

COMPUTATIONAL METHODS IN OPTIMIZATION

Volume 77

E. Polak

COMPUTATIONAL METHODS IN OPTIMIZATION

A Unified Approach

This is Volume 77 in
MATHEMATICS IN SCIENCE AND ENGINEERING
A series of monographs and textbooks
Edited by RICHARD BELLMAN, *University of Southern California*

A complete list of the books in this series appears at the end of this volume.

**COMPUTATIONAL METHODS
IN OPTIMIZATION**

A Unified Approach

E. Polak

DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCES
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA

1971

ACADEMIC PRESS



New York and London

COPYRIGHT © 1971, BY ACADEMIC PRESS, INC.

ALL RIGHTS RESERVED

NO PART OF THIS BOOK MAY BE REPRODUCED IN ANY FORM, BY PHOTOSTAT, MICROFILM, RETRIEVAL SYSTEM, OR ANY OTHER MEANS, WITHOUT WRITTEN PERMISSION FROM THE PUBLISHERS. REPRODUCTION IN WHOLE OR IN PART FOR ANY PURPOSE OF THE UNITED STATES GOVERNMENT IS PERMITTED.

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
Berkeley Square House, London W1X 6BA

LIBRARY OF CONGRESS CATALOG CARD NUMBER: 72-134540
AMS(MOS) 1970 SUBJECT CLASSIFICATIONS: 90C30, 90C50, 49D05,
49D10, 49D15, 49D30, 49D35, 49D40, 49D45, 49D99, 34B05, 34B15,
34B99, 3900, 65H10, 65K05, 65L10, 65Q05

PRINTED IN THE UNITED STATES OF AMERICA

TO OREN AND SHARON

This page intentionally left blank

CONTENTS

Preface	ix
Note to the Reader	xiii
Conventions and Symbols	xv
1 PRELIMINARY RESULTS	1
1.1 Nonlinear Programming and Optimal Control Problems	1
1.2 Optimality Conditions	7
1.3 Models and Convergence Conditions for Computational Methods	12
2 UNCONSTRAINED MINIMIZATION	28
2.1 Gradient and Quasi-Newton Methods in \mathbb{R}^n	28
2.2 Reduction of Derivative Calculations	40
2.3 Conjugate Gradient Methods in \mathbb{R}^n	44
2.4 Unconstrained Discrete Optimal Control Problems	66
2.5 Unconstrained Continuous Optimal Control Problems	71
3 EQUALITY CONSTRAINTS: ROOT AND BOUNDARY-VALUE PROBLEMS	79
3.1 Zeros of a Function and Problems with Equality Constraints in \mathbb{R}^n	79
3.2 Boundary-Value Problems and Discrete Optimal Control	83
3.3 Boundary-Value Problems and Continuous Optimal Control	103
4 EQUALITY AND INEQUALITY CONSTRAINTS	126
4.1 Penalty Function Methods	126
4.2 Methods of Centers	150

4.3	Methods of Feasible Directions	159
4.4	Second-Order Methods of Feasible Directions	180
4.5	Gradient Projection Methods	185
5	CONVEX OPTIMAL CONTROL PROBLEMS	208
5.1	Nonlinear Programming Algorithms Revisited	208
5.2	A Decomposition Algorithm of the Dual Type	211
5.3	A Decomposition Algorithm of the Primal Type	234
6	RATE OF CONVERGENCE	242
6.1	Linear Convergence	242
6.2	Superlinear Convergence: Quasi-Newton Methods	251
6.3	Superlinear Convergence: Conjugate Gradient Methods	259
6.4	Superlinear Convergence: the Variable Metric Algorithm	268
Appendices		
A	FURTHER MODELS FOR COMPUTATIONAL METHODS	283
A.1	A Model for the Implementation of Certain Conceptual Optimal Control Algorithms	283
A.2	An Open-Loop Model for the Implementation of Conceptual Algorithms	288
B	PROPERTIES OF CONTINUOUS FUNCTIONS	292
B.1	Expansions of Continuous Functions	292
B.2	Convex Functions	294
B.3	A Few Miscellaneous Results	295
C	A GUIDE TO IMPLEMENTABLE ALGORITHMS	299
C.1	General Considerations	299
C.2	Gradient Methods	301
C.3	Quasi-Newton Methods	304
C.4	Conjugate Gradient Algorithms	306
C.5	Penalty Function Methods	309
C.6	Methods of Feasible Directions with Linear Search	312
C.7	Methods of Feasible Directions with Quadratic Search	315
References		
Index		323

PREFACE

Algorithms are inventions which very often appear to have little or nothing in common with one another. As a result, it was held for a long time that a coherent theory of algorithms could not be constructed. The last few years have shown that this belief was incorrect, that most convergent algorithms share certain basic properties, and hence that a unified approach to algorithms is possible.

This book presents a large number of optimization, boundary-value and root-solving algorithms in the light of a theory developed by the author, which deals with algorithm convergence and implementation. The theory of algorithms presented in this book rests on two pillars. One of these consists of a few, very simple, algorithm models (prototypes) with corresponding convergence theorems. The great importance of these algorithm models lies in the fact that they fit almost all existing algorithms for solving problems in optimal control and in nonlinear programming. Consequently, they provide us with a highly systematic approach to the study of convergence properties of algorithms. This systematic approach is valuable to us in three ways: It guides our inventive process toward algorithms whose convergence is assured, it simplifies considerably our work in showing that a specific algorithm is convergent, and it makes the teaching of algorithms considerably less time-consuming.

The second pillar of the theory of algorithms presented in this book consists of a methodology of implementation. Algorithms are usually invented in a form which is conceptually simple, but which is not necessarily implementable on a digital computer. Thus, an algorithm will usually construct a sequence of points z_0, z_1, z_2, \dots which converges to a solution point z . In a “conceptual,” or theoretical, algorithm, the construction of the point z_{i+1} from the point z_i may require the use of a subalgorithm which sets $y_0 = z_i$

and then constructs an infinite sequence y_0, y_1, y_2, \dots which converges to z_{i+1} (for example, as in the case of penalty function algorithms). Theoretically, therefore, in the case of such an algorithm, we can never compute z_{i+1} in finite time. In practice, we truncate the construction of the sequence y_0, y_1, \dots after a finite number of elements have been constructed. This truncation must be done with care: If we let the construction of the sequence y_0, y_1, y_2, \dots run for too long at each iteration, we may be using up much more computer time than is really needed; if we truncate too soon, we may lose convergence. The methodology of algorithm implementation presented throughout this book enables us to set up efficient schemes for truncating the construction of these sequences y_0, y_1, y_2, \dots , which are compatible with good convergence properties in the resulting implementation of the conceptual algorithm.

This book can be used either as a graduate level or reference text. It has been used in the Department of Electrical Engineering and Computer Sciences of the University of California, Berkeley, as a text for a first-year graduate level course in computational methods in optimization. The book deals with optimal control and nonlinear programming problems in a unified way, following the pattern set up in M. D. Canon, C. D. Cullum and E. Polak, "Theory of Optimal Control and Mathematical Programming," McGraw-Hill, New York, 1970, where the reader will find all the required background material on conditions of optimality, linear and quadratic programming, and convexity. Otherwise, this book is self-contained, with Appendix B furnishing the reader with the few additional mathematical results that he will need.

To facilitate the use of this book as a text, the author has slightly modified a number of standard algorithms so as to fit them into a unified framework. These modifications do not seem to affect adversely the performance of the algorithms in question. In selecting algorithms for presentation in this book, the author has given preference to algorithms which can be discussed easily within the theoretical framework of the book and to algorithms which can be used both for nonlinear programming and for optimal control problems. As a result, set approximation and cutting plane methods, the reduced gradient method, the convex simplex method, and dynamic programming were omitted.

For those who will be using this book as a reference text, Appendix C was added, to help in the choice of an efficient implementable algorithm. This appendix sets down the author's personal preferences in algorithms and offers the reader parameters for adjusting these algorithms to his own taste.

In presenting algorithms, either in their original or modified form, the author has attempted to acknowledge their originators. However, since some algorithms have been discovered more than once, and since there is little agreement as to the extent to which one algorithm should differ from another

before a person has a "right" to put one's name on it, the author wishes to apologize if, inadvertently, he has failed to give proper credit. Also, since the origins of some algorithms appear to be obscure, a number of them have been presented without acknowledgement, on the assumption that they are part of our technical folklore. The list of references at the end of the book includes only books and papers which the author consulted, directly or indirectly, in the preparation of the manuscript. They do not constitute an exhaustive bibliography.

The author is grateful to Drs. E. Gilbert, H. Halkin, P. Huard, D.H. Jacobson, and E. J. Messerli for comments, criticisms and suggestions, and to the graduate students in the author's course on algorithms, Messrs. G. Gross, F. Wu and O. Pironneau, in particular, for comments that have resulted in various improvements in the text. The new algorithms presented in this text have been tested to some extent, and the author is indebted to Messrs. K. Jeyarasasingam, J. Spoerl and J. Raphel for the many hours they have spent programming and computing with these algorithms.

The author is particularly indebted to his former students, Drs. R. Klessig and G. Meyer, for their collaboration on algorithms and algorithm models; to Dr. R. Klessig for his invaluable assistance in proofreading the manuscript, to Dr. A. Cohen and Mr. M. J. D. Powell for supplying the author with their results on the rate of convergence of algorithms prior to the publication of these results in the technical literature, and to Dr. W. I. Zangwill for the pleasant discussions in 1967 which have strongly stimulated the author's work in the area of algorithms.

The preparation of this volume involved a great amount of preliminary research which would have been impossible without the generous support received from the National Aeronautics and Space Administration under Grant NGL 05-003-016 and supplements 4, 5 and 6, from the Joint Services Electronics Program under Grant AFOSR 68-1488, and from the University of California. This support is gratefully acknowledged.

This page intentionally left blank

NOTE TO THE READER

The system of numbering and cross-referencing is described as follows. Within each section, definitions, theorems, equations, remarks, and so forth, are numbered consecutively by means of boldface numerals appearing in the left-hand margin. In reference to a section within the same chapter, the section number only is used; in reference to a section in another chapter, both the chapter number and the section number are used. For example, “it was shown in Section 3” refers to Section 3 of the same chapter, while “it was shown in Section 2.3” refers to Section 3 of Chapter 2. Similarly, “substituting from (3)” refers to item 3 in the same section, “substituting from (2.3)” refers to item 3 in Section 2 of the same chapter, and “substituting from (3.2.3)” refers to item 3 in Section 2 of Chapter 3.

This page intentionally left blank

CONVENTIONS AND SYMBOLS

1 Conventions

1. \mathbb{R}^n denotes the euclidean space of ordered n -tuples of real numbers. Elements of \mathbb{R}^n are denoted by lower-case letters, with the components of a vector x in \mathbb{R}^n shown as follows: $x = (x^1, x^2, \dots, x^n)$. When an n -tuple is a vector in \mathbb{R}^n , it is always treated as a *column* vector in matrix multiplications, i.e., as an $n \times 1$ matrix, but with the transposition symbol omitted. The scalar product in \mathbb{R}^n is denoted by $\langle \cdot, \cdot \rangle$ and is defined by $\langle x, y \rangle = \sum_{i=1}^n x_i y_i$. The norm in \mathbb{R}^n is denoted by $\| \cdot \|$ and is defined by $\| x \| = \sqrt{\langle x, x \rangle}$.

2. $C_v[t_0, t_f]$ denotes the space of all piecewise continuously differentiable functions from $[t_0, t_f]$ into \mathbb{R}^μ with norm defined by

$$\| x \|_\infty = \sup_{t \in [t_0, t_f]} \| x(t) \|$$

3. $L_2^\mu[t_0, t_f]$ denotes the Hilbert space consisting of equivalence classes of square integrable functions from $[t_0, t_f]$ into \mathbb{R}^μ , with norm denoted by $\| \cdot \|_2$ and defined by $\| u \|_2 = (\int_{t_0}^{t_f} \| u(t) \|^2 dt)^{1/2}$, and with scalar product denoted by $\langle \cdot, \cdot \rangle_2$ and defined by $\langle u_1, u_2 \rangle_2 = \int_{t_0}^{t_f} \langle u_1(t), u_2(t) \rangle dt$, where $\| \cdot \|$ and $\langle \cdot, \cdot \rangle$ denote the norm and scalar product in \mathbb{R}^μ , respectively.

4. $L_\infty^\mu[t_0, t_f]$ denotes the Banach space consisting of equivalence classes of essentially bounded, measurable functions from $[t_0, t_f]$ into \mathbb{R}^μ , with norm denoted by $\| \cdot \|_\infty$ and defined by $\| u \|_\infty = \text{ess sup}_{t \in [t_0, t_f]} \| u(t) \|$.

5. $f(\cdot)$ or f denotes a function, with the dot standing for the undesignated variable; $f(z)$ denotes the value of $f(\cdot)$ at the point z . To indicate that the domain of $f(\cdot)$ is A and that its range is B , we write $f: A \rightarrow B$. Assuming that $f: A \rightarrow \mathbb{R}^m$, we write f in expanded form as follows: $f = (f^1, f^2, \dots, f^m)$, so that $f(z) = (f^1(z), f^2(z), \dots, f^m(z))$.

6. Given a function $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, we denote its Jacobian matrix at z by $\partial g(z)/\partial z$. This is an $m \times n$ matrix whose ij th element is $\partial g^i(z)/\partial z^j$.
7. Given a function $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$, we denote by $\nabla f^i(z)$ its gradient at z . We always treat $\nabla f^i(z)$ as a column vector, and hence, its transpose is equal to $\partial f^i(z)/\partial z$, i.e., for any $y \in \mathbb{R}^n$, $\langle \nabla f^i(z), y \rangle = (\partial f^i(z)/\partial z) y$. We denote by $\partial^2 f^i(z)/\partial z^2$ the Hessian of $f^i(\cdot)$ at z . The Hessian is an $n \times n$ matrix whose jk th element is $\partial^2 f^i(z)/\partial z^j \partial z^k$.
8. Superscript -1 denotes the inverse of a matrix, e.g., A^{-1} .
9. Superscript T denotes the transpose of a matrix, e.g. A^T .
10. To avoid the need for writing $\{z' \in T \mid \|z - z'\|_{\mathcal{B}} \leq \epsilon\}$, we abuse standard mathematical notation for balls and denote this set by $B(z, \epsilon)$, where $\|\cdot\|_{\mathcal{B}}$ denotes the norm in the particular Banach space under discussion.

2 Symbols

$A \triangleq B$	A equals B by definition; denotes
$A \supset B$	A contains B
$A \subset B$	A is contained in B
$A \cup B$	union of A and B
$A \cap B$	intersection of A and B
$A \times B$	Cartesian product of A and B
$\{z \mid P\}$	set of points z having property P
$z \in A$	z belongs to A
$z \notin A$	z does not belong to A
\mathring{C}	interior of C
∂C	boundary of C
\bar{C}	closure of C (C subset of Banach space)
\bar{I}	complement of I (I subset of the integers)
\emptyset	the empty set
I	identity matrix; subset of integers
$B(z, \epsilon) =$	$\{z' \in T \mid \ z' - z\ _{\mathcal{B}} \leq \epsilon\}$
(a, b)	open interval $\{z = \lambda a + (1 - \lambda) b \mid 0 < \lambda < 1\}$
$[a, b]$	closed interval $\{z = \lambda a + (1 - \lambda) b \mid 0 \leq \lambda \leq 1\}$
$[A]$	convex hull of A
$\ \cdot\ $	euclidean norm (for vector or matrix)
$\langle \cdot, \cdot \rangle$	euclidean scalar product
$\cdot \times \cdot$	dyad: for $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, $x \times y$ is the $n \times m$ matrix xy^T
$\ \cdot\ _2$	norm in $L_2^\mu[t_0, t_f]$
$\langle \cdot, \cdot \rangle_2$	scalar product in $L_2^\mu[t_0, t_f]$
$\ \cdot\ _\infty$	norm in $L_\infty^\mu[t_0, t_f]$
$\ \cdot\ _{\mathcal{B}}$	norm in Banach space \mathcal{B}

$\max_{i \in J}$	maximum over $i \in J$ (the set J need not be shown)
$\min\{f^0(z) z \in C\}$	minimum of $f^0(z)$ over $z \in C$
$\bigcup_{v \in W}$	union over all the elements v in W
$x \leq y$	for x, y in \mathbb{R}^n , $x \leq y$ if $x^i \leq y^i$ for $i = 1, 2, \dots, n$
$\mathbb{R}^+ =$	$\{x \in \mathbb{R}^1 x \geq 0\}$
2^T	set of all subsets of T
$\text{sgn}(\cdot)$	signum function: $\text{sgn}(x) = 1$ for $x > 0$, $\text{sgn}(x) = -1$ for $x < 0$, $\text{sgn}(0) = 0$.
$\text{sat}(\cdot)$	saturation function: $\text{sat}(x) = x$ for $ x \leq 1$, $\text{sat}(x) = \text{sgn}(x)$ for $ x > 1$.
■	end of proof, end of remark, end of example, etc.

This page intentionally left blank

1

PRELIMINARY RESULTS

1.1 Nonlinear Programming and Optimal Control Problems

This text will be concerned with a unified presentation of algorithms for solving the following three optimization problems, which we state in the order of increasing complexity:

- 1 **The Nonlinear Programming Problem.** Given continuously differentiable functions $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $r : \mathbb{R}^n \rightarrow \mathbb{R}^l$, find a \hat{z} in the set $\Omega = \{z \mid f(z) \leqq 0, r(z) = 0\}$ such that for all $z \in \Omega$, $f^0(\hat{z}) \leq f^0(z)$. *

We shall usually use the following shorthand method for stating problem (1):

- 2
$$\min\{f^0(z) \mid f(z) \leqq 0, r(z) = 0\}.$$
- 3 **The Discrete Optimal Control Problem.** Given a dynamical system described by the difference equation

$$4 \quad x_{i+1} - x_i = f_i(x_i, u_i), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^u, \quad i = 0, 1, 2, \dots, k-1,$$

where x_i is the state of the system and u_i is the control applied to the system at time i , find a control sequence $\hat{\mathcal{U}} = (\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1})$ and a corresponding

* The functions f and r have components f^1, f^2, \dots, f^m and r^1, r^2, \dots, r^l , respectively. Thus, in expanded form, $f(z) = (f^1(z), f^2(z), \dots, f^m(z))$, which though written as a row vector will always be treated as a column vector. By $f(z) \leqq f(y)$, we mean $f^i(z) \leq f^i(y)$ for $i = 1, 2, \dots, m$. The notation $f(z) < f(y)$ is used to denote that $f^i(z) < f^i(y)$ for $i = 1, 2, \dots, m$, and that $f^i(z) < f^i(y)$ for at least one i .

trajectory $\hat{x} = (\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k)$, determined by (4), which minimize the cost functional

$$5 \quad \sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k),$$

subject to the constraints

$$6 \quad s_i(u_i) \leq 0, \quad i = 0, 1, \dots, k-1,$$

$$7 \quad g_i(x_i) = 0, \\ q_i(x_i) \leq 0, \quad i = 0, 1, \dots, k,$$

where $f_i : \mathbb{R}^v \times \mathbb{R}^\mu \rightarrow \mathbb{R}^v$, $f_i^0 : \mathbb{R}^v \times \mathbb{R}^\mu \rightarrow \mathbb{R}^1$, $\varphi : \mathbb{R}^v \rightarrow \mathbb{R}^1$, $s_i : \mathbb{R}^\mu \rightarrow \mathbb{R}^{\mu_i}$, $g_i : \mathbb{R}^v \rightarrow \mathbb{R}^{l_i}$, and $q_i : \mathbb{R}^v \rightarrow \mathbb{R}^{m_i}$ are continuously differentiable functions. The integer k is the duration of the control process. ■

- 8 **The Continuous Optimal Control Problem.** Given a dynamical system described by the differential equation

$$9 \quad \frac{d}{dt} x(t) = f(x(t), u(t), t), \quad x(t) \in \mathbb{R}^v, \quad u(t) \in \mathbb{R}^\mu, \quad t \in [t_0, t_f],$$

where $x(t)$ is the state of the system at time t , $u(t)$ is the control applied to the system at time t , t_0 is the given starting time, and t_f is the final time and may or may not be specified in advance, find a measurable control function $\hat{u}(\cdot)$ defined on $[t_0, t_f]$, a corresponding trajectory $\hat{x}(\cdot)$, determined by (9), and the final time t_f , if it is not specified, which minimize the cost functional

$$10 \quad \int_{t_0}^{t_f} f^0(x(t), u(t), t) dt + \varphi(x(t_f)),$$

subject to the constraints

$$11 \quad s(u(t)) \leq 0 \quad \text{for } t \in [t_0, t_f],$$

$$12 \quad g(x(t), t) = 0, \\ q(x(t), t) \leq 0 \quad \text{for } t \in [t_0, t_f],$$

where $f : \mathbb{R}^v \times \mathbb{R}^\mu \times \mathbb{R}^1 \rightarrow \mathbb{R}^v$ and $f^0 : \mathbb{R}^v \times \mathbb{R}^\mu \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$ are continuously differentiable in x and in u , $\varphi : \mathbb{R}^v \rightarrow \mathbb{R}^1$, $g : \mathbb{R}^v \times \mathbb{R}^1 \rightarrow \mathbb{R}^l$ and $q : \mathbb{R}^v \times \mathbb{R}^1 \rightarrow \mathbb{R}^m$ are continuously differentiable in x , and $s : \mathbb{R}^\mu \rightarrow \mathbb{R}^{\mu'}$ is continuously differentiable in u . In addition, $f, f^0, \partial f / \partial x, \partial f / \partial u, \partial f^0 / \partial x, \partial f^0 / \partial u, g, q, \partial g / \partial x, \partial q / \partial x$ are piecewise continuous in t . ■

The differentiability assumptions stated above are more stringent than is necessary for the purpose of deriving conditions of optimality, but are usually required in the application of the computational methods.

The nonlinear programming problem (1) is the simplest of the three problems that we have introduced. Not surprisingly, therefore, the largest fraction of existing algorithms deals with this problem. We shall show that a number of nonlinear programming algorithms are applicable also to optimal control problems. In doing so we shall make constant use of the following transcriptions of discrete optimal control problems into the form of a nonlinear programming problem (see Section 1.5 in [C1]).

- 13 First Transcription.** Consider the discrete optimal control problem (3). Let $z = (x_0, x_1, \dots, x_k, u_0, u_1, \dots, u_{k-1})$, let

$$14 \quad f^0(z) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k);$$

let

$$15 \quad r(z) = \begin{pmatrix} x_1 - x_0 - f_0(x_0, u_0) \\ \vdots \\ x_k - x_{k-1} - f_{k-1}(x_{k-1}, u_{k-1}) \\ g_0(x_0) \\ \vdots \\ g_k(x_k) \end{pmatrix};$$

and let

$$16 \quad f(z) = \begin{pmatrix} q_0(x_0) \\ \vdots \\ q_k(x_k) \\ s_0(u_0) \\ \vdots \\ s_{k-1}(u_{k-1}) \end{pmatrix}.$$

■

Then we see that the discrete optimal control problem (3) assumes the form of problem (2) and hence, it becomes clear that, at least in principle, all the nonlinear programming algorithms are applicable to the discrete optimal control problem.

The very high dimensionality of the vector z in (13) makes this transcription unsatisfactory in many instances. For this reason, we introduce an alternative transcription which utilizes a certain amount of precomputing in order to yield a problem of the form (2) in which the dimension of the vector z is not unreasonably high.

- 17 Second Transcription.** Consider the discrete optimal control problem (3). Let $z = (x_0, u_0, u_1, \dots, u_{k-1})$ and let $x_i(x_0, \mathcal{U})$ denote the solution of (4) at time i corresponding to the control sequence $\mathcal{U} = (u_0, u_1, \dots, u_{k-1})$.

We now set

$$18 \quad f^0(z) = \sum_{i=0}^{k-1} f_i^0(x_i(x_0, \mathcal{U}), u_i) + \varphi(x_k(x_0, \mathcal{U})),$$

$$19 \quad r(z) = \begin{pmatrix} g_0(x_0) \\ \vdots \\ g_k(x_k(x_0, \mathcal{U})) \end{pmatrix},$$

and

$$20 \quad f'(z) = \begin{pmatrix} q_0(x_0) \\ q_1(x_1(x_0, \mathcal{U})) \\ \vdots \\ q_k(x_k(x_0, \mathcal{U})) \\ s_0(u_0) \\ \vdots \\ s_{k-1}(u_{k-1}) \end{pmatrix}$$

With these definitions, the discrete optimal control problem again becomes reduced to the form of the nonlinear programming problem (2). However, now the vector z is of smaller dimension than in (13) and, in addition, the number of equality constraints, which are governed by the dimensionality of the vector $r(z)$, has also been substantially reduced. ■

We shall now discuss the origin of the discrete optimal control problem and why in many instances the formulation of an optimal control problem in discrete form may be preferable to a formulation in continuous form. The continuous optimal control problem requires us to find a measurable function $u(\cdot)$. However, digital computers do not compute such functions; they can only compute a sequence of numbers. Consequently, any numerical method of solving continuous optimal control problems must involve some form of discretization of the problem. Discrete optimal control problems are frequently discretizations of continuous optimal control problems, with the discretization being carried out in such a way as to minimize the mathematical difficulties in dealing with the problem, as well as to minimize the digital computer memory requirements and to reduce the number of operations required per iteration. As a general rule, one may frequently be forced to resort to a discrete optimal control problem formulation whenever one is interested in on-line control of a dynamical system by means of a small digital computer, since in such situations the solution of a continuous optimal control problem may simply not be practicable.

A rather common manner of producing a discrete optimal control problem from a continuous optimal control problem is that given below. Note that it permits us to exercise careful control on the digital computer memory requirements in solving this problem through the choice of the integer k .

It may also result in a reduced number of numerical operations per iteration. Thus, suppose that we restrict $u(\cdot)$ to the class of piecewise constant functions with at most $k - 1$ equidistant discontinuities. Let $T = (t_f - t_0)/k$ and let (assuming that t_f is given)

$$21 \quad u(t) = u_i \quad \text{for } t \in [t_0 + iT, t_0 + (i+1)T], \quad i = 0, 1, \dots, k-1.$$

Then the control constraints (11) become, after renaming functions,

$$22 \quad s_i(u_i) = s(u_i) \leq 0 \quad \text{for } i = 0, 1, \dots, k-1.$$

Now, let $x_i(t)$ for $i = 0, 1, \dots, k-1$ be the solution of (9) for $t \in [t_0 + iT, t_0 + (i+1)T]$, satisfying $x_i(t_0 + iT) = x_i$ and corresponding to $u(t) = u_i$ for $t \in [t_0 + iT, t_0 + (i+1)T]$, with $x_0(t_0) = x_0$ and $x_{i+1} = x_i(t_0 + (i+1)T)$ for $i = 0, 1, \dots, k-1$. Then we find that

$$23 \quad x_{i+1} = x_i + \int_{t_0+iT}^{t_0+(i+1)T} f(x_i(t), u_i, t) dt, \quad i = 0, 1, \dots, k-1.$$

Note that (23) is of the same form as (4), i.e., it defines a discrete dynamical system. To compute $f_i(x_i, u_i) = \int_{t_0+iT}^{t_0+(i+1)T} f(x_i(t), u_i, t) dt$ we must solve (9) with the initial condition $x(t_0 + iT) = x_i$ and $u(t) = u_i$ for $t \in [t_0 + iT, t_0 + (i+1)T]$. Thus, this form of discretization determines the number of u_i and x_i which will have to be stored at each iteration when the problem is solved on a digital computer, but it has nothing to say as to how (9) should be integrated. To complete the generation of a discrete optimal control problem, we set

$$24 \quad \begin{aligned} g_i(x_i) &= g(x_i, t_0 + iT), \\ q_i(x_i) &= q(x_i, t_0 + iT), \end{aligned} \quad i = 0, 1, \dots, k.$$

The need for integrating differential equations in solving optimal control problems results in a number of practical difficulties. It also introduces theoretical difficulties whenever one is honest enough to admit that these integrations can only be performed approximately. We shall give a few results which bear on this subject later on. However, at the present stage of development, one mostly has no choice but to isolate these difficulties, deal with them heuristically, and hope that the future will bring us better understanding of this problem.

25 **Exercise.** Consider the continuous optimal control problem:

$$26 \quad \text{minimize} \quad \int_0^{t_f} u(t)^2 dt,$$

subject to the constraints

$$27 \quad \frac{d}{dt} x(t) = Ax(t) + bu(t), \quad x(0) = x_0, \quad t \in [0, t_f],$$

$$28 \quad |u(t)| \leq 1 \quad \text{for } t \in [0, t_f],$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^1$ for $t \in [0, t_f]$, A is a constant matrix, and b is a constant vector.

(a) Show that when a discrete optimal control problem is obtained from this continuous optimal control problem in the manner discussed previously, i.e., as in (22)–(24), one can get explicit expressions for the functions $f_i(x_i, u_i)$ and $f_i^0(x_i, u_i)$.

(b) After obtaining a discrete optimal control problem from the continuous optimal control problem above, transcribe the discrete optimal control problem into nonlinear programming problem form using transcriptions (13) and (17). ■

The discretization of the continuous optimal control problem can be performed not only in the manner indicated in (21), but also in many other ways. For example, provided one can still interpret the given constraints on the control, one could reduce the infinite dimensional problem (8) to the finite dimensional form (2) by requiring that

$$29 \quad u(t) = \sum_{i=0}^{k-1} u_i \sin w_i t \quad \text{for } t \in [t_0, t_f],$$

where the w_i are given and the u_i are to be found. Or else one could require that

$$30 \quad u(t) = \sum_{j=0}^t u_{ij} t^j \quad \text{for } t \in [t_0 + iT, t_0 + (i+1)T],$$

in which case the control vector $u_i = (u_{i0}, u_{i1}, \dots, u_{it})$ of the discrete optimal control problem has higher dimension than that of $u(t)$, the control vector for the continuous optimal control problem.

31 Exercise. When the discretizations (29) or (30) are used, it may be quite difficult to transcribe the constraint (11) into a suitable set of constraints for the u_i in (29) or the u_{ij} in (30). Suppose that $u(t) \in \mathbb{R}^1$ and that the constraint $s(u(t)) \leq 0$ is of the form $|u(t)| \leq 1$ (i.e., $s^1(u) = u - 1$, $s^2(u) = -u - 1$). Construct constraints on the u_i in (29) and on the u_{ij} in (30) which are sufficient to ensure that $u(t)$ satisfies (11). ■

1.2 Optimality Conditions

So far, there are no satisfactory tests for establishing whether a candidate solution to any one of the three general optimization problems presented in the preceding section is optimal or not. It is therefore not surprising that there are no general methods for solving such problems. The algorithms which we shall describe in the chapters to follow can only be used to construct a sequence whose limit satisfies a particular optimality condition. Generally, in the absence of convexity assumptions, such a condition is only a necessary condition of optimality.

The subject of optimality conditions is rather vast and the reader is therefore referred to [C1] for a presentation in depth. Here we shall content ourselves with stating, mostly without proof, the optimality conditions used most frequently in computational methods. We shall also identify a few cases where these optimality conditions are satisfied trivially, since any algorithm depending on these conditions would become useless for such a problem.

- 1 Theorem.** If \hat{z} is optimal for the nonlinear programming problem (1.1), i.e., $f^0(\hat{z}) = \min\{f^0(z) | f(z) \leq 0, r(z) = 0\}$, then there exist scalar multipliers $\mu^0 \leq 0, \mu^1 \leq 0, \dots, \mu^m \leq 0$, and $\psi^1, \psi^2, \dots, \psi^l$, not all zero, such that

$$2 \quad \sum_{i=0}^m \mu^i \nabla f^i(\hat{z}) + \sum_{i=1}^l \psi^i \nabla r^i(\hat{z}) = 0,$$

and

$$3 \quad \mu^i f^i(\hat{z}) = 0 \quad \text{for } i = 1, 2, \dots, m. \blacksquare$$

An early version of this theorem is due to John [J1]. In the form stated, this theorem is quite difficult to prove and the interested reader should consult either [C1] or [M4] if he wishes to see how this is done.

The following two corollaries are important special cases of the Kuhn-Tucker conditions [K4]:

- 4 Corollary.** If there exists a vector $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(\hat{z}), h \rangle > 0$ for all $i \in \{1, 2, \dots, m\}$ satisfying $f^i(\hat{z}) = 0$, then the multipliers satisfying (2) and (3) must also satisfy $(\mu^0, \psi^1, \psi^2, \dots, \psi^l) \neq 0$. \blacksquare

- 5 Exercise.** Prove corollary (4). \blacksquare

- 6 Corollary.** Suppose that the vectors $\nabla r^i(\hat{z}), i = 1, 2, \dots, l$ in (2) are linearly independent. If there exists a vector $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle > 0$ for all $i \in \{1, 2, \dots, m\}$ satisfying $f^i(\hat{z}) = 0$, and $\langle \nabla r^i(z), h \rangle = 0$ for $i = 1, 2, \dots, l$, then the multipliers satisfying (2) and (3) must also satisfy $\mu^0 < 0$. \blacksquare

7 Exercise. Prove corollary (6). ■

We now present a condition of optimality for a special case of the nonlinear programming problem (1.1), viz., the case when $r(\cdot) = 0$. This condition was probably first published by Zoutendijk [Z4] and provides a starting point for constructing a number of important algorithms (the method of feasible directions, the method of centers, etc.). It differs from theorem (1) in one very important respect: it does not involve any multipliers.

8 Theorem. Suppose that $r(\cdot) = 0$ in the nonlinear programming problem (1.1) and that \hat{z} is an optimal solution to (1.1), i.e., $f^0(\hat{z}) = \min\{f^0(z) | f(z) \leq 0\}$, then

$$9 \quad \min_{h \in S} \max_{i \in J_0(\hat{z})} \langle \nabla f^i(\hat{z}), h \rangle = 0,$$

where S is any subset of \mathbb{R}^n containing the origin in its interior, and

$$10 \quad J_0(\hat{z}) = \{0\} \cup \{i | f^i(\hat{z}) = 0, i \in \{1, 2, \dots, m\}\}.$$

Proof. Suppose that (9) is false. Then there must exist a vector $h' \in S$ such that

$$\langle \nabla f^i(\hat{z}), h' \rangle \leq \delta < 0 \quad \text{for all } i \in J_0(\hat{z}).$$

Now, setting $\nabla r^i(\hat{z}) = 0$ in (2), since $r(\cdot) = 0$ by assumption, and taking the scalar product of both sides of (2) with h' , we get a contradiction of theorem (1). ■

11 Exercise. Give a proof of theorem (8) which does not require the use of theorem (1). ■

12 Exercise. For the special case $r(\cdot) = 0$, theorems (1) and (8) are equivalent. Above, we have deduced theorem (8) from theorem (1). To complete the demonstration of equivalence, deduce theorem (1) from theorem (8) under the assumption that $r(\cdot) = 0$. ■

13 Proposition. Suppose that the set $\Omega = \{z | f^i(z) \leq 0, i = 1, 2, \dots, m\}$ has no interior, and that $r(\cdot) = 0$. Then (2) and (3) can be satisfied at every point $z \in \Omega$ (i.e., theorem (1) becomes trivial when the set Ω has no interior).

Proof. Let z^* be any point in Ω . Since Ω has no interior, it is clear that there is no vector $h \in \mathbb{R}^n$ such that

$$\langle \nabla f^i(z^*), h \rangle < 0 \quad \text{for all } i \in I(z^*),$$

where $I(z^*) \triangleq \{j \in \{1, 2, \dots, m\} | f^j(z^*) = 0\}$. Suppose that $I(z^*) = \{i_1, i_2, \dots, i_\alpha\}$. Then, because of the above observation, the linear subspace

$$L = \{v = (\langle \nabla q^{i_1}(z^*), h \rangle, \dots, \langle \nabla q^{i_\alpha}(z^*), h \rangle) | h \in \mathbb{R}^n\} \subset \mathbb{R}^\alpha$$

has no ray in common with the cone $C = \{v \mid v < 0\} \subset \mathbb{R}^n$, and hence L and \bar{C} can be separated, i.e., there exists a nonzero $\xi \in \mathbb{R}^n$ such that

$$14 \quad \langle \xi, v \rangle = 0 \quad \text{for all } v \in L$$

$$15 \quad \langle \xi, v \rangle \geq 0 \quad \text{for all } v \in \bar{C}.$$

It now follows from (15) that $\xi \leq 0$ and from (14) that

$$\sum_{j=1}^n \xi^j \langle \nabla q^{ij}(z^*), h \rangle = 0 \quad \text{for all } h \in \mathbb{R}^n.$$

But this implies that

$$16 \quad \sum_{j=1}^n \xi^j \nabla q^{ij}(z^*) = 0.$$

Setting $\mu^i = 0$ for all $i \in I(z^*)$ and $\mu^i j = \xi^j$ for all $i, j \in I(z^*)$, we see that (2) and (3) are satisfied. ■

- 17 **Exercise.** Show that under the assumptions of proposition (13), condition (9) can be satisfied at any point $z' \in \{z \mid f(z) \leq 0\}$. ■

So far, we have only presented necessary conditions of optimality for problem (1.1). We shall now give a sufficient condition.

- 18 **Theorem.** Consider the problem (1.1). Suppose that the functions $f^i(\cdot)$, $i = 0, 1, \dots, m$, are convex and that the function $r(\cdot)$ is affine. If \hat{z} satisfies $r(\hat{z}) = 0$, $f^i(\hat{z}) \leq 0$, for $i = 1, 2, \dots, m$, and there exist scalar multipliers $\mu^i \leq 0$ for $i = 1, 2, \dots, m$ and ψ^i for $i = 1, 2, \dots, l$ such that

$$19 \quad -\nabla f^0(\hat{z}) + \sum_{i=1}^m \mu^i \nabla f^i(\hat{z}) + \sum_{i=1}^l \psi^i \nabla r^i(\hat{z}) = 0,$$

$$20 \quad \mu^i f^i(\hat{z}) = 0 \quad \text{for } i = 1, 2, \dots, m,$$

then \hat{z} is optimal for (1.1). (This theorem was first presented in [K4].)

Proof. Let $\Omega' = \{z \mid f(z) \leq 0, r(z) = 0\}$. Then, since the $r^i(\cdot)$ are affine, for any $z \in \Omega'$, we have $\langle \nabla r^i(z), z - \hat{z} \rangle = 0$, and hence, by (19), for any $z \in \Omega'$,

$$21 \quad \langle \nabla f^0(\hat{z}), z - \hat{z} \rangle = \sum_{i=1}^m \mu^i \langle \nabla f^i(z), z - \hat{z} \rangle.$$

Now, since the functions $f^i(\cdot)$ are convex, we have, for any $z \in \Omega'$ and any $i \in \{i \mid f^i(\hat{z}) = 0, i \in \{1, 2, \dots, m\}\}$,

$$22 \quad \langle \nabla f^i(\hat{z}), z - \hat{z} \rangle \leq f^i(z) \leq 0.$$

Now making use of (19), (21) and (22), we obtain, since $f^0(\cdot)$ is convex, for any $z \in \Omega'$,

$$23 \quad f^0(\hat{z}) \leq f^0(z) - \langle \nabla f^0(\hat{z}), z - \hat{z} \rangle = f^0(z) - \sum_{i=1}^m \mu^i \langle \nabla f^i(\hat{z}), z - \hat{z} \rangle \leq f^0(z).$$

Thus, \hat{z} is optimal. ■

Since the discrete optimal control problem is transcribable into the form of the nonlinear programming problem, it is clear that one can obtain optimality conditions for the discrete optimal control problem from the ones above. For a study in depth, the reader should consult [C1]; here we shall content ourselves by illustrating the procedure to be followed by presenting a special case.

24 **Theorem.** Consider problem (1.3) and suppose that $g_1(\cdot) = g_2(\cdot) = \dots = g_{k-1}(\cdot) = q_1(\cdot) = \dots = q_{k-1}(\cdot) = 0$. If the control sequence $\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1}$ and the corresponding trajectory $\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k$ are optimal for the problem (1.3), then there exist a scalar multiplier $\hat{p}^0 \leq 0$ and vector multipliers $\hat{p}_0, \hat{p}_1, \dots, \hat{p}_k, \hat{\pi}_0, \hat{\pi}_k, \hat{\mu}_0 \leq 0, \dots, \hat{\mu}_{k-1} \leq 0, \hat{\xi}_0 \leq 0, \hat{\xi}_k \leq 0$, not all zero, such that

$$25 \quad \hat{p}_i - \hat{p}_{i+1} = \left(\frac{\partial f_i(\hat{x}_i, \hat{u}_i)}{\partial x_i} \right)^T \hat{p}_{i+1} + \hat{p}^0 \left(\frac{\partial f_i^0(\hat{x}_i, \hat{u}_i)}{\partial x_i} \right)^T, \quad i = 0, 1, \dots, k-1;$$

$$26 \quad -\hat{p}_0 = \left(\frac{\partial g_0(\hat{x}_0)}{\partial x_0} \right)^T \hat{\pi}_0 + \left(\frac{\partial q_0(\hat{x}_0)}{\partial x_0} \right)^T \hat{\xi}_0;$$

$$27 \quad \hat{p}_k = \left(\frac{\partial g_k(\hat{x}_k)}{\partial x_k} \right)^T \hat{\pi}_k + \left(\frac{\partial q_k(\hat{x}_k)}{\partial x_k} \right)^T \hat{\xi}_k + \left(\frac{\partial \varphi(x_k)}{\partial x_k} \right)^T \hat{p}^0;$$

$$28 \quad \hat{p}^0 \left(\frac{\partial f_i^0(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T + \left(\frac{\partial f_i(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T \hat{p}_{i+1} + \left(\frac{\partial s_i(\hat{u}_i)}{\partial u_i} \right)^T \hat{\mu}_i = 0, \\ i = 0, 1, 2, \dots, k-1;$$

$$29 \quad \langle \hat{\xi}_0, q_0(\hat{x}_0) \rangle = \langle \hat{\xi}_k, q_k(\hat{x}_k) \rangle = 0;$$

$$30 \quad \langle s_i(\hat{u}_i), \hat{\mu}_i \rangle = 0 \quad \text{for } i = 0, 1, 2, \dots, k-1$$

(and the \hat{x}_i, \hat{u}_i satisfy (1.4), (1.6) and (1.7)). ■

To obtain theorem (24) from theorem (1), we proceed as follows: First we note that (2) is equivalent to the statement $\partial L(\hat{z})/\partial z = 0$, where $L(z) = \mu^0 f^0(z) + \langle \mu, f(z) \rangle + \langle \psi, r(z) \rangle$, where $\mu = (\mu^1, \mu^2, \dots, \mu^m)$ and $\psi = (\psi^1, \psi^2, \dots, \psi^l)$. Then we transcribe problem (1.3) into the form of problem (1.1) by means of transcription (1.13) and set $\mu^0 = \hat{p}^0$, $\mu = (\hat{\xi}_0, \hat{\xi}_k, \hat{\mu}_0, \hat{\mu}_1, \dots, \hat{\mu}_{k-1})$ and $\psi = (-\hat{p}_1, -\hat{p}_2, \dots, -\hat{p}_k, \hat{\pi}_0, \hat{\pi}_k)$.

Expanding $L(z)$, we now obtain, with $z = (x_0, x_1, \dots, x_k, u_0, u_1, \dots, u_{k-1})$,

$$\begin{aligned} 31 \quad L(z) &= \hat{p}^0 \left[\sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k) \right] - \sum_{i=0}^{k-1} \langle \hat{p}_{i+1}, x_{i+1} - x_i - f_i(x_i, u_i) \rangle \\ &\quad + \langle \hat{\pi}_0, g_0(x_0) \rangle + \langle \hat{\pi}_k, g_k(x_k) \rangle + \langle \hat{\xi}_0, q_0(x_0) \rangle + \langle \hat{\xi}_k, q_k(x_k) \rangle \\ &\quad + \sum_{i=0}^{k-1} \langle \hat{\mu}_i, s_i(u_i) \rangle. \end{aligned}$$

Now, computing $\partial L(\hat{z})/\partial x_i$ for $i = 0, 1, \dots, k$ and $\partial L(\hat{z})/\partial u_i$ for $i = 0, 1, \dots, k-1$, and setting these equal to zero, we obtain (26)–(28).

We observe that theorem (24) differs from theorem (1) in that the considerable structure of the dynamics of the discrete optimal control problem enables us to introduce a corresponding amount of structure into the conditions of optimality. It should be noted, however, that when all the constraints stated in the general form (1.3) are present, the structure of the dynamics no longer leads to special theorems which are more useful than theorem (1). (For the case stated, theorem (24) is considerably more useful than theorem (1), as we shall see later.)

- 32 **Remark.** One may sometimes wish to eliminate the $\hat{\mu}_i$ in (28) and (30). This can be done by removing (30) completely and by substituting the following condition for (28):

$$33 \quad \left\langle \hat{p}^0 \left(\frac{\partial f_i^0(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T + \left(\frac{\partial f_i(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T \hat{p}_{i+1}, \delta u \right\rangle \leq 0$$

for all δu such that

$$34 \quad \left(\frac{\partial s_i(\hat{u}_i)}{\partial u_i} \right) \delta u \leq 0. \quad \blacksquare$$

For the continuous optimal control problem, we shall only need the Pontryagin maximum principle, which gives a necessary condition of optimality for a special case of the continuous optimal control problem (1.8). This is a very difficult result to derive and the reader may wish to look up a proof in [C1].

- 35 **Theorem** (the Pontryagin maximum principle [P7]). Consider the problem (1.8) and suppose that t_0, t_1 are given; that $g(\cdot, t) = 0$ for all $t \neq t_1$ in $[t_0, t_1]$ (we shall write $g(\cdot, t_1)$ as $g(\cdot)$); that $q(\cdot, t) = 0$ for all $t \neq t_0$ in $[t_0, t_1]$; and that ξ_0 is the only solution to $q(x, t_0) \leq 0$, i.e., we are given the initial state $x(t_0) = \xi_0$. If $\hat{u}(\cdot)$ is an optimal control for the problem (1.8) and $\hat{x}(\cdot)$ is the corresponding optimal trajectory, then there exists a scalar $\hat{p}^0 \leq 0$

and a multiplier function $\hat{p}(\cdot)$ mapping $[t_0, t_f]$ into \mathbb{R}^v , with $(\hat{p}^0, \hat{p}(\cdot)) \neq 0$, such that

$$36 \quad \frac{d}{dt} \hat{p}(t) = - \left(\frac{\partial f(\hat{x}(t), \hat{u}(t), t)}{\partial x} \right)^T \hat{p}(t) - \left(\frac{\partial f^0(\hat{x}(t), \hat{u}(t), t)}{\partial x} \right)^T \hat{p}^0 \quad \text{for } t \in [t_0, t_f]$$

$$37 \quad \hat{p}(t_f) = \left(\frac{\partial g(\hat{x}(t_f))}{\partial x} \right)^T \psi + \left(\frac{\partial \varphi(\hat{x}(t_f))}{\partial x} \right)^T \hat{p}^0 \quad \text{for some } \psi \in \mathbb{R}^v,$$

and for every vector $v \in \mathbb{R}^u$ satisfying $s(v) \leqq 0$ and almost all $t \in [t_0, t_f]$,

$$38 \quad \begin{aligned} & \hat{p}^0 f^0(\hat{x}(t), \hat{u}(t), t) + \langle \hat{p}(t), f(\hat{x}(t), \hat{u}(t), t) \rangle \\ & \geq \hat{p}^0 f^0(\hat{x}(t), v, t) + \langle \hat{p}(t), f(\hat{x}(t), v, t) \rangle. \end{aligned} \quad \blacksquare$$

- 39 **Exercise.** Suppose that t_f is not fixed in problem (1.8). Show that by introducing an auxiliary variable, the free-time problem can be transcribed into a fixed-time problem. Obtain the maximum principle for the free-time problem from theorem (35). \blacksquare

1.3 Models and Convergence Conditions for Computational Methods

Our study of computational methods will be made considerably easier by the simple models that we shall describe in this section. We shall use these models in three ways. First, they will enable us to introduce a certain amount of classification into computational methods. Second, they will make it possible for us to carry out crucial reasoning in establishing the convergence properties of an algorithm, without becoming submerged in the clutter that is otherwise introduced by the great complexity of the subprocedures which modern algorithms utilize. Our third, and most important use of models, will be in developing procedures for obtaining “implementable” algorithms from “conceptual” prototypes.

Throughout this book we shall make a distinction between *conceptual* and *implementable* algorithms. This distinction will be based on the following criterion: Each iteration of a *conceptual algorithm* may be made up of an *arbitrary* number of arithmetical operations and function evaluations, while each iteration of an *implementable algorithm* must be made up of a *finite* number of arithmetical operations and function evaluations. In this we assume that an acceptable approximation to the value of a function can be computed in finite time on a digital computer. It is immediately clear that the distinction we have just introduced is a subjective one, for the reader may decide to call the finding of a root of a transcendental equation a function evaluation, while the author insists that it should be regarded

as an infinite subprocedure and hence inadmissible as an operation to be performed in the course of each iteration of an implementable algorithm. As a rule of thumb, the author considers the computation of an acceptable approximation to the values of such functions as e^x and $\sin x$, as performable in finite time. However, he considers the computation of an acceptable approximation to the smallest positive root of the equation $e^{-x} + \sin x = 0$ as requiring infinite time, because any practical procedure for computing this root, such as Newton's method, will require a large number of values of e^{-x} and of $\sin x$.

In the process of constructing a computational method, one usually begins by inventing a conceptual algorithm. Then one modifies this conceptual algorithm in such a way as to reduce each of its iterations to a finite number of digital computer operations, i.e., one reduces it to a form that can be programmed for execution on a digital computer. In the past, the accepted practice was to implement a conceptual algorithm by doing one's "best" in approximating such infinite operations as the finding of a minimum of a function along a line, or the resolving of an implicit relationship. Doing one's best usually meant that the accuracy of these approximations in the subprocedures was the same whether one was near or far from a point that the algorithm was trying to find. The main drawback of such an approach is that it results in a great deal of time being wasted on very precise calculations when one is still quite far from the point one is trying to find. To make matters worse, the resulting algorithm may fail to converge. An alternative and very much more efficient approach to implementation is to use an adaptive, or closed-loop, method for truncating at least some of the infinite subprocedures which appear in conceptual algorithms. The models that we shall describe in this section will enable us to establish the principles of our approach to adaptive truncation of infinite subprocedures. In Appendix A we shall present an open-loop approach to truncation. While the idea of conceptual and implementable algorithms is quite subjective, the reader will find this concept extremely useful in deciding whether it does or does not make sense to add the special truncation procedures that we shall describe to a specific algorithm that he wishes to program. We shall clarify our thinking on this matter further in the chapters to follow, where we shall examine a large number of algorithms.

Throughout this text we shall consider algorithms for solving problems of the following form:

- 1 **Abstract Problem.** Given a closed subset T of a Banach space \mathcal{B} , construct points in T which have property P . ■

In the case of nonlinear programming problems and discrete optimal control problems, the space \mathcal{B} will be \mathbb{R}^n , while in the case of continuous

optimal control problems, the space \mathcal{B} will be either L_2 or L_∞ . Points with the property P will either be optimal for one of these optimization problems, or else they will satisfy some optimality condition. In order to avoid this complicated listing of possibilities, we shall call points in T with the property P , *desirable*. We shall always assume that T contains desirable points, and, in keeping with our intuitive idea of what is implementable and what is not, we shall allow a property P to be considered only if we have a reasonable test for determining whether a point z in T has the property P or not. For example, suppose that we wish to minimize $f^0(z)$ over \mathbb{R}^n . For this problem, $T = \mathbb{R}^n$ and, *conceptually*, a point in \mathbb{R}^n is desirable if it minimizes $f^0(z)$. However, unless the function $f^0(\cdot)$ is convex, we have no tests for determining whether a point z' minimizes $f^0(z)$ or not, and hence we cannot really hope to compute such a z' . Consequently, we shall not permit ourselves to consider such a point to be desirable. Instead, we might define a point z' to be desirable if $\nabla f^0(z') = 0$, i.e., if it passes a simple test.

The simplest algorithms for computing desirable points in a closed subset T of \mathcal{B} utilize a *search function* $a : T \rightarrow T$ and a *stop rule* $c : T \rightarrow \mathbb{R}^1$, and are of the following form:

2 Algorithm Model. $a : T \rightarrow T$, $c : T \rightarrow \mathbb{R}^1$.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Compute $a(z_i)$.

Step 3. Set $z_{i+1} = a(z_i)$.

Step 4. If $c(z_{i+1}) \geq c(z_i)$, stop;* else, set $i = i + 1$ and go to step 2. ■

We shall now show what one can hope to compute with such an algorithm.

3 Theorem. Suppose that

(i) $c(\cdot)$ is either continuous at all nondesirable points $z \in T$, or else $c(z)$ is bounded from below for $z \in T$;

(ii) for every $z \in T$ which is not desirable, there exist an $\epsilon(z) > 0$ and a $\delta(z) < 0$ such that

$$4 \quad c(a(z')) - c(z') \leq \delta(z) < 0, \quad \text{for all } z' \in B(z, \epsilon(z)),$$

where $B(z, \epsilon(z)) = \{z \in T \mid \|z' - z\|_{\mathcal{B}} \leq \epsilon(z)\}$.

Then, either the sequence $\{z_i\}$ constructed by algorithm (2) is finite and its next to last element is desirable, or else it is infinite and every accumulation point of $\{z_i\}$ is desirable.

* A direct test for determining if z_i is desirable may be substituted for the test $c(z_{i+1}) \geq c(z_i)$.

Proof. We begin with the observation that (by negation) (ii) implies that if

$$5 \quad c(a(z)) \geq c(z),$$

then z is desirable.

Now, suppose the sequence $\{z_i\}$ is finite, i.e., $\{z_i\} = \{z_0, z_1, \dots, z_k, z_{k+1}\}$. Then by step 4, $c(z_{k+1}) \geq c(z_k)$, and hence from (5), z_k is desirable. Now suppose that the sequence $\{z_i\}$ is infinite, and that it has a subsequence which converges to the point z' . We express this as $z_i \rightarrow z'$ for $i \in K \subset \{0, 1, 2, \dots\}$. Assuming that z' is not desirable, there exist an $\epsilon' > 0$, a $\delta' < 0$, and a $k \in K$ such that for all $i \geq k$, $i \in K$,

$$6 \quad \|z_i - z'\|_{\mathcal{B}} \leq \epsilon'$$

and

$$7 \quad c(z_{i+1}) - c(z_i) \leq \delta'.$$

Hence, for any two consecutive points z_i, z_{i+j} of the subsequence, with $i \geq k$ (and $i, (i+j) \in K$), we must have

$$8 \quad \begin{aligned} c(z_{i+j}) - c(z_i) &= [c(z_{i+j}) - c(z_{i+j-1})] + [c(z_{i+j-1}) - c(z_{i+j-2})] + \cdots \\ &\quad + [c(z_{i+1}) - c(z_i)] < c(z_{i+1}) - c(z_i) \leq \delta'. \end{aligned}$$

Now, for $i \in K$, the monotonically decreasing sequence $c(z_i)$ must converge either because $c(\cdot)$ is continuous at z' or else because $c(z)$ is bounded from below on T . But this is contradicted by (8), which shows that the sequence $c(z_i)$ is not a Cauchy sequence for $i \in K$, and hence the theorem must be true. ■

A somewhat more sophisticated and useful model is obtained by substituting for the search function $a : T \rightarrow T$ of algorithm (2) a *set-valued* search function A mapping T into the set of all nonempty subsets of T (which we write as $A : T \rightarrow 2^T$). We then get the following procedure (which also uses a stop rule $c : T \rightarrow \mathbb{R}^1$):

9 **Algorithm Model.** $A : T \rightarrow 2^T$, $c : T \rightarrow \mathbb{R}^1$.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Compute a point $y \in A(z_i)$.

Step 3. Set $z_{i+1} = y$.

Step 4. If $c(z_{i+1}) \geq c(z_i)$, stop;* else, set $i = i + 1$ and go to step 2. ■

10 **Theorem.** Consider algorithm (9). Suppose that

(i) $c(\cdot)$ is either continuous at all nondesirable points $z \in T$, or else $c(z)$ is bounded from below for $z \in T$;

* See footnote to algorithm (2).

(ii) for every $z \in T$ which is not desirable, there exist an $\epsilon(z) > 0$ and a $\delta(z) < 0$ such that

11 $c(z'') - c(z') \leq \delta(z) < 0,$

for all $z' \in T$ such that $\|z' - z\|_{\mathcal{B}} \leq \epsilon(z)$, and for all $z'' \in A(z')$.

Then, either the sequence $\{z_i\}$ constructed by algorithm (9) is finite and its next to last element is desirable, or else it is infinite and every accumulation point of $\{z_i\}$ is desirable. ■

12 **Exercise.** Show that (ii) of (10) implies that if $c(z') \geq c(z)$ for at least one $z' \in A(z)$, then z is desirable. ■

13 **Exercise.** Prove theorem (10). ■

14 **Remark.** The reader should be careful not to read more into the statements of the above convergence theorems than they actually say. Note that these theorems state only that *if* a convergent subsequence exists, then its limit point will be desirable. To ensure that accumulation points exist, it is necessary to make some additional assumptions. For example, one may assume that the set T is compact, or that the set $\{z \in T \mid c(z) \leq c(z_0)\}$ is compact, where z_0 is the starting point for the algorithm. The reason for not including such assumptions in the statement of theorems such as (3) and (10) is that it is usually better to determine whether an algorithm will produce compact sequences by examining the algorithm in the light of the specific problem to which one wishes to apply it. This point will become clear in the chapters to follow. ■

The assumptions (i) and (ii) of theorem (10) are not the only ones under which the conclusions of theorem (10) are valid. The following set of assumptions are due to Zangwill [Z1] and can be shown, though not very easily, to be stronger than the assumptions of theorem (10), i.e., whenever the assumptions below are satisfied, the assumptions of theorem (10) are also satisfied.

15 **Exercise.** Consider algorithm (9). Suppose that the set T in (1) is compact, that the stop rule $c(\cdot)$ is continuous, and that the map $A(\cdot)$ has the following property: If $\{z_i\}$ is any sequence in T converging to a point z' and $\{y_i\}$ is any sequence in T converging to a point y' and satisfying $y_i \in A(z_i)$ for $i = 0, 1, 2, \dots$, then $y' \in A(z')$. Show that the conclusions of theorem (10) remain valid under this assumption, provided $c(z') \geq c(z)$ for at least one $z' \in A(z)$ if and only if $z \in T$ is desirable. ■

16 **Exercise.** Show that whenever the search function $a(\cdot)$ (or $A(\cdot)$) and the stop rule $c(\cdot)$ are continuous, assumption (ii) of theorem (3) (theorem (10)) is satisfied. Show that this is also true for the assumptions stated in exercise (15). ■

The convergence theorems (3) and (10) can be thought of as being extensions of Lyapunov's second method for the study of the stability of

dynamical systems described by difference equations. Weaker versions of these theorems can be found in Polyak [P6] and in Zangwill [Z1] and [Z2]; related ideas appear in the work of Varaiya [V3], Levitin and Polyak [L2], Topkis and Veinott [T1], Hurt [H4], and, in somewhat less explicit form in Arrow *et al.* [A3], Zoutendijk [Z4], and Zukhovitskii *et al.* [Z7]. The author first presented these theorems in [P1–P4] without being aware of the rather close result due to his namesake, Polyak [P6], which was buried in a seemingly unimportant paper on gradient methods.

Algorithms of the form (2) or (9) make sense only if one can compute the point $z_{i+1} = a(z_i)$, or a point $z_{i+1} \in A(z_i)$, in a practically acceptable manner. Now, one often invents algorithms of the form (2) or (9) in which the calculation of $a(z_i)$, or of points in the set $A(z_i)$, cannot be carried out in any practical manner. For example, the computation of such a point may require us to find the limit point of a sequence which we must first construct. As we have already mentioned, to resolve the practical difficulty this introduces, we must introduce suitable truncation, or approximation procedures. We shall now describe a few approximation procedures which were first presented in [P3]. (An alternative approach is described and illustrated in Appendix A and in [M8].)

The most simple-minded approach to introducing a truncation procedure into algorithm (2) is to define an approximations set,

$$17 \quad A_\epsilon(z) = \{y \in T \mid \|y - a(z)\|_{\mathcal{B}} \leq \epsilon\}, \quad \text{where } \epsilon \geq 0, z \in T.$$

We can then modify (2) as follows:

18 **Algorithm Model.** Suppose that an $\epsilon_0 > 0$ is given.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.

Step 3. Compute a $y \in A_\epsilon(z_i)$.

Step 4. If $c(y) - c(z_i) \leq -\epsilon$, set $z_{i+1} = y$, set $i = i + 1$,* and go to step 2; else, set $\epsilon = \epsilon/2$ and go to step 3. ■

The above algorithm will never stop after a finite number of iterations, since no stop commands are included. One of two things will occur. Either algorithm (18) will construct an infinite sequence $\{z_i\}$, or else it will jam up at a point z_k and cycle between steps 3 and 4, dividing ϵ by 2 at each cycle. We shall now show what kind of points one can compute using algorithm (18).

19 **Theorem.** Suppose that the search function $a(\cdot)$ in (17) and the stop rule $c(\cdot)$ in step 4 of algorithm (18) satisfy condition (ii) of theorem (3) and

* Convention: In all algorithms, a sequence of commands such as "set," appearing in the instructions of a step, must be executed in the order in which they occur in the instructions.

that $c(\cdot)$ is uniformly continuous on T . Then either algorithm (18) jams up at a desirable point after a finite number of iterations, or else it constructs an infinite sequence $\{z_i\}$ and every accumulation point of this sequence is desirable.

Proof. Suppose that the algorithm jammed up at the point z_k . Then it must be cycling between steps 3 and 4, producing in this process a sequence of vectors y_j , $j = 0, 1, 2, \dots$, in T such that $\|y_j - a(z_k)\|_{\mathcal{B}} \leq \epsilon_0/2^j$ and $c(y_j) - c(z_k) > -\epsilon_0/2^j$. Hence, as $j \rightarrow \infty$, $y_j \rightarrow a(z_k)$, and since $c(\cdot)$ is assumed to be continuous, we must have $c(a(z_k)) \geq c(z_k)$, i.e., z_k is desirable.

Now suppose that the sequence $\{z_i\}$ is infinite and that $z_i \rightarrow z'$ for $i \in K \subset \{0, 1, 2, \dots\}$, with z' a nondesirable point. Then there exists an integer $k \in K$ such that for $i \in K$, $i \geq k$,

$$20 \quad \|z_i - z'\|_{\mathcal{B}} \leq \epsilon', \quad \epsilon' > 0,$$

and

$$21 \quad c(a(z_i)) - c(z_i) \leq \delta', \quad \delta' < 0,$$

where ϵ' in (20) and δ' in (21) satisfy condition (ii) of theorem (3) with respect to the point z' . Now, since $c(\cdot)$ is uniformly continuous on T , there must exist an $\epsilon'' > 0$ such that $c(y) - c(a(z)) \leq -\delta'/2$ for all $z \in T$ and for all $y \in A_{\epsilon''}(z)$, and hence we must have

$$22 \quad c(y) - c(z_i) \leq \delta'/2 \quad \text{for all } y \in A_{\epsilon''}(z_i), \quad \text{with } i \in K, \quad i \geq k.$$

Let $\tilde{\epsilon} = \min\{\epsilon'', -\delta'/2\}$, and let $\tilde{\epsilon} = \epsilon_0/2^j$, where j is an integer such that

$$23 \quad \epsilon_0/2^j \leq \tilde{\epsilon} \leq \epsilon_0/2^{j-1}.$$

Then, for every $i \in K$, $i \geq k$, and for every $y \in A_{\tilde{\epsilon}}(z_i)$,

$$24 \quad c(y) - c(z_i) \leq -\tilde{\epsilon}.$$

We therefore conclude that at every z_i , $i \in K$, $i \geq k$, the value of ϵ used in step 3 of (18) to compute the point z_{i+1} will be at least $\tilde{\epsilon}$. Consequently, if i , $i + j$ are two consecutive indices in K , with $i \geq k$, then

$$25 \quad c(z_{i+j}) - c(z_i) < \delta'/2 < 0$$

(as in (8)). Consequently, $c(z_i)$ cannot converge to $c(z')$, which contradicts our assumption that $c(\cdot)$ is continuous. Hence the theorem must be true. ■

By and large, to make sure that a point y is in $A_{\epsilon}(z)$ may be almost as difficult as to compute $a(z)$, a task we set out to avoid. Hence we are led to using approximations similar to $A_{\epsilon}(z)$, but ones that do not require us to perform difficult tests. We now introduce an algorithm model which generalizes (18) and which has a considerably broader range of applications. It constitutes a considerably more sophisticated approach to the implementa-

tion of conceptual algorithms, as will become clear from the chapters to follow.*

26 Algorithm Model. $A : \mathbb{R}^+ \times T \rightarrow 2^T$, $c : T \rightarrow \mathbb{R}^1$, $\epsilon_0 > 0$, $\epsilon' \in (0, \epsilon_0)$.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.

Step 3. Compute a $y \in A(\epsilon, z_i)$.

Step 4. If $c(y) - c(z_i) \leq -\epsilon$, set $z_{i+1} = y$, set $i = i + 1$, and go to step 2; else, go to step 5.

Step 5. If $\epsilon \leq \epsilon'$, perform a test to determine if z_i is desirable and go to step 6; else, set $\epsilon = \epsilon'/2$ and go to step 3.

Step 6. If z_i is desirable, set $z_{i+1} = z_i$ and stop; else, set $\epsilon = \epsilon/2$ and go to step 3. ■

27 Theorem. Consider algorithm (26). Suppose that

(i) $c(\cdot)$ is either continuous at all nondesirable points $z \in T$, or else $c(z)$ is bounded from below for $z \in T$;

(ii) for every $z \in T$ which is not desirable, there exist an $\epsilon(z) > 0$, a $\delta(z) < 0$, and a $\gamma(z) > 0$, such that

28

$$c(z'') - c(z') \leq \delta(z) < 0$$

for all $z' \in B(z, \epsilon(z))$, for all $z'' \in A(\gamma, z')$, for all $\gamma \in [0, \gamma(z)]$,

where $B(z, \epsilon(z)) = \{z' \in T \mid \|z' - z\|_{\mathcal{B}} \leq \epsilon(z)\}$.

Then either the sequence $\{z_i\}$ constructed by algorithm (26) is finite and its last element is desirable, or else it is infinite and every accumulation point of $\{z_i\}$ is desirable.

Proof. First we must show that algorithm (26) is well-defined, i.e., that it cannot jam up at a nondesirable point z_k . (It is quite clear that it cannot jam up at a desirable point without the stop command in step 6 being executed.) Thus, suppose that the algorithm jams up at z_k , a nondesirable point. Then the algorithm must be producing an infinite sequence of vectors $y_j \in A(\epsilon_0/2^j, z_k)$, $j = 0, 1, 2, \dots$, such that $c(y_j) - c(z_k) > -\epsilon_0/2^j$, which prevents the construction of z_{k+1} . However, by assumption (ii), since z_k is not desirable, there exist an $\epsilon(z_k) > 0$, a $\delta(z_k) < 0$, and a $\gamma(z_k) > 0$ for which (28) holds. Also, there exists an integer $j' > 0$ such that $\epsilon_0/2^{j'} \leq \min\{-\delta(z_k), \gamma(z_k)\}$, and hence, from (28), we must have $c(y_j) - c(z_k) \leq \delta(z_k) \leq -\epsilon_0/2^j$ for all $j \geq j'$, which contradicts our hypothesis that the algorithm jammed up at z_k . Consequently, algorithm (26) is well-defined.

Now, since the construction of a new point z_{i+1} can only stop when

*Note that we abuse notation in the use of the symbol A . The A in (26) is not the same as the A in (9) or (18).

the stop command in step 6 of (26) is executed, the case of the sequence $\{z_i\}$ being finite is trivial. Hence, let us suppose that the sequence $\{z_i\}$ is infinite and that $z_i \rightarrow z'$ for $i \in K \subset \{0, 1, 2, \dots\}$, where the accumulation point z' is not desirable. Then, by (ii), there exist an $\epsilon(z') > 0$, a $\delta(z') < 0$, and a $\gamma(z') > 0$ for which (28) holds. Since $z_i \rightarrow z'$ for $i \in K$, there must exist a $k' \in K$ such that $z_i \in B(z', \epsilon(z'))$ for all $i \in K$, $i \geq k'$, and there is also an integer k such that $\epsilon_0/2^k \leq \min\{-\delta(z'), \gamma(z')\}$. Hence, for any two consecutive points $z_i, z_{i+1}, i \geq k'$, of the subsequence $\{z_i\}_{i \in K}$, we must have,

$$\begin{aligned} 29 \quad c(z_{i+1}) - c(z_i) &= [c(z_{i+1}) - c(z_{i+1}-1)] + \cdots + [c(z_{i+1}) - c(z_i)] \\ &< c(z_{i+1}) - c(z_i) \leq -\epsilon_0/2^k, \end{aligned}$$

which shows that the sequence $c(z_i)$, $i \in K$, is not Cauchy (i.e., it does not converge). However, $c(z_i)$, $i \in K$, must converge because of assumption (i), and hence we get a contradiction. ■

- 30 **Remark.** In some of the algorithms, such as (2.1.16), that we shall encounter later, it is easy to determine whether z_i is desirable in the process of calculating a $y \in A(\epsilon, z_i)$. In such cases (as was done in (2.1.16)), step 5 of (26) can be eliminated altogether, provided that the test for the desirability of z_i together with the corresponding stop command are incorporated into an expanded version of step 3. The reader should have no difficulty in making a mental adjustment for variations of this kind. ■
- 31 **Exercise.** Show that under the assumptions of theorem (19), the approximation map $A_\epsilon(\cdot)$ and the stop rule $c(\cdot)$ satisfy condition (ii) of theorem (27). ■
- 32 **Exercise.** Show that under the assumptions of theorem (27), the algorithm below has the same convergence properties as algorithm (26). ■
- 33 **Algorithm Model.** $A : \mathbb{R}^+ \times T \rightarrow 2^T$, $c : T \rightarrow \mathbb{R}^1$, $\epsilon_0 > 0$, $\epsilon' \in (0, \epsilon_0)$, $\alpha > 0$, $\beta \in (0, 1)$.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.

Step 3. Compute a $y \in A(\epsilon, z_i)$.

Step 4. If $c(y) - c(z_i) \leq -\alpha\epsilon$, set $z_{i+1} = y$, set $i = i + 1$, and go to step 2; else, go to step 5.

Step 5. If $\epsilon \leq \epsilon'$, perform a test to determine if z_i is desirable and go to step 6; else, set $\epsilon = \beta\epsilon$ and go to step 3.

Step 6. If z_i is desirable, set $z_{i+1} = z_i$ and stop; else, set $\epsilon = \beta\epsilon$ and go to step 3. ■

In algorithm model (26) and in its generalization (33), we begin each iteration with $\epsilon = \epsilon_0$. In some situations this may prove to be inefficient, since, as z_i approaches a desirable point \hat{z} , we may be spending too much

time in the loop which decreases ϵ to an acceptable value. When this is known to be the case, all we need to do is to change the model (33) slightly, as shown below, to make it time-variant.

- 34 Algorithm Model.** $A : \mathbb{R}^+ \times T \rightarrow 2^T$, $c : T \rightarrow \mathbb{R}^1$, $\bar{\epsilon}_0 > 0$, $\epsilon' \in (0, \bar{\epsilon}_0)$, $\alpha > 0$, $\beta \in (0, 1)$.

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \bar{\epsilon}_0$.

Step 3. Compute a $y \in A(\epsilon, z_i)$.

Step 4. If $c(y) - c(z_i) \leq -\alpha\epsilon$, set $z_{i+1} = y$, set $\epsilon_i = \epsilon$, set $i = i + 1$, and go to step 3; else, go to step 5.

Comment. Algorithm (34) is time-varying, because from step 4 we return not to step 2, as in (33), but to step 3.

Comment. Do not store ϵ_i ; this quantity is only introduced for the purpose of proving the theorem below.

Step 5. If $\epsilon \leq \epsilon'$, perform a test to determine if z_i is desirable and go to step 6; else, set $\epsilon = \beta\epsilon$ and go to step 3.

Step 6. If z_i is desirable, set $z_{i+1} = z_i$ and stop; else, set $\epsilon = \beta\epsilon$ and go to step 3. ■

- 35 Theorem.** Suppose that the assumptions (i) and (ii) of (27) are satisfied by the maps $A(\cdot, \cdot)$ and $c(\cdot)$. If $\{z_i\}$ is a sequence constructed by algorithm (34), then either $\{z_i\}$ is finite and its last element is desirable, or $\{z_i\}$ is infinite and every accumulation point of this sequence is desirable.

Proof. First, making use of exactly the same arguments as in the first paragraph of the proof of theorem (27), we conclude that algorithm (34) cannot jam up at a point z_k , while constructing an infinite sequence of vectors $y_j \in A(\beta^j \epsilon_{k-1}, z_k)$ such that $c(y_j) - c(z_k) > \beta^j \epsilon_{k-1}$. Hence, algorithm (34) is well-defined.

Next, because of the nature of the stop command in step 5, it is quite obvious that if the sequence $\{z_i\}$ is finite, then both its last element and its next to last element are desirable (they are also identical). Consequently, we only need to consider the case when the sequence $\{z_i\}$ is infinite. Thus, suppose that the point \hat{z} is an accumulation point of $\{z_i\}$ and that it is not desirable. Then, according to (28), there exist an $\hat{\epsilon} > 0$, a $\delta < 0$, and a $\hat{\gamma} > 0$ such that

$$\begin{aligned} 36 \quad & c(z'') - c(z') \leq \delta < 0 \\ & \text{for all } z' \in B(\hat{z}, \hat{\epsilon}), \quad \text{for all } z'' \in A(\gamma, z'), \quad \text{for all } \gamma \in [0, \hat{\gamma}], \\ & \text{where } B(\hat{z}, \hat{\epsilon}) = \{z' \in T \mid \|z' - \hat{z}\|_{\mathcal{B}} \leq \hat{\epsilon}\}. \end{aligned}$$

Now suppose that $K \subset \{0, 1, 2, \dots\}$ is such that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, $i \in K$. Then there must exist a $k' \in K$ such that $z_i \in B(\hat{z}, \hat{\epsilon})$ for all $i \geq k'$, $i \in K$. Now, since $\{\epsilon_i\}_{i=0}^{\infty}$ is a monotonically decreasing sequence, we must have either that $\epsilon_i > \min\{-\delta, \hat{\gamma}\} \triangleq \tilde{\epsilon}$ for all $i = 0, 1, 2, \dots$, or else there exists an integer $k \geq k'$ such that $\epsilon_i \leq \tilde{\epsilon}$ for all $i \geq k'$. Suppose, therefore, that $\epsilon_i > \tilde{\epsilon}$ for $i = 0, 1, 2, \dots$. Then, according to the instruction in step 4 of (34),

$$37 \quad c(z_{i+1}) - c(z_i) \leq -\alpha \tilde{\epsilon} \quad \text{for all } i \geq 0.$$

Now, let $i, i + j$ be any two consecutive indices in K . Then

$$38 \quad c(z_{i+j}) - c(z_i) = [c(z_{i+j}) - c(z_{i+j-1})] + \dots + [(c(z_{i+1}) - c(z_i)] \leq -j\alpha \tilde{\epsilon},$$

which shows that the sequence $\{c(z_i)\}_{i \in K}$ is not Cauchy and hence cannot converge. But this contradicts assumption (i) of (27), since by (i) the monotonically decreasing sequence $\{c(z_i)\}_{i \in K}$ is bounded from below and therefore must converge. We therefore conclude that $\epsilon_i > \tilde{\epsilon}$, for $i = 0, 1, 2, \dots$, is not possible.

Since we must have $\epsilon_i \leq \tilde{\epsilon}$ for all $i \geq k$, we conclude from (36) that

$$39 \quad c(z_{i+1}) - c(z_i) \leq \delta < 0 \quad \text{for all } i \in K, \quad i \geq k.$$

Hence, let $i, i + j$ be any two consecutive indices in K , with $i \geq k$. Then,

$$40 \quad c(z_{i+j}) - c(z_i) = [c(z_{i+j}) - c(z_{i+j-1})] + \dots + [c(z_{i+1}) - c(z_i)] < \delta,$$

which shows again that the sequence $\{c(z_i)\}_{i \in K}$ is not Cauchy and hence cannot converge. However, this again contradicts assumption (i) of (27), since because of (i) in (27), the sequence $\{c(z_i)\}_{i \in K}$ must converge. We therefore conclude that if \hat{z} is an accumulation point of $\{z_i\}$, it must be desirable. ■

In some algorithms, such as the method of feasible directions (4.3.20), it would be too costly to perform a test of the form “ $c(y) - c(z_i) \leq -\epsilon$?” because of the cost of evaluating $c(y)$. In such cases it may be possible to use a simpler, substitute test involving a function different from $c(\cdot)$. We shall now describe a model which was constructed by the author in collaboration with Robert Klessig, a graduate student. This model was designed to help us understand the nature of the time-varying versions of methods of feasible directions and gradient projection algorithms that we shall encounter in Chapter 4. As we shall see, it permits us to use a substitute test, but because of this, it is much more complex than all the preceding models.

To define this model, we shall need four maps: $c : T \rightarrow \mathbb{R}^1$, $\psi : \mathbb{R}^+ \times T \rightarrow \mathbb{R}^-$, $H : \mathbb{R}^+ \times T \rightarrow S \subset \mathcal{B}$, and $M : T \times \mathcal{B} \rightarrow 2^T$. We shall state the properties with which we need to endow these functions and the subset S of \mathcal{B} later.

- 41 **Algorithm Model.** $H : \mathbb{R}^+ \times T \rightarrow S \subset \mathcal{B}$, $M : T \times \mathcal{B} \rightarrow 2^T$, $\psi : \mathbb{R}^+ \times T \rightarrow \mathbb{R}^-$, $c : T \rightarrow \mathbb{R}^1$, $\tilde{\epsilon}_0 > 0$, $\epsilon' \in (0, \tilde{\epsilon}_0)$, $\alpha > 0$, $\beta \in (0, 1)$.

- Step 0.* Compute a $z_0 \in T$.
Step 1. Set $i = 0$.
Step 2. Set $\epsilon = \bar{\epsilon}_0$.
Step 3. Compute a vector $h \in H(\epsilon, z_i)$.
Step 4. Compute $\psi(\epsilon, z_i)$.
Step 5. If $\psi(\epsilon, z_i) \leq -\alpha\epsilon$, go to step 8; else, go to step 6.
Step 6. If $\epsilon \leq \epsilon'$, compute $\psi(0, z_i)$ and go to step 7; else, set $\epsilon = \beta\epsilon$ and go to step 3.
Step 7. If $\psi(0, z_i) = 0$, set $z_{i+1} = z_i$ and stop; else, set $\epsilon = \beta\epsilon$ and go to step 3.
Step 8. Compute a vector $y \in M(z_i, h)$, and go to step 9.
Step 9. Set $z_{i+1} = y$, set $h_i = h$, set $\epsilon_i = \epsilon$, set $i = i + 1$, and go to step 3.

Comment. Do not store the quantities ϵ_i and h_i . These quantities are only introduced so as to facilitate the proof of the theorem below. ■

42 Theorem. Consider algorithm (41). Suppose that

- (i) if $\psi(0, z) = 0$, then z is desirable;
- (ii) the sets T and S are compact;
- (iii) $c(z)$ is bounded from below for $z \in T$;
- (iv) for any $\epsilon \geq 0$, for any $z \in T$, if

$$43 \quad \psi(\epsilon, z) \leq -\mu, \quad \mu \geq 0,$$

then there exists an $s(\mu) > 0$ such that

$$44 \quad c(z + th) - c(z) \leq -t\mu/2, \quad \text{for all } t \in [0, s(\mu)], \quad \text{for all } h \in H(\epsilon, z);$$

(v) for any $z \in T$, for any $\delta > 0$, there exists an $\epsilon(z, \delta) > 0$ such that

$$45 \quad \psi(\epsilon, z') \leq \psi(0, z) + \delta, \quad \text{for all } \epsilon \in [0, \epsilon(z, \delta)], \quad \text{for all } z' \in B(z, \epsilon(z, \delta)),$$

where $B(z, \epsilon(z, \delta)) = \{z' \in T \mid \|z' - z\|_{\mathcal{B}} \leq \epsilon(z, \delta)\}$, i.e., $\psi(\cdot, \cdot)$ is upper semicontinuous on $\{0\} \times T$;

(vi) given any $\epsilon > 0$, there exists a $\sigma(\epsilon) > 0$ such that if

$$46 \quad \psi(\epsilon, z) \leq -\alpha\epsilon,$$

then

$$47 \quad c(y) - c(z) \leq -\sigma(\epsilon)\epsilon,$$

for all $z \in T$, for all $h \in H(\epsilon, z)$, for all $y \in M(z, h)$;

(vii) for any $z \in T$, for any $h \in \mathcal{B}$,

$$48 \quad M(z, h) = \{(z + \lambda h) \in T \mid c(z + \lambda h) = \min\{c(z + \tilde{\lambda}h) \mid \tilde{\lambda} \geq 0, (z + \tilde{\lambda}h) \in T \text{ for all } \lambda' \in [0, \tilde{\lambda}]\}\}.$$

If $\{z_i\}$ is a sequence constructed by algorithm (41), then either $\{z_i\}$ is finite and its last element is desirable, or $\{z_i\}$ is infinite and every accumulation point of $\{z_i\}$ is desirable.

Proof. First we must show that algorithm (41) is well-defined, i.e., that it cannot jam up at a point z_k , while constructing an infinite sequence of vectors $h_j \in H(\beta^j \epsilon_{k-1}, z_k)^*$ and at the same time finding that $\psi(\beta^j \epsilon_{k-1}, z_k) \leq -\alpha \beta^j \epsilon_{k-1}$. Suppose that $\psi(0, z_k) = 0$. Then, for some integer $j \geq 0$, $\beta^j \epsilon_{k-1} \leq \epsilon'$ and hence at that point, algorithm (41), in step 6, will compute $\psi(0, z_k)$, pass on to step 7, set $z_{k+1} = z_k$ and stop. Next, suppose that $\psi(0, z_k) < 0$. Then, by (v) of (42), there exists an $\epsilon'' > 0$ such that $\psi(\epsilon, z_k) \leq \psi(0, z_k)/2$ for all $\epsilon \in [0, \epsilon'']$. Let j be any positive integer such that $\beta^j \epsilon_{k-1} \leq \min\{\epsilon'', -\psi(0, z_k)/2\alpha\}$. Then we must have $\psi(\beta^j \epsilon_{k-1}, z_k) \leq -\alpha \beta^j \epsilon_{k-1}$, and hence the algorithm cannot jam up at a point z_k .

Next, because of the nature of the stop command in step 7 of (41), we see that the case of $\{z_i\}$ finite is trivial. Hence, suppose that the sequence $\{z_i\}$ is infinite. We begin by showing that the associated sequence $\{\epsilon_i\}$ converges to zero. Since $\{\epsilon_i\}$ is a monotonically decreasing sequence, bounded from below by zero, it must converge. Hence, suppose that $\epsilon_i \rightarrow \epsilon^*$ as $i \rightarrow \infty$ and suppose that $\epsilon^* > 0$. Then, since $\epsilon_i = \beta^{j(i)} \tilde{\epsilon}_0$, where $j(i)$ is a positive integer, there exists an integer k such that $\epsilon_i = \epsilon^*$, for all $i \geq k$. Hence,

$$49 \quad \psi(\epsilon_{i-1}, z_i) = \psi(\epsilon^*, z_i) \leq -\alpha \epsilon^*, \quad \text{for all } i \geq k + 1.$$

Making use of (47), we conclude that

$$50 \quad c(z_{i+1}) - c(z_i) \leq -\sigma(\epsilon^*) \epsilon^*, \quad \text{for all } i \geq k + 1,$$

which implies that $c(z_i) \rightarrow -\infty$ as $i \rightarrow \infty$. But this contradicts (iii). Hence we must have $\epsilon^* = 0$.

Next, suppose that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$. Then, since $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$, there must exist an infinite set of indices $K \subset \{0, 1, 2, 3, \dots\}$ such that

$$51 \quad \psi(\epsilon_{i-1}, z_i) > -\alpha \epsilon_{i-1} \quad \text{for all } i \in K.$$

Combining (51) with assumption (v), and making use of the fact that $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$, we obtain

$$52 \quad 0 \leq \overline{\lim}_{i \in K} \psi(\epsilon_{i-1}, z_i) \leq \psi(0, \hat{z}) \leq 0,$$

which implies that $\psi(0, \hat{z}) = 0$, and hence that \hat{z} is desirable.

Finally, consider the last possibility, namely, suppose that $\{z_i\}$ does not converge. Since $\{z_i\} \subset T$ and T is compact by assumption (ii), the sequence

* These are not the h_i defined in step 9.

$\{z_i\}$ must have at least two accumulation points. Let us denote these two accumulation points by z^* and \hat{z} . Suppose that

$$53 \quad \psi(0, z^*) = -2\mu < 0,$$

i.e., suppose that z^* is not desirable. We shall show that this leads to a contradiction. First, because of assumption (v), there exists a $\delta_1 > 0$ such that

$$54 \quad \psi(\epsilon, z) \leq -\mu \quad \text{for all } \epsilon \in [0, \delta_1], \quad \text{for all } z \in B(z^*, \delta_1).$$

Next, since $z^* \neq \hat{z}$, there exists a $\delta_2 \in (0, \delta_1/2]$ such that

$$55 \quad B(z^*, 2\delta_2) \cap B(\hat{z}, 2\delta_2) = \emptyset.$$

Since z^* and \hat{z} are both accumulation points, $B(z^*, \delta_2)$ and $B(\hat{z}, 2\delta_2)$ must each contain an infinite number of elements of the sequence $\{z_i\}$. Suppose that for some $j \in \{0, 1, 2, \dots\}$, $z_j \in B(z^*, \delta_2)$. Then, since $\{z_i\}$ does not converge to z^* , and since \hat{z} is also an accumulation point of $\{z_i\}$, there must exist a finite integer $\bar{n}(j)$ such that $z_{j+\bar{n}(j)} \in B(\hat{z}, 2\delta_2)$. Because of (55), we must therefore have $z_{j+\bar{n}(j)} \notin B(z^*, 2\delta_2)$. Consequently, there must exist an integer $n(j)$, $j < n(j) \leq \bar{n}(j)$ such that $z_{j+n(j)-1} \in B(z^*, 2\delta_2)$, and $z_{j+n(j)} \notin B(z^*, 2\delta_2)$. Now, for $i = 0, 1, 2, \dots$, let λ_i denote the step size chosen by the algorithm, i.e., suppose that

$$56 \quad z_{i+1} = z_i + \lambda_i h_i, \quad i = 0, 1, 2, 3, \dots$$

Then we must have, since $z_j \in B(z^*, \delta_2)$ and $z_{j+n(j)} \notin B(z^*, 2\delta_2)$,

$$57 \quad \delta_2 \leq \|z_{j+n(j)} - z_j\|_{\mathcal{B}} = \left\| \sum_{p=0}^{n(j)-1} \lambda_{j+p} h_{j+p} \right\|_{\mathcal{B}} \leq \sum_{p=0}^{n(j)-1} \lambda_{j+p} \|h_{j+p}\|_{\mathcal{B}}.$$

Since S is compact by assumption (ii), $m = \max\{\|h\|_{\mathcal{B}} \mid h \in S\}$ exists and satisfies $m < \infty$. Hence, from (57),

$$58 \quad \sum_{p=0}^{n(j)-1} \lambda_{j+p} \geq \frac{\delta_2}{m} > 0, \quad \text{for all } j \text{ such that } z_j \in B(z^*, \delta_2).$$

Let $s(\mu)$ be as defined in assumption (iv) (see (44)), with μ as in (54), and let

$$59 \quad \eta_i = \min\{\lambda_i, s(\mu)\}, \quad \text{for } i = 0, 1, 2, \dots$$

Then, making use of (44) and (48),

$$60 \quad c(z_{i+1}) - c(z_i) \leq c(z_i + \eta_i h_i) - c(z_i) \leq -\eta_i \mu / 2 \quad \text{for all } z_i \in B(z^*, 2\delta_2).$$

Hence, for all $z_j \in B(z^*, \delta_2)$,

$$\begin{aligned} 61 \quad c(z_{j+n(j)}) - c(z_j) &= \sum_{p=0}^{n(j)-1} [c(z_{j+p+1}) - c(z_{j+p})] \\ &\leq \sum_{p=0}^{n(j)-1} [c(z_{j+p} + \eta_{j+p} h_{j+p}) - c(z_{j+p})] \leq - \left(\sum_{p=0}^{n(j)-1} \eta_{j+p} \right) \frac{\mu}{2}. \end{aligned}$$

Making use of (58) and (59), we now obtain,

$$62 \quad \sum_{p=0}^{n(j)-1} \eta_{j+p} \geq \min \left\{ \sum_{i=0}^{n(j)-1} \lambda_{j+p}, n(j) s(\mu) \right\} \geq \min \left\{ \frac{\delta_2}{m}, s(\mu) \right\}.$$

We now conclude from (61) and (62) that

$$63 \quad c(z_{j+n(j)}) - c(z_j) \leq - \frac{1}{2} \mu \min \left\{ \frac{\delta_2}{m}, s(\mu) \right\} \quad \text{for all } z_j \in B(z^*, \delta_2).$$

But since z^* and \hat{z} are both accumulation points of $\{z_i\}$, there exists a subsequence $\{z_{i_j}\}_{j=0}^\infty \in B(z^*, \delta_2)$ such that $i_{j+1} \geq i_j + n(i_j)$ for all $j \in \{0, 1, 2, 3, \dots\}$. Hence, because of (63),

$$64 \quad c(z_{i_{j+1}}) - c(z_{i_j}) \leq - \frac{1}{2} \mu \min \left\{ \frac{\delta_2}{m}, s(\mu) \right\} \quad \text{for all } j \in \{0, 1, 2, \dots\}.$$

But (64) implies that $c(z_{i_j}) \rightarrow -\infty$ as $j \rightarrow \infty$, which, in turn, implies that $c(z_i) \rightarrow -\infty$ as $i \rightarrow \infty$, since $\{c(z_i)\}$ is monotonically decreasing. Consequently (53) must be false, i.e., we must have $\psi(0, z^*) = 0$, which implies that z^* is desirable. ■

Finally, we state two important sufficient conditions which guarantee that an infinite sequence $\{z_i\}$, constructed by an algorithm model, converges to a desirable point.

- 65 **Theorem.** Consider any one of the algorithm models in this section and suppose that it satisfies the assumptions of the corresponding convergence theorem. In addition, suppose that $c(z)$ is bounded from below for all $z \in T$ and that if $z' \neq z''$ are two desirable points in T , then $c(z') \neq c(z'')$. Let $\{z_i\}_{i=0}^\infty$ be an infinite sequence in T constructed by this algorithm model. If either T or $C'(z_0) = \{z \in T \mid c(z) \leq c(z_0)\}$ is compact, then $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, where \hat{z} is a desirable point.

Proof. Since the sequence $\{z_i\}$ is compact, it must have accumulation points which, as we have already shown, must be desirable. Since $\{c(z_i)\}$ is a monotonically decreasing sequence which is bounded from below, it must

converge, i.e., $c(z_i) \rightarrow c_*$ as $i \rightarrow \infty$. Consequently, if $z' \neq z''$ are both accumulation points of $\{z_i\}$, then we must have $c(z') = c(z'') = c_*$, which contradicts our assumption that $c(z') \neq c(z'')$ whenever $z' \neq z''$ are both desirable. Hence, $\{z_i\}$ has exactly one accumulation point. ■

- 66 Theorem.** Consider any one of the algorithm models in this section and suppose that it satisfies the assumptions of the corresponding convergence theorem. In addition, suppose that the set T contains only a finite number of desirable points. Let $\{z_i\}_{i=0}^{\infty}$ be an infinite sequence constructed by this algorithm model. If either T or $C'(z_0) = \{z \in T \mid c(z) \leq c(z_0)\}$ is compact, and $(z_{i+1} - z_i) \rightarrow 0$ as $i \rightarrow \infty$, then $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, where \hat{z} is a desirable point. ■
- 67 Exercise.** Prove theorem (66). ■

We now proceed to examine algorithms in the light of the models and corresponding convergence theorems that we have introduced in this section.

2

UNCONSTRAINED MINIMIZATION

2.1 Gradient and Quasi-Newton Methods in \mathbb{R}^n

We begin this chapter by considering the problem of minimizing a continuously differentiable function $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$. We write this problem in the shorthand notation,

$$1 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\}.$$

We shall assume that we can find a $z_0 \in \mathbb{R}^n$ such that the set

$$2 \quad C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$$

is bounded (it is obviously closed).

As we have already pointed out in discussing optimality conditions, there is no satisfactory test for establishing, in the general case, whether a given $z' \in \mathbb{R}^n$ is a solution to (1) or not. Hence, all the existing algorithms that can be implemented are algorithms that find points $z' \in \mathbb{R}^n$ at which $\nabla f^0(z') = 0$. *We shall therefore say that a point $z' \in \mathbb{R}^n$ is desirable if $\nabla f^0(z') = 0$.* We recall from theorem (1.2.18) that under the assumption that $f^0(\cdot)$ is convex, points satisfying $\nabla f^0(z') = 0$ are *optimal* for (1), i.e., $f^0(z') = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$.

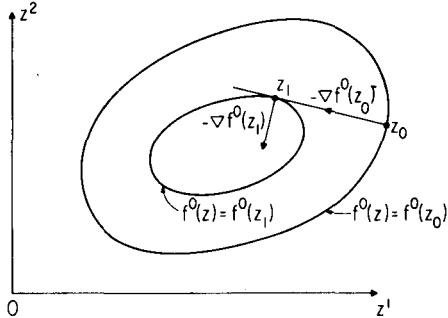
Gradient and quasi-Newton methods are derived from the following conceptual algorithm (it requires us to minimize a function along a line at each iteration; an operation which we declare to be inadmissible in a practical algorithm):

- 3 **Algorithm.** Let $D(z)$ be an $n \times n$ positive definite matrix whose elements are continuous functions of z .

- Step 0.* Select a $z_0 \in \mathbb{R}^n$ such that the set defined in (2) is bounded.
Step 1. Set $i = 0$.
Step 2. Compute $-D(z_i) \nabla f^0(z_i)$.
Step 3. Set $h(z_i) = -D(z_i) \nabla f^0(z_i)$. If $h(z_i) = 0$, stop; else, go to step 4.
Step 4. Compute the scalar $\lambda(z_i)$ to be the smallest nonnegative scalar satisfying

$$4 \quad f^0(z_i + \lambda(z_i) h(z_i)) = \min\{f^0(z_i + \lambda h(z_i)) \mid \lambda \geq 0\}.$$

- Step 5.* Set $z_{i+1} = z_i + \lambda(z_i) h(z_i)$, set $i = i + 1$, and go to step 2. ■



Method of steepest descent: algorithm (3) with $D(\cdot) = I$.

If we set $c(\cdot) = f^0(\cdot)$ and define the search function $a(\cdot)$ by

$$5 \quad a(z) = z + \lambda(z) h(z),$$

where $h(z)$ is as in step 3 of algorithm (3) and $\lambda(z)$ is defined by (4) (with the subscripts deleted), then we see that algorithm (3) is of the same type as algorithm (1.3.2). This means that we can establish its convergence properties by a direct application of theorem (1.3.3), as we shall now do.

- 6 **Theorem.** Suppose that $\{z_i\}$ is a sequence constructed by algorithm (3). Then, either the sequence $\{z_i\}$ is finite, terminating at z_k , and $\nabla f^0(z_k) = 0$, or else it is infinite and every accumulation point z' of $\{z_i\}$ satisfies $\nabla f^0(z') = 0$.

Proof. Since $f^0(\cdot)$ is continuous, it suffices to show that assumption (ii) of theorem (1.3.3) is satisfied for $c(\cdot) = f^0(\cdot)$ and $a(\cdot)$ defined as in (5). Obviously, for every desirable $z \in \mathbb{R}^n$, $a(z) = z$, since $\nabla f^0(z) = 0$. Now, if $z' \in \mathbb{R}^n$ is such that $\nabla f^0(z') \neq 0$, then $\langle \nabla f^0(z'), D(z') \nabla f^0(z') \rangle = \delta' > 0$. Since $\nabla f^0(\cdot)$ is continuous, we conclude from the mean-value theorem (B.1.1) that there exists an ϵ' such that for all $0 \leq \lambda \leq \epsilon'$,

$$7 \quad f^0(z' + \lambda h(z')) - f^0(z') = \lambda \langle \nabla f^0(z' + \alpha h(z')), h(z') \rangle \leq -\lambda \delta'/2,$$

where $\alpha \in [0, \lambda]$. Consequently, whenever z' is not desirable, we must have $\lambda(z') > 0$.

Let $z' \in \mathbb{R}^n$ be such that $\nabla f^0(z') \neq 0$. Then $\lambda(z') > 0$, and we define the map $\theta : \mathbb{R}^n \rightarrow \mathbb{R}^1$ by

$$8 \quad \theta(z) = f^0(z + \lambda(z') h(z)) - f^0(z).$$

By inspection, $\theta(\cdot)$ is a continuous function, and

$$9 \quad \theta(z') = f^0(z' + \lambda(z') h(z')) - f^0(z') = \theta' < 0.$$

Hence there exists an $\epsilon' > 0$ such that

$$|\theta(z) - \theta(z')| \leq -\theta'/2,$$

i.e., such that

$$10 \quad \theta(z) \leq \theta'/2 < 0$$

for all $z \in \{z \mid \|z - z'\| \leq \epsilon'\}$. But, according to (4),

$$11 \quad f^0(z + \lambda(z) h(z)) - f^0(z) \leq \theta(z),$$

and hence, setting $\epsilon(z') = \epsilon'$, $\delta(z') = \theta'/2$, we find that assumption (ii) of theorem (1.3.3) is satisfied. ■

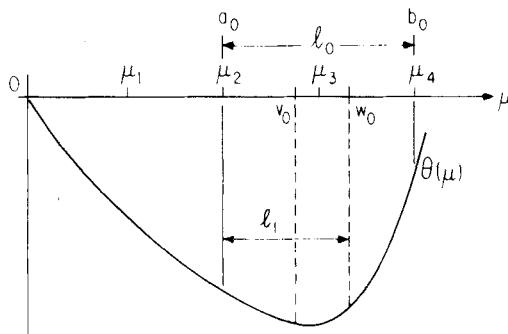
- 12 **Remark.** Note that the sequence of costs, $\{f^0(z_i)\}$, constructed by algorithm (3) is monotonically decreasing. Since $f^0(\cdot)$ is continuous and the set $C(z_0)$ defined in (2) is bounded, the sequence of costs converges ($f^0(z_i) \rightarrow f^0$ as $i \rightarrow \infty$, assuming the sequence is infinite). Also, the sequence $\{z_i\}$ is compact, i.e., it has convergent subsequences. However, there is no point in trying to determine a convergent subsequence of $\{z_i\}$ in order to locate a point z' such that $f^0(z') = f^0$, since the truncation (after a sufficiently large number of iterations) of the sequence $\{z_i\}$ will result in a point z_k close to *some* accumulation point z' of the sequence $\{z_i\}$, and all the accumulation points z' of $\{z_i\}$ result in the same cost, i.e., $f^0(z') = f^0$. ■

- 13 **Exercise.** Show that if in addition to satisfying the assumptions stated, the function $f^0(\cdot)$ is also convex, then algorithm (3) will compute the minima of $f^0(\cdot)$. What happens when $f^0(\cdot)$ is strictly convex? ■

Since the function $a(\cdot)$ defined by (5) cannot be implemented on a digital computer because of the optimization requirement defined by (4), we must take steps to modify algorithm (3) to make it implementable without affecting its convergence properties. When the function $f^0(\cdot)$ is convex and continuously differentiable, and the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is bounded for a particular known z_0 , the modification given below can be quite efficient.

We shall need the Golden section search as a subprocedure and hence we begin by describing it.

Suppose that $\theta : \mathbb{R}^+ \rightarrow \mathbb{R}^1$ is a convex function whose minimum occurs at $\lambda' > 0$, with λ' finite. The following algorithm will construct in a finite number of steps an interval of length $\epsilon > 0$ containing λ' :



Algorithm (14).

- 14 **Algorithm** (Golden section search). The $\epsilon > 0$, $\rho > 0$ to be supplied; $F_1 = (3 - \sqrt{5})/2 \approx 0.38$, $F_2 = (\sqrt{5} - 1)/2 \approx 0.68$.*

Comment. The first six steps of the algorithm compute an interval $[a_0, b_0]$ containing a minimizing point λ' . The remaining steps narrow down the length of this interval to the preassigned value ϵ .

Step 1. Compute $\theta(\rho)$, $\theta(0)$.

Step 2. If $\theta(\rho) \geq \theta(0)$, set $a_0 = 0$, $b_0 = \rho$, and go to step 7; else, go to step 3.

Step 3. Set $i = 0$, $\mu_0 = 0$.

Step 4. Set $\mu_{i+1} = \mu_i + \rho$.

Step 5. Compute $\theta(\mu_{i+1})$.

Step 6. If $\theta(\mu_{i+1}) \geq \theta(\mu_i)$, set $a_0 = \mu_{i-1}$, $b_0 = \mu_{i+1}$, and go to step 7; else, set $i = i + 1$ and go to step 4.

Comment. Now $\lambda' \in [a_0, b_0]$. We proceed to reduce the length of the interval containing λ' .

Step 7. Set $j = 0$.

Step 8. Set $l_j = (b_j - a_j)$.

Step 9. If $l_j \leq \epsilon$, go to step 12; else, go to step 10.

* The numbers F_1 and F_2 are called Fibonacci fractions; they have the property that $F_2 = 1 - F_1$ and that $F_1 = (F_2)^2$.

Step 10. Set $v_j = a_j + F_1 l_j$, $w_j = a_j + F_2 l_j$.

Step 11. If $\theta(v_j) < \theta(w_j)$, set $a_{j+1} = a_j$, set $b_{j+1} = w_j$, set $j = j + 1$, and go to step 8; else, set $a_{j+1} = v_j$, set $b_{j+1} = b_j$, set $j = j + 1$, and go to step 8.

Comment. Note that $l_j = F_2^j l_0 = (0.68)^j l_0$.

Step 12. Set $\bar{\mu} = (a_j + b_j)/2$ and stop. ■

- 15 **Exercise.** Show that when $a_{j+1} = a_j$ and $b_{j+1} = w_j$, we shall have $w_{j+1} = v_j$, and that whenever $a_{j+1} = v_j$ and $b_{j+1} = b_j$, we must have $v_{j+1} = w_j$. ■

Hence, the remarkable property of the Golden section search is that at each iteration we only need to carry out one and not two evaluations of the function $\theta(\cdot)$. In practice, therefore, algorithm (14) would be modified to take this fact into account.

We can now state an implementable modification of algorithm (3) which can be used for finding the minima of differentiable convex functions $f^0(\cdot)$, under the assumption that the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is bounded.

- 16 **Algorithm** (Polak [P3]). Let $D(z)$ be an $n \times n$ positive definite matrix whose elements are continuous functions of z ; $f^0(\cdot)$ is assumed to be convex.

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set defined in (2) is bounded; select an $\epsilon_0 > 0$, and select a $\rho > 0$ for algorithm (14).

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.

Step 3. Compute $-D(z_i) \nabla f^0(z_i)$.

Step 4. Set $h(z_i) = -D(z_i) \nabla f^0(z_i)$. If $h(z_i) = 0$, stop; else, go to step 5.

Step 5. Define $\theta : \mathbb{R}^+ \rightarrow \mathbb{R}^1$ by

$$17 \quad \theta(\mu) = f^0(z_i + \mu h(z_i)) - f^0(z_i).$$

Step 6. Use algorithm (14) to compute $\bar{\mu}$ (see step 12 of (14)), using the current value of ϵ .

Step 7. If $\theta(\bar{\mu}) \leq -\epsilon$, set $z_{i+1} = z_i + \bar{\mu} h(z_i)$, set $i = i + 1$, and go to step 3; else, set $\epsilon = \epsilon/2$ and go to step 6. ■

- 18 **Exercise.** Show that algorithm (16) is of the form of algorithm (1.3.18) and that it satisfies the assumptions of theorem (1.3.19). Hence, show that either algorithm (16) stops* at a point z_k , in which case z_k must minimize the function $f^0(\cdot)$, or else it generates an infinite sequence $\{z_i\}$ such that $f^0(z_i) \rightarrow f^0 = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$. Modify (16) to the form (1.3.33). ■

When the function $f^0(\cdot)$ is not necessarily convex, we can use the following modification of algorithm (3) which uses a step size rule probably first introduced by Goldstein [G3].

* Note that algorithm (16) cannot jam up because of the stop in step 4.

- 19 **Algorithm** (Goldstein [G3]). Let $D(z)$ be an $n \times n$ positive definite matrix whose elements are continuous functions of z .

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set defined in (2) is bounded; select an $\alpha \in (0, \frac{1}{2})$.

Comment. Here, $\alpha = 0.4$ seems to be a good choice; see Section 6.1.

Step 1. Set $i = 0$.

Step 2. Compute $-D(z_i) \nabla f^0(z_i)$.

Step 3. Set $h(z_i) = -D(z_i) \nabla f^0(z_i)$. If $h(z_i) = 0$, stop; else, go to step 4.

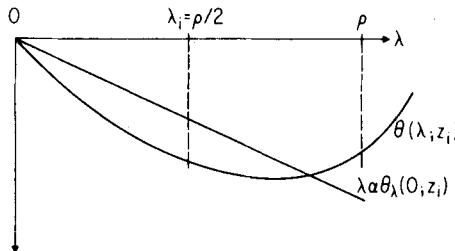
Step 4. Compute a $\lambda_i > 0$ such that

$$20 \quad \lambda_i(1 - \alpha)\langle \nabla f^0(z_i), h(z_i) \rangle \leq \theta(\lambda_i ; z_i) \leq \lambda_i \alpha \langle \nabla f^0(z_i), h(z_i) \rangle,$$

where

$$21 \quad \theta(\lambda_i ; z_i) = f^0(z_i + \lambda_i h(z_i)) - f^0(z_i).$$

Step 5. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 2. ■



Computation of λ_i according to (20): $\theta_\lambda(0; z_i) = \langle \nabla f^0(z_i), h(z_i) \rangle$.

We shall now show that algorithm (19) is of the form (1.3.9) and that it satisfies the assumptions of theorem (1.3.10). Recall that we have defined a point $z' \in \mathbb{R}^n$ to be desirable if $\nabla f^0(z') = 0$.

- 22 **Theorem.** Suppose that $\{z_i\}$ is a sequence constructed by algorithm (19); then, either the sequence $\{z_i\}$ is finite, terminating at z_k , and $\nabla f^0(z_k) = 0$, or else it is infinite and every accumulation point z' of $\{z_i\}$ satisfies $\nabla f^0(z') = 0$.

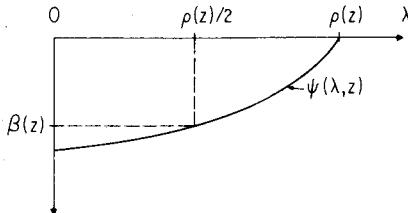
Proof. Referring to the model (1.3.9), we set $c(\cdot) = f^0(\cdot)$ and we define $A : \mathbb{R}^n \rightarrow 2^{\mathbb{R}^n}$ (with $T = \mathbb{R}^n$) as follows:

$$23 \quad A(z) = \{z' = z + \lambda h(z) \mid \lambda \geq 0, \theta(\lambda; z) \geq 0, \bar{\theta}(\lambda; z) \leq 0\},$$

where $h(z)$ is defined as in step 3 of (19) (with the subscripts deleted), and

$$24 \quad \underline{\theta}(\lambda; z) = [f^0(z + \lambda h(z)) - f^0(z)] - \lambda(1 - \alpha)\langle \nabla f^0(z), h(z) \rangle,$$

$$25 \quad \bar{\theta}(\lambda; z) = [f^0(z + \lambda h(z)) - f^0(z)] - \lambda\alpha\langle \nabla f^0(z), h(z) \rangle.$$



Construction for proof of theorem (22).

Now, since the algorithm stops constructing new points if and only if for some z_k , $h(z_k) = 0$ (see step 3), the first part of the theorem is trivial. To show that the second part of the theorem is true, we shall show that the maps $f^0(\cdot)$ and $A(\cdot)$ satisfy the assumptions (i) and (ii) of theorem (1.3.10). Obviously (i) of (1.3.10) is satisfied, since $f^0(\cdot)$ is continuous. Thus we are left with showing that (ii) is satisfied.

Let $\psi(\cdot, \cdot)$, mapping $\mathbb{R}^+ \times \mathbb{R}^n$ into \mathbb{R}^1 , be defined by

$$26 \quad \psi(\lambda, z) = \underline{\theta}(\lambda; z)/\lambda,$$

and let* $\lambda = \rho(z)$ be the smallest positive root of the equation $\psi(\lambda, z) = 0$. Then for every $z \in \mathbb{R}^n$ such that $\nabla f^0(z) \neq 0$, we have $\psi(0, z) = \alpha\langle \nabla f^0(z), h(z) \rangle < 0$, and hence $\rho(z) > 0$ and $\psi(\lambda, z) < 0$ for all $\lambda \in [0, \rho(z)]$. Consequently, for every z such that $\nabla f^0(z) \neq 0$,

$$27 \quad \beta(z) = \max\{\psi(\lambda, z) \mid \lambda \in [0, \rho(z)/2]\} < 0.$$

Now, let $z \in \mathbb{R}^n$ be such that $\nabla f^0(z) \neq 0$ (i.e., z is not desirable). Then, since the interval $[0, \rho(z)/2]$ is compact and since $\psi(\cdot, \cdot)$ is jointly continuous in both its arguments, there exists an $\epsilon' > 0$ such that for all $z' \in \{z' \mid \|z' - z\| \leq \epsilon'\}$ and for all $\lambda \in [0, \rho(z)/2]$,

$$28 \quad |\psi(\lambda, z') - \psi(\lambda, z)| \leq -\beta(z)/2,$$

and hence, since $\psi(\lambda, z) \leq \beta(z)$ for all $\lambda \in [0, \rho(z)/2]$,

$$29 \quad \psi(\lambda, z') \leq \beta(z)/2,$$

for all $z' \in \{z' \mid \|z' - z\| \leq \epsilon'\}$ for all $\lambda \in [0, \rho(z)/2]$. Hence, for all

* Since the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is bounded, $\rho(z)$ is well-defined.

$z' \in \{z' \mid \|z' - z\| \leq \epsilon'\}$, it must be true that if $z'' \in A(z')$, then $z'' = z' + \lambda' h(z')$ and $\lambda' \geq \rho(z)/2$. This follows from the fact that $\rho(z)$ is also the first strictly positive value of λ for which $\theta(\lambda, z) = 0$.

Now, since $\langle \nabla f^0(\cdot), h(\cdot) \rangle$ is continuous, there exists an $\epsilon'' > 0$ such that for all $z' \in \{z' \mid \|z' - z\| \leq \epsilon''\}$,

$$30 \quad \langle \nabla f^0(z'), h(z') \rangle \leq \frac{1}{2} \langle \nabla f^0(z), h(z) \rangle = \gamma(z) < 0.$$

Let $\epsilon(z) = \min\{\epsilon', \epsilon''\}$. Then for all $z' \in \{z' \mid \|z' - z\| \leq \epsilon(z)\}$ and for all $z'' = (z' + \lambda' h(z')) \in A(z')$, we must have

$$31 \quad \begin{aligned} f^0(z' + \lambda' h(z')) - f^0(z') &\leq \lambda' \alpha \langle \nabla f^0(z'), h(z') \rangle \\ &\leq \frac{1}{2} \rho(z) \alpha \langle \nabla f^0(z'), h(z') \rangle \leq \rho(z) \alpha \gamma(z)/2 < 0. \end{aligned}$$

Setting $\delta(z) = \rho(z) \alpha \gamma(z)/2$, we see that condition (ii) of (1.3.10) is satisfied by the map $A(\cdot)$ defined in (23) and the function $f^0(\cdot)$. ■

- 32 **Corollary.** Consider the set $Z = \{z \mid \nabla f^0(z) = 0\}$ and suppose that for every $z' \neq z''$ in Z , $f^0(z') \neq f^0(z'')$. If the set $C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$ is compact, then any infinite sequence $\{z_i\}_{i=0}^\infty$ constructed by algorithm (19) must converge to a point $\hat{z} \in Z$ (see theorem (1.3.65)).

Proof. Since $C(z_0)$ is compact, $\{z_i\}$ must have accumulation points. Next, since by construction, $\{f^0(z_i)\}_{i=0}^\infty$ is a monotonically decreasing sequence which is bounded from below, because $C(z_0)$ is compact and $f^0(\cdot)$ is continuous, we must have $f^0(z_i) \rightarrow f_*^0 > -\infty$, as $i \rightarrow \infty$. Suppose that $z' \neq z''$ are both accumulation points of $\{z_i\}$. Then, by theorem (22), z' and z'' are in Z , and, since $\{f^0(z_i)\}_{i=0}^\infty$ converges, we must have $f^0(z') = f^0(z'')$. But this contradicts our assumption that if $z' \neq z''$ are in Z , then $f^0(z') \neq f^0(z'')$. Consequently, $\{z_i\}$ can have only one accumulation point. ■

To compute a λ_i satisfying (20), we can use algorithm (33) below, which will find such a λ_i after a finite number of iterations.

- 33 **Algorithm.** Let $\theta(\cdot; z_i)$ be defined by (24), let $\bar{\theta}(\cdot; z_i)$ be defined by (25); $z_i, \alpha \in (0, \frac{1}{2})$, $\rho > 0$ to be supplied.

Step 1. Set $\mu = \rho$.

Step 2. Compute $\theta(\mu; z_i)$.

Step 3. If $\theta(\mu; z_i) = 0$, set $\lambda_i = \mu$ and stop; if $\theta(\mu; z_i) < 0$, set $\mu = \mu + \rho$ and go to step 2; if $\theta(\mu; z_i) > 0$, go to step 4.

Step 4. Compute $\bar{\theta}(\mu; z_i)$.

Step 5. If $\bar{\theta}(\mu; z_i) \leq 0$, set $\lambda_i = \mu$ and stop; else, set $a_0 = \mu - \rho$, $b_0 = \mu$, and go to step 6.

Comment. Now $\lambda_i \in [a_0, b_0]$.

Step 6. Set $j = 0$.

Step 7. Set $v_j = (a_j + b_j)/2$.

Step 8. Compute $\underline{\theta}(v_j; z_i)$, $\bar{\theta}(v_j; z_i)$.

Step 9. If $\underline{\theta}(v_j; z_i) \geq 0$ and $\bar{\theta}(v_j; z_i) \leq 0$, set $\lambda_i = v_j$ and stop; else, go to step 10.

Step 10. If $\underline{\theta}(v_j; z_i) > 0$, set $a_{j+1} = a_j$, set $b_{j+1} = v_j$, set $j = j + 1$, and go to step 7; else, set $a_{j+1} = v_j$, set $b_{j+1} = b_j$, set $j = j + 1$, and go to step 7. ■

- 34 Exercise.** Show that the convergence properties of algorithm (19) are preserved when the step length λ_i is computed not according to (20), but by means of algorithm (36) below, i.e., show that theorem (22) is also true for algorithm (35) below. Furthermore, suppose that the set $Z = \{z \mid \nabla f^0(z) = 0\}$ contains a finite number of points only (compare [G3], p. 31; see also (1.3.66)), and that the set $C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$ is compact. Show that if the sequence $\{z_i\}$, generated by algorithm (35) below is infinite, then it must converge to a point $z \in Z$. [*Hint:* Since $\lambda_i \leq \rho$ and since $\nabla f^0(z_i) \rightarrow 0$ as $i \rightarrow \infty$, $(z_{i+1} - z_i) \rightarrow 0$ as $i \rightarrow \infty$. Since all the accumulation points of $\{z_i\}$ must be in Z , $\{z_i\}$ can have only a finite number of accumulation points, and hence we can obtain a contradiction if we assume that the sequence has more than one accumulation point.] ■

A somewhat more efficient alternative to algorithm (19) chooses step size as follows:

- 35 Algorithm** (Armijo [A4]). Let $D(z)$ be an $n \times n$ positive definite matrix whose elements are continuous functions of z .

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set $C(z_0)$ (see (2)) is bounded; select an $\alpha \in (0, 1)$, a $\beta \in (0, 1)$, and a $\rho > 0$.

Comment. Here, $\alpha = \frac{1}{2}$, $\beta \in (0.5, 0.8)$ and $\rho = 1$ are recommended (see Section 6.1).

Step 1. Set $i = 0$.

Step 2. Compute $h(z_i) = -D(z_i) \nabla f^0(z_i)$.

Step 3. If $h(z_i) = 0$, stop; else, go to step 4.

Step 4. Use algorithm (36) to compute λ_i .

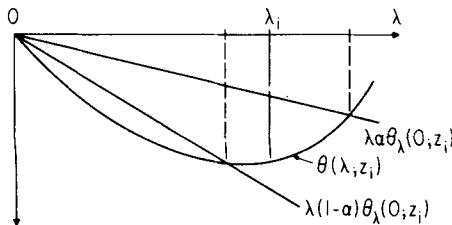
Step 5. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 2. ■

- 36 Algorithm.** Let $\bar{\theta}(\mu; z_i)$ be defined by (25); z_i , $\alpha \in (0, 1)$, $\beta \in (0, 1)$, $\rho > 0$ to be supplied.

Step 1. Set $\mu = \rho$.

Step 2. Compute $\bar{\theta}(\mu; z_i)$.

Step 3. If $\bar{\theta}(\mu; z_i) \leq 0$, set $\lambda_i = \mu$ and stop; else, set $\mu = \beta\mu$ and go to step 2. ■



Construction of λ_i according to algorithm (36): $\theta_\lambda(0; z_i) = \langle \nabla f^0(z_i), h(z_i) \rangle$.

The algorithms (16), (19) and (35) each contain a number of parameters. As we shall see in Section 6.1, there is good reason to believe that a choice of $\rho = 1$ and α close to $\frac{1}{2}$ in algorithms (19) and (36) should lead to a good rate of convergence. Note, however, that as α approaches $\frac{1}{2}$, one is bound to spend more and more time in the subprocedure (33), in the case of algorithm (19), and hence one may decide on $\alpha = 0.4$ as a reasonable choice. The choice of $\rho = 1$ is also probably good for (14). However, it may require some experimentation to arrive at a good choice of ϵ_0 in (16). One could always try to incorporate a heuristic into this algorithm which will decrease ϵ_0 whenever one spends too much time halving ϵ in step 7. Finally, the reader should realize that the step size algorithms (14), (33) and (36) can also be modified in various ways.

The algorithms (16), (19) and (35) also permit us to choose the matrix-valued function $D(\cdot)$, which also has an effect on rate of convergence. This subject will be discussed in some detail in Chapter 6; here we shall merely state a few of the most frequently used functions $D(\cdot)$.

One of the earliest algorithms to be used for function minimization is that of *steepest descent*. It is a conceptual algorithm of the form (3) with $D(\cdot) = I$, the identity matrix, and was first introduced by Cauchy [C2]. More recently, it was reintroduced and examined by Curry [C7]. It gives rise to the following class of linearly converging, “first-order” minimization algorithms (see Chapter 6 for a discussion of rate of convergence and ill conditioning):

- 37 **Algorithm.** Minimization of a continuously differentiable function $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$; “first-order” gradient methods.

Step 0. Select a step size procedure \mathcal{P} from (33) and (36), and fix its parameters.*

Step 1. Select a $z_0 \in \mathbb{R}^n$ such that the set defined by (2) is bounded.

Step 2. Set $i = 0$.

* First check whether $\lambda_i = \lambda_{i-1}$ is admissible. It is possible that the step size λ_i may become a constant for all i larger than some k . See theorem (6.1.53).

Step 3. Compute $\nabla f^0(z_i)$.

Step 4. If $\nabla f^0(z_i) = 0$, stop; else, go to step 5.

Step 5. Compute a λ_i by means of \mathcal{P} , with $h(z_i) = -\nabla f^0(z_i)$.

Step 6. Set $z_{i+1} = z_i - \lambda_i \nabla f^0(z_i)$, set $i = i + 1$, and go to step 3. ■

The second class of very important algorithms for the minimization of a function is derived from the Newton-Raphson method. (For an excellent presentation of the Newton-Raphson method in Banach spaces see [A2] or [K2].) Suppose that the function $f^0(\cdot)$ is twice continuously differentiable and that its Hessian, $\partial^2 f^0(z)/\partial z^2$, is nonsingular for all z in a sufficiently large set. Suppose that we are at a point z_i in this set. Expanding $f^0(\cdot)$ about z_i , we obtain the following quadratic approximation to $f^0(z)$:

$$f^0(z) \doteq f^0(z_i) + \langle \nabla f^0(z_i), z - z_i \rangle + \frac{1}{2} \left\langle z - z_i, \frac{\partial^2 f^0(z_i)}{\partial z^2} (z - z_i) \right\rangle.$$

Setting the gradient of the right-hand side above equal to zero, we obtain the following expression for z_{i+1} :

$$38 \quad z_{i+1} = z_i - \left(\frac{\partial^2 f^0(z_i)}{\partial z^2} \right)^{-1} \nabla f^0(z_i).$$

The function defined by the right-hand side of (38) is called the *Newton-Raphson iteration function*, and formula (38) gives the exact minimum of $f^0(\cdot)$ whenever $f^0(\cdot)$ is a positive, definite quadratic form. (See [C1] for conditions which ensure that a quadratic form has a minimum. Obviously, a quadratic function of the form $f^0(z) = \langle d, z \rangle + \langle z, Qz \rangle$ in which the matrix Q is not positive semidefinite assumes a minimum value of $-\infty$ and we then say that it has no (finite) minimum.) We shall now summarize our above discussion in a formal statement of the Newton-Raphson method.

- 39 **Algorithm** (Newton-Raphson). Minimization of a twice continuously differentiable function $f^0(\cdot)$ with an invertible Hessian matrix.

Step 0. Select a z_0 in \mathbb{R}^n .

Step 1. Set $i = 0$.

Step 2. Compute $\nabla f^0(z_i)$.

Step 3. If $\nabla f^0(z_i) = 0$, stop; else, compute z_{i+1} according to (38), set $i = i + 1$, and go to step 2. ■

- 40 **Exercise.** Use the continuity of the Newton-Raphson iteration function (38) to show that if the sequence $\{z_i\}$ constructed by (39) converges to a point z , then $\nabla f^0(z) = 0$. ■

The rate of convergence properties of the Newton-Raphson method are discussed to some extent in Sections 6.1 and 6.2. The size of the region of convergence of the Newton-Raphson method can be enlarged by means

of the following modification, due to Goldstein [G3], which results in the same rate of convergence as the original method (39). Since the algorithm below is of the form (19), it should only be used when the Hessian,

$$41 \quad H(z) = \frac{\partial^2 f^0(z)}{\partial z^2},$$

is positive definite (i.e., $f^0(\cdot)$ is convex in a suitably large region).

- 42 **Algorithm** (quasi-Newton, Goldstein [G3]). Minimization of a twice continuously differentiable function $f^0(\cdot)$ with a positive definite Hessian.

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set defined by (2) is bounded and the Hessian defined by (41) is positive definite everywhere in this set. Select an $\alpha \in (0, \frac{1}{2})$.

Step 1. Set $i = 0$.

Step 2. Compute $\nabla f^0(z_i)$.

Step 3. If $\nabla f^0(z_i) = 0$, stop; else, compute $H(z_i)$ and go to step 4.

Step 4. Set $h(z_i) = -H(z_i)^{-1} \nabla f^0(z_i)$.

Step 5. Set $\rho = 1$ in (33) and use this subprocedure to compute λ_i .

Step 6. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 2. ■

We shall discuss modifications of the Newton-Raphson method which apply to nonconvex functions and which also enlarge the region of convergence of the method (39) in the next section. In the meantime, the reader should attempt the following exercises:

- 43 **Exercise.** Show that under the assumptions stated on $f^0(\cdot)$, algorithm (42) will either stop after a finite number of iterations at a point z_k which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$, or else it will construct an infinite sequence which converges to a point \hat{z} which minimizes $f^0(z)$ over \mathbb{R}^n . ■

- 44 **Exercise.** Construct quasi-Newton algorithms using the step size subprocedures (14) and (36), respectively. ■

In conclusion, we should point out that we could have started out with a somewhat more general conceptual algorithm than (3), as follows: Let $h(\cdot, \cdot)$ be a continuous function from $\mathbb{R}^1 \times \mathbb{R}^n$ into \mathbb{R}^n such that for every $z \in \mathbb{R}^n$ satisfying $\nabla f^0(z) \neq 0$,

$$45 \quad \langle \nabla f^0(z), h(0, z) \rangle < 0.$$

Then we can construct the following procedure for minimizing a continuously differentiable function $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$:

46 Algorithm.

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set defined by (2) is bounded.

Step 1. Set $i = 0$.

Step 2. Compute $\nabla f^0(z_i)$.

Step 3. If $\nabla f^0(z_i) = 0$, stop; else, compute $\lambda_i > 0$ such that

$$47 \quad f^0(z_i + h(\lambda_i, z_i)) = \min\{f^0(z_i + h(\lambda, z_i)) \mid \lambda \geq 0\}$$

and go to step 4.

Step 4. Set $z_{i+1} = z_i + h(\lambda_i, z_i)$, set $i = i + 1$, and go to step 2. ■

48 Exercise. Show that theorem (6) is valid for algorithm (46). ■

There are hardly any algorithms described in the literature which relate to the form (46). One of the few is an algorithm due to Mayne [M5].

2.2 Reduction of Derivative Calculations

Consider again the problem

$$1 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\},$$

where $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is at least once continuously differentiable. A highly precise calculation of the derivatives of the function $f^0(\cdot)$ (as well as of the values of $f^0(\cdot)$) may be quite costly in terms of computer time, and one may therefore wish to avoid it for as long as possible in an iterative process for solving (1). There are basically two types of algorithms for solving (1) which avoid or reduce the calculation of derivatives of $f^0(\cdot)$. The first type derives from methods such as steepest descent or Newton-Raphson, and approximates derivatives with finite differences, the precision of the approximation being progressively increased as one approaches a solution of (1). The second type is conceptually independent of derivative calculations. We shall now give a few examples of algorithms which avoid either partly or completely the calculation of derivatives of $f^0(\cdot)$.

2 Algorithm (modified steepest descent, Polak [P3]). Let $\epsilon_0 > 0$, $\alpha \in (0, \frac{1}{2})$, $\alpha' > 0$, and $\beta > 0$.

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set

$$3 \quad \{z \mid f^0(z) \leq f^0(z_0)\}$$

is bounded. Select an $\epsilon_0 > 0$, an $\alpha \in (0, \frac{1}{2})$, an $\alpha' > 0$, and a $\beta > 0$.

Comment. Try $\alpha = 0.4$, $\alpha' \in [10^{-2}, 10^{-3}]$, $\epsilon_0 \in [10^{-2}, 10^{-3}]$, $\beta \in [5, 10]$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.

Step 3. Compute the vector $h(\epsilon, z_i) \in \mathbb{R}^n$ whose j th component, $h^j(\epsilon, z_i)$ is defined by

$$4 \quad h^j(\epsilon, z_i) = -\frac{1}{\epsilon} [f^0(z_i + \epsilon e_j) - f^0(z_i)], \quad j = 1, 2, \dots, n,$$

where e_j is the j th column of the $n \times n$ identity matrix, i.e., $e_1 = (1, 0, \dots, 0)$, $e_2 = (0, 1, 0, \dots, 0)$, etc.

Step 4. Compute $f^0(z_i + \beta \epsilon h(\epsilon, z_i)) - f^0(z_i) \triangleq \Delta(\epsilon, z_i)$.

Step 5. If $\Delta(\epsilon, z_i) < 0$, compute a $\bar{\lambda}$ such that

$$5 \quad \bar{\lambda}(1 - \alpha) \Delta(\epsilon, z_i)/\beta \epsilon \leq \theta(\bar{\lambda}, z_i, h(\epsilon, z_i)) \leq \lambda \alpha \Delta(\epsilon, z_i)/\beta \epsilon, *$$

where

$$6 \quad \theta(\bar{\lambda}, z_i, h(\epsilon, z_i)) = f^0(z_i + \bar{\lambda} h(\epsilon, z_i)) - f^0(z_i),$$

and go to step 6; else, set $\epsilon = \epsilon/2$ and go to step 3.

Comment. Algorithm (1.33) is easily adapted to calculate a $\bar{\lambda}$ which satisfies (5).

Step 6. If $\theta(\bar{\lambda}, z_i, h(\epsilon, z_i)) \leq -\alpha' \epsilon$, set $z_{i+1} = z_i + \bar{\lambda} h(\epsilon, z_i)$, set $i = i + 1$, and go to step 2; else, set $\epsilon = \epsilon/2$ and go to step 3. ■

7 **Exercise.** Use theorem (1.3.27) to show that if $\{z_i\}$ is a sequence constructed by algorithm (2), then either $\{z_i\}$ is finite (i.e., the algorithm jams up after a finite number of iterations at a point z_k , cycling between steps 3 and 5 or between steps 3 and 6 while continuing to divide ϵ by 2) and $\nabla f^0(z_k) = 0$, or else the sequence $\{z_i\}$ is infinite and every accumulation point z' of $\{z_i\}$ satisfies $\nabla f^0(z') = 0$. ■

8 **Exercise.** Construct an algorithm of the form (2), but using a step size calculated by means of a simple modification of algorithm (1.36). ■

9 **Exercise.** Suppose that the function $f^0(\cdot)$ is convex. Construct an algorithm of the form (2), but using a step size calculated by means of algorithm (1.14). ■

10 **Algorithm** (modified quasi-Newton method, Polak [P3], compare [G2]). Suppose that $f^0(\cdot)$ is strictly convex and twice continuously differentiable.

* An adaptation of step size subprocedure (1.36) can be used instead of (5) and may require less computation time per iteration.

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set

$$11 \quad \{z \mid f^0(z) \leq f^0(z_0)\}$$

is bounded. Select an $\epsilon_0 > 0$, an $\alpha \in (0, \frac{1}{2})$, and an $\alpha' > 0$.

Comment. Try $\epsilon_0 \in [10^{-2}, 10^{-3}]$, $\alpha = 0.4$, and $\alpha' \in [10^{-2}, 10^{-3}]$.

Step 1. Set $i = 0$.

Step 2. Set $\epsilon = \epsilon_0$.*

Step 3. Compute the $n \times n$ matrix $H(\epsilon, z_i)$ whose j th column is

$$12 \quad \frac{1}{\epsilon} [\nabla f^0(z_i + \epsilon e_j) - \nabla f^0(z_i)], \quad j = 1, 2, \dots, n,$$

where e_j is the j th column of the $n \times n$ identity matrix, i.e., $e_1 = (1, 0, \dots, 0)$, $e_2 = (0, 1, 0, \dots, 0)$, etc.

Step 4. If $H(\epsilon, z_i)^{-1}$ exists and $\langle \nabla f^0(z_i), H(\epsilon, z_i)^{-1} \nabla f^0(z_i) \rangle > 0$, set $h(\epsilon, z_i) = -H(\epsilon, z_i)^{-1} \nabla f^0(z_i)$ and go to step 5; else, set $\epsilon = \epsilon/2$ and go to step 3.

Step 5. Compute a $\bar{\lambda}$ such that

$$13 \quad \bar{\lambda}(1 - \alpha) \langle \nabla f^0(z_i), h(\epsilon, z_i) \rangle \leq \theta(\bar{\lambda}, z_i, h(\epsilon, z_i)) \leq \bar{\lambda}\alpha \langle \nabla f^0(z_i), h(\epsilon, z_i) \rangle,$$

where $\theta(\cdot, \cdot, \cdot)$ is defined as in (6). Set $\bar{\lambda} = 1$ if possible.

Comment. Use an adaptation of algorithm (1.33).

Step 6. If $\theta(\bar{\lambda}, z_i, h(\epsilon, z_i)) \leq -\alpha'\epsilon$, set $z_{i+1} = z_i + \bar{\lambda}h(\epsilon, z_i)$, set $i = i + 1$, and go to step 2; else, set $\epsilon = \epsilon/2$ and go to step 3. ■

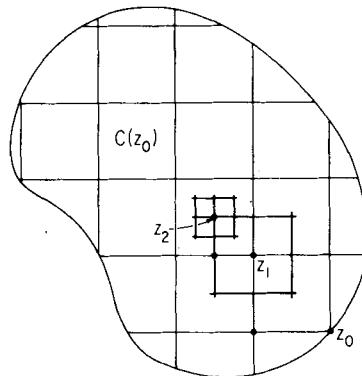
- 14 **Exercise.** Use theorem (1.3.27) to show that if $\{z_i\}$ is a sequence constructed by algorithm (10), then either $\{z_i\}$ is finite, i.e., the algorithm jams up at a point z_k , cycling between steps 3 and 6 while continuing to divide ϵ by 2, and $\nabla f^0(z_k) = 0$, or else $\{z_i\}$ is infinite and converges to a point \hat{z} such that $\nabla f^0(\hat{z}) = 0$, i.e., \hat{z} minimizes $f^0(z)$ over $z \in \mathbb{R}^n$ (see (1.3.65)). ■

We now present an algorithm for unconstrained minimization which does not require any derivative evaluations and which is not an obvious adaptation of an algorithm that does require derivatives. This algorithm is called the method of local variations in the Russian literature, and seems to have been known in one form or another for quite a long time. In particular, it is quite obviously related to the Gauss-Seidel algorithm (see [E2]). Recently, it has been described in [B3] and [C3]. It is particularly effective when the function $f^0(\cdot)$ is of the form

$$15 \quad f^0(z) = \sum_{i=1}^n f_i^0(z^i).$$

* To cause the algorithm to stop at stationary points and to guarantee superlinear convergence, replace the instruction in step 2 by “If $\nabla f^0(z_i) = 0$, stop; else, set $\epsilon = \min\{\epsilon_0, \|\nabla f^0(z_i)\|\}$ and go to step 3”. See Cohen [C4].

We shall need the following notation: For $i = 1, 2, \dots, n$, let e_i be the i th column of the $n \times n$ identity matrix, i.e., $e_1 = (1, 0, \dots, 0)$, $e_2 = (0, 1, 0, \dots, 0)$, etc., and let $d_1 = e_1$, $d_2 = -e_1$, $d_3 = e_2$, $d_4 = -e_2, \dots, d_{2n-1} = e_n$, $d_{2n} = -e_n$.



Method of local variations.

16 Algorithm (method of local variations, Banitchouk *et al.* [B2]).

Step 0. Select a $z_0 \in \mathbb{R}^n$ such that the set

$$17 \quad C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$$

is bounded. Select a $\rho_0 > 0$.

Step 1. Set $i = 0$, set $z = z_0$ and compute $f^0(z)$.

Step 2. Set $\rho = \rho_i$.

Step 3. Set $j = 1$.

Step 4. Compute $f^0(z + \rho d_j)$.

Step 5. If $f^0(z + \rho d_j) < f^0(z)$, set $z = z + \rho d_j$ and go to step 3; else, go to step 6.

Step 6. If $j < 2n$, set $j = j + 1$ and go to step 4; else, go to step 7.

Step 7. Set $z_{i+1} = z$, set $\rho_{i+1} = \rho/2$, set $i = i + 1$, and go to step 2. ■

18 Theorem. If $\{z_i\}_{i=0}^\infty$ is a sequence constructed by algorithm (16), then every accumulation point z' of $\{z_i\}$ satisfies $\nabla f^0(z') = 0$. (By assumption, following (1), $f^0(\cdot)$ is at least once continuously differentiable.)

Proof. Since the set $C(z_0)$ is compact, starting with $z = z_i$ and $\rho = \rho_i$, algorithm (16) can construct only a finite number of intermediate points z in the loop between steps 3 and 6 before it is forced to proceed to step 7. Hence algorithm (16) cannot jam up at a point z_i and it must construct an infinite sequence of vectors $\{z_i\}$ and a corresponding, strictly monotonically

decreasing sequence of scalars $\{\rho_i\}$. Also, since the set $C(z_0)$ is compact, $\{z_i\}$ must have accumulation points. Hence, suppose that $z_i \rightarrow z'$ as $i \rightarrow \infty$, with $i \in K \subset \{0, 1, 2, 3, \dots\}$. By construction, we must have, for $i = 1, 2, \dots$,

$$19 \quad f^0(z_i + \rho_{i-1} d_j) \geq f^0(z_i) \quad \text{for all } j \in \{1, 2, \dots, 2n\}.$$

Hence, by the mean-value theorem (B.1.1), we obtain, with $t_i^j \in [0, 1]$,

$$20 \quad f^0(z_i + \rho_{i-1} d_j) - f^0(z_i) = \rho_{i-1} \langle \nabla f^0(z_i + t_i^j \rho_{i-1} d_j), d_j \rangle \geq 0,$$

for all $j \in \{1, 2, \dots, 2n\}$ and for all $i \in K$. Since we have assumed $\nabla f^0(\cdot)$ to be continuous and since for $i \in K$, $z_i \rightarrow z'$ and $\rho_i \rightarrow 0$ as $i \rightarrow \infty$, we obtain from (20) that

$$21 \quad \langle \nabla f^0(z'), d_j \rangle \geq 0 \quad \text{for } j = 1, 2, \dots, 2n.$$

It now follows directly from the definition of the d_j that $\nabla f^0(z') = 0$. ■

- 22 **Corollary.** Suppose that the set $Z = \{z \in C(z_0) \mid \nabla f^0(z) = 0\}$ consists of a finite number of points only, and that every $z \in Z$ is either a local minimizer or local maximizer of $f^0(\cdot)$. Then the sequence $\{z_i\}$ constructed by algorithm (16) converges to a point \hat{z} such that $\nabla f^0(\hat{z}) = 0$. ■
- 23 **Exercise.** Prove corollary (22). [Hint: Note that $(z_{i+1} - z_i) \rightarrow 0$ as $i \rightarrow \infty$ (compare (1.3.66)).] ■

There are a number of other methods for computing the minima or stationary points of a function without derivative calculations. For an excellent survey of such methods based on heuristic considerations, the reader is referred to the paper by Powell [P8], which also contains a large bibliography on the subject of unconstrained optimization algorithms. For a discussion of the Gauss-Seidel and related methods, which can be treated rigorously, the reader is referred to the Ph.D. dissertation of Elkin [E2]. In conclusion, we wish to point out that the ideas found in the first algorithm presented in this section can also be used to modify the method of steepest descent in solving minimization problems involving integrals and differential equations; in particular, so as to adjust adaptively the integration step size. These problems will be discussed in Section 5.

2.3 Conjugate Gradient Methods in \mathbb{R}^n

While the rate of convergence of quasi-Newton methods is quite rapid, they do require the calculation of second partial derivatives and the inversion of a possibly large Hessian matrix at each iteration. These two sets of calculations obviously have a tendency to increase the time required for the construc-

tion of a minimizing sequence (i.e., a sequence $\{z_i\}$ which converges to a \hat{z} such that $f^0(\hat{z}) \leq f^0(z)$ for all $z \in \mathbb{R}^n$). We shall now describe a class of methods for the minimization of a differentiable function $f^0(\cdot)$. Under a slightly stronger assumption than that $f^0(\cdot)$ is strictly convex, we shall show that these methods can be used to compute the minimum of $f^0(z)$ for $z \in \mathbb{R}^n$. In Sections 6.3 and 6.4 we shall show that at least for some of these methods the rate of convergence is almost as good as that of the quasi-Newton methods. The conjugate gradient methods we are about to describe are frequently used in practice even when the function $f^0(\cdot)$ that one wishes to minimize is not convex, and there is reason to believe that such a use leads to the computation of local minima. However, at least so far, there is nothing we can say mathematically about the convergence properties of conjugate gradient methods when applied to the minimization of a non-convex function.

Conjugate gradient methods were first introduced by Hestenes and Stiefel [H1, H2] as a means for solving systems of linear equations. They have the interesting property that they require at most n iterations to solve the problem

$$1 \quad \min\{\langle d, z \rangle + \langle z, Hz \rangle \mid z \in \mathbb{R}^n\},$$

where H is an $n \times n$ positive definite matrix and $d \in \mathbb{R}^n$. We shall now give a heuristic reason to explain why one may expect conjugate gradient methods to have a good rate of convergence when applied to a strictly convex function. Thus, consider the problem $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$ and suppose that $f^0(\cdot)$ is strictly convex and twice continuously differentiable. Let \hat{z} be the optimal solution for this problem; then $\nabla f^0(\hat{z}) = 0$ and for all z close to \hat{z} we get the approximate relationship

$$2 \quad \begin{aligned} f^0(z) - f^0(\hat{z}) &\doteq \frac{1}{2}\langle z - \hat{z}, H(\hat{z})(z - \hat{z}) \rangle \\ &= -\langle H(\hat{z})\hat{z}, z \rangle + \frac{1}{2}\langle z, H(\hat{z})z \rangle + \frac{1}{2}\langle \hat{z}, H(\hat{z})\hat{z} \rangle, \end{aligned}$$

where $H(\hat{z}) = \partial^2 f^0(\hat{z})/\partial z^2$. Since by a simple extension of (B.2.6) the matrix $H(\hat{z})$ is positive definite as a result of $f^0(\cdot)$ being strictly convex, we see that in a neighborhood of \hat{z} , the problem $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$ approximately assumes the form of problem (1). Hence, it is not unreasonable to expect that any method, which is convergent when applied to the minimization of a strictly convex function and which solves problem (1) in a small number of steps, will also converge rapidly towards the end in the minimization of a strictly convex function.

In this section we shall consider conjugate gradient algorithms for solving the problem

$$3 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\},$$

under the assumption that $f^0(\cdot)$ is a strictly convex, twice continuously differentiable function such that for all $z \in \{z \mid f^0(z) \leq f^0(z_0)\}$ (where z_0 is our first guess at a minimum of $f^0(\cdot)$) and all $y \in \mathbb{R}^n$,

$$4 \quad 0 \leq m \|y\|^2 \leq \langle y, H(z)y \rangle \leq M \|y\|^2, \quad 0 < m \leq M < \infty,^*$$

where the bounds m and M may, but need not be, the smallest and largest eigenvalues, respectively, of the Hessian matrix $H(z) = \partial^2 f^0(z)/\partial z^2$.

All conjugate gradient algorithms are of the form given below. For every $z \in \mathbb{R}^n$, let $F(z)$ be defined as

$$5 \quad F(z) = \{h \in \mathbb{R}^n \mid \langle \nabla f^0(z), h \rangle < 0\}.$$

6 Algorithm (conjugate gradient method prototype).

Step 0. Select a $z_0 \in \mathbb{R}^n$ and set $i = 0$.

Step 1. Compute $\nabla f^0(z_i)$.

Step 2. If $\nabla f^0(z_i) = 0$, stop; else, compute an $h_i \in F(z_i)$ and go to step 3.

Step 3. Compute a $\lambda_i > 0$ such that

$$7 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 4. Set $z_{i+1} = z_i + \lambda_i h_i$, set $i = i + 1$, and go to step 1. ■

Note that the minimization operation required in step 3 makes all conjugate gradient methods purely conceptual. Normally, some form of approximation must be introduced in the computation of λ_i . The various approximations (such as result from polynomial expansions, or the use of the Golden section search (1.14)) which are currently used in practice upset the relationships which ensure the convergence to the conceptual form of the algorithm. As a result, in practice, one is forced to reinitialize the algorithms every so many iterations. We shall return to this problem later.

Before we give a detailed description of a few conjugate gradient methods, which differ only in the specific manner set up for calculating vectors $h_i \in F(z_i)$ in step 2 of (6), let us establish one sufficient condition for algorithm (6) to be convergent.

8 Theorem.

Consider problem (3), with the assumptions stated, and algorithm (6). Suppose that in step 2 of (6), the vector h_i is always chosen so that for a fixed $\rho > 0$,

$$9 \quad -\langle \nabla f^0(z_i), h_i \rangle \geq \rho \| \nabla f^0(z_i) \| \| h_i \|.$$

* Note that since $m > 0$, it follows from (B.2.8) that the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is compact. Hence, since the Hessian $H(\cdot)$ is continuous, the bound M always exists, and consequently its postulation is redundant.

Then, either algorithm (6) constructs a finite sequence $\{z_i\}$, whose last element, z_k , minimizes $f^0(z)$ over $z \in \mathbb{R}^n$, or else the algorithm constructs an infinite sequence $\{z_i\}$ which converges to a point \hat{z} which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$.

Proof. The first part of the theorem is trivial, since the algorithm stops at a point z_k if and only if $\nabla f^0(z_k) = 0$, and, by assumption, $f^0(\cdot)$ is strictly convex, which ensures that such a point solves (3).

To prove the second part of the theorem, we shall show that the maps $c(\cdot) = f^0(\cdot)$ and $A(\cdot)$ defined as below, satisfy assumptions (i) and (ii) of theorem (1.3.10).^{*} Thus, for every $z \in \mathbb{R}^n$, let $A(z)$ be defined by

$$10 \quad A(z) = \{y = z + \lambda(z, h)h \mid h \text{ satisfies (9), } \lambda(z, h) \text{ is determined by (7)}\}.$$

(We use (7) and (9) in (10) with the subscripts on z and h deleted, of course.) That assumption (i) is satisfied is clear. Hence we only need to establish that assumption (ii) of (1.3.10) is satisfied by the maps $f^0(\cdot)$ and $A(\cdot)$. Now, since $f^0(\cdot)$ is twice continuously differentiable, we have, by the Taylor expansion formula (B.1.12), that

$$11 \quad f^0(z + \lambda h) - f^0(z) = \lambda \langle \nabla f^0(z), h \rangle + \lambda^2 \int_0^1 (1-t) \langle h, H(z + t\lambda h) h \rangle dt,$$

where $H(z) = \partial^2 f^0(z) / \partial z^2$. Since, by assumption, (4) is satisfied and (9) holds for every h such that $(z + \bar{\lambda}h) \in A(z)$ (for some $\bar{\lambda} > 0$), we must have, for such an h and any $\lambda \geq 0$,

$$12 \quad f^0(z + \lambda h) - f^0(z) \leq -\lambda \rho \| \nabla f^0(z) \| \| h \| + \frac{1}{2} \lambda^2 M \| h \|^2.^\dagger$$

Now suppose that $\nabla f^0(z) \neq 0$. Then, for all $y \in A(z)$, we must have (by minimizing the right-hand side of (12) over λ) that

$$13 \quad f^0(y) - f^0(z) \leq -\frac{\rho^2}{2M} \| \nabla f^0(z) \|^2 < 0.$$

Since $\nabla f^0(\cdot)$ is continuous, there must exist an $\epsilon(z) > 0$ such that for all z' such that $\| z' - z \| \leq \epsilon(z)$,

$$14 \quad \| \nabla f^0(z') \|^2 \geq \frac{1}{2} \| \nabla f^0(z) \|^2.$$

Next, since (13) must also be true for all $y' \in A(z')$, with y' and z' taking

* For this purpose we set $T = \mathbb{R}^n$ and we define $z \in \mathbb{R}^n$ to be desirable if $\nabla f^0(z) = 0$.

† We assume, of course, that both z and $z + \lambda h$ are in the set $\{z \mid f^0(z) \leq f^0(z_0)\}$, where z_0 is some given starting point.

the place of y and z , respectively, we conclude that for all $z' \in \{z' \mid \|z - z'\| \leq \epsilon(z)\}$ and for all $y' \in A(z')$,

$$15 \quad f^0(y') - f^0(z') \leq -\frac{\rho^2}{4M} \|\nabla f^0(z)\|^2 = \delta(z) < 0,$$

and hence, (ii) of theorem (1.3.10) is satisfied. Thus, any accumulation point \hat{z} of the sequence $\{z_i\}$ constructed by (6) must satisfy $\nabla f^0(\hat{z}) = 0$, and hence minimize $f^0(z)$ over $z \in \mathbb{R}^n$. Now, since $f^0(\cdot)$ satisfies (4), the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is compact (see (B.2.8)) and hence the sequence $\{z_i\}$ must contain accumulation points, each of which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$. But $f^0(\cdot)$ is strictly convex, and hence there can be only one such minimizing point \hat{z} . Consequently, the sequence $\{z_i\}$ has only one limit point, \hat{z} , and it converges to this limit point which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$. ■

- 16 **Exercise.** Suppose that λ_i in step 3 of (6) is not computed to satisfy (7), but is instead computed by means of algorithm (1.33). Show that the conclusion of theorem (8) still remains valid. ■

Yet another sufficient condition for algorithm (6) to compute the solution of problem (3) was pointed out to the author by Zoutendijk. This condition (stated below) is of an entirely different nature from all the convergence theorems we have considered so far and we leave its proof as an exercise for the reader.

- 17 **Theorem.** (Zoutendijk [Z6a]). Consider problem (3), with the conditions stated, and algorithm (6). Suppose that algorithm (6) constructs an infinite sequence of points $\{z_i\}$ and an infinite sequence of direction vectors $\{h_i\}$ such that

$$18 \quad \langle \nabla f^0(z_i), h_i \rangle \leq t_i \|\nabla f^0(z_i)\| \|h_i\|, \quad i = 0, 1, 2, \dots,$$

with the $t_i < 0$ such that

$$19 \quad \sum_{i=0}^k t_i^2 \rightarrow \infty \quad \text{as } k \rightarrow \infty.$$

Then the sequence $\{z_i\}$ converges to a point \hat{z} such that $f^0(\hat{z}) \leq f^0(z)$ for all $z \in \mathbb{R}^n$. ■

- 20 **Exercise.** Prove theorem (17). ■

This completes our preliminaries, and we are now ready to discuss the specific rules for finding an $h_i \in F(z_i)$, which occur in various conjugate gradient algorithms. We begin our discussion with a description of a biorthogonalization process. (We follow Hestenes [H2] in our presentation.) Suppose that

we are given an $n \times n$ symmetric, positive definite matrix and that we wish to construct two sequences in \mathbb{R}^n , g_0, g_1, \dots , and h_0, h_1, \dots , such that

$$21 \quad \langle g_i, g_j \rangle = 0 \quad \text{for all } i \neq j,$$

and

$$22 \quad \langle h_i, Hh_j \rangle = 0 \quad \text{for all } i \neq j.$$

We can do this by means of the Gramm-Schmidt orthogonalization method, as follows, using a “bootstrap” approach.

Let $g_0 \in \mathbb{R}^n$ be arbitrary. Set $h_0 = g_0$. Now set

$$23 \quad g_1 = g_0 - \lambda_0 Hh_0, \quad \text{with } \lambda_0 = \frac{\langle g_0, g_0 \rangle}{\langle g_0, Hh_0 \rangle},$$

which ensures that $\langle g_0, g_1 \rangle = 0$. Next, set

$$24 \quad h_1 = g_1 + \gamma_0 h_0, \quad \text{with } \gamma_0 = -\frac{\langle Hh_0, g_1 \rangle}{\langle Hh_0, h_0 \rangle},$$

which ensures that $\langle h_0, Hh_1 \rangle = 0$. To continue, set

$$25 \quad g_2 = \alpha g_0 + g_1 - \lambda_1 Hh_1; \quad h_2 = g_2 + \gamma_1 h_1 + \beta h_0,$$

where λ_1, α are chosen so as to make $\langle g_0, g_2 \rangle = \langle g_1, g_2 \rangle = 0$, and γ_1, β are chosen so as to make $\langle h_0, Hh_2 \rangle = \langle h_1, Hh_2 \rangle = 0$. Obviously, this process can be continued until for some $m \leq n$ we obtain $g_m = h_m = 0$. The fact that the construction must stop in the manner indicated follows from the theorem below, which also shows that, rather remarkably, the coefficients α, β , etc. are all zero.

26 **Theorem.** Let H be a symmetric, positive definite $n \times n$ matrix and let $g_0 \in \mathbb{R}^n$ be arbitrary. Suppose that for $i = 0, 1, 2, \dots$,

$$27 \quad g_{i+1} = g_i - \lambda_i Hh_i; \quad h_{i+1} = g_{i+1} + \gamma_i h_i, \quad \text{with } h_0 = g_0,$$

where λ_i, γ_i are chosen so that $\langle g_{i+1}, g_i \rangle = 0, \langle h_{i+1}, Hh_i \rangle = 0$, i.e.,

$$28 \quad \lambda_i = \frac{\langle g_i, g_i \rangle}{\langle g_i, Hh_i \rangle}, \quad \gamma_i = -\frac{\langle Hh_i, g_{i+1} \rangle}{\langle Hh_i, h_i \rangle},$$

whenever the denominators are not zero, and $\lambda_i = 0, \gamma_i = 0$, otherwise. Then, for $i, j = 0, 1, 2, \dots$,

$$29 \quad \langle g_i, g_j \rangle = \delta_{ij} \|g_i\|^2; \quad \langle h_i, Hh_j \rangle = \delta_{ij} \langle h_i, Hh_i \rangle,$$

where δ_{ij} is the Kronecker symbol, and $g_i = h_i = 0$ for all $i > m$, with $m \leq n - 1$.

Proof. Suppose that the construction defined by (27) and (28) results in nonzero vectors g_0, g_1, \dots, g_m and h_0, h_1, \dots, h_m after m iterations. Suppose that g_{m+1} as constructed by (27) is zero; then we obtain from (27) and (28) that $h_{m+1} = 0$, and hence $g_{m+j} = h_{m+j} = 0$ for all $j \geq 1$. Now suppose that $h_{m+1} = 0$; we shall show that this implies that $g_{m+1} = 0$. For this purpose we must show that $\langle h_m, g_{m+1} \rangle = 0$. By construction, since $h_0 = g_0$, $\langle h_0, g_1 \rangle = 0$. So, let us suppose that $\langle h_{j-1}, g_j \rangle = 0$ for any $j \in \{1, 2, \dots, m\}$. Then,

$$\begin{aligned} 30 \quad \langle h_j, g_{j+1} \rangle &= \langle h_j, g_j - \lambda_j H h_j \rangle \\ &= \langle h_j, g_j \rangle - \lambda_j \langle h_j, H h_j \rangle \\ &= \langle g_j + \gamma_{j-1} h_{j-1}, g_j \rangle - \lambda_j \langle H h_j, h_j \rangle \\ &= \langle g_j, g_j \rangle - \lambda_j \langle H h_j, h_j \rangle = 0, \end{aligned}$$

since $\langle g_j, H g_j \rangle = \langle h_j - \gamma_{j-1} h_{j-1}, H h_j \rangle = \langle h_j, H h_j \rangle$. It now follows by induction that $\langle h_m, g_{m+1} \rangle = 0$. Since h_m and g_{m+1} are orthogonal, and $h_{m+1} = 0$, we conclude from (27) that $0 = \|g_{m+1}\|^2 + \gamma_m \langle g_{m+1}, h_m \rangle$, i.e., that for $h_{m+1} = 0$ we must have $g_{m+1} = 0$. Our conclusion is that the construction defined by (27) and (28) will result in two sequences, g_0, g_1, \dots and h_0, h_1, \dots , of vectors such that for some $m \geq 0$, both g_i and h_i are nonzero for all $0 \leq i \leq m$, and both g_i and h_i are zero for all $i > m$.

Suppose that the integer m is such that $g_i \neq 0$ and $h_i \neq 0$ for all $0 \leq i \leq m$ and $g_i = h_i = 0$ for all $i > m$. Clearly, the relations (29) are satisfied trivially whenever $i > m$ or $j > m$. Hence we need only to consider the case $0 \leq i, j \leq m$. We give a proof by induction. By construction, $\langle g_0, g_1 \rangle = 0$ and $\langle h_0, H h_1 \rangle = 0$. Suppose that for some integer $0 \leq k < m$,

$$31 \quad \langle g_i, g_j \rangle = \langle h_i, H h_j \rangle = 0 \quad \text{for all } i \neq j, \quad 0 \leq i, j \leq k.$$

Let $i \in \{1, 2, \dots, k-1\}$. Then,

$$\begin{aligned} 32 \quad \langle g_{k+1}, g_i \rangle &= \langle g_k - \lambda_k H h_k, g_i \rangle \\ &= -\lambda_k \langle H h_k, g_i \rangle \\ &= -\lambda_k \langle H h_k, h_i - \gamma_{i-1} h_{i-1} \rangle = 0. \end{aligned}$$

Also, $\langle g_{k+1}, g_k \rangle = 0$ by the choice of λ_k , and

$$33 \quad \langle g_{k+1}, g_0 \rangle = \langle g_k - \lambda_k H h_k, g_0 \rangle = -\lambda_k \langle H h_k, g_0 \rangle = 0 \quad (g_0 = h_0).$$

Similarly, $\langle h_{k+1}, Hh_k \rangle = 0$ by the choice of γ_k , and for $i \in \{0, 1, \dots, k-1\}$,

$$\begin{aligned} 34 \quad \langle h_{k+1}, Hh_i \rangle &= \langle g_{k+1} + \gamma_k h_k, Hh_i \rangle \\ &= \langle g_{k+1}, Hh_i \rangle \\ &= \left\langle g_{k+1}, \frac{-1}{\lambda_i} (g_{i+1} - g_i) \right\rangle = 0, \end{aligned}$$

since $\lambda_i \neq 0$ for $i = 0, 1, \dots, m$. Now (31) is true for $k = 1$ and hence, (29) must be true for all $0 \leq i, j \leq m$.

The vectors g_0, g_1, \dots, g_m are orthogonal to each other, and since they are all nonzero, their total number cannot exceed n , i.e., $m \leq n$. This completes our proof. ■

35 **Corollary.** Suppose that g_0, g_1, \dots, g_m and h_0, h_1, \dots, h_m are nonzero vectors constructed according to (27) and (28). Then

$$36 \quad \langle h_i, g_k \rangle = 0 \quad \text{for all } 0 \leq i < k \leq m,$$

$$37 \quad \lambda_i = \frac{\langle g_i, g_i \rangle}{\langle g_i, Hh_i \rangle} = \frac{\langle h_i, g_i \rangle}{\langle h_i, Hh_i \rangle}, \quad i = 0, 1, 2, \dots, m,$$

$$38 \quad \gamma_i = -\frac{\langle Hh_i, g_{i+1} \rangle}{\langle Hh_i, h_i \rangle} = \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle} = \frac{\langle g_{i+1}, g_{i+1} \rangle \pm \langle g_{i+1}, g_i \rangle}{\langle g_i, g_i \rangle}, \quad i = 0, 1, \dots, m.$$

Proof. To prove (36), we note that for all $0 \leq i < k \leq m$,

39 $\langle h_i, g_k \rangle = \langle h_i, g_{k-1} - \lambda_{k-1} Hh_{k-1} \rangle = \langle h_i, g_{k-1} \rangle = \dots = \langle h_i, g_{i+1} \rangle = 0$, since the last equality in (39) was established in the proof of theorem (26); see (30).

To prove (37), we observe that

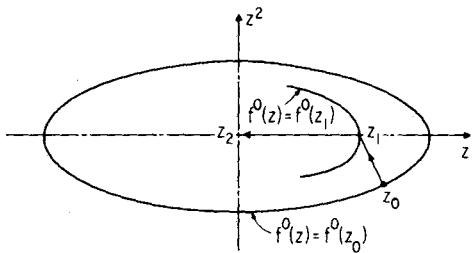
$$40 \quad \frac{\langle g_i, g_i \rangle}{\langle g_i, Hh_i \rangle} = \frac{\langle h_i - \gamma_{i-1} h_{i-1}, g_i \rangle}{\langle h_i - \gamma_{i-1} h_{i-1}, Hh_i \rangle} = \frac{\langle h_i, g_i \rangle}{\langle h_i, Hh_i \rangle}.$$

To prove (38), we proceed as follows:

$$\begin{aligned} 41 \quad -\frac{\langle Hh_i, g_{i+1} \rangle}{\langle Hh_i, h_i \rangle} &= -\frac{\langle g_{i+1} - g_i, g_{i+1} \rangle}{\langle g_{i+1} - g_i, h_i \rangle} \\ &= \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, h_i \rangle} \\ &= \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i + \gamma_{i-1} h_{i-1} \rangle} \\ &= \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle} \\ &= \frac{\langle g_{i+1}, g_{i+1} \rangle \pm \langle g_{i+1}, g_i \rangle}{\langle g_i, g_i \rangle}, \end{aligned}$$

since $\langle g_{i+1}, g_i \rangle = 0$. ■

We shall now see how the biorthogonalization process we have been discussing is utilized in conjugate gradient methods.



Minimization of quadratic function in \mathbb{R}^2 by the Fletcher-Reeves algorithm (42).

42 Algorithm (Fletcher-Reeves [F4]).*

Step 0. Select a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$ and set $g_0 = h_0 = -\nabla f^0(z_0)$.

Step 2. Compute $\lambda_i > 0$ such that

$$43 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 3. Set

$$44 \quad z_{i+1} = z_i + \lambda_i h_i.$$

Step 4. Compute $\nabla f^0(z_{i+1})$.

Step 5. If $\nabla f^0(z_{i+1}) = 0$, stop; else, set

$$45 \quad g_{i+1} = -\nabla f^0(z_{i+1}),$$

$$46 \quad h_{i+1} = g_{i+1} + \gamma_i h_i, \quad \text{with } \gamma_i = \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle},$$

set $i = i + 1$, and go to step 2. ■

Our first observation is that the choice of γ_i in (46) is exactly the same as in the biorthogonalization procedure we have discussed; see (38). Our next observation is that by (43) we must have $\langle g_{i+1}, h_i \rangle = 0$. We shall soon see that when $f^0(\cdot)$ is a quadratic function, this results in the same formula for λ_i as (37). Therefore, let us consider the quadratic case first.

47 Theorem. Suppose that $f^0(z) = \langle d, z \rangle + \frac{1}{2}\langle z, Hz \rangle$ for all $z \in \mathbb{R}^n$, where H is an $n \times n$ symmetric, positive definite matrix and $d \in \mathbb{R}^n$. Then the

* This algorithm had been proposed previously by Hestenes and Stiefel for minimizing a quadratic function, but with λ_i, γ_i defined as in (28); see [H1]. The contribution of Fletcher and Reeves lies in the idea of applying this algorithm to general convex functions and in developing alternative formulas for λ_i, γ_i , as in (43) and (46) (see (47)).

Fletcher-Reeves algorithm (42) constructs the \hat{z} which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$ in $k \leq n$ iterations.

Proof. For the case under consideration, $\nabla f^0(z_i) = Hz_i + d$. Hence, from (45) and (44),

$$48 \quad g_{i+1} = -H(z_i + \lambda_i h_i) - d = g_i - \lambda_i Hh_i.$$

The choice of λ_i defined by (43) implies that $\langle h_i, g_{i+1} \rangle = 0$. Hence, from (48), we obtain

$$49 \quad \lambda_i = \frac{\langle h_i, g_i \rangle}{\langle h_i, Hh_i \rangle},$$

which we see to be the same as (37). Comparing (46), (48) and (49) with (27), (28), (37) and (38), we conclude, making use of induction and of theorem (26), that for some $k \leq n$, the vector g_k , constructed by the Fletcher-Reeves algorithm for the quadratic function under consideration, will be zero; i.e., $\nabla f^0(z_k) = 0$, and hence z_k minimizes $f^0(z)$ over \mathbb{R}^n . ■

- 50 **Exercise.** Consider problem (3) with the assumptions stated. Show that for this problem the Fletcher-Reeves algorithm (42) constructs a sequence of points $\{z_i\}$ such that either $\{z_i\}$ is finite and its last element, z_k , satisfies $\nabla f^0(z_k) = 0$, or else $\{z_i\}$ is infinite and then it converges to the point \hat{z} which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$. [Hint: Show that in this case the Fletcher-Reeves algorithm satisfies the assumptions of theorem (17).] ■

We now present a conjugate gradient algorithm which coincides with the Fletcher-Reeves algorithm whenever it is applied to a quadratic function. However, its convergence is somewhat easier to establish and, in addition, one can obtain a bound on its rate of convergence. The rate of convergence of the algorithm below will be studied in Section 6.3.

- 51 **Algorithm** (Polak-Ribière [P5]).

Step 0. Select a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$ and set $g_0 = h_0 = -\nabla f^0(z_0)$.

Step 2. Compute $\lambda_i > 0$ such that

$$52 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 3. Set

$$53 \quad z_{i+1} = z_i + \lambda_i h_i.$$

Step 4. Compute $\nabla f^0(z_{i+1})$.

Step 5. If $\nabla f^0(z_{i+1}) = 0$, stop; else, set

$$54 \quad g_{i+1} = -\nabla f^0(z_{i+1}),$$

$$55 \quad h_{i+1} = g_{i+1} + \gamma_i h_i, \quad \text{with } \gamma_i = \frac{\langle g_{i+1} - g_i, g_{i+1} \rangle}{\langle g_i, g_i \rangle},$$

set $i = i + 1$, and go to step 2. ■

Note that the above algorithm differs from the Fletcher-Reeves algorithm only in the formula for γ_i (compare (55) and (46)). Note also that when $f^0(z) = \langle d, z \rangle + \frac{1}{2}\langle z, Hz \rangle$, where H is an $n \times n$ symmetric, positive definite matrix, the γ_i as given by (46) and the γ_i as given by (55) are identical according to (38), and hence in the case of such an $f^0(\cdot)$, the two methods become identical. Now let us see what happens in the general case of problem (3) under the assumptions stated.

- 56 **Theorem.** Consider problem (3) under the assumptions stated. If z_0, z_1, z_2, \dots and h_0, h_1, h_2, \dots are sequences constructed by algorithm (51), then there exists a $\rho > 0$ such that

$$57 \quad -\langle \nabla f^0(z_i), h_i \rangle \geq \rho \| \nabla f^0(z_i) \| \| h_i \|.$$

Proof. For every $z \in \mathbb{R}^n$, let $H(z) = \partial^2 f^0(z)/\partial z^2$ and let $g(z) = -\nabla f^0(z)$. Then, making use of the Taylor formula (B.1.3), we obtain

$$58 \quad -g_{i+1} = -g(z_{i+1}) = -g(z_i + \lambda_i h_i) = -g_i + \lambda_i \int_0^1 H(z_i + t\lambda_i h_i) dt h_i.$$

Since by construction, $\langle h_i, g_{i+1} \rangle = 0$, we get from (58),

$$59 \quad \lambda_i = \frac{\langle h_i, g_i \rangle}{\langle h_i, H_i h_i \rangle} = \frac{\langle g_i, g_i \rangle}{\langle h_i, H_i h_i \rangle},$$

where

$$60 \quad H_i = \int_0^1 H(z_i + t\lambda_i h_i) dt.$$

(Note that the second half of (59) is obtained from the fact that $\langle h_{i-1}, g_i \rangle = 0$ and that $h_i = g_i + \gamma_i h_{i-1}$.) Now, from (55) and (58), together with (59), we obtain that

$$61 \quad \begin{aligned} \gamma_i &= -\frac{\langle g_{i+1}, H_i h_i \rangle}{\langle g_i, g_i \rangle} \cdot \frac{\langle g_i, g_i \rangle}{\langle h_i, H_i h_i \rangle} \\ &= -\frac{\langle g_{i+1}, H_i h_i \rangle}{\langle h_i, H_i h_i \rangle}. \end{aligned}$$

But from (4) and (60) we deduce that for all $h \in \mathbb{R}^n$, $m \| h \|^2 \leq \langle h, H_i h \rangle \leq M \| h \|^2$, and hence we must have

$$62 \quad |\gamma_i| \leq \frac{\|g_{i+1}\| \|H_i\| \|h_i\|}{m \|h_i\|^2} \leq \frac{\|g_{i+1}\| M}{\|h_i\| m},$$

since, as can be easily shown, $\|H_i\| \leq M$. Now, by the triangle inequality,

$$63 \quad \|h_{i+1}\| \leq \|g_{i+1}\| + |\gamma_i| \|h_i\|,$$

which becomes, because of (62),

$$64 \quad \|h_{i+1}\| \leq \|g_{i+1}\| \left(1 + \frac{M}{m}\right).$$

Finally,

$$65 \quad \langle h_{i+1}, g_{i+1} \rangle = \langle g_{i+1} + \gamma_i h_i, g_{i+1} \rangle = \langle g_{i+1}, g_{i+1} \rangle, \quad i = 0, 1, 2, \dots.$$

Consequently, for $i = 0, 1, 2, \dots$,

$$66 \quad \begin{aligned} \frac{\langle h_{i+1}, g_{i+1} \rangle}{\|h_{i+1}\| \|g_{i+1}\|} &= \frac{\|g_{i+1}\|^2}{\|h_{i+1}\| \|g_{i+1}\|} = \frac{\|g_{i+1}\|}{\|h_{i+1}\|} \\ &\geq \frac{\|g_{i+1}\|}{\|g_{i+1}\| (1 + (M/m))} = \frac{1}{(1 + (M/m))}, \end{aligned}$$

which is the desired result. ■

The following corollary is a direct consequence of theorems (8) and (56):

- 67 **Corollary.** Consider problem (3) under the assumptions stated. Then either algorithm (51) constructs a finite sequence $\{z_i\}$ whose last element, z_k , satisfies $\nabla f^0(z_k) = 0$, or else algorithm (51) constructs an infinite sequence $\{z_i\}$ which converges to a point \hat{z} such that $\nabla f^0(\hat{z}) = 0$ (i.e., either z_k or \hat{z} minimizes $f^0(z)$ over all $z \in \mathbb{R}^n$). ■

Since it is impossible to calculate λ_i exactly according to (52) (or (43)) there is an accumulation of errors in an actual computation which can affect the convergence properties of algorithms (42) and (51). Usually, to avoid such an accumulation of errors, these algorithms are reinitialized after each $k \geq n + 1$ iterations. In algorithms (42) and (51), reinitialization after k iterations amounts to setting $h_{ik} = g_{ik}$, for $i = 0, 1, 2, \dots$, rather than computing h_{ik} according to (46) or according to (55). For a discussion of the rate of convergence of such conjugate gradient methods “with reinitialization,” see Section 6.3.

From a computational point of view, the algorithms (42) and (51) involve approximately the same amount of work per iteration. However, so far, a convergent implementation has been obtained only for (51); see (C.4.1) and [K4].

We conclude this section with the Fletcher-Powell version [F3] of the Davidon variable metric algorithm [D2]. The original description of this method was published by Davidon [D2] in 1959 and it included a number of empirical devices. Fletcher and Powell [F3] presented a simpler and more basic description in 1963. Their description assumes that all calculations are worked out exactly, i.e., their version is a purely conceptual algorithm. As with the other conjugate gradient methods, this may lead to a deleterious accumulation of errors in an actual computation, in which case it is dealt with by restarting the algorithm every so many iterations.

The variable metric algorithm is very popular and has good computational stability properties. Its only drawback is that at each iteration the computer is required to store an $n \times n$ matrix H_i , which may cause difficulties on a small machine when $n \geq 100$. In discrete optimal control applications, n is often as large as 1000, and in this case one would be more inclined to use one of the two preceding methods.

68 Algorithm (variable metric; Davidon [D2], Fletcher and Powell [F3]).

Step 0. Select a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$, set $H_0 = I$ (the $n \times n$ identity matrix), and set $g_0 = \nabla f^0(z_0)$.*

Comment. Note that both g_i and H_i are not defined in the same manner here as they are in (54) and (60), respectively. Note that h_i is also defined in a manner different from the one of algorithms (42) and (51).

Step 2. Set

$$69 \quad h_i = -H_i g_i .$$

Step 3. Compute $\lambda_i > 0$ such that

$$70 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 4. Compute $\nabla f^0(z_i + \lambda_i h_i)$.

Step 5. If $\nabla f^0(z_i + \lambda_i h_i) = 0$, stop; else, set

$$71 \quad z_{i+1} = z_i + \lambda_i h_i ,$$

$$72 \quad g_{i+1} = \nabla f^0(z_{i+1}) ,$$

$$73 \quad \Delta g_i = g_{i+1} - g_i ,$$

$$74 \quad \Delta z_i = z_{i+1} - z_i ,$$

* The choice $H_0 = I$ is not mandatory. We may choose H_0 to be any symmetric positive definite matrix.

$$75 \quad H_{i+1} = H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} \langle H_i \Delta g_i \rangle \left\langle H_i \Delta g_i + \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta z_i \right\rangle \langle \Delta z_i \rangle$$

and go to step 6.*

Step 6. Set $i = i + 1$ and go to step 2. ■

It has been shown by Meyer [M7] that when it is applied to a quadratic function $f^0(z) = \langle d, z \rangle + \frac{1}{2} \langle z, Hz \rangle$, the variable metric method constructs exactly the same sequences of vectors z_i and h_i as the methods (42) and (51) (see [M7]). Hence, this is a method of the same type as the two preceding ones.

In order to show that algorithm (68) is of the type (6), we only need to show that the matrices H_i are positive definite.

76 **Theorem** (Fletcher-Powell [F3]). For $i = 0, 1, 2, \dots$, the matrices H_i constructed by algorithm (68) are symmetric and positive definite.

Proof. For $i = 0$, $H_i = I$, a symmetric, positive definite matrix. By (75), H_{i+1} is symmetric if H_i is symmetric; hence, we only need to prove that the matrices H_i are positive definite. We give a proof by induction. Suppose that H_i is positive definite ($H_i > 0$). Then, for any nonzero vector $z \in \mathbb{R}^n$,

$$77 \quad \langle z, H_{i+1}z \rangle = \langle z, H_i z \rangle - \frac{\langle z, H_i \Delta g_i \rangle^2}{\langle \Delta g_i, H_i \Delta g_i \rangle} + \frac{\langle z, \Delta z_i \rangle^2}{\langle \Delta z_i, \Delta g_i \rangle}.$$

Since $H_i > 0$, $H_i^{1/2}$ is a well-defined positive definite matrix. Now let $p = H_i^{1/2}z$ and let $q = -H_i^{1/2}\Delta g_i$. Then (77) becomes

$$78 \quad \langle z, H_{i+1}z \rangle = \frac{\langle p, p \rangle \langle q, q \rangle - \langle p, q \rangle^2}{\langle q, q \rangle} + \frac{\langle z, \Delta z_i \rangle^2}{\langle \Delta z_i, \Delta g_i \rangle}.$$

Applying the Schwartz inequality, we now obtain $\langle p, p \rangle \langle q, q \rangle = \|p\|^2 \|q\|^2 \geq \langle p, q \rangle^2$, and hence,

$$79 \quad \langle z, H_{i+1}z \rangle \geq \frac{\langle z, \Delta z_i \rangle^2}{\langle \Delta z_i, \Delta g_i \rangle}.$$

Now, since to satisfy (70) we must have $\langle \Delta z_i, g_{i+1} \rangle = 0$, we obtain

$$80 \quad \begin{aligned} \langle \Delta z_i, \Delta g_i \rangle &= \langle \Delta z_i, g_{i+1} \rangle - \langle \Delta z_i, g_i \rangle \\ &= -\langle \Delta z_i, g_i \rangle \\ &= \lambda_i \langle H_i g_i, g_i \rangle > 0. \end{aligned}$$

Hence, we conclude that

$$81 \quad \langle z, H_{i+1}z \rangle \geq 0.$$

* For x, y in \mathbb{R}^n we denote by $x \times y$ the $n \times n$ matrix xy^T , i.e., the ij th component of $x \times y$ is $x^i y^j$. Note that $(x \times y)z = x \langle y, z \rangle = \langle y, z \rangle x$. (The matrix $x \times y$ is called a dyad.)

Now suppose that $z \neq 0$, but $\langle z, H_{i+1}z \rangle = 0$. Then, from (78) and (80), we obtain that

$$82 \quad \langle z, \Delta z_i \rangle = 0$$

$$83 \quad \langle p, p \rangle \langle q, q \rangle = \langle p, q \rangle^2.$$

But (83) implies that $p = \alpha q$ for some real number α , i.e., that $z = -\alpha \Delta g_i$. Hence, by (82), we must have $\langle \Delta g_i, \Delta z_i \rangle = 0$, which contradicts (80). Therefore, we conclude that if $H_i > 0$, then $H_{i+1} > 0$, and since $H_0 > 0$, we must have $H_i > 0$ for $i = 0, 1, 2, \dots$. ■

At the time the variable metric method was introduced, it was shown that it solved the problem $\min\{\langle z, d \rangle + \frac{1}{2}\langle z, Hz \rangle \mid z \in \mathbb{R}^n\}$ (where H is symmetric and positive definite) in no more than n iterations. Subsequently, as we have already pointed out, it was shown by Meyer [M7] that, when applied to the minimization of a quadratic form, the Fletcher-Reeves and the Davidon-Fletcher-Powell methods produce identical sequences of vectors z_i, h_i . However, in the ten years since its invention, no proof of convergence of the variable metric method for a more general case has been published. Quite recently, Powell has obtained both a proof of convergence and a rate of convergence for the variable metric method when applied to the minimization of strictly convex functions. Powell was kind enough to supply the author with a still unpublished manuscript [P9] and we shall now reproduce some of these new results. Powell's results on the rate of convergence will be presented in Section 6.4.

- 84 **Theorem (Powell).** Consider problem (3). If, in addition to the assumptions already stated on $f^0(\cdot)$, there exists a Lipschitz constant $L > 0$ such that, for all $z \in \{z \mid f^0(z) \leq f^0(z_0)\}$,

$$85 \quad \left\| \frac{\partial^2 f^0(z)}{\partial z^2} - \frac{\partial^2 f^0(\hat{z})}{\partial z^2} \right\| \leq L \|z - \hat{z}\|,$$

where \hat{z} minimizes $f^0(z)$ over $z \in \mathbb{R}^n$, then there exist $0 < \bar{m} \leq \bar{M} < \infty$ such that for $i = 0, 1, 2, \dots$,

$$86 \quad m \|y\|^2 \leq \langle y, H_i y \rangle \leq M \|y\|^2, \quad \text{for all } y \in \mathbb{R}^n. \blacksquare$$

We shall give a proof of this theorem in Section 6.4 (see (6.4.40)).

- 87 **Exercise.** Show that under the assumptions stated on $f^0(\cdot)$ in theorem (84), the variable metric algorithm (68) satisfies the assumptions of theorem (8). Hence, show that in this case the algorithm (68) either constructs a finite sequence $\{z_i\}$, whose last element z_k minimizes $f^0(z)$ over $z \in \mathbb{R}^n$, or else it constructs an infinite sequence $\{z_i\}$ which converges to a point \hat{z} which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$. (This result was also obtained by Daniel [D1].) ■

We shall now present Powell's proof of convergence for the variable metric method as it applies to problem (3) with the conditions stated. The proof makes use of the following four lemmas:

- 88 **Lemma.** For every $z_0 \in \mathbb{R}^n$, the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is convex and compact. ■

This lemma follows directly from (B.2.8); we therefore omit a proof.

- 89 **Lemma.** If z_i, z_{i+1} are any two points in the set $\{z \mid f^0(z) \leq f^0(z_0)\}$, where z_0 is any given point in \mathbb{R}^n , then the ratios $\|\Delta z_i\|/\|\Delta g_i\|$, $\|\Delta g_i\|/\|\Delta z_i\|$, $\langle\Delta z_i, \Delta g_i\rangle/\|\Delta z_i\|^2$, $\|\Delta z_i\|^2/\langle\Delta z_i, \Delta g_i\rangle$, $\langle\Delta z_i, \Delta g_i\rangle/\|\Delta g_i\|^2$, and

$$\|\Delta g_i\|^2/\langle\Delta z_i, \Delta g_i\rangle$$

are all bounded, where

90
$$\Delta z_i = z_{i+1} - z_i,$$

91
$$\Delta g_i = g(z_{i+1}) - g(z_i),$$

and $g(\cdot)$ is used to denote $\nabla f^0(\cdot)$.

Proof. Because of the Schwartz inequality, $\langle\Delta z_i, \Delta g_i\rangle \leq \|\Delta z_i\| \|\Delta g_i\|$. Hence, we only need to prove that the ratios $\|\Delta g_i\|/\|\Delta z_i\|$ and $\|\Delta z_i\|^2/\langle\Delta z_i, \Delta g_i\rangle$ are bounded, since the boundedness of the other ratios follows immediately from the boundedness of these ratios.

As before, let $H(z) = \partial^2 f^0(z)/\partial z^2$ for all $z \in \mathbb{R}^n$. We begin by showing that the ratio $\|\Delta g_i\|/\|\Delta z_i\|$ is bounded. By the generalized mean-value theorem (B.1.5),

92
$$\|\Delta g_i\| = \|g(z_i + \Delta z_i) - g(z_i)\| \leq \|\Delta z_i\| \sup_{z \in [z_i, z_{i+1}]} \|H(z)\|.$$

Since $H(\cdot)$ is assumed to be continuous and the convex set $\{z \mid f^0(z) \leq f^0(z_0)\}$ (where z_0 is the given starting point) is compact by lemma (88), it follows that there exists a finite number $V > 0$ such that $\|H(z)\| \leq V$ for all z satisfying $f^0(z) \leq f^0(z_0)$. Hence,

93
$$\frac{\|\Delta g_i\|}{\|\Delta z_i\|} \leq V < \infty.$$

To show that $\|\Delta z_i\|^2/\langle\Delta z_i, \Delta g_i\rangle$ is bounded, we make use of the Taylor formula (B.1.3) to obtain

94
$$\int_0^1 \langle \Delta z_i, H(z_i + t \Delta z_i) \Delta z_i \rangle dt = \langle \Delta z_i, \Delta g_i \rangle.$$

Making use of (4), we find that

95
$$\frac{\|\Delta z_i\|^2}{\langle \Delta z_i, \Delta g_i \rangle} \leq \frac{1}{m}. \quad \blacksquare$$

96 **Lemma.** Suppose that \hat{z} minimizes $f^0(z)$ over $z \in \mathbb{R}^n$ and that $g(\cdot)$ denotes $\nabla f^0(\cdot)$; then

$$97 \quad \|g(z)\|^2 \geq m[f^0(z) - f^0(\hat{z})]$$

for all $z \in \mathbb{R}^n$.

Proof. Since $f^0(\cdot)$ is convex, for any $z \in \mathbb{R}^n$ (see (B.2.4)),

$$98 \quad f^0(z) - f^0(\hat{z}) \leq \langle g(z), z - \hat{z} \rangle \leq \|g(z)\| \|z - \hat{z}\|.$$

Making use of (95) and the Schwarz inequality (after renaming the variables in (95)), we find that

$$99 \quad \|z - \hat{z}\|^2 \leq \frac{1}{m} \langle z - \hat{z}, g(z) - g(\hat{z}) \rangle \leq \frac{1}{m} \|z - \hat{z}\| \|g(z) - g(\hat{z})\|.$$

Since $g(\hat{z}) = 0$, we obtain from (98) and (99) that

$$100 \quad f^0(z) - f^0(\hat{z}) \leq \frac{1}{m} \|g(z)\|^2. \quad \blacksquare$$

101 **Lemma.** The sequence of points $\{z_i\}$ generated by the variable metric algorithm (68) has the property that

$$102 \quad \lim_{k \rightarrow \infty} \sum_{i=0}^k \|\Delta z_i\|^2 < \infty, \quad \lim_{k \rightarrow \infty} \sum_{i=0}^k \|\Delta g_i\|^2 < \infty,$$

where Δz_i and Δg_i are defined as in (90) and (91), respectively. (This lemma is trivially true when the sequence constructed by (68) is finite.)

Proof. Making use of the Taylor formula (B.1.12), we obtain

$$103 \quad \begin{aligned} f^0(z_i) - f^0(z_{i+1}) \\ = -\langle g_{i+1}, \Delta z_i \rangle + \int_0^1 (1-t) \langle \Delta z_i, H(z_{i+1} - t \Delta z_i) \Delta z_i \rangle dt. \end{aligned}$$

Now, $-\langle g_{i+1}, \Delta z_i \rangle = 0$ by construction of z_{i+1} . Because of this and because of (4), we now obtain

$$104 \quad f^0(z_i) - f^0(z_{i+1}) \geq \frac{m \|\Delta z_i\|^2}{2}.$$

Summing (104) from $i = 0$ to $i = \infty$, we obtain

$$105 \quad \sum_{i=0}^{\infty} \|\Delta z_i\|^2 \leq \frac{1}{2m} [f^0(z_0) - f^0(\hat{z})].$$

(Since the sequence $\{f^0(z_i)\}$ is monotonically decreasing and is bounded from below by $f^0(\hat{z})$, it must converge to a number $f^0 \geq f^0(\hat{z})$. Hence, $\sum_{i=0}^{\infty} [f^0(z_i) - f^0(z_{i+1})] = f^0(z_0) - f^0 \leq f^0(z_0) - f^0(\hat{z})$, which gives (105).) We have thus proved the first inequality in (102); the second inequality in (102) now follows from the bounds established in lemma (89). ■

- 106 **Theorem (Powell).** Consider problem (3) with the assumptions stated. If $\{z_i\}$ is an infinite sequence generated by the variable metric algorithm (68), then it converges to the point \hat{z} which minimizes $f^0(\hat{z})$ over $\hat{z} \in \mathbb{R}^n$. (Note that the algorithm stops constructing new points z_i if and only if it has constructed a z_{i+1} such that $\nabla f^0(z_{i+1}) = 0$. Hence, the case of a finite sequence is trivial.)

Proof. The proof is quite long and it may help the reader to have a plan which will be followed. The plan is to obtain an expression for the trace of the matrices $G_i = H_i^{-1}$, $i = 0, 1, 2, \dots$, and to show that one gets a contradiction unless the sequence of gradients $\{g_i\}$ converges to zero. Note that if $g_i = g(z_i) \rightarrow 0$, then, since the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ is compact and $g(\cdot)$ is continuous, the sequence $\{z_i\}$ must contain accumulation points z' , at each of which $g(z') = 0$. But $f^0(\cdot)$ is strictly convex, and hence there exists exactly one point $\hat{z} \in \mathbb{R}^n$ such that $g(\hat{z}) = 0$. Consequently, the sequence $\{z_i\}$ has only one accumulation point, \hat{z} , and it converges to this point. Therefore we only need to show that the sequence of gradients $\{g_i\}$ converges to zero.

We begin by obtaining an expression for the trace of the matrices $G_i = H_i^{-1}$. First, for the sake of convenience, we reproduce (75), the difference equation governing the evolution of the matrices H_i :

$$107 \quad H_{i+1} = H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i \Delta g_i \langle H_i \Delta g_i \rangle + \frac{1}{\langle z \Delta_i, \Delta g_i \rangle} \Delta z_i \langle \Delta z_i \rangle$$

It is not difficult to deduce from the properties of dyads that

$$108 \quad G_{i+1} = \left(I - \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta g_i \langle \Delta z_i \rangle \right) G_i \left(I - \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta z_i \langle \Delta g_i \rangle \right) \\ + \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta g_i \langle \Delta g_i \rangle,$$

where I is the $n \times n$ identity matrix. Formula (108) is easily verified by simply showing that $G_{i+1}H_{i+1} = I$, the $n \times n$ identity matrix, with H_{i+1} defined by (107) and G_{i+1} defined by (108).

Next, observing that the trace of an $n \times n$ dyad $x \otimes y$ is $\langle x, y \rangle$, we obtain from (108) that $\text{Tr}(G_{i+1})$, the trace of G_{i+1} , is given by

$$109 \quad \begin{aligned} \text{Tr}(G_{i+1}) &= \text{Tr}(G_i) - 2 \frac{\langle \Delta z_i, G_i \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle} + \frac{\langle \Delta z_i, G_i \Delta z_i \rangle \langle \Delta g_i, \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle^2} \\ &\quad + \frac{\langle \Delta g_i, \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle}. \end{aligned}$$

Making use of (69), (71), (73) and the fact that because of (70), $\langle \Delta z_i, g_{i+1} \rangle = 0$, we can simplify the two middle terms in (109) as follows:

$$110 \quad \begin{aligned} &-2 \frac{\langle \Delta z_i, G_i \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle} + \frac{\langle \Delta z_i, G_i \Delta z_i \rangle \langle \Delta g_i, \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle^2} \\ &= \frac{2 \langle g_i, \Delta g_i \rangle}{\langle H_i g_i, g_i \rangle} + \frac{\langle g_i, H_i g_i \rangle \langle \Delta g_i, \Delta g_i \rangle}{\langle H_i g_i, g_i \rangle^2} \\ &= \frac{2 \langle g_i, \Delta g_i \rangle + \langle \Delta g_i, \Delta g_i \rangle}{\langle H_i g_i, g_i \rangle} = \frac{\|g_{i+1}\|^2 - \|g_i\|^2}{\langle H_i g_i, g_i \rangle}. \end{aligned}$$

In order to further simplify (109) we shall now prove that

$$111 \quad \frac{1}{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle} = \frac{1}{\langle g_{i+1}, H_i g_{i+1} \rangle} + \frac{1}{\langle g_i, H_i g_i \rangle}.$$

Making use of (73), (74), (75) and the fact that $\langle g_{i+1}, \Delta z_i \rangle = 0$, we obtain that

$$112 \quad \langle g_{i+1}, H_{i+1} g_{i+1} \rangle = \left\langle g_{i+1}, \left(H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i \Delta g_i \langle H_i \Delta g_i \rangle g_{i+1} \right) g_{i+1} \right\rangle.$$

Next, substituting $g_i + \Delta g_i$ for g_{i+1} in (112), and making use of the fact that $\langle \Delta g_i, H_i g_i \rangle = \langle g_{i+1} - g_i, H_i g_i \rangle = -\langle g_i, H_i g_i \rangle$ (since $\langle g_{i+1}, H_i g_i \rangle = 0$), we obtain from (112) that

$$113 \quad \begin{aligned} \langle g_{i+1}, H_i g_{i+1} \rangle &= \left\langle g_i, \left(H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i \Delta g_i \langle H_i \Delta g_i \rangle g_i \right) g_i \right\rangle \\ &= \left\langle g_i, \left(H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i g_i \langle H_i g_i \rangle g_i \right) g_i \right\rangle \\ &= \frac{\langle g_i, H_i g_i \rangle \langle g_{i+1}, H_i g_{i+1} \rangle}{\langle g_i, H_i g_i \rangle + \langle g_{i+1}, H_i g_{i+1} \rangle}. \end{aligned}$$

The last line in (113) follows from the fact that

$$\begin{aligned} 114 \quad \langle \Delta g_i, H_i \Delta g_i \rangle &= \langle g_{i+1}, H_i g_{i+1} \rangle + \langle g_i, H_i g_i \rangle - 2\langle g_{i+1}, H_i g_i \rangle \\ &= \langle g_{i+1}, H_i g_{i+1} \rangle + \langle g_i, H_i g_i \rangle. \end{aligned}$$

Upon inverting both sides of (113), we obtain (111). Now substituting from (111) into (110) and hence into (109), we obtain that

$$\begin{aligned} 115 \quad \text{Tr}(G_{i+1}) &= \text{Tr}(G_i) + \frac{\|g_{i+1}\|^2}{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle} - \frac{\|g_i\|^2}{\langle g_i, H_i g_i \rangle} \\ &\quad - \frac{\|g_{i+1}\|^2}{\langle g_{i+1}, H_i g_{i+1} \rangle} + \frac{\langle \Delta g_i, \Delta g_i \rangle}{\langle \Delta z_i, \Delta g_i \rangle}, \quad i = 0, 1, 2, \dots . \end{aligned}$$

Solving the difference equation (115), we now obtain

$$\begin{aligned} 116 \quad \text{Tr}(G_{i+1}) &= \text{Tr}(G_0) + \frac{\|g_{i+1}\|^2}{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle} - \frac{\|g_0\|^2}{\langle g_0, H_0 g_0 \rangle} \\ &\quad - \sum_{j=0}^i \frac{\|g_{j+1}\|^2}{\langle g_{j+1}, H_j g_{j+1} \rangle} + \sum_{j=0}^i \frac{\|\Delta g_j\|^2}{\langle \Delta z_j, \Delta g_j \rangle}, \quad i = 0, 1, 2, 3, \dots . \end{aligned}$$

By lemma (89), each term in the last sum of (116) is bounded, and hence, there exists a finite number $w > 0$, depending on z_0 but not on i , such that for $i = 0, 1, 2, \dots$,

$$117 \quad \text{Tr}(G_{i+1}) \leq \frac{\|g_{i+1}\|^2}{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle} - \sum_{j=0}^i \frac{\|g_{j+1}\|^2}{\langle g_{j+1}, H_j g_{j+1} \rangle} + (i+1)w.$$

Most of the remainder of the proof consists of showing that if the sequence $\{g_i\}$ does not converge to zero, then the last two terms on the right-hand side of (117) must be negative. We shall prove this by showing that the sequence $\langle g_{i+1}, H_i g_{i+1} \rangle$ contains a subsequence ($j+1 \in K$) which converges to zero, and that there has to exist a $v > 0$ such that $\|g_{j+1}\|^2 \geq v$ for all the $j+1$ in K .

To show that the sequence $\langle g_{i+1}, H_i g_{i+1} \rangle$ contains a subsequence which converges to zero, we examine the trace of the matrix H_{i+1} . From (107),

$$118 \quad \text{Tr}(H_{i+1}) = \text{Tr}(H_i) - \frac{\|H_i \Delta g_i\|^2}{\langle \Delta g_i, H_i \Delta g_i \rangle} + \frac{\|\Delta z_i\|^2}{\langle \Delta z_i, \Delta g_i \rangle}.$$

Solving the difference equation (118), we now obtain

$$119 \quad \text{Tr}(H_{i+1}) = \text{Tr}(H_0) - \sum_{j=0}^i \frac{\| H_j \Delta g_j \|^2}{\langle \Delta g_j, H_j \Delta g_j \rangle} + \sum_{j=0}^i \frac{\| \Delta z_j \|^2}{\langle \Delta z_j, \Delta g_j \rangle}, \quad i = 0, 1, 2, \dots.$$

Because the matrix H_{i+1} is positive definite, $\text{Tr}(H_{i+1}) > 0$. Hence, because of lemma (89), there must exist a number $w' > 0$ such that

$$120 \quad \sum_{j=0}^i \frac{\| H_j \Delta g_j \|^2}{\langle \Delta g_j, H_j \Delta g_j \rangle} < (i + 1) w', \quad i = 0, 1, 2, 3, \dots.$$

Now, by the Schwarz inequality,

$$121 \quad \langle \Delta g_j, H_j \Delta g_j \rangle^2 \leq \| H_j \Delta g_j \|^2 \| \Delta g_j \|^2,$$

and since H_j is positive definite, (114) gives

$$122 \quad \langle \Delta g_j, H_j \Delta g_j \rangle > \langle g_{j+1}, H_j g_{j+1} \rangle.$$

Making use of these facts and (120), we obtain

$$123 \quad \sum_{j=0}^i \frac{\langle g_{j+1}, H_j g_{j+1} \rangle}{\| \Delta g_j \|^2} < (i + 1) w'.$$

Therefore, at least two-thirds of the terms in the sum (123) must satisfy

$$124 \quad \langle g_{j+1}, H_j g_{j+1} \rangle < 3w' \| \Delta g_j \|^2,$$

for otherwise the left side of the inequality (123) would be larger than the right side. Now, lemma (101) established that $\sum_{i=0}^{\infty} \| \Delta g_i \|^2 < \infty$ and hence, for $i = 0, 1, 2, \dots$, at least two-thirds of the elements in $\{\langle g_{j+1}, H_j g_{j+1} \rangle\}_{j=0}^i$ belong to a subsequence which converges to zero.

We now examine the numerators in (117). Continuing to assume that the sequence $\{g_i\}$ does not converge to zero, we conclude that the monotonically decreasing sequence $f^0(z_i)$ converges to $f^0 > f^0(\hat{z})$, and hence that

$$125 \quad f^0(z_i) - f^0(\hat{z}) > v' > 0, \quad i = 0, 1, 2, 3, \dots.$$

Setting $v = mv'$, we now obtain from (97) that

$$126 \quad \| g_i \|^2 > v > 0, \quad i = 0, 1, 2, 3, \dots.$$

Now, since by lemma (101), $\sum_{j=0}^{\infty} \|\Delta g_j\|^2 < \infty$, we must have $\|\Delta g_j\|^2 \rightarrow 0$, as $j \rightarrow \infty$, and hence there exists an integer k such that $3w' \|\Delta g_j\|^2 < v/3w$ for all $j \geq k$. Setting $i = 3k + 2$ in (117), we find that

$$127 \quad \sum_{j=0}^{3k+2} \frac{\|g_{j+1}\|^2}{\langle g_{j+1}, H_j g_{j+1} \rangle} > (k+1)w,$$

since by the preceding discussion, at least $(k+1)$ of the $3(k+1)$ terms in (127) satisfy

$$128 \quad \frac{\|g_{j+1}\|^2}{\langle g_{j+1}, H_j g_{j+1} \rangle} > v/(v/3w) = 3w, \quad \text{with } j \geq k.$$

Hence, from (117), for $i \geq 3k + 2$,

$$129 \quad \text{Tr}(G_{i+1}) < \frac{\|g_{i+1}\|^2}{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle}.$$

We shall now show that this inequality leads to a contradiction. The trace of a symmetric matrix is the sum of its eigenvalues, and therefore the trace of the positive definite matrix G_{i+1} is an upper bound on its largest eigenvalue which is the inverse of the smallest eigenvalue of H_{i+1} . Thus, let μ_{i+1} be the smallest eigenvalue of H_{i+1} ; then, because of (129),

$$130 \quad \mu_{i+1} \geq 1/\text{Tr}(G_{i+1}) > \frac{\langle g_{i+1}, H_{i+1} g_{i+1} \rangle}{\|g_{i+1}\|^2}.$$

However, since H_{i+1} is a positive definite matrix, we must also have

$$131 \quad \langle g_{i+1}, H_{i+1} g_{i+1} \rangle \geq \mu_{i+1} \|g_{i+1}\|^2,$$

which is contradicted by (130). We therefore conclude that $g_i \rightarrow 0$ as $i \rightarrow \infty$, which completes our proof. ■

Computational Aspects

To implement an algorithm such as (42), (51) or (68), Fletcher and Powell compute the step size λ_i as follows (see (52) and (70)): First they compute $\theta(\bar{\lambda}; z_i) = f^0(z_i + \bar{\lambda}h_i) - f^0(z_i)$ and $\theta_\lambda(\bar{\lambda}, z_i) = \langle \Delta f^0(z_i + \bar{\lambda}h_i), h_i \rangle$ for some fixed value of $\bar{\lambda}$ (say for $\bar{\lambda} = 1$). Then they use the four values $\theta(0; z_i)$, $\theta(\bar{\lambda}; z_i)$, $\theta_\lambda(0; z_i)$ and $\theta_\lambda(\bar{\lambda}; z_i)$ to construct an interpolating cubic polynomial $p(\cdot)$ to the function $\theta(\cdot; z_i)$, and they compute a λ' which minimizes $p(\lambda)$ for $\lambda \geq 0$. If $\theta(\lambda'; z_i) < 0$, they set $\lambda_i = \lambda'$; otherwise they set $\lambda = \beta\bar{\lambda}$ ($\beta \in (0, 1)$) and repeat. With this type of approach to the implementation

of the algorithms (42), (51) and (68), the Fletcher-Reeves (42) and Polak-Ribière (51) algorithms converge very slowly, while the variable metric algorithm (68), converges quite well. To restore the convergence rate of algorithms (42) and (51) (with λ_i computed as above), one has to reinitialize these algorithms after every $n + 1$ iterations (see (6.3.4), (6.3.60)).

Algorithm (1.16) suggests the following alternative implementation of algorithms (42), (51) and (68):

132 Algorithm (Polak).

Step 0. Select an $\epsilon_0 > 0$, an $\alpha > 0$, a $\beta \in (0, 1)$ and a $\rho > 0$; select a $z_0 \in \mathbb{R}^n$; and set $i = 0$.

Step 1. Set $\epsilon = \epsilon_0$.

Step 2. Compute $\nabla f^0(z_i)$.

Step 3. If $\nabla f^0(z_i) = 0$, stop; else, compute h_i (as in (42), (51) or (68), but always with the same rule).

Step 4. Set $\theta(\mu) = f^0(z_i + \mu h_i) - f^0(z_i)$ and use algorithm (1.14) with the current value of ϵ to compute $\bar{\mu}$.

Step 5. If $\theta(\bar{\mu}) \leq -\alpha\epsilon$, set $\lambda_i = \bar{\mu}$ and go to step 6; else, set $\epsilon = \beta\epsilon$ and go to step 4.

Step 6. Set $z_{i+1} = z_i + \lambda_i h_i$, set $i = i + 1$, and go to step 3. ■

The author has some experimental indications that when algorithm (51) (and presumably also (42)) is implemented as above, there is no need to reinitialize the algorithm as in (6.3.4). However, one probably does spend more time on function evaluations when using the implementation (132).

This concludes the first part of our discussion of conjugate gradient methods. We shall discuss their rate of convergence in Sections 6.3 and 6.4, and give additional implementations in Section C.4.

2.4 Unconstrained Discrete Optimal Control Problems

As we have already seen in Chapter 1, discrete optimal control problems can be viewed as nonlinear programming problems with a special structure and, usually, large dimensions. Therefore, all the algorithms we have discussed so far in this chapter are, at least in principle, applicable to unconstrained discrete optimal control problems. In this section we shall discuss the use of algorithms requiring gradient evaluations for solving discrete optimal control problems. We recall that such algorithms require us to calculate the gradient of the cost function at each iteration, a task which can become prohibitive when the dimension of the space over which we are trying to minimize the cost function is very large. Fortunately, as we shall soon see, the structure of the optimal control problem comes to our aid and enables us to substitute a number of “low-dimensional” operations for one “high-dimensional” operation.

Most frequently, unconstrained discrete optimal control problems arise when penalty functions, to be discussed in the next chapter, are used to cope with constraints on the states and controls of the dynamical system. These unconstrained problems are usually encountered in the following form:

$$1 \quad \text{minimize} \quad \sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^\mu,$$

subject to

$$2 \quad x_{i+1} - x_i = f_i(x_i, u_i), \quad i = 0, 1, \dots, k-1, \quad \text{with } x_0 = \hat{x}_0,$$

where the functions $f_i^0(\cdot, \cdot)$, $f_i(\cdot, \cdot)$ and $\varphi(\cdot)$ are continuously differentiable in all their arguments.

Since the initial state $x_0 = \hat{x}_0$ is given, the states x_i are uniquely determined through (2) by the control sequence $z = (u_0, u_1, \dots, u_{k-1})$, which we shall treat as a column vector in our equations to follow. Thus, we may write $x_i = x_i(z)$, and understand by this that $x_i(z)$ is computed by solving the system (2) with initial state $x_0 = \hat{x}_0$ and input sequence $z = (u_0, u_1, \dots, u_{k-1})$ up to time i . Because of this, we see that problem (1) is of the form

$$3 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\},$$

where $z = (u_0, u_1, \dots, u_{k-1}) \in \mathbb{R}^{k\mu}$, so that $n = k\mu$, and

$$4 \quad f^0(z) = \sum_{i=0}^{k-1} f_i^0(x_i(z), u_i) + \varphi(x_k(z)).$$

Since k may be quite large (say 1000), n in (1) may be even larger, which makes a brute force computation of $\nabla f^0(z)$ highly cumbersome, at best. We shall now develop a method for computing $\nabla f^0(z)$ which at no time requires us to handle vectors of dimension any higher than v , which is usually orders of magnitude smaller than n .

We begin by noting that*

$$5 \quad \nabla f^0(z)^T = \frac{\partial f^0(z)}{\partial z} = \left(\frac{\partial f^0(z)}{\partial u_0} \mid \frac{\partial f^0(z)}{\partial u_1} \mid \dots \mid \frac{\partial f^0(z)}{\partial u_{k-1}} \right),$$

and that for $i = 0, 1, 2, \dots, k-1$,

$$6 \quad \frac{\partial f^0(z)}{\partial u_i} = \frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} + \sum_{j=i+1}^{k-1} \frac{\partial f_j^0(x_j(z), u_j)}{\partial x_j} \cdot \frac{\partial x_j(z)}{\partial u_i} + \frac{\partial \varphi(x_k(z))}{\partial x_k} \cdot \frac{\partial x_k(z)}{\partial u_i}.$$

* Note that we always treat $\nabla f^0(z)$ as a column vector. However, $\partial f^0(z)/\partial z$ is a $1 \times n$ Jacobian matrix, i.e., a row vector.

Now, for $i = 0, 1, \dots, k$ and $i \leq j \leq k$, let $\Phi_{j,i}$ be a $\nu \times \nu$ matrix which is determined by solving the matrix difference equation,

$$7 \quad \Phi_{j+1,i} - \Phi_{j,i} = \frac{\partial f_j(x_j(z), u_j)}{\partial x_j} \Phi_{j,i}, \quad j = i, i+1, \dots, k-1,$$

with the initial condition $\Phi_{i,i} = I$, the $\nu \times \nu$ identity matrix.

8 Exercise. Show that

$$9 \quad \begin{aligned} \frac{\partial x_j(z)}{\partial u_i} &= \Phi_{j,i+1} \frac{\partial f_i(x_i(z), u_i)}{\partial u_i} && \text{for } j = i+1, i+2, \dots, k \\ &= 0 && \text{for } j = 1, 2, \dots, i. \end{aligned} \quad \blacksquare$$

Making use of (9), we obtain

$$10 \quad \begin{aligned} \frac{\partial f^0(z)}{\partial u_i} &= \frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} + \sum_{j=i+1}^{k-1} \frac{\partial f_j^0(x_j(z), u_j)}{\partial x_j} \Phi_{j,i+1} \cdot \frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \\ &\quad + \frac{\partial \varphi(x_k(z))}{\partial x_k} \Phi_{k,i+1} \frac{\partial f_i(x_i(z), u_i)}{\partial u_i}. \end{aligned}$$

Since we are in the habit of working with column vectors whenever that is possible, we transpose all the terms in (10) to obtain

$$11 \quad \begin{aligned} \left(\frac{\partial f^0(z)}{\partial u_i} \right)^T &= \left(\frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} \right)^T \\ &\quad + \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T \sum_{j=i+1}^{k-1} \Phi_{j,i+1}^T \left(\frac{\partial f_j^0(x_j(z), u_j)}{\partial x_j} \right)^T \\ &\quad + \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T \Phi_{k,i+1}^T \left(\frac{\partial \varphi(x_k(z))}{\partial x_k} \right)^T. \end{aligned}$$

Now, for $i = 1, 2, \dots, k$, let p_i be the solution of (compare (1.2.25))

$$12 \quad p_i - p_{i+1} = \left(\frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \right)^T p_{i+1} - \left(\frac{\partial f_i^0(x_i(z), u_i)}{\partial x_i} \right)^T, \quad i = 0, 1, \dots, k-1,$$

with

$$p_k = - \left(\frac{\partial \varphi(x_k(z))}{\partial x_k} \right)^T.$$

Then

$$13 \quad -p_i = \Phi_{k,i}^T \left(\frac{\partial \varphi(x_k(z))}{\partial x_k} \right)^T + \sum_{j=i}^{k-1} \Phi_{j,i}^T \left(\frac{\partial f_j^0(x_j(z), u_j)}{\partial x_j} \right)^T, \quad i = 1, 2, \dots, k,$$

and hence, by inspection of (11), we get

$$14 \quad \left(\frac{\partial f^0(z)}{\partial u_i} \right)^T = \left(\frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} \right)^T - \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T p_{i+1},$$

$$i = 0, 1, \dots, k-1.$$

Thus, to compute $\nabla f^0(z)$, we first solve (12) to obtain the vectors p_1, p_2, \dots, p_k and then use (14) to complete the evaluation of $\nabla f^0(z)$.

- 15 **Exercise.** Incorporate the procedure described above for calculating $\nabla f^0(z)$ into the algorithms (1.16), (1.19), (1.35), (3.42), (3.51) and (3.68). Would you consider using a nondiagonal matrix $D(z)$ in (1.16) and (1.19) when these algorithms are to be used for solving problem (1)? Can you foresee any difficulties with rapid access storage requirements when using (3.68) to solve problem (1) on a digital computer? ■

It is important to note that the formulas developed above for calculating $\nabla f^0(z)$ could also have been developed by expanding $f^0(z + \delta z) - f^0(z)$ to first-order terms about z and then reading off $\nabla f^0(z)$ by inspection of the result. To carry out this computation we would proceed as follows:

$$16 \quad f^0(z + \delta z) - f^0(z)$$

$$\doteq \sum_{i=0}^{k-1} \left(\frac{\partial f_i^0(x_i(z), u_i)}{\partial x_i} \delta x_i + \frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} \delta u_i \right) + \frac{\partial \varphi(x_k(z))}{\partial x_k} \delta x_k,$$

where, for $i = 0, 1, 2, \dots, k$, δx_i is calculated by solving

$$17 \quad \delta x_{i+1} - \delta x_i = \frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \delta x_i + \frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \delta u_i,$$

$$i = 0, 1, \dots, k-1, \quad \text{with} \quad \delta x_0 = 0,$$

i.e., we linearize both (1) and (2) about the nominal values (x_0, x_1, \dots, x_k) and $(u_0, u_1, \dots, u_{k-1})$.

- 18 **Exercise.** Make use of (16), (17) together with (12), (13), and the fact that $f^0(z + \delta z) - f^0(z) = \langle \nabla f^0(z), \delta z \rangle$ to first-order terms (where $\delta z = (\delta u_0, \delta u_1, \dots, \delta u_{k-1})$) to obtain (14). ■

Pursuing this idea of obtaining derivatives of $f^0(\cdot)$ by suitable expansions of (1) and (2), we might try to obtain formulas for use in the quasi-Newton-Raphson method (1.42) (or the version discussed in Section 2) by expanding (1) and (2) to second-order terms. However, this does not result in any obvious simplifications, because of the nonlinear relationship between the δx_i and the δu_i . Because of this, algorithms of the form (1.16) or (1.19)

have been introduced in which the direction vector $h(z)$ is computed by solving the problem*

$$19 \quad \begin{aligned} & \text{minimize} \quad \sum_{i=0}^{k-1} \left(\frac{\partial f_i^0(x_i(z), u_i)}{\partial x_i} \delta x_i + \frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} \delta u_i \right. \\ & \quad \left. + \frac{1}{2} \left\langle \delta x_i, \frac{\partial^2 f_i^0(x_i(z), u_i)}{\partial x_i^2} \delta x_i \right\rangle + \frac{1}{2} \left\langle \delta u_i, \frac{\partial^2 f_i^0(x_i(z), u_i)}{\partial u_i^2} \delta u_i \right\rangle \right) \\ & \quad + \sum_{i=0}^{k-1} \left\langle \delta u_i, \left(\frac{\partial^2 f_i^0(x_i(z), u_i)}{\partial x_i \partial u_j} \right)^T \delta x_i \right\rangle \\ & \quad + \frac{\partial \varphi(x_k(z))}{\partial x_k} \delta x_k + \frac{1}{2} \left\langle \delta x_k, \frac{\partial^2 \varphi(x_k(z))}{\partial x_k^2} \delta x_k \right\rangle, \end{aligned}$$

* Probably the easiest way to justify all the terms in (19) is as follows: Set $y = (x, u)$, with $x \in \mathbb{R}^v$, $u \in \mathbb{R}^u$, and treat it as a column vector. Then, to second-order terms,

$$A \quad f_i^0(y + \delta y) - f_i^0(y) = \frac{\partial f_i^0(y)}{\partial y} \delta y + \frac{1}{2} \left\langle \delta y, \frac{\partial^2 f_i^0(y)}{\partial y^2} \delta y \right\rangle,$$

where

$$\delta y = (\delta x^1, \delta x^2, \dots, \delta x^v, \delta u^1, \dots, \delta u^u),$$

$$B \quad \frac{\partial f_i^0(y)}{\partial y} = \left(\frac{\partial f_i^0(y)}{\partial x^1}, \dots, \frac{\partial f_i^0(y)}{\partial x^v} \mid \frac{\partial f_i^0(y)}{\partial u^1}, \dots, \frac{\partial f_i^0(y)}{\partial u^u} \right),$$

$$C \quad \frac{\partial^2 f_i^0(y)}{\partial y^2} = \left[\begin{array}{ccc|ccc} \frac{\partial^2 f_i^0(y)}{\partial x^1 \partial x^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial x^1 \partial x^v} & \frac{\partial^2 f_i^0(y)}{\partial x^1 \partial u^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial x^1 \partial u^u} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial^2 f_i^0(y)}{\partial x^v \partial x^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial x^v \partial x^v} & \frac{\partial^2 f_i^0(y)}{\partial x^v \partial u^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial x^v \partial u^u} \\ \hline \hline \frac{\partial^2 f_i^0(y)}{\partial u^1 \partial x^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial u^1 \partial x^v} & \frac{\partial^2 f_i^0(y)}{\partial u^1 \partial u^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial u^1 \partial u^u} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial^2 f_i^0(y)}{\partial u^u \partial x^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial u^u \partial x^v} & \frac{\partial^2 f_i^0(y)}{\partial u^u \partial u^1} & \dots & \frac{\partial^2 f_i^0(y)}{\partial u^u \partial u^u} \end{array} \right].$$

Identifying the four submatrices in (C), we obtain that

$$D \quad \frac{\partial^2 f^i(y)}{\partial y^2} = \left(\begin{array}{c|c} \frac{\partial^2 f_i^0(y)}{\partial x^2} & \frac{\partial^2 f_i^0(y)}{\partial x \partial u} \\ \hline \left(\frac{\partial^2 f_i^0(y)}{\partial x \partial u} \right)^T & \frac{\partial^2 f_i^0(y)}{\partial u^2} \end{array} \right),$$

and (19) now follows by inspection. (Note that $\frac{\partial^2 f_i^0(y)}{\partial x^i \partial u^j} = \frac{\partial^2 f_i^0(y)}{\partial u^j \partial x^i}$.)

subject to

$$20 \quad \delta x_{i+1} - \delta x_i = \frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \delta x_i + \frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \delta u_i, \\ i = 0, 1, 2, \dots, k-1, \quad \text{with } \delta x_0 = 0.$$

Although such algorithms have been found to have practical value, there are no theoretical results available to support a claim that they are always convergent. We shall present an algorithm for solving (19) in the next section, where we shall also develop formulas which considerably simplify the application of quasi-Newton-Raphson methods to problem (1). Before proceeding any further, however, the reader should consolidate the above discussion by working the exercise below.

- 21 **Exercise.** Write out an algorithm of the form (1.19) for solving problem (1) in which the vector $h(z)$ is calculated by solving (19) and then setting $h(z) = (\delta u_0, \delta u_1, \dots, \delta u_{k-1})$, with $z = (u_0, u_1, \dots, u_{k-1})$, as was the case in the rest of this section. ■

2.5 Unconstrained Continuous Optimal Control Problems

We shall now show how some of the algorithms, discussed in Sections 1 and 3, can be applied to solving optimal control problems of the form

$$1 \quad \text{minimize} \quad \int_{t_0}^{t_f} f^0(x(t), u(t), t) dt + \varphi(x(t_f)),$$

subject to

$$2 \quad \frac{d}{dt} x(t) = f(x(t), u(t), t), \quad t \in [t_0, t_f], \quad x(t_0) = x_0,$$

where $f^0 : \mathbb{R}^v \times \mathbb{R}^u \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$, $f : \mathbb{R}^v \times \mathbb{R}^u \times \mathbb{R}^1 \rightarrow \mathbb{R}^v$, and $\varphi : \mathbb{R}^v \rightarrow \mathbb{R}^1$.

We shall assume that the functions $f^0(\cdot, \cdot, \cdot)$ and $f(\cdot, \cdot, \cdot)$ are continuously differentiable in x and in u and that they are piecewise continuous in t . We shall also assume that for every $x \in \mathbb{R}^v$ and for every $u \in \mathbb{R}^u$, the elements of the matrix-valued functions $\partial f^0(x, u, \cdot)/\partial x$, $\partial f^0(x, u, \cdot)/\partial u$, $\partial f(x, u, \cdot)/\partial x$, and $\partial f(x, u, \cdot)/\partial u$ are piecewise continuous. Finally, we shall assume that $\varphi(\cdot)$ is a continuously differentiable function.

In the problem (1), (2), the times t_0 and t_f are assumed to be given. However, the methods we shall develop also apply to problems in which

the terminal time t_f , is free, but finite, because free-time problems can be transcribed into fixed-time problems as follows:

3 Exercise. Consider the *free-time* optimal control problem,

$$4 \quad \text{minimize} \quad \int_{t_0}^{t_f} f^0(x(t), u(t), t) dt$$

subject to

$$5 \quad \frac{d}{dt} x(t) = f(x(t), u(t), t), \quad t \in [t_0, t_f], \quad x(t_0) = \hat{x}_0,$$

where f^0, f are as in (1) and (2) and t_0 is specified, but not t_f . Next, consider the *fixed-time* optimal control problem,

$$6 \quad \text{minimize} \quad \int_0^1 f^0(\tilde{x}(s), \tilde{u}(s), \tilde{t}(s)) \tilde{v}(s)^2 ds$$

subject to

$$7 \quad \frac{d}{ds} \tilde{x}(s) = \tilde{v}(s)^2 f(\tilde{x}(s), \tilde{u}(s), \tilde{t}(s)), \quad s \in [0, 1], \quad \tilde{x}(0) = \hat{x}_0,$$

$$8 \quad \frac{d}{ds} \tilde{t}(s) = \tilde{v}(s)^2, \quad s \in [0, 1], \quad \tilde{t}(0) = t_0,$$

where the functions f^0 and f are the same as in (4) and (5). Note that in the problem (6)–(8), the dynamical system is described by (7) and (8) and that its state at time s is $(\tilde{x}(s), \tilde{t}(s))$, with $\tilde{t}(\cdot)$ real-valued, while its input at time s is $(\tilde{u}(s), \tilde{v}(s))$, with $\tilde{v}(\cdot)$ real-valued. Thus, the dimension both of the state vector and of the control vector of (6)–(8) is larger (by 1) than the corresponding dimension in problem (4), (5).

Show that if $(\dot{u}(\cdot), \dot{v}(\cdot))$ is an optimal control for (6)–(8) and $(\dot{x}(\cdot), \dot{t}(\cdot))$ is the corresponding optimal trajectory, then \hat{t}_f , $\hat{u}(\cdot)$ and $\hat{x}(\cdot)$, defined by

$$9 \quad \hat{t}_f = t_0 + \int_0^1 \dot{v}(s)^2 ds,$$

$$10 \quad \hat{u} \left(t_0 + \int_0^s \dot{v}(s)^2 ds \right) = \dot{u}(s), \quad s \in [0, 1],$$

$$11 \quad \hat{x} \left(t_0 + \int_0^s \dot{v}(s)^2 ds \right) = \dot{x}(s), \quad s \in [0, 1],$$

are, respectively, an optimal time, an optimal control, and the corresponding optimal trajectory for the problem (4), (5). ■

For the problem (1), (2) to be well-defined, we must at least require that the controls $u(\cdot)$ be measurable functions. We shall make a somewhat stronger assumption; we shall suppose that the functions $u(\cdot)$ in (1), (2) belong to $L_{\infty}^{\mu}[t_0, t_f]$, the space of equivalence classes of measurable functions from $[t_0, t_f]$ into \mathbb{R}^n which have the property that

$$12 \quad \text{ess sup}_{t \in [t_0, t_f]} \|u(t)\| \leq \infty.$$

The space $L_{\infty}^{\mu}[t_0, t_f]$ is a normed linear space. We shall denote the norm of $u(\cdot) \in L_{\infty}[t_0, t_f]$ by $\|u\|_{\infty}$; we recall that

$$13 \quad \|u\|_{\infty} = \text{ess sup}_{t \in [t_0, t_f]} \|u(t)\|.$$

We can also introduce a scalar product on $L_{\infty}^{\mu}[t_0, t_f]$, which we define as

$$14 \quad \langle u_1, u_2 \rangle_{\infty} = \int_{t_0}^{t_f} \langle u_1(t), u_2(t) \rangle dt,$$

(Note that this is the standard scalar product used in the Hilbert space $L_2^{\mu}[t_0, t_f]$.)

Let $x(t; u)$ denote the solution* of (2) at time t from the given initial state x_0 at $t = t_0$ and corresponding to the input $u(\cdot)$ (i.e., $x(t_0; u) = x_0$). Then, if we define the function $F^0 : L_{\infty}^{\mu}[t_0, t_f] \rightarrow \mathbb{R}^1$ by†

$$15 \quad F^0(u) = \int_{t_0}^{t_f} f^0(x(t; u), u(t), t) dt + \varphi(x(t_f; u)),$$

we see that problem (1), (2) is equivalent to the problem (compare (1.1)),

$$16 \quad \min\{F^0(u) \mid u \in L_{\infty}^{\mu}[t_0, t_f]\}.$$

Before we can proceed any further, we must define the gradient of $F^0(\cdot)$. Suppose that $\text{grad } F^0(\cdot)(\cdot)$ is a map from $L_{\infty}^{\mu}[t_0, t_f] \times [t_0, t_f]$ into \mathbb{R}^n such that for every $u \in L_{\infty}^{\mu}[t_0, t_f]$, $\text{grad } F^0(u) \in L_{\infty}^{\mu}[t_0, t_f]$, and‡

$$17 \quad \lim_{\|v\|_{\infty} \rightarrow 0} \frac{F^0(u + v) - F^0(u) - \langle \text{grad } F^0(u), v \rangle_{\infty}}{\|v\|_{\infty}} = 0.$$

* We assume that a solution exists. It then follows from the assumption stated that the solution must be unique.

† In the calculations to follow, it is convenient to consider $x(\cdot; u)$ to be an element of the space of continuous functions from $[t_0, t_f]$ into \mathbb{R}^n , with the sup norm.

‡ For our purposes, this simplified approach to gradients is adequate. In general, gradients must be discussed in terms of differentials. For a proof of the existence of $\text{grad } F^0(u)$, see [D3], theorem 10.7.3.

Then we say that $\text{grad } F^0(\cdot)(\cdot)$ is the gradient of $F^0(\cdot)$. Note that if $\text{grad } F^0(u)(\cdot) \neq 0$, then, because of (17), there must exist an $s' > 0$ such that for all $s \in (0, s']$,

$$18 \quad F^0(u - s \text{ grad } F^0(u)) - F^0(u) < 0,$$

i.e., whenever $\text{grad } F^0(u)(\cdot) \neq 0$, it defines a direction of descent (decrease), just as in the finite dimensional case, and it is not difficult to see that this is also the direction of *steepest* descent. It is also easy to see that, because of (18), if $\hat{u}(\cdot)$ minimizes $F^0(u)$ over $u \in L_\infty^\mu[t_0, t_f]$, then we must have $\text{grad } F^0(\hat{u}) = 0$, exactly as in the finite dimensional case.

We shall now obtain formulas for computing $\text{grad } F^0(u)(\cdot)$. We could proceed in exactly the same manner as in Section 4. However, to demonstrate an alternative approach, we shall follow the procedure suggested in exercise (4.18). Thus, we shall compute $F^0(u + \delta u) - F^0(u)$ to first-order terms, formally, by linearizing all the functions in (1) and (2) about a given control function $u(\cdot)$ and its corresponding trajectory $x(\cdot)$, and we shall then obtain $\text{grad } F^0(u)(\cdot)$ by inspection, since to first-order terms,

$$19 \quad F^0(u + \delta u) - F^0(u) = \langle \text{grad } F^0(u), \delta u \rangle_\infty.$$

Linearizing all the functions in (1) and (2) about the nominal control $u(\cdot)$ and its corresponding trajectory $x(\cdot, u)$, we obtain

$$20 \quad F^0(u + \delta u) - F^0(u) = \int_{t_0}^{t_f} \left[\frac{\partial f^0(x(t; u), u(t), t)}{\partial x} \delta x(t) + \frac{\partial f^0(x(t; u), u(t), t)}{\partial u} \delta u(t) \right] dt + \frac{\partial \varphi(x(t_f; u))}{\partial x} \delta x(t_f),$$

where $\delta x(t)$, $t \in [t_0, t_f]$, is computed by solving the linearized differential equation,

$$21 \quad \frac{d}{dt} \delta x(t) = \frac{\partial f(x(t; u), u(t), t)}{\partial x} \delta x(t) + \frac{\partial f(x(t; u), u(t), t)}{\partial u} \delta u(t),$$

for $t \in [t_0, t_f]$, with the initial condition $\delta x(t_0) = 0$. Now, for $s, t \in [t_0, t_f]$, let $\Phi(t, s)$ be a $\nu \times \nu$ matrix determined by the differential equation,

$$22 \quad \frac{d}{dt} \Phi(t, s) = \frac{\partial f(x(t; u), u(t), t)}{\partial x} \Phi(t, s), \quad \Phi(s, s) = I.$$

where I is the $\nu \times \nu$ identity matrix. Then we see that

$$23 \quad \delta x(t) = \int_{t_0}^t \Phi(t, s) \frac{\partial f(x(s; u), u(s), s)}{\partial u} \delta u(s) ds, \quad t \in [t_0, t_f],$$

and hence (20) becomes

$$\begin{aligned}
 24 \quad F^0(u + \delta u) - F^0(u) &= \int_{t_0}^{t_f} \int_{t_0}^t \frac{\partial f^0(x(t; u), u(t), t)}{\partial x} \\
 &\quad \times \Phi(t, s) \frac{\partial f(x(s; u), u(s), s)}{\partial u} \delta u(s) ds dt \\
 &\quad + \int_{t_0}^{t_f} \frac{\partial f^0(x(s; u), u(s), s)}{\partial u} \delta u(s) ds \\
 &\quad + \int_{t_0}^{t_f} \frac{\partial \varphi(x(t_f; u)}{\partial x} \Phi(t_f, s) \frac{\partial f(x(s; u), u(s), s)}{\partial u} \delta u(s) ds.
 \end{aligned}$$

Interchanging the order of integration in the first term in (24), we obtain,

$$\begin{aligned}
 25 \quad & \int_{t_0}^{t_f} \int_{t_0}^t \frac{\partial f^0(x(t; u), u(t), t)}{\partial x} \Phi(t, s) \frac{\partial f(x(s; u), u(s), s)}{\partial u} \delta u(s) ds dt \\
 &= \int_{t_0}^{t_f} \left[\int_s^{t_f} \frac{\partial f^0(x(t; u), u(t), t)}{\partial x} \Phi(t, s) dt \right] \frac{\partial f(x(s; u), u(s), s)}{\partial u} \delta u(s) ds.
 \end{aligned}$$

Now, for $t \in [t_0, t_f]$, let $p(t) \in \mathbb{R}^n$ be determined by the differential equation (compare (1.2.36)),

$$26 \quad \frac{d}{dt} p(t)^T = -p(t)^T \frac{\partial f(x(t; u), u(t), t)}{\partial x} + \frac{\partial f^0(x(t; u), u(t), t)}{\partial x},$$

with $p(t_f)^T = -\partial \varphi(x(t_f; u)) / \partial x$. Then it is easy to show that for any $s \in [t_0, t_f]$,

$$27 \quad p(s)^T = -\frac{\partial \varphi(x(t_f; u))}{\partial x} \Phi(t_f, s) + \int_{t_f}^s \frac{\partial f^0(x(t; u), u(t), t)}{\partial x} \Phi(t, s) dt.$$

Comparing (27) with (24) and (25), we conclude that

$$\begin{aligned}
 28 \quad F^0(u + \delta u) - F^0(u) &= \int_{t_0}^{t_f} \left[-p(s)^T \frac{\partial f(x(s; u), u(s), s)}{\partial u} + \frac{\partial f^0(x(s; u), u(s), s)}{\partial u} \right] \delta u(s) ds,
 \end{aligned}$$

to first-order terms. Finally, comparing (28) and (19), we conclude that for $t \in [t_0, t_f]$, $\text{grad } F^0(u)(\cdot)$ is defined by

$$29 \quad \text{grad } F^0(u)(t) = -\left(\frac{\partial f(x(t; u), u(t), t)}{\partial u} \right)^T p(t) + \left(\frac{\partial f^0(x(t; u), u(t), t)}{\partial u} \right)^T$$

We shall now show how some of the algorithms described in Sections 1 and 3 can be applied to problem (1).

- 30 **Algorithm** (gradient method, compare (1.19)). Let $\alpha \in (0, \frac{1}{2})$ be given.

Step 0. Select a $\overset{0}{\overset{i}{u}}(\cdot) \in L_{\infty}^{\mu}[t_0, t_f]$.

Step 1. Set $i = 0$.

Comment. To facilitate comparisons in subsequent sections, we write $\overset{i}{u}, \overset{i}{x}, \overset{i}{p}$, instead of u_i, x_i, p_i .

Step 2. Compute $\overset{i}{x}(t)$, $t \in [t_0, t_f]$, by solving (5) with $u(t) = \overset{i}{u}(t)$, $t \in [t_0, t_f]$.

Step 3. Compute $\overset{i}{p}(t)$, $t \in [t_0, t_f]$, by solving (26) with $u(t) = \overset{i}{u}(t)$ and $x(t) = \overset{i}{x}(t)$, $t \in [t_0, t_f]$.

Step 4. Compute, for $t \in [t_0, t_f]$,

$$31 \quad \text{grad } F^0(\overset{i}{u})(t) = - \left(\frac{\partial f(\overset{i}{x}(t), \overset{i}{u}(t), t)}{\partial u} \right)^T \overset{i}{p}(t) + \left(\frac{\partial f^0(\overset{i}{x}(t), \overset{i}{u}(t), t)}{\partial u} \right)^T.$$

Step 5. For $t \in [t_0, t_f]$, set $h(\overset{i}{u})(t) = -\text{grad } F^0(\overset{i}{u})(t)$. If $h(\overset{i}{u})(\cdot) = 0$, stop; else, go to step 6.

Step 6. Compute a $\lambda_i > 0$ such that

$$32 \quad -\lambda_i(1 - \alpha) \|\text{grad } F^0(\overset{i}{u})\|_2^2 \leq \theta(\lambda_i; \overset{i}{u}) \leq -\lambda_i \alpha \|\text{grad } F^0(\overset{i}{u})\|_2^2,$$

where

$$33 \quad \theta(\lambda_i; \overset{i}{u}) = F^0(\overset{i}{u} + \lambda_i h(\overset{i}{u})) - F^0(\overset{i}{u}).$$

Comment. A λ_i satisfying (32) can be computed by an obvious adaptation of algorithm (1.33), since we assume that $\min\{F^0(u) \mid u \in L_{\infty}^{\mu}[t_0, t_f]\}$ is finite.

Step 7. For $t \in [t_0, t_f]$, set $\overset{i+1}{u}(t) = \overset{i}{u}(t) + \lambda_i h(\overset{i}{u})(t)$, set $i = i + 1$, and go to step 2. ■

- 34 **Exercise.** Show that either the sequence of controls $\overset{i}{u}(\cdot)$ constructed by algorithm (30) is finite, terminating at $\overset{k}{u}(\cdot)$, and $\text{grad } F^0(\overset{k}{u})(\cdot) = 0$, or else it is infinite and then every accumulation point $\hat{u}(\cdot)$ of $\{\overset{i}{u}(\cdot)\}$ satisfies $\text{grad } F^0(\hat{u})(\cdot) = 0$. [Hint: Adapt the proof of theorem (1.22).] ■

- 35 **Exercise.** Suppose that problem (1) has the particular form

$$36 \quad \text{minimize} \quad \frac{1}{2} \int_{t_0}^{t_f} \langle x(t), R(t) x(t) \rangle + \langle u(t), Q(t) u(t) \rangle dt,$$

subject to

$$37 \quad \frac{d}{dt} x(t) = A(t) x(t) + B(t) u(t), \quad t \in [t_0, t_f], \quad x(t_0) = \dot{x}_0,$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $R(t)$, $Q(t)$ are symmetric positive definite matrices for all $t \in [t_0, t_f]$, and the matrix-valued maps $R(\cdot)$, $Q(\cdot)$, $A(\cdot)$, $B(\cdot)$ are all continuous. Develop an algorithm of the form (1.16) for this problem. Justify the use of your algorithm in solving (36), (37) and show that the algorithm will produce a sequence of controls that actually converges to the optimal control for this problem by showing that the conclusions indicated in (1.18) hold for this case also. ■

Problem (36), (37) can also be solved by conjugate gradient algorithms. We illustrate this by writing out an adaptation of the Fletcher-Reeves algorithm (3.42).

38 **Algorithm** (Fletcher-Reeves for (36), (37)).

Step 0. Select a $\overset{0}{u}(\cdot) \in L_\infty^m[t_0, t_f]$.

Step 1. Compute $\overset{0}{x}(t)$ for $t \in [t_0, t_f]$ by solving (37) with $u(\cdot) = \overset{0}{u}(\cdot)$.

Step 2. Compute $\overset{0}{p}(t)$ for $t \in [t_0, t_f]$ by solving

$$39 \quad \frac{d}{dt} \overset{0}{p}(t)^T = -\overset{0}{p}(t)^T A(t) + \overset{0}{x}(t)^T R(t), \quad \text{with } \overset{0}{p}(t_f)^T = 0.$$

Step 3. Compute

$$40 \quad \text{grad } F^0(\overset{0}{u})(t) = -B(t)^T \overset{0}{p}(t) + Q(t) \overset{0}{u}(t), \quad t \in [t_0, t_f].$$

Step 4. For $t \in [t_0, t_f]$, set $\overset{0}{g}(t) = \overset{0}{h}(t) = -\text{grad } F^0(\overset{0}{u})(t)$.

Comment. The algorithm is now initialized.

Step 5. If $\overset{0}{g}(\cdot) = 0$, stop; else, set $i = 0$ and go to step 6.

Step 6. Compute a $\lambda_i > 0$ such that

$$41 \quad F^0(\overset{i}{u} + \lambda_i \overset{i}{h}) = \min\{F^0(\overset{i}{u} + \lambda \overset{i}{h}) \mid \lambda > 0\}.$$

Step 7. For $t \in [t_0, t_f]$, set $\overset{i+1}{u}(t) = \overset{i}{u}(t) + \lambda_i \overset{i}{h}(t)$.

Step 8. Compute $\overset{i+1}{x}(t)$ for $t \in [t_0, t_f]$ by solving (37) with $u(\cdot) = \overset{i+1}{u}(\cdot)$.

Step 9. Compute $\overset{i+1}{p}(t)$ for $t \in [t_0, t_f]$ by solving

$$42 \quad \frac{d}{dt} \overset{i+1}{p}(t)^T = -\overset{i+1}{p}(t)^T A(t) + \overset{i+1}{x}(t)^T R(t), \quad \text{with } \overset{i+1}{p}(t_f)^T = 0.$$

Step 10. Compute

$$43 \quad \text{grad } F^0(\overset{i+1}{u})(t) = -B(t)^T \overset{i+1}{p}(t) + Q(t) \overset{i+1}{u}(t), \quad t \in [t_0, t_f]$$

Step 11. If $\text{grad } F^0(\overset{i+1}{u})(\cdot) = 0$, stop; else, set

$$44 \quad \overset{i+1}{g}(t) = -\text{grad } F^0(\overset{i+1}{u})(t), \quad t \in [t_0, t_f],$$

$$45 \quad \overset{i+1}{h}(t) = \overset{i+1}{g}(t) + \gamma_i \overset{i}{h}(t), \quad \text{with } \gamma_i = \| \overset{i+1}{g} \|_2^2 / \| \overset{i}{g} \|_2^2, *$$

set $i = i + 1$, and go to step 6. ■

46 Exercise. Obtain formulas which will enable you to apply the Fletcher-Reeves algorithm to the general unconstrained optimal control problem (1), (2). ■

47 Remark. To apply the Polak-Ribière algorithm (2.51) to the problem (36), (37), we only need to change the formula for γ_i in (45) to

$$48 \quad \gamma_i = \frac{1}{\| \overset{i}{g} \|_2^2} \int_{t_0}^{t_f} \langle \overset{i+1}{g}(t) - \overset{i}{g}(t), \overset{i}{g}(t) \rangle dt. \quad ■$$

* Recall that $\| \overset{i}{g} \|_2^2 = \int_{t_0}^{t_f} \| \overset{i}{g}(t) \|^2 dt$.

3

EQUALITY CONSTRAINTS: ROOT AND BOUNDARY-VALUE PROBLEMS

3.1 Zeros of a Function and Problems with Equality Constraints in \mathbb{R}^n

We shall now consider briefly the problem of finding a vector $z \in \mathbb{R}^n$ such that $g(z) = 0$, where $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is a continuously differentiable function. Obviously, we can convert the problem of finding the roots of the equation

$$1 \quad g(z) = 0$$

into the form

$$2 \quad \min\{f^0(z) \triangleq \frac{1}{2} \|g(z)\|^2 \mid z \in \mathbb{R}^n\},$$

and then apply any one of the minimization algorithms discussed in Section 2.1. When the function $f^0(\cdot)$ is defined as in (2), the “direction vector” $h(z)$, which appears in the algorithms discussed in Section 2.1, is given by the formula

$$3 \quad \begin{aligned} h(z) &= -D(z) \nabla f^0(z) \\ &= -D(z) \left(\frac{\partial g(z)}{\partial z} \right)^T g(z), \end{aligned}$$

where $\partial g(z)/\partial z$ is an $m \times n$ matrix whose ij th element is $\partial g^i(z)/\partial z^j$. Now, all the algorithms we have discussed stop at points z' such that $h(z') = 0$. Since by assumption, the matrix $D(z)$ is positive definite for all $z \in \{z \mid f^0(z) \leq f^0(z_0)\}$,

where z_0 is our initial guess at a root of (1), we conclude that the minimization algorithms of Section 2.1 can only be used to compute points z' such that

$$4 \quad \left(\frac{\partial g(z')}{\partial z} \right)^T g(z') = 0.$$

Thus, we shall be able to compute the roots of equation (1) provided a point z' satisfies (4) if and only if $g(z') = 0$. A sufficient condition for this to be true is that the rank of $\partial g(z')/\partial z$ be maximal for all z' such that $g(z') = 0$.

To apply the variants of the method of steepest descent to (2), we set $D(z) = I$, the identity matrix, in (3). To apply any one of the quasi-Newton methods to (2), we must assume that the function $f^0(\cdot)$ in (2) is twice continuously differentiable and strictly convex, with one important exception. This exception arises when the dimensions of the domain and range spaces of the function $g(\cdot)$ are the same, i.e., $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$. In this case, we only need to assume that the Jacobian matrix $\partial g(z)/\partial z$ is nonsingular for all $z \in \{z \mid f^0(z) \leq f^0(z_0)\}$, where z_0 is our initial guess at a root of (1). We then set

$$5 \quad D(z) = \left(\frac{\partial g(z)}{\partial z} \right)^{-1} \left(\left(\frac{\partial g(z)}{\partial z} \right)^T \right)^{-1},$$

which is positive definite by inspection, and obtain from (3),

$$6 \quad h(z) = - \left(\frac{\partial g(z)}{\partial z} \right)^{-1} g(z).$$

Note that when $g(z) = \nabla f^0(z)$, with $\tilde{f}^0(\cdot)$ a possibly nonconvex function whose stationary points we wish to find, we obtain, formally, from (6),

$$h(z) = - \left(\frac{\partial^2 \tilde{f}^0(z)}{\partial z^2} \right)^{-1} \nabla \tilde{f}^0(z),$$

which is exactly the same as given by (2.1.38).

We should like to point out that just as (2.1.38) could be derived by heuristic considerations using second-order expansions about a point z_i , so can (6) be obtained by means of first-order expansions about a point z_i . We shall now show how this is done. Suppose that we are at a point $z_i \in \mathbb{R}^n$. Expanding (1) to first-order terms about z_i , we obtain

$$7 \quad g(z) \doteq g(z_i) + \frac{\partial g(z_i)}{\partial z} (z - z_i) = 0.$$

Solving (7) for z_{i+1} , we now obtain

$$8 \quad z_{i+1} = z_i - \left(\frac{\partial g(z_i)}{\partial z} \right)^{-1} g(z_i), \quad i = 0, 1, 2, \dots,$$

which is exactly the same as (2.1.38) for $g(z) = \nabla f^0(z)$.

Hence, for solving the problem of finding a $z \in \mathbb{R}^n$ such that $g(z) = 0$, where $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a continuously differentiable function with a non-singular Jacobian, the Newton-Raphson method assumes the following form:

- 9 **Algorithm** (Newton-Raphson). Finds zeros of $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$, provided $(\partial g(z)/\partial z)^{-1}$ exists and is continuous.*

Step 0. Select a $z_0 \in \mathbb{R}^n$.

Step 1. Set $i = 0$.

Step 2. Compute $g(z_i)$.

Step 3. If $g(z_i) = 0$, stop; else, compute z_{i+1} according to (8), set $i = i + 1$, and go to step 2. ■

We shall now examine the problem of solving systems of equations which arise as part of necessary conditions of optimality. Thus, let us consider

$$10 \quad \min\{f^0(z) \mid r(z) = 0\},$$

where $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ and $r : \mathbb{R}^n \rightarrow \mathbb{R}^l$ are twice continuously differentiable functions. For problem (10), theorem (1.2.1) states that if \hat{z} is optimal for (10), then there exist multipliers $\hat{\mu}^0, \hat{\psi}^1, \dots, \hat{\psi}^l$, not all zero, such that

$$11 \quad \hat{\mu}^0 \nabla f^0(\hat{z}) + \sum_{i=1}^l \hat{\psi}^i \nabla r^i(\hat{z}) = 0,$$

and since \hat{z} must also be *feasible* (i.e., it must satisfy the constraints in (10)), we have in addition,

$$12 \quad r(\hat{z}) = 0.$$

(Note that the condition $\hat{\mu}^0 \leq 0$ in (1.2.1) loses its significance when there are no inequality constraints and is therefore omitted.)

We can use algorithms in the class described previously to find a vector \hat{z} and multipliers $\hat{\mu}^0, \hat{\psi}^1, \dots, \hat{\psi}^l$ satisfying (11) and (12) (with not all multipliers zero) as follows: There are two possibilities. Either we *must* set $\hat{\mu}^0 = 0$ in (11) or not. If we *must* set $\hat{\mu}^0 = 0$, then, setting $\psi = (\psi^1, \psi^2, \dots, \psi^l)$, $x = (z, \psi)$ and defining $g : \mathbb{R}^n \times \mathbb{R}^l \rightarrow \mathbb{R}^n \times \mathbb{R}^l$ as

$$13 \quad g(x) = \begin{pmatrix} \sum_{i=1}^l \psi^i \nabla r^i(z) \\ r(z) \end{pmatrix},$$

we can try to compute points \hat{x} satisfying $g(\hat{x}) = 0$ by means of the quasi-Newton methods or gradient methods that we have discussed in Section 2.1. Because of their much faster convergence, the quasi-Newton methods may

* It follows by continuity from (8) that if a sequence $\{z_i\}_{i=0}^\infty$ constructed by algorithm (9) converges to a point \hat{z} , then $g(\hat{z}) = 0$.

be preferred for this work, since they require no more function evaluations than the gradient methods and only an additional matrix inversion which can be carried out by solving a system of equations. We see this by comparing the formulas for $h(z)$ as given by (3), with $D(z) = I$, the identity matrix, and (6).

The second possibility is that (11) and (12) can be satisfied with $\hat{\mu}^0 \neq 0$. Then we may obviously set $\hat{\mu}^0 = -1$ and, with x defined as above, define the function $g : \mathbb{R}^n \times \mathbb{R}^l \rightarrow \mathbb{R}^n \times \mathbb{R}^l$ as

$$14 \quad g(x) = \begin{pmatrix} \sum_{i=1}^l \psi^i \nabla r^i(z) - \nabla f^0(z) \\ r(z) \end{pmatrix}$$

and again we can try to solve (14) by the methods we have discussed so far.

However, we do not, generally, know in advance whether we are in a situation characterized by the choice (13) or (14). Hence, we may proceed as follows: First we attempt to solve the equation $g(x) = 0$, with $g(\cdot)$ defined as in (14), by minimizing $\|g(x)\|^2$ over $\mathbb{R}^n \times \mathbb{R}^l$. If we find that the algorithms described in Section 2.1 construct a sequence of vectors x_i such that $\|g(x_i)\|^2$ does not appear to converge to zero, then we change over to the function $g(\cdot)$ defined by (13) and continue.

There is at least one case where we can tell for sure that $\hat{\mu}^0$ must be -1 . This is the case where the gradient vectors $\nabla r^i(\hat{z})$ in (11) are linearly independent (in the region of search).^{*} Another obvious case when $\hat{\mu}^0$ may be set to -1 occurs when the function $r(\cdot)$ is affine (i.e., $r(z) = Az + b$, where A is an $l \times n$ matrix and b is a vector in \mathbb{R}^l).

- 15 **Remark.** Suppose that for all $z \in \mathbb{R}^n$ such that $r(z) = 0$, the gradient vectors $\nabla r^i(z)$, $i = 1, 2, \dots, l$, are linearly *dependent*. Then for any z satisfying $r(z) = 0$ we can satisfy (11) after setting $\hat{\mu}^0 = 0$, and hence in such a case, we see that theorem (1.2.1) is trivial, since its conditions can be satisfied without any reference to optimality of the candidate vector z satisfying $r(z) = 0$. In such a case, one would do just as well by simply solving the equation $r(z) = 0$. ■

Before we leave the subject of solving systems of equations arising from optimality conditions, we wish to point out an interesting fact. Consider the nonlinear programming problem

$$16 \quad \min\{f^0(z) \mid r^i(z) = 0, i = 1, 2, \dots, l; f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

with $z \in \mathbb{R}^n$,

* In this case, $\hat{\psi} = ((\partial r(\hat{z})/\partial z)(\partial r(\hat{z})/\partial z)^T)^{-1}(\partial r(\hat{z})/\partial z)\nabla f^0(\hat{z})$, and hence we can reduce the number of variables in (14) at the expense of making $g(\cdot)$ more complex.

first defined in (1.1.1), and suppose that all the functions in (16) are continuously differentiable. This problem can always be transcribed into the form (10) as follows: For $i = 1, 2, \dots, m$, let

$$17 \quad r^{l+i}(z) = (\max\{0, f^i(z)\})^2.$$

Then $r^{l+i}(z) = 0$ if and only if $f^i(z) \leq 0$, $i = 1, 2, \dots, m$, and therefore (16) is equivalent to the problem

$$18 \quad \min\{f^0(z) \mid r^i(z) = 0, i = 1, 2, \dots, l + m\}.$$

However, for any z in the set $\{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}$ which also satisfies $r^i(z) = 0$ for $i = 1, 2, \dots, l$, we find that $\nabla r^i(z) = 0$ for $i = l + 1, l + 2, \dots, l + m$. Hence, for all such z , condition (11) can be satisfied trivially and we therefore conclude that the transcription of (16) into the form (18), which was originally motivated by our desire to solve (16) by means of the methods presented so far, is completely unproductive. Methods for solving (16) will be presented in the next chapter.

3.2 Boundary-Value Problems and Discrete Optimal Control

This section is devoted to the use of the Newton-Raphson method (2.1.39) (in the version defined by (1.9)) and of its modifications, such as (2.1.42), in the solution of boundary-value problems with discrete dynamics. We shall consider two general classes of problems. The first class of boundary-value problems is quite general and is not specifically identifiable with discrete optimal control problems. The second class of boundary-value problems to be considered arises as a necessary condition of optimality in discrete optimal control problems and has an additional amount of structure which can be exploited in its solution.

Thus, suppose that we are given a discrete dynamical system

$$1 \quad x_{i+1} - x_i = f_i(x_i), \quad i = 0, 1, 2, \dots, k - 1, \quad x_i \in \mathbb{R}^\nu,$$

where the $f_i(\cdot)$ are continuously differentiable functions, together with the boundary conditions

$$2 \quad P_0 x_0 = y_0, \quad P_k x_k = y_k,$$

where $y_0 \in \mathbb{R}^\alpha$, $y_k \in \mathbb{R}^{\nu-\alpha}$, $\alpha < \nu$, and P_0, P_k are full-rank matrices of dimension $\alpha \times \nu$ and $(\nu - \alpha) \times \nu$ respectively. To apply the Newton-Raphson

method, in the version (1.9), to the above boundary-value problem, we must first transcribe (1) and (2) into the form

$$3 \quad g(z) = 0.$$

We have already seen in (1.1.13) and (1.1.17) two ways in which this can be done. We shall do it both ways and show that each of the above-mentioned transcriptions leads to an entirely different implementation of the Newton-Raphson method. The first of these implementations is sometimes referred to as the *Goodman-Lance method* [G4]. The second one of these implementations is usually referred to as *quasi-linearization*. This version appears to have been first proposed by McGill and Kenneth [M2] and subsequently considerably developed by Bellman and Kalaba [B6]. (See also Lee [L1].)

We begin with the Goodman-Lance version. For this purpose, we set (compare (1.1.17))

$$4 \quad z = x_0.$$

Then, solving (1) with $x_0 = z$, we obtain x_k as a function of z , i.e., we obtain $x_k(z)$. The boundary conditions (2) can now be used to define the function $g(\cdot)$ as follows:

$$5 \quad g(z) = \begin{pmatrix} P_k x_k(z) - y_k \\ P_0 z - y_0 \end{pmatrix}.$$

It is evident from (5) that $g : \mathbb{R}^v \rightarrow \mathbb{R}^v$, i.e., it satisfies the dimension requirements for the Newton-Raphson method to apply. From now on we shall proceed formally, under the assumption that $(\partial g(z)/\partial z)^{-1}$ exists for all $z \in \mathbb{R}^v$ (or at least in a sufficiently large open set in \mathbb{R}^v). Generally speaking, this assumption cannot be verified a priori. As stated in (1.9), given a point $z_j \in \mathbb{R}^v$, the Newton-Raphson method computes the next point z_{j+1} , according to the formula

$$6 \quad z_{j+1} = z_j - \left(\frac{\partial g(z_j)}{\partial z} \right)^{-1} g(z_j),$$

i.e., it requires us to solve the equation

$$7 \quad \frac{\partial g(z_j)}{\partial z} z_{j+1} = \left(\frac{\partial g(z_j)}{\partial z} \right) z_j - g(z_j).$$

Since in the case under consideration, we already use subscripts on the x_i , $i = 0, 1, \dots, k$, we depart from the notation we have used until now and move the subscript j in z_j to the position \dot{z} (leaving the superscript position to indicate components of a vector, as usual). Hence, we shall write (7) as

$$8 \quad \frac{\partial g(\dot{z})}{\partial z} \dot{z}^{j+1} = \frac{\partial g(\dot{z})}{\partial z} \dot{z}^j - g(\dot{z})$$

From (5),

$$9 \quad \frac{\partial g(\vec{z})}{\partial z} = \begin{pmatrix} P_k \frac{\partial x_k(\vec{z})}{\partial z} \\ P_0 \end{pmatrix}.$$

Substituting from (9) into (8), we find that at the j th iteration we must solve the system of equations

$$10 \quad P_k \frac{\partial x_k(\vec{z})}{\partial z} \vec{z}^{j+1} = P_k \frac{\partial x_k(\vec{z})}{\partial z} \vec{z}^j - [P_k x_k(\vec{z}) - y_k],$$

$$11 \quad P_0 \vec{z}^{j+1} = P_0 \vec{z}^j - (P_0 \vec{z}^j - y_0) = y_0.$$

12 **Exercise.** Suppose that $\vec{x}_i = x_i(\vec{z})$, $i = 0, 1, 2, \dots, k$, is the solution of (1) at time i corresponding to the initial state $x_0 = \vec{z}$, and suppose that the matrices $\vec{\Phi}_{i,s}$, with $s = 0, 1, 2, \dots, k$ and $i = s, s+1, \dots, k$, are obtained by solving the matrix difference equation

$$13 \quad \vec{\Phi}_{i+1,s} - \vec{\Phi}_{i,s} = \frac{\partial f_i(\vec{x}_i)}{\partial x_i} \vec{\Phi}_{i,s}, \quad i = s, s+1, s+2, \dots, k-1,$$

with $\vec{\Phi}_{s,s} = I$, the $\nu \times \nu$ identity matrix. Show that

$$14 \quad \frac{\partial x_k(\vec{z})}{\partial z} = \vec{\Phi}_{k,0}.$$

Also show that $\vec{\Phi}_{i,s}$ satisfies the adjoint equation

$$15 \quad \vec{\Phi}_{i,s} - \vec{\Phi}_{i,s+1} = \vec{\Phi}_{i,s+1} \frac{\partial f_s(\vec{x}_s)}{\partial x_s}, \quad s = i-1, i-2, \dots, 0. \blacksquare$$

Substituting from (14) into (10), we now obtain

$$16 \quad P_k \vec{\Phi}_{k,0} \vec{z}^{j+1} = P_k \vec{\Phi}_{k,0} \vec{z}^j - P_k x_k(\vec{z}) + y_k.$$

We can, of course, calculate $\vec{\Phi}_{k,0}$ by solving (13) and then compute $P_k \vec{\Phi}_{k,0}$, however, it is possible to save some labor in carrying out this task by making use of (15), as follows: Let \vec{P}_i , $i = 0, 1, 2, \dots, k$, be $(\nu - \alpha) \times \nu$ matrices determined by

$$17 \quad \vec{P}_i - \vec{P}_{i+1} = \vec{P}_{i+1} \frac{\partial f_i(\vec{x}_i)}{\partial x_i}, \quad i = 0, 1, \dots, k-1, \quad \vec{P}_k = P_k.$$

Then we see that $P_k \vec{\Phi}_{k,0} = \vec{P}_0$. At this point we note that (17) is a difference

equation in fewer variables than (13) and hence requires fewer multiplications and therefore also less time to solve.

We now summarize the above discussion by presenting it in the form of an algorithm.

- 18 **Algorithm** (Goodman-Lance version of Newton-Raphson method [G4]).

Step 0. Select a $\overset{0}{z} \in \mathbb{R}^\nu$.

Step 1. Set $j = 0$.

Step 2. Solve (1) with $x_0 = \overset{j}{z}$ for the states $\overset{j}{x}_i = x_i(\overset{j}{z})$, $i = 0, 1, \dots, k$.

Step 3. Compute the Jacobian matrices

$$\frac{\partial f_i(\overset{j}{x}_i)}{\partial x_i}, \quad i = 0, 1, 2, \dots, k - 1.$$

Step 4. Compute $\overset{j}{P}_0$ by solving (17).

Step 5. Compute $\overset{j+1}{z}$ by solving

$$\overset{j}{P}_0 \overset{j+1}{z} = \overset{j}{P}_0 \overset{j}{z} - P_k \overset{j}{x}_k + y_k,$$

$$\overset{j+1}{P}_0 = y_0.$$

Step 6. Set $j = j + 1$ and go to step 2. ■

Thus, in the Goodman-Lance version of the Newton-Raphson method, at each iteration, we must solve the nonlinear difference equation (1) in the forward direction from the current initial state $x_0 = \overset{j}{z}$. Then we must solve the linear, variational adjoint equation (17) in the backward direction from the given initial condition $P_k = \overset{j}{P}_k$. Finally, we must solve the system of equations (19), (20).

- 21 **Remark.** Assuming that the adjoint system (17) is stable in the backward direction (i.e., $\overset{i}{\Phi}_{k,i} \rightarrow 0$ as $i \rightarrow 0$), the nonzero elements of $\overset{j}{P}_0$ may be quite small in comparison with the nonzero elements of P_0 , and hence it is possible for the system of equations (19), (20) to be badly ill-conditioned, especially when k is large. There appear to be no general procedures for resolving this difficulty at the present time. In some cases, however, the Abramov method [A1], to be presented later, can be utilized to reduce or eliminate this ill-conditioning effect. ■

- 22 **Exercise.** Modify algorithm (18) to make it applicable to solving (1) with boundary conditions of the form $g_0(x_0) = g_k(x_k) = 0$, where $g_0 : \mathbb{R}^\nu \rightarrow \mathbb{R}^\alpha$, $g_k : \mathbb{R}^\nu \rightarrow \mathbb{R}^{\nu-\alpha}$ are continuously differentiable functions whose Jacobian matrices $\partial g_0(x)/\partial x$ and $\partial g_k(x)/\partial x$ have full rank for x in a sufficiently large open set in \mathbb{R}^ν . ■

- 23 **Exercise.** Modify the Goodman-Lance version (18) of the Newton-Raphson method to obtain a quasi-Newton method of the form of algorithm (2.1.42). ■

To obtain the quasi-linearization version of the Newton-Raphson method, we let (see (1.1.13))

$$24 \quad z = (x_0, x_1, \dots, x_k),$$

and treat it as a column vector, as before. Then, from (1) and (2), we see that $g(\cdot)$ is defined by

$$25 \quad g(z) = \begin{pmatrix} x_1 - x_0 - f_0(x_0) \\ \vdots \\ x_k - x_{k-1} - f_{k-1}(x_{k-1}) \\ P_0 x_0 - y_0 \\ P_k x_k - y_k \end{pmatrix}.$$

Differentiating (25) in blocks of rows at a time, we obtain, upon substitution into (8), that

$$26 \quad \dot{x}_{i+1}^{j+1} - \dot{x}_i^{j+1} - \frac{\partial f_i(\dot{x}_i)}{\partial x_i} \dot{x}_i^{j+1} = \left(\dot{x}_{i+1}^j - \dot{x}_i^j - \frac{\partial f_i(\dot{x}_i)}{\partial x_i} \dot{x}_i^j \right) - [\dot{x}_{i+1}^j - \dot{x}_i^j - f_i(\dot{x}_i)], \quad i = 0, 1, \dots, k-1,$$

$$27 \quad P_0 \ddot{x}_0^{j+1} = P_0 \ddot{x}_0^j - (P_0 \dot{x}_0^j - y_0),$$

$$28 \quad P_k \ddot{x}_k^{j+1} = P_k \ddot{x}_k^j - (P_k \dot{x}_k^j - y_k).$$

Cancelling out and rearranging terms, we finally obtain

$$29 \quad \dot{x}_{i+1}^{j+1} - \dot{x}_i^{j+1} = \frac{\partial f_i(\dot{x}_i)}{\partial x_i} \dot{x}_i^{j+1} - \frac{\partial f_i(\dot{x}_i)}{\partial x_i} \dot{x}_i^j + f_i(\dot{x}_i), \quad i = 0, 1, \dots, k-1,$$

$$30 \quad P_0 \ddot{x}_0^{j+1} = y_0, \quad P_k \ddot{x}_k^{j+1} = y_k.$$

We now summarize the above discussion in the form of an algorithm.

- 31 **Algorithm** (quasi-linearization version of Newton-Raphson method [M2], [B6]).

Step 0. Select a $\overset{0}{z} = (\overset{0}{x}_0, \overset{0}{x}_1, \dots, \overset{0}{x}_k)$ such that $P_0 \overset{0}{x}_0 = y_0$, $P_k \overset{0}{x}_k = y_k$.

Comment. We choose to set $P_0 \overset{0}{x}_0 = y_0$, $P_k \overset{0}{x}_k = y_k$ only to make the initial guess better. Other choices are admissible. When we choose $\overset{0}{x}_0, \overset{0}{x}_k$ as in step 0, we might set $\overset{0}{x}_i = \overset{0}{x}_0 + (i/k)(\overset{0}{x}_k - \overset{0}{x}_0)$ for $i = 1, 2, \dots, k-1$.

Step 1. Set $j = 0$.

Step 2. Compute $f_i(\bar{x}_i)$, $\partial f_i(\bar{x}_i)/\partial x_i$ for $i = 0, 1, 2, \dots, k - 1$.

Step 3. Solve (29), (30) for $\bar{z}^{j+1} = (\bar{x}_0^{j+1}, \bar{x}_1^{j+1}, \dots, \bar{x}_k^{j+1})$.

Step 4. Set $j = j + 1$ and go to step 2. ■

32 Remark. Note that in the Goodman-Lance version (18) of the Newton-Raphson method we always have a sequence \bar{x}_i^j , $i = 0, 1, 2, \dots, k$, which satisfies the difference equation (1), but not the boundary conditions (2), while in the quasi-linearization version (31) we always have a sequence \bar{x}_i^j , $i = 0, 1, 2, \dots, k - 1$, which satisfies the boundary conditions (2), but not the difference equation (1). Since the labor of computing $f_i(\bar{x}_i^j)$, $i = 0, 1, \dots, k - 1$, in step 2 of (31) is comparable to the labor involved in solving (1) as required in step 2 of (18), we see that the two versions, as far as difference equations are concerned, are reasonably equal from the point of view of computational efficiency. In choosing between the two versions one is usually guided by whether it is more important to compute an approximation to a solution of (1) and (2) which satisfies (1), or whether it is more important that it satisfy (2). ■

As we have just seen, at each iteration of the quasi-linearization version of the Newton-Raphson method, we must solve the linear boundary-value problem (29), (30). It is sometimes possible to reduce the labor necessary to perform this task, which we shall now discuss.

To simplify notation, consider the problem of solving the linear boundary-value problem defined by

$$33 \quad x_{i+1} - x_i = A_i x_i + v_i, \quad i = 0, 1, \dots, k - 1,$$

$$34 \quad P_0 x_0 = y_0, \quad P_k x_k = y_k,$$

where $x_i \in \mathbb{R}^\nu$, $v_i \in \mathbb{R}^\nu$, $y_0 \in \mathbb{R}^\alpha$, $y_k \in \mathbb{R}^{\nu-\alpha}$ ($\alpha < \nu$), and P_0, P_k are full rank matrices of appropriate dimensions.

Let $\Phi_{j,i}$ be $\nu \times \nu$ matrices which, for $i = 0, 1, \dots, k$, and for $j = i, i + 1, \dots, k$ are determined by

$$35 \quad \Phi_{j+1,i} - \Phi_{j,i} = A_j \Phi_{j,i}, \quad j = i, i + 1, \dots, k - 1, \quad \Phi_{i,i} = I,$$

where I is the $\nu \times \nu$ identity matrix. Then it is easily seen that

$$36 \quad x_k = \Phi_{k,0} x_0 + \sum_{i=0}^{k-1} \Phi_{k,i+1} v_i.$$

Consequently, (34) becomes

$$37 \quad P_0 x_0 = y_0,$$

$$38 \quad P_k \Phi_{k,0} x_0 = y_k - \sum_{i=0}^{k-1} P_k \Phi_{k,i+1} v_i,$$

i.e., (34) reduces to a system of equations in the initial state x_0 only.* Clearly, once we have solved (37) and (38) for x_0 , we can compute the remaining x_i , $i = 1, 2, \dots, k-1$, by solving (33) from this x_0 . Note that, as in the Goodman-Lance version of the Newton-Raphson method, we can compute the matrix $P_k \Phi_{k,0}$, as well as the matrices $P_k \Phi_{k,i+1}$, $i = 0, 1, \dots, k-1$, by solving the matrix equation

$$39 \quad \dot{P}_i - \dot{P}_{i+1} = \dot{P}_{i+1} A_i, \quad i = 0, 1, \dots, k-1, \quad \text{with } \dot{P}_k = P_k,$$

to yield $\dot{P}_{i+1} = P_k \Phi_{k,i+1}$, $i = -1, 0, 1, \dots, k-1$. This is obviously less work than computing the matrices $P_k \Phi_{k,i}$, $i = 0, 1, \dots, k$, by first solving (35), since (39) is a difference equation in fewer variables than (35), and hence requires fewer multiplications at each step in its solution.

When the system (33) is stable and k is large, $P_k \Phi_{k,0}$ (as well as $P_k \Phi_{k,i}$ for i small) may be a matrix whose elements are extremely small compared with the elements of P_0 , and hence the system of equations (37), (38) may be badly ill-conditioned, a phenomenon we have already observed in the Goodman-Lance version.

In 1961, it was suggested by Abramov [A1] that this ill-conditioning effect can be reduced or eliminated by a device which we shall now describe.[‡]

We begin by observing that (38) represents a transfer of boundary conditions from the terminal time k to the initial time 0, and that this transfer is performed by utilizing the properties of the adjoint equation (39). Let us expand this observation. Thus, suppose that $p_{s,0}, p_{s,1}, \dots, p_{s,k}$ are vectors in \mathbb{R}^r defined by

$$40 \quad p_{s,i} - p_{s,i+1} = A_i^T p_{s,i+1}, \quad i = 0, 1, \dots, k-1, \quad p_{s,k} = P_{k,s}^T,$$

where $P_{k,s}$ is the s th row of the matrix P_k in (34). Further, let

$$41 \quad w_{s,i} = \langle p_{s,i}, x_i \rangle, \quad i = 0, 1, \dots, k,$$

* The system (37), (38) may have a unique solution, a family of solutions, or no solution. The solution is unique when the matrix

$$\begin{pmatrix} P_0 \\ P_k \Phi_{k,0} \end{pmatrix}$$

is nonsingular. When $(I + A_i)$ is nonsingular for $i = 0, 1, \dots, k-1$, we can also construct a system of equations in x_k only, but the effort is obviously much larger.

[†] Here, we use the superscript j as a distinguishing mark only.

[‡] We are indebted to Moiseev for the following condensation of Abramov's work [M8].

where the x_i satisfy (33) and (34). Then, by (34), $w_{s,k} = y_k^s$, the s th component of y_k , and, in addition,

$$\begin{aligned} 42 \quad w_{s,i+1} &= \langle p_{s,i+1}, x_{i+1} \rangle \\ &= \langle p_{s,i+1}, (I + A_i) x_i \rangle + \langle p_{s,i+1}, v_i \rangle \\ &= \langle (I + A_i^T) p_{s,i+1}, x_i \rangle + \langle p_{s,i+1}, v_i \rangle \\ &= \langle w_{s,i} \rangle + \langle p_{s,i+1}, v_i \rangle, \quad i = 0, 1, \dots, k-1. \end{aligned}$$

Rewriting (42), we obtain

$$43 \quad w_{s,i} - w_{s,i+1} = -\langle p_{s,i+1}, v_i \rangle, \quad i = 0, 1, \dots, k-1, \quad w_{s,k} = y_k^s.$$

We now see that (38) can be written in the alternative form

$$44 \quad \langle p_{s,0}, x_0 \rangle = w_{s,0}, \quad s = 1, 2, \dots, \nu - \alpha,$$

since $p_{s,i} = \Phi_{k,i}^T P_{k,s}^T$, $i = 0, 1, 2, \dots, k$. So far, we have merely rederived (38) by an alternative approach which consists of the following operations: (i) solve (40) $\nu - \alpha$ times, with the initial conditions $p_{s,k} = P_{k,s}^T$, $s = 1, 2, \dots, \nu - \alpha$; (ii) solve (43) $\nu - \alpha$ times, with the initial conditions $w_{s,k} = y_k^s$, $s = 1, 2, \dots, \nu - \alpha$; (iii) construct the system of equations (44). The labor in this is the same as in the previous derivation.

Abramov's device consists of not using the vectors $p_{s,i}$ to transfer the boundary conditions from time k to time 0, but the vectors $q_{s,i} = p_{s,i}/\|p_{s,i}\|$. The advantage of doing this lies in the fact that while all the elements of the vectors $p_{s,0}$ may be extremely small compared to the nonzero elements of the matrix P_0 (causing severe ill-conditioning in the system (37), (44)), the vectors $q_{s,0}$ certainly do not have this undesirable property. We now show that the vectors $q_{s,i}$ can indeed be used to construct a normalized system of equations to replace the possibly ill-conditioned system (44).

First, note that

$$\begin{aligned} 45 \quad q_{s,i} &= \frac{1}{\|p_{s,i}\|} p_{s,i} = \frac{1}{\|(I + A_i^T) p_{s,i+1}\|} (I + A_i^T) p_{s,i+1} \\ &= \frac{1}{\|(I + A_i^T) q_{s,i+1}\|} (I + A_i^T) q_{s,i+1} \\ &\quad i = 0, 1, \dots, k-1, \quad q_{s,k} = p_{s,k}/\|p_{s,k}\|, \quad s = 1, 2, \dots, \nu - \alpha. \end{aligned}$$

Next, we set

$$46 \quad \bar{w}_{s,i} = \langle q_{s,i}, x_i \rangle, \quad i = 0, 1, 2, \dots, k,$$

where the x_i satisfy (33) and (34). Then we find that

$$\begin{aligned} 47 \quad \bar{w}_{s,i+1} &= \langle q_{s,i+1}, x_{i+1} \rangle \\ &= \langle q_{s,i+1}, (I + A_i) x_i \rangle + \langle q_{s,i+1}, v_i \rangle \\ &= \langle (I + A_i^T) q_{s,i+1}, x_i \rangle + \langle q_{s,i+1}, v_i \rangle \\ &= \|(I + A_i^T) q_{s,i+1}\| \langle q_{s,i}, x_i \rangle + \langle q_{s,i+1}, v_i \rangle, \quad i = 0, 1, \dots, k-1. \end{aligned}$$

Rearranging terms, we now get

$$\begin{aligned} 48 \quad \bar{w}_{s,i} &= \frac{(\bar{w}_{s,i+1} - \langle q_{s,i+1}, v_i \rangle)}{\|(I + A_i^T) q_{s,i+1}\|}, \\ i &= 0, 1, \dots, k-1, \quad s = 1, 2, \dots, \nu - \alpha, \quad \text{with } \bar{w}_{s,k} = \langle q_{s,k}, x_k \rangle = \frac{y_k^s}{\|p_{s,k}\|}. \end{aligned}$$

We therefore see that we can substitute for the set of equations (44) the set of equations

$$49 \quad \langle q_{s,0}, x_0 \rangle = \bar{w}_{s,0}, \quad s = 1, 2, \dots, \nu - \alpha,$$

which is obtained by first solving the difference equation (45) $\nu - \alpha$ times, with initial conditions $q_{s,k} = P_{k,s}^T$ (where $P_{k,s}$ is the s th row of the matrix P_k in (34)), and then by solving (48) $\nu - \alpha$ times, with initial conditions $\bar{w}_{s,k} = y_k^s / (P_{k,s} P_{k,s}^T)^{1/2}$. Thus, to obtain an x_0 which satisfies (34) and which is such that the corresponding x_k , computed by solving (33) from this x_0 , also satisfies (34), we may solve the system of equations (37), (49).

Although in the system of equations (37), (49) we may find that the elements of the $q_{s,0}$ are not particularly small or large as compared to the elements of the matrix P_0 , we may still be faced with an ill-conditioning effect produced by the fact that the vectors $q_{s,0}$ may have become close to being parallel even if the vectors $q_{s,k}$ were orthogonal (this is particularly likely when k is large), i.e., we may find that $\langle q_{i,0}, q_{j,0} \rangle \neq 1$ even if $\langle q_{i,k}, q_{j,k} \rangle \neq 0$, $i \neq j$. At present there seem to be no general methods for eliminating all causes of ill-conditioning effects in boundary-value problems with difference equations. As we shall see later, one can do somewhat better in the case of differential equations.

- 50 **Exercise.** Modify the quasi-linearization version of the Newton-Raphson algorithm (31) so as to make it applicable for solving (1) with boundary conditions of the form $g_0(x_0) = 0, g_k(x_k) = 0$, where $g_0 : \mathbb{R}^\nu \rightarrow \mathbb{R}^\alpha$ and $g_k : \mathbb{R}^\nu \rightarrow \mathbb{R}^{\nu-\alpha}$ are continuously differentiable and the Jacobian matrices $\partial g_0(z)/\partial z, \partial g_k(z)/\partial z$ are of full rank for all z in a sufficiently large open set in \mathbb{R}^ν . ■

- 51 **Exercise.** Obtain a quasi-linearization version of the algorithm (2.1.42) in a form applicable for solving boundary-value problems with difference equations. ■
- 52 **Exercise.** Show how Abramov's method described above can also be used to reduce ill-conditioning effects in the Goodman-Lance version of the Newton-Raphson method (18). [Hint: Set $\delta x_k = (\partial x_k(z^j)/\partial z)(z^{j+1} - z^j)$ and show that δx_k is the solution at time k of the variational difference equation

$$53 \quad \delta x_{i+1} - \delta x_i = \frac{\partial f_i(x_i)}{\partial x_i} \delta x_i, \quad i = 0, 1, \dots, k-1, \quad \text{with} \quad \delta x_0 = z^{j+1} - z^j,$$

which must satisfy the *terminal* boundary condition

$$54 \quad P_k \delta x_k = -P_k x_k(z) - y_k.$$

Now apply Abramov's method to obtain a well-conditioned set of equations to replace (10)]. ■

We now turn to the discrete optimal control problem,

$$55 \quad \text{minimize} \quad \sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^\mu,$$

subject to

$$56 \quad x_{i+1} - x_i = f_i(x_i, u_i), \quad i = 0, 1, 2, \dots, k-1,$$

$$57 \quad x_0 = \hat{x}_0, \quad g_k(x_k) = 0,$$

where we shall assume all the functions $f_i^0(\cdot, \cdot)$, $f_i(\cdot, \cdot)$, $\varphi(\cdot)$ and $g_k(\cdot)$ ($g_k : \mathbb{R}^v \rightarrow \mathbb{R}^\alpha$) to be twice continuously differentiable. We include here the case $g_k(\cdot) = 0$, i.e., the unconstrained optimization problem considered in the previous section. When $g_k(\cdot) \neq 0$, we shall assume that $\alpha < v$ and that the Jacobian matrix $\partial g_k(x)/\partial x$ has full rank for all x in a sufficiently large open set in \mathbb{R}^v .

We recall (see (1.2.24)) that for the problem (55), if $\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1}$ is an optimal control sequence and $\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k$ is the corresponding optimal trajectory, then there exist multiplier vectors $\hat{p}_1, \hat{p}_2, \dots, \hat{p}_k, \hat{\pi}$ such that

$$58 \quad \hat{x}_{i+1} - \hat{x}_i - f_i(\hat{x}_i, \hat{u}_i) = 0, \quad i = 0, 1, \dots, k-1,$$

$$59 \quad g_k(\hat{x}_k) = 0,$$

$$60 \quad \hat{p}_i - \hat{p}_{i+1} - \left(\frac{\partial f_i(\hat{x}_i, \hat{u}_i)}{\partial x_i} \right)^T \hat{p}_{i+1} + \left(\frac{\partial f_i^0(\hat{x}_i, \hat{u}_i)}{\partial x_i} \right)^T = 0, \quad i = 1, 2, \dots, k-1,$$

$$61 \quad \hat{p}_k - \left(\frac{\partial g_k(\hat{x}_k)}{\partial x_k} \right)^T \pi + \left(\frac{\partial \varphi(\hat{x}_k)}{\partial x_k} \right)^T = 0,$$

$$62 \quad - \left(\frac{\partial f_i^0(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T + \left(\frac{\partial f_i(\hat{x}_i, \hat{u}_i)}{\partial u_i} \right)^T \hat{p}_{i+1} = 0, \quad i = 0, 1, \dots, k-1.$$

- 63 **Remark.** In writing out (58)–(62), we have assumed that problem (55) is “nondegenerate,” i.e., that the multiplier \hat{p}^0 stipulated in (1.2.24) can be assumed to be -1 . When the problem is degenerate, one must proceed in a manner analogous to the one explained in Section 1 for the problem (1.16). ■

In a number of important cases, it is possible to eliminate the \hat{u}_i , $i = 0, 1, 2, \dots, k-1$, from the system of equations (58)–(62), by means of (62), to obtain a smaller system of equations involving the \hat{x}_i and the \hat{p}_i only. We shall consider these special cases later. For the time being, let us assume that there are no obvious simplifications to the set of equations that can be made, i.e., let us consider the general case first. In an attempt to solve the system (58)–(62) by means of the Newton-Raphson method, we are faced with two obvious choices in defining the vector z . The first choice is to set

$$64 \quad z = (x_1, x_2, \dots, x_k, p_1, p_2, \dots, p_k, \pi, u_0, u_1, \dots, u_{k-1}).$$

The second choice is to set

$$65 \quad z = (u_0, u_1, \dots, u_{k-1}, p_1, p_2, \dots, p_k, \pi),$$

and to eliminate the x_i from (58)–(62) by means of (58), deleting (58) from the system of equations to be solved in the process. We shall develop formulas for use in the Newton-Raphson method based on the definition (64). The reader will be given the opportunity to explore the possibilities presented by definition (65). Note that the specific algorithm we shall obtain is of the quasi-linearization type.

With z defined as in (64), the function $g(\cdot)$ in (3) is defined by (58)–(62). From (64), $g(\cdot)$ has $k\nu + k\nu + \alpha + k\mu = k(2\nu + \mu) + \alpha$ arguments, while from (58)–(62), it has $k\nu + \alpha + (k-1)\nu + \nu + k\mu = k(2\nu + \mu) + \alpha$ components, and hence it satisfies the requirement of the Newton-Raphson method that it map z into the space of which it is an element (i.e., $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$, with $n = k(2\nu + \mu) + \alpha$). Next, we turn to equation (8), for which we must obtain an expression in this particular case. Because of the great complexity of the system (58)–(62), it is probably easiest to obtain the detailed expressions for (8) by setting $\delta z^j = z^{j+1} - z^j$ and computing (8) in the rearranged form,

$$66 \quad \frac{\partial g(\hat{z})}{\partial z} \delta z^j = -g(z^j),$$

which can be done by expanding the equations (58)–(62) to first-order terms in $\delta \dot{z}$ about the point \dot{z} . Performing this task, we obtain from (58),

$$67 \quad \delta \dot{x}_{i+1}^j - \delta \dot{x}_i^j - \frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i} \delta \dot{x}_i^j - \frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial u_i} \delta \dot{u}_i^j = -\dot{\xi}_i^j, \quad i = 0, 1, \dots, k-1,$$

where $\delta \dot{x}_0^j = 0$ and

$$68 \quad \dot{\xi}_i^j = \dot{x}_{i+1}^j - \dot{x}_i^j - f_i(\dot{x}_i^j, \dot{u}_i^j), \quad i = 0, 1, \dots, k-1.$$

Next, from (59), we obtain

$$69 \quad \frac{\partial g_k(\dot{x}_k^j)}{\partial x_k} \delta \dot{x}_k^j = -g_k(\dot{x}_k^j);$$

and from (60), we obtain

$$70 \quad \begin{aligned} \delta \dot{p}_i^j - \delta \dot{p}_{i+1}^j &= \left(\frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i} \right)^T \delta \dot{p}_{i+1}^j - \frac{\partial}{\partial x_i} \left[\left(\frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i} \right)^T \dot{p}_{i+1}^j \right] \delta \dot{x}_i^j \\ &\quad - \frac{\partial}{\partial u_i} \left[\left(\frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial u_i} \right)^T \dot{p}_{i+1}^j \right] \delta \dot{u}_i^j + \frac{\partial^2 f_i^0(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i^2} \delta \dot{x}_i^j \\ &\quad + \left(\frac{\partial^2 f_i^0(\dot{x}_i^j, \dot{u}_i^j)}{\partial u_i \partial x_i} \right)^T \delta \dot{u}_i^j = -\dot{\zeta}_i^j, \end{aligned} \quad i = 1, 2, \dots, k-1,$$

where $(\partial^2 f_i^0(x, u)/\partial u \partial x)^T$ is a $\nu \times \mu$ matrix whose j th row is $(\partial/\partial u) [(\partial f_i^0(x, u)/\partial x)^T]^j$, and

$$71 \quad \begin{aligned} \dot{\zeta}_i^j &= \dot{p}_i^j - \dot{p}_{i+1}^j - \left(\frac{\partial f_i(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i} \right)^T \dot{p}_{i+1}^j - \left(\frac{\partial f_i^0(\dot{x}_i^j, \dot{u}_i^j)}{\partial x_i} \right)^T, \\ &\quad i = 1, 2, \dots, k-1. \end{aligned}$$

From (61), we obtain

$$72 \quad \delta \dot{p}_k^j - \left(\frac{\partial g_k(\dot{x}_k^j)}{\partial x_k} \right)^T \delta \dot{\pi}^j - \frac{\partial}{\partial x_k} \left[\left(\frac{\partial g_k(\dot{x}_k^j)}{\partial x_k} \right)^T \dot{\pi}^j - \left(\frac{\partial \varphi(\dot{x}_k^j)}{\partial x_k} \right)^T \right] \delta \dot{x}_k^j = -\dot{\eta}^j,$$

where

$$73 \quad \dot{\eta}^j = \dot{p}_k^j - \left(\frac{\partial g_k(\dot{x}_k^j)}{\partial x_k} \right)^T \dot{\pi}^j + \left(\frac{\partial \varphi(\dot{x}_k^j)}{\partial x_k} \right)^T.$$

Finally, from (62), we obtain

$$\begin{aligned}
74 \quad & - \left(\frac{\partial^2 f_i^0(\dot{x}_i, \dot{u}_i)}{\partial x_i \partial u_i} \right)^T \delta \dot{x}_i - \left(\frac{\partial^2 f_i^0(\dot{x}_i, \dot{u}_i)}{\partial u_i^2} \right) \delta \dot{u}_i + \left(\frac{\partial f_i(\dot{x}_i, \dot{u}_i)}{\partial u_i} \right)^T \delta \dot{p}_{i+1} \\
& + \frac{\partial}{\partial x_i} \left[\left(\frac{\partial f_i(\dot{x}_i, \dot{u}_i)}{\partial u_i} \right)^T \dot{p}_{i+1} \right] \delta \dot{x}_i + \frac{\partial}{\partial u_i} \left[\left(\frac{\partial f_i(\dot{x}_i, \dot{u}_i)}{\partial u_i} \right)^T \dot{p}_{i+1} \right] \delta \dot{u}_i \\
& = -\tau_i^j, \quad i = 0, 1, \dots, k-1,
\end{aligned}$$

where

$$75 \quad \tau_i^j = - \left(\frac{\partial f_i^0(\dot{x}_i, \dot{u}_i)}{\partial u_i} \right)^T + \left(\frac{\partial f_i(\dot{x}_i, \dot{u}_i)}{\partial u_i} \right)^T \dot{p}_{i+1}, \quad i = 0, 1, \dots, k-1.$$

The system of equations (67)–(75) is a mixture of difference equations ((67) and (70)) and of algebraic equations. It presents us with two eventualities: either one can solve, uniquely, for $\delta \dot{u}_i$ the system (74), or not. When (74) cannot be solved to obtain a unique expression for the $\delta \dot{u}_i$ in terms of the $\delta \dot{x}_i$ and the $\delta \dot{p}_i$, we are faced with a horrendously large system of equations to solve and our chances of success are not too good, since it is not at all clear as to how one can efficiently decompose this task into a sequence of simpler ones. However, when one can obtain a unique expression for the $\delta \dot{u}_i$ in terms of the $\delta \dot{x}_i$ and the $\delta \dot{p}_i$ from (74), the situation becomes considerably improved, as we shall now show.

Thus, suppose that (74) can be solved for the $\delta \dot{u}_i$ in terms of the $\delta \dot{x}_i$ and the $\delta \dot{p}_i$, and that this solution is unique. Then, upon substituting for the $\delta \dot{u}_i$ into (67) and (70) and replacing $\delta \dot{x}_i$ by $\dot{x}_{i+1}^j - \dot{x}_i^j$ and $\delta \dot{p}_i$ by $\dot{p}_{i+1}^j - \dot{p}_i^j$, we obtain a system of equations of the form,

$$76 \quad \dot{x}_{i+1}^j - \dot{x}_i^j = A_i^j \dot{x}_i^j + B_i^j \dot{p}_{i+1}^j - v_i^j, \quad i = 0, 1, \dots, k-1,$$

$$77 \quad \dot{p}_i^j - \dot{p}_{i+1}^j = C_i^j \dot{x}_i^j + D_i^j \dot{p}_{i+1}^j - w_i^j, \quad i = 1, 2, \dots, k-1,$$

$$78 \quad \dot{x}_0^j = \dot{x}_0, \quad G_k^j \dot{x}_k^j = -g_k^j, \quad \dot{p}_k^j = G_k^j \dot{x}_{\pi}^j - \Phi_k^j \dot{x}_k^j + y_k^j, *$$

where (78) is obtained from (57), (69) and (72).

In developing a special method for solving the boundary-value problem (76)–(78), we shall assume that all the inverses used do indeed exist. When they do not exist, the implied simplifications obviously cannot be utilized and one has to revert to solving this boundary-value problem either as a system of algebraic equations which is defined by (76)–(78) or by other

* Do not confuse the Φ_k^j , w_i^j , v_i^j , p_i^j , and q_i^j appearing here with similarly denoted quantities used earlier in this section.

means which may appear to be expedient in a specific situation. In order to simplify notation, we shall drop the superscripts j and $j + 1$ from all the symbols appearing in (76)–(78). Thus, in the worst case, we have to solve the system

$$79 \quad \left[\begin{array}{ccccccccc|c} I & 0 & 0 & \cdots & 0 & 0 & -B_0 & 0 & \cdots & 0 & 0 & 0 \\ -\bar{A}_1 & I & 0 & \cdots & 0 & 0 & 0 & -B_1 & 0 & \cdots & 0 & 0 \\ 0 & -\bar{A}_2 & I & 0 & \cdots & 0 & 0 & 0 & 0 & -B_2 & 0 & \cdots \\ \vdots & & & & & & & & & & & 0 \\ 0 & \cdots & 0 & -\bar{A}_{k-1} & I & \cdots & & & 0 & -B_{k-1} & 0 \\ -C_1 & 0 & 0 & \cdots & I & -\bar{D}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -C_2 & 0 & \cdots & 0 & I & -\bar{D}_2 & 0 & \cdots & 0 & 0 \\ \vdots & & & & & & & & & & & \\ 0 & \cdots & 0 & -C_{k-1} & 0 & 0 & 0 & \cdots & I & -\bar{D}_{k-1} & 0 \\ 0 & \cdots & 0 & 0 & G_k & \cdots & 0 & \cdots & 0 & 0 & 0 \\ 0 & \cdots & 0 & \Phi_k & & \cdots & & & 0 & I & -G_k^T \end{array} \right] \times$$

$$\left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_k \\ p_1 \\ p_2 \\ \vdots \\ p_{k-1} \\ p_k \\ \pi \end{array} \right] = \left[\begin{array}{c} -v_0 + \bar{A}_0 x_0 \\ -v_1 \\ -v_2 \\ \vdots \\ -v_{k-1} \\ -w_1 \\ -w_2 \\ \vdots \\ -w_{k-1} \\ -g_k \\ y_k \end{array} \right],$$

where for $i = 0, 1, \dots, k - 1$, $\bar{A}_i = (I + A_i)$, and for $i = 1, 2, \dots, k - 1$, $\bar{D}_i = (I + D_i)$. Since the matrix in (79) is sparse (i.e., it has many zeros), the task of solving (79) need not be hopeless. However, we shall assume that the problem is such that one can utilize its dynamical structure to simplify computations, as shown below.

We begin by observing that from (77),

$$80 \quad p_{k-1} = C_{k-1}x_{k-1} + \bar{D}_{k-1}p_k - w_{k-1},$$

where $\bar{D}_{k-1} = (I + D_{k-1})$. If we treat p_k as a known constant, then we see that

$$81 \quad p_{k-1} = K_{k-1}x_{k-1} + q_{k-1},$$

where we have simply defined $K_{k-1} = C_{k-1}$, $q_{k-1} = \bar{D}_{k-1}p_k - w_{k-1}$. We shall now show that for $j = 1, 2, \dots, k$,

$$82 \quad p_j = K_jx_j + q_j,$$

where the matrices K_j and the vectors q_j will be defined later. Thus, suppose that for $j = i + 1$,

$$83 \quad p_{i+1} = K_{i+1}x_{i+1} + q_{i+1}.$$

Then, from (76), with $\bar{A}_i = (I + A_i)$ for $i = 0, 1, 2, \dots, k - 1$, we obtain

$$84 \quad x_{i+1} = \bar{A}_i x_i + B_i K_{i+1} x_{i+1} + B_i q_{i+1} - v_i.$$

Solving (84) for x_{i+1} , we now obtain

$$85 \quad x_{i+1} = (I - B_i K_{i+1})^{-1}(\bar{A}_i x_i + B_i q_{i+1} - v_i).$$

Substituting for p_{i+1} into (77) from (83) and (85), we finally obtain

$$86 \quad p_i = C_i x_i + \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} (\bar{A}_i x_i + B_i q_{i+1} - v_i) + \bar{D}_i q_{i+1} - w_i,$$

where we define $\bar{D}_i = (I + D_i)$ for $i = 1, 2, \dots, k - 1$. Collecting terms, we now get from (86) that

$$87 \quad p_i = [C_i + \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} \bar{A}_i] x_i + [\bar{D}_i + \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} B_i] q_{i+1} - \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} v_i - w_i,$$

i.e., we find that (82) is true for $j = i$, with K_i , q_i defined by

$$88 \quad K_i = C_i + \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} \bar{A}_i, \quad i = 1, 2, \dots, k - 1,$$

$$89 \quad q_i = \bar{D}_i [I + K_{i+1} (I - B_i K_{i+1})^{-1} B_i] q_{i+1} - \bar{D}_i K_{i+1} (I - B_i K_{i+1})^{-1} v_i - w_i, \quad i = 1, 2, \dots, k - 1.$$

Setting $K_{k-1} = C_{k-1}$ and $q_{k-1} = D_{k-1} p_k - w_{k-1}$, we see from (80) that (82) holds for $j = k - 1$. It now follows by induction that (82) holds for $j = k, k - 1, \dots, 1$, provided the inverses appearing in (86)–(89) exist, as we shall assume that they do. To complete our demonstration, we note that we are consistent if we set

$$90 \quad K_k = 0, \quad q_k = p_k.$$

91 **Exercise.** Show that there may exist valid boundary conditions for (88) and (89) other than those in (90). In particular, consider the choice $K_k = -\Phi_k$, $q_k = G_k^T \pi + y_k$. [Hint: See (120).] ■

We shall now summarize the procedure for solving (76)–(78) which is implied by the above development.

92 **Algorithm** (solves system (76)–(78)).*

Step 1. For $i = k, k - 1, k - 2, \dots, 1$, compute the K_i by solving (88) with the boundary condition in (90).

* When $G_k = 0$, the calculations below can compute q_i directly from (89), for $i = 1, 2, \dots, k - 1$, by setting $q_k = y_k$. (We do not know p_k in advance.)

Step 2. For $i = 1, 2, \dots, k - 1$, compute

$$93 \quad L_i = D_i[I + K_{i+1}(I - B_i K_{i+1})^{-1} B_i],$$

$$94 \quad l_i = -D_i K_{i+1}(I - B_i K_{i+1})^{-1} v_i - w_i.$$

Comment. Equation (89), with its boundary conditions, may now be written in the more compact form,

$$95 \quad q_i = L_i q_{i+1} + l_i, \quad i = 1, 2, \dots, k - 1, \quad q_k = p_k.$$

Step 3. For $i = 0, 1, 2, \dots, k - 1$, compute

$$96 \quad M_i = (I - B_i K_{i+1})^{-1} A_i,$$

$$97 \quad N_i = (I - B_i K_{i+1})^{-1} B_i,$$

$$98 \quad m_i = -(I - B_i K_{i+1})^{-1} v_i.$$

Comment. Equation (76), with its boundary conditions from (78), can now be written in the more compact form,

$$99 \quad x_{i+1} = M_i x_i + N_i q_{i+1} + m_i, \\ i = 0, 1, \dots, k - 1, \quad x_0 = \hat{x}_0, \quad G_k x_k = -g_k.$$

Step 4. For $i = 1, 2, \dots, k$, compute

$$100 \quad \tilde{L}_i = L_i L_{i+1} \cdots L_{k-1}, \quad \tilde{L}_k = I$$

$$101 \quad \tilde{l}_i = \sum_{j=k-1}^{i+1} L_i L_{i+1} \cdots L_{j-1} l_j + l_i, \quad \tilde{l}_k = 0.$$

Comment. The q_i can now be expressed as follows:

$$102 \quad q_i = \tilde{L}_i p_k + \tilde{l}_i, \quad i = 1, 2, \dots, k.$$

Step 5. For $i = 1, 2, \dots, k$, compute

$$103 \quad \tilde{M}_i = M_{k-1} M_{k-2} \cdots M_i N_{i-1}, \quad \tilde{M}_k = M_{k-1} M_{k-2} \cdots M_0,$$

$$104 \quad \hat{M}_i = M_{k-1} M_{k-2} \cdots M_i, \quad \hat{M}_k = 0.$$

Comment. With the above definitions, the solution of (99) is seen to be

$$105 \quad x_k = \tilde{M}_k \hat{x}_0 + \sum_{i=1}^{k-1} \tilde{M}_i q_i + \sum_{i=1}^{k-1} \hat{M}_i m_{i-1} + N_{k-1} q_k + m_{k-1} \\ = \tilde{M}_k \hat{x}_0 + \left(\sum_{i=1}^{k-1} \tilde{M}_i \tilde{L}_i + N_{k-1} \right) p_k + \sum_{i=1}^{k-1} \tilde{M}_i \tilde{l}_i + \sum_{i=1}^{k-1} \hat{M}_i m_{i-1} + m_{k-1},$$

i.e.,

$$106 \quad x_k = E_k p_k + d_k,$$

where E_k and d_k are defined to be the corresponding matrix and vector in (105) (d_k does not depend on p_k).

Step 6. Set

$$107 \quad E_k = \sum_{i=1}^{k-1} \tilde{M}_i \tilde{L}_i + N_{k-1},$$

$$108 \quad d_k = \tilde{M}_k \hat{x}_0 + \sum_{i=1}^{k-1} \tilde{M}_i \tilde{L}_i + \sum_{i=1}^{k-1} \hat{M}_i m_{i-1} + m_{k-1}.$$

Comment. Applying the boundary conditions (78) to (106), we obtain

$$\begin{aligned} 109 \quad x_k &= E_k p_k + d_k \\ &= E_k (G_k^T \pi - \Phi_k x_k + y_k) + d_k \\ &= (I + E_k \Phi_k)^{-1} [E_k (G_k^T \pi + y_k) + d_k], \end{aligned}$$

$$\begin{aligned} 110 \quad G_k x_k &= G_k (I + E_k \Phi_k)^{-1} E_k G_k^T \pi + G_k (I - E_k \Phi_k)^{-1} (E_k y_k + d_k) \\ &= -g_k. \end{aligned}$$

Step 7. Set

$$111 \quad \pi = -[G_k (I - E_k \Phi_k)^{-1} E_k G_k^T]^{-1} [g_k + G_k (I + E_k \Phi_k)^{-1} (E_k y_k + d_k)],$$

$$112 \quad x_k = (I + E_k \Phi_k)^{-1} [E_k (G_k^T \pi + y_k) + d_k],$$

$$113 \quad p_k = G_k^T \pi - \Phi_k x_k + y_k.$$

Step 8. For $i = 1, 2, \dots, k$, compute q_i by solving (95), x_i by solving (85) with $x_0 = \hat{x}_0$, and p_i from (82). ■

It should be clear by now that the solution of the discrete optimal control problem (55)–(57) could be extremely time-consuming, at least in the general case, with k and ν both large. However, there are a number of very important cases where considerable simplifications take place.

114 **Example.** Suppose that (55)–(57) have the specific form

$$115 \quad \text{minimize} \quad \sum_{i=0}^{k-1} f_i^0(x_i) + \frac{1}{2} \| u_i \|^2,$$

subject to

$$116 \quad x_{i+1} - x_i = f_i(x_i) + B u_i, \quad i = 0, 1, 2, \dots, k-1, \quad x_0 = \hat{x}_0, \quad x_k = \hat{x}_k.$$

In this case, $\varphi(\cdot) = 0$, $g_k(x_k) \equiv x_k - \hat{x}_k$, and hence (61) becomes $p_k = \pi$, i.e., (61) carries no information and can be dropped. At the same time, (62) becomes (under the non-degeneracy assumption stated, i.e., $p^0 = -1$),

$$117 \quad -\hat{u}_i + B^T \hat{p}_{i+1} = 0, \quad i = 0, 1, 2, \dots, k-1.$$

Substituting for u_i into (116) (and dropping the hats), we see that an optimal trajectory $\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k$ must satisfy the system of equations,

$$118 \quad x_{i+1} - x_i = f_i(x_i) + BB^T p_{i+1}, \quad i = 0, 1, \dots, k-1, \quad x_0 = \hat{x}_0, \quad x_k = \hat{x}_k,$$

$$119 \quad p_i - p_{i+1} = \left(\frac{\partial f_i(x_i)}{\partial x_i} \right)^T p_{i+1} - \left(\frac{\partial f_i^0(x_i)}{\partial x_i} \right)^T, \quad i = 1, 2, \dots, k-1,$$

with the optimal control sequence then determined by (117), where the \hat{p}_i satisfy (118), (119) with $x_i = \hat{x}_i$, $i = 0, 1, \dots, k$.

Obviously, (118), (119) is a much simpler system of equations to solve than (58)–(62), which was encountered in the general case. ■

120 **Exercise.** Obtain the formulas for solving (118), (119) by means of the Newton-Raphson method. ■

121 **Exercise.** Consider the quadratic control problem,

$$122 \quad \text{minimize} \quad \frac{1}{2} \left(\sum_{i=1}^k \langle x_i, R_i x_i \rangle + \sum_{i=0}^{k-1} \langle u_i, Q_i u_i \rangle \right),$$

subject to

$$123 \quad x_{i+1} = \bar{A}_i x_i + B_i u_i, \quad i = 0, 1, \dots, k-1, \quad x_0 = \hat{x}_0,$$

where the $\nu \times \nu$ matrices R_i and the $\mu \times \mu$ matrices Q_i are symmetric and positive definite. Show that under a nondegeneracy assumption, the optimal control sequence $\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1}$ and the optimal trajectory $\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k$ must satisfy

$$124 \quad x_{i+1} = \bar{A}_i x_i + B_i Q_i^{-1} B_i^T p_{i+1}, \quad i = 0, 1, 2, \dots, k-1, \quad x_0 = \hat{x}_0,$$

$$125 \quad p_i = -R_i x_i + \bar{A}_i^T p_{i+1}, \quad i = 1, 2, \dots, k-1, \quad p_k = -R_k x_k,$$

$$126 \quad \hat{u}_i = Q_i^{-1} B_i^T \hat{p}_{i+1}, \quad i = 0, 1, 2, \dots, k-1,$$

where the \hat{p}_i , $i = 1, 2, \dots, k$, satisfy (124), (125) together with the \hat{x}_i , $i = 0, 1, 2, \dots, k$. Also show that for this case,

$$127 \quad \hat{p}_i = K_i \hat{x}_i, \quad i = 1, 2, \dots, k,$$

where K_i is a $\nu \times \nu$ matrix determined by solving

$$128 \quad K_i = -R_i + \bar{A}_i^T K_{i+1} (I - B_i Q_i^{-1} B_i^T K_{i+1})^{-1} \bar{A}_i, \quad i = 1, 2, \dots, k-1,$$

with the boundary condition $K_k = -R_k$. Note that because of (127) we obtain a *feedback control law*:

$$129 \quad \hat{u}_i = Q_i^{-1} B_i^T K_{i+1} (I - B_i Q_i^{-1} B_i^T K_{i+1})^{-1} \bar{A}_i \hat{x}_i, \quad i = 0, 1, 2, \dots, k-1.$$

Show that we could have set

$$130 \quad \hat{p}_i = K_i \hat{x}_i + q_i, \quad i = 1, 2, \dots, k,$$

with K_i determined by solving (128) with the boundary condition $K_k = 0$ and q_i determined by solving

$$131 \quad q_i = \bar{A}_i^T [I + K_{i+1} (I - B_i Q_i^{-1} B_i^T K_{i+1})^{-1} B_i Q_i^{-1} B_i^T] q_{i+1}, \quad i = 1, 2, \dots, k-1,$$

with the boundary condition $q_k = -R_k \hat{x}_k$. Which of the two possibilities (127), (130) would you chose for carrying out calculations? ■

To conclude this section, we shall show that under the assumptions stated in exercise (121), the Riccati equation (128) always has a solution, independently of whether we set $K_k = -R_k$ or $K_k = 0$. This fact is a result of the two lemmas given below.

- 132 **Lemma.** Suppose that D is a symmetric, positive semidefinite $\nu \times \nu$ matrix, and that K is a $\nu \times \nu$ negative semidefinite matrix. Then the matrix $(I - DK)$ is nonsingular.

Proof. We begin by noting that since D is symmetric and positive semidefinite, $Dx = 0$ if and only if $\langle x, Dx \rangle = 0$,* and that $(I - DK)$ is nonsingular if and only if $(I - K^T D)$ is nonsingular. Now, suppose that $(I - K^T D)$ is singular. Then there must exist a nonzero vector x' such that

$$133 \quad (I - K^T D)x' = 0,$$

which leads us to the conclusion that $K^T D x' = x'$, and hence that

$$134 \quad \langle Dx', K^T D x' \rangle = \langle Dx', x' \rangle.$$

It now follows from the fact that K^T is negative semidefinite and that D is positive semidefinite that

$$135 \quad \langle Dx', K^T D x' \rangle = \langle Dx', x' \rangle = 0,$$

* This follows from the fact that since $D \geq 0$ is symmetric, there exists a diagonal matrix $A \geq 0$ and an $n \times n$ matrix T such that $D = (T^T A^{1/2})(A^{1/2} T)$.

which leads us to the conclusion that $Dx' = 0$, and hence that $K^T D x' = 0$. But this is a contradiction and hence the proof is complete. ■

- 136 **Lemma.** Suppose that K, D are as in lemma (132). Then the matrix $K(I - DK)^{-1}$ is negative semidefinite.

Proof. We have shown in lemma (132) that $(I - DK)^{-1}$ exists; hence the matrix in question is well-defined. Next, for any $x \in \mathbb{R}^n$,

$$\begin{aligned} 137 \quad \langle x, K(I - DK)^{-1}x \rangle &= \langle (I - DK)^{-1}x, (I - DK)^T K(I - DK)^{-1}x \rangle \\ &= \langle y, Ky \rangle - \langle y, K^T D K y \rangle \leq 0, \end{aligned}$$

where $y = (I - DK)^{-1}x$, since K is negative semidefinite and D is positive semidefinite. ■

- 138 **Theorem.** Consider the difference equation (128). Suppose that for $i = 1, 2, \dots, k$, the matrices R_i are symmetric and positive semidefinite, and that for $i = 0, 1, 2, \dots, k-1$, the matrices Q_i are symmetric and positive definite. Then the difference (128) has a well-defined solution $K_i, i = 1, 2, \dots, k$ for $K_k = -R_k$. (Note that we may choose $R_k = 0$.)

Proof. We give a proof by induction. We shall show that the K_i not only exist, but that they are negative semidefinite. First, $K_k = -R_k$ is negative semidefinite, by definition. Suppose that K_{i+1} is negative semidefinite. Then, setting $D_i = B_i Q_i^{-1} B_i^T$, we find that D_i is positive semidefinite and hence, by lemma (132), we see that K_i is well-defined by (128). By lemma (136), the matrix $K_{i+1}(I - D_i K_{i+1})^{-1}$ is negative semidefinite. Since K_i is the sum of two negative semidefinite matrices, K_i is negative semidefinite. Thus we have shown that if K_{i+1} exists and is negative semidefinite, then K_i exists and is also negative semidefinite. But $K_k = -R_k$ is negative semidefinite, and hence the K_i are well-defined by (128) for $i = 1, 2, \dots, k$. ■

- 139 **Exercise.** Use a Riccati equation approach to obtain the optimal control law for the problem

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^k [\langle d_i, x_i \rangle + \frac{1}{2} \langle x_i, R_i x_i \rangle] \\ & + \sum_{i=0}^{k-1} [\langle c_i, u_i \rangle + \frac{1}{2} \langle u_i, Q_i u_i \rangle + \langle x_i, S_i u_i \rangle], \end{aligned}$$

subject to

$$x_{i+1} = \bar{A}_i x_i + B_i u_i, \quad i = 0, 1, 2, \dots, k-1, \quad x_0 = \mathbf{x}_0,$$

where the matrices R_i and Q_i are symmetric and positive definite. ■

3.3 Boundary-Value Problems and Continuous Optimal Control

The material to be presented in this section closely parallels the material in Section 2. In fact, a number of the results in this section can be obtained formally from those presented in Section 2 simply by replacing x_i by $x(t)$, u_i by $u(t)$, $x_{i+1} - x_i$ by $(d/dt)x(t)$, and so forth, since one can always consider the difference equation (2.1) to be a discretization of a differential equation. However, as we shall soon see, there are significant differences between the formulas one obtains for the discrete and the continuous case. Also, a number of results which are valid for the continuous case cannot be adapted to the discrete case without special assumptions, if at all. As we shall see, there are more things one can do and say in the case of boundary-value problems with continuous dynamics than in the case of boundary-value problems with discrete dynamics.

As in Section 2, we shall first show how the Newton-Raphson method (1.9) can be used to solve a general class of boundary-value problems with differential equations. Then we shall examine a special class of boundary-value problems which arise as necessary conditions of optimality for continuous optimal control problems, as a result of the Pontryagin maximum principle (1.2.35). To facilitate a comparison between the results in this section and those in Section 2, we shall retain the notation introduced in Section 2 for numbering iterates, viz., we shall write $\dot{x}(t)$, $\dot{u}(t)$, $\dot{p}(t)$ and not $x_j(t)$, $u_j(t)$, $p_j(t)$, which would have been more consistent with the notation used in Section 1. In Section 2, we saw that the Newton-Raphson method could be developed in at least two versions in application to discrete dynamics boundary-value problems. These two versions, the Goodman-Lance version [G4] and the quasi-linearization version [M2], [B6], also apply to continuous dynamics boundary-value problems. We begin with the Goodman-Lance version, since it is the easier one of the two to understand.

Thus, suppose that we are given a continuous dynamical system described by the differential equation

$$1 \quad \frac{d}{dt}x(t) = f(x(t), t), \quad t \in [t_0, t_f],$$

where $f: \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^n$ is continuously differentiable in x and piecewise continuous in t . Furthermore, we shall assume that for every $x \in \mathbb{R}^n$ the elements of the Jacobian $\partial f(x, \cdot)/\partial x$ are piecewise continuous. In addition, we are given the boundary conditions

$$2 \quad P_0x(t_0) = y_0, \quad P_fx(t_f) = y_f,$$

where $y_0 \in \mathbb{R}^\alpha$, $y_t \in \mathbb{R}^{\nu-\alpha}$, $\alpha < \nu$, and P_0, P_t are full-rank matrices of dimension $\alpha \times \nu$ and $(\nu - \alpha) \times \nu$, respectively.

As in the discrete case, to solve the boundary-value problem (1), (2) by the Newton-Raphson method, we must first transcribe it into the form

$$3 \quad g(z) = 0.$$

To do this for the Goodman-Lance version of the Newton-Raphson method, we define $x(t; t_0, z)$ to be the solution of (1) at time t corresponding to the initial state z , i.e., $x(t_0; t_0, z) = z$. Then we define the function $x_t : \mathbb{R}^\nu \rightarrow \mathbb{R}^\nu$ by*

$$4 \quad x_t(z) = x(t_t; t_0, z),$$

and, finally, we define the function $g : \mathbb{R}^\nu \rightarrow \mathbb{R}^\nu$ as

$$5 \quad g(z) = \begin{pmatrix} P_t x_t(z) - y_t \\ P_0 z - y_0 \end{pmatrix}.$$

Obviously, the function $g(\cdot)$ in (5) is differentiable. With $g(\cdot)$ defined as in (5), we now see that the problem (1), (2) is equivalent to solving the equation $g(z) = 0$ for an initial state z and then computing $x(t; t_0, z)$ for $t \in [t_0, t_t]$, with $P_t x_t(t_t; t_0, z) = y_t$ ensured by the fact that $g(z) = 0$.

From here on we proceed on the assumption that $(\partial g(z)/\partial z)^{-1}$ exists for all $z \in \mathbb{R}^\nu$, or at least for all z in a sufficiently large subset of \mathbb{R}^ν . Obviously, it may not be possible to verify this assumption. In an actual situation, we would simply go ahead and use the Newton-Raphson method and would account for a contingency occurring by inserting into the program a test which would stop computation whenever it appeared that the sequence being constructed diverges or whenever $|\det \partial g(z)/\partial z| < \epsilon$, where ϵ is very small.

We recall from (1.9) that given a point $\hat{z} \in \mathbb{R}^\nu$, the Newton-Raphson method constructs its successor z^{j+1} by solving the equation

$$6 \quad \frac{\partial g(z)}{\partial z} (z^{j+1} - \hat{z}) = -g(\hat{z}), \quad j = 0, 1, 2, \dots$$

Substituting from (5) into (6), we obtain

$$7 \quad P_t \frac{\partial x_t(z)}{\partial z} (z^{j+1} - \hat{z}) = -P_t x_t(\hat{z}) + y_t,$$

$$8 \quad P_0 (z^{j+1} - \hat{z}) = -P_0 \hat{z} + y_0,$$

* We assume that (1) has a solution on $[t_0, t_t]$ for every initial state $z \in \mathbb{R}^\nu$.

which simplifies out to

$$9 \quad P_t \frac{\partial x_t(\dot{z})}{\partial z} \dot{z}^{j+1} = P_t \frac{\partial x_t(\dot{z})}{\partial z} \dot{z}^j - P_t x_t(\dot{z}) + y_t,$$

$$10 \quad P_0 \dot{z}^{j+1} = y_0.$$

Now, it is not difficult to show that given a perturbation δz about an initial state \dot{z} of (1) we can compute the perturbation $[x(t_f; t_0, \dot{z} + \delta z) - x(t_f; t_0, \dot{z})]$ to first-order terms by solving

$$11 \quad \frac{d}{dt} \delta x(t) = \frac{\partial f(x(t; t_0, \dot{z}))}{\partial x} \delta x(t), \quad t \in [t_0, t_f], \quad \delta x(t_0) = \delta z,$$

for $\delta x(t_f)$. To do this, suppose that for $t, s \in [t_0, t_f]$, $\dot{\Phi}(t, s)$ is a $\nu \times \nu$ matrix defined as the solution of the differential equation,

$$12 \quad \frac{d}{dt} \dot{\Phi}(t, s) = \frac{\partial f(x(t; t_0, \dot{z}))}{\partial x} \dot{\Phi}(t, s), \quad t \in [t_0, t_f], \quad \dot{\Phi}(s, s) = I,$$

where I is the $\nu \times \nu$ identity matrix. Then we see that

$$13 \quad \delta x(t_f) = \dot{\Phi}(t_f, t_0) \delta z,$$

and hence we conclude that

$$14 \quad \frac{\partial x_t(\dot{z})}{\partial z} = \dot{\Phi}(t_f, t_0).$$

Consequently, (9) and (10) become

$$15 \quad P_t \dot{\Phi}(t_f, t_0) \dot{z}^{j+1} = P_t \dot{\Phi}(t_f, t_0) \dot{z}^j - P_t x_t(\dot{z}) + y_t,$$

$$16 \quad P_0 \dot{z}^{j+1} = y_0.$$

- 17 **Exercise.** Show that the matrix $\dot{\Phi}(t, s)$ also satisfies the adjoint differential equation,

$$18 \quad \frac{d}{ds} \dot{\Phi}(t, s) = -\dot{\Phi}(t, s) \frac{\partial f(x(s; t_0, \dot{z}), s)}{\partial x}, \quad s \in [t_0, t_f], \quad t \in [t_0, t_f].$$

Hence, show that the matrix $P_t \dot{\Phi}(t_f, t_0)$ can be computed by solving the differential equation,

$$19 \quad \frac{d}{ds} P(s) = -P(s) \frac{\partial f(x(s; t_0, \dot{z}), s)}{\partial x}, \quad s \in [t_0, t_f], \quad P(t_f) = P_t,$$

to yield $P_t \dot{\Phi}(t_f, t_0) = P(t_0)$. ■

Note that just as in the discrete case (compare (2.17)), it is easier to compute $P_t \dot{\Phi}(t_1, t_0)$ by solving (19), than by first solving (12) to obtain $\dot{\Phi}(t_0, t_1)$ separately, since (19) is a differential equation in fewer variables than (12).

We now summarize the above discussion by presenting it in the form of an algorithm.

- 20 **Algorithm** (Goodman-Lance version of Newton-Raphson method [G4]).

Step 0. Select a $\overset{0}{z} \in \mathbb{R}^v$.

Step 1. Set $j = 0$.

Step 2. For $t \in [t_0, t_1]$, compute $x(t; t_0, \overset{j}{z})$ by solving (1) with $x(t_0) = \overset{j}{z}$, and set $x_t(\overset{j}{z}) = x(t_1; t_0, \overset{j}{z})$.

Step 3. For $s \in [t_0, t_1]$, compute the Jacobian matrix $\partial f(x(s; t_0, \overset{j}{z})) / \partial x$.

Step 4. Compute $\overset{j}{P}(t_0)$ by solving (19).

Step 5. Compute $\overset{j+1}{z}$ by solving

$$21 \quad \overset{j}{P}(t_0) \overset{j+1}{z} = \overset{j}{P}(t_0) \overset{j}{z} - P_t x_t(\overset{j}{z}) + y_1,$$

$$22 \quad P_0 \overset{j+1}{z} = y_0.$$

Step 6. Set $j = j + 1$ and go to step 2. ■

Thus, just as in the application of the Goodman-Lance version to the discrete case, in Section 2, at each iteration we solve the nonlinear differential equation (1) in the forward direction from the initial state $x(t_0) = \overset{j}{z}$. Then we solve the linear, variational adjoint equation (19) backwards in time. Finally, we solve a linear algebraic set of equations (21), (22). Note that remark (2.21) is relevant to the present case as well.

- 23 **Exercise.** Adapt algorithm (20) for the solution of (1) with boundary conditions of the form $g_0(x(t_0)) = 0$, $g_1(x(t_1)) = 0$, where $g_0 : \mathbb{R}^v \rightarrow \mathbb{R}^\alpha$, $g_1 : \mathbb{R}^v \rightarrow \mathbb{R}^{v-\alpha}$, $\alpha < v$, are continuously differentiable functions whose Jacobians are full-rank matrices for all $z \in \mathbb{R}^v$. ■

- 24 **Exercise.** Modify algorithm (20) to obtain a quasi-Newton method of the form of algorithm (2.1.42). ■

To define the quasi-linearization version of the Newton-Raphson method [M2], [B6] for the boundary-value problem (1), (2), we must first introduce the Newton-Raphson method for solving equations of the form

$$25 \quad g(z) = 0,$$

when $g : L \rightarrow L$, with L a Banach space. In this case, the derivative of $g(\cdot)$

at $z \in L$ is denoted by $(\partial g(z)/\partial z)(\cdot)$ and, if it exists, is defined to be the linear functional from L into L , with the property that

$$26 \quad \lim_{\|\delta z\|_L \rightarrow 0} \left\| g(z + \delta z) - g(z) - \frac{\partial g(z)}{\partial z} (\delta z) \right\|_L / \|\delta z\|_L = 0,$$

where $\|\cdot\|_L$ denotes the norm in L .

Assuming that the inverse map $(\partial g(\cdot)/\partial z)^{-1}(\cdot)$ of $(\partial g(\cdot)/\partial z(\cdot))$ is well-defined for all $z \in L$ and is continuous on $L \times L$, the Newton-Raphson method for solving (25) is defined by

$$27 \quad \frac{\partial g(z^j)}{\partial z} (z^{j+1} - z^j) = -g(z^j), \quad j = 0, 1, 2, \dots,$$

i.e., to compute z^{j+1} from z^j we solve (27), exactly as in the finite dimensional case. Also, as in the finite dimensional case, if \hat{z} is the limit point of the sequence \hat{z}^j constructed according to (27), then we must have $g(\hat{z}) = 0$ (assuming of course, that this sequence converges and that $(\partial g(\cdot)/\partial z)^{-1}(\cdot)$ is a continuous map from $L \times L$ into L).

For our purposes, the preceding discussion of the Newton-Raphson method in Banach spaces is sufficient. For a discussion of a number of aspects of importance in perturbation theory, as well as of rate of convergence, the reader may wish to consult the very lucid paper by Antosiewicz [A2], or the original work of Kantorovich [K2].

We can now proceed with the description of the quasi-linearization version of the Newton-Raphson method. Note that in what follows, the choice of the variable z is made in a slightly different manner than was done in Section 2. Let $L = \mathbb{R}^\nu \times C_\nu'[t_0, t_f]$, where $C_\nu'[t_0, t_f]$ is the space of all piecewise continuously differentiable functions from $[t_0, t_f]$ into \mathbb{R}^ν , with the norm of $z = (x_0, x(\cdot))$ in L defined by

$$28 \quad \|z\|_L = (\|x_0\|^2 + \sup_{t \in [t_0, t_f]} \|x(t)\|^2)^{1/2}.$$

Given a vector $z \in L$, we shall always decompose it into two parts, as follows:

$$29 \quad z = (x_0, x(\cdot)), \quad x_0 \in \mathbb{R}^\nu, \quad x(\cdot) \in C_\nu'[t_0, t_f].$$

To define the map $g(\cdot)$ for the problem (1), (2), we introduce the maps $\tilde{g} : \mathbb{R}^\nu \times C_\nu'[t_0, t_f] \rightarrow C_\nu'[t_0, t_f]$, $\tilde{P}_0 : \mathbb{R}^\nu \times C_\nu'[t_0, t_f] \rightarrow \mathbb{R}^\alpha$ and $\tilde{P}_f : \mathbb{R}^\nu \times C_\nu'[t_0, t_f] \rightarrow \mathbb{R}^{\nu-\alpha}$, defined by

$$30 \quad \tilde{g}(x_0, x(\cdot))(t) = x_0 + \int_{t_0}^t f(x(s), s) ds - x(t), \quad t \in [t_0, t_f],$$

$$31 \quad \tilde{P}_0(x_0, x(\cdot)) = P_0 x_0 - y_0,$$

$$32 \quad \tilde{P}_f(x_0, x(\cdot)) = P_f x(t_f) - y_f,$$

where all the quantities on the right of the equal signs are defined as in (1), (2). Now, let $g : L \rightarrow L$ be defined by

$$33 \quad g(x_0, x(\cdot)) = \begin{pmatrix} \tilde{g}(x_0, x(\cdot)) \\ \tilde{P}_0(x_0, x(\cdot)) \\ \tilde{P}_f(x_0, x(\cdot)) \end{pmatrix}.$$

Then, from (30), (31) we see that $g(\cdot)$ is a continuously differentiable map, whose derivative we shall obtain shortly. We shall assume from now on, as we did for the Goodman-Lance version of the Newton-Raphson method that $(\partial g(z)/\partial z)^{-1}$ exists for all z in a sufficiently large subset of L .

34 **Exercise.** Show that with $g(\cdot)$ defined as in (33), (27) becomes

$$35 \quad (\overset{j+1}{x}_0 - \overset{j}{x}_0) + \int_{t_0}^t \frac{\partial f(\overset{j}{x}(s), s)}{\partial x} [\overset{j+1}{x}(s) - \overset{j}{x}(s)] ds - [\overset{j+1}{x}(t) - \overset{j}{x}(t)] \\ = -\overset{j}{x}_0 - \int_{t_0}^t f(\overset{j}{x}(s), s) ds + \overset{j}{x}(t), \quad t \in [t_0, t_f],$$

$$36 \quad P_0(\overset{j+1}{x}_0 - \overset{j}{x}_0) = -P_0 \overset{j}{x}_0 + y_0,$$

$$37 \quad P_f[\overset{j+1}{x}(t_f) - \overset{j}{x}(t_f)] = -P_f \overset{j}{x}(t_f) + y_f. \quad \blacksquare$$

Simplifying out (35)–(37), we now find that (27), with $g(\cdot)$ defined by (33), assumes the form

$$38 \quad \overset{j+1}{x}(t) = \overset{j+1}{x}_0 + \int_{t_0}^t \frac{\partial f(\overset{j}{x}(s), s)}{\partial x} [\overset{j+1}{x}(s) - \overset{j}{x}(s)] ds + \int_{t_0}^t f(\overset{j}{x}(s), s) ds, \\ t \in [t_0, t_f],$$

$$39 \quad P_0 \overset{j+1}{x}_0 = y_0,$$

$$40 \quad P_f \overset{j+1}{x}(t_f) = y_f.$$

To solve the system of equations (38)–(40), we transcribe it into a linear boundary-value problem, as follows:

$$41 \quad \frac{d}{dt} \overset{j+1}{x}(t) = \frac{\partial f(\overset{j}{x}(t), t)}{\partial x} [\overset{j+1}{x}(t) - \overset{j}{x}(t)] + f(\overset{j}{x}(t), t), \quad t \in [t_0, t_f],$$

$$42 \quad P_0 \overset{j+1}{x}(t_0) = y_0, \quad P_f \overset{j+1}{x}(t_f) = y_f.$$

43 **Exercise.** Show that any solution $(\overset{j+1}{x}_0, \overset{j+1}{x}(\cdot))$ of (38)–(40) satisfies (41), (42) and that any solution $\overset{j+1}{x}(\cdot)$ of (41) together with $\overset{j+1}{x}_0 = \overset{j+1}{x}(t_0)$ satisfy (38)–(40), i.e., show that the systems (38)–(40) and (41), (42) are equivalent. \blacksquare

We now summarize the preceding results by restating them in the form of an algorithm. Note that in the end we obtain an algorithm which could have been derived formally from algorithm (2.31), simply by replacing $[x_{i+1} - x_i]$ by $(d/dt)x(t)$ and x_i by $x(t)$.

- 44 Algorithm** (quasi-linearization version of Newton-Raphson method [M2], [B6]).

Step 0. Select an $\overset{0}{x}(\cdot) \in C'_v[t_0, t_f]$ such that $P_0\overset{0}{x}(t_0) = y_0$, $P_f\overset{0}{x}(t_f) = y_f$.

Comment. Strictly speaking, it is not necessary to choose $\overset{0}{x}(\cdot)$ such that $P_0\overset{0}{x}(t_0) = y_0$ and $P_f\overset{0}{x}(t_f) = y_f$; however, such a choice seems to lead to a better initial guess.

Comment. Suppose that x_0, x_f are such that $P_0x_0 = y_0$, $P_fx_f = y_f$; then we may set $\overset{0}{x}(t) = x_0 + [(t - t_0)/(t_f - t_0)](x_f - x_0)$, $t \in [t_0, t_f]$.

Step 1. Set $j = 0$.

Step 2. For $t \in [t_0, t_f]$, compute $f(\overset{j}{x}(t), t)$ and $\partial f(\overset{j}{x}(t), t)/\partial x$.

Step 3. For $t \in [t_0, t_f]$, compute $\overset{j+1}{x}(t)$ by solving (41), (42).

Step 4. Set $j = j + 1$ and go to step 2. ■

- 45 Exercise.** Adapt algorithm (44) for the solution of (1) with boundary conditions of the form $g_0(x(t_0)) = 0$, $g_f(x(t_f)) = 0$, where $g_0 : \mathbb{R}^v \rightarrow \mathbb{R}^\alpha$, $g_f : \mathbb{R}^v \rightarrow \mathbb{R}^{v-\alpha}$, $\alpha < v$, are continuously differentiable functions whose Jacobian matrices are of full rank for all $z \in \mathbb{R}^v$. ■

- 46 Exercise.** Modify algorithm (44) to obtain a quasi-Newton method of the form of algorithm (2.1.42). ■

- 47 Remark.** Note that the first observation made in remark (2.32) also applies to the two versions of the Newton-Raphson method we have just discussed, i.e., in the Goodman-Lance version we always have an approximation to a solution of (1), (2), i.e., $\overset{j}{x}(\cdot)$, which satisfies (1), while in the quasi-linearization version we always have an approximation to a solution of (1), (2) which satisfies (2). This difference between the two versions may be decisive when one has to make a choice between the two. Among people dealing with trajectory optimization problems, the quasi-linearization version appears to be in favor, since it seems less costly in computer time to solve the linear differential equation boundary-value problem (41), (42) at each iteration than the nonlinear differential equation (1), the linear differential equation (19) and the algebraic system (21), (22). ■

Since the linear boundary-value problem (41), (42) must be solved at each iteration of the quasi-linearization version, it is important to perform this

task efficiently. We shall now describe two methods for solving boundary-value problems of the form

$$48 \quad \frac{d}{dt} x(t) = A(t) x(t) + v(t), \quad t \in [t_0, t_1],$$

$$49 \quad P_0 x(t_0) = y_0, \quad P_1 x(t_1) = y_1,$$

where $x(t), v(t) \in \mathbb{R}^v$, $A(\cdot)$ is a piecewise continuous map from $[t_0, t_1]$ into the space of all $v \times v$ matrices,* $y_0 \in \mathbb{R}^\alpha$, $y_1 \in \mathbb{R}^{v-\alpha}$, and P_0 , P_1 are maximum-rank matrices of dimension $\alpha \times v$ and $(v - \alpha) \times v$, respectively ($\alpha < v$).

For $t, s \in [t_0, t_1]$, let $\Phi(t, s)$ be a $v \times v$ matrix defined as the solution of

$$50 \quad \frac{d}{dt} \Phi(t, s) = A(t) \Phi(t, s), \quad t, s \in [t_0, t_1], \quad \Phi(s, s) = I,$$

where I is the $v \times v$ identity matrix. Then we see that

$$51 \quad x(t_1) = \Phi(t_1, t_0) x(t_0) + \int_{t_0}^{t_1} \Phi(t_1, s) v(s) ds,$$

$$52 \quad x(t_0) = \Phi(t_0, t_1) x(t_1) + \int_{t_1}^{t_0} \Phi(t_0, s) v(s) ds.$$

The reader should recall that the matrix $\Phi(t, s)$ has the following two well-known properties:

$$53 \quad \Phi(t, s)^{-1} = \Phi(s, t), \quad \Phi(t, s') \Phi(s', s) = \Phi(t, s), \\ t, s, s' \in [t_0, t_1].$$

We can now use (49) together with either (51) or (52) to obtain a complete system of linear equations in $x(t_0)$ or in $x(t_1)$. Thus, to obtain the initial state $x(t_0)$ of a solution $x(\cdot)$ satisfying (48), (49) we may solve the system of equations,

$$54 \quad P_0 x(t_0) = y_0,$$

$$55 \quad P_1 \Phi(t_1, t_0) x(t_0) = - \int_{t_0}^{t_1} P_1 \Phi(t_1, s) v(s) ds + y_1.$$

Note that for this solution to be unique, the matrix

$$56 \quad \begin{pmatrix} P_0 \\ P_1 \Phi(t_1, t_0) \end{pmatrix}$$

must be nonsingular.

* In this space we can use the Frobenius norm defined by $\|A\|_F^2 = \sum_{i,j} a_{ij}^2$, where $A = (a_{ij})$.

Similarly, to obtain the terminal state of a solution $x(\cdot)$ satisfying (48), (49), we may solve the system of equations,

$$57 \quad P_0\Phi(t_0, t_f) x(t_f) = - \int_{t_f}^{t_0} P_0\Phi(t_0, s) v(s) ds + y_0,$$

$$58 \quad P_t x(t_f) = y_f.$$

Note that in the discrete case we could only obtain equations in the initial state x_0 (see (2.37), (2.38)). To obtain a set of equations in the terminal state x_k , in the discrete case, we would have to require that the matrices $(I + A_i)$ be nonsingular for $i = 0, 1, \dots, k - 1$. Even then, the need to invert all these matrices would make the construction of a system of equations in x_k rather unattractive. We thus encounter an interesting difference between discrete and continuous systems: A difference equation is easily solved in the forward direction, but not in the reverse direction; a differential equation can be solved in either direction with equal ease (provided, of course, that the solution does not grow extremely large or small).

When (48) is stable (i.e., $\Phi(t, s) \rightarrow 0$ as $t \rightarrow \infty$) we would choose to solve (54), (55), while when (48) is unstable (i.e., $\Phi(t, s) \rightarrow 0$ as $s \rightarrow -\infty$), we would choose to solve (57), (58). In either event, however, when $t_f - t_0$ is large, both the system (54), (55) and the system (57), (58) may be badly ill-conditioned due to the fact that the elements of $P_t\Phi(t_f, t_0)$ appear to be zero compared to the elements of P_0 , or vice versa, with a similar statement holding for (57), (58). Fortunately, (54), (55) is not the only set of equations one can set up to compute an $x(t_0)$ such that the solution $x(t; t_0, x(t_0))$ of (48) satisfies (49), and similarly for (57), (58). We shall now describe a method due to Abramov [A1] for constructing an alternative set of equations in $x(t_0)$. These equations are somewhat more difficult to set up than (54), (55), however, they are much better conditioned than (54), (55). We shall leave the adaptation of Abramov's method to obtain a substitute system for (57), (58) as an exercise for the reader.

We begin by describing an efficient way for setting up (55) (compare (2.39)). For $t \in [t_0, t_f]$, let $P(t)$ be a $(\nu - \alpha) \times \nu$ matrix which satisfies the adjoint equation,

$$59 \quad \frac{d}{dt} P(t) = -P(t) A(t), \quad t \in [t_0, t_f], \quad P(t_f) = P_f.$$

Then

$$60 \quad P(t) = P_f \Phi(t_f, t), \quad t \in [t_0, t_f],$$

and hence (55) becomes

$$61 \quad P(t_0) x(t_0) = - \int_{t_0}^{t_f} P(s) v(s) ds + y_f.$$

Note that (59) is a differential equation in fewer variables than (50), and hence (61) is easier to compute than (55) if one were to compute (55) by first solving (50).

- 62 **Exercise.** Obtain a formula analogous to (61) to replace (57). [Hint: Use the adjoint equation (59), but with a different boundary condition.] ■

Abramov's method consists in "normalizing" the matrix $P(t)$. Thus, for $t \in [t_0, t_1]$, let $Q(t)$ be a $(\nu - \alpha) \times \nu$ matrix defined by

$$63 \quad Q(t) = M(t) P(t),$$

where $P(t)$ is the solution of (59) and $M(t)$ is a $(\nu - \alpha) \times (\nu - \alpha)$ nonsingular normalization matrix which we shall determine in such a way as to ensure that

$$64 \quad Q(t_1) = P(t_1) = P_1,$$

$$65 \quad \frac{d}{dt} [Q(t) Q(t)^T] = 0.$$

The normalization we use is expressed by (65). To simplify notation, we shall use a dot over the letter notation to denote differentiation with respect to t , i.e., $\dot{Q}(t) = (d/dt) Q(t)$, etc. Then, from (65), we obtain

$$66 \quad \dot{Q}(t) Q(t)^T + Q(t) \dot{Q}(t)^T = 0;$$

and from (63),

$$67 \quad \dot{Q}(t) = \dot{M}(t) P(t) + M(t) \dot{P}(t).$$

Substituting from (67) into (66), we now obtain, making use of (59),

$$\begin{aligned} 68 \quad & [\dot{M}(t) P(t) + M(t) \dot{P}(t)] Q(t)^T + Q(t)[P(t)^T \dot{M}(t)^T + \dot{P}(t)^T M(t)^T] \\ &= [\dot{M}(t) P(t) - M(t) P(t) A(t)] Q(t)^T \\ &\quad + Q(t)[P(t)^T \dot{M}(t)^T - A(t)^T P(t)^T M(t)^T] \\ &= [\dot{M}(t) M(t)^{-1} Q(t) Q(t)^T - Q(t) A(t) Q(t)^T] \\ &\quad + [Q(t) Q(t)^T (M(t)^{-1})^T \dot{M}(t)^T - Q(t) A(t)^T Q(t)^T] \\ &= 0. \end{aligned}$$

Clearly, (68) is satisfied if we choose $M(t)$ so that

$$69 \quad \dot{M}(t) M(t)^{-1} Q(t) Q(t)^T = Q(t) A(t) Q(t)^T.$$

Now, from (67), (59) and (63),

$$70 \quad \dot{Q}(t) = \dot{M}(t) M(t)^{-1} Q(t) - Q(t) A(t).$$

Assuming that we choose $M(t)$ to satisfy (69), we then obtain

$$71 \quad \dot{Q}(t) = Q(t) A(t) Q(t)^T [Q(t) Q(t)^T]^{-1} Q(t) - Q(t) A(t), \\ t \in [t_0, t_f],^* \quad Q(t_f) = P_t.$$

Now, for $t \in [t_0, t_f]$, let $w(t) \in \mathbb{R}^{r-\alpha}$ be defined by

$$72 \quad w(t) = Q(t) x(t),$$

where $Q(t)$ is the solution of (71) and $x(t)$ is a solution of (48), (49). Then we see that

$$73 \quad \begin{aligned} \dot{w}(t) &= \dot{Q}(t) x(t) + Q(t) \dot{x}(t) \\ &= \{Q(t) A(t) Q(t)^T [Q(t) Q(t)^T]^{-1} Q(t) - Q(t) A(t)\} x(t) \\ &\quad + Q(t)[A(t) x(t) + v(t)] \\ &= Q(t) A(t) Q(t)^T [Q(t) Q(t)^T]^{-1} w(t) + Q(t) v(t), \\ &\quad t \in [t_0, t_f], \quad w(t_f) = P_t x(t_f) = y_t. \end{aligned}$$

Hence we see that we can replace the system of equation (54), (55) by the system of equations,

$$74 \quad P_0 x(t_0) = y_0,$$

$$75 \quad Q(t_0) x(t_0) = w(t_0),$$

which is obviously better conditioned than (54), (55) and which is computed by first solving (71) and then (73).

76 Exercise. Adapt Abramov's method to obtain a substitute set of linear equations in $x(t_f)$. When would you prefer to use Abramov's method to obtain a system of equations in $x(t_f)$ rather than in $x(t_0)$? ■

Note that Abramov's method is much more powerful in the continuous case, where we can take all the constraints on $x(t_f)$ and transform them simultaneously into constraints on $x(t_0)$, than in the discrete case, where we could take the constraints on x_k only, equation by equation, and transform them into constraints on x_0 . Obviously, the amount of ill-conditioning that can be removed by means of Abramov's method in the continuous case is much larger than in the discrete case.

We now turn to boundary-value problems which arise as a necessary condition of optimality in certain continuous optimal control problems.

* Note that $Q(t)Q(t)^T \equiv P_t P_t^T$.

In particular, let us consider the problem,

$$77 \quad \text{minimize} \quad \int_{t_0}^{t_f} f^0(x(t), u(t), t) dt + \varphi(x(t_f)),$$

subject to

$$78 \quad \frac{d}{dt} x(t) = f(x(t), u(t), t), \quad t \in [t_0, t_f],$$

$$79 \quad x(t_0) = \hat{x}_0, \quad g_f(x(t_f)) = 0,$$

where $f^0 : \mathbb{R}^\nu \times \mathbb{R}^\mu \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$, $f : \mathbb{R}^\nu \times \mathbb{R}^\mu \times \mathbb{R}^1 \rightarrow \mathbb{R}^\nu$, $\varphi : \mathbb{R}^\nu \rightarrow \mathbb{R}^1$ and $g_f : \mathbb{R}^\nu \rightarrow \mathbb{R}^\alpha$ satisfy the assumptions in (1.1.8), i.e., $\partial f^0/\partial x$, $\partial f^0/\partial u$, $\partial f/\partial x$, $\partial f/\partial u$ exist and, together with f^0 and f are piecewise continuous in t . Furthermore, we assume that $\partial g/\partial x$ exists and has maximum rank in a sufficiently large subset of \mathbb{R}^n . As in Section 2.5, we shall assume that the controls $u(\cdot)$ belong to $L_\infty^\mu[t_0, t_f]$.

Applying the Pontryagin maximum principle (1.2.35) to the problem (77)–(79), we find that if $\hat{u}(\cdot)$ is an optimal control and $\hat{x}(\cdot)$ is the corresponding optimal trajectory, then there exists a vector $\hat{\pi} \in \mathbb{R}^\alpha$ and a multiplier function $\hat{p} : [t_0, t_f] \rightarrow \mathbb{R}^\nu$ (usually referred to as the co-state), such that (assuming that the problem is nondegenerate, i.e., that $\hat{p}^0 = -1$ in (1.2.35) for this case) $\hat{u}(\cdot)$, $\hat{x}(\cdot)$, $\hat{\pi}$ and $\hat{p}(\cdot)$ satisfy

$$80 \quad \frac{d}{dt} \hat{x}(t) = f(\hat{x}(t), \hat{u}(t), t), \quad t \in [t_0, t_f],$$

$$81 \quad \hat{x}(t_0) = \hat{x}_0, \quad g_f(\hat{x}(t_f)) = 0,$$

$$82 \quad \frac{d}{dt} \hat{p}(t) = - \left(\frac{\partial f(\hat{x}(t), \hat{u}(t), t)}{\partial x} \right)^T \hat{p}(t) + \left(\frac{\partial f^0(\hat{x}(t), \hat{u}(t), t)}{\partial x} \right)^T,$$

$$83 \quad \hat{p}(t_f) = \left(\frac{\partial g(\hat{x}(t_f))}{\partial x} \right)^T \hat{\pi} - \left(\frac{\partial \varphi(\hat{x}(t_f))}{\partial x} \right)^T,$$

and for almost all $t \in [t_0, t_f]$,

$$84 \quad -f^0(\hat{x}(t), \hat{u}(t), t) + \langle \hat{p}(t), f(\hat{x}(t), \hat{u}(t), t) \rangle \geq -f^0(\hat{x}(t), u, t) + \langle \hat{p}(t), f(\hat{x}(t), u, t) \rangle \quad \text{for all } u \in \mathbb{R}^\mu.$$

Hence, we see that $\hat{u}(\cdot)$, $\hat{x}(\cdot)$, $\hat{p}(\cdot)$, and $\hat{\pi}$ must also be a solution of the following system of equations:

$$85 \quad \hat{x}_0 + \int_{t_0}^t f(x(s), u(s), s) ds - x(t) = 0, \quad t \in [t_0, t_f],$$

$$86 \quad g_f(x(t_f)) = 0,$$

$$87 \quad \left(\frac{\partial g_1(x(t_f))}{\partial x} \right)^T \pi - \left(\frac{\partial \varphi(x(t_f))}{\partial x} \right)^T + \int_{t_0}^t \left[- \left(\frac{\partial f(x(s), u(s), s)}{\partial x} \right)^T p(s) \right. \\ \left. + \left(\frac{\partial f^0(x(s), u(s), s)}{\partial x} \right)^T \right] ds - p(t) = 0, \quad t \in [t_0, t_f],$$

$$88 \quad - \left(\frac{\partial f^0(x(t), u(t), t)}{\partial u} \right)^T + \left(\frac{\partial f(x(t), u(t), t)}{\partial u} \right)^T p(t) = 0, \quad t \in [t_0, t_f].$$

Now, since we have assumed that the controls $u(\cdot)$ are in $L_\infty^\mu[t_0, t_f]$, solutions $x(\cdot)$ and $p(\cdot)$ of (85)–(88) must be continuous because of the nature of (85) and (87), i.e., $x(\cdot)$ and $p(\cdot)$ are in $C_v[t_0, t_f]$, the space of continuous functions from $[t_0, t_f]$ into \mathbb{R}^v . Let $L = \mathbb{R}^x \times C_v[t_0, t_f] \times C_v[t_0, t_f] \times L_\infty^\mu[t_0, t_f]$. We shall always partition elements $z \in L$ as follows:

$$89 \quad z = (\pi, x(\cdot), p(\cdot), u(\cdot)).$$

To make L a Banach space we now introduce a norm which we denote by $\|\cdot\|_L$ and define by

$$90 \quad \|z\|_L = [\|\pi\|^2 + (\sup_{t \in [t_0, t_f]} \|x(t)\|)^2 + (\sup_{t \in [t_0, t_f]} \|p(t)\|)^2 + \|u\|_\infty^2]^{1/2}.$$

Now, let $g_1 : L \rightarrow C_v[t_0, t_f]$, $g_2 : L \rightarrow C_v[t_0, t_f]$, $g_3 : L \rightarrow L_\infty^\mu[t_0, t_f]$ be defined by

$$91 \quad g_1(\pi, x(\cdot), p(\cdot), u(\cdot))(t)$$

$$= \dot{x}_0 + \int_{t_0}^t f(x(s), u(s), s) ds - x(t), \quad t \in [t_0, t_f],$$

$$92 \quad g_2(\pi, x(\cdot), p(\cdot), u(\cdot))(t)$$

$$= \left(\frac{\partial g_1(x(t_f))}{\partial x} \right)^T \pi - \left(\frac{\partial \varphi(x(t_f))}{\partial x} \right)^T + \int_{t_0}^t \left[- \left(\frac{\partial f(x(s), u(s), s)}{\partial x} \right)^T p(s) \right. \\ \left. + \left(\frac{\partial f^0(x(s), u(s), s)}{\partial x} \right)^T \right] ds - p(t), \quad t \in [t_0, t_f],$$

$$93 \quad g_3(\pi, x(\cdot), p(\cdot), u(\cdot))(t)$$

$$= - \left(\frac{\partial f^0(x(t), u(t), t)}{\partial u} \right)^T + \left(\frac{\partial f(x(t), u(t), t)}{\partial u} \right)^T p(t), \quad t \in [t_0, t_f].$$

If we now define $g : L \rightarrow L$ by

$$94 \quad g(\pi, x(\cdot), p(\cdot), u(\cdot)) = \begin{pmatrix} g_1(\pi, x(\cdot), p(\cdot), u(\cdot)) \\ g_2(\pi, x(\cdot), p(\cdot), u(\cdot)) \\ g_3(\pi, x(\cdot), p(\cdot), u(\cdot)) \\ g_f(x(t_f)) \end{pmatrix},$$

we find that solving (85)–(88) is equivalent to solving the equation $g(z) = 0$. It is not difficult to see that $g(\cdot)$ is continuously differentiable, and hence that (27) becomes, in this case, for $t \in [t_0, t_1]$,

$$95 \quad \int_{t_0}^t \left(\frac{\partial f(\dot{x}(s), \dot{u}(s), s)}{\partial x} [\dot{x}^{j+1}(s) - \dot{x}^j(s)] + \frac{\partial f(\dot{x}(s), \dot{u}(s), s)}{\partial u} [\dot{u}^{j+1}(s) - \dot{u}^j(s)] \right) ds \\ - [\dot{x}^{j+1}(t) - \dot{x}^j(t)] \\ = -g_1(\pi, \dot{x}(\cdot), \dot{p}(\cdot), \dot{u}(\cdot))(t)$$

$$96 \quad \left(\frac{\partial g_1(\dot{x}(t_1))}{\partial x} \right)^T [\dot{x}^{j+1} - \dot{x}^j] + \frac{\partial}{\partial x} \left[\left(\frac{\partial g_1(\dot{x}(t_1))}{\partial x} \right)^T \dot{x}^j - \left(\frac{\partial \varphi(\dot{x}(t_1))}{\partial x} \right)^T \right] \\ \times [\dot{x}^{j+1}(t) - \dot{x}^j(t)] + \int_{t_1}^t \left\{ - \left(\frac{\partial f(\dot{x}(s), \dot{u}(s), s)}{\partial x} \right)^T [\dot{p}^{j+1}(s) - \dot{p}^j(s)] \right. \\ - \frac{\partial}{\partial x} \left[\left(\frac{\partial f(\dot{x}(s), \dot{u}(s), s)}{\partial x} \right)^T \dot{p}^j(s) \right] [\dot{x}^{j+1}(s) - \dot{x}^j(s)] \\ - \frac{\partial}{\partial u} \left[\left(\frac{\partial f(\dot{x}(s), \dot{u}(s), s)}{\partial x} \right)^T \dot{p}^j(s) \right] [\dot{u}^{j+1}(s) - \dot{u}^j(s)] \\ \left. + \frac{\partial^2 f^0(\dot{x}(s), \dot{u}(s), s)}{\partial x^2} [\dot{x}^{j+1}(s) - \dot{x}^j(s)] \right. \\ \left. + \left(\frac{\partial^2 f^0(\dot{x}(s), \dot{u}(s), s)}{\partial u \partial x} \right)^T [\dot{u}^{j+1}(s) - \dot{u}^j(s)] \right\} ds - [\dot{p}^{j+1}(t) - \dot{p}^j(t)] \\ = -g_2(\pi, \dot{x}(\cdot), \dot{p}(\cdot), \dot{u}(\cdot))(t),$$

$$97 \quad - \left(\frac{\partial^2 f^0(\dot{x}(t), \dot{u}(t), t)}{\partial x \partial u} \right)^T [\dot{x}^{j+1}(t) - \dot{x}^j(t)] - \frac{\partial^2 f^0(\dot{x}(t), \dot{u}(t), t)}{\partial u^2} [\dot{u}^{j+1}(t) - \dot{u}^j(t)] \\ + \left(\frac{\partial f(\dot{x}(t), \dot{u}(t), t)}{\partial u} \right)^T [\dot{p}^{j+1}(t) - \dot{p}^j(t)] \\ + \frac{\partial}{\partial x} \left[\left(\frac{\partial f(\dot{x}(t), \dot{u}(t), t)}{\partial u} \right)^T \dot{p}^j(t) \right] [\dot{x}^{j+1}(t) - \dot{x}^j(t)] \\ + \frac{\partial}{\partial u} \left[\left(\frac{\partial f(\dot{x}(t), \dot{u}(t), t)}{\partial u} \right)^T \dot{p}^j(t) \right] [\dot{u}^{j+1}(t) - \dot{u}^j(t)] \\ = -g_3(\pi, \dot{x}(\cdot), \dot{p}(\cdot), \dot{u}(\cdot))(t),$$

$$98 \quad \frac{\partial g_1(\dot{x}(t_1))}{\partial x} [\dot{x}^{j+1}(t_1) - \dot{x}^j(t_1)] = -g_1(\dot{x}(t_1)).$$

(For a definition of $(\partial^2 f^0(x, u, t)/\partial u \partial x)^T$ and of $(\partial^2 f^0(x, u, t)/\partial x \partial u)^T$ see the footnote on p. 70.)

- 99 **Exercise.** Convert (95) and (96) into a system of differential equations and state the boundary values for this system. [Hint: Proceed as in (43).] ■

As in the discrete case (2.67)–(2.75), (97) may or may not be solvable uniquely for $\overset{j+1}{u}(t)$, $t \in [t_0, t_f]$ [compare (2.74)]. When it is not solvable, the system of equations (95)–(98) becomes exceedingly difficult to solve. We shall therefore assume that (98) can be solved uniquely for $\overset{j+1}{u}(t)$, in which case $\overset{j+1}{u}(t)$ must be an affine function in $\overset{j}{x}(t)$ and $\overset{j}{p}(t)$, i.e., $\overset{j+1}{u}(t) = \overset{j}{N}(t) \overset{j}{x}(t) + \overset{j}{M}(t) \overset{j}{p}(t) + \overset{j}{m}(t)$, where $\overset{j}{N}(t)$, $\overset{j}{M}(t)$ are $\mu \times \nu$ matrices and $\overset{j}{m}(t) \in \mathbb{R}^\mu$. In this case, substitution for $\overset{j+1}{u}(t)$ into (95) and (96) reduces our problem to solving a boundary-value problem of the following form (compare (2.76)–(2.78)):

$$100 \quad \frac{d}{dt} \overset{j+1}{x}(t) = \overset{j}{A}(t) \overset{j+1}{x}(t) + \overset{j}{B}(t) \overset{j+1}{p}(t) - \overset{j}{v}(t), \quad t \in [t_0, t_f],$$

$$101 \quad \frac{d}{dt} \overset{j+1}{p}(t) = \overset{j}{C}(t) \overset{j+1}{x}(t) + \overset{j}{D}(t) \overset{j+1}{p}(t) - \overset{j}{w}(t), \quad t \in [t_0, t_f],$$

$$102 \quad \overset{j+1}{x}(t_0) = \overset{j}{x}_0, \quad G_t \overset{j+1}{x}(t_f) = -\overset{j}{g},$$

$$103 \quad \overset{j+1}{p}(t_f) = G_t^T \overset{j+1}{\pi} + \overset{j}{\Phi}_t \overset{j+1}{x}(t_f) + \overset{j}{y}_t.$$

- 104 **Exercise.** Assuming that $\overset{j+1}{u}(t) = \overset{j}{N}(t) \overset{j}{x}(t) + \overset{j}{M}(t) \overset{j}{p}(t) + \overset{j}{m}(t)$, $t \in [t_0, t_f]$, obtain expressions for all the quantities in (100)–(103) (i.e., for $\overset{j}{A}(t)$, $\overset{j}{B}(t)$, etc.). ■

To simplify notation, just as in the discrete case, we now drop the superscripts j and $j + 1$ on all the symbols in (100)–(103). We now have (at least) two alternatives in choosing a method for solving (100)–(103). The first of these alternatives is favored in the U.S.S.R. and consists of utilizing the Abramov method discussed a little earlier in this section. To apply Abramov's method to (100)–(103), we must first eliminate $\overset{j+1}{\pi}$ (i.e., $\overset{j+1}{\pi}$) from (103). This can be done as follows: Let P_f be any $(\nu - \alpha) \times \nu$ full-rank matrix satisfying

$$105 \quad P_f G_t^T = 0.$$

Then (103) is obviously equivalent to the constraint

$$106 \quad P_f p(t_f) - P_f \Phi_t x(t_f) = y_f.$$

To construct such a matrix P_f we may proceed as follows: Without loss of generality, suppose that the first α columns of G_t are linearly independent

(G_f is an $\alpha \times \nu$ matrix which was assumed to be of full rank, since $G_f = \partial g_t(x)/\partial x$ for $x = \vec{x}(t_f)$). Let G_f' be the $\alpha \times \alpha$ matrix consisting of the first α columns of G_f ; then we may write $G_f = [G_f', G_f'']$ to express G_f in partitioned form. Now consider the equation

$$107 \quad [G_f', G_f''] \begin{pmatrix} P_f'^T \\ P_f''^T \end{pmatrix} = G_f' P_f'^T + G_f'' P_f''^T = 0,$$

where $P_f = [P_f', P_f'']$, with P_f' an $(\nu - \alpha) \times \alpha$ submatrix and P_f'' a $(\nu - \alpha) \times (\nu - \alpha)$ submatrix. (We have simply expanded the transpose of (105).) Hence,

$$108 \quad P_f' = -P_f'' G_f''^T (G_f')^{-1}.$$

Now let P_f'' be the $(\nu - \alpha) \times (\nu - \alpha)$ identity matrix. Then $P_f = (P_f', P_f'')$, with P_f' as defined by (108) has full rank and satisfies (105).

Abramov's method then yields the following set of equations in $x(t_0), p(t_0)$ (compare (74), (75)):

$$109 \quad x(t_0) = \hat{x}_0,$$

$$110 \quad Q_1(t_0) x(t_0) + Q_2(t_0) p(t_0) = h(t_0),$$

where the $\nu \times \nu$ matrices $Q_1(t_0), Q_2(t_0)$ are obtained by solving the differential equation,

$$111 \quad \frac{d}{dt} Q(t) = Q(t) \tilde{A}(t) Q(t)^T [Q(t) Q(t)^T]^{-1} - Q(t) A(t), \quad t \in [t_0, t_f],$$

where $Q(t)$ is a $\nu \times 2\nu$ matrix which partitions into $[Q_1(t), Q_2(t)] = Q(t)$ with $Q_1(t), Q_2(t)$ $\nu \times \nu$ submatrices, $\tilde{A}(t)$ is a $2\nu \times 2\nu$ matrix defined by

$$112 \quad \tilde{A}(t) = \begin{pmatrix} A(t) & B(t) \\ \hline \hline C(t) & D(t) \end{pmatrix},$$

and

$$113 \quad Q(t_f) = \begin{pmatrix} -P_f \Phi_f & P_f \\ \hline \hline G_f & 0 \end{pmatrix}.$$

The value of $h(t_0)$ is computed by solving the differential equation,

$$114 \quad \frac{d}{dt} h(t) = Q(t) \tilde{A}(t) Q(t)^T [Q(t) Q(t)^T]^{-1} h(t) + Q(t) d(t), \quad t \in [t_0, t_f]$$

where $h(t) \in \mathbb{R}^v$, $d(t) = (-v(t), -w(t))$ (column vector) and

$$115 \quad h(t_f) = \begin{pmatrix} y_f \\ -g \end{pmatrix}.$$

The second alternative in solving (100)–(103) is favored in the U.S. and consists of utilizing the possibility that the relation

$$116 \quad p(t) = K(t) x(t) + q(t), \quad t \in [t_0, t_f]$$

is valid, with $K(t)$ a $v \times v$ matrix and $q(t) \in \mathbb{R}^v$. Assuming that (116) is true, we find from (116), (100) and (101) that

$$\begin{aligned} 117 \quad \dot{p}(t) &= \dot{K}(t) x(t) + K(t) \dot{x}(t) + \dot{q}(t) \\ &= \dot{K}(t) x(t) + K(t)[A(t) x(t) + B(t) K(t) x(t) + B(t) q(t) - v(t)] + \dot{q}(t) \\ &= C(t) x(t) + D(t) K(t) x(t) + D(t) q(t) - w(t), \end{aligned}$$

where we have used a super dot to denote differentiation with respect to t . Equating terms on both sides of (117), we find that we may set

$$118 \quad \frac{d}{dt} K(t) = -K(t) A(t) - K(t) B(t) K(t) + D(t) K(t) + C(t), \quad t \in [t_0, t_f],$$

$$119 \quad \frac{d}{dt} q(t) = [-K(t) B(t) + D(t)] q(t) - K(t) v(t) - w(t), \quad t \in [t_0, t_f],$$

and from (103), we may set

$$120 \quad K(t_f) = \Phi_f, \quad q(t_f) = G_f^T \pi + y_f.$$

Alternatively, we may set

$$121 \quad K(t_f) = 0, \quad q(t_f) = p(t_f)$$

and retain (103) as a condition to be made use of later. Thus, (116) is possible provided the Riccati type of differential equation (118) has a solution with one of the possible boundary conditions given.

Assuming that the Riccati equation (118) does have a solution for the boundary conditions indicated, we can summarize the procedure for solving (100)–(103), based on its use, as follows:

122 **Algorithm** (solves system (100)–(103)).*

Step 1. For $t \in [t_0, t_f]$, compute $K(t)$ by solving (119) with $K(t_f) = \Phi_f$.

* When $G_f = 0$ in (103), the calculations below can be simplified considerably.

Step 2. For $t \in [t_0, t_f]$, compute the $\nu \times \nu$ matrix $Y(t)$ by solving

$$123 \quad \frac{d}{dt} Y(t) = [-K(t) B(t) + D(t)] Y(t), \quad t \in [t_0, t_f], \quad Y(t_0) = I,$$

where I is the $\nu \times \nu$ identity matrix.

Step 3. For $t \in [t_0, t_f]$, compute

$$124 \quad l(t) = \int_{t_0}^t Y(s) Y(s)^{-1} [-w(s) - K(s) v(s)] ds.$$

Comment. With the boundary conditions (120) (see (53)),*

$$125 \quad q(t) = Y(t)(G_f^T \pi + y_f) + l(t), \quad t \in [t_0, t_f].$$

Step 4. For $t \in [t_0, t_f]$, compute the $\nu \times \nu$ matrix $X(t)$ by solving

$$126 \quad \frac{d}{dt} X(t) = [A(t) + B(t) K(t)] X(t), \quad t \in [t_0, t_f], \quad X(t_0) = I,$$

where I is the $\nu \times \nu$ identity matrix.

Comment. With $p(t)$ defined by (116), we have

$$127 \quad \frac{d}{dt} x(t) = [A(t) + B(t) K(t)] x(t) + B(t) q(t) + v(t), \quad t \in [t_0, t_f].$$

Hence, because of (125) and (126) (and because of (53)),

$$128 \quad x(t) = X(t) \hat{x}_0 + \int_{t_0}^t X(s) X(s)^{-1} [B(s) Y(s) y_f + v(s)] ds \\ + \int_{t_0}^{t_f} X(s) X(s)^{-1} B(s) l(s) ds + \left[\int_{t_0}^t X(s) X(s)^{-1} Y(s) ds \right] G_f^T \pi.$$

Step 5. Compute

$$129 \quad x_f = X(t_f) \hat{x}_0,$$

$$130 \quad m_f = \int_{t_0}^{t_f} X(s) X(s)^{-1} [B(s) Y(s) y_f + v(s)] ds + \int_{t_0}^{t_f} X(s) X(s)^{-1} B(s) l(s) ds,$$

$$131 \quad M_f = \int_{t_0}^{t_f} X(s) X(s)^{-1} Y(s) ds.$$

Comment. We now make use of (103) to compute π .

* Since we do not know π , we cannot compute $q(t)$ by solving (119) directly. However, when $G_f = 0$, this difficulty disappears.

Step 6. Set

$$132 \quad \pi = -(G_t M_t G_t^T)^{-1} (g + G_t x_t + G_t m_t).$$

Step 7. For $t \in [t_0, t_1]$, compute $x(t)$ according to (128) and $q(t)$ according to (125).

Step 8. For $t \in [t_0, t_1]$, compute $p(t)$ according to (116). ■

133 **Exercise.** Develop an algorithm for solving (100)–(103) which uses (118), (119) with the boundary conditions (121). ■

134 **Remark.** Both the Abramov method and (122) involve the solution of a Riccati type of differential equation. It has been claimed in the U.S.S.R. that it is much more likely that (111) will have a solution than that (118) will have a solution, and that the system (109), (110) is usually better conditioned than the system of equations implied by (132). In the U.S., critics of the Abramov method point out, however, that (111) is considerably harder to solve than (118) and that, in general, in the Abramov method one deals with differential equations of larger dimension than in (122). It may well be, however, that this geographically based difference of opinion is due mostly to a preference resulting from familiarity with one method or the other, rather than to extensive computational experience, since there seem to be no published comparisons of the two methods. The interested reader may wish to conduct an experiment or two for himself so as to decide which of the two approaches works better for the particular class of problems he is dealing with. ■

To conclude this section, we consider an important special case of the problem (77)–(79)—the linear-dynamics, quadratic-cost-regulator problem—which can be solved completely, as we shall soon see. In the process of solving this problem, we shall also establish a particular case of (118) for which a solution exists. We base our presentation on a paper by Bucy [B5]. Thus, consider the optimal control problem

$$135 \quad \text{minimize} \quad \frac{1}{2} \left[\langle x(T), \Phi x(T) \rangle + \int_0^T \langle x(t), Cx(t) \rangle + \langle u(t), Ru(t) \rangle dt \right],$$

subject to

$$136 \quad \frac{d}{dt} x(t) = Ax(t) + Fu(t), \quad t \in [0, T], \quad x(0) = x_0,$$

where, as before, $x(t) \in \mathbb{R}^\nu$, $u(t) \in \mathbb{R}^\mu$, and A, F, Φ, C , and R are constant matrices of dimension $\nu \times \nu$, $\nu \times \mu$, $\nu \times \nu$, $\nu \times \nu$, and $\mu \times \mu$, respectively. We shall assume that the matrix R is symmetric and positive definite, and that the matrices Φ and C are symmetric and positive semidefinite. (Note

that since all the matrices in (135), (136) are time-invariant, we lose no generality by setting $t_0 = 0$, $t_f - t_0 = T$; compare (77)–(79). For the problem (135), (136), the conditions (80)–(84) are not only necessary, but also sufficient, and they assume the specific form,

$$137 \quad \frac{d}{dt} \hat{x}(t) = A\hat{x}(t) + F\hat{u}(t), \quad t \in [0, T],$$

$$138 \quad \frac{d}{dt} \hat{p}(t) = C\hat{x}(t) - A^T\hat{p}(t), \quad t \in [0, T],$$

$$139 \quad \hat{x}(0) = \hat{x}_0, \quad \hat{p}(T) = -\Phi\hat{x}(T),$$

$$140 \quad -R\hat{u}(t) + F^T\hat{p}(t) = 0, \quad t \in [0, T],$$

with (140) being obtained by carrying out the maximization indicated in (84). Solving (140) for $\hat{u}(t)$ and substituting into (137), we find that $\hat{x}(\cdot)$, $\hat{p}(\cdot)$ must be a solution of the system of differential equations,

$$141 \quad \frac{d}{dt} x(t) = Ax(t) + Bp(t), \quad t \in [0, T],$$

$$142 \quad \frac{d}{dt} p(t) = Cx(t) - A^Tp(t), \quad t \in [0, T],$$

$$143 \quad x(0) = \hat{x}_0, \quad p(T) = -\Phi x(T),$$

where $B = FR^{-1}F^T$. Note that B is symmetric and positive semidefinite.

Now, for $t, s \in [0, T]$, let $X(t, s)$ be a $2\nu \times 2\nu$ matrix defined by

$$144 \quad \frac{d}{dt} X(t, s) = \begin{pmatrix} A & B \\ -C & -A^T \end{pmatrix} X(t, s), \quad t, s \in [0, T], \quad X(s, s) = I,$$

where I is the $2\nu \times 2\nu$ identity matrix. We shall partition the matrix $X(t, s)$ into four $\nu \times \nu$ blocks as follows:

$$145 \quad X(t, s) = \begin{pmatrix} X_{11}(t, s) & X_{12}(t, s) \\ X_{21}(t, s) & X_{22}(t, s) \end{pmatrix}.$$

We now see that for any $t \in [0, T]$,

$$146 \quad x(T) = X_{11}(T, t)x(t) + X_{12}(T, t)p(t),$$

$$147 \quad p(T) = X_{21}(T, t)x(t) + X_{22}(T, t)p(t).$$

Now because of (143), we obtain,

$$148 \quad p(T) + \Phi x(T) \\ = [\Phi X_{11}(T, t) + X_{21}(T, t)] x(t) + [\Phi X_{12}(T, t) + X_{22}(T, t)] p(t) = 0,$$

and hence, for $t \in [0, T]$,

$$149 \quad p(t) = -[\Phi X_{12}(T, t) + X_{22}(T, t)]^{-1} [\Phi X_{11}(T, t) + X_{21}(T, t)] x(t),$$

provided that the inverse of

$$150 \quad E(t) = \Phi X_{12}(T, t) + X_{22}(T, t)$$

exists for all $t \in [0, T]$. (Note that $E(T)$ is nonsingular, since $E(T) = I$, the $\nu \times \nu$ identity matrix.) Let us suppose for the moment that $E(t)$ is nonsingular for all $t \in [0, T]$. Then, for $t \in [0, T]$, defining the matrix $K(t)$ by

$$151 \quad K(t) = -[\Phi X_{12}(T, t) + X_{22}(T, t)]^{-1} [\Phi X_{11}(T, t) + X_{21}(T, t)],$$

we have

$$152 \quad p(t) = K(t) x(t), \quad t \in [0, T],$$

and hence (141) becomes

$$153 \quad \frac{d}{dt} x(t) = [A + BK(t)] x(t), \quad t \in [0, T], \quad x(0) = x_0.$$

Now, from (151),

$$154 \quad [\Phi X_{12}(T, t) + X_{22}(T, t)] K(t) = -[\Phi X_{11}(T, t) + X_{21}(T, t)].$$

Differentiating both sides of (154) with respect to t , we obtain

$$155 \quad [\Phi \dot{X}_{12}(T, t) + \dot{X}_{22}(T, t)] K(t) + [\Phi X_{12}(T, t) + X_{22}(T, t)] \dot{K}(t) \\ = -[\Phi \dot{X}_{11}(T, t) + \dot{X}_{21}(T, t)], \quad t \in [0, T],$$

where the super dot denotes differentiation.

Now, it is well-known that

$$156 \quad \frac{d}{dt} X(T, t) = X(T, t) \begin{pmatrix} -A & -B \\ -C & A^T \end{pmatrix}, \quad t \in [0, T],$$

and hence, in conjunction with (145), we now obtain for (155),

$$\begin{aligned} 157 \quad & \Phi[-X_{11}(T, t) B + X_{12}(T, t) A^T] K(t) + [-X_{21}(T, t) B + X_{22}(T, t) A^T] K(t) \\ & + [\Phi X_{12}(T, t) + X_{22}(T, t)] \dot{K}(t) \\ & = -\Phi[-X_{11}(T, t) A - X_{12}(T, t) C] - [-X_{21}(T, t) A - X_{22}(T, t) C], \end{aligned}$$

which yields, upon rearranging terms and multiplying both sides by $E(t)^{-1}$ (see (150)),

$$158 \quad \frac{d}{dt} K(t) = -K(t) A - A^T K(t) - K(t) B K(t) + C, \quad t \in [0, T], \quad K(T) = -\Phi,$$

where the boundary condition $K(T) = -\Phi$ is obtained directly from (151) by inspection. Note that (158) is identical with (118) for the case $A(t) = A$, $B(t) = B$, $C(t) = C$, $D(t) = A^T$. We shall now show that the matrix $E(t)$ in (150) is indeed nonsingular for all $t \in [0, T]$, and hence that (158) has a solution $K(t)$ for $t \in [0, T]$ which is given explicitly by (151).

159 **Theorem.** For $t \in [0, T]$, the matrix $E(t)$ defined by (150) is nonsingular.

Proof. We begin by observing that if $x'(t)$, $p'(t)$ satisfy (141), (142) for $t \in [t', T]$, with $t' \in [0, T]$, $x'(t') = x_0'$, $p'(t') = p_0'$, then

$$\begin{aligned} 160 \quad & \frac{d}{dt} \langle x'(t), p'(t) \rangle = \left\langle \frac{d}{dt} p'(t), x'(t) \right\rangle + \left\langle p'(t), \frac{d}{dt} x'(t) \right\rangle \\ & = \langle Cx'(t) - A^T p'(t), x'(t) \rangle + \langle p'(t), Ax'(t) + Bp'(t) \rangle \\ & = \langle x'(t), Cx'(t) \rangle + \langle p'(t), Bp'(t) \rangle. \end{aligned}$$

Hence, we must have

$$161 \quad \langle x'(T), p'(T) \rangle = \langle x_0', p_0' \rangle + \int_{t'}^T \langle x'(t), Cx'(t) \rangle + \langle p'(t), Bp'(t) \rangle dt.$$

Also, we observe that $E(T) = I$, the $\nu \times \nu$ identity matrix, and hence is nonsingular. Now suppose that $E(t')$ is singular for some $t' \in [0, T]$; then there must exist a nonzero vector $p_0' \in \mathbb{R}^\nu$, such that

$$162 \quad [\Phi X_{12}(T, t') + X_{22}(T, t')] p_0' = 0.$$

Now let $x'(t)$, $p'(t)$ satisfy (141), (142) for $t \in [t', T]$ with $x'(t') = 0$, $p'(t') = p_0'$. Then we must have

$$163 \quad x'(T) = X_{12}(T, t') p_0', \quad p'(T) = X_{22}(T, t') p_0',$$

and hence, because of (162),

$$164 \quad \Phi x'(T) + p'(T) = 0.$$

However, by (161), we must also have

$$165 \quad 0 = -\langle p'(T), x'(T) \rangle + \langle 0, p_0' \rangle + \int_{t'}^T \langle x'(t), Cx'(t) \rangle + \langle p'(t), Bp'(t) \rangle dt.$$

Making use of (164), we now obtain

$$166 \quad 0 = \langle x'(T), \Phi x'(T) \rangle + \int_{t'}^T \langle x'(t), Cx'(t) \rangle + \langle p'(t), Bp'(t) \rangle dt.$$

By assumption, the matrices B , C , and Φ are all symmetric and positive semidefinite. Consequently, we must have

$$167 \quad \Phi x'(T) = 0, \quad Cx'(t) = 0, \quad Bp'(t) = 0 \quad \text{for } t \in [t', T],$$

for (166) to hold. But $Cx'(t) = 0$ for $t \in [t', T]$ implies that

$$168 \quad \frac{d}{dt} p'(t) = -A^T p'(t), \quad t \in [t', T],$$

and hence,

$$169 \quad p'(T) = \exp[-(T - t') A^T] p_0' \neq 0.$$

But by (164) and (167),

$$170 \quad p'(T) = -\Phi x'(T) = 0,$$

which contradicts (169). Consequently, the only p_0' for which (162) can hold is the zero vector, i.e., $E(t)$ is nonsingular for all $t \in [0, T]$. ■

- 171 **Exercise.** Show that the matrix $K(t)$ defined by (158) is symmetric and negative semidefinite for all $t \in [0, T]$. ■

This concludes our discussion of unconstrained optimization and boundary-value problems as they occur in continuous optimal control.

4

EQUALITY AND INEQUALITY CONSTRAINTS

4.1 Penalty Function Methods

We now return to the constrained minimization problems (1.1.1), (1.1.3) and (1.1.8). Most of our results in this section will be for the nonlinear programming problem (1.1.1), with the discrete optimal control problem (1.1.3) being treated as a special case of (1.1.1). The treatment of the continuous optimal control problem (1.1.8) is considerably more difficult and more involved than that of the preceding finite dimensional problems and we shall content ourselves with only a descriptive presentation of some of the more important results.

For the purpose of presenting penalty function methods, it is convenient to write the nonlinear programming problem (1.1.1) in the following, more compact form:

$$1 \quad (\text{CP}) \quad \min\{f^0(z) \mid z \in C \subset \mathbb{R}^n\},$$

where $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a continuously differentiable function and C is a closed subset of \mathbb{R}^n .

The underlying idea behind penalty function methods is that of solving the problem (1), which we shall call (CP), by constructing a sequence of points $z_i \in \mathbb{R}^n$ which are optimal for a sequence of unconstrained minimization problems of the form

$$2 \quad (\text{UP})_i \quad \min\{f^0(z) + p_i(z) \mid z \in \mathbb{R}^n\}, \quad i = 0, 1, 2, \dots,$$

which we shall call $(\text{UP})_i$, the $(\text{UP})_i$ being so constructed that the $z_i \rightarrow \hat{z} \in C$ as $i \rightarrow \infty$ and \hat{z} is optimal for (CP). There are two major, basically different

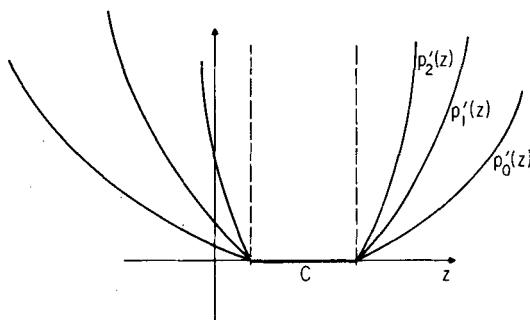
approaches to constructing the problems $(UP)_i$, or to be more specific, the *penalty functions* $p_i(\cdot)$. The first approach, which is known as the exterior penalty function method, was proposed by Courant in 1943 [C5], at a time when almost all the well-known nonlinear programming algorithms of today were still to be invented. In the exterior penalty function method, the functions $p_i(\cdot)$ are chosen so as to make it progressively more and more expensive to pick a point not in C , and so that either $p_i(z) = 0$ for all $z \in C$ or so that $p_i(z) \rightarrow 0$ as $i \rightarrow \infty$ for all $z \in C$. In presenting exterior penalty function methods, we shall draw upon the relatively recent work of Zangwill [Z3], rather than upon the early work of Courant which was specifically directed towards the solution of problems in differential equations.

The second approach is called the interior penalty function method. In it, the penalty functions $p_i(\cdot)$ are chosen so that the z_i which are optimal for the $(UP)_i$ must all belong to the interior of C , $f^0(z) + p_i(z) \rightarrow f^0(z)$ as $i \rightarrow \infty$ for all z in the interior of C , and $f^0(z) + p_i(z) \rightarrow \infty$ as the point z approaches the boundary of C from within. Because of this last property of the $p_i(\cdot)$, interior penalty function methods are also known as *barrier* methods, since they repel the points \bar{z}^j , $j = 0, 1, 2, \dots$, constructed by any one of the algorithms discussed in the preceding chapter, in the process of solving $(UP)_i$ from the boundary of C . Because of this, the initial point \bar{z}^0 used for solving $(UP)_i$ must be picked in the interior of C . In our exposition of interior penalty function methods we shall draw upon the work of the two best known contributors in this area, Fiacco and McCormack [F1], [F2]. Incidentally, their book [F2] is the most comprehensive exposition of penalty function methods presently available and the reader is referred to it for further reading. As we shall also see, there are many situations where interior and exterior penalty function methods can be combined profitably to yield a mixed approach. For additional reading on this subject the reader is referred to [F2] and to [L3].

In our presentation, we shall proceed as follows: First we shall establish the properties with which penalty functions must be endowed in order to guarantee that the z_i which are optimal for the $(UP)_i$ converge to a point which is optimal for (CP) . After that we shall examine the computational aspects of penalty function methods. Finally, we shall discuss their application to optimal control problems.

Exterior Penalty Function Methods

The reason for switching to the plural, i.e., for saying methods rather than method is that each choice of a family of penalty functions $p_i(\cdot)$ in (2) results in a different algorithm. Thus, there is a whole class of exterior penalty function methods.



Exterior penalty functions.

- 3 Definition.** Let C be a closed subset of \mathbb{R}^n . A sequence of continuous functions $p_i' : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, 2, \dots$, is called a sequence of *exterior penalty functions* for the set C if

- 4 $p_i'(z) = 0 \quad \text{for all } z \in C, \quad i = 0, 1, 2, \dots,$
- 5 $p_i'(z) > 0 \quad \text{for all } z \notin C, \quad i = 0, 1, 2, \dots,$
- 6 $p_{i+1}'(z) > p_i'(z) \quad \text{for all } z \notin C, \quad i = 0, 1, 2, \dots,$
- 7 $p_i'(z) \rightarrow \infty \quad \text{as } i \rightarrow \infty \quad \text{for all } z \notin C.$

■

Now, consider the problem (CP) defined in (1) and let $p_i'(\cdot)$, $i = 0, 1, 2, \dots$, be a sequence of exterior penalty functions for the set C in (1). We introduce a sequence of unconstrained minimization problems $(UP)_i^e$, defined as follows:

$$8 \quad (UP)_i^e \min\{f^0(z) + p_i'(z) \mid z \in \mathbb{R}^n\}, \quad i = 0, 1, 2, \dots.$$

To ensure that both the problem (CP) and the problems $(UP)_i^e$ have a solution, it suffices to make the following assumption:

- 9 Assumption.** We shall suppose that there is a point $z' \in C$ such that the set $Z' = \{z \mid f^0(z) \leq f^0(z')\}$ is compact. ■
- 10 Exercise.** Show that under assumption (9), (CP) must have a solution and that all the points which are optimal for (CP) must be in Z' . Show also, that under assumption (9), for $i = 0, 1, 2, \dots$, the set $\{z \mid f^0(z) + p_i(z) \leq f^0(z')\}$ is compact and is contained in Z' . Hence, show that, for $i = 0, 1, 2, \dots$, each $(UP)_i^e$ must have a solution and that all the points which are optimal for $(UP)_i^e$ must also be contained in Z' . ■

We shall now establish two preliminary lemmas which will be followed by the convergence theorem for exterior penalty function methods.

11 Lemma. Consider the problems (P) and $(UP)_i^e$. Let \bar{m} and \bar{m}_i , $i = 0, 1, 2, \dots$, be defined as

$$12 \quad \bar{m} = \min\{f^0(z) \mid z \in C\},$$

$$13 \quad \bar{m}_i = \min\{f^0(z) + p_i'(z) \mid z \in \mathbb{R}^n\}.$$

Then we must have $\bar{m}_0 \leq \bar{m}_1 \leq \bar{m}_2 \leq \dots \leq \bar{m}$.

Proof. For $i = 0, 1, 2, 3, \dots$, let $z_i \in \mathbb{R}^n$ be optimal for $(UP)_i^e$, i.e.,

$$14 \quad \bar{m}_i = f^0(z_i) + p_i'(z_i), \quad i = 0, 1, 2, \dots$$

Then, from (4) and (6), it follows that for $i = 0, 1, 2, \dots$,

$$15 \quad \bar{m}_i \leq f^0(z_{i+1}) + p_i'(z_{i+1}) \leq f^0(z_{i+1}) + p_{i+1}'(z_{i+1}) = \bar{m}_{i+1}.$$

Next, from (4) and (13), we conclude that

$$16 \quad \bar{m}_i \leq f^0(z) + p_i'(z) = f^0(z) \quad \text{for all } z \in C,$$

and hence, we must have

$$17 \quad \bar{m}_i \leq \min\{f^0(z) \mid z \in C\} = \bar{m}. \quad \blacksquare$$

18 Lemma. Let C be a closed subset of \mathbb{R}^n and let $p_i'(\cdot)$, $i = 0, 1, 2, \dots$, be a sequence of exterior penalty functions for C . Suppose that z_i , $i = 0, 1, 2, \dots$, is a sequence in \mathbb{R}^n which converges to a point \hat{z} . If $z_i \notin C$ for $i = 0, 1, 2, \dots$, and there exists a bound $M < \infty$ such that $p_i'(z_i) \leq M$ for $i = 0, 1, 2, \dots$, then $\hat{z} \in C$.

Proof. We shall construct a contradiction. Suppose that \hat{z} is not in C . By assumption, there exists an $M < \infty$ such that $p_i'(z_i) \in (0, M]$ for $i = 0, 1, 2, \dots$. Since $\hat{z} \notin C$, it follows from (7) that $p_i'(\hat{z}) \rightarrow \infty$ as $i \rightarrow \infty$. Consequently, there exists an integer N' such that $p_{N'}'(\hat{z}) > 2M$. Furthermore, since $p_{N'}'(\cdot)$ is continuous, there exists an open ball B with center \hat{z} such that

$$19 \quad p_{N'}'(z) > \frac{3M}{2} \quad \text{for all } z \in B.$$

Note that $B \cap C$ is the empty set, since $p_{N'}'(z) = 0$ for all $z \in C$. Since $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, there is an integer N'' such that $z_i \in B$ for all $i \geq N''$. Let $N = \max\{N', N''\}$; then for all $i \geq N$, $z_i \in B$ and in addition, from (6) and (19),

$$20 \quad p_i'(z_i) \geq p_{N'}'(z_i) \geq \frac{3M}{2},$$

which contradicts our assumption that $p_i'(z_i) \leq M$. Consequently, we must have $\hat{z} \in C$. ■

We are now ready to establish one of our main results.

- 21 **Theorem.** Consider the sequence of problems $(UP)_i^e$ defined in (8) and suppose that assumption (9) is satisfied. If z_i is optimal for $(UP)_i^e$, $i = 0, 1, 2, \dots$, then any accumulation point of the sequence $\{z_i\}_{i=0}^\infty$ is optimal for the problem (CP) defined in (1).

Proof. Without loss of generality, we may assume that the sequence $\{z_i\}_{i=0}^\infty$ converges to a point \hat{z} . Now, there are two possibilities: (i) there is an integer $N \geq 0$ such that $z_i \in C$ for all $i \geq N$; (ii) there is no such integer N , in which case $\{z_i\}_{i=0}^\infty$ must contain an infinite subsequence of points z_i , $i \in K \subset \{0, 1, 2, \dots\}$, such that $z_i \notin C$. Suppose (i) is true, then, since C is closed, $\hat{z} \in C$, and (see (11)) since for every $i \geq N$, $p_i'(z_i) = 0$, we must have $\bar{m}_i = \bar{m}$ for $i \geq N$ (since $f^0(z) = f^0(z) + p_i'(z)$ for all $z \in C$), i.e., z_i is also optimal for the problem (CP) for $i \geq N$. Since $f^0(\cdot)$ is continuous, $z_i \rightarrow \hat{z}$, and $f^0(z_i) = \bar{m}$, $i > N$, we must have $f^0(\hat{z}) = \bar{m}$, i.e., \hat{z} is optimal for (CP).

Now suppose that (ii) is true. Since $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, $i \in K$, and $f^0(\cdot)$ is continuous, $f^0(z_i) \rightarrow f^0(\hat{z})$ as $i \rightarrow \infty$, and hence there exists a bound $M' \in (0, \infty)$ such that

$$22 \quad |f^0(z_i)| \leq M' \quad \text{for all } i \in K.$$

Consequently, since $\bar{m}_i = f^0(z_i) + p_i'(z_i) \leq \bar{m}$, for $i \in K$, by lemma (11), we must have

$$23 \quad p_i'(z_i) \leq \bar{m} + M' \quad \text{for all } i \in K,$$

and therefore the sequence $p_i'(z_i)$, $i \in K$, is bounded by $M = \bar{m} + M'$. It now follows from lemma (18) that $\hat{z} \in C$ and therefore, $f^0(\hat{z}) \geq \bar{m}$. But for $i \in K$, we have $f^0(z_i) \leq \bar{m} - p_i'(z_i)$ and $p_i'(z_i) > 0$. Hence, we obtain, upon letting $i \rightarrow \infty$, $f^0(\hat{z}) \leq \bar{m}$. Consequently, we must have $f^0(\hat{z}) = \bar{m}$, i.e., \hat{z} must be optimal for (CP). ■

- 24 **Exercise.** Suppose that the sequence $\{z_i\}_{i=0}^\infty$ is defined as in theorem (21). Show that if $z_j \in C$, then $z_i \in C$ for all $i \geq j$. ■

- 25 **Exercise.** Suppose that $f^j : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $j = 1, 2, \dots, m$, are continuous functions, and that

$$26 \quad C = \{z \mid f^j(z) \leq 0, j = 1, 2, \dots, m\}.$$

Show that the functions $p_i' : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, 2, \dots$, defined by

$$27 \quad p_i'(z) = \alpha_i \sum_{j=1}^m [\max\{f^j(z), 0\}]^g, \quad i = 0, 1, 2, \dots,$$

where $\beta \geq 1$ and α_i , $i = 0, 1, 2, \dots$, is a strictly increasing sequence of positive numbers which tends to ∞ as $i \rightarrow \infty$, is a sequence of exterior penalty functions for the set C in (26). ■

- 28 **Exercise.** Suppose that the functions $f^j(\cdot)$ in (26) are continuously differentiable and that $\beta \geq 2$. Show that under this assumption the functions $p_i'(\cdot)$ defined by (27) are also continuously differentiable. ■

- 29 **Exercise.** Suppose that $r : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is continuous and that

$$30 \quad C = \{z \mid r(z) = 0\}.$$

Show that the functions $p_i'(\cdot)$, $i = 0, 1, 2, \dots$, defined by

$$31 \quad p_i'(z) = \alpha_i \|r(z)\|^\beta = \alpha_i \left(\sum_{j=1}^m (r^j(z))^2 \right)^{\beta/2},$$

where $\beta \geq 1$ and α_i , $i = 0, 1, 2, \dots$, is a strictly increasing sequence of positive numbers which tend to ∞ as $i \rightarrow \infty$, is a sequence of exterior penalty functions for the set C in (30). Also, show that $p_i'(\cdot)$ is continuously differentiable whenever $r(\cdot)$ is continuously differentiable and $\beta \geq 2$. ■

- 32 **Exercise.** Suppose that $\{p_i'(\cdot)\}_{i=0}^\infty$ is a sequence of exterior penalty functions for the set C and that $\{\tilde{p}_i'(\cdot)\}_{i=0}^\infty$ is a sequence of exterior penalty functions for the set \tilde{C} . Show that $\{p_i'(\cdot) + \tilde{p}_i'(\cdot)\}_{i=0}^\infty$ is a sequence of exterior penalty functions for the set $C \cap \tilde{C}$, and that $\{\min\{p_i'(\cdot), \tilde{p}_i'(\cdot)\}\}_{i=0}^\infty$ is a sequence of exterior penalty functions for the set $C \cup \tilde{C}$. ■

- 33 **Remark.** Exterior penalty functions can be used not only to transform a constrained minimization problem into a sequence of unconstrained minimization problems, but also to remove constraints which certain algorithms will not accept. Suppose that we wish to solve the general nonlinear programming problem (1.1.1), i.e., $\min\{f^0(z) \mid r(z) = 0, f(z) \leq 0\}$ and that the function $r(\cdot)$ is not affine. Virtually all the methods to be described later are inapplicable to this problem because of the nonlinear equality constraint $r(z) = 0$. However, suppose that $\{p_i'(\cdot)\}_{i=0}^\infty$ is a sequence of exterior penalty functions for the set $\{z \mid r(z) = 0\}$; then, under suitable assumptions, we can use a number of methods to solve the sequence of problems, $\min\{f^0(z) + p_i'(z) \mid f(z) \leq 0\}$, $i = 0, 1, 2, \dots$, to obtain a solution for the original problem. By removing only some of the constraints with penalty functions rather than all, we may hope to obtain better numerical behavior (better “conditioning”) in our computations. ■

- 34 **Exercise.** Consider the problem (1.1.1), i.e., $\min\{f^0(z) \mid r(z) = 0, f(z) \leq 0\}$, where all the functions are continuously differentiable. Let $\{p_i'(\cdot)\}_{i=0}^\infty$ be

a sequence of exterior penalty functions for the set $\{z \mid r(z) = 0\}$, and consider the sequence of problems $(P_i) : \min\{f^0(z) + p_i'(z) \mid f(z) \leq 0\}, i = 0, 1, 2, \dots$. Give sufficient conditions for the problems (1.1.1) and (P_i) to have solutions. Show that if z_i is an optimal point for the problem (P_i) , then any accumulation point of the sequence $\{z_i\}_{i=0}^\infty$ is an optimal point for the problem (1.1.1). ■

35 Exercise. Consider the problem

$$36 \quad (\text{CP})_L \quad \min\{f^0(z) \mid z \in C \subset L\},$$

where L is a Banach space, C is a closed subset of L , and $f^0 : L \rightarrow \mathbb{R}^1$ is a continuous function. Show that all the results presented so far remain valid for the problem $(\text{CP})_L$ provided we replace \mathbb{R}^n by L in (3), (8), (10), (11), (18), (21) and all the preceding exercises. ■

37 Exercise. Consider the continuous optimal control problem

$$38 \quad \text{minimize} \quad \int_0^T \|x(t) - x'(t)\|^2 dt,$$

subject to

$$39 \quad \frac{d}{dt} x(t) = Ax(t) + Bu(t), \quad t \in [0, T], \quad x(t) \in \mathbb{R}^v, \quad u(t) \in \mathbb{R}^u, \quad x(0) = \hat{x}_0,$$

$$40 \quad \int_0^T \|u(t)\|^2 dt \leq v,$$

where the matrices A, B are constant, and $x'(\cdot)$ is a given nominal trajectory. Show that by setting $L = L_2^u[0, T]$,

$$41 \quad C = \{u(\cdot) \in L_2^u[0, T] \mid \|u\|_2^2 \leq v\},$$

and by defining $p_i'(\cdot)$ as

$$42 \quad p_i'(u) = \alpha_i \max\{\|u\|_2^2 - v, 0\},$$

where $\alpha_i, i = 0, 1, 2, \dots$, is a sequence of strictly increasing positive numbers which tend to ∞ as $i \rightarrow \infty$, we obtain a sequence of unconstrained problems,

$$43 \quad \begin{aligned} \text{minimize} \quad & \int_0^T \|x(t; u) - x'(t)\|^2 dt \\ & + \alpha_i \max \left\{ \int_0^T [\|u(t)\|^2 - \frac{v}{T}] dt, 0 \right\}, \end{aligned}$$

subject to

$$u(\cdot) \in L_2^u[0, T], \quad i = 0, 1, 2, \dots,$$

whose solutions $\dot{u}^i(\cdot)$ and corresponding trajectories $x^i(\cdot; \dot{u})$ may converge to the optimal solution $\dot{u}(\cdot)$ and optimal trajectory $x(\cdot)$ of (38)–(40) as $i \rightarrow \infty$. In (43) we denoted by $x(t; u)$ the solution of (39) at time t corresponding to the given initial state x_0 and the indicated input $u(\cdot)$. ■

Interior Penalty Function Methods

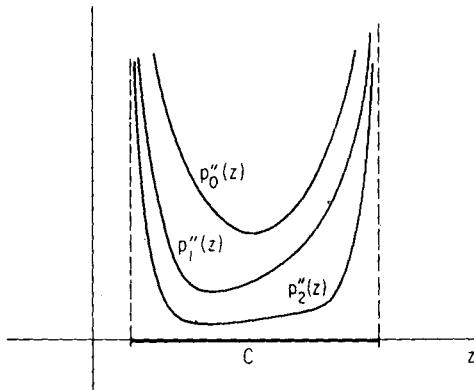
We now proceed to interior penalty function methods for solving the problem (CP) defined in (1). As in the case of exterior penalty function methods, the results we are about to present are trivially extendable to the infinite dimensional problem (CP)_L defined in exercise (35) by replacing \mathbb{R}^n with L in all the statements to follow.

- 44 **Assumption.** We shall assume (i) that the set C in (1) is closed, that it has an interior, and that the closure of the interior of C is equal to C , i.e., that $\bar{C} = C \neq \emptyset$, the empty set; (ii) that (9) is satisfied. (We denote the interior of C by \mathring{C}). ■

The effect of assumption (44) (i) is to rule out constraint sets with “whiskers,” such as the set in \mathbb{R}^n ,

$$45 \quad \{z \mid (\|z - c\|^2 - \|r\|^2)(\langle r, z - c \rangle - \|r\|^2)^2 \leq 0\},$$

which consists of a ball of radius $\|r\|$ and center c , and of a tangent line which passes through the point $c + r$ and is orthogonal to r . The reason for this assumption is that interior penalty function methods construct sequences in the interior of C and hence could never find an optimal point located on a “whisker.”



Interior penalty functions.

- 46 Definition.** Let C be a subset of \mathbb{R}^n which satisfies assumption (44). A sequence of continuous functions $p_i'': \dot{C} \rightarrow \mathbb{R}^1$, $i = 0, 1, 2, \dots$ (where \dot{C} is the interior of C) is said to be a sequence of *interior penalty functions* for the set C if

47 $0 < p_{i+1}''(z) < p_i''(z)$ for all $z \in \dot{C}$ and $i = 0, 1, 2, 3, \dots$,

48 $p_i''(z) \rightarrow 0$ as $i \rightarrow \infty$ for all $z \in \dot{C}$,

49 $p_i(\overset{j}{z}) \rightarrow \infty$ as $j \rightarrow \infty$ for any sequence $\{\overset{j}{z}\} \in \dot{C}$ such that $\overset{j}{z} \rightarrow z^* \in \partial C$, as $j \rightarrow \infty$, and $i = 0, 1, 2, \dots$.

(In (49) ∂C denotes the boundary of C .) ■

Now consider the sequence of problems

50 $(UP)_j^i \quad \min\{f^0(z) + p_j''(z) \mid z \in \dot{C}\}, \quad j = 0, 1, 2, 3, \dots, *$

where the $p_j''(\cdot)$ are interior penalty functions for the set C in (1).

- 51 Lemma.** Consider the problems $(UP)_j^i$ above and suppose that there is a $z'' \in C$ such that the set $\{z \mid f^0(z) \leq f^0(z'') + p_j''(z'')\}$ is compact. Then for $j = 0, 1, 2, \dots$, there exists a $z_j \in \dot{C}$ which minimizes $f^0(z) + p_j''(z)$ over $z \in \dot{C}$.

Proof. Let $z'' \in \dot{C}$ be the point stipulated above, and let

52 $C_j = \{z \in \dot{C} \mid f^0(z) + p_j''(z) \leq f^0(z'') + p_j''(z'')\}, \quad j = 0, 1, 2, \dots$

Then C_j is contained in the compact set $Z' = \{z \mid f^0(z) \leq f^0(z'') + p_0''(z'')\}$. Now let $\{z_i\}_{i=0}^\infty \subset C_j$ be such that $z_i \rightarrow z^*$ as $i \rightarrow \infty$. Then, because of (49) and the continuity of $f^0(\cdot)$ and of $p_j''(\cdot)$, we conclude that $z^* \in C_j$, i.e., that C_j is closed. Hence, C_j is compact and there exists a z_j in C_j which minimizes $f^0(z) + p_j''(z)$ over $z \in C_j$. But for all $z \in \dot{C}$ which are not in C_j , $f^0(z) + p_j''(z) > f^0(z'') + p_j''(z'')$, and hence z_j is optimal for $(UP)_j^i$. ■

- 53 Exercise.** Suppose that we have a point $\overset{0}{z} \in \dot{C}$. In practice, the $p_j''(\cdot)$ are defined for all $z \in \mathbb{R}^n$, $z \notin \partial C$, and to use any one of the algorithms described in Section 2.1 to solve the problems $(UP)_j^i$, $j = 0, 1, 2, \dots$, it is necessary to modify slightly the step size subprocedures so as to ensure that the sequences which they construct remain in \dot{C} .

Devise such modifications for the subprocedures (2.1.14), (2.1.33) and (2.1.36) to be used in the algorithms (2.1.16), (2.1.19) and (2.1.35), respectively. Make sure that your modifications result in algorithms for which the conclusions of theorem (2.1.22) remain valid. ■

* Note that we use the i in $(UP)_j^i$ to indicate “interior,” while $j = 0, 1, 2, 3, \dots$, is the running index.

- 54 **Theorem.** For $j = 0, 1, 2, \dots$, let z_j be optimal for the unconstrained problem $(UP)_j^i$. Then, assuming that (44) is satisfied, every accumulation point \hat{z} of the sequence $\{z_j\}_{j=0}^\infty$ is optimal for the constrained problem (CP).*

Proof. For $j = 0, 1, 2, \dots$, let

$$55 \quad b_j = \min\{f^0(z) + p_j''(z) \mid z \in \hat{C}\},$$

$$56 \quad b = \min\{f^0(z) \mid z \in C\}.$$

Then, because of (47), we have

$$57 \quad b_0 \geq b_1 \geq b_2 \geq \cdots \geq b_j \geq b_{j+1} \geq \cdots \geq b.$$

Since the b_j form a bounded, monotonically decreasing sequence, they must converge to some $b' \geq b$. Suppose that $b' > b$. Let \hat{z} be an optimal point for (CP), then, since $f^0(\cdot)$ is continuous and because of (i) in assumption (44), there must exist an open ball B , with center \hat{z} such that $B \cap \hat{C} \neq \emptyset$, the empty set, and for all $z'' \in B$,

$$58 \quad f^0(z'') < b' - \frac{(b' - b)}{2}.$$

Now take any point z'' in $B \cap \hat{C}$. Then, since by (48) $p_j''(z'') \rightarrow 0$ as $j \rightarrow \infty$, there exists an integer N such that for all $j \geq N$,

$$59 \quad p_j''(z'') < \frac{(b' - b)}{4},$$

and hence for all $j \geq N$,

$$60 \quad b_j \leq f^0(z'') + p_j''(z'') < b' - \frac{(b' - b)}{4},$$

which contradicts the fact that $b_j \rightarrow b'$. Hence we must have $b' = b$.

Now, let z^* be any accumulation point of the sequence $\{z_j\}_{j=0}^\infty$, i.e., $z_j \rightarrow z^*$ as $j \rightarrow \infty, j \in K \subset \{0, 1, 2, 3, \dots\}$. Suppose that z^* is not optimal for (CP). Then we must have $f^0(z^*) > b$, and hence the sequence $\{[f^0(z_j) - b] + p_j''(z_j)\}$, $j \in K$, cannot converge to zero, which contradicts the fact that $(b_j - b) \rightarrow 0$ as $j \rightarrow \infty$. Hence we must have $f^0(z^*) = b$, i.e., z^* is optimal for (CP). ■

- 61 **Exercise.** Suppose that for $i = 1, 2, \dots, m$, $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a continuous function, and consider the set

$$62 \quad C = \{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}.$$

* We call the problems $(UP)_j^i$ unconstrained because they can be solved by simple modifications of the algorithms in Section 2.1; see exercise (53).

Assuming that assumption (44) is satisfied, that for every $z \in \mathring{C}$, $f^i(z) < 0$, $i = 1, 2, \dots, m$, and that α_j , $j = 0, 1, 2, 3, \dots$, is a sequence of strictly decreasing positive numbers which converge to 0 as $j \rightarrow \infty$ ($\alpha_j \downarrow 0$), show that interior penalty functions for the set C can be defined at least in the following two ways:

$$63 \quad p_j''(z) = -\alpha_j \sum_{i=1}^m \frac{1}{f^i(z)}, \quad z \in \mathring{C}, \quad j = 0, 1, 2, \dots,$$

$$64 \quad p_j''(z) = -\alpha_j \sum_{i=1}^m \log[-f^i(z)] \\ = -\alpha_j \log((-1)^m f^1(z) f^2(z) \cdots f^m(z)), \quad z \in \mathring{C}, \quad j = 0, 1, 2, 3, \dots,$$

with (64) defining a penalty function only if $M = \max_{z \in C} (-\min_{z \in C} f^j(z)) < \infty$.*

- 65 **Exercise.** Consider again the optimal control problem (38)–(40). Show that it can be solved by obtaining the limit point $\hat{u}(\cdot)$ of the $\hat{u}(\cdot) \in L_2^\mu[0, T]$, which are optimal for the unconstrained sequence of problems,

$$66 \quad \text{minimize} \quad \int_0^T \|x(t; u) - x'(t)\|^2 dt - \alpha_j \log \left[v - \int_0^T \|u(t)\|^2 dt \right],$$

subject to

$$u(\cdot) \in L_2^\mu[0, T], \quad j = 0, 1, 2, \dots,$$

where the $\alpha_j > 0$ decrease strictly to zero as $j \rightarrow \infty$.

Exterior–Interior Penalty Function Methods

As we shall soon show, under suitable assumptions, exterior and interior penalty function methods can be combined to produce a mixed method for solving the problem (CP) defined in (1).

- 67 **Assumption.** We shall suppose (i) that the C in (1) is of the form $C = C' \cap C''$, where C' is closed and C'' is closed and satisfies (44), i.e., $\mathring{C}'' = C''$; (ii) that (9) is satisfied; (iii) that for at least one $\tilde{z} \in C$ which is optimal for (CP) there exists an open ball \tilde{B} such that $\tilde{z} \in \tilde{B}$ and the set $\tilde{B} \cap C' \cap \mathring{C}''$ is not empty.

Now consider the sequence of unconstrained minimization problems,

$$68 \quad (\text{UP})_i^m \quad \min\{f^0(z) + p_i'(z) + p_i''(z) \mid z \in \mathring{C}''\}, \quad i = 0, 1, 2, 3, \dots,$$

* Strictly speaking, we should set $p_j''(z) = -\alpha_j(\sum_{i=1}^m \log(-f^i(z)/M))$. However, the term $\alpha_j M$ has no effect on the optimal z_j and hence may be omitted.

where the $p_i'(\cdot)$ are exterior penalty functions for the set C' and the $p_i''(\cdot)$ are interior penalty functions for the set C'' , with C', C'' defined as in (67).

- 69 **Exercise.** Assuming that there is a $z'' \in \hat{C}''$ such that the set

$$\{z \mid f^0(z) \leq f^0(z'') + p_0'(z'') + p_0''(z'')\}$$

is compact, show that for $i = 0, 1, 2, \dots$, there is a point $z_i \in \hat{C}''$ which minimizes $f^0(z) + p_i'(z) + p_i''(z)$ over $z \in \hat{C}''$. [Hint: See lemma (51).] ■

- 70 **Theorem.** For $i = 0, 1, 2, \dots$, let z_i be optimal for the unconstrained problem $(UP)_i^m$. Then, assuming that (67) is satisfied, every accumulation point of the sequence $\{z_i\}_{i=0}^\infty$ is optimal for the constrained problem (CP) defined in (1).

Proof. Without loss of generality, we may assume that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$. We shall show that $\hat{z} \in C$. Since C'' is closed by assumption (67), $\hat{z} \in C''$; hence suppose that $\hat{z} \notin C'$. By assumption (67) (iii) there is a point $z'' \in C' \cap \hat{C}''$. For this z'' we must have, by definitions (3) and (46),

$$71 \quad f^0(z'') + p_i'(z'') + p_i''(z'') \rightarrow f^0(z'') \quad \text{as } i \rightarrow \infty.$$

(In fact, $p_i'(z'') = 0$ for $i = 0, 1, 2, \dots$) Let $\delta > 0$ be arbitrary; then there exists an integer $N' \geq 0$ such that

$$72 \quad f^0(z'') + p_i'(z'') + p_i''(z'') < f^0(z'') + \delta \quad \text{for all } i \geq N'.$$

Now, since $\hat{z} \notin C'$, there exists an integer N'' and an open ball B with center \hat{z} , such that

$$73 \quad f^0(z) + p_i'(z) > f^0(z'') + \delta, \quad \text{for all } z \in B, \quad i \geq N''$$

(compare the proof of lemma (18)). Since $z_i \rightarrow \hat{z}$, there is an integer $N''' \geq 0$ such that $z_i \in B$ for all $i \geq N'''$. Let $N = \max\{N', N'', N'''\}$; then

$$74 \quad f^0(z_i) + p_i'(z_i) + p_i''(z_i) > f^0(z'') + \delta > f^0(z'') + p_i'(z'') + p_i''(z'')$$

for all $i \geq N$,

which contradicts the optimality of the z_i , $i \geq N$. Hence we must have $\hat{z} \in C$.

Now suppose that \hat{z} is not optimal for (CP). Then by (67) (iii), there exists a $\tilde{z} \in C$ which is optimal for (CP), with an associated open ball \tilde{B} centered at \tilde{z} , such that

$$75 \quad f^0(z^*) < f^0(\hat{z}) \quad \text{for all } z^* \in \tilde{B},$$

since we must have $f^0(\hat{z}) > f^0(\tilde{z})$, and $f^0(\cdot)$ is continuous. Let z^* be any point in $\tilde{B} \cap C' \cap C''$, which is not empty by assumption (67) (iii). Then

$$76 \quad f^0(z^*) + p_i'(z^*) + p_i''(z^*) = f^0(z^*) + p_i''(z^*) \rightarrow f^0(z^*) < f^0(\hat{z}), \quad \text{as } i \rightarrow \infty,$$

because of (48). Now, since for $i = 0, 1, 2, 3, \dots$,

$$77 \quad f^0(z_i) + p_i'(z_i) + p_i''(z_i) \leq f^0(z^*) + p_i''(z^*) \leq f^0(z^*) + p_0''(z^*),$$

because of (47), and since, in addition, $p_i'(z_i) \geq 0$, $p_i''(z_i) > 0$ and $f^0(z_i)$ is bounded, $i = 0, 1, 2, 3, \dots$, there must exist a subsequence $f^0(z_i) + p_i'(z_i) + p_i''(z_i)$, $i \in K \subset \{0, 1, 2, 3, \dots\}$, such that* (since $f^0(z_i) \rightarrow f^0(\hat{z})$)

$$78 \quad f^0(z_i) + p_i'(z_i) + p_i''(z_i) \rightarrow b \geq f^0(\hat{z}), \quad \text{as } i \rightarrow \infty, \quad i \in K.$$

But then (76) contradicts the optimality of z_i for i sufficiently large, $i \in K$. Hence, \hat{z} must be optimal for (CP). ■

Computational Aspects

The preceding theorems have established the essential property of penalty function methods: that under rather mild assumptions they will construct sequences which converge to points that are optimal for the problem (CP) defined in (1). All of the preceding results implicitly depend upon the gross idealization that we can solve the unconstrained minimization problems $(UP)_i$, $i = 0, 1, 2, \dots$, where we drop the characterizing superscripts e (exterior), i (interior) and m (mixed). Let us continue for a little longer to assume that we can solve the problems $(UP)_i$, and in addition, that this can be done by means of the algorithms presented in the first section of Chapter 2. In this context we shall now comment on two computational aspects. The first is that of how to start and how to end, or rather truncate the process of constructing the “minimizing” sequence of points z_i . The second aspect concerns the relative merits of the various penalty function methods that we have presented.

Suppose that we are given an $\epsilon > 0$ and that we are willing to settle for a point \tilde{z} such that $|f^0(\tilde{z}) - \bar{m}| \leq \epsilon$ and $d(\tilde{z}, C) \leq \epsilon$, where $\bar{m} = \min\{f^0(z) | z \in C\}$, and $d(\cdot, \cdot)$ is a suitably defined distance function. All penalty function methods construct points z_i , which are optimal for $(UP)_i$, such that $f^0(z_i) \rightarrow \bar{m}$ and z_i is either in C or else $z_i \rightarrow \hat{z} \in C$. Hence, in principle, there is an integer j such that z_j satisfies our requirements. Suppose that we know this integer j , which in practice will be fairly large, and suppose that we proceed to minimize $f^0(z) + p_j(z)$, starting from some initial guess $\overset{0}{z}$, by means of any one of the algorithms in Chapter 2 which

* Since $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$ and $f^0(\cdot)$ is continuous, there exists a $\beta < \infty$ such that $|f^0(z_i)| \leq \beta$ for $i = 0, 1, 2, \dots$

require the computation of a gradient. We are very likely to find that $f^0(z)$ appears to be almost zero compared to $p_i(z)$, if $p_i(\cdot)$ is an exterior penalty function, or vice versa, if $p_i(\cdot)$ is an interior penalty function.* Assuming that all calculations are carried out on a finite precision machine, this will result in an extreme loss of accuracy in the computations to follow. However, suppose that we started with a moderate amount of penalty, so that $\partial f^0(z)/\partial z$ and $\partial p_i(z)/\partial z$ are of comparable size. Then the process of calculating z_i , as the limit point of a sequence z_0, z_1, z_2, \dots will proceed with much better efficiency and accuracy. Supposing that $p_i(\cdot) = \alpha_i p(\cdot)$, as in (27), (31), (63) and (64), once we have computed z_i , so that $\partial f^0(z_i)/\partial z + \alpha_i \partial p(z_i)/\partial z = 0$, we can construct $p_{i+1}(\cdot)$ by making α_{i+1} only moderately different from α_i , so that $\partial f^0(z_i)/\partial z$ and $\partial p_{i+1}(z_i)/\partial z$ are not too different in magnitude and hence, after setting $z = z_i$, ensure good numerical behavior in the calculations for z_{i+1} . Thus, the usual practice is to start with a moderate penalty and to let it progress as the calculations proceed. This practice is further indicated by the fact that usually there is no way to establish a priori the specific j stipulated above. However, once part of a minimizing sequence $z_0, z_1, z_2, \dots, z_N$ has been constructed, we can, as is shown in [F2], attempt to find a z which is optimal for (CP) by extrapolation, as follows: Assuming that $p_i(\cdot) = \alpha_i p(\cdot)$, we can obtain a good estimate for z by minimizing $f^0(z(t))$ over $z(t) \in C$, where $z(t)$ is a curve which interpolates the z_i constructed to date and is of the form

$$79 \quad z(t) = z_0 + ta_1 + t^2a_2 + t^3a_3 + \cdots + t^Na_N, \quad t \geq 0,$$

where the vector coefficients $a_j \in \mathbb{R}^n, j = 1, 2, \dots, N$, are determined by solving the system of linear equations

$$80 \quad z(\alpha_i) = z_i, \quad i = 1, 2, \dots, N.$$

The “optimal” t will satisfy $t > \alpha_N$ and most probably will be such that $z(t)$ is on the boundary of C . It can be found by means of one-dimensional search techniques, such as the Golden section search (2.1.14), if applicable. Thus, both from the point of view of starting and of stopping a calculation using penalty functions, there seem to be advantages to constructing a moderate rather than small number of points z_i which are optimal for the unconstrained problems (UP)_{*i*}.

In deciding which of the penalty function methods are most suitable for solving a specific problem, the reader may find himself guided by the following pros and cons: Exterior penalty function methods have the advantage that they can be started at any point z , as an initial guess, and that they can be used with both equality and inequality constraints. In addition, in penalty functions of the form of (27), whenever we find a

* Referring to (27), (31), (63) and (64), we see that the same disparities are likely to occur in the magnitudes of the gradients of these functions.

point \hat{z} in the process of unconstrained minimization such that $f^j(\hat{z}) \leq 0$, the corresponding term disappears from the sum in (27) and its derivatives do not enter into the calculation of the gradient of $f^0(z) + p_j'(z)$ at $z = \hat{z}$. This fact tends to simplify calculations. On the negative side, exterior penalty function methods usually construct points which are not feasible, i.e., which are not in C , and penalty functions such as (27) have few derivatives, a fact which may have an adverse effect on speed of convergence according to Fiacco and McCormack [F2]. Also, should one start calculations at a point \hat{z}^0 which is in the interior of a set such as in (26), the penalty function and all of its derivatives remain zero until \hat{z} crosses the boundary of C , thus providing the computational process with no guidance in the interior of C .

Interior penalty functions, on the other hand, construct sequences of feasible points only, usually have many derivatives, and do provide guidance to the computational process in the interior of C . Their disadvantages are that they can only be used with a restricted class of constraint sets C , those having an interior; that to start them out, one must have a point in the interior of C ; and that in computing, in the case of a C such as in (62), all the terms in a penalty function such as (63) or (64) contribute to the value of the gradient of the penalty function all the time, thus making the evaluation of the gradient of the penalty function more cumbersome than in the case of exterior penalty function methods.

Mixed, or exterior-interior penalty function methods enjoy the best of all possible worlds to some extent. Given an initial guess \hat{z}^0 for the problem (CP) and assuming that $C = \{z \mid f^j(z) \leq 0, j = 1, 2, \dots, m, r^j(z) = 0, j = 1, 2, \dots, l\}$, one can assign exterior penalty functions to the constraints $r^j(z) = 0, j = 1, 2, \dots, l$, and to the constraints $f^j(z) \leq 0$ whenever $f^j(\hat{z}^0) \geq 0$. One can then assign interior penalty functions to those constraints $f^j(z) \leq 0$ for which $f^j(\hat{z}^0) \leq 0$. [Write $C = C_1 \cap C_2 \cap \dots \cap C_m \cap C'_1 \cap C'_2 \cap \dots \cap C'_l$, with $C'_j = \{z \mid r^j(z) = 0\}$, $C_j = \{z \mid f^j(z) \leq 0\}$, and use the results in (32).]

So far, we have discussed penalty function methods as if we could solve the unconstrained minimization problems $(UP)_i$ exactly, and in finite time, on a digital computer. However, as we have seen in the preceding chapters, the various unconstrained minimization methods which are usually available to us compute only stationary points. Furthermore, they usually take an infinite number of iterations to compute these stationary points. Therefore, if we insisted on using penalty function methods in a literal sense, we could not even get past the minimization of $f^0(z) + p_0(z)$ over $z \in \mathbb{R}^n$, in finite time.* Consequently, we must use truncation procedures in approximating

* We denote a penalty function by $p_i(\cdot)$ (without superscripts) whenever it is not necessary to indicate whether it is an exterior or an interior penalty function.

the minimizing, or stationary points of $f^0(\cdot) + p_i(\cdot)$. Also, if we are going to apply to $f^0(\cdot) + p_i(\cdot)$ methods which can only compute points z_i such that $\nabla f^0(z_i) + \nabla p_i(z_i) = 0$, we must become concerned as to the nature of the accumulation points of such a sequence $\{z_i\}$. We shall now propose two truncation procedures, one for exterior and one for interior penalty functions, and we shall establish the properties of these procedures in a few special cases.

Thus consider again the problem (CP) in (1), suppose that $\{z \mid f^0(z) \leq a\}$ is compact for all $a \in \mathbb{R}^1$, and let $(1/\epsilon_i) p'(\cdot)$, $i = 0, 1, 2, \dots$, be a sequence of exterior penalty functions for the set C in (1). We shall suppose that $f^0(\cdot)$ and $p'(\cdot)$ are continuously differentiable functions and that $\epsilon_i = \epsilon/\beta^i$, where $\epsilon > 0$ and $\beta > 1$. We now propose a first-order penalty function type of algorithm for “solving” the problem (CP).

- 81 **Algorithm** (modified exterior penalty function method, Polak [P3]).

Step 0. Select an $\epsilon > 0$, an $\alpha \in (0, \frac{1}{2})$, a $\beta > 1$, a $\rho > 0$, and a $\overset{0}{z} \in \mathbb{R}^n$.

Step 1. Set $\epsilon_0 = \epsilon$, set $j = 0$, and set $i = 0$.

Step 2. Compute

$$82 \quad h(z; \epsilon_i) = - \left[\nabla f^0(z) + \frac{1}{\epsilon_i} \nabla p'(z) \right].$$

Step 3. If $\|h(z; \epsilon_i)\| > \epsilon_i$, go to step 4; else, set $\epsilon_{i+1} = \epsilon_i/\beta$, set $z_i = z$ set $i = i + 1$, and go to step 2.

Step 4. Use algorithm (2.1.33) to compute a λ such that

$$83 \quad \begin{aligned} -\lambda(1 - \alpha) \|h(z; \epsilon_i)\|^2 \\ \leq f^0(z + \lambda h(z; \epsilon_i)) + \frac{1}{\epsilon_i} p'(z + \lambda h(z; \epsilon_i)) - f^0(z) - \frac{1}{\epsilon_i} p'(z) \\ \leq -\lambda \alpha \|h(z; \epsilon_i)\|^2. \end{aligned}$$

Step 5. Set $\overset{j+1}{z} = z + \lambda h(z; \epsilon_i)$, set $j = j + 1$, and go to step 2. ■

- 84 **Lemma.** Algorithm (81) constructs an infinite sequence $\{z_i\}$ and an infinite sequence $\{\epsilon_i\}$ such that $h(z_i; \epsilon_i) \rightarrow 0$ as $i \rightarrow \infty$.

Proof. Since by assumption, the set $\{z \mid f^0(z) \leq a\}$ is compact for all $a \in \mathbb{R}^1$, the set

$$\{z \mid f^0(z) + p'(z)/\epsilon_{i+1} \leq f^0(z_i) + p'(z_i)/\epsilon_{i+1}\} \subset \{z \mid f^0(z) \leq f^0(z_i) + p'(z_i)/\epsilon_{i+1}\}$$

is also compact for $i = -1, 0, 1, 2, 3, \dots$, with $z_{-1} = \overset{0}{z}$. Hence, it follows from theorem (2.1.22) that, starting at $\overset{j}{z} = z_{i-1}$, $i = 0, 1, 2, \dots$, algorithm (81)

constructs a sequence $\bar{z}, \bar{z}^j, \bar{z}^{j+1}, \bar{z}^{j+2}, \dots$ such that the sequence $h(\bar{z}^k, \epsilon_i)$, $k = j, j+1, j+2, \dots$, contains a subsequence which converges to zero. Therefore there must exist an integer $N(i) \geq 0$ such that $\|h(\bar{z}^{j+N(i)}, \epsilon_i)\| \leq \epsilon_i$, and hence $z_i = \bar{z}^{j+N(i)}$, i.e., after a finite number of iterations, z_i and ϵ_{i+1} will be constructed. Therefore the sequences $\{z_i\}$ and $\{\epsilon_i\}$ constructed by algorithm (81) must be infinite, and since in that case $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$, we must have $h(z_i; \epsilon_i) \rightarrow 0$ as $i \rightarrow \infty$. ■

We shall now show that in a number of important cases, algorithm (81) will compute points \hat{z} which satisfy necessary conditions of optimality for the problem (CP).

Case 1. Suppose that $C = \{z \mid r(z) = 0\}$, where $r : \mathbb{R}^n \rightarrow \mathbb{R}^l$ is a continuously differentiable function whose Jacobian matrix $\partial r(z)/\partial z$ has maximum rank for all z (in a sufficiently large open set) in \mathbb{R}^n . Suppose that $p'(z) = \frac{1}{2}\|r(z)\|^2$, and consider the sequence of points \bar{z} constructed by algorithm (81). Within this sequence we have singled out a subsequence of points $\{z_i\}$, constructed in step 3, which satisfy

$$85 \quad h(z_i; \epsilon_i) \leq \epsilon_i, \quad i = 0, 1, 2, \dots$$

Since by assumption the set $\{z \mid f^0(z) \leq a\}$ is compact for all $a \in \mathbb{R}^1$, the sequences $\{z_i\}$ and $\{\epsilon_i\}$ are both infinite by lemma (84), and $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$, because $\epsilon_i = \epsilon/\beta^i$ and $\beta > 1$. Hence,

$$86 \quad h(z_i; \epsilon_i) \rightarrow 0 \quad \text{as } i \rightarrow \infty.$$

Now, with $p(z) = \frac{1}{2}\|r(z)\|^2$,

$$87 \quad h(z_i; \epsilon_i) = - \left[\nabla f^0(z_i) + \frac{1}{\epsilon_i} \left(\frac{\partial r(z_i)}{\partial z} \right)^T r(z_i) \right], \quad i = 0, 1, 2, \dots$$

Now suppose that $z_i \rightarrow z^*$ as $i \rightarrow \infty$;* then, since $\partial r(z)/\partial z$ has maximum rank for all $z \in \mathbb{R}^n$, and $f^0(\cdot)$ and $r(\cdot)$ are both continuously differentiable, we conclude from (86) and (87) that $r(z^*) = 0$, i.e., that $z^* \in C$. Next, since $\partial r(z_i)/\partial z$ has maximum rank,

$$88 \quad \frac{1}{\epsilon_i} r(z_i) = - \left[\left(\frac{\partial r(z_i)}{\partial z} \right) \left(\frac{\partial r(z_i)}{\partial z} \right)^T \right]^{-1} \left(\frac{\partial r(z_i)}{\partial z} \right) [h(z_i; \epsilon_i) + \nabla f^0(z_i)], \\ i = 0, 1, 2, \dots$$

Since $h(z_i; \epsilon_i) \rightarrow 0$ as $i \rightarrow \infty$, and since both $r(\cdot)$ and $f^0(\cdot)$ are continuously differentiable, we conclude that

$$89 \quad \lim_{i \rightarrow \infty} \frac{1}{\epsilon_i} r(z_i) = - \left[\left(\frac{\partial r(z^*)}{\partial z} \right) \left(\frac{\partial r(z^*)}{\partial z} \right)^T \right]^{-1} \left(\frac{\partial r(z^*)}{\partial z} \right) \nabla f^0(z^*) \triangleq \psi.$$

* There is no loss of generality in assuming that the entire sequence $\{z_i\}$ converges to \hat{z} .

Thus, in the limit, we obtain from (87) that

$$90 \quad \nabla f^0(z^*) + \left(\frac{\partial r(z^*)}{\partial z} \right)^T \psi = 0,$$

which we recognize, by comparison with (1.2.2), as being a necessary condition of optimality for the problem under consideration. Therefore, if the sequence $\{z_i\}$ constructed by algorithm (81) for the problem $\min\{f^0(z) | r(z) = 0\}$, under the assumptions stated earlier, remains in a compact set, it will have accumulation points z^* which are in $C = \{z | r(z) = 0\}$ and which satisfy the necessary condition of optimality (1.2.2). Thus, the use of the implementable algorithm (81) depends on a few ifs, and, unlike the purely conceptual case (see (10)), there seem to be no obvious conditions of any generality that ensure that the sequence $\{z_i\}$ will be contained in a compact set. The same seems to be true for any other implementable algorithm based on the use of penalty functions and gradient methods. ■

Case 2. Suppose that $C = \{z | f(z) \leq 0\}$ where $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ is a continuously differentiable function. Let

$$91 \quad p'(z) = \frac{1}{2} \sum_{l=1}^m (\max\{0, f^l(z)\})^2.$$

Setting $J(z) = \{l | f^l(z) \geq 0, l \in \{1, 2, \dots, m\}\}$, we may write

$$92 \quad p'(z) = \frac{1}{2} \sum_{l \in J(z)} [f^l(z)]^2.$$

We shall assume that for all $z \in \mathbb{R}^n$, the vectors $\nabla f^l(z)$, $l \in J(z)$, are linearly independent. Now let us consider the sequence $\{z_i\}$ which is constructed by algorithm (81) for this case. We have

$$93 \quad h(z_i; \epsilon_i) = - \left[\nabla f^0(z_i) + \frac{1}{\epsilon_i} \sum_{l \in J(z_i)} f^l(z_i) \nabla f^l(z_i) \right], \quad i = 0, 1, 2, \dots,$$

and, as before, $h(z_i; \epsilon_i) \rightarrow 0$ as $i \rightarrow \infty$. Suppose that $z_i \rightarrow z^*$ as $i \rightarrow \infty$. Then, since the vectors $\nabla f^l(z_i)$, $l \in J(z_i)$, are linearly independent, and $f^0(\cdot)$ and $f(\cdot)$ are continuously differentiable, it follows from (93) and the fact that $h(z_i; \epsilon_i) \rightarrow 0$ as $i \rightarrow \infty$, that $f(z^*) \leq 0$, i.e., that z^* is feasible. Also, we must have

$$94 \quad \nabla f^0(z^*) + \sum_{l \in J(z^*)} \mu^l \nabla f^l(z^*) = 0,$$

where $\mu^l = \lim_{i \rightarrow \infty} \max\{0, f^l(z_i)/\epsilon_i\}$, $l \in J(z^*)$, exists and satisfies $\mu^l \geq 0$. By inspection of (94), we see that the feasible point $z^* \in C$ satisfies the necessary condition of optimality (1.2.1). ■

Again, the successful use of algorithm (81) depends on the sequence $\{z_i\}$ converging, and again there seem to be no general conditions that guarantee such an outcome. In practice, however, algorithms such as (81) are found to perform well, since it is quite common for the sequence $\{z_i\}$ to converge. (Otherwise, of course, penalty function methods would be next to useless.)

We can easily construct an algorithm of the form of (81) which is based on interior rather than exterior penalty functions, as follows: Suppose that the set C in (1) satisfies assumption (44), and let $\epsilon_i p''(\cdot)$, $i = 0, 1, 2, \dots$, be a sequence of interior penalty functions for the set C . Let us suppose, in addition, that we can find a $\overset{0}{z} \in \overset{0}{C}$ and that the set $\{z \mid f^0(z) \leq f^0(\overset{0}{z}) + \epsilon_0 p''(\overset{0}{z})\}$ is compact. Furthermore, we must assume that the functions $f^0(\cdot)$ and $p''(\cdot)$ are both continuously differentiable.

95 Algorithm (Modified interior penalty function method, Polak [P3]).

Step 0. Select an $\epsilon > 0$, an $\alpha \in (0, \frac{1}{2})$, a $\beta > 1$, a $\rho > 0$, and a $\overset{0}{z} \in \overset{0}{C}$ such that $\{z \mid f^0(z) \leq f^0(\overset{0}{z}) + \epsilon p''(\overset{0}{z})\}$ is compact.

Step 1. Set $\epsilon_0 = \epsilon$, set $j = 0$, and set $i = 0$.

Step 2. Compute

$$96 \quad h(\overset{j}{z}; \epsilon_i) = -[\nabla f^0(\overset{j}{z}) + \epsilon_i \nabla p''(\overset{j}{z})].$$

Step 3. If $\|h(\overset{j}{z}; \epsilon_i)\| > \epsilon_i$, go to step 4; else, set $\epsilon_{i+1} = \epsilon_i/\beta$, set $z_i = \overset{j}{z}$, set $i = i + 1$, and go to step 2.

Step 4. Use algorithm (2.1.33) to compute a $\overset{j}{\lambda}$ such that $\overset{j}{z} + \overset{j}{\lambda} h(\overset{j}{z}; \epsilon_j) \in C$ and

$$97 \quad \begin{aligned} & -\overset{j}{\lambda}(1 - \alpha) \|h(\overset{j}{z}; \epsilon_i)\|^2 \\ & \leq f^0(\overset{j}{z} + \overset{j}{\lambda} h(\overset{j}{z}; \epsilon_i)) + \epsilon_i p''(\overset{j}{z} + \overset{j}{\lambda} h(\overset{j}{z}; \epsilon_i)) - f^0(\overset{j}{z}) - \epsilon_i p''(\overset{j}{z}) \\ & \leq -\overset{j}{\lambda}\alpha \|h(\overset{j}{z}; \epsilon_i)\|^2, \end{aligned}$$

Step 5. Set $\overset{j+1}{z} = \overset{j}{z} + \overset{j}{\lambda} h(\overset{j}{z}; \epsilon_i)$, set $j = j + 1$, and go to step 2. ■

98 Exercise. Show that lemma (84) is also valid for algorithm (95). ■

99 Exercise. Show that the sequence $\{z_i\}$ constructed by algorithm (95) is compact, and hence that it has accumulation points. ■

- 100 **Exercise.** Show that algorithm (95) can be used on the problem, $\min\{f^0(z) | f(z) \leq 0\}$, where $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$ is a continuously differentiable function satisfying not only the assumptions for the applicability of algorithm (95), but also those stated under case 2, a little earlier. Use either the penalty functions defined by (63) or by (64). Show that in this case, the accumulation points of the sequence $\{z_i\}$ are feasible and satisfy the necessary condition of optimality (1.2.1). ■
- 101 **Exercise.** Merge the algorithms (81) and (95) to produce an implementable algorithm based on the use of exterior–interior penalty functions. State assumptions under which this algorithm can be used to compute points satisfying necessary conditions of optimality. ■

Penalty Functions in Optimal Control

First let us consider the discrete optimal control problem (1.1.3), i.e.,

$$102 \quad \text{minimize} \quad \sum_{i=0}^{k-1} f_i^0(x_i, u_i) + \varphi(x_k),$$

subject to

$$103 \quad x_{i+1} - x_i = f_i(x_i, u_i), \quad i = 0, 1, 2, \dots, k-1, \quad x_i \in \mathbb{R}^r, \quad u_i \in \mathbb{R}^\mu,$$

$$104 \quad s_i(u_i) \leq 0, \quad i = 0, 1, 2, \dots, k-1,$$

$$105 \quad g_i(x_i) = 0, \quad q_i(x_i) \leq 0, \quad i = 0, 1, 2, \dots, k.$$

Assume that all the functions in (102)–(105) are continuously differentiable. Let $z = (x_0, u_0, u_1, \dots, u_{k-1}) \in \mathbb{R}^{r+k\mu}$, and let $x_i(z)$ be the solution to (103) at time i , corresponding to the initial state x_0 and the input sequence u_0, u_1, \dots, u_{k-1} . Then, for $i = 0, 1, \dots, k$, the function $x_i(\cdot)$ is continuously differentiable,* and hence so are the functions $\bar{s}_i(\cdot)$, $\bar{g}_i(\cdot)$, and $\bar{q}_i(\cdot)$, defined by

$$106 \quad \bar{s}_i(z) = s_i(u_i),$$

$$107 \quad \bar{g}_i(z) = g_i(x_i(z)),$$

$$108 \quad \bar{q}_i(z) = q_i(x_i(z)).$$

Consequently, the constraint sets,

$$109 \quad \{z | \bar{s}_i(z) \leq 0\}, \quad i = 0, 1, 2, \dots, k-1,$$

$$110 \quad \{z | \bar{g}_i(z) = 0\}, \quad i = 0, 1, 2, \dots, k,$$

$$111 \quad \{z | \bar{q}_i(z) \leq 0\}, \quad i = 0, 1, 2, \dots, k,$$

* See exercise (2.4.8) and note that $\partial x_i(z)/\partial x_0 = \Phi_{i,0}$.

are all closed and can be removed by interior or exterior penalty functions such as in (63), (64), (27), and (31).

Note that with z defined as above, the optimal control problem (102)–(105) assumes the form of a nonlinear programming problem, viz.,

$$112 \quad \min\{f^0(z) \mid \bar{s}_i(z) \leq 0, i = 0, 1, 2, \dots, k-1, \bar{g}_j(z) = 0, \bar{q}_j(z) \leq 0, \\ j = 0, 1, 2, \dots, k\}.$$

As we have already pointed out in exercise (34), one is not compelled to remove all the constraints from (112) by means of penalty functions when one wishes to solve this problem by means of a penalty function method. One only needs to remove those equality or inequality constraints which one finds untractable for one reason or another.

In practice, the sets in (109) usually have an interior, and it is not difficult to find a point in their interior. One would therefore be inclined to use interior penalty functions to remove these constraint sets. However, for the constraint sets in (110) and (111), one would normally have to use exterior penalty functions. Thus, in conjunction with optimal control problems one would often find it convenient to use mixed penalty function methods.

In defining z as we did above, we have eliminated the dynamics equation (103) by solving it. Alternatively, we could have defined $z = (x_0, x_1, x_2, \dots, x_k, u_0, u_1, \dots, u_{k-1})$, which would have resulted in a nonlinear programming problem of the form (112), to which we would have to add the additional constraints $r_i(z) = 0$, where, for $i = 0, 1, \dots, k-1$, $r_i(z) = x_{i+1} - x_i - f_i(x_i, u_i)$. These additional equality constraints could then also be removed by means of penalty functions.

Now let us consider a continuous optimal control problem which is slightly more general than (1.1.8) (t_0, t_f are given):

$$113 \quad \text{minimize} \quad \int_{t_0}^{t_f} f^0(x(t), u(t), t) dt + \varphi(x(t_f)), \quad x(t) \in \mathbb{R}^n, \quad u(t) \in \mathbb{R}^m,$$

subject to

$$114 \quad \frac{d}{dt} x(t) = f(x(t), u(t), t), \quad t \in [t_0, t_f],$$

$$115 \quad s(u(t)) \leq 0, \quad t \in [t_0, t_f],$$

$$116 \quad g(x(t), t) = 0, \quad q(x(t), t) \leq 0, \quad t \in [t_0, t_f],$$

$$117 \quad \int_{t_0}^{t_f} v(u(t)) dt - \hat{v} \leq 0, \quad \int_{t_0}^{t_f} w(x(t)) dt - \hat{w} \leq 0.$$

First let us show how one constructs penalty functions for the constraints in

(115)–(117). Suppose that $s(\cdot)$ is a continuous function from \mathbb{R}^n into $\mathbb{R}^{n'}$. Then, assuming as we did before, that $u(\cdot) \in L_\infty^n[t_0, t_f]$, the set

$$118 \quad \{u(\cdot) \in L_\infty^n[t_0, t_f] \mid s(u(t)) \leq 0, \text{ for almost all } t \in [t_0, t_f]\}$$

is closed and, for example, we can use exterior penalty functions of the form $(1/\epsilon_i) p'(\cdot)$, $i = 0, 1, 2, \dots$, with

$$119 \quad p'(u) = \int_{t_0}^{t_f} \sum_{j=1}^{n'} (\max\{0, s^j(u(t))\})^2 dt,$$

or interior penalty functions of the form $\epsilon_i p''(\cdot)$, $i = 0, 1, 2, \dots$, with

$$120 \quad p''(u) = \int_{t_0}^{t_f} \sum_{j=1}^{n'} \log(-s^j(u(t))) dt,$$

assuming that $s^j(u)$ is bounded from below for all $u \in \mathbb{R}^m$, $j = 1, 2, \dots, n'$.*

Next, let us consider the constraints in (116). Suppose that the function $f(\cdot, \cdot, \cdot)$ in (114) is continuously differentiable in x and in u , and that it is piecewise continuous in t . In addition, suppose that for every $x \in \mathbb{R}^v$ and every $u \in \mathbb{R}^n$, $\partial f(x, u, \cdot)/\partial x$ and $\partial f(x, u, \cdot)/\partial u$ are piecewise continuous on $[t_0, t_f]$. Then the solution $x(t; x_0, u)$ of (114), satisfying $x(t_0; x_0, u) = x_0$, is jointly continuous in x_0 and $u(\cdot)$ for each $t \in [t_0, t_f]$, and assuming that the functions $g(\cdot, \cdot)$ and $q(\cdot, \cdot)$ are both continuous in x , we now find that the sets

$$121 \quad \{z = (x_0, u) \mid x_0 \in \mathbb{R}^v, u \in L_\infty^n[t_0, t_f], g(x(t; x_0, u, t)) = 0, t \in [t_0, t_f]\},$$

$$122 \quad \{z = (x_0, u) \mid x_0 \in \mathbb{R}^v, u \in L_\infty^n[t_0, t_f], q(x(t; x_0, u, t)) \leq 0, t \in [t_0, t_f]\}$$

are both closed. Now, suppose that $g : \mathbb{R}^v \times \mathbb{R}^1 \rightarrow \mathbb{R}^{n'}$, and that for every $i \in J \subset \{1, 2, \dots, v'\}$, the function $g^i(\cdot, \cdot)$ is continuous in t , and that for all $i \notin J$, $g^i(x, t) = 0$ for all $x \in \mathbb{R}^v$ and all $t \notin \{t_{i1}, t_{i2}, \dots, t_{ik_i}\} \subset [t_0, t_f]$, i.e., for $i \notin J$, $g^i(x, t)$ is zero for all but a finite number of values of t in $[t_0, t_f]$. Then, if we define $p' : \mathbb{R}^v \times L_\infty^n[t_0, t_f] \rightarrow \mathbb{R}^1$ by

$$123 \quad p'(x_0, u) = \int_{t_0}^{t_f} \sum_{i \in J} [g^i(x(t; x_0, u), t)]^2 dt + \sum_{i \notin J} \sum_{j=1}^{k_i} [g^i(x(t_j; x_0, u), t_j)]^2,$$

we find that $(1/\epsilon_i) p'(\cdot, \cdot)$, $i = 0, 1, 2, 3, \dots$, is a sequence of penalty functions for the set in (121), assuming, of course, that $\epsilon_0 > \epsilon_1 > \epsilon_2, \dots$ and that $\epsilon_1 \rightarrow 0$ as $i \rightarrow \infty$.

124 **Exercise.** Making the same assumptions on $q(\cdot, \cdot)$ as we made above on

* See the footnote on p. 136.

$g(\cdot, \cdot)$, construct both interior and exterior penalty functions for the set in (122). ■

The constraints in (117) are the easiest of the lot to remove by penalty functions. Suppose that the functions $v(\cdot)$ and $w(\cdot)$ are continuous. Then, for example, under a boundedness assumption (see footnote to exercise (61)),

$$125 \quad -\frac{1}{i} \log \left[\hat{v} - \int_{t_0}^{t_f} v(u(t)) dt \right], \quad -\frac{1}{i} \log \left[\hat{w} - \int_{t_0}^{t_f} w(x(t; x_0, u)) dt \right], \\ i = 1, 2, 3, \dots,$$

can be seen to be sequences of interior penalty functions for removing these constraints, while

$$126 \quad i \left(\max \left\{ 0, \left[\int_{t_0}^{t_f} v(u(t)) dt - \hat{v} \right] \right\} \right)^2, \quad i \left(\max \left\{ 0, \left[\int_{t_0}^{t_f} w(x(t; x_0, u)) dt - \hat{w} \right] \right\} \right)^2, \\ i = 1, 2, 3, \dots,$$

are exterior penalty functions for the constraints in (117).

As we are well aware by now, when a penalty function method is applied to a constrained minimization problem, it constructs a sequence of points z_i , $i = 0, 1, 2, \dots$, which are optimal for the corresponding problems in an associated sequence of unconstrained (or simpler) minimization problems. In the finite dimensional case, i.e., for nonlinear programming or discrete optimal control problems, such a sequence $\{z_i\}$ is usually contained in a closed and bounded set, which is compact because the problems are finite dimensional. Consequently, in the finite dimensional case, the sequences $\{z_i\}$ are often found to converge.

In the infinite dimensional case, i.e., for continuous optimal control problems, the sequences of points $\{z_i\}$, $z_i = (x_{0i}, u_i(\cdot))$, where x_{0i} is an initial state and $u_i(\cdot)$ is an input for (114), will also frequently be contained in a closed and bounded set. However, since this set is a subset of an infinite dimensional space, such as $\mathbb{R}^p \times L_\infty^\mu[t_0, t_f]$, it will usually not be compact. As a result, sequences $\{z_i\}$ constructed by a penalty function method for a continuous optimal control problem very frequently do not converge and do not contain convergent subsequences. Hence, the question arises as to exactly what can be accomplished by using penalty function methods on a continuous optimal control problem. Some answers to this question were obtained by Russell [R2], and by Cullum [C6]. The nature of their results is best illustrated by considering a special case of the problem defined by (113)–(116) (without the constraints in (117)). Suppose that q in (116) does not depend on time, i.e., $q(x, t) = q(x)$, and that it is continuous in x ; and suppose that $g(x, t_0) = x - x_0$ and that $g(x, t) = 0$ for all $t \in (t_0, t_f]$, i.e., we are simply given an initial state for (114). Furthermore,

suppose that the sets $\{u \mid s(u) \leq 0\}$ and $\{x \mid q(x) \leq 0\}$ have an interior. To solve this problem, we could propose to use the following sequence of simpler constrained problems:

$$127 \quad (\text{UP})_i \quad \text{minimize} \quad \int_{t_0}^{t_f} [f^0(x(t), u(t), t) + i \sum_{j=1}^m (\max\{0, q^j(x(t; x_0, u))\})^2] dt \\ + \varphi(x(t_f)), \quad i = 0, 1, 2, \dots,$$

subject to (114), with $x(t_0) = x_0$, and (115). Suppose that $\hat{u}(\cdot)$ is an optimal control for (113)–(116) in the specific form under consideration, and that $u_i(\cdot)$ is optimal for $(\text{UP})_i$. Let \bar{m} be the value of the integral in (113) resulting from the use of the control $\hat{u}(\cdot)$, and let \bar{m}_i be the value of the integral in (127) resulting from the use of $u_i(\cdot)$. Then, since $(\text{UP})_i$ uses exterior penalty functions, as we have already shown in (11), we must have

$$128 \quad \bar{m}_0 \leq \bar{m}_1 \leq \dots \leq \bar{m}.$$

Hence, there must exist an $\bar{m}^* \leq \bar{m}$ such that $\bar{m}_i \rightarrow \bar{m}^*$. It has been shown by Cullum [C6] that usually we can expect to have $\bar{m}^* < \bar{m}$, and that \bar{m}^* will be the minimum value of the *relaxed optimal control problem* which consists of (113), (115) and (116), in the specific form under consideration, and in which (114) is replaced by the relaxed dynamic equation,

$$129 \quad \frac{d}{dt} x(t) \in [\bar{f}(x(t), U, t)], \quad t \in [t_0, t_f], \quad x(t_0) = x_0,$$

where $U = \{u \in \mathbb{R}^\mu \mid s(u) \leq 0\}$, $[\]$ denotes the convex hull of the points enclosed, and the bar denotes closure of the set.*

When the function $f(\cdot, \cdot, \cdot)$ is linear in u and the set U is convex, any pair $u(\cdot), x(\cdot)$ which satisfies (129) also satisfies (114) (the converse is obviously always true), and in that case we have $\bar{m}^* = \bar{m}$. It has been shown by Russell [R2] that for this case, under rather mild assumptions on $q(\cdot)$, we can also expect the $u_i(\cdot)$ to converge to $\hat{u}(\cdot)$ in the weak L_2 topology, and that the corresponding trajectories $x_i(t) = x(t; x_0, u_i)$, defined by (114), satisfy $\max\|x_i(t) - x(t; x_0, \hat{u})\|^2 \rightarrow 0$ as $i \rightarrow \infty$, with the max being over $t \in [t_0, t_f]$.

Before leaving the subject of penalty functions in continuous optimal control, we should point out yet another interesting use of penalty function methods which is presently being popularized by a group at the University of California, Los Angeles, headed by Balakrishnan (see [B1]) and by a group at the Institute of Automatics and Telemechanics in Moscow, headed by Moiseev. To illustrate their approach, consider the unconstrained optimal control problems discussed in Section 2.5., i.e., (2.5.1), (2.5.2).

* Obviously, in this case $\{u_i(\cdot)\}$ cannot have a subsequence which converges to $\hat{u}(\cdot)$, and its behavior is too complex for a description within our informal framework.

We recall that in the gradient methods which we described for the solution of these problems, various differential equations had to be solved at each iteration. This is a very time-consuming task. In order to avoid this difficulty, Balakrishnan [B1] suggests that we solve instead the following sequence of unconstrained, *finite dimensional* problems:

$$130 \quad (\text{UP})_i \quad \text{minimize} \quad \int_{t_0}^{t_1} \left[f^0(x(t), u(t), t) + \frac{1}{\epsilon_i} \left\| \frac{d}{dt} x(t) - f(x(t), u(t), t) \right\|^2 \right] dt + \varphi(x(t_1)),$$

(where $\epsilon_i > 0$ converges strictly monotonically to zero) under the assumption that $x(t) = \sum_{i=0}^{l'} a_i \cos 2i\pi t / (t_0 - t_1)$, $u(t) = \sum_{i=0}^{l''} b_i \cos 2i\pi t / (t_1 - t_0)$, where the vectors a_i , b_i are to be determined.

Obviously, the quality of the answer one can expect to obtain by such a method depends on the magnitude of the integers l' , l'' . If we make l' , l'' large so as to enhance our chances of getting a very good approximation to the optimal $\hat{u}(\cdot)$ and $\hat{x}(\cdot)$ for the problem (2.5.1), (2.5.2), then we have to solve problems of large dimensions $(\text{UP})_i$, which may turn out to be quite difficult. Hence, it would seem that the best use of such methods would be in conjunction with small integers l' , l'' , for the purpose of obtaining a good initial guess for another method, such as any one of the ones discussed in Section 2.5.

This concludes our discussion of penalty function methods.

4.2 Methods of Centers

In this section we shall present the first of several closely related methods for solving the problem

$$1 \quad \min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

where $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, 2, \dots, m$, are continuously differentiable functions.

The methods we are concerned with are referred to as methods of centers, for a reason which will shortly become obvious. They were introduced by Huard [H3], [B7]* and provide a transition between the penalty function methods described in the preceding section and the methods of feasible directions that we shall describe in the sections to follow. In retrospect,

* Huard first described a method of centers in 1963 and 1964 in internal memos of the Electricité de France.

the reader will find that they can be thought of either as parameter-free barrier methods, related to interior penalty function methods, or else as parameter-free feasible directions methods.

We now make two assumptions on the set

$$2 \quad C = \{z \mid f^i(z) \leq 0, i = 0, 1, 2, \dots, m\},$$

and on the cost function $f^0(\cdot)$. The first of these assumptions is essential, since without it the method about to be presented simply cannot be applied to problem (1). The effect of the second assumption is to rule out a number of problems to which these algorithms *can* be applied, but which may have solutions that these algorithms can never find. These observations will become obvious as we progress.

- 3 **Assumption.** We shall assume that there is a point $z_0 \in C$ such that the set $C'(z_0)$, defined by

$$4 \quad C'(z_0) = \{z \mid f^0(z) - f^0(z_0) \leq 0, f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

is compact and has an interior. ■

- 5 **Assumption.** We shall assume (i) that the set C in (2) has an interior and that the closure of the interior of C is equal to C , i.e., that $\bar{C} = C \neq \emptyset$; (ii) that for every $z \in \bar{C}$, $f^i(z) < 0$, $i = 1, 2, \dots, m$ (compare (1.44)). ■

The reason for (5)(i) is that given a point $z_0 \in C$, the methods of centers pick as its successor a point z_1 in the interior of the set $C'(z_0)$, and hence they can find an optimal point for (1) only if that point is in the closure of the interior of C . Note that we have already encountered such a situation in the case of interior penalty function methods. The reason for (5)(ii) will become clear in (8).

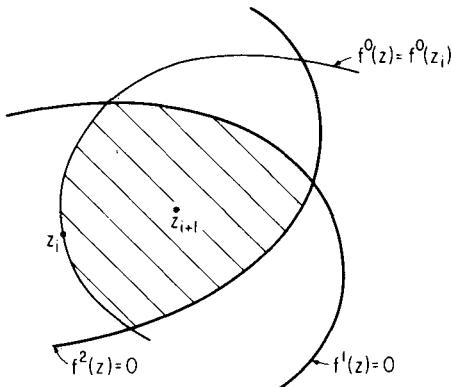
- 6 **Exercise.** Construct a set of the form (2) which does not satisfy the assumption (5). [Hint: See (1.45).] Note that one is hardly likely to encounter such a set in a practical situation. ■

We now present the simplest of the methods of centers, which takes for a successor to the given point z_0 a point z_1 in the “center” of the set $C'(z_0)$, the center of $C'(z_0)$ consisting of points which maximize a suitable distance from the boundaries of the set $C'(z_0)$. Let $d : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^1$ be defined by

$$7 \quad d(z', z) = \max\{f^0(z') - f^0(z), f^i(z'), i = 1, 2, \dots, m\}.$$

Then $-d(\cdot, \cdot)$ can be taken to be such a distance function. Many other choices are possible, however, and the interested reader should consult [B7], [H3] and [T2].

- 8 Exercise.** Suppose that $z_0 \in C$ is not optimal. Show that under assumption (5) the set $C'(z_0)$, in (4), has an interior and that for every point z' in the interior of $C'(z_0)$, $d(z', z_0) < 0$ ($d(z, z_0) = 0$ for every z on the boundary of $C'(z_0)$ by inspection). ■
- 9 Exercise.** Show that, under assumption (3), problem (1) must have a solution. ■



Method of centers.

- 10 Algorithm** (method of centers, Huard [H3]).

Step 0. Select a point $z_0 \in C$ satisfying (3) and set $i = 0$.

Step 1. Compute a point $z' \in \mathbb{R}^n$ such that

$$11 \quad d(z', z_i) = \min\{d(z, z_i) \mid z \in \mathbb{R}^n\}.$$

Comment. If z_0 satisfies (3), then such a z' must exist. By inspection, $z' \in C'(z_i)$.

Step 2. If $d(z', z_i) = 0$, set $z_{i+1} = z_i$ and stop; else, go to step 3.

Step 3. Set $z_{i+1} = z'$, set $i = i + 1$, and go to step 1. ■

Thus, just like the penalty function methods, algorithm (10) decomposes problem (1) into a sequence of unconstrained minimization problems (11). In the form stated, this is, obviously, a conceptual algorithm, since the minimization in (11) cannot be expected to be accomplishable in a finite number of digital computer operations.

- 12 Theorem.** Suppose that assumptions (3) and (5) are satisfied by problem (1). If the sequence $\{z_i\}$ constructed by algorithm (10) is infinite, then it has accumulation points, all of which are optimal for (1). (When $\{z_i\}$ is finite, its last element is optimal, by inspection.)

Proof. By inspection, we must have $C'(z_{i+1}) \subset C'(z_i)$, $i = 0, 1, 2, \dots$, where $C'(z_i)$ is defined as in (4), with z_i taking the place of z_0 . Since $C'(z_0)$ is compact by construction (which, in turn is possible because of (3)), and since $z_{i+1} \in C'(z_i) \subset C'(z_0)$, the sequence $\{z_i\}$ must contain accumulation points. To show that the accumulation points of $\{z_i\}$ are optimal for (1), we make use of theorem (1.3.10). For this purpose, we define $T = C$, $c(\cdot) = f^0(\cdot)$, and we define $\hat{z} \in C$ to be desirable if it is optimal for (1). We define the map $A(\cdot)$ by the instructions in algorithm (10), i.e., to compute a $z'' \in A(z)$, we set $z_i = z$ in step 1 of (10), then we compute z_{i+1} in step 2 or step 3, as appropriate, and then set $z'' = z_{i+1}$. Since (i) of (1.3.10) is obviously satisfied by $f^0(\cdot)$, we only need to prove that assumption (ii) of (1.3.10) is satisfied by the maps $A(\cdot)$ and $c(\cdot)$ that we have just defined.

Hence, let us show that (ii) of (1.3.10) is satisfied. Suppose that $z_i \in C$ is not optimal. Then, because of assumption (5), and because the set $A(z_i)$ is compact* and $d(\cdot, z_i)$ is continuous (by (B.3.13)),

$$13 \quad \max_{z_{i+1} \in A(z_i)} d(z_{i+1}, z_i) = 2\delta < 0.^\dagger$$

Next, since $A(z_i)$ is compact, it follows from (B.3.20) that $\max_{z' \in A(z_i)} d(z', z)$ is continuous in z . Hence there exists an $\epsilon > 0$ such that

$$14 \quad \max_{z' \in A(z_i)} d(z', z) \leq \delta, \quad \text{for all } z \in B(z_i, \epsilon) = \{z \in C \mid \|z - z_i\| \leq \epsilon\}.$$

But any $z'' \in A(z)$ is a point which minimizes $d(z', z)$ over $z' \in \mathbb{R}^n$, and hence we must have

$$15 \quad c(z'') - c(z) = f^0(z'') - f^0(z) \leq d(z'', z) \leq \max_{z' \in A(z_i)} d(z', z) \leq \delta,$$

for all $z \in B(z_i, \epsilon)$, for all $z'' \in A(z)$. Hence we see that (ii) of (1.3.10) is satisfied. It therefore follows that theorem (12) is true. ■

In practice, approximate versions of algorithm (10) have been found to be rather inefficient, because even an approximate calculation of a z' defined by (11) is quite time-consuming. This has led to the development of methods which truncate the search for a point z' satisfying (11) when one finds a point z'' which is an " ϵ center," with ϵ being made progressively smaller. We shall now present a method which truncates the search for a center z' of the set $C'(z_i)$ after a single iteration of the search subprocedure. In the form stated, this method is also not implementable, and we shall have to take up the question as to how it can be made implementable a little later.

* For any $z \in C$, the set $A(z)$ is compact because (as a result of (3)) the set $C'(z)$ is compact and $d(\cdot, z)$ is continuous.

† Note that for all $z_{i+1} \in A(z_i)$, $d(z_{i+1}, z_i) = 2\delta$.

We shall need the necessary condition of optimality and the definitions given below in our discussion of Huard's most efficient version of the method of centers [H3].

- 16 **Exercise.** Suppose that \hat{z} is optimal for problem (1). Show that the following must then be true:

$$17 \quad \min_{h \in S} (\max\{\langle \nabla f^0(\hat{z}), h \rangle; f^i(\hat{z}) + \langle \nabla f^i(\hat{z}), h \rangle, i = 1, 2, \dots, m\}) = 0,$$

where S is any subset of \mathbb{R}^n containing the origin in its interior. [Hint: Show that (17) holds if and only if (1.2.9) holds.] ■

Now let $h^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ be defined by

$$18 \quad h^0(z) = \min_{h \in S^*} (\max\{\langle \nabla f^0(z), h \rangle; f^i(z) + \langle \nabla f^i(z), h \rangle, i = 1, 2, \dots, m\}),$$

where

$$19 \quad S^* = \{h \in \mathbb{R}^n \mid |h^i| \leq 1, i = 1, 2, \dots, n\}.$$

Since the set S^* contains the origin in its interior, we see that if \hat{z} is optimal for problem (1), then $h^0(\hat{z}) = 0$. Note that it follows from (1.2.13) that whenever the set C in (2) does not have an interior, $h^0(z) = 0$ for all $z \in C$, and in that case, the condition (17) is trivial.

- 20 **Exercise.** Suppose that $h(z) \in S^*$ is such that

$$21 \quad h^0(z) = \max\{\langle \nabla f^0(z), h(z) \rangle; f^i(z) + \langle \nabla f^i(z), h(z) \rangle, i = 1, 2, \dots, m\},$$

and let $\bar{h} = (h^0, h) \in \mathbb{R}^{n+1}$. (i) Show that $\bar{h}(z) = (h^0(z), h(z))$ is optimal for the linear programming problem,*

$$22 \quad \text{minimize} \quad h^0,$$

subject to

$$23 \quad -h^0 + \langle \nabla f^0(z), h \rangle \leq 0,$$

$$24 \quad -h^0 + f^i(z) + \langle \nabla f^i(z), h \rangle \leq 0, \quad i = 1, 2, \dots, m,$$

$$25 \quad |h^i| \leq 1, \quad i = 1, 2, 3, \dots, n.$$

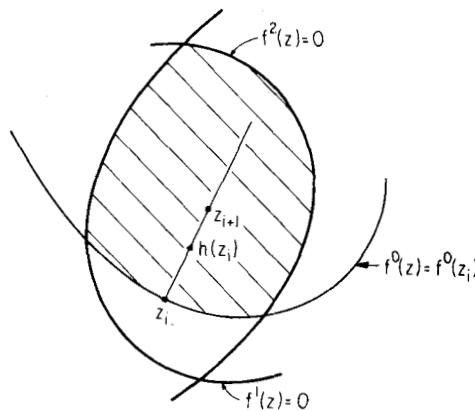
(ii) Show that any $\bar{h}(z) = (h^0(z), h(z))$ which is optimal for (22)–(25) satisfies (21). ■

- 26 **Remark.** Note that since by assumption the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$ are continuously differentiable, it follows from (B.3.13) that $h^0(\cdot)$ is a con-

* When m is large, it may be advantageous to use decomposition techniques in solving (22)–(25); see [V2].

tinous function. Also note that for all $z \in C$, $h^0(z) \leq 0$. (Set $h = 0$ in (18) to show that this is true.) ■

We can now state the most efficient version of the method of centers available to date, for solving problem (1) under assumptions (3) and (5) (or, to be more precise, for finding points $\hat{z} \in C$ such that $h^0(z) = 0$).



Modified method of centers.

27 Algorithm (modified method of centers, Huard [H3]).

Step 0. Compute a point $z_0 \in C$ (defined in (2)), satisfying (3), and set $i = 0$.

Step 1. Set $z = z_i$.

Step 2. Solve (22)–(25) to obtain $(h^0(z), h(z))$.

Step 3. If $h^0(z) = 0$, set $z_{i+1} = z_i$ and stop; else, go to step 4.

Step 4. Compute $\mu(z)$ to be the smallest positive scalar such that

$$28 \quad d(z + \mu(z) h(z), z) = \min\{d(z + \mu h(z), z) \mid \mu \geq 0\},$$

where $d(\cdot, \cdot)$ is defined as in (7).

Comment. It can be seen that $\mu(z)$ exists because of (3).

Step 5. Set $z_{i+1} = z + \mu(z) h(z)$, set $i = i + 1$, and go to step 1. ■

- 29 Remark.** The vector $h(z)$ computed in step 2 of algorithm (27) may not be uniquely defined as a solution of (22)–(25), since the solution of a linear programming problem may not be unique. We accept as $h(z)$ any vector which is optimal for (22)–(25) that has been computed by the simplex algorithm, or any other algorithm for solving (22)–(25). Note that the only

operation in algorithm (27) which requires modification for implementation purposes is the minimization defined by (28). ■

- 30 **Remark.** The computation of a $z_0 \in C$ which must be carried out in order to initialize algorithm (27) in step 0, can be performed by applying algorithm (27) to the problem (in \mathbb{R}^{n+1}),

$$31 \quad \min\{z^0 \mid -z^0 + f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

for which we construct an initial feasible solution $\bar{z}_0 = (z_0^0, z_0) \in \mathbb{R}^{n+1}$ by taking $z_0 \in \mathbb{R}^n$ arbitrary and by setting $z_0^0 = \max\{f^i(z_0), i \in \{1, 2, \dots, m\}\}$. Since (3) and (5) are satisfied, there will be a finite integer* k such that $\bar{z}_k = (z_k^0, z_k)$ will satisfy $z_k^0 < 0$ and $z_k \in C$. When this occurs, set $z_0 = z_k$ and proceed with the solution of (1) by means of (27). ■

- 32 **Theorem.** Suppose that (3) is satisfied. Let z_0, z_1, z_2, \dots be a sequence generated by algorithm (27), with z_0 as in (3). Then the sequence $\{z_i\}$ is finite, ending at $z_{k+1} = z_k$, and $h^0(z_k) = 0$, or it is infinite and every accumulation point z^* of $\{z_i\}$ satisfies $h^0(z^*) = 0$. (See also (1.3.65).)[†]

Proof. Since construction of new points stops in step 3 of algorithm (27) only when $h^0(z) = 0$, the case of $\{z_i\}$ finite is trivial. Hence, let us consider the case when the sequence $\{z_i\}$ is infinite. Since $h(z)$ as defined by (22)–(25) is not unique, we shall make use of theorem (1.3.10) (algorithm (27) is of the form of the model (1.3.9)).

Thus, let us set $T = C$, $c(\cdot) = f^0(\cdot)$; let us define $A(\cdot)$ (mapping T into the set of all the subsets of T) by the instructions[‡] in algorithm (27), and let us define $z \in C$ to be desirable if $h^0(z) = 0$. To show that every accumulation point \hat{z} of the sequence $\{z_i\}$ constructed by algorithm (27) satisfies $h^0(\hat{z}) = 0$ (i.e., that \hat{z} is desirable), when $\{z_i\}$ is infinite, we only need to show that (i) and (ii) of (1.3.10) are satisfied by the maps $c(\cdot)$ and $A(\cdot)$ under consideration. Clearly, (i) of (1.3.10) is satisfied, since $f^0(\cdot)$ is continuously differentiable by assumption. Hence, we only need to establish (ii), i.e., that for any $z^* \in C$ such that $h^0(z^*) < 0$, there exist an $\epsilon^* > 0$ and a $\delta^* < 0$ such that for all $z \in B(z^*, \epsilon^*) = \{z \in C \mid \|z - z^*\| \leq \epsilon^*\}$,

$$33 \quad f^0(z') - f^0(z) \leq \delta^* \quad \text{for all } z' \in A(z), \quad \text{for all } z \in B(z^*, \epsilon^*).$$

* Assuming, of course, that the sequence $\{\bar{z}_i\}$ being constructed is approaching a solution.

[†] Note that this theorem does not depend on assumption (5) being satisfied. However, when (5) is not satisfied, algorithm (27) may stop at z_0 ; i.e., it may be useless.

[‡] To compute a vector $z'' \in A(z')$, $z' \in C$, set $z_i = z'$ in step 1 of (27) and compute z_{i+1} in step 3 or in step 5, as may be appropriate. Then set $z'' = z_{i+1}$. Thus the set $A(z')$ is determined by the set of vectors $h(z') \in S^*$ which satisfy (21) for $z = z'$, together with the step size rule (28), when $h^0(z') < 0$, and $A(z') = \{z'\}$ when $h^0(z') = 0$.

(Recall that for any $z \in C$, $h^0(z) \leq 0$ (see remark (26).) Thus, suppose that $z^* \in C$ and that $h^0(z^*) = v^* < 0$. Since $h^0(\cdot)$ is continuous, there exists an $\epsilon > 0$ such that

$$34 \quad h^0(z) \leq \frac{v^*}{2} \quad \text{for all } z \in B(z^*, \epsilon).$$

Since S^* is compact, there exist, by theorem (B.3.7), an $\epsilon' \in (0, \epsilon]$ and a $\lambda' > 0$ such that

$$35 \quad |\langle \nabla f^i(z + \lambda h), h \rangle - \langle \nabla f^i(z), h \rangle| \leq \frac{-v^*}{8},$$

for all $z \in B(z^*, \epsilon')$, for all $h \in S^*$, for all $\lambda \in [0, \lambda']$, $i = 0, 1, \dots, m$.

Again, since S^* is compact, by theorem (B.3.1), there exist an $\epsilon^* \in (0, \epsilon']$ and a $\lambda^* \in (0, \lambda']$ such that

$$36 \quad |f^i(z + \lambda h) - f^i(z)| \leq \frac{-v^*}{8},$$

for all $z \in B(z^*, \epsilon^*)$, for all $h \in S^*$, for all $\lambda \in [0, \lambda^*]$, $i = 1, 2, \dots, m$.

Now let $z \in B(z^*, \epsilon^*)$ be arbitrary, and let $h(z)$ be any vector in S^* satisfying (21) for this z . Then, because of (34) and (21), we must have

$$37 \quad \langle \nabla f^0(z), h(z) \rangle \leq h^0(z) \leq \frac{v^*}{2},$$

$$38 \quad f^i(z) + \langle \nabla f^i(z), h(z) \rangle \leq h^0(z) \leq \frac{v^*}{2}, \quad i = 1, 2, \dots, m.$$

By the mean-value theorem (B.1.1),

$$39 \quad f^i(z + \lambda^* h(z)) - f^i(z) = \lambda^* \langle \nabla f^i(z + \xi^* h(z)), h(z) \rangle, \quad i = 0, 1, 2, \dots, m,$$

where $\xi^* \in [0, \lambda^*]$. Consequently, because of (39), (37) and (35),

$$40 \quad f^0(z + \lambda^* h(z)) - f^0(z) \leq \frac{3\lambda^* v^*}{8} < \frac{\lambda^* v^*}{8}.$$

Now, for $i \in \{1, 2, \dots, m\}$, either $f^i(z) > v^*/4$ or else $f^i(z) \leq v^*/4$. Suppose that for some $i \in \{1, 2, \dots, m\}$, $f^i(z) > v^*/4$; then from (38),

$$41 \quad \langle \nabla f^i(z), h(z) \rangle \leq \frac{v^*}{2} - f^i(z) < \frac{v^*}{2} - \frac{v^*}{4} = \frac{v^*}{4}.$$

Combining (39), (41) and (35), we obtain

$$42 \quad f^i(z + \lambda^* h(z)) - f^i(z) \leq \frac{\lambda^* v^*}{8}.$$

Since both $f^i(z) \leq 0$ and $v^* < 0$, we find that

$$43 \quad f^i(z + \lambda^* h(z)) \leq f^i(z) + \frac{\lambda^* v^*}{8} < 0.$$

Next, suppose that for some $i \in \{1, 2, \dots, m\}$, $f^i(z) \leq v^*/4$. Then, from (36),

$$44 \quad f^i(z + \lambda^* h(z)) \leq \frac{-v^*}{8} + \frac{v^*}{4} = \frac{v^*}{8} < 0.$$

Finally, let $z' = z + \mu(z) h(z)$, with $\mu(z)$ determined as in (28); then we must have

$$\begin{aligned} 45 \quad f^0(z') - f^0(z) &\leq d(z', z) \\ &\leq \max\{f^0(z + \lambda^* h(z)) - f^0(z); f^i(z + \lambda^* h(z)), i = 1, 2, \dots, m\} \\ &\leq \max\left\{\frac{v^*}{8}, \frac{\lambda^* v^*}{8}\right\} \triangleq \delta^* < 0. \end{aligned}$$

Since (45) is true for all $z \in B(z^*, \epsilon^*)$ and for all $z' \in A(z)$, our proof is completed. ■

46 Remark. One can construct a large family of methods of centers of the form of algorithm (27) by changing the two obvious parameters in the method. The first parameter is the set S^* in (18). We could have just as easily chosen $S^* = \{h \in \mathbb{R}^n \mid \|h\| \leq 1\}$, for example. However, the choice in (19) leads to a simpler subproblem which must be solved at each iteration to compute an $h(z)$ in step 2 of algorithm (27). The second parameter is the “distance” function $d(\cdot, \cdot)$, which can also be defined in other, though not necessarily more computationally favorable, ways than in (7) (see [B7] for a general characterization of functions which can be used as “distance” functions in methods of centers). ■

Let us now examine briefly how one can replace the conceptual operation (28) with one that can be implemented on a digital computer. The simplest case is when the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, are all convex. In that case it is easy to see that the function $d(\cdot, z)$ is also convex for any $z \in C$.

47 Algorithm (implementation of modified method of centers, Polak [P3]).

Step 0. Select an $\tilde{\epsilon}_0 > 0$, a $\rho \geq 1$, and an $\eta > 1$; compute a $z_0 \in C$ (defined in (2)) and set $i = 0$.

Comment. It is common to set $\eta = 2$.

Step 1. Set $z = z_i$.

Step 2. Solve (22)–(25) to obtain $(h^0(z), h(z))$.

Step 3. If $h^0(z) = 0$, set $z_{i+1} = z_i$ and stop; else, go to step 4.

Comment. Steps 4–7, below, replace step 4 in (27) for the computation of the step length $\mu(z)$.

Step 4. For $\lambda \geq 0$, set

$$48 \quad \theta(\lambda) = d(z + \lambda h(z), z).$$

Step 5. Set $\epsilon_0 = \bar{\epsilon}_0$ and set $j = 0$.

Step 6. Use the Golden section search algorithm (2.1.14), with $\theta(\cdot)$ defined as in (48), with ρ as chosen in step 0, and with $\epsilon = \epsilon_j$, to compute $\bar{\mu}$.

Step 7. If $d(z + \bar{\mu}h(z), z) \leq -\epsilon_j$, set $\mu(z) = \bar{\mu}$, and go to step 8; else, set $\epsilon_{j+1} = \epsilon_j/\eta$, set $j = j + 1$, and go to step 6.

Step 8. Set $z_{i+1} = z + \mu(z)h(z)$, set $i = i + 1$, and go to step 1. ■

- 49 **Exercise.** Suppose that the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, are convex. Show that algorithm (47) corresponds to the model (1.3.26), and that the map $A : \mathbb{R}^+ \times C \rightarrow 2^C$, defined by the instructions in algorithm (47), together with the map $c(\cdot) = f^0(\cdot)$ satisfy assumptions (i) and (ii) of theorem (1.3.27), provided we define $z \in C$ to be desirable if $h^0(z) = 0$. Hence, show that if $\{z_i\}$ is a sequence constructed by algorithm (47), then either this sequence is finite and its last element is desirable, or it is infinite and then every accumulation point of this sequence is desirable. Show that algorithm (47) cannot jam up at a point z_i , cycling in the loop defined by steps 4–8, while producing a sequence $\epsilon_j \rightarrow 0$, as $j \rightarrow \infty$. ■

The reader is now invited to use his ingenuity in constructing other implementations for algorithm (27). As the author has done in (47), he may make use of the guidance provided by the algorithm model (1.3.26) and the convergence theorem (1.3.27) and of an algorithm for searching a line for a minimum (or stationary point) of a real-valued function. Further guidance as to how this may be done may be obtained from the models presented in Appendix A, and from the examples and exercises to be found in the next two sections.

4.3 Methods of Feasible Directions

In our presentation of methods for solving the problem,

$$1 \quad \min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

where $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, 2, \dots, m$, are continuously differentiable functions, we are not always following the order in which they were invented. Rather, we are following the order of increasing complexity. It is somewhat amusing to observe that the methods of feasible directions which were invented by Zoutendijk [Z4] in 1959 and, independently, in 1962–1963 by

Zukhovitskii *et al.* [Z8], preceded by several years both the modified method of centers (2.27) and the algorithm below (due to Topkis and Veinott [T1]), from which, logically, they should have outgrown and in terms of which they are most easily understood.

- 2 Remark.** An algorithm for solving (1) is referred to as a method of feasible directions if given a point z_i in the set C , defined by

$$3 \quad C = \{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

it determines a half-line $\{z \mid z = z_i + \mu h_i, \mu \geq 0\}$ passing through the interior of the set C , and then it chooses $z_{i+1} = z_i + \mu_i h_i$ on this half-line to be such that $f^0(z_{i+1}) < f^0(z_i)$. Thus, methods of feasible directions can only be used for solving (1) when C has an interior or a relative interior (i.e., it can be contained in a linear manifold and it has an interior relative to this manifold). ■

An examination of (2.45) in the proof of theorem (2.32) shows that this theorem remains valid when $\mu(z)$ is determined according to the rule

$$4 \quad f^0(z + \mu(z) h(z)) = \min\{f^0(z + \mu h(z)) \mid \mu \geq 0, z + \mu h(z) \in C\},$$

rather than according to the rule (2.28), i.e., that theorem (2.32) also holds for the algorithm below, which can be used on problem (1) under assumption (2.3).

- 5 Algorithm** (method of feasible directions, Topkis and Veinott [T1]).

Step 0. Compute a point $z_0 \in C$ and set $i = 0$.

Comment. See (2.30) for method of computing z_0 .

Step 1. Set $z = z_i$.

Step 2. Solve (2.22)–(2.25) to obtain $(h^0(z), h(z))$.

Step 3. If $h^0(z) = 0$, set $z_{i+1} = z_i$ and stop; else, go to step 4.

Step 4. Compute $\mu(z)$ to be the smallest positive scalar such that

$$6 \quad f^0(z + \mu(z) h(z)) = \min\{f^0(z + \mu h(z)) \mid \mu \geq 0, z + \mu h(z) \in C\}.$$

Comment. It can be seen that $\mu(z)$ exists because of (2.3).

Step 5. Set $z_{i+1} = z + \mu(z) h(z)$, set $i = i + 1$, and go to step 1. ■

The computation of $\mu(z)$ according to (6) (i.e., its eventual approximation) appears to be more difficult than according to (2.28). However, it does result in a larger decrease of the cost at each iteration.

The main objection that one can raise both to algorithm (2.27) and to algorithm (5), is that at each iteration one has to solve (2.22)–(2.25), a possibly large linear programming problem which involves *all* the constraint functions $f^i(\cdot)$, $i = 1, 2, \dots, m$. One's natural inclination would be to reduce

the dimension of this linear programming problem by removing all the inequalities, $-h^0 + f^i(z) + \langle \nabla f^i(z), h \rangle \leq 0$, in which $f^i(z) < 0$. However, it is not possible to show that theorem (2.32) remains valid for the resulting algorithm. In fact, counterexamples have been constructed by Wolfe [W2], showing that a zigzagging (jamming) phenomenon takes place when inactive constraints are removed from (2.24), with the resulting sequence $\{z_i\}$ converging to a point z such that $h^0(z) < 0$. This is due to the fact that the inactive constraints, when not used in computing $h(z)$, may force the step size $\mu(z)$ to go to zero, even if one is nowhere near a point z satisfying $h^0(z) = 0$. Zoutendijk was well aware of this jamming phenomenon, and therefore proposed, in retrospect, to remove only those constraints $-h^0 + f^i(z) + \langle \nabla f^i(z), h \rangle \leq 0$ from (2.24) for which $f^i(z)$ was sufficiently negative to permit an adequate step size. This, of course, required him to invent a test for "sufficiently negative" to be inserted into algorithm (5). Before we can present one of Zoutendijk's algorithms (he invented several) and some of its modifications and implementations, we must introduce a few new quantities and discuss their properties.

We begin by defining the " ϵ -active index set." For any $\epsilon \geq 0$, and $z \in C$, let

$$7 \quad J_\epsilon(z) = \{0\} \cup \{i \mid f^i(z) + \epsilon \geq 0, i \in \{1, 2, \dots, m\}\},$$

and let $h_\epsilon^0: C \rightarrow \mathbb{R}^1$ be defined by

$$8 \quad h_\epsilon^0(z) = \min_{h \in S^*} \max_{i \in J_\epsilon(z)} \langle \nabla f^i(z), h \rangle,$$

where

$$9 \quad S^* = \{h \in \mathbb{R}^n \mid |h^i| \leq 1, i = 1, 2, \dots, n\}.$$

Hence, by (1.2.8), if \hat{z} is optimal for problem (1), then $h_0^0(\hat{z}) = 0$. Now, as in exercise (2.20), consider the problem in \mathbb{R}^{n+1} for $z \in C$:*

$$10 \quad \text{minimize} \quad h^0,$$

subject to

$$11 \quad -h^0 + \langle \nabla f^i(z), h \rangle \leq 0, \quad i \in J_\epsilon(z),$$

$$12 \quad |h^i| \leq 1, \quad i = 1, 2, \dots, n.$$

Let $h_\epsilon(z) \in S^*$ be any vector satisfying

$$13 \quad h_\epsilon^0(z) = \max_{i \in J_\epsilon(z)} \langle \nabla f^i(z), h_\epsilon(z) \rangle;$$

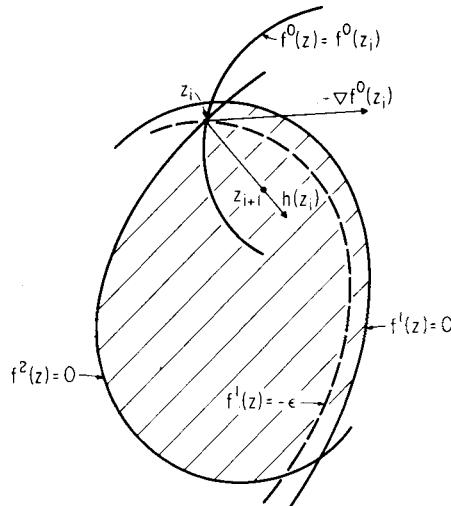
* When $J_\epsilon(z)$ contains a large number of indices, it may be advantageous to use decomposition techniques; see [V2].

then it is quite easy to see that $\bar{h}_\epsilon(z) = (h_\epsilon^0(z), h_\epsilon(z))$ is optimal for the problem (10)–(12). Conversely, suppose that $(h_\epsilon^0, h_\epsilon)$ is optimal for (10)–(12). Then we must have $h_\epsilon^0(z) = h_\epsilon^0$, and the vector $h_\epsilon(z) = h_\epsilon$ satisfies (13). Thus, given $\epsilon \geq 0$ and $z \in C$, we calculate $h_\epsilon^0(z)$ by solving (10)–(12).

- 14 Exercise.** Show that the function $h_\epsilon^0(\cdot)$ and the set $J_\epsilon(z)$ have the properties,

- 15 $h_\epsilon^0(z) \leq 0$ for any $z \in C$, for any $\epsilon \geq 0$;
- 16 $J_\epsilon(z) \supset J_{\epsilon'}(z)$ whenever $\epsilon > \epsilon'$, for any $z \in C$;
- 17 $h_\epsilon^0(z) \geq h_{\epsilon'}^0(z)$ whenever $\epsilon > \epsilon'$, for any $z \in C$;
- 18 for any $\epsilon \geq 0$, for any $z \in C$, there exists a $\rho > 0$ such that $J_{\epsilon+\rho}(z) = J_\epsilon(z)$;
- 19 for any $\epsilon \geq 0$, for any $z \in C$, there exists a $\rho > 0$ such that $J_\epsilon(z') \subset J_\epsilon(z)$ for all $z' \in B(z, \rho) = \{z \in C \mid \|z' - z\| \leq \rho\}$. ■

Zoutendijk's algorithm, below, can be used to compute points in C satisfying $h_0^0(z) = 0$, under assumption (2.3).*



Zoutendijk method of feasible directions.

* In principle, we do not need assumption (2.5) for the algorithms in this section. However, when (2.5) is not satisfied, they may stop at z_0 , or fail to compute a z_0 in a finite number of iterations, i.e., they may be useless (see (2.30)).

20 **Algorithm** (method of feasible directions, Zoutendijk [Z4]).

Step 0. Compute a $z_0 \in C$; select an $\epsilon' > 0$, an $\epsilon'' \in (0, \epsilon')$ and a $\beta \in (0, 1)$; set $i = 0$.

Comment. It is usual to set $\beta = 1/2$. See (2.30) for computation of z_0 .

Step 1. Set $\epsilon = \epsilon'$.

Step 2. Set $z = z_i$.

Step 3. Compute $\bar{h}(z) = (h_\epsilon^0(z), h_\epsilon(z))$ by solving (10)–(12).

Step 4. If $h_\epsilon^0(z) \leq -\epsilon$, set $h(z) = h_\epsilon(z)$ and go to step 7; else, go to step 5.

Step 5. If $\epsilon \leq \epsilon''$, compute $h_0^0(z)$ by solving (10)–(12) (with $\epsilon = 0$) and go to step 6; else, set $\epsilon = \beta\epsilon$ and go to step 3.

Step 6. If $h_0^0(z) = 0$, set $z_{i+1} = z$ and stop; else, set $\epsilon = \beta\epsilon$ and go to step 3.

Step 7. Compute $\lambda(z) \geq 0$ such that

21 $\lambda(z) = \max\{\lambda \mid f^i(z + \alpha h(z)) \leq 0 \text{ for all } \alpha \in [0, \lambda] \text{ and } i = 1, 2, \dots, m\}$.

Comment. Here, $\lambda(z)$ is the largest λ for which $z + \alpha h(z) \in C$, for all $\alpha \in [0, \lambda]$.

Step 8. Compute $\mu(z) \in [0, \lambda(z)]$ to be the smallest value in that interval such that

22 $f^0(z + \mu(z) h(z)) = \min\{f^0(z + \mu h(z)) \mid \mu \in [0, \lambda(z)]\}$.

Comment. It can be seen that $\mu(z)$ will always exist if z_0 satisfies assumption (2.3).

Step 9. Set $z_{i+1} = z + \mu(z) h(z)$, set $i = i + 1$, and go to step 2. ■

23 **Exercise.** Suppose that the set $C'(z_0) = \{z \in C \mid f^0(z) \leq f^0(z_0)\}$ is compact. Show that if the sequence $\{z_i\}$ generated by algorithm (20) is finite, then its last element, z_k , must satisfy $h_0^0(z_k) = 0$, and that if $\{z_i\}$ is infinite, then every accumulation point \hat{z} of $\{z_i\}$ must satisfy $h_0^0(\hat{z}) = 0$. [Hint: Use theorem (1.3.42) as follows: Set $S = S^*$, $c(z) = f^0(z)$, $\psi(\epsilon, z) = h_\epsilon^0(z)$, $H(\epsilon, z) = \{h \in S^* \mid h_\epsilon^0(z) = \max_{i \in J_\epsilon(z)} \langle \nabla f^i(z), h \rangle\}$, $M(\epsilon, z) = \{z + \mu h \in C \mid h \in H(\epsilon, z)\}$, $f^0(z + \mu h) = \min_{t \in [0, \lambda(z, h)]} f^0(z + th)$, where $\lambda(z, h) = \max\{\lambda \geq 0 \mid z + th \in C \text{ for all } t \in [0, \lambda]\}$, and $T = C'(z_0)$. Then prove the following two lemmas:

Lemma 1. For any $\mu > 0$, for any $\epsilon \geq 0$, and for any $i \in \{0, 1, 2, \dots, m\}$, there exists an $s^i(\mu) > 0$ such that if for $z \in C'(z_0)$, with the property that $i \in J_\epsilon(z)$,

24 $h_\epsilon^0(z) \leq -\mu$,

then

$$25 \quad f^i(z + sh) - f^i(z) \leq \frac{-s\mu}{2},$$

for all $s \in [0, s^i(\mu)]$, for all $h \in H(\epsilon, z)$.

Lemma 2. Given any $\epsilon > 0$ and any $i \in \{1, 2, \dots, m\}$, there exists a $t^i(\epsilon) > 0$ such that $f^i(z + th) \leq 0$ for all $z \in \{z \in C'(z_0) \mid f^i(z) < -\epsilon\}$ for all $h \in S^*$, and for all $t \in [0, t^i(\epsilon)]$. (Construct a contradiction.)

Once these two lemmas are established, it is easy to see that the conditions (1.3.42)(i) and (1.3.42)(ii) are satisfied. Next, (1.3.42)(iii) is satisfied, since $f^0(\cdot)$ is continuous and $C'(z_0)$ is compact. That condition (1.3.42) (iv) is satisfied follows directly from lemma 1, while (1.3.42)(v) can be established by means of arguments similar to the ones used for (36)–(39). Condition (1.3.42)(vi) can be established by making use of lemmas 1 and 2. Finally, we see that (1.3.42)(vii) is true by definition. ■

To remove the dependence of the construction of z_{i+1} on the current value of ϵ , we only need to change the instruction in step 9 of algorithm (20) from “go to step 2” to “go to step 1.” The resulting algorithm, stated below, can then be shown to be of the form of the model (1.3.9), as we shall soon see.

26 Algorithm (method of feasible directions, Polak [P1]).

Step 0. Compute a $z_0 \in C$, select an $\epsilon' > 0$, an $\epsilon'' \in (0, \epsilon')$, a $\beta \in (0, 1)$, and set $i = 0$.

Comment. It is usual to set $\beta = 1/2$. See (2.30) for a method to compute a $z_0 \in C$, using algorithm (2.27), or (5), or (20) or (26).

Step 1. Set $z = z_i$.

Step 2. Set $\epsilon_0 = \epsilon'$, and set $j = 0$.

Step 3. Compute a vector $\bar{h}_{\epsilon_j}(z) = (h_{\epsilon_j}^0(z), h_{\epsilon_j}(z))$ by solving (10)–(12) for $\epsilon = \epsilon_j$.

Step 4. If $h_{\epsilon_j}^0 \leq -\epsilon_j$, set $\epsilon(z) = \epsilon_j$, set $h(z) = h_{\epsilon_j}(z)$, and go to step 7; else, go to step 5.

Comment. Do not store $\epsilon(z)$; it is defined only because it will be needed in proving convergence properties of algorithm (26) later.

Step 5. If $\epsilon_j \leq \epsilon''$, compute $h_0^0(z)$ by solving (10)–(12) with $\epsilon = 0$, and go to step 6; else, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3.

Step 6. If $h_0^0(z) = 0$, set $z_{i+1} = z$, and stop; else, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3.

Step 7. Compute $\lambda(z) > 0$ such that (21) is satisfied.

Step 8. Compute $\mu(z) \in [0, \lambda(z)]$ to be the smallest value in that interval satisfying (22).

Comment. $\mu(z)$ will always exist if z_0 satisfies assumption (2.3).

Step 9. Set $z_{i+1} = z + \mu(z) h(z)$, set $i = i + 1$, and go to step 1. ■

Note that in the version (20), the value of ϵ is allowed to decrease continuously, while in the version (26), the value of $\epsilon(\epsilon_j)$ is reset to its original value of ϵ' at each iteration. Both of these approaches have their advantages and their disadvantages. For example, in the algorithm (20), for some reason, ϵ may become quite small while z_i is still quite far from a point \hat{z} satisfying $h_0^0(\hat{z}) = 0$. As a result, some of the inactive constraints, satisfying $-k\epsilon \leq f^j(z_i) < -\epsilon$, with k small, may force the step size $\mu(z)$ to become unnecessarily small, causing a slowdown in the convergence process. This would not occur in the version (26). However, as z_i approaches a point \hat{z} which satisfies $h_0^0(z) = 0$, algorithm (26) may spend too much time in decreasing ϵ_j , from the value ϵ' to the much smaller value $\epsilon(z)$ at each iteration. In practice, one might use some heuristic to switch from the version (20) to the version (26), and, if need be, back again, in the course of a calculation. This can obviously be done very easily, since the two algorithms differ in only one small detail.

- 27 **Lemma.** Algorithm (26) cannot cycle indefinitely between steps 3 and 6 while constructing a sequence $\epsilon_j \rightarrow 0$ as $j \rightarrow \infty$.

Proof. Suppose that $z_i \in C$ is such that $h_0^0(z_i) = 0$. Then, after a finite number of reductions of ϵ_j , the algorithm will construct an $\epsilon_j \leq \epsilon''$. It will then determine in step 6 that $h_0^0(z_i) = 0$, and it will stop after setting $z_{i+1} = z_i$. Next, suppose that $z_i \in C$ is such that $h_0^0(z_i) < 0$. Then, by (18), there exists a $\rho > 0$ such that for all $\epsilon_j \in [0, \rho]$, $J_{\epsilon_j}(z_i) = J_0(z_i)$, and hence,

$$h_{\epsilon_j}^0(z_i) = h_0^0(z_i) < 0 \quad \text{for all } \epsilon_j \in [0, \rho].$$

Let $j' \geq 0$ be the smallest integer such that $\beta^{j'}\epsilon' \leq \min\{\rho, -h_0^0(z_i)\}$; then we must have

$$28 \quad h_{\epsilon_{j'}}(z_i) \leq -\epsilon_{j'}, \quad \epsilon_{j'} = \beta^{j'}\epsilon'.$$

Hence, a new point, z_{i+1} , will be constructed after at most j' reductions of ϵ_j , i.e., when $j = j'$. Consequently, algorithm (26) is well-defined. ■

- 29 **Exercise.** Show that lemma (27) is also true for algorithm (20). ■

- 30 **Remark.** Suppose that $\{z_i\}$ is a sequence constructed by algorithm (26) (or algorithm (20)); then $f^0(z_0) > f^0(z_1) > f^0(z_2) > \dots$, as can be seen from the mean-value theorem (B.1.1) and the facts that $f^0(\cdot)$ is continuously differentiable and that $\langle \nabla f^0(z_i), h(z_i) \rangle \leq -\epsilon(z_i) < 0$. However, the sequence

$h_{\epsilon(z_i)}^0(z_i)$, $i = 0, 1, 2, \dots$, has no demonstrable monotonicity properties ($h(z_i) = h(z)$, $\epsilon(z_i) = \epsilon(z)$ for $z = z_i$ in step 4 of (26)). ■

Although (26) is an algorithm with “ ϵ reduction,” it is not of the form of the model (1.3.33), but of the form of the model (1.3.9), with $c(\cdot) = f^0(\cdot)$, $T = C$, $A: C \rightarrow 2^C$ defined by the instructions in (26), and $z \in C$ defined to be desirable if $h_0^0(z) = 0$. The reason we can identify (26) with the form (1.3.9) and not with the form (1.3.33) is that the reduction of ϵ_j in (26) is carried out on the basis of a test that does not involve the values of the function $f^0(\cdot)$.

The instructions in (26) defining the map $A: C \rightarrow 2^C$ are rather complex, and it may help the reader if we now exhibit it explicitly. First, suppose that $z_i \in C$ is such that $h_0^0(z_i) = 0$. Then, as we have seen in lemma (27), after a finite number of reductions of ϵ_j , algorithm (20) sets $z_{i+1} = z_i$, i.e.,

$$31 \quad A(z) = \{z\}, \quad \text{when } h_0^0(z) = 0.$$

Now suppose that $z_i \in C$ is such that $h_0^0(z_i) < 0$. Then algorithm (26) computes a unique integer j' such that $h_{\beta^{j'}\epsilon}^0(z_i) \leq -\beta^{j'}\epsilon$ and $h_{\beta^{j'}\epsilon}^0(z_i) > -\beta^{j'}\epsilon$, for $j > j'$, j an integer, and sets $\epsilon(z_i) = \beta^{j'}\epsilon$. Thus, $\epsilon(z_i)$ is uniquely determined by z_i . Next, let $S^*(z_i) \subset S^*$ be defined as follows:

$$32 \quad S^*(z_i) = \{h \in S^* \mid h_{\epsilon(z_i)}^0(z_i) = \max_{j \in J_{\epsilon(z_i)}(z_i)} \langle \nabla f^j(z_i), h \rangle\}.$$

Then we see that the point z_{i+1} constructed by algorithm (26) must belong to the set

$$33 \quad \{z' = z_i + \mu' h' \in C \mid h' \in S^*(z_i), \mu' \geq 0, f^0(z') = \min\{f^0(z_i + \mu h) \mid \mu \geq 0, z_i + \mu h \in C\}\}.$$

Consequently (under any assumption, such as (2.3), which ensures that the min in (33) exists),

$$34 \quad \begin{aligned} A(z) &= \{z' = z + \mu' h' \in C \mid h' \in S^*(z), \mu' \geq 0, f^0(z') \\ &= \min\{f^0(z + \mu h) \mid \mu \geq 0, z + \mu h \in C\}\}, \quad \text{when } h_0^0(z) < 0. \end{aligned}$$

Taken together, (31) and (34) define the set $A(z) \subset C$ for every $z \in C$.

We can now establish the convergence properties of algorithm (26).

- 35 **Theorem.** Suppose that $A(z)$ is well-defined by (31) and (34) for all $z \in C$. Let z_0, z_1, z_2, \dots be a sequence constructed by algorithm (26) for problem (1) (where we had assumed all functions to be continuously differentiable), and suppose that (2.3) is satisfied by z_0 . Then, either the sequence $\{z_i\}$ is finite and its last element, say z_k , satisfies $h_0^0(z_k) = 0$, or else $\{z_i\}$ is infinite and every accumulation point z of $\{z_i\}$ satisfies $h_0^0(z) = 0$ (see also (1.3.65)).

Proof. We have shown above that algorithm (26) is of the form of the model (1.3.9). To prove theorem (35) we only need to show that the assumptions (i) and (ii) of theorem (1.3.10) are satisfied by the maps $c(\cdot) = f^0(\cdot)$ and $A(\cdot)$ defined by (31), (34), for $z \in C$ defined to be desirable if $h_0^0(z) = 0$ (recall, we set $T = C$). (Since algorithm (26) can stop constructing new points only when it encounters a z_i such that $h_0^0(z_i) = 0$, the finite sequence case is trivial.)

Since by assumption $f^0(\cdot)$ is continuously differentiable, (i) is satisfied, and hence we are left with establishing (ii). Suppose that $z \in C$ is such that $h_0^0(z) < 0$. Then we must have $\epsilon(z) > 0$ and $h_{\epsilon(z)}(z) \leq -\epsilon(z) < 0$. We shall now show that for points $z' \in C$ sufficiently close to z , we must have $\epsilon(z') \geq \epsilon(z)/\beta$. From (19) it follows that there exists a $\rho > 0$ such that

$$36 \quad J_{\epsilon(z)}(z') \subset J_{\epsilon(z)}(z), \quad \text{for all } z' \in B(z, \rho).$$

Let $\theta: \mathbb{R}^n \rightarrow \mathbb{R}^1$ be defined by

$$37 \quad \theta(z') = \min_{h \in S^*} \max_{i \in J_{\epsilon(z)}(z)} \langle \nabla f^i(z'), h \rangle;$$

then, by (B.3.20), $\theta(\cdot)$ is a continuous function, and hence there exists a $\rho' \in (0, \rho]$ such that

$$38 \quad |\theta(z') - h_{\epsilon(z)}^0(z)| \leq \epsilon(z)(1 - \beta), \quad \text{for all } z' \in B(z, \rho'),$$

since $\theta(z) = h_{\epsilon(z)}^0(z)$. Consequently, because of (36),

$$39 \quad h_{\epsilon(z)}^0(z') \leq \theta(z') \leq -\epsilon(z)\beta, \quad \text{for all } z' \in B(z, \rho').$$

Finally, making use of (17), we obtain

$$40 \quad h_{\epsilon(z)\beta}^0(z') \leq h_{\epsilon(z)}^0(z') \leq -\epsilon(z)\beta, \quad \text{for all } z' \in B(z, \rho').$$

Consequently, since $\epsilon(z') = \beta^{j'}\epsilon'$ is such that $h_{\epsilon(z')}^0(z') \leq -\epsilon(z')$, we must have, because of (40),

$$41 \quad \epsilon(z') \geq \epsilon(z)\beta, \quad \text{for all } z' \in B(z, \rho').$$

Next, since the functions $f^i(\cdot)$, $i = 1, 2, \dots, m$, are continuous and S^* is compact, it follows from theorem (B.3.1) that there exists a $\rho'' \in (0, \rho']$ and a $t' > 0$ such that for $i = 1, 2, \dots, m$,

$$42 \quad |f^i(z' + th) - f^i(z')| \leq \epsilon(z)\beta, \quad \text{for all } z' \in B(z, \rho''), \quad h \in S^*, \quad t \in [0, t'].$$

Again, since the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, are continuously differen-

tiable and S is compact, it follows from theorem (B.3.7) that there exists a $\tilde{\rho} \in (0, \rho'']$ and a $t'' \in (0, t']$ such that for $i = 0, 1, 2, \dots, m$,

$$43 \quad |\langle \nabla f^i(z' + th), h \rangle - \langle \nabla f^i(z'), h \rangle| \leq \frac{\epsilon(z) \beta}{2},$$

for all $z' \in B(z, \tilde{\rho}), h \in S^*, t \in [0, t']$.

Now consider all the functions $f^i(\cdot)$, $i \in J_{\epsilon(z')}(z')$ for any $z' \in B(z, \tilde{\rho})$. Then, by the mean-value theorem, for a given $h(z') \in S^*(z')$, we must have, for any $t \in [0, t'']$, and $i \in \{0, 1, 2, \dots, m\}$,

$$44 \quad f^i(z' + th(z')) = f^i(z') + t \langle \nabla f^i(z' + \xi^i h(z')), h(z') \rangle,$$

where $\xi^i \in [0, t]$. Since for any $h(z') \in S^*(z')$, we must have $\langle \nabla f^i(z'), h(z') \rangle \leq -\epsilon(z') \leq -\epsilon(z) \beta$, it follows from (43) and (44) that for all $i \in J_{\epsilon(z')}(z')$,

$$45 \quad f^i(z' + th(z')) - f^i(z') \leq \frac{-\epsilon(z) \beta t}{2}, \quad t \in [0, t''].$$

Examining (45) and (42), we conclude that $\lambda(z')$, as computed in step 7 of (26), must satisfy $\lambda(z') \geq t''$, since because of (42) and (45),

$$46 \quad f^i(z' + th(z')) \leq 0 \quad \text{for all } t \in [0, t''], i = 1, 2, \dots, m.$$

Consequently, because of (46) and (45), we must have

$$47 \quad f^0(z' + \mu(z') h(z')) - f^0(z') \leq f^0(z' + t'' h(z')) - f^0(z') \leq \frac{-\epsilon(z) \beta t''}{2}.$$

Since (47) obviously holds for all $z' + \mu(z') h(z') \in A(z')$, with $z' \in B(z, \tilde{\rho})$, we conclude that (ii) of theorem (1.3.10) is satisfied by the maps $c(\cdot) = f^0(\cdot)$ and $A(\cdot)$ defined by (31) and (34), and we are done. ■

- 48 **Exercise.** Show that, in principle, one can substitute for the constraint (12) any constraint of the form $h \in S$, where S is a compact set containing the origin in its interior, and obtain an algorithm of the form of (26) for which the conclusions of the above theorem are valid. In particular, examine the advisability of using the constraint $h \in S' = \{h \in \mathbb{R}^n \mid \|h\| \leq 1\}$. (Do you get a subproblem that is readily solvable at each iteration?) ■

The only calculations indicated by the instructions in (26) which are not implementable, i.e., which cannot be carried out exactly in a finite number of operations (even on a digital computer with an infinite word length), are those of computing $\lambda(z)$, the upper bound on the step length $\mu(z)$, and the actual step length $\mu(z)$. (A similar situation is seen to exist also in the algorithms (5) and (20).) We now present two implementable substitutions for

the conceptual steps 7 and 8 in algorithm (26). These can also be used in (20), and, after appropriate modification, in (5).

- 49 **Proposition.** Suppose that a $\rho > 0$, a $\tilde{\beta} \in (0, 1)$ and an $\alpha \in (0, 1)$ are given, and suppose that steps 7 and 8 of algorithm (26) are replaced by the steps 7' and 8' below. Then theorem (35) remains valid for the resulting modification of (26) (compare (2.1.35)).*

Step 7'. Compute the smallest integer $k \geq 0$ such that

$$50 \quad f^0(z + \tilde{\beta}^k \rho h(z)) - f^0(z) - \tilde{\beta}^k \rho \alpha \langle \nabla f^0(z), h(z) \rangle \leq 0,$$

$$51 \quad f^i(z + \tilde{\beta}^k \rho h(z)) \leq 0 \quad \text{for } i = 1, 2, \dots, m.$$

Step 8'. Set $\mu(z) = \tilde{\beta}^k \rho$. ■

- 52 **Exercise.** Prove proposition (49), and show that if the set $\{z \in C \mid h_0^0(z) = 0\}$ consists of a finite number of points only, then the assumptions of theorem (1.3.66) are satisfied for the algorithm with step 7'. ■

- 53 **Proposition.** Suppose that the functions $f^i(\cdot)$ are convex and bounded from below, for $i = 0, 1, 2, \dots, m$, and that for a given $\alpha \in (0, 1/2)$, the steps 7 and 8 of algorithm (26) are replaced by the steps 7'' and 8'' below. Then theorem (35) remains valid for the resulting modification of algorithm (26) (compare (2.1.33)).

Step 7''. Compute $\lambda^i > 0$, $i = 0, 1, 2, \dots, m$, such that

$$54 \quad (1 - \alpha) \lambda^0 \langle \nabla f^0(z), h(z) \rangle \leq f^0(z + \lambda^0 h(z)) - f^0(z) \leq \alpha \lambda^0 \langle \nabla f^0(z), h(z) \rangle,$$

$$55 \quad f^i(z) + \alpha \lambda^i \langle \nabla f^i(z), h(z) \rangle \leq f^i(z + \lambda^i h(z)) \leq 0 \quad \text{for } i \neq 0, \quad i \in J_{\epsilon(z)}(z),$$

$$56 \quad -\alpha \lambda^i \leq f^i(z + \lambda^i h(z)) \leq 0 \quad \text{for } i \in \bar{J}_{\epsilon(z)}(z),$$

where $\bar{J}_{\epsilon(z)}(z)$ is the complement of $J_{\epsilon(z)}(z)$ in $\{0, 1, 2, \dots, m\}$.

Step 8''. Set $\mu(z) = \min\{\lambda^0, \lambda^1, \dots, \lambda^m\}$. ■

- 57 **Exercise.** Prove proposition (53), and show when theorem (1.3.65) applies to the algorithm with step 7''. ■

It is reasonably clear that the above two implementable modifications of steps 7 and 8 of algorithm (26), which preserve its convergence properties as expressed in theorem (35), are not the only ones possible. The reader is now invited to exercise his ingenuity in finding yet other ways of substituting for the conceptual steps 7 and 8 in (26).

We shall now examine the possibility of simplifying the calculations required to compute a feasible direction vector in an algorithm such as (5),

* We may use the same β as in step 0 of (26), i.e., we may set $\beta = \tilde{\beta}$, if we wish.

(20) and (26). We shall concentrate on (26), leaving the corresponding rather obvious modifications of (5) and (20) as an exercise for the reader. An examination of the linear programming problem (10)–(12) shows that the variable h^0 appears in every inequality containing a $\nabla f^i(z)$. It is possible to construct methods of feasible directions, similar to algorithm (26) (or to (20), or to (5)) in which this “coupling effect” is either reduced or eliminated, resulting in a somewhat simpler linear programming problem to be solved at each iteration. We shall describe two such methods, the first one devised by Zoutendijk [Z4], while the second one is due to Zukhovitskii *et al.* [Z8]. (To be completely precise, we shall present minor variations of the methods in [Z4], [Z8], the original versions being “time-varying” and having the structure of algorithm (20).)

We begin with the Zoutendijk approach. Consider again the problem (1) and suppose that some of the functions $f^i(\cdot)$, $i \in \{0, 1, 2, \dots, m\}$, are affine. For any $\epsilon \geq 0$ and any $z \in C$ (defined in (3)), let $J_\epsilon^A(z)$ and $J_\epsilon^N(z)$ be subsets of the index set $J_\epsilon(z)$ (defined in (7)) such that

$$58 \quad J_\epsilon^A(z) \cap J_\epsilon^N(z) = \emptyset, \quad J_\epsilon^A(z) \cup J_\epsilon^N(z) = J_\epsilon(z), \quad 0 \in J^N(z),$$

and such that for every $i \in J_\epsilon^A(z)$, $f^i(\cdot)$ is an affine function. Now, for any $\epsilon \geq 0$ and any $z \in C$, consider the linear programming problem,

$$59 \quad \text{minimize} \quad h^0,$$

subject to

$$60 \quad -h^0 + \langle \nabla f^i(z), h \rangle \leq 0, \quad i \in J_\epsilon^N(z),$$

$$61 \quad \langle \nabla f^i(z), h \rangle \leq 0, \quad i \in J_\epsilon^A(z),$$

$$62 \quad |h^i| \leq 1, \quad i = 1, 2, \dots, n.$$

Let us denote a solution of (58)–(61) by $(\tilde{h}_\epsilon^0(z), \tilde{h}_\epsilon(z)) \in \mathbb{R}^{n+1}$, where $\tilde{h}_\epsilon(\cdot)$ is not necessarily unique. It is not difficult to see that $\tilde{h}_\epsilon^0(\cdot)$ is a function from \mathbb{R}^n into \mathbb{R}^1 , with the following important properties: (i)

$$63 \quad \begin{aligned} \tilde{h}_\epsilon^0(z) &= \min\{h^0 \mid -h^0 + \langle \nabla f^i(z), h \rangle \leq 0, i \in J_\epsilon^N(z); \\ &\quad \langle \nabla f^i(z), h \rangle \leq 0, i \in J_\epsilon^A(z); |h^i| \leq 1, i = 1, 2, \dots, n\}, \end{aligned}$$

and (ii) for any $z \in C$ and any $\epsilon \geq 0$,

$$64 \quad \tilde{h}_\epsilon^0(z) \leq h_\epsilon^0(z),$$

where $h_\epsilon^0(\cdot)$ is defined as in (8).

65 **Proposition.** Suppose that \hat{z} is optimal for (1); then $\tilde{h}_0^0(\hat{z}) = 0$. ■

66 Exercise. Prove proposition (65). [Hint: Use the mean-value theorem (B.1.1) to construct a contradiction.] ■

67 Remark. Note that because of (64), $\tilde{h}_0^0(\cdot)$ may have fewer zeros in C than $h_0^0(\cdot)$, and hence, $\tilde{h}_0^0(z) = 0$ may be considered to be a stronger necessary condition of optimality than $h_0^0(z) = 0$. Note also that (59)–(62) is a simpler problem than (10)–(12), from the simplex algorithm point of view. ■

It is not difficult now to see that property (64), together with the property stated in (65), ensure that the substitution of (59)–(62) for (10)–(12) in the calculation of a feasible direction vector $h(z)$ in (26) (or in (20)), does not affect the convergence properties of the resulting algorithm, as stated in theorem (35).

68 Exercise. Use the properties of $\tilde{h}_\epsilon^0(\cdot)$ stated in (63) and (65) to show that theorem (35) remains valid for the modification of algorithm (26), resulting from the substitution of step 3' below for step 3 in algorithm (26).

Step 3'. Compute a vector $\bar{h}_{\epsilon_j}(z) = (h_{\epsilon_j}^0(z), h_{\epsilon_j}(z))$ by solving (59)–(62) for $\epsilon = \epsilon_j$. ■

The advantages of this modification are readily seen when many of the $f^i(\cdot)$ are of the form $f^i(z) = \langle q_i, z \rangle + d_i$, $q_i \in \mathbb{R}^n$, $d^i \in \mathbb{R}^1$. In particular, step 3' results in an enormous simplification when the constraints $z \geq 0$, or $|z^i| \leq 1$, $i = 1, 2, \dots, n$, are among those specifying the set C , in which case, the number of inequalities in (59)–(62) which are significant with respect to the simplex algorithm can be much smaller than in (10)–(12), and hence, (59)–(62) can be much easier to solve. As we shall see a little later, the modification of (26) using the step 3' in (68) is much more suitable for solving discrete optimal control problems than the original version (26).

69 Exercise. Suppose that some of the functions $f^i(\cdot)$, $i \in \{0, 1, 2, \dots, m\}$, are affine. Show that both algorithms (5) and (20) can be modified to take advantage of this fact by removing $-h^0$ from the corresponding inequalities in (2.22)–(2.25), and in (10)–(12), respectively. Show that this modification does not affect the convergence properties (as defined in (2.32) and in (35)) of the resultant modification. ■

We now present the approach of Zukhovitskii *et al.* [Z8], which differs from the Zoutendijk approach in that it does not depend on the presence of affine functions among the $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$. For any $\epsilon > 0$ and any $z \in C$, consider the linear programming problem,

$$70 \quad \text{minimize} \quad \langle \nabla f^0(z), h \rangle,$$

subject to

$$71 \quad \langle \nabla f^i(z), h \rangle + \epsilon \leq 0, \quad i \neq 0, \quad i \in J_\epsilon(z),$$

$$72 \quad |h^i| \leq 1, \quad i = 1, 2, \dots, n.$$

We shall denote a solution of (70)–(72) by $\hat{h}_\epsilon(z)$. Note that just as in the case of all the other linear programming problems we have seen in this section, $\hat{h}_\epsilon(z)$ may not be unique. Also, we shall use the notation $\hat{h}_\epsilon^0(z) = \langle \nabla f^0(z), \hat{h}_\epsilon(z) \rangle$, i.e.,

$$73 \quad \begin{aligned} \hat{h}_\epsilon^0(z) &= \min\{\langle \nabla f^0(z), h \rangle \mid \langle \nabla f^i(z), h \rangle + \epsilon \leq 0, i \neq 0, i \in J_\epsilon(z); \\ &\quad |h^i| \leq 1, i = 1, \dots, n\}. \end{aligned}$$

- 74 **Proposition.** Suppose that for every $z \in C$ there exists a vector $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \neq 0, i \in J_0(z)$. Then for any $z \in C$, $\hat{h}_0^0(z) = 0$ if and only if $h_0^0(z) = 0$ ($h_\epsilon^0(\cdot)$ was defined in (8)). (Note that the condition in this lemma ensures that the Kuhn-Tucker constraint qualification is satisfied at every point in C ; see [C1], theorem (3.3.21).)

Proof. We make use of contraposition to establish (74). Thus, suppose that for some $z \in C$, $h_0^0(z) < 0$; then, by inspection, we find that $\hat{h}_0^0(z) < 0$, i.e., if $\hat{h}_0^0(z) = 0$, then we must also have $h_0^0(z) = 0$.

Now suppose that for some $z \in C$, $h_0^0(z) < 0$. Then, for some $h' \in S^* = \{h \mid |h^i| \leq 1, i = 1, 2, \dots, n\}$, we must have $\hat{h}_0^0(z) = \langle \nabla f^0(z), h' \rangle < 0$ and $\langle \nabla f^i(z), h' \rangle \leq 0$ for $i \neq 0, i \in J_0(z)$. Let $h \in \mathbb{R}^n$ be such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \neq 0, i \in J_0(z)$. Then for some $\lambda^1 > 0$, we shall have $\langle \nabla f^i(z), h' + \lambda^1 h \rangle < 0$ for all $i \in J_0(z)$ and for some $\lambda^2 > 0$, $\lambda^2(h' + \lambda^1 h) \in S^*$. Hence, we must also have $h_0^0(z) < 0$. Consequently, if $h_0^0(z) = 0$, then we must also have $\hat{h}_0^0(z) = 0$. ■

Thus, under the assumption stated in (74), computing the zeros of the function $\hat{h}_0^0(\cdot)$ seems to be as good an idea as computing the zeros of $h_0^0(\cdot)$, as far as the confidence one can have that the point that one has computed is optimal for (1). However, the linear programming problem (70)–(72) contains one variable less than the linear programming problem (10)–(12). In addition, the function $\hat{h}_\epsilon^0(\cdot)$ is generally more suited to optimal control problems than the function $h_\epsilon^0(\cdot)$, as we shall see in the subsection on optimal control problems.

- 75 **Theorem.** Suppose that for every $z \in C$ there exists a vector $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \neq 0, i \in J_0(z)$. Then theorem (35) remains valid for the modification of algorithm (26) which results from the substitution of step 3" below for step 3 in (26).

Step 3". Compute a vector $\hat{h}_{\epsilon_j}(z) = (h_{\epsilon_j}^0, h_{\epsilon_j}(z))$ by solving (70)–(72), with $\epsilon = \epsilon_j$, for a vector $h_{\epsilon_j}(z)$ ($= \hat{h}_{\epsilon_j}(z)$) and then setting $h_{\epsilon_j}^0(z) = \langle \nabla f^0(z), h_{\epsilon_j}(z) \rangle$.

Proof. First, to eliminate the need for complex references, let us agree to call the modification of (26) resulting from the substitution of step 3" for step 3 in (26) the “ZPP version” (for Zukhovitskii–Polyak–Primak). Next,

we begin by noting that the ZPP version is well-defined, since lemma (27) is obviously true for the ZPP version, and, because of assumption (2.3), $\mu(z)$ exists.

Since the case of a finite sequence $\{z_i\}$ constructed by the ZPP version is trivial, just as was the case in algorithm (26), we only need to show that any accumulation point \hat{z} of an infinite sequence $\{z_i\}$ constructed by the ZPP version, is desirable, i.e., $h_0^0(\hat{z}) = h_0^0(\hat{z}) = 0$. As we did in the proof of theorem (35), we define $c(\cdot) = f^0(\cdot)$, $T = C$, and we define $A : C \rightarrow 2^C$ by the instructions in the ZPP version of algorithm (26). Since $f^0(\cdot)$ is continuous, to complete our proof, we only need to show that assumption (ii) of theorem (1.3.10) is satisfied by the maps $c(\cdot)$ and $A(\cdot)$ under consideration. Suppose that at $z = z_i$, with $h_0^0(z_i) < 0$, algorithm (26) constructs $\epsilon(z)$ and a vector $\bar{h}_{\epsilon(z)}(z) = (h_{\epsilon(z)}^0(z), h_{\epsilon(z)}(z))$. Then we have, by construction, $h_{\epsilon(z)}^0(z) \leq -\epsilon(z)$. Now, by inspection, the vector $h_{\epsilon(z)}(z)$ satisfies (71), (72) for $\epsilon = \epsilon(z)$, and hence we must have

$$76 \quad h_{\epsilon(z)}^0(z) \leq h_{\epsilon(z)}(z) \leq -\epsilon(z).$$

Consequently, if we denote by $\epsilon'(z)$ the value of $\epsilon(z)$ constructed by the ZPP version of (26) at $z = z_i$, then, because of (76), we must have $\epsilon'(z) \geq \epsilon(z)$, and for any $\bar{h}_{\epsilon'(z)}(z)$ which is optimal for (70)–(72) for $\epsilon = \epsilon'(z)$, we must have $\langle \nabla f^i(z), h_{\epsilon'(z)}(z) \rangle \leq -\epsilon'(z) \leq -\epsilon(z)$ for all $i \in J_{\epsilon'(z)}(z)$. It now follows directly from the arguments used in the proof of theorem (35) for algorithm (26) that this theorem is also valid for the ZPP version of (26). ■

77 **Exercise.** Show that theorems (35) and (75) both remain valid when we replace the set $S^* = \{h \mid |h^i| \leq 1, i = 1, 2, \dots, n\}$ in (12) and in (72) by any compact set S containing the origin in its interior. ■

78 **Exercise.** Consider the linear programming problem,

$$79 \quad \text{minimize} \quad \langle f^0, z \rangle,$$

subject to

$$80 \quad \langle f_i, z \rangle \leq b^i, \quad i = 1, 2, \dots, m,$$

where $f_i \in \mathbb{R}^n$ for $i = 0, 1, 2, \dots, m$ and $b^i \in \mathbb{R}^1$ for $i = 1, 2, \dots, m$.

Show that the version of (26) using step 3' defined in (68), when applied to (79), (80), transforms this problem into a sequence of linear programming problems which are likely to be of considerably smaller dimension than (79), (80), i.e., that it becomes a *decomposition algorithm*. (Actually, there is no reason why we should not think of the methods of feasible directions as decomposition algorithms with respect to any problem on which they

can be used, since they effectively replace the original problem by a sequence of simpler problems.) ■

- 81 **Exercise.** Consider the function $\hat{h}^0 : C \rightarrow \mathbb{R}^1$, where C is as in (3), and $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, is as in (1), defined by
- 82
$$\begin{aligned} \hat{h}^0(z) &= \min\{\langle \nabla f^0(z), h \rangle \mid f^i(z) + \langle f^i(z), h \rangle \leq 0, i = 1, 2, \dots, m; \\ &\quad |h^i| \leq 1, i = 1, 2, \dots, n\}. \end{aligned}$$

Show that if for every $z \in C$ there exists a vector $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \neq 0$, $i \in J_0(z)$, then for any $z \in C$, $\hat{h}^0(z) = 0$ if and only if $h^0(z) = 0$ (where $h^0(\cdot)$ was defined in (2.18)), and hence, if and only if $h_0^0(z) = 0$. Show also that the function $\hat{h}^0(\cdot)$ cannot be substituted for the function $h^0(\cdot)$ in algorithm (5) without affecting its convergence properties, as expressed by theorem (2.32). ■

Methods of Feasible Directions for Problems with Equality Constraints

We shall now consider the problem,

83
$$\min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m, r(z) = 0\},$$

where all the functions in (83) are continuously differentiable and $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, \dots, m$, $r : \mathbb{R}^n \rightarrow \mathbb{R}^l$. We shall assume that (2.3) is satisfied in order to eliminate trivial and doubtful cases.

In dealing with problem (83), we must distinguish between two essentially different situations. In the first situation $r(\cdot)$ is either affine or of the form $r(z) = z_2 - g(z_1)$, where $z = (z_1, z_2)$. In the second situation, $r(\cdot)$ is of neither of these two forms. The reasons for this will soon become clear.

- 84 **Proposition.** Suppose that \hat{z} is optimal for (83) and that $r(\cdot)$ is affine or that $\partial r(\hat{z})/\partial z$ has maximum rank. Then

85
$$\min_{h \in S'(\hat{z})} \max_{i \in J_0(\hat{z})} \langle \nabla f^i(\hat{z}), h \rangle = 0,$$

where $J_0(\hat{z})$ is as defined in (7) and $S'(\hat{z}) = \{h \in \mathbb{R}^n \mid |h^i| \leq 1, i = 1, 2, \dots, n; (\partial r(\hat{z})/\partial z) h = 0\}$. Furthermore, if there exists a vector $h' \in \mathbb{R}^n$, such that $\langle \nabla f^i(\hat{z}), h' \rangle < 0$ for all $i \in J_0(\hat{z})$, $i \neq 0$, and $(\partial r(\hat{z})/\partial z) h' = 0$, then

86
$$\min\{\langle \nabla f^0(\hat{z}), h \rangle \mid \langle \nabla f^i(\hat{z}), h \rangle \leq 0, i \neq 0, i \in J_0(\hat{z}); h \in S'(\hat{z})\} = 0.$$

- 87 **Remark.** In view of proposition (84), to apply algorithms (20) or (26) to problem (83) when $r(\cdot)$ is affine, we only need to add the constraint $(\partial r(\hat{z})/\partial z) h = 0$ to the subproblem (10)–(12). Similarly, to apply the other

algorithms that we have seen in this section to problem (83) when $r(\cdot)$ is affine we only need to add the constraint $(\partial r(\hat{z})/\partial z) h = 0$ to the subproblems (2.22)–(2.25), (59)–(62) and (70)–(72).

- 88 **Proposition.** Suppose that $\rho > 0$, $\beta \in (0,1)$ and $\alpha \in (0,1)$ are given; that, in (83), $r(z) = z_2 - g(z_1)$, where $z = (z_1, z_2)$; and that $\partial r(\hat{z})/\partial z$ has maximum rank for all $z \in \{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}$. If the constraint $(\partial r(\hat{z})/\partial z)h = 0$ is added to (10)–(12), and the steps 7 and 8 of algorithm (26) are replaced by the steps 7' and 8' below, then theorem (35) remains valid for the resulting modification of (26).

Step 7'. Compute the smallest integer $k \geq 0$ such that, with $h(z) = (h_1(z), h_2(z))$,

$$\begin{aligned} 89 \quad f^0(z + (\beta^k \rho h_1(z), g(\beta^k \rho h_1(z)))) - f^0(z) - \beta^k \rho \alpha \langle \nabla f^0(z), h(z) \rangle &\leq 0, \\ 90 \quad f^i(z + (\beta^k \rho h_1(z), g(\beta^k \rho h_1(z)))) &\leq 0 \quad \text{for } i = 1, 2, \dots, m. \end{aligned}$$

Step 8'. Set $\mu(z) = \beta^k \rho$. ■

Similar modifications can also be introduced into algorithm (20) and into the ZPP algorithm.

For the general case, we can combine a method of feasible directions with a penalty function method, as illustrated by the following modification of algorithm (26):

- 91 **Algorithm** (hybrid method, Polak [P3], solves (83)).

Step 0. Compute a $z_0 \in C = \{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}$; select $\tilde{\epsilon} > 0$, $\epsilon' \in (0, \tilde{\epsilon})$, $\epsilon'' > 0$, $\beta' \in (0, 1)$, $\beta'' \in (0, 1)$, $\beta \in (0, 1)$ and set $i = 0$.

Step 1. Set $z = z_i$.

Step 2. Set $\epsilon_0 = \tilde{\epsilon}$ and set $j = 0$.

Step 3. Compute a vector $(h_{\epsilon_j}^0(z), h_{\epsilon_j}(z))$ by solving (10)–(12) for $\epsilon = \epsilon_j$ and with $f_{\epsilon''}^0(\cdot)$ taking the place of $f^0(\cdot)$, where

$$92 \quad f_{\epsilon''}^0(z) = f^0(z) + \frac{1}{2\epsilon''} \|r(z)\|^2.$$

Step 4. If $h_{\epsilon_j}^0(z) \leq -\epsilon_j$, set $h(z) = h_{\epsilon_j}(z)$ and go to step 6; else, go to step 5.

Step 5. If $\epsilon_j \leq \epsilon'$, set $\epsilon' = \beta' \epsilon'$, $\epsilon'' = \beta'' \epsilon''$ and go to step 1; else, set $\epsilon_{j+1} = \epsilon_j/2$, set $j = j + 1$, and go to step 3.

Step 6. Compute the smallest integer k for which (50), (51) are satisfied, with $f^0(\cdot)$ replaced by $f_{\epsilon''}^0(\cdot)$.

Step 7. Set $\mu(z) = \beta^k$.

Step 8. Set $z_{i+1} = z + \mu(z) h(z)$, set $i = i + 1$, and go to step 1. ■

- 93 Exercise.** Suppose that the constraint set $\Omega = \{z \in \mathbb{R}^n \mid f(z) \leq 0, r(z) = 0\}$ appearing in (83) satisfies the Kuhn-Tucker constraint qualification (see [C1]) at every $z \in \Omega$. Show that if the sequence $\{z_i\}$ constructed by algorithm (91) has accumulation points, then these accumulation points, \hat{z} , must satisfy the Kuhn-Tucker necessary condition of optimality,

$$94 \quad -\nabla f^0(\hat{z}) + \left(\frac{\partial r(\hat{z})}{\partial z} \right)^T \psi + \sum_{i \in J_0(z)} \mu^i \nabla f^i(\hat{z}) = 0,$$

for some multiplier vector $\psi \in \mathbb{R}^l$ and $\mu^i \leq 0$, $i \in J_0(z)$ (see (1.2.2)). [Hint: See lemma (1.84) and the discussion that follows it.] ■

Methods of Feasible Directions in Optimal Control

We shall now show, by means of two examples, how the various algorithms that we have seen in this section can be applied to certain classes of optimal control problems.

- 95 Example.** Consider the discrete optimal control problem,

$$96 \quad \text{minimize} \quad \sum_{i=0}^{k-1} f_i^0(x_i, u_i), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^l,$$

subject to

$$97 \quad x_{i+1} - x_i = f_i(x_i, u_i), \quad i = 0, 1, 2, \dots, k-1;$$

$$98 \quad x_0 = \hat{x}_0; \\ q^j(x_k) \leq 0, \quad j = 1, 2, \dots, m';$$

$$99 \quad |u_i| \leq 1, \quad i = 0, 1, \dots, k-1,$$

where all the functions are continuously differentiable in all of their arguments.

We convert problem (96)–(99) to the form (1) by setting

$$z = (u_0, u_1, \dots, u_{k-1}) \in \mathbb{R}^k$$

and by defining

$$100 \quad f^0(z) = \sum_{i=0}^{k-1} f_i^0(x_i(z), u_i),$$

$$101 \quad f^j(z) = q^j(x_k(z)), \quad j = 1, 2, \dots, m',$$

where for $i = 0, 1, 2, \dots, k$, $x_i(z)$ is determined by $x_0(z) = \hat{x}_0$, and

$$102 \quad x_{i+1}(z) - x_i(z) = f_i(x_i(z), u_i), \quad i = 0, 1, 2, \dots, k-1$$

(with $z = (u_0, u_1, \dots, u_{k-1})$). With these definitions, problem (96)–(99) becomes

$$103 \quad \begin{aligned} & \min\{f^0(z) \mid f^j(z) \leq 0, j = 1, 2, \dots, m'; \\ & \quad -1 - u_i \leq 0, -1 + u_i \leq 0, i = 0, 1, \dots, k-1\}, \end{aligned}$$

which is obviously of the form of problem (1). We can expect m' to be quite small in comparison to k , and hence, since we have $2k$ affine inequalities in (103), the most suitable method of feasible directions for use with this problem is algorithm (20) or algorithm (26), with (59)–(62) replacing (10)–(12) in the calculation of the feasible direction vector $h(z)$.

The only aspect of interest of the calculations for setting up (59)–(62), for this case, is the manner of calculating the various gradients that are needed. Problem (59)–(62) assumes the form,

$$104 \quad \text{minimize } h^0,$$

subject to

$$105 \quad -h^0 + \langle \nabla f^0(z), h \rangle \leq 0;$$

$$106 \quad -h^0 + \langle \nabla f^i(z), h \rangle \leq 0 \quad \text{for all } i \text{ such that } q^i(x_k(z)) + \epsilon \leq 0, \\ i \in \{1, 2, \dots, m'\};$$

$$107 \quad -h^i \leq 0 \quad \text{for all } i \text{ such that } -1 - u_i + \epsilon \geq 0, \\ i \in \{0, 1, \dots, k-1\};$$

$$108 \quad h^i \leq 0 \quad \text{for all } i \text{ such that } -1 + u_i + \epsilon \geq 0, \\ i \in \{0, 1, \dots, k-1\};$$

$$109 \quad |h^i| \leq 1, \quad i = 1, 2, \dots, k.$$

An efficient method for computing $\nabla f^0(z)$ has already been presented in Section 2.4, which can also be adapted in a very simple manner to computing $\nabla f^i(z)$, $i = 1, 2, \dots, m'$. Thus, to compute $\nabla f^0(z)$, we solve for p_i , $i = 1, 2, \dots, k$, the difference equation,

$$110 \quad p_i - p_{i+1} = \left(\frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \right)^T p_{i+1} - \left(\frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \right)^T, \\ i = 0, 1, \dots, k-1,$$

with $p_k = 0$. Then, as has been shown in (2.4.14),

$$111 \quad \frac{\partial f^0(z)}{\partial u_i} = \frac{\partial f_i^0(x_i(z), u_i)}{\partial u_i} - \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T p_{i+1}, \quad i = 0, 1, \dots, k-1,$$

i.e., (111) gives the i th* component of $\nabla f^0(z)$ (recall that $u_i \in \mathbb{R}^1$).

Next, we note that

$$112 \quad \nabla_z f^j(z) = \left(\frac{\partial x_k(z)}{\partial z} \right)^T \nabla_x q^j(x_k(z)), \quad j = 1, 2, \dots, m',$$

where we use ∇_z , ∇_x rather than ∇ to avoid possible confusion. Hence,

$$113 \quad \frac{\partial f^j(z)}{\partial u_i} = \left(\frac{\partial x_k(z)}{\partial u_i} \right)^T \nabla q^j(x_k(z)), \quad j = 1, 2, \dots, m'.$$

Now, from (2.4.9),

$$114 \quad \frac{\partial x_k(z)}{\partial u_i} = \Phi_{k,i+1} \frac{\partial f_i(x_i(z), u_i)}{\partial u_i}, \quad i = 0, 1, 2, \dots, k-1,$$

where the matrix $\Phi_{k,i+1}$ is determined by solving (2.4.7), i.e.,

$$115 \quad \Phi_{j+1,i+1} - \Phi_{j,i+1} = \frac{\partial f_j(x_j(z), u_j)}{\partial x_j} \Phi_{j,i+1}, \quad j = i+1, i+2, \dots, k-1,$$

with $\Phi_{i+1,i+1} = I$, the $\nu \times \nu$ identity matrix. Substituting into (113), we obtain,

$$116 \quad \frac{\partial f^j(z)}{\partial u_i} = \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T \Phi_{k,i+1}^T \nabla_x q^j(x_k(z)).$$

Now, for $j = 1, 2, \dots, m'$, we calculate the vectors $p_{i,j}$ by solving

$$117 \quad p_{i,j} - p_{i+1,j} = \left(\frac{\partial f_i(x_i(z), u_i)}{\partial x_i} \right)^T p_{i+1,j}, \quad i = 1, 2, \dots, k-1,$$

with $p_{k,j} = \nabla_x q^j(x_k(z))$. Then it is easy to see that

$$118 \quad \frac{\partial f^j(z)}{\partial u_i} = \left(\frac{\partial f_i(x_i(z), u_i)}{\partial u_i} \right)^T p_{i+1,j}, \quad j = 1, 2, \dots, m', \quad i = 0, 1, \dots, k-1.$$

Thus, (118) gives the i th component of $\nabla f^j(z)$.

* Note that because of the numbering of the control sequence u_0, u_1, \dots , we are assuming that the components of $\nabla f^j(z), j = 0, 1, 2, \dots, m'$, will be ranked in the order $0, 1, 2, \dots, k-1$, and not in the order $1, 2, \dots, k$, for the purpose of evaluation.

We now see that to evaluate the gradient vectors $\nabla f^i(z)$, $i = 1, 2, \dots, m'$, we never have to manipulate matrices bigger than $v \times v$ in dimension. This fact tends to simplify calculations considerably, assuming, of course, that k is very much larger than v , as is usually the case. ■

119 **Example.** Consider the continuous optimal control problem,

$$120 \quad \text{minimize} \quad \int_0^T f^0(x(t), u(t)) dt, \quad x(t) \in \mathbb{R}^v, \quad u(t) \in \mathbb{R}^1, \quad t \in [0, T],$$

subject to

$$121 \quad \frac{d}{dt} x(t) = f(x(t), u(t)), \quad t \in [0, T],$$

$$122 \quad x(0) = \hat{x}_0, \quad q^i(x(T)) \leq 0, \quad i = 1, 2, \dots, m$$

where all the functions are continuously differentiable in all of their arguments. In addition, let us suppose that $u(\cdot) \in L_\infty^1[0, T]$ and that $x(\cdot) \in C_v[0, T]$, the space of all continuous functions from $[0, T]$ into \mathbb{R}^v , with norm $\|x\|_C = \sup_{t \in [0, T]} \|\dot{x}(t)\|$. Furthermore, let us suppose that the gradient of

$$123 \quad F^0(u) = \int_0^T f^0(x(t; u), u(t)) dt$$

exists, where $x(t; u)$ is the solution of (121) at time t , corresponding to the control $u(\cdot)$ and satisfying $x(0; u) = \hat{x}_0$. (See Section 2.5 for a derivation of $\text{grad } F^0(u)(\cdot)$ and for some of the conditions which guarantee its existence.)

A small amount of experimentation reveals that the ZPP version of algorithm (20) or of algorithm (26) is the easiest one to adapt to the solution of problem (120)–(122). (The ZPP version of these algorithms is obtained by substituting step 3" in (75) for step 3 in the original algorithm.) Referring to theorem (75), we find we need the following additional assumption: Let $u(\cdot)$ be any control in $L_\infty^1[0, T]$ such that $q^i(x(T; u)) \leq 0$ for $i = 1, 2, \dots, m$; then we shall suppose that there exists a $\delta u(\cdot) \in L_\infty^1[0, T]$ (equate $\delta u(\cdot)$ with the vector h in (75)) such that

$$124 \quad \langle \nabla q^i(x(T; u)), \delta x(T; \delta u) \rangle < 0 \quad \text{for all } i \text{ such that } q^i(x(T; u)) = 0.*$$

Next, by examining the various derivations in Section 2.5, we conclude that the subproblem (70)–(72), which must be solved at each iteration in order to compute a feasible direction vector, becomes, for the problem (120)–(122) (with $\delta u(\cdot)$ taking the place of h),

$$125 \quad \text{minimize} \quad \int_0^T \left[\frac{\partial f^0(x(t; u), u)}{\partial x} \delta x(t) + \frac{\partial f^0(x(t; u), u)}{\partial u} \delta u(t) \right] dt,$$

* Where $\delta x(T; \delta u)$ is determined by (126), with $\delta x(0) = 0$.

subject to

- $$126 \quad \frac{d}{dt} \delta x(t) = \frac{\partial f(x(t; u), u)}{\partial x} \delta x(t) + \frac{\partial f(x(t; u), u)}{\partial u} \delta u(t), \quad t \in [0, T],$$
- $$127 \quad \delta x(0) = 0, \quad \langle \nabla q^i(x(T; u)), \delta x(T; u) \rangle \leq -\epsilon, \\ \text{for all } i \text{ such that } q^i(x(T; u)) + \epsilon \geq 0, \quad i \in \{1, 2, \dots, m\},$$
- $$128 \quad |\delta u(t)| \leq 1 \quad \text{for almost all } t \in [0, T]. \blacksquare$$

(Note that if we had wished, we could have eliminated (126) by solving explicitly for $\delta x(t)$, as was done in Section 2.5.) We now observe a major difference between continuous and discrete optimal control problems: Although some of the methods to be described in the next chapter are applicable to problem (125)–(128), there are no methods that can solve this problem in a finite number of iterations, at least in the general case. Thus, we get a conceptual algorithm only, i.e., one that requires us to compute an accumulation point of an infinite sequence at each iteration. We are therefore faced with two alternatives. The first is to introduce yet one more ϵ procedure into the algorithm in an attempt to make it conform to the model (1.3.33). The reader may wish to undertake this task as a research project, since the specifics of such a scheme are still to be published. The second alternative, and the one which at present seems to be the more practical one, is to discretize problem (120)–(122), as described in Section 1.1, and then solve the discretized version. In this manner, one separates the difficulties of integration of differential equations from the difficulties of optimization, and one is likely to obtain a reasonably efficient algorithm. If we used a heuristic rule to truncate the search for a $\delta u(\cdot)$ which is optimal for (125)–(128), we would be introducing errors into the optimization whose effect we do not know how to predict. This, coupled with the fact that a discretization of one kind or another cannot be avoided in any calculation on an electronic digital computer, leads one to the conclusion that one should discretize problem (120)–(122), and problems like it, before attempting to solve them.*

4.4 Second-Order Methods of Feasible Directions

In the preceding section, we have presented methods of feasible directions based on the function $h^0(\cdot)$ (2.18) (algorithm (3.5)), methods based on the function $h_\epsilon^0(\cdot)$ (3.8) (algorithms (3.20) and (3.26)), and methods based on

* The preceding discussion was based on the optimistic assumption that a sequence $\{\bar{u}(\cdot)\}$ constructed by a feasible directions method would converge. However, this is not very likely, as we have already pointed out in Section 2, due to the fact that a closed and bounded subset of a Hilbert space may not be compact.

the function $\hat{h}_\epsilon^0(\cdot)$ (3.73) (see theorem (3.75)). (The methods based on the use of the function $\tilde{h}_\epsilon^0(\cdot)$ (3.63) are really a special case of the methods based on the function $h_\epsilon^0(\cdot)$.) All of these are first-order methods, since the direction-finding process involves only the first derivative of the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, which define the problem to be solved, i.e., problem (3.1). We shall now show how the functions $h^0(\cdot)$, $h_\epsilon^0(\cdot)$ and $\hat{h}_\epsilon^0(\cdot)$ can be modified by the addition of second-derivative terms in order to obtain algorithms with a Newton-Raphson-like appearance, and hopefully, with a better rate of convergence than that of the methods discussed in the preceding section.

As in Section 3, we shall consider the problem,

$$1 \quad \min\{f^0(z) \mid f^i(z) \leq 0, i = 0, 1, 2, \dots, m\},$$

where the $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ are continuously differentiable functions. (At some point it will become necessary to assume that these functions are twice continuously differentiable.) As before, we shall denote by C the constraint set,

$$2 \quad C = \{z \in \mathbb{R}^n \mid f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

and, in addition to the indicator set $J_\epsilon(z)$, defined in (3.7), we shall make use of the indicator set $I_\epsilon(z)$, which we define, for any $\epsilon \geq 0$ and for any $z \in C$, by

$$3 \quad I_\epsilon(z) = \{i \mid f^i(z) + \epsilon \geq 0, i \in \{1, 2, \dots, m\}\},$$

so that $J_\epsilon(z) = \{0\} \cup I_\epsilon(z)$. We now define the functions $h'^0(\cdot)$, $h'_\epsilon^0(\cdot)$, $\hat{h}'_0(\cdot)$ and $\hat{h}''_0(\cdot)$, mapping C into \mathbb{R}^1 as follows:

$$4 \quad h'^0(z) = \min_{h \in S^*} \max\{\langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle; f^i(z) + \langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle, i = 1, 2, \dots, m\};$$

$$5 \quad h'_\epsilon^0(z) = \min_{h \in S^*} \max_{i \in J_\epsilon(z)} \{\langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle\};$$

$$6 \quad \hat{h}'_0(z) = \min_{h \in S^*} \{\langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \mid \langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle \leq -\epsilon, i \in I_\epsilon(z)\};$$

$$7 \quad \hat{h}''_0(z) = \min\{\langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \mid \langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle \leq -\epsilon, i \in I_\epsilon(z)\};$$

where the $H_i(z)$, $i = 0, 1, 2, \dots, m$, are positive semidefinite $n \times n$ matrices whose elements are continuous functions of z and, in (5)–(7), $\epsilon \geq 0$. The set S^* was defined in (3.9), i.e., $S^* = \{h \in \mathbb{R}^n \mid |h^i| \leq 1, i = 1, 2, \dots, n\}$.

8 Proposition. For any $z \in C$, $h'^0(z) = 0$ if and only if $h^0(z) = 0$.

Proof. Suppose that for some $z \in C$, $h'^0(z) < 0$. Then, by inspection, we must also have $h^0(z) < 0$, i.e., $h^0(z) = 0$ implies that $h'^0(z) = 0$.*

Next, suppose that

$$h^0(z) = \max\{\langle \nabla f^0(z), h' \rangle; f^i(z) + \langle \nabla f^i(z), h' \rangle, i = 1, 2, \dots, m\} < 0,$$

for some $h' \in S^*$. Then there exists a $\lambda' > 0$ such that (because the quadratic terms will be dominated by the linear and affine ones)

$$\begin{aligned} & \max\{\lambda \langle \nabla f^0(z), h' \rangle + \lambda^2 \langle h', H_0(z) h' \rangle; f^i(z) + \lambda \langle \nabla f^i(z), h' \rangle \\ & \quad + \lambda^2 \langle h', H_i(z) h' \rangle, i = 1, 2, \dots, m\} < 0 \quad \text{for all } \lambda \in (0, \lambda'] \end{aligned}$$

Hence, $h'^0(z) < 0$, which implies that if $h'(z) = 0$, then $h^0(z) = 0$. ■

9 Proposition. For any $z \in C$, $h'_0(z) = 0$ if and only if $h^0(z) = 0$. ■

10 Exercise. Prove proposition (9). [Hint: Repeat the arguments used in the proof of proposition (8).] ■

11 Proposition. Suppose that the assumptions of proposition (3.74) are satisfied; then, for any $z \in C$, $h''_0(z) = 0$ if and only if $h^0(z) = 0$. ■

12 Exercise. Prove proposition (11). [Hint: Proceed as for (8).] ■

13 Proposition. Suppose that the assumptions of propositions (3.74) are satisfied and that for all $z \in C$, the matrix $H_0(z)$ is positive definite. Then, for any $z \in C$, $h''_0(z) = 0$ if and only if $h^0(z) = 0$.

Proof. When $H_0(z)$ is positive definite, the set

$$\{h \in \mathbb{R}^n \mid \langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \leq \alpha\}$$

is compact for every $\alpha \in \mathbb{R}^1$. Hence, $\hat{h}''_0(\cdot)$ is well-defined for every $\epsilon \geq 0$. Now, suppose that for some $z \in C$, $\hat{h}''_0(z) < 0$; then, by inspection, we must also have $\hat{h}''_0(z) < 0$ (since $\hat{h}''_0(z) \leq \hat{h}'_0(z)$ for all $z \in C$). Consequently, $\hat{h}''_0(z) = 0$ implies that $\hat{h}'_0(z) = 0$.

* This is a proof by contraposition, as are those of the propositions to follow. They are based on the fact that if the falsehood of statement A implies the falsehood of statement B , then the truth of B implies the truth of A .

Next, suppose that for some $z \in C$, $\dot{h}_0^{(0)}(z) < 0$, with

$$\dot{h}_0''(z) = \langle \nabla f^0(z), h' \rangle + \langle h', H_0(z) h' \rangle.$$

Then, for all $\lambda \in (0, 1)$, we must have $\lambda \langle \nabla f^i(z), h' \rangle + \lambda^2 \langle h', H_i(z) h' \rangle < 0$, $i \in I_0(z)$, and hence, since for some λ in $(0, 1)$ we must also have $\lambda h' \in S^*$, we conclude that $\dot{h}_0'(z) < 0$, so that $\dot{h}_0'(z) = 0$ implies that $\ddot{h}_0''(z) = 0$. In view of proposition (11), we are obviously done. ■

The relationship between the zeros of $h_0^0(\cdot)$ and of $\hat{h}_0^0(\cdot)$ was established in proposition (3.74). To complete our demonstration of relationships between all the functions which we shall show to be usable in a method of feasible directions, we shall now work exercise (2.16) for the reader.

- 14 Proposition.** For any $z \in C$, $h^0(z) = 0$ if and only if $h_0^0(z) = 0$.

Proof. Suppose that $h^0(z) = \max\{\langle \nabla f^0(z), h' \rangle; f^i(z) + \langle \nabla f^i(z), h' \rangle, i = 1, 2, \dots, m\} < 0$ for some $h' \in S^*$. Then $h_0^0(z) \leq \max_{i \in J_0(z)} \langle \nabla f^i(z), h' \rangle < 0$ and $h_0^0(z) = 0$ implies that $h^0(z) = 0$.

Now suppose that for some $z \in C$, $h_0^0(z) = \max_{i \in J_0(z)} \langle \nabla f^i(z), h'' \rangle < 0$, for some $h'' \in S^*$. Then for some $\lambda'' \in (0, 1]$, we must have $\max\{\lambda'' \langle \nabla f^0(z), h'' \rangle; f^i(z) + \lambda'' \langle \nabla f^i(z), h'' \rangle, i = 1, 2, \dots, m\} < 0$. Consequently, $h^0(z) < 0$. We therefore conclude that $h^0(z) = 0$ implies $h_0^0(z) = 0$. ■

- 15 Exercise.** The functions defined in (4)–(7) do not exhaust all the possibilities of modifying the functions $h^0(\cdot)$, $h_\epsilon^0(\cdot)$, and $\hat{h}_\epsilon^0(\cdot)$ in such a way as to obtain a new function whose zeros coincide with those of the old function. Thus, consider the functions from C into \mathbb{R}^n defined by

$$16 \quad h'^0(z) = \min_{h \in S^*} \{ \langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \mid f^i(z) + \langle \nabla f^i(z), h \rangle \\ + \langle h, H_i(z) h \rangle \leq 0, i = 1, 2, \dots, m \},$$

$$17 \quad h''^0(z) = \min\{\langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \mid f^i(z) + \langle \nabla f^i(z), h \rangle \\ + \langle h, H_i(z) h \rangle \leq 0, i = 1, 2, \dots, m\},$$

where the $n \times n$ matrices $H_i(z)$, $i = 0, 1, 2, \dots, m$, are positive definite and are continuous in z . Show that $\dot{h}^0(z) = 0$ if and only if $\ddot{h}^0(z) = 0$. Show that $\dot{h}^0(z) = 0$ if and only if $\dot{h}'^0(z) = 0$. ■

The manner in which these functions can be used to modify algorithms (3.5), (3.20) and (3.26) should be obvious. For example, to substitute $h^0(\cdot)$ for $h(\cdot)$ in (3.5), we change the instruction in step 2 of (3.5) from “Solve (2.22)–(2.25) to obtain $(h^0(z), h(z))$ ” to “Solve (4.18)–(4.20) to obtain $(h^0(z), h(z))$,” where (4.18)–(4.20) are as stated below.

subject to

$$\begin{aligned} 19 \quad & -h^0 + \langle \nabla f^0(z), h \rangle + \langle h, H_0(z) h \rangle \leq 0, \\ & -h^i + f^i(z) + \langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle \leq 0, \quad i = 1, 2, \dots, m, \\ 20 \quad & |h^i| \leq 1, \quad i = 1, 2, \dots, n. \end{aligned}$$

Similarly, to substitute $h'_\epsilon(\cdot)$ for $h_\epsilon^0(\cdot)$ in algorithm (3.26), we change the instruction in step 3 of (3.26) from “Compute a vector $\bar{h}_{\epsilon_j}(z) = (h_{\epsilon_j}^0(z), h_{\epsilon_j}(z))$ by solving (10)–(12) for $\epsilon = \epsilon_j$ ” to “Compute a vector $\bar{h}_{\epsilon_j}(z) = (h_{\epsilon_j}^0(z), h_{\epsilon_j}(z))$ by solving (4.21)–(4.23) for $\epsilon = \epsilon_j$,” where (4.21)–(4.23) are as stated below.

$$\begin{aligned} 21 \quad & \text{minimize} \quad h^0, \\ \text{subject to} \quad & \\ 22 \quad & -h^0 + \langle \nabla f^i(z), h \rangle + \langle h, H_i(z) h \rangle \leq 0, \quad \text{for all } i \in J_\epsilon(z); \\ 23 \quad & |h^i| \leq 1, \quad i = 1, 2, \dots, n. \end{aligned}$$

Note that the problems (18)–(20), (21)–(23), as well as those defined by (6), (7) and (16), (17), require us to minimize either a linear or a quadratic form on a constraint set defined by linear and quadratic inequalities. So far, there are no methods for solving such problems in a finite number of iterations. Consequently, these functions are not usable in an implementable algorithm of the form of (3.26) unless we set $H_i(z) \equiv 0$ for all i appearing in the constraints in the feasible direction-finding subproblem. These considerations seem to rule out an immediate interest in the functions $h'^0(\cdot)$ and $h'^0(\cdot)$, and suggest that we set $H_i(z) \equiv 0$ for $i = 1, 2, \dots, m$ in (6), (7), and (16), (17). To give an algorithm using the functions $\hat{h}'^0(\cdot)$, $\hat{h}''^0(\cdot)$, $\hat{h}^0(\cdot)$, or $\hat{h}''^0(\cdot)$ a Newton-Raphson-like appearance, we would set $H_0(z) = \partial^2 f^0(z)/\partial z^2$ for all $z \in C$, and $H_i(z) = 0$ for $i = 1, 2, \dots, m$.*

- 24 **Theorem.** Suppose that in the instruction of step 2 of algorithm (3.5), the words “Solve (2.22)–(2.25)” are replaced by the words “Solve (4.18)–(4.20).” Then the conclusions of theorem (2.32) remain valid for this modification of algorithm (3.5). ■
- 25 **Exercise.** Prove theorem (24). [Hint: Proceed essentially as in the proof of theorem (2.32).] ■
- 26 **Theorem.** Suppose that in the instruction of step 3 of algorithm (3.26), the words “by solving (10)–(12)” are replaced by the words “by solving (4.21)–(4.23).” Then the conclusions of theorem (3.35) remain valid for this modification of algorithm (3.26). ■

* Provided, of course, that $H_0(z)$ is positive semidefinite.

- 27 Exercise.** Prove theorem (26). [*Hint:* Proceed essentially as in the proof of theorem (3.35).] ■
- 28 Remark.** Theorems (24) and (26) only make sense when the algorithms in question are well-defined, i.e., only when they are applied to problems in which they will not jam up for some reason, such as the nonexistence of a $\mu(z)$. An assumption such as the requirement that the set $\{z \mid f^0(z) \leq f^0(z_0)\}$ be compact takes care of this difficulty provided one starts at z_0 . Also see lemma (3.27). ■
- 29 Theorem.** Suppose that for every $z \in C$ there exists an $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \in I_0(z)$. If in the instruction of step 3 of algorithm (3.26), the words “by solving (10)–(12)” are replaced by the words “by solving (4.6),” then the conclusions of theorem (3.35) remain valid. ■
- 30 Exercise.** Prove theorem (29). [*Hint:* Make use of theorem (26) and proceed essentially as in the proof of theorem (3.75).] ■
- 31 Theorem.** Suppose that for every $z \in C$ there exists an $h \in \mathbb{R}^n$ such that $\langle \nabla f^i(z), h \rangle < 0$ for all $i \in I_0(z)$, and that the matrix $H_0(z)$ is positive definite for all $z \in C$. If in the instructions of step 3 of algorithm (3.26), the words “by solving (10)–(12)” are replaced by the words “by solving (4.7),” then the conclusions of theorem (3.35) remain valid. ■
- 32 Exercise.** Prove theorem (31). [*Hint:* Proceed as in (30), showing in addition, that the vectors $h(z)$ computed by the algorithm for $\|z - z'\| \leq \beta$ are contained in a compact set, where $\beta > 0$ is arbitrary and $z' \in C$ is not desirable.] ■

The interested reader may also construct similar theorems involving the functions defined in (16) and (17). This concludes our discussion of methods of feasible directions.

4.5 Gradient Projection Methods

We conclude this chapter with two gradient projection methods for solving the problem,

$$1 \quad \min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m\},$$

under the assumption that the $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$, $i = 0, 1, \dots, m$, are continuously differentiable convex functions. As before, we shall denote by C the constraint set, i.e.,

$$2 \quad C = \{z \in \mathbb{R}^n \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}.$$

Since we assume that the functions $f^i(\cdot)$, $i = 1, 2, \dots, m$, are convex, it is clear that the set C must also be convex.

Let z be any point in C . We define the projection of $z - \nabla f^0(z)$ onto C to be the point $z_p \in C$ which satisfies

$$3 \quad \|z_p - (z - \nabla f^0(z))\| = \min\{\|z' - (z - \nabla f^0(z))\| \mid z' \in C\}.$$

- 4 **Exercise.** Show that z_p exists and is unique because C is closed. Also, show that

$$5 \quad \langle z' - z_p, z - \nabla f^0(z) - z_p \rangle \leq 0 \quad \text{for all } z' \in C. \blacksquare$$

- 6 **Proposition.** A point $\hat{z} \in C$ is optimal for (1) if and only if the projection \hat{z}_p of $\hat{z} - \nabla f^0(\hat{z})$ onto C satisfies $\hat{z} = \hat{z}_p$.

Proof. Suppose that $\hat{z}_p = \hat{z}$. Then the two convex sets C and $\{z \mid f^0(z) \leq f^0(\hat{z})\}$ must be separated by the hyperplane $\{z \mid \langle z - \hat{z}, \nabla f^0(\hat{z}) \rangle = 0\}$. Consequently, \hat{z} must be optimal for (1).

Conversely, suppose that \hat{z} is optimal for (1), but $\hat{z}_p \neq \hat{z}$. Then, setting $z' = \hat{z}$ in (5), we obtain,

$$7 \quad \langle \hat{z} - \hat{z}_p, \hat{z} - \hat{z}_p - \nabla f^0(\hat{z}) \rangle \leq 0.$$

Setting $h = \hat{z} - \hat{z}_p$, we now obtain from (7) that

$$8 \quad -\langle \nabla f^0(\hat{z}), h \rangle \leq -\|h\|^2 < 0,$$

since $h \neq 0$. Consequently, there must exist a point $z' = \hat{z} + \lambda h$, for some $\lambda \in (0, 1]$ such that $z' \in C$, because C is convex, and $f^0(z') < f^0(\hat{z})$, because of (8) and the fact that the first derivatives of $f^0(\cdot)$ are continuous. Since this contradicts the optimality of \hat{z} , we must have $\hat{z}_p = \hat{z}$. ■

Proposition (6) together with the relationship (8) provide the basis for gradient projection methods, most of which evolve from the following simple scheme: Given a point z_i in C , (i) compute π , the projection of the half-line $\{z \mid z = z_i - \lambda \nabla f^0(z_i), \lambda \geq 0\}$ onto C , and (ii) set z_{i+1} to be a point on π which minimizes $f^0(z)$ over $z \in \pi$. From (8) and the convexity of C , we see that if z_i is not optimal for C , $z_{i+1} \neq z_i$ and $f^0(z_{i+1}) < f^0(z_i)$. If z_i is optimal, then we conclude from proposition (6) that $z_{i+1} = z_i$.

- 9 **Exercise.** Consider the conceptual algorithm defined by the operations (i) and (ii) above. Suppose that $\{z_i\}$ is a sequence constructed by this algorithm. Show that if $\{z_i\}$ is finite (stopping when $z_{i+1} = z_i$), then its last element is optimal for (1), and if it is infinite, then every accumulation point of $\{z_i\}$ is optimal for (1). [Hint: Proceed as in the proof of theorem (2.1.6).] ■

In the algorithms that we shall consider, the projection will not be onto the set C , but onto a tangent linear manifold passing through z_i , and the projection operation will be carried out by means of suitably constructed projection matrices. We therefore digress for a moment to establish a few of the properties of projection matrices, which we shall need later on.

Suppose that we are given the vectors f_i , $i = 1, 2, \dots, m$, in \mathbb{R}^n . Let I be any subset of $\{1, 2, \dots, m\}$ containing m' elements. Then we shall denote by F_I the $n \times m'$ matrix whose columns are the vectors f_i , $i \in I$, ordered linearly on i . To record this definition in compact form, we write,

$$10 \quad F_I = (f_i)_{i \in I}.$$

Let L_I be the linear subspace spanned by the vectors f_i , $i \in I$. Then any vector $z \in L_I$ can be expressed in the form,

$$11 \quad z = F_I y,$$

for some $y \in \mathbb{R}^{m'}$. Now, given any $z \in \mathbb{R}^n$, the projection of z onto L_I is the unique solution of the problem,

$$12 \quad \min\{\|z - z'\|^2 \mid z' \in L_I\}.$$

Because of (11), (12) becomes

$$13 \quad \min\{\|z - F_I y'\|^2 \mid y' \in \mathbb{R}^{m'}\}.$$

Computing the gradient of $\|z - F_I y'\|^2$ with respect to y' and setting it equal to zero, we conclude that if z_p is the projection of z onto L_I , then $z_p = F_I y_p$ for a $y_p \in \mathbb{R}^{m'}$ which satisfies

$$14 \quad F_I^T(z - F_I y_p) = 0.$$

If we now assume that the f_i , $i \in I$, are linearly independent, then we find that

$$15 \quad z_p = F_I y_p = F_I (F_I^T F_I)^{-1} F_I^T z.$$

We now define P_I to be the matrix which projects \mathbb{R}^n onto the subspace L_I , i.e., given any vector $z \in \mathbb{R}^n$, $P_I z \in L_I$ and is the projection of z on L_I . Assuming that the vectors f_i , $i \in I$, which define the subspace L_I are linearly independent, we conclude from (15) that

$$16 \quad P_I = F_I (F_I^T F_I)^{-1} F_I^T.$$

When the vectors f_i , $i \in I$, are linearly dependent, there must exist a subset $I' \subset I$ such that the vectors f_i , $i \in I'$, are linearly independent and

span L_I , i.e., $L_{I'} = L_I$. In that case, the matrix P_I is identical with the matrix $P_{I'}$ which is obtained by substituting I' for I in (16). Thus, the projection matrix P_I always exists.

By inspection of (16), P_I is symmetric and positive semidefinite. In addition, it has the following properties: (i) for any $z \in L_I$, $P_I z = z$ (as can be seen from (15) by setting $z = F_I y$), and hence,

$$17 \quad P_I P_I = P_I;$$

(ii) for any $z \in \mathbb{R}^n$ which is orthogonal to L_I , i.e., for any $z \in \mathbb{R}^n$ satisfying $\langle f_i, z \rangle = 0$ for all $i \in I$, $P_I z = 0$ (since $\langle f_i, z \rangle = 0$ for all $i \in I$ implies that $F_I^T z = 0$ in (15)).

Now consider the matrix,

$$18 \quad P_I^\perp = I - P_I.$$

Suppose that $z \in L_I$; then we obtain $P_I^\perp z = z - z = 0$. Next, suppose that z is orthogonal to L_I ; then $P_I^\perp z = z - 0 = z$. Consequently,

$$19 \quad P_I^\perp P_I^\perp = P_I^\perp, \quad P_I^\perp P_I = P_I P_I^\perp = 0.$$

We conclude that P_I^\perp is also a projection matrix and that it projects \mathbb{R}^n onto the orthogonal complement of L_I , which we shall denote by L_I^\perp .

Now let z be any vector in \mathbb{R}^n ; then $z = z' + z''$, with $z' \in L_I$ and $z'' \in L_I^\perp$, and hence,

$$20 \quad (P_I + P_I^\perp) z = P_I z' + P_I^\perp z'' = z' + z'' = z,$$

i.e., $P_I + P_I^\perp$ is the $n \times n$ identity matrix, and any vector $z \in \mathbb{R}^n$ can be decomposed into a sum of orthogonal components as follows:

$$21 \quad z = P_I z + P_I^\perp z,$$

where, by (15),

$$22 \quad P_I z = F_I y,$$

with

$$23 \quad y = (F_I^T F_I)^{-1} F_I^T z.$$

We may therefore rewrite (21) as follows:

$$24 \quad z = F_I y + P_I^\perp z,$$

where y is as defined in (23).

Now suppose that $j \in I$, and let $I' = \{i \mid i \in I, i \neq j\}$. To simplify notation, we shall write

$$25 \quad I' = I - j \triangleq \{i \mid i \in I, i \neq j\}.$$

Denoting by $F_{I'}$ the $n \times (m' - 1)$ matrix whose columns are f_i , $i \in I'$, ordered linearly on i ; by $L_{I'}$, $L_{I'}^\perp$ the subspace spanned by the f_i , $i \in I'$, and its orthogonal complement, respectively; and by $P_{I'}$, $P_{I'}^\perp$ the matrices which project \mathbb{R}^n onto $L_{I'}$, $L_{I'}^\perp$, respectively, we find the following important results: Since $L_{I'} \subset L_I$, we have $L_{I'}^\perp \supset L_I^\perp$, and hence,

$$26 \quad P_{I-j}^\perp P_I^\perp = P_I^\perp, \quad P_I P_{I-j} = P_{I-j}.$$

We now state our final result. Let $z \in \mathbb{R}^n$ be arbitrary; then, by (24),

$$27 \quad z = F_I y + P_I^\perp z = \sum_{i \in I} \mu^i f_i + P_I^\perp z,$$

where $y = (y^1, y^2, \dots, y^{m'})$ is given by (23) and μ^i is the component of y which multiplies the column f_i of F when the operation $F_I y$ is expanded in columns, as in (27), i.e., $y = (\mu^{i_1}, \mu^{i_2}, \dots, \mu^{i_{m'}})$, with $i_j \in I$ for $j = 1, 2, \dots, m'$. This concludes our digression.

We begin with a gradient projection method for solving problem (1) when the set C is a convex polytope with interior, i.e., when the functions $f^i(\cdot)$, $i = 1, 2, \dots, m$, are affine, with

$$28 \quad f^i(z) = \langle f_i, z \rangle - b^i, \quad i = 1, 2, \dots, m,$$

where $f_i \in \mathbb{R}^n$ and $b^i \in \mathbb{R}^1$, for $i = 1, 2, \dots, m$. We continue to assume that the function $f^0(\cdot)$ is convex and continuously differentiable.

The method we are about to present is a modification of Rosen's gradient projection method [R1]. This modification was constructed by Polak [P1] and consists of an abridged version of the Rosen method together with an " ϵ procedure" which was added to the Rosen method to make it convergent. As in the preceding section, we shall make use of the ϵ -active constraints indicator set $I_\epsilon(z)$, which was defined in (4.3), for any $\epsilon \geq 0$ and any $z \in C$, as follows:

$$29 \quad I_\epsilon(z) = \{i \mid f^i(z) + \epsilon \geq 0, i \in \{1, 2, \dots, m\}\}.$$

When the functions $f^i(\cdot)$, $i = 1, 2, \dots, m$, are affine and as in (28), (29) becomes, for any $\epsilon \geq 0$ and for any $z \in C$,

$$30 \quad I_\epsilon(z) = \{i \mid \langle f_i, z \rangle - b^i + \epsilon \geq 0, i \in \{1, 2, \dots, m\}\}.$$

31 Proposition. Consider the problem,

$$32 \quad \min\{f^0(z) \mid \langle f_i, z \rangle - b^i \leq 0, i = 1, 2, \dots, m\},$$

where $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a convex, continuously differentiable function, $f_i \in \mathbb{R}^n$, $b^i \in \mathbb{R}^1$, $i = 1, 2, \dots, m$. A point $\hat{z} \in C = \{z \mid \langle f^i, z \rangle - b^i \leq 0, i = 1, 2, \dots, m\}$ is optimal for (32) if and only if

$$33 \quad \nabla f^0(\hat{z}) = F_{I_0(z)} y_0(\hat{z}), \quad \text{i.e., } P_{I_0(z)}^\perp \nabla f^0(\hat{z}) = 0, *$$

and

$$34 \quad y_0(\hat{z}) \leq 0,$$

where $F_{I_0(z)}$ is defined according to (10), and $P_{I_0(z)}^\perp$ is defined according to (18).

Proof. Proposition (31) is simply a special case of (1.2.1) combined with (1.2.18) into a statement of necessary and sufficient conditions of optimality for the problem (1.1.1). ■

To ensure that for every $\epsilon \geq 0$ and for every $z \in C$ that we shall be using, the matrix which projects \mathbb{R}^n onto $L_{I_\epsilon(z)}$ (the subspace spanned by the vectors $f_i, i \in I_\epsilon(z)$) is given by formula (16), we make the following hypothesis:

35 Assumption. We shall suppose that there exists an $\epsilon' > 0$ such that for every $z \in C$ and for every $\epsilon \in [0, \epsilon']$, the vectors $f_i, i \in I_\epsilon(z)$, are linearly independent. ■

This assumption can be removed at the expense of an increase in the complexity of the gradient projection algorithm and we shall indicate later how this can be done.

In the algorithm below, we shall use the notation introduced in (10), (16) and (18), setting $F_I = 0$, $P_I = 0$ when I is empty, and in addition, for every $\epsilon \in [0, \epsilon']$ and for every $z \in C$, we define

$$36 \quad y_\epsilon(z) = (F_{I_\epsilon(z)}^T F_{I_\epsilon(z)})^{-1} F_{I_\epsilon(z)} \nabla f^0(z).$$

37 Algorithm (gradient projection method for (32), Polak [P1]).

Step 0. Compute a $z_0 \in C$; select a $\beta \in (0, 1)$, and $\bar{\epsilon} \in (0, \epsilon')$, and an $\epsilon'' \in (0, \bar{\epsilon}]$; set $i = 0$.[†]

Comment. See (2.30) for a “bootstrap” method for computing a z_0 . It is common to set $\beta = \frac{1}{2}$.

Step 1. Set $z = z_i$.

Step 2. Set $\epsilon_0 = \bar{\epsilon}$ and set $j = 0$.

* When $I_0(z)$ is empty, we set $F_{I_0(z)} = 0$, $P_{I_0(z)} = 0$.

† The $\epsilon' > 0$ is assumed to be such that (35) holds.

Step 3. Compute the vector,

$$38 \quad h_{\epsilon_j}(z) = P_{I_{\epsilon_j}(z)}^\perp \nabla f^0(z).$$

Step 4. If $\|h_{\epsilon_j}(z)\|^2 > \epsilon_j$, set $h(z) = -h_{\epsilon_j}(z)$, set $\epsilon(z) = \epsilon_j$, and go to step 12; else, go to step 5.

Comment. Do not store the value of $\epsilon(z)$. It is introduced here only because it will be needed in the convergence proofs to follow.

Step 5. If $\epsilon_j \leq \epsilon''$, compute the vector $h_0(z)$ (as in (38), with ϵ_j replaced by 0), compute the vector $y_0(z)$ (according to (36)), and go to step 6; else, go to step 7.

Step 6. If $\|h_0(z)\|^2 = 0$ and $y_0(z) \leq 0$, set $z_{i+1} = z$, and stop; else, go to step 7.

Step 7. Compute the vector $y_{\epsilon_j}(z)$ (according to (36)).

Step 8. If $y_{\epsilon_j}(z) \leq 0$, set $\epsilon_{j+1} = \beta \epsilon_j$, set $j = j + 1$, and go to step 3; else, go to step 9.

Step 9. Assuming that $I_{\epsilon_j}(z) = \{k_1, k_2, \dots, k_m\}$ and that $k_1 < k_2 < \dots < k_{m'}$, set $\mu_{\epsilon_j}^{k_\alpha}(z) = y_{\epsilon_j}^\alpha(z)$ for $\alpha = 1, 2, \dots, m'$ (where $y_{\epsilon_j}^\alpha(z)$ is the α th component of the vector $y_{\epsilon_j}(z)$).

Step 10. Find the smallest $k \in I_{\epsilon_j}(z)$ such that the vector

$$39 \quad \bar{h}_{\epsilon_j}(z) = P_{I_{\epsilon_j}(z)-k}^\perp \nabla f^0(z)$$

satisfies the relation

$$40 \quad \|\bar{h}_{\epsilon_j}(z)\| = \max\{\|P_{I_{\epsilon_j}(z)-l}^\perp \nabla f^0(z)\| \mid l \in I_{\epsilon_j}(z), \mu_{\epsilon_j}^l(z) > 0\},$$

and set $h(z) = -\bar{h}_{\epsilon_j}(z)$.

Step 11. If $\|h(z)\|^2 \leq \epsilon_j$, set $\epsilon_{j+1} = \beta \epsilon_j$, set $j = j + 1$, and go to step 3; else, set $\epsilon(z) = \epsilon_j$, and go to step 12.

Step 12. Compute $\lambda(z) > 0$ to be the smallest scalar satisfying

$$41 \quad f^0(z + \lambda(z) h(z)) = \min\{f^0(z + \lambda h(z)) \mid \lambda \geq 0, (z + \lambda h(z)) \in C\}.$$

Step 13. Set $z_{i+1} = z + \lambda(z) h(z)$, set $i = i + 1$, and go to step 1. ■

Without any doubt, algorithm (37) is the most complex algorithm that we have encountered so far. Any modification of this algorithm designed to remove assumption (35) or to substitute an implementable step size rule for the purely conceptual one in (41) would obviously result in further complexity. However, such modifications are not difficult to conceive and to introduce once algorithm (37) is thoroughly understood.

The understanding of algorithm (37) becomes easier if we first examine it

stripped of the ϵ procedure. In that case, (37) reduces to the following, “stripped” algorithm which is obtained from (37) by setting $\epsilon = 0$, so that $\epsilon_j = 0$ always:

42 Algorithm (elementary gradient projection method).

Step 0. Compute a $z_0 \in C$ and set $i = 0$.

Step 1. Set $z = z_i$.

Step 2. Compute $h_0(z)$ (according to (38) with $\epsilon_j = 0$).

Step 3. If $\|h_0(z)\| > 0$, set $h(z) = -h_0(z)$ and go to step 6; else, compute $y_0(z)$ (according to (36)) and go to step 4.

Step 4. If $y_0(z) \leq 0$, set $z_{i+1} = z$ and stop; else, renumber the components of $y_0(z)$ to obtain the μ_0^k , as in step 9 of (37).

Step 5. Compute $\bar{h}_0(z)$ (according to (39), (40) with $\epsilon_j = 0$), and set $h(z) = -\bar{h}_0(z)$.

Step 6. Compute $\lambda(z)$ as in step 12 of (37) (see (41)).

Step 7. Set $z_{i+1} = z + \lambda(z) h(z)$, set $i = i + 1$, and go to step 1. ■

Now let us examine this stripped version of (37). Suppose that in step 3 we find that $h_0(z) = 0$; then we go to step 4 and if $y_0(z) \leq 0$, we stop. But by (31), z is optimal if $h_0(z) = 0$ and $y_0(z) \leq 0$. Hence, our stopping rule is properly chosen. Next, suppose that $h_0(z) \neq 0$. Then $h(z) = -h_0(z)$ and (making use of (20) and (19))

$$\begin{aligned} 43 \quad \langle \nabla f^0(z), h(z) \rangle &= -\langle (P_{I_0(z)} + P_{I_0(z)}^\perp) \nabla f^0(z), P_{I_0(z)}^\perp \nabla f^0(z) \rangle \\ &= -\|P_{I_0(z)}^\perp \nabla f^0(z)\|^2 \\ &= -\|h(z)\|^2 < 0. \end{aligned}$$

Next, for all $j \in I_0(z)$,

$$44 \quad \langle f_j, h(z) \rangle = \langle f_j, P_{I_0(z)}^\perp \nabla f^0(z) \rangle = \langle P_{I_0(z)}^\perp f_j, \nabla f^0(z) \rangle = 0,$$

and hence, for some $\lambda' > 0$, the linear segment $\{z' = z + \lambda h(z) \mid \lambda \in [0, \lambda']\}$ is contained in the edge of the polytope* C , which is formed by the intersection of the hyperplanes $\{z' \mid \langle f_j, z' \rangle - b_j = 0\}$, $j \in I_0(z)$. This edge of C cannot be of zero length, for if it were, we would have obtained $h_0(z) = 0$. (An edge of zero length is a vertex, in which case, $I_0(z)$ must contain exactly n indices, which results in $P_{I_0(z)}^\perp = 0$.) Consequently, because of (43), there must exist a point z_{i+1} on this linear segment contained in C such that $f^0(z_{i+1}) < f^0(z) = f^0(z_i)$. Thus, when $h_0(z) \neq 0$, the algorithm finds an improved feasible point.

Now suppose that $h_0(z) = 0$, and that $y_0(z) \not\leq 0$. This means that either z

* A polytope is a hyperpolyhedron. A convex polytope is the intersection of a finite number of closed half-spaces.

is a vertex of C or that $\nabla f^0(z)$ is orthogonal to the edge of C which consists of the intersection of the hyperplanes $\{z' \mid \langle f_j, z' \rangle - b^j = 0\}$, $j \in I_0(z)$, or both. Then we construct $\bar{h}_0(z)$ by projecting $\nabla f^0(z)$ on a larger subspace than the one used for computing $h_0(z)$, and obtained by removing the index k from $I_0(z)$, i.e., by projecting $\nabla f^0(z)$ onto $L_{I_0(z)-k}$ instead of $L_{I_0(z)}$. Setting $h(z) = -\bar{h}_0(z)$, we now find that

$$\begin{aligned} 45 \quad \langle \nabla f^0(z), h(z) \rangle &= -\langle \nabla f^0(z), P_{I_0(z)-k}^\perp \nabla f^0(z) \rangle \\ &= -\|P_{I_0(z)-k}^\perp \nabla f^0(z)\|^2. \end{aligned}$$

Now, since $P_{I_0(z)}^\perp \nabla f^0(z) = 0$, it follows from (27) that

$$46 \quad \nabla f^0(z) = P_{I_0(z)} \nabla f^0(z) = \sum_{j \in I_0(z)} \mu_0^j(z) f_j.$$

Hence,

$$47 \quad P_{I_0(z)-k}^\perp \nabla f^0(z) = \mu_0^k(z) P_{I_0(z)-k}^\perp f_k \neq 0,$$

since $\mu_0^k(z) > 0$ and $f_k \notin L_{I_0(z)-k}$, because the vectors f_j , $j \in I_0(z)$, are linearly independent by assumption (35). Consequently,

$$48 \quad \langle \nabla f^0(z), h(z) \rangle < 0.$$

Thus we see that $h(z)$ defines a direction along which the cost can be made to decrease. We must still show that $h(z)$ defines a feasible direction, i.e., that for some $\lambda' > 0$, the linear segment $\{z' = z + \lambda h(z) \mid \lambda \in [0, \lambda']\} \subset C$. To show this, we only need to show that

$$49 \quad \langle h(z), f_j \rangle \leq 0 \quad \text{for all } j \in I_0(z).$$

Now, making use of (47),

$$\begin{aligned} 50 \quad \langle h(z), f_j \rangle &= -\mu_0^k(z) \langle P_{I_0(z)-k}^\perp f_k, f_j \rangle \\ &= -\mu_0^k(z) \langle f_k, P_{I_0(z)-k}^\perp f_j \rangle. \end{aligned}$$

Hence,

$$51 \quad \langle h(z), f_j \rangle = 0 \quad \text{for all } j \in I_0(z), \quad j \neq k,$$

$$52 \quad \langle h(z), f_k \rangle = -\mu_0^k(z) \|P_{I_0(z)-k}^\perp f_k\|^2 < 0,$$

which shows that (49) is satisfied and also explains why in defining $\bar{h}_0(z)$ we used only those $j \in I_0(z)$ for which $\mu_0^j(z) > 0$, rather than all $j \in I_0(z)$ for which $\mu_0^j \neq 0$.

Algorithm (42) may construct a sequence of points $\{z_i\}$ which converges to a point z^* satisfying $P_{I_0(z^*)}^\perp \nabla f^0(z^*) = 0$, but not $y_0(z^*) \leqq 0$, i.e., to a nonoptimal z^* . The effect of the ϵ procedure, which was added to (42) so as to produce (37), is to keep track of the inactive constraints, very much in the same manner as in the feasible directions algorithm (3.26). (Rosen's original method [R1] contained one more feature than (42), which went some way toward preventing the sequence $\{z_i\}$ from converging to a non-optimal point.)

- 53 Exercise.** Show that there is a way of computing the matrix $P_{I_\epsilon(z)-j}$ by making use of the matrix $P_{I_\epsilon(z)}$. Also show that if $I_\epsilon(z_{i+1}) = I_\epsilon(z_i) - j + k$, where $j \in I_\epsilon(z_i)$ and $k \in \bar{I}_\epsilon(z_i)$ (the complement of $I_\epsilon(z_i)$ in $\{1, 2, \dots, m\}$), then $P_{I_\epsilon(z_{i+1})}^\perp$ can be computed by making use of the matrix $P_{I_\epsilon(z_i)}^\perp$. Show that these computations are simpler than a direct evaluation by formula (18). [Hint: Look up [R1] if unable to do this from first principles.] ■
- 54 Lemma.** Given any $z_i \in C$, algorithm (37) cannot cycle indefinitely between steps 3 and 11 while constructing an infinite sequence $\epsilon_j \rightarrow 0$.

Proof. Suppose that $z = z_i$ is optimal, i.e., suppose that $h_0(z) = 0$ and $y_0(z) \leqq 0$ (see (38), (36) and (31)). Then

$$55 \quad \nabla f^0(z) = \sum_{p \in I_0(z)} \mu_0{}^p(z) f_p \in L_{I_0(z)},$$

where $L_{I_0(z)}$ is the subspace of \mathbb{R}^n spanned by the f_p , $p \in I_0(z)$. Now, for any $\epsilon \geqq 0$, by (3.16), $I_\epsilon(z) \supset I_0(z)$ and hence $L_{I_\epsilon(z)} \supset L_{I_0(z)}$. Consequently, $\nabla f^0(z) \in L_{I_\epsilon(z)}$ for all $\epsilon \geqq 0$, and hence,

$$56 \quad \nabla f^0(z) = \sum_{p \in I_\epsilon(z)} \mu_\epsilon{}^p(z) f_p.$$

Assuming that $\epsilon \in [0, \bar{\epsilon}]$, the vectors f_p , $p \in I_\epsilon(z)$, are linearly independent, and hence we must have for all $\epsilon \in [0, \bar{\epsilon}]$,

$$57 \quad \mu_\epsilon{}^p(z) = \mu_0{}^p(z) \leqq 0 \quad \text{for all } p \in I_0(z),$$

$$58 \quad \mu_\epsilon{}^p(z) = 0 \quad \text{for all } p \in I_\epsilon(z), \quad p \notin I_0(z).$$

In addition, (56) implies that $h_\epsilon(z) = 0$ for all $\epsilon \in [0, \bar{\epsilon}]$.

Let $j' \geqq 0$ be an integer such that $\beta^{j'} \bar{\epsilon} \leqq \epsilon''$. Then for some $j \leqq j'$, algorithm (37) will determine in step 4 that $\|h_\epsilon(z)\|^2 = 0 \leqq \epsilon_j$, it will then pass on to step 5, where it will find that $\epsilon_j \leqq \epsilon''$, upon which it will compute $h_0(z) = 0$, $y_0(z) \leqq 0$ and it will stop.

Next, suppose that the point $z = z_i$ is not optimal. Then we must have either $h_0(z) \neq 0$, or else $h_0(z) = 0$ and $y_0(z) \not\leqq 0$. Suppose that $h_0(z) \neq 0$.

Then there exists an integer $j' \geq 0$ such that $\|h_0(z)\|^2 > \beta^{j'}\bar{\epsilon}$. Also, by (3.18), there exists a $\rho > 0$ such that $I_{\epsilon_j}(z) = I_0(z)$ for all $\epsilon_j \in [0, \rho]$. Therefore, for all $j \geq j''$, where $\beta^{j''}\bar{\epsilon} \leq \rho$, $I_{\epsilon_j}(z) = I_0(z)$, and hence, $h_{\epsilon_j}(z) = h_0(z)$, for all $j \geq j''$. Consequently, for some $j \leq \max\{j', j''\}$, algorithm (37) will determine in step 4 that $\|h_{\epsilon_j}(z)\|^2 = \|h_0(z)\|^2 > \epsilon_j$ and will proceed to construct a new point z_{i+1} in steps 12 and 13.

Now suppose that $h_0(z) = 0$ and that $y_0(z) \not\leq 0$. Then, again by (3.18), there exists a $\rho > 0$ such that $I_{\epsilon_j}(z) = I_0(z)$ for all $\epsilon_j \in [0, \rho]$, i.e., for all $j \geq j''$, where $\beta^{j''}\bar{\epsilon} \leq \rho$. Consequently, for all $j \geq j''$, $h_{\epsilon_j}(z) = \bar{h}_0(z) \neq 0$. Let $j' \geq 0$ be an integer such that $\|\bar{h}_0(z)\|^2 > \beta^{j'}\bar{\epsilon}$. Then, in step 11, for some $j \leq \max\{j', j''\}$, algorithm (37) will find that $\|\bar{h}_{\epsilon_j}(z)\|^2 > \epsilon_j$ and will proceed to construct a new point z_{i+1} in steps 12 and 13. Thus algorithm (37) cannot construct an infinite sequence $\{\epsilon_j\}$ which converges to zero. ■

Just as in the case of the method of feasible directions (3.26), despite the fact that algorithm (37) contains an ϵ procedure, it is easy to identify it with the model (1.3.9), or to be more precise, with the particular case of that model, the model (1.3.2). However, there seems to be no obvious way of identifying algorithm (37) with the model (1.3.33), because $\|h(z_i)\|^2$ cannot be shown to converge to zero monotonically, and hence $\|h(\cdot)\|^2$ cannot be identified with the function $c(\cdot)$ in (1.3.33).

To identify algorithm (37) with the model (1.3.2), we set $T = C$, $c(\cdot) = f^0(\cdot)$, and we define $a : C \rightarrow C$ by the instructions in steps 1–13 of algorithm (37), i.e.,

$$59 \quad a(z) = z \quad \text{if } z \in C \text{ is optimal for (32),}$$

$$60 \quad a(z) = z + \lambda(z) h(z) \quad \text{if } z \in C \text{ is not optimal for (32).}$$

Note that since $\epsilon(z)$ is uniquely determined by the point z , the vector $h(z)$ and the scalar $\lambda(z)$, if it exists, are both uniquely determined by z , so that $a(\cdot)$ is indeed a map from C into C . To conclude the identification of algorithm (37) with the model (1.3.2), we define $z \in C$ to be desirable if it is optimal for the problem (32). Obviously, if $\lambda(z)$ is not well-defined for all $z \in \{z \mid f^0(z) \leq f^0(z_0), \langle f_i, z \rangle - b^i \leq 0, i = 1, 2, \dots, m\} \triangleq C'(z_0)$, then algorithm (37) cannot be applied to the problem. A sufficient condition which ensures that $\lambda(z)$ will exist for all $z \in C'(z_0)$ is that $C'(z_0)$ is compact.

- 61 **Theorem.** Suppose that problem (32) is such that the function $a(\cdot)$ is well-defined by (59), (60) for all $z \in C'(z_0) = \{z \in C \mid f^0(z) \leq f^0(z_0)\}$. If $\{z_i\}$ is a sequence in C constructed by algorithm (37), then either $\{z_i\}$ is finite and its last element is optimal for (32), or $\{z_i\}$ is infinite and every accumulation point of $\{z_i\}$ is optimal for (32). (When $f^0(\cdot)$ is strictly convex, problem (32) has a unique optimal solution \hat{z} , and in such a case we shall have $z_i \rightarrow \hat{z}$.)

Proof. Since construction of the sequence can stop only because of the command in step 6 of (37), i.e., only if an optimal point has been found, the case of a finite sequence $\{z_i\}$ is trivial. To establish the second part of the theorem, we only need to show that assumptions (i) and (ii) of theorem (1.3.3) are satisfied by the maps $c(\cdot) = f^0(\cdot)$ and $a(\cdot)$ defined by (59), (60). Since $f^0(\cdot)$ is continuous by assumption, (i) is obviously satisfied, and hence we are left with showing that (ii) of theorem (1.3.3) holds for algorithm (37). Thus, we must show that if $z \in C$ is not optimal for (32), then there exist an $\epsilon > 0$ and a $\delta < 0$ such that

$$62 \quad f^0(z' + \lambda(z') h(z')) - f^0(z') \leq \delta \quad \text{for all } z' \in B(z, \epsilon)$$

(where $B(z, \epsilon) = \{z' \in C \mid \|z - z'\| \leq \epsilon\}$).

Therefore, suppose that $z \in C$ is not optimal. Then, according to the instructions in steps 4 and 11, algorithm (37) computes an $\epsilon(z) > 0$ such that $\|h(z)\|^2 > \epsilon(z)$. Since $h(z)$ is defined either by (38), or by (39) and (40), for $\epsilon_j = \epsilon(z)$, we must have, either

$$63 \quad \|h_{\epsilon(z)}(z)\|^2 > \epsilon(z),$$

or else,

$$64 \quad \|\bar{h}_{\epsilon(z)}(z)\|^2 > \epsilon(z).$$

We shall now construct a $B(z, \bar{\rho}) = \{z' \in C \mid \|z' - z\| \leq \bar{\rho}\}$, $\bar{\rho} > 0$, such that for all $z' \in B(z, \bar{\rho})$, $\epsilon(z')$ is bounded from below by a strictly positive number. Thus, suppose that $h(z) = -h_{\epsilon(z)}(z)$. Then (63) must hold, and since $\nabla f^0(\cdot)$ is continuous by assumption, there exists a $\rho' > 0$ such that

$$65 \quad \|P_{I_{\epsilon(z)}(z)}^\perp \nabla f^0(z')\|^2 > \beta \epsilon(z) \quad \text{for all } z' \in B(z, \rho').$$

Now, by (3.19), there exists a $\rho'' > 0$ such that $I_{\epsilon(z)}(z') \subset I_{\epsilon(z)}(z)$ for all $z' \in B(z, \rho'')$. Let $\rho = \min\{\rho', \rho''\}$, and, as before, let $L_{I_\alpha(z)}$, $L_{I_\alpha(z)}^\perp$ denote the subspace spanned by the vectors f_i , $i \in I_\alpha(z)$, and its orthogonal complement, respectively. Then, for every $z' \in B(z, \rho)$ and for every $\alpha \in [0, \epsilon(z)]$ (since $I_\alpha(z') \subset I_{\epsilon(z)}(z')$ by (3.16)), $L_{I_\alpha(z')}^\perp \supset L_{I_{\epsilon(z)}(z')}^\perp$, and therefore,

$$66 \quad \|P_{I_\alpha(z')}^\perp \nabla f^0(z')\|^2 \geq \|P_{I_{\epsilon(z)}(z')}^\perp \nabla f^0(z')\|^2 \geq \|P_{I_{\epsilon(z)}(z)}^\perp \nabla f^0(z')\|^2 > \beta \epsilon(z).$$

We therefore conclude that if (63) took place, then for all $z' \in B(z, \rho)$, algorithm (37) will set $\epsilon(z') \geq \beta \epsilon(z)$.

Now suppose that (64) took place, i.e., that $h(z) = -\bar{h}_{\epsilon(z)}(z)$. Then either

$h_{\epsilon(z)}(z) = P_{I_{\epsilon(z)}(z)}^\perp \nabla f^0(z)$ is zero, or not. Suppose that $h_{\epsilon(z)}(z) \neq 0$. Then, for some integer $j' \geq 0$, $\|h_{\epsilon(z)}(z)\|^2 > \beta^{j'} \bar{\epsilon} = \delta'$. Let $\rho'' > 0$ be defined as above, i.e., $I_{\epsilon(z)}(z') \subset I_{\epsilon(z)}(z)$ for all $z' \in B(z, \rho'')$. Then, since by (3.16), $I_\alpha(z') \subset I_{\epsilon(z)}(z')$ for all $\alpha \in [0, \epsilon(z)]$ and $\nabla f^0(\cdot)$ is continuous, there exists a $\tilde{\rho} \in (0, \rho'']$ such that for every $z \in B(z, \tilde{\rho})$ and for every $\alpha \in [0, \epsilon(z)]$,

$$67 \quad \|P_{I_\alpha(z')}^\perp \nabla f^0(z')\|^2 \geq \|P_{I_{\epsilon(z)}(z')}^\perp \nabla f^0(z')\|^2 \geq \|P_{I_{\epsilon(z)}(z)}^\perp \nabla f^0(z')\|^2 > \beta\delta'.$$

Now, for $\alpha \in [0, \epsilon(z)]$, and for any $k \in I_\alpha(z')$, $z' \in B(z, \tilde{\rho})$, we have

$$68 \quad \begin{aligned} P_{I_\alpha(z')-k}^\perp \nabla f^0(z') &= P_{I_\alpha(z')-k}^\perp [P_{I_\alpha(z')} \nabla f^0(z') + P_{I_\alpha(z')}^\perp \nabla f^0(z')] \\ &= P_{I_\alpha(z')-k}^\perp \left(\sum_{i \in I_\alpha(z')} \mu_\alpha^i f_i + P_{I_\alpha(z')}^\perp \nabla f^0(z') \right) \\ &= \mu_\alpha^k(z') P_{I_\alpha(z')-k}^\perp f_k + P_{I_\alpha(z')}^\perp \nabla f^0(z'), \end{aligned}$$

where we have made use of (20), (26) and (27). Note that because of (26) the two components of the last sum in (68) must be orthogonal, and hence we have, because of (67),

$$69 \quad \|P_{I_\alpha(z')-k}^\perp \nabla f^0(z')\|^2 = (\mu_\alpha^k(z'))^2 \|P_{I_\alpha(z')-k}^\perp f_k\|^2 + \|P_{I_\alpha(z')}^\perp \nabla f^0(z')\|^2 > \beta\delta'$$

for all $z' \in B(z, \tilde{\rho})$, for all $\alpha \in [0, \epsilon(z)]$, for all $k \in I_\alpha(z')$. Consequently, for every $z' \in B(z, \tilde{\rho})$, algorithm (37) will set $\epsilon(z') \geq \beta\delta'$, either because (67) applies to the construction of $h(z')$, or because (69) applies to the construction of $h(z')$. (By inspection, $\epsilon(z) \geq \delta'$.)

Finally, suppose that $h_{\epsilon(z)}(z) = 0$. Then,

$$70 \quad \nabla f^0(z) = \sum_{i \in I_{\epsilon(z)}} \mu_{\epsilon(z)}^i(z) f_i.$$

Now let

$$71 \quad \delta_1 = \min\{\|P_I^\perp \nabla f^0(z)\|^2 \mid I \subset I_{\epsilon(z)}(z), \|P_I^\perp \nabla f^0(z)\|^2 > 0\},$$

and let

$$72 \quad \delta_2 = \min \left\{ \max_{i \in I, \mu_i^i > 0} \|P_{I-i}^\perp \nabla f^0(z)\|^2 \mid I \subset I_{\epsilon(z)}(z), \|P_I^\perp \nabla f^0(z)\|^2 = 0 \right\},$$

where $P_I \nabla f^0(z) = \sum_{i \in I} \mu_i^i f_i$. Then, obviously, $\delta_1 > 0$ and $\delta_2 > 0$. Let j'' be a positive integer such that $\delta'' = \beta^{j''} \bar{\epsilon} \leq \min\{\delta_1, \delta_2\}$, and again let $\rho'' > 0$ be such that $I_{\epsilon(z)}(z') \subset I_{\epsilon(z)}(z)$ for all $B(z, \rho'')$. Then there exists a

$\hat{\rho} \in (0, \rho'']$ such that for every $z' \in B(z, \hat{\rho})$ and for every $\alpha \in [0, \epsilon(z)]$, $I_\alpha(z') \subset I_{\epsilon(z)}(z') \subset I_{\epsilon(z)}(z)$,

$$73 \quad \|P_{I_\alpha(z')}^\perp \nabla f^0(z')\|^2 > \beta\delta'', \quad \text{if } \|P_{I_\alpha(z')}^\perp \nabla f^0(z)\|^2 > 0,$$

and, defining $\mu_{I_\alpha(z')}^i$ by $P_{I_\alpha(z')} \nabla f^0(z) = \sum_{i \in I_\alpha(z')} \mu_{I_\alpha(z')}^i f_i$, for $i \in I_\alpha(z')$,

$$74 \quad \mu_\alpha^i(z') > 0, \quad \text{if } \mu_{I_\alpha(z')}^i > 0, \quad i \in I_\alpha(z'), \quad \text{and } \|P_{I_\alpha(z')}^\perp \nabla f^0(z)\|^2 = 0,$$

$$75 \quad \max\{\|P_{I_\alpha(z')-k}^\perp \nabla f^0(z')\|^2 \mid k \in I_\alpha(z'), \mu_\alpha^k(z') > 0\} > \beta\delta'', \\ \text{if } \|P_{I_\alpha(z')}^\perp \nabla f^0(z)\|^2 = 0.$$

That a $\hat{\rho} > 0$ exists for which (73) is true follows from the fact that $\|P_{I_\alpha(z')}^\perp \nabla f^0(z)\|^2 \geq \delta_1 \geq \delta''$, and from the continuity of the functions $\|P_{I_\alpha(z')}^\perp \nabla f^0(\cdot)\|^2$. That a $\hat{\rho} > 0$ exists for which (74) is true follows from the continuity of the functions $(F_{I_\alpha(z')}^\top F_{I_\alpha(z')})^{-1} F_{I_\alpha(z')} \nabla f^0(\cdot)$. Finally, that there exists a $\hat{\rho} > 0$ for which (75) is true follows from the continuity of the functions $\|P_{I_\alpha(z')-k}^\perp \nabla f^0(\cdot)\|^2$, in conjunction with (74) which ensures that if the index k' is such that

$$76 \quad \max_{\substack{i \in I_\alpha(z'), \\ \mu_{I_\alpha(z')}^i > 0}} \|P_{I_\alpha(z')-i}^\perp \nabla f^0(z)\|^2 = \|P_{I_\alpha(z')-k'}^\perp \nabla f^0(z)\|^2 \geq \delta_2 > \delta'',$$

then this index is also considered in the maximization in (75). In all of the above, we depend strongly on the fact that the set $I_{\epsilon(z)}(z)$ has only a finite number of subsets and hence that the worst case can be found by considering these subsets one after the other to determine a $\hat{\rho} > 0$.

Upon examining (73) and (75), we find that if $h_{\epsilon(z)}(z) = 0$, then algorithm (37) will set $\epsilon(z') \geq \beta\delta''$ for all $z' \in B(z, \hat{\rho})$. Since we have now exhausted all possibilities, we conclude that irrespective of whether $h(z)$ is determined according to the instruction in step 3 (i.e., by (38)), or according to the instruction in step 10 (i.e., by (39), (40)), we can find a $\hat{\rho} > 0$ and an $\hat{\epsilon} > 0$ such that for all $z' \in B(z, \hat{\rho})$, algorithm (37) will set $\epsilon(z') \geq \hat{\epsilon} > 0$.

Consider any $z' \in B(z, \hat{\rho})$. If $h(z') = -h_{\epsilon(z')}(z')$, then for every $i \in I_{\epsilon(z')}(z')$,

$$77 \quad \begin{aligned} -\langle f_i, h(z') \rangle &= \langle f_i, P_{I_{\epsilon(z')}(z')}^\perp \nabla f^0(z') \rangle \\ &= \langle P_{I_{\epsilon(z')}(z')}^\perp f_i, \nabla f^0(z') \rangle \\ &= 0. \end{aligned}$$

If $h(z') = -h_{\epsilon(z')}(z')$, then for every $i \in (I_{\epsilon(z')}(z') - k)$, where k is the integer

computed in step 10 to satisfy (39), (40) for $\epsilon_j = \epsilon(z')$, we have $\langle f_i, h(z') \rangle = 0$, which we obtain from a calculation entirely analogous to (77). In addition,

$$\begin{aligned} 78 \quad & \langle f_k, h(z') \rangle = - \langle f_k, P_{I_{\epsilon(z')}^{\perp}(z')-k}^{\perp} \nabla f^0(z') \rangle \\ &= - \left\langle f_k, P_{I_{\epsilon(z')}^{\perp}(z')-k}^{\perp} \left[\sum_{i \in I_{\epsilon(z')}^{\perp}(z')} \mu_{\epsilon(z')}^i(z') f_i + P_{I_{\epsilon(z')}^{\perp}(z')}^{\perp} \nabla f^0(z') \right] \right\rangle \\ &= - \langle f_k, \mu_{\epsilon(z')}^k(z') P_{I_{\epsilon(z')}^{\perp}(z')-k}^{\perp} f_k \rangle \\ &= - \mu_{\epsilon(z')}^k(z') \| P_{I_{\epsilon(z')}^{\perp}(z')-k}^{\perp} f_k \|^2 < 0, \end{aligned}$$

since $\mu_{\epsilon(z')}^k(z') > 0$ by construction and f_k does not belong to $L_{I_{\epsilon(z')}^{\perp}(z')}$. Hence, for every $i \in I_{\epsilon(z')}^{\perp}(z')$, $\langle f_i, h(z') \rangle \leq 0$, and therefore,

$$79 \quad \langle f_i, z' + \lambda h(z') \rangle - b^i \leq 0, \quad \text{for all } i \in I_{\epsilon(z')}^{\perp}(z'), \quad \text{for all } \lambda \geq 0.$$

Now, since $\nabla f^0(\cdot)$ is continuous and $B(z, \bar{\rho})$ is compact, there exists a constant $M > 0$ such that

$$80 \quad \| h(z') \| \leq \| \nabla f^0(z') \| \leq M \quad \text{for all } z' \in B(z, \bar{\rho}).$$

The first part of the inequality in (80) holds because the projection of a vector is always, in norm, smaller than or equal to the vector itself. Suppose that k is such that $\| f_k \| = \max \{ \| f_i \| \mid i \in \{1, 2, \dots, m\} \}$. Then we see that

$$81 \quad \langle f_i, z' + \lambda h(z') \rangle - b^i \leq 0, \quad \text{for } i = 1, 2, \dots, m, \quad z' \in B(z, \bar{\rho}),$$

will certainly be satisfied for all $\lambda \in [0, \bar{\lambda}_m]$, where $\bar{\lambda}_m = \hat{\epsilon}/M \| f_k \|$, since $\epsilon(z') \geq \hat{\epsilon}$.

It now follows from theorem (B.3.7) that there exists an $\epsilon \in (0, \bar{\rho}]$ and a $\lambda_m \in (0, \bar{\lambda}_m]$ such that

$$82 \quad |\langle \nabla f^0(z' + \lambda h(z')), h(z') \rangle - \langle \nabla f^0(z'), h(z') \rangle| \leq \frac{\hat{\epsilon}}{2} \quad \text{for all } z' \in B(z, \epsilon), \quad \text{for all } \lambda \in [0, \lambda_m].$$

Now, by construction, $\langle \nabla f^0(z'), h(z') \rangle = -\| h(z') \|^2 < -\epsilon(z') \leq -\hat{\epsilon}$. Hence, making use of the mean-value theorem (B.1.1), for any $z' \in B(z, \epsilon)$,

$$83 \quad f^0(z' + \lambda_m h(z')) - f^0(z') = \lambda_m \langle \nabla f^0(z' + \lambda h(z')), h(z') \rangle \leq \frac{-\lambda_m \hat{\epsilon}}{2},$$

since $\lambda \in [0, \lambda_m]$. But, by construction, we must have for any $z' \in B(z, \epsilon)$,

$$84 \quad f^0(z' + \lambda(z') h(z')) - f^0(z') \leq f^0(z' + \lambda_m h(z')) - f^0(z') \leq \frac{-\lambda_m \hat{\epsilon}}{2},$$

Setting $\delta = -\lambda_m \bar{\epsilon}/2$, we now see that assumption (ii) of theorem (1.3.10) is satisfied, and we are done. ■

- 85 Remark.** It may be computationally inefficient to reset j to zero and ϵ_0 to the fixed value $\bar{\epsilon}$, as required in step 2 of algorithm (37), at each iteration. We have two options in dealing with this possible source of difficulty. The first option is to set $\bar{\epsilon} = \epsilon(z)$ every so many iterations. The second option is to set $\bar{\epsilon} = \epsilon(z)$ at each iteration. The latter results in an algorithm analogous to (3.20), while (37) is analogous to (3.26). Both of these options lead to modifications of algorithm (37) for which theorem (61) remains valid. ■

- 86 Exercise.** Show that if step 13 of algorithm (37) is replaced by step 13' or by step 13'' below, the conclusions of theorem (61) remain valid, provided $\lambda(z)$ is well-defined at each iteration.

Step 13'. Set $z_{i+1} = z + \lambda(z) h(z)$, set $\bar{\epsilon} = \epsilon(z)$, set $i = i + 1$, and go to step 1.

Step 13''. If i/k' is a strictly positive integer, set $z_{i+1} = z + \lambda(z) h(z)$, set $\bar{\epsilon} = \epsilon(z)$, set $i = i + 1$, and go to step 1; else, set $z_{i+1} = z + \lambda(z) h(z)$, set $i = i + 1$, and go to step 1.

Comment. $k' \geq 1$ is an integer which must be preselected to govern the rate of decrease of $\bar{\epsilon}$. ■

- 87 Exercise.** Show how algorithm (37) can be modified to accomodate the case where the vectors f_i , $i \in I_0(z)$ are not linearly independent for all $z \in C$. [Hint: Consider only the case where C has an interior. If the vectors f_k , $k \in I_{\epsilon}(z)$ are not linearly independent, use a suitably chosen subset of $I_{\epsilon}(z)$ in determining $h(z)$.] ■

- 88 Exercise.** Show that the step length rules described in propositions (3.49) and (3.53) for the method of feasible directions (3.26), can also be used in algorithm (37), instead of (41), to produce an implementable version of that algorithm. ■

Since the decrease in the cost, $f^0(z_{i+1}) - f^0(z_i)$, is related to the quantity $\langle \nabla f^0(z_i), h(z_i) \rangle = -\|h(z_i)\|^2$, one may attempt to accelerate the convergence of algorithm (37) by modifying it in such a way as to make $\|h(z_i)\|^2$ as large as possible at each iteration. The acceleration procedure given below can be shown not to affect the conclusions of theorem (61) for the resulting algorithm. Since the acceleration procedure requires a minor amount of rearrangement in algorithm (37), we shall state the entire resulting algorithm rather than the acceleration procedure only.

89 **Algorithm** (accelerated version of algorithm (37)).

Step 0. Compute a $z_0 \in C$, select a $\beta \in (0, 1)$, an $\bar{\epsilon} \in (0, \epsilon')$, and an $\epsilon'' \in (0, \bar{\epsilon})$; set $i = 0$.

Step 1. Set $z = z_i$.

Step 2. Set $\epsilon_0 = \bar{\epsilon}$ and set $j = 0$.

Step 3. Compute $h_{\epsilon_j}(z)$ according to (38).

Step 4. If $\|h_{\epsilon_j}(z)\|^2 > \epsilon_j$, go to step 5; else, go to step 8.

Step 5. Compute $y_{\epsilon_j}(z)$ according to (36).

Step 6. If $y_{\epsilon_j}(z) \leq 0$, set $h(z) = -h_{\epsilon_j}(z)$ and go to step 14; else, compute $\bar{h}_{\epsilon_j}(z)$ according to (39), (40) and go to step 7.

Step 7. Set $h(z) = -\bar{h}_{\epsilon_j}(z)$ and go to step 14.

Step 8. If $\epsilon_j \leq \epsilon''$, compute $h_0(z)$ and $y_0(z)$ and go to step 9; else, go to step 10.

Step 9. If $\|h_0(z)\|^2 = 0$ and $y_0 \leq 0$, set $z_{i+1} = z$, and stop; else, go to step 10.

Step 10. Compute $y_{\epsilon_j}(z)$.

Step 11. If $y_{\epsilon_j}(z) \leq 0$, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3; else, go to step 12.

Step 12. Compute $\bar{h}_{\epsilon_j}(z)$ according to (39), (40), and set $h(z) = -\bar{h}_{\epsilon_j}(z)$.

Step 13. If $\|h(z)\|^2 \leq \epsilon_j$, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3; else, go to step 14.

Step 14. Compute $\lambda(z)$ to be the smallest $\lambda \in (0, \infty)$ satisfying (41).

Step 15. Set $z_{i+1} = z + \lambda(z) h(z)$, set $i = i + 1$, and go to step 1. ■

90 **Remark.** Rosen's original method [R1], though not always convergent, has a very nice computational feature. It attempts to choose as successive z_i , points which are adjacent vertices of the polytope C . Whenever this is possible, the successive computation of the projection matrices used to compute $h(z_i)$ is considerably simplified. A detailed discussion of this aspect of Rosen's method is somewhat beyond the scope of this text. The reader interested in writing a digital computer program should therefore consult [R1] before proceeding with this task. In addition, if the specific method for computing projection matrices described in [R1] will be used, then it may be better to use step 7' below, instead of step 7 in algorithm (89). (Step 7' is based on Rosen's original algorithm.)

Step 7'. If $\|\bar{h}_{\epsilon_j}(z)\|^2 \geq 2 \|h_{\epsilon_j}(z)\|^2$, set $h(z) = -\bar{h}_{\epsilon_j}(z)$ and go to step 14; else, set $h(z) = -h_{\epsilon_j}(z)$ and go to step 14.

Thus, step 7', effectively, says: Accelerate by increasing $\|h(z)\|$ only if this increase is going to be sufficiently big to offset the additional work needed in computing the next projection matrix. ■

This concludes our discussion of gradient projection methods for the problem (32), which, as we recall, is a special case of problem (1). We shall

now present an algorithm which is a cross between the gradient projection method (37) and the feasible directions method (3.26), and which is applicable to the general case of the convex programming problem (1). The algorithm that we are about to describe is due to Polak [P1] and was obtained by adding an ϵ procedure to an algorithm described by Kalfond *et al.* [K1]. The ϵ procedure was added so as to make the resulting algorithm demonstrably convergent (The version in [K1] is not.)

- 91 **Assumption.** For the rest of this section, we shall suppose that the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, are all convex. ■

In addition, we shall need the equivalent of assumption (35), which we state below.

- 92 **Assumption.** We shall suppose that there exists an $\epsilon' > 0$ such that for any $z \in C$ and any $\epsilon \in [0, \epsilon']$, the vectors $\nabla f^i(z)$, $i \in I_\epsilon(z)$, are linearly independent. ■

We retain the notation introduced earlier in this section with the following, rather obvious, adaptation: For any $\epsilon \in [0, \epsilon']$ and any $z \in C$, we shall denote by

$$93 \quad F_{I_\epsilon(z)} = (\nabla f^i(z))_{i \in I_\epsilon(z)}$$

a matrix whose columns are the vectors $\nabla f^i(z)$, $i \in I_\epsilon(z)$, ordered linearly on i . The projection matrices $P_{I_\epsilon(z)}$ and $P_{I_\epsilon(z)}^\perp$ will still be defined by (16) and (18), but with the matrix $F_{I_\epsilon(z)}$ now defined by (93).

- 94 **Algorithm** (hybrid gradient projection method, Polak [P1]).

Comment. The first eleven steps of this algorithm are the same as in (37).

Step 0. Compute a $z_0 \in C$; select a $\beta \in (0, 1)$, and $\bar{\epsilon} \in (0, \epsilon']$, and an $\epsilon'' \in (0, \bar{\epsilon})$; set $i = 0$.

Comment. See (2.30) for a method of computing a z_0 . It is common to set $\beta = \frac{1}{2}$.

Step 1. Set $z = z_i$.

Step 2. Set $\epsilon_0 = \bar{\epsilon}$ and set $j = 0$.

Step 3. Compute the vector

$$95 \quad h_{\epsilon_j}(z) = P_{I_{\epsilon_j}(z)}^\perp \nabla f^0(z).$$

Step 4. If $\|h_{\epsilon_j}(z)\|^2 > \epsilon_j$, set $h(z) = -h_{\epsilon_j}(z)$, set $\epsilon(z) = \epsilon_j$, and go to step 12; else, go to step 5.

Comment. Do not store $\epsilon(z)$, it is only introduced for the sake of the proofs to follow.

Step 5. If $\epsilon_j \leq \epsilon''$, compute the vector $h_0(z)$ (using formula (95) with ϵ_j replaced by 0), compute the vector $y_0(z)$ (using (36)), and go to step 6; else, go to step 7.

Step 6. If $\|h_0(z)\|^2 = 0$ and $y_0(z) \leq 0$, set $z_{i+1} = z$, and stop (z is optimal); else, go to step 7.

Step 7. Compute the vector $y_{\epsilon_j}(z)$ (using formula (36)).

Step 8. If $y_{\epsilon_j}(z) \leq 0$, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3; else, go to step 9.

Step 9. Assuming that $I_{\epsilon_j}(z) = \{k_1, k_2, \dots, k_m\}$ and that $k_1 < k_2 < \dots < k_m$, set $\mu_{\epsilon_j}^{k_\alpha}(z) = y_{\epsilon_j}^\alpha(z)$ for $\alpha = 1, 2, \dots, m'$.

Step 10. Find the smallest $k \in I_{\epsilon_j}(z)$ such that the vector

$$96 \quad \bar{h}_{\epsilon_j}(z) = P_{I_{\epsilon_j}(z)-k}^\perp \nabla f^0(z)$$

satisfies the relation

$$97 \quad \|\bar{h}_{\epsilon_j}(z)\| = \max\{\|P_{I_{\epsilon_j}(z)-l}^\perp \nabla f^0(z)\| \mid l \in I_{\epsilon_j}(z), \mu_{\epsilon_j}^l(z) > 0\};$$

set $h(z) = -\bar{h}_{\epsilon_j}(z)$.

Step 11. If $\|h(z)\|^2 \leq \epsilon_j$, set $\epsilon_{j+1} = \beta\epsilon_j$, set $j = j + 1$, and go to step 3; else, set $\epsilon(z) = \epsilon_j$, and go to step 12.

Comment. Generally, the half-line $\{z' = z + \lambda h(z) \mid \lambda \geq 0\}$ intersects the set C at the point z only, i.e., generally, $h(z)$ does not define a feasible direction. We now construct a vector $v(z)$ which does define a feasible direction.

Step 12. If $h(z) = -h_{\epsilon(z)}(z)$, set $K_{\epsilon(z)}(z) = I_{\epsilon(z)}(z)$, and go to step 13; else, set $K_{\epsilon(z)}(z) = I_{\epsilon(z)}(z) - k$ and go to step 13.

Step 13. Compute the vector

$$98 \quad v(z) = \rho(z) h(z) + F_{K_{\epsilon(z)}(z)}(F_{K_{\epsilon(z)}(z)}^T F_{K_{\epsilon(z)}(z)})^{-1} t,$$

where $t = -\epsilon(z)(1, 1, \dots, 1) \in \mathbb{R}^n$ and $\rho(z) \geq 1$ is the smallest scalar in $[1, \infty)$ such that

$$99 \quad \langle \nabla f^l(z), v(z) \rangle \leq -\epsilon(z),$$

for $l = 0$ when $K_{\epsilon(z)}(z) = I_{\epsilon(z)}(z)$, and for $l = 0, k$, when $K_{\epsilon(z)}(z) = I_{\epsilon(z)}(z) - k$.

Step 14. Compute $\lambda(z) > 0$ to be the smallest scalar satisfying

$$100 \quad f^0(z + \lambda(z)v(z)) = \min\{f^0(z + \lambda v(z)) \mid \lambda \geq 0, z + \lambda v(z) \in C\}.$$

Step 15. Set $z_{i+1} = z + \lambda(z)v(z)$, set $i = i + 1$, and go to step 1. ■

- 101 **Theorem.** Suppose that problem (1) is such that algorithm (94) is well-defined for all $z \in \{z \in C \mid f^0(z) \leq f^0(z_0)\}$, where z_0 is as determined in step 0

of (94). If $\{z_i\}$ is a sequence in C constructed by algorithm (94), then either $\{z_i\}$ is finite and its last element is optimal for (1), or else $\{z_i\}$ is infinite and every accumulation point of $\{z_i\}$ is optimal for (1).

Proof. To prove this theorem, we need to carry out the same kind of reasoning as was used in proving theorems (3.35) and (61). Since it would be tedious to reproduce much of the detail that appears in the proofs of those theorems, we shall assume that the reader is familiar with their proofs, and we shall omit details which can be directly deduced from the arguments used therein.

To prove theorem (101), we identify algorithm (94) with the model (1.3.2) by setting $T = C$, $c(\cdot) = f^0(\cdot)$, defining $a : C \rightarrow C$ by the instructions in algorithm (94), and defining $\hat{z} \in C$ to be desirable if \hat{z} is optimal for (1). Next, since an obvious adaptation of proposition (31) is valid for problem (1), we see that the case of $\{z_i\}$ being finite is trivially true.

Hence, to prove theorem (101), we only need to show that assumption (ii) of theorem (1.3.3) is satisfied by algorithm (94). (We do not have to worry about assumption (i) of (1.3.3) because $f^0(\cdot)$ is assumed to be continuously differentiable.) Thus, given any nonoptimal $z \in C$, we only need to exhibit the existence of an $\epsilon > 0$ and of a $\delta < 0$ such that

$$102 \quad f^0(z' + \lambda(z') v(z')) - f^0(z') \leq \delta \quad \text{for all } z' \in B(z, \epsilon).$$

First, proceeding exactly as in the proof of theorem (61), and, in addition, making use of the fact that the functions $f^i(\cdot)$, $i = 0, 1, \dots, m$, are continuously differentiable, we can show that if $z \in C$ is not optimal for (1), then there exists a $\bar{\rho} > 0$ and an $\hat{\epsilon} > 0$ such that algorithm (94) will set $\epsilon(z') \geq \hat{\epsilon}$ for all $z' \in B(z, \bar{\rho})$. Next, by construction in step 13 (see (99)),

$$103 \quad \langle \nabla f^0(z'), v(z') \rangle \leq -\epsilon(z') \leq -\hat{\epsilon};$$

$$104 \quad \langle \nabla f^l(z'), v(z') \rangle \leq -\epsilon(z') \leq -\hat{\epsilon} \quad \text{for } l \in I_{(z')}(z'), \quad l \notin K_{\epsilon(z')}(z');$$

$$105 \quad \langle \nabla f^l(z'), v(z') \rangle = \rho(z) \langle \nabla f^l(z'), h(z') \rangle$$

$$\begin{aligned} &+ \langle \nabla f^l(z'), F_{K_{\epsilon(z')}(z')} (F_{K_{\epsilon(z')}(z')}^T F_{K_{\epsilon(z')}(z')})^{-1} t \rangle \\ &= \langle (F_{K_{\epsilon(z')}}^T F_{K_{\epsilon(z')}(z')})^{-1} F_{K_{\epsilon(z')}}^T \nabla f^l(z'), t \rangle \\ &= \langle e_1, t \rangle = -\epsilon(z) \leq -\hat{\epsilon} \quad \text{for all } l \in K_{\epsilon(z')}(z'); \end{aligned}$$

where $e_1 = (1, 0, 0, \dots, 0)^T$, is the first column of the $n \times n$ unit matrix.

Finally, since the functions $f^i(\cdot)$, $i = 0, 1, 2, \dots, m$, are continuously differentiable, there exists an $M \in (0, \infty)$ such that $\|v(z')\| \leq M$ for all $z' \in B(z, \bar{\rho})$. Consequently, making use of theorems (B.3.1) and (B.3.7), and of the mean-value theorem (B.1.1), just as we did in the proof of theorem

(3.35), we conclude that there exists an $\epsilon \in (0, \bar{\rho})$ and a $\delta < 0$ such that (102) is satisfied. ■

Algorithm (94) can be modified to an implementable form in exactly the same manner as algorithm (37), i.e., by using the step size rules described in (3.49) and (3.53), for example, instead of the step size rule defined by (41) and used in step 14 of (94). It can also be accelerated by replacing its first eleven steps by the first thirteen steps of algorithm (89). The further modifications indicated in remark (90) can also be utilized.

Applications to Optimal Control

Gradient projection methods can, obviously, be used for solving discrete optimal control problems also. In applying gradient projection methods to such problems, we calculate derivatives in exactly the same manner as in the case of feasible directions, and the reader is therefore referred to Section 3 for details (in particular, see formulas (3.110)–(3.118)).

The application of gradient projection methods to continuous optimal control problems is somewhat less straightforward, and we shall therefore illustrate how this is done by means of an example. Thus, consider the problem,

$$106 \quad \text{minimize} \quad \int_{t_0}^{t_1} \frac{1}{2} [\langle x(t), R(t) x(t) \rangle + \langle u(t), Q(t) u(t) \rangle] dt,$$

subject to

$$107 \quad \frac{d}{dt} x(t) = A(t) x(t) + B(t) u(t), \quad t \in [t_0, t_1],$$

$$108 \quad x(t_0) = \hat{x}_0,$$

$$109 \quad \langle q_i, x(t_1) \rangle - b^i \leq 0, \quad i = 1, 2, \dots, m,$$

where $x(t) \in \mathbb{R}^r$, $u(t) \in \mathbb{R}^m$, for $t \in [t_0, t_1]$; the matrices $A(\cdot)$, $B(\cdot)$, $R(\cdot)$ and $Q(\cdot)$ are continuous functions of time, with $R(t)$ symmetric, positive semi-definite and with $Q(t)$ symmetric, positive definite for all $t \in [t_0, t_1]$; and $q_i \in \mathbb{R}^r$, $b^i \in \mathbb{R}^1$, $i = 1, 2, \dots, m$. We shall assume that $u(\cdot)$ is restricted to the space $L_2^m[t_0, t_1]$, which, as we know, is a Hilbert space consisting of equivalence classes of square integrable functions from $[t_0, t_1]$ into \mathbb{R}^m , with norm $\|u\|_2 = (\int_{t_0}^{t_1} \|u(t)\|^2 dt)^{1/2}$ and scalar product $\langle u', u'' \rangle_2 = \int_{t_0}^{t_1} \langle u'(t), u''(t) \rangle dt$.

Proceeding as in Section 2.5, we eliminate the state $x(\cdot)$ from the problem (106)–(109) by solving (107), (108), to obtain

$$110 \quad x(t) = x(t; u) = \Phi(t, t_0) x_0 + \int_{t_0}^t \Phi(t, s) B(s) u(s) ds, \quad t \in [t_0, t_1],$$

where $\Phi(t, s)$ is determined by the differential equation,

$$111 \quad \frac{d}{dt} \Phi(t, s) = A(t) \Phi(t, s), \quad \Phi(s, s) = I, \quad t, s \in [t_0, t_f],$$

where I is the $\nu \times \nu$ identity matrix. Substituting for $x(t)$ into (106) and (109), we find that the problem (106)–(109) assumes the form,

$$112 \quad \text{minimize} \quad f^0(u) = \int_{t_0}^{t_f} \frac{1}{2} [\langle x(t; u), R(t) x(t; u) \rangle + \langle u(t), Q(t) u(t) \rangle] dt,$$

subject to

$$113 \quad \int_{t_0}^{t_f} \langle q^i, \Phi(t_f, s) B(s) u(s) \rangle ds + \langle q^i, \Phi(t_f, t_0) x_0 \rangle - b^i \leq 0, \\ i = 1, 2, \dots, m.$$

Finally, we find that the problem (112), (113) can be written as

$$114 \quad \min\{f^0(u) \mid u \in L_2^\mu[t_0, t_f], \langle f_i, u \rangle_2 - b^i \leq 0, i = 1, 2, \dots, m\},$$

where

$$115 \quad f_i(t) = B(t)^T p_i(t) \quad \text{for } t \in [t_0, t_f], \quad i = 1, 2, \dots, m,$$

$$116 \quad b^i = \bar{b}^i - \langle p_i(t_0), x_0 \rangle, \quad i = 1, 2, \dots, m,$$

and the functions $p_i : [t_0, t_f] \rightarrow \mathbb{R}^\nu$ are solutions of the adjoint equation,

$$117 \quad \frac{d}{dt} p_i(t) = -A(t)^T p_i(t), \quad t \in [t_0, t_f], \quad p_i(t_f) = q_i, \quad i = 1, 2, \dots, m.$$

To apply a gradient projection method to problem (114), we only need to know how to compute $\text{grad } f^0(u)(\cdot)$ and how to project it onto a subspace $L_{I_\epsilon(u)}^\perp = \{u \in L_2^\mu[t_0, t_f] \mid \langle f_i, u \rangle_2 = 0, i \in I_\epsilon(u)\}$, where, for $\epsilon \geq 0$,

$$118 \quad I_\epsilon(u) = \{i \mid \langle f_i, u \rangle_2 - b^i + \epsilon \geq 0, i \in \{1, 2, \dots, m\}\}.$$

Turning to (2.5.29), we see that, if it exists, then

$$119 \quad \text{grad } f^0(u)(t) = -B(t)^T p(t) + Q(t) u(t), \quad t \in [t_0, t_f],$$

where $p(t)$ is determined by the forced adjoint equation,

$$120 \quad \frac{d}{dt} p(t) = -A(t)^T p(t) + R(t) x(t; u), \quad t \in [t_0, t_f], \quad p(t_f) = 0,$$

with $x(t; u)$ computed according to (110).

Now suppose that the functions $f_i(\cdot)$, $i \in I_\epsilon(u)$ are linearly independent on $[t_0, t_f]$. Then, by solving problem (12) for $z = \text{grad } f^0(u)(\cdot)$, we obtain

$$121 \quad P_{I_\epsilon(u)} \text{grad } f^0(u)(t) = F_{I_\epsilon(u)}(t) \left[\int_{t_0}^{t_f} F_{I_\epsilon(u)}^T(t) F_{I_\epsilon(u)}(t) dt \right]^{-1} \\ \times \int_{t_0}^{t_f} F_{I_\epsilon(u)}(t) \text{grad } f^0(u)(t) dt, \quad t \in [t_0, t_f],$$

$$122 \quad P_{I_\epsilon(u)}^\perp \text{grad } f^0(u)(t) = \text{grad } f^0(u)(t) - P_{I_\epsilon(u)} \text{grad } f^0(u)(t), \quad t \in [t_0, t_f],$$

where $F_{I_\epsilon(u)}(t)$ is a matrix whose columns are the vectors $f_i(t)$, ordered linearly on i ; $P_{I_\epsilon(u)}(\cdot)(\cdot)$ is a projection operator from $L_2^\mu[t_0, t_f]$ onto the subspace $L_{I_\epsilon(u)} = \{u \in L_2^\mu[t_0, t_f] \mid u(\cdot) = \sum_{i \in I_\epsilon(u)} \mu^i f_i(\cdot), \mu^i \in \mathbb{R}^1\}$; and $P_{I_\epsilon(u)}^\perp(\cdot)(\cdot)$ is a projection operator from $L_2^\mu[t_0, t_f]$ onto $L_{I_\epsilon(u)}^\perp$.

- 123 **Exercise.** Summarize the above developments in the form of a procedure, based on algorithm (37), for solving the optimal control problem (106)–(109). ■

5

CONVEX OPTIMAL CONTROL PROBLEMS

5.1 Nonlinear Programming Algorithms Revisited

A large part of this chapter will be devoted to discrete optimal control problems which transcribe into nonlinear programming problems of the form,

$$1 \quad \min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m, Rz - b = 0\},$$

where the $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ are convex functions, R is a matrix, and b is a vector of appropriate dimensions. In this chapter, we shall refer to problem (1) as the primal problem and to algorithms that solve it as primal methods. In the next section, we shall describe a dual algorithm for solving a problem related to (1), which we shall call the dual problem. As we shall see, once we have a solution to the dual problem, it takes little work to obtain a solution to the primal problem, at least in the cases we shall consider.

Before proceeding with dual methods in the next section, we should like to remind the reader that some discrete optimal control problems of the form (1) are solvable by linear programming algorithms (such as the simplex method, the bounded variable simplex method (see [C1]), the dual simplex method, etc.), some by quadratic programming algorithms (such as the Wolfe algorithm, [C1], [W1], or the algorithm described in Section 6.10 of [C1]), and most of the others by algorithms in the family of methods of feasible directions.

Let us examine a few specific cases so as to illustrate our contention.

2 Example. Consider the problem,

$$3 \quad \text{minimize} \quad \left[M \sum_{i=1}^{k-1} \sum_{j=1}^{\nu} |x_i^j - \hat{x}_i^j| + \sum_{i=0}^{k-1} |u_i| \right], \quad M > 0,$$

subject to

$$4 \quad x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, 2, \dots, k-1, \quad x_i \in \mathbb{R}^r, \quad u_i \in \mathbb{R}^1,$$

$$5 \quad x_0 = \hat{x}_0, \quad x_k = \hat{x}_k,$$

$$6 \quad |u_i| \leq 1, \quad i = 0, 1, 2, \dots, k-1.$$

In practice, such a problem may arise when we wish to minimize the amount of fuel required to take the state from \hat{x}_0 to \hat{x}_k while keeping the trajectory $(\hat{x}_0, x_1, \dots, \hat{x}_k)$ as close as possible (in the sense of the L_1 norm) to a nominal trajectory $(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_k)$.

Now, with $x_0 = \hat{x}_0$, we obtain,

$$7 \quad x_i = A^i \hat{x}_0 + \sum_{j=0}^{i-1} A^{i-1-j} Bu_j, \quad i = 1, 2, \dots, k.$$

Let $z^{j+1} = u_j$, for $j = 0, 1, 2, \dots, k-1$. Then (7) becomes

$$8 \quad x_i = A^i \hat{x}_0 + \sum_{j=1}^i A^{i-j} B z^j, \quad i = 1, 2, \dots, k.$$

Setting $z = (z^1, z^2, \dots, z^k) \in \mathbb{R}^k$, and, for $i = 1, 2, \dots, k$, defining $d_i = A^i \hat{x}_0$ and R_i to be a $\nu \times k$ matrix whose j th column, r_{ij} , is given by $r_{ij} = A^{i-j} B$ for $j = 1, 2, \dots, i$ and $r_{ij} = 0$ for $j = i+1, i+2, \dots, k$, we can rewrite (8) as

$$9 \quad x_i = d_i + R_i z, \quad i = 1, 2, \dots, k.$$

Finally, denoting by e_j , $j = 1, 2, \dots, \nu$, the j th column of the $\nu \times \nu$ identity matrix, i.e., $e_1 = (1, 0, 0, \dots, 0)$, $e_2 = (0, 1, 0, 0, \dots, 0)$, etc., and by \bar{e}_i , $i = 1, 2, \dots, k$, the i th column of the $k \times k$ identity matrix, we find that problem (3)–(6) can be rewritten in the form,

$$10 \quad \text{minimize} \quad \left[\sum_{i=1}^{k-1} \sum_{j=1}^{\nu} M |\langle e_j, d_i + R_i z - \hat{x}_i \rangle| + \sum_{i=1}^k |\langle \bar{e}_i, z \rangle| \right],$$

subject to

$$11 \quad R_k z = \hat{x}_k - d_k,$$

$$12 \quad |z^i| \leq 1, \quad i = 1, 2, \dots, k.$$

Rearranging (10), we find that problem (3)–(6) becomes reduced to

$$13 \quad \text{minimize} \quad \left[\sum_{i=1}^{k-1} \sum_{j=1}^v M |\langle R_i^T e_j, z \rangle + \langle e_j, d_i - \hat{x}_i \rangle| + \sum_{i=1}^k |\langle \bar{e}_i, z \rangle| \right],$$

subject to (11) and (12).

Referring to theorem (5.3.7) in [C1], we find that (13) can be solved by linear programming algorithms, since it can be transcribed into the form of a standard linear programming problem.* ■

- 14 **Exercise.** Show that if $\hat{z} \in \mathbb{R}^k$ is part of an optimal solution to the linear programming problem (15) below, then it is also optimal for (13).

$$15 \quad \text{minimize} \quad \left[\sum_{i=1}^{k-1} \sum_{j=1}^v M(v_j^i + y_j^i) + \sum_{i=1}^k (w^i + t^i) \right],$$

subject to

$$16 \quad R_k z = \hat{x}_k - d_k,$$

$$17 \quad v_j^i - y_j^i = \langle R_i^T e_j, z \rangle + \langle e_j, d_i - \hat{x}_i \rangle, \quad i = 1, 2, \dots, k-1, \quad j = 1, 2, \dots, v,$$

$$18 \quad w^i - t^i = \langle \bar{e}_i, z \rangle, \quad i = 1, 2, \dots, k,$$

$$19 \quad |z^i| \leq 1, \quad i = 1, 2, \dots, k. \quad \blacksquare$$

- 20 **Example.** Consider the problem,

$$21 \quad \text{minimize} \quad \frac{1}{2} \left(\sum_{i=1}^k \|x_i - \hat{x}_i\|^2 + \sum_{i=0}^{k-1} \|u_i\|^2 \right), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^1,$$

subject to

$$22 \quad x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, 2, \dots, k-1,$$

$$23 \quad x_0 = \hat{x}_0, \quad Gx_k = d,$$

$$24 \quad |u_i| \leq 1, \quad i = 0, 1, 2, \dots, k-1.$$

Proceeding as in example (2), we see that (21)–(24) assumes the form,

$$25 \quad \text{minimize} \quad \frac{1}{2} \left(\sum_{i=1}^k \|d_i + R_i z - \hat{x}_i\|^2 + \|z\|^2 \right),$$

* The evaluation of the matrices R_i reduces the number of equality constraints in the linear program to those in (11). Alternatively, one may use decomposition techniques; see [V2].

subject to

26

$$GR_k z = d - Gd_k, \\ |z^i| \leq 1, \quad i = 1, 2, \dots, k,$$

where $z^{i+1} = u_i$, $i = 0, 1, \dots, k-1$; and for $i = 1, 2, \dots, k$, $d_i = A^i x_0$, and R_i is a $\nu \times k$ matrix whose j th column is $A^{i-j} B$ for $j = 1, 2, \dots, i$, and whose remaining elements are all zero. By inspection, problem (25), (26) is a standard quadratic programming problem, and hence is solvable by quadratic programming algorithms such as those described in [C1].

A problem such as (21)–(24) arises when we wish to take a system from a given initial state \hat{x}_0 to a given terminal manifold, and in doing so, we wish to minimize the energy expended while keeping the actual trajectory reasonably close to a nominal one defined by $(\hat{x}_0, \hat{x}_1, \dots, \hat{x}_{k-1}, \hat{x}_k)$. ■

27

Example. Consider the problem (21)–(24), with the additional constraint that

28

$$q^i(x_k) \leq 0 \quad \text{for } i = 1, 2, \dots, m,$$

where the $q^i : \mathbb{R}^\nu \rightarrow \mathbb{R}^1$ are continuously differentiable convex functions. Then problem (21)–(24), (28) reduces to the form,

29

$$\text{minimize} \quad \frac{1}{2} \left(\sum_{i=1}^k \|d_i + R_i z - \hat{x}_i\|^2 + \|z\|^2 \right),$$

subject to

30

$$q^j(d_k + R_k z) \leq 0, \quad j = 1, 2, \dots, m, \\ |z^i| \leq 1, \quad i = 1, 2, \dots, k,$$

31

$$GR_k z = d - Gd_k.$$

Referring to Section 4.3, we find that this problem is solvable by algorithms in the family of methods of feasible directions. In particular, algorithms (4.3.20) and (4.3.26) are especially suitable, provided they are modified as explained in (4.3.68), with the addition of the constraint $GR_k h = 0$ to (4.3.59)–(4.3.62), as required by the optimality condition (4.3.85). ■

5.2 A Decomposition Algorithm of the Dual Type

We shall now present a few dual methods for solving various “convex” optimal control problems, i.e., problems with linear dynamics, and convex cost and constraint functions. Dual methods differ from primal methods in

that they iterate on a *multiplier vector* which enters the optimality conditions (1.2.25)–(1.2.30) (the vector p_k), or (1.2.36)–(1.2.38) (the vector $p(t_i)$), whichever is appropriate, rather than iterate on the control sequence, or the control function, as do primal methods. As we shall see, dual methods share an important property with the methods of feasible directions; they decompose the original problem into a sequence of much simpler subproblems.

The dual methods that we shall present depend on two facts. The first is that a number of optimal control problems can be transcribed into the form of a certain geometric problem: our primal problem. The second is that a solution of the primal problem can easily be recovered from a given solution of the dual problem. A general discussion of duality not being quite appropriate in this text, the reader is referred to Chapters 8–11 of Mangasarian [M4] for a very clear and very scholarly presentation of this subject.* To show how this geometric problem arises, together with the various accompanying assumptions, let us examine the simple problem below. We shall consider a few more complex problems later on.

Thus, consider the discrete optimal control problem,

$$1 \quad \text{minimize} \quad \frac{1}{2} \sum_{i=0}^{k-1} (\|x_i - \hat{x}_i\|_P^2 + u_i^2), \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^1,$$

subject to

$$2 \quad x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, 2, \dots, k-1,$$

$$3 \quad x_0 = \hat{x}_0,$$

$$4 \quad q(x_k) = \frac{1}{2} \|x_k - \hat{x}_k\|_Q^2 - \beta \leq 0, \quad \beta > 0,$$

$$5 \quad |u_i| \leq 1, \quad i = 0, 1, 2, \dots, k-1,$$

where A is a $v \times v$ constant matrix; $B \in \mathbb{R}^v$; P is a $v \times v$ symmetric, positive semidefinite matrix; Q is a $v \times v$ symmetric, positive definite matrix; and $\|x\|_P^2$, $\|x\|_Q^2$ are defined by

$$6 \quad \|x\|_P^2 = \langle x, Px \rangle, \quad \|x\|_Q^2 = \langle x, Qx \rangle.$$

Proceeding as in Section 1, we set $z^{i+1} = u_i$ for $i = 0, 1, \dots, k-1$, then we set $z = (z^1, z^2, \dots, z^k) \in \mathbb{R}^k$, and, finally, we set, for $i = 1, 2, \dots, k$, $d_i = A^i \hat{x}_0$, and we define R_i to be a $v \times k$ matrix whose j th column is $A^{i-j}B$ for $j = 1, 2, \dots, i$ and whose remaining elements are zero. In this notation, if $x_i(z)$ is the solution of (2), (3) at time i , corresponding to the control sequence z , then $x_0(z) = \hat{x}_0$, and

$$7 \quad x_i(z) = d_i + R_i z, \quad z = (z^1, z^2, \dots, z^k) = (u_0, u_1, \dots, u_{k-1}).$$

* For a treatment on a more advanced level, see [R3].

Let $\mathcal{R}(\alpha) \subset \mathbb{R}^v$ be the set of all states reachable at time k , by the dynamical system (2) from the initial state \hat{x}_0 , when forced by an input sequence satisfying (5), with the corresponding cost not exceeding α ($\alpha \geq 0$), i.e.,

$$8 \quad \mathcal{R}(\alpha) = \left\{ x \in \mathbb{R}^v \mid x = d_k + R_k z; |z^i| \leq 1, i = 1, 2, \dots, k; \right. \\ \left. \frac{1}{2} \left[\sum_{i=1}^{k-1} \|d_i + R_i z - \hat{x}_i\|_p^2 + \|z\|^2 \right] \leq \alpha \right\}.$$

In order to motivate some of the assumptions that we shall shortly introduce, we note that for every

$$\alpha \geq \alpha_{\min} \\ = \min \left\{ \frac{1}{2} \left[\sum_{i=1}^{k-1} \|d_i + R_i z - \hat{x}_i\|_p^2 + \|z\|^2 \right] \mid |z^i| \leq 1, i = 1, 2, \dots, k \right\},$$

$$9 \quad \mathcal{R}(\alpha) = \mathcal{E}(\alpha) \cap K,$$

where K is a fixed, convex polytope,

$$10 \quad K = \{x = d_k + R_k z \mid |z^i| \leq 1, i = 1, 2, \dots, k\},$$

and $\mathcal{E}(\alpha)$ is a hyperelipsoid which grows continuously and monotonically with α ,

$$11 \quad \mathcal{E}(\alpha) = \left\{ x = d_k + R_k z \mid \frac{1}{2} \left[\sum_{i=1}^{k-1} \|d_i + R_i z - \hat{x}_i\|_p^2 + \|z\|^2 \right] \leq \alpha \right\}.$$

Setting $C = \{x \in \mathbb{R}^v \mid q(x) \leq 0\}$, we see that problem (1)–(5) can be viewed as that of finding a control sequence $\hat{z} = (\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1})$ and an $\hat{\alpha} \geq 0$ such that

$$12 \quad \hat{\alpha} = \min\{\alpha \geq 0 \mid \mathcal{R}(\alpha) \cap C \neq \emptyset\},$$

$$13 \quad \hat{x}_k = (d_k + R_k \hat{z}) \in \mathcal{R}(\hat{\alpha}) \cap C.$$

Now, the set C is strictly convex and the set $\mathcal{R}(\alpha)$ is convex. Hence, we conclude that the point \hat{x}_k is unique and that there exists a vector $\hat{s} \in \mathbb{R}^v$, $\|\hat{s}\| = 1$, such that

$$14 \quad \langle y, \hat{s} \rangle \geq \langle x, \hat{s} \rangle \quad \text{for all } y \in \mathcal{R}(\hat{\alpha}) \quad \text{and for all } x \in C,$$

i.e., \hat{s} is an outward normal to the hyperplane which is tangent to C at x_k and which separates C from $\mathcal{R}(\hat{\alpha})$. As we shall soon see, if we know \hat{s} , then

we can calculate \hat{x}_k and $\hat{\alpha}$ from it. Thus, the problem will reduce to that of finding the required unit vector \hat{s} , which we can easily see to be such that if we set $\hat{p}_k = \hat{\lambda}\hat{s}$ ($\hat{\lambda} > 0$), then the optimality conditions (1.2.25)–(1.2.30) will be satisfied for our problem.

In view of the above discussion, let us concern ourselves for a while with the following problem:

- 15 The Primal Problem.** We are given a compact set $C \subset \mathbb{R}^v$ which is either strictly convex, or else it consists of a unique point, and we are also given a map $\mathcal{R} : [\alpha_{\min}, \infty) \rightarrow 2^{\mathbb{R}^v}$, $\alpha_{\min} \geq 0$, such that

(i) for every $\alpha \geq \alpha_{\min}$, $\mathcal{R}(\alpha)$ is a compact convex set which has an interior for every $\alpha > \alpha_{\min}$;

(ii) the map $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric;*

(iii) for every $\alpha \geq \alpha_{\min}$, $\mathcal{R}(\alpha) = \mathcal{E}(\alpha) \cap K$, where K is either \mathbb{R}^v or else a convex polytope with interior, and $\mathcal{E}(\alpha)$ is a strictly convex set, with the property that $\mathcal{E}(\alpha')$ is contained in the interior of $\mathcal{E}(\alpha'')$ whenever $\alpha' < \alpha''$.

We are required to find an $\hat{\alpha} \geq \alpha_{\min}$ and a vector $\hat{x} \in C$, such that

$$16 \quad \hat{\alpha} = \min\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \geq \alpha_{\min}\},$$

$$17 \quad \{\hat{x}\} = \mathcal{R}(\hat{\alpha}) \cap C. \quad \blacksquare$$

In addition, we may wish to find a unit vector \hat{s} such that

$$18 \quad \langle x - \hat{x}, \hat{s} \rangle \leq 0 \quad \text{for all } x \in C,$$

$$19 \quad \langle x - \hat{x}, \hat{s} \rangle \geq 0 \quad \text{for all } x \in \mathcal{R}(\hat{\alpha}).$$

- 20 Assumption.** To avoid dealing with the trivial case, we shall assume that α_{\min} has been chosen so that $\mathcal{R}(\alpha_{\min}) \neq \emptyset$ and $\mathcal{R}(\alpha_{\min}) \cap C = \emptyset$. To ensure the existence of a solution to problem (15) and to avoid having to discuss the degenerate case when $K \cap C$ consists of a single point, we shall assume that for some $\alpha' \in [\alpha_{\min}, \infty)$, the set C has points in $\mathcal{E}(\alpha') \cap \mathring{K}$, where \mathring{K} is the interior of K . ■

- 21 Lemma.** Under assumption (20), there exist an $\hat{\alpha} \in \mathbb{R}^+$, a unique $\hat{x} \in C$, and an $\hat{s} \in \mathbb{R}^v$ of unit norm, which satisfy (16)–(19).

Proof. By assumption (20) there exists an $\alpha' > \alpha_{\min}$ such that $\mathcal{R}(\alpha') \cap C \neq \emptyset$. Since $[\alpha_{\min}, \alpha']$ is compact, there exists an $\hat{\alpha} \in [\alpha_{\min}, \alpha']$ such that $\hat{\alpha} = \inf\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \geq \alpha_{\min}\}$. Let $\{\alpha_i\}_{i=0}^\infty$ be any sequence

* Given two compact sets, A , B , in \mathbb{R}^v , the Hausdorff distance between these sets is defined by $d(A, B) = \max\{d_1, d_2\}$, where $d_1 = \max_{x \in A} \min_{y \in B} \|x - y\|$ and $d_2 = \max_{y \in B} \min_{x \in A} \|x - y\|$.

in $[0, \hat{\alpha}']$ which decreases monotonically to $\hat{\alpha}$, satisfying, in addition, $\mathcal{R}(\alpha_i) \cap C \neq \emptyset$ for $i = 0, 1, 2, \dots$. Then, by (ii) of (15), the compact sets $\mathcal{R}(\alpha_i) \cap C$ form a monotonically decreasing sequence, and satisfy $\mathcal{R}(\alpha_{i+1}) \cap C \subset \mathcal{R}(\alpha_i) \cap C$ for $i = 0, 1, 2, \dots$. Consequently, the sequence of sets $\{\mathcal{R}(\alpha_i) \cap C\}_{i=0}^{\infty}$ converges to the set $\bigcap_{i=0}^{\infty} (\mathcal{R}(\alpha_i) \cap C) \neq \emptyset$. Now by assumption (15) (ii), $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric, and hence, $\mathcal{R}(\alpha_i) \rightarrow \mathcal{R}(\hat{\alpha})$ as $i \rightarrow \infty$. But $\mathcal{R}(\hat{\alpha})$ is compact by (15) (i), and hence,

$$\bigcap_{i=0}^{\infty} (\mathcal{R}(\alpha_i) \cap C) = \left(\bigcap_{i=0}^{\infty} \mathcal{R}(\alpha_i) \right) \cap C = \mathcal{R}(\hat{\alpha}) \cap C,$$

and

$$\hat{\alpha} = \min\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \geq \alpha_{\min}\}.$$

Next, suppose that $\mathcal{R}(\hat{\alpha}) \cap C$ consists of more than one point, i.e., suppose that $x' \neq x''$ are both in $\mathcal{R}(\hat{\alpha}) \cap C$. Then the linear segment $l = \{x = \lambda x' + (1 - \lambda)x'' \mid \lambda \in (0, 1)\} \subset \mathcal{R}(\hat{\alpha}) \cap C$ is in the interior of both C and $\mathcal{E}(\hat{\alpha})$, since both of these sets are strictly convex. Now $\mathcal{E}(\cdot)$ is continuous in the Hausdorff metric, and by (15) (iii), $\mathcal{E}(\alpha) \subset \mathcal{E}(\hat{\alpha})$ for every $\alpha \in [\alpha_{\min}, \hat{\alpha}]$. We therefore conclude that there exists an $\alpha'' < \hat{\alpha}$ such that $\mathcal{E}(\alpha'') \cap l \neq \emptyset$. But $l \subset K$, and hence, $K \cap \mathcal{E}(\alpha'') \cap l = \mathcal{R}(\alpha'') \cap l \neq \emptyset$. However, this contradicts the optimality of $\hat{\alpha}$, and hence, $\mathcal{R}(\hat{\alpha}) \cap C$ must consist of a unique point \hat{x} .

Finally, since both $\mathcal{R}(\hat{\alpha})$ and C are convex and since their intersection consists of a single point \hat{x} , there exists a hyperplane $H = \{x \mid \langle x - \hat{x}, \hat{s} \rangle = 0\}$ which separates $\mathcal{R}(\hat{\alpha})$ from C , i.e., there exists a vector \hat{s} of unit norm which satisfies (18) and (19). ■

We shall need the following observation in justifying some of the definitions to follow:

- 22 **Proposition.** Let $\hat{\alpha}$ be defined as in (16) and suppose that (20) is satisfied. Then for any $\alpha_{\min} \leq \alpha' < \alpha'' \leq \hat{\alpha}$, $\mathcal{R}(\alpha') \neq \mathcal{R}(\alpha'')$.

Proof. Suppose that for some $\alpha', \alpha'' \in [\alpha_{\min}, \hat{\alpha}]$ satisfying $\alpha' < \alpha''$, we have $\mathcal{R}(\alpha') = \mathcal{R}(\alpha'')$. Then, since $\mathcal{E}(\alpha') \subset \mathcal{E}(\alpha'')$, by (iii) of (15), we must have $\mathcal{R}(\alpha') = K$, but this contradicts our assumption, since $\mathcal{R}(\alpha') \subset \mathcal{R}(\hat{\alpha}) \subset K$, and $\mathcal{R}(\hat{\alpha}) \neq K$, according to (20). ■

As we have already mentioned, it is possible to reduce problem (15) to that of finding a vector $\hat{s} \in \mathbb{R}^v$, of unit norm, which satisfies (18) and (19). For that purpose, we shall need the following maps: Let

- 23 $S = \{s \in \mathbb{R}^v \mid \|s\| = 1\},$

and let $v : S \rightarrow C$ be defined by*

$$24 \quad \langle x - v(s), s \rangle \leq 0 \quad \text{for all } x \in C,$$

i.e., $v(\cdot)$ is a *contact map*: $v(s)$ is the point on the boundary of the compact set C with the property that the hyperplane $\{x \mid \langle x - v(s), s \rangle = 0\}$ through $v(s)$ is a supporting hyperplane to C , and its normal, s , is an *outward* normal. Since C is strictly convex and compact, $v(\cdot)$ is well-defined.

25 **Proposition.** The map $v : S \rightarrow C$ is continuous.

Proof. Suppose that $v(\cdot)$ is not continuous. Then there must exist a sequence $s_i \rightarrow s^*$ in S such that at least one limit point v^* of the sequence $\{v(s_i)\}$ satisfies $v^* \neq v(s^*)$. Since C is compact, $\{v(s_i)\}$ must have at least one limit point. Without loss of generality, we may assume that $v(s_i) \rightarrow v^*$. By definition, we must have (see (24)),

$$26 \quad \langle v(s_i) - v(s^*), s_i \rangle \geq 0 \quad \text{for } i = 0, 1, 2, \dots$$

Since the function $\langle \cdot, \cdot \rangle$ is continuous, and $s_i \rightarrow s^*$, $v(s_i) \rightarrow v^*$, we obtain

$$27 \quad \langle v^* - v(s^*), s^* \rangle \geq 0.$$

Now, by assumption, $v^* \neq v(s^*)$, and hence, we cannot have

$$28 \quad \langle v^* - v(s^*), s \rangle = 0,$$

for this would imply that $v^* = v(s^*)$ because of the strict convexity of C . Hence, we conclude that

$$29 \quad \langle v^* - v(s^*), s^* \rangle > 0,$$

which contradicts (24). Thus, $v(\cdot)$ must be continuous. ■

Now, for every $s \in S$, let

$$30 \quad P(s) = \{x \in \mathbb{R}^\nu \mid \langle x - v(s), s \rangle = 0\},$$

i.e., $P(s)$ is the support hyperplane to C at $v(s)$, with outward normal s . Next, let $T \subset S$ be defined by

$$31 \quad T = \{s \in S \mid \langle x - v(s), s \rangle \geq 0 \text{ for all } x \in \mathcal{R}(\alpha_{\min})\},$$

i.e., for every $s \in T$, the hyperplane $P(s)$ separates $\mathcal{R}(\alpha_{\min})$ from C . Now, suppose that $\hat{\alpha}$, \hat{x} , \hat{s} , satisfy (16)–(19). Then we must have $\hat{x} = v(\hat{s})$, and, since $\mathcal{R}(\alpha_{\min}) \subset \mathcal{R}(\hat{\alpha})$, $\hat{s} \in T$.

* See figure on p. 220.

† Note that (24) can also be written as $\langle v(s), s \rangle = \max\{\langle x, s \rangle \mid x \in C\}$.

Finally, let $d : T \rightarrow \mathbb{R}^1$ and $w : T \rightarrow \mathbb{R}^r$ be defined as follows:^{*}

$$\begin{aligned} 32 \quad d(s) &= \min\{\alpha \mid \mathcal{R}(\alpha) \cap P(s) \neq \emptyset, \alpha \geq \alpha_{\min}\}, \\ 33 \quad w(s) &= \mathcal{R}(d(s)) \cap P(s). \end{aligned}$$

34 Proposition. The maps $d(\cdot)$ and $w(\cdot)$ are well-defined.

Proof. Let us consider $d(\cdot)$ first. Suppose that for some $s \in T$, $\mathcal{R}(\alpha) \cap P(s) = \emptyset$ for all $\alpha \geq \alpha_{\min}$. Then, since $\mathcal{R}(\alpha_{\min}) \subset \mathcal{R}(\alpha)$ for all $\alpha \geq \alpha_{\min}$, $P(s)$ must separate $\mathcal{R}(\alpha)$ from C for all $\alpha \geq \alpha_{\min}$, in contradiction of our assumption that C has points in $\mathcal{R}(\alpha)$ for some $\alpha \geq \alpha_{\min}$ (see (20)). Hence, there exists an $\alpha' \geq \alpha_{\min}$ such that $\mathcal{R}(\alpha') \cap P(s) \neq \emptyset$, and therefore, $\alpha'' = \inf\{\alpha \mid \mathcal{R}(\alpha) \cap P(s) \neq \emptyset, \alpha \geq \alpha_{\min}\} \leq \alpha'$. Let $\{\alpha_i\}_{i=0}^{\infty}$ be a sequence in $[\alpha_{\min}, \alpha']$ which decreases monotonically to α'' such that $\mathcal{R}(\alpha_i) \cap P(s) \neq \emptyset$ for $i = 0, 1, 2, \dots$. Then the compact sets $\mathcal{R}(\alpha_i) \cap P(s)$ form a monotonically decreasing sequence, satisfying $\mathcal{R}(\alpha_{i+1}) \cap P(s) \subset \mathcal{R}(\alpha_i) \cap P(s)$, for $i = 0, 1, 2, \dots$, and hence, $\mathcal{R}(\alpha_i) \cap P(s)$ converges in the Hausdorff metric to a set $\mathcal{R}'' \cap P(s) \neq \emptyset$. But by assumption, $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric, hence, we must have $\mathcal{R}(\alpha'') = \mathcal{R}''$, and so $\mathcal{R}(\alpha'') \cap P(s) \neq \emptyset$. We therefore conclude that $d(\cdot)$ is well-defined.

Now let us turn to the map $w(\cdot)$. We have just shown that the set $\mathcal{R}(d(s)) \cap P(s)$ is not empty for all $s \in T$. It remains to show that it consists of a unique point. So, suppose that for some $s \in T$, the set $\mathcal{R}(d(s)) \cap P(s)$ contains two distinct points, $w_1 \neq w_2$. Then, since that set is convex, it must also contain the linear segment $\{w \mid w = \lambda w_1 + (1 - \lambda) w_2, 0 \leq \lambda \leq 1\}$. Since by assumption, $\mathcal{E}(d(s))$ is strictly convex (see (15) (iii)), this leads us to the conclusion that $P(s)$ must be a support hyperplane to the set K . However, this is impossible, since by assumption (20), C has points in the interior of K , and hence, $P(s)$ cannot separate C from K as it would if it were a support hyperplane to K . Hence, $w(\cdot)$ is well-defined. ■

Now suppose that $\hat{\alpha}$, \hat{x} and \hat{s} satisfy (16)–(19). Then we must have

$$35 \quad \hat{s} \in T, \quad \hat{\alpha} = d(\hat{s}), \quad \hat{x} = v(\hat{s}) = w(\hat{s}).$$

36 Lemma. Let $\hat{\alpha}$ be defined as in (16). Then for every $s \in T$, $d(s) \leq \hat{\alpha}$.

Proof. Suppose that for some $s \in T$, $d(s) > \hat{\alpha}$. Then, since $\hat{\alpha}$ is optimal for (15), since $\mathcal{R}(\alpha)$ is convex for every $\alpha \geq \alpha_{\min}$, since $\mathcal{E}(\hat{\alpha}) \subset \mathcal{E}(d(s))$, since C is strictly convex, and because of (20), the convex set $\mathcal{R}(d(s))$ must have points in the interior of C . But this is impossible, since, by construction, the hyperplane $P(s)$ separates $\mathcal{R}(d(s))$ from C . Hence, $d(s) \leq \hat{\alpha}$ for all $s \in T$. ■

Consequently, problem (15) reduces to the following *dual problem*:

37 Dual Problem. Find an $\hat{s} \in T$ such that $d(\hat{s}) = \max\{d(s) \mid s \in T\}$.

* See figure on p. 220.

Because of (35) and (36), we see that the dual problem (37) must have a solution whenever the primal problem (15) has a solution, with $\max\{d(s) \mid s \in T\} = \hat{\alpha}$, where $\hat{\alpha}$ is as defined by (16).

- 38 Lemma.** A vector $\hat{s} \in T$ is optimal for the dual problem (37) if and only if $b(\hat{s}) = w(\hat{s})$ (i.e., if and only if \hat{s} satisfies (35)).

Proof. Suppose that \hat{s} is optimal for (37). Then $d(\hat{s}), v(\hat{s})$ and \hat{s} are optimal for (15) and hence, $v(\hat{s}) \in \mathcal{R}(d(\hat{s}))$. Consequently, $v(\hat{s}) = w(\hat{s})$.

Now suppose that $v(\hat{s}) = w(\hat{s})$; then $v(\hat{s}) \in \mathcal{R}(d(\hat{s}))$. Since $\mathcal{R}(\alpha) \cap P(\hat{s}) = \emptyset$ for all $\alpha \in [\alpha_{\min}, d(\hat{s})]$, by definition of $d(\hat{s})$, and since $P(\hat{s})$ separates C from $\mathcal{R}(\alpha)$ for all $\alpha \in [\alpha_{\min}, d(\hat{s})]$, it follows that $C \cap \mathcal{R}(\alpha) = \emptyset$ for all $\alpha \in [\alpha_{\min}, d(\hat{s})]$. Consequently, $d(\hat{s}) = \min\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \geq \alpha_{\min}\}$, which implies that \hat{s} is optimal for (37). ■

We are now ready to define an algorithm for solving the dual problem (37). For any two vectors $x, y \in \mathbb{R}^v$, let $\pi(x, y)$ denote the orthogonal projection operator from \mathbb{R}^v into the subspace spanned by x, y , i.e., for any $z \in \mathbb{R}^v$, $\langle \pi(x, y) z, z - \pi(x, y) z \rangle = 0$ and $\pi(x, y) z = t'x + t''y$ for some $t', t'' \in \mathbb{R}^1$. Next, let $p : \mathbb{R}^v \times \mathbb{R}^v \rightarrow \mathbb{R}^1$ be defined by*

$$39 \quad p(x, y) = \min\{\alpha \mid [\pi(x, y) \mathcal{R}(\alpha)] \cap [\pi(x, y) C] \neq \emptyset, \alpha \geq 0\},^{\dagger}$$

and let $A : T \rightarrow 2^T$ be defined by

$$40 \quad A(s) = \{s' \in T \mid s' = \lambda s + \mu(w(s) - v(s)), \lambda, \mu \in (-\infty, \infty); \\ d(s') = p(s, w(s) - v(s))\},$$

i.e., $A(s) = \{s' \in \pi(s, w(s) - v(s)) T \mid d(s') = p(s, w(s) - v(s))\}.$ ‡

- 41 Lemma.** Suppose that $s \in T$ is not optimal for the dual problem (37). Then for every $s' \in A(s)$,

$$42 \quad d(s') = p(s, w(s) - v(s)) > d(s).$$

Proof. Since s is a normal to the hyperplane $P(s)$ which contains both $v(s)$ and $w(s)$, s is orthogonal to $[w(s) - v(s)]$. Hence, the two vectors, s and $w(s) - v(s)$, are linearly independent, and therefore the set

$$\{s' \in S \mid s' = \lambda s + \mu[w(s) - v(s)], \lambda, \mu \in (-\infty, \infty)\}$$

* By $\pi(x, y)C$, we denote the set $\{x' = \pi(x, y)x'' \mid x'' \in C\}$, etc.

† To show that $p(x, y)$ is well-defined, we essentially repeat the arguments in lemma (21).

‡ The set $\pi(s, w(s) - v(s))T$ is a circular arc of unit radius (it is the intersection of T with the plane $\{x \mid x = \lambda s + \mu(w(s) - v(s)), -\infty < \lambda, \mu < +\infty\}$).

is a unit circle in \mathbb{R}^v . Constructing a dual for (39) in the same fashion as we have constructed (37), we conclude that

$$43 \quad p(s, w(s) - v(s)) = \max\{d(s') \mid s' \in \sigma(s)\},$$

where

$$44 \quad \sigma(s) = \{s' \in T \mid s' = \lambda s + \mu(w(s) - v(s)), \lambda, \mu \in (-\infty, \infty)\}.$$

In obtaining (43), we make use of the fact that if $s' \in \pi(s, w(s) - v(s)) \mathbb{R}^v$, $\|s'\| = 1$, is such that $\pi(s, w(s) - v(s)) P(s')$, the one-dimensional projection of the hyperplane $P(s')$ onto the two-dimensional subspace spanned by s and $[w(s) - v(s)]$, separates $\pi(s, w(s) - v(s)) \mathcal{R}(\alpha_{\min})$ from $\pi(s, w(s) - v(s)) C$, then $P(s')$ separates $\mathcal{R}(\alpha_{\min})$ from C , i.e., $s' \in T$, because

$$P(s') \cap \pi(s, w(s) - v(s)) \mathbb{R}^v = \pi(s, w(s) - v(s)) P(s').$$

Conversely, if $s' \in \sigma(s)$, so that $s' = \pi(s, w(s) - v(s)) s'$, and $\langle y, s' \rangle \geq \langle x, s' \rangle$ for all $y \in \mathcal{R}(\alpha_{\min})$, for all $x \in C$, then $\langle y, s' \rangle \geq \langle x, s' \rangle$ for all $y \in \pi(s, w(s) - v(s)) \mathcal{R}(\alpha_{\min})$, for all $x \in \pi(s, w(s) - v(s)) C$. Consequently, since $s \in \sigma(s)$, we have

$$45 \quad p(s, w(s) - v(s)) \geq d(s).$$

Suppose that

$$46 \quad p(s, w(s) - v(s)) = d(s).$$

Then we must have

$$47 \quad \pi(s, w(s) - v(s)) v(s) = \pi(s, w(s) - v(s)) w(s).$$

But this is impossible, since $v(s) \neq w(s)$ and the vector $w(s) - v(s)$ is in the range of the projection operator $\pi(s, w(s) - v(s))$. Hence, (42) must hold. ■

In view of (43) and (44), we see that the map $A(\cdot)$ is well-defined, i.e., there is no $s \in T$ such that $A(s) = \emptyset$. We can now state our first algorithm for solving the dual problem (37). This algorithm has a rather interesting history, having evolved over a number of years through the work of Krassovskii [K3], Neustadt [N1], Eaton [E1], and Polak and Deparis [P4]. The version below was first presented by Polak in [P2].

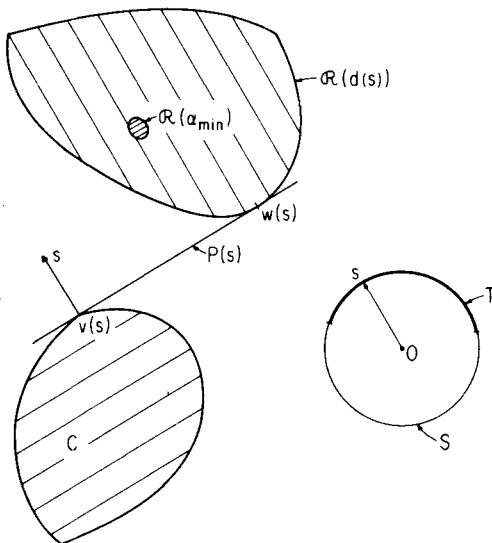
48 Algorithm (for dual problem (37), Polak [P2]).

Step 0. Compute a point $s_0 \in T$, and set $i = 0$.

Step 1. Set $s = s_i$.

Step 2. Compute a point $s' \in A(s)$.

Step 3. If $d(s') = d(s)$, set $s_{i+1} = s$ and stop; else, set $s_{i+1} = s'$, set $i = i + 1$, and go to step 1. ■



Sets and maps for algorithm (48).

- 49 **Theorem.** Let $\{s_i\}$ be any sequence in T constructed by algorithm (48). Then either $\{s_i\}$ is finite and its last element is optimal for the dual problem (37), or else $\{s_i\}$ is infinite and every accumulation point of $\{s_i\}$ is optimal for the dual problem (37).

Proof. Quite obviously, algorithm (48) is of the form of the model (1.3.9), except that it maximizes $d(s)$ instead of minimizing $c(s)$. We shall therefore need to put a minus sign in front of all the relations involving the map $d(\cdot)$ defined in (32), when showing that the assumptions of theorem (1.3.10) are satisfied by algorithm (48). For this purpose also, we define a point $s' \in T$ to be desirable if it is optimal for the dual problem (37).

Suppose that $s \in T$ is optimal for (37). Then we must have

$$50 \quad d(s) = \max\{d(s') \mid s' \in T\} \geq p(s, w(s) - v(s)) = \max\{d(s') \mid s' \in \sigma(s)\},$$

and hence, since $s \in \sigma(s)$, $d(s) = p(s, w(s) - v(s))$. (Note that when s is optimal, $v(s) = w(s)$, and therefore $\sigma(s) = \{s\}$.) Conversely, suppose that $d(s') = d(s)$ for any $s' \in A(s)$; then, by lemma (41), s must be optimal for (37). Because of this, the case of a finite sequence $\{s_i\}$ is trivial.

To establish the theorem for the case when $\{s_i\}$ is infinite, we must show that assumptions (i) and (ii) of theorem (1.3.10) are satisfied by the map $-d(\cdot)$ defined by (32) and the map $A(\cdot)$ defined by (40). Since $-d(s) \geq -\hat{\alpha}$ for all $s \in T$, where $\hat{\alpha}$ is defined as in (16), we see that (i) of (1.3.10) is satisfied

by $-d(\cdot)$, and hence we only need to show that for every nonoptimal $s \in T$, which could, *conceivably*, be an accumulation point of a sequence $\{s_i\}$ in T constructed by algorithm (48), there exist an $\epsilon(s) > 0$ and a $\delta(s) < 0$ such that

$$51 \quad -d(s'') + d(s') \leq \delta(s) \quad \text{for all } s' \in B(s, \epsilon(s)), \text{ for all } s'' \in A(s')$$

(where $B(s, \epsilon) = \{s' \in T \mid \|s' - s\| \leq \epsilon\}$). Since, by definition, for all $s'' \in A(s')$, $d(s'') = p(s', w(s') - v(s'))$, (51) becomes

$$52 \quad -\bar{d}(s') + d(s') \leq \delta(s) \quad \text{for all } s' \in B(s, \epsilon(s)),$$

where we define $\bar{d} : T \rightarrow \mathbb{R}^1$ by

$$53 \quad \bar{d}(s) = p(s, w(s) - v(s)).$$

Description of conceivable accumulation points of $\{s_i\}$. Now, let $s \in T$ be such that $\langle x' - v(s), s \rangle = 0$ for some $x' \in \mathcal{R}(\alpha_{\min})$. Then it is easy to see that we must have $w(s) = x'$ and $d(s) = \alpha_{\min}$. Further, let $\{s_i\}$ be any infinite sequence in T constructed by algorithm (48). Then, by construction, $\alpha_{\min} \leq d(s_0) < d(s_1) < d(s_2) < \dots$, and, in addition, for $i = 1, 2, 3, 4, \dots$, we must have $s_i \in T(d(s_i))$, where, for $\alpha > \alpha_{\min}$, we define

$$T(\alpha) = \{s \in T \mid \langle x - v(s), s \rangle \geq 0, \text{ for all } x \in \mathcal{R}(\alpha)\}.$$

Since $d(s_{i+1}) > d(s_i)$ for $i = 0, 1, 2, \dots$, we must have $\mathcal{R}(d(s_{i+1})) \supset \mathcal{R}(d(s_i))$, and hence, $T(d(s_{i+1})) \subset T(d(s_i)) \subset \dots \subset T(d(s_0))$ for $i = 0, 1, 2, \dots$. Now, let $s' \in T(d(s_1))$. Then we must have $d(s') \geq d(s_1)$, and since both $T(d(s_1))$ and $\mathcal{R}(\alpha_{\min})$ are compact, there must exist a $\beta > 0$ such that $\langle x - v(s'), s' \rangle \geq \beta$, for all $x \in \mathcal{R}(\alpha_{\min})$ for all $s' \in T(d(s_1))$. Otherwise, for some $s' \in T(d(s_1))$, there exists an $x' \in \mathcal{R}(\alpha_{\min})$ such that $\langle x' - v(s'), s' \rangle = 0$, implying that $d(s') = \alpha_{\min}$, in direct contradiction of the fact that $d(s') \geq d(s_1) > \alpha_{\min}$. Since $s_i \in T(d(s_1))$ for $i = 1, 2, \dots$, every accumulation point s^* of this sequence must satisfy $\langle x - v(s^*), s^* \rangle \geq \beta > 0$ for all $x \in \mathcal{R}(\alpha_{\min})$. Consequently, the only nonoptimal points $s \in T$ which can, conceivably, be limit points of a sequence $\{s_i\}$ constructed by (48), must satisfy

$$54 \quad \max\{\langle x - v(s), s \rangle \mid x \in \mathcal{R}(\alpha_{\min})\} > 0,$$

i.e., they must belong to the relative interior of T . Thus, we only need to prove (52) for all nonoptimal s in the relative interior of T . Obviously, it will suffice to show that $d(\cdot)$ and $\bar{d}(\cdot)$ are both continuous at any nonoptimal s in the relative interior of T .

Continuity of $d(\cdot)$. Let s be any point in the relative interior of T , i.e., $\langle x - v(s), s \rangle \geq \beta > 0$ for all $x \in \mathcal{R}(\alpha_{\min})$ and let $\delta' \in [\alpha_{\min}, d(s)]$ be

arbitrary. Then the sets $\mathcal{R}(d(s) - \delta')$ and $P(s)$ are strictly separated, since $\mathcal{R}(d(s) - \delta') \cap P(s) = \emptyset$, by the definition of $d(s)$. Let $w' \in P(s)$ and $w'' \in \mathcal{R}(d(s) - \delta')$ be such that

$$55 \quad \|w' - w''\| = \min\{\|x - y\| \mid x \in P(s), y \in \mathcal{R}(d(s) - \delta')\}.$$

Let $w = \frac{1}{2}(w' + w'')$; then there exists a $\gamma > 0$ such that

$$56 \quad \max_{x \in C} \langle x - w, s \rangle \leq -\gamma, \quad \min_{x \in \mathcal{R}(d(s) - \delta')} \langle x - w, s \rangle \geq \gamma.$$

Now, by theorem (B.3.20), the functions

$$\max_{x \in C} \langle x - w, \cdot \rangle \quad \text{and} \quad \min_{x \in \mathcal{R}(d(s) - \delta')} \langle x - w, \cdot \rangle$$

are continuous on \mathbb{R}^r , since both C and $\mathcal{R}(d(s) - \delta')$ are compact, and the scalar product is jointly continuous in both of its arguments. Hence, there exists an $\epsilon' > 0$ such that for all $s' \in T$, $\|s - s'\| \leq \epsilon'$,

$$57 \quad \max_{x \in C} \langle x - w, s' \rangle \leq \frac{-\gamma}{2}, \quad \min_{x \in \mathcal{R}(d(s) - \delta')} \langle x - w, s' \rangle \geq \frac{\gamma}{2},$$

which implies that

$$58 \quad d(s') > d(s) - \delta' \quad \text{for all } s' \in T, \quad \|s' - s\| < \epsilon'.$$

Now, let $\delta'' \in (d(s), \delta]$ be arbitrary. Then we must have

$$59 \quad \min_{x \in \mathcal{R}(d(s) + \delta'')} \langle x - v(s), s \rangle = -\lambda < 0.$$

Let $v' = v(s) - \gamma' s$, where $\gamma' \in (0, 1)$ is such that $v' \in C$. Then we have

$$60 \quad \min_{x \in \mathcal{R}(d(s) + \delta'')} \langle x - v', s \rangle \leq -(1 - \gamma') \lambda, \quad \max_{x \in C} \langle x - v', s \rangle = \gamma' \lambda.$$

Invoking once again theorem (B.3.20), we conclude that there exists an $\epsilon'' > 0$ such that for all $s' \in T$, satisfying $\|s' - s\| \leq \epsilon''$,

$$61 \quad \min_{x \in \mathcal{R}(d(s) + \delta'')} \langle x - v', s' \rangle \leq -\frac{(1 - \gamma') \lambda}{2}, \quad \max_{x \in C} \langle x - v', s' \rangle \geq \frac{\gamma' \lambda}{2}.$$

But (61) implies that

$$62 \quad d(s') < d(s) + \delta'' \quad \text{for all } s' \in T, \quad \|s' - s\| < \epsilon''.$$

Combining (58) and (62), we conclude that $d(\cdot)$ is continuous at s .

Continuity of $\bar{d}(\cdot)$. First, by arguments similar to the ones used above, it can be shown that the map $p : \mathbb{R}^v \times \mathbb{R}^v \rightarrow \mathbb{R}^1$, defined by (39), is continuous at every pair of vectors (x, y) which are linearly independent. Now, whenever s is not optimal for the dual problem (37), $w(s) - v(s) \neq 0$ and is orthogonal to s . Hence, we conclude that $\bar{d}(\cdot)$ is continuous at every nonoptimal $s \in T$ if $w(\cdot)$ is continuous at every nonoptimal $s \in T$ (recall that we have shown in (25) that $v(\cdot)$ is continuous).

Let $s \in T$ be nonoptimal and let $\{s_i\}_{i=0}^\infty$ be any sequence in T converging to s . Then, since $d(\cdot)$ is continuous, $d(s_i) \rightarrow d(s)$ as $i \rightarrow \infty$. Then, since $\mathcal{R}(\cdot)$ is continuous, $\mathcal{R}(d(s_i)) \rightarrow \mathcal{R}(d(s))$ as $i \rightarrow \infty$. Now, let w be any accumulation point of the sequence $\{w(s_i)\}_{i=0}^\infty$, i.e., $w(s_i) \rightarrow w$ for $i \in K \subset \{0, 1, 2, \dots\}$. Then $w(s_i) \in \mathcal{R}(d(s_i))$, and therefore $w \in \mathcal{R}(d(s))$. Also, since $w(s_i) \in P(s_i)$, we must have

$$63 \quad \langle w(s_i) - v(s_i), s_i \rangle = 0 \quad \text{for } i = 0, 1, 2, \dots.$$

But, for $i \in K$, $w(s_i) \rightarrow w$, $v(s_i) \rightarrow v(s)$, and $s_i \rightarrow s$, as $i \rightarrow \infty$. Consequently, since the scalar product is continuous,

$$64 \quad \langle w - v(s), s \rangle = 0,$$

i.e., $w \in P(s)$. But by the nature of problem (37), $\mathcal{R}(d(s)) \cap P(s)$ consists of exactly one point, $w(s)$. Hence, we must have $w = w(s)$, and as a result, $w(s_i) \rightarrow w(s)$ as $i \rightarrow \infty$, which proves that $w(\cdot)$ is continuous, and completes our proof. ■

We shall now show what is involved in applying algorithm (48) to the optimal control problem (1)–(5). First, to compute an $s_0 \in T$, we may proceed as follows: We solve the quadratic programming problem,

$$\alpha_{\min} = \min \left\{ \frac{1}{2} \left(\sum_{i=1}^{k-1} \|d_i + R_i z - x_i\|^2 + \|z\|^2 \right) \mid |z^i| \leq 1, i = 1, 2, \dots, k \right\},$$

for its unique solution \tilde{z} . Let $\tilde{w} = d_k + R_k \tilde{z}$. Then $\mathcal{R}(\alpha_{\min}) = \{\tilde{w}\}$, and by (20), $q(\tilde{w}) > 0$. We now compute $\tilde{\lambda} \in [0, 1]$ by solving $q(\lambda \tilde{w} + (1 - \lambda) \hat{x}_k) = 0$, i.e., by solving*

$$65 \quad \|\lambda \tilde{w} + (1 - \lambda) \hat{x}_k - \hat{x}_k\|_Q^2 + 2\beta = 0,$$

which yields

$$66 \quad \tilde{\lambda} = \frac{\sqrt{2\beta}}{\|\tilde{w} - \hat{x}_k\|_Q}.$$

The point $\tilde{x} = \tilde{\lambda} \tilde{w} + (1 - \tilde{\lambda}) \hat{x}_k$ is obviously on the boundary of C , and the tangent hyperplane to C passing through \tilde{x} must separate \hat{x}_k from $d_k + R_k \tilde{z}$

* Note that $x \in \mathring{C}$, where $C = \{x \mid q(x) \leq 0\}$.

and hence it must separate $\mathcal{R}(\alpha_{\min})$ from C . Consequently, the vector $s_0 = (1/\|\nabla q(\tilde{x})\|) \nabla q(\tilde{x}) \in T$, where

$$67 \quad \nabla q(\tilde{x}) = Q(\tilde{x} - \hat{x}_k).$$

Next, suppose that we have an $s \in T$ and that we wish to calculate $v(s)$. Then, because of the particular form of C , we do not need to solve the nonlinear programming problem (24), i.e., $\max\{\langle x, s \rangle \mid x \in C\}$. Instead, we see that we can compute $v(s)$ from the fact that

$$68 \quad \begin{aligned} \nabla q(v(s)) &= \lambda s, \\ q(v(s)) &= 0. \end{aligned} \quad \lambda > 0,$$

From the first part of (68), we get

$$69 \quad v(s) - \hat{x}_k = \lambda Q^{-1}s,$$

and from the second part of (68), we get

$$70 \quad \lambda^2 \langle Q^{-1}s, s \rangle - 2\beta = 0,$$

so that $\lambda = \sqrt{2\beta/\langle Q^{-1}s, s \rangle}$. Substituting back into (69), we finally obtain

$$71 \quad v(s) = \hat{x}_k + \left(\frac{2\beta}{\langle s, Q^{-1}s \rangle} \right)^{1/2} Q^{-1}s.$$

Thus, there is no difficulty, in this case, in computing $v(s)$.

To compute $d(s)$ and $w(s)$ we must solve the quadratic programming problem,

$$72 \quad \text{minimize} \quad \frac{1}{2} \left(\sum_{i=1}^{k-1} \|d_i + R_i z - x_i\|_P^2 + \|z\|^2 \right),$$

subject to

$$73 \quad \langle d_k + R_k z - v(s), s \rangle = 0, \quad |z^i| \leq 1, \quad i = 1, 2, \dots, k.$$

Suppose that $z(s)$ is optimal for (72), (73). Then it is easy to see that

$$74 \quad d(s) = \frac{1}{2} \left(\sum_{i=1}^{k-1} \|d_i + R_i z(s)\|_P^2 + \|z(s)\|^2 \right),$$

$$75 \quad w(s) = d_k + R_k z(s).$$

Thus, we can compute both $d(s)$ and $w(s)$ by means of finite procedures.

76 Remark. Suppose that the matrix P is the zero matrix. Then, in (65), $\tilde{w} = d_k$ and the evaluation of $d(s)$ and of $w(s)$ becomes considerably simplified, since, from the optimality conditions (1.2.18), it now follows that

$$77 \quad z^i(s) = \text{sat}(\psi \langle A^{k-i}B, s \rangle), \quad i = 1, 2, \dots, k. *$$

To compute ψ , we substitute from (77) into (73), to obtain the piecewise linear equation in ψ

$$78 \quad \sum_{i=1}^k \text{sat}(\psi \langle A^{k-i}B, s \rangle) \langle A^{k-i}B, s \rangle = -\langle d_k - v(s), s \rangle.$$

Since $-\langle d_k - v(s), s \rangle \leq 0$, because d_k is $\mathcal{R}(\alpha_{\min})$ and $s \in T$, and since the right-hand side of (78) is monotonically increasing in ψ , we conclude that $\psi \leq 0$. In any event, since the graph of the right-hand side of (78) is piecewise linear, the computation of ψ can be carried out quite simply and in a finite number of iterations. ■

79 Exercise. State a simple procedure for computing ψ satisfying (78). ■

So far, we have encountered no difficulties in applying algorithm (48) to problem (1)–(5). However, we are about to find one in the computation of $p(s, w(s) - v(s))$, and hence of a point $s' \in A(s)$, since this requires us to solve the problem,

$$80 \quad \min\{d(s') \mid s' \in T, s' = \lambda s + \mu(w(s) - v(s)), \lambda, \mu \in (-\infty, \infty)\}.$$

There is no finite procedure for solving (80) and hence, in practice, some sort of approximation must be introduced. It has been found empirically that one obtains satisfactory results by picking an integer M (usually $M = 3$ or 5 will do) and by examining the points

$$s_j = (1/\|s + (j/M)(w(s) - v(s))\|)[s + (j/M)(w(s) - v(s))] \quad \text{for } j = 1, 2, \dots, M,$$

and then setting $s' = s_{j'}$, where $d(s_{j'}) \geq d(s_j)$ for $j = 1, 2, \dots, M$. This procedure can be refined further by first multiplying $(w(s) - v(s))$ by a suitably chosen scale factor $\gamma \in (0, 1)$. On the problem described, this approach results in an algorithm that is considerably faster than the method of feasible directions, which is an obvious alternative. This is partly due to the fact that an s_0 for (48) is much easier to compute than a z_0 for (4.3.26).

Finally, suppose that we have an optimal \hat{s} for the dual problem which

* Recall, $\text{sat}(y) = y$ for all $y \in [-1, 1]$ and $\text{sat}(y) = \text{sgn } y$ otherwise.

corresponds to the optimal control problem (1)–(5). Then the optimal control sequence $\hat{u}_0, \hat{u}_1, \dots, \hat{u}_{k-1}$ for (1)–(5) is given by

$$81 \quad \hat{u}_i = z^{i+1}(\hat{s}), \quad i = 0, 1, 2, \dots, k-1,$$

where the $z^i(\hat{s})$ are determined from (72) and (73) (or from (77) and (78) when $P = 0$).

While in practice the heuristic procedure outlined above is quite adequate, the theoretically minded may feel more comfortable with an algorithm of the form of model (1.3.33), which can be constructed as follows: For $\delta \in [0, 2\pi]$, let*

$$82 \quad s'(\delta, s) = \left[\frac{1}{\|(\cos \delta)s + (\sin \delta)(w(s) - v(s))\|} \right] \times [(\cos \delta)s + (\sin \delta)[w(s) - v(s)]].$$

83 **Exercise.** Let $\bar{\delta} \in [0, 2\pi]$ be such that $\bar{\delta} = \max\{\delta \mid s'(\delta, s) \in T\}$. Show that $d(s'(\cdot, s))$ is quasi-concave on $[0, \bar{\delta}]$. (A function $d' : [0, \bar{\delta}] \rightarrow \mathbb{R}^1$ is said to be quasi-concave if the set $\{\delta \in [0, \bar{\delta}] \mid d'(\delta) \geq \gamma\}$ is convex for every real γ .) ■

84 **Exercise.** Show that if $\bar{\delta} \in [0, 2\pi]$ is such that

$$d(s'(\bar{\delta}, s)) = \max\{d(s'(\delta, s)) \mid \delta \in [0, 2\pi]\},$$

then $\bar{\delta} \in [0, \bar{\delta}]$, where $\bar{\delta}$ is as defined in (83). As a result, show that

$$85 \quad p(s, w(s) - v(s)) = \max\{d(s'(\delta, s)) \mid \delta \in [0, 2\pi]\} \\ = \max\{d(s'(\delta, s)) \mid \delta \in [0, \bar{\delta}]\}. \quad \blacksquare$$

86 **Exercise.** Show that the Golden section search procedure (2.1.14) can be used to compute $p(s, w(s) - v(s))$ because $d(s'(\cdot, s))$ is quasi-concave on $[0, \bar{\delta}]$. ■

Because of the facts which the reader was invited to prove for himself in the preceding three exercises, algorithm (48) can be extended to conform to the model (1.3.33), as follows:

87 **Algorithm** (for solving problem (37), Polak [P3]).

Step 0. Compute an $s_0 \in T$; select an $\bar{\epsilon} > 0$, a $\beta \in (0, 1)$, and a $\rho \geq 1$; set $i = 0$.

Step 1. Set $\epsilon = \bar{\epsilon}$.

Step 2. Set $s = s_i$.

Step 3. Compute $v(s)$, $w(s)$ and $d(s)$.

* Referring to (44), we see that an alternative description for $\sigma(s)$ (in spherical coordinates) is $\sigma(s) = \{s'(\delta, s) \in T \mid \delta \in [0, 2\pi]\}$.

Step 4. If $v(s) = w(s)$, set $s_{i+1} = s$ and stop; else, go to step 5.

Step 5. Set $\theta(\delta) = -d(s'(\delta, s))$ for $\delta \in [0, 2\pi]$.

Step 6. Use procedure (2.1.14) (with the current value of ϵ , and ρ as in step 0) to compute a $\bar{\mu}$.*

Step 7. Compute $d(s'(\bar{\mu}, s))$.

Step 8. If $d(s'(\bar{\mu}, s)) - d(s) \geq \epsilon$, set $s_{i+1} = s'(\bar{\mu}, s)$, set $i = i + 1$, and go to step 1; else, set $\epsilon = \beta\epsilon$, and go to step 6. ■

- 88 **Exercise.** Show that if $\{s_i\}$ is a sequence in T constructed by algorithm (87), then, either $\{s_i\}$ is finite and its last element is optimal for (37), or else $\{s_i\}$ is infinite, and then every accumulation point of $\{s_i\}$ is optimal for (37). [Hint: Show that the assumptions of theorem (1.3.27) are satisfied by algorithm (87).] ■

- 89 **Exercise.** Show that if algorithm (87) is modified into a time-varying version by replacing the words “go to step 1” by the words “go to step 2” in the instruction in step 8, then the convergence properties stated in (88) remain unaffected. [Hint: Make use of an appropriate model in Section 1.3.] ■

In problem (1)–(5), the constraint set C was described by a single quadratic inequality. As a result, the computation of the point $v(s)$ presented no difficulty whatsoever. However, suppose that

$$90 \quad C = \{x \in \mathbb{R}^v \mid q^i(x) \leq 0, i = 1, 2, \dots, m\},$$

and the $q^i(\cdot)$ are strictly convex, but not quadratic functions, and that C has an interior. Then there is no way for computing $v(s)$ in a finite number of iterations, and hence neither algorithm (48) nor algorithm (87) can be applied to solving the dual problem (37) without some additional modifications. To keep the discussion as simple as possible, we shall develop a heuristic elaboration of algorithm (48). The reader may wish to extend this elaboration to algorithm (87) for himself.

We shall suppose that the functions $q^i(\cdot)$ are not only strictly convex, but also continuously differentiable. We begin by introducing an exterior penalty function for the set C ; let $\bar{p} : \mathbb{R}^v \rightarrow \mathbb{R}^1$ be defined by

$$91 \quad \bar{p}(x) = \sum_{i=1}^m (\max\{0, q^i(x)\})^2,$$

which we then use to define an approximation set to $v(s)$, as follows: Let $\gamma \geq 0$, $\beta > 0$ be given scale factors; then, for every $\epsilon > 0$ and $s \in T$, we define

$$92 \quad V_\epsilon(s) = \left\{ x \in \mathbb{R}^v \mid \left\| \frac{\partial}{\partial x} \left(-\langle x, s \rangle + \frac{1}{\beta\epsilon} \bar{p}(x) \right) \right\|^2 \leq \gamma\epsilon \right\}.$$

* It is necessary to modify (2.1.14) slightly so as to ensure that $[a_0, b_0] \subset [0, \delta]$, where δ is as in (83).

Note that if γ is chosen to be zero, then $V_\epsilon(s)$ contains exactly one point, $v_\epsilon(s)$, which minimizes (the convex function) $-\langle x, s \rangle + (1/\beta\epsilon) \bar{p}(x)$ over \mathbb{R}^v (since $\langle x, s \rangle = 0$ is not possible for all $x \in C$ because C has an interior). Furthermore, by referring to Section 4.1, we find that if $\gamma = 0$, then (see theorem (4.1.21)) $v_\epsilon(s) \rightarrow v(s)$ as $\epsilon \rightarrow 0$, $\epsilon \in [0, \bar{\epsilon}], \bar{\epsilon} > 0$,* and (see lemma (4.1.11))

$$93 \quad -\langle v_\epsilon(s), s \rangle \leq -\langle v(s), s \rangle \quad \text{for all } \epsilon \in [0, \bar{\epsilon}].$$

Consequently, if $\gamma = 0$,

$$94 \quad \langle v_\epsilon(s) - v(s), s \rangle \geq 0 \quad \text{for all } \epsilon \in [0, \bar{\epsilon}],$$

and therefore, $v_\epsilon(s)$ is separated from C by the hyperplane $P(s)$ passing through $v(s)$, with normal s . Now, let

$$95 \quad P(x, s) = \{x' \in \mathbb{R}^v \mid \langle x' - x, s \rangle = 0\}, \quad x \in \mathbb{R}^v, \quad s \in S,$$

i.e., $P(x, s)$ is a hyperplane through x with normal s . Then the set C must lie to one side of $P(v_\epsilon(s), s)$, i.e.,

$$96 \quad \langle x - v_\epsilon(s), s \rangle \leq 0 \quad \text{for all } x \in C.$$

Next we extend our functions $d(\cdot)$ and $w(\cdot)$. Let $U' \subset \mathbb{R}^v \times S$ be such that for every $(x, s) \in U'$ there exists an $\alpha \geq \alpha_{\min}$ such that $\mathcal{R}(\alpha) \cap P(x, s) \neq \emptyset$. Then we define $\tilde{d} : U' \rightarrow \mathbb{R}^1$ as

$$97 \quad \tilde{d}(x, s) = \min\{\alpha \mid \mathcal{R}(\alpha) \cap P(x, s) \neq \emptyset, \alpha \geq \alpha_{\min}\}.$$

Let $U \subset U'$ be such that for every $(x, s) \in U$, $\mathcal{R}(\tilde{d}(x, s)) \cap P(x, s)$ consists of a unique point. Then we define $\tilde{w} : U \rightarrow \mathbb{R}^v$ by

$$98 \quad \{\tilde{w}(x, s)\} = \mathcal{R}(\tilde{d}(x, s)) \cap P(x, s).$$

We can now incorporate these maps into a heuristic algorithm, as follows:

99 **Algorithm** (for solving the dual problem (37), Polak).

Step 0. Compute an $s_0 \in T$; select an $\bar{\epsilon} > 0$, a $\beta \in (0, 1)$, and scale factors $\gamma \geq 0$, $\beta > 0$; set $i = 0$.

Step 1. Set $\epsilon = \bar{\epsilon}$.

Step 2. Set $s = s_i$.

Step 3. Compute a point $v_\epsilon(s) \in V_\epsilon(s)$ (as in (92)).

Step 4. Compute $\tilde{d}(v_\epsilon(s), s)$, $\tilde{w}(v_\epsilon(s), s)$.

Step 5. Set $s'(\delta) = [1/\|(\cos \delta)s + (\sin \delta)(\tilde{w}(v_\epsilon(s), s) - v_\epsilon(s))\|][(\cos \delta)s + (\sin \delta)(\tilde{w}(v_\epsilon(s), s) - v_\epsilon(s))]$ for $\delta \in [0, 2\pi]$.

* Assuming, of course, that $v_\epsilon(s)$ is well-defined and remains in a bounded set for all $\epsilon \in [0, \bar{\epsilon}], \bar{\epsilon} > 0$.

Step 6. For each $\delta \in [0, 2\pi]$, compute a vector $v_\epsilon(s'(\delta))$.

Step 7. Compute $\hat{\delta} \in [0, 2\pi]$ such that

$$100 \quad d(v_\epsilon(s'(\hat{\delta})), s'(\hat{\delta})) = \max\{d(v_\epsilon(s'(\delta)), s'(\delta)) \mid \delta \in [0, 2\pi]\}.$$

Step 8. If $d(v_\epsilon(s'(\hat{\delta})), s'(\hat{\delta})) - d(v_\epsilon(s), s) \geq \epsilon$, set $s_{i+1} = s'(\hat{\delta})$, set $i = i + 1$, and go to step 1; else, set $\epsilon = \beta\epsilon$ and go to step 3. ■

- 101 **Exercise.** Justify algorithm (99) by showing that if $\gamma = 0$, and $\{s_i\}$ is any infinite sequence constructed by algorithm (99), then any accumulation point of $\{s_i\}$ is optimal for the dual problem (37). [Hint: Show that the assumptions of theorem (1.3.27) are satisfied by algorithm (99) with $\gamma = 0$.] ■

In practice, of course, we cannot set $\gamma = 0$, since we would not be able to compute $v_\epsilon(s)$ in a finite number of iterations by any of the methods discussed in Chapter 2. In addition, we cannot possibly compute a $v_\epsilon(s'(\delta))$ for every $\delta \in [0, 2\pi]$, as specified in step 6 of (99). Thus, we would choose a $\gamma > 0$ and use a finite search over the circle $\{s'(\delta) \mid \delta \in [0, 2\pi]\}$. For example, we might restrict ourselves in steps 6 and 7 of (99) to the values $\delta = 0, 2\pi/M, 4\pi/M, 6\pi/M, \dots, 2\pi(M-1/M)$, where M is a judiciously chosen integer.

To further illustrate the applicability of algorithm (87) to optimal control problems, let us consider the two continuous optimal control problems given below.

- 102 **Example.** Consider the minimum-time optimal control problem,

$$103 \quad \text{minimize} \quad T,$$

subject to

$$104 \quad \frac{d}{dt} x(t) = Ax(t) + Bu(t), \quad t \in [0, T], \quad x(t) \in \mathbb{R}^v, \quad u(t) \in \mathbb{R}^1,$$

$$105 \quad x(0) = 0, \quad q(x(T)) \leq 0, \quad |u(t)| \leq 1, \quad t \in [0, T],$$

where A, B are constant matrices, $q(\cdot)$ is as defined in (4), and $u(\cdot)$ is a piecewise continuous function. We shall assume that (104) is completely controllable.

For this problem, we define, for $\alpha \geq 0$ ($\alpha_{\min} = 0$),

$$106 \quad \mathcal{R}(\alpha) = \left\{ x(\alpha') = \int_0^{\alpha'} e^{(\alpha'-t)A} Bu(t) dt \mid u(\cdot) \in \mathcal{U}, \alpha' \in [0, \alpha] \right\},^*$$

* That is, $x(\alpha')$ is the solution of (104) at $t = \alpha'$, with $x(0) = 0$, and corresponding to some admissible control $u(\cdot)$.

where \mathcal{U} is the set of all real-valued, piecewise continuous functions $u(\cdot)$, defined on $[0, \infty)$ and satisfying $|u(t)| \leq 1$ for all $t \in [0, \infty)$. It is easy to see that for every $\alpha \geq 0$, $\mathcal{R}(\alpha)$ is a compact, convex set, and that $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric. In addition, since (104) is completely controllable, it can be shown that the set $\mathcal{R}(\alpha)$ is compact and strictly convex for every $\alpha \geq 0$, and that $\mathcal{R}(\alpha')$ is contained in the interior of $\mathcal{R}(\alpha'')$ whenever $\alpha' < \alpha''$. (These facts can be found in most intermediate level texts on the theory of optimal control.) Thus, this problem satisfies the assumptions (15) (i)–(iii).

To compute $d(s)$ and $w(s)$ for this problem, we must solve the subproblem,

$$107 \quad \min\{T' \mid \xi(T', u) \in P(s), u \in \mathcal{U}\},$$

where $\xi(T', u)$ is the solution at T' of (104), corresponding to $x(0) = 0$ and to the indicated control $u(\cdot) \in \mathcal{U}$. Applying the maximum principle (1.2.35) to problem (107), we find that if $u(\cdot, s)$ is the optimal control, $T'(s)$ is the minimum time and $x(\cdot, s)$ is the corresponding optimal trajectory, then $\xi(T', u(\cdot, s)) = x(T', s)$,

$$108 \quad \frac{d}{dt} x(t, s) = Ax(t, s) + Bu(t, s), \quad t \in [0, T'(s)], \quad x(0, s) = 0, \quad u(\cdot, s) \in \mathcal{U},$$

the corresponding costate $p(\cdot, s)$ satisfies

$$109 \quad \frac{d}{dt} p(t, s) = -A^T p(t, s), \quad t \in [0, T'(s)], \quad p(T', s) = \psi s, \quad \psi < 0,$$

and the maximum relation (1.2.38) is satisfied, i.e.,

$$110 \quad B^T p(t, s) u(t, s) \geq B^T p(t, s) v \quad \text{for all } v \in [-1, +1], \quad \text{and almost all } t \in [0, T'(s)].$$

From (110), we conclude that

$$111 \quad u(t, s) = \operatorname{sgn} B^T p(t, s) \quad \text{for almost all } t \in [0, T'(s)].$$

Now, from (109),

$$112 \quad p(t, s) = e^{-(t-T'(s))A^T} \psi s \quad \text{for } t \in [0, T'(s)],$$

and hence,

$$113 \quad u(t, s) = \operatorname{sgn}(\psi \langle s, e^{-(t-T'(s))A} B \rangle), \quad \text{for almost all } t \in [0, T'(s)].$$

To determine ψ in (113), we make use of the boundary condition $x(T'(s), s) \in P(s)$, i.e., of the equation,

$$114 \quad \langle x(T'(s), s) - v(s), s \rangle = 0.$$

Now, from (108),

$$115 \quad \begin{aligned} x(T'(s), s) &= \int_0^{T'(s)} e^{(T'(s)-t)A} Bu(t, s) dt \\ &= \int_0^{T'(s)} e^{(T'(s)-t)A} B \operatorname{sgn}(\psi \langle s, e^{(T'(s)-t)A} B \rangle) dt. \end{aligned}$$

Substituting into (114) and rearranging terms, we obtain,

$$116 \quad \int_0^{T'(s)} \langle s, e^{(T'(s)-t)A} B \rangle \operatorname{sgn}(\psi \langle s, e^{(T'(s)-t)A} B \rangle) dt = \langle v(s), s \rangle.$$

Since we may set $\psi = -1$,* (116) must be solved for $T'(s)$ only. Obviously, this is much harder to do than to solve the piecewise linear equation (78) which we had encountered in the discrete optimal control problem (1)–(5). In fact, there is no procedure for solving (116) in a finite number of implementable operations (even on a digital computer with infinite word length), and hence, in practice, one must always use a heuristic method for truncating the search for a solution of (116) after a finite number of operations.

However, assuming we can solve (116), then $d(s) = T'(s)$, and $w(s) = x(T'(s), s)$. This example clearly illustrates the fact that, as a rule, continuous optimal control problems, even simple ones, are much harder to solve than similar discrete optimal control problems. It also helps to point up the fact that available algorithms, when applied to continuous optimal control problems, must usually be considered as conceptual only, since any implementation of these algorithms requires some, usually heuristic, modification. ■

117 **Example.** Consider the minimum energy optimal control problem,

$$118 \quad \text{minimize} \quad \frac{1}{2} \int_0^T u(t)^2 dt,$$

subject to

$$119 \quad \frac{d}{dt} x(t) = Ax(t) + Bu(t), \quad t \in [0, T], \quad x(t) \in \mathbb{R}^v, \quad u(t) \in \mathbb{R}^1,$$

$$120 \quad x(0) = \hat{x}_0, \quad q(x(T)) \leq 0, \quad |u(t)| \leq 1, \quad t \in [0, T],$$

* For all $s \in T$ such that $d(s) = T'(s) > 0$, we must have $\langle v(s), s \rangle < 0$, for $\mathcal{R}(0) = \{0\}$. Consequently, we are justified in setting $\psi = -1$.

where A, B are constant matrices, $q(\cdot)$ is defined as in (4) and $u(\cdot)$ is a piecewise continuous function. The final time T is given and is finite. We shall assume that (119) is completely controllable. For this problem, we define, for $\alpha \geq 0$,

$$121 \quad \begin{aligned} \mathcal{R}(\alpha) = & \left\{ x(T) = e^{TA}x_0 + \int_0^T e^{(T-t)A}Bu(t) dt \mid |u(t)| \leq 1 \right. \\ & \left. \text{for } t \in [0, T], \int_0^T u(t)^2 dt \leq \alpha \right\}. \end{aligned}$$

In addition, it is understood that the $u(\cdot)$ in (121) is piecewise continuous.

As in the preceding example, it can be shown that $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric, that $\mathcal{R}(\alpha)$ is compact and strictly convex for every $\alpha \geq 0$, and that $\mathcal{R}(\alpha')$ is contained in the interior of $\mathcal{R}(\alpha'')$ whenever $\alpha' < \alpha''$, i.e., that the assumptions (15) (i)–(iii) are satisfied.

To compute $d(s)$ and $w(s)$ for this problem, we must solve the subproblem,

$$122 \quad \min \left\{ \frac{1}{2} \int_0^T u(t)^2 dt \mid \xi(T, u) \in P(s), u \in \mathcal{U} \right\},$$

where $\xi(T, u)$ is the solution of (119) at time T , from the initial state $x(0) = \dot{x}_0$, and corresponding to the control $u(\cdot)$, and \mathcal{U} is as in (106). Applying the maximum principle (1.2.35), we conclude that if $u(\cdot, s)$ is an optimal control for the subproblem (122) and $x(\cdot, s)$ is the corresponding optimal trajectory, then $\xi(T, u(\cdot, s)) = x(T, s)$,

$$123 \quad \frac{d}{dt} x(t, s) = Ax(t, s) + Bu(t, s), \quad t \in [0, T], \quad x(0, s) = \dot{x}_0, \quad u(\cdot, s) \in \mathcal{U},$$

the corresponding costate $p(\cdot, s)$ satisfies

$$124 \quad \frac{d}{dt} p(t, s) = -A^T p(t, s), \quad t \in [0, T], \quad p(T, s) = \psi s, \quad \psi < 0,$$

and the maximum relation (1.2.38) is satisfied, with $p^0 = -1$, i.e.,

$$125 \quad -u(t, s)^2 + B^T p(t, s) u(t, s) \geq -v^2 + B^T p(t, s) v$$

for all $v \in [-1, +1]$, and for almost all $t \in [0, T]$.

From (125), we conclude that

$$126 \quad u(t, s) = \text{sat}(B^T p(t, s)) \quad \text{for almost all } t \in [0, T].$$

Since $p(t, s)$ is given by (112), (126) becomes

$$127 \quad u(t, s) = \text{sat}(\psi \langle e^{(T-t)A} B, s \rangle), \quad t \in [0, T].$$

To solve for ψ in (127), we make use of the boundary condition $x(T, s) \in P(s)$, i.e., of equation (114), which becomes

$$128 \quad \int_0^T \langle s, e^{(T-t)A} B \rangle \text{sat}(\psi \langle s, e^{(T-t)A} B \rangle) dt = -\langle e^{TA} \dot{x}_0 - v(s), s \rangle,$$

since $x(T, s) = e^{TA} x_0 + \int_0^T e^{(T-t)A} B u(t, s) dt$. Equation (128) is somewhat easier to solve than (116), but, like (116), (128) cannot be solved by a finite procedure, which means that algorithm (87) becomes a conceptual algorithm with respect to problem (118)–(120). However, assuming that we have computed the correct ψ , then $d(s) = \frac{1}{2} \int_0^T u(t, s)^2 dt$, and $w(s) = x(T, s)$. ■

- 129 **Remark.** The reader should not be discouraged by the fact that neither (116) nor (128) can be solved exactly by means of a finite number of implementable operations. There is adequate empirical evidence to assert that a moderately good approximation to the actual value of $T'(s)$ in (116), or ψ in (128), is all that one needs to ensure convergence of algorithm (87) (or to be more precise, of the resulting modification of (87)). Reasonable approximations to $T'(s)$ for (116) and to ψ for (128) are not difficult to compute in a finite number of operations.

To obtain an approximation to $T'(s)$ for (116), find an interval $[T', T''] \subset [0, \infty)$ (by trying $T'(s) = T, 2T, 3T, \dots, T > 0$ in (116)) such that

$$130 \quad -\int_0^{T'} \langle s, e^{(T'-t)A} B \rangle \text{sgn}(\langle s, e^{(T'-t)A} B \rangle) dt \\ < \langle v(s), s \rangle < -\int_0^{T''} \langle s, e^{(T''-t)A} B \rangle \text{sgn}(\langle s, e^{(T''-t)A} B \rangle) dt,*$$

and then continue to subdivide this interval to obtain new values for T' , T'' , satisfying (130) until $T'' - T'$ is adequately small. Finally, approximate $T'(s)$ by T' .

To compute an adequate approximation to ψ for (128), proceed in the same spirit as for $T'(s)$, above, i.e., find an interval $[\psi', \psi''] \subset (-\infty, 0]$ such that for $\psi = \psi'$, the left-hand side of (128) is smaller than $-\langle e^{TA} \dot{x}_0 - v(s), s \rangle$, and for $\psi = \psi''$, the left-hand side of (128) is larger than this quantity. Then reduce this interval to acceptable size by consecutive halving. ■

- 131 **Exercise.** Devise an algorithm for solving the minimum-time problem (103)–(105) by adding to algorithm (87) yet one more approximation procedure for calculating $d(s)$ and $w(s)$, making use of the suggestion in the preceding remark. ■

* See footnote on p. 231.

5.3 A Decomposition Algorithm of the Primal Type

In this section we shall present an algorithm for solving the same type of problem as the ones considered in the preceding section, but under somewhat less restrictive assumptions. In particular, referring to (2.15), we find that in the preceding section, we had assumed that both the target set C and the sets $\mathcal{E}(\alpha)$, $\alpha \geq \alpha_{\min}$ were *strictly* convex. However, in many cases of interest, this is not true, as, for example, in problem (1.3)–(1.6), where, for $\alpha \geq 0$,

$$1 \quad \mathcal{R}(\alpha) = \left\{ x = d_k + R_k z \mid \sum_{i=1}^{k-1} \sum_{j=1}^v M |\langle e_j, d_i + R_i z - \hat{x}_i \rangle| + \sum_{i=1}^k |\langle \bar{e}_i, z \rangle| \leq \alpha, \right. \\ \left. |z^i| \leq 1, i = 1, 2, \dots, k \right\}$$

(see (1.10) for notation), which (when not empty) is a convex polytope and hence is not strictly convex. Note also, that in (1) $\mathcal{R}(\alpha)$ may be empty for all $\alpha \leq \alpha_{\min}$, with $\alpha_{\min} \geq 0$. Consequently, we introduce the geometric problem below, which differs from the primal problem (2.15) only in the nature of the assumption made, into the form of which we can transcribe the optimal control problems considered in Section 2 as well as problems of the form (1.3)–(1.6).

- 2 **The Geometric Problem.** We are given a convex compact set $C \subset \mathbb{R}^v$, and a map $\mathcal{R} : [\alpha_{\min}, \infty) \rightarrow 2^{\mathbb{R}^v}$, $\alpha_{\min} \geq 0$, such that

- (i) for every $\alpha \geq \alpha_{\min}$, $\mathcal{R}(\alpha)$ is a compact convex set;
- (ii) $\mathcal{R}(\alpha') \subset \mathcal{R}(\alpha'')$ whenever $\alpha' < \alpha''$;
- (iii) for any $\alpha \geq \alpha_{\min}$, for any open set $\mathcal{O} \supset \mathcal{R}(\alpha)$, there exists an $\epsilon > 0$ such that for all $\alpha' \geq \alpha_{\min}$, satisfying $|\alpha' - \alpha| < \epsilon$, $\mathcal{R}(\alpha') \subset \mathcal{O}$.

We are required to find an $\hat{\alpha} \geq \alpha_{\min}$ and an $\hat{x} \in C$ such that

$$3 \quad \hat{\alpha} = \min\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \geq \alpha_{\min}\}, \\ 4 \quad \hat{x} \in \mathcal{R}(\hat{\alpha}) \cap C. \quad \blacksquare$$

- 5 **Remark.** Note that problem (2) differs from problem (2.15) not only in that the requirement of strict convexity for C and $\mathcal{E}(\alpha)$ in (2.15) has been relaxed to a requirement simply of convexity, but also in that the continuity specified in (iii) of (2) is more frequently satisfied than the continuity specified in (ii) of (2.15). In fact, if $\mathcal{R}(\cdot)$ is continuous in the Hausdorff metric, it is continuous in the sense defined in (iii) of (2); however, the converse is not true. \blacksquare

- 6 **Assumption.** To ensure that problem (2) has a solution, we shall assume that there exists an $\alpha_{\max} \geq \alpha_{\min}$ such that $\mathcal{R}(\alpha_{\max}) \cap C \neq \emptyset$. \blacksquare

7 Lemma. Assuming that (6) is satisfied, there exist an $\hat{\alpha} \in [\alpha_{\min}, \alpha_{\max}]$ and an $\hat{x} \in C$ which satisfy (3) and (4).

Proof. Since $\mathcal{A} = [\alpha_{\min}, \alpha_{\max}]$ is compact, there must exist an $\hat{\alpha} \in \mathcal{A}$, such that $\hat{\alpha} = \inf\{\alpha \mid \mathcal{R}(\alpha) \cap C \neq \emptyset, \alpha \in \mathcal{A}\}$. Let $\{\alpha_i\}_{i=0}^{\infty}$ be any sequence in \mathcal{A} which decreases monotonically to $\hat{\alpha}$, satisfying, in addition, $\mathcal{R}(\alpha_i) \cap C \neq \emptyset$ for $i = 0, 1, 2, \dots$. Then, by (ii) of (2), the compact sets $\mathcal{R}(\alpha_i) \cap C$ form a monotonically decreasing sequence, i.e., $\mathcal{R}(\alpha_{i+1}) \cap C \subset \mathcal{R}(\alpha_i) \cap C$ for $i = 0, 1, 2, \dots$. Consequently, the sequence of sets $\{\mathcal{R}(\alpha_i) \cap C\}_{i=0}^{\infty}$ converges to the set $\bigcap_{i=0}^{\infty} (\mathcal{R}(\alpha_i) \cap C) \neq \emptyset$. But assumptions (2) (ii) and (iii) imply that $\mathcal{R}(\alpha_i) \rightarrow \mathcal{R}(\hat{\alpha})$, as $i \rightarrow \infty$, and by (2) (i), $\mathcal{R}(\hat{\alpha})$ is compact. Hence we must have $\mathcal{R}(\hat{\alpha}) \cap C = \bigcap_{i=0}^{\infty} (\mathcal{R}(\alpha_i) \cap C)$, and therefore, $\hat{\alpha}$ satisfies (3). Since $\mathcal{R}(\hat{\alpha}) \cap C \neq \emptyset$, there obviously must exist an \hat{x} satisfying (4). ■

To define an algorithm for solving the geometric problem (2), we shall make use of the following four maps which are simple extensions of the maps in the preceding section: For every nonzero $s \in \mathbb{R}^v$, let $V(s)$ be defined by

$$8 \quad V(s) = \{v \in C \mid \langle x - v, s \rangle \leq 0 \text{ for all } x \in C\},$$

i.e., $V(s)$ consists of all the points v on the boundary of C which satisfy $\langle v, s \rangle = \max\{\langle x, s \rangle \mid x \in C\}$. When C is strictly convex, $V(s)$ consists of one point only, that point being $v(s)$, as defined in (2.24).

- 9 Exercise.** Show that $V(s)$ is compact and not empty because the scalar product is continuous and C is compact. (Note that $\{v \in \mathring{C} \mid \langle x - v, s \rangle \leq 0 \text{ for all } x \in \mathring{C}\}$ is empty.) ■
- 10 Exercise.** Show that for every nonzero $s \in \mathbb{R}^v$, for any open set $\mathcal{O} \supset V(s)$, there exists an $\epsilon > 0$ such that $\mathcal{O} \supset V(s')$ for all nonzero $s' \in \mathbb{R}^v$ satisfying $\|s' - s\| < \epsilon$, i.e., show that $V(\cdot)$ is upper semicontinuous. ■

Note that if $v' \in V(s)$, then $\langle v - v', s \rangle = 0$ for all $v \in V(s)$, i.e., $V(s)$ is contained in the hyperplane $\{x \in \mathbb{R}^v \mid \langle x - v', s \rangle = 0\}$. In view of this observation, we can state the following definition: For every nonzero $s \in \mathbb{R}^v$, let $P(s)$ be the hyperplane,

$$11 \quad P(s) = \{x \in \mathbb{R}^v \mid \langle x - v, s \rangle = 0, v \in V(s)\}.$$

(Note again that all the $v \in V(s)$ define exactly the same hyperplane $P(s)$, since, quite obviously, $V(s) = P(s) \cap C$.)

Next, again let $\mathcal{A} = [\alpha_{\min}, \alpha_{\max}]$. Then, for any nonzero $s \in \mathbb{R}^v$ such that $P(s) \cap \mathcal{R}(\alpha_{\max}) \neq \emptyset$, we define $\tilde{d}(s)$ as follows:

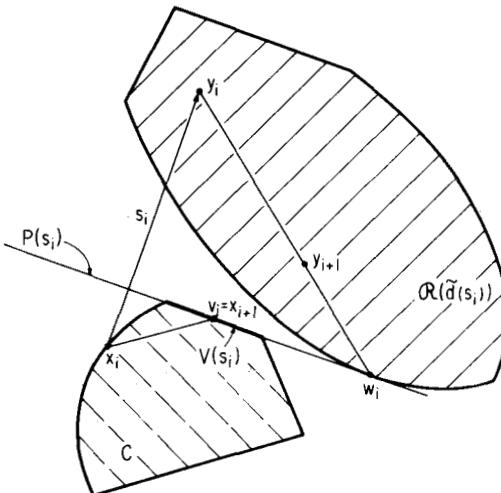
$$12 \quad \tilde{d}(s) = \begin{cases} \alpha_{\min}, & \text{if } \langle x - v, s \rangle \leq 0 \text{ for some } x \in \mathcal{R}(\alpha_{\min}), v \in V(s), \\ \min\{\alpha \mid P(s) \cap \mathcal{R}(\alpha) \neq \emptyset, \alpha \in \mathcal{A}\}, & \text{otherwise.} \end{cases}$$

Finally, for all $s \in \mathbb{R}^v$ such that $P(s) \cap \mathcal{R}(\alpha_{\max}) \neq \emptyset$, we define $W(s)$ by

$$13 \quad W(s) = \{w \in \mathcal{R}(\tilde{d}(s)) \mid \langle w - v, s \rangle \leq 0, v \in V(s)\}.$$

It is not difficult to see that, just as in the case of $P(s)$, $\tilde{d}(s)$ and $W(s)$ do not depend on the particular $v \in V(s)$ used in (12) or (13), i.e., they are functions of s only. The reason for the difference between the $d(\cdot)$ defined in (2.32) and the $\tilde{d}(\cdot)$ above, which is expressed by the first line of (12), is the fact that the domains of definition of these two functions are different. With T defined as in (2.31), we find that $d(s) = \tilde{d}(s)$ for all $s \in T$.

We shall now present an algorithm for solving the geometric problem (2), which combines the geometric ideas that were used in the construction of algorithm (2.48) with those of the Frank and Wolfe method [F5]. The algorithm below does not exhaust all the possibilities of combining the geometric ideas developed in Section 2 with those of the Frank and Wolfe algorithm. For an alternative approach, see [B3]. However, the algorithm below seems to have a greater range of applicability.



Sets and maps for algorithm (14).

14 Algorithm (Meyer-Polak [M6]).

Step 0. Compute an $x_0 \in C$ and a $y_0 \in \mathcal{R}(\alpha_{\min})$; set $i = 0$.

Step 1. Set $x = x_i$, $y = y_i$.

Step 2. If $y \in C$, set $x_{i+1} = y$, $y_{i+1} = y$, and stop; else, set $s = y - x$, and go to step 3.

Step 3. Compute a $v \in V(s)$.

Step 4. Compute $\tilde{d}(s)$.

Step 5. Compute a $w \in W(s)$.

Step 6. Compute* a $y' \in [y, w]$ and an $x' \in [x, v]$ such that

$$15 \quad \|y' - x'\| = \min\{\|y'' - x''\| \mid y'' \in [y, w], x'' \in [x, v]\}.$$

Step 7. Set $x_{i+1} = x'$, $y_{i+1} = y'$; set $i = i + 1$, and go to step 1. ■

We shall now show that algorithm (14) is of the form of model (1.3.9) and hence that its convergence properties can be deduced from theorem (1.3.10). For this purpose, we define $T = C \times \mathcal{R}(\mathcal{A})$, where \mathcal{A} is as in (3), and we define $z = (x, y)$ ($x \in C, y \in \mathcal{R}(\mathcal{A})$) to be desirable if $y \in C$. Next, for every $z = (x, y) \in T$, we define $c(z)$ by

$$16 \quad c(z) = \|x - y\|.$$

Finally, we define $A : T \rightarrow 2^T$ by

$$17 \quad A(z) = \begin{cases} \{(y, y)\}, & \text{if } z = (x, y) \text{ is desirable,} \\ \bigcup_{\substack{v \in V(y-x) \\ w \in W(y-x)}} \{z' = (x', y') \mid x' \in [x, v], y' \in [y, w]; \\ c(z') = \min\{c(z'') \mid x'' \in [x, v], y'' \in [y, w]\}\}, & \text{otherwise.} \end{cases}$$

18 **Exercise.** Suppose that $x \in C$ and that $y \in \mathcal{R}(\mathcal{A})$. Show that

$$P(y - x) \cap \mathcal{R}(\mathcal{A}) \neq \emptyset.$$

Hence, show that algorithm (14) and the corresponding map $A(\cdot)$ in (17) are well-defined. ■

19 **Exercise.** Suppose that $z = (x, y) \in T$, $x \neq y$. Show that if \mathcal{O} is any open set containing $W(y - x)$, then there exists an $\epsilon > 0$ such that $W(y' - x') \subset \mathcal{O}$, for all $z' = (x', y') \in T$ such that $\|x' - x\| < \epsilon$, $\|y' - y\| < \epsilon$. ■

20 **Exercise.** Suppose that $z = (x, y) \in T$, $x \neq y$. Show that

$$21 \quad \bar{c}(z) \triangleq \min\{c(z') \mid z' = (x', y'), x' \in [x, v], y' \in [y, w], \\ v \in V(y - x), w \in W(y - x)\} < c(z). \quad ■$$

22 **Exercise.** Let $\bar{c}(\cdot)$ be defined as in (21), and let $z = (x, y)$ be such that

* Given any $x, y \in \mathbb{R}^r$, we denote the line segment joining x and y by $[x, y]$, i.e., $[x, y] = \{v = \lambda x + (1 - \lambda)y \mid \lambda \in [0, 1]\}$.

$y \notin C$, i.e., suppose that z is not desirable. Show that for any $\delta > 0$ there exists an $\epsilon > 0$ such that

$$23 \quad \bar{c}(z') \leq \bar{c}(z) + \delta \quad \text{for all } z' \in \tilde{B}(z, \epsilon),$$

where $\tilde{B}(z, \epsilon) = \{z' \in T \mid \|x' - x\| \leq \epsilon, \|y' - y\| \leq \epsilon\}$. [Hint: See the proof of theorem (2.49).] ■

In view of (21), we see that if $z \in T$ is not desirable, then $\bar{c}(z) = \min\{c(z') \mid z' \in A(z)\} < c(z)$. Next, since $c(\cdot)$ is continuous and because of (23), we see that if $z \in T$ is not desirable, then there must exist an $\epsilon > 0$ such that, for all $z' \in B(z, \epsilon)$,

$$24 \quad c(z') \geq c(z) + \frac{\bar{c}(z) - c(z)}{3},$$

$$25 \quad \bar{c}(z') \leq \bar{c}(z) - \frac{\bar{c}(z) - c(z)}{3}.$$

Consequently, we must have

$$26 \quad \bar{c}(z') - c(z') \leq \frac{\bar{c}(z) - c(z)}{3} < 0 \quad \text{for all } z' \in B(z, \epsilon).$$

The relation (26) shows that assumption (ii) of theorem (1.3.10) is satisfied. We also note that assumption (i) of theorem (1.3.10) is satisfied by the function $c(\cdot)$ in (16), since it is obviously continuous. In view of the preceding discussion, the following result is obvious:

- 27 **Theorem.** If $\{(x_i, y_i)\}$ is a sequence constructed by algorithm (14), then either this sequence is finite and its last element (x_k, y_k) satisfies $x_k = y_k$, or else it is infinite and every accumulation point (x, y) of this sequence satisfies $x = y$. Furthermore,

$$28 \quad \sup_i d(y_i - x_i) = \hat{\alpha},$$

where $\hat{\alpha}$ is as in (3). ■

- 29 **Remark.** The most crucial difference between algorithms (14) and (2.48), is that algorithm (14), unlike (2.48), does not require us to minimize a function along a curve. As a result, algorithm (14) is implementable as is. ■

To conclude this section, let us examine what is involved in applying algorithm (14) to the following discrete optimal control problem:

$$30 \quad \text{minimize} \quad \sum_{i=0}^{k-1} u_i^2,$$

subject to

$$31 \quad x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, 2, \dots, k-1, \quad x_i \in \mathbb{R}^v, \quad u_i \in \mathbb{R}^1,$$

$$32 \quad x_0 = \hat{x}_0, \quad Qx_k \leq q, \\ |u_i| \leq 1 \quad \text{for } i = 0, 1, 2, \dots, k-1,$$

where the vector $q \in \mathbb{R}^u$ and A , B and Q are constant matrices. We assume that the set

$$33 \quad C = \{x \in \mathbb{R}^v \mid Qx \leq q\}$$

is compact, and that there is at least one control sequence u_0, u_1, \dots, u_{k-1} for which (32) is satisfied. Now, for $\alpha \geq 0$, we define

$$34 \quad \mathcal{R}(\alpha) = \left\{ y = A^k \hat{x}_0 + \sum_{i=0}^{k-1} A^{k-i-1} Bu_i \mid |u_i| \leq 1, \right. \\ \left. i = 0, 1, 2, \dots, k-1, \sum_{i=0}^{k-1} u_i^2 \leq \alpha \right\}.$$

It is not difficult to see that the $\mathcal{R}(\cdot)$ defined in (34) satisfies the assumptions stated in (2).

We can now apply algorithm (14) to problem (30)–(32), as follows: First, to avoid confusion between the x_i in (31), (32) and the x_i, y_i in (14), we shall write (as we have done on occasion before) $\overset{\circ}{x}, \overset{\circ}{y}$ for the quantities in (14). Then, we see that $\overset{\circ}{x} = 0$ is in C and $\overset{\circ}{y} = A^k \hat{x}_0$ is the only point in $\mathcal{R}(0)$. Hence, we can get started. Next, to compute a $\overset{\circ}{v} \in V(\overset{\circ}{s})$, $\overset{\circ}{s} = \overset{\circ}{y} - \overset{\circ}{x}$, we can solve the linear programming problem,

$$35 \quad \min \{ \langle \overset{\circ}{x}, \overset{\circ}{s} \rangle \mid Q\overset{\circ}{x} \leq q \},$$

by means of the simplex algorithm in a finite number of implementable operations. Next, we compute $\tilde{d}(\overset{\circ}{s}), \overset{\circ}{s} = \overset{\circ}{y} - \overset{\circ}{x}$, by solving the subproblem below, with $s = \overset{\circ}{s}$, $v = \overset{\circ}{v}$,

$$36 \quad \text{minimize} \quad \sum_{i=0}^{k-1} u_i^2,$$

subject to

$$37 \quad x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, 2, \dots, k-1,$$

$$38 \quad x_0 = \hat{x}_0, \quad \langle x_k - v, s \rangle = 0, \\ |u_i| \leq 1, \quad i = 0, 1, 2, \dots, k-1,$$

assuming, of course, that $\langle A^k \hat{x}_0 - \overset{0}{v}, \overset{0}{s} \rangle > 0$, for otherwise, $d(\overset{0}{s}) = 0$. This problem can also be solved in a finite number of implementable operations (as explained in remark (2.76)) to yield a control sequence $u_0(\overset{0}{s}), u_1(\overset{0}{s}), \dots, u_{k-1}(\overset{0}{s})$ which is optimal for (36)–(38) and, in addition satisfies

$$39 \quad d(\overset{0}{s}) = \sum_{i=0}^{k-1} u_i(\overset{0}{s})^2,$$

$$40 \quad \overset{0}{w} = \left[A^k \hat{x}_0 + \sum_{i=0}^{k-1} A^{k-i-1} B u_i(\overset{0}{s}) \right] \in W(\overset{0}{s}).$$

We now have a point $\overset{0}{x} \in C$, a point $\overset{0}{y} \in \mathcal{R}(0)$, a point $\overset{0}{v} \in V(\overset{0}{s})$ and a point $\overset{0}{w} \in W(\overset{0}{s})$. To compute $\overset{1}{x}, \overset{1}{y}$, we must solve the problem,

$$41 \quad \text{minimize} \quad \{\|x - y\|^2 \mid x \in [\overset{0}{x}, \overset{0}{v}], y \in [\overset{0}{y}, \overset{0}{w}]\}.$$

This is a simple quadratic programming problem whose solution results in two parameters, λ and μ , both contained in $[0, 1]$, such that

$$42 \quad \overset{1}{x} = \lambda \overset{0}{x} + (1 - \lambda) \overset{0}{v}, \quad \overset{1}{y} = \mu \overset{0}{y} + (1 - \mu) \overset{0}{w}.$$

Note that

$$43 \quad \overset{1}{y} = A^k \hat{x}_0 + \sum_{i=0}^{k-1} A^{k-i-1} B (1 - \mu) u_i(\overset{0}{s}).$$

Having obtained $\overset{1}{x}, \overset{1}{y}$, we now repeat our calculations to obtain $\overset{2}{x}, \overset{2}{y}$, etc. However, we are not interested in simply finding an $\overset{i}{x}$ in C and a $\overset{i}{y}$ in $\mathcal{R}(\overset{i}{x})$ (where $\overset{i}{x}$ is the minimum cost for (36)–(38)) which are sufficiently close together; we also wish to find an admissible control sequence, $\overset{i}{u}_0, \overset{i}{u}_1, \dots, \overset{i}{u}_{k-1}$, which takes the system (37) from the initial state \hat{x}_0 to the terminal state $x_k = \overset{i}{y}$, and which satisfies $\sum_{j=0}^{k-1} \overset{i}{u}_j^2 \leq \overset{i}{\alpha}$, where $\overset{i}{\alpha}$ is the minimum cost for (36)–(38). Because of this, we should organize our calculations as follows:

44 **Algorithm** (solves problem (36)–(38)).

Step 0. Set $\overset{0}{x} = 0, \overset{0}{y} = A^k \hat{x}_0$; set $\overset{0}{u}_i = 0, i = 0, 1, \dots, k-1$; set $i = 0$.

Step 1. Set $x = \overset{i}{x}, y = \overset{i}{y}$.

Step 2. If $Qy \leq q$, set $x_{i+1} = x, y_{i+1} = y$, and stop; else, set $s = y - x$, and go to step 3.

Step 3. Solve the linear programming problem, $\min\{\langle x, s \rangle \mid Qx \leq q\}$, for a vector v .

Step 4. If $\langle A^k \hat{x}_0 - v, s \rangle \leq 0$, set $w = A^k \hat{x}_0$; else, solve (36)–(38) (as explained in (2.76)) for a control sequence $u_0(s), u_1(s), \dots, u_{k-1}(s)$ and set $w = A^k \hat{x}_0 + \sum_{i=0}^{k-1} A^{k-i-1} B u_i(s)$.

Step 5. Compute $\bar{\lambda}, \bar{\mu} \in [0, 1]$ such that

$$\begin{aligned} 45 \quad & \|[\bar{\lambda}x + (1 - \bar{\lambda})v] - [\bar{\mu}y + (1 - \bar{\mu})w]\| \\ &= \min\{\|[\lambda x + (1 - \lambda)v] - [\mu y + (1 - \mu)w]\| \mid \lambda, \mu \in [0, 1]\}. \end{aligned}$$

Step 6. Set $\overset{i+1}{x} = \bar{\lambda}x + (1 - \bar{\lambda})v$; set $\overset{i+1}{y} = \bar{\mu}y + (1 - \bar{\mu})w$; set $\overset{i+1}{u_j} = \mu u_j + (1 - \mu) u_j(s)$, $j = 0, 1, 2, \dots, k - 1$; set $i = i + 1$, and go to step 1. ■

- 46 **Exercise.** Show that $\sum_{j=0}^{k-1} \overset{i}{u_j}^2 \leq \hat{\alpha}$, for $i = 0, 1, 2, \dots$, where $\hat{\alpha}$ is the minimum cost for (36)–(38). Also show that $\overset{i+1}{y} = A^k \hat{x}_0 + \sum_{j=0}^{k-1} A^{k-j-1} B \overset{i+1}{u_j}$. ■

In conclusion, we should note that while it may have been convenient, for the purpose of exposition, to characterize the algorithm in Section 2 as dual and the algorithm in this section as primal, this distinction is rather artificial. Their similarity, which results from the fact that they both emanate from the same geometric characterization of optimal problems, is much more significant than their differences, and leads us to classify them under the common and more sensible grouping of geometrically derived decomposition algorithms for optimal control problems.

6

RATE OF CONVERGENCE

6.1 Linear Convergence

The subject of rate of convergence can be treated in a very general and very abstract manner. However, for our purposes, we may restrict ourselves to a few rather simple concepts. In this section we shall only need the concept of *linear convergence*, which we define as follows: Let $\{z_i\}_{i=0}^{\infty}$ be a sequence in a Banach space \mathcal{B} which converges to a point z^* . We shall say that $\{z_i\}$ converges to z^* at least linearly (or that its rate of convergence to z^* is at least linear) if there exists an integer $k \geq 0$, a constant E and a $\theta \in [0, 1)$ such that

$$1 \quad \|z_i - z^*\|_{\mathcal{B}} \leq E\theta^i \quad \text{for all } i \geq k,$$

i.e., we say that the convergence of $\{z_i\}$ is at least linear if $\|z_i - z^*\|_{\mathcal{B}} \rightarrow 0$ as $i \rightarrow \infty$ at least as fast as a geometric progression (as before, we denote by $\|\cdot\|_{\mathcal{B}}$ the norm in \mathcal{B}).

In Chapter 2, we have presented a number of algorithms for solving the problem,

$$2 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\},$$

where $f^0(\cdot)$ was assumed to be a continuously differentiable function. With the exception of the algorithms in Section 2.2 and of the algorithms (2.3.42) and (2.3.68), all of these algorithms are characterized by the fact that the sequences $\{z_i\}$ that they construct satisfy the following relations:

$$3 \quad z_{i+1} = z_i + \lambda_i h_i, \quad i = 0, 1, 2, 3, \dots,$$

$$4 \quad -\langle \nabla f^0(z_i), h_i \rangle \geq \rho \|\nabla f^0(z_i)\| \|h_i\|, \quad \rho \in (0, 1], *$$

* Note that $\rho < 1$, because $\rho > 1$ would violate the Schwarz inequality: $|\langle \nabla f^0(z_i), h_i \rangle| \leq \|\nabla f^0(z_i)\| \|h_i\|$.

with the step size λ_i chosen according to one of the following three rules:

$$5 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}$$

(with $\lambda_i \geq 0$ the smallest value of λ for which (5) holds), or for $\alpha \in (0, \frac{1}{2})$, $\lambda_i \geq 0$ is such that

$$6 \quad \lambda_i(1 - \alpha)\langle \nabla f^0(z_i), h_i \rangle \leq f^0(z_i + \lambda_i h_i) - f^0(z_i) \leq \lambda_i \alpha \langle \nabla f^0(z_i), h_i \rangle,$$

or, for $\alpha \in (0, 1)$, $\beta \in (0, 1)$, $\bar{\rho} > 0$, $\lambda_i = \beta^j \bar{\rho}$, where j is the smallest positive integer such that

$$7 \quad f^0(z_i + \lambda_i h_i) - f^0(z_i) - \lambda_i \alpha \langle \nabla f^0(z_i), h_i \rangle \leq 0.$$

To be specific, among the algorithms that we have discussed in Chapter 2 and which are characterized by the relations (3)–(7), are the method of steepest descent (2.1.37) (where $h_i = -\nabla f^0(z_i)$, and the step size rule is (5)), the modified Newton-Raphson method (2.1.42) (where $h_i = -(\partial^2 f^0(z_i)/\partial z^2)^{-1} \nabla f^0(z_i)$ and the step size rule is (6)), various modifications of these two basic algorithms obtained by using the step size rules (5), (6) or (7), instead of the original ones, and the conjugate gradient method of Polak-Ribière (2.3.51). We recall that we have shown for all these algorithms that if the sequence $\{z_i\}$ that they construct is infinite and converges to a point z^* , then we must have $\nabla f^0(z^*) = 0$.

Throughout this section, we shall assume that we are dealing with a sequence $\{z_i\}$ constructed by an algorithm satisfying the relations (3) and (4) and using one of the step size rules (5), (6) or (7). We shall assume that this sequence converges, and that its limit point, \hat{z} , is a point where the function $f^0(\cdot)$ assumes a local minimum. Furthermore, we shall assume that the function $f^0(\cdot)$ is twice continuously differentiable and that it is strictly convex in a neighborhood of the local minimizer \hat{z} , i.e., we shall assume that there exist an $\epsilon > 0$, an $m > 0$ and an $M \geq m > 0$ such that

$$8 \quad m \|y\|^2 \leq \left\langle y, \frac{\partial^2 f^0(z)}{\partial z^2} y \right\rangle \leq M \|y\|^2,$$

for all $z \in \mathbb{R}^n$ such that $\|z - \hat{z}\| < \epsilon$ and for all $y \in \mathbb{R}^n$. Since we assume that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, there must exist an integer $k \geq 0$ such that for all $i \geq k$, $\|z_i - \hat{z}\| < \epsilon$.* Without loss of generality, we shall assume that $k = 0$. Finally, to simplify notation, we define

$$9 \quad H(z) = \frac{\partial^2 f^0(z)}{\partial z^2} \quad \text{for all } z \in \mathbb{R}^n.$$

* We recall that we have shown in theorem (2.3.8) that if (8) is true for all $z \in \{z \mid f^0(z) < f^0(z_0)\}$, then, if $\{z_i\}$ is constructed by an algorithm obeying (3), (4) and (5), it must converge to a \hat{z} which is a local minimizer of $f^0(\cdot)$. The same can also be shown for the other step size rules (6) and (7).

Before we can obtain bounds on $\|z_i - \hat{z}\|$ and on $|f^0(z_i) - f^0(\hat{z})|$, we must establish a few relationships, which we shall now proceed to do. First, by the Taylor formula for second-order expansions (B.1.12),

$$10 \quad f^0(z_i) - f^0(\hat{z}) = \langle \nabla f^0(\hat{z}), z_i - \hat{z} \rangle + \int_0^1 (1-t) \times \langle z_i - \hat{z}, H(tz_i + (1-t)\hat{z})(z_i - \hat{z}) \rangle dt, \quad i = 0, 1, 2, \dots$$

Making use of the fact that $\nabla f^0(\hat{z}) = 0$ and of (8), we now obtain

$$11 \quad m \|z_i - \hat{z}\|^2 \leq 2[f^0(z_i) - f^0(\hat{z})] \leq M \|z_i - \hat{z}\|^2, \quad i = 0, 1, 2, \dots$$

Next, making use of the Taylor formula for first-order expansions (B.1.3), we obtain, since $\nabla f^0(\hat{z}) = 0$,

$$12 \quad \nabla f^0(z_i) = \int_0^1 H(tz_i + (1-t)\hat{z})(z_i - \hat{z}) dt, \quad i = 0, 1, 2, \dots$$

Hence, making use of the Schwarz inequality for scalar products and of (8), we obtain, for $i = 0, 1, 2, \dots$,

$$13 \quad \begin{aligned} m \|z_i - \hat{z}\|^2 &\leq \int_0^1 \langle z_i - \hat{z}, H(tz_i + (1-t)\hat{z})(z_i - \hat{z}) \rangle dt \\ &= \langle \nabla f^0(z_i), z_i - \hat{z} \rangle \leq \|\nabla f^0(z_i)\| \|z_i - \hat{z}\|. \end{aligned}$$

It now follows directly from (13) that

$$14 \quad \|\nabla f^0(z_i)\| \geq m \|z_i - \hat{z}\|, \quad i = 0, 1, 2, \dots$$

Let $\Delta : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$ be defined by

$$15 \quad \Delta(z, h, \lambda) = f^0(z + \lambda h) - f^0(z).$$

Then, by the Taylor formula for second-order expansions (B.1.12),

$$16 \quad \Delta(z_i, h_i, \lambda) = \lambda \langle \nabla f^0(z_i), h_i \rangle + \lambda^2 \int_0^1 (1-t) \langle h_i, H(z_i + t\lambda h) h_i \rangle dt, \quad i = 0, 1, 2, \dots$$

Making use of (8) once again, and of (4), we obtain, for $\lambda \geq 0$,

$$17 \quad \Delta(z_i, h_i, \lambda) \leq -\lambda \rho \|\nabla f^0(z_i)\| \|h_i\| + \frac{1}{2} \lambda^2 M \|h_i\|^2, \quad i = 0, 1, 2, \dots$$

We are now ready to establish the rate of convergence of the algorithms under discussion. We begin by supposing that λ_i is chosen according to (5)

for $i = 0, 1, 2, \dots$, i.e., so as to minimize $\Delta(z_i, h_i, \lambda)$ for $\lambda \geq 0$. Then, from (17), we conclude that

$$\begin{aligned} 18 \quad \Delta(z_i, h_i, \lambda_i) &\leq \|h_i\| \left[\frac{-\rho^2 \|\nabla f^0(z_i)\|^2}{M \|h_i\|} + \frac{\rho^2 M \|\nabla f^0(z_i)\| \|h_i\|}{2M^2 \|h_i\|^2} \right] \\ &= -\frac{\rho^2}{2M} \|\nabla f^0(z_i)\|^2, \end{aligned}$$

which is the minimum of the quadratic expression in the right-hand side of (17). This minimum corresponds to $\lambda = \rho \|\nabla f^0(z_i)\| / M \|h_i\|$. Making use of (14), we find that

$$19 \quad \Delta(z_i, h_i, \lambda_i) \leq -\frac{\rho^2 m^2}{2M} \|z_i - \hat{z}\|^2, \quad i = 0, 1, 2, \dots$$

After substituting from (11) into (19), and taking into account (3), we obtain

$$\begin{aligned} 20 \quad \Delta(z_i, h_i, \lambda_i) = f^0(z_{i+1}) - f^0(z_i) &\leq -\left(\frac{\rho m}{M}\right)^2 [f^0(z_i) - f^0(\hat{z})], \\ &\quad i = 0, 1, 2, \dots, \end{aligned}$$

which leads us to the conclusion that

$$21 \quad f^0(z_{i+1}) - f^0(\hat{z}) \leq \left(1 - \left(\frac{\rho m}{M}\right)^2\right) [f^0(z_i) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots$$

Let $\theta = (1 - (\rho m/M)^2)$. Then, since $\rho \in (0, 1]^*$ and since $m \leq M$, we find that $\theta \in [0, 1)$. We now conclude from (21) that[†]

$$22 \quad f^0(z_i) - f^0(\hat{z}) \leq \theta^i [f^0(z_0) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots,$$

i.e., that $[f^0(z_i) - f^0(\hat{z})] \rightarrow 0$ as $i \rightarrow \infty$ at least at the rate of a geometric progression with constant $\theta \in [0, 1]$. In addition, since by (11),

$$23 \quad \|z_i - \hat{z}\|^2 \leq \frac{2}{m} [f^0(z_i) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots,$$

we find that

$$24 \quad \|z_i - \hat{z}\|^2 \leq \frac{2}{m} [f^0(z_0) - f^0(\hat{z})] \theta^i, \quad i = 0, 1, 2, \dots,$$

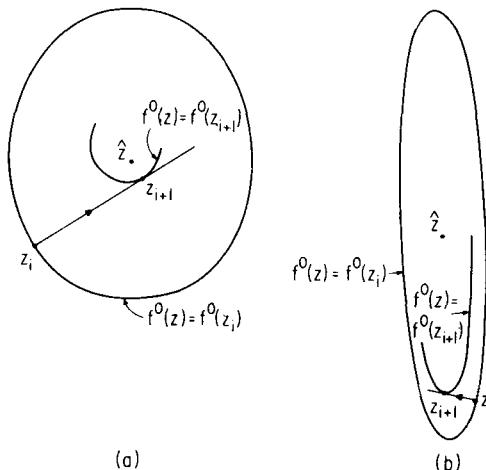
* Note that $\rho \in (0, 1]$ because $\rho > 1$ violates the Schwarz inequality.

[†] Suppose that $a_{i+1} \leq \theta a_i$, $i = 0, 1, 2, \dots$. Then, $a_1 \leq \theta a_0$, $a_2 \leq \theta a_1 \leq \theta^2 a_0$, $a_3 \leq \theta a_2 \leq \theta^3 a_0$, ..., $a_i \leq \theta^i a_0$, etc.

i.e., that $z_i - \hat{z}$ also converges to zero, at least as fast as a geometric progression with constant $\sqrt{\theta}$. Thus, setting $E = f^0(z_0) - f^0(\hat{z})$,* we find that

$$25 \quad f^0(z_i) - f^0(\hat{z}) \leq E\theta^i, \quad \|z_i - \hat{z}\| \leq \sqrt{\frac{2E}{m}} (\sqrt{\theta})^i, \quad i = 0, 1, 2, \dots,$$

which establishes at least linear convergence for algorithms satisfying the relations (3) and (4) and using the step size rule (5).



Ill-conditioning in the method of steepest descent: problem (a) is well-conditioned; problem (b) is ill-conditioned.

- 26 **Remark.** For fast rate of convergence, we must have θ in (25) small. It is quite clear that as θ approaches 1, an ill-conditioning phenomenon must appear. In constructing an algorithm characterized by (3), (4) and (5), to make $\theta = (1 - (\rho m/M)^2)$ small, we must try to make $\rho \in (0, 1]$ as large as possible.[†] We have no control over the ratio m/M , since that is entirely decided by the data of the particular problem of the form (2) that we are trying to solve. Ill-conditioning usually sets in when the graph of $f^0(z)$ versus z (in $(n + 1)$ -dimensional space) has a long, deep and narrow valley containing the limit point \hat{z} , in which case m/M is quite small. ■
- 27 **Remark.** While the formulas in (25) do provide bounds on the rate of convergence of the modified Newton-Raphson method (resulting from the

* Note that $E \geq 0$.

[†] This reasoning is valid only under the stated assumptions on $f^0(\cdot)$. Various algorithms set $\rho < 1$; yet, under stronger assumptions on $f^0(\cdot)$, they converge much faster (super-linearly) than the method of steepest descent, which sets $\rho = 1$.

use of step rule (5) instead of (6) in (2.1.42)), and of the Polak-Ribière conjugate gradient method (2.3.51), these bounds, as we shall show in the next two sections, are too conservative. However, in the case of the method of steepest descent (2.1.37), for which $\rho = 1$, the bounds on the rate of convergence given by (25) cannot be improved. (See exercise (51).) ■

Next, suppose that the step size λ_i is chosen according to (6) for $i = 0, 1, 2, \dots$. Let $\bar{A} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$ be defined by

$$28 \quad \bar{A}(z, h, \lambda) = A(z, h, \lambda) - \lambda(1 - \alpha)\langle \nabla f^0(z), h \rangle,$$

where $A(\cdot, \cdot, \cdot)$ is as defined in (15). Then, if λ_i is chosen according to (6), we must have, in addition to $\lambda_i \geq 0$, that

$$29 \quad \bar{A}(z_i, h_i, \lambda_i) \geq 0, \quad i = 0, 1, 2, \dots$$

Applying the Taylor formula for second-order expansions (B.1.12), we obtain (see (16))

$$30 \quad \bar{A}(z_i, h_i, \lambda) = \lambda\alpha\langle \nabla f^0(z_i), h_i \rangle + \lambda^2 \int_0^1 (1-t)\langle h_i, H(z_i + t\lambda h_i) h_i \rangle dt.$$

Making use of (4) and (8), we now get from (30) that

$$31 \quad \bar{A}(z_i, h_i, \lambda) \leq -\lambda\alpha\rho \|\nabla f^0(z_i)\| \|h_i\| + \frac{1}{2}\lambda^2 M \|h_i\|^2, \quad i = 0, 1, 2, \dots$$

Consequently, since λ_i satisfies (29), it must also satisfy

$$32 \quad \lambda_i \geq \frac{2\alpha\rho \|\nabla f^0(z_i)\|}{M \|h_i\|}, \quad i = 0, 1, 2, \dots$$

(since the right-hand side of (31) is negative for all $\lambda \in (0, 2\alpha\rho \|\nabla f^0(z_i)\| / M \|h_i\|)$). Now making use of (32), (4) and of the second inequality in (6), we conclude that

$$33 \quad A(z_i, h_i, \lambda_i) \leq \frac{2\alpha\rho \|\nabla f^0(z_i)\|}{M \|h_i\|} (\alpha\langle \nabla f^0(z_i), h_i \rangle) \leq -\frac{2\alpha^2\rho^2}{M} \|\nabla f^0(z_i)\|^2, \\ i = 0, 1, 2, \dots$$

Substituting from (14) into (33), we obtain

$$34 \quad A(z_i, h_i, \lambda_i) \leq -\frac{2\alpha^2\rho^2 m^2}{M} \|z_i - \hat{z}\|^2.$$

Finally, substituting from (11) into (34), we get

$$35 \quad A(z_i, h_i, \lambda_i) = f^0(z_{i+1}) - f^0(z_i) \leq -\frac{4\alpha^2\rho^2 m^2}{M^2} [f^0(z_i) - f^0(\hat{z})], \\ i = 0, 1, 2, \dots,$$

so that

$$36 \quad f^0(z_{i+1}) - f^0(\hat{z}) \leq \left(1 - \left(\frac{2\alpha\rho m}{M}\right)^2\right) [f^0(z_i) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots.$$

Consequently, upon setting $\bar{\theta} = (1 - (2\alpha\rho m/M)^2)$, we obtain

$$37 \quad f^0(z_i) - f^0(\hat{z}) \leq [f^0(z_0) - f^0(\hat{z})] \bar{\theta}^i, \quad i = 0, 1, 2, \dots,$$

and, making use of (11), we then get

$$38 \quad \|z_i - \hat{z}\| \leq \sqrt{\left(\frac{2}{m} [f^0(z_0) - f^0(\hat{z})]\right)} (\sqrt{\bar{\theta}})^i, \quad i = 0, 1, 2, \dots$$

Thus, we again get convergence at a rate at least as fast as that of a geometric progression. Note that since $\alpha \in (0, \frac{1}{2})$, $\rho \in (0, 1]$ and $m/M \leq 1$, $\bar{\theta} \in [0, 1)$.

- 39 **Remark.** For the gradient method (2.1.37), with step size rule (6), the rate of convergence cannot be shown to be faster than linear. However, for the quasi-Newton algorithm (2.1.42), the bound given by (38) is too conservative, as we shall show in the next section. ■

- 40 **Remark.** Comparing two versions of the same method, one using the step size rule (5) and the other the step size rule (6), with h_i selected in an identical manner by both versions, we find that $\theta < \bar{\theta}$ for all $\alpha \in (0, \frac{1}{2})$. We also note that $\bar{\theta}$ approaches its minimum, as a function of α , for $\alpha \in [0, \frac{1}{2}]$, at $\alpha = \frac{1}{2}$. Hence, we should use values of α that are close to 0.5, say around 0.4. Note, however, that if we make α too close to 0.5, we may spend far too much time finding a λ_i satisfying (6), since the interval containing λ_i shrinks to zero length as $\alpha \rightarrow 0.5$. Note also, that just as when the step size λ_i was determined by (5), problem (2) becomes ill-conditioned with respect to the algorithms characterized by (3), (4) and (6) when m/M becomes very small, since this leads to a value of $\bar{\theta}$ very close to 1. ■

Now let us consider the modification of the method of steepest descent characterized by the relations (3) and (4), with $h_i = -\nabla f^0(z_i)$, $i = 0, 1, 2, \dots$, and using the step size rule (7). According to (7), if we replace α by $(1 - \alpha)$ in (28), then we must have $\bar{A}(z_i, h_i, \lambda_j) \leq 0$. Referring to (31), where we replace α by $(1 - \alpha)$, and where we set $h_i = -\nabla f^0(z_i)$ and $\bar{\rho} = 1$, we conclude, by comparison with (32), that for $i = 0, 1, 2, \dots$, $\lambda_i \geq \beta^j \bar{\rho}$, where j is the smallest positive integer such that

$$41 \quad \beta^j \bar{\rho} \leq \frac{2(1 - \alpha)}{M}.$$

Consequently, we must have, according to (7) and (41) (since $h_i = -\nabla f^0(z_i)$),

$$42 \quad \Delta(z_i, h_i, \lambda_i) = f^0(z_{i+1}) - f^0(z_i) \leq -\beta^j \bar{\rho} \alpha \| \nabla f^0(z_i) \|^2, \quad i = 0, 1, 2, \dots$$

Substituting from (14) into (42), we now obtain

$$43 \quad f^0(z_{i+1}) - f^0(z_i) \leq -\beta^j \bar{\rho} \alpha m^2 \| z_i - \hat{z} \|^2.$$

Finally, substituting from (11) into (43), we get

$$44 \quad f^0(z_{i+1}) - f^0(z_i) \leq -\frac{2\beta^j \bar{\rho} \alpha m^2}{M} [f^0(z_i) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots,$$

and hence,

$$45 \quad f^0(z_{i+1}) - f^0(\hat{z}) \leq \left(1 - \frac{2\beta^j \bar{\rho} \alpha m^2}{M}\right) [f^0(z_i) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots$$

Finally, setting $\underline{\theta} = (1 - 2\beta^j \bar{\rho} \alpha m^2/M)$, we obtain from (45) that

$$46 \quad f^0(z_i) - f^0(\hat{z}) \leq [f^0(z_0) - f^0(\hat{z})] \underline{\theta}^i, \quad i = 0, 1, 2, \dots,$$

and from (46) and (11) that

$$47 \quad \| z_i - \hat{z} \| \leq \sqrt{\left(\frac{2}{m} [f^0(z_0) - f^0(\hat{z})]\right)} (\sqrt{\underline{\theta}})^i, \quad i = 0, 1, 2, \dots$$

Thus we again get convergence at a rate that is at least linear. Note that for $\alpha \in (0, 1)$, $\underline{\theta} \in [0, 1)$, because of (41) and because $m/M \leq 1$.

- 48 **Exercise.** Consider the modification of the method of steepest descent which uses the step size rule (7) (where $\rho = 1$ and $h(z) = -\nabla f^0(z)$). Show that $\underline{\theta} = (1 - 2\alpha\beta^j \bar{\rho} m^2/M)$ (in (46) and (47)) satisfies

$$49 \quad 1 - \frac{4\alpha(1 - \alpha)m}{M} \leq \underline{\theta} \leq 1 - \frac{4\alpha(1 - \alpha)\beta m^2}{M^2}$$

for all $\alpha \in (0, 1)$. Show that the upper bound on $\underline{\theta}$ is minimized by setting $\alpha = \frac{1}{2}$, which results in

$$50 \quad 1 - \frac{m}{M} \leq \underline{\theta} \leq 1 - \beta \left(\frac{m}{M}\right)^2.$$

Also show that if we set $\alpha = \frac{1}{2}$ in (7) and $\alpha = \bar{\alpha}$ in (6) (in both cases setting $\rho = 1$ and $h(\cdot) = -\nabla f^0(\cdot)$), then $\underline{\theta} \leq \bar{\theta}$ for all $\bar{\alpha} \in \{\bar{\alpha} \leq \sqrt{(\beta/2)} \mid \bar{\alpha} \in (0, \frac{1}{2})\}$. ■

- 51 **Exercise.** Obtain a bound on the rate of convergence of algorithm (2.1.35). ■

- 52 **Exercise.** To show that exponential bounds on the rate of convergence cannot be improved for first-order gradient methods such as (2.1.37), make use of the particular problem where $f^0 : \mathbb{R}^2 \rightarrow \mathbb{R}^1$ and is defined by $f^0(z) = (z^1/a)^2 + (z^2/b)^2$, $a > b > 0$. [Hint: This exercise is worked out in complete detail on p. 213 of Ostrowski [O1].] ■

We conclude this section with a result which has interesting computational implications.

- 53 **Theorem.** Consider algorithm (2.1.37) with step size procedure (33), i.e., consider the algorithm given by (3) and (6), for some $\alpha \in (0, \frac{1}{2})$, with $h_i = -\nabla f^0(z_i)$, $i = 0, 1, 2, \dots$. Suppose that problem (2) is such that (8) holds for all $z \in C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$ and that the α in (6) and the m, M in (8) satisfy

$$54 \quad \frac{\alpha}{m} \leq \frac{1 - \alpha}{M}.$$

Then we can always satisfy (6) at any $z_i \in C(z_0)$ by choosing λ_i so that

$$55 \quad \frac{2\alpha}{m} \leq \lambda_i \leq \frac{2(1 - \alpha)}{M}.$$

Proof. Let $\Delta' : \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$ be defined by

$$56 \quad \Delta'(z, \lambda) = \frac{1}{-\lambda \parallel \nabla f^0(z) \parallel^2} [f^0(z - \lambda \nabla f^0(z)) - f^0(z)].$$

Then, by the Taylor formula for second-order expansions (B.1.12), for any $z \in C(z_0)$ and any $\lambda \geq 0$,

$$57 \quad \Delta'(z, \lambda) = 1 - \frac{\lambda}{\parallel \nabla f^0(z) \parallel^2} \int_0^1 (1 - t) \langle \nabla f^0(z), H(z - t\lambda \nabla f^0(z)) \nabla f^0(z) \rangle dt.$$

Setting $z = z_i$ in (57) and invoking (8), we now obtain

$$58 \quad \frac{\lambda m}{2} \leq 1 - \Delta'(z_i, \lambda) \leq \frac{\lambda M}{2}.$$

Now, according to (6), we must choose λ_i so that

$$59 \quad \alpha \leq \Delta'(z_i, \lambda_i) \leq 1 - \alpha,$$

i.e., so that

$$60 \quad \alpha \leq 1 - \Delta'(z_i, \lambda_i) \leq 1 - \alpha.$$

Comparing (60) with (58), we find that if we choose λ_i to satisfy (55), then, because of (54), both (60) and (58) will be satisfied. ■

Thus theorem (53) indicates that at least in some cases there will be no need to construct a new step size λ_i after a certain number of iterations which may be required to enter a set in which (8) is satisfied. Note that since normally the constants m and M would not be available, the best one can do is to check, as suggested in the footnote accompanying algorithm (2.1.37), whether the previous step size λ_{i-1} is not satisfactory for the present iteration, i.e., whether it may be possible to set $\lambda_i = \lambda_{i-1}$.

6.2 Superlinear Convergence: Quasi-Newton Methods

In this section, we shall obtain bounds on the rate of convergence of the Newton-Raphson method (2.1.39), which solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$; of the Newton-Raphson method (3.1.9), which solves the problem of finding the roots of a continuously differentiable function $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ with nonsingular Jacobian $\partial g(z)/\partial z$; and of the quasi-Newton algorithm (2.1.42), which solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$. We begin with the Newton-Raphson method (3.1.9), of which (2.1.39) is a special case for $g(\cdot) = \nabla f^0(\cdot)$.

- 1 **Algorithm** (Newton-Raphson). Finds zeros of $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$, provided $(\partial g(z)/\partial z)^{-1}$ exists and is continuous.

Step 0. Select a $z_0 \in \mathbb{R}^n$.

Step 1. Set $i = 0$.

Step 2. Compute $g(z_i)$.

Step 3. If $g(z_i) = 0$, stop; else, compute $a(z_i)$ according to

$$2 \quad a(z_i) = z_i - \left(\frac{\partial g(z_i)}{\partial z} \right)^{-1} g(z_i),$$

and go to step 4.

Step 4. Set $z_{i+1} = a(z_i)$, set $i = i + 1$, and go to step 2. ■

To simplify notation, we shall denote the $n \times n$ Jacobian matrix $\partial g(z)/\partial z$ by $g'(z)$, i.e., we define

$$3 \quad g'(z) = \frac{\partial g(z)}{\partial z} \quad \text{for all } z \in \mathbb{R}^n.$$

We shall denote the second derivative of $g(\cdot)$, assuming that it exists, by $g''(\cdot)(\cdot)$. We recall from definition (B.1.7) that for any $z \in \mathbb{R}^n$, for any $y \in \mathbb{R}^n$, $g''(z)(y)$ is an $n \times n$ matrix, since $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

- 4 Proposition.** Suppose that $g'(\cdot)^{-1}$ is continuous.* If the sequence $\{z_i\}$ generated by the Newton-Raphson algorithm (1) is infinite and converges to a point \hat{z} , then $g(\hat{z}) = 0$.

Proof. Since $g'(\cdot)^{-1}$ is continuous, the map $a(\cdot)$ defined by (2) is also continuous. Now, $\{z_i\}$ satisfies

$$5 \quad z_{i+1} = a(z_i), \quad i = 0, 1, 2, \dots$$

Hence, letting $i \rightarrow \infty$, we obtain that $\hat{z} = a(\hat{z})$, i.e., that

$$6 \quad \hat{z} = \hat{z} - g'(\hat{z})^{-1} g(\hat{z}),$$

which implies that $g(\hat{z}) = 0$, since the matrix $g'(\hat{z})^{-1}$ is nonsingular by assumption. ■

Let $a'(z)$ denote the Jacobian matrix $\partial a(z)/\partial z$ for all $z \in \mathbb{R}^n$, where $a(\cdot)$ is defined as in (2). Then, we must have, for any $y, z \in \mathbb{R}^n$,

$$7 \quad \begin{aligned} a'(z) y &= y - \left[\frac{\partial}{\partial z} g'(z)^{-1} \right] (y) g(z) - g'(z)^{-1} g'(z) y \\ &= - \left[\frac{\partial}{\partial z} g'(z)^{-1} \right] (y) g(z), \end{aligned}$$

where I is the $n \times n$ identity matrix, and $[(\partial/\partial z) g'(z)^{-1}](\cdot)$ is a linear operator defined by

$$8 \quad \lim_{\|y\| \rightarrow 0} \frac{1}{\|y\|} \left(g'(z + y)^{-1} - g'(z)^{-1} - \left[\frac{\partial}{\partial z} g'(z)^{-1} \right] (y) \right) = 0,$$

provided this limit exists (compare (B.1.7)). We shall assume that the limit in (8) exists, in which case, for every $y \in \mathbb{R}^n$, we see that $[(\partial/\partial z) g'(z)^{-1}](y)$ is an $n \times n$ matrix.

Assuming that $g''(\cdot)(\cdot)$ exists and is continuous in z ,[†] we find, since

$$9 \quad \frac{\partial}{\partial z} [g'(z) g'(z)^{-1}] = \frac{\partial}{\partial z} I = 0 \quad \text{for all } z \in \mathbb{R}^n,$$

that

$$10 \quad g''(z)(\cdot) g'(z)^{-1} + g'(z) \left[\frac{\partial}{\partial z} g'(z)^{-1} \right](\cdot) = 0 \quad \text{for all } z \in \mathbb{R}^n.$$

* We say that a matrix-valued function $G(\cdot)$ from \mathbb{R}^n into the space of all $n \times n$ matrices is continuous if all the components of $G(\cdot)$ are continuous.

† We say that $g''(\cdot)(\cdot)$ is continuous in z if $z_i \rightarrow z$ as $i \rightarrow \infty$ always implies that $g''(z_i)(y) \rightarrow g''(z)(y)$ as $i \rightarrow \infty$, for any $y \in \mathbb{R}^n$.

Hence, substituting from (10) into (7), we obtain, for all $y \in \mathbb{R}^n$,

$$11 \quad a'(z)y = g'(z)^{-1}g''(z)(y)g'(z)^{-1}g(z).$$

Suppose that \hat{z} is a zero of $g(\cdot)$, i.e., that $g(\hat{z}) = 0$. Then, since $a'(\hat{z})y = 0$ for all $y \in \mathbb{R}^n$ (by (7) or (11)), we find that

$$12 \quad a'(\hat{z}) = 0 \quad \text{for all } \hat{z} \text{ such that } g(\hat{z}) = 0.$$

Proceeding as above, it is not difficult to show that $a''(\cdot)(\cdot)$, the second derivative of $a(\cdot)$ (as defined by (B.1.7)), exists and is continuous if the function $g(\cdot)$ is three times continuously differentiable, or, to state this in simpler language, if the elements $g^i(\cdot)$, $i = 1, 2, \dots, n$, of $g(\cdot)$, are three times continuously differentiable. Assuming that $a''(\cdot)(\cdot)$ exists and is continuous, and that the sequence $\{z_i\}$ constructed by algorithm (1) converges to a point \hat{z} , which as we have already shown must satisfy $g(\hat{z}) = 0$, we obtain from the Taylor formula for second-order expansions, that

$$13 \quad \|a(z_i) - a(\hat{z}) - a'(\hat{z})(z_i - \hat{z})\| \leq (\sup_{\xi \in [z_i, \hat{z}]} \|\frac{1}{2}a''(\xi)\|) \|z_i - \hat{z}\|^2,$$

where $\|a''(\xi)\|$, the norm of the operator $a''(\xi)(\cdot)$ is defined in (B.1.11) and $[z_i, \hat{z}] = \{\xi = \lambda z_i + (1 - \lambda)\hat{z} \mid \lambda \in [0, 1]\}$. In obtaining (13), we have made use of the inequality, $\|a''(z)(y)\| \leq \|a''(z)\| \|y\|$, which follows directly from definition (B.1.11). (The reader will recall that $\|a''(z)(y)\| = \max\{\|a''(z)(y)\|, \|a''(z)(y')\| \mid \|y'\| \leq 1\}$.)

Hence, since $a'(\hat{z}) = 0$, since $\hat{z} = a(\hat{z})$, and since $z_{i+1} = a(z_i)$ for $i = 0, 1, 2, \dots$, we obtain that

$$14 \quad \|z_{i+1} - \hat{z}\| \leq \frac{1}{2} (\sup_{\xi \in [z_i, \hat{z}]} \|a''(\xi)\|) \|z_i - \hat{z}\|^2, \quad \text{for } i = 0, 1, 2, \dots.$$

Since we have assumed $a''(\cdot)(\cdot)$ to be continuous, there must exist an $\tilde{M} < \infty$ such that for all $\xi \in \{\xi \mid \|\xi\| \leq \max_i \|z_i\|\}$, $\|a''(\xi)\| \leq 2\tilde{M}$. We now obtain from (14) that

$$15 \quad \|z_{i+1} - \hat{z}\| \leq \tilde{M} \|z_i - \hat{z}\|^2, \quad i = 0, 1, 2, \dots, *$$

and hence, that

$$16 \quad \|z_i - \hat{z}\| \leq \tilde{M}^{2^{i-1}} \|z_0 - \hat{z}\|^{2^i} = \frac{1}{\tilde{M}} (\tilde{M} \|z_0 - \hat{z}\|)^{2^i}, \quad i = 0, 1, 2, \dots.$$

Assuming that $\tilde{M} \|z_0 - \hat{z}\| < 1$, we find that $\|z_i - \hat{z}\| \rightarrow 0$ faster than any geometric progression $E\theta^i$, with $\theta \in (0, 1]$, $i = 0, 1, 2, \dots$, i.e., that

* At the expense of some additional work, it can be shown that (15) is also true under the weaker assumption that $g(\cdot)$ has a bounded and continuous second derivative. See Isaacs and Keller [11], pp. 115–119.

$\|z_i - \hat{z}\|/\theta^i \rightarrow 0$ as $i \rightarrow \infty$, for all $\theta \in (0, 1]$. Thus, $z_i \rightarrow \hat{z}$ at a rate that is *superlinear*. Because of (15), the rate at which z_i converges to \hat{z} is usually referred to as *quadratic*.

It is possible to show that the Newton-Raphson method (2.1.39) also converges quadratically under a somewhat different set of assumptions than the ones we have used so far. We shall now establish this fact. Parenthetically, the reader will need to be familiar with this result if he is to follow the proof of the rate of convergence for some of the conjugate gradient methods which will be discussed in the next section. We obtain quadratic convergence for (1) as a trivial corollary of the following theorem:

- 17 **Theorem.** Let $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ be a three times continuously differentiable function, let $g(\cdot) = \nabla f^0(\cdot)$, and suppose that there exist constants m and M , $0 < m \leq M < \infty$, such that

$$18 \quad m \|y\|^2 \leq \langle y, g'(z)y \rangle \leq M \|y\|^2 \quad \text{for all } y, z \text{ in } \mathbb{R}^n,$$

where $g'(z) = \partial g(z)/\partial z$. If $\hat{z} \in \mathbb{R}^n$ is such that $g(\hat{z}) = 0$, then there exist a $q \in (0, \infty)$ and a $\rho > 0$, such that

$$19 \quad \|a(z) - \hat{z}\| \leq q \|z - \hat{z}\|^2 \quad \text{for all } z \in B(\hat{z}, \rho),$$

where $B(\hat{z}, \rho) = \{z \mid \|z - \hat{z}\| \leq \rho\}$, and $a(\cdot)$ is defined as in (2).

Proof. First, since $g''(\cdot)(\cdot)$ is continuous in z , by assumption, there exists a bound $b < \infty$ such that $\|g''(z)\| \leq b$ for all $z \in B(\hat{z}, 1)$ (where $g''(\cdot)(\cdot)$ and its norm are defined in (B.1.7) and (B.1.11), respectively). Next, since $\|g'(z)^{-1} g(z)\| \leq \|g'(z)^{-1}\| \|g(z)\| \leq (1/m) \|g(z)\|$,* because of (18), and since $g(\hat{z}) = 0$, it follows that there exists a $\rho \in (0, 1/2)$ such that $\|g(z)\|/m \leq 1/2$ for all $z \in B(\hat{z}, \rho)$, and hence

$$20 \quad \begin{aligned} \|a(z) - \hat{z}\| &= \|z - \hat{z} - g'(z)^{-1} g(z)\| \\ &\leq \|z - \hat{z}\| + \|g'(z)^{-1} g(z)\| \leq \rho + \frac{1}{2} \leq 1, \end{aligned}$$

i.e., $a(z) \in B(\hat{z}, 1)$ for all $z \in B(\hat{z}, \rho)$.

Now, making use of the Taylor formula for second-order expansions (B.1.12), we find that

$$21 \quad \begin{aligned} g(a(z)) &= g(z - g'(z)^{-1} g(z)) \\ &= g(z) - g'(z)[g'(z)^{-1} g(z)] \\ &\quad + \int_0^1 (1-t) g''(z - tg'(z)^{-1} g(z))(g'(z)^{-1} g(z)) g'(z)^{-1} g(z) dt, \end{aligned}$$

* Since $g'(z)$ is a Hessian matrix, it is symmetric. It now follows from (18) that $\|g'(z)\| \leq M$ and $\|g'(z)^{-1}\| \leq 1/m$ for all $z \in \mathbb{R}^n$, where $\|g'(z)\| = \max\{\|g'(z)y\| \mid \|y\| \leq 1\}$, $\|g'(z)^{-1}\| = \max\{\|g'(z)^{-1}y\| \mid \|y\| \leq 1\}$.

and hence, for all $z \in B(\hat{z}, \rho)$,

$$\begin{aligned} 22 \quad \|g(a(z))\| &\leqslant \left[\int_0^1 (1-t) \|g''(z - tg'(z)^{-1} g(z))\| dt \right] \|g'(z)^{-1} g(z)\|^2 \\ &\leqslant \left[\int_0^1 (1-t) b dt \right] \frac{\|g(z)\|^2}{m^2} \\ &= \frac{b \|g(z)\|^2}{2m^2}, \end{aligned}$$

where we have again made use of the inequality $\|g(z)^{-1}\| \leqslant 1/m$.

Making use of Taylor's formula for first-order expansions (B.1.3), we obtain

$$\begin{aligned} 23 \quad \|g(z)\| &= \left\| g(\hat{z}) + \int_0^1 g'(\hat{z} + t(z - \hat{z})) dt (z - \hat{z}) \right\| \\ &\leqslant \int_0^1 \|g'(\hat{z} + t(z - \hat{z}))\| dt \|z - \hat{z}\| \leqslant M \|z - \hat{z}\|. \end{aligned}$$

Combining (22) with (23), we now obtain

$$24 \quad \|g(a(z))\| \leqslant \frac{bM^2}{2m^2} \|z - \hat{z}\|^2 \quad \text{for all } z \in B(\hat{z}, \rho).$$

Now, making use of Taylor's formula for first-order expansions (B.1.3), we obtain

$$25 \quad g(a(z)) = g(\hat{z}) + \int_0^1 g'(\hat{z} + t(a(z) - \hat{z})) dt [a(z) - \hat{z}].$$

Consequently (since $g(\hat{z}) = 0$), because of (18),

$$\begin{aligned} 26 \quad \langle a(z) - \hat{z}, g(a(z)) \rangle &= \int_0^1 \langle [a(z) - \hat{z}], g'(\hat{z} + t(a(z) - \hat{z})) [a(z) - \hat{z}] \rangle dt \\ &\geqslant m \|a(z) - \hat{z}\|^2, \end{aligned}$$

and hence

$$27 \quad m \|a(z) - \hat{z}\|^2 \leqslant \|a(z) - \hat{z}\| \|g(a(z))\|.$$

Now making use of (24), we obtain from (27) that

$$28 \quad \|a(z) - \hat{z}\| \leqslant \frac{bM^2}{2m^3} \|z - \hat{z}\|^2 \quad \text{for all } z \in B(\hat{z}, \rho),$$

which is the desired result. ■

We now turn to the quasi-Newton algorithm (2.1.42), which we restate below for the sake of convenience.

- 29 **Algorithm** (quasi-Newton). Minimizes a twice continuously differentiable function $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ with positive definite Hessian; Goldstein [G3].

Step 0. Select a $z_0 \in \mathbb{R}^n$, select an $\alpha \in (0, 1/2)$.

Step 1. Set $i = 0$.

Step 2. Compute $\nabla f^0(z_i)$.

Step 3. If $\nabla f^0(z_i) = 0$, stop; else, compute $H(z_i)$ and go to step 4.

Comment. As in Section 1, for all $z \in \mathbb{R}^n$, $H(z) = \partial^2 f^0(z)/\partial z^2$.

Step 4. Set $h(z_i) = -H(z_i)^{-1} \nabla f^0(z_i)$.

Step 5. If

$$\begin{aligned} 30 \quad -\lambda(1 - \alpha) \langle \nabla f^0(z_i), H(z_i)^{-1} \nabla f^0(z_i) \rangle &\leq f^0(z_i + \lambda h(z_i)) - f^0(z_i) \\ &\leq -\lambda \alpha \langle \nabla f^0(z_i), H(z_i)^{-1} \nabla f^0(z_i) \rangle \end{aligned}$$

is satisfied for $\lambda = 1$, set $\lambda_i = 1$ and go to step 6; else, compute a $\lambda_i > 0$ such that (30) is satisfied for $\lambda = \lambda_i$, and go to step 6.

Comment. Use procedure (2.1.33) to compute λ_i .

Step 6. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 2. ■

We shall now show that, under suitable assumptions, the quasi-Newton algorithm (29) will set $\lambda_i = 1$ for all i greater than some integer k , and hence that it has the same rate of convergence as the Newton-Raphson algorithm (1) (or, to be more precise, (2.1.39)). Note that the theorem below, which is due to Goldstein [G3], requires $f^0(\cdot)$ to be strictly convex, but only twice continuously differentiable. In obtaining the *quadratic* rate of convergence (15) for the Newton-Raphson algorithm (2.1.39), we had to assume that $f^0(\cdot)$ was four times continuously differentiable, but we did not need to assume that $f^0(\cdot)$ was strictly convex.* As we shall now see, when we assume that $f^0(\cdot)$ is only twice continuously differentiable, we cannot show that (2.1.39) converges quadratically, though we can still show that it converges *superlinearly*.

- 31 **Theorem** (Goldstein [G3]). Suppose that $f^0 : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is twice continuously differentiable and that there exist constants m and M , $0 < m \leq M$, such that

$$32 \quad m \|y\|^2 \leq \langle y, H(z)y \rangle \leq M \|y\|^2,$$

for all $z \in \mathbb{R}^n$ and for all $y \in \mathbb{R}^n$ (where, as before, $H(z) = \partial^2 f^0(z)/\partial z^2$). If $\{z_i\}_{i=0}^\infty$ is any sequence constructed by algorithm (29), then

- (i) the sequence $\{z_i\}$ converges to a point \hat{z} which minimizes $f^0(z)$ over $z \in \mathbb{R}^n$;
- (ii) there exists an integer k such that for all $i \geq k$, $\lambda_i = 1$;
- (iii) the convergence of $\{z_i\}$ to \hat{z} is superlinear, i.e., for any $\theta \in (0, 1]$, $\|z_i - \hat{z}\|/\theta^i \rightarrow 0$ as $i \rightarrow \infty$.

* Alternatively, from (17), we could have assumed that $f^0(\cdot)$ is three times continuously differentiable and that (18) holds.

Proof. We begin with (i). First, it follows from theorem (B.2.8) that the function $f^0(\cdot)$ is strictly convex and that the set $C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$ is compact. Hence, there exists a unique $\hat{z} \in C(z_0)$ such that $f^0(\hat{z}) \leq f^0(z)$ for all $z \in \mathbb{R}^n$. Next, since $C(z_0)$ is compact, the sequence $\{z_i\}$ must contain accumulation points. Since, by theorem (2.1.22), all the accumulation points z^* of $\{z_i\}$ satisfy $\nabla f^0(z^*) = 0$, and since \hat{z} is the only point in $C(z_0)$ with this property, we conclude that \hat{z} is the only accumulation point of $\{z_i\}$, i.e., that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$.

We shall now prove (ii). Let $\tilde{\Delta} : \mathbb{R}^n \times \mathbb{R}^1 \rightarrow \mathbb{R}^1$ be defined by

$$33 \quad \tilde{\Delta}(z, \lambda) = \frac{f^0(z - \lambda H(z)^{-1} \nabla f^0(z)) - f^0(z)}{-\lambda \langle \nabla f^0(z), H(z)^{-1} \nabla f^0(z) \rangle}.$$

Then, according to (30), for $\lambda = \lambda_i$,

$$34 \quad \tilde{\Delta}(z_i, \lambda_i) \in [\alpha, 1 - \alpha], \quad i = 0, 1, 2, \dots.$$

Applying the Taylor formula for second-order expansions (B.1.12) to (33), we obtain

$$35 \quad \tilde{\Delta}(z, \lambda) = 1 - \frac{\lambda}{2} \frac{\left\{ \begin{array}{l} 2 \int_0^1 [(1-t) \langle H(z)^{-1} \nabla f^0(z), \\ H(z - t\lambda H(z)^{-1} \nabla f^0(z)) H(z)^{-1} \nabla f^0(z) \rangle] dt \end{array} \right\}}{\langle \nabla f^0(z), H(z)^{-1} \nabla f^0(z) \rangle}$$

Now, since $H(\xi) = H(z) + H(\xi) - H(z)$, where $\xi = z + t\lambda H(z)^{-1} \nabla f^0(z)$, we obtain from (35) that

$$36 \quad \tilde{\Delta}(z, \lambda) = 1 - \frac{\lambda}{2} - \frac{\left\{ \begin{array}{l} \lambda \int_0^1 (1-t) \langle H(z)^{-1} \nabla f^0(z), \\ [H(z + t\lambda h(z)) - H(z)] H(z)^{-1} \nabla f^0(z) \rangle dt \end{array} \right\}}{\langle \nabla f^0(z), H(z)^{-1} \nabla f^0(z) \rangle}$$

where $h(z) = -H(z)^{-1} \nabla f^0(z)$. Setting $\lambda = 1$ and $z = z_i$, we now get

$$37 \quad \left| \tilde{\Delta}(z_i, 1) - \frac{1}{2} \right| \leq \frac{\|H(z_i)^{-1} \nabla f^0(z_i)\|^2 \int_0^1 (1-t) \|H(z_i + th(z_i)) - H(z_i)\| dt}{\|\nabla f^0(z_i)\|^2 / M} \\ \leq \frac{M}{m^2} \int_0^1 (1-t) \|H(z_i + th(z_i)) - H(z_i)\| dt.$$

Now let us consider the sequence $\{z_i\}$ constructed by algorithm (29). As we have already shown, the sequence $\{\nabla f^0(z_i)\}$ converges to the zero vector as $i \rightarrow \infty$, and since we have also shown that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$ and $H(\cdot)$ is continuous by assumption, we must have $H(z_i)^{-1} \nabla f^0(z_i) \rightarrow 0$ as $i \rightarrow \infty$. Consequently,

$$\sup_{t \in [0, 1]} \|H(z_i + th(z_i)) - H(z_i)\| \rightarrow 0 \quad \text{as } i \rightarrow \infty,$$

and therefore there must exist an integer $k \geq 0$ such that

$$38 \quad \frac{M}{m^2} \sup_{t \in [0,1]} \|H(z_i + th(z_i)) - H(z_i)\| \leq 1 - 2\alpha, \quad \text{for all } i \geq k.$$

Combining (37) and (38), we now obtain

$$39 \quad |\tilde{\Delta}(z_i, 1) - \frac{1}{2}| \leq \frac{1}{2} - \alpha, \quad \text{for all } i \geq k,$$

that is, for all $i \geq k$,

$$40 \quad \alpha \leq \tilde{\Delta}(z_i, 1) \leq 1 - \alpha,$$

and hence algorithm (29) will set $\lambda_i = 1$ for all $i \geq k$. This completes the proof of (ii).

To prove (iii), we assume that $i \geq k$. Then,

$$41 \quad z_{i+1} = z_i - H(z_i)^{-1} \nabla f^0(z_i), \quad i = k, k+1, k+2, \dots$$

By the Taylor formula for first-order expansions (B.1.3),

$$42 \quad \nabla f^0(z_i) = \nabla f^0(\hat{z}) + \left(\int_0^1 H(\hat{z} + t(z_i - \hat{z})) dt \right) (z_i - \hat{z}), \\ i = k, k+1, k+2, \dots$$

Substituting from (42) into (41), we now obtain (since $\nabla f^0(\hat{z}) = 0$)

$$43 \quad z_{i+1} - \hat{z} = \left(I - H(z_i)^{-1} \int_0^1 [H(z_i) + \bar{H}(z_i, t)] dt \right) (z_i - \hat{z}), \\ i = k, k+1, k+2, \dots$$

where $\bar{H}(z_i, t) = \bar{H}(\hat{z} + t(z_i - \hat{z})) - H(z_i)$. Hence,

$$44 \quad \|z_{i+1} - \hat{z}\| \leq \sup_{t \in [0,1]} \|H(z_i)^{-1} \bar{H}(z_i, t)\| \|z_i - \hat{z}\| \\ \leq \frac{1}{m} \sup_{t \in [0,1]} \|\bar{H}(z_i, t)\| \|z_i - \hat{z}\|, \quad i = k, k+1, k+2, \dots$$

Since $\sup_{t \in [0,1]} \|\bar{H}(z_i, t)\| \rightarrow 0$ as $i \rightarrow \infty$, because $z_i \rightarrow \hat{z}$ and $h(z_i) \rightarrow 0$ as $i \rightarrow \infty$, we find that $\|z_i - \hat{z}\| \rightarrow 0$ as $i \rightarrow \infty$ faster than any geometric progression $E\theta^i$, $i = k, k+1, k+2, \dots$, $\theta \in (0, 1]$. ■

The last two theorems may lead us to believe that the more the function $f^0(\cdot)$ is differentiable, the faster the Newton-Raphson algorithm (2.1.39) and hence also (2.1.42) will converge. Unfortunately, this is not true. It can easily be shown that even if we assume $f^0(\cdot)$ to be infinitely differentiable, the quadratic convergence bound given by (19) cannot be improved.

This concludes our discussion of the rate of convergence of the Newton-Raphson and quasi-Newton methods (2.1.39), (3.1.9) and (2.1.42), under various differentiability assumptions on the functions $g(\cdot)$ and $\nabla f^0(\cdot)$, whose roots these algorithms are designed to find.

6.3 Superlinear Convergence: Conjugate Gradient Methods

We shall now derive bounds on the rate of convergence for two conjugate gradient algorithms with reinitialization, which were briefly mentioned under the subheading of computational aspects in Section 2.3. These bounds on the rate of convergence were communicated to the author by Cohen [C4] (see also [D1a]).

We recall that these algorithms are intended for solving the problem,

$$1 \quad \min\{f^0(z) \mid z \in \mathbb{R}^n\}.$$

- 2 **Assumption.** We shall suppose that $f^0(\cdot)$ is a three times continuously differentiable function whose Hessian $\partial^2 f^0(z)/\partial z^2$, which we shall denote by $H(z)$ as before, satisfies

$$3 \quad m \|y\|^2 \leq \langle y, H(z)y \rangle \leq M \|y\|^2, \quad \text{for all } y, z \in \mathbb{R}^n,$$

where $0 < m \leq M < \infty$. ■

We begin with a modification of the Polak-Ribière algorithm (2.3.51), which we state below.

- 4 **Algorithm** (Polak-Ribière [P5]).

Step 0. Select a reinitialization integer $p \geq n$ and a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$ and set $g_0 = h_0 = -\nabla f^0(z_0)$.

Step 2. Compute $\lambda_i \geq 0$ such that

$$5 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 3. Set

$$6 \quad z_{i+1} = z_i + \lambda_i h_i.$$

Step 4. Compute $\nabla f^0(z_i)$.

Step 5. If $\nabla f^0(z_i) = 0$, stop; else, set

$$7 \quad g_{i+1} = -\nabla f^0(z_{i+1}),$$

$$8 \quad h_{i+1} = g_{i+1} + \gamma_i h_i, \quad \text{with } \gamma_i = \omega\left(\frac{i+1}{p}\right) \frac{\langle g_{i+1} - g_i, g_{i+1} \rangle}{\langle g_i, g_i \rangle}, *$$

set $i = i + 1$, and go to step 2. ■

* We define $\omega : \mathbb{R}^1 \rightarrow \mathbb{R}^1$ as follows: $\omega(x) = 0$ when x is an integer and $\omega(x) = 1$ otherwise.

- 9 Theorem** (Cohen [C4]). If $\{z_i\}_{i=0}^{\infty}$ is a sequence constructed by algorithm (4) for problem (1) (with assumption (2)), then $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, where $f^0(\hat{z}) \leq f^0(z)$ for all $z \in \mathbb{R}^n$, and there exists an integer $k \geq 0$ and a constant $q \in (0, \infty)$ such that

$$10 \quad \|z_{i+n} - \hat{z}\| \leq q \|z_i - \hat{z}\|^2 \quad \text{for all } i \in J_p, i \geq k,$$

where $J_p = \lambda \{0, p, 2p, 3p, \dots\}$.

Proof. Suppose that we augment algorithm (4) as follows. We set $j = 0$ in step 1 and insert Step 2': "If $\omega(i) = 0$, set $y_j = z_i$, set $j = j + 1$ and go to step 3; else, go to step 3." Then, because of (3), (B.2.8), (2.1.6) and the fact that $f^0(z_{i+1}) < f^0(z_i)$ for $i = 0, 1, 2, \dots$, where $\{z_i\}_{i=0}^{\infty}$ is constructed by (4), we conclude that $y_j \rightarrow \hat{z}$ as $j \rightarrow \infty$, where \hat{z} is the unique minimizer of $f^0(\cdot)$. It now follows from the fact that $f^0(z_{i+1}) < f^0(z_i)$ for $i = 0, 1, 2, \dots$ and the continuity of $f^0(\cdot)$ that $f^0(z_i) \rightarrow f^0(\hat{z})$ as $i \rightarrow \infty$, and hence, since the minimizer \hat{z} is unique, that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$.

We shall establish (10) by comparing algorithm (4) with the Newton-Raphson method (2.1.39).

At each point z_i of the sequence $\{z_i\}$ $i \in J_p$ we define the function $f_i^0(\cdot)$ by

$$11 \quad f_i^0(z) = f^0(z_i) + \langle \nabla f^0(z_i), z - z_i \rangle + \frac{1}{2} \langle z - z_i, H(z_i)(z - z_i) \rangle.$$

Now, for every $i \in J_p$, let $\overset{0}{z}_i = z_i$, and let $\overset{1}{z}_i, \overset{2}{z}_i, \dots, \overset{n}{z}_i$ be the corresponding sequence constructed by algorithm (4) when applied to the function $f_i^0(\cdot)$. (As we have done many times before, we have introduced superscripts to denote elements of a sequence in order to avoid confusion.) It now follows from theorem (2.3.47) and the fact that, for quadratic functions, algorithms (2.3.42) and (2.3.51) are identical, that $\overset{n}{z}_i$ must minimize $f_i^0(z)$ over $z \in \mathbb{R}^n$. Hence, we must have

$$12 \quad \overset{n}{z}_i = z_i - H(z_i)^{-1} \nabla f^0(z_i), \quad i \in J_p$$

i.e., $\overset{n}{z}_i$ is the successor to z_i constructed by the Newton-Raphson method (2.1.39) when applied to problem (1).

We shall show that (10) is satisfied if there exist constants $\overset{j}{q}' < \infty$ such that

$$13 \quad \varlimsup_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|(z_{i+j+1} - z_{i+j}) - (\overset{j+1}{z}_i - \overset{j}{z}_i)\|}{\|z_i - \hat{z}\|^2} \leq \overset{j}{q}', \quad \text{for } j = 0, 1, 2, \dots, n-1.$$

Note that (10) is true if and only if there exists a $q'' \in (0, \infty)$ such that

$$14 \quad \varlimsup_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - \hat{z}\|}{\|z_i - \hat{z}\|^2} \leq q''.$$

For the purposes of our proof, (14) is a more convenient statement than (10). Now,

$$15 \quad \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - \hat{z}\|}{\|z_i - \hat{z}\|^2} \leq \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - z_i^n\| + \|z_i^n - \hat{z}\|}{\|z_i - \hat{z}\|^2}$$

But, because of theorem (2.17), there exists a $\tilde{q} \in (0, \infty)$ such that

$$16 \quad \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_i^n - \hat{z}\|}{\|z_i - \hat{z}\|^2} = \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|a(z_i) - \hat{z}\|}{\|z_i - \hat{z}\|^2} \leq \tilde{q} < \infty,$$

where $a(\cdot)$ is the Newton-Raphson iteration function defined by (2.2), with $g(\cdot) = \nabla f^0(\cdot)$ (i.e., by (12)). Hence,

$$17 \quad \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - \hat{z}\|}{\|z_i - \hat{z}\|^2} \leq \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - z_i^n\|}{\|z_i - \hat{z}\|^2} + \tilde{q},$$

and therefore (14) is true if there exists a $\hat{q} \in (0, \infty)$ such that

$$18 \quad \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|z_{i+n} - z_i^n\|}{\|z_i - \hat{z}\|^2} \leq \hat{q}.$$

Now, for $i \in J_p$

$$19 \quad \begin{aligned} \|z_{i+n} - z_i^n\| &= \|(z_{i+n} - z_{i+n-1}) + (z_{i+n-1} - z_{i+n-2}) + \cdots + (z_{i+1} - z_i) \\ &\quad - (z_i^n - z_{i-1}^{n-1}) - (z_{i-1}^{n-1} - z_{i-2}^{n-2}) - \cdots - (z_i^1 - z_i^0)\| \\ &\leq \sum_{j=0}^{n-1} \|(z_{i+j+1} - z_{i+j}) - (z_i^{j+1} - z_i^j)\|. \end{aligned}$$

Consequently, if there exist constants $\hat{q}' \in (0, \infty)$ for which (13) holds, then there must exist a $\hat{q}'' \in (0, \infty)$ for which (14) holds.

We now proceed to show that (13) must hold. First we rewrite (13) into the form,*

$$20 \quad \overline{\lim}_{\substack{i \rightarrow \infty \\ i \in J_p}} \frac{\|\lambda_{i+j} h_{i+j} - \lambda_i h_i\|}{\|z_i - \hat{z}\|^2} \leq \hat{q}' \quad \text{for } j = 0, 1, 2, \dots, n-1,$$

where $\lambda_i h_i = z_i^{j+1} - z_i^j$, for $j = 0, 1, 2, \dots, n-1$. To establish (20) we shall need the following results, most of which have already been established in Section 2.3:

$$21 \quad \langle g_{i+1}, h_i \rangle = 0, \quad i = 0, 1, 2, \dots,$$

$$22 \quad \langle g_i, h_i \rangle = \langle g_i, g_i \rangle, \quad i = 0, 1, 2, \dots,$$

* In view of (1.25), we can simplify the proof considerably by assuming, as we do, without loss of generality, that $\hat{g}_i \neq 0$ for $j = 0, 1, \dots, n-1$ and all $i \in J_p$.

$$23 \quad \lambda_i = \frac{\langle g_i, g_i \rangle}{\langle h_i, H_i h_i \rangle} \leq \frac{1}{m}, \quad i = 0, 1, 2, \dots,$$

where, for $i = 0, 1, 2, \dots$,

$$24 \quad H_i = \int_0^1 H(z_i + t\lambda_i h_i) dt,$$

$$25 \quad \| h_{i+1} \|^2 = \| g_{i+1} \|^2 + \gamma_i^2 \| h_i \|^2,$$

$$26 \quad \| h_i \| \geq \| g_i \|,$$

$$27 \quad \| g_{i+1} \| \leq \left(1 + \frac{M}{m}\right) \| h_i \|,$$

$$28 \quad \gamma_i = -\omega(i+1) \frac{\langle g_{i+1}, H_i h_i \rangle}{\langle h_i, H_i h_i \rangle},$$

$$29 \quad |\gamma_i| \| h_i \| \leq \frac{M}{m} \| g_{i+1} \|,$$

$$30 \quad |\gamma_i| \leq \left(1 + \frac{M}{m}\right) \frac{M}{m},$$

$$31 \quad \| h_i \| \leq \left(1 + \frac{M}{m}\right) \| g_i \|,$$

$$32 \quad \| h_{i+1} \| \leq \left(1 + \frac{M}{m}\right)^2 \| h_i \|.$$

Relation (21) follows directly from (5); (22) was established in (2.3.65); (23) follows from (2.3.59), (3) and (26); (25) follows from (8) and (21); (26) follows from (25); (27) follows from (26) and the fact that $\| g_{i+1} \| \leq \| g_{i+1} - g_i \| + \| g_i \| \leq \lambda_i \| H_i h_i \| + \| h_i \|$ (see (2.3.58)); (28) was established in (2.3.61); (29) was established in (2.3.62); (30) follows from (29) and (27); (31) was established in (2.3.64); and (32) follows from (31) and (27).

Note that since $g_i \rightarrow 0$ as $i \rightarrow \infty$, it follows from (31) that $h_i \rightarrow 0$ as $i \rightarrow \infty$. We shall now show that there exists an integer k and constants $\hat{c}_3 < \infty$, $j = 0, 1, 2, \dots, n-1$, such that for all $i \in J_p$, $i \geq k$,

$$33 \quad \|\lambda_{i+j} h_{i+j} - \hat{\lambda}_j h_i\| \leq \hat{c}_3 \|h_i\|^2, \quad \text{for } j = 0, 1, \dots, n-1.$$

Since

$$34 \quad \|g_i\| \leq M \|z_i - \hat{z}\|, \quad i = 0, 1, 2, \dots,$$

because $g_i = \int_0^1 H(z + t(z_i - \hat{z})) dt (z_i - \hat{z})$ and because of (3), we find that (33) together with (34) and (31) imply the existence of a constant q' for

which (20) holds. We shall give a proof by induction, for the purpose of which we need to show that, for $j = 0, 1, 2, \dots, n - 1$, and $i \in J_p$

$$35 \quad \|g_{i+j} - \overset{j}{g}_i\| \leq O_1(\|h_i\|^2),$$

$$36 \quad \|h_{i+j} - \overset{j}{h}_i\| \leq O_2(\|h_i\|^2),$$

$$37 \quad \|\lambda_{i+j} h_{i+j} - \overset{j}{\lambda}_i \overset{j}{h}_i\| \leq O_3(\|h_i\|^2),$$

where $\overset{j}{g}_i = -\nabla f^0(\overset{j}{z}_i)$, for $j = 0, 1, 2, \dots, n - 1$ and $i \in J_p$, and $O_p : \mathbb{R}^1 \rightarrow \mathbb{R}^1$, $p = 1, 2, 3$, is a function with the property that $|O_p(x)| \leq \overset{j}{c}_p |x|$, for all $|x| \leq \overset{j}{r}_p$, for some $\overset{j}{c}_p < \infty$ and for some $\overset{j}{r}_p > 0$. Since $h_i \rightarrow 0$ as $i \rightarrow \infty$, it is clear that (37) implies (33) for i sufficiently large.

To establish (35)–(37), we shall need the following five lemmas:

Lemma 1.

$$38 \quad \|H(z_{i+j}) - H(z_i)\| \leq O(\|h_i\|), \quad j = 0, 1, \dots, n - 1, \quad i \in J_p,$$

where $|O(x)| \leq c|x|$ for all $|x| \leq r$, for some $c < \infty$ and for some $r > 0$.

Proof. First we note that

$$39 \quad \|H(z_{i+j}) - H(z_i)\| \leq \sum_{p=0}^{j-1} \|H(z_{i+p+1}) - H(z_{i+p})\|.$$

Next, with $g(\cdot) = -\nabla f^0(\cdot)$, we find that $H(z) = -g'(z)$, and hence, by the Taylor formula for first-order expansions (see (B.1.3)),

$$40 \quad H(z_{i+p+1}) - H(z_{i+p}) = - \int_0^1 g''(z_{i+p} + t\lambda_{i+p} h_{i+p})(\lambda_{i+p} h_{i+p}) dt,$$

where $g''(\cdot)(\cdot)$ is as defined in (B.1.11). Consequently,

$$41 \quad \|H(z_{i+p+1}) - H(z_{i+p})\| \leq \lambda_{i+p} \|h_{i+p}\| \int_0^1 \|g''(z_{i+p} + t\lambda_{i+p} h_{i+p})\| dt.$$

Since by assumption (2), $g''(\cdot)(\cdot)$ is continuous, and $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, we conclude that there exists a $b < \infty$ such that $\|g''(z_{i+p} + t\lambda_{i+p} h_{i+p})\| \leq b$ for all $t \in [0, 1]$, and $i = 0, 1, 2, \dots$, and hence, because of (23), that

$$42 \quad \|H(z_{i+p+1}) - H(z_{i+p})\| \leq \frac{b \|h_{i+p}\|}{m}.$$

Combining (42) with (32), we now obtain (38).

Lemma 2.

$$43 \quad \|H_{i+j} - H(z_i)\| = \tilde{O}(\|h_i\|), \quad j = 0, 1, 2, \dots, n-1, \quad i = 0, 1, 2, \dots,$$

where $|\tilde{O}(x)| \leq \tilde{c}|x|$ for all $|x| \leq \tilde{r}$, for some $\tilde{c} < \infty$ and for some $\tilde{r} > 0$.

Proof. First we note that

$$44 \quad \|H_{i+j} - H(z_i)\| \leq \|H_{i+j} - H(z_{i+j})\| + \|H(z_{i+j}) - H(z_i)\|.$$

Now (see (24)),

$$45 \quad \|H_{i+j} - H(z_{i+j})\| \leq \int_0^1 \|H(z_{i+j} + t\lambda_{i+j}h_{i+j}) - H(z_{i+j})\| dt.$$

Proceeding as we did in obtaining (42) from (41), we conclude that

$$46 \quad \|H_{i+j} - H(z_{i+j})\| \leq \frac{b\|h_{i+j}\|}{2m}.$$

Making use of (38) and of (32), we obtain (43) from (46).

Lemma 3. For $j = 0, 1, \dots, n-1$ and all $i \in J_p$,

$$47 \quad \|h_{i+j+1} - \overset{j+1}{h}_i\| = \tilde{O}_1(\|h_{i+j} - \overset{j}{h}_i\|) + \tilde{O}_2(\|g_{i+j+1} - \overset{j+1}{g}_i\|) + \tilde{O}_3(\|h_i\|^2),$$

where $|\tilde{O}_p(x)| \leq \tilde{c}_p|x|$, $p = 1, 2, 3$, for all $|x| \leq \tilde{r}_p$, for some $\tilde{c}_p < \infty$, for some $\tilde{r}_p > 0$.

Proof. First, suppose that $p = n$ and $j = n-1$. Then $h_{i+n} = g_{i+n}$ and $\overset{n}{h}_i = \overset{n}{g}_i$ for all $i \in J_p$, and hence, $\|h_{i+n} - \overset{n}{h}_i\| = \|g_{i+n} - \overset{n}{g}_i\|$. Therefore, suppose that either $p > n$ and $0 \leq j \leq n-1$, or $p = n$ and $0 \leq j \leq n-2$. Next, recall that

$$\begin{aligned} \|h_{i+j+1} - \overset{j+1}{h}_i\| &= \|g_{i+j+1} - \gamma_{i+j}h_{i+j} - \overset{j+1}{g}_i - \gamma_i^j h_i\| \\ &\leq \|g_{i+j+1} - \overset{j+1}{g}_i\| + \|\gamma_{i+j}h_{i+j} - \gamma_i^j h_i\|. \end{aligned}$$

Now making use of (28), we obtain

$$48 \quad \|\gamma_{i+j}h_{i+j} - \gamma_i^j h_i\| = \left\| \frac{\langle g_{i+j+1}, H_{i+j}h_{i+j} \rangle}{\langle h_{i+j}, H_{i+j}h_{i+j} \rangle} h_{i+j} - \frac{\langle \overset{j+1}{g}_i, H(z_i) \overset{j}{h}_i \rangle}{\langle \overset{j}{h}_i, H(z_i) \overset{j}{h}_i \rangle} \overset{j}{h}_i \right\|.$$

Let $c_{i+j} = \langle h_{i+j}, H_{i+j}h_{i+j} \rangle \langle h_i, H(z_i) h_i \rangle$, $j = 0, 1, 2, \dots, n-1$, $i \in J_p$. Then, from (48),

$$\begin{aligned} 49 \quad & \| \gamma_{i+j} h_{i+j} - \gamma_i^j h_i \| = \frac{1}{c_{i+j}} \| \langle g_{i+j+1}, H_{i+j}h_{i+j} \rangle \langle h_i, H(z_i) h_i \rangle h_{i+j} \\ & \quad - \langle g_i^{j+1}, H(z_i) h_i \rangle \langle h_{i+j}, H_{i+j}h_{i+j} \rangle h_i \| \\ & \leq \frac{1}{c_{i+j}} \| \langle g_{i+j+1}, H_{i+j}(h_{i+j} - h_i) \rangle \langle h_i, H(z_i) h_i \rangle h_{i+j} \| \\ & \quad + \| \langle g_{i+j+1}, H_{i+j}h_i \rangle \langle h_i - h_{i+j}, H(z_i) h_i \rangle h_{i+j} \| \\ & \quad + \| \langle g_{i+j+1} - g_i^{j+1}, H_{i+j}h_i \rangle \langle h_{i+j}, H(z_i) h_i \rangle h_{i+j} \| \\ & \quad + \| \langle g_i^{j+1}, H_{i+j}h_i \rangle \langle h_{i+j}, H(z_i)(h_i - h_{i+j}) \rangle h_{i+j} \| \\ & \quad + \| \langle g_i^{j+1}, (H_{i+j} - H(z_i)) h_i \rangle \langle h_{i+j}, H(z_i) h_{i+j} \rangle h_{i+j} \| \\ & \quad + \| \langle g_i^{j+1}, H(z_i) h_i \rangle \langle h_{i+j}, (H(z_i) - H_{i+j}) h_{i+j} \rangle h_{i+j} \| \\ & \quad + \| \langle g_i^{j+1}, H(z_i) h_i \rangle \langle h_{i+j}, H_{i+j}h_{i+j} \rangle (h_{i+j} - h_i) \| \}. \end{aligned}$$

Since because of (2),

$$50 \quad \frac{1}{c_{i+j}} \leq \frac{1}{m^2 \| h_{i+j} \|^2 \| h_i \|^2},$$

we can make use of (43), (49) and (27) to obtain (47). (Note that $\| g_i^{j+1} \| \leq (1 + (M/m)) \| h_i \|$ can be obtained in the same manner as (27).)

Lemma 4.

$$51 \quad \| g_{i+j+1} - g_i^{j+1} \| \leq \| g_{i+j} - g_i^j \| + O'(\| h_i \|^2) + M \| \lambda_{i+j} h_{i+j} - \lambda_i^j h_i \|, \\ j = 0, 1, 2, \dots, n-1, \quad i \in J_p,$$

where $|O'(x)| \leq c' |x|$ for all $|x| \leq r'$, for some $c' < \infty$, for some $r' > 0$.

Proof. By (2.3.58) (and making use of (23), (3) and (43)), we obtain

$$\begin{aligned} 52 \quad & \| g_{i+j+1} - g_i^{j+1} \| \\ & = \| g_{i+j} - \lambda_{i+j} H_{i+j} h_{i+j} - g_i^j + \lambda_i^j H(z_i) h_i \| \\ & \leq \| g_{i+j} - g_i^j \| + \| (H(z_i) - H_{i+j}) \lambda_i^j h_i \| + \| H_{i+j} (\lambda_i^j h_i - \lambda_{i+j} h_{i+j}) \| \\ & \leq \| g_{i+j} - g_i^j \| + \frac{1}{m} \| H(z_i) - H_{i+j} \| \| h_i \| + M \| \lambda_i^j h_i - \lambda_{i+j} h_{i+j} \| \\ & \leq \| g_{i+j} - g_i^j \| + \frac{1}{m} O(\| h_i \|) \| h_i \| + M \| \lambda_i^j h_i - \lambda_{i+j} h_{i+j} \|. \end{aligned}$$

Since an equivalent form of (32) is also valid for the vectors $\overset{j}{h}_i$, $j = 0, 1, 2, \dots, n - 1$, and since, by construction, $\overset{0}{h}_i = h_i$, (51) now follows directly from the last inequality in (52).

Lemma 5. (For $j = 0, 1, \dots, n - 1$ and $i \in J_p$.

$$53 \quad \|\lambda_{i+j}h_{i+j} - \overset{j}{\lambda}_i h_i\| \leq O_1'(\|g_{i+j} - \overset{j}{g}_i\|) + O_2'(\|h_{i+j} - \overset{j}{h}_i\|) + O_3'(\|h_i\|^2),$$

where $|O_p'(x)| \leq c_p' |x|$ for all $|x| \leq r_p'$, $p = 1, 2, 3$, for some $c_p' < \infty$, for some $r_p' > 0$.

Proof. Defining c_{i+j} as in (49), we obtain

$$\begin{aligned} 54 \quad \|\lambda_{i+j}h_{i+j} - \overset{j}{\lambda}_i h_i\| &= \left\| \frac{\|g_{i+j}\|^2}{\langle h_{i+j}, H_{i+j}h_{i+j} \rangle} h_{i+j} - \frac{\|g_i\|^2}{\langle h_i, H(z_i)h_i \rangle} \overset{j}{h}_i \right\| \\ &= \frac{1}{c_{i+j}} \{ \|g_{i+j}\|^2 \langle \overset{j}{h}_i, H(z_i) \overset{j}{h}_i \rangle h_{i+j} \\ &\quad - \|g_i\|^2 \langle h_{i+j}, H_{i+j}h_{i+j} \rangle \overset{j}{h}_i \| \} \\ &\leq \frac{1}{c_{i+j}} \{ \| \langle g_{i+j}, g_{i+j} - \overset{j}{g}_i \rangle \langle \overset{j}{h}_i, H(z_i) \overset{j}{h}_i \rangle h_{i+j} \| \\ &\quad + \| \langle g_{i+j}, \overset{j}{g}_i \rangle \langle \overset{j}{h}_i - h_{i+j}, H(z_i) \overset{j}{h}_i \rangle h_{i+j} \| \\ &\quad + \| \langle g_{i+j} - \overset{j}{g}_i, \overset{j}{g}_i \rangle \langle h_{i+j}, H(z_i) \overset{j}{h}_i \rangle h_{i+j} \| \\ &\quad + \| \|g_i\|^2 \langle h_{i+j}, H(z_i)(h_i - h_{i+j}) \rangle h_{i+j} \| \\ &\quad + \| \|g_i\|^2 \langle h_{i+j}, H(z_i) h_{i+j} \rangle \langle \overset{j}{h}_i - h_{i+j} \rangle \| \\ &\quad + \| \|g_i\|^2 \langle h_{i+j}, (H(z_i) - H_{i+j}) h_{i+j} \rangle \overset{j}{h}_i \| \}. \end{aligned}$$

Now making use of (50), (26), (32) and (43), we conclude that (54) implies (53). (Note that (26) and (32) are also true when h_i is replaced by $\overset{j}{h}_i$, h_{i+1} by $\overset{j+1}{h}_i$ and g_i by $\overset{j}{g}_i$. Also, recall that $\overset{0}{h}_i = h_i$ for $i \in J_p$.)

With lemmas 1–5 established, we can now obtain (35)–(37) as follows: First, for $j = 0$, and any $i \in J_p$, we have

$$55 \quad \|g_i - \overset{0}{g}_i\| = 0,$$

$$56 \quad \|h_i - \overset{0}{h}_i\| = 0,$$

$$57 \quad \|\lambda_i h_i - \overset{0}{\lambda}_i \overset{0}{h}_i\| \leq O_3'(\|h_i\|^2),$$

where (55) holds because $\overset{0}{g}_i = g_i$, (56) holds because $\overset{0}{h}_i = h_i$ by construction,

and (57) holds because of (55), (56) and (53). Hence, (35)–(37) hold for $j = 0$ and any $i \in J_p$. Now let us suppose that (35)–(37) hold for any $j' \in \{0, 1, \dots, n - 2\}$, $i \in J_p$. We shall show that they must then hold for $j = j' + 1$. First consider (35). Replacing j' by $j' + 1$ in (35), we get, making use of (51),

$$58 \quad \|g_{i+j'+1} - g_i^{j'+1}\| \leq \|g_{i+j'} - g_i^{j'}\| + O'(\|h_i\|^2) + M \|\lambda_{i+j'} h_{i+j'} - \lambda_i h_i\|.$$

Now making use of (35) for $j = j'$ and of (37) for $j = j'$, we find that (35) must be true for $j = j' + 1$.

Next, according to (47), for $j = j'$,

$$59 \quad \|h_{i+j'+1} - h_i^{j'+1}\| \leq \tilde{O}_1(\|h_{i+j'} - h_i^{j'}\|) + \tilde{O}_2(\|g_{i+j'+1} - g_i^{j'+1}\|) + \tilde{O}_3(\|h_i\|^2).$$

Making use of (36) for $j = j'$ and of (35) for $j = j' + 1$, we find that (59) implies that (36) holds for $j = j' + 1$.

Finally, setting $j = j'$ in (53), and making use of (35) and (36) for $j = j'$, we find that (37) must be true for $j = j' + 1$.

Thus, (37) must be true for $j = 0, 1, 2, \dots, n - 1$ and for all $i \in J_p$. Now, as we have already pointed out, $h_i \rightarrow 0$ as $i \rightarrow \infty$. There must therefore exist an integer k such that $\|h_i\|^2 \leq \min\{r_p \mid p \in \{1, 2, 3\}, j \in \{0, 1, 2, \dots, n - 1\}\}$, and hence (33) must hold for all $i \in J_p$ greater than or equal to this k . This completes our proof. ■

In addition to the above very beautiful result, Cohen has also obtained a bound on the rate of convergence of the truncated version of the Fletcher-Reeves algorithm. This truncated algorithm has the following form:

60 **Algorithm** (Fletcher-Reeves [F4]; solves problem (1) under assumption (2)).

Step 0. Select a reinitialization integer $p \geq n$; select a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$; set $g_0 = h_0 = -\nabla f^0(z_0)$.

Step 2. Compute $\lambda_i \geq 0$ such that

$$61 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 3. Set

$$62 \quad z_{i+1} = z_i + \lambda_i h_i.$$

Step 4. Compute $\nabla f^0(z_{i+1})$.

Step 5. If $\nabla f^0(z_{i+1}) = 0$, stop; else, go to step 6.

Step 6. If p is a factor of i (i.e., if $i = pq_i$, with q_i an integer), go to step 7; else, go to step 8.

Step 7. Set

63 $g_{i+1} = -\nabla f^0(z_{i+1}), \quad h_{i+1} = -\nabla f^0(z_{i+1});$

set $i = i + 1$, and go to step 2.

Step 8. Set

64 $g_{i+1} = -\nabla f^0(z_{i+1}),$

65 $\gamma_i = \|g_{i+1}\|^2/\|g_i\|^2,$

66 $h_{i+1} = g_{i+1} + \gamma_i h_i;$

set $i = i + 1$ and go to step 2. ■

- 67 **Exercise.** Suppose that $\{z_i\}$ is a sequence constructed by algorithm (60) for problem (1), and suppose that assumption (2) is satisfied. Show that either $\{z_i\}$ is finite and its last element z_k satisfies $\nabla f^0(z_k) = 0$, or else $\{z_i\}$ is infinite, in which case it converges to a point \hat{z} such that $\nabla f^0(\hat{z}) = 0$. [Hint: Every $(p+1)$ th iteration, algorithm (60) constructs a point in exactly the same manner as the method of steepest descent.] ■

- 68 **Theorem** (Cohen [C4]). Suppose that $\{z_i\}$ is an infinite sequence constructed by algorithm (60) for problem (1) and suppose that assumption (2) is satisfied. Then there exists an integer k and a constant $q < \infty$ such that

69 $\|z_{i+n} - \hat{z}\| \leq q \|z_i - \hat{z}\|^2$

for all $i \in J_p$ such that $jp \geq k$, where \hat{z} is the limit point of the sequence $\{z_i\}$ and J_p is as in (10). ■

The proof of this theorem is given in [C4] and is almost exactly the same as the proof of theorem (9).

- 70 **Remark.** Suppose that we set $p = n$ in the algorithms (4) or (60), and that we consider only the subsequences $\{y_i\}$, with $y_i = z_{ip}$, of the sequences $\{z_i\}$ constructed either by algorithm (4) or by algorithm (60). Then, from (10) and (67), we see that $y_i \rightarrow \hat{z}$ as $i \rightarrow \infty$ quadratically, i.e., algorithms (4) and (60) take n iterations to approximate one iteration of the Newton-Raphson algorithm (2.1.39).

6.4 Superlinear Convergence: the Variable Metric Algorithm

We now return to the variable metric algorithm (2.3.68). The rate of convergence results we are about to present for this method were communicated to the author by Powell [P8]. We recall that the variable metric algorithm solves the problem,

2 Assumption. We shall suppose that $f^0(\cdot)$ is convex, three times continuously differentiable, and that there exists a constant $0 < m$ such that

$$3 \quad m \|y\|^2 \leq \langle y, H(z)y \rangle \quad \text{for all } y \in \mathbb{R}^n, \quad \text{for all } z \in C(z_0),$$

where $H(z) = \partial^2 f^0(z)/\partial z^2$, z_0 is the initial point to be used for solving (1) by means of algorithm (2.3.68), and $C(z_0) = \{z \mid f^0(z) \leq f^0(z_0)\}$. ■

4 Remark. It follows from (B.2.8) that the set $C(z_0)$ is compact, because $m > 0$ in (3). Hence, since $H(\cdot)$ is continuously differentiable by (2), there exist constants $M \geq m$ and $L > 0$ such that

$$5 \quad m \|y\|^2 \leq \langle y, H(z)y \rangle \leq M \|y\|^2 \quad \text{for all } y \in \mathbb{R}^n, \quad \text{for all } z \in C(z_0),$$

and

$$6 \quad \|H(z) - H(\hat{z})\| \leq L \|z - \hat{z}\|, \quad \text{for all } z \in C(z_0),$$

where \hat{z} is such that $f^0(\hat{z}) = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$. (Note that \hat{z} is unique because $f^0(\cdot)$ is strictly convex on $C(z_0)$, which must contain \hat{z} .)

To reduce the need for leafing back and forth, we now restate the variable metric algorithm.

7 Algorithm (variable metric; Davidon [D2], Fletcher and Powell [F3]).

Step 0. Select a $z_0 \in \mathbb{R}^n$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $i = 0$, set $H_0 = I$ (the $n \times n$ identity matrix), and set $g_0 = \nabla f^0(z_0)$.*

Comment. Note that both g_i and H_i are not defined in the same way here as they are in Section 3.

Step 2. Set

$$8 \quad h_i = -H_i g_i.$$

Step 3. Compute $\lambda_i \geq 0$ such that

$$9 \quad f^0(z_i + \lambda_i h_i) = \min\{f^0(z_i + \lambda h_i) \mid \lambda \geq 0\}.$$

Step 4. Compute $\nabla f^0(z_i + \lambda_i h_i)$.

Step 5. If $\nabla f^0(z_i + \lambda_i h_i) = 0$, stop; else, set

$$10 \quad z_{i+1} = z_i + \lambda_i h_i,$$

$$11 \quad g_{i+1} = \nabla f^0(z_{i+1}),$$

$$12 \quad \Delta g_i = g_{i+1} - g_i,$$

* The choice $H_0 = I$ is not mandatory. We may choose H_0 to be any symmetric, positive definite matrix.

$$13 \quad \Delta z_i = z_{i+1} - z_i,$$

$$14 \quad H_{i+1} = H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i \Delta g_i \langle H_i \Delta g_i \rangle + \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta z_i \langle \Delta z_i \rangle,$$

and go to step 6.*

Step 6. Set $i = i + 1$ and go to step 2. ■

We recall from (2.3.76) that the matrices H_i are symmetric and positive definite for all $i = 0, 1, 2, \dots$, and we recall from (2.3.106) that if $\{z_i\}$ is an infinite sequence constructed by algorithm (7) for problem (1), under assumption (2), then $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, where $f^0(\hat{z}) = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$.

We begin by showing that $f^0(z_i) \rightarrow f^0(\hat{z})$ as $i \rightarrow \infty$ at least as fast as a geometric progression.

- 15 **Theorem (Powell).** Suppose that assumption (2) is satisfied. If $\{z_i\}$ is an infinite sequence constructed by algorithm (7) for problem (1), then there exists a constant $q(z_0) \in (0, 1)$ such that

$$16 \quad f^0(z_i) - f^0(\hat{z}) \leq q(z_0)^i [f^0(z_0) - f^0(\hat{z})], \quad i = 0, 1, 2, \dots,$$

where \hat{z} is the limit point of $\{z_i\}$.

Proof. We shall make use of some of the facts established in the proof of theorem (2.3.106). Since we have shown that (2.3.129) cannot be true, we conclude from (2.3.117) that

$$17 \quad \sum_{j=0}^i \frac{\|g_{j+1}\|^2}{\langle g_{j+1}, H_j g_{j+1} \rangle} \leq w(i+1), \quad i = 0, 1, 2, \dots.$$

Therefore, applying an argument similar to the one which we had used to obtain (2.3.124), we conclude that for at least two-thirds of the integers $j \in \{0, 1, 2, \dots, i\}$, the inequality,

$$18 \quad \|g_{j+1}\|^2 \leq 3w \langle g_{j+1}, H_j g_{j+1} \rangle,$$

must be satisfied. Therefore, both the inequality (2.3.124) and (18) must be satisfied simultaneously for at least one-third of the integers $j \in \{0, 1, 2, \dots, i\}$, and hence, for these integers j , we must have

$$19 \quad \|g_{j+1}\|^2 < 9ww' \|\Delta g_j\|^2.$$

Making use of (2.3.97), we now obtain, for these integers j ,

$$20 \quad m[f^0(z_{j+1}) - f^0(\hat{z})] < 9ww' \|\Delta g_j\|^2.$$

Since by lemma (2.3.89), $\|\Delta g_j\|^2/\|\Delta z_j\|^2$ is bounded for all $j = 0, 1, 2, \dots$,

* See footnote on p. 56.

and because of the bound on $\|\Delta z_i\|$ given by (2.3.104), we conclude that there must exist a constant q' such that

$$21 \quad f^0(z_{i+1}) - f^0(\hat{z}) < q'[f^0(z_i) - f^0(z_{i+1})]$$

for all those $j \in \{0, 1, 2, \dots, i\}$ for which (20) holds. Thus, for at least one-third of the integers $j \in \{0, 1, 2, \dots, i\}$, we must have

$$22 \quad f^0(z_{j+1}) - f^0(\hat{z}) < \frac{q'}{q' + 1} [f^0(z_j) - f^0(\hat{z})].$$

Now, for the remaining values of j , we must have $[f^0(z_{j+1}) - f^0(\hat{z})] < [f^0(z_j) - f^0(\hat{z})]$, and hence, we find that (16) is satisfied for $q(z_0) = (q'/1 + q')^{1/3}$. ■

The following result is an important consequence of theorem (15):

23 **Corollary.** There exists a $b < \infty$ such that

$$24 \quad \sum_{i=0}^{\infty} \|\Delta z_i\| < b.$$

Proof. According to (2.3.104),

$$25 \quad \|\Delta z_i\|^2 \leq \frac{2[f^0(z_i) - f^0(z_{i+1})]}{m},$$

and since \hat{z} minimizes $f^0(z)$, we must have $[f^0(z_i) - f^0(z_{i+1})] \leq [f^0(z_i) - f^0(\hat{z})]$. We now conclude from (16) that

$$26 \quad \|\Delta z_i\| \leq \sqrt{q(z_0)^i} \left\{ \frac{2[f^0(z_0) - f^0(\hat{z})]}{m} \right\}^{1/2},$$

which shows that (24) must hold. ■

The two theorems to follow will make frequent use of the above corollary for the following reason: Since $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$,

$$27 \quad \hat{z} = z_i + \sum_{j=i}^{\infty} \Delta z_j.$$

Consequently, because of (24),

$$28 \quad \|z_i - \hat{z}\| \leq \sum_{j=i}^{\infty} \|\Delta z_j\| < b, \quad i = 0, 1, 2, \dots$$

We shall use the notation,

$$29 \quad \Delta_i = \sum_{j=i}^{\infty} \|\Delta z_j\|, \quad i = 0, 1, 2, \dots$$

- 30 **Lemma.** Suppose that assumption (2) is satisfied. If $\{z_i\}$ is an infinite sequence constructed by algorithm (7), converging to the point \hat{z} , then

$$31 \quad \|\Delta g_i - H(\hat{z}) \Delta z_i\| \leq L \Delta_i \|\Delta z_i\|, \quad i = 0, 1, 2, \dots,$$

where, as before, $H(z) = \partial^2 f^0(z)/\partial z^2$ and L is the constant in (6).

Proof. By the Taylor formula for first-order expansions (B.1.3),

$$32 \quad \Delta g_i = \int_0^1 H(z_i + t \Delta z_i) dt \Delta z_i.$$

Consequently,

$$33 \quad \begin{aligned} \|\Delta g_i - H(\hat{z}) \Delta z_i\| &= \left\| \int_0^1 [H(z_i + t \Delta z_i) - H(\hat{z})] dt \Delta z_i \right\| \\ &\leq L \|z_i + t \Delta z_i - \hat{z}\| \|\Delta z_i\| \leq L \Delta_i \|\Delta z_i\|, \quad i = 0, 1, 2, \dots, \end{aligned}$$

where we have made use of (6), (29), and the fact that for $t \in [0, 1]$, $\|z_i + t \Delta z_i - \hat{z}\| \leq \max\{\|z_i - \hat{z}\|, \|z_{i+1} - \hat{z}\|\} \leq \Delta_i$. ■

- 34 **Lemma.** Let P be any nonsingular $n \times n$ matrix and let $\tilde{f}^0(\cdot)$ be defined by

$$35 \quad \tilde{f}^0(z) = f^0(P^{-1}z) \quad \text{for all } z \in \mathbb{R}^n.$$

Suppose that z_0, z_1, z_2, \dots is a sequence constructed by algorithm (7) when applied to the solution of problem (1), and suppose that $\bar{z}_0, \bar{z}_1, \bar{z}_2, \dots$ is a sequence constructed by algorithm (7) when applied to the problem, $\min\{\tilde{f}^0(z) \mid z \in \mathbb{R}^n\}$ but with $\bar{H}_0 = P P^T$ in step 1. If $\bar{z}_0 = P z_0$, then, for $i = 0, 1, 2, \dots$,

$$36 \quad \tilde{f}^0(\bar{z}_i) = f^0(z_i),$$

$$37 \quad \bar{z}_i = P z_i,$$

$$38 \quad \bar{g}_i = (P^{-1})^T g_i,$$

and

$$39 \quad \bar{H}_i = P H_i P^T,$$

where the bars over the letters indicate the quantities constructed by algorithm (7) in the process of solving the problem, $\min\{\bar{f}^0(z) \mid z \in \mathbb{R}^n\}$.

Proof. Suppose that (37) is true; then (36) follows from (35) and so does (38). Consequently, we only need to prove (37) and (39). Note that (37) and (39) are true for $i = 0$. Now suppose that (37) and (39) are true for some integer $i \geq 0$; then a direct application of algorithm (7) shows that (37) and (39) are also valid for $i + 1$. Consequently, since they are valid for $i = 0$, (37) and (39) must hold for all $i = 0, 1, 2, \dots$. ■

- 40 **Theorem (Powell).** Suppose that assumption (2) is satisfied, and that H_0, H_1, H_2, \dots is an infinite sequence of matrices constructed by algorithm (7) in the process of solving problem (1). Then there exist constants \bar{m} and \bar{M} , $0 < \bar{m} \leq \bar{M} < \infty$, such that

$$41 \quad \bar{m} \|y\|^2 \leq \langle y, H_i y \rangle \leq \bar{M} \|y\|^2 \quad \text{for all } y \in \mathbb{R}^n, \quad i = 0, 1, 2, \dots$$

Proof. Since, according to (2.3.76), the matrices H_i are symmetric and positive definite, (41) holds if and only if the eigenvalues $\mu_i^j, j = 1, 2, \dots, n$, of the $n \times n$ matrices H_i satisfy

$$42 \quad \bar{m} \leq \mu_i^j \leq \bar{M}, \quad j = 1, 2, \dots, n, \quad i = 0, 1, 2, \dots$$

Also note that since H_i is symmetric and positive definite, its Euclidean norm $\|H_i\| = \max_j \mu_i^j$, and hence, (41) implies that the matrices H_i are uniformly bounded in norm.

Now, again, let $\hat{z} \in \mathbb{R}^n$ be such that $f^0(\hat{z}) = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$, and let $H(z) = \partial^2 f^0(z)/\partial z^2$. Referring to lemma (34), let $P = H(\hat{z})^{-1/2}$, and let us apply algorithm (7) to the problem, $\min\{\bar{f}^0(z) \mid z \in \mathbb{R}^n\}$ from the initial point $\bar{z}_0 = H(\hat{z})^{1/2} z_0$, and with step 1 modified to set $\bar{H}_0 = H(\hat{z})$, where, as in lemma (34), the overbars denote the sequences constructed by algorithm (7) in the process of solving $\min\{\bar{f}^0(z) \mid z \in \mathbb{R}^n\}$, $\bar{f}^0(z) = f^0(H(\hat{z})^{-1/2} z)$, and z_0, z_1, z_2, \dots is the sequence constructed by algorithm (7) in the process of solving problem (1). Now, according to lemma (34), in the process of solving the problem,

$$43 \quad \min\{\bar{f}^0(z) \mid z \in \mathbb{R}^n\},$$

we would construct matrices \bar{H}_i which are given by

$$44 \quad \bar{H}_i = H(\hat{z})^{1/2} H_i H(\hat{z})^{1/2}, \quad i = 0, 1, 2, \dots$$

(Recall that $H(\hat{z})$ is a symmetric positive definite matrix, and hence, so is $H(\hat{z})^{1/2}$.) Hence, (41) holds if and only if there exist constants m', M' , $0 < m' \leq M' < \infty$, such that

$$45 \quad m' \|y\|^2 \leq \langle y, \bar{H}_i y \rangle \leq M' \|y\|^2, \quad \text{for all } y \in \mathbb{R}^n, \quad i = 0, 1, 2, \dots$$

In addition, note that the function $\tilde{f}^0(\cdot)$ satisfies assumption (2) because $f^0(\cdot)$ satisfies assumption (2) (with possibly different values for the constants $0 < m \leq M$). Consequently, theorem (40) is true for the case of algorithm (7) solving problem (1), if and only if it is true for the case of algorithm (7) solving problem (43) (with (45) taking the place of (41)). Noting that the minimum of $\tilde{f}^0(z)$ occurs for $z = z' \triangleq H(\hat{z})^{1/2} \hat{z}$, we find that $\bar{H}(z') = \partial^2 \tilde{f}^0(z') / \partial z^2 = H(\hat{z})^{-1/2} H(\hat{z}) H(\hat{z})^{-1/2} = I$, the $n \times n$ identity matrix. Therefore, there is no loss of generality if we prove theorem (40) under the assumption that $H(\hat{z})$ is the identity matrix. As we shall see, this assumption leads to substantial simplifications in the algebra of the proof. In particular, from lemma (30) we obtain*

$$46 \quad \|\Delta g_i - \Delta z_i\| \leq L \Delta_i \|\Delta z_i\|, \quad i = 0, 1, 2, \dots$$

We begin by showing that the matrices H_i are bounded in norm, i.e., that there exists an $\bar{M} < \infty$ for which the right-hand side of (41) holds. To do this, we shall show that the ratio $\|H_{i+1}\|/\max\{1, \|H_i\|\}$ is bounded from above by a number $M_i > 1$. Then we shall show that the product $M_0 M_1 M_2 \cdots M_i \cdots$ is finite. For, suppose that the product $\prod_{i=0}^{\infty} M_i \leq \bar{M} < \infty$, and that $M_i \geq 1$ for $i = 0, 1, 2, \dots$. Then, since $\|H_{i+1}\|/\max\{1, \|H_i\|\} \leq M_i$, for $i = 0, 1, 2, \dots$,

$$47 \quad \max\{1, \|H_i\|\} \leq M_{i-1} \max\{1, \|H_{i-1}\|\}, \quad i = 1, 2, \dots$$

and hence, we must have, for $i = 0, 1, 2, \dots$,

$$48 \quad \|H_{i+1}\| \leq M_i \max\{1, \|H_i\|\} \leq \max\{1, \|H_0\|\} \prod_{j=0}^i M_j \leq \max\{1, \|H_0\|\} \bar{M}.$$

To establish the bounds M_i , we shall make use of the matrices Q_i , $i = 0, 1, 2, \dots$, defined as follows:

$$49 \quad Q_i = I - \frac{1}{\|\Delta z_i\|^2} \Delta z_i \langle \Delta z_i \rangle + \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta g_i \langle \Delta g_i \rangle, \quad i = 0, 1, 2, \dots$$

It follows from theorem (2.3.76) (or rather from its proof) that the matrices Q_i , $i = 0, 1, 2, \dots$, are symmetric and positive definite. Note also that by direct calculation, we obtain

$$50 \quad Q_i \Delta z_i = \Delta g_i.$$

We shall now show that

$$51 \quad \|Q_i^{1/2} H_{i+1} Q_i^{1/2}\| \leq \max\{1, \|Q_i^{1/2} H_i Q_i^{1/2}\|\} \quad \text{for } i = 0, 1, 2, \dots$$

* From now on we assume that $H(\hat{z}) = I$.

To prove (51), we shall make use of the fact that the euclidean norm of a symmetric matrix is the largest of the absolute values of its eigenvalues. For $i = 0, 1, 2, \dots$, we define

$$52 \quad x_i = Q_i^{1/2} \Delta z_i = Q_i^{-1/2} \Delta g_i,$$

the second equality being a consequence of (50). We now express Δz_i and Δg_i in terms of x_i and substitute for these quantities in (14). We then obtain the identity,

$$53 \quad \begin{aligned} Q_i^{1/2} H_{i+1} Q_i^{1/2} &= J_i - \frac{1}{\langle x_i, J_i x_i \rangle} J_i x_i \rangle \langle J_i x_i + \frac{1}{\|x_i\|^2} x_i \rangle \langle x_i \\ &= R_i + \frac{1}{\|x_i\|^2} x_i \rangle \langle x_i, \end{aligned}$$

where $J_i = Q_i^{1/2} H_i Q_i^{1/2}$ and the definition of R_i is obvious from (53). To establish (51), we have to show that the eigenvalues of the matrix $Q_i^{1/2} H_{i+1} Q_i^{1/2}$ are bounded by $\max\{1, \|J_i\|\}$. Since

$$54 \quad R_i = J_i - \frac{1}{\langle x_i, J_i x_i \rangle} J_i x_i \rangle \langle J_i x_i,$$

we find that for all $y \in \mathbb{R}^n$,

$$55 \quad \langle y, R_i y \rangle = \frac{\|J_i^{1/2}y\|^2 \|J_i^{1/2}x_i\|^2 - \langle J_i^{1/2}y, J_i^{1/2}x_i \rangle^2}{\langle x_i, J_i x_i \rangle} \geq 0,$$

by the Schwarz inequality, i.e., we find that R_i is positive semidefinite. (Note that $R_i x_i = 0$.) Suppose that y_i is an eigenvector of R_i corresponding to the largest eigenvalue of R_i . Then $R_i y_i = \|R_i\| y_i$, and hence,

$$56 \quad \begin{aligned} \langle y_i, R_i y_i \rangle &= \|R_i\| \|y_i\|^2 = \langle y_i, J_i y_i \rangle - \frac{\langle J_i x_i, y_i \rangle^2}{\langle x_i, J_i x_i \rangle} \\ &\leq \langle y_i, J_i y_i \rangle \leq \|J_i\| \|y_i\|^2, \end{aligned}$$

which shows that $\|R_i\| \leq \|J_i\|$. Now, since x_i is an eigenvector of R_i corresponding to the zero eigenvalue, and R_i is symmetric, x_i is orthogonal to all the other eigenvectors of R_i . Consequently, the matrix,

$$57 \quad Q_i^{1/2} H_{i+1} Q_i^{1/2} = R_i + \frac{1}{\|x_i\|^2} x_i \rangle \langle x_i,$$

has the same eigenvectors as R_i , with the same corresponding eigenvalues, with the exception of the eigenvalue corresponding to x_i , which is 1 (it was zero in R_i). Since the matrix in (57) is symmetric, we conclude that (51) is true.

Since the matrix Q_i is symmetric, we deduce from (51) that

$$\begin{aligned} 58 \quad \|H_{i+1}\| &= \|Q_i^{-1/2} Q_i^{1/2} H_{i+1} Q_i^{1/2} Q_i^{-1/2}\| \\ &\leq \|Q_i^{-1/2}\|^2 \|Q_i^{1/2} H_{i+1} Q_i^{1/2}\| \\ &\leq \|Q_i^{-1/2}\|^2 \max\{1, \|Q_i^{1/2}\|^2 \|H_i\|\} \\ &= \|Q_i^{-1}\| \max\{1, \|Q_i\| \|H_i\|\}. \end{aligned}$$

We now define the M_i , $i = 0, 1, 2, \dots$, as

$$59 \quad M_i = \max\{1, \|Q_i^{-1}\|\} \max\{1, \|Q_i\|\},$$

and note that $M_i \geq 1$. Next, observe that (58) yields

$$60 \quad \|H_{i+1}\| \leq M_i \max\{1, \|H_i\|\}, \quad i = 0, 1, 2, \dots,$$

i.e., we have established the first half of the inequality in (48). We shall now show that $\prod_{i=0}^{\infty} M_i < \infty$, to complete the proof of (48). For this purpose, we calculate bounds on $\|Q_i^{-1}\|$ and $\|Q_i\|$, which will lead us to a bound on M_i . By (49),

$$\begin{aligned} 61 \quad \|Q_i - I\| &= \left\| \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta g_i \langle \Delta g_i - \frac{1}{\|\Delta z_i\|^2} \Delta z_i \rangle \langle \Delta z_i \rangle \right\| \\ &\leq \left\| \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} (\Delta g_i - \Delta z_i) \langle \Delta g_i \rangle \right\| \\ &\quad + \left\| \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta z_i \langle (\Delta g_i - \Delta z_i) \rangle \right\| \\ &\quad + \left\| \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta z_i \langle \Delta z_i - \frac{1}{\|\Delta z_i\|^2} \Delta z_i \rangle \langle \Delta z_i \rangle \right\|. \end{aligned}$$

Since for any dyad $a \rangle \langle b$, $\|a \rangle \langle b\| = \|a\| \|b\|$, we find that

$$\begin{aligned} 62 \quad \|Q_i - I\| &\leq \frac{\|\Delta g_i - \Delta z_i\| \|\Delta g_i\|}{|\langle \Delta g_i, \Delta z_i \rangle|} + \frac{\|\Delta z_i\| \|\Delta g_i - \Delta z_i\|}{|\langle \Delta g_i, \Delta z_i \rangle|} \\ &\quad + \frac{\|\Delta g_i - \Delta z_i\| \|\Delta z_i\|}{|\langle \Delta g_i, \Delta z_i \rangle|}. \end{aligned}$$

The last term in the sum in (62) comes from the fact that

$$63 \quad \|\Delta z_i\|^2 \left| \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} - \frac{1}{\|\Delta z_i\|^2} \right| \leq \frac{\|\Delta g_i - \Delta z_i\| \|\Delta z_i\|}{|\langle \Delta g_i, \Delta z_i \rangle|}.$$

Note that $\langle \Delta g_i, \Delta z_i \rangle = \langle g_{i+1} - g_i, \Delta z_i \rangle = -\langle g_i, \Delta z_i \rangle > 0$. Now referring to lemma (2.3.89), we conclude that there exists a constant D , independent of i , such that

$$64 \quad \|Q_i - I\| \leq \frac{D \|\Delta g_i - \Delta z_i\|}{\|\Delta z_i\|}, \quad i = 0, 1, 2, \dots .$$

Consequently, we deduce from (46) that (see (29))

$$65 \quad \|Q_i - I\| \leq D L \Delta_i, \quad i = 0, 1, 2, \dots ,$$

and hence, that

$$66 \quad \|Q_i\| \leq 1 + D L \Delta_i, \quad i = 0, 1, 2, \dots .$$

Since (24) implies that $\Delta_i \rightarrow 0$ as $i \rightarrow \infty$, we conclude from (65) that there exists a positive constant D' such that

$$67 \quad \|Q_i^{-1}\| \leq 1 + D' \Delta_i, \quad i = 0, 1, 2, \dots .^*$$

Substituting from (66) and (67) into the definition of the M_i , (59), we find that

$$68 \quad M_i \leq (1 + D' \Delta_i)(1 + D L \Delta_i), \quad i = 0, 1, 2, 3, \dots .$$

Hence, to establish (48), it is sufficient to show that the product $\prod_{i=0}^{\infty} (1 + D' \Delta_i)(1 + D L \Delta_i)$ is bounded. It is not difficult to show that this product is bounded if the sum $\sum_{i=0}^{\infty} \Delta_i$ is convergent. Now, from (26) and (29), we deduce that

$$69 \quad \Delta_i \leq \frac{\sqrt{q(z_0)^i} \{2[f^0(z_0) - f^0(z)]/m\}^{1/2}}{1 - \sqrt{q(z_0)}},$$

and since $q(z_0) \in (0, 1)$, we conclude that $\sum_{i=0}^{\infty} \Delta_i$ is convergent. Hence, we are done with the first part of theorem (40).

We still have to prove the first half of (41), or equivalently, the first half of the inequality (42), i.e., that there exists an $\bar{m} > 0$ such that $0 < \bar{m} \leq \mu_i^j$, $j = 1, 2, \dots, n$, $i = 0, 1, 2, 3, \dots$, where the μ_i^j are the eigenvalues of H_i . For this it is sufficient to show that the matrices $G_i = H_i^{-1}$ are uniformly bounded in norm. We shall show that we can replace H_i by G_i in the in-

* Note that $-1 + \|Q_i^{-1}\| \leq \|Q_i^{-1} - I\| = \|Q_i^{-1}(I - Q_i)\| \leq \|Q_i^{-1}\| \|I - Q_i\| \leq \|Q_i^{-1}\| D L \Delta_i$. Hence, there must exist a $b < \infty$ such that $\|Q_i^{-1}\| \leq (1/1-DL\Delta_i) \leq b$ for $i = 0, 1, 2, \dots$, and consequently, $\|Q_i^{-1} - I\| \leq b D L \Delta_i$, $i = 0, 1, 2, \dots$.

equality (48). To obtain this result, we use (52) to express z_i and g_i in terms of x_i in equation (2.3.108). We then obtain the identity,

$$\begin{aligned} 70 \quad Q_i^{-1/2} G_i Q_i^{-1/2} &= \left(I - \frac{1}{\langle x_i, x_i \rangle} x_i \rangle \langle x_i \right) Q_i^{-1/2} G_i Q_i^{-1/2} \left(I - \frac{1}{\langle x_i, x_i \rangle} x_i \rangle \langle x_i \right) \\ &\quad + \frac{1}{\langle x_i, x_i \rangle} x_i \rangle \langle x_i \\ &= S_i + \frac{1}{\langle x_i, x_i \rangle} x_i \rangle \langle x_i, \end{aligned}$$

where the definition of S_i is clear from an inspection of (70). Now, since the matrix $(I - (1/\|x_i\|^2)x_i\rangle\langle x_i)$ is a symmetric projection operator, we must have

$$71 \quad \|S_i\| \leq \|Q_i^{-1/2} G_i Q_i^{-1/2}\|.$$

Furthermore, x_i is an eigenvector of S_i corresponding to the eigenvalue zero. Since the eigenvectors of S_i are orthogonal to each other, we conclude that the matrix $[S_i + (1/\|x_i\|^2)x_i\rangle\langle x_i]$ has the same eigenvectors and eigenvalues as S_i , with the exception of the eigenvalue corresponding to the eigenvector x_i , which was zero in S_i and now becomes 1. Therefore, we are led to an inequality which is quite similar to (51), namely,

$$72 \quad \|Q_i^{-1/2} G_{i+1} Q_i^{-1/2}\| \leq \max\{1, \|Q_i^{-1/2} G_i Q_i^{-1/2}\|\}, \quad i = 0, 1, 2, \dots.$$

We now obtain from (72) that

$$\begin{aligned} 73 \quad \|G_{i+1}\| &\leq \|Q_i\| \max\{1, \|Q_i^{-1}\| \|G_i\|\} \\ &\leq M_i \max\{1, \|G_i\|\} \\ &\leq \left(\prod_{j=0}^i M_j\right) \max\{1, \|G_0\|\} \leq \tilde{M} \max\{1, \|G_0\|\}, \end{aligned}$$

and hence we are done. ■

As the reader may recall, theorem (40) was of some importance in Section (2.3), where it was stated without proof (see (2.3.84)). However, its main value lies in the fact that we need it to prove the following theorem, which shows that the variable metric method converges superlinearly:

- 74 **Theorem.** Suppose that assumption (2) is satisfied, and that $\hat{z} \in \mathbb{R}^n$ is such that $f^0(\hat{z}) = \min\{f^0(z) \mid z \in \mathbb{R}^n\}$. If $\{z_i\}$ is an infinite sequence constructed by the variable metric algorithm (7), then $(\|z_{i+1} - \hat{z}\|/\|z_i - \hat{z}\|) \rightarrow 0$ as $i \rightarrow \infty$, i.e., $\{z_i\}$ converges to \hat{z} superlinearly.

Proof. We begin by reusing the arguments appearing in the beginning of the proof of theorem (40), where we used the function $\tilde{f}^0(\cdot)$, defined as

in (35), with $P = H(\hat{z})^{1/2}$, and constructed a sequence $\{\bar{z}_i\}$ in the process of solving problem (43), with algorithm (7) altered to initialize in step 1, so that $\bar{H}_0 = H(\hat{z})$. Setting $z' = H(\hat{z})^{1/2}\hat{z}$, we find that z' minimizes $\tilde{f}^0(z)$ over $z \in \mathbb{R}^n$, and hence, $\bar{z}_i \rightarrow z'$ as $i \rightarrow \infty$. Now, making use of (37) for $P = H(\hat{z})^{1/2}$, we obtain

$$75 \quad \bar{z}_i - z' = H(\hat{z})^{1/2}(z_i - \hat{z}), \quad i = 0, 1, 2, \dots.$$

Hence, we find again that without loss of generality we may assume that $H(\hat{z}) = I$, the identity matrix, and thereby simplify the algebra of the proof.*

By the Taylor formula for first-order expansions (B.1.3),

$$76 \quad g_i = g(z_i) = g(\hat{z}) + \int_0^1 H(\hat{z} + t(z_i - \hat{z})) dt (z_i - \hat{z}).$$

Since $g(\hat{z}) = 0$, and because of (5), we must have

$$77 \quad m \|z_i - \hat{z}\| \leq \|g_i\| \leq M \|z_i - \hat{z}\|, \quad i = 0, 1, 2, \dots.$$

Hence, we obtain that

$$78 \quad \frac{\|g_{i+1}\|}{\|g_i\|} \geq \frac{m}{M} \frac{\|z_{i+1} - \hat{z}\|}{\|z_i - \hat{z}\|}, \quad i = 0, 1, 2, \dots.$$

Consequently, if we can show that $\|g_{i+1}\|/\|g_i\| \rightarrow 0$ as $i \rightarrow \infty$, we are done. For this purpose we shall examine the sequence $\|G_i - I\|$, $i = 0, 1, 2, \dots$, where, as before, $G_i = H_i^{-1}$. (Powell states that he has constructed examples where the sequence $\{\|G_i - I\|\}$ does not converge to zero.) We begin by obtaining from (2.3.108) the relation,

$$79 \quad G_{i+1} - I = \left(I - \frac{1}{\|\Delta g_i\|^2} \Delta g_i \langle \Delta g_i \rangle \right) (G_i - I) \left(I - \frac{1}{\|\Delta g_i\|^2} \Delta g_i \langle \Delta g_i \rangle \right) + T_i, \quad i = 0, 1, 2, 3, \dots,$$

where T_i is defined by

$$80 \quad \begin{aligned} T_i &= \left(\frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta g_i \langle \Delta g_i \rangle - \frac{1}{\|\Delta g_i\|^2} \Delta g_i \langle \Delta g_i \rangle \right) \\ &\quad + \left(\frac{1}{\|\Delta g_i\|^2} G_i \Delta g_i \langle \Delta g_i \rangle - \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} G_i \Delta z_i \langle \Delta g_i \rangle \right) \\ &\quad + \left(\frac{1}{\|\Delta g_i\|^2} g_i \langle G_i \Delta g_i \rangle - \frac{1}{\langle \Delta g_i, \Delta z_i \rangle} \Delta g_i \langle G_i \Delta z_i \rangle \right) \\ &\quad + \left(\frac{\langle \Delta z_i, G_i \Delta z_i \rangle}{\langle \Delta g_i, \Delta z_i \rangle^2} \Delta g_i \langle \Delta g_i \rangle - \frac{\langle \Delta g_i, G_i \Delta g_i \rangle}{\|\Delta g_i\|^4} \Delta g_i \langle \Delta g_i \rangle \right). \end{aligned}$$

* We therefore continue to assume that $H(\hat{z}) = I$.

The terms in expression (80) were grouped into pairs so that, using lemma (2.3.89), inequality (46) and the fact, established in theorem (40), that G_i is uniformly bounded in norm, we can show that $\|T_i\| \leq W'\Delta_i$, $i = 0, 1, 2, 3, \dots$, for some $W' < \infty$, where Δ_i was defined in (29). To do this, we proceed essentially as we did to obtain (65) from (61). To conclude the proof, we shall need the Frobenius norm of a matrix, and we therefore digress for a moment to introduce it.

Given an $n \times n$ matrix $A = [a_{ij}]$, the *Frobenius norm* of A will be denoted by $\|A\|_F$ and is defined by

$$81 \quad \|A\|_F = \sqrt{\left(\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2\right)}.$$

The Frobenius norm has the following very interesting property: Suppose that b_1, b_2, \dots, b_n is any orthonormal set of vectors in \mathbb{R}^n ; then

$$82 \quad \|A\|_F^2 = \sum_{i=1}^n \|Ab_i\|^2,$$

where $\|\cdot\|$ denotes, as usual, the euclidean norm. Note that if we set $b_1 = (1, 0, 0, \dots, 0)$, $b_2 = (0, 1, 0, \dots, 0), \dots, b_n = (0, 0, \dots, 0, 1)$, we get (81) from (82). It can be shown that

$$83 \quad \|A\|_F \leq \sqrt{n}\|A\|.$$

This ends our digression.

Because of (83) and (2.1.89), there must exist a $W < \infty$ such that*

$$84 \quad \|T_i\|_F \leq W\Delta_i, \quad i = 0, 1, 2, \dots.$$

Consequently, (79) leads us to the inequality,

$$85 \quad \|G_{i+1} - I\|_F \leq \left\| \left(I - \frac{1}{\|\Delta g_i\|^2} \Delta g_i \otimes \Delta g_i \right) (G_i - I) \left(I - \frac{1}{\|\Delta g_i\|^2} \Delta g_i \otimes \Delta g_i \right) \right\|_F + W\Delta_i = \beta_i + W\Delta_i, \quad i = 0, 1, 2, \dots,$$

where the definition of β_i should be clear from an inspection of (85). In order to relate β_i to $\|G_i - I\|_F$, we shall make use of (82). In particular, by letting $b_1 = (1/\|\Delta g_i\|) \Delta g_i$, we conclude that (since $I - (1/\|\Delta g_i\|^2) \Delta g_i \otimes \Delta g_i$ is a projection matrix)

$$86 \quad \|G_i - I\|_F^2 - \beta_i^2 \geq \frac{\|(G_i - I)\Delta g_i\|^2}{\|\Delta g_i\|^2}, \quad i = 0, 1, 2, \dots.^\dagger$$

* Recall that $\|\Delta z \otimes \Delta g\| = \|\Delta z\|\|\Delta g\|$, etc.

† Suppose that A, B are $n \times n$ matrices, that $\{b_i\}$ is an orthonormal basis for \mathbb{R}^n , and that B is a projection operator such that $Bb_1 = 0$, $Bb_i = b_i$, $i = 2, 3, \dots, n$. Then, since $\|BAB_i\| \leq \|Ab_i\|$, $i = 1, 2, \dots, n$, $\|A\|_F^2 - \|BAB\|_F^2 = \|Ab_1\|^2 + \sum_{i=2}^n (\|Ab_i\|^2 - \|BAB_i\|^2) \geq \|Ab_1\|^2$.

Next, by dividing (86) by $(\|G_i - I\|_F + \beta_i)$, we obtain

$$87 \quad \|G_i - I\|_F - \beta_i \geq \frac{\|(G_i - I)\Delta g_i\|^2}{2V\|\Delta g_i\|^2}, \quad i = 0, 1, 2, \dots,$$

where $V < \infty$ is an upper bound on $\|G_i - I\|_F$ and hence also on β_i , and must exist because of the arguments in the proof of theorem (40). Note that V is independent of i .

Substituting from (87) into (85), we now obtain (by iteration on i)

$$88 \quad \|G_{i+1} - I\|_F \leq \|G_0 - I\|_F - \sum_{j=0}^i \frac{\|(G_j - I)\Delta g_j\|^2}{2V\|\Delta g_j\|^2} + W \sum_{j=0}^i \Delta_j, \\ i = 0, 1, 2, \dots.$$

Note that since $H_0 = I$, $G_0 = I$, and hence the first term in (88) is zero. Now, by (73), G_i is uniformly bounded from above, and from (69) we conclude that the second sum in (88) is also uniformly bounded from above. Hence, we conclude that the first sum in (88) must also be uniformly bounded from above (we are not including the minus sign which precedes the sum in the assertion we have just made). (Powell makes the interesting comment that if G_i does not converge to the unit matrix, then the convergence of the first sum in (88) (with i replaced by ∞) implies that the g_i become restricted in direction.) Let

$$89 \quad \sigma = \sum_{j=0}^{\infty} \frac{\|(G_j - I)\Delta g_j\|^2}{\|\Delta g_j\|^2}.$$

We shall now show that the sum,

$$90 \quad \sum_{j=0}^{\infty} \frac{\|(G_j \Delta z_j - \Delta g_j)\|^2}{\|\Delta g_j\|^2},$$

exists and is finite. In fact, making use of lemma (2.3.89), theorem (40) and of the inequalities (46) and (69), we obtain the bound,

$$91 \quad \begin{aligned} \sum_{j=0}^{\infty} \frac{\|(G_j \Delta z_j - \Delta g_j)\|^2}{\|\Delta g_j\|^2} &\leq \sum_{j=0}^{\infty} \frac{\|(G_j - I)\Delta g_j\|^2 + \|G_j \Delta z_j - G_j \Delta g_j\|^2}{\|\Delta g_j\|^2} \\ &\leq \sigma + [\sup_j \|G_j\|^2] \left[\sup_j \frac{\|\Delta z_j\|^2}{\|\Delta g_j\|^2} \right] L^2 \sum_{j=0}^{\infty} \Delta_j^2 \\ &< \infty. \end{aligned}$$

Now, making use of the Schwarz inequality, we obtain

$$\begin{aligned}
 92 \quad \|G_j \Delta z_j - \Delta g_j\| &\geq \frac{-1}{\|H_j g_{j+1}\|} \langle H_j g_{j+1}, G_j \Delta z_j - \Delta g_j \rangle \\
 &= \frac{1}{\|H_j g_{j+1}\|} \langle g_{j+1}, H_j g_{j+1} \rangle \\
 &\geq \frac{m}{M} \|g_{j+1}\|,
 \end{aligned}$$

since $\langle g_{j+1}, \Delta z_j \rangle = 0$, and because of (8),* (10), (13), and theorem (40) (see (41)). Consequently, we obtain from (91) and (92) that

$$93 \quad \sum_{j=0}^{\infty} \frac{\|g_{j+1}\|^2}{\|\Delta g_j\|^2} < \infty.$$

Next, since

$$94 \quad \|g_{j+1} - g_j\|^2 + \|g_{j+1} + g_j\|^2 = 2[\|g_j\|^2 + \|g_{j+1}\|^2],$$

we must have,

$$95 \quad \|\Delta g_j\|^2 = \|g_{j+1} - g_j\|^2 \leq 2[\|g_{j+1}\|^2 + \|g_j\|^2].$$

We therefore conclude from (93) that

$$96 \quad \sum_{j=0}^{\infty} \frac{\|g_{j+1}\|^2}{\|g_j\|^2 + \|g_{j+1}\|^2} < \infty.$$

Consequently, we must have

$$97 \quad \lim_{j \rightarrow \infty} \frac{\|g_{j+1}\|^2}{\|g_j\|^2 + \|g_{j+1}\|^2} = \lim_{j \rightarrow \infty} \frac{\|g_{j+1}\|^2 / \|g_j\|^2}{1 + \|g_{j+1}\|^2 / \|g_j\|^2} = 0,$$

which implies that

$$98 \quad \lim_{j \rightarrow \infty} \frac{\|g_{j+1}\|^2}{\|g_j\|^2} = 0.$$

Combining (98) with (78), we find that we are done. ■

* Note that because of (9), $\langle h_j, g_{j+1} \rangle = 0$, and hence, because of (8), $\langle H_j g_j, g_{j+1} \rangle = 0$.

APPENDIX A

FURTHER MODELS FOR COMPUTATIONAL METHODS

A.1 A Model for the Implementation of Certain Conceptual Optimal Control Algorithms

For the person who has read this book, it should now be quite clear that in utilizing algorithm model (1.3.33), we usually set $c(\cdot)$ to be the cost function. Since in discrete optimal control and nonlinear programming problems the evaluation of $f^0(z)$ (see (1.1.1)) presents no serious difficulties, model (1.3.33), as well as all the other models in Section 1.3, assumes that $c(z)$ can be evaluated exactly. However, in continuous optimal control problems, to evaluate $c(z)$ (or $F^0(u)$, as in Section 2.5, for example), we may have to integrate a function. Since the precision of this evaluation (as well as time consumed) depends on the step size used, and since it may not be necessary to evaluate $c(z)$ with great accuracy while one is far from a desirable point, it makes sense to attempt to introduce an ϵ procedure into the evaluation of $c(z)$ as well.* While this approach is still not completely developed, there is enough evidence to indicate that it can result in the computation time being reduced by a factor well in excess of 10 for problems with differential equations. The model below sheds light on the nature of assumptions we may have to verify.

As in Section 1.3, we assume that we are given a closed subset T of a Banach space \mathcal{B} containing desirable points which we wish to find (see (1.3.1)). The following algorithm model was developed by the author in collaboration with Robert Klessig, a graduate student:

- 1 **Algorithm Model.** $A : \mathbb{R}^+ \times T \rightarrow 2^T$, $c : \mathbb{R}^+ \times T \rightarrow \mathbb{R}^1$, $\epsilon_0 > 0$, $\alpha > 0$, $\beta \in (0, 1)$.

* This makes sense also in algorithms for solving problems of the form $\min_{x \in Q_x} \max_{y \in Q_y} K(x, y)$; see [K2b].

Step 0. Compute a $z_0 \in T$.

Step 1. Set $i = 0, j = 0, q(0) = 0$, and $\epsilon = \epsilon_0$.

Step 2. Compute a $y \in A(\epsilon, z_i)$.

Step 3. If $c(\epsilon, y) - c(\epsilon, z_i) \leq -\alpha\epsilon$, go to step 4; else, set $j = j + 1$, $\epsilon = \beta\epsilon$, and go to step 2.

Step 4. Set $z_{i+1} = y$, set $\epsilon_{i+1} = \epsilon$, set $q(i+1) = j$, set $i = i + 1$, and go to step 2.

Comment. The index j and the variables $q(i)$ and ϵ_i are inserted only for the purpose of the proofs to follow. Note that

$$2 \quad \epsilon_i = \beta^{q(i)} \epsilon_0, \quad i = 0, 1, 2, 3, \dots. \blacksquare$$

3 **Assumptions.*** We shall suppose that $c(\cdot, \cdot)$ and $A(\cdot, \cdot)$ have the following properties:

(i) $c(0, \cdot)$ is continuous on T ;

(ii) for every $z \in T$ which is not desirable, there exist an $\epsilon(z) > 0$, a $\delta(z) < 0$, and a $\gamma(z) > 0$ such that (compare (1.3.28))

$$4 \quad c(\gamma, z'') - c(\gamma, z') \leq \delta(z)$$

for all $z' \in B(z, \epsilon(z)) = \{z^0 \in T \mid \|z^0 - z\| \leq \epsilon(z)\}$, for all $z'' \in A(\gamma, z')$, for all $\gamma \in [0, \gamma(z)]$;

(iii) there exists a sequence $\{\xi_s\}_{s=0}^\infty$ such that $\xi_s > 0$ for $s = 0, 1, 2, \dots$,

$$5 \quad \sum_{s=0}^{\infty} \xi_s < \infty$$

and

$$6 \quad |c(\beta^s \epsilon_0, z) - c(0, z)| \leq \xi_s, \quad \text{for all } z \in T. \blacksquare$$

7 **Lemma.** Suppose that (3) (i) and (iii) are satisfied. If $\{z_i\}$ is an infinite sequence constructed by algorithm (1), and $\{z_i\}$ has at least one accumulation point, then the accompanying sequence $\{\epsilon_i\}_{i=0}^\infty$ converges to zero.

Proof. By construction, $\{\epsilon_i\}$ is a monotonically decreasing sequence which is bounded from below by zero, and consequently, it must converge. Suppose, therefore, that

$$8 \quad \epsilon_i \rightarrow \epsilon^* > 0 \quad \text{as } i \rightarrow \infty.$$

We shall show that the inequality in (8) leads to a contradiction. Relations (2) and (8) imply that there exists an integer $k \geq 0$ such that

$$9 \quad q(i) = q^* \quad \text{and} \quad \epsilon_i = \beta^{q^*} \epsilon_0 = \epsilon^* \quad \text{for all } i \geq k.$$

* For use with algorithms for solving nondiscretized continuous optimal control problems, it is necessary to refine model (1) a little; see [K2a].

Also, because of the test in step 3 of (1), we must have

$$10 \quad c(\epsilon^*, z_{i+1}) - c(\epsilon^*, z_i) \leq -\alpha\epsilon^* \quad \text{for all } i \geq k.$$

Consequently, we must have $c(\epsilon_i, z_i) \rightarrow -\infty$ as $i \rightarrow \infty$. Now, by (6) (because of (2)),

$$11 \quad c(\epsilon_i, z_i) \geq c(0, z_i) - \xi_{q(i)}, \quad i = 0, 1, 2, \dots$$

Hence, because of (9), we must have

$$12 \quad c(\epsilon_i, z_i) \geq c(0, z_i) - \xi_{q^*} \quad \text{for all } i \geq k.$$

Now suppose that $z_i \rightarrow z^*$ as $i \rightarrow \infty$, $i \in K \subset \{0, 1, 2, \dots\}$ (i.e., z^* is an accumulation point of $\{z_i\}$). Then, by (3) (i), there exists an integer $k' \geq k$ such that

$$13 \quad c(0, z_i) - c(0, z^*) \leq \xi_{q^*} \quad \text{for all } i \in K, \quad i \geq k'.$$

Combining (12) and (13), we obtain

$$14 \quad c(\epsilon_i, z_i) \geq c(0, z^*) - 2\xi_{q^*} \quad \text{for all } i \in K, \quad i \geq k',$$

which contradicts our original conclusion that $c(\epsilon_i, z_i) \rightarrow -\infty$ as $i \rightarrow \infty$. Consequently, the inequality in (8) must be false, i.e., we must have $\epsilon^* = 0$. ■

15 **Definition.** Let K be any infinite subsequence of the integers. We define the *index function* $k: K \rightarrow K$ by

$$16 \quad k(i) = \min\{j \in K \mid j \geq i + 1\},$$

i.e., $k(\cdot)$ computes the successive points of the subsequence. ■

17 **Lemma.** Suppose that $\{z_i\}_{i=0}^\infty$ is a sequence constructed by algorithm (1) and that $K \subset \{0, 1, 2, 3, \dots\}$. If assumption (3)(iii) is satisfied, then

$$18 \quad c(\epsilon_{k(i)}, z_{k(i)}) \leq 2 \sum_{j=q(i+1)}^{q(k(i))} \xi_j + c(\epsilon_{i+1}, z_{i+1}) \quad \text{for all } i \in K.$$

Proof. Let $N = \{0, 1, 2, 3, \dots\}$ and let $b: N \rightarrow \{0, 1\}$ be defined by

$$19 \quad b(i) = \begin{cases} 0 & \text{if } q(i) = q(i-1) \\ 1 & \text{otherwise.} \end{cases}$$

Now, because of the test in step 3 of (1),

$$20 \quad c(\epsilon_{i+1}, z_{i+1}) \leq c(\epsilon_{i+1}, z_i) - \alpha\epsilon_{i+1}, \quad i = 0, 1, 2, \dots$$

Hence, if $b(i+1) = 0$, $\epsilon_{i+1} = \epsilon_i$, and (20) yields

$$21 \quad c(\epsilon_{i+1}, z_{i+1}) \leq c(\epsilon_i, z_i) - \alpha\epsilon_{i+1}, \quad \text{for } i \in N \text{ if } b(i+1) = 0.$$

If $b(i+1) = 1$, then making use of (6) twice, we obtain, from (20),

$$22 \quad \begin{aligned} c(\epsilon_{i+1}, z_{i+1}) &\leq c(\epsilon_{i+1}, z_i) - \alpha\epsilon_{i+1} \\ &\leq c(0, z_i) + \xi_{q(i+1)} - \alpha\epsilon_{i+1} \\ &\leq c(\epsilon_i, z_i) + \xi_{q(i+1)} + \xi_{q(i)} - \alpha\epsilon_{i+1}, \\ &\quad \text{for } i \in N \text{ if } b(i+1) = 1. \end{aligned}$$

Adding $\alpha\epsilon_{i+1}$ to the right-hand sides of (21) and (22), we see that

$$23 \quad c(\epsilon_{i+1}, z_{i+1}) \leq c(\epsilon_i, z_i) + b(i+1)[\xi_{q(i+1)} + \xi_{q(i)}], \quad i = 0, 1, 2, \dots$$

Making use of (23) recursively, we now obtain for all $i \in K$,

$$24 \quad c(\epsilon_{k(i)}, z_{k(i)}) \leq \sum_{j=i+2}^{k(i)} b(j)[\xi_{q(j)} + \xi_{q(j-1)}] + c(\epsilon_{i+1}, z_{i+1}).$$

Since $b(j) = 1$ implies that $q(i-1) < q(i)$, it follows that for any s such that $q(i+1) \leq s \leq q(k(i))$, ξ_s can be repeated at most twice in (24). Consequently, (24) yields

$$25 \quad \sum_{j=i+2}^{k(i)} b(j)[\xi_{q(j)} + \xi_{q(j-1)}] \leq 2 \sum_{j=q(i+1)}^{q(k(i))} \xi_j.$$

Combining (24) and (25), we now obtain (18). ■

- 26 **Theorem.** Suppose that assumption (3) is satisfied. Then, either algorithm (1) jams up at a desirable point z_i after a finite number of iterations, or else it constructs an infinite sequence $\{z_i\}$ such that every accumulation point of that sequence is desirable.

Proof. First, suppose that algorithm (1) jams up at a point z_k which is not desirable. Then, by (3)(ii) there must exist a $\delta(z_k) < 0$ and a $\gamma(z_k) > 0$ such that

$$27 \quad c(\gamma, z'') - c(\gamma, z_k) \leq \delta(z_k) \quad \text{for all } z'' \in A(\gamma, z_k), \quad \gamma \in [0, \gamma(z_k)].$$

Now, when the algorithm jams up, it cycles between steps 2 and 3. Hence, it must be generating a sequence $\{y_p\}_{p=0}^{\infty}$ such that $y_p \in A(\beta^{q(k)+p}\epsilon_0, z_k)$, $p = 0, 1, 2, \dots$, and

$$28 \quad c(\beta^{q(k)+p}\epsilon_0, y_p) - c(\beta^{q(k)+p}\epsilon_0, z_k) > -\alpha\beta^{q(k)+p}\epsilon_0, \quad p = 0, 1, 2, \dots$$

However, for some integer $p' \geq 0$, we must have

$$29 \quad \max\{\alpha\beta^{q(k)+p'}\epsilon_0, \beta^{q(k)+p'}\epsilon_0\} \leq \min\{-\delta(z), \gamma(z)\},$$

and hence, we see that (28) cannot hold if z_k is not desirable. Thus, algorithm (1) cannot jam up at a nondesirable point z_k .

Now suppose that the sequence $\{z_i\}$ is infinite and that z^* is an accumulation point of that sequence. Thus, suppose that $z_i \rightarrow z^*$ as $i \rightarrow \infty$, for $i \in K \subset \{0, 1, 2, \dots\}$, and that $k(\cdot)$ is the index function for K . Suppose that z^* is not desirable. We shall show that this leads to a contradiction.

First, because of lemma (7), we note that $q(i) \rightarrow \infty$ as $i \rightarrow \infty$ and that $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$. Next, by (3)(iii),

$$30 \quad c(\epsilon_i, z_i) \geq c(0, z_i) - \xi_{q(i)}, \quad i = 0, 1, 2, \dots$$

From (3)(i) we conclude that

$$31 \quad c(0, z_i) \rightarrow c(0, z^*) \quad \text{as } i \rightarrow \infty, \quad i \in K.$$

Consequently, since $z_{i+1} \in A(\epsilon_{i+1}, z_i)$, since $\epsilon_i \rightarrow 0$ as $i \rightarrow \infty$, and since $\xi_{q(i)} \rightarrow 0$ as $i \rightarrow \infty$ because of (5), we obtain from (30) and (31) that

$$32 \quad \lim_{i \in K} c(\epsilon_i, z_i) \geq \lim_{i \in K} [c(0, z_i) - \xi_{q(i)}] \geq c(0, z^*).$$

Now, since z^* is not desirable, by (3)(ii), there exist an $\epsilon^* > 0$, a $\delta^* < 0$ and a $\gamma^* > 0$ such that

$$33 \quad c(\gamma, z'') - c(\gamma, z') \leq \delta^* \\ \text{for all } z' \in B(z^*, \epsilon^*), \text{ for all } z'' \in A(\gamma, z'), \text{ for all } \gamma \in [0, \gamma^*].$$

Since $\epsilon_i \rightarrow 0$ and since $z_i \rightarrow z^*$ as $i \rightarrow \infty$, for $i \in K$, there must exist an integer k' such that for all $i \geq k'$, $z_i \in B(z^*, \epsilon^*)$ and $\epsilon_i \leq \gamma^*$. Hence, by (33),

$$34 \quad c(\epsilon_{i+1}, z_{i+1}) - c(\epsilon_{i+1}, z_i) \leq \delta^* \quad \text{for all } i \in K, \quad i \geq k'.$$

Referring to (3)(iii), we define

$$35 \quad b_n = \sum_{s=0}^n \xi_s.$$

Then $\{b_n\}_{n=0}^\infty$ is a monotonically increasing sequence which is bounded from above, and which therefore converges. Consequently, since $q(i) \rightarrow \infty$ as $i \rightarrow \infty$, there must exist an integer $k'' \geq k'$ such that

$$36 \quad b_{q(k(i))} - b_{q(i)} \leq -\frac{\delta^*}{8} \quad \text{for all } i \in K, \quad i \geq k'',$$

by the Cauchy condition. Furthermore, because of (3)(iii) and because $\xi_{q(i)} \rightarrow 0$ as $i \rightarrow \infty$, there exist must a $k_1 \geq k''$ such that

$$37 \quad |c(\epsilon_{i+1}, z_i) - c(0, z_i)| \leq -\frac{\delta^*}{8} \quad \text{for all } i \in K, \quad i \geq k_1,$$

$$38 \quad |c(\epsilon_i, z_i) - c(0, z_i)| \leq -\frac{\delta^*}{8} \quad \text{for all } i \in K, \quad i \geq k_1.$$

Hence,

$$39 \quad |c(\epsilon_{i+1}, z_i) - c(\epsilon_i, z_i)| \leq -\frac{\delta^*}{4} \quad \text{for all } i \in K, \quad i \geq k_1.$$

Now, from lemma (17) and (36) we conclude that

$$\begin{aligned} 40 \quad & c(\epsilon_{k(i)}, z_{k(i)}) - c(\epsilon_i, z_i) \\ & \leq 2[b_{q(k(i))} - b_{q(i)}] + c(\epsilon_{i+1}, z_{i+1}) - c(\epsilon_i, z_i) \\ & \leq -\frac{\delta^*}{4} + c(\epsilon_{i+1}, z_{i+1}) - c(\epsilon_i, z_i), \quad \text{for all } i \in K, \quad i \geq k_1. \end{aligned}$$

Adding and subtracting $c(\epsilon_{i+1}, z_i)$ to the right-hand side of (40) and making use of (34) and (39), we conclude that

$$\begin{aligned} 41 \quad & c(\epsilon_{k(i)}, z_{k(i)}) - c(\epsilon_i, z_i) \\ & \leq -\frac{\delta^*}{4} + c(\epsilon_{i+1}, z_{i+1}) - c(\epsilon_{i+1}, z_i) + c(\epsilon_{i+1}, z_i) - c(\epsilon_i, z_i) \\ & \leq -\frac{\delta^*}{4} + \delta^* - \frac{\delta^*}{4} = \frac{\delta^*}{2} \\ & < 0 \quad \text{for all } i \in K, \quad i \geq k_1. \end{aligned}$$

Now (41) implies that

$$42 \quad c(\epsilon_i, z_i) \rightarrow -\infty \quad \text{as } i \rightarrow \infty, \quad \text{for } i \in K,$$

and this contradicts (32). Hence, we conclude that z^* must have been desirable. ■

A.2 An Open-Loop Model for the Implementation of Conceptual Algorithms

In reading Section 1.3, the reader must have wondered whether it is absolutely necessary to have an ϵ procedure in algorithms of the “approxima-

tion” type (such as (1.3.33)). The following results show that the quality of the approximation need not be improved adaptively. It may be improved at a preselected rate. In practice, algorithms of the form given below are very efficient in a situation where one resolves essentially the same problem over and over again. In such a case, one would spend a certain amount of time carefully selecting the “truncation function” to be used, and one would then be likely to obtain an algorithm which is faster than an algorithm with an ϵ procedure.

As in Section 1.3, we assume that we are given a closed subset T of a Banach space which contains points with a property P . We call points in T with property P desirable. The algorithm we are about to describe finds desirable points in T .

Let $N = \{0, 1, 2, \dots\}$. We shall say that a function $t: N \rightarrow N$ is a *truncation function* if for every $k \in N$ there exists a $k' \in N$ such that $t(i) > k$ for all $i \geq k'$.

1 Algorithm Model.* $A: N \times T \rightarrow 2^T$, $c: T \rightarrow \mathbb{R}^1$, $t_1: N \rightarrow N$, $t_2: N \rightarrow N$.

Step 0. Compute a $z_0 \in T$ and set $i = 0$.

Step 1. Set $z = z_i$ and set $j = t_1(i)$.

Step 2. Compute a $y \in A(j, z)$.

Step 3. If $c(y) < c(z)$, set $z_{i+1} = y$, set $i = i + 1$, and go to step 1; else, set $z_{i+1} = z_i$, set $j = t_2(i)$, set $i = i + 1$, and go to step 2. ■

2 Theorem. Consider algorithm (1). Suppose that $t_1(\cdot)$ and $t_2(\cdot)$ are truncation functions and that $c(\cdot)$ and $A(\cdot, \cdot)$ have the following properties: (i) $c(\cdot)$ is either continuous at all nondesirable $z \in T$, or else $c(z)$ is bounded from below for $z \in T$; (ii) for every $z \in T$ which is not desirable, there exist an $\epsilon(z) > 0$, a $\delta(z) < 0$ and a $k(z) \in N$, such that

3 $c(z'') - c(z') \leq \delta(z) < 0$

for all $z' \in B(z, \epsilon(z))$, for all $z'' \in A(j, z')$, for all $j \geq k(z)$.

Then every accumulation point of an infinite sequence z_i constructed by algorithm (1) is desirable.

Proof. First suppose that there is an integer $i' \in N$ such that $z_{i'} = z_{i'+1} = z_{i'+2} = \dots$. Then we find that the algorithm keeps on constructing vectors $y \in A(t_1(i') + t_2(i' + q), z_{i'})$, $q = 1, 2, \dots$, such that $c(y) \geq c(z_{i'})$. Since $t_1(\cdot)$ and $t_2(\cdot)$ are both truncation functions, it now follows from assumption (ii) above that $z_{i'}$ must be desirable. Note that in this case $z_{i'}$ is the limit point point of the sequence constructed by the algorithm.

Now suppose that there is no integer $i \in N$ such that $z_i = z_{i+1} = \dots$ and that \hat{z} is an accumulation point of $\{z_i\}$. Thus, suppose that \hat{z} is not desirable

* This algorithm model and the accompanying convergence theorem were first presented in [P3a].

and that $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$ for $i \in K \subset \{0, 1, 2, \dots\}$. Then, by assumption (ii) above, there exist an $\epsilon > 0$, a $\delta < 0$ and a $k \in N$ such that

$$4 \quad c(z'') - c(z') \leq \delta$$

for all $z' \in B(\hat{z}, \hat{\epsilon})$, for all $z'' \in A(j, z')$, for all $j \geq k$. Now, since $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, for $i \in K$, there exists an integer $k \in N$ such that $t_1(i) \geq k$ and $z_i \in B(\hat{z}, \hat{\epsilon})$ for all $i \in K$, $i \geq k$. Let K' be any infinite subset of K such that if $i, i+p$ are two consecutive indices in K' , then $z_{i+p} \neq z_i$. Consequently, $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, for $i \in K'$, and in addition, because of (4), if $i, i+p$ are two consecutive indices in K' and $i \geq k$, then

$$5 \quad c(z_{i+p}) - c(z_i) = [c(z_{i+p}) - c(z_{i+p-1})] + \cdots + [c(z_{i+1}) - c(z_i)] < \delta.$$

But the sequence $\{c(z_i)\}_{i \in K'}$ must converge because of assumption (i), and, since this is contradicted by (5), we conclude that \hat{z} must have been desirable. ■

To illustrate the manner in which model (1) can be used, we shall modify once again the method of centers (4.2.27) (compare (4.2.47)). First, we modify the Golden section search algorithm as follows:

6 **Algorithm** (Golden section search). Integer $k \geq 0$ to be supplied; $F_1 = (3 - \sqrt{5})/2$, $F_2 = (\sqrt{5} - 1)/2$ (compare (2.1.14)). Computes an interval containing the minimizer of a convex function $\theta : \mathbb{R}^1 \rightarrow \mathbb{R}^1$ when this minimizer is in $[0, \infty)$.

- Step 0.* Select a $\rho > 0$.
- Step 1.* Compute $\theta(0)$, $\theta(\rho)$.
- Step 2.* If $\theta(\rho) \geq \theta(0)$, set $a_0 = 0$, $b_0 = \rho$, and go to step 7; else, go to step 3.
- Step 3.* Set $i = 0$ and set $\mu_0 = 0$.
- Step 4.* Set $\mu_{i+1} = \mu_i + \rho$.
- Step 5.* Compute $\theta(\mu_{i+1})$.
- Step 6.* If $\theta(\mu_{i+1}) \geq \theta(\mu_i)$, set $a_0 = \mu_{i-1}$, $b_0 = \mu_{i+1}$, and go to step 7; else, set $i = i + 1$ and go to step 4.

Comment. The desired minimizer is contained in the interval $[a_0, b_0]$.

- Step 7.* Set $j = 0$.
- Step 8.* Set $v_j = a_j + F_1(b_j - a_j)$, $w_j = a_j + F_2(b_j - a_j)$.
- Step 9.* If $\theta(v_j) < \theta(w_j)$, set $a_{j+1} = a_j$, set $b_{j+1} = w_j$, and go to step 10; else, set $a_{j+1} = v_j$, set $b_{j+1} = b_j$, and go to step 10.
- Step 10.* If $j < k$, set $j = j + 1$ and go to step 8; else, set $\bar{\mu} = (a_j + b_j)/2$ and stop. ■

We now incorporate this into algorithm (4.2.27), in accordance with model (1), as follows (compare (4.2.47)), to obtain an algorithm applicable under the same convexity assumptions as for (4.2.47):

7 Algorithm (implementation of modified method of centers, Polak [P3a]).

Step 0. Compute a $z_0 \in C$, and set $i = 0$; select truncation functions $t_1(\cdot)$, $t_2(\cdot)$.

Step 1. Set $z = z_i$ and set $k = t_1(i)$.

Step 2. Solve (4.2.22)–(4.2.25) to obtain $(h^0(z), h(z))$.

Step 3. If $h^0(z) = 0$, set $z_{i+1} = z_i$ and stop; else, go to step 4.

Step 4. Set $\theta(\mu) = d(z + \mu h(z), z)$ and use (6) to compute $\bar{\mu}$.

Step 5. If $f^0(z + \bar{\mu}h(z)) < f^0(z)$, set $z_{i+1} = z + \bar{\mu}h(z)$, set $i = i + 1$, and go to step 1; else, set $z_{i+1} = z_i$, set $k = t_2(i)$, set $i = i + 1$, and go to step 4. ■

8 Exercise. Show that algorithm (7) satisfies the assumptions of theorem (2). ■

Throughout this book, we have modeled all the calculations needed to compute a point z_{i+1} from a point z_i by means of a single map A (with $z_{i+1} \in A(z_i)$, or $z_{i+1} \in A(\epsilon, z_i)$, or $z_{i+1} \in A(j, z_i)$). Occasionally, it may be advantageous to decompose the map A into a composition of several maps and to develop conditions on the components of A , which ensure convergence of the algorithm in which it is used. The interested reader will find this process described in [M8].

APPENDIX B

PROPERTIES OF CONTINUOUS FUNCTIONS

B.1 Expansions of Continuous Functions

Throughout this text we make use of expansions of differentiable functions. These expansions can be found in any respectable textbook on analysis, and we summarize these here only for the sake of providing the reader with a handy source of reference. (For further details see, for example, Dieudonné [D3].)

- 1 **Mean-Value Theorem.** Suppose that $f^i(\cdot)$ is a continuously differentiable function from \mathbb{R}^n into \mathbb{R}^1 . Then, for any z, h in \mathbb{R}^n and for any $\lambda \in \mathbb{R}^1$,

$$2 \quad f^i(z + \lambda h) = f^i(z) + \lambda \langle \nabla f^i(\xi), h \rangle,$$

where $\xi \in [z, z + \lambda h]$ (with $[x, y]$ denoting the line segment joining x and y , i.e., $[x, y] = \{z = \lambda x + (1 - \lambda)y : \lambda \in [0, 1]\}$). ■

- 3 **Taylor's Formula for First-Order Expansions.** Suppose that $g(\cdot)$ is a continuously differentiable function from \mathbb{R}^n into \mathbb{R}^m ; then for any z, h in \mathbb{R}^n and for any $\lambda \in \mathbb{R}^1$,

$$4 \quad g(z + \lambda h) = g(z) + \left(\int_0^1 \frac{\partial g(z + t\lambda h)}{\partial z} dt \right) (\lambda h),$$

where $\partial g(z)/\partial z$ is an $m \times n$ matrix whose ij th component is $\partial g^i(z)/\partial z^j$ (usually referred to as a Jacobian matrix). ■

- 5 **Generalized Mean-Value Theorem.** Suppose that $g(\cdot)$ is a continuously

differentiable function from \mathbb{R}^n into \mathbb{R}^m , then for any z, h in \mathbb{R}^n and for any $\lambda \in \mathbb{R}^1$,

$$6 \quad \|g(z + \lambda h) - g(z)\| \leq \left[\sup_{\xi \in [z, z + \lambda h]} \left\| \frac{\partial g(\xi)}{\partial z} \right\| \right] \|\lambda h\|. \quad \blacksquare$$

Note that (6) follows easily from (4). However, it can also be proved independently by making use of the mean-value theorem.

- 7 **Definition.** Suppose that $g(\cdot)$ is a continuously differentiable function from \mathbb{R}^n into \mathbb{R}^m . The second derivative of $g(\cdot)$ at $z \in \mathbb{R}^n$ will be denoted by $g''(z)(\cdot)$ (or more simply by $g''(z)$) and is defined by

$$8 \quad \lim_{\|y\| \rightarrow 0} \frac{1}{\|y\|} \left[\frac{\partial g(z + y)}{\partial z} - \frac{\partial g(z)}{\partial z} - g''(z)(y) \right] = 0$$

whenever this limit exists and is independent of $y \in \mathbb{R}^n$. ■

We see from (8) that $g''(z)(y)$ is an $m \times n$ matrix. It may be easier to visualize this matrix if we proceed as follows: Expanding $g(z + y)$ to second-order terms about z , we must have

$$9 \quad g(z + y) = g(z) + \frac{\partial g(z)}{\partial z} y + \frac{1}{2} g''(z)(y) y + r(y),$$

where $\|r(y)\|/\|y\|^2 \rightarrow 0$ as $\|y\|^2 \rightarrow 0$. Examining (9), component by component, we obtain, for $i = 1, 2, \dots, m$,

$$10 \quad g^i(z + y) = g^i(z) + \langle \nabla g^i(z), y \rangle + \frac{1}{2} \left\langle \frac{\partial^2 g^i(z)}{\partial z^2} y, y \right\rangle + r^i(y),$$

and hence we see that $g''(z)(y)$ is an $m \times n$ matrix whose i th row is $[(\partial^2 g^i(z)/\partial z^2)y]^T$.

The norm of the linear operator $g''(z)$ is defined as follows:

$$11 \quad \|g''(z)\| = \max\{\|g''(z)(y')\| \mid \|y'\| \leq 1, \|y'\| \leq 1\}.$$

We can now state a formula for second-order expansions.

- 12 **Taylor's Formula for Second-Order Expansions.** Suppose that $g(\cdot)$ is a twice continuously differentiable function from \mathbb{R}^n into \mathbb{R}^m . Then, for any z, h in \mathbb{R}^n and for any $\lambda \in \mathbb{R}^1$,

$$13 \quad g(z + \lambda h) = g(z) + \lambda \frac{\partial g(z)}{\partial z} h + \lambda^2 \left(\int_0^1 (1-t) g''(z + t\lambda h)(h) dt \right) h. \quad \blacksquare$$

B.2 Convex Functions

We shall now summarize the few properties of convex functions which we require in this text. When proofs are not given, the reader may look them up in Berge [B8].

- 1 **Definition.** A function $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is said to be *convex* if for any y, z in \mathbb{R}^n and any $\lambda \in [0, 1]$,

$$2 \quad f^i(\lambda z + (1 - \lambda)y) \leq f^i(z) + (1 - \lambda)f^i(y).$$

The function $f^i(\cdot)$ is said to be *strictly convex* if the inequality in (2) is strict for all $z \neq y$ in \mathbb{R}^n and for all $\lambda \in (0, 1)$. ■

- 3 **Theorem.** Suppose that $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a convex function. Then $f^i(\cdot)$ is continuous. ■

The theorems below give alternative characterizations of a convex function for the case when this function is differentiable.

- 4 **Theorem.** Suppose that $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a continuously differentiable function. Then $f^i(\cdot)$ is convex if and only if for any y, z in \mathbb{R}^n ,

$$5 \quad f^i(z) - f^i(y) \leq \langle \nabla f^i(z), z - y \rangle.$$

When the inequality in (5) is strict for all $y \neq z$ in \mathbb{R}^n , the function $f^i(\cdot)$ is strictly convex, and vice versa. ■

- 6 **Theorem.** Suppose that $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is twice continuously differentiable. Then $f^i(\cdot)$ is convex if and only if the Hessian $\partial^2 f^i(z)/\partial z^2$ is positive semi-definite for all z in \mathbb{R}^n . ■

- 7 **Theorem.** Suppose that $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a continuously differentiable convex function. If $\hat{z} \in \mathbb{R}^n$ is such that $\nabla f^i(\hat{z}) = 0$, then $f^i(\hat{z}) \leq f^i(z)$ for all $z \in \mathbb{R}^n$. (This result follows directly from (5).) ■

- 8 **Theorem.** Suppose that $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ is a twice continuously differentiable function and that there exists $0 < m < \infty$ such that for any y, z in \mathbb{R}^n ,

$$9 \quad m \|y\|^2 \leq \langle y, H(z)y \rangle,$$

where $H(z) = \partial^2 f^i(z)/\partial z^2$; then $f^i(\cdot)$ is strictly convex and for any z_0 in \mathbb{R}^n the set,

$$10 \quad \{z \mid f^i(z) \leq f^i(z_0)\},$$

is bounded.

Proof. By the Taylor formula for second-order expansions (1.12), for any $y \neq z$ in \mathbb{R}^n ,

$$\begin{aligned} 11 \quad f^i(z) - f^i(y) &= \langle \nabla f^i(z), z - y \rangle \\ &\quad - \int_0^1 \langle z - y, H(z - t(z - y))(z - y) \rangle (1 - t) dt \\ &\leq \langle \nabla f^i(z), z - y \rangle - \frac{m}{2} \|z - y\|^2 \\ &< \langle \nabla f^i(z), z - y \rangle. \end{aligned}$$

It now follows from (4) that $f^i(\cdot)$ is strictly convex.

Let $z_0 \in \mathbb{R}^n$ be arbitrary and let h be any unit vector in \mathbb{R}^n . Then, by the Taylor formula for second-order expansions,

$$\begin{aligned} 12 \quad f^i(z_0 + \lambda h) - f^i(z_0) &= \lambda \langle \nabla f^i(z_0), h \rangle + \lambda^2 \int_0^1 \langle h, H(z_0 + t\lambda h) h \rangle (1 - t) dt \\ &\geq -\lambda \|\nabla f^i(z_0)\| + \frac{\lambda^2 m}{2}. \end{aligned}$$

We therefore conclude that

$$13 \quad \{z \mid f^i(z) \leq f^i(z_0)\} \subset \left\{ z = z_0 + \lambda h \mid \|\lambda h\| = 1, \lambda \in \left[0, \|\nabla f^i(z_0)\| \frac{2}{m}\right] \right\},$$

which completes our proof. ■

14 **Theorem.** Suppose that $f^i(\cdot)$ is a strictly convex function from \mathbb{R}^n into \mathbb{R}^1 . Then there can exist only one $z \in \mathbb{R}^n$ which minimizes $f^i(z)$. ■

15 **Exercise.** Prove theorem (14). ■

B.3. A Few Miscellaneous Results

The proofs given in this text require the frequent use of the assortment of results given below.

1 **Theorem.** Suppose that $f^i(\cdot)$ is a continuous function from \mathbb{R}^n into \mathbb{R}^1 and that S is a compact subset of \mathbb{R}^n . Then for each $z' \in \mathbb{R}^n$ and for each $\delta > 0$, there exist an $\epsilon' > 0$ and a $\lambda_m > 0$ such that for all $z \in B(z', \epsilon') = \{z \mid \|z - z'\| \leq \epsilon'\}$ and for all $h \in S$,

$$2 \quad |f^i(z + \lambda h) - f^i(z)| \leq \delta \quad \text{for all } \lambda \in [0, \lambda_m].$$

Proof. Let $z' \in \mathbb{R}^n$ and $0 < \delta < \infty$ be arbitrary. Let $\epsilon' > 0$ be arbitrary, but finite. Then $f^i(\cdot)$ is uniformly continuous on the compact ball $B(z', \epsilon') = \{z \mid \|z - z'\| \leq \epsilon'\}$, and hence, there exists an $\epsilon'' > 0$ such that

$$3 \quad |f^i(z) - f^i(z'')| \leq \delta$$

for all $z \in B(z', \epsilon')$, for all z'' satisfying $\|z - z''\| \leq \epsilon''$. Let $\epsilon' = \min\{\epsilon/2, \epsilon''\}$ and let $M = \max\{\|h\| \mid h \in S\}$. If we now set $\lambda_m = \epsilon'/M$, then for all $h \in S$ and for all $z \in B(z', \epsilon')$ we must have

$$4 \quad \|zh\| \leq \epsilon'' \quad \text{for all } \lambda \in [0, \lambda_m],$$

and

$$5 \quad (z + \lambda h) \in B(z, \epsilon'') \quad \text{for all } \lambda \in [0, \lambda_m].$$

Therefore, because of (3) and (5), for all $z \in B(z', \epsilon')$ and for all $h \in S$,

$$6 \quad |f^i(z + \lambda h) - f^i(z)| \leq \delta \quad \text{for all } \lambda \in [0, \lambda_m]. \blacksquare$$

7 **Theorem.** Suppose that $f^i(\cdot)$ is a continuously differentiable function from \mathbb{R}^n into \mathbb{R}^1 and that S is a compact subset of \mathbb{R}^n . Then for each $z' \in \mathbb{R}^n$ and for each $\delta > 0$, there exist an $\epsilon' > 0$ and a $\lambda_m > 0$ such that for all $z \in B(z', \epsilon') = \{z \mid \|z - z'\| \leq \epsilon'\}$ and for all $h \in S$,

$$8 \quad |\langle \nabla f^i(z + \lambda h), h \rangle - \langle \nabla f^i(z), h \rangle| \leq \delta \quad \text{for all } \lambda \in [0, \lambda_m].$$

Proof. Let $z' \in \mathbb{R}^n$ and $0 < \delta < \infty$ be arbitrary. Let $\epsilon' > 0$ be arbitrary, but finite. Then the function $\langle \nabla f^i(\cdot), \cdot \rangle$, from $\mathbb{R}^n \times \mathbb{R}^n$ into \mathbb{R}^1 is uniformly continuous on the compact set $B(z', \epsilon') \times S$, and hence, there exists an $\epsilon'' > 0$ such that

$$9 \quad |\langle \nabla f^i(z), h \rangle - \langle \nabla f^i(z''), h'' \rangle| \leq \delta,$$

for all (z, h) in $B(z', \epsilon') \times S$ and (z'', h'') satisfying $\|z - z''\| < \epsilon''$, $\|h - h''\| < \epsilon''$. Let $\epsilon' = \min\{\epsilon/2, \epsilon''\}$ and let $M = \max\{\|h\| \mid h \in S\}$. If we set $\lambda_m = \epsilon'/M$, then for all $h \in S$ and for all $z \in B(z', \epsilon')$, we have

$$10 \quad \|zh\| \leq \epsilon'' \quad \text{for all } \lambda \in [0, \lambda_m],$$

and therefore,

$$11 \quad (z + \lambda h) \in B(z, \epsilon'') \quad \text{for all } \lambda \in [0, \lambda_m].$$

Hence, because of (9), we must have for all $z \in B(z', \epsilon')$ and for all $h \in S$,

$$12 \quad |\langle \nabla f^i(z + \lambda h), h \rangle - \langle \nabla f^i(z), h \rangle| \leq \delta \quad \text{for all } \lambda \in [0, \lambda_m]. \blacksquare$$

- 13 **Theorem.** Suppose that for $i = 1, 2, \dots, m$, the functions $f^i : \mathbb{R}^n \rightarrow \mathbb{R}^1$ are continuous. Then the function $M : \mathbb{R}^n \rightarrow \mathbb{R}^1$ defined by

$$14 \quad M(z) = \max\{f^i(z) \mid i \in \{1, \dots, m\}\}$$

is also continuous.

Proof. Let $z' \in \mathbb{R}^n$ and $0 < \delta < \infty$ be arbitrary. Then there exists an $\epsilon > 0$, possibly depending on z' , such that for all $z \in \dot{B}(z', \epsilon) = \{z \mid \|z - z'\| < \epsilon\}$,

$$15 \quad f^i(z') - \delta < f^i(z) < f^i(z') + \delta \quad \text{with } i = 1, 2, \dots, m.$$

Hence, we obtain that, for all $z \in \dot{B}(z', \epsilon)$,

$$17 \quad f^i(z) < M(z') + \delta \quad \text{with } i = 1, 2, \dots, m,$$

and

$$18 \quad f^i(z') - \delta < M(z) \quad \text{with } i = 1, 2, \dots, m.$$

Taking the maximum over i in (17) and (18), we now obtain

$$19 \quad M(z') - \delta < M(z) < M(z') + \delta \quad \text{for all } z \in \dot{B}(z', \epsilon),$$

and hence, we conclude that $M(\cdot)$ is continuous. ■

The above theorem is actually a special case of the following one.

- 20 **Theorem.** Let $\psi(\cdot, \cdot)$ be a continuous function from $\mathbb{R}^p \times \mathbb{R}^q$ into \mathbb{R}^1 , and let S be a compact subset of \mathbb{R}^q . Then the function $\theta : \mathbb{R}^p \rightarrow \mathbb{R}^1$, defined by

$$21 \quad \theta(z) = \min\{\psi(z, h) \mid h \in S\}$$

is also continuous.

Proof. Let $z' \in \mathbb{R}^p$ be arbitrary. Let $\epsilon > 0$ be arbitrary, but finite. Then $\psi(\cdot, \cdot)$ is uniformly continuous on $B(z', \epsilon) \times S$, and, given any $\delta' > 0$, there exists an $\epsilon'' > 0$ such that*

$$22 \quad |\psi(z, h) - \psi(z'', h'')| < \delta',$$

for all (z, h) in $B(z', \epsilon) \times S$ and (z'', h'') satisfying $\|z - z''\| < \epsilon''$, $\|h - h''\| < \epsilon''$. Let $\epsilon' = \min\{\epsilon, \epsilon''\}$; then for all z in the interior of $B(z', \epsilon')$ and for all $h \in S$, we must have

$$23 \quad \psi(z, h) - \delta' < \psi(z', h) < \psi(z, h) + \delta'.$$

* Recall that $B(z', \epsilon) = \{z \mid \|z - z'\| \leq \epsilon\}$.

First minimizing the left side and then the right side of (23) over $h \in S$, we obtain, for all z in the interior of $B(z', \epsilon')$,

$$24 \quad \theta(z) - \delta' < \psi(z', h) \quad \text{for all } h \in S$$

$$25 \quad \theta(z') < \psi(z, h) + \delta' \quad \text{for all } h \in S.$$

Once again minimizing over $h \in S$, we obtain from (24) and (25) that

$$26 \quad |\theta(z) - \theta(z')| < \delta' \quad \text{for all } z \in B(z', \epsilon'),$$

which proves that $\theta(\cdot)$ is continuous. ■

APPENDIX C

A GUIDE TO IMPLEMENTABLE ALGORITHMS

C.1 General Considerations

In choosing an implementable algorithm for solving a specific optimization problem, we should attempt to evaluate the following aspects of its performance: The first aspect is that of numerical accuracy, or sensitivity to errors. The second aspect is that of the total time the algorithm is likely to require to solve the problem in question. Since there are virtually no theoretical results shedding light on the sensitivity of optimization algorithms to errors, and since available empirical results are extremely difficult to interpret, there is little we can do to enlighten the reader on this subject. About the only observation that we can make safely is that it is not uncommon to experience computational difficulties because of errors introduced through the evaluation of derivatives. These errors limit the accuracy with which an optimal point can be determined. When such difficulties are encountered, it may be advisable to substitute for the algorithm one is using an algorithm that uses fewer derivatives (see Section 2.2).

The situation is somewhat better as far as total computing time is concerned, and therefore, we can provide the reader with a few coarse guidelines for deciding whether a particular problem is well- or ill-suited for solution by means of a particular algorithm. The total computing time depends on three factors, not counting the specific digital computer that is to be used. These factors are the rate of convergence of the algorithm, the time taken per iteration, and the skill of the programmer who writes the digital computer program (code). The skill of the programmer, particularly if he inserts clever heuristics into the algorithm, plays a critical part in determining the total

computing time. For example, there are known cases where a linear programming code became four times faster when a programmer changed the rule for deciding when a very small number will be called zero. Consequently, we advise the reader to get the most skillful and experienced programmer that he can find.

Now let us turn to rate of convergence. For the unconstrained minimization and root finding methods discussed in Chapters 2 and 3, we have obtained convergence rates in Chapter 6, where we have classified algorithms as converging linearly, superlinearly and quadratically. While the rate of convergence, or rather the bound on the rate of convergence that we can obtain, is an intrinsic property of an algorithm, the various constants that enter our expression for the bound on the rate of convergence are problem-dependent. For example, as we have seen in Section 6.1, algorithms converging linearly construct a sequence $z_i \rightarrow \hat{z}$ as $i \rightarrow \infty$, with $\|z_i - \hat{z}\| \leq E\theta^i$, where both E and $\theta \in (0, 1)$ are problem-dependent. Consequently, in deciding whether a particular method is suitable for solving a given problem, we must take into consideration not only the rate of convergence of this algorithm, but also the "conditioning" caused by the nature of the problem under consideration. Generally speaking, when the region of search is very narrow or banana-shaped, algorithms with a search along the gradient of the objective function, or "first-order" methods of feasible directions (in Section 4.3) will be ill-conditioned, while quasi-Newton methods, conjugate gradient methods and "second-order" methods of feasible directions (in Section 4.4) will be much less sensitive to the shape of the region of search. Consequently, when the region of search is unfavorably shaped, one would be inclined to use one of the superlinearly converging algorithms, unless the time per iteration of this algorithm happened to be very large.

The computing time per iteration depends on the number and computational difficulty of the function and derivative evaluations required, as well as on the complexity of the arithmetical operations involved. The latter can be quite time-consuming when they take on the form of matrix inversions in linear or quadratic programming subprocedures, etc. Consequently, an algorithm may perform very well, relative to rival algorithms, when its computing time is dominated by function and derivative evaluations (because these are very time-consuming) and it may perform relatively rather poorly when function and derivative calculations are simple.

In view of the preceding discussion, it should be clear that it is not possible to state without ambiguity that an algorithm is superior to some other algorithm. About all one can say is that in making his choice of an algorithm, the reader should try to minimize the following crude index: (conditioning number) \times (time per iteration)/(rate of convergence). Denoting the rate of convergence by r , we find that $r = 1$ for algorithms converging linearly,

$r = 2$ for algorithms converging quadratically, and $1 < r \leq 2$ for the algorithms in this text, which converge superlinearly. The conditioning number is a fudge factor which should reflect the lack of sphericity in the region of search, with a convex, near spherical region being given a small number, and a badly shaped region being given a large number, the actual number also depending on the algorithm used. Normally, such a conditioning number must be guessed at on the basis of past experience. (In the case of gradient methods in unconstrained minimization, we can take as the conditioning number the ratio (largest eigenvalue of Hessian of objective function)/(smallest eigenvalue of Hessian of objective function), with the Hessian eigenvalues estimated at the optimal point.)

Having convinced the reader that the selection of an algorithm is anything but an exact science, we shall not attempt to classify implementable algorithms. Instead, in the pages to follow, we shall indicate and try to justify our preferences within some fairly narrow families of algorithms that were discussed in this book.

C.2 Gradient Methods

In the category of implementable gradient methods, we have seen (for $D(z) \equiv I$, the identity matrix) the algorithms (2.1.16) (with adaptive step size selection), (2.1.19) (with a two-line rule for step size selection), and (2.1.35) (with a one-line rule for step size selection). Figures 1 and 2 show (at least as far as this example is concerned) that algorithms (2.1.19) and (2.1.35) are not too sensitive to the parameter α , while Figs. 2 and 3 show that algorithm (2.1.35) may be somewhat sensitive to the parameter β , indicating that some experimentation with this parameter is justified. Figure 4 compares the best of the runs in Figs. 1, 2 and 3 with a run using algorithm (2.1.16), and shows that the rate of convergence of the three algorithms is quite comparable.

Since the rate of convergence of the three algorithms under discussion is about the same, as well as the conditioning with respect to a given problem, we favor algorithm (2.1.35) because it has by far the simplest step size rule and hence uses less time per iteration than the other two methods. We recall that because of theorem (1.3.66) and exercise (2.1.34), we can be reasonably sure that for many well-posed problems a sequence constructed by this algorithm will actually converge to a stationary point.

Because of the results in Section 6.1 and of some computational experience, we feel that the parameters in algorithm (2.1.35) should be set as follows: $\alpha = 0.5$, $\beta \in (0.5, 0.8)$, $\rho = 1$. For convenience, we summarize our preferences in the form of an algorithm statement.

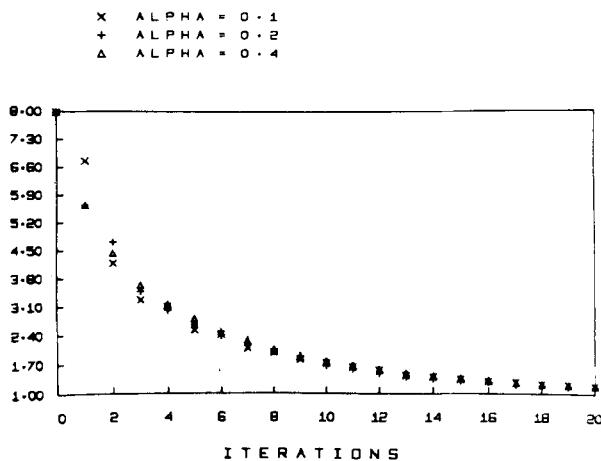


Fig. 1. Sensitivity of double-line gradient method (algorithm (2.1.37) with step size subprocedure (2.1.33)) to the choice of the parameter α . The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$; $z_0^1 = 1.32$, $z_0^2 = -0.07$; the parameter $\rho = 1$. The graph shows $f^0(z_i)$ versus i (iterations).

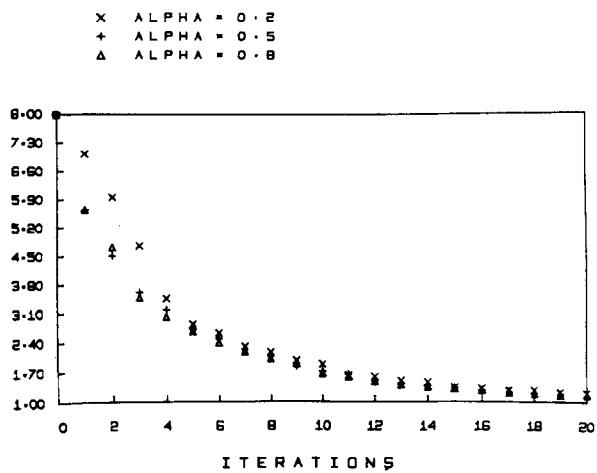


Fig. 2. Sensitivity of single-line gradient method (algorithm (2.1.37) with step size subprocedure (2.1.36)) to the choice of the parameter α . The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$; $z_0^1 = 1.32$, $z_0^2 = -0.07$; the parameter $\rho = 1$, the parameter $\beta = 0.5$. The graph shows $f^0(z_i)$ versus i (iterations).

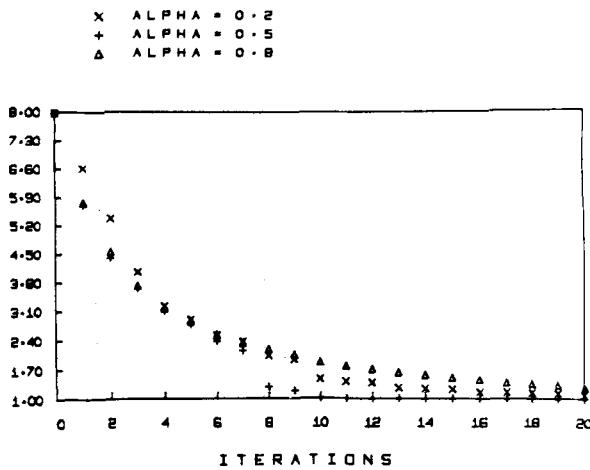


Fig. 3. Sensitivity of single-line gradient method (algorithm (2.1.37) with step size subprocedure (2.1.36)) to the choice of the parameter α . The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$; $z_0^1 = 1.32$, $z_0^2 = -0.07$; the parameter $\rho = 1$, the parameter $\beta = 0.8$. The graph shows $f^0(z_i)$ versus i (iterations).

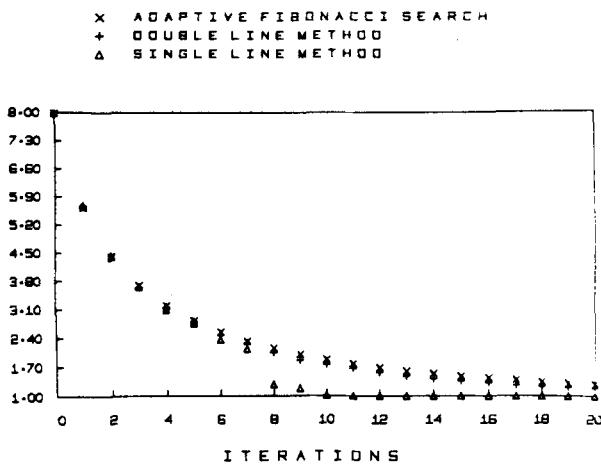


Fig. 4. A comparison of the method of steepest descent with adaptive Fibonacci search (algorithm (2.1.16), with $D(z) \equiv I$, $\rho = 1$, $\epsilon_0 = 0.00001$) with the double-line gradient method (algorithm (2.1.37) with step size subprocedure (2.1.33), for $\alpha = 0.4$, $\rho = 1$) and with the single-line gradient method (algorithm (2.1.37) with step size subprocedure (2.1.36), for $\alpha = 0.5$, $\beta = 0.8$, $\rho = 1$). The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$; $z_0^1 = 1.32$, $z_0^2 = -0.07$. The graph shows $f^0(z_i)$ versus i (iterations).

1 Algorithm (recommended gradient method).

Comment. Solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$, for $f^0(\cdot)$ continuously differentiable. For relevant theory, see Sections 2.1 and 6.1.

Step 0. Select a $z_0 \in \mathbb{R}^n$; select a $\beta \in (0.5, 0.8)$, and set $i = 0$.

Step 1. Compute $h(z_i) = -\nabla f^0(z_i)$.

Step 2. If $h(z_i) = 0$, stop; else, go to step 3.

Comment. We now compute the step size.

Step 3. Set $\lambda = 1$.

Step 4. Compute

$$2 \quad \Delta = f^0(z_i + \lambda h(z_i)) - f^0(z_i) + \frac{\lambda}{2} \|\nabla f^0(z_i)\|^2.$$

Step 5. If $\Delta \leq 0$, set $\lambda_i = \lambda$, and go to step 6; else, set $\lambda = \beta\lambda$ and go to step 4.

Step 6. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 1. ■

C.3 Quasi-Newton Methods

In the category of quasi-Newton methods, we find the algorithms (2.1.16), (2.1.19) and (2.1.35), with $D(z) \equiv (\partial^2 f^0(z)/\partial z^2)^{-1}$ (the Hessian of the objective function), as well as the Newton-Raphson method (2.1.39), itself. Again, we favor the version (2.1.35), because its rate of convergence and conditioning is the same as that of algorithms (2.1.19) and (2.1.39), but its step size rule is simpler than that of algorithm (2.1.19), and because it usually has a larger region of convergence than (2.1.39). Incidentally, so far it is not known whether the implementable algorithm (2.1.16), with $D(z)$ set equal to the inverse of the Hessian, has a better rate of convergence than the Newton-Raphson method or not. As was also the case with its gradient version, the quasi-Newton version of algorithm (2.1.35) can be expected to construct a sequence of points that will converge to a stationary point. (See theorem (1.3.66) and exercise (2.1.34).) Because of the results in Sections 6.1 and 6.2, we recommend the parameter values $\alpha = 0.5$, $\beta \in (0.5, 0.8)$, and $\rho = 1$. We now summarize our suggestions in the form of an algorithm.

1 Algorithm (recommended quasi-Newton method).

Comment. Solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$, for $f^0(\cdot)$ twice continuously differentiable. For relevant theory, see Sections 2.1, 3.1, 6.1 and 6.2.

Step 0. Select a $z_0 \in \mathbb{R}^n$; select a $\beta \in (0.5, 0.8)$, and set $i = 0$.

Step 1. Compute $\nabla f^0(z_i)$.

Step 2. If $\nabla f^0(z_i) = 0$, stop; else, go to step 3.

Step 3. Compute $H(z_i) = \partial f^0(z_i)/\partial z^2$.

Step 4. If $H(z_i)^{-1}$ exists, compute $h(z_i)$ by solving

$$2 \quad H(z_i) h(z_i) = -\nabla f^0(z_i),$$

and go to step 5; else, set $h(z_i) = -\nabla f^0(z_i)$ and go to step 5.

Comment. When $H(z_i)$ is singular, the quasi-Newton method is not applicable, and hence we revert to method (2.1).

Comment. We now compute the step size as in (2.1).

Step 5. Set $\lambda = 1$.

Step 6. Compute

$$3 \quad \Delta = f^0(z_i + \lambda h(z_i)) - f^0(z_i) - \frac{\lambda}{2} \langle \nabla f^0(z_i), h(z_i) \rangle.$$

Step 7. If $\Delta \leq 0$, set $\lambda_i = \lambda$, and go to step 8; else, set $\lambda = \lambda\beta$ and go to step 6.

Step 8. Set $z_{i+1} = z_i + \lambda_i h(z_i)$, set $i = i + 1$, and go to step 1. ■

A comparison between the quasi-Newton method above, the gradient algorithm (2.1.16) and the Polak-Ribière conjugate gradient method (2.3.51) is shown in Fig. 5.

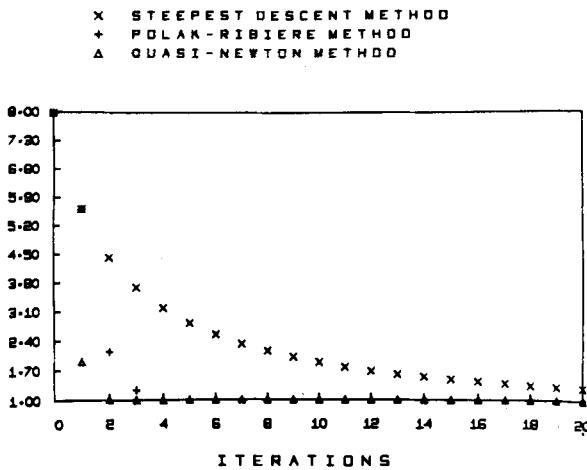


Fig. 5. A comparison of the method of steepest descent with adaptive Golden section search (algorithm (2.1.16) with $D(z) \equiv I$, $\rho = 1$, $\epsilon_0 = 0.00001$) with the Polak-Ribière conjugate gradient method (2.3.51) (implemented as in (2.3.132) with $\alpha = 1$, $\beta = 0.5$, $\rho = 1$, $\epsilon_0 = 0.00001$), and with the quasi-Newton method (3.1) (with $\beta = 0.5$). The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$; $z_0^1 = 1.32$, $z_0^2 = -0.07$. The graph shows $f^0(z_i)$ versus i (iterations).

C.4 Conjugate Gradient Algorithms

In the category of conjugate gradient algorithms, we have seen in Section 2.3 the Fletcher-Reeves algorithm (2.3.42), the Polak-Ribi  re algorithm (2.3.51) and the variable metric algorithm (2.3.68). The Polak-Ribi  re algorithm seems to be preferable to the Fletcher-Reeves algorithm for the following reasons: (i) we have shown for the Polak-Ribi  re algorithm that $h_i \rightarrow 0$ as $i \rightarrow \infty$ (see (2.3.64)); this is not certain to be the case for the Fletcher-Reeves algorithm; (ii) in the Polak-Ribi  re algorithm, the direction vector h_i stays within a fixed cone about $\nabla f^0(z_i)$, whereas in the case of the Fletcher-Reeves algorithm this cone grows with i . The time per iteration of the two algorithms is essentially the same, while the fact that $h_i \rightarrow 0$ as $i \rightarrow \infty$ and that h_i stays within a fixed cone about $\nabla f^0(z_i)$ makes any implementation of the Polak-Ribi  re algorithm more numerically stable (less sensitive to error accumulation) than the Fletcher-Reeves algorithm. Experiments support this last statement.

Since conjugate gradient methods are often used to minimize functions which are not necessarily convex, and since empirical results justify this practice, we suggest the following implementation for the Polak-Ribi  re algorithm, which is somewhat more general than (2.3.132). This algorithm has the same convergence properties as the Polak-Ribi  re algorithm (2.3.51) (see [K2c]). It has also been shown in [K2c] that when the algorithm below is reinitialized, it has the same rate of convergence as algorithm (6.3.4).

Algorithm (Klessig-Polak [K2c]; recommended implementation of Polak-Ribi  re algorithm).

Comment. Solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$, for $f^0(\cdot)$ twice continuously differentiable. For relevant theory, see [K3c] and Sections (2.3) and (6.3).

Step 0. Select a $z_0 \in \mathbb{R}^n$, and parameters $\bar{\epsilon}' > 0$, $\bar{\epsilon}'' > 0$, $\beta \in (0, 1)$, $\beta' \in (0, 1)$, $\beta'' \in (0, 1)$.

Comment. Try $\bar{\epsilon}' = \cos 85^\circ$, $\bar{\epsilon}'' = \cos 5^\circ$, $\beta = 0.6$, $\beta' = \beta'' = 0.8$.

Step 1. Set $g_0 = h_0 = -\nabla f^0(z_0)$; set $\epsilon' = \bar{\epsilon}'$, $\epsilon'' = \bar{\epsilon}''$; set $i = 0$.

Step 2. If $\nabla f^0(z_0) = 0$, stop; else, go to step 3.

Step 3. Set $z = z_i$, $h = (1/\|h_i\|)h_i$.

Step 4. Define $\theta : \mathbb{R}^1 \rightarrow \mathbb{R}^1$ by

2

$$\theta(x) = f^0(z + xh) - f^0(z).$$

Comment. To compute the step length, we minimize $\theta(x)$ by means of algorithm (2.1).

Step 5. Set $x = 0$.

Step 6. Compute

$$3 \quad \theta'(x) = \langle \nabla f^0(z + xh), h \rangle.$$

Step 7. If $\theta'(x) = 0$, go to step 15; else, go to step 8.

Step 8. Set $\lambda = 1$.

Step 9. Compute

$$4 \quad \Delta = \theta(x - \lambda\theta'(x)) - \theta(x) + \frac{1}{2}\lambda\theta'(x)^2.$$

Step 10. If $\Delta \leq 0$, set $x = x - \lambda\theta'(x)$ and go to step 11; else, set $\lambda = \beta\lambda$, and go to step 9.

Step 11. Compute $\nabla f^0(z + xh)$.

Step 12. If $\nabla f^0(z + xh) = 0$, set $z_{i+1} = z + xh$ and stop; else, go to step 13.

Step 13. Compute $\theta'(x)$ according to (3), and set $\delta = \theta'(x)/\|\nabla f^0(z + xh)\|$.

Step 14. If $|\delta| \leq \epsilon'$, go to step 15; else, go to step 8.*

Comment. The test on δ ensures that the minimization of $\theta(x)$ has been carried sufficiently far for the algorithm to be convergent.

Step 15. Set

$$5 \quad z_{i+1} = z + xh,$$

$$6 \quad g_{i+1} = -\nabla f^0(z + xh),$$

$$7 \quad h_{i+1} = g_{i+1} + \gamma_i h_i, \quad \text{with } \gamma_i = \frac{\langle g_{i+1} - g_i, g_{i+1} \rangle}{\|g_i\|^2},$$

Step 16. Set $i = i + 1$.

Step 17. If $\langle g_i, h_i \rangle \geq \epsilon'' \|g_i\| \|h_i\|$, go to step 3; else, set $\epsilon' = \beta'\epsilon'$, $\epsilon'' = \beta''\epsilon''$, and go to step 3.

Comment. The purpose of the operations in step 17 is to find an $\epsilon' > 0$ which is compatible with the convergence of the algorithm. This test will always be satisfied after a finite number of iterations. ■

Figure 5 shows a comparison between the steepest descent algorithm (2.1.16), the implementation (2.3.132) of the Polak-Ribière method, and the quasi-Newton method (3.1). While the quasi-Newton method has the best rate of convergence, it also requires more time per iteration than the other two methods. Based on this consideration, one would usually prefer algorithm (1) or (2.1.132) to a gradient or quasi-Newton method.[†]

While implementations of the variable metric method (2.3.68), using a cubic interpolation in computing the step size, are known to be more numerically

* To be absolutely sure of superlinear convergence, use the test $\delta \leq \min\{\epsilon', \|g_i\|\}$ in step 14, and replace γ_i by $\omega(i+1)\gamma_i$, in (7), as in (6.3.8).

[†] Note that in Fig. 5, algorithm (2.3.132) appears to converge superlinearly, satisfying (6.3.10) with $p = n$.

stable than either of the conjugate gradient methods we have seen, with the same type of rule for selecting the step size, it is not at all clear that such implementations of the variable metric method are superior to algorithm (1). In addition, the variable metric algorithm requires a computer with a much larger, fast access memory because of the need to store the matrix H_i . For the reader who has a preference for the variable metric algorithm, we suggest, instead of using polynomial interpolation in computing step size, that he proceed as in algorithm (1), since we believe that this will result in a more stable implementation.

8 Algorithm (recommended implementation of variable metric algorithm).

Comment. Solves the problem, $\min\{f^0(z) \mid z \in \mathbb{R}^n\}$, for $f^0(\cdot)$ twice continuously differentiable. For relevant theory, see Sections 2.3, 6.4 and A.2.

Step 0. Select a $z_0 \in \mathbb{R}^n$; select an integer k satisfying $1 \leq k \leq 10$; select $\beta \in (0.5, 0.8)$; set $i = j = 0$. If $\nabla f^0(z_0) = 0$, stop; else, go to step 1.

Step 1. Set $H_0 = I$ (the $n \times n$ identity matrix), set $g_0 = \nabla f^0(z_0)$.

Step 2. Set

$$9 \quad h_i = -H_i g_i.$$

Step 3. If $(i/k) = 0$ modulo k , set $j = j + 1$, $k = 3k$, and go to step 4; else, go to step 4 (see comment after step 3 in (1)).

Step 4. Set $q = 0$; set $z = z_i$; set $h = h_i$; and define $\theta(\cdot)$ as in (2).

Step 5. Set $x = 0$.

Step 6. Compute $\theta'(x)$ according to (3).

Step 7. If $\theta'(x) = 0$, set $\lambda_i = x$ and go to step 12; else, go to step 8.

Step 8. Set $\lambda = 1$.

Step 9. Compute Δ according to (4).

Step 10. If $\Delta \leq 0$, set $q = q + 1$ and go to step 11; else, set $\lambda = \beta\lambda$ and go to step 9.

Step 11. If $q < j$, set $x = x - \lambda\theta'(x)$ and go to step 6; else, set $\lambda_i = x - \lambda\theta'(x)$, and go to step 12;

Step 12. Compute $\nabla f^0(z_i + \lambda_i h_i)$.

Step 13. If $\nabla f^0(z_i + \lambda_i h_i) = 0$, stop; else, set

$$10 \quad z_{i+1} = z_i + \lambda_i h_i,$$

$$11 \quad g_{i+1} = \nabla f^0(z_{i+1}),$$

$$12 \quad \Delta g_i = g_{i+1} - g_i,$$

$$13 \quad \Delta z_i = z_{i+1} - z_i,$$

$$14 \quad H_{i+1} = H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} H_i \Delta g_i \langle H_i \Delta g_i \rangle + \frac{1}{\langle \Delta z_i, \Delta g_i \rangle} \Delta z_i \langle \Delta z_i \rangle,$$

set $i = i + 1$, and go to step 2. ■

C.5 Penalty Function Methods

There seems to be a feeling that penalty function methods cannot be used for determining a point satisfying optimality conditions with great accuracy, i.e., it is felt that they tend to suffer from numerical errors. As a result, they tend to become slow towards the end. Generally, exterior penalty function methods are considered to perform better than interior penalty function methods, as can be seen even in the very simple example in Fig. 6. However,

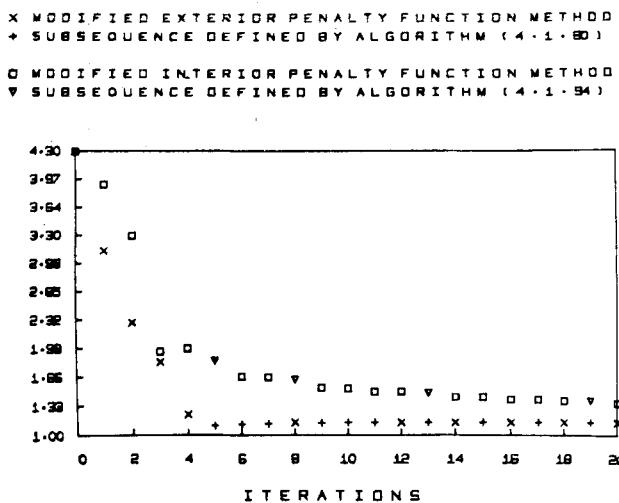


Fig. 6. A comparison of the modified exterior penalty function method (4.1.81) with the modified interior penalty function method (4.1.95). For both algorithms the choice of parameters was $\epsilon = 1$, $\alpha = 0.4$, $\rho = 1$, $\beta = 2$. The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$, subject to $(z^1) + 2(z^2)^2 - 1 \leq 0$, $(z^1)^2 + (z^2)^2 - 4z^1 + 1 \leq 0$, $(z^1)^2 + (z^2)^2 - z^1 - z^2 \leq 0$; $z_0^1 = 0.95$, $z_0^2 = 0.10$.

a study by Lootsma [L3] indicates that mixed penalty function methods converge more frequently than either exterior or interior penalty function methods. Because of this, we recommend the mixed penalty function method given below, which is a cross between algorithms (4.1.81) and (4.1.95). We recommend this algorithm on the basis of the theoretical results in Section 4.1, which show that it can only converge to points satisfying the Kuhn-Tucker optimality conditions. For heuristically based implementations, see [F2].

1 Algorithm (recommended mixed penalty function method).

Comment. Solves the problem, $\min\{f^0(z) | f^i(z) \leq 0, i = 1, 2, \dots, m, r^j(z) =$

$0, j = 1, 2, \dots, l\}$, where the $f^j(\cdot)$ and $r^j(\cdot)$ are continuously differentiable. Requires the assumption that the set $C = \{z \in \mathbb{R}^n \mid f^i(z) \leq 0, i = 1, 2, \dots, m, r^j(z) = 0, j = 1, 2, \dots, l\}$ satisfies the Kuhn-Tucker constraint qualification (see [C1]). For relevant theory see Section 4.1.

Step 0. Select a $\overset{0}{z} \in \mathbb{R}^n$; select $\alpha, \alpha', \alpha'' \in (0, 0.5)$; select a $\beta \in (0.5, 0.8)$.

Step 1. Set $z = \overset{0}{z}$ and set $i = 0$.

Step 2. Define the index sets,

$$2 \quad I' = \{j \in \{1, 2, \dots, m\} \mid f^j(z) \geq 0\},$$

$$3 \quad I'' = \{j \in \{1, 2, \dots, m\} \mid f^j(z) < 0\}.$$

Step 3. Define the exterior and interior penalty functions,

$$4 \quad p'(z) = \sum_{j=1}^l r^j(z)^2 + \sum_{j \in I'} [\max\{0, f^j(z)\}]^2,$$

$$5 \quad p''(z) = - \sum_{j \in I''} \frac{1}{f^j(z)}.$$

Step 4. If $i = 0$, go to step 5; else, go to step 8.

Step 5. Compute $\nabla f^0(z), \nabla p'(z), \nabla p''(z)$.

Step 6. Set $\epsilon' = \|\nabla p'(z)\|/\|\nabla f^0(z)\|, \epsilon'' = \|\nabla f^0(z)\|/\|\nabla p''(z)\|$.

Step 7. Select an $\epsilon_0 \in (0.1, 1)$.

Step 8. Compute

$$6 \quad h(z) = - \left[\nabla f^0(z) + \frac{1}{\epsilon'} \nabla p'(z) + \epsilon'' \nabla p''(z) \right].$$

Step 9. If $\|h(z)\| > \epsilon_i$, go to step 10; else, set $\epsilon_{i+1} = \alpha \epsilon_i, \epsilon' = \alpha' \epsilon', \epsilon'' = \alpha'' \epsilon'', z_i = z, i = i + 1$, and go to step 2.

Comment. We now compute step size as in (2.1).

Step 10. Set $\lambda = 1$.

Step 11. Compute

$$7 \quad \begin{aligned} \Delta = & f^0(z + \lambda h(z)) + \frac{1}{\epsilon'} p'(z + \lambda h(z)) + \epsilon'' p''(z + \lambda h(z)) \\ & - f^0(z) - \frac{1}{\epsilon'} p'(z) - \epsilon'' p''(z) + \frac{1}{2} \|h(z)\|^2. \end{aligned}$$

Step 12. If $\Delta \leq 0$, set $z = z + (\lambda h(z))$ and go to step 8; else, set $\lambda = \beta \lambda$ and go to step 11. ■

A rather common approach to speeding up the convergence of penalty function methods is to use a superlinearly convergent algorithm for solving the sequence of problems. This approach can also be used to speed up algo-

rithm(1). The following algorithm makes use of the Polak-Ribière conjugate gradient method with restart, because we cannot justify convexity assumptions when using penalty functions for equality constraints. (When the conjugate gradient algorithm is restarted every $n + 1$ iterations, its convergence to a stationary point can be established without assuming convexity of the objective function (see (6.3.9)).)

- 8 **Algorithm** (recommended mixed penalty function method with conjugate gradient subprocedure).

Comment. Solves the problem, $\min\{f^0(z) | f^i(z) \leq 0, i = 1, 2, \dots, m, r^i(z) = 0, i = 1, 2, \dots, l\}$, where the $f^i(\cdot)$ and the $r^i(\cdot)$ are continuously differentiable. Requires the assumption that the Kuhn-Tucker constraint qualification is satisfied. For relevant theory see Section 4.1.

Step 0. Select a $\overset{0}{z} \in \mathbb{R}^n$; select $\alpha, \alpha', \alpha'' \in (0, 0.5)$, select a $\beta \in (0.5, 0.8)$.

Step 1. Set $z = \overset{0}{z}$ and set $i = 0$.

Step 2. Define the index sets I', I'' as in (2), (3); define the penalty functions $p'(z), p''(z)$ as in (4), (5).

Step 3. If $i = 0$, go to step 4; else, go to step 7.

Step 4. Compute $\nabla f^0(z), \nabla p'(z), \nabla p''(z)$.

Step 5. Set $\epsilon' = \|\nabla p'(z)\|/\|\nabla f^0(z)\|, \epsilon'' = \|\nabla f^0(z)\|/\|\nabla p''(z)\|$.

Step 6. Select an $\epsilon_0 \in (0.1, 1)$.

Step 7. Define

$$9 \quad \bar{f}^0(z) = f^0(z) + \frac{1}{\epsilon'} p'(z) + \epsilon'' p''(z).$$

Comment. We now apply a simplified form of algorithm (4.1) with reinitialization to the minimization of \bar{f}^0 .

Step 8. Compute $\nabla \bar{f}^0(z)$.

Step 9. Set $g = h = -\nabla \bar{f}^0(z)$; set $j = 0$.

Step 10. Set $q = 0$.

Step 11. Define $\theta : \mathbb{R}^1 \rightarrow \mathbb{R}^1$ by

$$10 \quad \theta(x) = \bar{f}^0(z + xh) - \bar{f}^0(z).$$

Step 12. Set $x = 0$.

Step 13. Compute

$$11 \quad \theta'(x) = \langle \nabla \bar{f}^0(z + xh), h \rangle.$$

Step 14. If $\theta'(x) = 0$, set $\mu = x$ and go to step 19; else, go to step 15.

Step 15. Set $\lambda = 1$.

Step 16. Compute Δ according to (4.4).

Step 17. If $\Delta \leq 0$, set $q = q + 1$, and go to step 18; else, set $\lambda = \beta\lambda$, and go to step 16.

Step 18. If $q < 2$, set $x = x - \lambda\theta'(x)$, and go to step 13; else, set $\mu = x - \lambda\theta'(x)$, and go to step 19.

Step 19. Set $z = z + \mu h$.

Step 20. Compute $\nabla f^0(z)$.

Step 21. If $\|\nabla f^0(z)\| > \epsilon_i$, go to step 22; else, set $\epsilon_{i+1} = \alpha\epsilon_i$, $\epsilon' = \alpha'\epsilon'$, $\epsilon'' = \alpha''\epsilon''$, $z_i = z$, $i = i + 1$, and go to step 2.

Step 22. Set $j = j + 1$.

Step 23. If $j < n$, go to step 24; else, go to step 9.

Step 24. Set

$$12 \quad g' = -\nabla f^0(z), \quad h' = g' + \frac{\langle g' - g, g' \rangle}{\|g\|^2} h.$$

Step 25. Set $g = g'$, set $h = h'$, and go to step 10. ■

C.6 Methods of Feasible Directions with Linear Search

In this category, we find the modified method of centers (4.2.47), variations of the time-varying method of feasible directions (4.3.20) (Zoutendijk), variations of the time-invariant method of feasible directions (4.3.26) (Polak),

```

X  METHOD OF FEASIBLE DIRECTIONS - ZOUTENDIJK
+  METHOD OF FEASIBLE DIRECTIONS - POLAK
△  MODIFIED METHOD OF CENTERS - POLAK

```

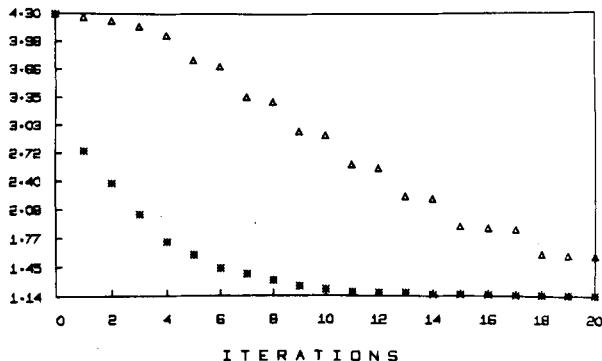


Fig. 7. A comparison of the Zoutendijk method of feasible directions (algorithm (4.3.20) modified to calculate step size according to (4.3.50), (4.3.51), with $\rho = 1$, $\alpha = 0.3$, $\beta = 0.8$, $\epsilon_0 = 0.1$), with the Polak method of feasible directions (algorithm (4.3.26) modified to calculate step size according to (4.3.50), (4.3.51), with $\rho = 1$, $\alpha = 0.3$, $\beta = 0.8$, $\epsilon_0 = 0.1$), and with the Polak implementation of the modified method of centers (algorithm (4.2.47), with $\rho = 1$, $\eta = 2.0$, $\epsilon_0 = 0.00001$). The problem: minimize $\exp[(z^1)^2 + 5(z^2)^2] + (z^1)^2 + 80(z^2)^2$, subject to $(z^1) + 2(z^2) - 1 \leq 0$, $(z^1)^2 + (z^2)^2 - 4z^1 + 1 \leq 0$, $(z^1)^2 + (z^2)^2 - z^1 - z^2 \leq 0$; $z^1 = 0.95$, $z^2 = 0.10$.

and both time-varying and time-invariant versions of methods of feasible directions (Zukhovitskii, Polyak and Primak) based on the subproblem (4.3.59)–(4.3.62). Figure 7 shows that, at least as far as our example is concerned, the rate of convergence of the modified method of centers is considerably worse than that of the Zoutendijk and Polak methods of feasible directions in the form described in (4.3.49). This is probably due to the fact that the step size rule in (4.2.47) is not as efficient as the one given in (4.3.49). In addition, the dimensions of the linear programming problem which one must solve at each iteration is larger in the modified method of centers than in most of the methods of feasible directions. All of these considerations indicate that the method of centers and its derivatives are less attractive than methods of feasible directions with an ϵ procedure.

We recommend two algorithms as having roughly equal merit. The first is a derivation of a cross between (4.3.20) and (4.3.26), and the other is similarly derived from the subproblem (4.3.70)–(4.3.72). The reason for crossing algorithms (4.3.20) and (4.3.26), which in our example (see Fig. 7) appear to be indistinguishable in performance, is that while the time-varying algorithm (4.3.20) tends to spend less time on each iteration than algorithm (4.3.26), because it does not always begin with the same value of ϵ (which then has to be adjusted to the right value), algorithm (4.3.26) is more prudent in a global sense.

1 Algorithm (recommended method of feasible directions with linear search, I).

Comment. Solves the problem, $\min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m, Az = b\}$, where the $f^i(\cdot)$ are continuously differentiable and A is a matrix ($z \in \mathbb{R}^n$).*

Step 0. Select an $\epsilon' > 0$, and $\epsilon'' \in (0, \epsilon')$, and $\alpha > 0$, a $\beta' \in (0, 1)$, a $\beta'' \in (0.5, 0.8)$, and an integer k satisfying $5 \leq k \leq 10$.

Step 1. If a vector z' satisfying $f^i(z') \leq 0$ for $i = 1, 2, \dots, m$, $Az' = b$, is available, go to step 5; else, go to step 2.

Comment. We begin by computing a feasible solution.

Step 2. Compute a vector z_0 satisfying $Az_0 = b$.

Step 3. Set $z_0^0 = \max\{f^i(z_0), i \in \{1, 2, \dots, m\}\}$; set $\bar{z}_0 = (z_0^0, z_0)$.

Step 4. Use steps 5–15 of algorithm (1) to solve the problem (in \mathbb{R}^{n+1}),

$$2 \quad \min\{z^0 \mid -z^0 + f^i(z) \leq 0, i = 1, 2, \dots, m, Az = b\},$$

using the initial feasible solution (z_0^0, z_0) , for a pair (z^0, z') , such that $f^i(z') \leq 0$, $i = 1, 2, \dots, m$, $Az' = b$.

Comment. A vector z' meeting the above specifications will be found after a finite number of iterations.

* It is understood that the set $\{z \mid f^i(z) \leq 0, i = 1, 2, \dots, m\}$ has an interior.

Step 5. Set $z_0 = z'$ and set $i = 0$.

Step 6. Set $\epsilon = \epsilon'$.

Step 7. Set $z = z_i$.

Step 8. Define the index sets $J_\epsilon^A, J_\epsilon^N$ as follows:

$$3 \quad J_\epsilon^A \cup J_\epsilon^N = \{0\} \cup \{j \in \{1, 2, \dots, m\} \mid f^j(z) + \epsilon \geq 0\},$$

$$4 \quad J_\epsilon^A \cap J_\epsilon^N = \emptyset,$$

$$5 \quad 0 \in J_\epsilon^N,$$

$$i \in J_\epsilon^A \text{ only if } f^i(\cdot) \text{ is affine.}$$

Step 9. Compute the vector $(h_\epsilon^0(z), h_\epsilon(z))$ by solving*

$$6 \quad \text{minimize} \quad h^0,$$

subject to

$$7 \quad -h^0 + \langle \nabla f^j(z), h \rangle \leq 0, \quad j \in J_\epsilon^N;$$

$$8 \quad \langle \nabla f^j(z), h \rangle \leq 0, \quad j \in J_\epsilon^A;$$

$$9 \quad |h^l| \leq 1, \quad l = 1, 2, \dots, n;$$

$$Ah = 0.$$

Step 10. If $h_\epsilon^0(z) \leq -\alpha\epsilon$, set $h(z) = h_\epsilon(z)$, and go to step 13; else, go to step 11.

Step 11. If $\epsilon \leq \epsilon''$, set $\tilde{\epsilon} = \epsilon$, solve (6)–(9) for $\epsilon = 0$, to obtain $(h_0^0(z), h_0(z))$, and go to step 12; else, set $\epsilon = \beta'\epsilon$, and go to step 9.

Step 12. If $h_0^0(z) = 0$, stop; else, set $\epsilon = \beta'\tilde{\epsilon}$, and go to step 9.

Step 13. Compute the smallest integer q such that

$$10 \quad f^0(z + (\beta'')^q h(z)) - f^0(z) - (\beta'')^q \frac{1}{2} \langle \nabla f^0(z), h(z) \rangle \leq 0,$$

$$11 \quad f^i(z + (\beta'')^q h(z)) \leq 0$$

$$\text{for } i = 1, 2, \dots, m.$$

Step 14. Set $z_{i+1} = z + (\beta'')^q h(z)$, and set $i = i + 1$.

Step 15. If $(i/k) = 0$ modulo k , go to step 6; else, go to step 7. ■

The following method of feasible directions allows us a greater amount of control over the part of the cone of feasible direction in which we shall pick a direction vector $h(z)$, and it is better suited for solving optimal control problems than algorithm (1). However (see proposition (4.3.74)), it really should only be used when the Kuhn-Tucker constraint qualification is satisfied by the problem that one wishes to solve.

* See (4.3.59)–(4.3.68) and (4.3.84).

12 Algorithm (recommended method of feasible directions with linear search, II).

Comment. Solves the problem, $\min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m, Az = b\}$, where the $f^i(\cdot)$ are continuously differentiable and A is a matrix ($z \in \mathbb{R}^n$). Requires that the Kuhn-Tucker constraint qualification be satisfied, see (4.3.74).

Step 0. Select an $\epsilon' > 0$, an $\epsilon'' \in (0, \epsilon')$, an $\alpha' > 0$, an $\alpha'' > 0$, a $\beta' \in (0, 1)$, a $\beta'' \in (0.5, 0.8)$, and an integer k satisfying $5 \leq k \leq 10$.

Step 1. Compute a vector z_0 satisfying $f^i(z_0) \leq 0, i = 1, 2, \dots, m, Az_0 = 0$; set $i = 0$.

Comment. For a method of computing z_0 , see steps 1–4 of algorithm (1).

Step 2. Set $\epsilon = \epsilon'$.

Step 3. Set $z = z_i$.

Step 4. Define the index sets $J_\epsilon^A, J_\epsilon^N$ as in (3)–(5).

Step 5. Compute the vector $(h_\epsilon^0(z), h_\epsilon(z))$ by solving the problem,

$$13 \quad \text{minimize} \quad \langle \nabla f^0(z), h \rangle,$$

subject to

$$14 \quad \langle \nabla f^j(z), h \rangle + \alpha' \epsilon \leq 0, \quad j \neq 0, \quad j \in J_\epsilon^N(z);$$

$$15 \quad \langle \nabla f^j(z), h \rangle \leq 0, \quad j \in J_\epsilon^A(z);$$

$$16 \quad |h^l| \leq 1, \quad l = 1, 2, \dots, n; \\ Ah = 0.$$

Step 6. If $h_\epsilon^0(z) \leq -\alpha'' \epsilon$, set $h(z) = h_\epsilon(z)$, and go to step 9; else, go to step 7.

Step 7. If $\epsilon \leq \epsilon''$, set $\bar{\epsilon} = \epsilon$, solve (13)–(16) for $\epsilon = 0$, to obtain $(\hat{h}_0^0(z), \hat{h}_0(z))$, and go to step 8; else, set $\epsilon = \beta' \epsilon$, and go to step 4.

Step 8. If $h_0^0(z) = 0$, stop; else, set $\epsilon = \beta' \bar{\epsilon}$, and go to step 4.

Step 9. Compute the smallest integer q such that (10) and (11) are satisfied.

Step 10. Set $z_{i+1} = z + (\beta'')^q h(z)$, and set $i = i + 1$.

Step 11. If $(i/k) = 0$ modulo k , go to step 2; else, go to step 3. ■

C.7 Methods of Feasible Directions with Quadratic Search

Of the various possibilities that we discussed in Section 4.4, we feel that the algorithm given below should be the most efficient one, because it is the simplest one.

1 Algorithm (recommended method of feasible directions with quadratic search).

Comment. Solves the problem, $\min\{f^0(z) \mid f^i(z) \leq 0, i = 1, 2, \dots, m, Az = b\}$, where $f^0(\cdot)$ is twice continuously differentiable, $f^i(\cdot)$, $i = 1, 2, \dots, m$, are continuously differentiable, and A is a matrix ($z \in \mathbb{R}^n$). Requires that the Kuhn-Tucker constraint qualification be satisfied, see (4.4.11), and that $H(z) = \partial^2 f^0(z)/\partial z^2$ be positive semidefinite on the set $\{z \mid f^0(z) \leq f^0(z_0); f^i(z) \leq 0, i = 1, 2, \dots, m; Az = b\}$, where z_0 is the initial feasible solution.

Step 0. Select an $\epsilon' > 0$, an $\epsilon'' \in (0, \epsilon')$, an $\alpha' > 0$, an $\alpha'' > 0$, a $\beta' \in (0, 1)$, a $\beta'' \in (0.5, 0.8)$, and an integer k satisfying $5 \leq k \leq 10$.

Step 1. Compute a vector z_0 satisfying $f^j(z_0) \leq 0, j = 1, 2, \dots, m, Az_0 = 0$; and set $i = 0$.

Comment. For a method of computing z_0 , see steps 1–4 of algorithm (6.1).

Step 2. Set $\epsilon = \epsilon'$.

Step 3. Set $z = z_i$.

Step 4. Define the index sets $J_\epsilon^A, J_\epsilon^N$ as in (6.3)–(6.5).

Step 5. Compute the vector $(\hat{h}'^0(z), \hat{h}'(z))$ by solving the problem,

$$2 \quad \text{minimize} \quad \langle \nabla f^0(z), h \rangle + \frac{1}{2} \langle h, H(z) h \rangle,$$

subject to

$$3 \quad \langle \nabla f^j(z), h \rangle + \alpha' \epsilon \leq 0, \quad j \neq 0, \quad j \in J_\epsilon^N(z);$$

$$4 \quad \langle \nabla f^j(z), h \rangle \leq 0, \quad j \in J_\epsilon^A(z);$$

$$5 \quad |h^l| \leq 1, \quad l = 1, 2, \dots, n; \\ Ah = b.$$

Step 6. If $\hat{h}'^0(z) \leq -\alpha'' \epsilon$, set $h(z) = \hat{h}'(z)$, and go to step 9; else, go to step 7.

Step 7. If $\epsilon \leq \epsilon''$, set $\bar{\epsilon} = \epsilon$, solve (2)–(5) for $\epsilon = 0$, to obtain $(\hat{h}'^0(z), \hat{h}'(z))$, and go to step 8; else, set $\epsilon = \beta' \epsilon$, and go to step 4.

Step 8. If $\hat{h}'^0(z) = 0$, stop; else, set $\epsilon = \beta' \bar{\epsilon}$, and go to step 4.

Step 9. Compute the smallest integer q such that (6.10) and (6.11) are satisfied.

Step 10. Set $z_{i+1} = z + (\beta'')^q h(z)$, and set $i = i + 1$.

Step 11. If $(i/k) = 0$ modulo k , go to step 2; else, go to step 3. ■

One may expect that algorithm (1) will converge faster than either algorithm (6.1) or (6.12), particularly when the sets $\{z \mid f^0(z) \leq d\}$ are long and narrow (i.e., when the eigenvalues of the Hessian $H(z) = \partial^2 f^0(z)/\partial z^2$ are wide apart, say, $\lambda_{\max}/\lambda_{\min} > 20$). However, this advantage is considerably dampened by the fact that the quadratic programming problem (2)–(5) takes a great deal more time to solve than either of the linear programming problems (6.6)–(6.9), (6.13)–(6.16).

REFERENCES

- A1. A. A. Abramov, On the transfer of boundary conditions for systems of ordinary linear differential equations, *Zh. Vychisl. Mat. Mat. Fiz.* **1** (3), 542–545 (1961).
- A2. H. A. Antosiewicz, Newton's method and boundary value problems, *J. Comput. Syst. Sci.* **2** (2), 177–202 (1968).
- A3. K. Arrow, L. Hurwitz and H. Uzawa, "Studies in Linear and Nonlinear Programming," Stanford Univ. Press, Stanford, California, 1958.
- A4. L. Armijo, Minimization of functions having continuous partial derivatives, *Pacific J. Math.* **16**, 1–3 (1966).
- B1. A. V. Balakrishnan, On a new computing technique in optimal control, *SIAM J. Contr.* **6** (2), 149–173 (1968).
- B2. N. V. Banitchouk, V. M. Petrov and R. L. Chernousko, Numerical solution of problems with variational limits by the method of local variations, *Zh. Vychisl. Mat. Mat. Fiz.* **6** (6), 947–961 (1966).
- B3. R. O. Barr and E. G. Gilbert, Some iterative procedures for computing optimal controls, *Proc. Third Congr. IFAC, London, June 20–25, 1966*, paper no. 24D.
- B4. R. O. Barr, "Computation of Optimal Controls by Quadratic Programming on Convex Reachable Sets," Ph.D. Dissertation, University of Michigan, Ann Arbor, Michigan 1966.
- B5. R. S. Bucy, Two point boundary value problems of linear Hamiltonian systems, *J. SIAM Appl. Math.* **15** (6), 1385–1389 (1967).
- B6. R. Bellman and R. Kalaba, "Quasilinearization and Boundary Value Problems," Elsevier, New York, 1965.
- B7. Bui-Trong-Lieu and P. Huard, La méthode des centres dans un espace topologique, *Num. Mat.* **8**, 65–67 (1966).
- B8. C. Berge, "Topological Spaces," Chap. VIII, Macmillan, New York, 1963.
- C1. M. D. Canon, C. D. Cullum and E. Polak, "Theory of Optimal Control and Mathematical Programming," McGraw-Hill, New York, 1970.
- C2. A. L. Cauchy, Méthode générale pour la résolution des systèmes d'équations simultanées, *Compt. Rend.* **25**, 536–38 (1847).
- C3. Y. Cherauault, Une méthode directe de minimization et applications, *Rev. Fr. Inform. Rech. Operation.* (10), 31–52 (July 1968).
- C4. A. Cohen, "Rate of Convergence for Root Finding and Optimization Algorithms," Ph. D. Dissertation, University of California, Berkeley, 1970.

- C5. R. Courant, Variational methods for the solution of problems of equilibrium and vibrations, *Bull. Am. Math. Soc.* **49**, 1–23 (1943).
- C6. J. Cullum, Penalty functions and nonconvex continuous optimal control problems, in “Computing Methods in Optimization Problems-2” (L. A. Zadeh, L. W. Neustadt and A. V. Balakrishnan, eds.), pp. 55–67, Academic Press, New York, 1969.
- C7. H. Curry, The method of steepest descent for nonlinear minimization problems, *Quart. Appl. Math.* **2**, 258–261 (1944).
- D1. J. W. Daniel, The conjugate gradient method for linear and nonlinear operator equations, *SIAM J. Num. Anal.* **4** (1), 10–28 (1967).
- D1a. J. W. Daniel, Convergence of the conjugate gradient method with computationally efficient modifications, *Num. Mat.* (10), 125–131 (1967).
- D2. W. C. Davidon, “Variable Metric Methods for Minimization,” AEC Research and Development Rept. ANL 5990 (Rev.), 1959.
- D3. J. Dieudonné, “Foundations of Modern Analysis,” Academic Press, New York, 1969.
- E1. J. H. Eaton, An iterative solution to time-optimal control, *J. Math. Anal. Appl.* **5** (2), 329–44 (1962).
- E2. R. M. Elkin, “Convergence Theorems for Gauss-Seidel and Other Minimization Algorithms,” Tech. Rept. 68–59, University of Maryland Computer Science Center, College Park, Maryland, January 1968.
- F1. A. V. Fiacco and G. P. McCormack, The sequential unconstrained minimization technique for nonlinear programming, a primal-dual method, *Management Sci.* **10** (2), 360–366 (1964).
- F2. A. V. Fiacco and G. P. McCormack, “Nonlinear Programming,” Wiley, New York, 1968.
- F3. R. Fletcher and M. J. D. Powell, A rapidly convergent descent method for minimization, *Comput. J.* **6**, 163–168 (1963).
- F4. R. Fletcher and C. M. Reeves, Function minimization by conjugate gradients, *Comput. J.* **7** (2), 149–154 (1964).
- F5. M. Frank and P. Wolfe, An algorithm for quadratic programming, *Nav. Res. Logistics Quart.* **3**, 95–110 (1956).
- G1. E. G. Gilbert, An iterative procedure for computing the minimum of a quadratic form on a convex set, *SIAM J. Contr.* **4** (1), 61–80 (1966).
- G2. A. A. Goldstein and J. F. Price, An effective algorithm for minimization, *Num. Mat.* **10**, 184–189 (1967).
- G3. A. A. Goldstein, “Constructive Real Analysis,” Harper, New York, 1967.
- G4. T. R. Goodman and G. N. Lance, Numerical integration of two point boundary value problems, *Math. Tables Aides Computat.* **10**, 82–86 (1956).
- H1. M. R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear equations, *J. Res. Nat. Bur. Stand.* **49**, 409 (1952).
- H2. M. R. Hestenes, The conjugate gradient method for solving linear systems, in “Proceedings of Symposia on Applied Mathematics,” Vol. VI, “Numerical Analysis,” pp. 83–102, McGraw-Hill, New York, 1956.
- H3. P. Huard, Programmation mathématique convexe, *Rev. Fr. Inform. Rech. Operation.* (7), 43–59 (1968).
- H4. J. Hurt, Some stability theorems for ordinary difference equations, *SIAM J. Num. Anal.* **4** (4), 582–596 (1967).
- I1. E. Isaacson and H. B. Keller, “Analysis of Numerical Methods,” Wiley, New York, 1966.
- J1. F. John, Extremum problems with inequalities as side conditions, in “Courant

- Anniversary Volume" (K. O. Friedrichs, O. E. Neugebauer and J. J. Stoker, eds.), pp. 187-204, Interscience, New York, 1948.
- K1. P. Kalfond, G. Ribière and J. C. Sogno, A method of feasible directions using projection operators, *Proc. IFIP Congr. 68, Edinburgh, August 1968*.
- K2. L. V. Kantorovich and G. P. Akilov, "Functional Analysis in Normed Spaces," Chap. 15, MacMillan, New York, 1964.
- K2a. R. Klessig, "Implementation of Conceptual Algorithms," Ph.D. dissertation, University of California, Berkeley, 1970.
- K2b. R. Klessig and E. Polak, "A Feasible Directions Algorithm Using Function Approximations, with Applications to Min Max Problems," Electronics Research Laboratory Memo No. M287, University of California, Berkeley, Oct. 1970.
- K2c. R. Klessig and E. Polak, "Efficient Implementations of the Polak-Ribière Conjugate Gradient Algorithm," Electronics Research Laboratory Memo No. M279, University of California, Berkeley, Aug. 1970.
- K3. N. N. Krassovskii, On an optimal control problem, *Prikl. Mat. Mekh.* **21** (5), 670-677 (1957).
- K4. H. W. Kuhn and A. W. Tucker, Nonlinear programming, in "Proceedings Second Berkeley Symposium on Mathematical Statistics and Probability," pp. 481-492, Univ. of California Press, Berkeley, California, 1951.
- L1. G. S. Lee, "Quasilinearization and Invariant Imbedding, with Applications to Chemical Engineering and Adaptive Control," Academic Press, New York, 1968.
- L2. E. S. Levitin and B. T. Polyak, Constrained minimization methods, *Zh. Vychisl. Mat. Mat. Fiz.* **6** (5), 787-823 (1966).
- L3. F. A. Lootsma, "Constrained Optimization via Penalty Functions," M. S. 5814, Philips Res. Repts., Eindhoven, Holland, 1968.
- M1. G. P. McCormack and W. I. Zangwill, "A Technique for Calculating Second Order Optima," Research Analysis Corp., McLean, Virginia, 1967 (mimeo).
- M2. R. McGill and P. Kenneth, A convergence theorem on the iterative solution of nonlinear two-point boundary-value systems, in "Proceedings XIV International Astronautical Congress, Paris 1963," Vol. 4, pp. 173-188, Gauthier-Villars, Paris, 1965.
- M3. O. L. Mangasarian and S. Fromovitz, The Fritz John necessary optimality conditions in the presence of equality and inequality constraints, *J. Math. Anal. Appl.* **17**, 37-47 (1967).
- M4. O. L. Mangasarian, "Nonlinear Programming," McGraw-Hill, New York, 1969.
- M5. D. Q. Mayne, A second-order gradient method for determining optimal trajectories of nonlinear discrete-time systems, *Int. J. Contr.* **3** (1), 85-95 (1966).
- M6. G. Meyer and E. Polak, A decomposition algorithm for solving a class of optimal control problems, *J. Math. Anal. Appl.* **3** (1), 118-140 (1970).
- M7. G. E. Meyer, "Properties of the Conjugate Gradient and Davidon Methods," Analytical Mechanics Associates, Inc., Wesbury, New York, 1967 (mimeo).
- M8. G. Meyer and E. Polak, "Abstract Models for the Synthesis of Optimization Algorithms," Electronics Research Laboratory Memo No. ERL-268, University of California, Berkeley, October 1969.
- M9. N. N. Moiseev, "Lecture Notes on Computational Methods," Institut de Recherche d'Informatique et d'Automatique, Versailles, France, Spring 1968.
- N1. L. W. Neustadt, Synthesis of time-optimal control systems, *J. Math. Anal. Appl.* **1**, 484-492 (1960).
- O1. A. M. Ostrowski, "Solution of Equations and Systems of Equations," 2nd ed., Academic Press, New York, 1966.

- P1. E. Polak, On the convergence of optimization algorithms, *Rev. Fr. Inform. Rech. Operation.* (16-R1), 17–34 (1969).
- P2. E. Polak, On primal and dual methods for solving discrete optimal control problems, in "Computing Methods in Optimization Problems-2" (L. A. Zadeh, L. W. Neustadt and A. V. Balakrishnan, eds.), pp. 317–331, Academic Press, New York, 1969.
- P3. E. Polak, "Computational Methods in Discrete Optimal Control and Nonlinear Programming: A Unified Approach," Electronics Research Laboratory Memo No. ERL-M261, University of California, Berkeley, February 1969.
- P3a. E. Polak, On the implementation of conceptual algorithms, *Proc. Nonlinear Programming Symp. Univ. Wisc., Madison, Wisc., May 4–6, 1970* (in press).
- P4. E. Polak and M. Deparis, An algorithm for minimum energy control, *IEEE Trans. AC-14* (4), 367–378 (1969).
- P5. E. Polak and G. Ribiére, Note sur la convergence de méthodes de directions conjuguées, *Rev. Fr. Inform. Rech. Operation.* (16-R1), 35–43 (1969).
- P6. B. T. Polyak, Gradient methods for the minimization of functionals, *Zh. Vychisl. Mat. Mat. Fiz.* 3 (4), 643–653 (1963).
- P7. L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mishchenko, "The Mathematical Theory of Optimal Processes," Wiley (Interscience), New York, 1962.
- P8. M. J. D. Powell, A survey of numerical methods for unconstrained optimization, *SIAM Rev.* 12 (1), 79–97 (1970).
- P9. M. J. D. Powell, "On the Convergence of the Variable Metric Algorithm," Mathematics Branch, Atomic Energy Research Establishment, Harwell, Berkshire, England, October 1969 (mimeo).
- R1. J. B. Rosen, The gradient projection method for nonlinear programming, Part I: Linear constraints, *J. SIAM* 8 (1), 181–217 (1960).
- R2. D. L. Russell, Penalty functions and bounded phase coordinate control, *J. SIAM Contr.* 2, 409–422 (1965).
- R3. R. T. Rockafellar, "Convex Analysis," Princeton University Press, Princeton, N.J., 1970.
- T1. D. M. Topkis and A. Veinott, Jr., On the convergence of some feasible directions algorithms for nonlinear programming, *J. SIAM Contr.* 5 (2), 268–79 (1967).
- T2. R. Tremolières, Méthode des centres à troncature variable, *Elec. Fr. Bull. Dir. Etudes Rech. Ser. C-Math. Inform.* (2), 57–64 (1968).
- T3. J. F. Traub, "Iterative Methods for the Solution of Equations," Prentice Hall, Englewood Cliffs, N.J., 1964.
- V1. R. M. Van Slyke, Generalized upper bounding techniques, *J. Comput. Syst. Sci.* 1 (3), 213–226 (1967).
- V2. R. M. Van Slyke and G. B. Dantzig, Generalized linear programming and decomposition theory, in "Multilevel Control Systems" (D. Wismer, ed.), Chap. 5, McGraw-Hill, New York, 1969.
- V3. P. P. Varaiya, "A Decomposition Technique for Nonlinear Programming," IBM Research Rept. RJ-345, July 1, 1965. (Also in L. A. Zadeh and E. Polak, eds., "System Theory," Chap. 12, McGraw-Hill, New York, 1969.)
- W1. P. Wolfe, The simplex method for quadratic programming, *Econometrica* 28 (3), 382–398 (1959).
- W2. P. Wolfe, "On the Convergence of Gradient Methods Under Constraints," IBM Research Rept. RC 1752, Yorktown Heights, New York, January 24, 1967.
- Z1. W. I. Zangwill, "Nonlinear Programming: A Unified Approach," Prentice-Hall, Englewood Cliffs, New Jersey, 1969.

- Z2. W. I. Zangwill, "Applications of the Convergence Conditions," presented at the Sixth Symposium on Mathematical Programming, August 14-18, 1967, Princeton University, Princeton, New Jersey.
- Z3. W. I. Zangwill, Nonlinear programming via penalty functions, *Management Sci.-A* **13** (5), 344-358 (1967).
- Z4. G. Zoutendijk, "Methods of Feasible Directions," Elsevier, Amsterdam, 1960.
- Z5. G. Zoutendijk, Nonlinear programming: A numerical survey, *J. SIAM Contr.* **4**, 194-210 (1966).
- Z6. G. Zoutendijk, "Computational Methods in Nonlinear Programming," presented at the SIAM 1968 National Meeting, Toronto, Canada, June 11-14, 1968.
- Z6a. G. Zoutendijk, Professor of Mathematics, Leiden University, Leiden, Holland, personal communication, Nov. 1968.
- Z7. S. I. Zukhovitskii and L. I. Avdeyeva, "Linear and Convex Programming," Saunders, Philadelphia, Pennsylvania, 1966.
- Z8. S. I. Zukhovitskii, R. A. Polyak and M. E. Primak, An algorithm for the solution of convex programming problems, *DAN USSR* **153** (5), 991-1000 (1963).

This page intentionally left blank

INDEX

A

- Abramov, A.A., 89, 90, 111, 117, 121, 317
Abramov's method
for difference equations, 90–92, 121
for differential equations, 111–113, 121
Abstract problem, 13, 283, 289
Akilov, G.P., 38, 107, 319
Algorithm models,
 $z_{i+1} = a(z_i)$, $c(z_{i+1}) < c(z_i)$, 14
 $z_{i+1} \in A(z_i)$, $c(z_{i+1}) < c(z_i)$, 15
for adaptive implementation
 $z_{i+1} \in A_\epsilon(z_i)$, $c(z_{i+1}) - c(z_i) \leq -\alpha\epsilon$, 17
 $z_{i+1} \in A(\epsilon, z_i)$, $c(z_{i+1}) - c(z_i) \leq -\alpha\epsilon$,
19
 $z_{i+1} \in A(\epsilon, z_i)$, $c(\epsilon, z_{i+1}) - c(\epsilon, z_i) \leq -\alpha\epsilon$, 283
for open-loop implementation
 $z_{i+1} \in A(j, z_i)$, $c(z_{i+1}) < c(z_i)$, 289
for time-varying methods of feasible
directions, 22
Antosiewicz, H.A., 38, 107, 317
Armijo, L., 36, 317
Arrow, K., 17, 317
Avdeyeva, L.I., 320

B

- Balakrishnan, A.V., 149, 150, 317
Banitchouk, N.V., 42, 43, 317
Barr, R.O., 236, 317
Bellman, R., 84, 87, 103, 106, 109, 317

C

- Canon, M.D., 3, 7, 208–211, 317
Cauchy, A.L., 37, 317
Centers, *see* Method of centers

- Chernousko, R.L., 42, 43, 317
 Cheruault, Y., 42, 317
 Cohen, A., 42, 259, 260, 268, 317
Conjugate gradient methods
 assumptions on problem
 for convergence, 46, 58
 for rate of convergence, 259, 269
 computational aspects, 65–66, 306–309
 Davidon algorithm, *see* variable metric algorithm
 Fletcher-Powell algorithm, *see* variable metric algorithm
 Fletcher-Reeves algorithm, 52
 for continuous optimal control problem, 77–78
 convergence, 48, 53
 for quadratic function, 52
 for discrete optimal control problem, 69
 implementation, 66
 with reinitialization, 66, 267–268
 convergence, 268
 rate of convergence, 268
 Polak-Ribi re algorithm, 53
 for continuous optimal control problem, 78
 convergence, 54–55
 for quadratic function, 54
 for discrete optimal control problem, 69
 implementations
 with Golden section step size search, 66
 with gradient method step size search, 306–307
 rate of convergence, 242–247
 with reinitialization, 66, 259
 convergence, 260
 rate of convergence, 260–267
 prototype algorithm, 46
 convergence, 46, 48
 rate of convergence, 242–246
 variable metric algorithm, 56
 assumptions on problem, 46, 58, 269
 convergence, 59–65
 implementations
 with Golden section step size search, 66
 with gradient method step size search, 308
 positive definiteness of the matrices H_i , 57
 rate of convergence, 268–282
Contact map, 216
Continuous optimal control problem, 2
 algorithms for
 conjugate gradient, 77–78
 feasible directions, 179–180
 gradient, 76
 gradient projection, 205–207
 penalty function, 145–150
 quasi-linearization, 115–117
 conversion of free-time problem to
 fixed-time problem, 72
 discretizations, 5–6
 with linear dynamics
 algorithms for
 minimum energy, 236
 minimum time, 240
 quadratic cost, 121–125
 optimality conditions for, 11
Convergence theorems for
 algorithm models, 14, 15, 17, 19, 21,
 22, 26, 27, 286, 289
conjugate gradient algorithms
 Davidon-Fletcher-Powell, 58–65
 Fletcher-Reeves, 48, 53, 268
 Polak-Ribi re, 54, 56, 260
 prototype, 46, 48
 variable metric, 58–65
decomposition algorithms
 of dual type, 220
 of primal type, 238
gradient algorithms
 with minimization step size search, 29
 with two-line step size search, 33
gradient projection algorithm, 194
hybrid gradient projection algorithm, 203
method of local variations, 43
methods of centers
 modified algorithm, 156
 theoretical algorithm, 152
methods of feasible directions
 Polak, 165
 Topkis-Veinott, 160, 156
 Zoutendijk, 163
 Zukhovitskii-Polyak-Primak (ZPP),
 172
Newton-Raphson, 38, 81, 252
penalty function methods

- exterior, 130
- exterior-interior, 137
- interior, 135
- modified exterior, 141
- quasi-Newton algorithm, 33
- variable metric algorithm, 58–65
- Convex functions, 294–295
- Courant, R., 127, 318
- Cullum, C.D., 3, 7, 208–211, 317
- Cullum, J., 148, 318
- Curry, H., 37, 318

- D**
- Daniel, J.W., 58, 259, 318
- Dantzig, G.B., 154, 161, 210, 320
- Davidon, W.C., 56, 318
- Decomposition algorithms
 - of dual type, 219, 226, 228
 - convergence, 220–223
 - of primal type, 236, 240
 - convergence, 238
- Deparis, M., 219, 319
- Desirable point, 13
- Dieudonné, J., 73, 292, 318
- Discrete optimal control problem, 1
 - algorithm model for, 283
 - algorithms for
 - conjugate gradient, 69
 - gradient, 69
 - gradient projection, 205
 - Newton-Raphson, 92–102
 - penalty function, 145–146
 - quasi-Newton, 92–102
 - with linear dynamics, algorithms for
 - piecewise linear cost, 208–211
 - quadratic cost, 100–102, 208–212, 223
- optimality conditions for, 10–11, 213–214
- transcriptions
 - into linear programming problem, 209–210
 - into nonlinear programming problem, 3–4
 - into quadratic programming problem, 210–211
- Discretizations of continuous optimal control problems, 5, 6, 283
- Distance function $d(\cdot, \cdot)$, 151
- Dual problem, 217
 - algorithm for, 219

- E**
- Eaton, J.H., 219, 318
- Elkin, R.M., 42, 44, 318

- F**
- Feasible directions, *see* Methods of feasible directions
- Fiacco, A.V., 127, 140, 309, 318
- Fibonacci fractions, 31
- Fletcher, R., 52, 54, 56, 57, 58, 77, 267, 306, 318
- Frank, M., 236, 318
- Fromovitz, S., 319

- G**
- Gamkrelidze, R.V., 11, 320
- Geometric problem, 234
 - algorithm for, 236
 - convergence, 238
- Gilbert, E.G., 236, 317, 318
- Golden section search, 31
 - in conjugate gradient methods, 66
 - in gradient methods, 32
 - in modified method of centers, 158
- Goldstein, A.A., 32, 33, 36, 39, 41, 256, 318
- Goodman, T.R., 84, 86, 103, 106, 318
- Goodman-Lance algorithm
 - for difference equations, 84–88
 - for differential equations, 103–106
- Gradient, calculation of
 - in continuous optimal control problems, 73–75
 - in discrete optimal control problems, 67–69
- Gradient methods
 - algorithms
 - for continuous optimal control problems, 76
 - with derivative approximations, 40
 - for discrete optimal control problems, 69
 - with Golden section step size search, 32
 - with minimization step size search, 28
 - convergence, 28

- rate of convergence, 242–246
 - with one-line step size search, 36, 37, 304
 - convergence, 36
 - rate of convergence, 247–248
 - with two-line step size search, 33, 35
 - convergence, 33–35
 - rate of convergence, 248–249
 - assumptions on problem
 - convergence, 28
 - for convergence, 243
 - Gradient projection algorithms
 - for affine constraints, 190
 - accelerated version, 201
 - assumptions on problem, 190
 - convergence, 194–200
 - elementary method, 192
 - time-varying versions, 200
 - for convex constraints, 202
 - assumptions on the problem, 202
 - convergence, 203–205
 - for optimal control problems, 205–207
- H**
- Hestenes, M.R., 32, 45, 48, 52, 318
 - Huard, P., 150–155, 158, 318
 - Hurt, J., 17, 318
 - Hurwiz, L., 17, 317
- I**
- Implementations for
 - conjugate gradient algorithms, 66, 306–309
 - decomposition algorithm of dual type, 226, 228
 - modified method of centers, 158
 - penalty function algorithms, 141, 144, 309
 - steepest descent algorithm, 32, 40
 - Isaacson, E., 253, 318
- J**
- John, F., 7, 318
- K**
- Kalaba, R., 84, 87, 103, 106, 109, 317
 - Kalfond, P., 202, 319
 - Kantorovich, L.V., 38, 107, 319
 - Keller, H.B., 253, 318
 - Kenneth, P., 84, 87, 103, 109, 319
 - Klessig, R., 22, 283, 306, 319
 - Krassovskii, N.N., 219, 319
 - Kuhn, H.W., 7, 319
- L**
- Lance, G.N., 84, 86, 103, 106, 318
 - Lee, G.S., 84, 319
 - Levitin, E.S., 17, 319
 - Local variations, *see* Method of local variations
 - Lootsma, F.A., 127, 309, 319
 - Lyapunov's method, 17
- M**
- McCormack, G.P., 127, 140, 309, 318, 319
 - McGill, R., 84, 87, 103, 109, 319
 - Mangasarian, O.L., 7, 212, 319
 - Maximum principle, 11
 - Mayne, D.Q., 40, 319
 - Mean-value theorems, 292
 - Method of centers
 - algorithm, 152
 - convergence, 152
 - distance function $d(\cdot, \cdot)$, 151
 - assumptions on problem, 151
 - modified, *see* Modified method of centers
 - Method of local variations, 43
 - convergence, 43
 - Method of steepest descent, 28
 - convergence, 28
 - implementation, 32
 - with derivative approximation, 40
 - rate of convergence, 242–246
 - Methods of feasible directions
 - algorithms
 - for optimal control problems, 176–180
 - Polak
 - for affine constraints, 171
 - convergence, 165–168
 - hybrid, for problems with equality constraints, 174–175
 - with minimization step size search, 164
 - with one-line step size search, 169, 313
 - with two-line step size search, 169
 - with quadratic search, 315
 - Topkis-Veinott, 160

- convergence, 160, 156
- Zoutendijk, 163
- convergence, 163
- Zukhovitskii-Polyak-Primak (ZPP), 171–172, 315
- for continuous optimal control problems, 179
 - convergence, 172
- assumptions on problem, 151, 162
- Meyer, G., 236, 291, 319
- Meyer, G.E., 57, 319
- Minimum energy problems
- continuous, 121–125, 231
 - discrete, 100–103, 210–211, 239
- Minimum time problem, 229
- Mishchenko, E.F., 11, 320
- Modified method of centers, 155
- assumptions on problem, 151, 155
 - convergence, 156
 - distance function $d(\cdot, \cdot)$, 151
 - implementation, 158
- Moiseev, N.N., 89, 149, 319
- N**
- Neustadt, L.W., 219, 319
- Newton-Raphson
- algorithm for
 - continuous optimal control problems, 114–117
 - discrete optimal control problems, 92–95
 - minimization, 38
 - convergence, 38, 252
 - rate of convergence, 252–254, 256
- root finding, 81, 252
- convergence, 81, 252
 - rate of convergence, 252–253
- algorithm in Banach space, 107
- algorithms for boundary-value problems
- Goodman-Lance version
 - with difference equations, 83, 84, 86, 88
 - with differential equations, 103–106
 - quasi-linearization version
 - with difference equations, 87–88
 - with differential equations, 107–109
- quasi-Newton algorithm, *see* Quasi-Newton methods
- Nonlinear programming problem, 1
- O**
- Optimality conditions for
- continuous optimal control problem, 11
 - discrete optimal control problem, 10
 - nonlinear programming problems, 7–10, 81, 154, 170, 174, 181–183, 190
- Optimality functions
- $h^0(\cdot)$, 154
 - $h_\epsilon^0(\cdot)$, 161, 162
 - $\hat{h}_\epsilon^0(\cdot)$, 170, 171
 - $\hat{h}^0(\cdot)$, 174
 - $\hat{h}_\epsilon^0(\cdot)$, 172
 - $h'^0(\cdot)$, 181
 - $h'_\epsilon^0(\cdot)$, 181
 - $\hat{h}'^0(\cdot)$, 183
 - $\hat{h}'_\epsilon^0(\cdot)$, 181
 - $\hat{h}''^0(\cdot)$, 183
 - $\hat{h}''_\epsilon^0(\cdot)$, 181
- Ostrowski, A.M., 250, 319
- P**
- Penalty function methods
- computational aspects, 138–145, 309
 - exterior, 127
 - assumptions on problem, 128
 - convergence, 130
 - modified (Polak), 141–144
 - for optimal control problems, 132, 145–150
- exterior-interior, 136
- assumptions on problem, 136
 - convergence, 137
 - modified (Polak), 309
- interior, 133
- assumptions on problem, 133
 - convergence, 135
 - modified (Polak), 144–145
- Penalty functions
- exterior, 128
 - for continuous optimal control problems, 132, 147
 - for equality constraints, 131
 - for inequality constraints, 130
- exterior-interior, 136
- interior, 134
- for inequality constraints, 136
- Petrov, V.M., 42, 43, 317

- Polak, E., 3, 7, 17, 32, 40–42, 53, 141, 144, 158, 164, 175, 189, 190, 202, 208–211, 219, 226, 228, 236, 243, 259, 306, 312, 317, 319, 320
 Polyak, B.T., 17, 319, 320
 Polyak, R.A., 17, 160, 170, 171, 313, 320
 Polytope, 192
 Pontryagin, L.S., 11, 320
 maximum principle, 11
 Powell, M.J.D., 44, 56, 58, 61, 268–282, 318, 320
 Price, J.F., 32, 41, 318
 Primak, M.E., 17, 160, 170, 171, 313, 320
 Primal problem, 214
 algorithm for, 236
 Problem
 abstract, 13, 283, 289
 continuous optimal control, 2
 discrete optimal control, 1
 dual, 217
 geometric, 234
 nonlinear programming, 1
 primal, 214
- Q**
- Quadratic cost
 continuous optimal control problem, 121–125, 231
 discrete optimal control problem, 100–103, 210–211, 223–225, 239
 Quasi-linearization algorithm
 for difference equations, 87–88
 for differential equations, 107–109
 Quasi-Newton methods
 for boundary-value problems, *see* Boundary value problems
 algorithm
 with approximation of Hessian, 41
 with one-line step size search, 304
 convergence, 36
 rate of convergence, 253
 with two-line step size search, 39
 convergence, 33
 rate of convergence, 253

R

- Rate of convergence
 for conjugate gradient algorithms

- Davidon–Fletcher–Powell, 278–282
 Fletcher–Reeves with reinitialization, 268
 Polak–Ribi  re, 242–247
 Polak–Ribi  re with reinitialization, 260–267
 variable metric, 278–282
 for gradient algorithms
 with minimization step size search, 242–247
 with one-line step size search, 248–249
 with two-line step size search, 247–248
 Reeves, C.M., 52, 77, 267, 306, 318
 Ribi  re, G., 53, 202, 243, 259, 306, 319
 Riccati equations
 in continuous optimal control, 119–125
 existence of solution, 121–125
 in discrete optimal control, 97–102
 existence of solution, 101–102
 Rockafellar, R.T., 212, 320
 Rosen, J.B., 189, 194, 201, 320
 Russell, D.L., 148, 320

S

- Second derivative, 293
 matrix of second derivatives for optimal control problem, 70
 norm of, 293
 Sogno, J.G., 202, 319
 Steepest descent, *see* Method of steepest descent
 Stiefel, E., 32, 45, 52, 318

T

- Taylor's formula
 for first-order expansions, 292
 for second-order expansions, 293
 Topkis, D.M., 17, 160, 320
 Tremoli  res, T., 151, 320
 Tucker, A.W., 7, 9, 319

U

- Uzawa, H., 17, 317

V

Van Slyke, R.M., 154, 161, 210, 320
Varaiya, P.P., 17, 320
Veinott, A., 17, 160, 320

W

Wolfe, P., 161, 208, 236, 318, 320

Z

Zangwill, W.I., 16, 17, 127, 320
Zoutendijk, G., 8, 17, 48, 159, 162, 163,
170, 312, 320
Zukhovitskii, S.I., 17, 160, 170, 171, 313,
320

This page intentionally left blank

Mathematics in Science and Engineering

A Series of Monographs and Textbooks

Edited by RICHARD BELLMAN, *University of Southern California*

1. T. Y. Thomas. Concepts from Tensor Analysis and Differential Geometry. Second Edition. 1965
2. T. Y. Thomas. Plastic Flow and Fracture in Solids. 1961
3. R. Aris. The Optimal Design of Chemical Reactors: A Study in Dynamic Programming. 1961
4. J. LaSalle and S. Lefschetz. Stability by Liapunov's Direct Method with Applications. 1961
5. G. Leitmann (ed.). Optimization Techniques: With Applications to Aerospace Systems. 1962
6. R. Bellman and K. L. Cooke. Differential-Difference Equations. 1963
7. F. A. Haight. Mathematical Theories of Traffic Flow. 1963
8. F. V. Atkinson. Discrete and Continuous Boundary Problems. 1964
9. A. Jeffrey and T. Taniuti. Non-Linear Wave Propagation: With Applications to Physics and Magnetohydrodynamics. 1964
10. J. T. Tou. Optimum Design of Digital Control Systems. 1963.
11. H. Flanders. Differential Forms: With Applications to the Physical Sciences. 1963
12. S. M. Roberts. Dynamic Programming in Chemical Engineering and Process Control. 1964
13. S. Lefschetz. Stability of Nonlinear Control Systems. 1965
14. D. N. Chorafas. Systems and Simulation. 1965
15. A. A. Pervozvanskii. Random Processes in Nonlinear Control Systems. 1965
16. M. C. Pease, III. Methods of Matrix Algebra. 1965
17. V. E. Benes. Mathematical Theory of Connecting Networks and Telephone Traffic. 1965
18. W. F. Ames. Nonlinear Partial Differential Equations in Engineering. 1965
19. J. Aczel. Lectures on Functional Equations and Their Applications. 1966
20. R. E. Murphy. Adaptive Processes in Economic Systems. 1965
21. S. E. Dreyfus. Dynamic Programming and the Calculus of Variations. 1965
22. A. A. Fel'dbaum. Optimal Control Systems. 1965
23. A. Halanay. Differential Equations: Stability, Oscillations, Time Lags. 1966
24. M. N. Oguztoreli. Time-Lag Control Systems. 1966
25. D. Sworder. Optimal Adaptive Control Systems. 1966
26. M. Ash. Optimal Shutdown Control of Nuclear Reactors. 1966
27. D. N. Chorafas. Control System Functions and Programming Approaches (In Two Volumes). 1966
28. N. P. Erugin. Linear Systems of Ordinary Differential Equations. 1966
29. S. Marcus. Algebraic Linguistics; Analytical Models. 1967
30. A. M. Liapunov. Stability of Motion. 1966
31. G. Leitmann (ed.). Topics in Optimization. 1967
32. M. Aoki. Optimization of Stochastic Systems. 1967
33. H. J. Kushner. Stochastic Stability and control. 1967
34. M. Urabe. Nonlinear Autonomous Oscillations. 1967
35. F. Calogero. Variable Phase Approach to Potential Scattering. 1967
36. A. Kaufmann. Graphs, Dynamic Programming, and Finite Games. 1967
37. A. Kaufmann and R. Cruon. Dynamic Programming: Sequential Scientific Management. 1967
38. J. H. Ahlberg, E. N. Nilson, and J. L. Walsh. The Theory of Splines and Their Applications. 1967

- 39.** Y. Sawaragi, Y. Sunahara, and T. Nakamizo. Statistical Decision Theory in Adaptive Control Systems. 1967
- 40.** R. Bellman. Introduction to the Mathematical Theory of Control Processes, Volume I. 1967; Volume II. 1971 (Volume III in preparation)
- 41.** E. S. Lee. Quasilinearization and Invariant Imbedding. 1968
- 42.** W. Ames. Nonlinear Ordinary Differential Equations in Transport Processes. 1968
- 43.** W. Miller, Jr. Lie Theory and Special Functions. 1968
- 44.** P. B. Bailey, L. F. Shampine, and P. E. Waltman. Nonlinear Two Point Boundary Value Problems. 1968
- 45.** Iu. P. Petrov. Variational Methods in Optimum Control Theory. 1968
- 46.** O. A. Ladyzhenskaya and N. N. Ural'tseva. Linear and Quasilinear Elliptic Equations. 1968
- 47.** A. Kaufmann and R. Faure. Introduction to Operations Research. 1968
- 48.** C. A. Swanson. Comparison and Oscillation Theory of Linear Differential Equations. 1968
- 49.** R. Hermann. Differential Geometry and the Calculus of Variations. 1968
- 50.** N. K. Jaiswal. Priority Queues. 1968
- 51.** H. Nikaido. Convex Structures and Economic Theory. 1968
- 52.** K. S. Fu. Sequential Methods in Pattern Recognition and Machine Learning. 1968
- 53.** Y. L. Luke. The Special Functions and Their Approximations (In Two Volumes). 1969
- 54.** R. P. Gilbert. Function Theoretic Methods in Partial Differential Equations. 1969
- 55.** V. Lakshmikantham and S. Leela. Differential and Integral Inequalities (In Two Volumes). 1969
- 56.** S. H. Hermes and J. P. LaSalle. Functional Analysis and Time Optimal Control. 1969
- 57.** M. Iri. Network Flow, Transportation, and Scheduling: Theory and Algorithms. 1969
- 58.** A. Blaquiere, F. Gerard, and G. Leitmann. Quantitative and Qualitative Games. 1969
- 59.** P. L. Falb and J. L. de Jong. Successive Approximation Methods in Control and Oscillation Theory. 1969
- 60.** G. Rosen. Formulations of Classical and Quantum Dynamical Theory. 1969
- 61.** R. Bellman. Methods of Nonlinear Analysis, Volume I. 1970
- 62.** R. Bellman, K. L. Cooke, and J. A. Lockett. Algorithms, Graphs, and Computers. 1970
- 63.** E. J. Beltrami. An Algorithmic Approach to Nonlinear Analysis and Optimization. 1970
- 64.** A. H. Jazwinski. Stochastic Processes and Filtering Theory. 1970
- 65.** P. Dyer and S. R. McReynolds. The Computation and Theory of Optimal Control. 1970
- 66.** J. M. Mendel and K. S. Fu (eds.). Adaptive, Learning, and Pattern Recognition Systems: Theory and Applications. 1970
- 67.** C. Derman. Finite State Markovian Decision Processes. 1970
- 68.** M. Mesarovic, D. Macko, and Y. Takahara. Theory of Hierarchical Multilevel Systems. 1970
- 69.** H. H. Happ. Diakoptics and Networks. 1971
- 70.** Karl Astrom. Introduction to Stochastic Control Theory. 1970
- 71.** G. A. Baker, Jr. and J. L. Gammel (eds.). The Padé Approximant in Theoretical Physics. 1970
- 72.** C. Berge. Principles of Combinatorics. 1971
- 73.** Ya. Z. Tsyplkin. Adaptation and Learning in Automatic Systems. 1971
- 74.** Leon Lapidus and John H. Seinfeld. Numerical Solution of Ordinary Differential Equations. 1971
- 75.** L. Mirsky. Transversal Theory, 1971
- 76.** Harold Greenberg. Integer Programming. 1971
- 77.** E. Polak. Computational Methods in Optimization: A Unified Approach, 1971
- 78.** Thomas G. Windeknecht. A Mathematical Introduction to General Dynamical Processes, 1971
- 79.** M. A. Aizerman, L. A. Gusev, L. I. Rozonoer, I. M. Smirnova, and A. A. Tal'. Logic, Automata, and Algorithms, 1971
- In preparation**
- Andrew P. Sage and James L. Melsa. System Identification
- R. Boudarel, J. Delmas, and P. Guichet. Dynamic Programming and Its Application to Optimal Control
- William Stenger and Alexander Weinstein. Methods of Intermediate Problems for Eigenvalues Theory and Ramifications