HOME-TAKEN EXAM 1B OF NONLINEAR ANALYSIS AND CONTROL IN MES-EMBEDDED CONTROL MODULE_

HAN University of Applied Sciences
School of Engineering and Automotive

Student 1: (AD Shinde, 2132290)

Lecturer: Amin Mannani

By submitting this exam the authors certify that this is their original work, and they have cited all the referenced materials, in the forms of texts, models, and books properly.

Note:
Do NOT Change the Format of This File.
Keep the Instructions; Start Your Part After them.

Contents

DOCUMENT HISTORY

Date	Change Summary	Authors
2024.03.17	Created the document	Amin Mannani

1 ASSIGNMENT 1

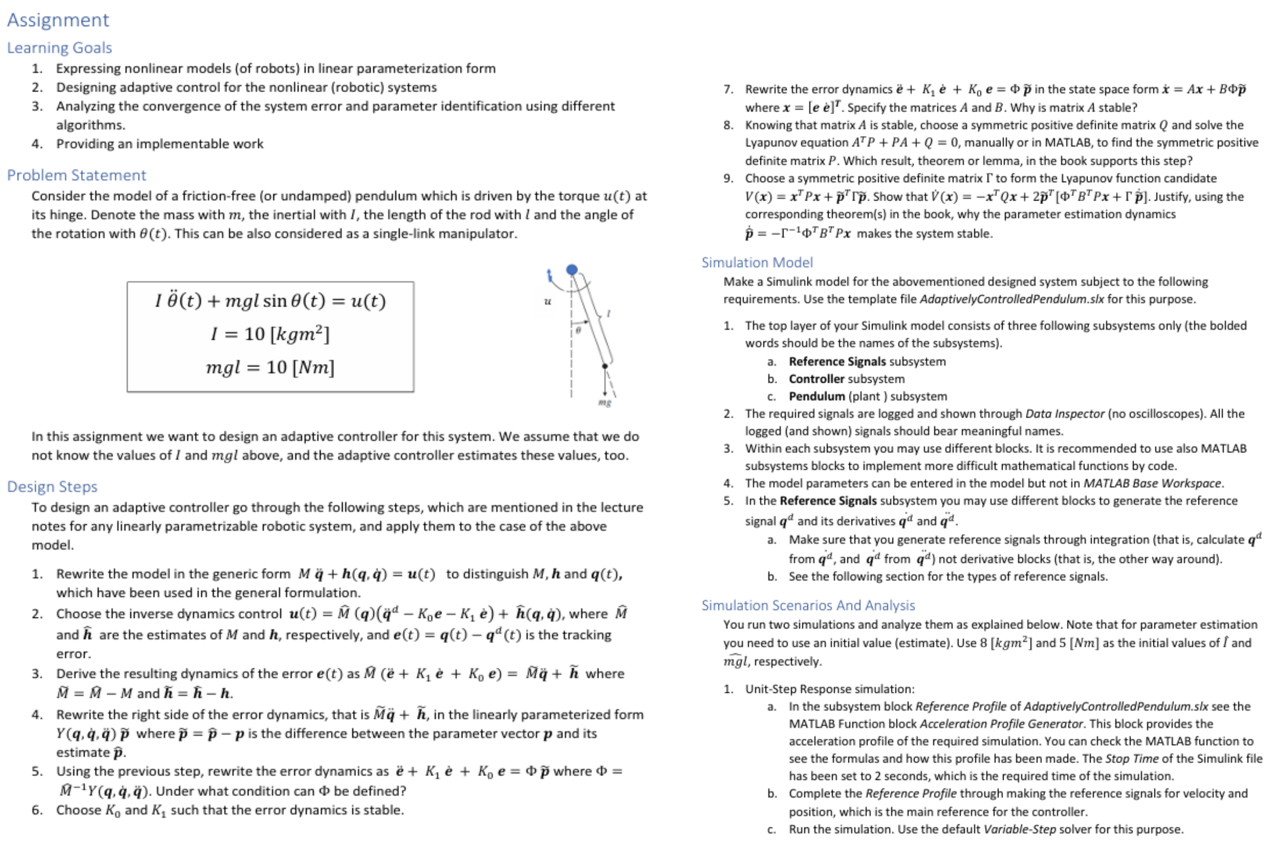
1.1 Problem statement

What to do	You rewite the assignment (or copy its picture) in this section. Furthermore 1. Explain what you learn from the assignment 2. Explain why this is important with reference to its applications.
Syllabus	The study material of this assignment as specified in the lesson.
Engineering indicators	The student shall identify what the problem is and why it is important.
Acceptance criteria	• Assignment is mentioned legibly, and its goal/importance is identified.
Size	Max 1 A4

The copies of the pictures of the assignment can be found on the next page.

1. This assignment helped me learn how to design an adaptive controller for an undamped pendulum model. And also helped me learn how to use Lyapunov's theory in a practical application.
2. This assignment is important as it helps to understand how to design and apply an adaptive controller for a practical application. As adaptive control algorithm can be implemented in real world applications such as the Autopilot system of an airplane where some parameters due to aerodynamics are unknown and can only be estimated. These kind of applications are very important in the real world out there, therefore, I believe trying to learn and use an adaptive control algorithm is very beneficial in the journey of learning control systems engineering.

HAN_UNIVERSITY OF APPLIED SCIENCES



Assignment

Learning Goals

1. Expressing nonlinear models (of robots) in linear parameterization form
2. Designing adaptive control for the nonlinear (robotic) systems
3. Analyzing the convergence of the system error and parameter identification using different algorithms.
4. Providing an implementable work

Problem Statement

Consider the model of a friction-free (or undamped) pendulum which is driven by the torque $u(t)$ at its hinge. Denote the mass with m, the inertial with I, the length of the rod with l and the angle of the rotation with $\theta(t)$. This can be also considered as a single-link manipulator.

$$\boxed{\begin{gathered} I\,\ddot{\theta}(t) + mgl\sin\theta(t) = u(t) \\ I = 10\;[kgm^2] \\ mgl = 10\;[Nm] \end{gathered}}$$

In this assignment we want to design an adaptive controller for this system. We assume that we do not know the values of I and mgl above, and the adaptive controller estimates these values, too.

Design Steps

To design an adaptive controller go through the following steps, which are mentioned in the lecture notes for any linearly parametrizable robotic system, and apply them to the case of the above model.

1. Rewrite the model in the generic form $M\,\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{u}(t)$ to distinguish $M, \boldsymbol{h}$ and $\boldsymbol{q}(t)$, which have been used in the general formulation.
2. Choose the inverse dynamics control $\boldsymbol{u}(t) = \widehat{M}\,(\boldsymbol{q})\big(\ddot{\boldsymbol{q}}^d - K_0\boldsymbol{e} - K_1\,\dot{\boldsymbol{e}}\big) + \widehat{\boldsymbol{h}}(\boldsymbol{q},\dot{\boldsymbol{q}})$, where $\widehat{M}$ and $\widehat{\boldsymbol{h}}$ are the estimates of M and $\boldsymbol{h}$, respectively, and $\boldsymbol{e}(t) = \boldsymbol{q}(t) - \boldsymbol{q}^d(t)$ is the tracking error.
3. Derive the resulting dynamics of the error $\boldsymbol{e}(t)$ as $\widehat{M}\,(\ddot{\boldsymbol{e}} + K_1\,\dot{\boldsymbol{e}} + K_0\,\boldsymbol{e}) = \widetilde{M}\ddot{\boldsymbol{q}} + \widetilde{\boldsymbol{h}}$ where $\widetilde{M} = \widehat{M} - M$ and $\widetilde{\boldsymbol{h}} = \widehat{\boldsymbol{h}} - \boldsymbol{h}$.
4. Rewrite the right side of the error dynamics, that is $\widetilde{M}\ddot{\boldsymbol{q}} + \widetilde{\boldsymbol{h}}$, in the linearly parameterized form $Y(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}})\,\widetilde{\boldsymbol{p}}$ where $\widetilde{\boldsymbol{p}} = \widehat{\boldsymbol{p}} - \boldsymbol{p}$ is the difference between the parameter vector $\boldsymbol{p}$ and its estimate $\widehat{\boldsymbol{p}}$.
5. Using the previous step, rewrite the error dynamics as $\ddot{\boldsymbol{e}} + K_1\,\dot{\boldsymbol{e}} + K_0\,\boldsymbol{e} = \Phi\,\widetilde{\boldsymbol{p}}$ where $\Phi = \widehat{M}^{-1}Y(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}})$. Under what condition can Φ be defined?
6. Choose K_0 and K_1 such that the error dynamics is stable.
7. Rewrite the error dynamics $\ddot{\boldsymbol{e}} + K_1\,\dot{\boldsymbol{e}} + K_0\,\boldsymbol{e} = \Phi\,\widetilde{\boldsymbol{p}}$ in the state space form $\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\Phi\widetilde{\boldsymbol{p}}$ where $\boldsymbol{x} = [\boldsymbol{e}\;\dot{\boldsymbol{e}}]^{\boldsymbol{T}}$. Specify the matrices A and B. Why is matrix A stable?
8. Knowing that matrix A is stable, choose a symmetric positive definite matrix Q and solve the Lyapunov equation $A^TP + PA + Q = 0$, manually or in MATLAB, to find the symmetric positive definite matrix P. Which result, theorem or lemma, in the book supports this step?
9. Choose a symmetric positive definite matrix Γ to form the Lyapunov function candidate $V(\boldsymbol{x}) = \boldsymbol{x}^TP\boldsymbol{x} + \widetilde{\boldsymbol{p}}^T\Gamma\widetilde{\boldsymbol{p}}$. Show that $\dot{V}(\boldsymbol{x}) = -\boldsymbol{x}^TQ\boldsymbol{x} + 2\widetilde{\boldsymbol{p}}^T[\Phi^TB^TP\boldsymbol{x} + \Gamma\,\dot{\widehat{\boldsymbol{p}}}]$. Justify, using the corresponding theorem(s) in the book, why the parameter estimation dynamics $\dot{\widehat{\boldsymbol{p}}} = -\Gamma^{-1}\Phi^TB^TP\boldsymbol{x}$ makes the system stable.

Simulation Model

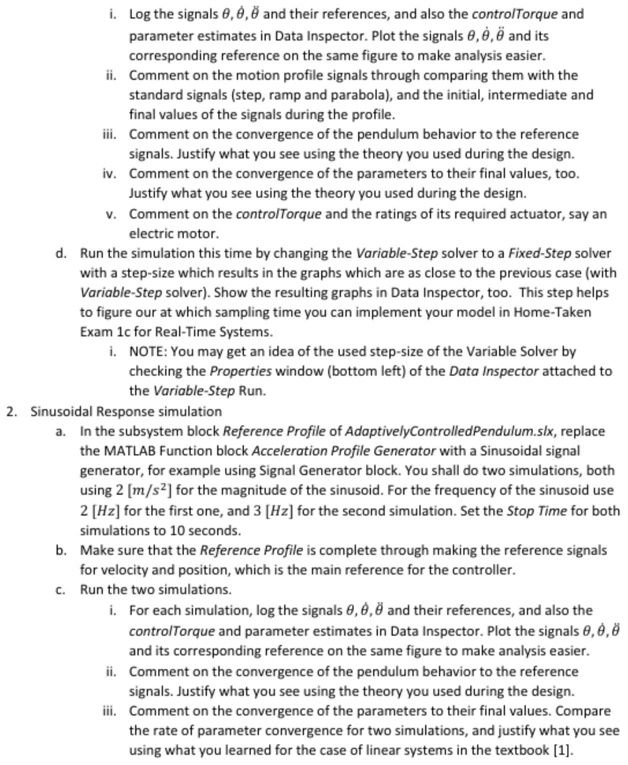
Make a Simulink model for the abovementioned designed system subject to the following requirements. Use the template file *AdaptivelyControlledPendulum.slx* for this purpose.

1. The top layer of your Simulink model consists of three following subsystems only (the bolded words should be the names of the subsystems).
   a. **Reference Signals** subsystem
   b. **Controller** subsystem
   c. **Pendulum** (plant) subsystem
2. The required signals are logged and shown through *Data Inspector* (no oscilloscopes). All the logged (and shown) signals should bear meaningful names.
3. Within each subsystem you may use different blocks. It is recommended to use also MATLAB subsystems blocks to implement more difficult mathematical functions by code.
4. The model parameters can be entered in the model but not in *MATLAB Base Workspace*.
5. In the **Reference Signals** subsystem you may use different blocks to generate the reference signal $\boldsymbol{q}^d$ and its derivatives $\dot{\boldsymbol{q}}^d$ and $\ddot{\boldsymbol{q}}^d$.
   a. Make sure that you generate reference signals through integration (that is, calculate $\boldsymbol{q}^d$ from $\dot{\boldsymbol{q}}^d$, and $\dot{\boldsymbol{q}}^d$ from $\ddot{\boldsymbol{q}}^d$) not derivative blocks (that is, the other way around).
   b. See the following section for the types of reference signals.

Simulation Scenarios And Analysis

You run two simulations and analyze them as explained below. Note that for parameter estimation you need to use an initial value (estimate). Use 8 $[kgm^2]$ and 5 $[Nm]$ as the initial values of $\hat{I}$ and $\widehat{mgl}$, respectively.

1. Unit-Step Response simulation:
   a. In the subsystem block *Reference Profile* of *AdaptivelyControlledPendulum.slx* see the MATLAB Function block *Acceleration Profile Generator*. This block provides the acceleration profile of the required simulation. You can check the MATLAB function to see the formulas and how this profile has been made. The *Stop Time* of the Simulink file has been set to 2 seconds, which is the required time of the simulation.
   b. Complete the *Reference Profile* through making the reference signals for velocity and position, which is the main reference for the controller.
   c. Run the simulation. Use the default *Variable-Step* solver for this purpose.



i. Log the signals $\theta, \dot{\theta}, \ddot{\theta}$ and their references, and also the *controlTorque* and parameter estimates in Data Inspector. Plot the signals $\theta, \dot{\theta}, \ddot{\theta}$ and its corresponding reference on the same figure to make analysis easier.

ii. Comment on the motion profile signals through comparing them with the standard signals (step, ramp and parabola), and the initial, intermediate and final values of the signals during the profile.

iii. Comment on the convergence of the pendulum behavior to the reference signals. Justify what you see using the theory you used during the design.

iv. Comment on the convergence of the parameters to their final values, too. Justify what you see using the theory you used during the design.

v. Comment on the *controlTorque* and the ratings of its required actuator, say an electric motor.

d. Run the simulation this time by changing the *Variable-Step* solver to a *Fixed-Step* solver with a step-size which results in the graphs which are as close to the previous case (with *Variable-Step* solver). Show the resulting graphs in Data Inspector, too. This step helps to figure our at which sampling time you can implement your model in Home-Taken Exam 1c for Real-Time Systems.

i. NOTE: You may get an idea of the used step-size of the Variable Solver by checking the *Properties* window (bottom left) of the *Data Inspector* attached to the *Variable-Step* Run.

2. Sinusoidal Response simulation
   a. In the subsystem block *Reference Profile* of *AdaptivelyControlledPendulum.slx*, replace the MATLAB Function block *Acceleration Profile Generator* with a Sinusoidal signal generator, for example using Signal Generator block. You shall do two simulations, both using 2 $[m/s^2]$ for the magnitude of the sinusoid. For the frequency of the sinusoid use 2 $[Hz]$ for the first one, and 3 $[Hz]$ for the second simulation. Set the *Stop Time* for both simulations to 10 seconds.
   b. Make sure that the *Reference Profile* is complete through making the reference signals for velocity and position, which is the main reference for the controller.
   c. Run the two simulations.
      i. For each simulation, log the signals $\theta, \dot{\theta}, \ddot{\theta}$ and their references, and also the *controlTorque* and parameter estimates in Data Inspector. Plot the signals $\theta, \dot{\theta}, \ddot{\theta}$ and its corresponding reference on the same figure to make analysis easier.
      ii. Comment on the convergence of the pendulum behavior to the reference signals. Justify what you see using the theory you used during the design.
      iii. Comment on the convergence of the parameters to their final values. Compare the rate of parameter convergence for two simulations, and justify what you see using what you learned for the case of linear systems in the textbook [1].

1.2 Solution

What to do	Here you solve the assignment; you can type directly or attach pictures of your handwritten notes. 1. Write the formulas using the correct symbols legibly. 2. Respect the logical/mathematical way of writing and citation. 3. If relevant, refer to the names of the models and functions which you have in the corresponding MATLAB Project. 4. If requested in the assignment, provide the graphs.
Syllabus	The study material of this assignment as specified in the lesson.
Engineering indicators	The student should be able to show both the solution and the process of arrivinging at it.
Acceptance criteria	• Solution approach is clarified and sound. • The text, formulas and figures are legible. • The formulas, final results and simulation/analysis graphs are compared and justified against each other.
Size	Max 15 A4

1.2.1 Design Steps

1) To rewrite the given model in the generic form

$$M\ddot{q} + h(q,\dot{q}) = u(t)$$

We have,

$M\ =\ I$

$h(q,\dot{q}) = mgl$

$q(t) = \theta(t)$

$u(t)\ is\ the\ control\ law$

Therefore, the equation becomes

$$I\ddot{q} + mglsin\big(q(t)\big) = u(t)$$

2) The given inverse dynamic control law:

$$u(t) = \hat{M}(q)\left(\ddot{q}^d - k_0 e - k_1 \dot{e} + \hat{h}(q,\dot{q})\right)$$

Where,

- q(t) is the current state of the angle $\theta(t)$
- $q^d(t)$ is the desired state trajectory
- $e(t) = q(t) - q^d(t)$ is the tracking error
- $\dot{e}$ is the derivative of the tracking error
- $\hat{M}(q)\ and\ \hat{h}(q,\dot{q})$ are the estimates of the systems inertia and the combined effects of gravitational and other places.
- $k_0 and k_1$ are positive definite gain matrices designed to ensure the system stability and desired dynamic performance.

3) Step by step derivation:

- Substituting the control law into the system dynamics

$$I\ddot{q} + mglsin(q) = \widehat{M}(q)(\ddot{q_d} - K_0 e - K_1 \dot{e}) + \hat{h}(q, \dot{q})$$

- Rearranging to highlight error dynamics

$$M\ddot{q} + h(q, \dot{q}) = \widehat{M}(\ddot{q_d} - K_0 e - K_1 \dot{e}) + \hat{h}(q, \dot{q})$$

Where, M = I and $h(q, \dot{q}) = mglsin(q)$

- Using the parameter estimation errors

$$\widehat{M}(\ddot{q} - \ddot{q_d} + K_1 \dot{e} + K_0 e) + \tilde{h} = \widetilde{M}\ddot{q} + \tilde{h}$$

Where, $\widetilde{M} = \widehat{M} - M \; and \; \tilde{h} = \hat{h} - h$

- Now, expressing in terms of error dynamics

Since, $\ddot{e} = \ddot{q} - \ddot{q_d}$

$$\widehat{M}(\ddot{e} + K_1 \dot{e} + K_0 e) = \widetilde{M}\ddot{q} + \tilde{h}$$

i.e. $\hat{I}(\ddot{e} + K_1 \dot{e} + K_0 e) = \tilde{I}\ddot{\theta} + \widetilde{mgl}$

4) We assume $\widehat{M}(q) and \hat{h}(q, \dot{q})$ have the same functional form as M and respectively, that is using the estimated parameters vector, therefore we have:

$$\widetilde{M}\ddot{q} + \tilde{h} = Y(q, \dot{q}, \ddot{q})\tilde{p}$$

Where, $Y(q, \dot{q}, \ddot{q}) = Y\left(\theta, \dot{\theta}, \ddot{\theta}\right) = \left[\ddot{\theta} \; sin(\theta)\right]$ and $\tilde{p} = \hat{p} - p = \begin{bmatrix} \tilde{I} \\ \widetilde{mgl} \end{bmatrix}$

Therefore, the equation becomes,

$$\hat{I}(\ddot{e} + K_1 \dot{e} + K_0 e) \; = \; \left[\ddot{\theta} \; sin(\theta)\right] \; \begin{bmatrix} \tilde{I} \\ \widetilde{mgl} \end{bmatrix}$$

5)

From the previous step we have,

$$\hat{I}(\ddot{e} + K_1 \dot{e} + K_0 e) \; = \; \left[\ddot{\theta} \; sin(\theta)\right] \; \begin{bmatrix} \tilde{I} \\ \widetilde{mgl} \end{bmatrix}$$

Multiplying both ides by $\widehat{M}^{-1} \equiv \hat{I}^{-1}$,

$$\hat{I}(\ddot{e} + K_1 \dot{e} + K_0 e) \, \hat{I}^{-1} \; = \; \hat{I}^{-1}\left[\ddot{\theta} \; sin(\theta)\right] \; \begin{bmatrix} \tilde{I} \\ \widetilde{mgl} \end{bmatrix}$$

i.e. $(\ddot{e} + K_1 \dot{e} + K_0 e) \; = \; \Phi \begin{bmatrix} \tilde{I} \\ \widetilde{mgl} \end{bmatrix}$,

where, $\Phi = \; \hat{I}^{-1}\left[\ddot{\theta} \; sin(\theta)\right]$

Condition for Φ to be defined:

The crucial condition that for Φ to be defined is that $\widehat{M}$, the estimate of the inertia matrix must be non-singular. This means it should not be zero and in the context of matrices it should have full rank.

In the adaptive control setting it is crucial to ensure that the initial estimates do not lead to a singular $\widehat{M}$ at any point in time.

6)

Selecting K_0 and K_1:

Choosing $K_0 \; = 40 \; and \; K_1 = 12$ such that $\ddot{e} + K_1\dot{e} + K_0 e$ is stable.

From the characteristic equation $s^2 + K_1 s + K_0 = 0$ with the chosen values of K_0 and K_1,

we have poles: $-6 \pm 2i$

Both poles have a real part of -6 which is negative, indicating that the error dynamics is stable and also make it quite robust by poles being farther towards the left.

7)

Define the state vector X as

$$X = \begin{bmatrix} e \\ \dot{e} \end{bmatrix}$$

And

$$\dot{X} = \begin{bmatrix} \dot{e} \\ \ddot{e} \end{bmatrix}$$

From the original second-order equation, we can express $\ddot{e}$ as:

$$\ddot{e} = -K_1\dot{e} - K_0 e + \Phi\tilde{p}$$

Thus, the derivative of the state vector $\dot{X}$ becomes:

$$\dot{X} = \begin{bmatrix} \dot{e} \\ -K_0 e - K_1\dot{e} + \Phi\tilde{p} \end{bmatrix}$$

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -K_0 & -K_1 \end{bmatrix}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}\Phi\tilde{p}$$

Where,

$$A = \begin{bmatrix} 0 & 1 \\ -K_0 & -K_1 \end{bmatrix}$$

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

Why is matrix A stable?

Matrix A is stable under the condition that all its eigenvalues have negative real parts.

Given the structure of A and the characteristic equation derived from it: $\mathrm{s}^2 + K_1\mathrm{s} + K_0 = 0$, with both poles having a negative real part, this makes the matrix A stable.

8) The system matrix A is given by:

$$A = \begin{bmatrix} 0 & 1 \\ -K_0 & -K_1 \end{bmatrix}$$

And we choose Q, which should be symmetric positive definite, as an identity matrix, as such matrices are guaranteed to be symmetric positive definite:

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

And solve the Lyapunov equation:

$$A^T P + PA + Q = 0$$

Now, from using the 'lyap' command in MATLAB, the Lyapunov equation can be solved:

which, gives the solution matrix P as: $P = \begin{bmatrix} 1.8583 & 0.0125 \\ 0.0125 & 0.0427 \end{bmatrix}$

(the code for this can be found in the 'MATLAB Scripts' folder in the assignment submission)

9) Choosing a symmetric positive definite matrix $\Gamma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Defining the Lyapunov function candidate:

Lets consider the Lyapunov function candidate,

$$V(x) = x^T Px + \tilde{p}^T \Gamma \tilde{p}$$

Here, P and Γ are symmetric positive definite matrices. P solves the Lyapunov equation $A^T P + PA + Q = 0$

Deriving the time derivative of the Lyapunov Function:

$$\dot{V}(x) = \dot{x}^T Px + x^T P\dot{x} + \dot{\tilde{p}}^T \Gamma \tilde{p} + \tilde{p}^T \Gamma \dot{\tilde{p}}$$

$$= (x^T A^T + \tilde{p}^T \Phi^T B^T)Px + x^T P(Ax + B\Phi\tilde{p}) + \dot{\tilde{p}}^T \Gamma \tilde{p} + \tilde{p}^T \Gamma \dot{\tilde{p}}$$

$$= x^T(A^T P + PA)x + \tilde{p}^T \Phi^T B^T Px + x^T PB\Phi\tilde{p} + \dot{\tilde{p}}^T \Gamma \tilde{p} + \tilde{p}^T \Gamma \dot{\tilde{p}}$$

Knowing $A^T P + PA = -Q$, we have:

$$= -x\text{\^{}}T\ Qx + [\tilde{p}^T(\Phi^T B^T Px + \Gamma\dot{\tilde{p}}) + x^T PB\Phi\tilde{p} + \dot{\tilde{p}}^T \Gamma \tilde{p}]$$

the two terms in the square brackets are equal to each other because they are scalar and symmetric to each other therefore,

$$= -x\text{\^{}}T\ Qx + 2\tilde{p}^T(\Phi^T B^T Px + \Gamma\dot{\tilde{p}})$$

The above equation suggests that if we choose $\dot{\tilde{p}}$ so that the term in the parenthesis becomes 0, then $\dot{V}$will be negative, that is:

$$\Phi^T B^T Px + \Gamma\dot{\tilde{p}} = 0 \rightarrow \Gamma\dot{\tilde{p}} = -\Phi^T B^T Px \rightarrow \begin{cases} \dot{\tilde{p}} = -\Gamma^{-1}\, \Phi^T B^T Px \\ \dot{\tilde{p}} = -\Gamma^{-1}\, \Phi^T B^T Px \end{cases}$$

(note: Γ is symmetric positive definite therefore $\Gamma^{-1}\ exists.$)

which results in, $\dot{V}(x) = -x^T Qx \leq 0$

which proves that the system remains stable in the Lyapunov sense i.e. x and $\tilde{p}$ remain bounded

Justification using theorem:

- To show that $\lim_{t \to 0} x = 0$
  - Φ is bounded : $\Phi = M^{-1}(q)Y(q, \dot{q}, \ddot{q})$

    We keep M^{-1} bounded by bpunding it with a 'reasonable region' around M^{-1}. Since M^{-1} is defined so should be $\widehat{M}^{-1}$.

    Then since Φ depends on $\ddot{q}$, we need to guarantee that $\ddot{q}$ remains bounded.

    x and $\tilde{p}$: bounded → $e, \dot{e}$: bounded,

$M(q)\ddot{q} + h(q,\dot{q}) = \widehat{M}(q)(\ddot{q}^d - k_0 e - k_1 \dot{e}) + \hat{h}(q,\dot{q})$

$\ddot{q}^d$: bounded

Therefore, $\ddot{q}$: bounded

- Observe that

$$\left.\begin{array}{c} \dot{x} = Ax + B\Phi\tilde{p} \\ x\colon bounded \\ \Phi:\ bounded \\ \tilde{p}\colon bounded \end{array}\right\} \rightarrow \dot{x}\colon bounded \rightarrow x\colon uniformly\ continous \rightarrow \lim_{t\to\infty} x \ = \ 0$$

- Also, by the Lyapunov stability theorem, if V(x) is positive definite and $\dot{V}(x)$ is negative definite, the system is globally asymptotically stable.

  Here:

  - $V(x) = x^T P x + \tilde{p}^T \Gamma \tilde{p}$ is positive definite since P and Γ are symmetric positive definite matrices.
  - $\dot{V}(x) = -x^T Q x$ is negative definite since Q is positive definite.

  Therefore, the Lyapunov function V(x) is decreasing, and the system is stable. The parameter estimation dynamics $\dot{\hat{p}} = -\Gamma^{-1}\,\Phi^T B^T P x$ ensure the convergence of the parameters by driving $\tilde{p}$ to zero. This guarantees that the estimated parameters $\hat{p}$ converge to the true parameters p, stabilizing the system.

1.2.2 Simulation Model

Below you can see the required parts from the Simulink model:

Top layer:

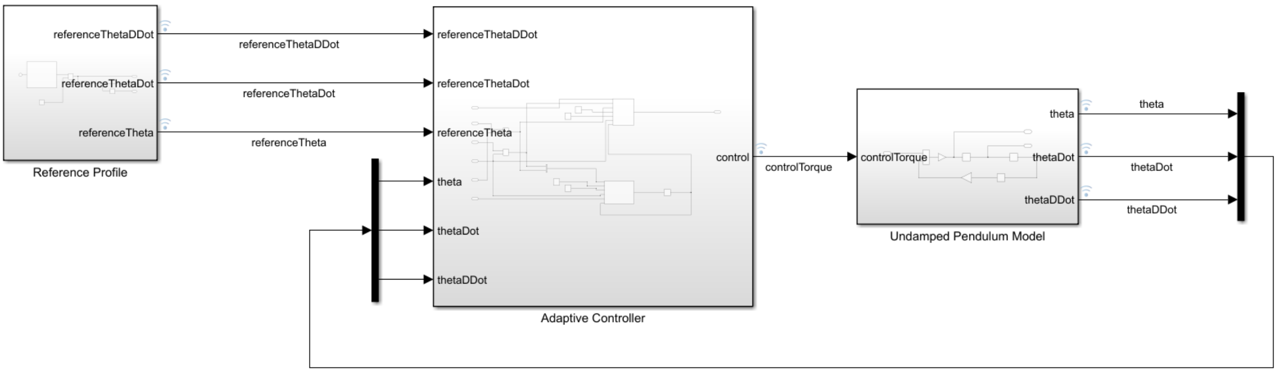


Figure 1 Top layer Simulink model

Reference Signals subsystem:

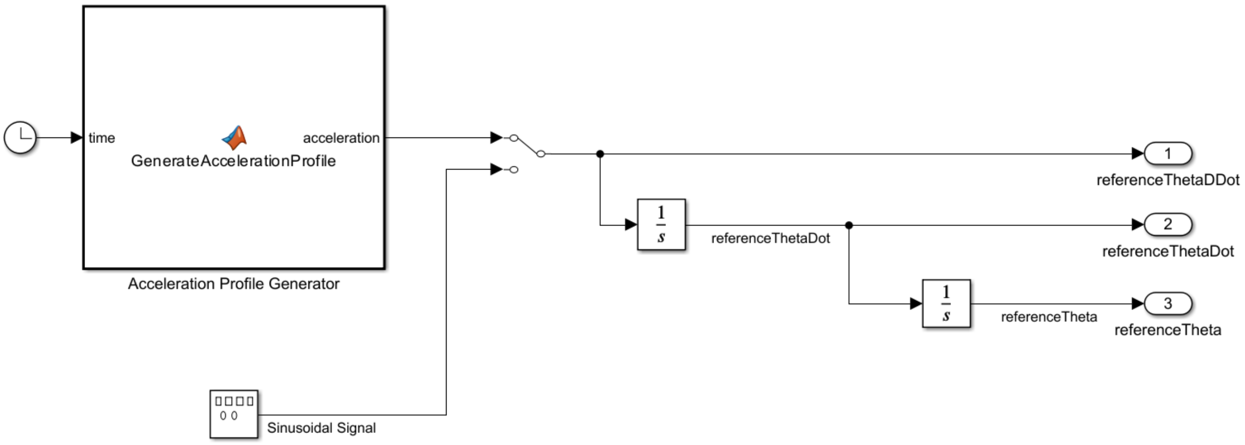


Figure 2 Reference Signals subsystem

Controller subsystem:

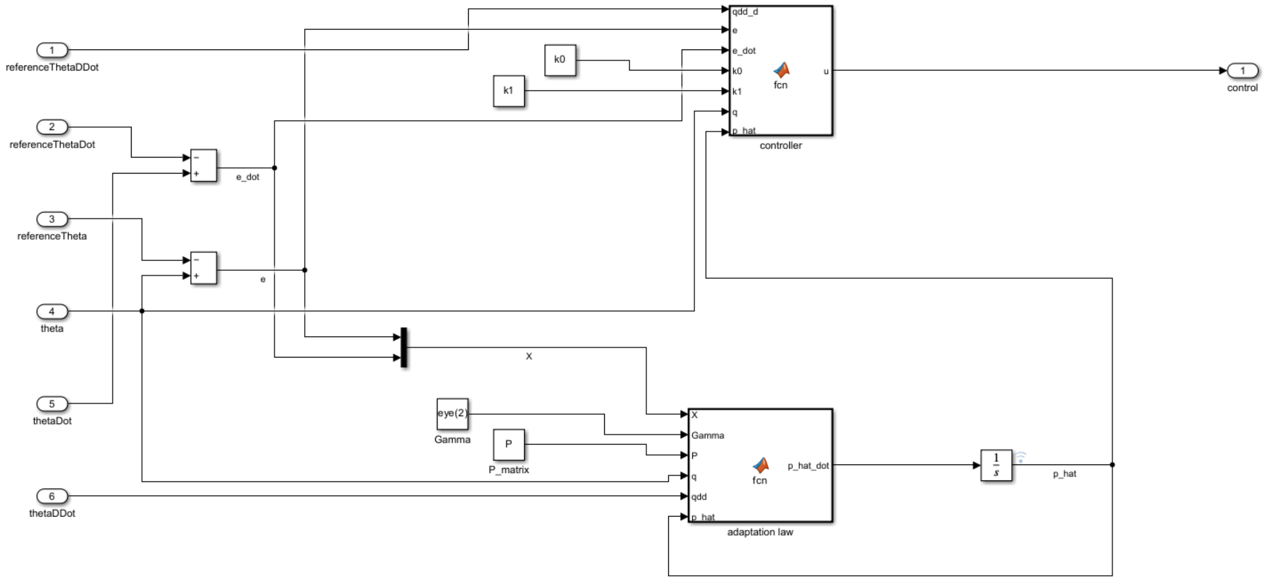


Figure 3 Controller subsystem

Pendulum (plant) subsystem:

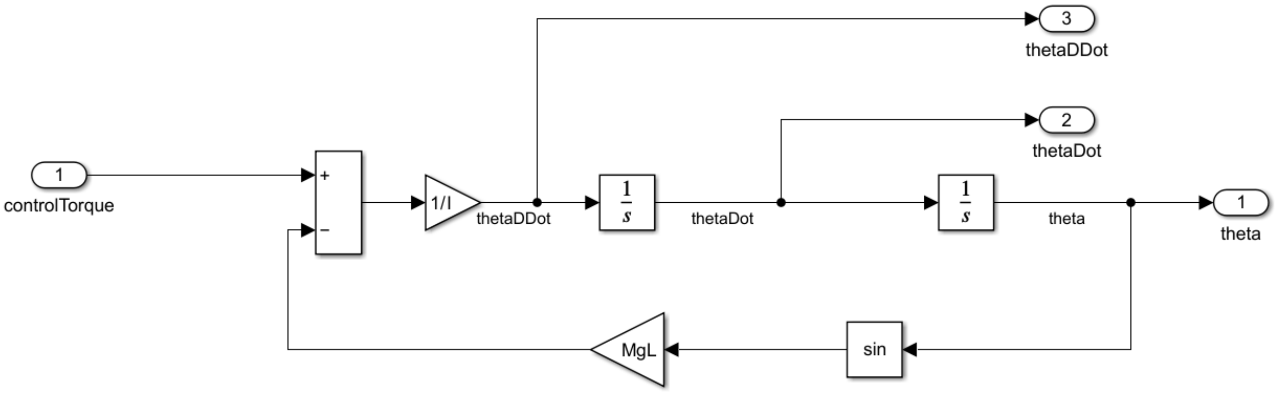


Figure 4 Plant subsystem

All the required signals are logged and shown through the data inspector in the next part of the report.

1.2.3 Simulation Scenarios and Analysis

Two simulations were run, analyzed and explained in this section below. As required the given initial values for parameter estimation were used, 8 [kgm^2] and 5 [Nm] as the initial values of $\hat{I}\ and\ \widehat{mgl}$ respectively.

1. Unit step response simulation:

   b. The reference profile was completed by making the reference signals for velocity and position.

   c. The simulation was run using the default Variable-Step solver.

   The required signals are logged and shown using the data inspector below:

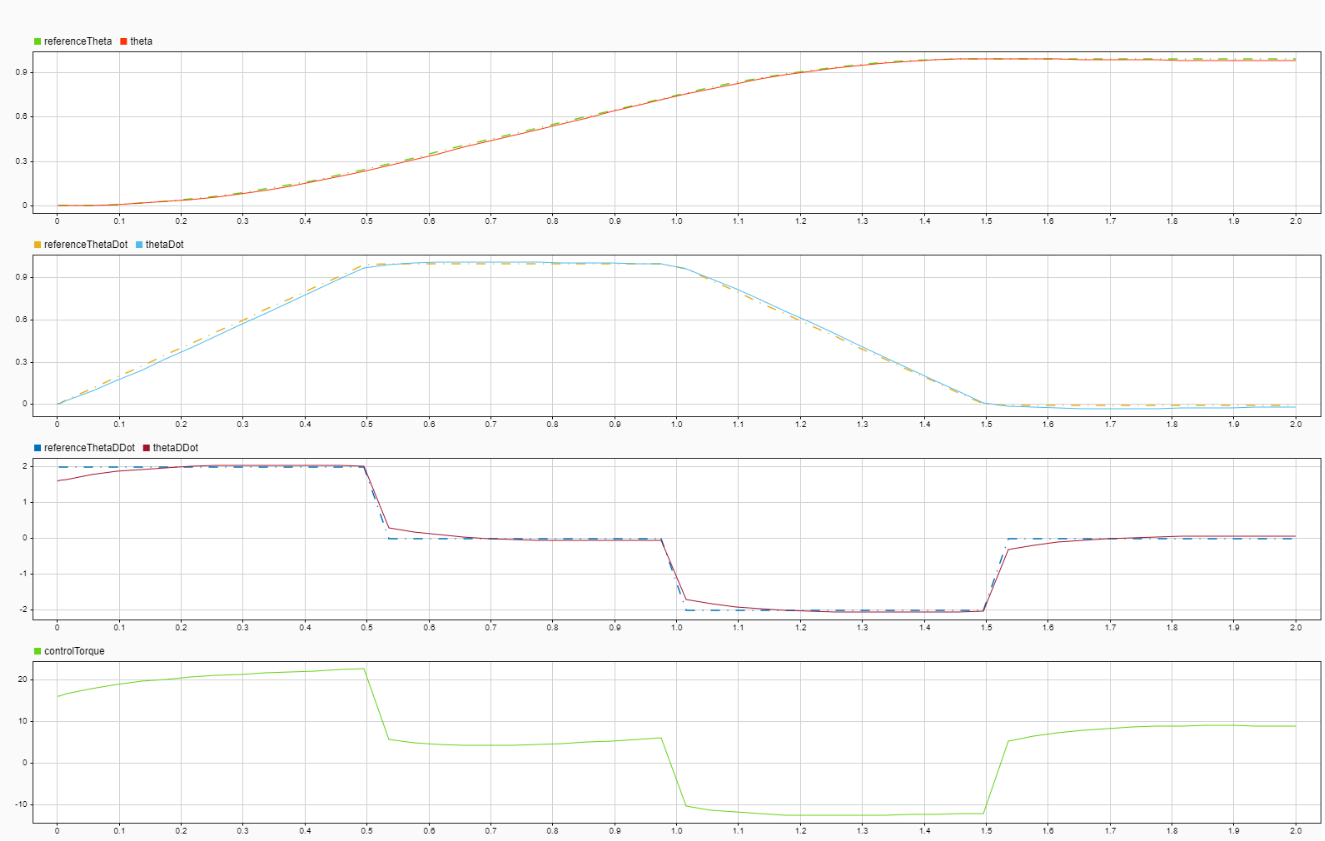


Figure 5 Reference and Plant Signals (Variable-Step Solver)

- Above you can see the reference and plant angle, velocity and acceleration plots. The dashed lines are from the reference generator. From the above plots it can be observed that reference theta follows a ramp pattern, gradually increasing over time and eventually stabilizing. Reference theta_dot also looks like a ramp response, increasing, stabilizing in the middle phase and eventually decreasing. The reference theta_ddot is like a step input. The initial values of reference velocity and displacement start from zero as they should with the type of signal they resemble, but the reference accelaration being a step input initial value starts at 2 m/s^2. Their final values go back to zero except for the angle of the pendulum.
- It can be observed that the convergence of the pendulum behaviour signals to the reference signals is very accurate, this indicates the effectiveness of the controller as the theory behind adaptive control suggests that even with unknown parameters, the system should converge to the desired performance through parameter estimation and control adjustments, which can be observed as working in the above plot.

- The control torque starts with a high initial value and has frequent changes and the signal is very similar to the acceleration signal.
- Based on the peak values and variation in the control torque the requirements for the electric motor (actuator) can be determined. This could be around 22-25 kgm^2.
- The parameters converge to their final values in a stable manner. Stability of the matrix A is achieved through the appropriate selection of control gains k0 and k1. The Lyapunov equation which is used to find a symmetric positive definite matrix P, which characterises the stability of the error dynamics, ensures system stability. This stability guarantees that the errors in parameter estimation decrease over time, leading to convergence to their final values.
- Relation between the references: The reference signals are inherently related through differentiation and integration. This means that any changes in the reference theta will directly affect the reference velocity and acceleration. The control system accounts for all these relationships simultaneously. For example, a sudden change in acceleration requires the controller to adjust the torque output rapidly to meet this new acceleration demand. The control torque needed at any given time is a direct response to the requirements set by the reference signals. If the reference acceleration requires a sudden increase, the torque must increase accordingly to achieve this. Similarly, maintaining a steady velocity as dictated by the reference might require varying levels of torque depending on external factors acting on the pendulum. And this behaviour can be clearly observed in the sognal plots above as well.

d. Running the simulation with Fixed-step Solver:

With the Fixed-Step size of 0.01 almost similar response was generated, as you can see in the plots below:

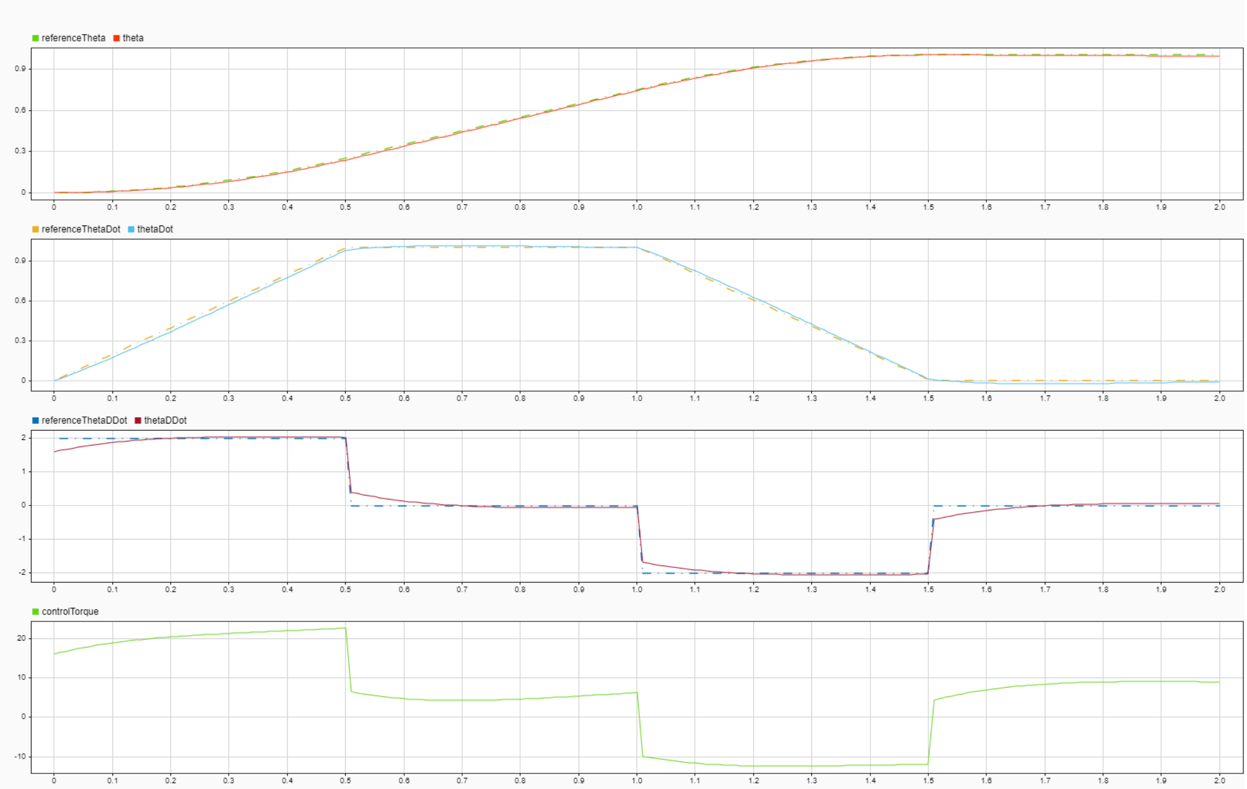


Figure 6 Reference and Plant Signals (Fixed Step Solver)

2) Sinusoidal Response Simulation:

First scenario:

For the first scenario the magnitude of the sinusoid was considered to be 2 m/s^2 and the frequency 2 Hz.

The below plots were generated:

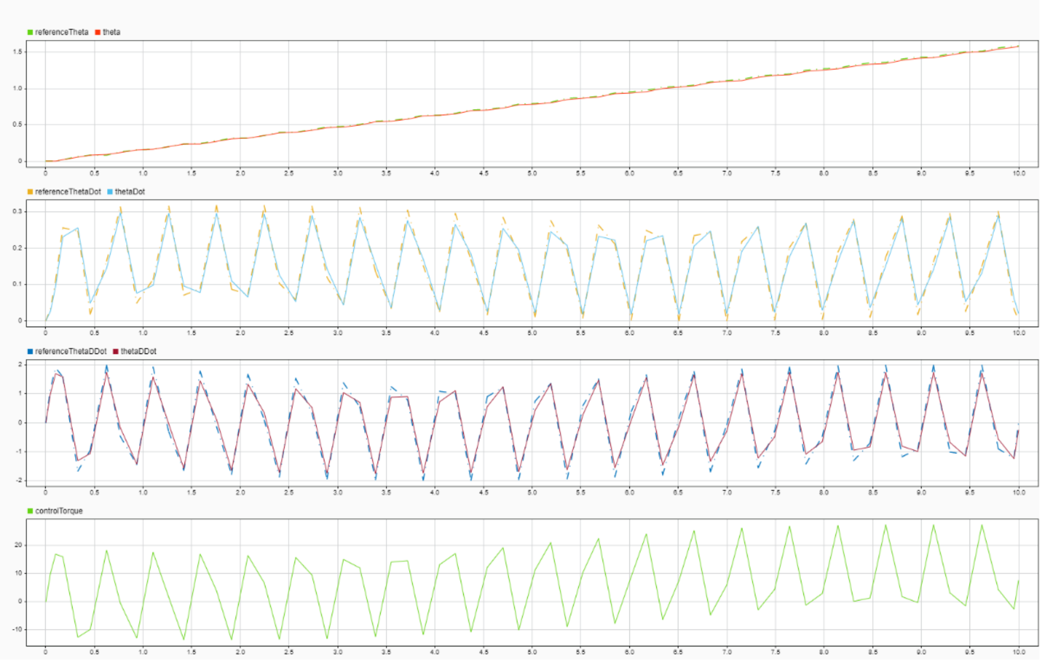


Figure 7 Reference inputs and Plant behaviour (Sinusoidal Input 1)

Second scenario:

For the second scenario the magnitude of the sinusoid was considered to be 2 m/s^2 and the frequency 3 Hz.

The below plots were generated:

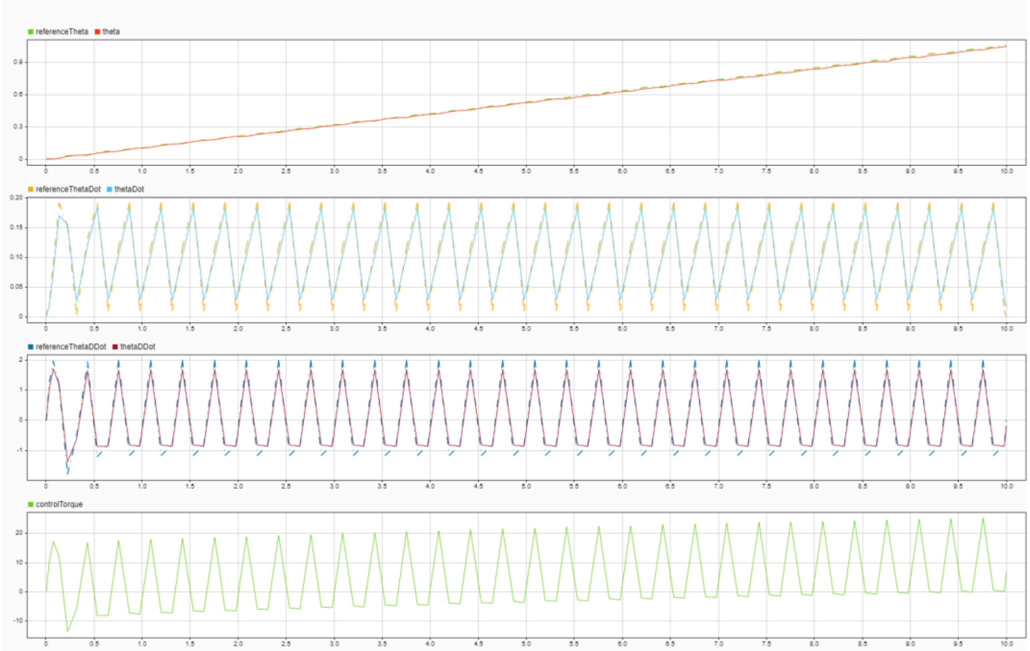


Figure 8 Reference inputs and Plant behaviour (Sinusoidal Input 2)

Analysis of results of sinusoidal response simulations:

- **Theta response for**

  1st scenario:Demonstrates close tracking with a smooth response to the sinusoidal input, maintaining a consistent phase and amplitude with the reference.

  2nd scenario: Maintains a similarly excellent tracking performance.

- **ThetaDot response for**

  1st scenario: Exhibits excellent matching with the reference, indicating the systems ability to respond correctly to velocity changes.

  2nd scenario: Similar to the 2 Hz scenario but with reduced phase lag, potentially indicating systems proactive adjustment to match the increased frequency.

- **ThetaDDot response for**

  1st scenario: Shows accurate tracking of sinusoidal acceleration profiles but with slight tracking erros around each peak.

  2nd scenario: Similar repose with marginal increase in tracking errors at each peak, due to the significant control effort required for the fast sinusoidal inputs.

- **Control torque**

  1st scenario: Displays sinusoidal variation with amplitude and frequency that effectively support the desired motion, indicating appropriate control action to meet the dynamics.

  2nd scenario: Reveals a higher amplitude and more frequent adjustments, reflecting the need for more significant control efforts to counteract the faster sinusoidal inputs.

- Both simulations show that the system is capable of closely following the given reference signals across all signals. However, the higher frequency 3 Hz demands more from the control system, leading to the slightly increased efforts visible in the control torque and marginal increase in tracking errors.
- Overall there is effective parameter convergence but there is faster convergence for the 1st scenario i.e. at lower frequency due to more extended periods between peaks allowing more time for adjustment compared to the 2nd scenario with a higher frequency. The estimation gain in adaptive control plays an important role for convergence speed, stability and robustness of the parameter estimation process. Increasing estimation gain can lead to faster parameter convergence. A fast convergence rate ensures that the control system quickly adapts to changes in the simulated system dynamics and accurately tracks the reference signals.
- It is useful to point out that the system defined by $\dot{\hat{p}}$ not only guarantees the boundedness of $\dot{\hat{p}}$ and B but also that of the whole state x.
- In conclusion, the system remains stable and effective across both frequencies which is a sign of robust control design.The ability of the adaptive controller to handle different frequencies without significant degradation in performance is indicative of well tuned adaptive mechanism within the controller, capable of dealing with the complexieties introduced by higher rates of change in the system.

REFERENCES

[1] The Mathworks, Inc, "Control Algorithm Modeling Guidelines Using MATLAB, Simulink, and Stateflow - Version 5.0," March 2020. [Online]. Available: nl.mathworks.com/help/simulink/mab-modeling-guidelines.html. [Accessed 01 2021].