# Optimal Control Problem D-T

Giorgio Medico

25 October 2025

## 1 Problem Formulation

The goal is to find the optimal state sequence $X = \{x_1, \ldots, x_T\}$ and control sequence $U = \{u_0, \ldots, u_{T-1}\}$ that minimize a cost function subject to the system's dynamics. This is a nonlinear optimal control problem, which can be stated in the following general form:

$$\min_{X,U} \quad \sum_{t=0}^{T-1} L(x_t, u_t) + L_f(x_T) \tag{1}$$

$$\begin{aligned}
\text{s.t.} \quad x_{t+1} &= f(x_t, u_t), && t = 0, \ldots, T-1 \\
x_0 &= x^0 \\
u_{min} \leq u_t &\leq u_{max}, && t = 0, \ldots, T-1 \\
q_{min} \leq q_t &\leq q_{max}, && t = 1, \ldots, T \\
\dot{q}_{min} \leq \dot{q}_t &\leq \dot{q}_{max}, && t = 1, \ldots, T \\
\overline{\eta}_i(x_t) &\leq \overline{\eta}_{lim}, && t = 1, \ldots, T, \quad i \in \{IC, EC\}
\end{aligned}$$

(where $q_t$ and $\dot{q}_t$ are sub-vectors of the state $x_t$)

### 1.1 Problem-Specific Definitions

The general terms $L, L_f, f, x, u,$ and $x^0$ are defined for our specific robot-and-sloshing problem as follows:

- **Time Horizon ($T$):** The number of discrete time steps, $T = t_{end}/dt$.

- **State Vector ($x_t$):** The $\mathbb{R}^{20}$ state at time $t$:

$$x_t = [q_t^T, \dot{q}_t^T, x_{IC,t}, y_{IC,t}, \dot{x}_{IC,t}, \dot{y}_{IC,t}, x_{EC,t}, y_{EC,t}, \dot{x}_{EC,t}, \dot{y}_{EC,t}]^T$$

(where $q_t$ and $\dot{q}_t$ are respectively joint position and velocity)

- **Control Vector ($u_t$):** The $\mathbb{R}^6$ joint torque vector at time $t$:

$$u_t = \tau_t$$

- **Stage Cost ($L$):** This includes pose tracking, sloshing suppression (with $\overline{\eta}^{des} = 0$), and control effort (with $u^{des} = 0$).

$$L(x_t, u_t) = \frac{1}{2}\|p_{EE}(q_t) - p_{ref,t}\|^2_{Q_p} + \frac{1}{2}\sum_{i \in \{IC,EC\}} \|\overline{\eta}_i(x_t)\|^2_{Q_\eta} + \frac{1}{2}\|u_t\|^2_R$$

Here, $p_{EE}(q_t)$ is the robot's **Forward Kinematics** function. The sloshing height $\overline{\eta}_i(x_t)$ is calculated from the state variables (using the paper's Eq. (18)):

$$\overline{\eta}_i(x_t) = \frac{\xi_{11}^2 h m_1}{m_F R}\sqrt{x_{i,t}^2 + y_{i,t}^2}$$

(where $x_{i,t}$ and $y_{i,t}$ are components of the state $x_t$).

- **Terminal Cost ($L_f$):** The cost on the final state, $x_T$. This heavily penalizes final pose error, final velocity, and residual sloshing.

$$L_f(x_T) = \frac{1}{2}\|p_{EE}(q_T) - p_{ref,T}\|^2_{P_p} + \frac{1}{2}\sum_{i \in \{IC,EC\}} \|\overline{\eta}_i(x_T)\|^2_{P_\eta} + \frac{1}{2}\|\dot{q}_T\|^2_{P_{\dot{q}}}$$

(Note: $\dot{q}_T$ is a sub-vector of $x_T$).

- **Discrete Dynamics ($f$):** The function that maps the current state and control to the next state.

$$x_{t+1} = f(x_t, u_t) = \mathrm{RK4}(\dot{x}, x_t, u_t, \Delta t)$$

This represents one step of a numerical integrator (like 4th-order Runge-Kutta). It is applied to the continuous dynamics function $\dot{x} = f_{cont}(x, u)$, which is the core of the coupled model:

To compute $\dot{x} = f_{cont}(x, u)$, the following steps are performed:

1. **Robot Dynamics:** Calculate joint accelerations $\ddot{q}$ from the current state $(q, \dot{q})$ and torque input $u = \tau$:

$$\ddot{q} = M(q)^{-1}(\tau - C(q, \dot{q})\dot{q} - G(q))$$

2. **EE Kinematics:** Calculate the EE (tray) linear and angular velocities and accelerations from the robot's joint state:

$$[\dot{r}^T, \omega^T]^T = J(q)\dot{q}$$
$$[\ddot{r}^T, \dot{\omega}^T]^T = \dot{J}(q, \dot{q})\dot{q} + J(q)\ddot{q}$$

3. **Container Acceleration:** Calculate the acceleration $\ddot{r}_i$ for each container (IC, EC) using the EE motion and relative position, as defined in the paper's Eq. (10) :

$$\ddot{r}_i = \ddot{r} + \dot{\omega} \times d_i + \omega \times (\omega \times d_i)$$

4. **Sloshing Dynamics:** Use $\ddot{r}_i$, $\omega$ (for $\dot{\theta}$), and $\dot{\omega}$ (for $\ddot{\theta}$) as inputs to the sloshing EOMs to solve for the sloshing accelerations $(\ddot{x}_n, \ddot{y}_n)$. From the paper's Eq. (16) :

$$\begin{bmatrix} 1 + P_n^2 x_n^2 & P_n^2 x_n y_n \\ P_n^2 x_n y_n & 1 + P_n^2 y_n^2 \end{bmatrix} \begin{bmatrix} \ddot{x}_n \\ \ddot{y}_n \end{bmatrix} = \begin{bmatrix} a_{i,n} \\ b_{i,n} \end{bmatrix}$$

   where $a_{i,n}$ and $b_{i,n}$ are functions of the state and the calculated accelerations (from paper's Eq. 17 ).

5. **State Derivative Vector:** Construct the full state derivative $\dot{x}$:

$$\dot{x} = [\dot{q}^T, \ddot{q}^T, \dot{x}_{IC}, \dot{y}_{IC}, \ddot{x}_{IC}, \ddot{y}_{IC}, \dot{x}_{EC}, \dot{y}_{EC}, \ddot{x}_{EC}, \ddot{y}_{EC}]^T$$

The RK4 integrator then uses this $\dot{x}$ vector to compute $x_{t+1}$.

- **Initial State ($x^0$):**

$$x^0 = [q_0^T, 0_6^T, 0_8^T]^T$$

  where $q_0 = \text{InverseKinematics}(p_{ref,0})$.

- **Path Constraints:** These are the physical limits on the system:
  - **Torque Limits:** $u_{min} = \tau_{min}$ and $u_{max} = \tau_{max}$.
  - **Joint Position Limits:** $q_{min}$ and $q_{max}$.
  - **Joint Velocity Limits:** $\dot{q}_{min}$ and $\dot{q}_{max}$.
  - **Sloshing Height Limit:** $\overline{\eta}_{lim}$, a positive scalar value.