Design Project: Phase Two

Jarod Tang

November 2, 2026

Contents

1 Introduction **2**

2 Preliminaries **2**
2.1 State-Space Model . . . 2
2.2 Performance Requirements . . . 3

3 Project Tasks **3**
3.1 Task A: Controllability and Observability Analysis . . . 3
3.2 Task B: System Analysis without Motor Dynamics . . . 4
3.3 Task C: State-Feedback Control . . . 5
3.3.1 Pole Placement . . . 6
3.3.2 C1: Continuous-Time Controller Design . . . 6
3.3.3 C2: Discrete-Time Controller Design . . . 7
3.3.4 Controller Performance . . . 7
3.4 Task D: Observer-Based Feedback Control . . . 7
3.4.1 Sensor Selection . . . 8
3.4.2 D1: Continuous-Time Observer Design . . . 9
3.4.3 D2: Discrete-Time Observer Design . . . 10
3.4.4 Controller Performance . . . 11
3.5 Task E: Sinusoidal Reference Tracking . . . 11
3.5.1 Performance Requirements . . . 11
3.5.2 Controller Design for Sinusoidal Reference . . . 12
3.5.3 Controller Performance Comparison . . . 13

4 Discussion **13**

5 Conclusion **13**

1 Introduction

The primary goal of Phase Two of the Design Project is to take a rigorous design approach to applying and contrasting the state-space controller and observer design techniques. Firstly, the characteristics of different motor and sensor combinations are to be analysed. After a reduced-order model of the system is constructed, arbitrary pole placement is used to design continuous and discrete full-state feedback controllers.

2 Preliminaries

2.1 State-Space Model

From Phase One, the state space is depicted by Equation 1, where θ_m, θ_l, $\dot{\theta}_m$, $\dot{\theta}_m$ and I_a is the angular position of the motor and load with respect to the fixed base, angular speed of the motor and load and the armature current respectively.

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \end{bmatrix} = \begin{bmatrix} \theta_m & \theta_l & \dot{\theta}_m & \dot{\theta}_l & I_a \end{bmatrix} \tag{1}$$

The plant and motor model provided by the Design Project specifications is as follows and depicted by Equations 2 and 3 respectively. Note that K_s is the spring constant and the output torque (T_m) is given by Equation 4.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{K_s}{J_m} & \frac{K_s}{J_m} & 0 & 0 \\ \frac{K_s}{J_l} & -\frac{K_s}{J_l} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{-b \cdot \text{sgn}(x_3) x_3^2}{J_m} \\ \frac{mgl \sin(x_2)}{J_l} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{J_m} \\ 0 \end{bmatrix} T_m \tag{2}$$

$$\dot{x}_5 = -\frac{R_a}{L_a} x_5 - \frac{K_m}{L_a} x_3 + \frac{1}{L_a} u \tag{3}$$

$$T_m = K_m I_a = K_m x_5 \tag{4}$$

In Phase One of the Design Project, Motor 2 and the vision sensor *Allied Vision Alvium 1800 U-050* were chosen as an appropriate sensor and motor combination. The linearisation point and K_s will remain the same as in Phase One, being 30° and 1.8 respectively. The equilibrium values are as follows:

$$\begin{bmatrix} x_1^* & x_2^* & x_3^* & x_4^* & x_5^* \end{bmatrix} = \begin{bmatrix} 31.56 & 30 & 0 & 0 & 1.8167 \end{bmatrix} \tag{5}$$

The incremental variables were set to the following:

$$\delta_x = \begin{bmatrix} \delta_{x_1^*} & \delta_{x_2^*} & \delta_{x_3^*} & \delta_{x_4^*} & \delta_{x_5^*} \end{bmatrix}^T = \begin{bmatrix} x_1 - x_1^* & x_2 - x_2^* & x_3 - x_3^* & x_4 - x_4^* & x_5 - x_5^* \end{bmatrix}^T \tag{6}$$

$$\delta_u = u - u^* \tag{7}$$

The selection of the b value remains trivial, as it does not carry over into future calculations once the linearisation point is substituted. The linear system model of the plant and motor combination is depicted by Equations 8 and 9.

$$\dot{\delta_x} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -\frac{K_s}{J_m} & \frac{K_s}{J_m} & -\frac{2b|x_3 j|}{J_m} & 0 & \frac{K_m}{J_m} \\ \frac{K_s}{J_l} & -\frac{K_s}{J_l} - \frac{mgl\cos(x_2 j)}{J_l} & 0 & 0 & 0 \\ 0 & 0 & -\frac{K_m}{L_a} & 0 & -\frac{R_a}{L_a} \end{bmatrix} \delta_x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{L_a} \end{bmatrix} \delta_u \tag{8}$$

$$y(t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \end{bmatrix} \bar{x} \tag{9}$$

2.2 Performance Requirements

As per Phase One, the control system was tasked with moving the arm repeatedly between 15° and 45° angular positions after 3 second intervals, where the arm's angle θ_l must track a square wave with a 15° amplitude, 30° offset, and a 6-second period. Moreover, its performance requirements are as follows:

- Steady state error of at most 2 degrees for the arm angular position θ_l.
- Maximum magnitude of 5 degrees for any overshoot/undershoot of θ_l when the reference signal changes direction.
- Settling time (t_s) of at most 2.0 seconds (for a less than 2% settling time of δ).
- Maximum controller sampling frequency of 50 Hz.

To impose tighter requirements, settling time was changed to be of at most 1 second (for a less than 1% settling time). The maximum overshoot/undershoot percentage (M_p), damping ratio(ζ), natural frequency (w_n) and crossover frequency (w_c) were calculated to be the values depicted in Table 1. Note that to avoid multiple crossings of the 0dB line, a lower bound on the crossover frequency of 9.11rad/s was set.

Parameter	Value
M_p	16.67%
ζ	0.4954
w_n	9.5797
w_c	9.1728

Table 1: System Requirements

3 Project Tasks

3.1 Task A: Controllability and Observability Analysis

Controllability is the ability of a control system to drive its state from any given initial condition to any desired final state within a finite time interval via an appropriate control input. The controllability matrix Γ_c can be derived using Equation 10. Moreover, the system is completely controllable if and only if the controllability matrix has full rank.

$$\Gamma_c = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} \tag{10}$$

Observability refers to whether the internal states of a system can be inferred or from its external outputs over time. The observability matrix Γ_o can be derived using equation 11. Similarly, the system is completely observable if and only if the observability matrix has full rank.

$$\Gamma_o = \begin{bmatrix} C & CA & CA^2 & \cdots & CA^{n-1} \end{bmatrix}^T \tag{11}$$

Using the inbuilt MATLAB function *rank*, observability and controllability can be obtained. The controllability and observability of Motors 1 and 2 in combination with different sensors is illustrated by Tables 2 3. Note that the tolerance (tol) was set to $1e-10$ For sensors with multiple models, their respective cost is separated by "/" and is in ascending price order. The cost of Motor 1 and 2 is \$197.45 AUD and \$244.61 respectively. Note that from Phase One, Motor 3 did not meet the torque requirements and hence, was not used in this comparison.

Sensor	$rank(\Gamma_c)$	$rank(\Gamma_o)$	Sensor Cost (AUD)	Total Cost (AUD)
Vision	5	5	\$39/135/416	\$236.45 - \$613.45
Absolute Encoder	5	5	\$4/15/70	\$201.45 - \$267.45
Potentiometer	5	5	\$3/19/25/34/34	\$200.45 - \$231.45
Tachometer	5	5	\$17/32/40	\$214.45 - \$237.45
Gyroscope	5	5	\$28/30	\$225.45 - \$227.45

Table 2: Controllability and Observability Analysis for Individual Sensor and Motor 1 Pairing

Sensor	$rank(\Gamma_c)$	$rank(\Gamma_o)$	Sensor Cost (AUD)	Total Cost (AUD)
Vision	5	5	\$39/135/416	\$283.61 - \$660.61
Absolute Encoder	5	5	\$4/15/70	\$248.61 - \$314.61
Potentiometer	5	5	\$3/19/25/34/34	\$243.61 - \$278.61
Tachometer	5	5	\$17/32/40	\$264.61 - \$284.61
Gyroscope	5	5	\$28/30	\$272.61 - \$274.61

Table 3: Controllability and Observability Analysis for Individual Sensor and Motor 2 Pairing

Comparing Tables 2 and 3, the system is both completely observable and controllable when using each sensor. Therefore, irrespective of the motor and sensor combination, the system will always be completely controllable and observable. Table 4 depicts three sensor combinations.

No	Sensor Combination	$rank(\Gamma_c)$	$rank(\Gamma_o)$	Sensor Cost (AUD)
1	Encoder + Gyroscope + Tachometer	5	5	\$53 - \$210
2	Encoder + Gyroscope	5	5	\$32 - \$100
3	Encoder + Tachometer	5	5	\$21 - \$110

Table 4: Sensor Combinations

3.2 Task B: System Analysis without Motor Dynamics

Motor dynamics introduce additional states due to electrical characteristics like armature inductance, resistance, and back EMF, increasing the system's order and complicating control design. These fast electrical dynamics occur much quicker than mechanical ones, adding complexity to controller tuning. In the context of this task, a quasi-steady-state system model is considered where $\dot{x}_5 = 0$. This enables the control design to be simplified whilst retaining its essential characteristics. By setting $\dot{x}_5 = 0$, Equation 3 can be rearranged to form shown in Equation 12.

$$x_5 = -\frac{K_m}{R_a}x_3 + \frac{1}{R_a}u \tag{12}$$

Equation 12 can now be substituted into Equation 4 to get x_5 in terms of x_3 and u as shown below.

$$T_m = K_m(-\frac{K_m}{R_a}x_3 + \frac{1}{R_a}u) \tag{13}$$

This expression can then be substituted into Equation 2. After linearisation, the simplified reduced-order model can then be derived from this new expression and is depicted by Equation 14.

$$\dot{\delta_{x_{red}}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{K_s}{J_m} & \frac{K_s}{J_m} & -\frac{2b|x_3^*|}{J_m} - \frac{K_m^2}{J_m R_a} & 0 \\ \frac{K_s}{J_l} & -\frac{K_s}{J_l} - \frac{mgl\cos(x_2 j)}{J_l} & 0 & 0 \end{bmatrix} \delta_{x_{red}} + \begin{bmatrix} 0 \\ 0 \\ \frac{K_m}{J_m R_a} \\ 0 \end{bmatrix} \delta_u \tag{14}$$

‘

The new incremental variables are as follows:

$$\delta_{x_{red}} = \begin{bmatrix} \delta_{x_1^*} & \delta_{x_2^*} & \delta_{x_3^*} & \delta_{x_4^*} \end{bmatrix}^T = \begin{bmatrix} x_1 - x_1^* & x_2 - x_2^* & x_3 - x_3^* & x_4 - x_4^* \end{bmatrix}^T$$

Substituting the same linearisation point used in Phase One, the matrices A_{red}, B_{red} and C_{red} are as follows:

$$A_{red} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -240000 & 240000 & -48.6 & 0 \\ 1800 & -1885 & 0 & 0 \end{bmatrix} \quad B_{red} = \begin{bmatrix} 0 \\ 0 \\ 1800 \\ 0 \end{bmatrix} \quad C_{red} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$$

Subsequently, the final reduced state-space representation is: $\dot{\delta_x} = A_{red}\delta_{x_{red}} + B_{red}\delta_u$. The observability and controllability matrix ranks of this system model are both equal to four. Note that this is for the selected motor and sensor combination used in Phase One. Since both Γ_c and Γ_o have full rank equal to the number of states, the reduced-order model is completely controllable and observable. Next, the eigenvalues of this simplified system are to be compared to the full system model. Since these eigenvalues encapsulate how the system behaves to external inputs and disturbances, this comparison is necessary to validate the simplified model's accuracy in capturing the essential dynamics. More simply, if the eigenvalues of the reduced-order model closely resemble that of the full model, then the reduced-order model can be considered a reliable representation for the controller design. However, a substantial difference in eigenvalues would suggest that the motor dynamics are integral to the system's behaviour. The eigenvalues of both the reduced-order and full models are depicted by Table 5. For reference, the full model is outlined in Section 2.1.

Reduced-Order Model	**Full Model**
$-24.119 \pm 49.112i$	-1493.3
$-0.181 \pm 9.181i$	$-22.42 \pm 49.85i$
	$-0.1811 \pm 9.183i$

Table 5: Comparison of Eigenvalues of the Reduced-order and Full System Models

According to Table 5, both models are stable since all eigenvalues have negative real parts. The close correspondence of the complex conjugate pairs indicates that the reduced-order model effectively captures the primary dynamics of the full model. Moreover, although the full model includes an additional fast pole at -1493.3rad/s representing the omitted motor dynamics, it settles rapidly and has a negligible long-term impact on system behaviour.

3.3 Task C: State-Feedback Control

For this Section and beyond, Motor 2 will be used. In full-state feedback control, the goal is to use the complete state information of a system to influence its behaviour by feeding back all measured state variables. Consider the system described by the state-space equations:

$$\begin{aligned} \dot{x}(t) &= A_{\text{red}}\, x(t) + B_{\text{red}}\, u(t) \\ y(t) &= C_{\text{red}}\, x(t) \end{aligned} \tag{15}$$

and suppose that the loop is closed using the control law:

$$u(t) = \bar{r} - Kx(t)$$

Substituting this control input into the state equation yields the following state equation:

$$\dot{x}(t) = A_{red}x(t) + B_{red}(\bar{r} - Kx(t)) = (A_{red} - B_{red}K)x + B_{red}\bar{r} \tag{16}$$

The expression $(A - BK)$ is the closed-loop A matrix. By adjusting the state-feedback gain matrix (K), pole placement is employed to reassign poles of the closed loop system such that system performance is enhanced w.r.t its performance requirements. Additionally, to eliminate steady-state errors in response to step inputs, integral control is employed. This ensures that the control error integrates over time towards zero, enhancing system robustness.

3.3.1 Pole Placement

To implement integral control, the integral error depicted below is added to the state vector.

$$\dot{x_1} = y - r = Cx - r = e$$

The augmented state-space model is :

$$\begin{bmatrix} \dot{x} \\ \dot{x}_I \end{bmatrix} = \begin{bmatrix} A_{red} & 0 \\ C_{red} & 0 \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix} + \begin{bmatrix} B_{red} \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ -I \end{bmatrix} r$$

where $A_{aug} = \begin{bmatrix} A_{red} & 0 \\ C_{red} & 0 \end{bmatrix}$ and $B_{aug} = \begin{bmatrix} B_{red} \\ 0 \end{bmatrix}$. The full-state feedback control law u for this model is defined as:

$$u = -\begin{bmatrix} K & K_I \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix}$$

Based on the constraints in Section 2.2, the lower bound for the dominant poles can be calculated by substituting those constraints into the following equation:

$$s = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \tag{17}$$

This yields the following poles:

$$s_{1,2} = -4.746 \pm 8.3214\,i$$

Since the complex conjugates $s_{1,2}$ have a multiplicity of two, this requires the rank of $\begin{bmatrix} B_{red} \\ 0 \end{bmatrix}$ to be at least two; however, its rank is only one. To resolve this, the second dominant pole is selected to be at $s_2 = -5$rad/s. This gives us the $s_1 = -4.746$ and $s_2 = -5$ as the dominant poles (note that the imaginary component of s_1 is ignored as a complex gain matrix is required when the pole locations are not specified as complex conjugate pairs).

3.3.2 C1: Continuous-Time Controller Design

The next step is to now select the remaining three poles. To start, these poles are selected to be 5 times larger than the dominant poles, and slowly increased until the response meets the performance criteria. Note that we should not place these poles excessively distant from the imaginary axis, as such placements typically correspond to high gains. These high gains demand actuators with substantial speed and authority, which may exceed the capabilities of the selected actuator. Consequently, this mismatch can cause the actuator to reach saturation, impairing system performance. This is accounted for via a saturation block within the continuous time simulation depicted in Figure 1 and is checked every time the poles are changed. Via this method, the selected poles are:

$$\begin{bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 \end{bmatrix} = \begin{bmatrix} -4.746 & -5 & 150 & -155 & -165 \end{bmatrix}$$

Using the inbuilt MATLAB function *place*, A_{aug} and B_{aug}, K and K_I are found to be:

$$K = \begin{bmatrix} -90.95 & 101.03 & 0.24 & 1.13 \end{bmatrix} \quad K_I = 28.1$$

To emulate the system at 50Hz, the controller can be discretised by placing zero-order hold blocks before and after the integral block as seen in Figure 2. The performance of the nonlinear and emulation controllers are depicted in Figure 2. Note that for task C, it is assumed that there exist sensors that give perfect measurements of the states.

3.3.3 C2: Discrete-Time Controller Design

Using the pole placement method, an appropriate discrete-time full-state feedback controller is designed. Firstly, the poles obtained in Section 3.3.2 are converted into discrete-time poles equivalence using the equation $z = e^{sT}$, where z are the discrete-time poles, s are the continuous-time poles and T is the sampling time. The discrete-times poles are shown below:

$$z = \begin{bmatrix} 0.9094 & 0.9048 & 0.0498 & 0.0450 & 0.0369 \end{bmatrix}$$

Since a sampling frequency of 50Hz must be used both for emulation and discrete-time design, $T = 0.02s$. Using the MATLAB routine *c2d*, the reduced matrices are discretised at the required sampling time using zero-order hold (ZOH). Similar to continuous time, the integral error depicted below is to be added to the state vector.

$$x_I(k+1) = x_I(k) + y(k) - r(k) = x_I(k) + Cx(k) - r(k)$$

Consequently, the augmented state-space model is

$$\begin{bmatrix} x_I(k+1) \\ x(k+1) \end{bmatrix} = \begin{bmatrix} 1 & C \\ 0 & \Phi \end{bmatrix} \begin{bmatrix} x_I(k) \\ x(k) \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & \Gamma \end{bmatrix} \begin{bmatrix} r(k) \\ u(k) \end{bmatrix}$$

where Φ and Γ are the discretised versions of A_{red} and B_{red} and the augmented matrices are $\Phi_{aug} = \begin{bmatrix} 1 & C \\ 0 & \Phi \end{bmatrix}$ and $\Gamma_{aug} = \begin{bmatrix} 0 \\ \Gamma \end{bmatrix}$. Again, *place* is used to derive the following controller gains:

$$K = \begin{bmatrix} -41.584 & 50.136 & 0.086 & 1.467 \end{bmatrix} \quad K_I = 0.5613$$

The discretised controller response is depicted in Figure 2.

3.3.4 Controller Performance

The controllers demonstrate strong performance in tracking the reference signal, with minimal steady-state error and controlled overshoot and undershoot, all within acceptable bounds. Both the discretised and emulation models closely align with the nonlinear model, indicating robustness across different system representations. The system exhibits a quick settling time, and while slight deviations are observed between the models, no significant performance degradation is observed. Overall, the controller meets the design objectives for accuracy and stability.

3.4 Task D: Observer-Based Feedback Control

Due to factors such as the impracticality of measuring certain variables or the prohibitive costs associated with additional sensors, direct measurement of all system states is sometimes not feasible. In such circumstances, observers can be used to facilitate the estimation of unmeasured states by utilising available outputs, inputs and the model of the plant dynamics. Based on the observability and cost analysis performed in Section 3.1, an observer-based feedback controller is to be designed.

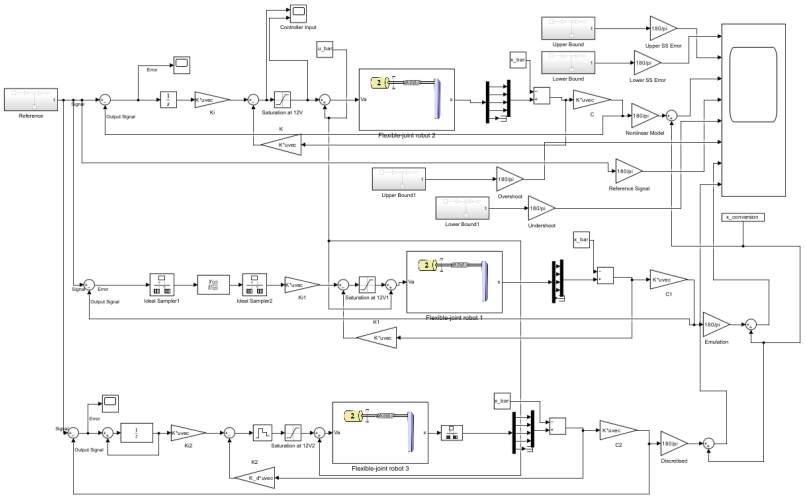


Figure 1: Task C Simulink

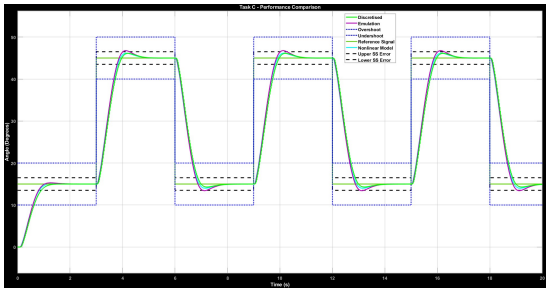


Figure 2: Task C Controller Comparison

3.4.1 Sensor Selection

Before designing an observer, appropriate sensors must be picked. From Section 3.1 and in the context of the current task, Tables 2 and 3 indicate that all states can be estimated irrespective of the sensor choice. Moreover, since these state estimations would depend on the quality of the measurements, it would be best to pick sensor combinations that provide the most accurate data, as noisy or inaccurate data will cause all the estimated states to be wrong. Firstly, the performance of the sensors is tested against the "ideal" sensor measurements, as illustrated by Figure 3. Observing this Figure, the absolute encoder seems to measure the truth the best. Next, the encoder results are parsed into the following observer block depicted by Figure 4, where the estimated states based on the encoder are to be compared to the "ideal" values. Note that for testing purposes, the observer gain (J) values are derived from the same observer poles (λ_{obs}) defined in Section 3.4.2. Observing Figures 5a and 5b, the estimated states track θ_m and θ_l quite well, however tracks $\dot{\theta_m}$ and $\dot{\theta_l}$ quite poorly.

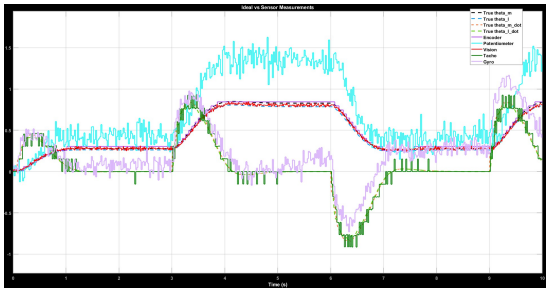


Figure 3: Ideal vs Sensor Measurements

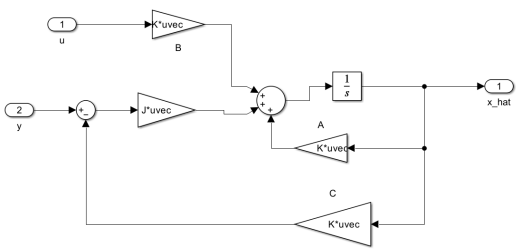
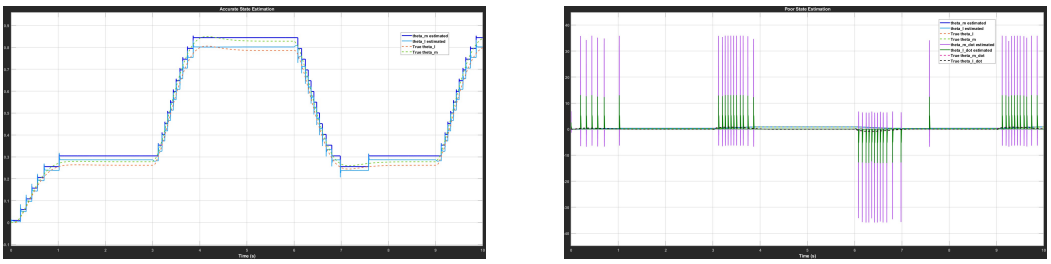


Figure 4: Full State Observer Block

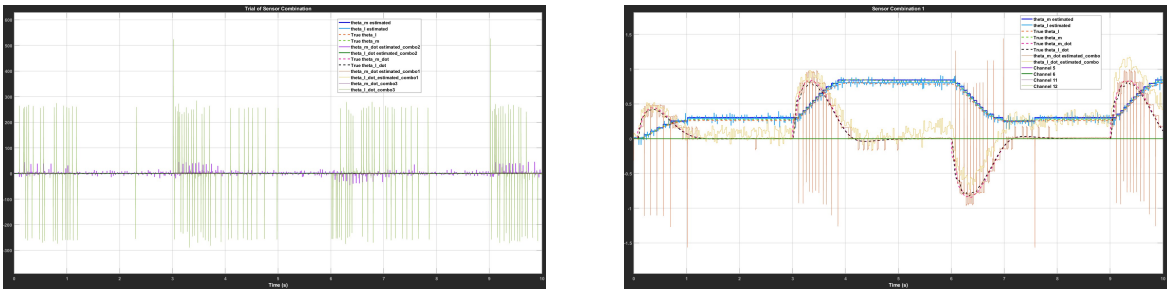


(a) Accurate State Estimation

(b) Poor State Estimation

Figure 5: Comparison of State Estimation Accuracy

To account for this issue, the sensor combinations from Table 4 are trialled. Observing Figures 6a and 6, combination 3 estimates $\dot{\theta}_l$ quite poorly. Combinations 2 and 1 estimates $\dot{\theta}_l$ relatively well in comparison however, combination 2 estimates $\dot{\theta_m}$ inaccurately. Whilst Combination 1 estimates $\dot{\theta_m}$ better than Combination 2, Combination 2 is roughly half the price of Combination 1 and therefore, is the selected sensor combination. Ultimately, the *Bourns* encoder and *nvensense* gyroscope were chosen, resulting in a total sensor price of $45. In addition to Motor 2, this comes to $289.61.



(a) Trial of Task A Sensor Combinations

(b) Sensor Combination 1

Figure 6: Comparison of Sensor Test Trials and Sensor Combination

3.4.2 D1: Continuous-Time Observer Design

For a continuous-time system described by Equation 15 and supposing that we have measurements $y(t)$ and wish to produce an estimate for $\hat{x(t)}$, a closed-loop observer can be designed with the following dynamics:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + J(y(t) - C\hat{x}(t))$$

It can be shown the error dynamics $\tilde{x} := x - \hat{x}$ can be described by

$$\dot{\tilde{x}}(t) = (A - JC)\tilde{x}(t)$$

Since the system is completely observable, as established in Section 3.1, we can place the observer poles arbitrarily by appropriately choosing the observer gain matrix J. Leveraging the principle of duality, the design of observer gains mirrors that of state feedback controller gains. An iterative process is employed to place the observer poles. Starting by assigning poles a decade to the left of the fastest pole designed for the full-state feedback controller, we incrementally adjust these values to meet the performance specifications. However, poles cannot be assigned excessively far from the imaginary axis, as high observer gains can amplify sensor noise. Moreover, measurement noise can indirectly affect system performance via potential actuator saturation due to large estimation errors. The resulting observer poles are chosen as follows:

$$\lambda_{obs} = \begin{bmatrix} -3000 & -2100 & -2200 & -2300 \end{bmatrix}$$

The resulting gain matrix is:

$$J = \begin{bmatrix} 4627.01 & 151.80 \\ -607.97 & -3177.19 \\ 4908449.58 & -17314.26 \\ 2317.77 & 4924.39 \end{bmatrix}$$

In this design, it is important to note the separation principle, which states that the observer and controller can be designed independently without affecting the overall system stability, meaning that the closed-loop poles of the combined observer-controller system is simply the "union" of these poles. The resulting nonlinear and emulated controllers can be observed in Figure 7.

3.4.3 D2: Discrete-Time Observer Design

As per Section 3.4.1, the states θ_m and $\dot{\theta}_l$ are being measured directly. Whilst a full-state estimator can serve as a low pass filter and "smooth" these measurements, it is not necessary to reconstruct the unmeasured states. A reduced-order observer can be used instead. This is advantageous as since a reduced order estimator is of lower order, it will require less computational power. Consider the reduced order model derived in Section 3.2. The states of the model can be partitioned into measured x_1 and estimated x_2.

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{12} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

For $\hat{x_2} = z + J_r x_1$, we can define the observer error as $\tilde{x_2} = z + J_r - \hat{x_2}$. The observer dynamics are given below and are constructed in Figure 8.

$$\dot{z} = J_r(-A_{11}x_1 - B_1 u) + A_{21}x_1 + B_2 u + (A_{22} - J_r A_{12})(z + J_r x_1)$$

The error dynamics are expressed as $\dot{\tilde{x_2}} = (A_{22} - J_r A_{12})\tilde{x_2}$. Again, J_r can be designed by selecting poles for $A_{22} - J_r A_{12}$. The poles and observer gain are as follows. Note that the poles shown here are in the continuous time domain which are converted into z domain.

$$\lambda_{\text{obs}} = \begin{bmatrix} -3000 & -4000 \end{bmatrix}$$

$$J_r = \begin{bmatrix} -120.91 & -50.595 \\ 0.651 & 0.0319 \end{bmatrix}$$

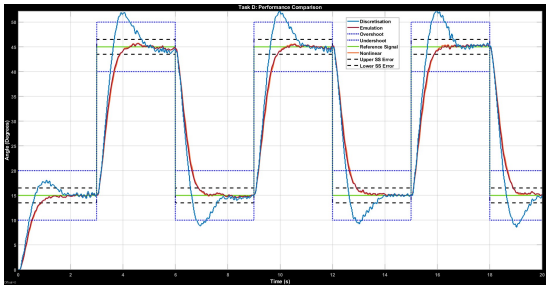


Figure 7: Task D: Controller Comparison

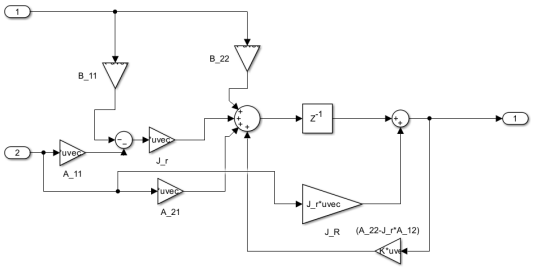


Figure 8: Reduced Order Observer Model

3.4.4 Controller Performance

Observing Figure 7, the nonlinear and emulated controllers satisfy all the performance requirements. The discretised controller only satisfies the steady state requirement as it exhibits slightly excessive overshoot. Whilst this could be attributed to poor controller tuning, it was found that further increasing controller gains lead to saturation. Therefore, this phenomenon could be due to insufficient sampling frequency. Additionally, there as an asymmetry present between the step-up and step-down response of the discrete controller. This could be attributed to system dynamics, particularly inertia and actuator limitations.

3.5 Task E: Sinusoidal Reference Tracking

In this section, a controller is designed to track the below sinusoidal signal of 25 degrees amplitude (A), 25 degrees offset (B) and a period of 4 seconds for the arm angular position θ_l. Note that $\omega = \frac{\pi}{2}$.

$$R(s) = \frac{A\omega}{s^2 + \omega^2} + \frac{B}{s} \tag{18}$$

Moreover, the controller is to have zero steady-state error and the actual steady-state error should be no greater than 5 degrees. To design an appropriate controller, the performance requirements are firstly converted into frequency domain.

3.5.1 Performance Requirements

Assuming a maximum allowable overshoot of 5° for a 25° reference change (i.e., $M_p = 20\%$), the damping ratio ζ is calculated using:

$$\zeta > \sqrt{\frac{(-\ln M_p)^2}{\pi^2 + (-\ln M_p)^2}} = 0.456$$

Selecting a settling time $t_s \leq 5$ seconds for a 5% criterion ($\delta = 5\%$), the natural frequency ω_n is determined by:

$$\omega_n = \frac{-\ln(\delta/100)}{\zeta t_s} = 1.365\,\text{rad/s}$$

The corresponding crossover frequency is then:

$$\omega_c = \omega_n \sqrt{\sqrt{1 + 4\zeta^2} - 2\zeta^2} \geq 1.322\,\text{rad/s}$$

Thus, the dominant poles are located at:

$$s_{1,2} = -\zeta\omega_n \pm j\,\omega_n\sqrt{1 - \zeta^2} = -0.623 \pm j\,1.22$$

3.5.2 Controller Design for Sinusoidal Reference

To track the sinusoidal reference signal, the internal model principle (IMP) is employed. By embedding the dynamics of the sinusoidal reference directly into the controller, the system is able to generate the necessary control actions to track the reference accurately. This is achieved by ensuring that the generating polynomial of the sinusoidal signal is included in the controller's design, allowing the system to eliminate steady-state error and respond effectively to the reference input. Thus, the controller achieves asymptotic reference tracking and meets the performance criteria. Observing Equation 18, the reference signal can be broken up into two components; a sinusoidal and step input. Converting this expression into state-space form, we obtain the following state-space matrices. Note that the system is third order because the sinusoidal component introduces two poles, while the step input adds an additional pole at the origin. Together, these dynamics result in a third-order system, ensuring accurate tracking of the reference signal with zero steady-state error.

$$A_d = \begin{bmatrix} 0 & -\omega^2 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B_d = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad C_d = \begin{bmatrix} A & 0 & 0 \\ 0 & A\omega & 0 \\ 0 & 0 & A\omega^2 \end{bmatrix}, \quad D_d = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Next let the tracking error $e(t) \in \mathbb{R}$ and the input $u_p(t) \in \mathbb{R}$ be related via the state-space model

$$\begin{aligned} \dot{\rho}(t) &= A_d\rho(t) + B_d e(t), \\ u_p(t) &= K_p\rho(t). \end{aligned} \tag{19}$$

Noting that $e(t) = r(t) - C_r x(t)$, the augmented system dynamics is described by

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\rho}(t) \end{bmatrix} = \begin{bmatrix} A & 0_{n\times q} \\ -B_d C_r & A_d \end{bmatrix} \begin{bmatrix} x(t) \\ \rho(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0_{n\times 1} \\ -1 \end{bmatrix} r(t),$$
$$y_r(t) = \begin{bmatrix} C_r & 0_{1\times q} \end{bmatrix} \begin{bmatrix} x(t) \\ \rho(t) \end{bmatrix}.$$

where

$$A_{aug} = \begin{bmatrix} A & 0_{n\times q} \\ -B_d C_r & A_d \end{bmatrix}, \quad B_{aug} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad C_{aug} = \begin{bmatrix} C_r & 0_{1\times q} \end{bmatrix} \tag{20}$$

Using the same pole placement method described in Section 3.3.2, the selected poles are:

$$\begin{bmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \end{bmatrix} = \begin{bmatrix} -0.623 & -0.723 & -50 & -60 & -65 & -75 & -80 \end{bmatrix}$$

Using the inbuilt MATLAB function place, A_{aug} and B_{aug}, K and K_p are found to be:

$$K = \begin{bmatrix} -110.093 & 138.264 & 0.157 & 0.695 \end{bmatrix} \quad K_p = \begin{bmatrix} -397.138 & -426.093 & -162.389 \end{bmatrix}$$

Following the same rationale and method described in Section 3.4.2, the observer poles are as follows:

$$J = \begin{bmatrix} -700 & -710 & -720 & -750 \end{bmatrix}$$

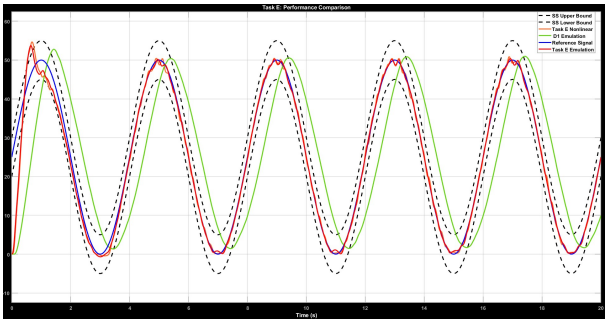


Figure 9: Controller Comparison

3.5.3 Controller Performance Comparison

Observing Figure 9, both the nonlinear and emulated controllers developed within task E successfully meet the $\pm 5^\circ$ steady state requirement. The initial response of both curves show a slight overshoot at the start, but then quickly settles within 2 seconds, well beyond the settling time of 5 seconds set in Section 3.5.1. Additionally, both curves exhibit oscillation at the peaks and troughs of the tracked signal. This might suggest that the controller is more sensitive to low rates of change in the reference signal and might require an increase in the damping of the controller. Alternatively, the change in motor direction at these points, where system inertia might contribute to the oscillations as the controller compensates to maintain tracking. The D1 Emulated controller whilst matching the shape of the reference signal, suffers from phase lag, demonstrating that this controller is less capable of accurately tracking the reference signal.

4 Discussion

While pole placement offers a systematic approach for assigning closed-loop poles to achieve desired dynamics, it has limitations such as sensitivity to model inaccuracies and the requirement for high gain values, which can lead to actuator saturation and amplification of sensor noise. Moreover, the trade-off between performance and control effort is not considered, potentially leading to inefficient control actions. In future iterations of this project, the potential adoption of Linear Quadratic Regulator (LQR) techniques could address these issues by optimising gain values based on a cost function, thus enhancing robustness and performance.

Typically, whilst direct discrete-time design is more complex, discrete controller design usually better accounts for sampling effects and can improve system performance over emulated continuous-time controllers. However, when observing the results from Tasks C and D, emulation performed better. Therefore, a revision of the controller design may be necessary.

5 Conclusion

Phase 2 of the Design Project was centred around the design of full-state feedback controllers and observers using arbitrary pole placement. Overall, the performance of the continuous-time and emulated controllers was acceptable, meeting all the performance requirements. The discrete controller designed in 3.3 also satisfied the specifications, however the one designed in 3.4 did not, demonstrating too much overshoot.