

The background of the book cover is a dark, atmospheric night scene. A bright, jagged lightning bolt strikes from the top right, illuminating the sky and casting a glow on the city lights below. The city lights are a mix of warm yellow and orange, creating a bokeh effect against the dark night. The overall mood is dramatic and powerful.

WARREN B. POWELL

REINFORCEMENT LEARNING AND STOCHASTIC OPTIMIZATION

A UNIFIED FRAMEWORK
FOR SEQUENTIAL DECISIONS

WILEY

**Reinforcement Learning and Stochastic Optimization:
A Unified Framework for Sequential Decisions**

Reinforcement Learning and Stochastic Optimization

A Unified Framework for Sequential Decisions

*Warren B. Powell
Princeton University
Princeton, NJ*

WILEY

This edition first published 2022

© 2022 John Wiley & Sons, Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by law. Advice on how to obtain permission to reuse material from this title is available at <http://www.wiley.com/go/permissions>.

The right of Warren B. Powell to be identified as the author of this work has been asserted in accordance with law.

Registered Office

John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, USA

Editorial Office

111 River Street, Hoboken, NJ 07030, USA

For details of our global editorial offices, customer services, and more information about Wiley products visit us at www.wiley.com.

Wiley also publishes its books in a variety of electronic formats and by print-on-demand. Some content that appears in standard print versions of this book may not be available in other formats.

Limit of Liability/Disclaimer of Warranty

The contents of this work are intended to further general scientific research, understanding, and discussion only and are not intended and should not be relied upon as recommending or promoting scientific method, diagnosis, or treatment by physicians for any particular patient. In view of ongoing research, equipment modifications, changes in governmental regulations, and the constant flow of information relating to the use of medicines, equipment, and devices, the reader is urged to review and evaluate the information provided in the package insert or instructions for each medicine, equipment, or device for, among other things, any changes in the instructions or indication of usage and for added warnings and precautions. While the publisher and authors have used their best efforts in preparing this work, they make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives, written sales materials or promotional statements for this work. The fact that an organization, website, or product is referred to in this work as a citation and/or potential source of further information does not mean that the publisher and authors endorse the information or services the organization, website, or product may provide or recommendations it may make. This work is sold with the understanding that the publisher is not engaged in rendering professional services. The advice and strategies contained herein may not be suitable for your situation. You should consult with a specialist where appropriate. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

A catalogue record for this book is available from the Library of Congress

Hardback ISBN: 9781119815037; ePub ISBN: 9781119815051; ePDF ISBN: 9781119815044;
Obook ISBN: 9781119815068.

Cover image: © DeniseSmit/Getty Images

Cover design by Wiley

Set in 9.5/12.5pt STIXTwoText by Integra Software Services Pvt. Ltd, Pondicherry, India

10 9 8 7 6 5 4 3 2 1

Contents

Preface *xxv*

Acknowledgments *xxxi*

Part I – Introduction *1*

1	Sequential Decision Problems	3
1.1	The Audience	7
1.2	The Communities of Sequential Decision Problems	8
1.3	Our Universal Modeling Framework	10
1.4	Designing Policies for Sequential Decision Problems	15
1.4.1	Policy Search	15
1.4.2	Policies Based on Lookahead Approximations	17
1.4.3	Mixing and Matching	18
1.4.4	Optimality of the Four Classes	19
1.4.5	Pulling it All Together	19
1.5	Learning	20
1.6	Themes	21
1.6.1	Blending Learning and Optimization	21
1.6.2	Bridging Machine Learning to Sequential Decisions	21
1.6.3	From Deterministic to Stochastic Optimization	23
1.6.4	From Single to Multiple Agents	26
1.7	Our Modeling Approach	27
1.8	How to Read this Book	27
1.8.1	Organization of Topics	28
1.8.2	How to Read Each Chapter	31
1.8.3	Organization of Exercises	32

1.9	Bibliographic Notes	33
	Exercises	34
	Bibliography	38
2	Canonical Problems and Applications	39
2.1	Canonical Problems	39
2.1.1	Stochastic Search – Derivative-based and Derivative-free	40
2.1.1.1	Derivative-based Stochastic Search	42
2.1.1.2	Derivative-free Stochastic Search	43
2.1.2	Decision Trees	44
2.1.3	Markov Decision Processes	45
2.1.4	Optimal Control	47
2.1.5	Approximate Dynamic Programming	50
2.1.6	Reinforcement Learning	50
2.1.7	Optimal Stopping	54
2.1.8	Stochastic Programming	56
2.1.9	The Multiarmed Bandit Problem	57
2.1.10	Simulation Optimization	60
2.1.11	Active Learning	61
2.1.12	Chance-constrained Programming	61
2.1.13	Model Predictive Control	62
2.1.14	Robust Optimization	63
2.2	A Universal Modeling Framework for Sequential Decision Problems	64
2.2.1	Our Universal Model for Sequential Decision Problems	65
2.2.2	A Compact Modeling Presentation	68
2.2.3	MDP/RL vs. Optimal Control Modeling Frameworks	68
2.3	Applications	69
2.3.1	The Newsvendor Problems	70
2.3.1.1	Basic Newsvendor – Final Reward	70
2.3.1.2	Basic Newsvendor – Cumulative Reward	71
2.3.1.3	Contextual Newsvendor	71
2.3.1.4	Multidimensional Newsvendor Problems	72
2.3.2	Inventory/Storage Problems	73
2.3.2.1	Inventory Without Lags	73
2.3.2.2	Inventory Planning with Forecasts	75
2.3.2.3	Lagged Decisions	75
2.3.3	Shortest Path Problems	76
2.3.3.1	A Deterministic Shortest Path Problem	76
2.3.3.2	A Stochastic Shortest Path Problem	77
2.3.3.3	A Dynamic Shortest Path Problem	78

2.3.3.4	A Robust Shortest Path Problem	78
2.3.4	Some Fleet Management Problems	78
2.3.4.1	The Nomadic Trucker	79
2.3.4.2	From One Driver to a Fleet	80
2.3.5	Pricing	80
2.3.6	Medical Decision Making	81
2.3.7	Scientific Exploration	82
2.3.8	Machine Learning vs. Sequential Decision Problems	83
2.4	Bibliographic Notes	85
	Exercises	90
	Bibliography	93
3	Online Learning	101
3.1	Machine Learning for Sequential Decisions	102
3.1.1	Observations and Data in Stochastic Optimization	102
3.1.2	Indexing Input x^n and Response y^{n+1}	103
3.1.3	Functions We are Learning	103
3.1.4	Sequential Learning: From Very Little Data to ... More Data	105
3.1.5	Approximation Strategies	106
3.1.6	From Data Analytics to Decision Analytics	108
3.1.7	Batch vs. Online Learning	109
3.2	Adaptive Learning Using Exponential Smoothing	110
3.3	Lookup Tables with Frequentist Updating	111
3.4	Lookup Tables with Bayesian Updating	112
3.4.1	The Updating Equations for Independent Beliefs	113
3.4.2	Updating for Correlated Beliefs	113
3.4.3	Gaussian Process Regression	117
3.5	Computing Bias and Variance*	118
3.6	Lookup Tables and Aggregation*	121
3.6.1	Hierarchical Aggregation	121
3.6.2	Estimates of Different Levels of Aggregation	125
3.6.3	Combining Multiple Levels of Aggregation	129
3.7	Linear Parametric Models	131
3.7.1	Linear Regression Review	132
3.7.2	Sparse Additive Models and Lasso	134
3.8	Recursive Least Squares for Linear Models	136
3.8.1	Recursive Least Squares for Stationary Data	136
3.8.2	Recursive Least Squares for Nonstationary Data*	138
3.8.3	Recursive Estimation Using Multiple Observations*	139
3.9	Nonlinear Parametric Models	140
3.9.1	Maximum Likelihood Estimation	141

3.9.2	Sampled Belief Models	141
3.9.3	Neural Networks – Parametric*	143
3.9.4	Limitations of Neural Networks	148
3.10	Nonparametric Models*	149
3.10.1	K-Nearest Neighbor	150
3.10.2	Kernel Regression	151
3.10.3	Local Polynomial Regression	153
3.10.4	Deep Neural Networks	154
3.10.5	Support Vector Machines	155
3.10.6	Indexed Functions, Tree Structures, and Clustering	156
3.10.7	Comments on Nonparametric Models	157
3.11	Nonstationary Learning*	159
3.11.1	Nonstationary Learning I – Martingale Truth	159
3.11.2	Nonstationary Learning II – Transient Truth	160
3.11.3	Learning Processes	161
3.12	The Curse of Dimensionality	162
3.13	Designing Approximation Architectures in Adaptive Learning	165
3.14	Why Does It Work?***	166
3.14.1	Derivation of the Recursive Estimation Equations	166
3.14.2	The Sherman-Morrison Updating Formula	168
3.14.3	Correlations in Hierarchical Estimation	169
3.14.4	Proof of Proposition 3.14.1	172
3.15	Bibliographic Notes	174
	Exercises	176
	Bibliography	180
4	Introduction to Stochastic Search	183
4.1	Illustrations of the Basic Stochastic Optimization Problem	185
4.2	Deterministic Methods	188
4.2.1	A “Stochastic” Shortest Path Problem	189
4.2.2	A Newsvendor Problem with Known Distribution	189
4.2.3	Chance-Constrained Optimization	190
4.2.4	Optimal Control	191
4.2.5	Discrete Markov Decision Processes	192
4.2.6	Remarks	192
4.3	Sampled Models	193
4.3.1	Formulating a Sampled Model	194
4.3.1.1	A Sampled Stochastic Linear Program	194
4.3.1.2	Sampled Chance-Constrained Models	195

4.3.1.3	Sampled Parametric Models	196
4.3.2	Convergence	197
4.3.3	Creating a Sampled Model	199
4.3.4	Decomposition Strategies*	200
4.4	Adaptive Learning Algorithms	202
4.4.1	Modeling Adaptive Learning Problems	202
4.4.2	Online vs. Offline Applications	204
4.4.2.1	Machine Learning	204
4.4.2.2	Optimization	205
4.4.3	Objective Functions for Learning	205
4.4.4	Designing Policies	209
4.5	Closing Remarks	210
4.6	Bibliographic Notes	210
	Exercises	212
	Bibliography	218

Part II – Stochastic Search 221

5	Derivative-Based Stochastic Search	223
5.1	Some Sample Applications	225
5.2	Modeling Uncertainty	228
5.2.1	Training Uncertainty W^1, \dots, W^N	228
5.2.2	Model Uncertainty S^0	229
5.2.3	Testing Uncertainty	230
5.2.4	Policy Evaluation	231
5.2.5	Closing Notes	231
5.3	Stochastic Gradient Methods	231
5.3.1	A Stochastic Gradient Algorithm	232
5.3.2	Introduction to Stepsizes	233
5.3.3	Evaluating a Stochastic Gradient Algorithm	235
5.3.4	A Note on Notation	236
5.4	Styles of Gradients	237
5.4.1	Gradient Smoothing	237
5.4.2	Second-Order Methods	237
5.4.3	Finite Differences	238
5.4.4	SPSA	240
5.4.5	Constrained Problems	242
5.5	Parameter Optimization for Neural Networks*	242
5.5.1	Computing the Gradient	244
5.5.2	The Stochastic Gradient Algorithm	246

- 5.6 Stochastic Gradient Algorithm as a Sequential Decision Problem 247
- 5.7 Empirical Issues 248
- 5.8 Transient Problems* 249
- 5.9 Theoretical Performance* 250
- 5.10 Why Does it Work? 250
- 5.10.1 Some Probabilistic Preliminaries 251
- 5.10.2 An Older Proof* 252
- 5.10.3 A More Modern Proof** 256
- 5.11 Bibliographic Notes 263
 - Exercises 264
 - Bibliography 270

- 6 Stepsize Policies 273**
 - 6.1 Deterministic Stepsize Policies 276
 - 6.1.1 Properties for Convergence 276
 - 6.1.2 A Collection of Deterministic Policies 278
 - 6.1.2.1 Constant Stepsizes 278
 - 6.1.2.2 Generalized Harmonic Stepsizes 279
 - 6.1.2.3 Polynomial Learning Rates 280
 - 6.1.2.4 McClain's Formula 280
 - 6.1.2.5 Search-then-Converge Learning Policy 281
 - 6.2 Adaptive Stepsize Policies 282
 - 6.2.1 The Case for Adaptive Stepsizes 283
 - 6.2.2 Convergence Conditions 283
 - 6.2.3 A Collection of Stochastic Policies 284
 - 6.2.3.1 Kesten's Rule 285
 - 6.2.3.2 Trigg's Formula 286
 - 6.2.3.3 Stochastic Gradient Adaptive Stepsize Rule 286
 - 6.2.3.4 ADAM 287
 - 6.2.3.5 AdaGrad 287
 - 6.2.3.6 RMSProp 288
 - 6.2.4 Experimental Notes 289
 - 6.3 Optimal Stepsize Policies* 289
 - 6.3.1 Optimal Stepsizes for Stationary Data 291
 - 6.3.2 Optimal Stepsizes for Nonstationary Data – I 293
 - 6.3.3 Optimal Stepsizes for Nonstationary Data – II 294
 - 6.4 Optimal Stepsizes for Approximate Value Iteration* 297
 - 6.5 Convergence 300
 - 6.6 Guidelines for Choosing Stepsize Policies 301
 - 6.7 Why Does it Work* 303

6.7.1	Proof of BAKF Stepsize	303
6.8	Bibliographic Notes	306
	Exercises	307
	Bibliography	314
7	Derivative-Free Stochastic Search	317
7.1	Overview of Derivative-free Stochastic Search	319
7.1.1	Applications and Time Scales	319
7.1.2	The Communities of Derivative-free Stochastic Search	321
7.1.3	The Multiarmed Bandit Story	321
7.1.4	From Passive Learning to Active Learning to Bandit Problems	323
7.2	Modeling Derivative-free Stochastic Search	325
7.2.1	The Universal Model	325
7.2.2	Illustration: Optimizing a Manufacturing Process	328
7.2.3	Major Problem Classes	329
7.3	Designing Policies	330
7.4	Policy Function Approximations	333
7.5	Cost Function Approximations	335
7.6	VFA-based Policies	338
7.6.1	An Optimal Policy	338
7.6.2	Beta-Bernoulli Belief Model	340
7.6.3	Backward Approximate Dynamic Programming	342
7.6.4	Gittins Indices for Learning in Steady State	343
7.7	Direct Lookahead Policies	348
7.7.1	When do we Need Lookahead Policies?	349
7.7.2	Single Period Lookahead Policies	350
7.7.3	Restricted Multiperiod Lookahead	353
7.7.4	Multiperiod Deterministic Lookahead	355
7.7.5	Multiperiod Stochastic Lookahead Policies	357
7.7.6	Hybrid Direct Lookahead	360
7.8	The Knowledge Gradient (Continued)*	362
7.8.1	The Belief Model	363
7.8.2	The Knowledge Gradient for Maximizing Final Reward	364
7.8.3	The Knowledge Gradient for Maximizing Cumulative Reward	369
7.8.4	The Knowledge Gradient for Sampled Belief Model*	370
7.8.5	Knowledge Gradient for Correlated Beliefs	375
7.9	Learning in Batches	380
7.10	Simulation Optimization*	382
7.10.1	An Indifference Zone Algorithm	383
7.10.2	Optimal Computing Budget Allocation	383
7.11	Evaluating Policies	385

7.11.1	Alternative Performance Metrics*	386
7.11.2	Perspectives of Optimality*	392
7.12	Designing Policies	394
7.12.1	Characteristics of a Policy	395
7.12.2	The Effect of Scaling	396
7.12.3	Tuning	398
7.13	Extensions*	398
7.13.1	Learning in Nonstationary Settings	399
7.13.2	Strategies for Designing Time-dependent Policies	400
7.13.3	A Transient Learning Model	401
7.13.4	The Knowledge Gradient for Transient Problems	402
7.13.5	Learning with Large or Continuous Choice Sets	403
7.13.6	Learning with Exogenous State Information – the Contextual Bandit Problem	405
7.13.7	State-dependent vs. State-independent Problems	408
7.14	Bibliographic Notes	409
	Exercises	412
	Bibliography	424

Part III – State-dependent Problems 429

8	State-dependent Problems	431
8.1	Graph Problems	433
8.1.1	A Stochastic Shortest Path Problem	433
8.1.2	The Nomadic Trucker	434
8.1.3	The Transformer Replacement Problem	435
8.1.4	Asset Valuation	437
8.2	Inventory Problems	439
8.2.1	A Basic Inventory Problem	439
8.2.2	The Inventory Problem – II	440
8.2.3	The Lagged Asset Acquisition Problem	443
8.2.4	The Batch Replenishment Problem	444
8.3	Complex Resource Allocation Problems	446
8.3.1	The Dynamic Assignment Problem	447
8.3.2	The Blood Management Problem	450
8.4	State-dependent Learning Problems	456
8.4.1	Medical Decision Making	457
8.4.2	Laboratory Experimentation	458
8.4.3	Bidding for Ad-clicks	459
8.4.4	An Information-collecting Shortest Path Problem	459

8.5	A Sequence of Problem Classes	460
8.6	Bibliographic Notes	461
	Exercises	462
	Bibliography	466
9	Modeling Sequential Decision Problems	467
9.1	A Simple Modeling Illustration	471
9.2	Notational Style	476
9.3	Modeling Time	478
9.4	The States of Our System	481
9.4.1	Defining the State Variable	481
9.4.2	The Three States of Our System	485
9.4.3	Initial State S_0 vs. Subsequent States S_t , $t > 0$	488
9.4.4	Lagged State Variables*	490
9.4.5	The Post-decision State Variable*	490
9.4.6	A Shortest Path Illustration	493
9.4.7	Belief States*	495
9.4.8	Latent Variables*	496
9.4.9	Rolling Forecasts*	497
9.4.10	Flat vs. Factored State Representations*	498
9.4.11	A Programmer's Perspective of State Variables	499
9.5	Modeling Decisions	500
9.5.1	Types of Decisions	502
9.5.2	Initial Decision x_0 vs. Subsequent Decisions x_t , $t > 0$	502
9.5.3	Strategic, Tactical, and Execution Decisions	503
9.5.4	Constraints	504
9.5.5	Introducing Policies	505
9.6	The Exogenous Information Process	506
9.6.1	Basic Notation for Information Processes	506
9.6.2	Outcomes and Scenarios	509
9.6.3	Lagged Information Processes*	510
9.6.4	Models of Information Processes*	511
9.6.5	Supervisory Processes*	514
9.7	The Transition Function	515
9.7.1	A General Model	515
9.7.2	Model-free Dynamic Programming	516
9.7.3	Exogenous Transitions	518
9.8	The Objective Function	518
9.8.1	The Performance Metric	518
9.8.2	Optimizing the Policy	519
9.8.3	Dependence of Optimal Policy on S_0	520

- 9.8.4 State-dependent Variations 520
- 9.8.5 Uncertainty Operators 523
- 9.9 Illustration: An Energy Storage Model 523
 - 9.9.1 With a Time-series Price Model 525
 - 9.9.2 With Passive Learning 525
 - 9.9.3 With Active Learning 526
 - 9.9.4 With Rolling Forecasts 526
- 9.10 Base Models and Lookahead Models 528
- 9.11 A Classification of Problems* 529
- 9.12 Policy Evaluation* 532
- 9.13 Advanced Probabilistic Modeling Concepts** 534
 - 9.13.1 A Measure-theoretic View of Information** 535
 - 9.13.2 Policies and Measurability 538
- 9.14 Looking Forward 540
- 9.15 Bibliographic Notes 542
 - Exercises 544
 - Bibliography 557

- 10 Uncertainty Modeling 559**
 - 10.1 Sources of Uncertainty 560
 - 10.1.1 Observational Errors 562
 - 10.1.2 Exogenous Uncertainty 564
 - 10.1.3 Prognostic Uncertainty 564
 - 10.1.4 Inferential (or Diagnostic) Uncertainty 567
 - 10.1.5 Experimental Variability 568
 - 10.1.6 Model Uncertainty 569
 - 10.1.7 Transitional Uncertainty 571
 - 10.1.8 Control/implementation Uncertainty 571
 - 10.1.9 Communication Errors and Biases 572
 - 10.1.10 Algorithmic Instability 573
 - 10.1.11 Goal Uncertainty 574
 - 10.1.12 Political/regulatory Uncertainty 574
 - 10.1.13 Discussion 574
 - 10.2 A Modeling Case Study: The COVID Pandemic 575
 - 10.3 Stochastic Modeling 575
 - 10.3.1 Sampling Exogenous Information 575
 - 10.3.2 Types of Distributions 577
 - 10.3.3 Modeling Sample Paths 578
 - 10.3.4 State-action-dependent Processes 579
 - 10.3.5 Modeling Correlations 581
 - 10.4 Monte Carlo Simulation 581

10.4.1	Generating Uniform $[0, 1]$ Random Variables	582
10.4.2	Uniform and Normal Random Variable	583
10.4.3	Generating Random Variables from Inverse Cumulative Distributions	585
10.4.4	Inverse Cumulative From Quantile Distributions	586
10.4.5	Distributions with Uncertain Parameters	587
10.5	Case Study: Modeling Electricity Prices	589
10.5.1	Mean Reversion	590
10.5.2	Jump-diffusion Models	590
10.5.3	Quantile Distributions	591
10.5.4	Regime Shifting	592
10.5.5	Crossing Times	593
10.6	Sampling vs. Sampled Models	595
10.6.1	Iterative Sampling: A Stochastic Gradient Algorithm	595
10.6.2	Static Sampling: Solving a Sampled Model	595
10.6.3	Sampled Representation with Bayesian Updating	596
10.7	Closing Notes	597
10.8	Bibliographic Notes	597
	Exercises	598
	Bibliography	601
11	Designing Policies	603
11.1	From Optimization to Machine Learning to Sequential Decision Problems	605
11.2	The Classes of Policies	606
11.3	Policy Function Approximations	610
11.4	Cost Function Approximations	613
11.5	Value Function Approximations	614
11.6	Direct Lookahead Approximations	616
11.6.1	The Basic Idea	616
11.6.2	Modeling the Lookahead Problem	619
11.6.3	The Policy-Within-a-Policy	620
11.7	Hybrid Strategies	620
11.7.1	Cost Function Approximation with Policy Function Approximations	621
11.7.2	Lookahead Policies with Value Function Approximations	622
11.7.3	Lookahead Policies with Cost Function Approximations	623
11.7.4	Tree Search with Rollout Heuristic and a Lookup Table Policy	623
11.7.5	Value Function Approximation with Policy Function Approximation	624
11.7.6	Fitting Value Functions Using ADP and Policy Search	624

11.8	Randomized Policies	626
11.9	Illustration: An Energy Storage Model Revisited	627
11.9.1	Policy Function Approximation	628
11.9.2	Cost Function Approximation	628
11.9.3	Value Function Approximation	628
11.9.4	Deterministic Lookahead	629
11.9.5	Hybrid Lookahead-Cost Function Approximation	629
11.9.6	Experimental Testing	629
11.10	Choosing the Policy Class	631
11.10.1	The Policy Classes	631
11.10.2	Policy Complexity-Computational Tradeoffs	636
11.10.3	Screening Questions	638
11.11	Policy Evaluation	641
11.12	Parameter Tuning	642
11.12.1	The Soft Issues	644
11.12.2	Searching Across Policy Classes	645
11.13	Bibliographic Notes	646
	Exercises	646
	Bibliography	651

Part IV – Policy Search 653

12	Policy Function Approximations and Policy Search	655
12.1	Policy Search as a Sequential Decision Problem	657
12.2	Classes of Policy Function Approximations	658
12.2.1	Lookup Table Policies	659
12.2.2	Boltzmann Policies for Discrete Actions	659
12.2.3	Linear Decision Rules	660
12.2.4	Monotone Policies	661
12.2.5	Nonlinear Policies	662
12.2.6	Nonparametric/Locally Linear Policies	663
12.2.7	Contextual Policies	665
12.3	Problem Characteristics	665
12.4	Flavors of Policy Search	666
12.5	Policy Search with Numerical Derivatives	669
12.6	Derivative-Free Methods for Policy Search	670
12.6.1	Belief Models	671
12.6.2	Learning Through Perturbed PFAs	672
12.6.3	Learning CFAs	675
12.6.4	DLA Using the Knowledge Gradient	677

12.6.5	Comments	677
12.7	Exact Derivatives for Continuous Sequential Problems*	677
12.8	Exact Derivatives for Discrete Dynamic Programs**	680
12.8.1	A Stochastic Policy	681
12.8.2	The Objective Function	683
12.8.3	The Policy Gradient Theorem	683
12.8.4	Computing the Policy Gradient	684
12.9	Supervised Learning	686
12.10	Why Does it Work?	687
12.10.1	Derivation of the Policy Gradient Theorem	687
12.11	Bibliographic Notes	690
	Exercises	691
	Bibliography	698
13	Cost Function Approximations	701
13.1	General Formulation for Parametric CFA	703
13.2	Objective-Modified CFAs	704
13.2.1	Linear Cost Function Correction	705
13.2.2	CFAs for Dynamic Assignment Problems	705
13.2.3	Dynamic Shortest Paths	707
13.2.4	Dynamic Trading Policy	711
13.2.5	Discussion	713
13.3	Constraint-Modified CFAs	714
13.3.1	General Formulation of Constraint-Modified CFAs	715
13.3.2	A Blood Management Problem	715
13.3.3	An Energy Storage Example with Rolling Forecasts	717
13.4	Bibliographic Notes	725
	Exercises	726
	Bibliography	729
	Part V – Lookahead Policies	731
14	Exact Dynamic Programming	737
14.1	Discrete Dynamic Programming	738
14.2	The Optimality Equations	740
14.2.1	Bellman's Equations	741
14.2.2	Computing the Transition Matrix	745
14.2.3	Random Contributions	746
14.2.4	Bellman's Equation Using Operator Notation*	746

14.3	Finite Horizon Problems	747
14.4	Continuous Problems with Exact Solutions	750
14.4.1	The Gambling Problem	751
14.4.2	The Continuous Budgeting Problem	752
14.5	Infinite Horizon Problems*	755
14.6	Value Iteration for Infinite Horizon Problems*	757
14.6.1	A Gauss-Seidel Variation	758
14.6.2	Relative Value Iteration	758
14.6.3	Bounds and Rates of Convergence	760
14.7	Policy Iteration for Infinite Horizon Problems*	762
14.8	Hybrid Value-Policy Iteration*	764
14.9	Average Reward Dynamic Programming*	765
14.10	The Linear Programming Method for Dynamic Programs**	766
14.11	Linear Quadratic Regulation	767
14.12	Why Does it Work?***	770
14.12.1	The Optimality Equations	770
14.12.2	Convergence of Value Iteration	774
14.12.3	Monotonicity of Value Iteration	778
14.12.4	Bounding the Error from Value Iteration	780
14.12.5	Randomized Policies	781
14.13	Bibliographic Notes	783
	Exercises	783
	Bibliography	793
15	Backward Approximate Dynamic Programming	795
15.1	Backward Approximate Dynamic Programming for Finite Horizon Problems	797
15.1.1	Some Preliminaries	797
15.1.2	Backward ADP Using Lookup Tables	799
15.1.3	Backward ADP Algorithm with Continuous Approximations	801
15.2	Fitted Value Iteration for Infinite Horizon Problems	804
15.3	Value Function Approximation Strategies	805
15.3.1	Linear Models	806
15.3.2	Monotone Functions	807
15.3.3	Other Approximation Models	810
15.4	Computational Observations	810
15.4.1	Experimental Benchmarking of Backward ADP	810
15.4.2	Computational Notes	815
15.5	Bibliographic Notes	816
	Exercises	816

Bibliography 821

16	Forward ADP I: The Value of a Policy	823
16.1	Sampling the Value of a Policy	824
16.1.1	Direct Policy Evaluation for Finite Horizon Problems	824
16.1.2	Policy Evaluation for Infinite Horizon Problems	826
16.1.3	Temporal Difference Updates	828
16.1.4	TD(λ)	829
16.1.5	TD(0) and Approximate Value Iteration	830
16.1.6	TD Learning for Infinite Horizon Problems	832
16.2	Stochastic Approximation Methods	835
16.3	Bellman's Equation Using a Linear Model*	837
16.3.1	A Matrix-based Derivation**	837
16.3.2	A Simulation-based Implementation	840
16.3.3	Least Squares Temporal Difference Learning (LSTD)	840
16.3.4	Least Squares Policy Evaluation	841
16.4	Analysis of TD(0), LSTD, and LSPE Using a Single State*	842
16.4.1	Recursive Least Squares and TD(0)	842
16.4.2	Least Squares Policy Evaluation	844
16.4.3	Least Squares Temporal Difference Learning	844
16.4.4	Discussion	844
16.5	Gradient-based Methods for Approximate Value Iteration*	845
16.5.1	Approximate Value Iteration with Linear Models**	845
16.5.2	A Geometric View of Linear Models*	850
16.6	Value Function Approximations Based on Bayesian Learning*	852
16.6.1	Minimizing Bias for Infinite Horizon Problems	852
16.6.2	Lookup Tables with Correlated Beliefs	853
16.6.3	Parametric Models	854
16.6.4	Creating the Prior	855
16.7	Learning Algorithms and Atepsizes	855
16.7.1	Least Squares Temporal Differences	856
16.7.2	Least Squares Policy Evaluation	857
16.7.3	Recursive Least Squares	857
16.7.4	Bounding $1/n$ Convergence for Approximate value Iteration	859
16.7.5	Discussion	860
16.8	Bibliographic Notes	860
	Exercises	862
	Bibliography	864

17	Forward ADP II: Policy Optimization	867
17.1	Overview of Algorithmic Strategies	869
17.2	Approximate Value Iteration and <i>Q</i> -Learning Using Lookup Tables	871
17.2.1	Value Iteration Using a Pre-Decision State Variable	872
17.2.2	<i>Q</i> -Learning	873
17.2.3	Value Iteration Using a Post-Decision State Variable	875
17.2.4	Value Iteration Using a Backward Pass	877
17.3	Styles of Learning	881
17.3.1	Offline Learning	882
17.3.2	From Offline to Online	883
17.3.3	Evaluating Offline and Online Learning Policies	885
17.3.4	Lookahead Policies	885
17.4	Approximate Value Iteration Using Linear Models	886
17.5	On-policy vs. off-policy learning and the exploration–exploitation problem	888
17.5.1	Terminology	889
17.5.2	Learning with Lookup Tables	890
17.5.3	Learning with Generalized Belief Models	891
17.6	Applications	894
17.6.1	Pricing an American Option	894
17.6.2	Playing “Lose Tic-Tac-Toe”	898
17.6.3	Approximate Dynamic Programming for Deterministic Problems	900
17.7	Approximate Policy Iteration	900
17.7.1	Finite Horizon Problems Using Lookup Tables	901
17.7.2	Finite Horizon Problems Using Linear Models	903
17.7.3	LSTD for Infinite Horizon Problems Using Linear Models	903
17.8	The Actor–Critic Paradigm	907
17.9	Statistical Bias in the Max Operator*	909
17.10	The Linear Programming Method Using Linear Models*	912
17.11	Finite Horizon Approximations for Steady-State Applications	915
17.12	Bibliographic Notes	917
	Exercises	918
	Bibliography	924
18	Forward ADP III: Convex Resource Allocation Problems	927
18.1	Resource Allocation Problems	930

18.1.1	The Newsvendor Problem	930
18.1.2	Two-Stage Resource Allocation Problems	931
18.1.3	A General Multiperiod Resource Allocation Model*	933
18.2	Values Versus Marginal Values	937
18.3	Piecewise Linear Approximations for Scalar Functions	938
18.3.1	The Leveling Algorithm	939
18.3.2	The CAVE Algorithm	941
18.4	Regression Methods	941
18.5	Separable Piecewise Linear Approximations	944
18.6	Benders Decomposition for Nonseparable Approximations**	946
18.6.1	Benders' Decomposition for Two-Stage Problems	947
18.6.2	Asymptotic Analysis of Benders with Regularization**	952
18.6.3	Benders with Regularization	956
18.7	Linear Approximations for High-Dimensional Applications	956
18.8	Resource Allocation with Exogenous Information State	958
18.9	Closing Notes	959
18.10	Bibliographic Notes	960
	Exercises	962
	Bibliography	967
19	Direct Lookahead Policies	971
19.1	Optimal Policies Using Lookahead Models	974
19.2	Creating an Approximate Lookahead Model	978
19.2.1	Modeling the Lookahead Model	979
19.2.2	Strategies for Approximating the Lookahead Model	980
19.3	Modified Objectives in Lookahead Models	985
19.3.1	Managing Risk	985
19.3.2	Utility Functions for Multiobjective Problems	991
19.3.3	Model Discounting	992
19.4	Evaluating DLA Policies	992
19.4.1	Evaluating Policies in a Simulator	994
19.4.2	Evaluating Risk-Adjusted Policies	994
19.4.3	Evaluating Policies in the Field	996
19.4.4	Tuning Direct Lookahead Policies	997
19.5	Why Use a DLA?	997
19.6	Deterministic Lookaheads	999
19.6.1	A Deterministic Lookahead: Shortest Path Problems	1001

19.6.2	Parameterized Lookaheads	1003
19.7	A Tour of Stochastic Lookahead Policies	1005
19.7.1	Lookahead PFAs	1005
19.7.2	Lookahead CFAs	1007
19.7.3	Lookahead VFAs for the Lookahead Model	1007
19.7.4	Lookahead DLAs for the Lookahead Model	1008
19.7.5	Discussion	1009
19.8	Monte Carlo Tree Search for Discrete Decisions	1009
19.8.1	Basic Idea	1010
19.8.2	The Steps of MCTS	1010
19.8.3	Discussion	1014
19.8.4	Optimistic Monte Carlo Tree Search	1016
19.9	Two-Stage Stochastic Programming for Vector Decisions*	1018
19.9.1	The Basic Two-Stage Stochastic Program	1018
19.9.2	Two-Stage Approximation of a Sequential Problem	1020
19.9.3	Discussion	1023
19.10	Observations on DLA Policies	1024
19.11	Bibliographic Notes	1025
	Exercises	1027
	Bibliography	1031

Part VI – Multiagent Systems 1033

20	Multiagent Modeling and Learning	1035
20.1	Overview of Multiagent Systems	1036
20.1.1	Dimensions of a Multiagent System	1036
20.1.2	Communication	1038
20.1.3	Modeling a Multiagent System	1040
20.1.4	Controlling Architectures	1043
20.2	A Learning Problem – Flu Mitigation	1044
20.2.1	Model 1: A Static Model	1045
20.2.2	Variations of Our Flu Model	1046
20.2.3	Two-Agent Learning Models	1050
20.2.4	Transition Functions for Two-Agent Model	1052
20.2.5	Designing Policies for the Flu Problem	1054
20.3	The POMDP Perspective*	1059
20.4	The Two-Agent Newsvendor Problem	1062
20.5	Multiple Independent Agents – An HVAC Controller Model	1067

20.5.1	Model	1067
20.5.2	Designing Policies	1069
20.6	Cooperative Agents – A Spatially Distributed Blood Management Problem	1070
20.7	Closing Notes	1074
20.8	Why Does it Work?	1074
20.8.1	Derivation of the POMDP Belief Transition Function	1074
20.9	Bibliographic Notes	1076
	Exercises	1077
	Bibliography	1083
	Index	1085

Preface

Preface to *Reinforcement Learning and Stochastic Optimization: A unified framework for sequential decisions*

This book represents a lifetime of research into what I now call sequential decision problems, which dates to 1982 when I was introduced to the problem arising in truckload trucking (think of Uber/Lyft for trucks) where we have to choose which driver to assign to a load, and which loads to accept to move, given the high level of randomness in future customer demands, representing requests to move full truckloads of freight.

It took me 20 years to figure out a practical algorithm to solve this problem, which led to my first book (in 2007) on approximate dynamic programming, where the major breakthrough was the introduction of the post-decision state and the use of hierarchical aggregation for approximating value functions to solve these high-dimensional problems. However, I would argue today that the most important chapter in the book (and I recognized it at the time), was chapter 5 on how to model these problems, without any reference to algorithms to solve the problem. I identified five elements to a sequential decision problem, leading up to the objective function which was written

$$\max_{\pi} \mathbb{E} \left\{ \sum_{t=0}^T C(S_t, X^{\pi}(S_t)) | S_0 \right\}.$$

It was not until the second edition (in 2011) that I realized that approximate dynamic programming (specifically, policies that depend on value functions) was not the only way to solve these problems; rather, there were four classes of policies, and only one used value functions. The 2011 edition of the book listed three of the four classes of policies that are described in this book, but most of the book still focused on approximating value functions. It was not until a 2014

paper (“Clearing the Jungle of Stochastic Optimization”) that I identified the four classes of policies I use now. Then, in 2016 I realized that the four classes of policies could be divided between two major strategies: the policy search strategy, where we search over a family of functions to find the one that works best, and the lookahead strategy, where we make good decisions by approximating the downstream impact of a decision made now.

Finally, I combined these ideas in a 2019 paper (“A Unified Framework for Stochastic Optimization” published in the *European Journal for Operational Research*) with a better appreciation of major classes of problems such as state-independent problems (the pure learning problems that include derivative-based and derivative-free stochastic search) and the more general state-dependent problems; cumulative and final reward objective functions; and the realization that any adaptive search algorithm was a sequential decision problem. The material in the 2019 paper is effectively the outline for this book.

This book builds on the 2011 edition of my approximate dynamic programming book, and includes a number of chapters (some heavily edited) from the ADP book. It would be nice to call this a third edition, but the entire framework of this book is completely different. “Approximate dynamic programming” is a term that still refers to making decisions based on the idea of approximating the downstream value of being in a state. After decades of working with this approach (which is still covered over a span of five chapters in this volume), I can now say with confidence that value function approximations, despite all the attention they have received, is a powerful methodology for a surprisingly narrow set of decision problems.

By contrast, I finally developed the confidence to claim that the four classes of policies are universal. This means that *any* method for making decisions will fall in one of these four classes, or a hybrid of two or more. This is a game changer, because it shifts the focus from an algorithm (the method for making decisions) to the model (specifically the optimization problem above, along with the state-transition function and the model of the exogenous information process). This means we write out the elements of a problem *before* we tackle the problem of designing policies to decisions. I call this:

Model first, then solve.

The communities working on sequential decision problems are very focused on methods, just as I was with my earlier work with approximate dynamic programming. The problem is that any particular method will be inherently limited to a narrow class of problems. In this book, I demonstrate how you can

take a simple inventory problem, and then tweak the data to make each of the four classes work best.

This new approach has opened up an entirely new way of approaching a problem class that, in the last year of writing the book, I started calling “sequential decision analytics,” which is any problem consisting of the sequence:

Decision, information, decision, information,

I allow decisions to range from binary (selling an asset) to discrete choices (favored in computer science) to the high-dimensional resource allocation problems popular in operations research. This approach starts with a problem, shifts to the challenging task of modeling uncertainty, and then finishes with designing policies to make decisions to optimize some metric. The approach is practical, scalable, and universally applicable.

It is exciting to be able to create a single framework that spans 15 different communities, and which represents every possible method for solving sequential decision problems. While having a common language to model any sequential decision problem, combined with the general approach of the four classes of policies, is clearly of value, this framework has been developed by standing on the shoulders of the giants who have laid the foundational work for all of these methods. I have had to make choices regarding the best notation and modeling conventions, but my framework is completely inclusive of all the methods that have been developed to solve these problems. Rather than joining the chorus of researchers promoting specific algorithmic strategies (as I once did), my goal is to raise the visibility of all methods, so that someone looking to solve a real problem is working with the biggest possible toolbox, rather than just the tools developed within a specific community.

A word needs to be said about the title of the book. As this is being written, there is a massive surge of interest in “reinforcement learning,” which started as a form of approximate dynamic programming (I used to refer to ADP and RL as similar to American English and British English). However, as the RL community has grown and started working on harder problems, they encountered the same experience that I and everyone else working in ADP found: value function approximations are not a panacea. Not only is it the case that they often do not work, they usually do not work. As a result, the RL community branched out (just as I did) into other methods such as “policy gradient methods” (my “policy function approximations” or PFA), upper confidence bounding (a form of “cost function approximation” or CFA), the original Q-learning (which produces a policy based on “value function approximations” or VFA), and finally

Monte Carlo tree search (a policy based on “direct lookahead approximations” or DLA). All of these methods are found in the second edition of Sutton and Barto’s landmark book *Reinforcement Learning: An introduction*, but only as specific methods rather than general classes. This book takes the next step and identifies the general classes.

This evolution from one core method to all four classes of policies is being repeated among other fields that I came to call the “jungle of stochastic optimization.” Stochastic search, simulation-optimization, and bandit problems all feature methods from each of the four classes of policies. Over time, I came to realize that all these fields (including reinforcement learning) were playing catchup to the grandfather of all of this work, which is optimal control (and stochastic control). The field of optimal control was the first to introduce and seriously explore the use of value function approximations (they call these cost-to-go functions), linear decision rules (a form of PFA), and the workhorse “model predictive control” (a great name for a simple rolling horizon procedure, which is a “direct lookahead approximation” in this book). I also found that my modeling framework was closest to that used in the optimal control literature, which was the first field to introduce the concept of a transition function, a powerful modeling device that has been largely overlooked by the other communities. I make a few small tweaks such as using state S_t instead of x_t , and decision x_t (widely used in the field of math programming) instead of u_t .

Then I introduce one big change, which is to maximize over all four classes of policies. Perhaps the most important innovation of this book is to break the almost automatic link between optimizing over policies, and then assuming that we are going to compute an optimal policy from either Bellman’s equation or the Hamilton-Jacobi equations. These are rarely computable for real problems, which then leads people to assume that the natural next step is to approximate these equations. This is simply false, supported by decades of research where people have developed methods that do not depend on HJB equations. I recognize this body of research developing different classes of policies by making the inclusion of all four classes of policies fundamental to the original statement of the optimization problem above.

It will take some time for people from the different communities to learn to speak this common language. More likely, there will be an adaptation of existing modeling languages to this framework. For example, the optimal control community could keep their notation, but learn to write their objective functions as I have above, recognizing that the search over policies needs to span all four classes (which, I might point out, they are already using). I would hope that the reinforcement learning community, which adopted the notation for discrete action a , might learn to use the more general x (as the bandit community has already done).

I have tried to write this book to appeal to newcomers to the field, as well as people who already have training in one or more of the subfields that deal with decisions and uncertainty; recognizing these two broad communities was easily the biggest challenge while writing this book. Not surprisingly, the book is quite long. I have tried to make it more accessible to people who are new to the field by marking many sections with an * as an indication that this section can be skipped on a first-read. I also hope that the book will appeal to people from many application domains. However, the core audience is people who are looking to solve real problems by modeling applications and implementing the work in software. The notation is designed to facilitate writing computer programs, where there should be a direct relationship between the mathematical model and the software. This is particularly important when modeling the flow of information, something that is often overlooked in mainstream reinforcement learning papers.

Warren B. Powell

Princeton, New Jersey
August, 2021