Dynamic Programming

Volume I: Foundations

QuantEcon Book II

Thomas J. Sargent and John Stachurski

March 15, 2023

Contents

Preface **vii**

Common Symbols **x**

Common Abbreviations **xi**

1 Introduction **1**

1.1 Bellman Equations 3

1.1.1 Finite-Horizon Job Search 3

1.1.2 Infinite Horizon 10

1.2 Stability and Contractions 12

1.2.1 Vector Space 12

1.2.2 Nonlinear Systems 18

1.2.3 Successive Approximation 23

1.2.4 Finite-Dimensional Function Space 28

1.3 Infinite-Horizon Job Search 32

1.3.1 Values and Policies 32

1.3.2 Computation 35

1.4 Chapter Notes 40

2 Operators and Fixed Points **42**

2.1 Order 42

2.1.1 Partial Orders 42

2.1.2 Order-Preserving Maps 48

2.1.3 Parametric Monotonicity 50

2.2 Stability and Algorithms 54

2.2.1 Conjugate Maps 54

2.2.2 Local Stability 55

2.2.3 Convergence Rates 57

2.2.4 Gradient-Based Methods 58

2.3 Matrices and Operators 61

2.3.1 Nonnegative Matrices 62

2.3.2 A Lake Model 64

2.3.3 Linear Operators 68

2.4 Chapter Notes 72

3 Markov Dynamics **73**

3.1 Foundations 73

3.1.1 Markov Chains 73

3.1.2 Stationarity and Ergodicity 78

3.1.3 Approximation 82

3.2 Conditional Expectations 84

3.2.1 Mathematical Expectations 84

3.2.2 Monotonicity 86

3.2.3 Geometric Sums 90

3.3 Job Search Revisited 94

3.3.1 Job Search with Markov State 94

3.3.2 Job Search with Separation 98

3.4 Chapter Notes 101

4 Optimal Stopping **102**

4.1 Introduction to Optimal Stopping 102

4.1.1 Theory 102

4.1.2 Firm Valuation with Exit 108

4.1.3 Monotonicity 111

4.1.4 Continuation Values 113

4.2 Further Applications 117

4.2.1 American Options 117

4.2.2 Research and Development 120

4.3 Chapter Notes 124

5 Markov Decision Processes **125**

5.1 Definition and Properties 125

5.1.1 The MDP Model 125

5.1.2 Examples 127

5.1.3 Optimality 131

5.1.4 Algorithms 136

5.2 Applications 143

5.2.1 Optimal Inventories 143

5.2.2 Optimal Savings with Labor Income 146

5.2.3 Optimal Investment 153

5.3 Modified Bellman Equations 159

5.3.1 Structural Estimation 159

5.3.2 The Gumbel Max Trick 164

5.3.3 Optimal Savings with Stochastic Returns on Wealth 165

5.3.4 Q-Factors 168

5.3.5 Operator Factorizations 169

5.4 Chapter Notes 176

6 Stochastic Discounting **178**

6.1 Time-Varying Discount Factors 178

6.1.1 Valuation 178

6.1.2 Necessary Conditions 182

6.2 Asset Pricing 183

6.2.1 Introduction to Asset Pricing 183

6.2.2 Nonstationary Dividends 189

6.2.3 Incomplete Markets 191

6.3 Optimality with State-Dependent Discounting 193

6.3.1 Eventual Contractions 193

6.3.2 MDPs with State-Dependent Discounting 195

6.3.3 Inventory Management Revisited 199

6.4 Chapter Notes 202

7 Nonlinear Valuation **204**

7.1 Beyond Contraction Maps 204

7.1.1 Order Intervals and Fixed Points 205

7.1.2 Concavity, Convexity and Stability 206

7.1.3 A Power-Transformed Affine Equation 210

7.2 Recursive Preferences 211

7.2.1 Motivation: Optimal Savings 211

7.2.2 Risk-Sensitive Preferences 214

7.2.3 Epstein–Zin Preferences 218

7.2.4 A General Representation 223

7.2.5 Stability 229

7.3 Chapter Notes 232

8 Abstract Dynamic Programming **234**
8.1 Recursive Decision Processes . . . 235
8.1.1 Defining RDPs . . . 235
8.1.2 Lifetime Value . . . 239
8.1.3 Optimality . . . 242
8.2 Types of RDPs . . . 246
8.2.1 Contracting RDPs . . . 246
8.2.2 Risk-Sensitive RDPs . . . 254
8.2.3 Eventually Contracting RDPs . . . 255
8.2.4 Convex and Concave RDPs . . . 259
8.2.5 Application: Adversarial Agents . . . 261
8.3 Abstract Theory . . . 262
8.3.1 Abstract Dynamic Programs . . . 262
8.3.2 Convergence of OPI . . . 270
8.4 Isomorphic Dynamic Programs . . . 272
8.4.1 Isomorphic ADPs . . . 273
8.4.2 Minimization Problems . . . 278
8.4.3 Application: Negative Discount Rate Optimality . . . 279
8.4.4 Epstein–Zin Revisited . . . 281
8.5 Chapter Notes . . . 283

I Appendices **285**

A Suprema and Infima **286**
A.1 Sets and Functions . . . 286
A.2 Sup, Inf, Max and Min . . . 287
A.3 Lattices . . . 289
A.3.1 Suprema and Infima in Partially Ordered Sets . . . 289
A.3.2 Suprema and the Pointwise Order . . . 290
A.3.3 Existence and Uniqueness . . . 290
A.3.4 Lattices . . . 291

B Remaining Proofs **292**

C Solutions to Selected Exercises **294**

Preface

This book is about dynamic programming and its applications in economics, finance, and adjacent fields like operations research. It brings together recent innovations in the theory of dynamic programming and also provides applications and code that will help a reader approach the research frontier. The book is suitable graduate students in economics and finance, as well as researchers in these fields. Most chapters are also accessible to undergraduates with good quantitative backgrounds.

While the book contains classical results on dynamic programming that are found in texts such as Bellman (1957), Denardo (1981), Puterman (2005), and Stokey and Lucas (1989), it also describes extensions created by researchers and practitioners during the last few decades as they wrestled with how to formulate and solve decision problems. These include recursive preference models, robust control, and models with time varying-discount rates. Such settings often fail to satisfy contraction-mapping restrictions upon which traditional methods are based. To accommodate these applications, the key theoretical chapter of this book (Chapter 8) builds on an abstract framework of Bertsekas (2022b). This approach provides extra generality while also offering transparent proofs. A number of results are new.

(A more detailed comparison between traditional approaches to optimality and the results provided in this book can be found in §8.1.3.4.)

Chapters 1–7 provide motivation and background material on solving fixed point problems and computing lifetime valuations. Chapters 4 and 5 cover optimal stopping and Markov decision processes respectively. The main theoretical results in on dynamic programming from these chapters are special cases of the general results found in Chapter 8.

Mathematically inclined readers might prefer to start with the general results in Chapter 8. Indeed, it is possible to read the text in reverse, jumping to Chapter 8 and then moving back to cover special cases according to ones interests. However, our teaching experience tells us that most students find the general results challenging on first pass, but considerably easier after they have practiced dynamic programming

through the earlier chapters. This is why the presentation starts with special cases and ends with general results.

Instructors wishing to use this book as a text for undergraduate students can start with Chapter 1, skim through Chapter 2, work through Chapters 3–5, and skip Chapter 8 entirely. For such a course the topic of convex and concave operators in Chapter 2 can be avoided, as can the discussion on topological conjugacy, along with the treatment of recursive preferences in §7.2.

This book deals only with finite state and action spaces, leaving more general settings to Volume II. In fact the theory for general state models is essentially the same as that for finite state models, with almost all differences involving qualifications and conditions to ensure that maximization and integration make sense. A good way to learn dynamic programming is first to cover the finite case in which these qualifications are not necessary. Without these distractions, the core theory of dynamic programming is especially clear.[1]

Another feature of the text is that computer code is a first-class citizen. The code embedded in the textbook is written in Julia and can be found at

`https://github.com/QuantEcon/book-dp1`

We chose Julia because it is open source and because Julia allows us to write computer code that is as close as possible to the relevant mathematical equations. Please note that all of the Julia code in the text is written to maximize clarity rather than speed.

We have also written matching Python code, which can be found in the same repository. We find Python very practical for efficient dynamic programming. But Python implementations are sometimes not as clean as Julia, and they tend to be very library specific. That is why we chose Julia for programs embedded in the text.

It will be straightforward to port our Python and Julia code to other platforms should readers wish to do so.

We worked hard to mix rigorous theory with exciting applications. Despite the various layers of abstractions used to unify the theory, the results are practical, being motivated by important optimization problems from economics and finance.

This book is one of a series being written in partnership with the QuantEcon organization, with funding generously provided by Schmidt Futures (see acknowledgments below). There is some overlap with the first book in the series, Sargent and

[1] In terms of computation, finite states and actions can closely approximate the continuous case. For example, the set of 64-bit floating point numbers typically ranges from 10^{-308} to 10^{308}, and in each unit length interval there are more than one billion floats.

Stachurski (2022), particularly on the topic of Markov chains. Although repetition is sometimes undesirable, we decided that some overlap would be useful, since it saves readers from having to jump between two documents.

We are greatly indebted to Jim Savage and Schmidt Futures for generous financial support, as well as to Shu Hu, Smit Lunagariya, Maanasee Sharma and Chien Yeh for outstanding research assistance. We are grateful to Makoto Nirei for hosting John Stachurski at the University of Tokyo in June and July 2022, where significant progress was made.

We also thank Quentin Batista, Fernando Cirelli, Chase Coleman, Ippei Fujiwara, Saya Ikegawa, Fazeleh Kazemian, Dawie van Lill, Qingyin Ma, Simon Mishricky, Pietro Monticone, Flint O'Neil, Zejin Shi, Akshay Shanker, Arnav Sood, Alexis Akira Toda, Natasha Watkins, and Jingni Yang for many important fixes, comments and suggestions. Alexis Akira Toda and Jingni Yang read the majority of the manuscript and made many constructive suggestions. Finally, Jesse Perla provided insightful comments on our code. This book would not have reached the finish line without so much support from our colleagues, students and friends.

Common Symbols

$\mathbb{1}\{P\}$	indicator, equal to 1 if statement P is true and 0 otherwise
$\alpha := 1$	α is defined as equal to 1
$f \equiv 1$	function f is everywhere equal to 1
$\wp(A)$	the power set of A; that is, the collection of all subsets of set A
$[n]$	$\{1, \ldots, n\}$
$\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$ and $\mathbb{C}$	the natural, integer, real and complex numbers respectively
$\mathbb{Z}_+$, $\mathbb{R}_+$, etc.	the nonnegative elements of $\mathbb{Z}$, $\mathbb{R}$, etc.
$\lvert x\rvert$ for $x \in \mathbb{R}$	the absolute value of x
$\lvert\lambda\rvert$ for $\lambda \in \mathbb{C}$	the modulus of λ (i.e., $\sqrt{a^2 + b^2}$ when $\lambda = a + ib$)
$\lvert B\rvert$ for set B	the cardinality of (i.e., number of elements in) B
$\mathbb{R}^n$	all n-tuples of real numbers
$x \leqslant y \quad (x, y \in \mathbb{R}^n)$	$x_i \leqslant y_i$ for $i = 1, \ldots n$ (pointwise partial order)
$x \ll y \quad (x, y \in \mathbb{R}^n)$	$x_i < y_i$ for $i = 1, \ldots n$
$\mathscr{D}(F)$	the set of distributions (or probability mass functions) on F
$\mathbb{R}^M$	all functions from M to $\mathbb{R}$
$i\mathbb{R}^M$	the set of increasing functions in $\mathbb{R}^M$
$\mathbb{L}(\mathsf{X})$	the set of linear operators on $\mathbb{R}^{\mathsf{X}}$
$\mathbb{M}(\mathsf{X})$	the set of Markov operators in $\mathbb{L}(\mathsf{X})$ (see §2.3.3.4)
$\langle a, b\rangle$	the inner product of vectors a and b
$\vee_{\alpha\in A} u_\alpha$	supremum of $\{u_\alpha\}_{\alpha\in A}$
$\wedge_{\alpha\in A} u_\alpha$	infimum of $\{u_\alpha\}_{\alpha\in A}$
IID	independent and identically distributed
$X \stackrel{d}{=} Y$	X and Y have the same distribution
$X \sim F$	X has distribution F
$F \preceq_{\mathrm{F}} G$	G first order stochastically dominates F

Common Abbreviations

SDF	Stochastic discount factor (see §6.2.1.2 and §6.2.1.3)
MDP	Markov decision process (see §5.1.1)
VFI	Value function iteration (see §5.1.4.1)
HPI	Howard policy iteration (see §5.1.4.2)
OPI	Optimistic policy iteration (see §5.1.4.4)
RDP	Recursive decision process (see §8.1.1)
ADP	Abstract dynamic program (see §8.3.1)
SDP	Stable dynamic program (see §8.3.2)

Chapter 1

Introduction

The temporal structure of a typical dynamic program is

```
an initial state X_0 is given
t ← 0
while t < T do
    the controller of the system observes the current state X_t
    the controller chooses an action A_t
    the controller receives a reward R_t that depends on the current state and
      action
    the state updates to X_{t+1}
    t ← t + 1
end
```

The state X_t is a vector listing the current values of a set of variables deemed relevant to choosing the current action. The action A_t is a vector describing choices of a set of decision variables. If $T < \infty$ then the problem has a **finite horizon**. Otherwise it is an **infinite horizon** problem. Figure 1.1 illustrates the first two rounds of the dynamic program. As shown in the figure, the rule for the state update depends on the current state and action. The update rule can also depend on shocks and other random elements.

For a decision maker facing such a system, dynamic programming provides a way to maximize expected *lifetime* reward. This value aggregates the reward sequence $(R_t)_{t \geqslant 0}$ according to some specified method that can vary across applications. A typical case takes an expected discounted sum, such as $\mathbb{E} \sum_{t \geqslant 0} \beta^t R_t$ for some $\beta \in (0, 1)$.

Figure 1.1: A dynamic program

Example 1.0.1. A manager wants to set prices and inventories to maximize a firm's **expected present value** (EPV), which, given interest rate r, is defined as

$$\mathbb{E}\left[\pi_0 + \frac{1}{1+r}\pi_1 + \left(\frac{1}{1+r}\right)^2 \pi_2 + \cdots\right]. \tag{1.1}$$

Here X_t will be a vector that quantifies the size of the inventories, prices set by competitors and other factors relevant to profit maximization. The action A_t sets current prices and orders of new stock. The current reward R_t is current profit π_t, and the profit stream $(\pi_t)_{t\geqslant 0}$ is aggregated into a lifetime reward via (1.1).

Dynamic programming has a *vast* array of applications, from robotics and artificial intelligence to the sequencing of DNA. Dynamic programming is used every day to control aircraft, route shipping, test products, recommend information on media platforms and solve major research problems. Some companies now produce specialized computer chips that are designed for specific dynamic programs.

Within economics and finance, dynamic programming is applied to topics including unemployment, monetary policy, fiscal policy, asset pricing, firm investment, wealth dynamics, inventory control, commodity pricing, sovereign default, the division of labor, natural resource extraction, human capital accumulation, retirement decisions, portfolio choice, and dynamic pricing. We discuss some of these applications in the chapters below.

The core theory of dynamic programming is relatively simple and concise. But implementation can be computationally demanding. That situation provides one of the major challenges facing the field of dynamic programming.

Example 1.0.2. To illustrate how computationally demanding problems can arise, consider again Example 1.0.1. Suppose that, for each book, a book retailer chooses to hold between 0 and 10 copies. If there are 100 books to choose from, then the number of possible combinations for her inventories is 11^{100}, about 20 orders of magnitude larger than the number of atoms in the known universe. In reality there are probably many more books to choose from, as well as other factors in the business environment that affect the choices of the retailer.

In this book we discuss fundamental theory, traditional economic applications and recent applications with computationally demanding environments. We also cover recent trends towards more sophisticated specifications of lifetime rewards, often called recursive preferences. Throughout the text, theory and computation are combined, since, for interesting problems, brute-force computation is futile, while theory alone provides limited insight. The interplay between interesting applications, fundamental theory, computational methods and evolving hardware capability makes dynamic programming exciting.

1.1 Bellman Equations

In this section we introduce the recursive structure of dynamic programming in a simple setting. After solving a finite-horizon model, we consider an infinite-horizon version and explain how it produces a system of nonlinear equations. Then we turn to methods for solving such systems.

1.1.1 Finite-Horizon Job Search

We begin with a celebrated model of job search created by McCall (1970). McCall analyzed the decision problem of an unemployed worker in terms of current and likely future wage offers, impatience, and the availability of unemployment compensation. Here we study a simple version of the model in which essential ideas of dynamic programming are particularly clear.

Readers who are familiar with Bellman equations can skim this section quickly and proceed directly to §1.2.

1.1.1.1 A Two-Period Problem

Imagine someone who begins her working life at time $t = 1$ without employment. While unemployed, she receives a new job offer paying wage W_t at each date t. She

can accept the offer and work *permanently* at that wage level or reject the offer, receive unemployment compensation c, and draw a new offer next period. We assume that the wage offer sequence is IID and nonnegative, with distribution φ. In particular,

- $\mathsf{W} \subset \mathbb{R}_+$ is a finite set of possible wage outcomes and
- $\varphi \colon \mathsf{W} \to [0, 1]$ is a probability distribution on W, assigning a probability $\varphi(w)$ to each possible wage outcome w.

The worker is impatient. Impatience is parameterized by a time discount factor $\beta \in (0, 1)$, so that the present value of a next-period payoff of y dollars is βy. Since $\beta < 1$, the worker will be tempted to accept reasonable offers, rather than to wait for better ones. A key question is how long to wait.

Suppose as a first step that working life is just two periods. To solve our problem we work backwards, starting at the final date $t = 2$, after W_2 has been observed.[1] If she is already employed, the agent has no decision to make: she continues working at her current wage. If she is unemployed, then she should take the largest of c and W_2.

Now we step back to $t = 1$. At this time, having received offer W_1, the unemployed worker's options are (a) accept W_1 and receive it in both periods or (b) reject it, receive unemployment compensation c, and then, in the second period, choose the maximum of W_2 and c.

Let's assume that the agent seeks to maximize expected present value. The EPV of option (a) is $W_1 + \beta W_1$, which is also called the **stopping value**. The EPV of option (b), also called the **continuation value**, is $h_1 := c + \beta \, \mathbb{E} \max\{c, W_2\}$. More explicitly,

$$h_1 = c + \beta \sum_{w' \in \mathsf{W}} v_2(w') \varphi(w') \quad \text{where} \quad v_2(w) := \max\{c, w\}. \tag{1.2}$$

The optimal choice at $t = 1$ is now clear: accept the offer if $W_1 + \beta W_1 \geqslant h_1$ and reject otherwise. A decision tree is shown in Figure 1.2.

1.1.1.2 Value Functions

A key idea in dynamic programming is to use "value functions" to track maximal lifetime rewards from a given state at a given time. The **time 2 value function** v_2 defined

[1] The procedure of solving the last period first and then working back in time is called **backward induction**. Starting with the last period makes sense because there is no future to consider.

Figure 1.2: Decision tree for a two period problem

Figure 1.3: The value function v_1 and the reservation wage

in (1.2) returns the maximum value obtained in the final stage for each possible realization of the time 2 wage offer. The **time 1 value function** v_1 evaluated at $w \in \mathsf{W}$ is

$$v_1(w) := \max\left\{w + \beta w,\ c + \beta \sum_{w' \in \mathsf{W}} v_2(w')\varphi(w')\right\}. \tag{1.3}$$

It represents the present value of expected lifetime income after receiving the first offer w, conditional on choosing optimally in both periods.

The value function is shown in Figure 1.3. Figure 1.3 also shows the **reservation wage**

$$w_1^* := \frac{h_1}{1+\beta}. \tag{1.4}$$

It is the w that solves the indifference condition

$$w + \beta w = c + \beta \sum_{w' \in \mathsf{W}} v_2(w')\varphi(w'),$$

and equates the value of stopping to the value of continuing. For an offer W_1 above w_1^*, the stopping value exceeds the continuation value. For an offer below the reservation wage, the reverse is true. Hence, the optimal choice for the agent at $t = 1$ is completely described by the reservation wage.

Parameters and functions underlying the figure are shown in Listing 1.

Studying (1.4) is already instructive. For example, we can see that higher unemployment compensation c shifts up the continuation value h_1 and increases the reservation wage. As a result, the agent will, on average, spend more time unemployed when unemployment compensation is higher.

Exercise 1.1.1. If unemployment compensation increases unemployment duration, should we conclude that increasing such compensation is detrimental to society? Provide some thoughts on this question based on intuition from the McCall model.

1.1.1.3 Three Periods

Now let's suppose that the agent works in period $t = 0$ as well as $t = 1, 2$. Figure 1.4 shows the decision tree for the three periods. Notice that the subtree containing nodes 1 and 2 is exactly the same as the whole decision tree for the two-period problem in Figure 1.2. We will use this to find optimal actions.

At $t = 0$, the value of accepting the current offer W_0 is $W_0 + \beta W_0 + \beta^2 W_0$, while maximal value of rejecting and waiting, is c plus, after discounting by β, the maximum value that can be obtained by behaving optimally from $t = 1$. Fortunately, this value has already been calculated: it is just $v_1(W_1)$, as given in (1.3)!

Maximal time zero value $v_0(w)$ is the maximum of the value of these two options, given $W_0 = w$, so we can write

$$v_0(w) = \max \left\{ w + \beta\, w + \beta^2\, w,\ c + \beta \sum_{w' \in \mathsf{W}} v_1(w')\varphi(w') \right\}. \tag{1.5}$$

By plugging v_1 from (1.3) into this expression, we can determine v_0, as well as the optimal action, which is the one that achieves the largest value in the max term in (1.5).

Figure 1.4 illustrates how the backward induction process works. The last period value function v_2 is trivial to obtain. With v_2 in hand we can compute v_1. With v_1 in hand we can compute v_0. Once all the value functions are available, we can calculate whether to accept or reject at each point in time.

Exercise 1.1.2. The optimal action at time $t = 0$ is determined by a time zero reservation wage w_0^*. The agent should accept the time zero wage offer if and only if W_0 exceeds w_0^*. Calculate w_0^* for this problem, by analogy with w_1^* in (1.4).

```julia
using Distributions

"Creates an instance of the job search model, stored as a NamedTuple."
function create_job_search_model(;
        n=50,         # wage grid size
        w_min=10.0,   # lowest wage
        w_max=60.0,   # highest wage
        a=200,        # wage distribution parameter
        b=100,        # wage distribution parameter
        β=0.96,       # discount factor
        c=10.0        # unemployment compensation
    )
    w_vals = collect(LinRange(w_min, w_max, n+1))
    ϕ = pdf(BetaBinomial(n, a, b))
    return (; n, w_vals, ϕ, β, c)
end

" Computes lifetime value at t=1 given current wage w_1 = w. "
function v_1(w, model)
    (; n, w_vals, ϕ, β, c) = model
    h_1 = c + β * max.(c, w_vals)'ϕ
    return max(w + β * w, h_1)
end

" Computes reservation wage at t=1. "
function res_wage(model)
    (; n, w_vals, ϕ, β, c) = model
    h_1 = c + β * max.(c, w_vals)'ϕ
    return h_1 / (1 + β)
end
```

Listing 1: Computing v_1 and w_1^* (`two_period_job_search.jl`)

Figure 1.4: Decision tree for job seeker

Notice how we broke the three period problem down into a pair of two period problems, given by the two equations (1.3) and (1.5). Breaking many-period problems down into a sequence of two period problems is the essence of dynamic programming. The recursive relationships between v_0 and v_1 in (1.5), as well as between v_1 and v_2 in (1.3), are examples of what are called **Bellman equations**. We will see many other examples shortly.

Exercise 1.1.3. Extend the above arguments to T time periods, where T can be any finite number. Using Julia or any other suitable programming language, write a function that takes T as an argument and returns $(w_0^*, \ldots, w_T^*)$, the sequence of reservation wages for each period.

1.1.2 Infinite Horizon

Next we consider an infinite horizon problem that in some ways is more challenging but in other ways simpler. On one hand, the lack of a terminal period means that backward induction requires a subtler justification. On the other hand, the infinite horizon means that the agent always faces an infinite future, so that we only have to study a single value function and need not keep track of the number of remaining periods in the problem. This will become clearer as the section unfolds.[2]

With the above discussion in mind, let us consider a worker who aims to maximize

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t R_t, \tag{1.6}$$

where $R_t \in \{c, W_t\}$ is earnings at time t. As before, jobs are permanent, so accepting a job at a given wage means earning that wage in every subsequent period.

Let's clarify our assumptions:

Assumption 1.1.1. The wage process satisfies $(W_t)_{t \geqslant 0} \overset{\text{IID}}{\sim} \varphi$ where $\varphi \in \mathscr{D}(\mathsf{W})$ and $\mathsf{W} \subset \mathbb{R}_+$ with $|\mathsf{W}| < \infty$. The parameters c and β are positive and $\beta < 1$.

Here and below, for any finite or countable set F, the symbol $\mathscr{D}(F)$ indicates the set of distributions (or probability mass functions) on F.

[2]Incidentally, imposing an infinite horizon is not the same as assuming humans live forever. Rather, it corresponds to the idea that humans have no specific "termination" date. More generally, we can understand an infinite horizon as an approximation to a finite horizon in which observations are recorded at relatively high frequency and no clear termination date exists.

As with the finite state case, infinite-horizon dynamic programming involves a two step procedure that first assigns values to states and then deduces optimal actions given those values. We begin with an intuitive discussion and then formalize the main ideas.

To trade off current and future rewards optimally, we need to compare current payoffs we get from our two choices with the states that those choices lead to and the maximum value that can be extracted from those states. But how do we calculate the maximum value that can be extracted from each state when lifetime is infinite?

Consider first the present expected lifetime value of being employed with wage $w \in \mathsf{W}$. This case is easy because, under the current assumptions, workers who accept a job are employed forever. Lifetime payoff is

$$w + \beta w + \beta^2 w + \cdots = \frac{w}{1-\beta}. \tag{1.7}$$

How about maximum present expected lifetime value attainable when entering the current period unemployed with wage offer w in hand? Denote this (as yet unknown) value by $v^*(w)$. We call v^* the **value function**. While v^* is not trivial to pin down, the task is not impossible. Our first step in the right direction is to observe that it satisfies the **Bellman equation**

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ c + \beta \sum_{w' \in \mathsf{W}} v^*(w')\varphi(w')\right\} \tag{1.8}$$

at every $w \in \mathsf{W}$. (Here w' is the offer next period.)

Our reasoning is as follows: The first term inside the max operation is the **stopping value**, or lifetime payoff from accepting current offer w. The second term inside the max operation is the **continuation value**, or current expected value of rejecting and behaving optimally thereafter. Maximal value is obtained by selecting the largest of these two alternatives.

At this point, you should note the similarity between (1.8) and our finite horizon Bellman equations (1.3) and (1.5). The only real difference is that the value function is no longer time-dependent. To repeat, this is because the worker always looks forward toward an infinite horizon, regardless of the current date.

Mathematically, (1.8) is viewed as an equation to be solved for a function $v^* \in \mathbb{R}^{\mathsf{W}}$, the set of all functions from W to $\mathbb{R}$. Once we have solved for v^* (assuming this is possible), optimal choices can be made by observing current w and then choosing the largest of the two alternatives on the right-hand side of (1.8), just as we did in the finite horizon case. This idea—that optimal choices can be made by computing the

value function and maximizing the right-hand side of the Bellman equation—is called **Bellman's principle of optimality**, and will be a cornerstone of what follows. Later we prove it carefully in a very general setting.

The first step of this approach is to solve for v^*. For this problem we use fixed point theory, which is the subject of the next section. Later, in §1.3, we return to the job search problem and apply fixed point theory to solving for v^*.

1.2 Stability and Contractions

In this section we cover enough fixed point theory to solve the infinite horizon job search problem. (In Chapter 2 we consider more general results.) Readers who are familiar with the Neumann series lemma and Banach's fixed point theorem can skim this section and proceed to §1.3.

1.2.1 Vector Space

To begin, we recall some fundamental properties of real numbers, finite-dimensional vector space, basic topology and equivalence of norms.

1.2.1.1 Real and Complex Vectors

For the most part, we are interested in vectors where the elements are real numbers (as opposed to complex numbers). Prior to investigating such vectors, let us recall several definitions concerning the real line $\mathbb{R}$.

Given $a, b \in \mathbb{R}$, let $a \vee b := \max\{a, b\}$ and $a \wedge b := \min\{a, b\}$. The **absolute value** of $a \in \mathbb{R}$ is defined as $|a| := a \vee (-a)$. At some point you might wish to review some of the elementary concepts from real analysis in Appendix §A, such as suprema, infima, minima, maxima, and convergence.

A **real-valued vector** $u = (u_1, \ldots, u_n)$ is a finite real sequence with $u_i \in \mathbb{R}$ as the i-th element. The set of all real vectors of length n is denoted by $\mathbb{R}^n$. The **inner product** of n-vectors $(u_1, \ldots, u_n)$ and $(v_1, \ldots, v_n)$ is $\langle u, v \rangle := \sum_{i=1}^n u_i v_i$.

The set $\mathbb{C}$ of complex numbers is defined in the appendix to Sargent and Stachurski (2022) and many other places; as is the set $\mathbb{C}^n$ of all complex-valued n-vectors. We assume readers have some idea of what complex numbers are and know how to compute their modulus.

Exercise 1.2.1. Let α, s and t be real numbers. Show that $\alpha \vee (s+t) \leqslant s + \alpha \vee t$ whenever $s \geqslant 0$.

1.2.1.2 Norms

Readers will be familiar with the ordinary **Euclidean norm** on real vector space, defined as

$$\|u\| := \sqrt{\langle u, u\rangle} \qquad (u \in \mathbb{R}^n).$$

We also wish to work with some alternative norms on $\mathbb{R}^n$, which are important for applications of fixed point theory because they provide more flexibility when checking the conditions of various results.

As a first step, we recall that a function $\|\cdot\|\colon \mathbb{R}^n \to \mathbb{R}$ is called a **norm** on $\mathbb{R}^n$ if, for any $\alpha \in \mathbb{R}$ and $u, v \in \mathbb{R}^n$,

(a)	$\|u\| \geqslant 0$	(nonnegativity)
(b)	$\|u\| = 0 \iff u = 0$	(positive definiteness)
(c)	$\|\alpha u\| = \|\alpha\|\|u\|$ and	(positive homogeneity)
(d)	$\|u+v\| \leqslant \|u\| + \|v\|$	(triangle inequality)

The Euclidean norm is a norm on $\mathbb{R}^n$, as suggested by its name. The Euclidean norm satisfies the **Cauchy–Schwarz inequality**

$$|\langle u, v\rangle| \leqslant \|u\| \cdot \|v\| \quad \text{for all } u, v \in \mathbb{R}^n.$$

This inequality can be used to prove that the triangle inequality is valid for the Euclidean norm (see, e.g., Kreyszig (1978)).

Example 1.2.1. The ℓ_1 **norm** of a vector $u = (u_1, \ldots, u_n) \in \mathbb{R}^n$ is defined by

$$\|u\|_1 := \sum_{i=1}^{n} |u_i|. \tag{1.9}$$

In machine learning applications, $\|\cdot\|_1$ is sometimes called the "Manhattan norm," and $d_1(u, v) := \|u - v\|_1$ is called the "Manhattan distance" or "taxicab distance" between vectors u and v. We will refer to it more simply as the ℓ_1 **distance** or ℓ_1 **deviation**.

Exercise 1.2.2. Verify that the ℓ_1 norm on $\mathbb{R}^n$ satisfies (a)–(d) above.

The ℓ_1 norm and the Euclidean norm are special cases of the so-called **ℓ_p norm**, which is defined for $p \geqslant 1$ by

$$\|u\|_p := \left(\sum_{i=1}^{n} |u_i|^p \right)^{1/p}. \tag{1.10}$$

It can be shown that $u \mapsto \|u\|_p$ is a norm for all $p \geqslant 1$, as suggested by the name (see, e.g., Kreyszig (1978)). For this norm, the subadditivity in (d) is called **Minkowski's inequality**.

Since the Euclidean case is obtained by setting $p = 2$, the Euclidean norm is also called the ℓ_2 norm, and we write $\|\cdot\|_2$ rather than $\|\cdot\|$ when extra clarity is required.

Exercise 1.2.3. Prove that the **supremum norm** (or **ℓ_∞ norm**), defined by $\|u\|_\infty := \max_{i=1}^n |u_i|$, is also a norm on $\mathbb{R}^n$.

(The symbol $\|u\|_\infty$ is used because, for all $u \in \mathbb{R}^n$, we have $\|u\|_p \to \|u\|_\infty$ as $p \to \infty$.)

For the next exercise, we recall that the **indicator function** of logical statement P, denoted here by $\mathbb{1}\{P\}$, takes value 1 (resp., 0) if P is true (resp., false). For example, if $x, y \in \mathbb{R}$, then

$$\mathbb{1}\{x \leqslant y\} = \begin{cases} 1 & \text{if } x \leqslant y \\ 0 & \text{otherwise.} \end{cases}$$

If $A \subset S$, where S is any set, then $\mathbb{1}_A(x) := \mathbb{1}\{x \in A\}$ for all $x \in S$.

Exercise 1.2.4. The so-called ℓ_0 "norm" $\|u\|_0 := \sum_{i=1}^n \mathbb{1}\{u_i \neq 0\}$, routinely used in data science applications, is *not* in fact a norm on $\mathbb{R}^n$. Prove this.

1.2.1.3 Equivalence of Vector Norms

One of the most fundamental properties of finite-dimensional normed vector space is that all norms are "equivalent." Let's review this result and discuss why it matters.

To begin, recall that when u and $(u_m) := (u_m)_{m \in \mathbb{N}}$ are all elements of $\mathbb{R}^n$, we say that (u_m) **converges** to u and write $u_m \to u$ if

$$\|u_m - u\| \to 0 \text{ as } m \to \infty \text{ for some norm } \|\cdot\| \text{ on } \mathbb{R}^n.$$

It might seem that this definition is imprecise. Don't we need to clarify that the convergence is with respect to a particular norm?

In fact we do not. This is because any two norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on $\mathbb{R}^n$ are **equivalent**, in the sense that there exist finite positive constants M, N such that

$$M\|u\|_a \leqslant \|u\|_b \leqslant N\|u\|_a \quad \text{for all } u \in \mathbb{R}^n. \tag{1.11}$$

(See, e.g., Kreyszig (1978).)

Exercise 1.2.5. Let us write $\|\cdot\|_a \sim \|\cdot\|_b$ if there exist finite M, N such that (1.11) holds. Prove that $\sim$ is an equivalence relation on the set of norms on $\mathbb{R}^n$.

Exercise 1.2.6. Let $\|\cdot\|_a$ and $\|\cdot\|_b$ be any two norms on $\mathbb{R}^n$. Given a point u in $\mathbb{R}^n$ and a sequence (u_m) in $\mathbb{R}^n$, use (1.11) to confirm that $\|u_m - u\|_a \to 0$ implies $\|u_m - u\|_b \to 0$ as $m \to \infty$.

The next exercise reminds us that, in finite dimensions, pointwise convergence and norm convergence are the same thing.

Exercise 1.2.7. Let $\|\cdot\|$ be any norm on $\mathbb{R}^n$. Fixing a point u in $\mathbb{R}^n$ and a sequence (u_m) in $\mathbb{R}^n$, let u^i and u_m^i be the i-th component of u and u_m respectively. Show that $u_m^i \to u^i$ for all $i \in \{1, \ldots, n\}$ if and only if $\|u_m - u\| \to 0$.

Recall that a set $C \subset \mathbb{R}^n$ is called **closed** in $\mathbb{R}^n$ if, for all $u \in \mathbb{R}^n$ and sequences $(u_m) \subset \mathbb{R}^n$ with $u_m \in C$ for all m such that $u_m \to u$ as $m \to \infty$, we also have $u \in C$. A set G is called **open** if G^c is closed. A self-map T on $U \subset \mathbb{R}^n$ is called **continuous at** $u \in U$ if $Tu_m \to Tu$ for any $(u_m) \subset U$ with $u_m \to u$; and **continuous** if T is continuous at every $u \in U$. These notions are independent of any norm, since convergence is independent of the choice of norm.

1.2.1.4 Neumann Series

Next we discuss geometric series in matrix space, along with the Neumann series lemma, which is one of the most useful results in applied and numerical analysis.

Before starting we recall that if $A = (a_{ij})$ is an $n \times n$ matrix with i, j-th element a_{ij}, then the definition of matrix multiplication tells us that for $u \in \mathbb{R}^n$, the i-th element of Au is $\sum_{j=1}^n a_{ij}u_j$, while the j-th element of $u^\top A$ is $\sum_{i=1}^n a_{ij}u_i$. We can think of $u \mapsto Au$ and $u \mapsto u^\top A$ is two different mappings, each of which takes an n-vector and produces a new n-vector.

Remark 1.2.1. In this text, we follow the mathematical convention that a vector in $\mathbb{R}^n$ is just an n-tuple of real values. This coincides with the viewpoint of languages like Julia and Python: vectors are, by default, "flat" arrays. At the same time, if we use vectors in matrix algebra, they should be understood as column vectors unless we state otherwise.

If A is an $n \times n$ matrix, then $\lambda \in \mathbb{C}$ is called an **eigenvalue** of A if there exists a nonzero $e \in \mathbb{C}^n$ such that $Ae = \lambda e$. (Here $\mathbb{C}$ is the set of complex numbers and $\mathbb{C}^n$ is the set of complex n-vectors.) The vector e satisfying this equality is called an **eigenvector** of A and (λ, e) is called an **eigenpair**.

In Julia, we can check for the eigenvalues of a given square matrix A via `eigvals(A)`. The code

```
using LinearAlgebra
A = [0 -1;
     1  0]
println(eigvals(A))
```

produces

```
2-element Vector{ComplexF64}:
 0.0 - 1.0im
 0.0 + 1.0im
```

Here `im` stands for i, the imaginary unit, so the eigenvalues of A are $-i$ and i.

Let's begin by discussing geometric series in one dimension. Consider the one-dimensional linear equation $u = au + b$, where a, b are given and u is unknown. Regarding the solution u^*, we have

$$|a| < 1 \quad \Longrightarrow \quad u^* = \frac{b}{1-a} = \sum_{k \geqslant 0} a^k b. \tag{1.12}$$

This scalar result extends naturally to vectors. To show this we suppose that u and b are column vectors in $\mathbb{R}^n$, and that A is an $n \times n$ matrix. We consider the vector equation $u = Au + b$. For the next result, we recall that the **spectral radius** of A is defined as

$$r(A) := \max\{|\lambda| : \lambda \text{ is an eigenvalue of } A\} \tag{1.13}$$

Here $|\lambda|$ indicates the modulus of the complex number λ.

With I as the $n \times n$ identity matrix, we can state the following result, which is the multivariate extension of (1.12):

Theorem 1.2.1 (Neumann Series Lemma)**.** *If $r(A) < 1$, then $I - A$ is nonsingular and $u = Au + b$ has the unique solution*

$$u^* = (I - A)^{-1}b = \sum_{k \geqslant 0} A^k b.$$

It is implicit in Theorem 1.2.1 that the sum $\sum_{k\geqslant 0} A^k$ converges. The code in Listing 2 shows how to compute the spectral radius of an arbitrary matrix A in Julia. The print statement produces `0.5828`, so, for this matrix, $r(A) < 1$.

```julia
using LinearAlgebra
r(A) = maximum(abs(λ) for λ in eigvals(A))  # Spectral radius
A = [0.4 0.1;                               # Test with arbitrary A
     0.7 0.2]
print(r(A))
```

Listing 2: Computing the spectral radius (`compute_spec_rad.jl`)

Exercise 1.2.8. Prove that $r(\alpha B) = |\alpha|\, r(B)$ for all $\alpha \in \mathbb{R}$.

The rest of this section works through the proof of the Neumann series lemma, with several parts left as exercises. An intuitive proof of the lemma runs as follows. If $S := \sum_{k\geqslant 0} A^k$, then

$$I + AS = I + A \sum_{k\geqslant 0} A^k = I + A + A^2 + \cdots = S.$$

Rearranging $I + AS = S$ gives $S = (I - A)^{-1}$. Since $u = Au + b$ is equivalent to $(I - A)u = b$, we have $u = (I - A)^{-1}b = Sb$, which matches the claim in the Neumann series lemma.

This argument lacks rigor, however. To complete it, we need to prove that (a) the sum $\sum_{k\geqslant 0} A^k$ converges and (b) the matrix $I - A$ is invertible.

To resolve these issues, we introduce the **matrix norm**

$$\|B\|_\infty := \max_{i,j} |b_{ij}| \qquad (B = (b_{ij})). \tag{1.14}$$

Lemma 1.2.2. *If B is any square matrix, then*

$$r(B)^k \leqslant \|B^k\|_\infty \text{ for all } k \in \mathbb{N} \qquad \text{and} \qquad \|B^k\|_\infty^{1/k} \to r(B) \text{ as } k \to \infty.$$

The second result in Lemma 1.2.2 is a version of **Gelfand's formula**.

Exercise 1.2.9. Using Lemma 1.2.2, show that

(i) $r(B) < 1$ implies $\|B^k\|_\infty \to 0$ as $k \to \infty$.

(ii) $r(B) > 1$ implies $\|B^k\|_\infty \to \infty$ as $k \to \infty$.

Exercise 1.2.10. Prove: $r(A) < 1$ implies that the series $\sum_{k\geqslant 0} A^k$ converges, in the sense that every element of the matrix $S_K := \sum_{k=0}^{K} A^k$ converges as $K \to \infty$.

From this last result, one can show that $(I - A)^{-1}$ exists:

Exercise 1.2.11. Prove this claim by showing that, when $\sum_{k\geqslant 0} A^k$ exists, the inverse of $I - A$ exists and indeed $(I - A)^{-1} = \sum_{k\geqslant 0} A^k$.[3]

Listing 3 helps illustrate the result in Exercise 1.2.11, although we truncate the infinite sum $\sum_{k\geqslant 0} A^k$ at 50.

The output is `5.621e-12`, which is essentially zero.

1.2.2 Nonlinear Systems

The equation $u = Au + b$ discussed in the previous section is called a linear system.[4] While the Neumann series lemma is important for such systems, it does not apply to *non*linear problems. There are many techniques for handling nonlinear systems. In this section, we present Banach's fixed point theorem. In Chapter 2, we will introduce other methods.

1.2.2.1 Fixed Points

Fixed point theory is a systematic way to approach solving equations. We say that T is a **self-map** on an arbitrary set U if T is a function from U into itself. For a self-map T on U, a point $u^* \in U$ is called a **fixed point** of T in U if $Tu^* = u^*$.

Remark 1.2.2. In fixed point theory, it is common to write Tu for the image of u under the function T, rather than $T(u)$; T is often called an **operator** rather than a function. One reason is that in applications u itself is often a function. In such settings, confusion can be avoided by calling T an operator.

[3] Hint: To prove that A is invertible and $B = A^{-1}$, it suffices to show that $AB = I$.

[4] Some people call it *affine* instead, since $u \mapsto Au + b$ is affine rather than linear when $b \neq 0$.

```
# Primitives
A = [0.4 0.1;
     0.7 0.2]

# Method one: direct inverse
B_inverse = inv(I - A)

# Method two: power series
function power_series(A)
    B_sum = zeros((2, 2))
    A_power = I
    for k in 1:50
        B_sum += A_power
        A_power = A_power * A
    end
    return B_sum
end

# Print maximal error
print(maximum(abs.(B_inverse - power_series(A))))
```

Listing 3: Matrix inversion vs power series (`power_series.jl`)

Example 1.2.2. Let $U = \mathbb{R}^n$ and let T be defined by $Tu = Au + b$, where A and b are as in §1.2.1.4. Since u is a fixed point of T if and only if $u = Au + b$, solving the equation $u = Au + b$ is the same as searching for the fixed point of T.

Example 1.2.3. Every u in set U is fixed under the identity map $I \colon u \mapsto u$.

Example 1.2.4. If $U = \mathbb{N}$ and $Gu = u + 1$, then G has no fixed point.

Figure 1.5 shows another example, for a self-map G on $U := [0, 2]$. Fixed points are numbers $x \in [0, 2]$ where G meets the 45 degree line. In this case there are three.

Exercise 1.2.12. Let U be any set and let T be a self-map on U. Suppose there exists an $\bar{u} \in U$ and an $m \in \mathbb{N}$ such that $T^k u = \bar{u}$ for all $u \in U$ and $k \geqslant m$. Prove that, under this condition, $\bar{u}$ is the unique fixed point of T in U.

Exercise 1.2.13. Let T be a self-map on $U \subset \mathbb{R}^d$. Prove the following: If $T^m u \to u^*$ as $m \to \infty$ for some pair $u, u^* \in U$ and, in addition, T is continuous at u^*, then u^* is a fixed point of T.

Figure 1.5: Graph and fixed points of $G\colon x \mapsto 2.125/(1+x^{-4})$

A natural way to write down general theorems about solving scalar equations, vector equations and more abstract equations is in terms of fixed points. Indeed, an abstract representation of a system of equations is $u = Tu$, where u takes values in an abstract set U and T is a self-map on U. By definition, solutions to this system coincide with fixed points of T.

When considering fixed points, given a self-map T on U, we typically seek conditions on T and U under which the following properties hold:

- T has at least one fixed point on U (existence)
- T has at most one fixed point on U (uniqueness)
- T has a fixed point on U and the fixed point can be computed numerically.

Example 1.2.5. If $U = \mathbb{R}^n$ and T is defined by $Tu = Au + b$, then, by the Neumann series lemma, T has a unique fixed point $u^* \in \mathbb{R}^n$ whenever $r(A) < 1$. Moreover, that fixed point can be computed, at least approximately, by using either $u^* = (I - A)^{-1}b$ or $u^* = \sum_{k \geqslant 0} A^k b$.

1.2.2.2 Global Stability

A self-map T on U is called **globally stable** on U if T has a unique fixed point u^* in U and, moreover, $T^k u \to u^*$ as $k \to \infty$ for all $u \in U$. Here T^k indicates k compositions of T with itself. Global stability is a desirable property in the setting of dynamic programming and a number of our results rely on it.

Exercise 1.2.14. As in Example 1.2.5, let $U = \mathbb{R}^n$ and let T be defined by $Tu = Au + b$. Using induction, prove that

$$T^k u = A^k u + A^{k-1} b + A^{k-2} b + \cdots + Ab + b \tag{1.15}$$

for all $u \in U$ and $k \in \mathbb{N}$. Next, show that T is globally stable on U whenever $r(A) < 1$.

Let T be a self-map on $U \subset \mathbb{R}^n$. We call T **invariant** on $C \subset U$ and call C an **invariant set** if T is also a self-map on C; that is, if $u \in C$ implies $Tu \in C$.

Exercise 1.2.15. Let T be a globally stable self-map on $U \subset \mathbb{R}^n$, with fixed point u^*. Prove the following: If C is closed and T is invariant on C, then $u^* \in C$.

1.2.2.3 Banach's Fixed Point Theorem

Next we present a workhorse theorem for analyzing nonlinear operators: the Banach fixed point theorem.

Let U be a nonempty subset of $\mathbb{R}^n$ and let $\|\cdot\|$ be a norm on $\mathbb{R}^n$. A self-map T on U is called a **contraction** on U with respect to $\|\cdot\|$ if there exists a $\lambda < 1$ such that

$$\|Tu - Tv\| \leqslant \lambda \|u - v\| \quad \text{for all} \quad u, v \in U. \tag{1.16}$$

The constant λ is called the **modulus of contraction**.

Exercise 1.2.16. Let T be a contraction on U with respect to a norm $\|\cdot\|$. Show that, T is continuous on U and has at most one fixed point in U.

Exercise 1.2.17. Let $U = \mathbb{R}^n$ and let $Tx = Ax + b$, where A is $n \times n$ and b is $n \times 1$. Prove that T is a contraction of modulus $\|A\|_\infty$ on U (see (1.14) for the definition) whenever $\|A\|_\infty < 1$.

The importance of contractions stems from the following theorem.

Theorem 1.2.3 (Banach's contraction mapping theorem)**.** *If U is closed in $\mathbb{R}^n$ and T is a contraction of modulus λ on U with respect to some norm $\|\cdot\|$ on $\mathbb{R}^n$, then T has a unique fixed point u^* in U and*

$$\|T^n u - u^*\| \leqslant \lambda^n \|u - u^*\| \quad \textit{for all } n \in \mathbb{N} \textit{ and } u \in U. \tag{1.17}$$

In particular, T is globally stable on U.

We prove Theorem 1.2.3 in stages that start with the following exercises.

Exercise 1.2.18. Let U and T have the properties stated in Theorem 1.2.3. Fix $u_0 \in U$ and let $u_m := T^m u_0$. Show that

$$\|u_m - u_k\| \leqslant \sum_{i=m}^{k-1} \lambda^i \|u_0 - u_1\|$$

holds for all $m, k \in \mathbb{N}$ with $m < k$.

Exercise 1.2.19. Using the results in Exercise 1.2.18, prove that (u_m) is a Cauchy sequence in $\mathbb{R}^n$.

Exercise 1.2.20. Using Exercise 1.2.19, argue that (u_m) has a limit $u^* \in \mathbb{R}^n$. Prove that $u^* \in U$.

Proof of Theorem 1.2.3. In the exercises we proved existence of a point $u^* \in U$ such that $T^m u \to u^*$. The fact that u^* is a fixed point of T now follows from Exercise 1.2.13 and Exercise 1.2.16. Uniqueness is implied by Exercise 1.2.16. The bound (1.17) follows from iteration on the contraction inequality (1.16) while setting $v = u^*$. □

Observe that Banach's contraction mapping theorem extends directly to operators on $\mathbb{R}^{\mathsf{X}}$ when X is finite. Indeed, if $|\mathsf{X}| = n$, C is closed in $\mathbb{R}^{\mathsf{X}}$ and T is a contraction on $C \subset \mathbb{R}^{\mathsf{X}}$, in the sense that, for some $\lambda < 1$,

$$\|Tf - Tg\| \leqslant \lambda \|f - g\| \quad \text{for all} \quad f, g \in C$$

then T has a unique fixed point f^* in C and

$$\|T^n f - f^*\| \leqslant \lambda^n \|f - f^*\| \quad \text{for all } n \in \mathbb{N} \text{ and } f \in \mathbb{R}^{\mathsf{X}}.$$

There is no need to supply a new proof: we just need to identify functions in $\mathbb{R}^{\mathsf{X}}$ with vectors in $\mathbb{R}^n$ under the correspondence (1.19).

Exercise 1.2.21. Let U be a closed subset of $\mathbb{R}^n$ with the property that $u, v \in U$ implies $u \vee v \in U$. Let T_1 and T_2 be contraction maps on U under the supremum norm $\|\cdot\|_\infty$. Prove that the self-map $T : U \to U$ defined by $Tu := (T_1 u) \vee (T_2 u)$ is also contraction on U under the supremum norm.

Exercise 1.2.22. Let T be a contraction of modulus β on $\mathbb{R}^n$ and fixed point $\bar{u}$. Consider the **damped iteration** scheme $u_{n+1} = (1-\alpha)u_n + \alpha T u_n$. Show that, for any choice of u_0, these iterates converge to $\bar{u}$ whenever $0 < \alpha \leqslant 1$.

1.2.3 Successive Approximation

Consider a self-map T on $U \subset \mathbb{R}^n$. We seek algorithms that compute fixed points of T whenever they exist.

1.2.3.1 Iteration

If T is globally stable on U, then a natural algorithm for approximating the unique fixed point u^* of T in U is to pick any $u \in U$ and iterate with T for some finite number of steps:

```
fix u_0 ∈ U and τ > 0
k ← 0
ε ← τ + 1
while ε > τ do
    u_{k+1} ← T u_k
    ε ← ‖u_{k+1} − u_k‖
    k ← k + 1
end
return u_k
```

By the definition of global stability, $(u_k)_{k \geqslant 0}$ converges to u^*. The algorithm just described is called either **successive approximation** or **fixed point iteration**. Listing 4 provides a function that implements this procedure. Distance between points is measured using the ℓ_∞ norm.

Listing 5 applies successive approximation to the map $Tu = Au + b$ using the function defined in `s_approx.jl`. Figure 1.6 shows the sequence of iterates generated by

```
"""
Computes the approximate fixed point of T via successive approximation.

"""
function successive_approx(T,                    # Operator (callable)
                           u_0;                  # Initial condition
                           tolerance=1e-6,       # Error tolerance
                           max_iter=10_000,      # Max iteration bound
                           print_step=25)        # Print at multiples
    u = u_0
    error = Inf
    k = 1

    while (error > tolerance) & (k <= max_iter)
        u_new = T(u)
        error = maximum(abs.(u_new - u))
        if k % print_step == 0
            println("Completed iteration $k with error $error.")
        end
        u = u_new
        k += 1
    end

    if error <= tolerance
        println("Terminated successfully in $k iterations.")
    else
        println("Termination Warning: Error is greater than tolerance.")
    end

    return u
end
```

Listing 4: Successive approximation (`s_approx.jl`)

```
include("s_approx.jl")
using LinearAlgebra

# Compute the fixed point of Tu = Au + b via linear algebra
A, b = [0.4 0.1; 0.7 0.2], [1.0; 2.0]
u_star = (I - A) \ b  # compute (I - A)^{-1} * b

# Compute the fixed point via successive approximation
T(u) = A * u + b
u_0 = [1.0; 1.0]
u_star_approx = successive_approx(T, u_0)

# Test for approximate equality (prints "true")
print(isapprox(u_star, u_star_approx, rtol=1e-5))
```

Listing 5: Using successive approximations to compute u^* (`linear_iter.jl`)

four runs of the successive approximation algorithm, each with a different starting condition u_0. The map and parameters are the same as in Listing 5. It is clear from the figure that a good choice of initial condition (i.e., one that is close to the fixed point) accelerates convergence.

Of course for $Tu = Au + b$ with $r(A) < 1$, there is a more direct method to compute the fixed point: the Neumann series lemma tells us that $u^* = (I - A)^{-1}b$ so we can apply a numerical linear equation solver. However, even for this case, sometimes successive approximation is used instead. One reason is that $(I - A)^{-1}$ can be very large, making application of a linear solver problematic. Another is that we might be satisfied with a quick approximation of the fixed point, computed with a few iterations of T. Both of these situations can arise in dynamic programming.

1.2.3.2 A One-Dimensional Example

To illustrate successive approximations in a nonlinear setting, we use the Solow–Swan growth model, which is a good place to begin presenting the theory of economic growth. A fixed point for the Solow-Swan model can be computed with pencil and paper. The model is a good laboratory for studying how successive approximations might converge to a fixed point.

Figure 1.6: Successive approximation from different initial conditions

One version of the Solow–Swan growth dynamics is

$$k_{t+1} = sf(k_t) + (1-\delta)k_t, \qquad t = 0, 1, \ldots, \tag{1.18}$$

where k_t is capital stock per worker, $f\colon (0,\infty) \to (0,\infty)$ is a production function, $s > 0$ is a saving rate and $\delta \in (0,1)$ is a rate of depreciation. If we set $g(k) := sf(k) + (1-\delta)k$, then iterating with g from some starting point k_0 (i.e., setting $k_{t+1} = g(k_t)$ for all $t \geqslant 0$) generates the sequence in (1.18). We can also understand this process as using successive approximation to compute the fixed point of g.

Exercise 1.2.23. Let $f(k) = Ak^\alpha$ with $A > 0$ and $0 < \alpha < 1$. Show that, while the Solow-Swan map $g(k) = sk^\alpha + (1-\delta)k$ sends $U := (0,\infty)$ into itself, g is *not* a contraction on U. [Hint: use the definition of the derivative of g as a limit and consider the derivative $g'(k)$ for k close to zero.]

Although the model specified in Exercise 1.2.23 does not generate a contraction, it is globally stable. The next exercise asks you to prove this.

Exercise 1.2.24. Show that, in the setting of Exercise 1.2.23, the unique fixed

Figure 1.7: Successive approximation for the Solow–Swan model

point of g in U is

$$k^* := \left(\frac{sA}{\delta}\right)^{1/(1-\alpha)}$$

Prove that, for $k \in U$,

(i) $k \leqslant k^*$ implies $k \leqslant g(k) \leqslant k^*$ and

(ii) $k^* \leqslant k$ implies $k^* \leqslant g(k) \leqslant k$.

Conclude that g is globally stable on U. (Why?)

Figure 1.7 illustrates the dynamics in a 45 degree diagram when $f(k) = Ak^\alpha$. In the top subfigure, $A = 2.0$, $\alpha = 0.3$, $s = 0.3$ and $\delta = 0.4$. The function g is plotted alongside the 45 degree line. When $g(k_t)$ lies strictly above the 45 degree line, then $k_{t+1} = g(k_t) > k_t$ and so capital per worker rises. If $g(k_t) < k_t$ then it falls. A trajectory $(k_t)_{t \geqslant 0}$ that is produced by starting from a particular choice of k_0 is traced out in the figure.

The bottom subfigure is similar, with parameters adjusted to $A = 3.0$, $\alpha = 0.05$, $s = 0.4$ and $\delta = 0.6$.

The figure illustrated that k^* is the unique fixed point of g in U and all sequences converge to it. The second statement can be rephrased as: successive approximation successfully computes the fixed point of g by stepping through the time path of capital.

1.2.4 Finite-Dimensional Function Space

In §1.1.2 we introduced a Bellman equation for the infinite horizon job search problem. The unknown in the Bellman equation is a value function v^*, defined on the set W of possible wage offers. Below we discuss how to solve for this unknown function.

Since the set of wage offers is finite we can write W as $\{w_1, \dots, w_n\}$ for some $n \in \mathbb{N}$. If we adopt this convention and also write $v^*(w_i)$ as v_i^*, then we can view v^* as a vector $(v_1^*, \dots, v_n^*)$ in $\mathbb{R}^n$. The vector interpretation is useful when coding, since vectors (arrays) are an efficient data type.

At the same time, for mathematical exposition, we usually find it more convenient to express function-like objects (e.g., value functions) as functions rather than vectors. Thus, we typically write $v^*(w)$ instead of v_i^*.

Remark 1.2.3. There is a deeper reason why we usually work with functions rather than vectors: when we shift to general state and action spaces in Volume II, objects such as value functions can no longer be represented by finite-dimensional vectors. Instead we must use the language of functional analysis. By adopting this language now, the jump to general spaces will be smoother because terminology and notation are largely unchanged.

The next section clarifies our notation with respect to functions and vectors.

1.2.4.1 Functions vs Vectors

If X is any set and u maps X to $\mathbb{R}$, then we call u a **real-valued function** on X and write $u \colon \mathsf{X} \to \mathbb{R}$. Throughout, the symbol $\mathbb{R}^{\mathsf{X}}$ denotes the set of all real-valued functions on X. This is a special case of the symbol B^A, where A and B are sets, which represents the set of all functions from A to B.

Now let X be finite, so that $\mathsf{X} = \{x_1, \dots, x_n\}$ for some $n \in \mathbb{N}$. The set $\mathbb{R}^{\mathsf{X}}$ is, in essence, the vector space $\mathbb{R}^n$ expressed in different notation. The next lemma clarifies.

Lemma 1.2.4. *If $\mathsf{X} = \{x_1, \dots, x_n\}$, then*

$$\mathbb{R}^{\mathsf{X}} \ni u \quad \longleftrightarrow \quad (u(x_1), \dots, u(x_n)) \in \mathbb{R}^n \tag{1.19}$$

is a one-to-one correspondence between the function space $\mathbb{R}^{\mathsf{X}}$ and the vector space $\mathbb{R}^n$.

The claim in Lemma 1.2.4 is obvious: a real-valued function u on X is uniquely identified by the set of values that it takes on X, which is an n-tuple of real numbers.

Throughout the text, whenever the supporting set X is finite, we freely use the identification in (1.19). For example, if $\|\cdot\|$ is any norm on $\mathbb{R}^n$, then $\|\cdot\|$ extends to $\mathbb{R}^{\mathsf{X}}$ via the identification in (1.19). That is, for $u \in \mathbb{R}^{\mathsf{X}}$, the value $\|u\|$ is given by the norm of the vector $(u(x_1), \ldots, u(x_n)) \in \mathbb{R}^n$.

We say that a subset of $\mathbb{R}^{\mathsf{X}}$ is closed (resp., open, compact, etc.) if the corresponding subset of $\mathbb{R}^n$ is closed (resp., open, compact, etc.)

With these conventions, the Neumann series lemma and Banach's contraction mapping theorem extend naturally from $\mathbb{R}^n$ to $\mathbb{R}^{\mathsf{X}}$. We use these facts without comment in what follows.

1.2.4.2 Distributions

Given finite X, the set of **distributions** on X is written as $\mathscr{D}(\mathsf{X})$ and contains all $\varphi \in \mathbb{R}_+^{\mathsf{X}}$ with $\sum_{x \in \mathsf{X}} \varphi(x) = 1$. Since we can identify any $f \in \mathbb{R}^{\mathsf{X}}$ with a corresponding vector in $\mathbb{R}^n$ (here $n = |\mathsf{X}|$), the set $\mathscr{D}(\mathsf{X})$ can also be thought of as a subset of $\mathbb{R}^n$. This set of vectors (i.e., the nonnegative vectors that sum to unity) is sometimes called the **unit simplex**.

Fix $h \in \mathbb{R}^{\mathsf{X}}$ and $\varphi \in \mathscr{D}(\mathsf{X})$. Let X be a random variable with distribution φ, so that $\mathbb{P}\{X = x\} = \varphi(x)$ for all $x \in \mathsf{X}$. The **expectation** of $h(X)$ is

$$\mathbb{E}h(X) := \sum_{x \in \mathsf{X}} h(x)\varphi(x) = \langle h, \varphi \rangle .$$

The **cumulative distribution function** (CDF) corresponding to φ is the map from X to $\mathbb{R}$ given by

$$\Phi(x) := \mathbb{P}\{X \leqslant x\} = \sum_{x' \in \mathsf{X}} \mathbb{1}\{x' \leqslant x\}\varphi(x').$$

If $\mathsf{X} \subset \mathbb{R}$ and $\tau \in [0, 1]$, then the τ-th **quantile** of X is

$$Q_\tau X := \min\{x \in \mathsf{X} : \Phi(x) \geqslant \tau\}. \tag{1.20}$$

If $\tau = 1/2$, then $Q_\tau X$ is called the **median** of X.

Example 1.2.6. Suppose $\mathsf{X} = \{1, 2, 3\}$. If $\varphi = (0.5, 0.0, 0.5)$ and $X \sim \varphi$, then $\Phi = (0.5, 0.5, 1)$ and $Q_{1/2}(X) = x_1$. The min in (1.20) allows us to select a unique median (even though x_2 is also a reasonable choice).

Intuitively, if the median of X is x, then the median of $X + \alpha$ will be $x + \alpha$. This same logic carries over to arbitrary quantiles, as the next exercise asks you to show.

Figure 1.8: Functions $u \vee v$ and $u \wedge v$

Exercise 1.2.25. Prove that the quantile function is additive over constants. That is, for any $\tau \in [0,1]$, random variable X on X and $\alpha \in \mathbb{R}$, we have $Q_\tau(X+\alpha) = Q_\tau(X)+\alpha$.

1.2.4.3 Pointwise Operations on Functions

If $u, v \in \mathbb{R}^{\mathsf{X}}$ and $\alpha, \beta \in \mathbb{R}$, then the expressions $\alpha u + \beta v$ and uv also represent elements of $\mathbb{R}^{\mathsf{X}}$, defined at $x \in \mathsf{X}$ by

$$(\alpha u + \beta v)(x) = \alpha u(x) + \beta v(x) \quad \text{and} \quad (uv)(x) = u(x)v(x). \tag{1.21}$$

Similarly, $|u|$, $u \vee v$ and $u \wedge v$ are real-valued functions on X defined by

$$|u|(x) = |u(x)|, \quad (u \vee v)(x) = u(x) \vee v(x) \quad \text{and} \quad (u \wedge v)(x) = u(x) \wedge v(x). \tag{1.22}$$

Figure 1.8 illustrates the functions $u \vee v$ and $u \wedge v$ when X is a subset of $\mathbb{R}$.

If $u, v \in \mathbb{R}^{\mathsf{X}}$ are represented by vectors $u = (u_i)_{i=1}^n$ and $v = (v_i)_{i=1}^n$ in $\mathbb{R}^n$ (see (1.19)), then the definitions in (1.22) become

$$|u| := (|u_i|)_{i=1}^n, \quad u \wedge v := (u_i \wedge v_i)_{i=1}^n \quad \text{and} \quad u \vee v := (u_i \vee v_i)_{i=1}^n. \tag{1.23}$$

Figure 1.9 illustrates in $\mathbb{R}^2$.

Let X be a nonempty set and let $\{v_\sigma\} := \{v_\sigma\}_{\sigma \in \Sigma}$ be a finite subset of $\mathbb{R}^{\mathsf{X}}$ (i.e., the index set Σ is finite). Generalizing the definitions above, we set $\vee_\sigma v_\sigma$ to be the

Figure 1.9: The vectors $u \vee v$ and $u \wedge v$ in $\mathbb{R}^2$

pointwise maximum and $\wedge_\sigma v_\sigma$ to be the pointwise minimum. Thus, for arbitrary $x \in \mathsf{X}$,

$$\left(\bigvee_\sigma v_\sigma\right)(x) := \max_{\sigma \in \Sigma} v_\sigma(x) \quad \text{and} \quad \left(\bigwedge_\sigma v_\sigma\right)(x) := \min_{\sigma \in \Sigma} v_\sigma(x).$$

1.2.4.4 A Useful Inequality

We note the following useful inequality.

Lemma 1.2.5. *Let D be a finite set. If f and g are elements of $\mathbb{R}^D$, then*

$$|\max_{z \in D} f(z) - \max_{z \in D} g(z)| \leqslant \max_{z \in D} |f(z) - g(z)|. \tag{1.24}$$

Proof. Fixing $f, g \in \mathbb{R}^D$, we have

$$f = f - g + g \leqslant |f - g| + g$$

$$\therefore \quad \max f \leqslant \max(|f - g| + g) \leqslant \max |f - g| + \max g$$

$$\therefore \quad \max f - \max g \leqslant \max |f - g|$$

Reversing the roles of f and g proves the claim. □

The inequality in Lemma 1.2.5 will be helpful for study dynamic programming problems involving maximization. The next exercise below concerns minimization.

Exercise 1.2.26. Prove that, in the setting of Lemma 1.2.5, we have

$$|\min_{z \in D} f(z) - \min_{z \in D} g(z)| \leqslant \max_{z \in D} |f(z) - g(z)|. \tag{1.25}$$

1.3 Infinite-Horizon Job Search

Armed with useful fixed point methods, let us now return to the job search problem discussed in §1.1.2 and solve for optimal choices more carefully.

1.3.1 Values and Policies

In this section we solve for the value function of an infinite horizon job search problem and use it to make optimal choices.

1.3.1.1 Optimal Choices

Let's recall the strategy for solving the infinite-horizon job search problem we proposed in §1.1.2. The first step is to compute the value function v^*, which solves the Bellman equation

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ c + \beta \sum_{w' \in \mathsf{W}} v^*(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}). \tag{1.26}$$

Suppose for a moment that we can compute v^*, and let

$$h^* := c + \beta \sum_{w'} v^*(w')\varphi(w') \tag{1.27}$$

be the infinite-horizon **continuation value**. The continuation value is the maximal lifetime value that the agent can receive, contingent on deciding to continue today.

With h^* in hand, the optimal decision at any given time, facing current wage draw $w \in \mathsf{W}$, is as follows:

(i) If $w/(1-\beta) \geqslant h^*$, then accept the job offer.

(ii) If not, then reject and wait for the next offer.

This decision maximizes lifetime value given the current offer.

(We will prove below that this decision process is optimal as claimed. For now, however, we focus on computing v^* and h^*.)

1.3.1.2 The Bellman Operator

The method proposed above requires that we solve for v^*. To do so, we introduce a **Bellman operator** T defined at $v \in \mathbb{R}^{\mathsf{W}}$ that is constructed to assure that any fixed point of T solves the Bellman equation and vice versa:

$$(Tv)(w) = \max\left\{\frac{w}{1-\beta},\ c + \beta \sum_{w' \in \mathsf{W}} v(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}). \tag{1.28}$$

Let $V := \mathbb{R}_+^{\mathsf{W}}$ and let $\|\cdot\|_\infty$ be the supremum norm on V. We measure the distance between two elements f, g of V by $\|f-g\| = \max_{w\in\mathsf{W}} |f(w)-g(w)|$. Under this distance, we have the following result.

Proposition 1.3.1. *T is a contraction of modulus β on V.*

A proof of Proposition 1.3.1 is given below. An implication of the proposition is that $T^k v \to v^*$ as $k \to \infty$ for any $v \in V$, so we can compute v^* to any required degree of accuracy by successive approximation.

Our proof of Proposition 1.3.1 uses the elementary bound

$$|\alpha \vee x - \alpha \vee y| \leqslant |x - y| \qquad (\alpha, x, y \in \mathbb{R}) \tag{1.29}$$

Exercise 1.3.1. Verify that (1.29) always holds. [Exercise 1.2.1 might be helpful.]

Proof of Proposition 1.3.1. Take any f, g in V and fix any $w \in \mathsf{W}$. Apply the bound in (1.29) to get

$$\begin{aligned}|(Tf)(w) - (Tg)(w)| &\leqslant \left|c + \beta \sum_{w'} f(w')\varphi(w') - \left(c + \beta \sum_{w'} g(w')\varphi(w')\right)\right| \\ &= \beta \left|\sum_{w'} [f(w') - g(w')]\varphi(w')\right|.\end{aligned}$$

Apply the triangle inequality to obtain

$$|(Tf)(w) - (Tg)(w)| \leqslant \beta \sum_{w'} |f(w') - g(w')|\varphi(w') \leqslant \beta\|f - g\|_\infty.$$

Taking the supremum over all w on the left hand side of this expression leads to

$$\|Tf - Tg\|_\infty \leqslant \beta\|f - g\|_\infty.$$

Since f, g were arbitrary elements of V, the contraction claim is verified. □

1.3.1.3 Optimal Policies

As will become clear over the next few chapters, the field of dynamic programming revolves around finding optimal policies. We briefly introduce the notion of a policy and relate it to the job search application.

In general, for a dynamic program, choices by the controller aim to maximize lifetime rewards and consist of a state-contingent sequence $(A_t)_{t \geqslant 0}$ specifying how the agent acts at each point in time. Since agents are not clairvoyant, it is natural to assume that A_t can depend on present and past events but not future ones. In other words, A_t is a function of the current state X_t and past state-action pairs (A_{t-i}, X_{t-i}) for $i \geqslant 1$. That is,

$$A_t = \sigma_t(X_t, A_{t-1}, X_{t-1}, A_{t-2}, X_{t-2}, \ldots, A_0, X_0)$$

for some function σ_t; σ_t is called a time t **policy function**.

A key insight of dynamic programming is that some problems can be set up so that *the optimal current action can be expressed as a function of the current state X_t.*

Example 1.3.1. In Example 1.0.1, the retailer chooses stock orders and prices in each period. Every quantity relevant to this decision belongs in the current state. It might include not just the level of current inventories and various measures of business conditions, but also information such as rates at which inventories have changed over each of the past six months.

If the current state X_t is enough to determine a current optimal action, then policies are just maps from states to actions. So we can write $A_t = \sigma(X_t)$ for some function σ. A policy function that depends only on the current state is often called a **Markov policy**. Since all policies we consider will be Markov policies, we refer to them more concisely as "policies."

Remark 1.3.1. In the last paragraph, we dropped the time subscript on σ with no loss of generality because we can always include the date t in the current state; i.e., if Y_t is the state without time, then we can set $X_t = (t, Y_t)$). Whether this is necessary depends on the problem at hand. For the job search model with finite horizon, the date matters because opportunities for future earnings decrease with the passage of time. For the infinite horizon version of the problem, in which an agent always looks forward toward an infinite horizon, the only current information that matters to the agent at time t is the wage offer W_t. As a result, the calendar date t does not affect the agent's decision at time t, so there is no need to include time in the state.

In the job search model, the state is the current wage offer and possible actions are to accept or to reject the current offer. With 0 interpreted as reject and 1 understood as accept, the action space is $\{0, 1\}$, so a policy is a map σ from W to $\{0, 1\}$. Let Σ be the set of all such maps.

A policy is an "instruction manual": for an agent following $\sigma \in \Sigma$, if current wage offer is w, the agent always responds with $\sigma(w) \in \{0, 1\}$. The policy dictates whether the agent accepts or rejects at any given wage.

For each $v \in V$, a **v-greedy policy** is a $\sigma \in \Sigma$ satisfying

$$\sigma(w) = \mathbb{1}\left\{\frac{w}{1-\beta} \geqslant c + \beta \sum_{w' \in \mathsf{W}} v(w')\varphi(w')\right\} \quad \text{for all } w \in \mathsf{W}. \tag{1.30}$$

Equation (1.30) says that an agent accepts if $w/(1-\beta)$ exceeds the continuation value computed using v and rejects otherwise. Our discussion of optimal choices in §1.3.1.1 can now be summarized as the recommendation

Adopt a v^*-greedy policy.

This statement is sometimes called Bellman's principle of optimality.

Inserting v^* into (1.30) and rearranging, we can express a v^*-greedy policy via

$$\sigma^*(w) = \mathbb{1}\{w \geqslant w^*\} \quad \text{where } \; w^* := (1-\beta)h^*. \tag{1.31}$$

The quantity w^* in (1.31) is called the **reservation wage**, and parallels the reservation wage that we introduced for the finite-horizon problem. Equation (1.31) states that value maximization requires accepting an offer if and only if it exceeds the reservation wage. Thus, w^* provides a scalar description of an optimal policy.

1.3.2 Computation

Let's turn to computation. In §1.3.2.1, we apply a standard dynamic programming method, called value function iteration. Below, in §1.3.2.2, we apply a more specialized method that uses the structure of the job search problem to accelerate computation.

1.3.2.1 Value Function Iteration

Recall that, by Proposition 1.3.1, we can compute an approximate optimal policy by applying successive approximation via the Bellman operator. In the language of dy-

namic programming, this is called **value function iteration**. Algorithm 1 provides a full description.

Algorithm 1: Value function iteration for job search

```
input v_0 ∈ V, an initial guess of v*
input τ, a tolerance level for error
ε ← τ + 1
k ← 0
while ε > τ do
    for w ∈ W do
        v_{k+1}(w) ← (T v_k)(w)
    end
    ε ← ‖v_k − v_{k+1}‖_∞
    k ← k + 1
end
Compute a v_k-greedy policy σ
return σ
```

While $T^k v$ never attains v^* in most cases, we can obtain a close approximation by monitoring distances between successive iterates, waiting until they become small enough. Later we will study how these distances depend on k, the number of iterations, as well as the parameters.

Listing 6 implements value function iteration for the infinite-horizon job search model, using the function for successive approximation from Listing 4.

Figure 1.10 shows a sequence of iterates $(T^k v)_k$ when $v \equiv 0$ and parameters are as given in Listing 1 (page 8). Iterates $0, 1$ and 2 are shown, in addition to a limiting function (iterate 1000). If you experiment with different initial conditions, you will see that they all converge to the same limit.

Figure 1.11 shows an approximation of v^* computed using the code in Listing 6, along with the stopping reward $w/(1-\beta)$ and the corresponding continuation value (1.27). As anticipated, the value function is the pointwise supremum of the stopping reward and the continuation value. The agent chooses to accept an offer only when that offer exceeds some value close to 43.5.

1.3.2.2 Computing the Continuation Value Directly

The technique we employed to solve the job search model in §1.3.1 follows a standard approach to dynamic programming. But for this particular problem, there is an easier

```
include("two_period_job_search.jl")
include("s_approx.jl")

" The Bellman operator. "
function T(v, model)
    (; n, w_vals, ϕ, β, c) = model
    return [max(w / (1 - β), c + β * v'ϕ) for w in w_vals]
end

" Get a v-greedy policy. "
function get_greedy(v, model)
    (; n, w_vals, ϕ, β, c) = model
    σ = w_vals ./ (1 - β) .>= c .+ β * v'ϕ  # Boolean policy vector
    return σ
end

" Solve the infinite-horizon IID job search model by VFI. "
function vfi(model=default_model)
    (; n, w_vals, ϕ, β, c) = model
    v_init = zero(model.w_vals)
    v_star = successive_approx(v -> T(v, model), v_init)
    σ_star = get_greedy(v_star, model)
    return v_star, σ_star
end
```

Listing 6: Value function iteration (`iid_job_search.jl`)

Figure 1.10: A sequence of iterates of the Bellman operator

Figure 1.11: The approximate value function for job search

way to compute the optimal policy that sidesteps calculating the value function. This section explains how.

Recall that the value function satisfies the Bellman equation

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\, c + \beta \sum_{w'} v^*(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}), \tag{1.32}$$

and that the continuation value is given by (1.27). We can use h^* to eliminate v^* from (1.32). First we insert h^* on the right hand side of (1.32) and then we replace w with w', which gives $v^*(w') = \max\{w'/(1-\beta),\, h^*\}$. Then we take mathematical expectations of both sides, multiply by β and add c to obtain

$$h^* = c + \beta \sum_{w'} \max\left\{\frac{w'}{1-\beta},\, h^*\right\} \varphi(w'). \tag{1.33}$$

To obtain the unknown value h^*, we introduce the mapping $g\colon \mathbb{R}_+ \to \mathbb{R}_+$ defined by

$$g(h) = c + \beta \sum_{w'} \max\left\{\frac{w'}{1-\beta},\, h\right\} \varphi(w'). \tag{1.34}$$

By construction, h^* solves (1.33) if and only if h^* is a fixed point of g.

Exercise 1.3.2. Show that g is a contraction map on $\mathbb{R}_+$. Conclude that h^* is the unique fixed point of g in $\mathbb{R}_+$.

Figure 1.12 shows the function g using the discrete wage offer distribution and parameters as adopted previously. The unique fixed point is h^*.

Exercise 1.3.2 implies that we can compute h^* by choosing arbitrary $h \in \mathbb{R}_+$ and iterating with g. Doing so produces a value of approximately 1086. (The associated reservation wage is $w^* = (1-\beta)h^* \approx 43.4$.) Computation of h^* using this method is much faster than value function iteration because the fixed point problem is in $\mathbb{R}_+$ rather than $\mathbb{R}_+^n$.

With h^* in hand we have solved the dynamic programming problem, since a policy σ^* is v^*-greedy if and only if it satisfies

$$\sigma^*(w) = \mathbb{1}\left\{\frac{w}{1-\beta} \geqslant h^*\right\} \qquad (w \in \mathbb{R}_+). \tag{1.35}$$

Exercise 1.3.3. As a computational exercise, compare the value function v^* com-

Figure 1.12: Computing the continuation value as the fixed point of g

puted via

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ h^*\right\}$$

with our previous result, shown in Figure 1.11. You should find them essentially identical.

1.4 Chapter Notes

Dynamic programming is often attributed to Richard Bellman (1920–1984). Both the term "dynamic programming" and the methodology were popularized by Bellman (1957). According to his autobiography, Bellman chose the name dynamic programming to avoid giving the impression that he was conducting mathematical research within RAND Corporation. His ultimate boss, Secretary of Defense Charles Wilson, apparently disliked such research (Bellman (1984)).

For treatments of dynamic programming from the perspective of economics and finance, see, for example, Sargent (1987), Stokey and Lucas (1989), Van and Dana (2003), Bäuerle and Rieder (2011), or Stachurski (2022).

The job search model was introduced by McCall (1970). The McCall model and its extensions transformed economists' way of thinking about labor markets (see, e.g., Lucas (1978b)). Influential extensions to the job search model include Burdett (1978), Jovanovic (1979), Pissarides (1979), Jovanovic (1984), Mortensen (1986),

Ljungqvist (2002) and Chetty (2008). Rogerson et al. (2005) provides a useful survey.

For elementary real analysis, the book by Bartle and Sherbert (2011) is excellent. Ok (2007) is a superb treatment of real analysis and how it is used throughout economic theory. Discussions of Banach's theorem and the Neumann series lemma can be found in Cheney (2013) and Atkinson and Han (2005). Martins-da Rocha and Vailakis (2010) provides an extension to Banach's theorem that requires only local contractivity.

Chapter 2

Operators and Fixed Points

Chapter 1 emphasized the importance of fixed point theory for dynamic programming and reviewed some fixed point results. This chapter contains a more extensive discussion of fixed point theory that ranges from analysis to algorithms. We focus on fixed point theorems that deliver global stability (as defined in §1.2.2.2), since global stability will play a key role in the optimality results in Chapters 4–8.

Some of the fixed point theory we discuss in this chapter is related to order. An investment in order-theoretic concepts will prove valuable not only for fixed point methods but also for understanding the main concepts in dynamic programming. (Chapter 8 will show that core components of dynamic programming can be expressed in terms of simple order-theoretic constructs.) We begin discussing order in §2.1 below.

2.1 Order

We begin with partial orders and then analyze fixed points of operators acting on partially ordered sets.

2.1.1 Partial Orders

Our first task is to define partially ordered sets and review some examples.

2.1.1.1 Partially Ordered Sets

A **partial order** on a nonempty set P is a relation $\preceq$ on $P \times P$ that, for any p, q, r in P, satisfies

$p \preceq p$	(reflexivity)
$p \preceq q$ and $q \preceq p$ implies $p = q$ and	(antisymmetry)
$p \preceq q$ and $q \preceq r$ implies $p \preceq r$	(transitivity)

When paired with a partial order $\preceq$, a set P (or the pair $(P, \preceq)$) is called a **partially ordered set**. For convenience, we sometimes write $p \preceq q$ as $q \succeq p$. The statement $p \preceq q \preceq r$ means $p \preceq q$ and $q \preceq r$.

Example 2.1.1. The usual order $\leqslant$ on $\mathbb{R}$ is a partial order on $\mathbb{R}$. For example, $a \leqslant b$ and $b \leqslant a$ implies $a = b$.

Exercise 2.1.1. Let P be any set and consider the relation induced by equality, so that $p \preceq q$ if and only if $p = q$. Show that this relation is a partial order on P.

Exercise 2.1.2. Let M be any set. Show that $\subset$ is a partial order on $\wp(M)$, the set of all subsets of M.

A partial order $\preceq$ on P is called a **total order** if, for all $p, q \in P$, either $p \preceq q$ or $q \preceq p$.

Example 2.1.2. The usual order $\leqslant$ on $\mathbb{R}$ is a total order, as is the same order on $\mathbb{N}$.

Exercise 2.1.3. Is the partial order defined in Exercise 2.1.2 a total order? Either prove that it is or provide a counterexample.

A subset B of a partially ordered set $(P, \preceq)$ is called **increasing** if $x \in B$, $y \in P$ and $x \preceq y$ implies $y \in B$. The set B is called **decreasing** if $x \in B$, $y \in P$ and $y \preceq x$ implies $y \in B$.

Exercise 2.1.4. Describe the set of increasing sets in $(\mathbb{R}, \leqslant)$.

2.1.1.2 Pointwise Orders

Most partial orders that we care about in this book are *pointwise* orders. All such orders are special cases of the following example.

Figure 2.1: Pointwise we have $u \leqslant v$ and $u \ll v$ but not $w \leqslant v$

Example 2.1.3 (Pointwise order over functions)**.** Let X be any set. For f, g in $\mathbb{R}^{\mathsf{X}}$, we write $f \leqslant g$ if $f(x) \leqslant g(x)$ for all $x \in \mathsf{X}$. The relation $\leqslant$ on $\mathbb{R}^{\mathsf{X}}$ is a partial order called the **pointwise order** on $\mathbb{R}^{\mathsf{X}}$. For example, looking at Figure 1.8 on page 30, we can see that $f \leqslant f \vee g$ and $g \leqslant f \vee g$. This makes sense, since $f \vee g$ is the pointwise maximum of the two functions.

Example 2.1.4. Let X be any set. For f, g in $\mathbb{R}^{\mathsf{X}}$, we write $f \ll g$ if $f(x) < g(x)$ for all $x \in \mathsf{X}$. The relation $\ll$ is *not* a partial order on $\mathbb{R}^{\mathsf{X}}$, since reflexivity fails.

The preceding examples extend immediately to vectors, since vectors are just real-valued functions under the identification in Lemma 1.2.4 (page 28). In particular, for vectors $u = (u_1, \dots, u_n)$ and $v = (v_1, \dots, v_n)$ in $\mathbb{R}^n$, we write

- $u \leqslant v$ if $u_i \leqslant v_i$ for all $i \in [n]$ and
- $u \ll v$ if $u_i < v_i$ for all $i \in [n]$.

Statements $u \geqslant v$ and $u \gg v$ are defined analogously. Figure 2.1 illustrates. As above, the relation $\leqslant$ is called the **pointwise order** on $\mathbb{R}^n$ (while $\ll$ is not a partial order).

In Figure 2.1 we can also see that $\leqslant$ is not a total order on $\mathbb{R}^n$. For example, neither $v \leqslant w$ nor $w \leqslant v$, since $w_1 > v_1$ but $w_2 < v_2$.

Exercise 2.1.5. Limits in $\mathbb{R}$ preserve weak inequalities. Use this property to prove that the same is true in $\mathbb{R}^n$. In particular, show that, for vectors $a, b \in \mathbb{R}^n$ and sequence (u_k) in $\mathbb{R}^n$ with $a \leqslant u_k \leqslant b$ for all $k \in \mathbb{N}$ and $u_k \to u \in \mathbb{R}^n$, we have $a \leqslant u \leqslant b$.

Example 2.1.5 (Pointwise order over matrices)**.** Analogous to vectors, for $n \times k$ matrices $A = (a_{ij})$ and $B = (b_{ij})$, we write

- $A \leqslant B$ if $a_{ij} \leqslant b_{ij}$ for all i, j.
- $A \ll B$ if $a_{ij} < b_{ij}$ for all i, j.

The relation $\leqslant$ is a partial order on $\mathbb{M}^{n\times k}$, the set of real-valued $n \times k$ matrices. As for vectors, we call this the **pointwise order**.

Exercise 2.1.6. Explain why the pointwise order introduced in Example 2.1.5 is also a special case of the order in Example 2.1.3.

Given a partially ordered set $(P, \preceq)$ and two elements a, b of P, the **order interval** $[a, b]$ is defined as all $p \in P$ such that $a \preceq p \preceq b$. (If $a \preceq b$ fails, the order interval is empty.)

Exercise 2.1.7. Let $C[0, 1]$ be the set of continuous functions on $[0, 1]$, partially ordered by the pointwise order $\leqslant$. Let f_i, g_i be elements of $C[0, 1]$ for $i = 1, 2$. Show that the intersection of the two order intervals $[f_1, f_2]$ and $[g_1, g_2]$ is an order interval in $C[0, 1]$.

Exercise 2.1.8. Prove the next two facts:

(i) If B is $m \times k$ and $B \geqslant 0$, then $|Bu| \leqslant B|u|$ for all $k \times 1$ column vectors u.

(ii) If A is $n \times n$ with $A \geqslant 0$ and (u_k) is a sequence in $\mathbb{R}^n$ satisfying $u_{k+1} \leqslant Au_k$ for all $k \geqslant 0$, then $u_k \leqslant A^k u_0$.

Exercise 2.1.9. Let A be $n \times k$ and let u and v be k-vectors. Prove that $A \gg 0$, $u \leqslant v$ and $u \neq v$ implies $Au \ll Av$.

2.1.1.3 Inequalities and Identities

The following inequalities and identities hold.

Lemma 2.1.1. *If X is finite and $f, g, h \in \mathbb{R}^{\mathsf{X}}$, then the following statements are true:*

(i) $|f + g| \leqslant |f| + |g|$.

(ii) $(f \wedge g) + h = (f + h) \wedge (g + h)$ *and* $(f \vee g) + h = (f + h) \vee (g + h)$.

(iii) $(f \vee g) \wedge h = (f \wedge h) \vee (g \wedge h)$ *and* $(f \wedge g) \vee h = (f \vee h) \wedge (g \vee h)$.

(iv) $|f \wedge h - g \wedge h| \leqslant |f - g|$.

(v) $|f \vee h - g \vee h| \leqslant |f - g|$.

These results follow immediately from proofs of analogous results when $f, g, h \in \mathbb{R}$. For example, by the usual triangle inequality for scalars, we have $|f(x) + g(x)| \leqslant |f(x)| + |g(x)|$ for all $x \in \mathsf{X}$. This is equivalent to the statement $|f + g| \leqslant |f + g|$ in (i). Similarly, inequality (v) follows directly from a corresponding scalar inequality that was already proved in Exercise 1.3.1, page 33.

A complete proof of lemma 2.1.1 can be found in Theorem 30.1 of Aliprantis and Burkinshaw (1998).

It is also true that, if $f, g, h \in \mathbb{R}_+^{\mathsf{X}}$, then

$$(f + g) \wedge h \leqslant (f \wedge h) + (g \wedge h). \tag{2.1}$$

Exercise 2.1.10. Prove: If $a, b, c \in \mathbb{R}_+$, then $|a \wedge c - b \wedge c| \leqslant |a - b| \wedge c$.

2.1.1.4 Least and Greatest Elements

In dynamic programming, our aim is to maximize lifetime value (or minimize lifetime cost), which is a function over the state space, rather than a number. Thus, the objective takes values in a partially ordered set (a set of functions over the state space). This leads us to consider "least" and "greatest" elements, rather than traditional maxima and minima (definitions of which can be found in Appendix A).

In particular, given a partially ordered set $(P, \preceq)$ and $A \subset P$, we say that

- $\ell \in P$ is a **least element** of A if $\ell \in A$ and $a \succeq \ell$ for all $a \in A$, and
- $g \in P$ is a **greatest element** of A if $g \in A$ and $a \preceq g$ for all $a \in A$.

Exercise 2.1.11. Let P be any partially ordered set and fix $A \subset P$. Prove that A has at most one greatest element and at most one least element.

Example 2.1.6. Let X be a nonempty set and V be a subset of $\mathbb{R}^{\mathsf{X}}$. We consider V as a partially ordered set, ordered by the pointwise order $\leqslant$. Let $\{v_\sigma\} := \{v_\sigma\}_{\sigma \in \Sigma}$ be a finite collection of functions in V. Let $\vee_\sigma v_\sigma$ be the pointwise maximum (as defined in §1.2.4.3). Clearly, if $\vee_\sigma v_\sigma \in \{v_\sigma\}$, then $\vee_\sigma v_\sigma$ is the greatest element of $\{v_\sigma\}$.

Figure 2.2: v^* is the upper envelope of $\{v_\sigma\}_{\sigma\in\Sigma}$

Exercise 2.1.12. Adopt the setting of Example 2.1.6 and prove the following: If $\{v_\sigma\}$ has a greatest element, then that greatest element is $\vee_\sigma v_\sigma$.

Figure 2.2 helps to illustrate Example 2.1.6. The function $\vee_\sigma v_\sigma$ is denoted by v^*. In this case, $\{v_\sigma\}$ has no greatest element, since neither $v_{\sigma'} \leqslant v_{\sigma''}$ nor $v_{\sigma''} \leqslant v_{\sigma'}$.

Exercise 2.1.13. Let M be a nonempty set and let $\wp(M)$ be the set of all subsets of M, partially ordered by $\subset$. Let $\{A_i\} := \{A_i\}_{i\in I}$ be a subset of $\wp(M)$, so that $A_i \in \wp(M)$ for each i in arbitrary (nonempty) index set I. Let $S := \bigcup_i A_i$. Then S is the greatest element of $\{A_i\}$ if and only if $S \in \{A_i\}$.

Example 2.1.7. Adopt the setting of Exercise 2.1.13 and suppose that $\{A_i\}$ is the set of bounded subsets of $\mathbb{R}^n$. Since the union of all such sets is $\mathbb{R}^n$ (which is not bounded), $\{A_i\}$ has no greatest element. Indeed, if G is the greatest element of $\{A_i\}$, then G contains every bounded subset of $\mathbb{R}^n$. But then G is not bounded. Contradiction.

If A is totally ordered, then a greatest element g of A is also called a **maximum** of A, while a least element ℓ of A is also called a **minimum**. See Appendix A for more about maxima and minima.

2.1.2 Order-Preserving Maps

In this section we introduce order-preserving maps, which occur throughout the theory of dynamic programming. We also discuss a well-known condition for contractivity that requires the order preserving property.

2.1.2.1 Definition

Given two partially ordered sets $(P, \preceq)$ and $(Q, \trianglelefteq)$, a map T from P to Q is called **order-preserving** if, given $p, p' \in P$, we have

$$p \preceq p' \quad \Longrightarrow \quad Tp \trianglelefteq Tp'. \tag{2.2}$$

T is called **order-reversing** if, instead,

$$p \preceq p' \quad \Longrightarrow \quad Tp' \trianglelefteq Tp. \tag{2.3}$$

Example 2.1.8. Let $\leqslant$ be the pointwise order on $\mathbb{R}^n$. If A is $n \times n$ with $A \geqslant 0$, then $T \colon \mathbb{R}^n \to \mathbb{R}^n$ defined by $Tu = Au + b$ is order preserving on $\mathbb{R}^n$, since $u \leqslant v$ implies $v - u \geqslant 0$, and hence $A(v - u) \geqslant 0$. But then $Au \leqslant Av$ and hence $Tu \leqslant Tv$.

Example 2.1.9. Given $a \leqslant b$ in $\mathbb{R}$, let $C[a, b]$ be all continuous functions from $[a, b]$ to $\mathbb{R}$ and let $\leqslant$ be the pointwise order on $C[a, b]$. Let

$$I(f) := \int_a^b f(x)dx \qquad (f \in C[a, b]).$$

Since $f \leqslant g$ implies $\int_a^b f(x)dx \leqslant \int_a^b g(x)dx$, the map I is order-preserving on $C[a, b]$.

Exercise 2.1.14. Let $(P, \preceq)$ be a partially ordered set and let A be an order-preserving self-map on P. Prove that A^k is order-preserving on P for any $k \in \mathbb{N}$.

Exercise 2.1.15. Let A be $n \times k$ with $A \geqslant 0$. Show that the map $u \mapsto Au$ is order-preserving on $\mathbb{R}^k$ under the pointwise order.

Exercise 2.1.16. Let A and B be $n \times n$ with $0 \leqslant A \leqslant B$. Prove that $A^k \leqslant B^k$ for all $k \in \mathbb{N}$ and, in addition, that $r(A) \leqslant r(B)$.

2.1.2.2 Blackwell's Condition

For an order-preserving operator, the following condition often simplifies the problem of testing contractivity. In the statement of the lemma, U is a subset of $\mathbb{R}^{\mathsf{X}}$ and X is finite. Also, U has the property that $u \in U$ and $c \in \mathbb{R}_+$ implies $u + c \in U$. (Here and below $c := c\mathbb{1}_{\mathsf{X}}$ when treated as a function.)

Lemma 2.1.2. *If T is an order preserving self-map on U and there exists a constant $\beta \in (0, 1)$ such that*

$$T(u + c) \leqslant Tu + \beta c \quad \textit{for all } u \in U \textit{ and } c \in \mathbb{R}_+, \tag{2.4}$$

then T is a contraction of modulus β on U with respect to the supremum norm.

Proof. Let U, T have the stated properties and fix $u, v \in U$. We have

$$Tu = T(v + u - v) \leqslant T(v + \|u - v\|_\infty) \leqslant Tv + \beta\|u - v\|_\infty.$$

Rearranging gives $Tu - Tv \leqslant \beta\|u - v\|_\infty$. Reversing the roles of u and v proves the claim. □

2.1.2.3 Real-Valued Functions

Regarding the definitions in (2.2)–(2.3), when $(Q, \trianglelefteq) = (\mathbb{R}, \leqslant)$, it is common to say "increasing" instead of order-preserving, and "decreasing" instead of order-reversing. We adopt this terminology. In particular, given partially ordered set $(P, \preceq)$, we call $h \in \mathbb{R}^P$

- **increasing** if $p \preceq p'$ implies $h(p) \leqslant h(p')$ and
- **decreasing** if $p \preceq p'$ implies $h(p) \geqslant h(p')$.

We use the symbol $i\mathbb{R}^P$ for the set of increasing functions in $\mathbb{R}^P$.[1]

Exercise 2.1.17. Prove: If P is any partially ordered set and $f, g \in i\mathbb{R}^P$, then

(i) $\alpha f + \beta g \in i\mathbb{R}^P$ whenever $\alpha, \beta \geqslant 0$.

(ii) $f \vee g \in i\mathbb{R}^P$ and $f \wedge g \in i\mathbb{R}^P$.

[1] Why not just use "increasing" and "decreasing" in all cases, rather than adopting the fancier terminology "order-preserving" and "order-reversing"? A short answer is that, for a general partial order, the concepts of order-preserving and order-reversing can be very different from usual notions of increasing and decreasing functions.

Exercise 2.1.18. Given finite P, show that $i\mathbb{R}^P$ is closed in $\mathbb{R}^P$.

Exercise 2.1.19. Let X be a random variable mapping Ω to finite X. Define $\ell\colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}$ by $\ell h = \mathbb{E}h(X)$. Show that ℓ is increasing when $\mathbb{R}^{\mathsf{X}}$ has the pointwise order.

As usual, if $h\colon P \to Q$ and $P, Q \subset \mathbb{R}$, then we will call h

- **strictly increasing** if $x < y$ implies $h(x) < h(y)$, and
- **strictly decreasing** if $x < y$ implies $h(x) > h(y)$.

2.1.3 Parametric Monotonicity

We are often interested in whether a change in a parameter shifts an outcome up or down. For example, a parameter might appear in a central bank decision rule for pegging an interest rate, and we want to know whether increasing that parameter will increase steady state inflation. By providing sufficient conditions for monotone shifts in fixed points, results in this section can help answer such questions.

Let $(P, \preceq)$ be a partially ordered set. Given two self-maps S and T on a set P, we write $S \preceq T$ if $Su \preceq Tu$ for every $u \in P$ and say that T **dominates** S on P.

Example 2.1.10. Let $P = \mathbb{R}^n_+$ with the pointwise order on vectors, let $Sx = Ax + b$ and $Tx = Bx + b$, where $b \in \mathbb{R}^n$ and A and B are $n \times n$. If $A \leqslant B$, then, for any $x \in \mathbb{R}^n_+$, we have $Ax \leqslant Bx$. Hence $Sx \leqslant Tx$ and T dominates S on P.

Exercise 2.1.20. Let $(P, \preceq)$ be a partially ordered set and let S and T be order-preserving self-maps such that $S \preceq T$. Show that $S^k \preceq T^k$ holds for all $k \in \mathbb{N}$.

Exercise 2.1.21. Let $(P, \preceq)$ be a partially ordered set, let $\mathcal{S}$ be the set of all self-maps on P and, as above, write $S \preceq T$ if T dominates S on P. Show that $\preceq$ is a partial order on $\mathcal{S}$.

One might assume that, in a setting where T dominates S, the fixed points of T will be larger. This can hold, as in Figure 2.3, but it can also fail, as in Figure 2.4. One difference between these two situations is that in Figure 2.3 the map T is globally stable. This leads us to our next result.

Figure 2.3: Ordered fixed points when global stability holds

Figure 2.4: Reverse-ordered fixed points when global stability fails

Proposition 2.1.3. *Let S and T be self-maps on $M \subset \mathbb{R}^n$ and let $\leqslant$ be the pointwise order. If T dominates S on M and, in addition, T is order-preserving and globally stable on M, then its unique fixed point dominates any fixed point of S.*

Proof of Proposition 2.1.3. Assume the conditions of the proposition and let u_T be the unique fixed point of T. Let u_S be any fixed point of S. Since $S \leqslant T$, we have $u_S = Su_S \leqslant Tu_S$. Applying T to both sides of this inequality and using the order-preserving property of T and transitivity of $\leqslant$ gives $u_S \leqslant T^2 u_S$. Continuing in this fashion yields $u_S \leqslant T^k u_S$ for all $k \in \mathbb{N}$. Taking the limit in k and using the fact that $\leqslant$ is closed under limits gives $u_S \leqslant u_T$. □

As an application of Proposition 2.1.3, consider again the Solow–Swan growth model $k_{t+1} = g(k_t) := sf(k_t) + (1-\delta)k_t$. We saw in §1.2.3.2 that if $f(k) = Ak^\alpha$ where $A > 0$ and $\alpha \in (0,1)$, then g is globally stable on $M := (0,\infty)$. Clearly $k \mapsto g(k)$ is order-preserving on M. If we now increase, say, the savings rate s, then g will be shifted up everywhere, implying, via Proposition 2.1.3, that the fixed point will also rise. Exercise 2.1.22 asks you to step through the details.

Exercise 2.1.22. Let $g(k) = sAk^\alpha + (1-\delta)k$ where all parameters are strictly positive, $\alpha \in (0,1)$ and $\delta \leqslant 1$. Let $k^*(s, A, \alpha, \delta)$ be the unique fixed point of g in M. Without using the expression we derived for k^* previously (Exercise 1.2.24), show that

(i) $k^*(s, A, \alpha, \delta)$ is increasing in s and A.

(ii) $k^*(s, A, \alpha, \delta)$ is decreasing in δ.

Figure 2.5 helps illustrate the results of Exercise 2.1.22. The top left sub-figure shows a baseline parameterization, with $A = 2.0$, $s = \alpha = 0.3$ and $\delta = 0.4$. The other sub-figures show how the steady state changes as parameters deviate from that baseline.

Exercise 2.1.23. In (1.34) on page 39, we defined a map g such that the optimal continuation value h^* is a fixed point. Using this construction, prove that h^* is increasing in β.

Figure 2.6 gives an illustration of the result in Exercise 2.1.23. Here an increase in β leads to a larger continuation value. This seems reasonable, since larger β indicates more concern about outcomes in future periods.

While the preceding examples of parametric monotonicity are all one-dimensional, we will soon see that Proposition 2.1.3 can also be applied in high-dimensional settings.

Figure 2.5: Parametric monotonicity for the Solow-Swan model

Figure 2.6: Parametric monotonicity in β for the continuation value

2.2 Stability and Algorithms

We are also interested in how to compute fixed points. In this section we discuss alternative methods for computing fixed points and their rates of convergence. We also present some useful results on stability theory.

2.2.1 Conjugate Maps

Next we treat an extremely useful technique for simplifying analysis of stability and fixed points. This method will have multiple applications in what follows.

Suppose we are concerned with the dynamics induced by an operator T mapping $\mathbb{R}^n$ into itself. For example, we might want to know if a unique fixed point of T exists, or if iterates of T always converge to a fixed point. One approach would be to apply fixed point theory to T.

Sometimes, however, there is an easier approach: transform T into a "simpler" operator $\hat{T}$ and then study the fixed point properties of $\hat{T}$. Of course, for this idea to work, we need to be sure that any properties we discover about $\hat{T}$ can be translated back to properties of T, the operator that we are actually interested in.

This section explains a notion of topological conjugacy that originated in the field of dynamical systems theory. Later we will apply this approach to operators that arise in recursive preference and dynamic programming contexts.

To explain the idea, let U and $\hat{U}$ be two subsets of $\mathbb{R}^n$. A function Φ from U to $\hat{U}$ is called a **homeomorphism** if it is continuous, a bijection, and its inverse Φ^{-1} is also continuous.

Example 2.2.1. The map $\Phi u = \ln u$ from $(0, \infty)$ to $\mathbb{R}$ is a homeomorphism, with continuous inverse $\Phi^{-1} y = \exp(y)$.

Example 2.2.2. Let Φ be an $n \times n$ matrix. We can regard Φ as a map sending column vector u into column vector Φu. This map is a homeomorphism from $\mathbb{R}^n$ to itself if and only if Φ is nonsingular.

A **dynamical system** is a pair (U, T), where U is a subset of $\mathbb{R}^n$ and T is a self-map on U. Two dynamical systems (U, T) and $(\hat{U}, \hat{T})$ are said to be

- **conjugate** under Φ if Φ is a bijection from U into $\hat{U}$ such that $T = \Phi^{-1} \circ \hat{T} \circ \Phi$ on U, and

- **topologically conjugate** under Φ if (U, T) and $(\hat{U}, \hat{T})$ are conjugate under Φ and, in addition, Φ is a homeomorphism.

Example 2.2.3. Let A be an $n \times n$ **diagonalizable matrix**, meaning that there exists a diagonal matrix D and a nonsingular matrix P such that $A = P^{-1}DP$. (The matrices D and P can be complex-valued.) We regard A as a self-map on $\mathbb{R}^n$ and D and a self-map on $\mathbb{C}^n$, the set of complex n-vectors The identity $A = P^{-1}DP$ implies that the dynamical systems $(A, \mathbb{R}^n)$ and $(D, \mathbb{C}^n)$ are conjugate. (In fact they are topologically conjugate.)

Conjugacy of (U, T) and $(\hat{U}, \hat{T})$ under Φ can be understood as follows: shifting a point $u \in U$ to Tu via T is equivalent to moving u into $\hat{U}$ via $\hat{u} = \Phi u$, applying $\hat{T}$, and then moving the result back using Φ^{-1}:

$$\begin{tikzcd} u \arrow[r, "T"] \arrow[d, "\Phi"] & Tu \\ \hat{u} \arrow[r, "\hat{T}"] & \hat{T}\hat{u} \arrow[u, "\Phi^{-1}"'] \end{tikzcd}$$

Exercise 2.2.1. Let $U := (0, \infty)$ and $\hat{U} := \mathbb{R}$. Let $Tu = Au^{\alpha}$, where $A > 0$ and $\alpha \in \mathbb{R}$, and let $\hat{T}\hat{u} = \ln A + \alpha\hat{u}$. Show that T and $\hat{T}$ are topologically conjugate under $\Phi := \ln$.

Exercise 2.2.2. Show that if (U, T) and $(\hat{U}, \hat{T})$ are conjugate under Φ, then $u \in U$ is a fixed point of T on U if and only if $\Phi u \in \hat{U}$ is a fixed point of $\hat{T}$ on $\hat{U}$.

Exercise 2.2.3. Consider again the setting of Exercise 2.2.2, but now suppose that (U, T) and $(\hat{U}, \hat{T})$ are topologically conjugate under Φ, Fixing $u, u^* \in U$, show that $\lim_{k\to\infty} T^k u = u^*$ if and only if $\lim_{k\to\infty} \hat{T}^k \Phi u = \Phi u^*$.

From the preceding exercises we can state the following useful result:

Proposition 2.2.1. *If (U, T) and $(\hat{U}, \hat{T})$ are topologically conjugate, then (U, T) is globally stable if and only if $(\hat{U}, \hat{T})$ is globally stable. The unique fixed points $u^* \in U$ and $\hat{u}^* \in \hat{U}$ satisfy $\hat{u}^* = \Phi u^*$.*

2.2.2 Local Stability

In §1.2.2.2 we investigated global stability, which is relatively strict. Here we introduce local stability and provide a standard sufficient condition in the case where the

map is smooth.

Let U be a subset of $\mathbb{R}^n$ and let T be a self-map on U. A fixed point u^* of T in U is called **locally stable** for the dynamical system (U, T) if there exists an open set $O \subset U$ such that $u^* \in O$ and $T^k u \to u^*$ as $k \to \infty$ for every $u \in O$. In other words, the domain of attraction for u^* contains an open neighborhood of u^*.

Example 2.2.4. Consider the self-map g on $\mathbb{R}$ defined by $g(x) = x^2$. The fixed point 1 is not stable (for example, $g^t(x) \to \infty$ for any $x > 1$). However, 0 is locally stable, because $-1 < x < 1$ implies that $g^t(x) \to 0$ as $t \to \infty$.

Exercise 2.2.4. Returning to the setting of Exercise 2.2.3, let (U, T) and $(\hat{U}, \hat{T})$ be topologically conjugate and let u^* be a fixed point of T in U. Show that u^* is locally stable for (U, T) if and only if Φu^* is locally stable for $(\hat{U}, \hat{T})$.

For an interior fixed point x^* of a smooth self-map g on an interval of $\mathbb{R}$, local stability holds whenever $|g'(x^*)| < 1$. The idea of the proof runs as follows: When $|g'(x^*)| < 1$, the first-order linear approximation

$$\hat{g}(x) := g(x^*) + g'(x^*)(x - x^*) = x^* + g'(x^*)(x - x^*)$$

is a contraction of modulus $|g'(x^*)|$ with unique fixed point x^*. Hence all trajectories of $\hat{g}$ converge to x^*. Moreover, since g and $\hat{g}$ are similar in a neighborhood of x^*, the same is true for trajectories of g starting close to x^*.

The next theorem formalizes this argument and extends it to multiple dimensions. In stating the theorem, we take T to be a self-map on U with fixed point u^* in U and assume that T is continuously differentiable on U. We recall that the **Jacobian** of T at $u \in U$ is

$$J_T(u) := \begin{pmatrix} \frac{\partial T_1}{\partial u_1}(u) & \cdots & \frac{\partial T_1}{\partial u_n}(u) \\ & \cdots & \\ \frac{\partial T_n}{\partial u_1}(u) & \cdots & \frac{\partial T_n}{\partial u_n}(u) \end{pmatrix} \quad \text{where} \quad Tu = \begin{pmatrix} T_1 u \\ \vdots \\ T_n u \end{pmatrix}.$$

We set $\hat{T}$ to be the first-order approximation to T at u^*:

$$\hat{T}u = u^* + J_T(u^*)(u - u^*) \qquad (u \in U).$$

Theorem 2.2.2 (Hartman–Grobman)**.** *If $J_T(u^*)$ is nonsingular and contains no eigenvalues on the unit circle in $\mathbb{C}$, then there exists an open neighborhood O of u^* such that (O, T) and $(O, \hat{T})$ are topologically conjugate.*

Combining this theorem with the result of Exercise 2.2.4, we see that, under the conditions of the theorem, u^* is globally stable for (O, T), and hence locally stable for

(U, T), whenever $(O, \hat{T})$ is globally stable. By the Neumann series lemma, the first-order approximation will be globally stable whenever $J_T(u^*)$ has spectral radius less than one. Thus, we have

Corollary 2.2.3. *Under the conditions of Theorem 2.2.2, the fixed point u^* is locally stable whenever $r(J_T(u^*)) < 1$.*

2.2.3 Convergence Rates

When comparing fixed point computation strategies a key issue is relative rates of convergence. Given some norm $\|\cdot\|$ on $\mathbb{R}^n$, we say that $(u_k)_{k\geqslant 0} \subset \mathbb{R}^n$ converges to $u^* \in \mathbb{R}^n$ **at rate at least** q if $q \geqslant 1$ and, for some $\beta \in (0, \infty)$ and $N \in \mathbb{N}$, we have

$$\|u_{k+1} - u^*\| \leqslant \beta \|u_k - u^*\|^q \quad \text{for all } k \geqslant N.$$

If, in addition,

$$\limsup_{k\to\infty} \frac{\|u_{k+1} - u^*\|}{\|u_k - u^*\|^q} = \beta$$

then we say that convergence occurs **at rate** q. If $q = 2$ then we say that convergence is (at least) **quadratic**. If $q = 1$ and $\beta < 1$, then we say that convergence is (at least) **linear**. Note that, in many sources, "at rate q" is expressed as **Q-order** q.

Example 2.2.5. Let T be a contraction of modulus λ on closed set $U \subset \mathbb{R}^n$. If u^* is the unique fixed point of T in U and $u_k := T^k u_0$, then (u_k) converges at least linearly to u^*, since $\|u_{k+1} - u^*\| = \|Tu_k - Tu^*\| \leqslant \lambda \|u_k - u^*\|$.

Orders of convergence are studied in the neighborhood of zero, implying that higher orders are faster. For example, suppose $\varepsilon_k := \|u_k - u^*\|$ is the size of the error and that u_k converges to u^* quadratically. If, say, $\varepsilon_k = 10^{-5}$, then $\varepsilon_{k+1} \approx \beta 10^{-10}$. Provided that β is not large, the number of accurate digits roughly doubles at each step.

Successive approximations of fixed points typically converge at a linear rate. To see this in one dimension, try the following exercise.

Exercise 2.2.5. Let $T : U \to U$ where T is twice continuously differentiable and U is an open interval in $\mathbb{R}$. Suppose that T has a fixed point $u^* \in U$ and that $u_k := T^k u_0$ converges to u^* as $k \to \infty$. Prove that the rate of convergence is linear whenever $0 < |T'u^*| < 1$. In completing the proof, you might find it helpful to use the fact that, by a second order Taylor expansion, there is a $v \in (u_k, u^*)$ such that

$$Tu_k = u^* + T'u^*(u_k - u^*) + \frac{T''v}{2}(u_k - u^*)^2. \tag{2.5}$$

(The assumption $0 < |T'u^*| < 1$ in Exercise 2.2.5 is a mild one. For example, given the convergence of successive approximation to the fixed point, we expect $|T'u^*| < 1$, since this inequality implies that u^* is locally stable.)

2.2.4 Gradient-Based Methods

While successive approximation always converges when global stability holds, faster algorithms can often be obtained by leveraging extra information, such as gradients. One important gradient-based technique is Newton's method. (As we discuss in §5.1.4.2, Newton's method relates to one of the core algorithms for solving dynamic programs .

While Newton's method is often used to solve for roots of a given function, here we use a version adapted to finding fixed points.

2.2.4.1 Newton Fixed Point Iteration

To illustrate the method, suppose first that T is a differentiable self-map on an open set $U \subset \mathbb{R}^n$ and that our aim is to find a fixed point of T. Our plan is to start with a guess u_0 of the fixed point and then update it to u_1. To do this we use the first-order approximation $\hat{T}$ of T around u_0 and solve for the fixed point of $\hat{T}$—which we can do exactly since $\hat{T}$ is linear. We take this new point as u_1 and then continue.

If T is one-dimensional then $\hat{T}u := Tu_0 + T'u_0(u_1 - u_0)$. For $n > 1$ we replace $T'u_0$ with the Jacobian of T at u_0, which we write as $J_T(u_0)$. We then solve $\hat{T}u_1 = u_1$ for u_1, which gives

$$u_1 = (I - J_T(u_0))^{-1}(Tu_0 - J_T(u_0)u_0) \qquad (I \text{ is the } n \times n \text{ identity}).$$

Figure 2.7 shows u_0 and u_1 when $n = 1$ and $Tu = 1 + u/(u+1)$ and $u_0 = 0.5$. The value u_1 is the fixed point of the first-order approximation $\hat{T}$. In this case it is closer to the fixed point of T than u_0, as desired.

Newton's (fixed point) method continues in the same manner, from u_1 to u_2 and so on, leading to the sequence of points

$$u_{k+1} = Qu_k \quad \text{where} \quad Qu := (I - J_T(u))^{-1}(Tu - J_T(u)u) \qquad k = 0, 1, \ldots \tag{2.6}$$

Notice that we do not need to write a new solver, since the successive approximation function in Listing 4 can be applied to Q defined in (2.6).

Figure 2.7: The first step of Newton's method applied to T

2.2.4.2 Rates of Convergence

Figure 2.8 shows both the Newton approximation sequence and the successive approximation sequence applied to computing the fixed point of the Solow–Swan model. We use two different initial conditions (top and bottom subfigures). For these initial conditions, both sequences converge but the Newton sequences converge faster.

This fast rate of convergence can be confirmed theoretically for the Newton scheme: under mild conditions, there exists a neighborhood of the fixed point within which the Newton iterates converge quadratically. See, for example, Theorem 5.4.1 of Atkinson and Han (2005). This fast rate of convergence is exploited in some dynamic programming algorithms (see §5.1.4.3).

2.2.4.3 Speed vs Robustness

Sometimes we can accelerate computations by exploiting a problem's special structure (e.g., differentiability, convexity, monotonicity). But we often face a trade-off between speed and robustness to details of problem specification. More robust methods impose less structure.

Relative to other algorithms, successive approximation tends to be robust but slow. We saw one illustration of the relatively slow rate of convergence in Figure 2.8. But we can also see its relatively strong robustness properties via the same example, by inspecting Figure 2.9, which compares the update rule of successive approximation

Figure 2.8: Newton's method applied to the Solow–Swan update rule

(the function g) with the update rule for Newton's method (the function Q in (2.6)). Also plotted is the dashed 45 degree line.

The parameterization for the model is the same as the top subfigure in Figure 1.7. As previously discussed, the shape of g implies global convergence of successive approximation. However, Q is well-behaved near the fixed point (i.e., very flat and hence strongly contractive) but badly behaved away from the fixed point. This illustrates that Newton's method is fast but generally less robust.

2.2.4.4 Parallelization

We have discussed rates of convergence for fixed point methods. Mathematicians and computer scientists also analyze algorithms via **worst case complexity**, which measures the number of fundamental operations (e.g., addition and multiplication of floating point numbers) when the algorithm acts on data that is least favorable for high performance. These measures are attractive because they are independent of the software and hardware platforms on which algorithms are implemented.

However, different software and hardware matter not just for absolute performance of algorithms but also for *relative* performance. For example, while a single update step in successive approximation can often be partially parallelized, the algorithm is inherently serial, in the sense that the $(k+1)$-th iterate cannot be computed

Figure 2.9: Robustness of successive approximation vs Newton's method

until iterate k is available. Moreover, because the rate of convergence is often slow, there can be many serial steps that are hard to parallelize over.

Newton's method is also serial to some degree, since we are just iterating with a different map (the operator Q in (2.6)). However, because it involves inverting matrices in potentially high dimensions, each step is computationally intensive. At the same time, since the rate of convergence is faster, we have a smaller number of steps. In this sense, the algorithm is less serial—it involves a smaller number of more expensive steps. Because it is less serial, Newton's method offers far more potential for parallelization. Thus, the speed gain associated with Newton's method can become very large when using effective parallelization.

2.3 Matrices and Operators

Many aspects of dynamic programming are most clearly framed using operator theory. In this section, we discuss linear operators and their connection to matrices. We put particular focus on nonnegative matrices and so-called positive linear operators, which arise naturally in dynamic programming applications.

2.3.1 Nonnegative Matrices

We begin by reviewing basic properties of nonnegative matrices. A key theoretical result is the Perron–Frobenius theorem.

2.3.1.1 Nonnegative Matrices and their Powers

We call a matrix A **nonnegative** and write $A \geqslant 0$ if all elements of A are nonnegative. We call A **positive**, and we write $A \gg 0$, if all elements of A are strictly positive. A nonnegative square matrix A is called **irreducible** if $\sum_{k \in \mathbb{N}} A^k \gg 0$, which is stronger than nonnegativity but weaker than positivity. An interpretation in terms of connected networks is given in Chapter 1 of Sargent and Stachurski (2022).

Let A be $n \times n$. It is not always true that the spectral radius $r(A)$ is an eigenvalue.[2] However, when $A \geqslant 0$, we have the theorem stated below. The theorem looks innocuous but turns out to be extremely useful.

Theorem 2.3.1 (Perron–Frobenius)**.** *If $A \geqslant 0$, then $r(A)$ is an eigenvalue of A with nonnegative, real-valued right and left eigenvectors. In particular, we can find a nonnegative, nonzero column vector e and a nonnegative, nonzero row vector ε such that*

$$Ae = r(A)e \quad \text{and} \quad \varepsilon A = r(A)\varepsilon. \tag{2.7}$$

If A is irreducible, then the right and left eigenvectors are everywhere positive and unique. Moreover, if A is positive, then with e and ε normalized so that $\langle \varepsilon, e \rangle = 1$, we have

$$r(A)^{-t} A^t \to e\,\varepsilon \qquad (t \to \infty). \tag{2.8}$$

The convergence in (2.8) provides a sharp characterization of large powers of A. In §2.3.2, we will apply it to a model of employment and unemployment flows.

Remark 2.3.1. Note that, in general, if v is a positive real-valued eigenvector for A, then so is αv for all $\alpha > 0$. Hence the uniqueness asserted in the Perron–Frobenius theorem is only up to positive multiples. It tells us that if e is the right eigenvector corresponding to $r(A)$ and $\hat{e}$ is another positive vector satisfying $A\hat{e} = r(A)\hat{e}$, then $\hat{e} = \alpha e$ for some $\alpha > 0$. A similar statement holds for the left eigenvalue ε.

Remark 2.3.2. The assumption that A is positive can be weakened without affecting the convergence asserted in (2.8). A complete statement and full proof of the theorem can be found in Meyer (2000).

[2] For example, if $A = \operatorname{diag}(-1, 0)$ then the eigenvalues of A are $\{-1, 0\}$. Hence $r(A) = |-1| = 1$, which is not an eigenvalue of A.

We can use the Perron–Frobenius theorem to provide bounds on the spectral radius of a nonnegative matrix. Fix $n \times n$ matrix $A = (a_{ij})$ and set

- $\mathrm{rowsum}_i(A) := \sum_j a_{ij}$ = the i-th row sum of A and
- $\mathrm{colsum}_j(A) := \sum_i a_{ij}$ = the j-th column sum of A.

Lemma 2.3.2. *If $A \geqslant 0$, then*

(i) $\min_i \mathrm{rowsum}_i(A) \leqslant r(A) \leqslant \max_i \mathrm{rowsum}_i(A)$ *and*

(ii) $\min_j \mathrm{colsum}_j(A) \leqslant r(A) \leqslant \max_j \mathrm{colsum}_j(A)$.

Exercise 2.3.1. Prove Lemma 2.3.2. (Hint: Since e and ε are nonnegative and nonzero, and since eigenvectors are defined only up to nonzero multiples, you can assume that both of these vectors sum to 1.)

2.3.1.2 Markov Matrices

Let $\mathbb{1}$ be a column vector of ones. An $n \times n$ matrix P is called a **Markov matrix** if

$$P \geqslant 0 \quad \text{and} \quad P\mathbb{1} = \mathbb{1}.$$

In other words, P is nonnegative and has unit row sums.

Exercise 2.3.2. Let P, Q be $n \times n$ Markov matrices. Prove the following facts.

(i) PQ is also a Markov matrix.

(ii) $r(P) = 1$.

(iii) There exists a row vector $\psi \in \mathbb{R}^n_+$ such that $\psi\mathbb{1} = 1$ and $\psi P = \psi$.

(iv) If P is irreducible, then the vector ψ in (iii) is everywhere positive and unique, in the sense that no other vector $\psi \in \mathbb{R}^n_+$ satisfies $\psi\mathbb{1} = 1$ and $\psi P = \psi$.

The vector ψ in part (iii) of Exercise 2.3.2 is called a **stationary distribution** for P. Such distributions play an important role in the theory of Markov chains and we discuss their interpretation and significance in §3.1.2.

Exercise 2.3.3. Given Markov matrix P and constant $\varepsilon > 0$, prove the following result: There exists no $h \in \mathbb{R}^X$ with $Ph \geqslant h + \varepsilon$.

Figure 2.10: Lake model transition dynamics

2.3.2 A Lake Model

In this section we illustrate the power of the Perron–Frobenius theorem by showing how it helps us analyze a model of employment and unemployment flows in a large population.

The model is sometimes called a "lake model" because there are two pools of workers: those who are currently employed and those who are currently unemployed but still seeking work. The flows between states are as follows:

- Workers exit the labor market at rate d.
- New workers enter the labor market at rate b.
- Employed workers separate from their jobs and become unemployed at rate α.
- Unemployed workers find jobs at rate λ.

We assume that all of these parameters lie in $(0, 1)$. New workers are initially unemployed.

The resulting rates of transition between the two pools are shown in Figure 2.10. For example, the rate of flow from employment to unemployment is $\alpha(1-d)$, which equals the fraction of employed workers who remained in the labor market and separated from their jobs.

Let e_t and u_t be the number of employed and unemployed workers at time t respectively. The total population (of workers) is $n_t := e_t + u_t$. In view of the rates stated

Figure 2.11: Time series for e_t, u_t and n_t, (`lake_2.jl`)

above, the number of unemployed workers evolves according to

$$u_{t+1} = (1-d)\alpha e_t + (1-d)(1-\lambda)u_t + bn_t.$$

These three terms on the right correspond to the newly unemployed (due to separation), the unemployed who failed to find jobs last period, and new entrants into the labor force. The number of employed workers evolves according to

$$e_{t+1} = (1-d)(1-\alpha)e_t + (1-d)\lambda u_t.$$

Evolution of the time series for u_t, e_t and n_t is illustrated in Figure 2.11. The parameters were set to $\alpha = 0.01$, $\lambda = 0.1$, $d = 0.02$, and $b = 0.025$. The initial population of unemployed and employed workers was set to $u_0 = 0.6$ and $e_0 = 1.2$ respectively. The series grow over the long run due to net population growth.

Can we say more about the dynamics of this system? For example, what long run unemployment rate should we expect? Also, do long run outcomes depend heavily on the initial conditions u_0 and e_0? Or are there some general statements we can make, which hold regardless of the initial state.

To begin to address these questions, we first organize the linear system for (e_t)

and (u_t) by setting

$$x_t := \begin{pmatrix} u_t \\ e_t \end{pmatrix} \quad \text{and} \quad A := \begin{pmatrix} (1-d)(1-\lambda)+b & (1-d)\alpha+b \\ (1-d)\lambda & (1-d)(1-\alpha) \end{pmatrix}. \tag{2.9}$$

With these definitions, we can write the dynamics as $x_{t+1} = Ax_t$. As a result, $x_t = A^t x_0$, where $x_0 = (u_0 \; e_0)^\top$.

The overall growth rate of the total labor force is $g = b - d$, in the sense that $n_{t+1} = (1+g)n_t$ for all t.

Exercise 2.3.4. Confirm this claim by using the equation $x_{t+1} = Ax_t$.

Exercise 2.3.5. Prove that $r(A) = 1+g$. [Hint: Use one of the results in §2.3.1.1.]

Exercise 2.3.6. By the Perron-Frobenius theorem, $1+g$ is an eigenvalue (in fact the dominant eigenvalue) of A. Show that $\mathbb{1}^\top := (1 \; 1)$ is a left eigenvector corresponding to this eigenvalue.

Exercise 2.3.7. Prove that the unique right eigenvector $\bar{x}$ satisfying $A\bar{x} = r(A)\bar{x}$ and $\mathbb{1}^\top \bar{x} = 1$ is given by

$$\bar{x} := \begin{pmatrix} \bar{u} \\ \bar{e} \end{pmatrix} \quad \text{with} \quad \bar{u} := \frac{1+g-(1-d)(1-\alpha)}{1+g-(1-d)(1-\alpha)+(1-d)\lambda} \tag{2.10}$$

and $\bar{e} := 1 - \bar{u}$.

In the language of Perron–Frobenius theory, the right eigenvector $\bar{x}$ is sometimes called the **dominant eigenvector**, since it corresponds to the dominant (i.e., largest) eigenvalue $r(A)$. It is also true that this eigenvector plays an important role in determining long run outcomes. In the remainder of this section we illustrate this fact.

To begin, recall that $\alpha\bar{x}$ is also a right eigenvector corresponding to the eigenvalue $r(A)$ when $\alpha > 0$. The set $D := \{x \in \mathbb{R}^2 : x = \alpha\bar{x} \text{ for some } \alpha > 0\}$ is shown as a dashed black line in Figure 2.12. The figure also shows two time paths, each of the form $(x_t)_{t\geqslant 0} = (A^t x_0)_{t\geqslant 0}$, generated from two different initial conditions. In both cases, we see that both paths converge to D over time. The figure suggests that all paths share strong similarities in the long run, with those similarities determined by the dominant eigenvector $\bar{x}$.

Figure 2.12: Time paths $x_t = A^t x_0$ for two choices of x_0 (`lake_1.jl`)

To see why this is so, we return (2.8) from to the Perron–Frobenius theorem, which tells us, since $A \gg 0$, we have

$$A^t \approx r(A)^t \cdot \bar{x}\mathbb{1}^\top = (1+g)^t \begin{pmatrix} \bar{u} & \bar{u} \\ \bar{e} & \bar{e} \end{pmatrix} \quad \text{for large } t.$$

As a result, for any initial condition $x_0 = (u_0 \; e_0)^\top$, we have

$$A^t x_0 \approx (1+g)^t \begin{pmatrix} \bar{u} & \bar{u} \\ \bar{e} & \bar{e} \end{pmatrix} \begin{pmatrix} u_0 \\ e_0 \end{pmatrix} = (1+g)^t (u_0 + e_0) \begin{pmatrix} \bar{u} \\ \bar{e} \end{pmatrix} = n_t \bar{x},$$

where $n_t = (1+g)^t n_0$ and $n_0 = u_0 + e_0$. This says that, regardless of the initial condition, the state x_t scales along $\bar{x}$ at the rate of population growth. This is precisely what we saw in Figure 2.12.

We can give an additional interpretation to the components $\bar{u}$ and $\bar{e}$ of $\bar{x}$. Since n_t is the size of the workforce at time t, the rate of unemployment is u_t/n_t. As just shown, for large t this is close to $(n_t \bar{u})/n_t = \bar{u}$. Hence $\bar{u}$ is the long term rate of unemployment along the stable growth path. Similarly, the other component $\bar{e}$ of the dominant eigenvector is the long run employment rate for this economy.

In summary, the dominant eigenvector provides with both the long-run rate of unemployment and the stable growth path, to which all trajectories with positive initial

conditions converge over time.

Remark 2.3.3. A more careful analysis of this problem would require us to think carefully about how the underlying rates α, λ, b and d are determined. For the hiring rate λ, we could use the job search model to fix the rate at which workers are matched to jobs. In particular, with w^* as the reservation wage, we could set

$$\lambda = \mathbb{P}\{w_t \geqslant w^*\} = \sum_{w \in \mathsf{W}} \varphi(\mathbb{1}\{w \geqslant w^*\}).$$

Doing so would allow us to study the crucial rate λ in terms of fundamental primitives, such as unemployment compensation and impatience of individual agents.

2.3.3 Linear Operators

There are two ways to think about a matrix. In one definition, an $n \times k$ matrix A is an $n \times k$ array of (real) numbers. In the second, A is a linear operator from $\mathbb{R}^k$ to $\mathbb{R}^n$ that takes a vector $u \in \mathbb{R}^k$ and sends it to Au in $\mathbb{R}^n$. Let's clarify these ideas in a setting where $n = k$. We will see that while the matrix representation is important, the linear operator representation is more fundamental and more general.

2.3.3.1 Matrices vs Linear Operators

A **linear operator** on $\mathbb{R}^n$ is a map L from $\mathbb{R}^n$ to $\mathbb{R}^n$ such that

$$L(\alpha u + \beta v) = \alpha L u + \beta L v \quad \text{for all } u, v \in \mathbb{R}^n \text{ and } \alpha, \beta \in \mathbb{R}. \tag{2.11}$$

(We write Lu instead of $L(u)$, etc.) For example, if A is an $n \times n$ matrix, then the map from u to Au defines a linear operator, since the rules of matrix algebra yield $A(\alpha u + \beta v) = \alpha A u + \beta A v$.

We just showed that each matrix can be regarded as a linear operator. The following fundamental result states that the converse is also true. A proof can be found in Kreyszig (1978) and many other sources.

Theorem 2.3.3. *If L is a linear operator on $\mathbb{R}^n$, then there exists an $n \times n$ matrix $A = (a_{ij})$ such that $Lu = Au$ for all $u \in \mathbb{R}^n$.*

Why introduce linear operators if they are essentially the same as matrices? One reason is that, while a one-to-one correspondence between linear operators and matrices holds in $\mathbb{R}^n$, the concept of linear operators is far more general. Linear operators

can be defined over many different kinds of sets, provided that the elements of the sets have vector-like properties. This is related to the point that we made about function spaces in Remark 1.2.3 on page 28.

Another reason is computational: the matrix representation of a linear operator can be tedious to construct and difficult to instantiate in memory in large problems. We illustrate this point in §2.3.3.3 below.

2.3.3.2 Linear Operators on Function Space

The definition of linear operators on $\mathbb{R}^n$ extends naturally to linear operators on $\mathbb{R}^{\mathsf{X}}$ when $\mathsf{X} = \{x_1, \ldots, x_n\}$: A **linear operator** on $\mathbb{R}^{\mathsf{X}}$ is a map L from $\mathbb{R}^{\mathsf{X}}$ to itself such that, for all $u, v \in \mathbb{R}^{\mathsf{X}}$ and $\alpha, \beta \in \mathbb{R}$, we have $L(\alpha u + \beta v) = \alpha Lu + \beta Lv$. In what follows,

$$\mathbb{L}(\mathsf{X}) := \text{ the set of all linear operators on } \mathbb{R}^{\mathsf{X}}.$$

Let L be a function from $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}$. This function induces an operator L from $\mathbb{R}^{\mathsf{X}}$ to itself via

$$(Lu)(x) = \sum_{x' \in \mathsf{X}} L(x, x')u(x') \qquad (x \in \mathsf{X},\ u \in \mathbb{R}^{\mathsf{X}}). \tag{2.12}$$

We use the same symbol L on both sides of the equals sign because both of these symbols represent essentially the same object, as clarified below.

The function L on the right-hand side of (2.12) is sometimes called the "kernel" of the operator L. However, we will call it a matrix in what follows, since $L(x, x') = L(x_i, x_j)$ is just a square $n \times n$ array of real numbers. When more precision is required, we will call it the **matrix representation** of L.

In essence, (2.12) is just matrix multiplication: we are extracting row x of the matrix product Lu.

Exercise 2.3.8. Prove that L is a linear operator (i.e., an element of $\mathbb{L}(\mathsf{X})$).

The eigenvalues and eigenvectors of the linear operator L are defined as the eigenvalues and eigenvectors of its matrix representation. The spectral radius $r(L)$ of L is defined analogously.

We used the same symbol for the operator in (2.12) and its matrix representation because these two objects are in one-to-one correspondence. Thus, every $L \in \mathbb{L}(\mathsf{X})$ can be expressed in the form of (2.12) for a suitable choice of matrix. Readers who are comfortable with these claim can skip ahead to §2.3.3.3. The next lemma provides more details.

Lemma 2.3.4. *When* $\mathsf{X} = \{x_1, \ldots, x_n\}$*, the following sets are in one-to-one correspondence:*

(a) *The set of all* $n \times n$ *real matrices.*

(b) *The set of all linear operators on* $\mathbb{R}^n$*.*

(c) *The set* $\mathbb{L}(\mathsf{X})$ *of linear operators on* $\mathbb{R}^{\mathsf{X}}$*.*

(d) *The set of all functions from* $\mathsf{X} \times \mathsf{X}$ *to* $\mathbb{R}$*.*

Lemma 2.3.4 needs no formal proof. Theorem 2.3.3 already tells us that (a) and (b) are in one-to-one correspondence. Also, (b) and (c) are in one-to-one correspondence because each $L \in \mathbb{L}(\mathsf{X})$ can be identified with a linear operator $u \mapsto Lu$ on $\mathbb{R}^n$ by pairing $u, Lu \in \mathbb{R}^{\mathsf{X}}$ with their vector representations in $\mathbb{R}^n$ (see §1.2.4.1). Finally, (d) and (a) are in one-to-one correspondence under the identification $L(x_i, x_j) \leftrightarrow L_{ij}$.

2.3.3.3 Computational Issues

At the end of §2.3.3.1 we claimed that working with linear operators holds certain computational advantages vis-à-vis working with matrices. This section fills in some details (Readers who prefer not to think about computational issues at this point can skip ahead to §2.3.3.4.)

To illustrate the main idea, consider a setting where the state space X takes the form $\mathsf{X} = \mathsf{Y} \times \mathsf{Z}$ with $|\mathsf{Y}| = j$ and $|\mathsf{Z}| = k$. A typical element of X is $x = (y, z)$. As we shall see, this kind of setting arises naturally in dynamic programming.

Let Q be a map from $\mathsf{Z} \times \mathsf{Z}$ to $\mathbb{R}$ (i.e., a $k \times k$ matrix) and consider the operator sending $u \in \mathbb{R}^{\mathsf{X}}$ to $Lu \in \mathbb{R}^{\mathsf{X}}$ according to the rule

$$(Lu)(x) = (Lu)(y, z) = \sum_{z' \in \mathsf{Z}} u(y, z')Q(z, z') \tag{2.13}$$

Exercise 2.3.9. Prove that $L \in \mathbb{L}(\mathsf{X})$.

Since L is a linear operator on $\mathbb{R}^{\mathsf{X}}$, we know that L can be represented as an $n \times n$ matrix $L(x_i, x_j) = L_{ij}$, where $n = |\mathsf{X}| = j \times k$. To construct this matrix, we first need to "flatten" $\mathsf{Y} \times \mathsf{Z}$ into a set $\mathsf{X} = \{x_1, \ldots, x_n\}$ with a single index. There are two natural ways to do this. Considering $\mathsf{Y} \times \mathsf{Z}$ as a two-dimensional array with typical element (y_i, z_j), we can stack the k columns vertically into one long column. Alternatively, we

can concatenate all j rows into one long row. The first arrangement is called **column-major ordering** and is adopted as standard by languages such as Julia and Fortran. The second is called **row-major ordering** and is adopted as standard by languages such as Python and C. Either way we obtain a set of elements indexed by $1, \ldots, n$.

After picking one of these conventions, Lemma 2.3.4 assures us we can construct a uniquely defined $n \times n$ matrix that represents L. Once we determine how to construct this matrix, we can instantiate it in computer memory and compute the operation $u \mapsto Lu$ by matrix multiplication.

There are, however, several disadvantages to implementing L using this matrix-based approach. One is that constructing the matrix representation is tedious. Another is that confusion can arise when swapping between column- and row-major orderings in order to shift between languages or communicate with colleagues. A third is that differences are introduced between computer code and the natural representation (2.13), which can be a source of bugs. A fourth issue is that an $n \times n$ matrix has to be instantiated in memory, even though the linear operation in (2.13) is only an inner product in $\mathbb{R}^k$. This issue can be alleviated in most languages by employing sparse matrices, but doing so adds boilerplate and can be a source of inefficiency.

Because of these issues, most modern scientific computing environments support linear operators directly, as well as actions on linear operators such as inverting linear maps. These factors further encourage us to take an operator-based approach.

2.3.3.4 Positive Operators and Markov Operators

Having agreed on the benefits of an operator-theoretic exposition, let us now describe some kinds of linear operators. We continue to assume that X is a finite set with n elements.

The set $\mathbb{R}_+^{\mathsf{X}}$ of all $u \in \mathbb{R}^{\mathsf{X}}$ with $u \geqslant 0$ is called the **positive cone** of $\mathbb{R}^{\mathsf{X}}$. An operator $L \in \mathbb{L}(\mathsf{X})$ is called **positive** if L is invariant on the positive cone; that is, if

$$u \geqslant 0 \implies Lu \geqslant 0. \tag{2.14}$$

Example 2.3.1. The operator $L \in \mathbb{L}(\mathsf{X})$ defined in (2.13) is positive whenever $Q \geqslant 0$. This is because

$$u \geqslant 0 \implies \sum_{z'} u(y, z')Q(z, z') \geqslant 0 \text{ for all } x = (y, z) \text{ in } \mathsf{X}.$$

Lemma 2.3.5. *An operator $L \in \mathbb{L}(\mathsf{X})$ is positive if and only if its matrix representation is a nonnegative matrix.*

Exercise 2.3.10. Prove Lemma 2.3.5.

Remark 2.3.4. The characterization in Lemma 2.3.5 suggests that we should call a linear operator satisfying (2.14) "nonnegative" rather than positive. Nevertheless, our terminology is standard (see, e.g., Zaanen (2012)).

Exercise 2.3.11. Given $L \in \mathbb{L}(\mathsf{X})$, prove the following statement: L is positive if and only if L is order-preserving on $\mathbb{R}^{\mathsf{X}}$ under the pointwise order.

An operator $P \in \mathbb{L}(\mathsf{X})$ is called a **Markov operator** on $\mathbb{R}^{\mathsf{X}}$ if P is positive and $P\mathbb{1} = \mathbb{1}$. We let

$$\mathbb{M}(\mathsf{X}) := \text{ the set of all Markov operators on } \mathbb{R}^{\mathsf{X}}$$

Exercise 2.3.12. Fix $P \in \mathbb{L}(\mathsf{X})$. Prove that $P \in \mathbb{M}(\mathsf{X})$ if and only if its matrix representation is nonnegative with rows summing to one.

Exercise 2.3.13. Fix $P \in \mathbb{L}(\mathsf{X})$. Prove that $P \in \mathbb{M}(\mathsf{X})$ if and only if the function φP defined by

$$(\varphi P)(x') = \sum_{x \in \mathsf{X}} P(x, x')\varphi(x) \qquad (x' \in \mathsf{X}) \tag{2.15}$$

is in $\mathscr{D}(\mathsf{X})$ whenever $\varphi \in \mathscr{D}(\mathsf{X})$.

Markov operators are important for us because they generate Markov dynamics, which is one of the foundations of dynamic programming. In fact (2.15) is a rule for updating distributions by one period under the Markov dynamics specified by P. We begin this discussion in the next chapter.

2.4 Chapter Notes

High quality textbooks on fixed points and numerical analysis include Cheney (2013) and Atkinson and Han (2005). Good background on order-theoretic fixed point methods can be found in Guo et al. (2004) and Zhang (2012). Some applications of Theorem 7.1.3 to network analysis can be found in Sargent and Stachurski (2022).

Chapter 3

Markov Dynamics

Most well-designed dynamic programs have a Markov structure, due to the fact the current action depends only on the current state (see the discussion in §1.3.1.3). In fact one of the most important classes of dynamic programs is the set of so-called Markov decision processes, which we treat in Chapter 5.

In this chapter, to prepare for our analysis of dynamic programs with Markov structure, we study a class of stochastic processes called Markov chains. Markov chains (and Markov processes on general state spaces) are fundamental not only to dynamic programming, but also a vast range of economic and financial models.

At the end of this chapter we return to the job search problem from Chapter 1 and weaken some of the assumptions we imposed there. One of the extensions we consider is allowing wage draws to be correlated over time (rather than IID), as is typically the case over the business cycle. Correlated wage draws are implemented as a Markov chain.

3.1 Foundations

To begin, we state and discuss foundational properties of Markov models, from fundamental definitions to expectations and dynamics.

3.1.1 Markov Chains

In this section we define Markov chains and discuss some fundamental properties. After recalling the basic properties of Markov operators, we show how these properties connect to transition probabilities and laws of motion for Markov chains.

3.1.1.1 Defining Markov Chains

Let $\mathsf{X} = \{x_1, \dots, x_n\}$. We will consider random processes $(X_t)_{t \geqslant 0}$ taking values in X and, in this setting, X is called the **state space** of the process. Here we focus on Markov chains, each one of which will be generated by some $P \in \mathbb{M}(\mathsf{X})$, the set of Markov operators on $\mathbb{R}^{\mathsf{X}}$. The element $P(x, x')$ of the matrix representation of P will give the probability of the chain moving from x to x'.

For this idea to make sense we need $P(x, x')$ to be nonnegative and $\sum_{x' \in \mathsf{X}} P(x, x')$ to equal 1 for every $x \in \mathsf{X}$, since the chain must go somewhere in the state space at each update. As shown in Exercise 2.3.12, these properties are guaranteed by the assumption $P \in \mathbb{M}(\mathsf{X})$.

Let $(X_t)_{t \geqslant 0}$ be a sequence of random variables taking values in X. We call (X_t) a **Markov chain** on X if there exists a $P \in \mathbb{M}(\mathsf{X})$ such that

$$\mathbb{P}\{X_{t+1} = x' \mid X_0, X_1, \dots, X_t\} = P(X_t, x') \quad \text{for all} \quad t \geqslant 0, \; x' \in \mathsf{X}. \tag{3.1}$$

In this context, P is also called the **transition matrix** of the Markov chain.

To simplify terminology, we call an X-valued random process $(X_t)_{t \geqslant 0}$ **P-Markov** when it satisfies (3.1). We call either X_0 or its distribution ψ_0 the **initial condition** of (X_t) depending on context.

The definition of a Markov chain says two things:

(i) When updating to X_{t+1} from X_t, earlier states are not required.

(ii) The matrix P encodes all of the information required to perform the update, given the current state X_t.

One way to think about Markov chains is algorithmically: Fix $P \in \mathbb{M}(\mathsf{X})$ and let ψ_0 be an element of $\mathscr{D}(\mathsf{X})$. Now generate (X_t) via Algorithm 2. The resulting sequence is P-Markov with initial condition ψ_0.

Algorithm 2: Generation of P-Markov (X_t) with initial condition ψ_0

```
t ← 0
X_t ← a draw from ψ_0
while t < ∞ do
    X_{t+1} ← a draw from the distribution P(X_t, ·)
    t ← t + 1
end
```

3.1.1.2 Application: S-s Dynamics

As an example, let us consider a firm whose inventory behavior follows S-s dynamics, meaning that the firm waits until its inventory of a given product falls below some level $s > 0$ and then replenishes by buying some fixed amount. This kind of behavior is reasonable if ordering inventory involves a fixed cost. (Later, in §5.2.1, we will show how S-s behavior arises naturally in a model where the firm chooses its inventory path to maximize its present value.)

To implement S-s dynamics, we suppose that a firm's inventory $(X_t)_{t \geqslant 0}$ of a given product obeys

$$X_{t+1} = \max\{X_t - D_{t+1}, 0\} + S\mathbb{1}\{X_t \leqslant s\},$$

where

- $(D_t)_{t \geqslant 1}$ is an exogenous IID demand process with $D_t \stackrel{d}{=} \varphi \in \mathscr{D}(\mathbb{Z}_+)$ for all t and
- S is the amount of stock ordered every time that inventory falls below s.

For the distribution φ of the demand process we take the geometric distribution, so that $\varphi(d) = \mathbb{P}\{D_t = d\} = p(1-p)^d$ for $d \in \mathbb{Z}_+$.

Exercise 3.1.1. A suitable state space for this model is $\mathsf{X} := \{0, \dots, S + s\}$, since

$$X_t \in \mathsf{X} \implies \mathbb{P}\{X_{t+1} \in \mathsf{X}\} = 1$$

for all t. Verify this claim.

If we define

$$h(x, d) = \max\{x - d, 0\} + S\mathbb{1}\{x \leqslant s\},$$

so that $X_{t+1} = h(X_t, D_{t+1})$ for all t, then the transition matrix can be expressed as

$$P(x, x') = \mathbb{P}\{h(x, D_{t+1}) = x'\} = \sum_{d \geqslant 0} \mathbb{1}\{h(x, d) = x'\}\varphi(d)$$

for $(x, x') \in \mathsf{X} \times \mathsf{X}$. In calculations we can truncate the infinite sum and still obtain a good approximation to P.

Listing 7 provides Julia code that implements the model, simulates inventory paths and computes other objects of interest. Since the state space $\mathsf{X} = \{x_1, \dots, x_n\}$ corresponds to $\{0, \dots, S + s\}$, we have $x_i = i - 1$. This convention is used when computing `P[i, j]`, which corresponds to $P(x_i, x_j)$. The code in the listing is used to produce the simulation of inventories in Figure 3.1.

Figure 3.1: Inventory simulation (`inventory_sim.jl`)

The function `compute_mc` returns an instance of a `MarkovChain` object, which can store both the state X and the transition probabilities. The `QuantEcon.jl` library defines this data type and provides functions that act on it, in order to facilitate simulation of Markov chains, computation of stationary distributions and other related tasks.

3.1.1.3 Higher Order Transition Matrices

Given a finite state space X, $k \geqslant 0$ and $P \in \mathbb{M}(\mathsf{X})$, let P^k is the k-th power of P. (If $k = 0$, then P^k is the identity matrix.) Since $\mathbb{M}(\mathsf{X})$ is closed under multiplication (Exercise 2.3.2), P^k is in $\mathbb{M}(\mathsf{X})$ for all $k \geqslant 0$. In this context, P^k is sometimes called the **k-step transition matrix** corresponding to P. In what follows, $P^k(x, x')$ denotes the (x, x')-th element of the matrix representation of P^k.

The k-step transition matrix has the following interpretation: If (X_t) is P-Markov, then, for any $t, k \in \mathbb{Z}_+$ and $x, x' \in \mathsf{X}$,

$$P^k(x, x') = \mathbb{P}\{X_{t+k} = x' \mid X_t = x\}. \tag{3.2}$$

In other words, P^k provides the k-step transition probabilities for the P-Markov chain (X_t), as suggested by its name.

This claim can be verified by induction. Fix $t \geqslant 0$. The claim is obviously true

```julia
using Distributions, IterTools, QuantEcon

function create_inventory_model(; S=100,  # Order size
                                  s=10,   # Order threshold
                                  p=0.4)  # Demand parameter
    ϕ = Geometric(p)
    h(x, d) = max(x - d, 0) + S*(x <= s)
    return (; S, s, p, ϕ, h)
end

"Simulate the inventory process."
function sim_inventories(model; ts_length=200)
    (; S, s, p, ϕ, h) = model
    X = Vector{Int32}(undef, ts_length)
    X[1] = S  # Initial condition
    for t in 1:(ts_length-1)
        X[t+1] = h(X[t], rand(ϕ))
    end
    return X
end

"Compute the transition probabilities and state."
function compute_mc(model; d_max=100)
    (; S, s, p, ϕ, h) = model
    n = S + s + 1  # Size of state space
    state_vals = collect(0:(S + s))
    P = Matrix{Float64}(undef, n, n)
    for (i, j) in product(1:n, 1:n)
        P[i, j] = sum((h(i-1, d) == j-1)*pdf(ϕ, d) for d in 0:d_max)
    end
    return MarkovChain(P, state_vals)
end

"Compute the stationary distribution of the model."
function compute_stationary_dist(model)
    mc = compute_mc(model)
    return mc.state_values, stationary_distributions(mc)[1]
end
```

Listing 7: An implementation of S-s inventory dynamics (`inventory_sim.jl`)

when $k = 0, 1$. Suppose the claim is also true at k and now consider the case $k+1$. By the law of total probability, for given $x, x' \in \mathsf{X}$, we have

$$\mathbb{P}\{X_{t+k+1} = x' \mid X_t = x\} = \sum_z \mathbb{P}\{X_{t+k+1} = x' \mid X_{t+k} = z\}\mathbb{P}\{X_{t+k} = z \mid X_t = x\}.$$

The induction hypothesis allows us to use (3.2) at k, so the last equation becomes

$$\mathbb{P}\{X_{t+k+1} = x' \mid X_t = x\} = \sum_z P^k(x, z)P(z, x') = P^{k+1}(x, x').$$

The law (3.2) is now verified at $k+1$, completing our proof by induction.

We can now give the following useful characterization of irreducibility:

Lemma 3.1.1. *Given $P \in \mathbb{M}(\mathsf{X})$, the following statements are equivalent:*

(i) *P is irreducible.*

(ii) *For any P-chain (X_t) and any $x, x' \in \mathsf{X}$, there exists a $k \geqslant 0$ such that*

$$\mathbb{P}\{X_k = x' \mid X_0 = x\} > 0.$$

Thus, irreducibility of P means that the P-chain eventually visit any state from any other state with positive probability.

Proof of Lemma 3.1.1. Fix $P \in \mathbb{M}(\mathsf{X})$. Recall that P is irreducible if and only if $\sum_{k \geqslant 0} P^k \gg 0$. This is equivalent to the statement that, for each $(x, x') \in \mathsf{X} \times \mathsf{X}$, there exists a $k \geqslant 0$ such that $P^k(x, x') > 0$, which is in turn equivalent to part (ii) of Lemma 3.1.1. □

Exercise 3.1.2. Using Lemma 3.1.1, prove that the stochastic matrix associated with the S-s inventory dynamics in §3.1.1.2 is irreducible.

Several libraries have code for testing irreducibility. For Julia, see `QuantEcon.jl`. See Listing 8 for an example of a call to this functionality. In this case, irreducibility fails because state 2 is an **absorbing state**. Once entered, the probability of ever leaving that state is zero. (A subset Y of X with this property is called an **absorbing set**.)

3.1.2 Stationarity and Ergodicity

In this section we review aspects of Markov dynamics, including stationarity and ergodicity.

```
using QuantEcon
P = [0.1 0.9;
     0.0 1.0]
mc = MarkovChain(P)
print(is_irreducible(mc))
```

Listing 8: Testing irreducibility (`is_irreducible.jl`)

Fix $P \in \mathbb{M}(\mathsf{X})$ and let (X_t) be a P-chain. Let ψ_t be the distribution of X_t for all t. For each $t \geqslant 0$,

$$\psi_{t+1}(x') = \sum_{x \in \mathsf{X}} P(x, x')\psi_t(x) \quad \text{for all } x \in \mathsf{X}. \tag{3.3}$$

This just states that

$$\mathbb{P}\{X_{t+1} = x'\} = \sum_{x \in \mathsf{X}} \mathbb{P}\{X_{t+1} = x' \mid X_t = x\}\mathbb{P}\{X_t = x\}$$

for all $x, x' \in \mathsf{X}$, which is true by the law of total probability. With each ψ_t regarded as a row vector, (3.3) can also be written as $\psi_{t+1} = \psi_t P$. Iterating on this equation, we get $\psi_t = \psi_0 P^t$ for all t. In summary,

$$(X_t)_{t \geqslant 0} \text{ is } P\text{-Markov with } X_0 \stackrel{d}{=} \psi_0 \implies X_t \stackrel{d}{=} \psi_0 P^t \text{ for all } t \geqslant 0. \tag{3.4}$$

Note 3.1.1. The fundamental relation $\psi_{t+1} = \psi_t P$ and the result (3.4) require that each ψ_t is a row vector. In what follows, we always treat distributions ψ_t of $(X_t)_{t \geqslant 0}$ as row vectors.

Consistent with our definition of stationary distributions in §2.3.1.2, a distribution $\psi^* \in \mathscr{D}(\mathsf{X})$ is called **stationary** for P if

$$\sum_{x \in \mathsf{X}} P(x, x')\psi^*(x) = \psi^*(x') \quad \text{for all } x \in \mathsf{X}.$$

Since distributions are regarded as row vectors, we can write this expression more simply as $\psi^* P = \psi^*$. In view of (3.3), if ψ^* is stationary and X_t has distribution ψ^*, then so does X_{t+1}, and hence X_{t+k} for all $k \geqslant 1$.

We saw in Exercise 2.3.2 that every $P \in \mathbb{M}(\mathsf{X})$ has at least one stationary distribution, and that uniqueness in $\mathscr{D}(\mathsf{X})$ holds whenever P is irreducible. The next result is also fundamental.

Figure 3.2: Ergodicity (`inventory_sim.jl`)

Theorem 3.1.2. *If P is irreducible with stationary distribution ψ^*, then, for any P-Markov chain (X_t) and any $x \in \mathsf{X}$, we have*

$$\mathbb{P}\left\{\lim_{k\to\infty} \frac{1}{k}\sum_{t=0}^{k-1} \mathbb{1}\{X_t = x\} = \psi^*(x)\right\} = 1. \tag{3.5}$$

A proof of (3.5) can be found in Brémaud (2020).

Property (3.5) tells us that, with probability one (i.e., for almost every P-Markov chain that we generate), the fraction of time that the chain spends in any given state is, in the limit, equal to the probability assigned to that state by the stationary distribution. Markov chains with this property are sometimes said to be **ergodic**.

Since the S-s inventory model from §3.1.1.2 is irreducible, the ergodicity result from Theorem 3.1.2 applies. In particular, the model has only one stationary distribution ψ^* in $\mathscr{D}(\mathsf{X})$, where $\mathsf{X} = \{0, \ldots, S + s\}$, and (3.5) is valid whenever (X_t) is generated by the model. Figure 3.2 illustrates this by plotting both the stationary distribution ψ^* (which is computed using the code in Listing 7), and the value $m(y) := \frac{1}{k}\sum_{t=0}^{k-1} \mathbb{1}\{X_t = y\}$ at each $y \in \mathsf{X}$. The value of k is set to $1,000,000$. As predicted by the theorem, the fraction of time spent by the chain in each state is close to the probability assigned by ψ^*.

```
function create_laborer_model(; α=0.3, β=0.2)
    return (; α, β)
end

function laborer_update(x, model)  # update X from t to t+1
    (; α, β) = model
    if x == 1
        x′ = rand() < α ? 2 : 1
    else
        x′ = rand() < β ? 1 : 2
    end
    return x′
end
```

Listing 9: Updating the state of the day laborer (`laborer_sim.jl`)

Exercise 3.1.3. Let (X_t) be P-Markov on X with $X_0 \overset{d}{=} \psi_0$. Show that

$$\mathbb{E}h(X_t) = \psi_0 P^t h = \langle \psi_0 P^t, h \rangle \quad \text{for all } t \in \mathbb{N} \text{ and } h \in \mathbb{R}^{\mathsf{X}}. \tag{3.6}$$

3.1.2.1 Application: Day Laborer

Suppose that a day laborer is either unemployed ($X_t = 1$) or employed ($X_t = 2$) in each period. In state 1 he is hired with probability $\alpha \in (0, 1)$. In state 2 he is fired with probability $\beta \in (0, 1)$. The corresponding state space and transition matrix are

$$\mathsf{X} = \{1, 2\} \quad \text{and} \quad P = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}. \tag{3.7}$$

Listing 9 provides a function to update from X_t to X_{t+1}, using the fact that `rand()` generates a draw from the uniform distribution on $[0, 1)$.

Exercise 3.1.4. Explain why Listing 9 updates the current state according to the probabilities in P.

Exercise 3.1.5. P is positive it must be irreducible, so P has the unique stationary distribution in $\psi^* \in \mathscr{D}(\mathsf{X})$. Show that ψ^* is given by

$$\psi^* = \frac{1}{\alpha + \beta} \begin{pmatrix} \beta & \alpha \end{pmatrix}. \tag{3.8}$$

It is also true that $\psi P^t \to \psi^*$ as $t \to \infty$ for any $\psi \in \mathscr{D}(\mathsf{X})$. In other words, the operator P when understood as the mapping $\psi \mapsto \psi P$, is globally stable on $\mathscr{D}(\mathsf{X})$

Exercise 3.1.6. Prove this using the Perron–Frobenius theorem. (More generally, show that this global stability result holds for any $P \in \mathbb{M}(\mathsf{X})$ with $P \gg 0$.)

Exercise 3.1.7. Fix $\alpha = 0.3$ and $\beta = 0.2$. Compute the sequence (ψP^t) for different choices of ψ and confirm that your results are consistent with the claim that $\psi P^t \to \psi^*$ as $t \to \infty$ for any $\psi \in \mathscr{D}(\mathsf{X})$.

Exercise 3.1.8. Since P is irreducible, ergodicity holds. Simulate a long realization Markov of a P-Markov chain from an arbitrary initial condition and confirm that your results are consistent with (3.5).

3.1.3 Approximation

It can be helpful to reduce continuous state Markov models to finite state models in order to simplify numerical calculations. For an example of this form of discretization, consider a **linear Gaussian AR(1)** model, where $(X_t)_{t \geqslant 0}$ evolves in $\mathbb{R}$ according to

$$X_{t+1} = \rho X_t + b + \nu \varepsilon_{t+1}, \quad (\varepsilon_t) \overset{\text{IID}}{\sim} N(0, 1). \tag{3.9}$$

Here we discuss one technique for discretizing (3.9), often called **Tauchen's method**, and use it to illustrate concepts related to stationarity.[1]

We assume throughout that $|\rho| < 1$. Under this assumption, (3.9) has a unique **stationary distribution** ψ^* given by

$$\psi^* = N(\mu_x, \sigma_x^2) \quad \text{with} \quad \mu_x := \frac{b}{1 - \rho} \quad \text{and} \quad \sigma_x^2 := \frac{\nu^2}{1 - \rho^2}.$$

[1]We feature Tauchen's method because of its simplicity. Tauchen's method can be sub-optimal in some cases. For a more general discretization method and a survey of the literature, see Farmer and Toda (2017).

This means that ψ^* has the following property:

$$X_t \stackrel{d}{=} \psi^* \text{ and } X_{t+1} = \rho X_t + b + \nu\varepsilon_{t+1} \text{ implies } X_{t+1} \stackrel{d}{=} \psi^*.$$

Exercise 3.1.9. Suppose that $X_t \stackrel{d}{=} \psi^*$, $\varepsilon_{t+1} \stackrel{d}{=} N(0,1)$ and X_t and ε_{t+1} are independent. Prove that $\rho X_t + b + \nu\varepsilon_{t+1}$ has distribution ψ^*. Is this still true if we drop the independence assumption made above?

When $|\rho| < 1$, this model is ergodic in a similar sense to (3.5) on page 80: on average, realizations of the process spend most of their time in regions of the state where the stationary distribution puts high probability mass. (You can check this via simulations if you wish.) Hence, in the discretization that follows, the discrete state space will be centered in this area.

Exercise 3.1.10. Set $b = 0$ in (3.9) and let F be the $N(0,\nu^2)$ cdf. Show that

$$\mathbb{P}\{t - \delta < X_{t+1} \leqslant t + \delta \,|\, X_t = x\} = F(t - \rho x + \delta) - F(t - \rho x - \delta) \tag{3.10}$$

for all $\delta, t \in \mathbb{R}$.

We start with the case $b = 0$. As a first step, we choose n as the number of states for the discrete approximation and m as an integer that parameterizes the width of the state space. Then we create a state space $\mathsf{X} := \{x_1, \dots, x_n\} \subset \mathbb{R}$ as a linear grid that brackets the stationary mean on both sides by m standard deviations:

- set $x_1 = -m\,\sigma_x$,
- set $x_n = m\,\sigma_x$ and
- set $x_{i+1} = x_i + s$ where $s = (x_n - x_1)/(n-1)$ and i in $[n-1]$.

The next step is to create an $n \times n$ matrix P computed to approximate the dynamics in (3.9). For $i, j \in [n]$,

(i) if $j = 1$, then set $P(x_i, x_j) = F(x_1 - \rho x_i + s/2)$.

(ii) If $j = n$, then set $P(x_i, x_j) = 1 - F(x_n - \rho x_i - s/2)$.

(iii) Otherwise, set $P(x_i, x_j) = F(x_j - \rho x_i + s/2) - F(x_j - \rho x_i - s/2)$.

The first two are boundary rules and the third applies Exercise 3.1.10.

Exercise 3.1.11. Prove that $\sum_{j=1}^n P(x_i, x_j) = 1$ for all $i \in [n]$.

Figure 3.3: Comparison of $\psi^* = N(\mu_x, \sigma_x^2)$ and its discrete approximant

Finally, if $b \neq 0$, then we shift the state space to center it on the mean μ_x of the stationary distribution $N(\mu_x, \sigma_x^2)$. This is done by replacing x_i with $x_i + \mu_x$ for each i.

Julia routines for computing X and P can be found in the library QuantEcon.jl.

Figure 3.3 compares the continuous stationary distribution ψ^* and the unique stationary distribution of the discrete approximation when X and P are constructed as above, under the parameterization $\rho = 0.9$, $b = 0.0$, $\nu = 1.0$. The discretization parameters were set to $n = 15$ and $m = 3$.

3.2 Conditional Expectations

In this section we discuss how to take conditional expectations with respect to Markov chains. The theory will be essential for the study of finite MDPs, since, in these models, lifetime rewards are mathematical expectations of flow reward functions with respect to Markov chains.

3.2.1 Mathematical Expectations

We begin with some foundational results concerning expectations of functions of Markov chains.

3.2.1.1 Conditional Expectations

Fix $P \in \mathbb{M}(\mathsf{X})$. For each $h \in \mathbb{R}^{\mathsf{X}}$, we define

$$(Ph)(x) = \sum_{x' \in \mathsf{X}} h(x')P(x, x') \qquad (x \in \mathsf{X}). \tag{3.11}$$

Noting that $P(x, \cdot)$ is the distribution of X_{t+1} given $X_t = x$, we can equivalently write

$$(Ph)(x) = \mathbb{E}[h(X_{t+1}) \,|\, X_t = x], \tag{3.12}$$

where (X_t) is any P-Markov chain on X. In terms of matrix algebra, viewing h has an $n \times 1$ column vector, the expression $(Ph)(x)$ is one element of the vector Ph obtained by premultiplying h by P.

The interpretation in (3.12) extends to powers of P. In particular, we have

$$(P^k h)(x) = \sum_{x' \in \mathsf{X}} h(x')P^k(x, x') = \mathbb{E}[h(X_{t+k}) \,|\, X_t = x]. \tag{3.13}$$

Exercise 3.2.1. Show that

(i) Every constant function $h \in \mathbb{R}^{\mathsf{X}}$ is a fixed point of P (i.e., $Ph = h$).

(ii) $\max_x |Ph(x)| \leqslant \max_x |h(x)|$ for all $h \in \mathbb{R}^{\mathsf{X}}$.

3.2.1.2 The Law of Iterated Expectations

The **law of iterated expectations** appears time and again in dynamic modeling, particularly in economics and finance. One common version of the law states that if X and Y are two random variables, then $\mathbb{E}[\mathbb{E}[Y \,|\, X]] = \mathbb{E}[Y]$. Let's show this law holds in the Markov case.

Let (X_t) be P-Markov with $X_0 \stackrel{d}{=} \psi_0$. Fix $t, k \in \mathbb{N}$. Set $\mathbb{E}_t := \mathbb{E}[\cdot \,|\, X_t]$. We claim that

$$\mathbb{E}[\mathbb{E}_t[h(X_{t+k})]] = \mathbb{E}[h(X_{t+k})] \quad \text{for any } h \in \mathbb{R}^{\mathsf{X}}. \tag{3.14}$$

To see this, recall that $\mathbb{E}[h(X_{t+k}) \,|\, X_t = x] = (P^k h)(x)$. Hence $\mathbb{E}[h(X_{t+k}) \,|\, X_t] = (P^k h)(X_t)$. Therefore,

$$\mathbb{E}[\mathbb{E}_t[h(X_{t+k})]] = \mathbb{E}[(P^k h)(X_t)] = \sum_{x'} (P^k h)(x')\psi_t(x') = \sum_{x'} (P^k h)(x')(\psi_0 P^t)(x').$$

Since $\psi_0 P^t$ is a row vector, we can write the last expression as

$$\psi_0 P^t P^k h = \psi_0 P^{t+k} h = \psi_{t+k} h = \mathbb{E} h(X_{t+k}).$$

Hence (3.14) holds.

3.2.2 Monotonicity

In this section we connect Markov chains to order theory via stochastic dominance. These ideas will have many applications below.

3.2.2.1 Increasing Functions

Let X be a finite set partially ordered by $\preceq$. Let $i\mathbb{R}^{\mathsf{X}}$ be the set of increasing functions in $\mathbb{R}^{\mathsf{X}}$. Thus, for $h \in \mathbb{R}^{\mathsf{X}}$,

$$h \in i\mathbb{R}^{\mathsf{X}} \quad \iff \quad x, y \in \mathsf{X} \text{ and } x \preceq y \text{ implies } h(x) \leqslant h(y).$$

Example 3.2.1. If $\mathsf{X} = \{1, \ldots, n\}$ and $\preceq$ is the usual order $\leqslant$ on $\mathbb{R}$, then $x \mapsto 2x$ and $x \mapsto \mathbb{1}\{2 \leqslant x\}$ are in $i\mathbb{R}^{\mathsf{X}}$ but $x \mapsto -x$ and $x \mapsto \mathbb{1}\{x \leqslant 2\}$ are not.

The next exercise shows that an increasing function can be represented as the sum of increasing binary functions. This fact will be valuable when we characterize orders over distributions, in §3.2.2.2.

Exercise 3.2.2. Let $\mathsf{X} = \{x_1, \ldots, x_n\}$ where $x_k \preceq x_{k+1}$ for all k. Show that, for any $u \in i\mathbb{R}^{\mathsf{X}}$, there exist $s_1, \ldots, s_n$ in $\mathbb{R}_+$ such that $u(x) = \sum_{k=1}^n s_k \mathbb{1}\{x \geqslant x_k\}$ for all $x \in \mathsf{X}$.

3.2.2.2 Stochastic Dominance

It is useful to have a partial order over distributions that tells us when one distribution is "larger" than another. One of many possible applications for such an order is as follows: we could apply our parametric monotonicity results from §2.1.3 to study what kinds of modifications to the wage offer distribution (e.g., shifts up under the stated partial order) would lead a worker in §1.3 to increase his or her reservation wage.

Figure 3.4: Two binomial distributions

There are several partial orders over distributions routinely used in economics and finance. In this section we consider an important one called "first order stochastic dominance."

Let's start with an example. Recall that a random variable X is binomial $B(n, 0.5)$ if it counts the number of heads in n flips of a fair coin. Figure 3.4 shows two probability mass functions, one of distribution $\varphi \stackrel{d}{=} X \sim B(10, 0.5)$ and another of $\psi \stackrel{d}{=} Y \sim B(18, 0.5)$. Since Y counts over more flips, we expect it to take larger values, in some sense, and we also expect its distribution ψ to reflect this. But how can we make this idea precise?

The standard order over distributions, which captures this idea, is defined as follows: Given finite set X partially ordered by $\preceq$ and $\varphi, \psi \in \mathscr{D}(\mathsf{X})$, we say that ψ **stochastically dominates** φ and write $\varphi \preceq_{\mathrm{F}} \psi$ if

$$\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x) \quad \text{for every } u \text{ in } i\mathbb{R}^{\mathsf{X}} \tag{3.15}$$

The relation $\preceq_{\mathrm{F}}$ is also called **first order stochastic dominance** to differentiate it from other forms of stochastic order.

Example 3.2.2. Consider the distributions $\varphi = B(10, 0.5)$ and $\psi = B(18, 0.5)$ defined above. For the outcome space we take $\mathsf{X} = \{0, \ldots, 18\}$. It is not hard to see that $\varphi \preceq_{\mathrm{F}} \psi$ holds. Indeed, if we take $W_1, \ldots, W_{18}$ to be IID binary random variables with $\mathbb{P}\{W_i = 1\} = 0.5$ for all i, then $X := \sum_{i=1}^{10} W_i$ has distribution φ and $Y := \sum_{i=1}^{18} W_i$ has distribution ψ. In addition, we can see that $X \leqslant Y$ with probability one (i.e., for any outcome of the draws $W_1, \ldots, W_{18}$). Hence, for any given $u \in i\mathbb{R}^{\mathsf{X}}$, we have $u(X) \leqslant u(Y)$

with probability one. Hence $\mathbb{E}u(X) \leqslant \mathbb{E}u(Y)$ holds, which is the same statement as (3.15).

A good way to understand the definition of first order stochastic dominance is to suppose that an agent has preferences over outcomes in X described by a utility function $u \in \mathbb{R}^{\mathsf{X}}$. Suppose in addition that the agent prefers more to less, in the sense that $u \in i\mathbb{R}^{\mathsf{X}}$, and that the agent ranks lotteries over X according to expected utility, so that the agent evaluates $\varphi \in \mathscr{D}(\mathsf{X})$ according to $\sum_x u(x)\varphi(x)$. Then the agent (weakly) prefers ψ to φ whenever $\varphi \preceq_{\mathrm{F}} \psi$.

We can go one step further. Consider now the class $\mathscr{A}$ of *all* agents who (a) have preferences over outcomes in X, (b) prefer more to less, and (c) rank lotteries over X according to expected utility. Then $\varphi \preceq_{\mathrm{F}} \psi$ if and only if every agent in $\mathscr{A}$ prefers ψ to φ.

Remark 3.2.1. The last paragraph helps explain the pervasiveness of stochastic dominance in economics. It is standard to assume that economic agents have increasing utility functions and use expected utility to evaluate lotteries. In such environments, an upward shift in a lottery, as measured by stochastic dominance, makes all agents better off. Such a change is unambiguously welfare enhancing.

Exercise 3.2.3. A simple setting in which we can study stochastic dominance is where $\mathsf{X} = \{1, 2\}$ and X is partially ordered by $\leqslant$. In this case, $\varphi \preceq_{\mathrm{F}} \psi$ if and only φ puts more mass on 1 than ψ, and, equivalently, less mass on 2. That is,

$$\varphi \preceq_{\mathrm{F}} \psi \iff \psi(1) \leqslant \varphi(1) \iff \varphi(2) \leqslant \psi(2).$$

Verify the equivalence of these statements.

Another perspective on stochastic dominance is helpful. To state it, we first introduce the notation

$$G^{\varphi}(y) := \sum_{x \succeq y} \varphi(x) \qquad (\varphi \in \mathscr{D}(\mathsf{X}),\ y \in \mathsf{X}).$$

For a given distribution φ, the function G^{φ} is sometimes called the **counter** CDF (counter cumulative distribution function) of φ.

Lemma 3.2.1. *For each $\varphi, \psi \in \mathscr{D}(\mathsf{X})$, the following statements hold:*

(i) $\varphi \preceq_F \psi \implies G^{\varphi} \leqslant G^{\psi}$.

(ii) *If X is totally ordered by $\preceq$, then $G^{\varphi} \leqslant G^{\psi} \implies \varphi \preceq_F \psi$.*

Figure 3.5: Visualization of $\varphi \preceq_{\mathrm{F}} \psi$

The proof is given on page 292. Figure 3.5 helps to illustrate. Here $\mathsf{X} \subset \mathbb{R}$ and φ and ψ are distributions on X. We can see that $\varphi \preceq_{\mathrm{F}} \psi$ because the counter CDFs are ordered, in the sense that $G^\varphi \leqslant G^\psi$ pointwise on X.

Lemma 3.2.2. *Stochastic dominance is a partial order on $\mathscr{D}(\mathsf{X})$.*

Exercise 3.2.4. Prove the transitivity component of Lemma 3.2.2. That is, prove that $\preceq_{\mathrm{F}}$ is transitive on $\mathscr{D}(\mathsf{X})$.

Exercise 3.2.5. Fix $\tau \in (0, 1]$ and let Q_τ be the quantile function defined on page 29. Choose $\varphi, \psi \in \mathscr{D}(\mathsf{X})$ and let X, Y be X-valued random variables with distributions φ and ψ respectively. Prove that $\varphi \preceq_{\mathrm{F}} \psi$ implies $Q_\tau(X) \leqslant Q_\tau(Y)$.

3.2.2.3 Monotone Markov Chains

Let X be a finite set partially ordered by $\preceq$. A Markov operator $P \in \mathbb{M}(\mathsf{X})$ is called **monotone increasing** if

$$x, y \in \mathsf{X} \text{ and } x \preceq y \quad \Longrightarrow \quad P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot).$$

In other words, P is monotone increasing if shifting up the current state shifts up the next period state, in the sense that its distribution increases in the stochastic dominance ordering on $\mathscr{D}(\mathsf{X})$. Below, we will see that monotonicity of Markov operators is closely related to monotonicity in value functions in some important applications.

Monotonicity of Markov operators is related to positive autocorrelation. To illustrate the idea, consider the AR(1) model $X_{t+1} = \rho X_t + \sigma \varepsilon_{t+1}$ from §3.1.3 and suppose we apply Tauchen discretization, mapping the parameters ρ, σ and a discretization size n into a Markov operator P on state space $\mathsf{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}$. If $\rho \geqslant 0$, so that positive autocorrelation holds, then P is monotone increasing.

Exercise 3.2.6. Verify this claim.

Exercise 3.2.7. In §3.1.2.1 we discussed a setting where

$$\mathsf{X} = \{1, 2\} \quad \text{and} \quad P_w = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

for some $\alpha, \beta \in [0, 1]$. Show that P_w is monotone increasing if and only if $\alpha + \beta \leqslant 1$.

Exercise 3.2.8. Prove that P is monotone increasing if and only if P is invariant on $i\mathbb{R}^{\mathsf{X}}$; that is, if $h \in i\mathbb{R}^{\mathsf{X}}$ implies $Ph \in i\mathbb{R}^{\mathsf{X}}$.

Exercise 3.2.9. Prove: If P is monotone increasing then so is P^t for all $t \in \mathbb{N}$.

3.2.3 Geometric Sums

Dynamic programs often combine a reward sequence $(R_t)_{t \geqslant 0}$ into a lifetime value V_0 is via a geometric sum with constant discount factor, so that $V_0 = \mathbb{E} \sum_{t=0}^{\infty} \beta^t R_t$ for some $\beta > 0$. We saw this in (1.1) on page 2, where we aggregated a profit stream $(\pi_t)_{t \geqslant 0}$ into an expected present value of the firm, and again in (1.6) on page 10, where a worker evaluated lifetime earnings in a similar way. In this section we investigate expectations of geometric sums in more detail.

3.2.3.1 Theory

Consider a conditional mathematical expectation of a discounted sum of future measurements:

$$v(x) := \mathbb{E}_x \sum_{t \geqslant 0} \beta^t h(X_t) := \mathbb{E}\left[\sum_{t \geqslant 0} \beta^t h(X_t) \mid X_0 = x\right] \tag{3.16}$$

for some constant $\beta \in \mathbb{R}_+$ and $h \in \mathbb{R}^{\mathsf{X}}$. Here

- (X_t) is P-Markov on some finite set X,
- $v(x)$ is the **lifetime reward** starting from state x, and
- $\mathbb{E}_x$ indicates we are conditioning on $X_0 = x$.

With I as the identity matrix, the next result describes v as function of the primitives β, P and h.

Lemma 3.2.3. *If $\beta < 1$, then $I - \beta P$ is invertible and*

$$v = \sum_{t \geqslant 0} (\beta P)^t h = (I - \beta P)^{-1} h. \tag{3.17}$$

Proof. Under the stated conditions we have

$$\mathbb{E}_x \sum_{t \geqslant 0} \beta^t h(X_t) = \sum_{t \geqslant 0} \beta^t \mathbb{E}_x h(X_t) = \sum_{t \geqslant 0} \beta^t (P^t h)(x), \tag{3.18}$$

where the first equality in (3.18) uses linearity of expectations and the second follows from (3.13) and the assumption that (X_t) is P-Markov starting at x.[2] Applying the Neumann Series Lemma (p. 17) to the matrix βP, we see that $\sum_{t \geqslant 0} (\beta P)^t = (I - \beta P)^{-1}$. The lemma is applicable because $r(\beta P) = \beta r(P) = \beta < 1$, as follows from Exercise 2.3.2. □

3.2.3.2 Application: Valuation of Firms

Consider a firm that receives profit stream $(\pi_t)_{t \geqslant 0}$, where each π_t is a random variable. For a shareholder, the total valuation of the firm is the expected present value of its

[2] Regarding the first equality, care must be taken when pushing mathematical expectations through infinite sums. In the present setting, justification can be provided by appealing to the dominated convergence theorem, which is one of the fundamental results of measure theory. See, for example, Williams (1991).

profit stream. Suppose for now that the interest rate is constant at $r > 0$. With $\beta := 1/(1+r)$, total valuation is

$$V_0 = \mathbb{E} \sum_{t=0}^{\infty} \beta^t \pi_t. \tag{3.19}$$

To compute this value, we need a model of how profits evolve. A common strategy is to set $\pi_t = \pi(X_t)$ for some fixed $\pi \in \mathbb{R}^{\mathsf{X}}$, where $(X_t)_{t \geqslant 0}$ is a state process. For known dynamics of (X_t) and function π, the value V_0 in (3.19) can be computed.

Here we assume that (X_t) is P-Markov for $P \in \mathbb{M}(\mathsf{X})$ with finite X. Then, conditioning on $X_0 = x$, we can write the value as

$$v(x) := \mathbb{E}_x \sum_{t=0}^{\infty} \beta^t \pi_t := \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t \pi_t \,|\, X_0 = x\right].$$

By Lemma 3.2.3, the value $v(x)$ is finite and the function $v \in \mathbb{R}^{\mathsf{X}}$ can be obtained by

$$v = \sum_{t \geqslant 0} \beta^t P^t \pi = (I - \beta P)^{-1} \pi.$$

It seems natural that valuation will be increasing if higher states generate higher profits and also predict higher states in the future. The next exercise confirms this.

Exercise 3.2.10. Let X be partially ordered and suppose that $\pi \in i\mathbb{R}^{\mathsf{X}}$ and that P is monotone increasing. (See §3.2.2 for terminology and notation.) Prove that, under these conditions, v is increasing on X.

3.2.3.3 Application: Valuing Consumption Streams

Household choices over consumption and savings are important topics. To model choices we need to attach values to different consumption paths. Here, a **consumption path** is a nonnegative random sequence $(C_t)_{t \geqslant 0}$. In what follows we consider consumption paths such that $C_t = c(X_t)$ for all $t \geqslant 0$, where $c \in \mathbb{R}_+^{\mathsf{X}}$ and $(X_t)_{t \geqslant 0}$ is P-Markov on finite set X. Thus, consumption streams are stationary functions of a finite state Markov chain.

In the standard "additively separable" model of consumer preferences with constant geometric discounting, the time zero value of a consumption stream $(C_t)_{t \geqslant 0}$,

Figure 3.6: The value of $(C_t)_{t\geqslant 0}$ given $X_t = x$

given current state $X_0 = x \in \mathsf{X}$, is

$$v(x) = \mathbb{E}_x \sum_{t\geqslant 0} \beta^t u(C_t), \tag{3.20}$$

where $\beta \in (0,1)$ is a discount factor and $u\colon \mathbb{R}_+ \to \mathbb{R}$ is called the **flow utility function**. Dependence of $v(x)$ on x comes from the initial condition $X_0 = x$ influencing the Markov state process and, therefore, the time path of consumption.

Using $C_t = c(X_t)$ and defining $r := u \circ c$ we can write $v(x) = \mathbb{E}_x \sum_{t\geqslant 0} \beta^t r(X_t)$. By Lemma 3.2.3, this sum is finite and v can be expressed as

$$v = (I - \beta P)^{-1} r. \tag{3.21}$$

Figure 3.6 shows an example when u has the constant relative risk aversion (**CRRA**) specification

$$u(c) = \frac{c^{1-\gamma}}{1-\gamma} \qquad (c \geqslant 0,\ \gamma > 0), \tag{3.22}$$

while $c(x) = \exp(x)$, so that consumption takes the form $C_t = \exp(X_t)$, and $(X_t)_{t\geqslant 0}$ is a Tauchen discretization (see §3.1.3) of $X_{t+1} = \rho X_t + \nu W_{t+1}$ where $(W_t)_{t\geqslant 1}$ is IID and standard normal. The parameters are $n = 25$, $\beta = 0.98$, $\rho = 0.96$, $\nu = 0.05$ and $\gamma = 2$. We set $r = u \circ c$ and solved for v via (3.21).

Exercise 3.2.11. Replicate Figure 3.6.

Exercise 3.2.12. The value function in Figure 3.6 appears to be increasing in the state x. Prove this for the CRRA model when $\rho \geqslant 0$.

3.3 Job Search Revisited

In this section we return to the job search problem Chapter 1, aiming to weaken some restrictive assumptions. First we show how the analysis extends to the case of Markov wage offers. Then we discuss how the solution acquires additional structure when the Markov operator for wage offers is monotone increasing. Finally, we will remove the unrealistic assumption that no job separations occur.

3.3.1 Job Search with Markov State

We allow wage offers to be correlated rather than IID and then analyze optimality.

3.3.1.1 Value Function Iteration

Assume that the wage process $(W_t)_{t \geqslant 0}$ is P-Markov on a finite set $\mathsf{W} \subset \mathbb{R}_+$, where $P \in \mathbb{M}(\mathsf{W})$. The value function v^* is defined in an analogous manner to the IID case: $v^*(w)$ is the maximum lifetime value that can be obtained when the agent is unemployed with current wage offer is w in hand.

As for the IID case, the value function satisfies a Bellman equation, which in the present setting states takes the form

$$v^*(w) = \max \left\{ \frac{w}{1-\beta}, \, c + \beta \sum_{w' \in \mathsf{W}} v^*(w') P(w, w') \right\} \qquad (w \in \mathsf{W}). \tag{3.23}$$

The intuition is analogous to the IID case (cf. (1.26) on page 32). We continue to assume that $c > 0$ and $\beta \in (0, 1)$. A full proof of the validity of (3.23) is given in Chapter 4.

The Bellman operator corresponding to (3.23) is

$$(Tv)(w) = \max \left\{ \frac{w}{1-\beta}, \, c + \beta \sum_{w' \in \mathsf{W}} v(w') P(w, w') \right\} \qquad (w \in \mathsf{W}).$$

Extending the IID definition (see (1.30) on page 35), a policy $\sigma\colon \mathsf{W} \to \{0,1\}$ is called v-greedy if

$$\sigma(w) = \mathbb{1}\left\{\frac{w'}{1-\beta} \geqslant c + \beta \sum_{w' \in \mathsf{W}} v(w')P(w,w')\right\}$$

for all $w \in \mathsf{W}$.

Let $V := \mathbb{R}_+^{\mathsf{W}}$ and endow V with the supremum norm, so that the distance between two elements f, g of V is measured by $\|f - g\| = \max_{w \in \mathsf{W}} |f(w) - g(w)|$.

Exercise 3.3.1. Prove that

(i) T is an order-preserving self-map on V.

(ii) T is a contraction of modulus β on V.

We recommend that you read the proof of the next lemma, since the same style of argument occurs many times below.

Lemma 3.3.1. *v^* is increasing on W whenever P is monotone increasing.*

Proof. Let iV be the increasing functions in V and suppose that P is monotone increasing. The operator T is a self-map on iV in this setting, since $v \in iV$ implies $h(w) := c + \beta \sum_{w' \in \mathsf{W}} v(w')P(w,w')$ is in iV. Hence, for such a v, both h and the stopping value function $e(w) := w/(1-\beta)$ are in iV. It follows that $Tv = \max\{h, e\}$ is in iV.

Since iV is a closed subset of V and T is a self-map on iV, the fixed point v^* is in iV (see Exercise 1.2.15 on page 21). □

In view of the contraction property established in Exercise 3.3.1, we can use value function iteration to (i) compute an approximation v to the value function and (ii) calculate the v-greedy policy, which approximates the optimal policy. Code for implementing this procedure is shown in Listing 10. The definition of a v-greedy policy is analogous to that for the IID case (see (1.30) on page 35).

3.3.1.2 Continuation Values

The continuation value h^* from the IID case (see page 32) is now replaced by a **continuation value function**, given by

$$h^*(w) := c + \beta \sum_{w' \in \mathsf{W}} v^*(w')P(w,w') \qquad (w \in \mathsf{W}).$$

```
using QuantEcon, LinearAlgebra
include("s_approx.jl")

"Creates an instance of the job search model with Markov wages."
function create_markov_js_model(;
        n=200,       # wage grid size
        ρ=0.9,       # wage persistence
        ν=0.2,       # wage volatility
        β=0.98,      # discount factor
        c=1.0        # unemployment compensation
    )
    mc = tauchen(n, ρ, ν)
    w_vals, P = exp.(mc.state_values), mc.p
    return (; n, w_vals, P, β, c)
end

" The Bellman operator Tv = max{e, c + β P v} with e(w) = w / (1-β)."
function T(v, model)
    (; n, w_vals, P, β, c) = model
    h = c .+ β * P * v
    e = w_vals ./ (1 - β)
    return max.(e, h)
end

" Get a v-greedy policy."
function get_greedy(v, model)
    (; n, w_vals, P, β, c) = model
    σ = w_vals / (1 - β) .>= c .+ β * P * v
    return σ
end

"Solve the infinite-horizon Markov job search model by VFI."
function vfi(model)
    v_init = zero(model.w_vals)
    v_star = successive_approx(v -> T(v, model), v_init)
    σ_star = get_greedy(v_star, model)
    return v_star, σ_star
end
```

Listing 10: Job search with Markov state (`markov_js.jl`)

Figure 3.7: Value, stopping, and continuation for Markov job search

The continuation value depends on w because the current offer helps predict the offer next period, which in turn affects the value of continuing. The functions $w \mapsto w/(1-\beta)$, h^* and v^* corresponding to the default model in Listing 10 are shown in Figure 3.7.

Exercise 3.3.2. Explain why the continuation value function is increasing in Figure 3.7. If possible, provide a mathematical explanation and economic intuition.

Exercise 3.3.3. Using the Bellman equation (3.23), show that h^* obeys

$$h^*(w) := c + \beta \sum_{w' \in \mathsf{W}} \max\left\{\frac{w'}{1-\beta}, h^*(w')\right\} P(w, w') \qquad (w \in \mathsf{W}).$$

Exercise 3.3.4. Let Q be the operator on V defined at $h \in V$ by

$$(Qh)(w) := c + \beta \sum_{w' \in \mathsf{W}} \max\left\{\frac{w'}{1-\beta}, h(w')\right\} P(w, w') \qquad (w \in \mathsf{W}). \tag{3.24}$$

Prove that Q is (a) an order-preserving self-map on V and (b) a contraction of modulus β on V under the supremum norm.

Exercise 3.3.4 suggests to us a way to solve the job search problem without using

value function iteration: iterate with Q to obtain the continuation value function h^* and then use the policy

$$\sigma^*(w) = \mathbb{1}\left\{\frac{w'}{1-\beta} \geqslant h^*(w)\right\} \qquad (w \in \mathsf{W})$$

that tells the agent to accept when the current stopping value exceeds the current continuation value.

We saw in the IID case that a computational strategy based on continuation values is far more efficient than value function iteration (see §1.3.2.2). Here, however, the two approaches (iterating with T vs iterating with Q) are relatively similar, since continuation values are functions rather than scalars. In Chapter 4 we discuss alternative computational strategies in more detail, seeking conditions under which one approach will be more efficient than the other.

3.3.2 Job Search with Separation

As a final application for this chapter, we modify the job search problem discussed in §3.3.1 by adding one natural extension: separations occur. An existing match between worker and firm terminates with probability α every period. (The modification is an extension because, under $\alpha = 0$, we recover the permanent job scenario from §3.3.1.)

Once separation enters the picture, the agent comes to view the loss of a job as a capital loss, and a spell of unemployment as an investment in searching for an acceptable job. In what follows, the wage process and discount factor are unchanged from §3.3.1. As before, $V := \mathbb{R}_+^{\mathsf{W}}$ is endowed with the supremum norm.

The value function for an unemployed worker is denoted by v_u^*. This function satisfies the recursion

$$v_u^*(w) = \max\left\{v_e^*(w),\ c + \beta \sum_{w' \in \mathsf{W}} v_u^*(w')P(w,w')\right\} \qquad (w \in \mathsf{W}). \tag{3.25}$$

The function v_e^* that appears in this equation is the value function for employed agents. In particular, $v_e^*(w)$ is the lifetime value of obtained by an agent who starts the period employed at wage w. The function v_e^* satisfies the recursion

$$v_e^*(w) = w + \beta\left[\alpha \sum_{w'} v_u^*(w')P(w,w') + (1-\alpha)v_e^*(w)\right] \qquad (w \in \mathsf{W}). \tag{3.26}$$

This equation states that value accruing to an employed agent is current wage plus the discounted expected value next of being either employed or unemployed next period, weighted by their probabilities.

We claim that, when $0 < \alpha, \beta < 1$, the equations (3.25) and (3.26) both have unique solutions in V. To help prove this, we solve (3.26) in terms of $v_e^*(w)$ to obtain

$$v_e^*(w) = \frac{1}{1-\beta(1-\alpha)} \left(w + \alpha\beta(Pv_u^*)(w)\right). \tag{3.27}$$

(Recall $(Ph)(w) := \sum_{w'} h(w')P(w,w')$ for $h \in \mathbb{R}^{\mathsf{W}}$ and $w \in \mathsf{W}$.) Substituting into (3.25) yields

$$v_u^*(w) = \max\left\{\frac{1}{1-\beta(1-\alpha)} \left(w + \alpha\beta(Pv_u^*)(w)\right),\ c + \beta\,(Pv_u^*)(w)\right\}. \tag{3.28}$$

Exercise 3.3.5. Prove that there exists a unique $v_u^* \in V$ that solves (3.28). Propose a convergent method for solving for both v_u^* and v_e^*. [Hint: See Exercise 1.2.21 on page 23.]

Figure 3.8 shows the value function v_u^* for an unemployed worker, which is the fixed point of (3.28), as well as the stopping and continuation values, which are given by

$$s^*(w) := \frac{1}{1-\beta(1-\alpha)} \left(w + \alpha\beta(Pv_u^*)(w)\right) \quad \text{and} \quad h_e^*(w) := c + \beta\,(Pv_u^*)(w)$$

respectively, for each $w \in \mathsf{W}$. Parameters are as in Listing 11. The value function v_u^* is the pointwise maximum (i.e., $v_u^* = s^* \vee h^*$). The agent's optimal policy while unemployed is

$$\sigma^*(w) := \mathbb{1}\{s^*(w) \geqslant h^*(w)\}.$$

As before, the smallest w such that $\sigma^*(w) = 1$ is called the **reservation wage**.

Figure 3.9 shows how the reservation wage changes with α. To produce this figure we solved the model for the reservation wage at 10 values of α in an evenly spaced grid ranging 0 to 1. The reservation wage falls with α, since time spent unemployed is a capital investment in better wages, and the value of this investment declines as the separation rate rises.

Exercise 3.3.6. Replicate Figure 3.9.

```
using QuantEcon, LinearAlgebra

"Creates an instance of the job search model with separation."
function create_js_with_sep_model(;
        n=200,          # wage grid size
        ρ=0.9, ν=0.2,   # wage persistence and volatility
        β=0.98, α=0.1,  # discount factor and separation rate
        c=1.0)          # unemployment compensation
    mc = tauchen(n, ρ, ν)
    w_vals, P = exp.(mc.state_values), mc.p
    return (; n, w_vals, P, β, c, α)
end
```

Listing 11: Job search with separation model (`markov_js_with_sep.jl`)

Figure 3.8: Value function with job separation

Figure 3.9: The reservation wage vs separation rate

3.4 Chapter Notes

Many excellent textbooks on Markov chains exist, including Norris (1998), Häggström et al. (2002) and Privault (2013). Sargent and Stachurski (2022) provides a relatively comprehensive treatment from a network perspective. This perspective is a very natural one for Markov chains. More economic applications are discussed in Stokey and Lucas (1989) and Ljungqvist and Sargent (2018). Meyer (2000) gives a detailed account of the theory of nonnegative matrices. Another useful reference is Horn and Johnson (2012).

The systematic study of monotone Markov chains was initiated by Daley (1968). Monotone Markov methods have many important applications in economics. See, for example, Hopenhayn and Prescott (1992), Kamihigashi and Stachurski (2014), Jaśkiewicz and Nowak (2014), Balbus et al. (2014), Foss et al. (2018) and Hu and Shmaya (2019).

Chapter 4

Optimal Stopping

In this chapter we turn to the problem of maximizing lifetime rewards in a setting where agents must choosing when to act in the face of risk and uncertainty. The job search model studied in Chapters 1–3 is one such application. Others include if or when to exit or enter a market, default on a loan, exploit some new technology, or exercise a real or financial option.

All of these problems can be solved using dynamic programming. Moreover, they have common features that allow us to find sharp characterizations of optimality. Finally, they offer an excellent laboratory for studying dynamic programming because the binary choice (stop or continue) makes the recursive representations particularly clear.

4.1 Introduction to Optimal Stopping

In this section we begin with the standard theory of optimal stopping and then consider alternative approaches, based around continuation values and threshold policies. One key objective is to provide a rigorous discussion of optimality, which improves on our intuitive analysis in the context of job search in §1.3.

4.1.1 Theory

Our first step is to set out the fundamental theory of discrete time infinite-horizon optimal stopping problems.

4.1.1.1 The Stopping Problem

Let X be a finite set. An **optimal stopping problem** with state space X consists of

(i) a Markov operator $P \in \mathbb{M}(\mathsf{X})$,

(ii) a discount factor $\beta \in (0, 1)$,

(iii) a **continuation reward function** $c \in \mathbb{R}^{\mathsf{X}}$, and

(iv) an **exit reward function** $e \in \mathbb{R}^{\mathsf{X}}$.

Given a P-Markov chain $(X_t)_{t \geqslant 0}$, the problem evolves as follows: An agent observes the state X_t in each period and decides whether to continue or stop. If she chooses to stop, she receives $e(X_t)$ and the process terminates. If she decides to continue, then she receives $c(X_t)$ and the process repeats next period. Lifetime rewards are given by

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t R_t,$$

where R_t equals $c(X_t)$ while the agent continues, $e(X_t)$ when the agent stops, and zero thereafter.

Example 4.1.1. In the infinite-horizon job search problem from Chapter 1, the wage offer process (W_t) is IID with common distribution φ on finite set W, and the choice is between accepting the job offer and receiving unemployment compensation and waiting till next period. This is an optimal stopping problem with state space $\mathsf{X} = \mathsf{W}$ and $P \in \mathbb{M}(\mathsf{X})$ having all rows equal to φ, so that all draws are IID from φ. The exit reward function is $e(x) = x/(1 - \beta)$ and the continuation reward function is constant and equal to unemployment compensation.

Example 4.1.2. Consider an infinite-horizon American call option, which provides the right to buy a given asset at strike price K at each future point in time. The market price of the asset is given by $S_t = s(X_t)$, where (X_t) is P-Markov on finite set X. The interest rate is $r > 0$. The decision of when to exercise is an optimal stopping problem, with exit corresponding to exercise of the option. The discount factor is $1/(1+r)$, the exit reward function is $e(x) = s(x) - K$ and the continuation reward is zero.[1]

As for the job search problem, the actions of the agent will be expressed in terms of a **policy function**, which is a map σ from X to $\{0, 1\}$. The interpretation is that, on observing state x at any given time, the agent responds with action $\sigma(x)$, where 0

[1] We are studying American options in discrete time. Options with discrete exercise times are sometimes called **Bermudan options**. References for the continuous time case are provided in §4.3.

means "continue" and 1 means "stop." Implicit in this formulation is the assumption that the current state contains enough information for the agent to decide whether or not to stop.

Let Σ be the set of functions from X to $\{0, 1\}$. Let $v_\sigma(x)$ denote the expected lifetime value of following policy σ now and in every future period, given current state $x \in \mathsf{X}$. We call v_σ the **σ-value function**. Below, in §4.1.1.3, we show that v_σ is well defined and describe how to calculate it.

The function v_σ is an essential object in what follows, since our aim is to choose a policy that maximizes lifetime value. In particular, a policy $\sigma^* \in \Sigma$ is called **optimal** if

$$v_{\sigma^*}(x) = \max_{\sigma \in \Sigma} v_\sigma(x) \quad \text{for all } x \in \mathsf{X}. \tag{4.1}$$

4.1.1.2 Policy Valuation

Fixing $\sigma \in \Sigma$, let us think about how to pin down the σ-value function v_σ. Recall that $v_\sigma(x)$ is the lifetime value of following σ conditional on state x. Some thought will convince you that v_σ must satisfy

$$v_\sigma(x) = \sigma(x)e(x) + (1 - \sigma(x)) \left[c(x) + \beta \sum_{x' \in \mathsf{X}} v_\sigma(x')P(x, x') \right] \quad \text{for all } x \in \mathsf{X}. \tag{4.2}$$

To see this, suppose first that $\sigma(x) = 1$. In this case, (4.2) states that $v_\sigma(x) = e(x)$, which is what we expect: choosing to stop yields the exit reward. If instead $\sigma(x) = 0$, then (4.2) becomes

$$v_\sigma(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} v_\sigma(x')P(x, x'), \tag{4.3}$$

which is what we expect: the value of continuing is the current reward plus the discounted expected reward obtained by continuing with policy σ next period.

Now all that remains is to solve (4.2) for the function v_σ. To do this, we set

$$r_\sigma(x) := \sigma(x)e(x) + (1 - \sigma(x))c(x) \quad \text{and} \quad P_\sigma(x, x') := (1 - \sigma(x))P(x, x').$$

With this notation, we can write (4.2) pointwise as $v_\sigma = r_\sigma + \beta P_\sigma v_\sigma$. If $r(\beta P_\sigma) < 1$, then we have

$$v_\sigma = (I - \beta P_\sigma)^{-1} r_\sigma. \tag{4.4}$$

Exercise 4.1.1. Confirm that $r(\beta P_\sigma) < 1$ holds for any optimal stopping problem.

By Exercise 4.1.1 and the Neumann series lemma, v_σ is uniquely defined by (4.4).

4.1.1.3 Policy Operators

For the proofs below, it is helpful to view v_σ as the fixed point of a certain operator. In particular, we pair each $\sigma \in \Sigma$ with a corresponding **policy operator**, denoted by T_σ, and defined at $v \in \mathbb{R}^{\mathsf{X}}$ by

$$(T_\sigma v)(x) = \sigma(x)e(x) + (1 - \sigma(x)) \left[c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x') \right] \tag{4.5}$$

for each $x \in \mathsf{X}$. With this notation, (4.2) can be written as $v_\sigma = T_\sigma v_\sigma$.

Exercise 4.1.2. Prove that, for any $\sigma \in \Sigma$, the operator T_σ is order-preserving with respect to the pointwise partial order $\leqslant$ on $\mathbb{R}^{\mathsf{X}}$.

Using the notation in §4.1.1.2, we can also define T_σ via

$$T_\sigma v = r_\sigma + \beta P_\sigma v.$$

Proposition 4.1.1. *For any $\sigma \in \Sigma$, the policy operator T_σ is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ under to the supremum norm.*

The significance of Proposition 4.1.1 is as follows: By construction, v_σ is a fixed point of T_σ. By the contraction property in Proposition 4.1.1, v_σ is the only fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ and, moreover, iterates of T_σ always converge to v_σ.

Exercise 4.1.3. Prove Proposition 4.1.1.

4.1.1.4 The Value Function

In the job search problem, we found the optimal policy by computing the fixed point of the Bellman operator. Here we do the same while also explaining more carefully the relationship between optimality and the fixed point of the Bellman operator.

First we define the **value function** v^* of the optimal stopping problem as

$$v^*(x) := \max_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{4.6}$$

In particular, $v^*(x)$ is the maximal lifetime value available to an agent facing current state x. In view of the discussion in §1.2.4.3, we can also write $v^* = \vee_\sigma v_\sigma$.

How should we obtain the value function, given that solving the maximization in (4.6) is, in general, a hard problem? Our steps are as follows: We

(i) introduce the Bellman equation for the optimal stopping problem, which is

$$v(x) = \max\left\{e(x), c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x')\right\} \qquad (x \in \mathsf{X}), \tag{4.7}$$

(ii) prove that the Bellman equation has a unique solution in $\mathbb{R}^{\mathsf{X}}$, and, finally,

(iii) show that this solution equals the value function, as defined in (4.6).

These steps are completed in §4.1.1.5 below.

4.1.1.5 The Bellman Operator

The **Bellman operator** for the optimal stopping problem is the operator T such that any fixed point of T solves the Bellman equation and vice versa. This is true by construction for T defined by $v \mapsto Tv$,

$$(Tv)(x) = \max\left\{e(x),\ c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x')\right\} \qquad (x \in \mathsf{X}). \tag{4.8}$$

Pointwise, we can express T via $Tv = e \vee (c + \beta Pv)$.

Exercise 4.1.4. Prove that T is an order preserving self-map on $\mathbb{R}^{\mathsf{X}}$.

Here is the main result for this section:

Proposition 4.1.2. *For the optimal stopping problem defined in §4.1.1.1,*

(i) *T is a contraction map of modulus β on $\mathbb{R}^{\mathsf{X}}$, under the supremum norm $\|\cdot\|_\infty$ and*

(ii) *the unique fixed point of T on $\mathbb{R}^{\mathsf{X}}$ is the value function v^*.*

Exercise 4.1.5. Prove the claim in (i) of Proposition 4.1.2.

Proof of Proposition 4.1.2. With the result of Exercise 4.1.5 in hand, we need only show that the unique fixed point of T in $\mathbb{R}^{\mathsf{X}}$, denoted by $\bar{v}$, is equal to $v^* = \vee_\sigma v_\sigma$. We show $\bar{v} \leqslant v^*$ and then $\bar{v} \geqslant v^*$.

For the first inequality, let $\sigma \in \Sigma$ be defined by

$$\sigma(x) = \mathbb{1}\left\{e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x')\right\} \quad \text{for all } x \in \mathsf{X}.$$

Observe that, for this choice of σ, we have, for any $x \in \mathsf{X}$,

$$(T_\sigma \bar{v})(x) = \sigma(x)e(x) + (1 - \sigma(x)) \left[c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x') \right]$$
$$= \max \left\{ e(x),\ c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x') \right\} = (T\bar{v})(x) = \bar{v}(x).$$

In particular, $T_\sigma \bar{v} = T\bar{v} = \bar{v}$. But the only fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ is v_σ, so it must be the case that $\bar{v} = v_\sigma$. But then $\bar{v} \leqslant v^*$, by the definition of v^*. This is our first inequality.

Regarding the second inequality, fix $\sigma \in \Sigma$ and observe that $Tv \geqslant T_\sigma v$ for all $v \in \mathbb{R}^{\mathsf{X}}$. Since T is order-preserving and globally stable, Proposition 2.1.3 on page 52 implies that $v_\sigma \leqslant \bar{v}$. Taking the maximum over $\sigma \in \Sigma$ yields $v^* \leqslant \bar{v}$. □

4.1.1.6 Optimal Policies

Paralleling the definition provided in the discussion of job search (§1.3), for each $v \in \mathbb{R}^{\mathsf{X}}$, we call $\sigma \in \Sigma$ **v-greedy** if

$$\sigma(x) = \mathbb{1}\left\{ e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x') \right\} \quad \text{for all } x \in \mathsf{X}. \tag{4.9}$$

A v-greedy policy uses v to assign values to states and then chooses to stop or continue based on the action that generates a higher payoff.

With this language in place, our informal argument in §1.1.2 that optimal choices can be made using the value function becomes precise in the next proposition.

Proposition 4.1.3. *Policy $\sigma \in \Sigma$ is optimal if and only if it is v^*-greedy.*

Proposition 4.1.3 is a version of **Bellman's principle of optimality**. We prove this result in a more general setting in Chapter 5.

Corollary 4.1.4. *The optimal stopping problem has exactly one optimal policy.*

Proof. This follows directly from Proposition 4.1.3, since, given v^*, the greedy policy

$$\sigma^*(x) := \mathbb{1}\left\{ e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} v^*(x')P(x, x') \right\} \qquad (x \in \mathsf{X}) \tag{4.10}$$

is clearly uniquely defined. □

4.1.1.7 Value Function Iteration

The theory stated above tells us that successive approximation using the Bellman operator converges to v^* and v^*-greedy policies are optimal. These facts make value function iteration (VFI) a natural algorithm for solving optimal stopping problems. Since VFI for optimal stopping problems is directly analogous VFI for job search, as shown on page 36, we do not repeat it here.

4.1.2 Firm Valuation with Exit

In §3.2.3.2 we discussed firm valuation using expected present value of the cash flow generated by profits. This is a standard and well-used methodology. It does, however, ignore an important fact: firms have the option to cease operations and sell all remaining assets. In this section, we consider firm valuation in the presence of an exit option.

4.1.2.1 Optional Exit

Consider a firm where productivity is exogenous and evolves according to a Q-Markov chain (Z_t) on finite set $\mathsf{Z} \subset \mathbb{R}$. Profits are given by $\pi_t = \pi(Z_t)$ for some fixed $\pi \in \mathbb{R}^{\mathsf{Z}}$. At the start of each period, the firm decides whether to remain in operation, receiving current profit π_t, or to exit, receiving scrap value $s > 0$ for sale of physical assets. Discounting is at fixed rate r and $\beta := 1/(1+r)$. We assume that $r > 0$.

Let Σ be all $\sigma\colon \mathsf{Z} \to \{0, 1\}$. For given $\sigma \in \Sigma$ and $v \in \mathbb{R}^{\mathsf{Z}}$, the corresponding policy operator is

$$(T_\sigma v)(z) = \sigma(z)s + (1 - \sigma(z)\left[\pi(z) + \beta \sum_{z'} v(z')Q(z, z')\right] \qquad (z \in \mathsf{Z}).$$

We saw in §4.1.1.2–§4.1.1.3 that T_σ has a unique fixed point v_σ and that $v_\sigma(z)$ represents the value of following policy σ forever, conditional on $Z_0 = z$.

The Bellman operator for the firm's problem is the order-preserving self-map T on $\mathbb{R}^{\mathsf{Z}}$ defined by

$$(Tv)(z) = \max\left\{s, \pi(z) + \beta \sum_{z'} v(z')Q(z, z')\right\} \quad (z \in \mathsf{Z}).$$

Pointwise, T can be written as $Tv = s \vee (\pi + \beta Qv)$.

Figure 4.1: Value function for firms with exit option

Let v^* be the value function for this problem. By Proposition 4.1.2, v^* is the unique fixed point of T in $\mathbb{R}^{\mathsf{Z}}$ and the unique solution to the Bellman equation. Moreover, successive approximation from any $v \in \mathbb{R}^{\mathsf{Z}}$ converges to v^*. Finally, by Proposition 4.1.3, a policy is optimal if and only if it is v^*-greedy.

Figure 4.1 shows the value function v^*, along with the stopping value s and the continuation value function $h^* = \pi + \beta Q v^*$, under the parameterization given in Listing 12. As implied by the Bellman equation, v^* is the pointwise maximum of s and h^*. The v^*-greedy policy $\sigma^*(z) = \mathbb{1}\{s \geqslant h^*(z)\}$ instruct the firm to exit when the continuation value of the firm falls below the scrap value.

Exercise 4.1.6. Replicate Figure 4.1 by using the parameters in Listing 12 and applying value function iteration. Reviewing the code for job search on page 96 should be helpful.

4.1.2.2 Exit vs No-Exit

If we define w by $w(z) = \mathbb{E}_z \sum_{t \geqslant 0} \beta^t \pi_t$ for all $z \in \mathsf{Z}$, then $w(z)$ is the value of the firm given $Z_0 = z$ when the firm never exits. In other words, w evaluates the firm according to expected present value of the profit stream. Figure 4.2 shows w, denoted as the no-exit value, based on the parameterization in Listing 12.

In Figure 4.2, we see that $w \leqslant v^*$ on Z. Let's now prove that this is always true.

```
"Creates an instance of the firm exit model."
function create_exit_model(;
        n=200,                  # productivity grid size
        ρ=0.95, μ=0.1, ν=0.1,   # persistence, mean and volatility
        β=0.98, s=100.0         # discount factor and scrap value
    )
    mc = tauchen(n, ρ, ν, μ)
    z_vals, Q = mc.state_values, mc.p
    return (; n, z_vals, Q, β, s)
end
```

Listing 12: Firm exit model (`firm_exit.jl`)

Figure 4.2: Firm value with and without exit

To show $w \leqslant v^*$, first observe that $w = (I - \beta Q)^{-1}\pi$, by $\beta < 1$ and Lemma 3.2.3 on page 91. Rearranging gives $w = \pi + \beta Q w$.

Now note that under the policy $\sigma \equiv 0$, where the firm never chooses to exit, we have $T_\sigma v = \pi + \beta Q v$. Hence the unique fixed point of T_σ is w. As a result, $w = v_\sigma$ for $\sigma \equiv 0$. But $v^* \geqslant v_\sigma$ for all $\sigma \in \Sigma$. This proves that $w \leqslant v^*$.

In terms of intuition, choosing to never exit is a feasible policy. Since v^* involves maximization of firm value over the set of all feasible policies, it must be at least as large.

Exercise 4.1.7. Prove the following: If $Q \gg 0$ and $s > w(z)$ for at least one $z \in \mathsf{Z}$, then $w \ll v^*$. Provide some intuition for this result.

Exercise 4.1.8. Consider a version of the model of firm value with exit where productivity is constant but prices are stochastic. In particular, the prices process (P_t) for the final good is Q-Markov. Suppose further that one-period profits for a given price p are $\max_{\ell \geqslant 0} \pi(\ell, p)$, where ℓ is labor input. Suppose that $\pi(\ell, p) = p\ell^{1/2} - w\ell$, where the wage rate w is constant. Write down the Bellman equation for this model.

4.1.3 Monotonicity

In this section we consider monotonicity in values and actions. In doing so, we return to the general optimal stopping problem described in §4.1.1, with X as the state space, e as the exit reward function and c as the continuation reward function.

4.1.3.1 Monotone Values

Let v^* be the value function of the optimal stopping problem defined by X, P, β, c and e. We define the corresponding **continuation value function** h^* to be

$$h^*(x) := c(x) + \beta \sum_{x' \in \mathsf{X}} v^*(x') P(x, x') \qquad (x \in \mathsf{X}). \tag{4.11}$$

(Please be sure to avoid confusing the continuation reward function c and the continuation value function h^*.)

Let X be partially ordered and let $i\mathbb{R}^{\mathsf{X}}$ be the increasing functions in $\mathbb{R}^{\mathsf{X}}$.

Lemma 4.1.5. *If $e, c \in i\mathbb{R}^{\mathsf{X}}$ and P is monotone increasing, then h^* and v^* are both increasing.*

Proof. Let the stated conditions hold. The Bellman operator can be written pointwise as $Tv = e \vee (c + \beta Pv)$. Since P is monotone increasing, P is invariant on $i\mathbb{R}^{\mathsf{X}}$. It follows from this fact and the conditions on e and c that T is invariant on $i\mathbb{R}^{\mathsf{X}}$. Hence, by Exercise 1.2.15 on page 21, v^* is in $i\mathbb{R}^{\mathsf{X}}$. Since $h^* = c + \beta Pv^*$, the same is true for h^*. □

Example 4.1.3. Consider the firm problem with exit, as described in §4.1.2, with Bellman operator $Tv = s \vee (\pi + \beta Qv)$. Since s is constant, it follows directly that v^* and h^* are both increasing functions when $\pi \in i\mathbb{R}^{\mathsf{Z}}$ and Q is monotone increasing.

4.1.3.2 Monotone Actions

The optimal policy in the IID job search problem takes the form $\sigma^*(w) = \mathbb{1}\{w \geqslant w^*\}$ for all $w \in \mathsf{W}$, where $w^* := (1-\beta)h^*$ is the reservation wage and h^* is the continuation value (see page 35). This optimal policy is of threshold type: once the wage offer exceeds the threshold, the agent always stops.

Since threshold policies are convenient, let us now try to characterize them.

Throughout this section, we take X to be a subset of $\mathbb{R}$. Elements of X are ordered by $\leqslant$, the usual order on $\mathbb{R}$.

Exercise 4.1.9. Prove that the optimal policy σ^* is decreasing on X whenever e is decreasing on X and h^* is increasing on X.

For a binary function on $\mathsf{X} \subset \mathbb{R}$, the condition that σ^* is decreasing means that the controller exists when x is sufficiently small and continues otherwise.

Example 4.1.4. In the firm problem with exit, as described in §4.1.2, h^* is increasing whenever $\pi \in i\mathbb{R}^{\mathsf{Z}}$ and Q is monotone increasing. Since the scrap value is constant, Exercise 4.1.9 applies under these conditions. Hence the optimal policy is decreasing. This reasoning agrees with Figure 4.1, where exit is optimal when the state is small and continuing is optimal when z is large. This makes sense, since Q is monotone increasing, so low current values of z predict low future values of z (and the profits associated with continuing will also be low).

Exercise 4.1.10. Show that the conditions of Exercise 4.1.9 hold when e is constant on X, c is increasing on X and P is monotone increasing.

Exercise 4.1.11. Prove that the optimal policy σ^* is increasing on X whenever e is increasing on X and h^* is decreasing on X.

Example 4.1.5. In the IID job search problem, $e(w) = w/(1-\beta)$ is increasing and h^* is constant. Hence the result in Exercise 4.1.11 applies. This is why the optimal policy $\sigma^*(w) = \mathbb{1}\{w \geqslant (1-\beta)h^*\}$ is increasing. The agent accepts all sufficiently large wage offers.

In the settings of Exercises 4.1.9–4.1.11, the optimal policy is either increasing or decreasing. Since X is totally ordered, monotonicity implies that the policy is of threshold type. For example, if σ^* is increasing, then we take x^* to be the smallest $x \in \mathsf{X}$ such that $\sigma^*(x) = 1$. For such an x^* we have

$$x < x^* \implies \sigma^*(x) = 0 \quad \text{and} \quad x \geqslant x^* \implies \sigma^*(x) = 1.$$

Remark 4.1.1. The conditions in Exercises 4.1.9–4.1.11 are sufficient but not necessary for monotone policies. For example, Figure 3.7 on 97 provides an example of a setting where the policy is increasing (the agent accepts for sufficiently large wage offers) even though both $e(x) = x/(1-\beta)$ and h^* are strictly increasing.

4.1.4 Continuation Values

In §1.3.2.2 we used a "continuation value" approach to solving the job search problem with IID draws, which involved computing the continuation value h^* directly and then setting the optimal policy to $\sigma^*(w) = \mathbb{1}\{w/(1-\beta) \geqslant h^*\}$. We saw that this approach is more efficient than first computing the value function, since the continuation value is one-dimensional rather than $|\mathsf{W}|$-dimensional.

In §3.3.1.2, we tried the same approach for the job search problem with Markov state, where wage draws are correlated. We found that there is no clear benefit to the continuation value approach in that setting, since the continuation value function has the same dimensionality as the value function.

These observations motivate us to explore continuation value methods more carefully. In this section, we state the continuation value approach for the general optimal stopping problem and verify convergence. We will see that, while all relevant state components must be included in the value function, purely transitory components do not affect continuation values. Hence the continuation value approach is at least as efficient and sometimes radically more so.

Another asymmetry between value functions and continuation value functions is that the latter are typically smoother. For example, in job search problems, the value function is usually kinked at the reservation wage, while the continuation value function is smooth. Relative smoothness comes from taking expectations over stochastic

transitions, since integration is a smoothing operation. Like lower dimensionality, increased smoothness helps with both analysis and computation.

4.1.4.1 Useful Techniques

Let h^* be the continuation value function for the optimal stopping problem, as defined in (4.11). To compute h^* directly we begin with the optimal stopping version of the Bellman equation evaluated at v^* and rewrite it as

$$v^*(x') = \max\{e(x'), h^*(x')\} \qquad (x' \in \mathsf{X}). \tag{4.12}$$

Taking expectations of both sides of this equation conditional on current state x produces $\sum_{x' \in \mathsf{X}} v^*(x')P(x,x') = \sum_{x' \in \mathsf{X}} \max\{e(x'), h^*(x')\} P(x,x')$. Multiplying by β, adding $c(x)$, and using the definition of h^*, we get

$$h^*(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} \max\{e(x'), h^*(x')\} P(x,x') \qquad (x \in \mathsf{X}). \tag{4.13}$$

This expression motivates us to introduce the **continuation value operator** $C \colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{X}}$ via

$$(Ch)(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} \max\{e(x'), h(x')\} P(x,x') \qquad (x \in \mathsf{X}). \tag{4.14}$$

Proposition 4.1.6. *The operator C is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$. Moreover, the unique fixed point of C in $\mathbb{R}^{\mathsf{X}}$ is h^*.*

Proposition 4.1.6 provides us with an alternative method to compute the optimal policy, which does not involve value function iteration:

(i) Use successive approximation with C to compute h^* (at least approximately) and

(ii) Calculate σ^* via $\sigma^*(x) = \mathbb{1}\{e(x) \geqslant h^*(x)\}$ for each $x \in \mathsf{X}$.

In §4.1.4.2 we discuss settings where this approach is advantageous.

Proof of Proposition 4.1.6. Fix $f, g \in \mathbb{R}^{\mathsf{X}}$ and $x \in \mathsf{X}$. By the triangle inequality and the bound $|\alpha \vee x - \alpha \vee y| \leqslant |x - y|$ from page 33, we have

$$\begin{aligned} |(Cf)(x) - (Cg)(x)| &\leqslant \beta \sum_{x' \in \mathsf{X}} |\max\{e(x'), f(x')\} - \max\{e(x'), g(x')\}| \, P(x,x') \\ &\leqslant \beta \sum_{x' \in \mathsf{X}} |f(x') - g(x')| \, P(x,x'). \end{aligned}$$

The right-hand side is dominated by $\beta\|f-g\|_\infty$. Taking the maximum on the left-hand side gives

$$\|Cf - Cg\|_\infty \leqslant \beta\|f - g\|_\infty,$$

which confirms that C is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$.

From the contraction property, we know that C has exactly one fixed point in $\mathbb{R}^{\mathsf{X}}$. Let $\bar{h}$ be this function. We claim that $\bar{h} = h^*$.

Let $\bar{v} := e \vee \bar{h}$. (We use functional notation here and for the rest of the proof, so that operations and relations are all pointwise.) To show that $\bar{h} = h^*$, it suffices to show that $\bar{v} = v^*$. Indeed, if $\bar{v} = v^*$, then

$$\bar{h} = C\bar{h} = c + \beta P(e \vee \bar{h}) = c + \beta P\bar{v} = c + \beta P v^* = h^*.$$

To see that $\bar{v} = v^*$, we again use $\bar{h} = C\bar{h}$ to obtain

$$e \vee \bar{h} = e \vee (C\bar{h}) = e \vee \left[c + \beta P(e \vee \bar{h})\right].$$

Using $\bar{v} = e \vee \bar{h}$, we can write this as $\bar{v} = e \vee [c + \beta P\bar{v}]$. This is just the Bellman equation in functional notation. As v^* is the only solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$ (Proposition 4.1.3), we have $\bar{v} = v^*$, as claimed. □

4.1.4.2 Dimensionality Reduction

In the discussion at the start of §4.1.4, we mentioned that switching from value function iteration to continuation value iteration can greatly reduce the dimensionality of the problem in some cases. Here we try to pin down the cases where this works.

To begin, let W and Z be two finite sets and suppose that $\varphi \in \mathscr{D}(\mathsf{W})$ and $Q \in \mathbb{M}(\mathsf{Z})$. Let (W_t) be IID with distribution φ and let (Z_t) be an Q-Markov chain on Z. If (W_t) and (Z_t) are independent, then (X_t) defined by $X_t = (W_t, Z_t)$ is P-Markov on X, where

$$P(x, x') = P((w, z), (w', z')) = \varphi(w')Q(z, z').$$

Suppose that the continuation reward depends only on z. In this case, we can write the Bellman operator as

$$(Tv)(w, z) = \max\left\{e(w, z),\ c(z) + \beta \sum_{w' \in \mathsf{W}} \sum_{z' \in \mathsf{Z}} v(w', z')\varphi(w')Q(z, z')\right\}. \tag{4.15}$$

Since the right-hand side depends on both w and z, the Bellman operator acts in an n-dimensional space, where $n := |\mathsf{X}| = |\mathsf{W}| + |\mathsf{Z}|$.

However, if we inspect the right-hand side of (4.15), we see that the continuation value function depends only on z. Dependence on w is absent because w does not help predict w'. Thus, the continuation value function is an object in $|\mathsf{Z}|$-dimensional space. The continuation value operator

$$(Ch)(z) = c(z) + \beta \sum_{w' \in \mathsf{X}} \sum_{z' \in \mathsf{X}} \max\{e(w', z'), h(z')\} \, \varphi(w') Q(z, z') \qquad (z \in \mathsf{Z}) \tag{4.16}$$

acts in this lower dimensional-space.

Example 4.1.6. We can embed the IID the job search problem into this setting by taking (W_t) to be the wage offer process and (Z_t) to be constant. This is why the IID case offers a large dimensionality reduction when we switch to continuation values.

More examples of dimensionality reduction are shown in the applications below.

4.1.4.3 Application to Firm Value

Consider the firm valuation problem from §4.1.2 but suppose now that scrap value fluctuates over time, according to the prices of the underlying assets. For simplicity let's assume that scrap value at each time t is given by the IID sequence (S_t), where each S_t has density φ on $\mathbb{R}_+$. The corresponding Bellman operator is

$$(Tv)(z, s) = \max\left\{s, \pi(z) + \beta \sum_{z'} \int v(z', s') \varphi(s') \, \mathrm{d}s' Q(z, z')\right\}.$$

We can convert this problem to a finite state space optimal stopping problem by discretizing the distribution φ onto a finite grid contained in $\mathbb{R}_+$. However, a better approach is to switch to the continuation value operator, since continuation values depend only on z.

Exercise 4.1.12. Write down the continuation value operator for this function as a mapping from $\mathbb{R}^{\mathsf{Z}}$ to itself.

Exercise 4.1.13. In §3.2.2.2 we defined stochastic dominance for distributions on finite sets. For densities φ and ψ on $\mathbb{R}_+$, the definition is similar: we say that ψ stochastically dominates φ and write $\varphi \preceq_{\mathrm{F}} \psi$ if $\int u(x)\varphi(x) \, \mathrm{d}x \leqslant \int u(x)\psi(x) \, \mathrm{d}x$ for every u in $i\mathbb{R}^{\mathsf{X}}$.[2] With this definition, show that if φ_a and φ_b are two alternative

[2]Actually, in most definitions, u is also restricted to be bounded and measurable, in order to ensure that the integrals are finite. These technicalities can be ignored in the exercise.

distributions for scrap value and $\varphi_a \preceq_{\mathrm{F}} \varphi_b$, then $\sigma_a^* \geqslant \sigma_b^*$ pointwise on Z, where σ_i^* is the optimal policy corresponding to distribution φ_i for $i \in \{a, b\}$. Interpret this result.

4.2 Further Applications

In this section we discuss some further applications of optimal stopping and apply the results described above.

4.2.1 American Options

American options were introduced briefly in Example 4.1.2 on page 103. Here we investigate this class of derivatives more carefully. We focus on American call options, which provide the right to buy a given asset (e.g., 1,000 shares in some underlying equity) at any time during some specified period at some fixed **strike price** K. The market price of the asset at time t is denoted by S_t.

The infinite horizon case was discussed in Example 4.1.2. However, options without termination dates—also called perpetual options—are rare in practice. Hence we focus on the finite-horizon case. We are interested in computing the expected value of holding the option when discounting with a fixed interest rate. This is a standard approach to pricing American options.

Finite horizon American options can be solved by backward induction, analogous to the finite horizon job search problem discussed in Chapter 1. Alternatively, we can embed finite horizon options into the theory of infinite-horizon optimal stopping. This second approach is convenient for us, since the theory of infinite-horizon optimal stopping has already been presented.

To perform this embedding, we take $T \in \mathbb{N}$ to be a fixed integer indicating the date of expiration. The option is purchased at $t = 0$ and can be exercised at $t \in \mathbb{N}$ with $t \leqslant T$. To include t in the current state, we set

$$\mathsf{T} := \{1, \ldots, T+1\} \quad \text{and} \quad m(t) := \min\{t+1, T+1\} \text{ for all } t \in \mathsf{T}.$$

The idea is that time is updated via $t' = m(t)$, so that time increments at each update until $t = T + 1$. After that we hold t constant. Bounding time at $T + 1$ keeps the state space finite.

We assume that the stock price S_t evolves according to

$$S_t = Z_t + W_t \quad \text{where} \quad (W_t)_{t \geqslant 0} \overset{\text{IID}}{\sim} \varphi \in \mathscr{D}(\mathsf{W}).$$

Here $(Z_t)_{t \geqslant 0}$ is Q-Markov on finite set Z for some $Q \in \mathbb{M}(\mathsf{Z})$ and W is also finite. This means that the share price is affected by a persistent and purely transient component. We choose parameters such that $(Z_t)_{t \geqslant 0}$ is close to a random walk, implying that price changes are difficult to predict.

To convert these update rules into an optimal stopping problem, as defined in §4.1.1.1, we need to specify the state and clarify the $P \in \mathbb{M}(\mathsf{X})$ that maps to the state process. We set the state space to $\mathsf{X} := \mathsf{T} \times \mathsf{W} \times \mathsf{Z}$ and

$$P((t, w, z), (t', w', z')) := \mathbb{1}\{t' = m(t)\} \varphi(w') Q(z, z').$$

In other words, time updates deterministically via $t' = m(t)$ and z' and w' are drawn independently from $Q(z, \cdot)$ and φ respectively.

As in the perpetual option case, the continuation reward is zero and the discount rate is $\beta := 1/(1 + r)$, where $r > 0$ is a fixed risk-free rate. The exit reward can be expressed as $\mathbb{1}\{t \leqslant T\}(S_t - K)$. In other words, exercise at time t earns the owner $S_t - K$ up to expiry and zero thereafter. In terms of the state (t, z), the exit reward is

$$e(t, w, z) := \mathbb{1}\{t \leqslant T\}[z + w - K].$$

The Bellman equation can be written as

$$v(t, w, z) = \max \left\{ e(t, w, z),\ \beta \sum_{w'} \sum_{z'} v(t', w', z') \varphi(w') Q(z, z') \right\},$$

where $t' = m(t)$. This relationship neatly captures the value of the option: It is the maximum of current exercise value and the discounted expected value of carrying the option over to the next period.

Since the problem described above is an optimal stopping problem in the sense of §4.1.1.1, all of the optimality results described above apply. In particular, iterates of the Bellman operator converge to the value function v^* and, moreover, a policy is optimal if and only if it is v^*-greedy.

We can do better than value function iteration. Since $(W_t)_{t \geqslant 0}$ is IID and appears only in the exit reward, we can reduce dimensionality by switching to the continuation

```julia
using QuantEcon, LinearAlgebra, IterTools

"Creates an instance of the option model with log S_t = Z_t + W_t."
function create_american_option_model(;
        n=100, μ=10.0,  # Markov state grid size and mean value
        ρ=0.98, ν=0.2,  # persistence and volatility for Markov state
        s=0.3,          # volatility parameter for W_t
        r=0.01,         # interest rate
        K=10.0, T=200)  # strike price and expiration date
    t_vals = collect(1:T+1)
    mc = tauchen(n, ρ, ν)
    z_vals, Q = mc.state_values .+ μ, mc.p
    w_vals, φ, β = [-s, s], [0.5, 0.5], 1 / (1 + r)
    e(t, i_w, i_z) = (t ≤ T) * (z_vals[i_z] + w_vals[i_w] - K)
    return (; t_vals, z_vals, w_vals, Q, φ, T, β, K, e)
end
```

Listing 13: Pricing and American option (`american_option.jl`)

value operator, which, in this case, can be expressed as

$$(Ch)(t,z) = \beta \sum_{z'} \sum_{w'} \max\left\{e(t',w',z'),\, h(t',z')\right\} \varphi(w')Q(z,z'). \tag{4.17}$$

As proved in §4.1.4, the unique fixed point of C is the continuation value function h^*, and $C^k h \to h^*$ as $k \to \infty$ for all $h \in \mathbb{R}^{\mathsf{X}}$. With the fixed point in hand, we can compute the optimal policy as

$$\sigma^*(t,w,z) = \mathbb{1}\left\{e(t,w,z) \geqslant h^*(t,z)\right\}.$$

Here $\sigma^*(t,w,z) = 1$ indicates exercise of the option at time t.

Figure 4.3 provides a visual representation of optimal actions under the default parameterization described in Listing 13. Each of the three figures show contour lines of the net exit reward $f(t,w,z) := e(t,w,z) - h^*(w,z)$, viewed as a function of (w,z), when t is held fixed. The date t for each subfigure is shown in the title. The optimal policy exercises the option when $f(t,w,z) \geqslant 0$.

In each subfigure, the **exercise region**, which is the set (w,z) such that $f(t,w,z) \geqslant 0$, correspond to the northeast part of the figure, where w and z are both large. The boundary between exercise and continuing is the zero contour line, which is shown in black. Notice that the size of the exercise region expands with t. This is because

the value of waiting decreases when the set of possible exercise dates declines.

Figure 4.4 provides some simulations of the stock price process $(S_t)_{t \geqslant 0}$ over the lifetime of the option, again using the default parameterization described in Listing 13. The blue region in the top part of each subfigure is the values of the stock price $S_t = Z_t + W_t$ such that $S_t \geqslant K$. An option traded in this configuration (where the price of the underlying exceeds the strike price) is said to be "in the money." The figure also shows the optimal exercise date in each of the simulations, which is the first t such that $e(t, W_t, Z_t) \geqslant h^*(W_t, Z_t)$.

4.2.2 Research and Development

Consider a firm that engages in costly research and development (R&D) in order to develop a new product. The dynamic problem faced by the firm is whether to hold back and continue investing in the project or stop and bring the product to market. For simplicity, we assume here that the value of bringing the product to market is a one-off payoff $\pi_t = \pi(X_t)$, where (X_t) is P-Markov chain on finite set X with $P \in \mathbb{M}(\mathsf{X})$. The flow cost of investing in R&D is C_t per period, where (C_t) is a stochastic process. Future payoffs are discounted at rate $r > 0$ and we set $\beta := 1/(1+r)$.

4.2.2.1 Constant R&D Costs

As a first take on this problem, suppose that $C_t \equiv c \in \mathbb{R}_+$ for all t. This is an optimal stopping problem with exit reward $e = \pi$ and constant continuation reward $-c$. The Bellman equation for this problem is

$$v(x) = \max \left\{ \pi(x), -c + \beta \sum_{x'} v(x') P(x, x') \right\} \qquad (x \in \mathsf{X}). \tag{4.18}$$

Exercise 4.2.1. Write down the continuation value operator for this problem. Prove that the continuation value function h^* is increasing in x whenever $\pi \in i\mathbb{R}^{\mathsf{X}}$ and P is monotone increasing.

Exercise 4.2.2. Prove that the optimal policy σ^* is increasing whenever π is increasing and (X_t) is IID (so that all rows of P are identical). Provide economic intuition for this result.

Figure 4.3: Exercise region for the American option

Figure 4.4: Simulations for the American option process

4.2.2.2 IID R&D Costs

Let's suppose now that $(C_t)_{t\geqslant 0}$ is IID with common distribution $\varphi \in \mathscr{D}(\mathsf{W})$. The Bellman equation becomes

$$v(c,x) = \max\left\{\pi(x), -c + \beta \sum_{x'}\sum_{c'} v(c',x')\varphi(c')P(x,x')\right\}. \tag{4.19}$$

Since (C_t) is IID, we would ideally like to integrate it out in the manner of §4.1.4.2, thereby lowering the dimensionality of the problem. However, if we look at the continuation value associated with (4.19), we get

$$h(c,x) := -c + \beta \sum_{x'}\sum_{c'} v(c',x')\varphi(c')P(x,x'),$$

which still depends on c.

Fortunately, with a bit more thought, we can find a way to eliminate c. To this end, we define

$$g(x) := \sum_{x'}\sum_{c'} v(c',x')\varphi(c')P(x,x'), \tag{4.20}$$

which is the expected discounted value in state x. Rewriting the Bellman equation using g and replacing (c,x) with (c',x') gives

$$v(c',x') = \max\left\{\pi(x'), -c' + \beta g(x')\right\}.$$

Averaging over (c',x') and using the definition of g again gives

$$g(x) = \sum_{x'}\sum_{c'} \max\left\{\pi(x'), -c' + \beta g(x')\right\}\varphi(c')P(x,x'). \tag{4.21}$$

This is a functional equation in g, which depends only on x. To solve it, we introduce the operator R defined by

$$(Rg)(x) = \sum_{x'}\sum_{c'} \max\left\{\pi(x'), -c' + \beta g(x')\right\}\varphi(c')P(x,x') \quad (x \in \mathsf{X}).$$

Exercise 4.2.3. Prove that R is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$.

From Exercise 4.2.3, we see that (4.21) has a unique solution in $\mathbb{R}^{\mathsf{X}}$, which we denote by g^*, and that g^* can be computed by successive approximation. With g^* in

hand, we can compute the optimal policy via

$$\sigma^*(c, x) = \mathbb{1}\left\{\pi(x), -c + \beta g^*(x)\right\}.$$

Remark 4.2.1. The technique we just used works by solving for the expected value function, as defined in (4.20). In §5.3 we analyze this method again in a more general setting and discuss its convergence properties.

4.3 Chapter Notes

Numerous textbooks treat optimal stopping in depth, although most use continuous time. Peskir and Shiryaev (2006) and Shiryaev (2007) are useful examples.

There are many applications of optimal stopping in economics and finance, with influential early research papers including McCall (1970), Jovanovic (1982), Hopenhayn (1992), and Ericson and Pakes (1995). In more recent literature, Arellano (2008) considers borrowing on international financial markets with the option of sovereign default (see §8.2.1.5). Riedel (2009) studies optimal stopping under Knightian uncertainty. Fajgelbaum et al. (2017) include an optimal stopping problem for firms in a model of uncertainty traps.

The firm problem with optimal exit is routinely used to analyze firm dynamics and firm size distributions in equilibrium models with heterogeneous firms. Hopenhayn (1992) is a classic reference. Perla and Tonetti (2014) study a variation where growth is created by firms at the bottom of the productivity distribution imitating more productive firms. Carvalho and Grassi (2019) analyze business cycles in a setting of firm growth with exit and a Pareto distribution of firms.

Regarding American options, the perpetual case is studied in Mordecki (2002). Practical methods for solving American option models are provided in Longstaff and Schwartz (2001), Rogers (2002), and Kohler et al. (2010).

Replacement problems are an important kind of optimal stopping problem not treated in this chapter. A classic early example is the paper by Rust (1987), which uses dynamic programming to consider optimal replacement of engine parts and structural estimation to fit parameters. Some discussion of structural estimation is provided in §5.3.1.

Chapter 5

Markov Decision Processes

In this chapter we study a class of discrete time, infinite horizon dynamic programs called Markov decision processes (MDPs). This class of problems is broad enough to encompass many applications, including the optimal stopping problems in Chapter 4. MDPs can also be combined with reinforcement learning to tackle settings where important inputs to an MDP are not known.

5.1 Definition and Properties

In this section we define MDPs and investigate optimality.

5.1.1 The MDP Model

MDPs are dynamic programs characterized by two features: rewards are additively separable and the discount rate is constant. (Additive separability was defined and discussed in §7.2.)

We require the following definition: A **correspondence** Γ from one set X to another set A is a function from X into $\wp(\mathsf{A})$, the set of all subsets of A. The correspondence is called **nonempty** if $\Gamma(x) \neq \emptyset$ for all $x \in \mathsf{X}$. For example, the map Γ defined by $\Gamma(x) = [-x, x]$ is a nonempty correspondence from $\mathbb{R}$ to $\mathbb{R}$.

We study a controller who interacts with a state process $(X_t)_{t\geqslant 0}$ by choosing an action path $(A_t)_{t\geqslant 0}$ to maximize expected discounted rewards

$$\mathbb{E} \sum_{t\geqslant 0} \beta^t r(X_t, A_t), \tag{5.1}$$

taking an initial state X_0 as given. As with all dynamic programs, we insist that the controller is not clairvoyant: he or she cannot choose actions that depend on future states.

To formalize the problem, we fix a finite set X, henceforth called the **state space**, and a finite set A, henceforth called the **action space**. Given X and A, we define a (finite) **Markov decision process** (**MDP**) to be a tuple (Γ, β, r, P) consisting of

(i) a nonempty correspondence Γ from $\mathsf{X} \to \mathsf{A}$, referred to as the **feasible correspondence**, which in turn defines the **feasible state-action pairs**

$$\mathsf{G} := \{(x, a) \in \mathsf{X} \times \mathsf{A} : a \in \Gamma(x)\},$$

(ii) a constant β in $(0, 1)$, referred to as the **discount factor**,

(iii) a function r from G to $\mathbb{R}$, referred to as the **reward function**, and

(iv) a **stochastic kernel** P from G to X; that is, P is a map from $\mathsf{G} \times \mathsf{X}$ to $\mathbb{R}_+$ satisfying

$$\sum_{x' \in \mathsf{X}} P(x, a, x') = 1 \quad \text{for all } (x, a) \text{ in } \mathsf{G}.$$

The feasible correspondence restricts actions, in the sense that $\Gamma(x) \subset \mathsf{A}$ is the set of actions available to the controller in state x. Given a feasible state-action pair (x, a), reward $r(x, a)$ is received and the next period state x' is selected from $P(x, a, \cdot)$, which is an element of $\mathscr{D}(\mathsf{X})$. The dynamics and reward flow are summarized in Algorithm 3.

Algorithm 3: MDP dynamics: states, actions, and rewards

```
t ← 0
input X_0
while t < ∞ do
    observe X_t
    choose action A_t
    receive reward r(X_t, A_t)
    draw X_{t+1} from P(X_t, A_t, ·)
    t ← t + 1
end
```

The **Bellman equation** for with this problem is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \right\} \quad \text{for all } x \in \mathsf{X}. \tag{5.2}$$

This can be understood as an equation in the unknown function $v \in \mathbb{R}^{\mathsf{X}}$. Below we define the value function v^* as maximal lifetime rewards and show that v^* is the unique solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$.

As for optimal stopping, the Bellman equation reduces an infinite-horizon problem to a two period problem. Current actions influence the two terms on the right hand side: current rewards and expected discounted value from future states. In every case we examine, there is a trade-off between maximizing current rewards and influencing the distribution $P(x, a, \cdot)$ of the next period state in order to obtain high future rewards.

5.1.2 Examples

Many dynamic optimization models can be framed as MDPs. Some models natural fit the MDP structure, while others have to be coerced into the MDP framework by adding states or applying other tricks. (In Chapter 8 we provide a more general framework where these extra steps are unnecessary. At the same time, the MDP model is the most important special case of this general framework, which is why we invest time in understanding its strengths and limitations.)

5.1.2.1 A Renewal Problem

Rust (1987) ignited the field of dynamic structural estimation by examining an engine replacement problem for a bus workshop. In each period the superintendent decides whether or not to replace the engine of a given bus. Early replacement is costly but late replacement risks unexpected failure. This trade-off is solved via dynamic programming.

We consider an abstract version of Rust's problem involving maximization of rewards (rather than minimization of costs). The action A_t at time t is binary (e.g., replace or don't replace). When $A_t = 1$, the state resets to some fixed **renewal state** $\bar{x}$ in a finite set X (e.g., mileage resets to zero when an engine is replaced). When $A_t = 0$, the state updates according to $Q \in \mathbb{M}(\mathsf{X})$ (e.g., mileage increases stochastically when the engine is not replaced). Given current state x and action a, current reward $r(x, a)$ is received. The discount factor is $\beta \in (0, 1)$.

For this problem, the Bellman equation has the form

$$v(x) = \max \left\{ r(x, a) + \beta v(\bar{x}),\ r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') Q(x, x') \right\}. \tag{5.3}$$

To set the problem up as an MDP we set $\mathsf{A} = \{0, 1\}$ and $\Gamma(x) = \mathsf{A}$ for all $x \in \mathsf{X}$. We define

$$P(x, a, x') := a\mathbb{1}\{x' = \bar{x}\} + (1 - a)Q(x, x') \qquad ((x, a) \in \mathsf{G},\ x' \in \mathsf{X}). \tag{5.4}$$

Exercise 5.1.1. Prove that P is a stochastic kernel from G to X.

The primitives (Γ, β, r, P) form an MDP with a Bellman equation that agrees with (5.3). Indeed, using $\Gamma(x) = \{0, 1\}$ and P from (5.4), the MDP Bellman equation (5.2) can be rewritten as

$$v(x) = \max_{a \in \{0,1\}} \left\{ r(x, a) + \beta \left[av(\bar{x}) + (1 - a) \sum_{x' \in \mathsf{X}} v(x')Q(x, x') \right] \right\}.$$

Considering the cases $a = 0$ and $a = 1$ clarifies that this is identical to (5.3).

5.1.2.2 Optimal Inventory Management

We study a firm where a manager tries to maximize shareholder value. To simplify the problem, we ignore exit options (so that firm value is the expected present value of profits) and assume that the firm only sells one product. Letting π_t be profits at time t and $r > 0$ be the interest rate, the value of the firm is

$$V_0 = \mathbb{E} \sum_{t \geqslant 0} \beta^t \pi_t \qquad \text{where} \quad \beta := \frac{1}{1 + r}. \tag{5.5}$$

Suppose the firm faces exogenous demand process $(D_t)_{t \geqslant 0}$. We assume $(D_t)_{t \geqslant 0}$ is IID with common distribution $\varphi \in \mathscr{D}(\mathbb{Z}_+)$. Inventory $(X_t)_{t \geqslant 0}$ of the product obeys

$$X_{t+1} = m(X_t - D_{t+1}) + A_t \qquad \text{where} \quad m(y) := y \vee 0. \tag{5.6}$$

The term A_t is units of stock ordered this period, which take one period to arrive. We assume that the firm can store at most K items at one time.

Profits are given by

$$\pi_t := X_t \wedge D_{t+1} - cA_t - \kappa \mathbb{1}\{A_t > 0\}.$$

We take the minimum of current stock and demand because orders in excess of inventory are assumed to be lost rather than back-filled. Here c is unit product cost and κ is a fixed cost of ordering inventory.

We can map our inventory problem into a finite state MDP with state space $\mathsf{X} := \{0, \dots, K\}$ and action space $\mathsf{A} := \mathsf{X}$. The feasible correspondence Γ is

$$\Gamma(x) := \{0, \dots, K - x\}, \tag{5.7}$$

which represents the set of feasible orders when the current inventory state is x. The reward function is expected current profits, or

$$r(x, a) := \sum_{d \geqslant 0} (x \wedge d)\varphi(d) - ca - \kappa \mathbb{1}\{a > 0\}. \tag{5.8}$$

The stochastic kernel from the set of feasible state-action pairs G induced by Γ is, in view of (5.6),

$$P(x, a, x') := \mathbb{P}\{m(x - D_{t+1}) + a = x'\} \qquad \text{when} \quad D_{t+1} \sim \varphi. \tag{5.9}$$

Exercise 5.1.2. Suppose that φ is the geometric distribution on $\mathbb{Z}_+$ with parameter p. Write down an expression for the stochastic kernel (5.9) using only x, a, x' and the parameters of the model.

The Bellman equation for this optimal inventory problem is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{d \geqslant 0} v(m(x - d) + a)\varphi(d) \right\} \tag{5.11}$$

at each $x \in \mathsf{X}$, where $r(x, a)$ is as given in (5.8) and the aim is to solve for v. We introduce the Bellman operator

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{d \geqslant 0} v(m(x - d) + a)\varphi(d) \right\}. \tag{5.12}$$

This operator maps $\mathbb{R}^{\mathsf{X}}$ to itself and is designed so that its set of fixed points in $\mathbb{R}^{\mathsf{X}}$ coincide with solutions to (5.11) in $\mathbb{R}^{\mathsf{X}}$.

Exercise 5.1.3. Prove that T is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ when paired with the supremum norm $\|v\|_\infty := \max_{x \in \mathsf{X}} |v(x)|$.

5.1.2.3 Example: Cake Eating

Many dynamic programming problems in economics involve a trade-off between current and future consumption. The simplest example in this class is the "cake eating" problem, where initial household wealth is given but no labor income is received. Wealth evolves according to

$$W_{t+1} = R(W_t - C_t) \qquad (t = 0, 1, \ldots)$$

Here R is a gross rate of interest, so that investing d dollars today returns Rd next period, and C_t is current consumption. The agent seeks to maximize

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t u(C_t) \quad \text{given } W_0 = w.$$

We assume that $C_t \geqslant 0$ and $W_t \geqslant 0$, so that the agent cannot borrow. Consumption level C_t generates utility $u(C_t)$. Assuming that wealth takes values in a finite set $\mathsf{W} \subset \mathbb{R}_+$, the Bellman equation for this problem can be written as

$$v(w) = \max_{0 \leqslant w' \leqslant Rw} \left\{ u\left(w - \frac{w'}{R}\right) + \beta v(w') \right\}. \tag{5.13}$$

In (5.13) we are using $w' = R(w - c)$ to obtain $c = (w - w'/R)$. The household uses (5.13) to trade-off current utility of consumption against the value of future wealth.

Exercise 5.1.4. Frame this model as an MDP with W as the state space.

5.1.2.4 Example: Optimal Stopping

The optimal stopping problem we studied in Chapter 4 can be framed as an MDP. On one hand, doing so allows us to apply results obtained for MDPs to optimal stopping. On the other hand, expressing an optimal stopping problem as an MDP requires an additional state variable, which complicates the exposition. The exercise below helps to illustrate the key ideas.

Remark 5.1.1. While readers interested in the connection between optimal stopping and MDPs will benefit from this section, others can freely skip to §5.1.3 without losing continuity. Later, in Chapter 8, we will show how optimal stopping problems can be embedded in a very general framework without adding state variables.

Let's focus on the job search problem with Markov state discussed in §3.3.1 (although the arguments for the general optimal stopping problem in §4.1.1.1 are very

similar). As before, W is the set of wage outcomes. Since we need the symbol P for other purposes, we let Q be the Markov matrix for wages, so that $(W_t)_{t \geqslant 0}$ is Q-Markov on W.

To express the job search problem as an MDP, let $\mathsf{X} = \{0, 1\} \times \mathsf{W}$ be a state space whose typical element is (e, w), with e representing either unemployment ($e = 0$) or employment ($e = 1$) and w being the current wage offer. An action $a \in \mathsf{A} := \{0, 1\}$ indicates rejection or acceptance of the current wage offer.

Exercise 5.1.5. Express the job search problem as an MDP, with state space X and action space A as described in the previous paragraph.

5.1.3 Optimality

In this section we return to the general MDP setting of §5.1.1, define optimal policies and state our main optimality result. As was the case for job search, actions are governed by policies, which are maps from states to actions (see, in particular, §1.3.1.3, where policies were introduced).

5.1.3.1 Policies and Value

Given a fixed MDP (Γ, β, r, P), the set of **feasible policies** is

$$\Sigma := \{\sigma \in \mathsf{A}^{\mathsf{X}} : \sigma(x) \in \Gamma(x) \text{ for all } x \in \mathsf{X}\}. \tag{5.16}$$

If we select a particular policy σ from Σ, it is understood that we respond to state X_t with action $A_t := \sigma(X_t)$ at every date t. As a result, the state evolves by drawing X_{t+1} from $P(X_t, \sigma(X_t), \cdot)$ at each $t \geqslant 0$. In other words, $(X_t)_{t \geqslant 0}$ is P_σ-Markov for P_σ defined by

$$P_\sigma(x, x') := P(x, \sigma(x), x') \qquad (x, x' \in \mathsf{X}).$$

Note that $P_\sigma \in \mathbb{M}(\mathsf{X})$. Fixing a policy "closes the loop" in the state transition process and defines a Markov chain for the state.

Under the policy σ, rewards at state x are $r(x, \sigma(x))$. If we introduce the notation

$$r_\sigma(x) := r(x, \sigma(x)) \qquad (x \in \mathsf{X})$$

and $\mathbb{E}_x := \mathbb{E}[\,\cdot\,|X_0 = x]$, then the lifetime value of following σ starting from state x

can be written as

$$v_\sigma(x) = \mathbb{E}_x \sum_{t \geqslant 0} \beta^t r_\sigma(X_t) \quad \text{where } (X_t) \text{ is } P_\sigma\text{-Markov with } X_0 = x. \tag{5.17}$$

Since $\beta < 1$, applying Lemma 3.2.3 on page 91 to this expression yields

$$v_\sigma = \sum_{t \geqslant 0} \beta^t P_\sigma^t r_\sigma = (I - \beta P_\sigma)^{-1} r_\sigma. \tag{5.18}$$

Another way to compute v_σ is to use the **policy operator** T_σ defined at $v \in \mathbb{R}^{\mathsf{X}}$ by

$$(T_\sigma v)(x) = r(x, \sigma(x)) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, \sigma(x), x') \qquad (x \in \mathsf{X}). \tag{5.19}$$

(This is analogous to the policy operator defined for the optimal stopping problem in §4.1.1.3.) In vector notation, we can express the operator via

$$T_\sigma v = r_\sigma + \beta P_\sigma v. \tag{5.20}$$

The next exercise shows how T_σ can be put to work.

Exercise 5.1.6. Fixing σ in Σ, prove that

(i) T_σ is an order-preserving self-map on $\mathbb{R}^{\mathsf{X}}$,

(ii) T_σ is a contraction on $\mathbb{R}^{\mathsf{X}}$ of modulus β under the norm $\|\cdot\|_\infty$,

(iii) the σ-value function v_σ is the unique fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$, and

(iv) $T_\sigma^k v \to v_\sigma$ as $k \to \infty$ for all $v \in \mathbb{R}^{\mathsf{X}}$.

Computationally, this means that we can pick $v \in \mathbb{R}^{\mathsf{X}}$ and iterate with T_σ to obtain an approximation to v_σ.

Exercise 5.1.7. Prove that, when the initial condition for iteration is $v \equiv 0 \in \mathbb{R}^{\mathsf{X}}$, the k-th iterate $T_\sigma^k v$ is equal to the truncated sum $\sum_{t=0}^{k-1} \beta^t P_\sigma^t r_\sigma$.

Remark 5.1.2. To compute v_σ, should we use the expression $(I - \beta P_\sigma)^{-1} r_\sigma$ in (5.18) or should we iterate with T_σ? For small state spaces, the first option is typically faster. However, it is easy to write down dynamic programming problems where X is very large (see, e.g., Example 1.0.2 on page 2). If, say, X has 10^9 elements, then $I - \beta P_\sigma$ is $10^9 \times 10^9$. Matrices of this size are difficult to invert—or even store in memory. In such settings, iterating with T_σ is often faster.

The next exercise extends Exercise 5.1.7 and aids interpretation of policy operators.

EXERCISE 5.1.8. Fix $\sigma \in \Sigma$ and let (X_t) be P_σ-Markov with initial condition $x \in \mathsf{X}$. Prove that, for given $v \in \mathbb{R}^{\mathsf{X}}$, and $k \in \mathbb{N}$, we have

$$(T_\sigma^k v)(x) = \mathbb{E}_x \left[\sum_{t=1}^{k-1} \beta^t r(X_t, \sigma(X_t)) + \beta^k v(X_k) \right].$$

In other words, show that $(T_\sigma^k v)(x)$ is the payoff from following policy σ and starting in state x when lifetime is truncated to the finite horizon k and v provides a terminal payoff in each state.

5.1.3.2 Defining Optimality

The **value function** is defined as $v^* := \vee_{\sigma \in \Sigma} v_\sigma$, where, as usual, the maximum is pointwise. More explicitly,

$$v^*(x) = \max_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{5.21}$$

This is consistent with our definition of the value function in the optimal stopping case (see page 105). It is the maximal lifetime value we can extract from each state using optimal behavior. The maximum in (5.21) exists at each x because Σ is a finite set.

A policy $\sigma \in \Sigma$ is called **optimal** if $v_\sigma = v^*$. In other words, a policy is optimal if its lifetime value is maximal at each state.

As one illustration, consider again Figure 2.2 on page 47, supposing that $\{v_{\sigma'}, v_{\sigma''}\}$ is the entire set of σ-policy functions. As drawn, there is no optimal policy, since v^* differs from both $v_{\sigma'}$ and $v_{\sigma''}$. Below, in Proposition 5.1.1, we will show that such an outcome is *not* possible for MDPs.

Our optimality results are easier to follow with some additional terminology. To start, given $v \in \mathbb{R}^{\mathsf{X}}$, we define a policy $\sigma \in \Sigma$ to be **v-greedy** if

$$\sigma(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x,a,x') \right\} \quad \text{for all } x \in \mathsf{X}. \tag{5.22}$$

In essence, a v-greedy policy treats v as the correct value function and sets all actions accordingly.

Bellman's principle of optimality is said to hold if

$$\sigma \in \Sigma \text{ is optimal} \quad \iff \quad \sigma \text{ is } v^*\text{-greedy.}$$

The **Bellman operator** for the MDP is the self-map T on $\mathbb{R}^{\mathsf{X}}$ defined by

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x'} v(x') P(x,a,x') \right\} \qquad (x \in \mathsf{X}). \tag{5.23}$$

Obviously $Tv = v$ if and only if v satisfies the Bellman equation (5.2).

Exercise 5.1.9. Given $v \in \mathbb{R}^{\mathsf{X}}$, prove that

(i) at least one v-greedy policy exists,

(ii) $\sigma \in \Sigma$ is v-greedy if and only if $T_\sigma v = Tv$, and

(iii) $(Tv)(x) = \max_{\sigma \in \Sigma} (T_\sigma v)(x)$ for all $x \in \mathsf{X}$.

The last part of Exercise 5.1.9 tells us that T is the pointwise maximum of $\{T_\sigma\}_{\sigma \in \Sigma}$, which can be expressed as $T = \vee_\sigma T_\sigma$. Figure 5.1 illustrates this relationship in one dimension.

Exercise 5.1.10. Prove that T is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ under the norm $\|\cdot\|_\infty$. [Hint: Apply the max-inequality lemma (page 31) along with Exercise 5.1.9.]

5.1.3.3 Optimality Theory

We can now state our main optimality result for MDPs:

Proposition 5.1.1. *For the MDP described in §5.1.1,*

(i) *the value function v^* is the unique solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$,*

(ii) $\lim_{k \to \infty} T^k v = v^*$ *for all $v \in \mathbb{R}^{\mathsf{X}}$,*

(iii) *Bellman's principle of optimality holds, and*

(iv) *at least one optimal policy exists.*

We prove the results of Proposition 5.1.1 in a more general setting in Chapter 8, as part of a larger set of results. Nevertheless, we provide a direct proof here:

Figure 5.1: T is the pointwise maximum of $\{T_\sigma\}_{\sigma \in \Sigma}$ (one-dimensional setting)

Proof. In Exercise 5.1.10 we showed that T is a contraction mapping on the closed set $\mathbb{R}^{\mathsf{X}}$. Hence T is globally stable on $\mathbb{R}^{\mathsf{X}}$ and therefore has a unique fixed point $\bar{v} \in \mathbb{R}^{\mathsf{X}}$. Our first claim is that $\bar{v} = v^*$. We show $\bar{v} \leqslant v^*$ and then $\bar{v} \geqslant v^*$.

For the first inequality, let $\sigma \in \Sigma$ be $\bar{v}$-greedy. Recalling Exercise 5.1.9, we have $T_\sigma \bar{v} = T\bar{v} = \bar{v}$. Hence $\bar{v}$ is also a fixed point of T_σ. But the only fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ is v_σ, so $\bar{v} = v_\sigma$. But then $\bar{v} \leqslant v^*$, since, by definition, $v^* = \vee_\sigma v_\sigma$. This is our first inequality.

As for the second inequality, fix $\sigma \in \Sigma$ and observe that $T_\sigma v \leqslant Tv$ for all $v \in \mathbb{R}^{\mathsf{X}}$. Since T is order-preserving and globally stable, Proposition 2.1.3 on page 52 implies that $v_\sigma \leqslant \bar{v}$. Taking the supremum over $\sigma \in \Sigma$ yields $v^* \leqslant \bar{v}$.

Hence v^* is a fixed point of T in $\mathbb{R}^{\mathsf{X}}$. Since T is globally stable on $\mathbb{R}^{\mathsf{X}}$, the remaining claims in parts (i)–(ii) follow immediately.

As for part (iii), it follows from Exercise 5.1.9 and part (i) of this theorem that

$$\sigma \text{ is } v^*\text{-greedy} \quad \iff \quad T_\sigma v^* = Tv^* \quad \iff \quad T_\sigma v^* = v^*.$$

The right hand side of this expression tells us that v^* is a fixed point of T_σ. But the only fixed point of T_σ is v_σ, so the right hand side is equivalent to the statement $v_\sigma = v^*$.

Hence, by this chain of logic and the definition of optimality,

$$\sigma \text{ is } v^*\text{-greedy} \iff v^* = v_\sigma \iff \sigma \text{ is optimal} \tag{5.24}$$

Hence (iii) holds.

Part (iv) is left as an exercise immediately below. □

Exercise 5.1.11. Prove that, in Proposition 5.1.1, (iii) implies (iv).

Figure 5.2 illustrates Proposition 5.1.1 in a very simple case, where X is a singleton $\{x\}$. We write v instead of $v(x)$ for the value of state x and place v on the horizontal axis. In the figure, the set of policies is $\Sigma = \{\sigma', \sigma''\}$. For given $\sigma \in \Sigma$, the map T_σ is an affine function $T_\sigma v = r_\sigma + \beta P_\sigma v$ and the fixed point is v_σ. The Bellman operator T is the upper envelope of the functions $\{T_\sigma\}$, as shown in (ii) of Exercise 5.1.9. By definition,

(i) v^* is the largest of these fixed points, which equals $v_{\sigma''}$, and

(ii) σ'' is the optimal policy, since $v_{\sigma''} = v^*$.

In accordance with Proposition 5.1.1, v^* is also the fixed point of the Bellman operator.

It is important to understand the significance of (iii) in Proposition 5.1.1. Greedy policies are relatively easy to compute, in the sense that solving (5.22) at each x is easier than trying to directly solve the problem of maximizing lifetime value, since Σ is in general far larger than $\Gamma(x)$. Part (iii) tells us that solving the overall problem reduces to computing a v-greedy policy with the right choice of v. For optimal stopping problems, that choice is the value function v^*. Intuitively, v^* assigns a "correct" value to each state, in the sense of maximal lifetime value the controller can extract, so using v^* to calculate greedy policies leads to the optimal outcome.

5.1.4 Algorithms

In previous chapters we solved job search and optimal stopping problems using value function iteration. In this section we present a generalization suitable for arbitrary MDPs and then discuss two important alternatives.

5.1.4.1 Value Function Iteration

Value function iteration (VFI) for MDPs is similar to VFI for the job search model (see page 36): we use successive approximation on T to compute an approximation

Figure 5.2: Illustration of optimality for MDPs

v_k to the value function v^* and then take a v_k-greedy policy. The general procedure is given by Algorithm 4.

The fact that the sequence $(v_k)_{k \geqslant 0}$ produced by VFI converges to v^* is immediate from Proposition 5.1.1 (as the tolerance τ is taken toward zero). It is also true that the greedy policy produced in the last step is approximately optimal when τ is small. Further details on convergence are deferred to Chapter 8, where we examine VFI in a more general setting.

VFI is robust, easy to understand and easy to implement. These properties explain its enduring popularity. At the same time, in terms of efficiency, VFI is often dominated by alternative algorithms, two of which are discussed below.

5.1.4.2 Howard Policy Iteration

Unlike VFI, **Howard policy iteration (HPI)** (or just **policy iteration**) computes optimal policies by iterating between computing the value of a given policy and computing the greedy policy associated with that value. The full technique is described in Algorithm 5.[1]

[1]Notice that we use the norm distance $\|\sigma_k - \sigma_{k+1}\|_\infty$ between policies. This requires that all policies are either real-valued or vector-valued, which is not a strong assumption. (The vector-valued case

Algorithm 4: Value function iteration for MDPs

```
input v_0 ∈ R^X, an initial guess of v*
input τ, a tolerance level for error
ε ← τ + 1
k ← 0
while ε > τ do
    for x ∈ X do
        v_{k+1}(x) ← (T v_k)(x)
    end
    ε ← ||v_k − v_{k+1}||_∞
    k ← k + 1
end
Compute a v_k-greedy policy σ
return σ
```

Algorithm 5: Howard policy iteration for MDPs

```
input σ_0 ∈ Σ, an initial guess of σ*
k ← 0
ε ← 1
while ε > 0 do
    v_k ← the σ_k-value function (I − βP_{σ_k})^{−1} r_{σ_k}
    σ_{k+1} ← a v_k-greedy policy
    ε ← ||σ_k − σ_{k+1}||_∞
    k ← k + 1
end
return σ_k
```

Figure 5.3: Howard policy iteration algorithm (HPI)

A visualization of HPI is given in Figure 5.3, where σ is the initial choice. Next we compute the lifetime value v_σ, and then the v_σ-greedy policy σ', and so on. The computation of lifetime value is called the **policy evaluation** step, while the computation of greedy policies is called **policy improvement**.

HPI has two very attractive features. One is that, in a finite state setting, the algorithm always converges to an exact optimal policy in a finite number of steps, regardless of the initial condition. We provide a proof of this assertion in a more general setting in Chapter 8. The second is that the rate of convergence is faster than VFI, as shown below in §5.1.4.3.

Figure 5.4 gives another illustration, presented in the one-dimensional setting that we used for Figure 5.2. In this illustration, we imagine that there are many optimal policies, and hence many functions in $\{T_\sigma\}$, so that their upper envelope, which is the Bellman operator, becomes a smoother curve. The figure shows the update from v_σ to the next lifetime value $v_{\sigma'}$, via the following two steps:

(i) Take σ' to be v_σ-greedy, which means that $T_{\sigma'} v_\sigma = T v_\sigma$ (see Exercise 5.1.9).

(ii) Take $v_{\sigma'}$ to be the fixed point of $T_{\sigma'}$.

The next step, from $v_{\sigma'}$ to $v_{\sigma''}$ is analogous.

Comparison of this figure with Figure 2.7 on page 59 suggests that HPI is an implementation of Newton's method, applied to the Bellman operator. We confirm this in §5.1.4.3.

allows for multiple actions.) If $\sigma_k(x)$ is a vector for each x, then $\sigma_k - \sigma_{k+1}$ is a multi-dimensional array,

Figure 5.4: HPI as a version of Newton's method

5.1.4.3 HPI as Newton Iteration

In discussing the connection between HPI and Newton iteration, one issue is that T is not always differentiable, as seen in Figure 5.2. But T is convex, and this lets us substitute *subgradients* for derivatives. Once we make this modification, HPI and Newton iteration are identical, as we now show.

First, recall that, given a self-map T from $S \subset \mathbb{R}^n$ to itself, an $n \times n$ matrix D is called a **subgradient** of T at $v \in S$ if

$$Tu \geqslant Tv + D(u - v) \quad \text{for all } u \in S. \tag{5.25}$$

Figure 5.5 illustrates the definition in one dimension, where D is just a scalar determining the slope of a tangent line at v. In the left subfigure, T_1 is convex and differentiable at v, which means that only one subgradient exists (since any other choice of slope implies that the inequality in (5.25) will fail for some u). In the right subfigure, T_2 is convex but nondifferentiable at v, so multiple subgradients exist.

In the next result, we take (Γ, β, r, P) to be a given MDP and let T be the associated Bellman operator.

and the supremum norm distance is just the maximum deviation over all states and actions.

Figure 5.5: Subgradients of convex functions

Lemma 5.1.2. *If $v \in \mathbb{R}^{\mathsf{X}}$ and $\sigma \in \Sigma$ is v-greedy, then βP_σ is a subgradient of T at v.*

Proof. Fix $v \in \mathbb{R}^{\mathsf{X}}$ and let σ be v-greedy. Using $T \geqslant T_\sigma$ and $T_\sigma v = Tv$, we have

$$Tu = Tv + Tu - Tv \geqslant Tv + T_\sigma u - T_\sigma v.$$

Applying the definition of T_σ now gives

$$Tu \geqslant Tv + \beta P_\sigma u - \beta P_\sigma v = Tv + \beta P_\sigma (u - v).$$

Hence βP_σ is a subgradient of T at v, as claimed. □

Now let's consider Newton's method applied to the problem of finding the fixed point of T. Since T is nondifferentiable and convex, we replace the Jacobian in Newton's method (see (2.6) on page 58) with the subdifferential. This leads us to iterate on

$$v_{k+1} = Qv_k \quad \text{where} \quad Qv := (I - \beta P_\sigma)^{-1}(Tv - \beta P_\sigma v).$$

In the definition of Q, the policy σ is v-greedy. Using $Tv = T_\sigma v$, the map Q reduces to $Qv := (I - \beta P_\sigma)^{-1} r_\sigma$, which is exactly the update step to produce the next σ-value function in HPI (i.e., the lifetime value of a v-greedy policy).

The fact that HPI is a version of Newton's method suggests that its iterates $(v_k)_{k \geqslant 0}$ enjoy quadratic convergence. This is indeed the case: under mild conditions one can show there exists a constant N such that, for all $k \geqslant 0$,

$$\|v_{k+1} - v_k\| \leqslant N \|v_k - v_{k-1}\|^2 \tag{5.26}$$

(see, e.g., Puterman (2005), Theorem 6.4.8). Hence HPI enjoys both a fast convergence rate and the robustness of global convergence.

However, HPI is not always optimal in terms of efficiency, since the size of the constant term in (5.26) also matters. This term can be large because, at each step, the

update from v_σ to $v_{\sigma'}$ requires computing the exact lifetime value $v_{\sigma'}$ of the v_σ-greedy policy σ'. Computing this fixed point exactly can be computationally expensive in high dimensions.

One way around this issue is to forgo computing the fixed point $v_{\sigma'}$ exactly, replacing it with an approximation. The next section takes up this idea.

5.1.4.4 Optimistic Policy Iteration

Optimistic policy iteration (OPI) is an algorithm that borrows from both VFI and HPI. In short, the algorithm is the same as HPI except that, instead of computing the full value v_σ of a given policy, the approximation $T_\sigma^m v$ discussed in Exercise 5.1.6 is used instead. Algorithm 6 provides details.

Algorithm 6: Optimistic policy iteration for MDPs

```
input v_0 ∈ ℝ^X, an initial guess of v*
input τ, a tolerance level for error
input m ∈ ℕ, a step size
k ← 0
ε ← τ + 1
while ε > τ do
    σ_k ← a v_k-greedy policy
    v_{k+1} ← T^m_{σ_k} v_k
    ε ← ‖v_k − v_{k+1}‖_∞
    k ← k + 1
end
return σ_k
```

In the algorithm, the policy operator T_{σ_k} is applied m times to generate an approximation of v_{σ_k}. The constant step size m can also be replaced with a sequence $(m_k) \subset \mathbb{N}$. In either case, for MDPs, convergence to an optimal policy is guaranteed. We prove this in a more general setting in Chapter 8.

Notice that, as $m \to \infty$, the algorithm increasingly approximates Howard policy iteration, since $T_{\sigma_k}^m v_k$ converges to v_{σ_k}. At the same time, if $m = 1$, the algorithm is essentially the same as VFI. Hence, with intermediate m, OPI can be seen as a combination of HPI and VFI.

In many practical dynamic programming problems, there exist choices of m that lead to faster convergence than both VFI and HPI. We investigate these ideas in the applications below.

5.2 Applications

This section gives several applications of the MDP model to economic problems. The applications illustrate the ease with which MDPs can be implemented on a computer (provided that the state and action spaces are not too large).

5.2.1 Optimal Inventories

In §3.1.1.2 we studied a firm whose inventory behavior was specified to follow S-s dynamics. In §5.1.2.2 we introduced a model where investment behavior is endogenous, determined by the desire to maximize firm value. In this section, we show that this endogenous inventory behavior can replicate the S-s dynamics from §3.1.1.2.

We saw in §5.1.2.2 that the optimal inventory model is an MDP, so the Proposition 5.1.1 optimality and convergence results apply. In particular, the unique fixed point of the Bellman operator is the value function v^*, and a policy σ^* is optimal if and only if σ^* is v^*-greedy.

Let's now solve this model numerically. As in Exercise 5.1.2, we take φ to be the geometric distribution on $\mathbb{Z}_+$ with parameter p. We use the default parameter values shown in Listing 14. The code listing also presents an implementation of the Bellman operator. We use the `OffSetArrays` package to index arrays on the custom set `0:K`, since this corresponds to the state space.

Figure 5.6 exhibits an approximation of the value function v^*, computed by iterating with T starting at $v \equiv 1$. Figure 5.6 also shows the approximate optimal policy, obtained as a v^*-greedy policy:

$$\sigma^*(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{d \geqslant 0} v^*(m(x-d)+a)\varphi(d) \right\}$$

The plot of the optimal policy shows that there is a threshold region below which the firm orders large batches and above which the firm orders nothing. This makes sense, since the firm wishes to economize on the fixed cost of ordering. Figure 5.7 shows a simulation of inventory dynamics under the optimal policy, starting from $X_0 = 0$. The time path closely approximates the S-s dynamics discussed in §3.1.1.2.

Exercise 5.2.1. Compute the optimal policy by extending the code given in Listing 14. Replicate Figure 5.7, modulo randomness, by sampling from a geometric distribution and implementing the dynamics in (5.6). At each X_t, the action A_t should be chosen according to the optimal policy $\sigma^*(X_t)$.

```julia
using Distributions, OffsetArrays
m(x) = max(x, 0)  # Convenience function

function create_inventory_model(; β=0.98,     # discount factor
                                  K=40,       # maximum inventory
                                  c=0.2, κ=2, # cost paramters
                                  p=0.6)      # demand parameter
    ϕ(d) = (1 - p)^d * p  # demand pdf
    return (; β, K, c, κ, p, ϕ)
end

"The function B(x, a, v) = r(x, a) + β Σ_x′ v(x′) P(x, a, x′)."
function B(x, a, v, model; d_max=100)
    (; β, K, c, κ, p, ϕ) = model
    reward = sum(min(x, d)*ϕ(d) for d in 0:d_max) - c * a - κ * (a > 0)
    continuation_value = β * sum(v[m(x - d) + a] * ϕ(d) for d in 0:d_max)
    return reward + continuation_value
end

"The Bellman operator."
function T(v, model)
    (; β, K, c, κ, p, ϕ) = model
    new_v = similar(v)
    for x in 0:K
        Γx = 0:(K - x)
        new_v[x], _ = findmax(B(x, a, v, model) for a in Γx)
    end
    return new_v
end

"Get a v-greedy policy.  Returns a zero-based array."
function get_greedy(v, model)
    (; β, K, c, κ, p, ϕ) = model
    σ_star = OffsetArray(zeros(Int32, K+1), 0:K)
    for x in 0:K
        Γx = 0:(K - x)
        _, a_idx = findmax(B(x, a, v, model) for a in Γx)
        σ_star[x] = Γx[a_idx]
    end
    return σ_star
end
```

Listing 14: Solving the optimal inventory model (`inventory_dp.jl`)

Figure 5.6: The value function and optimal policy for the inventory problem

Figure 5.7: Optimal inventory dynamics

5.2.2 Optimal Savings with Labor Income

As our next example of an MDP, we modify the cake eating problem in §5.1.2.3 to add labor income. Wealth evolves according to

$$W_{t+1} = R(W_t + Y_t - C_t) \qquad (t = 0, 1, \ldots), \tag{5.27}$$

where (W_t) takes values in finite set $\mathsf{W} \subset \mathbb{R}_+$ and (Y_t) is a Markov chain on finite set Y with transition matrix Q.[2] Other parts of the problem are unchanged.

The Bellman operator can be written as

$$(Tv)(w, y) = \max_{w' \in \Gamma(w,y)} \left\{ u\left(w + y - \frac{w'}{R}\right) + \beta \sum_{y' \in \mathsf{Y}} v(w', y') Q(y, y') \right\}. \tag{5.28}$$

5.2.2.1 MDP Representation

To frame this as an MDP, we set the state to $x := (w, y)$, representing current wealth and income, taking values in the state space $\mathsf{X} := \mathsf{W} \times \mathsf{Y}$. The action is savings s, which takes values in W and equals w'. The feasible correspondence is the set of feasible savings values

$$\Gamma(w, y) = \{s \in \mathsf{W} : s \leqslant R(w + y)\}.$$

The current reward is utility of consumption $r(w, s) = u(w + y - s/R)$. The stochastic kernel is

$$P((w, y), s, (w', y')) = \mathbb{1}\{w' = s\} Q(y, y').$$

Having framed an MDP, the Proposition 5.1.1 optimality results apply.

5.2.2.2 Implementation

To implement the algorithms discussed in §5.1.4, we use the Bellman operator (5.28), and the corresponding definition of a v-greedy policy, which is

$$\sigma((w, y) \in \operatorname*{argmax}_{w' \in \Gamma(w,y)} \left\{ u\left(w + y - \frac{w'}{R}\right) + \beta \sum_{y' \in \mathsf{Y}} v(w', y') Q(y, y') \right\}$$

[2] See Marcet et al. (2007) and Zhu (2020) for more extensive analysis of how adding a labor supply choice can affect outcomes in a consumption-savings model.

for all (w, y). The policy operator for given $\sigma \in \Sigma$ is

$$(T_\sigma v)(w, y) = u\left(w + y - \frac{\sigma(w, y)}{R}\right) + \beta \sum_{y' \in \mathsf{Y}} v(\sigma(w, y), y')Q(y, y'). \tag{5.29}$$

Code for implementing the model and these two operators is given in Listing 15. Income is constructed as a discretized AR(1) process using the method from §3.1.3. Exponentiation is applied to the grid so that income takes positive values.

The function `get_value` in Listing 16 uses the expression $v_\sigma = (I - \beta P_\sigma)^{-1} r_\sigma$ from (5.18) to obtain the value of a given policy σ. The matrix P_σ and vector r_σ take the form

$$\begin{aligned} P_\sigma((w, y), (w', y')) &= \mathbb{1}\{\sigma(w, y) = w'\}Q(y, y') \\ r_\sigma(w, y) &= u(w + y - \sigma(w, y)/R). \end{aligned}$$

5.2.2.3 Timing

Since all results for MDPs apply, we know that the value function v^* is the unique fixed point of the Bellman operator in $\mathbb{R}^{\mathsf{X}}$, and that value function iteration, Howard policy iteration and optimistic policy iteration all converge. Listing 17 implements these three algorithms. Since the state and action space are finite, Howard policy iteration is guaranteed to return an exact optimal policy.

Figure 5.8 shows the wall time taken to solve the finite optimal savings model under the default parameters when executed on a laptop machine with 8 CPUs running at 2.8GHz. Time is measured in seconds. The horizontal axis corresponds to the step parameter m in optimistic policy iteration (Algorithm 6). The two other algorithms do not depend on m and hence their timings are constant. The figure shows that policy iteration is an order of magnitude faster than value function iteration and that optimistic policy iteration is even faster for moderate values of m.

Although these timings are comparable, run-times are always dependent on implementation and relative speed varies significantly with how algorithms are written, the extent to which parallelization can be exploited and the parameters and description of the problem. However, our results are consistent with other applications in this text: for many choices of the step size m, optimistic policy iteration dominates both value function iteration and Howard policy iteration.

```
using QuantEcon, LinearAlgebra, IterTools

function create_savings_model(; R=1.01, β=0.98, γ=2.5,
                                w_min=0.01, w_max=20.0, w_size=200,
                                ρ=0.9, ν=0.1, y_size=5)
    w_grid = LinRange(w_min, w_max, w_size)
    mc = tauchen(y_size, ρ, ν)
    y_grid, Q = exp.(mc.state_values), mc.p
    return (; β, R, γ, w_grid, y_grid, Q)
end

"B(w, y, w′, v) = u(R*w + y - w′) + β Σ_y′ v(w′, y′) Q(y, y′)."
function B(i, j, k, v, model)
    (; β, R, γ, w_grid, y_grid, Q) = model
    w, y, w′ = w_grid[i], y_grid[j], w_grid[k]
    u(c) = c^(1-γ) / (1-γ)
    c = w + y - (w′ / R)
    @views value = c > 0 ? u(c) + β * dot(v[k, :], Q[j, :]) : -Inf
    return value
end

"The Bellman operator."
function T(v, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    v_new = similar(v)
    for (i, j) in product(w_idx, y_idx)
        v_new[i, j] = maximum(B(i, j, k, v, model) for k in w_idx)
    end
    return v_new
end

"The policy operator."
function T_σ(v, σ, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    v_new = similar(v)
    for (i, j) in product(w_idx, y_idx)
        v_new[i, j] = B(i, j, σ[i, j], v, model)
    end
    return v_new
end
```

Listing 15: Discrete optimal savings model (`finite_opt_saving_0.jl`)

```
include("finite_opt_saving_0.jl")

"Compute a v-greedy policy."
function get_greedy(v, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    σ = Matrix{Int32}(undef, length(w_idx), length(y_idx))
    for (i, j) in product(w_idx, y_idx)
        _, σ[i, j] = findmax(B(i, j, k, v, model) for k in w_idx)
    end
    return σ
end

"Get the value v_σ of policy σ."
function get_value(σ, model)
    # Unpack and set up
    (; β, R, γ, w_grid, y_grid, Q) = model
    w_idx, y_idx = (eachindex(g) for g in (w_grid, y_grid))
    wn, yn = length(w_idx), length(y_idx)
    n = wn * yn
    u(c) = c^(1-γ) / (1-γ)
    # Build P_σ and r_σ as multi-index arrays
    P_σ = zeros(wn, yn, wn, yn)
    r_σ = zeros(wn, yn)
    for (i, j) in product(w_idx, y_idx)
            w, y, w′ = w_grid[i], y_grid[j], w_grid[σ[i, j]]
            r_σ[i, j] = u(w + y - w′/R)
        for (i′, j′) in product(w_idx, y_idx)
            if i′ == σ[i, j]
                P_σ[i, j, i′, j′] = Q[j, j′]
            end
        end
    end
    # Reshape for matrix algebra
    P_σ = reshape(P_σ, n, n)
    r_σ = reshape(r_σ, n)
    # Apply matrix operations --- solve for the value of σ
    v_σ = (I - β * P_σ) \ r_σ
    # Return as multi-index array
    return reshape(v_σ, wn, yn)
end
```

Listing 16: Discrete optimal savings model (`finite_opt_saving_1.jl`)

```
include("s_approx.jl")
include("finite_opt_saving_1.jl")

"Value function iteration routine."
function value_iteration(model, tol=1e-5)
    vz = zeros(length(model.w_grid), length(model.y_grid))
    v_star = successive_approx(v -> T(v, model), vz, tolerance=tol)
    return get_greedy(v_star, model)
end

"Howard policy iteration routine."
function policy_iteration(model)
    wn, yn = length(model.w_grid), length(model.y_grid)
    σ = ones(Int32, wn, yn)
    i, error = 0, 1.0
    while error > 0
        v_σ = get_value(σ, model)
        σ_new = get_greedy(v_σ, model)
        error = maximum(abs.(σ_new - σ))
        σ = σ_new
        i = i + 1
        println("Concluded loop $i with error $error.")
    end
    return σ
end

"Optimistic policy iteration routine."
function optimistic_policy_iteration(model; tolerance=1e-5, m=100)
    v = zeros(length(model.w_grid), length(model.y_grid))
    error = tolerance + 1
    while error > tolerance
        last_v = v
        σ = get_greedy(v, model)
        for i in 1:m
            v = T_σ(v, σ, model)
        end
        error = maximum(abs.(v - last_v))
    end
    return get_greedy(v, model)
end
```

Listing 17: Discrete optimal savings model (`finite_opt_saving_2.jl`)

Figure 5.8: Timings for alternative algorithms, savings model

5.2.2.4 Outputs

Figure 5.9 shows a typical time series for the wealth of a single household under the optimal policy. The series is created by computing an optimal policy σ^*, generating $(Y_t)_{t=0}^{m-1}$ as a Q-Markov chain on Y and then computing $(W_t)_{t=0}^{m}$ via $W_{t+1} = \sigma^*(W_t, Y_t)$ for t running from 0 to $m-1$. Initial wealth W_0 is set to 1.0 and $m = 2000$.

Figure 5.10 shows the result of computing and histogramming a longer time series, with m set to 1,000,000. This histogram approximates the stationary distribution of wealth for a large population, each updating via σ^* and each with independently generated labor income series $(Y_t)_{t=0}^{m-1}$. (This is due to ergodicity of the wealth-income process. For a discussion of the connection between stationary distributions and time series under ergodicity see, for example, Sargent and Stachurski (2022).)

The shape of the wealth distribution in Figure 5.10 is unrealistic. In almost all countries, the wealth distribution has a very long right tail. The Gini coefficient of the distribution is in Figure 5.10 is 0.54, which is too low. For example, World Bank data for 2019 produces a wealth Gini for the US equal to 0.852. For Germany and Japan the figures are 0.816 and 0.627 respectively.

In §5.3.3 we discuss a variation on the optimal savings model that can produce a more realistic wealth distribution.

Figure 5.9: Time series for wealth

Figure 5.10: Histogram of wealth

5.2.3 Optimal Investment

As our next application, we consider a monopolist facing adjustment costs and stochastically evolving demand. The monopolist balances setting enough capacity to meet demand against costs of adjusting capacity.

5.2.3.1 Problem Description

We assume that the monopolist produces a single product and faces an inverse demand function of the form

$$P_t = a_0 - a_1 Y_t + Z_t,$$

where a_0, a_1 are positive parameters, Y_t is output, P_t is price and the demand shock Z_t follows

$$Z_{t+1} = \rho Z_t + \sigma \eta_{t+1}, \qquad \{\eta_t\} \overset{\text{IID}}{\sim} N(0,1).$$

Current profits are

$$\pi_t := P_t Y_t - c Y_t - \gamma (Y_{t+1} - Y_t)^2.$$

Here $\gamma(Y_{t+1} - Y_t)^2$ represents costs associated with adjusting production scale, parameterized by γ, and c is unit cost of current production. Costs are convex, so rapid changes to capacity are expensive.

The monopolist chooses (Y_t) to maximize the expected discounted value of its profit flow, which we write as

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t \pi_t. \tag{5.30}$$

Here $\beta = 1/(1+r)$, where $r > 0$ is a fixed interest rate.

A way to start thinking about the optimal time path of output is to consider what would happen if $\gamma = 0$. Without adjustment costs there is no intertemporal trade-off, so the monopolist should choose output to maximize current profit in each period. The implied level of output at time t is

$$\bar{Y}_t := \frac{a_0 - c + Z_t}{2a_1}. \tag{5.31}$$

Exercise 5.2.2. Show that $\bar{Y}_t$ maximizes current profit when $\gamma = 0$.

For $\gamma > 0$, we expect the following behavior.

- If γ is close to zero, then the optimal output path Y_t will track the time path of $\bar{Y}_t$ relatively closely, while
- if γ is larger, then Y_t will be significantly smoother than $\bar{Y}_t$, as the monopolist seeks to avoid adjustment costs.

5.2.3.2 MDP Representation

We can represent this as an MDP. To do so we let Y be a grid contained in $\mathbb{R}_+$ that lists possible output values. To conform to the finite state setting, we discretize the shock process (Z_t) using Tauchen's method, as described in §3.1.3. For convenience we again use (Z_t) to represent the discrete process, which is a finite Markov chain on $\mathsf{Z} \subset \mathbb{R}$ with transition matrix Q.

The state space for this MDP is $\mathsf{X} = \mathsf{Y} \times \mathsf{Z}$, while the action space is Y. The feasible correspondence is defined by $\Gamma(x) = \mathsf{Y}$, meaning that choice of output is not restricted by the state. Thus, the feasible policy set Σ is all $\sigma \colon \mathsf{Y} \times \mathsf{Z} \to \mathsf{Y}$.

We write (y, z) for the current state, q for the action (which chooses next period output) and (y', z') for the next period state. The current reward function is current profits, which we can write as

$$r((y, z), q) = (a_0 - a_1 y + z - c)y - \gamma(q - y)^2.$$

The stochastic kernel is

$$P((y, z), q, (y', z')) = \mathbb{1}\{y' = q\}Q(z, z').$$

The term $\mathbb{1}\{y' = q\}$ states that next period output y' is equal to our current choice q for next period output. With these definitions, the problem defines an MDP and all of the optimality theory for MDPs applies.

5.2.3.3 Implementation

The Bellman operator can be expressed as

$$(Tv)(y, z) = \max_{y' \in \mathbb{R}} \left\{ r(y, z, y') + \beta \sum_{z' \in \mathsf{Z}} v(y', z')Q(z, z') \right\}.$$

Given $\sigma \in \Sigma$, we can express the policy operator as

$$(T_\sigma v)(y, z) = r(y, z, \sigma(y, z)) + \beta \sum_{z' \in \mathsf{Z}} v(\sigma(y, z), z') Q(z, z').$$

A v-greedy policy is a $\sigma \in \Sigma$ that obeys

$$\sigma(y, z) = \operatorname*{argmax}_{y' \in \mathsf{Y}} \left\{ r(y, z, y') + \beta \sum_{z' \in \mathsf{Z}} v(y', z') Q(z, z') \right\}.$$

By combining iteration with the policy operator and computation of greedy policies, we can implement optimistic policy iteration, compute the optimal policy σ^*, and study output choices generated by this policy. We are particularly interested in how output responds over time to randomly generated demand shocks.

Figure 5.11 shows the result of a simulation designed to shed light on how output responds to demand. After choosing initial values (Y_1, Z_1) and generating a Q-Markov chain $(Z_t)_{t=1}^T$, we simulated optimal output via $Y_{t+1} = \sigma^*(Y_t, Z_t)$. The default parameters are shown in Listing 18. In the figure, the adjustment cost parameter γ is varied as shown in the title. In addition to the optimal output path, the path of $(\bar{Y}_t)$ as defined in (5.31) is also presented.

The figure shows how increasing γ promotes smoothing, as predicted in our discussion above. For small γ, adjustment costs have only minor effects on choices, so output closely follows $(\bar{Y}_t)$, the optimal path when output responds immediately to demand shocks. Conversely, larger values of γ make adjustment expensive, so the operator responds relatively slowly to changes in demand.

Figure 5.12 compares timings value function iteration (VFI) and optimistic policy iteration (OPI). Timings appear in Listing 18. As in Figure 5.8, which gave timings for the optimal savings model, the horizontal axis shows m, which is the step parameter in OPI (Algorithm 6). VFI does not depend on m and hence its timings are constant. The vertical axis is time in seconds. In this figure, we do not show the result for Howard policy iteration (HPI) because the time taken was around 12 times larger than VFI.

Note that HPI is far slower than VFI, in contrast to our findings for the optimal savings problem. One might suspect this is due to the fact that HPI computes the exact optimal policy (whereas VFI is not guaranteed to do so) but this is not the case. Indeed, in this instance, HPI, VFI and OPI all returned the same policy, which is equal to the optimal policy. Rather, the main difference is in the discount factors. VFI can be very slow when the discount factor is close to one. (This is because VFI convergence is linear in β, whereas HPI convergence is quadratic.) This was the setting of the optimal savings case (where $\beta = 0.99$). Here, however, $r = 0.04$ and $\beta = 1/(1 + r)$, so

Figure 5.11: Simulation of optimal output with different adjustment costs

```
using QuantEcon, LinearAlgebra, IterTools
include("s_approx.jl")

function create_investment_model(;
        r=0.04,                                # Interest rate
        a_0=10.0, a_1=1.0,                     # Demand parameters
        γ=25.0, c=1.0,                         # Adjustment and unit cost
        y_min=0.0, y_max=20.0, y_size=100,     # Grid for output
        ρ=0.9, ν=1.0,                          # AR(1) parameters
        z_size=25)                             # Grid size for shock
    β = 1/(1+r)
    y_grid = LinRange(y_min, y_max, y_size)
    mc = tauchen(y_size, ρ, ν)
    z_grid, Q = mc.state_values, mc.p
    return (; β, a_0, a_1, γ, c, y_grid, z_grid, Q)
```

Listing 18: Discrete optimal investment model (`finite_lq.jl`)

β is relatively small and VFI performs quite well.

More important is that, for almost all values of m, OPI dominates both VFI and HPI in terms of speed, which is consistent with our findings for the optimal savings model. At $m = 60$, OPI is more than 20 times faster than VFI.

Exercise 5.2.3. Consider a firm that maximizes expected discounted value in a setting where future profits are discounted at rate $\beta = 1/(1+r)$, the only production input is labor and hiring involves fixed costs. Let ℓ_t be employment at the firm at time t. Current profits are

$$\pi_t = pZ_t\ell_t^\alpha - w\ell_t - \kappa\mathbb{1}\{\ell_{t+1} \neq \ell_t\},$$

where p is the output price, w is the wage rate, α is a production parameter, the productivity shock is Q-Markov on Z and κ is a fixed cost of hiring and firing. This fixed cost induces lumpy adjustment, as shown in Figure 5.13. Show that this model is an MDP. Write the Bellman equation and the procedure for optimistic policy iteration in the context of this model. Replicate Figure 5.13, modulo randomness, using the parameters shown in Listing 19.

Figure 5.12: Timings for alternative algorithms, investment model

```
using QuantEcon, LinearAlgebra, IterTools

function create_hiring_model(;
        r=0.04,                                # Interest rate
        κ=1.0,                                 # Adjustment cost
        α=0.4,                                 # Production parameter
        p=1.0, w=1.0,                          # Price and wage
        l_min=0.0, l_max=30.0, l_size=100,     # Grid for labor
        ρ=0.9, ν=0.4, b=1.0,                   # AR(1) parameters
        z_size=100)                            # Grid size for shock
    β = 1/(1+r)
    l_grid = LinRange(l_min, l_max, l_size)
    mc = tauchen(z_size, ρ, ν, b, 6)
    z_grid, Q = mc.state_values, mc.p
    return (; β, κ, α, p, w, l_grid, z_grid, Q)
end
```

Listing 19: Firm hiring model (`firm_hiring.jl`)

Figure 5.13: Optimal shifts in the stock of labor

5.3 Modified Bellman Equations

Direct application of MDP theory is sometimes suboptimal. For example, we saw in §1.3.2.2 that solving the job search problem with IID wage draws is best accomplished by generating a recursion on the continuation value, which reduces dimensionality for iterative solution methods. Separately, in §4.2.2.2, we saw how a different manipulation of the Bellman equation also increased efficiency.

Now we aim to study such modifications systematically. We begin by providing other examples of how manipulating a Bellman equation can facilitate computation and analysis. Then we establish a solid theoretical foundation for these kinds of manipulations, and to show how similar ideas can also be applied to policy operators and greedy policies.

5.3.1 Structural Estimation

As a first illustration of the ideas in this section, we discuss a connection between econometric estimation and dynamic programs. Our focus is on some modifications that econometricians often make to Bellman equations and how they affect computation and optimality.

5.3.1.1 What is Structural Estimation?

Structural estimation is a branch of quantitative social science in which, in a quest to understand observed quantities and prices, researchers attribute Markov decision problems to economic agents. A key step in this approach is to formulate dynamic programs in terms of functional forms and parameters. The econometric challenge is to infer parameters that bring the model outputs as close as possible to actual data.

Structural estimation aims to discover objects that are invariant to hypothetical interventions that the analysis wants to investigate. Examples of such invariant objects are parameters of utility functions, discount factors, and production technologies. Agents inside model solve their MDPs. A policy intervention that systematically alters the Markov processes that they face will alter agents' optimal policies, i.e., their decision rules. Various examples of such interventions involving aspects of fiscal and monetary policy are described in various chapters of Lucas and Sargent (1981) a compendium of early papers that were written in response to the Lucas (1976) Critique of then prevailing dynamic econometric models.[3]

Example 5.3.1. Gillingham et al. (2022) study the used car market in Denmark by modeling consumers who trade cars in the new and used car markets. By modeling consumers' decision problems, the authors are able to investigate how consumers would react to a hypothetical modification in automobile taxes. The study finds that automobile taxes were too high in the sense that the government could have raised more tax revenue by lowering tax rates.

Efficient solution methods are essential in structural estimation because the underlying dynamic program must be solved repeatedly in order to search the parameter space for a good fit to data. Moreover, these dynamic programs are often high-dimensional, due to shocks to preferences and other random variables that the agents inside the model are assumed to see but that the econometrician does not. When these shocks are persistent, the dimension of the state grows.[4]

In order to maintain focus on dynamic programming, we will not describe the details of the estimation step required for structural estimation (although §5.4 contains references for those who wish to learn about that). Instead, we focus on the kinds of

[3]Rational expectations econometrics was a response to that Critique. While early work on rational expectations originated from the macroeconomics community (e.g. Hansen and Sargent (1980), Hansen and Sargent (1990)), many of their examples were actually about industrial organization and other microeconomic models. This work was part of a broad process that erased many boundaries between micro and macro theory.

[4]Hansen and Sargent (1980) analyze the implications of such "Shiller errors" for efficient estimation procedures in a class of linear structural models.

dynamic programs treated in structural estimation and techniques for solving them efficiently.

5.3.1.2 An Illustration

Let us look at an example of a dynamic program with preference shocks used in structural estimation, which is taken from a study of labor supply by married women (Keane et al., 2011). The husband of the decision maker, a married woman, is already working. The couple has young children and the mother is deciding whether to work. Her utility function is

$$u(c, d, \xi) = c + (\alpha n + \xi)(1 - d),$$

where c is consumption, α is a parameter, n is the number of children, ξ is a preference shock and d is the action variable. The action is binary, with $d = 1$ representing the decision to work in the current period and $d = 0$ representing the decision not to work.[5]

The budget constraint for the household is

$$c_t = f_t + w_t d_t - \pi n d_t,$$

where f_t is the father's income, w_t is the mother's wage and π is the cost of child care. Wages depend on human capital h_t, which increases with experience. In particular,

$$w_t = \gamma h_t + \eta_t, \quad \text{with} \quad h_t = h_{t-1} + d_{t-1}.$$

Here η_t is random and γ is a parameter. We assume that $(f_t)_{t \geqslant 0}$ is F-Markov on some finite set. In the model, $(\xi_t)_{t \geqslant 0}$ and $(\eta_t)_{t \geqslant 0}$ are IID. We denote their joint distribution by φ.

With constant discount factor β, the problem of maximizing expected discounted utility is an MDP with the Bellman equation

$$v(f, h, \xi) = \max_d \left\{ c + (\alpha n + \xi)(1 - d) + \beta \sum_{f', \xi', \eta'} v(f', h + d, \xi', \eta') F(f, f') \varphi(\xi', \eta') \right\}.$$

While we can proceed directly with a technique such as VFI to obtain optimal choices, we can simplify.

[5] Here, the woman is the primary carer of the child; she derives no utility from children in periods in which she works. See Keane et al. (2011) for further discussion.

One way is by reducing the number of states. A hint comes from looking at the expected value function

$$g(f,h,d) := \sum_{f',\xi',\eta'} v(f',h+d,\xi',\eta')F(f,f')\varphi(\xi',\eta')$$

While this function also depends on three arguments, we know that d is binary. Hence we can break it down into two functions $g(f,h,0)$ and $g(f,h,1)$, each of which depends only on the pair (f,h).

These functions are substantially simpler than v when the domain of ξ is large. Hence, it is natural to consider whether we can solve our problem using g rather than v.

5.3.1.3 Expected Value Functions

Rather than address this question within the context of the preceding model, let's shift to a generic version of the dynamic program used in structural estimation and how it can be solved using expected value methods. Our generic version takes the form

$$v(y,\varepsilon) = \max_{a\in\Gamma(y)} \left\{ r(y,\varepsilon,a) + \beta \sum_{y'} \int v(y',\varepsilon')P(y,a,y')\varphi(\varepsilon')\,\mathrm{d}\varepsilon' \right\} \tag{5.32}$$

for all $y \in \mathsf{Y}$ and $\varepsilon \in \mathsf{E}$. Here Y is a finite set, often determined by discretization of a continuous space, while E, the outcome space for ε, is allowed to be continuous. The state y will be called the endogenous state, while ε is the preference shock. In practice, ε will often be a vector of shocks that affect current rewards. The integral can therefore be multivariate and is over all of E.

The problem represented by (5.32) is a version of a regular MDP, with state $x = (y,\varepsilon)$ taking values in $\mathsf{X} := \mathsf{Y} \times \mathsf{E}$. If we discretize the space E, then all the optimality theory for MDPs applies. Instead of taking this approach, however, we draw on our discussion of labor choice in §5.3.1.2. In particular, to enhance efficiency, we will work with the **expected value function**

$$g(y,a) := \sum_{y'} \int v(y',\varepsilon')P(y,a,y')\varphi(\varepsilon')\,\mathrm{d}\varepsilon' \tag{5.33}$$

There are several potential advantages associated with working with g rather than v. One is that the set of actions A can be much smaller than the set of states that would be created by discretization of the preference shock space E (especially if ε_t takes values

in a high-dimensional space). Another is that the integral provides smoothing, so that g is typically a smooth function. This can accelerate structural estimation procedures.

5.3.1.4 Optimality via EV Methods

To exploit the relative simplicity of the expected value function, we rewrite the Bellman equation (5.32) as

$$v(y,\varepsilon) = \max_{a\in\Gamma(y)} \{r(y,\varepsilon,a) + \beta g(y,a)\}.$$

Taking expectations of both sides and using (5.33) again gives

$$g(y,a) = \sum_{y'} \int \max_{a'\in\Gamma(y')} \{r(y',\varepsilon',a') + \beta g(y',a')\}\, \varphi(\varepsilon')\, \mathrm{d}\varepsilon' P(y,a,y').$$

To solve this functional equation we introduce the **expected value Bellman operator** R defined at $g \in \mathbb{R}^{\mathsf{G}}$ by

$$(Rg)(y,a) = \sum_{y'} \int \max_{a'\in\Gamma(y')} \{r(y',\varepsilon',a') + \beta g(y',a')\}\, \varphi(\varepsilon')\, \mathrm{d}\varepsilon' P(y,a,y'). \tag{5.34}$$

Here G is the set of feasible state-action pairs (y,a).

Exercise 5.3.1. Prove that R is order-preserving and a contraction of modulus β on $\mathbb{R}^{\mathsf{G}}$ (with respect to the supremum norm).

In what follows, we let g^* be the fixed point of R in $\mathbb{R}^{\mathsf{G}}$. Since R is a contraction map, g^* can be computed by successive approximation. The next result shows that knowing this fixed point is enough to solve the dynamic program.

Proposition 5.3.1. *A policy $\sigma \in \Sigma$ is optimal if and only if*

$$\sigma(y,\varepsilon) \in \operatorname*{argmax}_{a\in\Gamma(y)} \{r(y,\varepsilon,a) + \beta g^*(y,a)\} \qquad \textit{for all } (y,\varepsilon) \in \mathsf{Y}\times\mathsf{E}.$$

We postpone proving Proposition 5.3.1 until §5.3.5, where we prove a more general result.

Example 5.3.2. In the labor supply problem in §5.3.1.2, the expected value Bellman

operator becomes

$$(Rg)(f,h,d) = \sum_{f',\xi',\eta'} \max_{d'} \{c + (\alpha n + \xi)(1-d') + \beta g(f', h+d, d')\} F(f,f')\varphi(\xi',\eta').$$

Iterating from an arbitrary guess of g converges to the unique fixed point g^* of R. By Proposition 5.3.1, we can then compute the optimal policy σ^* at (f,h) by taking

$$\sigma^*(f,h) \in \operatorname*{argmax}_d \{c + (\alpha n + \xi)(1-d') + \beta g^*(f,h,d)\}.$$

5.3.2 The Gumbel Max Trick

§5.3.1.3 described how using expected values can reduce dimensionality by smoothing. But there is another feature of an expected value formulation of a Bellman equation that we can take advantage of when we are prepared to impose extra structure on preference shocks. This section provides details.

A real-valued random variable Z is said to have a **Gumbel distribution** (or a "type 1 generalized extreme value distribution") with mean $\mu \in \mathbb{R}$ if its cumulative distribution function takes the form $F(z) = \exp(-\exp(z-\mu))$. To denote a random variable with a Gumbel distribution, we write $Z \sim G(\mu)$. The expectation of Z is $\mu + \gamma$, where $\gamma \approx 0.577$ is the **Euler–Mascheroni constant**.

Exercise 5.3.2. Prove: if $Z \sim G(\mu)$ and $\lambda \in \mathbb{R}$, then $Z + \lambda \sim G(\mu + \lambda)$.

The Gumbel distribution has the following useful stability property, a proof of which can be found in Huijben et al. (2022).

Lemma 5.3.2. *If $Z_1, \ldots, Z_k \overset{\text{IID}}{\sim} G(0)$ and $c_1, \ldots, c_k$ are real numbers, then*

$$\max_{1 \leqslant i \leqslant k} (Z_i + c_i) \sim G\left\{-\gamma + \ln\left[\sum_{i=1}^{k} \exp(c_i)\right]\right\}.$$

To exploit Lemma 5.3.2, we continue the discussion in the previous section but assume now that $\mathsf{A} = \{a_1, \ldots, a_k\}$, that $\Gamma(y') = \mathsf{A}$ for all y' (so that actions are unrestricted), that ε' in (5.34) is additive in rewards and indexed by actions, so that $r(y', \varepsilon', a') = r(y', a') + \varepsilon'(a')$ for all feasible (y', a'), and that, conditional on y', the vector $(\varepsilon(a_1), \ldots, \varepsilon(a_k))$ consists of k independent $G(0)$ shocks. Thus, each feasible choice returns a rewards perturbed by an independent Gumbel shock.

From these assumptions and Lemma 5.3.2, the term inside the integral in (5.34) satisfies

$$\max_{a'} \{r(y', \varepsilon', a') + \beta g(y', a')\} = \max_{a'} \{r(y', a') + \varepsilon'(a') + \beta g(y', a')\}$$
$$\sim G\left\{-\gamma + \ln\left[\sum_{a'} \exp\left(r(y', a') + \beta g(y', a')\right)\right]\right\}$$

Recalling our rule for computing mathematical expectations of Gumbel distributed random variables, the expected value Bellman operator R in (5.34) becomes

$$(Rg)(y, a) = \sum_{y'} \ln\left[\sum_{a'} \exp\left(r(y', a') + \beta g(y', a')\right)\right] P(y, a, y'). \tag{5.35}$$

This operator is convenient because the absence of a max operator permits fast evaluation. Notice also that R is smooth in g, which suggests that we can use gradient information to compute its fixed points.

Proposition 5.3.3. *The operator R in (5.35) is a contraction of modulus β on $\mathbb{R}^{\mathsf{G}}$.*

Proof. The operator R is order-preserving on $\mathbb{R}^{\mathsf{G}}$. Straightforward algebra shows that, for $c \in \mathbb{R}_+$ and $g \in \mathbb{R}^{\mathsf{G}}$, we have $R(g + c) = Rg + \beta c$. The claim now follows from Blackwell's sufficient condition for a contraction (page 49). □

Notice how the Gumbel max trick that exploits Lemma 5.3.2 depends crucially on the expected value formulation of the Bellman equation, rather than the standard formulation (5.32). This is because the expected value formulation puts the max inside the expectation operator, unlike the standard formulation, where the max is on the outside.

Variations of the Gumbel max trick have many uses in structural econometrics (see §5.4).

5.3.3 Optimal Savings with Stochastic Returns on Wealth

We modify the §5.2.2 optimal savings problem by replacing a constant gross rate of interest R by an IID sequence $(\eta_t)_{t \geqslant 0}$ with common distribution φ on finite set E. So the consumer faces a fluctuating rate of returns on financial wealth. In each period t, the consumer knows η_t, the gross rate of interest between t and $t+1$, before deciding how much to consume and how much to save. Other aspects of the problem are unchanged.

We have two motivations. One is computational, namely, to illustrate how framing a decision in terms of expected values can reduce dimensionality, analogous to the results in §5.3.1.4. The other is to generate a more realistic wealth distribution than that generated by the §5.2.2.4 optimal savings model.

With stochastic returns on wealth, the Bellman equation becomes

$$v(w, y, \eta) = \max_{w' \leqslant \eta(w+y)} \left\{ u\left(w + y - \frac{w'}{\eta}\right) + \beta \sum_{y', \eta'} v(w', y', \eta') Q(y, y') \varphi(\eta') \right\}.$$

Both w and w' are constrained to a finite set $\mathsf{W} \subset \mathbb{R}_+$. The expected value function can be expressed as

$$g(y, w') := \sum_{y', \eta'} v(w', y', \eta') Q(y, y') \varphi(\eta'). \tag{5.36}$$

In the remainder of this section, we will say that a savings policy σ is **g-greedy** if

$$\sigma(y, w, \eta) \in \operatorname*{argmax}_{w' \leqslant \eta(w+y)} \left\{ u\left(w + y - \frac{w'}{\eta}\right) + \beta g(y, w') \right\}.$$

Since it is an MDP, we can see immediately that if we replace v in (5.36) with the value function v^*, then a g-greedy policy will be an optimal one.

Using manipulations analogous to those we used in §5.3.1.4, we can rewrite the Bellman equation in terms of expected value functions via

$$g(y, w') = \sum_{y', \eta'} \max_{w'' \leqslant \eta'(w'+y')} \left\{ u\left(w' + y' - \frac{w''}{\eta'}\right) + \beta g(y', w'') \right\} Q(y, y') \varphi(\eta').$$

From here we could proceed by introducing an expected value Bellman operator analogous to η in (5.34), proving it to be a contraction map and then showing that greedy policies taken with respect to the fixed point are optimal. All of this can be accomplished without too much difficulty—we prove more general results in §5.3.5.

However, we also know that optimistic policy iteration (OPI) is, in general, more efficient than value function iteration. This motivates us to introduce the modified σ-value operator

$$(R_\sigma g)(y, w') = \sum_{y', \eta'} \left\{ u\left(w' + y' - \frac{\sigma(w', y', \eta')}{\eta'}\right) + \beta g(y', \sigma(w', y', \eta')) \right\} Q(y, y') \varphi(\eta').$$

This is a modification of the regular σ-value operator T_σ that makes it act on expected

```
using QuantEcon, LinearAlgebra, IterTools

function create_savings_model(; β=0.98, γ=2.5,
                                w_min=0.01, w_max=20.0, w_size=100,
                                ρ=0.9, ν=0.1, y_size=20,
                                η_min=0.75, η_max=1.25, η_size=2)
    η_grid = LinRange(η_min, η_max, η_size)
    ϕ = ones(η_size) * (1 / η_size)  # Uniform distributoin
    w_grid = LinRange(w_min, w_max, w_size)
    mc = tauchen(y_size, ρ, ν)
    y_grid, Q = exp.(mc.state_values), mc.p
    return (; β, γ, η_grid, ϕ, w_grid, y_grid, Q)
end
```

Listing 20: Optimal savings parameters (`modified_opt_savings.jl`)

value functions.

A suitably modified OPI routine that is adapted from the regular OPI algorithm in §5.1.4.4 can be found in Algorithm 7 on page 175. The routine is convergent. We discuss this in greater detail in §5.3.5.

Figure 5.14 shows a histogram of a long wealth time series that parallels Figure 5.10 on page 152. The only significant difference is the switch to stochastic returns (as described above). Parameters are as in Listing 20. Now the wealth distribution has a more realistic long right tail (a few observations are in the far right tail, although they are difficult to see). Now the Gini coefficient is 0.72, which is closer to typical country values recorded in World Bank data (but still lower than the US). In essence, this occurs because return shocks are have multiplicative rather than additive effects on wealth, so a sequence of high draws compounds to make wealth grow fast.

Exercise 5.3.3. Consider a version of the optimal savings problem from §5.2.2 where labor income has both persistent and transient components. In particular, assume that $Y_t = Z_t + \varepsilon_t$ for all t, where $(\varepsilon_t)_{t \geqslant 0}$ is IID with common distribution φ on E, while $(Z_t)_{t \geqslant 0}$ is Q-Markov on Z. Such a specification of labor income can capture how households should react differently to transient and "permanent" shocks (see §5.4 for more discussion). Following the pattern developed for the savings model with stochastic returns on wealth, write down both the Bellman equation and the Bellman equation in terms of expected value functions.

Figure 5.14: Histogram of wealth (stochastic returns)

5.3.4 Q-Factors

Q-factors assign values to state-action pairs. They set the stage for Q-learning, an application of reinforcement learning, a recursive algorithm for estimating parameters. Q-learning uses stochastic approximation techniques to learn Q-factors. Under special conditions Q-learning eventually learns optimal Q-factors for a finite MDP.

Q-learning is connected to the topic of this chapter because it relies on a Bellman operator for the Q-factor. We discuss that Bellman operator, but we don't discuss Q-learning here.

To begin, we fix an MDP (Γ, β, r, P) with state space X and action space A. For each $v \in \mathbb{R}^{\mathsf{X}}$, the **$Q$-factor** corresponding to v is the function

$$q(x, a) = r(x, a) + \beta \sum_{x'} v(x') P(x, a, x') \qquad ((x, a) \in \mathsf{G}).$$

We can convert the Bellman equation into an equation in Q-factors by observing that, given such a q, the Bellman equation can be written as $v(x) = \max_{a \in \Gamma(x)} q(x, a)$. Taking the mean and discounting on both sides of this equation gives

$$\beta \sum_{x'} v(x') P(x, a, x') = \beta \sum_{x'} \max_{a' \in \Gamma(x')} q(x', a') P(x, a, x').$$

Adding $r(x, a)$ and using the definition of q again gives

$$q(x, a) = r(x, a) + \beta \sum_{x'} \max_{a' \in \Gamma(x')} q(x', a') P(x, a, x').$$

This functional equation motivates us to introduce the **Q-factor Bellman operator**

$$(Sq)(x, a) = r(x, a) + \beta \sum_{x'} \max_{a' \in \Gamma(x')} q(x', a') P(x, a, x') \qquad ((x, a) \in \mathsf{G}). \tag{5.38}$$

Exercise 5.3.4. Prove that S is order-preserving and a contraction of modulus β on $\mathbb{R}^{\mathsf{G}}$ (with respect to the supremum norm).

Let q^* be the unique fixed point of S in $\mathbb{R}^{\mathsf{G}}$.

Proposition 5.3.4. *A policy $\sigma \in \Sigma$ is optimal if and only if*

$$\sigma(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} q^*(x, a) \qquad \textit{for all } (x, a) \in \mathsf{G}.$$

Enthusiastic readers might like to try to prove Proposition 5.3.4 directly. We defer the proof until §5.3.5, where a more general result is obtained.

5.3.5 Operator Factorizations

Our study of structural estimation in §5.3.1, optimal savings in §5.3.3 and Q-factors in §5.3.4 all involved manipulations of the Bellman and policy operators that presented alternative perspectives on the respective optimization problems. Rather than offering additional applications that apply such ideas, we now develop a general theoretical framework from which to understand manipulations of the Bellman and policy operators for general MDPs. The framework clarifies when and how these techniques can be applied.

5.3.5.1 Refactoring the Bellman Operator

Fix an MDP (Γ, β, r, P) with state space X and action space A. As usual, Σ is the set of feasible policies, G is the set of feasible state, action pairs, T is the Bellman operator and v^* denotes the value function. Our first step is to decompose T. To do this we introduce three auxiliary operators:

Figure 5.15: Multiple Bellman operators (EV = expected value)

- $E\colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{G}}$ defined by $(Ev)(x,a) = \sum_{x'} v(x')P(x,a,x')$,
- $D\colon \mathbb{R}^{\mathsf{G}} \to \mathbb{R}^{\mathsf{G}}$ defined by $(Dg)(x,a) = r(x,a) + \beta g(x,a)$ and
- $M\colon \mathbb{R}^{\mathsf{G}} \to \mathbb{R}^{\mathsf{X}}$ defined by $(Mq)(x) = \max_{a\in\Gamma(x)} q(x,a)$.

Evidently the action of the Bellman operator T on a given $v \in \mathbb{R}^{\mathsf{X}}$ is the composition of these three steps:

(i) take conditional expectations given $(x,a) \in \mathsf{G}$ (applying E),

(ii) discount and adding current rewards (applying D), and

(iii) maximize with respect to current action (applying M).

As a result, we can write $T = MDE := M \circ D \circ E$ (apply E first, D second, and M third). This decomposition is visualized in Figure 5.15. The action of T is a round trip from the top node, which is the set of value functions.

If we stare at Figure 5.15, we can imagine two other round trips. One is a round trip from the set of expected value functions, obtained by the sequence EMD. The other is a round trip from the set of Q-factors, obtained by the sequence DEM. Let's name these additional round trips R and S respectively, so that, collecting all three,

$$R = EMD, \quad S = DEM, \quad T = MDE. \tag{5.39}$$

Both R and S act on functions in $\mathbb{R}^{\mathsf{G}}$. The next exercise provides an explicit representation of these operators.

Exercise 5.3.5. Show that for any $g, q \in \mathbb{R}^{\mathsf{G}}$ and $(x, a) \in \mathsf{G}$ we have

$$(Rg)(x,a) = \sum_{x'} \max_{a' \in \Gamma(x')} \{r(x',a') + \beta g(x',a')\} P(x,a,x') \quad \text{and}$$
$$(Sq)(x,a) = r(x,a) + \beta \sum_{x'} \max_{a' \in \Gamma(x')} q(x',a') P(x,a,x').$$

Let's connect our "refactored" Bellman operators R and S to our preceding examples. Inspection of (5.38) confirms that S is exactly the Q-factor Bellman operator. In addition, R is a general version of the expected value Bellman operator defined in (5.34).

Exercise 5.3.6. Show that, for all $k \in \mathbb{N}$, the following relationships hold

- $R^k = ET^{k-1}MD = EMS^{k-1}D$
- $S^k = DR^{k-1}ME = MT^{k-1}ED$
- $T^k = MS^{k-1}DE = MDR^{k-1}E$

(Here, for any operator A, we take A^0 to be the identity map.)

While the equalities in Exercise 5.3.6 can be proved by induction via the logic revealed by (5.39), the intuition is straightforward from Figure 5.15. For example, the relationship $R^k = ET^{k-1}MD$ states that round-tripping k times from the space of expected values (EV function space) is the same as shifting to value function space by applying MD, round-tripping $k-1$ times using T, and then shifting one more step to EV function space via E.

Although the relationships in Exercise 5.3.6 are easy to prove, they are already useful. For example, suppose that in a computational setting R is easier to iterate than T. Then to iterate with T k times, we can instead use $T^k = MDR^{k-1}E$: We apply E once, R $k-1$ times, and M and D once each. If k is large, this might be more efficient.

In the next exercise and the next section, we let $\|\cdot\| := \|\cdot\|_\infty$, the supremum norm on either $\mathbb{R}^{\mathsf{X}}$ or $\mathbb{R}^{\mathsf{G}}$.

Exercise 5.3.7. Prove the following facts:

(i) $\|Ev - Ev'\| \leqslant \|v - v'\|$ for all $v, v' \in \mathbb{R}^{\mathsf{X}}$,

(ii) $\|Mg - Mg'\| \leqslant \|g - g'\|$ for all $g, g' \in \mathbb{R}^{\mathsf{G}}$, and

(iii) $\|Dq - Dq'\| \leqslant \beta\|q - q'\|$ for all $q, q' \in \mathbb{R}^{\mathsf{G}}$.

We can say that E and M are **nonexpansive** on $\mathbb{R}^{\mathsf{X}}$ and $\mathbb{R}^{\mathsf{G}}$ respectively, while D is a contraction on $\mathbb{R}^{\mathsf{G}}$.

Lemma 5.3.5. *The operators R, S and T are all contraction maps of modulus β under the supremum norm.*

Proof. That T is a contraction of modulus β was proved in Proposition 5.1.1, on page 134. We can prove this more easily now by applying Exercise 5.3.7, which, for arbitrary $v, v' \in \mathbb{R}^{\mathsf{X}}$, gives

$$\|Tv - Tv'\| = \|MDEv - MDEv'\| \leqslant \|MDv - MDv'\| \leqslant \beta\|Mv - Mv'\| \leqslant \beta\|v - v'\|.$$

Proofs for $R = EMD$ and $S = DEM$ are similar. □

In the next section we clarify relationships between these operators and prove Propositions 5.3.1 and 5.3.4.

5.3.5.2 Refactorizations and Optimality

From Lemma 5.3.5 we see that R, S and T all have unique fixed points. We denote them by g^*, q^* and v^* respectively, so that

$$Rg^* = g^*, \quad Sq^* = q^*, \quad \text{and} \quad Tv^* = v^*.$$

We already know that v^* is the value function (Proposition 5.1.1). The results below show that the other two fixed points are, like the value function, sufficient to determine optimality.

Proposition 5.3.6. *The fixed points of R, S and T are connected by the following relationships:*

(i) $g^* = Ev^*$,

(ii) $q^* = Dg^*$, *and*

(iii) $v^* = Mq^*$.

Proof. To prove (i), first observe that, in the notation of (5.39), we have $Ev^* = ETv^* = EMDEv^* = REv^*$. Hence Ev^* is a fixed point of R. But R has only one fixed point, which is g^*. Therefore, $g^* = Ev^*$. The proofs of (ii) and (iii) are analogous. □

The results in Proposition 5.3.6 can be written more explicitly as

- $g^*(x,a) = \sum_{x'} v^*(x')P(x,a,x')$ for all $(x,a) \in \mathsf{G}$,
- $q^*(x,a) = r(x,a) + \beta g^*(x,a)$ for all $(x,a) \in \mathsf{G}$, and
- $v^*(x) = \max_{a\in\Gamma(x)} q^*(x,a)$ for all $x \in \mathsf{X}$.

In the next result and the discussion that follows, given $g, q \in \mathbb{R}^{\mathsf{G}}$, we will call $\sigma \in \Sigma$

- **g-greedy** if $\sigma(x) \in \operatorname{argmax}_{a\in\Gamma(y)} \{r(x,a) + \beta g(x,a)\}$ for all $x \in \mathsf{X}$, and
- **q-greedy** if $\sigma(x) \in \operatorname{argmax}_{a\in\Gamma(y)} q(x,a)$ for all $x \in \mathsf{X}$.

These definitions are exact analogs of the v-greedy concept, applied to expected value functions and Q-factors respectively.

Corollary 5.3.7. *For $\sigma \in \Sigma$, the following statements are equivalent:*

(i) *σ is v-greedy when $v = v^*$.*

(ii) *σ is g-greedy when $g = g^*$.*

(iii) *σ is q-greedy when $q = q^*$.*

In particular, σ is optimal if and only if any one (and hence all) of (i)–(iii) holds.

Proof. To see that (i) implies (ii), suppose that σ is v-greedy when $v = v^*$. Then for arbitrary $x \in \mathsf{X}$

$$\begin{aligned}\sigma(x) &\in \operatorname*{argmax}_{a\in\Gamma(y)} \left\{ r(x,a) + \beta \sum_{x'} v^*(x')P(x,a,x') \right\} \\ &= \operatorname*{argmax}_{a\in\Gamma(y)} \{ r(x,a) + \beta g^*(x,a) \} .\end{aligned}$$

Hence σ is g-greedy when $g = g^*$, and (i) implies (ii). The proofs of the remaining equivalences (ii) $\implies$ (iii) $\implies$ (i) are similar. The claim that σ is optimal if and only if if any one of (i)–(iii) holds now follows from Proposition 5.1.1, which asserts that σ is optimal if and only if σ is v^*-greedy. □

Notice that Proposition 5.3.4 is a special case of Corollary 5.3.7.

The results in Corollary 5.3.7 can be understood as "refactored" versions of Bellman's principle of optimality. A consequence of these results is that we can solve a given MDP by modifying VFI to operate either on expected value functions or on Q-factors. For example, if we find it more convenient to iterate in expected value space, then (informally) we can proceed as follows:

(i) Fix $g \in \mathbb{R}^{\mathsf{G}}$.

(ii) Iterate with R to obtain $g_k := R^k g \approx g^*$.

(iii) Compute a g_k-greedy policy.

Since $g_k \approx g^*$, the resulting policy will be approximately optimal.

5.3.5.3 Refactored OPI

In Chapter 5 we found that VFI is often outperformed by HPI or OPI. Our next step is to apply these methods to modified versions of the Bellman equation, as discussed in the previous section. This allows us to combine advantages of HPI/OPI with the potential efficiency gains obtained by refactoring the Bellman equation.

We illustrate these ideas below by producing a version of OPI that can compute Q-factors and expected value functions. (The same is true for HPI, although we leave details of that construction to interested readers.)

To begin, we introduce a new operator, denoted M_σ, that, for fixed $\sigma \in \Sigma$ and $q \in \mathbb{R}^{\mathsf{G}}$, produces

$$(M_\sigma q)(x) := q(x, \sigma(x)) \qquad (x \in \mathsf{X}).$$

This operator is the policy analog of the maximization operator M defined by $(Mq)(x) = \max_{a \in \Gamma(x)} q(x, a)$ in §5.3.5.1. Analogous to (5.39), we set

$$R_\sigma := E\, M_\sigma\, D, \quad S_\sigma := D\, E\, M_\sigma, \quad T_\sigma := M_\sigma\, D\, E.$$

You can verify that T_σ is the ordinary σ-policy operator (defined in (5.19)). The operators R_σ and S_σ are the expected value and Q-factor equivalents.

Exercise 5.3.8. The relationships in Exercise 5.3.6 continue to hold after we swap R, S, T, M with $R_\sigma, S_\sigma, T_\sigma, M_\sigma$. Confirm the first of these relationships, showing in particular that

$$R_\sigma^k = E\, T_\sigma^{k-1}\, M_\sigma\, D \qquad \text{for all } k \in \mathbb{N}. \tag{5.40}$$

Let's now show that OPI can be successfully modified via these alternative operators. We will focus on the expected value viewpoint (value functions are replaced by expected value functions), which is often helpful in the applications we wish to consider.

Our modified OPI routine is given in Algorithm 7. It makes the obvious modifications to regular OPI, switching to working with expected value functions in $\mathbb{R}^{\mathsf{G}}$ and from iteration with T_σ to iteration with R_σ.

Algorithm 7: Refactored OPI for expected value functions

```
input g_0 ∈ R^G, an initial guess of g*
input τ, a tolerance level for error
input m ∈ N, a step size
k ← 0
ε ← τ + 1
while ε > τ do
    σ_k ← a g_k-greedy policy
    g_{k+1} ← R^m_{σ_k} g_k
    ε ← ||g_k − g_{k+1}||_∞
    k ← k + 1
end
return σ_k
```

Algorithm 7 is globally convergent in the same sense as regular OPI (Algorithm 6 on page 142). In fact, if we pick $v_0 \in \mathbb{R}^{\mathsf{X}}$ and apply regular OPI with this initial condition, as well as applying Algorithm 7 with initial condition $g_0 := Ev_0$, then the sequences $(v_k)_{k\geqslant 0}$ and $(g_k)_{k\geqslant 0}$ generated by the two algorithms are connected via $g_k = Ev_k$ for all $k \geqslant 0$. If greedy policies are unique, then it is also true that the policy sequences generated by the two algorithms are identical.

Let's prove these claims, assuming for convenience that greedy policies are unique. Consider first the claim that $g_k = Ev_k$ for all $k \geqslant 0$. This is true by assumption when $k = 0$. Suppose, as an induction hypothesis, that $g_k = Ev_k$ holds at arbitrary k. Let σ be g_k-greedy. Then

$$\sigma(x) = \operatorname*{argmax}_{a\in\Gamma(y)} \{r(x,a) + \beta g_k(x,a)\} = \operatorname*{argmax}_{a\in\Gamma(y)} \left\{ r(x,a) + \beta \sum_{x'} v_k(x')P(x,a,x') \right\},$$

where the second equality is implied by $g_k = Ev_k$. Hence σ is both g_k-greedy and v_k-greedy and so is the next policy selected by both modified and regular OPI. Moreover, updating via Algorithm 7 and applying (5.40), we have

$$g_{k+1} = R_\sigma^m g_k = ET_\sigma^{m-1} M_\sigma D g_k = ET_\sigma^{m-1} M_\sigma D E v_k = ET_\sigma^m v_k.$$

Since σ is v_k-greedy, $T_\sigma^m v_k$ is the next function selected by regular OPI. Hence $v_{k+1} = T_\sigma^m v_k$. Connecting with the last chain of equalities yields $g_{k+1} = Ev_{k+1}$. This completes the proof that $g_k = Ev_k$ for all k. Policy functions generated by the algorithms are identical as well.

The preceding discussion provides a justification for the modified OPI algorithm

we adopted in §5.3.3.

5.4 Chapter Notes

Detailed treatment of MDPs can be found in Howard (1960), Bellman (1957), Denardo (1981), Puterman (2005), Bertsekas (2012), Hernández-Lerma and Lasserre (2012a, 2012b), and Kochenderfer et al. (2022). The books by Hernández-Lerma and Lasserre (2012a, 2012b) provide excellent coverage of theory, while Puterman (2005) gives a clear and detailed exposition of algorithms and techniques. Further discussion of the connection between HPI and Newton iteration can be found in Section 6.4 of Puterman (2005).

HPI is routinely used in artificial intelligence applications, including during the training of AlphaZero by DeepMind. Further discussion of these variants of HPI and their connection to Newton iteration can be found in Bertsekas (2021) and Bertsekas (2022a).

The optimal savings problem is a workhorse in macroeconomics and has been treated extensively in the literature. Early references include Brock and Mirman (1972), Mirman and Zilcha (1975), Schechtman (1976), Deaton and Laroque (1992), and Carroll (1997). For more recent studies, see, for example, Li and Stachurski (2014), Açıkgöz (2018), Light (2018), Lehrer and Light (2018), or Ma et al. (2020). Recent applications involving optimal savings in a representative agent framework include Bianchi (2011), Paciello and Wiederholt (2014), Rendahl (2016), Heathcote and Perri (2018), Paroussos et al. (2019), Erosa and González (2019), Herrendorf et al. (2021), and Michelacci et al. (2022). For more on the long right tail of the wealth distribution (as discussed in §5.3.3), see, for example, Benhabib et al. (2015), Krueger et al. (2016), or Stachurski and Toda (2019).

Households solving optimal savings problems are often embedded in heterogeneous agent models in order to study income distributions, wealth distributions, business cycles and other macroeconomic phenomena. Representative examples include Aiyagari (1994), Huggett (1993), Krusell and Smith (1998), Miao (2006), Algan et al. (2014), Toda (2014), Benhabib et al. (2015), Stachurski and Toda (2019), Toda (2019), Light (2020), Hubmer et al. (2020), or Cao (2020). A typical example of continuous time heterogeneous agent models can be found in Kaplan et al. (2018).

Exercise §5.3.3 considered optimal savings and consumption in the present of transient and persistent shocks to labor income. For research in this vein, see, for example, Quah (1990), Carroll (2009), De Nardi et al. (2010), or Lettau and Ludvigson (2014). For empirical work on labor income dynamics, see, for example, Newhouse (2005),

Guvenen (2007), Guvenen (2009), or Blundell et al. (2015). For analysis of optimal savings in a very general setting, see Ma et al. (2020) or Ma and Toda (2021).

The optimal investment problem dates back to Lucas and Prescott (1971). Textbook treatments can be found in Stokey and Lucas (1989) and Dixit and Pindyck (2012). Sargent (1980) and Hayashi (1982) used optimal investment problems to connect optimal capital accumulation with Tobin's q (which is the ratio between a physical asset's market value and its replacement value). Other influential papers in the field include Lee and Shin (2000), Hassett and Hubbard (2002), Bloom et al. (2007), Bond and Van Reenen (2007), Bloom (2009), and Wang and Wen (2012). Carruth et al. (2000) contains a survey.

Classic papers about S-s inventory models include Arrow et al. (1951) and Dvoretzky et al. (1952). Optimality of S-s policies under certain conditions was first established by Scarf (1960). Kelle and Milne (1999) study the impact of S-s inventory policies on the supply chain, including connection to the "bullwhip" effect. The connection between S-s inventory policies and macroeconomic fluctuations is studied in Nirei (2006).

The model in Exercise 5.2.3 is loosely adapted from Bagliano and Bertola (2004).

Rust (1994) is a classic and highly readable reference in the area of structural estimation of MDPs. Keane and Wolpin (1997) provides an influential study of the career choices of young men. Roberts and Tybout (1997) analyze the decision to export in the presence of sunk costs. Keane et al. (2011) provide an overview of structural estimation applied to labor market problems. Gentry et al. (2018) review analysis of auctions using structural estimation. Legrand (2019) surveys the use of structural models to study the dynamics of commodity prices. Calsamiglia et al. (2020) use structural estimation to study school choices. Iskhakov et al. (2020) provide a thoughtful discussion on the differences between structural estimation and machine learning. Luo and Sang (2022) propose structural estimation via sieves.

Theoretical analysis of expected value functions in discrete choice models and other settings can be found in Rust (1994), Norets (2010), Mogensen (2018) and Kristensen et al. (2021). The expected value Gumbel max trick is due to Rust (1987) and builds on work by McFadden (1974). The Gumbel max trick is also used in machine learning methods (see, e.g., Jang et al. (2016)).

In §5.3.4 we mentioned Q-learning, which was originally proposed by Watkins (1989). Tsitsiklis (1994) and Melo (2001) studied convergence of Q-learning.

The theory in §5.3.5 on optimality under modifications of the Bellman equation is loosely based on Ma and Stachurski (2021). That paper considers arbitrary modifications in a very general setting.

Chapter 6

Stochastic Discounting

A recent trend in economics is use of time-varying discount factors to bring models closer to data. In this chapter we provide background discussion and then show how to extend the MDP model to handle time-varying discount factors.

6.1 Time-Varying Discount Factors

In this section we introduce lifetime valuation under stochastic discounting and provide necessary and sufficient conditions for existence of finite solutions to infinite sums that represent these valuations.

6.1.1 Valuation

Our first step is to motivate and understand lifetime valuation when discount factors vary over time.

6.1.1.1 Motivation

In §3.2.3.2 we discussed firm valuation in a setting where the interest rate is constant. A quick look at data belies this assumption. Interest rates are stochastic and time-varying, even for safe assets like US Treasury bills. To illustrate this, Figure 6.1 shows the nominal interest rate on 1 Year Treasury bills since the 1950s, while Figure 6.2 shows dynamics of the real interest rate for 10 year T-bills since 2012. Both the nominal and the real interest rates are significantly time varying.

Figure 6.1: Nominal US interest rates (`plot_interest_rates_nominal.jl`)

Figure 6.2: Real US interest rates (`plot_interest_rates_real.jl`)

Remark 6.1.1. Should a given firm's profit stream be discounted by nominal or real interest rates? The answer depends on the costs and revenue streams of the firm, and how closely they co-move with inflation. In fact many analysts use another alternative called "weighted average cost of capital," which measures average cost of raising funds from bonds, common stock, and other sources. All of these alternative discount rates exhibit significant variation over time.

Example 6.1.1. When a period of rising interest rates is anticipated by the market, the share prices of newer and more technology-heavy firms typically face strong headwinds. This is because the profit streams from such firms are usually tilted towards the future, in the sense that dividends are initially low or zero (while profits are reinvested) and eventually high (if the business model is successful). A period of rising interest rates indicates that such profit streams should be heavily discounted.

We have just motivated time-varying discount factors for the firm problem. How about the consumer problem discussed in §3.2.3.3? Here, too, time-varying discount factors have become popular in quantitative work, particularly in macroeconomics. (An early example is Krusell and Smith (1998) and more citations are provided in §7.3.) One motivation is to inject demand-side shocks into models of business cycles and asset prices.

6.1.1.2 Theory

In order to handle time-varying discount factors, we now present and prove a general result on geometric sums. Throughout this section, we work in the following setting:

- X is a finite set, $P \in \mathbb{M}(\mathsf{X})$, and $(X_t)_{t \geqslant 0}$ is P-Markov.
- h is in $\mathbb{R}^{\mathsf{X}}$ and b is a map from $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}$.
- A is the matrix on X defined by $A(x, x') := b(x, x')P(x, x')$.

With the convention $\prod_{i=1}^{0} b(X_{i-1}, X_i) = 1$, we have the following key result.

Theorem 6.1.1. *If $r(A) < 1$, then*

$$v(x) := \mathbb{E}_x \sum_{t=0}^{\infty} \left\{ \prod_{i=1}^{t} b(X_{i-1}, X_i) \right\} h(X_t) \tag{6.1}$$

is finite for all $x \in \mathsf{X}$. Moreover, $I - A$ is nonsingular and

$$v = (I - A)^{-1} h. \tag{6.2}$$

Theorem 6.1.1 generalizes Lemma 3.2.3 on page 91. Indeed, if $b \equiv \beta \in (0,1)$, then $r(A) = \beta r(P) = \beta < 1$, and the result in Theorem 6.1.1 reduces to Lemma 3.2.3.

One way to prove Theorem 6.1.1 is via the law of iterated expectations: With $B_t := b(X_{t-1}, X_t)$ for all $t \geqslant 1$ and $\mathbb{E}_{X_1}$ denoting expectation conditioned on X_1, we have

$$\begin{aligned}\mathbb{E}_x \sum_{t=1}^{\infty} \left(\prod_{i=1}^{t} B_i \right) h(X_t) &= \mathbb{E}_x \, \mathbb{E}_{X_1} \sum_{t=1}^{\infty} \left(\prod_{i=1}^{t} B_i \right) h(X_t) \\ &= \mathbb{E}_x \, B_1 \mathbb{E}_{X_1} \sum_{t=1}^{\infty} \left(\prod_{i=2}^{t} B_i \right) h(X_t).\end{aligned}$$

By the definition of v in (6.1), the right-hand side is just

$$\mathbb{E}_x \, B_1 v(X_1) = \mathbb{E}_x \, b(X_0, X_1) v(X_1) = \sum_{x'} b(x, x') v(x') P(x, x') = \sum_{x'} v(x') A(x, x').$$

As a consequence, applying (6.1) again,

$$v(x) = h(x) + \mathbb{E}_x \sum_{t=1}^{\infty} \left(\prod_{i=1}^{t} B_i \right) h(X_t) = h(x) + \sum_{x'} v(x') A(x, x').$$

In vector form, the last equation becomes $v = h + Av$. Using $r(A) < 1$ and applying the Neumann series lemma gives nonsingularity of $I - A$ and the representation $v = (I - A)^{-1} h$.

The proof above uses a somewhat sophisticated version of the law of iterated expectations, which we have not proved in this text. To keep this book relatively self-contained, we provide an alternative proof of Theorem 6.1.1 on page 293, which uses a longer but more straightforward approach.

6.1.1.3 Application: Firm Valuation with Stochastic Interest Rates

Now we return to a firm valuation problem and allow the interest rate to follow a stochastic process $(r_t)_{t \geqslant 0}$. Under the convention that the interest rate over the period between t and $t+1$ is known at time t and written as r_t, the time zero expected present value of time t profit π_t is

$$\mathbb{E} \left\{ \beta_0 \cdots \beta_{t-1} \cdot \pi_t \right\} \qquad \text{where } \beta_t := \frac{1}{1 + r_t}.$$

The expected present value of the firm is

$$V_0 = \mathbb{E} \sum_{t=0}^{\infty} \left(\prod_{i=1}^{t} \beta_{i-1} \right) \pi_t \qquad \text{where } \prod_{i=1}^{0} \beta_{i-1} := 1. \tag{6.3}$$

Suppose that $\beta_t = \beta(X_t) := 1/(1+r(X_t))$ for some $r \in \mathbb{R}^{\mathsf{X}}$, so that randomness in interest rates is a function of the same Markov state that influences profits.[1] Conditioning on $X_0 = x$, the value in (6.3) becomes

$$v(x) = \mathbb{E}_x \sum_{t=0}^{\infty} \left\{ \prod_{i=1}^{t} \beta(X_{i-1}) \right\} \pi(X_t). \tag{6.4}$$

With $A(x, x') := \beta(x)P(x, x')$, we have

Proposition 6.1.2. *If $r(A) < 1$, then the firm valuation (6.4) is finite for all $x \in \mathsf{X}$ and*

$$v = (I - A)^{-1}\pi.$$

Exercise 6.1.1. Verify Proposition 6.1.2 via Theorem 6.1.1.

Exercise 6.1.2. Let X be partially ordered and assume $r(A) < 1$. Prove that v is in $i\mathbb{R}^{\mathsf{X}}$ whenever P is monotone increasing and $\beta, \pi \in i\mathbb{R}^{\mathsf{X}}$.

6.1.2 Necessary Conditions

In §6.1.1 we studied settings where lifetime value is a function v on a state space X that satisfies an equation of the form $v = h + Av$. The unknown is $v \in \mathbb{R}^{\mathsf{X}}$ where X is a finite set, $h \in \mathbb{R}^{\mathsf{X}}$ is given and A is a linear operator from $\mathbb{R}^{\mathsf{X}}$ to itself. We discussed the fact that $r(A) < 1$ is sufficient for $v = h + Av$ to have a unique solution.

In some settings the condition $r(A)$ is also necessary. For example, let

- V be all $v \in \mathbb{R}^{\mathsf{X}}$ such that $v \gg 0$,
- h be an element of V, and
- A be a positive linear operator on $\mathbb{R}^{\mathsf{X}}$.

[1] There is very little loss of generality in making this assumption. In fact, the two processes can still be statistically independent. For example, if we take X_t to have the form $X_t = (Y_t, Z_t)$, where (Y_t) and (Z_t) are independent Markov chains, then we can take β_t to be a function of Y_t and π_t to be a function of Z_t. The resulting interest and profit processes are statistically independent.

In this setting we have the following result:

Lemma 6.1.3. *If A is irreducible, then the next two statements are equivalent.*

(i) $r(A) < 1$.

(ii) *The equation $v = h + Av$ has a unique solution in V.*

Proof. Regarding (i) $\implies$ (ii), existence of a unique $v \in \mathbb{R}^{\mathsf{X}}$ satisfying $v = h + Av$ follows from the Neumann series lemma. Since $v = \sum_{t \geqslant 0} A^t h \geqslant h \gg 0$, we have $v \in V$.

For (ii) $\implies$ (i), let v be any solution to $v = h + Av$ in V. By the Perron–Frobenius theorem (page 62) and irreducibility of A, we can select a left eigenvector e such that $e \gg 0$ and $e^\top A = r(A)e^\top$. For this e, we have $e^\top v = e^\top Av + e^\top h = r(A)e^\top v + e^\top h$. Since $e, h \gg 0$, it must be that $e^\top h > 0$ and $e^\top v > 0$. Therefore $r(A)$ satisfies $(1 - r(A))\alpha = \beta$ for $\alpha, \beta > 0$. Hence $r(A) < 1$. □

In §7.1.3 we will extend lemma 6.1.3 to handle certain nonlinear equations.

6.2 Asset Pricing

In this section we provide a brief introduction to a theory of asset pricing in a Markov environment. While the topic of asset pricing is fascinating in its own right, our main aim is to provide additional practice in handling linear valuation problems. (Readers who lack interest in asset pricing and wish to push ahead with their study of dynamic programming can skip to §6.3.)

6.2.1 Introduction to Asset Pricing

We first discuss risk-neutral pricing and show why this assumption is typically implausible. Next, we introduce stochastic discount factors and stationary asset pricing.

6.2.1.1 Risk-Neutral Pricing?

Consider the problem of assigning a current price Π_t to an asset that confers on its owner the right to payoff G_{t+1}. The payoff is stochastic and realized next period. One simple idea is to use the **risk-neutral pricing**, which implies that

$$\Pi_t = \mathbb{E}_t \, \beta \, G_{t+1} \tag{6.5}$$

for some constant discount factor $\beta \in (0,1)$. If the payoff is in k periods, then we modify the price to $\mathbb{E}_t \beta^k G_{t+k}$. In essence, risk-neutral pricing says that cost equals expected reward, discounted to present value by compounding a constant rate of discount. (A *rate* of discount, say ρ, is linked to a discount *factor*, say β, by $\beta = \frac{1}{1+\rho} \approx \exp(\rho)$.)

Example 6.2.1. Let S_t be the price of a stock at each point in time t. A **European call option** gives its owner the right to purchase the stock at price K at time $t+k$. There is no obligation to exercise the option, so the payoff at $t+k$ is $\max\{S_{t+k} - K, 0\}$. Under risk-neutral pricing, the time t price of this option is

$$\Pi_t = \mathbb{E}_t \beta^k \max\{S_{t+k} - K, 0\}.$$

Although risk neutrality allows for simple pricing, assuming risk neutrality for all investors is *not* plausible.

To give one example, suppose that we take the asset that pays G_{t+1} in (6.5) and replace it with another asset that pays $H_{t+1} = G_{t+1} + \varepsilon_{t+1}$, where ε_{t+1} is independent of G_{t+1}, $\mathbb{E}_t \varepsilon_{t+1} = 0$ and $\operatorname{Var} \varepsilon_{t+1} > 0$. In effect, we are adding risk to the original payoff without changing its mean.

Under risk neutrality, the price of this new asset is

$$\Pi_t^H = \mathbb{E}_t \beta \left[G_{t+1} + \varepsilon_{t+1}\right] = \Pi_t + \beta \, \mathbb{E}_t \varepsilon_{t+1} = \Pi_t.$$

Thus, H_{t+1} and G_{t+1} are priced identically, even though their means are both $\mathbb{E}_t G_{t+1}$ and their variances satisfy

$$\operatorname{Var} H_{t+1} = \operatorname{Var} G_{t+1} + \operatorname{Var} \varepsilon_{t+1} > \operatorname{Var} G_{t+1}.$$

This outcome contradicts the idea that investors typically want to be compensated for bearing risk.

A helpful way to think about the same point is to consider the rate of return $r_{t+1} := (G_{t+1} - \Pi_t)/\Pi_t$ on holding an asset with payoff G_{t+1}. From (6.5) we have $\mathbb{E}_t \beta(1+r_{t+1}) = 1$, or

$$\mathbb{E}_t r_{t+1} = \frac{1-\beta}{\beta}.$$

Since the right-hand side does not depend on G_{t+1}, risk neutrality implies that all assets have the same expected rate of return. But this contradicts the finding that, on average, riskier assets tend to have higher rates of return that compensate investors for bearing risk.

Example 6.2.2. The **risk premium** on a given asset is defined as the expected rate of return minus the rate of return on a risk-free asset. If we assume risk neutrality then, by the preceding discussion, the risk premium is zero for all assets. However, calculations based on post-war US data show that the average return premium on equities over safe assets is around 8% per annum (see, e.g., Cochrane (2009)).

6.2.1.2 A Stochastic Discount Factor

To go beyond risk neutral-pricing, let's start with a model containing one asset and one agent. It is straightforward to price the asset and compare it to the risk neutral case.

A representative agent takes the price Π_t of a risky asset as given and solves

$$\max_{0 \leqslant \alpha \leqslant 1} \{u(C_t) + \beta \mathbb{E}_t u(C_{t+1})\}$$
$$\text{subject to} \quad C_t = E_t - \Pi_t \alpha \quad \text{and} \quad C_{t+1} = E_{t+1} + \alpha G_{t+1}.$$

Here

- u is a flow utility function,
- G_{t+1} is the payoff of the asset and Π_t is the time-t price,
- β is a constant discount factor measuring impatience of the agent,
- E_t and E_{t+1} are endowments and
- α is the share of the asset purchased by the agent.

Rewriting as $\max_\alpha \{u(E_t - \Pi_t \alpha) + \beta \mathbb{E}_t u(E_{t+1} + \alpha G_{t+1})\}$ and differentiating with respect to α leads to the first order condition

$$u'(E_t - \Pi_t \alpha)\Pi_t = \beta \mathbb{E}_t u'(E_{t+1} + \alpha G_{t+1}) G_{t+1}.$$

Rearranging gives us

$$\Pi_t = \mathbb{E}_t \left[\beta \frac{u'(C_{t+1})}{u'(C_t)} G_{t+1} \right]. \tag{6.6}$$

Comparing (6.6) with (6.5), we see that the payoff is now multiplied by a positive random variable rather than a constant. The random variable

$$M_{t+1} := \beta \frac{u'(C_{t+1})}{u'(C_t)} \tag{6.7}$$

is called the **stochastic discount factor** or **pricing kernel**. We call this particular form of the pricing kernel shown in (6.7) the **Lucas stochastic discount factor** (Lucas SDF) in honor of Lucas (1978a).

Example 6.2.3. If u is linear, so that $u(c) = ac + b$ for some $a, b \in \mathbb{R}$, then $u'(c) = a$ for all c, so $M_{t+1} = \beta$. If the utility function has no curvature, then pricing is risk neutral.

Example 6.2.4. If utility has the CRRA form $u(c) = c^{1-\gamma}/(1-\gamma)$ for some $\gamma > 0$, then the Lucas SDF takes the form

$$M_{t+1} = \beta \left(\frac{C_{t+1}}{C_t} \right)^{-\gamma}, \tag{6.8}$$

which we can also write as $M_{t+1} = \beta \exp(-\gamma g_{t+1})$ when $g_{t+1} := \ln(C_{t+1}/C_t)$ is the growth rate of consumption. This SDF is a positive random variable taking relatively small values in states of the world where consumption growth is high.

In the CRRA case, the Lucas SDF applies heavier discounting to assets that concentrate payoffs in states of the world where the agent is already enjoying strong consumption growth. Conversely, the SDF attaches higher weights to future payoffs that occur when consumption growth is low because such payoffs hedge against the risk of drawing low consumption states.

6.2.1.3 A General Specification

The standard neoclassical theory of asset pricing generalizes the Lucas discounting specification by assuming only that there exists a positive random variable M_{t+1} such that the price of an asset with payoff G_{t+1} is

$$\Pi_t = \mathbb{E}_t \, M_{t+1} \, G_{t+1} \qquad (t \geqslant 0). \tag{6.9}$$

As above, M_{t+1} is called a **stochastic discount factor** (SDF). Equation 6.9 generalizes (6.6) by refraining from restricting the SDF (apart from assuming positivity).

Actually, it can be shown that there exists an SDF M_{t+1} such that (6.9) is always valid under relatively weak assumptions. In particular, a single SDF M_{t+1} can be used to price *any* asset in the market, so if H_{t+1} is a another stochastic payoff then the current price of an asset with this payoff is $\mathbb{E}_t \, M_{t+1} \, H_{t+1}$.

We skip a proof of these claims, since our main interest is in understanding forward-looking equations in Markov environments that are needed for our discussion of dynamic programming below. References for asset pricing theory with full proofs are listed in §3.4.

6.2.1.4 Markov Pricing

A common assumption in quantitative applications is that all underlying randomness is driven by a Markov model. In this spirit, we take (X_t) to be P-Markov on finite state X, where $P \in \mathbb{M}(\mathsf{X})$, and suppose further that the SDF and payoff have the forms

$$M_{t+1} = m(X_t, X_{t+1}) \quad \text{and} \quad G_{t+1} = g(X_t, X_{t+1})$$

for fixed functions m, g mapping $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}_+$. Since m is arbitrary at this point, we don't assume a particular specification for the SDF.

In this setting, conditioning on $X_t = x$, the standard asset pricing equation $\Pi_t = \mathbb{E}_t M_{t+1} G_{t+1}$ becomes

$$\pi(x) = \sum_{x' \in \mathsf{X}} m(x, x') g(x, x') P(x, x') \qquad (x \in \mathsf{X}), \tag{6.10}$$

where $\pi(x)$ is the price of the asset conditional on $X_t = x$. (That is, $\Pi_t = \pi(X_t)$.)

6.2.1.5 Pricing a Stationary Dividend Stream

Now we are ready to look at pricing a stationary cash flow over an infinite horizon, a basic problem in asset pricing. We will apply the Markov structure assumed in §6.2.1.4. In all that follows, (X_t) is P-Markov on X.

We seek the time t price, denoted by Π_t, for an **ex-dividend contract** on the dividend stream $(D_t)_{t \geqslant 0}$. The contract provides the owner with the right to the dividend stream. The "ex-dividend" component means that, should the dividend stream be traded at time t, the dividend paid at time t goes to the seller rather than the buyer. As a result, purchasing at t and selling at $t + 1$ pays $\Pi_{t+1} + D_{t+1}$. Hence, applying the fundamental asset pricing equation, at time t price Π_t of the contract must satisfy

$$\Pi_t = \mathbb{E}_t M_{t+1} (\Pi_{t+1} + D_{t+1}). \tag{6.11}$$

We assume the existence of a $d \in \mathbb{R}_+^{\mathsf{X}}$ such that $D_t = d(X_t)$ for all t. Using (6.10), we can write this as

$$\pi(x) = \sum_{x'} m(x, x')(\pi(x') + d(x')) P(x, x') \qquad (x \in \mathsf{X}), \tag{6.12}$$

or, equivalently,

$$\pi = A\pi + Ad \quad \text{when } A(x, x') := m(x, x') P(x, x'). \tag{6.13}$$

By the Neumann series lemma, when $r(A) < 1$, the system (6.13) has the unique solution

$$\pi^* := (I - A)^{-1} A d = \sum_{k=1}^{\infty} A^k d. \tag{6.14}$$

The vector π^* is called an **equilibrium price function**. In view of Theorem 7.1.4, the condition $r(A) < 1$ is necessary (as well as sufficient) for existence of a positive solution π^* when A is irreducible and $d \gg 0$.

Exercise 6.2.1. As discussed in §6.2.1.1, the case $m \equiv \beta$ for some $\beta \in \mathbb{R}_+$ is called the risk-neutral case. Provide a condition on β under which $r(A) < 1$.

Exercise 6.2.2. Confirm that $(\Pi_t)_{t \geqslant 0}$ generated by $\Pi_t = \pi^*(X_t)$ solves (6.11).

Remark 6.2.1. We can call A an **Arrow–Debreu discount operator**. Its powers apply discounting: the valuation of any random payoff g in k periods is $A^k g$.

Exercise 6.2.3. Derive the price for a **cum-dividend contract** on the dividend stream $(D_t)_{t \geqslant 0}$, with the model otherwise unchanged. Under this contract, should the right to the dividend stream be traded at time t, the dividend paid at time t goes to the buyer rather than the seller.

6.2.1.6 Forward Sum Representation

Asset prices can be expressed as infinite sums under the assumptions stated above. Let's show this for cum-dividend contracts (although the case of ex-dividend contracts is similar). In Exercise 6.2.3 you found that the state-contingent price vector π for a cum-dividend contract on the dividend stream $(D_t)_{t \geqslant 0}$ obeys

$$\pi = d + A\pi \quad \text{when } A(x, x') := m(x, x') P(x, x') \tag{6.16}$$

and $r(A) < 1$. As before, $D_t = d(X_t)$ and $(X_t)_{t \geqslant 0}$ is P-Markov on X. Applying the uniqueness component of the Neumann series lemma (page 17) and Theorem 6.1.1, we see that the function π also obeys

$$\pi(x) = \mathbb{E}_x \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=1}^{t} M_i \right] D_t \right\}$$

where $M_{t+1} := m(X_t, X_{t+1})$ and $\prod_{i=1}^{0} M_i := 1$. This expression agrees with our intuition: The price of the contract is the expected present value of the dividend stream, with the time t dividend discounted by the composite factor $M_1 \cdots M_t$.

6.2.2 Nonstationary Dividends

Until now, our discussion of asset pricing has assumed that dividends are stationary. However, dividends typically grow over time, along with other economic measures such as GDP. In this section we solve for the price of a dividend stream when dividends exhibit random growth.

6.2.2.1 Price-Dividend Ratios

A standard model of dividend growth is

$$\ln \frac{D_{t+1}}{D_t} = \kappa(X_t, \eta_{t+1}) \qquad t = 0, 1, \ldots,$$

where κ is a fixed function, (X_t) is the state process and (η_t) is IID. We let φ be the density of each η_t and assume that (X_t) is P-Markov on a finite set X. Let's suppose as before that the SDF obeys $M_{t+1} = m(X_t, X_{t+1})$ for some positive function m.

Since dividends grow over time, so will the price of the asset. As such, we should no longer seek a fixed function π such that $\Pi_t = \pi(X_t)$ for all t, since the resulting price process (Π_t) will fail to grow. Instead, we try to solve for the **price-dividend ratio** $V_t := \Pi_t / D_t$, which we hope will be stationary.

Exercise 6.2.4. Using $\Pi_t = \mathbb{E}_t \left[M_{t+1}(D_{t+1} + \Pi_{t+1}) \right]$, show that

$$V_t = \mathbb{E}_t \left[M_{t+1} \exp(\kappa(X_t, \eta_{t+1})) \, (1 + V_{t+1}) \right]. \tag{6.17}$$

After conditioning on $X_t = x$, (6.17) leads us to conjecture existence of a function v such that

$$v(x) = \sum_{x' \in \mathsf{X}} m(x, x') \int \exp(\kappa(x, \eta)) \varphi(\mathrm{d}\eta) \, [1 + v(x')] \, P(x, x') \tag{6.18}$$

for all $x \in \mathsf{X}$. We understand (6.18) as an equation to be solved for the unknown object $v \in \mathbb{R}^{\mathsf{X}}$. If we can find a solution v^* to (6.18), then setting $V_t = v^*(X_t)$ yields a process (V_t) that obeys (6.17).

Exercise 6.2.5. Let

$$A(x, x') := m(x, x') \int \exp(\kappa(x, \eta))\varphi(\mathrm{d}\eta)P(x, x') \qquad (x, x' \in \mathsf{X}). \tag{6.19}$$

Show that (6.17) has a unique solution v^* in $\mathbb{R}^{\mathsf{X}}$ when $r(A) < 1$, and

$$v^* = (I - A)^{-1}A\mathbb{1} = \sum_{t \geqslant 1} A^t \mathbb{1}. \tag{6.20}$$

The price-dividend process (V_t^*) defined by $V_t^* = v^*(X_t)$ solves (6.17). The price can be recovered via $\Pi_t = V_t^* D_t$.

6.2.2.2 Application: Markov Growth with a Lucas SDF

As an example, suppose that dividend growth obeys

$$\kappa(X_t, \eta_{d,t+1}) = \mu_d + X_t + \sigma_d \, \eta_{d,t+1}$$

where $(\eta_{d,t})_{t \geqslant 0}$ is IID and standard normal. Consumption growth is given by

$$\ln \frac{C_{t+1}}{C_t} = \mu_c + X_t + \sigma_c \, \eta_{c,t+1},$$

where $(\eta_{c,t})_{t \geqslant 0}$ is also IID and standard normal. We use the Lucas SDF in (6.8), implying that

$$M_{t+1} = \beta \left(\frac{C_{t+1}}{C_t} \right)^{-\gamma} = \beta \exp(-\gamma(\mu_c + X_t + \sigma_c \eta_{c,t+1}))$$

Exercise 6.2.6. Using (6.19), show that

$$A(x, x') = \beta \exp\left(-\gamma \mu_c + \mu_d + (1 - \gamma)x + \frac{\gamma^2 \sigma_c^2 + \sigma_d^2}{2} \right) P(x, x').$$

Figure 6.3 shows the price-dividend ratio function v^* for the specification given in Listing 21, as well as for an alternative mean dividend growth rate μ_d. The state process is a Tauchen discretization of an AR(1) process with positive autocorrelation. An increase in the state predicts higher dividends, which tends to increase the price.

```
using QuantEcon, LinearAlgebra

"Creates an instance of the asset pricing model with Markov state."
function create_asset_pricing_model(;
        n=200,                # state grid size
        ρ=0.9, ν=0.2,         # state persistence and volatility
        β=0.99, γ=2.5,        # discount and preference parameter
        μ_c=0.01, σ_c=0.02, # consumption growth mean and volatility
        μ_d=0.02, σ_d=0.1)  # dividend growth mean and volatility
    mc = tauchen(n, ρ, ν)
    x_vals, P = exp.(mc.state_values), mc.p
    return (; x_vals, P, β, γ, μ_c, σ_c, μ_d, σ_d)
end
```

Listing 21: Asset pricing model with Lucas SDF (`pd_ratio.jl`)

At the same time, higher x also predicts higher consumption growth, which acts negatively on the price. For values of γ greater than 1, the second effect dominates and the price-dividend ratio slopes down.

Exercise 6.2.7. Complete the code in Listing 21 and replicate Figure 6.3. Add a test to your code that checks $r(A) < 1$ before computing the price-dividend ratio.

6.2.3 Incomplete Markets

In §6.2.1.5, the problem of solving for the equilibrium price vector π was analyzed using the Neumann series lemma. However, some modifications to the basic model bring nonlinearities that make use of the Neumann series lemma impossible. For example, Harrison and Kreps (1978) analyze a setting with heterogeneous beliefs and incomplete markets, leading to failure of the standard asset pricing equation. This results in a nonlinear equation for prices.

We treat the Harrison and Kreps model only briefly. There are two types of agents. Type i believes that the state updates according to stochastic matrix P_i for $i = 1, 2$. Agents are risk-neutral, so $m(x, y) \equiv \beta \in (0, 1)$. Harrison and Kreps (1978) show that, for their model, the equilibrium condition (6.12) becomes

$$\pi(x) = \max_i \beta \sum_{x'} [\pi(x') + d(x')] P_i(x, x') \tag{6.21}$$

Figure 6.3: Price-dividend ratio as a function of the state

for $x \in \mathsf{X}$ and $i \in \{1, 2\}$. Setting aside the details that lead to this equation, our objective is simply to obtain a vector of prices π that solves (6.21).

As a first step, we introduce an operator $T \colon \mathbb{R}_+^{\mathsf{X}} \to \mathbb{R}_+^{\mathsf{X}}$ that maps π to $T\pi$ via

$$(T\pi)(x) = \max_i \beta \sum_{x'} [\pi(x') + d(x')] P_i(x, x') \qquad (x \in \mathsf{X}). \tag{6.22}$$

We are assuming $d \geqslant 0$, so T is indeed a self-map on $\mathbb{R}_+^{\mathsf{X}}$.

By construction, a vector $\pi \in \mathbb{R}_+^{\mathsf{X}}$ is a fixed point of T if and only if it is a vector of prices that solves (6.21). Hence, we have successfully converted our equilibrium problem into a fixed point problem.

We aim to show that T is a contraction. To this end, pick any $p, q \in \mathbb{R}_+^{\mathsf{X}}$. Applying the inequality from Lemma 1.2.5 on page 31, we obtain

$$|(Tp)(x) - (Tq)(x)| \leqslant \beta \max_i \left| \sum_{x'} [p(x') + d(x')] P_i(x, x') - \sum_{x'} [q(x') - d(x')] P_i(x, x') \right|.$$

Using the triangle inequality and canceling terms leads to

$$|(Tp)(x) - (Tq)(x)| \leqslant \beta \max_{i \in \{1,2\}} \sum_{x'} |p(x') - q(x')| P_i(x, x') \leqslant \beta \|p - q\|_\infty.$$

Since this bound holds for all x, we can take the maximum with respect to x and obtain

$$\|Tp - Tq\|_\infty \leqslant \beta \|p - q\|_\infty.$$

Thus, on $\mathbb{R}^{\mathsf{X}}_+$, the map T is a contraction of modulus β with respect to the sup norm.

Since $\mathbb{R}^{\mathsf{X}}_+$ is a closed subset of $\mathbb{R}^{\mathsf{X}}$, we conclude that T has a unique fixed point in this set. Hence, the system (6.21) has a unique solution π^* in $\mathbb{R}^{\mathsf{X}}_+$, representing equilibrium prices. This fixed point can be computed by successive approximation.

Exercise 6.2.8. Provide an alternative proof of contractivity of T on $\mathbb{R}^{\mathsf{X}}_+$ using Blackwell's condition (§2.1.2.2).

6.3 Optimality with State-Dependent Discounting

We can now turn to problems where the objective is to maximize lifetime value in the presence of state-dependent discounting. After reviewing necessary fixed point theory, we present an extension of the MDP model from Chapter 5 that admits state-dependent discounting. We provide weak conditions under which optimal policies exist and Bellman's principle of optimality holds.

6.3.1 Eventual Contractions

State-dependent discounting breaks the contractivity properties we exploited in Chapter 5, when we studied optimality of MDPs (see, e.g., the proof of Proposition 5.1.1). Hence our first step is to introduce generalization of Banach's fixed point theorem that can establish global stability under weaker conditions.

6.3.1.1 A Stability Result

Fix $U \subset \mathbb{R}^{\mathsf{X}}$. We call a self-map T on U **eventually contracting** if there exists a $k \in \mathbb{N}$ and a norm $\|\cdot\|$ on $\mathbb{R}^{\mathsf{X}}$ such that T^k is a contraction on U under $\|\cdot\|$.

Theorem 6.3.1. *Let U be a closed subset of $\mathbb{R}^{\mathsf{X}}$ and let T be a self-map on U. If T is eventually contracting on U, then T is globally stable on U.*

Exercise 6.3.1. Prove Theorem 6.3.1. [Hint: Theorem 1.2.3 is self-improving, in the sense that it implies this seemingly stronger result.]

It is helpful to recognize the connection between Theorem 6.3.1 and the Neumann series lemma on page 17. If $U = \mathbb{R}^n$ and $Tu = Au + b$ with $r(A) < 1$, then

$$\|T^k u - T^k v\|_\infty = \|A^k u - A^k v\|_\infty = \|A^k(u - v)\|_\infty \leqslant \|A^k\|_\infty \|u - v\|_\infty.$$

Since $r(A) < 1$, we can choose a $k \in \mathbb{N}$ such that $\|A^k\|_\infty < 1$ (see Exercise 1.2.9). Hence T is eventually contracting and Theorem 6.3.1 yields global stability. We do not need to call on the Neumann series lemma.

On one hand, eventual contractions have much wider scope than the Neumann series lemma, since they can also be applied in nonlinear settings. On the other hand, the Neumann series lemma is preferred when applicable, since it provides inverse and power series representations of the fixed point.

On one hand, if T is a contraction map on $U \subset \mathbb{R}^{\mathsf{X}}$ with respect to a given norm $\|\cdot\|_a$, we cannot necessarily say that T is a contraction with respect to some other norm $\|\cdot\|_b$ on $\mathbb{R}^{\mathsf{X}}$. On the other hand, if T is an eventual contraction on U with respect to some given norm on $\mathbb{R}^{\mathsf{X}}$, then T is eventually contracting with respect to every norm on $\mathbb{R}^{\mathsf{X}}$. The next exercise asks you to verify this.

Exercise 6.3.2. Let $\|\cdot\|_a$ and $\|\cdot\|_b$ be norms on $\mathbb{R}^{\mathsf{X}}$ and let T be a self-map on $U \subset \mathbb{R}^{\mathsf{X}}$ such that T^k is a contraction on U with respect to $\|\cdot\|_a$ for some $k \in \mathbb{N}$. Prove that there exists an $\ell \in \mathbb{N}$ such that T^ℓ is a contraction on U with respect to $\|\cdot\|_b$.

6.3.1.2 A Spectral Radius Condition

The following sufficient condition for eventual contractivity will be helpful when we study dynamic programs with state-dependent discounting.

Proposition 6.3.2. *Let T be a self-map on $U \subset \mathbb{R}^{\mathsf{X}}$. If there exists a positive linear operator L on $\mathbb{R}^{\mathsf{X}}$ such that $r(L) < 1$ and*

$$|Tv - Tw| \leqslant L|v - w| \tag{6.23}$$

for all $v, w \in U$, then T is an eventual contraction on U.

Proof. Fix $v, w \in U$. Pick any $k \in \mathbb{N}$. We have $|T^k v - T^k w| \leqslant L|T^{k-1}v - T^{k-1}w|$, or

$$e_k \leqslant L e_{k-1} \quad \text{where} \quad e_k := |T^k v - T^k w|. \tag{6.24}$$

Since L is a positive linear operator, L is order-preserving on U by Exercise refex:plop. As a result, we can iterate on (6.24) to obtain $e_k \leqslant L^k e_0$, or

$$|T^k v - T^k w| \leqslant L^k |v - w| \leqslant L^k \mathbb{1} \|v - w\|_\infty.$$

Taking the maximum on the right and then the left,

$$\|T^k v - T^k w\|_\infty \leqslant \|L^k \mathbb{1}\|_\infty \|v - w\|_\infty \leqslant n \|L^k\|_\infty \|v - w\|_\infty,$$

where $\|L^k\|_\infty$ is defined via the matrix norm in §1.2.1.4 and $n = |X|$. Since $r(L) < 1$, we have $\|L^k\|_\infty \to 0$ as $k \to \infty$ (see Exercise 1.2.9 on page 18). Hence T is eventually contracting on U. □

6.3.1.3 A Blackwell-Type Condition

In §2.1.2.2 we studied a sufficient condition for order-preserving self maps to be contractions. The next proposition provides an analogous result for eventual contractions. In the statement of the proposition, U is a subset of $\mathbb{R}^X$ such that $v, c \in U$ and $c \geqslant 0$ implies $v + c \in U$.

Proposition 6.3.3. *Let T be an order-preserving self-map on U. If there exists a positive linear operator L on $\mathbb{R}^X$ such that $r(L) < 1$ and*

$$T(v + c) \leqslant Tv + Lc \quad \textit{for all } c, v \in \mathbb{R}^X \textit{ with } c \geqslant 0,$$

then T is eventually contracting on U.

Proof. Fix $v, w \in U$ and let T and L be as in the statement of the proposition. By the assumed properties on T, we have

$$Tv = T(v + w - w) \leqslant T(w + |v - w|) \leqslant Tw + L|v - w|$$

Rearranging gives $Tv - Tw \leqslant L|v - w|$. Reversing the roles of v and w yields $|Tv - Tw| \leqslant L|v - w|$. The claim in Proposition 6.3.3 now follows from Proposition 6.3.2. □

6.3.2 MDPs with State-Dependent Discounting

We are now ready to extend the MDP model to include state-dependent discounting. We construct a framework and then provide weak conditions for optimality. These conditions rely on eventual contractivity.

6.3.2.1 A General Version

Our first step is to provide a framework for MDPs with state-dependent discounting. We begin with an MDP (Γ, β, r, P) with state space X, action space A and feasible state-action pairs G. We then replace the constant discount factor β with a function β from $\mathsf{G} \times \mathsf{X}$ to $\mathbb{R}_+$, so that the Bellman equation takes the form

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a) + \sum_{x'} v(x')\beta(x,a,x')P(x,a,x') \right\} \tag{6.25}$$

where $x \in \mathsf{X}$ and $v \in \mathbb{R}^{\mathsf{X}}$. Notice that we allow the discount factor to depend on all relevant information: the current action, the current state and the stochastically determined next period state. We call this model an **MDP with state-dependent discounting**.

Let Σ be the set of all feasible policies, defined as for regular MDPs. The policy operator T_σ corresponding to $\sigma \in \Sigma$ is represented by

$$(T_\sigma v)(x) = \left\{ r(x,\sigma(x)) + \sum_{x'} v(x')\beta(x,\sigma(x),x')P(x,\sigma(x),x') \right\}. \tag{6.26}$$

Given $v \in \mathbb{R}^{\mathsf{X}}$, a policy σ is called v-greedy if $\sigma(x)$ is a maximizer of the right-hand side of (6.25) for all x in X. Equivalently, σ is v-greedy whenever $T_\sigma v = Tv$.

When T_σ has a unique fixed point v_σ for each $\sigma \in \Sigma$, we let v^* denote the value function, which is defined as $v^* := \vee_{\sigma \in \Sigma} v_\sigma$. As for the regular MDP case, a policy σ is called optimal if $v_\sigma = v^*$.

6.3.2.2 Optimality

For MDPs with state-dependent discounting, we can easily obtain standard optimality results by assuming a strict bound on discounting. For example if we assume that there exists a $b < 1$ such that $\beta(x,a,x') \leqslant b$ for all $(x,a,x') \in \mathsf{G} \times \mathsf{X}$, then the Bellman and policy operators are all contraction maps. In this setting it is easy to extend Proposition 5.1.1 on page 134, which provided optimality results for regular MDPs.

Unfortunately, the assumption discussed in the previous paragraph is too strict for many applications. We return to this point in §6.3.2.5. For now we state an optimality result under a significantly weaker condition.

Proposition 6.3.4. *If there exists an $L \in \mathbb{L}(\mathsf{X})$ such that $r(L) < 1$ and*

$$\beta(x, a, x')P(x, a, x') \leqslant L(x, x') \tag{6.27}$$

for all $(x, a) \in \mathsf{G}$ and $x' \in \mathsf{X}$, then

(i) *T_σ has a unique fixed point v_σ in $\mathbb{R}^{\mathsf{X}}$ for all $\sigma \in \Sigma$,*

(ii) *the value function v^* is the unique fixed point of the Bellman operator,*

(iii) *a policy $\sigma \in \Sigma$ is optimal if and only if it is v^*-greedy, and*

(iv) *at least one optimal policy exists.*

In §8.2.3 we prove a result that includes Proposition 6.3.4 as a special case.

Inequality 6.27 also implies that L is a positive linear operator.

Exercise 6.3.3. Prove that, under the conditions of Proposition 6.3.4, the Bellman operator T defined in (6.25) is globally stable on $\mathbb{R}^{\mathsf{X}}$.

Exercise 6.3.4. Fix $\sigma \in \Sigma$ and prove that, under the conditions of Proposition 6.3.4, the operator T_σ has a unique fixed point v_σ in $\mathbb{R}^{\mathsf{X}}$ and, moreover,

$$v_\sigma(x) = \mathbb{E}_x \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=1}^{t} \beta(X_{i-1}, \sigma(X_{i-1}), X_i) \right] r_\sigma(X_t) \right\} \qquad (x \in \mathsf{X}), \tag{6.28}$$

when (X_t) is P_σ-Markov with initial condition x and, by convention, $\prod_{i=1}^{0} := 1$. [Hint: Compare (6.28) with the expression for firm value given on page 182.]

The term $\prod_{i=1}^{t} \beta(X_{i-1}, \sigma(X_{i-1}), X_i)$ is the discount factor applied to current reward $r_\sigma(X_t)$. Equation (6.28) tells us that a lifetime reward equals an expected value of the sum of these discounted rewards.

6.3.2.3 Algorithms

Given an MDP with state-dependent discounting and $r(L) < 1$, we can compute the optimal policy by any one of

(i) value function iteration,

(ii) Howard policy iteration, or

(iii) optimistic policy iteration.

The algorithms are identical to those given for regular MDPs (see §5.1.4), provided that the correct operators T and T_σ are used, and that the definition of a v-greedy policy is set to (6.30). All of the algorithms are convergent. See Chapter 8 for details.

6.3.2.4 Exogenous Discounting

In applications it is common to specialize the framework above by introducing an exogenous state component that drives the discount factor process. In this section we set up a generic version of this model and discuss optimality.

The first step is to decompose the state X_t into a pair (Y_t, Z_t), where $(Y_t)_{t\geqslant 0}$ is endogenous (i.e., affected by the actions of the controller) and $(Z_t)_{t\geqslant 0}$ is purely exogenous. In particular, the primitives consist of

(i) a nonempty correspondence Γ from $\mathsf{Y} \times \mathsf{Z}$ to A,

(ii) a function β from Z to $\mathbb{R}_+$,

(iii) a function r from $\mathsf{G} := \{(y, a) \in \mathsf{Y} \times \mathsf{A} : a \in \Gamma(y)\}$ to $\mathbb{R}$,

(iv) a stochastic matrix Q on Z and

(v) a stochastic kernel R from G to Y.

The corresponding Bellman equation is

$$v(y, z) = \max_{a\in\Gamma(y,z)} \left\{ r(y, a) + \beta(z) \sum_{z'\in\mathsf{Z}} \sum_{y'\in\mathsf{Y}} v(y', z')Q(z, z')R(y, a, y') \right\} \tag{6.29}$$

for all $(y, z) \in \mathsf{X}$. Given $v \in \mathbb{R}^{\mathsf{X}}$, a policy $\sigma \in \Sigma$ is called **v-greedy** if

$$\sigma(y, z) \in \operatorname*{argmax}_{a\in\Gamma(y)} \left\{ r(y, a) + \beta(z) \sum_{z'\in\mathsf{Z}} \sum_{y'\in\mathsf{Y}} v(y', z')Q(z, z')R(y, a, y') \right\} \tag{6.30}$$

for all $(y, z) \in \mathsf{X}$.

This exogenous discount model is a special case of the general MDP with state-dependent discounting. Indeed, we can write (6.29) as (6.25) by setting $x := (y, z)$ and defining

$$P(x, a, x') := P((y, z), a, (y', z')) := Q(z, z')R(y, a, y').$$

The following proposition provides a relatively simple sufficient condition for the core optimality results in the setting of the exogenous discount model.

Proposition 6.3.5. *Let $L(z, z') := \beta(z)Q(z, z')$. If $r(L) < 1$, then all of the optimality results in Proposition 6.3.4 hold.*

Exercise 6.3.5. Prove Proposition 6.3.3.

6.3.2.5 Comments on the Spectral Radius Condition

In §6.3.2.2 we mentioned that the restriction $\sup \beta < 1$ is too strict for useful applications. As one illustration of this fact, observe that the real interest rate r_t shown in Figure 6.2 is sometimes negative. This means that, when discounting with real rates, the associated discount factor $\beta_t = 1/(1+r_t)$ is sometimes greater than 1. Hence strict discounting fails.

In addition, in the macroeconomic literature, empirically motivated time-varying discount factor specifications lead to models where $\beta_t > 1$ occurs with positive probability. For example, Hills et al. (2019) study a model that can be embedded in the MDP framework just described. Figure 6.4 shows a simulation of one of the discount factor processes used in their model, prior to discretization. The exogenous state and discount factor process takes the form $\beta_t = bZ_t$, where (Z_t) is an exogenous state obeying $Z_{t+1} = 1 - \rho + \rho Z_t + \sigma\varepsilon_{t+1}$ with (ε_t) IID and standard normal. Clearly $\sup \beta < 1$ fails for this model too.

Let's now consider the weaker condition $r(L) < 1$ described in Proposition 6.3.3 and check whether it holds. Following Hills et al. (2019), we discretize the dynamics of (Z_t) via a Tauchen approximation, producing a stochastic matrix Q on a finite set Z.[2] The set of values for β_t ranges between 0.95 and 1.04, so that $\beta_t > 1$ remains possible. Nonetheless, with $L(z, z') = \beta(z)Q(z, z')$ we obtain $r(L) = 0.9996$.

It follows from Proposition 6.3.3 that the fundamental optimality results apply for any state-dependent MDP using this exogenous discount factor process.

6.3.3 Inventory Management Revisited

In this section, we modify the inventory management model from §5.2.1 to include time-varying interest rates.

[2]The parameters are $\rho = 0.85$, $\sigma = 0.0062$, and $b = 0.99875$. In line with Hills et al. (2019), we discretize the model via `mc = tauchen(n, ρ, σ, 1 - ρ, m)` with $m = 4.5$ and $n = 15$.

Figure 6.4: Discount factor process $(\beta)_{t\geqslant 0}$ in Hills et al. (2019).

Recall that, in the model of §5.2.1, the Bellman equation takes the form

$$v(x) = \max_{a\in\Gamma(x)} \left\{ r(x,a) + \beta \sum_{d\geqslant 0} v(m(x-d)+a)\varphi(d) \right\} \tag{6.31}$$

at each $x \in \mathsf{X}$, where $\mathsf{X} := \{0,\dots,K\}$, x is the current inventory level, a is the current inventory order, $r(x,a)$ is current profits (defined in (5.8)), $m(y) := y \vee 0$ and d is an IID demand shock with distribution φ. Let's now add a time-varying discount rate and investigate its impact on optimal choices.

To add time-varying discounting we replace the constant β in (6.31) with a stochastic process (β_t) where $\beta_t = 1/(1+r_t)$. We suppose that the dynamics can be expressed as $\beta_t = \beta(Z_t)$, where the exogenous process $(Z_t)_{t\geqslant 0}$ is Q-Markov on Z. After relabeling the endogenous state X_t as Y_t and x as y, in line with the notation in §6.3.2.4, the Bellman equation becomes

$$B((y,z),a,v) = r(y,a) + \beta(z) \sum_{d\geqslant 0} \sum_{z'} v(m(y-d)+a,z')\varphi(d)Q(z,z').$$

If we set

$$R(y,a,y') := \mathbb{P}\{m(y-D)+a = y'\} \quad \text{when} \quad D \sim \varphi,$$

then $R(y,a,y')$ is the probability of realizing next period inventory level y' when the

Figure 6.5: Inventory dynamics with time-varying interest rates

current level is y and the action is a. Hence we can rewrite the aggregator as

$$B((y,z),a,v) = r(y,a) + \beta(z) \sum_{y',z'} v(y',z') R(y,a,y') Q(z,z').$$

Let $L(z,z') := \beta(z)Q(z,z')$. This is a version of the MDP with exogenous state-dependent discounting described in §6.3.2.4. By Proposition , all of the standard optimality results hold whenever $r(L) < 1$.

Figure 6.5 shows how inventory evolves under an optimal program when the parameters of the problem are as given in Listing 22. (The code includes a test for $r(L) < 1$.) We set $\beta(z) = z$ and take (Z_t) to be a discretization of an AR(1) process. Figure 6.5 was created by simulating (Z_t) according to Q and inventory (X_t) according to $X_{t+1} = m(X_t - D_{t+1}) + A_t$, where A_t follows the optimal policy.

The outcome is similar to Figure 5.7, in the sense that inventory falls slowly and then jumps up. As before, fixed costs induce this lumpy behavior. However, a new phenomenon is now present: inventories move up or down on average, persistently moving up as interest rates fall and down as interest rates rise. (The interest rate r_t is calculated via $\beta_t = 1/(1+r_t)$ at each t.) High interest rates foreshadow high interest rates due to positive autocorrelation ($\rho > 0$), which in turn devalue future profits and hence encourage managers to economize on stock.

```
using LinearAlgebra, Distributions, OffsetArrays, QuantEcon

function create_sdd_inventory_model(;
        ρ=0.98, ν=0.002, n_z=20, b=0.97,  # Z state parameters
        K=40, c=0.2, κ=0.8, p=0.6)        # firm and demand parameters
    ϕ(d) = (1 - p)^d * p                  # demand pdf
    mc = tauchen(n_z, ρ, ν)
    z_vals, Q = mc.state_values .+ b, mc.p
    rL = maximum(abs.(eigvals(z_vals .* Q)))
    @assert  rL < 1 "Error: r(L) ≥ 1."    # check r(L) < 1
    return (; K, c, κ, p, ϕ, z_vals, Q)
end
```

Listing 22: Investment model with time-varying discounting (`inventory_sdd.jl`)

6.4 Chapter Notes

Asset pricing is discussed in many sources, including Hansen and Renault (2010), Ross (2009), Cochrane (2009), Duffie (2010) and Campbell (2017). It provides a foundation and benchmark for many applications and extensions in macroeconomics, public finance, international economics, and other fields. Some of these are described in Ljungqvist and Sargent (2018).

Dynamic programming with state state-dependent discounting is now common in macroeconomics and finance. Representative examples include Krusell and Smith (1998), Woodford (2011), Christiano et al. (2014), Albuquerque et al. (2016), Saijo (2017), Basu and Bundick (2017), de Groot et al. (2018), Schorfheide et al. (2018), Hills et al. (2019), Toda (2019), Fagereng et al. (2019), Hubmer et al. (2020) and Cao (2020). For more on the theory of state-dependent discounting, see Jasso-Fuentes et al. (2020), Toda (2021) or Stachurski and Zhang (2021). An analysis of sovereign default with time-varying interest rates is provided by Bloise and Vailakis (2022).

Another challenge to the standard model with constant discount rates comes from empirical and experimental studies that find evidence of "hyperbolic discounting," where valuations across time fall rapidly at first and then more slowly. Thoughtful reviews of hyperbolic and quasi-hyperbolic discounting can be found in Frederick et al. (2002) and Rubinstein (2003). Cao and Werning (2018) provide conditions under which predictions from optimal savings models with quasi-hyperbolic discounting are robust. Balbus et al. (2018) analyze uniqueness of time-consistent stationary Markov policies for quasi-hyperbolic households under uncertainty. Balbus et al.

(2022) study equilibria in dynamic models with recursive payoffs and generalized discounting. Noor and Takeoka (2022) addresses the topic of optimal discounting. Additional references include Diamond and Köszegi (2003), Dasgupta and Maskin (2005), Karp (2005), Balbus et al. (2018), Hens and Schindler (2020), Jaśkiewicz and Nowak (2021), and Drugeon and Wigniolle (2021).

Chapter 7

Nonlinear Valuation

Lifetime value is the objective to be optimized in every dynamic program. One of the key issues in dynamic programming is how to combine a flow of rewards (or losses) into lifetime value. So far we have considered linear valuation based on summation over expected discounted rewards, using either constant discount rates (Chapters 1–5) or state-dependent discounting (Chapter 6).

In this chapter we drop the linearity assumption, so that lifetime value can no longer be written as a sum of discounted rewards. Instead, lifetime value is computed from a recursion that is typically nonlinear. This "recursive preference" approach is becoming increasingly popular in applied research, as detailed in §6.4.

Throughout this chapter we focus on valuation rather than dynamic programming *per se*. Later, in Chapter 8, we will show how to maximize lifetime value in settings where recursive preferences are adopted.

7.1 Beyond Contraction Maps

Typically, lifetime value in recursive preference environments is expressed as a fixed point of a nonlinear operator. One challenge of this field is that many recursive preference specifications induce operators that fail to be contractions (or even eventual contractions). For this reason, we now invest in some additional fixed point theory. All of this theory concerns order-preserving maps, since the operators in question inherit monotonicity from underlying preferences.

7.1.1 Order Intervals and Fixed Points

In this section we prove a simple but useful version of perhaps the most important order-theoretic fixed point theorem, typically attributed to Bronislaw Knaster (1893–1980) and Alfred Tarski (1901–1983). The theorem provides conditions for existence of fixed points for order-preserving maps as well as restrictions on the structure of the set of all fixed points.

To begin, set $P := [0, 1]$ and try to draw an increasing function T that maps P to itself and yet fails to touch the 45 degree line. You will find that it's impossible. The next theorem generalizes this idea:

Theorem 7.1.1 (Knaster–Tarski)**.** *Let X be a finite set. If $I = [\varphi, \psi]$ is a nonempty order interval in $(\mathbb{R}^{\mathsf{X}}, \leqslant)$ and T is an order-preserving self-map on $(I, \leqslant)$, then the set of fixed points of T is nonempty and contains least and greatest elements $u^* \leqslant u^{**}$. Moreover,*

$$T^k\varphi \leqslant u^* \leqslant u^{**} \leqslant T^k\psi \quad \textit{for all } k \geqslant 0. \tag{7.1}$$

Proof. Let U be all $u \in I$ such that $u \leqslant Tu$. Let u^{**} be defined by $u^{**}(x) := \sup_{u \in U} u(x)$. By Theorem A.2.1 on page 287, u^{**} is a well-defined element of I. (The set U is nonempty because $\varphi \in U$.) We claim that u^{**} is a fixed point, for which it suffices to obtain the two inequalities $u^{**} \leqslant Tu^{**}$ and $Tu^{**} \leqslant u^{**}$.

Regarding the first inequality, given any $u \in U$, we have $u \leqslant u^{**}$, and hence $Tu \leqslant Tu^{**}$. But $u \in U$ also implies $u \leqslant Tu$, so $u \leqslant Tu^{**}$. Since this bound holds for all $u \in U$, it follows from the definition of the supremum that $u^{**} \leqslant Tu^{**}$. This gives us one of our inequalities.

To complete the argument, we use the last inequality to obtain $Tu^{**} \leqslant TTu^{**}$, which implies that $Tu^{**} \in U$. Since u^{**} is a pointwise supremum over U, this yields $Tu^{**} \leqslant u^{**}$. We can now conclude that $Tu^{**} = u^*$.

Let D be all $u \in I$ such that $Tu \leqslant u$. Let u^* be defined by $u^*(x) := \inf_{u \in U} u(x)$. Using analogous arguments with inequalities reversed shows that u^* is also a fixed point.

Now let $\bar{u}$ be any fixed point of T. Since $\bar{u} \in U$, we have $\bar{u} \leqslant u^{**}$. Since $\bar{u} \in D$, we also have $u^* \leqslant \bar{u}$. In particular, for any fixed point $\bar{u}$ of T in I, we have $u^* \leqslant \bar{u} \leqslant u^{**}$. □

Unlike Banach's contraction mapping theorem, Theorem 7.1.1 only yields existence. Uniqueness and stability both can fail—as you can easily convince yourself by sketching the one-dimensional case $I = [0, 1]$. In the next section we seek additional conditions that guarantee uniqueness and stability.

Figure 7.1: Global stability induced by increasing concave functions

7.1.2 Concavity, Convexity and Stability

In this section we introduce a set of sufficient conditions for global stability that replace contractivity with shape properties on the operator such as concavity and convexity. These results have many applications in economics and finance.

7.1.2.1 The One-Dimensional Case

To build intuition, we start with the one-dimensional case, where the fixed-point problem can be visualized and proofs are relatively simple. We show how concavity and monotonicity can be combined to achieve stability.

In fact we saw how this pairing can produce global stability in §1.2.3.2, where we established stability for a discrete time Solow–Swan model under conditions on the parameters that imply concavity and monotonicity. However, our proof was quite specialized. Here is a more general result.

Proposition 7.1.2. *Let g be an increasing concave self-map on $U := (0, \infty)$. If, for all $x \in U$, there exists a pair $a, b \in U$ with $a \leqslant x \leqslant b$, $a < g(a)$ and $g(b) \leqslant b$, then g is globally stable on U.*

The proof is below. Figure 7.1 gives one example, where $g(x) = 1 + \sqrt{x}/2$. For such a function, given any positive number x, we can find a number $a < x$ that gets mapped strictly up (i.e., $g(a)$ is above the 45 line) and a point $b > x$ that gets mapped strictly

down (i.e., $g(b)$ is below the 45 degree line). Under iteration all trajectories converge to the unique fixed point x^*.

Before reading the proof we recommend that you sketch your own examples to anticipate roles of various conditions.

Exercise 7.1.1. Prove that the map g and set U defined in the discussion of the Solow-Swan model above Proposition 7.1.2 satisfies the conditions of the proposition.

Exercise 7.1.2. Dropping the Cobb-Douglas specification on production, suppose $g(k) = sf(k) + (1-\delta)k$ where $0 < s, \delta < 1$ and f is a strictly positive increasing concave production function on $U = (0, \infty)$ satisfying the **Inada conditions**

$$f'(k) \to \infty \text{ as } k \to 0 \quad \text{and} \quad f'(k) \to 0 \text{ as } k \to \infty,$$

Use Proposition 7.1.2 to prove that g is globally stable on U.

Exercise 7.1.3. Fajgelbaum et al. (2017) study a law of motion for aggregate uncertainty given by

$$s_{t+1} = g(s_t) \quad \text{where} \quad g(s) := \rho^2 \left[\frac{1}{s} + a^2 \frac{1}{\eta} \right]^{-1} + \gamma.$$

Let a, η and γ be positive constants and assume $0 < \rho < 1$. Prove that g is globally stable on $M := (0, \infty)$.

Proof of Proposition 7.1.2. First we prove existence of a fixed point $x^* \in U$. Fix $x \in U$. Suppose first that $x \leqslant g(x)$. Since g is increasing, we then have $g(x) \leqslant g^2(x)$. Continuing in this fashion shows that $(g^n(x))$ is monotone increasing. Moreover, there exists a $b \in U$ such that $x \leqslant b$ and $g(b) \leqslant b$. Hence $g(x) \leqslant g(b) \leqslant b$. Continuing in this fashion yields $g^n(x) \leqslant b$ for all n. We now see that $(g^n(x))$ is increasing and bounded above. Thus, there exists an $x^* \in U$ such that $(x_n) := (g^n(x))$ converges to x^*. Since g is concave and, therefore, continuous on any open set (see, e.g., Barbu and Precupanu (2012)), the result in Exercise 1.2.13 implies that $x^* = g(x^*)$.

We have treated the case $x \leqslant g(x)$ and shown existence of a fixed point. If, instead, $x \geqslant g(x)$, then $(g^n(x))$ can be shown to be decreasing and bounded by a symmetric argument. In the same way, we obtain a fixed point x^* with $g^n(x) \to x^*$.

To show the uniqueness of the fixed point, assume $g(x) = x$ and $g(y) = y$ for some $x, y \in U$. We claim that $x = y$. To see this, suppose without loss of generality that $x \leqslant y$. By assumption, there exists an $a \in U$ such that $a \leqslant x \leqslant y$ and $g(a) > a$. Because

$a \leqslant x \leqslant y$, we can take $\lambda \in [0,1]$ such that $x = \lambda a + (1-\lambda)y$. If $\lambda > 0$, then concavity of g and $g(a) > a$ implies the contradiction

$$g(x) = g(\lambda a + (1-\lambda)y) \geqslant \lambda g(a) + (1-\lambda)g(y) > \lambda a + (1-\lambda)y = x = g(x).$$

Hence $\lambda = 0$. Since $x = \lambda a + (1-\lambda)y$, this yields $x = y$.

We have proved existence of a unique fixed point in U to which every trajectory converges. □

7.1.2.2 The Multidimensional Case

Proposition 7.1.2 extends naturally to multiple dimensions. In this section we present a multidimensional version that covers both convex and concave functions.

In order to state our result, we extend the definition of convexity and concavity to vector-valued self-maps. In fact the conditions mirror those for scalar-valued functions: a self-map T on a convex subset D of $\mathbb{R}^n$ is called **convex** if

$$T(\lambda u + (1-\lambda)v) \leqslant \lambda Tu + (1-\lambda)Tv \text{ whenever } u, v \in D \text{ and } \lambda \in [0,1];$$

and **concave** if

$$\lambda Tu + (1-\lambda)Tv \leqslant T(\lambda u + (1-\lambda)v) \text{ whenever } u, v \in D \text{ and } \lambda \in [0,1].$$

Here $\leqslant$ is, as usual, the pointwise order.

We are now ready to state our next fixed point result, which was first proved in an infinite-dimensional setting by Du (1990).

Theorem 7.1.3 (Du). *Let X be a finite set, let $I := [\varphi, \psi]$ be a nonempty order interval in $(\mathbb{R}^{\mathsf{X}}, \leqslant)$ and let T be a self-map on I. If T is order-preserving, then T is globally stable on I under any one of (i)–(iv) below.*

(i) *T is concave and $T\varphi \gg \varphi$, or*

(ii) *T is concave and there exists a $\delta > 0$ such that $T\varphi \geqslant \varphi + \delta(\psi - \varphi)$, or*

(iii) *T is convex and $T\psi \ll \psi$, or*

(iv) *T is convex and there exists a $\delta > 0$ such that $T\psi \leqslant \psi - \delta(\psi - \varphi)$.*

Conditions (i) and (ii) are very similar: both require that T is concave and maps φ up strictly. In fact (ii) holds whenever (i) holds, so (ii) is the weaker (but slightly more

Figure 7.2: Du's theorem: convex and concave cases

complicated) condition. Conditions (iii) and (iv) are similar, after replacing concavity with convexity. Figure 7.2 illustrates the convex and the concave versions of the result in the simple case $n = 1$. We encourage you to sketch your own variations to get a feeling for the roles that different conditions play.

Exercise 7.1.4. The presentation in Du (1990) and Zhang (2012) requires that δ in conditions (ii) and (iv) is in $(0, 1)$. Show that this restriction is unnecessary for the conclusions of Theorem 7.1.3 to hold.

Exercise 7.1.5. Let F and G be self-maps on convex $D \subset \mathbb{R}^n$. Show that $T := F \circ G$ is concave on D whenever F and G are order-preserving and concave on D.

A full proof of Theorem 7.1.3 can be found in Du (1990) or Theorem 2.1.2 and Corollary 2.1.1 of Zhang (2012). Here we prove only uniqueness. (In our setting, existence obviously follows from Theorem 7.1.1.)

Proof of uniqueness for Theorem 7.1.3. We focus on the concave case. Let I be as in Theorem 7.1.3 and suppose that T is an order-preserving concave self map on I with $T\varphi \gg \varphi$. By Theorem 7.1.1, T has least and greatest fixed points in I that we denote by u^* and u^{**}, respectively. Let

$$\lambda := \min_{x \in \mathsf{X}} \frac{u^*(x) - \varphi(x)}{u^{**}(x) - \varphi(x)}$$

and let $\bar{x}$ be a minimizer. It follows immediately from its definition that λ obeys

$0 \leqslant \lambda \leqslant 1$ and

$$u^*(x) \geqslant \lambda u^{**}(x) + (1-\lambda)\varphi(x) \quad \text{for all } x \in \mathsf{X} \text{ with equality at } \bar{x}.$$

As a result, applying the assumed properties of T, we have

$$u^* = Tu^* \geqslant T(\lambda u^{**} + (1-\lambda)\varphi) \geqslant \lambda u^{**} + (1-\lambda)T\varphi.$$

Suppose now that $\lambda < 1$. Since $T\varphi \gg \varphi$, we get $u^* \gg \lambda u^{**} + (1-\lambda)\varphi$ and evaluating this at $\bar{x}$ yields

$$u^*(\bar{x}) > \lambda u^{**}(\bar{x}) + (1-\lambda)\varphi(\bar{x}) = u^*(\bar{x}),$$

which is a contradiction. Hence $\lambda = 1$ and, therefore, $u^* \geqslant u^{**}$. Since all fixed points $\bar{u}$ of T in I obey $u^* \leqslant \bar{u} \leqslant u^{**}$, we see that $u^* = u^{**}$ is the unique fixed point of T in I. □

7.1.3 A Power-Transformed Affine Equation

Du's theorem provides conditions under which concave or convex order-preserving self-maps on order intervals attain global stability. In this section we study maps of this type that have additional structure. While this additional structure is restrictive, it allows us to obtain global stability on unbounded subsets rather than order intervals.

To begin, consider the equation

$$v = [h + (Av)^{1/\theta}]^\theta \qquad (v \in V), \tag{7.2}$$

where θ is a nonzero parameter and V, h and A are as in §6.1.2. This system reduces to the affine model studied in Lemma 6.1.3 when $\theta = 1$.

To analyze (7.2), we introduce the self-map

$$Gv := [h + (Av)^{1/\theta}]^\theta \qquad (v \in V). \tag{7.3}$$

Continuing to assume that $h \gg 0$ and A is a positive linear operator, we can use Du's theorem to establish the next result.

Theorem 7.1.4. *If A is irreducible, then the following statements are equivalent.*

(i) $r(A)^{1/\theta} < 1$.

(ii) *G is globally stable on V.*

In the case $r(A)^{1/\theta} \geqslant 1$, the map G has no fixed point in V.

Theorem 7.1.4 implies Lemma 6.1.3. The key to proving (i) implies (ii) is that G is order-preserving and either convex or concave, depending on the value of θ. The remaining conditions in Du's theorem are established over order intervals using $r(A)^{1/\theta} < 1$. By applying an approximation argument, global stability is extended from order intervals to all of V.

Some parts of the proof of Theorem 7.1.4 are contained in the following exercises. A full proof can be found in Stachurski et al. (2022).

Let

$$F_x(t) := \left\{h(x) + t^{1/\theta}\right\}^{\theta} \qquad (t > 0).$$

Exercise 7.1.6. Prove that, for all $x \in \mathsf{X}$, the function F_x is increasing on $(0, \infty)$ and, in addition,

(i) convex on $(0, \infty)$ whenever $\theta \in (0, 1]$, and

(ii) concave otherwise (i.e., for other nonzero θ).

Exercise 7.1.7. Using Exercise 7.1.6, prove that G is order-preserving on V, convex on V whenever $\theta \in (0, 1]$, and concave otherwise.

7.2 Recursive Preferences

In this section we continue computing lifetime values of given reward processes. Our aim is to introduce more general specifications of lifetime value. In particular, we consider *recursive preferences*, which provide a much richer way of specifying how rewards are aggregated into total lifetime value. Such preferences are increasingly popular but also involve nontrivial technical problems. We will show how some common specifications of recursive utility can be handled via the fixed point methods introduced in Chapter 2.

After we have understood the process of translating recursive preferences into lifetime value, we will move on to maximizing lifetime value, i.e., dynamic programming.

7.2.1 Motivation: Optimal Savings

Here we motivate recursive preference models by analyzing consumption decisions.

7.2.1.1 A Recursive View of a Standard Model

The additively separable model of valuation in §3.2.3.3 can be studied from a purely recursive point of view. As a starting point, we state that the value V_t of current and future consumption is defined at each point in time t by the recursion

$$V_t = u(C_t) + \beta \, \mathbb{E}_t V_{t+1}. \tag{7.4}$$

The random variables V_t and V_{t+1} are the unknown objects in this expression. The expectation $\mathbb{E}_t$ is conditional on knowledge of $X_0, \ldots, X_t$ and, as before, $C_t = c(X_t)$.

Since consumption is a function of $(X_t)_{t \geqslant 0}$ and knowledge of the current state X_t is sufficient to forecast future values (by the Markov property), it is natural to guess that V_t will depend on the Markov chain only through X_t. Hence we guess there is a solution of (7.4) takes the form $V_t = v(X_t)$ for some $v \in \mathbb{R}^{\mathsf{X}}$.

Remark 7.2.1. Here v is an *ansatz*, meaning "educated guess." First we guess the form of a solution and then we try to verify that the guess is correct. So long as we carry out the second step, starting with a guess brings no loss of rigor.

Under this conjecture, (7.4) can be rewritten as $v(X_t) = u(c(X_t)) + \beta \mathbb{E}_t v(X_{t+1})$. Conditioning on $X_t = x$ and using $r := u \circ c$, this becomes

$$v(x) = r(x) + \beta \, \mathbb{E}_x \, v(X_{t+1}) = r(x) + \beta (Pv)(x) \qquad (x \in \mathsf{X}). \tag{7.5}$$

In vector form, we get $v = r + \beta P v$. From the Neumann Series Lemma, the solution is $v^* = (I - \beta P)^{-1} r$, which is identical to (3.21) on page 93.

Exercise 7.2.1. Verify our guess: Show $(V_t^*)_{t \geqslant 0}$ obeys (7.4) when $V_t^* := v^*(X_t)$.

In summary, (7.4) and the sequential representation (3.20) specify the same valuation for consumption paths.

Perhaps the recursive formulation in (7.4) seems redundant, given that it produces the same result that we obtained from the sequential approach. But, as we will see, other preferences over consumption paths imply that the sequential approach has no natural counterpart. Nonexistence of a sequential representation occurs when current value V_t is *nonlinear* in current rewards and continuation values (unlike the linear specification (7.4)). Such specifications are called **recursive preferences**. When dealing with recursive preference models, the lack of a sequential counterpart means that we are forced to proceed recursively.

Remark 7.2.2. The term "recursive preferences" is confusing, since, as we have just agreed, additively separable preferences also admit the recursive specification (7.4). The key point to remember is that when economists say "recursive preferences," they almost always refer to settings where lifetime utility can *only* be expressed recursively. We follow this convention.

7.2.1.2 Limitations of Additive Separability

Preferences that can only be expressed recursively allow us to capture choices, values and behavior that cannot be rationalized using additively separable preferences. In this section we try to clarify these ideas by highlighting some of the limitations of additively separable preferences.

One limitation of additive separability is that agents are risk-neutral in future utility (see, e.g., (7.4), where current value depends linearly on future value). Although risk aversion over consumption can be built in through curvature of u, this same curvature also determines the intertemporal elasticity of intertemporal substitution, meaning that the two aspects of preferences are tied together. (We elaborate on this point in §7.2.4.3.)

Agents with additively separable preferences are indifferent to variations in the joint distribution of rewards that leave marginal distributions unchanged. To get a sense of what this means, we present a simple setting with random consumption. In this example, you accept a new job and will be employed by this firm for the rest of your life. Your daily consumption will be entirely determined by your daily wage. Your boss offers you two options:

(A) Your boss will flip a coin at the start of your first day on the job. If the coin is heads, you will receive \$10,000 a day for the rest of your life. If the coin is tails, you will receive \$1 per day for the rest of your life.

(B) Your boss will flip a coin at the start of every day. If the coin is heads, you will receive \$10,000. If the coin is tails, you will receive \$1.

If you have a strict preference between options A and B, then your choice cannot be rationalized with additively separable preferences.

To see why, let φ be a probability distribution that represents the lottery described above, putting mass 0.5 on 10,000 and mass 0.5 on 1. Under option A, consumption $(C_t)_{t\geqslant 1}$ is given by $C_t = C_1$ for all t, where $C_1 \sim \varphi$. Under option B, consumption $(C_t)_{t\geqslant 1}$

is an IID sequence drawn from φ. Either way, lifetime utility is

$$\mathbb{E}\sum_{t\geqslant 1}\beta^t u(C_t) = \sum_{t\geqslant 1}\beta^t \mathbb{E}u(C_t) = \frac{\bar{u}}{1-\beta},$$

where $\bar{u} := \mathbb{E}u(C_1) = u(1)/2 + u(10,000)/2$.

The critical part of this argument is the passing of expectations through the sum, which uses additive separability. The implication is that lifetime utility depends only on the marginal distribution of each C_t, rather than on the joint distribution of the stochastic process $(C_t)_{t\geqslant 0}$.

7.2.2 Risk-Sensitive Preferences

Having motivated recursive preferences, let's turn to our first example: **risk-sensitive preferences**. For the consumption problem described in §7.2.1.1, imposing risk-sensitive preferences means replacing the recursion $v = r + \beta P v$ for v with

$$v(x) = r(x) + \beta\frac{1}{\theta}\ln\left\{\sum_{x'}\exp(\theta v(x'))P(x,x')\right\} \qquad (x \in \mathsf{X}). \tag{7.6}$$

As before, $r(x) = u(c(x))$ represents current utility when the current state is x. The parameter θ is a nonzero constant in $\mathbb{R}$.

In (7.6), the transform $f(v) = \exp(\theta v)$ is applied to v before expectation is taken. After the expectation is computed, the transform is undone via $f^{-1}(v) = (1/\theta)\ln(v)$. We show below that the agent can be either risk-averse or risk-loving with respect to future outcomes, depending on the value of θ.

We understand the functional equation (7.6) as "defining" lifetime utility under risk-sensitive preferences. The function v solving (7.6) gives lifetime value conditional on each current state, given θ and other primitives. Due to the nonlinearities introduced by risk sensitivity, under (7.6), there is no neat sequential representation analogous to $v(x) = \mathbb{E}_x \sum_t \beta^t u(C_t)$ from the additively separable case.

In the previous paragraph we wrote "defining" in scare quotes because we can't be sure we have a definition at this point. Just because we write down a recursive expression for lifetime utility doesn't mean that corresponding lifetime utility is actually well defined. (For example, we can happily write down the recursive vector equation $v = v + \mathbb{1}$ but no solution for this equation exists.) One aim of this chapter is to provide conditions under which recursions like (7.6) have unique and well-defined solutions.

7.2.2.1 Entropic Risk Measures

We want to understand the "expectation-like" expression on the right hand side of (7.6) that replaces the ordinary conditional expectation $\sum_{x'} v(x')P(x,x')$ from the additively separable case. To this end, we define, for arbitrary random variable ξ and $\theta \in \mathbb{R}$,

$$\mathscr{E}_\theta[\xi] := \frac{1}{\theta} \ln \{\mathbb{E}[\exp(\theta\xi)]\}.$$

The value $\mathscr{E}_\theta[\xi]$ is called the **entropic risk-adjusted expectation** of ξ given θ.

Exercise 7.2.2. Prove that, for any random variable ξ any nonzero θ and any constant c, we have $\mathscr{E}_\theta[\xi + c] = \mathscr{E}_\theta[\xi] + c$.

The key idea behind the entropic risk-adjusted expectation measure is that decreasing θ lowers appetite for risk and increasing θ does the opposite.

Exercise 7.2.3. Prove that, if ξ is normally distributed, then

$$\mathscr{E}_\theta[\xi] = \mathbb{E}[\xi] + \theta\frac{\mathrm{Var}[\xi]}{2}. \tag{7.7}$$

Expression (7.7) above shows that, for the Gaussian case, $\mathscr{E}_\theta[\xi]$ equals the mean plus a term that penalizes variance when $\theta < 0$ and rewards it when $\theta > 0$.

More generally, we have the following result.

Lemma 7.2.1. *For any random variable ξ taking values in X, we have*

(i) $\mathscr{E}_\theta[\xi] \leqslant \mathbb{E}[\xi]$ *for all* $\theta < 0$.

(ii) $\mathscr{E}_\theta[\xi] \geqslant \mathbb{E}[\xi]$ *for all* $\theta > 0$.

Moreover, both of these inequalities are strict when $\mathrm{Var}[\xi] > 0$.

Proof. Fix $\theta \in \mathbb{R}$ and let $f\colon \mathbb{R} \to (0,\infty)$ be defined by $f(x) = \exp(\theta x)$. Note that $f'(x) = \theta\exp(\theta x)$ and $f''(x) = \theta^2\exp(\theta x)$. Thus f is convex and either increasing or decreasing depending on whether θ is positive or negative. Then $\mathscr{E}_\theta[\xi] = f^{-1}(\mathbb{E}f(\xi))$. By Jensen's inequality,

$$\mathbb{E}[f(\xi)] \geqslant f(\mathbb{E}[\xi]).$$

If $\theta > 0$, then f^{-1} is increasing, so applying f^{-1} to both sides gives $\mathscr{E}_\theta[\xi] \geqslant \mathbb{E}[\xi]$. If $\theta < 0$, then f^{-1} is decreasing, so applying f^{-1} to both sides gives $\mathscr{E}_\theta[\xi] \leqslant \mathbb{E}[\xi]$. This proves the two weak inequalities in Lemma 7.2.1. To obtain strict inequalities we can

apply the same argument using a strict version of Jensen's inequality (see, e.g., Liao and Berg (2018)), which is valid when $\mathrm{Var}[\xi] > 0$. □

7.2.2.2 Existence and Uniqueness

Let's return to investigating lifetime utility under risk-sensitive preferences. To this end, we introduce the **risk-sensitive Koopmans operator** K_θ on $\mathbb{R}^{\mathsf{X}}$ via

$$(K_\theta\, v)(x) = r(x) + \beta \frac{1}{\theta} \ln \left\{ \sum_{x' \in \mathsf{X}} \exp(\theta v(x')) P(x, x') \right\} \qquad (x \in \mathsf{X}). \tag{7.8}$$

Evidently, for given nonzero θ, a function $v \in \mathbb{R}^{\mathsf{X}}$ solves the risk-sensitive preference lifetime utility specification (7.6) if and only if v is a fixed point of K_θ. This explains the significance of the following result:

Proposition 7.2.2. *If $0 < \beta < 1$, then K_θ is globally stable on $\mathbb{R}^{\mathsf{X}}$.*

We postpone a proof of Proposition 7.2.2 because we will prove a more general result in §7.2.5.1. For now we note the following implications.

(i) For each nonzero θ, lifetime utility is both well-defined and uniquely defined for risk-sensitive preference (i.e., (7.6) has a unique solution).

(ii) The unique solution, denoted henceforth by v^*, can be computed by successive approximation using K_θ.

7.2.2.3 The Gaussian Case

As a tractable case, let's suppose that $r(x) = x$ and that $X_{t+1} = \rho X_t + \sigma W_{t+1}$ where $(W_t)_{t \geqslant 1}$ is IID and standard normal. Here $|\rho| < 1$ and $\sigma \geqslant 0$ controls volatility of the state. Rather than discretizing the state process, we leave it as continuous and proceed by hand.

In this setting, the functional equation (7.6) for v becomes

$$v(x) = x + \beta \mathscr{E}_\theta[v(\rho x + \sigma W)] \tag{7.9}$$

for each $x \in \mathsf{X}$, where W is standard normal.

Since $\rho x + \sigma W$ is Gaussian, the expression (7.7) for the risk-adjusted expectation of a normal random variable leads us to conjecture that the solution v will be affine, i.e., $v(x) = ax + b$ for some $a, b \in \mathbb{R}$. This conjecture turns out to be correct:

Figure 7.3: Approximate and true solutions in the Gaussian case

Exercise 7.2.4. Verify that $v(x) = ax + b$ solves (7.9) when

$$a := \frac{1}{1 - \rho\beta} \quad \text{and} \quad b := \theta \frac{\beta}{1 - \beta} \frac{(a\sigma)^2}{2}.$$

We can see that, under the stated assumptions, lifetime value v is increasing in the state variable x. However, impacts of the parameters generally depend on θ. For example, if $\theta > 0$, increasing σ shifts up lifetime utility. If $\theta < 0$, then lifetime value decreases with σ. This is as we expect: lifetime utility is affected positively or negatively by volatility, depending on whether or not the agent is risk averse or risk loving.

Figure 7.3 shows the true solution $v(x) = ax + b$ to the risk-sensitive lifetime utility model, as well as an approximate fixed point from a discrete approximation. The discrete approximation is computed by applying successive approximation to K_θ after discretizing the state process via Tauchen's method. The parameters and discretization are shown in Listing 23.

Exercise 7.2.5. Replicate Figure 7.3.

Exercise 7.2.6. Dropping the Gaussian assumption, suppose now that consumption is IID with $C_t = c(X_t)$ where $(X_t)_{t \geqslant 0}$ is IID with distribution φ on finite set X. Now

```
using LinearAlgebra, QuantEcon

function create_rs_utility_model(;
        n=180,      # size of state space
        β=0.95,     # time discount factor
        ρ=0.96,     # correlation coef in AR(1)
        σ=0.1,      # volatility
        θ=-1.0)     # risk aversion
    mc = tauchen(n, ρ, σ, 0, 10)  # n_std = 10
    x_vals, P = mc.state_values, mc.p
    r = x_vals      # special case u(c(x)) = x
    return (; β, θ, ρ, σ, r, x_vals, P)
end
```

Listing 23: Risk sensitive utility model parameters (`rs_utility.jl`)

the operator K_θ becomes

$$(K_\theta v)(x) = r(x) + \beta \frac{1}{\theta} \ln \left\{ \sum_{x' \in \mathsf{X}} \exp(\theta v(x')) \varphi(x') \right\} \qquad (x \in \mathsf{X}).$$

Although iterating on K_θ is convergent, there is a more efficient method that reduces to solving a one-dimensional equation. Propose such a method and confirm that it is convergent. [Hint: Consider reviewing §4.2.2.2.]

7.2.3 Epstein–Zin Preferences

One of the most popular specifications of recursive preferences in quantitative research is Epstein–Zin utility.[1] This class of preferences has been used to study asset pricing, business cycles, monetary policy, fiscal policy, optimal taxation, climate policy, pension plans, and other topics. In this section we introduce the Epstein–Zin specification and discuss how to solve it. We will see that the specification, while highly nonlinear, is nonetheless well behaved.

[1] Epstein–Zin preferences were popularized in Epstein and Zin (1989). They are a special case of preferences defined by Kreps and Porteus (1978). Further discussion can be found in §7.3.

7.2.3.1 Specification

With **Epstein–Zin** preferences, the additively separable relationship $V_t = u(C_t) + \beta \mathbb{E}_t V_{t+1}$ is replaced by

$$V_t = \left\{ (1-\beta) C_t^\alpha + \beta [\mathbb{E}_t V_{t+1}^\gamma]^{\alpha/\gamma} \right\}^{1/\alpha}, \tag{7.10}$$

where γ, α are nonzero parameters and $\beta > 0$. As for risk-sensitive preferences, lack of additive separability implies that there is no neat sequential representation for lifetime value. As a result, we must work directly with the recursive expression (7.10).

Assume as before that $C_t = c(X_t)$, where $c \in \mathbb{R}_+^{\mathsf{X}}$ and $(X_t)_{t \geqslant 0}$ is P-Markov on finite set X. Hence we conjecture a solution of the form $V_t = v(X_t)$ for some $v \in V := \mathbb{R}_+^{\mathsf{X}}$. Under this conjecture, the **Epstein–Zin Koopmans operator** corresponding to (7.10) is

$$(Kv)(x) = \left\{ (1-\beta) c(x)^\alpha + \beta \left[\sum_{x'} v(x')^\gamma P(x, x') \right]^{\alpha/\gamma} \right\}^{1/\alpha} \qquad (x \in \mathsf{X}), \tag{7.11}$$

As discussed further in §7.2.4.2, the parameter γ governs risk aversion with respect to temporal gambles (where outcomes are resolved in the next period), while $\beta \in (0, 1)$ controls impatience and α parametrizes the intertemporal elasticity of substitution. The fact that all three parameters have distinct effects helps fit data. For example, see Tallarini Jr (2000) and Barillas et al. (2009).

An important question is whether Epstein–Zin preferences are well defined. In particular, what conditions do we need on primitives such that the Koopmans operator K in (7.11) has a unique fixed point?

7.2.3.2 Properties of the Koopmans Operator

To address this question we begin by slightly generalizing (7.11) and writing it in vector form as

$$Kv = \left\{ h + \beta [P v^\gamma]^{\alpha/\gamma} \right\}^{1/\alpha} \tag{7.12}$$

where $h \in \mathbb{R}^{\mathsf{X}}$. This version reduces to (7.11) when $h = (1-\beta) c^\alpha$.

In studying K, it is typically assumed that $h \geqslant 0$ in order to avoid problems such as fractional powers of negative numbers. We go one step further and assume $h \gg 0$. Although this strict inequality is not needed in all cases we consider, it eliminates negative exponents of zero and is innocuous for quantitative work, since $\min h$ can be taken arbitrarily close to zero.

Figure 7.4: Convergence of Koopmans iterates for Epstein–Zin utility

Under this assumption, it is easy to to check that

$$K \text{ is a self-map on } V := \left\{ v \in \mathbb{R}^{\mathsf{X}} \,:\, v \gg 0 \right\}.$$

Unfortunately, K is not easy to work with for two reasons. First, linear and nonlinear transformations are intertwined. Second, there are several cases for the parameters that we need to handle in order to understand stability. Nonetheless, by applying a smooth transformation, we will find it easy to show that the Epstein–Zin Koopmans operator K is globally stable under mild conditions. In particular,

Proposition 7.2.3. *If P is irreducible and $0 < \beta < 1$, then the Epstein–Zin Koopmans operator* (7.12) *is globally stable on V.*

A proof of Proposition 7.2.3 is provided in §7.2.3.3.

Proposition 7.2.3 implies that Epstein–Zin utility is well-defined under the stated conditions and, moreover, that the solution can be computed via successive approximation on K. Listing 24 provides code for performing this operation. Figure 7.4 shows convergence of the sequence of iterates to the fixed point v^*, under the parameters in Listing 24, given an initial condition v_0. The figure plots every 10th iterate, repeated 100 times.

```
include("s_approx.jl")
using LinearAlgebra, QuantEcon

function create_ez_utility_model(;
        n=200,      # size of state space
        ρ=0.96,     # correlation coef in AR(1)
        σ=0.1,      # volatility
        β=0.99,     # time discount factor
        α=0.75,     # EIS parameter
        γ=-2.0)     # risk aversion parameter

    mc = tauchen(n, ρ, σ, 0, 5)
    x_vals, P = mc.state_values, mc.p
    c = exp.(x_vals)

    return (; β, ρ, σ, α, γ, c, x_vals, P)
end

function K(v, model)
    (; β, ρ, σ, α, γ, c, x_vals, P) = model

    R = (P * (v.^γ)).^(1/γ)
    return ((1 - β) * c.^α + β * R.^α).^(1/α)
end

function compute_ez_utility(model)
    v_init = ones(length(model.x_vals))
    v_star = successive_approx(v -> K(v, model),
                               v_init,
                               tolerance=1e-10)
    return v_star
end
```

Listing 24: Epstein–Zin utility model and Koopmans operator (`ez_utility.jl`)

7.2.3.3 Proof of the Stability Result

We prove Proposition 7.2.3 by

(i) Introducing an operator $\hat{K}$ obtained from K via a smooth transformation,

(ii) proving that $\hat{K}$ and K are topologically conjugate, and

(iii) obtaining conditions under which $\hat{K}$ is globally stable.

To begin we define $\hat{K}$ via

$$\hat{K}v = \left\{h + \beta(Pv)^{1/\theta}\right\}^{\theta} \qquad \text{where} \quad \theta := \frac{\gamma}{\alpha}. \tag{7.13}$$

The operator $\hat{K}$ is simpler to work with than K because it unifies α, γ into a single parameter θ and decomposes the Epstein–Zin update rule into two parts: a linear map P and a separate nonlinear component.

Exercise 7.2.7. Prove that

(i) $\hat{K}$ is a self-map on V and

(ii) $v \in V$ is a fixed point of K if and only if v^{γ} is a fixed point of $\hat{K}$.

Lemma 7.2.4. *Let Φ be defined by $\Phi v = v^{\gamma}$. The map Φ is a homeomorphism from V to itself and (V, K) and $(V, \hat{K})$ are topologically conjugate under Φ.*

Proof. Evidently Φ is a continuous bijection from V to itself, with continuous inverse $\Phi^{-1}v = v^{1/\gamma}$. Hence Φ is a homeomorphism. In addition, for $v \in V$,

$$\hat{K}\Phi v = \left\{h + \beta(P\Phi v)^{1/\theta}\right\}^{\theta} = \left\{h + \beta[P(v^{\gamma})]^{\alpha/\gamma}\right\}^{\gamma/\alpha} = \Phi K v.$$

Thus, $\hat{K}\Phi = \Phi K$ on V. Rearranging gives $K = \Phi^{-1}\hat{K}\Phi$, so (V, K) and $(V, \hat{K})$ are topologically conjugate, as claimed. □

Proof of Proposition 7.2.3. Set $A := \beta^{\theta}P$. With this notation we have $\hat{K}v = [h+(Av)^{1/\theta}]^{\theta}$. In view of in Theorem 7.1.4 on page 210, this operator is globally stable on V whenever $r(A)^{1/\theta} < 1$. In our case $r(A) = r(\beta^{\theta}P) = \beta^{\theta}$, so $r(A)^{1/\theta} = \beta$. It follows that $\hat{K}$ is globally stable on V whenever $\beta < 1$. Since (V, K) and $(V, \hat{K})$ are topologically conjugate, the proof of Proposition 7.2.3 is complete. □

Figure 7.5: Shape properties of $\hat{K}$ in one dimension

7.2.3.4 Why Not Use Contractivity?

Can we also study stability of $\hat{K}$ using contraction arguments? Unfortunately this approach fails under useful parameterizations. To see why, suppose that X contains a single element, so that h is a constant, P is the identity, w is a scalar and $\hat{K}v = F(v)$ with $F(v) := \left\{h + \beta v^{1/\theta}\right\}^{\theta}$, as shown in Figure 7.5. Here $\theta = 5$, $h = 0.5$ and $\beta = 0.5$. We see that $\hat{K}$ has infinite slope in the neighborhood of zero, so the contraction property fails.[2]

Exercise 7.2.8. Prove that, given the parameter values used for Figure 7.5, the function F satisfies $F'(t) \to \infty$ and $t \downarrow 0$.

7.2.4 A General Representation

Now let us define recursive preferences more generally, rather than just listing examples. There are various constructions available in the literature. Many are challenging for applied researchers hoping to do quantitative work. Here we give a parsimonious operator-theoretic definition.

While this section is technical, those who work through it will be well prepared when we consider the theory of dynamic programming in depth in Chapter 8. One

[2]We could try to truncate the interval to a neighborhood of the fixed point and hope that $\hat{K}$ is a contraction when restricted to this interval. But in higher dimensions we are not sure that a fixed point exists for a broad range of parameters, which makes this idea hard to implement.

reason is that our treatment of dynamic programming is emphasizes recursive representation over sequential ones.

7.2.4.1 An Operator-Theoretic View

What is a general definition of recursive utility? A possible answer follows: Given a set X and a function class $V \subset \mathbb{R}^{\mathsf{X}}$, we define a **Koopmans operator** to be any globally stable order-preserving self-map K on V. A solution to the recursive utility problem, also called **lifetime utility**, is a fixed point of K in V. Given such a v, the value $v(x)$ is interpreted as lifetime value conditional on initial state x.

Example 7.2.1. In the case of additively separable preferences, lifetime value was defined in (3.21) by $v := (I - \beta P)^{-1} r$. Equivalently, v is the fixed point of the operator K defined by $Kv = r + \beta P v$. We know that K is globally stable and order-preserving on $V := \mathbb{R}^{\mathsf{X}}$ whenever $\beta \in (0, 1)$. Hence K is a Koopmans operator on V with a fixed point that reflects lifetime value.

Example 7.2.2. The operator K_θ defined in (7.8) is order-preserving and, by Proposition 7.2.2, globally stable on V when $\beta \in (0, 1)$. Hence K_θ is a Koopmans operator. Moreover, K_θ has the property that lifetime value under risk-sensitive preferences corresponds to a $v \in V := \mathbb{R}^{\mathsf{X}}$ satisfying $v = K_\theta v$.

Our definition of a Koopmans operator requires some justification. First, it is generally appropriate to suppose that a uniform increase in continuation values will increase current value. Such monotonicity justifies the assumption that K is order-preserving. Second, regarding global stability, it is natural to assume that K has a unique fixed point, since that fixed point defines lifetime value. If K fails to have a unique fixed point then the recursive preference specification is not well posed. Our assumption of global stability strengthens existence and uniqueness by requiring global convergence to the fixed point under iteration.

This global stability condition has the following interpretation. For given $m \in \mathbb{N}$, the function $K^m w$ computes utility over m periods with terminal condition $w \in V$. Hence global stability is the requirement that, for any choice of terminal condition, finite-horizon utility converges to infinite-horizon utility as the time horizon converges to infinity. The next exercise helps to illustrate this point.

Exercise 7.2.9. Consider again the additively separable preferences $V_t = u(C_t) + \beta \mathbb{E}_t V_{t+1}$ in Example 7.2.1. Suppose that the time horizon is finite, with some exogenous terminal value $V_m = w(X_m)$ at time m. Letting $v_m(x)$ represent lifetime value up until time m, conditional on initial state x, show that

(i) $v_m = \sum_{t=0}^{m-1} (\beta P)^t r + (\beta P)^m w$,

(ii) $v_m = K^m w$, where K is the associated Koopmans operator $Kv := r + \beta P v$ and,

(iii) $K^m w \to v^* := (I - \beta P)^{-1} r$ as $m \to \infty$.

Exercise 7.2.9 confirms that, at least for the additively separable case, global stability of K is equivalent to the statement that a finite-horizon valuation with arbitrary terminal condition w converges to the infinite-horizon valuation.

7.2.4.2 Risk-Adjusted Expectation

The operator-theoretic definition of Koopmans operators in §7.2.4.1 is general but also rather abstract. Let's now outline some additional structure that can be combined to generate Koopmans operators in a Markov environment. The two key components are an aggregation function and a certainty equivalent operator. In the remainder of this section, we will build Koopmans operators from these primitives and connect them to applications.

Given $V \subset \mathbb{R}^{\mathsf{X}}$, we define a **certainty equivalent operator** on V to be an order-preserving self-map R on V such that all constants are fixed points. More precisely,

$$R\lambda\mathbb{1} = \lambda\mathbb{1} \quad \text{for all } \lambda \in \mathbb{R} \text{ such that } \lambda\mathbb{1} \in V.$$

(Here $\mathbb{1} := \mathbb{1}_{\mathsf{X}}$.)

Example 7.2.3. The usual conditional expectations operator is a certainty equivalent operator. To see this, set $V := \mathbb{R}^{\mathsf{X}}$ and fix $P \in \mathcal{S}(\mathsf{X})$. The operator R on V defined by

$$(R\,v)(x) := (P\,v)(x) = \sum_{x' \in \mathsf{X}} v(x')P(x, x') \qquad (v \in V,\ x \in \mathsf{X})$$

preserves order and $R\lambda\mathbb{1} = P\lambda\mathbb{1} = \lambda P\mathbb{1} = \lambda\mathbb{1}$.

In the last example, the certainty equivalent is a linear operator. The next example is nonlinear. It treats the risk-adjusted expectation that appears in risk-sensitive preferences.

Example 7.2.4. Let $V = \mathbb{R}^{\mathsf{X}}$ and fix $P \in \mathcal{S}(\mathsf{X})$. The operator

$$(R_e\,v)(x) := \frac{1}{\theta} \ln \left\{ \sum_x \exp(\theta v(x'))P(x, x') \right\} \qquad (v \in V,\ x \in \mathsf{X})$$

is a certainty equivalent operator on V for all $\theta \neq 0$. R_e is called the **entropic** risk-adjusted conditional expectation.

Exercise 7.2.10. Confirm that R_e is a certainty equivalent operator.

Example 7.2.5. As a third example, let $V = \mathbb{R}_+^{\mathsf{X}}$. The operator

$$(R_\gamma v)(x) := \left\{ \sum_x v^\gamma(x') P(x, x') \right\}^{1/\gamma} \qquad (v \in V,\ x \in \mathsf{X}) \tag{7.14}$$

is a certainty equivalent operator on V for all $\gamma \neq 0$.[3] The map R_γ is sometimes called the **Kreps-Porteus expectations operator** in honor of Kreps and Porteus (1978).

We met R_γ in §7.2.3, when we discussed Epstein–Zin preferences.

Exercise 7.2.11. Confirm that R_γ is a certainty equivalent operator.

Exercise 7.2.12. Let $V = \mathbb{R}^{\mathsf{X}}$ and fix $P \in \mathcal{S}(\mathsf{X})$ and $\tau \in [0, 1]$. Let $(R_\tau v)(x) := Q_\tau v(X)$ where $X \sim P(x, \cdot)$ and Q_τ is the quantile functional (see page 29). More specifically,

$$(R_\tau v)(x) = \min \left\{ y \in \mathbb{R} \;\middle|\; \sum_{x'} \mathbb{1}\{v(x') \leqslant y\} P(x, x') \geqslant \tau \right\} \qquad (v \in V,\ x \in \mathsf{X}).$$

Show that R_τ defines a certainty equivalent operator on V.

The set of certainty equivalent operators on $\mathbb{R}^{\mathsf{X}}$ is invariant under convex combinations, as the next exercise asks you to confirm.

Exercise 7.2.13. Let $\mathscr{R}$ be the set of certainty equivalent operators on $\mathbb{R}^{\mathsf{X}}$ and prove the following: If $R_a, R_b \in \mathscr{R}$ and $R := \alpha R_a + (1 - \alpha) R_b \in \mathscr{R}$ for some $0 \leqslant \alpha \leqslant 1$, then $R \in \mathscr{R}$.

We call a certainty equivalent operator R on V **subadditive** on V whenever

$$R(v + \lambda) \leqslant Rv + \lambda \quad \text{for all } v \in V \text{ and } \lambda \in \mathbb{R}_+ . \tag{7.15}$$

While the subadditive condition (7.15) is nontrivial, it holds for some useful cases.

[3] The function v^γ is defined by $v^\gamma(x) = (v(x))^\gamma$ for all x. We restrict attention to $V = \mathbb{R}_+^{\mathsf{X}}$ to ensure that (7.14) is well defined. If $\gamma < 0$ and $v(x) = 0$ for some $x \in \mathsf{X}$ we set $R_\gamma(v, \varphi) := 0$.

Exercise 7.2.14. Prove that the quantile certainty equivalent operator R_τ from Exercise 7.2.12 is subadditive.

Exercise 7.2.15. Show that the entropic certainty equivalent operator R_e from Example 7.2.4 is subadditive.

Exercise 7.2.16. Prove: If R is subadditive on V, then R is nonexpansive with respect to the supremum norm. That is,

$$\|Rv - Rw\|_\infty \leqslant \|v - w\|_\infty \quad \text{for all } v, w \in \mathbb{R}^{\mathsf{X}}.$$

7.2.4.3 Aggregation

We mentioned above that Koopmans operators are typically constructed by combining a certainty equivalent operator and an aggregation function. Let's now discuss the second of these components.

For the purposes of this chapter, an **aggregator** A is a map from $(x, y) \in \mathsf{X} \times \mathbb{R}$ to $A(x, y) \in \mathbb{R}$ such that

(i) $w(x) := A(x, v(x))$ is in V whenever $v \in V$ and

(ii) $y \mapsto A(x, y)$ is increasing for all $x \in \mathsf{X}$.

Intuitively, an aggregator combines current state and continuation values to measure lifetime value.

Common types of aggregators include

- the **additive aggregator** $A(x, y) = r(x) + \beta y$ with $r \in \mathbb{R}^{\mathsf{X}}$ and $\beta \in \mathbb{R}$,
- the **Leontief aggregator** $A(x, y) = \min\{r(x), \beta y\}$ with $r \in \mathbb{R}^{\mathsf{X}}$ and $\beta \in \mathbb{R}$,
- the **Uzawa aggregator** $A(x, y) = r(x) + b(x)y$ with $r, b \in \mathbb{R}^{\mathsf{X}}$, and
- the **CES aggregator** $A(x, y) = \{r(x)^\alpha + \beta y^\alpha\}^{1/\alpha}$ with $r \in \mathbb{R}^{\mathsf{X}}$, $\beta \in \mathbb{R}$ and $\alpha \neq 0$.

Here CES stands for "constant elasticity of substitution."

Below, when we discuss stability properties associated with different preferences, it will be useful to place restrictions on aggregators. One useful restriction is related to

Blackwell's sufficient condition for a contraction (§2.1.2.2): we say that an aggregator A satisfies **Blackwell's condition** if there exists a $\beta \in (0,1)$ such that

$$A(x, y + \lambda) \leqslant A(x, y) + \beta\lambda \tag{7.16}$$

for all $x \in \mathsf{X}$, $y \in \mathbb{R}$ and $\lambda \in \mathbb{R}_+$. In §7.2.5 we discuss the connection between Blackwell's condition and stability.

Exercise 7.2.17. Fix $\beta \in (0,1)$ and any $r \in \mathbb{R}^{\mathsf{X}}$. Show that, under this restriction on β, Blackwell's condition holds for both the additive aggregator $A(x, y) = r(x) + \beta y$ and the Leontief aggregator $A(x, y) = \min\{r(x), \beta y\}$.

7.2.4.4 Building Koopmans Operators

We are now ready to build Koopmans operators by combining certainty equivalents and aggregators. Formally, given $V \subset \mathbb{R}^{\mathsf{X}}$, we combine an aggregator A and a certainty equivalent operator R on V to construct

$$(Kv)(x) := A(x, (Rv)(x)) \qquad (v \in V,\ x \in \mathsf{X}). \tag{7.17}$$

It follows directly from the definitions of A and R that K is an order-preserving self-map on V. We cannot yet call K in (7.17) a Koopmans operator, however, since global stability may fail without more conditions on the primitives. In §7.2.5 we address the issue of stability.

Example 7.2.6. For risk-sensitive preferences, the Koopmans operator defined on page 216 can be expressed as

$$(K_\theta v)(x) = r(x) + \beta(R_e v)(x) \qquad (x \in \mathsf{X}),$$

where R_e is the entropic risk-adjusted conditional expectation. This is a special case of (7.17) with $V = \mathbb{R}^{\mathsf{X}}$, and $A(x, y) = r(x) + \beta y$.

Example 7.2.7. The Epstein–Zin Koopmans operator can be expressed as

$$(Kv)(x) = \left\{(1-\beta)c(x)^\alpha + \beta((R_\gamma v)(x))^\alpha\right\}^{1/\alpha} \qquad (x \in \mathsf{X}), \tag{7.18}$$

where R_γ is the Kreps–Porteus expectations operator, as defined in (7.14). This is a version of (7.17) under the CES aggregator.

Remark 7.2.3. We defined additive separability somewhat loosely in §3.2.3.3. Here is a better definition: The Koopmans operator K in (7.17) represents **additively separable preferences** if (a) the certainty equivalent operator R is ordinary expectations (as in Example 7.2.3) and (b) A is an additive aggregator.

7.2.4.5 Comments on the Elasticity of Intertemporal Substitution

The CES aggregator is so-named because, in a static utility maximization problem where c and y are two goods and utility is $U(c, y) = ((1-\beta)c^\alpha + \beta y^\alpha)^{1/\alpha}$, the elasticity of substitution is constant and given by $1/(1-\alpha)$. In the present setting, where aggregation is across time, $1/(1-\alpha)$ is usually called the **elasticity of intertemporal substitution** (EIS). The next exercise explains.

Exercise 7.2.18. Consider $U(c, y) = ((1-\beta)c^\alpha + \beta y^\alpha)^{1/\alpha}$ as a utility function over current and future goods c and y. Then

$$\text{EIS} := \frac{d\ln(y/c)}{d\ln(U_c/U_y)} \quad \text{where} \quad U_c := \frac{\partial U(c,y)}{\partial c} \quad \text{and} \quad U_y := \frac{\partial U(c,y)}{\partial y}.$$

Confirm that EIS $= 1/(1-\alpha)$.

The fact that EIS $= 1/(1-\alpha)$ under the CES aggregator is significant because the EIS can be measured from data using regression and other techniques. While estimates vary significantly, the detailed meta-analysis by Havranek et al. (2015) suggests 0.5 as a plausible average value for international studies, with rich countries tending slightly higher. Basu and Bundick (2017) use 0.8 when calibrating to US data.

Under these estimates, the relationship EIS $= 1/(1-\alpha)$ implies a value for α between -1.0 and -0.25. These numbers provide further evidence against additive separability (which requires $\alpha = 1$ in the CES aggregator).

7.2.5 Stability

As stated above, for K in (7.17) to qualify as a Koopmans operator we require it to be globally stable on its domain V. We discussed special cases of stability above. Here we present some more general results.

7.2.5.1 The Subadditive Case

Let R be a certainty equivalent operator on $V := \mathbb{R}^{\mathsf{X}}$ and let A be an aggregator. Let K be defined by $(Kv)(x) := A(x, (Rv)(x))$. We have the following stability result.

Proposition 7.2.5. *If A satisfies Blackwell's condition and R is subadditive, then K is a contraction map on $(V, \|\cdot\|_\infty)$. In particular, K is a Koopmans operator.*

Proof. Let the primitives be as stated. In view of Lemma 2.1.2 on page 49, and taking into account the fact that K is order-preserving, we need only show that there exists a $\beta \in (0,1)$ with $K(v+\lambda) \leqslant Kv + \beta\lambda$ for all $v \in V$ and $\lambda \in \mathbb{R}_+$. To see that this holds, fix $v \in V$ and $\lambda \in \mathbb{R}_+$. Applying subadditivity and monotonicity of A, and writing $\lambda\mathbb{1}$ as λ, we have

$$K(v+\lambda) = A(\cdot, R(v+\lambda)) \leqslant A(\cdot, Rv+\lambda) \leqslant A(\cdot, Rv) + \beta\lambda,$$

where β is as defined in (7.16). In other words, $K(v+\lambda) \leqslant Kv + \beta\lambda$. Hence K is a contraction of modulus β on V. □

We can now complete the proof of Proposition 7.2.2.

Proof of Proposition 7.2.2. Fix $\beta \in (0,1)$ and note that K_θ in (7.8) can be expressed as $(K_\theta v)(x) := A(x, (R_e v)(x))$ when A is the additive aggregator and R_e is the entropic risk-adjusted conditional expectation. Since the additive aggregator satisfies Blackwell's condition $A(x, y+\lambda) \leqslant A(x,y) + \beta\lambda$ and R_e is subadditive (Exercise 7.2.15), Proposition 7.2.5 applies. In particular, K_θ is globally stable on $\mathbb{R}^{\mathsf{X}}$. □

Exercise 7.2.19. Under the Leontief aggregator, the operator (7.17) becomes

$$(Kv)(x) = \min\{r(x), \beta(Rv)(x)\}.$$

Prove that K is globally stable whenever R is subadditive and $\beta \in (0,1)$.

7.2.5.2 Quantile Preferences

Consider a setting where $V = \mathbb{R}^{\mathsf{X}}$ and quantile conditional expectations are combined with additive aggregation to obtain

$$(Kv)(x) = r(x) + \beta(R_\tau v)(x) \qquad (x \in \mathsf{X}) \tag{7.19}$$

where $\beta \in (0,1)$, $\tau \in [0,1]$, $r \in \mathbb{R}^{\mathsf{X}}$ and R_τ is as described in Exercise 7.2.12. Since R_τ is subadditive and the additive aggregator satisfied Blackwell's condition, K is globally stable and hence a Koopmans operator. The operator K represents quantile preferences, as described in de Castro and Galvao (2019) and other studies (see 7.3). The value τ parameterizes attitude to risk, a point we return to in §8.2.1.4.

Exercise 7.2.20. Consider replacing the operator K in (7.19) with $(Kv)(x) = \min\{r(x), \beta(R_\tau v)(x)\}$. Under the same assumptions as above (apart from the switch to Leontief aggregator), prove that K is globally stable.

7.2.5.3 Uzawa Aggregation

Let's consider the operator K defined by $(Kv)(x) = A(x, (Rv)(x))$ when A is of Uzawa type (see §7.2.4.3). Fixing $P \in \mathcal{S}(\mathsf{X})$ and $r, b \in \mathbb{R}^{\mathsf{X}}$, we set $V = \mathbb{R}^{\mathsf{X}}$, $R = P$ and $A(x,y) = r(x) + b(x)y$ to obtain

$$(Kv)(x) = r(x) + b(x) \sum_{x'} v(x')P(x,x') \qquad (x \in \mathsf{X},\ v \in V). \tag{7.20}$$

We can write this in vector form as $Kv = r + Lv$ where L is the linear operator with matrix $L(x,x') = b(x)P(x,x')$. It follows from Exercise 1.2.14 on page 21 that K is globally stable whenever $r(L) < 1$.

With some additional requirements, the condition $r(L) < 1$ is necessary as well as sufficient. Indeed, if $b \geqslant 0$ and P is irreducible, then L is also irreducible and a positive linear operator.

Exercise 7.2.21. Confirm that L is irreducible when $b \gg 0$ and P is irreducible.

Applying Lemma 6.1.3, we see that $r \gg 0$ and $r(L) \geqslant 1$ implies $Kv = r + Lv$ has no fixed point in $V := \{v \in \mathbb{R}^{\mathsf{X}} : v \gg 0\}$.

These ideas connect to ongoing research. For example, in Krusell and Smith (1998), Toda (2019), Cao (2020), Hubmer et al. (2020), and related sources, households maximize utility subject to a budget constraint while their discount factor fluctuates according to some state variable. For a given consumption path (C_t), lifetime values takes the form

$$v(x) := \mathbb{E}_x \sum_{t=0}^{\infty} \left(\prod_{i=1}^{t} \beta_i \right) u(C_t) \tag{7.21}$$

where u is a flow utility function. Suppose that $C_t = c(X_t)$ and $\beta_t = b(X_t)$ for some P-Markov chain (X_t) and $P \in \mathcal{S}(\mathsf{X})$. Set $r := u \circ c$ and $L(x,x') := b(x)P(x,x')$. In

view of Theorem 6.1.1, the condition $r(L) < 1$ implies that the value function v obeys $v = r + Lv$. In other words, v is a fixed point of the operator $Kv = r + Lv$. As just mentioned, K is globally stable when $r(L) < 1$.

To summarize the preceding discussion, lifetime value under (7.21) is the unique fixed point of the Koopmans operator K when the aggregator is of Uzawa type and certainty equivalents are ordinary mathematical expectations.

In §8.2.3 we will discuss how to maximize lifetime value under these preferences.

7.3 Chapter Notes

The discounted additively separable preference structure introduced in §7.2.1 was used by Samuelson (1939). An axiomatic foundation was supplied by Koopmans (1960). A critical review can be found in Frederick et al. (2002).

In §7.2.1.2 we adopted an artificial setting to motivate recursive preferences. Do the restrictions in additively separability really matter for economic modeling? Evidence suggests that the answer is affirmative. In macroeconomics and asset pricing in particular, researchers increasingly use non-separable preferences in order to bring model outputs closer to the data. For example, many quantitative models of asset pricing rely heavily on Epstein–Zin preferences. Representative examples include Tallarini Jr (2000), Bansal and Yaron (2004), Hansen et al. (2008), Bansal et al. (2012), and Schorfheide et al. (2018). Theoretical properties and solution methods are discussed in Epstein and Zin (1989), Weil (1990), Epstein and Zin (1991), Pohl et al. (2018) and De Groot et al. (2022).

An accessible introduction to recursive preference models can be found in Backus et al. (2004). Our use of the term "Koopmans operator," which is used by some authors but not entirely standard, honors early contributions by Nobel laureate Tjalling Koopmans on recursive preferences. Theoretical properties of recursive preference models are studied in many papers, including Boyd (1990), Hansen and Scheinkman (2009), Marinacci and Montrucchio (2010), Bommier et al. (2017), Bloise and Vailakis (2018), Marinacci and Montrucchio (2019), Pohl et al. (2019), Balbus (2020), Borovička and Stachurski (2020), DeJarnette et al. (2020), and Christensen (2022). The paper by Marinacci and Montrucchio (2019) provides a useful alternative approach to existence of unique fixed points in the setting of order-preserving maps.

There is a strong connection between risk-sensitive preferences and the literature on robust control. See, for example, Cagetti et al. (2002), Hansen and Sargent (2007), and Barillas et al. (2009). Risk-sensitivity is studied in an optimal growth

setting by Bäuerle and Jaśkiewicz (2018). Risk-sensitivity is also used in applications of reinforcement learning, where the underlying state process is not known. See, for example, Shen et al. (2014), Majumdar et al. (2017) or Gao et al. (2021).

The quantile preferences we considered in §7.2.5.2 have been analyzed in static and dynamic settings by Giovannetti (2013), de Castro and Galvao (2019), de Castro and Galvao (2022) and de Castro et al. (2022). Recursive components of the analysis of quantile and Uzawa preference models build on the study of monotone preferences in Bommier et al. (2017).

Some recursive preference specifications involve ambiguity aversion. An introduction to this literature and its applications can be found in Klibanoff et al. (2009), Hayashi and Miao (2011), Hansen and Miao (2018), Bommier et al. (2019) and Hansen and Sargent (2020).

Chapter 8

Abstract Dynamic Programming

While the MDP model from Chapters 5–6 is elegant and highly influential, researchers in economics, finance, and other fields have wanted to extend that framework. Reasons include:

(i) MDP theory cannot be applied to settings where lifetime values are described by the kinds of nonlinear recursive preferences discussed in Chapter 7.

(ii) Equilibria in some models of production and economic geography can be computed using dynamic programming but not all of these programming problems fit within the framework of MDPs.

(iii) Dynamic programming problems that acknowledge model uncertainty by including adversarial agents to promote robust decision rules can fail to be MDPs.

(Section 8.5 provides notes and references.)

To handle such departures from the MDP assumptions, we construct an *abstract dynamic programming* framework, building on an approach to optimization initially developed by Eric Denardo (1937–) and Loring Mitten (1921–2000), and recently extended and promoted by Bertsekas (2022b). Further references are provided in §8.5.

Apart from generality, another motivation for greater abstraction is a desire for clarity: while we have already provided some optimality proofs (such as the proof of Proposition 5.1.1 on optimality for MDPs), the abstract DP framework allows us to strip away extraneous structure and isolate key assumptions.

We begin §8.1 with a framework that centers on an abstract representation of the Bellman equation. We then state optimality results and show how they can be verified in a range of applications.

We defer proofs of optimality results until §8.3, where we strip dynamic programs down to their essence by adopting a purely operator-theoretic approach. This Puritanical approach simplifies proofs and allows us to extend dynamic programming to a broad class of applications.

8.1 Recursive Decision Processes

In this section we introduce recursive decision processes that by extending the MDP model allow us to treat not only MDPs but also models with recursive preferences, state-dependent discounting, and other useful features.

8.1.1 Defining RDPs

Consider a generic Bellman equation of the form

$$v(x) = \max_{a \in \Gamma(x)} B(x, a, v). \tag{8.1}$$

Here x is the state, a is an action, Γ is a feasible correspondence, and B is an "aggregator" function. We understand $\Gamma(x)$ as all actions available to the controller in state x. The function v assigns values to states and is a member of some class $V \subset \mathbb{R}^{\mathsf{X}}$. This "abstract" Bellman equation generalizes all of the Bellman equations presented in previous chapters.

Our plan is to analyze the Bellman equation (8.1) and state conditions on B and the other primitives that make strong optimality properties hold. As a first step, we introduce two finite sets,

- an **action space** A and
- a **state space** X

Given X and A, we define a **recursive decision process** (RDP) to be a triple (Γ, V, B) consisting of

(i) a **feasible correspondence** Γ that is a nonempty correspondence from X to A; we set

- the feasible state-action pairs

$$\mathsf{G} := \{(x, a) \in \mathsf{X} \times \mathsf{A} : a \in \Gamma(x)\},$$

- a set of feasible policies

$$\Sigma := \{\sigma \in \mathsf{A}^{\mathsf{X}} : \sigma(x) \in \Gamma(x) \text{ for all } x \in \mathsf{X}\};$$

(ii) a **value space** V that is a sublattice of $\mathbb{R}^{\mathsf{X}}$, and

(iii) a **value aggregator** B that maps $\mathsf{G} \times V$ to $\mathbb{R}$ and satisfies both the monotonicity condition

$$v, w \in V \text{ and } v \leqslant w \implies B(x, a, v) \leqslant B(x, a, w) \text{ for all } (x, a) \in \mathsf{G}, \tag{8.2}$$

and the consistency condition

$$\sigma \in \Sigma \text{ and } v \in V \implies w \in V \quad \text{where } w(x) := B(x, \sigma(x), v). \tag{8.3}$$

The statement that V is a **sublattice** of $\mathbb{R}^{\mathsf{X}}$ means that $f, g \in V$ implies $f \vee g \in V$ and $f \wedge g \in V$. Throughout, $\leqslant$ represents the pointwise order on $\mathbb{R}^{\mathsf{X}}$.

The definition of the feasible correspondence in (i) is identical to that for the MDP in Chapter 5. As for (ii), we understand V to be a class of functions that assign values to states. In (iii), the interpretation of the aggregator B is:

> $B(x, a, v)$ = total lifetime rewards, contingent on current action a, current state x, and using v to evaluate future states.

As we will see, determining optimal choices is greatly simplified if we can find the "right" v for evaluating states. Locating and calculating this v is one of our major concerns.

The monotonicity condition (8.2) is natural: if, relative to v, rewards are at least as high for w in every future state, then the total rewards one can extract under w should be at least as high. The consistency condition in (8.3) ensures that as we consider values of different policies we remain within the value space V.

Every MDP is an RDP, as the next example clarifies.

Example 8.1.1. Consider an arbitrary MDP (Γ, β, r, P) with state space X and action space A (see, e.g., §5.1.1). We can frame this model as an RDP (Γ, V, B) such that the resulting abstract Bellman equation $v(x) = \max_{a \in \Gamma(x)} B(x, a, v)$ coincides with the MDP Bellman equation. To do so, we take Γ as the feasible correspondence for the RDP and $V = \mathbb{R}^{\mathsf{X}}$ as the value space. For the aggregator we set

$$B(x, a, v) = r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \qquad ((x, a) \in \mathsf{G},\ v \in V). \tag{8.4}$$

Now (Γ, V, B) forms an RDP. The set $\mathbb{R}^{\mathsf{X}}$ is certainly a sublattice of $\mathbb{R}^{\mathsf{X}}$. The monotonicity condition (8.2) clearly holds and the consistency condition (8.3) is trivial in this case, since V is all of $\mathbb{R}^{\mathsf{X}}$. Inserting the aggregator (8.4) into $v(x) = \max_{a \in \Gamma(x)} B(x, a, v)$ recovers the MDP Bellman equation (see (5.2) on page 126).

Example 8.1.2. Consider a basic "cake eating" problem where X is a finite subset of $\mathbb{R}_+$ and $x \in \mathsf{X}$ is understood to be the number of remaining slices of cake. Let x' be the number of remaining slices next period and $u(x - x')$ be the utility from slices enjoyed today. The function $u \colon \mathbb{R}_+ \to \mathbb{R}$ is given. With $\mathsf{A} = \mathsf{X}$, $V = \mathbb{R}^{\mathsf{X}}$,

$$B(x, x', v) = u(x - x') + \beta v(x')$$

and $\Gamma(x) = \{x' \in \mathsf{X} : x' \leqslant x\}$ we obtain an RDP. The monotonicity condition (8.2) clearly holds and the consistency condition (8.3) is trivial in this case.

The last example is redundant: we showed that the cake eating problem is an MDP in §5.1.2.3 and Example 8.1.1 demonstrates that every MDP is an RDP. Nonetheless, Example 8.1.2 is instructive because, for cake eating, the MDP construction is tedious (one reason is that we need to define a stochastic kernel P even though transitions are deterministic), while the RDP construction is straightforward.

The next example makes a related point.

Example 8.1.3. In §5.1.2.4 we showed that the job search model is an MDP but the construction was tedious. But we can also represent job search as an RDP and the embedding is straightforward. To see this, recall that, for an arbitrary optimal stopping problem with primitives as described in Chapter 4, the Bellman equation is

$$v(x) = \max \left\{ e(x), c(x) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, x') \right\} \qquad (x \in \mathsf{X}). \tag{8.5}$$

If we set $V = \mathbb{R}^{\mathsf{X}}$, let $\Gamma(x) = \{0, 1\}$ and

$$B(x, a, v) = a e(x) + (1 - a) \left[c(x) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, x') \right] \tag{8.6}$$

for $x \in \mathsf{X}$ and $a \in \mathsf{A} := \{0, 1\}$, then (Γ, V, B) is an RDP. Inserting B from (8.6) into the RDP Bellman equation $v(x) = \max_{a \in \Gamma(x)} B(x, a, v)$ and setting $\Gamma(x) = \{0, 1\}$ reproduces the Bellman equation (8.5). Job search is a special case of optimal stopping.

Example 8.1.4. The dynamic programming framework popularized by Stokey and Lucas (1989) is characterized by two features: First, the state is divided into an exogenous process (Z_t) and an endogenous process (Y_t). In addition, the next period

endogenous state is directly chosen by the current action. The Bellman equation takes the form

$$v(y,z) = \max_{y' \in \Gamma(y,z)} \left\{ F(y,z,y') + \beta \sum_{z'} v(y',z')Q(z,z') \right\}. \tag{8.7}$$

We assume that (Z_t) is Q-Markov on finite set Z and (Y_t) takes values in finite set Y. With state space $\mathsf{X} := \mathsf{Y} \times \mathsf{Z}$, value space $\mathbb{R}^{\mathsf{X}}$, action space Y, feasible correspondence $x \mapsto \Gamma(x)$ and aggregator

$$B(x,a,v) = B((y,z),y',v) = F(y,z,y') + \beta \sum_{z'} v(y',z')Q(z,z'),$$

we obtain an RDP with Bellman equation identical to (8.7).

Exercise 8.1.1. Show that the RDP in Example 8.1.4 can be expressed as an MDP.

Examples 8.1.1–8.1.4 treated RDPs that can be embedded into the MDP framework. In the remaining examples, we consider models that cannot be represented as MDPs and show how each one can nonetheless be viewed an RDP.

Example 8.1.5 (State-Dependent Discounting)**.** We can add state-dependent discounting to Example 8.1.1 by changing the aggregator to

$$B(x,a,v) = r(x,a) + \sum_{x'} v(x')\beta(x,a,x')P(x,a,x'), \tag{8.8}$$

where β is some function from $\mathsf{G} \times \mathsf{X}$ to $\mathbb{R}_+$. As before we take $V = \mathbb{R}^{\mathsf{X}}$. The resulting RDP (Γ, V, B) represents the MDP with state-dependent discounting we analyzed on Chapter 6. In §8.2.3 we will use RDP theory developed in this chapter to verify the optimality results claimed Chapter 6.

Exercise 8.1.2. Verify that (Γ, V, B) as defined in Example 8.1.5 is in fact an RDP.

Example 8.1.6. We can modify the MDP in Example 8.1.1 to use risk-sensitive preferences (see Jacobson (1973), Whittle (1981), and for a discounted case, Hansen and Sargent (1995).) We do this by taking V to be all of $\mathbb{R}^{\mathsf{X}}$ and setting

$$B(x,a,v) = r(x,a) + \beta \frac{1}{\theta} \ln \left\{ \sum_{x' \in \mathsf{X}} \exp(\theta v(x'))P(x,a,x') \right\} \tag{8.9}$$

for all $(x,a) \in \mathsf{G}$ and $v \in V$. Here G is generated by the feasible correspondence Γ.

Exercise 8.1.3. Confirm that the risk-sensitive model (Γ, V, B) in Example 8.1.6 is an RDP for all nonzero θ.

Example 8.1.7 (Epstein–Zin Preferences)**.** We can also modify the MDP in Example 8.1.1 to use the Epstein–Zin specification (see (7.10) on page 219) by setting

$$B(x, a, v) = \left\{ r(x, a) + \beta \left[\sum_{x' \in \mathsf{X}} v(x')^{\gamma} P(x, a, x') \right]^{\alpha/\gamma} \right\}^{1/\alpha}, \tag{8.10}$$

where $\beta \in (0, 1)$ and γ and α are nonzero parameters. Assume, as in §7.2.3.2, that $r \gg 0$ and P is irreducible. If V is all $v \in \mathbb{R}^{\mathsf{X}}$ such that $v \gg 0$, then the triple (Γ, V, B) is an RDP.

Exercise 8.1.4. Confirm the last claim in Example 8.1.7.

Example 8.1.8. The **shortest path problem** considers optimal traversal of a directed graph $\mathscr{G} = (\mathsf{X}, E)$, where X is the vertices of the graph and E is the edges. A weight function $c \colon E \to (0, \infty)$ associates positive cost to each edge $(x, x') \in E$. The aim is to find the minimum cost path from x to a specified node d for every $x \in \mathsf{X}$. The problem can be solved by applying a Bellman operator of the form

$$(Tv)(x) = \min_{x' \in \mathscr{O}(x)} \{c(x, x') + v(x)\} \qquad (x \in \mathsf{X}), \tag{8.11}$$

where $\mathscr{O}(x) := \{x' \in \mathsf{X} : (x, x') \in E\}$ is the direct successors of x and $v(x')$ is the minimum cost-to-go from state x'. The problem is not an MDP because future values are not discounted. However, it can be framed as an RDP (Γ, V, B), by taking X as the state and action spaces, $V = \mathbb{R}^{\mathsf{X}}$, $\Gamma(x) = \mathscr{O}(x)$, and $B(x, x', v) = c(x, x') + v(x')$.

The last application is an example of a minimization problem. While we treat minimization explicitly in §8.4.2, the shortest path setting can easily be converted maximization by replacing $c(x, x')$ with $-c(x, x')$. This produces an application very similar to the cake eating problem in Example 8.1.2, although discounting is eliminated and network structure shows up in the constraint.

8.1.2 Lifetime Value

We aim to discuss optimality of RDPs. To prepare for this topic, we now clarify lifetime values associated with different policy choices in the RDP setting.

8.1.2.1 Policies and Operators

Let (Γ, V, B) be an RDP with state and action spaces X and A, and let Σ be the set of all feasible policies. For each $\sigma \in \Sigma$ we introduce the **policy operator** T_σ as a self-map on V defined by

$$(T_\sigma v)(x) = B(x, \sigma(x), v) \qquad (x \in \mathsf{X}). \tag{8.12}$$

The RDP policy operator is a direct generalization of the MDP policy operator defined on page 132, as well as the optimal stopping policy operator defined on page 105.

Exercise 8.1.5. Show that T_σ is an order-preserving self-map on V for all $\sigma \in \Sigma$.

Given $v \in V$, a policy $\sigma \in \Sigma$ is called v-**greedy** if

$$\sigma(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} B(x, a, v) \quad \text{for all } x \in \mathsf{X}. \tag{8.13}$$

Since $\Gamma(x)$ is finite and nonempty at each $x \in \mathsf{X}$, at least one such policy exists. As with policy operators, the notion of greedy policies extends existing definitions from earlier chapters.

Exercise 8.1.6. Show that, for each $v \in V$, the set $\{T_\sigma v\}_{\sigma \in \Sigma} \subset V$ contains a least and greatest element (see §2.1.1.4 for the definitions).

Given (Γ, V, B), the operator T on V defined by

$$(Tv)(x) = \max_{a \in \Gamma(x)} B(x, a, v) \qquad (x \in \mathsf{X})$$

is called the **Bellman operator**. Analogously, we say that $v \in V$ satisfies the **Bellman equation** if (8.1) holds.

Example 8.1.9. For the Epstein–Zin RDP in (8.10), the Bellman operator is given by

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x,a)^\alpha + \beta \left[\sum_{x' \in \mathsf{X}} v(x')^\gamma P(x,a,x') \right]^{\alpha/\gamma} \right\}^{1/\alpha} \qquad (x \in \mathsf{X}).$$

Exercise 8.1.7. Given an RDP (Γ, V, B) with policy operators $\{T_\sigma\}$ and Bellman operator T, show that, for each $v \in V$,

(i) $Tv = \vee_\sigma T_\sigma v := \vee_{\sigma \in \Sigma} T_\sigma v$ and

(ii) σ is v-greedy if and only if $Tv = T_\sigma v$.

8.1.2.2 Stable RDPs

We call an RDP (Γ, V, B) **stable** if, for all $\sigma \in \Sigma$, the policy operator T_σ is globally stable on V. We will show that stability is sufficient for strong optimality results.

Example 8.1.10. The RDP generated by an MDP (Example 8.1.1) is stable, since $T_\sigma = r_\sigma + \beta P_\sigma$ is globally stable on $\mathbb{R}^{\mathsf{X}}$ for all $\sigma \in \Sigma$ (see Exercise 5.1.6 on page 132).

Example 8.1.11. The optimal stopping RDP we introduced in Example 8.1.3 is stable. Indeed, for each $\sigma \in \Sigma$, the policy operator T_σ is globally stable on $\mathbb{R}^{\mathsf{X}}$ by Proposition 4.1.1 on page 105.

In the last two examples the policy operators are contractions, with modulus of contraction given by the discount factor. In nontraditional settings, contractivity fails but stability can be established using other methods. The next proposition helps to illustrate.

Proposition 8.1.1. *The Epstein–Zin RDP (Γ, B, V) introduced in Example 8.1.7 is stable.*

Proof. Let (Γ, B, V) be as stated. Under this specification, V is all $v \in \mathbb{R}^{\mathsf{X}}$ with $v \gg 0$ and, for given $\sigma \in \Sigma$, the corresponding policy operator T_σ takes the form

$$(T_\sigma v)(x) = \left\{ r(x, \sigma(x)) + \beta \left(\sum_{x'} v(x')^\gamma P(x, \sigma(x), x') \right)^{\alpha/\gamma} \right\}^{1/\alpha}. \tag{8.14}$$

The operator T_σ is a special case of the generic Epstein–Zin Koopmans operator displayed in (7.12) on page 219. By assumption $r \in V$, P is irreducible and $0 < \beta < 1$. Proposition 7.2.3 on page 220 tells us that, under these conditions, the operator (8.14) is globally stable on V. Hence (Γ, B, V) is a stable RDP. □

8.1.2.3 Lifetime Value for RDPs

One implication of stability for a given RDP (Γ, V, B) is that each T_σ has a unique fixed point, which we denote by v_σ and call the **σ-value function**. By construction, v_σ obeys

$$v_\sigma(x) = B(x, \sigma(x), v_\sigma) \quad \text{for all } x \in \mathsf{X}. \tag{8.15}$$

We interpret v_σ as representing the lifetime value of following policy σ. The next two examples illustrate why this interpretation is natural.

Example 8.1.12. For the optimal stopping problem discussed in Chapter 4, the function v_σ that records the lifetime value of a policy σ from any given state is the unique fixed point of the optimal stopping policy operator T_σ. See §4.1.1.3.

Example 8.1.13. For the MDP model discussed in Chapter 5, lifetime value is given by $v_\sigma = (I - \beta P_\sigma)^{-1} r_\sigma$. As discussed in Exercise 5.1.6, v_σ is the unique fixed point of the MDP policy operator $T_\sigma = r_\sigma + \beta P_\sigma$.

Example 8.1.14. For the MDP model with state-dependent discounting introduced in Chapter 6, Exercise 6.3.4 shows that the lifetime value of following policy σ is the unique fixed point of the policy operator T_σ defined in (6.26) on page 196.

Notice the connection with the generic nonlinear recursive preference specification discussed in §7.2.4.1, where a Koopmans operator on a given set V is an order-preserving and globally stable self-map K on V, and lifetime value associated with K is defined as the unique fixed point of K. The policy operators of a stable RDP fit this scenario: every policy operator T_σ is order-preserving and globally stable, and, analogously, the fixed point of each such operator is identified with lifetime value.

8.1.3 Optimality

In this section we present optimality theory for RDPs.

8.1.3.1 Defining Optimality

For a stable RDP, we define the **value function** v^* via $v^* := \vee_\sigma v_\sigma$. More explicitly,

$$v^*(x) := \max_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{8.16}$$

A policy σ is called **optimal** if $v_\sigma = v^*$; that is, if

$$v_{\sigma^*}(x) \geqslant v_\sigma(x) \quad \text{for all } \sigma \in \Sigma \text{ and all } x \in \mathsf{X}.$$

Both of these definitions generalize the definitions we used for MDPs and optimal stopping.

Bellman's principle of optimality is said to hold if

$$\sigma \in \Sigma \text{ is optimal} \quad \iff \quad \sigma \text{ is } v^*\text{-greedy}.$$

Exercise 8.1.8. Show that, for a given RDP (Γ, V, B) and fixed $v \in V$, the Bellman operator T obeys

$$(T^k v)(x) = \max_{a \in \Gamma(x)} B(x, a, T^{k-1} v) \tag{8.17}$$

for all $k \in \mathbb{Z}_+$ and all $x \in \mathsf{X}$. Show, in addition, that for any policy $\sigma \in \Sigma$, the policy operator T_σ obeys

$$(T_\sigma^k v)(x) = B(x, \sigma(x), T_\sigma^{k-1} v) \tag{8.18}$$

for all $k \in \mathbb{Z}_+$ and all $x \in \mathsf{X}$.

8.1.3.2 Algorithms for RDPs

The three major algorithms for MDPs (value function iteration (VFI), Howard policy iteration (HPI), and optimistic policy iteration (OPI)) extend naturally to stable RDPs. Here we state these algorithms in the RDP setting. We discuss convergence in §8.1.3.3.

Let (Γ, V, B) be a stable RDP with feasible policy set Σ, Bellman operator T and policy operator T_σ for each $\sigma \in \Sigma$. In this setting, the HPI algorithm is essentially identical to the one given for MDPs in §5.1.4.2, except that v_σ is calculated as the fixed point of T_σ, rather than taking the specific form $(I - \beta P_\sigma)^{-1} r_\sigma$. The details are in Algorithm 8.

Algorithm 8: Howard policy iteration for RDPs

```
input σ_0 ∈ Σ, an initial guess of σ*
k ← 0
ε ← 1
while ε > 0 do
    v_k ← the fixed point of T_{σ_k}
    σ_{k+1} ← a v_k greedy policy
    ε ← ‖σ_k − σ_{k+1}‖_∞
    k ← k + 1
end
return σ_k
```

The VFI and OPI algorithms for RDP look identical to those given for MDPs, as stated in §5.1.4.1 and §5.1.4.4 respectively, after replacing the specific MPD value space $\mathbb{R}^{\mathsf{X}}$ with the abstract RDP value space V. For convenience we state the RDP OPI routine in Algorithm 9.

One way to understand the connection between Algorithm 8 and Algorithm 9 is as follows: While the HPI algorithm (Algorithm 8) is theoretically important, one could

Algorithm 9: Optimistic policy iteration for RDPs

```
input σ ∈ Σ and set v_0 ← v_σ
input τ, a tolerance level for error
input m ∈ ℕ, a step size
k ← 0 and ε ← τ + 1
while ε > τ do
    σ_k ← a v_k-greedy policy
    v_{k+1} ← T^m_{σ_k} v_k
    ε ← ||v_k − v_{k+1}||
    k ← k + 1
end
return v_k, σ_k
```

make the argument that this set of instructions is not a valid algorithm, since it is not always obvious how to implement the instruction "$v_k \leftarrow$ the fixed point of T_{σ_k}". The only pertinent information we have is that T_{σ_k} is globally stable, so an approximation of the fixed point can be calculated by iterating with T_{σ_k}. This leads to Algorithm 9.

Notice that in Algorithm 9 we require that $v_0 = v_\sigma$ for some $\sigma \in \Sigma$. We cannot guarantee convergence of OPI from any $v \in V$, since we only have the weak assumption that every policy operator is globally stable. For practical purposes, it is usually straightforward to initialize OPI (or VFI) with $v_0 = v_\sigma$ for some simple choice of σ.

8.1.3.3 Optimality Results

We can now state our main optimality result for RDPs.

Theorem 8.1.2. *For every stable RDP, the following statements are true:*

(i) *the value function v^* is the unique solution to the Bellman equation in V,*

(ii) *HPI converges to v^* in finitely many steps,*

(iii) *the sequence (v_k) generated by OPI converges to v^* as $k \to \infty$.*

(iv) *Bellman's principle of optimality holds, and*

(v) *at least one optimal policy exists.*

The proof of Theorem 8.1.2 is deferred to §8.3.

Example 8.1.15. The optimality results for optimal stopping problems we presented in Chapter 4 are a special case of Theorem 8.1.2, since each policy operator is globally stable on $V := \mathbb{R}^{\mathsf{X}}$ (Proposition 4.1.1).

Example 8.1.16. The optimality results for MDPs we presented in Chapter 5 are a special case of Theorem 8.1.2, since each policy operator is globally stable on $V := \mathbb{R}^{\mathsf{X}}$ (see Exercise 5.1.6 on page 132).

Example 8.1.17. We saw in §8.1.2.2 that, under the stated assumptions, the Epstein–Zin RDP is stable. Hence all of the optimality results in Theorem 8.1.2 hold. (Later, in §8.4.4, we return to Epstein–Zin preferences and provide an alternative treatment.)

8.1.3.4 Comments on the Optimality Theorem

In terms of applications, Theorem 8.1.2 is the most important optimality result in this book. In this section we briefly comment on the assumptions behind Theorem 8.1.2 and how they relate to other theorems in related texts.

In standard treatments of dynamic programming, optimality theory for discounted dynamic programs is built around contractivity (see, e.g., Puterman (2005) or Stokey and Lucas (1989), Section 4.2). In particular, assumptions are constructed so that the policy operators and Bellman operator are all contraction mappings.

While such assumptions are sufficient for Theorem 8.1.2 (since contractivity of the policy operators implies stability of the RDP), they are certainly not necessary. There are many different ways to prove global stability, such as the monotonicity-based methods discussed in §7.1.2, or the spectral methods in §6.3.1.2. These alternatives will prove useful in settings where contractivity fails, as we shall see in §8.2.

Another point worth noting about the conditions in Theorem 8.1.2 is that no assumptions are placed on the Bellman operator. Rather, one only needs to check stability of the policy operators. This is advantageous because, unlike the Bellman operator, the policy operators do not involve maximization. Maximization adds complications to stability proofs.

8.1.3.5 Further Applications

In Examples 8.1.15–8.1.17 we noted that MDPs, optimal stopping problems and certain Epstein–Zin RDPs models are stable, and hence all of the optimality results in Theorem 8.1.2 hold. In §8.2, to simplify the process of verifying stability (and hence the optimality), we describe several classes of RDPs that exhibit stability and then show how specific applications fit into these classes.

8.1.3.6 Value Continuity

In general, one hopes that a v-greedy policy will be approximately optimal when v is close to v^*. Here we provide a condition on RDPs under which this hope is realized. (Readers focused on applications can skip to §8.2 without loss of continuity.)

Given an RDP (Γ, V, B) with state space X and action space A, we call the value aggregator B **value-continuous** if, for any $v \in V$ and sequence $(v_k)_{k \geqslant 1}$ in V, we have

$$v_k \to v \quad \implies \quad B(x, a, v_k) \to B(x, a, v)$$

as $k \to \infty$, for all $(x, a) \in \mathsf{G}$. (Here $v_k \to v$ means pointwise or norm convergence, which are equivalent by Exercise 1.2.7 on page 15.)

Value-continuity is satisfied by all applications considered in this text. For example, for the RDP generated by an MDP (Example 8.1.1), the deviation $|B(x, a, v_k) - B(x, a, v)|$ is dominated by $\beta \|v_k - v\|_\infty$ for all $(x, a) \in \mathsf{G}$. Hence value-continuity holds.

Now suppose that the RDP is stable, with value function v^*. The next results imply that, under value-continuity, if $v_k \to v^*$, then lifetime value of a v_k-greedy policy converges to the value of an optimal policy.

Proposition 8.1.3. *Let $(v_k)_{k \geqslant 1}$ be a sequence in V such that $v_k \to v^*$ as $k \to \infty$, and let σ_k be v_k-greedy for all k. If B is value-continuous, then $v_{\sigma_k} \to v^*$.*

Exercise 8.1.9. Prove Proposition 8.1.3.

8.2 Types of RDPs

In §8.1 we showed that RDPs have strong optimality properties whenever they are stable. But what conditions are sufficient for this property? We start with a relatively strict condition based on contractivity and then progress to models that fail to be contractive.

8.2.1 Contracting RDPs

In this section we study RDPs with strong contraction properties. Many traditional dynamic programs fit into this framework.

8.2.1.1 Definition and Examples

Let $\mathscr{R} = (\Gamma, V, B)$ be an RDP with state space X, action space A, and feasible state-action pair set G. We call $\mathscr{R}$ **contracting** if V there exists a $\beta \in [0, 1)$ such that

$$|B(x, a, v) - B(x, a, w)| \leqslant \beta \|v - w\|_\infty \quad \text{for all } (x, a) \in \mathsf{G} \text{ and } v, w \in V. \tag{8.20}$$

In line with the terminology for contraction maps, we call β the **modulus of contraction** for $\mathscr{R}$ when (8.20) holds.

Example 8.2.1. The optimal stopping RDP from Example 8.1.3 is contracting with modulus β, since, for B in (8.6), an application of the triangle inequality gives

$$|B(x, a, v) - B(x, a, w)| = (1 - a)\beta \left| \sum_{x' \in \mathsf{X}} [v(x') - w(x')] P(x, x') \right| \leqslant \beta \|v - w\|_\infty.$$

Exercise 8.2.1. Show that any MDP is a contracting RDP.

Exercise 8.2.2. Show that every contracting RDP is value-continuous.

Proposition 8.2.1. *If $\mathscr{R}$ is contracting with modulus β, then T and $\{T_\sigma\}_{\sigma \in \Sigma}$ are all contractions of modulus β on V under the norm $\|\cdot\|_\infty$.*

Proof. Let $\mathscr{R} = (\Gamma, V, B)$ be contracting with modulus β. Fix $\sigma \in \Sigma$ and let v and w be elements of V. By (8.20) we have

$$|(T_\sigma v)(x) - (T_\sigma w)(x)| = |B(x, \sigma(x), v) - B(x, \sigma(x), w)| \leqslant \beta \|v - w\|_\infty \tag{8.21}$$

for every $x \in \mathsf{X}$. Maximizing over x proves that T_σ is a contraction of modulus β with respect to the supremum norm.

Regarding contractivity of T, fix $v, w \in V$ and observe that, by the max-inequality lemma (page 31), we have

$$|(Tv)(x) - (Tw)(x)| = \left| \max_{\sigma \in \Sigma} (T_\sigma v)(x) - \max_{\sigma \in \Sigma} (T_\sigma w)(x) \right| \leqslant \max_{\sigma \in \Sigma} |(T_\sigma v)(x) - (T_\sigma w)(x)|.$$

Applying (8.21) and then maximizing the left-hand side over $x \in \mathsf{X}$ gives $\|Tv - Tw\|_\infty \leqslant \beta \|v - w\|_\infty$, as required. □

The following corollary to Proposition 8.2.1 is immediate from Banach's contraction mapping theorem.

Corollary 8.2.2. *If $\mathscr{R}$ is contracting and V is closed in $\mathbb{R}^{\mathsf{X}}$, then $\mathscr{R}$ is stable and hence all of the optimality and convergence results in Theorem 8.1.2 apply.*

8.2.1.2 Error Bounds

Corollary 8.2.2 tells us that contracting RDPs are stable and, as a result, the sequence of functions in V generated by VFI (Algorithm 9 with $m = 1$) converges to v^*. However this result is asymptotic and conditions on $v_0 = v_\sigma$ for some $\sigma \in \Sigma$. We can improve this result in the current setting by leveraging the contraction property:

Proposition 8.2.3. *Let (Γ, V, B) be a contracting RDP with modulus of contraction β and Bellman operator T. Fix $v \in V$ and consider the sequence in V defined by $v_k := T^k v$. If $\sigma \in \Sigma$ is v_k-greedy, then*

$$\|v^* - v_\sigma\|_\infty \leqslant \frac{2\beta}{1-\beta}\|v_k - v_{k-1}\|_\infty \quad \textit{for all } k \in \mathbb{N}. \tag{8.22}$$

Since the VFI algorithm terminates when $\|v_k - v_{k-1}\|_\infty$ falls below a given tolerance, the result in (8.22) directly provides a quantitative bound on the performance of the policy returned by VFI.

Proof of Proposition 8.2.3. Let (Γ, V, B) and v be as stated and let v^* be the value function. Note that

$$\|v^* - v_\sigma\|_\infty \leqslant \|v^* - v_k\|_\infty + \|v_k - v_\sigma\|_\infty. \tag{8.23}$$

To bound the first term on the right-hand side of (8.23), we use the fact that v^* is a fixed point of T, obtaining

$$\|v^* - v_k\|_\infty \leqslant \|v^* - Tv_k\|_\infty + \|Tv_k - v_k\|_\infty \leqslant \beta\|v^* - v_k\|_\infty + \beta\|v_k - v_{k-1}\|_\infty.$$

Hence

$$\|v^* - v_k\|_\infty \leqslant \frac{\beta}{1-\beta}\|v_k - v_{k-1}\|_\infty. \tag{8.24}$$

Now consider the second term on the right-hand side of (8.23). Since σ is v_k-greedy, we have $Tv_k = T_\sigma v_k$, and

$$\|v_k - v_\sigma\|_\infty \leqslant \|v_k - Tv_k\|_\infty + \|Tv_k - v_\sigma\|_\infty \leqslant \|Tv_{k-1} - Tv_k\|_\infty + \|T_\sigma v_k - T_\sigma v_\sigma\|_\infty.$$

$$\therefore \quad \|v_k - v_\sigma\|_\infty \leqslant \beta\|v_{k-1} - v_k\|_\infty + \beta\|v_k - v_\sigma\|_\infty.$$

$$\therefore \quad \|v_k - v_\sigma\|_\infty \leqslant \frac{\beta}{1-\beta}\|v_k - v_{k-1}\|_\infty. \tag{8.25}$$

Together, (8.23), (8.24), and (8.25) give us (8.22). □

8.2.1.3 A Blackwell-Type Condition for Contractivity

Next we state a useful condition for contractivity that is related to Blackwell's sufficient condition discussed in §2.1.2.2. We say that RDP (Γ, V, B) satisfies **Blackwell's condition** if $v \in V$ implies $v + \lambda := v + \lambda \mathbb{1}$ is in V for every $\lambda \geqslant 0$ and, in addition, there exists a $\beta \in [0, 1)$ such that

$$B(x, a, v + \lambda) \leqslant B(x, a, v) + \beta\lambda \qquad \text{for all } \lambda \in \mathbb{R}_+ \text{ and all } (x, a) \in \mathsf{G}.$$

Exercise 8.2.3. Prove the following: If $\mathscr{R} = (\Gamma, B, V)$ is an RDP that satisfies Blackwell's condition, then $\mathscr{R}$ is contracting with modulus β.

Exercise 8.2.4. Prove that the RDP for the state-dependent discounting model in Example 8.1.5 is contracting on $V = \mathbb{R}^{\mathsf{X}}$ whenever there exists a $b < 1$ with $\beta(x, a, x') \leqslant b$ for all $(x, a) \in \mathsf{G}$ and $x' \in \mathsf{X}$.

Exercise 8.2.5. Prove that the discrete optimal savings model from §5.2.2 satisfies Blackwell's condition.

8.2.1.4 Application: Job Search with Quantile Preferences

Consider the job search problem with correlated wage draws first investigated in §3.3.1. With finite wage offer set W, wage offer process generated by $P \in \mathbb{M}(\mathsf{W})$ and $\beta \in (0, 1)$, we can frame this as an RDP (Γ, V, B) with $V = \mathbb{R}^{\mathsf{W}}$, $\Gamma(w) = \{0, 1\}$ for $w \in \mathsf{W}$ and

$$B(w, a, v) := a\frac{w}{1 - \beta} + (1 - a)[c + \beta(Pv)(w)].$$

Since the model just described is an optimal stopping problem, Example 8.2.1 tells us that (V, Γ, B) is contracting.

Now consider the following modification, where Γ and V are as before but B is replaced by

$$B_\tau(w, a, v) := a\frac{w}{1 - \beta} + (1 - a)[c + \beta(R_\tau v)(w)],$$

where $\tau \in [0, 1]$ and R_τ is the quantile-based certainty equivalent operator described in Exercise 7.2.12 (page 226).

Figure 8.1: The reservation wage as a function of τ

Exercise 8.2.6. Prove that (V, Γ, B_τ) is a contracting RDP.

Figure 8.1 shows the reservation wage for a range of τ values, computed using optimistic policy iteration (and taking the smallest $w \in \mathsf{W}$ such that $\sigma^*(w) = 1$). The stationary distribution of P is also shown in the figure, tilted 90 degrees.

The parameters and the code for applying T_σ and evaluating greedy functions is shown in Listing 26. That listing includes the quantile operator R_τ, which is implemented in Listing 25. (Quantiles of discrete random variables can also be computed using functionality contained in `Distributions.jl`.)

The main message of Figure 8.1 is that the reservation wage rises in τ. In essence, higher τ focuses the attention of the worker on the right tail of the distribution of continuation values. This encourages the worker to take on more risk, which leads to a higher reservation wage (i.e., reluctance to accept a given current offer).

8.2.1.5 Application: Optimal Default

In this section we consider a small open economy that borrows in international financial markets in order to smooth consumption and has the option to default. We show that the model is a contractive RDP.

Income $(Y_t)_{t \geqslant 0}$ is exogenous and Q-Markov on finite set Y. A representative house-

```
"Compute the τ-th quantile of v(X) when X ~ ϕ and v = sort(v)."
function quantile(τ, v, ϕ)
    for (i, v_value) in enumerate(v)
        p = sum(ϕ[1:i])  # sum all ϕ[j] s.t. v[j] ≤ v_value
        if p ≥ τ         # exit and return v_value if prob ≥ τ
            return v_value
        end
    end

end

"For each i, compute the τ-th quantile of v(Y) when Y ~ P(i, ·)"
function R(τ, v, P)
    return [quantile(τ, v, P[i, :]) for i in eachindex(v)]
end
```

Listing 25: Conditional quantile operator (`quantile_function.jl`)

hold faces budget constraint

$$C_t = Y_t + b_t - qb_{t+1} \qquad (t \geqslant 0),$$

where C_t is consumption at time t, q is the price at time t of a risk-free claim on one unit of time $t+1$ consumption; q is determined outside the model, say international markets; b_t measures foreign lending. Purchasing a claim on b_{t+1} units of time $t+1$ consumption costs qb_{t+1}. Purchasing bond with *negative* face value b_{t+1} pays qb_{t+1} in current consumption goods and promises to deliver b_{t+1} next period.

Trading bonds is managed by a benevolent government that wants to maximize household utility. Households discount future utility at rate $\beta \in (0,1)$ and current consumption C_t generates current utility $u(C_t)$. The government faces borrowing constraint $b_t \geqslant -m$ where $m \geqslant 0$. The government maximizes expected discounted utility for the households.

The government can default on foreign loans. In this case, output available for consumption drops from Y_t to $h(Y_t)$, where h is a function satisfying $h(y) < y$ for all y. After a country defaults, it temporarily loses access to the international credit market.

At the end of each period during which the country is in default, it regains access to international credit markets with probability $\theta \in (0,1)$. With probability $1-\theta$ it remains in financial autarky. When a country regains access to foreign borrowing, its

```
using QuantEcon
include("quantile_function.jl")

"Creates an instance of the job search model."
function create_markov_js_model(;
        n=100,       # wage grid size
        ρ=0.9,       # wage persistence
        ν=0.2,       # wage volatility
        β=0.98,      # discount factor
        c=1.0,       # unemployment compensation
        τ=0.5        # quantile parameter
    )
    mc = tauchen(n, ρ, ν)
    w_vals, P = exp.(mc.state_values), mc.p
    return (; n, w_vals, P, β, c, τ)
end

"""
The policy operator

    (T_σ v)(w) = σ(w) (w / (1-β)) + (1 - σ(w))(c + β (R_τ v)(w))

"""
function T_σ(v, σ, model)
    (; n, w_vals, P, β, c, τ) = model
    h = c .+ β * R(τ, v, P)
    e = w_vals ./ (1 - β)
    return σ .* e + (1 .- σ) .* h
end

" Get a v-greedy policy."
function get_greedy(v, model)
    (; n, w_vals, P, β, c, τ) = model
    σ = w_vals / (1 - β) .≥ c .+ β * R(τ, v, P)
    return σ
end
```

Listing 26: Job search with quantile operator (`quantile_js.jl`)

debt is reset to zero.

We can cast this as an RDP by considering the value of each state and action. We set the state space X to be the set of all (y, b, d) in $\mathsf{Y} \times \mathsf{B} \times \{0, 1\}$, where B is a finite subset of $[-m, \infty)$ indicating possible choices for bond holdings b_t and d is a binary variable indicating whether the country is in default ($d = 0$ means not in default and $d = 1$ means in default).

The value space V is all of $\mathbb{R}^{\mathsf{X}}$. The action space is $(b_a, d_a) \in \mathsf{B} \times \{0, 1\}$ indicating choices for bond holdings and default. The feasible correspondence specifies feasible (b_a, d_a) at given state (y, b, d) and is given by

$$\Gamma(y, b, d) = \begin{cases} \mathsf{B} \times \{0, 1\} & \text{if } d = 0 \text{ and} \\ \{0\} \times \{1\} & \text{if } d = 1. \end{cases}$$

In other words, if $d = 0$, so the country is not in default, the government can choose any $b_a \in \mathsf{B}$ and also any $d_a \in \{0, 1\}$ (i.e., default or not default). If $d = 1$, however, the government has no choices. We represent this situation by $b_a = 0$ and $d_a = 1$.

The value aggregator takes the form

$$B((y, b, d), (b_a, d_a), v) = \text{value in state } (y, b, d) \text{ under action } (b_a, d_a).$$

To specify it we decompose the problem across cases for d and d_a. First consider the case where $d = 0$ (not currently in default) and $d_a = 0$ (the government chooses not to default). For this case $y + b - qb_a$ is current consumption, so we set

$$B((y, b, 0), (b_a, 0), v) = u(y + b - qb_a) + \beta \sum_{y'} v(y', b_a, 0) Q(y, y') \tag{8.26}$$

Now consider the case where $d = 0$ and $d_a = 1$, so the government chooses to default. Then current consumption is $h(y)$ and we set

$$B((y, b, 0), (b_a, 1), v) = u(h(y)) + \beta \left[\theta \sum_{y'} v(y', 0, 0) Q(y, y') + (1 - \theta) \sum_{y'} v(y', 0, 1) Q(y, y') \right]. \tag{8.27}$$

The term $\sum_{y'} v(y', 0, 0) Q(y, y')$ is the expected value next period when the country is readmitted to international financial markets (with $b' = 0$ and $d' = 0$), while the term $\sum_{y'} v(y', 0, 1) Q(y, y')$ is the expected value next period when default continues (with $b' = 0$ and $d' = 1$).

Since $B((y, b, 1), (b_a, 0), v)$ is not feasible (a defaulted country cannot itself directly

choose to reenter financial markets), the only other possibility is $B((y, b, 1), (b_a, 1), v)$, which is the expected value when the country remains in default. But this is the same as $B((y, b, 0), (b_a, 1), v)$ specified above: the value for a country that stays in default is the same as that for a country that newly enters default.

Exercise 8.2.7. By working through cases (8.26)–(8.27) for the value aggregator B, show that the model described above is a contractive RDP.

8.2.2 Risk-Sensitive RDPs

In §7.2.2 we introduced risk-sensitive preferences and discussed a recursive utility problem. Now we embed risk-sensitive preferences into a dynamic program and apply the preceding optimality results compute optimal policies.

8.2.2.1 Optimality Results

We begin with a small extension of the risk-sensitive preference RDP in Example 8.1.6. Our starting point is an MDP (Γ, β, r, P) with state space X and action space A. As usual, $\beta \in (0, 1)$. Let $V = \mathbb{R}^{\mathsf{X}}$ and let R be a certainty equivalent operator on V (see §7.2.4.2). For $(x, a) \in \mathsf{G}$ and $v \in V$, let

$$B(x, a, v) := r(x, a) + \beta R(v, P(x, a, \cdot))$$

Exercise 8.2.8. With Γ, V and B as above, verify that (Γ, V, B) forms an RDP.

Recall from §7.2.5.1 that R is called subadditive when $R(v+\lambda\mathbb{1}, \varphi) \leqslant R(v, \varphi)+\lambda$ for all $v \in V$, $\varphi \in \mathscr{D}(\mathsf{X})$ and $\lambda \in \mathbb{R}_+$. The next result shows that subadditivity is sufficient for all of our core optimality results.

Proposition 8.2.4. *If R is is subadditive, then (Γ, V, B) is a contracting RDP.*

Proof. We show that (Γ, V, B) obeys Blackwell's condition. Fix $v \in V$, $(x, a) \in \mathsf{G}$, and $\lambda \geqslant 0$. Applying subadditivity gives

$$B(x, a, v + \lambda\mathbb{1}) = r(x, a) + \beta R(v + \lambda\mathbb{1}, P(x, a, \cdot)) \leqslant r(x, a) + \beta R(v, P(x, a, \cdot)) + \beta\lambda.$$

The right-hand side equals $B(x, a, v) + \beta\lambda$, so Blackwell's condition holds. The claim in Proposition 8.2.4 now follows from Exercise 8.2.3. □

From Corollary 8.2.2 and Proposition 8.2.4, we see that any risk-sensitive RDP with subadditive certainty equivalent operator is stable. Hence all of the fundamental optimality results hold.

8.2.2.2 An Example

Let's consider a job search problem where future wage outcomes are evaluated via entropic risk-adjusted expectations. The associated Bellman operator is

$$(Tv)(w) = \max\left\{\frac{w}{1-\beta},\, c + \frac{\beta}{\theta}\ln\left[\sum_{w'}\exp(\theta v(w'))P(w,w')\right]\right\} \qquad (w \in \mathsf{W}).$$

Here θ is a nonzero parameter and other details are as in §3.3.1. We can represent the problem as an RDP with state space W, action space $\mathsf{A} = \{0, 1\}$, feasible correspondence $\Gamma(w) = \mathsf{A}$, value space $V := \mathbb{R}^{\mathsf{W}}$, and value aggregator

$$B(w,a,v) = a\frac{w}{1-\beta} + (1-a)\left\{c + \frac{\beta}{\theta}\ln\left[\sum_{w'}\exp(\theta v(w'))P(w,w')\right]\right\}.$$

If $\theta < 0$, then the agent is risk-averse with respect to the gamble associated with continuing and waiting for new wage draws. If $\theta > 0$ then the agent is risk-loving with respect to such gambles. For $\theta \approx 0$, the agent is close to risk-neutral.

Figure 8.2 shows how the continuation value, value function and optimal decision vary with θ. Apart from θ, parameters are identical to those in Listing 10 on page 96. Indeed, for θ close to zero, as in the middle sub-figure of Figure 8.2, we see that the value function and reservation wage are almost identical to those from the risk-neutral model in Figure 3.7 on page 97.

As expected, a negative value of θ tends to reduce the continuation value and hence the reservation wage, since the agent's dislike of risk encourages early acceptance of an offer. For positive values of θ the reverse is true, as seen in the bottom sub-figure.

Exercise 8.2.9. Replicate Figure 8.2. The simplest method is to modify the code in Listing 10 and use value function iteration.

8.2.3 Eventually Contracting RDPs

Many RDPs are not contracting. There is no single method for handling all types of non-contractive RDPs, so we introduce alternative techniques over the next few

Figure 8.2: Job search with risk-sensitive preferences

sections. The first such technique, treated in this section, handles RDPs that contract "eventually," even though they may fail to contract in one step. We show that these eventually contracting RDPs are stable, so all of the fundamental optimality results apply.

One application for these results is the MDP model with state-dependent discounting treated in Chapter 6. This section contains a proof of the main optimality result in that chapter (Proposition 6.3.4 on page 196).

8.2.3.1 Definition and Properties

Let $\mathscr{R} = (\Gamma, V, B)$ be an RDP. We call $\mathscr{R}$ **eventually contracting** if there an $L \in \mathbb{L}(\mathsf{X})$ such that $r(L) < 1$ and

$$|B(x,a,v) - B(x,a,w)| \leqslant \sum_{x'} |v(x') - w(x')| L(x,x') \quad \text{for all } (x,a) \in \mathsf{G}. \tag{8.28}$$

Exercise 8.2.10. Prove the following: If $\mathscr{R}$ is eventually contracting and, if in addition, $L(x,x') = \beta Q(x,x')$ for a $\beta \in (0,1)$ and a $Q \in \mathbb{M}(\mathsf{X})$, then $\mathscr{R}$ is contracting (in the sense of §8.2.1.1).

Proposition 8.2.5. *Let $\mathscr{R} = (\Gamma, V, B)$ be an RDP. If $\mathscr{R}$ is eventually contracting and V is closed in $\mathbb{R}^{\mathsf{X}}$, then $\mathscr{R}$ is also stable and hence all of the optimality and convergence results in Theorem 8.1.2 apply.*

Proof. Let $\mathscr{R}$ be as stated and fix $\sigma \in \Sigma$. Let T_σ be the associated policy operator. We need to show that T_σ is globally stable on V. To this end, we fix $v, w \in V$ and use (8.28) to obtain

$$\begin{aligned} |(T_\sigma v)(x) - (T_\sigma w)(x)| &= |B(x,\sigma(x),v) - B(x,\sigma(x),w)| \\ &\leqslant \sum_{x'} |v(x') - w(x')| \; L(x,x'). \end{aligned}$$

Since $L \geqslant 0$ and $r(L) < 1$, Proposition 6.3.2 on page 194 implies that T_σ is eventually contracting on V. Since V is closed in $\mathbb{R}^{\mathsf{X}}$, it follows that T_σ is globally stable (Theorem 6.3.1, page 193). □

Exercise 8.2.11. Prove: If $\mathscr{R} = (\Gamma, V, B)$ is eventually contracting, V is closed in $\mathbb{R}^{\mathsf{X}}$ and T is the Bellman operator generated by $\mathscr{R}$, then T is globally stable on V.

8.2.3.2 State-Dependent Discounting

With the results stated above, we can complete the proof of Proposition 6.3.4 on page 196, which made claims about optimality properties for MDPs with state-dependent discounting.

In Example 8.1.5 we embedded the MDP model with state-dependent discounting into the RDP framework. As in that example, we begin with an MDP (Γ, β, r, P) with state space X and then replace the constant β with a function β from $\mathsf{G} \times \mathsf{X}$ to $\mathbb{R}_+$, so that the aggregator takes the form

$$B(x, a, v) = r(x, a) + \sum_{x'} v(x')\beta(x, a, x')P(x, a, x'). \tag{8.29}$$

The value space V is all of $\mathbb{R}^{\mathsf{X}}$. The tuple $\mathscr{R} := (\Gamma, V, B)$ forms an RDP (Exercise 8.2.4).

Proposition 8.2.6. *If there exists an $L \in \mathbb{L}(\mathsf{X})$ such that $r(L) < 1$ and*

$$\beta(x, a, x')P(x, a, x') \leqslant L(x, x') \tag{8.30}$$

for all $(x, a) \in \mathsf{G}$ and $x' \in \mathsf{X}$, then $\mathscr{R}$ is eventually contracting.

Proof. Let $\mathscr{R}$ and L be as stated and fix $(x, a) \in \mathsf{G}$. For given $v, w \in V$, applying the definition (8.29) and the triangle inequality, we have

$$\begin{aligned} |B(x, a, v) - B(x, a, w)| &\leqslant \left| \sum_{x'} [v(x') - w(x')]\beta(x, a, x')P(x, a, x') \right| \\ &\leqslant \sum_{x'} |v(x') - w(x')|\beta(x, a, x')P(x, a, x') \end{aligned}$$

Now (8.30) implies that $\mathscr{R}$ is eventually contracting, as claimed. □

We can now complete the

Proof of Proposition 6.3.4. Let $\mathscr{R} := (\Gamma, V, B)$ be as described above and assume the conditions of Proposition 6.3.4. It follows that the conditions of Proposition 8.2.6 all hold. Hence $\mathscr{R}$ is eventually contracting. Since V is closed, Proposition 8.2.5 implies that $\mathscr{R}$ is a stable RDP. The claims in Proposition 6.3.4 now follow from Theorem 8.1.2. □

8.2.4 Convex and Concave RDPs

We have seen that RDPs have excellent optimality properties when all policy operators are globally stable self-maps on value space. So far we have looked at conditions for stability based on contractions (§8.2.1) and eventual contractions (§8.2.3). But sometimes both of these approaches fail and we need alternative conditions. In this section we explore such conditions by turning to Du's theorem (page 208). It provides an alternative approach to global stability of policy operators. Du's theorem is well suited to the task, since it leverages the fact that all policy operators are order-preserving maps.

8.2.4.1 Convex RDPs

Let $\mathscr{R} = (\Gamma, V, B)$ be an RDP. We call $\mathscr{R}$ **convex** if

(i) $V = [v_1, v_2]$ for some pair $v_1 \leqslant v_2$ in $\mathbb{R}^{\mathsf{X}}$,

(ii) for all $(x, a) \in \mathsf{G}$, $\lambda \in [0, 1]$ and v, w in V, we have

$$B(x, a, \lambda v + (1 - \lambda)w) \leqslant \lambda B(x, a, v) + (1 - \lambda)B(x, a, w) \quad \text{and,} \tag{8.31}$$

(iii) there exists a $\delta > 0$ such that

$$B(x, a, v_2) \leqslant v_2(x) - \delta[v_2(x) - v_1(x)] \text{ for all } (x, a) \in \mathsf{G}. \tag{8.32}$$

The following condition is often easier to check than (8.32).

Exercise 8.2.12. Prove that (iii) holds whenever

$$B(x, a, v_2) < v_2(x) \text{ for all } (x, a) \in \mathsf{G}. \tag{8.33}$$

Convexity yields stability, as the next proposition shows.

Proposition 8.2.7. *If $\mathscr{R} = (\Gamma, V, B)$ is a convex RDP, then $\mathscr{R}$ is also stable and hence all of the optimality and convergence results in Theorem 8.1.2 apply.*

Proof. Fix $\sigma \in \Sigma$. By the monotonicity property of RDPs, T_σ is an order-preserving self-map on V. Since (8.31) holds, T_σ is also a convex operator on V. Moreover, $Tv_1 \geqslant v_1$ because $T \colon V \to V$ and, by (8.33), $T_\sigma v_2 \leqslant v_2 - \delta(v_2 - v_1)$. Hence Du's Theorem on page 208 applies and T_σ is globally stable on V. This shows that $\mathscr{R}$ is a stable RDP. □

8.2.4.2 Concave RDPs

Analogous to the convex case, we call $\mathscr{R}$ **concave** if

(i) $V = [v_1, v_2]$ for some pair $v_1 \leqslant v_2$ in $\mathbb{R}^{\mathsf{X}}$,

(ii) for all $(x, a) \in \mathsf{G}$, $\lambda \in [0, 1]$ and v, w in V, we have

$$B(x, a, \lambda v + (1 - \lambda)w) \geqslant \lambda B(x, a, v) + (1 - \lambda)B(x, a, w) \quad \text{and,} \tag{8.34}$$

(iii) there exists a $\delta > 0$ such that

$$B(x, a, v_1) \geqslant v_1(x) + \delta[v_2(x) - v_1(x)] \text{ for all } (x, a) \in \mathsf{G}. \tag{8.35}$$

The following condition is often easier to check than (8.35).

Exercise 8.2.13. Prove that (iii) holds whenever

$$B(x, a, v_1) > v_1(x) \text{ for all } (x, a) \in \mathsf{G}. \tag{8.36}$$

These conditions also yield stability, as the next proposition shows.

Proposition 8.2.8. *If $\mathscr{R} = (\Gamma, V, B)$ is a concave RDP, then $\mathscr{R}$ is also stable and hence all of the optimality and convergence results in Theorem 8.1.2 apply.*

Exercise 8.2.14. Verify Proposition 8.2.8 (modify the proof of Proposition 8.2.7).

As an application, consider the RDP $\mathscr{R} = (\Gamma, B, V)$ generated by an ordinary MDP (Γ, β, r, P), so that $V = \mathbb{R}^{\mathsf{X}}$ and $B(x, a, v) = r(x, a) + \beta \sum_{x'} v(x')P(x, a, x')$. In §8.2.1 we saw that $\mathscr{R}$ is a contracting RDP. In fact $\mathscr{R}$ is also convex *and* concave if we fix some $\varepsilon > 0$ and redefine V to

$$V := [v_1, v_2] \quad \text{where} \quad v_1 := \frac{r_1 - \varepsilon}{1 - \beta} \text{ and } v_2 := \frac{r_2 + \varepsilon}{1 - \beta}. \tag{8.37}$$

(The functions v_1 and v_2 are constant.) For example, (8.36) holds because, for given $(x, a) \in \mathsf{G}$,

$$B(x, a, v_1) = r(x, a) + \beta \sum_{x'} v_1(x')P(x, a, x') \geqslant r_1 + \beta \frac{r_1 - \varepsilon}{1 - \beta}.$$

The right-hand side can be re-written as $v_1 + \varepsilon$, so (8.36) is confirmed.

Exercise 8.2.15. Complete the proof that $\mathscr{R}$ is both concave and convex.

8.2.5 Application: Adversarial Agents

Some problems in economics, finance and artificial intelligence assume that decisions emerge from a two-person zero sum game in which the two agents' preferences are perfectly *mis*aligned. Some problems obtain decisions that are robust to misspecification of a Markov transition kernel by letting an adversarial agent perturb the decision maker's baseline specification of that kernel. This can lead to a modified MDP setting in which the Bellman equation takes a form

$$v(x) = \max_{a \in \Gamma(x)} \min_{d \in D(x)} \left\{ r(x,a,d) + \beta \sum_{x'} v(x') P(x,a,d,x') \right\} \qquad (x \in \mathsf{X}) \tag{8.38}$$

where, for each x, the choice perturbation $d \in D(x)$ is made by the adversary. The object P is a stochastic kernel, in the sense that $P(x,a,d,\cdot)$ is a distribution over X for each feasible (x,a,d). The understanding here is that when the decision maker chooses action $a \in \Gamma(x)$, she does so with the knowledge that the opponent will then choose $d \in D(x)$ to minimize her aggregate value. Bellman equation (8.38) describes a two-player zero-sum dynamic game.

For each $a \in \Gamma(x)$ we get an RDP $\mathscr{R} = (\Gamma, B, V)$ when we take $V = \mathbb{R}^{\mathsf{X}}$ and

$$B(x,a,v) = \min_{d \in D(x)} \left\{ r(x,a,d) + \beta \sum_{x'} v(x') P(x,a,d,x') \right\} \qquad ((x,a) \in \mathsf{G}).$$

Such structures arise **robust control** theory. Here a decision maker knows her reward function but distrusts her specification of the stochastic kernel P that describes the evolution of the state. Sometimes this distrust can be expressed by assuming that she knows that P belongs to some class $\mathscr{P}$ of stochastic kernels from $\mathsf{G} \times \mathsf{X}$ to X. Under some further assumptions, this can lead to modifying the aggregator above to become

$$B(x,a,v) = \inf_{P \in \mathscr{P}} \left\{ r(x,a) + \beta \sum_{x'} v(x') P(x,a,x') \right\} \tag{8.39}$$

for $(x,a) \in \mathsf{G}$. The decision maker can construct a policy that is robust to her distrust of the stochastic kernel by using this aggregator V.[1]

Let's analyze the robust control model with aggregator B in (8.39). Here, as usual, $\beta \in (0,1)$. We take V is as defined in (8.37) and set $\mathscr{R} = (\Gamma, V, B)$. The set $\mathscr{P}$ of

[1] Positing that the decision maker knows a nontrivial set of stochastic kernels is a way of modeling *uncertainty* as distinguished from *risks* that are described by known probability distributions. Here uncertainty means not knowing a probability distribution.

stochastic kernels is entirely arbitrary.

Proposition 8.2.9. *$\mathscr{R}$ is a concave RDP.*

The proof is established in a series of exercises.

Exercise 8.2.16. Let f and g be bounded functions from nonempty set D into $\mathbb{R}$. Prove that

$$\inf_{d \in D}(f(d) + g(d)) \geqslant \inf_{d \in D} f(d) + \inf_{d \in D} g(d).$$

Exercise 8.2.17. Check the following:

- $\mathscr{R}$ is an RDP.
- The concavity condition (8.34) holds (hint: apply Exercise 8.2.16).
- The ε bound in (8.36) holds.

We conclude from the discussion above that the robust control RDP is stable, and hence all of the fundamental optimality properties hold.

8.3 Abstract Theory

In §8.1 we introduced RDPs, stated their optimality properties and investigated applications that satisfy the optimality conditions. The main task remaining is to prove the core optimality results in Theorem 8.1.2.

Rather than proving this result directly, we now present a completely abstract version of a dynamic programming problem, focusing purely on policy operators. Doing so allows us to simplify proofs and further extend the reach of dynamic programming theory.

Once the proofs are done, we will return to RDPs and show they are a special case.

8.3.1 Abstract Dynamic Programs

In this section we formalize abstract dynamic programs and present fundamental optimality results. Since this section is quite technical, §8.3.1.1 starts the ball rolling with an informal overview.

8.3.1.1 Prelude

We saw in §8.1 that a stable RDP yields a set of feasible policies Σ and, for each $\sigma \in \Sigma$, a policy operator T_σ defined on the value space $V \subset \mathbb{R}^{\mathsf{X}}$. Notice that the dynamic programming problem is fully specified by the family of operators $\{T_\sigma\}_{\sigma\in\Sigma}$ and the space V that they act on. From this set of operators we obtain the set of lifetime values $\{v_\sigma\}_{\sigma\in\Sigma}$, with each v_σ uniquely identified as a fixed point of T_σ. These lifetime values define the value function v^* as the pointwise maximum $v^* = \vee_\sigma v_\sigma$. An optimal policy is then defined as a $\sigma \in \Sigma$ obeying $v_\sigma = v^*$.

To shed unnecessary structure before the main optimality proofs, a natural idea is to start directly with an abstract set of "policy operators" $\{T_\sigma\}$ acting on some set V. One can then define lifetime values and optimality as in the previous paragraph and start to investigate conditions on the family of operators $\{T_\sigma\}$ that lead to characterizations of optimality that we hope to obtain.

Although we seek to maximize generality, we need at least some structure on the operator family $\{T_\sigma\}$. At minimum, we need each operator T_σ to have a unique fixed point v_σ that describes the lifetime value of following a given policy. Another useful property of the family $\{T_\sigma\}$ of policy operators, which holds in the RDP case, is that, for each $v \in V$, there is a $\sigma \in \Sigma$ such that $T_\sigma v = \vee_{\tau\in\Sigma} T_\tau v$ pointwise on the state space—in fact every v-greedy policy has this property (see, e.g., Exercise 8.1.6 on page 240).

We use these ideas as our starting point. We take a family $\{T_\sigma\}$ of operators that are order-preserving, admit greedy policies and possess unique fixed points. We then show that, for such a family, we can define optimality and prove that abstract versions of the basic optimality results hold (the value function satisfies the Bellman equation, Bellman's principle of optimality holds, etc.).

8.3.1.2 Lattice Structure

In the discussion below, we use a lattice V as an abstract representation of the value space V from the RDP setting. A **lattice** is a partially ordered set that is closed under the taking of finite suprema and infima. A complete definition is given in §A.3.

For those who are unfamiliar with lattices and wish to proceed directly to the theory, it suffices to continue to view the value space V as a sublattice of $\mathbb{R}^{\mathsf{X}}$, where X is an arbitrary finite set, and to understand the abstract partial order $\preceq$ as the pointwise order $\leqslant$.

8.3.1.3 Set Up

An **abstract dynamic program (ADP)** is a pair $(V, \{T_\sigma\}_{\sigma\in\Sigma})$ such that $V = (V, \preceq)$ is a lattice and $\{T_\sigma\}_{\sigma\in\Sigma}$ is a finite family of self-maps on V such that

(i) for each $\sigma \in \Sigma$,

  (a) T_σ is an order-preserving self-map on V,

  (b) T_σ has a unique fixed point in V, denoted henceforth by v_σ,

  (c) $v \in V$ and $T_\sigma v \preceq v$ implies $v_\sigma \preceq v$,

  (d) $v \in V$ and $v \preceq T_\sigma v$ implies $v \preceq v_\sigma$, and

(ii) for all $v \in V$, the set $\{T_\sigma v\}_{\sigma\in\Sigma}$ contains a least and greatest element.

Below, we often write $\{T_\sigma\}$ in place of $\{T_\sigma\}_{\sigma\in\Sigma}$. Elements of Σ are called **policies** and elements of $\{T_\sigma\}$ are called **policy operators**. Analogous to the RDP setting, we call v_σ the **σ-value function**.

For ease of reference, we refer to (i)-(c) as **downward stability** and (i)-(d) as **upward stability**. Figure 8.3 gives a stylized illustration of an operator T_σ that satisfies all conditions in (i) when $V = [0, 1]$. For example, T_σ is downward stable because any v that is mapped down by T_σ (i.e., $T_\sigma v \leqslant v$) satisfies $v_\sigma \leqslant v$.

Property (ii) we call **existence of greedy policies** for reasons that are clarified below. We require existence of a least element of $\{T_\sigma v\}_{\sigma\in\Sigma}$ as well as a greatest element because we want to consider minimization as well as maximization problems (see §8.4.2). For applications that involve only maximization, it is enough that a greatest element exists for all v. Moreover, in this case, one can replace the assumption that V is a lattice with a weaker condition that V is an **upper semi-lattice**; that is $f \vee g \in V$ whenever $f, g \in V$.

For a given ADP and fixed $v \in V$, a policy $\hat\sigma$ in Σ is called **v-greedy** if $T_{\hat\sigma} v \succeq T_\sigma v$ for all $\sigma \in \Sigma$. Existence of a greatest element in property (ii) of the definition is equivalent to the statement that a v-greedy policy exists for every $v \in V$.

8.3.1.4 MDPs are ADPs

One reason we constructed ADPs is to generalize MDPs, as defined in Chapter 5. Let's now verify that every MDP is an ADP.

To see this, fix an MDP (Γ, β, r, P) on with state space X, let Σ be the associated feasible policies and let $\{T_\sigma\}_{\sigma\in\Sigma}$ be the policy operators. Our claim is that $(V, \{T_\sigma\}_{\sigma\in\Sigma})$ is an ADP when V is $\mathbb{R}^{\mathsf{X}}$ and $\preceq$ is the pointwise order $\leqslant$.

Figure 8.3: T_σ satisfies all conditions (one-dimensional setting)

Note that $V = \mathbb{R}^{\mathsf{X}}$ is a lattice under the pointwise order $\leqslant$. Part (i) of the ADP definition holds because, given $\sigma \in \Sigma$, each policy operator takes the form $T_\sigma v = r_\sigma + \beta P_\sigma v$ for some $r_\sigma \in \mathbb{R}^{\mathsf{X}}$ and $\beta \in (0,1)$, where P_σ is a Markov matrix. We saw in Chapter 5 that T_σ satisfies (i)-(a) and (i)-(b). As for upward stability, suppose that $v \in \mathbb{R}^{\mathsf{X}}$ and $v \leqslant T_\sigma v$, or, equivalently, $v \leqslant r_\sigma + \beta P_\sigma v$. A simple rearrangement transforms this to $v \leqslant (I - \beta P_\sigma)^{-1} r_\sigma$, or, equivalently $v \leqslant v_\sigma$. A similar argument shows that downward stability also holds.

It only remains to check part (ii) of the definition of ADPs on existence of least and greatest elements. To this end, we fix $v \in V$ and take $\hat{\sigma}$ to be v-greedy in the sense of Chapter 5, so that

$$\hat{\sigma}(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x,a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x,a,x') \right\} \quad \text{for all } x \in \mathsf{X} \tag{8.40}$$

(see §5.1.3.2). If σ is any other feasible policy, then

$$r(x, \hat{\sigma}(x)) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, \hat{\sigma}(x), x') \geqslant r(x, \sigma(x)) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, \sigma(x), x')$$

at all x. In operator form, this is $T_{\hat{\sigma}} v \geqslant T_\sigma v$. Since σ is an arbitrary greedy policy, we

have shown that $T_{\hat{\sigma}}\, v$ is the greatest element of $\{T_\sigma\, v\}_{\sigma\in\Sigma}$.

A similar argument replacing argmax with argmin in (8.40) shows that a least element also exists. This completes the proof that every MDP generates an ADP.

Exercise 8.3.1. Consider an MDP with state-dependent discounting on state space X (see §6.3.2). Let $V = \mathbb{R}^{\mathsf{X}}$ and let $\{T_\sigma\}$ be the associated set of policy operators (as defined in (6.26) on page 196). Prove that $(V, \{T_\sigma\})$ is an ADP.

8.3.1.5 Stable RDPs are ADPs

Not all RDPs generate an ADP. For example, let (Γ, V, B) be an RDP and let $\{T_\sigma\}$ be the policy operators. We cannot say that $(V, \{T_\sigma\})$ is an ADP because we cannot be sure that each T_σ has a unique fixed point.

On the other hand, every stable RDP generates an ADP. The next lemma gives details.

Lemma 8.3.1. *Let $\mathscr{R} := (\Gamma, V, B)$ be an RDP with policy operators $\{T_\sigma\}_{\sigma\in\Sigma}$. If $\mathscr{R}$ is stable, then $(V, \{T_\sigma\}_{\sigma\in\Sigma})$ is an abstract dynamic program.*

Proof. By assumption, V is a sublattice of $\mathbb{R}^{\mathsf{X}}$. Hence $(V, \leqslant)$ is a lattice. Now fix $\sigma \in \Sigma$ and observe that, by Exercise 8.1.5, T_σ is order-preserving on V. By the stable RDP assumption, T_σ is globally stable on V and therefore has a unique fixed point v_σ. Hence, to verify part (i) of the definition of an ADP, it remains only to show that T_σ is downward and upward stable (i.e., parts (i)-(c) and (i)-(d) of the definition hold).

To see that this is so, fix $v \in V$ and suppose first that $v \leqslant T_\sigma\, v$. Iterating on this inequality and using the fact that T_σ is order-preserving gives $v \leqslant T_\sigma^k\, v$ for all $k \in \mathbb{N}$. Applying global stability and taking the limit gives $v \leqslant v_\sigma$. Hence upward stability holds. The proof of downward stability is analogous.

Regarding part (ii) of the definition, fix $v \in V$ and let $\hat{\sigma} \in \Sigma$ be v-greedy. Then $(T_{\hat{\sigma}}\, v)(x) \geqslant (T_\sigma\, v)(x)$ for all $x \in \mathsf{X}$. Hence $\{T_\sigma\, v\}_{\sigma\in\Sigma}$ has a greatest element. A similar argument replacing argmax with argmin in (8.13) shows that a least element also exists. This completes the proof that every stable RDP generates an ADP. □

8.3.1.6 Values

Consider again a general ADP and let $\{v_\sigma\}_{\sigma\in\Sigma}$ be the set of σ-value functions. From these elements we construct the **value function** as the element of V defined by

$$v^* := \bigvee_{\sigma\in\Sigma} v_\sigma. \tag{8.41}$$

In addition,

(i) A policy $\sigma \in \Sigma$ is called **optimal** for the ADP if $v_\sigma = v^*$.

(ii) **Bellman's principle of optimality** is said to hold if the set of optimal policies in Σ is identical to the set of v^*-greedy policies.

All of the definitions given in this section are direct generalizations of definitions that we used for optimal stopping problems, MDPs and RDPs.

Bellman's way of solving dynamic programs was to exploit what is now widely called the Bellman equation. To state the equation, we first define the **Bellman operator** associated with a fixed ADP as the map T on V defined by

$$Tv = \bigvee_{\sigma \in \Sigma} T_\sigma v \qquad (v \in V). \tag{8.42}$$

The definition of the Bellman operator in (8.42) includes all of the Bellman operators we have met as special cases. For example, given an RDP (Γ, V, B) with policy operators $\{T_\sigma\}$, the Bellman operator $(Tv)(x) = \max_{a \in \Gamma(x)} B(x, a, v)$ can also be characterized by the definition in (8.42). (See Exercise 8.1.7 on page 240.)

Exercise 8.3.2. Show that T is an order-preserving self-map on V.

Exercise 8.3.3. Show that (analogous to Exercise 5.1.9 on page 134),

$$Tv = T_\sigma v \iff \sigma \text{ is } v\text{-greedy}. \tag{8.43}$$

A function $v \in V$ is said to satisfy the **Bellman equation** if it is a fixed point of T; that is, if $v = Tv$.

8.3.1.7 An HPI Operator

We define a self-map H on V via

$$Hv = v_\sigma \text{ where } \sigma \text{ is } v\text{-greedy} \tag{8.44}$$

Iterating with H is an abstract version of the Howard policy iteration (HPI) algorithm defined for the MDP case on page 138. In particular, if we fix $\sigma \in \Sigma$ and set $v_k = H^k v_\sigma$, then $(v_k)_{k \geqslant 0}$ is the sequence of σ-value functions generated by HPI.

Below, we say that **HPI converges in finitely many steps** if, for all $\sigma \in \Sigma$, there exists a $K \in \mathbb{N}$ such that $k \geqslant K$ implies $H^k v_\sigma = v^*$.

For the operator H to be well-defined, we need to always select the same v-greedy policy when H is applied to v (since two v-greedy policies can have different lifetime values). Hence, for this purpose, we enumerate the policy set Σ and, when applying H, always choose the first v-greedy policy. (For example, if the set of v-greedy policies is $\{\sigma_2, \sigma_3, \sigma_4\}$, then we choose σ_2 and set $Hv = v_{\sigma_2}$.) This particular choice of convention makes no difference to our results.

8.3.1.8 Optimality Results

We can now state our main result on abstract dynamic programming.

Theorem 8.3.2. *For any ADP, the following statements are true:*

(i) *the value function v^* is the unique solution to the Bellman equation in V,*

(ii) *v^* is the unique fixed point of H in V,*

(iii) *HPI converges in finitely many steps,*

(iv) *Bellman's principle of optimality holds, and*

(v) *at least one optimal policy exists.*

Theorem 8.3.2 informs us that all ADPs have first-rate optimality properties. Bellman's principle of optimality provides a natural characterization of optimal policies, as well as a straightforward way to compute such policies provided we can calculate the value function. The fact that HPI converges to the value function in finite time confirms that this is possible.

The proof of Theorem 8.3.2 is constructed through a sequence of intermediate results in the remainder of this section. We begin with the following lemma.

Lemma 8.3.3. *If v^* satisfies the Bellman equation, then Bellman's principle of optimality holds and at least one optimal policy exists.*

Proof. Suppose v^* satisfies the Bellman equation. To see that Bellman's principle of optimality holds, recall that σ is optimal if and only if $v_\sigma = v^*$. Since v_σ is the unique fixed point of T_σ, this is equivalent to $T_\sigma v^* = v^*$. Since v^* satisfies the Bellman equation, the last statement is equivalent to $T_\sigma v^* = Tv^*$, which is, in turn equivalent to the statement that σ is v^*-greedy. In other words, Bellman's principle of optimality holds.

Existence of an optimal policy is now trivial, since $v^* \in V$ and hence there exists at least one v^*-greedy policy. By Bellman's principle of optimality, every such policy is optimal. □

Lemma 8.3.4. *If $\sigma \in \Sigma$ and $Tv_\sigma = v_\sigma$, then $v_\sigma = v^*$ and σ is optimal.*

Proof. Let v_σ be as stated. To prove optimality of σ, we need to show that $v_\sigma = v^*$. By definition, $v_\sigma \preceq v^*$, so it suffices to obtain $v_\sigma \succeq v^*$. For this we fix $\tau \in \Sigma$ and observe that $v_\sigma = Tv_\sigma \succeq T_\tau v_\sigma$. Hence, by downward stability, $v_\sigma \succeq v_\tau$. Since $\tau \in \Sigma$ was arbitrary, we have $v_\sigma \succeq v^*$. Hence $v_\sigma = v^*$, as claimed. □

The next lemma tells us that if H has a fixed point, then that fixed point is v^*.

Lemma 8.3.5. *If $v \in V$ and $Hv = v$, then*

(i) *v equals the value function v^*, and*

(ii) *the value function satisfies the Bellman equation.*

Proof. Fix $v \in V$ with $Hv = v$ and let σ be the first v-greedy policy, so that $Hv = v_\sigma$. From the last two equations we obtain $v_\sigma = v$. Combining this equality with the greedy characterization $T_\sigma v = Tv$ yields $v_\sigma = T_\sigma v_\sigma = Tv_\sigma$. Lemma 8.3.4 now implies $v = v_\sigma = v^*$. In particular, $v = v^*$, which proves (i). Regarding (ii), substituting $v_\sigma = v^*$ into the previous equation $v_\sigma = Tv_\sigma$ yields $v^* = Tv^*$, so the value function satisfies the Bellman equation. □

Lemma 8.3.6. *For all $\sigma \in \Sigma$ we have $v_\sigma \preceq Hv_\sigma$.*

Proof. Fix $\sigma \in \Sigma$ and let τ be the first v_σ-greedy policy, so that $Hv_\sigma = v_\tau$. The greedy property of τ and the definition of T yield

$$v_\sigma = T_\sigma v_\sigma \preceq Tv_\sigma = T_\tau v_\sigma.$$

Upward stability now gives $v_\sigma \preceq v_\tau = Hv_\sigma$. □

Lemma 8.3.7. *For any $\sigma \in \Sigma$, the sequence $(v_k)_{k \geqslant 0}$ defined by $v_0 = v_\sigma$ and $v_{k+1} = Hv_k$ for $k \geqslant 0$ converges to v^* in finitely many steps. In particular, there exists a $K \in \mathbb{N}$ such that $H^K v_0 = v^*$ and, moreover, $Hv^* = v^*$.*

Proof. Regarding finite termination, observe from Lemma 8.3.6 that $v_{k+1} \succeq v_k$ for all k. Since the sequence (v_k) is contained in the finite set $\{v_\sigma\}_{\sigma \in \Sigma}$, it must be that $v_{K+1} = v_K$ for some $K \in \mathbb{N}$ (since otherwise we have an infinite sequence that never cycles). But then $Hv_K = v_{K+1} = v_K$, so v_K is a fixed point of H. Lemma 8.3.5 now implies that $v_K = v^*$. □

Proof of Theorem 8.3.2. Lemma 8.3.7 implies that HPI converges in finite time and $Hv^* = v^*$. From Lemma 8.3.5 it follows that $Tv^* = v^*$. Since $Tv^* = v^*$, Lemma 8.3.3 now implies that Bellman's principle of optimality holds and at least one optimal policy exists. It remains to show that v^* is the only solution to the Bellman equation in V.

To see this, suppose that $\bar{v} = T\bar{v}$. On one hand, existence of greedy policies implies that there is a policy σ such that $T_\sigma \bar{v} = T\bar{v}$, so $T_\sigma \bar{v} = \bar{v}$ and hence $\bar{v} = v_\sigma \preceq v^*$. On the other hand, given $\sigma \in \Sigma$, we have $T_\sigma \bar{v} \preceq T\bar{v} = \bar{v}$. Downward stability now gives $v_\sigma \preceq \bar{v}$. Since σ was an arbitrarily chosen policy, we have $v^* \preceq \bar{v}$. Hence $\bar{v} = v^*$, which proves the uniqueness claim.

Finally, combining Lemmas 8.3.5 and 8.3.7 proves that v^* is the unique fixed point of H in V. The proof of Theorem 8.3.2 is now complete. □

8.3.2 Convergence of OPI

In the ADP framework of §8.3.1, the underlying space V lacks any form of topology. This means that we cannot discuss convergence of algorithms such as VFI and OPI. Here specialize to the case of stable RDPs, where suitable topology is available. We prove that OPI (and hence VFI) converges under mild conditions.

8.3.2.1 Convergence

Throughout this section, $\mathcal{S} = (\Gamma, V, B)$ is a stable RDP with feasible policy set Σ, policy operators $\{T_\sigma\}$, Bellman operator T, and value function v^*. As usual, v_σ denotes the unique fixed point of T_σ for all $\sigma \in \Sigma$. Since a stable RDP is an ADP (Lemma 8.3.1), the optimality results in Theorem 8.1.2 all apply.

In this setting we discuss convergence of OPI, as expressed in Algorithm 9. Since VFI is a special case of OPI (with $m = 1$), any convergence results for OPI imply convergence of VFI.

The main result of this section is as follows.

Proposition 8.3.8. *For all $m \in \mathbb{N}$, the sequence (v_k) in V generated by Algorithm 9 obeys $v_k \to v^*$ as $k \to \infty$.*

The proof of the proposition proceeds through a series of exercises and lemmas. For the purposes of the current section we set

$$V_\Sigma := \{v_\sigma\}_{\sigma \in \Sigma} \quad \text{and} \quad V_u := \{v \in V : Tv \geqslant v\}.$$

Exercise 8.3.4. Prove that $V_\Sigma \subset V_u$.

Lemma 8.3.9. *If $v \in V_\Sigma$, then $T^k v \to v^*$ as $k \to \infty$.*

Proof. Fix $v \in V_\Sigma$. On one hand, $v \leqslant v^*$ and hence $T^k v \leqslant T^k v^* = v^*$ for all k. On the other hand, if σ is any policy, then $T_\sigma^k v \leqslant T^k v$ for all k. Hence $T_\sigma^k v \leqslant T^k v \leqslant v^*$ for all k. If we now take σ to be an optimal policy, which exists under the stated assumptions, we have $T_\sigma^k v \to v^*$ as $k \to \infty$. Hence $T^k v \to v^*$, as required. □

Next, fixing $\sigma \in \Sigma$, we define the operator W from V to itself via

$$Wv := T_\sigma^m v \quad \text{where} \quad \sigma \text{ is the first } v\text{-greedy policy.}$$

In the definition, "first" is a convention, as discussed after the definition of the operator H in (8.44). In fact W is an approximation of H, since $T_\sigma^m v \to v_\sigma = Hv$ as $m \to \infty$.

Lemma 8.3.10. *W is an order-preserving self-map on V_u. Moreover,*

$$v \in V_u \implies Tv \leqslant Wv \leqslant T^m v.$$

Proof. As for the self-map property, pick any $v \in V_u$. Since T and T_σ are order-preserving, $v \leqslant Tv$ and σ is v-greedy, we have

$$Wv = T_\sigma T_\sigma^{m-1} v \leqslant T T_\sigma^{m-1} v \leqslant T T_\sigma^{m-1} T v = T T_\sigma^m v = TWv.$$

Hence $Wv \in V_u$ and W is invariant on V_u.

As for the order-preserving property, this is immediate from the definition, since powers of order-preserving self-maps are order-preserving (Exercise 2.1.14).

As for the inequality $Tv \leqslant Wv$, fix $v \in V_u$. Since T_σ is order-preserving, $v \leqslant Tv$ and σ is v-greedy, we have

$$T_\sigma^{m-1} v \leqslant T_\sigma^{m-1} T v = T_\sigma^{m-1} T_\sigma v = Wv.$$

Continuing in the same manner gives $T_\sigma^{m-j} v \leqslant Wv$ for $j < m$ and, in particular, $T_\sigma v \leqslant Wv$. Because σ is v-greedy, this yields $Tv \leqslant Wv$.

Regarding the second inequality, we use the fact that $T_\sigma \leqslant T$ on V and T and T_σ are both order-preserving to obtain $Wv = T_\sigma^m v \leqslant T^m v$ (see Exercise 2.1.20 on page 50). □

Proof of Proposition 8.3.8. Fix $v_0 \in V_\Sigma$. We claim that

$$T^k v_0 \leqslant W^k v_0 \leqslant T^{km} v_0 \quad \text{for all } k \in \mathbb{N}. \tag{8.45}$$

Indeed, this holds for $k = 1$ by $v_0 \in V_u$ and Lemma 8.3.10. Since all operators are order-preserving and invariant on V_u, the extension to arbitrary k follows from Exercise 2.1.20 on page 50.

Now let $(v_k)_{k \geqslant 0}$ be the value sequence generated by OPI. Since $v_k = W^k v_0$, (8.45) holds and $T^i v_0 \to v^*$ as $i \to \infty$, we see that $v_k \to v^*$ as $k \to \infty$. □

8.3.2.2 Tidying Up

To tidy up this section let's spell out the proof of Theorem 8.1.2, which listed all the optimality properties of stable RDPs.

Proof of Theorem 8.1.2. Let $\mathscr{R} := (\Gamma, V, B)$ be a stable RDP and let $(V, \{T_\sigma\})$ be the ADP generated by $\mathscr{R}$. Theorem 8.3.2 tells us that the value function v^* is the unique solution to the Bellman equation in V, that HPI converges to v^* in finitely many steps, that Bellman's principle of optimality holds, and that at least one optimal policy exists. The only claim we still need to prove is that the value sequence (v_k) generated by OPI converges to v^*. But this also holds for stable RDPs by Proposition 8.3.8. The proof is now complete. □

8.4 Isomorphic Dynamic Programs

One service provided by mathematics is to determine when two different problems are essentially the same. For example, students of algebra study group isomorphisms, which tell us when two groups are identical in terms of their characteristics under group operations. Topologically isomorphisms play a similar role for topological spaces—these kinds of isomorphisms are also called homeomorphisms, which we introduced in §2.2.1. We used homeomorphisms to identify topologically conjugate dynamical systems, which is itself a method for connecting essentially identical dynamic models.

Not surprisingly, it is also useful to have a notion of "isomorphic" dynamic programs. In this section we set out such a notion and show that an isomorphic relationship leads to essentially equivalent optimality properties. We exploit these ideas to study additional dynamic programming problems, including minimization problems and decision problems with ambiguity.

8.4.1 Isomorphic ADPs

We begin by reviewing order isomorphisms and then, in §8.4.1.3, we turn to the topic of "isomorphic" dynamic programs. This section assumes that you are familiar with the material on suprema and infima in §A.3.

8.4.1.1 Order Isomorphisms

Our first step is to identify an isomorphic relationship between sets with partial orders. To this end, let P and $\hat{P}$ be two partially ordered sets and let Φ be a map from P to $\hat{P}$. The map Φ is called an **order isomorphism** if Φ is bijective and order-preserving. Similarly, Φ is called an **order anti-isomorphism** if Φ is bijective and order-reversing.

Example 8.4.1. If $P = \hat{P} = \mathbb{R}^n_+$ with the usual pointwise order, then Φ mapping $p \in P$ to $p^2 \in \hat{P}$, where the operation $p \mapsto p^2$ acts pointwise, is an order isomorphism, since Φ is bijective and preserves the pointwise order.

Exercise 8.4.1. Let P and $\hat{P}$ be order-isomorphic; that is, there exists an order isomorphism from P to $\hat{P}$. Prove: If P is totally ordered, then so is $\hat{P}$.

Exercise 8.4.2. Check the following claims:

(i) If Φ is an order isomorphism from P to $\hat{P}$, then Φ^{-1} is order-preserving.

(ii) If Φ is an order anti-isomorphism from P to $\hat{P}$, then Φ^{-1} is order-reversing.

Exercise 8.4.3. Let L and $\hat{L}$ be two lattices and let $\{f_i\}_{i\in I}$ be a finite subset of L. Prove the following statements:

(i) If Φ is an order-preserving map from L to $\hat{L}$, then

$$\Phi \bigvee_i f_i = \bigvee_i \Phi f_i \quad \text{and} \quad \Phi \bigwedge_i f_i = \bigwedge_i \Phi f_i.$$

(ii) If Φ is an order-reversing map from L to $\hat{L}$, then

$$\Phi \bigvee_i f_i = \bigwedge_i \Phi f_i \quad \text{and} \quad \Phi \bigwedge_i f_i = \bigvee_i \Phi f_i.$$

8.4.1.2 Minimization Terminology

Up until now all theory and applications have concerned maximization of lifetime values. However, as well as isomorphisms between dynamic programs, we also wish to consider "anti-isomorphisms" that reverse the order of valuations. For this reason, as well as for the desire to state minimization results, now is a good time to introduce minimization terminology.

To improve clarity, whenever we discuss minimization and maximization together, what were previously called optimal policies (which are maximizers) will be called **max-optimal policies**. Similarly, greedy policies will be called **max-greedy** policies, value functions will be called **max-value functions** and so on. The minimization case will use the prefix "min-" instead, as per the definitions below.

To begin the theory, $\mathscr{A} := (V, \{T_\sigma\})$ be an ADP (see §8.3.1). As usual, for each $\sigma \in \Sigma$, the symbol v_σ denotes the σ-value function; that is the unique fixed point of T_σ. A policy $\sigma^* \in \Sigma$ will be called **min-optimal** if $v_{\sigma^*} \leqslant v_\sigma$ for every $\sigma \in \Sigma$. The **min-value function** is

$$u^* := \inf_{\sigma \in \Sigma} v_\sigma.$$

A policy σ is called **v-min-greedy** if $T_\sigma v \leqslant T_\tau v$ for all $\tau \in \Sigma$.

Example 8.4.2. If $\mathscr{A}$ is generated by an RDP (Γ, V, B), then $\sigma \in \Sigma$ is v-min-greedy if and only if

$$\sigma(x) \in \operatorname*{argmin}_{a \in \Gamma(x)} B(x, a, v) \quad \text{at each } x \in \mathsf{X}.$$

One purpose of the following discussion is to find and characterize min-optimal policies for the ADP $(V, \{T_\sigma\})$. To this end, we define $S \colon V \to V$ via $Sv = \inf_{\sigma \in \Sigma} T_\sigma v$. We call S the **min-Bellman operator**.

Example 8.4.3. In the RDP setting of Example 8.4.2, this definition is equivalent to

$$(Sv)(x) = \min_{a \in \Gamma(x)} B(x, a, v) \qquad (x \in \mathsf{X}). \tag{8.46}$$

Exercise 8.4.4. Show that σ is v-min-greedy if and only if $T_\sigma v = Sv$.

8.4.1.3 Definitions and Results

To begin, we define ADPs $\mathscr{A} = (V, \{T_\sigma\}_{\sigma \in \Sigma})$ and $\hat{\mathscr{A}} = (\hat{V}, \{\hat{T}_\sigma\}_{\sigma \in \Sigma})$ to be **isomorphic** if $\mathscr{A}$ and $\hat{\mathscr{A}}$ have the same policy sets Σ and, in addition, there exists an order isomorphism

Φ from V to $\hat{V}$ such that T_σ and $\hat{T}_\sigma$ are conjugate under Φ for all $\sigma \in \Sigma$. As per the discussion in §2.2.1, the last statement means that

$$T_\sigma = \Phi^{-1} \circ \hat{T}_\sigma \circ \Phi \quad \text{for all } \sigma \in \Sigma. \tag{8.47}$$

In addition, we call $\mathscr{A}$ and $\hat{\mathscr{A}}$ **anti-isomorphic** if the same definition holds after replacing "order isomorphism" with "order anti-isomorphism."

In what follows, for ADPs $\mathscr{A} = (V, \{T_\sigma\})$ and $\hat{\mathscr{A}} = (\hat{V}, \{\hat{T}_\sigma\})$, we let

- v_σ (resp., $\hat{v}_\sigma$) be the fixed point of T_σ (resp., $\hat{T}_\sigma$),
- T (resp., $\hat{T}$) be the Bellman operator of $\mathscr{A}$ (resp., $\hat{\mathscr{A}}$),
- v^* (resp., $\hat{v}^*$) be the max-value function of $\mathscr{A}$ (resp., $\hat{\mathscr{A}}$) and
- u^* (resp., $\hat{u}^*$) be the min-value function of $\mathscr{A}$ (resp., $\hat{\mathscr{A}}$)

We seek relationships between value functions and optimal properties of isomorphic ADP pairs.

Theorem 8.4.1. *If the ADPs $\mathscr{A}$ and $\hat{\mathscr{A}}$ are isomorphic under Φ, then*

(i) $T = \Phi^{-1} \circ \hat{T} \circ \Phi$ *and* $S = \Phi^{-1} \circ \hat{S} \circ \Phi$ *on* V,

(ii) $\hat{v}^* = \Phi\, v^*$ *and* $\hat{u}^* = \Phi\, u^*$,

(iii) σ *is max-optimal for* $\mathscr{A}$ *if and only if* σ *is max-optimal for* $\hat{\mathscr{A}}$.

(iv) σ *is min-optimal for* $\mathscr{A}$ *if and only if* σ *is min-optimal for* $\hat{\mathscr{A}}$.

Thus, under the isomorphic relationship, the max- and min-value functions are identified under the order-isomorphism and $\mathscr{A}$ and $\hat{\mathscr{A}}$ have identical optimal policies.

Proof of Theorem 8.4.1. Assume the conditions of Theorem 8.4.1. For given $v \in V$, applying the supremum pass-through for order isomorphisms in Exercise 8.4.3 and (8.47) yields

$$Tv = \bigvee_\sigma T_\sigma\, v = \bigvee_\sigma \Phi^{-1}\hat{T}_\sigma\, \Phi\, v = \Phi^{-1} \bigvee_\sigma \hat{T}_\sigma\, \Phi\, v = \Phi^{-1}\hat{T}\, \Phi\, v.$$

Hence (V, T) and $(\hat{V}, \hat{T})$ are conjugate under Φ, and, therefore, their fixed points are connected by $\hat{v}^* = \Phi\, v^*$ (see Exercise 2.2.2 on page 55). Similarly,

$$Sv = \bigwedge_\sigma T_\sigma\, v = \bigwedge_\sigma \Phi^{-1}\hat{T}_\sigma\, \Phi\, v = \Phi^{-1} \bigwedge_\sigma \hat{T}_\sigma\, \Phi\, v = \Phi^{-1}\hat{S}\, \Phi\, v.$$

Hence (V, S) and $(\hat{V}, \hat{S})$ are conjugate under Φ, and therefore $\hat{u}^* = \Phi u^*$. Together, these results prove (i)–(ii).

Regarding claim (iii), note that σ is max-optimal for $\mathscr{A}$ if and only if $T_\sigma v^* = v^*$, which, by the bijection property of Φ, is also equivalent to $\Phi T_\sigma v^* = \hat{v}^*$. Using conjugacy, we can also write this as $\hat{T}_\sigma \hat{v}^* = \hat{v}^*$, which is equivalent to the statement that σ is max-optimal for $\hat{\mathscr{A}}$.

The proof of (iv) is similar. □

Here is a result for the order-reversing case that corresponds to Theorem 8.4.1:

Theorem 8.4.2. *If the ADPs $\mathscr{A}$ and $\hat{\mathscr{A}}$ are anti-isomorphic under Φ, then*

(i) $T = \Phi^{-1} \circ \hat{S} \circ \Phi$ *and* $S = \Phi^{-1} \circ \hat{T} \circ \Phi$ *on* V,

(ii) $\hat{u}^* = \Phi v^*$ *and* $\hat{v}^* = \Phi u^*$, *and*

(iii) σ *is max-optimal for* $\mathscr{A}$ *if and only if* σ *is min-optimal for* $\hat{\mathscr{A}}$.

(iv) σ *is min-optimal for* $\mathscr{A}$ *if and only if* σ *is max-optimal for* $\hat{\mathscr{A}}$.

Proof. Assume the conditions of Theorem 8.4.2. For given $v \in V$, applying the properties of order anti-isomorphisms in Exercise 8.4.3, we get

$$Tv = \bigvee_\sigma T_\sigma v = \bigvee_\sigma \Phi^{-1}\hat{T}_\sigma \Phi v = \Phi^{-1} \bigwedge_\sigma \hat{T}_\sigma \Phi v = \Phi^{-1}\hat{S}\, \Phi v.$$

Hence (V, T) and $(\hat{V}, \hat{S})$ are conjugate under Φ. Since $\hat{u}^*$ is the fixed point of $\hat{S}$, we obtain $\hat{u}^* = \Phi v^*$. Similarly, for $v \in V$,

$$Sv = \bigwedge_\sigma T_\sigma v = \bigwedge_\sigma \Phi^{-1}\hat{T}_\sigma \Phi v = \Phi^{-1} \bigvee_\sigma \hat{T}_\sigma \Phi v = \Phi^{-1}\hat{T}\, \Phi v.$$

Since $\hat{v}^*$ is the fixed point of $\hat{T}$, we obtain $\hat{v}^* = \Phi u^*$. This proves claims (i)–(ii).

Regarding claim (iii), note that σ is max-optimal for $\mathscr{A}$ if and only if $T_\sigma v^* = v^*$, which, by the bijection property of Φ, is also equivalent to $\Phi T_\sigma v^* = \Phi v^* = \hat{u}^*$. Using conjugacy, we can also write this as $T_\sigma \Phi^{-1} v^* = \hat{T}_\sigma \hat{u}^* = \hat{u}^*$, which is equivalent to the statement that σ is min-optimal for $\hat{\mathscr{A}}$.

The proof of (iv) is similar. □

8.4.1.4 Isomorphic RDPs

The preceding results are simple and broad but the isomorphic (or anti-isomorphic) relationships are not trivial to establish. To this end, let us give conditions on RDPs under which they generate isomorphic (or anti-isomorphic) ADPs. These conditions are more restrictive but easier to check in applications.

To this end, let $\mathscr{R} = (\Gamma, V, B)$ and $\hat{\mathscr{R}} = (\Gamma, \hat{V}, \hat{B})$ be two RDPs with identical state space X, action space A and feasible correspondence Γ. Suppose that

$$V = \mathbb{M}^{\mathsf{X}} \quad \text{and} \quad \hat{V} = \hat{\mathbb{M}}^{\mathsf{X}} \quad \text{where } \mathbb{M}, \hat{\mathbb{M}} \subset \mathbb{R},$$

and, in addition, that there exists a homeomorphism φ from $\mathbb{M}$ onto $\hat{\mathbb{M}}$ such that

$$B(x, a, v) = \varphi^{-1}[\hat{B}(x, a, \varphi \circ v)] \quad \text{for all } v \in V \text{ and } (x, a) \in \mathsf{G}. \tag{8.48}$$

In this setting, we have the following result:

Lemma 8.4.3. *The RDP $\mathscr{R}$ is stable if and only if $\hat{\mathscr{R}}$ is stable.*

Proof. Let Φ be defined by $\Phi v = \varphi \circ v$. Since φ is a homeomorphism from $\mathbb{M}$ to $\hat{\mathbb{M}}$, Φ is a homeomorphism from V to $\hat{V}$. Moreover, for any $\sigma \in \Sigma$, the respective policy operators T_σ and $\hat{T}_\sigma$ are linked by

$$(T_\sigma v)(x) = B(x, \sigma(x), v) = \varphi^{-1}[\hat{B}(x, \sigma(x), \varphi \circ v)] = \varphi^{-1}[(\hat{T}_\sigma \varphi \circ v)(x)].$$

In other words, $T_\sigma v = \Phi^{-1} \hat{T}_\sigma \Phi v$. Hence (V, T_σ) and $(\hat{V}, \hat{T}_\sigma)$ are topologically conjugate, from which it follows that T_σ is global stable if and only if $\hat{T}_\sigma$ is global stable. This completes the proof of Lemma 8.4.3. □

In what follows, we call stable RDPs $\mathscr{R}$ and $\hat{\mathscr{R}}$ **isomorphic** if the ADPs they generate are isomorphic. Likewise, we call $\mathscr{R}$ and $\hat{\mathscr{R}}$ **anti-isomorphic** if the ADPs they generate are anti-isomorphic.

Proposition 8.4.4. *If (8.48) holds and $\mathscr{R}$ and $\hat{\mathscr{R}}$ are both stable, then the following statements are true:*

(i) *$\mathscr{R}$ and $\hat{\mathscr{R}}$ are isomorphic whenever φ is increasing on $\mathbb{M}$.*

(ii) *$\mathscr{R}$ and $\hat{\mathscr{R}}$ are anti-isomorphic whenever φ is decreasing on $\mathbb{M}$.*

Proof. Let $\mathscr{A}$ and $\hat{\mathscr{A}}$ be the ADPs generated by $\mathscr{R}$ and $\hat{\mathscr{R}}$ respectively. Let Φ be defined by $\Phi v = \varphi \circ v$. Consider the first case in which φ is increasing on $\mathbb{M}$. Since φ is

increasing, Φ is an order isomorphism from V to $\hat{V}$. Moreover, we saw in the proof of Lemma 8.4.3 that the respective policy operators are linked by $T_\sigma v = \Phi^{-1}\hat{T}_\sigma \Phi v$. Hence $\mathscr{A}$ and $\hat{\mathscr{A}}$ are isomorphic ADPs. Thus, by definition, $\mathscr{R}$ and $\hat{\mathscr{R}}$ are isomorphic RDPs, as claimed.

The proof of the anti-isomorphic case is analogous and omitted. □

8.4.2 Minimization Problems

In this section we consider minimization problems. We then provide an application that concerns loss minimization in the setting of a negative discount rate.

The next theorem tells us that the ADPs have strong optimality properties vis-à-vis minimization as well as vis-a-vis maximization.

Theorem 8.4.5. *If $\mathscr{A} = (V, \{T_\sigma\})$ is an ADP with min-value function u^* and min-Bellman operator S, then*

(i) *u^* is the unique fixed point of S in V,*

(ii) *a policy $\sigma \in \Sigma$ is min-optimal if and only if σ is u^*-min greedy, and*

(iii) *at least one min-optimal policy exists.*

Item (iii) is a version of **Bellman's principle of optimality**, rephrased for the minimization case. To Theorem 8.4.5 we can also add a statement that a min-oriented version of HPI converges in finite time; in the interests of brevity, we leave such details aside. Given our preceding work on anti-isomorphic ADPs, we will find that the proof of Theorem 8.4.5 is just a matter of finding the right anti-isomorphism.

Proof of Theorem 8.4.5. We take $\mathscr{A}$ as given and introduce a second ADP $\hat{\mathscr{A}}$ as follows. Let $\preceq$ be the partial order on V, so that $V = (V, \preceq)$, and let $\hat{V}$ be the partially ordered space $(\hat{V}, \hat{\preceq})$, where $\hat{V}$ is a copy of V and $\hat{\preceq}$ is the reversed partial order; that is, for $u, v \in \hat{V}$, we have $u \hat{\preceq} v$ in $\hat{V}$ if and only if $v \preceq u$ in V. Under this reversed partial order, the identity map I from V to $\hat{V}$ defined by $Iv = v$ is an anti-isomorphism. Now we define $\hat{\mathscr{A}}$ as $(\hat{V}, \{\hat{T}_\sigma\})$ where $\hat{V}$ is as just defined and $\hat{T}_\sigma = T_\sigma = I \circ T_\sigma \circ I^{-1}$ for all σ.

By construction, $\mathscr{A}$ and $\hat{\mathscr{A}}$ are anti-isomorphic under the I. From Theorem 8.4.2, we obtain $T = \hat{S}$ and $S = \hat{T}$ on V, while $\hat{u}^* = v^*$ and $\hat{v}^* = u^*$. Since $\hat{v}^*$ is the unique fixed point of $\hat{T}$, it follows that u^* is the unique fixed point of S. This proves claim (i).

Regarding claim (ii), Theorem 8.4.2 tells us that that σ is min-optimal for $\mathscr{A}$ if and only if σ is max-optimal for $\hat{\mathscr{A}}$; which is in turn true if and only if it $\hat{T}_\sigma \hat{v}^* = \hat{v}^*$. From

the above identities, this is the same statement as $T_\sigma u^* = u^* = Su^*$, which is equivalent to the statement that σ is u^*-min-greedy.

Regarding (iii), since u^* has at least one min-greedy policy, a min-optimal policy exists. □

8.4.3 Application: Negative Discount Rate Optimality

When discussing MDPs we used β to represent the discount factor. Given β, the **discount rate** or **rate of time preference** is the value ρ that solves $\beta = 1/(1+\rho)$. The standard MDP assumption $\beta < 1$ implies this rate is positive. You will recall from Chapter 5 that the condition $\beta < 1$ is essential to the general theory of MDPs, since it yields global stability of the Bellman and policy operators on $\mathbb{R}^{\mathsf{X}}$ (via the Neumann series lemma or Banach's fixed point theorem).

But behavior consistent with positive discount rates is not universal. For example, negative rates of time preference are commonly inferred when people face unpleasant tasks. Subjects of studies often prefer getting such tasks "over and done with" rather than postponing them. (Negative discount rates are inferred in other settings as well. §8.5 provides background and references.)

In this section, we model optimal choice under a zero or negative discount rate. Taking our cue from the discussion above, we consider a scenario where a task generates disutility but has to be completed. In particular, we assume that

$$B(x, x', v) = c(x, x') + \beta v(x') \qquad (x, x' \in \mathsf{X}) \tag{8.49}$$

where X is a finite set and β is some positive constant. We take X to be both the state and action space, so the action is choosing a feasible next-period state. The function c gives the cost of transitioning from x to the new state x' and we assume that $c(x, x') > 0$ when $x \neq x'$ and $c(x, x) = 0$ for all x.

(If c is understood as utility rather than cost, then (8.49) generalizes the cake eating problem in Example 8.1.2.)

There are several different interpretations we can give this problem. To focus on one, let's consider the problem of optimal traversal of a network, along the lines of the shortest path problem considered in Example 8.1.8. To repeat, the problem is to traverse a directed graph $\mathscr{G} = (\mathsf{X}, E)$, where X is the vertices of the graph and E is the edges (intuitively, $(x, x') \in E$ means that there exists an "arrow" from x to x'). The aim is to find the minimum cost path from given x to a specified destination node d.

Letting $\mathscr{O}(x) := \{y \in \mathsf{X} : (x, y) \in E\}$ be the direct successors of x, we define the feasible correspondence Γ by setting $\Gamma(x)$ to be all x' such that

$$x' \in \mathscr{O}(x), \qquad x' \neq x, \quad \text{and} \quad c(x, x') + \beta c(x', d) \leqslant c(x, d).$$

The first condition is just feasibility. The second and third conditions are restrict choices to "rational" decisions: the first says that the controller cannot stay still, while the second says that the controller should not move to d in two steps when doing so is more costly than moving there in one. The latter two restrictions narrow down the policy space and help us solve the problem even when the discount rate is negative.

We can frame this problem as an RDP $\mathscr{R} = (\Gamma, B, V)$ with Γ, B as just described and

$$V := [v_1, v_2] \quad \text{where} \quad v_1(x) := 0 \quad \text{and} \quad v_2(x) := c(x, d). \tag{8.50}$$

Here is our main result for this section.

Proposition 8.4.6. *$\mathscr{R}$ is a concave RDP.*

It follows that $\mathscr{R}$ is stable and the fundamental optimality results apply (see Proposition 8.2.8). For example, from Theorem 8.4.5, we see that the min-value function u^* is the unique solution to

$$u^*(x) = \min_{x' \in \Gamma(x)} \{c(x, x') + \beta u^*(x')\}$$

in V and a policy $\sigma \in \Sigma$ is min-optimal if and only if

$$\sigma(x) \in \operatorname*{argmin}_{x' \in \Gamma(x)} \{c(x, x') + \beta u^*(x')\}$$

for all $x \in \mathsf{X}$. Notice that, in line with zero or negative discounting, Proposition 8.4.6 permits $\beta \geqslant 1$.

Proof of Proposition 8.4.6. We must first check that $\mathscr{R}$ is an RDP. The only nontrivial assertion is consistency, in the sense that $v_1 \leqslant T_\sigma v \leqslant v_2$ for all $\sigma \in \Sigma$ and all $v \in V$ (this is condition (8.3)). Since T_σ is order-preserving, it suffices to show that $T_\sigma v_1 \geqslant v_1$ and $T_\sigma v_2 \leqslant v_2$. Since $v_1 = 0$, the first inequality is obvious. Regarding the second, fix $\sigma \in \Sigma$ and $x \in \mathsf{X}$. Observe that, by the definition of v_2 and Γ in (8.50),

$$(T_\sigma v_2)(x) = c(x, x') + \beta c(x', d) \leqslant c(x, d) = v_2(x),$$

so $T_\sigma v_2 \leqslant v_2$ is confirmed.

Now we show that $\mathscr{R}$ is also concave. For this we must confirm the concavity property (8.34) on page 260, but this is obvious because our aggregator B in (8.49) is linear in v. We also need to confirm (8.35) on page 260, which, since $v_1 = 0$, reduces to showing existence of a $\delta > 0$ such that $B(x, x', v_1) \geqslant \delta v_2(x)$ for all $(x, x') \in \mathsf{G}$, or, equivalently,

$$c(x, x') \geqslant \delta c(x, d) \text{ for all } (x, x') \in \mathsf{G}. \tag{8.51}$$

To this end, let

$$\delta := \min \left\{ \frac{c(x, x')}{c(x, d)} \ : \ x' \in \Gamma(x) \text{ and } x \neq d \right\}$$

Fix $x \in \mathsf{X}$ and $x' \in \Gamma(x)$. For $x = d$ (8.51) is trivial, since $c(d, d) = 0$. For $x \neq d$ we have $c(x, x') \geqslant \delta c(x, d)$ by the definition of δ, so (8.51) is confirmed for all x. This concludes the proof that $\mathscr{R}$ is a concave RDP. □

Remark 8.4.1. With $\beta = 1$, the argument above provides a new proof of the traditional shortest path problem introduced in Example 8.1.8.

8.4.4 Epstein–Zin Revisited

In this section we return to the Epstein–Zin specification. We saw in Proposition 8.1.1 that the Epstein–Zin RDP (Γ, V, B) is stable when $r \gg 0$, P is irreducible and $0 < \beta < 1$. The value space V for this result is all $v \in \mathbb{R}^{\mathsf{X}}$ such that $v \gg 0$, while the aggregator is

$$B(x, a, v) = \left\{ r(x, a) + \beta \left(\sum_{x'} v(x')^{\gamma} P(x, a, x') \right)^{\alpha/\gamma} \right\}^{1/\alpha}, \tag{8.52}$$

In this section we study the same model from a different angle, which unifies a number of ideas we have used to treat the Epstein–Zin specification up till now. Our main aim is to show that the original model is either isomorphic or anti-isomorphic to a simpler and more tractable model. Throughout, we continue to assume that $r \gg 0$, P is irreducible and $0 < \beta < 1$.

Our first step is to introduce another RDP by setting

$$\hat{B}(x, a, v) = B\left(x, a, v^{1/\gamma}\right)^{\gamma} \tag{8.53}$$

We set $\mathscr{R} := (\Gamma, B, V)$ and $\hat{\mathscr{R}} := (\Gamma, \hat{B}, V)$.

The value of of introducing $\hat{\mathscr{R}}$ comes from the fact that $\hat{\mathscr{R}}$ is easier to work with than $\mathscr{R}$ (just as the modified Epstein–Zin Koopmans operator $\hat{K}$ defined in §7.2.3.3

turned out to be easier to work with than the original Epstein–Zin Koopmans operator K introduced in §7.2.3.2). This will be evident in the proof of the next lemma.

Lemma 8.4.7. *Under the stated assumptions, $\hat{\mathscr{R}}$ is a stable RDP.*

Proof. To see this, fix $(x, a) \in \mathsf{G}$ and observe that

$$\hat{B}(x, a, w) = \left\{ r(x, a) + \beta \left(\sum_{x'} w(x') P(x, a, x') \right)^{1/\theta} \right\}^{\theta} \quad \text{where} \quad \theta := \frac{\gamma}{\alpha}.$$

As a result, each policy operator T_σ takes the form

$$(T_\sigma v)(x) = \left\{ r(x, \sigma(x)) + \beta \left(\sum_{x'} w(x') P(x, \sigma(x), x') \right)^{1/\theta} \right\}^{\theta}$$

The operator T_σ is a special case of $\hat{K}$ in (7.13), on page 222. We saw in §7.2.3.3 that this operator is globally stable under the stated assumptions. Hence $\hat{\mathscr{R}}$ is a stable RDP. □

By combining Lemma 8.4.7 with Lemma 8.4.3, we see that $\mathscr{R}$ is also a stable RDP.

Lemma 8.4.8. *The following statements are true:*

(i) *If $\gamma > 0$, then $\mathscr{R}$ and $\hat{\mathscr{R}}$ are isomorphic.*

(ii) *If $\gamma < 0$, then $\mathscr{R}$ and $\hat{\mathscr{R}}$ are anti-isomorphic.*

Proof. The map $\varphi(m) = m^\gamma$ is a homeomorphism from V to itself and (8.48) holds under φ. The claims in Lemma 8.4.8 follow from Proposition 8.4.4. □

To put the above ideas to work, suppose, for example, that that $\gamma < 0$, as often occurs in the asset pricing literature (the case $\gamma > 0$ is analogous). In view of Lemma 8.4.8 and Theorem 8.4.2, any min-optimal policy for $\hat{\mathscr{R}}$ will be max-optimal for $\mathscr{R}$. Hence we can solve for the Epstein–Zin optimal policy either by directly solving $\mathscr{R}$ for a max-optimal policy or by solving $\hat{\mathscr{R}}$ for a min-optimal policy. The best choice depends on simplicity and numerical stability.

8.5 Chapter Notes

The main optimality results in this chapter draw on the excellent textbook by Bertsekas (2022b), who in turn credits Mitten (1964) as the first research paper to frame Richard Bellman's dynamic programming problems in an abstract setting. Related contemporaneous publications include Denardo and Mitten (1967) and Denardo (1967). Our central optimality results from this chapter (Theorem 8.3.2 and Theorem 8.1.2) are constructed from our own research, although closely related results appear in Bertsekas (2022b), Bloise et al. (2023), and other sources. Our strategy of discussing abstract dynamic programming in a setting with no topology is partly inspired by Kamihigashi (2014).

The job search application with quantile preferences in §8.2.1.4 is based on de Castro et al. (2022). The same reference includes a general theory of dynamic programming when certainty equivalents are computed using quantile operators and aggregation is additively separable.

The optimal default application in §8.2.1.5 is loosely based on Arellano (2008). Influential contributions to this line of work include, Yue (2010), Chatterjee and Eyigungor (2012), Arellano and Ramanarayanan (2012), Cruces and Trebesch (2013), Ghosh et al. (2013), Gennaioli et al. (2014), and Bocola et al. (2019).

As for negative discounting, Loewenstein and Sicherman (1991) found that the majority of workers they surveyed reported a preference for increasing wage profiles over decreasing ones that yield the same undiscounted sum, even when it was pointed out that the latter could be used to construct a dominating consumption sequence. Loewenstein and Prelec (1991) obtained similar results. In summarizing their study, they argue that, in the context of the choice problems they examined, "sequences of outcomes that decline in value are greatly disliked, indicating a negative rate of time preference" (Loewenstein and Prelec, 1991, p. 351).

At the start of the chapter we provided some motivation for RDPs by mentioning that equilibria in some models of production and economic geography can be computed using dynamic programming. Examples include Hsu (2012), Hsu et al. (2014), Antràs and De Gortari (2020), Kikuchi et al. (2021) and Tyazhelnikov (2022).

Dynamic programming problems that acknowledge model uncertainty by including adversarial agents to promote robust decision rules can be found in Cagetti et al. (2002), Hansen and Sargent (2011), and other related papers. Al-Najjar and Shmaya (2019) study the connection between Epstein–Zin utility and parameter uncertainty. Ruszczyński (2010) considers risk averse dynamic programming and time consistency.

An algorithm that we neglected to discuss is stochastic gradient descent (or ascent) in policy space. Typically policies are parameterized an approximation architecture

that consists of basis functions, activation functions, and compositions of them (e.g., a neural network). In large models, such approximation is used even when the state and action spaces are finite, simply because the curse of dimensionality makes exact representations infeasible. For recent discussions of gradient descent in policy spaces see Nota and Thomas (2019), Mei et al. (2020), and Bhandari and Russo (2022).

Part I

Appendices

Appendix A

Suprema and Infima

This section of the appendix contains an extremely brief review of basic facts concerning sets, functions, suprema and infima. We recommend Bartle and Sherbert (2011) for those who wish to learn more.

A.1 Sets and Functions

A **set** is a collection of objections viewed as a whole. Examples include the set of **natural numbers** $\mathbb{N} := \{1, 2, \ldots\}$ and $[n] := \{1, 2, \ldots, n\}$ when $n \in \mathbb{N}$. The set that contains no elements is called the **empty set** and denoted by $\emptyset$. A set X is called **finite** if we can write $\mathsf{X} = \{x_1, \ldots, x_n\}$ for some $n \in \mathbb{N}$. Note that, according to our definition, every finite set is automatically nonempty.

A **function** f from set A to set B, written $A \ni x \mapsto f(x) \in B$ or $f\colon A \to B$, is a rule associating to each and every element a in A one and only one element $b \in B$. The point b is also written as $f(a)$, and called the **image** of a under f. For $C \subset A$, the set $f(C)$ is the set of all images of points in C, and is called the image of C under f. Also, for $D \subset B$, the set $f^{-1}(D)$ is all points in A that map into D under f, and is called the **preimage** of D under f.

A function $f\colon A \to B$ is called **one-to-one** if distinct elements of A are always mapped into distinct elements of B, and **onto** if every element of B is the image under f of at least one point in A. A **bijection** or **one-to-one correspondence** from A to B is a function f from A to B that is both one-to-one and onto.

If $f\colon A \to B$ and $g\colon B \to C$, then the **composition** of f and g is the function $f \circ g$ from A to C defined at $a \in A$ by $(f \circ g)(x) := g(f(a))$.

A.2 Sup, Inf, Max and Min

Given a subset A of $\mathbb{R}$, we call $u \in \mathbb{R}$ an **upper bound** of A if $a \leqslant u$ for all a in A. Let $U(A)$ be the set of all upper bounds of A. An element $\bar{u}$ of $\mathbb{R}$ is called a **supremum** or **least upper bound** of A if

(i) $\bar{u} \in U(A)$ and

(ii) $\bar{u} \leqslant u$ for every $u \in U(A)$.

Example A.2.1. For the set $I := [0,1] \subset \mathbb{R}$, the number 1 is an upper bound of I. Moreover, for if u is an upper bound of I, then $u \geqslant 1$. Hence 1 is the supremum of I.

Example A.2.2. $\mathbb{N}$ has no supremum in $\mathbb{R}$, since the set of upper bounds is empty.

Exercise A.2.1. Show that, for all of the sets $(0,1)$, $[0,1)$ and $(0,1]$, the number 1 is the supremum of the set.

Exercise A.2.2. Fix $A \subset \mathbb{R}$. Prove that, for $s \in U(A)$, we have $s = \sup A$ if and only if, for all $\varepsilon > 0$, there exists a point $a \in A$ with $a > s - \varepsilon$.

Exercise A.2.3. Fix $A \subset \mathbb{R}$. Prove that A has at most one supremum.

One of the most important properties of $\mathbb{R}$ is stated below.

Theorem A.2.1 (Least upper bound property)**.** *Every nonempty subset of $\mathbb{R}$ with an upper bound in $\mathbb{R}$ has a supremum in $\mathbb{R}$.*

Theorem A.2.1 is often taken as axiomatic in formal constructions the real numbers. (Alternatively, one may assume completeness of the reals and then prove Theorem A.2.1 using this property. See, e.g., Bartle and Sherbert (2011).)

For $A \subset \mathbb{R}$, a **lower bound** of A is any number ℓ such that $\ell \leqslant a$ for all $a \in A$. If $i \in \mathbb{R}$ is an lower bound for A and also satisfies $i \geqslant \ell$ for every lower bound ℓ of A, then i is called the **infimum** of A and we write $i = \inf A$. At most one such i exists, and every nonempty subset of $\mathbb{R}$ bounded from below has an infimum.

A number m contained in a subset A of $\mathbb{R}$ is called the **maximum** of A and we write $m = \max A$ if $a \leqslant m$ for every $a \in A$. It is called the **minimum** of A if $a \geqslant m$ for every $a \in A$.

Exercise A.2.4. Prove: If A is a finite subset of $\mathbb{R}$, then $\sup A = \max A$.

Exercise A.2.5. Show that, if A is a closed and bounded subset of $\mathbb{R}$, then A has both a maximum and a minimum.

Exercise A.2.6. Prove the following statements:

(i) If $A \subset B$, then $\sup A \leqslant \sup B$.

(ii) If $s = \sup A$ and $s \in A$, then $s = \max A$.

(iii) If $i = \inf A$ and $i \in A$, then $i = \min A$.

Given an arbitrary set D and a function $f\colon D \to \mathbb{R}$, define

$$\sup_{x\in D} f(x) := \sup\{f(x) : x \in D\} \quad \text{and} \quad \max_{x\in D} f(x) := \max\{f(x) : x \in D\}$$

whenever the latter exists. A point $x^* \in D$ is called a

- **maximizer** of f on D if $x^* \in D$ and $f(x^*) \geqslant f(x)$ for all $x \in D$, and a
- **minimizer** of f on D if $x^* \in D$ and $f(x^*) \leqslant f(x)$ for all $x \in D$.

Equivalently, $x^* \in D$ is a maximizer of f on D if $f(x^*) = \max_{x\in D} f(x)$, and a minimizer if $f(x^*) = \min_{x\in D} f(x)$. We define

$$\operatorname*{argmax}_{x\in D} f(x) := \{x^* \in X : f(x^*) \geqslant f(x) \text{ for all } x \in D\}.$$

The set $\operatorname{argmin}_{x\in D} f(x)$ is defined analogously.

A **real sequence** is a map x from $\mathbb{N}$ to $\mathbb{R}$, with the value of the function at $k \in \mathbb{N}$ typically denoted by x_k rather than $x(k)$. A real sequence $x = (x_k)_{k\geqslant 1} := (x_k)_{k\in\mathbb{N}}$ is said to **converge** to $\bar{x} \in \mathbb{R}$ if, for each $\varepsilon > 0$, there exists an $N \in \mathbb{N}$ such that $|x_k - \bar{x}| < \varepsilon$ for all $k \geqslant N$. In this case we write $\lim_k x_k = x$ or $x_k \to x$ Bartle and Sherbert (2011) give an excellent introduction to real sequences and their basic properties.

Exercise A.2.7. Let (x_k) be a bounded monotone increasing sequence in $\mathbb{R}$. Prove that $\sup_k x_k = \lim_k x_k$.

A.3 Lattices

A.3.1 Suprema and Infima in Partially Ordered Sets

The definitions of suprema and infima on the real line in Appendix A extend naturally to partially ordered sets. For example, given a partially ordered set $(P, \preceq)$ and a nonempty subset A of P, we call $u \in P$ an **upper bound** of A if $a \preceq u$ for all a in A. An element $\bar{u}$ of P is called a **supremum** of A if

(i) $\bar{u}$ is an upper bound of A and

(ii) $\bar{u} \preceq u$ for every upper bound u of A.

If it exists, then

- the supremum of A is typically written as $\sup A$.
- If $A = \{a_i\}_{i \in I}$ for some index set I, we also write $\sup A$ as $\vee_i a_i$.
- If $A = \{a, b\}$, then $\sup A$ is also written as $a \vee b$.

Exercise A.3.1. Prove the following statements in the setting described above:

(i) If $\bar{u}$ is a supremum of A and $\bar{u} \in A$, then $\bar{u}$ is a greatest element (see §2.1.1.4) of A.

(ii) If g is a greatest element of A then $g = \sup A$.

We call $\ell \in P$ a **lower bound** of A if $a \succeq \ell$ for all a in A. An element $\bar{\ell}$ of P is called a **infimum** of A if $\bar{\ell}$ is an upper bound of A and $\bar{\ell} \succeq u$ for every upper bound ℓ of A. We use analogous notation to denote the infimum. For example, if $A = \{a, b\}$, then the infimum $\inf A$ is also written as $a \wedge b$.

Example A.3.1. Let M be a nonempty set and let $\wp(M)$ be the set of all subsets of M, partially ordered by $\subset$. Let $\{A_i\}_{i \in I}$ be a subset of $\wp(M)$, so that $A_i \in \wp(M)$ for each i in nonempty index set I. Then the supremum and infimum of $\{A_i\}_{i \in I}$ in $(\wp(M), \subset)$ are given by

$$\bigvee_i A_i = \bigcup_i A_i \quad \text{and} \quad \bigwedge_i A_i = \bigcap_i A_i$$

respectively. To see the former, observe that $A_j \subset \cup_i A_i$ for all $j \in I$. Hence $\cup_i A_i$ is an upper bound of $\{A_i\}$. Moreover, if $B \subset M$ and $A_j \subset B$ for all $i \in I$, then $\cup_i A_i \subset B$. This proves that $\cup_i A_i$ is the supremum. The proof of the infimum case is similar.

A.3.2 Suprema and the Pointwise Order

Let X be a nonempty set and V be a subset of $\mathbb{R}^{\mathsf{X}}$. We treat V as a partially ordered set, with partial order given by the pointwise order $\leqslant$. Let $\{f_i\} := \{f_i\}_{i \in I}$ be a finite collection of functions in V. As usual, we take $\vee_i f_i$ to be the pointwise maximum of the functions $\{f_i\}$ (see §1.2.4.3).

Lemma A.3.1. *In the setting described above, $\vee_i f_i$ is the supremum of $\{f_i\}$ in V whenever $\vee_i f_i \in V$.*

Proof. Suppose first that $\vee_i f_i \in V$. Then, under the pointwise order $\leqslant$, we have $f_i \leqslant \vee_i f_i$ for all $i \in I$ and $u \in V$. Hence $\vee_i f_i$ is an upper bound of $\{f_i\}$ in V. Moreover, if $u \in V$ and $u \geqslant f_i$ for all $i \in I$, then $u \geqslant \vee_i f_i$. Hence $\vee_i f_i$ is the supremum of $\{f_i\}$ in V, as claimed. □

One corollary of Lemma A.3.1 is that $\vee_i f$ is always the supremum of $\{f_i\}$ in $(\mathbb{R}^{\mathsf{X}}, \leqslant)$.

The discussion above also tells us that we can compute the supremum of a collection of vectors $\{u_i\} \subset \mathbb{R}^n$ by taking the pointwise maximum. Similarly, the infimum is obtained by taking the pointwise infimum.

Exercise A.3.2. Let X be any set and let $\mathbb{R}^{\mathsf{X}}$ have the pointwise order. Is it true that, for a finite subset $F := \{f_i\}_{i \in I}$ of $\mathbb{R}^{\mathsf{X}}$, the set F contains a greatest element.

Remark A.3.1. Lemma A.3.1 tells us that, for $f, g \in \mathbb{R}^{\mathsf{X}}$, the supremum $f \vee g$ is the pointwise maximum and $f \wedge g$ is the pointwise minimum. This agrees with the terminology and notation we introduced in §1.2.4.1.

A.3.3 Existence and Uniqueness

Let P be any partially ordered set and let A be a nonempty subset of P. The following uniqueness results are simple to verify:

Exercise A.3.3. Prove that A has at most one supremum and one infimum in P.

Existence of suprema and infima is less trivial in partially ordered sets than the totally ordered case. For example, we know that $A \subset \mathbb{R}$ has a supremum in $\mathbb{R}$ whenever A is bounded above (Theorem A.2.1). But now let $P = (0, 1)$, partially ordered by the usual order $\leqslant$. The set $A = [1/2, 1)$ is bounded above in $\mathbb{R}$ but has no supremum in P. Indeed, if $s = \sup A$, the $s \in P$ and $a \leqslant s$ for all $s \in A$. This is clearly impossible.

A.3.4 Lattices

A nonempty partially ordered set P is called a **lattice** if, given any two elements p and q of P, both $p \vee q$ and $p \wedge q$ exist in P.

Example A.3.2. As follows from the discussion in §A.3.2, if X is any nonempty set, then the set $\mathbb{R}^{\mathsf{X}}$ paired with the pointwise order $\leqslant$ is a lattice.

Example A.3.3. Let $C[a, b]$ be the set of all continuous functions from $[a, b] \subset \mathbb{R}$ to $\mathbb{R}$. The set $C[a, b]$ is a lattice when paired with the pointwise order. Indeed, given $f, g \in C[a, b]$, the functions $f \vee g$ and $f \wedge g$ are also continuous, and therefore lie in $C[a, b]$.

Example A.3.4. Consider the unit disc $D := \{v \in \mathbb{R}^2 : \|v\| \leqslant 1\}$ endowed with the usual pointwise order. Let $e_1 := (1, 0)$ and $e_2 := (0, 1)$ be the canonical basis vectors in $\mathbb{R}^2$. The set $\{e_1, e_2\}$ lies in D but has no upper bound in D. Hence D is not a lattice.

Exercise A.3.4. Let P be a lattice. Prove that $\wedge_i p_i$ and $\vee_i p_i$ are both in P whenever $\{p_i\}_{i \in I} \subset P$ and I is finite.

Appendix B

Remaining Proofs

Proof of Lemma 3.2.1. Regarding (i), fix $\varphi, \psi \in \mathscr{D}(\mathsf{X})$ with $\varphi \preceq_{\mathrm{F}} \psi$. Pick any $y \in \mathsf{X}$. By transitivity of partial orders, the function $u(x) := \mathbb{1}\{y \preceq x\}$ is in $i\mathbb{R}^{\mathsf{X}}$. Hence $\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x)$. Given the definition of u, this is equivalent to $G^{\varphi}(y) \leqslant G^{\psi}(y)$. As y was chosen arbitrarily, we have $G^{\varphi} \leqslant G^{\psi}$ pointwise on X.

Regarding (ii), let $\varphi, \psi \in \mathscr{D}(\mathsf{X})$ be such that $G^{\varphi} \leqslant G^{\psi}$ and let X be totally ordered by $\preceq$. We can write X as $\{x_1, \ldots, x_n\}$ with $x_i \preceq x_{i+1}$ for all i. Pick any $u \in i\mathbb{R}^{\mathsf{X}}$ and let $\alpha_i = u(x_i)$. By Exercise 3.2.2, we can write u as $u(x) = \sum_{i=1}^n s_i \mathbb{1}\{x \geqslant x_i\}$ at each $x \in \mathsf{X}$, where $s_i \geqslant 0$ for all i. Hence

$$\sum_{x \in \mathsf{X}} u(x)\varphi(x) = \sum_{x \in \mathsf{X}} \sum_{i=1}^n s_i \mathbb{1}\{x \geqslant x_i\}\varphi(x) = \sum_{i=1}^n s_i \sum_{x \in \mathsf{X}} \mathbb{1}\{x \geqslant x_i\}\varphi(x) = \sum_{i=1}^n s_i\, G^{\varphi}(x_i).$$

A similar argument gives $\sum_{x \in \mathsf{X}} u(x)\psi(x) = \sum_{i=1}^n s_i\, G^{\psi}(x_i)$. Since $G^{\varphi} \leqslant G^{\psi}$, we have

$$\sum_{x \in \mathsf{X}} u(x)\varphi(x) = \sum_{i=1}^n s_i\, G^{\varphi}(x_i) \leqslant \sum_{i=1}^n s_i\, G^{\psi}(x_i) = \sum_{x \in \mathsf{X}} u(x)\psi(x).$$

We conclude that $\varphi \preceq_{\mathrm{F}} \psi$, as was to be shown. □

Below we prove Theorem 6.1.1. We begin with the following result concerning expectations over products.

Lemma B.0.1. *For each $t \in \mathbb{N}$ and $x \in \mathsf{X}$, we have*

$$\mathbb{E}_x \left\{ \left[\prod_{i=1}^t B_i \right] h(X_t) \right\} = \sum_{x' \in \mathsf{X}} A^t(x, x') h(x'). \tag{B.1}$$

In the proof and remainder of this section, we set $H_t := h(X_t)$ for all t.

Proof. We verify the claim in Lemma B.0.1 using induction on t. The claim holds at $t = 1$ because, for any such h and x,

$$\mathbb{E}_x\left[B_1 H_1\right] = \sum_{x'} b(x,x')h(x')P(x,x') = \sum_{x'} A(x,x')h(x').$$

Now suppose it holds at t. We claim it also holds at $t+1$. To show this we apply the law of iterated expectations to obtain

$$\mathbb{E}_x\left[B_1 \cdots B_{t+1}H_{t+1}\right] = \mathbb{E}_x\left[\mathbb{E}_t\left[B_1 \cdots B_{t+1}H_{t+1}\right]\right] = \mathbb{E}_x\left[B_1 \cdots B_t \mathbb{E}_t[B_{t+1}H_{t+1}]\right].$$

Since $\mathbb{E}_t B_{t+1}H_{t+1} = \sum_y b(X_t,y)h(y)P(X_t,y)$, we can now write

$$\mathbb{E}_x\left[B_1 \cdots B_{t+1}H_{t+1}\right] = \mathbb{E}_x\left[B_1 \cdots B_t f(X_t)\right] \quad \text{where} \quad f(x) := \sum_y A(x,y)h(y). \tag{B.2}$$

Applying the induction hypothesis (B.1) to the right-hand side of the first equation in (B.2) (with $h = f$), we now have

$$\mathbb{E}_x\left[B_1 \cdots B_{t+1}H_{t+1}\right] = \sum_{x'} A^t(x,x')f(x') = \sum_{x'} A^t(x,x') \sum_y A(x',y)h(y).$$

But $\sum_{x'} A^t(x,x')A(x',y) = A^{t+1}(x,y)$, so (B.1) holds at $t+1$ as well. The proof is now complete. □

Now we can complete the proof of Theorem 6.1.1.

Proof of Theorem 6.1.1. We fix $x \in \mathsf{X}$ and use Lemma B.0.1 to obtain

$$v(x) = \mathbb{E}_x\left\{\sum_{t=0}^{\infty}\left[\prod_{i=1}^{t} B_i\right]H_t\right\} = \sum_{t=0}^{\infty}\mathbb{E}_x\left\{\left[\prod_{i=1}^{t} B_i\right]H_t\right\} = \sum_{t=0}^{\infty}(A^t h)(x). \tag{B.3}$$

Writing (B.3) pointwise gives, $v = \sum_{t\geqslant 0} A^t h$.[1] By the Neumann series lemma and $r(A) < 1$, this sum converges and equals $(I - A)^{-1}h$. □

[1] In (B.3) we again passed expectations through an infinite sum. This operation takes some care but is valid under the assumption $r(A) < 1$. Footnote 2 on page 91 provides more information.

Appendix C

Solutions to Selected Exercises

Solution to Exercise 1.1.1. Here is one possible answer: On one hand, providing additional unemployment compensation is costly for taxpayers and tends to increase the unemployment rate. On the other hand, unemployment compensation encourages the worker to reject low initial offers, leading to a better lifetime wage. This can enhance worker welfare and expand the tax base. A larger model is needed to disentangle these effects.

Solution to Exercise 1.2.1. Fix α, s and t with $s \geqslant 0$. Suppose first that $\alpha \geqslant s + t$. Then $\alpha \vee (s + t) = \alpha \leqslant \alpha \vee t \leqslant s + \alpha \vee t$, as claimed. Suppose next that $\alpha \leqslant s + t$. Then $\alpha \vee (s + t) = s + t \leqslant s + \alpha \vee t$, as required.

Solution to Exercise 1.2.4. For $\alpha > 0$ we always have $\|\alpha u\|_0 = \|u\|_0$, which violates positive homogeneity.

Solution to Exercise 1.2.12. Let T and U be as stated in the exercise. Regarding uniqueness, suppose that T has two distinct fixed points u and y in U. Since $T^m u = \bar{u}$ and $T^m y = \bar{u}$, we have $T^m u = T^m y$. But u and y are distinct fixed points, so $u = T^m u$ must be distinct from $y = T^m y$. Contradiction.

Regarding the claim that $\bar{u}$ is a fixed point, we recall that $T^k u = \bar{u}$ for $k \geqslant m$. Hence $T^m \bar{u} = \bar{u}$ and $T^{m+1} \bar{u} = \bar{u}$. But then

$$T\bar{u} = TT^m \bar{u} = T^{m+1} \bar{u} = \bar{u},$$

so $\bar{u}$ is a fixed point of T.

Solution to Exercise 1.2.13. Assume the hypotheses of the exercise and let $u_m := T^m u$ for all $m \in \mathbb{N}$. By continuity and $u_m \to u^*$ we have $Tu_m \to Tu^*$. But the sequence (Tu_m) is just (u_m) with the first element omitted, so, given that $u_m \to u^*$, we must have $Tu_m \to u^*$. Since limits are unique, it follows that $u^* = Tu^*$.

Solution to Exercise 1.2.15. Let the stated hypotheses hold and fix $u \in C$. By global stability we have $T^k u \to u^*$. Since T is invariant on C we have $(T^k u)_{k\in\mathbb{N}} \subset C$. Since C is closed, this implies that the limit is in C. In other words, $u^* \in C$, as claimed.

Solution to Exercise 1.2.17. By the definition of the operator norm we have $\|Au\| \leqslant \|A\|_\infty \|u\|$ for all $u \in \mathbb{R}^n$. If $\|A\|_\infty < 1$, then T is a contraction of modulus $\|A\|_\infty$, since, for any $x, y \in U$,

$$\|Ax + b - Ay - b\| = \|A(x - y)\| \leqslant \|A\|_\infty \|x - y\|.$$

Solution to Exercise 1.2.17. By the definition of the derivative, for any $x \in U := (0, \infty)$, we have

$$\lim_{y\to x} \left| \frac{g(y) - g(x)}{y - x} - g'(x) \right| = 0.$$

Hence, by the reverse triangle inequality, for fixed $\varepsilon > 0$, we can take a $\delta > 0$ such that

$$\left| \frac{g(y) - g(x)}{y - x} \right| > |g'(x)| - \varepsilon = g'(x) - \varepsilon$$

for all $y \in (x - \delta, x + \delta)$. Rearranging gives

$$|g(x) - g(y)| > [g'(x) - \varepsilon]|x - y|$$

for all $y \in (x - \delta, x + \delta)$. But $g'(x) = s\alpha x^{\alpha-1} + 1 - \delta$, which converges to $+\infty$ as $x \to 0$. It follows that, for any $\lambda \in [0, 1)$, we can find a pair x, y such that $|g(x) - g(y)| > \lambda|x - y|$. Hence g is not a contraction map under $|\cdot|$.

Solution to Exercise 1.2.19. From the bound in Exercise 1.2.18, we obtain

$$\|u_m - u_k\| \leqslant \frac{\lambda^m}{1 - \lambda} \lambda^i \|u_0 - u_1\| \qquad (m, k \in \mathbb{N} \text{ with } m < k).$$

Hence (u_m) is Cauchy, as claimed.

Solution to Exercise 1.2.21. Let T_1, T_2 be contraction maps on U of modulus λ_1

and λ_2 respectively. Fix $u, v \in U$. We have

$$\|Tu - Tv\|_\infty = \|(T_1 u) \vee (T_2 u) - (T_1 v) \vee (T_2 v)\|_\infty = \max_i |\max_j (T_j u)_i - \max_j (T_j v)_i|,$$

where i ranges over $1, \ldots, n$ and where j ranges over $1, 2$. Applying Lemma 1.2.5 and reversing the order of maxima gives

$$\|Tu - Tv\|_\infty \leqslant \max_i \max_j |(T_j u)_i - (T_j v)_i| = \max_j \max_i |(T_j u)_i - (T_j v)_i|.$$

From the definition of the supremum norm and our assumptions on T_1, T_2, this becomes

$$\|Tu - Tv\|_\infty \leqslant \max_j \|T_j u - T_j v\|_\infty \leqslant \max_j \lambda_j \|u - v\|_\infty.$$

Hence T is a contraction of modulus $\lambda := \max_j \lambda_j$.

Solution to Exercise 1.2.22. Fix $\alpha \in (0, 1]$ and let F be defined by $Fu = (1 - \alpha)u + \alpha Tu$. Readers will be able to verify that F is also a contraction with identical fixed point $\bar{u}$, and that damped iteration is just iteration with F. The claim follows.

Solution to Exercise 1.2.25. Fix $\tau \in [0, 1]$, $X \sim \varphi \in \mathscr{D}(\mathsf{X})$ and $\alpha \in \mathbb{R}$. Let Φ_X be the CDF of X. Let $Y := X + \alpha$, let $\mathsf{Y} := \{x + \alpha : x \in \mathsf{X}\}$ and let Φ_Y the CDF of Y. Note that $\Phi_Y(y) = \mathbb{P}\{Y \leqslant y\} = \mathbb{P}\{X \leqslant y - \alpha\} = \Phi_X(y - \alpha)$ for all $y \in \mathsf{Y}$.

Let $x^* := Q_\tau X$ and let $y^* = Q_\tau(X + \alpha) = \min\{y \in \mathsf{Y} : \Phi_Y(y) \geqslant \tau\}$. We need to show that $y^* = x^* + \alpha$. We do this by proving $y^* \geqslant x^* + \alpha$ and $y^* \leqslant x^* + \alpha$.

For the first inequality, fix $y \in \mathsf{Y}$ such that $\Phi_Y(y) \geqslant \tau$. Let $x = y - \alpha$. We then have $\Phi_Y(x + \alpha) \geqslant \tau$ and hence $\Phi_X(x) \geqslant \tau$. Hence $x \geqslant x^*$, or $y \geqslant x^* + \alpha$. Since this last inequality holds for any $y \in \mathsf{Y}$ with $\Phi_Y(y) \geqslant \tau$, we have $y^* \geqslant x^* + \alpha$.

For the reverse inequality, fix $x \in \mathsf{X}$ with $\Phi_X(x) \geqslant \tau$ and set $y = x + \alpha$. We have $\Phi_Y(y) = \Phi_X(y - \alpha) = \Phi_X(x) \geqslant \tau$, so $y \geqslant y^*$, or $x \geqslant y^* - \alpha$. Since the last inequality holds for all $x \in \mathsf{X}$ with $\Phi_X(x) \geqslant \tau$, we have $x^* \geqslant y^* - \alpha$. Rearranging gives $y^* \leqslant x^* + \alpha$, as was to be shown.

Solution to Exercise 1.2.26. Since $\min f = -\max(-f)$ and similarly for g, we can apply Lemma 1.2.5 to obtain

$$|\min f - \min g| = |\max(-g) - \max(-f)| \leqslant \max |(-g) - (-f)| = \max |f - g|.$$

Solution to Exercise 1.3.1. Fix $\alpha, x, y \in \mathbb{R}$. We have $x = x - y + y \leqslant |x - y| + y$. Applying the result in Exercise 1.2.1 yields

$$\alpha \vee x \leqslant |x - y| + \alpha \vee y \quad \iff \quad \alpha \vee x - \alpha \vee y \leqslant |x - y|.$$

Reversing the roles of x and y completes the proof.

Solution to Exercise 2.1.3. Let $M = \{1, 2\}$, let $A = \{1\}$ and let $B = \{2\}$. Then $A \subset B$ and $B \subset A$ both fail. Hence $\subset$ is not a total order on $\wp(M)$.

Solution to Exercise 2.1.6. Just set $\mathsf{X} = [n] \times [k]$.

Solution to Exercise 2.1.7. Let $I_f := [f_1, f_2]$ and $I_g := [g_1, g_2]$. Consider the order interval $I := [f_1 \wedge g_1, f_2 \vee g_2]$. If $h \in I$, then $h \geqslant f_1 \wedge g_1$, so $h \geqslant f_1$ and $h \geqslant g_1$. A similar argument gives $h \leqslant f_2$ and $h \leqslant g_2$. Hence $h \in I_f \cap I_g$. Working in the other direction, it is not difficult to show that $h \in I_f \cap I_g$ implies $h \in I$. Hence $I = I_f \cap I_g$. In particular, $I_f \cap I_g$ is an order interval in $C[0, 1]$.

Solution to Exercise 2.1.8. Regarding the first claim, fix $B \in \mathbb{M}^{m \times k}$ with $b_{ij} \geqslant 0$ for all i, j. Pick any $i \in [m]$ and $u \in \mathbb{R}^k$. By the triangle inequality, we have $|\sum_j b_{ij} u_j| \leqslant \sum_j b_{ij} |u_j|$. Stacking these inequalities yields $|Bu| \leqslant B|u|$, as was to be shown.

Regarding the second, let A and (u_k) be as stated, with $u_{k+1} \leqslant Au_k$ for all k. We aim to prove $u_k \leqslant A^k u_0$ for all k using induction. In doing so, we observe that $u_1 \leqslant Au_0$, so the claim is true at $k = 1$. Suppose now that it holds at $k - 1$. Then $u_k \leqslant Au_{k-1} \leqslant AA^{k-1}u_0 = A^k u_0$, where the last step used nonnegativity of A and the induction hypothesis. The claim is now proved.

Solution to Exercise 2.1.9. Assume the stated conditions. Let $h := v - u$ and let a_{ij} be the i, j-th element of A. We have $h \geqslant 0$ and $h_j > 0$ at some j. Hence $\sum_j a_{ij} h_j > 0$. This says that every row of Ah is strictly positive. In other words $Ah = A(v - u) \gg 0$. The claim follows.

Solution to Exercise 2.1.10. Fix $a, b \in \mathbb{R}_+$ and $c \in \mathbb{R}_+$. By (2.1), we have

$$a \wedge c = (a - b + b) \wedge c \leqslant (|a - b| + b) \wedge c \leqslant |a - b| \wedge c + b \wedge c.$$

Thus, $a \wedge c - b \wedge c \leqslant |a - b| \wedge c$. Reversing the roles of a and b gives $b \wedge c - a \wedge c \leqslant |a - b| \wedge c$. This proves the claim in Exercise 2.1.10.

Solution to Exercise 2.1.11. We prove the claim concerning greatest elements: Suppose that g and g' are greatest elements of A. Then, since both are in A, we have $g \preceq g'$ and $g' \preceq g$. Hence, by asymmetry, $g = g'$.

Solution to Exercise 2.1.12. If $\{v_\sigma\}$ has a greatest element $\bar{v}$, then, since $\bar{v}$ is in the set $\{v_\sigma\}$ and dominates every other function, we have $\bar{v}(x) = \max_\sigma v(x)$ at all x. Hence $\bar{v}$ is the pointwise maximum.

Solution to Exercise 2.1.13. To see this, suppose first that $S \in \{A_i\}$. Since $A_j \subset \cup_i A_i =: S$ for all $j \in I$, the set S is a greatest element of $\{A_i\}$. Conversely, if S is not in $\{A_i\}$, then S is not a greatest element (since the definition directly requires that $S \in \{A_i\}$.

Solution to Exercise 2.1.14. Let A and P be as stated. The claim that A^k is order-preserving on P holds at $k = 1$. Suppose now that it holds at k and fix $p, q \in P$ with $p \preceq q$. By the induction hypothesis and the fact that A is order-preserving, we have $AA^k p \preceq AA^k q$. Hence $A^{k+1}p \preceq A^{k+1}q$. We conclude that A^{k+1} is also order-preserving, as was to be shown.

Solution to Exercise 2.1.15. Fix an $n \times k$ matrix A with $A \geqslant 0$, along with $u, v \in \mathbb{R}^k$. We need to show that $u \leqslant v$ implies $Au \leqslant Av$ for any conformable vectors u, v. This holds because if $u \leqslant v$ we have $v - u \geqslant 0$, so $A(v - u) \geqslant 0$. But then $Av - Au \geqslant 0$, or $Au \leqslant Av$.

Solution to Exercise 2.1.16. Fix square A, B with $0 \leqslant A \leqslant B$. It follows from the rules of matrix multiplication that, for arbitrary nonnegative square matrices E, F, G with $F \leqslant G$, we have $EF \leqslant EG$ and $FE \leqslant GE$. Hence, if $A^k \leqslant B^k$ for some $k \in \mathbb{N}$, then $A^{k+1} = AA^k \leqslant BA^k \leqslant BB^k = B^{k+1}$. Thus, by induction, $A^k \leqslant B^k$ for all $k \in \mathbb{N}$, which verifies the first claim. Regarding the second, it is clear that for nonnegative matrices E, F with $E \leqslant F$ we have $\|E\|_\infty \leqslant \|F\|_\infty$. Hence $\|A^k\|_\infty \leqslant \|B^k\|_\infty$ for all $k \in \mathbb{N}$. Raising both sides to the power $1/k$ and applying Gelfand's lemma verifies $r(A) \leqslant r(B)$.

Solution to Exercise 2.1.18. Take $(f_k)_{k \geqslant 1}$ in $i\mathbb{R}^P$ and $f \in \mathbb{R}^P$ with $f_k \to f$ as $k \to \infty$. Since $f_k \to f$ we have $f_k(z) \to f(z)$ for all $z \in P$. (Norm convergence implies pointwise convergence.) Fix $x, y \in P$ with $x \preceq y$. From $(f_k) \subset i\mathbb{R}^P$ we have $f_k(x) \leqslant f_k(y)$ for all k. Since weak inequalities are preserved under limits, $f(x) \leqslant f(y)$. Hence $f \in i\mathbb{R}^P$.

Solution to Exercise 2.1.20. Let P, S, T be as described in the exercise. We aim to show that $S^k \preceq T^k$ holds for all $k \in \mathbb{N}$. Clearly it holds for $k = 1$. If it also holds at $k-1$, then, for any $u \in P$, we have $S^k u = SS^{k-1}u \leqslant ST^{k-1}u \leqslant TT^{k-1}u = T^k u$, where we used the induction hypothesis, the order-preserving property of S and the assumption that $S \preceq T$.

Solution to Exercise 2.1.23. Fix $\beta_1 \leqslant \beta_2$. Let g_1 and g_2 be the corresponding fixed point maps, as defined in (1.34). Since $\beta_1 \leqslant \beta_2$, we have $g_1(h) \leqslant g_2(h)$ for all $h \in \mathbb{R}_+$ and, in addition, g_2 is a contraction map (and hence globally stable), Proposition 2.1.3 applies. In particular, the fixed point h_1^* corresponding to β_1 is less than or equal to h_2^*, the fixed point corresponding to β_2.

Solution to Exercise 2.2.1. To show that $T = \Phi^{-1} \circ \hat{T} \circ \Phi$ holds, we can equivalently prove that $\Phi \circ T = \hat{T} \circ \Phi$. For $u \in \mathbb{R}$, we have $\Phi T u = \ln A + \alpha \ln u$ and $\hat{T}\Phi u = \ln A + \alpha \ln u$. Hence $\Phi \circ T = \hat{T} \circ \Phi$, as was to be shown.

Solution to Exercise 2.2.2. Let (U, T) and $(\hat{U}, \hat{T})$ be conjugate under Φ, with $\hat{T} \circ \Phi = \Phi \circ T$. The stated equivalence holds because

$$Tu = u \iff \Phi T u = \Phi u \iff \hat{T}\Phi u = \Phi u.$$

Solution to Exercise 2.2.3. From $\hat{T} = \Phi \circ T \circ \Phi^{-1}$ we have $\hat{T}^2 = \Phi \circ T \circ \Phi^{-1} \circ \Phi \circ T \circ \Phi^{-1} = \Phi \circ T^2 \circ \Phi^{-1}$ and, continuing in the same way (or using induction), $\hat{T}^k = \Phi \circ T^k \circ \Phi^{-1}$ for all $k \in \mathbb{N}$. Equivalently, $\hat{T}^k \circ \Phi = \Phi \circ T^k$ for all $k \in \mathbb{N}$. Hence, using continuity of Φ and Φ^{-1},

$$T^k u \to u^* \iff \Phi T^k u \to \Phi u^* \iff \hat{T}^k \Phi u \to \Phi u^*.$$

Solution to Exercise 2.2.5. Since $u_{k+1} = Tu_k$, we have

$$\frac{u_{k+1} - u^*}{u_k - u^*} = T' u^* + \frac{T'' v}{2}(u_k - u^*).$$

Taking absolute values and using $u_k \to u^*$ confirms the linear rate claimed in the exercise.

Solution to Exercise 2.3.1. Let A be as stated and let e be the right eigenvector in (2.7). Since e is nonnegative and nonzero, and since eigenvectors are defined only up to constant multiples, we can and do assume that $\sum_j e_j = 1$. From $Ae = r(A)e$ we

have $\sum_j a_{ij} e_j = r(A) e_i$ for all i. Summing with respect to i gives $\sum_j \operatorname{colsum}_j(A) e_j = r(A)$. Since the elements of e are nonnegative and sum to one, $r(A)$ is a weighted average of the column sums. Hence the second pair of bounds in Lemma 2.3.2 holds. The remaining proof is similar (use the left eigenvector).

Solution to Exercise 2.3.2. Let P and Q be as stated. Evidently $PQ \geqslant 0$. Moreover, $PQ\mathbb{1} = P\mathbb{1} = \mathbb{1}$, so PQ is Markov. That $r(P) = 1$ follows directly from Lemma 2.3.2. By the Perron–Frobenius theorem, there exists a nonzero, nonnegative row vector φ satisfying $\varphi P = \varphi$. Rescaling φ to $\varphi/(\varphi\mathbb{1})$ gives the desired vector ψ.

The final positivity and uniqueness claim is also by the Perron–Frobenius theorem, and its consequences for irreducible matrices. Indeed, if φ is another nonnegative vector satisfying $\varphi\mathbb{1} = 1$ and $\varphi P = \varphi$, then, by the Perron–Frobenius theorem, $\varphi = \alpha\psi$ for some $\alpha > 0$. But then $\alpha\psi\mathbb{1} = 1$ and $\psi\mathbb{1} = 1$, which gives $\alpha = 1$. Hence $\varphi = \psi$.

Solution to Exercise 2.3.3. Let P and ε have the stated properties. Fix $h \in \mathbb{R}^{\mathsf{X}}$. It suffices to show that for this arbitrary h we can find an $x \in \mathsf{X}$ such that $(Ph)(x) < h(x) + \varepsilon$. This is easy to verify, since, for $\bar{x} \in \operatorname{argmax}_{x \in \mathsf{X}} h(x)$ we have $(Ph)(\bar{x}) = \sum_{x'} h(x') P(\bar{x}, x') \leqslant h(\bar{x})$.

Solution to Exercise 2.3.4. It is straightforward to confirm that both columns of A sum to $1 + g$. As a result, with $\mathbb{1}^\top$ as a row vector of ones, we have

$$n_{t+1} = \mathbb{1}^\top x_{t+1} = \mathbb{1}^\top A x_t = (1+g)\mathbb{1}^\top x_t = (1+g) n_t,$$

as was to be shown.

Solution to Exercise 2.3.10. Fix $L \in \mathbb{L}(\mathsf{X})$ with $(Lu)(x) = \sum_{x' \in \mathsf{X}} L(x, x') u(x')$ for all $x \in \mathsf{X}$ and $u \in \mathbb{R}^{\mathsf{X}}$. Positivity of L requires that

$$u \geqslant 0 \implies \sum_{x' \in \mathsf{X}} L(x, x') u(x') \geqslant 0 \text{ for all } x \in \mathsf{X}.$$

Clearly, this holds whenever $L(x, x') \geqslant 0$ for all $x, x' \in \mathsf{X}$.

Regarding the converse, suppose that L is positive. Seeking a contradiction, suppose in addition that we can find a pair $(x_a, x_b) \in \mathsf{X} \times \mathsf{X}$ such that $L(x_a, x_b) < 0$. With $u(x) := \mathbb{1}\{x = x_b\}$, we have $(Lu)(x_a) = \sum_{x' \in \mathsf{X}} L(x_a, x') u(x') = L(x_a, x_b) < 0$. This contradicts positivity of L.

Solution to Exercise 2.3.11. Suppose first that L is positive. Fix $u \leqslant v$ in $\mathbb{R}^{\mathsf{X}}$ and observe that, by positivity, $L(v-u) \geqslant 0$. But then $Lv - Lu \geqslant 0$ and hence $Lu \leqslant Lv$. This shows that L is order-preserving.

Regarding the converse, if L is order-preserving, then $u \geqslant 0$ implies $Lu \geqslant L0$. But for every linear operator we have $L0 = 0$, so $Lu \geqslant 0$. Hence L is a positive operator.

Solution to Exercise 3.1.1. Let $X_t = x \in \mathsf{X}$, so that $X_{t+1} = \max\{x - D_{t+1}, 0\} + S\mathbb{1}\{x \leqslant s\}$. Evidently X_{t+1} is integer-valued and nonnegative. If $x \leqslant s$, then $X_{t+1} \leqslant \max\{s - D_{t+1}, 0\} + S \leqslant s + S$. Similarly, if $s < x \leqslant S + s$, then $X_{t+1} \leqslant \max\{x - D_{t+1}, 0\} \leqslant S + s$. The claim is verified.

Solution to Exercise 3.1.2. Let $x \in \mathsf{X}$ be the current state at time t and suppose first that $s < x$. The next period state X_{t+1} hits s with positive probability, since $\varphi(d) > 0$ for all $d \in \mathbb{Z}_+$. The state X_{t+2} hits $S + s$ with positive probability, since $\varphi(0) > 0$. From $S + s$, the inventory level reaches any point in $\mathsf{X} = \{0, \ldots, S + s\}$ in one step with positive probability. Hence, from current state x, inventory reaches any other state y with positive probability in three steps.

The logic for the case $x \leqslant s$ is similar and left to the reader.

Solution to Exercise 3.1.3. Fix $t \in \mathbb{N}$. Under the stated hypotheses, we have $X_t \stackrel{d}{=} \psi_0 P^t$ (see (3.4)). Hence

$$\mathbb{E}h(X_t) = \sum_{x'} h(x')\mathbb{P}\{X_t = x'\} = \sum_{x'} h(x')(\psi_0 P^t)(x') = \left\langle \psi_0 P^t, h \right\rangle .$$

Solution to Exercise 3.1.6. Assume P is positive with unique stationary distribution ψ^*. Since $r(P) = 1$, the last part of the Perron–Frobenius theorem tells us that $P^t \to e\,\varepsilon$ as $t \to \infty$, where e and ε are the dominant right and left eigenvectors, normalized such that $\langle e, \varepsilon \rangle = 1$. In this case we know ψ^* is the dominant left eigenvector and $\mathbb{1}$ is the dominant right eigenvector. Moreover, $\psi^* \in \mathscr{D}(\mathsf{X})$ yields $\langle \psi^*, \mathbb{1} \rangle = 1$. Hence, for any $\psi \in \mathscr{D}(\mathsf{X})$, we have

$$\psi P^t \to \psi \mathbb{1}\, \psi^* = \psi^* \quad \text{as} \quad t \to \infty.$$

Hence global stability holds, as claimed.

Solution to Exercise 3.1.10. Since we are conditioning on $X_t = x$, we can replace X_{t+1} with $\rho x + \nu \varepsilon_{t+1}$. The result then follows from $\mathbb{P}\{\alpha < \nu\varepsilon_{t+1} \leqslant \beta\} = F(\beta) - F(\alpha)$.

Solution to Exercise 3.2.2. Set $\alpha_k := u(x_k)$ for all k and $s_k := \alpha_k - \alpha_{k-1}$ with $\alpha_0 := 0$. Fix $x_j \in \mathsf{X}$. Then

$$\sum_{k=1}^{n} s_k \mathbb{1}\{x_j \geqslant x_k\} = \sum_{k=1}^{j} s_k = (\alpha_1 - \alpha_0) + (\alpha_2 - \alpha_1) + \ldots + (\alpha_j - \alpha_{j-1}) = \alpha_j.$$

In other words, $\sum_{k=1}^{n} s_k \mathbb{1}\{x_j \geqslant x_k\} = u(x_j)$. This completes the proofs.

Solution to Exercise 3.2.3. Fix $\varphi, \psi \in \mathsf{X}$ and suppose that $\varphi \preceq_{\mathrm{F}} \psi$. Let $u \in \mathbb{R}^{\mathsf{X}}$ be defined by $u(1) = 0$ and $u(2) = 1$. Then, by the definition of stochastic dominance, we have $\varphi(2) \leqslant \psi(2)$. Since $\varphi(1) = 1 - \varphi(2)$ and $\psi(1) = 1 - \psi(2)$, this inequality is equivalent to $\psi(1) \leqslant \varphi(1)$. Finally, suppose that $\psi(1) \leqslant \varphi(1)$ and fix $u \in i\mathbb{R}^{\mathsf{X}}$. Let $h = u(2) - u(1) \geqslant 0$. Then

$$\sum_x u(x)\varphi(x) = u(1)\varphi(1) + (u(1) + h)(1 - \varphi(1)) = u(1) + h(1 - \varphi(1)).$$

Similarly, $\sum_x u(x)\psi(x) = u(1) + h(1 - \psi(1))$. Since $h \geqslant 0$ and $\psi(1) \leqslant \varphi(1)$, we have $\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x)$. Thus, $\varphi \preceq_{\mathrm{F}} \psi$. This chain of implications proves the equivalences in the exercise.

Solution to Exercise 3.2.4. Suppose $f, g, h \in \mathscr{D}(\mathsf{X})$ with $f \preceq_{\mathrm{F}} g$ and $g \preceq_{\mathrm{F}} h$. Fixing $u \in i\mathbb{R}^{\mathsf{X}}$, we have

$$\sum_x u(x)f(x) \leqslant \sum_x u(x)g(x) \quad \text{and} \quad \sum_x u(x)g(x) \leqslant \sum_x u(x)h(x)$$

Hence $\sum_x u(x)f(x) \leqslant \sum_x u(x)h(x)$. Since u was arbitrary in $i\mathbb{R}^{\mathsf{X}}$, we are done.

Solution to Exercise 3.2.5. Let $F^\varphi := 1 - G^\varphi$ be the CDF of φ and let F^ψ be the CDF of ψ. In view of Lemma 3.2.1, we have $F^\psi \leqslant F^\varphi$. As a consequence,

$$\{x \in \mathsf{X} : F^\psi(x) \geqslant \tau\} \subset \{x \in \mathsf{X} : F^\varphi(x) \geqslant \tau\}.$$

It follows directly that

$$\min\{x \in \mathsf{X} : F^\varphi(x) \geqslant \tau\} \leqslant \min\{x \in \mathsf{X} : F^\psi(x) \geqslant \tau\}.$$

That is, $Q_\tau(X) \leqslant Q_\tau(Y)$.

Solution to Exercise 3.2.6. Using Exercise 3.1.10 and the definition of P, it can

be shown that

$$G(x, x_k) := \sum_{j=k}^{n} P(x, x_j) = \mathbb{P}\{x_k - s/2 < X_{t+1} \mid X_t = x\}.$$

Rewriting the probability in terms of ε_{t+1}, we get

$$G(x, x_k) = \mathbb{P}\{\varepsilon_{t+1} > (x_k - s/2 - \rho x)/\sigma\}.$$

Since $\rho \geqslant 0$, we can now see that $x \leqslant y$ implies $G(x, x_k) \leqslant G(y, x_k)$ for all k, or, equivalently, $G(x, \cdot) \leqslant G(y, \cdot)$ pointwise on X. By Lemma 3.2.1, this is equivalent to the statement that $P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot)$, which confirms that P is monotone increasing.

Solution to Exercise 3.2.7. This matrix P_w is monotone increasing if and only if $(1 - \alpha, \alpha) \preceq_{\mathrm{F}} (\beta, 1 - \beta)$. From Exercise 3.2.3, we know that this is equivalent to $\beta \leqslant 1 - \alpha$, or $\beta + \alpha \leqslant 1$.

Solution to Exercise 3.2.8. Suppose that P is monotone increasing and fix $h \in i\mathbb{R}^{\mathsf{X}}$. We claim that $Ph \in i\mathbb{R}^{\mathsf{X}}$. To see this, pick any $x, y \in \mathsf{X}$ with $x \preceq y$. Since $x \preceq y$ we have $P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot)$. Hence $\sum_{x'} h(x')P(x, x') \leqslant \sum_{x'} h(x')P(y, x')$. This shows that $Ph \in i\mathbb{R}^{\mathsf{X}}$.

To see the converse, suppose that P is invariant on $i\mathbb{R}^{\mathsf{X}}$. Fix $x, y \in \mathsf{X}$ with $x \preceq y$. We claim that $P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot)$. To see this, fix $u \in i\mathbb{R}^{\mathsf{X}}$. $Pu \in i\mathbb{R}^{\mathsf{X}}$ by invariance, so $(Pu)(x) \leqslant (Pu)(y)$ and hence $\sum_{x'} u(x')P(x, x') \leqslant \sum_{x'} u(x')P(y, x')$. Since u was chosen arbitrarily from $i\mathbb{R}^{\mathsf{X}}$, we have $P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot)$. Hence P is monotone increasing, as was to be shown.

Solution to Exercise 3.2.9. Clearly this is true for $t = 1$. Suppose it is also true for arbitrary t. Then, for any $h \in i\mathbb{R}^{\mathsf{X}}$, the function $P^t h$ is again in $i\mathbb{R}^{\mathsf{X}}$. From this it follow that $P^{t+1}h = PP^t h$ is also in $i\mathbb{R}^{\mathsf{X}}$, since P is monotone increasing. This proves that P^{t+1} is invariant on $i\mathbb{R}^{\mathsf{X}}$, and therefore monotone increasing.

Solution to Exercise 3.2.10. Let π and P satisfy the stated conditions. By Exercise 3.2.9, P^t is monotone increasing for all t. By this fact and the assumption $\pi \in i\mathbb{R}^{\mathsf{X}}$, we see that $P^t\pi \in i\mathbb{R}^{\mathsf{X}}$ for all t. Hence $v = \sum_{t \geqslant 0} \beta^t P^t \pi$ is also increasing.

Solution to Exercise 3.2.12. Both u and $\exp$ are increasing on X, so r is in $i\mathbb{R}^{\mathsf{X}}$. Since $\rho \geqslant 0$, P is monotone increasing (see §3.2.2.3). Clearly βP shares this property. It follows that $\beta P r \in i\mathbb{R}^{\mathsf{X}}$. Applying βP again, we have $(\beta P)^2 r \in i\mathbb{R}^{\mathsf{X}}$. Continuing in

this way, we see that $(\beta P)^k r$ is increasing for all k. Hence $\sum_{k \geqslant 0} (\beta P)^k r$ is increasing. By the Neumann series lemma, this sum is equal to v, so $v \in i\mathbb{R}^{\mathsf{X}}$.

Solution to Exercise 3.3.1. We start with part (i). To show that T is a self-map on $V := \mathbb{R}_+^{\mathsf{W}}$, we just need to verify that $v \in V$ implies $Tv \in V$, which only requires us to verify that T maps nonnegative functions into nonnegative functions. This is clear from the definition. Regarding the order-preserving property, fix $f, g \in V$ with $f \leqslant g$. We claim that $Tf \leqslant Tg$. Indeed, if $w \in \mathsf{W}$, then $\sum_{w' \in \mathsf{W}} f(w')P(w, w') \leqslant \sum_{w' \in \mathsf{W}} g(w')P(w, w')$, which in turn implies that $(Tf)(w) \leqslant (Tg)(w)$. Since w was an arbitrary wage value, we have $Tf \leqslant Tg$, so T is order preserving.

Regarding part (ii), let $e(w) := w/(1 - \beta)$ and fix f, g in V. Writing the operators pointwise and applying the last result in Lemma 2.1.1 (page 45) gives

$$\begin{aligned} |Tf - Tg| &= |e \vee (c + \beta Pf) - e \vee (c + \beta Pg)| \\ &\leqslant |\beta Pf - \beta Pg| \\ &= \beta \, |P(f - g)| \\ &\leqslant \beta P \, |f - g| \, . \end{aligned}$$

(Here the last inequality uses the result in Exercise 2.1.8 on page 45.) Since $P \geqslant 0$ we have $P|f - g| \leqslant P\|f - g\|_\infty \mathbb{1} = \|f - g\|_\infty \mathbb{1}$, so

$$|Tf - Tg| \leqslant \beta \|f - g\|_\infty \mathbb{1}.$$

Taking the maximum on both sides gives $\|Tf - Tg\|_\infty \leqslant \beta \|f - g\|_\infty$. Since f, g were arbitrary elements of V, the contraction claim is verified.

Solution to Exercise 3.3.2. The code in Listing 10 creates a Markov chain via Tauchen approximation of an AR(1) process with positive autocorrelation parameter. By Exercise 3.2.6, P is monotone increasing. Hence, by Lemma 3.3.1, the value function is increasing. Since $h^* = c + \beta P v^*$, it follows that h^* is increasing. Regarding intuition, positive autocorrelation in wages means that high current wages predict high future wages. It follows that the value of waiting for future wages rises with current wages.

Solution to Exercise 3.3.5. Let T be the operator on V such that $(Tv_u)(w)$ is the right-hand side of (3.28). To solve the exercise, it suffices to prove that T is a contraction map on V. (Then v_u can be obtained, in the limit, by applying successive approximation to T and, once the approximate fixed point is computed, v_e can be obtained via (3.27).) To show that T is a contraction, we let T_1 and T_2 be the operators

on V defined by

$$(T_1 v)(w) = \frac{1}{1-\beta(1-\alpha)}\,(w + \alpha\beta(Pv)(w)) \quad \text{and} \quad (T_2 v)(w) = c + \beta\,(Pv)(w).$$

Since $Tv = (T_1 v) \vee (T_2 v)$, Exercise 1.2.21 on page 23 tells us that T will be a contraction provided that T_1 and T_2 are both contraction maps. For the case of T_2, we have

$$\|T_2 f - T_2 g\|_\infty = \max_w |c + \beta\,(Pf)(w) - c - \beta\,(Pg)(w)| \leqslant \max_w \beta \sum_{w'} |f(w') - g(w')| P(w, w').$$

The last term is dominated by $\beta\|f - g\|_\infty$, so T_2 is a contraction. The proof for T_1 is similar in spirit and hence left to the reader.

Solution to Exercise 4.1.1. Pointwise on X we have $1 - \sigma \leqslant 1$, so $P_\sigma \leqslant P$. By Exercise 2.1.16 on page 48, we then have $r(P_\sigma) \leqslant r(P) = 1$. Hence $r(\beta P_\sigma) = \beta\, r(P_\sigma) \leqslant \beta < 1$.

Solution to Exercise 4.1.2. Fix $\sigma \in \Sigma$. If $f, g \in \mathbb{R}^{\mathsf{X}}$, $f \leqslant g$ and $x \in \mathsf{X}$, then

$$\begin{aligned}(T_\sigma g)(x) - (T_\sigma f)(x) &= (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} g(x') P(x, x') - \beta \sum_{x' \in \mathsf{X}} f(x') P(x, x') \\ &= (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} (g(x') - f(x')) P(x, x').\end{aligned}$$

Since $g(x') \geqslant f(x')$ for all x' this expression is nonnegative. Hence $(T_\sigma g)(x) \geqslant (T_\sigma f)(x)$ for all x.

Solution to Exercise 4.1.3. Fix $\sigma \in \Sigma$. Given $f, g \in \mathbb{R}^{\mathsf{X}}$ and $x \in \mathsf{X}$, we have

$$\begin{aligned}|(T_\sigma f)(x) - (T_\sigma g)(x)| &= \left| (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} (g(x') - f(x')) P(x, x') \right| \\ &\leqslant \beta \left| \sum_{x'} [f(x') - g(x')] P(x, x') \right|.\end{aligned}$$

Applying the triangle inequality and $\sum_{x' \in \mathsf{X}} P(x, x') = 1$, we obtain

$$|(T_\sigma f)(x) - (T_\sigma g)(x)| \leqslant \beta \sum_{x'} |f(x') - g(x')| P(x, x') \leqslant \beta \|f - g\|_\infty.$$

Taking the supremum over all x on the left hand side of this expression leads to

$$\|T_\sigma f - T_\sigma g\|_\infty \leqslant \beta \|f - g\|_\infty.$$

Since f, g were arbitrary elements of $\mathbb{R}^{\mathsf{X}}$, the contraction claim is proved.

Solution to Exercise 4.1.4. Fix $f, g \in \mathbb{R}^{\mathsf{X}}$ with $f \leqslant g$. Since $P \geqslant 0$, we have $Pf \leqslant Pg$. Hence $c + \beta Pf \leqslant c + \beta Pg$. As a result,

$$Tf = e \vee (c + \beta Pf) \leqslant e \vee (c + \beta Pg) = Tg.$$

Solution to Exercise 4.1.5. Take any f, g in $\mathbb{R}^{\mathsf{X}}$. Writing the operators pointwise and applying the last result in Lemma 2.1.1 (page 45) gives

$$\begin{aligned} |Tf - Tg| &= |e \vee (c + \beta Pf) - e \vee (c + \beta Pg)| \\ &\leqslant |\beta Pf - \beta Pg| \\ &= \beta \, |P(f - g)| \\ &\leqslant \beta P \, |f - g| \, . \end{aligned}$$

(Here the last inequality uses the result in Exercise 2.1.8 on page 45.) Since $P \geqslant 0$ we have $P|f - g| \leqslant P\|f - g\|_\infty \mathbb{1} = \|f - g\|_\infty \mathbb{1}$, so

$$|Tf - Tg| \leqslant \beta \|f - g\|_\infty \mathbb{1}.$$

Taking the maximum on both sides gives $\|Tf - Tg\|_\infty \leqslant \beta \|f - g\|_\infty$. Since f, g were arbitrary elements of $\mathbb{R}^{\mathsf{X}}$, the contraction claim is verified.

Solution to Exercise 4.1.7. First observe that, since $v^* \geqslant w$ and T is order preserving, we have $v^* = Tv^* \geqslant Tw = s \vee (\pi + \beta Qw) = s \vee w$. From this we get $v^* \geqslant s \vee w$ and applying T to both sides gives $v^* \geqslant T(s \vee w)$.

Next, observe that

$$T(s \vee w) = s \vee (\pi + \beta Q(s \vee w)) \geqslant \pi + \beta Q(s \vee w) \gg \pi + \beta Qw = w$$

where the strict inequality is by Exercise 2.1.9 on page 45. We conclude that $v^* \geqslant T(s \vee w) \gg w$, as was to be shown.

Intuitively, the option to exit adds value to firms everywhere in the state space, since $Q \gg 0$ implies that the state can shift to a region of the state space where exit is optimal in a later period.

Solution to Exercise 4.1.8. For the model described, the Bellman equation takes the form

$$v(p) = \max\left\{s,\ \max_{\ell \geqslant 0} \pi(\ell, p) + \beta \sum_{p'} v(p')Q(p, p')\right\}.$$

Straightforward calculus shows that maximized one-period profits are $\pi(p) = p^2/(4w)$. Hence the final expression is

$$v(p) = \max\left\{s,\ \frac{p^2}{4w} + \beta \sum_{p'} v(p')Q(p, p')\right\}$$

Solution to Exercise 4.1.9. Fix $x, x' \in \mathsf{X}$ with $x \leqslant x'$. Since σ^* is binary, to show σ^* is decreasing it suffices to show that $\sigma^*(x) = 0$ implies $\sigma^*(x') = 0$. Hence we suppose that $\sigma^*(x) = 0$. This in turn implies that $e(x) < h^*(x)$. As $x \leqslant x'$, e is decreasing and h^* is increasing on X, we have $e(x') < h^*(x')$. Hence $\sigma^*(x') = 0$. We conclude that σ^* is decreasing on X, as claimed.

Solution to Exercise 4.1.11. The solution to Exercise 4.1.11 is similar to that of Exercise 4.1.9 and hence omitted.

Solution to Exercise 4.1.12. Either by manipulating the Bellman equation or appealing to (4.16) on page 116, we see that the continuation value operator is defined at $h \in \mathbb{R}^{\mathsf{Z}}$ by

$$(Ch)(z) = \pi(z) + \beta \sum_{z'} \int \max\{s', h(z')\}\varphi(s')\,\mathrm{d}s' Q(z, z') \qquad (z \in \mathsf{Z}).$$

The next period scrap value S_{t+1} is integrated out and the remaining function depends only on $z \in \mathsf{Z}$.

Solution to Exercise 4.1.13. Let φ_a and φ_b be as stated. For $i \in \{a, b\}$ and $h \in \mathbb{R}^{\mathsf{Z}}$, let

$$(C_i h)(z) = \pi(z) + \beta \sum_{z'} \int \max\{s', h(z')\}\varphi_i(s')\,\mathrm{d}s' Q(z, z').$$

Since, for each $z' \in \mathsf{Z}$, the function $s' \mapsto \max\{s', h(z')\}$ is increasing, we have

$$\sum_{z'} \int \max\{s', h(z')\}\varphi_a(s')\,\mathrm{d}s' Q(z, z') \leqslant \sum_{z'} \int \max\{s', h(z')\}\varphi_b(s')\,\mathrm{d}s' Q(z, z').$$

Hence $C_a h \leqslant C_b h$ for all $h \in \mathbb{R}^{\mathsf{Z}}$. As C_b is order-preserving and globally stable, Proposition 2.1.3 on page 52 implies that the fixed point of C_b dominates the fixed point of C_a. That is, $h_a^* \leqslant h_b^*$. But then, for any $z \in \mathsf{Z}$, we have $h_a^*(z) \leqslant h_b^*(z)$ and hence

$$\sigma_b^*(z) = \mathbb{1}\{s \geqslant h_b^*(z)\} \leqslant \mathbb{1}\{s \geqslant h_a^*(z)\} = \sigma_a^*(z).$$

The interpretation of $\sigma_b^* \leqslant \sigma_a^*$ is that firm exits at fewer states when the scrap value distribution is φ_b^*. This makes sense, since the current scrap value offer s is already known, while future offers are more promising under φ_b^* than φ_a^*. Hence continuing is more attractive.

Solution to Exercise 4.2.1. In view of (4.14), the continuation value operator for this problem is

$$(Ch)(x) = -c + \beta \sum_{x'} \max\{\pi(x'), h(x')\} P(x, x') \qquad (x \in \mathsf{X}).$$

The monotonicity result for h^* follows from Lemma 4.1.5 on page 111.

Solution to Exercise 4.2.2. If (X_t) is IID with common distribution φ, then the continuation value h^* is constant; in particular, it is the unique solution to

$$h = -c + \beta \sum_{x'} \max\{\pi(x'), h(x')\} \varphi(x').$$

Since constant functions are (weakly) decreasing, Exercise 4.1.11 applies and σ^* is increasing. Intuitively, the value of waiting is independent of the current state, while the value of bringing the product to market is increasing in the current state. Hence, if the firm brings to the product to market in state x, then it should also do so at any $x' \geqslant x$.

Solution to Exercise 5.1.2. The stochastic kernel is

$$P(x, a, y) = \begin{cases} 0 & \text{if } y < a \\ (1-p)^x & \text{if } y = a \\ (1-p)^{x+a-y} p & \text{if } y > a \end{cases} \tag{5.10}$$

The middle case is obtained by observing that the next period state hits y when $y = a$ if and only if $D_{t+1} \geqslant x$ and then using the expression for the PMF of the geometric distribution.

Solution to Exercise 5.1.3. T is a sup norm contraction mapping on $\mathbb{R}^{\mathsf{X}}$ because, in view of the max-inequality lemma (page 31), for any v, w in $\mathbb{R}^{\mathsf{X}}$,

$$\begin{aligned} |(Tv)(x)| - (Tw)(x)| &\leqslant \beta \max_{a \in \Gamma(x)} \left| \sum_{d \geqslant 0} [v(m(x-d)+a) - w(m(x-d)+a)] \, \varphi(d) \right| \\ &\leqslant \beta \max_{a \in \Gamma(x)} \sum_{d \geqslant 0} |v(m(x-d)+a) - w(m(x-d)+a)| \, \varphi(d) \end{aligned}$$

Since $\sum_{d \geqslant 0} \varphi(d) = 1$, it follows that, for arbitrary $x \in \mathsf{X}$,

$$|(Tv)(x) - (Tw)(x)| \leqslant \beta \|v - w\|_\infty$$

Taking the supremum over all $x \in \mathsf{X}$ yields the desired result.

Solution to Exercise 5.1.4. We take the action A_t to be the choice of next period wealth W_{t+1}, so that the action space is also W. The feasible correspondence is

$$\Gamma(w) = \{a \in \mathsf{W} : a \leqslant Rw\} \qquad (w \in \mathsf{W}),$$

implying that $\mathsf{G} = \{(w, a) \in \mathsf{W} \times \mathsf{W} : a \leqslant Rw\}$. The current reward is utility of consumption, or

$$r(w, a) = u\left(w - \frac{a}{R}\right) \qquad ((w, a) \in \mathsf{G}).$$

The stochastic kernel is $P(w, a, w') = \mathbb{1}\{w' = a\}$. This just states that next period wealth w' is equal to the action a with probability one.

Solution to Exercise 5.1.5. To impose that workers never leave the firm, we require $a \geqslant e$. Thus, the feasible correspondence is

$$\Gamma(x) = \Gamma(e, w) = \{a \in \{0, 1\} : a \geqslant e\}.$$

The set of feasible state-action pairs is $\mathsf{G} = \{((e, w), a) \in \mathsf{X} \times \mathsf{A} : a \geqslant e\}$. The reward function is

$$r(x, a) = r((e, w), a) = aw + (1 - a)c.$$

Regarding the stochastic kernel, we need to define state transition probabilities for all feasible state-action pairs. Letting $P[(e, w), a, (e', w')]$ be the probability of transitioning to state (e', w') given current state (e, w) and current action $a \leqslant e$, we set

$$P[(0, w), a, (e', w')] = \mathbb{1}\{e' = a\} \cdot [\, a\mathbb{1}\{w' = w\} + (1 - a)Q(w, w') \,] \tag{5.14}$$

and $P[(1, w), 1, (e', w')] = \mathbb{1}\{e' = 1\}\mathbb{1}\{w' = w\}$. Equation (5.14) says that if $a = 0$ then

$e' = 0$ and the next wage is drawn from $Q(w, w')$, while if $a = 1$ then $e' = 1$ and the next wage is w. You can verify that P is a stochastic kernel from G to X.

To double check that these definitions work, we can verify that they lead to the same Bellman equations that we saw in §3.3.1. Under the definitions of Γ, r and P just provided, we have $v(1, w) = w + \beta \mathbb{E} v(1, w)$. This implies that $v(1, w) = w/(1 - \beta)$, which is what we expect for lifetime value of an agent employed with wage w.

Moreover, the Bellman equation for $v(0, w)$ agrees with the one we obtained for an unemployed agent on page 94. To see this when $e = 0$, observe that the Bellman equation is

$$
\begin{aligned}
v(0, w) &= \max_{a \in \{0,1\}} \left\{ aw + (1 - a)c + \beta \sum_{(e', w')} v(e', w') P[(0, w), a, (e', w')] \right\} \\
&= \max_{a \in \{0,1\}} \left\{ aw + (1 - a)c + \beta \left[av(a, w) + (1 - a) \sum_{w'} v(a, w') Q(w, w') \right] \right\},
\end{aligned}
$$

where the second equation follows from (5.14). (You can see this by checking the cases $a = 0$ and $a = 1$.) Rearranging and using $v(1, w) = w/(1 - \beta)$ now gives

$$
v(0, w) = \max \left\{ \frac{w}{1 - \beta}, \; c + \beta \sum_{w' \in \mathsf{W}} v(0, w') Q(w', w') \right\}. \tag{5.15}
$$

This is the Bellman equation for an unemployed agent from the job search problem we saw previously on page 94.

Solution to Exercise 5.1.6. Fix $\sigma \in \Sigma$. It is obvious that T_σ is a self-map on $\mathbb{R}^{\mathsf{X}}$ and T_σ is clearly order-preserving, since $v \leqslant w$ implies $P_\sigma v \leqslant P_\sigma w$ and hence $T_\sigma v \leqslant T_\sigma w$.

Also, T_σ is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ under the supremum norm, since, for any v, w in $\mathbb{R}^{\mathsf{X}}$ we have

$$
\begin{aligned}
|(T_\sigma v)(x) - (T_\sigma w)(x)| &= \beta \left| \sum_y P(x, \sigma(x), y) v(y) - \sum_y P(x, \sigma(x), y) w(y) \right| \\
&\leqslant \sum_y P(x, \sigma(x), y) \beta \; |v(y) - w(y)| \leqslant \beta \|v - w\|_\infty
\end{aligned}
$$

Taking the supremum over all $x \in \mathsf{X}$ yields the desired result. This contraction property combined with Banach's fixed point theorem implies that T_σ has a unique fixed point.

Now suppose that v is the unique fixed point of T_σ. Then $v = r_\sigma + \beta P_\sigma v$. But then $v = (I - \beta P_\sigma)^{-1} r_\sigma$. Hence $v = v_\sigma$. This establishes all claims in the lemma.

Solution to Exercise 5.1.9. Fix $v \in \mathbb{R}^{\mathsf{X}}$. Part (i) follows from the fact that $\Gamma(x)$ is finite and nonempty at each $x \in \mathsf{X}$. Hence we can select an element a_x^* from the argmax in the definition of a v-greedy policy at each x in X. The resulting policy is v-greedy. For part (ii) we need to show that $\sigma \in \Sigma$ is v-greedy if and only if

$$r(x, \sigma(x)) + \beta \sum_{x'} v(x') P(x, \sigma(x), x') = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x'} v(x') P(x, a, x') \right\}$$

for all $x \in \mathsf{X}$. But this immediate from the definition.

Regarding part (iii), it follows from the definitions that $(T_\sigma v)(x) \leqslant (Tv)(x)$ for all $x \in \mathsf{X}$. At the same time, for any v-greedy $\sigma \in \Sigma$, we have $(T_\sigma v)(x) = (Tv)(x)$ for all x. Hence $Tv = \vee_\sigma T_\sigma v$, as was to be shown.

Solution to Exercise 5.1.10. Fix $v, w \in \mathbb{R}^{\mathsf{X}}$ and $x \in \mathsf{X}$. By Exercise 5.1.9 and the max-inequality lemma (page 31), we have

$$\begin{aligned} |(Tv)(x) - (Tw)(x)| &= \left| \max_{\sigma \in \Sigma} (T_\sigma v)(x) - \max_{\sigma \in \Sigma} (T_\sigma w)(x) \right| \\ &\leqslant \max_{\sigma \in \Sigma} |(T_\sigma v)(x) - (T_\sigma w)(x)| = \|T_\sigma v - T_\sigma w\|_\infty. \end{aligned}$$

Applying contractivity of T_σ (Exercise 5.1.6), we get $\|Tv - Tw\|_\infty \leqslant \beta \|v - w\|_\infty$.

Solution to Exercise 5.1.11. Part (iii) of Proposition 5.1.1 implies (iv) because every $v \in \mathbb{R}^{\mathsf{X}}$ has at least one greedy policy (Exercise 5.1.9). In particular, at least one v^*-greedy policy exists.

Solution to Exercise 5.3.3. The Bellman equation becomes

$$v(w, z, \varepsilon) = \max_{w' \leqslant R(w+z+\varepsilon)} \left\{ u\left(w + z + \varepsilon - \frac{w'}{R} \right) + \beta \sum_{z', \varepsilon'} v(w', z', \varepsilon') Q(z, z') \varphi(\varepsilon') \right\}.$$

Both w and w' are constrained to a finite set $\mathsf{W} \subset \mathbb{R}_+$. The expected value function can be expressed as

$$g(z, w') := \sum_{z', \varepsilon'} v(w', z', \varepsilon') Q(z, z') \varphi(\varepsilon'). \tag{5.37}$$

In the remainder of this section, we will say that a savings policy σ is **g-greedy** if

$$\sigma(z, w, \varepsilon) \in \operatorname*{argmax}_{w' \leqslant R(w+z+\varepsilon)} \left\{ u\left(w + z + \varepsilon - \frac{w'}{R}\right) + \beta g(z, w') \right\}.$$

Since it is an MDP, we can see immediately that if we replace v in (5.37) with the value function v^*, then a g-greedy policy will be an optimal one. We can rewrite the Bellman equation in terms of expected value functions via

$$g(z, w') = \sum_{z',\, \varepsilon'} \max_{w'' \leqslant R(w'+z'+\varepsilon')} \left\{ u\left(w' + z' + \varepsilon' - \frac{w''}{R}\right) + \beta g(z', w'') \right\} Q(z, z')\varphi(\varepsilon').$$

Solution to Exercise 6.1.1. Proposition 6.1.2 follows directly from Theorem 6.1.1 when $b(X_{t-1}, X_t) = \beta(X_{t-1})$ and $h = \pi$.

Solution to Exercise 6.2.3. Under a cum-dividend contract, purchasing at t and selling at $t+1$ pays $D_t + \Pi_{t+1}$. Hence, applying the fundamental asset pricing equation, the time t price Π_t of the contract must satisfy

$$\Pi_t = D_t + \mathbb{E}_t M_{t+1} \Pi_{t+1}. \tag{6.15}$$

Proceeding as for the ex-dividend contract, the price conditional on current state x is $\pi(x) = d(x) + \sum_{x'} m(x, x')\pi(x')P(x, x')$. In vector form, this is $\pi = d + A\pi$. Solving out for prices gives $\pi^* = (I - A)^{-1} d$.

Solution to Exercise 6.2.5. We seek a v that solves

$$v(x) = \sum_{x' \in \mathsf{X}} [1 + v(x')]\, A(x, x') \qquad (x, x' \in \mathsf{X}).$$

Treating A as a matrix and v as a column vector, this equation becomes $v = A\mathbb{1} + Av$, where $\mathbb{1}$ is a column vector of ones. By the Neumann series lemma, $r(A) < 1$ implies that this equation has the unique solution $v^* = (I - A)^{-1} A\mathbb{1}$. By the same lemma, v^* has the alternative representation $v^* = \sum_{t \geqslant 0} A^t (A\mathbb{1}) = \sum_{t \geqslant 1} A^t \mathbb{1}$.

Solution to Exercise 6.3.1. Let U be closed in $\mathbb{R}^{\mathsf{X}}$ and let T be a self-map on U such that T^k is a contraction. Let u^* be the unique fixed point of T^k. Fix $\varepsilon > 0$. We can choose m such that $\|T^{mk} T u^* - u^*\| < \varepsilon$. Then

$$\|T T^{mk} u^* - u^*\| = \|T u^* - u^*\| < \varepsilon.$$

Since ε was arbitrary we have $\|Tu^* - u^*\| = 0$, implying that u^* is a fixed point of T. The proof that $T^m u \to u^*$ for any u is left to the reader.

Solution to Exercise 6.3.3. For fixed $v, c \in \mathbb{R}^{\mathsf{X}}$ with $c \geqslant 0$, we have

$$\begin{aligned}(T(v+c))(x) &= \max_{a \in \Gamma(x)} \left\{ r(x,a) + \sum_{x'} [v(x') + c(x')] \beta(x,a,x') P(x,a,x') \right\} \\ &\leqslant (Tv)(x) + \max_{a \in \Gamma(x)} \sum_{x'} c(x') \beta(x,a,x') P(x,a,x')\end{aligned}$$

Applying (6.27) we get $T(v+c) \leqslant Tv + Lc$. Proposition 6.3.3 now implies that T is eventually contracting. Since $\mathbb{R}^{\mathsf{X}}$ is closed (as a subset of itself), we conclude that T is globally stable on $\mathbb{R}^{\mathsf{X}}$.

Solution to Exercise 6.3.5. With this choice of L we have $L \geqslant 0$ and

$$\beta(x,a,x')P(x,a,x') = \beta(z)Q(z,z')R(y,a,y') \leqslant \beta(z)Q(z,z') = L(z,z).$$

This means that the bound (6.27) in Proposition 6.3.4 is achieved by L. If $r(L) < 1$, then all of the conditions in Proposition 6.3.4 hold and hence its implications are valid.

Solution to Exercise 7.1.4. We begin with the concave case. Suppose there exists a $\delta > 0$ such that $T\varphi \geqslant \varphi + \delta(\psi - \varphi)$. Set $\delta' := \delta/(\delta+1)$. Then $\delta' \in (0,1)$ and $\delta' < \delta$, so $T\varphi \geqslant \varphi + \delta'(\psi - \varphi)$. In other words, if the bound holds at arbitrary positive δ, then it holds for some $\delta' \in (0,1)$.

The case of convex T is similar.

Solution to Exercise 7.1.6. It is straightforward to show that $F'_x > 0$ on $(0,\infty)$, which proves that F_x is increasing. Some additional algebra confirms that $\theta \in (0,1]$ implies $F''_x > 0$, while $\theta < 0$ and $\theta > 1$ both imply $F''_x < 0$. Details are left to the reader.

Solution to Exercise 7.1.7. Observe that $(Gu)(x) = F_x[(Au)(x)]$. Since A is order-preserving and F is increasing, $u \leqslant v$ implies $Gu \leqslant Gv$. In particular, G is order-preserving. If $\theta \in (0,1]$, then F is convex. Hence, fixing $u, v \in V$ and $\lambda \in [0,1]$ (and dropping x from our notation), we have

$$FA(\lambda u + (1-\lambda)v) = F(\lambda Au + (1-\lambda)Av) \leqslant \lambda FAu + (1-\lambda)FAv.$$

Hence G is convex. The proof that G is concave for other values of θ is similar and omitted.

Solution to Exercise 7.2.1. Fixing t, rearranging $v^* = (I - \beta P)^{-1} r$ to $v^* = r + \beta P v^*$ and evaluating at X_t gives

$$\begin{aligned} V_t^* = v^*(X_t) &= r(X_t) + \beta \sum_{x'} v^*(x') P(X_t, x') \\ &= u(C_t) + \beta \mathbb{E}[v^*(X_{t+1}) \mid X_t] = u(C_t) + \beta \mathbb{E}_t V_{t+1}^*. \end{aligned}$$

Hence $(V_t^*)_{t \geqslant 0}$ obeys (7.4), as claimed.

Solution to Exercise 7.2.7. If $v \gg 0$, then, since $h \gg 0$ also holds, we have

$$\hat{K} v = \left\{ h + \beta (Pv)^{1/\theta} \right\}^\theta \geqslant h^\theta \gg 0.$$

Hence $\hat{K}$ is a self-map on V. In addition, the statement $Kv = v$ is equivalent to $v^\alpha = h + \beta (Pv^\gamma)^{\alpha/\gamma}$. Using $\theta = \gamma/\alpha$, we can rewrite the last equation as $v^\gamma = [h + \beta (Pv^\gamma)^{1/\theta}]^\theta$. In other words, $v = Kv$ if and only if $v^\gamma = \hat{K} v^\gamma$.

Solution to Exercise 7.2.9. Iterating forward from V_0 gives

$$V_0 = u(C_0) + \beta \mathbb{E}_0 V_1 = u(C_0) + \beta \mathbb{E}_0 [u(C_1) + \beta \mathbb{E}_1 V_2] = u(C_0) + \beta \mathbb{E}_0 u(C_1) + \beta^2 \mathbb{E}_0 V_2.$$

Continuing forward until time m yields $V_0 = \sum_{t=0}^{m-1} \beta^t \mathbb{E}_0 u(C_t) + \beta^m \mathbb{E}_0 V_m$. Shifting to functional form and using $r = u \circ c$, the last expression becomes

$$v = \sum_{t=0}^{m-1} (\beta P)^t r + (\beta P)^m w.$$

By Exercise 1.2.14 on page 21, this is just $K^m w$ when K is the associated Koopmans operator $Kv := r + \beta P v$ and, moreover, $K^m w \to v^* := (I - \beta P)^{-1} r$.

Solution to Exercise 7.2.12. Since V is all of $\mathbb{R}^{\mathsf{X}}$, the condition $R_\tau \colon V \to V$ is trivially satisfied. Regarding monotonicity, fix $v, w \in V$ with $v \leqslant w$ and $x \in \mathsf{X}$. Let X be a draw from $P(x, \cdot)$. Then $(R_\tau v)(x) = Q_\tau[v(X)] \leqslant Q_\tau[w(X)] = R_\tau(v, \varphi)$, where the inequality is by Exercise 3.2.5 on page 89. Moreover, given $\lambda \in \mathbb{R}$ and a random variable Y with $\mathbb{P}\{Y = \lambda\} = 1$, we clearly have $Q_\tau(Y) = \lambda$. It follows that $R_\tau \lambda \mathbb{1} = \lambda \mathbb{1}$. Hence R_τ is a certainty equivalent operator, as was to be shown.

Solution to Exercise 7.2.14. Fix $v \in V = \mathbb{R}^{\mathsf{X}}$, $\lambda \in \mathbb{R}_+$ and $x \in \mathsf{X}$. If $X \sim P(x, \cdot)$, then

$$(R_\tau (v + \lambda))(x) = Q_\tau (v(X) + \lambda) = Q_\tau (v(X)) + \lambda,$$

where the second equality is by Exercise 1.2.25 on page 30. Since x was arbitrary, we have $R_\tau(v + \lambda) = R_\tau v + \lambda$. Hence R_τ is subadditive, as claimed.

Solution to Exercise 7.2.15. Fix $v \in V$, $P \in \mathcal{S}(\mathsf{X})$ and $\lambda \in \mathbb{R}_+$. Let X be a draw from $P(x, \cdot)$. We have

$$\begin{aligned}(R_e(v + \lambda))(x) &= \frac{1}{\theta} \ln \{\mathbb{E} \exp[\theta(v(X) + \lambda)]\} \\ &= \frac{1}{\theta} \ln \{\mathbb{E} \exp[\theta v(X)] \cdot \exp(\theta\lambda)\} \\ &= \frac{1}{\theta} \ln \{\mathbb{E} \exp[\theta v(X)]\} + \lambda.\end{aligned}$$

Hence condition (7.15) holds.

Solution to Exercise 7.2.16. Let the primitives be as stated. Fix $v, w \in V$. By monotonicity and subadditivity of R, we have

$$Rv = R(v - w + w) \leqslant R(\|v - w\|_\infty \mathbb{1} + w) \leqslant Rw + \|v - w\|_\infty \mathbb{1}.$$

Hence $(Rv)(x) - (Rw)(x) \leqslant \|v - w\|_\infty$ for all $x \in \mathsf{X}$. Reversing the roles of v and w proves the claim.

Solution to Exercise 7.2.17. The additive case is obvious. Regarding the Leontief case, fix $x \in \mathsf{X}$, $y \in \mathbb{R}$ and $\lambda \in \mathbb{R}_+$. We have

$$\min\{r(x), \beta(y + \lambda)\} \leqslant \min\{r(x) + \beta\lambda, \beta y + \beta\lambda\} = \min\{r(x), \beta y\} + \beta\lambda.$$

That is, $A(x, y + \lambda) \leqslant A(x, y) + \beta\lambda$. Hence Blackwell's condition holds.

Solution to Exercise 7.2.18. It is not difficult to show that $(U_c/U_y) = ((1 - \beta)/\beta)(y/c)^{1-\alpha}$. Taking logs and rearranging gives $\ln(y/c) = (1/(1 - \alpha)) \ln(U_c/U_y) + k$, where k is a constant. Using the definition in the exercise yields EIS $= 1/(1 - \alpha)$.

Solution to Exercise 7.2.20. We already know from Exercise 7.2.17 that the Leontief aggregator satisfies Blackwell's condition when $\beta \in (0, 1)$. Since R_τ is subadditive, global stability follows from Proposition 7.2.5.

Solution to Exercise 8.1.1. This problem can be formulated as an MDP by setting the state to $x := (y, z)$, taking values in $\mathsf{X} := \mathsf{Y} \times \mathsf{Z}$. The action space is Y. The feasible correspondence is $x \mapsto \Gamma(x)$ and current reward is $r(x, a) = r((y, z), y') = F(y, z, y')$.

The stochastic kernel is

$$P(x, a, x') = P((y, z), a, (y', z')) = \mathbb{1}\{y' = a\}Q(z, z').$$

This MDP (Γ, β, r, P) has a Bellman equation identical to (8.7).

Solution to Exercise 8.1.2. Since $V = \mathbb{R}^{\mathsf{X}}$ is a sublattice of $\mathbb{R}^{\mathsf{X}}$, we only need to check that (Γ, V, B) satisfies the monotonicity condition (8.2) and the consistency condition (8.3). The monotonicity condition holds because β is nonnegative, so $w \leqslant v$ implies

$$\sum_{x'} w(x')\beta(x, a, x')P(x, a, x') \leqslant \sum_{x'} v(x')\beta(x, a, x')P(x, a, x') \quad \text{for all } (x, a) \in \mathsf{G}.$$

The consistency condition (8.3) is trivial because V is all of $\mathbb{R}^{\mathsf{X}}$.

Solution to Exercise 8.1.4. The set V is a sublattice because the pointwise maximum and pointwise minimum of two everywhere positive functions are positive. The consistency condition (8.3) requires that, for any σ, the operator T_σ maps V to itself. This holds because T_σ is a version of the operator K defined in (7.12) on page 219, and K maps V to itself (Exercise 7.2.7 on page 222). The monotonicity condition (8.2) holds by inspection of B in (8.10). Hence (Γ, V, B) is an RDP, as claimed.

Solution to Exercise 8.1.5. Fix $\sigma \in \Sigma$. The claim that T_σ is a self-map on V follows immediately from the consistency condition in (8.3). The order-preserving property follows from the monotonicity condition in (8.2).

Solution to Exercise 8.1.6. Fix $v \in V$ and consider the set $\{T_\sigma v\}_{\sigma\in\Sigma} \subset V$. We first show that $\{T_\sigma v\}_{\sigma\in\Sigma}$ contains a greatest element. Suppose that $\bar{\sigma}$ is v-greedy. If σ is any other policy, then

$$(T_\sigma v)(x) = B(x, \sigma(x), v) \leqslant (T_{\bar{\sigma}} v)(x) \quad \text{for all } x \in \mathsf{X}.$$

Hence $T_{\bar{\sigma}} v$ is a greatest element of $\{T_\sigma v\}_{\sigma\in\Sigma}$.

The proof that $\{T_\sigma v\}_{\sigma\in\Sigma}$ contains a least element is analogous, after replacing argmax with argmin.

Solution to Exercise 8.1.7. Regarding part (i), fix $v \in V$. For any $\sigma \in \Sigma$ and $x \in \mathsf{X}$,

we have

$$(Tv)(x) = \max_{a \in \Gamma(x)} B(x, a, v) = \max_{\sigma \in \Sigma} B(x, \sigma(x), v) = \max_{\sigma \in \Sigma} (T_\sigma v)(x).$$

Since x was chosen arbitrarily, we have confirmed that $Tv = \vee_{\sigma \in \Sigma} T_\sigma v$.

Regarding part (ii), it follows from the definitions that $(T_\sigma v)(x) \leqslant (Tv)(x)$ for all $x \in \mathsf{X}$. At the same time, for any v-greedy $\sigma \in \Sigma$, we have $(T_\sigma v)(x) = (Tv)(x)$ for all x. Hence $Tv = \vee_\sigma T_\sigma v$, as was to be shown.

Solution to Exercise 8.1.8. Here's a proof for T and fixed $k \in \mathbb{N}$: At arbitrary $x \in \mathsf{X}$,

$$(T^k v)(x) = (T(T^{k-1} v))(x) = \max_{a \in \Gamma(x)} B(x, a, T^{k-1} v) \tag{8.19}$$

This confirms the claim in (8.17).

Solution to Exercise 8.1.9. Let v and σ_k be as stated. Fix $\varepsilon > 0$ and pick any $x \in \mathsf{X}$. From the definition of greedy policies and the fact that v^* satisfies the Bellman equation, we have

$$v_{\sigma_k}(x) = \max_{a \in \Gamma(x)} B(x, a, v_k) \quad \text{and} \quad v^*(x) = \max_{a \in \Gamma(x)} B(x, a, v^*).$$

Applying the max-inequality lemma (page 31) yields

$$|v_{\sigma_k}(x) - v^*(x)| \leqslant \max_{a \in \Gamma(x)} |B(x, a, v_k) - B(x, a, v^*)|.$$

Since $\Gamma(x)$ is finite, we can choose $m \in \mathbb{N}$ such that $k \geqslant m$ implies $|B(x, a, v_k) - B(x, a, v^*)| < \varepsilon$ for all $a \in \Gamma(x)$. From this fact and the previous bound we get $|v_{\sigma_k}(x) - v^*(x)| < \varepsilon$ whenever $k \geqslant m$. Hence v_{σ_k} converges to v^* on X.

Solution to Exercise 8.2.3. Let (Γ, V, B) satisfy Blackwell's condition. Fix $v, w \in V$ and $(x, a) \in \mathsf{G}$. Observe that $v = w + v - w \leqslant w + \|v - w\|_\infty$. By monotonicity of B and Blackwell's condition, we have

$$B(x, a, v) \leqslant B(x, a, w + \|v - w\|_\infty) \leqslant B(x, a, w) + \beta \|v - w\|_\infty.$$

As a result, $B(x, a, v) - B(x, a, w) \leqslant \beta \|v - w\|_\infty$. Reversing the roles of v and w yields

$$|B(x, a, v) - B(x, a, w)| \leqslant \beta \|v - w\|_\infty.$$

Since $\beta < 1$, the RDP is contracting.

Solution to Exercise 8.2.11. Let $\mathscr{R} = (\Gamma, V, B)$ be as stated. Fix $v, w \in V$ and $x \in \mathsf{X}$. Using the max-inequality (page 31) and the bounds in the proof of Proposition 8.2.5, we obtain

$$|(Tv)(x) - (Tw)(x)| \leqslant \max_{\sigma \in \Sigma} |(T_\sigma v)(x) - (T_\sigma w)(x)| \leqslant \sum_{x'} |v(x') - w(x')| \, L(x, x').$$

Proposition 6.3.2 on page 194 implies that T is eventually contracting on V. Since V is closed, it follows that T is globally stable (Theorem 6.3.1, page 193).

Solution to Exercise 8.2.12. When (8.33) holds, by finiteness of G, we can take an $\varepsilon > 0$ such that

$$B(x, a, v_2) \leqslant v_2(x) - \varepsilon \text{ for all } (x, a) \in \mathsf{G}.$$

We then have

$$\varepsilon \leqslant v_2(x) - B(x, a, v_2) \leqslant v_2(x) - B(x, a, v_1) \leqslant v_2(x) - v_1(x)$$

for all x, so $0 < \varepsilon \leqslant \|v_2 - v_1\|_\infty$. Set $\delta := \varepsilon / \|v_2 - v_1\|_\infty$. From (8.33) we get

$$B(x, a, v_2) \leqslant v_2(x) - \delta \|v_2 - v_1\|_\infty \leqslant v_2(x) - \delta [v_2(x) - v_1(x)]$$

for arbitrary $(x, a) \in \mathsf{G}$. Hence (8.32) holds.

Solution to Exercise 8.3.2. T is a self-map on V because, at each $v \in V$, there exists a policy σ such that $Tv = T_\sigma v$. Hence $Tv \in V$. Regarding the order-preserving property, take $v, w \in V$ with $v \preceq w$. Since T_σ is order-preserving, we have $T_\sigma v \preceq T_\sigma w$ for all $\sigma \in \Sigma$. Hence $T_\sigma v \preceq Tw$ for all $\sigma \in \Sigma$. By the definition of the supremum, we now have $Tv \preceq Tw$.

Solution to Exercise 8.3.4. For all $v \in V_\Sigma$, we have $v = v_\sigma$ for some σ, and hence $Tv \geqslant T_\sigma v = T_\sigma v_\sigma = v_\sigma = v$.

Solution to Exercise 8.4.3. For part (i), we prove the infimum case. (The supremum case is similar.) Let L, $\hat{L}$ and $\{f_i\}_{i \in I}$ be as stated. On one hand, $f_i \succeq \wedge_i f_i$ for all i, so $\Phi f_i \succeq \Phi \wedge_i f_i$ for all i. Hence $\Phi \wedge_i f_i$ is a lower bound and, therefore, $\wedge_i \Phi f_i \succeq \Phi \wedge_i f_i$. On the other hand, for the reverse inequality $\wedge_i \Phi f_i \preceq \Phi \wedge_i f_i$, we observe that $\wedge_i \Phi f_i \preceq \Phi f_i$ for all i. Since Φ^{-1} is order-preserving, this yields $\Phi^{-1} \wedge_i \Phi f_i \preceq f_i$ for all i. Hence $\Phi^{-1} \wedge_i \Phi f_i \preceq \wedge_i f_i$. Applying Φ to both sides yields $\wedge_i \Phi f_i \preceq \Phi \wedge_i f_i$. The proof of part (i) is now done.

Part (ii) is left to the reader.

Solution to Exercise A.3.2. This statement is false. To see this, let $\mathsf{X} = \mathbb{R}$, let $f_1(x) = x$ and let $f_2(x) = -x$. If $g \geqslant f_i$ for $i = 1, 2$, then $g(x) \geqslant |x|$. But then $g \notin \{f_1, f_2\}$.

Solution to Exercise A.3.3. Suppose that s and s' are both suprema of A in P. Then both s and s' are upper bounds, so $s \preceq s'$ and $s' \preceq s$. Hence $s = s'$.

Bibliography

Açıkgöz, Ö. T. (2018). On the existence and uniqueness of stationary equilibrium in Bewley economies with production. *Journal of Economic Theory*, 173:18–55.

Aiyagari, S. R. (1994). Uninsured idiosyncratic risk and aggregate saving. *The Quarterly Journal of Economics*, 109(3):659–684.

Al-Najjar, N. I. and Shmaya, E. (2019). Recursive utility and parameter uncertainty. *Journal of Economic Theory*, 181:274–288.

Albuquerque, R., Eichenbaum, M., Luo, V. X., and Rebelo, S. (2016). Valuation risk and asset pricing. *The Journal of Finance*, 71(6):2861–2904.

Algan, Y., Allais, O., den Haan, W. J., and Rendahl, P. (2014). Solving and simulating models with heterogeneous agents and aggregate uncertainty. *Handbook of Computational Economics*, 3.

Aliprantis, C. D. and Burkinshaw, O. (1998). *Principles of real analysis*. Academic Press, 3 edition.

Antràs, P. and De Gortari, A. (2020). On the geography of global value chains. *Econometrica*, 88(4):1553–1598.

Arellano, C. (2008). Default risk and income fluctuations in emerging economies. *American Economic Review*, 98(3):690–712.

Arellano, C. and Ramanarayanan, A. (2012). Default and the maturity structure in sovereign bonds. *Journal of Political Economy*, 120(2):187–232.

Arrow, K. J., Harris, T., and Marschak, J. (1951). Optimal inventory policy. *Econometrica: Journal of the Econometric Society*, pages 250–272.

Atkinson, K. and Han, W. (2005). *Theoretical numerical analysis*, volume 39. Springer.

Backus, D. K., Routledge, B. R., and Zin, S. E. (2004). Exotic preferences for macroeconomists. *NBER Macroeconomics Annual*, 19:319–390.

Bagliano, F.-C. and Bertola, G. (2004). *Models for dynamic macroeconomics*. Oxford University Press.

Balbus, Ł. (2020). On recursive utilities with non-affine aggregator and conditional certainty equivalent. *Economic Theory*, 70(2):551–577.

Balbus, Ł., Reffett, K., and Woźny, Ł. (2014). A constructive study of markov equilibria in stochastic games with strategic complementarities. *Journal of Economic Theory*, 150:815–840.

Balbus, Ł., Reffett, K., and Woźny, Ł. (2018). On uniqueness of time-consistent markov policies for quasi-hyperbolic consumers under uncertainty. *Journal of Economic Theory*, 176:293–310.

Balbus, Ł., Reffett, K., and Woźny, Ł. (2022). Time-consistent equilibria in dynamic models with recursive payoffs and behavioral discounting. *Journal of Economic Theory*, page 105493.

Bansal, R., Kiku, D., and Yaron, A. (2012). An empirical evaluation of the long-run risks model for asset prices. *Critical Finance Review*, 1(1):183–221.

Bansal, R. and Yaron, A. (2004). Risks for the long run: A potential resolution of asset pricing puzzles. *The Journal of Finance*, 59(4):1481–1509.

Barbu, V. and Precupanu, T. (2012). *Convexity and optimization in Banach spaces*. Springer Science & Business Media.

Barillas, F., Hansen, L. P., and Sargent, T. J. (2009). Doubts or variability? *journal of economic theory*, 144(6):2388–2418.

Bartle, R. G. and Sherbert, D. R. (2011). *Introduction to real analysis*. Hoboken, NJ: Wiley, 4 edition.

Basu, S. and Bundick, B. (2017). Uncertainty shocks in a model of effective demand. *Econometrica*, 85(3):937–958.

Bäuerle, N. and Jaśkiewicz, A. (2018). Stochastic optimal growth model with risk sensitive preferences. *Journal of Economic Theory*, 173:181–200.

Bäuerle, N. and Rieder, U. (2011). *Markov decision processes with applications to finance*. Springer Science & Business Media.

Bellman, R. (1957). *Dynamic programming*. American Association for the Advancement of Science.

Bellman, R. (1984). *Eye of the Hurricane*. World Scientific.

Benhabib, J., Bisin, A., and Zhu, S. (2015). The wealth distribution in bewley economies with capital income risk. *Journal of Economic Theory*, 159:489–515.

Bertsekas, D. (2012). *Dynamic programming and optimal control: Volume I*, volume 1. Athena scientific.

Bertsekas, D. (2021). *Rollout, policy iteration, and distributed reinforcement learning*. Athena Scientific.

Bertsekas, D. (2022a). Newton's method for reinforcement learning and model predictive control. *Results in Control and Optimization*, 7:100121.

Bertsekas, D. P. (2022b). *Abstract dynamic programming*. Athena Scientific, 3 edition.

Bhandari, J. and Russo, D. (2022). Global optimality guarantees for policy gradient methods. Technical report, arXiv preprint arXiv:1906.01786.

Bianchi, J. (2011). Overborrowing and systemic externalities in the business cycle. *American Economic Review*, 101(7):3400–3426.

Bloise, G., Le Van, C., and Vailakis, Y. (2023). Do not blame bellman: It is koopmans' fault. *Econometrica*, in press.

Bloise, G. and Vailakis, Y. (2018). Convex dynamic programming with (bounded) recursive utility. *Journal of Economic Theory*, 173:118–141.

Bloise, G. and Vailakis, Y. (2022). On sovereign default with time-varying interest rates. *Review of Economic Dynamics*, 44:211–224.

Bloom, N. (2009). The impact of uncertainty shocks. *econometrica*, 77(3):623–685.

Bloom, N., Bond, S., and Van Reenen, J. (2007). Uncertainty and investment dynamics. *The review of economic studies*, 74(2):391–415.

Blundell, R., Graber, M., and Mogstad, M. (2015). Labor income dynamics and the insurance from taxes, transfers, and the family. *Journal of Public Economics*, 127:58–73.

Bocola, L., Bornstein, G., and Dovis, A. (2019). Quantitative sovereign default models and the european debt crisis. *Journal of International Economics*, 118:20–30.

Bommier, A., Kochov, A., and Le Grand, F. (2017). On monotone recursive preferences. *Econometrica*, 85(5):1433–1466.

Bommier, A., Kochov, A., and Le Grand, F. (2019). Ambiguity and endogenous discounting. *Journal of Mathematical Economics*, 83:48–62.

Bond, S. and Van Reenen, J. (2007). Microeconometric models of investment and employment. In *Handbook of econometrics*, volume 6, pages 4417–4498. Elsevier.

Borovička, J. and Stachurski, J. (2020). Necessary and sufficient conditions for existence and uniqueness of recursive utilities. *The Journal of Finance*.

Boyd, J. H. (1990). Recursive utility and the ramsey problem. *Journal of Economic Theory*, 50(2):326–345.

Brémaud, P. (2020). *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*, volume 31. Springer Nature.

Brock, W. A. and Mirman, L. J. (1972). Optimal economic growth and uncertainty: The discounted case. *Journal of Economic Theory*, 4(3):479–513.

Burdett, K. (1978). A theory of employee job search and quit rates. *The American Economic Review*, 68(1):212–220.

Cagetti, M., Hansen, L. P., Sargent, T. J., and Williams, N. (2002). Robustness and pricing with uncertain growth. *The Review of Financial Studies*, 15(2):363–404.

Calsamiglia, C., Fu, C., and Güell, M. (2020). Structural estimation of a model of school choices: The boston mechanism versus its alternatives. *Journal of Political Economy*, 128(2):642–680.

Campbell, J. Y. (2017). *Financial decisions and markets: a course in asset pricing*. Princeton University Press.

Cao, D. (2020). Recursive equilibrium in Krusell and Smith (1998). *Journal of Economic Theory*, 186.

Cao, D. and Werning, I. (2018). Saving and dissaving with hyperbolic discounting. *Econometrica*, 86(3):805–857.

Carroll, C. D. (1997). Buffer-stock saving and the life cycle/permanent income hypothesis. *Quarterly Journal of Economics*, 112(1):1–55.

Carroll, C. D. (2009). Precautionary saving and the marginal propensity to consume out of permanent income. *Journal of monetary Economics*, 56(6):780–790.

Carruth, A., Dickerson, A., and Henley, A. (2000). What do we know about investment under uncertainty? *Journal of economic surveys*, 14(2):119–154.

Carvalho, V. M. and Grassi, B. (2019). Large firm dynamics and the business cycle. *American Economic Review*, 109(4):1375–1425.

Chatterjee, S. and Eyigungor, B. (2012). Maturity, indebtedness, and default risk. *American Economic Review*, 102(6):2674–99.

Cheney, W. (2013). *Analysis for applied mathematics*, volume 208. Springer Science & Business Media.

Chetty, R. (2008). Moral hazard versus liquidity and optimal unemployment insurance. *Journal of political Economy*, 116(2):173–234.

Christensen, T. M. (2022). Existence and uniqueness of recursive utilities without boundedness. *Journal of Economic Theory*, page 105413.

Christiano, L. J., Motto, R., and Rostagno, M. (2014). Risk shocks. *American Economic Review*, 104(1):27–65.

Cochrane, J. H. (2009). *Asset pricing: Revised edition*. Princeton University Press.

Cruces, J. J. and Trebesch, C. (2013). Sovereign defaults: The price of haircuts. *American economic Journal: macroeconomics*, 5(3):85–117.

Daley, D. (1968). Stochastically monotone markov chains. *Probability Theory and Related Fields*, 10(4):305–317.

Dasgupta, P. and Maskin, E. (2005). Uncertainty and hyperbolic discounting. *American Economic Review*, 95(4):1290–1299.

de Castro, L. and Galvao, A. F. (2019). Dynamic quantile models of rational behavior. *Econometrica*, 87(6):1893–1939.

de Castro, L. and Galvao, A. F. (2022). Static and dynamic quantile preferences. *Economic Theory*, 73(2-3):747–779.

de Castro, L. I., Galvao, A. F., and Nunes, D. (2022). Dynamic economics with quantile preferences. *Available at SSRN 4108230*.

de Groot, O., Richter, A. W., and Throckmorton, N. A. (2018). Uncertainty shocks in a model of effective demand: Comment. *Econometrica*, 86(4):1513–1526.

De Groot, O., Richter, A. W., and Throckmorton, N. A. (2022). Valuation risk revalued. *Quantitative Economics*, 13(2):723–759.

De Nardi, M., French, E., and Jones, J. B. (2010). Why do the elderly save? the role of medical expenses. *Journal of political economy*, 118(1):39–75.

Deaton, A. and Laroque, G. (1992). On the behaviour of commodity prices. *Review of Economic Studies*, 59(1):1–23.

DeJarnette, P., Dillenberger, D., Gottlieb, D., and Ortoleva, P. (2020). Time lotteries and stochastic impatience. *Econometrica*, 88(2):619–656.

Denardo, E. V. (1967). Contraction mappings in the theory underlying dynamic programming. *Siam Review*, 9(2):165–177.

Denardo, E. V. (1981). *Dynamic Programming: Models and Applications*. Prentice Hall PTR.

Denardo, E. V. and Mitten, L. (1967). Elements of sequential decision processes. *Journal of Industrial Engineering*, 18:106–112.

Diamond, P. and Köszegi, B. (2003). Quasi-hyperbolic discounting and retirement. *Journal of Public Economics*, 87(9-10):1839–1872.

Dixit, R. K. and Pindyck, R. S. (2012). *Investment under uncertainty*. Princeton University Press.

Drugeon, J.-P. and Wigniolle, B. (2021). On markovian collective choice with heterogeneous quasi-hyperbolic discounting. *Economic Theory*, 72(4):1257–1296.

Du, Y. (1990). Fixed points of increasing operators in ordered banach spaces and applications. *Applicable Analysis*, 38(01-02):1–20.

Duffie, D. (2010). *Dynamic asset pricing theory*. Princeton University Press.

Dvoretzky, A., Kiefer, J., and Wolfowitz, J. (1952). The inventory problem: I. case of known distributions of demand. *Econometrica*, pages 187–222.

Epstein, L. G. and Zin, S. E. (1989). Risk aversion and the temporal behavior of consumption and asset returns: A theoretical framework. *Econometrica*, 57(4):937–969.

Epstein, L. G. and Zin, S. E. (1991). Substitution, risk aversion, and the temporal behavior of consumption and asset returns: An empirical analysis. *Journal of Political Economy*, 99(2):263–286.

Ericson, R. and Pakes, A. (1995). Markov-perfect industry dynamics: A framework for empirical work. *The Review of Economic Studies*, 62(1):53–82.

Erosa, A. and González, B. (2019). Taxation and the life cycle of firms. *Journal of Monetary Economics*, 105:114–130.

Fagereng, A., Holm, M. B., Moll, B., and Natvik, G. (2019). Saving behavior across the wealth distribution: The importance of capital gains. Technical report, National Bureau of Economic Research.

Fajgelbaum, P. D., Schaal, E., and Taschereau-Dumouchel, M. (2017). Uncertainty traps. *The Quarterly Journal of Economics*, 132(4):1641–1692.

Farmer, L. E. and Toda, A. A. (2017). Discretizing nonlinear, non-gaussian markov processes with exact conditional moments. *Quantitative Economics*, 8(2):651–683.

Foss, S., Shneer, V., Thomas, J. P., and Worrall, T. (2018). Stochastic stability of monotone economies in regenerative environments. *Journal of Economic Theory*, 173:334–360.

Frederick, S., Loewenstein, G., and O'donoghue, T. (2002). Time discounting and time preference: A critical review. *Journal of economic literature*, 40(2):351–401.

Gao, Y., Lui, K. Y. C., and Hernandez-Leal, P. (2021). Robust risk-sensitive reinforcement learning agents for trading markets. Technical report, arXiv preprint arXiv:2107.08083.

Gennaioli, N., Martin, A., and Rossi, S. (2014). Sovereign default, domestic banks, and financial institutions. *The Journal of Finance*, 69(2):819–866.

Gentry, M. L., Hubbard, T. P., Nekipelov, D., and Paarsch, H. J. (2018). Structural econometrics of auctions: a review. *Foundations and Trends in Econometrics*, 9(2-4):79–302.

Ghosh, A. R., Kim, J. I., Mendoza, E. G., Ostry, J. D., and Qureshi, M. S. (2013). Fiscal fatigue, fiscal space and debt sustainability in advanced economies. *The Economic Journal*, 123(566):F4–F30.

Gillingham, K., Iskhakov, F., Munk-Nielsen, A., Rust, J., and Schjerning, B. (2022). Equilibrium Trade in Automobiles. *Journal of Political Economy*.

Giovannetti, B. C. (2013). Asset pricing under quantile utility maximization. *Review of Financial Economics*, 22(4):169–179.

Guo, D., Cho, Y. J., and Zhu, J. (2004). *Partial ordering methods in nonlinear problems*. Nova Publishers.

Guvenen, F. (2007). Learning your earning: Are labor income shocks really very persistent? *American Economic Review*, 97(3):687–712.

Guvenen, F. (2009). An empirical investigation of labor income processes. *Review of Economic dynamics*, 12(1):58–79.

Häggström, O. et al. (2002). *Finite Markov chains and algorithmic applications*. Cambridge University Press.

Hansen, L. P., Heaton, J. C., and Li, N. (2008). Consumption strikes back? measuring long-run risk. *Journal of Political economy*, 116(2):260–302.

Hansen, L. P. and Miao, J. (2018). Aversion to ambiguity and model misspecification in dynamic stochastic environments. *Proceedings of the National Academy of Sciences*, 115(37):9163–9168.

Hansen, L. P. and Renault, E. (2010). Pricing kernels. *Encyclopedia of Quantitative Finance*.

Hansen, L. P. and Sargent, T. J. (1980). Formulating and estimating dynamic linear rational expectations models. *Journal of Economic Dynamics and control*, 2:7–46.

Hansen, L. P. and Sargent, T. J. (1990). *Rational expectations econometrics*. CRC Press.

Hansen, L. P. and Sargent, T. J. (1995). Discounted linear exponential quadratic gaussian control. *IEEE Transactions on Automatic control*, 40(5):968–971.

Hansen, L. P. and Sargent, T. J. (2007). Recursive robust estimation and control without commitment. *Journal of Economic Theory*, 136(1):1–27.

Hansen, L. P. and Sargent, T. J. (2011). *Robustness*. Princeton university press.

Hansen, L. P. and Sargent, T. J. (2020). Structured ambiguity and model misspecification. *Journal of Economic Theory*, page 105165.

Hansen, L. P. and Scheinkman, J. A. (2009). Long-term risk: An operator approach. *Econometrica*, 77(1):177–234.

Harrison, J. M. and Kreps, D. M. (1978). Speculative investor behavior in a stock market with heterogeneous expectations. *The Quarterly Journal of Economics*, 92(2):323–336.

Hassett, K. A. and Hubbard, R. G. (2002). Tax policy and business investment. In *Handbook of public economics*, volume 3, pages 1293–1343. Elsevier.

Havranek, T., Horvath, R., Irsova, Z., and Rusnak, M. (2015). Cross-country heterogeneity in intertemporal substitution. *Journal of International Economics*, 96(1):100–118.

Hayashi, F. (1982). Tobin's marginal q and average q: A neoclassical interpretation. *Econometrica: Journal of the Econometric Society*, pages 213–224.

Hayashi, T. and Miao, J. (2011). Intertemporal substitution and recursive smooth ambiguity preferences. *Theoretical Economics*, 6(3):423–472.

Heathcote, J. and Perri, F. (2018). Wealth and volatility. *The Review of Economic Studies*, 85(4):2173–2213.

Hens, T. and Schindler, N. (2020). Value and patience: The value premium in a dividend-growth model with hyperbolic discounting. *Journal of Economic Behavior & Organization*, 172:161–179.

Hernández-Lerma, O. and Lasserre, J. B. (2012a). *Discrete-time Markov control processes: basic optimality criteria*, volume 30. Springer Science & Business Media.

Hernández-Lerma, O. and Lasserre, J. B. (2012b). *Further topics on discrete-time Markov control processes*, volume 42. Springer Science & Business Media.

Herrendorf, B., Rogerson, R., and Valentinyi, A. (2021). Structural change in investment and consumption—a unified analysis. *The Review of Economic Studies*, 88(3):1311–1346.

Hills, T. S., Nakata, T., and Schmidt, S. (2019). Effective lower bound risk. *European Economic Review*, 120:103321.

Hopenhayn, H. A. (1992). Entry, exit, and firm dynamics in long run equilibrium. *Econometrica: Journal of the Econometric Society*, pages 1127–1150.

Hopenhayn, H. A. and Prescott, E. C. (1992). Stochastic monotonicity and stationary distributions for dynamic economies. *Econometrica*, pages 1387–1406.

Horn, R. A. and Johnson, C. R. (2012). *Matrix analysis*. Cambridge university press.

Howard, R. A. (1960). *Dynamic programming and markov processes*. John Wiley.

Hsu, W.-T. (2012). Central place theory and city size distribution. *The Economic Journal*, 122(563):903–932.

Hsu, W.-T., Holmes, T. J., and Morgan, F. (2014). Optimal city hierarchy: a dynamic programming approach to central place theory. *Journal of Economic Theory*, 154:245–273.

Hu, T.-W. and Shmaya, E. (2019). Unique monetary equilibrium with inflation in a stationary bewley–aiyagari model. *Journal of Economic Theory*, 180:368–382.

Hubmer, J., Krusell, P., and Smith, Jr, A. A. (2020). Sources of us wealth inequality: Past, present, and future. *NBER Macroeconomics Annual 2020, volume 35*.

Huggett, M. (1993). The risk-free rate in heterogeneous-agent incomplete-insurance economies. *Journal of economic Dynamics and Control*, 17(5-6):953–969.

Huijben, I. A., Kool, W., Paulus, M. B., and Van Sloun, R. J. (2022). A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Iskhakov, F., Rust, J., and Schjerning, B. (2020). Machine learning and structural econometrics: contrasts and synergies. *The Econometrics Journal*, 23(3):S81–S124.

Jacobson, D. H. (1973). Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic differential games. *IEEE Transactions for Automatic Control*, AC-18:1124–131.

Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Jaśkiewicz, A. and Nowak, A. S. (2014). Stationary markov perfect equilibria in risk sensitive stochastic overlapping generations models. *Journal of Economic Theory*, 151:411–447.

Jaśkiewicz, A. and Nowak, A. S. (2021). Markov decision processes with quasi-hyperbolic discounting. *Finance and Stochastics*, 25(2):189–229.

Jasso-Fuentes, H., Menaldi, J.-L., and Prieto-Rumeau, T. (2020). Discrete-time control with non-constant discount factor. *Mathematical Methods of Operations Research*, pages 1–23.

Jovanovic, B. (1979). Firm-specific capital and turnover. *Journal of political economy*, 87(6):1246–1260.

Jovanovic, B. (1982). Selection and the evolution of industry. *Econometrica*, pages 649–670.

Jovanovic, B. (1984). Matching, turnover, and unemployment. *Journal of political Economy*, 92(1):108–122.

Kamihigashi, T. (2014). Elementary results on solutions to the bellman equation of dynamic programming: existence, uniqueness, and convergence. *Economic Theory*, 56:251–273.

Kamihigashi, T. and Stachurski, J. (2014). Stochastic stability in monotone economies. *Theoretical Economics*, 9(2):383–407.

Kaplan, G., Moll, B., and Violante, G. L. (2018). Monetary policy according to hank. *American Economic Review*, 108(3):697–743.

Karp, L. (2005). Global warming and hyperbolic discounting. *Journal of public economics*, 89(2-3):261–282.

Keane, M. P., Todd, P. E., and Wolpin, K. I. (2011). The structural estimation of behavioral models: Discrete choice dynamic programming methods and applications. In *Handbook of labor economics*, volume 4, pages 331–461. Elsevier.

Keane, M. P. and Wolpin, K. I. (1997). The career decisions of young men. *Journal of political Economy*, 105(3):473–522.

Kelle, P. and Milne, A. (1999). The effect of (s, s) ordering policy on the supply chain. *International Journal of Production Economics*, 59(1-3):113–122.

Kikuchi, T., Nishimura, K., Stachurski, J., and Zhang, J. (2021). Coase meets bellman: Dynamic programming for production networks. *Journal of Economic Theory*, 196:105287.

Klibanoff, P., Marinacci, M., and Mukerji, S. (2009). Recursive smooth ambiguity preferences. *Journal of Economic Theory*, 144(3):930–976.

Kochenderfer, M. J., Wheeler, T. A., and Wray, K. H. (2022). *Algorithms for decision making*. The MIT Press.

Kohler, M., Krzyżak, A., and Todorovic, N. (2010). Pricing of high-dimensional american options by neural networks. *Mathematical Finance*, 20(3):383–410.

Koopmans, T. C. (1960). Stationary ordinal utility and impatience. *Econometrica: Journal of the Econometric Society*, pages 287–309.

Kreps, D. M. and Porteus, E. L. (1978). Temporal resolution of uncertainty and dynamic choice theory. *Econometrica: journal of the Econometric Society*, pages 185–200.

Kreyszig, E. (1978). *Introductory functional analysis with applications*, volume 1. wiley New York.

Kristensen, D., Mogensen, P. K., Moon, J. M., and Schjerning, B. (2021). Solving dynamic discrete choice models using smoothing and sieve methods. *Journal of Econometrics*, 223(2):328–360.

Krueger, D., Mitman, K., and Perri, F. (2016). Macroeconomics and household heterogeneity. In *Handbook of macroeconomics*, volume 2, pages 843–921. Elsevier.

Krusell, P. and Smith, Jr, A. A. (1998). Income and wealth heterogeneity in the macroeconomy. *Journal of political Economy*, 106(5):867–896.

Lee, J. and Shin, K. (2000). The role of a variable input in the relationship between investment and uncertainty. *American Economic Review*, 90(3):667–680.

Legrand, N. (2019). The empirical merit of structural explanations of commodity price volatility: Review and perspectives. *Journal of Economic Surveys*, 33(2):639–664.

Lehrer, E. and Light, B. (2018). The effect of interest rates on consumption in an income fluctuation problem. *Journal of Economic Dynamics and Control*, 94:63–71.

Lettau, M. and Ludvigson, S. C. (2014). Shocks and crashes. *NBER Macroeconomics Annual*, 28(1):293–354.

Li, H. and Stachurski, J. (2014). Solving the income fluctuation problem with unbounded rewards. *Journal of Economic Dynamics and Control*, 45:353–365.

Liao, J. and Berg, A. (2018). Sharpening jensen's inequality. *The American Statistician*.

Light, B. (2018). Precautionary saving in a markovian earnings environment. *Review of Economic Dynamics*, 29:138–147.

Light, B. (2020). Uniqueness of equilibrium in a bewley–aiyagari model. *Economic Theory*, 69(2):435–450.

Ljungqvist, L. (2002). How do lay-off costs affect employment? *The Economic Journal*, 112(482):829–853.

Ljungqvist, L. and Sargent, T. J. (2018). *Recursive macroeconomic theory*. MIT press, 4 edition.

Loewenstein, G. and Prelec, D. (1991). Negative time preference. *The American Economic Review*, 81:347–352.

Loewenstein, G. and Sicherman, N. (1991). Do workers prefer increasing wage profiles? *Journal of Labor Economics*, 9:67–84.

Longstaff, F. A. and Schwartz, E. S. (2001). Valuing american options by simulation: a simple least-squares approach. *The review of financial studies*, 14(1):113–147.

Lucas, R. E. (1976). Econometric policy evaluation: A critique. In *Carnegie-Rochester conference series on public policy*, volume 1, pages 19–46. North-Holland.

Lucas, R. E. (1978a). Asset prices in an exchange economy. *Econometrica: Journal of the Econometric Society*, pages 1429–1445.

Lucas, R. E. (1978b). Unemployment policy. *The American Economic Review*, 68(2):353–357.

Lucas, R. E. and Prescott, E. C. (1971). Investment under uncertainty. *Econometrica: Journal of the Econometric Society*, pages 659–681.

Lucas, R. E. and Sargent, T. J. (1981). *Rational expectations and econometric practice*, volume 2. U of Minnesota Press.

Luo, Y. and Sang, P. (2022). Penalized sieve estimation of structural models. Technical report, arXiv preprint arXiv:2204.13488.

Ma, Q. and Stachurski, J. (2021). Dynamic programming deconstructed: Transformations of the bellman equation and computational efficiency. *Operations Research*, 69(5):1591–1607.

Ma, Q., Stachurski, J., and Toda, A. A. (2020). The income fluctuation problem and the evolution of wealth. *Journal of Economic Theory*, 187:105003.

Ma, Q. and Toda, A. A. (2021). A theory of the saving rate of the rich. *Journal of Economic Theory*, 192:105193.

Majumdar, A., Singh, S., Mandlekar, A., and Pavone, M. (2017). Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: Science and Systems*, volume 16, page 117.

Marcet, A., Obiols-Homs, F., and Weil, P. (2007). Incomplete markets, labor supply and capital accumulation. *Journal of Monetary Economics*, 54(8):2621–2635.

Marinacci, M. and Montrucchio, L. (2010). Unique solutions for stochastic recursive utilities. *Journal of Economic Theory*, 145(5):1776–1804.

Marinacci, M. and Montrucchio, L. (2019). Unique tarski fixed points. *Mathematics of Operations Research*, 44(4):1174–1191.

Martins-da Rocha, V. F. and Vailakis, Y. (2010). Existence and uniqueness of a fixed point for local contractions. *Econometrica*, 78(3):1127–1141.

McCall, J. J. (1970). Economics of Information and Job Search. *The Quarterly Journal of Economics*, 84(1):113–126.

McFadden, D. (1974). The measurement of urban travel demand. *Journal of public economics*, 3(4):303–328.

Mei, J., Xiao, C., Szepesvari, C., and Schuurmans, D. (2020). On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pages 6820–6829. PMLR.

Melo, F. S. (2001). Convergence of q-learning: A simple proof. Technical report, Institute Of Systems and Robotics, Tech. Rep.

Meyer, C. D. (2000). *Matrix analysis and applied linear algebra*, volume 71. Siam.

Miao, J. (2006). Competitive equilibria of economies with a continuum of consumers and aggregate shocks. *Journal of Economic Theory*, 128(1):274–298.

Michelacci, C., Paciello, L., and Pozzi, A. (2022). The extensive margin of aggregate consumption demand. *The Review of Economic Studies*, 89(2):909–947.

Mirman, L. J. and Zilcha, I. (1975). On optimal growth under uncertainty. *Journal of Economic Theory*, 11(3):329–339.

Mitten, L. (1964). Composition principles for synthesis of optimal multistage processes. *Operations Research*, 12(4):610–619.

Mogensen, P. K. (2018). Solving dynamic discrete choice models: Integrated or expected value function? *arXiv preprint arXiv:1801.03978*.

Mordecki, E. (2002). Optimal stopping and perpetual options for lévy processes. *Finance and Stochastics*, 6(4):473–493.

Mortensen, D. T. (1986). Job search and labor market analysis. *Handbook of labor economics*, 2:849–919.

Newhouse, D. (2005). The persistence of income shocks: Evidence from rural indonesia. *Review of development Economics*, 9(3):415–433.

Nirei, M. (2006). Threshold behavior and aggregate fluctuation. *Journal of Economic Theory*, 127(1):309–322.

Noor, J. and Takeoka, N. (2022). Optimal discounting. *Econometrica*, 90(2):585–623.

Norets, A. (2010). Continuity and differentiability of expected value functions in dynamic discrete choice models. *Quantitative economics*, 1(2):305–322.

Norris, J. R. (1998). *Markov chains*. Cambridge university press.

Nota, C. and Thomas, P. S. (2019). Is the policy gradient a gradient? Technical report, arXiv preprint arXiv:1906.07073.

Ok, E. A. (2007). *Real analysis with economic applications*, volume 10. Princeton University Press.

Paciello, L. and Wiederholt, M. (2014). Exogenous information, endogenous information, and optimal monetary policy. *Review of Economic Studies*, 81(1):356–388.

Paroussos, L., Mandel, A., Fragkiadakis, K., Fragkos, P., Hinkel, J., and Vrontisi, Z. (2019). Climate clubs and the macro-economic benefits of international cooperation on climate policy. *Nature Climate Change*, 9(7):542–546.

Perla, J. and Tonetti, C. (2014). Equilibrium imitation and growth. *Journal of Political Economy*, 122(1):52–76.

Peskir, G. and Shiryaev, A. (2006). *Optimal Stopping and Free-boundary Problems*. Springer.

Pissarides, C. A. (1979). Job matchings with state employment agencies and random search. *The Economic Journal*, 89(356):818–833.

Pohl, W., Schmedders, K., and Wilms, O. (2018). Higher order effects in asset pricing models with long-run risks. *The Journal of Finance*, 73(3):1061–1111.

Pohl, W., Schmedders, K., and Wilms, O. (2019). Relative existence for recursive utility. Technical report, SSRN 3432469.

Privault, N. (2013). Understanding markov chains. *Examples and Applications, Publisher Springer-Verlag Singapore*, 357:358.

Puterman, M. L. (2005). *Markov decision processes: discrete stochastic dynamic programming*. Wiley Interscience.

Quah, D. (1990). Permanent and transitory movements in labor income: An explanation for excess smoothness in consumption. *Journal of Political Economy*, 98(3):449–475.

Rendahl, P. (2016). Fiscal policy in an unemployment crisis. *The Review of Economic Studies*, 83(3):1189–1224.

Riedel, F. (2009). Optimal stopping with multiple priors. *Econometrica*, 77(3):857–908.

Roberts, M. J. and Tybout, J. R. (1997). The decision to export in colombia: An empirical model of entry with sunk costs. *The american economic review*, pages 545–564.

Rogers, L. C. (2002). Monte carlo valuation of american options. *Mathematical Finance*, 12(3):271–286.

Rogerson, R., Shimer, R., and Wright, R. (2005). Search-theoretic models of the labor market: A survey. *Journal of economic literature*, 43(4):959–988.

Ross, S. A. (2009). *Neoclassical finance*. Princeton University Press.

Rubinstein, A. (2003). Economics and psychology? the case of hyperbolic discounting. *International Economic Review*, 44(4):1207–1216.

Rust, J. (1987). Optimal replacement of gmc bus engines: An empirical model of harold zurcher. *Econometrica*, pages 999–1033.

Rust, J. (1994). Structural estimation of markov decision processes. *Handbook of econometrics*, 4:3081–3143.

Ruszczyński, A. (2010). Risk-averse dynamic programming for markov decision processes. *Mathematical programming*, 125(2):235–261.

Saijo, H. (2017). The uncertainty multiplier and business cycles. *Journal of Economic Dynamics and Control*, 78:1–25.

Samuelson, P. A. (1939). Interactions between the multiplier analysis and the principle of acceleration. *The Review of Economics and Statistics*, 21(2):75–78.

Sargent, T. J. (1980). "tobin's q" and the rate of investment in general equilibrium. In *Carnegie-Rochester Conference Series on Public Policy*, volume 12, pages 107–154. Elsevier.

Sargent, T. J. (1987). *Dynamic macroeconomic theory*. Harvard University Press.

Sargent, T. J. and Stachurski, J. (2022). Economic networks: Theory and computation.

Scarf, H. (1960). The optimality of (s, s) policies in the dynamic inventory problem. *Mathematical Methods in the Social Sciences*, pages 196–202.

Schechtman, J. (1976). An income fluctuation problem. *Journal of Economic Theory*, 12(2):218–241.

Schorfheide, F., Song, D., and Yaron, A. (2018). Identifying long-run risks: A bayesian mixed-frequency approach. *Econometrica*, 86(2):617–654.

Shen, Y., Tobia, M. J., Sommer, T., and Obermayer, K. (2014). Risk-sensitive reinforcement learning. *Neural computation*, 26(7):1298–1328.

Shiryaev, A. N. (2007). *Optimal stopping rules*, volume 8. Springer Science & Business Media.

Stachurski, J. (2022). *Economic dynamics: theory and computation*. MIT Press, 2 edition.

Stachurski, J. and Toda, A. A. (2019). An impossibility theorem for wealth in heterogeneous-agent models with limited heterogeneity. *Journal of Economic Theory*, 182:1–24.

Stachurski, J., Wilms, O., and Zhang, J. (2022). Unique solutions to power-transformed affine systems. *arXiv preprint arXiv:2212.00275*.

Stachurski, J. and Zhang, J. (2021). Dynamic programming with state-dependent discounting. *Journal of Economic Theory*, 192:105190.

Stokey, N. L. and Lucas, R. E. (1989). *Recursive methods in dynamic economics*. Harvard University Press.

Tallarini Jr, T. D. (2000). Risk-sensitive real business cycles. *Journal of monetary Economics*, 45(3):507–532.

Toda, A. A. (2014). Incomplete market dynamics and cross-sectional distributions. *Journal of Economic Theory*, 154:310–348.

Toda, A. A. (2019). Wealth distribution with random discount factors. *Journal of Monetary Economics*, 104:101–113.

Toda, A. A. (2021). Perov's contraction principle and dynamic programming with stochastic discounting. *Operations Research Letters*, 49(5):815–819.

Tsitsiklis, J. N. (1994). Asynchronous stochastic approximation and q-learning. *Machine learning*, 16:185–202.

Tyazhelnikov, V. (2022). Production clustering and offshoring. *American Economic Journal: Microeconomics*.

Van, C. and Dana, R.-A. (2003). *Dynamic Programming in Economics*. Springer.

Wang, P. and Wen, Y. (2012). Hayashi meets kiyotaki and moore: A theory of capital adjustment costs. *Review of Economic Dynamics*, 15(2):207–225.

Watkins, C. J. C. H. (1989). Learning from delayed rewards. Technical report, PhD Thesis, King's College, Cambridge United Kingdom.

Weil, P. (1990). Nonexpected utility in macroeconomics. *The Quarterly Journal of Economics*, 105(1):29–42.

Whittle, P. (1981). Risk-sensitive linear/quadratic/gaussian control. *Advances in Applied Probability*, 13(4):764–777.

Williams, D. (1991). *Probability with martingales*. Cambridge university press.

Woodford, M. (2011). Simple analytics of the government expenditure multiplier. *American Economic Journal: Macroeconomics*, 3(1):1–35.

Yue, V. Z. (2010). Sovereign default and debt renegotiation. *Journal of international Economics*, 80(2):176–187.

Zaanen, A. C. (2012). *Introduction to operator theory in Riesz spaces*. Springer.

Zhang, Z. (2012). *Variational, topological, and partial order methods with their applications*, volume 29. Springer.

Zhu, S. (2020). Existence of stationary equilibrium in an incomplete-market model with endogenous labor supply. *International Economic Review*, 61(3):1115–1138.

Author Index

Açıkgöz, Ömer T. 211
Aiyagari, S Rao 211
Al-Najjar, Nabil I 265
Albuquerque, Rui 137, 264
Algan, Yann 211
Aliprantis, Charalambos D 46
Allais, Olivier 211
Antràs, Pol 265
Arellano, Cristina 162, 264, 265
Arrow, Kenneth J 212
Atkinson, Kendall 41, 64, 67

Backus, David K 138
Bagliano, Fabio-Cesare 212
Balbus, Łukasz 99, 138, 139, 264
Bansal, Ravi 138
Barillas, Francisco 128, 138
Bartle, Robert Gardner 41, 67, 267–269
Basu, Susanto 137
Bäuerle, Nicole 138, 211
Bellman, Richard vi, 40, 211
Benhabib, Jess 211
Berg, Arthur 124
Bertola, Giuseppe 212
Bertsekas, Dimitri 211
Bertsekas, Dimitri P vi, 213, 264
Bhandari, Jalaj 264
Bianchi, Javier 211
Bisin, Alberto 211
Bloise, Gaetano 138, 264
Bloom, Nicholas 212
Bloom, Nick 212
Blundell, Richard 212
Bocola, Luigi 265
Bommier, Antoine 138
Bond, Stephen 212
Bornstein, Gideon 265
Borovička, Jaroslav 138
Boyd, John H 138
Brémaud, Pierre 82
Brock, William A 211
Bundick, Brent 137
Burdett, Kenneth 41
Burkinshaw, Owen 46

Cagetti, Marco 138, 265
Calsamiglia, Caterina 212
Campbell, John Y 138
Cao, Dan 139, 211, 264
Carroll, Christopher D 211, 212
Carruth, Alan 212
Carvalho, Vasco M 162
Chatterjee, Satyajit 265
Cheney, Ward 41, 67
Chetty, Raj 41
Cho, Yeol Je 67
Christensen, Timothy M 138
Christiano, Lawrence J 137
Cochrane, John H 111, 138

Cruces, Juan J 265

Daley, DJ 99
Dana, Rose-Anne 211
Dasgupta, Partha 138
De Gortari, Alonso 265
De Groot, Oliver 138
De Nardi, Mariacristina 212
Deaton, Angus 211
den Haan, Wouter J 211
Denardo, Eric V vi, 211, 264
Diamond, Peter 138
Dickerson, Andy 212
Dixit, Robert K 212
Dovis, Alessandro 265
Drugeon, Jean-Pierre 139
Du, Yihong 56, 57
Duffie, Darrell 138
Dvoretzky, Aryeh 212

Eichenbaum, Martin 137, 264
Epstein, Larry G. 132, 138
Ericson, Richard 162
Erosa, Andrés 211
Eyigungor, Burcu 265

Fagereng, Andreas 264
Fajgelbaum, Pablo D 55, 162
Farmer, Leland E 84
Foss, Sergey 99
Fragkiadakis, Kostas 211
Fragkos, Panagiotis 211
Frederick, Shane 137, 264
French, Eric 212
Fu, Chao 212

Gao, Yue 138
Gennaioli, Nicola 265
Gentry, Matthew L 212
Ghosh, Atish R 265
Gillingham, Kenneth 197
González, Beatriz 211
Graber, Michael 212
Grassi, Basile 162
Güell, Maia 212
Guo, Dajun 67
Guvenen, Fatih 212

Häggström, Olle 99
Han, Weimin 41, 64, 67
Hansen, Lars Peter 128, 138, 197, 217, 265
Harris, Theodore 212
Harrison, J Michael 119
Hassett, Kevin A 212
Hayashi, Fumio 212
Hayashi, Takashi 138
Heathcote, Jonathan 211
Heaton, John C 138
Henley, Andrew 212
Hens, Thorsten 139
Hernandez-Leal, Pablo 138
Hernández-Lerma, Onésimo 211, 223
Herrendorf, Berthold 211
Hills, Timothy S 137, 236, 237, 264
Hinkel, Jochen 211
Holm, Martin Blomhoff 264
Holmes, Thomas J 265
Hopenhayn, Hugo A 99, 162
Horn, Roger A 99
Howard, Ronald A 211
Hsu, Wen-Tai 265
Hu, Tai-Wei 99
Hubbard, R Glenn 212
Hubbard, Timothy P 212
Hubmer, Joachim 137, 211, 264
Huggett, Mark 211

Iskhakov, Fedor 197, 212

Jacobson, D. H. 217
Jaśkiewicz, Anna 99, 138, 139

Jasso-Fuentes, Héctor 264
Johnson, Charles R 99
Jones, John B 212
Jovanovic, Boyan 41, 162

Kamihigashi, Takashi 99, 264
Kaplan, Greg 211
Karp, Larry 138
Keane, Michael P 197, 198, 212
Kelle, Peter 212
Kiefer, Jack 212
Kiku, Dana 138
Kikuchi, Tomoo 265
Kim, Jun I 265
Klibanoff, Peter 138
Kochenderfer, Mykel J 211
Kochov, Asen 138
Kohler, Michael 162
Koopmans, Tjalling C 137
Köszegi, Botond 138
Kreps, David M 119, 132, 134
Kreyszig, Erwin 13–16
Kristensen, Dennis 212
Krusell, Per 106, 137, 211, 264
Krzyżak, Adam 162

Laroque, Guy 211
Lasserre, Jean B 211, 223
Le Grand, François 138
Le Van, Cuong 264
Lee, Jaewoo 212
Legrand, Nicolas 212
Lehrer, Ehud 211
Lettau, Martin 212
Li, Huiyu 211
Li, Nan 138
Liao, JG 124
Light, Bar 211
Ljungqvist, Lars 41, 99, 138
Loewenstein, George 137, 264, 265
Longstaff, Francis A 162
Lucas, Robert E vi, 41, 99, 112, 197, 211, 223
Ludvigson, Sydney C 212
Lui, Kry Yik Chau 138
Luo, Victor Xi 137, 264
Luo, Yao 212

Ma, Qingyin 211, 212
Majumdar, Anirudha 138
Mandel, Antoine 211
Mandlekar, Ajay 138
Marcet, Albert 184
Marinacci, Massimo 138
Marinacci, Massino 138
Marschak, Jacob 212
Martin, Alberto 265
Martins-da Rocha, V Filipe 41
Maskin, Eric 138
McCall, J J 3, 40, 162
Mei, Jincheng 264
Melo, Francisco S 212
Menaldi, José-Luis 264
Mendoza, Enrique G 265
Meyer, Carl D 69, 99
Miao, Jianjun 138, 211
Michelacci, Claudio 211
Milne, Alistair 212
Mirman, Leonard J 211
Mitten, LG 264
Mogensen, Patrick K 212
Mogstad, Magne 212
Moll, Benjamin 211, 264
Montrucchio, Luigi 138
Moon, Jong Myun 212
Mordecki, Ernesto 162
Morgan, Frank 265
Mortensen, Dale T 41
Motto, Roberto 137
Mukerji, Sujoy 138

Munk-Nielsen, Anders 197

Nakata, Taisuke 137, 236, 237, 264
Natvik, Gisle 264
Nekipelov, Denis 212
Newhouse, David 212
Nirei, Makoto 212
Nishimura, Kazuo 265
Noor, Jawwad 264
Norets, Andriy 212
Norris, James Robert 99
Nota, Chris 264
Nowak, Andrzej S 99, 139

Obermayer, Klaus 138
Obiols-Homs, Francesc 184
O'donoghue, Ted 137, 264
Ok, Efe A 41
Ostry, Jonathan D 265

Paarsch, Harry J 212
Paciello, Luigi 211
Pakes, Ariel 162
Paroussos, Leonidas 211
Pavone, Marco 138
Perla, Jesse 162
Perri, Fabrizio 211
Peskir, Goran 162
Pindyck, Robert S 212
Pissarides, Christopher A 41
Pohl, Walt 138
Pohl, Walter 138
Porteus, Evan L 132, 134
Pozzi, Andrea 211
Prelec, Drazen 265
Prescott, Edward C 99, 211
Prieto-Rumeau, Tomás 264
Privault, Nicolas 99
Puterman, Martin L vi, 180, 211, 223

Quah, Danny 212
Qureshi, Mahvash S 265

Ramanarayanan, Ananth 265
Rebelo, Sergio 137, 264
Reffett, Kevin 99, 139, 264
Renault, Eric 138
Rendahl, Pontus 211
Richter, Alexander W 138
Riedel, Frank 162
Rieder, Ulrich 211
Roberts, Mark J 212
Rogers, Leonard CG 162
Rogerson, Richard 41, 211
Ross, Stephen A 138
Rossi, Stefano 265
Rostagno, Massimo 137
Routledge, Bryan R 138
Rubinstein, Ariel 138, 264
Russo, Daniel 264
Rust, John 162, 165, 197, 212
Ruszczyński, Andrzej 265

Saijo, Hikaru 264
Samuelson, Paul A 137, 138
Sang, Peijun 212
Sargent, Thomas J 99, 128, 138, 197, 211, 212, 217, 265
Scarf, H 212
Schaal, Edouard 55, 162
Schechtman, Jack 211
Scheinkman, José A. 138
Schindler, Nilüfer 139
Schjerning, Bertel 197, 212
Schmedders, Karl 138
Schmidt, Sebastian 137, 236, 237, 264
Schorfheide, Frank 137, 138, 264
Schuurmans, Dale 264
Schwartz, Eduardo S 162
Shen, Yun 138
Sherbert, Donald R 41, 67, 267–269

Shimer, Robert 41
Shin, Kwanho 212
Shiryaev, Albert 162
Shiryaev, Albert N 162
Shmaya, Eran 99, 265
Shneer, Vsevolod 99
Sicherman, Nachum 265
Singh, Sumeet 138
Smith, Anthony A, Jr 106, 137, 211, 264
Sommer, Tobias 138
Song, Dongho 137, 138, 264
Stachurski, John vii, 12, 67, 69, 99, 110, 137, 138, 211, 212, 264, 265
Stokey, Nancy L vi, 99, 211, 223
Szepesvari, Csaba 264

Takeoka, Norio 264
Tallarini Jr, Thomas D 128, 138
Taschereau-Dumouchel, Mathieu 55, 162
Thomas, Jonathan P 99
Thomas, Philip S 264
Throckmorton, Nathaniel A 138
Tobia, Michael J 138
Toda, Alexis Akira 84, 137, 211, 212
Todd, Petra E 197, 198, 212
Todorovic, Nebojsa 162
Tonetti, Christopher 162
Trebesch, Christoph 265
Tsitsiklis, John N 212
Tyazhelnikov, Vladimir 265
Tybout, James R 212

Vailakis, Yiannis 41, 138, 264
Valentinyi, Akos 211
Van, Cuong 211
Van Reenen, John 212
Violante, Giovanni L 211
Vrontisi, Zoi 211

Wang, Pengfei 212
Watkins, Christopher John Cornish Hellaby 212
Weil, Philippe 184
Wen, Yi 212
Werning, Iván 139, 264
Wheeler, Tim A 211
Whittle, Peter 217
Wiederholt, Mirko 211
Wigniolle, Bertrand 139
Williams, David 101
Williams, Noah 138, 265
Wilms, Ole 110, 138
Wolfowitz, Jacob 212
Wolpin, Kenneth I 197, 198, 212
Woodford, Michael 264
Worrall, Tim 99
Woźny, Łukasz 99, 139, 264
Wray, Kyle H 211
Wright, Randall 41

Xiao, Chenjun 264

Yaron, Amir 137, 138, 264
Yue, Vivian Z 264

Zhang, Junnan 110, 137, 264, 265
Zhang, Zhitao 57, 67
Zhu, Jiang 67
Zhu, Shenghao 184, 211
Zilcha, Itzhak 211
Zin, Stanley E 132, 138

Subject Index

P-Markov, 74
Q-order convergence, 57
$\varphi \preceq_{\mathrm{F}} \psi$, 87
σ-value function, 264

Absolute value, 12
Absorbing set, 78
Absorbing state, 78
Abstract dynamic program, 264
Action, 1
Action space, 126, 235
Additive separability, 92, 229
ADP, 264
Aggregator, 227
American option, 103, 117
Anti-isomorphic RDPs, 277
Arrow–Debreu discount operator, 188

Bellman equation, 11, 126, 267
Bellman operator, 33, 106, 134, 240, 267
Bellman's principle, 134, 242, 267
Bellman's principle of optimality, 278
Bermudan options, 103
Bijection, 286
Blackwell's conditions, 249

Call option, 117
Cauchy–Schwarz inequality, 13
Certainty equivalent, 225
CES aggregator, 227
Closed set, 15
Column major, 71
Complex numbers, 12
Complex vectors, 12
Composition, 286
Computational complexity, 60
Concave, 208
Conjugate dynamics, 54
Continuation reward, 103
Continuation value, 4, 11
Continuation value function, 95
Contraction, 21
Convergence of real sequences, 288
Convergence of vectors, 14
Convex, 208
Correspondence, 125
Cost of capital, 180
CRRA utility, 93
Cum-dividend contract, 188
Cumulative distribution, 29

Damped iteration, 23
Decreasing function, 49
Decreasing set, 43
Diagonalizable matrix, 55
Discount factor, 126
Discount factors, 178
Discount rate, 279
Distribution, 29

Dominant eigenvector, 66
Dominate, 50

Eigenvalue, 16
Eigenvector, 16
Equivalence, 15
Ergodicity, 80
Euclidean norm, 13
Eventually contracting, 193
Eventually contracting RDP, 257
Ex-dividend contract, 187
Exit reward, 103
Expectation, 29
Expected value operator, 163

Feasible state-action pairs, 235
Feasible correspondence, 126
Feasible correspondence, 235
Feasible policy, 131, 236
Feasible state-action pairs, 126
Finite MDP, 126
Finite set, 286
Flow utility, 93
Function
    real-valued, 28

Gelfand's formula, 18
Globally stable, 21
Greatest element, 46
Greedy policy, 35, 107, 133, 173, 198, 240, 264, 274
Gumbel distribution, 164

Homeomorphism, 54
Howard policy iteration, 137
HPI, 137
Hyperbolic discounting, 202

Inada conditions, 207
Increasing function, 49
Increasing set, 43
Indicator function, 14
Infimum, 287, 289
Initial condition, 74
Inner product, 12
Interest rates, 178
Invariant sets, 21
Irreducibility, 62
Isomorphic ADPs, 274
Isomorphic RDPs, 277

Jacobian, 56

Kernel, 69
Knaster–Tarski theorem, 205
Koopmans operator, 224

Lattice, 263, 291
Least element, 46
Leontief aggregator, 227
Lifetime reward, 91
Lifetime utility, 224
Linear aggregator, 227
Linear convergence, 57
Linear operator, 68, 69
Locally stable, 56
Lower bound, 287
Lucas SDF, 186

Markov chain, 74
Markov decision process, 126
Markov matrix, 63
Markov operator, 72
Markov policy, 34
Matrix norm, 17
Max-greedy policies, 274
Max-optimal policies, 274
Max-value functions, 274
Maximizer, 288
Maximum, 287
Median, 29
Minimizer, 288

Minimum, 287
Minkowski's inequality, 14
Modulus of contraction, 247
Monotone increasing matrix, 89
Monotonicity, 236

Negative discounting, 279
Newton's method, 58
Nonexpansive operator, 172
Nonnegative matrix, 62
Norm, 13

One-to-one, 286
One-to-one correspondence, 286
Onto, 286
Open set, 15
Optimal policy, 267
Optimal policy, 104, 133, 274
Optimal stopping, 103
Option, 117
Order anti-isomorphism, 273
Order interval, 45
Order isomorphism, 273
Order theory, 42
Order-preserving, 48
Order-reversing, 48

Partial order, 42
Partially ordered set, 43
Pointwise order, 44, 45
Policy, 34
Policy functions, 264
Policy operator, 132, 240, 264
Positive cone, 71
Positive linear operator, 71
Positive matrix, 62
Pricing kernel, 186

Q-Learning, 168
Quadratic convergence, 57
Quantile, 29

Rates of convergence, 57
RDP, 235
Real sequence, 288
Real vector, 12
Recursive decision process, 235
Recursive preferences, 212
Reinforcement Learning, 168
Reservation wage, 6, 35
Reward function, 126
Risk-neutral pricing, 183
Robust control, 261
Row major, 71

Self-map, 18
Set, 286
Spectral radius, 16
Stable RDP, 241
State space, 74, 126, 235
State variable, 1
State-dependent discounting, 178, 196
Stationary distribution, 63, 79, 82
Stochastic discount factor, 186
Stochastic dominance, 87, 116
Stochastic kernel, 126
Stopping value, 4
Strictly decreasing, 50
Strictly increasing, 50
Strike price, 117
Subadditivity, 226, 254
Sublattice, 236
Successive approximation, 23
Supremum, 287, 289
Supremum norm, 14

Time-varying discount factors, 178
Topological conjugacy, 55
Total order, 43
Transition matrix, 74
Triangle inequality, 46

Unit simplex, 29

Upper bound, 287
Uzawa aggregator, 227

Value aggregator, 236
Value function, 11, 105, 133, 242, 266, 274
Value function iteration, 36, 136
Value space, 236
Value-continuity, 246