Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for Dynamic Systems and Control
IDSC
Institut für Dynamische Systeme und Regelungstechnik

Quiz 1 **October 24th, 2012**

Dynamic Programming & Optimal Control (151-0563-00)

Angela Schoellig

Solutions

Duration:	**45 minutes**
Number of Problems:	**2**
Permitted Aids:	None. Use only the prepared sheets for your solutions.

Problem 1 **20%**

Given is the following system equation

$$x_{k+1} = x_k + 2\,x_{k-2}^2 - x_{k-3} + 3\,u_k^2 - u_{k-1} + w_k\,u_k\,x_k$$

with $x_k,\ u_k,\ w_k \in \mathbb{R}$ and $w_k \sim P(\,\cdot \mid x_k, u_k)$.

To apply the Dynamic Programming algorithm, the system must be reformulated into the basic problem format:

$$\tilde{x}_{k+1} = \tilde{f}_k(\tilde{x}_k, u_k, w_k)$$

with $\tilde{x}_k = [\tilde{x}_{k,1}, \dots, \tilde{x}_{k,m}]^T \in \mathbb{R}^m$ being the augmented state.

a) What dimension m does the augmented state $\tilde{x}_k$ have?

b) Please circle the correct representation of the augmented state $\tilde{x}_k$.

$$\tilde{x}_k = \begin{bmatrix} x_k \\ u_k \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \end{bmatrix}$$

$$\tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \\ u_k \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \\ u_k \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_k \end{bmatrix}$$

$$\tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \\ u_{k-1} \\ w_k \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-2} \\ x_{k-3} \\ u_k \\ u_{k-1} \\ w_k \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_k \\ u_{k-1} \end{bmatrix} \qquad \tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_k \\ u_{k-1} \\ w_k \end{bmatrix}$$

c) Rewrite the given system in the form $\tilde{x}_{k+1} = \tilde{f}_k(\tilde{x}_k, u_k, w_k)$ using the augmented state $\tilde{x}_k = [\tilde{x}_{k,1}, \dots, \tilde{x}_{k,m}]^T$.

Solution 1

a) 5

b) $$\tilde{x}_k = \begin{bmatrix} x_k \\ x_{k-1} \\ x_{k-2} \\ x_{k-3} \\ u_{k-1} \end{bmatrix} = \begin{bmatrix} \tilde{x}_{k,1} \\ \tilde{x}_{k,2} \\ \tilde{x}_{k,3} \\ \tilde{x}_{k,4} \\ \tilde{x}_{k,5} \end{bmatrix}$$

c) $$\tilde{x}_{k+1} = \begin{bmatrix} \tilde{x}_{k,1} + 2\tilde{x}_{k,3}^2 - \tilde{x}_{k,4} + 3u_k^2 - \tilde{x}_{k,5} + w_k u_k \tilde{x}_{k,1} \\ \tilde{x}_{k,1} \\ \tilde{x}_{k,2} \\ \tilde{x}_{k,3} \\ u_k \end{bmatrix}$$

Problem 2 **80%**

Consider the system

$$x_{k+1} = x_k + u_k + w_k, \qquad k = 0, 1\,,$$

with initial state $x_0 = 0$. The disturbance w_k takes values -1 and 1 with equal probability. The cost for the above system is given by

$$(x_2 - x_2^*)^2 + (x_1 - x_1^*)^2 + u_1^2 + u_0^2$$

with $x_1^* = 2$ and $x_2^* = 1$.

a) Assume a discrete input u_k, $u_k \in \{-1, 0, 1\}$ for $k = 0, 1$:

- i) Find all states x_k that can be reached from $x_0 = 0$ in stage 1 ($k = 1$) and in stage 2 ($k = 2$).
- ii) Calculate the optimal cost-to-go $J_0(x_0)$ and the optimal policy $\{\mu_0^*(x_0), \mu_1^*(x_1)\}$.
- iii) Will the optimal policy be affected by adding $(x_0 - x_0^*)^2$, $x_0^* = 1$, to the cost? If so, how?

b) Assume a continuous input u_k, $u_k \in [-1, 1]$ for $k = 0, 1$:

- i) Calculate the optimal cost-to-go $J_1(x_1)$ and the optimal policy $\mu_1^*(x_1)$ for $x_1 = 2,\ 1,\ -2$.

Solution 2

a) Discrete input case:

i) State evolution for $k = 1$ and 2:
For k=1 and $x_0 = 0$,

$$\begin{aligned}
\{u_0 = 1, w_0 = 1\} &\quad\Rightarrow\quad x_1 = 2\\
\{u_0 = 0, w_0 = 1\} &\quad\Rightarrow\quad x_1 = 1\\
\{u_0 = 1, w_0 = -1\} \text{ or } \{u_0 = -1, w_0 = 1\} &\quad\Rightarrow\quad x_1 = 0\\
\{u_0 = 0, w_0 = -1\} &\quad\Rightarrow\quad x_1 = -1\\
\{u_0 = -1, w_0 = -1\} &\quad\Rightarrow\quad x_1 = -2
\end{aligned}$$

Similarly for $k = 2$,

$$\begin{aligned}
x_1 = 2 &\quad\Rightarrow\quad x_2 \in \{4, 3, 2, 1, 0\}\\
x_1 = 1 &\quad\Rightarrow\quad x_2 \in \{3, 2, 1, 0, -1\}\\
x_1 = 0 &\quad\Rightarrow\quad x_2 \in \{2, 1, 0, -1, -2\}\\
x_1 = -1 &\quad\Rightarrow\quad x_2 \in \{1, 0, -1, -2, -3\}\\
x_1 = -2 &\quad\Rightarrow\quad x_2 \in \{0, -1, -2, -3, -4\}.
\end{aligned}$$

This implies:
$x_1 \in \{2, 1, 0, -1, -2\}$ and $x_2 \in \{4, 3, 2, 1, 0, -1, -2, -3, -4\}$.

ii) Optimal cost-to-go and the optimal policy:
Calculate the terminal cost $J_2(x_2) = (x_2 - x_2^*)^2$, for all possible states at $k = 2$.

$$\begin{array}{ccc}
J_2(4) = 9 & J_2(3) = 4 & J_2(2) = 1\\
J_2(1) = 0 & J_2(0) = 1 & J_2(-1) = 4\\
J_2(-2) = 9 & J_2(-3) = 16 & J_2(-4) = 25.
\end{array}$$

Now, using the the Dynamic Programming Algorithm for $k = 1$,

$$J_1(x_1) = \min_{u_1 \in \{-1,0,1\}} \underset{w_1}{E}\Big\{(x_1 - x_1^*)^2 + u_1^2 + J_2\big(x_1 + u_1 + w_1\big)\Big\},$$

gives

$$\begin{aligned}
J_1(2) = 2 &\quad \text{with} \quad \mu_1^*(2) = 0 \text{ or } -1,\\
J_1(1) = 2 &\quad \text{with} \quad \mu_1^*(1) = 0,\\
J_1(0) = 6 &\quad \text{with} \quad \mu_1^*(0) = 0 \text{ or } 1,\\
J_1(-1) = 12 &\quad \text{with} \quad \mu_1^*(-1) = 1,\\
J_1(-2) = 22 &\quad \text{with} \quad \mu_1^*(-2) = 1.
\end{aligned}$$

Similarly, using the the Dynamic Programming Algorithm for $k = 0$,

$$J_0(x_0) = \min_{u_0 \in \{-1,0,1\}} \underset{w_0}{E}\Big\{u_0^2 + J_1\big(x_0 + u_0 + w_0\big)\Big\},$$

gives

$$J_0(0) = 5 \quad \text{with} \quad \mu_0^*(2) = 1.$$

iii) Effect of adding $(x_0 - x_0^*)^2$: There won't be any effect on the optimal policy since it is a fixed constant value.

b) Continuous input case where $u_k \in [-1, 1]$ for $k = 0, 1$:

i) Optimal cost-to-go and the optimal policy for $x_1 = 2, 1, \text{and} -2$
Terminal cost: $(x_2 - x_2^*)^2 = J_2(x_2)$
Now, using the the Dynamic Programming Algorithm for $k = 1$ gives

$$\begin{aligned} J_1(x_1) &= \min_{u_1 \in [-1,1]} \mathbb{E}_{w_1}\{(x_1 - x_1^*)^2 + u_1^2 + J_2(x_1 + u_1 + w_1)\} \\ &= \min_{u_1 \in [-1,1]} \{(x_1 - x_1^*)^2 + u_1^2 + 0.5(x_1 + u_1 + 1 - x_2^*)^2 \\ &\qquad + 0.5(x_1 + u_1 - 1 - x_2^*)^2\}. \end{aligned} \quad (1)$$

Since u_1^2 in (1) has a positive coefficient, the minimization can be done by setting the derivative with respect to u_1 to zero.

$$0 = 4u_1 + 2x_1 - 2x_2^*.$$

This implies,

$$\mu_1(x_1) = \frac{x_2^* - x_1}{2}. \quad (2)$$

Now, for $x_1 = 2$ and 1 a straight forward substitution yields,

$$\begin{aligned} \mu_1^*(x_1 = 2) = -0.5 &\qquad J_1(x_1 = 2) = 1.5 \\ \mu_1^*(x_1 = 1) = 0 &\qquad J_1(x_1 = 1) = 2 \end{aligned}$$

For $x_1 = -2$ an additional step is required since $\mu_1(x_1 = -2) = 1.5$ is not a feasible input. Since the function to be minimized in (1) is monotonically decreasing in u_1's feasible range $[-1, 1]$, $\mu_1(x_1 = -2) = 1$ is the feasible minimizer. Finally,

$$\mu_1^*(x_1 = -2) = 1 \qquad \Longrightarrow \qquad J_1(x_1 = -2) = 22.$$