QUASI-NEWTON METHODS FOR UNCONSTRAINED FUNCTION MINIMIZATION AND THE SOLUTION OF SYSTEMS OF NONLINEAR EQUATIONS

A THESIS PRESENTED BY

CAESAR P.VIAZMINSKY

TO THE

UNIVERSITY OF ST.ANDREWS

IN APPLICATION FOR THE DEGREE OF

MASTER OF SCIENCE

OCTOBER,1974

Declaration

The accompanying thesis is my own composition.It is based on work carried out by me and no part of it has previously been presented in application for a higher degree.

Certificate

I certify that Mr C.Viazminsky who was admitted as a research student under ordinance general no 12 in october 1972 has fulfilled the conditions of the ordinance and regulations.

Research Supervisor

ABSTRACT

This thesis is concerned with the unconstrained minimization of a function of n variables, and, to a lesser extent, with the numerical solution of systems of nonlinear equations.

The first chapter contains an account of the fundamental ideas and theorems which are related to the subject of this thesis, and also gives a brief description of some methods which historically precede quasi-Newton methods, such as the method of steepest descent, Newton's method, the conjugate direction methods, the contraction mapping method, and the parameter variation method.

Newton's method, among the aforementioned methods, is considered the most effective one. It is rapidly convergent, and is capable of handling a variety of problems efficiently. But from a computational point of view, Newton's method is expensive. The second chapter of this thesis demonstrates how quasi-Newton methods are considered as an improvement of Newton's method by being able to circumvent the difficulties which face Newton's method. Also a general procedure for deriving quasi-Newton algorithms is described.

All methods generate a sequence of estimates which tend to the solution of the problem. In general all the methods which precede quasi-Newton methods employ information at the present stage, but quasi-Newton methods employ information at the present stage, and at the stage immediately previous to the present. In chapters 3 and 4 we will discuss methods which employ information from previous stages. Such methods are unified in one general scheme called "supermemory descent methods". Numerical experience with members of this class of methods is reported and compared with quasi-Newton methods.

CONTENTS

CHAPTER 4

SUPERMEMORY DESCENT METHODS FOR UNCONSTRAINED MINIMIZATION

CHAPTER 1

FUNDAMENTALS AND EARLIER METHODS

§(1.0) Introduction

In this thesis we consider the problem of finding unconstrained local minima of a differentiable function $f: R^n \to R^1$ by quasi-Newton methods. We consider also to a lesser extent the related problem of solving systems of nonlinear equations of the form $\underline{F}(\underline{x}) = \underline{0}$, where $\underline{F}: R^n \to R^n$ is given, also by quasi-Newton methods.

The thesis falls into two principal parts. In the first part comprising chapters 1 and 2, some well-established methods for minimizing f and for finding zeros of $\underline{F}$ are discussed. Chapter 1 contains a brief account of the fundamental ideas and theorems which are used subsequently, and also gives a review of some methods for unconstrained minimization and the solution of systems of nonlinear equations which historically precede quasi-Newton methods. Chapter 2 contains an account of quasi-Newton methods. The second part of the thesis comprising chapters 3 and 4 contains a report of some original work consisting of an extension of the method of Wolfe (1974) for unconstrained minimization.

§(1.1) Historical survey

In order to indicate how quasi-Newton methods for unconstrained minimization and for the solution of systems of nonlinear equations arose, we give a brief historical survey of some of the work which preceded the development of quasi-Newton methods.

In 1847 Cauchy used the method of steepest descent for unconstrained minimization, and it was subsequently used by Courant (1943), Curry (1944) and others. The convergence of the method of steepest descent is unacceptably slow and several attempts were made to improve on the method by, among others, Forsythe and Motzkin (1951), Akaike (1959), and Schinzinger (1966).

A more rapidly convergent method for both unconstrained minimization

and the solution of systems of nonlinear equations is Newton's method. Unlike the method of steepest descent however, Newton's method requires the evaluation of first and second derivatives of the function to be minimized, and when used for the solution of systems of nonlinear equations, it requires the evaluation of the Jacobian matrix. Newton's method is therefore computationally expensive when it converges. This led Levenberg (1944), Crockett and Chernoff (1955), Marquardt (1963) and others to modify Newton's method in order to increase its efficiency as a method for unconstrained minimization. The work of Barnes (1965) on the solution of systems of nonlinear equations is historically important in that it appears to be the first paper in which Newton's method is modified to produce an essentially quasi-Newton method.

A different approach to unconstrained minimization was initiated by the work of Hestenes and Stiefel (1952) on the solution of systems of linear equations by conjugate direction methods. In particular Fletcher and Reeves (1964) used the idea of Hestenes and Stiefel to construct a conjugate gradient method for unconstrained minimization, and Powell (1964) constructed a conjugate direction method for unconstrained minimization which does not require the evaluation of derivatives of the objective function.

An important development with far reaching consequences for unconstrained minimization and the solution of systems of nonlinear equations had, however, taken place in 1959 with the advent of Davidon's paper on the variable metric method of unconstrained minimization. This work may be said to be the true source of the quasi-Newton methods. Davidon's method was modified by Fletcher and Powell (1963) who showed that the modified form of Davidon's method, the so-called Davidon-Fletcher-Powell method, is also a conjugate direction method.

Broyden (1965) showed that the Davidon -Fletcher-Powell method could be regarded as a modification of Newton's method for finding zeros of the gradient of the objective function and proposed a class of methods, which

he called quasi-Newton methods, for the solution of general systems of nonlinear equations. Since the appearance of Broyden's paper in 1965 many other quasi-Newton methods have been proposed, some of which will be discussed subsequently.

A number of excellent reviews of unconstrained minimization and the solution of systems of nonlinear equations have been written, in particular, by Spang (1962), Powell (1966), Broyden (1970), Rosen (1966), Zeleznik (1968), Fletcher (1969), Powell (1971), and Dixon (1973).

§(1.2) Fundamental Definitions and Theorems Related to Unconstrained Minimization

We shall review here some basic definitions and theorems relevant to unconstrained minimization. For more detail see Ortega and Rheinboldt (1970).

Definition (1.1)

A point $\underline{x}^* \in R^n$, is a weak local minimum of a function $f: R^n \to R^1$, iff there exists an open neighbourhood S of $\underline{x}^*$ such that

$$f(\underline{x}) \geq f(\underline{x}^*) \quad (\forall \underline{x} \in S). \tag{1.1}$$

The point $\underline{x}^*$ is a strong local minimum of f, iff the strict inequality holds in (1.1) ($\forall\, \underline{x} \in S$, $\underline{x} \neq \underline{x}^*$). The point $\underline{x}^*$ is a global minimum of f on some subset D of R^n which contain x* iff (1.1) holds in D.

Since the algorithms to be described here are able to find only local minima, all kinds of minima will be referred to as minima. The term minimum will mean weak minimum unless a statement is made to the contrary. The process of finding a minimum of f is called minimization of f, and the function f itself is called the objective function.

Local and global maxima of f are defined in an analogous way, but since the maximization of f is equivalent to the minimization of -f, we restrict our work to the minimization problem.

Definition (1.2)

A point $\underline{x}^* \in R^n$ is a <u>critical point</u> of $f: R^n \to R^1$, iff f is G-differentiable at $\underline{x}^*$, and $f'(\underline{x}^*) = \underline{0}$, where f' denotes the first G-derivative of f. □

Thus $\underline{x}^*$ is a critical point of f iff

$$(\partial_1 f(\underline{x}^*), \ldots, \partial_n f(\underline{x}^*))^T = \underline{0}, \tag{1.2a}$$

or equivalently

$$\underline{g}(\underline{x}^*) \stackrel{D}{=} f'^T(\underline{x}^*) = \underline{0}, \tag{1.2b}$$

where $\underline{g}: R^n \to R^n$ is the gradient of f.

A critical point is sometimes called a <u>stationary point</u>. The relation between the concept of a minimum and that of a critical point is given by the following basic theorem.

Theorem (1.1)

If (i) $f: R^n \to R^1$ is G-differentiable at $\underline{x}^*$;

(ii) $\underline{x}^*$ is a local minimum of f,

then

$$\underline{g}(\underline{x}^*) = \underline{0}. \quad \square \tag{1.3}$$

It is clear that a critical point is not necessarily a local minimum. For example $\underline{x}^* = \underline{0}$ is a critical point of $f(x) = x^3$, $(x \in R)$, but it is not a minimum. In such a case $\underline{x}^*$ is called a <u>saddle point</u>.

The second derivative of f, when it exists, can characterize the nature of the critical point.

Theorem (1.2)

If (i) $f: R^n \to R^1$ is twice G-differentiable at $\underline{x}^*$;

(ii) $f''(\underline{x}^*)$ is positive definite;

(iii) $\underline{x}^*$ is a critical point of f;

then, $\underline{x}^*$ is a strong local minimum of f.

Conversely if (i) and (ii) hold and if also

(iv) $\underline{x}^*$ is a local minimum of f,

then, $f''(\underline{x}^*)$ is positive semi-definite. □

It is clear that for $f: R^n \to R^1$, $f''(\underline{x}^*)$ is positive definite (semi-definite) iff the Hessian matrix A defined by

$$A \triangleq (\partial_i \partial_j f(\underline{x})) \quad (i,j=1,\ldots,n), \tag{1.4}$$

is positive definite (semi-definite).

In T(1.1) we saw that finding a critical point of a function f is equivalent to finding a solution of the system $\underline{g}(\underline{x}) \triangleq f'(\underline{x})^T = \underline{0}$, and that the problem of determining a critical point of f is sometimes the same as that of finding a minimum of f. Hence if $\underline{g}$ is the gradient of a function f, then the problem of solving $\underline{g}(\underline{x}) = \underline{0}$ can sometimes be replaced by the problem of minimizing f. To determine a class of equations $\underline{F}(\underline{x}) = \underline{0}$, which can be solved in this manner, the following definition may be used.

Definition (1.3)

The function $\underline{F}: R^n \to R^n$ is a gradient mapping on R^n iff there exists a G-differentiable function $f: R^n \to R^1$, such that

$$\underline{F}(\underline{x}) = f'(\underline{x}) \quad (\forall \underline{x} \in R^n). \tag{1.5}$$ □

We then have the following theorem, referred to by Ortega and Rheinboldt (1970) as the symmetry principle.

Theorem (1.3)

If a function $\underline{F}: R^n \to R^n$, is continuously differentiable on R^n, then $\underline{F}$ is a gradient mapping iff $\underline{F}'(x)$ is symmetric for all $\underline{x} \in R^n$. □

From the gradient mapping $\underline{F}$, we can construct f by means of the formula

$$f(\underline{x}) = \int_0^1 (\underline{x} - \underline{x}^{(0)})^T \underline{F}(\underline{x}^{(0)} + t(\underline{x} - \underline{x}^{(0)}))\, dt. \tag{1.6}$$

The problem of solving a system of non-linear equations $\underline{F}(\underline{x}) = \underline{0}$ can always be converted into a minimization problem as follows.

Let $h:R^n \to R^1$ be such that $\underline{x} = \underline{o}$ is a unique global minimum of h. For example, $h(\underline{x}) = \| \underline{x} \|$, where $\|.\|$ is some arbitrary norm on R^n; a special case is the norm defined by $\| x \|_A \stackrel{D}{=} (\underline{x}^T A \underline{x})^{1/2}$ where A is a symmetric positive definite matrix. Let $f: R^n \to R^1$ be defined by

$$f(\underline{x}) = h(\underline{F}(\underline{x})) . \tag{1.7}$$

If the system $\underline{F}(\underline{x}) = \underline{0}$ has a solution $\underline{x}^* \in R^n$, then $\underline{x}^*$ is a global minimum of f. If $\underline{F}(\underline{x}) = \underline{0}$ has no solution on R^n, then $\underline{x}^*$ is called an h-minimal solution of $\underline{F}(\underline{x}) = \underline{0}$. If h is taken to be the Euclidean norm, so that $h(\underline{x}) = (\underline{x}^T \underline{x})^{1/2} (\underline{x} \in R^n)$, then an h-minimal solution is called a least squares solution of the problem.

The concept of h-minimal solution can be extended to a mapping $\underline{F}:R^n \to R^m$. In this case h is a function on R^n. If $n > m$ the system of equations is called underdetermined, for the number of equations is not enough to determine the solution. If $n < m$ the system is called overdetermined, for there are more equations to be satisfied than there are variables.

Sufficient conditions to ensure that a critical point of f is also a solution of $\underline{F}(\underline{x}) = \underline{0}$ are given in the following theorem.

Theorem (1.4)

Let $h:R^n \to R^1$ be F-differentiable in R^n and have a unique critical point at $\underline{x} = \underline{o}$. Let $\underline{F}: R^n \to R^n$ and define $f: R^n \to R^1$ by (1.7). Suppose that at $\underline{x}^* \in R^n$, $\underline{F}$ has a nonsingular G-derivative. Then $\underline{x}^*$ is a critical point of f iff $\underline{F}(\underline{x}^*) = \underline{o}$. □

The concept of convexity is of special importance in optimization theory, because in many cases, the convergence of a minimization algorithm can often be proved for convex functions.

Definition (1.4)

A set $D \subset R^n$ is convex iff

$$\lambda \underline{x} + (1-\lambda)\underline{y} \in D \quad (\forall \underline{x}, \underline{y} \in D)(\forall \lambda \in (0,1)). \qquad \square \tag{1.8}$$

Definition (1.5)

The function $f: R^n \to R^1$ is <u>convex</u> on the convex set $D \subset R^n$ iff

$$f(\lambda \underline{x} + (1-\lambda)\underline{y}) \leq \lambda f(\underline{x}) + (1-\lambda) f(\underline{y}) \quad (\forall \underline{x}, \underline{y} \in D; \forall \lambda \in (0,1)), \tag{1.9}$$

and f is <u>strictly convex</u> on D iff the strict inequality holds in (1.9) whenever $\underline{x} \neq \underline{y}$. $\square$

We then have the following important results.

Theorem (1.5)

If $f: R^n \to R^1$ is convex on the open convex set $D \subset R^n$, then f is continuous in D. $\square$

Theorem (1.6)

If $f: R^n \to R^1$ is G-differentiable on a convex set $D \subset R^n$, then f is convex in D iff

$$f'(\underline{x})(\underline{y}-\underline{x}) \leq f(\underline{y}) - f(\underline{x}) \quad (\forall \underline{x}, \underline{y} \in D). \tag{1.10}$$

Moreover, f is strictly convex on D iff strict inequality holds in (1.10), whenever $\underline{x} \neq \underline{y}$. $\square$

Theorem (1.7)

If $f: R^n \to R^1$, is G-differentiable on a convex set $D \subset R^n$, then f is convex in D iff

$$(f'(\underline{y}) - f'(\underline{x}))(\underline{y}-\underline{x}) \geq 0 \quad (\forall \underline{x}, \underline{y} \in D), \tag{1.11}$$

and f is strictly convex in D iff strict inequality holds in (1.11). $\square$

Theorem (1.8)

If $f: R^n \to R^1$, is twice G-differentiable on the convex set $D \subset R^n$, then f is convex on D iff $f''(\underline{x})$ is positive semi-definite for all $\underline{x} \in D$. Moreover, f is strictly convex on D if $f''(\underline{x})$ is positive definite for

all $\underline{x} \in D$. □

Theorem (1.9)

If (i) $f: R^n \to R^1$, is convex;

(ii) $\underline{x}^*$ is a local minimum of f on R^n;

then, $\underline{x}^*$ is a global minimum of f on R^n. □

We can deduce from T(1.9) that if f is convex, and $\underline{x}^*$ is a strong local minimum of f, then this minimum is unique.

A very important concept in the work to be described, and in the theory of unconstrained optimization in general is contained in the following definition.

Definition (1.6)

The function $f: R^n \to R^1$ is a quadratic function of $\underline{x}$ iff

$$f(\underline{x}) = \frac{1}{2}\underline{x}^T A \underline{x} + \underline{b}^T \underline{x} + c \quad , \tag{1.12}$$

where c is a constant scalar, $\underline{b}$ is a constant $(n \times 1)$ vector, and A is a constant symmetric $(n \times n)$ matrix. □

By T(1.2), f defined by (1.12) has a minimum if A is positive definite.

The gradient $\underline{g}$ of f defined by (1.12) is given by

$$\underline{g}(\underline{x}) = A\underline{x} + \underline{b} \, . \tag{1.13}$$

If A is positive definite then a necessary and sufficient condition for $\underline{x}^*$ to minimize f is

$$\underline{g}(\underline{x}^*) = \underline{0} \quad , \tag{1.14}$$

whence

$$\underline{x}^* = -A^{-1}\underline{b} \, . \tag{1.15}$$

From (1.13), if $\underline{g}^{(k)} = \underline{g}(\underline{x}^{(k)})$ $(k \geq 0)$, we have

$$\Delta \underline{g}^{(k)} = A \Delta \underline{x}^{(k)} \, , \tag{1.16}$$

where

$$\Delta \underline{g}^{(k)} = \underline{g}^{(k+1)} - \underline{g}^{(k)}, \tag{1.17}$$

and

$$\Delta \underline{x}^{(k)} = \underline{x}^{(k+1)} - \underline{x}^{(k)} . \tag{1.18}$$

Definition (1.7)

The set $\{\underline{p}^{(o)},\ldots,\underline{p}^{(m)}\}$ $(m < n)$ of $(n \times 1)$ vectors is <u>A-conjugate</u> where A is an $(n \times n)$ matrix iff

$$\underline{p}^{(i)T} A \underline{p}^{(j)} = 0 \quad (i \neq j;\ i,j = 0,\ldots,m). \square \tag{1.19}$$

If A is positive definite and $\{\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}\}$ are A-conjugate, then $\{\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}\}$ are linearly independent, and span R^n. See Luenberger (1973).

In general the Gram-Schmidt orthogonalization procedure can be used to produce a set of n A-conjugate directions starting from any n linearly independent vectors. See T(1.27).

Let $\{\underline{x}^{(k)}\}$, where $\underline{x}^{(o)}$ is given, be a sequence of vectors generated from

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \alpha^{(k)} \underline{p}^{(k)} \qquad (k \geqslant 0), \tag{1.20}$$

where $\{\underline{p}^{(k)}\}$ is a sequence of A-conjugate vectors and $\alpha^{(k)}$ $(k \geqslant o)$, is defined by

$$f(\underline{x}^{(k)} + \alpha^{(k)} \underline{p}^{(k)}) = \min_{\alpha} f(\underline{x}^{(k)} + \alpha \underline{p}^{(k)}), \tag{1.21}$$

in which f is defined by (1.12).

From (1.20), (1.21) we deduce that

$$\underline{g}^{(k+1)T} \underline{p}^{(k)} = 0 \qquad (k \geqslant 0). \tag{1.22}$$

Theorem (1.10)

The vectors $\underline{p}^{(o)},\ldots,\underline{p}^{(k)}$ are A-conjugate iff

$$\underline{g}^{(k+1)T}\underline{P}^{(i)}=0 \quad (i=0,1,\ldots,k<n). \qquad (1.23)$$

□

Proof

Let $\underline{p}^{(o)},\ldots,\underline{p}^{(k)}$ be A-conjugate. Then, from (1.16), (1.17), (1.20)

$$\underline{g}^{(k+1)}=\underline{g}^{(i+1)}+\sum_{j=i+1}^{k}\alpha^{(j)}A\underline{P}^{(j)} \quad (i=0,1,\ldots,k-1). \qquad (1.24)$$

Post-multiplying both sides of the transpose of (1.24) by $\underline{p}^{(i)}$ we find

$$\underline{g}^{(k+1)T}\underline{P}^{(i)}=\underline{g}^{(i+1)T}\underline{P}^{(i)}+\sum_{j=i+1}^{k}\alpha^{(j)}\underline{P}^{(j)T}A\underline{P}^{(i)}. \qquad (1.25)$$

By (1.22) and (1.19) we find that (1.23) is satisfied.

Conversely, if (1.23) is satisfied, then

$$\Delta\underline{g}^{(k)T}\underline{P}^{(i)}=0 \quad (i=0,1,\ldots,k-1;k<n). \qquad (1.26)$$

Hence, by (1.16), (1.20),

$$\underline{P}^{(k)T}A\underline{P}^{(i)}=0 \quad (i=0,1,\ldots,k-1;k<n).$$

□

We denote the subspace of R^n spanned by the independent vectors $\{\underline{p}^{(o)},\ldots,\underline{p}^{(k-1)}\}$ by $S^{(k)}$. Then we have the following definition.

Definition (1.8)

The k dimensional linear variety $\underline{x}^{(o)}+S^{(k)}$ is defined by

$$\underline{x}^{(o)}+S^{(k)}\triangleq\left\{\underline{x}\,\middle|\,\underline{x}=\underline{x}^{(o)}+\sum_{i=0}^{k-1}\alpha^{(i)}\underline{P}^{(i)},\ \alpha^{(i)}\in R\ (i=0,\ldots,k-1)\right\}. \qquad (1.27)$$

□

Sometimes the linear variety $\underline{x}^{(o)}+S^{(k)}$ is referred to as the k-dimensional hyperplane spanned by $\{\underline{p}^{(o)},\ldots,\underline{p}^{(k-1)}\}$ and passing through $\underline{x}^{(o)}$.

Theorem (1.11)

If the iteration given by (1.20), (1.21) is applied to the function f defined by (1.12), then $\underline{x}^{(k)}$ is a minimum of f on the linear variety $\underline{x}^{(o)} + S^{(k)}$ iff $\underline{g}^{(k)}$ is orthogonal to the k-dimensional subspace $S^{(k)}$. □

For a proof see Pearson (1969).

Theorem (1.12)

If $\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}$ are n A-conjugate vectors, and the iteration given by (1.20), (1.21) is applied to the function f defined by (1.12), then $\underline{x}^{(k)}$ minimizes f on the linear variety $\underline{x}^{(o)} + S^{(k)}$. Moreover $\underline{x}^{(m)}$ is the minimum of f on R^n, for some $m \leqslant n$. □

The first part of T(1.12) is a result of T(1.10), T(1.11). The second part is clear by noticing that $\underline{x}^{(n)}$ is a minimum of f on $\underline{x}^{(o)} + S^{(n)}$.

Theorem (1.13)

If (i) $\underline{x}^{(o)} \in R^n$ is arbitrary;

(ii) $\underline{x}^{(k)}$ (k = o,...,n-1) are generated by using (1.20), (1.21), with $\underline{p}^{(k)}$ (k = o,...,n-1) generated by using

$$\underline{p}^{(o)} = -\underline{g}^{(o)}, \tag{1.28}$$

$$\underline{p}^{(k+1)} = -\underline{g}^{(k+1)} + \beta^{(k)}\underline{p}^{(k)} \quad (k=0,1,\ldots,n-2), \tag{1.29}$$

where

$$\beta^{(k)} = \frac{\underline{g}^{(k+1)T}\underline{g}^{(k+1)}}{\underline{g}^{(k)T}\underline{g}^{(k)}}; \tag{1.30}$$

(iii) f is given by (1.12),

then $\underline{p}^{(k)}$, (k = o,...,n-1) are A-conjugate. □

Proof

We have from (iii),

$$\Delta\underline{g}^{(k)} = \alpha^{(k)}A\underline{p}^{(k)} \quad (k \geqslant o).$$

By (ii)

$$\underline{p}^{(1)} A \underline{p}^{(0)} = -\underline{g}^{(1)T} A \underline{p}^{(0)} + \beta^{(0)} \underline{p}^{(0)} A \underline{p}^{(0)}$$

$$= -\frac{1}{\alpha^{(0)}} \underline{g}^{(1)T} (\underline{g}^{(1)} - \underline{g}^{(0)}) - \frac{\beta^{(0)}}{\alpha^{(0)}} \underline{g}^{(0)T} (\underline{g}^{(1)} - \underline{g}^{(0)}). \qquad (1.31)$$

Now by virtue of the line search

$$\underline{g}^{(1)T} \underline{p}^{(0)} = -\underline{g}^{(1)T} \underline{g}^{(0)} = 0,$$

and by (ii)

$$\beta^{(0)} \underline{g}^{(0)T} \underline{g}^{(0)} = \underline{g}^{(1)T} \underline{g}^{(1)}.$$

Hence

$$\underline{p}^{(1)T} A \underline{p}^{(0)} = 0.$$

So $\underline{p}^{(0)}, \underline{p}^{(1)}$ are A-conjugate.

Suppose that $\underline{p}^{(0)}, \ldots, \underline{p}^{(m)}$ are A-conjugate for some $m < n-1$. Then for $i = 0, \ldots, m-1$

$$\underline{p}^{(i)T} A \underline{p}^{(m+1)} = \underline{p}^{(i)T} A (-\underline{g}^{(m+1)} + \beta^{(m)} \underline{p}^{(m)})$$

$$= -\underline{p}^{(i)T} A \underline{g}^{(m+1)}$$

$$= -\frac{1}{\alpha^{(i)}} (\underline{g}^{(i+1)} \underline{g}^{(m+1)} - \underline{g}^{(i)T} \underline{g}^{(m+1)}). \qquad (1.32)$$

But

$$\underline{p}^{(k)T} \underline{g}^{(m+1)} = (-\underline{g}^{(k)T} + \beta^{(k-1)} \underline{p}^{(k-1)})^T \underline{g}^{(m+1)}.$$

Hence

$$\underline{g}^{(k)T} \underline{g}^{(m+1)} = \beta^{(k-1)} \underline{p}^{(k-1)T} \underline{g}^{(m+1)} - \underline{p}^{(k)T} \underline{g}^{(m+1)}$$

$$= 0 \quad (k = 1, \ldots, m) \quad (\text{by } T(1.10)),$$

and

$$\underline{g}^{(0)T}\underline{g}^{(m+1)} = -\underline{p}^{(0)}\underline{g}^{(m+1)} = 0 \quad (\text{by } T(1.10)).$$

So

$$\underline{g}^{(k)T}\underline{g}^{(m+1)} = 0 \quad (k = 0, \ldots, m). \tag{1.33}$$

Hence by (1.32) and (1.33)

$$\underline{p}^{(i)T} A \underline{p}^{(m+1)} = 0 \quad (i = 0, \ldots, m-1). \tag{1.34}$$

Finally

$$\underline{p}^{(m)T} A \underline{p}^{(m+1)} = \underline{p}^{(m)T} A (-\underline{g}^{(m+1)} + \beta^{(m)} \underline{p}^{(m)})$$

$$= -\underline{p}^{(m)T} A \underline{g}^{(m+1)} + \beta^{(m)} \underline{p}^{(m)T} A (-\underline{g}^{(m)} + \beta^{(m-1)} \underline{p}^{(m-1)})$$

$$= -\frac{1}{\alpha^{(m)}} (\underline{g}^{(m+1)} - \underline{g}^{(m)})^T \underline{g}^{(m+1)} - \frac{\beta^{(m)}}{\alpha^{(m)}} (\underline{g}^{(m+1)} - \underline{g}^{(m)})^T \underline{g}^{(m)}$$

$$= -\frac{1}{\alpha^{(m)}} (\underline{g}^{(m+1)T}\underline{g}^{(m+1)} - \beta^{(m)} \underline{g}^{(m)T} \underline{g}^{(m)}) = 0,$$

since by (1.32) $\underline{g}^{(m)T}\underline{g}^{(m+1)} = 0$, and by (ii),

$$\underline{g}^{(m+1)T}\underline{g}^{(m+1)} = \beta^{(m)} \underline{g}^{(m)T} \underline{g}^{(m)}. \tag{1.35}$$

Hence

$$\underline{p}^{(i)T} A \underline{p}^{(j)} = 0 \quad (i \neq j;\ i, j = 0, 1, \ldots, m+1). \tag{1.36}$$

So by induction, $\underline{p}^{(k)}$, $(k = 0, \ldots, n-1)$ are A-conjugate. □

Definition (1.9)

If $\underline{x} + S^{(k)}$ and $\underline{y} + S^{(k)}$ are two k-dimensional linear varieties, then $\underline{x} + S^{(k)}$ and $\underline{y} + S^{(k)}$ are <u>parallel</u> iff $\underline{x} + S^{(k)}$ and $\underline{y} + S^{(k)}$ are disjoint. □

Theorem (1.14)

If (i) $\underline{x} + S^{(k)}$ and $\underline{y} + S^{(k)}$ are k-dimensional parallel linear varieties generated by the independent vectors $\underline{p}^{(o)},\ldots,\underline{p}^{(k-1)}$;

(ii) f is defined by (1.12);

(iii) $\underline{z}^{(1)}$ and $\underline{z}^{(2)}$ are minima of f in $\underline{x} + S^{(k)}$ and $\underline{y} + S^{(k)}$ respectively;

then, $\underline{z}^{(2)} - \underline{z}^{(1)}$ is conjugate to $\underline{p}^{(o)},\ldots,\underline{p}^{(k-1)}$.

Proof

By T(1.11) we have

$$\underline{g}(\underline{z}^{(2)})^T \underline{p}^{(i)} = \underline{g}(\underline{z}^{(1)})\underline{p}^{(i)} = 0 \quad (i = 0,1,\ldots,k-1). \tag{1.37}$$

Hence

$$(\underline{g}(\underline{z}^{(2)}) - \underline{g}(\underline{z}^{(1)}))\,\underline{p}^{(i)} = 0 \qquad (i = 0,1,\ldots,k-1). \tag{1.38}$$

By (1.16)

$$(\underline{z}^{(2)} - \underline{z}^{(1)})^T A\, \underline{p}^{(i)} = 0 \qquad (i = 0,1,\ldots,k-1). \tag{1.39}$$

□

Definition (1.10)

A method for function minimization is quadratically convergent iff it is capable of minimizing a strictly convex quadratic function in n steps at most. □

Definition (1.11)

A sequence $\{\underline{x}^{(k)}\}$ in a normed linear space X converges superlinearly to $\underline{x}^* \in X$ iff

$$\lim_{k\to\infty} \frac{\|\underline{x}^{(k+1)} - \underline{x}^*\|}{\|\underline{x}^{(k)} - \underline{x}^*\|} = 0 . \tag{1.40}$$

□

Definition (1.12)

A sequence $\{\underline{x}^{(k)}\}$ in a normed linear space X converges to $\underline{x}^* \in X$

with order of convergence p iff $\exists$ K > 0 such that

$$\|\underline{x}^{(k+1)} - \underline{x}^*\| \leq K \|\underline{x}^{(k)} - \underline{x}^*\|^p \quad (\forall k \geq 0), \tag{1.41}$$

where K is a constant independent of k. □

§(1.3) Some Methods for Unconstrained Minimization

In this section we give a brief description of some methods for unconstrained minimization, excluding quasi-Newton methods which will be considered subsequently.

All methods considered in this chapter are descent methods of the form given in the following algorithm.

Algorithm (1.1)

Given an initial estimate $\underline{x}^{(o)}$ of a minimum $\underline{x}^*$ of f.

1. Set k = o.
2. Compute $\underline{x}^{(k+1)}$ from

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \alpha^{(k)} \underline{p}^{(k)}, \tag{1.42}$$

where

$$f(\underline{x}^{(k)} + \alpha^{(k)} \underline{p}^{(k)}) = \min_{\alpha} f(\underline{x}^{(k)} + \alpha \underline{p}^{(k)}), \tag{1.43}$$

and

$$\underline{g}^{(k)T} \underline{p}^{(k)} < o. \tag{1.44}$$

3. Set k = k+1 and go to 2. □

Condition (1.44) ensures that a positive value of $\alpha^{(k)}$ exists such that the descent property

$$f(\underline{x}^{(k+1)}) < f(\underline{x}^{(k)}) \quad (\forall k \geq o), \tag{1.45}$$

is satisfied.

Different methods are obtained by making various choices of $\underline{p}^{(k)}$. We consider here the following methods.

1. Steepest Descent

The method of steepest descent is the oldest in the history of function minimization. It was used for the first time by Cauchy (1847), and since then many others used it (Levenberg (1943), Curry (1944), Booth (1957), etc..). The search direction $\underline{p}$ in this method is chosen to be the negative gradient $-\underline{g}$. It is clear that this choice produces the fastest local decrease in the objective function $f(\underline{x})$.

The Taylor expansion of $f(\underline{x})$ in a neighbourhood of $\underline{x}^{(k)}$ is given by

$$f(\underline{x}^{(k)}+\underline{p}) = f(\underline{x}^{(k)}) + \underline{p}^T \underline{g}^{(k)} + \frac{1}{2}\underline{p}^T A^{(k)} \underline{p} + o(\|\underline{p}\|^3). \qquad (1.46)$$

The significance of the previous choice of search is that it minimizes the first variation $\underline{p}^T \underline{g}^{(k)}$ subject to the constraint

$$\underline{p}^T \underline{p} = c, \qquad (1.47)$$

where c is a constant (Murray (1972)).

Steepest descent converges to a stationary point under weak conditions on the function f (Curry (1944)). However, the method is slowly convergent in general. Forsythe and Motzkin (1951) noticed that the points predicted by the steepest descent algorithm tend to alternate between two straight lines, so they incorporated acceleration searches along these directions. Akaike (1959) proved a conjecture of Forsythe and Motzkin on the ultimate nature of convergence of this method for a positive definite quadratic form, namely that the directions generated by the method are ultimately asymptotic to two directions, so that eventually the minimum is approached in a two dimensional subspace; this accounts for the slow convergence of the method. Booth (1957) suggested that an incomplete step be taken in the search direction. This ensures that the directions of

successive steps are less likely to oscillate when a steep-sided valley is encountered. This suggestion is reported to have achieved some remarkable success, which is to be expected from Akaike's analysis of the ultimate nature of convergence. Schinzinger (1966) suggested that after each sequence of n steepest descent searches a step in the resultant direction be taken.

No modification to steepest descent was satisfactory, and it became clear that a good method would have the characteristic of generating a set of linearly independent search directions, which is the case for the class of methods based on the use of conjugate directions for a convex quadratic function.

2. Newton's Method

The search direction for Newton's method is given by the equation

$$\underline{p}^{(k)} = -A^{(k)-1}\,\underline{g}^{(k)} \qquad (k \geq 0), \tag{1.48}$$

where $A^{(k)}$ is the Hessian matrix of the objective function f evaluated at $\underline{x}^{(k)}$. This choice of $\underline{p}^{(k)}$ minimizes the first variation $\underline{p}^{(k)T}\underline{g}^{(k)}$ of f subject to the constraint

$$\left\| \underline{p}^{(k)} \right\|_A^2 = \underline{p}^{(k)T} A^{(k)} \underline{p}^{(k)} = c\,, \tag{1.49}$$

where c is a constant.

Relation (1.49) is meaningful provided that $A^{(k)}$ is positive definite if the norm $\|.\|_A$ is used. When A varies from one point to another but is still positive definite the metric is variable, and Newton's method is said to be a variable metric method. Using equations (1.42), (1.48), (1.14), (1.15) we deduce that Newton's method is capable of minimizing a convex quadratic function in one step with $\alpha = 1$. Thus Newton's method works very well near a minimum where the function is approximately quadratic.

For general functions, Newton's search direction given in (1.48) is not the steepest direction because the Hessian A is not necessarily positive

definite so that $(\underline{p}^{(k)T}A\underline{p}^{(k)})^{\frac{1}{2}}$ is not an admissible norm of $\underline{p}^{(k)}$. It follows that $\alpha^{(k)}$ is not necessarily positive (Murray (1972)).

Newton's method has ultimate second order convergence; that is for $\underline{x}^*$ sufficiently close to $\underline{x}^*$,

$$\|\underline{x}^{(k+1)} - \underline{x}^*\| \leqslant K \|\underline{x}^{(k)} - \underline{x}^*\|^2, \tag{1.50}$$

where K is a positive finite number. See (Ortega and Rheinboldt (1970), p.184).

Newton's method is very efficient in handling a variety of problems, but has the disadvantage of being laborious from a computational point of view, because of the necessity of evaluating the second partial derivatives of f.

3. Conjugate Direction Methods

Conjugate direction methods were originated by Hestenes and Stiefel (1952). Two methods based on conjugate direction are considered here, namely the methods of Fletcher and Reeves (1964) and of Powell (1964).

(i) The Method of Fletcher and Reeves (FR)

The search direction $\underline{p}^{(k)}$ for the (FR) method is given by

$$\underline{p}^{(0)} = -\underline{g}^{(0)}, \tag{1.51a}$$

$$\underline{p}^{(k)} = -\underline{g}^{(k)} + \frac{\underline{g}^{(k)T}\underline{g}^{(k)}}{\underline{g}^{(k-1)T}\underline{g}^{(k-1)}}\underline{p}^{(k-1)} \quad (k \geqslant 1). \tag{1.51b}$$

By T(1.13), T(1.12) the FR method is quadratically convergent. We have the following algorithm, which includes resetting.

Algorithm (1.2)

Given an initial estimate $\underline{x}^{(0)}$ of a minimum $\underline{x}^*$ of f.

1. Compute $\underline{g}^{(0)}$ from

$$\underline{g}^{(0)} = \underline{g}(\underline{x}^{(0)}).$$

2. Set $k = o$.

3. Set $\underline{p}^{(o)} = -\underline{g}^{(o)}$.

4. Compute $\alpha^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} + \alpha^{(k)}\underline{p}^{(k)}) = \min_{\alpha} f(\underline{x}^{(k)} + \alpha \underline{p}^{(k)}),$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \alpha^{(k)}\underline{p}^{(k)}.$$

5. Compute $\underline{g}^{(k+1)} = \underline{g}(\underline{x}^{(k+1)})$.

6. Set $k = k+1$.

7. If $k = n$, set $\underline{x}^{(o)} = \underline{x}^{(n)}$, $\underline{g}^{(o)} = \underline{g}^{(n)}$ and go to 2; otherwise go to 8.

8. Set $\beta^{(k)} = \dfrac{\underline{g}^{(k+1)T}\underline{g}^{(k+1)}}{\underline{g}^{(k)T}\underline{g}^{(k)}}$.

9. Set $\underline{p}^{(k+1)} = -\underline{g}^{(k+1)} + \beta^{(k)}\underline{p}^{(k)}$, and go to 4. □

(ii) Powell's Method Without Derivatives

Smith (1962) was the first to suggest the use of the property of parallel spaces given in T(1.14) to obtain conjugacy, and devised a method capable of minimizing a convex quadratic function in n steps. No derivatives were needed.

Unfortunately difficulties arise when applying the method iteratively to general functions of many variables, (Fletcher (1965), Box (1966)). The method of Smith will not be discussed here, but a more elaborate method of using the parallel subspace property will be described, namely the method of Powell (1964).

Algorithm (1.3)

Given an initial estimate $\underline{x}^{(o)}$ of a minimum $\underline{x}^*$ of f.

1. For $k = 0,1,\ldots,n-1$, calculate $\alpha^{(k)}$ to minimize $f(\underline{x}^{(k)} + \alpha^{(k)}\underline{p}^{(k)})$ and set $\underline{x}^{(k+1)} = \underline{x}^{(k)} + \alpha^{(k)}\underline{p}^{(k)}$.

2. For $k = 0,1,\ldots,n-2$ replace $\underline{p}^{(k)}$ by $\underline{p}^{(k+1)}$, and replace $\underline{p}^{(n-1)}$ by $\underline{x}^{(n)} - \underline{x}^{(o)}$.

3. Choose α to minimize $f(\underline{x}^{(n)} + \alpha(\underline{x}^{(n)} - \underline{x}^{(o)}))$, replace $\underline{x}^{(o)}$ by

$\underline{x}^{(o)} + \alpha(\underline{x}^{(n)} - \underline{x}^{(o)})$, and start the next iteration from 1.

In practice, it is found that for some problems the set $\{\underline{p}^{(k)}\}$ tends to become linearly dependent, so Powell modified his algorithm omitting step (2) when $\{\underline{p}^{(k)}\}$ tends to become dependent, but this modification although circumventing the difficulty, deprives the method of its quadratic convergence.

Fletcher (1965) and Box (1966) compared this method with other methods which use only function values, such as the methods of (Smith (1962), Nelder and Mead (1965), Davies, Swann and Campey (1965)). This comparison confirmed that Powell's method is more efficient than the method of Smith, and of Davies, Swann and Campey. For a large number of variables Powell's method is more efficient than the simplex method of Nelder and Mead. However, Zangwill (1967) posed a counter example showing that the original algorithm of Powell does not converge to a minimum of a strictly convex quadratic function in any number of iterations. He also developed a method in which step (2) is retained, and quadratic convergence preserved, and proved that the method converges to the minimum of a strictly convex function provided an exact line search is performed. No numerical experiments were, however performed by Zangwill.

§(1.4) Line Search Procedures

Essentially the line search is the estimation of the value of $\alpha^{(k)}$ which minimizes the objective function $f(\underline{x})$ along the straight line $\underline{x} = \underline{x}^{(k)} + \alpha\underline{p}^{(k)}$, so that

$$\phi'(\alpha^{(k)}) = 0, \tag{1.52}$$

where

$$\phi(\alpha) = f(\underline{x}^{(k)} + \alpha\underline{p}^{(k)}). \tag{1.53}$$

In general, $\phi'(\alpha) = 0$ is a nonlinear equation which cannot be solved

analytically. Hence $\alpha^{(k)}$ must be determined numerically. The methods which are used in practice for estimating $\alpha^{(k)}$ are based upon fitting ϕ at a number of points with a polynomial Φ of degree 2 or 3 and determining the minimum of the polynomial analytically. The point obtained replaces one of the points at which the polynomial is fitted to ϕ, and the procedure is repeated. In this case a sequence of estimates of $\alpha^{(k)}$ is obtained which under very general conditions will converge to $\alpha^{(k)}$.

The polynomial of degree 2 which interpolates ϕ at α_1, α_2, α_3 is $\Phi(\alpha)$ where

$$\Phi(\alpha) = \sum_{k=1}^{3} \phi_k \prod_{\substack{j=1 \\ k \neq j}}^{3} \frac{(\alpha - \alpha_j)}{(\alpha_k - \alpha_j)} \,. \tag{1.54}$$

By evaluating $\Phi'(\alpha)$ and setting $\Phi'(\alpha) = 0$, we find

$$\alpha = \frac{1}{2} \frac{(\alpha_2^2 - \alpha_3^2)\phi_1 + (\alpha_3^2 - \alpha_1^2)\phi_2 + (\alpha_1^2 - \alpha_2^2)\phi_3}{(\alpha_2 - \alpha_3)\phi_1 + (\alpha_3 - \alpha_1)\phi_2 + (\alpha_1 - \alpha_2)\phi_3} \,. \tag{1.55}$$

Now, (1.55) gives a turning point of the interpolating polynomial of degree 2 which interpolates ϕ at α_1, α_2, α_3. If $\alpha_1 < \alpha_2 < \alpha_3$ and $\phi_1 \geq \phi_2 \leq \phi_3$, then α must lie between α_1 and α_3 and must be a minimum of Φ. Let α computed from (1.55) be called α_4, and let $\phi_4 = \phi(\alpha_4)$. The new set of interpolating points $\{\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3\}$ is selected so that

$$(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3) = (\alpha_1, \alpha_4, \alpha_2), \quad \text{if} \quad \phi_1 \geq \phi_4 \leq \phi_2 \,;$$

$$(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3) = (\alpha_2, \alpha_4, \alpha_3), \quad \text{if} \quad \phi_2 \geq \phi_4 \leq \phi_3 \,;$$

$$(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3) = (\alpha_1, \alpha_2, \alpha_4), \quad \text{if} \quad \phi_2 < \phi_4 \leq \phi_3 \,;$$

$$(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3) = (\alpha_4, \alpha_2, \alpha_3), \quad \text{if} \quad \phi_1 \geq \phi_4 > \phi_2 \,.$$

It can be shown (Luenberger (1973)) that if we proceed in this manner we obtain convergence for any three initial estimates of $\alpha^{(k)}$ for which $\alpha_1 < \alpha_2 < \alpha_3$ and $\phi_1 \geq \phi_2 \leq \phi_3$, for a function ϕ which has only one minimum.

To obtain α_1, α_2, α_3 initially we could proceed as follows. Compute $\phi_k = \phi(2^k h)$ $(k = 0,1,2,3,\ldots)$ where h is given until a value k is obtained such that $\phi_{k-1} < \phi_k$. Then we could take

$$\alpha_1 = 2^{k-2}h \ , \ \alpha_2 = 2^{k-1}h \ , \ \alpha_3 = 2^k h. \tag{1.56}$$

We note that the procedure does not require the evaluation of any derivatives of $\phi(\alpha)$ and so could be used in conjunction with those methods for unconstrained minimization which do not require the evaluation of derivatives, and in certain methods for the solution of systems of nonlinear equations.

The majority of unconstrained minimization methods require the computation of the gradient of f. The first derivative $\phi'(\alpha)$ of $\phi(\alpha)$ defined by (1.53) is given by

$$\phi'(\alpha) = \underline{g}^T(\underline{x}^{(k)} + \alpha \underline{p}^{(k)})\,\underline{p}^{(k)} . \tag{1.57}$$

Thus $\phi(\alpha)$ and $\phi'(\alpha)$ are computable for any α. In particular

$$\phi(0) = f^{(k)}, \tag{1.58a}$$

and

$$\phi'(0) = \underline{p}^{(k)T}\underline{g}^{(k)} \quad (\leq 0), \tag{1.58b}$$

which are already available.

The cubic search technique was proposed by Davidon (1959) and also used by Fletcher and Powell (1963), and Fletcher and Reeves (1964). The calculation is in three stages:

(i) estimation of the order of magnitude of $\alpha^{(k)}$;

(ii) establishment of bounds on $\alpha^{(k)}$;

(iii) cubic interpolation to estimate the value of $\alpha^{(k)}$.

(i) We choose a unit for α which corresponds to a displacement along

$\underline{p}^{(k)}$ of unit length in the $\underline{x}$-space. Let f_e be the estimated value of $f(\underline{x}^*)$. If $\underline{x}^*$ lies on the line $\underline{x}^{(k)} = \underline{x}^{(k)} + \alpha \underline{p}^{(k)}$, and if f is a quadratic function of $\underline{x}$, then it can be shown that

$$\alpha^* = \frac{-2(f^{(k)} - f_e)}{\underline{p}^{(k)T} \underline{g}^{(k)}} , \tag{1.59}$$

where α^* is clearly the value of α which corresponds to the minimum $\underline{x}^*$.

In fact the unconstrained minimum $\underline{x}^*$ will not lie on the line $\underline{x} = \underline{x}^{(k)} + \alpha \underline{p}^{(k)}$ in general, so that (1.59) will tend to overestimate $\alpha^{(k)}$. We therefore follow Fletcher and Reeves in making a tentative step length h given by

$$h = S \quad (0 < S < 1/(\underline{p}^{(k)T} \underline{p}^{(k)})^{\frac{1}{2}}) , \tag{1.60a}$$

$$h = 1/(\underline{p}^{(k)T} \underline{p}^{(k)})^{\frac{1}{2}} \quad (\text{otherwise}). \tag{1.60b}$$

where

$$S = \frac{-2(f^{(k)} - f_e)}{\underline{p}^{(k)T} \underline{g}^{(k)}} . \tag{1.61}$$

In some cases there is an expected value for α. For instance, in the method of Fletcher and Powell, it is known that as $\underline{x}^{(k)} \to \underline{x}^*$, $\alpha^{(k)} \to 1$, so they take

$$h = S \quad (0 \leqslant S \leqslant 1/(\underline{p}^{(k)T} \underline{p}^{(k)})^{\frac{1}{2}}) , \tag{1.62a}$$

$$h = 1 \quad (\text{otherwise}). \tag{1.62b}$$

(ii) In the second stage, $\phi'(\alpha)$ is examined at $\alpha = 0, h, 2h, 4h, \ldots, a, b$, where α is doubled each time and where b is the first of these values at which either $\phi'(\alpha) \geqslant 0$ or $\phi(b) \geqslant \phi(a)$. It then follows that $a < \alpha^{(k)} < b$.

(iii) The third stage uses cubic interpolation given by Davidon (1959). If

$$Z = 3\,\frac{(\phi(a)-\phi(b))}{(b-a)} + \phi'(a) + \phi'(b), \tag{1.63a}$$

$$w = (z^2 - \phi'(a)\,\phi'(b))^{\frac{1}{2}}, \tag{1.63b}$$

then the estimate c of $\alpha^{(k)}$ is given by

$$c = b - \frac{(\phi'(b)+w-z)}{(\phi'(b)-\phi'(a)+2w)}(b-a). \tag{1.64}$$

See appendix (1).

If neither $\phi(a)$ nor $\phi(b)$ is less than $\phi(c)$ then c is accepted as a final estimate of $\alpha^{(k)}$. Otherwise according as $\phi'(c)$ is positive or negative the interpolation is repeated over the subinterval (a,c) or (c,b) respectively.

A single application of the interpolation formula produces the exact value of $\alpha^{(k)}$ in the limit as $f(\underline{x})$ becomes quadratic near $\underline{x}^*$. Increased accuracy in the general case is obtained only at the cost of further evaluations of $f(\underline{x})$ and $\underline{g}(\underline{x})$. As the only region where the interpolation is likely to be inaccurate is that remote from $\underline{x}^*$, it is uneconomic to require high accuracy in this region. Numerical tests have shown that no significant reduction in the number of iterations when using the FR method can be achieved by using higher accuracy interpolation. From the point of view of stability, however, what must be done is to ensure that the values of $f(\underline{x}^{(k)})$ do form a decreasing sequence; hence the provision for repeating the interpolation over small intervals. A flow diagram for the line search procedure is given in appendix (1) as well.

§(1.5) Fundamental Definitions and Theorems Related to the Solution of Systems of Nonlinear Equations

Definition (1.13)

Let $\underline{P}$: $R^n \to R^n$ be given, then $\underline{x}^* \in R^n$ is a <u>fixed point</u> of $\underline{P}$ iff

$$\underline{x}^* = \underline{P}(\underline{x}^*). \quad \square \tag{1.65}$$

Hence, finding a fixed point of a function $\underline{P}$ is equivalent to solving (1.65).

Given an initial point $\underline{x}^{(o)}$ and a mapping $\underline{P}$: $R^n \to R^n$, let the sequence $\{\underline{x}^{(k)}\}$ be defined by

$$\underline{x}^{(k+1)} = \underline{P}(\underline{x}^{(k)}) \quad (k \geq 0). \tag{1.66}$$

Then we have the following theorem

Theorem (1.15)

If (i) $\underline{P}$: $R^n \to R^n$ is continuous on R^n;

(ii) The sequence $\{\underline{x}^{(k)}\}$ defined by (1.66) converges for some $\underline{x}^{(o)}$ in R^n to a point $\underline{x}^*$,

then, $\underline{x}^*$ is a fixed point of $\underline{P}$ on R^n. $\square$

Definition (1.14)

A function $\underline{P}$: $R^n \to R^n$, is a <u>contraction mapping</u> on the closed ball $\bar{B}(\underline{x}^{(o)}, r)$, iff there exists a number $0 \leq \alpha < 1$ (called the <u>contraction factor</u>) such that

$$\|\underline{P}(\underline{x}) - \underline{P}(\underline{y})\| \leq \alpha \|\underline{x} - \underline{y}\| \quad (\forall \underline{x}, \underline{y} \in \bar{B}(\underline{x}^{(o)}, r)). \quad \square \tag{1.67}$$

It follows from D(1.14) that $\underline{P}$ is continuous on $B(\underline{x}^{(o)}, r)$. We also have the following theorem.

Theorem (1.16)

If (i) $\{\underline{x}^{(k)}\}$ is a sequence defined by (1.66) in which $\underline{P}$: $R^n \to R^n$ is a continuous mapping;

(ii) $\underline{x}^{(k)} \in B(\underline{x}^{(o)}, r)$ $(\forall k \geq 0)$;

(iii) $\{\underline{x}^{(k)}\}$ is convergent to $\underline{x}^*$ in $B(\underline{x}^{(o)}, r)$,

then, $\underline{x}^*$ is a fixed point of $\underline{P}$ in R^n. $\square$

The question of existence, and uniqueness of a fixed point of a

contraction mapping and of the convergence rate of the sequence $\{\underline{x}^{(k)}\}$ generated by (1.66) are answered in the following theorem.

Theorem (1.17)

Let $\underline{P}$: $R^n \to R^n$, be a contraction mapping on $\bar{B}(\underline{x}^{(o)}, r)$ where,

$$r \geq \frac{1}{(1-\alpha)} \| \underline{x}^{(o)} - \underline{P}(\underline{x}^{(o)}) \| = r_o ,$$

and α is the contraction factor. Then

(i) $\underline{P}$ has a fixed point $\underline{x}^*$ in $\bar{B}(\underline{x}^{(o)}, r)$;

(ii) $\underline{x}^*$ is unique;

(iii) $\{\underline{x}^{(k)}\}$ converges to $\underline{x}^*$;

(iv) $\| \underline{x}^{(k)} - \underline{x}^* \| \leq \alpha^k r$ ($k \geq o$). (1.68)

□

For proofs of T(1.15), T(1.16) and T(1.17), see Rall (1969).

Let

$$\underline{F}(\underline{x}) = \underline{0} , \tag{1.69}$$

be a system of n nonlinear equations, where $\underline{F}$: $R^n \to R^n$ is twice F-differentiable on R^n.

Definition (1.15)

If (i) $\underline{x}^{(o)}$ is a given point in R^n;

(ii) $\{\underline{x}^{(k)}\}$ is a sequence in R^n, given by

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - [\underline{F}'(\underline{x}^{(k)})]^{-1} \underline{F}(\underline{x}^{(k)}) \quad (k = 0, 1, 2, \ldots), \tag{1.70}$$

then $\{\underline{x}^{(k)}\}$ is called the Newton's sequence corresponding to the point $\underline{x}^{(o)}$ and the mapping $\underline{F}$. □

If $\underline{F}'(\underline{x}^{(k)})$ is singular for some k, and $\underline{F}(\underline{x}^{(k)}) \neq \underline{0}$, then we call the finite sequence $\{\underline{x}^{(o)}, \ldots, \underline{x}^{(k)}\}$ obtained in this case <u>divergent</u> owing to inversion failure at the kth step. On the other hand, if $\underline{F}(\underline{x}^{(k)}) = \underline{0}$

(whether or not $\underline{F}'(\underline{x}^{(k)})$ is invertible) and $\underline{F}(\underline{x}^{(m)}) \neq \underline{0}$ for any $m < k$, then $\underline{x}^* = \underline{x}^{(k)}$ is a solution of (1.69), and the sequence is said to converge to $\underline{x}^*$ at the kth step.

If $\underline{F}$ and $\underline{F}'$ are defined on the entire space R^n, then $\{\underline{x}^{(k)}\}$ will be an infinite sequence if neither of the two possibilities discussed above is met. Assume that the infinite Newton sequence converges to $\underline{x}^*$; that is

$$\underline{x}^* = \lim_{k\to\infty} \underline{x}^{(k)}. \tag{1.71}$$

The three following theorems proved in Rall (1969) give the sufficient conditions for $\underline{x}^*$ to be a solution of (1.69).

Theorem (1.18)

If $\underline{F}'$ is continuous at $\underline{x} = \underline{x}^*$, then

$$\underline{F}(\underline{x}^*) = 0. \quad \square$$

Theorem (1.19)

If

$$\|\underline{F}'(\underline{x})\| < C,$$

in some closed ball which contains $\{\underline{x}^{(k)}\}$, and c is finite number, then $\underline{x}^*$ is a solution of (1.69). $\square$

Theorem (1.20)

If

$$\|\underline{F}''(\underline{x})\| < K, \qquad (K \text{ is finite number}).$$

in some closed ball $\bar{B}(\underline{x}^{(o)}, r)$, $r > o$, which contains $\{\underline{x}^{(k)}\}$, then $\underline{x}^*$ is a solution of (1.69). $\square$

Under the hypotheses of the foregoing theorems the convergence of the

Newton sequence $\{\underline{x}^{(k)}\}$ implies the existence of a solution $\underline{x} = \underline{x}^*$ of (1.69). Thus if the existence and convergence of the Newton sequence can be established, then we can assure ourselves that (1.69) has a solution.

The following theorem by Kantorovič gives not only sufficient conditions for the existence of $\underline{x}^*$ but also information concerning the regions of existence and uniqueness of $\underline{x}^*$ and an error bound for the term $\underline{x}^{(k)}$ of the Newton sequence as an approximation to $\underline{x}^*$.

Let $\underline{x}^{(o)}$ be a given point, and assume that $[\underline{F}'(\underline{x}^{(o)})]^{-1}$ exists. Hence

$$\underline{x}^{(1)} = \underline{x}^{(o)} - [\underline{F}'(\underline{x}^{(o)})]^{-1} \underline{F}(\underline{x}^{(o)}). \tag{1.74}$$

Let U_o, η_o be two constants such that

$$\| [\underline{F}'(\underline{x}^{(o)})]^{-1} \| \leqslant U_o , \tag{1.75}$$

$$\| \underline{x}^{(1)} - \underline{x}^{(o)} \| \leqslant \eta_o . \tag{1.76}$$

Theorem (1.21)

If (i) $\| \underline{F}''(\underline{x}) \| \leqslant K \quad (\forall\, \underline{x} \in \bar{B}(\underline{x}^{(o)}, r))$; (1.77)

(ii) $h_o = U_o \eta_o K < \frac{1}{2}$; (1.78)

(iii) $r \geqslant r_o = \dfrac{1 - \sqrt{1 - 2h_o}}{h_o} \eta_o$; (1.79)

then, the Newton sequence (1.70) starting from $\underline{x}^{(o)}$, will converge to a solution of (1.69) which exists in $\bar{B}(\underline{x}^{(o)}, r)$. □

Theorem (1.22)

If the conditions of T(1.21) are satisfied, then

$$\| \underline{x}^* - \underline{x}^{(k)} \| \leqslant \frac{1}{2^{k-1}} (2h_o)^{2^k - 1} \eta_o \qquad (k \geqslant 0) . \tag{1.80}$$

□

Theorem (1.23)

If (i) $h_o < \frac{1}{2}$; (1.81)

(ii) $\tilde{r} = \dfrac{1 + \sqrt{1 - 2h_o}}{h_o}\,\eta_o$; (1.82)

(iii) $\| \underline{F}''(\underline{x}) \| \leq K \quad (\forall\, \underline{x} \in B(\underline{x}^{(o)}, \tilde{r}))$, (1.83)

then (1.69) has a solution $\underline{x}^*$ which is unique in $B(\underline{x}^{(o)}, \tilde{r})$. □

Theorem (1.24)

If the hypotheses of T(1.21) are satisfied and $h_o = \frac{1}{2}$, then there exists a unique solution $\underline{x}^*$ of (1.69) in the closed ball $\bar{B}(\underline{x}^{(o)}, r^{(o)}) = \bar{B}(\underline{x}^{(o)}, 2\eta^{(o)})$. □

For proofs of T(1.21), T(1.22), T(1.23) and T(1.24), see Rall (1969).

Theorem (1.25)

If $\underline{L}:R^n \to R^n$ is a nonsingular linear mapping and $\underline{P}:R^n \to R^n$ satisfies

$$\| \underline{P}\underline{x} - \underline{P}\underline{y} \| \leq \alpha \| \underline{x} - \underline{y} \| \quad (\forall \underline{x}, \underline{y} \in R^n),$$

in which $\alpha < \| A^{-1} \|^{-1}$, then $\underline{F} = \underline{L} - \underline{P}$ is homeomorphism from R^n onto R^n. □

For proof see Ortega and Rheinboldt (1970), p.123.

Definition (1.16)

An algorithm for solving a system of n equations in n unknowns has the property of <u>linear termination</u> if it solves a linear system with a positive definite symmetric matrix within n steps. □

Let

$$A\underline{x} = \underline{b} \tag{1.84}$$

be a system of n linear equations in n unknowns, in which A is a positive definite symmetric matrix.

Theorem (1.26)

If (i) $\{\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}\}$ is a set of A-conjugate vectors;

(ii) $\underline{x}^* = A^{-1}\underline{b}$ is the solution of (1.84);

then

$$\underline{x}^* = \sum_{k=0}^{n-1} \frac{\underline{p}^{(k)T}\underline{b}}{\underline{p}^{(k)T}A\,\underline{p}^{(k)}}\,\underline{p}^{(k)} \cdot \square \tag{1.85}$$

Proof.

Since A is positive definite, the vectors $\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}$ are linearly independent, and span R^n. Hence

$$\underline{x}^* = \sum_{k=0}^{n-1} \alpha^{(k)}\,\underline{p}^{(k)}, \tag{1.86}$$

where $\alpha^{(o)},\ldots,\alpha^{(n-1)}$ are n constants to be determined. From (i), (ii) and (1.86),

$$\underline{p}^{(k)T}A\,\underline{x}^* = \underline{p}^{(k)T}\underline{b} = \alpha^{(k)}\,\underline{p}^{(k)}A\,\underline{p}^{(k)}. \tag{1.87}$$

Hence,

$$\alpha^{(k)} = \frac{\underline{p}^{(k)T}\underline{b}}{\underline{p}^{(k)T}A\,\underline{p}^{(k)}}. \tag{1.88}$$

So

$$\underline{x}^* = \sum_{k=0}^{n-1} \frac{\underline{p}^{(k)T}\underline{b}}{\underline{p}^{(k)T}A\,\underline{p}^{(k)}}\,\underline{p}^{(k)} \cdot \square \tag{1.89}$$

If $\underline{x}^* = A^{-1}\underline{b}$, then by (1.89)

$$\underline{x}^* = \sum_{k=0}^{n-1} \frac{\underline{p}^{(k)}\,\underline{p}^{(k)T}}{\underline{p}^{(k)T}A\,\underline{p}^{(k)}}\,\underline{b}. \tag{1.90}$$

This relation is true for any $\underline{b}$. From hypothesis (ii) in T(1.26) and (1.90), we have

$$A^{-1} = \sum_{k=1}^{n-1} \frac{\underline{p}^{(k)} \underline{p}^{(k)T}}{\underline{p}^{(k)T} A \underline{p}^{(k)}} \quad . \tag{1.91}$$

Let $\Delta \underline{x}^{(k)} = \alpha^{(k)} \underline{p}^{(k)}$. From (1.16), (1.91) we find

$$A^{-1} = \sum_{k=1}^{n-1} \frac{\Delta \underline{x}^{(k)} \Delta \underline{x}^{(k)T}}{\Delta \underline{x}^{(k)T} \Delta \underline{g}^{(k)}} \quad . \tag{1.92}$$

Theorem (1.27) (Gram-Schmidt orthogonalization procedure).

If $\{\underline{F}^{(o)}, \ldots, \underline{F}^{(n-1)}\}$ is a set of n linearly independent vectors, then a set $\{\underline{p}^{(o)}, \ldots, \underline{p}^{(n-1)}\}$ of A-conjugate vectors can be constructed by means of the following formulae.

$$\underline{p}^{(o)} = \underline{F}^{(o)} , \tag{1.93a}$$

$$\underline{p}^{(k+1)} = \underline{F}^{(k+1)} - \sum_{i=o}^{k} \frac{\underline{F}^{(k+1)T} A \underline{p}^{(i)}}{\underline{p}^{(k)T} A \underline{p}^{(k)}} \underline{p}^{(k)} \quad (o \leq k \leq n-2). \tag{1.93b}$$

□

Proof.

Let $\underline{p}^{(o)} = \underline{F}^{(o)}$, and choose $\underline{p}^{(1)}$ as a linear combination of $\underline{F}^{(1)}$ and $\underline{p}^{(o)}$; that is

$$\underline{p}^{(1)} = \underline{F}^{(1)} + \beta_o^{(1)} \underline{p}^{(o)} .$$

The vector $\underline{p}^{(1)}$ will be A-conjugate to $\underline{p}^{(o)}$ if

$$\underline{p}^{(1)T} A \underline{p}^{(o)} = \underline{F}^{(1)T} A \underline{p}^{(o)} + \beta_o^{(1)} \underline{p}^{(o)T} A \underline{p}^{(o)} = o .$$

Hence

$$\beta_o^{(1)} = - \frac{\underline{F}^{(1)T} A \underline{p}^{(o)}}{\underline{p}^{(o)T} A \underline{p}^{(o)}} \quad .$$

Having selected $\underline{p}^{(o)}, \ldots, \underline{p}^{(k)}$ we can choose $\underline{p}^{(k+1)}$ of the form

$$\underline{p}^{(k+1)} = \underline{F}^{(k+1)} + \sum_{i=o}^{k} \beta_i^{(k+1)} \underline{p}^{(i)} ,$$

which will conjugate to $\underline{p}^{(i)}$ $(i \leq k)$ if

$$\beta_i^{(k+1)} = - \frac{\underline{F}^{(k+1)T} A \underline{P}^{(i)}}{\underline{P}^{(i)T} A \underline{P}^{(i)}} \quad . \square \tag{1.94}$$

§(1.6) Some Methods for the Solution of Systems of Nonlinear Equations

In this section we give a brief description of some methods for solving a system of equations of the form

$$\underline{F}(\underline{x}) = \underline{0} \ , \tag{1.95}$$

where $\underline{F}: D \subset R^n \to R^n$ is given.

1. Contraction Mapping Method

In order to express (1.95) as a fixed point problem, define a mapping $\underline{P}: D \subset R^n \to R^n$ by

$$\underline{P}(\underline{x}) = \underline{x} + \underline{F}(\underline{x}) \quad (\underline{x} \in D). \tag{1.96}$$

It is clear that finding a fixed point of the mapping $\underline{P}$ defined by (1.96) is equivalent to solving (1.95). Another way of formulating (1.95) as a fixed point problem is to consider the mapping $\underline{G}: R^n \to R^n$ such that

$$\underline{G}(\underline{x}) = \underline{0} \iff \underline{x} = \underline{0} \ , \tag{1.97}$$

and to define $\underline{P}$ by

$$\underline{P}(\underline{x}) = \underline{x} + \underline{G}(\underline{F}(\underline{x})). \tag{1.98}$$

Having formulated (1.95) as a fixed point problem, the sequence $\{\underline{x}^{(k)}\}$ is generated from

$$\underline{x}^{(k+1)} = \underline{P}(\underline{x}^{(k)}) \qquad (k \geq 0), \tag{1.99}$$

where $\underline{x}^{(0)} \in R^n$ is given.

If the hypotheses of T(1.17) are satisfied, $\{\underline{x}^{(k)}\}$ converges to a fixed point $\underline{x}^*$ of $\underline{P}$ which is the solution of the system (1.95). The convergence of (1.99) depends on the initial estimate $\underline{x}^{(o)}$, and on the way in which the contraction mapping is defined. For further detail see Rall (1969).

Although evaluation of $\underline{F}$ only is required, the convergence rate of the contraction mapping method is linear by T(1.17). Also a sufficiently close initial estimate of the fixed point $\underline{x}^*$ is necessary for convergence.

2. Newton's Method

In Newton's method the system (1.95) is replaced by a linear system

$$\underline{L}(\underline{x}) = \underline{y}, \tag{1.100}$$

where (1.100) approximates (1.95) is some sense. If $\underline{F}$ is twice differentiable on D, then by the Taylor expansion

$$\underline{F}(\underline{x}) = \underline{F}(\underline{x}^{(k)}) + \underline{F}'(\underline{x}^{(k)})(\underline{x}-\underline{x}^{(k)}) + \underline{R}(\underline{x},(\underline{x}-\underline{x}^{(k)})), \tag{1.101}$$

where $\| \underline{R}(\underline{x},(\underline{x}-\underline{x}^{(k)})) \| = O(\| \underline{x}-\underline{x}^{(k)} \|^2)$. For $\underline{x} = \underline{x}^*$, (1.101) can be written in the form

$$\underline{F}(\underline{x}^{(k)}) + \underline{F}'(\underline{x}^{(k)})(\underline{x}^*-\underline{x}^{(k)}) = -\underline{R}(\underline{x}^*,(\underline{x}^*-\underline{x}^{(k)})). \tag{1.102}$$

If $\underline{x}^{(o)}$ is sufficiently close to $\underline{x}^*$, it is reasonable to assume that the quantity on the right hand side can be neglected. Hence an approximation $\underline{x}^{(k+1)}$ to $\underline{x}^*$ is given by

$$\underline{F}(\underline{x}^{(k)}) + \underline{F}'(\underline{x}^{(k)})(\underline{x}^{(k+1)}-\underline{x}^{(k)}) = \underline{0}, \tag{1.103}$$

where

$$\underline{F}'(\underline{x}^{(k)}) = (\partial_j F_i(\underline{x}^{(k)})), (i,j=1,\ldots,n).$$

If $\underline{F}'(\underline{x}^{(k)})$ is nonsingular equation (1.100) has a unique solution $\underline{x}^{(k+1)}$.

By equation (1.102) this solution is expected to be a good approximation to $\underline{x}^*$ with

$$\underline{F}(\underline{x}^{(k+1)}) \approx \underline{R}(\underline{x}^{(k+1)}, (\underline{x}^{(k+1)} - \underline{x}^{(k)})). \tag{1.104}$$

Clearly, in this case,

$$\underline{L} = \underline{F}'(\underline{x}^{(k)}), \tag{1.105}$$

$$\underline{y} = \underline{x}^{(k)} \underline{F}'(\underline{x}^{(k)}) - \underline{F}(\underline{x}^{(k)}). \tag{1.106}$$

If $\underline{F}'(\underline{x}^{(k)})$ is nonsingular $(\forall k \geq 0)$, then Newton's iteration is given by

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - [\underline{F}'(\underline{x}^{(k)})]^{-1} \underline{F}(\underline{x}^{(k)}) \quad (k = 0, 1, \ldots). \tag{1.107}$$

Another form of (1.107) is

$$\underline{F}'(\underline{x}^{(k)}) \Delta \underline{x}^{(k)} = -\underline{F}(\underline{x}^{(k)}) \quad (k = 0, 1, \ldots). \tag{1.108}$$

Thus we must solve n linear equations at each iteration.

Newton's method enjoys second order convergence in some neighbourhood of the minimum. This advantage makes Newton's method a focal point in the study of iteration methods for nonlinear equations. On the other hand, the necessity of computing the inverse Jacobian at each iteration makes it computationally laborious.

3. Parameter Variation Methods

The content of T(1.25) is that the linear homeomorphism $\underline{L}$ may be perturbed by a nonlinear mapping $\underline{P}$ provided that $\underline{P}$ is suitably small.

Let $\underline{F}: D \subset R^n \to R^m$ be given and suppose that the equation

$$\underline{F}(\underline{x}) = \underline{\lambda} \quad (\underline{\lambda} \in R^m) \tag{1.109}$$

has a solution $\underline{x}^*$. Then, three questions arise immediately. Firstly,

is $\underline{x}^*$ an isolated solution, that is, is there a neighbourhood of $\underline{x}^*$ which contains no other solutions of (1.109)? This question is answered affirmatively if $\underline{F}$ is one-to-one in a neighbourhood of $\underline{x}^*$. Secondly, if $\underline{\lambda}$ is changed by a small amount, will (1.109) still have a solution? This is indeed the case if $\underline{F}(D)$ contains an open neighbourhood of $\underline{\lambda}$. Finally, does the solution of the perturbed equation vary continuously with $\underline{\lambda}$? This is true also, if $\underline{F}$ is a local homeomorphism.

The more general situation corresponds to the case in which $\underline{F}$ is a function of two vector variables and the equation $\underline{F}(\underline{x},\underline{\lambda}) = \underline{0}$ $(\underline{\lambda} \in R^m)$ is known to have a solution $\underline{x}^{(o)}$ for $\underline{\lambda} = \underline{\lambda}^{(o)}$. Then we may ask whether this equation also has solutions when $\underline{\lambda} = \underline{\lambda}^{(1)}$ is close to $\underline{\lambda}^{(o)}$, and, if so, what is the behaviour of the corresponding $\underline{x}$ as a function of $\underline{\lambda}$? These questions are considered in Ortega and Rheinboldt (1970), chapter 5.

Davidenko (1953) was the first to suggest that if the solution of the equation

$$\underline{F}(\underline{x},\underline{\lambda}) = \underline{0}\,, \quad (\underline{\lambda} \in R^m) \tag{1.110}$$

is known for $\underline{\lambda} = \underline{\lambda}^{(o)}$ and is required for $\underline{\lambda} = \underline{\lambda}^{(1)}$, then moving $\underline{\lambda}$ incrementally from $\underline{\lambda}^{(o)}$ to $\underline{\lambda}^{(1)}$, produces a sequence of intermediate problems, each with the virtue that it has a good initial estimate of its solution, namely the solution of the preceding problem. Differentiating (1.110) with respect to λ_i we find

$$J(\underline{x},\underline{\lambda})\,{}_{\lambda}\partial_i \underline{x} + {}_{\lambda}\partial_i \underline{F}(\underline{x},\underline{\lambda}) = 0 \quad (i = 1,\ldots,m), \tag{1.111}$$

where ${}_{\lambda}\partial_i \underline{x}$, ${}_{\lambda}\partial_i \underline{F}$ are the partial derivatives of $\underline{x}$, $\underline{F}$ respectively with respect to λ_i, and where $J(\underline{x},\underline{\lambda})$ is the (n x n) matrix $({}_{x}\partial_{\ell}\, F_j(\underline{x},\underline{\lambda}))$, $(j,\ell = 1,\ldots,n)$. Solving (1.111) for ${}_{\lambda}\partial_i \underline{x}$ we obtain

$$ {}_{\lambda}\partial_i \underline{x} = -J^{-1}(\underline{x},\underline{\lambda})\,{}_{\lambda}\partial_i \underline{F}(\underline{x},\underline{\lambda}) \quad (i = 1,\ldots,m), \tag{1.112}$$

which represent n.m first order differential equations. (1.112) can be written in the more compact form

$$\underline{x}'_\lambda = -J^{-1}\underline{F}'_\lambda \,, \tag{1.113}$$

where

$$\underline{x}'_\lambda = ({}_\lambda\partial_1 \underline{x}, {}_\lambda\partial_2 \underline{x}, \cdots, {}_\lambda\partial_m \underline{x})' \,, \tag{1.114}$$

and

$$\underline{F}'_\lambda = ({}_\lambda\partial_1 \underline{F}, {}_\lambda\partial_2 \underline{F}, \cdots, {}_\lambda\partial_m \underline{F}). \tag{1.115}$$

When implementing the method on the computer, we require algorithms which will obtain the zeros of the intermediate problems. The first approach is suggested by the operator equation (1.113). The Jacobian matrix must be available. Another distinct approach utilizes any of the available iteration methods for the solution of the intermediate problems. Freudenstien and Roth (1963), and Diest and Sefor (1967) employed Newton's method in order to solve the sequence of intermediate problems; Broyden (1969) used quasi-Newton methods (namely Broyden's (1965) method) for the same purpose. If Newton's method is employed, then iteration k of the intermediate problem i can be written in the following way

$$\Delta \underline{x}^{(i,k)} = -J^{(i,k)-1}\underline{F}(\underline{x}^{(i,k)}, \underline{\lambda}^{(i)}). \tag{1.116}$$

Broyden (1969) suggested a practical way to implement Davidenko's idea by constructing an auxiliary function $\underline{G}(\underline{x},\lambda)$ with $\lambda \in R^1$, such that

$$\underline{G}(\underline{x}, 0) = \underline{F}(\underline{x}), \tag{1.117a}$$

and

$$\underline{G}(\underline{x}, 1) = \underline{0} \tag{1.117b}$$

has an obvious solution.

If $\underline{x}^{(o)}$ is an initial estimate of a solution of (1.95), then

$$\underline{G}(\underline{x},\lambda) \stackrel{D}{=} \underline{F}(\underline{x}) - \lambda \underline{F}(\underline{x}^{(o)}), \tag{1.118}$$

satisfies (1.117). If $\underline{x}$ is a solution of

$$\underline{G}(\underline{x},\lambda) = \underline{0}, \tag{1.119}$$

then $\underline{x}$ is a function of λ. We refer to the curve in R^n with parametric equation $\underline{x} = \underline{x}(\lambda)$ as a Davidenko path. Let $\underline{G}(\underline{x},\lambda)$ be defined by (1.118). If we differentiate (1.119) with respect to λ we obtain

$$\underline{x}'_\lambda = J^{-1}(\underline{x}^{(o)})\underline{F}(\underline{x}^{(o)}). \tag{1.120}$$

From (1.118), (1.119), (1.120) we have

$$\underline{x}'_\lambda = \frac{1}{\lambda} J^{-1}(\underline{x})\underline{F}(\underline{x}). \tag{1.121}$$

The matrix J in (1.121) is purely the Jacobian matrix of $\underline{F}(\underline{x})$. Thus the direction of the Davidenko path in this case is the direction of the corresponding Newton step. The Davidenko path exists if the inverse Jacobian J^{-1} exists at all points of the path; that is if $\underline{F}$ is a local homeomorphism at each point of the path.

Meyer (1967) gave another form of an auxiliary function in which

$$\underline{G}(\underline{x},\lambda) = \lambda(\underline{x} - \underline{x}^{(o)}) + (1-\lambda)\underline{F}(\underline{x}), \tag{1.122}$$

where $\underline{x}^{(o)}$ is some estimate of the root of (1.95). The Davidenko path in this case is given by

$$\underline{x}'_\lambda = \frac{1}{\lambda}\left[(1-\lambda)J(\underline{x},\lambda) + \lambda I\right]^{-1}\underline{F}(\underline{x}), \tag{1.123}$$

where I is the (n x n) unit matrix. The direction of this path varies

between the steepest descent (if $\underline{F}$ were a gradient mapping) at $\lambda = 1$ to the Newton direction as λ approaches zero. Since steepest descent works well far from the solution, it might be conjectured that (1.122) would produce better results than (1.118). No numerical evidence is available, however.

4. Conjugate Direction Methods

Consider the system of n linear equations in n unknowns

$$\underline{F}(\underline{x}) \stackrel{D}{=} \underline{b} - A\underline{x} = \underline{0} \ , \tag{1.124}$$

where A is positive definite symmetric (n x n) matrix. Let $\underline{F}^{(k)}$ denote the residual vector $\underline{b} - A\underline{x}^{(k)}$. If n A-conjugate vectors $\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}$ are available, then by T(1.26) the solution of (1.129) is given by (1.85). The conjugate direction method of Hestenes and Stiefel (1952) for solving the linear system (1.124) is based on forming a set of residual vectors $\{\underline{F}^{(k)}\}$ and a set of direction vectors $\{\underline{p}^{(k)}\}$ in the order $\underline{F}^{(o)}$, $\underline{p}^{(o)}$, $\underline{F}^{(1)}$, $\underline{p}^{(1)}$, $\underline{F}^{(2)},\ldots,\underline{F}^{(n-1)}$, $\underline{p}^{(n-1)}$. The set of residuals $\{\underline{F}^{(k)}\}$ is formed by applying an orthogonalization procedure to the vectors $\underline{F}^{(o)}$, $A\underline{p}^{(o)}$, $A\underline{p}^{(1)},\ldots,A\underline{p}^{(n-2)}$, while the set of directions $\{\underline{p}^{(k)}\}$ is formed by producing a set of A-conjugate vectors from $\underline{F}^{(o)}$, $\underline{F}^{(1)},\ldots,\underline{F}^{(n-1)}$. Now, by T(1.26) the algorithm which produces the set $\{\underline{p}^{(o)},\ldots,\underline{p}^{(n-1)}\}$ of conjugate directions solve (1.124) in n steps at most.

Algorithm (1.4)

Given an initial estimate $\underline{x}^{(o)}$ of the solution $\underline{x}^*$ of (1.124), and a parameter $\varepsilon > o$

1. Set $k = o$
2. Set $\underline{p}^{(o)} = \underline{F}^{(o)} = \underline{b} - A\underline{x}^{(o)}$. (1.125)
3. Compute $\alpha^{(k)}$ from

$$\alpha^{(k)} = \frac{\underline{F}^{(k)T}\underline{p}^{(k)}}{\underline{p}^{(k)T}A\,\underline{p}^{(k)}} \ . \tag{1.126}$$

4. Set $\underline{x}^{(k+1)} = \underline{x}^{(k)} + \alpha^{(k)}\ \underline{p}^{(k)}$. (1.127)

5. Compute $\underline{F}^{(k+1)}$ from

$$\underline{F}^{(k+1)} = \underline{F}^{(k)} - \alpha^{(k)} A \underline{P}^{(k)}. \tag{1.128}$$

6. Compute $\beta^{(k)}$ from

$$\beta^{(k)} = -\frac{\underline{F}^{(k+1)T} A \underline{P}^{(k)}}{\underline{P}^{(k)T} A \underline{P}^{(k)}}. \tag{1.129}$$

7. Set $\underline{p}^{(k+1)} = \underline{F}^{(k+1)} + \beta^{(k)}\ \underline{p}^{(k)}$. (1.130)

8. If $k = n-1$ go to 11.

9. If $\underline{F}^{(k)T}\underline{F}^{(k)} \leqslant \varepsilon$ go to 11.

10. Set $k = k+1$ and go to 2.

11. Stop. □

The property of linear termination of this algorithm is established by T(1.26) and T(1.27).

If A is an arbitrary nonsingular (n x n) matrix, then A^TA is symmetric positive definite. Hence the system $A\underline{x} = \underline{b}$ may be converted into a system with a symmetric positive definite matrix by premultiplying both sides by A^T yielding

$$A^TA\,\underline{x} = A^T\underline{b}. \tag{1.131}$$

Now A(1.4) is applicable to (1.131).

CHAPTER 2

QUASI-NEWTON METHODS

§(2.1) Objections to Newton's Method

Newton's iteration for unconstrained minimization is given by

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \alpha^{(k)} A^{(k)-1} \underline{g}^{(k)} \quad (k = 0, 1, 2, \cdots), \tag{2.1}$$

where $A^{(k)}$ is the Hessian matrix of the objective function f evaluated at $\underline{x}^{(k)}$, $\underline{g}^{(k)}$ is the gradient of f at $\underline{x}^{(k)}$, and $\alpha^{(k)}$ is a scalar taken to be unity in the simplest form of Newton's method. Provided that the first and second derivatives can be evaluated for any $\underline{x}$, and $A^{(k)}$ never becomes singular, then Newton's method can be applied in principle. The discussion which follows in this section may be applied with little change to Newton's method for solving a set of nonlinear equations $\underline{F}(\underline{x}) = \underline{0}$ where $\underline{F}: R^n \to R^n$ is given. In this case the Jacobian matrix $J^{(k)}$ of $\underline{F}$ replaces $A^{(k)}$, and the value of the residual vector $\underline{F}^{(k)}$ replaces the gradient value $\underline{g}^{(k)}$. Generally the Jacobian matrix $J^{(k)}$ is not symmetric while the Hessian $A^{(k)}$ is symmetric.

Newton's method, for unconstrained minimization, if it works, does so very well, and if a good initial estimate of the minimum is available no line search is needed and α is kept equal to unity.

However, it has some disadvantages. We mention here the most serious of them.

(i) The method in its simplest form with $\alpha^{(k)} = 1$ $(k \geq o)$ often fails to converge from a poor estimate to the minimum. To prevent divergence $\alpha^{(k)}$ is chosen such that

$$f(\underline{x}^{(k+1)}) < f(\underline{x}^{(k)}) \qquad (k \geq o), \tag{2.2}$$

so that the algorithm has the descent property. To achieve this aim a line search is needed. If $\alpha^{(k)}$ is chosen to minimize $f(\underline{x}^{(k)} + \alpha \underline{p}^{(k)})$

along the direction $p^{(k)} \stackrel{D}{=} -A^{(k)^{-1}} g^{(k)}$, then an exact line search is required. Iteration (2.1) with a varied α such that (2.2) is satisfied was considered formally for the first time by Crockett and Chernoff (1955).

(ii) Newton's method fails if the Hessian at any stage becomes singular. For some problems where a good initial estimate of the minimum is not available it is found that steepest descent is an effective way of getting reasonably close to the minimum. Hence an algorithm based on the method of steepest descent followed by Newton's method may do well. Levenberg (1944) suggested the iteration

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - [A^{(k)} + \lambda I]^{-1} \underline{g}^{(k)}, \tag{2.3}$$

where $\lambda > 0$ is a scalar and I is $(n \times n)$ unit matrix. If λ is sufficiently large the algorithm based on (2.3) has the descent property (2.2). Another feature of this algorithm is that if $\lambda > 0$ the inverse of $A^{(k)} + \lambda I$ always exists (Marqaurdt (1963)). If λ is decreased systematically from a large value to zero, then algorithm (2.3) varies between steepest descent and Newton's method combining the features of both methods. However, Newton's method is not the only method which breaks down when the Hessian is singular, for all quasi-Newton methods have this disadvantage when the approximated inverse Hessian becomes singular at any step (Broyden (1972)). Another unpleasant feature of Newton's method is that iteration (2.1) even after the introduction of the parameter α to satisfy (2.2) does not ensure convergence. Such a situation may occur when $p^{(k)} \stackrel{D}{=} -A^{(k)^{-1}} g^{(k)}$ is perpendicular to $g^{(k)}$; thus no advance from $x^{(k)}$ can be made which causes the algorithm to break down even when the Hessian is nonsingular and $g^{(k)}$ is nonzero; this disadvantage is shared by quasi-Newton methods. Actually any algorithm is required to satisfy the nonorthogonality condition

$$\underline{p}^{(k)T} \underline{g}^{(k)} \neq 0 \quad (k = 0, 1, \cdots). \tag{2.4}$$

(iii) The most objectionable feature of Newton's method is the necessity of evaluating the Hessian matrix $A(\underline{x})$ at each iteration, because the objective function $f(\underline{x})$ in many cases is very complicated. In such cases A may be approximated by a matrix $A(\underline{x}^{(k)}, h^{(k)})$, where the elements of $A(\underline{x}^{(k)}, h^{(k)})$ are given by the formulae

$$\partial_i \partial_j f(\underline{x}^{(k)}) \approx (\underline{g}(\underline{x}^{(k)} - h_{ij}^{(k)} \underline{e}^{(j)}) - \underline{g}(\underline{x}^{(k)}))^T \underline{e}^{(i)} / h_{ij}^{(k)} \quad (i,j=1,\ldots,n), \tag{2.5}$$

where h_{ij} are given discretization parameters and $\underline{e}^{(i)}$, $\underline{e}^{(j)}$ are the ith and jth coordinate unit vectors. The problem now is how to choose $h_{ij}^{(k)}$. Let $\Delta_{ij}(\underline{x},h)$ denote any difference approximation of $\partial_i \partial_j f(\underline{x})$ such that

$$\lim_{h \to 0} \Delta_{ij}(\underline{x},h) = \partial_i \partial_j f(\underline{x}) \quad (i,j=1,\ldots,n). \tag{2.6}$$

If $A(\underline{x},h) = (\Delta_{ij}(\underline{x},h))$ we have the discretized Newton iteration

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - A^{-1}(\underline{x}^{(k)}, h^{(k)})\, \underline{g}^{(k)} \quad (k=0,1,\ldots). \tag{2.7}$$

If $h^{(k)}$ is chosen to be equal to a constant h, where h is a suitable small number, it can be proved in this case that iteration (2.7) in general has a linear rate of convergence. In order to retain the second order convergence of Newton's method it is necessary that $\lim_{k\to\infty} h^{(k)} = 0$. (Ortega and Rheinboldt (1970), Chapter 11).

In practice $h^{(k)}$ cannot tend to zero. Thus, the second order convergence of Newton's method is lost. Usually $h^{(k)}$ is taken to be equal to a small constant h. However this process of approximation is an expensive use of machine time, since in order to compute $A(\underline{x},h)$, it is required to apply (2.5) $\frac{n}{2}(n+1)$ times. When solving a system of nonlinear equations where the Jacobian is to be approximated, then n evaluations of the vector function $\underline{F}$ are required in addition to its value at $\underline{x}^{(k)}$.

Another way of overcoming the difficulty of computing the Hessian A at each iteration is to compute A every m iterations. The iteration in this case is given by

$$\underline{x}^{(k,0)} = \underline{x}^{(k)}, \tag{2.8a}$$

$$\underline{x}^{(k,i)} = \underline{x}^{(k,i-1)} - A^{-1}(\underline{x}^{(k)})\underline{g}(\underline{x}^{(k,i-1)}), \tag{2.8b}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k,m)}, \tag{2.8c}$$

$$(i = 1, \ldots, m), (k = 1, 2, \cdots).$$

This iteration represents a simple way of generating higher order methods. But even with the best choice of m, this iteration still remains inferior to quasi-Newton iteration (Broyden (1972)).

(iv) The last disadvantage of Newton's method is the necessity of inverting A or equivalently solving a set of n linear equations at each iteration. Although this is not a serious problem, it requires more time and additional programming.

We will see in this chapter that quasi-Newton methods are an efficient way of overcoming the third and fourth disadvantages of Newton's method.

§(2.2) The Nature of Quasi-Newton Methods

We shall consider a class of methods for unconstrained minimization in which some of the objectionable features of Newton's method have been overcome. This class of method is called the class of quasi-Newton methods, for they are constructed so as to simulate the desirable properties of Newton's method.

Suppose that at some stage k, an approximation $H^{(k)}$ to $A^{-1}(\underline{x}^{(k)})$ has been obtained, and that $\underline{x}^{(k+1)}$ has been computed from

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)}. \tag{2.9}$$

By T(1.2), if $f: R^n \to R^1$ has continuous first and second partial derivatives in R^n, $\underline{g}(\underline{x}^*) = 0$, and $A(\underline{x}^*)$ is positive definite then $\underline{x}^*$ is strong local minimum of f. Also since $A(\underline{x})$ is continuous at $\underline{x}^*$ there is a convex neighbourhood S of $\underline{x}^*$ in which $A(\underline{x})$ is positive definite. If $\underline{x}^{(k)} \in S$, then $-A^{-1}(\underline{x}^{(k)})\underline{g}^{(k)}$ is downhill for f at $\underline{x}^{(k)}$, for since $A^{-1}(\underline{x}^{(k)})$ is positive definite,

$$-\underline{g}^{(k)T} A^{-1}(\underline{x}^{(k)})\underline{g}^{(k)} < 0. \tag{2.10}$$

Hence we may determine $\alpha^{(k)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)}) = \min_{\alpha} f(\underline{x}^{(k)} - \alpha H^{(k)} \underline{g}^{(k)}) \tag{2.11}$$

if $H^{(k)}$ is sufficiently good approximation to $A^{-1}(\underline{x}^{(k)})$.

If, however, $H^{(k)}$ is any positive definite (n x n) matrix, then $-H^{(k)}\underline{g}^{(k)}$ is downhill for f at $\underline{x}^{(k)}$ since $\underline{g}^{(k)T} H^{(k)} \underline{g}^{(k)} > 0$, and we may still determine $\alpha^{(k)}$ from (2.11).

We next consider how to obtain a sequence $\{H^{(k)}\}$ of matrices which in some sense simulate the behaviour of $\{A^{-1}(\underline{x}^{(k)})\}$.

By the Taylor theorem if f is sufficiently differentiable then

$$\underline{g}(\underline{x}^{(k)}) = \underline{g}(\underline{x}^{(k+1)}) - A(\underline{x}^{(k+1)})\Delta\underline{x}^{(k)} + O(\|\Delta\underline{x}^{(k)}\|_E^2). \tag{2.12}$$

Neglecting the term $O(\|\Delta\underline{x}^{(k)}\|_E^2)$, we obtain

$$\Delta\underline{g}^{(k)} \approx A^{(k+1)}\Delta\underline{x}^{(k)}. \tag{2.13}$$

Hence we shall require that the matrix $H^{(k+1)}$ which simulates $A^{-1}(\underline{x}^{(k+1)})$ satisfies the quasi-Newton formula

$$H^{(k+1)}\Delta\underline{g}^{(k)} = \Delta\underline{x}^{(k)}. \tag{2.14}$$

Now (2.14) does not define $H^{(k+1)}$ uniquely and there are many ways in which $H^{(k+1)}$ can be chosen so as to satisfy (2.14). For this reason (2.9), (2.11), (2.14) define a class of methods for minimizing f rather than a single method.

Clearly if $H^{(k)}$ is to simulate the behaviour of $A^{-1}(\underline{x}^{(k)})$ very well, it is advantageous that $H^{(k+1)}$ retains as far as possible the desirable properties of $H^{(k)}$ such as positive definiteness. This suggests that $H^{(k+1)}$ be formed by adding a suitable correction $C^{(k)}$ to $H^{(k)}$ such that

$$H^{(k+1)} = H^{(k)} + C^{(k)}, \tag{2.15}$$

where $C^{(k)}$ in general is of rank 1 or 2, and $H^{(k+1)}$ satisfies (2.14).

From the preceding considerations we obtain the following algorithm.

Algorithm (2.1) (General quasi-Newton method)

Given an initial estimate $\underline{x}^{(o)}$ of a minimum $\underline{x}^*$ of f and an initial estimate $H^{(o)}$ of the inverse Hessian $A^{-1}(\underline{x}^*)$.

1. Set $k = o$.
2. Compute $\underline{g}^{(o)}$ from
$$\underline{g}^{(o)} = \underline{g}(\underline{x}^{(o)}).$$
3. Compute $\alpha^{(k)}$, $\underline{x}^{(k+1)}$ from
$$f(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)}) = \min_{\alpha} f(\underline{x}^{(k)} - \alpha H^{(k)} \underline{g}^{(k)}),$$
$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)}.$$
4. Compute $\underline{g}^{(k+1)}$, $\Delta\underline{x}^{(k)}$, $\Delta\underline{g}^{(k)}$ from
$$\underline{g}^{(k+1)} = \underline{g}(\underline{x}^{(k+1)}),$$
$$\Delta\underline{x}^{(k)} = \underline{x}^{(k+1)} - \underline{x}^{(k)},$$
$$\Delta\underline{g}^{(k)} = \underline{g}^{(k+1)} - \underline{g}^{(k)}.$$
5. Compute $H^{(k+1)}$ from
$$H^{(k+1)} = H^{(k)} + C^{(k)}$$

6. Set $k = k+1$ and go to 3.

□

Now, if $\underline{F}$: $R^n \to R^n$ is the gradient of f: $R^n \to R^1$, then the previous discussion can be applied with no change to solve the system of nonlinear equations $\underline{F}(\underline{x}) = \underline{0}$, provided that we replace in the previous discussion the quantities A, $\underline{g}$, $\Delta\underline{g}$ by J, $\underline{F}$, $\Delta\underline{F}$ in order, where J represents the Jacobian matrix of $\underline{F}$. When $\underline{F}$ is not a gradient mapping then $\alpha^{(k)}$ cannot be given any more by (2.11). Instead $\alpha^{(k)}$ is chosen such that it minimizes some norm $\|\ .\ \|$ of $\underline{F}(\underline{x}^{(k+1)})$; that is,

$$\|\underline{F}(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{F}^{(k)})\| = \min_{\alpha} \|\underline{F}(\underline{x}^{(k)} - \alpha H^{(k)} \underline{F}^{(k)})\| . \qquad (2.16)$$

Alternatively $\alpha^{(k)}$ may be chosen to reduce the norm of $\underline{F}$; that is

$$\|\underline{F}(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{F}^{(k)})\| < \|\underline{F}(\underline{x}^{(k)})\| . \qquad (2.17)$$

Corresponding to each method for computing $C^{(k)}$ in A(2.1) there is a quasi-Newton method. If we wish that $H^{(k+1)}$ be symmetric positive definite when $H^{(k)}$ is, then clearly $C^{(k)}$ must be symmetric positive definite. The symmetry of the current matrix $H^{(k)}$ is a desirable feature for function minimization algorithms, but imposing such a property on algorithms for the solution of systems of nonlinear equations may give bad results since the Jacobian usually is unsymmetric. The positive (negative) definiteness of the current matrix $H^{(k)}$ is a desirable feature for function minimization, since it ensures that the search direction $\underline{p}^{(k)}$ is downhill (uphill) for f, and thus the direction of the step is already known to be along $+\underline{p}^{(k)}$ $(-\underline{p}^{(k)})$. Hence $\alpha^{(k)}$ is always positive (negative) (Fletcher(1970)). However, $H^{(k)}$ can be nondefinite. Hence the step $\Delta\underline{x}^{(k)}$ is taken along $+\underline{p}^{(k)}$ or $-\underline{p}^{(k)}$ in such a way that $\|\ \underline{F}(\underline{x})\ \|$ is reduced always. It follows that $\alpha^{(k)}$ can have different signs for different steps.

In addition to (2.14), another condition should be imposed on the

matrix $H^{(k+1)}$ which is constructed by (2.15), in order to determine $C^{(k)}$ more precisely. The most used conditions to specify $C^{(k)}$ are:

1. $C^{(k)}$ to be a rank 1 matrix (Davidon (1968), Murtagh and Sargent (1969), Barnes (1965), Broyden (1965), Rosen (1966), Zeleznik (1968));
2. $C^{(k)}$ to be rank 2 matrix (Davidon (1959), Fletcher and Powell (1963), Broyden (1967), Pearson (1969), Fletcher (1970));
3. $C^{(k)}$ to be a matrix of minimum norm (Greenstadt (1970), Goldfarb (1970)).

The correction $C^{(k)}$ derived by imposing condition 3 is of rank 3, as will be seen from the general formula derived to satisfy condition 3, but most of the practical algorithms obtained from that formula are of rank 1 or 2. In all quasi-Newton methods of practical interest for function minimization (solving systems of nonlinear equations), C_k depends upon $H^{(k)}$, $\Delta\underline{x}^{(k)}$, $\Delta\underline{g}^{(k)}$ ($\Delta\underline{F}^{(k)}$).

By means of A(2.1), objections (iii) and (iv) to Newton's method have been overcome.

§(2.3) Quasi-Newton Methods for Unconstrained Minimization

In this section we consider a unified approach to the problem of constructing algorithms which satisfy the following requirements:

1. Only a one dimensional search is used; this is necessary to avoid using a multidimensional search.
2. The algorithms are quadratically convergent; this usually ensures superlinear convergence near a minimum of a nonquadratic objective function.
3. The algorithms employ the function and its gradient only; this avoids the computation of second derivatives.
4. The algorithms employ only information at the present stage, and at the stage immediately previous to the present; this reduces computer storage requirements as well as the computational work involved in each iteration.

Huang (1970) gave a unified approach to quadratically convergent algorithms for function minimization satisfying the aforementioned requirements, and derived a general updating formula containing rank 1 and rank 2 algorithms as special cases.

Consider an algorithm in which

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta \underline{x}^{(k)}, \tag{2.17}$$

where

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} \underline{p}^{(k)}, \tag{2.18}$$

$\underline{p}^{(k)}$ is a prescribed (n x 1) vector, and $\alpha^{(k)}$ is determined so that

$$f(\underline{x}^{(k+1)}) = \min_{\alpha} f(\underline{x}^{(k)} - \alpha \underline{p}^{(k)}) . \tag{2.19}$$

Then, requirement 1 is satisfied. If the algorithm is to have the descent property, then we must have

$$\underline{p}^{(k)T} \underline{g}^{(k)} \neq 0 \quad (\forall k \geq 0, \underline{g}^{(k)} \neq \underline{0}). \tag{2.20}$$

In this section the objective function is taken to be the quadratic function

$$f = \frac{1}{2} \underline{x}^T A \underline{x} + \underline{b}^T \underline{x} + C , \tag{2.21}$$

where A is an (n x n) positive definite symmetric matrix.

It was shown in §(1.2) that if the set of search directions $\{\underline{p}^{(0)}, \underline{p}^{(1)}, \ldots, \underline{p}^{(n-1)}\}$ are A-conjugate; so that

$$\underline{g}^{(k)T} \underline{p}^{(j)} = 0 \quad (j = 0, 1, \ldots, k-1), \tag{2.22}$$

then, A(2.1) is quadratically convergent by T(1.12). Thus requirement (2) is satisfied for any algorithm capable of generating such directions. Assume that

$$\underline{P}^{(k)T} A \underline{P}^{(j)} = 0 \quad (k-1 \geq j \geq 0;\ k < n). \tag{2.23}$$

If

$$\underline{P}^{(k)} = H^{(k)T} \underline{g}^{(k)}, \tag{2.24}$$

where $H^{(k)}$ is an (n x n) matrix (to be discussed later), then by (2.22)

$$\underline{g}^{(k)T} H^{(k)} A \underline{P}^{(j)} = 0 \quad (k-1 \geq j \geq 0). \tag{2.25}$$

Comparing (2.25) with (2.22) we deduce that (2.25) can be satisfied if the matrix $H^{(k)}$ is chosen such that

$$H^{(k)} A \underline{P}^{(j)} = \rho \underline{P}^{(j)} \quad (k-1 \geq j \geq 0). \tag{2.26}$$

where ρ is an arbitrary constant. Writing (2.26) for the previous iteration, we obtain

$$H^{(k-1)} A \underline{P}^{(j)} = \rho \underline{P}^{(j)} \quad (k-2 \geq j \geq 0). \tag{2.27}$$

By (2.26),

$$H^{(k)} A \underline{P}^{(j)} = \rho \underline{P}^{(j)} \quad (k-2 \geq j \geq 0), \tag{2.28}$$

$$H^{(k)} A \underline{P}^{(k-1)} = \rho \underline{P}^{(k-1)}. \tag{2.29}$$

By (2.27), (2.28),

$$(H^{(k)} - H^{(k-1)}) A \underline{P}^{(j)} = 0 \quad (k-2 \geq j \geq 0). \tag{2.30}$$

Let

$$C^{(k-1)} = H^{(k)} - H^{(k-1)}. \tag{2.31}$$

Then by (2.30), (2.31),

$$C^{(k-1)} A \underline{P}^{(j)} = 0 \quad (k-2 \geqslant j \geqslant 0). \tag{2.32}$$

Also, (2.29) is satisfied if

$$C^{(k-1)} A \underline{P}^{(k-1)} = \rho \underline{P}^{(k-1)} - H^{(k-1)} A \underline{P}^{(k-1)} . \tag{2.33}$$

Conditions (2.32), (2.33) can be expressed in the form

$$C^{(k-1)} \Delta \underline{g}^{(j)} = 0 \quad (k-2 \geqslant j \geqslant 0), \tag{2.34}$$

$$C^{(k-1)} \Delta \underline{g}^{(k-1)} = \rho \Delta \underline{x}^{(k-1)} - H^{(k-1)} \Delta \underline{g}^{(k-1)} . \tag{2.35}$$

If $C^{(k-1)}$ is expressed in the form

$$C^{(k-1)} = \rho \frac{\Delta \underline{x}^{(k-1)} \underline{z}^{(k-1)T}}{\underline{z}^{(k-1)T} \Delta \underline{g}^{(k-1)}} - \frac{H^{(k-1)} \Delta \underline{g}^{(k-1)} \underline{q}^{(k-1)T}}{\underline{q}^{(k-1)T} \Delta \underline{g}^{(k-1)}} , \tag{2.36}$$

where $\underline{z}^{(k-1)}$, $\underline{q}^{(k-1)}$ are two (n x 1) arbitrary vectors, apart from the conditions

$$\underline{z}^{(k+1)T} \Delta \underline{g}^{(k-1)} \neq 0 , \quad \underline{q}^{(k-1)T} \Delta \underline{g}^{(k-1)} \neq 0 , \tag{2.37}$$

then (2.35) is satisfied.

Also $C^{(k-1)}$ satisfies (2.34) if $\underline{z}^{(k-1)}$ and $\underline{q}^{(k-1)}$ satisfy

$$\underline{z}^{(k-1)T} \Delta \underline{g}^{(j)} = 0 , \quad \underline{q}^{(k-1)T} \Delta \underline{g}^{(j)} = 0 \quad (k-2 \geqslant j \geqslant 0). \tag{2.38}$$

It can be shown (Huang (1970) that (2.38) are satisfied if $\underline{z}^{(k-1)}$ and $\underline{q}^{(k-1)}$ are chosen as a linear combination of $\Delta \underline{x}^{(k-1)}$ and $H^{(k-1)T} \Delta \underline{g}^{(k-1)}$; that is,

$$\underline{z}^{(k-1)} = K_1 \Delta \underline{x}^{(k-1)} + K_2 H^{(k-1)T} \Delta \underline{g}^{(k-1)} , \tag{2.39}$$

and

$$\underline{q}^{(k-1)} = K_3 \Delta \underline{x}^{(k-1)} + K_4 H^{(k-1)T} \Delta \underline{g}^{(k-1)}, \tag{2.40}$$

where K_1, K_2, K_3, K_4 are scalar coefficients. With this choice of $\underline{z}^{(k-1)}$, $\underline{q}^{(k-1)}$ the requirements (3) and (4) are satisfied.

Thus from (2.31), (2.36), (2.39), (2.40) the matrix $H^{(k)}$ is updated by the formula

$$H^{(k)} = H^{(k-1)} + \rho \frac{\Delta \underline{x}^{(k-1)} (K_1 \Delta \underline{x}^{(k-1)} + K_2 H^{(k-1)T} \Delta \underline{g}^{(k-1)})^T}{(K_1 \Delta \underline{x}^{(k-1)} + K_2 H^{(k-1)T} \Delta \underline{g}^{(k-1)})^T \Delta \underline{g}^{(k-1)}}$$
$$- \frac{H^{(k+1)} \Delta \underline{g}^{(k-1)} (K_3 \Delta \underline{x}^{(k-1)} + K_4 H^{(k-1)T} \Delta \underline{g}^{(k-1)})^T}{(K_3 \Delta \underline{x}^{(k-1)} + K_4 H^{(k-1)} \Delta \underline{g}^{(k-1)})^T \Delta \underline{g}^{(k-1)}}, \tag{2.41}$$

where ρ, K_1, K_2, K_3, K_4 are arbitrary scalars apart from the conditions that K_3, K_4 must not vanish simultaneously, and when $\rho \neq o$ then K_1, K_2 must not vanish simultaneously.

Huang showed that in order to ensure the satisfaction of the nonorthogonality condition (2.20), the initial matrix $H^{(o)}$ must be such that

$$V = \frac{1}{2}(H^{(o)} + H^{(o)T}) \tag{2.42}$$

is positive definite or negative definite. If $H^{(o)}$ is symmetric, then $H^{(o)}$ must be positive definite or negative definite.

It is important to indicate that the sequence $\{H^{(k)}\}$ does not tend necessarily to the inverse Hessian A^{-1}. Writing (2.26) for $k = n$, we find

$$H^{(n)} A \underline{P}^{(j)} = \rho \underline{P}^{(j)} \quad (n-1 \geq j \geq o). \tag{2.43}$$

Since $\underline{p}^{(o)}, \underline{p}^{(1)}, \ldots, \underline{p}^{(n-1)}$ are n linearly independent vectors we have

$$H^{(n)} A = \rho I. \tag{2.44}$$

Hence,

$$H^{(n)} = \rho A^{-1} . \tag{2.45}$$

If $\rho > o$, then $H^{(n)}$ is positive definite. If $\rho = o$, then $H^{(n)} = 0$. If $\rho < o$, then $H^{(n)}$ is negative definite. Only when $\rho = 1$, we do have $H^{(n)} = A^{-1}$.

All known algorithms of single or double rank can be obtained from the general formula (2.41) by giving special values to the parameters ρ, K_1, K_2, K_3, K_4. For instance the Davidon -Fletcher-Powell (DFP) algorithm corresponding to the update

$$H^{(k)} = H^{(k-1)} - \frac{H^{(k-1)} \Delta \underline{g}^{(k-1)} \Delta \underline{g}^{(k-1)T} H^{(k-1)T}}{\Delta \underline{g}^{(k-1)T} H^{(k-1)} \Delta \underline{g}^{(k-1)}} + \frac{\Delta \underline{x}^{(k-1)} \Delta \underline{x}^{(k-1)T}}{\Delta \underline{g}^{(k-1)T} \Delta \underline{x}^{(k-1)}} \tag{2.46}$$

can be obtained from the general updating formula (2.43) by setting $\rho = K_1 = K_4 = 1$, $K_2 = K_3 = 0$.

A special but important subclass of Huang's family given by (2.41) is Broyden's family. This family of updates corresponds to the case when $\rho = 1$, and the correction $C^{(k-1)}$ is symmetric and rank 2. Broyden's family is given by

$$H^{(k)} = H_D^{(k)} + \phi \, \underline{v}^{(k-1)} \underline{v}^{(k-1)T}, \tag{2.47}$$

where

$$\underline{v}^{(k-1)} = \frac{\Delta \underline{x}^{(k-1)}}{\Delta \underline{g}^{(k-1)T} \Delta \underline{x}^{(k-1)}} - \frac{H^{(k+1)} \Delta \underline{g}^{(k+1)}}{\Delta \underline{g}^{(k-1)T} H^{(k-1)} \Delta \underline{g}^{(k-1)}} , \tag{2.48}$$

and $H_D^{(k)}$ is the DFP matrix given by (2.46).

As the scalar ϕ is varied in (2.47), we obtain most of the important symmetric quasi-Newton methods such as those of Fletcher and Powell (1963), Fletcher (1970), Goldfarb (1970), Greenstadt (1970). Also the rank 1 algorithms of Davidon (1968), Murtagh and Sargent (1969) and Barns (1965)

are also members. It is clear that $H^{(k)}$ for all members of this family tends to the inverse Hessian A^{-1}.

For a positive definite quadratic form, Huang (1970) proved that for a given initial point $\underline{x}^{(o)}$ and initial matrix $H^{(o)}$, the sequence of directions $\underline{p}^{(o)}, \underline{p}^{(1)},\ldots,\underline{p}^{(n-1)}$ generated by different algorithms of Huang's family are parallel; thus the sequence of points $\underline{x}^{(1)}, \underline{x}^{(2)},\ldots,\underline{x}^{(n)}$ are identical for all the algorithms, if the line search is exact.

Huang and Levy (1970) tested nine algorithms of Huang's family, and they showed that if high precision arithmetic together with high accuracy in the one dimensional search are employed, then all algorithms behave identically for any positive definite quadratic function. For non-quadratic functions the results show that some of these algorithms behave identically and thus any one of them can be considered to be representative of the entire class. Dixon (1971) proved that for a given initial point $\underline{x}^{(o)}$ and initial matrix $H^{(o)}$, all members of Broyden's family generate the same sequence of points for a general function, provided that an accurate line search is carried out and the matrix $H^{(k)}$ never becomes singular. The numerical experiments carried out by Huang and Levy (1970) and mentioned above confirmed the identical behaviour of the members of the family for a general function. If $\phi > o$ (in (2.47)), then $H^{(k)}$ remains positive definite if $H^{(o)}$ is positive definite (Broyden (1967)). Thus as a special case, if $\phi > o$, and $H^{(o)}$ is positive definite, then all corresponding members of Huang's family behave identically for a general function.

We conclude this section with a brief discussion of Greenstadt's approach to determine the correction matrix $C^{(k-1)}$. Greenstadt (1970) suggested that a good criterion for finding the best correction $C^{(k-1)}$ is to ask for the smallest correction in the sense of some norm. This would tend to keep the elements of $H^{(k-1)}$ from growing too large, which might cause undesired instability. The correction $C^{(k-1)}$ was required to minimize the weighted Euclidean norm defined by

$$C^{(k)} = Tr(WC^{(k-1)}WC^{(k-1)}), \tag{2.49}$$

where W is an (n x n) symmetric positive definite matrix, subject to the conditions

$$C^{(k-1)T} - C^{(k-1)} = 0, \tag{2.50}$$

$$C^{(k-1)}\Delta\underline{g}^{(k-1)} = \Delta\underline{x}^{(k-1)} - H^{(k-1)}\Delta\underline{g}^{(k-1)}. \tag{2.51}$$

The condition (2.50), which is a symmetry condition, insures that $H^{(k)}$ will remain symmetric as long as the initial estimate $H^{(o)}$ is chosen to be symmetric. The second condition (2.51) insures that the quasi-Newton formula is satisfied.

Using Lagrange multipliers to solve this constrained minimization problem, Greenstadt derived the updating form of the correction $C^{(k-1)}$, given by

$$C^{(k-1)} = \frac{1}{\Delta\underline{g}^T M\Delta\underline{g}}\left\{\left[\Delta\underline{x} - H\Delta\underline{g} - \frac{(\Delta\underline{g}^T\Delta\underline{x} - \Delta\underline{g}^T H\Delta\underline{g})(M\Delta\underline{g})}{\Delta\underline{g}^T M\Delta\underline{g}}\right]\right. \tag{2.52}$$
$$\left.(M\Delta\underline{g})^T + M\Delta\underline{g}\Delta\underline{x}^T - M\Delta\underline{g}(H\Delta\underline{g})^T\right\},$$

where $M = W^{-1}$ and H, $\Delta\underline{g}$, $\Delta\underline{x}$ denote $H^{(k-1)}$, $\Delta\underline{x}^{(k-1)}$, $\Delta\underline{g}^{(k-1)}$ respectively. The correction $C^{(k-1)}$ given by (2.52) is rank 3 if $M\Delta\underline{g}^{(k-1)}$, $\Delta\underline{x}^{(k-1)}$, and $H^{(k-1)}\Delta\underline{g}^{(k-1)}$ are linearly independent. Some choices of M can reduce the rank of $C^{(k-1)}$. Two interesting choices of M are $M = H^{(k-1)}$, and $M = H^{(k)}$. Both yield rank 2 correction matrices. If we denote them by $C_1^{(k-1)}$ and $C_2^{(k-1)}$ respectively, then

$$C_1^{(k-1)} = \frac{1}{\Delta\underline{g}^T H\Delta\underline{g}}\left\{\Delta\underline{x}\Delta\underline{g}^T H + H\Delta\underline{g}\Delta\underline{x}^T - \left(1 + \frac{\Delta\underline{g}^T\Delta\underline{x}}{\Delta\underline{g}^T H\Delta\underline{g}}\right)H\Delta\underline{g}\Delta\underline{g}^T H\right\}, \tag{2.53}$$

and

$$C_2^{(k-1)} = \frac{1}{\Delta\underline{x}^T\Delta\underline{g}}\left\{-\Delta\underline{x}\Delta\underline{g}^T H - H\Delta\underline{g}\Delta\underline{x}^T + \left(1 + \frac{\Delta\underline{g}^T H\Delta\underline{g}}{\Delta\underline{g}^T\Delta\underline{x}}\right)\Delta\underline{x}\Delta\underline{x}^T\right\}, \tag{2.54}$$

where $\Delta\underline{x}$, $\Delta\underline{g}$, H denote $\Delta\underline{x}^{(k-1)}$, $\Delta\underline{g}^{(k-1)}$, $H^{(k-1)}$ respectively. These two variationally derived corrections give rise to the one-parameter family of corrections given by

$$C^{(k-1)} = t\,C_1^{(k-1)} + (1-t)\,C_2^{(k-1)} . \tag{2.55}$$

This formula comprises most of the known symmetric quasi-Newton corrections such as those of Fletcher and Powell (1963), Davidon (1968), Broyden (1967). It can be shown that the family of updates corresponding to (2.55) is equivalent to Broyden's family given by (2.47), (2.48) (Goldfarb (1970)).

§(2.4) Relation between Conjugate Gradient Methods and Quasi-Newton Methods

In this section it is shown that conjugate gradient methods are related, by virtue of their projection properties, to quasi-Newton methods. A similar notation to that used by Pearson (1969) is adopted. In this notation two (n x k) matrices are defined by

$$Y^{(k)} = \left[\Delta\underline{g}^{(0)}, \ldots, \Delta\underline{g}^{(k-1)}\right], \tag{2.56}$$

$$S^{(k)} = \left[\Delta\underline{x}^{(0)}, \ldots, \Delta\underline{x}^{(k-1)}\right]. \tag{2.57}$$

The conjugacy conditions given by (2.23) can be written in one of the two equivalent forms

$$\Delta\underline{x}^{(k)T}\,\Delta\underline{g}^{(j)} = 0 \quad (k-1 \geq j \geq 0), \tag{2.58}$$

or

$$\Delta\underline{g}^{(k)T}\,\Delta\underline{x}^{(j)} = 0 \quad (k-1 \geq j \geq 0). \tag{2.59}$$

Using notations (2.56), (2.57) we find that the conjugacy of the search directions $\underline{p}^{(k)}$ can be written as

$$S^{(k)T}\,\Delta\underline{g}^{(j)} = 0 \quad (j<k), \tag{2.60}$$

or

$$Y^{(k)T} \Delta \underline{x}^{(j)} = 0 \quad (j<k). \tag{2.61}$$

Consequently, from (2.61) any method which can generate new directions $\Delta \underline{x}^{(k+1)}$ such that (2.61) is satisfied for k = 1,2,... will be able to minimize a positive definite quadratic form in at most n steps, since it generates n conjugate directions.

Let us choose the search direction $\underline{p}^{(k)}$ such that

$$\underline{p}^{(0)} = \underline{g}^{(0)}, \tag{2.62}$$

$$\underline{p}^{(k)} = -\underline{g}^{(k)} - \sum_{j=0}^{k-1} \frac{\underline{g}^{(k)T} \Delta \underline{g}^{(j)}}{\Delta \underline{g}^{(j)T} \Delta \underline{g}^{(j)}} \Delta \underline{g}^{(j)}. \tag{2.63}$$

Equation (2.63) means that $\underline{p}^{(k)}$ is equal to the component of $-\underline{g}^{(k)}$ which is orthogonal to $\Delta \underline{g}^{(o)}$, $\Delta \underline{g}^{(1)}$,...,$\Delta \underline{g}^{(k-1)}$. This choice ensures that the new step $\Delta \underline{x}^{(k)} = \alpha^{(k)} \underline{p}^{(k)}$ satisfies the conjugacy conditions (2.58).

One way of achieving (2.62), (2.63) is to define $\underline{p}^{(k)}$ by

$$\underline{p}^{(k)} = -Q^{(k)} \underline{g}^{(k)} \qquad (k \geq 0), \tag{2.64}$$

where

$$Q^{(k)} = I - Y^{(k-1)} \left(Y^{(k-1)T} Y^{(k-1)} \right)^{-1} Y^{(k-1)}. \tag{2.65}$$

The matrix $Q^{(k)}$ projects the gradient $\underline{g}^{(k)}$ orthogonal to $\Delta \underline{g}^{(o)}$,...,$\Delta \underline{g}^{(k-1)}$. This matrix has the properties

$$Q^{(0)} = I, \tag{2.66}$$

$$Q^{(n)} = 0, \tag{2.67}$$

$$Q^{(k)} \Delta \underline{g}^{(j)} = 0 \qquad (j<k), \tag{2.68}$$

$Q^{(k)} \underline{w} = \underline{w}$ for any w such that (2.69)

$$\underline{w}^T \Delta \underline{g}^{(j)} = 0 \quad (j = 0, 1, \ldots, k-1).$$

It is clear that the algorithm based on (2.65) is not a practical one, unless a formula for updating $Q^{(k)}$ is found. From (2.68) the new matrix $Q^{(k+1)}$ should satisfy

$$Q^{(k+1)} \Delta \underline{g}^{(j)} = 0 \quad (j = 0, 1, \ldots, k). \tag{2.70}$$

Let us separate (2.70) into two groups as follows

$$Q^{(k+1)} \Delta \underline{g}^{(k)} = 0 , \tag{2.71}$$

$$Q^{(k+1)} \Delta \underline{g}^{(j)} = 0 \quad (j = 0, 1, \ldots, k-1) . \tag{2.72}$$

Following the way used in updating the current matrix in quasi-Newton methods, we construct $Q^{(k+1)}$ from $Q^{(k)}$ by adding a correction $C^{(k)}$, so that

$$Q^{(k+1)} = Q^{(k)} + C^{(k)} . \tag{2.73}$$

Now, by (2.71), (2.73), (2.72), (2.68) we have

$$C^{(k)} \Delta \underline{g}^{(k)} = -Q^{(k)} \Delta \underline{g}^{(k)} , \tag{2.74}$$

$$C^{(k)} \Delta \underline{g}^{(j)} = 0 \quad (j = 0, 1, \ldots, k-1) . \tag{2.75}$$

One possible solution of (2.74) is

$$C^{(k)} = - \frac{Q^{(k)} \Delta \underline{g}^{(k)} \underline{z}^{(k)T}}{\underline{z}^{(k)T} \Delta \underline{g}^{(k)}} , \tag{2.76}$$

where $\underline{z}^{(k)}$ is an (n x 1) arbitrary vector apart from the condition

$$\underline{z}^{(k)T} \Delta \underline{g}^{(k)} \neq 0 \ . \tag{2.77}$$

If we choose $\underline{z}^{(k)} = Q^{(k)} \Delta \underline{g}^{(k)}$ then by (2.68) we find that $C^{(k)}$ satisfies (2.75) as well. By means of this choice (2.76) takes the form

$$C^{(k)} = -\frac{Q^{(k)} \Delta \underline{g}^{(k)} \Delta \underline{g}^{(k)} Q^{(k)}}{\Delta \underline{g}^{(k)T} Q^{(k)} \Delta \underline{g}^{(k)}} \ . \tag{2.78}$$

Hence

$$Q^{(k+1)} = Q^{(k)} - \frac{Q^{(k)} \Delta \underline{g}^{(k)} \Delta \underline{g}^{(k)T} Q^{(k)}}{\Delta \underline{g}^{(k)} Q^{(k)} \Delta \underline{g}^{(k)}} \ , \tag{2.79}$$

giving rise to a quadratically convergent conjugate gradient algorithm.

Actually this algorithm is exactly A(2.1) provided that $C^{(k)}$ in A(2.1) is updated by (2.78). The initial matrix $Q^{(o)}$ can be taken to be the unit matrix I or any positive definite symmetric matrix. Iteration (2.79) can be obtained from the Huang's formula (2.41) by setting $\rho = K_1 = K_2 = K_3 = 0$ and $K_4 = 1$. However, this algorithm is not a quasi-Newton algorithm for the matrix $Q^{(k)}$ does not satisfy the quasi-Newton formula as indicated by (2.70). Moreover, by (2.67) we have $Q^{(n)} = 0$. Pearson (1969) states that in certain cases the algorithm corresponding to (2.79) is preferable to the Fletcher-Reeves algorithm. Unfortunately the storage requirements are comparable to those of quasi-Newton methods, and the DFP method in particular is even better (Fletcher (1972)). Clearly the algorithm corresponding to (2.79) is related very closely to the DFP algorithm given by

$$H^{(k+1)} = H^{(k)} - Z^{(k)} + W^{(k)} \quad (k \geq 0) \ , \tag{2.80}$$

where

$$Z^{(k)} = \frac{H^{(k)} \Delta \underline{g}^{(k)} \Delta \underline{g}^{(k)T} H^{(k)}}{\Delta \underline{g}^{(k)T} H^{(k)} \Delta \underline{g}^{(k)}} \ , \tag{2.81}$$

and

$$W^{(k)} = \frac{\Delta \underline{x}^{(k)} \Delta \underline{x}^{(k)T}}{\Delta \underline{g}^{(k)T} \Delta \underline{x}^{(k)}} \quad . \tag{2.82}$$

Suppose that $H^{(o)} = I$. Then (2.80) can be written in the form

$$H^{(k+1)} = I - \sum_{i=o}^{k} Z^{(i)} + \sum_{i=o}^{k} W^{(i)} \quad . \tag{2.83}$$

It can be shown (Myers(1968)) that $H^{(k+1)}$ can be written in the form

$$H^{(k+1)} = Q^{(k)} + \sum_{i=o}^{k} W^{(i)} \quad . \tag{2.84}$$

By (2.67), (1.92),

$$Q^{(n)} = 0 \quad , \tag{2.85}$$

$$\sum_{i=o}^{n} W^{(i)} = A^{-1} \quad . \tag{2.86}$$

The search direction in the DFP method is given by

$$\underline{p}^{(k+1)} = -H^{(k+1)} \underline{g}^{(k+1)} \quad . \tag{2.87}$$

Since the directions generated by the DFP method are conjugate, we have, by (2.82) and T(1.10)

$$\sum_{i=o}^{k} W^{(i)} \underline{g}^{(k+1)} = 0 \quad (k \geq o) \quad . \tag{2.88}$$

For a quadratic function the matrix $Q^{(k)}$ generates the conjugate search directions by virtue of its projection properties, while the term $\sum_{i=o}^{k} W^{(i)}$ of $H^{(k+1)}$ is of no use in the minimization process. For non-quadratic functions the matrix H, and not Q, tends to A^{-1} which produces rapid convergence near the minimum.

§(2.5) Quasi-Newton Methods for Solving Systems of Nonlinear Equations

We consider in this section the problem of solving a system of n nonlinear equations in n unknowns, namely

$$\underline{F}(\underline{x}) = \underline{0}\,, \tag{2.89}$$

where $\underline{F}:R^n \to R^n$ is given.

Newton's method for solving (2.89) can be written in the form

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \alpha^{(k)} J^{(k)-1} \underline{F}^{(k)}\,, \tag{2.90}$$

where $\underline{F}^{(k)} = \underline{F}(\underline{x}^{(k)})$, $J^{(k)}$ is the Jacobian matrix $(\partial_j F_i(\underline{x}^{(k)}))$, $(i,j = 1,\ldots,n)$, and $\alpha^{(k)}$ is a scalar chosen such that

$$\|\underline{F}^{(k+1)}\| < \|\underline{F}^{(k)}\|\,. \tag{2.91}$$

Quasi-Newton methods for solving (2.89) are of the form

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}, \tag{2.92}$$

$$\Delta\underline{x}^{(k)} = \alpha^{(k)}\,\underline{P}^{(k)}\,, \tag{2.93}$$

$$\underline{P}^{(k)} = -H^{(k)}\,\underline{F}^{(k)}, \tag{2.94}$$

where $\alpha^{(k)}$ is a scalar, and $H^{(k)}$ is an (n x n) matrix representing the kth approximation to $J^{-1}(\underline{x}^*)$.

On the space of residuals we introduce the norm

$$\|\underline{F}\|_M \stackrel{D}{=} (\underline{F}^T M \underline{F})^{\frac{1}{2}}, \tag{2.95}$$

where M is a positive definite symmetric matrix. By Taylor's theorem the system (2.89) can be approximated by a linear system in some neighbourhood of the solution $\underline{x}^*$, (see §(1.5)), so an iterative method which works well for a linear system, may also work well for the general

case. Newton's method solves a linear system (when it is solvable) in one step, so quasi-Newton methods are required to resemble Newton's method as much as possible while avoiding the objectionable features of the latter (see §(2.1)). For the linear system

$$\underline{F}(\underline{x}) \overset{D}{\equiv} J\underline{x} - \underline{b} = \underline{0}\ , \tag{2.96}$$

where J is an (n x n) nonsingular constant matrix, and $\underline{b}$ is (n x 1) constant vector, the Jacobian J satisfies the relation

$$J^{-1}\Delta\underline{F}^{(j)} = \Delta\underline{x}^{(j)} \quad (j \geq 0). \tag{2.97}$$

Since $H^{(k)}$ is an approximation to $J^{(k)-1}$ we would like it also to satisfy (2.97), but since we cannot compute $\underline{F}^{(k+1)}$ and hence $\Delta\underline{F}^{(k)}$ until we have determined $H^{(k)}$ this is clearly impossible. However, we can require that the next approximation to $J^{(k)-1}$, namely $H^{(k+1)}$, satisfy the quasi-Newton formula

$$H^{(k+1)}\Delta\underline{F}^{(j)} = \Delta\underline{x}^{(j)} \quad (j = 0, 1, \cdots, k). \tag{2.98}$$

Let

$$H^{(k+1)} = H^{(k)} + C^{(k)}\ . \tag{2.99}$$

Then (2.98) can be written in the form

$$C^{(k)}\Delta\underline{F}^{(j)} = \Delta\underline{x}^{(j)} - H^{(k)}\Delta\underline{F}^{(j)} \quad (j = 0, 1, \ldots, k). \tag{2.100}$$

By (2.100),

$$C^{(k)}\Delta\underline{F}^{(k)} = \Delta\underline{x}^{(k)} - H^{(k)}\Delta\underline{F}^{(k)}, \tag{2.101}$$

$$C^{(k)}\Delta\underline{F}^{(j)} = \Delta\underline{x}^{(j)} - H^{(k)}\Delta\underline{F}^{(j)} \quad (j = 0, 1, \ldots, k-1)\ . \tag{2.102}$$

From (2.98) written for the previous iteration, we have by (2.102),

$$C^{(k)} \Delta \underline{F}^{(j)} = 0 \quad (j = 0, \ldots, k-1) . \tag{2.103}$$

From (2.101) we find that $C^{(k)}$ can assume the following form

$$C^{(k)} = \frac{\Delta \underline{x}^{(k)} (M \underline{z}^{(k)})^T}{\underline{z}^{(k)T} M \Delta \underline{F}^{(k)}} - \frac{H^{(k)} \Delta \underline{F}^{(k)} (M \underline{q}^{(k)})^T}{\underline{q}^{(k)T} M \Delta \underline{F}^{(k)}} , \tag{2.104}$$

where $\underline{z}^{(k)}$ and $\underline{q}^{(k)}$ are two arbitrary (n x 1) vectors, apart from the conditions

$$\underline{z}^{(k)T} M \Delta \underline{F}^{(k)} \neq 0 , \quad \underline{q}^{(k)T} M \Delta \underline{F}^{(k)} \neq 0 . \tag{2.105}$$

The matrix $C^{(k)}$ given by (2.104) satisfies (2.101). To satisfy (2.103) as well, $\underline{z}^{(k)}$ and $\underline{q}^{(k)}$ should be chosen such that

$$\underline{z}^{(k)T} M \Delta \underline{F}^{(j)} = \underline{q}^{(k)T} M \Delta \underline{F}^{(j)} = 0 \quad (j<k) . \tag{2.106}$$

Multiplying both sides of (2.98) by J and using (2.97) we find for k = n-1,

$$J H^{(n)} \Delta \underline{F}^{(j)} = \Delta \underline{F}^{(j)} \quad (j = 0, \ldots, n-1) \tag{2.107}$$

Assuming that $\Delta \underline{F}^{(0)}, \Delta \underline{F}^{(1)}, \ldots, \Delta \underline{F}^{(n-1)}$ are linearly independent, we deduce from (2.107) that

$$H^{(n)} = J^{-1} . \tag{2.108}$$

Thus the next step $\Delta \underline{x}^{(n)}$ will lead to the solution of (2.89).

Formula (2.104) is rank 2. If we set $\underline{z}^{(k)} = \underline{q}^{(k)}$ then we get the rank 1 formula

$$C^{(k)} = \frac{(\Delta \underline{x}^{(k)} - H^{(k)} \Delta \underline{F}^{(k)})(M \underline{z}^{(k)})^T}{\underline{z}^{(k)T} M \Delta \underline{F}^{(k)}} , \tag{2.109}$$

where $\underline{z}^{(k)}$ is an (n x 1) arbitrary vector, which satisfies (2.105) and (2.106). Combining (2.109) and (2.99) we get the general updating formula

$$H^{(k+1)} = H^{(k)} + \frac{(\Delta \underline{x}^{(k)} - H^{(k)} \Delta \underline{F}^{(k)})(M \underline{z}^{(k)})^T}{\underline{z}^{(k)T} M \Delta \underline{F}^{(k)}} . \qquad (2.110)$$

Formulae (2.104) and (2.109) were derived by Zeleznik (1968). Previous quasi-Newton algorithms such as those of Barnes (1965), Broyden (1965), Rosen (1966) are special cases of (2.110) corresponding to special choices of the arbitrary vector $\underline{z}^{(k)}$. Also in all of them the matrix M is taken to be the unit matrix I.

From the previous consideration we have the following general rank 1 algorithm for solving a system of nonlinear equations.

Algorithm (2.2)

Given an initial estimate $\underline{x}^{(o)}$ of the solution $\underline{x}^*$ of $\underline{F}(\underline{x}) = \underline{0}$, an initial estimate $H^{(o)}$ of the inverse Jacobian $J^{-1}(x^*)$, and a positive definite symmetric matrix M.

1. Set k = o.
2. Compute $\underline{F}^{(o)}$ from

$$\underline{F}^{(o)} = \underline{F}(\underline{x}^{(o)}).$$

3. Compute $\alpha^{(k)}$, $\underline{x}^{(k+1)}$ from

$$\| \underline{F}(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)}) \|_M = \min_{\alpha} \| \underline{F}(\underline{x}^{(k)} - \alpha H^{(k)} \underline{F}^{(k)}) \|_M ,$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{F}^{(k)} .$$

4. Compute $\underline{F}^{(k+1)}$, $\Delta \underline{x}^{(k)}$, $\Delta \underline{F}^{(k)}$ from

$$\underline{F}^{(k+1)} = \underline{F}(\underline{x}^{(k+1)}),$$

$$\Delta \underline{x}^{(k)} = \underline{x}^{(k+1)} - \underline{x}^{(k)},$$

$$\Delta \underline{F}^{(k)} = \underline{F}^{(k+1)} - \underline{F}^{(k)} .$$

5. Choose $\underline{z}^{(k)}$ such that

if $k \leq n-1$ then $\underline{z}^{(k)T} M \Delta\underline{F}^{(j)} = o \ (j=o,\ldots,k-1)$, (2.111a)

and if $k > n-1$ then $\underline{z}^{(k)T} M \Delta\underline{F}^{(j)} = o \ (j=k-n,\ldots,k-1)$. (2.111b)

6. Update $C^{(k)}$ by means of (2.109)
7. Compute $H^{(k+1)}$ from

$$H^{(k+1)} = H^{(k)} + C^{(k)}.$$

8. Set $k = k+1$ and go to 3.

□

The remainder of this section is devoted to a brief survey of some algorithms for solving systems of n nonlinear equations.

The method of Barnes (1965) is a generalization of the secant method for the solution of one equation in one unknown.

Algorithm (2.3)

For a given initial point $\underline{x}^{(o)}$ and initial estimate $B^{(o)}$ to the Jacobian matrix $J(\underline{x}^*)$

1. Set $k = o$.
2. Compute $\underline{F}^{(k)}$ from

$$\underline{F}^{(k)} = \underline{F}(\underline{x}^{(k)}).$$

3. Calculate the inverse matrix $B^{(k)-1}$ of the approximated Jacobian $B^{(k)}$.
4. Compute $\Delta\underline{x}^{(k)}$ from

$$\Delta\underline{x}^{(k)} = -B^{(k)-1}\underline{F}^{(k)}. \qquad (2.112)$$

5. Compute $\underline{x}^{(k+1)}$, $\underline{F}^{(k+1)}$ from

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta \underline{x}^{(k)},$$

$$\underline{F}^{(k+1)} = \underline{F}(\underline{x}^{(k+1)}).$$

6. If $k \leq n-1$ choose a vector $\underline{z}^{(k)}$ orthogonal to the previous k steps.
If $k > n-1$ choose $\underline{z}^{(k)}$ orthogonal to the n-1 previous steps..

7. Compute the matrix $D^{(k)}$ from

$$D^{(k)} = \frac{\underline{F}^{(k+1)} \underline{z}^{(k)T}}{\underline{z}^{(k)T} \Delta \underline{x}^{(k)}} . \tag{2.113}$$

8. Compute $B^{(k+1)}$ from

$$B^{(k+1)} = B^{(k)} + D^{(k)} . \tag{2.114}$$

9. Set $k = k+1$, and go to 3. □

Barnes suggested that (6) can be done via Gram-Schmidt orthogonalization procedure given by T(1.27). Barnes' algorithm enjoys the property of linear termination; $B^{(k)}$ becomes equal to the exact Jacobian after n steps, and a further Newton's step gives the solution. When the algorithm is applied to a set of nonlinear equations it suffers from two serious disadvantages:

(i) the algorithm as it is presented is not stable, and there is no guarantee that $\| \underline{F}^{(k+1)} \| < \| \underline{F}^{(k)} \|$. This can be avoided by taking in (4) a step $\Delta \underline{x}^{(k)}$ defined by

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} B^{(k)} \underline{F}^{(k)}, \tag{2.115}$$

where $\alpha^{(k)}$ is chosen such that $\| \underline{F}^{(k+1)} \| < \| \underline{F}^{(k)} \|$.

(ii) as long as the Jacobian and not its inverse is being approximated each step involves a solution of a system of linear equations.

The updating formula given by Barnes is

$$B^{(k+1)} = B^{(k)} + \frac{\underline{F}^{(k+1)} \underline{Z}^{(k)T}}{\underline{Z}^{(k)T} \Delta \underline{x}^{(k)}} \,. \tag{2.116}$$

If we set in (2.116) $\Delta \underline{x}^{(k)} = \alpha^{(k)} \underline{p}^{(k)}$ and $B^{(k)} = H^{(k)-1}$, then by applying Householder's formula to (2.116) we find Barnes' algorithm as represented by Rosen, namely

$$H^{(k+1)} = H^{(k)} - \frac{(H^{(k)} \Delta \underline{F}^{(k)} - \alpha^{(k)} \underline{p}^{(k)}) \underline{Z}^{(k)T} H^{(k)}}{\underline{Z}^{(k)T} H^{(k)} \Delta \underline{F}^{(k)}} \,, \tag{2.117}$$

in which $\alpha^{(k)}$ is chosen such that the descent property $\| \underline{F}^{(k+1)} \| < \| \underline{F}^{(k)} \|$ is satisfied. Thus, this algorithm circumvents the difficulties (i) and (ii) related to Barnes' algorithm. If $\underline{z}^{(k)}$ is chosen as in Barnes' method, then Rosen's algorithm has the property of linear termination. Finally, Rosen's algorithm (2.117) can be derived from the general rank 1 formula (2.110) by setting

$$M = I,$$
$$\underline{Z}^{(k)} = H^{(k)T} \underline{u}^{(k)},$$

where $\underline{u}^{(k)}$ is unspecified.

If we set in formula (2.110)

$$M = I,$$
$$\underline{Z}^{(k)} = H^{(k)} \Delta \underline{x}^{(k)},$$

then, the correction $C^{(k)}$ is given by

$$C^{(k)} = (\alpha^{(k)} \underline{p}^{(k)} - H^{(k)} \Delta \underline{F}^{(k)}) \frac{\Delta \underline{x}^{(k)T} H^{(k)}}{\Delta \underline{F}^{(k)T} H^{(k)} \Delta \underline{x}^{(k)}} \,, \tag{2.118}$$

which is one of the Broyden's (1965) algorithms for solving a system of nonlinear equations. Another of Broyden's algorithms can be obtained from (2.110) by setting $M = I$ and $\underline{z}^{(k)} = \Delta \underline{F}^{(k)}$. Broyden indicated

that the latter algorithm seems in practice to be unsatisfactory.

Zeleznik (1968) criticised the choice of $\underline{z}^{(k)}$ given by Broyden in both algorithms. He explained that one has no assurance that $\Delta\underline{x}^{(k)}$ will be orthogonal to the subspace spanned by the previous steps $\Delta\underline{x}^{(j)}$ $(j < k)$, and thus one can not be sure of convergence even for the linear case, but in the light of recent work (Broyden (1970)) this is not a valid criticism. However, in the following discussion (Broyden (1965)) we will show that for the linear case, the approximated Jacobian $B^{(k+1)}$ is not worse than $B^{(k)}$, in the sense that $\|J-B^{(k+1)}\|$ is not greater than $\|J-B^{(k)}\|$, where $\|.\|$ is the Euclidean norm. In Broyden's algorithm $B^{(k+1)}$ is given by

$$B^{(k+1)} = B^{(k)} - \frac{(J\underline{P}^{(k)} - B^{(k)}\underline{P}^{(k)})\underline{P}^{(k)T}}{\underline{P}^{(k)T}\underline{P}^{(k)}},$$

$$= B^{(k)} - (B^{(k)} - J)\frac{\underline{P}^{(k)}\underline{P}^{(k)T}}{\underline{P}^{(k)T}\underline{P}^{(k)}} \quad . \tag{2.119}$$

Define the error matrix by

$$R^{(k)} = B^{(k)} - J. \tag{2.120}$$

Then from (2.119), (2.120)

$$R^{(k+1)} = R^{(k)}\left(I - \frac{\underline{P}^{(k)}\underline{P}^{(k)T}}{\underline{P}^{(k)T}\underline{P}^{(k)}}\right). \tag{2.121}$$

Hence,

$$\|R^{(k+1)}\| \leq \|R^{(k)}\| \cdot \left\|I - \frac{\underline{P}^{(k)}\underline{P}^{(k)T}}{\underline{P}^{(k)T}\underline{P}^{(k)}}\right\| \tag{2.122}$$

But the matrix $\left(I - \frac{\underline{P}^{(k)}\underline{P}^{(k)T}}{\underline{P}^{(k)T}\underline{P}^{(k)}}\right)$ has one eigenvalue equal to zero and (n-1) eigenvalues equal to unity (see appendix (2)). Thus its Euclidean norm is equal to one. Hence the sequence $\{\|R^{(k)}\|\}$ is decreasing monotonically, and thus the method is of <u>bounded deterioration</u> (Broyden (1972)).

This algorithm is stable (of bounded deterioration) provided that $B^{(k)}$ approximates $J^{(k)}$ sufficiently well. If the approximation is poor, then the algorithm is unstable and the performance of the method consequently suffers (Broyden (1972)). As indicated before, this algorithm does not enjoy the property of linear termination, even when an exact line search is carried out. On the contrary an exact line search gives poorer results than merely reducing $\|\underline{F}\|$. Thus α is taken just to reduce $\|\underline{F}\|$ as found by Broyden (1965).

CHAPTER 3

QUASI-NEWTON METHODS WITH MEMORY FOR UNCONSTRAINED MINIMIZATION

§(3.1) Introduction

In this chapter a new approach to the problem of unconstrained minimization is considered. Miele and Cantrell (1969) showed that if the method of steepest descent is modified by adding a so-called memory term, the resulting method, called by them the memory gradient method, is in general superior to the method of steepest descent and to the method of Fletcher and Reeves (1964). Each iteration of the memory gradient method involves the unconstrained minimization of a function of two variables, the so-called two dimensional search, instead of the one-dimensional line search used in all other methods so far considered in this thesis.

Cantrell (1969) showed that for a strictly convex quadratic function the algorithm of Miele and Cantrell and that of Fletcher and Reeves give identical results. Cragg and Levy (1969) extended the memory gradient method by including additional memory terms. This extension gave rise to super memory gradient methods. Wolfe (1974) introduced the idea of memory into quasi-Newton methods, producing what he called quasi-Newton methods with memory. Wolfe proved that any algorithm of Broyden's family can be provided with memory without violating its quadratic convergence, for his algorithm and the corresponding quasi-Newton algorithm give identical results for a strictly convex quadratic function. For nonquadratic objective functions, numerical evidence suggests that quasi-Newton methods with memory are usually more rapidly convergent and often more computationally efficient than the corresponding quasi-Newton methods.

§(3.2) The Memory Gradient Method

The memory gradient method for unconstrained minimization is due to Miele and Cantrell (1969) who introduced a memory term into the steepest descent method. The algorithm corresponding to the memory gradient method (MG) is as follows.

Algorithm (3.1)

Given an initial estimate $\underline{x}^{(o)}$ of an unconstrained minimum $\underline{x}^*$ of f.

1. Compute $\underline{g}^{(o)}$ from

$$\underline{g}^{(o)} = \underline{g}(\underline{x}^{(o)}).$$

2. Compute $\alpha^{(o)}$, $\Delta\underline{x}^{(o)}$, $\underline{x}^{(1)}$ from

$$f(\underline{x}^{(o)} - \alpha^{(o)}\underline{g}^{(o)}) = \min_{\alpha} f(\underline{x}^{(o)} - \alpha\underline{g}^{(o)}),$$

$$\Delta\underline{x}^{(o)} = -\alpha^{(o)}\underline{g}^{(o)},$$

$$\underline{x}^{(1)} = \underline{x}^{(o)} + \Delta\underline{x}^{(o)}.$$

3. Set k = 1.

4. Compute $\underline{g}^{(k)}$ from

$$\underline{g}^{(k)} = \underline{g}(\underline{x}^{(k)}).$$

5. Compute $\alpha^{(k)}$, $\beta^{(k)}$, $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)}\underline{g}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)}) = \min_{\alpha,\beta} f(\underline{x}^{(k)} - \alpha\underline{g}^{(k)} + \beta\Delta\underline{x}^{(k-1)}), \tag{3.1}$$

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{g}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)}. \tag{3.2}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}. \tag{3.3}$$

6. Set k = k+1 and go to 4.

□

To first order terms, the values of the function at $\underline{x}^{(k+1)}$ and $\underline{x}^{(k)}$ are related by

$$f(\underline{x}^{(k+1)}) \approx f(\underline{x}^{(k)}) + \delta f(\underline{x}^{(k)}), \tag{3.4}$$

where the first variation $\delta f(\underline{x}^{(k)})$ is given by

$$\delta f(\underline{x}^{(k)}) = \underline{g}^{(k)T}\Delta\underline{x}^{(k)}. \tag{3.5}$$

The choice of $\Delta\underline{x}^{(k)}$ as given in (3.2) minimizes the first variation $\delta f(\underline{x})$ of f at $\underline{x}^{(k)}$ subject to the constraint

$$K = (\Delta\underline{x} - \gamma\,\Delta\underline{x}^{(k-1)})^{T}(\Delta\underline{x} - \gamma\Delta\underline{x}^{(k-1)}), \tag{3.6}$$

where K and γ are prescribed. Let $\underline{h} = \Delta\underline{x}^{(k)}$ minimize $\underline{g}^{(k)T}\underline{h}$ subject to (3.6). The Lagrange function $\phi(\underline{x}^{(k)}, \underline{h})$ for this constrained minimization problem is given by

$$\phi(\underline{x}^{(k)}, \underline{h}) = \underline{g}^{(k)T}\underline{h} + \lambda(\underline{h} - \gamma\Delta\underline{x}^{(k-1)})^{T}(\underline{h} - \gamma\Delta\underline{x}^{(k-1)}). \tag{3.7}$$

Then $\Delta\underline{x}^{(k)}$ satisfies

$$\left[\frac{\partial\phi(\underline{x}^{(k)}, \underline{h})}{\partial h_i}\right]_{\underline{h}=\Delta\underline{x}^{(k)}} = 0 \quad (i = 1, \ldots, n) \ . \tag{3.8}$$

Hence

$$\Delta\underline{x}^{(k)} = -\frac{1}{2\lambda}\,\underline{g}^{(k)} + \gamma\,\Delta\underline{x}^{(k-1)} \ . \tag{3.9}$$

Write

$$\alpha = \frac{1}{2\lambda} \ , \quad \beta = \gamma \ . \tag{3.10}$$

Then

$$\Delta\underline{x}^{(k)} = -\alpha\underline{g}^{(k)} + \beta\,\Delta\underline{x}^{(k-1)} \ . \tag{3.11}$$

Let

$$\psi(\alpha, \beta) = f(\underline{x}^{(k)} - \alpha\underline{g}^{(k)} + \beta\,\Delta\underline{x}^{(k-1)}) \ . \tag{3.12}$$

Necessary conditions for $\psi(\alpha,\beta)$ to have a minimum at (α,β) are

$$\psi_{\alpha}(\alpha,\beta) \overset{D}{=} \frac{\partial\psi}{\partial\alpha}(\alpha,\beta) = 0 \ , \tag{3.13a}$$

$$\psi_{\beta}(\alpha,\beta) \overset{D}{=} \frac{\partial\psi}{\partial\beta}(\alpha,\beta) = 0 \ . \tag{3.13b}$$

By (3.12) and (3.2), (3.13) can be written in the form

$$\psi_\alpha(\alpha,\beta) = -\underline{g}^{(k+1)T}\underline{g}^{(k)} = 0, \tag{3.14a}$$

$$\psi_\beta(\alpha,\beta) = \underline{g}^{(k+1)T}\Delta\underline{x}^{(k-1)} = 0. \tag{3.14b}$$

From (3.2), (3.14) we have

$$\underline{g}^{(k+1)T}\Delta\underline{x}^{(k)} = 0. \tag{3.15}$$

It is clear from (3.1) that the memory gradient method requires a two dimensional search. The one described by Miele and Cantrell (1969) is based on a quasi-linearization with built-in safeguards to ensure that $f(\underline{x})$ decreases at every step of the iterative search.

Let α_o, β_o be initial estimates of α, β, and let

$$\Delta\alpha = \alpha - \alpha_o \quad , \quad \Delta\beta = \beta - \beta_o . \tag{3.16}$$

If quasilinearization is applied to (3.13), we obtain the linear equations

$$\psi_{\alpha\alpha}\Delta\alpha + \psi_{\alpha\beta}\Delta\beta + \psi_\alpha = 0, \tag{3.17a}$$

$$\psi_{\beta\alpha}\Delta\alpha + \psi_{\beta\beta}\Delta\beta + \psi_\beta = 0, \tag{3.17b}$$

where ψ_α, ψ_β, $\psi_{\alpha\alpha}$, $\psi_{\alpha\beta}$, $\psi_{\beta\alpha}$, $\psi_{\beta\beta}$ are computed at α_o, β_o. Next, we imbed (3.17) in the more general equations

$$\psi_{\alpha\alpha}\Delta\alpha + \psi_{\alpha\beta}\Delta\beta + \mu\rho\psi_\alpha = 0, \tag{3.18a}$$

$$\psi_{\beta\alpha}\Delta\alpha + \psi_{\beta\beta}\Delta\beta + \mu\rho\psi_\beta = 0, \tag{3.18b}$$

where

$$0 \leq \mu \leq 1 \quad , \quad \rho = \pm 1. \tag{3.19}$$

Equations (3.18) admit the solution

$$\Delta\alpha = -\mu\rho\,\frac{D_1}{D_3}, \tag{3.20a}$$

$$\Delta\beta = -\mu\rho\,\frac{D_2}{D_3}, \tag{3.20b}$$

where

$$D_1 = \psi_\alpha\,\psi_{\beta\beta} - \psi_\beta\,\psi_{\alpha\beta}, \tag{3.21a}$$

$$D_2 = \psi_\beta\,\psi_{\alpha\alpha} - \psi_\alpha\,\psi_{\alpha\beta}, \tag{3.21b}$$

$$D_3 = \psi_{\alpha\alpha}\,\psi_{\beta\beta} - \psi_{\alpha\beta}^2 \quad . \tag{3.21c}$$

The direction factor ρ is determined such that the first variation $\delta\psi$ in ψ at (α_o, β_o), defined by

$$\delta\psi(\alpha_o,\beta_o) = \psi_\alpha\,\Delta\alpha + \psi_\beta\,\Delta\beta, \tag{3.22}$$

is negative. From (3.20), (3.21), (3.22) we obtain

$$\delta\psi(\alpha_o,\beta_o) = -\mu\rho\,\frac{D_4}{D_3}, \tag{3.23}$$

where

$$D_4 = \psi_\alpha^2\,\psi_{\beta\beta} - 2\,\psi_\alpha\,\psi_\beta\,\psi_{\alpha\beta} + \psi_\beta^2\,\psi_{\alpha\alpha}. \tag{3.24}$$

Therefore $\delta\psi(\alpha_o, \beta_o) < 0$ if

$$\rho = sign\left(\frac{D_4}{D_3}\right). \tag{3.25}$$

Because of this choice, the corrections (3.20) to α_o and β_o become

$$\Delta\alpha = -\mu\,(D_1/D_3)\,sign\,(D_4/D_3), \tag{3.26a}$$

$$\Delta\beta = -\mu\,(D_2/D_3)\,sign(D_4/D_3). \tag{3.26b}$$

The partial derivatives appearing in D_1, D_2, D_3, D_4 are computed at α_o, β_o. Therefore, they are given by

$$\psi_\alpha = -\underline{g}^T(\tilde{\underline{x}})\,\underline{g}^{(k)}, \tag{3.27a}$$

$$\psi_\beta = \underline{g}^T(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-1)}, \tag{3.27b}$$

and

$$\psi_{\alpha\alpha} = \underline{g}^{(k)T} A(\tilde{\underline{x}})\,\underline{g}^{(k)}, \tag{3.28a}$$

$$\psi_{\alpha\beta} = -\underline{g}^{(k)T} A(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-1)}, \tag{3.28b}$$

$$\psi_{\beta\beta} = \Delta\underline{x}^{(k)T} A(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-1)}, \tag{3.28c}$$

in which $A(\underline{x})$ is the Hessian of f computed at $\underline{x}$ where,

$$\tilde{\underline{x}} = \underline{x}^{(k)} - \alpha_o \underline{g}^{(k)} + \beta_o\,\Delta\underline{x}^{(k-1)}. \tag{3.29}$$

In practice, equations (3.28) are not used in order to calculate the second derivatives. Instead one can use the difference scheme

$$\psi_{\alpha\alpha} = \left\{\underline{g}(\tilde{\underline{x}} + \varepsilon_1 \underline{g}^{(k)}) - \underline{g}(\tilde{\underline{x}} - \varepsilon_1 \underline{g}^{(k)})\right\}^T \underline{g}^{(k)} / 2\varepsilon_1, \tag{3.30a}$$

$$\psi_{\beta\beta} = \left\{\underline{g}(\tilde{\underline{x}} + \varepsilon_2 \Delta\underline{x}^{(k-1)}) - \underline{g}(\tilde{\underline{x}} - \varepsilon_2 \Delta\underline{x}^{(k-1)})\right\}^T \Delta\underline{x}^{(k-1)} / 2\varepsilon_2, \tag{3.30b}$$

$$\psi_{\alpha\beta} = \left\{\underline{g}(\tilde{\underline{x}} - \varepsilon_1 \underline{g}^{(k)}) - \underline{g}(\tilde{\underline{x}} + \varepsilon_1 \underline{g}^{(k)})\right\}^T \Delta\underline{x}^{(k-1)} / 2\varepsilon_1$$

$$= \left\{\underline{g}(\tilde{\underline{x}} - \varepsilon_2 \Delta\underline{x}^{(k-1)}) - \underline{g}(\tilde{\underline{x}} + \varepsilon_2 \Delta\underline{x}^{(k-1)})\right\}^T \underline{g}^{(k)} / 2\varepsilon_2. \tag{3.30c}$$

Miele and Cantrell made the choice

$$\varepsilon_1 = \varepsilon / (\underline{g}^{(k)T} \underline{g}^{(k)})^{\frac{1}{2}}, \quad \varepsilon_2 = \varepsilon / (\Delta\underline{x}^{(k-1)T} \Delta\underline{x}^{(k-1)})^{\frac{1}{2}},$$

where ε is a small number.

To perform the search, nominal values must be given to α_o, β_o. Then, one sets $\mu = 1$, and computes $\Delta\alpha$, $\Delta\beta$ from (2.26). If

$\psi(\alpha,\beta) < \psi(\alpha_o, \beta_o)$ the scaling factor $\mu = 1$ is acceptable. If $\psi(\alpha,\beta) > \psi(\alpha_o, \beta_o)$, the previous value of μ must be replaced by some smaller value in the range $o \leqslant \mu \leqslant 1$ until the condition $\psi(\alpha,\beta) < \psi(\alpha_o, \beta_o)$ is met; this can be obtained through bisection. At this point the search step is completed. The values obtained for α,β become the nominal values α_o, β_o for the next search step, and the procedure is repeated until a desired degree of accuracy on α,β is obtained. In the absence of better information, the first step in the search procedure can be made with $\alpha_o = \beta_o = 0$.

The stopping conditions in the two dimensional search are

$$\psi_\alpha^2(\alpha,\beta)+\psi_\beta^2(\alpha,\beta) \leqslant 10^{-10}, \tag{3.31a}$$

and

$$\psi_\alpha^2(\alpha,\beta)+\psi_\beta^2(\alpha,\beta) \leqslant [\psi_\alpha^2(0,0)+\psi_\beta^2(0,0)]\times 10^{-4}. \tag{3.31b}$$

Miele and Cantrell suggested that restarting the process every n or $n+1$ iterations may prove helpful. Since, each time the process is restarted, a step is taken along the negative gradient, a line search is required. The one used by Miele and Cantrell is based upon quasi-linearization with built-in safeguards to ensure the decrease of the function at every step of the iterative search. Let $\phi(\alpha)$ be defined by

$$\phi(\alpha) \triangleq f(\underline{x}^{(k)} - \alpha\, \underline{g}^{(k)}). \tag{3.32}$$

Let α_o be an initial estimate of α. A necessary condition for $\phi(\alpha)$ to have a minimum at α is

$$\phi_\alpha(\alpha) = 0, \tag{3.33}$$

If we apply quasi-linearization to (3.33) we obtain the linear equation

$$\phi_{\alpha\alpha}\,\Delta\alpha + \phi_\alpha = 0, \tag{3.34}$$

where $\phi_{\alpha\alpha}$, ϕ_{α} are computed at α_o. Now, we imbed (3.34) in the more general equation

$$\phi_{\alpha\alpha}\Delta\alpha + \mu\rho\,\phi_{\alpha} = 0 . \tag{3.35}$$

Hence

$$\Delta\alpha = -\mu\rho\,\frac{\phi_{\alpha\alpha}}{\phi_{\alpha}} . \tag{3.36}$$

The direction factor ρ is determined such that

$$\delta\phi(\alpha_o) = \phi_{\alpha}\Delta\alpha \tag{3.37}$$

is negative. From (3.36), (3.37) we find

$$\rho = sign(\phi_{\alpha\alpha}). \tag{3.38}$$

Thus

$$\Delta\alpha = -\mu\, sign(\phi_{\alpha\alpha})\cdot\frac{\phi_{\alpha\alpha}}{\phi_{\alpha}} . \tag{3.39}$$

The scaling factor μ is chosen in a similar way to that which is mentioned in the two dimensional case. The first step in the search procedure can be made with $\alpha_o = 0$, if no better information is available. The stopping conditions are

$$\psi_{\alpha}^2(\alpha) \leq 10^{-10}, \tag{3.40a}$$

and

$$\psi_{\alpha}^2(\alpha) \leq \psi_{\alpha}^2(0)\times 10^{-4} . \tag{3.40b}$$

§(3.3) The Relation between the Methods of Miele and Cantrell and of Fletcher and Reeves

In this section, we will show that the memory gradient algorithm and the Fletcher and Reeves algorithm give identical results for a strictly convex quadratic function (Cantrell (1969)).

By (1.17), (1.16), (3.14),

$$\underline{g}^{(k)T}\underline{g}^{(k)}+\Delta\underline{x}^{(k)T}A\underline{g}^{(k)}=0, \tag{3.41a}$$

$$\underline{g}^{(k)T}\Delta\underline{x}^{(k-1)}+\Delta\underline{x}^{(k)T}A\Delta\underline{x}^{(k-1)}=0. \tag{3.41b}$$

By (3.2), (3.41),

$$\underline{g}^{(k)T}\underline{g}^{(k)}-\alpha^{(k)}\underline{g}^{(k)T}A\underline{g}^{(k)}+\beta^{(k)}\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)}=0, \tag{3.42a}$$

$$-\alpha^{(k)}\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)}+\beta^{(k)}\Delta\underline{x}^{(k-1)T}A\Delta\underline{x}^{(k-1)}=0. \tag{3.42b}$$

Equations (3.42) admit the solutions

$$\alpha^{(k)}=\lambda\Delta\underline{x}^{(k-1)T}A\Delta\underline{x}^{(k-1)}, \tag{3.43a}$$

$$\beta^{(k)}=\lambda\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)}, \tag{3.43b}$$

where λ is a scalar given by

$$\lambda=\frac{\underline{g}^{(k)T}\underline{g}^{(k)}}{[(\underline{g}^{(k)T}A\underline{g}^{(k)})(\Delta\underline{x}^{(k-1)T}A\Delta\underline{x}^{(k-1)})-(\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)})^2]}. \tag{3.44}$$

By (3.43),

$$\beta^{(k)}=\alpha^{(k)}\frac{\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)}}{\Delta\underline{x}^{(k-1)T}A\Delta\underline{x}^{(k-1)}}. \tag{3.45}$$

But

$$\underline{g}^{(k)T}A\Delta\underline{x}^{(k-1)}=\underline{g}^{(k)T}\Delta\underline{g}^{(k-1)}=\underline{g}^{(k)T}\underline{g}^{(k)}, \tag{3.46}$$

and

$$\Delta \underline{x}^{(k-1)T} A \Delta \underline{x}^{(k-1)} = \Delta \underline{g}^{(k-1)T} (-\alpha^{(k-1)} \underline{g}^{(k-1)} + \beta^{(k-1)} \Delta \underline{x}^{(k-2)})$$
$$= \alpha^{(k-1)} \underline{g}^{(k-1)T} \underline{g}^{(k-1)}. \tag{3.47}$$

From (3.45), (3.46), (3.47),

$$\beta^{(k)} = \frac{\alpha^{(k)}}{\alpha^{(k-1)}} \frac{\underline{g}^{(k)T} \underline{g}^{(k)}}{\underline{g}^{(k-1)T} \underline{g}^{(k-1)}}. \tag{3.48}$$

Therefore, for $\beta^{(k)}$ given by (3.48) equation (3.2) takes the form

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} \underline{g}^{(k)} + \frac{\alpha^{(k)}}{\alpha^{(k-1)}} \frac{\underline{g}^{(k)T} \underline{g}^{(k)}}{\underline{g}^{(k-1)T} \underline{g}^{(k-1)}} \Delta \underline{x}^{(k-1)}, \tag{3.49}$$

Let

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} \underline{p}^{(k)} \quad (k \geq 0). \tag{3.50}$$

Then by (3.49),

$$\underline{p}^{(k)} = -\underline{g}^{(k)} + \frac{\underline{g}^{(k)T} \underline{g}^{(k)}}{\underline{g}^{(k-1)T} \underline{g}^{(k-1)}} \underline{p}^{(k-1)} \quad (k \geq 1), \tag{3.51}$$

and the method of Miele and Cantrell and of Fletcher and Reeves produce identical sequences of points if the initial iterates are identical and the one and two-dimensional searches are exact.

§(3.4) The Supermemory Gradient Method

The memory gradient method based on remembering one previous iteration was generalized by Cragg and Levy (1969). The generalization is based on remembering m ($1 \leq m < n$) previous iterations and is called the supermemory gradient method (SMG). The added complication is the need of an (m+1)-dimensional search at each iteration.

Let $\{\underline{x}^{(k)}\}$ be a sequence of points generated from

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta \underline{x}^{(k)} \quad (k = 0, 1, \ldots) \tag{3.52}$$

where

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i^{(k)} \Delta \underline{x}^{(k-i)}, \tag{3.53}$$

and $\alpha^{(k)}$, $\beta_1^{(k)}, \ldots, \beta_m^{(k)}$ are chosen such that

$$f(\underline{x}^{(k+1)}) = \min_{\alpha, \beta_1, \cdots, \beta_m} f(\underline{x}^{(k)} - \alpha \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i \Delta \underline{x}^{(k-i)}), \tag{3.54}$$

and m is a given integer less than n.

Cragg and Levy proved that $\Delta \underline{x}^{(k)}$ given by (3.53) minimizes the first variation of f at $\underline{x}^{(k)}$ given by (3.5) subject to the constraint

$$K = (\Delta \underline{x} - \sum_{i=1}^{m} \gamma_i \Delta \underline{x}^{(k-i)})^T (\Delta \underline{x} - \sum_{i=1}^{m} \gamma_i \Delta \underline{x}^{(k-i)}), \tag{3.55}$$

where $\Delta \underline{x}$ is the nominal displacement and K, γ_i (i = 1,...,m) are constants.

Let

$$\psi(\alpha, \beta_1, \ldots, \beta_m) = f(\underline{x} - \alpha \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i \Delta \underline{x}^{(k-i)}). \tag{3.56}$$

Then, the greatest decrease in the function ψ occurs if the parameters α and β_i satisfy

$$\psi_\alpha = 0, \quad \psi_{\beta_i} = 0 \quad (i = 1, \ldots, m), \tag{3.57}$$

or equivalently

$$\underline{g}^{(k+1)T} \underline{g}^{(k)} = 0 \tag{3.58a}$$

$$\underline{g}^{(k+1)T} \Delta \underline{x}^{(k-i)} = 0 \quad (i = 0, 1, \ldots, m). \tag{3.58b}$$

From (3.53), (3.58) we have

$$\underline{g}^{(k+1)T} \Delta \underline{x}^{(k)} = 0. \tag{3.59}$$

In order to implement the algorithm based on (3.52), (3.53), (3.54), the vectors $\Delta\underline{x}^{(k-1)}$, $\Delta\underline{x}^{(k-2)}$,...,$\Delta\underline{x}^{(k-m)}$ must be available. Since this is not the case for the first k iterations, some assumption concerning $\Delta\underline{x}^{(k-i)}$ or β_i must be made in order to start the algorithm. The simplest one is

$$\beta_i = 0 \quad (i = k+1, \ldots, m; k = 0, \ldots, m). \tag{3.60}$$

For k = o, (3.57) must be replaced by

$$\psi_\alpha = 0, \tag{3.61}$$

which is equivalent to stating that the first iteration is a gradient iteration. For k = 1, (3.57) must be replaced by

$$\psi_\alpha = 0, \ \psi_{\beta_1} = 0, \tag{3.62}$$

so that the second iteration is a memory gradient iteration. Finally for k = m-1, (3.57) is replaced by

$$\psi_\alpha = 0, \psi_{\beta_1} = 0, \ldots, \psi_{\beta_{m-1}} = 0 \tag{3.63}$$

so that the (m-1)th iteration is a supermemory gradient iteration of order m-1.

It is clear from (3.54) that a supermemory gradient method requires an (m+1) dimensional search. The one adopted by Cragg and Levy is based on quasilinearization with built-in safeguards to ensure that the function decreases at every step of the iterative search.

Let the corrections to $\alpha, \beta_1, \ldots, \beta_m$ from arbitrary nominal values $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_m$ be denoted by

$$\delta\alpha = \alpha - \hat{\alpha}, \ \delta\beta_i = \beta_i - \hat{\beta}_i \quad (i = 1, \ldots, m). \tag{3.64}$$

If quasilinearization is applied to (3.57), we obtain the linear algebraic equations

$$\psi_{\alpha\alpha}\delta\alpha+\sum_{j=1}^{m}\psi_{\alpha\beta_j}\delta\beta_j+\psi_\alpha=0, \tag{3.65a}$$

$$\psi_{\beta_i\alpha}\delta\alpha+\sum_{j=1}^{m}\psi_{\beta_i\beta_j}\delta\beta_j+\psi_{\beta_i}=0 \quad (i=1,\ldots,m), \tag{3.65b}$$

where all partial derivatives are evaluated at $(\hat{\alpha}, \hat{\beta}_1,\ldots,\hat{\beta}_m)$. Next we imbed (3.65) in the more general equations

$$\psi_{\alpha\alpha}\Delta\alpha+\sum_{j=1}^{m}\psi_{\alpha\beta_j}\Delta\beta_j+\mu\rho\psi_\alpha=0, \tag{3.66a}$$

$$\psi_{\beta_i\alpha}\Delta\alpha+\sum_{j=1}^{m}\psi_{\beta_i\beta_j}\Delta\beta_j+\mu\rho\psi_{\beta_i}=0 \quad (i=1,\ldots,m), \tag{3.66b}$$

where μ denotes a scaling factor and ρ a direction factor such that

$$0\leqslant\mu\leqslant 1, \quad \rho=\pm 1. \tag{3.67}$$

Equations (3.66) admit the solutions

$$\Delta\alpha=\mu\rho\delta\alpha, \tag{3.68a}$$

$$\Delta\beta_i=\mu\rho\delta\beta_i \quad (i=1,\ldots,m), \tag{3.68b}$$

where $\delta\alpha$, $\delta\beta_i$ $(i = 1,\ldots,m)$ are the solution of (3.65). The direction factor ρ is determined in such a way that the first variation

$$\delta\psi=\psi_\alpha\Delta\alpha+\sum_{i=1}^{m}\psi_{\beta_i}\Delta\beta_i \tag{3.69}$$

is negative. From (3.68), (3.69) we obtain

$$\delta\psi=\mu\rho(\psi_\alpha\delta\alpha+\sum_{i=1}^{m}\psi_{\beta_i}\delta\beta_i). \tag{3.70}$$

Therefore $\delta\psi < 0$ if

$$\rho=-\operatorname{sign}(\psi_\alpha\delta\alpha+\sum_{i=1}^{m}\psi_{\beta_i}\delta\beta_i). \tag{3.71}$$

The partial derivatives appearing in (3.70) are given by

$$\psi_\alpha = -\underline{g}^T(\tilde{\underline{x}})\,\underline{g}^{(k)},$$

$$\psi_{\beta_i} = +\underline{g}^T(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-i)}, \tag{3.72}$$

and

$$\psi_{\alpha\alpha} = \underline{g}^{(k)T} A(\tilde{\underline{x}})\,\underline{g}^{(k)}, \tag{3.73a}$$

$$\psi_{\alpha\beta_i} = -\underline{g}^{(k)T} A(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-i)} \quad (i=1,\ldots,m), \tag{3.73b}$$

$$\psi_{\beta_i\beta_j} = \Delta\underline{x}^{(k-i)T} A(\tilde{\underline{x}})\,\Delta\underline{x}^{(k-j)} \quad (i,j=1,\ldots,m), \tag{3.73c}$$

where $A(x)$ is the Hessian matrix of f computed at

$$\tilde{\underline{x}} = \underline{x}^{(k)} - \hat{\alpha}\,\underline{g}^{(k)} + \sum_{i=1}^{m} \hat{\beta}_i\,\Delta\underline{x}^{(k-i)}. \tag{3.74}$$

If the matrix $A(\underline{x})$ is not explicitly available, the second derivative cannot be computed with (3.73). In this case, we can use the following second order difference scheme:

$$\psi_{\alpha\alpha} \approx \frac{1}{2\delta_0}\left[\underline{g}(\tilde{\underline{x}}+\delta_0\underline{g}^{(k)}) - \underline{g}(\tilde{\underline{x}}-\delta_0\underline{g}^{(k)})\right]^T\underline{g}^{(k)}, \tag{3.75a}$$

$$\psi_{\alpha\beta_i} \approx \frac{1}{2\delta_i}\left[\underline{g}(\tilde{\underline{x}}-\delta_i\Delta\underline{x}^{(k-i)}) - \underline{g}(\tilde{\underline{x}}+\delta_i\Delta\underline{x}^{(k-i)})\right]^T\underline{g}^{(k)}$$

$$\approx \frac{1}{2\delta_0}\left[\underline{g}(\tilde{\underline{x}}-\delta_0\underline{g}^{(k)}) - \underline{g}(\tilde{\underline{x}}+\delta_0\underline{g}^{(k)})\right]^T\Delta\underline{x}^{(k-i)}, \quad (i=1,\ldots,m) \tag{3.75b}$$

$$\psi_{\beta_i\beta_j} \approx \frac{1}{2\delta_i}\left[\underline{g}(\tilde{\underline{x}}+\delta_i\Delta\underline{x}^{(k-i)}) - \underline{g}(\tilde{\underline{x}}-\delta_i\Delta\underline{x}^{(k-i)}\right]^T\Delta\underline{x}^{(k-j)}$$

$$\approx \frac{1}{2\delta_j}\left[\underline{g}(\tilde{\underline{x}}+\delta_j\Delta\underline{x}^{(k-j)}) - \underline{g}(\tilde{\underline{x}}-\delta_j\Delta\underline{x}^{(k-j)})\right]^T\Delta\underline{x}^{(k-i)} \tag{3.75c}$$

$$(i,j=1,\ldots,m).$$

where

$$\delta_0 = \varepsilon/(\underline{g}^{(k)T}\underline{g}^{(k)})^{\frac{1}{2}}, \tag{3.76a}$$

$$\delta_i = \varepsilon/(\Delta\underline{x}^{(k-i)T}\Delta\underline{x}^{(k-i)})^{\frac{1}{2}} \quad (i=1,\ldots,m), \tag{3.76b}$$

and where ε is a small number.

To perform the search, nominal values $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_m$ must be given; then we compute $\delta\alpha, \delta\beta_1, \ldots, \delta\beta_m$ with (3.65) and ρ with (3.71). Next, we set $\mu = 1$ and compute $\Delta\alpha, \Delta\beta_1, \ldots, \Delta\beta_m$ with (3.68) and $\alpha, \beta_1, \ldots, \beta_m$ with (3.64). If

$$\psi(\alpha, \beta_1, \cdots, \beta_m) \leqslant \psi(\hat{\alpha}, \hat{\beta}_1, \cdots, \hat{\beta}_m) \tag{3.77}$$

the scaling factor $\mu = 1$ is acceptable. Otherwise, the previous value of μ must be replaced by some smaller value in the range $0 \leqslant \mu \leqslant 1$ until (3.77) is satisfied. This can be obtained through bisection. At this point, the search step is completed. The values obtained for $\alpha, \beta_1, \ldots, \beta_m$ become the nominal values $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_m$ for the next step, and the procedure is repeated until a desired degree of accuracy on $\alpha, \beta_1, \ldots, \beta_m$ is obtained. In the absence of better information, we set

$$\hat{\alpha} = \hat{\beta}_1 = \cdots = \hat{\beta}_m = 0 \tag{3.78}$$

initially.

The search can be stopped when a predetermined condition is satisfied, such as

$$\chi(\alpha, \beta, \cdots, \beta_m) \leqslant \varepsilon_1, \tag{3.79}$$

and

$$\chi(\alpha, \beta_1, \cdots, \beta_m) \leqslant \varepsilon_2 \chi(0, 0, \cdots, 0), \tag{3.80}$$

where

$$\chi = \psi_\alpha^2 + \psi_{\beta_1}^2 + \cdots + \psi_{\beta_m}^2, \tag{3.81}$$

and where ε_1, ε_2 are small positive numbers.

§(3.5) The Fletcher-Powell Memory Method

The memory gradient method may be regarded as being derived from the method of steepest descent by the introduction of a memory term. For a quadratic function the method is identical to the method of Fletcher and Reeves. This, together with the fact that the method of Fletcher and Powell is usually more rapidly convergent than that of Fletcher and Reeves, lead to the conjecture that it might be possible to derive from the method of Fletcher and Powell, by the introduction of a memory term, a method equivalent to that of Fletcher and Powell itself when applied to quadratic objective functions, and which is more rapidly convergent than the memory gradient method and possibly the method of Fletcher and Powell itself when applied to non-quadratic objective functions.

The preceding consideration led Wolfe (1974) to propose the following algorithm for the unconstrained minimization of f, which he called the general descent algorithm with memory.

Algorithm (3.2)

Given an initial estimate $\underline{x}^{(o)}$ of a minimum $\underline{x}^*$ of f

1. Compute $\alpha^{(o)}$, $\Delta\underline{x}^{(o)}$, $\underline{x}^{(1)}$ from

$$f(\underline{x}^{(o)} - \alpha^{(o)}\underline{p}^{(o)}) = \min_{\alpha} f(\underline{x}^{(o)} - \alpha\underline{p}^{(o)}), \tag{3.82}$$

$$\Delta\underline{x}^{(o)} = -\alpha^{(o)}\underline{p}^{(o)}, \tag{3.83}$$

$$\underline{x}^{(1)} = \underline{x}^{(o)} + \Delta\underline{x}^{(o)}, \tag{3.84}$$

where $\underline{p}^{(o)}$ is a given vector such that $\underline{g}^{(o)T}\underline{p}^{(o)} > 0$, in which $\underline{g}^{(o)} = \underline{g}(\underline{x}^{(o)})$.

2. Set k = 1.

3. Compute $\alpha^{(k)}$, $\beta^{(k)}$, $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)}\underline{p}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)}) = \min_{\alpha,\beta} f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)}), \tag{3.85}$$

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{p}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)}, \tag{3.86}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}, \tag{3.87}$$

where $\underline{p}^{(k)}$ is a given vector such that $\underline{g}^{(k)T}\,\underline{p}^{(k)} > 0$, in which $\underline{g}^{(k)} = \underline{g}(\underline{x}^{(k)})$.

4. Set $k = k+1$ and go to 3.

□

It is easy to prove that the previous algorithm has the descent property, and that

$$\underline{g}^{(k+1)T}\,\underline{p}^{(k)} = 0, \tag{3.88}$$

$$\underline{g}^{(k+1)T}\,\Delta\underline{x}^{(k-1)} = 0, \tag{3.89}$$

$$\underline{g}^{(k+1)T}\,\Delta\underline{x}^{(k)} = 0. \tag{3.90}$$

Consider the application of A(3.2) to the quadratic function given by (1.12). From (3.86), (3.88), (3.89), (3.90), (1.16), (1.17)

$$\alpha^{(k)} = \lambda\,\Delta\underline{x}^{(k-1)T} A\,\Delta\underline{x}^{(k-1)}, \tag{3.91}$$

$$\beta^{(k)} = \lambda\,\underline{p}^{(k)} A\,\Delta\underline{x}^{(k-1)}, \tag{3.92}$$

where

$$\lambda = \underline{g}^{(k)T}\,\underline{p}^{(k)} / [(\underline{g}^{(k)T} A\,\underline{p}^{(k)})(\Delta\underline{x}^{(k-1)T} A\,\Delta\underline{x}^{(k-1)}) - (p^{(k)T} A\,\Delta x^{(k-1)})^2]. \tag{3.93}$$

Hence by (3.90), (3.91), (3.92), (1.16), (1.17),

$$\beta^{(k)} = -\alpha^{(k)}\,\frac{p^{(k)T}\,\Delta\underline{g}^{(k-1)}}{\Delta\underline{x}^{(k-1)T}\,\Delta\underline{g}^{(k-1)}}. \tag{3.94}$$

Since

$$\Delta\underline{x}^{(k-1)T}\,\Delta\underline{g}^{(k-1)} = -\alpha^{(k-1)}\,\underline{p}^{(k-1)}\,\Delta\underline{g}^{(k-1)}, \tag{3.95}$$

(3.94) takes the form

$$\beta^{(k)} = \frac{\alpha^{(k)}}{\alpha^{(k-1)}}\,\frac{\underline{p}^{(k)T}\,\Delta\underline{g}^{(k-1)}}{\underline{p}^{(k-1)T}\,\Delta\underline{g}^{(k-1)}} \quad (k \geq 0). \tag{3.96}$$

So for f defined by (1.12)

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} \left[\underline{p}^{(k)} - \frac{\underline{p}^{(k)T} \Delta \underline{g}^{(k-1)}}{\underline{p}^{(k-1)T} \Delta \underline{g}^{(k-1)}} \frac{\Delta \underline{x}^{(k-1)}}{\alpha^{(k-1)}} \right] \quad (k \geq 1). \tag{3.97}$$

Now, if

$$\underline{p}^{(k)} = -H^{(k)} \underline{g}^{(k)}, \tag{3.98}$$

where $H^{(k)}$ is a positive definite symmetric matrix such that

$$H^{(k+1)} \Delta \underline{g}^{(k)} = \Delta \underline{x}^{(k)} \qquad (k \geq 0), \tag{3.99}$$

then from (3.97), (3.90),

$$\Delta \underline{x}^{(k)} = -\alpha^{(k)} H^{(k)} \underline{g}^{(k)}, \tag{3.100}$$

Hence when applied to a quadratic objective function as defined by (1.12), A(3.2) is equivalent to a quasi-Newton method, in which $H^{(k)}$ are estimates of A^{-1}. The matrix $H^{(k)}$ can be any matrix from Broyden's family. One obvious choice is the DFP matrix given by (2.46).

The preceding considerations lead to the Fletcher-Powell Memory method (FPM), which is given by the following algorithm.

Algorithm (3.3)

Given an initial estimate $\underline{x}^{(0)}$ of a minimum $\underline{x}^*$ of f.

1. Compute $\underline{g}^{(0)}$ from

$$\underline{g}^{(0)} = \underline{g}(\underline{x}^{(0)}).$$

2. Set $\underline{p}^{(0)} = \underline{g}^{(0)}$, $H^{(0)} = I$, where I is the (n x n) unit matrix.

3. Compute $\alpha^{(0)}$, $\Delta \underline{x}^{(0)}$, $\underline{x}^{(1)}$ from

$$f(\underline{x}^{(0)} - \alpha^{(0)} \underline{p}^{(0)}) = \min_{\alpha} f(\underline{x}^{(0)} - \alpha \underline{p}^{(0)}),$$
$$\Delta \underline{x}^{(0)} = -\alpha^{(0)} \underline{p}^{(0)},$$
$$\underline{x}^{(1)} = \underline{x}^{(0)} + \Delta \underline{x}^{(0)}.$$

4. Set $k = 1$.

5. Compute $\underline{g}^{(k)}$, $\Delta\underline{g}^{(k-1)}$ from

$$\underline{g}^{(k)} = \underline{g}(\underline{x}^{(k)}),$$

$$\Delta\underline{g}^{(k-1)} = \underline{g}^{(k)} - \underline{g}^{(k-1)}.$$

6. Compute $H^{(k)}$ from (2.46)

7. Compute $\underline{p}^{(k)}$ from

$$\underline{p}^{(k)} = H^{(k)}\underline{g}^{(k)}.$$

8. Compute $\alpha^{(k)}$, $\beta^{(k)}$, $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)}\underline{p}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)}) = \min_{\alpha,\beta} f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)}),$$

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{p}^{(k)} + \beta^{(k)}\Delta\underline{x}^{(k-1)},$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}.$$

9. Set $k = k+1$ and go to 5.

□

Two search procedures are required - a one dimensional search for estimating α initially, and a two dimensional search for estimating α and β thereafter. Let $\phi(\alpha)$ and $\psi(\alpha,\beta)$ be defined to be

$$\phi(\alpha) = f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)}), \tag{3.101}$$

$$\psi(\alpha,\beta) = f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta x^{(k-1)}). \tag{3.102}$$

In the one dimensional search $\phi_{\alpha\alpha}(\alpha)$ is estimated from

$$\phi_{\alpha\alpha} \approx -[\underline{g}(\underline{x}^{(k)} - (\alpha+\delta)\underline{p}^{(k)}) - \underline{g}(\underline{x}^{(k)} - \alpha\underline{p}^{(k)})]^T\underline{p}^{(k)}/\delta\,, \tag{3.103}$$

in which

$$\delta = \varepsilon/(\underline{p}^{(k)T}\underline{p}^{(k)})^{\frac{1}{2}}, \tag{3.104}$$

and ε is a given small parameter. In the two dimensional search $\psi_{\alpha\alpha}(\alpha,\beta)$, $\psi_{\alpha\beta}(\alpha,\beta)$, and $\psi_{\beta\beta}(\alpha,\beta)$ are estimated from

$$\psi_{\alpha\alpha} \approx -\left[\underline{g}(\underline{x}^{(k)} - (\alpha+\delta_1)\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)}) - \underline{g}(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)})\right]^T \underline{p}^{(k)}/\delta_1, \quad (3.105a)$$

$$\psi_{\alpha\beta} \approx \left[\underline{g}(\underline{x}^{(k)} - (\alpha+\delta_1)\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)}) - \underline{g}(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)})\right]^T \Delta\underline{x}^{(k-1)}/\delta_1, \quad (3.105b)$$

$$\psi_{\beta\beta} \approx -\left[\underline{g}(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + (\beta+\delta_2)\Delta\underline{x}^{(k-1)}) - \underline{g}(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \beta\Delta\underline{x}^{(k-1)})\right]^T \Delta\underline{x}^{(k-1)}/\delta_2, \quad (3.105c)$$

where

$$\delta_1 = \varepsilon/(\underline{p}^{(k)T}\underline{p}^{(k)})^{\frac{1}{2}}, \quad (3.106a)$$

$$\delta_2 = \varepsilon/(\Delta\underline{x}^{(k-1)T}\Delta\underline{x}^{(k-1)}). \quad (3.106b)$$

In the one dimensional search the required value of α^* of α satisfies

$$\phi_\alpha(\alpha^*) = 0, \quad (3.107)$$

and in the two dimensional search the required values α^*, β^* of α,β satisfy

$$\psi_\alpha(\alpha^*, \beta^*) = 0, \quad (3.108a)$$

$$\psi_\beta(\alpha^*, \beta^*) = 0. \quad (3.108b)$$

The quasilinearization method of Miele and Cantrell is equivalent to solving (3.107) and (3.108) by Newton's method with safeguards to prevent divergence. If a third order iterative procedure for solving (3.107) and (3.108) is used instead of Newton's method, with appropriate safeguards to prevent divergence an improvement in computational efficiency is obtained if the increased computational labour per iteration required to obtain third order convergence is more than offset by the reduction in computational

labour due to the smaller number of iterations required for effective convergence.

Wolfe adopted the third order iterative method for solving systems of equations described by Bosarge and Falb (1969), who showed that under certain conditions on $\underline{F}:X \to X$ where X is a Banach space, the sequence $\{\underline{x}^{(k)}\}$ generated from

$$\underline{z}^{(k)} = \underline{x}^{(k)} - \underline{F}'(\underline{x}^{(k)})^{-1}\underline{F}(\underline{x}^{(k)}), \tag{3.109a}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \underline{F}'(\underline{x}^{(k)})^{-1}[\underline{F}(\underline{x}^{(k)}) + \underline{F}(\underline{z}^{(k)})] \quad (k=0,1,\ldots), \tag{3.109b}$$

converges cubically to a zero of $\underline{F}$. In comparison with Newton's method only one additional evaluation of $\underline{F}$ per iteration is required.

Consider the application of (3.109) to the system (3.108). If $\hat{\alpha},\hat{\beta}$ are initial estimates of $\overset{*}{\alpha},\overset{*}{\beta}$ respectively, then $\bar{\alpha},\bar{\beta}$ are computed from $\bar{\alpha} = \hat{\alpha} + \Delta\alpha$, $\bar{\beta} = \hat{\beta} + \Delta\beta$ where

$$\Delta\alpha = -\mu\left(\frac{D_1}{D_3}\right)\operatorname{sign}\left(\frac{D_4}{D_3}\right), \quad \Delta\beta = -\mu\left(\frac{D_2}{D_3}\right)\operatorname{sign}\left(\frac{D_4}{D_3}\right), \tag{3.110}$$

in which D_1, D_2, D_3, D_4 are given by (3.21), (3.24) computed at $\hat{\alpha},\hat{\beta}$ and μ is determined in the same way as described by Miele and Cantrell (1969). The estimates $\bar{\alpha},\bar{\beta}$ correspond to $\underline{z}^{(k)}$ of (3.109a). Next, α,β are computed from $\alpha = \hat{\alpha} + \Delta\bar{\alpha}$, $\beta = \hat{\beta} + \Delta\bar{\beta}$ where

$$\Delta\bar{\alpha} = -\nu\left(\frac{E_1}{D_3}\right)\operatorname{sign}\{(\hat{\psi}_\alpha E_1 + \hat{\psi}_\beta E_2)D_3\},$$

$$\Delta\bar{\beta} = -\nu\left(\frac{E_2}{D_3}\right)\operatorname{sign}\{(\hat{\psi}_\alpha E_1 + \hat{\psi}_\beta E_2)D_3\},$$

in which

$$E_1 = D_1 + (\bar{\psi}_\alpha \hat{\psi}_{\beta\beta} - \bar{\psi}_\beta \hat{\psi}_{\alpha\beta}),$$

$$E_2 = D_2 + (\bar{\psi}_\beta \hat{\psi}_{\alpha\alpha} - \bar{\psi}_\alpha \hat{\psi}_{\alpha\beta}),$$

and $0 < \nu \leq 1$ is such that $\bar{\psi} < \hat{\psi}$. The estimates α,β corresponds to

$\underline{x}^{(k+1)}$ of (3.109b). A similar procedure is used for solving (3.107).

From the numerical evidence presented by Wolfe, we can conclude that the FPM method is superior to the memory gradient method and can frequently be superior to the DFP method, both with respect to convergence rate and computational labour.

CHAPTER 4

SUPERMEMORY DESCENT METHODS FOR UNCONSTRAINED MINIMIZATION

§(4.1) Introduction

Cragg and Levy (1969) have constructed a generalization of the (MG) method of Miele and Cantrell (1969) for unconstrained minimization which, like the (MG) method, is equivalent to the method of Fletcher and Reeves (1964) when applied to quadratic objective functions and is therefore quadratically convergent. Wolfe (1974) has shown that the method of Miele and Cantrell is a special case of a class of methods which he calls descent methods with memory. Wolfe (1974) has considered in particular, the method of Fletcher and Powell (1963) with memory which numerical experience has shown to be much more rapidly convergent, in general, than the method of Fletcher and Powell without memory.

Having regarded the work of Cragg and Levy (1969), who investigated the effect of additional memory terms upon the (MG) method, it would seem natural to enquire what improvement if any is obtained if additional memory terms are introduced into the class of quasi-Newton methods with memory proposed by Wolfe. In this chapter some new results concerning the effect of introducing additional memory terms into the class of quasi-Newton methods with memory are described. It will also be shown that the (MG) method of Miele and Cantrell, the (SMG) method of Cragg and Levy and the quasi-Newton methods with memory are all special cases of a general class of methods which will be called <u>supermemory descent methods</u>. Numerical experience with members of the class of supermemory descent methods is reported and compared with the results obtained by Wolfe (1974).

§(4.2) Supermemory Descent Methods

The following algorithm for estimating an unconstrained minimum $\underline{x}^*$ of f will be called the <u>general supermemory descent algorithm</u>.

Algorithm (4.1) (General supermemory descent algorithm)

Given an initial estimate $\underline{x}^{(o)}$ of a minimum $\underline{x}^*$ of f.

1. Compute $\alpha^{(o)}$, $\Delta\underline{x}^{(o)}$, $\underline{x}^{(1)}$ from

$$f(\underline{x}^{(o)} - \alpha^{(o)}\underline{p}^{(o)}) = \min_{\alpha} f(\underline{x}^{(o)} - \alpha\underline{p}^{(o)}), \tag{4.1}$$

where $\underline{p}^{(o)}$ is a given vector such that $\underline{p}^{(o)T}\underline{g}^{(o)} \neq 0$,

$$\Delta\underline{x}^{(o)} = -\alpha^{(o)}\underline{p}^{(o)}, \tag{4.2}$$

$$\underline{x}^{(1)} = \underline{x}^{(o)} + \Delta\underline{x}^{(o)}. \tag{4.3}$$

2. For $k = 1,\ldots,m-1$ $(m < n)$ compute $\alpha^{(k)}$, $\beta_i^{(k)}$ $(i = 1,\ldots,k)$, $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)}\underline{p}^{(k)} + \sum_{i=1}^{k} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}) = \min_{\alpha,\beta_1,\cdots,\beta_k} f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \sum_{i=1}^{k} \beta_i \Delta\underline{x}^{(k-i)}), \tag{4.4}$$

where $\underline{p}^{(k)}$ is a given vector such that $\underline{p}^{(k)T}\underline{g}^{(k)} \neq 0$,

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{p}^{(k)} + \sum_{i=1}^{k} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}, \tag{4.5}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}. \tag{4.6}$$

3. Set $k = m$.

4. Compute $\alpha^{(k)}$, $\beta_i^{(k)}$ $(i = 1,\ldots,m)$, $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$ from

$$f(\underline{x}^{(k)} - \alpha^{(k)}\underline{p}^{(k)} + \sum_{i=1}^{m} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}) = \min_{\alpha,\beta_1,\cdots,\beta_m} f(\underline{x}^{(k)} - \alpha\underline{p}^{(k)} + \sum_{i=1}^{m} \beta_i \Delta\underline{x}^{(k-i)}), \tag{4.7}$$

where $\underline{p}^{(k)}$ is a given vector such that $\underline{p}^{(k)T}\underline{g}^{(k)} \neq 0$,

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{p}^{(k)} + \sum_{i=1}^{m} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}, \tag{4.8}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)}.$$

5. Set $k = k+1$ and go to 4.

□

We shall consider the application of this algorithm to the strictly convex quadratic function given by (1.12).

By (4.1) - (4.4)

$$\underline{g}^{(k+1)T}\underline{p}^{(k)} = 0 \quad (k = 0, \ldots, m-1), \tag{4.9}$$

$$\underline{g}^{(k+1)T}\Delta\underline{x}^{(k-i)} = 0 \quad (i = 0, \ldots, k;\ k = 0, \ldots, m-1). \tag{4.10}$$

By (4.10) and T(1.10), $\Delta\underline{x}^{(j)}$ (j = o,...,m-1) are A-conjugate, so that

$$\Delta\underline{x}^{(i)T} A\, \Delta\underline{x}^{(j)} = 0 \quad (i \neq j;\ i,j = 0, \ldots, m-1). \tag{4.11}$$

From (1.16), (1.17), (4.5), (4.11), (4.10),

$$\beta_i^{(k)} = \alpha^{(k)} \frac{\underline{p}^{(k)T} A\, \Delta\underline{x}^{(k-i)}}{\Delta\underline{x}^{(k-i)T} A\, \Delta\underline{x}^{(k-i)}} \quad (i = 1, \ldots, k;\ k = 1, \ldots, m-1). \tag{4.12}$$

Substituting into (4.5) from (4.12) for $\beta_i^{(k)}$ and using (4.11) we obtain

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\left[\underline{p}^{(k)} - \sum_{i=1}^{k} \frac{\underline{p}^{(k)T}\Delta\underline{g}^{(k-i)}}{\Delta\underline{x}^{(k-i)T}\Delta\underline{g}^{(k-i)}}\Delta\underline{x}^{(k-i)}\right] \quad (k = 1, \ldots, m-1). \tag{4.13}$$

If $\underline{q}^{(k)}$ (k = o,...,m-1) are defined by

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\underline{q}^{(k)} \quad (k = 0, \ldots, m-1), \tag{4.14}$$

then by (4.13)

$$\underline{q}^{(k)} = \underline{p}^{(k)} + \sum_{i=1}^{k} \frac{\underline{p}^{(k)T}\Delta\underline{g}^{(k-i)}}{\Delta\underline{x}^{(k-i)T}\Delta\underline{g}^{(k-i)}}\alpha^{(k-i)}\underline{q}^{(k-i)} \quad (k = 1, \ldots, m-1), \tag{4.15}$$

and $\underline{q}^{(k)}$ (k = o,...,m-1) are A-conjugate, so that if $\underline{x}^{(m)} \neq \underline{x}^*$ then m < n. Suppose that m < n. Then by (4.7), (4.1)

$$\underline{g}^{(k+1)T}\underline{p}^{(k)} = 0 \quad (k \geq m), \tag{4.16}$$

$$\underline{g}^{(k+1)T}\Delta\underline{x}^{(k-i)} = 0 \quad (i = 0, \ldots, m;\ k \geq m). \tag{4.17}$$

By (1.16), (1.17), (4.17), $\Delta\underline{x}^{(k-i)}$ $(i = 1,\ldots,m;\ k \geq m)$ are A-conjugate, so that

$$\Delta\underline{x}^{(k-i)T} A \Delta\underline{x}^{(k-j)} = 0 \quad (i \neq j;\ i,j = 1,\ldots,m;\ k \geq n). \tag{4.18}$$

Substituting into (1.16) from (4.8) and using (4.18) we obtain from (4.17)

$$\beta_i^{(k)} = \alpha^{(k)} \frac{\underline{p}^{(k)T} A \Delta\underline{x}^{(k-i)}}{\Delta\underline{x}^{(k-i)T} A \Delta\underline{x}^{(k-i)}} \quad (i = 1,\ldots,m;\ k \geq m), \tag{4.19}$$

substituting into (4.8) from (4.19) and using (1.16) we obtain

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\left[\underline{p}^{(k)} - \sum_{i=1}^{m} \frac{\underline{p}^{(k)T} \Delta\underline{g}^{(k-i)}}{\Delta\underline{x}^{(k-i)T} \Delta\underline{g}^{(k-i)}} \Delta\underline{x}^{(k-i)}\right] \quad (k \geq m). \tag{4.20}$$

If $\underline{q}^{(k)}$ $(k \geq m)$ are defined by

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)} \underline{q}^{(k)} \quad (k \geq m), \tag{4.21}$$

then by (4.20)

$$\underline{q}^{(k)} = \underline{p}^{(k)} + \sum_{i=1}^{m} \frac{\underline{p}^{(k)T} \Delta\underline{g}^{(k-i)}}{\Delta\underline{x}^{(k-i)T} \Delta\underline{g}^{(k-i)}} \alpha^{(k-i)} \underline{q}^{(k-i)} \quad (k \geq m). \tag{4.22}$$

Suppose that for some $k \geq m$, $\underline{q}^{(o)},\ldots,\underline{q}^{(k-1)}$ are A-conjugate. This is certainly true when $k = m$. Then by (4.22), since for $j = 1,\ldots,k$

$$\underline{p}^{(k)T} \Delta\underline{g}^{(k-i)} = -\alpha^{(k-j)} \underline{p}^{(k)T} A \underline{q}^{(k-j)},$$

$$\Delta\underline{x}^{(k-j)T} \Delta\underline{g}^{(k-j)} = \alpha^{(k-j)2} \underline{q}^{(k-j)T} A \underline{q}^{(k-j)},$$

we have

$$\underline{q}^{(k)T} A \underline{q}^{(k-j)} = 0 \quad (j = 1,\ldots,k). \tag{4.23}$$

Hence by induction the directions $\underline{q}^{(k)}$ $(k \geq o)$ generated from (4.22) are A-conjugate, and A(4.1) is therefore quadratically convergent.

From (4.16), (4.8), and (4.19),

$$\alpha^{(k)} = \underline{g}^{(k)T}\underline{p}^{(k)} / \left[\underline{p}^{(k)T} A \underline{p}^{(k)} + \sum_{i=1}^{m} \frac{(\Delta \underline{x}^{(k-i)T} A \underline{p}^{(k)})^2}{\Delta \underline{x}^{(k-i)T} A \Delta \underline{x}^{(k-i)}}\right], \quad (k \geq m). \tag{4.24}$$

Hence if $\underline{g}^{(k)T}\underline{p}^{(k)} = 0$ then $\alpha^{(k)} = 0$, and $\Delta\underline{x}^{(k)}$ is a linear combination of $\Delta\underline{x}^{(k-i)}$ $(i = 1,\ldots,k)$. Now if $\underline{q}^{(i)}$ $(i = o,\ldots,k-1)$ are A-conjugate then these vectors span a k-dimensional subspace S of R^n and the unconstrained minimum of f in $\underline{x}^{(o)} + S$ is $\underline{x}^{(k)}$, so since $\Delta\underline{x}^{(k)}$ is a linear combination of $\Delta\underline{x}^{(k-i)}$ $(i = 1,\ldots,k)$, $f(\underline{x}^{(k+1)})$ cannot be less than $f(\underline{x}^{(k)})$. Hence if, for k, $\underline{g}^{(k)T}\underline{p}^{(k)} = 0$, the method breaks down. If however $\underline{g}^{(k)T}\underline{p}^{(k)} \neq 0$, then $\alpha^{(k)} \neq 0$ and since for f defined by (1.12)

$$\Delta f^{(k)} = -\frac{\alpha^{(k)}}{2} \underline{g}^{(k)T}\underline{p}^{(k)},$$

then $\Delta f^{(k)} < 0$ and the descent property of the algorithm is assured.

§(4.3) Supermemory Gradient Methods

In A(4.1) let

$$\underline{p}^{(k)} = \underline{g}^{(k)} \quad (k \geq o), \tag{4.25}$$

corresponding to the supermemory gradient method of Cragg and Levy (1969). Then by (4.4), (4.7), (4.10), (4.17), (4.25), for $k \geq 1$,

$$\underline{g}^{(k)T} \Delta \underline{g}^{(k-i)} = \begin{cases} \underline{g}^{(k)T}\underline{g}^{(k)} & (i = 1) \\ o & (1 < i \leq k) \end{cases}, \tag{4.26}$$

$$\Delta \underline{x}^{(k-i)T} \Delta \underline{g}^{(k-i)} = \underline{g}^{(k-i)T}\underline{g}^{(k-i)} \quad (i = o,\ldots,k). \tag{4.27}$$

So by (4.15), (4.22),

$$\underline{q}^{(k)} = \underline{g}^{(k)} + \frac{\underline{g}^{(k)T}\underline{g}^{(k)}}{\underline{g}^{(k-1)T}\underline{g}^{(k-1)}} \underline{q}^{(k-1)} \quad (k \geq 1), \tag{4.28}$$

and the supermemory gradient method is equivalent to the (FR) method when applied to f defined by (1.12) as stated, but not proved, by Cragg and Levy.

§(4.4) Supermemory Quasi-Newton Methods

As explained in §(2.3), Huang (1970) has given a unified treatment of conjugate gradient and variable metric methods in which a very general class of update formulae is proposed. The update formulae is given by (2.41). In all these updates $H^{(k+1)}$ is obtained from $H^{(k)}$ by adding a correction term $C^{(k)}$ such that

$$H^{(k+1)} = H^{(k)} + C^{(k)}. \tag{4.29}$$

Also

$$H^{(k+1)} \Delta \underline{g}^{(k)} = \rho \, \Delta \underline{x}^{(k)}, \tag{4.30}$$

and

$$C^{(k)} \Delta \underline{g}^{(i)} = 0 \quad (i = 0, 1, \ldots, k-1) \tag{4.31}$$

The search directions for all algorithms corresponding to (2.41) are given by

$$\underline{p}^{(k)} = H^{(k)T} \underline{g}^{(k)} \qquad (k \geq 0). \tag{4.32}$$

Now, if in A(4.1) we define $\underline{p}^{(k)}$ by (4.32) where $\{H^{(k)}\}$ is a sequence of matrices generated from an update formula of Huang's family, then from (4.4), (4.7), (4.10), (4.17), (4.32), (4.30)

$$\underline{g}^{(k)T} H^{(k)} \Delta \underline{g}^{(k-i)} = \rho \, \underline{g}^{(k)T} \Delta \underline{x}^{(k-i)} = 0 \qquad (i = 1, \ldots, k),$$

and

$$\Delta \underline{x}^{(k-i)T} \Delta \underline{g}^{(k-i)} = \alpha^{(k-i)} \, \underline{g}^{(k-i)T} H^{(k-i)} \underline{g}^{(k-i)} \qquad (i = 1, \ldots, k),$$

so by (4.15), (4.22)

$$\underline{q}^{(k)} = H^{(k)T} \underline{g}^{(k)} . \tag{4.33}$$

Hence, A(4.1) with $\underline{p}^{(k)}$ defined by (4.32); where $H^{(k)}$ is updated to satisfy relation of the form (4.29), (4.30), (4.31) is equivalent to the algorithm based on the update (2.41).

In particular Broyden's family given by (2.47), (2.48), (2.46) is a subclass of Huang's family. Broyden's family consists of all symmetric updates of Huang's family with $\rho = 1$. Because of the symmetry of all updates generated by this family the search direction given in (4.32) takes the form

$$\underline{p}^{(k)} = H^{(k)} \underline{g}^{(k)} . \tag{4.34}$$

We therefore obtain the class of <u>supermemory quasi-Newton methods</u> given by the following algorithm.

Algorithm (4.2)

Given an initial $\underline{x}^{(o)}$ of $\underline{x}^*$.

1. Set $H^{(o)} = I$, where I is the (n x n) unit matrix.
2. Compute $\alpha^{(o)}$, $\Delta\underline{x}^{(o)}$, $\underline{x}^{(1)}$ from

$$f(\underline{x}^{(o)} - \alpha^{(o)} H^{(o)} \underline{g}^{(o)}) = \min_{\alpha} f(\underline{x}^{(o)} - \alpha H^{(o)} \underline{g}^{(o)}),$$

$$\Delta\underline{x}^{(o)} = -\alpha^{(o)} H^{(o)} \underline{g}^{(o)},$$

$$\underline{x}^{(1)} = \underline{x}^{(o)} + \Delta\underline{x}^{(o)} .$$

3. Compute $\underline{g}^{(1)}$, $\Delta\underline{g}^{(o)}$ from

$$\underline{g}^{(1)} = \underline{g}(\underline{x}^{(1)}),$$

$$\Delta\underline{g}^{(o)} = \underline{g}^{(1)} - \underline{g}^{(o)} .$$

4. For k = 1,...,m-1:

compute $H^{(k)}$, $\alpha^{(k)}$, $\beta_i^{(k)}$ (i = 1,...,k), $\Delta\underline{x}^{(k)}$, $\underline{x}^{(k+1)}$, $\underline{g}^{(k+1)}$, $\Delta\underline{g}^{(k)}$

from

$$H^{(k)} = H^{(k-1)} + C^{(k-1)},$$

$$f(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{k} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}) = \min_{\alpha,\beta_1,\dots,\beta_k} f(\underline{x}^{(k)} - \alpha H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{k} \beta_i \Delta\underline{x}^{(k-i)}),$$

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)} H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{k} \beta_i^{(k)} \Delta\underline{x}^{(k-i)},$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)},$$

$$\underline{g}^{(k+1)} = \underline{g}(\underline{x}^{(k+1)}),$$

$$\Delta\underline{g}^{(k)} = \underline{g}^{(k+1)} - \underline{g}^{(k)}.$$

5. Set k = m.

6. Compute $H^{(k)}$, $\alpha^{(k)}$, $\beta_i^{(k)}$ (i = 1,...,m), $\Delta x^{(k)}$, $\underline{x}^{(k+1)}$, $\underline{g}^{(k+1)}$, $\Delta\underline{g}^{(k)}$

from

$$H^{(k)} = H^{(k-1)} + C^{(k-1)},$$

$$f(\underline{x}^{(k)} - \alpha^{(k)} H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i^{(k)} \Delta\underline{x}^{(k-i)}) = \min_{\alpha,\beta_1,\dots,\beta_m} f(\underline{x}^{(k)} - \alpha H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i \Delta\underline{x}^{(k-i)}),$$

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)} H^{(k)} \underline{g}^{(k)} + \sum_{i=1}^{m} \beta_i^{(k)} \Delta\underline{x}^{(k-i)},$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \Delta\underline{x}^{(k)},$$

$$\underline{g}^{(k+1)} = \underline{g}(\underline{x}^{(k+1)}),$$

$$\Delta\underline{g}^{(k)} = \underline{g}^{(k+1)} - \underline{g}^{(k)}.$$

7. Set k = k+1 and go to 6.

□

The preceding analysis shows that the supermemory quasi-Newton methods corresponding to A(4.2) with various values of m and various update formulae from Broyden's family are quadratically convergent as are the quasi-Newton methods corresponding to all members of Broyden's family. As shown by Wolfe (1974) however, the introduction of memory in some manner allows for the non-quadratic nature of an arbitrary objective function at points far from the minimum and considerably reduces the number of iterations required to obtain effective convergence in general.

§(4.5) Supermemory Fletcher-Powell Method

If in (2.41) we set $\rho = K_1 = K_4 = 1$, $K_2 = K_3 = 0$, we obtain the update used in the method of Davidon (1959) as modified by Fletcher and Powell (1963). With this update A(4.2), defines for each value $m > 1$, a supermemory Fletcher-Powell method. For $m = 1$, and the (DFP) update, A(4.2) defines the (FPM) method described by Wolfe (1974), who showed that it is in general more rapidly convergent than the (DFP) method without memory, although with the two-dimensional search used, the computational labour required in some cases more than the offsets the saving obtained by the increased rapidity of convergence. Clearly if values of m much larger than 1 are taken, the computational labour required to perform the $(m+1)$-dimensional search at each iteration will be excessive. A reasonable value to take in practice is $m = 2$, and at each iteration a three-dimensional search must be performed. The resulting method will be referred to as the supermemory Fletcher and Powell (SFP) method.

§(4.6) Some Modifications of the Supermemory Gradient Method

The supermemory gradient method (CL) of Cragg and Levy with a given value of m may be modified by replacing the parameters $\beta_i^{(k)}$ $(i = 1,\ldots,\ell;\ell\leq m)$ in (4.8) by the values given by (4.19) to obtain

$$\Delta\underline{x}^{(k)} = -\alpha^{(k)}\left[\underline{g}^{(k)} - \sum_{i=1}^{\ell} \frac{\underline{g}^{(k)T}\Delta\underline{g}^{(k-i)}}{\Delta\underline{x}^{(k-i)T}\Delta\underline{g}^{(k-i)}}\,\Delta\underline{x}^{(k-i)}\right] + \sum_{i=\ell+1}^{m}\beta_i^{(k)}\Delta\underline{x}^{(k-i)} \quad (k\leq m). \tag{4.35}$$

The methods obtained still have quadratic convergence but only $m-\ell+1$ parameters have to be determined at each iteration. As for the (SFP) method a reasonable value of m in practice is $m = 2$, and the methods CL1 and CL2 corresponding to $\ell = 1$ and $\ell = 2$ respectively are obtained.

§(4.7) The Search Technique

The major objection to supermemory descent methods is that the computational labour required to perform the $(m+1)$-dimensional search at each iteration may well offset the saving obtained through the methods requiring fewer iterations to attain effective convergence. Cragg and Levy (1969) adapted the very robust quasilinearization method of Meile and Cantrell (1969) to perform a four-dimensional search. This procedure although very reliable, is computationally expensive because the second derivatives required by the method are estimated by using an approximation which requires two evaluations of the gradient $\underline{g}$ of the objective function for each second derivative. Wolfe (1974) has suggested that the third order procedure for solving systems of nonlinear equations discussed by Bosarge and Falb (1969) with safeguards to prevent divergence similar to those employed by Meile and Cantrell be used to reduce the computational labour. This was found to be more efficient for the (FPM) method than the original search method of Meile and Cantrell. Wolfe used a first order approximation for the second derivatives, which makes the search technique rather sensitive to the step length used in the difference formulae, although producing a great saving in the number of evaluation of $\underline{g}$. As a result of numerical experiments with supermemory descent methods corresponding to $m = 2$, however, it has been found that the original quasilinearization method of Meile and Cantrell as adapted by Cragg and Levy with first order approximation of the second derivatives is insensitive to the step length used in the difference formulae. More explicitly, if

$$\psi(\alpha,\beta_1,\beta_2) = f(\underline{x}^{(k)} - \alpha \underline{p}^{(k)} + \beta_1 \Delta \underline{x}^{(k-1)} + \beta_2 \Delta \underline{x}^{(k-2)}),$$

then the first order approximation in the second derivatives of ψ is given by

$$\psi_{\alpha\alpha} \approx [\psi_\alpha(\alpha+\delta_0, \beta_1, \beta_2) - \psi_\alpha(\alpha, \beta_1, \beta_2)]/\delta_0,$$

$$\psi_{\alpha\beta_1} \approx [\psi_\alpha(\alpha, \beta_1+\delta_1, \beta_2) - \psi_\alpha(\alpha, \beta_1, \beta_2)]/\delta_1,$$

$$\psi_{\alpha\beta_2} \approx [\psi_\alpha(\alpha, \beta_1, \beta_2+\delta_2) - \psi_\alpha(\alpha, \beta_1, \beta_2)]/\delta_2,$$

$$\psi_{\beta_1\beta_2} \approx [\psi_{\beta_1}(\alpha, \beta_1, \beta_2+\delta_2) - \psi_{\beta_1}(\alpha, \beta_1, \beta_2)]/\delta_2,$$

$$\psi_{\beta_1\beta_1} \approx [\psi_{\beta_1}(\alpha, \beta_1+\delta_1, \beta_2) - \psi_{\beta_1}(\alpha, \beta_1, \beta_2)]/\delta_1,$$

$$\psi_{\beta_2\beta_2} \approx [\psi_{\beta_2}(\alpha, \beta_1, \beta_2+\delta_2) - \psi_{\beta_2}(\alpha, \beta_1, \beta_2)]/\delta_2,$$

where

$$\delta_0 = \varepsilon/(\underline{P}^{(k)T}\underline{P}^{(k)})^{\frac{1}{2}},$$

$$\delta_1 = \varepsilon/(\Delta\underline{x}^{(k-1)T}\Delta\underline{x}^{(k-1)}),$$

$$\delta_2 = \varepsilon/(\Delta\underline{x}^{(k-2)T}\Delta\underline{x}^{(k-2)}),$$

in which ε is a given parameter. It would appear from numerical experience that the first order approximation is increasingly adequate for use with the quasilinearization technique of Miele and Cantrell as the number of parameters sought increases. For the three-dimensional search required in a general step of the supermemory descent methods corresponding to $m = 2$ the first order approximation is so satisfactory that the same value of ε could be used for the minimization of all the objective functions listed in Appendix (3).

§(4.8) Numerical Results

To illustrate the numerical performance of supermemory quasi-Newton methods and the modifications of the supermemory gradient method corresponding to (4.35), the ten objective functions listed in Appendix (3) were minimized by using the methods SFP, FPM, DFP, CL, CL1, CL2. In each case convergence was considered to have been attained when $f(\underline{x}) \leqslant 10^{-13}$. The three-dimensional search described in §(4.7) was used for SFP, CL, CL1, and CL2 with $\varepsilon = 10^{-12}$ in each case. The results for FPM and DFP were taken from Wolfe (1974). All calculations were performed in Fortran IV with double precision arithmetic on the IBM 360 computer at St. Andrews.

Table 1 shows the results obtained by using SFP, FPM and DFP methods and Table 2 shows the results obtained by CL, CL1, and CL2 methods. In both tables, n_I is the number of iterations required for effective convergence, n_F is the number of evaluations of f, n_G is the number of evaluations of g, and $N = n_F + n\, n_G$, where n is the number of variables in f. Clearly N is an index of the amount of computational labour required.

§(4.9) Conclusions

From Table 1 we conclude that in 6 cases out of 10 the SFP method is more efficient than the DFP method, and that in 9 cases out of 10 the SFP method is more efficient than the FPM method. These results, together with the fact that the search technique for determining the parameters is reliable, and is successful with the same value of ε for all functions minimized make the SFP method competitive with quasi-Newton methods without memory.

From Table 2 we conclude that reducing the number of parameters by using (4.35) leads in general to a poorer performance than is obtained by using the method of Cragg and Levy.

The work reported in chapter 4 has been submitted to the Journal of Optimization Theory and its Applications, in the form of a paper entitled 'Supermemory Descent Methods for Unconstrained Minimization'.

REFERENCES

AKAIKE, H. 1959 On successive transformation of Probability Distribution and its application to the Analysis of the Optimum Gradient Method. Annals of the Institute of Statistical Mathematics Tokyo, 11, 1-16.

BARNES, J.G.P. 1965 An algorithm for solving nonlinear equations based on the secant method. Comp. J. 8, 66-72.

BECKMAN, F.S. 1960 The solution of linear equations by the conjugate gradient method. Mathematical Methods for Digital Computers. Edited by RALSTON, A. & WILF, H.S. John Wiley & Sons, N.Y. New York.

BIGGS, M.C. 1970 Minimization algorithms making use of non-quadratic properties of objective function. J. Inst. Maths. Appl. 8, 315-327.

BOOTH, A.D. 1957 Numerical Methods. Butterworths.

BOSARGE, W.E. & FALB, P.L. 1969 A multipoint method of third order. JOTA, 4, 156-166.

BOX, M.J. 1966 A comparison of several current optimization methods, and the use of transformations in constrained problems. Comp. J. 9, 67-77.

BROYDEN, C.G. 1965 A class of methods for solving nonlinear simultaneous equations. Maths. Comp. 19, 577-593.

BROYDEN, C.G. 1967 Quasi-Newton methods and their application to function minimization. Maths. Comp. 21, 368-381.

BROYDEN, C.G. 1969 A new method for solving nonlinear simultaneous equations. Comp. J. 12, 1, 95-100.

BROYDEN, C.G. 1970 The convergence of single-rank quasi-Newton methods. Maths. Comp. 24, 365-382.

BROYDEN, C.G. 1970 Recent developments in solving nonlinear algebraic systems. Edited by RABINOWITZ, P. Gordon & Breach. N.Y.

BROYDEN, C.G. 1972 Quasi-Newton methods. Numerical methods for unconstrained optimization. Edited by MURRAY, W. Academic Press.

CANTRELL, J.W. 1969 Relation between the memory gradient method and the Fletcher-Reeves method. JOTA, 4.

CAUCHY, A. 1847 Methode generate pour la resolution des systemes d'equations simultaneis. Comptes Rendus 25.

COURANT, R. 1943 Variational methods for the solution of problems of equilibrium and variations. Bull. Amer. Math. Soc. 49.

CRAGG, E.E. & LEVY, A.V. 1969 Study on a supermemory gradient method for the minimization of functions. JOTA, 4, 191-205.

CROCKETT, J.B. & CHERNOFF, H. 1955 Gradient methods of maximization. Pacific Journal of Mathematics 5, 33-50.

CURRY, H.B. 1944 The method of steepest descent of non-linear minimization problems. Quarterly Journal of Applied Mathematics 2, 258-261.

DAVIDENKO, D.F. 1953 On new method of numerical solution of systems of nonlinear equations. Doklady Akad. Nauk SSSR. NS. 88, 601-602.

DAVIDON, W.C. 1959 Variable metric method for minimization. Argonne National Laboratory, ANL - 5990.

DAVIDON, W.C. 1968 Variance algorithm for minimization. Comp. J. 10, 406-410.

DIEST, F.H. & SEFOR, L. 1967 Solution of systems of nonlinear equations by parameter variation. Comp. J. 10, 78-82.

DIXON, L.C.W. 1971 Quasi-Newton technique generate identical points. Technical Report No 28. Numerical Optimization Centre.

DIXON, L.C.W. 1972 Nonlinear Optimization. The English Universities Press Limited.

DIXON, L.C.W. 1973 Conjugate directions without linear searches. J. Inst. Maths. Appl. 11, 317-328.

DIXON, L.C.W. 1973 Nonlinear Optimization: A survey of the state of the art. Technical Report no 42. Numerical Optimization Centre, Hatfield Polytechnic.

FLETCHER, R. & POWELL, M.J.D. 1963 A rapidly convergent descent method for minimization. Comp. J. 6, 163-168.

FLETCHER, R. & REEVES, C.M. 1964 Function minimization by conjugate gradients. Comp. J. 7, 149-154.

FLETCHER, R. 1965 Function minimization without evaluating derivatives Review. Comp. J. 8, 33-41.

FLETCHER, R. 1969 A class of methods for nonlinear programming with termination and convergence properties. A.E.R.E. (Harwell) T.P. 368.

FLETCHER, R. 1970 A new approach to variable metric algorithms. Comp. J. 13, 317, 322.

FLETCHER, R. 1972 Conjugate direction methods. Numerical method for unconstrained optimization. Edited by MURRAY, W. Academic Press.

FORSYTHE, G.E. & MOTZIN, T.S. 1951 Asymptotic properties of the optimum gradient method. Bull. Amer. Math. Soc. 57, 183.

FREUDENSTEIN, F. & ROTH, B. 1963 Numerical solution of systems of nonlinear equations. J. Assoc. Comp. Maths. 10, 550-556.

GOLDFARB, D. 1970 A family of variable metric methods derived by variational means. Maths. Comp. 24, 23-26.

GREENSTADT, J.L. 1970 Variations of variable metric methods. Maths. Comp. 24, 1-22.

HESTENES, M.R. & STIEFEL, E. 1952 Methods of conjugate gradients for solving linear systems. J. Res. Nat. Bur. Stand. 49, 409-436.

HUANG, H.U. 1970 Unified approach to quadratically convergent algorithms for function minimization. JOTA, 5, 405-423.

HUANG, H.U. & LEVY, A.V. 1970 Numerical experiments on quadratically convergent algorithms for function minimization. JOTA, 6, 269-282.

KOWALIK, J. & OSBORNE, M.R. Method for unconstrained optimization problem. American Elsevier Publishing Company. Inc. New York 1968.

LEVENBERG, K. 1944 A method for the solution of certain nonlinear problems in least squares. Q. Appl. Maths. 2, 164-168.

LUENBERGER, D. 1973 Introduction to linear and nonlinear programming. Addison Wesley Publishing Company.

MARQUARDT, D.W. 1963 An algorithm for least squares estimation of nonlinear parameters. J. Soc. Indust. Appl. Maths. 11, 431-441.

MEILE, A. & CANTRELL, J.W. 1969 Study on memory gradient method for the minimization of functions. JOTA, 5, 147-151.

MEYER, G.H. 1967 On solving nonlinear equations with a one parameter operator embedding. SIAM. J. Numer. Anal. 5, 739-752.

MURTAGH, B.A. & SARGENT, R.W.H. 1970 A constrained minimization method with quadratic convergence. Optimization. Edited by FLETCHER, R. Academic Press, London & New York.

MURRAY, W. 1972 Fundamentals. Numerical methods for unconstrained optimization. Edited by MURRAY, W. Academic Press.

MYERS, G.E. 1968 Properties of the conjugate gradient and Davidon methods. JOTA, 4, 209-219.

NELDER, J.A. & MEAD, R. 1965 A simplex method for function minimization. Comp. J. 7, 308-313.

ORTEGA, J.M. & RHEINBOLDT, W.C. 1970 Iterative solution of nonlinear equations in several variables. Academic Press, New York and London.

PEARSON, J.D. 1969 Variable metric methods for minimization. Comp. J. 12, 171-178.

POWELL, M.J.D. 1962 An iterative method for finding stationary values of a function of several variables. Comp. J. 5, 147-151.

POWELL, M.J.D. 1964 An efficient method of finding the minimum of a function of several variables without calculating derivatives. Comp. J. 7, 155-162.

POWELL, M.J.D. 1965 Minimization of functions of several variables. Numerical Analysis (An Introduction) Edited by WALSH, J. Academic Press. London 1966.

RALL, L.B. 1969 Computational solution of nonlinear operator equations. John Wiley & Sons, N.Y. Inc. 1969.

ROSEN, E.M. 1966 A review of quasi-Newton methods in nonlinear equations solving and unconstrained optimization. Proc. 21st Nat. Conference ACM. Thompson Book Company. Washington, D.C. 1966, 37-41.

SCHINZINGER, R. 1966 Optimization in electromagnetic system design. In LAVI, A. & VOGL, T.P. Recent advances in optimization techniques. John Wiley 1966.

SMITH, H.H. 1962 The automatic computation of minimum likelihood estimate. NCB. Sc. Dept. Report SC 846/MR/40.

SORENSON, H.W. 1969 Comparison of some conjugate direction procedures for function minimization. Journal of Franklin Institute. 288, 421-441.

SPANG, H.A. 1962 A review of minimization techniques for nonlinear functions. SIAM. Rev. 4, 343-365.

SWANN, W.H. 1972 Direct search methods. Numerical methods for unconstrained optimization. Edited by MURRAY, W. Academic Press 1972.

WOLFE, M.A. 1974 A quasi-Newton method with memory for unconstrained function minimization. J. Inst. Maths. Appl. to be published.

WOLFE, P. 1959 The secant method for simultaneous nonlinear equations. Ass. Comp. Maths. Commun. 2 (12), 12-13.

ZANGWILL, W.I. 1967 Minimizing a function without calculating derivatives. Comp. J. 10, 293-296.

ZELEZNIK, F.J. 1968 Quasi-Newton methods for nonlinear equations. J. Assoc. Comp. Maths. 15, 265-271.

APPENDIX (1)

DERIVATION OF THE FORMULA FOR ESTIMATING MINIMA BY CUBIC INTERPOLATION

Let

$$f(x) = ax^3 + bx^2 + cx + d.$$

Let

$$f(o) = f_o,\ f'(o) = g_o,$$
$$f(x_1) = f_1,\ f'(x_1) = g_1.$$

Then

$$d = f_1,\ c = g_o,\ \text{so that}$$
$$f(x) = ax^3 + bx^2 + g_o x + f_o. \tag{1}$$

Hence

$$f'(x) = 3ax^2 + 2bx + g_o, \tag{2}$$
$$f''(x) = 6ax + 2b. \tag{3}$$

The extrema of f(x) are at the points x_1^*, x_2^* which satisfy

$$3ax^2 + 2bx + g_o = 0,$$

namely

$$x_i^* = \frac{-b \pm (b^2 - 3ag_o)^{\frac{1}{2}}}{3a} \quad (i=1,2).$$

The minimum of f(x) satisfies

$$3ax + b > o.$$

Clearly, therefore, the minimum $\hat{x}$ of f(x) is x_1^*, if $a \neq o$. Hence

$$\hat{x} = \frac{-b + (b^2 - 3ag_o)^{\frac{1}{2}}}{3a}. \tag{4}$$

If $a = o$,

$$\hat{x} = \frac{-g_o}{2b} . \tag{5}$$

Then

$$f''(\hat{x}) = 2b,$$

so that if $\hat{x}$ corresponds to a minimum then $b > o$.

Let

$$\gamma \overset{D}{=} -3\frac{(f_o - f_1)}{x_1} = 3(ax_1^2 + bx_1 + g_o), \tag{6}$$

$$\beta \overset{D}{=} \gamma - g_o - g_1 = bx_1 + g_o, \tag{7}$$

$$\alpha^2 \overset{D}{=} \beta^2 - g_o g_1 = x_1^2 (b^2 - 3ag_o). \tag{8}$$

These results are obtained from

$$ax_1^3 + bx_1^2 + g_o x_1 + f_o = f_1, \tag{9}$$

$$3ax_1^2 + 2bx_1 + g_o = g_1. \tag{10}$$

Then

$$x_1[-b + (b^2 - 3ag_o)^{\frac{1}{2}}] = g_o - \beta + \alpha, \tag{11}$$

$$3ax_1^2 = g_o + g_1 - 2\beta. \tag{12}$$

Hence when $a \neq o$,

$$\hat{x} = x_1 \frac{g_o - \beta + \alpha}{g_o + g_1 - 2\beta} . \tag{13}$$

When $a = o$, $g_o - \beta + \alpha = g_o + g_1 - 2\beta = o$, so that the right hand side of (13) is indeterminate. However, in this case, from (5)

$$\hat{x} = -\frac{g_o}{2b} .$$

Now from (6), (7), (8) with $a = o$,

$$\gamma = 3(bx_1 + g_o)$$

$$\beta = bx_1 + g_o$$

$$\alpha = bx_1.$$

Hence

$$g_1 + \alpha + \beta = g_1 + g_o + 2bx_1,$$

and

$$g_1 - g_o + 2\alpha = g_1 - g_o + 2bx_1.$$

Hence

$$x_1\left(1 - \frac{g_1 + \alpha + \beta}{g_1 - g_o + 2\alpha}\right) = x_1\left(\frac{-2g_o}{g_1 - g_o + 2bx_1}\right). \tag{14}$$

Multiplying (10) by x_1 and subtracting from it the multiple of (9) by 2 we find

$$ax_1^3 - g_o x_1 - 2f_o = g_1 x - 2f_1,$$

so if a = o we get

$$(g_o + g_1)x_1 = 2(f_1 - f_o).$$

Hence from (6) and (7)

$$\gamma = 3\,\frac{f_1 - f_o}{x_1} = \frac{3}{2}\,(g_o + g_1)$$

$$\beta = \gamma - g_1 - g_o = \tfrac{1}{2}(g_o + g_1) = bx_1 + g_o.$$

Hence

$$g_1 - g_o = 2bx_1. \tag{15}$$

Using (14) and (15),

$$x_1\left(1 - \frac{g_1 + \alpha + \beta}{g_1 - g_o + 2\alpha}\right) = x_1\,\frac{-2g_o}{4bx_1} = -\,\frac{g_o}{2b} = \hat{x}.$$

Hence when a = o, we may compute $\hat{x}$ from

$$\hat{x} = x_1\left(1 - \frac{g_1 + \alpha + \beta}{g_1 - g_o + 2\alpha}\right). \tag{16}$$

Consider again the case for which $a \neq o$, and $\hat{x}$ is given by (13). We shall show that in this case also, $\hat{x}$ may be completed from (16). This is easily done by noting that the right hand sides of (16) and (13) are equal iff

$$(g_1 - g_o + 2\alpha)(g_o - \beta + \alpha) = (\alpha - \beta - g_o)(g_o + g_1 - 2\beta).$$

Now

$$(g_1 - g_o + 2\alpha)(g_o - \beta + \alpha) = g_1g_o - g_1\beta - g_1\alpha - g_o^2 + g_o\beta + g_o\alpha - 2\alpha\beta + 2\alpha^2,$$

and

$$(\alpha - \beta - g_o)(g_o + g_1 - 2\beta) = g_o\alpha + g_1\alpha - 2\alpha\beta + \beta g_o - \beta g_1 + 2\beta^2 - g_o^2 - g_og_1.$$

Hence

$$\begin{aligned}&(g_1 - g_o + 2\alpha)(g_o - \beta + \alpha) - (\alpha - \beta - g_o)(g_o + g_1 - 2\beta)\\ &= 2g_og_1 + 2\alpha^2 - 2\beta = o \quad \text{by (8)}.\end{aligned}$$

Hence for all values of a, $\hat{x}$, the minimum of f(x), may be computed from (16).

Suppose now that $f(x_o) = f_o$, $f'(x_o) = g_o$ where $x_o \neq o$. To obtain the formula for $\hat{x}$ corresponding to (16) for this case, we introduce the variable u = u(x), where

$$u(x_o) = o, \quad u(x_1) = x_1 - x_o.$$

This is accomplished by setting

$$u = x - x_o.$$

Then

$$u(\hat{x}) = u(x_1)(1 - \frac{g_1 + \alpha + \beta}{g_1 - g_o + 2\alpha}), \tag{17}$$

where if

$$u_o = u(x_o),\ u_1 = u(x_1)$$

$$f_o = f\ (u_o + x_o) = f\ (x_o)$$

$$f_1 = f\ (u_1 + x_o) = f\ (x_1)$$

$$g_o = f'(u_o + x_o) = f'(x_o)$$

$$g_1 = f'(u_1 + x_o) = f'(x_1)$$

and α, β, γ, are defined as before by (6), (7), (8), so that

$$\gamma = -3\,\frac{f_o - f_1}{u_1}\ ,\ \beta = \gamma - g_o - g_1,\ \alpha^2 = \beta^2 - g_o g_1.$$

Hence from (17) we obtain

$$\hat{x} = x_o + (x_1 - x_o)(1 - \frac{g_1 + \alpha + \beta}{g_1 - g_o + 2\alpha}), \tag{18}$$

where

$$\beta = -3\,\frac{f_o - f_1}{x_1 - x_o} - g_o - g_1,$$

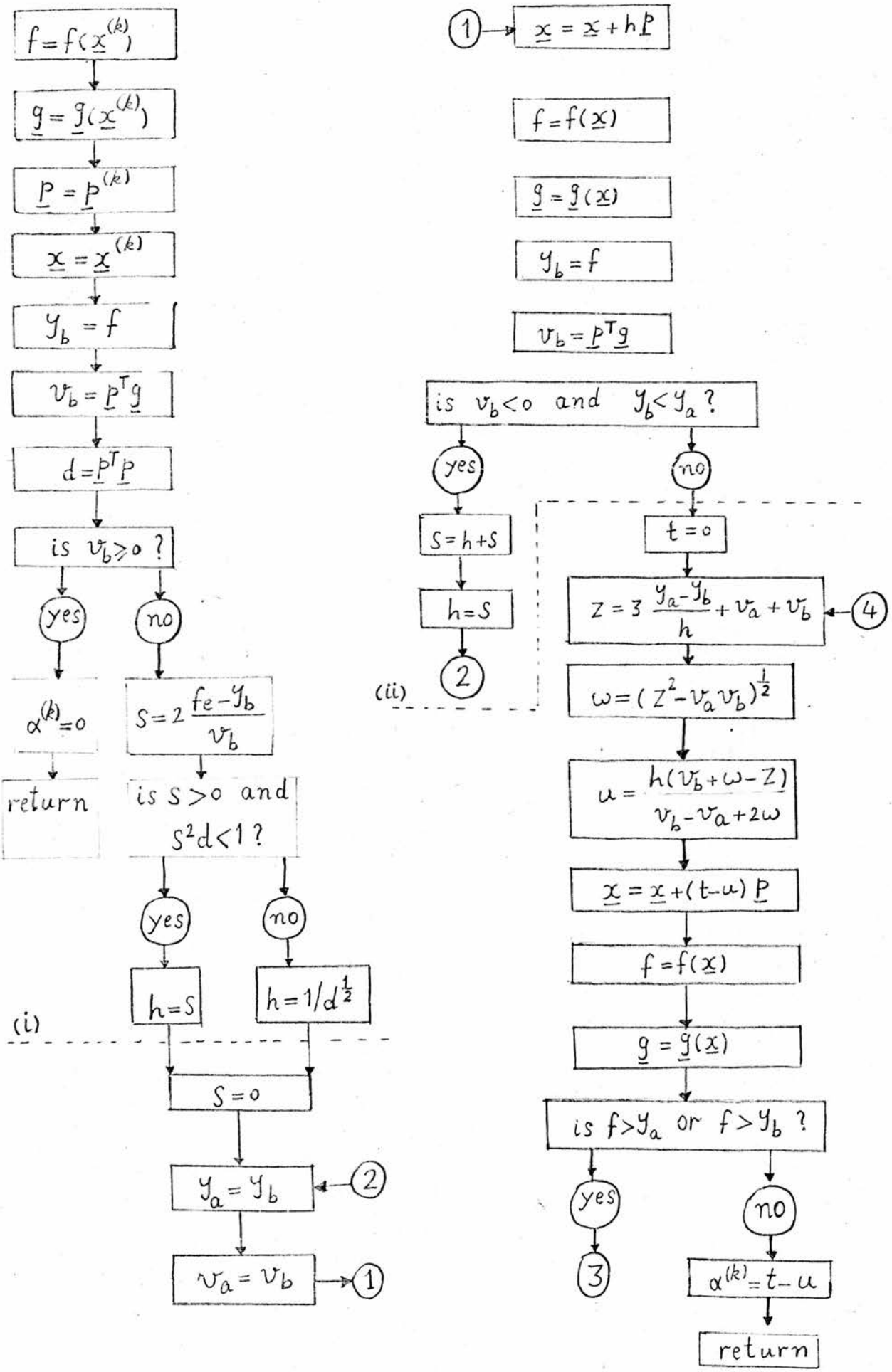
$$\alpha = (\beta^2 - g_o g_1)^{\frac{1}{2}}.$$

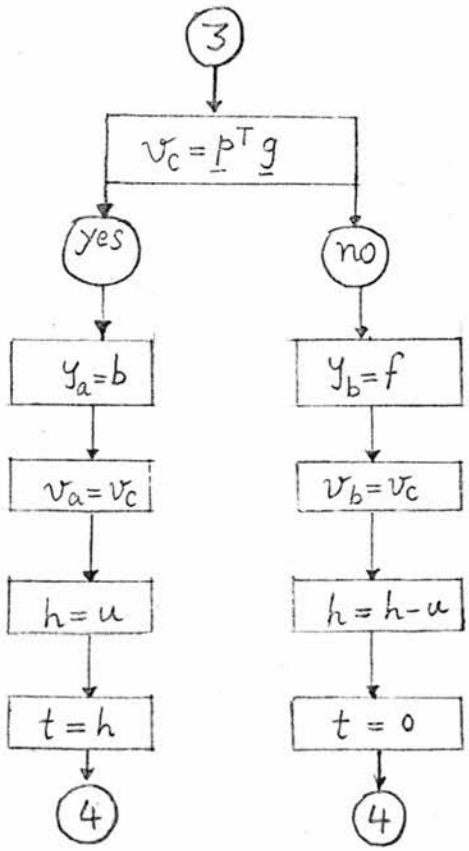
Alternatively, with β defined by

$$\beta = 3\,\frac{f_o - f_1}{x_1 - x_o} + g_o + g_1,$$

$$\hat{x} = x_o + (x_1 - x_o)(1 - \frac{g_1 + \alpha - \beta}{g_1 - g_o + 2\alpha}). \tag{19}$$

Flow Diagram for Cubic Search Technique




3
$v_c = \underline{p}^T \underline{g}$
yes
no
$y_a = b$
$y_b = f$
$v_a = v_c$
$v_b = v_c$
$h = u$
$h = h - u$
$t = h$
$t = 0$
4
4

(iii)

APPENDIX (2)

THE EIGENVALUES OF THE MATRIX $B = (I - \frac{\underline{p}\underline{p}^T}{\underline{p}^T\underline{p}})$.

The characteristic equation is given by

$$|B - \lambda I| = (1 - \lambda)^n |I - \frac{1}{1-\lambda} \frac{\underline{p}\underline{p}^T}{\underline{p}^T\underline{p}}| = 0.$$

But

$$|I - \frac{1}{1-\lambda} \frac{\underline{p}\underline{p}^T}{\underline{p}^T\underline{p}}| = (1 - \frac{1}{1-\lambda} \frac{\underline{p}^T\underline{p}}{\underline{p}^T\underline{p}}) = - \frac{\lambda}{1-\lambda}.$$ (See Pearson (1969), Appendix B).

Hence

$$|B - \lambda I| = -\lambda(1-\lambda)^{n-1}.$$

Thus B has one eigenvalue equal to 0 and (n-1) eigenvalues equal to 1.

APPENDIX (3)

(I) Rosenbrock (1960)

$$f = 100(x_1^2 - x_2)^2 + (1 - x_1)^2.$$
$$x^{(0)} = (-1{\cdot}2,\ 1{\cdot}0)^T,\ x^* = (0,0)^T.$$

(II) Wood [See Pearson (1969).]

$$f = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4) + 10{\cdot}1\{(x_2 - 1)^2 + (x_4 - 1)^2\} + 19{\cdot}8(x_2 - 1)(x_4 - 1).$$
$$x^{(0)} = (-3{\cdot}0,\ -1{\cdot}0,\ -3{\cdot}0,\ -1{\cdot}0)^T,\ x^* = (1,\ 1,\ 1,\ 1)^T.$$

(III) Miele and Cantrell (1969)

$$f = (\exp(x_1) - x_2)^4 + 100(x_2 - x_3)^6 + \{\tan^{-1}(x_3 - x_4)\}^4 + x_1^8.$$
$$x^{(0)} = (1{\cdot}0,\ 2{\cdot}0,\ 2{\cdot}0,\ 2{\cdot}0)^T,\ x^* = (0,\ 1,\ 1,\ 1)^T.$$

(IV) Powell (1962)

$$f = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4.$$
$$x^{(0)} = (3{\cdot}0,\ -1{\cdot}0,\ 0{\cdot}0,\ 1{\cdot}0)^T,\ x^* = (0,\ 0,\ 0,\ 0)^T.$$

(V) Fletcher and Powell (1963)

$$f = 100\left[(x_3 - 10\theta)^2 + (r - 1)^2\right] + x_3^2,$$

where

$$r = \left|(x_1^2 + x_2^2)^{1/2}\right|,$$

$$\theta = \begin{cases} \frac{1}{2\pi}\tan^{-1}\left(\frac{x_2}{x_1}\right) & (x_1 > 0) \\ \frac{1}{2\pi}\tan^{-1}\left(\frac{x_2}{x_1}\right) + \frac{1}{2} & (x_1 < 0). \end{cases}$$

$$x^{(0)} = (-1{\cdot}0,\ 0{\cdot}0,\ 0{\cdot}0)^T,\ x^* = (1,\ 0,\ 0)^T.$$

(VI) Box (1966)

$$f = \sum_{k=1}^{10} \left[(\exp(-kx_1/10) - \exp(-kx_2/10)) - (\exp(-k/10) - \exp(-k)) \right]^2 .$$

$$x^{(0)} = (5{\cdot}0,\ 0{\cdot}0)^T, \quad x^* = (1,\ 10)^T .$$

(VII) Biggs (1971)

$$f = \sum_{k=1}^{10} \left[(\exp(-kx_1/10) - 5\exp(-kx_2/10)) - (\exp(-k/10) - 5\exp(-k)) \right]^2 .$$

$$x^{(0)} = (1{\cdot}0,\ 2{\cdot}0)^T, \quad x^* = (1,\ 10)^T .$$

(VIII) Biggs (1971)

$$f = \sum_{k=1}^{10} \left[(\exp(-kx_1/10) - x_3\exp(-kx_2/10)) - (\exp(-k/10) - 5\exp(-k)) \right]^2 .$$

$$x^{(0)} = (1{\cdot}0,\ 2{\cdot}0,\ 1{\cdot}0)^T, \quad x^* = (1,\ 1,\ 5)^T .$$

(IX) Biggs (1971)

$$f = \sum_{k=1}^{10} \left[(x_3\exp(-kx_1/10) - x_4\exp(-kx_2/10)) - (\exp(-k/10) - 5\exp(-k)) \right]^2 .$$

$$x^{(0)} = (1{\cdot}0,\ 2{\cdot}0,\ 1{\cdot}0,\ 1{\cdot}0)^T, \quad x^* = (1,\ 10,\ 1,\ 5)^T .$$

(X) Dixon (1973)

$$f = (1 - x_1)^2 + (1 - x_{10})^2 + \sum_{i=1}^{9} (x_i^2 - x_{i+1})^2 .$$

$$x^{(0)} = (-2{\cdot}0, \ldots, -2{\cdot}0)^T, \quad x^* = (1, \ldots, 1)^T .$$

Function	SFP				FPM				DFP			
	n_I	n_F	n_G	N	n_I	n_F	n_G	N	n_I	n_F	n_G	N
I	1	142	161	464	2	55	73	201	20	82	82	246
II	6	34	108	466	12	124	196	908	44	294	294	1470
III	4	73	190	833	19	199	333	1531	44	310	310	1550
IV	11	71	248	1063	26	141	284	1277	29	179	179	895
V	1	25	65	220	14	106	205	721	21	84	84	336
VI	1	30	53	136	2	42	57	156	10	64	64	192
VII	1	13	31	75	2	17	31	79	6	24	24	72
VIII	1	19	47	160	9	47	92	323	12	46	46	184
IX	10	48	156	672	15	89	169	765	21	105	105	525
X	16	61	205	2111	20	123	224	2363	61	545	545	5995

Table 1.

Function	CL				CL1				CL2			
	n_I	n_F	n_G	N	n_I	n_F	n_G	N	n_I	n_F	n_G	N
I	1	104	130	364	1	121	146	413	53	148	291	730
II	18	44	149	640	70	224	683	2956	47	171	286	1315
III	6	60	193	832	36	172	428	1884	33	112	211	956
IV	609	803	2909	11636	239	304	1091	4668	73	159	313	1411
V	1	31	71	244	18	59	168	563	29	96	182	642
VI	1	20	40	100	1	40	86	212	39	71	133	337
VII	1	31	64	159	1	18	40	98	30	68	133	334
VIII	1	24	52	180	12	36	105	351	22	49	93	328
IX	19	61	203	873	54	95	317	1363	84	135	267	1203
X	26	65	218	2245	46	107	336	3467	no convergence.			

Table 2.

Appendix (4)

If the objective function is a convex quadratic function, then A(1.2) with the first step taken along the negative gradient, is capable of minimizing $f(\underline{x})$ within n steps. In this case resetting is not necessary. In general $f(\underline{x})$ is not a quadratic function, so $\underline{x}^*$ cannot be found within n steps. However, since A(1.2) is a stable algorithm, i.e., it enjoys the descent property, $\underline{x}^*$ is always approached. In some vicinity S of $\underline{x}^*$, the objective function becomes approximately quadratic. Hence, in order to preserve the superlinear convergence of A(1.2) in this region, A(1.2) should be restarted by a step along the negative gradient. Thus it is necessary after each n steps to take the next step along the negative gradient. In this way we guarantee the superlinear convergence of A(1.2) as soon as we come close enough to the minimum.