MDP Course

Witek ten Hove, Ritsaart Bergsma, Jeroen Landman

October 2024

1 Assignment - Airline overbooking

An airline seeks a reservation policy for a flight with S seats that maximizes its expected profit from the flight. Reservation requests arrive hourly according to a Bernoulli process with p being the probability of a reservation request per hour (at most one reservation request will arrive per hour). A passenger with a booked reservation pays the fare $f > 0$ at flight time. If $b \geq 0$ passengers with booked reservations are denied boarding at flight time, they do not pay the fare, and the airline pays them a penalty $c(b)$ (divided among them) where $b \mapsto c(b)$ is increasing with $c(0) = 0$.

Consider the n-th hour before flight time T. At the beginning of the hour, the airline reviews the number of booked reservations on hand, r say, and decides whether to book (accept) or decline a reservation request arriving during the next hour. Each of the r booked reservations may cancel during the hour, independently of each other, with probability q.

For this reason, the airline is considering the possibility of overbooking the flight to compensate for cancellations. Let $V_n^*(r)$ be the maximum expected future profit when there are r booked reservations at the beginning of the hour, before the accept/decline decision has been taken, and reservation requests and cancellations during the hour have occurred. Let $W_n^*(r)$ be the maximum expected future profit when there are r booked reservations after booking or declining a reservation request, but before cancellations. The aim is to determine an optimal reservation policy for any value of the number of booked reservations at the beginning of each hour till the flight time T.

a) Markov decision model

Formulate the problem as a Markov decision model, by determining the state space, action spaces, rewards, terminal rewards, and the transition probabilities. Formulate the optimality equation from which an optimal reservation policy can be determined.

b) Optimality of booking-limit policies

Assume, as can be shown, that if g is a quasiconcave function on the integers, then $r \mapsto \mathbb{E}(g(B_r))$ is quasiconcave, where B_r is a sum of independent identically distributed Bernoulli random variables. We recall that g is quasiconcave on the (positive) integers when there exists a number a such that g is increasing on $[0, a]$ and decreasing on $[a, \infty]$.

Use this result to show the following facts. First, show that $r \mapsto W_n^*(r)$ is quasiconcave.

Let $b_n = \arg\max_r W_n^*(r)$. Call b_n the booking limit. Then show that $r \mapsto V_n^*(r)$ is quasiconcave with maximum b_n. Finally, show that it is optimal to accept a reservation if and only if $r < b_n$, with r the number of booked reservations on hand at the beginning of the hour (before a decision has been taken).

c) Solving the problem

Solve the problem when the parameters are as follows:

- $T = 30$
- $c(b) = f \cdot b$
- $S = 10$

- $f = €\ 300$
- $p = 0.2$ and 0.3
- $q = 0.05$ and 0.10
- $r \leq 20$ (so there is an upper bound on the total number of reservations).

Make graphs of the different combinations. In each case, estimate the booking limit ten hours before flight time from your graphs. Discuss whether your graphs confirm the claim in (b) that $r \mapsto V_n^*(r)$ is quasiconcave.

What conjectures do the graphs suggest about the optimal reservation policy and/or maximum expected reward and their variation with the various data elements? You will lose points on your conjectures only if your graphs are inconsistent with or do not support your conjectures, or if you don't make enough interesting conjectures. The idea here is to brainstorm intelligently.

Solutions

1.1 a) Markov decision model

We define the state r as the number of accepted bookings at hand at the beginning of each epoch. This accounts for both cancellations and accepted new booking requests, before a new accept/decline decisions is made. Because there can be mostly one new booking each epoch the number of states is bounded by the number of epochs N.

$$r \in \mathbf{R} = \{0, ..., N\}$$

$$A = \{Accept, Decline\}$$

Rewards only take place at the moment of flight, and consists of two components, the fares payed by customers and the penalty paid for overbooking if there are more bookings than available seats S.

$$R_n(r, a) = \begin{cases} 0 & \text{for all } n < T, \\ f \cdot r & \text{for } n = T, r \leq S, \\ f \cdot S - c(r - S) & \text{for } n = T, r > S \end{cases} \tag{1}$$

The transition probabilities are dependent on the probability of a customer arriving, p, the probability of some of the r customers canceling, and the action. Since there can at most be one customer arriving during the hour, the probability of transitioning from a state in which there are r booked flights, to more than $r+1$ booked flights is zero. The transition from state r to $r-k$ can happen both by a customer arriving with probability p and $k+1$ customers canceling or with no customer arriving (probability $1-p$) and k customers leaving the system. When taking accepting new customers, the transition probabilities are given by:

$$P_{r,r'}(\text{Accept}) = \begin{cases} p \cdot (1-q)^r, & \text{if } r' = r+1, \\ p \cdot \binom{r}{r-k-1} q^{k+1}(1-q)^{r-k-1} & \\ \quad +(1-p) \cdot \binom{r}{r-k} q^k (1-q)^{r-k}, & \text{if } r' = r-k,\ r-k \geq 0,\ k \geq 0, \\ (1-p), & \text{if } r = r' = 0, \\ 0, & \text{if } r' > r+1. \end{cases} \tag{2}$$

When declining incoming customers, the transition probabilities are given by:

$$P_{r,r'}(Decline) = \begin{cases} \binom{r}{r-k} q^k (1-q)^{r-k} & r' = r-k, k \geq 0 \\ 0 & r' > r \end{cases} \tag{3}$$

The optimality equations for the optimal value function $V_n^*(r)$ are

$$\begin{cases} V_T^*(r) = R_T(r) \\ V_n^*(r) = \max(P_{r,r'}(Accept)V_{n+1}^*(r'), P_{r,r'}(Decline)V_{n+1}^*(r')) \quad 0 \leq n \leq T-1 \end{cases} \tag{4}$$

These can be solved by dynamic programming. The optimal policy takes the maximizing action for every epoch $0 \leq n \leq T-1$ and state r.

1.2 b) Optimality of booking-limit policies

1.2.1 Proof: $r \mapsto W_n^*(r)$ is quasi concave with maximum $\tilde{r} = b_n$

Step 1: Define $\tilde{r} \mapsto W_n^*(\tilde{r})$

We define a new value function W_n which is based on a slightly different state definition than the original state r. We use variable $\tilde{r}$ for the state of W_n and $\tilde{r}_n$ for the state $\tilde{r}$ at epoch n.
The state $\tilde{r}_n$ contains the number of bookings at epoch n including the possible acceptance of a new reservation arrived at epoch n but without subtracting the cancellations arrived at epoch n. The cancellations at epoch n will be accounted in the state $\tilde{r}_{n+1}$ of the next epoch.

To be able to relate the possible cancellations at epoch n (which will be accounted in $\tilde{r}_{n+1}$) to the state $\tilde{r}_n$ we allow a new reservation at epoch n (which is already accounted in $\tilde{r}_n$) to be canceled immediately at epoch n.

Then we can describe the transition of $\tilde{r}_n$ to $\tilde{r}_{n+1}$ as follows.

If at epoch n the decision is "reject" then:

$$\tilde{r}_{n+1} = \tilde{r}_n - \text{cancellations}(\tilde{r}_n)$$

If at epoch n the decision is "accept" then
with probability p:

$$\tilde{r}_{n+1} = \tilde{r}_n + 1 - \text{cancellations}(\tilde{r}_n)$$

with probability $1-p$:

$$\tilde{r}_{n+1} = \tilde{r}_n - \text{cancellations}(\tilde{r}_n)$$

Here $\tilde{r} - \text{cancellations}(\tilde{r})$ has the distribution of $\tilde{r}$ i.i.d. Bernoulli variables (actually the sum of $\tilde{r}$ non-cancellations).
We will write $B_{\tilde{r}} = \tilde{r} - \text{cancellations}(\tilde{r})$ for the $\tilde{r}$ non-cancellations.
Then the transition equations become:

If at epoch n the decision is "reject" then:

$$\tilde{r}_{n+1} = B_{\tilde{r}_n}$$

If at epoch n the decision is "accept" then
with probability p:

$$\tilde{r}_{n+1} = 1 + B_{\tilde{r}_n}$$

with probability $1-p$:

$$\tilde{r}_{n+1} = B_{\tilde{r}_n}$$

step 2: $\tilde{r} \mapsto W_n^*(\tilde{r})$ is quasi concave with maximum $\tilde{r} = b_n$

First we show that $W_T^*(\tilde{r})$ is quasi-concave.
We define $g(r)$ as the reward at the final epoch as a function of state r, where r is the state of all remaining booking with subtraction of cancellations.

It's clear that $g(r)$ is a quasi-concave function.
Then $W_T^*(\tilde{r}) = \mathbb{E}(\tilde{r}_T - \text{cancellations}(\tilde{r}_T)) = \mathbb{E}(g(B_{\tilde{r}_T}))$.
Using the fact that the expectation of a quasi-concave function of $\tilde{r}$ Bernoulli variables is quasi-concave it follows that $W_T^*(\tilde{r})$ is quasi-concave.

Now we show that if $W_{n+1}^*(\tilde{r})$ is quasi-concave then $W_n^*(\tilde{r})$ is also quasi-concave.
Based on the transition equations the Bellman optimality equation becomes:

$$W_n^*(\tilde{r}) = \max(\mathbb{E}(W_{n+1}^*(B_{\tilde{r}})), p * \mathbb{E}(W_{n+1}^*(1 + B_{\tilde{r}})) + (1-p) * \mathbb{E}(W_{n+1}^*(B_{\tilde{r}})))$$

Because we have assumed $W_{n+1}^*(\tilde{r})$ to be quasi-concave it follows that $\mathbb{E}(W_{n+1}^*(B_{\tilde{r}}))$ is quasi-concave in $\tilde{r}$ also.
We write:

$$g_{n+1}(\tilde{r}) = \mathbb{E}(W_{n+1}^*(B_{\tilde{r}}))$$
$$\tilde{g}_{n+1}(\tilde{r}) = \mathbb{E}(W_{n+1}^*(1 + B_{\tilde{r}}))$$
$$h_{n+1}(\tilde{r}) = p * \tilde{g}_{n+1}(\tilde{r}) + (1-p) * g_{n+1}(\tilde{r}))$$

Then

$$W_n^*(\tilde{r}) = \max(g_{n+1}(\tilde{r}), h_{n+1}(\tilde{r}))$$

Because $W_{n+1}^*(\tilde{r})$ is quasi-concave it follows that $W_{n+1}^*(1 + \tilde{r})$ is quasi-concave in $\tilde{r}$ as well (because it's just shifted in $\tilde{r}$). And therefore $\tilde{g}_{n+1}(\tilde{r})$ is quasi-concave too.
Also $\arg\max \tilde{g}_{n+1}(\tilde{r}) < \arg\max g_{n+1}(\tilde{r})$ (because of the $+1$ the max in $\tilde{g}_{n+1}$ is attained earlier with increasing $\tilde{r}$).
And $\max_{\tilde{r}} \tilde{g}_{n+1}(\tilde{r}) \geq \max_{\tilde{r}} g_{n+1}(\tilde{r})$ (because g needs more bernoulli's than $\tilde{g}$ to reach the maximum, these have larger spread which may reduce the maximum expected value).

We write $\tilde{a} = \arg\max \tilde{g}_{n+1}(\tilde{r})$ and $a = \arg\max g_{n+1}(\tilde{r})$ (with $\tilde{a} < a$).
And we define $a^* : \tilde{g}_{n+1}(a^*) = g_{n+1}(a^*)$.

Then we have the following:

- $\tilde{r} < \tilde{a}$: Both $g_{n+1}(\tilde{r})$ and $h_{n+1}(\tilde{r})$ are increasing hence the maximum is as well. On this interval $\tilde{g}_{n+1}(\tilde{r}) > g_{n+1}(\tilde{r})$ and hence, $h_{n+1}(\tilde{r}) > g_{n+1}(\tilde{r})$. Therefore the max function will follow $h_{n+1}(\tilde{r})$ which is increasing.
- $\tilde{a} \leq \tilde{r} < a^*$ Because at the start of this interval $\tilde{g}_{n+1}(\tilde{r})$ is decreasing and $g_{n+1}(\tilde{r})$ is still increasing we reach the point a^* where both are equal. Here $g_{n+1}(\tilde{r})$ becomes equal to $h_{n+1}(\tilde{r})$ as well. Before we reach a^*, $g_{n+1}(\tilde{r})$ is still increasing and $\tilde{g}_{n+1}(\tilde{r}) > g_{n+1}(\tilde{r})$ so $h_{n+1}(\tilde{r})$ is also still increasing. The max function will still return $h_{n+1}(\tilde{r})$ since $h_{n+1}(\tilde{r}) > g_{n+1}(\tilde{r})$ still holds.
- $a^* \leq \tilde{r} < a$: From a^*, $g_{n+1}(\tilde{r})$ increases until it reaches it's maximum at a. The max function will return $g_{n+1}(\tilde{r})$.
- $\tilde{r} \geq a$: both functions are decreasing and hence the maximum is decreasing as well. The max function will continue to return $g_{n+1}(\tilde{r})$.

So $W_n^*(\tilde{r}) = \max(g_{n+1}(\tilde{r}), h_{n+1}(\tilde{r}))$ reaches its maximum when $r = a$ and we have shown that it is increasing on $[0, a)$ and decreasing on $[a, \inf]$, hence $W_n^*(\tilde{r})$ is quasi concave.
Because we have already proven that $W_T^*(\tilde{r})$ is quasi-concave, using complete induction we can now conclude that $W_n^*(\tilde{r})$ is quasi-concave for all $0 \leq n \leq T$.
We write $b_n = \arg\max W_n^*(\tilde{r})$.

1.2.2 Proof: $r \mapsto V_n^*(r)$ is quasi concave with maximum $r = b_n$

The value function $V_n^*(r)$ is based on the original state r.

We can relate the state $\tilde{r}_{n+1}$ at epoch $n+1$ to the state r at epoch n as follows (state $\tilde{r}_{n+1}$ is just r_n plus new booked reservations because the cancellations from r_n are not yet subtracted in $\tilde{r}_{n+1}$):

If at epoch n the decision is "reject" then:

$$\tilde{r}_{n+1} = r_n$$

If at epoch n the decision is "accept" then
with probability p:

$$\tilde{r}_{n+1} = r_n + 1$$

with probability $1 - p$:

$$\tilde{r}_{n+1} = r_n$$

Then via the optimality equations we can relate $V_n^*(r)$ to $W_{n+1}^*(r)$:

$$V_n^*(r) = \max\{p \cdot W_{n+1}^*(r+1) + (1-p) \cdot W_{n+1}^*(r), W_{n+1}^*(r)\}$$

Then because $W_n^*(r)$ is quasi-concave we can follow the same logic as in the previous subsection to prove that that $V_n^*(r)$ is quasi-concave as well:

- $r < b_n - 1$: $V_n^*(r)$ follows $p \cdot W_{n+1}^*(r+1) + (1-p) \cdot W_{n+1}^*(r)$ because at this interval $W_{n+1}^*(r+1) > W_{n+1}^*(r)$ and this function is increasing.
- $r = b_n - 1$: $V_n^*(r)$ still follows $p \cdot W_{n+1}^*(r+1) + (1-p) \cdot W_{n+1}^*(r)$ and this function reaches its maximum.
- $r = b_n$: $V_n^*(r)$ starts following $W_{n+1}^*(r)$ which is at its maximum. The value of $V_n^*(r)$ is now higher than it was at $r = b_n - 1$ because at that point, it only contained the fraction of 1 optimal term while now it contains the full optimal term $W_{n+1}^*(b_n)$.
- $r > b_n$: $V_n^*(r)$ is decreasing because both terms in the max function are decreasing.

Therefore, $V_n^*(r)$ is quasi concave with optimum b_n.

Optimal to accept if $r < b_n$

At epoch n and state r we have the following value for the "Accept" decision:

$$V_{n,\text{accept}}(r) = p \cdot W_{n+1}^*(r+1) + (1-p) \cdot W_{n+1}^*(r)$$

for the "Reject" decision:

$$V_{n,\text{reject}}(r) = W_{n+1}^*(r)$$

Then because $W_n^*(r)$ is quasi-concave with $b_n = \arg\max W_n^*(r)$ it follows that the optimal policy is to accept if $r < b_n$ and reject if $r \geq b_n$.

1.3 c) Charts

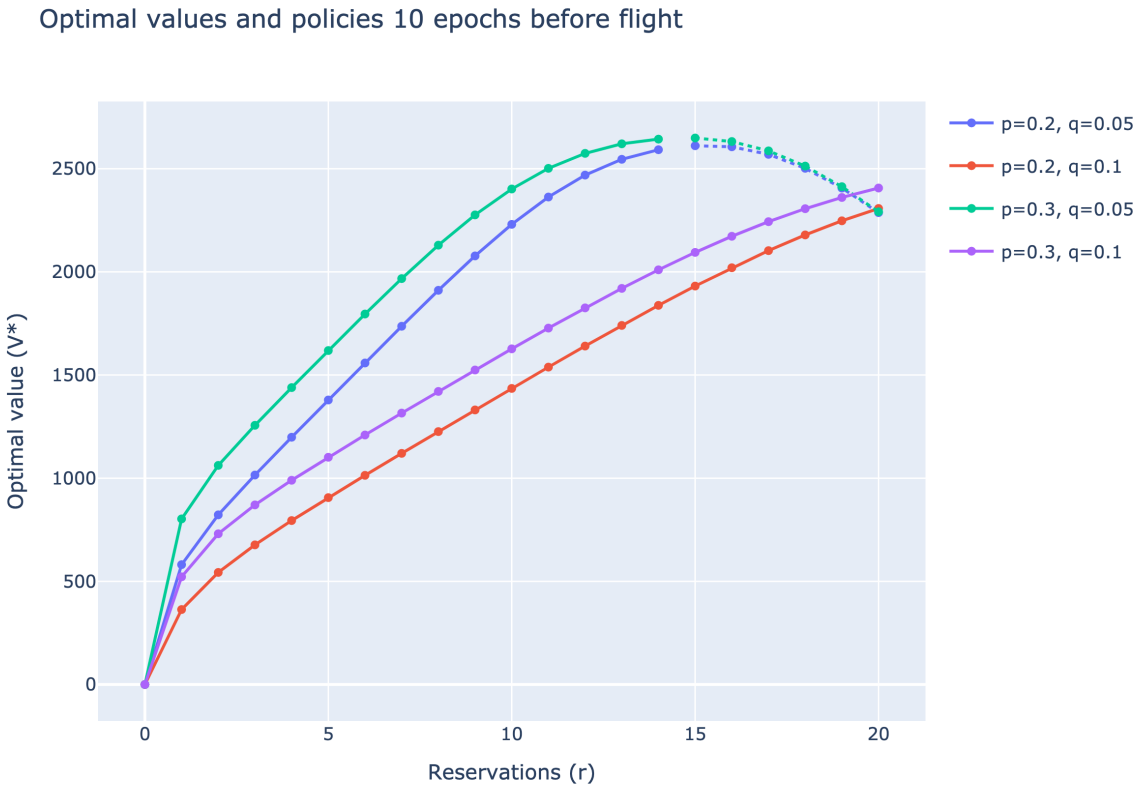
Conjectures

1. The booking limit b_n depends only on q and not on p.
2. For fixed p the booking limit b_n increases with q to anticipate on a higher number of cancellations.
3. The optimal value $V_n^*(r)$ decreases with q and increases with p

Code solutions

All code is available on our Github repo

Figure 1: Optimal policies



2 Assignment - Service rate control

Consider a discrete-time single-server queueing system that is observed every $\eta > 0$ units of time. The controller makes decisions at times $0, \eta, 2\eta, \ldots$. Jobs arrive following a Poisson distribution with rate 1.5 jobs per period of length η. There is a finite system capacity of eight units; that is, if arriving jobs cause the system content to exceed eight units, excess jobs do not enter the system and are lost.

At each decision epoch, the controller observes the number of jobs in the system and selects the service rate from a set of probability distributions indexed by elements of the set $B = \{0, 1, 2\}$. For each $b \in B$, let $f_b(n)$ denote the probability of n service completions within a period of length η with:

- $f_0(1) = 0.8$, $f_0(2) = 0.2$
- $f_1(1) = 0.5$, $f_1(2) = 0.5$
- $f_2(1) = 0.2$, $f_2(2) = 0.8$

The stationary reward structure consists of four components:

1. A constant reward $R = 5$ for every completed service.
2. An expected holding cost $h(s) = 2s$ per period when there are s jobs in the system.
3. A fixed cost $K = 3$ for changing the service rate.
4. A per-period cost $d(b)$ for using service rate b, where $d(0) = 0$, $d(1) = 2$, and $d(2) = 5$.

Determine a minimum-cost service rate control policy.

a) Model, algorithm and optimal policies

- Formulate the problem above as an infinite horizon Markov decision problem.
- Choose the optimality criterion that you find most reasonable (average costs or discounted costs). Also, choose a method (or methods) for computing the optimal policies and the value. Motivate your choices.
- Develop the model and the algorithm. Compute the optimal policies and the value. *(Note: you should write your own code for the algorithm, i.e., do not use an existing MDP implementation that is available as a code library or on the internet).*

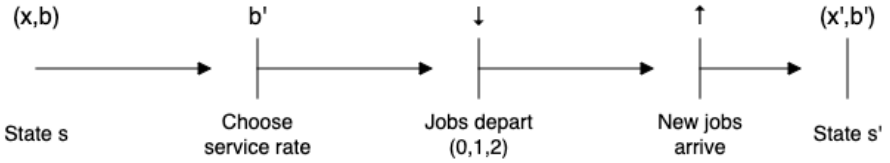
Please Report:

- Model description
- Your choice for an optimality criterion including motivation
- Solution algorithm (including motivation)
- Numerical results and a discussion of those

b) Model with service rate constraint

Now, we require that the server may work at service rate $b = 2$ at most 25% of the time. Model and solve this adjusted problem.

Figure 2: System state update between observations t and $t+1$



Solution

a) Model, algorithm and optimal policies

This model is also described in Puterman's book (section 3.7.2) and we follow the same approach but we will motivate the modeling choices.

The transition from state s to s' of the queuing system is as follows:

- After state s the controller chooses a service rate b'.
- After 0, 1 or 2 jobs have departed the system,
- new jobs arrive.
- The current number of jobs in the system is state s'

To formulate this problem as an infinite horizon Markov decision process (MDP), we define the key components:

Decision epochs (T)

$T = \{0, 1, 2, \dots\}$,
corresponding to observation times $0, \eta, 2\eta, \dots$

State space (S)

The state $s = (x, b)$ consists of the number of jobs $x \in X$ in the system:

$$X = \{0, 1, 2, \dots, 8\}$$

and the current service rate $b \in B$:

$$B = \{0, 1, 2\}$$

.

Action space (A)

At each decision epoch, the controller can take an action $a \in A = \{0, 1, 2\}$ equivalent to selecting a service rate $b' \in B$ for the next epoch.

Transition probabilities (P)

The transition probabilities depend on the arrival rate (Poisson distribution with a rate of 1.5 jobs per period of length η) and the service rate b' chosen.

- Let $f^a(k)$ denote the probability of k job arrivals in a period of length η. Since arrivals follow a Poisson distribution with rate 1.5:

$$f_a(k) = \begin{cases} \frac{(1.5)^k e^{-1.5}}{k!}, & \text{if} \quad k = 0, 1, 2, \dots \\ 0, & \text{otherwise} \end{cases}$$

- Let $f_b(n)$ denote the probability of $n \in \{1, 2\}$ service completions under service rate b, where the distributions of n are given for each $b \in B$. Of course this only applies to starting states s with $x > 1$ when the system contains enough jobs to be processed.

Define $p(s'|s, a)$ as the probability of transitioning to state $s' = (x', b')$ when the system is in state $s = (x, b)$ and action a is chosen. We can then distinguish the following cases:

$$p(s'|s, a \neq b') = 0, \tag{5}$$

$$p(s'|s, a = b') = \begin{cases} f^a(x') & \text{if } x \in \{0, 1\} \text{ and } x' < 8, \\ 1 - \sum_{k=0}^{7} f^a(k) & \text{if } x \in \{0, 1\} \text{ and } x' = 8, \\ \sum_{n=1}^{2} f_{b'}(n) \cdot f^a(x' - x + n) & \text{if } 1 < x \leq 8 \text{ and } x' < 8, \\ \sum_{n=1}^{2} f_{b'}(n) \cdot \left(1 - \sum_{k=0}^{7-x+n} f^a(k)\right) & \text{if } 1 < x \leq 8 \text{ and } x' = 8. \end{cases} \tag{6}$$

- $p(s'|s, a) = 0$, if $a \neq b'$. A choice by the controller for a service rate a results by definition in service rate b'.

- $p(s'|s, b') = f^a(x')$ if $x \in \{0, 1\}$ and $x' < 8$. The system is empty or contains only one job and the controller has to choose a service rate b'. The choice does not affect x' because there are either no jobs in the system to be processed or the system will be empty with probability 1 after processing. The number of jobs x' in state s' equals the number of arrivals.

- $p(s'|s, b') = 1 - \sum_{k=0}^{7} f^a(k)$ if $x \in \{0, 1\}$ and $x' = 8$. The system is empty or contains only one job and the controller has to choose a service rate b'. The choice does not affect x' because there are either no jobs in the system to be processed or the system will be empty with probability 1 after processing. The number of jobs x' in state s' equals the number of arrivals, but jobs arriving when the system has reached maximum capacity are rejected.

- $p(s'|s, b') = \sum_{n=1}^{2} f_{b'}(n) \cdot f^a(x' - x + n)$ if $1 < x \leq 8$ and $x' < 8$. The system contains more than 1 job, and the controller has to choose a service rate b'. The number of items in the system after processing depends on the number of processed and arriving jobs.

- $p(s'|s, b') = \sum_{n=1}^{2} f_{b'}(n) \cdot \left(1 - \sum_{k=0}^{7-x+n} f^a(k)\right)$ if $1 < x \leq 8$ and $x' = 8$. The system contains more than 1 job, and the controller has to choose a service rate b'. The number of jobs in the queue after the update of the states is a result of the number of processed and arriving jobs, but can not exceed 8.

Cost (C)

The cost function consists of four components:

1. **Reward for service completions**: For each random number of service completions $n \sim f_b(n)$, the system receives a reward of $R = 5$. The total reward for service completions when the system is in state s and action $a = b'$ is chosen equals: $5 \cdot \mathbb{E}(n)$.

2. **Holding cost**: The expected holding cost per period is $h(x) = 2x$ when there are x jobs in the system.

3. **Cost of changing service rate**: If the controller changes the service rate from one period to the next ($b \neq b'$), a fixed cost of $K = 3$ is incurred.

4. **Per-period cost for using service rate**: The cost $d(b')$ depends on the chosen service rate b', where:

$$d(0) = 0, \quad d(1) = 2, \quad d(2) = 5$$

Define the cost function $C(s, a)$ for state $s = (x, b)$ and chosen action (service rate) a as follows:

$$C(s,a) = \begin{cases} d(a) - K \cdot \mathbb{1}_{\{b \neq a\}}, & \text{if } s = 0 \\ -R + h(1, H) + d(a) + K \cdot \mathbb{1}_{\{b \neq a\}}, & \text{if } s = 1 \\ -R \cdot \sum_{n=1}^{2} f_a(n) \cdot n + h(x, H) + d(a) + K \cdot \mathbb{1}_{\{b \neq a\}}, & \text{if } s > 1 \end{cases}$$

where:

- $d(a)$ is the per-period cost of using service rate a.
- $K \cdot \mathbb{1}_{\{b \neq a\}}$ represents the cost K for changing the service rate (incurred only if $b \neq a$).
- R is the reward for processing a job.
- $h(x, H)$ is the holding cost, where H is a constant per-unit cost and x is the number of jobs in the system in state s.
- $f_a(n)$ is the probability of processing n jobs under chosen service rate $b = a$.

Optimality criterion

Assumption 1 (Unichain Assumption)**.** *For each stationary policy π, the associated Markov chain $\{X_n\}$ has a single recurrent class of states (Unichain Assumption).*

Proof that assumption 1 applies:

- Choose an arbitrary state $s(x + 1, b')$. There exists a one-step path from $s(x + 1, b')$ to $s(x, b)$ with $P(k = n - 1) > 0$ under arbitrary policy $\pi(s(x + 1, b'))$. This is evident as k can take any value from 0 to ∞ with probability greater than 0, and $n \subset k$ with positive probabilities for all n.
- Choose an arbitrary state $s(x - 1, b')$. There exists a one-step path from $s(x - 1, b')$ to $s(x, b)$ with $P(k = n + 1) > 0$ under arbitrary policy $\pi(s(x - 1, b'))$. Same reasoning as above.
- Choose an arbitrary state $s(x, b)$. There exists a one-step path from $s(x, b)$ to itself with $P(k = x) > 0$ if $x = \{0, 1\}$ and $P(k = n) > 0$ if $x > 1$, under arbitrary policy $\pi(s(x, b))$. Same reasoning as above.

By recurrence and transitivity, all states under an arbitrary policy π are in the same recurrent class. Therefore, the Markov chain is unichain.

For solving an infinite MDP problem, there are several methods available. The most common methods are:

1. Value iteration
2. Policy iteration
3. Linear programming

For this particular instance, we will use the linear programming approach to solve the MDP problem. Although part (a) of the problem set could be solved with all of the above-mentioned methods, the linear programming approach is particularly well suited for part (b) of the problem as it allows for a direct formulation of the MDP problem and the addition of special constraints. In this case the constraint that the server may work at service rate $b = 2$ at most 25

Linear programming algorithm

- Define $q_{s,a}$: the long-run fraction of decision epochs at which the system is in state s and action a is chosen.
- So, $\sum_{a \in A(s)} q_{s,a}$: the long-run fraction of decision epochs at which the system is in state s.
- Also, $\sum_{s \in S} \sum_{a \in A(s)} q_{s,a} = 1$.
- Under the unichain assumption, an average cost optimal policy can be computed using:
  - Minimize
$$\sum_{s \in S} \sum_{a \in A(s)} C(s,a) q_{s,a}$$
  - Subject to
$$\sum_{s \in S} \sum_{a \in A(s)} p(s'|s,a) q_{s,a} = \sum_{a \in A(s)} q_{s',a}, \quad \forall s' \in S$$
$$\sum_{s \in S} \sum_{a \in A(s)} q_{s,a} = 1$$
$$q_{s,a} \geq 0, \quad \forall a \in A(s), s \in S$$

Code solutions

All code is available on our Github repo

3 Assignment - A bandit problem

a

Show how 2.7.3 and 2.7.4 follow from 2.7.2.

$$V_\alpha(i) = \max\{\lambda + \alpha V_\alpha(i), r_i(c) + \alpha \sum_j p_{ij}(c) V_\alpha(j)\} \tag{2.7.2}$$

$$V_\alpha(i) = \max\{\frac{\lambda}{1-\alpha}, r_i(c) + \alpha \sum_j p_{ij}(c) V_\alpha(j)\} \tag{2.7.3}$$

$$V_\alpha(i) = \max\{\frac{\lambda}{1-\alpha}, \sup_{\tau > 0} \mathbf{E}_i(\sum_{n=0}^{\tau-1} \alpha^n r X_n(c) + \alpha^\tau \frac{\lambda}{1-\alpha})\} \tag{2.7.4}$$

2.7.3 follows from 2.7.2

The state i denotes whether we are currently operating bandit 1 $i = 0$ or we are operating bandit 2 $i \in \{1, ..., S\}$. From the moment we start operating bandit one, we have that the maximum reward is $\lambda + \alpha V_\alpha(i)$. In the subsequent epochs, this will not change (because the state doesn't change) and it will remain optimal to keep operating bandit 1. To see why this is the case, notice that the second arm freezes, hence the right hand side in the maximization function does not change anymore. So once the it is optimal to pull arm 1, this will remain the case. Hence, we can rewrite the reward we will receive at the moment we switch from bandit 1 to bandit 2 as an infinite amount of discounted λ's:

$$V_\alpha(i) = \lambda + \alpha V_\alpha(i) = \lambda + \alpha(\lambda + \alpha V_\alpha(i)) = \lambda \sum_{s=0}^{\infty} \alpha^s = \frac{\lambda}{1-\alpha} \tag{2.7.2}$$

Hence 2.7.3 follows from 2.7.2.

2.7.4 follows from 2.7.2

Following the same logic, we can prove that 2.7.4 follows from 2.7.3 we can reformulate our bandit problem to a stopping problem where we receive a reward $rXn(c)$ to continue and reward $\frac{\lambda}{1-\alpha}$ to stop. Then it follows that there is a certain epoch, say τ in which we expect it would be optimal to switch from bandit 2 to bandit 1, hence when it is optimal to pull arm 2, we have the following expected reward:

$$V_\alpha(i) = \sup_{\tau>0} \mathbf{E}_i(\sum_{n=0}^{\tau-1} \alpha^n r X_n(c) + \alpha^\tau \frac{\lambda}{1-\alpha}) \tag{7}$$

The first term in the expectation, $\sum_{n=0}^{\tau-1} \alpha^n r X_n(c)$, represents the discounted rewards up until period $\tau - 1$ from pulling arm 1. The second term, $\alpha^\tau \frac{\lambda}{1-\alpha}$, represents the stopping reward discounted by α^τ because we switch to arm 1 in period τ.

b

We can view this problem as a 2 armed bandit problem where we keep administering medicine A (bandit arm 2), and receive expected reward $\frac{x+1}{x+y+2}$ until we stop administering medicine 2 and start using medicine B with reward p. Similarly to the two armed bandit problem, once we stop using medicine A, the state freezes and we receive no update in x and y. Therefore, if it is optimal to start using the known medicine B, it will remain optimal to do so. The optimality equation is therefore:

$$V_\alpha(x,y) = \max\{p + \alpha V_\alpha(x,y), \frac{x+1}{x+y+2} + \alpha \sum_{(x',y')\in\Xi} P_{(x,y)(x',y')}(c) V_\alpha(x',y')\} \tag{8}$$

Where Ξ represents the possible next states when administering medicine A, $\Xi = \{(x+1,y),(x,y+1)\}$. Similar to the two armed bandit, if at a certain point we start using medicine B, it will remain optimal to do so. Hence we can rewrite the left part of the maximization function as an infinite stream of p rewards discounted by α, which simplifies to $\frac{p}{1-\alpha}$ because it is a geometric series. So we can rewrite:

$$V_\alpha(x,y) = \max\{\frac{p}{1-\alpha}, \frac{x}{x+y+1} + \alpha \sum_{(x',y')\in\Xi} P_{(x,y)(x',y')}(c) V_\alpha(x',y')\} \tag{9}$$

Since there are only two possible states we after (x,y), we can write the right term with ease:

$$V(x,y)_\alpha = \max\left\{\frac{p}{1-\alpha}, \frac{x}{x+y+1} + \alpha\left[\frac{x+1}{x+y+2} V_\alpha(x+1,y) + \frac{y+1}{x+y+2} V_\alpha(x,y+1)\right]\right\} \tag{10}$$

We can also express the optimality equations using the Gittins index (with discount factor α) $G(x,y)$ of state (x,y):

$$V(x,y)_\alpha = \max\left\{\frac{p}{1-\alpha}, \frac{G(x,y)}{1-\alpha}\right\} \tag{11}$$

Which shows that the optimal policy is to continue with medicine A as long as the Gittins index of the state (x,y) is greater than p.

c

When the estimated cure probability $\frac{(x+1)}{(x+y+2)}$ of medicine A is much higher then the cure probability $p = 0.4$ of medicine A then the optimal policy will be to continue with medicine A and the Gittins index of the corresponding state (x,y) will be higher than p. Reversely when the estimated cure probability $\frac{(x+1)}{(x+y+2)}$ of medicine A is much lower then the cure probability $p = 0.4$ of medicine B then the optimal policy will be to stop with medicine A and the Gittins index of the corresponding state (x,y) will be lower than $p = 0.4$.
Therefore we can expect that for a given number of experiments $n = x_n + y_n$ with x_n cures and y_n non-cures there will be a critical state (x_n^*, y_n^*) with $\frac{(x_n^*+1)}{(x_n^*+y_n^*+2)} \approx 0.4$ such that $G(x_n, y_n) > 0.4$ for

$x_n > x_n^*$ and $G(x_n, y_n) \leq 0.4$ for $x_n \leq x_n^*$.
Therefore only the Gittins indexes for states close to (x_n^*, y_n^*) are relevant for deciding to continue with medicine A or to stop.
Because we don't need the Gittins index for all states but for a (small) subset of states only, for the calculation of the Gittins indexes we prefer to use an online method instead of an offline method.

From the paper "2013-bandit-computations-annotated" we have selected the online-method Restart Formulation (Katehakis and Veinott).

According to the Restart Formulation we can calculate the Gittins index for a specific state $(\tilde{x}, \tilde{y})$ by formulating the optimality equation for a restart problem:

$$v = \max\{r^{(0)} + \alpha Q^{(0)} v, r^{(1)} + \alpha Q^{(1)} v\} \tag{12}$$

Here we have:

- v is the optimal value function (on the original state space (x, y)) for the restart problem.
- $r^{(0)}$ is the reward when restarting from $(\tilde{x}, \tilde{y})$.
- $Q^{(0)}$ is the transition matrix when restarting from $(\tilde{x}, \tilde{y})$.
- $r^{(1)}$ is the reward when continuing from (x, y).
- $Q^{(1)}$ is the transition matrix for continuing from (x, y).

We can then use Value Iteration (but also Policy Iteration or LP) to compute the value function v of the restart problem.
The Gittins index of $(\tilde{x}, \tilde{y})$ then follows from the computed value function as:

$$G(\tilde{x}, \tilde{y}) = (1 - \alpha) v(\tilde{x}, \tilde{y}) \tag{13}$$

For our specific problem we have:

$$r^{(0)}(x, y) = \frac{\tilde{x} + 1}{\tilde{x} + \tilde{y} + 2}$$

$$Q^{(0)}_{(x,y),(x',y')} = \begin{cases} \frac{\tilde{x}+1}{\tilde{x}+\tilde{y}+2} & x' = \tilde{x} + 1, y' = \tilde{y} \\ \frac{\tilde{y}+1}{\tilde{x}+\tilde{y}+2} & x' = \tilde{x}, y' = \tilde{y} + 1 \\ 0 & \text{otherwise} \end{cases}$$

$$r^{(1)}(x, y) = \frac{x + 1}{x + y + 2}$$

$$Q^{(1)}_{(x,y),(x',y')} = \begin{cases} \frac{x+1}{x+y+2} & x' = x + 1, y' = y \\ \frac{y+1}{x+y+2} & x' = x, y' = y + 1 \\ 0 & \text{otherwise} \end{cases}$$

We have used Value Iteration to compute the value functions of the restart formulations.

The table below shows the computed Gittins index for selected states. The states are selected around critical points where the optimal action changes from "stop" to "continue" (with medicine A). For each state the table shows the cure probability $\frac{x_n+1}{x_n+y_n+2}$ as well.

n	(x_n, y_n)	$G(x_n, y_n)$	$\frac{x_n+1}{x_n+y_n+2}$	optimal action
1	$(0, 1)$	0.500	0.333	continue
	$(1, 0)$	0.800	0.667	continue
10	$(3, 7)$	0.390	0.333	stop
	$(4, 6)$	0.475	0.417	continue
20	$(7, 13)$	0.397	0.364	stop
	$(8, 12)$	0.443	0.409	continue
30	$(11, 19)$	0.399	0.375	stop
	$(12, 18)$	0.430	0.406	continue
40	$(15, 25)$	0.399	0.381	stop
	$(16, 24)$	0.423	0.405	continue
50	$(18, 32)$	0.380	0.365	stop
	$(19, 31)$	0.400	0.385	stop
	$(20, 30)$	0.419	0.404	continue

When $G(x_n, y_n) > 0.4$ it's optimal to continue with medicine A and when $G(x_n, y_n) \leq 0.4$ it's optimal to stop.
The Gittins index is always higher or equal than the immediate reward from a state. Therefore for each state the Gittins index should be higher or equal than the cure probability (immediate reward). We see this confirmed in the table values.
We also see that for small n the difference between the Gittins index and the cure probability is higher than for larger values of n.
This makes sense because for small n the chance to get into a state with higher cure probability than 0.4 is higher than for larger values of n.
This means for lower n when there is more uncertainty about the cure probability there is a higher willingness to accept lower cure numbers (more exploration), which makes sense intuitively.

Code solutions

All code is available on our Github repo