ELEN90064
Advanced Control Systems

Phase 2 - Single-Link Flexible-Joint Robot

Xing Yang Goh
1001969
October 1, 2022

1 Introduction

The system a controller will be designed for is a single-link flexible-joint robot with motor 1 as the actuator as shown in figure 1. The governing equations have been previously derived, with a chosen linearisation point is at $x_{1e} = 45°$, $x_{2e} = 45°$.. In this report, state-space design methodologies using full state controller and observer designs will be covered, using the separation principle to develop these controllers and observers.

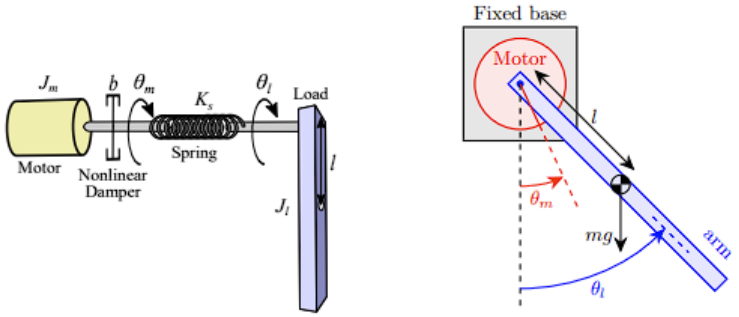


Figure 1: Schematic of a single-link flexible-joint robot

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} \theta_m \\ \theta_l \\ \dot{\theta}_m \\ \dot{\theta}_l \\ I_a \end{bmatrix} \quad ; \quad \begin{bmatrix} \dot{\tilde{x}}_1 \\ \dot{\tilde{x}}_2 \\ \dot{\tilde{x}}_3 \\ \dot{\tilde{x}}_4 \\ \dot{\tilde{x}}_5 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -\frac{K_s}{J_m} & \frac{K_s}{J_m} & 0 & 0 & \frac{K_m}{J_m} \\ \frac{K_s}{J_l} & -\frac{K_s}{J_l} - \frac{mgl}{\sqrt{2}J_l} & 0 & 0 & 0 \\ 0 & 0 & -\frac{K_m}{L_a} & 0 & -\frac{R_a}{L_a} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \\ \tilde{x}_4 \\ \tilde{x}_5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{L_a} \end{bmatrix} \tilde{u} \qquad (1)$$

The specifications for the physical non-linear system are the following parameters:

1. Maximum overshoot of at most 5 degrees.
2. Settling time of at most 2.0 seconds (for a less than 2% settling criterion).
3. Steady state error of at most 2 degrees.
4. Maximum controller sampling frequency of 50 Hz.

2 Controllability and Observabillity

To calculate the controllability of the system, the rank of the matrix $\Gamma_c[\mathbf{A}, \mathbf{B}] \triangleq \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A^2B} & \mathbf{A^3B} & \mathbf{A^4B} \end{bmatrix}$ is calculated. Using the A and B state space equation defined in equation 1, we can determine that the system is completely controllable since the row rank of the controllability matrix is 5.

For the observability calculations, a matrix rank calculation is performed for the observability matrix $\Gamma_c[\mathbf{A}, \mathbf{C}] \triangleq \begin{bmatrix} \mathbf{C} & \mathbf{CA} & \mathbf{CA^2} & \mathbf{CA^3} & \mathbf{CA^4} \end{bmatrix}^T$. Using calculations for each sensor individually, we conclude that the system is completely observable with a full column rank of 5 for each individual sensor.

2.1 Absolute Encoder & Vision Sensor

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

Looking at sensor combinations, the first combination is using an absolute encoder and a vision sensor. This combination produces a completely observable system, with the absolute encoder with 4096 counts per revolution priced at $70 and the 50fps vision sensor priced at $135, leading to a total cost of $205

2.2 Absolute Encoder & Vision Sensor

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

Looking at sensor combinations, the first combination is using an absolute encoder and a vision sensor. This combination produces a completely observable system, with the absolute encoder with 4096 counts per revolution priced at $70 and the 50fps vision sensor priced at $135, leading to a total cost of $205.

2.3 Absolute Encoder & Gyro

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

The absolute encoder and gyro sensor combination produces a completely observable system, with the absolute encoder with 4096 counts per revolution priced at $70 and the 5deg/s^2 gyroscope priced at $28, leading to a total cost of $98.

2.4 Vision Sensor & Tachometer

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

The vision sensor and tachometer combination produce a completely observable system, with the tachometer with 2048 counts per revolution priced at $40 and the 50fps vision sensor priced at $135, leading to a total cost of $175.

2.5 Gyro & Potentiometer

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

The gyro and tachometer combination produces a completely observable system, with the tachometer with 2048 counts per revolution priced at $40 and the 5deg/s^2 gyroscope priced at $28, leading to a total cost of $68.

2.6 Tachometer & Potentiometer & Gyro

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

The gyro, tachometer and potentiometer combination produces a completely observable system, with the tachometer with 2048 counts per revolution priced at $40, the 5deg/s^2 gyroscope priced at $28 and the 0.2% linearity potentiometer priced at $34 leading to a total cost of $102.

3 Full State Feedback Control

3.1 Continuous Controller

Using the separation principle, we will begin by designing a controller for the completely controllable system that satisfies the performance specifications assuming ideal sensors that measure the full state of the system. Using the pole placement design strategies, we will utilise second-order design principles as a guideline for pole placements. For a maximum overshoot of at most 5 degrees, we can estimate a minimum overshoot percentage M_p when the system moves 30°, which is the maximum step size for the system due to the nominal operation of the system between 30° and 60°. $M_p = \frac{5}{30} \times 100 = 16.67\%$. Using the relationship $M_p = \exp\left(\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}\right)$, we obtain $\zeta = 0.495$. Therefore, the bound for the complex domain angle beginning at the imaginary axis is:

$$\arcsin 0.495 = 29.7^\circ \tag{2}$$

For a setting time of at most 2 seconds, the 2% settling time criterion is $\sigma \geq \frac{4}{t_s}$, where sigma represents the distance between the imaginary axis and the pole boundary in the LHP.

$$\sigma \geq 2 \tag{3}$$

This leads to the bound for the complex domain angle from the imaginary axis at 29.7° and poles smaller than -2 as seen in figure 2.

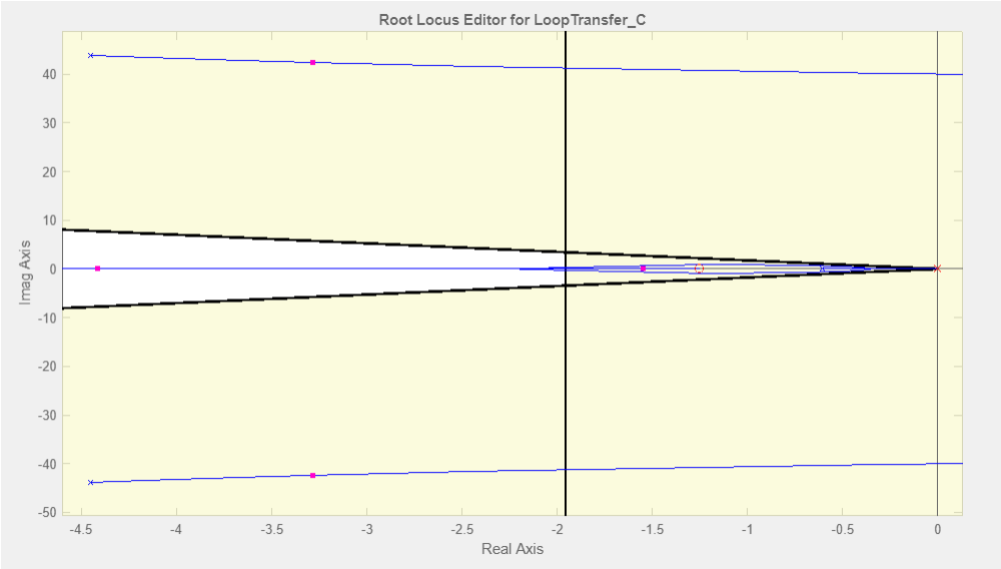


Figure 2: Pole placement bounds for the controller

For the choice controller, the system will be used to track signals between 30° and 60°. This can be modelled as reference pulse signals for the controller to track. Using this assumption, an integral controller will be the best choice as pulse functions are a sequence of step functions, in which a $\frac{1}{s}$ integrator term in the controller will allow the controller to converge to the reference signal with the internal model principle. This is required as the linearised plant model does not contain a $\frac{1}{s}$ term to satisfy the internal model principle. To implement this controller we first augment the state matrices to include the integral of the output error x_I, which will be from the link angle θ_l.

$$\begin{aligned} \begin{bmatrix} \dot{x}_I \\ \dot{\mathbf{x}} \end{bmatrix} &= \begin{bmatrix} 0 & \mathbf{C} \\ 0 & \mathbf{A} \end{bmatrix} \begin{bmatrix} x_I \\ \mathbf{x} \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} r \\ u \end{bmatrix} \\ u &= -\begin{bmatrix} K_I & \mathbf{K}_0 \end{bmatrix} \begin{bmatrix} x_I \\ \mathbf{x} \end{bmatrix} + \mathbf{K}_0 \mathbf{N}_x r \end{aligned} \tag{4}$$

Using these augmented matrices, we will place the poles of the controller at s = [-15+2.5j, -15-2.5j, -60, -70, -80, -90]. Note that this arbitrary pole placement method can be performed since the model is completely controllable, such that there exists $\mathbf{K} \triangleq [k_0, k_1, \ldots, k_{n-1}]$ such that $u(t) = \bar{r} - \mathbf{K}x(t)$ yields the closed-loop characteristic polynomial that defines the arbitrary pole locations. We can obtain the following state feedback gains and integral gain that corresponds to these pole locations to design our integral controller. These controller poles are chosen such that the pair of conjugate poles at -15±2.5j are slower than the remaining 4 poles, allowing the system to be approximated as a second-order system, allowing us to use the discussed design strategies as a rough guideline, however, manual tuning of these poles were performed in Simulink to ensure the motor does not saturate from overly large poles from large gains and to meet the design specifications on the nonlinear system model. Note that a saturation block of $\pm 6V$ is placed before the actuator input. Using Ackermann's formula, we can find a transformation to a controllable canonical form and extract the corresponding state gains that achieve the placed closed-loop poles. Note that for this case, the feed-forward gain $\mathbf{N}_x$ is set to 0 since the response of the controller is sufficiently fast.

3.2 Discrete Controller

To design the discrete controller in the z domain, we first determine the sampling time of the controller, which runs at 50Hz or a corresponding sampling time of 0.02s. Following this, we convert the state space model of

the linearised system in equation 1 into a discrete-time representation using zero-order hold. With this, we can produce the discrete-time augmented matrix for an integral controller.

$$\begin{aligned} \begin{bmatrix} x_I(k+1) \\ \mathbf{x}(k+1) \end{bmatrix} &= \begin{bmatrix} 1 & \mathbf{C} \\ \mathbf{0} & \mathbf{\Phi} \end{bmatrix} \begin{bmatrix} x_I(k) \\ \mathbf{x}(k) \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ \mathbf{0} & \mathbf{\Gamma} \end{bmatrix} \begin{bmatrix} r(k) \\ u(k) \end{bmatrix} \\ u(k) &= -\begin{bmatrix} K_I & \mathbf{K} \end{bmatrix} \begin{bmatrix} x_I(k) \\ \mathbf{x}(k) \end{bmatrix} + \mathbf{K}\mathbf{N}_x r(k) \end{aligned} \tag{5}$$

For the controller pole locations in the z-plane, the following matched z-transformation is used $z = e^{sT}$ since for z-domain poles, anything that falls out of the unit circle around the origin leads to an unstable controller that has exponential terms blowing to infinity. This transformation leads to poles of the controller at z = [0.74+0.037i, 0.74-0.037i, 0.301, 0.247, 0.202, 0.165]. This z-transformation method preserves the stability and minimum phase of the system for robustness.

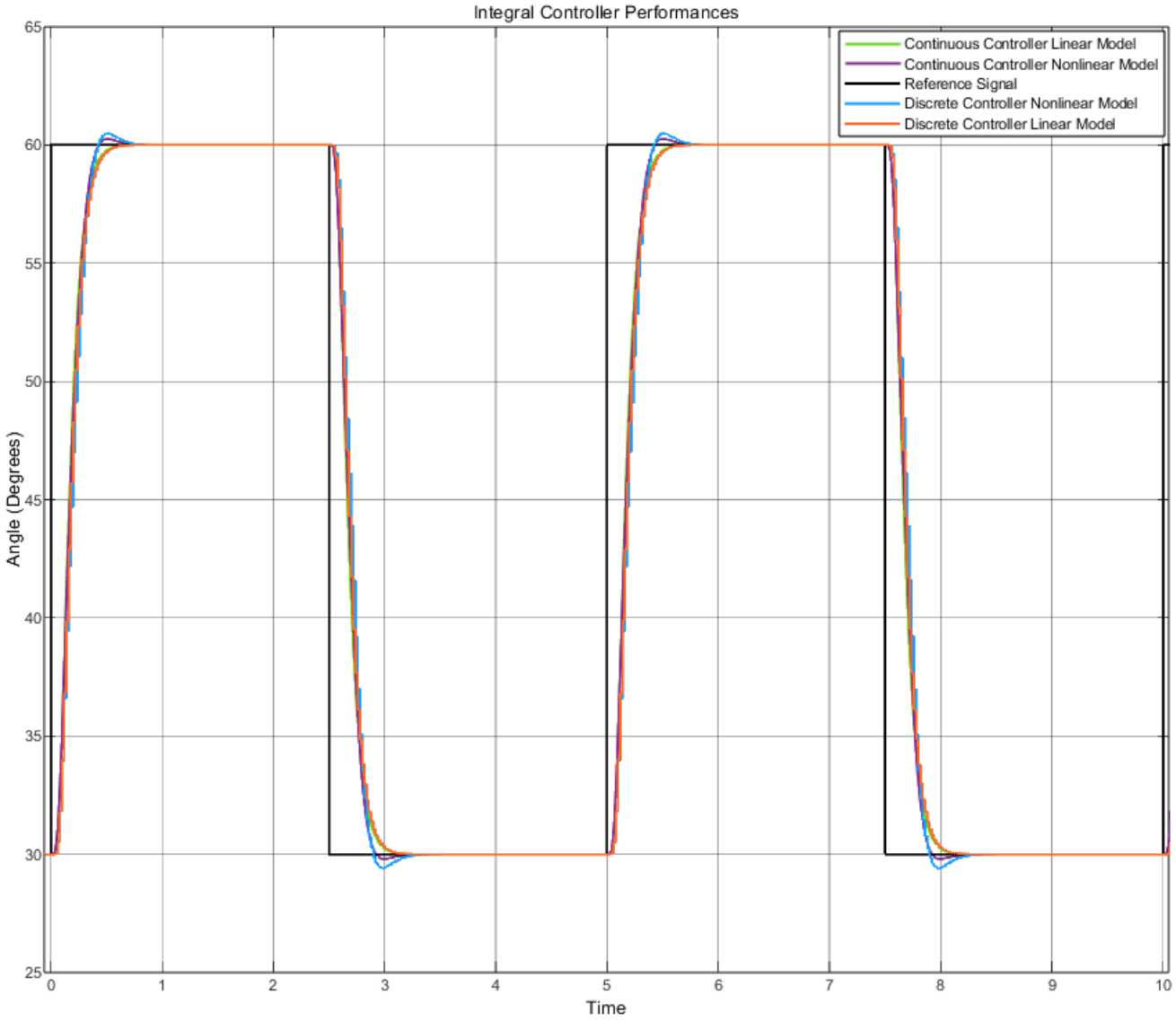
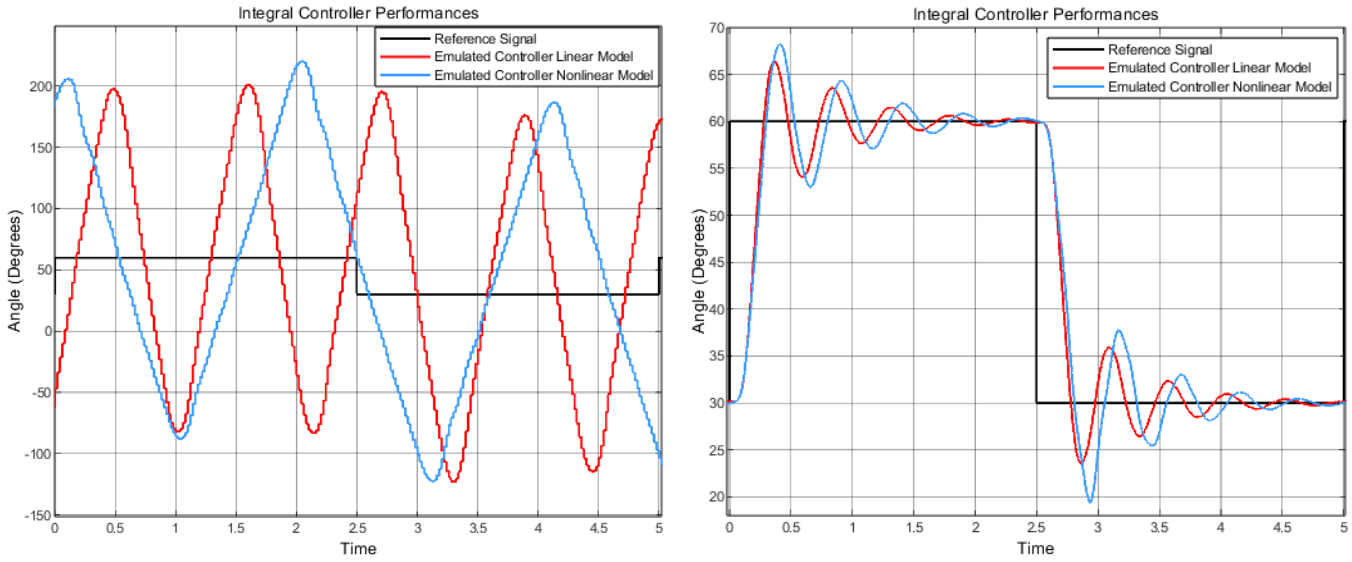


Figure 3: Performance of the linear and discrete controllers

Analysing the performance of the controllers, the controllers have a settling time of around 0.7s on the linear models and a settling time of 0.8s on the nonlinear model, which is almost a negligible difference and demonstrates the controller robustness from modelling uncertainties and linearisation assumptions. For both nonlinear models, there is some overshoot, with the discrete-time controller exhibiting a 0.5° overshoot and the continuous time controller exhibiting a 0.25° overshoot, which is also a negligible difference that is caused by the delay from the discretised system with a 0.02 sampling time. Overall, both the continuous and discrete controllers easily satisfy the performance specifications provided on the actual non-linear model.

3.3 Emulated Continuous Controller

Another method to discretise the continuous controller without z domain design techniques is to use emulation with zero-order hold blocks for the input and output of the plant models.



(a) Emulation with 0.02 sampling time

(b) Emulation with 0.004 sampling time

Figure 4: Performance of the emulated controllers

At the defined controller frequency of 50Hz, the emulated continuous controller is unstable, with large unstable oscillations due to aliasing, which is caused by a sampling time that is too low to capture the current state of the system. Increasing the controller frequency to 250Hz, the performance is improved with the controller becoming stable and able to converge, however, there is oscillatory behaviour in the system, with an overshoot of over 10°. Comparing the performance of this emulated controller with the discretised controller, the performance of the discretised system using the matched z-transform is vastly better, with a very high controller frequency required to meet the design specifications, which is very computationally expensive.

4 Observer Design

4.1 Luenberger Observer

Now to design a full state Luenberger Observer, we will be using the absolute encoder and vision sensor suite described in section 2.1. Note that there is a constant bias of 1° for the vision sensor which will be accounted for. This provides a completely observable system, which will be used to produce a full state estimate. Generally, we would place the observer poles one decade larger than the controller poles, which is done since observers are not physical systems with saturation points. This means that large gains in observers will not pose a physical constraint, however, this makes it susceptible to sensor noise as the large gains will amplify these disturbances. Through a rigorous design process for the observer poles, the poles are placed at s = [-9600, -10800, -12000, -13200, -14400]. This makes it 2 decades larger than the controller poles, however, the testing performance of these poles are able to stabilise the system. Using the duality property, we can find the observer gains $\mathbf{J}$ required to achieve these closed loop poles using Ackermann's formula on the transposed matrices. These observer gains will be used to estimate the full state of the system with $\dot{\hat{x}}(t) = \mathbf{A_o}\hat{x}(t) + \mathbf{B_o}u(t) + \mathbf{J}\left(y(t) - \mathbf{C_o}\hat{x}(t)\right)$.

We will begin by implementing this Luenberger observer into the continuous integral controller, assuming the output states θ_m and θ_l are taken as ideal measurements into the observer. As seen in figure 5, the response of the continuous controller matches the ideal full state continuous controllers in figure 3. This is because the Luenberger observer converges to a zero-steady state error quickly using ideal sensor measurements as inputs. However, when we introduce the absolute encoder with 4096 counts per revolution and the 50fps vision sensor measurements, there is a non-zero steady-state error and a larger overshoot of 2.5°. This is due

to the noisy measurements from the sensors which cause imperfect state estimations. Overall, this continuous time controller and Luenberger observer pair satisfy the performance specifications.

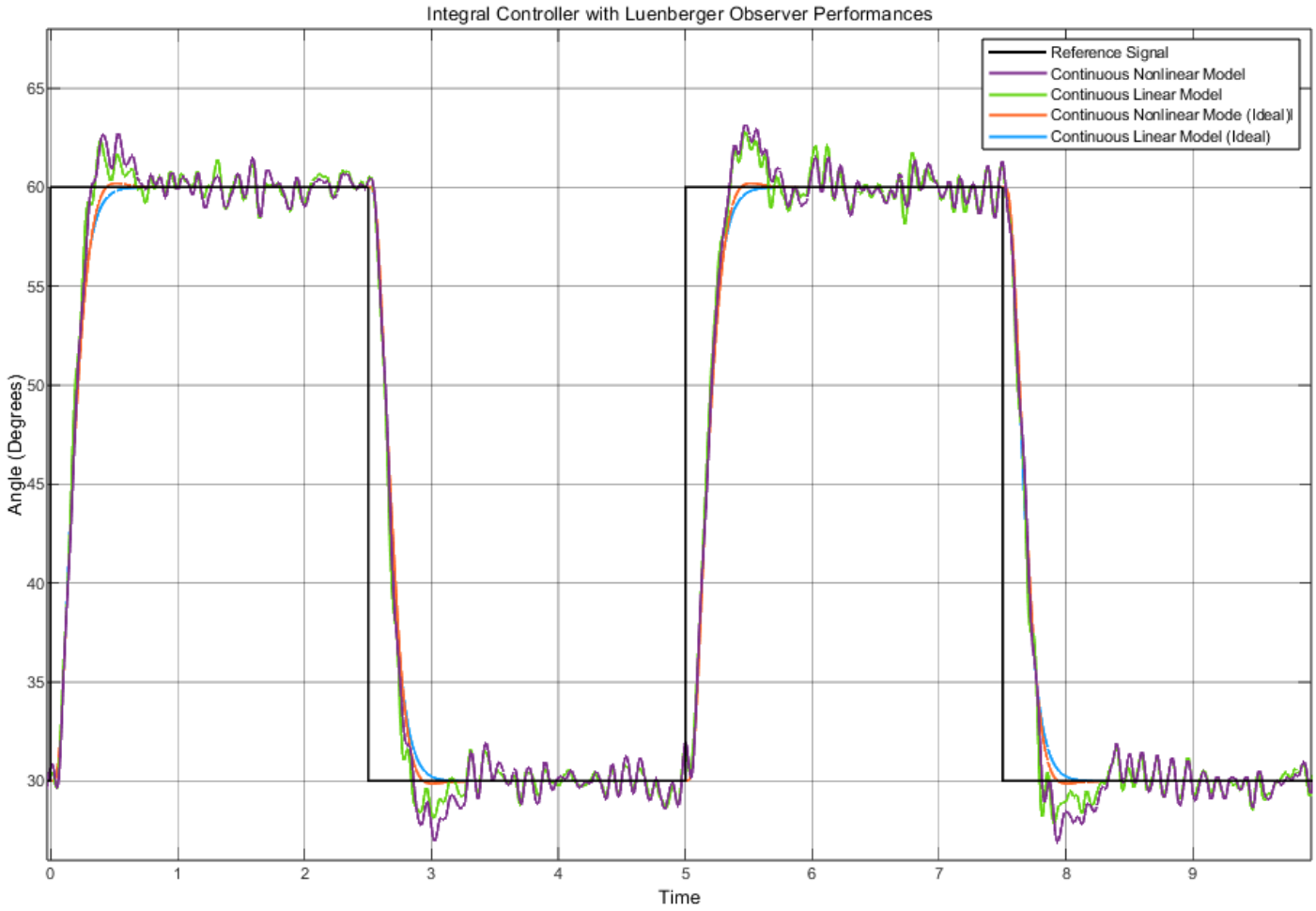


Figure 5: Performance of the linear and discrete controllers

Using emulation with zero order hold conversions before the observer and after the controller to simulate a discrete response, we obtain the response in figure 6. This emulation of the continuous observer and controller is unable to track the reference pulse signal and does not satisfy any of the performance specifications.

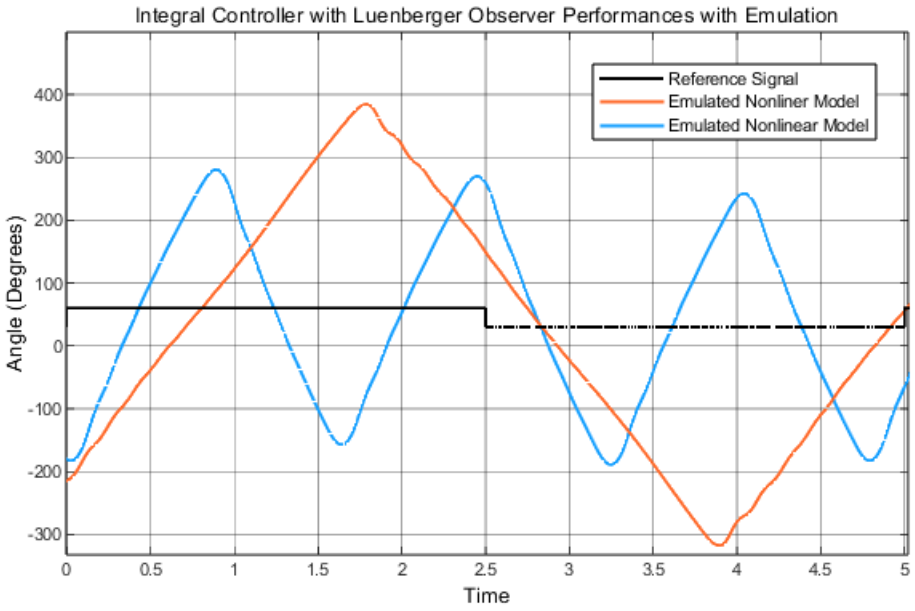
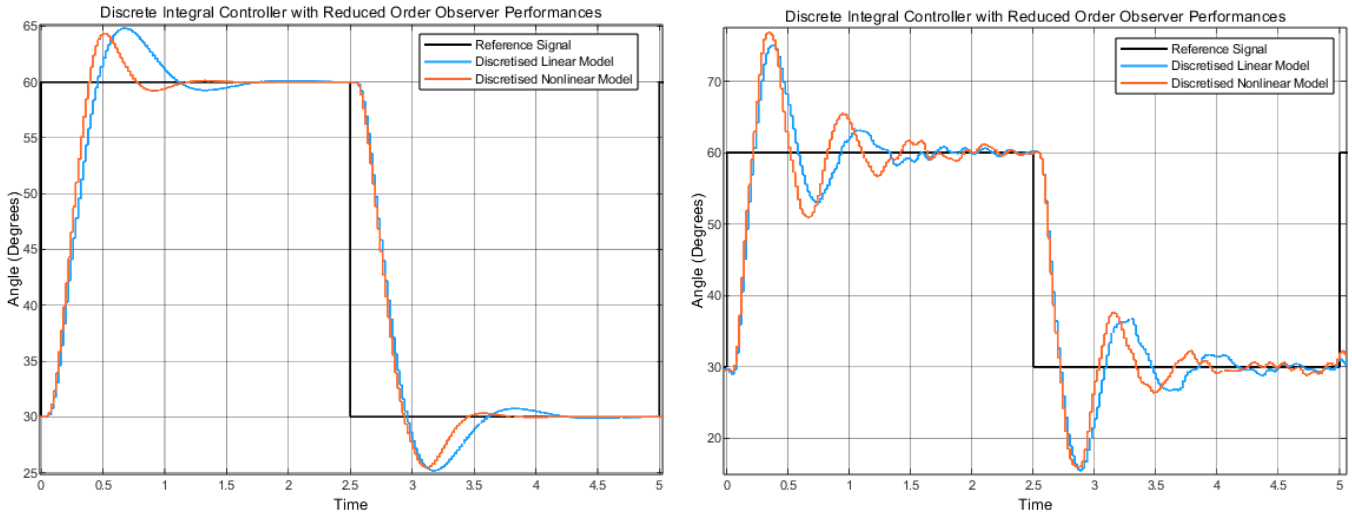


Figure 6: Performance of the linear and discrete controllers

4.2 Discrete Reduced Order Observer

Since we already have measurements of two states, we do not need to reconstruct the entire estimator but rather only estimate the other three unmeasured states. This will reduce the computational power for the controller as the observer has a lower order. This process involves separating the state matrix into measured states x_a and estimated states x_b, placing the poles of the observer for the estimated states. The poles are placed at s = [-700, -900, -1200] and discretised into z domain poles with the matched Z-transform. This discrete time reduced state observer will be paired with the discrete controller designed in section 3.2 to evaluate performance.

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}}_a(t) \\ \dot{\mathbf{x}}_b(t) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_{aa} & \mathbf{A}_{ab} \\ \mathbf{A}_{ba} & \mathbf{A}_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{x}_a(t) \\ \mathbf{x}_b(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_a \\ \mathbf{B}_b \end{bmatrix} u(t) \\ \mathbf{y}(t) &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_a(t) \\ \mathbf{x}_b(t) \end{bmatrix} \end{aligned} \tag{6}$$



(a) Performance using ideal sensor measurements

(b) Performance using physical sensor measurements

Figure 7: Performance of the Integral Controller with Reduced Order Observer

Although the system is able to meet performance specifications feeding the ideal sensor measurements into the reduced order observer with an overshoot just below 5° and a settling time of around 1.5s, using the physical sensor measurement increases this overshoot to around 15°.

5 Disturbance Rejection

For tracking of 60°, a constant torque disturbance of 0.05Nm will be added to the system after steady state is achieved. Since the choice of controller is an integral controller, the internal model principle means that the controller will be able to converge to steady state in the presence of this disturbance. Applying this step torque disturbance at 2 seconds to the continuous integral controller in figure 9, we see that there is an overshoot of 11°, however, the controller easily accounts for this disturbance and converges back to the 60° reference signal in approximately half a second.

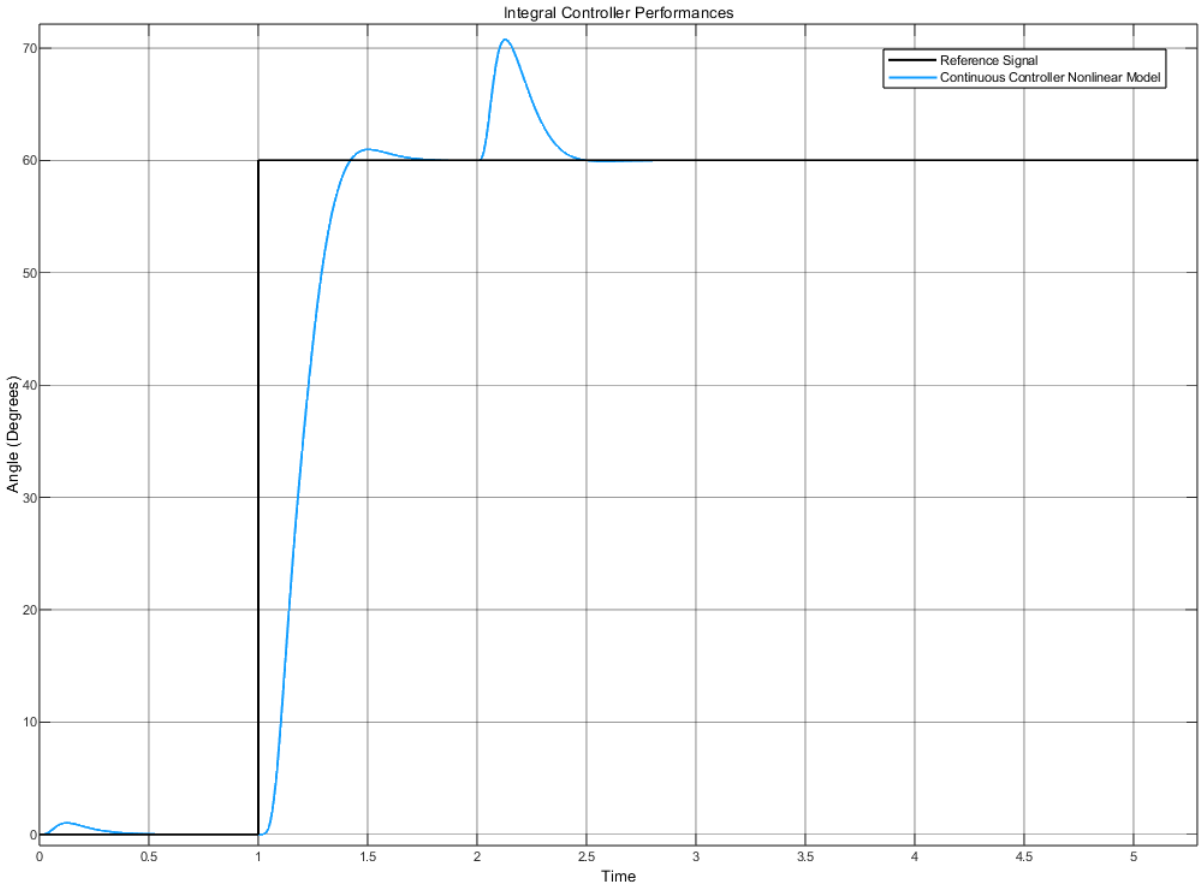


Figure 8: Performance of the linear and discrete controllers

6 Sinusoidal Tracking

In order to track sinusoidal signals, we need augment the state space model to include the sinusoidal dynamics to satisfy the internal model principle. For sinusoidal signals, the model must contain the frequency of the signal, which will allow the model to account for sinusoidal signals of that frequency of varying amplitudes and phase. To do this, we define the tracking frequency as $\omega = 2\frac{\pi}{4}$ and produce the model as:

$$\begin{aligned} A_d &= \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix} \\ C_d &= \begin{bmatrix} 1 & 0 \end{bmatrix} \end{aligned} \tag{7}$$

Following this, we produce the augmented model of the state space with:

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{x}_d(t) \end{bmatrix} &= \begin{bmatrix} A & BC_d \\ 0 & A_d \end{bmatrix} \begin{bmatrix} x(t) \\ x_d(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ x_d(t) \end{bmatrix} \end{aligned} \tag{8}$$

Finally, we can design the observer for the system states and disturbances as:

$$\begin{bmatrix} \dot{\hat{x}}(t) \\ \dot{\hat{x}}_d(t) \end{bmatrix} = A_{\text{aug}} \begin{bmatrix} \hat{x}(t) \\ \hat{x}_d(t) \end{bmatrix} + B_{\text{aug}}\, u(t) + J \left(y(t) - C_{aug} \begin{bmatrix} \hat{x}(t) \\ \hat{x}_d(t) \end{bmatrix} \right) \tag{9}$$

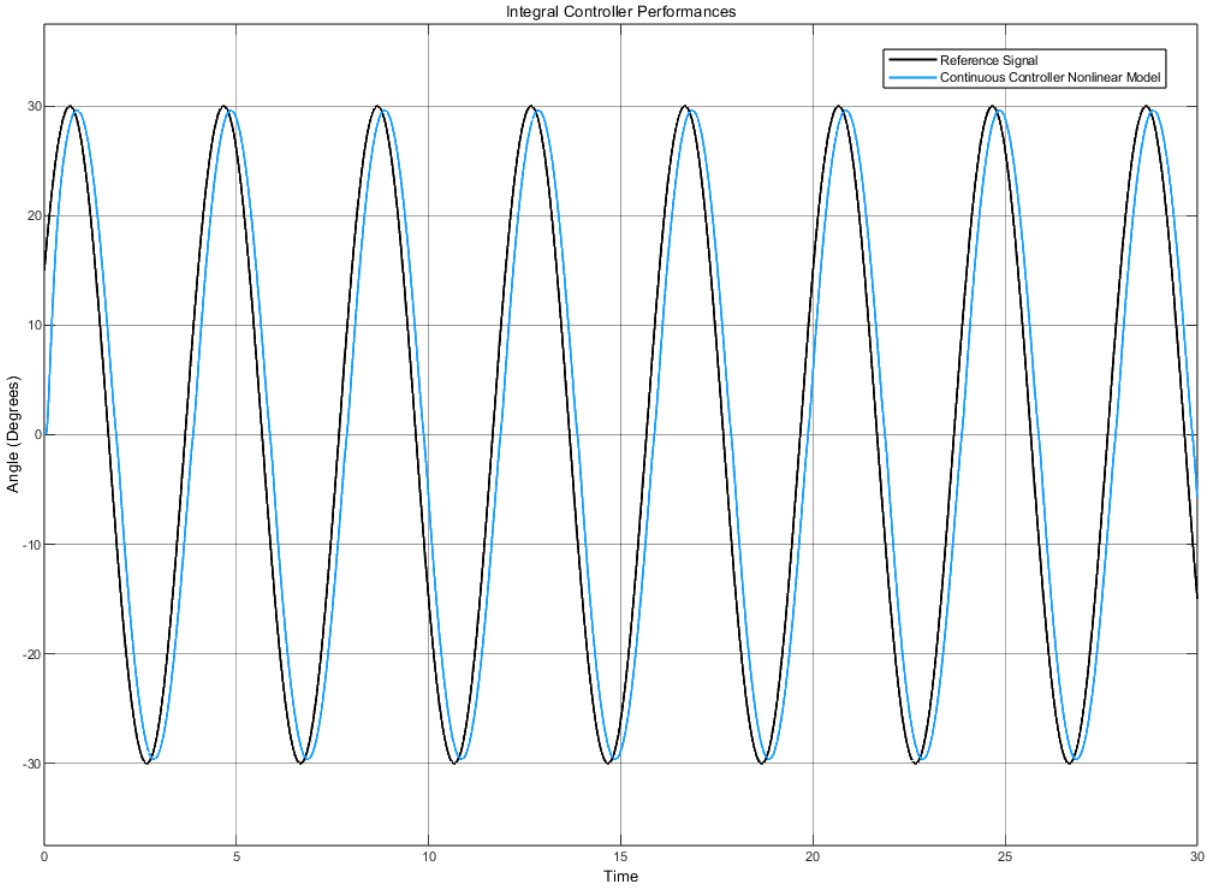


Figure 9: Tracking a sinusoidal signal with integral controller

The tracking performance of the sinusoidal signal using the integral control exhibits a constant phase offset between the reference signal and the link angle.

7 Conclusion

Using the separation principle for the completely observable and controllable system to design the integral controller first, followed by the observer allowed for simplicity in the design process using arbitrary pole placement techniques. With the integral controller design able to achieve the performance specifications with a 0.5° overshoot and a 0.7s settling time. Emulation techniques for the continuous time controllers and observers did not lead to good performance, with large oscillatory and unstable behaviour. Overall, the discretised observer and controller system is unable to meet the performance specifications. In order to meet these specifications, several adjustments can be attempted. Firstly, a different combination of sensors could reduce this large overshoot seen in figure 7b since the vision sensor contains a large amount of noise. Alternatively, a full state discrete Luenberger observer could be implemented instead of a reduced order observer as it will act as a filter for the measured states, however, this comes at the cost of increased computational power requirements.