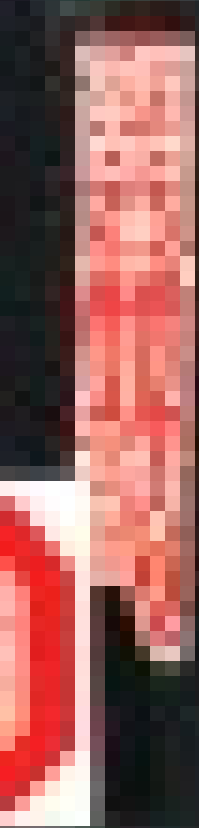


2

13

14



PRACTICAL OPTIMIZATION

PHILIP E. GILL
WALTER MURRAY
MARGARET H. WRIGHT

*Systems Optimization Laboratory
Department of Operations Research
Stanford University
California, USA*

ACADEMIC PRESS INC (LONDON) LTD
24/28 Oval Road,
London NW1

United States Edition published by
ACADEMIC PRESS INC
111 Fifth Avenue
New York, New York 10003

Copyright © 1981 by
ACADEMIC PRESS INC (LONDON) LTD
Second Printing 1982

All rights reserved
No part of this book may be reproduced in any form by photostat, microfilm, or any other
means, without written permission from the publishers

British Library Cataloguing in Publication Data
Gill, P E

Practical optimization

1 Mathematics optimization

I Title II Murray, W III Wnght, M H
515 QA402 5

ISBN 0-12-283952 8

LCCCN 81-66366

Printed in Great Britain by,
St Edmundsbury Press, Bury St Edmunds, Suffolk

PREFACE

As researchers at the National Physical Laboratory and Stanford University, and as contributors to the Numerical Algorithms Group (NAG) software library, we have been involved for many years in the development of numerical methods and software for the solution of optimization problems. Within the past twenty years, there has been a dramatic increase in the efficiency and reliability of optimization methods for almost all problem categories. However, this improved capability has been achieved by the use of more complicated ideas, particularly from the areas of numerical linear algebra and finite-precision calculation.

The best methods available today are extremely complex, and their manner of operation is far from obvious, especially to users from other disciplines. This book is intended as a treatment — necessarily broad — of the subject of *practical optimization*. The word “practical” is included in the title in order to convey our concern not only with the motivation for optimization methods, but also with details of implementation that affect the performance of a method in practice. In particular, we believe that some consideration of the effects of finite-precision computation is essential in order for any description of a method to be useful. We also believe that it is important to discuss the linear algebraic processes that are used to perform certain portions of all optimization methods.

This book is meant to be largely self-contained; we have therefore devoted one chapter to a description of the essential results from numerical linear algebra and the analysis of rounding errors in computation, and a second chapter to a treatment of optimality conditions.

Selected methods for unconstrained, linearly constrained, and nonlinearly constrained optimization are described in three chapters. This discussion is intended to present an overview of the methods, including the underlying motivation as well as particular theoretical and computational features. Illustrations have been used wherever possible in order to stress the geometric interpretation of the methods. The methods discussed are primarily those with which we have had extensive experience and success; other methods are described that provide special insights or background. References to methods not discussed and to further details are given in the extensive Notes and Bibliography at the end of each section. The methods have been presented in sufficient detail to allow this book to be used as a text for a university-level course in numerical optimization.

Two chapters are devoted to selected less formal, but nonetheless crucial, topics that might be viewed as “advice to users”. For example, some suggestions concerning modelling are included because we have observed that an understanding of optimization methods can have a beneficial effect on the modelling of the activities to be optimized. In addition, we have presented an extensive discussion of topics that are crucial in using and understanding a numerical optimization method — such as selecting a method, interpreting the computed results, and diagnosing (and, if possible, curing) difficulties that may cause an algorithm to fail or perform poorly.

In writing this book, the authors have had the benefit of advice and help from many people. In particular, we offer special thanks to our friend and colleague Michael Saunders, not only for many helpful comments on various parts of the book, but also for all of his characteristic good humour and patience when the task of writing this book seemed to go on forever. He has played a major role in much of the recent work on algorithms for large-scale problems described in Chapters 5 and 6.

We gratefully acknowledge David Martin for his tireless help and support for the optimization group at the National Physical Laboratory, and George Dantzig for his efforts to assemble the algorithms group at the Systems Optimization Laboratory, Stanford University.

We thank Brian Hinde, Susan Hodson, Enid Long and David Rhead for their work in developing and testing many of the algorithms described in this book.

We are grateful to Greg Dobson, David Fuchs, Stefania Gai, Richard Stone and Wes Winkler, who have been helpful in many ways during the preparation of this manuscript. The clarity of certain parts of the text has been improved because of helpful comments from Paulo Benevides-Soares, Andy Conn, Laureano Escudero, Don Iglehart, James Lyness, Jorge Moré, Michael Overton and Danny Sorensen.

We thank Clive Hall for producing the computer-generated figures on the Laserscan plotter at the National Physical Laboratory. The remaining diagrams were expertly drawn by Nancy Cimina.

This book was typeset by the authors using the \TeX mathematical typesetting system of Don Knuth*. We are grateful to Don Knuth for his efforts in devising the \TeX system, and for making available to us various \TeX macros that improved the quality of the final text. We also thank him for kindly allowing us to use the Alphatype CRS typesetter, and Chris Tucci for his substantial help in producing the final copy.

Finally, we offer our deepest personal thanks to those closest to us, who have provided encouragement, support and humour during the time-consuming process of writing this book.

Stanford University
May, 1981

P. E. G.
W. M.
M. H. W.

*D. E. Knuth, *TEX and METAFONT, New Directions in Typesetting*, American Mathematical Society and Digital Press, Bedford, Massachusetts (1979).

CONTENTS

Those sections marked with “*” contain material of a rather specialized nature that may be omitted on first reading.

CHAPTER 1. INTRODUCTION	1
1.1. DEFINITION OF OPTIMIZATION PROBLEMS	1
1.2. CLASSIFICATION OF OPTIMIZATION PROBLEMS	3
1.3. OVERVIEW OF TOPICS	5
CHAPTER 2. FUNDAMENTALS	7
2.1. INTRODUCTION TO ERRORS IN NUMERICAL COMPUTATION	7
2.1.1. Measurement of Error	7
2.1.2. Number Representation on a Computer	8
2.1.3. Rounding Errors	9
2.1.4. Errors Incurred during Arithmetic Operations	11
2.1.5. Cancellation Error	11
2.1.6. Accuracy in a Sequence of Calculations	13
2.1.7. Error Analysis of Algorithms	13
Notes and Selected Bibliography for Section 2.1	14
2.2. INTRODUCTION TO NUMERICAL LINEAR ALGEBRA	14
2.2.1. Preliminaries	14
2.2.1.1. Scalars	14
2.2.1.2. Vectors	14
2.2.1.3. Matrices	15
2.2.1.4. Operations with vectors and matrices	16
2.2.1.5. Matrices with special structure	18
2.2.2. Vector Spaces	19
2.2.2.1. Linear combinations	19
2.2.2.2. Linear dependence and independence	20
2.2.2.3. Vector spaces; subspaces; basis	21
2.2.2.4. The null space	22
2.2.3. Linear Transformations	23
2.2.3.1. Matrices as transformations	23
2.2.3.2. Properties of linear transformations	23
2.2.3.3. Inverses	24
2.2.3.4. Eigenvalues; eigenvectors	24

Table of Contents

2.2.3.5.	Definiteness	25
2.2.4.	Linear Equations	25
2.2.4.1.	Properties of linear equations	25
2.2.4.2.	Vector and matrix norms	27
2.2.4.3.	Perturbation theory; condition number	28
2.2.4.4.	Triangular linear systems	30
2.2.4.5.	Error analysis	31
2.2.5.	Matrix Factorizations	32
2.2.5.1.	The LU factorization; Gaussian elimination	33
2.2.5.2.	The LDL^T and Cholesky factorizations	36
2.2.5.3.	The QR factorization	37
2.2.5.4.	The spectral decomposition of a symmetric matrix	40
2.2.5.5.	Singular-value decomposition	40
2.2.5.6.	The pseudo-inverse	41
2.2.5.7.	Updating matrix factorizations	41
2.2.6.	Multi-dimensional Geometry	43
	Notes and Selected Bibliography for Section 2.2	45
2.3.	ELEMENTS OF MULTIVARIATE ANALYSIS	45
2.3.1.	Functions of Many Variables; Contour Plots	45
2.3.2.	Continuous Functions and their Derivatives	46
2.3.3.	Order Notation	52
2.3.4.	Taylor's Theorem	52
2.3.5.	Finite-Difference Approximations to Derivatives	54
2.3.6.	Rates of Convergence of Iterative Sequences	56
	Notes and Selected Bibliography for Section 2.3	58
CHAPTER 3.	OPTIMALITY CONDITIONS	59
3.1.	CHARACTERIZATION OF A MINIMUM	59
3.2.	UNCONSTRAINED OPTIMIZATION	61
3.2.1.	The Univariate Case	61
3.2.2.	The Multivariate Case	63
3.2.3.	Properties of Quadratic Functions	65
3.3.	LINEARLY CONSTRAINED OPTIMIZATION	67
3.3.1.	Linear Equality Constraints	68
3.3.2.	Linear Inequality Constraints	71
3.3.2.1.	General optimality conditions	71
3.3.2.2.	Linear programming	75
3.3.2.3.	Quadratic programming	76
3.3.2.4.	Optimization subject to bounds	77
3.4.	NONLINEARLY CONSTRAINED OPTIMIZATION	77
3.4.1.	Nonlinear Equality Constraints	78
3.4.2.	Nonlinear Inequality Constraints	81
	Notes and Selected Bibliography for Section 3.4	82

CHAPTER 4. UNCONSTRAINED METHODS	83
4.1. METHODS FOR UNIVARIATE FUNCTIONS	83
4.1.1. Finding a Zero of a Univariate Function	83
4.1.1.1. The method of bisection	84
4.1.1.2. Newton's method	84
4.1.1.3. Secant and regula falsi methods	85
*4.1.1.4. Rational interpolation and higher-order methods	87
4.1.1.5. Safeguarded zero-finding algorithms	87
4.1.2. Univariate Minimization	88
4.1.2.1. Fibonacci search	89
4.1.2.2. Golden section search	90
4.1.2.3. Polynomial interpolation	91
4.1.2.4. Safeguarded polynomial interpolation	92
Notes and Selected Bibliography for Section 4.1	92
4.2. METHODS FOR MULTIVARIATE NON-SMOOTH FUNCTIONS	93
4.2.1. Use of Function Comparison Methods	93
4.2.2. The Polytope Algorithm	94
*4.2.3. Composite Non-Differentiable Functions	96
Notes and Selected Bibliography for Section 4.2	98
4.3. METHODS FOR MULTIVARIATE SMOOTH FUNCTIONS	99
4.3.1. A Model Algorithm for Smooth Functions	99
4.3.2. Convergence of the Model Algorithm	99
4.3.2.1. Computing the step length	100
4.3.2.2. Computing the direction of search	102
Notes and Selected Bibliography for Section 4.3	104
4.4. SECOND DERIVATIVE METHODS	105
4.4.1. Newton's Method	105
4.4.2. Strategies for an Indefinite Hessian	107
4.4.2.1. A method based on the spectral decomposition	107
*4.4.2.2. Methods based on the Cholesky factorization	108
Notes and Selected Bibliography for Section 4.4	111
4.5. FIRST DERIVATIVE METHODS	115
4.5.1. Discrete Newton Methods	115
4.5.2. Quasi-Newton Methods	116
4.5.2.1. Theory	116
4.5.2.2. Implementation	122
*4.5.2.3. Convergence; least-change characterization	123
Notes and Selected Bibliography for Section 4.5	125
4.6. NON-DERIVATIVE METHODS FOR SMOOTH FUNCTIONS	127
4.6.1. Finite-Difference Approximations to First Derivatives	127
4.6.1.1. Errors in a forward-difference approximation	127
4.6.1.2. Choice of the finite-difference interval	128
4.6.1.3. Estimation of a set of finite-difference intervals	129
4.6.1.4. The choice of finite-difference formulae	130

Table of Contents

4.6.2.	Non-Derivative Quasi-Newton Methods	131
	Notes and Selected Bibliography for Section 4.6	131
4.7.	METHODS FOR SUMS OF SQUARES	133
4.7.1.	Origin of Least-Squares Problems; the Reason for Special Methods	133
4.7.2.	The Gauss-Newton Method	134
4.7.3.	The Levenberg-Marquardt Method	136
*4.7.4.	Quasi-Newton Approximations	137
*4.7.5.	The Corrected Gauss-Newton Method	138
*4.7.6.	Nonlinear Equations	139
	Notes and Selected Bibliography for Section 4.7	140
4.8.	METHODS FOR LARGE-SCALE PROBLEMS	141
4.8.1.	Sparse Discrete Newton Methods	141
*4.8.2.	Sparse Quasi-Newton Methods	143
4.8.3.	Conjugate-Gradient Methods	144
4.8.3.1.	Quadratic functions	144
4.8.3.2.	The linear conjugate-gradient method	146
4.8.3.3.	General nonlinear functions	147
*4.8.3.4.	Conjugate-gradient methods with restarts	149
*4.8.3.5.	Convergence	149
*4.8.4.	Limited-Memory Quasi-Newton Methods	150
*4.8.5.	Preconditioned Conjugate-Gradient Methods	151
*4.8.5.1.	Quadratic functions	151
*4.8.5.2.	Nonlinear functions	152
*4.8.6.	Solving the Newton Equation by Linear Conjugate-Gradients	153
	Notes and Selected Bibliography for Section 4.8	153
CHAPTER 5. LINEAR CONSTRAINTS		155
5.1.	METHODS FOR LINEAR EQUALITY CONSTRAINTS	155
5.1.1.	The Formulation of Algorithms	156
5.1.1.1.	The effect of linear equality constraints	156
5.1.1.2.	A model algorithm	157
5.1.2.	Computation of the Search Direction	158
5.1.2.1.	Methods of steepest descent	158
5.1.2.2.	Second derivative methods	159
5.1.2.3.	Discrete Newton methods	160
5.1.2.4.	Quasi-Newton methods	160
5.1.2.5.	Conjugate-gradient-related methods	161
5.1.3.	Representation of the Null Space of the Constraints	162
5.1.3.1.	The LQ factorization	162
5.1.3.2.	The variable-reduction technique	163
5.1.4.	Special Forms of the Objective Function	163
5.1.4.1.	Linear objective function	163
5.1.4.2.	Quadratic objective function	164
5.1.5.	Lagrange Multiplier Estimates	164

Table of Contents

5.1.5.1.	First-order multiplier estimates	165
5.1.5.2.	Second-order multiplier estimates	166
	Notes and Selected Bibliography for Section 5.1	166
5.2.	ACTIVE SET METHODS FOR LINEAR INEQUALITY CONSTRAINTS	167
5.2.1.	A Model Algorithm	168
5.2.2.	Computation of the Search Direction and Step Length	169
5.2.3.	Interpretation of Lagrange Multiplier Estimates	170
*5.2.4.	Changes in the Working Set	172
*5.2.4.1.	Modification of Z	172
*5.2.4.2.	Modification of other matrices	173
	Notes and Selected Bibliography for Section 5.2	174
5.3.	SPECIAL PROBLEM CATEGORIES	176
5.3.1.	Linear Programming	176
5.3.2.	Quadratic Programming	177
5.3.2.1.	Positive-definite quadratic programming	177
5.3.2.2.	Indefinite quadratic programming	178
*5.3.3.	Linear Least-Squares with Linear Constraints	180
	Notes and Selected Bibliography for Section 5.3	181
*5.4.	PROBLEMS WITH FEW GENERAL LINEAR CONSTRAINTS	182
*5.4.1.	Positive-Definite Quadratic Programming	183
*5.4.2.	Second Derivative Methods	184
*5.4.2.1.	A method based on positive-definite quadratic programming	184
*5.4.2.2.	A method based on an approximation of the projected Hessian	185
	Notes and Selected Bibliography for Section 5.4	185
5.5.	SPECIAL FORMS OF THE CONSTRAINTS	186
5.5.1.	Minimization Subject to Simple Bounds	186
*5.5.2.	Problems with Mixed General Linear Constraints and Bounds	188
	Notes and Selected Bibliography for Section 5.5	190
5.6.	LARGE-SCALE LINEARLY CONSTRAINED OPTIMIZATION	190
5.6.1.	Large-scale Linear Programming	190
5.6.2.	General large-scale linearly constrained optimization	193
*5.6.2.1.	Computation of the change in the superbasic variables	194
*5.6.2.2.	Changes in the active set	195
	Notes and Selected Bibliography for Section 5.6	196
*5.7.	FINDING AN INITIAL FEASIBLE POINT	198
	Notes and Selected Bibliography for Section 5.7	199
*5.8.	IMPLEMENTATION OF ACTIVE SET METHODS	199
*5.8.1.	Finding the Initial Working Set	199
*5.8.2.	Linearly Dependent Constraints	201
*5.8.3.	Zero Lagrange Multipliers	201
	Notes and Selected Bibliography for Section 5.8	203

CHAPTER 6. NONLINEAR CONSTRAINTS	205
6.1. THE FORMULATION OF ALGORITHMS	206
6.1.1. The Definition of a Merit Function	206
6.1.2. The Nature of Subproblems	206
6.1.2.1. Adaptive and deterministic subproblems	206
6.1.2.2. Valid and defective subproblems	207
6.2. PENALTY AND BARRIER FUNCTION METHODS	207
6.2.1. Differentiable Penalty and Barrier Function Methods	207
6.2.1.1. The quadratic penalty function	208
6.2.1.2. The logarithmic barrier function	212
6.2.2. Non-Differentiable Penalty Function Methods	214
6.2.2.1. The absolute value penalty function	215
6.2.2.2. A method for general non-differentiable problems	217
Notes and Selected Bibliography for Section 6.2	218
6.3. REDUCED-GRADIENT AND GRADIENT-PROJECTION METHODS	219
6.3.1. Motivation for Reduced-Gradient-Type Methods	219
6.3.2. Definition of a Reduced-Gradient-Type Method	220
6.3.2.1. Definition of the null-space component	220
6.3.2.2. Restoration of feasibility	222
6.3.2.3. Reduction of the objective function	222
6.3.2.4. Properties of reduced-gradient-type methods	223
6.3.3. Determination of the Working Set	223
Notes and Selected Bibliography for Section 6.3	224
6.4. AUGMENTED LAGRANGIAN METHODS	225
6.4.1. Formulation of an Augmented Lagrangian Function	225
6.4.2. An Augmented Lagrangian Algorithm	226
6.4.2.1. A model algorithm	227
6.4.2.2. Properties of the augmented Lagrangian function	228
*6.4.3. Variations in Strategy	230
Notes and Selected Bibliography for Section 6.4	231
6.5. PROJECTED LAGRANGIAN METHODS	233
6.5.1. Motivation for a Projected Lagrangian Method	233
6.5.1.1. Formulation of a linearly constrained subproblem	233
6.5.1.2. Definition of the subproblem	233
6.5.2. A General Linearly Constrained Subproblem	234
6.5.2.1. Formulation of the objective function	234
6.5.2.2. A simplified model algorithm	235
*6.5.2.3. Improvements to the model algorithm	236
6.5.3. A Quadratic Programming Subproblem	237
6.5.3.1. Motivation	237
6.5.3.2. A simplified model algorithm	238
6.5.3.3. Use of a merit function	240
*6.5.3.4. Other formulations of the subproblem	241
*6.5.4. Strategies for a Defective Subproblem	242
*6.5.4.1. Incompatible linear constraints	242

Table of Contents

*6.5.4.2. Poor approximation of the Lagrangian function	243
*6.5.5. Determination of the Active Set	243
*6.5.5.1. An equality-constrained subproblem	244
*6.5.5.2. An inequality-constrained subproblem	244
Notes and Selected Bibliography for Section 6.5	245
6.6. LAGRANGE MULTIPLIER ESTIMATES	247
6.6.1. First-Order Multiplier Estimates	248
6.6.2. Second-Order Multiplier Estimates	248
*6.6.3. Multiplier Estimates for Inequality Constraints	250
6.6.4. Consistency Checks	250
Notes and Selected Bibliography for Section 6.6	251
*6.7. LARGE-SCALE NONLINEARLY CONSTRAINED OPTIMIZATION	251
*6.7.1. The Use of a Linearly Constrained Subproblem	252
*6.7.2. The Use of a QP Subproblem	253
*6.7.2.1. Representing the basis inverse	254
*6.7.2.2. The search direction for the superbasic variables	255
Notes and Selected Bibliography for Section 6.7	256
6.8. SPECIAL PROBLEM CATEGORIES	256
6.8.1. Special Non-Differentiable Functions	257
6.8.2. Special Constrained Problems	257
6.8.2.1. Convex programming	257
6.8.2.2. Separable programming	258
6.8.2.3. Geometric programming	258
Notes and Selected Bibliography for Section 6.8	259
CHAPTER 7. MODELLING	261
7.1. INTRODUCTION	261
7.2. CLASSIFICATION OF OPTIMIZATION PROBLEMS	262
7.3. AVOIDING UNNECESSARY DISCONTINUITIES	263
7.3.1. The Role of Accuracy in Model Functions	263
7.3.2. Approximation by Series or Table Look-Up	265
7.3.3. Subproblems Based on Iteration	266
Notes and Selected Bibliography for Section 7.3	267
7.4. PROBLEM TRANSFORMATIONS	267
7.4.1. Simplifying or Eliminating Constraints	267
7.4.1.1. Elimination of simple bounds	268
7.4.1.2. Elimination of inequality constraints	269
7.4.1.3. General difficulties with transformations	270
7.4.1.4. Trigonometric transformations	271
7.4.2. Problems Where the Variables are Continuous Functions	272
Notes and Selected Bibliography for Section 7.4	273
7.5. SCALING	273
7.5.1. Scaling by Transformation of Variables	273
7.5.2. Scaling Nonlinear Least-Squares Problems	275
7.6. FORMULATION OF CONSTRAINTS	276

Table of Contents

7.6.1.	Indeterminacy in Constraint Formulation	276
7.6.2.	The Use of Tolerance Constraints	277
	Notes and Selected Bibliography for Section 7.6	280
7.7.	PROBLEMS WITH DISCRETE OR INTEGER VARIABLES	281
7.7.1.	Pseudo-Discrete Variables	281
7.7.2.	Integer Variables	282
	Notes and Selected Bibliography for Section 7.7	283
CHAPTER 8.	PRACTICALITIES	285
8.1.	USE OF SOFTWARE	285
8.1.1.	Selecting a Method	285
8.1.1.1.	Selecting an unconstrained method	286
8.1.1.2.	Selecting a method for linear constraints	287
8.1.1.3.	Selecting a method for nonlinear constraints	290
8.1.2.	The User Interface	290
8.1.2.1.	Default parameters	291
8.1.2.2.	Service routines	291
8.1.3.	Provision of User-Defined Parameters	292
8.1.3.1.	The precision of the problem functions	292
8.1.3.2.	Choice of step-length algorithm	293
8.1.3.3.	Step-length accuracy	294
8.1.3.4.	Maximum step length	294
8.1.3.5.	A bound on the number of function evaluations	295
8.1.3.6.	Local search	295
8.1.3.7.	The penalty parameter in an augmented Lagrangian method	295
8.1.3.8.	The penalty parameter for a non-smooth problem	296
8.1.4.	Solving the Correct Problem	296
8.1.4.1.	Errors in evaluating the function	296
8.1.4.2.	Errors in computing derivatives	297
8.1.5.	Making the Best of the Available Software	298
8.1.5.1.	Nonlinear least-squares problems	298
8.1.5.2.	Missing derivatives	298
8.1.5.3.	Solving constrained problems with an unconstrained routine	299
8.1.5.4.	Treatment of linear and nonlinear constraints	299
8.1.5.5.	Nonlinear equations	299
	Notes and Selected Bibliography for Section 8.1	300
8.2.	PROPERTIES OF THE COMPUTED SOLUTION	300
8.2.1.	What is a Correct Answer?	300
8.2.2.	The Accuracy of the Solution	301
8.2.2.1.	Unconstrained problems	301
8.2.2.2.	Accuracy in constrained problems	303
8.2.3.	Termination Criteria	305
8.2.3.1.	The need for termination criteria	305
8.2.3.2.	Termination criteria for unconstrained optimization	306
8.2.3.3.	Termination criteria for linearly constrained optimization	308

Table of Contents

8.2.3.4.	Termination criteria for nonlinearly constrained optimization . . .	308
8.2.3.5.	Conditions for abnormal termination	309
8.2.3.6.	The selection of termination criteria	310
	Notes and Selected Bibliography for Section 8.2	312
8.3.	ASSESSMENT OF RESULTS	312
8.3.1.	Assessing the Validity of the Solution	312
8.3.1.1.	The unconstrained case	312
8.3.1.2.	The constrained case	315
8.3.2.	Some Other Ways to Verify Optimality	319
8.3.2.1.	Varying the parameters of the algorithm	319
8.3.2.2.	Using a different method	319
8.3.2.3.	Changing the problem	320
8.3.3.	Sensitivity Analysis	320
8.3.3.1.	The role of the Hessian	320
8.3.3.2.	Estimating the condition number of the Hessian	320
8.3.3.3.	Sensitivity of the constraints	323
	Notes and Selected Bibliography for Section 8.3	323
8.4.	WHAT CAN GO WRONG (AND WHAT TO DO ABOUT IT)	324
8.4.1.	Overflow in the User-Defined Problem Functions	324
8.4.2.	Insufficient Decrease in the Merit Function	324
8.4.2.1.	Errors in programming	325
8.4.2.2.	Poor scaling	325
8.4.2.3.	Overly-stringent termination criteria	327
8.4.2.4.	Inaccuracy in a finite-difference approximation	327
8.4.3.	Consistent Lack of Progress	328
8.4.3.1.	Unconstrained optimization	328
8.4.3.2.	Linearly constrained optimization	328
8.4.3.3.	Nonlinearly constrained optimization	328
8.4.4.	Maximum Number of Function Evaluations or Iterations	329
8.4.5.	Failure to Achieve the Expected Convergence Rate	329
8.4.6.	Failure to Obtain a Descent Direction	330
8.5.	ESTIMATING THE ACCURACY OF THE PROBLEM FUNCTIONS	331
8.5.1.	The Role of Accuracy	331
8.5.1.1.	A definition of accuracy	331
8.5.1.2.	How accuracy estimates affect optimization algorithms	331
8.5.1.3.	The expected accuracy	332
8.5.2.	Estimating the Accuracy	333
8.5.2.1.	Estimating the accuracy when higher precision is available	333
8.5.2.2.	Estimating the accuracy when derivatives are available	334
8.5.2.3.	Estimating the accuracy when only function values are available	335
8.5.2.4.	Numerical examples	336
8.5.3.	Re-Estimation of the Accuracy	338
	Notes and Selected Bibliography for Section 8.5	339

Table of Contents

8.6. COMPUTING FINITE DIFFERENCES	339
8.6.1. Errors in Finite-Difference Approximations; The Well-Scaled Case	339
8.6.1.1. The forward-difference formula	339
8.6.1.2. The central-difference formula	340
8.6.1.3. Second-order differences	341
8.6.2. A Procedure for Automatic Estimation of Finite-Difference Intervals	341
8.6.2.1. Motivation for the procedure	342
8.6.2.2. Statement of the algorithm	343
8.6.2.3. Numerical examples	344
8.6.2.4. Estimating the finite-difference interval at an arbitrary point	344
8.6.2.5. Finite-difference approximations in constrained problems	345
Notes and Selected Bibliography for Section 8.6	346
8.7. MORE ABOUT SCALING	346
8.7.1. Scaling the Variables	346
8.7.1.1. Scale-invariance of an algorithm	346
8.7.1.2. The conditioning of the Hessian matrix	347
8.7.1.3. Obtaining well-scaled derivatives	348
8.7.2. Scaling the Objective Function	351
8.7.3. Scaling the Constraints	352
8.7.3.1. Some effects of constraint scaling	352
8.7.3.2. Methods for scaling linear constraints	353
8.7.3.3. Methods for scaling nonlinear constraints	354
Notes and Selected Bibliography for Section 8.7	354
QUESTIONS AND ANSWERS	357
BIBLIOGRAPHY	363
INDEX	389

CHAPTER ONE

INTRODUCTION

Mankind always sets itself only such problems as it can solve. . .

—KARL MARX (1859)

1.1. DEFINITION OF OPTIMIZATION PROBLEMS

An optimization problem begins with a set of independent variables or parameters, and often includes conditions or restrictions that define acceptable values of the variables. Such restrictions are termed the *constraints* of the problem. The other essential component of an optimization problem is a single measure of “goodness”, termed the *objective function*, which depends in some way on the variables. The solution of an optimization problem is a set of allowed values of the variables for which the objective function assumes an “optimal” value. In mathematical terms, optimization usually involves *maximizing* or *minimizing*; for example, we may wish to maximize profit or minimize weight.

Problems in all areas of mathematics, applied science, engineering, economics, medicine, and statistics can be posed in terms of optimization. In particular, mathematical models are often developed in order to analyze and understand complex phenomena. Optimization is used in this context to determine the form and characteristics of the model that corresponds most closely to reality. Furthermore, most decision-making procedures involve explicit solution of an optimization problem to make the “best” choice. In addition to their role *per se*, optimization problems often arise as critical sub-problems within other numerical processes. This situation is so common that the existence of the optimization problem may pass unremarked — for example, when an optimization problem must be solved to find points where a function reaches a certain critical value.

Throughout this book, it will be assumed that the ultimate objective is to *compute* the solution of an optimization problem. In order to devise solution techniques, it is helpful to assume that optimization problems can be posed in a standard form. It is clearly desirable to select a standard form that arises naturally from the nature of most optimization problems, in order to reduce the need for re-formulation. The general form of optimization problem to be considered may be expressed in mathematical terms as:

NCP	minimize	$F(x)$
	$x \in \mathbb{R}^n$	
	subject to	$c_i(x) = 0, \quad i = 1, 2, \dots, m';$
		$c_i(x) \geq 0, \quad i = m' + 1, \dots, m.$

The objective function F and constraint functions $\{c_i\}$ (which, taken together, are termed the *problem functions*) are real-valued scalar functions.

To illustrate some of the flavour and diversity of optimization problems, we consider two specific examples. Firstly, the layout of the text of this book was designed by the \TeX computer

typesetting system using optimization techniques. The aim of the system is to produce a visually pleasing arrangement of text with justified margins. Two means are available to achieve this objective: the spaces between letters, words, lines and paragraphs can be adjusted, and words can be split between lines by hyphenation. An “ideal” spacing is specified for every situation in which a space may occur. These spaces may then be stretched or compressed within given limits of acceptability, subject to penalties for increasing the amount of deviation from the ideal. Varying penalties are also imposed to minimize undesirable features, such as two consecutive lines that end with hyphenated words or a page that begins with a displayed equation. The process of choosing a good text layout includes many of the elements of a general optimization problem: a single function that measures quality, parameters that can be adjusted in order to achieve the best objective, and restrictions on the form and extent of the allowed variation.

We next consider a simplified description of a real problem that illustrates the convenience of the standard form. The problem is to design the nose cone of a vehicle, such that, when it travels at hypersonic speeds, the air drag is minimized. Hence, the function to be optimized is the drag, and the parameters to be adjusted are the specifications of the structure of the nose cone. In order to be able to compute the objective function, it is necessary first to devise a model of the nose cone in terms of the chosen parameters. For this problem, the nose cone is represented as a series of conical sections, with a spherical section as the front piece and a fixed final radius R . Figure 1a illustrates the chosen model, and shows the parameters to be adjusted. Although the idealized model deviates from a real-world nose cone, the approximation should not noticeably impair the quality of the solution, provided that the number of conical sections is sufficiently large.

The next step is to formulate the drag as a scalar function of the eight parameters $\alpha_1, \dots, \alpha_4, r_1, \dots, r_4$. The function $D(\alpha_1, \dots, \alpha_4, r_1, \dots, r_4)$ will be assumed to compute an estimate of the drag on the nose cone for a set of particular values of the free variables, and thus D will be the objective function of the resulting optimization problem.

In order to complete the formulation of the problem, some restrictions must be imposed on the values of the design parameters in order for the mathematical model to be meaningful and for the optimal solution to be implementable. In particular, the radii r_1, \dots, r_4 must not be negative, so that the constraints

$$r_i \geq 0, \quad i = 1, \dots, 4,$$

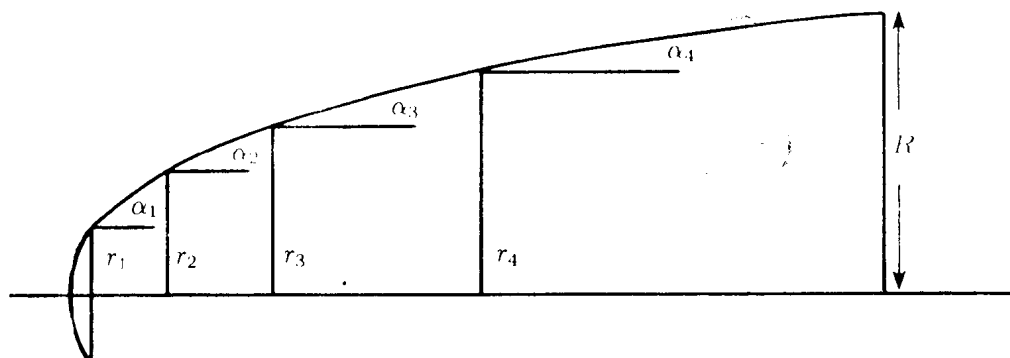


Figure 1a. Cross-section of a conical representation of a nose cone.