

Model Predictive Control : Exercise 6 - Solutions

Prob 1 | Solving Explicit MPC using parametric LCPs

Consider the discrete-time linear time-invariant system defined by the dynamics

$$x^+ = 2x + u - 1$$

with constraints

$$U = \{u \mid 0 \leq u \leq 2\}$$

We formulate the following MPC problem with horizon $N = 1$:

$$\begin{aligned} f(x) = \min \quad & x^2 + u^2 + (x^+)^2 \\ \text{s.t.} \quad & x^+ = 2x + u - 1 \\ & 0 \leq u \leq 2 \end{aligned}$$

Your goal is to calculate the explicit solution $f^*(x)$ of the parametric program and the corresponding explicit control policy $u^*(x)$.

Tasks:

- To simplify the problem, eliminate the decision variable x^+ .

After elimination, we get the following problem:

$$\begin{aligned} f(x) = \min \quad & 5x^2 - 4x + 4xu + 2u^2 - 2u + 1 \\ & u \geq 0, \quad 2 - u \geq 0 \end{aligned}$$

- Write down the Lagrangian function $\mathcal{L}(x, u, \lambda, \nu)$ where λ corresponds to the constraint $0 \leq u$ and ν corresponds to the constraint $u \leq 2$.

$$\mathcal{L}(x, u, \lambda, \nu) = 5x^2 - 4x + 4xu + 2u^2 - 2u + 1 - \lambda u - \nu(2 - u)$$

- Write down the KKT conditions (stationarity, primal/dual feasibility, complementarity).

The KKT conditions are given by:

Stationarity

$$[\nabla_u \mathcal{L}(x, u, \lambda, \nu) = 4u + 4x - \lambda + \nu - 2 = 0,$$

Primal feasibility

$$[s = 2 - u, \quad u, s \geq 0,$$

Dual feasibility

$$[\lambda, \nu \geq 0,$$

Complementarity

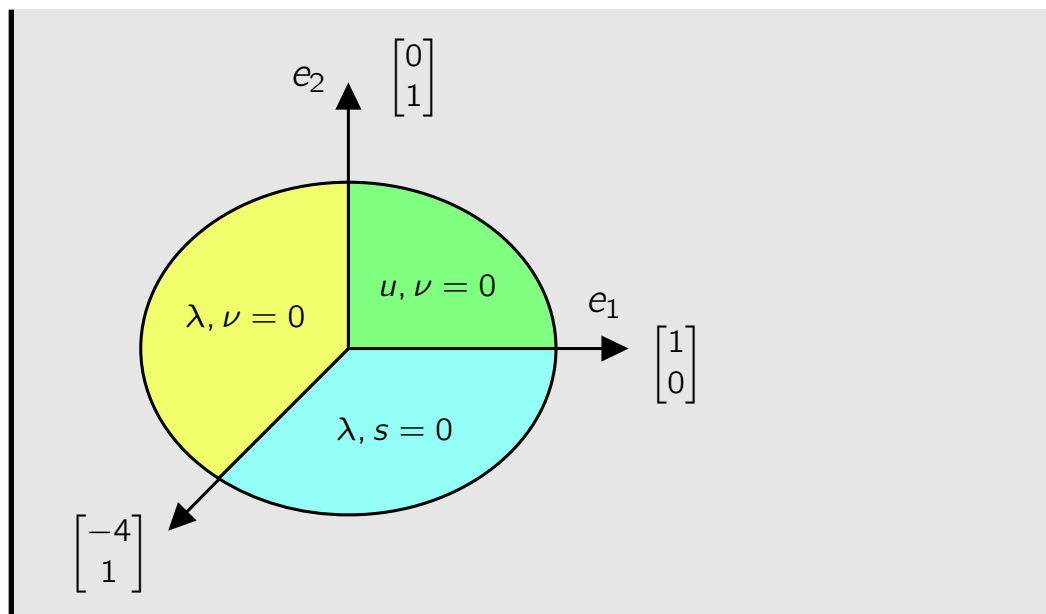
$$[\lambda u = 0, \quad \nu s = 0,$$

- Give matrices M , Q and vector q such that the optimal solution of the problem is a linear transform of the solution $y(x)$ to the following parametric LCP:

$$w - My = Qx + q \quad w, y \geq 0 \quad w^T y = 0$$

$$\underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_w \underbrace{\begin{bmatrix} \lambda \\ s \end{bmatrix}}_y - \underbrace{\begin{bmatrix} 4 & 1 \\ -1 & 0 \end{bmatrix}}_M \underbrace{\begin{bmatrix} u \\ \nu \end{bmatrix}}_y = \underbrace{\begin{bmatrix} 4 \\ 0 \end{bmatrix}}_Q x + \underbrace{\begin{bmatrix} -2 \\ 2 \end{bmatrix}}_q, \quad \underbrace{\begin{bmatrix} \lambda \\ s \end{bmatrix}}^T \underbrace{\begin{bmatrix} u \\ \nu \end{bmatrix}} = 0$$

- Draw the complementarity cones of the pLCP.



- Compute the optimal value function $f^*(x)$ and its corresponding control policy $u^*(x)$.

We are consider 3 different cases:

$$- \lambda = 0, s = 0$$

$$\begin{aligned} \begin{bmatrix} u \\ \nu \end{bmatrix} &= - \begin{bmatrix} 4 & 1 \\ -1 & 0 \end{bmatrix}^{-1} \left(\begin{bmatrix} 4 \\ 0 \end{bmatrix} x + \begin{bmatrix} -2 \\ 2 \end{bmatrix} \right) = - \begin{bmatrix} 0 & -1 \\ 1 & 4 \end{bmatrix}^{-1} \left(\begin{bmatrix} 4 \\ 0 \end{bmatrix} x + \begin{bmatrix} -2 \\ 2 \end{bmatrix} \right) \\ &= \begin{bmatrix} 0 \\ -4 \end{bmatrix} x + \begin{bmatrix} 2 \\ -6 \end{bmatrix} \geq 0 \implies x \leq -1.5 \end{aligned}$$

$$u^*(x) = 2, \quad f^*(x) = 5x^2 + 4x + 5$$

$$- \lambda = 0, \nu = 0$$

$$\begin{aligned} \begin{bmatrix} u \\ s \end{bmatrix} &= \begin{bmatrix} -4 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \left(\begin{bmatrix} 4 \\ 0 \end{bmatrix} x + \begin{bmatrix} -2 \\ 2 \end{bmatrix} \right) = -\frac{1}{4} \begin{bmatrix} 1 & 0 \\ -1 & -4 \end{bmatrix}^{-1} \left(\begin{bmatrix} 4 \\ 0 \end{bmatrix} x + \begin{bmatrix} -2 \\ 2 \end{bmatrix} \right) \\ &= \begin{bmatrix} -1 \\ 1 \end{bmatrix} x + \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} \geq 0 \implies -1.5 \leq x \leq 0.5 \end{aligned}$$

$$u^*(x) = 0.5 - x, \quad f^*(x) = 3x^2 - 2x + 0.5$$

$$- u = 0, \nu = 0$$

$$\begin{bmatrix} \lambda \\ s \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \end{bmatrix} x + \begin{bmatrix} -2 \\ 2 \end{bmatrix} \geq 0 \implies 0.5 \leq x$$

$$u^*(x) = 0, \quad f^*(x) = 5x^2 - 4x + 1$$

Hence, this gives us the solutions:

$$u^*(x) = \begin{cases} 0, & 0.5 \leq x \\ 0.5 - x, & -1.5 \leq x \leq 0.5 \\ 2, & x \leq -1.5 \end{cases}$$

$$f^*(x) = \begin{cases} 5x^2 + 4x + 5, & 0.5 \leq x \\ 3x^2 - 2x + 0.5, & -1.5 \leq x \leq 0.5 \\ 5x^2 - 4x + 1, & x \leq -1.5 \end{cases}$$

- Use Matlab to estimate $u^*(x)$ and $f^*(x)$ by solving the optimization problem for a number of different values of x and compare this to your parametric solution.

Prob 2 | Implement explicit MPC using MPT3

We revisit the MPC problem from exercise 4, where we considered the discrete-time linear time-invariant system defined by

$$x^+ = \begin{bmatrix} 0.9752 & 1.4544 \\ -0.0327 & 0.9315 \end{bmatrix} x + \begin{bmatrix} 0.0248 \\ 0.0327 \end{bmatrix} u$$

with constraints

$$X = \{x \mid |x_1| \leq 5, |x_2| \leq 0.2\} \quad U = \{u \mid |u| \leq 1.75\}$$

This is a second-order system with a natural frequency of $0.15r/s$, a damping ratio of $\zeta = 0.1$ which has been discretized at $1.5r/s$. The first state is the position, and the second is velocity.

Your goal is to implement an explicit MPC controller for this system with a horizon of $N = 10$ and a stage cost given by $l(x, u) := 10x^T x + u^T u$ using the MPT3 toolbox.

Tasks:

- Define your MPC problem using MPT3. You can proceed as follows:
 - Define the system `sys = LTISystem('A', A, 'B', B)`
 - Define the constraints on the signals by setting the values `sys.x.max = ..., sys.x.min = ..., etc`
 - Define the stage costs by setting the penalty terms for x and u , e.g., `sys.x.penalty = QuadFunction(Q)`
 - Extract desired sets and weights with `sys.LQRGain`, `sys.LQRPenalty.weight` and `sys.LQRSet`,
 - Set the terminal cost and terminal set with `model.x.with('terminalPenalty')`, `model.x.terminalPenalty = QuadFunction(Qf)` and `model.x.with('terminalSet')`, `model.x.terminalSet = Xf`,
 - Define the MPC controller with `controller = MPCController(sys, N)`.
- Generate the explicit MPC with `empc = controller.toExplicit()`.
- Plot the generated solution, including regions, with `empc.feedback.fplot()`.
- Simulate the closed-loop system starting from the state $x = [3 \ 0]^T$. Confirm that your constraints are met. Reuse the simulation code from exercise 4. You can evaluate the explicit controller with `empc.evaluate(x)`.

Prob 3 | Compare explicit MPC with YALMIP implementation

We now compare the explicit MPC with the YALMIP implementation from exercise 4.

Tasks:

- If (for some reason) you skipped exercise 4, implement the controller using YALMIP.
- Plot the position, velocity, and input of the system using the YALMIP controller. Confirm that your solution is the same as for the explicit MPC case.
- Compare the solve times of the explicit MPC against the YALMIP implementation. What do you notice, is it as expected?