Mathematic Modeling of Optimal Control

1 Introduction

The equation of motion for the constrained robotic manipulator with n degrees of freedom, considering the contact force and the constraints, is given in the joint space as follows:

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + G(\dot{\boldsymbol{q}}) = \boldsymbol{u} \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^{n\times 1}$ denotes the joint angles or link displacements of the manipulator, $M(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the robot inertia matrix which is symmetric and positive definite, $C(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ contains the centripetal and Coriolis terms and $G(\boldsymbol{q}) \in \mathbb{R}^{n\times 1}$ are the gravity terms and $\boldsymbol{u} \in \mathbb{R}^{n\times 1}$ denotes the torque or the force.

Let's define $x_1 = q$, $x_2 = \dot{q}$, $\boldsymbol{x} = \binom{\boldsymbol{q}}{\dot{\boldsymbol{q}}}$, and we can rewrite everything as follows:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = -M^{-1}C\boldsymbol{x}_2 + M^{-1}(\boldsymbol{u} - G) \end{cases} \tag{2}$$

2 Optimal Control

The optimal control will be designed for the trajectory tracking case where the objective is to obtain that the tracking error of the states converges to zero, that is, the states of the robotic arm converge to the desired trajectories.
We'll be optimising for $\hat{\boldsymbol{u}}$, defined as:

$$\hat{\boldsymbol{u}} = \boldsymbol{u} - G(\boldsymbol{q}) \tag{3}$$

There are a couple of reason we decided to optimize for $\hat{\boldsymbol{u}}$ and not $\boldsymbol{u}$. The main one is that we noticed that, even if the mathematical problem is *exactly* the same, the initial guess of $[0, ..., 0]$ is really better suited for $\hat{\boldsymbol{u}}$ rather than $\boldsymbol{u}$ and this result in a considerably faster convergence of the optimal problem.
Using $\hat{\boldsymbol{u}}$ we managed to increase the performance (in terms of time) by 2 order of magnitude.

Reference trajectory Let's say we wanna reach a trajectory specified as $\boldsymbol{q}_d(t)$ and we define $x_d = \binom{q_d}{\dot{q}_d}$. So our tracking error $\tilde{x}$ will be:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}_d \tag{4}$$

With that in mind we can write our cost for this system as follows:

$$J = \tilde{\boldsymbol{x}_{\boldsymbol{f}}}^T H \tilde{\boldsymbol{x}_{\boldsymbol{f}}} + \int_{t_i}^{t_f} \left(\tilde{\boldsymbol{x}}^T Q \tilde{\boldsymbol{x}} + \boldsymbol{u}^T R \boldsymbol{u}\right) \tag{5}$$

The Problem itself We can now rewind what we've said until now and define our problem in terms of differential equations, keeping in mind that we're losing the final position term in the cost function:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = -M^{-1}C\boldsymbol{x}_2 + M^{-1}(\hat{\boldsymbol{u}}) \\ \dot{J} = (\boldsymbol{x} - \boldsymbol{x}_d)^T Q(\boldsymbol{x} - \boldsymbol{x}_d) + \boldsymbol{u}^T R\hat{\boldsymbol{u}} \end{cases} \implies \begin{pmatrix} \dot{\boldsymbol{x}}_{\boldsymbol{1}} \\ \dot{\boldsymbol{x}}_{\boldsymbol{2}} \\ \dot{J} \end{pmatrix} = f\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \hat{\boldsymbol{u}}\right) \tag{6}$$

3 What we do

We solve symbolically the differential equation above with RK4 for a single time step, so we get the following function F:

$$\begin{pmatrix} \boldsymbol{x}_f \\ J_f \end{pmatrix} = F(\boldsymbol{x}_i, J_i) \tag{7}$$

With that, we can integrate from the initial conditions up to the last time step and get $J(u_0, u_1, ..., u_{N-1})$. In the same integration loop we also store $x_k(u) \forall k \in \{0, ..., N\}$ in order to set boundaries in the minimization problem, according to the URDF joint limits.

Once we get that, we start an optimization algorithm that minimize J and respect the boundaries we defined before. The result will be $\hat{\boldsymbol{u}}_{opt}$.
From that, we can retrive also the optimal trajectory by simply inserting the optimal u in the F function we defined before.