



Linear Optimization and Extensions: Theory and Algorithms

Shu-Cherng Fang
North Carolina State University

Sarat Puthenpura
AT&T Bell Labs



Prentice-Hall International, Inc.

This edition may be sold only in those countries to which it is consigned by Prentice-Hall International. It is not to be re-exported and it is not for sale in the U.S.A., Mexico, or Canada.



© 1993 by AT&T. All Rights Reserved
Published by Prentice-Hall, Inc.
A Simon & Schuster Company
Englewood Cliffs, New Jersey 07632

The authors and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher make no warranty of any kind, expressed or implied, with regard to these programs or the documentation contained in this book. The author and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs.

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

ISBN 0-13-037581-0

Prentice-Hall International (UK) Limited, London
Prentice-Hall of Australia Pty. Limited, Sydney
Prentice-Hall Canada Inc., Toronto
Prentice-Hall Hispanoamericana S.A., Mexico
Prentice-Hall of India Private Limited, New Delhi
Prentice-Hall of Japan, Inc., Tokyo
Simon & Schuster Asia Pte. Ltd., Singapore
Editora Prentice-Hall do Brasil, Ltda., Rio de Janeiro
Prentice-Hall, Inc., Englewood Cliffs, New Jersey

*Dedicated to our families:
Chi-Hsin Chao Fang
Mini and Vidya Puthenpura*

Contents

PREFACE	xiii
1 INTRODUCTION	1
1.1 History of Linear Programming	1
1.2 The Linear Programming Problem	2
1.2.1 Standard-Form Linear Program	3
1.2.2 Embedded Assumptions	3
1.2.3 Converting to Standard Form	4
1.3 Examples of Linear Programming Problems	5
1.4 Mastering Linear Programming	9
References for Further Reading	10
Exercises	11
2 GEOMETRY OF LINEAR PROGRAMMING	14
2.1 Basic Terminologies of Linear Programming	14
2.2 Hyperplanes, Halfspaces, and Polyhedral Sets	15
2.3 Affine Sets, Convex Sets, and Cones	17
	vii

2.4	Extreme Points and Basic Feasible Solutions	19
2.5	Nondegeneracy and Adjacency	21
2.6	Resolution Theorem for Convex Polyhedrons	23
2.7	Fundamental Theorem of Linear Programming	24
2.8	Concluding Remarks: Motivations of Different Approaches	25
	References for Further Reading	26
	Exercises	26

3 THE REVISED SIMPLEX METHOD

29

3.1	Elements of an Iterative Scheme	29
3.2	Basics of the Simplex Method	30
3.3	Algebra of the Simplex Method	31
	3.3.1 Stopping the Simplex Method—Checking for Optimality,	33
	3.3.2 Iterations of the Simplex Method—Moving for Improvement,	33
3.4	Starting the Simplex Method	39
	3.4.1 Two-Phase Method,	39
	3.4.2 Big-M Method,	41
3.5	Degeneracy and Cycling	42
3.6	Preventing Cycling	44
	3.6.1 Lexicographic Rule,	44
	3.6.2 Bland's Rule,	44
3.7	The Revised Simplex Method	45
3.8	Concluding Remarks	50
	References for Further Reading	50
	Exercises	51

4 DUALITY THEORY AND SENSITIVITY ANALYSIS

55

4.1	Dual Linear Program	56
4.2	Duality Theory	57

4.3	Complementary Slackness and Optimality Conditions	61
4.4	An Economic Interpretation of the Dual Problem	63
	4.4.1 Dual Variables and Shadow Prices,	63
	4.4.2 Interpretation of the Dual Problem,	64
4.5	The Dual Simplex Method	65
	4.5.1 Basic Idea of the Dual Simplex Method,	65
	4.5.2 Sherman-Morrison-Woodbury Formula,	66
	4.5.3 Computer Implementation of the Dual Simplex Method,	70
	4.5.4 Find an Initial Dual Basic Feasible Solution,	72
4.6	The Primal Dual Method	73
	4.6.1 Step-by-Step Procedure for the Primal-Dual Simplex Method,	75
4.7	Sensitivity Analysis	78
	4.7.1 Change in the Cost Vector,	78
	4.7.2 Change in the Right-Hand-Side Vector,	80
	4.7.3 Change in the Constraint Matrix,	82
4.8	Concluding Remarks	86
	References for Further Reading	87
	Exercises	87

5 COMPLEXITY ANALYSIS AND THE ELLIPSOID METHOD

92

5.1	Concepts of Computational Complexity	93
5.2	Complexity of the Simplex Method	94
5.3	Basic Ideas of the Ellipsoid Method	96
5.4	Ellipsoid Method for Linear Programming	100
5.5	Performance of the Ellipsoid Method for LP	103
5.6	Modifications of the Basic Algorithm	104
	5.6.1 Deep Cuts,	104
	5.6.2 Surrogate Cuts,	106
	5.6.3 Parallel Cuts,	106
	5.6.4 Replacing Ellipsoid by Simplex,	107
5.7	Concluding Remarks	108
	References for Further Reading	108
	Exercises	109

6	KARMARKAR'S PROJECTIVE SCALING ALGORITHM	112
6.1	Basic Ideas of Karmarkar's Algorithm	112
6.2	Karmarkar's Standard Form	114
6.2.1	<i>The Simplex Structure, 115</i>	
6.2.2	<i>Projective Transformation on the Simplex, 116</i>	
6.3	Karmarkar's Projective Scaling Algorithm	117
6.4	Polynomial-Time Solvability	120
6.5	Converting to Karmarkar's Standard Form	126
6.6	Handling Problems with Unknown Optimal Objective Values	128
6.7	Unconstrained Convex Dual Approach	135
6.7.1	<i>ϵ-Optimal Solution, 136</i>	
6.7.2	<i>Extension, 139</i>	
6.8	Concluding Remarks	141
	References for Further Reading	141
	Exercises	142
7	AFFINE SCALING ALGORITHMS	144
7.1	Primal Affine Scaling Algorithm	145
7.1.1	<i>Basic Ideas of Primal Affine Scaling, 145</i>	
7.1.2	<i>Implementing the Primal Affine Scaling Algorithm, 155</i>	
7.1.3	<i>Computational Complexity, 160</i>	
7.2	Dual Affine Scaling Algorithm	165
7.2.1	<i>Basic Ideas of Dual Affine Scaling, 165</i>	
7.2.2	<i>Dual Affine Scaling Algorithm, 167</i>	
7.2.3	<i>Implementing the Dual Affine Scaling Algorithm, 169</i>	
7.2.4	<i>Improving Computational Complexity, 172</i>	
7.3	The Primal-Dual Algorithm	177
7.3.1	<i>Basic Ideas of the Primal-Dual Algorithm, 178</i>	
7.3.2	<i>Direction and Step-Length of Movement, 180</i>	
7.3.3	<i>Primal-Dual Algorithm, 184</i>	
7.3.4	<i>Polynomial-Time Termination, 184</i>	
7.3.5	<i>Starting the Primal-Dual Algorithm, 188</i>	
7.3.6	<i>Practical Implementation, 189</i>	

7.3.7	<i>Accelerating via Power-Series Method, 193</i>	
7.4	Concluding Remarks	194
	References for Further Reading	195
	Exercises	197
8	INSIGHTS INTO THE INTERIOR-POINT METHODS	201
8.1	Moving Along Different Algebraic Paths	201
8.1.1	<i>Primal Affine Scaling with Logarithmic Barrier Function, 203</i>	
8.1.2	<i>Dual Affine Scaling with Logarithmic Barrier Function, 204</i>	
8.1.3	<i>The Primal-Dual Algorithm, 205</i>	
8.2	Missing Information	207
8.2.1	<i>Dual Information in the Primal Approach, 207</i>	
8.2.2	<i>Primal Information in the Dual Approach, 207</i>	
8.3	Extensions of Algebraic Paths	208
8.4	Geometric Interpretation of the Moving Directions	209
8.4.1	<i>Primal Affine Scaling with Logarithmic Barrier Function, 211</i>	
8.4.2	<i>Dual Affine Scaling with Logarithmic Barrier Function, 212</i>	
8.4.3	<i>The Primal-Dual Algorithm, 213</i>	
8.5	General Theory	217
8.5.1	<i>General Primal Affine Scaling, 217</i>	
8.5.2	<i>General Dual Affine Scaling, 219</i>	
8.6	Concluding Remarks	220
	References for Further Reading	221
	Exercises	221
9	AFFINE SCALING FOR CONVEX QUADRATIC PROGRAMMING	224
9.1	Convex Quadratic Program with Linear Constraints	225
9.1.1	<i>Primal Quadratic Program, 225</i>	
9.1.2	<i>Dual Quadratic Program, 225</i>	
9.2	Affine Scaling for Quadratic Programs	227
9.2.1	<i>Primal Affine Scaling for Quadratic Programming, 227</i>	

9.2.2	<i>Improving Primal Affine Scaling for Quadratic Programming, 237</i>	
9.3	Primal-Dual Algorithm for Quadratic Programming	241
9.3.1	<i>Basic Concepts, 241</i>	
9.3.2	<i>A Step-By-Step Implementation Procedure, 243</i>	
9.3.3	<i>Convergence Properties of the Primal-Dual Algorithm, 245</i>	
9.4	Convex Programming with Linear Constraints	246
9.4.1	<i>Basic Concepts, 246</i>	
9.4.2	<i>A Step-by-Step Implementation Procedure, 248</i>	
9.5	Concluding Remarks	249
	References for Further Reading	249
	Exercises	250
10	IMPLEMENTATION OF INTERIOR-POINT ALGORITHMS	253
10.1	The Computational Bottleneck	253
10.2	The Cholesky Factorization Method	254
10.2.1	<i>Computing Cholesky Factor, 255</i>	
10.2.2	<i>Block Cholesky Factorization, 257</i>	
10.2.3	<i>Sparse Cholesky Factorization, 259</i>	
10.2.4	<i>Symbolic Cholesky Factorization, 263</i>	
10.2.5	<i>Solving Triangular Systems, 263</i>	
10.3	The Conjugate Gradient Method	265
10.4	The LQ Factorization Method	268
10.5	Concluding Remarks	275
	References for Further Reading	276
	Exercises	277
	BIBLIOGRAPHY	280
	INDEX	295

Preface

Since G. B. Dantzig first proposed the celebrated simplex method around 1947, the wide applicability of linear programming models and the evolving mathematical theory and computational methodology under these models have attracted an immense amount of interest from both practitioners and academicians. In particular, in 1979, L. G. Khachian proved that the ellipsoid method of N. Z. Shor, D. B. Yudin, and A. S. Nemirovskii could outperform the simplex method in theory by exhibiting polynomial-time performance; and, in 1984, N. Karmarkar designed a polynomial-time interior-point algorithm that rivals the simplex method even in practice. These three methods present different and yet fundamental approaches to solving linear optimization problems.

This book provides a unified view that treats the simplex, ellipsoid, and interior-point methods in an integrated manner. It is written primarily as a textbook for those graduate students who are interested in learning state-of-the-art techniques in the area of linear programming and its natural extensions. In addition, the authors hope it will serve as a useful handbook for people who pursue research and development activities in the relatively new field of interior-point methods for optimization.

We have organized the book into ten chapters. In the first chapter, we introduce the linear programming problems with modeling examples and provide a short review of the history of linear programming. In the second chapter, basic terminologies are defined to build the fundamental theory of linear programming and to form a geometric interpretation of the underlying optimization process. The third chapter covers the classical simplex method—in particular, the revised simplex method. Duality theory, the dual simplex method, the primal-dual method, and sensitivity analysis are the topics of Chapter 4. In the fifth chapter, we look into the concept of computational complexity and show that the simplex method, in the worst-case analysis, exhibits exponential

complexity. Hence the ellipsoid method is introduced as the first polynomial-time algorithm for linear programming. From this point onward, we focus on the nonsimplex approaches. Naturally, the sixth chapter is centered around the recent advances of Karmarkar's algorithm and its polynomial-time solvability. Chapter 7 essentially covers the affine scaling variants, including the primal, dual, and primal-dual algorithms, of Karmarkar's method. The concepts of central trajectory and path-following are also included. The eighth chapter reveals the insights of interior-point methods from both the algebraic and geometric viewpoints. It provides a platform for the comparison of different interior-point algorithms and the creation of new algorithms. In Chapter 9, we extend the results of interior-point-based linear programming techniques to quadratic and convex optimization problems with linear constraints. The important implementation issues for computer programming are addressed in the last chapter. Without understanding these issues, it is impossible to have serious software development that achieves the expected computational performance.

The authors see three key elements in mastering linear optimization and its extensions, namely, (1) the intuitions generated by geometric interpretation, (2) the properties proven by algebraic expressions, and (3) the algorithms validated by computer implementation; and the book is written with emphasis on both theory and algorithms. Hence it is implied that a user of this book should have some basic understanding in mathematical analysis, linear algebra, and numerical methods. Since an ample number of good reference books are available in the market, we decided not to include additional mathematical preliminaries.

This book pays special attention to the practical implementation of algorithms. Time has proven that the practical value of an algorithm, and hence its importance among practitioners, is largely determined by its numerical performance including robustness, convergence rate, and ease of computer implementation. With the advent of digital computer technology, iterative solution methods for optimization have become extremely popular. Actually, this book explains various algorithms in the framework of an iterative scheme with three principal aspects: (a) how to obtain a starting solution, (b) how to check if a current solution is optimal, and (c) how to move to an improved solution. We have attempted to cast all the algorithms discussed in the book within the purview of this philosophy. In this manner, computer implementation follows naturally.

The material in this book has been used by the authors to teach several graduate courses at North Carolina State University, University of Pennsylvania, and Rutgers University since 1988. According to our experience, Chapters 1 through 6 together with a brief touch of Chapter 7 comprise the material for a one-semester first graduate course in Linear Programming. A review of Chapters 3 and 5 together with Chapters 6 through 10 could serve for another one-semester course in Advanced Linear Programming, or Special Topics on Interior-Point Methods. This book can also be used as a "cookbook" for computer implementation of various optimization algorithms, without actually going deep into the theoretical aspects. For this purpose, after introducing each algorithm, we have included a step-by-step implementation recipe.

We have tried to incorporate the most salient results on the subject matter into this book. Despite our efforts, however, owing to the tremendous ongoing research activities

in the field of interior-point methods, we may have unintentionally left out some of the important and recent work in the area.

ACKNOWLEDGMENTS

Writing this book has been a long and challenging task. We could not have carried on this endeavor without persistent help and encouragement from our colleagues and friends, in addition to our families. The first and foremost of such people is Mr. Steve Chen, Head of the Global Network and Switched Services Planning Department, AT&T Bell Laboratories. He envisioned the importance of this work and provided us with tremendous support in time, equipment, periodic suggestions for improving the book, and every other aspect one can think of. Also, in particular, we wish to thank Professors Romesh Saigal (University of Michigan), Jong-Shi Pang (Johns Hopkins University), Jie Sun (Northwestern University), Robert J. Vanderbei (Princeton University), and Yin-Yu Ye (University of Iowa) for reviewing our book proposal and/or drafts; Professor Salah E. Elmaghraby (North Carolina State University) for encouraging and scheduling one of us in the teaching of linear programming courses; Professor Elmor L. Peterson (North Carolina State University) for his invaluable advisory work; Dr. L. P. Sinha and Mr. W. Radwill (AT&T Bell Laboratories) for their valuable support and constant encouragement. Also, successful completion of this work would not have been possible without the support we received from Dr. Phyllis Weiss (AT&T Bell Laboratories). Besides, we would like to thank Drs. Jun-Min Liu, Lev Slutsman, David Houck Jr., Mohan Gawande, Gwo-Min Jan (AT&T Bell Laboratories), and Dr. Ruey-Lin Sheu (North Carolina State University) for their constructive suggestions. We express also the greatest appreciation to those students who have tolerated the unpolished manuscript and helped us improve the quality of this book. The final thanks go to Dr. Bruce Loftis of the North Carolina Supercomputing Center, the Cray Research Grants, and our publishers in Prentice Hall.

Shu-Cherng Fang
Raleigh, North Carolina

Sarat Puthenpura
Murray Hill, New Jersey

***Linear Optimization
and Extensions***