ROB6323: Reinforcement learning and optimal control for autonomous systems I

Exercise series 3

For questions requesting a written answer, please provide a detailed explanation and typeset answers (e.g. using LaTeX[1]). Include plots where requested in the answers (or in a Jupyter notebook where relevant). For questions requesting a software implementation, please provide your code in runnable Jupyter Notebook. Include comments explaining how the functions work and how the code should be run if necessary. Code that does not run out of the box will be considered invalid.

Exercise 1 [25 points]

Implement in a Jupyter Notebook the optimization algorithm that uses randomized smoothing (without line search) to minimize the following functions. Use $\alpha = 0.01$, $\mu = 1$ and $N_{samples} = 10$ for the Monte-Carlo estimation.

- $-\mathrm{e}^{-(x-1)^2}$, starting with $x_0 = 0$
- $(1-x)^2 + 100(y-x^2)^2$, starting with $x_0 = y_0 = 1.2$
- $x^T \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} x + \begin{bmatrix} -1 & 1 \end{bmatrix} x$, staring with $x_0 = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\frac{1}{2} x^T \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix} x - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x$, staring with $x_0 = \begin{pmatrix} -10 \\ -10 \\ -10 \end{pmatrix}$

For each function:

- Show the convergence of the method by plotting the norm of the distance to the optimum as a function of the number of iterations.
- How does this compare to the gradient descent method you implemented in the previous series?

Exercise 2 [40 points]

Answer the questions in Jupyter notebook *exercise_2.ipynb*

[1] https://en.wikibooks.org/wiki/LaTeX, NYU provides access to Overleaf to all the community https://www.overleaf.com/edu/nyu

Exercise 3 [35 points]

Consider the following dynamical system

$$x_{n+1} = \begin{cases} -x_n + 1 + u_n & \text{if } -2 \leq -x_n + 1 + u_n \leq 2 \\ 2 & \text{if } -x_n + 1 + u_n > 2 \\ -2 & \text{else} \end{cases}$$

where $x_n \in \{-2, -1, 0, 1, 2\}$ and $u_n \in \{-1, 0, 1\}$, and the cost function

$$J = \left(\sum_{k=0}^{2} 2|x_k| + |u_k| \right) + x_3^2 \tag{1}$$

- Use the dynamic programming algorithm to solve the finite horizon optimal control problem that minimizes J. Show the different steps of the algorithms and present the results in a table including the cost to go and the optimal control at every stage.
- What is the sequence of control actions, states and the optimal cost if $x_0 = 0$, if $x_0 = -2$ and if $x_0 = 2$?