

工・土木

Nonlinear Programming: Sequential Unconstrained Minimization Techniques

ANTHONY V. FIACCO
GARTH P. McCORMICK

RESEARCH ANALYSIS CORPORATION
MCLEAN, VIRGINIA

JOHN WILEY AND SONS, INC.
New York • London • Sydney • Toronto



2 3 4 5 6 7 8 9 10

Copyright © 1968 by Research Analysis Corporation

Reproduction in whole or in part is permitted for any purpose
of the United States Government without payment of royalties.

Preparation of this book was undertaken in the course of
research sponsored by contract DA-44-188-ARO-1 between the
U.S. Army Research Office and the Research Analysis Corporation,
Library of Congress Catalog Card Number: 68-30909 SBN 471 25810 5

Printed in the United States of America

exterior point methods in [48], leading to a class of new procedures that may aptly be described as "exterior centers methods."

Further connections analogous to the above have also been indicated in results recently obtained by Falk [42].

7.3 CONTINUOUS VERSION OF INTERIOR POINT TECHNIQUES

A natural variation of interior point unconstrained techniques is to let the parameter r vary continuously as a decreasing function of time and solve a differential equation of the form $dx/dt = -\nabla_x U(x, t)$. (Considered here is Problem B with inequality constraints only.) Motion is analogous to Cauchy's method of steepest descent (Section 8.2). Such a method is suitable for use on an analog computer.

For definiteness let $r = e^{-t}$, and let U be the P function of (3.10). Then $P(x, t) = f(x) + e^{-t} \sum_{i=1}^m 1/g_i(x)$, and

$$\dot{x} = -\left[\nabla f(x) - e^{-t} \sum_{i=1}^m \frac{1}{g_i^2(x)} \nabla g_i(x) \right]. \quad (7.5)$$

One can, under the same general conditions of Theorem 8, Section 3.3, prove that solving (7.5) results in the generation of points satisfying the first-order conditions of (2.5), (2.7), (2.8), and (2.9), Section 2.1. This is done in Theorem 33. Fortunately, the first-order necessary conditions are sufficient that a point be a global solution of the convex programming problem. Hence in Theorem 34 convergence to a global solution is proved.

Theorem 33 [Stability of Stationary Points of the Lagrangian]. If (a) f, g_1, \dots, g_m are continuously differentiable functions of x , (b) R^0 is nonempty, (c) the closure of R^0 is R , and (d) a set of points A^* that are local minima corresponding to the local minimum value v^* is a nonempty isolated compact set, then there exists a set $N^* \supset A^*$ and a value $t^0 > 0$ such that, for the starting point (x^0, t^0) , where $x^0 \in N^*$, every limiting point x^* of the solution of (7.5) satisfies (2.5), (2.7), (2.8), and (2.9) (where the corresponding u^* is also generated by the differential equation). Note that u^* is not necessarily finite.

Proof. By the chain rule,

$$\frac{dP}{dt} = -|\nabla P[x(t), t]|^2 - e^{-t} \sum_{i=1}^m \frac{1}{g_i[x(t)]} < 0 \quad (7.6)$$

when $x(t) \in R^0$. Thus if $x(0) \in R^0$, the trajectory generated by (7.5) can never

leave the feasible region, since P would have infinite value there, contradicting the strictly monotonic decreasing property ensured by (7.6).

Let S be the compact perturbation set (about A^*) guaranteed by Theorem 7 (because of our Hypothesis d). By continuity and the definition of $P(x, t)$ it can be assumed that $P(x, t) \geq v^* + \lambda$ on the boundary of $S \cap R$ for some $\lambda > 0$. Let $N^* = S \cap R^0 \cap \{x \mid f(x) \leq v^* + \lambda/2\}$. The set N^* is not empty by our Assumptions a, b, and c. (Equation 7.5 is well defined by Assumption a). Let t^0 be such that $e^{-t^0} \sum_{i=1}^m 1/g_i(x^0) < \lambda/2$. Since $P[x(t), t]$ decreases strictly, $x(t)$ can never exit from the compact set $S \cap R$. Expanding P in a Taylor series,

$$P[x(2t), 2t] - P[x(t), t] = t \left\{ -|\nabla P[x(\xi), \xi]|^2 - e^{-\xi} \sum_{i=1}^m \frac{1}{g_i[x(\xi)]} \right\}, \quad (7.7)$$

where $t \leq \xi \leq 2t$.

Since the left-hand side goes to zero as $t \rightarrow \infty$, each term in brackets on the right-hand side of (7.7) must go to zero. Let

$$u_i(t) = \frac{e^{-t}}{g_i^2[x(t)]}, \quad i = 1, \dots, m. \quad (7.8)$$

Then every limit point of $[x(t), u(t)]$ satisfies the first-order necessary conditions of Section 2.1 for a local constrained minimum. Q.E.D.

In general, then, (7.5) can lead only to a stationary point (in x) of the Lagrangian function $\mathcal{L}(x, u) = f(x) - \sum u_i g_i(x)$. For convex programming problems this fact, along with complementarity, is enough to ensure convergence to a global solution.

Theorem 34 [Global Stability for the Convex Programming Problem]. If (a) $f, -g_1, \dots, g_m$ are convex continuously differentiable function of x , (b) R^0 is nonempty, and (c) the set of points that solve the convex programming problem is bounded (and hence compact), then from any starting point (x^0, t^0) , where $x^0 \in R^0$, every limit point of $[x(t), u(t)]$ [where $u(t)$ is given by (7.8)] satisfies the sufficient conditions for a minimum to the convex programming problem (6.10–6.13).

Proof. Let x^0 be any point in R^0 , and t^0 be any value (possibly negative). From our Assumptions a–c and Theorem 24, if $x(t)$ has an unbounded limit point and remains feasible, $f[x(t)] \rightarrow +\infty$. But, using the analogous steps of the proof of Theorem 33, P strictly decreases and hence $x(t)$ remains in R^0 . Thus $x(t)$ must have bounded limit points; otherwise, since $P > f$,

$$P[x(t), t] \rightarrow +\infty.$$

By the same arguments as those used in the proof of Theorem 26, every limit point of $u(t)$ is finite. Thus, since

$$\lim_{t \rightarrow \infty} \left| \nabla f[x(t)] - \sum_{i=1}^m u_i(t) \nabla g_i[x(t)] \right|^2 + \sum u_i(t) g_i[x(t)] = 0,$$

every limit point of $[x(t), u(t)]$ satisfies the sufficiency conditions (6.10–6.13) that a limit point of $x(t)$ be a minimum of the convex programming problem.

Q.E.D.

The following example, solved on an analog computer, illustrates the use of (7.5).

Example.

$$\text{minimize } \sum_{j=1}^5 e_j x_j + \sum_{i=1}^5 \sum_{j=1}^5 c_{ij} x_i x_j + \sum_{j=1}^5 d_j x_j^3$$

subject to

$$g_i = \sum_{j=1}^5 a_{ij} x_j + b_i \geq 0, \quad i = 1, \dots, 10, \quad x_j \geq 0, \quad j = 1, \dots, 5.$$

When $[c_{ij}]$ is a positive semidefinite matrix, and $d_j \geq 0, j = 1, \dots, 5$, the objective function is convex in the feasible region. Using data given in [85], page 18, this example was solved on an analog computer [13]. The theoretical solution is approximately $x^* = (0.300, 0.333, 0.400, 0.428, 0.224)$.

In Figure 10 the components of the solution of (7.5) are graphed, starting from $x^0 = (0.1, 0.2, 0.2, 0.5, 0.5)$. Equilibrium was reached at 0.8 seconds.

The accuracy of the analog solution is considerably less than that obtained by a digital computer (see [49] for the digital solution). For some problems this may be offset by the flexibility of altering the inputs once the problem setup has been accomplished.

Comparison with Classical Arrow-Hurwicz Results

In a paper in 1951 [6], Arrow and Hurwicz proposed a “differential” method for solving convex programming problems. If the problem is written as Problem C their method is essentially as follows. Find a saddle point in (x, u) of the Lagrangian function $\mathcal{L}(x, u) = f(x) - \sum u_i g_i(x)$. If there is a point (\bar{x}, \bar{u}) such that $\mathcal{L}(\bar{x}, u) \leq \mathcal{L}(\bar{x}, \bar{u}) \leq \mathcal{L}(x, \bar{u})$ for all $u \geq 0$, then \bar{x} solves the convex programming problem (C). This is easily seen by relating this to the sufficiency conditions (6.10–6.13).

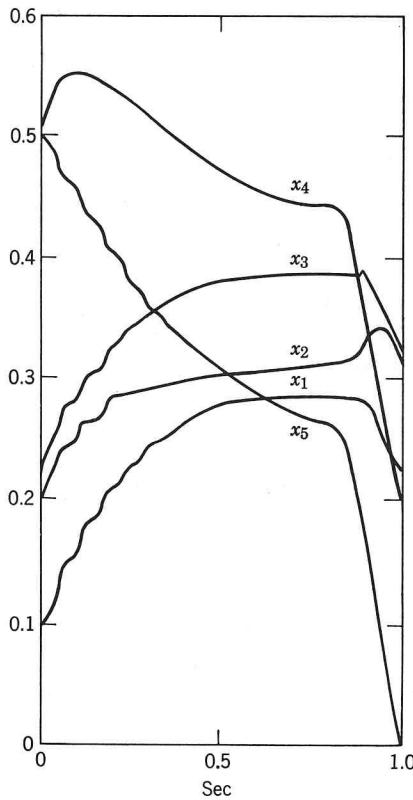


Figure 10 Analog computer solution.

The differential equations of Arrow and Hurwicz are given as

$$\dot{x} = -\nabla_x \mathcal{L}(x, u), \quad (7.9)$$

$$\dot{u}_i = 0 \quad \text{if } u_i = 0, \quad g_i(x) > 0,$$

and

$$\dot{u}_i = -g_i(x) \quad \text{otherwise, } i = 1, \dots, m. \quad (7.10)$$

Thus there is independent variation of both x and u , as opposed to the method presented in (7.5). In the latter method, as seen in (7.8), any u_i is generated explicitly in the differential equation form of the P function as the ratio $e^{-t}/g_i^2[x(t)]$.

In order to prove convergence of (7.9) and (7.10), Arrow and Hurwicz [5] showed that if the matrix of second partial derivatives of the Lagrangian is positive definite, solution of (7.9) and (7.10) generates a minimum of the convex programming problem.

8

Computational Aspects of Unconstrained Minimization Algorithms

8.1 INTRODUCTION—SUMMARY OF COMPUTATIONAL ALGORITHM

A vital test and justification of any body of theory of how to solve problems is the feasibility of computational implementation and practical application. In this chapter the computational questions implicit in the theoretical development of unconstrained minimization algorithms are discussed in detail.

The algorithm described will be applicable to any mixed interior-exterior point unconstrained algorithm as defined in Section 4.3. For definiteness we assume that the logarithmic penalty function is to be applied to the constraints where interior feasibility is required, and the quadratic loss function as the exterior point penalty term. The problem to be solved is

$$\text{minimize } f(x) \quad (\text{M})$$

subject to

$$g_i(x) \geq 0, \quad i = 1, \dots, m, \quad (8.1)$$

$$g_i(x) \geq 0, \quad i = m + 1, \dots, q. \quad (8.2)$$

(Equality constraints may appear among the last $q - m$ constraints.) The unconstrained function has the form

$$W(x, r_k) = f(x) - r_k \sum_{i=1}^m \ln g_i(x) + \sum_{i=m+1}^q \frac{\{\min [0, g_i(x)]\}^2}{r_k}. \quad (8.3)$$

Most of the computational results are general and apply to any unconstrained function. The sense in which the W function is the “best” mixed unconstrained function is pointed out in later development.

The computational algorithm is summarized as follows.

1. Find a point x^0 in $R^0 = \{x \mid g_i(x) > 0, i = 1, \dots, m\}$. If such a point is not readily available the optimization method itself can be used to find it (Section 8.5).
2. Determine r_1 , the initial value of r . For the W function we have assumed that the same r value weights all the constraints. As already discussed in Section 7.1, there are various criteria for choosing different weights for each constraint. The discussion when just one r value is used is contained in Section 8.5.
3. Determine the unconstrained minimum of W for the current value of r_k . This step constitutes the bulk of the work required by unconstrained algorithms. Section 8.2 is devoted to ways of minimizing unconstrained functions, and Section 8.3 discusses efficiencies that result when the programming problems have special structure that can be exploited. In Section 8.5 the problem of minimizing a function along a vector is considered. This is a one-dimensional subproblem of the problem of minimizing a function of several variables and can be considered separately.
4. Estimate the solution of Problem M using the extrapolation theory of Chapter 5. The formulas used to do this are contained in Section 8.4.
5. Terminate computations if the estimated solution is "acceptable." A natural criterion for convex programming problems is available via the theory of duality of Section 6.2. Discussion of convergence criteria is given in Section 8.5.
6. Select r_{k+1} (Section 8.5).
7. Estimate the next unconstrained minimum of the W function using the extrapolation formulas of Section 8.4. Continue from Step 3.

8.2 MINIMIZING AN UNCONSTRAINED FUNCTION

Background

Numerous proposals for minimizing an unconstrained function have been made, particularly since Cauchy's "steepest descent" [22] technique was introduced in 1847. Occasional reviews of the proposals have appeared [80, 103, 55], and experience on test problems is published. Unfortunately these reviews are rarely critical. The result is that variations of the old techniques are reused and the same difficulties are often perpetuated. The published experience tends to give the impression that these methods invariably work. Sometimes a failure is published [46], but it is often quickly forgotten.

In the first part of this section an attempt is made to give a brief critique of some of these methods. In the second part rational bases for developing general methods for minimizing an unconstrained function are given, and three new methods presented.

It should be emphasized that the problem of minimizing an unconstrained function is not an extraneous difficulty introduced by unconstrained minimization techniques. An unconstrained minimization problem is nothing more than a programming problem with no effective side conditions. Therefore no general method for programming can bypass the essence of the problem of unconstrained minimization. Efforts to develop efficient techniques for unconstrained minimization are thus quite pertinent to the development of mathematical programming methodology.

Without Derivatives

Consider the problem of finding the unconstrained minimum of a continuous function $f(x)$. The derivatives may not be continuous or they may not be explicitly available. This sort of problem can arise when the function values are given by a simulation or possibly a computer-stored table. In many cases, where the function may be differentiable but very involved, coding and round-off error may justify the use of a minimization method requiring only the values of the function.

Methods that do not use derivatives can be divided into at least two classes, search techniques and methods of conjugate directions. The first type is discussed in [103, 109, 80]. A review of the second type is contained in [55].

Many of the search methods lack convergence proofs, are subject to premature convergence, and require manual (as opposed to automatic) parameter adjustment. The published literature for these methods deals mainly with problems having few variables. An interesting example is a small nonconvex test problem, on which the P function [(3.10, Section 3.1] was first tested. This was successfully solved [58, pages 25–26] by a search technique.

Because of their heavy use of univariate probes, search techniques can effectively solve certain problems that give the more conventional gradient techniques appreciable difficulty. The relative efficiency of search techniques generally increases with the “separability” of the functions involved. For example, the problem minimize $[x_1^4 + 10,000x_2^4]$ can be solved in two steps by most search techniques. The isocontours of this function are very elongated ellipses, which make movement along any ray not parallel to the major axis extremely difficult.

Based on the material in [55] there is reason to believe that some newer methods based on moving in “conjugate directions” will apply very successfully to minimizing unconstrained functions of the W type. Powell’s method

[95] is guaranteed to converge to the minimum of a positive definite quadratic form in n^2 iterations. He reports success for problems in 20 variables. A modification of this has been proposed by Zangwill [116].

Most of the problems solved by the authors have had at least first partial derivatives that were analytically computable and reasonably tractable. Because of this the methods above have not been explored in any depth. Development of methods in this area is important, since it should permit a user the option of furnishing function values only, rather than requiring the calculation of first derivatives, in utilizing a nonlinear programming algorithm. An unconstrained minimization technique requiring only function values is presented in the second half of this section.

With First Derivatives

Three methods for minimizing an unconstrained function that use first partial derivatives are presented here. One of the oldest and best known is the method of "steepest descent," first proposed by Cauchy [22]. Iterations are made according to the following equation:

$$x^{i+1} = x^i - \lambda^i \nabla f(x^i), \quad (8.4)$$

where λ^i is the smallest non-negative value of λ that locally minimizes f along $-\nabla f(x^i)$ starting from x^i . It follows from (8.4) that successive gradients in the procedure are orthogonal; that is, for all i ,

$$\nabla^T f(x^{i+1}) \nabla f(x^i) = 0.$$

Curry [32] showed that any limit point \bar{x} of the sequence $\{x^i\}$ is a stationary point; that is, $\nabla f(\bar{x}) = 0$. Under stronger conditions, Goldstein [66] has proved this also.

Unfortunately, "steepest descent" has several disadvantages that often make it a poor method for minimizing an unconstrained function. These are as follows.

1. It is not generally a finite procedure for minimizing a positive definite quadratic form.
2. Each iteration is calculated independently of the others; that is, no information is stored and used that might accelerate convergence.
3. The rate of convergence depends strongly on the graph of the function. If the ratio of the maximum to minimum eigenvalues of the matrix of second partial derivatives of the function at any solution point is large, steepest descent generates short zigzagging moves in a neighborhood of such a point. Thousands of such moves may be required to give a satisfactory approximation of a limit point [49, page 41].

Various attempts to accelerate convergence in these cases have been made, often without rigorous basis. All gradient methods for solving nonlinear programming problems must eventually deal with this convergence problem.

The rationale for the next two methods is the minimization in n or fewer steps of a positive definite quadratic form. These methods are based on generating vectors in "conjugate directions."

Suppose that f is of the form

$$f(x) = b^T x + \frac{x^T G x}{2}, \quad (8.5)$$

where G is a positive definite matrix. Let s^0, \dots, s^{n-1} be nonzero vectors and the λ^i be scalars with the property that

$$f(x^{k+1}) = f\left[x^0 + \sum_{i=0}^{k-1} \lambda^i s^i + \lambda^k s^k\right]$$

is the minimum of f along s^k , starting from the point $x^k = x^0 + \sum_{i=0}^{k-1} \lambda^i s^i$, for $k = 0, \dots, n-1$. The s^i are called conjugate, with respect to the matrix G , if

$$(s^i)^T G s^j = 0, \quad i \neq j. \quad (8.6)$$

It follows that x^n is the unconstrained minimum of f in the entire space. To prove this, note first that the positive definiteness of G , the fact that the s^i are nonzero, and (8.6) imply that the s^i are independent. It follows that

$$\begin{aligned} \nabla f(x^n) &= \nabla f\left[x^k + \sum_{i=k}^{n-1} \lambda^i s^i\right], \quad k = 0, \dots, n-1, \\ &= b + G[x^k + \lambda^k s^k] + \sum_{i=k+1}^{n-1} \lambda^i G s^i, \quad k = 0, \dots, n-1, \\ &= \nabla f[x^{k+1}] + \sum_{i=k+1}^{n-1} \lambda^i G s^i, \quad k = 0, \dots, n-1. \end{aligned}$$

Multiplying both sides by $(s^k)^T$ yields

$$(s^k)^T \nabla f(x^n) = (s^k)^T \nabla f(x^{k+1}) + (s^k)^T \sum_{i=k+1}^{n-1} \lambda^i G s^i = 0$$

by the orthogonality of consecutive moves generated by choice of the λ^i and conjugacy of the s^i , for $k = 0, \dots, n-1$. Since the s^k are independent, it follows that $\nabla f(x^n) = 0$, a sufficient condition that x^n be the minimum.

The methods to be presented are ways of generating successive conjugate directions.

Fletcher and Reeves [57] proposed the following. Let x^0 be the starting point, let $s^0 = -\nabla f(x^0)$, let x^{i+1} be a point yielding the minimum value of $f(x)$ on the line from x^i in direction s^i , and let

$$\beta^i = |\nabla f(x^{i+1})|^2 / |\nabla f(x^i)|^2.$$

Then the new direction s^{i+1} is given by

$$s^{i+1} = -\nabla f(x^{i+1}) + \beta^i s^i.$$

The directions s^i , $i = 0, \dots, n-1$ are conjugate if $f(x)$ is of the form (8.5).

Thus for quadratic functions of the above form the algorithm will locate the minimum in n steps regardless of the starting point. For general functions Fletcher and Reeves suggest restarting the algorithm every $(n+1)$ steps.

The third method, called the “variable metric” method, is a modification by Fletcher and Powell [56] of a technique proposed by W. C. Davidon [35]. In addition to n -step convergence for a positive definite quadratic, the inverse of the matrix of second partial derivatives at the optimum is generated. (This means storage of the order of $n^2/2$ locations.) The algorithm is as follows. Let x^0 be the initial point and H^0 an arbitrary positive definite matrix. Let

$$s^i = -H^i \nabla f(x^i),$$

$$\sigma^i = \lambda^i s^i,$$

$$x^{i+1} = x^i + \sigma^i,$$

where λ^i is chosen to minimize f along s^i , starting at x^i . If H^i is positive definite, λ^i must be greater than zero unless x^i is a minimum of $f(x)$. Let

$$y^i = \nabla f(x^{i+1}) - \nabla f(x^i).$$

The new approximation of the inverse of the matrix of second partial derivatives of f is given by

$$H^{i+1} = H^i + \frac{(\sigma^i)(\sigma^i)^T}{(\sigma^i)^T y^i} - \frac{H^i(y^i)(y^i)^T H^i}{(y^i)^T H^i y^i}.$$

It is shown in [56] that H^{i+1} is positive definite if H^i is positive definite, ensuring that f is decreased on each move. The vectors $\sigma^0, \dots, \sigma^{n-1}$ are conjugate with respect to the matrix G if f is given by (8.5). A method similar to this, containing several additional properties, is given in the second part of this section.

Using Second Partial Derivatives

The most successful method used to date by the authors for minimizing an unconstrained convex function is the generalized Newton method, which

requires second partial derivatives. For notational convenience let $F(x^i)$ represent $\nabla^2 f(x^i)$. It is assumed that $F(x^i)$ is nonsingular. The iterations are given by

$$x^{i+1} = x^i - \lambda^i F(x^i)^{-1} \nabla f(x^i), \quad (8.7)$$

where $\lambda^i > 0$ is chosen to minimize f along $-F(x^i)^{-1} \nabla f(x^i)$, starting from x^i . With $\lambda^i \equiv 1$, this is called Newton's method. The motivation for this is seen by first expanding the function in a two-term Taylor series and solving for the vector, which yields the minimum of the quadratic approximation. A fuller discussion may be found in [31].

If f is a positive definite quadratic form (8.7) with $\lambda^i = 1$ will yield the minimum in one step. When f is a convex function (not necessarily quadratic) (8.7) will guarantee a decrease in f with every iteration if x^i is not a minimum.

The disadvantages of the generalized Newton, or "second-order," method are the following.

1. An inverse of $F(x^i)$ may not always exist.
2. In a nonconvex problem the function may not decrease when the current point x^i is not near the minimum.
3. For some problems with continuous second partial derivatives it may be tedious and troublesome to compute the analytic forms accurately.
4. Storage of the order of $n^2/2$ locations is required.

Nonetheless, for problems in which second partial derivatives are readily available, the method has proved far superior to any other tried. In Section 8.3 the computation of (8.7) for the logarithmic quadratic penalty function in many special cases of mathematical programming is shown to be of much less than the expected order of $n^3/3$ operations. This follows from the special form of the matrix of second partial derivatives of this function.

A revised Newton method overcoming the first two of the above criticisms is discussed in the second half of this section.

Comparison of the Methods

To do justice to these methods would require much more space, analysis, and experimentation. We shall give a small example that was solved using each of the four methods just discussed to illustrate their convergence properties.

Example. The function to be minimized is

$$P(x, r) = f(x) + r \sum_{i=1}^{10} \frac{1}{g_i} + r \sum_{j=1}^5 \frac{1}{x_j}, \quad r = 1, x \in R^0$$

where f and $\{g_i\}$ are the functions given in Section 7.3. In the region where $R^0 = \{x \mid g_i(x) > 0, i = 1, \dots, 10, x_j > 0 \text{ all } j\}$, $P(x, 1)$ is a *strictly convex*

function. Tables 6, 7, 8, and 9 summarize the “work” required to minimize $P(x, 1)$ by each of the four techniques just described.

Table 6 Gradient Descent on Example

Iteration	x_1	x_2	x_3	x_4	x_5	$ \nabla_x P $	$P(x, 1)$
0	10^{-4}	10^{-4}	10^{-4}	10^{-4}	1.1		
1	0.0798	0.0798	0.0798	0.0798	1.1000	6×10^2	88.919014
10	0.1194	0.0694	0.1932	0.2001	1.0040	1×10^2	44.533934
20	0.1732	0.0982	0.2772	0.3285	0.8691	8×10^1	21.944428
30	0.1835	0.1344	0.2887	0.4086	0.7786	5×10^1	14.884308
50	0.1814	0.1726	0.2898	0.4913	0.6670	3×10^1	10.084040
100	0.1776	0.2260	0.2889	0.5537	0.5157	1×10^1	7.329493
125	0.1762	0.2395	0.2898	0.5605	0.4775	8×10^0	7.058207
151	0.1763	0.2477	0.2894	0.5642	0.4543	4×10^0	6.960957

Table 7 Fletcher-Reeves Method on Example

Iteration	x_1	x_2	x_3	x_4	x_5	$ \nabla_x P $	$P(x, 1)$
0	10^{-4}	10^{-4}	10^{-4}	10^{-4}	1.1		
5	0.1921	0.0865	0.2794	0.3517	0.9154	2×10^2	24.626489
10	0.1747	0.2321	0.2921	0.6065	0.5376	4×10^1	7.812791
15	0.1781	0.2435	0.2883	0.5460	0.4629	1×10^1	7.036639
20	0.1750	0.2555	0.2879	0.5735	0.4350	2×10^0	6.926209
30	0.1764	0.2573	0.2889	0.5700	0.4278	3×10^{-1}	6.919394
40	0.1764	0.2575	0.2890	0.5699	0.4272	4×10^{-2}	6.919354

Table 8 Variable Metric Method on Example

Iteration	x_1	x_2	x_3	x_4	x_5	$ \nabla_x P $	$P(x, 1)$
0	10^{-4}	10^{-4}	10^{-4}	10^{-4}	1.1		
5	0.0651	0.0833	0.0933	0.0775	0.9584	4×10^2	75.359926
10	0.0646	0.0643	0.0998	0.0907	0.9128	4×10^2	73.072791
15	0.1601	0.0937	0.2521	0.2162	0.8253	1×10^2	27.639421
20	0.1857	0.2229	0.2864	0.5966	0.5013	7×10^1	7.772631
25	0.1736	0.2578	0.2862	0.5713	0.4342	5×10^0	6.933100
28	0.1764	0.2575	0.2889	0.5699	0.4271	8×10^{-3}	6.919352

The method of steepest descent failed to converge after 151 iterations when the iterations were manually terminated. The method of Fletcher and Reeves took 40 iterations to converge, the variable metric method, 28; and the “second order” method, 8. The effort per iteration is approximately the same

Table 9 Generalized Newton or "Second-Order" Method on Example

Iteration	x_1	x_2	x_3	x_4	x_5	$ \nabla_x P $	$P(x, 1)$
0	10^{-4}	10^{-4}	10^{-4}	10^{-4}	1.1		
1	8.2×10^{-4}	8.2×10^{-4}	8.2×10^{-4}	8.2×10^{-4}	1.4204	3×10^6	5.00×10^3
2	0.0501	0.0501	0.0501	0.0501	0.9296	1×10^3	114.854
3	0.1749	0.1962	0.2596	0.2381	0.6438	4×10^2	23.263122
4	0.1979	0.2332	0.3165	0.4213	0.4637	2×10^2	11.29304
5	0.1786	0.2885	0.2957	0.5896	0.3888	7×10^1	7.512202
6	0.1766	0.2577	0.2882	0.5632	0.4380	9×10^0	6.947183
7	0.1764	0.2380	0.2890	0.5702	0.4263	5×10^{-1}	6.919448
8	0.1764	0.2375	0.2889	0.5699	0.4271	7×10^{-3}	6.919353

for each method, so that the required computer time is in about the same ratio as the number of iterations.

The tremendous power of the second order method for problems of this kind (as exemplified by this example), that is, for convex programming problems, has been seen by the authors in many computer solved examples. In this example the $n^3/3$ operations (multiplications and additions) required to compute the direction vector for each iteration of the second-order method did not significantly increase the time required to solve the problem, since n was so small. For "large" problems this cubic law can become prohibitive. Some techniques for reducing the computational effort are discussed in Section 8.3.

Even though the Fletcher-Reeves and variable metric methods require on the order of n and $n^2/2$ operations, respectively, to compute the direction vectors, both require approximately n times as many moves to converge. (This is to be expected, since the second-order method minimizes a positive definite form in one step requiring $n^3/3$ operations, and the others require n iterations of n and $n^2/2$ operations, respectively.)

It is worth examining the Fletcher-Reeves method a little more closely, since the above remarks seem to imply that the number of operations required to minimize a positive definite quadratic form varies as n^2 rather than n^3 . However, the fact is that the amount of work required to find the minimum of a quadratic along a vector is of the order of n^2 operations.

The area of investigation on ways to minimize an unconstrained function has only been touched on. Much good work has been done in recent years in this field. Modifications of the last three methods discussed will probably accelerate the process for problems with special characteristics or structure. For the second-order method, Section 8.3 contains an attempt to utilize the special structure of the W function to simplify the second-order computations.

In the last part of this section we present three methods for minimizing unconstrained functions, one for each order of differentiability assumed.

Three New Methods for Minimizing an Unconstrained Function Using a Second-Order Point of View

In this part of the section we discuss three new methods of minimizing an unconstrained function. We discuss, first, a modified Newton method that uses second partial derivatives; second, a modified variable metric method that uses first partials and can take advantage of second partial information if available; and, third, a method that employs only function values.

As already mentioned, two traditional theoretical arguments against the generalized Newton method are, first, if $F(x^i)$ is not a positive definite

separable. The complete procedure for a mixed separable-factorable problem proceeds as follows.

Include in the matrix D_1 all matrices in the expansion $\nabla^2 W(x, r)$ that are diagonal. This matrix is easily inverted. Then iterate on the matrices $\sum_{i=1}^m \nabla g_i(x) [r/g_i^2(x)] \nabla^T g_i(x)$ and the proportional matrices resulting from the "factorable" constraints using the scheme of (8.52) and (8.53).

Most real world convex problems seen by the authors are factorable. Several well-known problems in the published literature illustrate this.

One of the best known "quadratic" programming problems is Markowitz's portfolio selection problem [83]. The objective function to be minimized is the variance of a probability distribution and is usually given as $f(x) = x^T C x$, where C is a positive semidefinite variance-covariance matrix. But C is a matrix computed from empirical observations. Let M be a $n \times l$ matrix of observations, where l is the number of time periods. Then m_{jk} is the value of the j th variable at the k th time period. The variance-covariance matrix C is given by

$$C = MM^T - M(e_l) \left(\frac{1}{l} \right) e_l^T M^T, \quad (8.56)$$

where e_l is an $l \times 1$ vector of ones. Since $\nabla^2 f(x) = 2C$, $f(x)$ is a factorable function with $l + 1$ factors. From the point of view of the above discussion, using unconstrained minimization algorithms it is reasonable to work with the matrix of original data. (This avoids, incidentally, the $nl^2/2$ computations required to calculate C .)

Another standard objective function is of the form

$$\text{minimize } f(x) = \sum_{j=1}^q w_j \exp \left\{ - \sum_{i=1}^p \alpha_{ij} x_{ij} \right\}. \quad (8.57)$$

The $\{x_{ij}\}$ are double subscripted variables ($p \cdot q$ in number), the w_j are strictly positive constants, and the α_{ij} are non-negative constants for all i, j . (See [14, Chapter 2] for an example.) The function (8.57) is convex and factorable. The exact form of the second partial derivative matrix is the sum of q factors, each factor having only p nonzero elements. If the variables are ordered as $x = (x_{11}, \dots, x_{p1}, \dots, x_{1q}, \dots, x_{pq})$, then

$$\nabla^2 f(x) = \sum_{j=1}^q w_j \alpha_j^T \exp \left\{ - \sum_{i=1}^p \alpha_{ij} x_{ij} \right\} \alpha_j,$$

where $\alpha_j = \{0, \dots, 0, \alpha_{1j}, \dots, \alpha_{pj}, 0, \dots, 0\}$.

Other examples of factorable convex nonlinear functions are to be found in [107] and [24].

Example [The Chemical Equilibrium Problem]. To illustrate how the development discussed in the preceding sections can reduce the computational effort required to solve nonlinear programming problems, a specific example—the chemical equilibrium problem—was solved. The form of the problem is as follows.

$$\text{minimize } f(x) = \sum_{k=1}^q \left\{ \sum_{j=1}^{n_k} x_{jk} \left[c_{jk} + \ln \left(\frac{x_{jk}}{\sum_{i=1}^{n_k} x_{ik}} \right) \right] \right\}$$

subject to

$$Hx = b,$$

$$x \geq 0.$$

[The vector $x = (x_{1,1}, \dots, x_{n_1,1}, \dots, x_{1,q}, \dots, x_{n_q,q})$.] For a description of the physical meaning of the problem the reader is referred to [109].

The matrix of second partial derivatives of f has the form

$$\nabla^2 f = \text{diag} \left(\frac{1}{x_j} \right) + \sum_{k=1}^q I_k \left(\frac{-1}{\sum_{i=1}^{n_k} x_{ik}} \right) I_k^T,$$

where I^k is an $n \times 1$ vector with ones in the positions associated with $(x_{1,k}, \dots, x_{n_k,k})$ as it appears in the vector x , and zeros elsewhere. The mixed interior-exterior point unconstrained function used to solve this was $P = f + r \sum 1/x_i + \sum (h_i^T x - b_i)^2/r$, whose second partial matrix is

$$\nabla^2 P = \text{diag} \left(\frac{1}{x_j} + \frac{2r}{x_j^3} \right) + \sum_{i=1}^p h_i \left[\frac{2(h_i^T x - b_i)}{r} \right] h_i^T + \sum_{k=1}^q I_k \left(\frac{-1}{\sum_{i=1}^{n_k} x_{ik}} \right) I_k^T \quad (8.58)$$

where h_i^T is the i th row of H . The matrix $(\nabla^2 P)^{-1}$ was obtained by the method just described. Iteration using formula (8.53) was done $p + q$ times, the order as indicated in (8.58). Since consecutive vectors resulting from (8.53) are sparse, they were stored in packed form.

The data were taken from [36, pages 174 and 175]. For this problem $n = 45$ (11 of the 56 variables in [36] are kept at zero and thus do not enter in the computations), $p = 16$, and $q = 7$. The matrix H has 88 nonzero elements.

Using the regular sequential unconstrained minimization technique (SUMT) computer code [85] each computation of $(\nabla^2 P)^{-1} \nabla P$ took 7.2 sec. Using the modification above, each computation took only 3.5 sec. An additional benefit was increased accuracy, which resulted in fewer moves required to solve the problem. Using the packing techniques, only 603

memory locations were required to store $(\nabla^2 P)^{-1}$, compared to 2025 for the regular way.

Total time to solve this problem was 18 min. on the IBM 7040 using the revised method. The usual method—using the Newton second-order method in a straightforward way—was not run to completion, but based on its progress it would have taken about 90 min.

8.4 ACCELERATION BY EXTRAPOLATION

A very powerful computational tool is available when the conditions guaranteeing the existence of $D^i[x(0)]$ hold as developed in Chapter 5. In the following discussion these appropriate conditions are assumed to hold.

Suppose the W function has been uniquely minimized for $r_1 > \dots > r_p > 0$ at x^1, \dots, x^p . A polynomial in r that yields x^1, \dots, x^p is given by a set of equations of the form

$$x^k = \sum_{j=0}^{p-1} a_j(r_k)^j, \quad k = 1, \dots, p, \quad (8.59)$$

where the a_j are n -component vectors.

The determinant of the matrix

$$R = \begin{bmatrix} r_1^0 & \dots & r_p^0 \\ \vdots & & \vdots \\ \vdots & & \vdots \\ r_1^{p-1} & \dots & r_p^{p-1} \end{bmatrix}$$

(called the Vandermonde determinant [105, page 120]) is equal to

$$\prod_{i < j} (r_j - r_i)$$

and, since $r_j \neq r_i$ ($i \neq j$), R is nonsingular. Thus the vectors a_j are uniquely determined by (8.59). Then $\sum_{j=0}^{p-1} a_j(r)^j$ is an approximation of $x(r)$ in the interval $[0, r_1]$, and $x(0) = x^*$ (a solution) is approximated by a_0 . That this approximation converges to a solution, and in fact that the estimate improves with each minimum that is determined, is seen as follows.

The exact Taylor series expansion of x^k in r_k about $x(0)$ is

$$x^k = \sum_{j=0}^{p-1} (r_k)^j \frac{D^j x(0)}{j!} + \epsilon^k, \quad k = 1, \dots, p, \quad (8.60)$$

where

$$\epsilon^k = \left(\frac{r_k^p}{k!} \right) \left[\frac{d^p x_1(\eta_{1k})}{dr^p}, \dots, \frac{d^p x_n(\eta_{nk})}{dr^p} \right], \quad 0 \leq \eta_{jk} \leq r_k, j = 1, \dots, n.$$

accordingly. Finally, when S is empty, an interior point is attained. If at a solution of the auxiliary problem S is not empty, the problem is considered infeasible.

If the constraints are concave, the auxiliary problem is a convex programming problem. Because of this, an easy check on the nonfeasibility of the problem is available at any unconstrained minimum. Using the inequalities resulting from the dual theory (Section 6.2), it follows that if $G^k > 0$ at the k th minimum the problem is not feasible. Even though the current auxiliary problem may be far from being solved, further computations are not needed to determine nonfeasibility. This is a distinct advantage of the dual aspects of this approach.

REFERENCES

1. Abadie, J., J. Carpentier, and C. Hensgen, "Generalization of the Wolfe Reduced Gradient Method to the Case of Nonlinear Constraints," paper presented at the Joint European Meeting of the Econometric Society/The Institute of Management Science, Warsaw, September 1966.
2. Ablow, C. M., and G. Brigham, "An Analog Solution of Programming Problems," *Operations Res.*, 3:388-394 (1955).
3. Apostol, T. M., *Mathematical Analysis*, Addison-Wesley, Reading, Mass., 1957.
4. Arrow, K. J., "A Gradient Method for Approximating Saddle Points and Constrained Maxima," RAND Corp., Santa Monica, Calif., Paper P-223, 15 pp., 1951.
5. Arrow, K., and L. Hurwicz, "Reduction of Constrained Maxima to Saddle Point Problems," in J. Neyman (Ed.), *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, 1956, pp. 1-20.
6. Arrow, K. J., L. Hurwicz, and H. Uzawa, *Studies in Linear and Nonlinear Programming*, Stanford University Press, Palo Alto, Calif., 1958.
7. Beale, E. M. L., "On Minimizing a Convex Function Subject to Linear Inequalities," *J. Roy. Statist. Soc., Ser. B*, 17(2):173-184 (1955).
8. Beale, E. M. L., "On Quadratic Programming," *Naval Res. Logist. Quart.*, 6:227-243 (September 1959).
9. Beltrami, E. J., and R. McGill, "A Class of Variational Problems in Search Theory and the Maximum Principle," *Operations Res.*, 14(2):267-278 (1966).
10. Berge, C., and A. Ghouila-Houri, *Programmes, Games and Transportation Networks*, Wiley, New York, 1965.
11. Berkovitz, L. D., "Variational Methods in Problems of Control and Programming," *J. Math. Anal. Appl.*, 3:145-169 (1961).
12. Bochner, S., and W. T. Martin, *Several Complex Variables*, Princeton University Press, Princeton, N.J., 1948.
13. Bombart, J., and F. C. Hipp, "Analogue Computer Solution of Programming Problems," Memorandum, Research Analysis Corp., McLean, Virginia.
14. Bracken, J., and G. P. McCormick, *Selected Applications of Nonlinear Programming*, Wiley, New York, 1968.
15. Broyden, C. G., "A Class of Methods for Solving Nonlinear Simultaneous Equations," *Math. Computation*, 19(92):577-593 (1965).

16. Bui Trong Lieu and P. Huard, "La Méthode des Centres dans un Espace Topologique," *Numér. Math.*, 3(1):56–67 (1966).
17. Butler, T., and A. V. Martin, "On a Method of Courant for Minimizing Functionals," *J. Math. Phys.*, 41:291–299 (1962).
18. Camp, G. D., "Inequality-Constrained Stationary Value Problems," *J. Operations Res. Soc. Am.*, 3:548–550 (1955).
19. Carathéodory, C., *Variationsrechnung und Partielle Differentialgleichungen erste Ordnung*, Teubner, Leipzig, 1935.
20. Carroll, C. W., *An Operations Research Approach to the Economic Optimization of a Kraft Pulping Process*, Ph.D. Dissertation, Institute of Paper Chemistry, Appleton, Wis., 1959.
21. Carroll, C. W., "The Created Response Surface Technique for Optimizing Nonlinear Restrained Systems," *Operations Res.*, 9(2):169–184 (1961).
22. Cauchy, A., "Méthode Générale pour la Resolution des Systèmes D'équations Simultanées," *Compt. Rend.*, 25:536–538 (1847).
23. Charnes, A., and W. W. Cooper, Letter to the Editor, "Such Solutions are Very Little Solved," *Operations Res.*, 3(3):345–346 (1955).
24. Charnes, A., and W. W. Cooper, "Chance-Constrained Programming," *Management Sci.*, 6:73–79 (1959).
25. Charnes, A., and W. W. Cooper, *Management Models and Industrial Applications of Linear Programming*, Wiley, New York, 1961.
26. Charnes, A., W. W. Cooper, and A. Henderson, *An Introduction to Linear Programming*, Wiley, New York, 1953.
27. Charnes, A., and C. E. Lemke, "Minimization of Non-linear Separable Convex Functionals," *Naval Res. Logist. Quart.*, 1:301–302 (1954).
28. Cheney, E. W., and A. A. Goldstein, "Newton's Method for Convex Programming and Tchebycheff Approximation," *Numer. Math.*, 1:253–268 (1959).
29. Courant, R., "Variational Methods for the Solution of Problems of Equilibrium and Vibrations," *Bull. Am. Math. Soc.*, 49:1–23 (1943).
30. Courant, R., "Calculus of Variations and Supplementary Notes and Exercises" (Mimeo graphed Notes), Supplementary Notes by M. Kruskal and H. Rubin, revised and amended by J. Moser, New York University, 1956–1957.
31. Crockett, J. B., and H. Chernoff, "Gradient Methods of Maximization," *Pacific J. Math.*, 5:33–50 (1955).
32. Curry, H. B., "The Method of Steepest Descent for Non-linear Minimization Problems," *Quart. Appl. Math.*, 2(3):250–261 (1944).
33. Dantzig, G. B., "Maximization of a Linear Function of Variables Subject to Linear Inequalities," in T. C. Koopmans (Ed.), *Activity Analysis of Production and Allocation*, Wiley, New York, 1951, Chapter 21.
34. Dantzig, G., *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.
35. Davidon, W. C., "Variable Metric Method for Minimization," Research and Development Report ANL-5990 (Revised), Argonne National Laboratory, U.S. Atomic Energy Commission, 1959.
36. DeHaven, J. C., and E. C. Deland, "Reactions of Hemoglobin and Steady States in the Human Respiratory System: An Investigation Using Mathematical Models and an Electronic Computer," RAND Corp., Santa Monica, Calif., Memorandum RM-3212-PR, December 1962, pp. 174–175.
37. Dorn, W. S., "On Lagrange Multipliers and Inequalities," *J. Operations Res. Soc. Am.*, 9:95–104 (1961).

198 Computational Aspects of Unconstrained Minimization Algorithms

38. Duffin, R. J., "Cost Minimization Problems Treated by Geometric Means," *Operations Res.*, **10**:668–675 (1962).
39. Duffin, R. J., E. L. Peterson, and C. Zener, *Geometric Programming—Theory and Application*, Wiley, New York, 1967.
40. Everett, H., "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," *Operations Res.*, **11**:399–417 (1963).
41. Falk, J. E., "A Constrained Lagrangian Approach to Nonlinear Programming," Ph.D. Dissertation, University of Michigan, Ann Arbor, June 1965.
42. Falk, J. E., "A Relaxed Interior Approach to Nonlinear Programming," Technical Paper RAC-TP-279, Research Analysis Corporation, McLean, Va., 1967.
43. Falk, J. E., "Lagrange Multipliers and Nonlinear Programming," *J. Math. Anal. Appl.*, **19**(1) (July 1967).
44. Farkas, J., "Über die Theorie der einfachen Ungleichungen," *J. Reine Angew. Math.*, **124**:1–27 (1901).
45. Faure, P., and P. Huard, "Résolution des Programmes Mathématiques à Fonction Non-linéaire par la Méthode du Gradient Reduit," *Rev. Franc. Recherche Opérationnelle*, **9**:167–205 (1965).
46. Feder, D. P., "Automatic Lens Design Methods," *J. Opt. Soc. Am.*, **47**:902–912 (1957).
47. Fiacco, A. V., "Comments on the Paper of C. W. Carroll," *Operations Res.*, **9**:184–185 (1961).
48. Fiacco, A. V., "Sequential Unconstrained Minimization Methods for Nonlinear Programming," Ph.D. Dissertation, Northwestern University, Evanston, Ill., June 1967.
49. Fiacco, A. V., and G. P. McCormick, "Programming under Nonlinear Constraints by Unconstrained Minimization: A Primal-Dual Method," Technical Paper RAC-TP-96, Research Analysis Corporation, McLean, Va., 1963.
50. Fiacco, A. V., and G. P. McCormick, "The Sequential Unconstrained Minimization Technique for Nonlinear Programming: A Primal-Dual Method," *Management Sci.*, **10**(2):360–366 (1964).
51. Fiacco, A. V., and G. P. McCormick, "Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming," *Management Sci.*, **10**(2):601–617 (1964).
52. Fiacco, A. V., and G. P. McCormick, "SUMT without Parameters", System Research Memorandum No. 121, Technical Institute, Northwestern University, Evanston, Ill., 1965.
53. Fiacco, A. V., and G. P. McCormick, "Extensions of SUMT for Nonlinear Programming: Equality Constraints and Extrapolation," *Management Sci.*, **12**(11):816–829 (1966).
54. Fiacco, A. V., and G. P. McCormick, "The Slacked Unconstrained Minimization Technique for Convex Programming," *SIAM J. Appl. Math.*, **15**(3):505–515 (1967).
55. Fletcher, R., "Function Minimization without Evaluating Derivatives—A Review," *Computer J.*, **8**:33–41 (1965).
56. Fletcher, R., and M. J. D. Powell, "A Rapidly Convergent Descent Method for Minimization," *Computer J.*, **6**:163–168 (1963).
57. Fletcher, R., and C. M. Reeves, "Function Minimization by Conjugate Gradients," *Computer J.*, **7**:149:154 (1964).
58. Flood, M. M., and A. Leon, "A Generalized Search Code for Optimization," Preprint 129, Mental Health Research Institute, University of Michigan, Ann Arbor, 1964.

59. Frisch, K. R., "Principles of Linear Programming—With Particular Reference to the Double Gradient Form of the Logarithmic Potential Method," Memorandum of October 18, 1954, University Institute of Economics, Oslo.
60. Frisch, K. R., "The Logarithmic Potential Method of Convex Programming," Memorandum of May 13, 1955, University Institute of Economics, Oslo.
61. Gale, D., "Convex Polyhedral Cones and Linear Inequalities," in T. C. Koopmans (Ed.), *Activity Analysis of Production and Allocation*, Wiley, New York, 1951.
62. Gass, S. I., *Linear Programming: Methods and Applications*, McGraw-Hill, New York, 1958.
63. Gelfand, I. M., and S. V. Fomin, *Calculus of Variations*, Revised English Edition, Translated and Edited by R. A. Silverman, Prentice-Hall, Englewood Cliffs, N.J., 1963, Chapter 2.
64. Geoffrion, A. M., "Strictly Concave Parametric Programming, Part I: Basic Theory," *Management Sci.*, 13:244–253 (1966).
65. Geoffrion, A. M., "Strictly Concave Parametric Programming, Part II: Additional Theory and Computational Considerations," *Management Sci.*, 13(5):359–370 (1967).
66. Goldstein, A. A., "Cauchy's Method of Minimization," *Numer. Math.*, 4:146–150 (1962).
67. Goldstein, A. A., and B. R. Kripke, "Mathematical Programming by Minimizing Differentiable Functions," *Numer. Math.*, 6:47–48 (1964).
68. Hadley, G., *Nonlinear and Dynamic Programming*, Addison-Wesley, Reading, Mass., 1964.
69. Hancock, H., *Theory of Maxima and Minima*, Dover, New York, 1960.
70. Hartley, H. O., and R. R. Hocking, "Convex Programming by Tangential Approximation," *Management Sci.*, 9:600–612 (1963).
71. Hestenes, M. R., *Calculus of Variations and Optimal Control Theory*, Wiley, New York, 1966.
72. Householder, A. S., *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964.
73. Huard, P., "Résolution des P. M. à Contraintes Non-linéaires par la Méthode des Centres," Note E.D.F. HR 5.690, May 6, 1964.
74. Kelley, J. E., Jr. "The Cutting Plane Method for Solving Convex Programs," *J. Soc. Ind. Appl. Math.*, 8(4):703–712 (1960).
75. Kiefer, J., "Optimum Sequential Search and Approximation Methods under Minimum Regularity Assumptions," *SIAM J.*, 5:105–136 (1957).
76. Klein, B., "Direct Use of Extremal Principles in Solving Certain Optimizing Problems Involving Inequalities," *J. Operations Res. Soc. Am.*, 3:548 (1955).
77. Kuhn, H. W., and A. W. Tucker, "Non-linear Programming," in J. Neyman (Ed.), *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, 1951, pp. 481–493.
78. Leitmann, G. (Ed.), *Optimization Techniques: With Applications to Aerospace Systems*, Academic Press, New York, 1962.
79. Lemke, C. E., "A Method for Solution of Quadratic Programming Problems," *Management Sci.*, 8:442–453 (1962).
80. Leon, A., "A Comparison among Eight Known Optimizing Procedures," in A. Lavi and T. P. Vogl (Eds.), *Proceedings of Symposium on Recent Advances in Optimization Techniques*, Wiley, New York, 1965.
81. Leon, A., "An Annotated Bibliography on Optimization," Preprint 162, Mental Health Research Institute, University of Michigan, Ann Arbor, 1965.
82. Mangasarian, O. L., and S. Fromowitz, "The Fritz John Necessary Optimality

200 Computational Aspects of Unconstrained Minimization Algorithms

- Conditions in the Presence of Equality and Inequality Constraints," *J. Math. Anal. Appl.*, 17(1):37-47 (1967).
- 83. Markowitz, H., *Portfolio Selection*, Wiley, New York, 1959.
 - 84. McCormick, G. P., "Second Order Conditions for Constrained Minima," *SIAM J. Appl. Math.*, 15(3):641-652 (1967).
 - 85. McCormick, G. P., W. C. Mylander, III, and A. V. Fiacco, "Computer Program Implementing the Sequential Unconstrained Minimization Technique for Nonlinear Programming," Technical Paper RAC-TP-151, Research Analysis, McLean, Va., 1965.
 - 86. McCormick, G. P., and W. I. Zangwill, "A Technique for Calculating Second Order Optima," Technical Paper, Research Analysis Corp., McLean, Virginia.
 - 87. McGill, R., and P. Kenneth, "Solution of Variational Problems by Means of a Generalized Newton-Raphson Operator," *AIAA J.*, 2(10):1761-1766 (1964).
 - 88. Miller, C. E., "The Simplex Method for Local Separable Programming," in R. L. Graves and P. Wolfe (Eds.), *Recent Advances in Mathematical Programming*, McGraw-Hill, New York, 1963, pp. 89-100.
 - 89. Motzkin, T. S., "New Techniques for Linear Inequalities and Optimization," in *Project SCOOP, Symposium on Linear Inequalities and Programming*, Planning Research Division, Director of Management Analysis Service, U.S. Air Force, Washington, D.C., No. 10, 1952.
 - 90. Murray, W., "Ill-Conditioning in Barrier and Penalty Functions Arising in Constrained Non-linear Programming," paper presented at the Princeton Mathematical Programming Symposium, August 14-18, 1967.
 - 91. Parisot, G. R., "Résolution Numérique Approchée du Problème de Programmation Linéaire par Application de la Programmation Logarithmique," Ph.D. Dissertation, University of Lille, France, 1961.
 - 92. Pietrzykowski, T., "Application of the Steepest Descent Method to Concave Programming," in *Proceedings of the IFIPS Congress*, Munich, 1962, North-Holland Publishing Company, Amsterdam, 1962, pp. 185-89.
 - 93. Pomentale, T., "A New Method for Solving Conditioned Maxima Problems," *J. Math. Anal. Appl.*, 10:216-220 (1965).
 - 94. Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*, Translated by K. N. Trinogoff, Interscience, New York, 1962.
 - 95. Powell, M. J. D., "An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives," *Computer J.*, 7(2):155-162 (1964).
 - 96. Riley, V., and S. I. Gass, *Linear Programming and Associated Techniques*, Johns Hopkins Press, Baltimore, 1958.
 - 97. Ritter, K., "Stationary Points of Quadratic Maximum Problems," *Z. Wahrscheinlichkeitstheorie verwandte Gebiete*, 4:149-158 (1965).
 - 98. Rosen, J. B., "The Gradient Projection Method for Nonlinear Programming, Part I: Linear Constraints," *J. Soc. Ind. Appl. Math.*, 8(1):181-217 (1960).
 - 99. Rosen, J. B., "The Gradient Projection Method for Nonlinear Programming Part II: Nonlinear Constraints," *J. Soc. Ind. Appl. Math.*, 9:414-443 (1961).
 - 100. Rosenbrock, H. H., "Automatic Method for Finding the Greatest or Least Value of a Function," *Computer J.*, 3:175-184 (1960).
 - 101. Rubin, H., and P. Ungar, "Motion under a Strong Constraining Force," *Commun. Pure Appl. Math.*, 10:65-87 (1957).
 - 102. Rudin, W., *Principles of Mathematical Analysis*, McGraw-Hill, New York, 1953.
 - 103. Spang, H. A., "A Review of Minimization Techniques for Nonlinear Functions," *SIAM Rev.*, 4(4):343-365 (1962).

104. Stong, R. E., "A Note on the Sequential Unconstrained Minimization Technique for Non-linear Programming," *Management Sci.*, **12**(1):142-144 (1965).
105. Thrall, R. M., and L. Tornheim, *Vector Spaces and Matrices*, Wiley, New York, 1957.
106. Valentine, F. A., "The Problem of Lagrange with Differential Inequalities as Added Side Conditions," in *Contributions to the Calculus of Variations, 1933-1937*, University of Chicago Press, Chicago, 1937, pp. 407-448.
107. White, W. B., S. H. Johnson, and G. B. Dantzig, "Chemical Equilibrium in Complex Mixtures," *J. Chem. Phys.*, **28**:751-755 (1958).
108. Widder, D. V., *Advanced Calculus*, 2nd Ed., Prentice-Hall, Englewood Cliffs, N.J., 1961, Chapter 4.
109. Wilde, D. J., *Optimum Seeking Methods*, Prentice-Hall, Englewood Cliffs, N.J., 1964.
110. Wolfe, P., "The Simplex Method for Quadratic Programming," *Econometrica*, **27**(3):382-398 (1959).
111. Wolfe, P., "A Duality Theorem for Nonlinear Programming," *Quart. Appl. Math.*, **19**(3):239-244 (1961).
112. Wolfe, P., "Methods of Nonlinear Programming," in R. L. Graves and P. Wolfe (Eds.), *Recent Advances in Mathematical Programming*, McGraw-Hill, New York, 1963, pp. 67-86.
113. Wolfe, P., "Foundations of Nonlinear Programming: Notes on Linear Programming and Extensions—Part 65," Memo RM-4669-PR, RAND Corp., Santa Monica, Calif., 1965.
114. Woodbury, M., "Inverting Modified Matrices," Memorandum Report 42, Statistical Research Group, Princeton, N.J., 1950.
115. Zangwill, W. I., "Non-linear Programming by Sequential Unconstrained Maximization," Working Paper 131, Center for Research in Management Science, University of California, Berkeley, 1965.
116. Zangwill, W. I., "On Minimizing a Function without Calculating Derivatives," Working Paper 210, Center for Research in Management Science, University of California, Berkeley, March 1967.
117. Zangwill, W. I., "Non-linear Programming via Penalty Functions," *Management Sci.*, **13**(5):344-358 (1967).
118. Zoutendijk, G., *Methods of Feasible Directions*, Elsevier, Amsterdam, 1960.
119. Zoutendijk, G., "Nonlinear Programming: A Numerical Survey," *SIAM J. Control*, **4**:194-210 (1966).