RL HOMEWORK ASSIGNMENT 3

- By Jainam Kothari(jdk9165)

November 15, 2024

Exercise 1 [40 points]

a) 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:

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

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.

b) What is the sequence of control actions, states, and the optimal cost if $x_0 = 0$, if $x_0 = -2$, and if $x_0 = 2$.

c) Assume now that the constant term 1 in the previous dynamics can sometimes be 0 with a probability of 0.4. We can now write the dynamics as:

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

where $x_n \in \{-2, -1, 0, 1, 2\}$, $u_n \in \{-1, 0, 1\}$, and $\omega_n \in \{0, 1\}$ is a random variable with probability distribution $p(\omega_n = 0) = 0.4$, $p(\omega_n = 1) = 0.6$. The cost function to minimize becomes:

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

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.

ANSWER

1 A)
Here in the question we are given
State Range: $x \in \{-2, -1, 0, 1, 2\}$

Control Range : $u \in \{-1, 0, 1\}$

Horizon: $N = 3$ (i.e., three steps, $n = 0, 1, 2$, with the terminal cost at step 3)
Cost and Transition Equations

The state transition function can be defined as this :

$$x_{n+1} = -x_n + 1 + u_n$$

- This function determines the next state based on the current state x_n and control u_n.
- To ensure x_{n+1} remains in $[-2, 2]$, we cap it as follows:

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

From the The cost function to minimize :

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

Using backward dynamic programming, we solve the finite-horizon optimal control problem by minimizing the cost-to-go at each stage. This involves computing the cost-to-go function $J_n(x_n)$ for each stage n and each state x_n, as follows:

$$J_n(x_n) = \min_{u_n} (g(x_n, u_n) + (J_{n+1}(f(x_n, u_n))$$

where:

- $g(x_n, u_n) = 2|x_n| + |u_n|$ is the immediate cost.

- $J_{n+1}(f(x_n, u_n$ is the cost-to-go for the next state, determined by the dynamics.
Now we can compute the cost function and the control value.
Step 3 : Terminal Step $n = 3$
At the terminal stage, the cost-to-go function $J_3(x)$ is simply the terminal cost:

$$J_3(x) = \text{cost-to-go[3][x]} = x^2$$

Thus, we have:

State x	**Terminal Cost** $J_3(x)$
-2	4
-1	1
0	0
1	1
2	4

Table 1: Terminal Costs for Each State

Step 2: Backward Induction for Step $n = 2$
For each state x at step $n = 2$, we calculate the total cost for each control u, then select the control that minimizes this cost.

1. **State** $x = -2$:

   - **Control** $u = -1$:

$$x_{\text{next}} = -(-2) + 1 + (-1) = 2$$

$$\text{total cost} = (2 \cdot |-2| + |-1|) + J_3(f(x, u)) = \text{total cost} = (2 \cdot |-2| + |-1|) + \text{cost-to-go[3][2]} = (4+1) + 4 = 9$$

- **Control** $u = 0$:

$$x_{\text{next}} = -(-2) + 1 + 0 = 3 \to 2 \text{ (capped)}$$

$$\text{total cost} = (4 + 0) + 4 = 8$$

- **Control** $u = 1$:

$$x_{\text{next}} = -(-2) + 1 + 1 = 4 \to 2 \text{ (capped)}$$

$$\text{total cost} = (4 + 1) + 4 = 9$$

- **Optimal control**: $u = 0$, minimum cost 8.

2. **State** $x = -1$:

- **Control** $u = -1$:

$$x_{\text{next}} = -(-1) + 1 + (-1) = 1$$

$$\text{total cost} = (2 + 1) + 1 = 4$$

- **Control** $u = 0$:

$$x_{\text{next}} = -(-1) + 1 + 0 = 2$$

$$\text{total cost} = (2 + 0) + 4 = 6$$

- **Control** $u = 1$:

$$x_{\text{next}} = -(-1) + 1 + 1 = 3 \to 2$$

$$\text{total cost} = (2 + 1) + 4 = 7$$

- **Optimal control**: $u = -1$, minimum cost 4.

3. **State** $x = 0$:

- **Control** $u = -1$:

$$x_{\text{next}} = -(0) + 1 + (-1) = 0$$

$$\text{total cost} = (0 + 1) + 0 = 1$$

- **Control** $u = 0$:

$$x_{\text{next}} = -(0) + 1 + 0 = 1$$

$$\text{total cost} = (0 + 0) + 1 = 2$$

- **Control** $u = 1$:

$$x_{\text{next}} = -(0) + 1 + 1 = 2$$

$$\text{total cost} = (0 + 1) + 4 = 5$$

- **Optimal control**: $u = -1$, minimum cost 1.

4. **State** $x = 1$:

   - **Control** $u = -1$:

$$x_{\text{next}} = -(1) + 1 + (-1) = -1$$

$$\text{total cost} = (2 + 1) + 1 = 4$$

   - **Control** $u = 0$:

$$x_{\text{next}} = -(1) + 1 + 0 = 0$$

$$\text{total cost} = (2 + 0) + 0 = 2$$

   - **Control** $u = 1$:

$$x_{\text{next}} = -(1) + 1 + 1 = 1$$

$$\text{total cost} = (2 + 1) + 1 = 3$$

   - **Optimal control**: $u = 0$, minimum cost 2.

5. **State** $x = 2$:

   - **Control** $u = -1$:

$$x_{\text{next}} = -(2) + 1 + (-1) = -2$$

$$\text{total cost} = (4 + 1) + 4 = 9$$

   - **Control** $u = 0$:

$$x_{\text{next}} = -(2) + 1 + 0 = -1$$

$$\text{total cost} = (4 + 0) + 1 = 5$$

   - **Control** $u = 1$:

$$x_{\text{next}} = -(2) + 1 + 1 = 0$$

$$\text{total cost} = (4 + 0) + 1 = 5$$

   - **Optimal control**: $u = 0$, minimum cost 5.

Step 1 and Step 0: Backward Induction for Step $n = 1$ Step $n = 0$

Applying similar calculations for each state, we continue back to step $n = 1$ and step $n = 0$, each time choosing the control that minimizes the total cost based on the cost-to-go at the subsequent stage.

Table of Optimal Cost-to-Go and Controls

Step n	**State** x	**Cost-to-Go** $J(x)$	**Optimal Control** u
0	-2	10	0
0	-1	6	-1
0	0	3	-1
0	1	4	0
0	2	7	1
1	-2	9	0
1	-1	5	-1
1	0	2	-1
1	1	3	0
1	2	6	1
2	-2	8	0
2	-1	4	-1
2	0	1	-1
2	1	2	0
2	2	5	0
3 (Terminal)	-2	4	-
3 (Terminal)	-1	1	-
3 (Terminal)	0	0	-
3 (Terminal)	1	1	-
3 (Terminal)	2	4	-

Table 2: Optimal Cost-to-Go and Controls for Each Step and State

Exercise: 1 (b)

Question

What is the sequence of control actions, states, and the optimal cost if $x_0 = 0$, if $x_0 = -2$, and if $x_0 = 2$?

Answer

To find the sequence of optimal control actions, we will simply: 1. **Find the optimal control for the current state** 2. **Apply the optimal control to the current state and obtain the next state** 3. **Repeat the above steps until we reach the end**

For the first case with $x_0 = 0$:

(a) For $x_0 = 0$, $u_0 = -1 \rightarrow x_1 = f(x_0, u_0) = 0$ with $J_0 = 3$
(b) For $x_1 = 0$, $u_1 = -1 \rightarrow x_2 = f(x_1, u_1) = 0$ with $J_1 = 2$
(c) For $x_2 = 0$, $u_2 = -1 \rightarrow x_3 = f(x_2, u_2) = 0$ with $J_2 = 1$
(d) For $x_3 = 0$, $J = 0$

Thus, we reached the end and minimized the cost to 0.

For the second case with $x_0 = -2$:

(a) For $x_0 = -2$, $u_0 = 0 \rightarrow x_1 = f(x_0, u_0) = 2$ with $J_0 = 10$

(b) For $x_1 = 2$, $u_1 = 1 \rightarrow x_2 = f(x_1, u_1) = 0$ with $J_1 = 6$

(c) For $x_2 = 0$, $u_2 = -1 \rightarrow x_3 = f(x_2, u_2) = 0$ with $J_2 = 1$

(d) For $x_3 = 0$, $J_3 = 0$

Thus, we reached the end and minimized the cost to 0.

For the third case with $x_0 = 2$:

(a) For $x_0 = 2$, $u_0 = 1 \rightarrow x_1 = f(x_0, u_0) = 0$ with $J_0 = 7$

(b) For $x_1 = 0$, $u_1 = -1 \rightarrow x_2 = f(x_1, u_1) = 0$ with $J_1 = 2$

(c) For $x_2 = 0$, $u_2 = -1 \rightarrow x_3 = f(x_2, u_2) = 0$ with $J_2 = 1$

(d) For $x_3 = 0$, $J_3 = 0$

Thus, we reached the end and minimized the cost to 0.

1 C)

To minimize the total expected cost in this finite-horizon optimal control problem, we use dynamic programming to determine the optimal control u_n for each state x_n at each stage n, backward in time. Here in the question The system dynamics and the cost function are as follows:

1. System Dynamics:

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

where $x_n \in \{-2, -1, 0, 1, 2\}$, $u_n \in \{-1, 0, 1\}$, and $\omega_n \in \{0, 1\}$ with probabilities $p(\omega_n = 0) = 0.4$ and $p(\omega_n = 1) = 0.6$.

2. Cost Function: The total cost to minimize is:

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

Dynamic Programming Approach

Using backward dynamic programming, we solve the finite-horizon optimal control problem by minimizing the cost-to-go at each stage. This involves computing the cost-to-go function $J_n(x_n)$ for each stage n and each state x_n, as follows:

$$J_n(x_n) = \min_{u_n}\left(g(x_n, u_n) + E(J_{n+1}(f(x_n, u_n, \omega_n)))\right)$$

where:

- $g(x_n, u_n) = 2|x_n| + |u_n|$ is the immediate cost.

- $J_{n+1}(f(x_n, u_n, \omega_n))$ is the cost-to-go for the next state, determined by the dynamics.

Now we can compute the cost function and the control value.

Step 3 : Terminal Step $n = 3$

At the terminal stage, the cost-to-go function $J_3(x)$ is simply the terminal cost:

$$J_3(x) = x^2$$

Thus, we have:

State x	**Terminal Cost** $J_3(x)$
-2	4
-1	1
0	0
1	1
2	4

Table 3: Terminal Costs for Each State

Step 2: $n = 2$

To find the optimal cost-to-go $J(x)$ and optimal control u for each state x at stage $n = 2$, we perform calculations for each possible control $u \in \{-1, 0, 1\}$ and take the expectation of the resulting costs based on the probabilities of $\omega \in \{0, 1\}$.

1. For $x = -2$:

Control $u = -1$**:**

$$J_2(-2) = g(-2, -1) + p(0)J_3(f(-2, -1, 0)) + p(1)J_3(f(-2, -1, 1))$$

Substituting values:

$$J_2(-2) = 5 + 0.4 \times J_3(1) + 0.6 \times J_3(2) = 5 + 0.4 \times 1 + 0.6 \times 4 = 7.8$$

Control $u = 0$**:**

$$J_2(-2) = g(-2, 0) + p(0)J_3(f(-2, 0, 0)) + p(1)J_3(f(-2, 0, 1))$$

Substituting values:

$$J_2(-2) = 4 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 4 + 0.4 \times 4 + 0.6 \times 4 = 9$$

Control $u = 1$**:**

$$J_2(-2) = g(-2, 1) + p(0)J_3(f(-2, 1, 0)) + p(1)J_3(f(-2, 1, 1))$$

Substituting values:

$$J_2(-2) = 5 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 5 + 0.4 \times 4 + 0.6 \times 4 = 9$$

Minimum Cost: $J_2(-2) = 7.8$ with $u = -1$.

2. For $x = -1$:

Control $u = -1$**:**

$$J_2(-1) = g(-1, -1) + p(0)J_3(f(-1, -1, 0)) + p(1)J_3(f(-1, -1, 1))$$

Substituting values:

$$J_2(-1) = 3 + 0.4 \times J_3(0) + 0.6 \times J_3(1) = 3 + 0.4 \times 0 + 0.6 \times 1 = 3.6$$

Control $u = 0$:

$$J_2(-1) = g(-1, 0) + p(0)J_3(f(-1, 0, 0)) + p(1)J_3(f(-1, 0, 1))$$

Substituting values:

$$J_2(-1) = 2 + 0.4 \times J_3(1) + 0.6 \times J_3(2) = 2 + 0.4 \times 1 + 0.6 \times 4 = 4.8$$

Control $u = 1$:

$$J_2(-1) = g(-1, 1) + p(0)J_3(f(-1, 1, 0)) + p(1)J_3(f(-1, 1, 1))$$

Substituting values:

$$J_2(-1) = 3 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 3 + 0.4 \times 4 + 0.6 \times 4 = 7$$

Minimum Cost: $J_2(-1) = 3.6$ with $u = -1$.

3. For $x = 0$:

Control $u = -1$:

$$J_2(0) = g(0, -1) + p(0)J_3(f(0, -1, 0)) + p(1)J_3(f(0, -1, 1))$$

Substituting values:

$$J_2(0) = 1 + 0.4 \times J_3(-1) + 0.6 \times J_3(0) = 1 + 0.4 \times 1 + 0.6 \times 0 = 1.4$$

Control $u = 0$:

$$J_2(0) = g(0, 0) + p(0)J_3(f(0, 0, 0)) + p(1)J_3(f(0, 0, 1))$$

Substituting values:

$$J_2(0) = 0 + 0.4 \times J_3(0) + 0.6 \times J_3(1) = 0 + 0.4 \times 0 + 0.6 \times 1 = 0.6$$

Control $u = 1$:

$$J_2(0) = g(0, 1) + p(0)J_3(f(0, 1, 0)) + p(1)J_3(f(0, 1, 1))$$

Substituting values:

$$J_2(0) = 1 + 0.4 \times J_3(1) + 0.6 \times J_3(2) = 1 + 0.4 \times 1 + 0.6 \times 4 = 3.4$$

Minimum Cost: $J_2(0) = 0.6$ with $u = 0$.

4. For $x = 1$:

Control $u = -1$:

$$J_2(1) = g(1, -1) + p(0)J_3(f(1, -1, 0)) + p(1)J_3(f(1, -1, 1))$$

Substituting values:

$$J_2(1) = 3 + 0.4 \times J_3(0) + 0.6 \times J_3(1) = 3 + 0.4 \times 0 + 0.6 \times 1 = 3.6$$

Control $u = 0$:

$$J_2(1) = g(1, 0) + p(0)J_3(f(1, 0, 0)) + p(1)J_3(f(1, 0, 1))$$

Substituting values:

$$J_2(1) = 2 + 0.4 \times J_3(1) + 0.6 \times J_3(2) = 2 + 0.4 \times 1 + 0.6 \times 4 = 4.8$$

Control $u = 1$**:**

$$J_2(1) = g(1, 1) + p(0)J_3(f(1, 1, 0)) + p(1)J_3(f(1, 1, 1))$$

Substituting values:

$$J_2(1) = 3 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 3 + 0.4 \times 4 + 0.6 \times 4 = 7$$

Minimum Cost: $J_2(1) = 3.6$ with $u = -1$.

5. For $x = 2$:

Control $u = -1$**:**

$$J_2(2) = g(2, -1) + p(0)J_3(f(2, -1, 0)) + p(1)J_3(f(2, -1, 1))$$

Substituting values:

$$J_2(2) = 5 + 0.4 \times J_3(1) + 0.6 \times J_3(2) = 5 + 0.4 \times 1 + 0.6 \times 4 = 7.8$$

Control $u = 0$**:**

$$J_2(2) = g(2, 0) + p(0)J_3(f(2, 0, 0)) + p(1)J_3(f(2, 0, 1))$$

Substituting values:

$$J_2(2) = 4 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 4 + 0.4 \times 4 + 0.6 \times 4 = 9$$

Control $u = 1$**:**

$$J_2(2) = g(2, 1) + p(0)J_3(f(2, 1, 0)) + p(1)J_3(f(2, 1, 1))$$

Substituting values:

$$J_2(2) = 5 + 0.4 \times J_3(2) + 0.6 \times J_3(2) = 5 + 0.4 \times 4 + 0.6 \times 4 = 9$$

Minimum Cost: $J_2(2) = 7.8$ with $u = -1$.

Step 1 and Step 0 :

Similarly, we can perform these calculations for Steps $n = 1$ and $n = 0$ using the same method and then compile the values into a table format.

Here is the complete table for the cost to go and the optimal control

Step n	**State** x	**Cost-to-Go** $J(x)$	**Optimal Control** u
0	-2	10.600000000000001	-1
0	-1	5.952	-1
0	0	2.952	0
0	1	4.880000000000001	0
0	2	7.88	1
1	-2	9.200000000000001	-1
1	-1	4.680000000000001	-1
1	0	1.6800000000000002	0
1	1	3.8	0
1	2	6.8	1
2	-2	7.8	-1
2	-1	3.6	-1
2	0	0.6	0
2	1	2.4000000000000004	0
2	2	5.4	1
Step 3 (Terminal)	**State** x	**Cost-to-Go** $J(x)$	**Optimal Control** u
3	-2	4	-
3	-1	1	-
3	0	0	-
3	1	1	-
3	2	4	-

Table 4: Terminal Cost-to-Go for Step 3

1 D)

ANSWER

The main difference are given here :

Key Differences

Values of J

The expected cost-to-go values (J^*) under the deterministic policy (Table 1) are generally higher (more positive) compared to the stochastic policy (Table 2). This indicates that the deterministic policy results in higher expected costs than the stochastic policy for all states. The deterministic policy also shows a wider range of J values. For instance, in state -2, the highest J value is 10, while it is smaller for the stochastic policy (9.8).

Optimal Actions (u^*)

The optimal actions change between deterministic and stochastic policies. In the deterministic case, actions are more constrained (hence more predictable), represented by single actions chosen for each state that lead to the associated best value. The stochastic policy lists multiple optimal actions for each state, where there may not be a single choice but rather several actions that could yield good expected values. Thus u_i^* shows a more varied set of choices compared to deterministic policies.

Stability of Values

The entries in Table 2 show that provided actions yield more moderate J values, suggesting that a stochastic policy integrates more flexibility and robustness against uncertainties compared to the deterministic scenario. These lower values in the stochastic policy could reflect the notion of averaging over multiple possible outcomes, which typically results in a more conservative estimate of costs.