Stochastic Optimisation - Problem Sheet 5

Dom Hutchinson

December 4, 2020

Answer 3) (a)
Let X_t be the system state at the start of time-period t and Y_t be the action the agent takes at the start of time-period t.

Let W_t represent the number of people wishing to join the queue in time-period t and Z_t represent the number of customers to leave the queue in time-period t. They have the following distribution

$$\begin{aligned}\mathbb{P}(W_t = w) &= \begin{cases} p & w = 1 \\ 1-p & w = 0 \end{cases} \\ \mathbb{P}(Z_t = z) &= \begin{cases} q & z = 1 \\ 1-q & z = 0 \end{cases}\end{aligned}$$

- *Decision Epochs* - At the start of each period.
- *Time-Horizon* - $T = \{0, \dots, N-1\}$.
- *Action-Space.*

  We allow the agent to decide whether to allow customers to join the queue of not. This can be encoded into Y_t as

  $$Y_t = \begin{cases} 1 & \text{if accepting customers} \\ 0 & \text{if } \underline{\text{not}} \text{ accepting customers} \end{cases}$$

  As at most 1 customer may wish to join the queue in a given time-period, Y_t is equivalent to the maximum amount of customer being allowed to join the queue under each case.

  The *Action-Space* is $A = \{0, 1\}$.
- *State-Space*

  Let X_t take the value of the number of customers in the queue at the start of time-period t. As n is the capacity of the queue $X_t \in [0, n]$ and can take any value in this interval. This means the state-space is $S = \{0, 1, \dots, n\}$.

  This gives us the state-equation $X_{t+1} = X_t + Y_t W_t - Z_t$.
- *Admissible Action-Space* - $A(s) = \begin{cases} \{0,1\} & \text{if } s \in [0, \dots, n-1] \\ \{0\} & \text{if } s = n \end{cases}.$
- *Transition Probabilities* - The definition of transition probabilities state

  $$p_t(s'|s, a) := \mathbb{P}^\pi(X_{t+1} = s' | X_t = s, Y_t = a)$$

Specifically, from the state equation for X_{t+1} we want to derive

$$\begin{array}{rcl} p_t(s'|s,a) & = & \mathbb{P}^\pi(X_{t+1}=s'|X_t=s,Y_t=a) \\ & = & \mathbb{P}^\pi(X_t+Y_tW_t-Z_t=s'|X_t=s,Y_t=a) \\ & = & \mathbb{P}(s+aW_t-Z_t=s') \end{array}$$

Consider the following cases involving s' wrt s

i). $s=0$.
If the queue is empty then no-one can leave so $\mathbb{P}(Z_t=0)=1$.

$$\begin{array}{rcl} p_t(s'|0,a) & = & \mathbb{P}(aW_t=s') \\ & = & \begin{cases} 1 & \text{if } a=0, s'=0 \\ \mathbb{P}(W_t=0) & \text{if } a=1, s'=0 \\ \mathbb{P}(W_t=1) & \text{if } a=1, s'=1 \\ 0 & \text{otherwise} \end{cases} \\ & = & \begin{cases} 1 & \text{if } a=0, s'=0 \\ 1-p & \text{if } a=1, s'=0 \\ p & \text{if } a=1, s'=1 \\ 0 & \text{otherwise} \end{cases} \end{array}$$

ii). $s=n$.
If the queue is full then no-one can join the queue, so $\mathbb{P}(W_t=0)=1$.

$$\begin{array}{rcl} p_t(s'|n,a) & = & \mathbb{P}(n-Z_t=s') \\ & = & \begin{cases} \mathbb{P}(Z_t=0) & \text{if } s'=n \\ \mathbb{P}(Z_t=1) & \text{if } s'=n-1 \\ 0 & \text{if } s'<n-1 \end{cases} \\ & = & \begin{cases} 1-q & \text{if } s'=n \\ q & \text{if } s'=n-1 \\ 0 & \text{if } s'<n-1 \end{cases} \end{array}$$

Note that these results are independent of the action taken a.

iii). $s'=s,\ s\not\in\{0,n\}$.
The queue has not changed size, is non-empty and non-full. Thus, either no movements have occurred or, one customer joined the queue and another customer left the queue.

$$\begin{array}{rcl} p_t(s|s,a) & = & \mathbb{P}(s+aW_t-Z_t=s) \\ & = & \mathbb{P}(aW_t-Z_t=0) \\ & = & \begin{cases} \mathbb{P}(Z_t=0) & \text{if } a=0 \\ \mathbb{P}(W_t=Z_t) & \text{if } a=1 \end{cases} \\ & = & \begin{cases} 1-q & \text{if } a=0 \\ pq+(1-p)(1-q) & \text{if } a=1 \end{cases} \end{array}$$

iv). $s'=s+1,\ s\not\in\{0,n\}$.
In this case we must have allowed people to join the queue, thus $a=1$

$$\begin{array}{rcl} p_t(s+1|s,a) & = & \mathbb{P}(s+aW_t-Z_t=s+1) \\ & = & \mathbb{P}(W_t-Z_t=1) \\ & = & \mathbb{P}(W_t=1)\mathbb{P}(Z_t=0) \\ & = & p(1-q) \end{array}$$

v). $s' = s-1,\ s \notin \{0,n\}$.

$$\begin{array}{rcl} p_t(s-1|s,a) & = & \mathbb{P}(s+aW_t-Z_t=s-1) \\ & = & \mathbb{P}(aW_t-Z_t=-1) \\ & = & \begin{cases} \mathbb{P}(Z_t=1) & \text{if } a=0 \\ \mathbb{P}(W_t=0,Z_t=1) & \text{if } a=1 \end{cases} \\ & = & \begin{cases} q & \text{if } a=0 \\ (1-p)q & \text{if } a=1 \end{cases} \end{array}$$

vi). *All other cases.*

All other cases require either the number of people in the queue to become negative or to change by more than 1 in a single time-step, both of these are impossible so have 0 probability.

$$p_t(s'|0,a)=0$$

- *Immediate Costs*

For costs in time-period t we always have to pay cX_t for the length of the queue. Additionally, if a customer is rejected C is payed as well. Mathematically, a customer is rejected if $W_t=1$ and $Y_t=0$. We can summarise the total cost G_t incurred in time-period t as

$$G_t := g_t(W_t,X_t,Y_t) = cX_t + \mathbb{1}\{W_t=1,Y_t=0\}\cdot C$$

This assumes that $Y_t=0$ if the queue is already full.

- *Equivalent Objective*

The objective of this problem is to minimise expected total cost

$$\mathbb{E}^\pi\left[\sum_{t=0}^{N-1} G_t\right] = \mathbb{E}^\pi\left[\sum_{t=0}^{N-1} g_t(W_t,X_t,Y_t)\right]$$

This expectations depends upon the number of customers wishing to join the queue $Y_0,\dots,Y_{N-1}$ which is independent of a policy π so does not fit within the framework a *Markov Decision Problem.* Thus I shall transform the expect total cost.

$$\begin{array}{rcl} \mathbb{E}^\pi\left[\sum_{t=0}^{N-1} G_t\right] & = & \sum_{t=0}^{N-1}\mathbb{E}^\pi[G_t] \\ & = & \sum_{t=0}^{N-1}\mathbb{E}^\pi\left[\mathbb{E}^\pi[G_t|X_t,Y_t]\right] \text{ by Tower Property} \\ & = & \sum_{t=0}^{N-1}\mathbb{E}^\pi\left[\mathbb{E}^\pi[g_t(W_t,X_t,Y_t)|X_t,Y_t]\right] \end{array}$$

Define reward functions $r_t(s,a)$ and $r_N(s)$

$$\begin{array}{rcl} r_N(s) & = & 0 \\ r_t(s,a) & = & -\mathbb{E}^\pi[g_t(W_t,X_t,Y_t)|X_t=s,Y_t=a] \\ & = & -\mathbb{E}^\pi[g_t(W_t,s,a)|X_t=s,Y_t=a] \\ & = & -\mathbb{E}^\pi[g_t(W_t,s,a)] \\ & = & -p\cdot g_t(1,s,a)-(1-p)\cdot g_t(0,s,a) \\ & = & -p(cs+\mathbb{1}\{a=0\}C)-(1-p)cs \\ & = & -cs-\mathbb{1}\{a=0\}\cdot pC \end{array}$$

Using these reward functions the total expected reward can be rephrased

$$-\mathbb{E}^{\pi}\left[r_N(X_n)+\sum_{t=0}^{N-1}r_t(X_t,Y_t)\right] \tag{1}$$

The equivalent objective is to find a policy $\pi\in HR(T)$ which maximises (1).

Answer 3) (b)
From the markov decision problem formulated in `3) (a)` and the given conditions we can state the following properties of the system

$$\begin{array}{rcl} T &=& \{0,1\}\\ S &=& \{0,1,2,3\}\\ A &=& \{0,1\}\\ A(s) &=& \begin{cases}\{0,1\} & \text{if } s\in\{0,1,2\}\\ \{0,\} & \text{if } s=3\end{cases}\end{array}$$

The transition probabilities $p_t(s'|s,a)$ are defined in the tables below, separated by what action a is taken.

$p_t(s'|s,0) =$

s\s'	0	1	2	3
0	1	0	0	0
1	q	1-q	0	0
2	0	q	1-q	0
3	0	0	q	1-q

$=$

s\s'	0	1	2	3
0	1	0	0	0
1	1/4	3/4	0	0
2	0	1/4	3/4	0
3	0	0	1/4	3/4

$p_t(s'|s,1) =$

s\s'	0	1	2	3
0	1-p	p	0	0
1	q(1-p)	pq+(1-p)(1-q)	p(1-q)	0
2	0	q(1-p)	pq+(1-p)(1-q)	p(1-q)
3	0	0	q	1-q

$=$

s\s'	0	1	2	3
0	1/2	1/2	0	0
1	1/4	1/8	3/8	0
2	0	1/4	1/8	3/8
3	0	0	1/4	3/4

The terminal cost value is $r_2(s)=0$. The cost function values $r_t(s,a)$ are given in the table below

$r_t(s,a) =$

s\a	0	1
0	$-pC$	0
1	$-c-pC$	$-c$
2	$-2c-pC$	$-2c$
3	$-3c-pC$	$-3c$

$=$

s\a	0	1
0	-1	0
1	-2	-1
2	-3	-2
3	-4	-3

To find the optimal policy π^* we use the *dynamic programming algorithm* which is defined as

$$\begin{array}{rcl} w_t^*(s,a) &:=& r_t(s,a)+\sum_{s'\in S}u_{t+1}^*(s')p_t(s'|s,a)\\ u_t^*(s) &=& \max_{a\in A(s)}(w_t^*)\\ d_t^*(s) &=& \text{argmax}_{a\in A(s)}(w_t^*)\end{array}$$

where $u_2^*(s):=r_2(s)=0\ \forall\ s\in S$. Specifically, we need to determine $u_t^*(s),\ d_t^*(s)$ for all states s in each time-period $t\in\{1,0\}$.

- *Time-Period* $t=1$.

  In this time-period

$$\begin{aligned} w_1^*(s,a) &= r_1(s,a) + \sum_{s'\in\{0,1,2,3\}} u_2^*(s)p_t(s'|s,a) \\ &= r_1(s,a) \text{ since } u_2^*(s)=0\ \forall\ s\in S \end{aligned}$$

  This gives the following table of values for $w_1^*(s,a)$

$w_1^*(s,a) =$

s\a	0	1
0	-1	0
1	-2	-1
2	-3	-2
3	-4	-3

  From this table, taking action $a=1$ produces the greatest expected value in all states. This is summarised in the following table for $u_1^*(s), d_1^*(s)$

s	$u_1^*(s)$	$d_1^*(s)$
0	0	1
1	-1	1
2	-2	1
3	-3	1

- *Time-Period* $t=0$.

  In this time-period

$$\begin{aligned} w_0^*(s,a) &= r_0(s,a) + \sum_{s'\in\{0,1,2,3\}} u_1^*(s)p_t(s'|s,a) \\ &= r_0(s,a) - p_t(1|s,a) - 2\cdot p_t(2|s,a) - 3\cdot p_t(3|s,a) \end{aligned}$$

  This gives the following table of values for $w_0^*(s,a)$

$w_1^*(s,a) =$

s\a	0	1
0	-1+0+0+0	0-1/2+0+0
1	-2-3/4+0+0	-1-1/8-6/8+0+0
2	-3-1/4-6/4+0	-2-1/4-2/8-9/8
3	-4+0-2/4-9/4	-3+0-2/4-9/4

$=$

s\a	0	1
0	-1	-1/2
1	-11/4	-15/8
2	-19/4	-29/8
3	-27/4	-23/4

  Again, from this table, taking action $a=1$ produces the greatest expected value in all states. This is summarised in the following table for $u_1^*(s), d_1^*(s)$

s	$u_0^*(s)$	$d_0^*(s)$
0	-1/2	1
1	-15/8	1
2	-29/8	1
3	-23/4	1

The optimal policy is

$$\pi^* = (d_0^*(s), d_1^*(s)) = (1,1)\ \forall\ s$$

The optimal value function is

$$u_0^*(s) = \begin{cases} -1/2 & \text{if } s = 0 \\ -15/8 & \text{if } s = 1 \\ -29/8 & \text{if } s = 2 \\ -23/4 & \text{if } s = 3 \end{cases}$$

Answer 4) (a)
Let X_t be the system state at the start of time-period X_t and Y_t be the action the agent takes at the start of time-period Y_t.

- *Decision Epochs* - Start of each time-period.
- *Time-Horizon* - $T = \{0, \dots, N-1\}$.
- *User Actions.*

  At the start of each turn the agent can either repair the machine to a specified state $s \in \{1, \dots, M-1\}$ or not repair. This can be encoded into Y_t as
  $$Y_t = \begin{cases} 0 & \text{if machine is \underline{not} repaired} \\ s & \text{if machine is repaired to state } s \end{cases}$$
- *Action-Space.*

  Given the specification of Y_t the action-space is $A := \{0, \dots, M-1\}$.
- *State-Space.*

  When the user makes their decision the current state of the machine at the beginning of the time-period is only required piece of knowledge. Thus the state-space is the set of states the machine can take $S = \{1, \dots, M\}$.

  This means that X_t denotes the state of the machine at the start of period t.
- *Admissible Action-Space.*

  The agent can always choose not to repair the machine, but if they choose to repair the machine they can only repair it to a better state. This gives admissible action-spaces as
  $$A(s) = \{0, \dots, s-1\},\ s \in S$$
- *Immediate Costs.*

  In each time-period there is always a cost for running the machine $c_o(s)$, where s is the state after the agent has taken their action. There is an additional cost $c_r(s, s')$ if the agent chooses to repair the machine from state s to state s' (ie if $Y_t = s'$).

  This is summarised in the following cost function
  $$G_t := g_t(X_t, Y_t) = \begin{cases} c_o(X_t) & \text{if } Y_t = 0 \\ c_o(Y_t) + c_r(X_t, Y_t) & \text{if } Y_t \in \{1, \dots, M-1\} \end{cases}$$
- *Transition Probabilities.*

  The definition of transition probabilities state
  $$\begin{aligned} p_t(s'|s, a) &:= \mathbb{P}^\pi(X_{t+1} = s' | X_t = s, Y_t = a) \\ &= \begin{cases} p(s'|s) & \text{if } a = 0 \\ p(s'|a) & \text{if } a \in A(s) \setminus \{0\} \end{cases} \end{aligned}$$
- *Equivalent Objective* The objective of this problem is to minimises the total expected cost
  $$\begin{aligned} \mathbb{E}^\pi\left[\sum_{t=0}^{N-1} G_t\right] &= \mathbb{E}^\pi\left[\sum_{t=0}^{N-1} g_t(X_t, Y_t)\right] \\ &= \sum_{t=0}^{N-1} \mathbb{E}^\pi[g_t(X_t, Y_t)] \\ &= \sum_{t=0}^{N-1} \mathbb{E}^\pi\big[\mathbb{E}^\pi[g_t(X_t, Y_t)]\big| X_t = s, Y_t = a\big] \text{ by Tower Property.} \end{aligned}$$

Minimisation does not fit within the Markov decision problem framework, so I shall make this a maximisation problem. Consider the following reward functions

$$\begin{array}{rcl} r_N(s) & := & 0 \\ r_t(s,a) & := & -\mathbb{E}^\pi[g_t(X_t,Y_t)|X_t=s,Y_t=a] \\ & = & -\mathbb{E}^\pi[g_t(s,a)] \\ & = & \begin{cases} -c_o(s) & \text{if } a=0 \\ -c_o(s)-c_r(s,a) & \text{if } a\in\{1,\dots,s-1\} \end{cases} \end{array}$$

Using these definitions, our objective can be restated as wishing to maximise the following

$$-\mathbb{E}^\pi\left[r_N(s)+\sum_{t=0}^{N-1}r_t(s,a)\right]$$

Answer 4) (b)
The question defines the following sets of values

$$p(s'|s) = \begin{array}{c|ccc} s\backslash s' & 1 & 2 & 3 \\ \hline 1 & 1/2 & 1/4 & 1/4 \\ 2 & 0 & 1/4 & 3/4 \\ 3 & 0 & 0 & 1 \end{array}$$

$$c_r(s,s') = \begin{array}{c|ccc} s'\backslash s & 1 & 2 & 3 \\ \hline 1 & \text{ND} & 1 & 4 \\ 2 & \text{ND} & \text{ND} & 2 \\ 3 & \text{ND} & \text{ND} & \text{ND} \end{array}$$

From the markov decision problem formulated in `4) (a)` and the given conditions we can state the following properties of the system

$$\begin{array}{rcl} T & = & \{0,1\} \\ S & = & \{1,2,3\} \\ A & = & \{0,1,2\} \\ A(s) & = & \{0,\dots,s-1\} \end{array}$$

The transition probabilities $p_t(s'|s,a)$ are defined in the tables below, separated by what action a is taken.

$$p_t(s'|s,0) = \begin{array}{c|ccc} s\backslash s' & 1 & 2 & 3 \\ \hline 1 & 1/2 & 1/4 & 1/4 \\ 2 & 0 & 1/4 & 3/4 \\ 3 & 0 & 0 & 1 \end{array}$$

$$p_t(s'|s,1) = \begin{array}{c|ccc} s\backslash s' & 1 & 2 & 3 \\ \hline 1 & \text{NA} & \text{NA} & \text{NA} \\ 2 & 1/2 & 1/4 & 1/4 \\ 3 & 1/2 & 1/4 & 1/4 \end{array}$$

$$p_t(s'|s,2) = \begin{array}{c|ccc} s\backslash s' & 1 & 2 & 3 \\ \hline 1 & \text{NA} & \text{NA} & \text{NA} \\ 2 & \text{NA} & \text{NA} & \text{NA} \\ 3 & 0 & 1/4 & 3/4 \end{array}$$

"NA" denotes that a is not an admissible action for that s (ie $a \notin A(s)$).

The terminal cost value is $r_2(s) = 0$. The cost function values $r_t(s,a)$ are given in the table below

$$r_t(s,a) = \begin{array}{c|ccc} s\backslash a & 0 & 1 & 2 \\ \hline 1 & c_o(1) & \text{NA} & \text{NA} \\ 2 & c_o(2) & c_o(2)+c_r(2,1) & \text{NA} \\ 3 & c_o(3) & c_o(3)+c_r(3,1) & c_o(3)+c_r(3,2) \end{array} = \begin{array}{c|ccc} s\backslash a & 0 & 1 & 2 \\ \hline 1 & -1 & \text{NA} & \text{NA} \\ 2 & -2 & -3 & \text{NA} \\ 3 & -5 & -9 & -7 \end{array}$$

To find the optimal policy π^* we use the *dynamic programming algorithm* which is defined as

$$\begin{array}{rcl} w_t^*(s,a) & := & r_t(s,a) + \sum_{s' \in S} u_{t+1}^*(s') p_t(s'|s,a) \\ u_t^*(s) & = & \max_{a \in A(s)} (w_t^*) \\ d_t^*(s) & = & \text{argmax}_{a \in A(s)} (w_t^*) \end{array}$$

where $u_2^*(s) := r_2(s) = 0 \ \forall \ s \in S$. Specifically, we need to determine $u_t^*(s), \ d_t^*(s)$ for all states s in each time-period $t \in \{1, 0\}$.

- *Time-Period* $t = 1$.

  In this time-period

$$\begin{array}{rcl} w_1^*(s,a) & = & r_1(s,a) + \sum_{s' \in \{1,2,3\}} u_2^* p_t(s'|s,a) \\ & = & r_1(s,a) \text{ since } u_2^*(s) = 0 \ \forall \ s \in S \end{array}$$

  This gives the following table of values for $w_1^*(s,a)$

$$w_1^*(s,a) = \begin{array}{c|ccc} s\backslash a & 0 & 1 & 2 \\ \hline 1 & -1 & \text{NA} & \text{NA} \\ 2 & -2 & -3 & \text{NA} \\ 3 & -5 & -9 & -7 \end{array}$$

  In all cases taking action $a = 0$ (ie not repairing) yields the best results. This is summarised in the following table of results for $u_1^*(s), d_1^*(s)$.

s	$u_1^*(s)$	$d_1^*(s)$
1	-1	0
2	-2	0
3	-5	0

- *Time-Period* $t = 0$.

  In this time-period

$$\begin{array}{rcl} w_0^*(s,a) & = & r_0(s,a) + \sum_{s' \in \{1,2,3\}} u_1^* p_t(s'|s,a) \\ & = & r_0(s,a) - p_t(1|s,a) - 2p_t(2|s,a) - 5p_t(3|s,a) \end{array}$$

  This gives the following table of values for $w_0^*(s,a)$

$$w_0^*(s,a) = \begin{array}{c|ccc} s\backslash a & 0 & 1 & 2 \\ \hline 1 & -1-1/2-2/4-5/4 & \text{NA} & \text{NA} \\ 2 & -2+0-2/4-15/4 & -3-1/2-2/4-5/4 & \text{NA} \\ 3 & -5+0+0-5 & -9-1/2-2/4-5/4 & -7+0-2/4-15/4 \end{array} = \begin{array}{c|ccc} s\backslash a & 0 & 1 & 2 \\ \hline 1 & -13/4 & \text{NA} & \text{NA} \\ 2 & -25/4 & -21/4 & \text{NA} \\ 3 & -10 & -45/4 & -45/4 \end{array}$$

This time the optimal action is different for different states. I summarise the optimal actions in the table below

s	$u_1^*(s)$	$d_1^*(s)$
1	-13/4	0
2	-21/4	1
3	-10	0

The optimal strategy π^* is

$$\pi^* := (d_1^*(s), d_2^*(s))$$

and the optimal value function is

$$u_0^*(s) = \begin{cases} -13/4 & \text{if } s = 0 \\ -21/4 & \text{if } s = 1 \\ -10 & \text{if } s = 2 \end{cases}$$