CONTROL & ESTIMATION THEORY

ZICHI ZHANG AND SHIHAO CHEN

Collaboration Statement

From the whole view, coursework Part I is divided into 3 parts which Zichi focused on the 1.1 and Shihao focused on the 1.2 then worked together on the 1.3. Of course, everyone has thought about all questions and proposed appropriate solutions. The above division of work is just to ensure that everyone has a focus.

Since the first time that Part I began, Zichi created a repository on GitHub and shared to Shihao immediately. Thus, we uploaded Python code and LaTeX code, LaTeX output and LaTeX figures to the repository. Everyone uploaded his part on Python code and Zichi was responsible for the design of typing LaTeX while both of us give our own opinions to make the PDF file looks more professional and closer to what Dr Pantelis Sopasakis shows on Canvas. For example, we have tried our best to plot similar figures and they are really much better than the past ones. It really cost us a lot of time on debugging every parameter including the gird and the color and thickness of every line.

On time management, we did not have an exact time to have a meet with each other but every time if anyone was trapped by a challenge for a long time then we would discuss it on the phone or have a Teams meeting to try to complete it and if both of the ways do not work that Zichi would ask Dr Pantelis Sopasakis for a possible hint. Besides this, we spent 8 hours every week on average and we had 5 face-to-face meetings to work these problems out. That was an efficient way when we got together and shared different ideas.

Zichi really helped Shihao a lot on coding and knowledge of this course, Shihao needs to concentrate more on class that can more easily work these out.

During this coursework, we both learned how to use GitHub and used Git commands in our coursework to commit changes. Since Dec. 2021, we totally created more than 14 commits in this repository, we try our best to make this collaborative project better with these useful tools.

(Zichi Zhang and Shihao Chen) EMAIL ADDRESSES: ZZHANG54@QUB.AC.UK AND SCHEN45@QUB.AC.UK.
Thanks Dr. Pantelis Sopasakis. Version 1.1.3. Last updated: January 22, 2022.

1 Part I: Control

1.1 Optimal Control.

(i) Prove that the function $g : \mathbb{R}^n \to \mathbb{R}$ given by $g(x) = \frac{1}{2}x^\intercal Qx + q^\intercal x + c$ with $Q \in \mathbb{S}^n_+$ is level-bounded if and only if $Q \in \mathbb{S}^n_{++}$.

Proof:

$$\begin{aligned}\mathrm{lev}_{\leq\alpha} f &= \{x \in \mathbb{R}^n : f(x) \leq \alpha\} \\ &= \{x \in \mathbb{R}^n : x^\intercal Qx \leq \alpha\} \end{aligned} \tag{1}$$

We are trying to prove that $\mathrm{lev}_{\leq\alpha} f$ are bounded, i.e., there is an $M \geq 0$ such that

$$\{x \in \mathbb{R}^n : x^\intercal Qx \leq \alpha\} \subseteq \mathcal{B}_M \tag{2}$$

$$\lambda_{\min}(Q)\,\|x\|^2 \leq x^\intercal Qx \leq \lambda_{\max}(Q)\,\|x\|^2 \tag{3}$$

$$\Longrightarrow \lambda_{\min}(Q)\,\|x\|^2 \leq \alpha \tag{4}$$

$$\Longrightarrow \|x\| \leq \sqrt{\frac{\alpha}{\lambda_{\min}(Q)}} := M \tag{5}$$

Thus, if $M \geq 0$ exists, then

$$\lambda_{\min}(Q) \geq 0 \tag{6}$$

$$\Longrightarrow \lambda(Q) \geq 0 \tag{7}$$

$$\Longrightarrow Q \in \mathbb{S}^n_{++} \tag{8}$$

Conclusion: the function $g : \mathbb{R}^n \to \mathbb{R}$ given by $g(x) = \frac{1}{2}x^\intercal Qx + q^\intercal x + c$ with $Q \in \mathbb{S}^n_+$ is level-bounded if and only if $Q \in \mathbb{S}^n_{++}$.

Next, consider the finite-horizon linear-quadratic optimal control problem

$$\mathbb{P}_N(x) : \underset{\substack{u_0,u_1,\ldots,u_{N-1}\\ x_0,x_1,\ldots,x_N}}{\text{minimise}} \sum_{t=0}^{N-1} \frac{1}{2} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + \frac{1}{2} x_N^\intercal P_f x_N, \tag{9a}$$

$$\text{subject to: } x_{t+1} = Ax_t + Bu_t, \forall t \in \mathbb{N}_{[0,N-1]}, \tag{9b}$$

$$x_0 = x, \tag{9c}$$

where $\left[\begin{smallmatrix} Q & S \\ S^\intercal & R \end{smallmatrix}\right] \succcurlyeq 0, Q \in \mathbb{S}^n_+, R \in \mathbb{S}^m_{++}, P_f \in \mathbb{S}^n_+$, and x is a given initial state.

(ii) Is Problem $\mathbb{P}_N$ convex? Does $\mathbb{P}_N$ have a minimiser?

Answer: Problem $\mathbb{P}_N$ is convex. Problem $\mathbb{P}_N$ has a minimiser.

(iii) Solve Problem $\mathbb{P}_N$ by eliminating the state sequence: determine the optimal sequence of control actions, the optimal sequence(s) of states and the optimal cost.

Solusion:

(iv) Solve Problem $\mathbb{P}_N$ by using the dynamic programming method.

Solusion:

$$\begin{aligned}
&\begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} \\
&= x_t^\intercal Q x_t + u_t R u_t + x_t^\intercal S u_t + u_t^\intercal S^\intercal x_t \\
&= x_t^\intercal Q x_t + u_t R u_t + x_t^\intercal S u_t + u_t^\intercal S^\intercal x_t + x_t^\intercal S R^{-1} S^\intercal x_t - x_t^\intercal S R^{-1} S^\intercal x_t \\
&= x_t^\intercal (Q - S R^{-1} S^\intercal) x_t + (u_t + R^{-1} S^\intercal x_t)^\intercal R (u_t + R^{-1} S^\intercal x_t)
\end{aligned} \tag{10}$$

Let

$$\begin{cases} \tilde{Q} = Q - S R^{-1} S^\intercal \\ \tilde{u}_t = u_t + R^{-1} S^\intercal x_t \end{cases} \tag{11}$$

$$\begin{aligned}
x_{t+1} &= Ax + B(\tilde{u}_t - R^{-1} S^\intercal x_t) \\
&= (A - B R^{-1} S^\intercal) x_t + B \tilde{u}_t, \forall t \in \mathbb{N}_{[0,N-1]}
\end{aligned} \tag{12}$$

Let

$$\tilde{A} = A - B R^{-1} S^\intercal \tag{13}$$

The problem changes to

$$\mathbb{P}_N(x) : \underset{\substack{u_0, u_1, \ldots, u_{N-1} \\ x_0, x_1, \ldots, x_N}}{\text{minimise}} \sum_{t=0}^{N-1} \left(\tfrac{1}{2} x_t^\intercal \tilde{Q} x_t + \tilde{u}_t^\intercal R \tilde{u}_t \right) + \frac{1}{2} x_N^\intercal P_f x_N, \tag{14a}$$

$$\text{subject to: } x_{t+1} = \tilde{A} x_t + B \tilde{u}_t, \forall t \in \mathbb{N}_{[0,N-1]}, \tag{14b}$$

$$x_0 = x, \tag{14c}$$

where $Q \in \mathbb{S}_{+}^{n}, R \in \mathbb{S}_{++}^{m}, P_f \in \mathbb{S}_{+}^{n}, \tilde{Q} = Q - S R^{-1} S^\intercal, \tilde{u}_t = u_t + R^{-1} S^\intercal x_t, \tilde{A} = A - B R^{-1} S^\intercal$, and x is a given initial state.

We identify the terminal cost, the stage cost, and the system dynamics. We have

$$V_f(x) = \tfrac{1}{2} x^\intercal P_f x, \tag{15}$$

$$\ell(x, \tilde{u}) = \tfrac{1}{2} x^\intercal \tilde{Q} x + \tilde{u}^\intercal R \tilde{u}, \tag{16}$$

$$f(x, \tilde{u}) = \tilde{A} x + B \tilde{u}. \tag{17}$$

We define $V_0^\star(x) = V_f(x)$, i.e.,

$$V_0^\star(x) = \tfrac{1}{2} x^\intercal P_f x. \tag{18}$$

Then, we know that $V_{t+1}^\star(x) = (\mathbb{T} V_t^\star)(x)$. We assume that

$$V_t^\star(x) = \tfrac{1}{2} x^\intercal P_t x. \tag{19}$$

We have

$$\begin{aligned}
V_t^\star(x) = (\mathbb{T} V_t^\star)(x) &= \min_{\tilde{u}} \left\{ \ell(x, \tilde{u}) + V_t^\star(f(x, \tilde{u})) \right\} \\
&= \min_{\tilde{u}} \left\{ \tfrac{1}{2} x^\intercal \tilde{Q} x + \tilde{u}^\intercal R \tilde{u} + \tfrac{1}{2} (\tilde{A} x + B \tilde{u})^\intercal P_t (\tilde{A} x + B \tilde{u}) \right\}.
\end{aligned} \tag{20}$$

After some algebraic manipulations

$$V_{t+1}^\star(x) = \tfrac{1}{2} x^\intercal (\tilde{Q} + \tilde{A}^\intercal P_t \tilde{A}) x + \min_{\tilde{u}} \left\{ \tfrac{1}{2} \tilde{u}^\intercal (R + B^\intercal P_t B) \tilde{u} + (B^\intercal P_t \tilde{A} x)^\intercal \tilde{u} \right\}. \tag{21}$$

The minimiser is...

$$\kappa_{t+1}^\star(x) = -(R + B^\intercal P_t B)^{-1} B^\intercal P_t A x. \tag{22}$$

It is convenient to write $\kappa_{t+1}^\star$ as an affine function, i.e., $\kappa_{t+1}^\star(x) = K_{t+1} x$, where K_{t+1} is given by

$$K_{t+1} = -(R + B^\intercal P_t B)^{-1} B^\intercal P_t A. \tag{23}$$

We can then substitute $\tilde{u} = \kappa^\star_{t+1}$ in Equation (21):

$$\begin{aligned} V^\star_{t+1}(x) &= \tfrac{1}{2}x^\intercal(\tilde{Q} + \tilde{A}^\intercal P_t \tilde{A})x + \min_{\tilde{u}} \left\{ \tfrac{1}{2}\tilde{u}^\intercal(R + B^\intercal P_t B)\tilde{u} + (B^\intercal P_t \tilde{A}x)^\intercal \tilde{u} \right\} \\ &= \tfrac{1}{2}x^\intercal(\tilde{Q} + \tilde{A}^\intercal P_t \tilde{A})x + \tfrac{1}{2}(K_{t+1}x)^\intercal(R + B^\intercal P_t B)(K_{t+1}x) + (B^\intercal P_t \tilde{A}x)^\intercal(K_{t+1}x). \end{aligned} \tag{24}$$

and we can rearrange the terms to write $V^\star_{t+1}$ in the form $V^\star_{t+1}(x) = \frac{1}{2}x^\intercal P_{t+1}x$.
We find that

$$P_{t+1} = \tilde{Q} + \tilde{A}^\intercal P_t \tilde{A} + K^\intercal_{t+1}(R_t + B^\intercal P_t B)K_{t+1} + 2\tilde{A}^\intercal P_t^\intercal B K_{t+1}. \tag{25}$$

We have that

$$V^\star_N(x) = \tfrac{1}{2}x^\intercal P_N x. \tag{26}$$

We have

$$\begin{aligned} x^\star_0 =& x, \\ \tilde{u}^\star_0 =& \kappa^\star_N(x^\star_0) = K_N x^\star_0, \\ x^\star_1 =& \tilde{A}x^\star_0 + B\tilde{u}^\star_0, \\ \tilde{u}^\star_1 =& \kappa^\star_{N-1}(x^\star_1) = K_{N-1}x^\star_1, \\ x^\star_2 =& \tilde{A}x^\star_1 + B\tilde{u}^\star_1, \\ &\vdots \\ \tilde{u}^\star_{N-1} =& \kappa^\star_1(x^\star_{N-1}) = K_1 x^\star_{N-1}, \\ x^\star_N =& \tilde{A}x^\star_{N-1} + B\tilde{u}^\star_{N-1}. \end{aligned}$$

Next, consider the following infinite-horizon optimal control problem

$$\mathbb{P}_\infty(x) : \underset{(u_t)_{t\in\mathbb{N}},(x_t)_{t\in\mathbb{N}}}{\text{minimise}} \sum_{t=0}^{\infty} \tfrac{1}{2} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix}, \tag{27a}$$

$$\text{subject to: } x_{t+1} = Ax_t + Bu_t, t \in \mathbb{N}, \tag{27b}$$

$$x_0 = x, \tag{27c}$$

where $\left[\begin{smallmatrix} Q & S \\ S^\intercal & R \end{smallmatrix}\right] \succcurlyeq 0, Q \in \mathbb{S}^n_+, R \in \mathbb{S}^m_{++}$.

(v) Under what conditions do the dynamic programming iterates, $V^\star_t$, converge? Justify your answer.

Anwser: In order for the dynamic programming iterates, $V^\star_t$, converge, it is necessary that there is a sequence of inputs $(u_t)_{t\in\mathbb{N}}$, such that the corresponding states, $(x_t)_{t\in\mathbb{N}}$, are such that the cost function is finite

$$\sum_{t=0}^{\infty} \tfrac{1}{2} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} < \infty \tag{28}$$

We denote the value function of $\mathbb{P}_\infty(x)$ by $V^\star_\infty(x)$, that is

$$V^\star_\infty(x) = \inf_{(u_t)_{t\in\mathbb{N}},(x_t)_{t\in\mathbb{N}}} \left\{ \sum_{t=0}^{\infty} \tfrac{1}{2} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} \middle| \begin{matrix} x_{t+1} = Ax_t + Bu_t, \forall t \in \mathbb{N}, \\ x_0 = x \end{matrix} \right\} \tag{29}$$

The value function $V^\star_\infty$, if it exists, must satisfy $V^\star_\infty = \mathbb{T}V^\star_\infty$, which can be equivalently written as

$$V^\star_\infty(x) = \min_u \left\{ \sum_{t=0}^{\infty} \tfrac{1}{2} \begin{bmatrix} x_t \\ u_t \end{bmatrix}^\intercal \begin{bmatrix} Q & S \\ S^\intercal & R \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + V^\star_\infty(x)(Ax + Bu) \right\} \tag{30}$$

By Equations (10), (11), (12) and (13), the problem changes to

$$\mathbb{P}_\infty(x): \underset{(u_t)_{t\in\mathbb{N}},(x_t)_{t\in\mathbb{N}}}{\text{minimise}} \sum_{t=0}^{\infty}\left(\tfrac{1}{2}x_t^\intercal \tilde{Q} x_t + \tilde{u}_t^\intercal R\tilde{u}_t\right), \tag{31a}$$

$$\text{subject to: } x_{t+1} = \tilde{A}x_t + B\tilde{u}_t, t\in\mathbb{N}, \tag{31b}$$

$$x_0 = x, \tag{31c}$$

where $Q\in\mathbb{S}^n_+, R\in\mathbb{S}^m_{++}, \tilde{Q} = Q - SR^{-1}S^\intercal, \tilde{u}_t = u_t + R^{-1}S^\intercal x_t, \tilde{A} = A - BR^{-1}S^\intercal$.

We have known a matrix P satisfying

$$P = Q + A^\intercal(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)A, \tag{DARE}$$

which is known a *discrete algebraic Riccati equation* (DARE), and

$$V^\star_\infty = \tfrac{1}{2}x^\intercal Px. \tag{32}$$

In $\mathbb{P}_\infty(x)$, we have

$$P = \tilde{Q} + \tilde{A}^\intercal(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)\tilde{A}. \tag{33}$$

where $Q\in\mathbb{S}^n_+, R\in\mathbb{S}^m_{++}, \tilde{Q} = Q - SR^{-1}S^\intercal, \tilde{A} = A - BR^{-1}S^\intercal$.

In those conditions, the dynamic programming iterates, $V_t^\star$, converge

$$\lim_{t\to\infty} V_t^\star = \tfrac{1}{2}x_t^\intercal Px_t. \tag{34}$$

(vi) Prove that the optimal value of $\mathbb{P}_\infty(x)$, if it exists, is given by $V(x) = \frac{1}{2}x^\intercal Px$ where $P\in\mathbb{S}^n_{++}$ Snsatisfies the algebraic equation

$$P = A^\intercal PA - (A^\intercal PB + S)(R + B^\intercal PB)^{-1}(B^\intercal PA + S^\intercal) + Q. \tag{35}$$

Proof: In $\mathbb{P}_\infty(x)$, we have

$$P = \tilde{Q} + \tilde{A}^\intercal(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)\tilde{A}. \tag{36}$$

where $Q\in\mathbb{S}^n_+, R\in\mathbb{S}^m_{++}, \tilde{Q} = Q - SR^{-1}S^\intercal, \tilde{A} = A - BR^{-1}S^\intercal$.

$$\begin{aligned} P &= \tilde{Q} + \tilde{A}^\intercal(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)\tilde{A} \\ &= Q - SR^{-1}S^\intercal + (A - BR^{-1}S^\intercal)^\intercal(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)(A - BR^{-1}S^\intercal) \\ &= Q - SR^{-1}S^\intercal + (A^\intercal - SR^{-1\intercal}B^\intercal)(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)(A - BR^{-1}S^\intercal) \\ &= Q - SR^{-1}S^\intercal + (A^\intercal - SR^{-1\intercal}B^\intercal)(P - PB(R + B^\intercal PB)^{-1}B^\intercal P)(A - BR^{-1}S^\intercal) \end{aligned} \tag{37}$$

(vii) Let $A = \left[\begin{smallmatrix}1 & 0\\ 1 & 1\end{smallmatrix}\right], B = \left[\begin{smallmatrix}1\\ 0\end{smallmatrix}\right], Q = I_2, R = 2$ and $S = \left[\begin{smallmatrix}-0.5\\ 0.5\end{smallmatrix}\right]$. The optimal value of $\mathbb{P}_N$ that you determined in question (iv) has the form $V_N^\star(x) = \frac{1}{2}x^\intercal P_N x$, where $P_N = P_N(P_0)$ depends on $P_0 = P_f$. Write a Python program that computes P_N (given N and P_f) and approximate $\lim_{N\to\infty} P_N(P_0)$ for different values of P_0. Verify that this limit satisfies Equation (35).

Answer: The Python code: Question 1.2 (vii).
By using Equation (33) and Python, we set $N = 50$ and different values of $P_0 = P_f : P_0 = \left[\begin{smallmatrix}0 & 0\\ 0 & 0\end{smallmatrix}\right]$ and $P_0 = \left[\begin{smallmatrix}1 & 1\\ 1 & 1\end{smallmatrix}\right]$, we can get the same P_N:

$$P_N = \begin{bmatrix} 7 & 2.5\\ 2.5 & 3\end{bmatrix}. \tag{38}$$

Thus, we can approximate $\lim_{N\to\infty} P_N(P_0)$:

$$\lim_{N\to\infty} P_N(P_0) = \begin{bmatrix} 7 & 2.5\\ 2.5 & 3\end{bmatrix}. \tag{39}$$

And we also can see that this limit satisfies Equation (35).

1.2 Linearisation theorem.

Let $f:\mathbb{R}^n \to \mathbb{R}^n$ be a continuously differentiable function with $f(0)=0$. Suppose that there are constants $\beta>0$ and $\eta>0$ so that $\|Jf(x)-Jf(x')\| \le \beta\|x-x'\|$, for all $x,x' \in \mathcal{B}_\eta$. Define $A = Jf(0)$.

(i) Prove that for all $x \in \mathcal{B}_\eta$, it is $\|f(x)-Ax\| \le \frac{\beta}{2}\|x\|^2$.

Proof. We have

$$\left\|Jf(x)-Jf(x')\right\| \le \beta\left\|x-x'\right\| \tag{40}$$

We know the fact that (from *the Fundamental Theorem of Calculus*)

$$f(x) = \int_0^1 Jf(\tau x)\,\mathrm{d}\tau x \tag{41}$$

Then we know (from *LEMMA 1.2.3, Page 22*) [1]

$$\begin{aligned} f(y) &= f(x) + \int_0^1 \langle Jf(x+\tau(y-x)), y-x\rangle\ \mathrm{d}\tau && (42)\\ &= f(x) + \langle Jf(x), y-x\rangle + \int_0^1 \langle Jf(x+\tau(y-x)) - Jf(x), y-x\rangle\ \mathrm{d}\tau && (43)\end{aligned}$$

Therefore, for all $x \in \mathcal{B}_\eta$, we have

$$\begin{aligned} &\|f(x)-Ax\| && (44)\\ &= \|f(x)-\langle A,x\rangle\| && (45)\\ &\le |f(x)-\langle A,x\rangle| && (46)\\ &= |f(x)-f(0)-\langle Jf(0), x-0\rangle| && (47)\\ &= \left|\int_0^1 \langle Jf(\tau x)-Jf(0), x\rangle\ \mathrm{d}\tau\right| && (48)\\ &\le \int_0^1 |\langle Jf(\tau x)-Jf(0), x\rangle|\ \mathrm{d}\tau && (49)\\ &\le \int_0^1 \|Jf(\tau x)-Jf(0)\|\cdot\|x\|\ \mathrm{d}\tau && (50)\\ &\le \int_0^1 \tau\beta\|x\|\,\|x\|\ \mathrm{d}\tau && (51)\\ &= \frac{\beta}{2}\|x\|^2 && (52)\end{aligned}$$

□

(ii) Prove that if $\rho(A)<1$, then the dynamical system $x_{t+1}=f(x_t)$ is locally exponentially stable.

Proof. We have $A=Jf(0)$. Hence $A \in \mathbb{R}^{n\times n}$ has exactly n (complex) eigenvalues, $\lambda_1,\ldots,\lambda_n$. The *spectral radius* of A is defined as

$$\rho(A) = \max\{|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|\}. \tag{53}$$

Specifically, we define $V: V(x) = x^\intercal P x, x \in \mathbb{R}^n, P \in \mathbb{S}^n_{++}$.
From *Lyapunov exponential stability theorem*, we have

$$V(f(x)) - V(x) \le -c\|x\|^\lambda, \tag{LDC}$$

where $X \to [0,\infty)$ and $\lambda, c > 0$, for all $x \in X$. If additionally, there exist $\underline{c}, \overline{c}, r > 0$ such that

$$\underline{c}\,\|x\|^{\lambda} \leq V(x), \forall x \in X, \tag{GLB}$$

$$V(x) \leq \overline{c}\,\|x\|^{\lambda}, \forall x \in X \cap \mathcal{B}_r, \tag{LUB}$$

$$V(f(x)) - V(x) \leq -c\,\|x\|^2, \tag{54}$$

The candidate function, $V(x) = x^{\intercal}Px$ satisfies (LDC), $V(f(x)) - V(x) \leq -\alpha(\|x\|)$ if

$$\underbrace{x^{\intercal}A^{\intercal}PAx}_{V(Ax)} - \underbrace{x^{\intercal}Px}_{V(x)} \leq -\alpha(\|x\|), \tag{55}$$

for all $x \in \mathbb{R}^n$. Equivalently,

$$x^{\intercal}(A^{\intercal}PA - P)x \leq -\alpha(\|x\|), \tag{LDC$_{\text{lin}}$}$$

for all $x \in \mathbb{R}^n$.
We know that $\rho(A) < 1$, we have the norm $\|x\| = \sqrt{x^{\intercal}Qx}$, where $Q \in \mathbb{S}^n_{++}$ and $\alpha(s) = s^2$. Then, (LDC$_{\text{lin}}$) becomes

$$x^{\intercal}(A^{\intercal}PA - P)x \leq -x^{\intercal}Qx, \tag{56}$$

for all $x \in \mathbb{R}^n$. This holds if

$$A^{\intercal}PA - P = -Q. \tag{57}$$

We have

$$x_{t+1} = Ax_t + h(x_t), \tag{58}$$

where $A = Jf(0)$ and h is such that $\|h(x)\| \leq \frac{L}{2}\|x\|^2$. The dynamical system

$$x_{t+1} = Ax_t, \tag{59}$$

is called the linearisation of $x_{t+1} = f(x_t)$ at 0.

$$\begin{aligned} \frac{\partial V(f(x_t))}{\partial t} &= 2f(x_t)Jf(x_t) \\ &= 2x_{t+1}Jf(x_t) \end{aligned} \tag{60}$$

□

(iii) Prove or disprove (by providing a counterexample) that if $\rho(A) = 1$, the system $x_{t+1} = f(x_t)$ is asymptotically stable.

Proof:

1.3 Model predictive controller design.

Consider the linear dynamical system

$$x_{t+1} = \begin{bmatrix} 1 & 0.7 \\ -0.1 & 1 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_t, \tag{61}$$

with $x_t \in \mathbb{R}^2$ and $u_t \in \mathbb{R}$. The system is subject to the state and input constraints

$$-\left[\begin{smallmatrix} 2 \\ 2 \end{smallmatrix}\right] \leq x_t \leq \left[\begin{smallmatrix} 2 \\ 2 \end{smallmatrix}\right], \text{ and } -1 \leq u_t \leq 1, \tag{62}$$

and the stage cost is $\ell(x,u) = \|x\|_2^2 + u^2$.

(i) Design an MPC using the terminal cost $V_f(x) = 0$ and the terminal set $X_f = \{0\}$. Compute the sets of feasible states with a prediction horizon $N = 1, \dots, 6$. Simulate the MPC-controlled dynamical system with $N = 10$ using one of the extreme points of X_N as the initial state, x_0.

Answer: The Python code: Question 1.3 (i).
The MPC-controlled dynamical system with $N = 10$, using one of the extreme points of X_N as

the initial state, $x_0 = \left[\begin{smallmatrix} 2 \\ -2 \end{smallmatrix}\right]$.
MPC controller:

$$\mathbb{P}_N(x): \underset{\substack{u_0,u_1,\dots,u_{N-1}\\ x_0,x_1,\dots,x_N}}{\text{Minimise}} \sum_{t=0}^{N-1} (\|x_t\|_2^2 + u_t^2), \tag{63a}$$

$$\text{subject to: } x_{t+1} = \left[\begin{smallmatrix} 1 & 0.7 \\ -0.1 & 1 \end{smallmatrix}\right] x_t + \left[\begin{smallmatrix} 1 \\ 0.5 \end{smallmatrix}\right] u_t, t \in \mathbb{N}_{[0,N-1]}, \tag{63b}$$

$$\left[\begin{smallmatrix} -2 \\ -2 \end{smallmatrix}\right] \leq x_t \leq \left[\begin{smallmatrix} 2 \\ 2 \end{smallmatrix}\right], t \in \mathbb{N}_{[1,N]}, \tag{63c}$$

$$-1 \leq u_t \leq 1, t \in \mathbb{N}_{[0,N-1]}, \tag{63d}$$

$$x_0 = \left[\begin{smallmatrix} 2 \\ -2 \end{smallmatrix}\right]. \tag{63e}$$

The set of feasible states is shown in Figure 1 below.

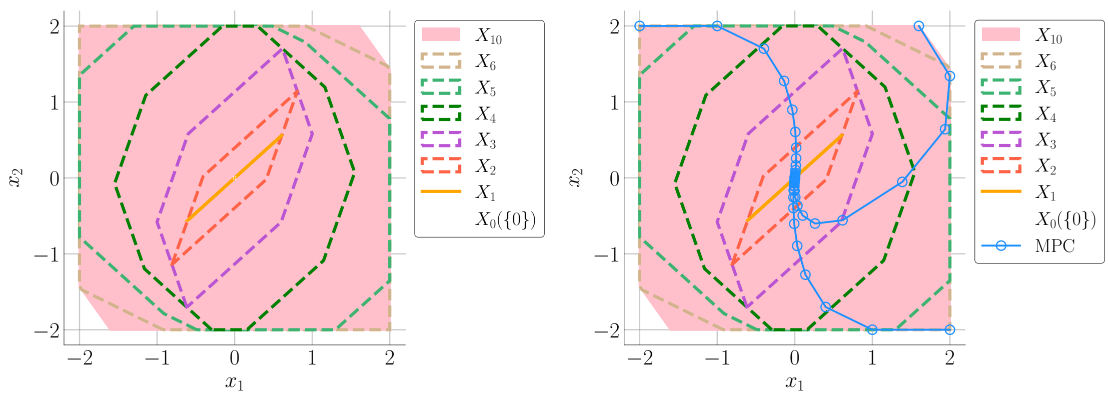

FIGURE 1. We denote by $X_t(X_f)$ the set of states that can be steered in no more than t steps to X_f. Observe that $Xt(\{0\}) \subseteq Xt'(\{0\})$ for $t < t'$. In other words, the larger the terminal set or the larger the prediction horizon is, the larger the set of feasible states will be. In the right plot we see three trajectories of the MPC-controlled system with $N = 10$, starting from three extreme points of $X_{10}(\{0\})$. The extreme points are $\left[\begin{smallmatrix} -2 \\ 2 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1.6 \\ 2 \end{smallmatrix}\right], \left[\begin{smallmatrix} 2 \\ -2 \end{smallmatrix}\right]$. The set $X_0(\{0\})$ is shown in all plots.

The simulation of the MPC-controlled dynamical system is shown in Figure 2 and Figure 3 below.

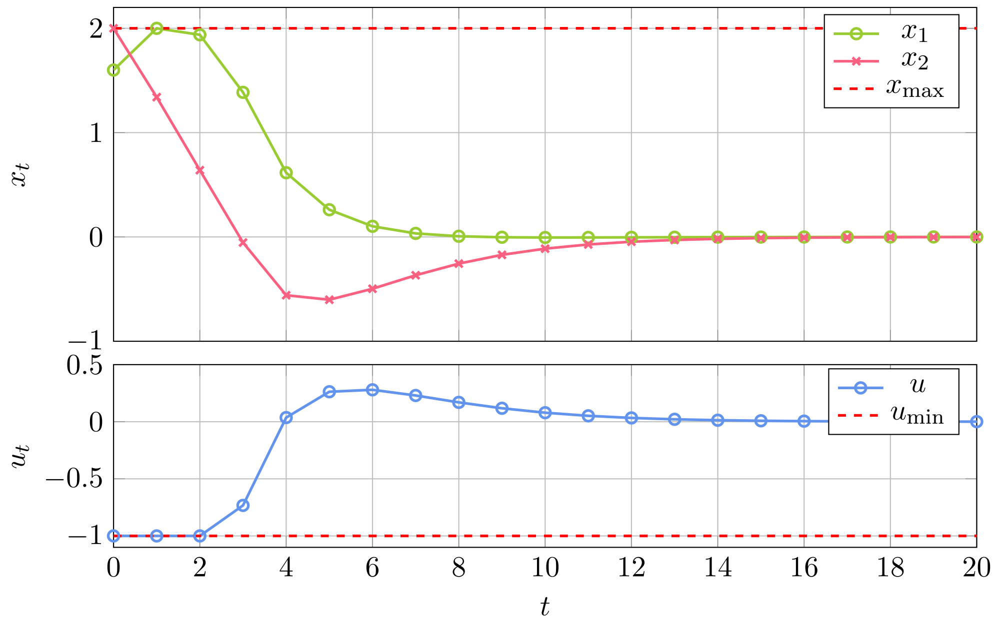

FIGURE 2. In the above plot we see the states changing of the MPC- controlled system with $N = 10$, and in the below plot we see the control actions changing of the MPC- controlled system with $N = 10$, starting from the extreme point $\left[\begin{smallmatrix}1.6\\2\end{smallmatrix}\right]$. We can observe that x_t is steered in 20 steps to $X_0(\{0\})$.

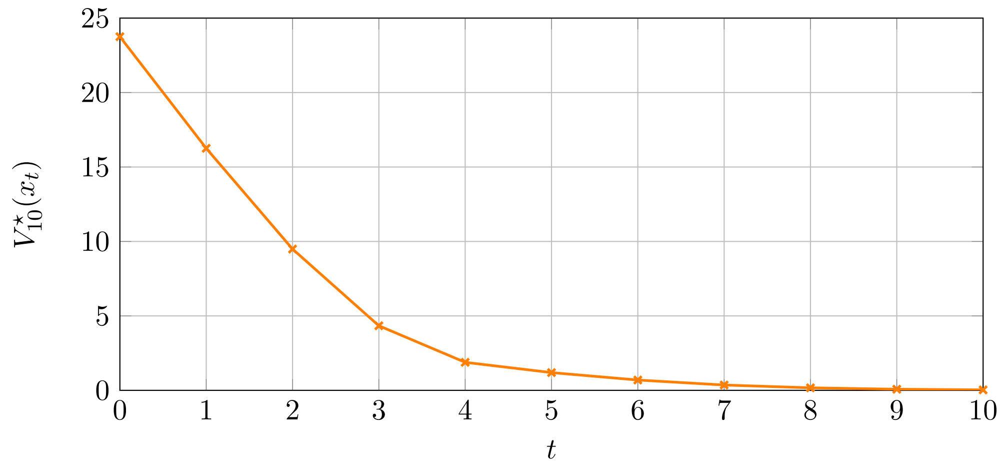

FIGURE 3. The "energy" of the system as measured by the Lyapunov function $V^\star_{10}$ for the MPC-controlled system with $N = 10$.

(ii) Design an MPC by following the procedure outlined in Handout 10, Sections 10.2 and 10.3. Use a prediction horizon $N = 10$ and compute the set of feasible states, X_N. Simulate the MPC-controlled system starting from the extreme points of X_N.

Answer: The Python code: Question 1.3 (ii).
The MPC-controlled dynamical system with $N = 10$, using one of the extreme points of X_N as the initial state, $x_0 = \left[\begin{smallmatrix}2\\-2\end{smallmatrix}\right]$.

MPC controller:

$$
\begin{aligned}
\mathbb{P}_N(x): \quad & \underset{\substack{u_0,u_1,\dots,u_{N-1}\\ x_0,x_1,\dots,x_N}}{\text{Minimise}} \sum_{t=0}^{N-1} (\|x_t\|_2^2 + u_t^2), && \text{(64a)}\\
\text{subject to: } & x_{t+1} = \left[\begin{smallmatrix} 1 & 0.7 \\ -0.1 & 1 \end{smallmatrix}\right] x_t + \left[\begin{smallmatrix} 1 \\ 0.5 \end{smallmatrix}\right] u_t, t \in \mathbb{N}_{[0,N-1]}, && \text{(64b)}\\
& \left[\begin{smallmatrix} -2 \\ -2 \end{smallmatrix}\right] \leq x_t \leq \left[\begin{smallmatrix} 2 \\ 2 \end{smallmatrix}\right], t \in \mathbb{N}_{[1,N]}, && \text{(64c)}\\
& -1 \leq u_t \leq 1, t \in \mathbb{N}_{[0,N-1]}, && \text{(64d)}\\
& x_0 = \left[\begin{smallmatrix} 2 \\ -2 \end{smallmatrix}\right]. && \text{(64e)}
\end{aligned}
$$

The set of feasible states is shown in Figure 4 below.

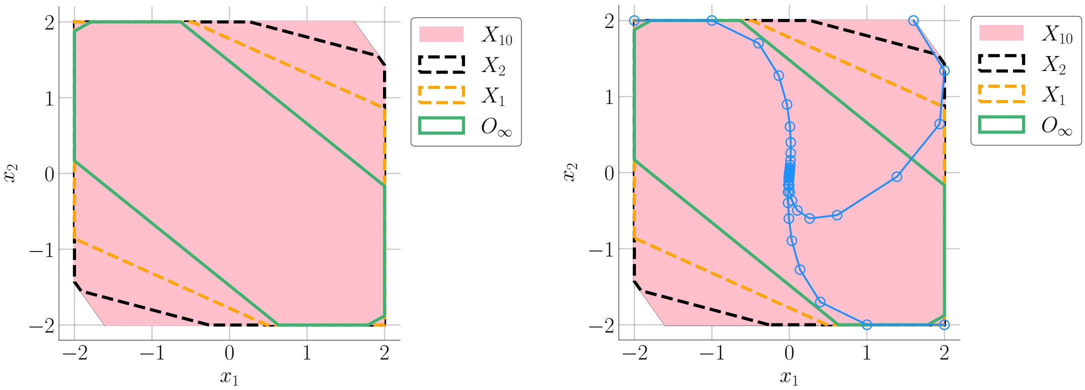

FIGURE 4. We denote by $X_t(X_f)$ the set of states that can be steered in no more than t steps to X_f. Observe that $Xt(O_\infty) \subseteq Xt'(O_\infty)$ for $t < t'$. In other words, the larger the terminal set or the larger the prediction horizon is, the larger the set of feasible states will be. In the right plot we see three trajectories of the MPC-controlled system with $N = 10$, starting from three extreme points of $X_{10}(O_\infty)$. The extreme points are $\left[\begin{smallmatrix} -2 \\ 2 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1.6 \\ 2 \end{smallmatrix}\right], \left[\begin{smallmatrix} 2 \\ -2 \end{smallmatrix}\right]$. The set O_∞ is shown in all plots with green colour.

The simulation of the MPC-controlled dynamical system is shown in Figure 5 and Figure 6 below.

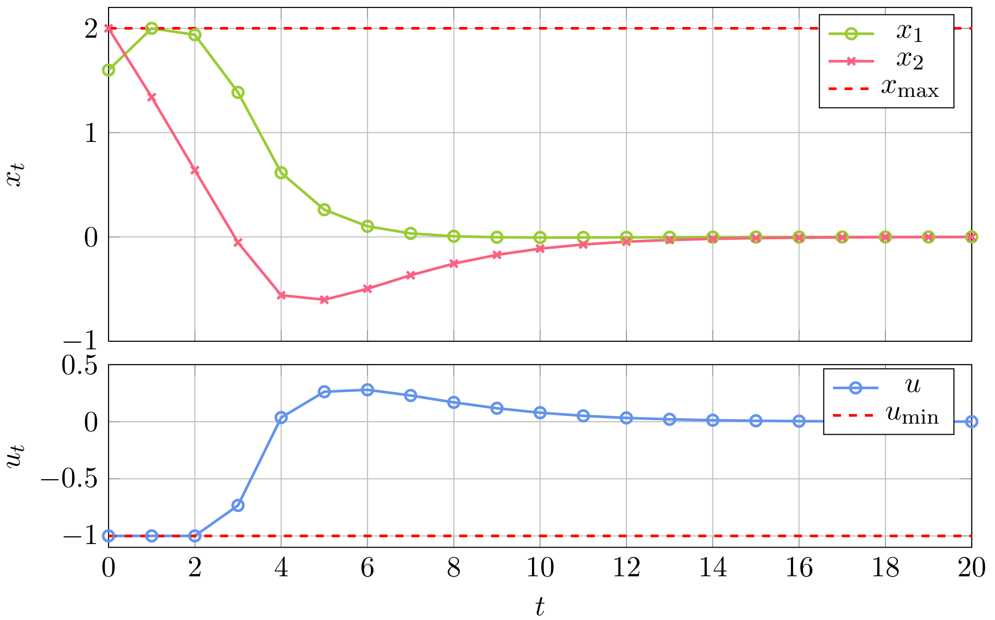

FIGURE 5. In the above plot we see the states changing of the MPC- controlled system with $N = 10$, and in the below plot we see the control actions changing of the MPC- controlled system with $N = 10$, starting from the extreme point $\begin{bmatrix}1.6\\2\end{bmatrix}$. We can observe that x_t is steered in 20 steps to O_∞.

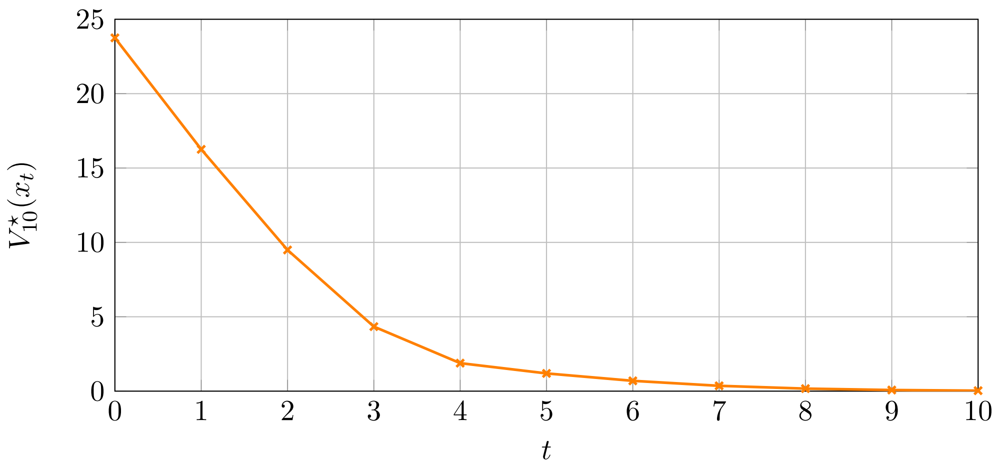

FIGURE 6. The "energy" of the system as measured by the Lyapunov function $V_{10}^\star$ for the MPC-controlled system with $N = 10$.

(iii) Design an MPC controller using an ellipsoidal terminal set and prediction horizon $N = 10$. Provide simulation results starting from different initial states.

Answer: The Python code: Question 1.3 (iii).
Using an ellipsoidal terminal set, let

$$\alpha \leq \min_{i \in \mathbb{N}_{[1,s]}} \frac{b_i^2}{\|P^{-1/2} h_i\|_2^2} \tag{65}$$

MPC controller:

$$\mathbb{P}_N(x): \underset{\substack{u_0,u_1,\ldots,u_{N-1}\\ x_0,x_1,\ldots,x_N}}{\text{Minimise}} \sum_{t=0}^{N-1}(\|x_t\|_2^2+u_t^2)+\tfrac{1}{2}x_N^\intercal P x_N, \tag{66a}$$

$$\text{subject to: } x_{t+1}=\begin{bmatrix}1 & 0.7\\ -0.1 & 1\end{bmatrix}x_t+\begin{bmatrix}1\\ 0.5\end{bmatrix}u_t, t\in\mathbb{N}_{[0,N-1]}, \tag{66b}$$

$$x_N^\intercal P x_N \leq \alpha, \tag{66c}$$

$$\begin{bmatrix}-2\\ -2\end{bmatrix}\leq x_t\leq\begin{bmatrix}2\\ 2\end{bmatrix}, t\in\mathbb{N}_{[1,N]}, \tag{66d}$$

$$-1\leq u_t\leq 1, t\in\mathbb{N}_{[0,N-1]}, \tag{66e}$$

$$x_0=x. \tag{66f}$$

The simulation of the MPC-controlled dynamical system is shown in Figure 7 and Figure 8 below.

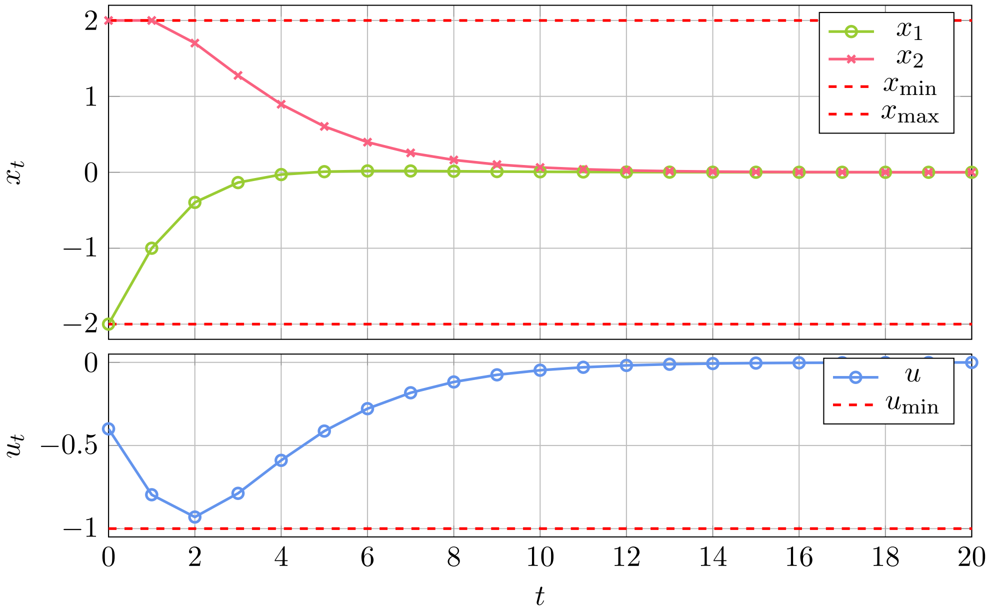

FIGURE 7. In the above plot we see the states changing of the MPC- controlled system with $N=10$, and in the below plot we see the control actions changing of the MPC- controlled system with $N=10$, starting from the extreme point $\begin{bmatrix}-2\\ 2\end{bmatrix}$. We can observe that x_t is steered in 20 steps to X_f.

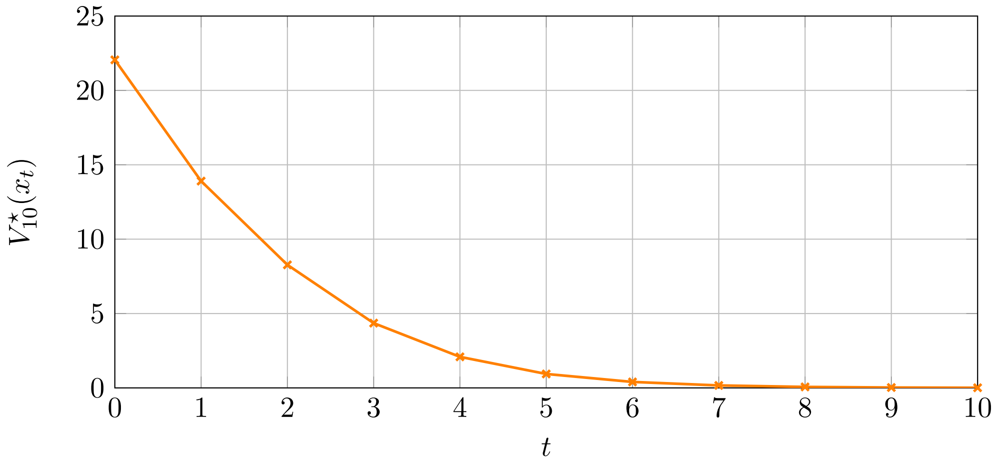

FIGURE 8. The "energy" of the system as measured by the Lyapunov function $V_{10}^\star$ for the MPC-controlled system with $N=10$.

Now consider the nonlinear dynamical system[†].

$$x_{t+1} = \begin{bmatrix} 1 & 0.7 \\ -0.1 & 1 \end{bmatrix} x_t + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_t + \frac{1}{20} \begin{bmatrix} x_t^\intercal x_t \\ \sin^2(x_{t,1}) \end{bmatrix}, \tag{67}$$

which is subject to the constraints given in (62).

(iv) Design a nonlinear model predictive controller using the methodology of Section 10.6 in Handout 10: determine the terminal cost function and the terminal set of constraints.

Answer: The Python code: Question 1.3 (iv).
Choose $\ell(x,u) = \frac{1}{2}(x^\intercal Q x + u^\intercal R u) = \|x\|_2^2 + u^2$, hence $Q = \left[\begin{smallmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{smallmatrix}\right], R = \sqrt{2}$. Let K be a stabilising gain for (A,B). Define $\bar{A} = A + BK, \bar{Q} = Q + K^\intercal R K$. Choose $P \in \mathbb{S}^2_{++}$ such that

$$P = \bar{A}^\intercal P \bar{A} + 2\bar{Q}. \tag{68}$$

Calculate it by Python, we can get

$$P = \begin{bmatrix} 5.95663 & -1.17022 \\ -1.17022 & 6.60196 \end{bmatrix}. \tag{69}$$

Choose $V_f(x) = \frac{1}{2}x^\intercal P x$ and $X_f = \{x \in \mathbb{R}^2 : V_f(x) \leq \frac{\alpha}{2}\}$, for appropriately small $\alpha > 0$.
We can get the terminal cost function

$$V_f(x) = \frac{1}{2} x^\intercal \begin{bmatrix} 5.95663 & -1.17022 \\ -1.17022 & 6.60196 \end{bmatrix} x. \tag{70}$$

$$\begin{aligned} X_f &= \{x \in \mathbb{R}^2 : V_f(x) \leq \tfrac{\alpha}{2}\} \\ &= \{x \in \mathbb{R}^2 : x^\intercal P x \leq \alpha\} \end{aligned} \tag{71}$$

NMPC controller:

$$\mathbb{P}_N(x) : \operatorname*{Minimise}_{\substack{u_0,u_1,\ldots,u_{N-1} \\ x_0,x_1,\ldots,x_N}} \sum_{t=0}^{N-1} (\|x_t\|_2^2 + u_t^2) + \tfrac{1}{2} x_N^\intercal \left[\begin{smallmatrix} 5.95663 & -1.17022 \\ -1.17022 & 6.60196 \end{smallmatrix}\right] x_N, \tag{72a}$$

$$\text{subject to: } x_{t+1} = \left[\begin{smallmatrix} 1 & 0.7 \\ -0.1 & 1 \end{smallmatrix}\right] x_t + \left[\begin{smallmatrix} 1 \\ 0.5 \end{smallmatrix}\right] u_t + \tfrac{1}{20} \begin{bmatrix} x_t^\intercal x_t \\ \sin^2(x_{t,1}) \end{bmatrix}, t \in \mathbb{N}_{[0,N-1]}, \tag{72b}$$

$$x_N^\intercal P x_N \leq \alpha, \tag{72c}$$

$$\left[\begin{smallmatrix} -2 \\ -2 \end{smallmatrix}\right] \leq x_t \leq \left[\begin{smallmatrix} 2 \\ 2 \end{smallmatrix}\right], t \in \mathbb{N}_{[1,N]}, \tag{72d}$$

$$-1 \leq u_t \leq 1, t \in \mathbb{N}_{[0,N-1]}, \tag{72e}$$

$$x_0 = x. \tag{72f}$$

Calculate α by Python, we can get $\alpha = 5.749210061145554$.
Thus, we can get the terminal set of constraints

$$X_f = \{x \in \mathbb{R}^2 : x^\intercal P x \leq 5.749210061145554\}. \tag{73}$$

References

[1] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course.* Kluwer Academic Publishers, 2004.

[†] We denote the two coordinates of $x_t \in \mathbb{R}^2$ by $x_{t,1}$ and $x_{t,2}$