Markov Decision Processes

Floske Spieksma

adaptation of the text by
R. Núñez-Queija

to be used at your own expense

October 30, 2015

Markov Decision Theory

In practice, decision are often made without a precise knowledge of their impact on future behaviour of systems under consideration. The field of Markov Decision Theory has developed a versatile appraoch to study and optimise the behaviour of random processes by taking appropriate actions that influence future evlotuion. Besides theory, this course also contains many application examples. The course assumes knowledge of basic concepts from the theory of Markov chains and Markov processes. The theory of (semi)-Markov processes with decision is presented interspersed with examples.

The following topics are covered: stochastic dynamic programming in problems with finite decision horizons; the Bellman optimality principle; optimisation of total, discounted and average expected cost or reward (infinite horizon); the methods of successive approximation, policy iteration and linear programming. Applications are taken from inventory, production and queueuing systems. Basic references to introductory textbooks are Derman [6], Howard [7], S.M. Ross [9], Puterman [8], Tijms [12, Chapter 3].

Contents

Chapter 1

Finite horizon decision problems

This chapter will treat stochastic decision problems defined over a finite period. A finite planning horizon arises naturally in many decision problems. Sometimes the planning period is exogeneously pre-determined. We will see examples of both cases. We will also introduce the basic concepts of Markov decision theory and the notation that will be used in the remanider. Concepts and notation will be motivated through the following example.

1.1 Investment example

Suppose an investor has €10.000 to his disposal and he must decide how to invest it, so as to maximise his total exepected returns. The investor may choose between investing all of his capital either in stock from company A or in stock from company B.

Investing in company A renders a profit of 100% (i.e. his investment is doubled) after one year with probability 0,10. With probability 0,90 however, there will no profit after one year, and the investor will get his investment back.

Company B has a higher risk profile, but renders higher expected returns. With probability 0,6 the investment is doubled, whereas with probability 0,4 the investment is completely lost.

As a consequence, the expected profit from investing €10,000 in company A is

$$0,10 \times \text{€}10,000 = \text{€1,000},$$

and in company B this is

$$0,60 \times \text{€10,000} + 0,4 \times (-\text{10,000})\text{€} = \text{€}2,000.$$

If not bankrupt, the investor can re-invest his money every year (each time €10,000 due to the popularity of both investments). What is the best investment strategy for the investor, if his goal is to maximize the expected profit after 5 years?

In this example, the planning horizon is exogeneously given and equal to five decision epochs. Clearly, the decision in later years depend on the profit made during the first year. A *strategy* assigns a sequence of decisions (one for each year) for each for each possible outcome of the process. While not bankrupt, the investor must choose between the two possible investments. In principle the investor could choose not to invest, but this is not an interesting option, since investment in A implies no rick on the invested capital.

Every year the capital either increases by €10,000, decreases by €10,000 or remains unchanged. The first two outcomes are only possible as long as the investor has not gone bankrupt. Thus, after t years, the capital can be one of $0, 10,000, \ldots, (t+1) \times 10,000$ euros.

In principle, a strategy must return a decision at each stage for *every possible sequence of previous decisions and outcomes of the investments so far.* For this simple example this amounts to 640 possible combinations. Appliccation of the technique of *dynamic programming* can drastically reduce the number of relevant decision rules. The technique will be formally described in section 1.3, but here we already illustrate it for this example.

The key idea is that we know what to do at the last stage. If the investor has not yet gone bankrupt, he has a capital $K_4 \geq 10,000$ (we leave out the reference to euros from now on). In order to maximise his return at time $T = 5$ he has to invest in B, making the final expected capital equal to $K_4 + 2.000$.

With this information, it is simple to determine the optimal decision at the last but one decision stage, i.e. time 3. Suppose that the capital at time 3 equals $K_3 \geq 10.000$. Investing in A leads to a capital K_3 or $K_3 + 10.000$. In both cases, we know that it is optimal to invest in B resulting in an additional expected profit of 2.000. Thus, investing in A leads to a total expected capital of

$$0,9 \times (K_3 + 2.000) + 0,1 \times (K_3 + 10.000 + 2.000) = K_3 + 3.000.$$

Next we evaluate what happens, if we invest in B at the last but one decision stage, time 3. The capital increases either to $K_3 + 10,000$ or $K_3 - 10,000$. A disctinction must now be made, wheter $K_3 = 10,000$ or $K_3 \geq 20,000$, because with a capital of 10.000 the decision to invest in B may lead to bankruptcy, disabling future revenues.

By a similar reasoning as before, we may conclude that the final expected capital after investing in B at time 3, equals

$$\begin{cases} 0,6 \times (K_3 + 10,000 + 2,000) + 0,4 \times (K_3 - 10,000 + 2,000) & \\ \qquad\qquad\qquad\qquad = K_3 + 4,000, & K_3 \geq 20,000 \\ 0,6 \times (K_3 + 10,000 + 2,000) + 0,4 \times 0 = 0,6K_3 + 7,200 = 13,200, & K_3 = 10,000. \end{cases}$$

In either case, the expected return is larger when investing in B than when investing in A, at the last but one decision stage time 3. Hence it is optimal to invest in B at time 3, if the investor is not yet bankrupt.

Of course, one can repeat the same arguments to decide what to do one stage earlier, time 2, and so on. This results in the outcomes listed below. The table only reports relevant information to answer the question raised (i.e. what is a good investment strategy over a period of 5 years starting with €10,000).

$\max \mathsf{E}(K_T \mid K_{T-n})$, (action)					potential end capital
$n=5, t=0$	$n=4, t=1$	$n=3, t=2$	$n=2, t=3$	$n=1, t=4$	K_5
0 (-)	0 (-)	0 (-)	0 (-)	0 (-)	0
16.711,20 (A)	15.528 (A)	14.400 (B)	13.200 (B)	12.000 (B)	10.000
	27.360 (B)	25.680 (B)	24.000 (B)	22.000 (B)	20.000
		36.000 (B)	34.000 (B)	32.000 (B)	30.000
			44.000 (B)	42.000 (B)	40.000
				52.000 (B)	50.000

Table 1.1: Optimal decisions and corresponding returns

Interpret the table as follows. At the end of the horizon, at time $T = 5$, the potential capitals can be $0, 10.000, \ldots, 50.000$. At time $t = 4$, the last decision epoch, the capitals can be $10.000, \ldots, 40.000$ with a maximum expected profit of 2.000 by investing in B. If the investor is bankrupt at time $t = 4$, he remains so. Time 4 corresponds with $n = 1$, the number of decision stages to go, etc.

Since we are given the starting capital of 10.000 we can omit calculating e.g. the best investment decision at time 1, for the situation of a capital of at least 30.000. That is why proceeding backwards to the present, less and less cases need be considered.

The final answer can be easily inferred from the table: the investor should invest €10.000 in A at time 0, and then his expected total capital at time $T = 5$ is €16.711,20. The optimal investment decision on subsequent time points depend on the realised capital. E.g., if at time $t = 2$ ($n = 3$) the capital is €20.000, then the investor should invest in B, and his expected total capital is €25.680.

1.2 The model

Before formalising the technique illustrated in the example of the previous section, we introduce some notation. We shall assume that there is a stochastic (discrete-time) process X_n, $n = 0, \ldots$ on a state $\boldsymbol{S}$. Given that $X_n = i$, a decision is chosen from the action set $\mathsf{A}(i)$. A priori, the action set may depend on both time and state, but for notational convenience we will only assume dependence on the state. From now on we will index time by n.

The probabilistic law according to which the process subsequently evolves, may depend on X_n and the action $A_n \in \mathsf{A}(X_n)$ chosen.

Assumption 1.2.1 The state space $\boldsymbol{S}$ is countable and the action space $\mathsf{A} = \cup_i \mathsf{A}(i)$ is finite.

In the example $\boldsymbol{S} = \{0, 10.000, \ldots, 60.000\}$, $\mathsf{A}(0) = \{0\}$, meaning that no investment can be done, $\mathsf{A}(i) = \{A, B\}$ for $n = 0, \ldots, 4$, $i \neq 0$.

We further specifically assume that

$$\begin{aligned}
\mathsf{P}\{X_{n+1} = i_{n+1} \mid X_0 = i_0, A_0 = a_0, \ldots, X_{n-1} = i_n, A_n = a_n\} = \\
&= \mathsf{P}\{X_{n+1} - i_{n+1} \mid X_n = i_n, A_n = a_n\} \\
&=: p_{i_n, i_{n+1}}(a_n),
\end{aligned}$$

where $i_0, \ldots, i_{n+1} \in \boldsymbol{S}$, and $a_0 \in \mathsf{A}_1(i_0), \ldots, a_n \in \mathsf{A}_{n+1}(i_{n+1})$. This relation states that if state and action chosen at time n are known, then the state at time $n+1$ is *independent of the history*

$$H_{n-1} = (X_0, A_0, \ldots, X_{n-1}, A_{n-1})$$

before time n, i.o.w. the transition mechanism has the Markov property. Suppose that the action $a = f(i)$ is a given function f of the state, then $\{X_n\}_n$ is a Markov chain with transition matrix $P(f) = \{p_{i,j}(f(i))\}_{i,j\in S}$. If $a = f_n(i)$ is a time-dependent function of the state, then $\{X_n\}_n$ is a non-stationary Markov chain with transition matrix $P(f_n) = \{p_{i,j}(f_n(i))\}_{i,j\in S}$ at time n. Suppose a(n immediate) reward $r_{i,j}(a)$ is earned, whenever the process X_n is in state i at time n, action a is chosen and the process moves to state j. Then

$$r_i(a) = \sum_{j\in S} p_{ij}(a) r_{ij}(a)$$

represents the *expected* reward, if action a is taken while in state i. In most problems we therefore model the reward to depend only on the current state and action. At the end of the horizon T, we may wish to allow a terminal reward to be earned: q_i will denote the terminal reward, when in state i, at time T.

To facilitate the analysis, we shall make the following technical assumption. This assumption can be relaxed, see section 1.3.

Assumption 1.2.2 The expected (immediate) rewards and terminal rewards are uniformly bounded, i.e. $\sup_{i\in S} \max_{a\in \mathsf{A}(i)} |r_i(a)| < \infty$ and $\sup_{i\in S} |q(i)| < \infty$.

Decision rules and strategies Given history h_{n-1} and state i_n, a *decision rule* $\sigma^n_{h_{n-1},i_n}$ at stage n is a probability distribution on the action space $\mathsf{A}(i_n)$ associated with state i_n. $\sigma^n_{h_{n-1},i_n}(a)$ is then the probability that action $a \in \mathsf{A}(i_n)$ is selected.

The decision rule σ^n is a map that associates with each history upto stage $n-1$ and state at stage n a probability distribution.

A *strategy* is a sequence of decision rules: $\boldsymbol{\sigma} = (\sigma^0, \ldots, \sigma^{T-1})$. We distinguish several types of strategies:

- $\boldsymbol{\sigma} = (\sigma^0, \ldots, \sigma^{T-1})$ is a *Markov strategy* if for each stage n, the decision rule σ^n is independent of the history h_{n-1}, e.g. $\sigma^n_{h_{n-1},i_n}(a) = \sigma^n_{h'_{n-1},i_n}(a)$, $a \in \mathsf{A}(i_n)$, $i_n \in \boldsymbol{S}$, for any history h_{n-1}, h'_{n-1}. We may then write $\sigma^n_{i_n}$ and $\sigma^n_{i_n}$ to indicate decision rule and probabilities.

  Because we consider only objective functionals based on expectations of rewards, one can show that to each strategy there exists a Markov strategy achieving the same value. We will therefore restrict to Markov strategies in the sequel.

- $\boldsymbol{\sigma}$ is a *stationary strategy*, if all decision rules are equal: $\sigma^n = \sigma^0$ for $n = 0, \ldots, T-1$.

- $\boldsymbol{\sigma}$ is a *deterministic (Markov) strategy*, if for each n, the probability distribution $\sigma^n_{i_n}$ is degenerate. In this case we write $\boldsymbol{\sigma} = \boldsymbol{f} = (f^0, f^1, \ldots, f^{T-1})$ where $f^n_{i_n} \in \mathsf{A}(i_n)$ is the decision that is selected with probability 1, in state i_n at stage n.

Exercise 1.1 Determine the transition probabilities, rewards and terminal rewards for the investment example. Classify the optimal policy.

Objective function Given (Markov) strategy $\boldsymbol{\sigma}$, the transition probability mechanism at stage n is completely specified by the $\boldsymbol{S} \times \boldsymbol{S}$-matrix P^{σ^n}, with elements

$$p_{ij}(\sigma^n) = \sum_{a \in \mathsf{A}(i)} p_{ij}(a)\sigma_i^n(a), \quad i, j \in \boldsymbol{S}.$$

Similarly, the associated expected rewards are specified by the vector r^{σ^n} with components

$$r_i(\sigma^n) = \sum_{a \in \mathsf{A}(i)} r_i(a)\sigma_i^n(a), \quad i, j \in \boldsymbol{S}.$$

One can write the total expected reward vector $V_T^{\boldsymbol{\sigma}}$ associated with strategy s as follows: for $i \in \boldsymbol{S}$

$$\begin{aligned} V_T^{\boldsymbol{\sigma}}(i) &= \mathsf{E}^{\boldsymbol{\sigma}}\Big[\sum_{n=0}^{T-1} r_{X_n}(A_n) + q_{X_T} \,|\, X_0 = i\Big] \\ &= \sum_{n=0}^{T-1} \mathsf{E}^{\boldsymbol{\sigma}}(r_{X_n}(A_n) \,|\, X_0 = i) + \mathsf{E}^{\boldsymbol{\sigma}}(q_{X_T} \,|\, X_0 = i) \\ &= \sum_{n=0}^{T-1} \mathsf{E}_i^{\boldsymbol{\sigma}}(r_{X_n}(A_n)) + \mathsf{E}_i^{\boldsymbol{\sigma}}(q_{X_T}). \end{aligned}$$

The superscript in the expectation operator $\mathsf{E}^{\boldsymbol{\sigma}}[\ldots]$ reflects the fact that the strategy determines the probability law according to which the process $\{(X_n, A_n)\}_n$ evolves. The subscript in $\mathsf{E}_i^{\boldsymbol{\sigma}}[\ldots]$ is a shorthand notation for the conditional expectation given initial state $X_0 = i$. The analogous notation will be used for the probability operator, i.e. $\mathsf{P}^{\boldsymbol{\sigma}}[\ldots]$, $\mathsf{P}_i^{\boldsymbol{\sigma}}[\ldots]$.

Since we restrict to Markov strategies, the expected reward vector can be expressed more explicitly as follows:

$$\begin{aligned} V_T^{\boldsymbol{\sigma}}(i) &= r(\sigma^0) + P(\sigma^0)r(\sigma^1) + P(\sigma^0)P(\sigma^1)r(\sigma^2) + \cdots + \\ &\qquad P(\sigma^0)\cdots P(\sigma^{T-2})r(\sigma^{T-1}) + P(\sigma^0)\cdots P(\sigma^{T-1})q \\ &= \sum_{n=0}^{T-2} P(\sigma^0)\cdots P(\sigma^{n-1})r(\sigma^n) + P(\sigma^0)\cdots P(\sigma^{T-1})q. \end{aligned}$$

Let $\boldsymbol{\sigma'} = (\sigma^1, \ldots, \sigma^{T-1})$ be the restricted strategy for a $T-1$-horizon problem starting at time 1 and running till time T, and $V_{T-1}^{\boldsymbol{\sigma}'}$ the associated total expected reward. It immediately follows from the above expressions, that

$$V_T^{\boldsymbol{\sigma}} = r(\sigma^0) + P(\sigma^0)V_{T-1}^{\boldsymbol{\sigma}'}.$$

This shows that we can calculate $V_T^{\boldsymbol{\sigma}}$ recursively 'backwards' in time, for a fixed strategy. In the investment example we chose an optimal decision rule at each recursive step. The validity of this procedure will be discussed next.

1.3 Bellman's optimality principle

Suppose our goal is to maximise $V_T^{\boldsymbol{\sigma}}$ over all strategies $\boldsymbol{\sigma}$.

Definition 1.3.1 Let

$$V_T^* = \sup_{\boldsymbol{\sigma}} V_T^{\boldsymbol{\sigma}}.$$

The function V_T^* is called the (T-horizon) value function. $\boldsymbol{\sigma}^*$ is called an *optimal strategy* when

$$V_T^{\boldsymbol{\sigma}^*} = V_T^*.$$

Note that the assumptions imply that V_T^* is always finite. Further $\boldsymbol{\sigma}^*$ is optimal, when it attains the maximum value *for each initial state*. A priori it is not clear, that there is a strategy that attains the maximum value, and even the maximum value for all initial states.

Bellman's optimality princinple (or equivalently, the stochastic dynamic programming optimality equation) given in (1.3.1) paves the way to determining an optimal strategy, which turns out to be deterministic, non-stationary!

Theorem 1.3.2 *Let $V_0(i) = q_i$ and let $V_n(i)$, $n = 1, 2, \ldots$ be recursively given by*

$$V_n(i) = \max_{a \in \mathsf{A}(i)} \Big\{ r_i(a) + \sum_{j \in S} p_{ij}(a) V_{n-1}(j) \Big\}, \quad i \in \boldsymbol{S}, \tag{1.3.1}$$

then

$$V_n(i) = V_n^*(i), \quad i \in \boldsymbol{S}, n = 0, 1, \ldots$$

and any strategy $\mathbf{f}^n = (f^n, f^{n-1}, \ldots, f^1)$ determined by

$$f_i^n = \arg\max_{a \in \mathsf{A}(i)} \{ r_i(a) + \sum_{j \in S} p_{ij}(a) V_{n-1}(j) \Big\}, \quad i \in \boldsymbol{S},$$

attains the optimal exptected total reward over the periods $T - n, \ldots, T$, and hence over periods $1, \ldots, n$. In other words, $\mathbf{f^n}$ is n-horizon optimal.

Proof. The proof is by induction on n. It is left as an exercise. QED

Remark 1.3.1

- V_T^* is the optimal reward over time horizon T and $\mathbf{f}^T$ is an optimal strategy (there may be more than one!).
- We emphasize that, given an optimal strategy for n periods, to determine an optimal strategy for the $(n+1)$-period maximisation, we only need to compute f^{n+1} and then use the optimal stategy for the n-period maximisation.

Note One can generalise Theorem 1.3.2 to allow the state space, transition probabilities and rewards to depend on time.

Exercise 1.2 Prove Theorem 1.3.2.

Example 1.3.1 Consider a Markov decision problem with two states, 0 and 1, and two decisions, 1 and 2, per state. This means that $\boldsymbol{S} = \{0, 1\}$ and $\mathsf{A}(0) = \mathsf{A}(1) = \{1, 2\}$. The rewards are given by

$$r(1) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad r(2) = \begin{pmatrix} 2 \\ 2 \end{pmatrix},$$

and the transition probabilities by

$$P(1) = \begin{pmatrix} 1/2 & 1/2 \\ 2/3 & 1/3 \end{pmatrix}, \quad P(2) = \begin{pmatrix} 1/4 & 3/4 \\ 1/3 & 2/3 \end{pmatrix}$$

The terminal rewards are given by

$$q = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

We wish to determine the *minimum* reward over a period with two decision epochs, i.e. $T = 2$. First,

$$V_0^*(0) = 2, \quad V_0^*(1) = 1.$$

Further,

$$\begin{aligned}
V_1^*(0) &= \min\{0 + \tfrac{1}{2} \cdot 2 + \tfrac{1}{2} \cdot 1, 0 + \tfrac{1}{4} \cdot 2 + \tfrac{3}{4} \cdot 1\} = \tfrac{5}{4}, \quad \text{for } f_0^1 = 2 \\
V_1^*(1) &= \min\{2 + \tfrac{2}{3} \cdot 2 + \tfrac{1}{3} \cdot 1, 2 + \tfrac{1}{3} \cdot 2 + \tfrac{2}{3} \cdot 1\} = \tfrac{10}{3}, \quad \text{for } f_1^1 = 2 \\
V_2^*(0) &= \min\{0 + \tfrac{1}{2} \cdot \tfrac{5}{4} + \tfrac{1}{2} \cdot \tfrac{10}{3}, 0 + \tfrac{1}{4} \cdot \tfrac{5}{4} + \tfrac{3}{4} \cdot \tfrac{10}{3}\} = 2\tfrac{13}{16}, \quad \text{for } f_0^2 = 2 \\
V_2^*(1) &= \min\{2 + \tfrac{2}{3} \cdot \tfrac{5}{4} + \tfrac{1}{3} \cdot \tfrac{10}{3}, 2 + \tfrac{1}{3} \cdot \tfrac{5}{4} + \tfrac{2}{3} \cdot \tfrac{10}{3}\} = 3\tfrac{17}{18}, \quad \text{for } f_1^2 = 1.
\end{aligned}$$

Example 1.3.2 (Inventory control) A storage depot is used to keep production items in stock. At most 2 items can can be stored at the same time. At the end of each week the inventory level (i.e. the number of items in stock) is monitored and a decision is made about the number of new items to be ordered from the production facility. An order that is placed on Friday is delivered on Monday at 7.30 am. The cost of an order consists of a fixed amount of €100 and an additional €100 per ordered item. Requests for items arrive randomly at the storage depot; with probability 1/4 there is no demand during the week, with probability 1/2 exactly one item is requested and with probability 1/4 two items.

If the weekly demand exceeds the inventory stock, it is delivered directly from the production facility at the expense of €300 per item. The depot manager wishes to minimise the expected ordering cost over a pre-determined finite horizon planning period. The items in stock at the end of the planning period render no value.

Exercise 1.3 a) Formulate the problem as a Markov Decision problem, by determining the state space, action spaces, rewards, terminal rewards and transition probabilities.

b) Determine for each possible initial state the minimum total expected cost over a period of 2 weeks.

c) Suppose that the value of each item in stock at the end of the planning period of 2 weeks equals q euro. For which value(s) of q does the optimal strategy change?

It may be beneficial to first study a class of Markov decision problems from a theoretical point of view. If one can e.g. prove that an optimal policy within such a class of problems has a specific structure, then this can be used to reduce the computational complexity of determining the optimal policy for a given practical application.

Example 1.3.3 (Exercising a call option) One of the most active places in which dynamic programming is used today is Wall Street. To illustrate, consider the problem of determining when to exercise an (American) call option to buy a stock ignoring commissions. The option gives the purchaser the right to buy the stock at the *strike price* s^* on any of the next T days.

Two questions arise. When should the option be exercised? What is its value? To answer these questions requires to formulate the problem as a Markov decision problem, and to solve the optimality equation (1.3.1). This allows to find the value of the option as well as an optimal option-exercise policy, for any day before the expiration day T.

Consider the following stock-price model. Suppose that the stock price is s on day $T-n$, i.e. n days before the option expires.. Index n will count the number of days till expiration.

Let $n < T$. Assume that the stock price on the next day $T-n+1$ equals sR_n, where $R_0, R_1, \ldots$ are independent, identically distributed nonnegative, finite-valued discrete random variables with expectation ρ. Then $r_n = R_n - 1$ is the rate of return for day n, and $\mathsf{E}r_n = \rho - 1$ is the expected rate of return that day.

Let $V_n^*(s)$ be the maximum value of the option on day n, when the market price of the stock is s. There are two alternatives that day. One is to exercise the option to buy the stock at the strike price and immediately resell it, which earns $s - s^*$. The other is not to exercise the option that day, in which case the maximum expected income in the remaining days is $\mathsf{E}V_{n-1}^*(sR_n)$. Since one seeks the alternative with higher expected future income, the value of the option on expiration day is $V_0^*(s) = (s-s^*)^+$ and on day $T-n$

$$V_n^*(s) = \max\{s - s^*, \mathsf{E}V_{n-1}^*(sR_n)\}. \tag{1.3.2}$$

Thus it is optimal to exercise the option on expiration day if $s - s^* > 0$, and not to do so otherwise. Further it is optimal to exercise the option on day $T-n$ if $s - s^* > \mathsf{E}V_{n-1}^*(sR_n)$, and not to do so otherwise.

Nonnegative Expected Rate of Return. Consider now the question when it is optimal to exercise the option. It turns out that as long as the expected rate of return is nonnegative, $\rho \geq 1$, the answer is to wait until the expiration day. To establish this fact, it suffices to prove that

$$V_n^*(s) = \mathsf{E}V_{n-1}^*(sR_n), s \geq 0, n = 1, \ldots, T. \tag{1.3.3}$$

To that end, observe that $V_n^*(s) \geq s - s^*$ for $n = 0, \ldots, T$ and all s. Hence $V_{n-1}^*(sR_n) \geq sR_n - s^*$. By taking expectations and using $\rho \geq 1$, $\mathsf{E}V_{n-1}^*(sR_n) \geq s\rho - s^* \geq s - s^*$. This proves (1.3.3).

This allows to explicitly compute the value of the option at day $T-n$, n days before expiration:

$$V_n^*(s) = \mathsf{E}(s\prod_{i=1}^n R_i - s^*)^+,$$

and $s\prod_{i=1}^n R_i$ is the price at expiration.

Negative Expected Rate of Return. Suppose now that the expected rate of return is negative, i.e. $\rho < 1$. To analyse Eqn. (1.3.2), it turns out to be useful to consider a associated problem with optimal expected net reward $U_n^*(s) = V_n^*(s) - (s - s^*)$ on day $T-n$, when the stock price is s. One can check that the associated terminal reward is $(s^* - s)^+$, when the stock price is s, and that the optimality equation is given by

$$U_n^*(s) = \max\{0, s\rho - s + \mathsf{E}U_{n-1}^*(sR_n)\}.$$

Note that the two problems have the same optimal strategies. Further, it turns out that $U_n^*(s)$ has nice properties that can be easily analysed.

In particular, $s \mapsto U_n^*(s)$ is non-increasing, with $\lim_{s\to\infty} U_n^*(s) = 0$, for $n = 0, \ldots, T$. This can be shown inductively. It clearly holds for $s \mapsto U_0^*(s)$. By the induction assumpion it holds for $s \mapsto U_{n-1}^*(sR_n)$ for each realisation R_n. Hence, for $s \mapsto \mathsf{E}U_{n-1}^*(sR_n)$, and so $s \mapsto s\rho - s + \mathsf{E}U_{n-1}^*(sR_n)$ is decreasing, unbounded below.

Then there exists a smallest value s_n such that $s(\rho - 1) + \mathsf{E}U_{n-1}^*(sR_n) \leq 0$, $s \geq s_n$. This yields that for the original problem it is optimal to wait at time $T - n$, when $s < s_n$, and it is optimal to exercise the option when $s \geq s_n$.

It directly follows that $s \mapsto U_n^*(s)$ is non-increasing.

Exercise 1.4 (Exercising a put option) Consider the reverse problem of deciding when to exercise a put option to sell a stock at the strike price s^* during any of the next T days. Suppose that if the price of the stock on day $T-n$ is s, then the price the next day equals sR_n, as in the call option model. The random variables $R_1, R_2, \ldots$, have the same characteristics as in the above model.

The goal is to maximize the expected net revenue from an option to sell the stock any time in the next T days at the strike price when the price with n days to expiration is s. Assume that if one exercises the option to sell the stock at the strike price, then one first buys the stock that day at the market price.

i) Formulate the problem as a Markov decision model, by determining the state space, action spaces, reards, terminal rewards and the transition probabilities. Formulate the optimality equation (1.3.1) for $V_n^*(s)$.

ii) Nonpositive Expected Rate of Return. Show that if $\rho \leq 1$, then it is optimal not to exercise the option before expiration. *Hint: show that* $V_n(s) = \mathsf{E}V_{n-1}(sR_n)$ *for all* $n \geq 1$, *explain why this yields the desired result.*

iii) Positive Expected Rate of Return. Determine the optimal exercise policy with T days to expiration by induction on n, if $\rho > 1$. **Hint**: First show that $V_n^*(0) = s^*$, for each $n = 0, \ldots$.

Next observe that at time $T-n$, if the stock price falls from a positive level to 0, which occurs with probability $\mathsf{P}\{R_n = 0\}$, then the stock price can never become positive again. For this reason, it suffices to consider the case of positive stock prices $s > 0$. For that case, rewrite the optimality equation in (i) to reflect this fact. Then consider the model with optimal expected net reward $U_n^*(s) = \frac{1}{s}(V_n^*(s) - (s^* - s))$. Show that the optimality equation for $U_n^*(s)$ is given by: $U_n^*(s) = \max\{0, 1-\rho+\mathsf{E}R_nU_{n-1}^*(sR_n)\mathbf{1}_{\{R_n>0\}}\}$ for $s > 0$. Show inductively that $U_n^*(s)$ is increasing in s and compute the limits as $s \to \infty$ and $s \downarrow 0$.

Exercise 1.5 (Airline overbooking) An airline seeks a reservation policy for a flight with S seats that maximises 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 reservation pays the fare $f > 0$ at flight time. If $b \geq 0$ passengers with 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 reservations on hand and decides whether to *book* (accept) or decline any reservation request during the hour. Say the number of reservations is r after the decision taken has been implemented. Then each of these r reservations may cancel during the hour, independently of each other, with probability q (this means *all* booked reservations so far).

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 r seats have been booked at the beginning of the hour, before the accept/decline decision has been taken. reservation requests and cancellations during the hour. Let $W^*_n(r)$ be the maximum expected future profit when r seats have been booked after booking or declining any reservation request, but before cancellations. The aim is to determine an optimal reservation strategy for any value of the number of booked seats at the beginning of each hour till the flight time T.

a) Formulate the problem as a Markov decision model, by determining the state space, action spaces, reards, 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 $r \mapsto \mathsf{E}(g(B_r))$ is quasiconcave B_r 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 reservations on hand at the beginning of the hour (before a decision has been taken).

(c) Solve the problem when the parameters are as follows. Assume that T=30, $c(b) = fb$; S=10; $f = €300 + F$, where F is the sum of the first numbers associated with the first letters of the first and last name (of one of the couple..). E.g. letter F becomes 6; $p = 0,2$ and $0,3$, $q = 0,05$ and $0,10$; and $r \leq 20$ (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 that you found, 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.

1.3.1 Unbounded rewards

We have already mentioned that Theorem 1.3.2 remains valid, when we allow the transition probabilities $p_{ij}(a)$ and rewards $r_i(a)$ to depend on time. In other words, let $p_{ij}^n(a)$, $a \in \mathsf{A}(i)$, $i, j \in \boldsymbol{S}$, and $r_i^n(a)$, $a \in \mathsf{A}(i)$, $i \in \boldsymbol{S}$, be the transition probabilities and rewards at time $T - n > 0$, with T the time-horizon. Then the optimality equation (1.3.1) translated the time-dependent case becomes

$$V_n^*(i) = \max_{a \in \mathsf{A}} \{r_i^n(a) + \sum_j p_{ij}^n(a) V_{n-1}^*(j)\}, \quad i \in \boldsymbol{S}. \tag{1.3.4}$$

As has been indicated, the restriction to bounded rewards is not amenable. We can often use the following transformation trick.

Suppose that there exists a function $M : \mathcal{I} \to (0, \infty)$ and a constant $c \in \mathbf{R}$, such that for $a \in \mathsf{A}(i), i \in \boldsymbol{S}$

$$\begin{array}{rcl} \sum_j p_{ij}(a) M(j) & \leq & cM(i), \\ \sup\limits_{i \in S} \max\limits_{a \in \mathsf{A}(i)} \dfrac{|r_i(a)|}{M(i)} & < & \infty \\ \sup\limits_{i \in S} \dfrac{|q_i|}{M(i)} & < & \infty \end{array} \tag{1.3.5}$$

Enlarge $\boldsymbol{S}$ with a coffin state $\Delta \not\in \boldsymbol{S}$, and write $\boldsymbol{S}_\Delta = \boldsymbol{S} \cup \{\Delta\}$. We define a transformed problem with transition probabilities

$$\tilde{p}_{ij}(a) = \begin{cases} \frac{p_{ij}(a) M(j)}{cM(i)}, & a \in \mathsf{A}(i), i, j \in \boldsymbol{S} \\ 1 - \sum_j \frac{p_{ij}(a) M(j)}{cM(i)}, & a \in \mathsf{A}(i), i \in \boldsymbol{S}, j = \Delta \\ 1, & j = i = \Delta, a \in \mathsf{A}(\Delta), \end{cases}$$

where we take $\mathcal{A}(\Delta) = \{0\}$. De coffin state is needed to make the transition probabilities sum up to 1. The rewards are changed accordingly by putting

$$\tilde{r}_i(a) = \begin{cases} \frac{r_i(a)}{M(i)} & a \in \mathsf{A}(i), i \in \boldsymbol{S} \\ 0, & a = 0, i = \Delta. \end{cases}$$

Similarly, define terminal rewards $\tilde{q}_i = q_i / M(i)$.

The transformed problem has bounded rewards and satisfies the assumptions made. Clearly the associated total expected rewards associated with strategy $\boldsymbol{\sigma}$ in the transformed problem are not equal to the total expected rewards associated with $\boldsymbol{\sigma}$ in the original problem.

Exercise 1.6 a) Reformulate and prove Theorem 1.3.2 for the time-dependent case, described in the first paragraph of this section.

b) Define stage dependent rewards $\tilde{r}_i^n(a)$, $n = 0, \ldots, T-1$, such that $V_T^{\boldsymbol{\sigma}}(i) = M(i) \tilde{V}_T^{\boldsymbol{\sigma}}(i)$, where $\tilde{V}_T^{\boldsymbol{\sigma}}$ is the total expected reward for de T-horizon problem with transformed transition probabilities given above, and rewards $\tilde{r}_i^n(a)$, $i \in \boldsymbol{S}$.

c) Write the recursion (1.3.4) for the case described in (b).

d) Show that the optimal strategy for the transformed problem from (b) is optimal for the original problem.

e) Suppose now that $\mathsf{A}(i)$ is not finite but compact, in the time-independent problem. Give suggestions as to what additional assumptions could be made, so that solutions to (1.3.1) exist and yield an optimal strategy.

Chapter 2

Infinite horizon: total expected rewards

2.1 Introduction

There are several reasons why it is not always desirable to restrict to a finite horizon. For instance the planning period is long, but one does not want to specify it beforehand. In this case one might expect that the optimal decision at the present time 0 is rather insensitive to the precise horizon length. Then the problem is better analysed by assuming that the horizon is infinite.

A second reason is that the problem does not admit a natural definition of a horizon. E.g. in the investment problem one might want to maximise the probability of attaining a capital of at least €50.000.

Clearly, we have to impose conditions that ensure that the expected infinite horizon rewards are finite. A useful approach is the following.

Assumption 2.1.1 There exists a state, $\Delta \in \boldsymbol{S}$ say, with the following properties:

- Δ is an absorbing, zero reward state under any strategy, i.e. $\mathsf{A}(\Delta) = \{0\}$, $p_{\Delta,\Delta}(0) = 1$, $r_\Delta(0) = 0$;

- the expected time to reach the absorbing state Δ is uniformly bounded in initial states and strategies. In particular, define let $y_i^0 = 1$, $i \in \boldsymbol{S}$, and let

  $$y_i^n = \max_{a \in \mathsf{A}(i)} \sum_{j \neq \Delta} p_{ij}(a) y_j^{n-1}, \quad i \in \boldsymbol{S},$$

  for $n \geq 1$. Assume that there exist constants $c > 0$, $\gamma < 1$ such that $y_i^n \leq c\gamma^n$, $i \in \boldsymbol{S}$, $n = 0, \ldots$.

To see that indeed the expected time to reach state Δ is bounded, note that y_i^n is an upperbound for the probability that the process has not yet been absorbed in state Δ at time n. Writing $\tau_\Delta = \min\{n \mid X_n = \Delta\}$, it can be checked for any strategy $\boldsymbol{\sigma}$ that

$$\mathsf{P}_i^{\boldsymbol{\sigma}}\{\tau_\Delta > n\} \leq y_i^n \leq c\gamma^n, \tag{2.1.1}$$

and hence

$$\mathsf{E}_i^{\boldsymbol{\sigma}}\tau_\Delta = \sum_{n\geq 1} \mathsf{P}_i^{\boldsymbol{\sigma}}\{\tau_\Delta \geq n\} \leq \frac{c\gamma}{1-\gamma}. \tag{2.1.2}$$

Exercise 2.1 Prove (2.1.1) and (2.1.2).

Let next $r = \sup_{a\in\mathsf{A}(i),i\in S} |r_i(a)|$. We do not take into account terminal reward, i.e. we assume that $q_i = 0$ for all $i \in \boldsymbol{S}$. Let $\boldsymbol{\sigma} = (\sigma^0, \sigma^1, \ldots)$ be a strategy. Define

$$V^{\boldsymbol{\sigma}}(i) = \sum_{n=0}^{\infty} \mathsf{E}_i^{\boldsymbol{\sigma}} r_{X_n}(A_n).$$

Note that

$$V_N^{\boldsymbol{\sigma}}(i) = \sum_{n=0}^{N} \mathsf{E}_i^{\boldsymbol{\sigma}} r_{X_n}(A_n)$$

is the expected N-horizon reward, when strategy $(\sigma^0, \ldots, \sigma^{N-1})$ is employed and the terminal reward is $r_{X_N}(\sigma^N)$.

We are interested in determining the *(total reward) value function*

$$V^*(i) = \sup_{\boldsymbol{\sigma}} V^{\boldsymbol{\sigma}}(i), \tag{2.1.3}$$

and a strategy that attains this maximum reward, provided it exists. Note that $|V^{\boldsymbol{\sigma}}(i)| \leq r \cdot c \cdot \gamma/(1-\gamma)$ and so the $V^*(i)$ is well-defined.

2.2 Fixed stationary, deterministic strategy

Before focussing on the maximisation problem (2.1.3), we first concentrate on the expected rewards when using a fixed stationary, deterministic strategy $\boldsymbol{f} = (f, f, \ldots)$. As we shall see in Theorem 2.3.2 there exists $\boldsymbol{f}^*$ that attains the optimal rewards, i.e. $V^{\boldsymbol{f}^*}(i) = V^*(i)$.

Therefore, given $\boldsymbol{f}$ and a bounded function $v : \boldsymbol{S} \to \mathbf{R}$ (and $v(\Delta) = 0$), we define the mapping $\mathcal{T}^{\boldsymbol{f}}v$ by

$$(\mathcal{T}^{\boldsymbol{f}}v)(i) := r_i(f) + \sum_j p_{ij}(f)v(j), \quad i \in \boldsymbol{S}. \tag{2.2.1}$$

The n-th iterate of this mapping is inductively defined by

$$(\mathcal{T}_n^{\boldsymbol{f}}v)(i) := (\mathcal{T}^{\boldsymbol{f}}(\mathcal{T}_{n-1}^{\boldsymbol{f}}v))(i) = r_i(f) + \sum_j p_{ij}(f)(\mathcal{T}_{n-1}^{\boldsymbol{f}}v)(j), \quad i \in \boldsymbol{S}, \tag{2.2.2}$$

where $\mathcal{T}_1^{\boldsymbol{f}} = \mathcal{T}^{\boldsymbol{f}}$. For notational convenience we will leave out the brackets, i.e. we write $\mathcal{T}^{\boldsymbol{f}}v(i)$ instead of $(\mathcal{T}^{\boldsymbol{f}}v)(i)$. The connection with the expected N-horizon reward, when using decision rule f at each time point is given by

$$\mathcal{T}_N^{\boldsymbol{f}}v(i) = \sum_{n=0}^{N-1} \mathsf{E}_i^{\boldsymbol{f}} r_{X_n}(A_n) + \mathsf{E}_i^{\boldsymbol{f}} v(X_N),$$

with terminal reward v.

The mapping $\mathcal{T}^{\boldsymbol{f}}$ and its iterates have the following important properties.

Lemma 2.2.1 *If $u, v : \boldsymbol{S} \to \mathbf{R}$ are bounded functions, then*

i) *(Monotonicity) $u(i) \le v(i)$, $i \in \boldsymbol{S}$ implies $\mathcal{T}^{\boldsymbol{f}} u(i) \le \mathcal{T}^{\boldsymbol{f}} v(i)$, $i \in \boldsymbol{S}$;*

ii) *(Convergence) $\lim_{n\to\infty} \mathcal{T}_n^{\boldsymbol{f}} v(i) = V^{\boldsymbol{f}}(i)$, for all $i \in \boldsymbol{S}$;*

iii) *(Unique fix point) $v = V^{\boldsymbol{f}}$ is the unique solution to the functional equation $\mathcal{T}^{\boldsymbol{f}} v = v$.*

Proof. Discussed at lectures, or exercise.

Lemma 2.2.1 has a useful consequence. If strategy $\boldsymbol{f}'$ improves on strategy $\boldsymbol{f}$ *in one step*, then $\boldsymbol{f}'$ attains at least the same reward as $\boldsymbol{f}$ for all states, and his higher rewards for at least one state. In other words: $\boldsymbol{f}'$ stricly improves on $\boldsymbol{f}$. This will be used lateron to test a policy for optimality.

Corollary 2.2.2 *Suppose that $\boldsymbol{f}'$ and $\boldsymbol{f}$ are such that*

$$\mathcal{T}^{\boldsymbol{f}'} V^{\boldsymbol{f}} \ge V^{\boldsymbol{f}} + v,$$

for some bounded function $v \ge 0$, $v \not\equiv 0$. Then $V^{\boldsymbol{f}'} \ge V^{\boldsymbol{f}} + v$.

Proof. We claim that $\mathcal{T}_n^{\boldsymbol{f}'} V^{\boldsymbol{f}} \ge V^{\boldsymbol{f}} + v$. The proof is by induction. By assumption it holds for $n = 1$. Assume it is true for $n = 1, \ldots, N$. By Eqn. (2.2.2) for $n = N + 1$

$$\begin{aligned} \mathcal{T}_{N+1}^{\boldsymbol{f}'} V^{\boldsymbol{f}} &\ge \mathcal{T}^{\boldsymbol{f}'}(\mathcal{T}_N^{\boldsymbol{f}'} V^{\boldsymbol{f}}) \\ &\ge \mathcal{T}^{\boldsymbol{f}'}(V^{\boldsymbol{f}} + v) \\ &\ge V^{\boldsymbol{f}} + \mathcal{T}^{\boldsymbol{f}'} v + v \ge V^{\boldsymbol{f}} + v, \end{aligned}$$

where we have used Lemma 2.2.1(i) in the second inequality, and in the last inequality together with the non-negativity of v. Taking the limit $N \to \infty$ and again applying Lemma 2.2.1 (iii) yields the required assertion. QED

2.3 Optimality Equation

We next define the mapping $\mathcal{T}^*$ for any bounded function $v : \boldsymbol{S} \to \mathbf{R}$ (where again we leave out brackets):

$$\mathcal{T}^* v(i) := \max_{a \in \mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a) v(j)\}, \quad i \in \boldsymbol{S}. \tag{2.3.1}$$

The n-iterate is defined by

$$\mathcal{T}_n^* v(i) := \mathcal{T}^*(\mathcal{T}_{n-1}^* v)(i) = \max_{a \in \mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a) \mathcal{T}_{n-1}^* v(j)\}, \quad i \in \boldsymbol{S}.$$

This is precisely the dynamic programming equation for computing the n-horizon maximum reward, with terminal reward v.

We can derive an equivalent statement to Lemma 2.2.1.

Lemma 2.3.1 *Let $u, v : \boldsymbol{S} \to \mathbf{R}$ be bounded.*

i) *$u \leq v$ implies $\mathcal{T}^*u \leq \mathcal{T}^*v$.*

ii) *Further*

$$|\mathcal{T}_n^* v(i) - V^*(i)| \leq \frac{r\cdot c\cdot\gamma^n}{1-\gamma} + \sup_i |v(i)|\cdot c\cdot\gamma^n$$

Hence $\lim_{n\to\infty}\mathcal{T}_n^* v(i) = V^*(i)$.

Proof. We only prove the second assertion. Let $\boldsymbol{\sigma} = (\sigma^0, \sigma^1, \ldots)$ be a Markov strategy. For notational convenience, we let $V_n^{\boldsymbol{\sigma},v}$ denote the n-horizon cost under policy $(\sigma^0, \ldots, \sigma^{n-1})$ when the terminal reward equals v. Put $\phi = \sup_i |v(i)|$.

$$\begin{aligned} |V^{\boldsymbol{\sigma}}(i) - V_n^{\boldsymbol{\sigma},v}(i)| &= |\sum_{t=n-1}^{\infty} P(\sigma^0)\cdots P(\sigma^t)r(\sigma^n) - P(\sigma^0)\cdots P(\sigma^{n-1})v(i)| \\ &\leq \sum_{t\geq n-1} r\cdot c\cdot\gamma^{t+1} + \phi\cdot c\cdot\gamma^n = \frac{r\cdot c\cdot\gamma^n}{1-\gamma} + \phi\cdot c\cdot\gamma^n, \end{aligned} \tag{2.3.2}$$

which bound is independent of the strategy $\boldsymbol{\sigma}$. Write

$$\epsilon = \frac{r\cdot c\cdot\gamma^n}{1-\gamma} + \phi\cdot c\cdot\gamma^n.$$

Then by the above

$$V^{\boldsymbol{\sigma}}(i) \leq V_n^{\boldsymbol{\sigma},v}(i) + \epsilon \leq \mathcal{T}_n^* v(i) + \epsilon.$$

Therefore, also

$$V^*(i) \leq \mathcal{T}_n^* v(i) + \epsilon.$$

On the other hand, let $\boldsymbol{\sigma}^n = (\sigma^0, \ldots, \sigma^{n-1})$ be the optimal n-horizon strategy (for terminal reward v). and $\boldsymbol{\sigma}$ any strategy that uses $\boldsymbol{\sigma}^n$ the first n time units. Then

$$V_n^{\boldsymbol{\sigma},v}(i) = \mathcal{T}_n^* v(i). \tag{2.3.3}$$

Using (2.3.2) we obtain

$$V^*(i) \geq V^{\boldsymbol{\sigma}}(i) \geq \mathcal{T}_n^* v(i) - \epsilon.$$

This yields the desired result. The limit result follows directly. QED

The result states, that for any terminal reward, the n-horizon maximum reward converges to the infinite horizon maximum reward, geometrically quickly. This property will give rise to the so-called Successive Approximations algorithm, discussed below.

We first formulate the (total reward) optimality equation.

Theorem 2.3.2 *The function V^* is the unique solution to the optimality equation $\mathcal{T}^*v = v$, i.e.*

$$V^*(i) = \max_{a\in\mathsf{A}(i)}\{r_i(a) + \sum_j p_{ij}(a)V^*(j)\}, \quad i \in \boldsymbol{S}. \tag{2.3.4}$$

Any stationary, deterministic strategy $\boldsymbol{f} = (f, f, \ldots)$ satisfying

$$f(i) \in \arg\max_{a\in\mathsf{A}(i)}\{r_i(a) + \sum_j p_{ij}(a)V^*(j)\}, \quad i \in \boldsymbol{S}, \tag{2.3.5}$$

attains the maximum reward: $V^{\boldsymbol{f}} = V^*$.

Proof. One can use Lemma 2.3.1 to derive from the finite horizon optimality equation that V^* solves the optimality equation (2.3.4). For unicity, assume that there is another solution v. Suppose that $\boldsymbol{f}$ is the maximising policy in (2.3.4) and and that $\boldsymbol{g} = (g, g, \dots)$ the maximiser for v. Then

$$P(g)(V^* - v) \leq V^* - v \leq P(f)(V^* - v).$$

Iterate this by applying $P(f)$ to the right inequality and $P(g)$ to the left and repeat.

To show that the maximiser $\boldsymbol{f}$ is optimal, use Lemma 2.2.1 (iii). QED

We have already seen hints for two possible algorithms for computing the value function and an optimal (stationary, deterministic!) policy. These will be discussed next.

2.4 Algorithms for computing value function and optimal strategy

Policy Iteration Corollary 2.2.2 is the basis for the policy iteration (PI) algorithm. If we determine $\boldsymbol{f}' = (f', f', \dots)$ from $\boldsymbol{f} = (f, f, \dots)$ using

$$f'(i) \in \arg\max_{a \in \mathsf{A}(i)} \{r_a(i) + \sum_j p_{ij}(a) V^{\boldsymbol{f}}(j)\},$$

then either the conditions of Corollary 2.2.2 are satisfied, or $V^{\boldsymbol{f}}$ satisfies (2.3.4). In the first case $\boldsymbol{f}'$ strictly improves on $\boldsymbol{f}$, in the second case $\boldsymbol{f}$ is optimal by virtue of Theorem 2.3.2.

Policy Iteration Algorithm

0) Set $n := 0$. Choose any initial stationary, deterministic strategy $\boldsymbol{f}_0 = (f_0, \dots)$.

1) Compute $V^{\boldsymbol{f}_n}$ by solving $V^{\boldsymbol{f}_n} = \mathcal{T}^{\boldsymbol{f}_n} V^{\boldsymbol{f}_n} = r(f_n) + P(f_n) V^{\boldsymbol{f}_n}$. For small problems this can be done by inversion: $V^{\boldsymbol{f}_n} = (\mathbf{I} - P(f_n))^{-1} r(f_n)$.

2) Put $\boldsymbol{f} := \boldsymbol{f}_n$ and compute $\boldsymbol{f}_{n+1} = \boldsymbol{f}'$ from (2.3.4), taking $\boldsymbol{f} = \boldsymbol{f}'$ if possible.

3) If $\boldsymbol{f}_{n+1} = \boldsymbol{f}_n$ then this strategy is optimal. Stop.
Otherwise set $n := n + 1$, and go to step 1.

Step 1 requires solving a set of linear equations, which is infeasible if the state space is large. The number of iterations needed for convergence tends to be small. This method is therefore well suited for problems with a moderately sized state space.

Another suitable application that works well in practice is the following practically observed fact, that the first iteration of the PI gives the largest additional benefit. Hence, applying a one step improvement to a strategy that is expected to work reasonably well, or a strategy that is already employed, tends to yields the largest increase of performance. The actual application of such a one step improvement may not be easy to implement, because of the dimensionality of the problems from practice.

We did not discuss convergence of the PI algorithm yet.

Lemma 2.4.1 *The PI algorithm converges in the following sense:*

- $V^{\boldsymbol{f}_n} \uparrow V^*$, $n \to \infty$;

- *the sequence $\{f_n\}_n$ contains at least one limit point; each limit point defines an optimal stationary, deterministic strategy;*
- *if $\boldsymbol{S}$ is finite, PI converges in finitely many steps.*

Proof. By a Cantor diagonalisation argument, it can be shown that the sequence $\{f_n\}_n$ has at least one limit point. Here finiteness of the action spaces is crucial. By virtue of Corollary 2.2.2 $V^{\boldsymbol{f}_{n+1}} \geq V^{\boldsymbol{f}_n}$. It follows that $\{V^{\boldsymbol{f}_n}(i)\}_n$ is a non-decreasing bounded sequence, for each $i \in \boldsymbol{S}$. It therefore has a limit, $v(i)$ say, for each $i \in \boldsymbol{S}$.

Let f^* be any limit point, and let $\{f_{n_k}\}_k$ be a subsequence with limit point f^*. By construction

$$\mathcal{T}^{\boldsymbol{f}_{n+1}} V^{\boldsymbol{f}_n} \geq \mathcal{T}^{\boldsymbol{g}} V^{\boldsymbol{f}_n},$$

for any deterministic, stationary strategy $\boldsymbol{g}$, with equality for $g = f_{n+1}$. Fix initial state i. Then there exists an index K such that $f^{n_k}(i) = f^*(i)$ for $k \geq K$. Taking the limit $k \to \infty$ in the above (in)equality and using the dominated convergence theorem, it follows that

$$\mathcal{T}^{\boldsymbol{f}^*} v \geq T^{\boldsymbol{g}} v,$$

for any deterministic, stationary strategy $\boldsymbol{g}$, with equality for $g = f^*$. Hence v is a solution of the optimality equation (3.7). By virtue of Theorem 3.2.1 $v = V^*$ and f^* is optimal.

The fact that PI converges in finitely many steps in the case of a finite state space, stems from the fact that the number of stationary, deterministic policies is finite. Note that cycling cannot occur! QED

Successive Approximations Next we discuss the successive approximations (SA) algorithm, or alternatively, the Value Iteration algorithm. The computational complexity per iteration using this approach is less sensitive to the number of states than PI, but it may (and generally will) require a large number of iterations to get satisfactory results. Lemma 2.3.1(ii) provides the necessary ingredients to formulate the SA algorithm.

Successive Approximations Algorithm

0) Set $n := 0$. Choose any bounded (suitable) function $v_0 : \boldsymbol{S} \to \mathbf{R}$ (a common choice is $v_0 \equiv 0$). Choose $\epsilon > 0$.

1) Compute

$$V_{n+1}(i) = \max_{a \in \mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a) V_{n+1}(j)\},$$

and let

$$f_{n+1}(i) \in \arg\max_{a \in \mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a) V_{n+1}(j)\}.$$

2) Let $\mu_n = \sup_i |V_n(i) - V_{n-1}(i)|$. Stop, if

$$\frac{c\gamma\mu_n}{1-\gamma} < \epsilon.$$

Otherwise, set $n := n + 1$, goto step 1.

Lemma 2.3.1 (ii) ensures that $V_n(i) \to V^*(i)$, as $n \to \infty$, for all $i \in \boldsymbol{S}$. By Assumption 1.2.1 it is possible derive bounds, both for the difference $|V_n(i) - V^*(i)|$ as for the difference of $|V^{\boldsymbol{f}_n}(i) - V^*(i)|$. The latter is the difference in values between using strategy $\boldsymbol{f}_n$ from time n on, versus the optimal policy.

Theorem 2.4.2 *Let $V_n(i)$ be obtained from (3.4.3) and $f_n(i)$ from (3.4.4). Then*

- $$\mu_n = \sup_i |V_n(i) - V_{n-1}(i)| \leq c\gamma^{n-1} \cdot \sup_i |V_1(i) - V_0(i)|$$

Let further $c \cdot \gamma\mu_n/(1-\gamma) < \epsilon$.

- $\sup_i |V_n(i) - V^*(i)| \leq \epsilon$ *and*
- $$V_n(i) - \mu_n \frac{c\gamma}{1-\gamma} \leq V^{\boldsymbol{f}_n}(i) \leq V^*(i) \leq V_n(i) + \mu_n \frac{c\gamma}{1-\gamma},$$

  so that $0 \leq \sup_i (V^*(i) - V^{\boldsymbol{f}_n}(i)) \leq 2\epsilon$.
- *The sequence $\{f_n\}_n$ contains at least one limit point; each limit point defines an optimal stationary, deterministic strategy.*

Apart from practical purposes, the SA algorithm can be used in a theoretical sense, to determine structural properties of optimal strategies, e.g. control limit properties, etc. This is done, by choosing V_0 suitably, and by applying an iterative argument to the n-horizon solutions. An example of the application of this mechanism is provided next.

Example 2.4.1 (Quality control and repair) A firm manufactures a product under a continuing contract with university, that allows the university to cancel at any time without penalty. During any given day, the process for producing the product is either *in control* or *out of control*. Whether the process is out of control is not observed directly, but can only be inferred from the results of production. Each day, the firm produces one item, inspects it and classifies it as good or defective. The university accepts a good item and pays $r > 0$ for it. The form discards a defective item. When the process is out of control, each item is defective. Thus, if a good item is produced, the process was in control during its production.

When the process is in control, the (known) probability that the process produces a good item is p, $0 < p < 1$. The (known) probability that the process is in control at the time of production of an item given that it is in control at the time of production of its predecessor is q, $0 < q < 1$. Once the process is out of control, it remains so until it is repaired.

Independently of the production process, there is a probability β, $0 < \beta < 1$, that the firm will retain the contract for another day. The university informs the firm at the beginning of a day whether or not the contract is to be continued that day. If the decision is to cancel, the firm receives no further revenue from the university and inucrs no further cost. If the decision is to continue, there are two possibilities that day: immediately repair at cost $K > 0$ or don't repair. Repair is done quickly and brings the process into control with probability q (regardless whether or not it was in control at the time of repair). Repair is the only permissible option when S consecutive defectives have been produced since the latter of the times of the last repair and the last good item. The decision problem is to choose a repair policy that maximises the expected value of profits before the university cancels the contract.

Exercise 2.2 **a)** Let p_s be the conditional probability that item $s+1$ is good, given that items $1, \ldots, s$ were defective and either item 0 was good or the process was just repaired before producing item 1. Give a formula for p_s in terms of p, q, and s.

b) Model the decision problem as a Markov decision problem. Define states, actions and transitions probabilities (in terms of β, K, r and the p_s). Write down the corresponding optimality equation Does the problem satisfy Assumption 2.1.1? Explain.

c) Show that a *control-limit* policy is optimal, i.e. there is a number s^*, such that if the number of defective items since the latter of the last good item or repair is s, it is optimal to repair if $s \geq s^*$ and to repair otherwise. *Hint:* first show that p_s is decreasing in s. Next show by induction on N that the maximum expected profit V_N^* over N days is decreasing as a function of state, if we take terminal reward equal to 0. Then use successive approximations to show that the maximum expected profit V^* over an infinite horizon is decreasing as a function of state. Note: this iterative method is the most commonly used way to establish the form of the optimal policy in an infinite-horizon problem.

Before proceeding to discussing algorithms, we will discuss three classes of models that fit the framework exposed so far. Further, note that the additional absorbing state Δ does not play an active role in the optimisation, it has been included for theoretical completeness.

2.5 Maximise the entrance probability of a set

In many applications it is required that the probability of reaching some particular state i^* (within a given time limit or without time limit) is maximised. At first sight it may not be obvious that such a problem fits the framework described in these notes. In principle, after having entered state i^*, the process may again move to other states. Since the posterior evolution after having visited i^* does not affect the probability to reach it, we may as well require that in i^* the action set consists of element only, say $\{0\}$, and under this action the process is absorbed into the additional state Δ with probability 1.

There are several ways to model the direct rewards: one can set $r_{i,i^*}(a) = 1$ and then take expectations to calculate $r_i(a)$ as the probability of reaching i^* at the next step, if action a is selected. Or, put $r_i(0) = 1$.

An application of such a problem is the Investment example of Chapter 1. Instead of determining an optimal finite horizon strategy, one might determine the strategy that maximises the probability of taking home at least €40.000. This will be discussed at the lectures.

Another application is the following roulette problem. You may assume that it satisfies Assumption 2.1.1. This can be rigorously proven.

Roulette problem

An amateur gambler goes to the casino to play roulette with a budget of €20. In each round, he chooses to play either red or black. Therefore, in each round, the probability of doubling the bet is 18/37 and the probability of losing the bet is 19/37. Each round, the gambler places a bet with an integer amount of euros. The goal is to maximise the probability of taking home at least €50.

Exercise 2.3 **a)** First assume a finite planning horizon. Formulate this game as a Markov Decision problem, assuming a finite planning horizon T. (Specify the state space, action spaces, direct rewards, terminal reward and transition probabilities).

b) Formulate the infinite horizon optimality equation for the probability of ending up with at least €50.

c) Determine an optimal policy for $T = 2$ (and initial capital €20).

d) Write a computer program that computes the optimal first action and the corresponding maximum probabilities $\mathsf{P}\{X_T \geq 50\}$ for $50 \leq T \leq 100$, and initial capitals $i \in \{1, \ldots, 49\}$. Indicate in this table, from which value of T on the optimal action does not change anymore. Can you characterise the optimal action chosen as a function of state? Print the list of optimal first actions, and hand in a copy of the code.

e) Let f denote the decision rule assigning the optimal first actions from (d) to the states. Is $\boldsymbol{f}$ an optimal strategy? Verify and explain.

2.6 Optimal stopping

This section considers a class of controlled processes with the following characteristics:

- $\boldsymbol{S}$ is finite;
- in each state there are two possible decisions: s and c, where s stands for the stopping decision, and c for the continuation decision;
- if the controller selects decision c in state i, two things happen: cost $\gamma_i \geq 0$ has to be paid, and with probability p_{ij} the next state is j, $\sum_j p_{ij} = 1$;
- if the controller selects decision s in state i, a reward $r_i \geq 0$ is earned and the system stops.

Clearly, this model does not fit our requirements. First of all, never stopping is a valid decision. If $\gamma_i < 0$ for all i, this leads to the associated total expected cost being $-\infty$, which contradicts the conditions in Section 2.1.

However, there are states in which the controller will always stop. To see this, notice that $P = \{p_{ij}\}_{i,j \in S}$ is the transition matrix of a Markov chain on $\boldsymbol{S}$. This Markov chain has finitely many positive recurrent classes, ν say, plus possibly a finite set of transient states. Let $i_1, \ldots, i_\nu$ be the states earning the maximum stopping reward within their class. Then the controller will always stop in these states.

Transience and finiteness of the state space imply the validity of Assumption 2.1.1. Indeed, one can determine feasible constants c and γ as follows.

Computation of γ, c in Assumption 2.1.1

1. Compute the largest (positive) eigenvalue of the matrix P, call this ρ. Since P is transient, $\rho < 1$.
2. Select $\gamma \in (\rho, 1)$;

3. let $\mathbf{1} : \boldsymbol{S} \to \{1\}$ be the function indentically equal to 1; determine the solution $v : \boldsymbol{S} \to \mathbf{R}$ of the equation
$$v = \mathbf{1} + \gamma^{-1} P v.$$
Put $c = \max_i v_i$.

Let us formulate the Markov decision characteristics. This amounts to the following. We now put $\boldsymbol{S}_\Delta = \boldsymbol{S} \cap \{\Delta\}$, and we write $\mathcal{I} = \{i_1, \ldots, i_\nu, \Delta\}$:

- $\mathsf{A}(i) = \{s, c\}$, $i \notin \mathcal{I}$; $\mathsf{A}(i_l) = \{s\}$, $l = 1, \ldots, \nu$; $\mathsf{A}(\Delta) = \{0\}$;
- $p_{i\Delta}(s) = 1$, for all i; $p_{ij}(c) = p_{ij}$, $i \notin \mathcal{I}$, $j \neq \Delta$;
- $r_i(s) = r_i$, $i \neq \Delta$, $r_i(c) = -\gamma_i$, $i \notin \mathcal{I}$, $r_\Delta(0) = 0$.

Since it is optimal to stop in i_l, it follows that $V^*(i_l) = r_{i_l} \geq -\gamma_{i_l} + \sum_j p_{ij} V^*(j)$. For notational simplicity, we therefore need not exclude the continuation action in state i_l in the optimality equation, for $l = 1, \ldots, \nu$. Since state Δ has no contribution to the total expected reward, we will exclude it from the optimality equation (as we will do in the remainder of this chapter).

The optimality equation then becomes (restricting to $\boldsymbol{S}$)

$$V^*(i) = \max\{r_i, -\gamma_i + \sum_j p_{ij} V^*(j)\}. \tag{2.6.1}$$

Example 2.6.1 (Selling a house) Suppose someone would like to sell his house. Each day, a potential buyer might make an offer, to which the owner must react immediately. He can either accept or reject the offer (no bargaining is allowed). Each rejection of an offer implies a daily maintenance cost of C euros to the owner. A rejected offer is lost. The daily offer equals i euros with probability p_i, where $i \leq B$, independent of the offers on other days (p_0 is the probability that no offer is made). B stands for a reasonably upper bound on the potential offers. The goal is to determine an optimal acceptance strategy.

To this end we will model the problem as an optimal stopping problem. The state space is equal to the potential offers and the absorbing state Δ. We leave out Δ in our description. Then

- $\boldsymbol{S} = \{0, 1, \ldots, B\}$, state i corresponds to offer i;
- $\mathsf{A}(i) = \{s, c\}$, where s corresponds to accepting the present offer,
  and c to not accepting it;
- $r_i = i$; $\gamma_i = C$;
- $p_{ij} = p_j$.

The optimality equation becomes

$$V^*(i) = \max\{-C + \sum_j p_j V^*(j), i\}.$$

One can show that there is a control-limit optimal strategy i.e. there exists a threshold i^*, such that accepting offers $i \geq i^*$, and rejecting offers $i < i^*$ is optimal. The idea is to use PI, starting with the strategy $\boldsymbol{f}_0$ that always stops. Then $V^{\boldsymbol{f}_0}(i) = i$.

Use PI: this gives $\boldsymbol{f}_1$, with $f_1(i) = s$ iff $i \geq i_1$ and

$$i_1 = \min\{i \,|\, -C + \sum_j p_j V^{\boldsymbol{f}_0}(j) = -C + \sum_j p_j j \leq i\},$$

i.o.w. $\boldsymbol{f}_1$ is a control-limit strategy. Clearly, $i_1 \geq i_0$. By Corollary 2.2.2(ii) $V^{\boldsymbol{f}_1}(i) \geq V^{\boldsymbol{f}_0}(i) = i$, $i \leq B$.

Let us look at the $(n+1)$-th iteration. Strategy $\boldsymbol{f}_{n+1}$, is determined by from

$$\boldsymbol{f}_{n+1}(i) = \arg\max\{-C + \sum_j p_j V^{\boldsymbol{f}_n}(j), i\}.$$

The first term is a constant, and hence $\boldsymbol{f}_{n+1}$ is again a control-limit strategy with threshold i_{n+1}. By PI

$$-C + \sum_j p_j V^{\boldsymbol{f}_n}(j) \geq -C + \sum_j V^{\boldsymbol{f}_{n-1}}(j),$$

we infer that $i_{n+1} \geq i_n$.

Since the state space is finite, there exists n such that $\boldsymbol{f}_n$ is optimal.

Under extra conditions one can show that the optimal strategy always has a *control-limit* type of structure. To this end, denote $S^* = \{i \in \boldsymbol{S} \,|\, r_i \geq -\gamma_i + \sum_j p_{ij} r_j\}$, i.o.w. S^* is the collection of states in which stopping is as at least as good as stopping at the next time instant. This is called a *1-step look ahead strategy.*

Definition 2.6.1 The stopping problem is called *monotonic*, if the process cannot leave set S^* once it has entered it. More specificly: $i \in S^*$, $p_{ij} > 0$ implies $j \in S^*$.

Note that the House selling example 2.6.1 is *not monotonic*!

Theorem 2.6.2 *In a monotonic stopping problem, the strategy* $\boldsymbol{f}$, *given by*

$$f(i) = \begin{cases} c, & i \notin S^* \\ s, & i \in S^* \end{cases}$$

is optimal.

For the proof, see Exercise 2.9.

Example 2.6.2 Best choice This is a generic class of problems. For instance the famous Secretary problem is of this type, or the following medical treatment problem.

Suppose a doctor has a waiting list of 100 patients to administer a special treatment to. However, treatments never are guaranteed to be succesful, this will be the case only for a known fraction $p \in (0, 1)$ of the patients. On the one hand the doctor wants to minimise the negative effect of failed treatments, on the other hand he would like to treat all patients for which the treatment is succesful. He therefore would like to stop treating patients, after the last succesful treatment. However, he is not omniscient. His goal is then to treat all patients, and stop after the succesful treatment that has the maximum probability of its being the *last* succesful treatment. How can he accomplish this goal?

The generic problem is as follows. Given are N potential strategic decisions, that have to be sequentially taken by a controller. Strategic decision i has value $X_i = 1$ if it turns out to be succesful, but value $X_i = 0$ if it is failure. The probability of succes is $p_i \in (0, 1)$. These sequential outcomes are mutually independent, and success probabilities are assumed known.

After the outcome of a each strategic decision is known, the controller can decide to stop or to continue to take the next strategic decision. He desires to know the strategy that maximises the probability to stop after the last succesful one.

The aim is to model this as a monotone stopping problem. Due to the criterion, the controller will never stop after a failure. So, stopping or continuation decisions are moments that the last strategic decision was succesful. This motivates the following model, where we only list the problem specific parameters.

- $\boldsymbol{S} = \{1, \ldots, N\}$, with the interpretation that state i corresponds to strategic decision i being the last succesful decision so far. N.B. if no succesful strategic decision ever occurs, the controller simply takes all of them.
- $p_{ij} = \prod_{k=i+1}^{j-1}(1 - p_k)p_j$ is the probability, that strategic decision j is the first one after strategic decision i to be succesful.
- $\gamma_i = 0$, $r_i = \mathsf{P}\{i \text{ is the last succesful decision, given it was succesful}\} = \prod_{k>i}(1 - p_k)$.

Then $S^* = \{i \,|\, r_i \geq \sum_j p_{ij} r_j\} = \{i \,|\, 1 \geq \sum_{k>i} \frac{p_k}{1-p_k}\}$, which is clearly monotonic (S^* might be empty, but we will not consider that case). One can describe it in threshold form $S^* = \{i \,|\, i \geq i^*\}$, where $i^* = \min\{i \,|\, \sum_{k>i} \frac{p_k}{1-p_k} \leq 1\}$.

Then one can also compute the value function. We give the result.

$$V^*(i) = \begin{cases} r_i = \prod_{k>i}(1 - p_k) & i \geq i^* \\ \sum_{k \geq i^*} \frac{p_k}{1 - p_k} \prod_{l \geq i^*}(1 - p_l), & i < i^*. \end{cases}$$

The quantity $V^*(1)$ is the desired maximum probability!

2.7 Discounting future rewards

Another class of problems is where rewards are discounted. This can be motivated as follows.

Suppose that we have a choice to earn €10 now, or in one year. Since we may put the money on the bank and get interest rate ρ say, in one year time the €10 that we received now, will have increased (well... in times of higher interest rates than at present...) to €$(1 + \rho) \cdot 10$. So it is preferred to earn the €10 now rather than in one year. Further, earning €10, will be worth to us €$10/(1 + \rho) = 10 \cdot \alpha$ now, where $\alpha = 1/(1 + \rho) \in (0, 1)$.

α is called the discount rate (per unit time). So earning $r_{X_n}(A_n)$ at time n, is worth $\alpha^n r_{X_n}(A_n)$ to us now. This gives rise to the following reward criterion.

Definition 2.7.1

$$V_\alpha^{\boldsymbol{\sigma}} = \sum_{n=0}^{\infty} \alpha^n \mathsf{E}^{\boldsymbol{\sigma}} r_{X_n}(A_n)$$

is the *expected discounted reward* under strategy $\boldsymbol{\sigma}$. The α-*discounted value function* is

$$V_\alpha = \sup_{\boldsymbol{\sigma}} V_\alpha^{\boldsymbol{\sigma}}$$

is the maximum expected discounted reward.

How does this fit within the framework of this chapter?

Suppose $\boldsymbol{\sigma} = (\sigma^0, \sigma^1, \ldots)$ is a Markov policy. Then

$$V_\alpha^{\boldsymbol{\sigma}} = r(\sigma^0) + \sum_{n=1}^{\infty} \alpha^n P(\sigma^0) \cdots P(\sigma^{n-1}) r(\sigma^n).$$

One can 'take α inside the probabilities' by interpreting the discounted reward under $\boldsymbol{\sigma}$ as follows:

1. at time 0 you recieve $r(\sigma^0)$;
2. then you throw a two-sided biased coin:
   - with probability $1-\alpha$ 'head' comes up and you stop (go to Δ);
   - with probability α 'tails' comes up and you continue. Then you select the next state according to rule σ^1 and you receive $r_{X_1}(\sigma^1)$.
3. Repeat this process indefinitely.

In this case the optimality equation becomes

$$V_\alpha(i) = \max\{r_i(a) + \alpha \sum_j p_{ij}(a) V_\alpha(j)\}, \quad i \in \boldsymbol{S}. \tag{2.7.1}$$

This implies in fact that *Assumption 2.1.1 is satisfied with $c = 1$ and $\gamma = \alpha$!* We will not further explicitly use Δ!.

To motivate an interesting type of problems, consider the next problem.

Exercise 2.4 (A simple bandit model) A decision maker observes a discrete-time system which moves between states $\{1, 2, 3, 4\}$ according to the following transition probability matrix

$$P = \begin{pmatrix} 0.3 & 0.4 & 0.2 & 0.1 \\ 0.2 & 0.3 & 0.5 & 0 \\ 0.1 & 0 & 0.8 & 0.1 \\ 0.4 & 0 & 0 & 0.6 \end{pmatrix}$$

At each point of time, the decision maker may leave the system and receive a reward of $R = 20$ units, or alternatively remain in the system and receive a reward of $r(i)$ units, if the system state is i. If the decision maker decided to remain in the system, his state at the next time instant is determined by P. Assume a discount rate of $\alpha = 0.9$ and that $r(i) = i$.

i) Formulate the model as an MDP.

ii) Use policy iteration to find a stationary policy that maximised the expected total discounted reward.

iii) Find the smallest value of R so that it is optimal to leave in state 2.

iv) Show for arbitrary P and $r(\cdot)$, and state space $\boldsymbol{S} = \{1, \ldots, N\}$, $N < \infty$ given, that: for each $i \in \boldsymbol{S}$ there exists a number R_i such that it is optimal to leave the system in state i only if $R \geq R_i$. Hint: show that $V_\alpha(i, R) - R$ is decreasing in R, where $V_\alpha(\cdot, R)$ is the value function of the problem with terminal reward R.

The index R_i in (iv) is related to the so-called Gittins index, and the problem can be viewed as a so-called bandit problem. This comes from the world of gambling.

A gambler has the choice to play different slot machines (sometimes known as ”one-armed bandits”), how often to play a selected machine and in which order to play each of them. Each machine provides a random reward from a distribution specific to that machine, when it is selected to be played. The objective of the gambler is to maximize the sum of the discounted rewards earned through.

Bandit processes This gives rise to the notion of a bandit process modelling one slot machine, or, a one-armed bandit. A bandit process is a Markov decision process where there can be just two actions: continue c or freeze ϕ. If action ϕ is chosen, and the state is i, the state remains the same ($p_{ii}(\phi) = 1$), and no reward is paid ($r_i(\phi) = 0$). If action c is chosen, there is a reward $r_i(c)$ when the state of the bandit process is i and with probability $p_{ij}(c)$ the new state is j.

A two-armed bandit process is a collection of two such bandit processes. The process in Exercise 2.4 is a special case of such a process. Bandit 1 has one state only that always pays a fixed amount, when not frozen. Bandit 2 has the transition mechanism described above under the continuation action.

We will study this. Let us define the following special two-armed bandit process.

- Bandit 1 has a state space consisting of one state only, say $\{0\}$. Under the continuation action for bandit 1, it will pay a reward λ.
- Bandit 2 has a non-trivial state space $\boldsymbol{S}$. The transition probabilities and rewards have the general form described above.
- The relation between the two bandit processes, is that the controller can either either freeze bandit 1 and continue on 2 or vice versa, so as to maximise the expected discounted reward.

The state space $\{0\} \times \boldsymbol{S}$, which can be identified with $\boldsymbol{S}$. By the structure of the actions, the action of bandit 2 completely defines the strategy. The discount optimality equation is then given by:

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

After some reflection (see Exercise 2.5), this is equivalent to

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

The left-hand choice corresponds to always continuing with bandit 1. The right-hand choice pulls bandit 2 for at least once, till it switches to 1. Once the optimal strategy switches to 1, it will never change back (why?).

Fix a state $i \in \boldsymbol{S}$. Suppose that we now vary the reward λ from bandit 1. In Exercise 2.4 (iv) it has been shown that there is a unique value λ^* for which both terms within the max are equal, in the equation for state i (note that λ corresponds to $R(1-\alpha)$ in Exercise 2.4). We put $G(i) = \lambda^*$. According to that same exercise *the optimal strategy is then to select bandit 2 in state i only if $\lambda < G(i)$, otherwise select bandit 1 forever after.*

We will derive other characterisations of the Gittins index. By the fact that the optimal policy never switches back from bandit 1 to bandit 2, there is a random time τ specifying the instant that bandit 1 is pulled for the first time, never to switch back again.

The problem then reduces to determine the switching time optimally, so as to maximise the discounted rewards. I.o.w. the discount optimality equation reduces to

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

After some algebra, we find that

$$G(i) = \sup_{\tau>0} \frac{\mathsf{E}_i \sum_{n=0}^{\tau-1} \alpha^n r_{X_n}(c)}{\mathsf{E}_i \sum_{n=0}^{\tau-1} \alpha^n}.$$

This implies that $G(i)$ can be computed *independently of the other bandit!* This in turn allows to characterise an optimal strategy also in the case that the bandit 1 process has the general form.

For the computation of the Gittins index for a finite state space, see the (corrected) paper 2013-bandit-computations-annotated.pdf on the MDP-website.

Example 2.7.1 (Testing a new medicine) Efficient testing of a new medicine can be solved by modelling it as a two-armed bandit problem of the type described above. Suppose a new medicine has been developed for a given disease, say medicine A. There already exists a suitable medicine, medicine B, of which the probability of success is known to be $p \in (0,1)$.

The cure probability of A is not known. However, if the doctor would administer it to 100 patients, then a good estimate of that probablity would be the fraction of the patients that were cured by it. If that fraction turns out to be low compared to the cure probability of B, too many patients will have suffered unnecessary damage. It is therefore desirable to develop a safer method. The idea is to decide for each subsequent patient whether to administer the new medicine or not on the basis of the known results so far. The objective is to maximise the discounted expected number of cured patients.

The relevant information to base this decision on, is the number of cured (x) and non-cured (y) patients that took the new medicine. The information on the effect of the medicine B has already been known, does not yield any extra information, and so we do not need to keep track of that. This defines the following Markov decision process.

- $\boldsymbol{S} = \{(x,y)\,|, x,y \in \mathbf{Z}_+\}$, where x is the number of cured patients by A, and y the number of non-cured ones.
- For each subsequent patient the doctor has the choice to administer A (action 0) or B (action 1). Hence $\mathsf{A}(x,y) = \{0,1\}$ for each (x,y).
- Suppose the present state is (x,y). Then a good choice for the transition probabilities will be (this will be explained in the second half of the lectures)

  $$p_{(x,y),(x',y')}(0) = \begin{cases} \frac{x+1}{x+y+2}, & x' = x+1, y' = y \\ \frac{y+1}{x+y+2}, & x' = x, y' = y+1. \end{cases}$$

  The reward of a cured patient is 1, i.e. $r_{(x,y)(x+1,y)}(0) = 1$ and $r_{(x,y)(x,y+1)}(0)$. Hence $r_{(x,y)}(0) = (x+1)/(x+y+2)$. Under action 1 the state does not change, however $r_{(x,y)}(1) = p$.

Exercise 2.5 a) Explain how (2.7.3) and (2.7.4) follow from (2.7.2). Why does the optimal strategy continue forever with bandit 1, once it has been selected?

b) Formulate the optimality equation for the medicine problem.

c) Suppose that the number of patients is 50, and suppose that $\alpha = 0,9$ and $p = 0,4$. Compute the Gittins index for 10 states (x,y) with $x/(x+y) \approx 0,4$ (see the url above). Compare with the estimated cure probability $(x+1)/(x+y+2)$ of A. Hand in a copy of your code.

2.8 Linear programming

will be incorporated next time MDP will be taught.

2.9 Dealing with unbounded rewards

In Section 1.3.1, we have discussed a transformation trick allowing to translate the unbounded reward case to the bounded reward case. Also in the discounted reward case this trick is applicable.

Let the discount factor α be given. Assume that there exists a constant $c < 1/\alpha$ such that (1.3.5) holds for all $a \in \mathcal{A}(i)$ and $i \in \boldsymbol{S}$.

Exercise 2.6 a) Determine an appropriate discount rate β, such that (using the notation in Section 1.3.1)

$$\tilde{V}_\beta^{\boldsymbol{\sigma}}(i) = M(i) V_\alpha^{\boldsymbol{\sigma}}(i), \quad i \in \mathcal{I}, \text{ and all Markov policies } s.$$

b) Show that the statements in Lemma 2.3.1 and Theorem 2.3.2 apply to the unbounded reward model satisfying the assumptions above.

c) Deduce the appropriate bounds in Theorem 2.4.2.

d) Construct the function M in the case of the single server queueing system below.

For the applicability of the result it is important that the conditions are verifiable. Below we give a model where this is indeed the case.

As a consequence of the transformation, the convergence of the Successive Approximation and Policy Iteration algorithms is guaranteed.

Arrival control in a single server queueing system We consider a time-discretised single server queueing system. X_t denotes the number of customers at time t in the system. In the next time-slot (i.e. between t and $t+1$) there is a customer departure with probability $\mu < 1$, if $X_t > 0$, due to a service completion. In the same time-slot a new customer arrives with probability $\lambda = 1 - \mu$.

At time t the system controller decides whether or not to accept a potential arrival, and he earns a reward K if he does. His decision is based on the present system state X_t. On the other hand, he has to pay $c \cdot X_t$, as a penalty for having a long queue. The question is: what strategy minimises the total discounted expected reward? There is a trade-off between the gain obtained by accepting customers versus the penalty of having many customers waiting in the queue to be served.

We can model this as an Markov Decision Process, with $\boldsymbol{S} = \mathbf{Z}_+$, $\mathsf{A}(i) = \{1, 2\}$, where 1 stands for the acceptance decision and 2 for the rejection decision, $i \in \boldsymbol{S}$. The transition probabilities and reward are given by

$$p_{ij}(a) = \begin{cases} \lambda, & a = 2, j = i+1, \text{ or } a = 1, j = i \\ \mu, & i > 0, j = i-1 \text{ or } i = 0, j = 0 \end{cases} \quad \text{and} \quad r_i(a) = \begin{cases} K - ci, & a = 1 \\ -ci & a = 2 \end{cases}$$

This is clearly an MDP with unbounded rewards! In Exercise 2.6 we have asked you to construct a suitable bounding vector M, guaranteeing that a discount optimal strategy exists and can be determined by one of the algorithms that have been discussed. In the next exercise you may assume that such a vector exists.

Exercise 2.7 i) Formulate the optimality equation for the maximum expected discounted rewards.

Intuitively it is clear that an optimal strategy that rejects potentially arriving customers in the next time-slot, when the system state is i, will also reject when the system state is larger than i, i.o.w. it is a control-limit strategy (see also the house selling example). Denote by $\boldsymbol{f}_i$ the threshold strategy that accepts whenever the system state is at most i and rejects in all system states larger than i.

One may prove threshold optimality by Successive Approximations, using an induction argument. Let $v_0(i) = 0$, $i \in \boldsymbol{S}$.

ii) Show that

- **a)** $f_{n+1}(i) = 2$ if and only if $v_n(i+1) - v_n(i) \leq -\frac{K}{\alpha \cdot \lambda}$;
- **b)** f_{n+1} is threshold if v_n is concave, i.e. $v_n(i+2) - v_n(i+1) \leq v_n(i+1) - v_n(i)$, $i \in \boldsymbol{S}$.
- **c)** Show by induction that v_n concave implies that v_{n+1} is concave.
- **d)** Conclude that there exists a threshold optimal strategy.

Exercise 2.8 (Continuation) i) Show that $v_n(i+1) - v_n(i) \le v_{n-1}(i+1) - v_{n-1}(i)$ for all $i \in \boldsymbol{S}$ implies $v_{n+1}(i+1) - v_{n+1}(i) \le v_n(i+1) - v_n(i)$ for all $i \in \mathcal{I}$. Show the same implication with $\ge$-signs.

Hence, if

$$v_1(i+1) - v_1(i) \le (\ge) v_0(i+1) - v_0(i), i \in \boldsymbol{S}, \tag{2.9.1}$$

then $v_{n+1}(i+1) - v_{n+1}(i) \le (\ge) v_n(i+1) - v_n(i)$ for all n, $i \in \boldsymbol{S}$. It follows that $\{\boldsymbol{f}_n\}_n$ form a non-increasing (non-decreasing) sequence of threshold strategies.

ii) Explain this.

iii) Choose $\lambda = 1/4 = 1 - \mu$, $\alpha = 0.9$, $c = 10$ and $K = 4$. Determine two initial vectors v_0, one such that (2.9.1) holds with $\le$, and one such that (2.9.1) holds with $\ge$. Compute upper- and lower bounds for the optimal threshold.

2.10 Additional exercises

Exercise 2.9 a) Prove the correctness of the procedure in Section 2.6 described to compute the constants γ and c in Assumption 2.1.1 for Stopping problems.

b) Use a successive approximations argument to prove Theorem 2.6.2.

Exercise 2.10 (Finding a lost object) A favourite but very small object has been lost, but you do not know precisely where. There are various possibilities for the place where this could have occurred. You will start to investigate all these possible places. This will take time. The longer you search a given spot, the less likely it is that the object is actually there. How long should you search?

An abstraction of this problem is as follows. Given is a space, where an object that we wish to find, is located with (known) probability p. Each investigation of the space costs c (this might reflect time). If the object is in the space, it will be found with probability β each time the space is investigated. A maximum of N investigations is allowed. If the object is found, investigation of the space is immediately stopped.

Clearly, the more times the space is investigated, the less the probability that the object is actually there. The problem can be modelled as an optimal stopping problem. The aim is find the best balance between the cost of investigating the space and the probability that the object is found.

a) Let p_i be the probability that the object is in the space, given i unsuccesful investigations. Show that

$$p_i = \frac{p(1-\beta)^i}{p(1-\beta)^i + (1-p)}.$$

b) Formulate as a monotonic optimal stopping problem, by specifying state space, reward, cost, and transition probabilities. Justify your choice. The expected total reward under an arbitrary strategy should reflect the cost of investigations versus the probability of finding the object.

c) What is the optimal strategy if the object is in the space with probability $p = 1$?

d) Suppose that the object has value r. The higher the value, the more investigations will be performed. Reformulate the monotonic optimal stopping problem.

Exercise 2.11 (Monitor a production line for a change in quality) A production is producing high quality products. The quality of products is not constant over time but subject to random fluctuations, which have a known distribution. However, the random fluctuations might be due to the machine starting suddenly to malfunction, so that the quality of products will deteriorate. The goal is to detect this change of production quality as quick as possible.

This problem amounts to detecting a change in distribution. Let be given a sequence of random variables $X_1, X_2, \ldots$ with values in a countable space $\mathcal{I}$, and with known distribution $p = \{p_k\}_{k\in\mathcal{I}}$. At some unknown point in time τ, the distribution changes to another known distribution, say $q = \{q_k\}_{k\in\mathcal{I}}$ Assume that the distribution of τ is known as well.

It is important to detect this change as quick as possible. If your check at time n whether a change has occurred falsely (the change has not occurred yet), then this costs $c > 0$. If the change has already occurred then this costs $(n - \tau)$. The total cost at time n will therefore be $C_n = \mathbf{1}_{\{\tau>n\}}c + \mathbf{1}_{\{\tau\leq n\}}(n-\tau)^+$.

Unfortunately τ is not observable. Since the information available consists of all realisations of the random variables so far, instead of C_n, we consider information dependent expected cost $c_n(x_1, \ldots, x_n) = \mathsf{E}(C_n \,|\, X_i = x_i, i = 1, \ldots, n\}$. We wish to find the check-up strategy with minimum expected cost. Assume a large, but finite horizon T. Further assume that given $\tau = n$, $X_1, \ldots$ are independent, with $X_1, \ldots, X_{n-1}$ having distribution p and $X_n, \ldots$ having distribution q.

We can then model the optimisation problem as a finite horizon optimal stopping problem. The question is under what conditions it will be a monotonic stopping problem.

a) Formulate the optimal stopping problem, i.e. specify state space, reward, cost and transition probabilities. Formulate the optimality equation.

b) In order that the optimal stopping problem be monotonic, we have to consider the equivalent of the generic inequality $r_i \leq -\gamma_i + \sum_j p_{ij} r_j$ for each state i. Given that $X_i = x_i$, $i = 1, \ldots, n$, show that this inequality reduces to $U(x_1, \ldots, x_n) \geq c$, where

$$U(x_1, \ldots, x_n) = \frac{\mathsf{P}\{\tau \leq n \,|\, X_i = x_i, i = 1, \ldots, n\}}{\mathsf{P}\{\tau = n+1 \,|\, X_i = x_i, i = 1, \ldots, n\}}.$$

c) Suppose that τ has a geometric distribution, i.e. $\mathsf{P}\{\tau = n\} = \phi^n/(1-\phi)$, $\phi \in (0, 1)$. Determine a recurrence relation for U. I.o.w. express $U(x_1, \ldots, x_n)$ in terms of $U(x_1, \ldots, x_{n-1})$. Determine $U(x_1)$.

d) Next suppose that p and q are geometric as well, with parameters ρ and γ respectively. Find sufficient conditions on ρ and γ so that the stopping problem is monotonic.

Exercise 2.12 (Dynamic location model) A repairman services $Q = 4$ work sites, with site 1 denoting the home office, and sites 2,3 and 4 denoting remote sites. He moves from site i to site j in one time-slot with probability p_{ij}, the actual numbers are given in the following matrix

$$P = \begin{pmatrix} 0.1 & 0.3 & 0.3 & 0.3 \\ 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0.8 & 0.2 \\ 0.4 & 0 & 0 & 0.6 \end{pmatrix}.$$

An equipment trailer carrying spare parts and tools may be located at any one of the 4 sites. If the trailer is at site j and the repairman at site i then the cost of obtaining material from the trailer is $c(i, j)$. In particular $c(i, j) = 100$ if the repairman and the trailer are at different remotes sites $i, j > 1$, $i \neq j$; $c(i, j) = 50$ if they are at the same remote site $i = j > 1$; $c(i, j) = 200$ if the repairman is at a remote site $i > 1$, but the trailer is at the home office $j = 1$. If the repairman is at the home office, no work is carried out, and so the cost of using the trailer can be taken equal to 0.

Additionally, there are costs to relocate the trailer: $d(i, j)$ from site i to site j. Here $d(i, j) = 300$ for $i \neq j$.

Assume discount rate $\alpha = 0.95$. The objective is to determine the strategy with minimum total expected discounted cost. It is assumed that the decision maker observes the location of the trailer and the repairman, he relocates the trailer and then the repairman moves to a site to do a repair.

a) Formulate this model as an MDP.

b) Find a relocation strategy that minimises the total expected discounted cost.

c) Describe the structure of the optimal policy.

Chapter 3

Infinite horizon: average expected rewards

Another way of dealing with an infinite planning horizon is to maximise the *expected average reward*. For a fixed strategy σ we define the expected average reward, after starting in state i, by

$$g^{\sigma}(i)=\limsup_{T\to\infty}\frac{V_T^{\sigma}(i)}{T}=\limsup_{T\to\infty}\frac{1}{T}\sum_{n=0}^{T-1}\mathsf{E}_i^{\sigma}r_{X_n}(A_n). \tag{3.0.1}$$

(Take terminal reward $q\equiv 0$.) In (3.0.1) we take the lim sup to avoid technicalities regarding the existence of the limit. Further define

$$g^*(i)=\sup_{\sigma}g^{\sigma}(i),\quad i\in\boldsymbol{S}, \tag{3.0.2}$$

and σ is called an average optimal strategy, if $g^*\sigma=g^*$.

We are interested in finding a strategy (provided it exists) that attains this maximum (expected) average reward. As we shall see, for a broad class of models not only such a strategy exists, but there even is an *optimal deterministic stationary strategy*, just like in the expected total reward model from Chapter 2. Very often the expected average reward does not depend on the initial state and we simply have $g^*(i)\equiv g^*$, $i\in\boldsymbol{S}$.

However, contrary to the finite and infinite horizon problems discussed in the previous chapter, an average optimal policy may not exist.

Example 3.0.1 Let $\boldsymbol{S}=\{1,2,\ldots\}\cup\{1',2',\ldots\}$. Fix $i\in\{1,2,\ldots\}$. The action spaces are given by $\mathsf{A}(i)=\{1,2\}$ and $\mathsf{A}(i')=\{1\}$. The transition probabilities are $p_{i\,i+1}(1)=1=p_{i\,i'}(2)=p_{i'\,i'}(1)$. The direct rewards are $r_i(1)=r_i(2)=-1$, $i=1,2,\ldots$ and $r_{i'}(1)=-1/i$, $i'=1',2',\ldots$.

Define the stationary, deterministic strategy $\boldsymbol{f}^n=(f^n,f^n,\ldots)$ by

$$f^n(i)=\begin{cases}1, & i<n\\ 2, & i\geq n.\end{cases}$$

Then $g^{\boldsymbol{f}^n}(1)=-1/n$, and $\sup_{\sigma}g^{\sigma}(1)=0$. However, the value 0 is not attained by any stationary strategy.

Notice that $g^{\boldsymbol{f}^n}$ is not a constant vector!

What happens for the α-discounted rewards? It is not difficult to show that $\boldsymbol{f}^\alpha$, with

$$f^\alpha(i) = \begin{cases} 1, & i < \frac{1}{\sqrt{1-\alpha}} \\ 2, & i > \frac{1}{\sqrt{1-\alpha}} \end{cases}$$

is optimal. As we shall see later, $\lim_{\alpha\to 0} f^\alpha$ defines an average optimal strategy under certain conditions. However, in this example $\lim_{\alpha\to 0} f^\alpha = f$ with $f(i) = 1$ for $i \in \boldsymbol{S}$, which is clearly very non-optimal!

This example shows the expected average rewards to exhibit weird phenomena. In the next section we will recall basic properties of (stationary) Markov chains that have an important implication for the (expected) average reward of stationary strategies.

3.1 Average reward using strationary strategies

What can one say about the average reward when using the stationary, deterministic strategy $\boldsymbol{f} = (f, f, \ldots)$? Stationarity of the strategy implies that $\{X_n\}_{n=0,1,\ldots}$ is a Markov chain with transition probabilities

$$p_{ij}(f) = \mathsf{P}\{X_{n+1} = j \mid X_n = i\} = \mathsf{P}^{\boldsymbol{f}}\{X_{n+1} = j \mid X_n = i, A_n = f(i)\},$$

which are the elements of the transition matrix $P(f)$. Similarly, for the n-step transition probabilities we get

$$p_{ij}^{(n)}(f) = \mathsf{P}^{\boldsymbol{f}}\{X_n = j \mid X_0 = i\}.$$

By definition $p_{ij}^{(1)}(f) = p_{ij}(f)$. By conditioning on the state after one step, we have for $n = 2, 3, \ldots$

$$p_{ij}^{(n)}(f) = \sum_k p_{ik}(f) p_{kj}^{(n-1)}(f).$$

These equations are known as the Chapman-Kolmogorov equations.

Let $\tau_{i_0} = \min\{n \geq 1 \mid X_n = i_0\}$, and write $T_{i\,i_0}(f) = \mathsf{E}_i^{\boldsymbol{f}} \tau_{i_0}$ for the expected time to reach the state i_0, given that the Markov chain starts at i (provided it is finite). The first assumption that we make is that this is finite, i.o.w.,

$$T_{i\,i_0}(f) = \mathsf{E}_i^{\boldsymbol{f}} \sum_{n=0}^{\tau_{i_0}-1} 1 < \infty, \quad \forall i \in \boldsymbol{S}. \tag{3.1.1}$$

We further define $R_{i\,i_0}(f)$ to stand for the total expected reward till incurred before entering state i_0, given the Markov chain starts in i (provided that it is finite)

$$R_{i\,i_0}(f) = \mathsf{E}_i^{\boldsymbol{f}} \sum_{n=0}^{\tau_{i_0}-1} r_{X_n}(f).$$

The second assumption that we make is that the above summation is absolutely convergent, i.o.w.

$$\mathsf{E}_i^{\boldsymbol{f}} \sum_{n=0}^{\tau_{i_0}-1} |r_{X_n}(f)| < \infty. \tag{3.1.2}$$

Then the following properties hold.

First of all: state i_0 can be reached from any other state. This implies that $\{X_n\}_n$ has at most one closed class. We can now apply Renewal Theory[1] to obtain the following results:

- The closed class is positive recurrent, and absorption into it takes place in finite expected time and with finite expected reward. Hence it has a stationary distribution, $\pi(f)$ say, which is a unique solution of the system
$$\left.\begin{array}{rcl} x_i &=& \sum_j x_j p_{ji}(f), \quad i \in \boldsymbol{S} \\ \sum_i x_i &=& 1. \end{array}\right\} \tag{3.1.3}$$
- For all $i \in \boldsymbol{S}$
$$\frac{V_T^{\boldsymbol{f}}(i)}{T} \to \sum_j \pi_i(f) r_j(f(j)) = g_i(f),$$
in particular $g_i^f = g^f$ is a constant.
- $g^f = R_{i_0\, i_0}(f)/T_{i_0\, i_0}(f)$.

The main question now is: how to compute the maximum expected average reward, if this is a constant, as in the above case? To set up an equation for a constant does not help. To get an intuition, we will use the α-discount equation for $V_\alpha(f)$, as well as the following result.

Abel and Cesaro limit Let $\{x_n\}_n$ be a bounded sequence on numbers or a non-negative sequence of numbers. Then
$$\lim_{T\to\infty} \frac{1}{T} \sum_{n=0}^{T-1} x_n$$
is called the Cesaro-limit of the sequence $\{x_n\}_n$ (provided it exists). Similarly, one can define the Cesaro lim sup and the Cesaro lim inf.

Further
$$\lim_{\alpha\uparrow 1} (1-\alpha) \sum_{n=0}^{\infty} \alpha^n x_n$$
is called the Abel limit of the sequence $\{x_n\}_n$ (provided it exists). Similarly one can define the Abel lim sup and Abel lim inf.

The following assertion holds.

Lemma 3.1.1 a)
$$\liminf_{n\to\infty} x_n \le \liminf_{T\to\infty} \frac{1}{T} \sum_{n=0}^{T-1} x_n \le \liminf_{\alpha\uparrow 1} (1-\alpha) \sum_{n=0}^{\infty} \alpha^n x_n \le$$
$$\le \limsup_{\alpha\uparrow 1} (1-\alpha) \sum_{n=0}^{\infty} \alpha^n x_n \le \limsup_{T\to\infty} \frac{1}{T} \sum_{n=0}^{T-1} x_n \le \limsup_{n\to\infty} x_n.$$

[1] see the papers by M. Vlasiou (there is an error in one of the proofs) on the course website, and and Asmussen[2, Chapters V, VI]. For a useful variant of regenerative theory, see [2, Chapter VII.5].

b) *Equivalent are*

> **i)** $\lim_{T\to\infty} \frac{1}{T}\sum_{n=0}^{T-1} x_n$ *exists and is finite.*
>
> **ii)** $\lim_{\alpha\uparrow 1}(1-\alpha)\sum_{n=0}^{\infty}\alpha^n x_n$ *exists and is finite.*
>
> *If the limit in (i) or (ii) exists, then they both exist and are equal.*

Note that b)$(i) \Rightarrow (ii)$ follows from (a). The reverse $((ii) \Rightarrow (i))$ is a very interesting result by Karamata, see [10, Thm. A.4.2].

The result can be applied to our situation by plugging in $x_n = P^{(n)}(f)r(f)$ (and assuming that these numbers are either bounded or non-negative). By Lemma 3.1.1 (b),

$$g(f) = \lim_{T\to\infty}\frac{V_T(i)}{T} = \lim_{\alpha\uparrow 1}(1-\alpha)V_\alpha(f),$$

thus connecting the expected average reward and the α-discounted reward!

3.2 Heuristics

We continue to focus on the Markov chain operated under the strategy $\boldsymbol{f}$. We will use the so-called *vanishing discount* technique, via the α-discount rewards to motivate a meaningful optimality equation for the average rewards.

Vanishing discount approach Here we again restrict to strategy $\boldsymbol{f}$, and assume that the properties from Section 3.1 and we start with the α-discount equation

$$V_\alpha^{\boldsymbol{f}}(i) = r_i(f) + \alpha\sum_j p_{ij}(f)V_\alpha^{\boldsymbol{f}}(j). \tag{3.2.1}$$

Clearly $V_\alpha^{\boldsymbol{f}}(i)$ is generally not bounded as $\alpha\uparrow 1$: the expected total reward over an infinite horizon will generally be infinitely large (or even not well-defined), without assumptions as have been made in Chater 2. However, the difference between the α-discount reward between two different initial states will remain bounded as $\alpha \uparrow 1$ (under reasonable assumptions, like we did in the previous section). In this case we substract $\alpha V_\alpha^{\boldsymbol{f}}(i_0)$ and get

$$\begin{aligned} V_\alpha^{\boldsymbol{f}}(i) - V_\alpha^{\boldsymbol{f}}(i_0) &= \mathsf{E}_i^{\boldsymbol{f}}\sum_{n=0}^{\tau_{i_0}-1}\alpha^n r_{X_n}(f) + \mathsf{E}_i^{\boldsymbol{f}}\alpha^{\tau_{i_0}} - V_\alpha^{\boldsymbol{f}}(i_0)\\ &= \mathsf{E}_i^{\boldsymbol{f}}\sum_{n=0}^{\tau_{i_0}-1}\alpha^n r_{X_n}(f) - \mathsf{E}_i^{\boldsymbol{f}}(1-\alpha^{\tau_{i_0}})V_\alpha^{\boldsymbol{f}}(i_0)\\ &= \mathsf{E}_i^{\boldsymbol{f}}\sum_{n=0}^{\tau_{i_0}-1}\alpha^n r_{X_n}(f) - \mathsf{E}_i^{\boldsymbol{f}}\sum_{n=0}^{\tau_{i_0}-1}\alpha^n\cdot(1-\alpha)V_\alpha^{\boldsymbol{f}}(i_0). \end{aligned}$$

By our assumptions, the limit on the right-hand side exists, hence the limit on the left-hand side. Denote this by $d^{\boldsymbol{f}}(i)$ then

$$\lim_{\alpha\uparrow 1}(V_\alpha^{\boldsymbol{f}}(i) - V_\alpha^{\boldsymbol{f}}(i_0)) = R_{i\,i_0}^{\boldsymbol{f}} - T_{i\,i_0}^{\boldsymbol{f}}\cdot g^{\boldsymbol{f}} =: d^{\boldsymbol{f}}(i). \tag{3.2.2}$$

To get an equation for the average expected reward, we substract $\alpha V_\alpha^{\boldsymbol{f}}(i_0)$ on both sides of (3.2.1) to obtain

$$(V_\alpha^{\boldsymbol{f}}(i) - V_\alpha^{\boldsymbol{f}}(i_0)) + (1-\alpha)V_\alpha^{\boldsymbol{f}}(i_0) = r_i(f) + \alpha \sum_j p_{ij}(f)(V_\alpha^{\boldsymbol{f}}(j) - V_\alpha^{\boldsymbol{f}}(i_0)).$$

Taking the limit $\alpha \uparrow 1$ on both sides yields

$$d^{\boldsymbol{f}}(i) + g^{\boldsymbol{f}} = r_i(f) + \sum_j p_{ij}(f) d^{\boldsymbol{f}}(j), \tag{3.2.3}$$

provided we can interchange summation and limit. This is allowed by the dominated convergence theorem, using that $\sum_j p_{ij}(f)T_{j\,i_0}^{\boldsymbol{f}}$, $\sum_j p_{ij}(f)R_{j\,i_0}^{\boldsymbol{f}} < \infty$.

Eqn. (3.2.3) will be the basis for the average reward optimality equation (abbreviated: AOE). The function $d^{\boldsymbol{f}}$ playing a role in this equation has the interpretation of being the *relative reward* through Eqn. 3.2.2: it measures the total expected reward earned until state i_0 is reached, compared to what we would have received if instead we would have earned $g^{\boldsymbol{f}}$ each time till absorption in i_0.

Another interpretation will be given, after a heuristic derivation of the AOE, using the *dynamic programming principle for a finite horizon optimisation problem.*

Before to studying this, we will provide a small example showing how to compute solutions to Eqn. 3.2.3.

Example 3.2.1 Let $\boldsymbol{S} = \{1, 2\}$, with one action per state: $\mathsf{A}(1) = \mathsf{A}(2) = \{1\}$. Put $\boldsymbol{f} = \binom{1}{1}$. Further

$$P^{\boldsymbol{f}} = \begin{pmatrix} 1/2 & 1/2 \\ 2/3 & 1/3 \end{pmatrix}, \quad r(f) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

Eqn 3.2.3 becomse

$$\begin{aligned} d^{\boldsymbol{f}}(1) + g^{\boldsymbol{f}} &= 1 + \frac{1}{2} d^{\boldsymbol{f}}(1) + \frac{1}{2} d^{\boldsymbol{f}}(2) \\ d^{\boldsymbol{f}}(2) + g^{\boldsymbol{f}} &= 2 + \frac{2}{3} d^{\boldsymbol{f}}(1) + \frac{1}{3} d^{\boldsymbol{f}}(2). \end{aligned}$$

This is a system of two linear equations with three unknowns The solution is not unique: indeed with $(d^{\boldsymbol{f}}, g^{\boldsymbol{f}})$ also $(d^{\boldsymbol{g}} + c\mathbf{1}, g^{\boldsymbol{g}})$ is a solution, where $c \in \mathbf{R}$ and $\mathbf{1}$ is the vector consisting of ones. As a consequence, we may choose $d^{\boldsymbol{f}}(1) = 0$ and solve for the other values. This gives the solution $d^{\boldsymbol{f}}(2) = 6/7$ and $g^{\boldsymbol{f}} = 10/7$.

Alternatively, we may compute $g^{\boldsymbol{f}}$ through the stationary distribution $\pi(f)$ by solving (3.1.3). Check that it is given by

$$\pi(f) = (\pi_1(f), \pi_2(f)) = (4/7, 3/7),$$

and so $g^{\boldsymbol{f}} = 1 \cdot 4/7 + 2 \cdot 3/7 = 10/7$.

Finite horizon approach In this case we will not fix a strategy, but consider a sequence of decision rules generated from Belmann's optimality equation. I.o.w., put terminal reward $q \equiv 0$, and compute the corresponding T-horizon maximum expected reward V_T^*, and optimal strategy $(f^T, f^{T-1}, \dots, f^1)$.

Suppose that for T large enough $f^T = f^*$ is fixed, and that the strategy $\boldsymbol{f}^*$ satisfies conditions (3.1.1) and (3.1.2) (with reference state i_0). So, for horizon $T+n$ we have $f^{T+n} = f^*$, i.o.w.

$$\begin{aligned} V_{T+n}^*(i) &= r_i(f^*) + \sum_j p_{ij}(f^*) V_{T+n-1}^*(j) \\ &= \max_{a \in \mathsf{A}(i)} \Big\{ r_i(a) + \sum_j p_{ij}(a) V_{T+n-1}^*(j) \Big\}. \qquad (3.2.4) \end{aligned}$$

Iterating yields

$$V_{T+n}^*(i) = V_n^{\boldsymbol{f}^*}(i) + \sum_j p_{ij}^{(n)}(f^*) V_T^*(j). \qquad (3.2.5)$$

In words: the maximum expected revenue onver $T+n$ periods is the sum of the revenue of the stationary strategy $\boldsymbol{f}^*$ over the first n periods and the maximum expected revenue over the last T periods. We know that $V_n^{\boldsymbol{f}^*}(i)$ will grow approximately as $ng^{\boldsymbol{f}^*}$, for all i (since ultimately on the average $g^{\boldsymbol{f}^*}$ is earned per unit time. For $n \geq 1$, let us therefore look at the *relative rewards* over finite periods, defined by

$$d_n^{\boldsymbol{f}^*}(i, i_0) := V_n^{\boldsymbol{f}^*}(i) - V_n^{\boldsymbol{f}^*}(i_0). \qquad (3.2.6)$$

Substituting (3.2.6) and (3.2.5) into (3.2.4) yields

$$\begin{aligned} & d_n^{\boldsymbol{f}^*}(i, i_0) + V_n^{\boldsymbol{f}^*}(i_0) - V_{n-1}^{\boldsymbol{f}^*}(i_0) + \sum_j p_{ij}^{(n)}(f^*) V_T^*(j) \\ &\quad = \max \Big\{ r_i(a) + \sum_j p_{ij}(a) \Big(d_{n-1}^{\boldsymbol{f}^*}(j, i_0) + \sum_k \sum_j p_{jk}^{(n-1)}(f^*) V_T^*(k) \Big\}. \qquad (3.2.7) \end{aligned}$$

Action $a = f^*(i)$ achieves the maximum on the right-hand side. Next, let $n \to \infty$ in Eqn. 3.2.7.

Let us assume that we may interchange limit and summation to obtain that

$$\lim_{n \to \infty} \sum_j p_{ij}^{(n)}(f^*) V_T^*(j) = \sum_j \pi_j(f^*) V_T^*(j).$$

The corresponding terms in Eqn. (3.2.7) will therefore cancel out in the limit $n \to \infty$. Therefore, we may concentrate on the remaining terms. If the conditions assumed guarantee existence of the limit

$$d^{\boldsymbol{f}^*}(i\, i_0) = \lim_{n \to \infty} d_n^{\boldsymbol{f}^*}(i\, i_0)$$

as well as

$$\lim_{n \to \infty} V_n^{\boldsymbol{f}^*}(i) - V_{n-1}^{\boldsymbol{f}^*}(i) = g^{\boldsymbol{f}^*},$$

then we get the AOE

$$d^{\boldsymbol{f}^*}(i\, i_0) + g^{\boldsymbol{f}^*} = \max_{a \in \mathsf{A}(i)} \Big\{ r_i(a) + \sum_j p_{ij}(a) d^{\boldsymbol{f}^*}(j\, i_0) \Big\}, \qquad (3.2.8)$$

and $a = f^*(i)$ attains the maximum in the right-hand side.

Hence $d^{\boldsymbol{f}^*}(i\,i_0)$, $i \in \boldsymbol{S}$, solves Eqn.(3.2.3) for $f = f^*$. Combination with the results in the previous paragraph for the α-discounted case, and $f = f^*$, yields that

$$d^{\boldsymbol{f}^*}(i\,i_0) = d^{\boldsymbol{f}^*}(i),$$

since $d^{\boldsymbol{f}^*}(i_0\,i_0) = 0 = d^{\boldsymbol{f}^*}(i_0)$, provided that (3.2.3) has a unique solution (upto the constant vector) in a certain space. This yields two different probabilistic interpretations of solutions to Eqn. (3.2.3)!

Unfortunately, the expected average reward problem resists answering many important questions. One of these is the following. The above procedure is nothing but the convergence of the successive approximations algorithm to a solution of the AOE (3.2.8), provided the limit operations assumed are justified. Only very strong but verifiable or less strong but hardly verifiable conditions seem to exist so far that do this job.

In the next paragraph we will present some sufficient conditions so that solutions to the AOE yield the maximum expected reward, and an optimal strategy.

3.3 Average reward optimality equation and conditions

We will present two types of conditions that guarantee that solutions to the AOE (3.2.8) yields desired optimality results. Apart from thos, we still assume that $\mathsf{A}(i) \subset \mathsf{A}$, with A finite, but we do not assume any boundedness conditions on the direct rewards any more.

Non-negative cost condition (NNCC) Consider a cost minimisation problem where instead of direct rewards $r_i(a)$, $a \in \mathsf{A}(i)$, $i \in \boldsymbol{S}$, the system controller incurs a direct cost $c_i(a)$, $a \in \mathsf{A}(i)$, $i \in \boldsymbol{S}$. The following conditions are assumed.

a) $c_i(a) \geq 0$, for all $a \in \mathsf{A}(i)$, $i \in \boldsymbol{S}$.

b) there exists a strategy $\boldsymbol{f}_0$ and a state i_0, such that

- $T_{i\,i_0}(f_0) < \infty$ and $C_{i\,i_0}(f_0) = \mathsf{E}_i \sum_{n=0}^{\tau_{i_0}-1} c_{X_n}(f_0) < \infty$ for all $i \in \boldsymbol{S}$; and
- there exists $\epsilon > 0$ such that the set $\mathcal{C} = \{i \,|\, c_i(a) \leq g^{\boldsymbol{f}_0} + \epsilon,$ for some $a \in \mathsf{A}(i)\}$ is finite.

c) for any $i \in \mathcal{C}$ there exists a strategy $\boldsymbol{f}_i$, such that $T_{i_0\,i}(f_i), C_{i_0\,i}(f_i) < \infty$ (i.o.w., any state in the set $\mathcal{C}$ can be reached within finite expected time and cost under some strategy from i_0).

The idea is that if there is one well-behaved strategy ($\boldsymbol{f}_0$), then a cheaper strategy should also be well-behaved. This cheaper strategy should return in finite expected time to the set $\mathcal{C}$, without 'escaping'. In fact, it guarantees that for each α the set $\mathcal{C}$ contains a minimum discounted cost state i_α with $V_\alpha(i_\alpha) = \min_i V_\alpha(i)$.

An analysis of these conditions can be found in [10], based on work by V. Borkar, as well as an upcoming paper of the lecturer with H. Blok. Another type of condition has been analysed in [3], [4], [5], for the average reward case. The result in these papers is more general.

Given a strategy $\boldsymbol{f}$, denote by $\nu(f)$ the number of positive recurrent classes in the Markov chain generated by $\boldsymbol{f}$.

M-geometric recurrence (MGR) The following conditions are assumed.

a) $\nu(f)=1$ and $P(f)$ generates an aperiodic Markov chain for all $\boldsymbol{f}$;

b) there exists a function $M:\boldsymbol{S}\to[1,\infty)$, a finite set B, constants $\beta\in(0,1)$, c, such that

$$\sum_j p_{ij}(a)M(j)\leq\beta M(i)+c\mathbf{1}_{\{B\}}(i),\quad a\in\mathsf{A}(i),i\in\boldsymbol{S}.$$

$$\sup_i\frac{\max_a|r_i(a)|}{M(i)}<\infty.$$

That these conditions are very strong, follows from their impact. In fact, as a result there exist constants $\gamma\in(0,1)$ and d such that for all $\boldsymbol{f}$

$$\sum_j|p_{ij}^{(n)}(f)-\pi_j(f)|r_j(f)\leq d\gamma^nM(i),\quad i\in\boldsymbol{S}$$

so convergence of the expected reward at time n to the average reward takes place at exponential rate, for any strategy $\boldsymbol{f}$. Transient Markov chains are therefore not allowed. On the other hand, the SA and PI algorithms (the version for the expected average rewards will be discussed below) do converge.

The following theorem holds true. For notational convenience, we write $r_i(a)=-c_i(a)$ under the NNCC to put it in the maximum expected reward framework. A solution to the AOE (3.3.1) below has to be multiplied by -1 to obtain the AOE for the minimum cost case.

For convenience, we say that the function $d:\boldsymbol{S}\to\mathbf{R}$ is *feasible* if either

- $\sup_i d(i)<\infty$, in case of NNCC, or
- $\sup_i|d(i)|/M(i)<\infty$ in case of MGR.

Theorem 3.3.1 *Assume that either NNCC or MGR holds. There exists a feasible function $d:\boldsymbol{S}\to\mathbf{R}$ and a constant g, such that*

$$d(i)+g=\max_{a\in\mathsf{A}(i)}\Big\{r_i(a)+\sum_j p_{ij}(a)d(j)\Big\},\quad i\in\boldsymbol{S},\tag{3.3.1}$$

$g=g^*(i)=g^*$ and any strategy $\boldsymbol{f}$ with

$$f(i)\in\arg\max_{a\in\mathsf{A}(i)}\Big\{r_i(a)+\sum_j p_{ij}(a)d(j)\Big\}$$

is average reward optimal, i.e. $g^{\boldsymbol{f}}(i)=g^*$ for $i\in\boldsymbol{S}$.

Under MGR, if the pair (w,j) with $w:\boldsymbol{S}\to\mathbf{R}$ and $j\in\mathbf{R}$ is another solution satisfying $\sup_i|w(i)|/M(i)<\infty$, then $j=g^*$ and $w(i)=d(i)+c\mathbf{1}$ for some constant $c\in\mathbf{R}$. The same holds true if NNCC holds and $\boldsymbol{S}$ is finite.

The function d is called the *(relative) value function*. By our previous discussion, if $\boldsymbol{f}^*$ is average reward optimal, then $d=d^{\boldsymbol{f}^*}+c\mathbf{1}=d^{\boldsymbol{f}^*}(\cdot,i_0)$, for some constant c and state i_0 that is positive recurrent under $\boldsymbol{f}^*$, see Section 3.2. Whence the name relative value function!

Remark 3.3.1 Suppose that $\boldsymbol{S}$ is finite. To what condition will MGR reduce in this case? And to which NNCC?

The proofs of this result use the vanishing discount approach. Thus also a relation with the α-discounted value function can be established. This can sometimes be exploited to derive the structure of an average reward optimal strategy, because for the α-discounted reward case convergence of the PI and SA algorithms is true under much more general conditions. Essentially, condition NNCC seems to be mainly applicable in conjunction with a vanishing discount argument, in an infinite state space case.

Theorem 3.3.2 *Under either condition NNCC and MGR, the following holds true.*

- $\lim_{\alpha\uparrow 1}(1-\alpha)V_\alpha(i) = g^*$.
- *Let* $h(i) = \lim_{n\to\infty}(V_{\alpha_n}(i) - V_\alpha(i_0))$, *for some sequence* $\alpha_n \uparrow 1$, *and any* $i_0 \in \boldsymbol{S}$. *Then* (h, g^*) *is a solution to the AOE (3.3.1), with the understanding that* $\sup_i h(i) < \infty$ *under NNCC, and* $\sup_i |h(i)|/M(i) < \infty$, *under MGR.*
- *Let* $f' = \lim_{n\to\infty} f_{\alpha_n}$, *where* $\boldsymbol{f}_{\alpha_n}$ *is* α_n*-discount optimal. Then* f' *is average reward optimal and* $g^{\boldsymbol{f}'}(i) = g^*$ *for* $i \in \boldsymbol{S}$.

Next we will discuss the average reward variants of the SA and PI algorithms.

3.4 Algorithms for computing value function and optimal strategy

Policy Iteration We now assume that either NNCC or MGR holds. Let be given a fixed strategy $\boldsymbol{f}$, and suppose that there exists a feasible function $d : \boldsymbol{S} \to \mathbf{R}$ and a constant g such that

$$g + d = r(f) + P(f)d. \tag{3.4.1}$$

Let us suppose that

$$f^1(i) \in \arg\max_{a\in \mathsf{A}(i)}\{r_i(a) + \sum_j p_{ij}(a)d(j)\}, \quad i \in \boldsymbol{S} \tag{3.4.2}$$

Then one can show the following policy improvement result.

Lemma 3.4.1

- $g^{\boldsymbol{f}^1}(i) \geq g$, $i \in \boldsymbol{S}$.
- *Suppose that either (i) MGR holds, or that (ii) NNCC holds,* $\boldsymbol{S}$ *is finite and the Markov chain under* $\boldsymbol{f}^1$ *has only one positive recurrent class. Then* $g^{\boldsymbol{f}^1}(i) = g^{\boldsymbol{f}^1}$, $i \in \boldsymbol{S}$. *Moreover, either* $g^{\boldsymbol{f}^1} > g^{\boldsymbol{f}}$ *or* $g^{\boldsymbol{f}^1} = g^{\boldsymbol{f}}$ *and there exists a feasible function* d^1 *with*

$$g^{\boldsymbol{f}} + d^1(i) = r_i(\boldsymbol{f}^1) + \sum_j p_{ij}(f^1)d^1(j),$$

and $d^1(i) \geq d(i)$ *for* $i \in \boldsymbol{S}$ *and there exists* $i_0 \in \boldsymbol{S}$ *such that* $d^1(i_0) > d(i_0)$.

The construedness of this description shows all the more that the average reward case is a complex one!

Note Consider Eqn. (3.4.1). It is an easy consequence of Lemma 3.4.1 that

$$f(i) \in \arg\max_{a \in \mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a) d(j)\}, \quad i \in \mathsf{A}(i)$$

implies that $\boldsymbol{f}$ is average optimal.

Lemma 3.4.1 justifies validity of the PI algorithm.

Policy Iteration Algorithm

0) Set $n := 0$. Choose any initial stationary, deterministic strategy $\boldsymbol{f}_0 = (f_0, \ldots)$.

1) Compute $d^n : \boldsymbol{S} \to \mathbf{R}$, $g^n \in \mathbf{R}$ by solving $d^n + g^n = r(f_n) + P(f_n) d_n$.

2) Put $\boldsymbol{f} := \boldsymbol{f}_n$ and compute $\boldsymbol{f}_{n+1} = \boldsymbol{f}'$ from (3.4.2), taking $\boldsymbol{f} = \boldsymbol{f}'$ if possible.

3) If $\boldsymbol{f}_{n+1} = \boldsymbol{f}_n$ then this strategy is optimal. Stop.
   Otherwise set $n := n + 1$, and go to step 1.

The same remarks made after the formulation of the PI algorithm in Chapter 2 apply here as well.

Lemma 3.4.2 *PI converges if in each step the strategy $\boldsymbol{f}^n$ satisfies the conditions on the solutions of the AOE from Lemma 3.4.1. Additionally, PI converges in finitely many steps if $\boldsymbol{S}$ is finite.*

Example 3.4.1 Inventory control A class of expensive goods kept in stock at a warehouse is sold directly to customers. The inventory level can be increased by placing a new order at the beginning of each period. Lead times are negligible, so that we assume the order to be available immediately. At most three items are kept on stock. Items that are not sold at the end of the period, can be kept for the next period, but imply an inventory cost of $h = 4$ per item. An order of n items costs $K + r \cdot n$, where $K = 4$ and $r = 2$ are the fixed and variable ordering costs.

If the demand in a period exceeds the number of items in stock, a penalty cost of $p = 12$ is incurred per item that can not be delivered. The demands in the subsequent periods form a sequence of independent and identically distributed random variables. Each period, the demand equals 0,1,2 or 3 items, each of which occurs with probability 1/4. Future cost of lost demand is already accounted for in the penalty cost p. Hence, the goal is to minimise the average cost per period.

Hint to simplify calculations. If the i-th and j-th row of the matrix B are equal, and $y = z + By$, then the column vectors y and z satisfy $y_i - z_i = y_j - z_j$.

a) Formulate this problem as a Markov decision problem. Describe the state space and the possible actions; determine the direct costs and the transition probabilities.

b) Suppose the stock level is increased to the maximum level of 3 items at the beginning of each period (if there are already 3 items in stock then no order is placed). What are the average cost and relative values of this strategy?

c) Perform one step of the PI algorithm, starting with the strategy described in b). It suffices to determine the new strategy.

d) Show that it is optimal to order 3 items when there is no item in stock, 2 items when there is 1 item in stock, and that no order should be placed if there are 2 (or 3) items in stock.

There are two ways to interpret inventory cost: charge the expected cost for the current period, or charge the cost for the inventory of the previous period. Note that for average optimality this does not make a difference since all periods are equally important. For the discounted case though it would matter! We start with the first.

Alternative 1

(a)

- State at beginning of period n: X_n =number of items in stock $\in \{0,1,2,3\}$.
- Action in state i: number of items ordered $\in \mathsf{A}(i) = \{0,\ldots,3-i\}$.
- In general: $r_i(a) = h\mathsf{E}(i+a-D)^+) + K\mathbf{1}_{\{a>0\}} + ra + p\mathsf{E}(D-i-a)^+$. Note that $\mathsf{E}(D-i-a)^+ = 3/2, 3/4, 1/4, 0$ if $i+a = 0,1,2,3$ respectively;
  $\mathsf{E}(i+a-D)^+ = 0, 1/4, 3/4, 3/2$ if $i+a = 0,1,2,3$ respectively.
  It follows that

$$r(0) = \begin{pmatrix} 18 \\ 10 \\ 6 \\ 6 \end{pmatrix}, \quad r(1) = \begin{pmatrix} 16 \\ 12 \\ 2 \\ * \end{pmatrix}, \quad r(2) = \begin{pmatrix} 14 \\ 14 \\ * \\ * \end{pmatrix}, \quad r(3) = \begin{pmatrix} 16 \\ * \\ * \\ * \end{pmatrix}.$$

  Further

$$P(0) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 3/4 & 1/4 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix}, \qquad P(1) = \begin{pmatrix} 3/4 & 1/4 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ * & * & * & * \end{pmatrix}$$

$$P(2) = \begin{pmatrix} 1/2 & 1/4 & 1/4 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ * & * & * & * \\ * & * & * & * \end{pmatrix}, \qquad P(3) = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix}$$

(b)

$$f = \begin{pmatrix} 3 \\ 2 \\ 1 \\ 0 \end{pmatrix}, \quad r(f) = \begin{pmatrix} 16 \\ 14 \\ 12 \\ 6 \end{pmatrix}, \quad P(f) = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix}$$

Because of the structure we immediately see see that each of the four states occurs with equal probability, so $g^{\boldsymbol{f}} = 1/4(16+14+12+6) = 12$. The relative rewards also follow easily (following the hint): $d^{\boldsymbol{f}} = (0,-2,-4,-10)$ (upto a constant vector, putting $d^{\boldsymbol{f}}(0) = 0$).

(c) $f'(0) = \arg\min\{18+0, 16-1/2, 14-3/2, 16-4\} = 3$;
$f'(1) = \arg\min|\{10-1/2, 12-3/2, 14-4\} = 0$;
$f'(2) = \arg\min\{6-3/2, 12-4\} = 0$;
$f'(3) = 0$. (d)

$$f = \begin{pmatrix} 3 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \quad r(f) = \begin{pmatrix} 16 \\ 14 \\ 6 \\ 6 \end{pmatrix}, \quad P(f) = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix}$$

Using the hint, we already know that taking $d^{\boldsymbol{f}f}(0) = 0$ we get $d^{\boldsymbol{f}}(1) = -2$, and $d^{\boldsymbol{f}}(3) = -10$. Using the equations yields $g^{\boldsymbol{f}} = 11 + 1/8$ and $d^{\boldsymbol{f}}(2) = -7 - 1/2$. Apply PI once yields:
$f'(0) = \arg\min\{18+0, 16-1/2, 14-1/4\times 19/2, 16-1/4\times 39/2\} = 3$;
$f'(1) = \arg\min|\{10-1/2, 12-1/4\times 19/2, 14-1/4\times 39/2\} = 2$;
$f'(2) = \arg\min\{6-14\times 19/2, 12-1/4\times 39/2\} = 0$;
$f'(3) = 0$. The policy does not change, so it is optimal.

Alternative 2
We charge the cost for the inventory of the previous period.
(a) We only specify the parameters if they are different from the ones in Alternative 1.
$r_i(a) = hi + K\mathbf{1}_{\{a>0\}} + ra + p\mathsf{E}(D-i-a)^+$. This yields

$$r(0) = \begin{pmatrix} 18 \\ 13 \\ 11 \\ 12 \end{pmatrix}, \quad r(1) = \begin{pmatrix} 15 \\ 13 \\ 14 \\ * \end{pmatrix}, \quad r(2) = \begin{pmatrix} 11 \\ 12 \\ * \\ * \end{pmatrix}, \quad r(3) = \begin{pmatrix} 10 \\ * \\ * \\ * \end{pmatrix}.$$

(b) $d^{\boldsymbol{f}} = (0,2,4,0)$. (d) $f^{\boldsymbol{f}} = (0,2,1/2,2)$.

Successive Approximations In Section 3.2 we have already discussed how SA works. Here we will formulate the precise convergence result.

Successive Approximations Algorithm

0) Set $n := 0$. Choose any (suitable) function $v_0 : \boldsymbol{S} \to \mathbf{R}$ (a common choice is $v_0 \equiv 0$). Choose $\epsilon > 0$.

1) Compute

$$V_{n+1}(i) = \max_{a\in\mathsf{A}(i)} \{r_i(a) + \sum_j p_{ij}(a)V_{n+1}(j)\}, \tag{3.4.3}$$

and let

$$f_{n+1}(i) \in \arg\max_{a\in\mathsf{A}(i)}\{r_i(a) + \sum_j p_{ij}(a)V_{n+1}(j)\}. \tag{3.4.4}$$

2) Let $b_n = \inf_i(V_n(i) - V_{n-1}(i))$, $B_n = \sup_i(V_n(i) - V_{n-1}(i))$. Stop, if $B_n - b_n < \epsilon$. Otherwise, set $n := n+1$, goto step 1.

In the case of discounted rewards, the stopping criterion is valied in both finite and infinite state space models. In case of the average rewards, this is not clear. The following assertion does hold.

Theorem 3.4.3 *Suppose that all stationary, deterministic strategies $\boldsymbol{f}$ generate aperiodic Markov chains.*

a) *(cf. [12, Thm 3.4.1]) Suppose that $\nu(f) = 1$ for all strategies $\boldsymbol{f}$. Suppose that $\boldsymbol{S}$ is finite. If $\boldsymbol{f}_n$ satisfies Eqns. (3.1.1) and (3.1.2) then*

$$b_n \leq g^{\boldsymbol{f}_n} \leq g^* \leq B_n,$$

where b_n is non-decreasing and B_n non-increasing in n.

b) *(cf. [11], [1]) Assume MGR (the state space may be countable). Fix any $i_0 \in \boldsymbol{S}$. Then $\lim_{n\to\infty}(V_n(i) - V_n(i_0)) =: d(i)$, $n \to \infty$, $i \in \boldsymbol{S}$ and $\lim V_n/n = g^*$, and (d, g) are a solution to the AOE (3.3.1). Further, any limit point of the sequence $\{f_n\}_n$ defines an average optimal strategy.*

The second statement has merely theoretical value, and can be used to derive results on the structure of optimal strategies. In turn, this limit the search for an optimal one.

Example 3.4.2 A two-state MDP Consider a Markov decision problem with two states (0 and 1) and two actions (1 and 2) per state. The direct cost is given by

$$c(1) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad c(2) = \begin{pmatrix} 0 \\ 2 \end{pmatrix},$$

the transition probabilities are

$$P(1) = \begin{pmatrix} 1/2 & 1/2 \\ 2/3 & 1/3 \end{pmatrix}, \quad P(2) = \begin{pmatrix} 1/4 & 3/4 \\ 1/3 & 2/3 \end{pmatrix}.$$

For a finite planning horizon, the terminal costs are $q = \binom{2}{1}$.

a) Determine the minimum cost over a period with twe deicision epochs. What is the corresponding optimal strategy?

Nest we would like to minimise the average cost for an infinite time horizon.

b) Find the strategy corresponding to the second iteration of the SA algorithm. Also determine the corresponding lower and upper bounds for the average cost.

c) Determine the average cost and corresponding relative values of the strategy found in (b).

d) Carry our one step of PI, starting with the strategy of part (b).

(a) $V_0(i) = q_i$, hence $V_0 = \binom{2}{1}$.
$V_1(0) = \min\{1 + \frac{1}{2} \cdot 2 + \frac{1}{2} \cdot 1, 0 + \frac{1}{4} \cdot 2 + \frac{3}{4} \cdot 1\} = \frac{5}{4}$;
$V_1(1) = \min\{2 + \frac{2}{3} \cdot 2 + \frac{1}{3} \cdot 1, 2 + \frac{1}{3} \cdot 2 + \frac{2}{3} \cdot 1\} = \frac{10}{3}$;

$V_2(0) = \min\{1 + \frac{1}{2} \cdot \frac{5}{4} + \frac{10}{3} \cdot 1, 0 + \frac{1}{4} \cdot \frac{5}{4} + \frac{3}{4} \cdot \frac{10}{3}\} = 2\frac{13}{16}$;
$V_1(1) = \min\{2 + \frac{2}{3} \cdot \frac{5}{4} + \frac{1}{3} \cdot \frac{10}{3}, 2 + \frac{1}{3} \cdot \frac{5}{4} + \frac{2}{3} \cdot \frac{10}{3}\} = 3\frac{17}{18}$.

The optimal strategy for two periods is:

$$f_1 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \quad f_2 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

(b) There are two possibilities.

1. The initial value function of SA may be chosen arbitrarily. By choosing $V_0 = q$ we can conclude from (a) that the corresponding rule is $f_2 = \binom{2}{1}$. Since $V_2(0) - V_1(0) = 25/16$ and $V_2(1) - V_2(1) = 11/18$, this gives upper bound $B_2 = 25/16$ and lower bound $b_2 = 11/18$ for the average cost.

2. If one chooses the standard initialisation $V_0(0) = V_0(1) = 0$, one needs to carry out the same steps as in (a). This gives

$$V_1 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \quad V_2 = \begin{pmatrix} \frac{3}{2} \\ \frac{8}{3} \end{pmatrix},$$

with f_2 identical to the computation under (1). This gives $B_2 = 3/2$ and $b_2 = 2/3$. Hence, in this case the standard initialisation gives sharper upper and lower bounds.

(c) Strategy $\boldsymbol{f}$ is determined by $f = f_2$. Hence

$$P(f) = \begin{pmatrix} \frac{1}{4} & \frac{3}{4} \\ \frac{2}{3} & \frac{1}{3} \end{pmatrix}, \quad r(f) = \begin{pmatrix} 0 \\ 2 \end{pmatrix}.$$

The relative values and average rewards satisfy

$$\begin{aligned} d(0) + g^* &= 0 + \frac{1}{4}d(0) + \frac{3}{4}d(1) \\ d(1) + g^* &= 2 + \frac{2}{3}d(0) + \frac{1}{3}d(1). \end{aligned}$$

Set $d(0) = 0$, then $g^* = \frac{3}{4}d(1)$. Hence $(1 + 3/4 - 1/3)d(1) = 2$ and so $d(1) = 24/17$ and $g = 18/17$.

(c) $f'(0) \in \arg\min\{1 + \frac{1}{2} \cdot 0 + \frac{1}{2} \cdot \frac{24}{17}, 0 + \frac{3}{4} \cdot \frac{24}{17}\} = \arg\min\{\frac{29}{17}, \frac{18}{17}\} = \{2\}$, and so $f'(0) = 2$; similarly $f'(1) = \arg\min\{2 + \frac{1}{3} \cdot \frac{24}{17}, 2 + \frac{2}{3}\frac{24}{17}\} = \{1\}$, and so $f'(1) = 1$. Since $f' = f$ we may conclude that f is optimal.

3.5 Generalisations

We have concentrated on the case where stationary deterministic decision rules give rise to a Markov chain with one positive recurrent class. When this is not satisfied, it does not mean that the framework would break down. Some modifications will be needed though.

3.6 Exercises

Exercise 3.1 The owner of a race horse wants to maximise the (discounted) returns of his horse. The (daily) discount factor is 2/3. It is possible to participate in a race every day, but after participating the horse may not be fit next day. If the horse is fit, the expected return for that day is €2,000,000. If the horse is still too tired, the expected return is only € 1,000,000.

Participation in a race is for free. If the horse is fit and participates in a race, it is fit the next day with probability 2/3 and with probability 1/3 it is still tired the next day. If the horse is fit and does not participate in a race, it will still be fit the next day. Similarly, the horse will not be fit the next day, if it participates in a race while not being fit. If a tired horse rests for a day, it will be fit the next day with probability 1/2 and it is still tired the next day with probability 1/2.

i) Formulate this problem as a Markov decision problem. Describe the state and action spaces and give the transition probablities and direct rewards.

ii) Apply two steps of the SA algorithm. In each step give the candidate strategy, as well as lower and upper bounds for the discount value function.

iii) Show that it is optimal to let the horse race every day and determine the optimal discounted rewards (the value function).

If, instead of discounted rewards, we wish to maximise the long-run average reward, it turns out not to be optimal anymore to let the horse race every day.

iv) Show that it is average optimal to only let the horse participate if it is fit. Which condition do you use to justify your result?

Animal protection regulation does not allow the horse to participate in more than 50% of the races.

v) Describe a method how you could obtain an optimal strategy under this restriction. If possible, compute an optimal strategy.

Exercise 3.2 Consider the arrival control model described in Section 2.9.

a) For which values of λ and μ does there exists a function M such that the model satisfies condition MGR? (*Hint*: if there is such a function, it should be of the form $M(i) = \alpha^i$, for some $\alpha > 1$).

b) If λ and μ are such that MGR holds, what methods can one apply (according to the lecture notes) to show the existence of an average optimal policy of threshold type? If MGR fails, what methods can one use instead?

c) Perform the steps in Exercise 2.7, but now for the average reward case. This implies that you may have to reformulate the questions, where the discount factor plays an explicit role.

Exercise 3.3 Machine repair A production facility has 3 machines. If a machine starts up correctly in the morning, it renders a daily production of €1. A machine that does not start up correctly, needs to be repaired. A visit of a repair man costs €3 per day. The repair man repairs all *broken machines* in the same day (the repair cost is a lump cost, so it does not depend on the number of machines repaired). A machine that has been repaired always starts up correctly the next day. The number of machines that start up correctly the next day depends on the number of correctly working machines at present. This probability distirbution is given in the table below, where m stands for the number of (presently) working machines and n stands for the number of the ones that start up correctly the next day.

m	$n = 0$	$n = 1$	$n = 2$	$n = 3$
1	$\frac{1}{2}$	$\frac{1}{2}$	0	0
2	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0
3	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

a) Formulate this problem as a Markov decision problem. Describe the state and action spaces; determine the direct rewards/costs and the transition probabilities.

b) Suppose that the decision is to never repair. Calculate the average rewards and corresponding relative values.

c) Apply the PI algorithm once, starting with the strategy from b). Calculate the average rewards and relative values corresponding to that strategy if the new strategy satisfies Eqns. (3.1.1) and (3.1.2). Otherwise indicate which of (3.1.1) and (3.1.2) is not satisfied and why.

d) Show that it is optimal to only let the repair man come when all machines are broken.

Exercise 3.4 Drill platform The maximum daily output of a drill platform in the North Sea is €10,000,000 per day. For security reasons the process is paused at night. It is possible that the interruption leads to a pollution of the installation, giving a daily production of only €5,000,000. If that is the case, it is possible to clean the installation, at the cost of losing the production of one day. The cleaning cost is negligible.

The probability that the installation is polluted after a day of maximum production is 1/3 (and with probability 2/3 the installation can work at full capacity). A polluted installation that is not cleaned, remains polluted. Cleaning the installation has the desired effect with probability 1/2, and with probability 1/2 it remains polluted. The next day, the decision be taken anew to clean the installation. The aim is too maximise the average output.

a) Formulate this problem as a Markov decision problem. Describe state and action spaces.

b) Carry out one step of SA. Give the corresponding candidate strategy, as well as lower and upper bounds for the optimal rewards.

c) Explain whether the algorithm will converge in this example. Motivate your answer by discussing potential problems with this algorithm.

d) Compute the optimal strategy and the corresponding maximum reward (numerically using a computer program - hand in the code).

Bibliography

[1] E. Altman, A. Hordijk, and F.M. Spieksma (1997), Contraction conditions for average and α-discount optimality in countable Markov games with unbounded rewards. *Math. Operat. Res.* **22**, 588–619.

[2] S. Asmussen (2003), *Applied Probability and Queues.* J. Wiley, New York, 2d edition.

[3] R. Dekker and A. Hordijk (1988), Average, sensitive and Blackwell optimal policies in denumerable Markov decision chains with unbounded rewards. *Math. Operat. Res.* **13**, 395–421.

[4] R. Dekker and A. Hordijk (1992), Recurrence conditions for average and Blackwell optimality in denumerable Markov decision chains. *Math. Operat. Res.* **17**, 271–289.

[5] R. Dekker, A. Hordijk, and F.M. Spieksma (1994), On the relation between recurrence and ergodicity properties in denumerable Markov decision chains. *Math. Operat. Res.* **19**, 539–559.

[6] C. Derman (1970), *Finite state Markovian decision processes.* Academis Press, New York.

[7] R.A. Howard (1960), *Dynamic Programming and Markov Processes.* Wiley, New York.

[8] M.L. Puterman (2005), *Markov Decision Processes: Discrete Stochastic Programming.* J. Wiley & Sons, New Jersey, 2d edition edition.

[9] S.M. Ross (1970), *Applied Probability Models with Optimization Applications.* Holden Day, San Francisco.

[10] L.I. Sennott (1999), *Stochastic Dynamic Programming and the Control of Queueing Systems.* Wiley Series in Probability and Statistics. Wiley, New York.

[11] F.M. Spieksma (1990), *Geometrically ergodic Markov Chains and the optimal Control of Queues.* PhD thesis. Available on request from the author.

[12] H.C. Tijms (1994), *Stochastic Models – An Algorithmic Approach.* Wiley, New York.