TUM SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY (CIT)

TECHNICAL UNIVERSITY OF MUNICH

Report Submitted to Robot Programming and Control for Human Interaction (IN2308)

Final Report

Jesus Arturo Sol Navarro

TECHNICAL UNIVERSITY OF MUNICH

Report Submitted to Robot Programming and Control for Human Interaction (IN2308)

Final Report

Author: Jesus Arturo Sol Navarro
Submission Date: January 6, 2025

I confirm that this report is my own work and i have documented all sources and material used.

Munich, January 6, 2025
Jesus Arturo Sol Navarro

Contents

1 Introduction

This report provides an analysis of the dynamics and control of a 3-degree-of-freedom (3DOF) planar manipulator.

The first section covers the theoretical foundations, including kinematics, dynamics, joint control, Cartesian impedance control, collision detection strategies and Null-space Optimization.

In the subsequent section, various simulations are presented to validate the theoretical concepts. These include tasks such as implementing PD controllers, tuning stiffness and damping parameters, and simulating system behavior under different configurations. All simulations were conducted using MATLAB Simulink.

2 Theory

2.1 Kinematics and Dynamics

2.2 Joint Control

2.2.1 Problem 1

The simplest way to control the configuration of a manipulator is to control directly the joint positions. Given a vector of desired joint angles $\mathbf{q}_d \in \mathbb{R}^n$, the error between the current and desired manipulator configuration is:

$$\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q}$$

Considering only the case of regulation ($\dot{\mathbf{q}}_d = 0$), write the law of the PD controller for reaching the desired joint positions $\mathbf{q}_d$.

$$\boldsymbol{\tau} = -\mathbf{K}_p(\mathbf{q} - \mathbf{q}_d) - \mathbf{K}_d\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q})$$

2.2.2 Problem 2

Problem: Using the control law derived in Problem 2.2.1, write the equation of the dynamics of the controlled system.

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}$$

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + (\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{K}_d)\dot{\mathbf{q}} + \mathbf{K}_p(\mathbf{q} - \mathbf{q}_d) = 0$$

2.2.3 Problem 3

Neglecting the centrifugal and Coriolis forces and assuming a quasi-stationary variation, the controlled dynamics can be written as:

$$M(\mathbf{q}_0)\ddot{\tilde{\mathbf{q}}} + K_d\dot{\tilde{\mathbf{q}}} + K_p\tilde{\mathbf{q}} = 0$$

Problem: Considering a constant stiffness matrix $\mathbf{K}_p$, derive a damping matrix $\mathbf{K}_d$ for achieving a given damping factor ζ of the system .

Neglecting centrifugal and Coriolis forces and comparing to a second-order linear system in standard form ($\ddot{\mathbf{q}} + 2\zeta\omega_n\dot{\mathbf{q}} + \omega_n^2\mathbf{q} = 0$).

$$\ddot{\tilde{\mathbf{q}}} + \mathbf{M}(\mathbf{q_0})^{-1}\mathbf{K}_d\dot{\tilde{\mathbf{q}}} + \mathbf{M}(\mathbf{q_0})^{-1}\mathbf{K}_p\tilde{\mathbf{q}} = 0$$

$$\omega_n = \sqrt{\mathbf{M}(\mathbf{q_0})^{-1}\mathbf{K}_p}$$
$$\mathbf{K}_d = 2\zeta\mathbf{M}(\mathbf{q_0})\sqrt{\mathbf{M}(\mathbf{q_0})^{-1}\mathbf{K}_p}$$

2.3 Translational Cartesian Impedance Control

2.3.1 Problem 1

Problem: The mapping from joint to Cartesian velocity space is given by $\dot{x} = J(q)\dot{q}$. Derive the mapping between Cartesian forces and joint torques.

To derive the mapping between Cartesian forces and joint torques, we can use the concept of power, which states that the total power in a system is conserved.Thus:

$$P_{\text{Cartesian}} = P_{\text{Joint}}$$
$$\dot{x}^T F = \dot{q}^T \tau.$$

Inserting $\dot{x} = J(q)\dot{q}$.

$$\tau = J^T F.$$

2.3.2 Problem 2

Problem: Considering a translational error $\tilde{x} = x - x_d$ and a constant, positive definite, symmetric stiffness matrix $K_t \in \mathbb{R}^2$, write the law of a spring force F_K between the TCP of the manipulator and the desired point.

$$F_K = -K_t(x - x_d)$$

2.3.3 Problem 3

Problem: Calculate the planar Cartesian force F_D resulting from a viscous damper with the constant symmetric damping matrix $D_t \in \mathbb{R}^2$.

$$F_D = -D_t(\dot{x} - \dot{x}_d)$$

2.3.4 Problem 4

Problem: The Cartesian impedance control law for the translational case can now be set up as the superposition of the solutions of the Problems 2.3.2 and 2.3.3. The transformation from Cartesian forces to joint torques was derived in Problem 2.3.1. Write the complete control law for the translational Cartesian impedance controller such that it can be commanded to the joint actuators.

$$\tau = J^T \left(F_K + F_D\right)$$
$$\tau = J^T \left(-K_t(x - x_d) - D_t\dot{x}\right)$$

2.4 Collision Detection

2.4.1 Problem 1

Problem: Draw a scheme for estimating only the external torques.

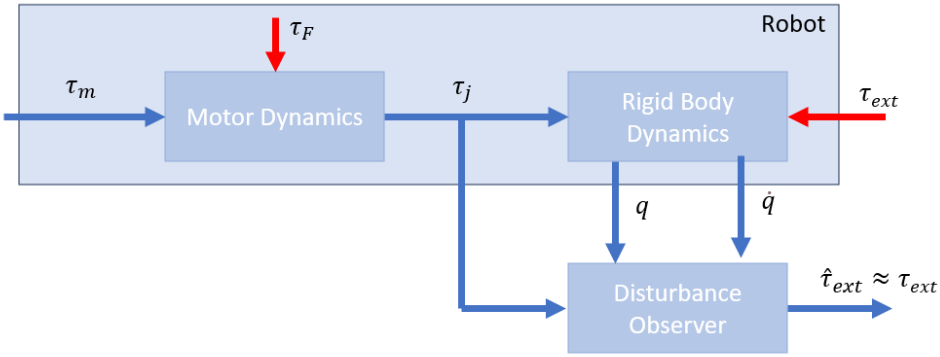

Figure 2.1

2.4.2 Problem 2

The effective torque τ_e can be deduced either from the measurements or from the desired trajectory using the dynamic model of the manipulator:

$$\tau_e = \mathbf{M}(\mathbf{q}_n)\dot{\mathbf{q}}_n + \mathbf{C}(\mathbf{q}_n, \dot{\mathbf{q}}_n)\dot{\mathbf{q}}_n + \mathbf{g}(\mathbf{q}_n) \quad \text{from measurements}$$

$$\tau_e = \mathbf{M}(\mathbf{q}_d)\dot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)\dot{\mathbf{q}}_d + \mathbf{g}(\mathbf{q}_d) \quad \text{from desired trajectory}$$

Problem: Explain the disadvantages of both approaches.

- From Measurements

  This approach does not account for the controller dynamics. As a result, the estimated torques may not accurately reflect the real interaction forces, leading to potential errors in collision detection.

- From Desired Trajectory

  The calculation relies heavily on the acceleration signal, which is often noisy and prone to inaccuracies. This noise will lead to incorrect interpretations of the system's behavior and external disturbances.

2.4.3 Problem 3

A better approach is to use the momentum instead of directly the torques. The momentum can be estimated by comparing the expected momentum caused by the commanded torques and the effective momentum of the system (obtained from the measurement of the state of the

system). The generalized momentum of the manipulator is given by: $\mathbf{p} = \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}$. Considering the manipulator dynamics $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}$, derive the equation of the variation of the momentum.

Problem: Considering the manipulator dynamics in Eq.(4), derive the equation of the variation of the momentum.

Derivative w.r.t.

$$\dot{\mathbf{p}} = \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}$$

From manipulator dynamics.

$$\dot{\mathbf{p}} = \dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} + \boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q})$$

Using standard robotic's property $\dot{\mathbf{M}} = \mathbf{C} + \mathbf{C}^\top$.

$$\dot{\mathbf{p}} = \mathbf{C}^\top(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \boldsymbol{\tau} - \mathbf{g}(\mathbf{q})$$

2.4.4 Problem 4

Problem: Draw the scheme of a collision estimator based on the momentum.

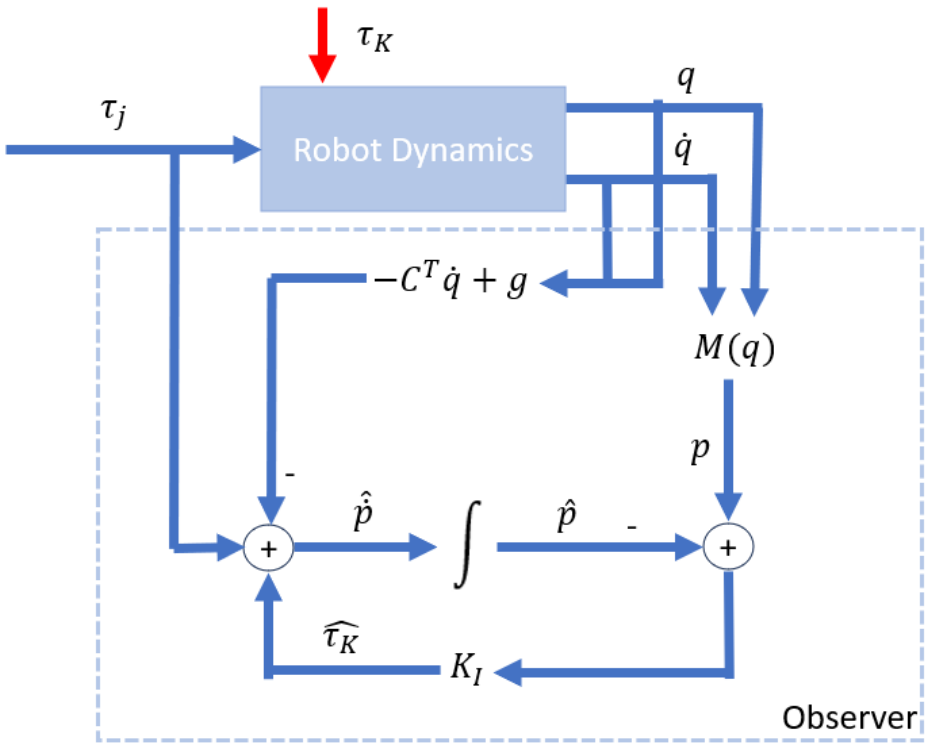

Figure 2.2

2.4.5 Problem 5

Problem: Derive the equation of the dynamics of the estimator.

From Figure 2.2.

$$\hat{\tau}_k(t) = K_I \left[p(t) - \int_0^t \left(\tau + \hat{\tau}_k + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) \right) dt - p(0) \right]$$

Derivative w.r.t.

$$\dot{\hat{\tau}}_k(t) = K_I \left[\dot{p}(t) - \left(\tau + \hat{\tau}_k + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}) \right) \right]$$

Inserting $\dot{p} = C^\top(q, \dot{q})\dot{q} + \tau + \tau_k - g(q)$, the decoupled, first order dynamics of the estimator are found.

$$\dot{\hat{\tau}}_k(t) + K_I \hat{\tau}_k = K_I \tau_k$$

2.4.6 Problem 6

Which reaction strategies can be adopted in case of a collision?

- Stop Motion: Immediately halt all joint movements .
- Low Stiffness and High Damping: Reduce stiffness to make the robot compliant .
- Retreat : Actively move the robot away from the collision source.
- Force Limitation: Limit the applied forces or torques to safe thresholds.
- Gravity Compensation.
- Trajectory Modification: Slow down or reverse the trajectory by adjusting the interpolator in position control mode to avoid further contact.

2.5 (NO) Nullspace Optimizations

2.5.1 Problem 1

In **Problem 2.3.4**, a control law was treated which determines the translational Cartesian forces, i.e., the task space has $m = 2$ DOF. Since the considered manipulator has $n = 3$ joints, the configuration (joint angles) could change while the end-effector is in a fixed position. The resulting motion is called a *nullspace motion*. In order to control the nullspace motion, a control law for the $r = n - m = 1$ redundant DOF will be derived. The complete impedance controller can be obtained by summing up the TCP impedance torque $\boldsymbol{\tau}_i$ derived in Eq.(2.3.4) and a nullspace torque $\boldsymbol{\tau}_n$:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_i + \boldsymbol{\tau}_n$$

Problem: Write a control law τ_n for the nullspace and demonstrate that it does not interfere with the TCP torque.

Defining the control law $\tau_n = N\tau_{\text{orig}}$ with a nullspace projection matrix $N = I - J^T \left(JM^{-1}J^T \right)^{-1} JM^{-1}$.

$$\ddot{x} = JM^{-1} \left(I - J^T \left(JM^{-1}J^T \right)^{-1} JM^{-1} \right) \tau_{\text{orig}}$$

$$\ddot{x} = \left(JM^{-1} - JM^{-1}J^T \left(JM^{-1}J^T \right)^{-1} JM^{-1} \right) \tau_{\text{orig}}$$

$$\ddot{x} = \left(JM^{-1} - JM^{-1}\right)\tau_{\text{orig}}$$
$$\ddot{x} = (0)\,\tau_{\text{orig}}$$

The nullspace torque τ_n does not influence the task-space torque. This ensures that the task-space objectives are unaffected, while the nullspace can be used for secondary tasks.

2.5.2 Problem 2

Problem: Write the expression of the pseudo-inverse J^+ and demonstrate that $JJ^+ = I$.

$$J^+ = J^T\left(JJ^T\right)^{-1}$$

Proving $JJ^+ = I$.

$$JJ^+ = J\left(J^T\left(JJ^T\right)^{-1}\right) = JJ^T\left(JJ^T\right)^{-1} = I$$

2.5.3 Problem 3

Problem: Define a damping nullspace torque τ_0.

$$\tau_0 = -ND\dot{q}$$

2.5.4 Problem 4

The nullspace additional degree of freedom can be also used for making some optimizations on the configuration of the manipulator. A possible use case is the avoidance of singularities by means of additional nullspace torques that don't interfere with the TCP motion.

Problem: Using the mapping from Cartesian forces to joint torques $\boldsymbol{\tau} = \mathbf{J}^T\mathbf{F}$, give a qualitative interpretation of the joint torques that result from forces acting in singular directions (how does the robot behave at a singularity?).

When a robot operates near or at a singularity, the Jacobian matrix J loses rank, leading to problematic behavior in the mapping from Cartesian forces to joint torques.

In singular configurations:

- Forces applied in specific directions require very large joint torques. This increments actuator demands and can result in motor failure.
- The system may also exhibit instability, where small joint motions cause large end-effector displacements,making control difficult.

2.5.5 Problem 5

An index of manipulability is $m(q) = \sqrt{\det(JJ^\top)}$ and for the considered 3DOF manipulator, it's given by:

$$m_{\text{kin}}(q) = \sqrt{l_1^2 l_2^2 \sin^2(q_2)}.$$

As the determinant vanishes for singular $J(q)$, this measure gives locally the distance to the singularity. A possible approach is to define a force field that repels the manipulator from singularities when the manipulability is too low.

Problem: Define a quadratic potential function with a scalar coefficient k_s controlling the gain of the singularity avoidance. Use a piecewise definition to restrict the potential to the vicinity of the singularity.

- Define a task coordinate.

$$p(q) = \frac{1}{m_{\text{kin}}}$$

with

$$m_{\text{kin}}(q) = \sqrt{l_1^2 l_2^2 \sin^2(q_2)}.$$

- Define the potential function.

$$U(p) = \begin{cases} \frac{1}{2} k_s p^2, & \text{if } m_{\text{kin}} < m_{\text{threshold}}, \\ 0, & \text{otherwise.} \end{cases}$$

It can be seen, that m_{kin} reaches it's maximum when $q_2 = \frac{\pi}{2}$. This configuration maximizes the ability of the end-effector to move in different directions.

$$m_{\text{kin}}, max = \sqrt{l_1^2 l_2^2 \sin^2(\frac{\pi}{2})} = \sqrt{l_1^2 l_2^2} = l_1 l_2$$

Thus, $m_{\text{threshold}}$ can be set to activate the singularity avoidance when m_{kin} drops, for instance, below 25% of the maximum manipulability.

$$m_{\text{threshold}} = 0.25 m_{\text{kin}}, max = 0.25 l_1 l_2$$

2.5.6 Problem 6

Problem: Write a control torque that implements the singularity avoidance by means of the potential defined in 2.5.5

- Differentiate the potential.

$$F_p = -\frac{\partial}{\partial p} U(p) = -k_s p$$

- Apply the chain rule to get „elastic“ joint torques.

$$\tau = J_p^\top F_p,$$

where

$$J_p = \frac{\partial p}{\partial q} = -\frac{1}{2} \cdot \frac{1}{\left(l_1^2 l_2^2 \sin^2 q_2\right)^{3/2}} \cdot l_1^2 l_2^2 \cdot 2 \sin(q_2) \cos(q_2) = -\frac{cos(q_2)}{\left(l_1^2 l_2^2\right)^{1/2} \cdot \left(\sin(q_2)\right)^2}$$

thus

$$\tau = J_p^\top F_p = -\frac{\cos(q_2)}{\left(l_1^2 l_2^2\right)^{1/2} \cdot \left(\sin(q_2)\right)^2} \cdot \left(-k_s \frac{1}{\sqrt{l_1^2 l_2^2 \sin^2(q_2)}} \right) = \frac{k_s \cos(q_2)}{l_1^2 l_2^2 (\sin(q_2))^3}$$

piecewise difinition

$$\tau = \begin{cases} \frac{k_s \cos(q_2)}{l_1^2 l_2^2 (\sin(q_2))^3}, & \text{if } m_{\text{kin}} < m_{\text{threshold}}, \\ 0, & \text{otherwise.} \end{cases}$$

2.5.7 Problem 7

Problem: Write the complete control law for the Cartesian impedance controller from Problem 2.3.4 and the nullspace optimizations (nullspace damping and singularity avoidance).

$$\tau = \tau_i + \tau_n = \tau_i + \tau_0 + \tau_{null}$$

with

$$\tau_i = J^T \left(-K_t(x - x_d) - D_t \dot{x}\right)$$

$$\tau_0 = -(I - J^T J^{+T}) D\dot{q}$$

$$\tau_{null} = \begin{cases} -(I - J^T J^{+T}) \frac{k_s \cos(q_2)}{l_1^2 l_2^2 (\sin(q_2))^3}, & \text{if } m_{\text{kin}} < 0.25 l_1 l_2, \\ 0, & \text{otherwise.} \end{cases}$$

3 Simulation

3.1 Kinematics and Dynamics

3.1.1 Task 2

Simulate the system from the initial position $q_i = \left[-60°, -30°, 20°\right]^T$ and no input torque $\tau = \left[0, 0, 0\right]^T$. Plot the joint angles and comment on the results.

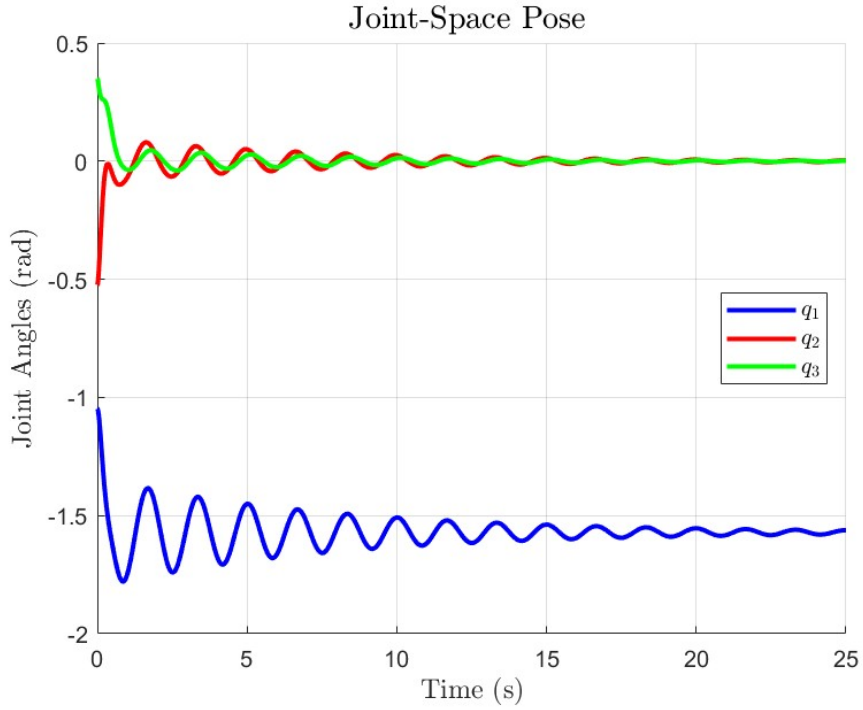

Figure 3.1

Since no torque is applied, the effect of gravity dominates, causing the robot to naturally settle into the configuration $q_{steady} = \left[-90°, 0°, 0°\right]^T$.

3.1.2 Task 3

Implement the calculation of the Cartesian pose and velocity using the forward kinematics and the Jacobian matrix and plot the TCP position and velocity over time for the same experiment as in the previous task.

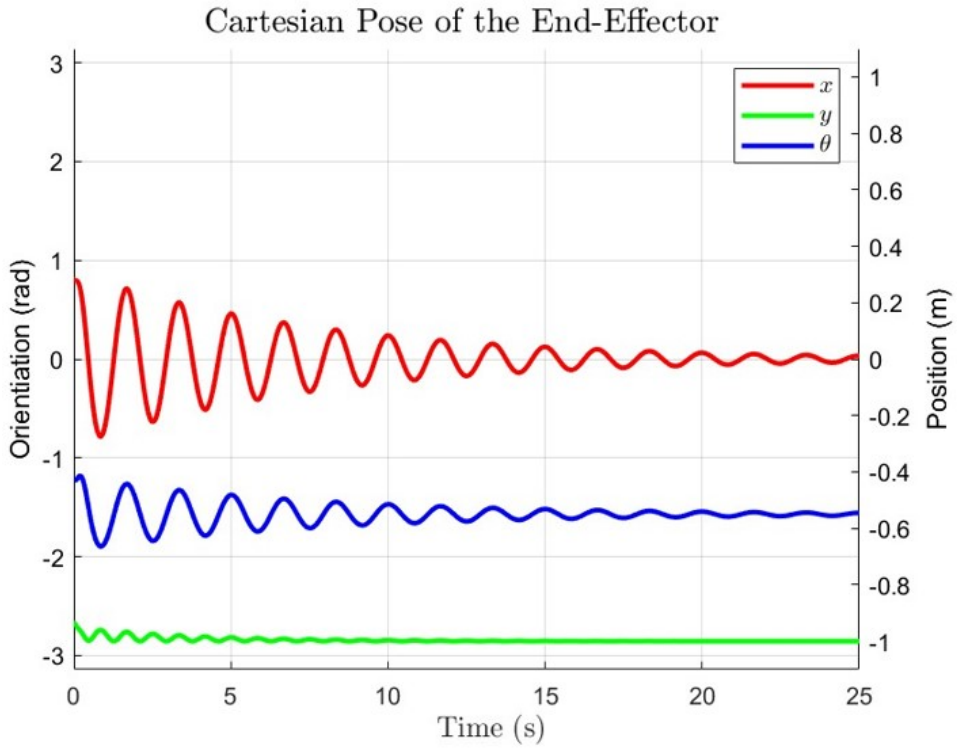

Figure 3.2

Analogously to Figure 3.1,with no torque applied the end-effector naturally settles into the configuration $x_{steady} = \left[0, -1, -90°\right]^T$.

As anticipated, the velocities settle into the steady-state configuration: $\dot{\mathbf{x}}_{steady} = \left[0, 0, 0\right]^T$. As a validation step, it is observed that when the end-effector position reaches its peak (e.g., at second 5), the corresponding velocities drop to zero, consistent with expectations.

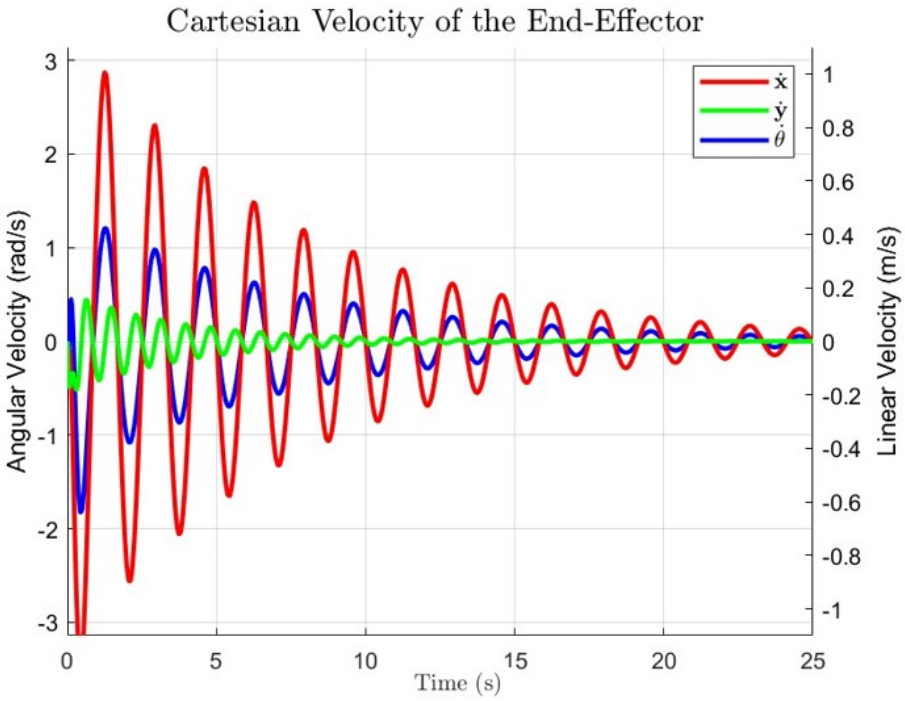

Figure 3.3

3.2 Joint Control

3.2.1 Task 4

Implement a gravity compensation using the gravity torques from the online model. Start the simulation again and compare the robot behavior.

In Figure 3.4, it can be seen that the gravity compensation works as expected. q_{steady} is equal to $q_0 = \left[90°, -45°, 35°\right]^T$

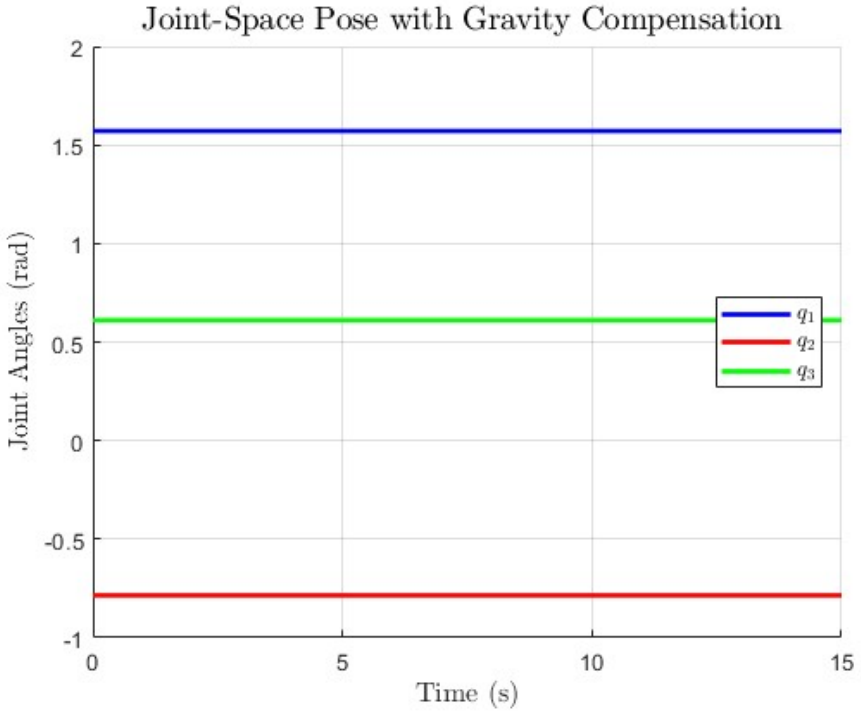

Figure 3.4

3.2.2 Task 5

Implement the joint PD controller described in (2.2) with a constant damping. Compare different settings for the stiffness and damping matrices $\mathbf{K_p}, \mathbf{K_d}$ and tune them to achieve a fast and well-damped response. Plot the joint angles and torques with respect to time for some representative cases of $\mathbf{K_p}$ and $\mathbf{K_d}$.

For the following task consider $q_0 = \left[90°, -45°, 0°\right]^T$ and $q_d = \left[-165°, 45°, -60°\right]^T$.

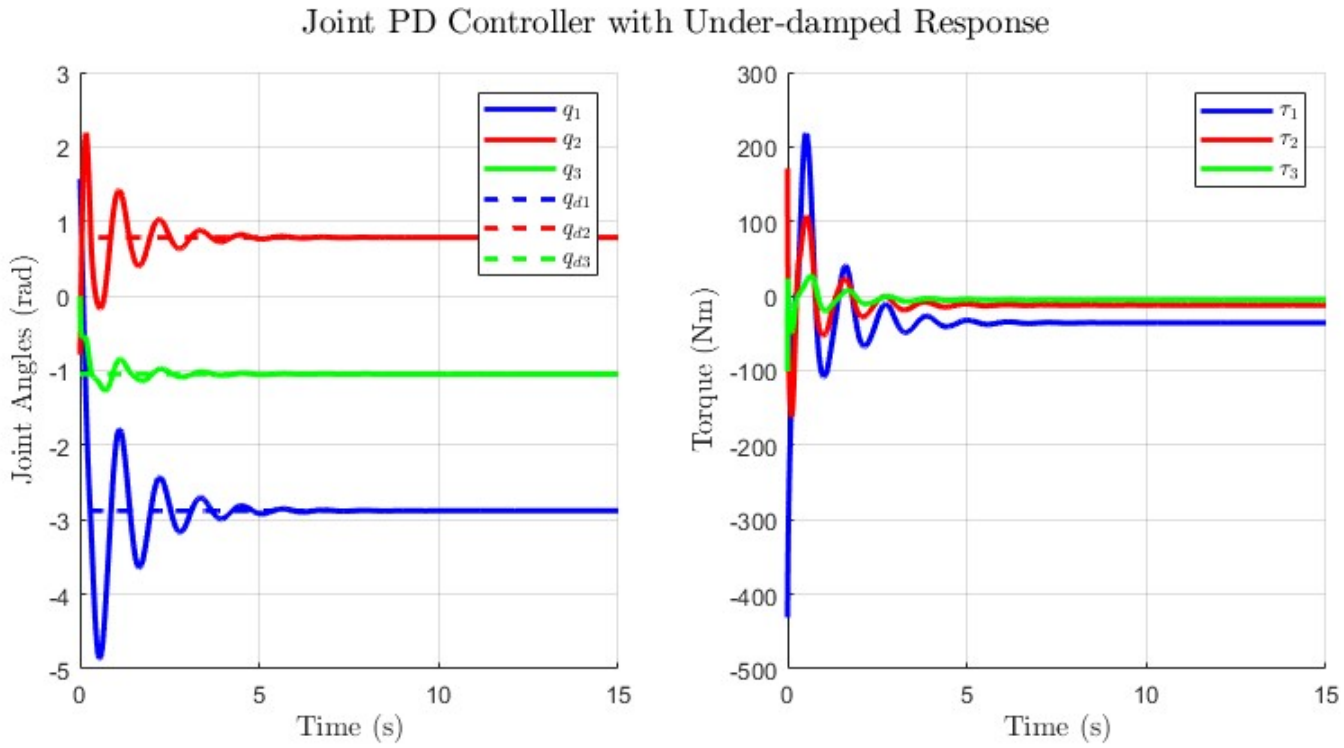

Figure 3.5

The system initially exhibits oscillations and overshoot, which are characteristic of under-damped dynamics. There are large torque spikes as the controller compensates for the error. This behavior arose from the selection of the following stiffness and damping matrices:

$$\mathbf{K}_p = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

.

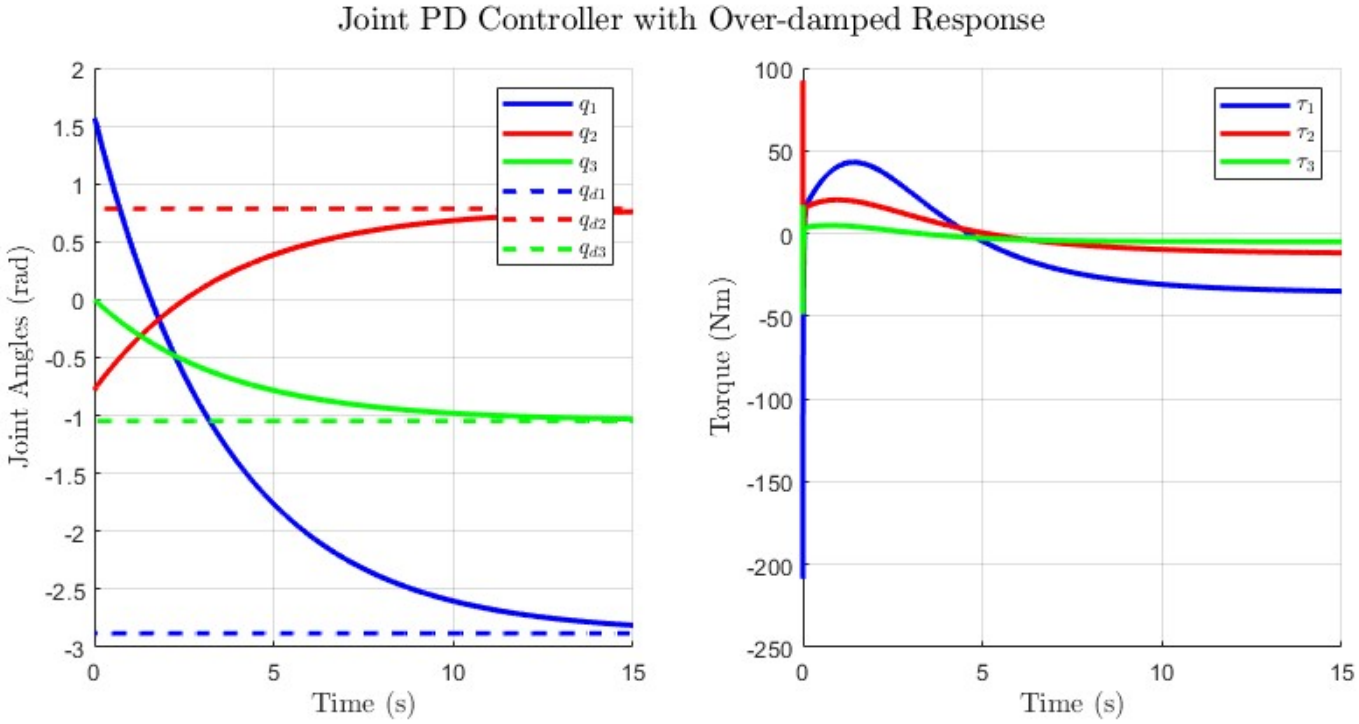

Figure 3.6

An over-damped system avoids overshoot and oscillations, but the transient time increases significantly due to the slower response. Despite this, torque spikes are still present at the beginning of the motion, which are lower in magnitude. This behavior is a consequence from the selection of the following stiffness and damping matrices:

$$\mathbf{K}_p = \begin{bmatrix} 50 & 0 & 0 \\ 0 & 50 & 0 \\ 0 & 0 & 50 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 180 & 0 & 0 \\ 0 & 180 & 0 \\ 0 & 0 & 180 \end{bmatrix}$$

.

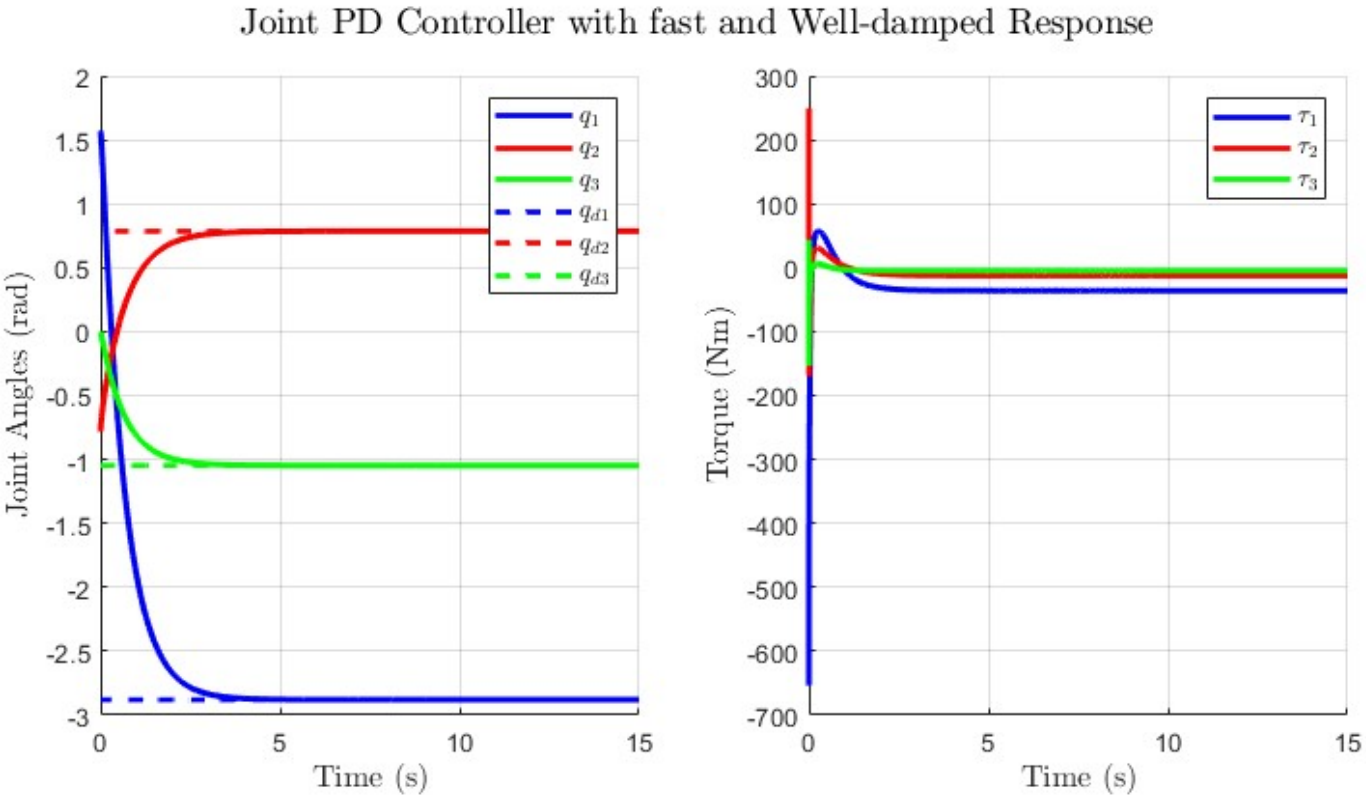

Figure 3.7

The system rapidly converges to the desired joint positions without overshoot or oscillations. The trajectories stabilize smoothly within a short time. Initially, there are torque spikes as the controller compensates for the large initial errors. Figure 3.6 shows a well-tuned PD controller that achieves fast and stable joint motion with minimal transient effects.The following matrices were used to achieve this behaviour:

$$\mathbf{K}_p = \begin{bmatrix} 150 & 0 & 0 \\ 0 & 150 & 0 \\ 0 & 0 & 150 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 100 \end{bmatrix}$$

.

3.2.3 Task 6

Run the simulation again with the gravity compensation and the PD controller, what happens if the gravity compensation is deactivated?

Not using gravity compensation introduces steady-state errors and increases torque demands. It can also lead to possible instability, particularly if the system starts from a position where gravitational forces heavily influence certain joints. Gravity compensation simplifies the controller's task, improving performance.

3.2.4 Task 7

Implement the damping design with the method of the square root matrices. Plot and compare the temporal response of the joint angles for three different damping factors ζ .

For the following task consider $q_0 = \begin{bmatrix}90°, -45°, 60°\end{bmatrix}^T$ and $q_d = \begin{bmatrix}-75°, 45°, -60°\end{bmatrix}^T$ and a constant stiffness matrix $\mathbf{K}_p = \begin{bmatrix}10 & 0 & 0\\ 0 & 10 & 0\\ 0 & 0 & 10\end{bmatrix}$.

The simulation results demonstrate the influence of different damping factors on the temporal response of the joint angles:

- For $\zeta = 0.2$, the system exhibits underdamped behavior with significant oscillations and overshoot, reflecting insufficient damping (Figure 3.8).
- At $\zeta = 1$, the system achieves a critically damped response, converging to the desired positions quickly without oscillations (Figure 3.9).
- Finally, with $\zeta = 1.8$, the system becomes overdamped, leading to slower convergence to the desired positions without oscillations (Figure 3.10).

Tuning ζ is crucial for balancing response speed and stability to meet specific application requirements.

The resulting damping matrices, computed using the method of square root matrices, are tailored for each damping factor ζ. For $\zeta = 0.2$, $\zeta = 1.0$, and $\zeta = 1.8$, the respective damping matrices are $K_d^{0.2}$, $K_d^{1.0}$, and $K_d^{1.8}$.

$$K_d^{0.2} = \begin{bmatrix}1.86 & 0.74 & 0.21\\ 0.74 & 0.76 & 0.14\\ 0.21 & 0.14 & 0.34\end{bmatrix}, K_d^{1.0} = \begin{bmatrix}9.28 & 3.72 & 1.06\\ 3.72 & 8.82 & 0.70\\ 1.06 & 0.70 & 1.68\end{bmatrix}, K_d^{1.8} = \begin{bmatrix}16.71 & 6.69 & 1.91\\ 6.70 & 6.87 & 1.25\\ 1.90 & 1.26 & 3.03\end{bmatrix}$$

.

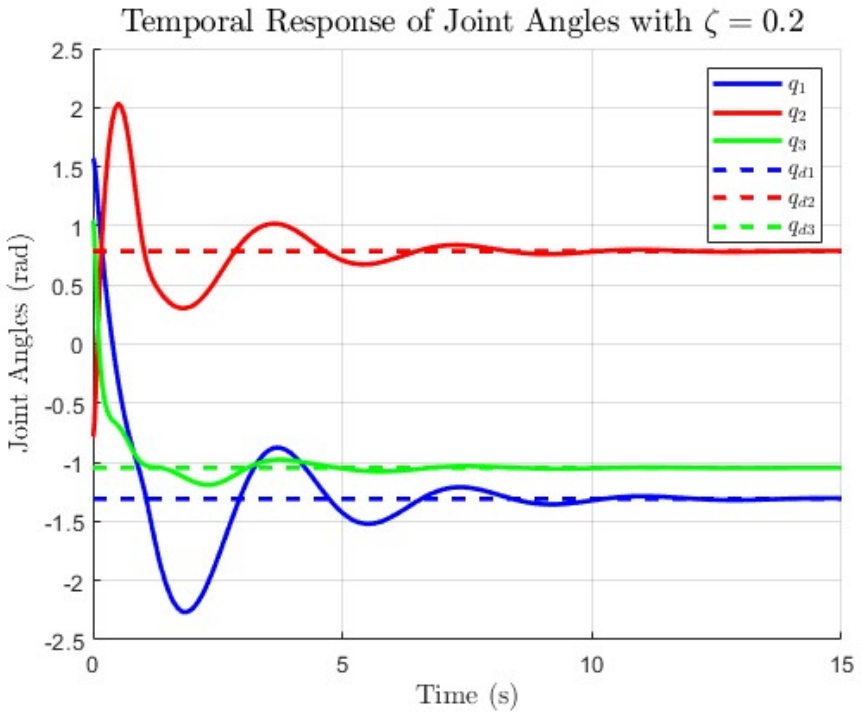

Figure 3.8

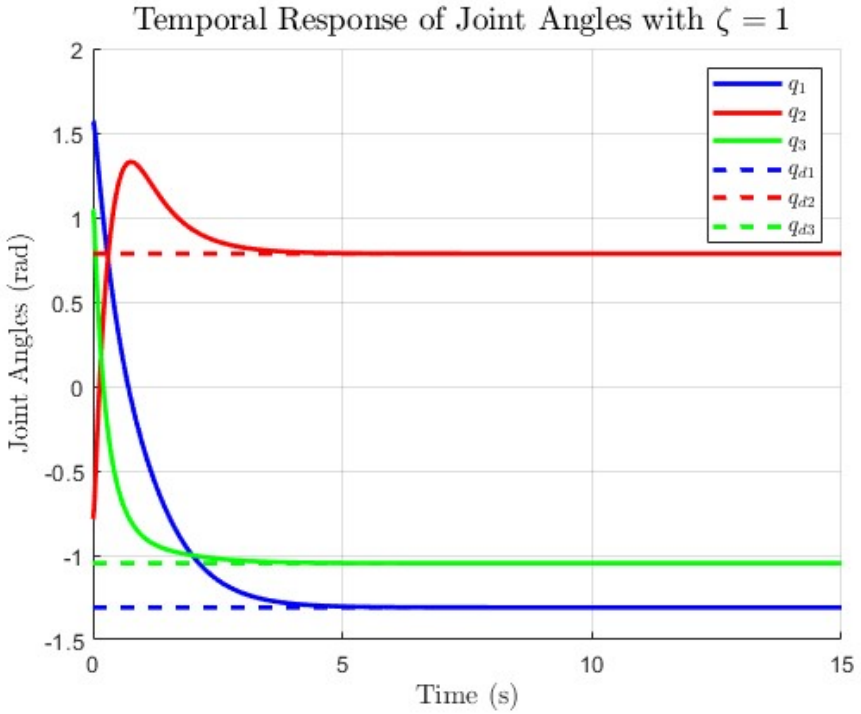

Figure 3.9

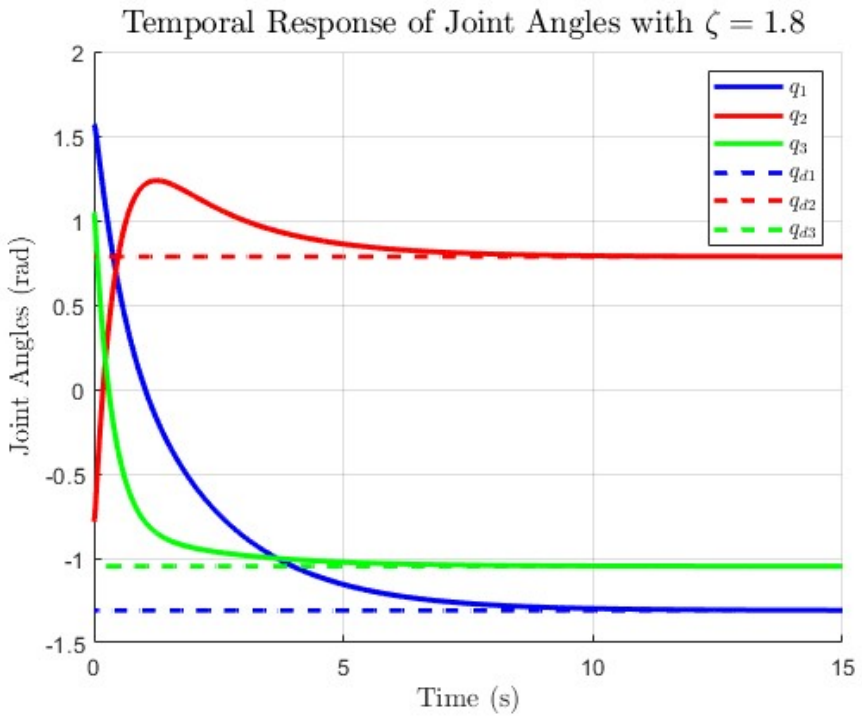

Figure 3.10

3.3 Translational Cartesian Impedance Control

3.3.1 Task 8

Implement the translational Cartesian controller derived in the problem 2.3.4 and tune the stiffness and damping matrices. After trying various desired positions, command the position $[x, y] = [0, 1.5]$. What happens in this case? Show temporal responses for a reachable and for the unreachable target position.

For the following task consider $q_0 = \left[-90°, -45°, 0°\right]^T$ and following constant matrices :

$$\mathbf{K}_p = \begin{bmatrix} 40 & 0 \\ 0 & 40 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 15 & 0 \\ 0 & 15 \end{bmatrix}$$

.

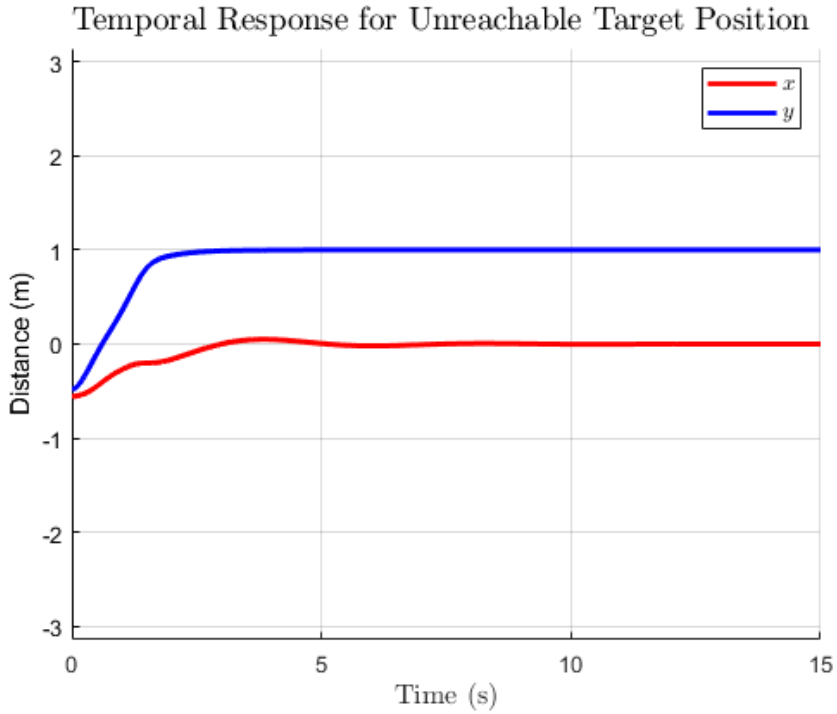

Figure 3.11

For Figure 3.11, consider $[x_d, y_d] = [0, 1.5]$. For this simulation , abrupt changes or oscillations in the end-effector position were expected as the robotic arm attempted to reach the desired configuration: $[x, y] = [0, 1.5]$ However, the simulation does not exhibit the expected instability, suggesting that the system might have compensated for the singularity.

Taking into account the non-zero determinant of $J_{Base}, steady$ = -0.0153 , the base jacobian is not singular.However, the determinant's small magnitude suggests that the matrix is close to being singular, indicating a potential numerical instability or near-singularity.

$$J_{Base}, steady = \begin{bmatrix} 0.4828 & 0.1494 & -0.0863 \\ -0.5577 & -0.5577 & -0.3220 \\ 1.0000 & 1.0000 & 1.0000 \end{bmatrix}$$

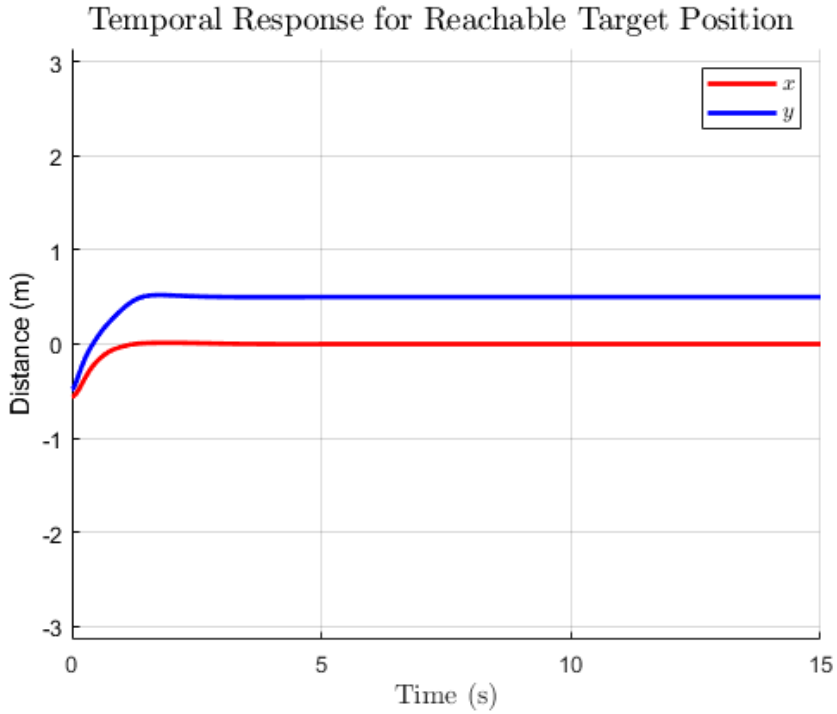

Figure 3.12

For Figure 3.12, consider $[x_d, y_d] = [0, 0.5]$. The TCP reaches the desired position with a well-damped response.

In this case, $J_{Base}, steady = -0.1103$, which is an order of magnitude larger compared to the determinant from the last simulation -0.0153. This indicates that the system is significantly farther from a singularity, suggesting a more stable and robust configuration in the current setup.

$$J_{Base}, steady = \begin{bmatrix} -0.5000 & -0.4455 & -0.1125 \\ -0.0000 & 0.3289 & 0.3138 \\ 1.0000 & 1.0000 & 1.0000 \end{bmatrix}$$

3.3.2 Task 9

Plot the position of the TCP with respect to time for some representative cases of K_p and K_d when setting the desired TCP position to a reachable position.

In the following section, three different system responses will be presented, evaluating the joint influence of the damping and stiffness matrices on the system's behavior. Consider $q_0 = \begin{bmatrix} 90°, -45°, 35° \end{bmatrix}^T$ and $[x_d, y_d] = [-0.2, -0.5]$.

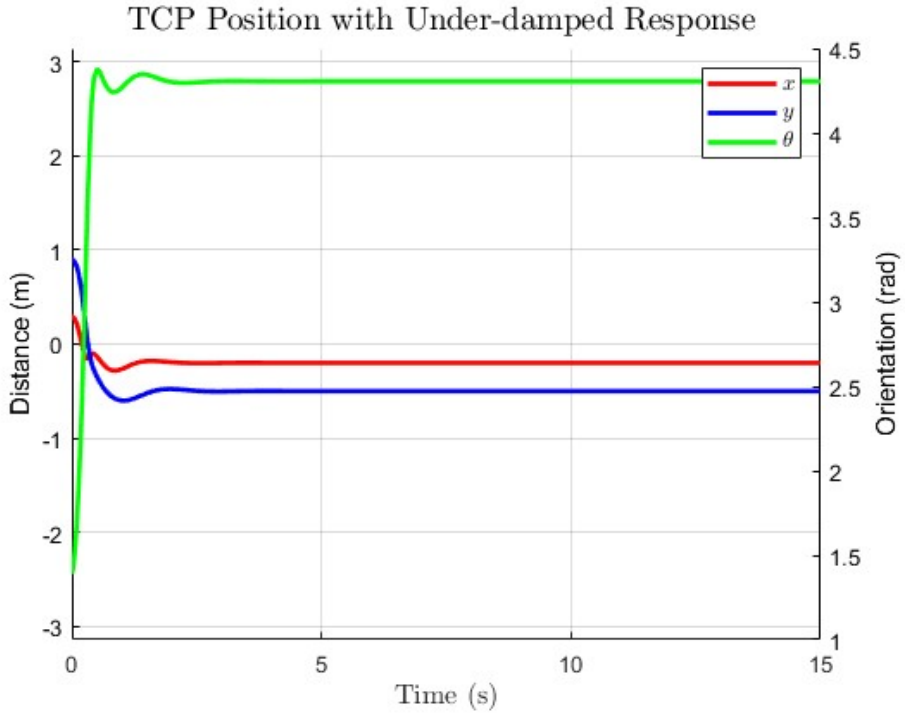

Figure 3.13

For the experiment shown in Figure 3.123, the system eventually stabilizes at the desired position, but the transient period includes overshoot and oscillations. Following matrices were used:

$$\mathbf{K}_p = \begin{bmatrix} 80 & 0 \\ 0 & 80 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$$

.

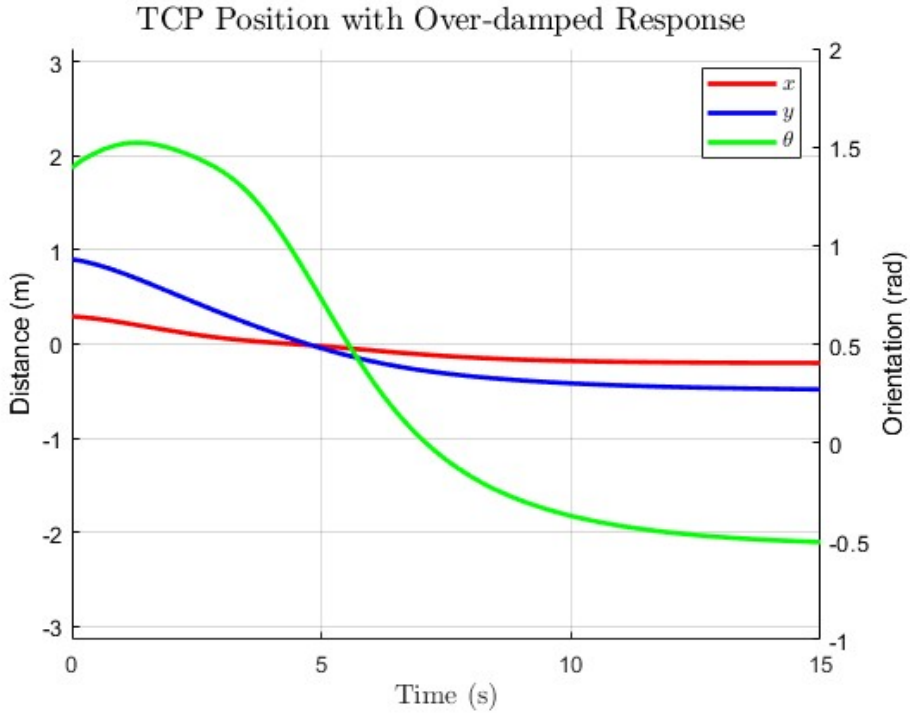

Figure 3.14

For the experiment shown in Figure 3.14, an overdamped behavior can be observed.The robot arm moves to the desired position at a slower rate . The damping is so high that it resists motion. Following matrices were used:

$$\mathbf{K}_p = \begin{bmatrix} 8 & 0 \\ 0 & 8 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 20 & 0 \\ 0 & 20 \end{bmatrix}$$

.

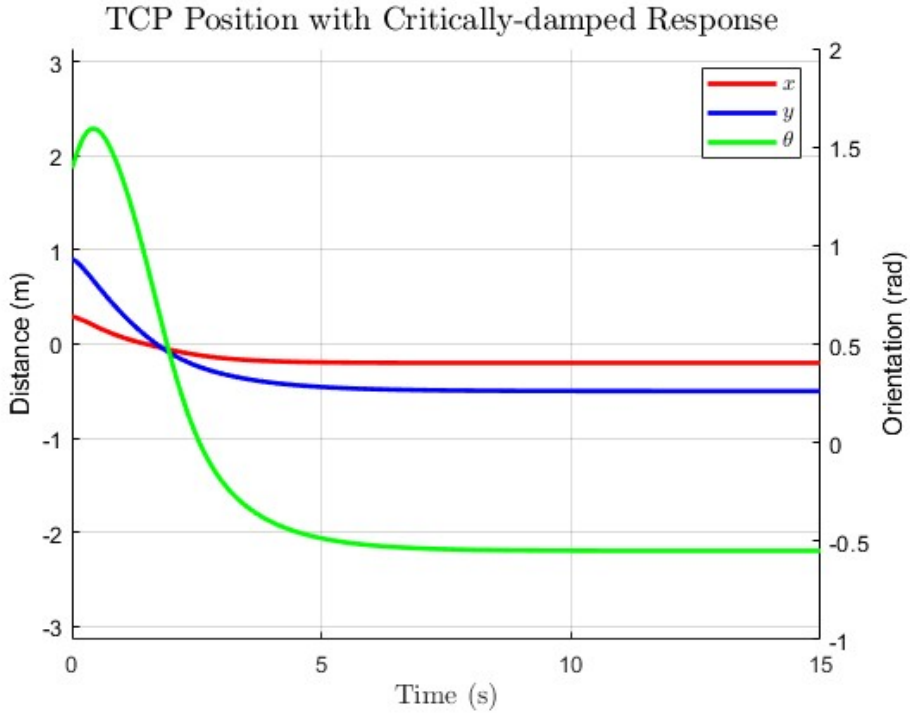

Figure 3.15

For the experiment shown in Figure 3.15, a critically-damped response can be seen. The system goes to the desired cartesian position in the shortest possible time without oscillating. This happens when the damping and stiffness are balanced to avoid oscillations while ensuring a rapid stabilization.Following matrices were used:

$$\mathbf{K}_p = \begin{bmatrix} 40 & 0 \\ 0 & 40 \end{bmatrix}, \mathbf{K}_d = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}$$

.

3.4 Collision Detection

3.4.1 Task 10

Implement the external torque observer derived in Problem 2.4.4. To simulate collisions, you can use the Wall block provided to you in the library. Make some representative plots that the observer is correctly detecting the impacts against the wall.

In the following task, the starting position will be set as $q_0 = \begin{bmatrix} 90°, -45°, 35° \end{bmatrix}^T$ and the desired position as $[x_d, y_d] = [-0.5, 0.5]$. The wall is placed at $Position = \begin{bmatrix} -0.2, 0 \end{bmatrix}$ and $Orientation = \begin{bmatrix} 1, 0 \end{bmatrix}$. Consider a constant observer gain =

$$K_I = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

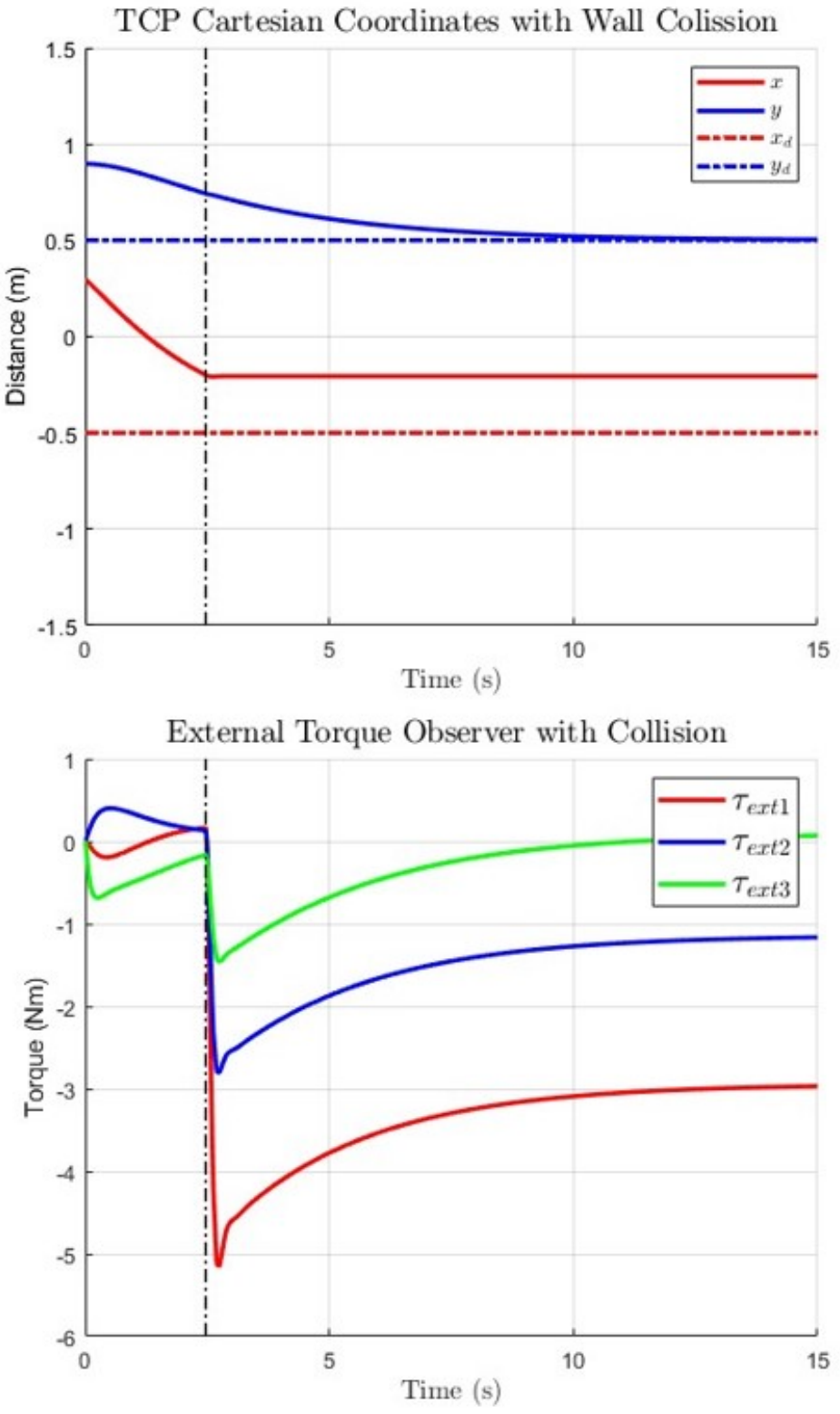

Figure 3.16

By examining the top graph in Figure 3.16, it is evident that the X-coordinate of the TCP is unable to reach the desired position $X_d = -0.5$ and instead stabilizes at $X_{steady} = -0.2$,which corresponds precisely to the location of the obstacle.

In the lower graph, it can be observed that the observer detects a sudden spike in external torques, which then gradually stabilizes.

At t=2.47047 seconds, marked by a dotted black line, the TCP makes contact with the wall, and the observer successfully detects the event.

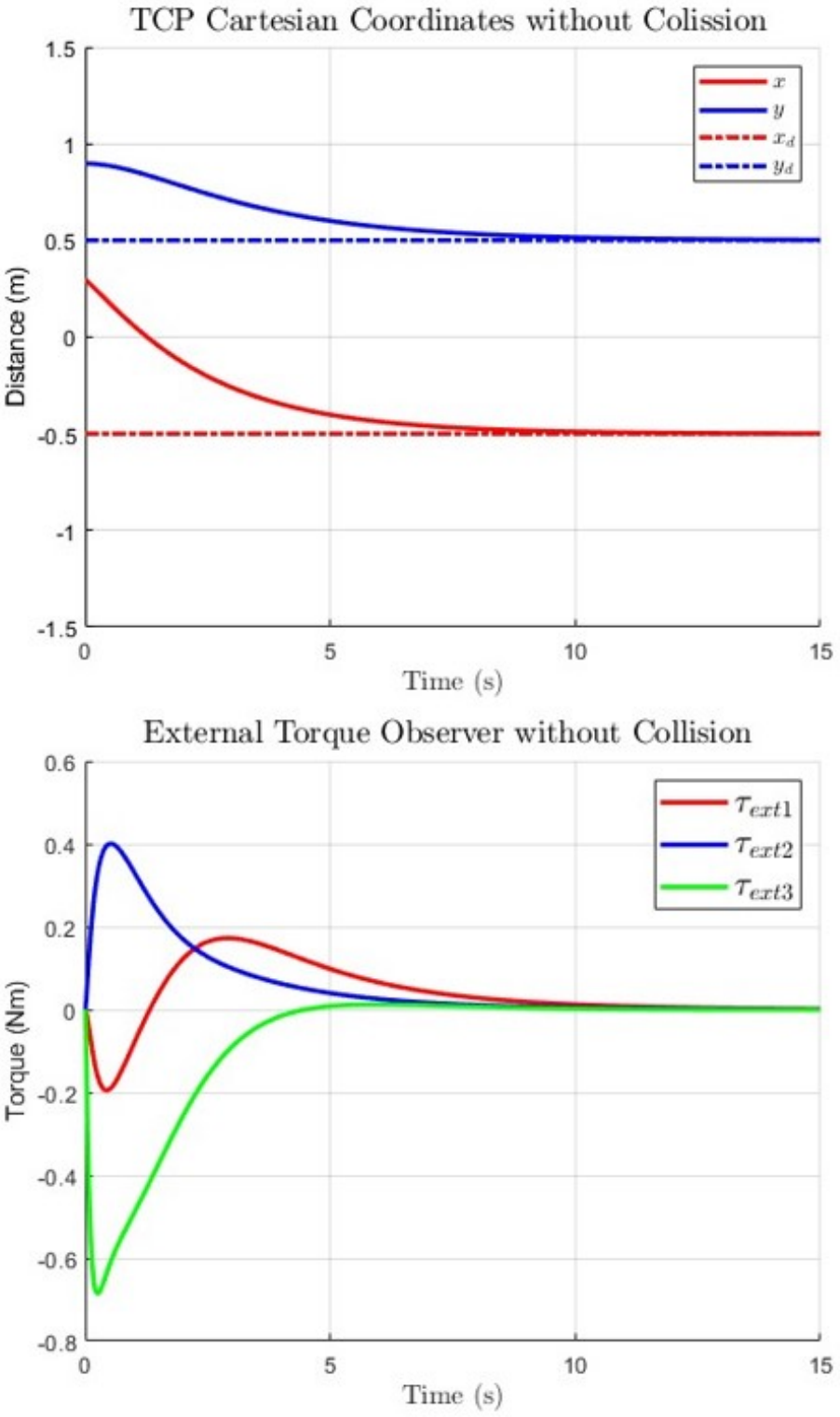

Figure 3.17

By examining the top graph in Figure 3.17, it can be observed that the X-coordinate of the TCP converges to $X_d = -0.5$ in the absence of the wall.

In the lower graph,as expected, no significant external torques are detected, as they are one order of magnitude smaller than those observed in Figure 3.16.

3.4.2 Task 11

Add a source of noise to the measurements of the joint angles and velocities, q and $\dot{q}$. Design the gain K_I such that the estimator achieves a fast response while effectively rejecting the noisy components of the measurements.

Band-limited white noise block was used to inject noise in the joint position and joint velocity.For q, the noise power was set to 1×10^{-4} and for q_dot 1×10^{-7} with sample time of 0.1 and Seed 23341.

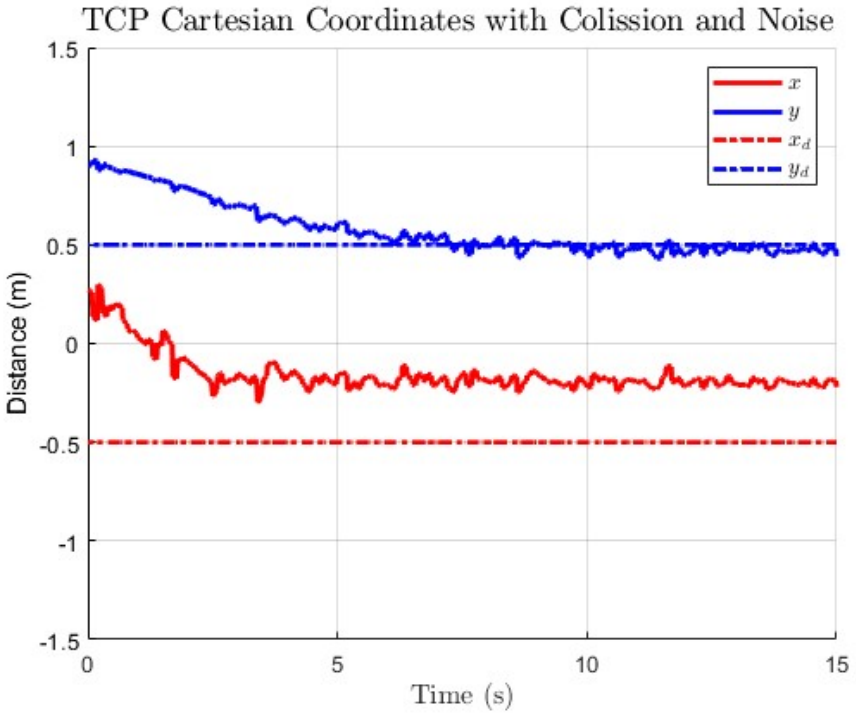

Figure 3.18

In Figure 3.18, the behaviour of the system with injected noise can be observed. Same conditions as in Task 10 were used.

3.4.3 Task 12

Plot the behavior of the observer for different cases of K_I and comment the differences.

In the following task, same as task 10, the starting position will be set as $q_0 = \left[90°, -45°, 35°\right]^T$ and the desired position as $[x_d, y_d] = [-0.5, 0.5]$. The wall is placed at $Position = \left[-0.2, 0\right]$

and $Orientation = \begin{bmatrix} 1, 0 \end{bmatrix}$.

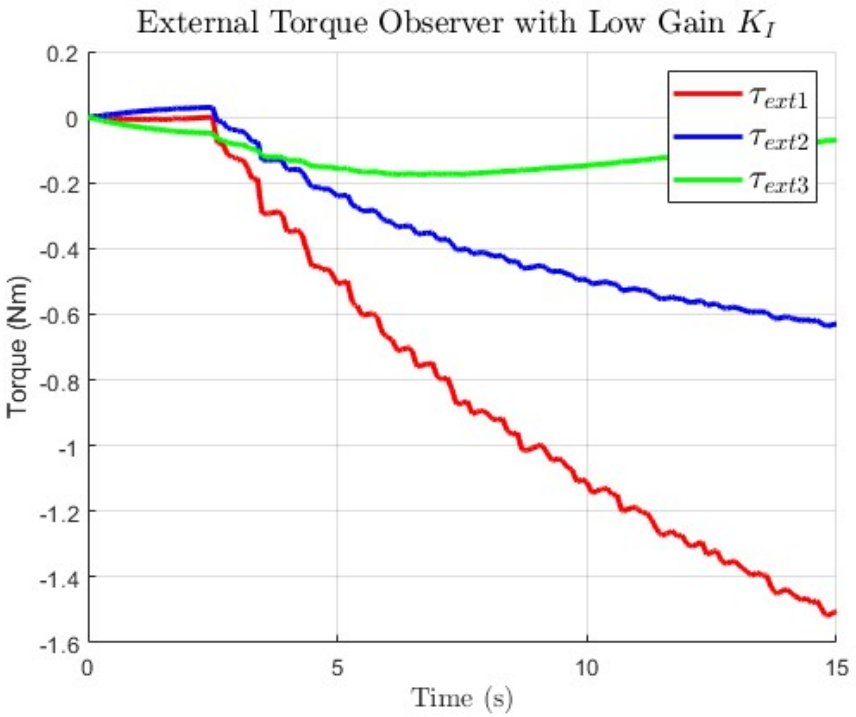

Figure 3.19

In Figure 3.19, the behaviour of a low gain K_I is observed. The noise is filtered and the signal tends to be smooth. On the other hand, the response of the observer is rather slow in comparison to the input signal. Following gain was used:

$$K_I = \begin{bmatrix} 0.05 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.05 \end{bmatrix}$$

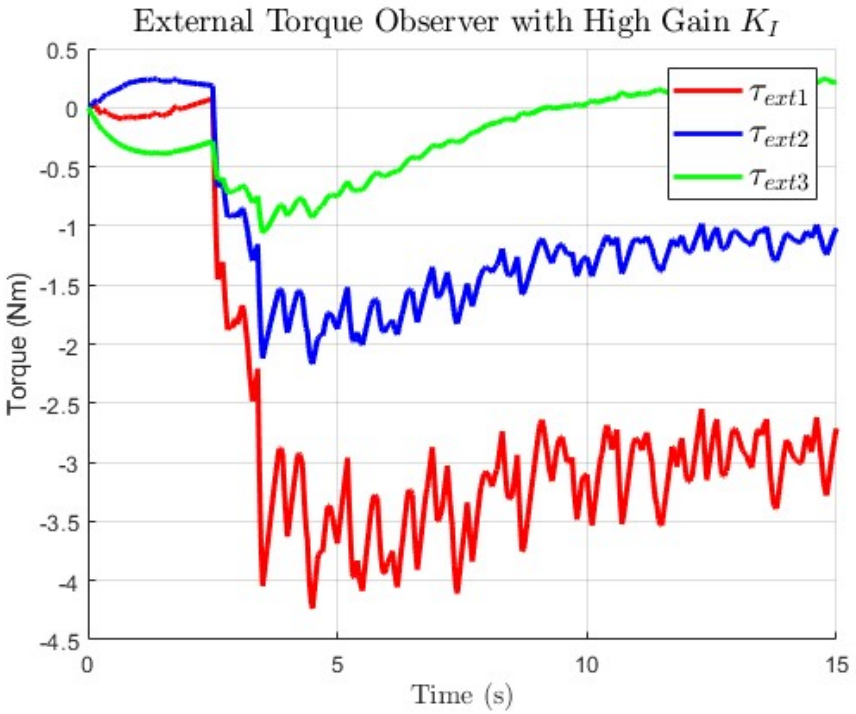

Figure 3.20

In Figure 3.20, the behavior of a high gain K_I is analyzed. While the observer's response is relatively fast, the system demonstrates increased sensitivity to noise . This compromises the ability to effectively distinguish between actual signal changes and noise.. Following gain was used:

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

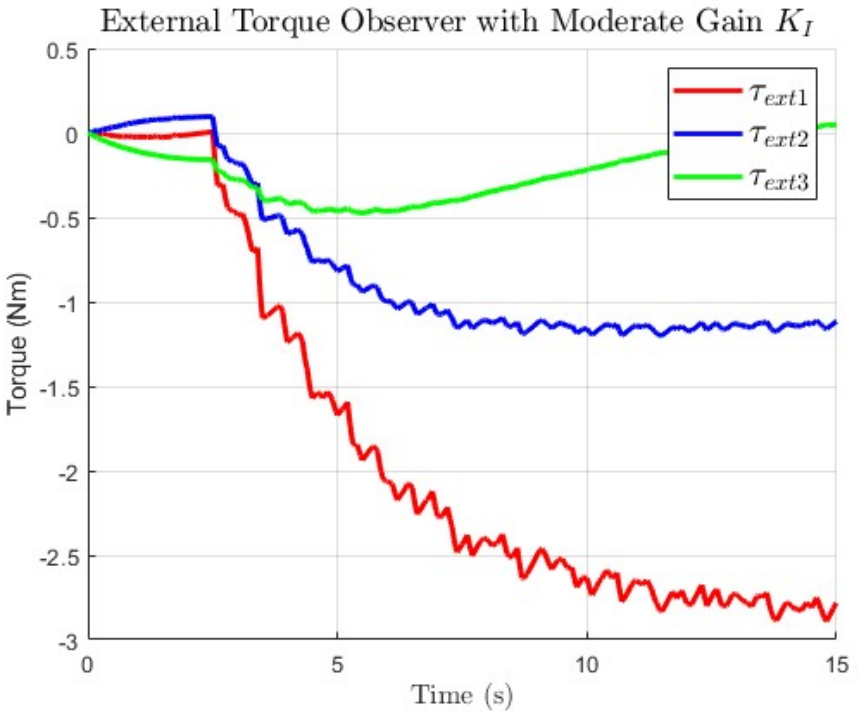

Figure 3.21

The system in Figure 3.21 achieves a balanced level of noise rejection, where some noise filtering is applied, but the observer remains sufficiently responsive to changes in the input signal. Following gain was used:

$$K_I = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$$

In conclusion, an external torque observer with an adequate gain shows a balance between noise rejection and response speed.

3.5 (NO) Nullspace Optimizations

The translational Cartesian controller leaves 1DOF of the manipulator out of control. If the simulation is appropriately realized, you should see the manipulator moving while the TCP remains fixed in the desired position (null space motion). This null space motion can be further damped by means of an appropriately designed null space controller.

3.5.1 Task 16

Implement a null space damping to not interfere with the TCP motion.

For the following task consider $q_0 = \left[90°, -45°, 35°\right]^T$ and $q_d = \left[-120°, 45°, -110°\right]^T$ and a

following constant matrices $\mathbf{K}_p = \begin{bmatrix} 15 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 15 \end{bmatrix}$ $\mathbf{K}_d = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}$ $\mathbf{D}_{nullspace} = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$.

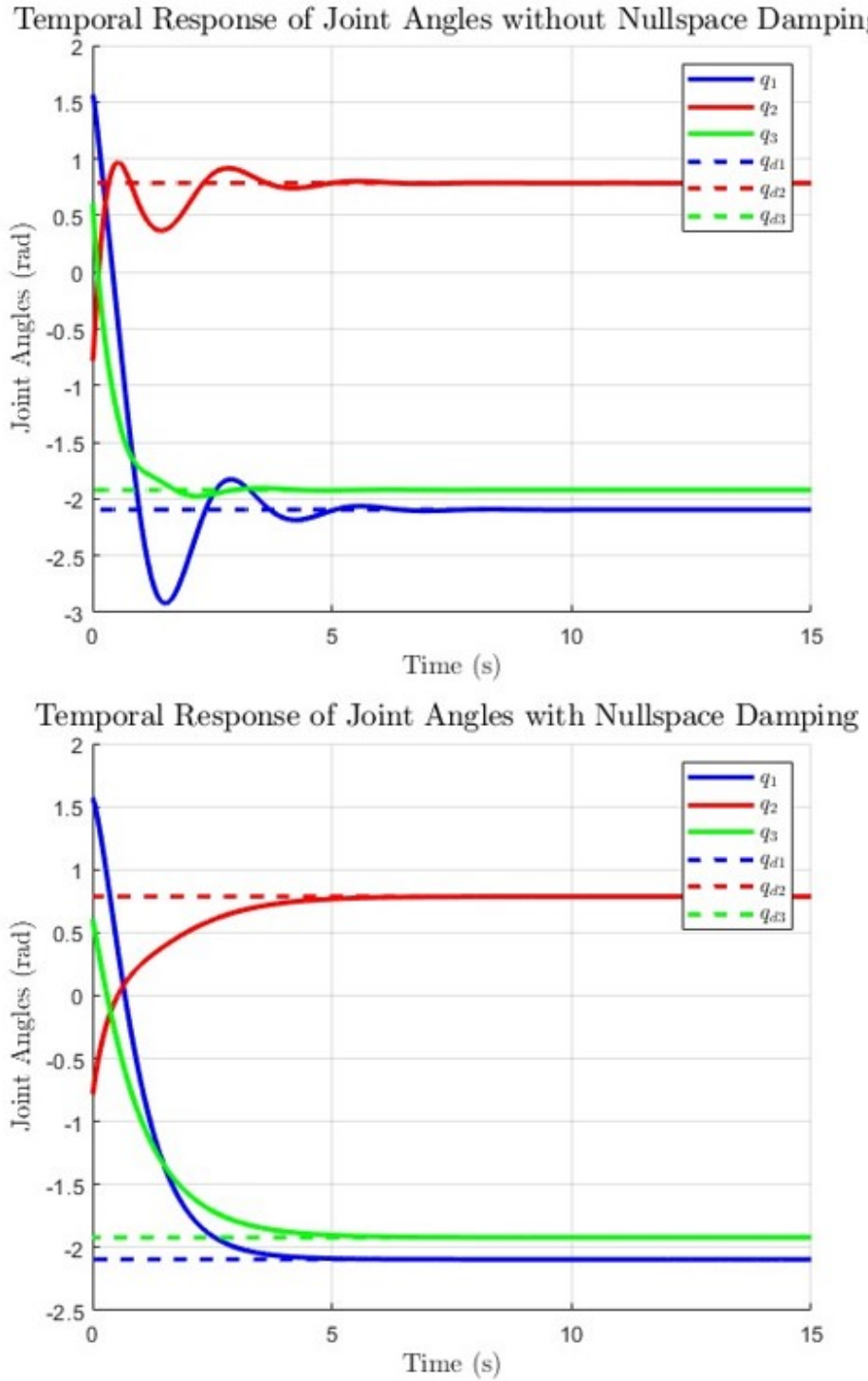

Figure 3.22

In the top graph, which represents the response without null space damping, the system demonstrates oscillatory behaviour.

In contrast, the bottom graph shows the response when null space damping is applied. Here, the joint angles converge smoothly to their desired values without any overshoot. The system reaches the steady state much faster without interfering with the desired objectives.

3.5.2 Task 17

Write an Embedded Matlab Function to implement the results of Problem 2.5.2.

```
function tao_sing = matrix_sqrt(q2, percentage, Ks, l1, l2)
    m_kin = sqrt((l1^2) * (l2^2) * (sin(q2)^2));

    m_threshold = percentage*l1*l2;

    if m_kin < m_threshold
       tao2_sing= (Ks*cos(q2))/((l1^2) * (l2^2)*(sin(q2)^3)+0.000001);
    else
       tao2_sing=0;
    end

    tao_sing= [0; tao2_sing; 0];
end
```