Contents

List of Figures xvi
List of Tables xxi
Acknowledgments xxiii
Preface xxv

1. BACKGROUND 1
   1.1 Why this book was written 1
   1.2 Simulation-based optimization and modern times 3
   1.3 How this book is organized 7

2. NOTATION 9
   2.1 Chapter Overview 9
   2.2 Some Basic Conventions 9
   2.3 Vector notation 9
      2.3.1 Max norm 10
      2.3.2 Euclidean norm 10
   2.4 Notation for matrices 10
   2.5 Notation for n-tuples 11
   2.6 Notation for sets 11
   2.7 Notation for Sequences 11
   2.8 Notation for Transformations 11
   2.9 Max, min, and arg max 12
   2.10 Acronyms and Abbreviations 12
   2.11 Concluding Remarks 12

3. PROBABILITY THEORY: A REFRESHER 15
   3.1 Overview of this chapter 15
      3.1.1 Random variables 15
   3.2 Laws of Probability 16
### Contents

6.3.2.4 Piecewise regression 65  
6.3.2.5 Fitting non-linear forms 66  
6.3.3 How good is the metamodel? 67  
6.3.4 Optimization with a metamodel 68  
6.4 Neuro-Response Surface Methods 69  
6.4.1 Linear Neural Networks 69  
6.4.1.1 Steps in the Widrow-Hoff Algorithm 72  
6.4.1.2 Incremental Widrow-Hoff 72  
6.4.1.3 Pictorial Representation of a Neuron 73  
6.4.2 Non-linear Neural Networks 73  
6.4.2.1 The Basic Structure of a Non-Linear Neural Network 75  
6.4.2.2 The Backprop Algorithm 78  
6.4.2.3 Deriving the backprop algorithm 79  
6.4.2.4 Backprop with a Bias Node 82  
6.4.2.5 Deriving the algorithm for the bias weight 82  
6.4.2.6 Steps in Backprop 84  
6.4.2.7 Incremental Backprop 86  
6.4.2.8 Example D 88  
6.4.2.9 Validation of the neural network 89  
6.4.2.10 Optimization with a neuro-RSM model 90  
6.5 Concluding Remarks 90  
6.6 Bibliographic Remarks 90  
6.7 Review Questions 91  

7. PARAMETRIC OPTIMIZATION 93  
7.1 Chapter Overview 93  
7.2 Continuous Optimization 94  
7.2.1 Gradient Descent 94  
7.2.1.1 Simulation and Gradient Descent 98  
7.2.1.2 Simultaneous Perturbation 101  
7.2.2 Non-derivative methods 104  
7.3 Discrete Optimization 106  
7.3.1 Ranking and Selection 107  
7.3.1.1 Steps in the Rinott method 108  
7.3.1.2 Steps in the Kim-Nelson method 109  
7.3.2 Meta-heuristics 110  
7.3.2.1 Simulated Annealing 111  
7.3.2.2 The Genetic Algorithm 117  
7.3.2.3 Tabu Search 119  
7.3.2.4 A Learning Automata Search Technique 123  
7.3.2.5 Other Meta-Heuristics 128  
7.3.2.6 Ranking and selection & meta-heuristics 128  
7.4 Hybrid solution spaces 128  
7.5 Concluding Remarks 129
## Bibliographic Remarks

7.6 Bibliographic Remarks 129

## Review Questions

7.7 Review Questions 131

### 8. Dynamic Programming

8.1 Chapter Overview 133

8.2 Stochastic processes 133

8.3 Markov processes, Markov chains and semi-Markov processes 136

8.3.1 Markov chains 139

8.3.1.1 $n$-step transition probabilities 140

8.3.2 Regular Markov chains 142

8.3.2.1 Limiting probabilities 143

8.3.3 Ergodicity 145

8.3.4 Semi-Markov processes 146

8.4 Markov decision problems 148

8.4.1 Elements of the Markov decision framework 151

8.5 How to solve an MDP using exhaustive enumeration 157

8.5.1 Example A 158

8.5.2 Drawbacks of exhaustive enumeration 161

8.6 Dynamic programming for average reward 161

8.6.1 Average reward Bellman equation for a policy 162

8.6.2 Policy iteration for average reward MDPs 163

8.6.2.1 Steps 163

8.6.3 Value iteration and its variants: average reward MDPs 165

8.6.4 Value iteration for average reward MDPs 165

8.6.4.1 Steps 166

8.6.5 Relative value iteration 168

8.6.5.1 Steps 168

8.6.6 A general expression for the average reward of an MDP 169

8.7 Dynamic programming and discounted reward 170

8.7.1 Discounted reward 171

8.7.2 Discounted reward MDP 171

8.7.3 Bellman equation for a policy: discounted reward 173

8.7.4 Policy iteration for discounted reward MDPs 173

8.7.4.1 Steps 174

8.7.5 Value iteration for discounted reward MDPs 175

8.7.5.1 Steps 176

8.7.6 Getting value iteration to converge faster 177

8.7.6.1 Gauss Siedel value iteration 178

8.7.6.2 Relative value iteration for discounted reward 179

8.7.6.3 Span seminorm termination 180

8.8 The Bellman equation: An intuitive perspective 181

8.9 Semi-Markov decision problems 182

8.9.1 The natural process and the decision-making process 184

8.9.2 Average reward SMDPs 186
## Contents

8.9.2.1 Exhaustive enumeration for average reward SMDPs 186  
8.9.2.2 Example B 187  
8.9.2.3 Policy iteration for average reward SMDPs 189  
8.9.2.4 Value iteration for average reward SMDPs 191  
8.9.2.5 Counterexample for regular value iteration 192  
8.9.2.6 Uniformization for SMDPs 193  
8.9.2.7 Value iteration based on the Bellman equation 194  
8.9.2.8 Extension to random time SMDPs 194  
8.9.3 Discounted reward SMDPs 194  
8.9.3.1 Policy iteration for discounted SMDPs 195  
8.9.3.2 Value iteration for discounted reward SMDPs 195  
8.9.3.3 Extension to random time SMDPs 196  
8.9.3.4 Uniformization 196  

8.10 Modified policy iteration 197  
8.10.1 Steps for discounted reward MDPs 198  
8.10.2 Steps for average reward MDPs 199  

8.11 Miscellaneous topics related to MDPs and SMDPs 200  
8.11.1 A parametric-optimization approach to solving MDPs 200  
8.11.2 The MDP as a special case of a stochastic game 201  
8.11.3 Finite Horizon MDPs 203  
8.11.4 The approximating sequence method 206  

8.12 Conclusions 207  
8.13 Bibliographic Remarks 207  
8.14 Review Questions 208  

9. REINFORCEMENT LEARNING 211  
9.1 Chapter Overview 211  
9.2 The Need for Reinforcement Learning 212  
9.3 Generating the TPM through straightforward counting 214  
9.4 Reinforcement Learning: Fundamentals 215  
9.4.1 \( Q \)-factors 218  
9.4.1.1 A \( Q \)-factor version of value iteration 219  
9.4.2 The Robbins-Monro algorithm 220  
9.4.3 The Robbins-Monro algorithm and \( Q \)-factors 221  
9.4.4 Simulators, asynchronous implementations, and step sizes 222  
9.5 Discounted reward Reinforcement Learning 224  
9.5.1 Discounted reward RL based on value iteration 224  
9.5.1.1 Steps in \( Q \)-Learning 225  
9.5.1.2 Reinforcement Learning: A “Learning” Perspective 227  
9.5.1.3 On-line and Off-line 229  
9.5.1.4 Exploration 230  
9.5.1.5 A worked-out example for \( Q \)-Learning 231  
9.5.2 Discounted reward RL based on policy iteration 234
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.5.2.1</td>
<td>$Q$-factor version of regular policy iteration</td>
<td>235</td>
</tr>
<tr>
<td>9.5.2.2</td>
<td>Steps in the $Q$-factor version of regular policy iteration</td>
<td>235</td>
</tr>
<tr>
<td>9.5.2.3</td>
<td>Steps in $Q$-$P$-Learning</td>
<td>237</td>
</tr>
<tr>
<td>9.6</td>
<td>Average reward Reinforcement Learning</td>
<td>238</td>
</tr>
<tr>
<td>9.6.1</td>
<td>Discounted RL for average reward MDPs</td>
<td>238</td>
</tr>
<tr>
<td>9.6.2</td>
<td>Average reward RL based on value iteration</td>
<td>238</td>
</tr>
<tr>
<td>9.6.2.1</td>
<td>Steps in Relative $Q$-Learning</td>
<td>239</td>
</tr>
<tr>
<td>9.6.2.2</td>
<td>Calculating the average reward of a policy in a simulator</td>
<td>240</td>
</tr>
<tr>
<td>9.6.3</td>
<td>Other algorithms for average reward MDPs</td>
<td>241</td>
</tr>
<tr>
<td>9.6.3.1</td>
<td>Steps in $R$-Learning</td>
<td>241</td>
</tr>
<tr>
<td>9.6.3.2</td>
<td>Steps in SMART for MDPs</td>
<td>242</td>
</tr>
<tr>
<td>9.6.4</td>
<td>An RL algorithm based on policy iteration</td>
<td>244</td>
</tr>
<tr>
<td>9.6.4.1</td>
<td>Steps in $Q$-$P$-Learning for average reward</td>
<td>244</td>
</tr>
<tr>
<td>9.7</td>
<td>Semi-Markov decision problems and RL</td>
<td>245</td>
</tr>
<tr>
<td>9.7.1</td>
<td>Discounted Reward</td>
<td>245</td>
</tr>
<tr>
<td>9.7.1.1</td>
<td>Steps in $Q$-Learning for discounted reward DTMDPs</td>
<td>245</td>
</tr>
<tr>
<td>9.7.1.2</td>
<td>Steps in $Q$-$P$-Learning for discounted reward DTMDPs</td>
<td>246</td>
</tr>
<tr>
<td>9.7.2</td>
<td>Average reward</td>
<td>247</td>
</tr>
<tr>
<td>9.7.2.1</td>
<td>Steps in SMART for SMDPs</td>
<td>248</td>
</tr>
<tr>
<td>9.7.2.2</td>
<td>Steps in $Q$-$P$-Learning for SMDPs</td>
<td>250</td>
</tr>
<tr>
<td>9.8</td>
<td>RL Algorithms and their DP counterparts</td>
<td>252</td>
</tr>
<tr>
<td>9.9</td>
<td>Actor-Critic Algorithms</td>
<td>252</td>
</tr>
<tr>
<td>9.10</td>
<td>Model-building algorithms</td>
<td>253</td>
</tr>
<tr>
<td>9.10.1</td>
<td>$H$-Learning for discounted reward</td>
<td>254</td>
</tr>
<tr>
<td>9.10.2</td>
<td>$H$-Learning for average reward</td>
<td>255</td>
</tr>
<tr>
<td>9.10.3</td>
<td>Model-building $Q$-Learning</td>
<td>257</td>
</tr>
<tr>
<td>9.10.4</td>
<td>Model-building relative $Q$-Learning</td>
<td>258</td>
</tr>
<tr>
<td>9.11</td>
<td>Finite Horizon Problems</td>
<td>259</td>
</tr>
<tr>
<td>9.12</td>
<td>Function approximation</td>
<td>260</td>
</tr>
<tr>
<td>9.12.1</td>
<td>Function approximation with state aggregation</td>
<td>260</td>
</tr>
<tr>
<td>9.12.2</td>
<td>Function approximation with function fitting</td>
<td>262</td>
</tr>
<tr>
<td>9.12.2.1</td>
<td>Difficulties</td>
<td>262</td>
</tr>
<tr>
<td>9.12.2.2</td>
<td>Steps in $Q$-Learning coupled with neural networks</td>
<td>264</td>
</tr>
<tr>
<td>9.12.3</td>
<td>Function approximation with interpolation methods</td>
<td>265</td>
</tr>
<tr>
<td>9.12.4</td>
<td>Linear and non-linear functions</td>
<td>269</td>
</tr>
<tr>
<td>9.12.5</td>
<td>A robust strategy</td>
<td>269</td>
</tr>
<tr>
<td>9.12.6</td>
<td>Function approximation: Model-building algorithms</td>
<td>270</td>
</tr>
<tr>
<td>9.13</td>
<td>Conclusions</td>
<td>270</td>
</tr>
<tr>
<td>9.14</td>
<td>Bibliographic Remarks</td>
<td>271</td>
</tr>
<tr>
<td>9.14.1</td>
<td>Early works</td>
<td>271</td>
</tr>
<tr>
<td>9.14.3</td>
<td>RL algorithms based on $Q$-factors</td>
<td>271</td>
</tr>
<tr>
<td>9.14.4</td>
<td>Actor-critic Algorithms</td>
<td>272</td>
</tr>
<tr>
<td>9.14.5</td>
<td>Model-building algorithms</td>
<td>272</td>
</tr>
</tbody>
</table>
12.2.5 Taylor’s theorem 320
12.3 Convergence of gradient-descent approaches 323
12.4 Perturbation Estimates 327
  12.4.1 Finite Difference Estimates 327
  12.4.2 Notation 328
  12.4.3 Simultaneous Perturbation Estimates 328
12.5 Convergence of Simulated Annealing 333
12.6 Concluding Remarks 341
12.7 Bibliographic Remarks 341
12.8 Review Questions 341

13. CONVERGENCE: CONTROL OPTIMIZATION 343
13.1 Chapter Overview 343
13.2 Dynamic programming transformations 344
13.3 Some definitions 345
13.4 Monotonicity of $T, T_{\mu}, L$, and $L_{\mu}$ 346
13.5 Some results for average & discounted MDPs 347
13.6 Discounted reward and classical dynamic programming 349
  13.6.1 Bellman Equation for Discounted Reward 349
  13.6.2 Policy Iteration 356
  13.6.3 Value iteration for discounted reward MDPs 359
13.7 Average reward and classical dynamic programming 364
  13.7.1 Bellman equation for average reward 365
  13.7.2 Policy iteration for average reward MDPs 368
  13.7.3 Value Iteration for average reward MDPs 372
13.8 Convergence of DP schemes for SMDPs 379
13.9 Convergence of Reinforcement Learning Schemes 379
13.10 Background Material for RL Convergence 380
  13.10.1 Non-Expansive Mappings 380
  13.10.2 Lipschitz Continuity 380
  13.10.3 Convergence of a sequence with probability 1 381
13.11 Key Results for RL convergence 381
  13.11.1 Synchronous Convergence 382
  13.11.2 Asynchronous Convergence 383
13.12 Convergence of RL based on value iteration 392
  13.12.1 Convergence of $Q$-Learning 392
  13.12.2 Convergence of Relative $Q$-Learning 397
  13.12.3 Finite Convergence of $Q$-Learning 397
13.13 Convergence of $Q$-$P$-Learning for MDPs 400
  13.13.1 Discounted reward 400
  13.13.2 Average Reward 401
13.14 SMDPs 402
### Contents

13.14.1 Value iteration for average reward 402  
13.14.2 Policy iteration for average reward 402  
13.15 Convergence of Actor-Critic Algorithms 404  
13.16 Function approximation and convergence analysis 405  
13.17 Bibliographic Remarks 406  
13.17.1 DP theory 406  
13.17.2 RL theory 406  
13.18 Review Questions 407  

14. CASE STUDIES 409  
14.1 Chapter Overview 409  
14.2 A Classical Inventory Control Problem 410  
14.3 Airline Yield Management 412  
14.4 Preventive Maintenance 416  
14.5 Transfer Line Buffer Optimization 420  
14.6 Inventory Control in a Supply Chain 423  
14.7 AGV Routing 424  
14.8 Quality Control 426  
14.9 Elevator Scheduling 427  
14.10 Simulation optimization: A comparative perspective 429  
14.11 Concluding Remarks 430  
14.12 Review Questions 430  

15. CODES 433  
15.1 Introduction 433  
15.2 C programming 434  
15.3 Code Organization 436  
15.4 Random Number Generators 437  
15.5 Simultaneous Perturbation 439  
15.6 Dynamic Programming Codes 441  
15.6.1 Policy Iteration for average reward MDPs 442  
15.6.2 Relative Iteration for average reward MDPs 447  
15.6.3 Policy Iteration for discounted reward MDPs 450  
15.6.4 Value Iteration for discounted reward MDPs 453  
15.6.5 Policy Iteration for average reward SMDPs 460  
15.7 Codes for Neural Networks 464  
15.7.1 Neuron 465  
15.7.2 Backprop Algorithm — Batch Mode 470  
15.8 Reinforcement Learning Codes 478  
15.8.1 Codes for Q-Learning 478  
15.8.2 Codes for Relative Q-Learning 486  
15.8.3 Codes for Relaxed-SMART 495