Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

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

Final Exam **January 30th, 2016**

Dynamic Programming & Optimal Control (151-0563-01) Prof. R. D'Andrea

Solutions

Exam Duration: **150 minutes**

Number of Problems: **4**

Permitted aids: One A4 sheet of paper.

No calculators allowed.

Problem 1 25%

Perform the first iteration of the Dynamic Programming Algorithm for each of the following problems. In particular, define the **state variable** x_k and the **system dynamics** $f_k(x_k, u_k, w_k)$, compute the **final cost** $J_k(x_k)$ for $k = N$, compute the **cost-to-go** $J_k(x_k)$ for $k = N-1$, and compute an **optimal policy** $\mu_k(x_k)$ for $k = N-1$.

a) The system dynamics are given by

$$y_{k+1} = y_k u_k + y_{k-1} w_k, \quad k = 0, 1, \ldots, N-1,$$

with $y_0 = 0$, $y_{-1} = 0$, and input $u_k \in \mathbb{R}$ (continuous input space). The disturbance w_k takes the value 0 or 1 with equal probability. The cost function to be minimized is given by

$$\text{cost} = \mathrm{E}\left\{ \sum_{k=0}^{N-1} k(y_k^2 + u_k^2) \right\}.$$

b) The system dynamics are given by

$$y_{k+1} = y_k + u_k + w_k, \quad k = 0, 1, \ldots, N-1,$$

with $y_0 = 0$ and input $u_k \in [0, 1]$ (continuous input space). The disturbance w_k takes the value 1 with probability u_k^2 and the value y_k with probability $1 - u_k^2$. The cost function to be minimized is given by

$$\text{cost} = \mathrm{E}\left\{ y_N \right\}.$$

c) The system dynamics are given by

$$y_{k+1} = y_k + u_k w_k, \quad k = 0, 1, \ldots, N-1,$$

with $y_0 = 0$ and input $u_k \in \mathbb{R}$ (continuous input space). The disturbance w_k is a random walk with $\mathrm{E}\{w_k - w_{k-1}\} = 0$ and $\mathrm{Var}\{w_k - w_{k-1}\} = 1$. We assume that after each time step k we can observe the value of the disturbance w_k (i.e. at time step k we know the value of the previous disturbance w_{k-1}), and we assume that the random walk starts at $w_{-1} = 0$. The cost function to be minimized is given by

$$\text{cost} = \mathrm{E}\left\{ y_N^2 + \sum_{k=0}^{N-1} y_k^2 + u_k^2 \right\}.$$

Solution 1

a)

- State variable: $x_k = \begin{bmatrix} y_k \\ y_{k-1} \end{bmatrix}$
- System dynamics: $f_k(x_k) = \begin{bmatrix} y_k u_k + y_{k-1} w_k \\ y_k \end{bmatrix}$
- Final cost: $J_N(x_N) = 0$
- Cost-to-go at $k = N-1$:
  We apply the dynamic programming recursion formula to compute the cost-to-go at time step $k = N-1$:

$$\begin{aligned} J_k(x_k) &= \min_{u_k \in \mathbb{R}} \mathop{\mathrm{E}}_{w_k} \left\{ k(y_k^2 + u_k^2) + J_{k+1}(x_{k+1}) \right\} \\ &= \min_{u_k \in \mathbb{R}} \left[k(y_k^2 + u_k^2) \right] \\ &\Rightarrow \underline{\underline{J_k(x_k) = k y_k^2}} \text{ with } \underline{\underline{\mu_k(x_k) = 0}} \end{aligned}$$

b)

- State variable: $x_k = y_k$
- System dynamics: $f_k(x_k) = y_k + u_k + w_k$
- Final cost: $J_N(x_N) = y_N$
- Cost-to-go at $k = N-1$:
  We apply the dynamic programming recursion formula to compute the cost-to-go at time step $k = N-1$:

$$\begin{aligned} J_k(x_k) &= \min_{u_k \in [0,1]} \mathop{\mathrm{E}}_{w_k} \left\{ J_{k+1}(x_{k+1}) \right\} \\ &= \min_{u_k \in [0,1]} \mathop{\mathrm{E}}_{w_k} \left\{ y_k + u_k + w_k \right\} \\ &= \min_{u_k \in [0,1]} \left[y_k + u_k + u_k^2 + y_k(1 - u_k^2) \right] \\ &= \min_{u_k \in [0,1]} \left[2y_k + u_k + (1 - y_k) u_k^2 \right] \end{aligned}$$

For $y_k \geq 1$, the coefficient of u_k^2 is negative or zero and the minimum is obtained at a boundary:

$$J_k(x_k) = \min\left[2y_k, 2 + y_k\right] \Rightarrow u_k^* = \begin{cases} 0 & \text{if } 1 \leq y_k \leq 2 \\ 1 & \text{if } y_k > 2 \end{cases}$$

For $y_k < 1$, the coefficient of u_k^2 is positive and the minimum is obtained either at a boundary or where the derivative vanishes:

$$\frac{d}{du_k} = 1 + 2(1 - y_k) u_k \overset{!}{=} 0 \Rightarrow u_k = \frac{1}{2y_k - 2}$$

This expression for u_k is negative for any $y_k < 1$, hence the optimal input is at the lower boundary, i.e. $u_k^* = 0$.
Hence, the cost-to-go and optimal policy is

$$\underline{\underline{J_k(x_k) = \begin{cases} 2y_k & \text{with } \mu_k(x_k) = 0 \quad \text{if } y_k \leq 2 \\ 2 + y_k & \text{with } \mu_k(x_k) = 1 \quad \text{if } y_k > 2 \end{cases}}}$$

c)

- State variable: $x_k = \begin{bmatrix} y_k \\ w_{k-1} \end{bmatrix}$
- System dynamics: $f_k(x_k) = \begin{bmatrix} y_k + u_k w_k \\ w_k \end{bmatrix}$
- Final cost: $J_N(x_N) = y_N^2$
- Cost-to-go at $k = N-1$:
  We apply the dynamic programming recursion formula to compute the cost-to-go at time step $k = N-1$:

$$\begin{aligned}
J_k(x_k) &= \min_{u_k \in \mathbb{R}} \mathop{\mathrm{E}}_{w_k} \left\{ y_k^2 + u_k^2 + J_{k+1}(x_{k+1}) \right\} \\
&= \min_{u_k \in \mathbb{R}} \mathop{\mathrm{E}}_{w_k} \left\{ y_k^2 + u_k^2 + (y_k + u_k w_k)^2 \right\} \\
&= \min_{u_k \in \mathbb{R}} \mathop{\mathrm{E}}_{w_k} \left\{ 2y_k^2 + u_k^2 + 2y_k u_k w_k + u_k^2 w_k^2 \right\} \\
&= \min_{u_k \in \mathbb{R}} \left[2y_k^2 + u_k^2 + 2y_k u_k w_{k-1} + u_k^2 (w_{k-1}^2 + 1) \right] \\
&= \min_{u_k \in \mathbb{R}} \left[2y_k^2 + 2y_k w_{k-1} u_k + (2 + w_{k-1}^2) u_k^2 \right]
\end{aligned}$$

Note that we used the formula $\mathrm{Var}\{X\} = \mathrm{E}\{X^2\} - \mathrm{E}\{X\}^2$ when computing the expected value of the expression above. The coefficient of u_k^2 is positive and the minimum is obtained where the derivative vanishes:

$$\frac{d}{du_k} = 2y_k w_{k-1} + 2(2 + w_{k-1}^2) u_k \overset{!}{=} 0 \Rightarrow u_k^* = \frac{-y_k w_{k-1}}{2 + w_{k-1}^2}$$

Hence, the cost-to-go and optimal policy is

$$\underline{\underline{J_k(x_k) = 2y_k^2 - \frac{y_k^2 w_{k-1}^2}{2 + w_{k-1}^2})}} \text{ with } \underline{\underline{\mu_k(x_k) = \frac{-y_k w_{k-1}}{2 + w_{k-1}^2}}}$$

Problem 2 25%

The following questions are about shortest path problems. Please follow the instructions below. If you are given any output of the *Label Correcting Algorithm*, you can assume it has been generated according to these instructions.

Instructions: Recall that in the Label Correcting Algorithm only one instance of a node can be in the `OPEN` *bin at any time. If a node already in the* `OPEN` *bin enters the* `OPEN` *bin again, treat this node as if it entered the* `OPEN` *bin at the current iteration. If two nodes enter the* `OPEN` *bin in the same iteration, add the one with the lowest node number first. The nodes in the* `OPEN` *bin are expressed as a list, written out from left to right, where new nodes are added from the right-hand side.*
Example: `OPEN` *bin: 2, 3, 4; Node exiting* `OPEN` *2; Nodes entering* `OPEN`*: 3, 5; new* `OPEN` *bin: 4, 3, 5.*

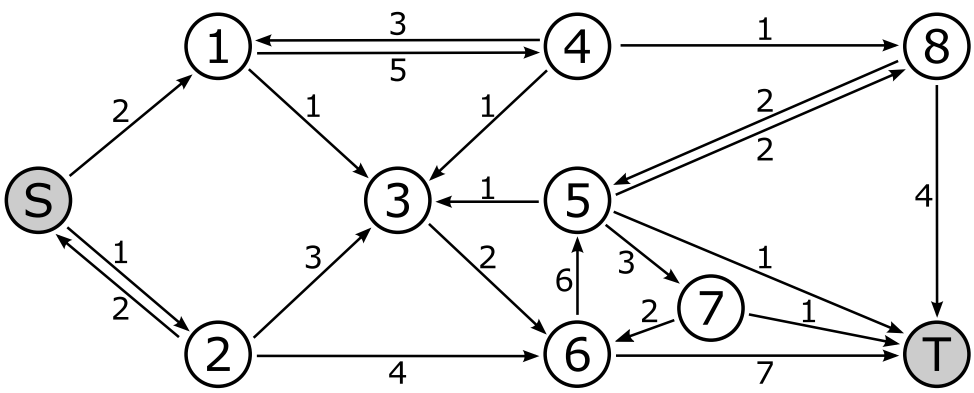

Figure 1

a) Consider the shortest path problem shown in Figure 1. Continue table 1 until termination using best-first search to determine at each iteration which node to remove from the `OPEN` bin. State the resulting shortest path and its cost. **Use the solution sheet with the prepared tables.**

Iteration	Node exiting `OPEN`	`OPEN`	d_S	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_T
0	-	...										
1	S	...										

Table 1

b) Consider the shortest path problem shown in Figure 1 and the lower bound h_j of the cost to move from node j to the terminal node T:

j	1	2	3	4	5	6	7	8
h_j	4	5	9	2	1	6	3	0

Continue table 2 for three iterations (i.e. iteration 4, 5 and 6) by applying the A^*-algorithm and using depth-first search to determine at each iteration which node to remove from the `OPEN` bin. **Use the solution sheet with the prepared tables.**

Iteration	Node exiting `OPEN`	`OPEN`	d_S	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_T
3	6	1, 3, 5	0	2	1	4	∞	11	5	∞	∞	12
4	...											

Table 2

c) Mark the correct answer for each statement.

Grading: Each correct answer is worth 1%. Answers left blank are worth 0%. Wrong answers are penalized with -1%. The minimum score of this subproblem 2.c) is 0%.

i) Consider the shortest path problem shown in Figure 1. $h_8 = 5$ is a lower bound to the cost to move from node 8 to the terminal node T.
☐ true ☐ false

ii) The label correcting algorithm finds the shortest path for any finite graph with both positive and negative arc lengths but no cycles.
☐ true ☐ false

iii) If the check $d_i + a_{ij} < \min\{d_j, d_T\}$ in the label correcting algorithm is replaced with $d_i + a_{ij} \leq \min\{d_j, d_T\}$, the label correcting algorithm still terminates with the shortest path for any finite graph with non-negative arc lengths.
☐ true ☐ false

iv) The label correcting algorithm terminates with $d_T < \infty$ for any finite graph with non-negative arc lengths.
☐ true ☐ false

Solution 2

a)

Iteration	Node exiting `OPEN`	`OPEN`	d_S	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_T
0	-	S	0	∞	∞	∞	∞	∞	∞	∞	∞	∞
1	S	1, 2	0	2	1	∞	∞	∞	∞	∞	∞	∞
2	2	1, 3, 6	0	2	1	4	∞	∞	5	∞	∞	∞
3	1	6, 3, 4	0	2	1	3	7	∞	5	∞	∞	∞
4	3	6, 4	0	2	1	3	7	∞	5	∞	∞	∞
5	6	4, 5	0	2	1	3	7	11	5	∞	∞	12
6	4	5, 8	0	2	1	3	7	11	5	∞	8	12
7	8	5	0	2	1	3	7	10	5	∞	8	12
8	5	-	0	2	1	3	7	10	5	∞	8	11

Shortest path: $S \rightarrow 1 \rightarrow 4 \rightarrow 8 \rightarrow 5 \rightarrow T$
Cost: 11

b)

Iteration	Node exiting `OPEN`	`OPEN`	d_S	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_T
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
3	6	1, 3, 5	0	2	1	4	∞	11	5	∞	∞	12
4	5	1, 3	0	2	1	4	∞	11	5	∞	∞	12
5	3	1	0	2	1	4	∞	11	5	∞	∞	12
6	1	4	0	2	1	4	7	11	5	∞	∞	12
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

c)
- i) False.
- ii) False.
- iii) False.
- iv) False.

Problem 3 25%

Consider the following dynamic system:

$$\begin{aligned} x_{k+1} &= w_k, \\ x_k &\in \{1,2,3\}, \\ u_k &\in \{A,B\}. \end{aligned}$$

The transition probabilities $p_{ij}(u_k) := P(w_k = j | x_k = i, u_k)$ between the states are given by

$$\begin{array}{lll} p_{11}(A) = 0.2, & p_{12}(A) = 0.4, & p_{13}(A) = 0.4, \\ p_{21}(A) = 0.4, & p_{22}(A) = 0.6, & p_{23}(A) = 0, \\ p_{31}(A) = (1-\gamma)0.4, & p_{32}(A) = (1-\gamma)0.6, & p_{33}(A) = \gamma, \end{array}$$

$$\begin{array}{lll} p_{11}(B) = 0.4, & p_{12}(B) = 0.2, & p_{13}(B) = 0.4, \\ p_{21}(B) = 0, & p_{22}(B) = 0.6, & p_{23}(B) = 0.4, \\ p_{31}(B) = (1-\gamma)0.6, & p_{32}(B) = (1-\gamma)0.4, & p_{33}(B) = \gamma, \end{array}$$

with $0 \leq \gamma \leq 1$. The cost function to be minimized is given by

$$\lim_{N\to\infty} \mathrm{E}\left\{ \sum_{k=0}^{N-1} \alpha^k g(x_k, u_k) \right\},$$

with

$$\begin{array}{lll} g(1,A) = 16, & g(2,A) = 10, & g(3,A) = \beta, \\ g(1,B) = 8, & g(2,B) = 8. & g(3,B) = 2\beta, \end{array}$$

where $\alpha \in \mathbb{R}$ and $\beta \in \mathbb{R}$ are some parameters.

Note: All parts can be solved independently.

a) Assume $\alpha = 1$. State the range of β and γ for which the above problem is a well-defined infinite horizon problem with finite cost. Give a short explanation.

b) Assume $\beta = 1$ and $\gamma = 0.5$. State the range of α for which the above problem is a well-defined infinite horizon problem with finite cost. Give a short explanation.

c) Assume $\alpha = 0.5$, $\beta = 1$, and $\gamma = 0$. Perform one iteration of the **Value Iteration Algorithm** for the resulting Discounted Cost Problem, i.e. compute $J_1(1)$, $J_1(2)$, and $J_1(3)$. Use $J_0(1) = 10$, $J_0(2) = 10$, and $J_0(3) = 10$ as initial guess.

d) Assume $\alpha = 1$, $\beta = 0$, and $\gamma = 1$. Perform one iteration of the **Policy Iteration Algorithm** for the resulting Stochastic Shortest Path Problem, i.e. compute $\mu^1(1)$, $\mu^1(2)$, and $\mu^1(3)$. Use $\mu^0(1) = A$, $\mu^0(2) = B$, and $\mu^0(3) = A$ as initial guess.

e) Assume $\alpha = 0.5$, $\beta = 1$, and $\gamma = 0$. We denote the optimal cost vector as $\mathbf{x} = [J(1), J(2), J(3)]^T$. The optimal cost vector can be obtained by solving a **Linear Program** of the generic form

$$\begin{array}{ll} \text{minimize} & \mathbf{f}^T\mathbf{x} \\ \text{subject to} & \mathbf{M}\mathbf{x} \leq \mathbf{h} \end{array}$$

where $\mathbf{f}$ and $\mathbf{h}$ are vectors, and $\mathbf{M}$ is a matrix. Write down a choice for $\mathbf{f}$, $\mathbf{h}$, and $\mathbf{M}$ such that the optimal cost vector $\mathbf{x}$ is obtained by solving the above linear program.

Solution 3

a) For a meaningful, finite cost value there must be a cost-free termination state, i.e. $\beta = 0$. The probability of leaving the termination state must be zero, i.e. $\gamma = 1$.

b) Without a termination state, the stage cost must be discounted exponentially with increasing time step in order to get a finite cost value, i.e. $\alpha \in (-1, 1)$.

c) We initialize the value iteration algorithm and perform one iteration:

- Initial guess:

$$\underline{J_0(1) = 10},\ \underline{J_0(2) = 10},\ \underline{J_0(3) = 10}$$

- Iteration 1:

$$\begin{aligned}
J_1(1) &= \min_{u \in \{A,B\}} \Big[g(1,u) + \alpha p_{11}(u) J_0(1) + \alpha p_{12}(u) J_0(2) + \alpha p_{13}(u) J_0(3) \Big] \\
&= \min \Big[16 + 0.1 \cdot 10 + 0.2 \cdot 10 + 0.2 \cdot 10, 8 + 0.2 \cdot 10 + 0.1 \cdot 10 + 0.2 \cdot 10 \Big] \\
&= \min \Big[21, 13 \Big] \quad \Rightarrow \quad \underline{\underline{J_1(1) = 13}} \\
J_1(2) &= \min_{u \in \{A,B\}} \Big[g(2,u) + \alpha p_{21}(u) J_0(1) + \alpha p_{22}(u) J_0(2) + \alpha p_{23}(u) J_0(3) \Big] \\
&= \min \Big[10 + 0.2 \cdot 10 + 0.3 \cdot 10 + 0 \cdot 10, 8 + 0 \cdot 10 + 0.3 \cdot 10 + 0.2 \cdot 10 \Big] \\
&= \min \Big[15, 13 \Big] \quad \Rightarrow \quad \underline{\underline{J_1(2) = 13}} \\
J_1(3) &= \min_{u \in \{A,B\}} \Big[g(3,u) + \alpha p_{31}(u) J_0(1) + \alpha p_{32}(u) J_0(2) + \alpha p_{33}(u) J_0(3) \Big] \\
&= \min \Big[1 + 0.2 \cdot 10 + 0.3 \cdot 10 + 0 \cdot 10, 2 + 0.3 \cdot 10 + 0.2 \cdot 10 + 0 \cdot 10 \Big] \\
&= \min \Big[6, 7 \Big] \quad \Rightarrow \quad \underline{\underline{J_1(3) = 6}}
\end{aligned}$$

d) State 3 is the termination state and has zero cost, independent of the policy. The termination state does not have to be considered during policy evaluation and improvement. For the states 1 and 2, we initialize the policy iteration algorithm and perform one iteration:

- Initial guess:

$$\underline{\mu^0(1) = A},\ \underline{\mu^0(2) = B}$$

- Iteration 1:

- Policy evaluation:

$$\begin{aligned}
J_{\mu^0}(1) &= g(1, A) + p_{11}(A)J_{\mu^0}(1) + p_{12}(A)J_{\mu^0}(2) \\
&= 16 + 0.2 \cdot J_{\mu^0}(1) + 0.4 \cdot J_{\mu^0}(2) \\
&\Rightarrow J_{\mu^0}(1) = 20 + \frac{1}{2} J_{\mu^0}(2) \\
J_{\mu^0}(2) &= g(2, B) + p_{21}(B)J_{\mu^0}(1) + p_{22}(B)J_{\mu^0}(2) \\
&= 8 + 0 \cdot J_{\mu^0}(1) + 0.6 \cdot J_{\mu^0}(2) \\
&\Rightarrow \underline{J_{\mu^0}(2) = 20} \\
&\Rightarrow \underline{J_{\mu^0}(1) = 30}
\end{aligned}$$

- Policy improvement:

$$\begin{aligned}
\mu^1(1) &= \arg \min_{u \in \{A,B\}} \Big[g(1, u) + p_{11}(u)J_{\mu^0}(1) + p_{12}(u)J_{\mu^0}(2) \Big] \\
&= \arg \min \Big[16 + 0.2 \cdot 30 + 0.4 \cdot 20, 8 + 0.4 \cdot 30 + 0.2 \cdot 20 \Big] \\
&= \arg \min \Big[30, 24 \Big] \quad \Rightarrow \quad \underline{\mu^1(1) = B} \\
\mu^1(2) &= \arg \min_{u \in \{A,B\}} \Big[g(2, u) + p_{21}(u)J_{\mu^0}(1) + p_{22}(u)J_{\mu^0}(2) \Big] \\
&= \arg \min \Big[10 + 0.4 \cdot 30 + 0.6 \cdot 20, 8 + 0 \cdot 30 + 0.6 \cdot 20 \Big] \\
&= \arg \min \Big[34, 20 \Big] \quad \Rightarrow \quad \underline{\mu^1(2) = B}
\end{aligned}$$

Hence, the new policy is given by

$$\underline{\underline{\mu^1(1) = B}}, \ \underline{\underline{\mu^1(2) = B}}, \ \underline{\underline{\mu^1(3) = A}} \text{ or } \underline{\underline{\mu^1(3) = B}}.$$

e) For this discounted cost problem the optimal costs are obtained by

$$\begin{aligned}
&\text{maximize} \quad \sum_{i=1}^{3} J(i) \\
&\text{subject to} \quad J(i) \leq g(i, u) + \alpha \sum_{j=1}^{3} p_{ij}(u)J(j), \quad \forall i, \forall u
\end{aligned}$$

Hence we get:

$$\mathbf{f} = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}, \ \mathbf{M} = \begin{bmatrix} \mathbf{M}_A \\ \mathbf{M}_B \end{bmatrix}, \ \mathbf{h} = \begin{bmatrix} \mathbf{h}_A \\ \mathbf{h}_B \end{bmatrix},$$

with

$$\mathbf{M}_A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \alpha \begin{bmatrix} p_{11}(A) & p_{12}(A) & p_{13}(A) \\ p_{21}(A) & p_{22}(A) & p_{23}(A) \\ p_{31}(A) & p_{32}(A) & p_{33}(A) \end{bmatrix} = \begin{bmatrix} 0.9 & -0.2 & -0.2 \\ -0.2 & 0.7 & 0 \\ -0.2 & -0.3 & 1 \end{bmatrix},$$

$$\mathbf{M}_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \alpha \begin{bmatrix} p_{11}(B) & p_{12}(B) & p_{13}(B) \\ p_{21}(B) & p_{22}(B) & p_{23}(B) \\ p_{31}(B) & p_{32}(B) & p_{33}(B) \end{bmatrix} = \begin{bmatrix} 0.8 & -0.1 & -0.2 \\ 0 & 0.7 & -0.2 \\ -0.3 & -0.2 & 1 \end{bmatrix},$$

$$\mathbf{h}_A = \begin{bmatrix} g(1, A) \\ g(2, A) \\ g(3, A) \end{bmatrix} = \begin{bmatrix} 16 \\ 10 \\ 1 \end{bmatrix},$$

$$\mathbf{h}_B = \begin{bmatrix} g(1, B) \\ g(2, B) \\ g(3, B) \end{bmatrix} = \begin{bmatrix} 8 \\ 8 \\ 2 \end{bmatrix}.$$

Problem 4 25%

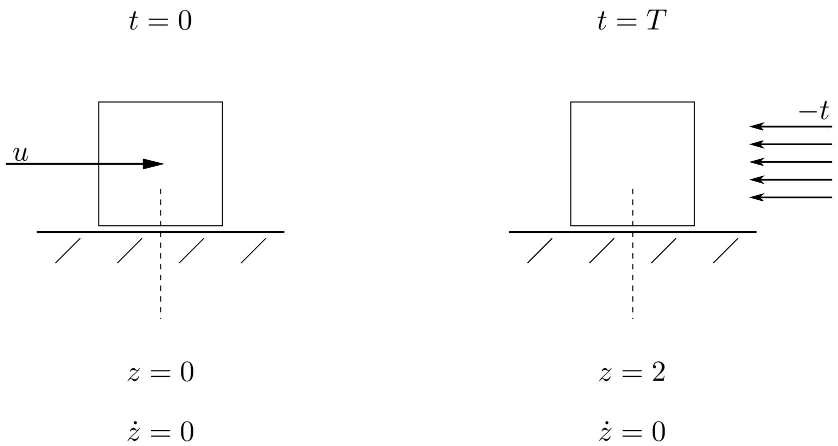

Figure 2

At time $t = 0$, a unit mass is at rest at location $z = 0$. The mass is on a frictionless surface and it is desired to apply a force $u(t)$, $0 \leq t \leq T$, such that the mass is at rest at location $z = 2$ at a free terminal time T. An additional force is acting on the mass, such that the dynamics are given by

$$\ddot{z}(t) = u(t) - t, \quad 0 \leq t \leq T.$$

Of all the functions $u(t)$ that achieve the above objective, find the one that minimizes

$$\int_0^T \frac{1}{2} u(t)^2 dt.$$

a) Augment the system dynamics using the fact that $\dot{t} = 1$ to obtain a time-invariant system. In particular, define the **state variable** $x(t)$ and write down the **system dynamics** $f(x(t), u(t))$ and all **boundary conditions** of the augmented system.

b) Compute the Hamiltonian function for the augmented problem.

c) Write down the Minimum Principle's necessary conditions for optimality for the augmented problem.

d) Compute the terminal time T and optimal control input $u^*(t)$.

Solution 4

a) Because the system dynamics are time-varying, we introduce an additional state for the time such that the Hamiltonian will be constant along the optimal trajectory. We define the state vector to be

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = \begin{bmatrix} z(t) \\ \dot{z}(t) \\ t \end{bmatrix}.$$

The dynamics can then be written as

$$f\left(x(t), u(t)\right) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ u(t) - x_3(t) \\ 1 \end{bmatrix},$$

with the boundary conditions

$$\begin{aligned} x_1(0) &= 0, \quad x_1(T) = 2, \\ x_2(0) &= 0, \quad x_2(T) = 0, \\ x_3(0) &= 0. \end{aligned}$$

Note that there is no constraint on the terminal state $x_3(T)$ as this is part of the optimization.

b) The Hamiltonian function is given by

$$\begin{aligned} H\left(x(t), u(t), p(t)\right) &= g\left(x(t), u(t)\right) + p(t)^T f\left(x(t), u(t)\right), \\ &= \frac{1}{2}u(t)^2 + p_1(t)x_2(t) + p_2(t)\left(u(t) - x_3(t)\right) + p_3(t). \end{aligned}$$

c) The Minimum Principle's necessary conditions for optimality can be written as

State equation: $\dot{x}(t) = \dfrac{\partial H(x(t), u(t), p(t))}{\partial p}$.

$$\begin{aligned}
\Rightarrow \dot{x}_1(t) &= x_2(t), & x_1(0) &= 0, & x_1(T) &= 2,\\
\dot{x}_2(t) &= u(t) - x_3(t), & x_2(0) &= 0, & x_2(T) &= 0,\\
\dot{x}_3(t) &= 1, & x_3(0) &= 0, & &\rightarrow x_3(t) = t.
\end{aligned}$$

Adjoint equation: $\dot{p}(t) = -\dfrac{\partial H(x(t), u(t), p(t))}{\partial x}$.

$$\begin{aligned}
\Rightarrow \dot{p}_1(t) &= 0\\
\dot{p}_2(t) &= -p_1(t),\\
\dot{p}_3(t) &= p_2(t), & p_3(T) = \frac{\partial h(x(T))}{\partial x_3} = 0.
\end{aligned}$$

Control input: $u^*(t) = \arg\min_{u \in U} H(x^*(t), u, p(t))$.

$$\begin{aligned}
\Rightarrow \frac{\partial H(x^*(t), u^*(t), p(t))}{\partial u^*} &= u^*(t) + p_2(t) \overset{!}{=} 0,\\
u^*(t) &= -p_2(t).
\end{aligned}$$

We verify that $u^*(t)$ is a minimum by evaluating the second derivative:
$$\frac{\partial^2 H(x^*(t), u^*(t), p(t))}{\partial u^{*2}} = 1.$$

Hamiltonian: $H(x^*(t), u^*(t), p(t)) = 0$ (free terminal time), $\quad \forall t \in [0, T]$.

d) First, the adjoint equations are integrated resulting in the following equations for the co-states:

$$\begin{aligned}
p_1(t) &= c_1, && c_1 \text{ constant},\\
p_2(t) &= -c_1 t + c_2, && c_2 \text{ constant},\\
p_3(t) &= -\frac{1}{2}c_1 t^2 + c_2 t + c_3, && c_3 \text{ constant}.
\end{aligned}$$

The optimal input $u^*(t)$ is thus

$$u^*(t) = c_1 t - c_2$$

and integrating the system equation yields

$$\begin{aligned}
\dot{x}_2(t) &= c_1 t - c_2 - t && \Rightarrow x_2(t) = \frac{1}{2}(c_1 - 1)t^2 - c_2 t + c_4, && c_4 \text{ constant},\\
\dot{x}_1(t) &= x_2(t) && \Rightarrow x_1(t) = \frac{1}{6}(c_1 - 1)t^3 - \frac{1}{2}c_2 t^2 + c_3 t + c_5, && c_5 \text{ constant}.
\end{aligned}$$

Using $x_1(0) = x_2(0) = 0$, it follows that c_4 and c_5 are both 0. Furthermore, at time T, the terminal conditions $x_1(T) = 2$ and $x_2(T) = 0$ need to be satisfied, i.e.

$$x_1(T) = \left(\frac{1}{6}(c_1 - 1)T - \frac{1}{2}c_2\right)T^2 = 2, \tag{1}$$

$$x_2(T) = \left(\frac{1}{2}(c_1 - 1)T - c_2\right)T = 0. \tag{2}$$

Solving (2) for c_2 and inserting this in (1), we obtain the following expressions for c_1 and c_2:

$$\begin{aligned}
c_1 &= -\frac{24}{T^3} + 1,\\
c_2 &= -\frac{12}{T^2}.
\end{aligned}$$

The final time T can be computed by using the fact that the Hamiltonian is zero over the entire time interval. For simplicity, we evaluate the Hamiltonian at time $t = T$:

$$H = \frac{1}{2}u(T)^2 + p_1(T)x_2(T) + p_2(T)\left(u(T) - x_3(T)\right) + p_3(T) \stackrel{!}{=} 0.$$

Because $x_2(T) = p_3(T) = 0$, the above equation simplifies to

$$\begin{aligned}
\frac{1}{2}u(T)^2 - u(T)\left(u(T) - T\right) &= 0,\\
-\frac{1}{2}u(T)^2 + u(T)T &= 0,\\
u(T)\left(T - \frac{1}{2}u(T)\right) &= 0,
\end{aligned}$$

which yields the following two solutions for T:

$$\begin{aligned}
u(T) = 0 \quad &\Rightarrow \quad T = \sqrt[3]{12},\\
T - \frac{1}{2}u(T) = 0 \quad &\Rightarrow \quad T = \sqrt[3]{-12}.
\end{aligned}$$

The final time is thus $T = \sqrt[3]{12}$ and the optimal control input is

$$u^*(t) = -t + \sqrt[3]{12}.$$