Task Solution Report

Modeling of a Flexible Robot Joint

Ahmed Medhat Mohamed

June 1st, 2025

Table of Contents

I. Kinetic & Potential Energies of the System

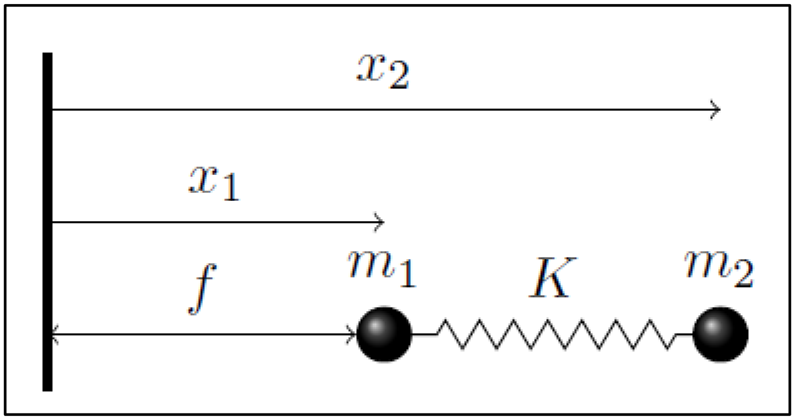

Figure 1. Abstract dynamics model of a flexible joint.

Assumptions

- Angular displacements are represented in terms of effective linear displacement via a constant radius r. Hence, the following holds:

$$x_1 = r\theta_1, x_2 = r\theta_2$$

- Similarly, rotational inertias are represented in terms of effective masses:

$$m_1 = \frac{J_1}{r^2}, m_2 = \frac{J_2}{r^2}$$

- Assume the motion in the joint is either linear or angular and restricted to small displacements such that linearization of the motion still holds.
- The elasticity between the motor and the link is linear.
- Force of the motor f is assumed to be zero.

Kinetic Energy T

$$T_1 = \frac{1}{2}m_1\dot{x}_1^2$$

$$T_2 = \frac{1}{2}m_2\dot{x}_2^2$$

$$T = \sum_{i=1}^{2} T_i = \frac{1}{2}m_1\dot{x}_1^2 + \frac{1}{2}m_2\dot{x}_2^2$$

Potential Energy V

The source of the potential energy in the system comes from the elastic linear spring. Hence:

$$F = -K\Delta x$$

The potential energy is the work done. Hence:

$$V = \int_0^{\Delta x} F(x)\,dx = \frac{1}{2}K(\Delta x)^2$$

$$\therefore V = \frac{1}{2}K(x_2 - x_1)^2$$

II. Equations of Motion using Lagrange

$$L = T - V$$

$$\therefore L = \frac{1}{2}m_1\dot{x}_1^2 + \frac{1}{2}m_2\dot{x}_2^2 - \frac{1}{2}K(x_2 - x_1)^2$$

Equations of Motion

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_1}\right) - \frac{\partial L}{\partial x_1} = 0$$

$$\left(\frac{\partial L}{\partial \dot{x}_1}\right) = m_1\dot{x}_1$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_1}\right) = m_1\ddot{x}_1 \longrightarrow (\mathbf{1})$$

$$\frac{\partial L}{\partial x_1} = K(x_2 - x_1) \;\longrightarrow (\mathbf{2})$$

From (1) & (2):

$$m_1\ddot{x}_1 - K(x_2 - x_1) = 0$$

Similarly, for x_2:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_2}\right) - \frac{\partial L}{\partial x_2} = 0$$

$$\left(\frac{\partial L}{\partial \dot{x}_2}\right) = m_2\dot{x}_2$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}_2}\right) = m_2\ddot{x}_2 \dashrightarrow (\mathbf{3})$$

$$\frac{\partial L}{\partial x_2} = -K(x_2 - x_1) \dashrightarrow (\mathbf{4})$$

From (3) & (4):

$$m_2\ddot{x}_2 + K(x_2 - x_1) = 0$$

Hence, equations of motion are as follows:

$$m_1\ddot{x}_1 - K(x_2 - x_1) = 0$$

$$m_2\ddot{x}_2 + K(x_2 - x_1) = 0$$

III. Simulink Model

Equations Preparation

It's required to add both motor-side viscous friction and link-side viscous friction. Hence, the updated equations are as follows, assuming b_1 and b_2 are the damping coefficients for the motor and link, respectively:

$$m_1\ddot{x}_1 + b_1\dot{x}_1 - K(x_2 - x_1) = 0$$

$$m_2\ddot{x}_2 + b_2\dot{x}_2 + K(x_2 - x_1) = 0$$

The values given are as follows:

- $m_1 = 0.6\ kg$
- $m_2 = 1\ kg$
- $K = 350\frac{N}{m}$

Damping Coefficients Assumptions

1. Critical Damping

Recall the damping ratio formula, where $\zeta < 1$ refers to an underdamped system, $\zeta = 1$ is critically damped, and $\zeta > 1$ is overdamped.

$$\zeta = \frac{b}{2\sqrt{Km}}$$

To design a smooth response that is free of oscillations, the damping coefficients must be chosen such that ζ equals 1. Hence, this leads to the following calculations:

$$b_1 = 2\zeta\sqrt{Km} = 2\sqrt{350 * 0.6} \approx \mathbf{28.98}\,\frac{\boldsymbol{Ns}}{\boldsymbol{m}}$$

$$b_2 = 2\zeta\sqrt{Km} = 2\sqrt{350 * 1} \approx \mathbf{37.42}\,\frac{\boldsymbol{Ns}}{\boldsymbol{m}}$$

This would lead to the perfect response without overshooting to dampen any disturbances. However, this is unrealistic in most real-world cases for mechanical systems, as noted by Nise (2011) and Ogata (2010), who explain that damping due to friction is typically small compared to the values of both stiffness and inertia. Hence, an acceptable trade-off is to choose the damping ratio to be slightly underdamped, thereby achieving tolerable oscillations that are later compensated in the control system part.

2. Underdamped Case:

In this section, an underdamped case is considered to provide values for the damping coefficients for both the motor-side and link-side with $\zeta = 0.2, 0.4, 0.6, 0.8$. Comparison between the critically damped and underdamped cases is then simulated in Simulink to test performance.

Simulink Model Setup

Recalling EOM:

$$m_1\ddot{x}_1 + b_1\dot{x}_1 - K(x_2 - x_1) = 0$$

$$\therefore\ \boldsymbol{\ddot{x}_1} = \frac{\boldsymbol{-b_1\dot{x}_1 + K(x_2 - x_1)}}{\boldsymbol{m_1}}$$

$$m_2\ddot{x}_2 + b_2\dot{x}_2 + K(x_2 - x_1) = 0$$

$$\therefore\ \boldsymbol{\ddot{x}_2} = \frac{\boldsymbol{-b_2\dot{x}_2 - K(x_2 - x_1)}}{\boldsymbol{m_2}}$$

The system is at equilibrium when its states x_1 and x_2 are at rest, meaning at zero initial position along with zero velocities and accelerations. Hence, the case tested here is with an initial perturbation in $x_1(0) = 5\ m$, while initial velocities are kept at zero. This initial perturbation in x_1 initial condition will excite the system such that the response could be measured and analyzed.

IV. Simulation Results

The model was simulated with different zeta models, and the corresponding statistics were computed, settling time, % overshoot, and final settling value, yielding the following:

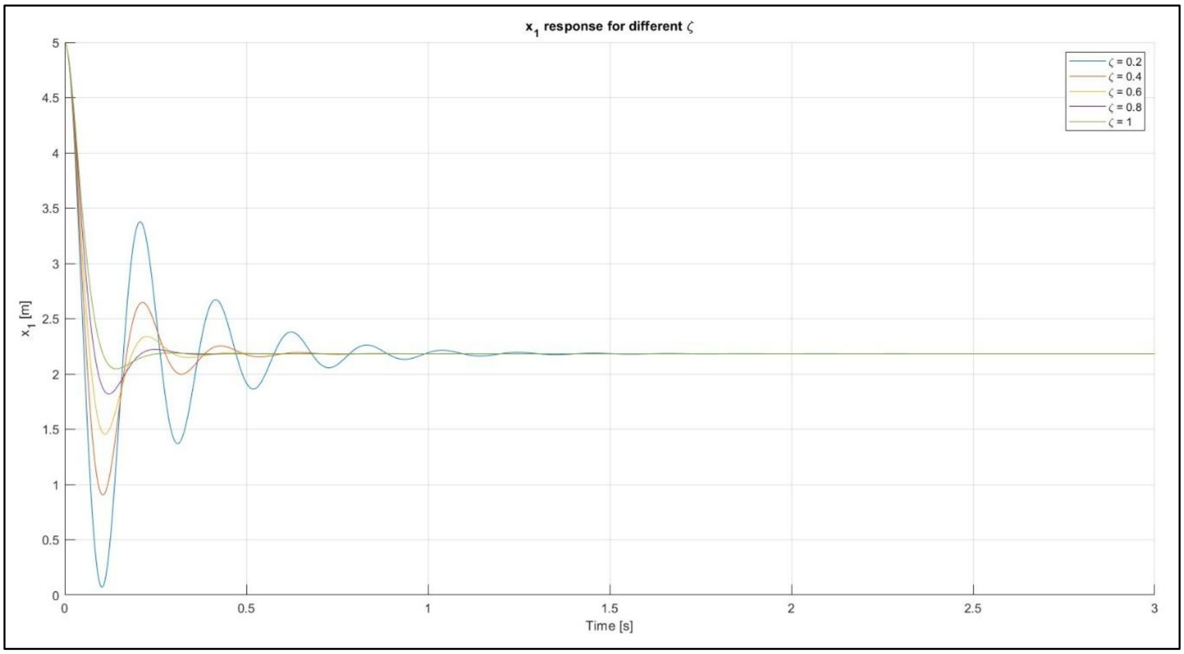

Figure 2. x_1 response for different ζ values.

Comment

As expected, the value of $\zeta = 1$, corresponding to critical damping yielded the smoothest response with the least %undershoot of 6.2287% with no oscillations. In addition, it's visible that this value increases with decreasing zeta values, with the signal exhibiting transient behavior and oscillations around the settling value, hence taking more time to reach the steady state.

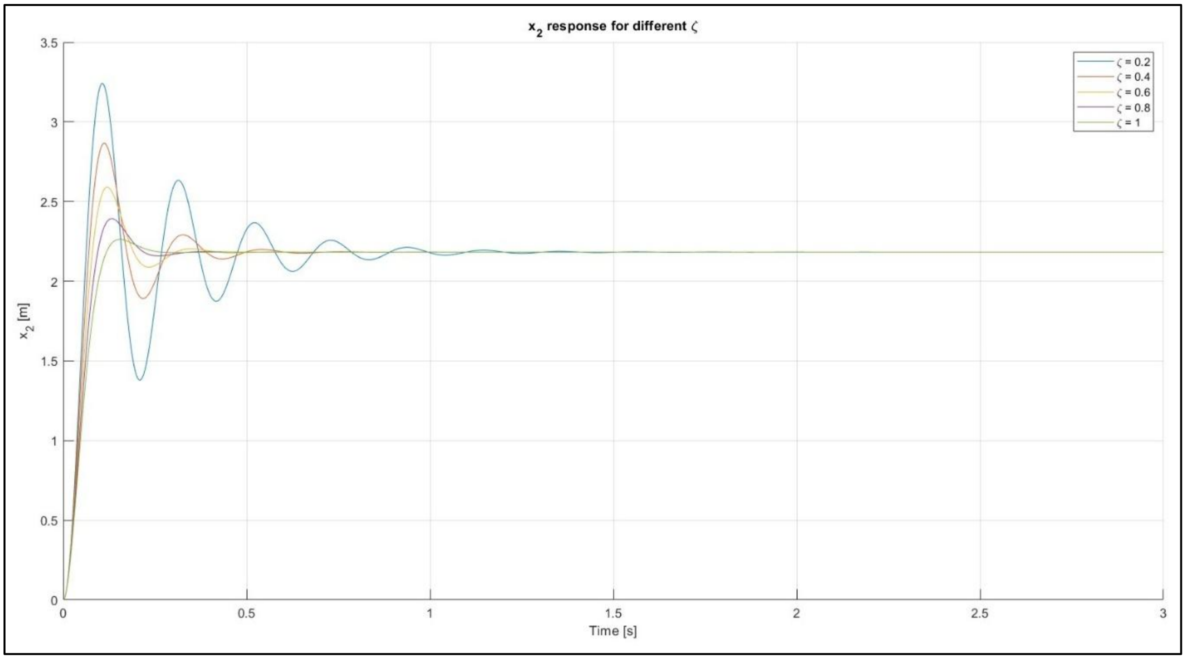

Figure 3. x_2 response for different ζ values.

Comment

The same behavior is observed with the response of x_2. As shown in the figure, the response varies from smooth with no oscillations at the critical damping value to oscillatory behavior with decreasing damping ratios. The critical damping case scored 3.6759% overshoot over the steady state value.

The complete set of values is depicted in Table 1 below.

Zeta	FinalValue_x1	Undershoot_x1	SettlingTime_x1	FinalValue_x2	Overshoot_x2	SettlingTime_x2
0.2	2.1825	96.731	0.95327	2.1825	48.52	0.84827
0.4	2.1825	58.542	0.46227	2.1825	31.311	0.36927
0.6	2.1825	33.376	0.28027	2.1825	18.674	0.27727
0.8	2.1825	16.695	0.19627	2.1825	9.6407	0.19827
1	2.1825	6.2287	0.20527	2.1825	3.6759	0.19927

Table 1. Complete Simulation Statistics for x_1 and x_2.

V. Plausibility Check

One possible plausibility check is to monitor the energy of the system, where the total energy of the system should dissipate over time due to the damping present in both the motor-side and link-side. This could be modeled as follows by recalling the total energy equation constructed earlier:

$$E_{total} = T + V$$

$$\therefore E_{total} = \frac{1}{2}m_1\dot{x}_1^2 + \frac{1}{2}m_2\dot{x}_2^2 + \frac{1}{2}K(x_2 - x_1)^2$$

This equation is then modeled in Simulink, yielding the following graph:

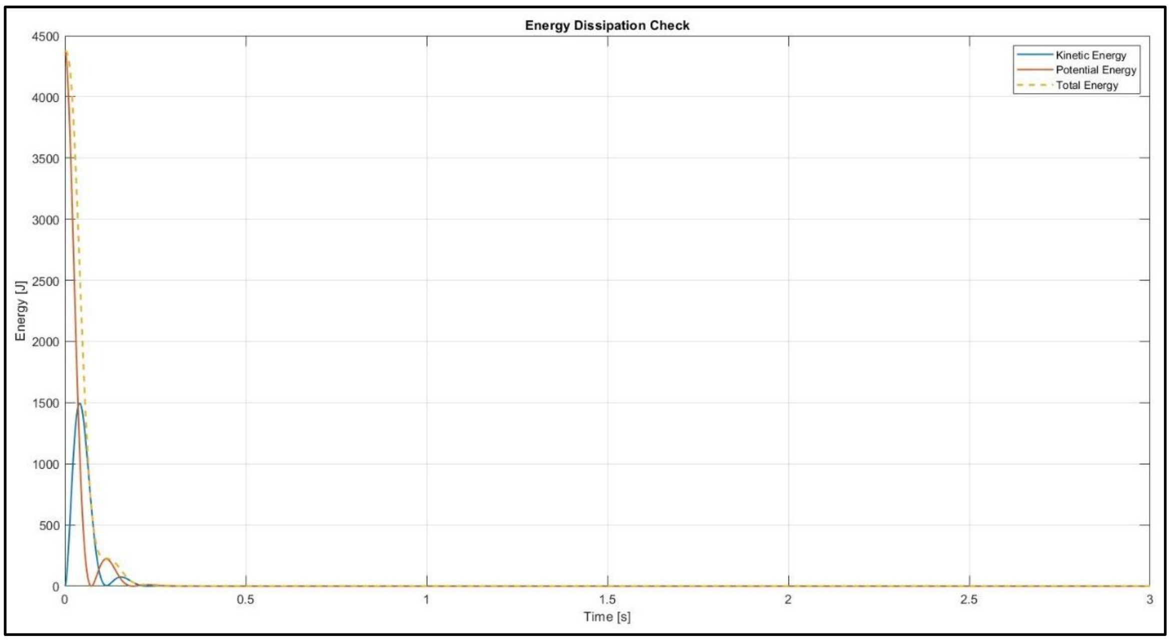

Figure 4. Energy Dissipation Plausibility Check for Zeta = 0.6 Case.

Comment

The graph shown above demonstrates the energy dissipation for the case when zeta = 0.6. The choice of this case is because the solution with zeta = 0.6 exhibits moderate oscillations, which can demonstrate the energy oscillations better for visualization. As seen in the graph, the potential energy starts at its highest at the beginning due to the initial perturbation in x_2. Moreover, the kinetic energy starts at zero and exhibits decreasing oscillations along with the potential energy graph, both coming to zero again by the end of the simulation (total energy). This, in turn, verifies the correctness of the Simulink model.

VI. Extra Part – Control

In this section, the assumption that the motor force f is zero is removed, allowing for controlling the flexible robot joint. For this part, the previous model is used with the damping parameters corresponding to $\zeta = 0.6$. This is done such that the model response still exhibits oscillations until the steady state convergence is achieved. Hence, the efficiency of the control system can be tested in terms of its ability to smooth the response and accelerate convergence. As for the control part, two methods of control are performed and compared with each other: (i) PD Controller, (ii) State Feedback with LQR.

(i) PD Controller

The updated equations of motion are now as follows, with f added to the motor equation as an external acting force.

$$m_1\ddot{x}_1 + b_1\dot{x}_1 - K(x_2 - x_1) = f$$

$$m_2\ddot{x}_2 + b_2\dot{x}_2 + K(x_2 - x_1) = 0$$

Simulink Equations Setup

$$\boldsymbol{\ddot{x}_1 = \frac{f - b_1\dot{x}_1 + K(x_2 - x_1)}{m_1}}$$

$$\boldsymbol{\ddot{x}_2 = \frac{-b_2\dot{x}_2 - K(x_2 - x_1)}{m_2}}$$

Define the error signal e:

$$e = x_{ref} - x_1$$

Control Law:

$$f = K_p * e - K_d * \dot{x}_1$$

Simulation

To test the controller, x_{ref} is set to 7 m so that the controller should move the position to 7 after starting initially at 5 m.

Results

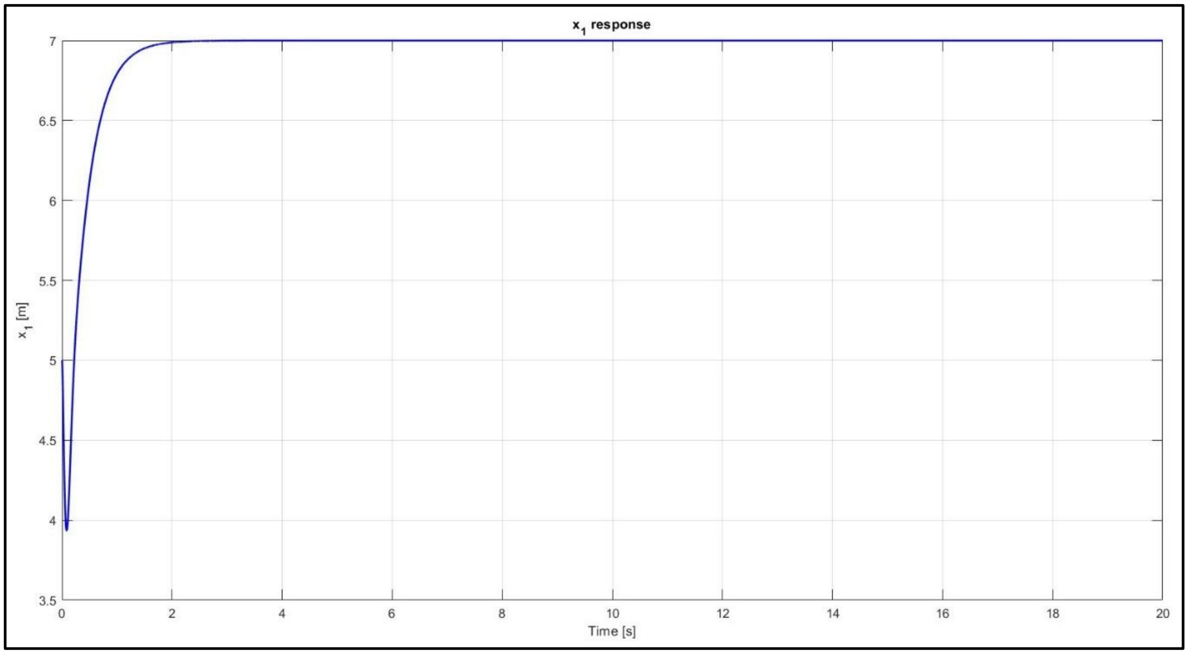

Figure 5. x_1 PD Control Response.

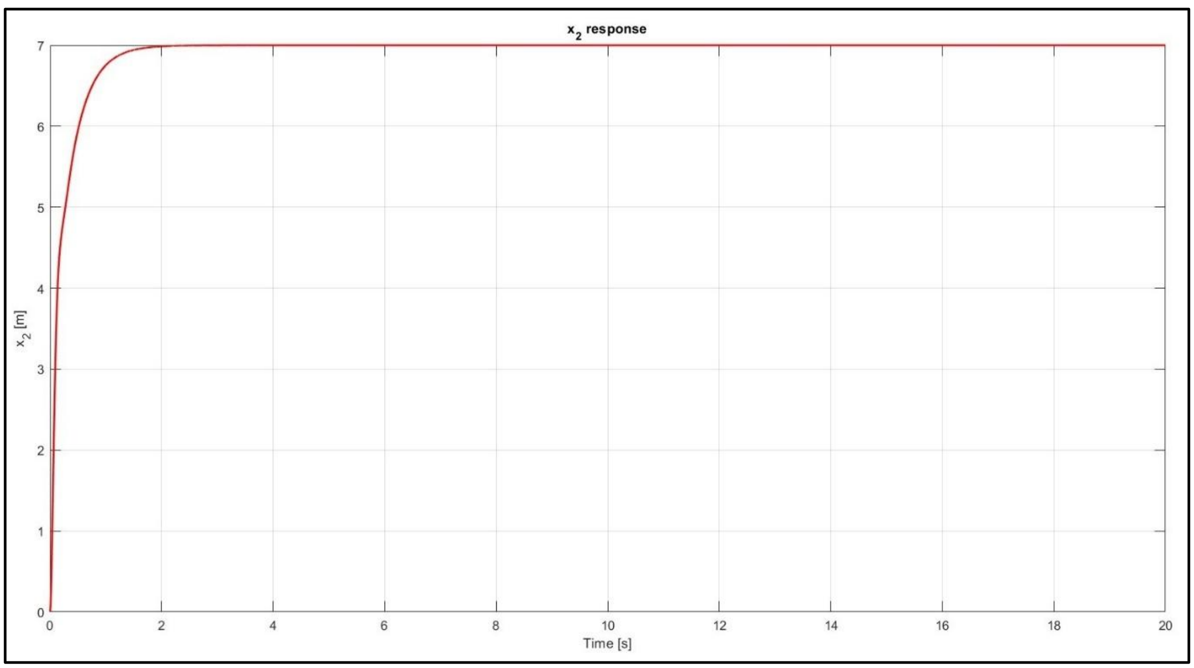

Figure 6. x_2 PD Control Response.

Zeta	FinalValue_x1	Undershoot_x1	SettlingTime_x1	FinalValue_x2	Overshoot_x2	SettlingTime_x2
0.6	7	43.814	1.1363	7	0	1.1963

Table 2. Complete Simulation Statistics for x_1 and x_2 (PD Control with no Feedforward).

Comment

The settling time for both is around 1 second, and the response is smooth, but for x_1, the response experiences a huge undershoot of 43.8% before rising and settling again. This problem is caused by the spring's initial stretching with $x_1(0) = 5m$. Hence, the spring exerts a restoring force on x_1 causing it to initially drop before the controller eventually reacts and recovers it.

Feedforward Addition

Problem Solution

To fix this issue, a feedforward part is added to the control law to compensate for the spring effect as follows:

$$f = K_p * e - K_d * \dot{x}_1 - K(x_2 - x_1)$$

Results

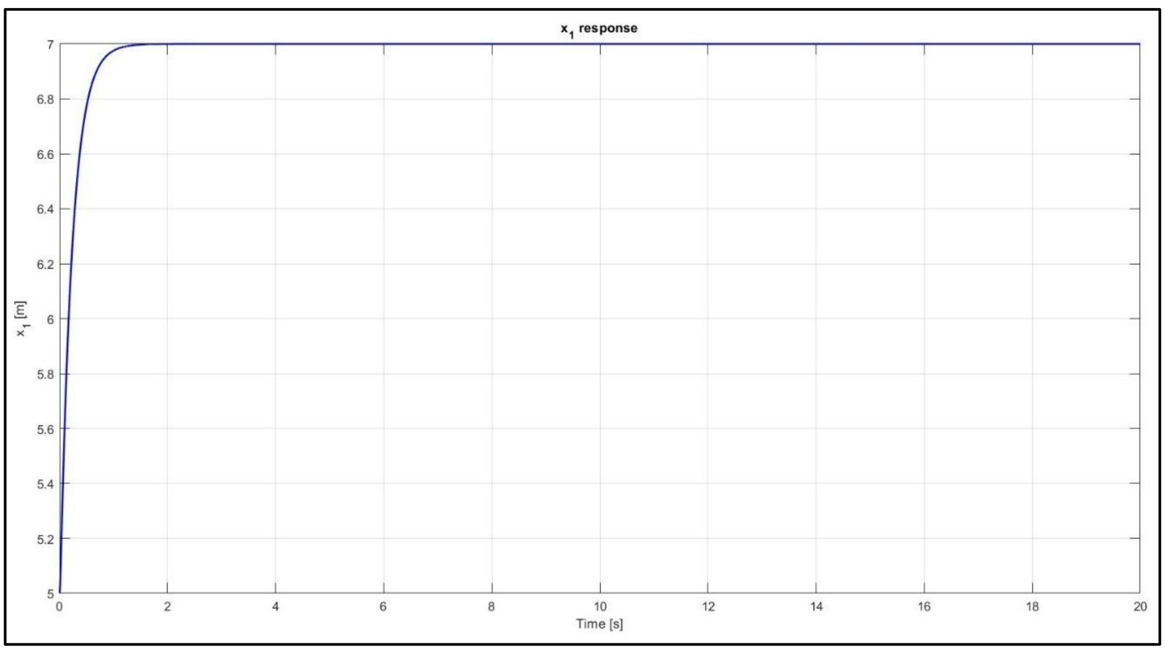

Figure 7. x_1 PD Control Response with Feedforward.

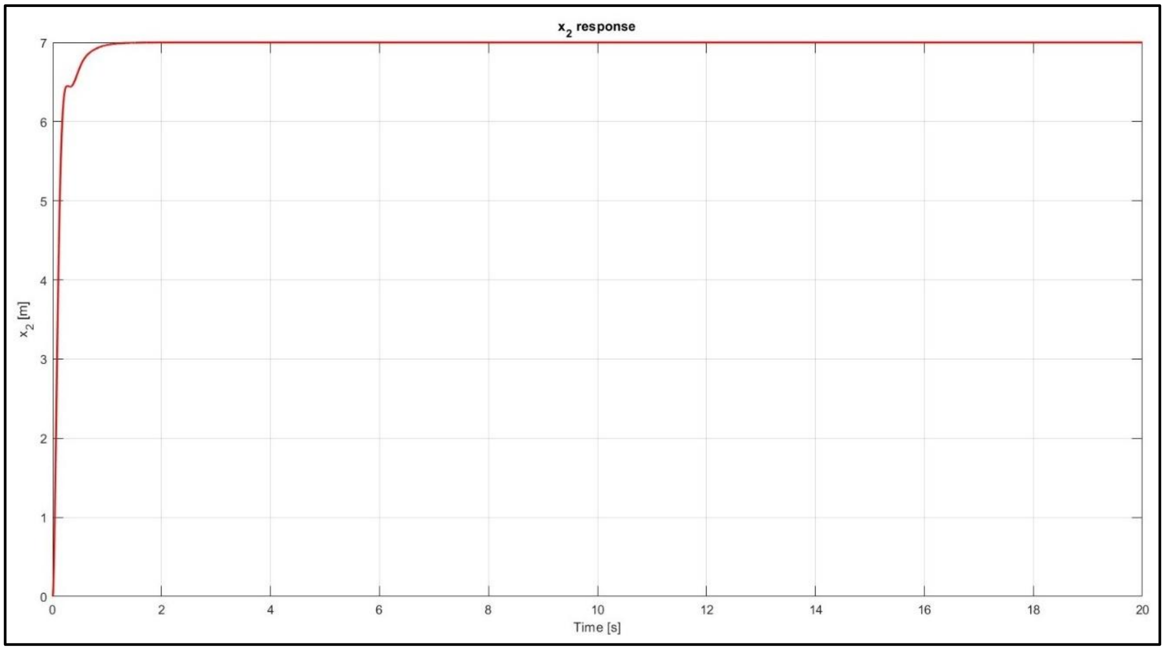

Figure 8. x_2 PD Control Response with Feedforward.

Comment

As shown in the plots above, the undershoot problem was fixed by adding the feedforward part, resulting in a smooth, quick response with zero overshoot and a short settling time of 0.6 s.

(ii) State Feedback with LQR Control

In this section, another control method is implemented to control the robotic joint. Recalling equations of motion once more:

$$\ddot{x}_1 = \frac{f - b_1\dot{x}_1 + K(x_2 - x_1)}{m_1}$$

$$\ddot{x}_2 = \frac{-b_2\dot{x}_2 - K(x_2 - x_1)}{m_2}$$

State Space Representation

$$\dot{X} = AX + Bf$$

$$\therefore \dot{X} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\frac{K}{m_1} & -\frac{b_1}{m_1} & \frac{K}{m_1} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K}{m_2} & 0 & -\frac{K}{m_2} & -\frac{b_2}{m_2} \end{pmatrix} X + \begin{pmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{pmatrix} f$$

Controllability Check

Matrix A does not have full rank. Hence, we cannot control the full system states. Therefore, we reduce the controller to control the state x_1 only and add a spring compensation instead.

Reduced State Space Model

$$X = \begin{bmatrix} x_1 \\ \dot{x}_1 \end{bmatrix}$$

$$\dot{X} = \begin{bmatrix} \dot{x}_1 \\ \ddot{x}_1 \end{bmatrix}$$

$$\dot{X} = AX + Bf$$

$$\therefore \dot{X} = \begin{pmatrix} 0 & 1 \\ -\frac{K}{m_1} & -\frac{b_1}{m_1} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{m_1} \end{pmatrix} f$$

Control Law

$$f = -K_x X - \frac{K}{m_1}(x_2 - x_1) + k_r x_{ref}$$

Reference Tracking Part

$$C = [1 \quad 0]$$

$$k_r = \frac{-1}{C(A - BK)^{-1}B}$$

Upon implementing k_r to perform reference tracking, it did not work properly, possibly because the system is weakly controllable and very lightly damped. Hence, this caused the system to respond weakly to the input (since B is small).

Alternative Approach

k_r is removed, and instead of computing the control law based on the system state vector, it will rely on the state error instead.

$$f = -K_x \begin{bmatrix} e \\ \dot{x}_1 \end{bmatrix} - \frac{K}{m_1}(x_2 - x_1)$$

$$where, e = x_1 - x_{ref}$$

Results

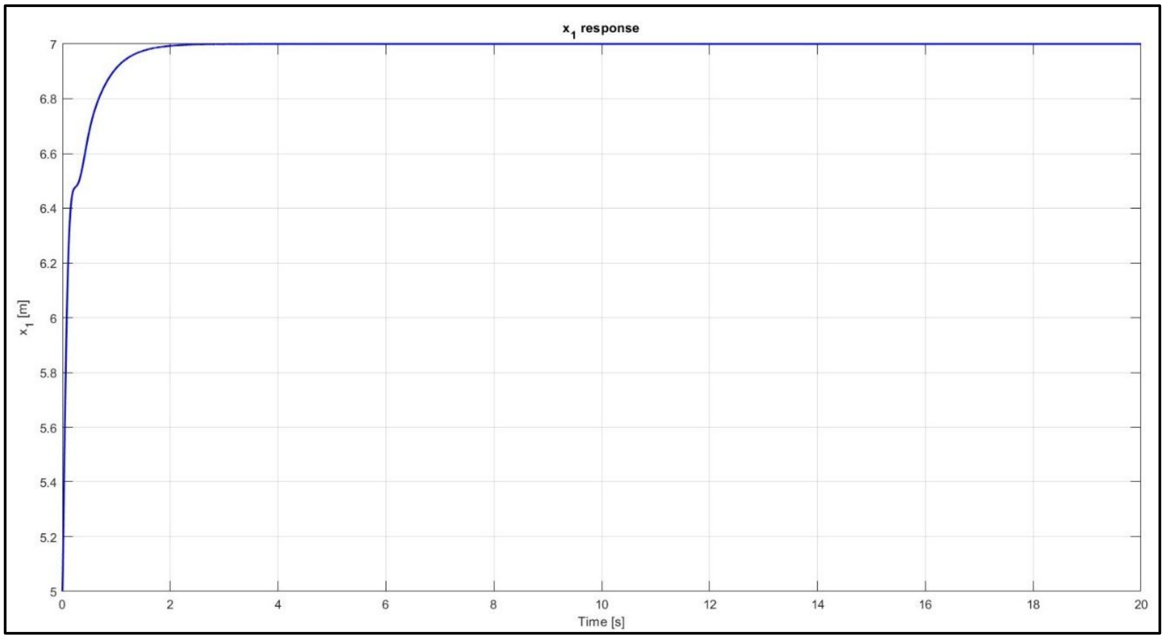

Figure 9. x_1 State Feedback Response with LQR.

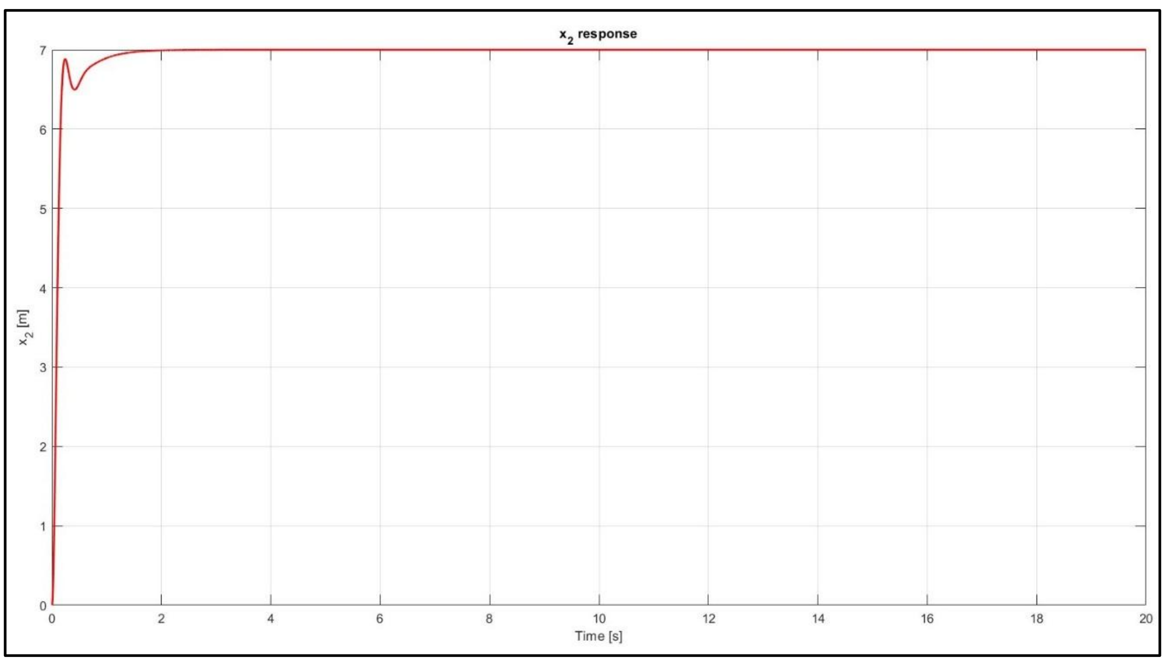

Figure 10. x_2 State Feedback Response with LQR.

Comment

The response for both x_1 and x_2 achieved a steady state value of 7 m within only 0.6 sec with minimal to no overshoot/undershoot, confirming the robustness of LQR control.

VII. References

Nise, N. S. (2011). *Control systems engineering* (6th ed.). Wiley.

Ogata, K. (2010). *Modern control engineering* (5th ed.). Prentice Hall.