Reinforcement Learning and Optimal Control

Ghanashyama Prabhu – N15244467

September 2025

1 Exercise 1

Problem: Find and classify the minima of each function.

1. $f(x) = -e^{-(2x-1)^2}$, $x \in \mathbb{R}$.

$$f(1/2) = -e^0 = -1, \qquad -e^{-(2x-1)^2} \in (-1, 0].$$

$$f'(x) = -e^{-(2x-1)^2} \cdot 4(2x-1), \quad f'(x) = 0 \iff 2x - 1 = 0 \iff x = \tfrac{1}{2}.$$

$$f''(x) = 8e^{-(2x-1)^2}\big[\, 1 - 2(2x-1)^2 \,\big], \quad f''(1/2) = 8 > 0.$$

Hence, $x = \frac{1}{2}$ is a *strict global minimum* with $f_{\min} = -1$.

2. $f(x, y) = (1-x)^2 + 50\,(2y - x^2 + 5)^2$.

Since this is a sum of squares, $f \geq 0$. The minimum 0 is attained iff

$$1 - x = 0, \quad 2y - x^2 + 5 = 0 \;\Rightarrow\; (x^\star, y^\star) = (1, -2).$$

Gradient:

$$\partial_x f = 2(x-1) - 200x(2y - x^2 + 5), \qquad \partial_y f = 200(2y - x^2 + 5).$$

The only critical point is $(1, -2)$. The Hessian at $(1, -2)$ is

$$\nabla^2 f(1, -2) = \begin{pmatrix} 402 & -400 \\ -400 & 400 \end{pmatrix},$$

whose leading principal minors are $402 > 0$ and $\det = 402 \cdot 400 - (-400)^2 = 800 > 0$, so it is positive definite. Therefore $(1, -2)$ is a *strict global minimizer* with $f_{\min} = 0$.

3. $f(x, y) = 10x + x^2 + y - 4y^2$.

$$\nabla f = \begin{pmatrix} 10 + 2x \\ 1 - 8y \end{pmatrix} = 0 \;\Rightarrow\; x = -5,\; y = \tfrac{1}{8}.$$

$$\nabla^2 f = \begin{pmatrix} 2 & 0 \\ 0 & -8 \end{pmatrix},$$

which is indefinite (eigenvalues 2 and -8), hence the critical point is a *saddle*; no local minimum.

4. $f(x) = x^\top \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} x + \begin{pmatrix} -1 \\ 1 \end{pmatrix}^\top x, \quad x \in \mathbb{R}^2.$

Expanding,

$$f(x_1, x_2) = 3x_1^2 + 3x_2^2 + 2x_1x_2 - x_1 + x_2.$$

For a quadratic $x^\top Qx + c^\top x$ with symmetric Q, $\nabla f = 2Qx + c$:

$$\nabla f = \begin{pmatrix} 6 & 2 \\ 2 & 6 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \end{pmatrix} = 0 \;\Rightarrow\; (x_1^\star, x_2^\star) = \left(\tfrac{1}{4}, -\tfrac{1}{4}\right).$$

$$\nabla^2 f = 2\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} = \begin{pmatrix} 6 & 2 \\ 2 & 6 \end{pmatrix},$$

whose eigenvalues are 8 and 4 (both > 0), so the Hessian is positive definite. Thus we have a *strict global minimum* at $(\frac{1}{4}, -\frac{1}{4})$.

5. $f(x) = x^\top \begin{pmatrix} 1 & 3 \\ 3 & 1 \end{pmatrix} x + \begin{pmatrix} 10 \\ 1 \end{pmatrix}^\top x, \quad x \in \mathbb{R}^2.$

Here $\nabla^2 f = 2\begin{pmatrix} 1 & 3 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 6 \\ 6 & 2 \end{pmatrix}$ has eigenvalues 8 and -4, hence is indefinite. Therefore the unique stationary point is a *saddle*; there is no (global) minimum.

6. $f(x) = \frac{1}{2}\, x^\top \begin{pmatrix} 4 & 4 & 0 \\ 4 & 4 & 0 \\ 0 & 0 & 2 \end{pmatrix} x - \begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}^\top x, \quad x \in \mathbb{R}^3.$

Let $H = \begin{pmatrix} 4 & 4 & 0 \\ 4 & 4 & 0 \\ 0 & 0 & 2 \end{pmatrix}$, $c = \begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}$. Then $\nabla f = Hx - c$:

$$\begin{cases} 4x_1 + 4x_2 = 0, \\ 4x_1 + 4x_2 = 0, \\ 2x_3 - 4 = 0. \end{cases} \qquad \Rightarrow \qquad x_2 = -x_1, \quad x_3 = 2.$$

The Hessian is $H \succeq 0$ (PSD but not PD), so the minimizers form the affine set

$$\{\, (t,\, -t,\, 2) \;:\; t \in \mathbb{R} \,\}.$$

Hence, there is a *global (non-strict) minimum* attained on this line.

2 Exercise 2

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \tfrac{1}{2}\, x^\top Q\, x \\ \text{subject to} \quad & Ax = b. \end{aligned}$$

Assume $Q \in \mathbb{R}^{n\times n}$ with $Q \succ 0$, $A \in \mathbb{R}^{m\times n}$ has full row rank with $m < n$, and $b \in \mathbb{R}^m$.

Lagrangian and KKT conditions.

$$\mathcal{L}(x,\lambda) = \tfrac{1}{2}\, x^\top Q x + \lambda^\top (Ax - b).$$

First-order (KKT) conditions:

$$\nabla_x \mathcal{L} = Qx + A^\top \lambda = 0, \qquad \nabla_\lambda \mathcal{L} = Ax - b = 0.$$

Block KKT system:

$$\begin{pmatrix} Q & A^\top \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}.$$

Closed-form solution. From $Qx = -A^\top \lambda$ we get $x = -Q^{-1}A^\top \lambda$. Substituting into $Ax = b$:

$$-AQ^{-1}A^\top \lambda = b \;\Rightarrow\; \lambda^\star = -(AQ^{-1}A^\top)^{-1} b,$$

$$x^\star = -Q^{-1}A^\top \lambda^\star = Q^{-1}A^\top \left(AQ^{-1}A^\top\right)^{-1} b.$$

Minimum objective value:

$$f^\star \;=\; \tfrac{1}{2}\, b^\top \left(AQ^{-1}A^\top\right)^{-1} b.$$

Numerical example.

$$Q = \begin{pmatrix} 100 & 2 & 1 \\ 2 & 10 & 3 \\ 1 & 3 & 1 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}, \quad b = 1.$$

Then

$$\lambda^\star \approx -0.1983805668, \qquad x^\star \approx \begin{bmatrix} -0.00404858 \\ -0.40080972 \\ 1.40485830 \end{bmatrix}, \qquad Ax^\star = 1, \quad f^\star \approx 0.0991902834.$$

3 Exercise 3

Optimal Control for Quadrotor Point Tracking

We want to generate a control input that moves the drone from $(0,0)$ toward the point $(-3,3)$ using a finite-horizon linear-quadratic tracking problem:

$$\begin{aligned} \min_{\{x_n,u_n\}_{n=0}^N} \quad & \frac{1}{2}\sum_{n=0}^{N} \Big[(x_n - x_{\text{des}})^\top Q (x_n - x_{\text{des}}) + u_n^\top R u_n\Big] \\ \text{s.t.} \quad & x_{n+1} = Ax_n + Bu_n, \qquad n = 0,\dots,N-1, \\ & x_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^\top, \end{aligned} \tag{1}$$

where

$$x_{\text{des}} = \begin{bmatrix} -3 \\ 0 \\ 3 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad Q \succeq 0,\ R \succ 0,$$

and $x_n \in \mathbb{R}^6$, $u_n \in \mathbb{R}^2$.

Tasks.

1. Write down the first–order optimality (KKT) conditions for (1).
2. For $N = 500$, choose diagonal weights $Q \succ 0$ and $R \succ 0$ and solve the problem by forming the KKT linear system and using `NumPy`'s `solve` (avoid explicit matrix inverses).
3. Plot all state components of the optimal trajectory as functions of time.
4. Plot the optimal control inputs as functions of time.

solution We introduce Lagrange multipliers $\lambda_{k+1} \in \mathbb{R}^6$ for the dynamics constraints $x_{k+1} = Ax_k + Bu_k$. The Lagrangian is

$$\mathcal{L} \;=\; \frac{1}{2}\sum_{k=0}^{N}\Big[(x_k - x_{\text{des}})^\top Q (x_k - x_{\text{des}}) + u_k^\top R u_k\Big] \;+\; \sum_{k=0}^{N-1} \lambda_{k+1}^\top\big(Ax_k + Bu_k - x_{k+1}\big).$$

The first-order conditions are:

Primal dynamics: $\quad x_{k+1} = Ax_k + Bu_k, \quad x_0 = \bar{x}_0, \qquad (2)$

Costate recursion: $\quad \lambda_N = Q(x_N - x_{\text{des}}), \qquad (3)$

$$\lambda_k = Q(x_k - x_{\text{des}}) + A^\top \lambda_{k+1}, \qquad k = N-1, \dots, 0, \tag{4}$$

Stationarity w.r.t. u_k: $\quad 0 = Ru_k + B^\top \lambda_{k+1}, \qquad k = 0, \dots, N-1. \qquad (5)$

Stack all decision variables

$$z = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \\ u_0 \\ \vdots \\ u_{N-1} \end{bmatrix}, \qquad \lambda = \text{Lagrange multipliers for the equality constraints.}$$

The quadratic cost can be written as

$$\frac{1}{2}z^\top H z + f^\top z,$$

with block diagonal matrices

$$H = \mathrm{blkdiag}(\underbrace{Q,\ldots,Q}_{N+1\ \text{times}}, \underbrace{R,\ldots,R}_{N\ \text{times}}), \qquad f = \begin{bmatrix} -Qx_{\text{des}} \\ \vdots \\ -Qx_{\text{des}} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

The equality constraints are collected as

$$Ez = b,$$

where E encodes the dynamics

$$x_{k+1} - Ax_k - Bu_k = 0, \quad k = 0,\ldots,N-1,$$

and the initial condition $x_0 = \bar{x}_0$.

The Karush–Kuhn–Tucker system is then

$$\begin{bmatrix} H & E^\top \\ E & 0 \end{bmatrix} \begin{bmatrix} z \\ \lambda \end{bmatrix} = \begin{bmatrix} -f \\ b \end{bmatrix}.$$

$$Q = \begin{pmatrix} 50 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 50 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} 0.05 & 0 \\ 0 & 0.05 \end{pmatrix}$$

Tuning notes for Q and R. We use quadratic penalties with state weight Q (state cost) and control weight R (control cost).

- Placing larger weights on the position states (e.g., x and z)—say $Q_{xx} = Q_{zz} = 50$—drives the quadrotor to the goal more aggressively.
- A smaller R makes control "cheap," encouraging larger/faster thrust commands and quicker convergence.
- A larger R penalizes effort more, producing smoother/smaller thrust inputs and a slower approach to the goal.

- **High Q, Low R:** leads to fast, precise convergence to the goal, but at the cost of aggressive thrust inputs and sharp tilting maneuvers.
- **Low Q, High R:** yields smoother and more energy-efficient trajectories, but results in looser tracking and possibly failure to exactly reach the goal.