KU LEUVEN

FACULTEIT INGENIEURSWETENSCHAPPEN

Optimization of Mechatronic Systems
Robot Arm

Report

Group 7
Andrea Alboni - r1030437
Cheng Cui - r0964608
Academic year 2024-2025

Contents

1.0 Introduction

In this report, we explore the optimal control of the two-link planar robot arm shown in Figure 1.1. The study involves deriving the state-space representations of the system dynamics and using them to formulate and solve optimal, pose to pose, control problems under two distinct control strategies: acceleration and torque control.

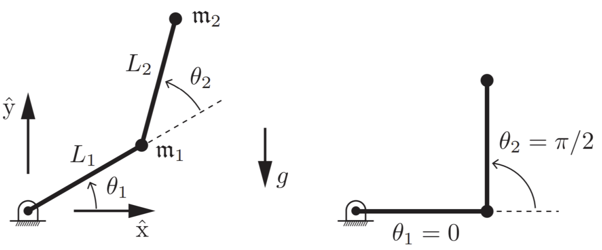


Figure 1.1: 2R open chain under gravity

Parameter	Value
m_1	$3\,\mathrm{kg}$
m_2	$1\,\mathrm{kg}$
l_1	$0.5\,\mathrm{m}$
l_2	$0.3\,\mathrm{m}$

Additionally, it is addressed the derivation of the forward kinematics equations, which relate the joint angles to the pose of the end effector. It is also examined a time-optimal control problem for the acceleration-controlled robot, where the goal is to minimize the time required to move between specified joint positions while adhering to control bounds and avoiding collisions with both itself and the environment.
The results of these optimization problems are analyzed to assess the feasibility and convexity of the formulations. The outcomes are visualized through state and control trajectories, which provide insights into the robot's motion and the effort required to achieve the desired performance.

1.1 Question 1 - System Dynamics Equations for State-Space Formulation

The dynamics of the robot manipulator can be expressed using the equations of motion derived from the Lagrangian formulation. These equations capture the relationship between joint positions θ, velocities $\dot{\theta}$, accelerations $\ddot{\theta}$, and the applied torques τ. The general equation is given by:

$$\tau = M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta)$$

We now formulate the state-space representation for the two configurations. In both cases, the states of the system are the joint positions θ and the joint velocities $\dot{\theta}$:

$$x = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

The configurations differ in their control inputs: either joint accelerations $u = [\ddot{\theta}_1, \ddot{\theta}_2]^T$ (acceleration controlled), or joint torque $u = [\tau_1, \tau_2]^T$ (torque control). Both torque and acceleration controllers are assumed to be perfect.

1.1.1 Acceleration-Controlled Robot

Given that:

$$\dot{x} = \begin{bmatrix} \dot{\theta} \\ \ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ u \end{bmatrix}$$

where u represents the control input, the general state-space representation in linear form is structured as:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

Therefore, for the acceleration controlled case, the state-space model is:

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}$$

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

1.1.2 Torque-Controlled Robot

Using the equation of motion, we can solve for $\ddot{\theta}$ as:

$$\ddot{\theta} = M(\theta)^{-1}\Big(\tau - c(\theta, \dot{\theta}) - g(\theta)\Big)$$

Thus, the state-space representation is nonlinear and can be written as:

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \mathbf{M}(\theta)^{-1}[\tau(\theta) - \mathbf{c}(\theta, \dot{\theta}) - \mathbf{g}(\theta)] \end{bmatrix}$$

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

1.2 Question 2 - Optimal Control Problem Formulation and Solution

Two optimal control problems, corresponding to the two control strategies, are formulated. For both, the aim is to control the robot arm from $\theta_{\text{initial}} = [0.1, 0.5]^T$ rad to $\theta_{\text{terminal}} = [1.0, 1.9]^T$ rad in 2 seconds. Initial and terminal joint velocities are equal to 0. During the motion, self collisions must be avoided which generally implies that crashes between links and between the end-effector and the base must be prevented. Being l_2 shorter than l_1 no collision is possible between the end-effector and base, meanwhile to avoid links collisions a constraint on θ_2 was added preventing it to be equal to π. The implementation of this constraint, in code, and of the objective functions is detailed below, along with the results of solving the optimization problems for each control strategy.

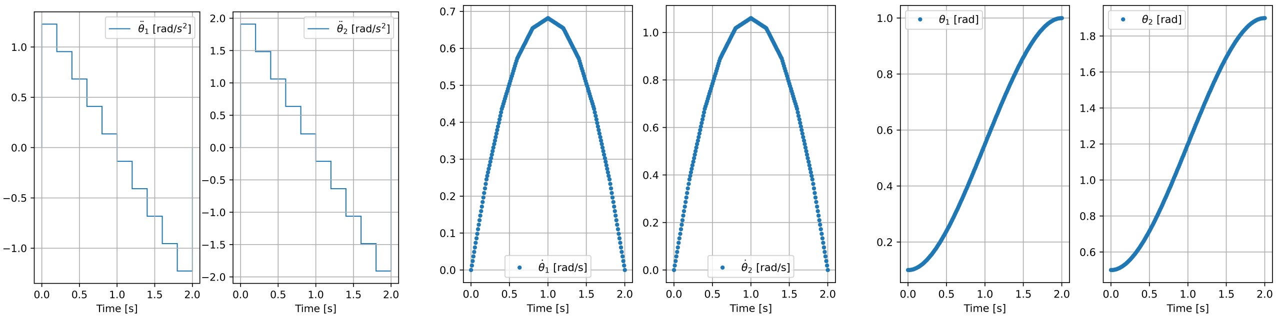
```python
# Motion constraints - Avoid self-collision
epsilon = 0.1
# Ensure theta2 is less than pi - epsilon
ocp.subject_to(theta2 <= pi - epsilon)
# Ensure theta2 is greater than -pi + epsilon
ocp.subject_to(theta2 >= -pi + epsilon)
```

1.2.1 Acceleration-Controlled Robot

The objective function to be minimized is the mean-square control effort which for a system with control inputs $u(t)$ over a time horizon $[0, T]$ can be expressed as:

$$J = \frac{1}{T}\int_0^T \|u(t)\|^2 \, dt = \frac{1}{T}\int_0^T \left(\ddot{\theta}_1^2 + \ddot{\theta}_2^2\right) \, dt \tag{1.1}$$

In Figure 1.2 is shown the evolution over time of the control inputs $\ddot{\theta}_1$ and $\ddot{\theta}_2$, joint velocities and positions obtained by solving the presented optimal control problem.



(a) Joint Accelerations (b) Joint Velocities (c) Joint Positions

Figure 1.2: Acceleration controlled optimization problem

Convexity analysis

- Objective function:
  The mean-square control effort, equation 1.1, is a quadratic function in the control inputs $u(t)$:

$$f(u) = u(t)^t Q u(t)$$

  For the objective function to be convex, the Hessian matrix Q, must be positive semi-definite ($Q \geq 0$). In this case, Q is a 2×2 identity matrix, which satisfies the positive semi-definiteness condition. Therefore, the objective function is convex.

- System dynamics:
  As shown in Section 1.1.1, the state-space model of the system is linear, with the matrices A, B and C defining the dynamics. A linear system, when paired with a quadratic objective function, preserves convexity. Consequently, the system dynamics contribute to the convexity of the problem.

- Constraints:
  The problem constraints are affine, as such they do not affect the convexity of the optimization problem. These constraints are given as:

$$
\begin{aligned}
&\theta_{\text{initial}} = \begin{bmatrix} 0.1 \\ 0.5 \end{bmatrix} \text{ rad}, \quad \theta_{\text{terminal}} = \begin{bmatrix} 1.0 \\ 0.9 \end{bmatrix} \text{ rad}, \\
&\dot{\theta}_{\text{initial}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{ rad/s}, \quad \dot{\theta}_{\text{terminal}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{ rad/s}, \\
&|\theta_2| < \pi.
\end{aligned}
$$

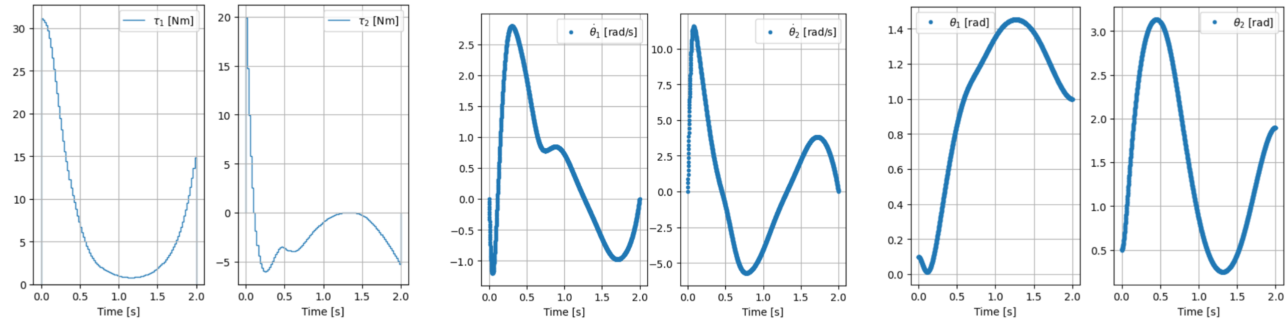
Based on this analysis, the optimization problem satisfies the conditions for convexity: a quadratic objective function with a positive semi-definite Hessian, linear system dynamics, and affine constraints. Therefore, the optimization problem is convex.

1.2.2 Torque-Controlled Robot

Similarly, for the torque controlled version, the objective function can be expressed as:

$$
J = \frac{1}{T} \int_0^T \|u(t)\|^2 \, dt = \frac{1}{T} \int_0^T \left(\tau_1^2 + \tau_2^2\right) \, dt \tag{1.2}
$$

The evolution of the control inputs τ_1 and τ_2, joint velocities and joint positions over time, as obtained by solving the optimal control problem, is presented in Figure 1.3.



(a) Torques (b) Joint Velocities (c) Joint Positions

Figure 1.3: Torque controlled optimization problem

Convexity analysis

- Objective function:
  The objective function for the torque-controlled, equation 1.2, case shares the same structure as the one used in the acceleration-controlled case. The difference lies in the definition of $u(t)$, which does not affect the validity of the previously presented convexity analysis for the objective function.
- Constraints:
  The problem constraints are affine, as such they do not affect the convexity of the optimization problem. These constraints are given as:

$$
\begin{aligned}
&\theta_{\text{initial}} = \begin{bmatrix} 0.1 \\ 0.5 \end{bmatrix} \text{ rad}, \quad \theta_{\text{terminal}} = \begin{bmatrix} 1.0 \\ 0.9 \end{bmatrix} \text{ rad}, \\
&\dot{\theta}_{\text{initial}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{ rad/s}, \quad \dot{\theta}_{\text{terminal}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{ rad/s}, \\
&|\theta_2| < \pi.
\end{aligned}
$$

According to this analysis, the optimization problem satisfies the conditions for convexity: a quadratic objective function with a positive semi-definite Hessian and affine constraints.

1.3 Question 3 - Time-Optimal Control

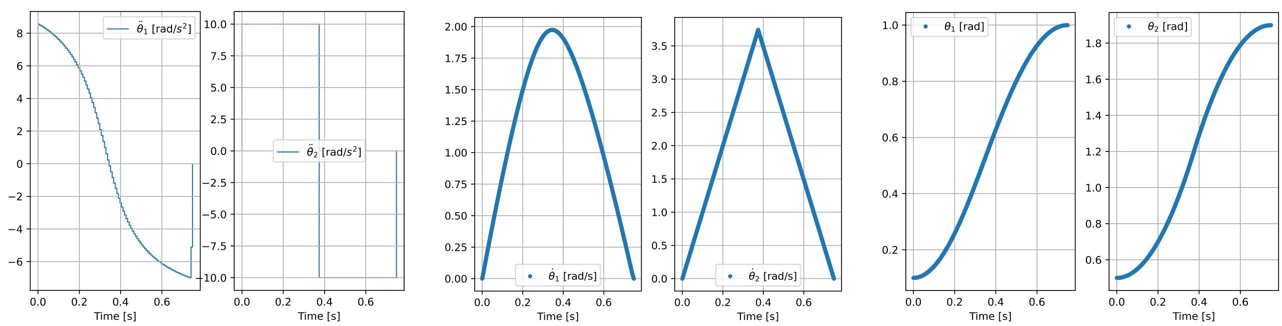
In this section, a time-optimal control problem is formulated for the acceleration-controlled robot. The objective is to minimize the time required for the robot to move from the initial joint position $\theta_{\text{initial}} = [0.1, 0.5]^T$ rad to the target terminal position $\theta_{\text{terminal}} = [1.0, 1.9]^T$ rad. The problem includes constraints to ensure that the robot starts and ends its motion with zero joint velocities and satisfies the acceleration limits: $-10\ \text{rad/s}^2 \leq \ddot{\theta_r} \leq 10\ \text{rad/s}^2$. To guarantee safety, additional constraints are incorporated to prevent self-collision, presented before, and avoid contact with the ground, which is assumed to be at $y = 0$. This second constraint is ensured by computing the position of both masses, positioned at the end of each link, and imposing their y-coordinate to be greater than 0. The convexity of this problem is analyzed and, as its implementation in code is presented below.

```
# Lagrange objective
ocp.add_objective(ocp.T)

# Motion constraints
epsilon = 1e-2
# acceleration limits
ocp.subject_to(theta1_ddot <= 10)
ocp.subject_to(theta1_ddot >= -10)
ocp.subject_to(theta2_ddot <= 10)
ocp.subject_to(theta2_ddot >= -10)

# Avoid collision with the ground
y_position_link1 = l1 * sin(theta1)
y_position_end_effector = l1 * sin(theta1) + l2 * sin(theta1 + theta2)
ocp.subject_to(y_position_link1 > 0)
ocp.subject_to(y_position_end_effector > 0)
```

The obtained evolution of the joint accelerations, velocities and positions over time, is presented in Figure 1.4. It can be seen that the robot arm reaches the desired final position in $0.75s$ remaining compliant with the constraints along the whole trajectory.



(a) Joint Accelerations (b) Joint Velocities (c) Joint Positions

Figure 1.4: Time-Optimal optimization problem(shooting interval = 100)

Convexity analysis

The system dynamics are the same as those presented in Section 1.1.1, and they can be expressed linearly in state-space form. Two new nonlinear constraints were added in order to avoid gorund collision. The optimization problem can be formulated as:

$$
\begin{aligned}
\min \quad & T, \\
\text{subject to} \quad & \theta_{\text{initial}} = \begin{bmatrix} 0.1 \\ 0.5 \end{bmatrix} \text{rad}, \quad \theta_{\text{terminal}} = \begin{bmatrix} 1.0 \\ 1.9 \end{bmatrix} \text{rad}, \\
& \dot{\theta}_{\text{initial}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{rad/s}, \quad \dot{\theta}_{\text{terminal}} = \begin{bmatrix} 0.0 \\ 0.0 \end{bmatrix} \text{rad/s}, \\
& -10\,\text{rad/s}^2 \leq \ddot{\theta}_r \leq 10\,\text{rad/s}^2, \quad |\theta_2| < \pi, \\
& -l_1 \sin\theta_1 < 0, \quad -l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) < 0, \\
& \text{for } t \in [0, T]
\end{aligned}
$$

In order for an optimization problem to be convex, it is required that the feasible region identified by its constraints is convex. As shown in Figure 1.5, this is not true. As a consequence, the time optimal control problem is not convex.

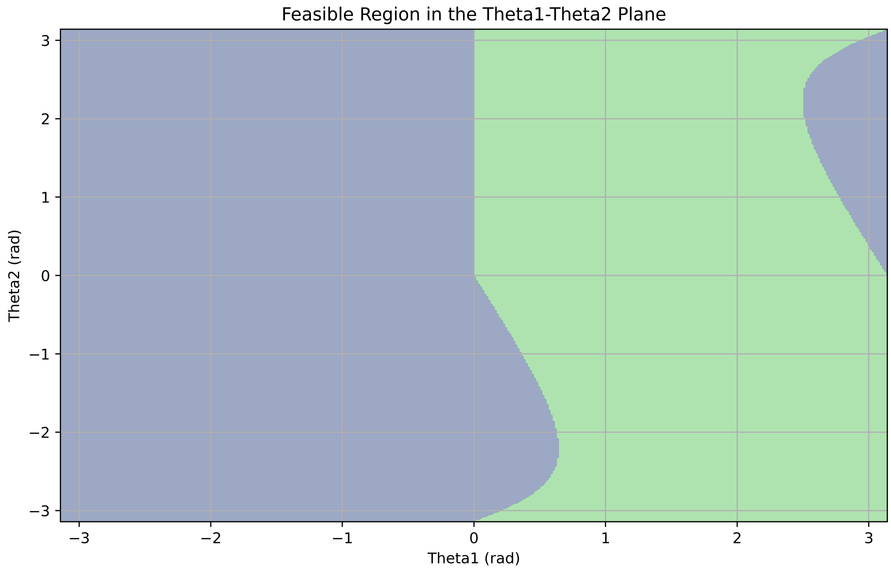


Figure 1.5: Feasible region identified by the nonlinear constraints

1.4 Question 4 - Forward Kinematics of the Robot

To derive the forward kinematics for the given robot, two transformation matrices were defined in order to describe the relative positions and orientations of the links. The first transformation matrix, ${}^0T_1$, represents the pose of the first link with respect to the base frame. It is given by:

$$
{}^0T_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & l_1\cos(\theta_1) \\ \sin(\theta_1) & \cos(\theta_1) & l_1\sin(\theta_1) \\ 0 & 0 & 1 \end{bmatrix}
$$

The transformation matrix ${}^1T_2$, representing the pose of the second link relative to the first link, is given by:

$$
{}^1T_2 = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & l_2\cos(\theta_2) \\ \sin(\theta_2) & \cos(\theta_2) & l_2\sin(\theta_2) \\ 0 & 0 & 1 \end{bmatrix}
$$

The overall transformation matrix ${}^0T_2$, describing the pose of the end effector relative to the base frame, is obtained by multiplying these two matrices:

$$^{0}T_2 = {}^{0}T_1{}^{1}T_2 = \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & l_1\cos(\theta_1)+l_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & l_1\sin(\theta_1)+l_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 \end{bmatrix}$$

From ${}^{0}T_2$, the position and orientation of the end effector can be derived as follows:

- the x-coordinate of the end effector: $x_{\text{ee}} = {}^{0}T_2[0,2] = l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2)$,
- the y-coordinate of the end effector: $y_{\text{ee}} = {}^{0}T_2[1,2] = l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2)$,
- the yaw angle (orientation) of the end effector: $yaw_{\text{ee}} = arctg\Big({}^{0}T_2[1,0], {}^{0}T_2[0,0]\Big)$

1.5 Question 5 - Optimal Control for Torque-Controlled Robot with End-Effector Constraints

An optimal control problem is formulated for the torque-controlled robot to minimize the mean-square joint torques required to move the robot's end effector from the initial pose $[x_{\text{ee}}, y_{\text{ee}}, yaw_{\text{ee}}]^T = [0.8\text{ m}, 0.0\text{ m}, 0.0\text{ rad}]^T$ to the terminal pose $[0.4\text{ m}, 0.4\text{ m}, 1.8\text{ rad}]^T$ over a duration of 2 seconds. The optimization problem includes constraints to ensure that the robot starts and ends its motion with zero joint velocities and avoids self-collision as well as collision with the ground, considered to be at $y = 0$. The constraints' implementation in code is:

```
# Motion constraints - Initial constraints
ee_pos = forward_kinematics(theta1, theta2)
ocp.subject_to(ocp.at_t0(ee_pos[0]) == 0.8) # x_ee = 0.8
ocp.subject_to(ocp.at_t0(ee_pos[1]) == 0.)  # y_ee = 0.0
ocp.subject_to(ocp.at_t0(ee_pos[2]) == 0.)  # yaw_ee = 0.0

ocp.subject_to(ocp.at_t0(theta1_dot) == 0.)
ocp.subject_to(ocp.at_t0(theta2_dot) == 0.)

# End constraints
ocp.subject_to(ocp.at_tf(ee_pos[0]) == 0.4) # x_ee = 0.4
ocp.subject_to(ocp.at_tf(ee_pos[1]) == 0.4) # y_ee = 0.4
ocp.subject_to(ocp.at_tf(ee_pos[2]) == 1.8) # yaw_ee = 1.8

ocp.subject_to(ocp.at_tf(theta1_dot)==0.)
ocp.subject_to(ocp.at_tf(theta2_dot)==0.)
```

1.5.1 Infeasibility analysis

The problem of controlling a 2-link robotic arm to move from an initial position to an end position is infeasible due to conflicting constraints on its geometry and motion.

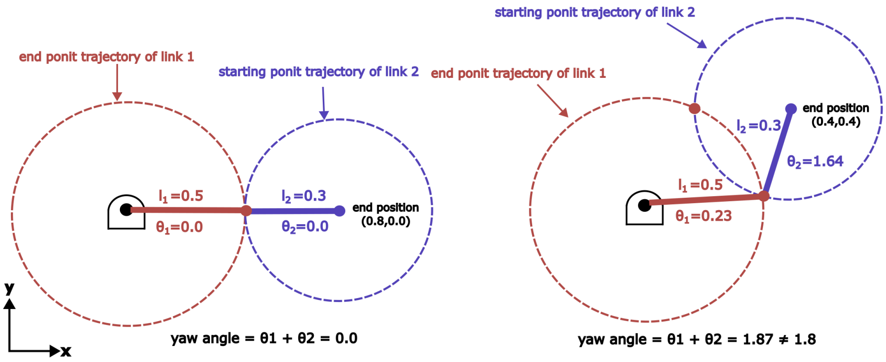


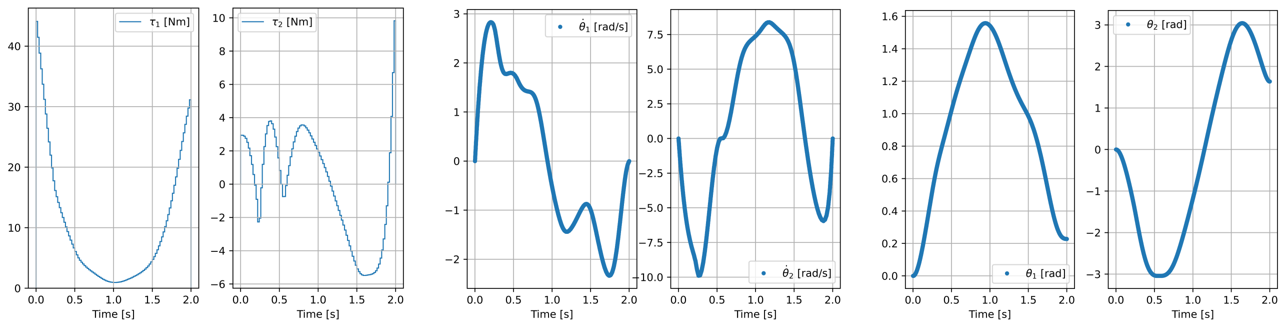
Figure 1.6: End-effector pose analysis

As shown in Figure 1.6, the initial position of the end-effector is specified to be at $[0.8\,\mathrm{m}, 0.0\,\mathrm{m}]^T$, with an additional constraint that the yaw angle is zero, given that the total length of the arm is $l_1 + l_2 = 0.8$ m, the only possible configuration is when both links are aligned, that is, the arm is fully extended in the x-direction.

For the final pose, when the end-effector reaches $[0.4\,\mathrm{m}, 0.4\,\mathrm{m}]^T$, the required joint angles can be determined using inverse kinematics. Specifically, solving for the joint angles to achieve this position reveals that the yaw angle is approximately 1.87 rad ($\theta_1 + \theta_2 = 0.23 + 1.64 = 1.87$ rad), rather than 1.8 rad. Therefore, the constraint on the final yaw angle of the end-effector makes the desired configuration geometrically infeasible.

1.5.2 Relaxed End-Effector Constraints

As previously mentioned, for the problem to be feasible, it is necessary to remove the constraint on the end-effector's final yaw angle. By solving the relaxed version of the problem, we obtain the following evolution of joint positions, velocities, and torques, Figure 1.7.



(a) Torques (b) Joint Velocities (c) Joint Positions

Figure 1.7: Torque-Controlled Robot with Relaxed End-Effector Constraints