Robot Learning

Winter Semester 2020/2021, Homework 2

Prof. Dr. J. Peters, J. Watson, J. Carvalho, J. Urain and T. Dam

TECHNISCHE UNIVERSITÄT DARMSTADT

Total points: 20
Due date: Midnight, Monday, 7 December, 2020

Name, Surname, ID Number

Problem 2.1 Optimal Control [20 Points]

In this exercise, we consider a finite-horizon discrete time-varying Stochastic Linear Quadratic Regulator with Gaussian noise and time-varying quadratic reward function. Such system is defined as

$$\boldsymbol{s}_{t+1} = \boldsymbol{A}_t \boldsymbol{s}_t + \boldsymbol{B}_t a_t + \boldsymbol{w}_t, \tag{1}$$

where $\boldsymbol{s}_t$ is the state, a_t is the control signal, $\boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{b}_t, \boldsymbol{\Sigma}_t)$ is Gaussian additive noise with mean $\boldsymbol{b}_t$ and covariance $\boldsymbol{\Sigma}_t$ and $t = 0, 1, \ldots, T$ is the time horizon. The control signal a_t is computed as

$$a_t = -\boldsymbol{K}_t \boldsymbol{s}_t + k_t \tag{2}$$

and the reward function is

$$reward_t = \begin{cases} -(\boldsymbol{s}_t - \boldsymbol{r}_t)^\top \boldsymbol{R}_t (\boldsymbol{s}_t - \boldsymbol{r}_t) - a_t^\top \boldsymbol{H}_t a_t & \text{when} \quad t = 0, 1, \ldots, T-1 \\ -(\boldsymbol{s}_t - \boldsymbol{r}_t)^\top \boldsymbol{R}_t (\boldsymbol{s}_t - \boldsymbol{r}_t) & \text{when} \quad t = T \end{cases} \tag{3}$$

a) Implementation [8 Points]

Implement the LQR with the following properties

$$\boldsymbol{s}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \qquad T = 50$$

$$\boldsymbol{A}_t = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix} \qquad \boldsymbol{B}_t = \begin{bmatrix} 0 \\ 0.1 \end{bmatrix}$$

$$\boldsymbol{b}_t = \begin{bmatrix} 5 \\ 0 \end{bmatrix} \qquad \boldsymbol{\Sigma}_t = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

$$\boldsymbol{K}_t = \begin{bmatrix} 5 & 0.3 \end{bmatrix} \qquad k_t = 0.3$$

$$\boldsymbol{H}_t = 1 \qquad \boldsymbol{R}_t = \begin{cases} \begin{bmatrix} 100000 & 0 \\ 0 & 0.1 \end{bmatrix} & \text{if} \quad t = 14 \text{ or } 40 \\ \begin{bmatrix} 0.01 & 0 \\ 0 & 0.1 \end{bmatrix} & \text{otherwise} \end{cases} \qquad \boldsymbol{r}_t = \begin{cases} \begin{bmatrix} 10 \\ 0 \end{bmatrix} & \text{if} \quad t = 0, 1, \ldots, 14 \\ \begin{bmatrix} 20 \\ 0 \end{bmatrix} & \text{if} \quad t = 15, 16, \ldots, T \end{cases}$$

Execute the system 20 times. Plot the mean and 95% confidence (see "68–95–99.7 rule" and matplotlib.pyplot.fill_between function) over the different experiments of the state $\boldsymbol{s}_t$ and of the control signal $\boldsymbol{a}_t$ over time. How does the system behave? Compute and write down the mean and the standard deviation of the cumulative reward over the experiments. Attach a snippet of your code.

b) LQR as a P controller [4 Points]

The LQR can also be seen as a simple P controller of the form

$$a_t = \boldsymbol{K}_t \left(\boldsymbol{s}_t^{\text{des}} - \boldsymbol{s}_t \right) + k_t \,, \tag{4}$$

which corresponds to the controller used in the canonical LQR system with the introduction of the target $\boldsymbol{s}_t^{\text{des}}$. Assume as target

$$\boldsymbol{s}_t^{\text{des}} = \boldsymbol{r}_t = \begin{cases} \begin{bmatrix} 10 \\ 0 \end{bmatrix} & \text{if} \quad t = 0, 1, \ldots, 14 \\ \begin{bmatrix} 20 \\ 0 \end{bmatrix} & \text{if} \quad t = 15, 16, \ldots, T \end{cases} \tag{5}$$

Use the same LQR system as in the previous exercise and run 20 experiments. Plot in one figure the mean and 95% confidence (see "68–95–99.7 rule" and matplotlib.pyplot.fill_between function) of the first dimension of the state, for both $\boldsymbol{s}_t^{\text{des}} = \boldsymbol{r}_t$ and $\boldsymbol{s}_t^{\text{des}} = \boldsymbol{0}$.

c) Optimal LQR [8 Points]

To compute the optimal gains $\boldsymbol{K}_t$ and $\boldsymbol{k}_t$, which maximize the cumulative reward, we can use an analytic optimal solution. This controller recursively computes the optimal action by

$$a_t^* = -\left(\boldsymbol{H}_t + \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{B}_t\right)^{-1} \boldsymbol{B}_t^T \left(\boldsymbol{V}_{t+1} \left(\boldsymbol{A}_t \boldsymbol{s}_t + \boldsymbol{b}_t\right) - v_{t+1}\right), \tag{6}$$

which can be decomposed into

$$\boldsymbol{K}_t = -\left(\boldsymbol{H}_t + \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{B}_t\right)^{-1} \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{A}_t, \tag{7}$$

$$\boldsymbol{k}_t = -\left(\boldsymbol{H}_t + \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{B}_t\right)^{-1} \boldsymbol{B}_t^T \left(\boldsymbol{V}_{t+1} \boldsymbol{b}_t - v_{t+1}\right). \tag{8}$$

where

$$\boldsymbol{M}_t = \boldsymbol{B}_t \left(\boldsymbol{H}_t + \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{B}_t\right)^{-1} \boldsymbol{B}_t^T \boldsymbol{V}_{t+1} \boldsymbol{A}_t \tag{9}$$

$$\boldsymbol{V}_t = \begin{cases} \boldsymbol{R}_t + \left(\boldsymbol{A}_t - \boldsymbol{M}_t\right)^T \boldsymbol{V}_{t+1} \boldsymbol{A}_t & \text{when} \quad t = 1 \ldots T-1 \\ \boldsymbol{R}_t & \text{when} \quad t = T \end{cases} \tag{10}$$

$$v_t = \begin{cases} \boldsymbol{R}_t \boldsymbol{r}_t + \left(\boldsymbol{A}_t - \boldsymbol{M}_t\right)^T \left(v_{t+1} - \boldsymbol{V}_{t+1} \boldsymbol{b}_t\right) & \text{when} \quad t = 1 \ldots T-1 \\ \boldsymbol{R}_t \boldsymbol{r}_t & \text{when} \quad t = T \end{cases} \tag{11}$$

Run 20 experiments with: the controller from a); the P controller from b) with $\boldsymbol{s}_t^{\text{des}} = \boldsymbol{r}_t$ (with $\boldsymbol{r}_t$ as defined in a)); and with the controller c) resulting from computing the optimal gains $\boldsymbol{K}_t$ and $\boldsymbol{k}_t$. Plot the mean and 95% confidence (see "68–95–99.7 rule" and matplotlib.pyplot.fill_between function) of both states for all three different controllers used so far. Use one figure per state. Report the mean and std of the cumulative reward for each controller and comment the results. Attach a snippet of your code.