

Contents

| | |
|---|-----------|
| Preface | xi |
| 1. Introduction | 1 |
| 1.1. Variants of the linear programming problem | 2 |
| 1.2. Examples of linear programming problems | 6 |
| 1.3. Piecewise linear convex objective functions | 15 |
| 1.4. Graphical representation and solution | 21 |
| 1.5. Linear algebra background and notation | 26 |
| 1.6. Algorithms and operation counts | 32 |
| 1.7. Exercises | 34 |
| 1.8. History, notes, and sources | 38 |
| 2. The geometry of linear programming | 41 |
| 2.1. Polyhedra and convex sets | 42 |
| 2.2. Extreme points, vertices, and basic feasible solutions | 46 |
| 2.3. Polyhedra in standard form | 53 |
| 2.4. Degeneracy | 58 |
| 2.5. Existence of extreme points | 62 |
| 2.6. Optimality of extreme points | 65 |
| 2.7. Representation of bounded polyhedra* | 67 |
| 2.8. Projections of polyhedra: Fourier-Motzkin elimination* | 70 |
| 2.9. Summary | 75 |
| 2.10. Exercises | 75 |
| 2.11. Notes and sources | 79 |
| 3. The simplex method | 81 |
| 3.1. Optimality conditions | 82 |
| 3.2. Development of the simplex method | 87 |
| 3.3. Implementations of the simplex method | 91 |

| viii | Contents | |
|-----------|--|------------|
| 3.4. | Anticycling: lexicography and Bland's rule | 108 |
| 3.5. | Finding an initial basic feasible solution | 111 |
| 3.6. | Column geometry and the simplex method | 119 |
| 3.7. | Computational efficiency of the simplex method | 124 |
| 3.8. | Summary | 128 |
| 3.9. | Exercises | 129 |
| 3.10. | Notes and sources | 137 |
| 4. | Duality theory | 139 |
| 4.1. | Motivation | 140 |
| 4.2. | The dual problem | 142 |
| 4.3. | The duality theorem | 146 |
| 4.4. | Optimal dual variables as marginal costs | 155 |
| 4.5. | Standard form problems and the dual simplex method | 156 |
| 4.6. | Farkas' lemma and linear inequalities | 165 |
| 4.7. | From separating hyperplanes to duality* | 169 |
| 4.8. | Cones and extreme rays | 174 |
| 4.9. | Representation of polyhedra | 179 |
| 4.10. | General linear programming duality* | 183 |
| 4.11. | Summary | 186 |
| 4.12. | Exercises | 187 |
| 4.13. | Notes and sources | 199 |
| 5. | Sensitivity analysis | 201 |
| 5.1. | Local sensitivity analysis | 202 |
| 5.2. | Global dependence on the right-hand side vector | 212 |
| 5.3. | The set of all dual optimal solutions* | 215 |
| 5.4. | Global dependence on the cost vector | 216 |
| 5.5. | Parametric programming | 217 |
| 5.6. | Summary | 221 |
| 5.7. | Exercises | 222 |
| 5.8. | Notes and sources | 229 |
| 6. | Large scale optimization | 231 |
| 6.1. | Delayed column generation | 232 |
| 6.2. | The cutting stock problem | 234 |
| 6.3. | Cutting plane methods | 236 |
| 6.4. | Dantzig-Wolfe decomposition | 239 |
| 6.5. | Stochastic programming and Benders decomposition | 254 |
| 6.6. | Summary | 260 |
| 6.7. | Exercises | 260 |
| 6.8. | Notes and sources | 263 |

| Contents | ix |
|---|------------|
| 7. Network flow problems | 265 |
| 7.1. Graphs | 267 |
| 7.2. Formulation of the network flow problem | 272 |
| 7.3. The network simplex algorithm | 278 |
| 7.4. The negative cost cycle algorithm | 291 |
| 7.5. The maximum flow problem | 301 |
| 7.6. Duality in network flow problems | 312 |
| 7.7. Dual ascent methods* | 316 |
| 7.8. The assignment problem and the auction algorithm | 325 |
| 7.9. The shortest path problem | 332 |
| 7.10. The minimum spanning tree problem | 343 |
| 7.11. Summary | 345 |
| 7.12. Exercises | 347 |
| 7.13. Notes and sources | 356 |
| 8. Complexity of linear programming and the ellipsoid method | 359 |
| 8.1. Efficient algorithms and computational complexity | 360 |
| 8.2. The key geometric result behind the ellipsoid method | 363 |
| 8.3. The ellipsoid method for the feasibility problem | 370 |
| 8.4. The ellipsoid method for optimization | 378 |
| 8.5. Problems with exponentially many constraints* | 380 |
| 8.6. Summary | 387 |
| 8.7. Exercises | 388 |
| 8.8. Notes and sources | 392 |
| 9. Interior point methods | 393 |
| 9.1. The affine scaling algorithm | 395 |
| 9.2. Convergence of affine scaling* | 404 |
| 9.3. The potential reduction algorithm | 409 |
| 9.4. The primal path following algorithm | 419 |
| 9.5. The primal-dual path following algorithm | 431 |
| 9.6. An overview | 438 |
| 9.7. Exercises | 440 |
| 9.8. Notes and sources | 448 |
| 10. Integer programming formulations | 451 |
| 10.1. Modeling techniques | 452 |
| 10.2. Guidelines for strong formulations | 461 |
| 10.3. Modeling with exponentially many constraints | 465 |
| 10.4. Summary | 472 |
| 10.5. Exercises | 472 |

| | | |
|------------|--|------------|
| 10.6. | Notes and sources | 477 |
| 11. | Integer programming methods | 479 |
| 11.1. | Cutting plane methods | 480 |
| 11.2. | Branch and bound | 485 |
| 11.3. | Dynamic programming | 490 |
| 11.4. | Integer programming duality | 494 |
| 11.5. | Approximation algorithms | 507 |
| 11.6. | Local search | 511 |
| 11.7. | Simulated annealing | 512 |
| 11.8. | Complexity theory | 514 |
| 11.9. | Summary | 522 |
| 11.10. | Exercises | 523 |
| 11.11. | Notes and sources | 530 |
| 12. | The art in linear optimization | 533 |
| 12.1. | Modeling languages for linear optimization | 534 |
| 12.2. | Linear optimization libraries and general observations | 535 |
| 12.3. | The fleet assignment problem | 537 |
| 12.4. | The air traffic flow management problem | 544 |
| 12.5. | The job shop scheduling problem | 551 |
| 12.6. | Summary | 562 |
| 12.7. | Exercises | 563 |
| 12.8. | Notes and sources | 567 |
| | References | 569 |
| | Index | 579 |

Preface

The purpose of this book is to provide a unified, insightful, and modern treatment of linear optimization, that is, linear programming, network flow problems, and discrete linear optimization. We discuss both classical topics, as well as the state of the art. We give special attention to theory, but also cover applications and present case studies. Our main objective is to help the reader become a sophisticated practitioner of (linear) optimization, or a researcher. More specifically, we wish to develop the ability to formulate fairly complex optimization problems, provide an appreciation of the main classes of problems that are practically solvable, describe the available solution methods, and build an understanding of the qualitative properties of the solutions they provide.

Our general philosophy is that insight matters most. For the subject matter of this book, this necessarily requires a geometric view. On the other hand, problems are solved by algorithms, and these can only be described algebraically. Hence, our focus is on the beautiful interplay between algebra and geometry. We build understanding using figures and geometric arguments, and then translate ideas into algebraic formulas and algorithms. Given enough time, we expect that the reader will develop the ability to pass from one domain to the other without much effort.

Another of our objectives is to be comprehensive, but economical. We have made an effort to cover and highlight all of the principal ideas in this field. However, we have not tried to be encyclopedic, or to discuss every possible detail relevant to a particular algorithm. Our premise is that once mature understanding of the basic principles is in place, further details can be acquired by the reader with little additional effort.

Our last objective is to bring the reader up to date with respect to the state of the art. This is especially true in our treatment of interior point methods, large scale optimization, and the presentation of case studies that stretch the limits of currently available algorithms and computers.

The success of any optimization methodology hinges on its ability to deal with large and important problems. In that sense, the last chapter, on the art of linear optimization, is a critical part of this book. It will, we hope, convince the reader that progress on challenging problems requires both problem specific insight, as well as a deeper understanding of the underlying theory.