Exercise set 1 – MPC (SC42125)

Mattia Bianchi Barbara Franci Sergio Grammatico

1 Inverted pendulum: Modeling and control

Consider the following model of a rigid inverted pendulum

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} \dot{\theta}(t) \\ \dot{\omega}(t) \end{bmatrix} = \begin{bmatrix} \omega(t) \\ \frac{g}{\ell}\sin(\theta(t)) + \frac{1}{m\ell^2}u(t) \end{bmatrix} \\ y(t) = x(t) \\ x(0) = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} \end{cases}$$

with θ being the angular displacement (in rad) from the vertical position above the pivot, ω the angular velocity (rad/s), u the torque ($N\,m$) applied to the center of the pendulum, g the gravity acceleration, $\ell = 0.79$ m the length of the pendulum, $m = 0.24$ kg its mass. By using MATLAB:

1. Perform a 10 s simulation of the system with a variable-step solver (e.g. `ode45`) choosing the inputs $u = 0$, $u = -3\theta$, $u = -3\theta - \omega$, and compare the results.

2. Perform a 10 s fixed-step simulation of the system (with or without input), using Euler method. Compare the results for two different step sizes: $T_1 = 0.01$ s, $T_2 = 1$ s.

2 State-space representation

Define a state vector and realize the following models as state-space models *by hand*:

a) $G(s) = \dfrac{1}{(2s+1)(3s+1)}$;

b) $G(s) = \dfrac{2s+1}{3s+1}$;

c) $y(k) = y(k-1) + 2u(k-1)$;

d) $y(k) = a_1 y(k-1) + a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2)$

1. What is the connection between the poles of G and the state-space representation?
2. For what classes of $G(s)$ does one obtain a nonzero D matrix?
3. What is the order and gain of these systems?
4. Is there a connection between order and the numbers of inputs and outputs?

Find minimal realizations of the state space models $a) - d)$ via MATLAB.

5. Are some realizations non-minimal?

3 Predicted output computation

Consider the following system, for some parameter $a > 0$:

$$\begin{cases} x(k+1) = \begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u(k), \\ \qquad y(k) = [1 \quad 0]\; x(k), \end{cases}$$

with $x(0) = [1 \quad 1]^\top$ and

$$u(j) = \begin{cases} 1 & \text{if } \; j = 20 \\ 0 & \text{otherwise} \end{cases}$$

1. Compute the predicted output for $N = 25$, i.e., $y(N)$.

4 Lyapunov function for exponential stability

Let us consider the system $x(k+1) = f(x(k))$ with the origin as equilibrium, i.e., $f(0) = 0$. We say that the origin is **globally exponentially stable (GES)** if there exists $c > 0$ and $\gamma \in (0,1)$ such that

$$\|x(k)\| \leq c\|x(0)\|\gamma^k, \qquad \text{for all } k \geq 0.$$

Then, let $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ be a Lyapunov function for the origin of the system. Show that, if the following properties hold for all $x \in \mathbb{R}^n$

$$a_1\|x(k)\|^\sigma \leq V(x(k)) \leq a_2\|x(k)\|^\sigma \tag{1}$$

$$V(f(x(k))) - V(x(k)) \leq -a_3\|x(k)\|^\sigma \tag{2}$$

with a_1, a_2, a_3, $\sigma > 0$, then the origin is GES.

5 Exponential stability implies existence of a Lyapunov function

Let us consider the system $x(k+1) = f(x(k))$ with the origin as equilibrium, i.e., $f(0) = 0$, where $f : \mathcal{D} \to \mathbb{R}^n$ is continuously differentiable and $\mathcal{D} \coloneqq \{x \in \mathbb{R}^n \mid \|x\| < r\}$, for some $r > 0$. Let a, λ, and r_0 be positive constants with $r_0 < r/a$. Let $\mathcal{D}_0 \coloneqq \{x \in \mathbb{R}^n \mid \|x\| < r_0\}$. Assume that the solutions of the system satisfy

$$\|x(k)\| \leq a\|x(0)\|e^{-\lambda k}, \quad \forall x(0) \in \mathcal{D}_0, \forall k \geq 0. \tag{3}$$

Show that there exists a function $V : \mathcal{D}_0 \to \mathbb{R}$ that satisfies

$$V(f(x(k))) \leq V(x(k)) - a_1\|x(k)\|^2 \tag{4}$$

$$|V(x) - V(y)| \leq a_2\|x - y\|(\|x\| + \|y\|) \tag{5}$$

for all x, $y \in \mathcal{D}_0$ and for some positive constants a_1 and a_2.

Hint: Consider summing the solution $\|\phi(k; x)\|^2$ over k as candidate Lyapunov function, $V(\cdot)$. To prove (4), note that $\phi(k; f(x)) = \phi(k; \phi(1; x)) = \phi(k + 1; x)$, while to prove (5), note that continuous differentiability of $f(\cdot)$ implies Lipschitz continuity.

6 Comparison functions

- Given the following functions, determine if they belong to the class $\mathcal{K}$, $\mathcal{K}_\infty$ or $\mathcal{KL}$.

  a) $\alpha(x) = \tan^{-1}(x)$ *b)* $\alpha(x) = x^c,\ c > 0$

  c) $\alpha(x) = \min\left(x, x^2\right)$ *d)* $\beta(x, y) = \dfrac{x}{(kxy + 1)},\ k > 0$

  e) $\beta(x, y) = x^c e^{-y}$.

- Let α be a function of class $\mathcal{K}$ on $[0, a)$. Show that

$$\alpha(x + y) \leq \alpha(2x) + \alpha(2y), \qquad \forall x, y \in [0, a).$$

7 Positive semidefinite Q penalty and its square root

Consider the linear quadratic problem with system

$$x(k+1) = Ax(k) + Bu(k) \tag{6}$$
$$y(k) = Q^{1/2}x(k) \tag{7}$$

and infinite-horizon cost function

$$V(\boldsymbol{y}, \boldsymbol{u}) = \sum_{k=0}^{\infty} y(k)^\top y(k) + u(k)^\top R u(k) \tag{8}$$

with $Q \succcurlyeq 0$, $R \succ 0$ and (A, B) stabilizable pair.

- Show that if $Q \succcurlyeq 0$, then there exist a real, symmetric matrix $Q^{1/2}$ such that $Q^{1/2}Q^{1/2} = Q$ and $Q^{1/2} \succcurlyeq 0$.

  Hint: If a matrix Q is real and symmetric, it can be written as $Q = T^\top \Lambda T$, where T is an orthogonal matrix, i.e., $T^\top = T^{-1}$, and Λ is a diagonal matrix, and both matrices are real.

- Show that $(A, Q^{1/2})$ is detectable (observable) if and only if (A, Q) is detectable (observable).

8 An MPC problem

Consider the *linear quadratic* optimal control problem with discrete-time *linear* system

$$x(k+1) = Ax(k) + Bu(k) \tag{9}$$
$$y(k) = Cx(k) \tag{10}$$
$$x(0) = x_0 \tag{11}$$

and a finite-horizon *quadratic* cost function

$$V_N(\boldsymbol{y}_N, \boldsymbol{u}_N) = \frac{1}{2} \sum_{k=0}^{N-1} y(k)^\top y(k) + u(k)^\top R u(k) \tag{12}$$

with $R \succ 0$.

1. Compute prediction matrices P and S such that

$$\boldsymbol{y}_N = Px_0 + S\boldsymbol{u}_N, \tag{13}$$

$\boldsymbol{y}_N = (y(0), y(1), \ldots, y(N-1))$ and $\boldsymbol{u}_N = (u(0), u(1), \ldots, u(N-1))$. Then write V_N in the form

$$V_N(\boldsymbol{u}) = \frac{1}{2}\boldsymbol{u}^\top H\boldsymbol{u} + h^\top \boldsymbol{u} + \text{constant} \tag{14}$$

2. By using MATLAB (interfaced with CVX or YALMIP) solve the optimal control problem

$$\min_{\boldsymbol{u}} V_N(\boldsymbol{u}) \tag{15}$$

with problem data

$$R = 2,\ N = 5,\ A = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 1 \\ 1 & 3 & 1 \end{bmatrix}\ B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\ C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{bmatrix}\ x(0) = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

3. Given the optimal input $u(0)$ computed in point 2, compute $x(1) = x^+$. Then repeat 2, but now set $x(0) = x^+$.

4. Using the same data of point 2, solve the optimal control problem

$$\begin{cases} \min\limits_{\boldsymbol{u}} & V_N(\boldsymbol{u}) \\ \text{s.t.} & |u(k)| \leq 5,\ \forall k. \end{cases} \tag{16}$$

The students are strongly recommended to become familiar with (one of) the following open-source interfaces\optimization solvers:

- **CVX** (`cvxr.com/cvx`)

CVX is a MATLAB-based modeling system for **convex** optimization. CVX turns MATLAB into a modeling language, allowing constraints and objectives to be specified using standard MATLAB expression syntax. It is not a general-purpose tool for nonlinear optimization, nor is it a tool for checking whether or not your model is convex.

→ *Users' Guide* at cvxr.com/cvx/doc

• **YALMIP** (yalmip.github.io)

YALMIP is an open-source MATLAB toolbox that can be used to model and solve optimization problems typically occurring in systems and control theory. This toolbox makes development of optimization problems in general, and control oriented semidefinite programming (SDP) problems in particular.

→ *Installation* at yalmip.github.io/tutorial/installation
→ *Getting started* at yalmip.github.io/tutorial/basics

• **Gurobi** (gurobi.com – Academic license available)

The Gurobi optimizer is designed to solve Linear/Quadratic Programming, Quadratically Constrained Programming, and Mixed-Integer Programming (linear, quadratic, and quadratically constrained) problems. It supports a variety of programming and modeling languages including matrix-oriented interfaces for MATLAB, and it also includes a number of features to support the building of optimization models including models with convex, piecewise-linear objective functions, to capture certain non-linear problems.

→ *How to create an Academic license* at
gurobi.com/documentation/8.1/quickstart_mac/...
creating_a_new_academic_li.html
→ *Quick start* at
gurobi.com/documentation/8.1/quickstart_mac/index.html
→ *MATLAB API* at
gurobi.com/documentation/8.1/refman/...
matlab_api_overview.html#matlab:MATLAB