

# OCP/MPC Workshop 2024

## Impact tutorial: part 2

Alejandro Astudillo Vigoya, Wilm Decré, Louis Callens, Alex Gonzalez García, Dries Dirckx

July 17, 2024

### 1 Point-to-point MPC for a robot manipulator

In this assignment, you will create a model predictive controller for a point-to-point motion of a robot manipulator (Franka Panda).

Given:

- $\mathbf{x}_{\text{current}} = \begin{bmatrix} \mathbf{q}_0 \\ \dot{\mathbf{q}}_0 \end{bmatrix}$  : Initial state of the robot
- $\mathbf{p}_f$  : Desired position to reach in Cartesian coordinates

We want to minimize the position error along and at the end of the horizon  $T$  and regularize the joint variables over the trajectory. The state vector is  $\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}$  and the control input is  $\ddot{\mathbf{q}}$ . The optimization problem can be formulated as:

$$\underset{\mathbf{x}, \mathbf{u}}{\text{minimize}} \quad \int_0^T w_1 (\|\mathbf{p}(\mathbf{q}(t)) - \mathbf{p}_f\|^2 + w_2 \|\mathbf{q}(t)\|^2 + w_3 \|\dot{\mathbf{q}}(t)\|^2 + w_4 \|\ddot{\mathbf{q}}(t)\|^2) dt + w_5 \|\mathbf{p}(\mathbf{q}(T)) - \mathbf{p}_f\|^2 \quad (1a)$$

$$\text{subject to} \quad \mathbf{x}(0) = \mathbf{x}_{\text{current}}, \quad (1b)$$

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \quad (1c)$$

$$\mathbf{q}_{\min} \leq \mathbf{q}(t) \leq \mathbf{q}_{\max}, \quad \forall t \in [0, T], \quad (1d)$$

$$\dot{\mathbf{q}}_{\min} \leq \dot{\mathbf{q}}(t) \leq \dot{\mathbf{q}}_{\max}, \quad \forall t \in [0, T], \quad (1e)$$

$$\ddot{\mathbf{q}}_{\min} \leq \ddot{\mathbf{q}}(t) \leq \ddot{\mathbf{q}}_{\max}, \quad \forall t \in [0, T], \quad (1f)$$

$$\dot{\mathbf{q}}(T) = \mathbf{0} \quad (1g)$$

where:

- $\mathbf{p}(\mathbf{q})$  is the forward kinematics mapping from joint space to Cartesian space.
- $w_i$  is a weighting factor for the control effort.
- $\mathbf{q}_{\min}$  and  $\mathbf{q}_{\max}$  are the lower and upper bounds on joint positions.
- $\dot{\mathbf{q}}_{\min}$  and  $\dot{\mathbf{q}}_{\max}$  are the lower and upper bounds on joint velocities.
- $\ddot{\mathbf{q}}_{\min}$  and  $\ddot{\mathbf{q}}_{\max}$  are the lower and upper bounds on joint accelerations.

## 2 Getting started

This exercise uses the robot manipulator of a Franka Panda robot provided in the Robotics Toolbox for Python. This library can be installed by running

```
pip install roboticstoolbox-python
```

To generate robot dynamics and kinematics, we use the interface between the state-of-the-art rigid-body dynamics library Pinocchio. You can install Pinocchio by running:

```
conda install conda-forge:pinocchio
```

and the interface by executing:

```
pip install git+https://gitlab.kuleuven.be/meco-software/robot-models-meco.git@pin3-devel
```

All these libraries are already included in the provided `mecoverse-robotics` conda environment, which you can create by running

```
conda env create -f mecoverse_robotics_environment.yml
```

Once you have your environment ready, go to the directory `Tutorials/3_impact/part-2`. You should work on the `controller.py` script to specify the optimal control problem underpinning the model predictive controller for the task at hand.