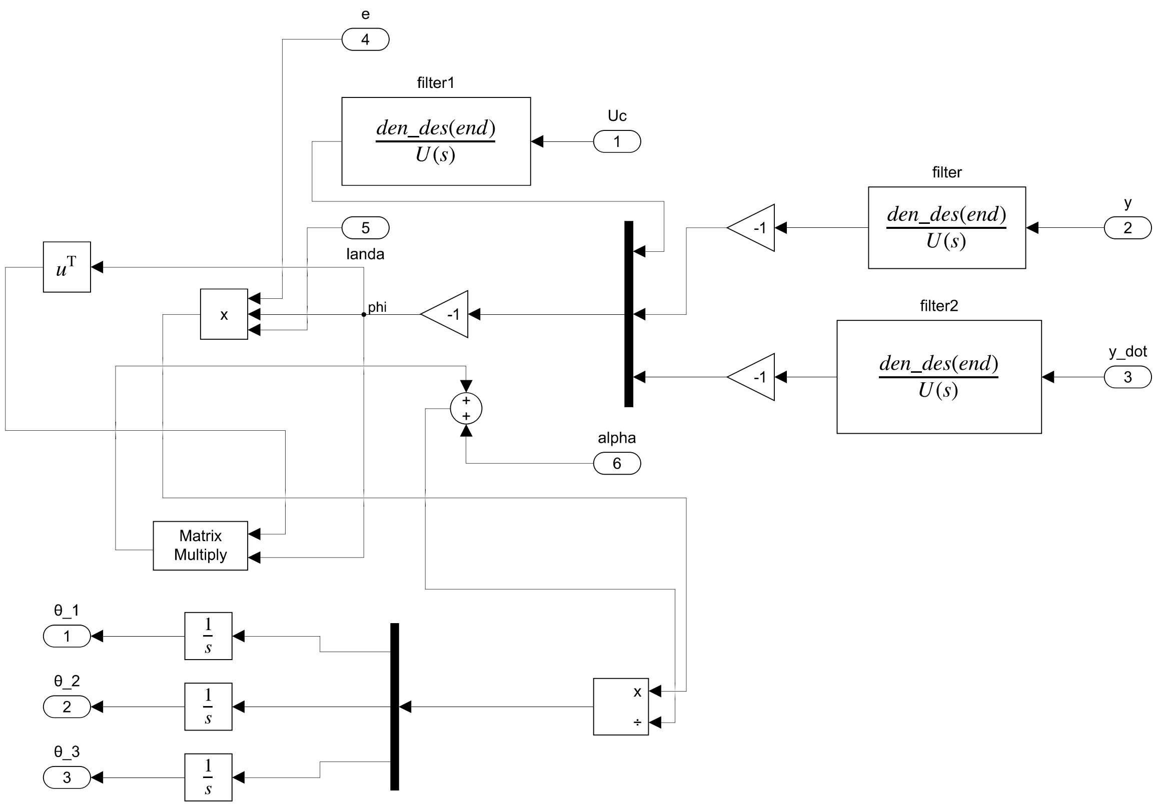


Figure 7: Adaptation Subsystem (Normalized MIT)

The "Controller" subsystem generates the control signal (u) based on the current system states and the error signal (e). The control law implemented in this subsystem takes the form $u = \theta_1 u_c - \theta_2 y - \theta_3 \dot{y}$. This control signal is then used to drive the system towards the desired state.

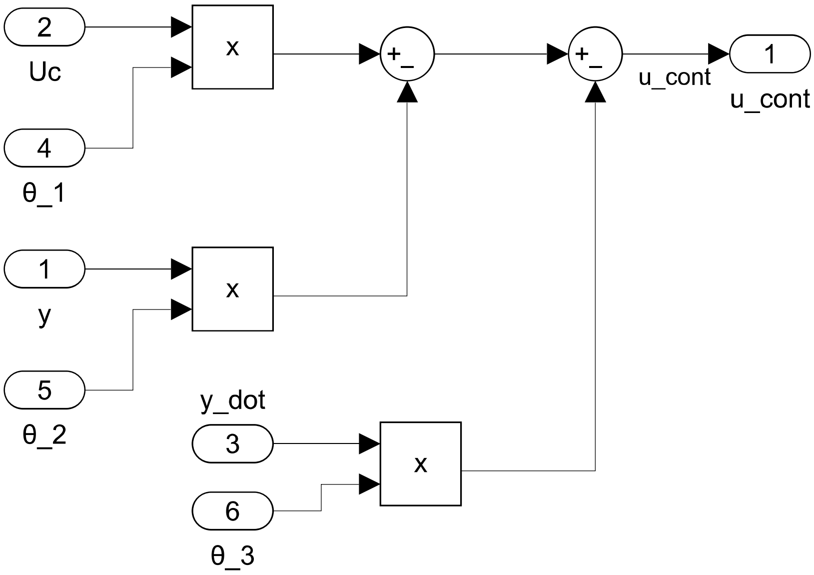


Figure 8: Controller Subsystem (Normalized MIT)

5 Lyapanov MRAS

The parts related to the remaining controller are the same. In this section, it is necessary to first derive the relationship related to the error (e):

$$e = y - y_m$$

By differentiating this value once, we get:

$$\dot{e} = \dot{y} - \dot{y}_m$$

Since the order of the system is 2, we must differentiate this value once more:

$$\ddot{e} = \ddot{y} - \ddot{y}_m$$

Thus, we have:

$$\ddot{y} = -(b\theta_3 + a_1)\dot{y} - (b\theta_2 + a_2)y + (b\theta_1)u_c$$

$$\ddot{y}_m = -a_{m1}\dot{y}_m - a_{m2}y_m + b_m u_c$$

Therefore, the value of $\ddot{e}$ will be:

$$\ddot{e} = -(b\theta_3 + a_1)\dot{y} - (b\theta_2 + a_2)y + (b\theta_1)u_c - a_{m1}\dot{y}_m - a_{m2}y_m + b_m u_c$$

So we can rewrite y and y_m in terms of e:

$$\ddot{e} = -(b\theta_3 + a_1)\dot{y} - (b\theta_2 + a_2)y + (b\theta_1)u_c + a_{m1}(\dot{y} - \dot{e}) + a_{m2}(y - e) - b_m u_c$$

$$= -a_{m1}\dot{e} - a_{m2}e + (a_{m1} - b\theta_3 - a_1)\dot{y} + (a_{m2} - b\theta_2 - a_2)y + (b\theta_1 - b_m)u_c$$

Now we define the Lyapunov function in the following form:

$$V(e, \dot{e}, \theta_1, \theta_2, \theta_3) = \frac{1}{2}(e^2 + \dot{e}^2 + \frac{(a_{m1} - b\theta_3 - a_1)^2}{\gamma} + \frac{(a_{m2} - b\theta_2 - a_2)^2}{\gamma} + \frac{(b\theta_1 - b_m)^2}{\gamma})$$

and we have:

$$\frac{dV}{dt} = e\dot{e} + \dot{e}\ddot{e} - \frac{(a_{m1} - b\theta_3 - a_1)}{\gamma}\frac{d\theta_3}{dt} - \frac{(a_{m2} - b\theta_2 - a_2)}{\gamma}\frac{d\theta_2}{dt} + \frac{(b\theta_1 - b_m)}{\gamma}\frac{d\theta_1}{dt}$$

$$\frac{dV}{dt} = (1 - a_{m2})e\dot{e} - a_{m1}\dot{e}^2 - \frac{(a_{m1} - b\theta_3 - a_1)}{\gamma}\left(\frac{d\theta_3}{dt} - \dot{y}\dot{e}\right) -$$
$$\frac{(a_{m2} - b\theta_2 - a_2)}{\gamma}\left(\frac{d\theta_2}{dt} - y\dot{e}\right) + \frac{(b\theta_1 - b_m)}{\gamma}\left(\frac{d\theta_1}{dt} + \gamma e u_c\right)$$

Thus, for this value to always be negative, we must have:

$$\frac{d\theta_1}{dt} = -\gamma e u_c$$
$$\frac{d\theta_2}{dt} = y\dot{e}$$
$$\frac{d\theta_3}{dt} = \dot{y}\dot{e}$$

And the remaining terms are zero.

$$\frac{dV}{dt} = (1 - a_{m2})e\dot{e} - a_{m1}\dot{e}^2 = \dot{e}(1 - a_{m2}e - a_{m1}\dot{e})$$
$$= \dot{e}(1 - a_{m1}(\dot{y}_m - \dot{e}) - a_{m2}(y_m - y))$$
$$= \dot{e}(1 - \dot{y}_m + \dot{y} - u_c(b - b_m))$$

Since the values of e, u_c, and $\dot{y}$ are bounded, we can say that the second derivative of V is also bounded.

$$\frac{d^2V}{dt^2} = (1 - a_{m2})(e^2 + \dot{e}^2) - 2a_{m1}\dot{e}\ddot{e}$$

If this value is bounded, according to Theorem 5.4 from the book by Astrom, it can be said that the function V is bounded and the value of e tends to zero over time, even though the system parameters do not converge to their true values.

6 Lyapanov MRAS Implementation

The whole Simulink model for the Lyapunov Model Reference Adaptive System (MRAS) is shown as follows. The primary components include the reference model, the plant, the adaptation mechanism, and the controller.

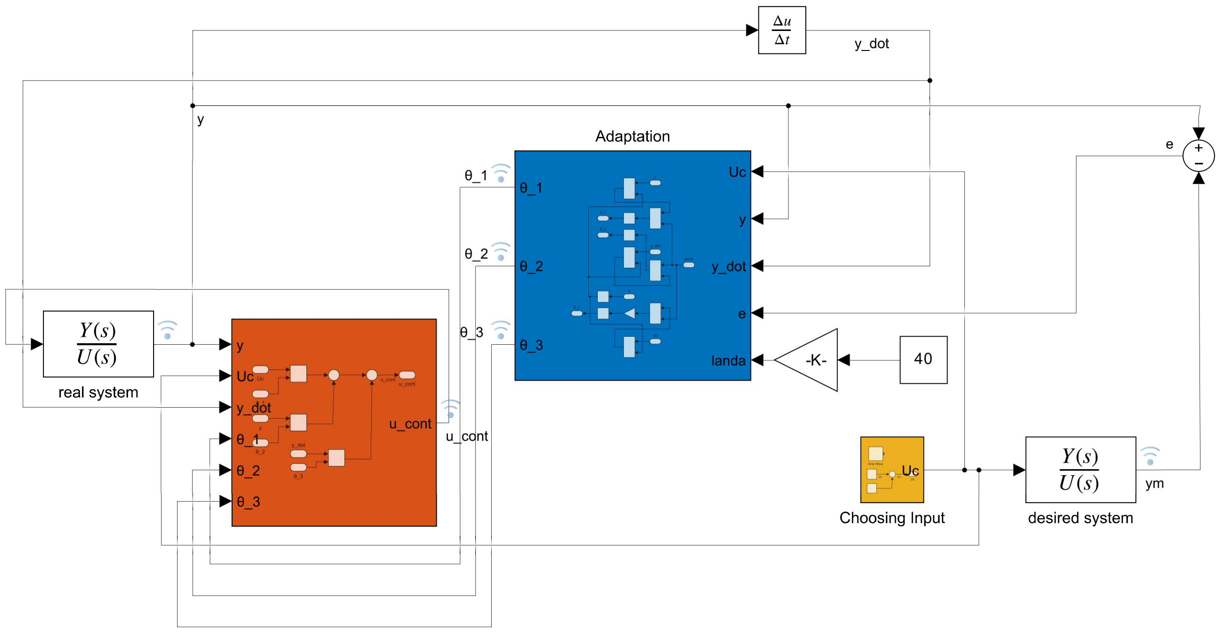


Figure 9: Whole Simulink Model for Lyapunov MRAS

The "Adaptation" subsystem within the Lyapunov MRAS plays a critical role in the adaptive control process. It uses the principles of Lyapunov's direct method to derive the adaptation laws. These laws dictate how the parameters of the control system should be updated to ensure that the system error converges to zero. The adaptation mechanism monitors the error between the reference model output and the actual system output. Based on this error and its derivatives, the subsystem calculates the necessary adjustments to the parameters $(\theta_1, \theta_2, \theta_3)$.

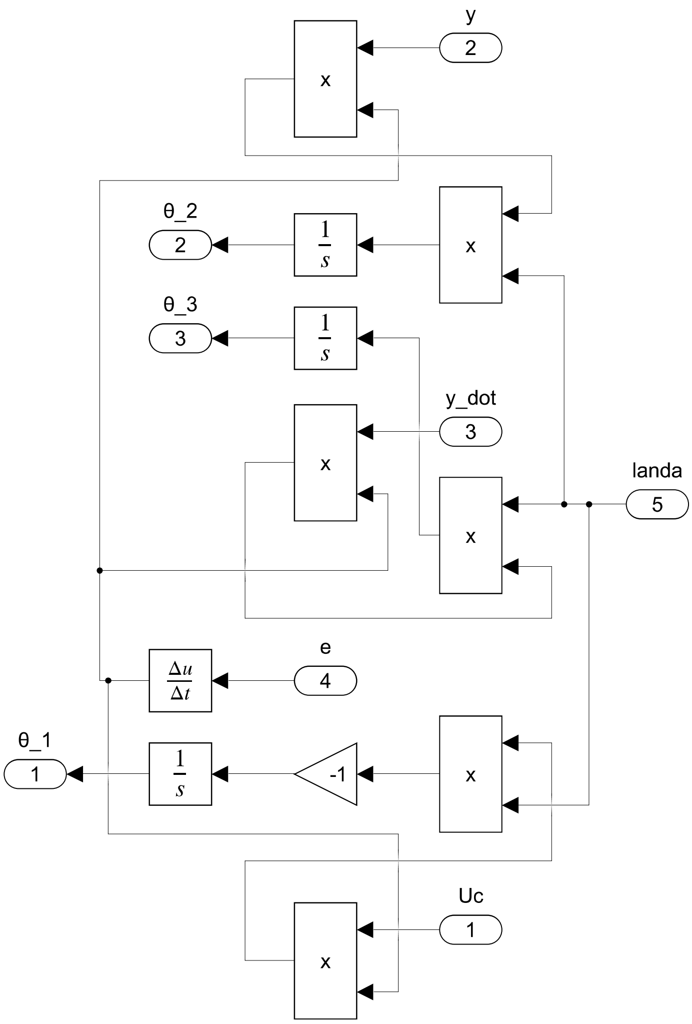


Figure 10: Adaptation Subsystem for Lyapunov MRAS

7 Simulation Results Sine Input Reference

In this section the simulation results for different methods which were implemented in the previous sections is brought. We will apply the changes in the Simulink model and use data logger in order to get the signals.

7.1 λ And α Impact On The Controller Performance

We will change the value for λ and α (for normalized MIT) and compare the output and desired output and the control effort and also the convergence of the system parameters.

7.1.1 MIT $\lambda = 20$

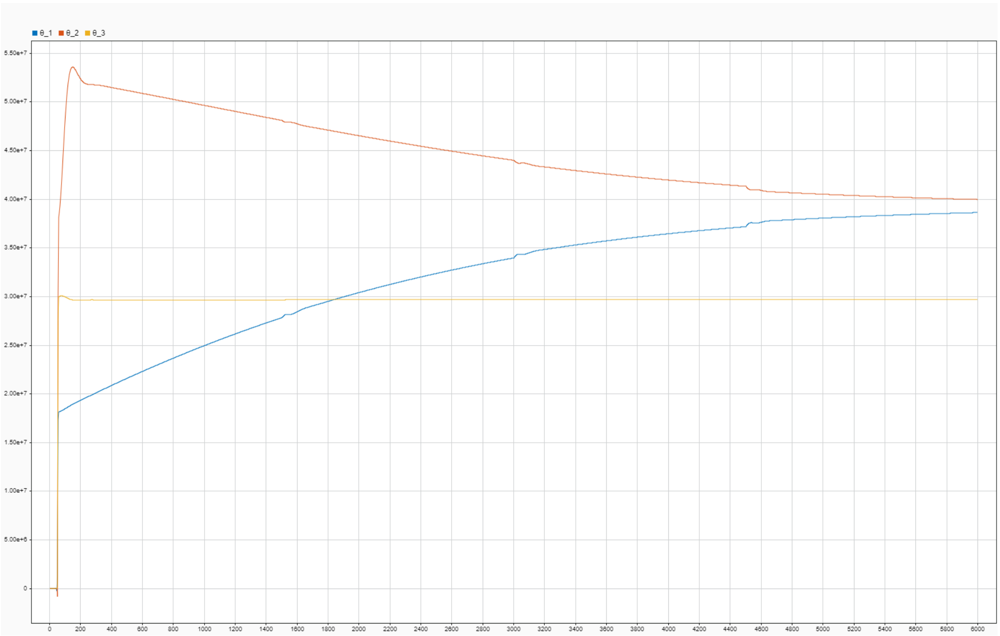


Figure 11: Parameters convergence (MIT, $\lambda = 20$)

As we can see the parameters converged very good.

Figure 12: Control effort (MIT, $\lambda = 20$)

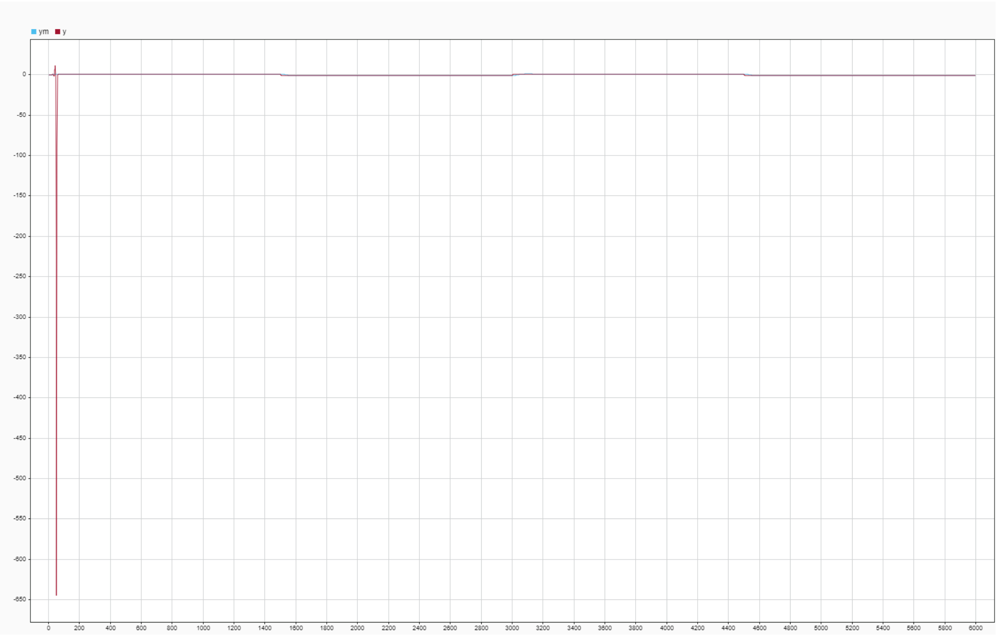


Figure 13: System's output Vs. Desired Output (MIT, $\lambda = 20$)

7.1.2 MIT $\lambda = 60$

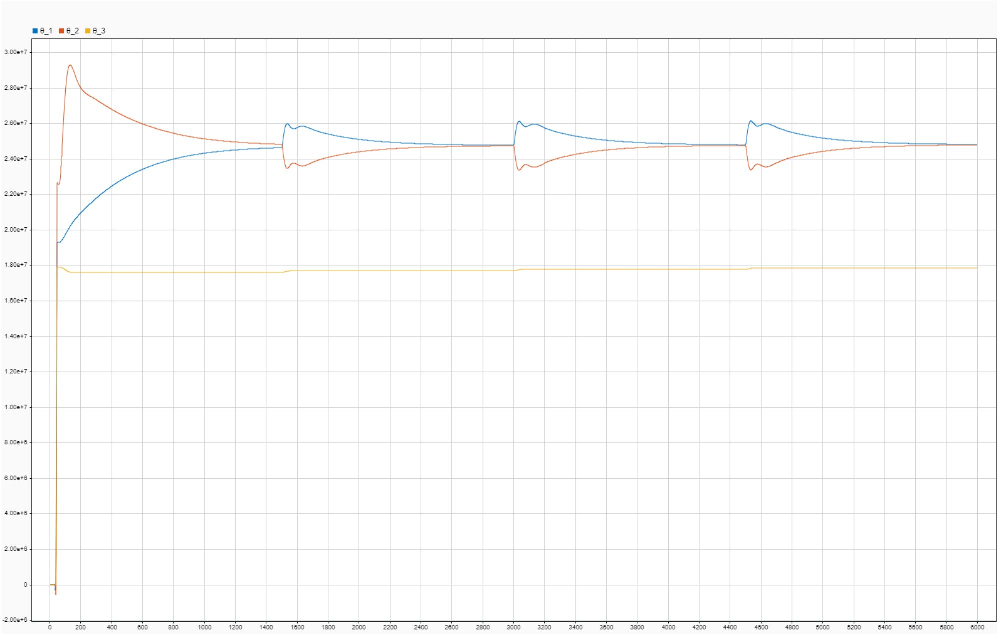


Figure 14: Parameters convergence (MIT, $\lambda = 60$)

As we can see the parameters converged very good. increasing the value for λ will make the parameter estimation and control effort better. The tracking however has got better but the main thing is the overshoot reduction the tracking at the rest times is not that different both $\lambda = 20$ and $\lambda = 60$ have good tracking.

Figure 15: Control effort (MIT, $\lambda = 60$)

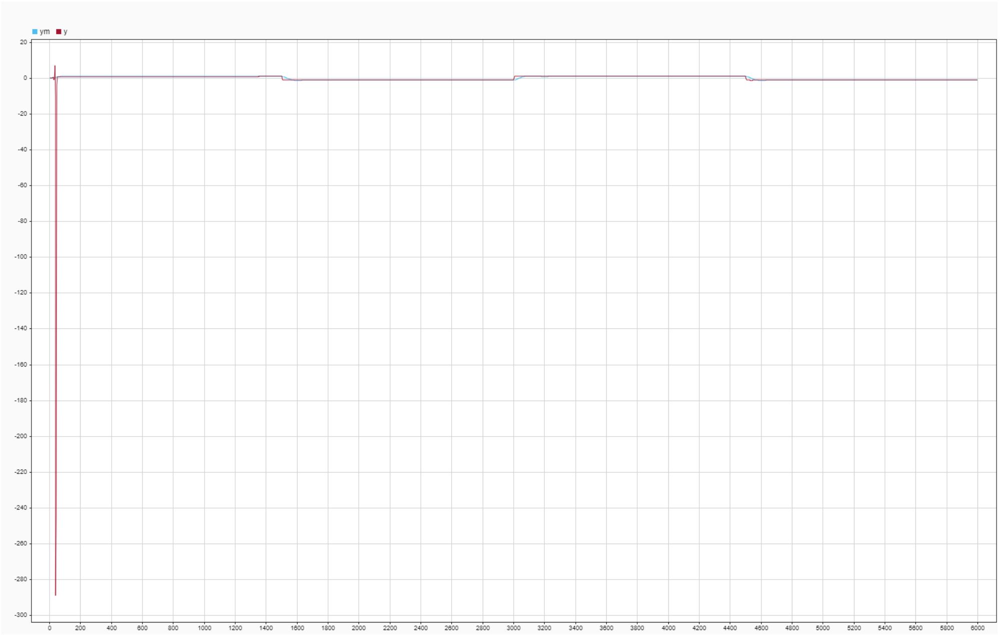


Figure 16: System's output Vs. Desired Output (MIT, $\lambda = 60$)

7.1.3 Normalized MIT $\lambda = 20$ And $\alpha = 3$

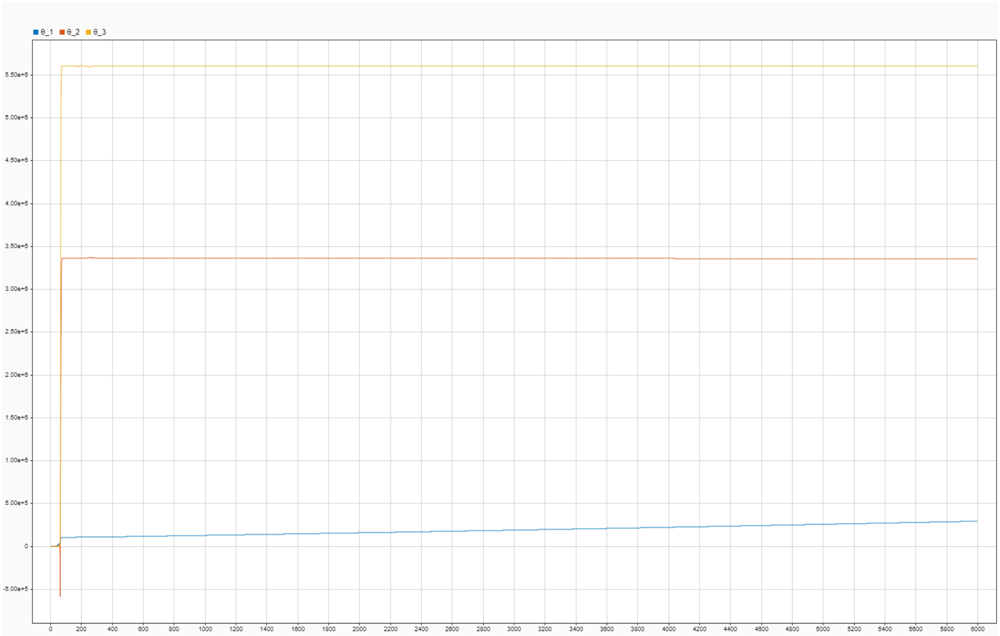


Figure 17: Parameters convergence (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

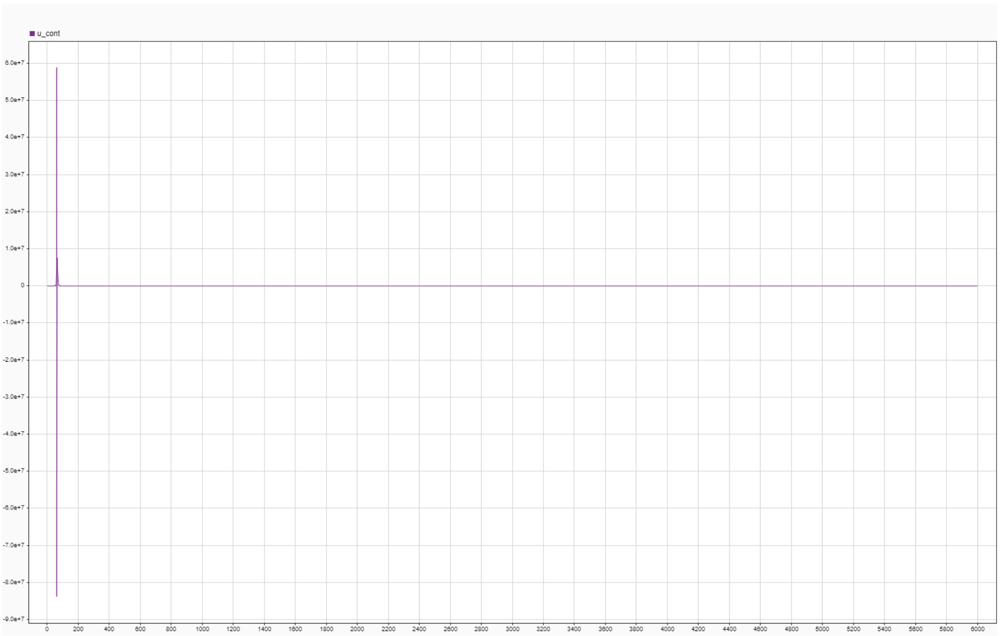


Figure 18: Control effort (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

Figure 19: System's output Vs. Desired Output (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

7.1.4 Normalized MIT $\lambda = 60$ And $\alpha = 3$

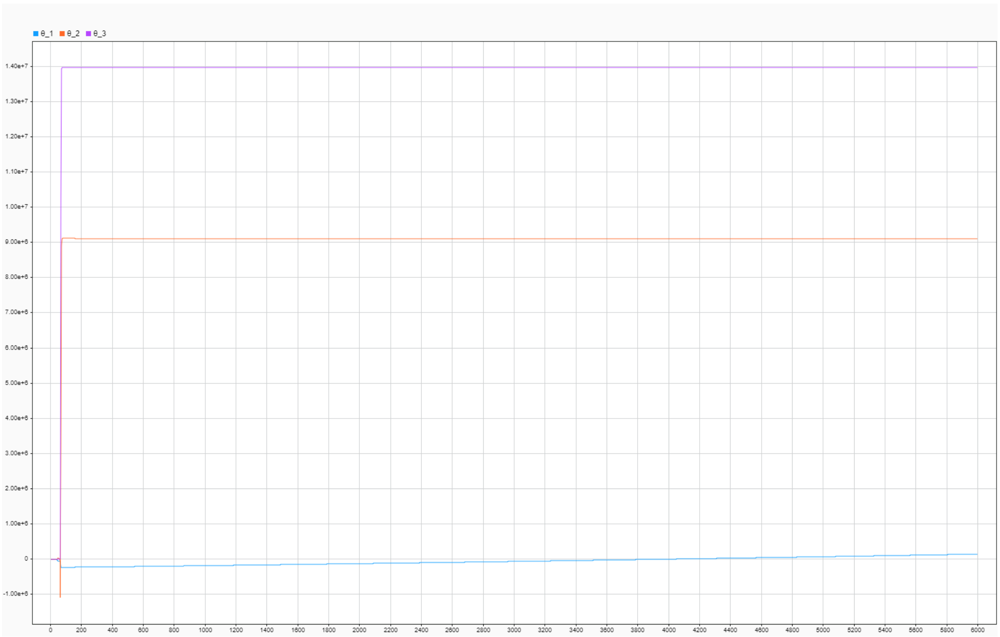


Figure 20: Parameters convergence (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

Figure 21: Control effort (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

Figure 22: System's output Vs. Desired Output (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

7.1.5 Normalized MIT $\lambda = 20$ And $\alpha = 6$

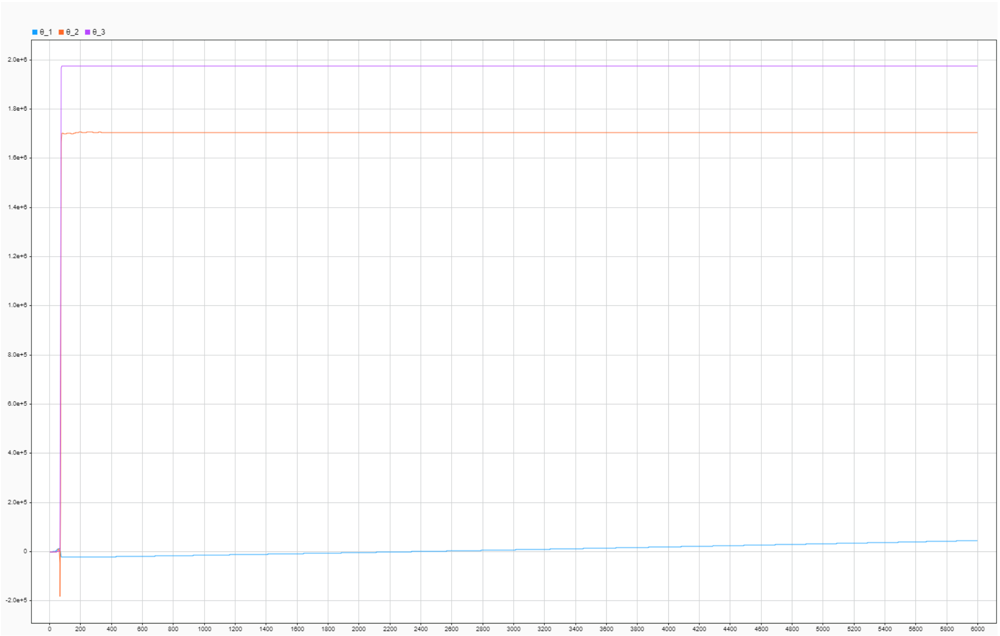


Figure 23: Parameters convergence (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

Figure 24: Control effort (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

Figure 25: System's output Vs. Desired Output (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

7.1.6 Lyapunov MRAS $\lambda = 20$

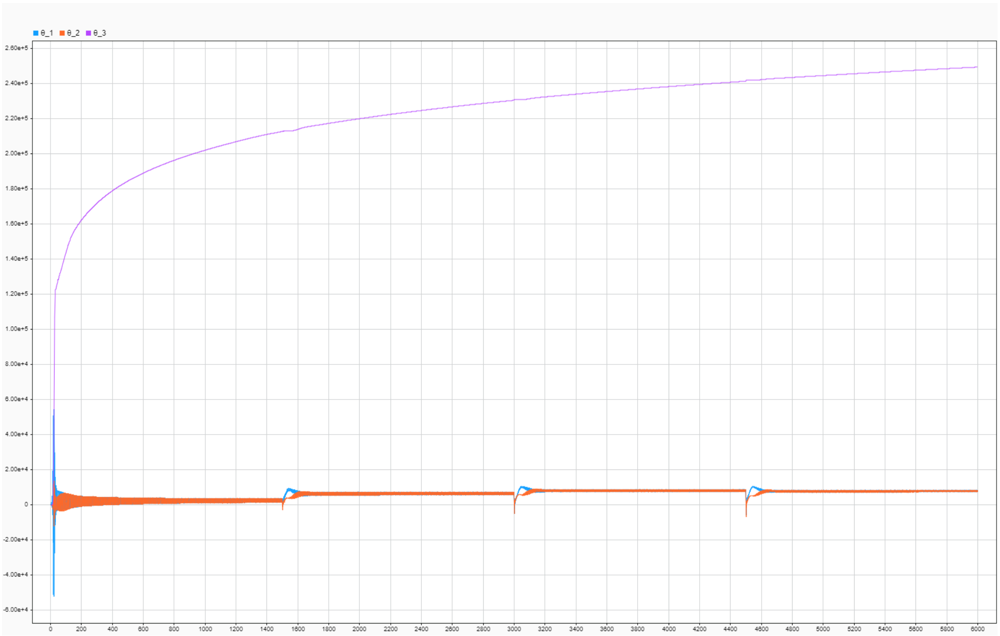


Figure 26: Parameters convergence (Lyapunov MRAS, $\lambda = 20$)

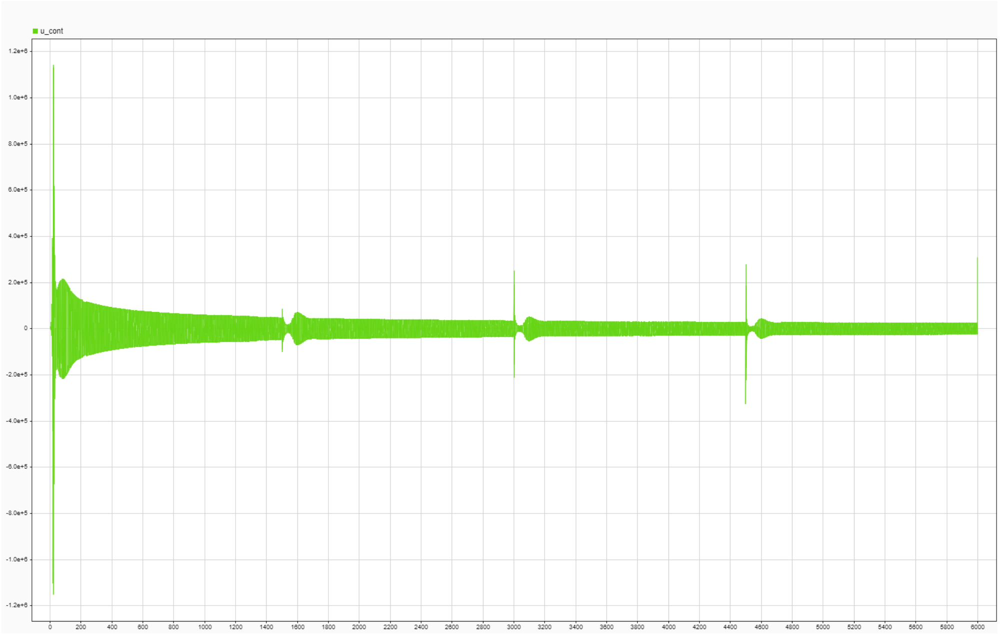


Figure 27: Control effort (Lyapunov MRAS, $\lambda = 20$)

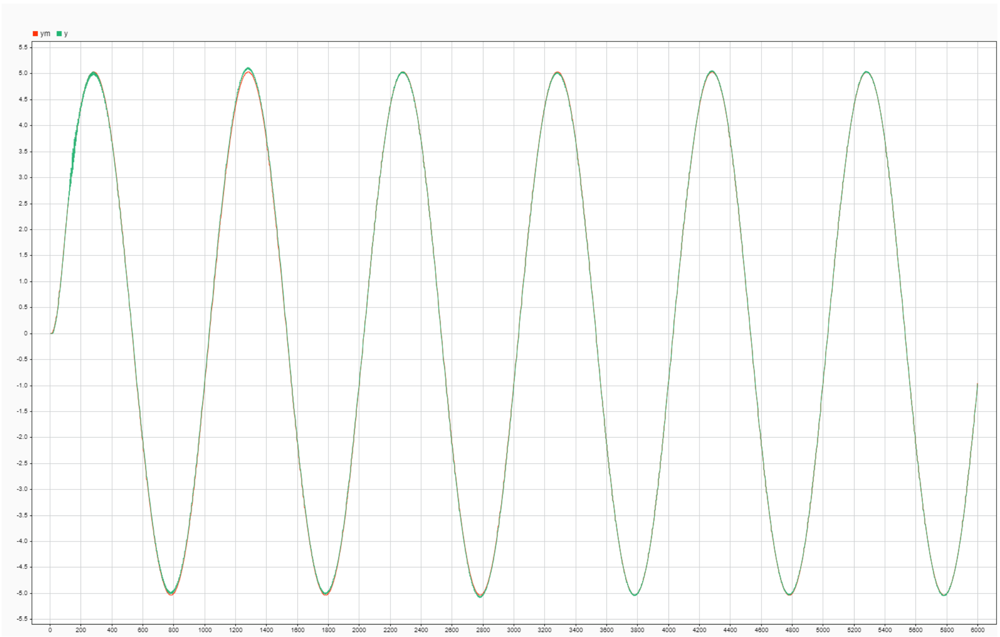


Figure 28: System's output Vs. Desired Output (Lyapunov MRAS, $\lambda = 20$)

7.1.7 Lyapunov MRAS $\lambda = 60$

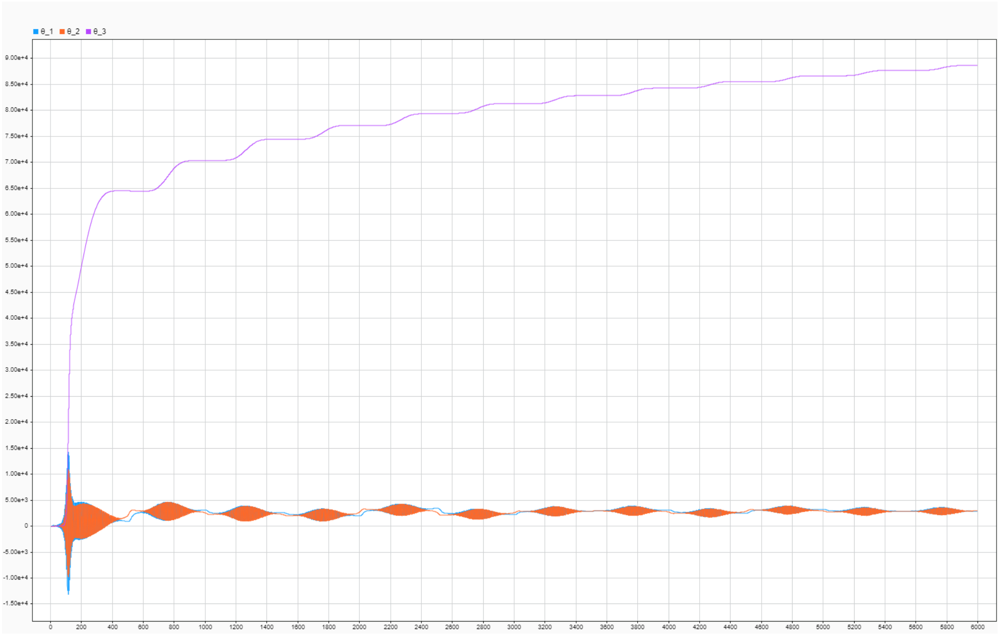


Figure 29: Parameters convergence (Lyapunov MRAS, $\lambda = 60$)

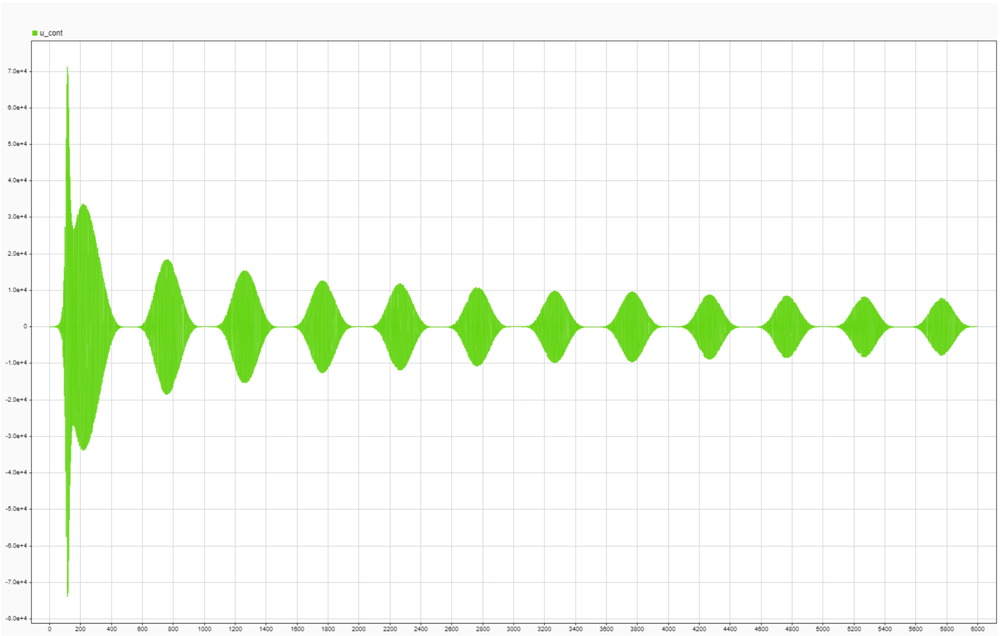


Figure 30: Control effort (Lyapunov MRAS, $\lambda = 60$)

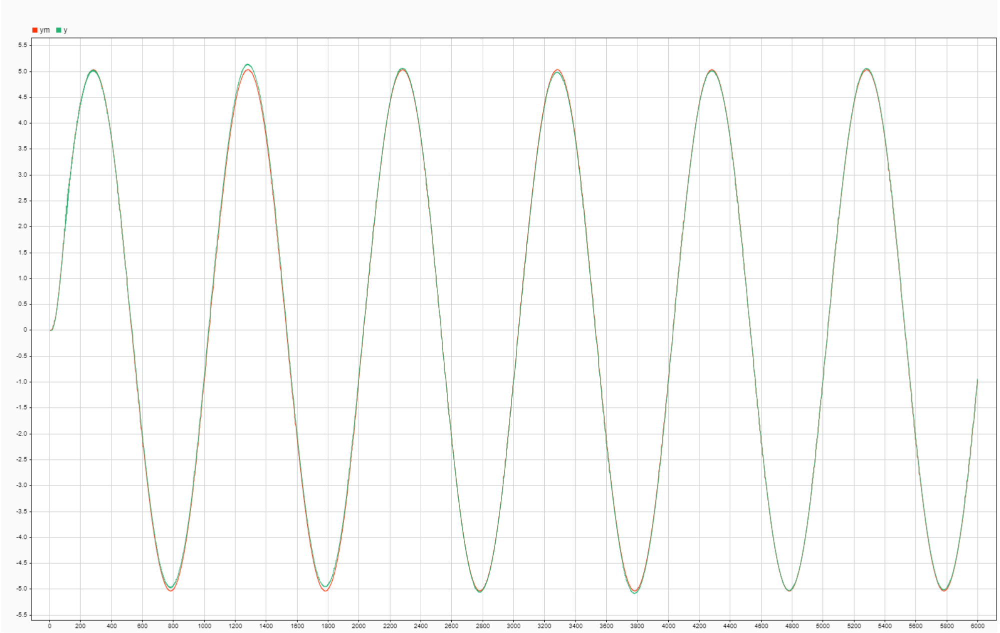


Figure 31: System's output Vs. Desired Output (Lyapunov MRAS, $\lambda = 60$)

8 Simulation Results Square Wave Input Reference

In this section the simulation results for different methods which were implemented in the previous sections is brought. We will apply the changes in the Simulink model and use data logger in order to get the signals.

8.1 λ And α Impact On The Controller Performance

We will change the value for λ and α (for normalized MIT) and compare the output and desired output and the control effort and also the convergence of the system parameters.

8.1.1 MIT $\lambda = 20$

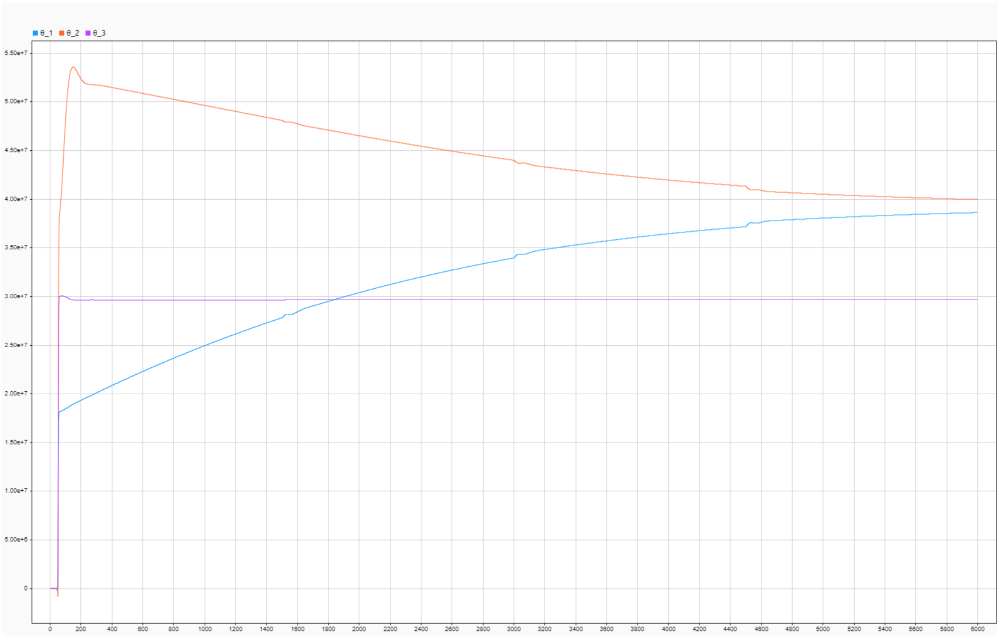


Figure 32: Parameters convergence (MIT, $\lambda = 20$)

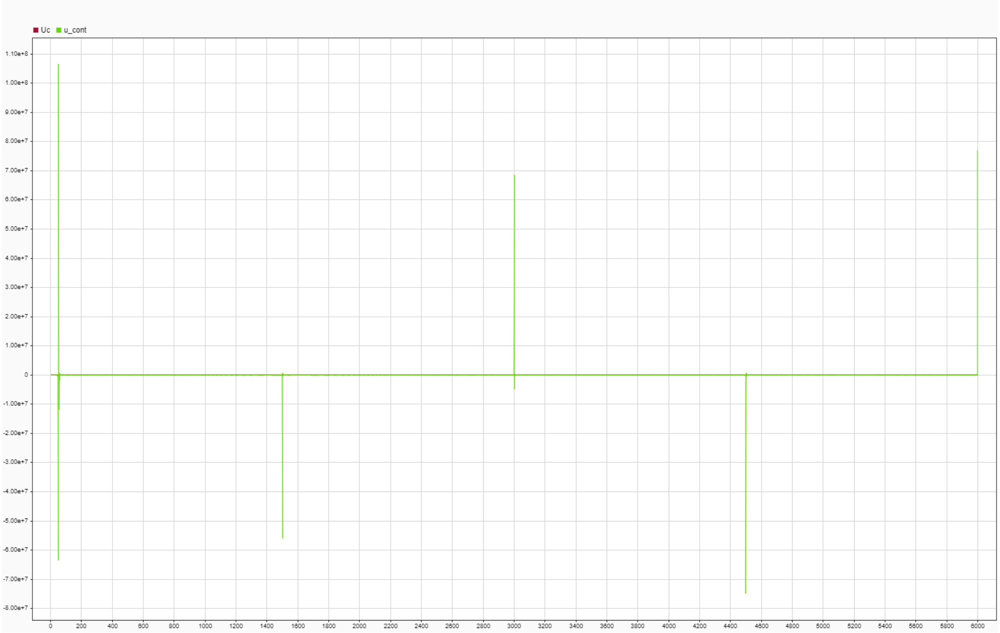


Figure 33: Control effort (MIT, $\lambda = 20$)

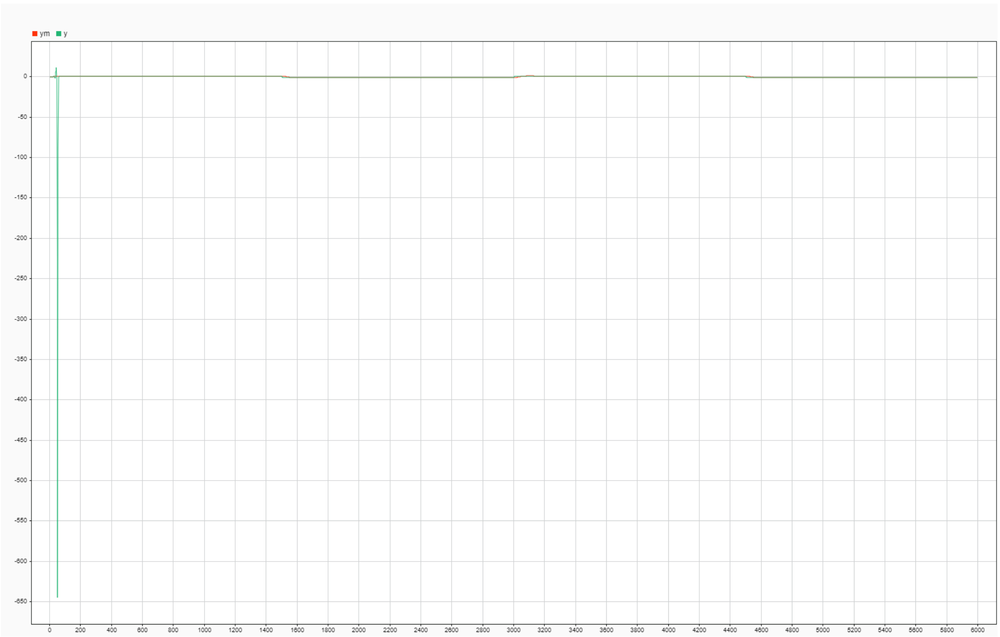


Figure 34: System's output Vs. Desired Output (MIT, $\lambda = 20$)

8.1.2 MIT $\lambda = 60$

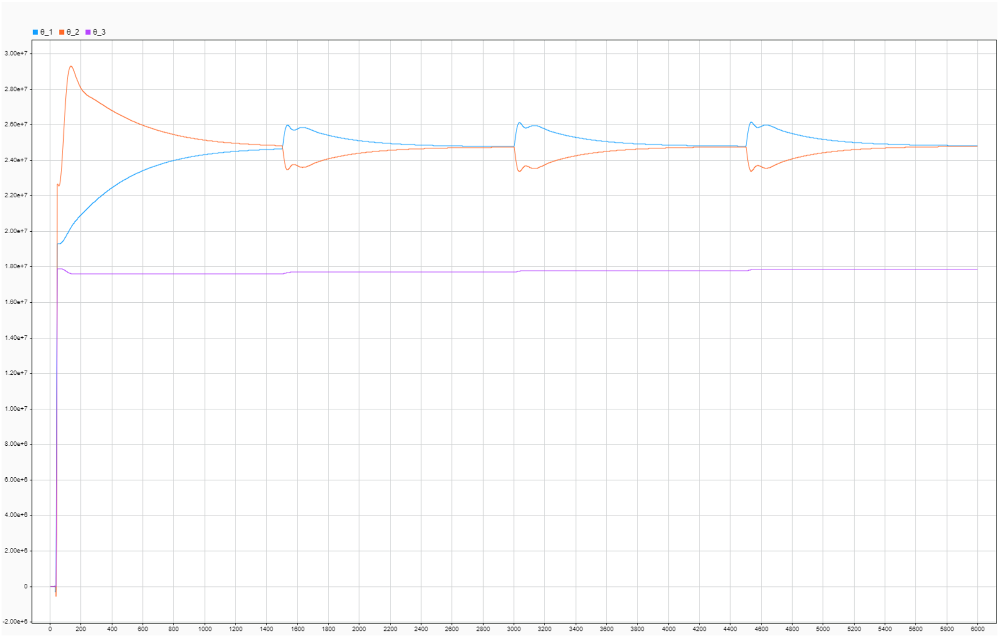


Figure 35: Parameters convergence (MIT, $\lambda = 60$)

Figure 36: Control effort (MIT, $\lambda = 60$)

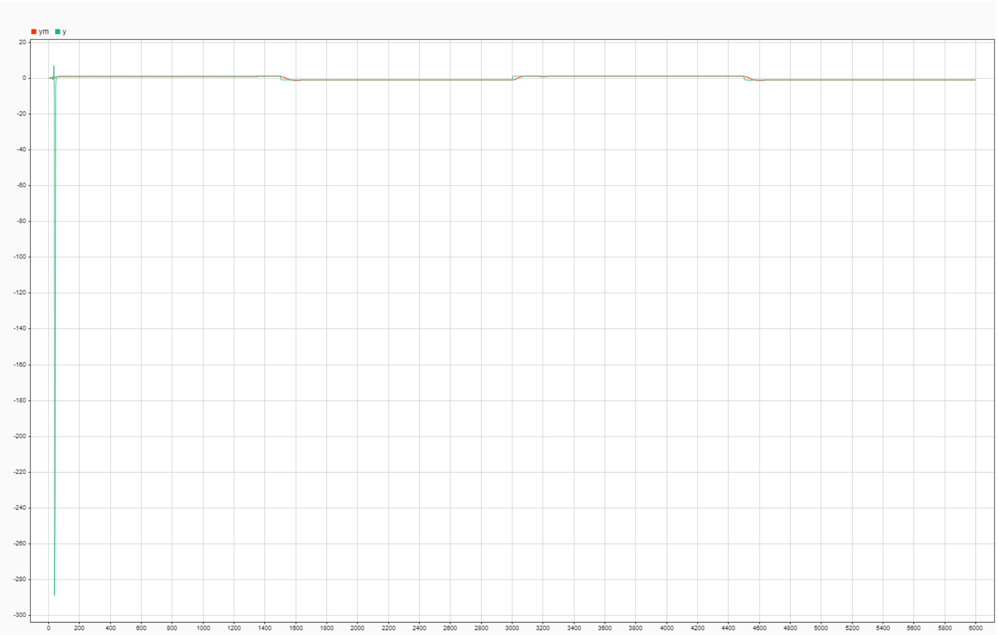


Figure 37: System's output Vs. Desired Output (MIT, $\lambda = 60$)

8.1.3 Normalized MIT $\lambda = 20$ And $\alpha = 3$

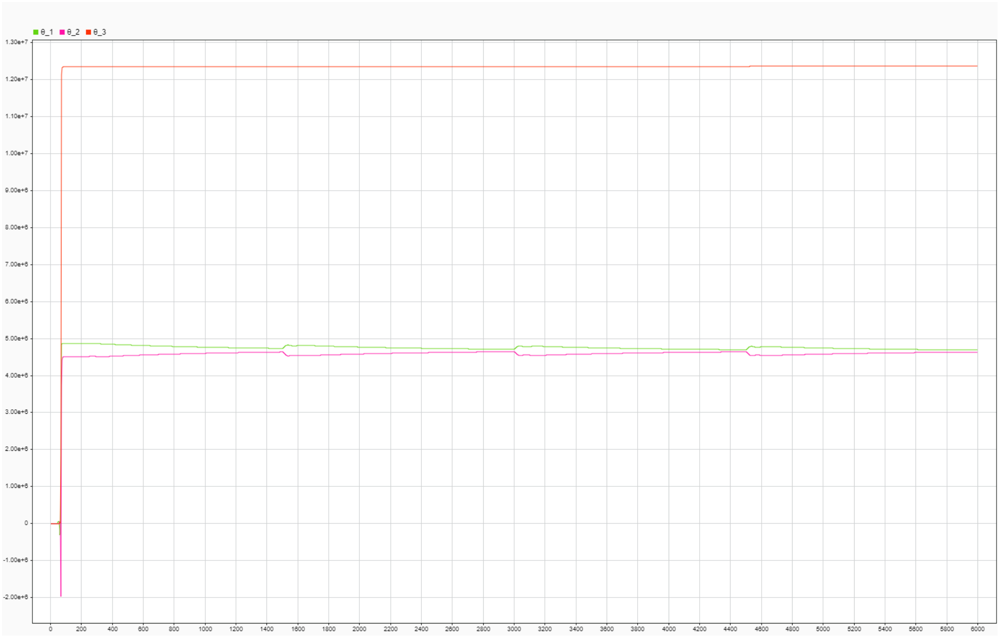


Figure 38: Parameters convergence (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

Figure 39: Control effort (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

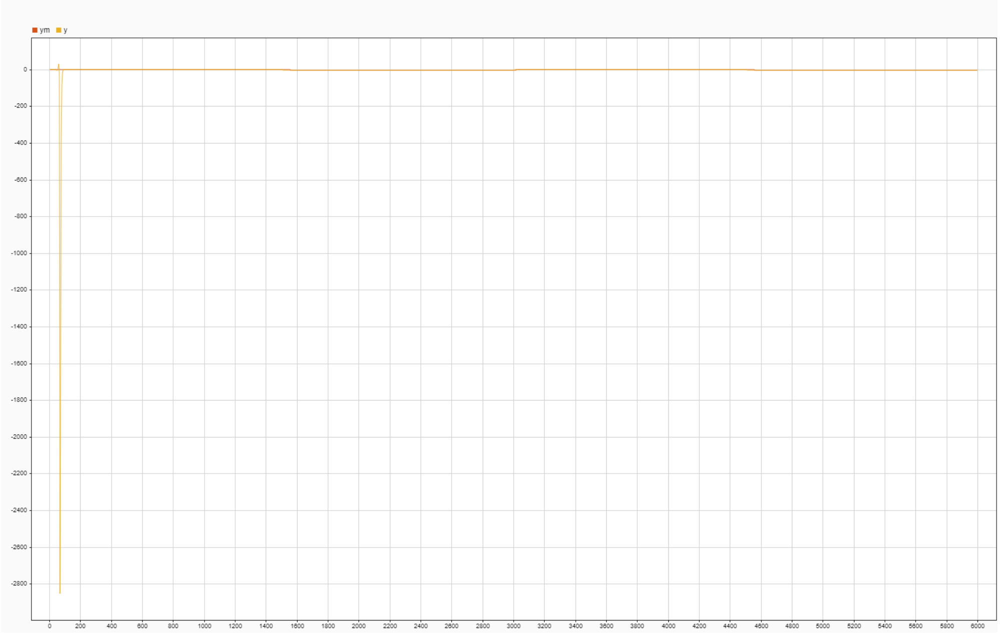


Figure 40: System's output Vs. Desired Output (Normalized MIT, $\lambda = 20$, $\alpha = 3$)

8.1.4 Normalized MIT $\lambda = 60$ And $\alpha = 3$

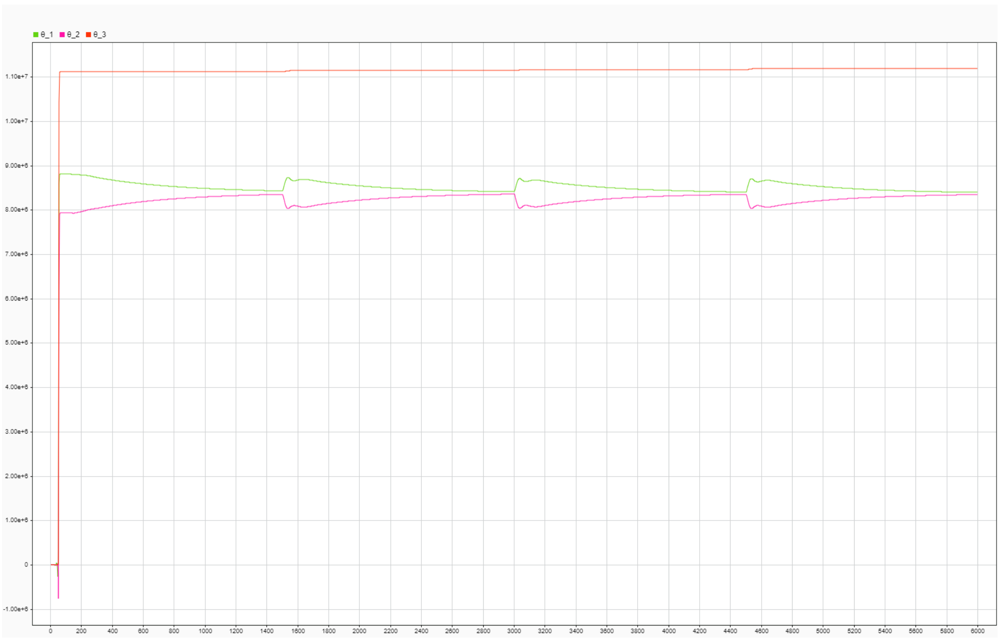


Figure 41: Parameters convergence (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

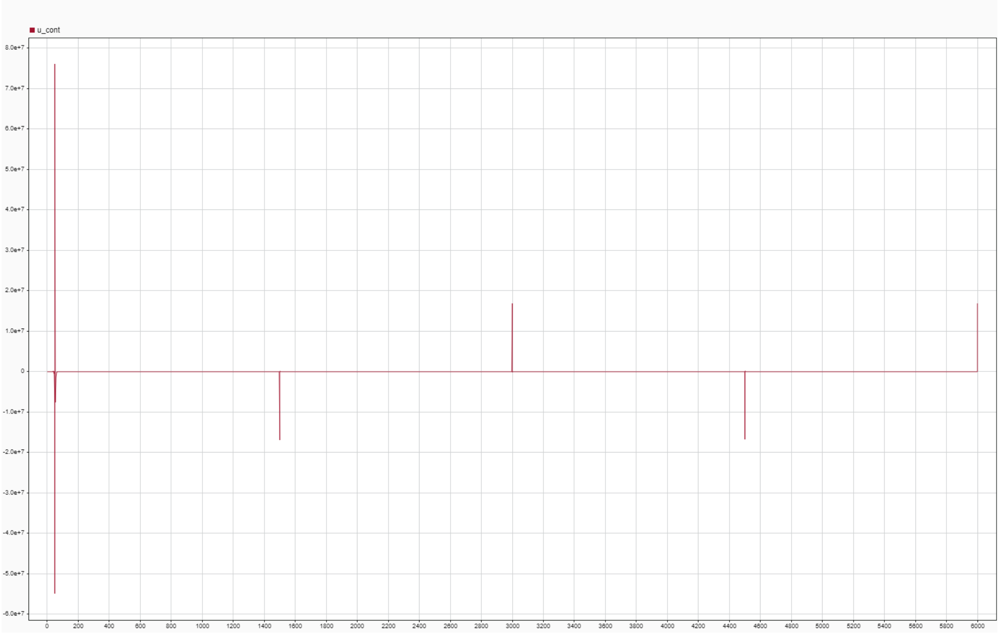


Figure 42: Control effort (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

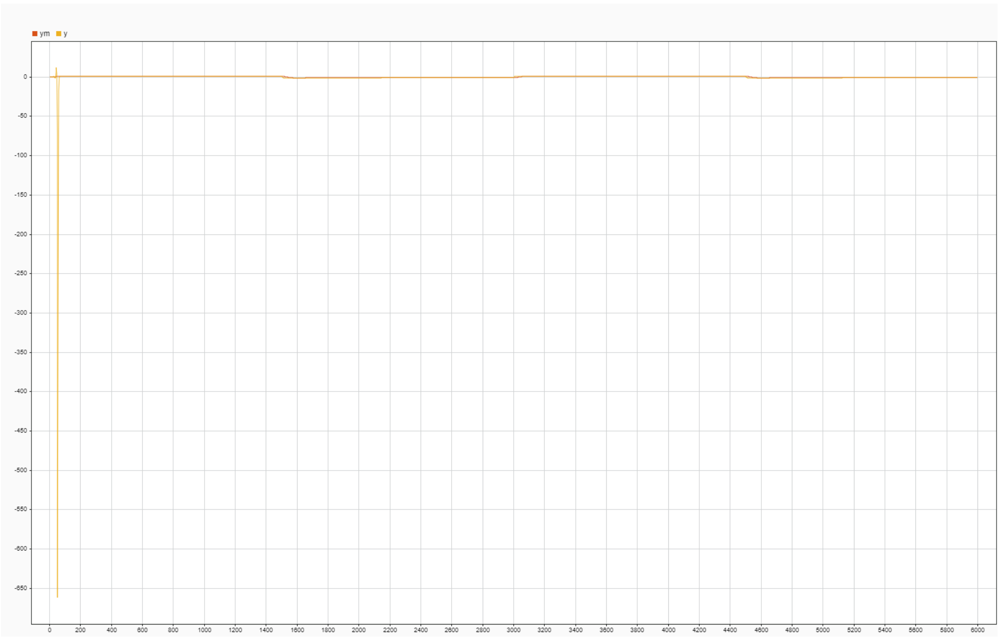


Figure 43: System's output Vs. Desired Output (Normalized MIT, $\lambda = 60$, $\alpha = 3$)

8.1.5 Normalized MIT $\lambda = 20$ And $\alpha = 6$

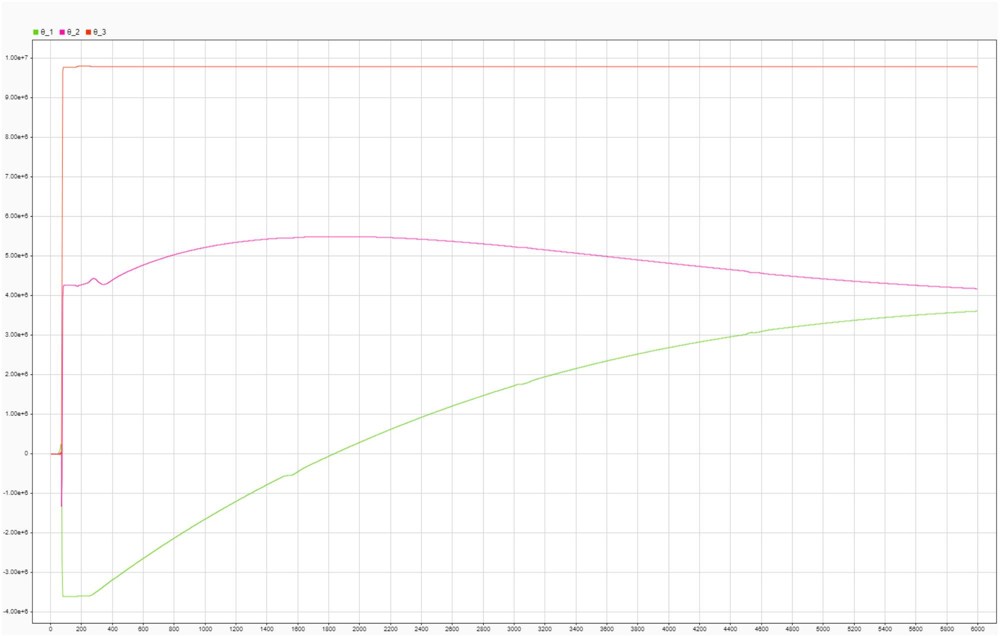


Figure 44: Parameters convergence (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

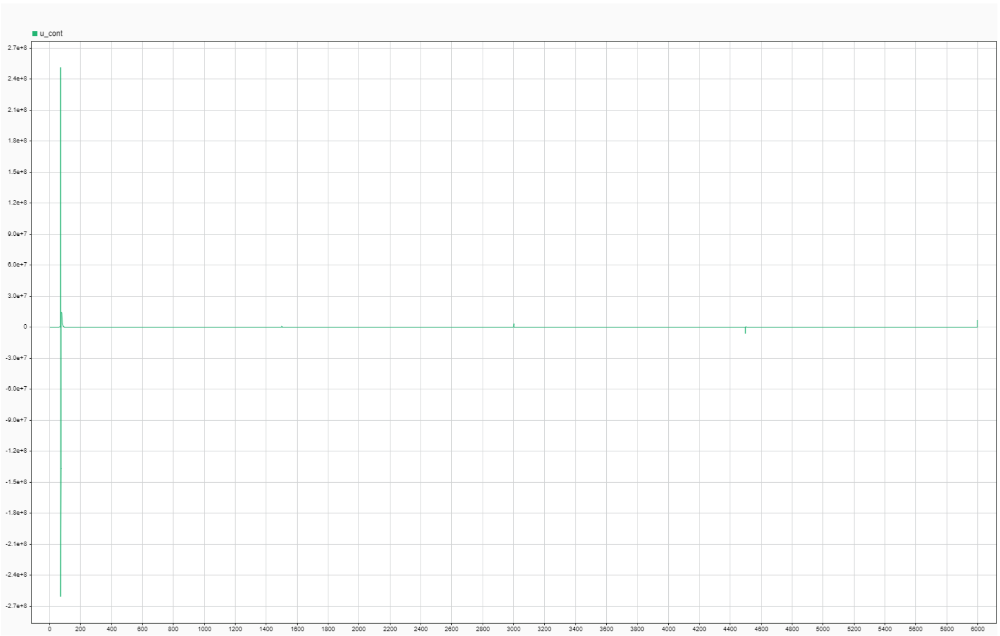


Figure 45: Control effort (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

Figure 46: System's output Vs. Desired Output (Normalized MIT, $\lambda = 20$, $\alpha = 6$)

8.1.6 Lyapunov MRAS $\lambda = 20$

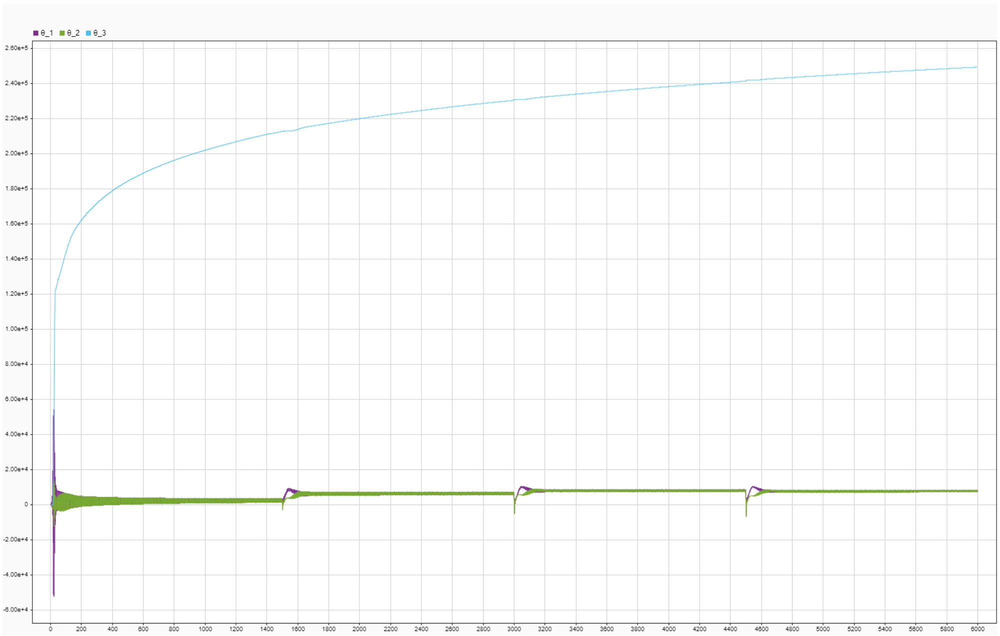


Figure 47: Parameters convergence (Lyapunov MRAS, $\lambda = 20$)

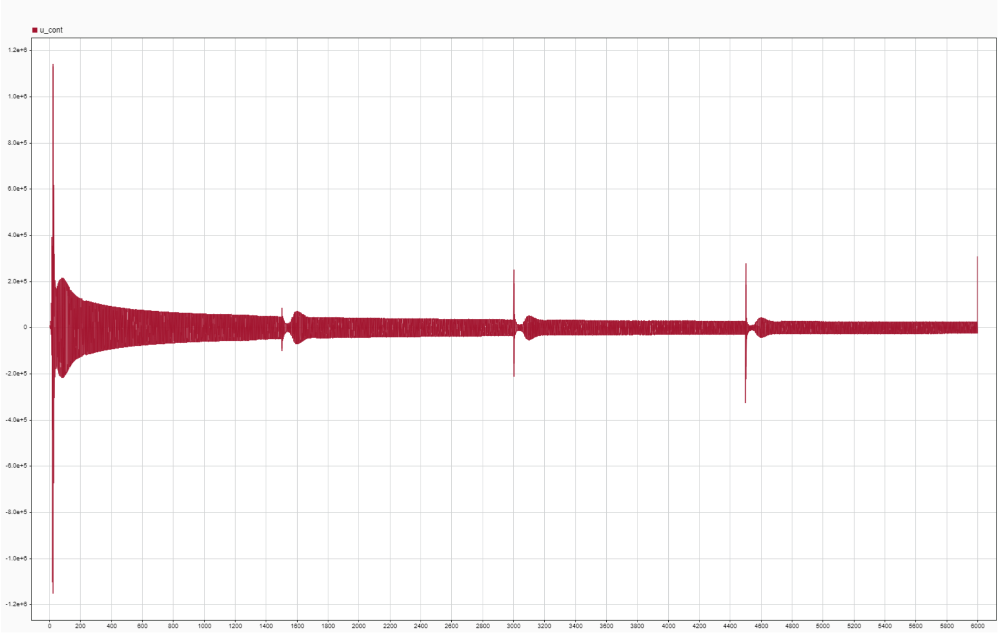


Figure 48: Control effort (Lyapunov MRAS, $\lambda = 20$)

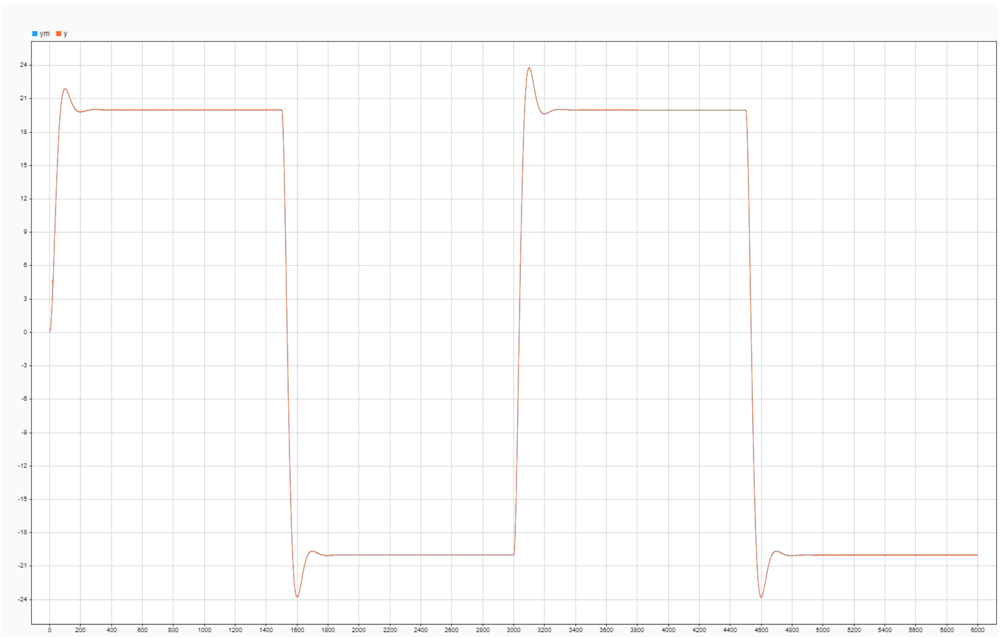


Figure 49: System's output Vs. Desired Output (Lyapunov MRAS, $\lambda = 20$)

8.1.7 Lyapunov MRAS $\lambda = 60$

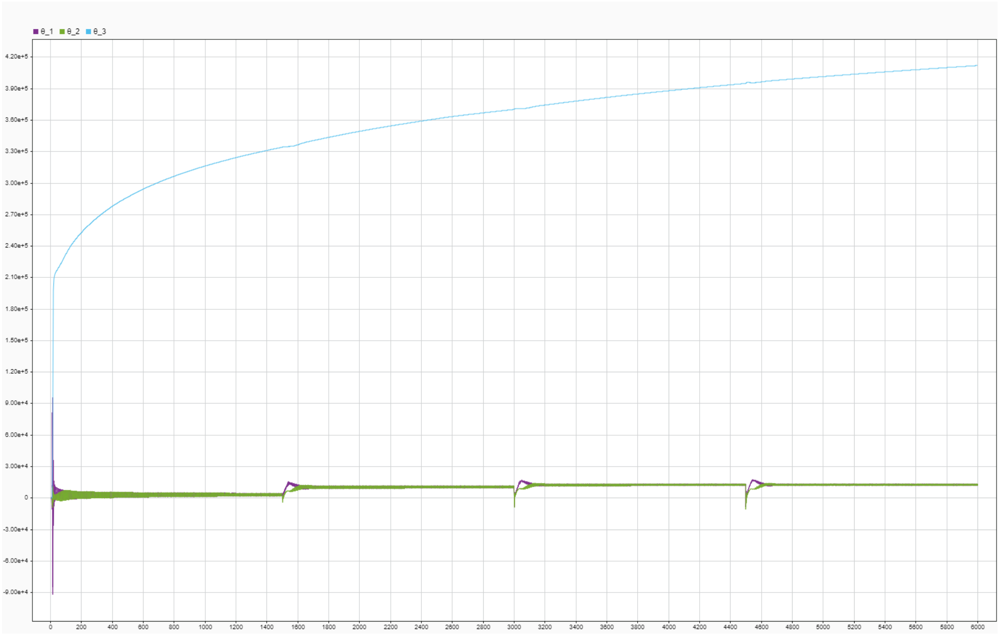


Figure 50: Parameters convergence (Lyapunov MRAS, $\lambda = 60$)

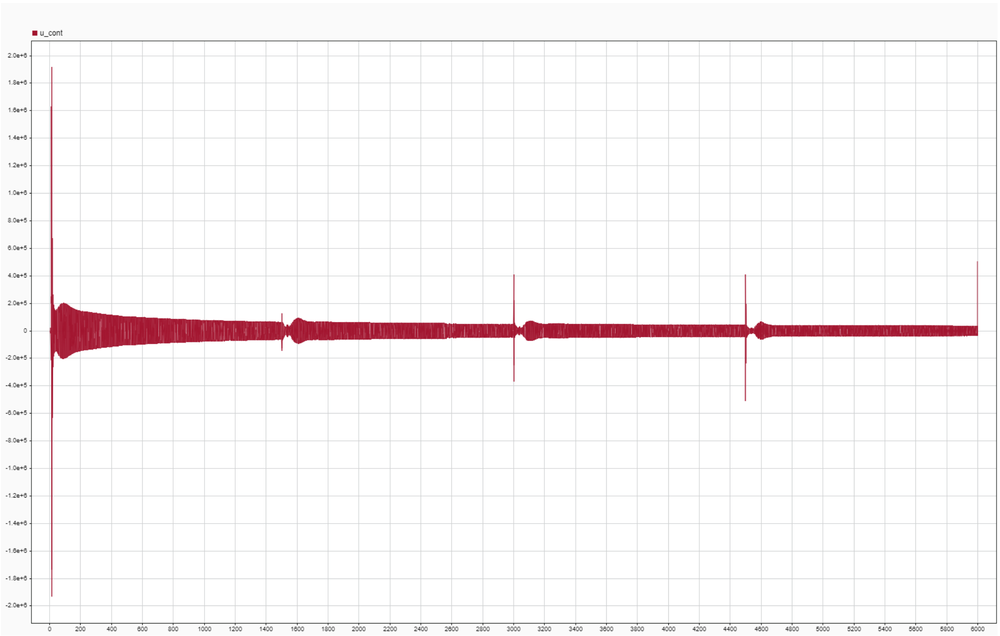


Figure 51: Control effort (Lyapunov MRAS, $\lambda = 60$)

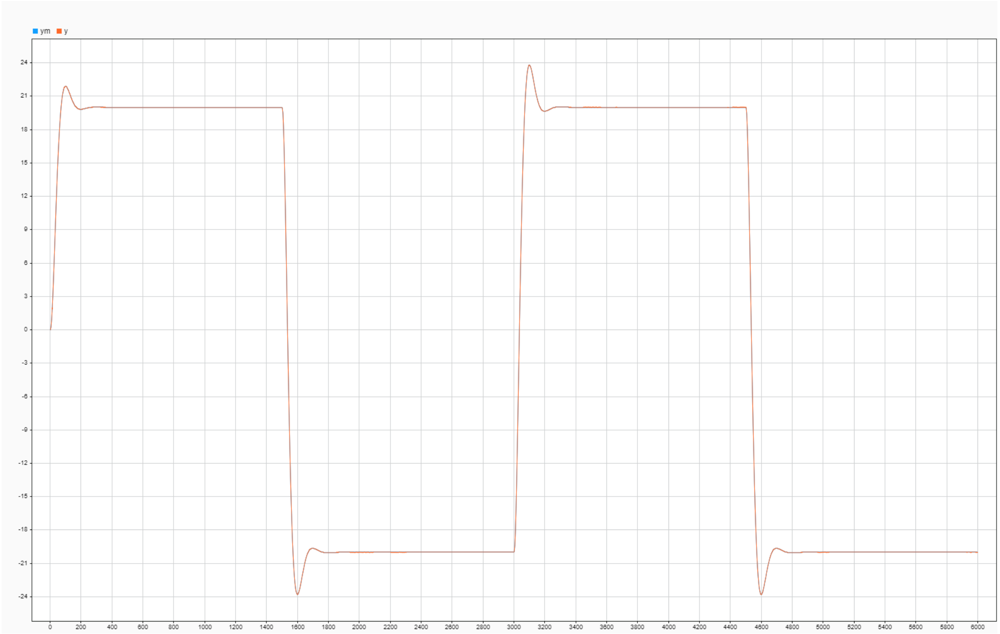


Figure 52: System's output Vs. Desired Output (Lyapunov MRAS, $\lambda = 60$)

9 Conclusion On The Results

When analyzing the three methods (MIT, normalized MIT, and Lyapunov MRAS), the effects of varying λ are observed as follows:

- **MIT Method**:
  - Increasing λ improves the simulation results.
  - The system parameter estimation remains relatively constant.
  - Both control effort and tracking overshoot show improvement.
- **Normalized MIT**:
  - Increasing λ worsens the simulation results across all metrics.
  - Both system estimation and control effort degrade.
  - Tracking the desired output becomes less accurate.
  - Another factor, α, when increased, improves all metrics.
- **Lyapunov MRAS**:
  - Increasing λ worsens the simulation results across all metrics.
  - Control effort multiplies approximately, which is undesirable.
  - System estimation degrades slightly.
  - Tracking performance remains excellent and better than the other two methods, regardless of λ.
- **Overall Observations**:
  - The type of input signal (square or sine wave) does not significantly affect the results.
  - The Lyapunov method is generally superior in all aspects except that the control effort is very noisy.
  - Although the value of the control effort is not high, the frequency of changes in the signal is too high, which many actuators may not handle well.

Conclusion: The Lyapunov method shows the best overall performance, despite the high frequency of changes in control effort which might be challenging for some actuators.