

Optimization Algorithms

AI TECHNIQUES FOR DESIGN,
PLANNING, AND CONTROL PROBLEMS

ALAA KHAMIS



MANNING
SHELTER ISLAND

contents

preface x
acknowledgments xii
about this book xiv
about the author xviii
about the cover illustration xx

PART I DETERMINISTIC SEARCH ALGORITHMS.....1

1

Introduction to search and optimization 3

- 1.1 Why care about search and optimization? 5
- 1.2 Going from toy problems to the real world 6
- 1.3 Basic ingredients of optimization problems 7
 - Decision variables* 9 ■ *Objective functions* 10
 - Constraints* 14
- 1.4 Well-structured problems vs. ill-structured problems 15
 - Well-structured problems* 15 ■ *Ill-structured problems* 16
 - WSP, but ISP in practice* 18
- 1.5 Search algorithms and the search dilemma 20

2

A deeper look at search and optimization 23

- 2.1 Classifying optimization problems 24
 - Number and type of decision variables* 26 ■ *Landscape and number of objective functions* 32 ■ *Constraints* 37 ■ *Linearity of objective functions and constraints* 40 ■ *Expected quality and permissible time for the solution* 45

- 2.2 Classifying search and optimization algorithms 50
- 2.3 Heuristics and metaheuristics 52
- 2.4 Nature-inspired algorithms 59

3

***Blind search algorithms* 62**

- 3.1 Introduction to graphs 63
- 3.2 Graph search 72
- 3.3 Graph traversal algorithms 74
 - Breadth-first search* 74 ■ *Depth-first search* 84
- 3.4 Shortest path algorithms 89
 - Dijkstra's search* 90 ■ *Uniform-cost search (UCS)* 94
 - Bidirectional Dijkstra's search* 96
- 3.5 Applying blind search to the routing problem 98

4

***Informed search algorithms* 103**

- 4.1 Introducing informed search 104
- 4.2 Minimum spanning tree algorithms 105
- 4.3 Shortest path algorithms 114
 - Hill climbing algorithm* 115 ■ *Beam search algorithm* 121
 - A* search algorithm* 124 ■ *Hierarchical approaches* 129
- 4.4 Applying informed search to a routing problem 146
 - Hill climbing for routing* 146 ■ *Beam search for routing* 148
 - A* for routing* 149 ■ *Contraction hierarchies for routing* 151

PART 2 TRAJECTORY-BASED ALGORITHMS..... 155

5

***Simulated annealing* 157**

- 5.1 Introducing trajectory-based optimization 158
- 5.2 The simulated annealing algorithm 159
 - Physical annealing* 159 ■ *SA pseudocode* 161 ■ *Acceptance probability* 165 ■ *The annealing process* 168 ■ *Adaptation in SA* 172
- 5.3 Function optimization 175
- 5.4 Solving Sudoku 180

- 5.5 Solving TSP 185
- 5.6 Solving a delivery semi-truck routing problem 190

6 **Tabu search 195**

- 6.1 Local search 196
- 6.2 Tabu search algorithm 197
 - Memory structure 201* ■ *Aspiration criteria 205* ■ *Adaptation in TS 205*
- 6.3 Solving constraint satisfaction problems 207
- 6.4 Solving continuous problems 213
- 6.5 Solving TSP and routing problems 215
- 6.6 Assembly line balancing problem 221

PART 3 EVOLUTIONARY COMPUTING ALGORITHMS 231

7 **Genetic algorithms 233**

- 7.1 Population-based metaheuristic algorithms 234
- 7.2 Introducing evolutionary computation 241
 - A brief recap of biology fundamentals 241* ■ *The theory of evolution 242* ■ *Evolutionary computation 243*
- 7.3 Genetic algorithm building blocks 246
 - Fitness function 249* ■ *Representation schemes 249*
 - Selection operators 251* ■ *Reproduction operators 257*
 - Survivor selection 260*
- 7.4 Implementing genetic algorithms in Python 262

8 **Genetic algorithm variants 271**

- 8.1 Gray-coded GA 272
- 8.2 Real-valued GA 275
 - Crossover methods 275* ■ *Mutation methods 278*
- 8.3 Permutation-based GA 282
 - Crossover methods 283* ■ *Mutation methods 290*
- 8.4 Multi-objective optimization 291
- 8.5 Adaptive GA 298

- 8.6 Solving the traveling salesman problem 300
- 8.7 PID tuning problem 304
- 8.8 Political districting problem 312

PART 4 SWARM INTELLIGENCE ALGORITHMS 319

9 *Particle swarm optimization* 321

- 9.1 Introducing swarm intelligence 322
- 9.2 Continuous PSO 325
 - Motion equations* 328 ■ *Fitness update* 330
 - Initialization* 332 ■ *Neighborhoods* 333
- 9.3 Binary PSO 339
- 9.4 Permutation-based PSO 342
- 9.5 Adaptive PSO 343
 - Inertia weight* 344 ■ *Cognitive and social components* 346
- 9.6 Solving the traveling salesman problem 348
- 9.7 Neural network training using PSO 351

10 *Other swarm intelligence algorithms to explore* 362

- 10.1 Nature's tiny problem-solvers 363
- 10.2 ACO metaheuristics 366
- 10.3 ACO variants 369
 - Simple ACO* 370 ■ *Ant system* 373 ■ *Ant colony system* 374 ■ *Max-min ant system* 375 ■ *Solving open TSP with ACO* 376
- 10.4 From hive to optimization 383
- 10.5 Exploring the artificial bee colony algorithm 385

PART 5 MACHINE LEARNING-BASED METHODS 395

11 *Supervised and unsupervised learning* 397

- 11.1 A day in the life of AI-empowered daily routines 398
- 11.2 Demystifying machine learning 399

- 11.3 Machine learning with graphs 403
 - Graph embedding* 407 ■ *Attention mechanisms* 415 ■ *Pointer networks* 418
- 11.4 Self-organizing maps 422
- 11.5 Machine learning for optimization problems 424
- 11.6 Solving function optimization using supervised machine learning 427
- 11.7 Solving TSP using supervised graph machine learning 432
- 11.8 Solving TSP using unsupervised machine learning 438
- 11.9 Finding a convex hull 441

12 **Reinforcement learning** 450

- 12.1 Demystifying reinforcement learning 451
 - Markov decision process (MDP)* 452 ■ *From MDP to reinforcement learning* 453 ■ *Model-based versus model-free RL* 458 ■ *Actor-critic methods* 459 ■ *Proximal policy optimization* 460 ■ *Multi-armed bandit (MAB)* 464
- 12.2 Optimization with reinforcement learning 470
- 12.3 Balancing CartPole using A2C and PPO 473
- 12.4 Autonomous coordination in mobile networks using PPO 481
- 12.5 Solving the truck selection problem using contextual bandits 485
- 12.6 Journey's end: A final reflection 490
- references* 492
- index* 497

The following bonus appendices are available in the ePub and Kindle versions of this book, and you can read them online in liveBook:

- appendix A Search and optimization libraries in Python*
- appendix B Benchmarks and datasets*
- appendix C Exercises and solutions*