Institut für Regelungstechnik
Leibniz Universität Hannover

Leibniz Universität Hannover

Institut für Regelungstechnik
Prof. Dr.-Ing. Matthias Müller

MPC – Exercise 2

The main goals of this exercise are:

- Implement a Quasi-infinite horizon MPC scheme.
- Compute a suitable terminal set and terminal cost to ensure closed-loop stability.

Consider the continuous-time constrained optimal control problem (see, e.g., [1])

$$\begin{aligned}
&\underset{\bar{u}(\cdot;t)}{\text{minimize}} && \int_t^{t+T} \|\bar{x}(\tau;t)\|_Q^2 + \|\bar{u}(\tau;t)\|_R^2 \, \mathrm{d}\tau + \|\bar{x}(t+T;t)\|_P^2 && (1a)\\
&\text{subject to} && \dot{\bar{x}} = f(\bar{x}, \bar{u}) && (1b)\\
& && \bar{x}(t;t) = x(t) && (1c)\\
& && \bar{u}(\tau;t) \in \mathcal{U} && (1d)\\
& && \bar{x}(t+T;t) \in \Omega_\alpha, && (1e)
\end{aligned}$$

where $\|x\|_Q = x^\top Q x$, with the same nonlinear system as in the previous exercise

$$\begin{aligned}
\dot{x}_1 &= x_2 + u\,(\mu + (1-\mu)\,x_1)\\
\dot{x}_2 &= x_1 + u\,(\mu - 4\,(1-\mu)\,x_2)
\end{aligned} \tag{2}$$

and the constraint set

$$\mathcal{U} := \{u \in \mathbb{R} : |u| \leq 2\}.$$

In the following, we set $\mu = 0.5$. Moreover, the weighting matrices are given by

$$Q = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \text{ and } R = 1.0 \ .$$

The initial state is $x(0) = \begin{bmatrix} -0.4 \\ -0.5 \end{bmatrix}$ and we choose a prediction horizon $T = 1.5$.

Problem 1 In a first consideration, neglect the terminal cost in (1a) and the terminal constraint (1e). Simulate the closed loop. In order to do so, set the simulation time to 2 and discretize the optimal control problem (1) using the RK4 method.
Hint: You can modify your solution from Exercise 2 to solve this problem.
What happens? Why?

Problem 2 Derive a suitable terminal set and a terminal cost for this problem using the file `computeAlpha.m`, according to the procedure presented in the lecture on *quasi-infinite horizon MPC*. As a local auxiliary controller, the corresponding LQR controller for the system linearized at the origin should be used. The parameter κ is chosen to be 0.95. Compute the parameters P and α for the terminal set

$$\Omega_\alpha = \{x \in \mathbb{R}^2 : x^T P x \leq \alpha\}.$$

For this,

a) Linearize the system (2) at the origin and implemented the linearized matrices

$$A = \left.\frac{\partial f(x,u)}{\partial x}\right|_{(u,x)=0}, \qquad B = \left.\frac{\partial f(x,u)}{\partial u}\right|_{(u,x)=0}.$$

Appelstr. 11, D-30167 Hannover Tel.: +49-511-762-4512, Fax.: +49-511-762-4536 E-Mail: office@irt.uni-hannover.de

b) Compute the corresponding LQR controller.
Hint: Be careful when using MATLAB's `lqr` command. The `lqr` command assumes $A - BK$ for the closed-loop system (compared to $A + BK$ in the lecture).

c) Solve the Lyapunov equation from the lecture to get P.
Hint: Use MATLAB's `lyap` command. You can type `help lyap` in MATLAB's command window for a short documentation.

d) Steps 3 and 4 are already implemented. The function `FcnL_phi.m` calculates L_ϕ from the lecture.

Problem 3 Add the terminal region constraint (1e) and terminal cost to your simulation.

a) Simulate the closed loop.

b) Change the initial state to $x_0 = [-0.7, -0.8]^\top$. What happens?

c) How would you change the MPC scheme to stabilize this initial condition (without changing the terminal region, cost, and constraints)? Change your implementation accordingly and simulate the closed loop.

Problem 4 Derive a less conservative terminal set and a terminal cost for this problem according to the alternative procedure presented in the same lecture.

a) Implement the alternative step 4 in `computeAlpha.m`.
Hint: Solve the optimization problem in the alternative procedure by using `fmincon`. You can type `doc fmincon` in MATHALB's command window for a documentation of `fmincon` and some examples. The function $\phi(x) = f(x,Kx) - A_K x$ is already implemented (compare line 91 in `computeAlpha.m`) and can be called by writing `phi(x)`. Maximization of $L(x)$ can be done by minimizing $-L(x)$. Use bisection similar to the way it is already implemented in the previous step 4. The inner `for`-loop is necessary to make sure the 'correct' initialization is chosen for `fmincon` (i.e., in order not to get stuck in a local minimum). Set the variable `alternative_procedure` to `true`.

b) Implement the alternative terminal set constraint and terminal cost in your algorithm in the file `MPC_Exercise3.m`.

c) Simulate the closed loop again with initial condition $x(0) = \begin{bmatrix} -0.7 & -0.8 \end{bmatrix}^\top$ and $T = 1.5$ and compare the results to Problem 3b.

Literatur

[1] Hong Chen and Frank Allgöwer. A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability. *Automatica*, 34(10):1205–1217, 1998.