Exercise 2 - Swing-up control of the Furuta pendulum with NMPC

In this exercise, the swing-up control of the Furuta pendulum is considered. The Furuta pendulum is a nonlinear system whose system dynamics and stabilization control have already been considered in Exercise 1. The aim of this exercise is to use `CasADi` to solve a parametrized discrete-time optimal control problem of the form

$$\begin{aligned} \min_{x_0,u_0,\ldots,u_{N-1},x_N} & \sum_{k=0}^{N-1} L(x_k,u_k) + E(x_N) \\ \text{s.t.} \quad x_{k+1} - f(x_k,u_k) &= 0, \quad k = 0,\ldots,N-1, \\ h(x_k,u_k) &\leq 0, \quad k = 0,\ldots,N-1, \\ r(x_0,x_N,p) &= 0 \end{aligned}$$

in an NMPC scheme – to solve it again and again depending on the current state of the system during the upswing process.

1. Use the following specifications to write down the optimal control problem shown above in a sequential approach on a sheet of paper: $N = 10$ steps, the cost functions $L(x,u) = x^\top Q x + Ru^2$, $Q = \mathrm{diag}(50,500,1,1)$, $R = 1000$, $E(x) = L(x,0)$, and the initial condition $r(x_0,x_n,p) = x_0 - p$ should be used. Initially, no inequality constraints should be used. In addition, the discrete-time dynamics $f(x,u)$ is defined in such a way that the explicit RK4 method with step size $h = 0.005$ s is used to approximate the continuous-time dynamics within one step in 10 immediate steps. Therefore, the same manipulated variable should be applied over 50 ms. What is the dimension of the decision variable of the optimisation problem?

2. Define the optimization problem in the `MATLAB` script using the `Opti` environment, which you have already become familiar with in previous exercises. A parameter can be defined within the `Opti` environment using the function `opti.parameter()`.

3. Get an overview of the functions that are called within the simulation of the controlled system (second `for` loop). Complete these as described.

4. Test the implemented controller.

5. Add the constraints $-0.5 \leq u_k \leq 0.5$, $k = 0,\ldots,N-1$ and test the controller again.

6. Vary the specifications defined in 1. and analyse their influence on the behaviour of the controlled system.

7. Bonus: Design a NMPC controller employing `acados` `https://docs.acados.org/`.