Markov Decision Processes: Lecture Notes for STP 425

Jay Taylor

November 26, 2012

Contents

Chapter 1

Overview

1.1 Sequential Decision Models

This course will be concerned with sequential decision making under uncertainty, which we will represent as a discrete-time stochastic process that is under the partial control of an external observer. At each time, the state occupied by the process will be observed and, based on this observation, the controller will select an action that influences the state occupied by the system at the next time point. Also, depending on the action chosen and the state of the system, the observer will receive a reward at each time step. The key constituents of this model are the following:

- a set of decision times (epochs);
- a set of system states (state space);
- a set of available actions;
- the state- and action-dependent rewards or costs;
- the state- and action-dependent transition probabilities on the state space.

Given such a model, we would like to know how the observer should act so as to maximize the rewards, possibly subject to some constraints on the allowed states of the system. To this end, we will be interested in finding *decision rules*, which specify the action be chosen in a particular epoch, as well as *policies*, which are sequences of decision rules.

In general, a decision rule can depend not only on the current state of the system, but also on all previous states and actions. However, due to the difficulty of analyzing processes that allow arbitrarily complex dependencies between the past and the future, it is customary to focus on Markov decision processes (MDPs), which have the property that the set of available actions, the rewards, and the transition probabilities in each epoch depend only on the current state of the system.

The principle questions that we will investigate are:

1. When does an optimal policy exist?
2. When does it have a particular form?
3. How can we efficiently find an optimal policy?

These notes are based primarily on the material presented in the book 'Markov Decision Processes: Discrete Stochastic Dynamic Programming' by Martin Puterman (Wiley, 2005).

1.2 Examples

Chapter 1 of Puterman (2005) describes several examples of how Markov decision processes can be applied to real-world problems. I will describe just three of these in lecture, including:

1. Inventory management.
2. SIR models with vaccination (not in Puterman).
3. Evolutionary game theory: mate desertion by Cooper's Hawks.

Chapter 2

Discrete-time Markov Chains

2.1 Formulation

A **stochastic process** is simply a collection of random variables $\{X_t : t \in T\}$ where T is an index set that we usually think of as representing time. We will say that the process is E-valued if each of the variables X_t takes values in a set E. In this course we will mostly be concerned with discrete-time stochastic processes and so we will usually consider sequences of variables of the form $(X_n : n \geq 0)$ or occasionally $(X_n : n \in \mathbb{Z})$. Often we will interpret X_n to be the value of the process at time n, where time is measured in some specified units (e.g., days, years, generations, etc.), but in principle there is no need to assume that the times are evenly spaced or even that the index represents time.

When we model the time evolution of a physical system using a deterministic model, one desirable property is that the model should be dynamically sufficient. In other words, if we know that the system has state $X_t = x$ at time t, then this should be sufficient to determine all future states of the system no matter how the system arrived at state x at time t. While dynamical sufficiency is too much to ask for from a stochastic process, a reasonable counterpart would be to require the future states of the process to be conditionally independent of the past given the current state $X_t = x$. Stochastic processes that have this property are called **Markov processes** in general and **Markov chains** in the special case that the state space E is either finite or countably infinite.

Definition 2.1. *A stochastic process $X = (X_n; n \geq 0)$ with values in a set E is said to be a* ***discrete time Markov process*** *if for every $n \geq 0$ and every set of values $x_0, \cdots, x_n \in E$, we have*

$$\mathbb{P}\left(X_{n+1} \in A | X_0 = x_0, X_1 = x_1, \cdots, X_n = x_n\right) = \mathbb{P}\left(X_{n+1} \in A | X_n = x_n\right), \tag{2.1}$$

whenever A is a subset of E such that $\{X_{n+1} \in A\}$ is an event. In this case, the functions defined by

$$p_n(x, A) = \mathbb{P}(X_{n+1} \in A | X_n = x)$$

are called the ***one-step transition probabilities*** *of X. If the functions $p_n(x, A)$ do not depend on n, i.e., if there is a function p such that*

$$p(x, A) = \mathbb{P}(X_{n+1} \in A | X_n = x)$$

for every $n \geq 0$, then we say that X is a ***time-homogeneous Markov process*** *with* ***transition function*** *p. Otherwise, X is said to be* ***time-inhomogeneous****.*

In light of condition (2.1), Markov processes are sometimes said to lack memory. More precisely, it can be shown that this condition implies that conditional on the event $\{X_n = x_n\}$, the variables $(X_{n+k}; k \geq 1)$ are independent of the variables $(X_{n-k}; k \geq 1)$, i.e., the future is conditionally independent of the past given the present. This is called the **Markov property** and we will use it extensively in this course.

We can think about Markov processes in two ways. On the one hand, we can regard $X = (X_n; n \geq 0)$ as a collection of random variables that are all defined on the same probability space. Alternatively, we can regard X itself as a random variable which takes values in the space of functions from $\mathbb{N}$ into E by defining

$$X(n) \equiv X_n.$$

In this case, X is said to be a function-valued or path-valued random variable and the particular sequence of values $(x_n; n \geq 0)$ that the process assumes is said to be a **sample path** of X.

Example 2.1. *Any i.i.d. sequence of random variables $X_1, X_2, \cdots$ is trivially a Markov process. Indeed, since all of the variables are independent, we have*

$$\mathbb{P}\left(X_{n+1} \in A | X_1 = x_1, \cdots, X_n = x_n\right) = \mathbb{P}\left(X_{n+1} \in A\right) = p(x_n, A),$$

and so the transition function $p(x, A)$ does not depend on x.

Example 2.2. ***Discrete-time Random Walks***

Let $Z_1, Z_2, \cdots$ be an i.i.d. sequence of real-valued random variables with probability density function $f(x)$ and define the process $X = (X_n; n \geq 0)$ by setting $X_0 = 0$ and

$$X_{n+1} = X_n + Z_{n+1}.$$

X is said to be a discrete-time random walk and a simple calculation shows that X is a time-homogeneous Markov process on $\mathbb{R}$ with transition function

$$\begin{aligned} \mathbb{P}\left(X_{n+1} \in A | X_0 = x_0, \cdots, X_n = x\right) &= \mathbb{P}\left(X_{n+1} \in A | X_n = x\right) \\ &= \mathbb{P}\left(Z_{n+1} - x \in A | X_n = x\right) \\ &= \mathbb{P}\left(Z_{n+1} - x \in A\right) \\ &= \int_A f(z - x) dz. \end{aligned}$$

One application of random walks is to the kinetics of particles moving in an ideal gas. Consider a single particle and suppose that its motion is completely determined by a series of collisions with other particles present in the gas, each of which imparts a random quantity $Z_1, Z_2, \cdots$ to the velocity of the focal particle. Since particles move independently between collisions in ideal gases, the velocity of the focal particle following the n'th collision will be given by the sum $X_n = Z_1 + \cdots + Z_n$, implying that the velocity evolves as a random walk. Provided that the variables Z_i have finite variance, one prediction of this model (which follows from the Central Limit Theorem) is that for large n the velocity will be approximately normally distributed. Furthermore, if we extend this model to motion in a three-dimensional vessel, then for large n the speed of the particle (the Euclidean norm of the velocity vector) will asymptotically have the Maxwell-Boltzmann distribution. (Note: for a proper analysis of this model, we also need to consider the correlations between particle velocities which arise when momentum is transferred from one particle to another.)

Random walks also provide a simple class of models for stock price fluctuations. For example, let Y_n be the price of a particular stock on day n and suppose that the price on day $n+1$ is given by $Y_{n+1} = D_{n+1} Y_n$, where $D_1, D_2, \cdots$ is a sequence of i.i.d. non-negative random variables. Then the variables $X_n = \log(Y_n)$ will form a random walk with step sizes $\log(D_1), \log(D_2), \cdots$. In this case, the CLT implies that for sufficiently large n, the price of the stock will be approximately log-normally distributed provided that the variables $\log(D_i)$ have finite variance.

We can also construct a more general class of random walks by requiring the variables $Z_1, Z_2, \cdots$ to be independent but not necessarily identically-distributed. For example, if each variable Z_n has its own density f_n, then the transition functions $p_n(x, A)$ will depend on n,

$$\mathbb{P}\left(X_{n+1} \in A | X_n = x\right) = \int_A f_n(y - x) dy,$$

and so the process $X = (X_n; n \geq 0)$ will be time-inhomogeneous. Time-inhomogeneous Markov processes are similar to ordinary differential equations with time-varying vector fields in the sense that the 'rules' governing the evolution of the system are themselves changing over time. If, for example, the variables X_n denote the position of an animal moving randomly in its home territory, then the distribution of increments could change as a function of the time of day or the season of the year.

Definition 2.2. *A stochastic process $X = (X_n; n \geq 0)$ with values in the countable set $E = \{1, 2, \cdots\}$ is said to be a* ***time-homogeneous discrete-time Markov chain*** *with* ***initial distribution*** *ν and* ***transition matrix*** *$P = (p_{ij})$ if*

1. *for every $i \in E$, $\mathbb{P}\left(X_0 = i\right) = \nu_i$;*
2. *for every $n \geq 0$ and every set of values $x_0, \cdots, x_{n+1} \in E$, we have*

$$\begin{aligned} \mathbb{P}\left(X_{n+1} = x_{n+1} | X_0 = x_0, X_1 = x_1, \cdots, X_n = x_n\right) &= \mathbb{P}\left(X_{n+1} = x_{n+1} | X_n = x_n\right) \\ &= p_{x_n x_{n+1}}. \end{aligned}$$

In these notes, we will say that X is a DTMC for short.

Since p_{ij} is just the probability that the chain moves from i to j in one time step and since the variables X_n always take values in E, each vector $p_i = (p_{i1}, p_{i2}, \cdots)$ defines a probability distribution on E and

$$\sum_{j \in E} p_{ij} = \mathbb{P}\left(X_1 \in E | X_0 = i\right) = 1$$

for every $i \in E$. In other words, all of the row sums of the transition matrix are equal to 1. This motivates our next definition.

Definition 2.3. *Suppose that E is a countable (finite or infinite) index set. A matrix $P = (p_{ij})$ with indices ranging over E is said to be a* ***stochastic matrix*** *if all of the entries p_{ij} are non-negative and all of the row sums are equal to one:*

$$\sum_{j \in E} p_{ij} = 1 \quad \text{for every } i \in E.$$

Thus every transition matrix of a Markov chain is a stochastic matrix, and it can also be shown that any stochastic matrix with indices ranging over a countable set E is the transition matrix for a DTMC on E.

Remark 2.1. *Some authors define the transition matrix to be the transpose of the matrix P that we have defined above. In this case, it is the column sums of P that are equal to one.*

Example 2.3. *The transition matrix P of any Markov chain with values in a two state set $E = \{1, 2\}$ can be written as*

$$P = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix},$$

where $p, q \in [0, 1]$. Here p is the probability that the chain jumps to state 2 when it occupies state 1, while q is the probability that it jumps to state 1 when it occupies state 2. Notice that if $p = q = 1$, then the chain cycles deterministically from state 1 to 2 and back to 1 indefinitely.

Theorem 2.1. *Let X be a time-homogeneous DTMC with transition matrix $P = (p_{ij})$ and initial distribution ν on E. Then*

$$\mathbb{P}\left(X_0 = x_0, X_1 = x_1, \cdots, X_n = x_n\right) = \nu(x_0) \prod_{i=0}^{n-1} p_{x_i, x_{i+1}}.$$

Proof. By repeated use of the Markov property, we have

$$\begin{aligned}
&\mathbb{P}\left(X_0 = x_0, X_1 = x_1, \cdots, X_n = x_n\right) = \\
&\qquad = \mathbb{P}\left(X_0 = x_0, \cdots, X_{n-1} = x_{n-1}\right) \cdot \mathbb{P}\left(X_n = x_n | X_0 = x_0, \cdots, X_{n-1} = x_{n-1}\right) \\
&\qquad = \mathbb{P}\left(X_0 = x_0, \cdots, X_{n-1} = x_{n-1}\right) \cdot \mathbb{P}\left(X_n = x_n | X_{n-1} = x_{n-1}\right) \\
&\qquad = \mathbb{P}\left(X_0 = x_0, \cdots, X_{n-1} = x_{n-1}\right) \cdot p_{x_{n-1}, x_n} \\
&\qquad = \cdots \\
&\qquad = \mathbb{P}\left(X_0 = x_0, X_1 = x_1\right) \cdot p_{x_1, x_2} \cdots p_{x_{n-1}, x_n} \\
&\qquad = \mathbb{P}\left(X_0 = x_0\right) \cdot \mathbb{P}\left(X_1 = x_1 | X_0 = x_0\right) \cdot p_{x_1, x_2} \cdots p_{x_{n-1}, x_n} \\
&\qquad = \nu(x_0) \prod_{i=0}^{n-1} p_{x_i, x_{i+1}},
\end{aligned}$$

where $\nu(x_0)$ is the probability of x_0 under the initial distribution ν. □

One application of Theorem (2.1) is to likelihood inference. For example, if the transition matrix of the Markov chain depends on a set of parameters, Θ, i.e., $P = P^{(\Theta)}$, that we wish to estimate using observations of a single chain, say $x = (x_0, \cdots, x_n)$, then the likelihood function will take the form

$$L(\Theta | x) = \nu(x_0) \prod_{i=0}^{n-1} p^{(\Theta)}_{x_i, x_{i+1}},$$

and the maximum likelihood estimate of Θ will be the value of Θ that maximizes $L(\Theta|x)$.

The next theorem expresses an important relationship that holds between the n-step transition probabilities of a DTMC X and its r- and $n-r$-step transition probabilities. As the name suggests, the **n-step transition probabilities** $p_{ij}^{(n)}$ of a DTMC X are defined for any $n \geq 1$ by

$$p_{ij}^{(n)} = \mathbb{P}\left(X_n = j | X_0 = i\right).$$

In fact, it will follow from this theorem that these too are independent of time whenever X is time-homogeneous, i.e., for every $k \geq 0$,

$$p_{ij}^{(n)} = \mathbb{P}\left(X_{n+k} = j | X_k = i\right),$$

which means that $p_{ij}^{(n)}$ is just the probability that the chain moves from i to j in n time steps.

Theorem 2.2. ***(Chapman-Kolmogorov Equations)*** *Assume that X is a time-homogeneous DTMC with n-step transition probabilities $p_{ij}^{(n)}$. Then, for any non-negative integer $r < n$, the identities*

$$p_{ij}^{(n)} = \sum_{k \in E} p_{ik}^{(r)} p_{kj}^{(n-r)} \tag{2.2}$$

hold for all $i, j \in E$.

Proof. By using first the law of total probability and then the Markov property, we have

$$\begin{aligned} p_{ij}^{(n)} &= \mathbb{P}(X_n = j | X_0 = i) \\ &= \sum_{k \in E} \mathbb{P}(X_n = j, X_r = k | X_0 = i) \\ &= \sum_{k \in E} \mathbb{P}(X_n = j | X_r = k, X_0 = i) \cdot \mathbb{P}(X_r = k | X_0 = i) \\ &= \sum_{k \in E} \mathbb{P}(X_n = j | X_r = k) \cdot \mathbb{P}(X_r = k | X_0 = i) \\ &= \sum_{k \in E} p_{ik}^{(r)} p_{kj}^{(n-r)}. \end{aligned}$$

□

One of the most important features of the Chapman-Kolmogorov equations is that they can be succinctly expressed in terms of matrix multiplication. If we write $P^{(n)} = (p_{ij}^{(n)})$ for the matrix containing the n-step transition probabilities, then (2.2) is equivalent to

$$P^{(n)} = P^{(r)} P^{(n-r)}.$$

In particular, if we take $n = 2$ and $r = 1$, then since $P^{(1)} = P$, we see that

$$P^{(2)} = PP = P^2.$$

This, in turn, implies that $P^{(3)} = PP^2 = P^3$, and continuing in this fashion shows that $P^{(n)} = P^n$ for all $n \geq 1$. Thus, **the n-step transition probabilities of a DTMC can be calculated by raising the one-step transition matrix to the n'th power.** This observation is important for several reasons, one being that if the state space is finite, then many of the properties of a Markov chain can be deduced using methods from linear algebra.

Example 2.4. *Suppose that X is the two-state Markov chain described in Example 2.3. Although the n-step transition probabilities can be calculated by hand in this example, we can more efficiently calculate the powers of P by diagonalizing the transition matrix. In the following, we will let $d = p + q \in [0, 2]$. We first solve for the eigenvalues of P, which are the roots of the characteristic equation:*

$$\lambda^2 - (2 - d)\lambda + (1 - d) = 0$$

giving $\lambda = 1$ and $\lambda = 1 - d$ as the eigenvalues. As an aside, we note that any stochastic matrix P has $\lambda = 1$ as an eigenvalue and that $v = (1, \cdots, 1)^T$ is a corresponding right eigenvector (here T denotes the transpose). We also need to find a right eigenvector corresponding to $\lambda = 1 - d$ and a direct calculation shows that $v = (p, -q)^T$ suffices. If we let Λ be the matrix formed from these two eigenvectors by setting

$$\Lambda = \begin{pmatrix} 1 & p \\ 1 & -q \end{pmatrix},$$

and we let D be the diagonal matrix with entries $D_{11} = 1$ and $D_{22} = 1 - d$, then we can write the transition matrix P as the product

$$P = \Lambda D \Lambda^{-1}, \tag{2.3}$$

where the matrix inverse Λ^{-1} is equal to

$$\Lambda^{-1} = \frac{1}{d}\begin{pmatrix} q & p \\ 1 & -1 \end{pmatrix}.$$

The representation given in (2.3) is useful in part because it allows us to calculate all of the powers of P in one fell swoop:

$$\begin{aligned} P^n &= \Lambda D^n \Lambda^{-1} = \begin{pmatrix} 1 & p \\ 1 & -q \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & (1-d)^n \end{pmatrix}\frac{1}{d}\begin{pmatrix} q & p \\ 1 & -1 \end{pmatrix} \\ &= \frac{1}{d}\begin{pmatrix} q + p\cdot\alpha^n & p(1-\alpha^n) \\ q(1-\alpha^n) & p + q\cdot\alpha^n \end{pmatrix}, \end{aligned}$$

where $\alpha = 1 - d$. This shows, for example, that if $X_0 = 1$, then the probability that the chain is still in state 1 at time n is equal to $(q + p\cdot\alpha^n)/d$, which decreases to q/d monotonically when $\alpha \in [0, 1)$ and tends to this limit in an oscillating fashion when $\alpha \in (-1, 0)$. Thus the magnitude of the constant α determines how rapidly this Markov chain 'forgets' its initial condition.

Theorem 2.3. *Suppose that X is a time-homogeneous DTMC with transition matrix P and initial distribution ν. Then the distribution of X_n is given by the vector of probabilities*

$$\nu P^n,$$

where $\nu = (\nu_1, \nu_2, \cdots)$ is the vector representation of the initial distribution.

Proof. The result follows from the law of total probability:

$$\begin{aligned} \mathbb{P}(X_n = j) &= \sum_{i\in E}\mathbb{P}(X_n = j|X_0 = i)\cdot\mathbb{P}(X_0 = i) \\ &= \sum_{i\in E}\nu_i\left(P^{(n)}\right)_{ij} \\ &= (\nu P^n)_j\,. \end{aligned}$$

□

2.2 Asymptotic Behavior of Markov Chains

Theorem (2.1) in the previous section told us how to calculate the probability that a DTMC X assumes any particular finite sequence of values. This is important, for example, if the transition matrix P of the chain depends on a group of parameters Θ and our aim is to use a set of observations $(x_1, x_2, \cdots, x_n)$ to identify the maximum likelihood estimate (MLE) of Θ. In this section, our focus will turn to the long-term behavior of DTMC's. In biology, such considerations are important when we are interested, for example, in the fate of a new mutation in a population or in the long-term persistence of an infectious disease or in the steady-state distribution of transcription factors and proteins in a noisy cell. We begin by introducing some new terminology and notation.

2.2.1 Class Structure

The terminology of this section is motivated by the observation that we can sometimes decompose the state space of a Markov chain into subsets called communicating classes on which the chain has relatively simple behavior.

Definition 2.4. *Let X be a DTMC on E with transition matrix P.*

1. *We say that i **leads to** j, written $i \rightarrow j$, if for some integer $n \geq 0$*

$$p_{ij}^{(n)} = P_i\left(X_n = j\right) > 0.$$

   In other words, $i \rightarrow j$ if the process X beginning at $X_0 = i$ has some positive probability of eventually arriving at j.

2. *We say that i **communicates** with j, written $i \leftrightarrow j$, if i leads to j and j leads to i.*

It can be shown that the relation $i \leftrightarrow j$ is an equivalence relation on E:

1. Each element communicates with itself: $i \leftrightarrow j$;
2. i communicates with j if and only if j communicates with i;
3. If i communicates with j and j communicates with k, then i communicates with k. This follows from the Chapman-Kolmogorov equations by first choosing r and n so that $p_{ij}^{(r)} > 0$ and $p_{jk}^{(n-r)} > 0$ (which we can do since $i \leftrightarrow j$ and $j \leftrightarrow k$), and then observing that

$$p_{ik}^{(n)} \geq p_{ij}^{(r)} p_{jk}^{(n-r)}.$$

Our next definition is motivated by the fact that any equivalence relation on a set defines a partition of that set into equivalence classes: $E = C_1 \cup C_2 \cup C_3 \cup \cdots$.

Definition 2.5. *Let X be a DTMC on E with transition matrix P.*

1. *A nonempty subset $C \subset E$ is said to be a **communicating class** if it is an equivalence class under the relation $i \leftrightarrow j$. In other words, each pair of elements in C is communicating, and whenever $i \in C$ and $j \in E$ are communicating, $j \in C$.*
2. *A communicating class C is said to be **closed** if whenever $i \in C$ and $i \rightarrow j$, we also have $j \in C$. If C is a closed communicating class for a Markov chain X, then that means that once X enters C, it never leaves C.*
3. *A state i is said to be **absorbing** if $\{i\}$ is a closed class, i.e., once the process enters state i, it is stuck there forever.*
4. *A Markov chain is said to be **irreducible** if the entire state space E is a communicating class.*

2.2.2 Hitting Times and Absorption Probabilities

In this section we will consider the following two problems. Suppose that X is a DTMC and that $C \subset E$ is an absorbing state or, more generally, any closed communicating class for X. Then two important questions are: (i) what is the probability that X is eventually absorbed by A?; and (ii) assuming that this probability is 1, how long does it take on average for absorption to occur? For example, in the context of the Moran model without mutation, we might be interested in knowing the probability that A is eventually fixed in the population as well as the mean time for one or the other allele to be fixed. Clearly, the answers to these questions will typically depend on the initial distribution of the chain. Because the initial value X_0 is often known, e.g., by direct observation or because we set it when running simulations, it will be convenient to introduce the following notation. We will use $\mathbb{P}_i$ to denote the conditional distribution of the chain given that $X_0 = i$,

$$\mathbb{P}_i\left(A\right) = \mathbb{P}\left(A|X_0 = i\right)$$

where A is any event involving the chain. Similarly, we will use $\mathbb{E}_i$ to denote conditional expectations given $X_0 = i$,

$$\mathbb{E}_i\left[Y\right] = \mathbb{E}\left[Y|X_0 = i\right],$$

where Y is any random variable defined in terms of the chain.

Definition 2.6. *Let X be a DTMC on E with transition matrix P and let $C \subset E$ be a closed communicating class for X.*

1. *The* ***absorption time*** *of C is the random variable $\tau^C \in \{0, 1, \cdots \infty\}$ defined by*

$$\tau^C = \begin{cases} \min\{n \geq 0 : X_n \in C\} & \text{if } X_n \in C \text{ for some } n \geq 0 \\ \\ \infty & \text{if } X_n \notin C \text{ for all } n. \end{cases}$$

2. *The* ***absorption probability*** *of C starting from i is the probability*

$$h_i^C = \mathbb{P}_i\left(\tau^C < \infty\right).$$

3. *The* ***mean absorption time*** *by C starting from i is the expectation*

$$k_i^C = \mathbb{E}_i\left[\tau^C\right].$$

The following theorem allows us, in principle, to calculate absorption probabilities by solving a system of linear equations. When the state space is finite, this can often be done explicitly by hand or by numerically solving the equations. In either case, this approach is usually much faster and more accurate than estimating the absorption probabilities by conducting Monte Carlo simulations of the Markov chain.

Theorem 2.4. *The vector of absorption probabilities $h^C = (h_1^C, h_2^C, \cdots)$ is the minimal non-negative solution of the system of linear equations,*

$$\begin{cases} h_i^C = 1 & \text{if } i \in C \\ h_i^C = \sum_{j \in E} p_{ij} h_j^C & \text{if } i \notin C \end{cases}$$

To say that h^C is a minimal non-negative solution means that each value $h_i^C \geq 0$ and that $h_i^C \leq x_i$ if $x = (x_1, x_2, \cdots)$ is another non-negative solution to this linear system.

Proof. We will show that h^C is a solution to this system of equations; see Norris (1997) for a proof of minimality.

Clearly, $h_i^C = 1$ by definition if $i \in C$. If $i \notin C$, then the law of total probability and the Markov property imply that

$$\begin{aligned}
h_i^C &= \mathbb{P}_i\left(X_n \in C \text{ for some } n < \infty\right) \\
&= \sum_{j \in E} \mathbb{P}_i\left(X_n \in C \text{ for some } n < \infty | X_1 = j\right) \cdot \mathbb{P}_i\left(X_1 = j\right) \\
&= \sum_{j \in E} \mathbb{P}_j\left(X_n \in C \text{ for some } n < \infty\right) \cdot p_{ij} \\
&= \sum_{j \in E} p_{ij} h_j^C .
\end{aligned}$$

□

If C is closed and $i \in C$, then $p_{ij} = 0$ for any $j \notin C$. Since $h_j^C = 1$ for all $j \in C$, this implies that

$$h_i^C \;=\; 1 \;=\; \sum_{j \in E} p_{ij} \;=\; \sum_{j \in C} p_{ij} \;=\; \sum_{j \in C} p_{ij} h_j^C \;=\; \sum_{j \in E} p_{ij} h_j^C ,$$

which shows that the second identity asserted in Theorem 2.4 holds even when $i \in C$. In particular, this shows that the (column) vector of absorption probabilities h^C is a right eigenvector of the transition matrix P corresponding to eigenvalue 1, i.e.,

$$P h^C = h^C. \tag{2.4}$$

A similar approach can be used to derive a linear system of equations for the mean absorption times of a Markov chain.

Theorem 2.5. *The vector of mean hitting times $k^C = (k_1^C, k_2^C, \cdots)$ is the minimal non-negative solution of the system of linear equations,*

$$\begin{cases} k_i^C = 0 & \text{if } i \in C \\ k_i^C = 1 + \sum_{j \in E} p_{ij} k_j^C & \text{if } i \notin C \end{cases}$$

Proof. We again give just an outline of the proof that the mean absorption times solve this system of equations. Clearly, $k_i^C = 0$ whenever $i \in C$. On the other hand, if $i \notin C$, then by conditioning on the location of the chain at time 1, we have

$$\begin{aligned}
k_i^C &= 1 + \sum_{j \in E} \mathbb{E}_i\left[H^C | X_1 = j\right] \cdot \mathbb{P}_i\left(X_1 = j\right) \\
&= 1 + \sum_{j \in E} p_{ij} k_j^C ,
\end{aligned}$$

where the last identity holds because X is a Markov process.

□

2.2.3 Stationary Distributions

When a Markov chain X has absorbing states, we can use Theorem 2.5 to predict where the chain is likely to have settled after a sufficiently long period of time. In other words, there is a sense in which a chain with absorbing states becomes progressively less random as time goes on. For example, death is an absorbing state in demographic models and we can, for instance, predict that any human being is exceedingly likely to be dead 150 years after their birth, whatever shape their life takes in between birth and death.

In contrast, when a Markov chain has no absorbing states, then it is usually impossible to predict which state will be occupied by X_n when n is large, even if we know the initial state exactly. Indeed, many chains have the property that, as time goes on, all information about the initial location X_0 is progressively lost, i.e., in effect, the chain gradually forgets where it has been. Surprisingly, in these cases, it may still be possible to say something meaningful about the distribution of a chain that is known to have been running for a long time period even if we have no knowledge of the initial state. The key idea is contained in the next definition.

Definition 2.7. *A distribution π on E is said to be a* ***stationary distribution*** *for a DTMC X with transition matrix P if*

$$\pi P = \pi. \tag{2.5}$$

In the language of matrix theory, a distribution π is stationary for a DTMC X with transition matrix P if and only if the corresponding row vector π is a left eigenvector for P corresponding to the eigenvalue 1. Compare this with equation (2.4), which asserts that any vector of absorption probabilities is a right eigenvector corresponding to eigenvalue 1. Although this algebraic condition is useful when trying to identify stationary distributions, the next theorem gives more insight into their probabilistic meaning.

Theorem 2.6. *Suppose that π is a stationary distribution for a Markov chain $X = (X_n; n \geq 0)$ with transition matrix P. If π is the distribution of X_0, then π is also the distribution of X_n for all $n \geq 0$.*

Proof. According to Theorem 4.3, the distribution of X_n is equal to

$$\pi P^n = (\pi P)\, P^{n-1} = \pi P^{n-1} = \cdots = \pi.$$

□

In other words, any stationary distribution of a Markov chain is also time-invariant: if ever the process has π as its distribution, then it will retain this distribution for all time. For this reason, stationary distributions are also called equilibrium distributions or steady-state distributions, and they play a similar role in the theory of Markov chains to that played by stationary solutions of deterministic dynamical systems. One difference, of course, is that if we observe a stationary Markov process, then stationarity will be lost as soon as we have any additional information about the chain: even if the initial distribution is π, the conditional distribution of X_n given some information about the value of X_n will typically not be π.

Although we might hope that every Markov chain would have a unique stationary distribution, unfortunately this is not true in general: stationary distributions need not exist and, if they do exist, they need not be unique.

Example 2.5. *Let $Z_1, Z_2, \cdots$ be a sequence of i.i.d. Bernoulli random variables with success probability $p > 0$ and let $X = (X_n; n \geq 0)$ be the random walk defined in Example 2.2: set $X_0 = 0$ and*

$$X_n = Z_1 + \cdots + Z_n.$$

Then X_n tends to infinity almost surely and X has no stationary distribution on the integers.

The problem that arises in this example is that as n increases, the probability mass 'runs off to infinity.' This cannot happen when the state space is finite and, in fact, it can be shown that:

Theorem 2.7. *Any DTMC X on a finite state space E has at least one stationary distribution.*

Proof. Let P be the transition matrix of X. Since P is stochastic, 1 is the dominant eigenvalue of P and then the Perron-Frobenius theorem tells us that there is a left eigenvector corresponding to 1 with non-negative entries. Normalizing this vector so that the entries sum to 1 supplies the stationary distribution π. □

Recall from Definition 2.5 that a Markov chain is irreducible if all states in the state space are communicating, i.e., if the chain can move from any state i to any other state j in some finite period of time. Under certain additional conditions, one can expect the distribution of an irreducible Markov chain starting from any initial distribution to tend to the same unique stationary distribution as time passes. Sufficient conditions for this to be true are given in the next theorem, but we first need to introduce the following concept.

Definition 2.8. *A DTMC X with values in E and transition matrix P is said to be* ***aperiodic*** *if for every state $i \in E$, $p_{ii}^{(n)} > 0$ for all sufficiently large n.*

Example 2.6. *As the name suggests, an aperiodic chain is one in which there are no periodic orbits. For an example of a periodic Markov chain, take $E = \{1, 2\}$ and let X be the chain with transition matrix*

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

If $X_0 = 1$, then $X_{2n} = 1$ and $X_{2n+1} = 2$ for all $n \geq 0$, i.e., the chain simply oscillates between the values 1 and 2 forever. Also, although $\pi = (1/2, 1/2)$ is a stationary distribution for X, if we start the process with any distribution $\nu \neq \pi$, then the distribution of X_n will never approach π. Aperiodicity rules out the possibility of such behavior.

Theorem 2.8. *Suppose that P is irreducible and aperiodic, and that π is a stationary distribution for P. If μ is a distribution on E and $X = (X_n; n \geq 0)$ is a DTMC with transition matrix P and initial distribution μ, then*

$$\lim_{n \to \infty} \mathbb{P}(X_n = j) = \pi_j$$

for every $j \in E$. In particular,

$$\lim_{n \to \infty} p_{ij}^{(n)} = \pi_j$$

for all $i, j \in E$.

In other words, any irreducible and aperiodic DTMC X has at most one stationary distribution and, if such a distribution π exists, then the distribution of the chain will tend to π no matter what the initial distribution was. Continuing the analogy with deterministic processes, such a distribution is analogous to a globally-attracting stationary solution of a dynamical system. In practice, the existence of such a stationary distribution means that if a system is modeled by such a Markov chain and if we have no prior knowledge of the state of the system, then it may be reasonable to assume that the distribution of the state of the system is at equilibrium. For example, in population genetics, it has been common practice to assume that the distribution of allele frequencies is given by the stationary distribution of a Markov process when analyzing sequence data.

Example 2.7. ***Discrete-time birth and death processes***

Let $E = \{0, \cdots, N\}$ for some $N \geq 1$ and suppose that $X = (X_n; n \geq 0)$ is the DTMC with transition matrix $P = (p_{ij})$ given by

$$P = \left(\begin{array}{cccccccc} 1-b_0 & b_0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ d_1 & 1-(b_1+d_1) & b_1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & d_2 & 1-(b_2+d_2) & b_2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & & & & & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & d_{N-1} & 1-(b_{N-1}+d_{N-1}) & b_{N-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & d_N & 1-d_N \end{array} \right).$$

X is said to be a (discrete-time) birth and death process and has the following interpretation if we think of X_n as the number of individuals present in the population at time n. If $X_n = k \in \{1, \cdots, N-1\}$, then there are three possibilities for X_{n+1}. First, with probability b_k, one of the individuals gives birth to a single offspring causing the population size to increase to $X_{n+1} = k+1$. Secondly, with probability d_k, one of the individuals dies and the population size decreases to $X_{n+1} = k-1$. Finally, with probability $1 - b_k - d_k$, no individual reproduces or dies during that time step and so the population size remains at $X_{n+1} = k$. The case $X_n = 0$ needs a separate interpretation since clearly neither death nor birth can occur in the absence of any individuals. One possibility is to let b_0 be the probability that a new individual migrates into the region when the population has gone extinct. Also, in this model we are assuming that when $X_n = N$, density dependence is so strong that no individual can reproduce. (It is also possible to take $N = \infty$, in which case this is not an issue.)

If all of the birth and death probabilities b_k, d_k that appear in P are positive, then X is an irreducible, periodic DTMC defined on a finite state space and so it follows from Theorems 2.7 and 2.8 that X has a unique stationary distribution π that satisfies the equation $\pi P = \pi$. This leads to the following system of equations,

$$\begin{aligned} (1-b_0)\pi_0 + d_1\pi_1 &= \pi_0 \\ b_{k-1}\pi_{k-1} + (1-b_k-d_k)\pi_k + d_{k+1}\pi_{k+1} &= \pi_k \qquad k = 1, \cdots, N-1 \\ b_{N-1}\pi_{N-1} + (1-d_N)\pi_N &= \pi_N, \end{aligned}$$

which can be rewritten in the form

$$\begin{aligned} -b_0\pi_0 + d_1\pi_1 &= 0 \\ b_{k-1}\pi_{k-1} - (b_k+d_k)\pi_k + d_{k+1}\pi_{k+1} &= 0 \qquad k = 1, \cdots, N-1 \\ b_{N-1}\pi_{N-1} - d_N\pi_N &= 0. \end{aligned}$$

The first equation can be rewritten as $d_1\pi_1 = b_0\pi_0$, which implies that

$$\pi_1 = \frac{b_0}{d_1}\pi_0.$$

Taking $k = 1$, we have

$$b_0\pi_0 - d_1\pi_1 - b_1\pi_1 + d_2\pi_2 = 0.$$

However, since the first two terms cancel, this reduces to

$$-b_1\pi_1 + d_2\pi_2 = 0.$$

which shows that

$$\pi_2 = \frac{b_1}{d_2}\pi_1 = \frac{b_1 b_0}{d_2 d_1}\pi_0.$$

Continuing in this way, we find that

$$\pi_k = \frac{b_{k-1}}{d_k}\pi_{k-1} = \left(\frac{b_{k-1}\cdots b_0}{d_k\cdots d_1}\right)\pi_0$$

for $k = 1, \cdots, N$. All that remains to be determined is π_0. However, since π is a probability distribution on E, the probabilities must sum to one, which gives the condition

$$1 = \sum_{k=0}^{N}\pi_k = \pi_0\left(1 + \sum_{k=1}^{N}\frac{b_{k-1}\cdots b_0}{d_k\cdots d_1}\right).$$

This forces

$$\pi_0 = \left(1 + \sum_{k=1}^{N}\frac{b_{k-1}\cdots b_0}{d_k\cdots d_1}\right)^{-1} \tag{2.6}$$

and then

$$\pi_k = \left(\frac{b_{k-1}\cdots b_0}{d_k\cdots d_1}\right)\left(1 + \sum_{j=1}^{N}\frac{b_{j-1}\cdots b_0}{d_j\cdots d_1}\right)^{-1} \tag{2.7}$$

for $k = 1, \cdots, N$.

Chapter 3

Model Formulation

3.1 Definitions and Notation

Recall that a **Markov decision process** (MDP) consists of a stochastic process along with a decision maker that observes the process and is able to select actions that influence its development over time. Along the way, the decision maker receives a series of rewards that depend on both the actions chosen and the states occupied by the process. A MDP can be characterized mathematically by a collection of five objects

$$\{T, S, A_s, p_t(\cdot|s,a), r_t(\cdot|s,a) : t \in T, s \in S, a \in A_s\}$$

which are described below.

(1) $T \subset [0, \infty)$ is the **set of decision epochs**, which are the points in time when the external observer decides on and then executes some action. We will mainly consider processes with countably many decision epochs, in which case T is said to be discrete and we will usually take $T = \{1, 2, \cdots, N\}$ or $T = \{1, 2, \cdots\}$ depending on whether T is finite or countably infinite. Time is divided into **time periods** or **stages** in discrete problems and we assume that each decision epoch occurs at the beginning of a time period. A MDP is said to have either a **finite horizon** or **infinite horizon**, respectively, depending on whether the least upper bound of T (i.e., the supremum) is finite or infinite. If $T = \{1, \cdots, N\}$ is finite, we will stipulate that no decision is taken in the final decision epoch N.

(2) S is the set of states that can be assumed by the process and is called the **state space**. S can be any measurable set, but we will mainly be concerned with processes that take values in state spaces that are either countable or which are compact subset of $\mathbb{R}^n$.

(3) For each state $s \in S$, A_s is the set of actions that are possible when the state of the system is s. We will write $A = \cup_{s \in S} A_s$ for the set of all possible actions and we will usually assume that each A_s is either a countable set or a compact subset of $\mathbb{R}^n$.

Actions can be chosen either deterministically or randomly. To describe the second possibility, we will write $\mathcal{P}(A_s)$ for the set of probability distributions on A_s, in which case a randomly chosen action can be specified by a probability distribution $q(\cdot) \in \mathcal{P}(A_s)$, e.g., if A_s is discrete, then an action $a \in A_s$ will be chosen with probability $q(a)$.

(4) If T is discrete, then we must specify how the state of the system changes from one decision epoch to the next. Since we are interested in Markov decision processes, these changes are chosen at random from a probability distribution $p_t(\cdot|s,a)$ on S that may depend on the current time t, the current state of the system s, and the action a chosen by the observer.

(5) As a result of choosing action a when the system is in state s at time t, the observer receives

a **reward** $r_t(s,a)$ which can be regarded as a profit when positive or as a cost when negative. We assume that the rewards can be calculated, at least in principle, by the observer prior to selecting a particular action. We will also consider problems in which the reward obtained at time t can be expressed as the expected value of a function $r_t(s_t,a,s_{t+1})$ that depends on the state of the system at that time and at the next time, e.g.,

$$r_t(s,a) = \sum_{j\in S} p_t(j|s,a) r_t(s,a,j)$$

if S is discrete. (If S is uncountable, then we need to replace the sum by an integral and the transition probabilities by transition probability densities.) If the MDP has a finite horizon N, then since no action is taken in the last period, the value earned in this period will only depend on the final state of the system. This value will be denoted $r_N(s)$ and is sometimes called the **salvage value** or **scrap value**.

Recall that a **decision rule** d_t tells the observer how to choose the action to be taken in a given decision epoch $t \in T$. A decision rule is said to be **Markovian** if it only depends on the current state of the system, i.e., d_t is a function of s_t. Otherwise, the decision rule is said to be **history-dependent**, in which case it may depend on the entire history of states and actions from the first decision epoch through the present. Such histories will be denoted $h_t = (s_1, a_1, s_2, a_2, \cdots, s_{t-1}, a_{t-1}, s_t)$ and satisfy the recursion

$$h_t = (h_{t-1}, a_{t-1}, s_t).$$

We will also write H_t for the set of all possible histories up to time t. Notice that the action taken in decision epoch t is not included h_t. Decision rules can also be classified as either **deterministic**, in which case they prescribe a specific action to be taken, or as **randomized**, in which case they prescribe a probability distribution on the action set A_s and the action is chosen at random using this distribution. Combining these two classifications, there are four classes of decision rules, Markovian and deterministic (MD), Markovian and randomized (MR), history-dependent and deterministic (HD), and history-dependent and randomized (HR), and we will denote the sets of decision rules of each type available at time t by D_t^K, where $K = MD, MR, HD, HR$. In each case, a decision rule is just a function from S or H_t into A or $\mathcal{P}(A)$:

- if $d_t \in D_t^{MD}$, then $d_t : S \to A$;
- if $d_t \in D_t^{MR}$, then $d_t : S \to \mathcal{P}(A)$;
- if $d_t \in D_t^{HD}$, then $d_t : H_t \to A$;
- if $d_t \in D_t^{HR}$, then $d_t : H_t \to \mathcal{P}(A)$.

Since every Markovian decision rule is history-dependent and every deterministic rule can be regarded as a randomized rule (where the randomization is trivial), the following inclusions hold between these sets:

$$\begin{gathered} D_t^{MD} \subset D_t^{MR} \subset D_t^{HR} \\ D_t^{MD} \subset D_t^{HD} \subset D_t^{HR}. \end{gathered}$$

In particular, Markovian deterministic rules are the most specialized, whereas history-dependent randomized rules are the most general.

A **policy** π is a sequence of decision rules $d_1, d_2, d_3, \cdots$ for every decision epoch and a policy is said to be Markovian or history-dependent, as well as deterministic or randomized, if the decision rules specified by the policy have the corresponding properties. We will write Π^K, with $K = MD, MR, HD, HR$, for the sets of policies of these types. A policy is said to be **stationary** if the same decision rule is used in every epoch. In this case, $\pi = (d, d, \cdots)$ for some Markovian decision rule d and we denote this policy by d^∞. Stationary policies can either be deterministic or randomized and the sets of stationary policies of either type are denoted Π^{SD} or Π^{SR}, respectively.

Because a Markov decision process is a stochastic process, the successive states and actions realized by that process form a sequence of random variables. We will introduce the following notation for these variables. For each $t \in T$, let $X_t \in S$ denote the state occupied by the system at time t and let $Y_t \in A_s$ denote the action taken at the start of that time period. It follows that any discrete-time process can be represented as a sequence of such variables $X_1, Y_1, X_2, Y_2, X_3, \cdots$. Likewise, we will define the history process $Z = (Z_1, Z_2, \cdots)$ by setting $Z_1 = s_1$ and

$$Z_t = (s_1, a_1, s_2, a_2, \cdots, s_t).$$

The initial distribution of a MDP is a distribution on S and will be denoted $\mathbb{P}_1(\cdot)$. Furthermore, any randomized history-dependent policy $\pi = (d_1, d_2, \cdots, d_{N-1})$ induces a probability distribution $\mathbb{P}^\pi$ on the set of all possible histories $(s_1, a_1, s_2, a_2, \cdots, a_{N-1}, s_N)$ according to the following identities:

$$\begin{aligned}
\mathbb{P}^\pi\{X_1 = s_1\} &= \mathbb{P}_1(s_1), \\
\mathbb{P}^\pi\{Y_t = a | Z_t = h_t\} &= q_{d_t(h_t)}(a), \\
\mathbb{P}^\pi\{X_{t+1} = s | Z_t = (h_{t-1}, a_{t-1}, s_t), Y_t = a_t\} &= p_t(s|s_t, a_t).
\end{aligned}$$

Here $q_{d_t(h_t)}$ is the probability distribution on A_{s_t} which the decision rule d_t uses to randomly select the next action a_t when the history of the system up to time t is given by $h_t = (h_{t-1}, a_{t-1}, s_t)$. The probability of any particular sample path $(s_1, a_1, \cdots, a_{N-1}, s_N)$ can be expressed as a product of such probabilities:

$$\begin{aligned}
\mathbb{P}^\pi(s_1, a_1, \cdots, a_{N-1}, s_{N-1}) &= \mathbb{P}_1(s_1) q_{d_1(s_1)}(a_1) p_1(s_2|s_1, a_1) q_{d_2(h_2)}(a_2) \\
&\quad \cdots q_{d_{N-1}(h_{N-1})}(a_{N-1}) p_{N-1}(s_N|s_{N-1}, a_{N-1}).
\end{aligned}$$

If π is a Markovian policy, then the process $X = (X_t : t \in T)$ is a Markov process, as is the process $(X_t, r_t(X_t, Y_t) : t \in T)$, which we refer to as a **Markov reward process**. The Markov reward process tracks the states occupied the system as well as the sequence of rewards received.

3.2 Example: A One-Period Markov Decision Problem

By way of illustration, we describe a one-period MDP with $T = \{1, 2\}$ and $N = 2$. We will assume that that the state space S is finite and also that the action sets A_s are finite for each $s \in S$. Let $r_1(s, a)$ be the reward obtained when the system is in state s and action a is taken at the beginning of stage 1, and let $v(s')$ be the terminal reward obtained when the system is in state s' at the end of this stage. Our objective is to identify policies that maximize the sum of $r_1(s, a)$ and the expected terminal reward. Since there is only one period, a policy consists of a single decision rule and every history-dependent decision rule is also Markovian. (Here, as throughout these lectures, we are assuming that the process begins at time $t = 1$, in which case there is no prior history to be considered when deciding how to act during the first decision epoch.)

If the observer chooses a deterministic policy $\pi = (d_1)$ and $a' = d_1(s)$, then the total expected reward when the initial system state is s is equal to

$$R(s,a') \equiv r_1(s,a') + \mathbb{E}_s^\pi[v(X_2)] = r_1(s,a') + \sum_{j\in S} p_1(j|s,a')v(j),$$

where $p_1(j|s,a')$ is the probability that the system occupies state j at time $t = 2$ given that it was in state s at time $t = 1$ and action a' was taken in this decision epoch. The observer's problem can be described as follows: for each state $s \in S$, find an action $a^* \in A_s$ that maximizes the expected total reward, i.e., choose a_s^* so that

$$R(s,a_s^*) = \max_{a\in A_s} R(s,a).$$

Because the state space and action sets are finite, we know that there is at least one action a^* that achieves this maximum, although it is possible that there may be more than one. It follows that an optimal policy $\pi = (d_1^*)$ can be constructed by setting $d_1^*(s) = a_s^*$ for each $s \in S$. The optimal policy will not be unique if there is a state $s \in S$ for which there are multiple actions that maximize the expected total reward.

The following notation will sometimes be convenient. Suppose that X is a set and that $g : X \to \mathbb{R}$ is a real-valued function defined on X. We will denote the set of points in X at which g is maximized by

$$\arg\max_{x\in X} g(x) \equiv \{x' \in X : g(x') \geq g(y) \text{ for all } y \in X\}.$$

If g fails to have a maximum on X, we will set $\arg\max_{x\in X} g(x) = \emptyset$. For example, if $X = [-1,1]$ and $g(x) = x^2$, then

$$\arg\max_{x\in[-1,1]} g(x) = \{-1,1\}$$

since the maximum of g on this set is equal to 1 and -1 and 1 are the two points where g achieves this maximum. In contrast, if $X = (-1,1)$ and $g(x) = x^2$, then

$$\arg\max_{x\in(-1,1)} g(x) = \emptyset$$

since g has no maximum on $(-1,1)$. With this notation, we can write

$$a_s^* \in \arg\max_{a'\in A_s} R(s,a').$$

We next consider randomized decision rules. If the initial state of the system is s and the observer chooses action $a \in A_s$ with probability $q(a)$, then the expected total reward will be

$$\mathbb{E}_q[R(s,\cdot)] = \sum_{a\in A_s} q(a)R(s,a).$$

However, since

$$\max_{q\in\mathcal{P}(A_s)} \left\{ \sum_{a\in A_s} q(a)R(s,a) \right\} = \max_{a'\in A_s} R(s,a'),$$

it follows that a randomized rule can at best do as well as the best deterministic rule. In fact, a randomized rule with $d(s) = q_s(\cdot)$ will do as well as the best deterministic rule if and only if for each $s \in S$,

$$\sum_{a^*\in\arg\max_{A_s} R(s,a^*)} q_s(a^*) = 1.$$

In other words, the randomized rule should always select one of the actions that maximizes the expected total reward.

Chapter 4

Examples of Markov Decision Processes

4.1 A Two-State MDP

We begin by considering a very simple MDP with a state space containing just two elements; this toy model will be used for illustrative purposes throughout the course. The constituents of this model are described below.

- Decision epochs: $T = \{1, 2, \cdots, N\}, N \leq \infty$.
- States: $S = \{s_1, s_2\}$.
- Actions: $A_{s_1} = \{a_{11}, a_{12}\}, A_{s_2} = \{a_{21}\}$.
- Rewards:

$$\begin{array}{lll} r_t(s_1, a_{11}) = 5 & r_t(s_1, a_{12}) = 10 & r_t(s_2, a_{21}) = -1 \\ r_N(s_1) = 0 & r_N(s_2) = -1 & \end{array}$$

- Transition probabilities:

$$\begin{array}{ll} p_t(s_1|s_1, a_{11}) = 0.5, & p_t(s_2|s_1, a_{11}) = 0.5 \\ p_t(s_1|s_1, a_{12}) = 0, & p_t(s_2|s_1, a_{12}) = 1 \\ p_t(s_1|s_2, a_{21}) = 0, & p_t(s_2|s_2, a_{21}) = 1 \end{array}$$

In words, when the system is in state s_1, the observer can either choose action a_{11}, in which case they receive an immediate reward of 5 units and the system either remains in that state with probability 0.5 or moves to state s_2 with probability 0.5, or they can choose action a_{12}, in which case they receive an immediate reward of 10 units and the system is certain to transition to state s_2. In contrast, s_2 is an absorbing state for this process and the observer incurs a cost of one unit in each time step. Notice that action a_2 has no effect on the state of the system or on the reward received.

We next consider some examples that illustrate the different types of policies introduced in the last chapter. We will assume that $N = 3$ and represent the policies by $\pi^K = (d_1^K, d_2^K)$, where $K = MD$, MR, HD or HR.

A deterministic Markovian policy: π^{MD}

$$\begin{array}{lll}
\textbf{Decision epoch 1:} & d_1^{MD}(s_1) = a_{11}, & d_1^{MD}(s_2) = a_{21}, \\
\textbf{Decision epoch 2:} & d_2^{MD}(s_1) = a_{12}, & d_2^{MD}(s_2) = a_{21}.
\end{array}$$

A randomized Markovian policy: π^{MR}

$$\begin{array}{lll}
\textbf{Decision epoch 1:} & q_{d_1^{MR}(s_1)}(a_{11}) = 0.7, & q_{d_1^{MR}(s_1)}(a_{12}) = 0.3 \\
 & q_{d_1^{MR}(s_2)}(a_{21}) = 1.0; & \\
\textbf{Decision epoch 2:} & q_{d_2^{MR}(s_1)}(a_{11}) = 0.4, & q_{d_2^{MR}(s_1)}(a_{12}) = 0.6 \\
 & q_{d_2^{MR}(s_2)}(a_{21}) = 1.0. &
\end{array}$$

This model has the unusual property that the set of history-dependent policies is identical to the set of Markovian policies. This is true for two reasons. First, because the system can only remain in state s_1 if the observer chooses action a_{11}, there is effectively only one sample path ending in s_1 in any particular decision epoch. Secondly, although there are multiple paths leading to state s_2, once the system enters this state, the observer is left with no choice regarding its actions. To illustrate history-dependent policies, we will modify the two-state model by adding a third action a_{13} to A_{1,s_1} which causes the system to remain in state s_1 with probability 1 and provides a zero reward $r_t(s_1, a_{13}) = 0$ for every $t \in T$. With this modification, there are now multiple histories which can leave the system in state s_1, e.g., (s_1, a_{11}, s_1) and (s_1, a_{13}, s_1).

A deterministic history-dependent policy: π^{HD}

$$\begin{array}{lll}
\textbf{Decision epoch 1:} & d_1^{HD}(s_1) = a_{11}, & d_1^{MD}(s_2) = a_{21}, \\
\textbf{Decision epoch 2:} & d_2^{HD}(s_1, a_{11}, s_1) = a_{13}, & d_2^{HD}(s_1, a_{11}, s_2) = a_{21}, \\
 & d_2^{HD}(s_1, a_{12}, s_1) = \text{undefined}, & d_2^{HD}(s_1, a_{11}, s_2) = a_{21}, \\
 & d_2^{HD}(s_1, a_{13}, s_1) = a_{11}, & d_2^{HD}(s_1, a_{13}, s_2) = \text{undefined}, \\
 & d_2^{HD}(s_2, a_{21}, s_1) = \text{undefined}, & d_2^{HD}(s_2, a_{21}, s_2) = a_{21}.
\end{array}$$

We leave the decision rules undefined when evaluated on histories that cannot occur, e.g., the history (s_1, a_{12}, s_1) will never occur because the action a_{12} forces a transition from state s_1 to state s_2. Randomized history-dependent policies can be defined in a similar manner.

4.2 Single-Product Stochastic Inventory Control

Suppose that a manager of a warehouse is responsible for maintaining the inventory of a single product and that additional stock can be ordered from a supplier at the beginning of each month. The manager's goal is to maintain sufficient stock to fill the random number of orders that will arrive each month, while limiting the costs of ordering and holding inventory. This problem can be modeled by a MDP which we formulate using the following simplifying assumptions.

1. Stock is ordered and delivered at the beginning of each month.

2. Demand for the item arrives throughout the month, but orders are filled on the final day of the month.

3. If demand exceeds inventory, the excess customers go to alternative source, i.e., unfilled orders are lost.

4. The revenues, costs and demand distribution are constant over time.

5. The product is sold only in whole units.

6. The warehouse has a maximum capacity of M units.

We will use the following notation. Let s_t denote the number of units in the warehouse at the beginning of month t, let a_t be the number of units ordered from the supplier at the beginning of that month, and let D_t be the random demand during month t. We will assume that the random variables $D_1, D_2, \cdots$ are independent and identically-distributed with distribution $p_j = \mathbb{P}(D_t = j)$. Then the inventory at decision epoch $t+1$ is related to the inventory at decision epoch t through the following equation:

$$s_{t+1} = \max\{s_t + a_t - D_t, 0\} \equiv [s_t + a_t - D_t]^+ .$$

The revenue and costs used to calculate the reward function are evaluated at the beginning of each month and are called **present values**. We will assume that the cost of ordering u units is equal to the sum of a fixed cost $K > 0$ for placing orders and a variable cost $c(u)$ that increases with the number of units ordered, i.e.,

$$O(u) = \begin{cases} 0 & \text{if } u = 0 \\ \\ K + c(u) & \text{if } u > 0. \end{cases}$$

Likewise, let $h(u)$ be a non-decreasing function that specifies the cost of maintaining an inventory of u units for a month and let $g(u)$ be the value of any remaining inventory in the last decision epoch of a finite horizon model. Finally, let $f(j)$ be the revenue earned from selling j units of inventory and assume that $f(0) = 0$. Assuming that the revenue is only gained at the end of the month when the month's orders are filled, the reward depends on the state of the system at the start of the next decision epoch:

$$r_t(s_t, a_t, s_{t+1}) = -O(a_t) - h(s_t + a_t) + f(s_t + a_t - s_{t+1}).$$

However, since s_{t+1} is still unknown during decision epoch t, it will be more convenient to work with the expected present value at the beginning of the month of the revenue earned throughout that month. This will be denoted $F(u)$, where u is the number of units present at the beginning of month t, and is equal to

$$F(u) = \sum_{j=0}^{u-1} p_j f(j) + q_u f(u),$$

where

$$q_u = \sum_{j=u}^{\infty} p_j = \mathbb{P}(D_t \geq u)$$

is the probability that the demand equals or exceeds the available inventory.

The MDP can now be formulated as follows:

- Decision epochs: $T = \{1, 2, \cdots, N\}, N \leq \infty$;
- States: $S = \{0, 1, \cdots, M\}$;
- Actions: $A_s = \{0, 1, \cdots, M - s\}$;

- Expected rewards:

$$\begin{aligned} r_t(s,a) &= F(s+a) - O(a) - h(s+a), \quad t = 1, \cdots, N-1; \\ r_N(s) &= g(s) \end{aligned}$$

- Transition probabilities:

$$p_t(j|s,a) = \begin{cases} 0 & \text{if } M \geq j \geq s+a \\ p_{s+a-j} & \text{if } M \geq s+a \geq j > 0 \\ q_{s+a} & \text{if } M \geq s+a; j = 0. \end{cases}$$

Suppose that $\Sigma > \sigma > 0$ are positive numbers. A (σ, Σ) policy is an example of a stationary deterministic policy which implements the following decision rule in every decision epoch:

$$d_t(s) = \begin{cases} 0 & \text{if } s \geq \sigma \\ \Sigma - s & \text{if } s < \sigma. \end{cases}$$

In other words, sufficient stock is ordered to raise the inventory to Σ units whenever the inventory level at the beginning of a month is less than σ units. Σ is said to be the **target stock** while $\Sigma - \sigma$ is the **minimum fill**.

4.3 Deterministic Dynamic Programs

A **deterministic dynamic program** (DDP) is a type of Markov decision process in which the choice of an action determines the subsequent state of the system with certainty. The new state occupied by the system following an action is specified by a **transfer function**, which is a mapping $\tau_t : S \times A_s \rightarrow S$. Thus $\tau_t(s,a) \in S$ is the new state that will be occupied by the system at time $t+1$ when the previous state was s and the action selected was $a \in A_s$. A DDP can be formulated as a MDP by using the transfer function to define a degenerate transition probability:

$$p_t(j|s,a) = \begin{cases} 1 & \text{if } \tau_t(s,a) = j \\ 0 & \text{if } \tau_t(s,a) \neq j. \end{cases}$$

As in the previous examples, the reward earned in time epoch t will be denoted $r_t(s,a)$.

When the total reward is used to compare policies, every DDP with finite T, S, and A is equivalent to a shortest or longest route problem through an acyclic finite directed graph. Indeed, any such DDP can be associated with an acyclic finite directed graph with the following sets of vertices and edges:

$$\begin{aligned} V &= \{(s,t) : s \in S, t \in T\} \cup \{O, D\} \\ E &= \{((s_1,t),(s_2,t+1)) : \tau_t(s_1,a) = s_2 \text{ for some } a \in A_{s_1}\} \cup \{(O,(s,1)) : s \in S\} \cup \\ &\quad\ \{((s,N),D) : s \in S\}. \end{aligned}$$

Here, O and D are said to be the origin and destination of the graph and $(v_1, v_2) \in E$ if and only if there is a directed edge connecting v_1 to v_2. Thus, apart from O and D, each vertex corresponds to a state s and a time t and a directed edge connects any vertex (s_1, t) to a vertex $(s_2, t+1)$ if and only if there is an action $a \in A_{s_1}$ such that this action changes the state of the system from s_1 at time t to s_2 at time $t+1$. In addition, there are directed edges connecting the origin to each of the possible initial states $(s, 1)$ as well as directed edges connecting each possible terminal state (s, N) to the destination. Weights are assigned to the edges as follows. Each edge connecting a vertex (s_1, t) to a vertex $(s_2, t+1)$ and corresponding to an action a is assigned a weight equal to the reward $r_t(s, a)$. Likewise, each edge connecting a terminal state (s, N) to D is assigned a weight equal to the reward $r_N(s)$. Finally, each edge connecting the origin to a possible initial state $(s, 1)$ is assigned a weight either equal to $L \gg 1$ if s is the actual initial state and equal to 0 otherwise. Choosing a policy that maximizes the total reward is equivalent to finding the longest route through this graph from the origin to the destination. As explained on p. 43 of Puterman (2005), the longest route problem is also central to **critical path analysis**.

Certain kinds of **sequential allocation models** can be interpreted as deterministic dynamic programs. In the general formulation of a sequential allocation model, a decision maker has a fixed quantity M of resources to be consumed or used in some manner over N periods. Let x_t denote the quantity of resources consumed in period t and suppose that $f(x_1, \cdots, x_N)$ is the **utility** (or reward) for the decision maker of the allocation pattern $(x_1, \cdots, x_N)$. The problem faced by the decision maker is to choose an allocation of the resources that maximizes the utility $f(x_1, \cdots, x_N)$ subject to the constraints

$$\begin{aligned} x_1 + \cdots + x_N &= M \\ x_t \geq 0, t &= 1, \cdots, N. \end{aligned}$$

Such problems are difficult to solve in general unless the utility function has special properties that can be exploited, either analytically or numerically, in the search for the global maximum. For example, the utility function $f(x_1, \cdots, x_N)$ is said to be **separable** if it can be written as a sum of univariate functions of the form

$$f(x_1, \cdots, x_N) = \sum_{t=1}^{N} g_t(x_i),$$

where $g_t : [0, M] \to \mathbb{R}$ is the utility gained from utilizing x_t resources during the t'th period and we assume that g_t is a non-decreasing function of its argument. In this case, the sequential allocation model can be formulated as a DDP/MDP as follows:

- Decision epochs: $T = \{1, \cdots, N\}$;
- States: $S = [0, M]$;
- Actions: $A_s = [0, s]$;
- Rewards: $r_t(s, a) = g_t(a)$;
- Transition probabilities:

$$p_t(j|s, a) = \begin{cases} 1 & \text{if } j = s - a \\ 0 & \text{otherwise .} \end{cases}$$

There are also stochastic versions of this problem in which either the utility or the opportunity to allocate resources in each time period is random.

4.4 Optimal Stopping Problems

We first formulate a general class of optimal stopping problems and then consider specific applications. In the general problem, a system evolves according to an uncontrolled Markov chain with values in a state space S' and the only actions available to the decision maker are to either do nothing, in which case a cost $f_t(s)$ is incurred if the system is in state s at time t, or to stop the chain, in which case a reward $g_t(s)$ is received. If the process has a finite horizon, then the decision maker received a reward $h(s)$ if the unstopped process is in state s at time N. Once the chain is stopped, there are no more actions or rewards. We can formulate this problem as a MDP as follows:

- Decision epochs: $T = \{1, \cdots, N\}, N \leq \infty$.
- States: $S = S' \cup \{\Delta\}$.
- Actions:
$$A_s = \begin{cases} \{C, Q\} & \text{if } s \in S' \\ \{C\} & \text{if } s = \Delta. \end{cases}$$
- Rewards:
$$\begin{aligned} r_t(s,a) &= \begin{cases} -f_t(s) & \text{if } s \in S', a = C \\ g_t(s) & \text{if } s \in S', a = Q, \\ 0 & \text{if } s = \Delta \end{cases} \qquad (t < N) \\ r_N(s) &= h(s). \end{aligned}$$
- Transition probabilities:
$$p_t(j|s,a) = \begin{cases} p_t(j|s) & \text{if } s, j \in S', a = C \\ 1 & \text{if } s \in S', j = \Delta, a = Q \\ 1 & \text{if } s = j = \Delta, a = C \\ 0 & \text{otherwise.} \end{cases}$$

Here Δ is an absorbing state (sometimes called a **cemetery state**) that is reached only if the decision maker decides to stop the chain. This is added to the state space S' of the original chain to give the extended state space S. While the chain is in S', two actions are available to the decision maker: either to continue the process (C) or to quit it (Q). Continuation allows the process to continue to evolve in S' according to the transition matrix of the Markov chain, while quitting causes the process to move to the cemetery state where it remains forever. The problem facing the decision maker is to find a policy that will specify when to quit the process in such a way that maximizes the difference between the gain at the stopping time and the costs accrued up to that time.

Example 4.1. ***Selling an asset.*** *Suppose that an investor owns an asset (such as a property) the value of which fluctuates over time and which must be sold by some final time N. Let X_t denote the price at time t, where time could be measured in days, weeks, months or any other discrete time unit, and assume that $X = (X_1, X_2, \cdots : t \geq 0)$ is a Markov process with values in the set $S' = [0, \infty)$. If the investor retains the asset in the t'th period, then she will incur a cost $f_t(s)$ that includes property taxes, advertising costs, etc. If the investor chooses to sell the property in the t'th decision epoch, then she will earn a profit $s - K(s)$ where $s = X_t$ is the value of the asset at that time and $K(s)$ is the cost of selling the asset at that price. If the property is still held at time N, then it must be sold at a profit (or loss) of $s - K(s)$, where $s = X_N$ is the final value.*

Optimal policies for this problem often take the form of ***control limit policies*** *which have decision rules of the following form:*

$$d_t(s) = \begin{cases} Q & \text{if } s \geq B_t \\ C & \text{if } s < B_t. \end{cases}$$

Here B_t is the control limit at time t and its value will usually change over time.

Example 4.2. ***The Secretary Problem.*** *Suppose that an employer seeks to fill a vacancy for which there are N candidates. Candidates are interviewed sequentially and following each interview, the employer must decide whether or not to offer the position to the current candidate. If the employer does not offer them the position, then that individual is removed from the pool of potential candidates and the next candidate is interviewed. There are many variations on this problem, but here we will assume that the candidates can be unambiguously ranked from best to worst and that the employer's objective is to maximize the probability of offering the job to the most-preferred candidate.*

We can reformulate the problem as follows. Suppose that a collection of N objects is ranked from 1 to N and that the lower the ranking the more preferred the object is. An observer encounters these objects sequentially, one at a time, in a random order, where each of the $N!$ possible orders is equally likely to occur. Although the absolute rank of each object is unknown to the observer, the relative ranks are known without error, e.g., the observer can tell whether the second object encountered is more or less preferred than the first object, but the observer cannot determine at that stage whether the first or second object is the best in the entire group of N if $N > 2$.

To cast this problem as a MDP, we will let $T = \{1, \cdots, N\}$ and $S' = \{0, 1\}$, where $s_t = 1$ indicates that the t'th object observed is the best one encountered so far and $s_t = 0$ indicates that a better object was encountered at an earlier time. If $t < N$, then the observer can either select the current object and quit (Q) or they can reject the current object and proceed to the next one in the queue (C). If $t = N$, then the observer is required to select the last object in the queue. We will assume that there are no continuation costs, i.e., $f_t(0) = f_t(1) = 0$, and we will take the reward at stopping $g_t(s)$ to be equal to be equal to the probability of choosing the best object in the group. Notice that the terminal reward $h(s) = g_N(s)$ is equal to 1 if $s = 1$ and 0 otherwise. Indeed, the N'th object will be the best one in the group if and only if it is the best one encountered. Furthermore, $g_t(0) = 0$ for all $t = 1, \cdots, N$, since if we select an object for which $s_t = 0$, then that means that we know that there is another object in the group that is better than the one that we are selecting. To calculate $g_t(1)$, we first observe that because we are equally likely to observe the objects in any order, the conditional probability that the t'th object is the best in the group given that it is the best in the first t objects observed is equal to the probability that the best object in the group is one of the first t objects encountered. Accordingly,

$$\begin{aligned} g_t(1) &= \mathbb{P}\big(\textit{the best object in the first } t \textit{ objects is the best overall}\big) \\ &= \frac{\textit{number of subsets of } \{1, \cdots, N\} \textit{ of size } t \textit{ containing } 1}{\textit{number of subsets of } \{1, \cdots, N\} \textit{ of size } t} \\ &= \frac{\binom{N-1}{t-1}}{\binom{N}{t}} = \frac{t}{N}. \end{aligned}$$

Again because of permutation invariance, the transition probabilities $p_t(j|s)$ do not depend on the current state. Instead, the probability that the $t+1$'st object encountered is the best amongst first $t+1$ objects is equal to $1/(t+1)$ and so

$$p_t(j|s) = \begin{cases} \frac{1}{t+1} & \textit{if } j = 1 \\ \frac{t}{t+1} & \textit{if } j = 0, \end{cases}$$

for both $s = 0$ and $s = 1$.

Similar problems arise in behavioral ecology, where an individual of the 'choosy sex' sequentially encounters individuals of the opposite sex and must choose whether to mate with the t'th individual or defer mating until a subsequent encounter with a different individual.

4.5 Controlled Discrete-Time Dynamical Systems

We consider a class of stochastic dynamical systems that are governed by a recursive equation of the following form:

$$s_{t+1} = f_t(s_t, u_t, w_t), \tag{4.1}$$

where $s_t \in S$ is the state of the system at time t, $u_t \in U$ is the control used at time t, and $w_t \in W$ is the disturbance of the system at time t. As above, S is called the state space of the system, but now we also have a **control set** U as well as a **disturbance set** W. Informally, we can think of the sequence $s_0, s_1, \cdots$ as a deterministic dynamical system that is perturbed both by a sequence of control actions $u_0, u_1, \cdots$ chosen by an external observer (the 'controller') as well as by a sequence of random disturbances $w_0, w_1, \cdots$ that are not under the control of that observer. To be concrete, we will assume that $S \subset \mathbb{R}^k$, $U \subset \mathbb{R}^m$, and $W \subset \mathbb{R}^n$, and that $f : S \times U \times W \to S$ maps triples (s, u, w) into S. We will also assume that the random disturbances are governed by a sequence of independent random variables $W_0, W_1, \cdots$ with values in the set W and we will let $q_t(\cdot)$ be the probability mass function or the probability density function of W_t. When the system is in state s_t and a control u_t is chosen from a set $U_s \subset U$ of admissible control in state s, then the controller will receive a reward $g_t(s, u)$. In addition, if the horizon is finite and the system terminates in state s_N at time N, then the controller will receive a terminal reward $g_N(s_N)$ that depends only on the final state of the system. We can formulate this problem as a MDP as follows:

- Decision epochs: $T = \{1, \cdots, N\}, N \leq \infty$.
- States: $S \subset \mathbb{R}^k$.
- Actions: $A_s = U_s \subset U \subset \mathbb{R}^m$.
- Rewards:

$$\begin{aligned} r_t(s,a) &= g_t(s,a), t < N \\ r_N(s) &= g_N(s). \end{aligned}$$

- Transition probabilities (discrete noise):

$$\begin{aligned} p_t(j|s,a) &= \mathbb{P}\big(j = f_t(s,a,W_t)\big) \\ &= \sum_{\{w \in W : j = f_t(s,a,w)\}} q_t(w). \end{aligned}$$

(If the disturbances are continuously-distributed, then the sum appearing in the transition probability must be replaced by an integral.)

The only substantial difference between a controlled discrete-time dynamical system and a Markov decision process is in the manner in which randomness is incorporated. Whereas MDPs are defined using transition probabilities that depend on the current state and action, the transition probabilities governing the behavior of the dynamical system corresponding to equation (4.1) must be derived from the distributions of the disturbance variables W_t. In the language of control theory, a decision rule is called a **feedback control** and **open loop control** is a decision rule which does not depend on the state of the system, i.e., $d_t(s) = a$ for all $s \in S$.

Example 4.3. ***Economic growth models.*** *We formulate a simple stochastic dynamical model for a planned economy in which capital can either be invested or consumed. Let $T = \{1, 2, \cdots\}$ be time measured in years and let $s_t \in S = [0, \infty)$ denote the capital available for investment in year t. (By requiring $s_t \geq 0$, we stipulate that the economy is debt-free.) After observing the level* *of capital available at time t, the planner chooses a level of consumption $u_t \in U_{s_t} = [0, s_t]$ and invests the remaining capital $s_t - u_t$. Consumption generates an immediate utility $\Psi_t(s_t)$ and the investment produces capital for the next year according to the dynamical equation*

$$s_{t+1} = w_t F_t(s_t - u_t)$$

where F_t determines the expected return on the investment and w_t is a non-negative random variable with mean 1 that accounts for disturbances caused by random shocks to the system, e.g., due to climate, political instability, etc.

4.6 Bandit Models

In a **bandit model** the decision maker observes K independent Markov processes $X^{(1)}, \cdots, X^{(K)}$ and at each decision epoch selects one of these processes to use. If the process i is chosen at time t when it is in state $s^i \in S^i$, then

1. the process i moves from state s^i to state j^i according to the transition probability $p^i_t(j^i|s^i)$;
2. the decision maker receives a reward $r^i_t(s^i)$;
3. all other processes remain in their current states.

Usually, the decision maker wishes to choose a sequence of processes in such a way that maximizes the total expected reward or some similar objective function. We can formulate this as a Markov decision process as follows:

- Decision epochs: $T = \{1, 2, \cdots, N\}, N \leq \infty$.
- States: $S = S^1 \times S^2 \times \cdots \times S^K$.
- Actions: $A_s = \{1, \cdots, K\}$.
- Rewards: $r_t((s^1, \cdots, S^K), i) = r^i_t(s^i)$.
- Transition probabilities:

$$p_t((u^1, \cdots, u^K)|(s^1, \cdots, s^K), i) = \begin{cases} p(j^i|s^i) & \text{if } u^i = j^i \text{ and } u^m = s^m \text{ for all } m \neq i \\ 0 & \text{if } u^m \neq s^m. \end{cases}$$

There are numerous variations on the basic bandit model, including **restless bandits** which allow the states of the unselected processes to change between decision epochs and **arm-acquiring bandits** which allow the number of processes to increase or decrease at each decision epoch.

Example 4.4. ***Gambling.*** *Suppose that a gambler in a casino can pay c units to pull the lever on one of K slot machines and that the i'th machine pays 1 unit with probability q_i and 0 units with probability $1 - q_i$. The values of the probabilities q_i are unknown, but the gambler gains information concerning the distribution of q_i each time that she chooses to play the game using the i'th machine. The gambler seeks to maximize her expected winnings, but to do so, she faces a tradeoff between exploiting the machine that appears to be best based on the information collected thus far and exploring other machines that might have higher probabilities of winning.*

This problem can be formulated as a ***multi-armed bandit problem*** *as follows. For each $i = 1, \cdots, K$, let S^i be the space of probability density functions defined on $[0, 1]$ and let $s^i_t = f \in S^i$*

if the density of the posterior distribution of the value of q_i given the data available up to time t is equal to $f(q)$. At time $t = 1$, the gambler begins by choosing a set of prior distributions for the values of the q_i's; these can be based on previous experience with these or similar slot machines, or they can be chosen to be 'uninformative'. At each decision epoch t, the gambler can choose to play the game using one of the K machines. If the i'th machine is chosen and if the posterior density for the value of q_i at that time is $s^i_t = f$, then the expected reward earned in that period is equal to

$$r_t((s^1, \cdots, s^K), i) = \mathbb{E}[Q] - c = \int_0^1 qf(q) - c,$$

where Q is a $[0, 1]$-valued random variable with density $f(q)$.

Let W be an indicator variable for the event that the gambler wins when betting with slot machine i at time t, i.e., $W = 1$ if the gambler wins and $W = 0$ otherwise. Then the distribution of q_i at decision epoch $t + 1$ depends on the value of W, and Bayes' formula can be used to calculate the posterior density f' of q_i given W:

$$f'(q_i|W) \quad = \quad \frac{q_i^W(1-q_i)^{1-W}f(q_i)}{\int_0^1 q^W(1-q)^W f(q)dq}.$$

In other words, if the gambler wins when using the i'th machine, then she updates the density of q_i from f to $q_i f(q_i)/\mathbb{E}_f[Q]$ and this occurs with probability $\mathbb{E}_f[Q]$. On the other hand, if the gambler loses when using this machine, then she updates the distribution to $(1 - q_i)f(q_i)/\mathbb{E}_f[1 - Q]$ and this occurs with probability $1-\mathbb{E}_f[Q]$. In the meantime, the distributions of the other probabilities $q_j, j \neq i$, do not change when i is played.

Due to the sequential nature of the updating, the state spaces used to represent the gambler's beliefs concerning the values of the q_i can be simplified to $S^i = \mathbb{N} \times \mathbb{N}$. For each $i = 1, \cdots, K$, let $f_{i,0}(q_i)$ be the density of the gambler's prior distribution for q_i and let $W_{i,t}$ and $L_{i,t}$ be the number of times that the gambler has either won or lost, respectively, using the i'th slot machine up to time t. Then the density of the posterior distribution for q_i given the sequence of wins and losses depends only on the numbers $W_{i,t}$ and $L_{i,t}$ and is equal to

$$f_{i,t}(q_i|W_{i,t}, L_{i,t}) = \frac{q_i^{W_{i,t}}(1-q_i)^{L_{i,t}}f_{i,0}(q_i)}{\int_0^1 q^{W_{i,t}}(1-q)^{L_{i,t}}f_{i,0}(q)dq}.$$

Variations of the multi-armed bandit model described in the previous example have been used to design sequential clinical trials. Patients sequentially enter a clinical trial and are assigned to one of K groups receiving different treatments. The aim is to find the most effective treatment while causing the least harm (or giving the greatest benefit) to the individuals enrolled in the trial.

4.7 Discrete-Time Queuing Systems

Controlled queuing systems can be studied using the machinery of MDPs. In an uncontrolled single-server queue, jobs arrive, enter the queue, wait for service, receive service, and then are discharged from the system. Controls can be added which either regulate the number of jobs that are admitted to the queue (admission control) or which adjust the service rate (service rate control).

Here we will consider admission control. In these models, jobs arrive for service and are placed into a 'potential job queue.' At each decision epoch, the controller observes the number of jobs in the system and then decides how many jobs to admit from the potential queue to the eligible queue. Jobs not admitted to the eligible queue never receive service. To model this as a MDP, let X_t denote the number of jobs in the system just before decision epoch t and let Z_{t-1} be the number of jobs arriving in period $t-1$ and entering the potential job queue. At decision epoch t, the controller admits u_t jobs from the potential job queue into the eligible job queue. Let Y_t be the number of possible service completions and notice that the actual number of completions is equal to the minimum of Y_t and $X_t + u_t$, the latter quantity being the number of jobs in the system at time t that can be serviced during this period.

The state of the system at decision epoch t can be represented by a pair (X_t, V_t), where V_t is the number of jobs in the potential queue at that time. These variables satisfy the following dynamical equation:

$$\begin{aligned} X_{t+1} &= [X_t + u_t - Y_t]^+ \\ V_{t+1} &= Z_t, \end{aligned}$$

where $0 \leq u_t \leq V_t$, since only jobs in the potential queue can be admitted to the system. Here we will assume that the integer-valued variables $Y_1, Y_2, \cdots$ are i.i.d. with probability mass function $f(n) = \mathbb{P}(Y_t = n)$ and likewise that $Z_1, Z_2, \cdots$ are i.i.d. with probability mass function $g(n) = \mathbb{P}(Z_1 = n)$. We will also assume that there is both a constant reward of R units for every completed job and a holding cost of $h(x)$ per period when there are x jobs in the system.

To formulate this as a MDP, let

- Decision epochs: $T = \{0, 1, \cdots, N\}, N \leq \infty$.
- States: $S = S_1 \times S_2 = \mathbb{N} \times \mathbb{N}$, where s_1 is the number in the system and s_2 is the number in the potential job queue.
- Actions: $A_{s_1,s_2} = \{0, 1, \cdots, s_2\}$.
- Rewards:
$$r_t(s_1, s_2, a) = R \cdot \mathbb{E}[\min(Y_t, s_1 + a)] - h(s_1 + a)$$
- Transition probabilities:
$$p_t(s_1', s_2' | s_1, s_2, a) = \begin{cases} f(s_1 + a - s_1')g(s_2') & \text{if } a + s_1 > s_1' > 0 \\ \left[\sum_{i=s_1+a}^{\infty} f(i)\right] g(s_2') & \text{if } s_1' = 0, a + s_1 > 0 \\ g(s_2') & \text{if } s_1' = a + s_1 = 0 \\ 0 & \text{if } s_1' > a + s_1 \geq 0. \end{cases}$$

Chapter 5

Finite-Horizon Markov Decision Processes

5.1 Optimality Criteria

Recall that a discrete-time Markov decision process is said to be a finite-horizon process if the set of decision epochs is finite. Without loss of generality, we will take this set to be $T = \{1, 2, \cdots, N\}$, where $N < \infty$. We also recall the following notation from Chapter 2:

- S is the state space.
- A_s is the set of possible actions when the system is in state s; $A = \cup_{s \in S} A_s$ is the set of all possible actions.
- $r_t(s, a)$ is the reward earned when the system is in state s and action a is selected in decision epoch $t \in \{1, \cdots, N-1\}$. $r_N(s)$ is the reward earned if the system is in state s at the final time $t = N$.
- $p_t(j|s, a)$ is the transition probability from state s at time t to state j at time $t + 1$ when action a is selected in decision epoch t.

Further, we will write $h_t = (s_1, a_1, s_2, a_2, \cdots, s_t)$ for the history of the process up to time t and we will let H_t be the set of all possible histories up to time t. Notice that each history h_t includes all of the states occupied by the system from time 1 to time t, as well as all of the actions chosen by the decision maker from time 1 to time $t - 1$; however, the action chosen at time t is not included. It follows that h_{t-1} and h_t are related by $h_t = (h_{t-1}, a_{t-1}, s_t)$.

Let X_t and Y_t be random variables which specify the state of the system and the action taken at decision epoch t, respectively, and let $R_t = r_t(X_t, Y_t)$ and $R_N = r_N(X_N)$ be real-valued random variables that denote the rewards received at times $t < N$ and N. In general, the distributions of these random variables will depend on the policy that the decision maker uses to select an action in each decision epoch. Recall that if $\pi = (d_1, \cdots, d_N) \in \Pi^{HR}$ is a history-dependent randomized policy, then for each t, $d_t : H_t \to \mathcal{P}(A)$ is a decision rule that depends on the history h_t and which prescribes a probability distribution $q_{d_t(h_t)} \in \mathcal{P}(A_{s_t})$ on the set of possible of actions: when following policy π, the decision maker will randomly choose an action $a_t \in A_{s_t}$ according to this distribution. Once a policy has been selected, this induces a probability distribution $\mathbb{P}^\pi$ on the **reward sequence** $R = (R_1, \cdots, R_N)$. This allows us to compare the suitability of different policies based on the decision maker's preferences for different reward sequences and on the probabilities with which these different sequences occur.

Notice that the distribution of the reward sequence can be simplified if we restrict our attention to deterministic policies. In particular, if the policy π is deterministic and history-dependent, then the reward sequence can be written as

$$\Big(r_1(X_1, d_1(h_1)), \cdots, r_{N-1}(X_{N-1}, d_{N-1}(h_{N-1})), r_N(X_N)\Big)$$

while if π is deterministic and Markovian, then $a_t = d_t(s_t)$ and so we can instead write

$$\Big(r_1(X_1, d_1(s_1)), \cdots, r_{N-1}(X_{N-1}, d_{N-1}(s_{N-1})), r_N(X_N)\Big).$$

One way to compare distributions on the space of reward sequences is to introduce a **utility function** $\Psi : \mathbb{R}^N \to \mathbb{R}$ which has the property that $\Psi(u) \geq \Psi(v)$ whenever the decision maker prefers the reward sequence $u = (u_1, \cdots, u_N)$ over the reward sequence $v = (v_1, \cdots, v_N)$. It should be emphasized that the choice of the utility function is very much dependent on the particular problem and the particular decision maker, e.g., different decision makers might choose different utility functions for the same problem. According to the **expected utility criterion** the decision maker should favor policy π over policy ν whenever the expected utility of π is greater than the expected utility under ν, i.e., when

$$\mathbb{E}^\pi[\Psi(R)] \geq \mathbb{E}^\nu[\Psi(R)],$$

where $\mathbb{E}^\pi[\Psi(R)]$ is the expected utility under policy π.

Because the computation of the expected utility of a policy is difficult for arbitrary utility functions, we will usually assume that Ψ is a linear function of its arguments or, more specifically, that

$$\Psi(r_1, \cdots, r_N) = \sum_{t=1}^{N} r_t.$$

This leads us to the **expected total reward criterion**, which says that we should favor policies that have higher expected total rewards. The **expected total reward** of a policy $\pi \in \Pi^{HR}$ when the initial state of the system is s will be denoted $v_N^\pi(s)$ and is equal to

$$v_N^\pi(s) \equiv \mathbb{E}_s^\pi\left[\sum_{t=1}^{N-1} r_t(X_t, Y_t) + r_N(X_N)\right].$$

If $\pi \in \Pi^{HD}$, then this expression can be simplified to

$$v_N^\pi(s) \equiv \mathbb{E}_s^\pi\left[\sum_{t=1}^{N-1} r_t(X_t, d_t(h_t)) + r_N(X_N)\right].$$

The expected total reward criterion presumes that the decision maker is indifferent to the timing of the rewards, e.g., a reward sequence in which a unit reward is received in each of the N decision periods is no more or less valuable than a reward sequence in which all N units are received in the first or the last decision period. However, we can also introduce a **discount factor** to account for scenarios in which the value of a reward does depend on when it is received. In this setting, a discount factor will be a real number $\lambda \in [0, \infty)$ which measures the value at time t of a one unit reward received at time $t+1$. We will generally assume that $\lambda < 1$, meaning that rewards received in the future are worth less in the present than rewards of the same size that are received in the present. Furthermore, if we assume that λ is constant for the duration of the Markov decision process, then a one unit reward received t periods in the future will have present value equal to λ^t. For a policy $\pi \in \Pi^{HR}$, the **expected total discount reward** is equal to

$$v_{N,\lambda}^\pi(s) \equiv \mathbb{E}_s^\pi\left[\sum_{t=1}^{N-1} \lambda^{t-1} r_t(X_t, Y_t) + \lambda^{N-1} r_N(X_N)\right].$$

Discounting will have little effect on the theory developed for finite-horizon MDP's, but will play a central role when we consider infinite-horizon MDP's.

A policy $\pi^* \in \Pi^{HR}$ will be said to be **optimal** with respect to the expected total reward criterion if it is true that

$$v_N^{\pi^*}(s) \geq v_N^{\pi}(s)$$

for all $s \in S$ and all $\pi \in \Pi^{HD}$. In other words, an optimal policy is one that maximizes the expected total reward for every initial state of the process. In some cases, optimal policies will not exist and we will instead seek what are called **ϵ-optimal policies**. A policy π^* will be said to be ϵ-optimal for some positive number $\epsilon > 0$ if it is true that

$$v_N^{\pi^*}(s) \geq v_N^{\pi}(s) - \epsilon$$

for all $s \in S$ and all $\pi \in \Pi^{HD}$. In other words, a policy π^* is ϵ-optimal if there is no other policy with an expected reward that has an expected total reward for some initial state s that is more than ϵ units larger than the expected total reward for π^* for that state.

We will define the **value** of a Markov decision problem to be the function $v_N^* : S \to \mathbb{R}$ given by

$$v_N^*(s) \equiv \sup_{\pi \in \Pi^{HR}} v_N^{\pi}(s),$$

where the supremum is needed to handle cases in which the maximum is not attained. Notice that the value depends on the initial state of the process. Furthermore, a policy π^* is an optimal policy if and only if $v_N^{\pi^*}(s) = v_N^*(s)$ for every $s \in S$.

5.2 Policy Evaluation

Before we can look for optimal or ϵ-optimal policies for a Markov decision problem, we need to be able to calculate the expected total reward of a policy. If the state space and action set are finite, then in principle this could be done by enumerating all of the possible histories beginning from a given initial state and calculating their probabilities. For example, if $\pi \in \Pi^{HD}$ is a deterministic history-dependent policy, then its expected total reward when the initial state is s is equal to

$$v_N^{\pi}(s) = \sum_{\{h_N \in H_N : s_1 = s\}} \mathbb{P}_s^{\pi}(h_N) \left(\sum_{t=1}^{N-1} r_t(s_t, d_t(h_t)) + r_N(s_N) \right),$$

where the probabilities $\mathbb{P}_s^{\pi}$ appearing in the sum can be calculated using the formula

$$\mathbb{P}_s^{\pi}(h_N) = \prod_{t=2}^{N} p_t(s_t | s_{t-1}, d_{t-1}(h_{t-1}))$$

for any history $h_N = (s_1, a_1, \cdots, s_{N-1}, a_{N-1}, s_N)$ that satisfies the conditions $s_1 = s$ and $a_t = d_t(h_t)$ for $t = 1, \cdots, N-1$. In practice, this calculation may be intractable for two reasons: (i) H_N can be a very large set, with $K^N L^{N-1}$ histories if S contains K element and A_s contains L actions for each s; and (ii) because we need to both evaluate and multiply together $N-1$ transition probabilities to evaluate the probability of each history $h_N \in H_N$.

Although there isn't much that can be done about the number of histories other than to restrict attention to Markovian policies, we can at least avoid the calculation of the sample path probabilities by using a recursive method from dynamic programming known as **backward induction**. Let $\pi \in \Pi^{HR}$ be a randomized history-dependent policy and for each $t = 1, \cdots, N$, let $u_t^\pi : H_t \to \mathbb{R}$ be the expected total reward obtained by using policy π in decision epochs $t, t+1, \cdots, N$:

$$u_t^\pi(h_t) \equiv \mathbb{E}_{h_t}^\pi \left[\sum_{n=t}^{N-1} r_n(X_n, Y_n) + r_N(X_N) \right]. \tag{5.1}$$

Notice that $u_N^\pi(h_N) = r_N(s_N)$ while $u_1^\pi(h_1) = v_N^\pi(s)$ when $h_1 = (s)$. The idea behind backward induction is that we can recursively calculate the quantities $u_t^\pi(h_t)$ in terms of the quantities $u_{t+1}^\pi(h_{t+1})$ for histories that satisfy the condition that $h_{t+1} = (h_t, d_t(h_t), s_{t+1})$ for some s_{t+1}. Following Puterman, we will refer to this method as the **finite-horizon policy evaluation algorithm**. If $\pi \in \Pi^{HD}$ is a deterministic history-dependent policy, then this algorithm can be implemented by following these four steps:

1. Set $t = N$ and $u_N^\pi(h_N) = r_N(s_N)$ for every $h_N = (h_{N-1}, d_{N-1}(h_{N-1}), s_N) \in H_N$.

2. If $t = 1$, then stop; otherwise, proceed to step 3.

3. Set $t = t - 1$ and then calculate $u_t^\pi(h_t)$ for each history $h_t = (h_{t-1}, d_{t-1}(h_{t-1}), s_t) \in H_t$ using the recursive formula

$$u_t^\pi(h_t) = r_t(s_t, d_t(h_t)) + \sum_{j \in S} p_t(j|s_t, d_t(h_t)) u_t^\pi((h_t, d_t(h_t), j)), \tag{5.2}$$

   where $h_{t+1} = (h_t, d_t(h_t), j) \in H_{t+1}$.

4. Return to step 2.

Equation (5.2) can also be written in the following form

$$u_t^\pi(h_t) = r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}^\pi \left[u_{t+1}^\pi((h_t, d_t(h_t), X_{t+1})) \right]. \tag{5.3}$$

In other words, the expected total reward of policy π over periods $t, t+1, \cdots, N$ when the history at decision epoch t is h_t is equal to the immediate reward $r_t(s_t, d_t(h_t))$ received by selecting action $d_t(h_t)$ in decision epoch t plus the expected total reward received over periods $t+1, \cdots, N$. The next theorem asserts that the backward induction correctly calculate the expected total reward of a deterministic history-dependent policy.

Theorem 5.1. *Let $\pi \in \Pi^{HD}$ and suppose that the sequence $u_N^\pi, u_{N-1}^\pi, \cdots, u_1^\pi$ has been generated using the finite-horizon policy evaluation algorithm. Then equation (5.1) holds for all t $\le N$ and $v_N^\pi(s) = u_1^\pi(s)$ for all $s \in S$.*

Proof. We prove the theorem by backwards induction on t. For the base case, notice that the result is true when $t = N$. Suppose then that (5.1) holds for $t+1, t+2, \cdots, N$ (the induction hypothesis). Using (5.3) and then the induction hypothesis, we have

$$\begin{aligned}
u_t^\pi(h_t) &= r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}^\pi \left[u_{t+1}^\pi((h_t, d_t(h_t), X_{t+1})) \right] \\
&= r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}^\pi \left[\mathbb{E}_{h_{t+1}}^\pi \left[\sum_{n=t+1}^{N-1} r_n(X_n, Y_n) + r_N(X_N) \right] \right] \\
&= r_t(s_t, d_t(h_t)) + \mathbb{E}_{h_t}^\pi \left[\sum_{n=t+1}^{N-1} r_n(X_n, Y_n) + r_N(X_N) \right] \\
&= \mathbb{E}_{h_t}^\pi \left[\sum_{n=t}^{N-1} r_n(X_n, Y_n) + r_N(X_N) \right].
\end{aligned}$$

Note that the last identity holds because $h_t = (h_{t-1}, d_{t-1}(h_{t-1}), h_t)$ and therefore

$$\begin{aligned} r_t(s_t, d_t(h_t)) &= \mathbb{E}^{\pi}_{h_t}\left[r_t(s_t, d_t(h_t))\right] \\ &= \mathbb{E}^{\pi}_{h_t}\left[r_t(X_t, Y_t))\right]. \end{aligned}$$

This shows that the identity hold for t as well, which completes the induction step. □

The policy evaluation algorithm can also be adapted for randomized history-dependent policies. For such policies, the recursion takes the form:

$$u^{\pi}_t(h_t) = \sum_{a \in A_{s_t}} q_{d_t(h_t)}(a) \left[r_t(s_t, a_t) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}(h_t, a, j) \right]. \tag{5.4}$$

The next theorem can be established using a proof similar to that given for Theorem 5.1.

Theorem 5.2. *Let $\pi \in \Pi^{HR}$ and suppose that the sequence $u^{\pi}_N, u^{\pi}_{N-1}, \cdots, u^{\pi}_1$ has been generated using the finite-horizon policy evaluation algorithm with recursion (5.4). Then equation (5.1) holds for all $t \le N$ and $v^{\pi}_N(s) = u^{\pi}_1(s)$ for all $s \in S$.*

If π is a deterministic Markovian policy, then the recursion (5.2) can be written in a much simpler form,

$$u^{\pi}_t(s_t) = r_t(s_t, d_t(s_t)) + \sum_{j \in S} p_t(j|s_t, d_t(s_t)) u^{\pi}_{t+1}(j), \tag{5.5}$$

where the quantities $u^{\pi}_t(s_t)$ now depend on states rather than on entire histories up to time t. One consequence of this is that fewer operations are required to implement the policy evaluation algorithm for a deterministic Markovian policy than for a deterministic history-dependent policy. For example, only $(N-2)K^2$ multiplications are required if $\pi \in \Pi^{MD}$ versus $(K^2 + \cdots + K^N) \approx K^N$ if $\pi \in \Pi^{HD}$.

5.3 Optimality Equations

Given a finite-horizon Markov decision problem with decision epochs $T = \{1, \cdots, N\}$, define the **optimal value functions** $u^*_t : H_t \to \mathbb{R}$ by setting

$$u^*_t(h_t) = \sup_{\pi \in \Pi^{HR}} u^{\pi}_t(h_t),$$

where $u^{\pi}_t(h_t)$ is the expected total reward earned by using policy π from time t to N and the supremum is taken over all history-dependent randomized policies. Then the **optimality equations** (also known as the **Bellman equations**) are

$$u_t(h_t) = \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}(h_t, a, j) \right\} \tag{5.6}$$

for $t = 1, \cdots, N-1$ and $h_t = (h_{t-1}, a_{t-1}, s_t)$, along with the boundary condition

$$u_N(h_N) \equiv r_N(s_N) \tag{5.7}$$

for $h_N = (h_{N-1}, a_{N-1}, s_N)$. The supremum in (5.6) is taken over the set of all possible actions that are available when the system is in state s_t and this can be replaced by a maximum when all of the action sets are finite. The Bellman equations are important because they can be used to verify that a policy is optimal and can sometimes be used to identify optimal policies.

The next theorem states that solutions to the Bellman equations have certain optimality properties. Part (a) implies that the solutions are the optimal value functions from period t onward, while (b) implies that the solution obtained at $n = 1$ is the value function for the MDP.

Theorem 5.3. *Suppose that the functions $u_1, \cdots, u_N$ satisfy equations (5.6) - (5.7). Then*

(a) $u_t(h_t) = u_t^*(h_t)$ for all $h_t \in H_t$;

(b) $u_1(s_1) = v_N^*(s_1)$ for all $s_1 \in S$.

Proof. The proof of (a) is divided into two parts, both of which rely on backwards induction on t. We begin by showing:

Claim 1: $u_t(h_t) \geq u_t^*(h_t)$ for all $t = 1, \cdots, N$ and all $h_t \in H_t$.

To set up the induction argument, first observe that the result is true when $t = N$, since condition (5.7) guarantees that $u_N(h_N) = u_N^\pi(h_N) = r_N(s_N)$ for all $\pi \in \Pi^{HR}$ when $h_N = (h_{N-1}, a_{N-1}, s_N)$, which implies that $u_N(h_N) = u_N^*(h_N)$. Suppose that the result is true for $t = n+1, \cdots, N$ (the induction hypothesis). Then, for any policy $\pi' = (d_1', \cdots, d_N') \in \Pi^{HR}$, the optimality equation for $t = n$ and the induction hypothesis for $t = n+1$ imply that

$$\begin{aligned} u_n(h_n) &= \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}(h_n, a, j) \right\} \\ &\geq \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}^*(h_n, a, j) \right\} \\ &\geq \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}^{\pi'}(h_n, a, j) \right\} \\ &\geq \sum_{a \in A_{s_n}} q_{d_n'(h_n)}(a) \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}^{\pi'}(h_n, a, j) \right\} \\ &= u_n^{\pi'}(h_n). \end{aligned}$$

However, since π' is arbitrary, it follows that

$$u_n(h_n) \geq \sup_{\pi \in \Pi^{HR}} u_n^\pi(h_n) = u_n^*(h_n),$$

which completes the induction.

We next show that:

Claim 2: For any $\epsilon > 0$, there exists a deterministic policy $\pi' \in \Pi^{HD}$ for which

$$u_t^{\pi'}(h_t) + (N - t)\epsilon \geq u_t(h_t)$$

for all $h_t \in H_t$ and $1 \leq t \leq N$.

Suppose that $\pi' = (d_1, \cdots, d_{N-1})$ is constructed by choosing $d_t(h_t) = a \in A_{s_t}$ so that

$$r_t(s_t, d_t(h_t)) + \sum_{j \in S} p_t(j|s_t, d_t(h_t)) u_{t+1}^{\pi'}(s_t, d_t(h_t), j) + \epsilon \geq u_t(h_t).$$

(Such a choice is possible because we are assuming that the u_n satisfy equation (5.6).) To prove Claim 2 by induction on t, observe that the result holds when $t = N$ since $u_N^{\pi'}(h_N) = u_N(h_N) = r_N(s_N)$. Suppose that the result is also true for $t = n+1, \cdots, N$: $u_t^{\pi'}(h_t) + (N-t)\epsilon \geq u_t(h_t)$ (the induction hypothesis). Then

$$\begin{aligned} u_n^{\pi'}(h_n) &= r_n(s_n, d_n(h_n)) + \sum_{j \in S} p_n(j|s_n, d_n(h_n)) u_{n+1}^{\pi'}(s_n, d_n(h_n), j) \\ &\geq r_n(s_n, d_n(h_n)) + \sum_{j \in S} p_n(j|s_n, d_n(h_n)) u_{n+1}(s_n, d_n(h_n), j) - (N-n-1)\epsilon \\ &\geq u_n(h_n) - (N-n)\epsilon, \end{aligned}$$

which completes the induction on t.

Taken together, Claims 1 and 2 show that for any $\epsilon > 0$, there exists a policy $\pi' \in \Pi^{HR}$ such that for every $1 \leq t \leq N$ and every $h_t \in H_t$,

$$u_t^*(h_t) + (N-t)\epsilon \geq u_t^{\pi'}(h_t) + (N-t)\epsilon \geq u_t(h_t) \geq u_t^*(h_t).$$

Since N is fixed, we can let $\epsilon \to 0$, which establishes (a). Part (b) then follows from the fact that $u_1(s_1) = u_1^*(s_1) = v_N^*(s_1)$. □

Although Theorem 5.3 provides us with an algorithm that can be used to compute the optimal value functions, it doesn't tell us how to find optimal policies, assuming that these even exist. This limitation is addressed by our next theorem, which shows how the Bellman equations can be used to construct optimal policies under certain conditions.

Theorem 5.4. *Suppose that the functions $u_1^*, \cdots, u_N^*$ satisfy equations (5.6) - (5.7) and assume that the policy $\pi^* = (d_1^*, \cdots, d_N^*) \in \Pi^{HD}$ satisfies the following recursive sequence of identities,*

$$\begin{aligned} & r_t(s_t, d_t^*(h_t)) + \sum_{j \in S} p_t(j|s_t, d_t^*(h_t)) u_{t+1}^*(s_t, d_t^*(h_t), j) \\ & \qquad = \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}^*(h_t, a, j) \right\} \end{aligned}$$

for all $t = 1, \cdots, N-1$ and $h_t \in H_t$. Then

(a) For each $t = 1, \cdots, N$, $u_t^{\pi^*}(h_t) = u_t^*(h_t)$.

(b) π^* is an optimal policy and $v_N^{\pi^*}(s) = v_N^*(s)$ for every $s \in S$.

Proof. Part (a) can be proved by backwards induction on t. In light of Theorem 5.3, we know that the functions $u_t^*, 1 \leq t \leq N$ are the optimal value functions for the MDP and consequently

$$u_N^{\pi^*}(h_N) = u_N^*(h_N) = r_N(s_N), \quad h_N \in H_N.$$

Suppose that the result holds for $t = n+1, \cdots, N$. Then, for $h_n = (h_{n-1}, d_{n-1}^*(h_{n-1}), s_n)$, we have

$$\begin{aligned} u_n^*(h_n) &= \max_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}^*(h_n, a, j) \right\} \\ &= r_n(s_n, d_n^*(h_n)) + \sum_{j \in S} p_n(j|s_n, d_n^*(h_n)) u_{n+1}^*(h_n, d_n^*(h_n), j) \\ &= r_n(s_n, d_n^*(h_n)) + \sum_{j \in S} p_n(j|s_n, d_n^*(h_n)) u_{n+1}^{\pi^*}(h_n, d_n^*(h_n), j) \\ &= u_n^{\pi^*}(h_n). \end{aligned}$$

The first equality holds because the functions u_n^* satisfy the Bellman equations and here we are assuming that the supremum on the right-hand side of (5.6) can be replaced by a maximum; the second equality holds because d_n^* is assumed to be a decision rule that achieves this maximum; the third equality is a consequence of the induction hypothesis; lastly, the fourth equality is a consequence of Theorem 5.1. This completes the induction and establishes part (a). Part (b) follows from Theorem 5.1 and Theorem 5.3 (b), along with part (a) of this theorem:

$$v_N^{\pi^*}(s) = u_1^{\pi^*}(s) = u_1^*(s) = v_N^*(s).$$

□

It follows from this theorem that an optimal policy can be found by first solving the optimality equations for the functions $u_1^*, \cdots, u_N^*$ and then recursively choosing a sequence of decision rules $d_1^*, \cdots, d_N^*$ such that

$$d_t^*(h_t) \in \arg\max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}^*(h_t, a, j) \right\}. \tag{5.8}$$

This is sometimes known as the **"Principle of Optimality"**, which says that a policy that is optimal over decision epochs $1, \cdots, N$ is also optimal when restricted to decision epochs $t, \cdots, N$. On the other hand, optimality usually is not preserved if we truncate the time interval on the right: a policy that is optimal over decision epochs $1, \cdots, N$ need not be optimal when restricted to decision epochs $1, \cdots, t$ for $t < N$.

Provided that the action sets are finite, there will always be at least one action that maximizes the right-hand side of (5.8) and thus at least one optimal policy will exist under these conditions. Furthermore, if there is more than one action that maximizes the right-hand side, then there will be multiple optimal policies (albeit with the same expected total reward). On the other hand, if the supremum is not attained in (5.6) for some $t \in \{1, \cdots, N\}$ and some $h_t \in H_t$, then an optimal policy will not exist and we must instead make do with ϵ-optimal policies. These can be found using the following procedure.

Theorem 5.5. *Suppose that the functions $u_1^*, \cdots, u_N^*$ satisfy equations (5.6) - (5.7). Given $\epsilon > 0$, let $\pi^\epsilon = (d_1^\epsilon, \cdots, d_N^\epsilon) \in \Pi^{HD}$ be a policy which satisfies*

$$\begin{aligned} r_t(s_t, d_t^\epsilon(h_t)) + \sum_{j \in S} p_t(j|s_t, d_t^\epsilon(h_t)) u_{t+1}^*((s_t, d_t^\epsilon(h_t), j)) + \frac{\epsilon}{N-1} \\ \geq \max_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}^*((h_t, a, j)) \right\} \end{aligned} \tag{5.9}$$

for $t = 1, \cdots, N-1$. Then

(a) *For every $t = 1, \cdots, N$ and every $h_t \in H_t$,*

$$u_t^{\pi^\epsilon}(h_t) + (N-t)\frac{\epsilon}{N-1} \geq u_t^*(h_t).$$

(b) *π^ϵ is an ϵ-optimal policy and for every $s \in S$*

$$v_N^{\pi^\epsilon}(s) + \epsilon \geq v_N^*(s).$$

This can be proved by modifying the arguments used to establish claim (b) in Theorem 5.3.

5.4 Optimality of Deterministic Markov Policies

Because deterministic Markov policies are generally easier to implement and require less computational effort than randomized history-dependent policies, it is important to know when the existence of an optimal or ϵ-optimal policy of this type is guaranteed. In this section, we will show that if an optimal policy exists for a finite-horizon MDP, then there is an optimal policy that is deterministic and Markovian. Our first theorem summarizes some properties of the deterministic history-dependent policies that are constructed in the proofs of Theorems 5.3 - 5.5.

Theorem 5.6. ***Existence of deterministic history-dependent policies.***

(a) *For any $\epsilon > 0$, there exists an ϵ-optimal policy which is deterministic and history-dependent. Furthermore, any policy $\pi \in \Pi^{HD}$ which satisfies the inequalities given in (5.9) is ϵ-optimal.*

(b) *Let $(u_t^*, 1 \leq t \leq N)$ satisfy the optimal value equations (5.6) - (5.7) and suppose that for each t and each $s_t \in S$, there exists an $a' \in A_{s_t}$ such that*

$$\begin{aligned} r_t(s_t, a') &+ \sum_{j \in S} p_t(j|s_t, a')u_{t+1}^*(h_t, a', j) \\ &= \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a)u_{t+1}^*(h_t, a_t, j) \right\} \end{aligned} \tag{5.10}$$

for all histories $h_t = (s_{t-1}, a_{t-1}, s_t) \in H_t$. Then there exists a deterministic history-dependent policy which is optimal.

Our next theorem strengthens these results by asserting the existence of deterministic Markov policies which are ϵ-optimal or optimal.

Theorem 5.7. *Let $(u_t^*, 1 \leq t \leq N)$ satisfy the optimal value equations (5.6) - (5.7). Then*

(a) *For each $t = 1, \cdots, N$, $u_t^*(h_t)$ depends on h_t only through s_t.*

(b) *For any $\epsilon > 0$, there exists an ϵ-optimal policy which is deterministic and Markov.*

(c) *If there exists an $a' \in A_{s_t}$ such that equation (5.10) holds for each $s_t \in S$ and $t = 1, 2, \cdots, N-1$, then there exists an optimal policy which is deterministic and Markov.*

Proof. Part (a) can be proved by backwards induction on t. Since $u_N^*(h_N) = r_N(s_N)$, the result is clearly true for $t = N$. Thus, let the induction hypothesis be that the result is true for $t = n+1, \cdots, N$. Then, according to equation (5.6),

$$\begin{aligned} u_n^*(h_n) &= \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}((h_n, a, j)) \right\} \\ &= \sup_{a \in A_{s_n}} \left\{ r_n(s_n, a) + \sum_{j \in S} p_n(j|s_n, a) u_{n+1}(j) \right\}, \end{aligned}$$

where the second equality follows from the induction hypothesis for $t = n+1$. This shows that $u_n^*(h_n)$ depends on h_n only through s_n, which completes the induction.

To prove part (b), choose $\epsilon > 0$ and let $\pi^\epsilon = (d_1^\epsilon, \cdots, d_{N-1}^\epsilon) \in \Pi^{MD}$ be any policy which satisfies the inequalities

$$\begin{aligned} & r_t(s_t, d_t^\epsilon(s_t)) + \sum_{j \in S} p_t(j|s_t, d_t^\epsilon(s_t)) u_{t+1}^\epsilon(j) + \frac{\epsilon}{N-1} \\ & \qquad \geq \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}^\epsilon(j) \right\}, \end{aligned}$$

Then π^ϵ satisfies the conditions of Theorem 5.6 (a) and thus is ϵ-optimal.

Lastly, part (c) follows because we can construct a policy $\pi^* = (d_1^*, \cdots, d_{N-1}^*) \in \Pi^{MD}$ such that

$$\begin{aligned} & r_t(s_t, d_t^*(s_t)) + \sum_{j \in S} p_t(j|s_t, d_t^*(s_t)) u_{t+1}^*(j) \\ & \qquad = \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u_{t+1}^*(j) \right\}, \end{aligned}$$

and then Theorem 5.4 (b) implies that π^* is an optimal policy. □

It follows from Theorem 5.7 that

$$v_N^*(s) = \sup_{\pi \in \Pi^{HR}} v_N^\pi(s) = \sup_{\pi \in \Pi^{MD}} v_N^\pi(s), \quad s \in S,$$

and thus it suffices to restrict attention to deterministic Markov policies when studying finite-horizon Markov decision problems with the total expected reward criterion.

Although not every Markov decision problem has an optimal policy (of any type), there are several simple conditions that guarantee the existence of at least one optimal policy. We begin with a definition.

Definition 5.1. *Suppose that $f : D \to [-\infty, \infty]$ is a real-valued function defined on a domain $D \subset \mathbb{R}^n$. Then f is said to be* ***upper semicontinuous*** *if*

$$\limsup_{x_n \to x} f(x_n) \leq f(x)$$

for any sequence $(x_n; n \geq 1)$ converging to x. Similarly, f is said to be ***lower semicontinuous*** *if*

$$\liminf_{x_n \to x} f(x_n) \geq f(x)$$

for any sequence $(x_n; n \geq 1)$ converging to x.

Some important properties of upper and lower semicontinuous functions are listed below.

1. f is continuous if and only if it is both upper and lower semicontinuous.
2. f is upper semicontinuous if and only if $-f$ is lower semicontinuous.
3. If f is upper semicontinuous and $C \subset D$ is a compact set, then f achieves its maximum on C, i.e., there is a point $x^* \in C$ such that
$$f(x^*) = \sup_{x \in C} f(x).$$

Theorem 5.8. *Assume that the state space S is countable and that one of the following three sets of criteria are satisfied:*

(a) A_s is finite for each $s \in S$, or

(b) A_s is compact for each $s \in S$; $r_t(s,a)$ is a continuous function of a for all $t \in T$ and $s \in S$, and there exists a constant $M < \infty$ such that $|r_t(s,a)| \leq M$ for all $t \in T$, $s \in S$, and $a \in A_s$; and $p_t(j|s,a)$ is continuous in a for all $s, j \in S$ and $t \in T$, or

(c) A_s is compact for each $s \in S$; $r_t(s,a)$ is upper semicontinuous in a for all $t \in T$ and $s \in S$, and there exists a constant $M < \infty$ such that $|r_t(s,a)| \leq M$ for all $t \in T$, $s \in S$ and $a \in A_s$; and $p_t(j|s,a)$ is lower semicontinuous in a for all $s, j \in S$ and $t \in T$.

Then there exists a deterministic Markovian policy which is optimal.

Proof. According to Theorem 5.7 (c), it suffices to show that for every $s \in S$ and $t \in T$, there exists an action $a' \in A_s$ such that

$$\begin{aligned} r_t(s_t, a') &+ \sum_{j \in S} p_t(j|s_t, a') u^*_{t+1}(j) \\ &= \sup_{a \in A_{s_t}} \left\{ r_t(s_t, a) + \sum_{j \in S} p_t(j|s_t, a) u^*_{t+1}(j) \right\}. \end{aligned}$$

This is clearly true if each action set A_s is finite. Alternatively, if the conditions in (b) are true, then it suffices to show that the functions

$$\Psi_t(s,a) \equiv r_t(s,a) + \sum_{j \in S} p_t(j|s,a) u^*_{t+1}(j)$$

are continuous in a for all $t \in T$ and $s \in S$, since we know that the sets A_s are compact and any continuous function achieves its maximum on a compact set. However, since the rewards $r_t(s,a)$ are assumed to be continuous in a, it suffices to show that the sum

$$\sum_{j \in S} p_t(j|s,a) u^*_{t+1}(j)$$

is a continuous function of a for all $t \in T$ and $s \in S$. To this end, we first show that the conditions in (b) imply that the functions u^*_t are bounded on S. Indeed, since the optimal value functions satisfy the recursive equations,

$$u^*_t(s) = \sup_{a \in A_s} \left\{ r_t(s,a) + \sum_{j \in S} p_t(j|s,a) u^*_{t+1}(j) \right\},$$

it follows that

$$|u_t^*(s)| \leq M + \sup_{s' \in S} |u_{t+1}^*(s')|,$$

for all $s \in S$, which implies that

$$\sup_{s \in S} |u_t^*(s)| \leq M + \sup_{s \in S} |u_{t+1}^*(s)|$$

for all $t = 1, \cdots, N$. However, since

$$\sup_{s \in S} |u_N^*(s)| = \sup_{s \in S} |r_N(s)| \leq M,$$

a simple induction argument shows that

$$\sup_{s \in S} |u_t^*(s)| \leq NM$$

for $t = 1, \cdots, N$, and so the optimal value functions are bounded as claimed above.

To complete the argument, let $\epsilon > 0$ be given and suppose that $(a_n : n \geq 0)$ is a sequence in A_s which converges to a_∞. Since the functions $p_t(j|s,a)$ are continuous in a for all $t \in T$ and $s, j \in S$, we know that

$$p_t(j|s, a_\infty) = \lim_{n \to \infty} p_t(j|s, a_n). \tag{5.11}$$

Furthermore, since for each $t \in T$, $s \in S$ and $a \in A_s$, we know that $p_t(\cdot|s,a)$ is a probability distribution on S, we can choose a finite subset $K \equiv K(t,s,a) = \{j_1, \cdots, j_M\} \subset S$ such that

$$\sum_{j \in K} p_t(j|s,a) > 1 - \epsilon.$$

For each element $j_i \in K$, use (5.11) to choose N_i large enough that for all $n \geq N_i$, we have

$$|p_t(j_i|s,a) - p_t(j_i|s,a_n)| < \frac{\epsilon}{M}.$$

Summing over the elements in K and using the triangle inequality gives

$$\begin{aligned} \left|1 - \sum_{j \in K} p_t(j|s,a_n)\right| &\leq \left|1 - \sum_{j \in K} p_t(j|s,a) + \sum_{j \in K} p_t(j|s,a) - \sum_{j \in K} p_t(j|s,a_n)\right| \\ &\leq \epsilon + \sum_{j \in K} |p_t(j|s,a) - p_t(j|s,a_n)| \\ &\leq \epsilon + M \cdot \frac{\epsilon}{M} = 2\epsilon \end{aligned}$$

for all $n \geq \tilde{N} \equiv \max\{N_1, \cdots, N_M\} < \infty$. Thus

$$\sum_{j \in K^c} p_t(j|s,a_n) \leq 2\epsilon$$

for all $n \geq \tilde{N}$ and consequently

$$\left|\sum_{j \in K^c} p_t(j|s,a_n) u_{t+1}^*(j)\right| \leq \sum_{j \in K^c} p_t(j|s,a_n) \left|u_{t+1}^*(j)\right| \leq 2NM\epsilon$$

for all such n (including $n = \infty$). Finally, if $n \geq \tilde{N}$, then these results give

$$\begin{aligned}
\left|\sum_{j\in S} p_t(j|s,a)u^*_{t+1}(j) - \sum_{j\in S} p_t(j|s,a_n)u^*_{t+1}(j)\right| &\leq \sum_{j\in S} \left|p_t(j|s,a)u^*_{t+1}(j) - p_t(j|s,a_n)u^*_{t+1}(j)\right| \\
&= \sum_{j\in K} \left|p_t(j|s,a)u^*_{t+1}(j) - p_t(j|s,a_n)u^*_{t+1}(j)\right| + \\
&\quad \sum_{j\in K^c} \left|p_t(j|s,a)u^*_{t+1}(j) - p_t(j|s,a_n)u^*_{t+1}(j)\right| \\
&\leq \sum_{j\in K} \left|p_t(j|s,a) - p_t(j|s,a_n)\right| \left|u^*_{t+1}(j)\right| + \\
&\quad \sum_{j\in K^c} \left|p_t(j|s,a)u^*_{t+1}(j)\right| + \sum_{j\in K^c} \left|p_t(j|s,a_n)u^*_{t+1}(j)\right| \\
&\leq NM\epsilon + 2NM\epsilon + 2NM\epsilon \\
&= 5NM\epsilon.
\end{aligned}$$

However, since N and M are fixed and $\epsilon > 0$ can be made arbitrarily small, it follows that the difference

$$\left|\sum_{j\in S} p_t(j|s,a)u^*_{t+1}(j) - \sum_{j\in S} p_t(j|s,a_n)u^*_{t+1}(j)\right| \to 0$$

as n tends to infinity. This establishes that the sum appearing in $\Psi_t(s,a)$ is continuous in a and therefore so is $\Psi_t(s,a)$ itself, which is what we needed to show.

For the proof that (c) is sufficient to guarantee the existence of an optimal policy, see p. 91 and appendix B of Puterman (2005). □

5.5 The Backward Induction Algorithm

In the last section, we showed that a finite-horizon MDP is guaranteed to have at least one optimal policy that is both Markovian and deterministic if the optimality equations (5.10) can be solved for every $t = 1, \cdots, N-1$ and $s \in S$. In this case, all such optimal policies can be found with the help of the **backward induction algorithm**, which consists of the following steps:

1. Set $t = N$ and let
$$u^*_N(s) = r_N(s) \quad \text{for all } s \in S.$$

2. Substitute $t-1$ for t and let
$$\begin{aligned}
u^*_t(s) &= \max_{a\in A_s}\left\{r_t(s,a) + \sum_{j\in S} p_t(j|s,a)u^*_{t+1}(j)\right\} &(5.12)\\
A^*_{s,t} &= \arg\max_{a\in A_s}\left\{r_t(s,a) + \sum_{j\in S} p_t(j|s,a)u^*_{t+1}(j)\right\}. &(5.13)
\end{aligned}$$

3. Stop if $t = 1$. Otherwise return to step 2.

It then follows that $v_N(s) = u_1^*(s)$ is the value of the Markov decision problem and that any policy $\pi^* = (d_1^*, \cdots, d_{N-1}^*) \in \Pi^{MD}$ with the property that

$$d_t^*(s) \in A_{s,t}^* \tag{5.14}$$

for every $t = 1, \cdots, N-1$ and $s \in S$ is optimal, i.e., $v_N^{\pi^*} = v_N^*(s)$ for every $s \in S$. Conversely, if $\pi^* \in \Pi^{MD}$ is an optimal policy for the MDP, then π^* satisfies condition (5.14). Notice that there will be more than one optimal policy if and only if at least one of the sets $A_{s,t}^*$ contains more than one element.

If $|S| = K$ and $|A_s| = L$ for each $s \in S$, then full implementation of the backward induction algorithm will require at most $(N-1)LK^2$ multiplications and an equivalent number of evaluations of the transition probabilities. This assumes that every transition probability $p_t(j|s,a)$ is positive. If, instead, most transitions are forbidden, i.e., if $p_t(j|s,a) = 0$ for most $s, j \in S$, then much less effort will be needed. Likewise, if the sets A_s are subsets of $\mathbb{R}^n$, then it may be possible to use analytical or numerical tools to solve the maximization problem in (5.12).

5.6 Examples

5.6.1 The Secretary Problem

Recall the problem: N candidates are interviewed sequentially, in random order, and at the conclusion of each interview we can either offer the position to that candidate or reject them and move on to the next. The aim is to maximize the probability that we offer the position to the best candidate, assuming that we can discern the relative ranks of the candidates interviewed but not their absolute ranks.

Let $u_t^*(1)$ be the maximum probability that we choose the best candidate given that the t'th candidate is the best interviewed so far and let $u_t^*(0)$ be the maximum probability that we choose the best candidate given that the t'th candidate is not the best one interviewed so far. These are the optimal value functions for this problem and thus they satisfy the following recursions:

$$u_N^*(1) = 1, \quad u_N^*(0) = 0, \quad u_N^*(\Delta) = 0,$$

and, for $t = 1, \cdots, N-1$,

$$\begin{aligned}
u_t^*(0) &= \max\left\{g_t(0) + u_{t+1}^*(\Delta), 0 + p_t(1|0)u_{t+1}^*(1) + p_t(0|0)u_{t+1}^*(0)\right\} \\
&= \max\left\{0, \frac{1}{t+1}u_{t+1}^*(1) + \frac{t}{t+1}u_{t+1}^*(0)\right\} \\
&= \frac{1}{t+1}u_{t+1}^*(1) + \frac{t}{t+1}u_{t+1}^*(0), \\
u_t^*(1) &= \max\left\{g_t(1) + u_{t+1}^*(\Delta), 0 + p_t(1|1)u_{t+1}^*(1) + p_t(0|1)u_{t+1}^*(0)\right\} \\
&= \max\left\{\frac{t}{N}, \frac{1}{t+1}u_{t+1}^*(1) + \frac{t}{t+1}u_{t+1}^*(0)\right\} \\
&= \max\left\{\frac{t}{N}, u_t^*(0)\right\}, \\
u_t^*(\Delta) &= u_{t+1}^*(\Delta) \;=\; 0.
\end{aligned}$$

From these equations, we see that an optimal policy for this problem can be formulated as follows. In state 0, the optimal action is always to continue (if $t < N$), as we know that the current candidate is not best overall. In contrast, in state 1, the optimal action is to continue if $u_t^*(0) > t$ and to stop if $u_t^*(0) < t/N$; if $u_t^*(0) = t/N$, then either action is optimal.

A more explicit representation of the optimal policy can be obtained in the following way. Suppose that $u_n^*(1) > n/N$ for some $n < N$. Then, from the optimal value equations, we know that $u_n^*(0) = u_n^*(1) > n/N$ must also hold for this t and consequently

$$u_{n-1}^*(0) = \frac{1}{n} u_n^*(1) + \frac{n-1}{n} u_n^*(0) = u_n^*(1) > \frac{n}{N}.$$

Thus

$$u_{n-1}^*(1) = \max\left\{\frac{n-1}{N}, u_{n-1}^*(0)\right\} = u_n^*(1) \geq \frac{n}{N} > \frac{n-1}{N}$$

and it follows that $u_t^*(1) = u_t^*(0) > t/N$ for all $t = 1, \cdots, n$. Furthermore, since the optimal decision rule is to continue if and only if $u_t^*(0) > t/N$, it follows that there is a constant τ with

$$\tau = \max\{t \leq N : u_t^*(1) > t/N\}$$

such that the optimal decision rule has the form: *"Observe the first τ candidates without choosing any of these; then choose the first candidate who is better than all previous ones."*

Before we derive an explicit formula for τ, we first show that $\tau \geq 1$ whenever $N > 2$. Were this not the case, then for all $t = 1, \cdots, N$, we would have $u_t^*(1) = t/N$, so that

$$u_t^*(0) = \frac{1}{t+1}\frac{t+1}{N} + \frac{t}{t+1} u_{t+1}^*(0) = \frac{1}{N} + \frac{t}{t+1} u_{t+1}^*(0).$$

However, since $u_N^*(0) = 0$, this implies that

$$u_t^*(0) = \frac{t}{N}\left[\frac{1}{t} + \frac{1}{t+1} + \cdots + \frac{1}{N-1}\right], \quad 1 \leq t < N,$$

for $N > 2$. Taking $t = 1$, we then have $u_1^*(0) > 1/N = u_1^*(1) \geq u_1^*(0)$, a contradiction, and so we can conclude that $\tau \geq 1$. Thus, when $N > 2$, we have

$$u_1^*(0) = u_1^*(1) = \cdots u_\tau^*(0) = u_\tau^*(1),$$

and

$$\begin{aligned} u_t^*(1) &= \frac{t}{N} \\ u_t^*(0) &= \frac{t}{N}\left[\frac{1}{t} + \frac{1}{t+1} + \cdots + \frac{1}{N-1}\right] \end{aligned}$$

for $t > \tau$. However, since $u_t^*(0) \leq t/N$ when $t > \tau$, we see that

$$\tau = \max\left\{t \geq 1 : \left[\frac{1}{t} + \frac{1}{t+1} + \cdots + \frac{1}{N-1}\right] > 1\right\}. \tag{5.15}$$

Suppose that N is large and let $\tau(N)$ be the value of τ for the secretary problem with N candidates. Then

$$\begin{aligned} 1 &\approx \sum_{k=\tau(N)}^{N-1} \frac{1}{k} \\ &\approx \int_{\tau(N)}^{N} \frac{1}{x} dx \\ &= \ln\left(\frac{N}{\tau(N)}\right). \end{aligned}$$

In fact, these approximations can be replaced by identities if we take the limit as $N \to \infty$, which shows that

$$\lim_{N\to\infty} \frac{\tau(N)}{N} = e^{-1}$$

and also

$$\lim_{N\to\infty} u_1^*(0) = \lim_{N\to\infty} u_1^*(1) = \lim_{N\to\infty} u_{\tau(N)}^*(0) = \lim_{N\to\infty} \frac{\tau(N)}{N} = e^{-1}.$$

Thus, when N is sufficiently large, the optimal policy is to observe the first N/e candidates and then then choose the next one to come along with the highest relative rank, in which case the probability that that candidate is the best one overall is approximately $1/e \approx 0.368$.

5.7 Monotone Policies

Definition 5.2. *Suppose that the state space S and all of the action sets A_s for a finite-horizon Markov decision problem are ordered sets, and let $\pi = (d_1, \cdots, d_{N-1})$ be a deterministic Markov policy for this problem.*

*(a) π is said to be **non-decreasing** if for each $t = 1, \cdots, N-1$ and any pair of states $s_1, s_2 \in S$ with $s_1 < s_2$, it is true that $d_t(s_1) \leq d_t(s_2)$.*

*(b) π is said to be **non-increasing** if for each $t = 1, \cdots, N-1$ and any pair of states $s_1, s_2 \in S$ with $s_1 < s_2$, it is true that $d_t(s_1) \geq d_t(s_2)$.*

*(c) π is said to be **monotone** if it is either non-decreasing or non-increasing.*

Monotone policies are of special interest because they are sometimes easier to compute and easier to implement. For example, **control limit policies**, which have decision rules of the form

$$d_t(s) = \begin{cases} a_1 & \text{if } s < s_t^* \\ a_2 & \text{if } s \geq s_t^*, \end{cases}$$

can be interpreted as monotone policies if the set $\{a_1, a_2\}$ is equipped with an ordering such that $a_1 < a_2$, say.

In this section, we will describe some conditions on Markov decision problems which guarantee the existence of optimal policies which are monotone. We will then see how the backward induction algorithm can be modified to efficiently search for such policies. However, we begin by introducing a special class of bivariate functions which have been found to be useful in a variety of problems involving dynamic programming.

Definition 5.3. *Suppose that X and Y are partially-ordered sets. A real-valued function $g : X \times Y \to \mathbb{R}$ is said to be **superadditive** if for all $x^- \leq x^+$ in X and all $y^- \leq y^+$ in Y,*

$$g(x^-, y^-) + g(x^+, y^+) \geq g(x^-, y^+) + g(x^+, y^-). \tag{5.16}$$

*Alternatively, g is said to be **subadditive** if the reverse inequality holds.*

Superadditive (subadditive) functions are sometimes said to be **supermodular** (**submodular**), while (5.16) is sometimes called the **quadrangle inequality**. Any separable function $h(x, y) = f(x) + g(y)$ is both super- and subadditive, and clearly a function $h(x, y)$ is superadditive if and only if the function $-h(x,y)$ is subadditive. Furthermore, if $X, Y \subset \mathbb{R}$ and g is a twice continuously differentiable function with

$$\frac{\partial^2 g(x,y)}{\partial x \partial y} \geq 0$$

then g is superadditive.

We will need two technical lemmas which we state without proof (see Section 4.7 in Puterman (2005)). Our first lemma establishes an important relationship between superadditive functions and nondecreasing functions.

Lemma 5.1. *Suppose that $g : X \times Y \to \mathbb{R}$ is superadditive and that for each $x \in X$ the maximum $\max_y g(x,y)$ is realized. Then the function*

$$f(x) \equiv \max\{y' \in \arg\max_{y \in Y} g(x,y)\}$$

is monotone nondecreasing in x.

Lemma 5.2. *Suppose that $(p_n; n \geq 0)$ and $(p'_n; n \geq 0)$ are sequences of non-negative real numbers such that the inequality*

$$\sum_{j=k}^{\infty} p_j \leq \sum_{j=k}^{\infty} p'_j$$

holds for every $k \geq 1$, with equality when $k = 0$. If $(v_j; j \geq 0)$ is a nondecreasing sequence of real numbers (not necessarily non-negative), then

$$\sum_{j=0}^{\infty} p_j v_j \leq \sum_{j=0}^{\infty} p'_j v_j.$$

In particular, if X and Y are non-negative integer-valued random variables with $p_j = \mathbb{P}(X = j)$ and $p'_j = \mathbb{P}(Y = j)$, then the theorem asserts that if Y is **stochastically greater** than X, then $\mathbb{E}[f(X)] \leq \mathbb{E}[f(Y)]$ for any nondecreasing function $f : \mathbb{N} \to \mathbb{R}$.

Before we can prove that monotone optimal policies exist under certain general conditions, we first need to investigate the monotonicity properties of the optimal value functions. For the rest of this section we will assume that $S = \mathbb{N}$, that $A_s = A'$ for all $s \in S$ and that the maximum

$$\max_{a \in A'} \left\{ r_t(s,a) + \sum_{j=0}^{\infty} p(j|s,a) u(j) \right\}$$

is attained for all $s \in S$, $t \in T$ and all monotone functions $u : S \to \mathbb{R}$. We will also let

$$q_t(k|s,a) \equiv \sum_{j=k}^{\infty} p_t(j|s,a)$$

denote the probability that the system moves to a state $j \geq k$ when it was in state s at time t and action a was selected.

Proposition 5.1. *Let $(u_1^*, \cdots, u_N^*)$ be the optimal value functions for a MDP and assume that* *(a) $r_t(s,a)$ is a nondecreasing (nonincreasing) function of s for all $a \in A'$, $t \in T$ and $r_N(s)$ is a nondecreasing (nonincreasing) function of S.*

(b) $q_t(k|s,a)$ is nondecreasing in s for all $k \in S$, $a \in A'$ and $t \in T$.

Then $u_t^*(s)$ is a nondecreasing (nonincreasing) function of s for $t = 1, \cdots, N$.

Proof. It suffices to prove the theorem under the assumption that the functions $r_t(s,a)$ and $r_N(s)$ are nondecreasing, and we will do this by backwards induction on t. Since $u_N^*(s) = r_N(s)$ is nondecreasing by assumption, the result holds for $t = N$. Suppose then that the result holds for $t = n+1, n+2, \cdots N$. We know that u_n^* solves the optimal value equations and so, by assumption, for each $s \in S$ there is an action $a_s^* \in A'$ such that

$$u_n^*(s) = r_t(s, a_s^*) + \sum_{j=0}^{\infty} p_t(j|s, a_s^*) u_{n+1}^*(j).$$

Suppose that $s' \in S$ is another element with $s' \geq s$. By the induction hypothesis, u_{n+1}^* is a nondecreasing function and so $(u_{n+1}^*(j); j \geq 0)$ is a nondecreasing sequence of real numbers. Also, by assumption (b), we know that

$$\sum_{j=k}^{\infty} p_t(j|s, a_s^*) = q_t(k|s, a_s^*) \leq q_t(k|s', a_s^*) = \sum_{j=k}^{\infty} p_t(j|s', a)$$

for every $k \geq 1$, while

$$\sum_{j=0}^{\infty} p_t(j|s, a_s^*) = 1 = \sum_{j=0}^{\infty} p_t(j|s', a_s^*),$$

since $p_t(\cdot|s,a)$ defines a probability distribution on S for all $s \in S$ and $a \in A'$. Invoking Lemma 5.2 then allows us to conclude that

$$\sum_{j=1}^{\infty} p_t(j|s, a_s^*) u_{n+1}^*(j) \leq \sum_{j=1}^{\infty} p_t(j|s', a_s^*) u_{n+1}^*(j).$$

Since $r_t(s,a)$ is nondecreasing in s for every a, it follows that

$$\begin{aligned} u_n^*(s) &\leq r_t(s', a_s^*) + \sum_{j=1}^{\infty} p_t(j|s', a_s^*) u_{n+1}^*(j) \\ &\leq \max_{a \in A'} r_t(s', a) + \sum_{j=1}^{\infty} p_t(j|s', a) u_{n+1}^*(j) \\ &= u_n^*(s'), \end{aligned}$$

which confirms that u_n^* is nondecreasing and completes the induction. □

The next theorem provides a set of conditions which are sufficient to guarantee the existence of at least one optimal policy that is monotone.

Theorem 5.9. *Suppose that*

(1) $r_t(s,a)$ is nondecreasing in s for all $a \in A'$;

(2) $r_t(s,a)$ is superadditive (subadditive) on $S \times A$;

(3) $q_t(k|s,a)$ *is nondecreasing in* s *for all* $k \in S$ *and* $a \in A'$*;*

(4) $q_t(k|s,a)$ *is superadditive (subadditive) on* $S \times A$ *for every* $k \in S$*;*

(5) $r_N(s)$ *is a nondecreasing in* s*.*

Then there exist optimal decision rules $d_t^*(s)$ *which are nondecreasing (nonincreasing) in* s *for* $t = 1, \cdots, N-1$*.*

Proof. The assumption that q_t is superadditive implies that for every $k \geq 1$

$$\sum_{j=k}^{\infty} \left[p_t(j|s^-,a^-) + p_t(j|s^+,a^+)\right] \geq \sum_{j=k}^{\infty} \left[p_t(j|s^-,a^+) + p_t(j|s^+,a^-)\right]$$

for all $s^- \leq s^+$, $a^- \leq a^+$. However, since Proposition 5.1 implies that the optimal value functions $u_t^*(s)$ are nondecreasing, we can use Lemma 5.2 to conclude that

$$\sum_{j=k}^{\infty} \left[p_t(j|s^-,a^-) + p_t(j|s^+,a^+)\right] u_t(j) \geq \sum_{j=k}^{\infty} \left[p_t(j|s^-,a^+) + p_t(j|s^+,a^-)\right] u_t(j),$$

which shows that the functions $\sum_{j\geq 0} p_t(j|s,a)u_t(j)$ are superadditive for every $t = 1, \cdots, N-1$. By assumption, the rewards $r_t(s,a)$ are superadditive and thus so are the functions

$$w_t(s,a) \equiv r_t(s,a) + \sum_{j=0}^{\infty} p_t(j|s,a)u_t^*(j)$$

(since sums of superadditive functions are superadditive). It then follows from Lemma 5.1 that if we define decision rules $d_t(s)$ by setting

$$d_t(s) = \max\left\{a^* \in \arg\max_{A_s} w_t(s,a)\right\},$$

then d_t is a nondecreasing function and so the policy $\pi = (d_1, \cdots, d_{N-1})$ is optimal and monotone.

□

Example 5.1. ***A Price Determination Model.*** *Suppose that a manager wishes to determine optimal price levels based on current sales with the goal of maximizing revenue over some fixed period. This problem can be formulated as a Markov decision process by taking*

- *Decision epochs:* $T = \{1, 2, \cdots, N\}$.
- *States:* $S = \mathbb{N}$*, where* s_t *is the number of products sold in the previous month.*
- *Actions:* $A_s = A' = [a_L, a_U]$*, where* a_L *and* a_M *are the minimum and maximum price levels and* $a_t = a$ *is the price assigned to the product during the* t*'th decision epoch.*
- *Rewards: let* $r_t(s,a)$ *denote the expected revenue in month* t *if the previous month's sales were* s *and the price is set to* a *in the current month. For simplicity, assume that the product has a limited shelf life and that* $r_N(s) = 0$ *for all* $s \geq 0$*.*
- *Transition probabilities: let* $p_t(j|s,a)$ *be the probability that* j *units are sold in month* t *at price* a *given that* s *units were sold in the previous month.*

If we assume that $r_t(s,a)$ and $p_t(s,a)$ are continuous functions of a, then Theorem 5.8 guarantees the existence of an optimal policy which is deterministic and Markov. Furthermore, if the conditions of Theorem 5.9 are satisfied, then we can conclude that there is an optimal policy which is nondecreasing in s, i.e., the greater the level of sales in the previous month, the higher the price should be in the present month. Since $r_N(s) \equiv 0$, condition (5) is trivially satisfied and so we need only consider the first four conditions.

(1) To stipulate that $r_t(s,a)$ is a nondecreasing function of s means that for each price a, the expected revenue in the current month will be an increasing function of the previous month's sales.

(2) Superadditivity of $r_t(s,a)$ requires that for $s^+ \geq s^-$ and $a^+ \geq a^-$,

$$r_t(s^+,a^+) - r_t(s^+,a^-) \geq r_t(s^-,a^+) - r_t(s^-,a^-),$$

which says that the effect of increasing the price on revenue is greater when the previous month's sales have been greater.

(3) To stipulate that $q_t(k|s,a)$ is nondecreasing means that sales one month ahead are stochastically increasing with respect to current sales.

(4) Superadditivity of $q_t(j|s,a)$ requires that for every $k \geq 0$,

$$q_t(k|s^+,a^+) - q_t(k|s^+,a^-) \geq q_t(k|s^-,a^+) - q_t(k|s^-,a^-),$$

which says that the incremental effect of a price increase on the probability that sales exceed a fixed level is greater if current sales are greater.

The backward induction algorithm of section 5.5 can be simplified when it is known that a monotone optimal policy exists. Assuming that this is true and that the state space $S = \{0, 1, \cdots, M\}$ is finite, the **Monotone Backward Induction Algorithm** consists of the following steps:

1. Set $t = N$ and
$$u_N^*(s) = r_N(s) \quad \text{for all } s \in S.$$

2. Substitute $t-1$ for t, set $s = 1$ and $A'_1 = A'$, and
$$u_t^*(s) = \max_{a \in A'_s} \left\{ r_t(s,a) + \sum_{j \in S} p_t(j|s,a) u_{t+1}^*(j) \right\} \tag{5.17}$$
$$A_{s,t}^* = \arg\max_{a \in A'_s} \left\{ r_t(s,a) + \sum_{j \in S} p_t(j|s,a) u_{t+1}^*(j) \right\}. \tag{5.18}$$

3. If $s = M$, go to Step 4. Otherwise, set
$$A'_{s+1} = \{a \in A' : a \geq [\max A_{s,t}]\}$$

4. Stop if $t = 1$. Otherwise return to step 2.

A monotone optimal policy can then be found by forming decision rules which select actions from $A_{s,t}^*$ in state s at decision epoch t. One advantage that this approach has over the standard backward induction algorithm is that the maximizations performed at each epoch t are carried out over sets A'_s which can shrink as s increases.

Chapter 6

Infinite-Horizon Models: Foundations

6.1 Assumptions and Definitions

Our focus in this chapter will be on infinite-horizon Markov decision processes with stationary rewards and transition probabilities. Throughout we will assume that the set of decision epochs is $T = \{1, 2, \cdots\}$ and that the reward functions and transition probabilities do not vary over time, i.e., for every $t \in T$ we have $r_t(s, a) = r(s, a)$ and $p_t(j|s, a) = p(j|s, a)$. We will be especially interested in stationary policies, which which we denote $\pi = d^\infty = (d, d, \cdots)$, meaning that the same decision rule d is used in every decision epoch.

As before, let X_t denote the random state occupied by the process at time t, let Y_t denote the action taken in this decision epoch, and let Z_t be the history of the process up to time t. If $\pi = (d_1, d_2, \cdots)$ is a deterministic policy, then Y_t will be a deterministic function either of the system state at time t, i.e., $Y_t = d_t(X_t)$, or of the process history up to time t, i.e., $Y_t = d_t(Z_t)$, depending on whether π is Markov or history-dependent. If, instead, π is a randomized policy, then Y_t is chosen randomly according to a probability distribution q_{d_t} on the set of actions and this distribution will take the form

$$\mathbb{P}(Y_t = a) = \begin{cases} q_{d_t(X_t)}(a) & \text{if } \pi \text{ is Markov} \\ \\ q_{d_t(Z_t)}(a) & \text{if } \pi \text{ is history-dependent.} \end{cases}$$

Lastly, if π is Markov, then $(X_t, r(X_t, Y_t); t \geq 1)$ is called a Markov reward process.

There are several performance metrics that can be used to assign a value to a policy $\pi \in \Pi^{HR}$ which depend on the initial state of the system. The first of these is the **expected total reward** of π, which is defined to be

$$v^\pi(s) \equiv \lim_{N\to\infty} \mathbb{E}_s^\pi \left[\sum_{t=1}^{N} r(X_t, Y_t) \right] = \lim_{N\to\infty} v_{N+1}^\pi(s), \tag{6.1}$$

where, as before, $v_{N+1}^\pi(s)$ is the total expected reward in a model restricted to N decision epochs and terminal reward 0. In general, this limit need not exist or it may exist but be equal to $\pm\infty$. Moreover, even when the limit exists, it need not be the case that the limit and expectation can be interchanged, i.e., we cannot simply assume that

$$\lim_{N\to\infty} \mathbb{E}_s^\pi \left[\sum_{t=1}^{N} r(X_t, Y_t) \right] = \mathbb{E}_s^\pi \left[\sum_{t=1}^{\infty} r(X_t, Y_t) \right], \tag{6.2}$$

although we will give some conditions under which this identity does hold. Because of these complications, the total expected value is not always suitable for infinite-horizon MDPs.

An alternative performance metric is the **expected total discounted reward** of π, which is defined to be

$$v_\lambda^\pi(s) \equiv \lim_{N\to\infty} \mathbb{E}_s^\pi \left[\sum_{t=1}^N \lambda^{t-1} r(X_t, Y_t) \right], \tag{6.3}$$

where $\lambda \in [0, 1)$ is the **discount rate**. Notice that the limit is guaranteed to exist if the absolute value of the reward function is uniformly bounded, i.e., if $|r(s, a)| \leq M$ for all $s \in S$ and $a \in A_s$, in which case the total expected discounted reward is also uniformly bounded: $v_\lambda^\pi(s) \leq M/(1-\lambda)$. Furthermore, if equality does hold in (6.2), then we also have

$$v^\pi(s) = \lim_{\lambda\uparrow 1} v_\lambda^\pi(s).$$

Our last performance metric is the **average reward** (**average gain**) of π, which is defined by

$$g^\pi(s) \equiv \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_s^\pi \left[\sum_{t=1}^N r(X_t, Y_t) \right] = \lim_{N\to\infty} \frac{1}{N} v_{N+1}^\pi(s), \tag{6.4}$$

provided that the limit exists. If the limit in (6.4) does not exist, then we can define the **lim inf average reward** g_-^π and the **lim sup average reward** g_+^{pi} by

$$g_-^\pi(s) \equiv \liminf_{N\to\infty} \frac{1}{N} v_{N+1}^\pi(s), \qquad g_+^\pi(s) \equiv \limsup_{N\to\infty} \frac{1}{N} v_{N+1}^\pi(s).$$

These are guaranteed to exist (although they could be infinite) and they provide upper and lower bounds on the average reward attainable by policy π.

Example 6.1. *(Puterman, Example 5.1.2) Let $S = \{1, 2, \cdots\}$ and $A_s = \{a\}$ for all $s \geq 1$, i.e., there is a single action available in all states, and suppose that the transition probabilities are equal to*

$$p_t(j|s, a) = \begin{cases} 1 & \textit{if } j = s + 1 \\ 0 & \textit{otherwise,} \end{cases}$$

so that the state increases by 1 at each time point, and that the rewards are equal to

$$r(s, a) = (-1)^{s+1} s.$$

Since there is only action available in each state, it follows that there is only one policy $\pi = d^\infty$ where $d(s) = a$ for all $s \geq 1$. Furthermore, a simple calculation shows that the total expected reward up to time N when the initial state is $s_1 = 1$ is equal to

$$v_N^\pi = \begin{cases} k & \textit{if } N = 2k \textit{ is even} \\ -k & \textit{if } N = 2k + 1 \textit{ is odd} \end{cases}$$

and so the total expected reward

$$v^\pi(1) \equiv \lim_{N\to\infty} v_N^\pi(1)$$

does not exist. Similarly, the gain $g^\pi(1)$ does not exist because

$$g_+^\pi(s) = \limsup_{N\to\infty} \frac{1}{N} v_{N+1}^\pi(s) = \frac{1}{2}$$

and

$$g_-^\pi(s) = \liminf_{N\to\infty} \frac{1}{N} v_{N+1}^\pi(s) = -\frac{1}{2}.$$

On the other hand, the expected total discounted reward for this policy does exist and is finite for every $\lambda \in [0,1)$ *since*

$$\begin{aligned} v_\lambda^\pi(1) &= \lim_{N\to\infty} \sum_{t=1}^N \lambda^{t-1}(-1)^{t-1}t \;=\; -\lim_{N\to\infty} \frac{d}{d\lambda}\left(\sum_{t=1}^N (-\lambda)^t\right) \;=\; -\lim_{N\to\infty}\frac{d}{d\lambda}\left(-\lambda\frac{1-(-\lambda)^N}{1+\lambda}\right) \\ &= \lim_{N\to\infty}\left(\frac{1-(-\lambda)^N}{1+\lambda} + \lambda\frac{(1+\lambda)N(-\lambda)^{N-1} - (1-(-\lambda)^N)}{(1+\lambda)^2}\right) \\ &= \frac{1}{1+\lambda} - \frac{\lambda}{(1+\lambda)^2} = \frac{1}{(1+\lambda)^2}. \end{aligned}$$

Also, notice that the limit as λ *increases to* 1 *of the expected total discounted reward exists and is finite,*

$$\lim_{\lambda\uparrow 1} v_\lambda^\pi(1) = \frac{1}{4},$$

even though the total expected reward is not defined in this case.

6.2 The Expected Total Reward Criterion

Although we saw in the previous section that the limit that defines the total expected reward,

$$v^\pi(s) = \lim_{N\to\infty} v_N^\pi(s), \tag{6.5}$$

does not always exist, there are relatively general conditions on a Markov decision problem which suffice to guarantee convergence of this sequence within the extended real numbers $[-\infty,\infty]$. Define the quantities

$$v_+^\pi(s) \equiv \mathbb{E}_s^\pi\left[\sum_{t=1}^\infty r^+(X_t,Y_t)\right]$$

and

$$v_-^\pi(s) \equiv \mathbb{E}_s^\pi\left[\sum_{t=1}^\infty r^-(X_t,Y_t)\right],$$

where $r^+(s,a) \equiv \max\{r(s,a),0\}$ and $r^-(s,a) \equiv \max\{-r(s,a),0\}$ are the positive and negative parts of $r(s,a)$, respectively. Since $r^+(s,a)$ and $r^-(s,a)$ are both non-negative, the quantities $v_+^\pi(s)$ and $v_-^\pi(s)$ are guaranteed to exist, although they could be equal to ∞.

For the total expected reward to be well-defined, we need to rule out the possibility that the sequence $(v_N^\pi(s); N \geq 1)$ assumes arbitrarily large positive and arbitrarily large negative values. To this end, we will consider Markov decision processes that satisfy the following condition. Suppose that for every policy $\pi \in \Pi^{HR}$ and every $s \in S$, at least one of the the quantities $v_+^\pi(s)$ and $v_-^\pi(s)$ is finite. Then the limit (6.5) exists and the total expected reward is equal to

$$v^\pi(s) = v_+^\pi(s) - v_-^\pi(s). \tag{6.6}$$

Although this condition excludes certain kinds of Markov decision processes, it is sufficiently general that there are several classes of models encountered in practice which do satisfy it. We will introduce three such classes in the following definition.

Definition 6.1. *Suppose that* $\{\mathbb{N}, S, A_s, r(s,a), p(j|s,a)\}$ *is an infinite horizon Markov decision process with stationary rewards and transition probabilities.*

(i) We will say that the process belongs to the class of ***positive bounded models*** *if for each* $s \in S$*, there exists an* $a \in A_s$ *such that* $r(s,a) \geq 0$ *and* $v_+^\pi(s)$ *is finite for all policies* $\pi \in \Pi^{HR}$.

(ii) We will say that the process belongs to the class of ***negative models*** *if for each $s \in S$ and $a \in A_s$ the reward $r(s,a) \leq 0$ is non-positive, and for some policy $\pi \in \Pi^{HR}$ we have $v^\pi(s) > -\infty$ for all $s \in S$.*

(iii) We will say that the process belongs to the class of ***convergent models*** *if for each $s \in S$ both $v_+^\pi(s)$ and $v_-^\pi(s)$ are finite for all $s \in S$.*

Positive bounded models have the property that there exists at least one stationary policy with a finite non-negative total expected reward. Indeed, we can construct such a policy by defining a decision rule $d(s) = a_s$, where a_s is an action such that $r(s, a_s) \geq 0$. Furthermore, if the state space S is finite, then such a model also has the property that under every policy the system eventually absorbs in a class of states in which the rewards are non-negative. Were this not the case, there would be a policy under which the process had a positive probability of visiting a state with a positive reward infinitely often, in which case the quantity $v_+^\pi(s)$ would be infinite for some s. Examples of positive bounded models include many optimal stopping problems as well as problems in which the goal is to maximize the probability of reaching a certain desirable state.

Negative models are in some sense more restricted than positive bounded models since for the former we have $v_+^\pi(s) = 0$ for every policy π and every state s. Also, while it may be the case that $v_-^\pi(s) = \infty$ for some policies and states, the existence of at least one policy π for which $v_-^\pi(s)$ is finite for all $s \in S$ means that we can use the total expected reward to distinguish between policies. Indeed, the goal becomes to find a policy π that minimizes $v_-^\pi(s)$ for all initial states, where we often interpret $v_-^\pi(s)$ as a cost. Other examples include Markov decision processes in which the aim is either to minimize the probability of reaching an undesirable state (e.g., bankruptcy or a disease outbreak) or to minimize the expected time to reach a desirable state (e.g., time to recovery following illness).

Finally, the class of convergent models is the most restrictive and has the property that the expectation

$$\mathbb{E}_s^\pi \left[\sum_{t=1}^{\infty} |r(X_t, Y_t)| \right] = v_+^\pi(s) + v_-^\pi(s) < \infty$$

is finite for all $\pi \in \Pi^{HR}$ and $s \in S$.

6.3 The Expected Total Discounted Reward Criterion

The expected total discounted reward can be interpreted in several ways. On the one hand, discounting is natural whenever the value of a reward changes over time. For example, an immediate cash payment of 100 dollars may be worth more than a delayed reward of the same amount if the recipient can earn additional income by investing the money as soon as it is received. Similarly, offspring born during the current generation may be ‘worth more’ (in an evolutionary sense) than offspring produced in a future generation if additional parental genes are transmitted to the population through reproduction by the offspring. In both scenarios, the discount rate λ is defined to be the present value of a unit of reward (e.g., one dollar or one offspring) received one period in the future. For technical reasons we generally assume that $\lambda \in [0,1)$, but in principal the discount rate could exceed unity as is true, for example, of reproductive value in a declining population.

The expected total discounted reward can also be interpreted as the expected total reward of a random horizon Markov decision process with a geometrically-distributed horizon length. Random horizon MDPs with horizon lengths that are independent of the history of the process up until the final decision epoch can be constructed using the following procedure. Let τ be a random variable with values in $\mathbb{N} \cup \{\infty\}$ and probability mass function $p_\tau(n) = \mathbb{P}(\tau = n)$ and suppose that $(\mathbb{N}, S, A_s, p(j|s,a), r(s,a))$ is an infinite horizon MDP with stationary rewards and transition probabilities. A random horizon MDP $(\mathbb{N}, S', A_s, p'_t(j|s,a), r(s,a))$ with horizon length τ can be constructed by adding a cemetery state Δ to the state space, $S' = S \cup \{\Delta\}$, and letting $A_\Delta = \{a_\Delta\}$, and then modifying the transition probabilities so that

$$p'_t(j|s,a) = \begin{cases} (1-h(t))p(j|s,a) & \text{if } j, s \neq \Delta \\ h(t) & \text{if } j = \Delta \text{ and } s \neq \Delta \\ 1 & \text{if } j = s = \Delta \end{cases}$$

where

$$h(t) = \mathbb{P}(\tau = t | \tau \geq t) = \frac{\mathbb{P}(\tau = t)}{\mathbb{P}(\tau \geq t)}$$

is the probability that the process terminates at the end of decision epoch t given that it has persisted up until time t. ($h(t)$ is the discrete-time hazard function for the variable τ.) We also need to extend the reward function to S' by setting $r(\Delta, a_\Delta) = 0$. The resulting process coincides with the original MDP up to the random time τ, after which it is absorbed by Δ. The total expected value of a policy π for the random horizon process is denoted v^π_τ and is equal to

$$v^\pi_\tau(s) = \mathbb{E}^\pi_s\left[\sum_{t=1}^{\tau} r(X_t, Y_t)\right] = \mathbb{E}^\pi_s\left[\sum_{n=1}^{\infty} \mathbb{P}(\tau = n) \sum_{t=1}^{n} r(X_t, Y_t)\right]$$

provided that the sums and expectations are well-defined.

Although the expected total reward of the random horizon process is not guaranteed to exist even if τ is finite, we will show that it does exist and is equal to the expected total discounted reward when τ is geometrically distributed with parameter λ, i.e., when

$$p_\tau(n) = (1-\lambda)\lambda^{n-1}, \quad n \geq 1.$$

In other words, we now assume that the process has probability $1-\lambda$ of terminating in each decision epoch, starting with decision epoch 1. Thus λ can be interpreted as a survival or persistence probability. This leads to the following result.

Proposition 6.1. *Consider a random horizon MDP with uniformly bounded rewards and assume that the horizon length τ has a geometric distribution with parameter $\lambda < 1$. Then $v^\pi_\tau(s) = v^\pi_\lambda(s)$ for any policy $\pi \in \Pi^{HR}$ and any state s, where as above we define $v^\pi_\lambda(s)$ to be the expected total discounted reward for this policy when used in the corresponding infinite horizon MDP.*

Proof. Since τ is geometrically-distributed and independent of the process $((X_t, Y_t); t \geq 1)$, the expected total reward obtained by using policy π in the random horizon process is equal to

$$\begin{aligned} v^\pi_\tau(s) &= \mathbb{E}^\pi_s\left[\sum_{n=1}^{\infty}(1-\lambda)\lambda^{n-1}\sum_{t=1}^{n} r(X_t, Y_t)\right] \\ &= \mathbb{E}^\pi_s\left[\sum_{t=1}^{\infty} r(X_t, Y_t)\sum_{n=t}^{\infty}(1-\lambda)\lambda^{n-1}\right] = \mathbb{E}^\pi_s\left[\sum_{t=1}^{\infty}\lambda^{t-1} r(X_t, Y_t)\right] = v^\pi_\lambda(s), \end{aligned}$$

provided that we can interchange the order of summation over n and t. However, this interchange can be justified by Fubini's theorem once we observe that

$$\sum_{n=1}^{\infty}\sum_{t=1}^{n}\left|(1-\lambda)\lambda^{n-1} r(X_t, Y_t)\right| = \frac{M}{1-\lambda} < \infty,$$

where $M < \infty$ is chosen so that $r(s,a) < M$ for all $s \in S$ and $a \in A_s$. □

6.4 Optimality Criteria

An important difference between finite-horizon and infinite-horizon Markov decision problems is that there are many more optimality criteria in use for the latter. Six of these are introduced in the following definition and we will see even more below.

Definition 6.2. *Suppose that $\pi^* \in \Pi^{HR}$ is a history-dependent, randomized policy for an infinite-horizon Markov decision process.*

(1) *The value of the MDP is defined to be the quantity*

$$v^*(s) \equiv \sup_{\pi \in \Pi^{HR}} v^\pi(s),$$

*provided that the expected total reward $v^\pi(s)$ exists for every policy π and every state $s \in S$. Furthermore, in this case, a policy π^* is said to be **total reward optimal** if*

$$v^{\pi^*}(s) \geq v^\pi(s) \quad \text{for every } s \in S \text{ and all } \pi \in \Pi^{HR}.$$

(2) *Let $\lambda \in [0,1)$ be given. The λ-discounted value of the MDP is defined to be the quantity*

$$v_\lambda^*(s) \equiv \sup_{\pi \in \Pi^{HR}} v_\lambda^\pi(s),$$

*provided that the expected total discounted reward $v_\lambda^\pi(s)$ exists for every policy π and every state $s \in S$. In this case, a policy π^* is said to be **discount optimal** if*

$$v_\lambda^{\pi^*}(s) \geq v_\lambda^\pi(s) \quad \text{for every } s \in S \text{ and all } \pi \in \Pi^{HR}.$$

(3) *The optimal gain of the MDP is defined to be the quantity*

$$g^*(s) \equiv \sup_{\pi \in \Pi^{HR}} v^\pi(s),$$

*provided that the average gain $g^\pi(s)$ exists for every policy π and every state $s \in S$. In this case, a policy π^* is said to be **gain optimal** or **average optimal** if*

$$g^{\pi^*}(s) \geq g^\pi(s) \quad \text{for every } s \in S \text{ and all } \pi \in \Pi^{HR}.$$

(4) *A policy π^* is said to be **limit point average optimal** if*

$$g_-^{\pi^*}(s) = \liminf_{N\to\infty} \frac{1}{N} v_{N+1}^{\pi^*} \geq \limsup_{N\to\infty} \frac{1}{N} v_{N+1}^{\pi} = g_+^\pi(s)$$

for every $s \in S$ and all $\pi \in \Pi^{HR}$.

(5) *A policy π^* is said to be **lim sup average optimal** if*

$$g_+^{\pi^*}(s) \geq g_+^\pi(s) \quad \text{for every } s \in S \text{ and all } \pi \in \Pi^{HR}.$$

(6) *Similarly, a policy π^* is said to be **lim inf average optimal** if*

$$g_-^{\pi^*}(s) \geq g_-^\pi(s) \quad \text{for every } s \in S \text{ and all } \pi \in \Pi^{HR}.$$

Notice that the optimality criteria defined in parts (1) - (3) can only be applied to Markov decision problems that have the property that the limits used to define the expected total reward, the expected total discounted reward, or the average gain, respectively, exist for all policies $\pi \in \Pi^{HR}$. In contrast, the criteria defined in parts (4) - (6) can be applied to every MDP since these depend only on lim sup's and lim inf's, which always exist.

Example 6.2. *Consider the infinite-horizon Markov decision process defined as follows:*

- *States:* $S = \{s_1, s_2, s_3\}$*;*
- *Action sets:* $A_{s_1} = \{a_{1,1}\}$, $A_{s_2} = \{a_{2,1}, a_{2,2}\}$ *and* $A_{s_3} = \{a_{3,1}\}$*;*
- *Transition probabilities:* $p(s_2|s_1, a_{11}) = p(s_1|s_2, a_{2,1}) = p(s_3|s_2, a_{2,2}) = p(s_2|s_3, a_{3,1}) = 1$*;*
- *Rewards:* $r(s_1, a_{1,1}) = r(s_2, a_{2,2}) = 0$, $r(s_2, a_{2,1}) = r(s_3, a_{3,1}) = 1$.

The decision maker only has a choice to make in state s_2 and this choice determines one of two stationary policies. Let d^∞ be the stationary policy which uses action $a_{2,1}$ and let e^∞ be the stationary policy which uses action $a_{2,2}$. Both of these policies have the same average gain

$$g^{d^\infty}(s) = g^{e^\infty}(s) = 0.5$$

and indeed it can be shown that $g^\pi(s) = 0.5$ for any policy $\pi \in \Pi^{HR}$ since every policy earns its user exactly 1 unit of reward every second decision epoch. This shows that every policy is average optimal for this MDP. On the other hand, different policies can generate different reward streams, e.g., if we start in state s_2, then policy d^∞ generates the reward stream $(1, 0, 1, 0, \cdots)$ while e^∞ generates the reward stream $(0, 1, 0, 1, \cdots)$.

Example 6.2 demonstrates that the average reward criterion is **unselective** because this criteria fails to differentiate between optimal policies that generate different reward streams. In such cases, it may be necessary to turn to one of the alternative optimality criteria described below.

Definition 6.3. *A policy π^* is said to be **overtaking optimal** if*

$$\liminf_{N\to\infty} \left[v_N^{\pi^*}(s) - v_N^{\pi}(s) \right] \geq 0$$

for all $\pi \in \Pi^{HR}$ and all $s \in S$.

Since this criterion is defined in terms of a lim inf, it is applicable to all MDPs. For example, in Example 6.2 the stationary policies d^∞ and e^∞ generate the following reward streams when beginning in state s_2:

$$\begin{aligned} v_N^{d^\infty}(s_2): &\quad 1, 1, 2, 2, 3, 3, \cdots \\ v_N^{e^\infty}(s_2): &\quad 0, 1, 1, 2, 2, 3, \cdots \\ v_N^{d^\infty}(s_2) - v_N^{e^\infty}(s_2): &\quad 1, 0, 1, 0, \cdots . \end{aligned}$$

Thus

$$\begin{aligned} \liminf_{N\to\infty} \left(v_N^{d^\infty}(s) - v_N^{e^\infty}(s) \right) &= 0 \\ \liminf_{N\to\infty} \left(v_N^{e^\infty}(s) - v_N^{d^\infty}(s) \right) &= -1 \end{aligned}$$

and so d^∞ is overtaking optimal.

There are several other variants of overtaking optimality, all of which are based on comparisons of the limit points of the reward sequence or average reward sequence. An alternative approach is to base optimality on the limiting behavior of the discounted rewards as the discount rate λ increases to 1. These are called **sensitive discount optimality criteria**.

Definition 6.4. *As above, let π^* be a policy for an infinite-horizon MDP.*

1. *π^* is said to be n-discount optimal for a constant $n \geq -1$ if*

$$\liminf_{\lambda \uparrow 1} (1-\lambda)^{-n} \left[v_\lambda^{\pi^*}(s) - v_\lambda^\pi(s) \right] \geq 0$$

*for all $\pi \in \Pi^{HR}$ and all $s \in S$. Furthermore, if π^* is 0-discount optimal, then π^* is also said to be **bias optimal**.*

2. *π^* is said to be **∞-discount optimal** if it is n-discount optimal for all $n \geq -1$.*

3. *π^* is said to be **1-optimal** or **Blackwell optimal** if for each $s \in S$ there exists a $\lambda^*(s)$ such that*

$$v_\lambda^{\pi^*}(s) - v_\lambda^\pi(s) \geq 0$$

*for all $\pi \in \Pi^{HR}$ and all $\lambda \in [\lambda^*(s), 1)$. Furthermore, if the quantity $\lambda^* \equiv \sup_s \lambda^*(s) < 1$, then the policy is said to be **strongly Blackwell optimal**.*

Notice that if $n_1 > n_2$ then n_1-discount optimality implies n_2-discount optimality, i.e., n_1-discount optimality is more sensitive than n_2-discount optimality. Furthermore, Blackwell optimality implies n-discount optimality for all $n \geq 1$.

6.5 Markov policies

Our aim in this section is to show that for any stationary infinite-horizon Markov decision problem there is always a randomized Markov policy which has the same expected total reward, the same expected total discounted reward, and also the same average award. Later, we will show that we can also replace randomized policies by deterministic policies for most Markov decision problems.

Theorem 6.1. *Let $\pi = (d^1, d^2, \cdots) \in \Pi^{HR}$. Then, for each $s \in S$, there exists a policy $\pi' = (d^1, d^2, \cdots) \in \Pi^{MR}$ satisfying*

$$\mathbb{P}^{\pi'}\left\{X_t = j, Y_t = a | X_1 = s\right\} = \mathbb{P}^{\pi}\left\{X_t = j, Y_t = a | X_1 = s\right\} \tag{6.7}$$

for every $t \geq 1$.

Proof. Fix $s \in S$ and define the randomized Markov decision rule d'_t by

$$q_{d'_t(j)}(a) \equiv \mathbb{P}^\pi\left\{Y_t = a | X_t = j, X_1 = s\right\}$$

for $j \in S$ and $t \geq 1$. Let $\pi' = (d'_1, d'_2, \cdots) \in \Pi^{MR}$ and observe that

$$\begin{aligned} \mathbb{P}^{\pi'}\left\{Y_t = a | X_t = j\right\} &= \mathbb{P}^{\pi'}\left\{Y_t = a | X_t = j, X_1 = s\right\} \\ &= \mathbb{P}^{\pi}\left\{Y_t = a | X_t = j, X_1 = s\right\}. \end{aligned}$$

We show that the identity (6.7) holds by (forward) induction on t. First, observe that when $t = 1$, this identity follows from the fact that

$$\begin{aligned}\mathbb{P}^{\pi'}\left\{X_1 = j, Y_1 = a|X_1 = s\right\} &= \mathbb{P}^{\pi'}\left\{Y_1 = a|X_1 = s\right\} \\ &= \mathbb{P}^{\pi}\left\{Y_1 = a|X_1 = s\right\} \;=\; \mathbb{P}^{\pi}\left\{X_1 = j, Y_1 = a|X_1 = s\right\},\end{aligned}$$

which is clearly true. Next, suppose that (6.7) holds for $t = 1, \cdots, n-1$. Then

$$\begin{aligned}\mathbb{P}^{\pi'}\left\{X_n = j|X_1 = s\right\} &= \sum_{k\in S}\sum_{a\in A_k}\mathbb{P}^{\pi'}\left\{X_{n-1} = k, Y_{n-1} = a|X_1 = s\right\}p(j|k,a) \\ &= \sum_{k\in S}\sum_{a\in A_k}\mathbb{P}^{\pi}\left\{X_{n-1} = k, Y_{n-1} = a|X_1 = s\right\}p(j|k,a) \\ &= \mathbb{P}^{\pi}\left\{X_n = j|X_1 = s\right\},\end{aligned}$$

where the second inequality follows from the induction hypothesis. Consequently,

$$\begin{aligned}\mathbb{P}^{\pi'}\left\{X_n = j, Y_n = a|X_1 = s\right\} &= \mathbb{P}^{\pi'}\left\{Y_n = a|X_n = j\right\}\mathbb{P}^{\pi'}\left\{X_n = j|X_1 = s\right\} \\ &= \mathbb{P}^{\pi'}\left\{Y_n = a|X_n = j\right\}\mathbb{P}^{\pi'}\left\{X_n = j|X_1 = s\right\} \\ &= \mathbb{P}^{\pi}\left\{X_n = j, Y_n = a|X_1 = s\right\},\end{aligned}$$

which completes the induction argument.

□

A similar result holds when the initial state of the MDP is itself randomly distributed.

Corollary 6.1. *For each distribution ν of X_1 and any history-dependent policy π, there exists a randomized Markov policy π' for which*

$$\mathbb{P}^{\pi'}\left\{X_t = j, Y_t = a\right\} = \mathbb{P}^{\pi}\left\{X_t = j, Y_t = a\right\}$$

for all $j \in S$, $a \in A_j$, and $t \geq 1$.

It should be emphasized that Theorem 6.1 and Corollary 6.1 only guarantee that the marginal probabilities $\mathbb{P}^{\pi'}\left\{X_t = j, Y_t = a\right\}$ (also known as the **state-action frequencies**) are the same as those obtained under policy π. In contrast, the joint probabilities of the states and/or actions at multiple decision epochs will usually differ between the two policies. However, the equivalence of the marginal probabilities is enough to ensure that the expected value, expected discounted value, and the expected gain are the same under either policy since the expectations

$$\begin{aligned}v_N^{\pi}(s) &= \sum_{t=1}^{N-1}\sum_{j\in S}\sum_{a\in A_j} r(j,a)\mathbb{P}(X_t = j, Y_t = a) \\ v_\lambda^{\pi}(s) &= \sum_{t=1}^{\infty}\sum_{j\in S}\sum_{a\in A_j} \lambda^{t-1}r(j,a)\mathbb{P}(X_t = j, Y_t = a),\end{aligned}$$

only depend on the state-action frequencies.

Theorem 6.2. *Given any $\pi \in \Pi^{HR}$ and any $s \in S$, there exists a policy $\pi' \in \Pi^{MR}$ such that*

(a) $v_N^{\pi'}(s) = v_N^{\pi}(s)$ for $N \geq 1$ and $v^{\pi'}(s) = v^{\pi}(s)$ when the relevant limits exist;

(b) $v_\lambda^{\pi'}(s) = v_\lambda^{\pi}(s)$ for $\lambda \in [0,1)$;

(c) $g_\pm^{\pi'}(s) = g_\pm^{\pi}(s)$ and $g^{\pi'}(s) = g^{\pi}(s)$, when the relevant limits exist.

Chapter 7

Discounted Markov Decision Processes

7.1 Notation and Conventions

This chapter will be concerned with infinite-horizon Markov decision processes which satisfy the following conditions:

(i) discrete state space: S is finite or countably infinite;

(ii) stationary rewards and transition probabilities: $r(s, a)$ and $p(j|s, a)$ do not vary over time;

(iii) bounded rewards: for some $M < \infty$, $|r(s, a)| \leq M$ for all $s \in S$ and $a \in A_s$;

(iv) discounting: future rewards are discounted at rate $\lambda \in [0, 1)$.

Before venturing into the theory of discounted MDPs, we need to introduce some terminology and notation. Throughout the chapter we will let V denote the set of bounded real-valued functions on S with norm

$$||v|| \equiv \sup_{s \in S} |v(s)| < \infty. \tag{7.1}$$

Notice that V is a vector space and that we can identify each element $v \in V$ with a column vector $(v_1, v_2, \cdots)^T$, where $v_i = v(s_i)$ and $S = \{s_1, s_2, \cdots\}$. This amounts to choosing a basis for V consisting of elements $\{e_1, e_2, \cdots\}$, where the e_i is defined by the condition $e_i(s_j) = \delta_{ij}$. Also, we will let $e \in V$ denote the constant function equal to 1 everywhere, i.e., $e(s) = 1$ for all $s \in S$.

Because V is a vector space, we can represent linear operators on V by matrices indexed by the elements of S, i.e., if $H : V \to V$ is a linear operator on V, then H can be identified with a $S \times S$ matrix with components $H_{sj} = H(j|s)$ such that for every element $v \in V$,

$$(Hv)(s) = \sum_{j \in S} H(j|s) v_j,$$

i.e., the element Hv is obtained by left multiplication of the column vector corresponding to v by the matrix corresponding to H. Furthermore, because we have fixed a basis for V, we can and will use H interchangeably to mean both the operator and the matrix corresponding to that operator. We can then define a matrix norm on such operators by

$$||H|| \equiv \sup_{s \in S} \sum_{j \in S} |H(j|s)| \tag{7.2}$$

and we will say that H is a bounded linear operator if $||H|| < \infty$. For example, if S is finite, then all linear operators on V are bounded. Likewise, every stochastic matrix P on S corresponds to a bounded linear operator on V with norm $||P|| = 1$.

The matrix norm defined in (7.2) has several important properties. First note that it is consistent with the norm on V is the sense that if H is a bounded linear operator on V, then for any element $v \in V$, we have

$$||Hv|| = \sup_{s\in S} |(Hv)(s)| = \sup_{s\in S} \left|\sum_{j\in S} H(j|s)v_j\right| \leq \sup_{s} \sum_{j\in S} |H(j|s)| \cdot ||v|| = ||H|| \cdot ||v||.$$

Similarly, it can be shown that if A and B are bounded linear operators on V, then both the sum $A+B$ and the product AB are bounded linear operators on V and

$$\begin{aligned} ||A+B|| &\leq ||A|| + ||B|| && (7.3) \\ ||AB|| &\leq ||A|| \cdot ||B|| && (7.4) \end{aligned}$$

so that, in particular,

$$||A^n|| \leq ||A||^n$$

for any positive integer $n \geq 1$. Here, the product of two bounded linear operators A and B is defined both in a functional sense, i.e., AB is the operator which maps an element $v \in V$ to the element $A(Bv)$, and in an algebraic sense via matrix multiplication of the matrices representing the two operators. Furthermore, we say that a sequence of bounded linear operators $(H_n; n \geq 1)$ converges in norm to a bounded linear operator H if the following is true

$$\lim_{n\to\infty} ||H_n - H|| = 0;$$

notice, as well, that convergence in norm implies that

$$\lim_{n\to\infty} ||H_n v - Hv|| = 0$$

for all $v \in V$. The set of all bounded linear operators on V is said to be a **Banach algebra**, meaning that it is a normed vector space which is complete (i.e., all Cauchy sequences converge) and which satisfies inequality (7.4).

Given a MDP satisfying the above assumptions and a Markovian decision rule d for that process, let the quantities $r_d(s)$ and $p_d(s)$ be defined by

$$r_d(s) \equiv r(s, d(s)) \quad \text{and} \quad p_d(j|s) \equiv p(j|s, d(s))$$

if $d \in D^{MD}$ is deterministic, or

$$r_d(s) \equiv \sum_{a\in A_s} q_{d(s)}(a) r(s,a) \quad \text{and} \quad p_d(j|s) \equiv \sum_{a\in A_s} q_{d(s)}(a) p(j|s,a)$$

if $d \in D^{MR}$ is randomized. In either case, let $r_d \in V$ be the vector with components $r_d(s)$ and let P_d be the stochastic matrix with components $p_d(j|s)$. We will call r_d the **reward vector** and P_d the **transition probability matrix** corresponding to the decision rule d. For future reference, notice that if $v \in V$, then $r_d + \lambda P_d v \in V$.

If $\pi = (d_1, d_2, \cdots)$ is a Markovian policy (randomized or deterministic), then the (s,j) component of the t-step transition matrix P_π^t is given by the following formula:

$$P_\pi^t(j|s) \equiv \mathbb{P}^\pi(X_{t+1} = j | X_1 = s) = \left[P_{d_1} P_{d_2} \cdots P_{d_t}\right](j|s),$$

where $[P_{d_1}P_{d_2}\cdots P_{d_t}]$ is the product of the first t transition matrices. In fact, the processes $(X_t; t \geq 1)$ and $((X_t, r_{d_t}(X_t)); t \geq 1)$ are both Markov processes under the probability distribution induced by π, and if $v \in V$ is a bounded function defined on S then the expected value of the quantity $v(X_t)$ can be calculated using the formula

$$\mathbb{E}_s^\pi\left[v(X_t)\right] = P_\pi^{t-1}v(s) = \sum_{j\in S} P_\pi^{t-1}(j|s)v(j)$$

Furthermore, the expected total discounted reward of policy π can be calculated using the formula

$$v_\lambda^\pi \;=\; \mathbb{E}^\pi\left[\sum_{t=1}^{\infty}\lambda^{t-1}r_t(X_t,Y_t)\right] \;=\; \sum_{t=1}^{\infty}\lambda^{t-1}P_\pi^{t-1}r_{d_t}.$$

7.2 Policy Evaluation

Suppose that $\pi = (d_1, d_2, \cdots) \in \Pi^{MR}$ is a Markovian policy for a MDP and observe that the expected total discounted reward for π can be expressed as

$$\begin{aligned} v_\lambda^\pi &= \sum_{t=1}^{\infty}\lambda^{t-1}P_\pi^{t-1}r_{d_t} \\ &= r_{d_1} + \lambda P_{d_1}r_{d_2} + \lambda^2 P_{d_1}P_{d_2}r_{d_3} + \lambda^3 P_{d_1}P_{d_2}P_{d_3}r_{d_4}\cdots \\ &= r_{d_1} + \lambda P_{d_1}\left(r_{d_2} + \lambda P_{d_2}r_{d_3} + \lambda^2 P_{d_2}P_{d_3}r_{d_4}\cdots\right), \end{aligned}$$

which we can rewrite as

$$v_\lambda^\pi = r_{d_1} + \lambda P_{d_1}v_\lambda^{\pi'}, \tag{7.5}$$

where $\pi^{'} = (d_2, d_3, \cdots)$ is the policy derived from π by dropping the first decision rule and executing each of the remaining rules one epoch earlier than prescribed by π. This identity can be expressed component-wise as

$$v_\lambda^\pi(s) = r_{d_1}(s) + \lambda\sum_{j\in S}p_{d_1}(j|s)v_\lambda^{\pi'}(j),$$

and we refer to this system of equations as the **policy evaluation equations**.

If $\pi = d^\infty$ is a stationary policy, then $\pi' = \pi$ and so the reward vector $v = v_\lambda^{d^\infty}$ satisfies the equation

$$v = r_d + \lambda P_d v.$$

Equivalently, if we define the operator $L_d : V \to V$ by

$$L_d v \equiv r_d + \lambda P_d v, \tag{7.6}$$

then it follows that $v_\lambda^{d^\infty}$ is a **fixed point** of L_d, i.e., $v^{d^\infty} = L_d v^{d^\infty}$. In fact, it can be shown that L_d has a unique fixed point in V, which therefore must be $v_\lambda^{d^\infty}$. This is a consequence of the fact that because P_d is a stochastic matrix, the operator $I - \lambda P_d$ is a contraction on V for any $\lambda \in (0, 1)$, i.e.,

$$||I - \lambda P_d|| < 1,$$

and therefore $I - \lambda P_d$ is invertible. Here I is the identity operator on V, i.e., $Iv = v$ for all $v \in V$.

Theorem 7.1. *For any stationary policy $d^\infty \in \Pi^{MR}$ and any $\lambda \in [0,1)$, the expected total discounted value $v_\lambda^{d^\infty}$ is the unique fixed point in V of the operator L_d defined in (7.6) and is equal to*

$$v_\lambda^{d^\infty} = (I - \lambda P_d)^{-1} r_d = \sum_{t=1}^{\infty} \lambda^{t-1} P_d^{t-1} r_d,$$

where the infinite series is guaranteed to converge in norm.

Example: We can illustrate this result with the two-state MDP described in Section 4.1 (pp. 23 - 24 of these notes). For ease of reference, we recall the formulation of this problem:

- States: $S = \{s_1, s_2\}$.
- Actions: $A_{s_1} = \{a_{11}, a_{12}\}, A_{s_2} = \{a_{21}\}$.
- Rewards:

$$r(s_1, a_{11}) = 5 \qquad r(s_1, a_{12}) = 10 \qquad r(s_2, a_{21}) = -1$$

- Transition probabilities:

$$\begin{aligned} p_t(s_1|s_1, a_{11}) &= 0.5, & p_t(s_2|s_1, a_{11}) &= 0.5 \\ p_t(s_1|s_1, a_{12}) &= 0, & p_t(s_2|s_1, a_{12}) &= 1 \\ p_t(s_1|s_2, a_{21}) &= 0, & p_t(s_2|s_2, a_{21}) &= 1 \end{aligned}$$

There are exactly two Markovian deterministic policies: δ^∞, which chooses action a_{11} in state s_1, and γ^∞, which instead chooses action a_{12} in that state. The policy evaluation equations for δ^∞ take the form

$$\begin{aligned} v(s_1) &= 5 + \lambda(0.5v(s_1) + 0.5v(s_2)) \\ v(s_2) &= -1 + \lambda v(s_2), \end{aligned}$$

which shows that

$$v_\lambda^{\delta^\infty}(s_1) = \frac{5 - 5.5\lambda}{(1 - 0.5\lambda)(1-\lambda)}, \quad v_\lambda^{\delta^\infty}(s_2) = -\frac{1}{1-\lambda}.$$

Similar calculations show that

$$v_\lambda^{\gamma^\infty}(s_1) = \frac{10 - 11\lambda}{(1-\lambda)}, \quad v_\lambda^{\gamma^\infty}(s_2) = -\frac{1}{1-\lambda}.$$

Comparing these, we see that $v_\lambda^{\gamma^\infty}(s_1) \geq v_\lambda^{\delta^\infty}(s_1)$ for all $\lambda \in [0,1)$.

The inverse operator $(I - \lambda P_d)^{-1}$ will play an important role throughout this chapter and so we summarize several of its properties in the following lemma. We will use the following notation: if $u, v \in V$, then we write $u \geq v$ if $u(s) \geq v(s)$ for all $s \in S$. In particular, $u \geq 0$ means that every value $u(s)$ is non-negative. We also write v^T for the vector transpose of v, i.e., v^T is a column vector.

Lemma 7.1. *For any $d \in D^{MR}$,*

(a) If $u \geq 0$, then $(I - \lambda P_d)^{-1} u \geq u$.

(b) If $u \geq v$, then $(I - \lambda P_d)^{-1} u \geq (I - \lambda P_d)^{-1} v$.

(c) If $u \geq 0$, then $u^T (I - \lambda P_d)^{-1} \geq u^T$.

For a proof, see Appendix C of Puterman (2005). Because of (a), we say that $(I - \lambda P_d)^{-1}$ is a **positive operator** and we write $(I - \lambda P_d)^{-1} \geq 0$.

7.3 Optimality Equations

If we let $v_n(s)$ denote the finite-horizon discounted value of an optimal policy, we know from our previous work that $v_n(s)$ satisfies the optimality equations:

$$v_n(s) = \sup_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v_{n+1}(j) \right\}.$$

This suggests that the infinite-horizon discounted rewards $v(s) = \lim_{n \to \infty} v_n(s)$ will satisfy the equations

$$v(s) = \sup_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v(j) \right\} \tag{7.7}$$

for every $s \in S$ and we call these equations the **optimality equations** or **Bellman equations** for the infinite-horizon process.

It will be convenient to define a non-linear operator $\mathcal{L}$ on V by setting

$$\mathcal{L} v \equiv \sup_{d \in D^{MD}} \left\{ r_d + \lambda P_d v \right\}, \tag{7.8}$$

where the supremum is evaluated component-wise, i.e., for each $s \in S$. When the sup is attained for all $v \in V$ we will also define the operator L on V by

$$L v \equiv \max_{d \in D^{MD}} \left\{ r_d + \lambda P_d v \right\}. \tag{7.9}$$

The next result explains why it is enough to consider only Markovian deterministic decision rules: in effect, we can always find a deterministic Markovian decision rule that performs as well as any randomized Markovian decision rule.

Proposition 7.1. *For all $v \in V$ and $0 \leq \lambda \leq 1$,*

$$\sup_{d \in D^{MD}} \left\{ r_d + \lambda P_d v \right\} = \sup_{d \in D^{MR}} \left\{ r_d + \lambda P_d v \right\}.$$

Proof. Since $D^{MD} \subset D^{MR}$, the RHS is at least as large as the LHS. Let $v \in V$, $\delta \in D^{MR}$ and observe that

$$\sup_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v(j) \right\} \geq \sum_{a \in A_s} q_\delta(a) \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v(j) \right\}.$$

Since this holds for every $s \in S$, it follows that for any $\delta \in D^{MR}$,

$$\sup_{d \in D^{MD}} \left\{ r_d + \lambda P_d v \right\} \geq r_\delta + \lambda P_\delta v,$$

and thus the LHS is at least as large as the RHS. □

In light of Proposition 7.1, we will write D for D^{MD}. Then, in the vector notation introduced in Section 7.1, the optimality equations (7.7) can be written as

$$v = \sup_{d \in D} \{r_d + \lambda P_d v\} = \mathcal{L}v. \tag{7.10}$$

If it is known that the supremum over D is achieved, then we can instead express the optimality equations as

$$v = \max_{d \in D} \{r_d + \lambda P_d v\} = Lv. \tag{7.11}$$

In either case, we see that the solutions of the optimality equations are **fixed points** of the operator $\mathcal{L}$ or L.

The following theorem provides lower and upper bounds on the discounted value v_λ^* in terms of sub-solutions and super-solutions of the operator $\mathcal{L}$. More importantly, it asserts that any fixed point of $\mathcal{L}$ (or, indeed, of L when that operator exists) is equal to v_λ^*.

Theorem 7.2. *Suppose that $v \in V$.*

(a.) If $v \geq \mathcal{L}v$, then $v \geq v_\lambda^*$.

(b.) If $v \leq \mathcal{L}v$, then $v \leq v_\lambda^*$.

(c.) If $v = \mathcal{L}v$, then $v = v_\lambda^*$.

Proof. It suffices to prove (a), since (b) follows by a similar argument, while (c) can be deduced from (a) and (b) in combination. Thus, suppose that $v \geq \mathcal{L}v$ and choose $\pi = (d_1, d_2, \cdots) \in \Pi^{MR}$. From Proposition 7.1, we know that

$$v \geq \mathcal{L}v = \sup_{d \in D^{MD}} \{r_d + \lambda P_d v\} = \sup_{d \in D^{MR}} \{r_d + \lambda P_d v\},$$

which (by Lemma 7.1) implies that

$$\begin{aligned} v &\geq r_{d_1} + \lambda P_{d_1} v \\ &\geq r_{d_1} + \lambda P_{d_1}(r_{d_2} + \lambda P_{d_2} v) \\ &= r_{d_1} + \lambda P_{d_1} r_{d_2} + \lambda^2 P_{d_1} P_{d_2} v. \end{aligned}$$

Continuing in this way, it follows that for every $n \geq 1$,

$$v \geq r_{d_1} + \lambda P_{d_1} r_{d_2} + \cdots + \lambda^{n-1} P_{d_1} \cdots P_{d_{n-1}} r_{d_n} + \lambda^n P_\pi^n v,$$

while

$$v_\lambda^\pi = r_{d_1} + \lambda P_{d_1} r_{d_2} + \cdots + \lambda^{n-1} P_{d_1} \cdots P_{d_{n-1}} r_{d_n} + \sum_{t=n}^{\infty} \lambda^t P_\pi^t r_{d_{t+1}},$$

and so

$$v - v_\lambda^\pi \geq \lambda^n P_\pi^n v - \sum_{t=n}^{\infty} \lambda^k P_\pi^t r_{d_{t+1}}.$$

Choose $\epsilon > 0$. Since $\lambda \in [0, 1)$, $||P_n|| \leq 1$ and $||r_d|| \leq M < \infty$ for all $d \in D^{MR}$, it follows that for all sufficiently large values of n we have

$$||\lambda^n P_\pi^n v|| \leq \frac{\epsilon}{2} \quad \text{and} \quad \left|\left|\sum_{k=n}^{\infty} \lambda^k P_\pi^k r_{d_{n+1}}\right|\right| \leq \frac{M\lambda^n}{1-\lambda} \leq \frac{\epsilon}{2}.$$

Taken together, these two inequalities imply that

$$v(s) \geq v_\lambda^\pi(s) - \epsilon$$

for every $s \in S$ and all $\epsilon > 0$, and since $\pi \in \Pi^{MR}$ is arbitrary, we then have

$$v(s) \geq \sup_{\pi \in \Pi^{MR}} v_\lambda^\pi(s) = v_\lambda^*(s)$$

for every $s \in S$. This establishes (a).

□

To establish the existence of a solution to the optimality equations, we will appeal to a powerful result in fixed-point theory known as the Banach fixed-point theorem. Suppose that U is a normed vector space. A sequence $(x_n : n \geq 1) \subset U$ is said to be a **Cauchy sequence** if for every $\epsilon > 0$ there is a positive integer $N = N(\epsilon)$ such that for all $n, m \geq N$

$$||x_n - x_m|| < \epsilon,$$

i.e., the sequence is Cauchy if for any $\epsilon > 0$ there is a ball of radius ϵ such that all but finitely many of the points in the sequence are contained in that ball. It can be shown, for instance, that any convergent sequence is a Cauchy sequence. If the converse is also true, i.e., if every Cauchy sequence in a normed vector space U converges to a point in U, then U is said to be a **Banach space**. This is relevant to discounted Markov decision problems because the space V of bounded real-valued functions, equipped with the supremum norm, is a Banach space.

We say that an operator $T : V \to V$ is a **contraction mapping** if there is a constant $\lambda \in [0, 1)$ such that

$$||Tv - Tu|| \leq \lambda ||v - u||$$

for all $u, v \in U$. In other words, the distance between the points Tu and Tv is shrunk by at least a factor λ relative to the distance between the points u and v. Notice that we do not require T to be a linear mapping.

Theorem 7.3. ***(Banach Fixed-Point Theorem)*** *Suppose that U is a Banach space and that $T : U \to U$ is a contraction mapping. Then*

(a.) T has a unique fixed point $v^* \in U$: $Tv^* = v^*$;

(b.) for any $v_0 \in U$, the sequence $(v_n : n \geq 0)$ defined by $v_n = T(v_{n-1}) = T_{n+1}v_0$ converges in norm to v^*.

Proof. Let $v_n = T^{n+1}v_0$. Then, for any $n, m \geq 1$,

$$\begin{aligned}
||v_{n+m} - v_n|| &\leq \sum_{k=0}^{m-1} ||v_{n+k+1} - v_{n+k}|| \\
&= \sum_{k=0}^{m-1} ||T^{n+k}v_1 - T^{n+k}v_0|| \\
&\leq \sum_{k=0}^{m-1} \lambda^{n+k} ||v_1 - v_0|| \\
&= \lambda^n \frac{1 - \lambda^m}{1 - \lambda} ||v_1 - v_0||,
\end{aligned}$$

which can be made arbitrarily small for all m by taking n sufficiently large. This shows that $(v_n : n \geq 0)$ is a Cauchy sequence and therefore it has a limit $v^* \in U$.

To see that v^* is a fixed point of T, observe that

$$\begin{aligned} 0 &\leq ||Tv^* - v^*|| \\ &= ||Tv^* - v_n|| + ||v_n - v^*|| \\ &= ||Tv^* - Tv_{n-1}|| + ||v_n - v^*|| \\ &\leq \lambda||v^* - v_{n-1}|| + ||v_n - v^*||. \end{aligned}$$

However, we know that $||v_n - v^*|| \to 0$ as $n \to \infty$ and so we can make $||Tv^* - v^*|| \geq 0$ arbitrarily small, which in turn means that it must be equal to 0, i.e., $Tv^* = v^*$ and so v^* is a fixed point of T as claimed.

□

Although the existence and uniqueness of the fixed point is the main content of the preceding theorem, the content of part (b) is also important because it shows that we can approximate the fixed point arbitrarily accurately by the iterates $T^n v_0$ of an arbitrary point v_0. Of course, the closer v_0 is to the fixed point and the smaller λ is, the more accurate this approximation will be. These facts are useful here because, as we will now show, the non-linear operators $\mathcal{L}$ and L are contractions on the Banach space V.

Proposition 7.2. *Let $\mathcal{L} : V \to V$ and $L : V \to V$ be defined by (7.8) and (7.9) for $\lambda \in [0,1)$. Then both operators are contraction mappings on V.*

Proof. We will show that L is a contraction and leave the corresponding proof for $\mathcal{L}$ to the reader. Given $u, v \in V$ and $s \in S$, we can assume without loss of generality that $Lv(s) \geq Lu(s)$ and let

$$a_s^* \in \arg\max_{a \in A_s} \left\{ r(s,a) + \sum_{j \in S} p(j|s,a)v(j) \right\}.$$

Then,

$$Lv(s) = r(s, a_s^*) - \lambda \sum_{j \in S} p(j|s, a_s^*)v(j)$$

and

$$Lu(s) \geq r(s, a_s^*) - \lambda \sum_{j \in S} p(j|s, a_s^*)u(j),$$

and so

$$\begin{aligned} 0 &\leq Lv(s) - Lu(s) \\ &\leq \left\{ r(s,a_s^*) + \lambda \sum_{j \in S} p(j|s,a_s^*)v(j) \right\} - \left\{ r(s,a_s^*) - \lambda \sum_{j \in S} p(j|s,a_s^*)u(j) \right\} \\ &= \lambda \sum_{j \in S} p(j|s,a_s^*)[v(j) - u(j)] \\ &\leq \lambda \sum_{j \in S} p(j|s,a_s^*)||v - u|| \\ &= \lambda||v - u||. \end{aligned}$$

This shows that

$$|Lv(s) - Lu(s)| \leq \lambda ||v - u||$$

for all s and taking the supremum over $s \in S$ then gives

$$||Lv - Lu|| \leq \lambda ||v - u||.$$

□

Theorem 7.4. *Suppose that S is countable, that the rewards are uniformly bounded, and that $\lambda \in [0, 1)$. Then*

(a) There exists a unique element $v^* \in V$ satisfying $Lv^* = v^*$ (or $\mathcal{L}v^* = v^*$) and $v^* = v^*_\lambda$ is the discounted value of the Markov decision problem.

(b) For each $d \in D^{MR}$, there exists a unique $v \in V$ satisfying $L_d v = v$ and $v = v_\lambda^{d^\infty}$ is the expected total discounted value of the stationary policy corresponding to d.

Proof. Since V is a Banach space and $\mathcal{L}$ and L are both contraction mappings on V, the Banach fixed-point theorem implies that each operator has a unique fixed point v^*. It then follows from Theorem 7.2 that $v^* = v^*_\lambda$. Part (b) follows from (a) by taking $D = \{d\}$ in the definitions of (7.10) and (7.11).

□

Although Theorem 7.4 tells us how (at least in principle) we may calculate the discounted value of a Markov decision problem, it does not tell us whether discount-optimal policies exist or indeed how to find them if they do exist. This issue is addressed by the next four theorems.

Theorem 7.5. *A policy $\pi^* \in \Pi^{HR}$ is discount optimal if and only if $v_\lambda^{\pi^*}$ is a solution of the optimality equation.*

Proof. If π^* is discount optimal, then $v_\lambda^{\pi^*} = v^*_\lambda$ and so Theorem 7.4 (a) tells us that $v_\lambda^{\pi^*}$ is the unique fixed point of $\mathcal{L}$, i.e., $v_\lambda^{\pi^*}$ is a solution of the optimality equation. Conversely, if $v_\lambda^{\pi^*}$ is a fixed point of $\mathcal{L}$, then Theorem 7.2 (c) tells us that $v_\lambda^{\pi^*} = v^*_\lambda$, which shows that π^* is discount-optimal.

□

This shows that we can assess whether a policy is discount optimal by checking whether the discounted value of that policy is a solution of the optimality equation.

Definition 7.1. *Given $v \in V$, a decision rule $d_v \in D^{MD}$ is said to be v-**improving** if*

$$r_{d_v} + \lambda P_{d_v} v = \max_{d \in D^{MD}} \{r_d + \lambda P_d v\}.$$

*In particular, a decision rule $d^* \in D^{MD}$ is said to be **conserving** for a λ-discounted Markov decision problem if d^* is v^*_λ-improving.*

The condition for d_v to be v-improving can also be written as

$$L_{d_v} v = Lv$$

or, component-wise, as

$$r(s, d_v(s)) + \lambda \sum_{j \in S} p(j|s, d_v(s))v(j) = \max_{a \in A_s} \left\{ r(s, a) + \lambda \sum_{j \in S} p(j|s, a)v(j) \right\}.$$

Similarly, d^* is conserving if and only if

$$L_{d^*}v^* = r_{d^*} + \lambda P_{d^*}v^* = v^*.$$

Conserving rules are particularly important because they give rise to optimal policies that are also stationary.

Theorem 7.6. *Suppose that the supremum is attained in the optimality equations for every $v \in V$. Then*

(a) there exists a conserving decision rule $d^* \in D^{MD}$;

(b) if d^* is conserving, the deterministic stationary policy $(d^*)^\infty$ is discount optimal;

(c) $v^*_\lambda = \sup_{d \in D} v^{d^*}_\lambda$.

Proof. Because we are assuming that the supremum is attained, we can define a decision rule d^* by choosing

$$d^*(s) \in \arg\max_{a \in A_s} \{r(s, a) + \lambda P(j|s, a)v^*_\lambda(j)\}$$

for each $s \in S$. It then follows that $L_{d^*}v^*_\lambda = v^*_\lambda$, which implies that d^* is conserving and also that

$$v^{d^*}_\lambda = v^*_\lambda.$$

This verifies both (a) and (b), while (c) follows from (b).

□

Theorem 7.7. *Suppose that there exists either a conserving decision rule or an optimal policy. Then there exists a deterministic stationary policy which is optimal.*

Proof. The sufficiency of the first claim follows from Theorem 7.6. Thus, suppose that $\pi^* = (d', \pi')$ is an optimal policy with $d_1 = d' \in D^{MR}$. Then, since $v^{\pi'}_\lambda \leq v^{\pi*}_\lambda$ and $P_{d'}$ is a positive operator,

$$\begin{aligned} v^*_\lambda &= r_{d'} + \lambda P_{d'} v^{\pi'}_\lambda \\ &\leq r_{d'} + \lambda P_{d'} v^{\pi^*}_\lambda \\ &\leq \sup_{d \in D} \left\{ r_d + \lambda P_d v^{\pi^*}_\lambda \right\} \\ &= v^*_\lambda. \end{aligned}$$

This implies that

$$r_{d'} + \lambda P_{d'} v^{\pi^*}_\lambda = \sup_{d \in D} \left\{ r_d + \lambda P_d v^{\pi^*}_\lambda \right\},$$

which shows that d' is a conserving decision rule and therefore $(d')^(\infty)$ is a stationary optimal policy by Theorem 7.6.

□

Theorem 7.8. *Assume that S is discrete and that one of the following conditions holds:*

(a) A_s is finite for every $s \in S$;

(b) A_s is compact for every $s \in S$, $r(s,a)$ is continuous in a and for all $j, s \in S$ $p(j|s,a)$ is continuous in a.

Then there exists an optimal deterministic stationary policy.

The proof is similar to that given for Proposition 5.1.

The following example demonstrates that some Markov decision processes have no discount optimal policies. Let $S = \{s\}$, $A_s = \{1, 2, 3, \cdots\}$ and $r(s,a) = 1 - 1/a$. Then

$$v_\lambda^*(s) = (1-\lambda)^{-1},$$

but no policy π exists with this value.

When optimal policies do not exist, we instead seek **ϵ-optimal** policies. A policy π_ϵ^* is said to be ϵ-optimal if for all $s \in S$

$$v_\lambda^{\pi_\epsilon}(s) \geq v_\lambda^*(s) - \epsilon,$$

or equivalently,

$$v_\lambda^{\pi_\epsilon^*} \geq v_\lambda^\pi - \epsilon e,$$

where $e = (1, 1, \cdots)$ is a vector of 1's.

Theorem 7.9. *If S is countable, then for every $\epsilon > 0$ there exists an ϵ-optimal decision rule.*

Proof. Given $\epsilon > 0$, choose a decision rule $d_\epsilon \in D^{MD}$ such that

$$r_{d_\epsilon} + \lambda P_{d_\epsilon} v_\lambda^* \geq v_\lambda^* - (1-\lambda)\epsilon e.$$

Since

$$\begin{aligned} v_\lambda^{d_\epsilon^{(\infty)}} &= (I - \lambda P_{d_\epsilon})^{-1} r_{d_\epsilon} \\ (I - \lambda P_{d_\epsilon})^{-1} e &= (1-\lambda)^{-1} e, \end{aligned}$$

it follows that

$$r_{d_\epsilon} \geq (I - \lambda P_{d_\epsilon})^{-1} v_\lambda^* - (1-\lambda)\epsilon e,$$

and

$$v_\lambda^{d_\epsilon^{(\infty)}} \geq v_\lambda^* - \epsilon e,$$

which shows that $d_\epsilon^{(\infty)}$ is ϵ-optimal. □

7.4 Value Iteration

The **value iteration algorithm** is a method that can be used to find ϵ-optimal policies for discounted Markov decision processes. The algorithm consists of the following steps:

(1) Set $n = 0$ and choose an error tolerance $\epsilon > 0$ and an initial condition $v^0 \in V$.

(2) For each $s \in S$, compute $v^{n+1}(s)$ by

$$v^{n+1}(s) = \max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v^{(n)}(j) \right\}$$

(3) If

$$||v^{n+1} - v^n|| < \frac{\epsilon(1-\lambda)}{2\lambda},$$

go to step 4. Otherwise increase n to $n+1$ and return to step 2.

(4) For each $s \in S$, choose

$$d_\epsilon(s) \in \arg\max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v^{n+1}(j) \right\}$$

and stop.

In vector notation, this algorithm can be expressed as:

$$\begin{array}{rcl} v^{n+1} & = & Lv^n \\ d_\epsilon & \in & \arg\max_{d \in D} \left\{ r_d + \lambda P_d v^{n+1} \right\}. \end{array}$$

The fact that value iteration leads to an ϵ-optimal policy is established in the proof of the next theorem.

Theorem 7.10. *Given v^0 and $\epsilon > 0$, let $(v^n : n \geq 0)$ be the sequence of values and d_ϵ the decision rule produced by the value iteration algorithm. Then*

(a) v^n converges in norm to v^*_λ;

(b) the stationary policy $d_\epsilon^{(\infty)}$ is ϵ-optimal;

(c) $||v^{n+1} - v^*_\lambda|| < \epsilon/2$ for any n that satisfies the inequality in step 3.

Proof. Convergence of the values v^n to the fixed point v^*_λ follows from the Banach fixed-point theorem. Choose n so that the condition

$$||v^{n+1} - v^n|| < \frac{\epsilon(1-\lambda)}{2\lambda},$$

is satisfied. Then

$$||v_\lambda^{d_\epsilon^{(\infty)}} - v^*_\lambda|| \leq ||v_\lambda^{d_\epsilon^{(\infty)}} - v^{n+1}|| + ||v^{n+1} - v^*_\lambda||.$$

Since $v_\lambda^{d_\epsilon^{(\infty)}}$ is a fixed point of L_{d_ϵ} and $L_{d_\epsilon} v^{n+1} = Lv^{n+1}$, it follows that

$$\begin{array}{rcl} ||v_\lambda^{d_\epsilon^{(\infty)}} - v^{n+1}|| & = & ||L_{d_\epsilon} v_\lambda^{d_\epsilon^{(\infty)}} - v^{n+1}|| \\ & \leq & ||L_{d_\epsilon} v_\lambda^{d_\epsilon^{(\infty)}} - L_{d_\epsilon} v^{n+1}|| + ||L_{d_\epsilon} v^{n+1} - v^{n+1}|| \\ & = & ||L_{d_\epsilon} v_\lambda^{d_\epsilon^{(\infty)}} - L_{d_\epsilon} v^{n+1}|| + ||Lv^{n+1} - Lv^n|| \\ & \leq & \lambda ||v_\lambda^{d_\epsilon^{(\infty)}} - v^{n+1}|| + \lambda ||v^{n+1} - v^n||. \end{array}$$

This shows that

$$||v_\lambda^{d_\epsilon^{(\infty)}} - v^{n+1}|| \leq \frac{\lambda}{1-\lambda}||v^{n+1} - v^n|| \leq \frac{1}{2}\epsilon.$$

Similarly,

$$\begin{aligned} ||v^{n+1} - v_\lambda^*|| &\leq \sum_{k=1}^{\infty} ||L^k v^n - L^k v^{n+1}|| \\ &\leq \sum_{k=0}^{\infty} \lambda^k ||v^n - v^{n+1}|| \\ &= \frac{\lambda}{1-\lambda}||v^n - v^{n+1}|| \\ &\leq \frac{1}{2}\epsilon, \end{aligned}$$

which establishes (c). Furthermore,

$$||v_\lambda^{d_\epsilon^{(\infty)}} - v_\lambda^*|| \leq \epsilon,$$

and this shows that $d_\epsilon^{(\infty)}$ is ϵ-optimal.

□

Theorem 7.11. *Given v^0 and $\epsilon > 0$, let $(v^n : n \geq 0)$ be the sequence of values produced by the value iteration algorithm. Then*

(a) convergence is linear at rate λ;

(b) the asymptotic average rate of convergence (AARC) is λ;

(c) for all $n \geq 1$,

$$||v^n - v_\lambda^*|| \leq \frac{\lambda^n}{1-\lambda}||v^1 - v^0||;$$

(d) for any $d_n \in \arg\max_{d \in D^{MR}}\{r_d + \lambda P_d v^n\}$,

$$||v_\lambda^{(d_n)^\infty} - v_\lambda^*|| \leq \frac{2\lambda^n}{1-\lambda}||v^1 - v^0||.$$

Proof. We first observe that for any $v^0 \in V$, the iterates of the algorithm satisfy

$$||v^{n+1} - v_\lambda^*|| = ||Lv^n - Lv_\lambda^*|| \leq \lambda||v^n - v_\lambda^*||.$$

This shows that convergence is at least linear with rate λ. Furthermore, if $v^0 = v_\lambda^* + c \cdot e$ for some scalar constant c, then

$$\begin{aligned} v^1 = Lv^0 &= \max_{d \in D} ||r_d + \lambda P_d(v_\lambda^* + c \cdot e)|| \\ &= \max_{d \in D} ||r_d + \lambda P_d v_\lambda^*|| + \lambda c \cdot e \\ &= v_\lambda^* + \lambda c \cdot e, \end{aligned}$$

which shows that $v^1 - v^0 = \lambda c \cdot e$ and, more generally, that $v^n - v^0 = \lambda^n c \cdot e$. Consequently, convergence of the value iteration algorithm is exactly linear with rate λ and the AARC is

$$AARC = \limsup_{n \to \infty} \left\{ \frac{||v^n - v_\lambda^*||}{||v^0 - v_\lambda^*||} \right\}^{1/n} = \lambda.$$

To verify (c), observe that

$$\begin{aligned} ||v^n - v^*_\lambda|| &\leq ||v^n - v^{n+1}|| + ||v^{n+1} - v^*_\lambda|| \\ &= ||L^n v^0 - L^n v^1|| + ||Lv^n - Lv^*_\lambda|| \\ &\leq \lambda^n ||v^1 - v^0|| + \lambda ||v^n - v^*_\lambda||. \end{aligned}$$

This can be rearranged to give

$$||v^n - v^*_\lambda|| \leq \frac{\lambda^n}{1-\lambda} ||v^1 - v^0||.$$

□

The number of iterations required to arrive at an ϵ-optimal policy can be estimated with the help of part (d) of Theorem 7.11 by setting

$$\epsilon = \frac{2\lambda^n}{1-\lambda} ||v^1 - v^0||$$

and then solving for n. This gives the formula

$$n \approx \frac{\ln\left(\frac{\epsilon(1-\lambda)}{2||v^1-v^0||}\right)}{\ln(\lambda)}$$

which, as expected, diverges as λ approaches 1. For example, if $\epsilon = 0.01$, $\lambda = 0.95$ and $||v^1 - v^0|| = 1$, then approximately $n = 162$ iterations will be required to find an ϵ-optimal policy.

Splitting Methods: The efficiency of the value iteration algorithm can sometimes be improved by using a technique from numerical linear algebra known as splitting. Suppose that the state space $S = \{s_1, \cdots, s_N$ is finite. To illustrate this approach, we will describe the **Gauss-Seidel value iteration algorithm**, which consists of the following steps:

(1) Set $n = 0$ and choose an error tolerance $\epsilon > 0$ and an initial condition $v^0 \in V$.

(2) For each value of $j = 1, \cdots, N$, compute $v^{n+1}(s_j)$ by

$$v^{n+1}(s_j) = \max_{a \in A_{s_j}} \left\{ r(s_j, a) + \lambda \left[\sum_{i<j} p(i|s,a) v^{(n+1)}(i) + \lambda \sum_{i \geq j} p(i|s,a) v^{(n)}(i) \right] \right\}.$$

(3) If

$$||v^{n+1} - v^n|| < \frac{\epsilon(1-\lambda)}{2\lambda},$$

go to step 4. Otherwise increase n to $n+1$ and return to step 2.

(4) For each $s \in S$, choose

$$d_\epsilon(s) \in \arg\max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v^{n+1}(j) \right\}$$

and stop.

The only difference between the ordinary value iteration algorithm and the variant based on the Gauss-Seidel method comes in step 2 where the latter algorithm replaces the current estimate $v^n(s_i)$ of the value of state s_i by the new estimate $v^{n+1}(s_i)$ as soon as it becomes available. It can be shown that this algorithm converges to the optimal value v_λ^*, that its order of convergence is linear, and that the rate of convergence is less than or equal to λ. Furthermore, for many problems, the rate of convergence is strictly less than λ, which means that the Gauss-Seidel algorithm is then strictly better than the ordinary value iteration.

7.5 Policy Iteration

Policy iteration works by constructing a sequence of policies with monotonically increasing rewards. Throughout this section we will assume that every vector $v \in V$ has an improving decision rule $d_v \in D^{MD}$, i.e.

$$d_v \in \arg\max_{d \in D^{MD}} \{r_d + \lambda P_d v\}$$

or equivalently

$$d_v(s) \in \arg\max_{a \in A_s} \left\{ r(s,a) + \lambda \sum_{j \in S} p(j|s,a) v(j) \right\}$$

for every $s \in S$. This will hold, for instance, if the action sets A_s are finite. The **policy iteration algorithm** consists of the following steps:

(1) Set $n = 0$ and choose an initial decision rule $d_0 \in D^{MD}$.

(2) **Policy evaluation:** Obtain $v^n = v_\lambda^{d_n^{(\infty)}}$ by solving the linear equation

$$(I - \lambda P_{d_n}) v^n = r_{d_n}.$$

(3) **Policy improvement:** Choose d_{n+1} so that

$$d_{n+1} \in \arg\max_{d \in D^{MD}} \{r_d + \lambda P_d v^n\},$$

setting $d_{n+1} = d_n$ whenever possible.

(4) If $d_{n+1} = d_n$, stop and set $d^* = d_n$. Otherwise, increase n by 1 and return to step 2.

An important property of the policy iteration algorithm is that the sequence of values v^n generated by the algorithm is non-decreasing. This is a consequence of the policy improvement step, which always selects a rule that is at least as good as the current rule.

Proposition 7.3. *Let v^n and v^{n+1} be successive values generated by the policy iteration algorithm. Then $v^{n+1} \geq v^n$.*

Proof. If d_{n+1} is a decision rule generated by the policy evaluation algorithm, then the policy improvement step and the fact that $L_{d_n} v^n = v^n$ imply that

$$r_{d_{n+1}} + \lambda P_{d_{n+1}} v^n \geq r_{d_n} + \lambda P_{d_n} v^n = v^n.$$

This shows that

$$r_{d_{n+1}} \geq (I - \lambda P_{d_{n+1}}) v^n$$

and consequently

$$v^{n+1} = (I - \lambda P_{d_{n+1}})^{-1} r_{d_{n+1}} \geq v^n.$$

□

In general, there is no guarantee that the policy evaluation algorithm will ever terminate, as the condition provided in step 4 might not ever be satisfied. The next theorem shows that this is not a concern whenever the state space and the action sets are all finite.

Theorem 7.12. *Suppose that S is finite and that all of the action sets A_s are finite. Then the policy iteration algorithm will terminate after finitely many iterations and the stationary policy $(d^*)^\infty = d_n^{(\infty)}$ will be discount optimal.*

Proof. By Proposition 7.3, the values v^n of successive stationary policies are non-decreasing. However, since there are finitely many deterministic stationary policies, the algorithm must terminate after finitely many iterations, since otherwise it would generate a sequence of infinitely many distinct reward vectors, contradicting the finiteness of the number of policies. At termination, $d_{n+1} = d_n$ and so

$$v^n = r_{d_n} + \lambda P_{d_n} v^n = \max_{d \in D^{MD}} \{r_d + \lambda P_d v^n\} = Lv^n.$$

This shows that v^n solves the optimal value equations and so $v^n = v_\lambda^*$. Also, since $v^n = v_\lambda^{d_n^{(\infty)}}$, it follows that $d_n^{(\infty)}$ is discount optimal.

□

Example: We again consider the two-state MDP from Section 4.1 and Section 7.2. Choose $\lambda = 0.95$ and $d_0(s_1) = a_{1,2}$ and $d_0(s_2) = a_{2,1}$. Evaluation of the policy $d_0^{(\infty)}$ leads to the system of equations

$$\begin{aligned} v(s_1) - 0.95v(s_2) &= 10 \\ 0.05v(s_2) &= -1, \end{aligned}$$

which can be solved to give $v^0(s_1) = -9$ and $v^0(s_2) = -20$. The policy improvement step requires us to evaluate

$$\max\{5 + 0.475v^0(s_1) + 0.475v^0(s^2), 10 + 0.95v^0(s_2)\} = \max\{-8.775, -9\}$$

so that $d_1(s_1) = a_{1,1}$ and $d_1(s_2) = a_{2,1}$. A second round of policy evaluation gives

$$\begin{aligned} .525v(s_1) - 0.475v(s_2) &= 5 \\ 0.05v(s_2) &= -1, \end{aligned}$$

which yields $v^1(s_1) = -8.571$ and $v^1(s_2) = -20$. This time the policy improvement step shows that $d_2 = d_1$ and so we set $d^* = d_1$, which is then the optimal policy.

Unfortunately, the conclusions of Theorem 7.12 might not hold when either the state space or the action sets are infinite. To analyze the performance of the algorithm in these more general settings, it will be helpful to be able to represent the algorithm in terms of the recursive application of an operator on the vector space V. This will also allow us to compare the performance of the policy iteration algorithm with the value iteration algorithm, which will be a key ingredient of the convergence proof for the former. To this end we will introduce a new operator $B : V \to V$, which is defined by

$$Bv \equiv Lv - v = \max_{d \in D^{MD}} \{r_d + (\lambda P_d - I)v_d\},$$

and we observe that the optimality equation can then be written as

$$Bv = 0,$$

i.e., the value of the discounted Markov decision problem is the unique root of the operator B. Below, we will show that the policy evaluation algorithm can be interpreted as a generalized form of Newton's method applied to this root-finding problem. We begin with a proposition that shows that B satisfies a generalized notion of convexity. To this end, given $v \in V$, let D_v be the collection of v-improving decision rules, i.e., $d_v \in D_v$ if and only if

$$d_v \in \arg \max_{d \in D^{MD}} \{r_d + \lambda P_d v\} = \arg \max_{d \in D^{MD}} \{r_d + \lambda (P_d - I)v\}.$$

We will refer to D_v as the set of **supporting decision rules** at v and we will say that $\lambda P_{d_v} - I$ is the support of B at v.

Proposition 7.4. ***(Support inequality)*** *For $u, v \in V$ and $d_v \in D_v$, we have*

$$Bu \geq Bv + (\lambda P_{d_v} - I)(u - v).$$

Proof. The result in a consequence of the following identities/inequalities:

$$\begin{aligned} Bu &\geq r_{d_v} + (\lambda P_{d_v} - I)u \\ Bv &= r_{d_v} + (\lambda P_{d_v} - I)v. \end{aligned}$$

□

The next theorem shows how the operator B can be used to give a recursive representation of the policy iteration algorithm. As explained below, this representation can also be interpreted as an application of Newton's method to B.

Theorem 7.13. *Suppose that the sequence $(v^n; n \geq 0)$ is generated by the policy iteration algorithm. Then, for any supporting rule $d_{v^n} \in D_{v_n}$, we have*

$$v^{n+1} = v^n - (\lambda P_{d_{v^n}} - I)^{-1} B v^n$$

Proof. By the definition of D_{v^n}, we have

$$\begin{aligned} v^{n+1} &= (I - \lambda P_{d_{v^n}})^{-1} r_{d_{v^n}} - v^n + v^n \\ &= (I - \lambda P_{d_{v^n}})^{-1} \left[r_{d_{v^n}} + (\lambda P_{d_{v^n}} - I)\, v^n \right] + v^n \\ &= v^n - (I - \lambda P_{d_{v^n}})^{-1} B v^n. \end{aligned}$$

□

To understand the connection with Newton's method, recall that if f is a continuously differentiable function on $\mathbb{R}$, then Newton's method generates the sequence $(x_n; n \geq 0)$ using the recursion

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)};$$

geometrically, this amounts to following the line tangent to the graph of f at x_n down to its intersection with the x-axis, which is then x_{n+1}. In Theorem 7.13, the support $\lambda P_{d_{v^n}} - I$ plays the role of the derivative (or Jacobean) of the function B.

Our next results are concerned with the convergence of the policy iteration algorithm. Define $V_B \equiv \{v \in V : Bv \geq 0\}$. It follows from Theorem 7.2 that if $v \in V_B$, then $Lv \geq v$ and so $v \leq v^*_\lambda$, i.e., v is a lower bound for v^*_λ.

Lemma 7.2. *Let $v \in V_B$ and $d_v \in D_v$. Then*

(a) $Zv \equiv v + (I - \lambda P_{d_v})^{-1} Bv \geq Lv$;

(b) $Zv \in V_B$;

(c) $Zv \geq v$.

Proof. Since $Bv \geq 0$, Lemma 7.1 tells us that $(I - \lambda P_{d_v})^{-1} Bv \geq 0$ and so

$$Zv = v + (I - \lambda P_{d_v})^{-1} Bv \geq v + Bv = Lv.$$

Furthermore, by the support inequality for B (Proposition 7.4), we have

$$B(Zv) \geq Bv + (\lambda P_{d_v} - I)(Zv - v) = Bv - Bv = 0.$$

Part (c) follows from the assumption that $Bv \geq 0$.

□

Theorem 7.14. *The sequence of values $(v^n; n \geq 0)$ generated by the policy iteration algorithm converges monotonically and in norm to v^*_λ.*

Proof. Let $u^n = L^n v^0$ be the sequence of values generated by the value iteration algorithm. We will use induction to show that $u^n \leq v^n \leq v^*_\lambda$ and $v^n \in V_B$ for all $n \geq 0$. First observe that

$$Bv^0 = \max_{d \in D^{MD}} \left\{ r_d + (\lambda P_d - I) v^0 \right\} \geq r_{d_0} + (\lambda P_{d_0} - I) v^0 = 0,$$

so that $v^0 \in V_B$ and consequently $v^0 \leq v^*_\lambda$. Since $u^0 = v^0$, this verifies the induction hypothesis when $n = 0$.

Now assume that the hypothesis holds for all $0 \leq k \leq n$ for some n. Since $v^{n+1} = Zv^n$, Lemma 7.2 implies that $v^{n+1} \in V_B$, $v^n \leq v^*_\lambda$, and $v^{n+1} \geq Lv^n \geq Lu^n = u^{n+1}$. This completes the induction.

The theorem conclusion then follows from the fact that the sequence $(u^n; n \geq 0)$ converges to v^*_λ in norm. □

Since value iteration has linear convergence, it follows that policy iteration has at least this order of convergence. In fact, our next result shows that the order of convergence can be quadratic under some conditions.

Theorem 7.15. *Suppose that the sequence $(v^n; n \geq 0)$ is generated by the policy iteration algorithm, that $d_{v^n} \in D_{v^n}$ is a supporting decision rule for each $n \geq 0$, and that there exists a finite positive number $K \in (0, \infty)$ such that*

$$||P_{d_{v^n}} - P_{d_{v^*_\lambda}}|| \leq K ||v^n - v^*_\lambda||.$$

Then

$$||v^{n+1} - v^*_\lambda|| \leq \frac{K\lambda}{1 - \lambda} ||v^n - v^*_\lambda||^2.$$

Proof. If we define the operators

$$U_n \equiv (\lambda P_{d_{v^n}} - I) \text{ and } U_* \equiv (\lambda P_{d_{v^*_\lambda}} - I),$$

then the support inequality implies that

$$Bv^n \geq Bv^*_\lambda + U_*(v^n - v^*_\lambda) = U_*(v^n - v^*_\lambda)$$

which along with the positivity of $-U_n^{-1}$ gives

$$U_n^{-1}Bv^n \leq U_n^{-1}U_*(v^n - v^*_\lambda).$$

Since the values v^n generated by the policy iteration algorithm increase monotonically to v^*_λ, it follows that

$$\begin{aligned} 0 &\leq v^*_\lambda - v^{n+1} \\ &= v^*_\lambda - v^n + U_n^{-1}Bv^n \\ &\leq U_n^{-1}U_n(v^*_\lambda - v^n) - U_n^{-1}U_*(v^*_\lambda - v^n). \end{aligned}$$

This implies that

$$||v^*_\lambda - v^{n+1}|| \leq ||U_n^{-1}||\, ||U_n - U_*||\, ||v^*_\lambda - v^n||.$$

However, since

$$||U_n^{-1}|| \leq \frac{1}{1-\lambda}$$

and

$$||U_n - U^*_\lambda|| = \lambda ||P_{d_{v^n}} - P_{d_{v^*_\lambda}}||,$$

the result then follows from the assumption that $||P_{d_{v^n}} - P_{d_{v^*_\lambda}}|| \leq K||v^n - v^*_\lambda||$. □

Corollary 7.1. *Suppose that there exists a positive constant $K \in (0, \infty)$ such that*

$$||P_{d_v} - P_{d_u}|| \leq K||v - u||$$

holds for all $u, v \in V$ whenever $d_u \in D_u$ and $d_v \in D_v$. Then the conditions of Theorem 7.15 are satisfied and so the policy iteration algorithm converges quadratically.

Sufficient conditions for this inequality to hold are that for each $s \in S$,

(i) A_s is compact and convex,

(ii) $p(j|s, a)$ is affine in a, and

(iii) $r(s, a)$ is strictly concave and twice continuously differentiable in a.

7.6 Modified Policy Iteration

Although the policy iteration algorithm has quadratic convergence, the actual numerical implementation of this algorithm may be computationally expensive due to the need to solve a linear equation,

$$(I - \lambda P_{d_n})v = r_{d_n},$$

during the evaluation step of each iteration. Fortunately, in many problems, it suffices to find an approximate solution to this equation, which can be done much more efficiently. This is the motivation for the **modified policy iteration algorithm**, which combines features of both the value iteration algorithm and the policy iteration algorithm. Suppose that $\{m_n; n \geq 0\}$ is a sequence of non-negative integers. Then the modified policy iteration algorithm consists of the following steps:

(1) Set $n = 0$ and select an error threshold $\epsilon > 0$ and an initial vector $v^0 \in V$.

(2) **Policy improvement:** Choose $d_{n+1} \in D_{v^n} = \arg\max\{r_d + \lambda P_d v^n\}$, setting $d_{n+1} = d_n$ whenever possible.

(3) **Partial policy evaluation:**

(a) Set $k = 0$ and

$$u_n^0 \equiv \max_d \{r_d + \lambda P_d v^n\}.$$

(b) If $||u_n^0 - v^n|| < \epsilon(1-\lambda)/2\lambda$, go to step 4. Otherwise go to step 3 (c).

(c) If $k = m_n$, go to (e). Otherwise, compute u_n^{k+1} by

$$u_n^{k+1} = r_{d_{n+1}} + \lambda P_{d_{n+1}} u_n^k = L_{d_{n+1}} u_n^k.$$

(d) Increase k to $k+1$ and return to (c).

(e) Set $v^{n+1} = u_n^{m_n}$, increase n to $n+1$, and go to step 2.

(4) Set $d_\epsilon = d_{n+1}$ and stop.

Notice that the modified policy iteration algorithm includes both a policy improvement and an evaluation step, but that policy evaluation is only done approximately, by iterating the operator $L_{d_{n+1}}$ m_n times:

$$v^{n+1} = L_{d_{n+1}}^{m_n+1} v^n.$$

Of course, $L_{d_{n+1}}$ is a contraction mapping for each n and so the sequence $u_n^0, u_n^1, \cdots$ converges linearly to the unique fixed point of this operator, which is $v_\lambda^{d_{n+1}}$. However, in general, we only have $u_\lambda^{m_n} \approx v_\lambda^{d_{n+1}}$ with larger values of m_n giving more accurate approximations. The sequence $\{m_n; n \geq 0\}$ is called the **order sequence** and determines the rate of convergence of modified policy iteration as well as the computational requirements per step.

Proposition 7.5. *Suppose that $\{v^n; n \geq 0\}$ is a sequence generated by the modified policy iteration algorithm. Then*

$$v^{n+1} = v^n + \sum_{k=0}^{m_n} \left(\lambda P_{d_{n+1}}\right)^k B v^n.$$

Proof. Since

$$L_{d_{n+1}} v = r_{d_{n+1}} + \lambda P_{d_{n+1}} v,$$

it follows that

$$\begin{aligned} v^{n+1} &= L_{d_{n+1}}^{m_n+1} v^n \\ &= r_{d_{n+1}} + \lambda P_{d_{n+1}} r_{d_{n+1}} + \cdots + (\lambda P_{d_{n+1}})^{m_n} r_{d_{n+1}} + (\lambda P_{d_{n+1}})^{m_n+1} v^n \\ &= v^n + \sum_{k=0}^{m_n} (\lambda P_{d_{n+1}})^k \left[r_{d_{n+1}} + \lambda P_{d_{n+1}} v^n - v^n\right] \\ &= v^n + \sum_{k=0}^{m_n} (\lambda P_{d_{n+1}})^k B v^n. \end{aligned}$$

□

Notice that if $m_n = \infty$, then

$$\begin{aligned} v^{n+1} &= v^n + \sum_{k=0}^{\infty} \left(\lambda P_{d_{n+1}}\right)^k Bv^n \\ &= v^n + (I - \lambda P_{d_{n+1}})^{-1} Bv^n, \end{aligned}$$

in which case the sequence $(v^n; n \geq 0)$ satisfies the recursion generated by the policy iteration algorithm. This shows that the modified policy iteration algorithm arises by truncating the Neumann expansion that represents the exact solution needed for the policy evaluation step. Similarly, if $m_n = 0$, then

$$v^{n+1} = v^n + Bv^n = Lv^n,$$

and the modified policy iteration algorithm reduces to the value iteration algorithm.

Our next task is to show that a sequence generated by the modified policy algorithm converges to the optimal value of the discounted Markov decision problem. To this end, for each $m \geq 0$ we will define operators $U^m : V \to V$ and $W^m : V \to W$ by

$$\begin{aligned} U^m v &\equiv \max_d \left\{ \sum_{k=0}^{m} (\lambda P_d)^k r_d + (\lambda P_d)^{m+1} v \right\} \\ W^m v &\equiv v + \sum_{k=0}^{m} (\lambda P_{d_v})^k Bv, \quad d_v \in D_v. \end{aligned}$$

From Proposition 7.5, we know that $v^{n+1} = W^{m_n} v^n$. Furthermore, since

$$Bv = Lv - v = r_{d_v} + \lambda P_{d_v} v - v$$

for any v-improving decision rule d_v, it follows that

$$\begin{aligned} W^m v &= v + \sum_{k=0}^{m} (\lambda P_{d_v})^k r_{d_v} + (\lambda P_d)^{m+1} v - v \\ &= \sum_{k=0}^{m} (\lambda P_{d_v})^k r_{d_v} + (\lambda P_d)^{m+1} v. \end{aligned}$$

The next three lemmas set forth some useful properties of these operators.

Lemma 7.3. *For any $w^0 \in V$,*

(a) U^m is a contraction operator with constant λ^{m+1};

(b) the sequence $w^{n+1} = U^m w^n, n \geq 0$ converges in norm to v^*_λ;

(c) v^*_λ is the unique fixed point of U_m;

(d) $||w^{n+1} - v^*_\lambda|| \leq \lambda^{m+1} ||w^n - v^*_\lambda||$.

Proof. To prove that U^m is a contraction on V, let $u, v \in V$, fix s ∈S, and suppose that $U^m v(s) \geq U^m u(s)$. Then, for any

$$d^* \in \arg\max_{d \in D^{MD}} \left\{ \sum_{k=0}^{m} (\lambda P_d)^k r_d + (\lambda P_d)^{m+1} v \right\},$$

we have

$$\begin{aligned} 0 \ \leq \ & U_m v(s) - U_m u(s) \leq \left(\sum_{k=0}^{m} (\lambda P_{d^*})^k r_{d^*} + (\lambda P_{d^*})^{m+1} v\right)(s) \\ & \qquad - \left(\sum_{k=0}^{m} (\lambda P_{d^*})^k r_{d^*} + (\lambda P_{d^*})^{m+1} u\right)(s) \\ = \ & \lambda^{m+1} P_d^{m+1}(v-u) \ \leq \lambda^{m+1} ||v-u||. \end{aligned}$$

A similar argument applies if $U^m v(s) \leq U^m u(s)$ and this establishes (a). It then follows from the contraction mapping theorem that U^m has a unique fixed point, say w^*, and that any sequence of iterates of U^m converges in norm to w^*. To show that $w^* = v^*_\lambda$, let d^* be a v^*_λ-improving decision rule. Then

$$\begin{aligned} v^*_\lambda \ = \ L^m v^*_\lambda \ & = \ \sum_{k=0}^{m} (\lambda P_{d^*})^k r_{d^*} + (\lambda P_{d^*})^{m+1} v^*_\lambda \\ \leq \ U^m v^*_\lambda. & \end{aligned}$$

Furthermore, since U^m is order-preserving, it follows that $v^*_\lambda \leq (U^m)^n v^*_\lambda$ for all $n \geq 1$ and taking the limit $n \to \infty$ shows that $v^*_\lambda \leq w^*$. Similarly, $w^* = U^m w^* \leq L^m w^*$ and iteration of this expression shows that $w^* \leq (L^m)^n w^*$ for all $n \geq 1$ and consequently $w^* \leq v^*_\lambda$. Collectively, these two identities show that $w^* = v^*_\lambda$, as claimed by the lemma. □

Lemma 7.4. *If $u, v \in V$ with $u \geq v$, then $U^m u \geq W^m v$. Furthermore, if $u \in V_B$, then $W^m u \geq U^0 v = Lv$.*

Proof. If d_v is v-improving, then in light of the alternate expression for $W^m v$ given above, we have

$$\begin{aligned} U^m u - W^m v \ & \geq \ \sum_{k=0}^{m} (\lambda P_{d_v})^k r_{d_v} + (\lambda P_{d_v})^{m+1} u - \sum_{k=0}^{m} (\lambda P_{d_v})^k r_{d_v} - (\lambda P_{d_v})^{m+1} v \\ & \geq \ (\lambda P_{d_v})^{m+1}(u-v) \geq 0. \end{aligned}$$

Now suppose that $u \in V_B$ and that d_u is u-improving. Then

$$\begin{aligned} W^m u \ & = \ u + \sum_{k=0}^{m} (\lambda P_{d_u})^k Bu \\ & \geq \ u + Bu \ = \ Lu \\ & \geq \ Lv. \end{aligned}$$

□

Lemma 7.5. *If $u \in V_B$, then $W^m u \in V_B$.*

Proof. If $w = W^m u$, then the support inequality implies that

$$\begin{aligned} Bw \ & \geq \ Bu + (\lambda P_{d_u} - I)(w-u) \\ & = \ Bu + (\lambda P_{d_u} - I) \sum_{k=0}^{m} (\lambda P_{d_u})^k Bu \\ & = \ (\lambda P_{d_v})^{m+1} Bu \ \geq \ 0. \end{aligned}$$

□

Finally, we arrive at the main result.

Theorem 7.16. *Suppose that $v^0 \in V_B$. Then, for any order sequence $\{m_n; n \geq 0\}$,*

(i) the iterates $\{v^n; n \geq 0\}$ of the modified policy iteration algorithm converge monotonically and in norm to v^*_λ, and

(ii) the algorithm terminates in finitely many iterations with an ϵ-optimal policy.

Proof. Define the sequences $\{y^n\}$ and $\{w^n\}$ by setting $y^0 = w^0 = v^0$, $y^{n+1} = Ly^n$, and $w^{n+1} = U^{m_n}w^n$. We will show by induction on n that $v^n \in V_B$, $v^{n+1} \geq v^n$, and $w^n \geq v^n \geq y^n$.

By assumption, these claims are satisfied when $n = 0$. Suppose then that they also hold for $k = 1, \cdots, n$. In particular, since $v^n \in V_B$ and $v^{n+1} = W^{m_n}v^n$ (by Proposition 7.5), Lemma 7.5 shows that $v^{n+1} \in V_B$. Appealing again to Proposition 7.5, we see that

$$v^{n+1} = v^n + \sum_{k=0}^{m_n} \left(\lambda P_{d_{n+1}}\right)^k Bv^n \geq v^n,$$

which establishes that the sequence $\{v^n\}$ is increasing. Since $w^n \geq v^n \geq y^n$ and $v^n \in V_B$, Lemma 7.4 implies that

$$\begin{aligned} w^{n+1} &= U^{m_n}w^n \\ &\geq W^{m_n}v^n = v^{n+1} \\ &\geq Ly^n = y^{n+1}. \end{aligned}$$

This verifies that the induction hypothesis holds for $n+1$ and thus completes the induction. Then, since the sequences w^n and y^n both converge to v^*_λ in norm as $n \to \infty$, it follows that v^n also converges to this limit in norm. Finally, by Theorem 7.10, we know that the value iteration algorithm terminates after finitely many steps with an ϵ-optimal policy and the above shows that the same must be true of the modified policy iteration algorithm.

□

The previous theorem implies that modified policy iteration converges at least as rapidly as value iteration (since the iterates of the former are bounded between the iterates of the latter and the optimal value of the Markov decision problem). Thus, the modified policy iteration algorithm converges at least linearly. The following theorem provides a more precise statement of this result.

Theorem 7.17. *Suppose that $v^0 \in B_V$ and let $\{v^n; n \geq 0\}$ be a sequence generated by the modified policy iteration algorithm. If d_n is a v^n-improving decision rule and d^* is a v^*_λ-improving decision rule, then*

$$||v^{n+1} - v^*_\lambda|| \leq \left(\frac{\lambda\left(1 - \lambda^{m_n+1}\right)}{1-\lambda}||P_{d_n} - P_{d^*}|| + \lambda^{m_n+1}\right)||v^n - v^*_\lambda||.$$

Proof. In light of Theorem 7.16 and the support inequality

$$Bv^n \geq Bv^*_\lambda + (\lambda P_{d^*} - I)(v^n - v^*_\lambda) = (\lambda P_{d^*} - I)(v^n - v^*_\lambda),$$

we have

$$\begin{aligned}
0 &\leq v_\lambda^* - v^{n+1} \\
&= v_\lambda^* - v^n - \sum_{k=0}^{m_n} (\lambda P_{d_n})^k \, B v^n \\
&\leq v_\lambda^* - v^n + \sum_{k=0}^{m_n} (\lambda P_{d_n})^k \, (I - \lambda P_{d^*})(v^n - v_\lambda^*) \\
&= v_\lambda^* - v^n + \sum_{k=0}^{m_n} (\lambda P_{d_n})^k \, (I - \lambda P_{d_n} + \lambda P_{d_n} - \lambda P_{d^*})(v^n - v_\lambda^*) \\
&= \lambda \sum_{k=0}^{m_n} (\lambda P_{d_n})^k \, (P_{d_n} - P_{d^*}) \, (v^n - v_\lambda^*) - \lambda^{m_n+1} P_{d_n}^{m_n+1} (v^n - v_\lambda^*).
\end{aligned}$$

The result follows upon taking norms in the preceding expressions.

□

Corollary 7.2. *Suppose that the hypotheses of Theorem 7.16 are satisfied and that*

$$\lim_{n\to\infty} ||P_{d_n} - P_{d^*}|| = 0.$$

Then, for any $\epsilon > 0$, there is an N such that

$$||v^{n+1} - v_\lambda^*|| \leq \left(\lambda^{m_n+1} + \epsilon\right) ||v^n - v_\lambda^*||,$$

for all $n \geq N$.

This shows that as long as the transition matrices converge in norm, we can make the rate constant arbitrarily close to 0 by taking m_n sufficiently large. Of course, a trade-off is encountered in the selection of the order sequence. Larger values of m_n accelerate convergence but also require more work per iteration.