The use of non-quadratic models in optimization

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

• A Doctoral Thesis. Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University

Metadata Record: https://dspace.lboro.ac.uk/2134/13652

Publisher: © A. Tassopoulos

Please cite the published version.
This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (https://dspace.lboro.ac.uk/) under the following Creative Commons Licence conditions.

For the full text of this licence, please go to: http://creativecommons.org/licenses/by-nc-nd/2.5/
<table>
<thead>
<tr>
<th>LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY LIBRARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHOR/FILING TITLE</td>
</tr>
<tr>
<td>TASSOPOULOS, A</td>
</tr>
<tr>
<td>ACCESSION/COPY NO.</td>
</tr>
<tr>
<td>192479/02</td>
</tr>
<tr>
<td>VOL. NO.</td>
</tr>
<tr>
<td>CLASS MARK</td>
</tr>
<tr>
<td>Loan copy</td>
</tr>
<tr>
<td>Date due for return:</td>
</tr>
<tr>
<td>14 JUN 1985</td>
</tr>
<tr>
<td>Loan 1 MTH + 2 UNLESS RECALLED U.S.A.</td>
</tr>
<tr>
<td>Date due for return:</td>
</tr>
<tr>
<td>24 JUN 1985</td>
</tr>
<tr>
<td>No Renewal</td>
</tr>
<tr>
<td>Date:</td>
</tr>
<tr>
<td>1 JUL 1986</td>
</tr>
<tr>
<td>4 JUL 1986</td>
</tr>
<tr>
<td>4 JUL 1986</td>
</tr>
</tbody>
</table>
THE USE OF NON-QUADRATIC MODELS
IN OPTIMIZATION

BY

A. TASSOPOULOS

A doctoral thesis submitted in partial
fulfilment of the requirements of the award
of Doctor of Philosophy at Loughborough
University of Technology.

March 1982.

Supervisor: Professor C. Storey,
Department of Mathematics.

© by A. Tassopoulos, 1982.
στους γόνεις μου κωστά και ελέυθ
# CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGEMENTS.</td>
<td>(i)</td>
</tr>
<tr>
<td>ABSTRACT.</td>
<td>(ii)</td>
</tr>
<tr>
<td>LIST OF SYMBOLS.</td>
<td>(iii)</td>
</tr>
<tr>
<td><strong>CHAPTER I : INTRODUCTION</strong></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Statement of the problem and historical review.</td>
<td>2</td>
</tr>
<tr>
<td>1.2 Outline of the thesis.</td>
<td>4</td>
</tr>
<tr>
<td>1.3 The role of modelling in optimization.</td>
<td>5</td>
</tr>
<tr>
<td><strong>CHAPTER II : ONE DIMENSIONAL MINIMIZATION.</strong></td>
<td>10</td>
</tr>
<tr>
<td>2.1 Introduction.</td>
<td>11</td>
</tr>
<tr>
<td>2.2 Brief description of the Algorithm - Formulation of the one-dimensional model.</td>
<td>14</td>
</tr>
<tr>
<td>2.3 Main results.</td>
<td>18</td>
</tr>
<tr>
<td>Appendix.</td>
<td>30</td>
</tr>
<tr>
<td><strong>CHAPTER III: A DERIVATIVE FREE ALGORITHM USING NON-QUADRATIC MODELS.</strong></td>
<td>32</td>
</tr>
<tr>
<td>3.1 Introduction.</td>
<td>33</td>
</tr>
<tr>
<td>3.2 Formulation of the multi-dimensional model.</td>
<td>34</td>
</tr>
<tr>
<td>3.3 Description of the algorithm.</td>
<td>35</td>
</tr>
<tr>
<td>3.4 Constrained minimization of a rational function.</td>
<td>41</td>
</tr>
<tr>
<td>3.5 Numerical results.</td>
<td>43</td>
</tr>
<tr>
<td>3.6 Discussion.</td>
<td>80</td>
</tr>
<tr>
<td>Appendix.</td>
<td>81</td>
</tr>
<tr>
<td><strong>CHAPTER IV : CONJUGATE GRADIENT ALGORITHMS USING A NON-QUADRATIC MODEL WITH EXACT OR INEXACT LINE SEARCHES.</strong></td>
<td>83</td>
</tr>
<tr>
<td>4.1 Introduction.</td>
<td>84</td>
</tr>
<tr>
<td>4.2 The new model and the use of restarts.</td>
<td>88</td>
</tr>
</tbody>
</table>
CHAPTER V: VARIABLE METRIC METHODS BASED ON A NON-QUADRATIC MODEL.

5.1 Introduction. 137
5.2 Properties of the BFGS update using a new model. 141
5.3 The conditioning of the variable metric methods using the proposed updates. 149
5.4 The eigenvalue structure of the updating formula. 158
5.5 Comparison of the condition numbers. 162
5.6 Local convergence theorem. 174
5.7 Computational experiments. 183
5.8 Discussion. 197
Appendix. 198

CHAPTER VI: CONSTRAINED MINIMIZATION TECHNIQUE USING A NEW VARIABLE METRIC UPDATE.

6.1 Introduction. 206
6.2 Statement of the algorithm. 208
6.3 The updating of $E_k$. 212
6.4 Local convergence results. 215
6.5 Computational experience. 220
6.6 Discussion. 224
Appendix. 225

CHAPTER VII: SUMMARY AND CONCLUSIONS.

REFERENCES. 230
ACKNOWLEDGEMENTS

I would like to express my very sincere thanks to my supervisor, Professor C. Storey for his guidance and patience throughout my work.

Many thanks go to Dr. G.A. Evans for many interesting discussions during the first years of my research.

My gratitude to my parents is beyond words for their unconditional love and support of my education.

Finally I am grateful to Mrs. B. Wright for her excellent typing.
This thesis is concerned with the development and implementation of a number of numerical algorithms for solving finite dimensional nonlinear optimization problems. Most of the currently used optimization methods use directly or indirectly a local quadratic representation of the objective function.

The need to introduce non-quadratic models arises from the fact that objective functions may not be represented adequately by quadratic functions. Although complicated models can represent general nonlinear functions more accurately than quadratics, they are more difficult to deal with both analytically and numerically. Therefore there must be a compromise between the generality of the model and the ease with which it can be used in the context of optimization.

Various non-quadratic models have been tested in both gradient and nongradient methods. Sophisticated gradient techniques such as conjugate gradient and variable metric methods have been used with the proposed models and compared with the traditional methods over a variety of standard test functions.

One measure of the complexity of an optimization problem is its size, measured in terms of the number of unknown variables. Despite the fact that computation has been seriously dealt with and good algorithms have been proposed in the past insufficient work has been done on very large problems. Therefore most of the standard test problems were used in their generalized form to gain insight into the efficiency of the use of the non-quadratic models as the dimension of the problem increases.

The numerical results show that the use of non-quadratic models is beneficial in most of the problems considered especially when the dimensionality of the problem increases.
LIST OF SYMBOLS

\[ g(x) = \left( \frac{\partial f(x)}{\partial x_1}, \ldots, \frac{\partial f(x)}{\partial x_n} \right)^T \]  
- Gradient vector of \( f(x) \)

\[ B(x) = G(x) = \begin{pmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \ldots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \ldots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{pmatrix} \]  
- Hessian matrix of \( f(x) \)

\[ E_k \]  
- \( k \)th approximate to \( \nabla^2 L(x^*, \lambda^*) \)

\[ H_k \]  
- \( k \)th approximate to \( G^{-1}(x^*) \)

\( x^* \)  
- a local minimum of \( f(x) \)

\[ g_k = g(x_k) \]  
- gradient vector at \( x_k \)

\[ f_k = f(x_k) \]  
- function value at \( x_k \)

Scalars are denoted by lower case Greek letters, vectors are denoted by lower case Latin letters and matrices are denoted by upper case Latin letters.
CHAPTER I

INTRODUCTION
1.1 STATEMENT OF THE PROBLEM AND HISTORICAL REVIEW.

The general class of finite optimization problems consists of all those problems where the maximum or the minimum of a function of a number of variables is required. Sometimes the variables will be restricted in the values they may take.

Optimization problems are as old as mankind. Throughout history men have employed elaborate methods to help them reach the best decision. They have put their faith in either proverbs or sophisticated rules to take some of the decisions out of living. Optimization problems have applications in almost every branch of science and technology. Therefore there are well known problems which have involved scientists for centuries. The Greek mathematician Apollonius, author of the eight-volume work "Conic Sections", in the third century B.C. was perhaps the earliest scientist to be involved with optimization problems. In the fifth volume of his work he is concerned with the problem of finding the lines of maximum and minimum length that can be drawn from a point to a conic.

However, it is only in the seventeenth century that the progress of calculus brought the major advances in optimization. Newton and Leibnitz showed that the first derivative of a function vanishes at a critical point. Maclaurin used the second derivative of a function to distinguish between maxima and minima. Lagrange proposed an algorithm for equality constrained problems.

The extensive use and perfection of electronic computers during the last twenty years have stimulated interest in optimization.
Several numerical methods have been developed and good algorithms have been proposed to deal with the complicated optimization problems that have arisen.

A wide range of optimization problems occur in practice and these are not all considered in this thesis. Attention has been restricted to the development and implementation of numerical algorithms for solving finite dimensional nonlinear optimization problems.

To be more precise the following problems will be considered in this thesis:

\[
\begin{align*}
\text{Minimize} & \quad f(x), \quad x \in \mathbb{R}^n \\
\text{Minimize} & \quad f(x), \quad x \in \mathbb{R}^n \\
\text{Subject to} & : \quad c_i(x) = 0, \quad i = 1, \ldots, m' \quad \text{(P1)} \\
& \quad c_i(x) \geq 0, \quad i = m'+1, \ldots, m \quad \text{(P2)}
\end{align*}
\]

where \( f(x) \) is a real-valued function and \( c_i \) are constraint functions.

Note that only minimization problems are considered but there is no loss in generality since clearly:

\[
\text{maximum } \{f(x)\} = -\text{minimum } \{-f(x)\}.
\]

The function \( f \) and the constraints \( c_i(x) \) are assumed to be twice continuously differentiable, unless otherwise stated. If \( f \) has several minima then our aim will be to find a local minimum.

A brief description of the work included in this thesis follows in the next section.
1.2 OUTLINE OF THE THESIS.

Numerical methods for optimization are generally iterative in nature. They require a starting approximation to the minimum point, an initial direction and the distance to move along this direction to find the next point.

These methods can be divided into two classes according to whether or not derivatives are evaluated. Methods which do not use derivatives are termed non-gradient or derivative free methods.

The second and third chapters of the thesis are concerned with non-gradient methods. In the second chapter a one-dimensional optimization method is examined. A rational model (which interpolates function values only) is used instead of a quadratic (Powell, 1964) or a cubic model (Davidon, 1959). Previous work with rational models is also presented. Some new theoretical results are given at the end of the chapter.

The third chapter extends the use of rational models to more than one dimension. It follows a method proposed by Winfield (1973) in which a quadratic model is interpolated through a data table. Several rational models have been used and compared - using the same algorithm - with the quadratic model. The effect that different parameters of the algorithm have on the rate of convergence is presented in tables and graphs at the end of the chapter.

Methods which evaluate derivatives are called gradient methods and are dealt with in the next three chapters.

Chapter four treats different existing conjugate gradient algorithms using as a base a nonlinear scaling of a quadratic function. It consists of two parts. Firstly, an accurate line
search is employed and secondly, it deals with conjugate gradient methods using a relaxed line search. In both of these parts a quadratic model is compared with two nonquadratic models using the same conjugate gradient method. The computational results which are included give more insight into the use and importance of the various parameters used.

In the fifth chapter a Variable Metric Method is considered for the solution of problem (P1). A nonquadratic model is compared with the standard quadratic one using the BFGS update. A theoretical analysis of the condition number of the update used is given. A hybrid algorithm based on these results is used but unfortunately the numerical results were not as promising as the theory.

The sixth chapter deals with the constrained optimization problem (P2). It is based on an algorithm recently proposed by Powell (1977). It is a natural extension of the method used in chapter five for the constrained problem.

Finally the discussion which follows in chapter seven summarizes the important aspects and the difficulties associated with the use of nonquadratic models. Some concluding remarks and suggestions for future research are also included in the last chapter.

1.3 THE ROLE OF MODELLING IN OPTIMIZATION.

Models are widely used by scientists for replacing a phenomenon in an unfamiliar field by one with which the model user is better acquainted. There are many cases where studies or experiments on
the prototype cannot be carried out at all without the use of models. In most of the cases models enable the users to carry out studies on the prototype using less computational effort than would have been needed if the prototype had been used.

Optimization is an area where by using models, complicated objective functions can be simplified, relevant properties can be extracted and effects can be scaled up or down to obtain the appropriate level of detail.

The simplest model of an objective function can be obtained by using the first order terms of a series expansion of the function around a point. This is an inexact linear model of the function having a low level of detail.

An optimization method which uses gradient information only to obtain the next point from that currently used is that of steepest descent. It was first described by A. Cauchy in 1847. Although easy to implement it generally converges slowly because the direction of steepest descent and the direction to the minimum are nearly perpendicular. An example is shown in Figure 1.

This is to be expected since the "steepest descent" depends not only upon the function being minimized but also on the metric used.

If the function $f$ has continuous second partial derivatives then by Taylor's expansion around a point $x$ a quadratic approximation to $f$ is obtained by using second order terms. Therefore second derivatives information can be employed for the construction of the next point. If the function $f$ is itself a quadratic function and the Hessian matrix is positive definite, then the knowledge of
FIGURE 1.
$G^{-1}\nabla f$ at the point $x$ would suffice to determine the minimum. However, a representation of a general nonlinear function by a quadratic is not always successful away from the minimum. A more realistic modelling of a general function should be sought amongst functions with better approximation ability.

The degree of model detail must be determined before the elements of the models are defined and the model is developed. The objective function structure and behaviour is an important factor in determining the detail of the model. Such information combined with the modelling purpose define the complexity of the model.

Another factor that should be taken into consideration is how much usefulness the model has. The condition which a model should satisfy to be considered as a useful model, is that the model's experiments should simulate the prototype's behaviour using less computational effort. Too little information and model detail may lead to a simple and meaningless model. A too complex model, on the other hand, could equally violate the condition for a useful model because of the extensive analysis and computations such a detailed model requires.

The issue of model detail is a conflict of the model realism versus simplicity and practicality. Very detailed models require an extensive understanding of the objective function, much data and more computations needed in the modelling development and solution algorithm.

Since in optimization modelling has a local nature the
following question arises: Do detailed models always provide more detailed information about the behaviour of the objective function without requiring excessive computational effort? There is no general answer to this question. Sometimes they do provide more information about the behaviour of the objective function but in general this depends on several factors such as the amount of computation involved and the nature of the numerical problems.

Finally, trying to balance modelling detail with our degree of understanding of the general properties of objective functions and the available modelling data, several functions from the class of rational functions have been chosen and used as models. These models are either the ratio of two quadratics or the nonlinear scaling of a quadratic and include quadratic functions as special cases.
CHAPTER II

One-Dimensional Minimization
2.1 **INTRODUCTION.**

The problem considered in this chapter is that of minimizing a function of one variable $f(x)$, i.e. the problem (P1) where $x \in \mathbb{R}$. For the study of convergence the function is assumed to be unimodal and differentiable in a subset of $\mathbb{R}$.

The problem of minimization in one dimension is not only important in its own right but also an essential part of many multi-dimensional methods. Two classes of method are available for single variable minimization:

(i) The function $f(x)$ is approximated by a known function which can be analyzed to find its minimum. Usually the approximating function is a polynomial of a low degree. Second order polynomials are used when function values only are available, and cubic ones when first derivatives are also available.

(ii) The size of the interval which contains the minimum is reduced by evaluation of the function $f(x)$ at suitable search points. Although several techniques for placing these points are available, they are all basically similar. Such techniques are termed search methods and two well known methods are Fibonacci search and Golden Section search.

The main objections to approximation methods for single variable optimization are the following:

(i) Functions which are discontinuous cannot be minimized with accuracy since approximating such functions with a continuous one gives a poor fit to a discontinuous curve.
(ii) When a minimum lies outside the range of the points used to determine the polynomial coefficients the accuracy of the minimization is poor. This is because the algorithm might fail to converge or may converge to the wrong type of turning point.

The currently used methods are combinations of these two classes of methods. Such an example is the algorithm of Davies, Swann and Campey (1964) which combines a search method which produces three equally spaced points that span the minimum and an approximation method by using a quadratic fitted to the function \( f(x) \) at these three points.

A combination of Davies, Swann and Campey's method and Powell's method, used in Powell (1964) can be found in Himmelblau (1972), p.46. Numerical results are also reported which show that this hybrid algorithm gives better results than the individual algorithms.

Brent's method combines golden-section search and parabolic interpolation in the same way as bisection and linear interpolation can be combined to find the zero of a function. More details can be found in Brent (1972), p.72 and an Algol 60 procedure in p.79.*

A cubic interpolation method due to Davidon (1959) uses function values and first derivatives and is widely used in algorithms for multi-dimensional unconstrained minimization.

A more recent idea proposed by Storey (1978) is to use a rational approximation to the one-dimensional function. The motivation arises in approximation theory where rational function

* The NAG library uses similar hybrid methods due to Gill and Murray (1973).
approximations are accepted as having certain advantages over
the corresponding polynomial approximations, (see Cheney (1966)).

A preliminary implementation of this idea was carried out
by Moufti (1977). The implementation lacks robustness and an
improved version was proposed by Otolorin (1979).

The possibility of using nonpolynomial interpolation functions
received some attention from other authors as well. Björstad
and Nocedal (1979) analyze the rate of convergence of an algorithm
based on the interpolating function
\[ \frac{ax^2 + bx + c}{(dx + 1)^2} \]

This function is the one-dimensional restriction of the conic
model function suggested by Davidon (1980).

A rational model with a linear denominator has been suggested
by Barzilai and Ben-Tal (1980) which has been analyzed in Barzilai

Recently Yanai et al. (1981) proposed a class of rational
functions and reported some numerical results in favour of some
members of this class.

In this chapter the algorithm proposed by Otolorin (1979),
is examined from a convergence point of view, following the work
of Tamir (1976) and the algorithm is proved to possess a
superlinear rate of convergence.
2.2  **BRIEF DESCRIPTION OF THE ALGORITHM - FORMULATION OF THE ONE-DIMENSIONAL MODEL.**

A one-dimensional search based on rational function interpolation can be described briefly as follows.

Let \( x \) be a scalar variable and \( f(x) \) a differentiable objective function to be minimized. A stationary point of \( f(x) \) occurs at \( x^* \), where \( x^* \) is a solution of the equation:

\[
f'(x) = 0.
\]

Let \( x_j, x_{j+1}, \ldots, x_{j+N-1} \) be \( N \) approximations to \( x^* \) ordered by decreasing function values and let

\[
R^n_{m-1}(x) = \frac{P_n(x)}{Q_{m-1}(x)} \quad (N = n + m)
\]

be the unique rational function which interpolates the function \( f(x) \) through these \( N \) points:

\[
R^n_{m-1}(x_{j+k}) = f(x_{j+k}), \quad k = 0, 1, \ldots, N-1.
\]

A new point \( x_{j+N} \) is then selected as a new approximation to \( x^* \) by satisfying the equations:

\[
(R^n_{m-1}(x_{j+N}))' = 0.
\]

If the new point \( x_{j+N} \) and the function value \( f(x_{j+N}) \) satisfy the convergence criteria, the procedure is terminated. If the convergence criteria are not satisfied the procedure continues with the new set of points:

\[
x_{j+1}, x_{j+2}, \ldots, x_{j+N-1}, x_{j+N},
\]

where the point with the highest function value has been discarded.
This algorithm will be called the Sequential Rational Fitting Algorithm (SRFA). The algorithm uses a preprocessor to select five points spanning the minimum. This was also found to be an important part of the algorithm and improved robustness (see G. Evans (1979)). Its revised form is given by Otolorin (1979). A description of the special model used is given below.

The model selected from the class of the rational functions is the ratio of two quadratics, i.e.

\[ R(x) = \frac{a_1 x^2 + a_2 x + a_3}{x^2 + a_4 x + a_5} \]  \hspace{1cm} (2.2.1)

The turning points of \( R(x) \) and their nature are easily determined analytically. For the construction of the initial rational model an estimate of the minimum is used to find a set of five different points, (five is the number of the coefficients of the model). Then the requirement that \( R(x) \) interpolates \( f(x) \) at points \( x_i, \ i = 1, 2, 3, 4, 5 \) leads to the linear equations:

\[ f(x_i) = R(x_i) \quad , \quad i = 1, \ldots, 5 \]  \hspace{1cm} (2.2.2)

In general the five unknown coefficients, \( a_i, \ i = 1, \ldots, 5 \) may be computed by solving the linear equations (2.2.2). A new point is then determined as the minimum of \( R(x) \), if it exists. The method of finding \( x_{\min} \) is given in the Appendix. If there are no turning points in the rational model \( R(x) \), the point of inflection of (2.2.2) is selected (see Appendix). The new point \( x_{\min} \) replaces the point \( x_k, \ k \in \{i : i = 1, \ldots, 5\} \) which gives the highest function value.
The new set of five points is then used to obtain another rational model. The procedure is stopped when specified stopping criteria are satisfied. The procedure can be divided into four parts:

1. Preprocessor.
2. Rational function approximation.
3. Finding the stationary points of $R(x)$ and their nature.
4. Convergence criteria.

If the objective function behaves locally like either

$$\frac{a_1 x^2 + a_2 x + a_3}{x + a_4}$$

or as a quadratic, then the choice of the coefficient of $x^2$ in the denominator of the rational model (2.2.1) as unity can cause the coefficients $a_1, a_2, a_3, a_4, a_5$ in (2.2.1) to become very large or even overflow. Large coefficients can cause complete numerical cancellation in calculating the local minimum and hence a complete loss of significant digits, (see Evans, (1979)). In practice it was found that only a small proportion of the functions tested used the quadratic fit but without this safeguard the algorithm was extremely unreliable.

The first main part of the algorithm consists of the preprocessor. An effective preprocessor is necessary to yield the five points being fitted so that a good approximation to the minimum is found. However, the preprocessor may give points which span a minimum in such a way that the spanning is spurious.
Alternatively the points may fail to span a minimum and yet yield a good approximation to the minimum. In Otolorin's work the algorithm proceeds in both cases and the minimum point given by the five points is found.

The second part of the algorithm has already been described. The third part is given in the Appendix. The last part is the convergence criteria used. A combination of stopping criteria on both function values and points was employed, i.e.

\[ |f(x_k) - f(x_{k+1})| < \varepsilon^2 |f(x_{k+1}) + 0.1| \]

\[ |x_k - x_{k+1}| < \varepsilon |x_k + 0.1| \]

where \( \varepsilon \) is never smaller than half machine precision. These conditions are in spirit of Himmelblau (1972) and avoid problems when the converged result in \( x \) or \( f \) is either:

(i) Very large, when the correct number of significant digits are yielded due to the scale of \( x_k \) on the right hand side.

(ii) Very small or zero, when the \( x_k \) scale becomes useless and the constant 0.1 takes over to yield the correct number of absolute digits, or

(iii) Neither, when a dual effect of 0.1 and \( x_k \) also gives the correct number of significant digits.

Preliminary studies and numerical tests on rational models, (see Moufti, (1977) and Otolorin (1979)) have proved encouraging enough to consider a theoretical approach to the convergence rate of the algorithm.
The main results of this analysis are included in the next section.

2.3 **MAIN RESULTS.**

The concept of convergence used in this section is now quoted:

**Definition (2.3.1):**

Let the sequence \( \{x_k\} \) converge to \( x^* \) in the normed space \( \mathbb{R}^n, \| \cdot \| \). Then the order of convergence of \( \{x_k\} \) is \( p \), where

\[
p = \sup \left\{ q \in \mathbb{R} : \limsup_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^q} \in [0, \infty], \text{ and } q \geq 1 \right\}.
\]

Let \( L \) be a positive scalar so \( f(x) \) is unimodal in \( D \subset \mathbb{R} \), where

\[
D = \left\{ x \in \mathbb{R} : |x - x^*| \leq L \right\}
\]

and assume that \( f(x) \) satisfies the following conditions:

(i) \( f^{(2)}(x) \neq 0 \) for all \( x \in D \). Since \( x^* \) is a minimum point of \( f(x) \) this is equivalent to \( f^{(2)}(x) > 0 \), for all \( x \in D \).

(ii) \( f^{(N)}(x^*) \neq 0 \).

(iii) \( f^{(N+1)} \) is continuous on \( D \).

(iv) The three constants \( C_0, C_1, C_2 \) can be defined as follows for all \( x \in D \).

\[
f^{(2)}(x) > C_0
\]
\[
\left| \frac{f^{(N)}(x)}{N!} \right| \leq C_1
\]

\[
\left| \frac{f^{(N+1)}(x)}{(N+1)!} \right| \leq C_2
\]

Then the two main theorems will state the convergence of the sequence \( \{x_k\} \) to the minimum \( x^* \).

**Theorem (2.3.1):**

Let \( R_{m-1}^n(x) \) be a rational function which interpolates the function \( f(x) \) through the points \( x_j, x_{j+1}, \ldots, x_{j+N-1} \), in \( D \). Under the assumptions (i), (ii) and (iii) the equation

\[
\left( R_{m-1}^n(x) \right)' = 0
\]

(2.3.1)

has a real root in \( D \).

**Proof:**

Define the function

\[
G(x) = \prod_{k=0}^{N-1} (x - x_{k+j})
\]

and form the function \( H(t) \):

\[
H(t) = F_1(x) G(t) - F_1(t) G(x)
\]

(2.3.2)

where

\[
F_1(x) = F(x) Q_{m-1}^n(x) \; , \; F(x) \text{ is the interpolation error},
\]

and

\[
R_{m-1}^n(x) = \frac{P_n(x)}{Q_{m-1}^n(x)} = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0
\]

\[
= \frac{a_0 x^{m-1} + b_{m-2} x^{m-2} + \ldots + b_0}{x^{m-1} + x^{m-2} + \ldots + 1}, \; N = n+m.
\]

(It can always be assumed that \( Q_{m-1}^n(x) > 0 \).)
From equation \( F(x) = f(x) - R_{m-1}^n(x) \) it follows that:

\[
F(x) = f(x) - \frac{P_n(x)}{Q_{m-1}(x)}
\]

so

\[
F(x) Q_{m-1}(x) = f(x) Q_{m-1}(x) - P_n(x).
\]

Differentiation of equation (2.3.2) gives:

\[
H^{(N)}(t) = F_1(x) G^{(N)}(t) - F_1^{(N)}(t) G(x)
\]

(2.3.3)

It is obvious that

\[
P_n^{(N)}(t) = 0 \quad \text{and} \quad Q_{m-1}^{(N)}(t) = 0,
\]

and

\[
F_1^{(N)}(t) = f^{(N)}(t) Q_{m-1}(t).
\]

Therefore, equation (2.3.3) becomes:

\[
H^{(N)}(t) = (N!) F_1(x) - G(x) f^{(N)}(t) Q_{m-1}(t).
\]

(2.3.4)

Using the mean value theorem an \( \xi \) is found such that:

\[
x_j < \xi < x_{N-1+j}
\]

and

\[
H^{(N)}(\xi) = 0.
\]

Letting \( t = \xi \) in equation (2.3.4) we finally get:

\[
* \text{Provided } m \leq 2. \]
\[ F(x) = \frac{f^{(N)}(\xi)}{N!} G(x) \tag{2.3.5} \]

So the function \( f(x) \) can be rewritten as follows:

\[ f(x) = R(x) + \frac{f^{(N)}(\xi)}{N!} G(x) \tag{2.3.6} \]

where \( \xi \) lies in the interpolation interval \( x_j, x_{j+1}, \ldots, x_{j+N-1} \).

A result due to Ralston (1965) states that:

\[ \frac{1}{(N)!} \frac{d}{dx} \left[ f^{(N)}(\xi) \right] = \frac{1}{(N+1)!} f^{(N+1)}(n) \]

where \( n \) lies in the interpolation interval \( x_j, x_{j+1}, \ldots, x_{j+N-1} \).

Differentiation of (2.3.6) gives:

\[
R'(x) = f'(x) - \frac{f^{(N)}(\xi)}{(N)!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} (x - x_{j+k}) - \\
- \frac{f^{(N+1)}(n)}{(N+1)!} \prod_{k=0}^{N-1} (x - x_{j+k}) \tag{2.3.7}
\]

If it can be shown that the function \( R'(x) \), has alternative signs at the points \( (x^* - L) \) and \( (x^* + L) \), then the theorem will follow.

Since \( x^* \) is the minimum point of \( f(x) \), \( x \in D \), \( f'(x) \) can be expressed as follows:

\[ f'(x) = f'(x) - f'(x^*) = (x - x^*) f^{(2)}(\theta) \tag{2.3.8} \]

where \( \theta \in D \).
Substituting \( x = x^* - L \) in (2.3.7) and using (2.3.8) yields:

\[
R'(x^* - L) = - L f^{(2)}(\theta) - \frac{f^{(N)}(\xi)}{N!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} (x^* - L - x_{j+k})
\]

\[
- \frac{f^{(N+1)}(n)}{(N+1)!} \prod_{k=0}^{N-1} (x^* - L - x_{j+k}) =
\]

\[
= - L f^{(2)}(\theta) \left\{ 1 + \left[ \frac{f^{(N)}(\xi)}{N!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} (x^* - L - x_{j+k})}{L f^{(2)}(\theta)} \right] +
\]

\[
+ \frac{f^{(N+1)}(n)}{(N+1)!} \prod_{k=0}^{N-1} (x^* - L - x_{j+k}) \right\} \right) =
\]

\[
= - L f^{(2)}(\theta) M.
\quad (2.3.9)
\]

Since \( L > 0 \) and \( f^{(2)}(\theta) > 0 \) \( \forall \theta \in D \) it has to be proved that \( M > 0 \). This implies:

\[
\frac{1}{L f^{(2)}(\theta)} \left\{ - \frac{f^{(N)}(\xi)}{N!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} (x^* - L - x_{j+k}) -
\]

\[
\frac{f^{(N+1)}(n)}{(N+1)!} \prod_{k=0}^{N-1} (x^* - L - x_{j+k}) \right\} \leq 1
\]

\[
M \leq |M| \leq \frac{1}{L c_0} \left[ c_1(N) (2L)^{(N-1)} + c_2 (2L)^{(N)} \right] \quad (2.3.10)
\]
The right hand side of the inequality (2.3.10) could be less than or equal to 1, depending on the length of the interval $L$. So, assuming that $L$ is small enough, then

$$R'(x^* - L) < 0.$$ 

In the same way, it is proved that

$$R'(x^* + L) > 0,$$

which proves the theorem.

**Theorem (Convergence)** (2.3.2):

Suppose that the above theorem holds. Let $x_{j+N}$ be a real root of $R'(x)$ in $D$. Then the sequence \( \{x_{j+1}, \ldots, x_{j+N}\} \) converges to the minimum point $x^*$ and the order of convergence is superlinear.

**Proof:**

Let $x = x_{j+N}$ in equation (2.3.7):

$$f'(x_{j+N}) = \frac{f^{(N)}(\theta_1)}{(N)!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} \frac{(x_{j+N} - x_{j+k})}{x_{j+N} - x_{j+k}} +$$

$$+ \frac{f^{(N+1)}(\theta_2)}{(N+1)!} \prod_{k=0}^{N-1} (x_{j+N} - x_{j+k})$$

(2.3.11)

where $\theta_1, \theta_2 \in D$. Let $e_{j+N} = x_{j+N} - x^*$, $j = 1, 2, \ldots$. From equation (2.3.8)

$$f'(x_{j+N}) = e_{j+N} f^{(2)}(\theta_N).$$

Using the above result, equation (2.3.11) can be rewritten:
This implies:

\[ |e_{ij+N}| \leq \left( \frac{C_1}{C_0} \frac{(2L)^{N-1}}{L} (N) + \frac{C_2}{C_0} \frac{(2L)^N}{L} \right) |e_{ij+N}| + \]

\[ + \frac{C_1}{C_0} (N) \max_{0<k\leq N-1} |e_{ij+k}|^{N-1} + \frac{C_2}{C_0} \max_{0<k\leq N-1} |e_{ij+k}|^N. \quad (2.3.12) \]

Under the assumption that the interval \( L \) is sufficiently small (2.3.12) can be rewritten:

\[ |e_{ij+N}| \leq W \max_{0<k\leq N-1} |e_{ij+k}|^{N-1} \quad (2.3.13) \]

where \( W = 2 \left( \frac{C_1}{C_0} \frac{N}{C_0} + \frac{C_2}{C_0} \frac{L}{C_0} \right) \).

Let \( \hat{e}_i = |e_i| W^{1/(N-1)} \) and \( \alpha = W^{1/(N-1)} \).

The constants \( C_0, C_1, C_2 \) and the interval \( L \) are selected in such a way so that \( \alpha < 1 \).

Then it is implied that

\[ \hat{e}_N \leq \max \left\{ \hat{e}_{ij}^{N-1}, \hat{e}_{ij}^{N-1}, \ldots, \hat{e}_{ij}^{N-1} \right\} \leq \alpha^{N-1}. \]
For \( r \gg 1 \)

\[
\hat{e}_{rN} \leq \hat{e}_{(r-1)N} \leq \hat{e}_{(r-2)N} \leq \ldots \leq \alpha^{(N-1)^r}.
\]

So, in general:

\[
\hat{e}_k \leq \alpha^{(N-1)k/N}
\]

\[
|e_k| \leq L \cdot w^{1/N-1} \cdot \alpha^{r(N,k)}
\]

(2.3.14)

where \( r(N,k) = (N-1)^k/N \).

The definition of the average rate of convergence (Luenberger, 1972 p.129) will be required next.

**Definition.**

The average order of convergence of a sequence \( \{e_k\} \) converging to \( * \) is the infimum of the numbers \( q > 1 \) such that:

\[
\lim_{k \to \infty} |e_k - *|^{1/q^k} = 1.
\]

(2.3.15)

Hence equation (2.3.14) can be rewritten as:

\[
|e_k| < B \alpha^q
\]

(2.3.16)

where \( B = L \cdot w^{1/N-1} \) and \( q = (N-1)^{1/N} \).

Equation (2.3.16) implies:

\[
|e_k|^{1/q^k} < (B^{1/q^k})^\alpha.
\]

Using expression (2.3.15) and the fact that \( \alpha < 1 \), in the
last relation:

\[
\lim_{k \to \infty} |e_{k+1}^{1/q} - e_{k}^{1/q}| \leq (B^{1/q}^k) \alpha \leq 1 .
\]

Therefore \( q = (N-1)^{1/N} \) is the average rate of convergence.

Now the order of convergence is shown to be superlinear.

From equation (2.3.11), knowing that

(i) \( (x_{j+N} - x_{j+k}) = (e_{N+j} - e_{k+j}) \) and

(ii) \( f'(x_{j+N}) = e_{j+N} f^{(2)}(\theta_N) \) where \( \theta_N \in [x_{j+N}, x] \)

hence:

\[
e_{N+j} f^{(2)}(\theta_N) = \frac{f^{(N)}(\theta_1)}{N!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} (e_{j+N} - e_{j+k}) + \]

\[
\frac{f^{(N+1)}(\theta_2)}{(N+1)!} \prod_{k=0}^{N-1} (e_{j+N} - e_{j+k}) .
\]  

(2.3.17)

Suppose that \( e_{j+k} \neq 0 \) \( k = 0, 1, \ldots, N-1 \) then equation (2.3.17) becomes:

\[
e_{N+j} f^{(2)}(\theta_N) = \prod_{k=1}^{N-1} e_{j+k} \left\{ \frac{f^{(N)}(\theta_1)}{N!} \sum_{\ell=0}^{N-1} \prod_{k=0}^{N-1} \left( e_{j+N} - e_{j+k} \right) \right\}
\]

\[
+ \sum_{\ell=1}^{N-1} \left( e_{j+N} - e_{j+\ell} \right) \prod_{k=1}^{N-1} \left( e_{j+N} - e_{j+k} \right) \]

\[
+ \frac{f^{(N+1)}(\theta_2)}{(N+1)!} \left( e_{j+N} - e_j \right) \prod_{k=1}^{N-1} \left( e_{j+N} - e_{j+k} \right) .
\]  

(2.3.18)
It is assumed that the limit \( \frac{e_j}{e_{j-1}} \) exists and is finite:

\[
\lim_{j \to \infty} \frac{e_{j+N}}{e_{j-1+N}} = e .
\]

By equation (2.3.18) it follows that:

\[
e^{N+j} = A_{N+j+1} \prod_{k=1}^{N-1} e^{j+k}
\]

and

\[
\lim_{j \to \infty} A_{N+j} = A \neq 0
\]

From (2.3.18) and (2.3.19) it follows that:

\[
\frac{e^{N+j}}{e^{N+j-1}} = A_{N+j+1} \prod_{k=1}^{N-1} e^{j+k}
\]

\[
\frac{e^{j+N-2}}{e^{j+N-1}} = e^{-1}
\]

and therefore

\[
\lim_{j \to \infty} \frac{e^{N+j}}{e^{N+j-1}} = 0 .
\]

This completes the proof of the convergence theorem.

It is clear that the order of convergence of the SRFA, increases as the degree of the rational function increases.

The following table gives the changes of the average rate of convergence (second column) as the degree of the interpolating polynomial or equivalently the number of coefficients less one used in the rational interpolating function increase.
Table 2.1 shows that a rational function of the form (2.2.1) has the same average rate of convergence as an interpolating polynomial of the fourth order.

Remark: If Thiele's interpolation formula is expressed as:
(Ralston, 1965)

\[ f(x) = F(x) + R(x) = \frac{P_n(x)}{Q_{m-1}(x)} + F(x), \quad N = m+n. \]

Therefore the error of the rational interpolation is:

\[ F(x) = \frac{G(x)}{N! Q_{m-1}^2(x)} \frac{d^N}{dx^N} \left\{ Q_{m-1}^2(x) f(x) \right\}_{x=\xi}, \quad (2.3.20) \]

where \( \xi \) is in the interval spanned by \( x_0, \ldots, x_{N-1} \). The number of points needed to determine the coefficients of the rational function, is \( N \).

The degree of the polynomial \( Q_{m-1}^2(x) \) is \( 2(m-1) \). In the case of polynomials \( P(x) \) and \( Q(x) \) of the same degree we have that:
N = n + m = 2n > 2(n-1).

So, equation (2.3.20) becomes

\[ F(x) = \frac{G(x)}{N!} f^{(N)}(\xi) \]

which is the same as equation (2.3.6).
Appendix

The turning points of \( R(x) \), given by (2.2.1), are given by:

\[
\frac{dR(x)}{dx} = \frac{(a_1a_4 - a_2)x^2 - 2(a_3 - a_1a_5)x + a_2a_5 - a_3a_4}{(x^2 + a_4x + a_5)^2} = 0
\]

(2.A.1)

setting: \( A = a_2a_5 - a_3a_4 \), \( B = a_3 - a_1a_5 \) and \( C = a_1a_4 - a_2 \)

in (2.A.1) yields:

\[
Cx^2 - 2Bx + A = 0
\]

(2.A.2)

Therefore, there are turning points if

\[
B^2 - AC > 0
\]

(2.A.3)

and no turning points if:

\[
B^2 - AC < 0.
\]

The case \( B^2 - AC = 0 \) would result from the ratio of two linear forms and therefore does not arise.

If \( B^2 - AC > 0 \), the turning points are given by:

\[
\theta_1 = \frac{B + \sqrt{B^2 - AC}}{C} \quad \text{and}
\]

\[
\theta_2 = \frac{B - \sqrt{B^2 - AC}}{C} \quad \text{if} \quad C \neq 0
\]

or by \( \theta_3 = A/2B = -a_4/2 \), if \( C = 0 \).

Two basic cases arise:

(i) If the roots of the denominator of \( R(x) \) are complex, the minimum of \( r(x) \) is \( \theta_1 \). This is because the denominator is positive for all \( x \) and \( R(x) \) is continuous for all real \( x \). Thus,
\[
\frac{d^2 R(x)}{dx^2} > 0
\]
for a minimum and this condition is satisfied by \( \theta_1 \).

(ii) If the roots of the denominator of \( R(x) \) are real then consider the function \( g(z) \), where:

\[
z = B - Cx
\]

\[
g(z) = \frac{(R(x) - a_1)/C^2}{\frac{B - Cx}{C^2(x^2 + a_4x + a_3)}} = \frac{z}{(z-p)(z-q)}.
\]

The turning points of \( g(z) \) are the same as those of \( R(x) \) and are given by

\[
\frac{dg(z)}{dz} = 0
\]

i.e.

\[
z^2 - pq = 0.
\]

Therefore the turning points are \( z = + \sqrt{pq} \) and \( z = - \sqrt{pq} \), if \( pq > 0 \) which is the condition \( B^2 - AC > 0 \). Now, the point \( z = - \sqrt{pq} \) gives a minimum of \( g(z) \), therefore the minimum point of \( R(x) \) is given by:

\[
z = - \sqrt{pq} = B - Cx
\]

which is

\[
x = (B + \sqrt{B^2 - AC})/C = \theta_1 \quad \text{if} \quad C \neq 0.
\]

If \( C \) is zero, there is only one turning point which is a minimum if \( B > 0 \), namely, \( A/2B \).

If there are no turning points, \( x_{\text{min}} \) is taken as the turning point of \( (2.\text{A.2}) \), i.e. \( (B/C) \).
CHAPTER III

A DERIVATIVE FREE ALGORITHM,

USING NONQUADRATIC MODELS
3.1 **INTRODUCTION.**

In this chapter an algorithm is developed for solving problem (P1), which does not require analytic expressions for the gradient of a multi-dimensional function. It may therefore be used when there is no analytic expression for the objective function, e.g. when the function is simply the measured value of cost when a plant is operated. Even when analytic expressions for the objective function are available it may still be the case that analytic expressions for the gradients are either difficult or impossible to obtain. Therefore a derivative-free algorithm is often used in such cases.

It is certainly more convenient for the user to code only the function subroutine and convenience for the user is particularly important in engineering applications.

A new approach to the problem (P1) which was presented by Winfield (1973), consists of a direct search algorithm. In his method every new trial point is located after direct minimization of a quadratic model in n-space. Winfield calls his algorithm the Sequence of Quadratic Models algorithm and emphasizes the quadratic termination of his method.

In this chapter the same idea as the one proposed by Winfield (1973) is followed but more general functions than quadratics, namely rational functions, are employed. The two different models are compared using the same pilot programme and their performance on several low dimensionality problems is given. The important role of several parameters and their influence on the algorithms are also presented in tables and graphs at the end of this chapter.
3.2 FORMULATION OF THE MULTI-DIMENSIONAL MODEL.

The method which is developed in this chapter is based on rational models of the form:

\[ R(x) = \frac{1}{2} x^T Q x + d^T x + c }{ q^T x + 1 } \quad (3.2.1) \]

where

- \( Q \) is an \( n \times n \) symmetric matrix,
- \( c \) is a scalar,
- \( q, x, d \) are \( n \) dimensional vectors.

The relation (3.2.1) can be rewritten as:

\[(x^T Q \ c \ d^T \ q^T) \begin{bmatrix} x/2 \\ 1 \\ x \\ -xR(x) \end{bmatrix} = R(x) . \quad (3.2.2)\]

Define \( \mathbf{a}^T \equiv (x^T Q \ c \ d^T \ q^T) \)

\[ y \equiv \begin{bmatrix} x/2 \\ 1 \\ x \\ -xR(x) \end{bmatrix}. \]

Then (3.2.2) can be written in vector form as:

\[ \mathbf{a}^T y(x) = R(x) . \quad (3.2.3) \]

The requirement that \( R(x) \) interpolates \( f(x) \) at points \( x_i \),

\[ i = 1, 2, \ldots, N \]

can be met by choosing the elements of \( Q, d, c, q \) to satisfy the equations:

\[ f(x_i) = \frac{1}{2} x_i^T Q x_i + d_i^T x_i + c }{ q^T x_i + 1 } , \quad i = 1, 2, \ldots, N . \quad (3.2.4) \]
In general the N unknowns of Q, d, c, q may be computed by solving the N linear equations (3.2.4). If the objective function \( f(x) \) is a rational function of the form (3.2.1), then only

\[
N = \frac{(n+1)(n+2)}{2} + n
\]

evaluations of the function \( f(x) \) are required to construct \( R(x) \), which will be an exact model of \( f(x) \).

For large \( n \) the \( N \) evaluations required by direct rational interpolation are approximately half the \( n(n+3) \) evaluations required by Stewart's method, (Stewart, 1967).

The fact that rational interpolation in \( n \)-space locates the minimum of a rational function of the form of (3.2.1) in fewer function evaluations than Stewart's method suggests its use as part of an algorithm for locating the minimum of an arbitrary function.

This method will be called the Sequence of Rational Models (SRM) algorithm and it will be based on repeated rational interpolations in \( n \)-space.

3.3 DESCRIPTION OF THE ALGORITHM.

The algorithm is presented in a conceptual form. The notation used is as follows:

- \( M \) : the number of grid points,
- \( N \) : the number of unknowns,
- \( M = N \) : the grid is exact,
- \( M > N \) : the grid is overdetermined,
- \( f(x) : f : \mathbb{R}^n \rightarrow \mathbb{R} \) the objective function to be minimized with respect to \( x \in \mathbb{R}^n \).
The system of linear equations (3.2.4) can be rewritten as:

\[ AY = R \quad (3.3.1) \]

where

\[
Y = \begin{bmatrix}
y(x_1) \\
y(x_2) \\
\vdots \\
y(x_M)
\end{bmatrix}, \quad R = \begin{bmatrix}
R(x_1) \\
R(x_2) \\
\vdots \\
R(x_M)
\end{bmatrix}, \quad A = \begin{bmatrix}
a(x_1) \\
a(x_2) \\
\vdots \\
a(x_M)
\end{bmatrix}
\]

\[ x_1 \] = the initial point,
\[ L \] : a large positive number (e.g. \(10^{40}\))
\[ EPS \] : a small positive number (e.g. \(10^{-6}\)).

Given the above notation, the general algorithm can be stated as follows:

1. Evaluate \(f(x_i)\) at an initial grid of points \(x_i\), \(i = 1, 2, \ldots, M\).

2. Select as the base point of the grid, \(x_b\), the point at which \(f(x_i)\) has the smallest value.

3. Order the points by increasing Euclidean distance from the basepoint, i.e. \(x_1 = x_b\) and \(x_M\) the point farthest from the basepoint.

4. Calculate \(Q, d, c, q\) from (3.3.1) where \(A, Y, R\) are defined by the model and data grid used. If \(Y\) is singular then set \(x_1 = x_b\) and go to (1). Else continue.

5. Find the maximum and the minimum of the components of the model. If the ratio \(\frac{\text{max coefficient}}{\text{min coefficient}}\) is greater than \(L\), set \(x_1 = x_b\) and go to (1). Then use as a model a quadratic function (i.e. the numerator of (3.2.1)) instead of (3.2.1). Else continue.
(6) Define a region of validity $S$ of the rational model

$$R(x) = \frac{\frac{1}{2} x^T Q x + d^T x + c}{q^T x + 1}.$$ 

For the first model and after every success in locating a new basepoint, let $S$ be a sphere of radius $a_1 \, \|x_M - x_b\|$ centred at the basepoint, where $a_1 \in (0, 2)$.

(7) Choose the next trial point by minimizing the rational model $R(x)$, i.e. $x' = \min_{x \in S} R(x)$.

(8) Compute $x = x_b + x'$ and evaluate $f(x)$.

(9) If $f(x) \leq f(x_b)$ then $x$ becomes a new basepoint. Else $x_b$ remains as the basepoint and the volume of $S$ is reduced by the factor $a_2$ where $a_2 \in (0, 0.9)$. If $x$ becomes a new basepoint all the data is reordered by distance from the new basepoint. If $x_b$ remains the old basepoint then $x$ becomes one of the $M-1$ points closest to $x_b$. If $f(x) > f(x_i)$, $i = 1, 2, \ldots, M$ then the volume of $S$ is reduced by the factor $a_3$ where $a_3 \in (0, 0.1)$.

(10) When the stopping criteria are satisfied then stop the programme, otherwise go to step (4).

In step (1) of the algorithm the initial grid may be chosen in many different ways. It may be any set of points for which $x, f(x)$ are available, provided that their location uniquely defines the model. They may also be chosen, in a methodical manner so as to represent the behaviour of the function around the starting point as accurately as possible. One possible way of selecting the initial grid points would be to have the grid include the
initial point $x_1$, and $M-1$ other points selected from the set

$$x_1 + e_j \quad j = 1, 2, \ldots, M,$$

where $e_j$ can either be zero, positive or negative vectors along the coordinate axes.

The location of the interpolation points is discussed more explicitly in the next section.

In step 3, the grid points are kept in the order of increasing Euclidean distance from the basepoint. By ordering the points in this way, at each iteration the point in the last position is excluded (the point having the largest distance from $x_1$), and the new trial point takes its position amongst the remaining points of the grid. $M$ points resulted at each iteration. Winfield (1973) suggests a different method of using the data table. His method requires more storage space in the memory of the computer, because the data table has almost $2^M$ entries. His method has the advantage that points, which are not amongst the $M-1$ closest points to the basepoint have not been discarded but may be re-used.

Winfield's data table was considered computer-storage consuming, and the exact grid was used in most of the cases.

In step 5 a safeguarding procedure is incorporated which is necessary for rational models if the objective function behaves locally as a quadratic (Evans, G.A. (1979)).

In step 7 a constrained minimum of (3.2.1) is looked for within $S$. Except for the case when $R(x)$ has an unconstrained minimum, within $S$, the minimization of (3.2.1) requires the solution of the constrained minimization problem:

$$\minimize (3.2.1)$$

subject to

$$x^T x - r^2 = 0$$
where \( r \) is the radius of \( S \). The minimum of this problem occurs at a stationary point of the Lagrangian function:

\[
\text{Lag. Fun.} = R(x) + \frac{1}{2} \lambda (x^T x - r^2)
\]

At this point the attempt to perform an exact minimization was abandoned. An approximate method was used instead. This was done because of the complexity of the problem of finding the stationary points of (3.3.2) and the excessive computational work involved. The method used is described in Section 4.

Finally in step 9 three positive numbers \( a_1, a_2, \) and \( a_3 \) are used. Computational experience suggests that after a failure it is desirable to reduce the volume of the region of validity by the factor \( a_2 \) (usually taken as \((3/4)^2\)) and to increase the reduction even more, either as a result of consecutive failures, or when condition:

\[
f(x) \geq f(x_i), \quad i = 1, 2, \ldots, M
\]

holds. Except for the case near the final stage, the radius of \( S \) increases after every successful trial point. (A trial point is considered as successful if it gives a smaller objective function value). Therefore, a sequence of successful trials causes the points to become spread out along the valley which results in a poor distribution. So the area which has to be spanned is too large for successful modelling by a rational function and it is followed by a failure. This has a pulsating effect on the sphere of validity \( S \), as a sequence of successful trials (which is leading to the expansion of the region of validity) is followed by a
sequence of failures (leading to contraction).

The adjustment of various arbitrary parameters can influence the tendency of SRM to explore. A small tendency to explore has as a result the concentration of the data in a small region. The probability of a success is high but the rate of objective function reduction per step is low, because the steps are small. On the other hand a large tendency to explore scatters the data over a wide range, and a compromise has to be made between these two extremes for some desirable level of the exploring ability of the SRM.

The stopping criteria used in the algorithm are the following:

(i) \( ||x_b - x'|| \leq E_1 \)

(ii) \( |f(x_b) - f(x')| \leq E_2 \)

where \( x' \) is the new trial point and \( E_1, E_2 \) are respectively \( 10^{-5} \) and \( 10^{-12} \). The algorithm terminates when both conditions (i) and (ii) are satisfied.

Remark: The independent coefficients of the model (3.2.1) are calculated, using Gauss elimination for solving a system of \( N \) linear equations in \( N \) unknowns, e.g. \( Ax = b \). The matrix \( A \) is singular if and only if the distinct points \( x_i \), \( i = 1, 2, \ldots, N \) lie on the same second order curve (see Berezin & Zidkho, 1965, p.157).

That is for \( n > 2 \) the matrix is singular if there are constants \( k_j, j=1, \ldots, 7 \) that are not all zero such that:

\[
\frac{1}{2} (x_{1,i}^2) k_1 + \frac{1}{2} (x_{2,i}^2) k_2 + (x_{1,i} x_{2,i}) k_3 + (x_{1,i}) k_4 +
\]

\[
(x_{2,i}) k_5 - (f(x_{1,i}) x_{1,i}) k_6 - (f(x_{2,i}) x_{2,i}) k_7 = 0 , \quad i = 1, 2, \ldots, N.
\]
3.4 CONSTRAINED MINIMIZATION OF A RATIONAL FUNCTION.

In the previous section the problem of minimizing a rational form arose at the seventh step of the algorithm. The relation (3.2.1) can also be rewritten as:

\[ R(x) = \frac{N(x)}{L(x)} \]  

(3.4.1)

where \( N(x) \), \( L(x) \) are continuous real valued functions, \( x \in S \).

Let \( S \) be a compact and connected subset of \( \mathbb{R}^n \). Furthermore the following assumption is made:

\[ L(x) > 0 \quad \text{for all} \ x \in S. \]

It was shown in Dinkelbach (1967) that the following problems:

(A) \( \min \left\{ \frac{N(x)}{L(x)} \mid x \in S \right\} \)

and (B) \( \min \left\{ N(x) - CL(x) \mid C \in \mathbb{R} \text{ and } x \in S \right\} \)

are equivalent problems. In other words

\[ C_o = \min \left\{ \frac{N(x)}{L(x)} \mid x \in S \right\}, \]

if and only if

\[ \min \left\{ N(x) - C_o L(x) \mid x \in S \right\} = 0. \]

Therefore the initial problem (A) was transformed to the problem (B), which is an optimization problem involving the parameter \( C \). Problem (B) is easier to solve than problem (A), and it will be solved instead of problem (A).

Choosing an appropriate initial point \( \hat{x} \) (i.e. the current
basepoint) and assuming that the minimum point \( x^* \) of problem (A) exists, then

\[
\frac{N(\hat{x})}{L(\hat{x})} \geq \frac{N(x^*)}{L(x^*)} \quad \text{for all } \hat{x} \in S.
\]

The first problem (B) then becomes:

\[
F(x') = \min \left\{ \frac{N(x)\hat{t} - \hat{k}L(x)}{L(x)} > 0, \quad \hat{t} = L(\hat{x}), \quad \hat{k} = N(\hat{x}) \quad \text{for all } x \in S \right\}
\]

where \( x' \) is the solution of the previous problem.

If \( F(x') < \delta \), where \( \delta \) is a small positive number then the point \( x' \) is the solution of problem (A).

If \( F(x') \geq \delta \) then evaluate \( t_1 = L(x') \), \( k_1 = N(x') \)

and find the point \( x_1 \) such that

\[
F(x_1) = \min \left\{ \frac{N(x)t_1 - k_1L(x)}{L(x)} > 0, \quad \text{for all } x \in S \right\}.
\]

Hence a sequence of subproblems is constructed which produces a sequence of points \( \{x_i\} \). This sequence of points \( \{x_i\} \) converges to the optimal point of problem (B). Computational experience showed that the sequence \( \{x_i\} \) converges to \( x^* \) after three or four steps.

Although an accurate minimization of problem (B) is not very expensive computationally, it demands a bigger computational effort than the algorithm using the quadratic model. For this reason a maximum of two steps was allowed for the minimization of problem (B). This makes the solution of the problem (B) not accurate. However, the numerical results indicate that this does not influence the overall performance of the algorithm.
The next section deals with the numerical experiments. Seven standard test as well as three non-standard functions were used to compare the different models used.

3.5 **NUMERICAL RESULTS.**

**GENERAL RATIONAL FUNCTIONS.**

In this first part the rational model (3.2.1) is tested on two dimensional functions and compared with the quadratic model used by Winfield (1973). The numerical experiments are performed in a way to gain more insight into the role of the different parameters employed in the algorithm.

The test functions used are given in the Appendix and were chosen because they are among the most common functions in the literature used to check the performance of algorithms.

For the sake of simplicity the notation and parameters which have been used are given below:

\[ E_1, E_2 : \text{convergence criteria} \]

\[ H : \text{Parameter indicating grid size.} \]

(This is the distance in \((x,y)\) plane of the points of the initial grid, from the starting point \(x_0\)).

\[ a_1, a_2, a_3 : \text{The reducing factors of the radius of the sphere } S. \]

**Method A** : Winfield's, S.Q.M. method.

**Method B** : Rational model method.
The results presented in this section were obtained from implementable algorithms, which have one major difference from the conceptual algorithms.

In the implementable algorithms, the points of the initial grid were selected using the starting point and a preprocessor (Wolfe, (1978), p. 64). The preprocessor consisted of two linear searches along the coordinate axes. As will be shown, the initial grids produced with the help of the preprocessor do not result in a faster convergence to the minimum or more representative approximation to the objective function. The numbers under the columns A and B represent the number of function evaluations, required for convergence.

i) The effect of $a_1$, $a_2$, $a_3$ on the performance of the algorithms.

**TEST FUNCTION 1. $H = 2$**

<table>
<thead>
<tr>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$a_3$</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>.4</td>
<td>.6</td>
<td>.001</td>
<td>72</td>
<td>60</td>
</tr>
<tr>
<td>.29</td>
<td>.7</td>
<td>.001</td>
<td>72</td>
<td>83</td>
</tr>
<tr>
<td>.39</td>
<td>.7</td>
<td>.001</td>
<td>67</td>
<td>80</td>
</tr>
<tr>
<td>.49</td>
<td>.7</td>
<td>.001</td>
<td>46</td>
<td>78</td>
</tr>
<tr>
<td>.59</td>
<td>.7</td>
<td>.001</td>
<td>31</td>
<td>47</td>
</tr>
<tr>
<td>.69</td>
<td>.7</td>
<td>.001</td>
<td>33</td>
<td>37</td>
</tr>
<tr>
<td>.79</td>
<td>.7</td>
<td>.001</td>
<td>30</td>
<td>42</td>
</tr>
<tr>
<td>.89</td>
<td>.7</td>
<td>.001</td>
<td>32</td>
<td>39</td>
</tr>
<tr>
<td>.99</td>
<td>.7</td>
<td>.001</td>
<td>43</td>
<td>49</td>
</tr>
<tr>
<td>.9</td>
<td>.001</td>
<td>.01</td>
<td>37</td>
<td>44</td>
</tr>
<tr>
<td>.9</td>
<td>.004</td>
<td>.01</td>
<td>37</td>
<td>52</td>
</tr>
<tr>
<td>.9</td>
<td>.008</td>
<td>.01</td>
<td>37</td>
<td>42</td>
</tr>
</tbody>
</table>
The initial grid of the points used, to construct the previous table included the point (0.8, 1.0). This point was near to the minimum point of Test function 1, and this resulted in fast convergence of the algorithms to this minimum.

**TEST FUNCTION 2.** \( H = 2 \)

<table>
<thead>
<tr>
<th>( a_1 )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( A )</th>
<th>( B )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.35</td>
<td>.7</td>
<td>.01</td>
<td>65</td>
<td>202</td>
</tr>
<tr>
<td>.4</td>
<td>.7</td>
<td>.01</td>
<td>90</td>
<td>160</td>
</tr>
<tr>
<td>.45</td>
<td>.7</td>
<td>.01</td>
<td>139</td>
<td>164</td>
</tr>
<tr>
<td>.5</td>
<td>.7</td>
<td>.01</td>
<td>102</td>
<td>140</td>
</tr>
<tr>
<td>.55</td>
<td>.7</td>
<td>.01</td>
<td>95</td>
<td>154</td>
</tr>
<tr>
<td>.6</td>
<td>.7</td>
<td>.01</td>
<td>106</td>
<td>134</td>
</tr>
<tr>
<td>.65</td>
<td>.7</td>
<td>.01</td>
<td>84</td>
<td>136</td>
</tr>
<tr>
<td>.7</td>
<td>.7</td>
<td>.01</td>
<td>88</td>
<td>136</td>
</tr>
<tr>
<td>.75</td>
<td>.7</td>
<td>.01</td>
<td>81</td>
<td>148</td>
</tr>
<tr>
<td>.8</td>
<td>.7</td>
<td>.01</td>
<td>125</td>
<td>125</td>
</tr>
<tr>
<td>.85</td>
<td>.7</td>
<td>.01</td>
<td>88</td>
<td>118</td>
</tr>
<tr>
<td>.9</td>
<td>.7</td>
<td>.01</td>
<td>111</td>
<td>116</td>
</tr>
<tr>
<td>.95</td>
<td>.7</td>
<td>.01</td>
<td>111</td>
<td>117</td>
</tr>
<tr>
<td>1.0</td>
<td>.7</td>
<td>.01</td>
<td>109</td>
<td>120</td>
</tr>
<tr>
<td>1.05</td>
<td>.7</td>
<td>.01</td>
<td>95</td>
<td>108</td>
</tr>
<tr>
<td>1.1</td>
<td>.7</td>
<td>.01</td>
<td>96</td>
<td>105</td>
</tr>
<tr>
<td>1.2</td>
<td>.7</td>
<td>.01</td>
<td>103</td>
<td>125</td>
</tr>
<tr>
<td>1.3</td>
<td>.7</td>
<td>.01</td>
<td>103</td>
<td>82</td>
</tr>
<tr>
<td>1.4</td>
<td>.7</td>
<td>.01</td>
<td>106</td>
<td>76</td>
</tr>
<tr>
<td>1.5</td>
<td>.7</td>
<td>.01</td>
<td>107</td>
<td>88</td>
</tr>
<tr>
<td>1.6</td>
<td>.7</td>
<td>.01</td>
<td>48</td>
<td>96</td>
</tr>
<tr>
<td>1.7</td>
<td>.7</td>
<td>.01</td>
<td>68</td>
<td>83</td>
</tr>
<tr>
<td>1.8</td>
<td>.7</td>
<td>.01</td>
<td>58</td>
<td>83</td>
</tr>
</tbody>
</table>
TEST FUNCTION 3. \( H = 2 \)

<table>
<thead>
<tr>
<th>( a_1 )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( A )</th>
<th>( B )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.8</td>
<td>.7</td>
<td>.01</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>.9</td>
<td>.7</td>
<td>.01</td>
<td>34</td>
<td>36</td>
</tr>
<tr>
<td>1.0</td>
<td>.7</td>
<td>.01</td>
<td>30</td>
<td>39</td>
</tr>
<tr>
<td>1.1</td>
<td>.7</td>
<td>.01</td>
<td>31</td>
<td>36</td>
</tr>
<tr>
<td>1.2</td>
<td>.7</td>
<td>.01</td>
<td>31</td>
<td>34</td>
</tr>
<tr>
<td>1.3</td>
<td>.7</td>
<td>.01</td>
<td>41</td>
<td>38</td>
</tr>
<tr>
<td>1.4</td>
<td>.7</td>
<td>.01</td>
<td>39</td>
<td>35</td>
</tr>
<tr>
<td>1.5</td>
<td>.7</td>
<td>.01</td>
<td>35</td>
<td>40</td>
</tr>
<tr>
<td>1.6</td>
<td>.7</td>
<td>.01</td>
<td>38</td>
<td>39</td>
</tr>
<tr>
<td>1.7</td>
<td>.7</td>
<td>.01</td>
<td>38</td>
<td>40</td>
</tr>
</tbody>
</table>

ii) The effect of the preprocessor on the performance of the algorithms.

TEST FUNCTION 1. \( H = 0.7, a_1 = 2, a_3 = 0.001 \)

<table>
<thead>
<tr>
<th>( a_2 )</th>
<th>( B(\text{with}) )</th>
<th>( B(\text{without}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.5</td>
<td>160</td>
<td>152</td>
</tr>
<tr>
<td>.4</td>
<td>143</td>
<td>135</td>
</tr>
<tr>
<td>.3</td>
<td>184</td>
<td>176</td>
</tr>
</tbody>
</table>

Finally the influence of the initial grid size on the behaviour of the algorithms was tested. The first test function was also used with a standard NAGF library subroutine (E04DCF). This subroutine
was based on an hybrid method by Powell and converged after 43 function and 86 gradient evaluations.

iii) Comparison of A and B, using different initial grids.

TEST FUNCTION 1. $a_1 = 0.9$, $a_2 = 0.5$, $a_3 = 0.001$.

<table>
<thead>
<tr>
<th>H</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>121</td>
<td>186</td>
</tr>
<tr>
<td>0.2</td>
<td>137</td>
<td>197</td>
</tr>
<tr>
<td>0.3</td>
<td>106</td>
<td>146</td>
</tr>
<tr>
<td>0.4</td>
<td>118</td>
<td>121</td>
</tr>
<tr>
<td>0.5</td>
<td>119</td>
<td>150</td>
</tr>
<tr>
<td>1.0</td>
<td>108</td>
<td>152</td>
</tr>
<tr>
<td>2.0</td>
<td>59</td>
<td>62</td>
</tr>
<tr>
<td>2.5</td>
<td>115</td>
<td>151</td>
</tr>
<tr>
<td>3.0</td>
<td>114</td>
<td>176</td>
</tr>
<tr>
<td>3.1</td>
<td>126</td>
<td>171</td>
</tr>
<tr>
<td>3.2</td>
<td>103</td>
<td>139</td>
</tr>
<tr>
<td>3.3</td>
<td>103</td>
<td>117</td>
</tr>
<tr>
<td>3.4</td>
<td>119</td>
<td>113</td>
</tr>
<tr>
<td>3.5</td>
<td>128</td>
<td>134</td>
</tr>
<tr>
<td>3.6</td>
<td>147</td>
<td>135</td>
</tr>
<tr>
<td>3.7</td>
<td>118</td>
<td>195</td>
</tr>
<tr>
<td>3.8</td>
<td>125</td>
<td>152</td>
</tr>
<tr>
<td>3.9</td>
<td>138</td>
<td>141</td>
</tr>
<tr>
<td>4.0</td>
<td>131</td>
<td>125</td>
</tr>
<tr>
<td>4.5</td>
<td>99</td>
<td>123</td>
</tr>
<tr>
<td>5.0</td>
<td>102</td>
<td>124</td>
</tr>
<tr>
<td>5.5</td>
<td>127</td>
<td>200</td>
</tr>
<tr>
<td>6.0</td>
<td>105</td>
<td>151</td>
</tr>
</tbody>
</table>
SPECIAL RATIONAL MODELS.

In the previous sections, an optimization algorithm using the rational model (3.2.1) was formulated, described and tested.

In this section, a new rational model is introduced which requires less analytical work than (3.2.1), but lacks the approximation ability of this model.

If \( q(x) \) is a general polynomial of the second degree, for all \( x \in \mathbb{R}^n \), then the new model can be written as:

\[
R_1(x) = \frac{\epsilon_1 q(x) - 1}{\epsilon_2 q(x)}
\]

(3.5.1)

where \( \epsilon_1, \epsilon_2 > 0 \), \( q(x) \neq 0 \).

For both quadratic and (3.5.1) models, the same amount of computational work is required for locating new trial points.

i) A comparison of the model (3.5.1), with a quadratic model and a model of the form:

\[
R_2(x) = \frac{q(x)}{q(x)+1}
\]

(3.5.2)

was performed on the Prime computer. The following parameters were used:
TEST FUNCTION 1.  \( a_1 = 0.9, \ a_2 = 0.5, \ a_3 = 0.01, \ H = 3 \).

Double Precision implementation has been used.

<table>
<thead>
<tr>
<th>Function Evaluations</th>
<th>A</th>
<th>Model (3.5.1)</th>
<th>Model (3.5.2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function Value at the point</td>
<td>0</td>
<td>24.2</td>
<td>24.2</td>
</tr>
<tr>
<td>(-1.2,1.0)</td>
<td>(-1.2,1.0)</td>
<td>(-1.2,1.0)</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>92</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>0.884 ( 10^{-1} )</td>
<td>0.734 ( 10^{-1} )</td>
<td>0.933 ( 10^{-1} )</td>
<td></td>
</tr>
<tr>
<td>(0.704,0.493)</td>
<td>(0.745,0.545)</td>
<td>(0.697,0.482)</td>
<td></td>
</tr>
<tr>
<td>122</td>
<td>123</td>
<td>141</td>
<td></td>
</tr>
<tr>
<td>0.6 ( 10^{-15} )</td>
<td>0.8 ( 10^{-13} )</td>
<td>0.3 ( 10^{-15} )</td>
<td></td>
</tr>
<tr>
<td>(1.0,1.0)</td>
<td>(1.0,1.0)</td>
<td>(1.0,1.0)</td>
<td></td>
</tr>
</tbody>
</table>

Next, a comparison between the model (3.5.1) and the quadratic model (method A) was performed in the ICL computer (19045*), using single precision implementation, the same stopping criteria as before, and the following parameters:
TEST FUNCTION 1.  $a_1 = 0.9$, $a_2 = 0.5$, $a_3 = 0.001$.

<table>
<thead>
<tr>
<th>H</th>
<th>A</th>
<th>Model (3.5.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>119</td>
<td>152</td>
</tr>
<tr>
<td>0.2</td>
<td>133</td>
<td>166</td>
</tr>
<tr>
<td>0.3</td>
<td>112</td>
<td>166</td>
</tr>
<tr>
<td>0.4</td>
<td>100</td>
<td>99</td>
</tr>
<tr>
<td>0.5</td>
<td>133</td>
<td>142</td>
</tr>
<tr>
<td>1.0</td>
<td>121</td>
<td>194</td>
</tr>
<tr>
<td>1.5</td>
<td>118</td>
<td>149</td>
</tr>
<tr>
<td>2.0</td>
<td>36</td>
<td>61</td>
</tr>
<tr>
<td>2.5</td>
<td>107</td>
<td>96</td>
</tr>
<tr>
<td>3.0</td>
<td>122</td>
<td>186</td>
</tr>
<tr>
<td>3.5</td>
<td>141</td>
<td>82</td>
</tr>
<tr>
<td>4.0</td>
<td>120</td>
<td>92</td>
</tr>
<tr>
<td>4.5</td>
<td>108</td>
<td>181</td>
</tr>
<tr>
<td>5.0</td>
<td>122</td>
<td>127</td>
</tr>
<tr>
<td>5.5</td>
<td>104</td>
<td>184</td>
</tr>
</tbody>
</table>

ii) A comparison was made of the (3.5.1) and quadratic models using different reducing factors $a_1$, $a_2$. The parameters used in the (3.5.1) model were: $\varepsilon_1 = 5$, $\varepsilon_2 = 2$. The first test function was used with grid size 2.5. The numerical results were obtained using the Prime computer and double precision implementation.
<table>
<thead>
<tr>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$a_3$</th>
<th>A</th>
<th>Model (3.5.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>0.1</td>
<td>0.001</td>
<td>111</td>
<td>164</td>
</tr>
<tr>
<td>2.0</td>
<td>0.2</td>
<td>0.001</td>
<td>133</td>
<td>131</td>
</tr>
<tr>
<td>2.0</td>
<td>0.3</td>
<td>0.001</td>
<td>116</td>
<td>137</td>
</tr>
<tr>
<td>2.0</td>
<td>0.4</td>
<td>0.001</td>
<td>116</td>
<td>134</td>
</tr>
<tr>
<td>2.0</td>
<td>0.5</td>
<td>0.001</td>
<td>125</td>
<td>81</td>
</tr>
<tr>
<td>2.0</td>
<td>0.6</td>
<td>0.001</td>
<td>108</td>
<td>134</td>
</tr>
<tr>
<td>2.0</td>
<td>0.7</td>
<td>0.001</td>
<td>114</td>
<td>143</td>
</tr>
<tr>
<td>2.0</td>
<td>0.8</td>
<td>0.001</td>
<td>120</td>
<td>147</td>
</tr>
<tr>
<td>2.0</td>
<td>0.9</td>
<td>0.001</td>
<td>110</td>
<td>119</td>
</tr>
<tr>
<td>2.0</td>
<td>1.0</td>
<td>0.001</td>
<td>108</td>
<td>129</td>
</tr>
<tr>
<td>2.0</td>
<td>1.1</td>
<td>0.001</td>
<td>117</td>
<td>118</td>
</tr>
<tr>
<td>2.0</td>
<td>1.2</td>
<td>0.001</td>
<td>116</td>
<td>137</td>
</tr>
<tr>
<td>2.0</td>
<td>1.3</td>
<td>0.001</td>
<td>114</td>
<td>140</td>
</tr>
<tr>
<td>2.0</td>
<td>1.4</td>
<td>0.001</td>
<td>107</td>
<td>112</td>
</tr>
<tr>
<td>0.9</td>
<td>0.5</td>
<td>0.001</td>
<td>107</td>
<td>96</td>
</tr>
<tr>
<td>1.0</td>
<td>0.5</td>
<td>0.001</td>
<td>123</td>
<td>94</td>
</tr>
<tr>
<td>1.1</td>
<td>0.5</td>
<td>0.001</td>
<td>145</td>
<td>95</td>
</tr>
<tr>
<td>1.2</td>
<td>0.5</td>
<td>0.001</td>
<td>116</td>
<td>96</td>
</tr>
<tr>
<td>1.3</td>
<td>0.5</td>
<td>0.001</td>
<td>123</td>
<td>83</td>
</tr>
<tr>
<td>1.4</td>
<td>0.5</td>
<td>0.001</td>
<td>100</td>
<td>88</td>
</tr>
<tr>
<td>1.5</td>
<td>0.5</td>
<td>0.001</td>
<td>114</td>
<td>107</td>
</tr>
<tr>
<td>1.6</td>
<td>0.5</td>
<td>0.001</td>
<td>122</td>
<td>96</td>
</tr>
<tr>
<td>3.0</td>
<td>0.5</td>
<td>0.001</td>
<td>126</td>
<td>98</td>
</tr>
<tr>
<td>4.0</td>
<td>0.5</td>
<td>0.001</td>
<td>127</td>
<td>125</td>
</tr>
<tr>
<td>5.0</td>
<td>0.5</td>
<td>0.001</td>
<td>99</td>
<td>100</td>
</tr>
<tr>
<td>6.0</td>
<td>0.5</td>
<td>0.001</td>
<td>112</td>
<td>116</td>
</tr>
<tr>
<td>7.0</td>
<td>0.5</td>
<td>0.001</td>
<td>111</td>
<td>118</td>
</tr>
<tr>
<td>8.0</td>
<td>0.5</td>
<td>0.001</td>
<td>128</td>
<td>119</td>
</tr>
<tr>
<td>9.0</td>
<td>0.5</td>
<td>0.001</td>
<td>103</td>
<td>124</td>
</tr>
<tr>
<td>10.0</td>
<td>0.5</td>
<td>0.001</td>
<td>96</td>
<td>100</td>
</tr>
</tbody>
</table>
iii) TEST FUNCTION 4. \( H \) varies, \( a_1 = 0.9, a_2 = 0.5, a_3 = 0.01 \)

<table>
<thead>
<tr>
<th>( H )</th>
<th>( A )</th>
<th>Model (3.5.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>59</td>
<td>79</td>
</tr>
<tr>
<td>0.2</td>
<td>36</td>
<td>68</td>
</tr>
<tr>
<td>0.3</td>
<td>38</td>
<td>68</td>
</tr>
<tr>
<td>0.4</td>
<td>43</td>
<td>66</td>
</tr>
<tr>
<td>0.5</td>
<td>46</td>
<td>49</td>
</tr>
<tr>
<td>0.6</td>
<td>39</td>
<td>43</td>
</tr>
<tr>
<td>0.7</td>
<td>34</td>
<td>62</td>
</tr>
<tr>
<td>0.8</td>
<td>39</td>
<td>45</td>
</tr>
<tr>
<td>0.9</td>
<td>34</td>
<td>56</td>
</tr>
<tr>
<td>1.0</td>
<td>37</td>
<td>45</td>
</tr>
<tr>
<td>1.1</td>
<td>30</td>
<td>48</td>
</tr>
<tr>
<td>1.6</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>2.1</td>
<td>34</td>
<td>39</td>
</tr>
<tr>
<td>2.6</td>
<td>36</td>
<td>28</td>
</tr>
<tr>
<td>3.1</td>
<td>34</td>
<td>56</td>
</tr>
<tr>
<td>3.6</td>
<td>49</td>
<td>74</td>
</tr>
<tr>
<td>4.1</td>
<td>54</td>
<td>58</td>
</tr>
<tr>
<td>4.6</td>
<td>38</td>
<td>49</td>
</tr>
<tr>
<td>5.1</td>
<td>37</td>
<td>51</td>
</tr>
<tr>
<td>5.6</td>
<td>32</td>
<td>53</td>
</tr>
<tr>
<td>6.1</td>
<td>34</td>
<td>56</td>
</tr>
<tr>
<td>6.6</td>
<td>36</td>
<td>56</td>
</tr>
<tr>
<td>7.1</td>
<td>34</td>
<td>83</td>
</tr>
<tr>
<td>7.6</td>
<td>32</td>
<td>80</td>
</tr>
<tr>
<td>8.1</td>
<td>39</td>
<td>61</td>
</tr>
</tbody>
</table>
EFFECT OF VOLUME REDUCTION FACTOR ON THE NUMBER OF ITERATIONS REQUIRED FOR CONVERGENCE.

For each problem the volume reduction factor, $a_2$, is varied and the number of function evaluations required to reduce the function to a prescribed number is recorded. The model tested is (3.5.1).

In each problem the size of the data table is constant. Figures (i) through (iv) show the dependence of the number of function evaluations on the volume reduction factor (VRF).

The figures show that the number of iterations required for convergence is large for very small and very large values of the volume reduction factor, and smaller for intermediate values.

A reasonable rule which can be deduced from the previous figures for the Volume Reduction Factor is:

$$VRF = (0.75)^n.$$
Figure (i)
Test Function 1
Figure (ii)
Test Function 2
Figure (iii)

Test Function 3
Figure (iv)
Test Function 4
EFFECT OF THE NUMBER OF ITERATIONS REQUIRED FOR CONVERGENCE ON THE MEMORY SIZE.

The effect of the number of iterations required for convergence on the memory size when model (3.5.1) is used, is shown in the figures below. The minimum number of entries required is called NACT, (the number of active points).

Figure (v) below shows that providing a 40% excess over this minimum can yield a 26% decrease in the number of function evaluations required for convergence.

Figure (vi) shows that a 33% increase in the memory size yield a 17% decrease in the number of function evaluations.

Figure (vii) shows that the 33% increase in the memory size gave a 31% decrease in the number of function evaluations.
Figure (v)
Test Function 4

Number of points kept in memory
Figure (vi)
Test Function 2

Number of points kept in memory
Figure (vii)
Test Function 1

Number of points kept in memory
Pulsation in Size of the Sphere of Validity.

Figure (viii) shows how the sphere of validity contracts and expands during a typical Sequential Rational model run. The model used is again the model (3.5.1).

The circled points represent instances in which the iterate is constrained to lie within the sphere having as a centre a new basepoint. Figure (viii) shows that a successful trial is likely to be followed by another successful trial, and a failure by another failure.

If these oscillations are smoothed by using a different policy then the exploration of the method is restrained and the average rate of cost reduction is reduced.
Figure (viii)

Test Function 1

Radius of Sphere of Validity during descent of Rosenbrock's Valley.
All the standard test functions tried so far belong to the polynomial class of functions. Therefore two non-polynomial functions were used to further test the new model, (3.5.1). The first one is the kinetic problem, (see Rosenbrock and Storey, (1966), p.102).

Problem 8: \[ k_i = c_i \exp \left\{ -\frac{E_i}{R} \left( \frac{1}{T} - \frac{1}{658} \right) \right\}, \quad i = 1, \ldots, 5 \]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_i )</td>
<td>1.02</td>
<td>0.93</td>
<td>0.386</td>
<td>3.28</td>
<td>0.084</td>
</tr>
<tr>
<td>( e_i )</td>
<td>16.000</td>
<td>14,000</td>
<td>15,000</td>
<td>10,000</td>
<td>15,000</td>
</tr>
</tbody>
</table>

\[
f(T,t) = \frac{k_4 k_1 \exp(- k_5 t)}{k-k_4} \left\{ \frac{1 - \exp\left[ - \frac{(k_4-k_5)t}{k-k_5} \right]}{k_4-k_5} - \frac{1 - \exp\left[ - \frac{(k-k_5)t}{k-k_5} \right]}{k-k_5} \right\}
\]

where \( k = k_1 + k_2 + k_3 \).

Three different starting points were tried. Namely: (1073, .5), (873, .5), (773, .5)

The progress of the two models for the three different starting points is given below:
<table>
<thead>
<tr>
<th>Funct. Eval.</th>
<th>Rational</th>
<th>Quadratic</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$T$</td>
<td>$t$</td>
</tr>
<tr>
<td>1</td>
<td>1073.05</td>
<td>0.500000</td>
</tr>
<tr>
<td>12</td>
<td>1071.84</td>
<td>0.261322</td>
</tr>
<tr>
<td>18</td>
<td>1071.64</td>
<td>0.020567</td>
</tr>
<tr>
<td>39</td>
<td>1072.03</td>
<td>0.042914</td>
</tr>
<tr>
<td>55</td>
<td>1071.01</td>
<td>0.045243</td>
</tr>
<tr>
<td>68</td>
<td>1034.90</td>
<td>0.055165</td>
</tr>
<tr>
<td>91</td>
<td>983.03</td>
<td>0.075901</td>
</tr>
<tr>
<td>108</td>
<td>983.03</td>
<td>0.075903</td>
</tr>
<tr>
<td>156</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Funct. Eval.</th>
<th>Rational</th>
<th>Quadratic</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$T$</td>
<td>$t$</td>
</tr>
<tr>
<td>1</td>
<td>873.00</td>
<td>0.500000</td>
</tr>
<tr>
<td>11</td>
<td>870.38</td>
<td>0.197678</td>
</tr>
<tr>
<td>30</td>
<td>870.43</td>
<td>0.173137</td>
</tr>
<tr>
<td>50</td>
<td>920.99</td>
<td>0.118491</td>
</tr>
<tr>
<td>84</td>
<td>963.94</td>
<td>0.085733</td>
</tr>
<tr>
<td>116</td>
<td>978.36</td>
<td>0.078234</td>
</tr>
<tr>
<td>131</td>
<td>983.03</td>
<td>0.075903</td>
</tr>
<tr>
<td>145</td>
<td>983.03</td>
<td>0.075903</td>
</tr>
<tr>
<td>218</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
In the previous tables the symbol "-" was used to indicate that the algorithm using the rational model has already converged. From the same tables it is clear that the rational model is more efficient than the quadratic one. If the computational labour is proportional to the number of function evaluations needed for convergence, then the rational model needs overall 25% less computational labour than the quadratic one.

The second non-polynomial test function is a generalization of the one-dimensional function given in Brent (1973), p.77.

Problem 9:

\[ f(x_1, x_2) = \sum_{i=1}^{20} \left( \frac{(2i-5)}{x_1 - i} \right)^2 + \left( \frac{2i-5}{x_2 - i} \right)^2 \]
This function has poles at \((x_1, x_2) = (1^2, 1^2), (2^2, 2^2), \ldots, (20^2, 20^2)\). Restricted to the open interval \((i^2, (i+1)^2)\), for \(i = 1, 2, \ldots, 19\) it is unimodal with an interior minimum.

The next table gives the final point and function value reached by both models as well as the number of function evaluations needed for convergence. The point, \((i^2+0.5), i^2+0.5\) is taken as an initial point for \(i = 1, 2, \ldots, 19\).

<table>
<thead>
<tr>
<th>i</th>
<th>( (x_1^<em>, x_2^</em>) )</th>
<th>( f(x_1^<em>, x_2^</em>) )</th>
<th>Funct. Eval.</th>
<th>Quadratic</th>
<th>Rational</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>((3.0229153, 3.0229153))</td>
<td>7.3533980</td>
<td>27</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>((6.6837536, 6.6837540))</td>
<td>2.2237000</td>
<td>31</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>((11.2387031, 11.2387028))</td>
<td>2.4364435</td>
<td>31</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>((19.6760000, 19.6760000))</td>
<td>4.3242206</td>
<td>41</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>((29.8282272, 29.8282280))</td>
<td>6.0645810</td>
<td>47</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>((41.9061171, 41.9061165))</td>
<td>7.5167713</td>
<td>50</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>((55.9535963, 55.9535962))</td>
<td>8.7108208</td>
<td>53</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>((71.985662, 71.9856662))</td>
<td>9.6965919</td>
<td>48</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>((90.0088693, 90.0088687))</td>
<td>10.5175173</td>
<td>54</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>((110.0265329, 110.0265327))</td>
<td>11.2073056</td>
<td>57</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>((132.0405521, 132.0405523))</td>
<td>11.7912086</td>
<td>62</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>((156.0521148, 156.0521151))</td>
<td>12.2877231</td>
<td>60</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>((182.0620595, 182.0620602))</td>
<td>12.71015292</td>
<td>60</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>((210.0711011, 210.0711011))</td>
<td>13.06673240</td>
<td>58</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>((240.0800483, 240.0800483))</td>
<td>13.36072797</td>
<td>59</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>((272.0902664, 272.0902671))</td>
<td>13.58770767</td>
<td>59</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>((306.1051234, 306.1051234))</td>
<td>13.72699621</td>
<td>62</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>((342.1369455, 342.1369444))</td>
<td>13.70780493</td>
<td>61</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>((380.2687097, 380.2687097))</td>
<td>13.20169410</td>
<td>66</td>
<td>37</td>
<td></td>
</tr>
</tbody>
</table>
The above table shows that the rational model needs less computational labour than the quadratic one to converge to each one of the minima of the ninth problem. If SQ represents the sum of the fourth column and SR the sum of the last column, then the ratio \((SR/SQ) = 41\) gives the percentage save of the rational model on computational labour, for the ninth problem. The results of the previous table are also reflected on the following figure (ix).
Figure (ix)

+ Rational Model

0 Quadratic Model
The last test function used is the four-dimensional Shekel's function (De Baise & Frontini (1978)).

Problem 10:

\[
 f(x_1, x_2, x_3, x_4) = \sum_{i=1}^{7} \left( \frac{1}{\sum_{k=1}^{4} (x_k - a_{ik})^2 + c_i} \right)
\]

where \( c_i \) and \( a_{ik} \) are given by the following table:

<table>
<thead>
<tr>
<th>( i )</th>
<th>( c_i )</th>
<th>( a_{ik} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1</td>
<td>0.7</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td>5</td>
<td>0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>6</td>
<td>0.6</td>
<td>0.8</td>
</tr>
<tr>
<td>7</td>
<td>0.7</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Different starting points were tried for \( 0 \leq x_k \leq 1 \), \( k = 1, \ldots, 4 \). The following figures represent the computational effort for each model measured in number of function evaluations for the following starting points:

(1) (.8, .7, .6, .1)
(2) (1., .0, 1., .0)
(3) (0., 0., 0., 0.)
(4) (1., 1., .5, .3)
(5) (.5, .5, .5, .5)
(6) (.5, .3, .1, .1)
Both algorithms converge to the point
\[ x^* = (0.6822524755, 0.678445338, 0.5211067878, 0.1258426519) \]
for all the starting points used. The final function value is:
\[ f(x^*) = 16.39827730. \]
A concise summary of the numerical results of Problem 10 is given in the following table:

<table>
<thead>
<tr>
<th>Starting Point</th>
<th>Number of Function Evaluations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rational</td>
</tr>
<tr>
<td>1</td>
<td>64</td>
</tr>
<tr>
<td>2</td>
<td>71</td>
</tr>
<tr>
<td>3</td>
<td>81</td>
</tr>
<tr>
<td>4</td>
<td>91</td>
</tr>
<tr>
<td>5</td>
<td>130</td>
</tr>
<tr>
<td>6</td>
<td>77</td>
</tr>
</tbody>
</table>

It can be seen from the table above that the rational model needed 26.3% less computational effort overall than the quadratic model.
Figure (xi)
St. point 2

Function Evaluations \times 10^3
Figure (xii)

St. point 3

Function Evaluations
Figure (xiii)
St. point 4

Function Evaluations
Figure (xiv)
St. point 5
Figure (xv)

St. point 6

+ Rational Model

0 Quadratic Model
The last Figure of this chapter presents a concise comparison of the models (3.5.1) and the quadratic one over seven standard test functions.

The x-axis represents the problem under consideration and the y-axis the number of function evaluations needed for convergence. Inspection of this figure shows that the rational model behaves slightly better for the two-dimensional problems (1,2,3), about the same for the three-dimensional one (4) and shows a considerable improvement for four-dimensional problems (5,6,7).
3.6 DISCUSSION.

In this chapter new models for a multi-dimensional unconstrained optimization algorithm have been proposed. Although the algorithm can be classified as a "direct search method", it does not generate search directions, but solves a constrained minimization problem instead. To discuss the convergence of this algorithm the following difference should be pointed out between the SRM and other algorithms. The SRM algorithm proceeds by minimizing a sequence of rational models within bounded regions, while all the other algorithms do one-dimensional searches along lines. Therefore the distance which SRM can travel in any iteration is limited. Because of this if the initial point is far away from the minimum the SRM will require many iterations until the progress of the basepoint and the growth of the sphere of validity permit the sphere to enclose the minimum.

If the surface of the objective function has a winding valley, SRM may profit by its ability to model non-convex contours.

The model (3.2.1) has been compared with the quadratic model proposed in Winfield (1973), using the same computer and the same algorithm. The results of these comparisons indicate that since the exact constrained minimization of the model (3.2.1) is time-consuming, the quadratic model is more efficient in terms of function evaluations. Furthermore, it should be noted that rational models which require the same computational effort as the quadratic ones, proved to be more efficient in many cases, especially when the objective function is either highly nonlinear (e.g. Problem (8)) or rational itself (e.g. Problems (9) and (10)). In such cases the rational model (3.5.1) gives more satisfactory results than the quadratic one.
APPENDIX

Test function 1, ROS(2), Rosenbrock (1960):

\[ f(x) = 100 \left( x_2^2 - x_1^2 \right)^2 + (1 - x_1)^2. \]
Starting approximation : \( x = (-1.2, 1.0) \), \( x^* = (1, 1) \)

Test function 2, CUBIC, White and Holst (1964):

\[ f(x) = 100 \left( x_2^3 - x_1^2 \right)^2 + (1 - x_1)^2. \]
Starting approximation : \( x = (-1.2, 1.0) \), \( x^* = (1, 1) \)

Test function 3, BEALE, Beale (1958):

\[ f(x) = \left[ 1.5 - x_1(1-x_2) \right]^2 + \left[ 2.25 - x_1(1-x_2^2) \right]^2 + \left[ 2.625 - x_1(1-x_2^3) \right]^2 \]
Starting approximation : \( x = (1.0, 0.8) \), \( x^* = (3.0, 0.5) \)

Test function 4, BOX, Box (1966):

\[ f(x) = \sum_{i=1}^{10} \left[ \exp(-x_1 t_i) - \exp(-x_2 t_i) - x_3 \left( \exp(-t_i) + \exp(-10t_i) \right) \right]^2 \]
\[ (t_i = 0.1, 0.2, \ldots, 1.0) \]
Starting approximation : \( x = (0, 10, 20) \), \( x^* = (1, 10, 1) \)

Test function 5, POW, Powell (1964):

\[ f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4 \]
Starting approximation : \( x = (3, -1, 0, 1) \), \( x^* = (0, 0, 0, 0) \)
Test function 6, WOOD, quoted by Colville (1968):

\[ f(x) = 100(x_2 - x_1)^2 + (1 - x_1)^2 + 90(x_4 - x_3)^2 + (1 - x_3)^2 + \\
+ 10.1 \left[ (x_2 - 1)^2 + (x_4 - 1)^2 \right] + 19.8(x_2 - 1)(x_4 - 1) \]

Starting approximation: \( x = (-3, -1, -3, -1) \), \( x^* = (1, 1, 1, 1) \)

Test function 7, MIELE & CANTRELL, Miele & Cantrell (1969):

\[ f(x) = (\exp(x_1) - x_2)^4 + 100(x_2 - x_3)^6 + \left[ \tan(x_3 - x_4) \right]^4 + x_1^8 \]

Starting approximation: \( x = (1, 2, 2, 2) \), \( x^* = (0, 1, 1, 1) \)
CHAPTER IV

Conjugate Gradient Algorithms using a Non-Quadratic Model with Exact or Inexact Line Searches
4.1 INTRODUCTION.

Until now derivative-free methods were discussed, for the solution of the problem (P1). It is, however, most helpful - when a reasonable amount of computational effort is required - to make use of gradients when minimizing a function mainly because of the information included about the function.

In the present and the next chapter, two of the most important unconstrained minimization methods will be considered. This chapter focuses upon the conjugate gradient method.

The method examined in this chapter belongs to the more general class of conjugate direction methods. The reason they are called the "conjugate direction methods" is because in the case of the quadratic function:

\[ q(x) = \frac{1}{2} x^T Q x + b^T x + c \]  

(4.1.1)

where \( Q \) is a symmetric positive definite \( n \times n \) matrix, they calculate the exact minimum after finite number of iterations and if exact arithmetic is used they generate "conjugate directions".

Definition (4.1.1):

A set of nonzero vectors \( \{ \xi_i \} \) are said to be conjugate with respect to the quadratic function (4.1.1) if they satisfy the relation:

\[ \xi_i^T Q \xi_j = 0 , \text{ for } i \neq j. \]

The first conjugate gradient method was published by Hestenes and Stiefel in 1952, for solving a system of linear algebraic equations. Fletcher and Reeves (1964) were amongst the first to
use this technique to minimize a function of several variables. Since then the method has been used successfully on many problems.

The conjugate gradient algorithm applied to a function $f(x)$ of class $C^1$ has the following form:

Algorithm (A): Given $x_0 \in \mathbb{R}^n$, an initial estimate of the minimizer $x^*$, let:

$$d_0 = -g(x_0) = -g_0$$

then for $i = 0, 1, 2, \ldots$ set

$$x_{i+1} = x_i + a_i \, d_i$$

$$d_{i+1} = -g_{i+1} + \beta_i \, d_i$$

where $\beta_i$ is one of the following expressions:

$$\beta_i = \frac{\|g_{i+1}\|^2}{\|g_i\|^2} \quad \text{(Fletcher-Reeves)}$$

$$\beta_i = \frac{d_i^T (g_{i+1} - g_i)}{d_i^T (g_{i+1} - g_i)} \quad \text{(Hestenes -Stiefel)}$$

$$\beta_i = \frac{-g_{i+1}^T g_i + 1}{d_i^T g_i} \quad \text{(Dixon 1972)}$$

$$\beta_i = \frac{g_{i+1}^T (g_{i+1} - g_i)}{g_i^T g_i} \quad \text{(Polak-Ribiére)}$$

and $a_i$ minimizes $f(x_{i+1})$. 

Definition (4.1.2):

For algorithm \((A)\) it will be said that \(a_i\) is an exact minimizer if

\[
f(x_i + a_i d_i) \leq f(x_i + t d_i) \quad \text{for all } t \in [0, \infty)
\]

Since for the first minimization problem \((P1)\) it is impossible to calculate \(a_i\) exactly in general, there is an accumulation of errors in an actual computation which can affect the convergence property. Several algorithms using conjugate gradient methods select \(a_i\) to be the exact minimizer of \(f(x_i + a_i d_i)\). They will be referred to as conjugate gradient algorithms using exact line searches. However for some algorithms it is adequate that \(a_i\) is such that the descent condition:

\[
f(x_i + a_i d_i) < f(x_i)
\]  

(4.1.2)

is satisfied at each step. Such algorithms will be referred to as conjugate gradient algorithms using inexact (or inaccurate) line searches.

Remark: A direction \(d\) is said to be a descent direction for \(f\) at \(x\) if:

\[
g^T(x) d < 0
\]

(4.1.3)
Polak and Ribièrè (1969) using $\beta_i$ defined by

$$
\beta_i = \frac{g_{i+1}^T(g_{i+1} - g_i)}{||g_i||^2}
$$

obtained quadratic convergence for a function of class $C^3$ whose Hessian matrix has a positive and bounded spectrum, (Polak (1971)). Ortega and Rheinboldt (1970) obtained a convergence result for a function of class $C^3$ whose Hessian matrix has a positive spectrum. Superlinear convergence was shown by McCormick and Pearson (1969) under the assumption of the almost conjugacy property for a function of class $C^2$ whose Hessian matrix has a bounded and positive spectrum. Powell (1977a) reports a numerical test where the expression of $\beta_i$ given by Polak-Ribièrè is more efficient than the expression given by Fletcher-Reeves. In the same paper some new theoretical results are given in favour of the Polak-Ribièrè formula $\beta_i$ and an algorithm is proposed with a restart procedure that takes account of the objective function automatically.

In the next paragraph the implementation of a nonquadratic model follows which is the basis for two conjugate gradient algorithms using exact and inexact line searches.
4.2 THE NEW MODEL AND THE USE OF RESTARTS.

Amongst the existing conjugate direction algorithms the updating process rarely takes into account any deviation of the objective function from quadratic behaviour. However, quadratic models may not always be adequate to incorporate all the information which might be needed to represent the objective function \( f(x) \) successfully.

A more general model than quadratic is proposed in this chapter as a basis for a conjugate gradient algorithm. The model which is the quotient of two quadratic functions, belongs to the class of rational functions and it is obtained by a special nonlinear scaling of a quadratic function.

If \( q(x) \) is the quadratic function (4.1.1) then a function \( f \) is a nonlinear scaling of \( q(x) \) if the following conditions hold:

\[
f = f(q(x)), \quad \frac{df}{dq} > 0 \quad \text{for } x \neq x^*
\]

where \( x^* \) is the minimizer of \( q(x) \). The following properties are immediately derived from the above conditions:

(a) every contour line of \( q(x) \) is a contour line of \( f \);

(b) if \( x^* \) is a minimizer of \( q(x) \), then it is also a minimizer of \( f \).

Various authors have published work in this area. A conjugate direction method which minimizes the function

\[
f(x) = (q(x))^p, \quad p > 0 \quad \text{and} \quad x \in \mathbb{R}^n,
\]

in at most \( n \) steps has been described by Fried (1971). The
following minimization problem:

$$\text{minimize } f(x) = f(q(x))$$

where

$$\left\{ \frac{df}{dq} = f' > 0 \text{ and } q > 0 \right\} \quad (4.2.1)$$

has been considered by Goldfarb (1972). The property of the nonlinear scaling $f(q(x))$, given by the relation (4.2.1), has been called invariancy to nonlinear scaling by Spedicato (1976). The special case

$$f(q(x)) = \varepsilon_1 q(x) + \frac{\varepsilon_2}{2} q^2(x),$$

where $\varepsilon_1$ and $\varepsilon_2$ are scalars, has been investigated by Boland et al. (1979). In this chapter the new model is investigated and tested on a set of standard functions. (See Appendix for details of these functions). It is assumed that conditions (4.2.1) hold and an extended conjugate gradient algorithm is developed which is based on the model:

$$f(q(x)) = \frac{\varepsilon_1 q(x) + 1}{\varepsilon_2 q(x)} \quad (4.2.2)$$

and so clearly $\varepsilon_2 < 0$. It is first observed that $q(x)$ and $f(q(x))$ given by (4.2.2) have identical contours with different numerical values and the same unique minimum point which is denoted by $x^*$. For any $f$ satisfying the conditions (4.2.1) it is shown in Boland et al. (1979) that the updating process given by Algorithm (B) below generates identical conjugate directions and the same sequence of approximations $x_i$ to the minimizer $x^*$, as the original method of Fletcher-Reeves, when applied to $f(x) = q(x)$. 
Algorithm (B): Given \( x_0 \in \mathbb{R}^n \) an initial estimate of the minimizer \( x^* \), let:

\[
d_0 = -g_0
\]

\[
x_{i+1} = x_i + a_i d_i \quad i \geq 1
\]

\[
d_{i+1} = -g_{i+1} + \beta_i d_i \quad i \geq 1
\]

where a numerical expression for \( \rho_i \) used in the modified formulae below, will be given at the end of this section.

\( \beta_i \) is one of the following expressions:

\[
\beta_i = \frac{\rho_i \|g_{i+1}\|^2}{\|g_i\|^2} \quad \text{(modified Fletcher-Reeves)}
\]

\[
\beta_i = \frac{\sum_{j=1}^{i} (\rho_i g_{i+1} - g_i)^T d_i^T (\rho_i g_{i+1} - g_i)}{d_i^T (\rho_i g_{i+1} - g_i)} \quad \text{(modified Hestenes-Stiefel)}
\]

\[
\beta_i = -\frac{\rho_i \|g_{i+1}\|^2}{d_i^T g_i} \quad \text{(modified Dixon)}
\]

\[
\beta_i = \frac{\sum_{j=1}^{i} (\rho_i g_{i+1} - g_i)^T g_i}{g_i^T g_i} \quad \text{(modified Polak-Ribiere)}
\]

where \( a_i \) is as before (Algorithm (A)).
Conjugate gradient methods are usually implemented with restarts in order to avoid an accumulation of errors. Fletcher and Reeves (1964) suggested resets (or restarts) every $n$ iterations, where $n$ is the number of the unknown variables.

The numerical performance of the conjugate gradient algorithms is greatly improved by using restarts. In the usual reset one defines:

$$d_k = -g_k \quad \text{for} \quad k = 0, n, 2n, \ldots$$

It was shown by Cohen (1972) that several restarted conjugate gradient methods have $n$-step quadratic convergence. It was pointed out by Crowder and Wolfe (1972) that if restarting is not employed the convergence is only linear. They came to the conclusion that conjugate gradient methods without restarts converge no better than linearly for the case of a quadratic function and linearly for a general nonlinear function. Powell (1976) showed that for a convex quadratic function, under certain conditions, the convergence rate is linear.

A restart direction different from the negative gradient was proposed by Beale (1972), which can be used to devise a sophisticated restart procedure. The motivation for the use of another starting direction is that an increase of the immediate reduction in the function value is possible when using conjugate gradient methods to minimize a nonquadratic objective function.

Beale's algorithm defines an iteration of $n$-steps. When the iteration is completed a new direction, $d_0$, is defined and
a new cycle is started. It can be noted that if \( d_0 \) is the negative gradient the usual conjugate gradient method is obtained with restarts. Therefore, it seems to be in general agreement that restarting is very helpful in practice and different restarting procedures have been studied and analysed.

The implementation of the extended conjugate gradient method has been performed for general functions \( f(q(x)) \) of the form of (4.2.2). The unknown quantities \( \rho_i \) were expressed in terms of available quantities of the algorithm (i.e. function and gradient values of the objective function).

It is first assumed that neither \( \epsilon_1 \) nor \( \epsilon_2 \) is zero in (4.2.2). Solving (4.2.2) for \( q(x) \) then:

\[
q_i = \frac{1}{\epsilon_2 \frac{\dot{f}}{f} - \epsilon_1} \quad (4.2.4)
\]

and using the expression for \( \rho_i \):

\[
\rho_i = \frac{\frac{\dot{f}}{f}}{\frac{\ddot{f}}{f}} = \left( \frac{f_{i-1} - \epsilon_1/\epsilon_2}{f_i - \epsilon_1/\epsilon_2} \right)^2 \quad (4.2.5)
\]

The quantity which has to be determined explicitly is \( \epsilon_1/\epsilon_2 \). During every iteration \( \epsilon_1/\epsilon_2 \) must be evaluated as a function of known available quantities. From the relations

\[
\begin{align*}
\dot{g}_{i-1} &= f_{i-1}^1 Q(x_{i-1} - x^*) \\
g_i &= f_{i}^1 Q(x_i - x^*)
\end{align*}
\quad (4.2.6)
\]

and (4.2.2), it follows that:
\[
\rho_i = \frac{g_{i-1}^T(x_i - x^*)}{g_i^T(x_i - x^*)}.
\] (4.2.7)

Then
\[
g_{i-1}^T(x_i - x^*) = g_{i-1}^T(x_{i-1} + a_{i-1} d_{i-1} - x^*) =
\]
\[
= g_{i-1}^T(x_{i-1} - x^*) + a_{i-1} g_{i-1}^T d_{i-1}
\] (4.2.8)

and
\[
g_i^T(x_i - x^*) = g_i^T(x_i - x^*)
\] (4.2.9)

since
\[
g_i^T d_{i-1} = 0.
\]

Therefore \( \rho_i \) can be expressed as:
\[
\rho_i = \frac{g_{i-1}^T(x_{i-1} - x^*) + a_{i-1} g_{i-1}^T d_{i-1}}{g_i^T(x_i - x^*)}. (4.2.10)
\]

From (4.2.6) and (4.2.10) it follows that:
\[
\rho_i = \frac{2f'_{i-1} q_{i-1} + a_{i-1} g_{i-1}^T d_{i-1}}{2f'_{i} q_i} =
\]
\[
= \rho_i \left( \frac{q_{i-1}}{q_i} \right) + a_{i-1} g_{i-1}^T d_{i-1}/2f'_{i} q_i. (4.2.11)
\]

The quantities \( \frac{q_{i-1}}{q_i} \) and \( f'_{i} q_i \) can be rewritten as:
\[
(\frac{q_{i-1}}{q_i}) = (1/\sqrt{\rho_i}) (4.2.12)
\]
\[
f'_{i} q_i = (1/\epsilon_2 q_i) = -(\frac{f_i}{\epsilon_1} - \frac{\epsilon_i}{\epsilon_2}). (4.2.13)
\]
Substituting (4.2.12) and (4.2.13) in (4.2.11) gives:

\[ \rho_i = \sqrt{\rho_{i-1}} + \left( a_{i-1} g_{i-1}^T d_{i-1} \right) / 2(\varepsilon_1 / \varepsilon_2 - f_i) \]. \hspace{1cm} (4.2.14)

From (4.2.14) and (4.2.5) it follows that:

\[ \frac{f_{i-1} - \varepsilon_1 / \varepsilon_2}{f_i - \varepsilon_1 / \varepsilon_2} = \frac{f_{i-1} - \varepsilon_1 / \varepsilon_2}{f_i - \varepsilon_1 / \varepsilon_2} + \frac{a_{i-1} g_{i-1}^T d_{i-1}}{2(\varepsilon_1 / \varepsilon_2 - f_i)} \]. \hspace{1cm} (4.2.15)

Equation (4.2.15) can be rewritten as:

\[ (f_{i-1} - \varepsilon_1 / \varepsilon_2)^2 = (f_{i-1} - \varepsilon_1 / \varepsilon_2)(f_i - \varepsilon_1 / \varepsilon_2) - \frac{a_{i-1} g_{i-1}^T d_{i-1}}{2}(f_i - \varepsilon_1 / \varepsilon_2) \]. \hspace{1cm} (4.2.16)

Using the following transformations:

\[ f_{i-1} - \varepsilon_1 / \varepsilon_2 = y \], \hspace{1cm} f_i - f_{i-1} = w

the above equation can be written as:

\[ f_i - \varepsilon_1 / \varepsilon_2 = f_i + (f_{i-1} - \varepsilon_1 / \varepsilon_2) - f_{i-1} = (y + w) \]

\[ y^2 = y(w+y) + (w+y) \left( - \frac{a_{i-1} g_{i-1}^T d_{i-1}}{2} \right) \]. \hspace{1cm} (4.2.17)

Finally,

\[ y = w \left( - \frac{a_{i-1} g_{i-1}^T d_{i-1}}{2} \right) / (w + \left( - \frac{a_{i-1} g_{i-1}^T d_{i-1}}{2} \right) \}. \hspace{1cm} (4.2.18)

Therefore, from (4.2.18) the quantity \( \varepsilon_1 / \varepsilon_2 \) can be expressed as:

\[ \varepsilon_1 / \varepsilon_2 = f_{i-1} - \frac{(f_i - f_{i-1})(- a_{i-1} g_{i-1}^T d_{i-1}/2)}{(f_i - f_{i-1}) + (- a_{i-1} g_{i-1}^T d_{i-1}/2)} \]. \hspace{1cm} (4.2.19)
The cases $e_1 = 0$ or $e_2 = 0$ are now considered. If $e_2 = 0$ then it is obvious that model (4.2.2) is not valid and a quadratic model is used instead, for which the corresponding $p_i$ equals one. If $e_1 = 0$ then $p_i = \sqrt{f_{i-1}/f_i}$. The information about $e_1$ or $e_2$ taking the zero value can be provided from equation (4.2.18).

Using the transformation:

$$(-a_{i-1} g_{i-1} d_{i-1})^T = 2n$$

then equation (4.2.19) can be rewritten as:

$$\frac{e_1/e_2}{f_{i-1}} = n - \frac{n w}{n+w} \quad (4.2.20)$$

If neither $(n+w)$ nor $(f_{i-1}(n+w) - n w)$ vanishes then neither $e_1$ nor $e_2$ is zero. Assuming that $(n+w) = 0$ then $w = -n$ and

$$f_{i-1}(n+w) = -n w = n^2 = \left(-a_{i-1} g_{i-1} d_{i-1}/2\right)^2 \neq 0 \quad (4.2.21)$$

Equation (4.2.20) can be rewritten as:

$$e_1(n+w) = e_2(f_{i-1}(n+w) - n w) \quad (4.2.22)$$

From (4.2.22) and (4.2.21) and the assumption that $(n+w) = 0$ it follows that:

$$e_2 = 0 \quad (4.2.23)$$

If $(f_{i-1}(n+w) - n w) = 0$ then $(n+w) = \frac{n w}{f_{i-1}} \neq 0$ and from (4.2.22) $e_1 = 0$. Therefore the following algorithm is obtained to calculate $p_i$ and $d_i$ at each step: (It is clear that $f_i$ denotes...
the objective function under consideration and \( g_i \) its gradient at the point \( x_i \).

(i) Define

\[
\begin{align*}
\psi = \frac{- (a_{i-1}^T g_{i-1} d_{i-1})}{2} \\
\omega = f_i - f_{i-1}
\end{align*}
\]

and compute: \((\psi + \omega), ((\psi + \omega) f_{i-1} - \omega)\).

(ii) If \(|\psi + \omega| \leq \text{EPS}\), then set \( p_i = 1 \) and go to step (v), where EPS is a small number (i.e. 0.1E-10).

(iii) If \(|(\psi + \omega) f_{i-1} - \omega| \leq \text{EPS}\) then \( p_i = \sqrt{f_{i-1}/f_i} \) and go to step (v).

(iv) Compute:

\[
\rho_i = \left( \frac{n}{2n + \omega} \right)^2
\]

(v) Calculate the new direction:

\[
d_i = - g_i + \rho_i \left( \frac{\|g_i\|^2}{\|g_{i-1}\|^2} \right) d_{i-1}
\]

This direction is then used instead of the direction used in the standard Fletcher-Reeves formula and since the new model satisfies conditions (4.2.1) the resulting algorithm has finite convergence on the model (4.2.2).

4.2.1 Convergence of the Conjugate Gradient Method - Accurate Line Searches.

In this section convergence results for conjugate gradient methods using accurate line searches will be given, when the proposed new model (4.2.2) is considered. Throughout this section

\( \dagger \) It is clear that any other modified formula given in Algorithm (B) can be used instead of the Fletcher-Reeves one.
Algorithm (B) will be considered as the conjugate gradient algorithm with restarts every \( m \) iterations (where \( m \) is usually a positive integer greater than or equal to the number of variables).

**Definition (4.2.I.1):**

A non empty set of the form

\[
S(x_0) = \left\{ x \in \mathbb{R}^n : f(x) \leq f(x_0) \right\},
\]

where \( f \) is a real valued function, is called a level set of \( f \) at \( x_0 \). Some useful properties of \( S(x_0) \) are given below.

**Lemma (4.2.I.1).**

Let \( f \) be a convex real valued function in \( \mathbb{R}^n \). Then the level set is convex. (Proof in the Appendix).

As an immediate consequence of the fact that a continuous function on a compact set takes on its minimum, it is observed that whenever \( f \) has a compact level set then \( f \) has a global minimizer (Ortega and Rheinboldt, 1970, p.104). The conditions on the function \( f \) will be examined next which will ensure that \( f \) has a compact level set. The compactness of the level set is essentially equivalent to the existence of the minimizer. Therefore, in general, it is difficult to give useful conditions for \( f \) which ensure that some level set is compact. However, if the spectrum of the Hessian matrix of the function \( f \) is bounded below by a positive constant, then it can be show that \( S(x_0) \) is compact.

* (This is a well known result therefore the proof is omitted from the text).
Lemma (4.2.1.2):

Let $f$ be a real valued function from the $C^2$ class on $\mathbb{R}^n$ and the spectrum of the Hessian matrix of $f(x)$ be bounded below by $\mu > 0$ on $\mathbb{R}^n$. Then the level set

$$S(x_0) = \{ x \in \mathbb{R}^n : f(x) \leq f(x_0) \}$$

is compact. (Proof in the Appendix).

It can now be proved that the conjugate gradient algorithm (B) is well defined, i.e.

$$x_i \in S(x_0), \forall i = 1, 2, \ldots .$$

Lemma (4.2.1.3):

Let $f$ be a vector function of class $C^2$ on an open convex set $D$ containing the level set $S(x_0)$ and $S(x_0)$ be bounded. Then Algorithm (B) is well defined.

Proof: The proof will be accomplished inductively.

$x_0 \in S(x_0)$, $d_0 = -g_0$ and $x_1 = x_0 - a_0 g_0$. Moving along the direction of steepest descent a scalar $a_0$ is selected so it gives the minimum of the function $f(x)$ in this direction, i.e.:

$$f(x_0 - a_0 g_0) \leq f(x_0),$$

therefore, $x_1 \in S(x_0)$.

Assume that $x_0, \ldots, x_{i-1}$ are well defined and lie in $S(x_0)$.

Because

$$-d_i^T g_i = -(-g_i^T + \rho_i \beta_i d_{i-1}^T)g_i = g_i^T g_i = \|g_i\|^2,$$

then

$$-d_i^T g_i = 0.$$
implies that 

$$x_{i+1} = x_i,$$ so $$x_{i+1} \in S(x).$$

Suppose, therefore that 

$$- d_i^T g_i > 0.$$ 

Set 

$$\hat{a}_i = \sup_{a \in J_i} a,$$ 

where 

$$J_i = \left\{ a > 0 : \left[ x_i, x_i + a d_i \right] \subseteq D \right\} \text{ and}$$ 

$$f(x_i + \lambda d_i) < f(x_i), \quad \forall \lambda \in (0, \hat{a}_i).$$ 

If $$J_i$$ is an empty set, then 

$$\frac{f(x_i + a d_i) - f(x_i)}{a} \geq 0 \quad \text{for all } a > 0.$$ 

If the limit is taken as $$a$$ tends to zero this gives: 

$$- d_i^T g_i = \| g_i \|^2 > 0,$$ 

which contradicts the fact that $$(-d_i^T g_i)$$ is only a positive number. 

Hence $$\hat{a}_i$$ is well defined. Moreover since $$S(x_0)$$ is compact, 

$$\hat{a}_i < \infty \quad \text{and} \quad \left[ x_i, x_i + \hat{a}_i d_i \right] \subseteq S(x_0).$$ 

Now suppose that 

$$f(x_i + \hat{a}_i d_i) < f(x_i).$$ 

Then since $$D$$ is open and $$f$$ is continuous a $$\delta > 0$$ may be chosen such that 

$$x_i + a d_i \in D \quad \text{and} \quad f(x_i + a d_i) < f(x_i),$$ 

for 

$$a \in \left[ \hat{a}_i, \hat{a}_i + \delta \right].$$ 

* The set $$[x_i, x_i + a d_i]$$ represents the line segment joining the points $$x_i$$ and $$x_i + a d_i$$ along the direction $$d_i$. 
This contradicts the definition of $\hat{a}_i$ and hence
\[
f(x_i) = f(x_i + \hat{a}_i d_i).
\]

By the mean value theorem, there exists a $a^*_i \in (0, \hat{a}_i)$ such that
\[
d^T_i g(x_i + a^*_i d_i) = 0.
\]

If $a^*_i$ is chosen as the least positive $a^*_i$ for which the above equality holds so
\[
x_{i+1} \in S(x_0).
\]

The stability of the conjugate gradient method using Algorithm (B) is indicated by the proof of the previous Lemma and can be stated as the following corollary.

**Corollary (4.2.I.1):** Let $f$ be a function satisfying the conditions of Lemma (4.2.I.3). Then the conjugate gradient Algorithm (B) is stable, i.e. if
\[
g(x_i) \neq 0 \text{ then } f(x_{i+1}) < f(x_i).
\]

In any iterative method using line searches, the knowledge of upper and lower bounds on $a_i$ is desirable for showing the rate of convergence of the method. The bounds on $a_i$ for the CG method are shown in the following lemma.

**Lemma (4.2.I.4):**

Let $f$ be a real valued vector function of class $C^2$ on an open convex set $D$ which contains $S(x_0)$, where $x_0$ is an arbitrary initial point chosen for Algorithm (B). Let the spectrum of the Hessian
matrix \( G(x) \) be bounded below by \( \mu > 0 \) and above by \( \lambda \) for all \( x \in S(x_0) \). Then the stepsize \( a_i \) satisfies the inequality:

\[
\frac{1}{\lambda} \frac{\|g_i\|^2}{\|d_i\|^2} \leq a_i \leq \frac{1}{\mu} \frac{\|g_i\|^2}{\|d_i\|^2} \leq \frac{1}{\mu} \quad (4.2.1.2)
\]

Proof: Let

\[
\phi(a_i) = d_i^T g(x_i + a_i d_i) \quad .
\]

Then \( \phi(0) = d_i^T g_i = -\|g_i\|^2 \) and \( \phi'(a) = d_i^T G(x_i + a d_i)d_i \quad a \in (0, a_i) \)

implies that \( \mu \|d_i\|^2 \leq \phi'(a) \leq \lambda \|d_i\|^2 \).

Hence \( a_{\min} \leq a_i \leq a_{\max} \)

where \( a_{\min} = \frac{\|g_i\|^2}{\lambda \|d_i\|^2} \)

and \( a_{\max} = \frac{\|g_i\|^2}{\mu \|d_i\|^2} \).

But \( d_i = -g_i + \rho_i \beta_i d_{i-1} \)

so \( \|d_i\| \geq \|g_i\| \quad \forall \ i = 1, 2, \ldots \).

Hence \( a_{\max} \leq \frac{1}{\mu} \).

It can be noticed that the upper bound on \( a_i \) is independent of \( i \), but the same cannot be said for the lower bound.
It will next be shown that the norm of the direction vector is bounded by a scalar multiple of the norm of the gradient vector at every iteration. This will be helpful in the following convergence results.

Lemma (4.2.1.5):

Let $f$ be a real valued vector function of class $C^2$ on an open convex set $D$ which contains $S(x_0)$. Let the spectrum of the Hessian matrix $G(x)$ be bounded below and above by $\mu > 0$ and $\lambda$ respectively for all $x \in S(x_0)$. Then for all $i = 0, 1, 2, \ldots$ there holds:

$$\|d_i\| \leq (q + 1)(1 + \frac{\lambda}{\mu})^q \|g_i\|$$

(4.2.1.3)

(where $q$ is the number of iterations after which the algorithm is restarted, using the Fletcher-Reeves formula).

Proof: If Algorithm (B) is considered with $\beta_i$ the Fletcher-Reeves formula and $i_0$ denotes the largest integer not exceeding $i$ which is congruent to zero modulo $q$, then:

$$d_i = -g_i + \rho_i \frac{\|g_i\|^2}{\|g_{i-1}\|^2} d_{i-1}$$

$$= -g_i - g_{i-1} \rho_i \frac{\|g_i\|^2}{\|g_{i-1}\|^2} + \rho_i \rho_{i-1} \frac{\|g_i\|^2}{\|g_{i-2}\|^2} d_{i-2}$$

$$= -\sum_{j=i_0}^{i} \frac{\|g_i\|^2}{\|g_j\|^2} g_j \prod_{k=j+1}^{i} \rho_k$$
Hence

\[ \| d_i \| \leq \sum_{j=i_0}^i \frac{\| g_i \|^2}{\| g_j \|^2} g_j \sum_{k=j+1}^{i} \rho_k = \| g_i \| \sum_{j=i_0}^i \frac{\| g_i \|}{\| g_j \|} \sum_{k=j+1}^{i} \rho_k. \]

Therefore

\[ \| d_i \| \leq \| g_i \| (i - i_0 + 1) \max_{i_0 \leq j \leq i} \frac{\| g_i \|}{\| g_j \|}. \]

But

\[ \frac{\| g_i \|}{\| g_j \|} = \frac{\| g_i \|}{\| g_{i-1} \|} \cdots \frac{\| g_{j+1} \|}{\| g_j \|} \leq \left( 1 + \frac{\lambda}{\mu} \right)^{i-j}, \]

and because \( q \geq i-j \) \( \forall \) \( i_0 \leq j \leq i \)

then

\[ \| d_i \| \leq \| g_i \| (q+1) \left( 1 + \frac{\lambda}{\mu} \right)^q. \]

A lemma concerning upper bounds on \( \beta_i \) and the ratio

\[ \frac{\| d_{i+1} \|}{\| d_i \|} \]

is next stated.

**Lemma (4.2.1.6):**

Let \( f \) be a vector function of \( C^2 \) class on an open convex set \( D \) and \( S(x_0) \subset D \). Assume that the spectrum of \( G(x) \) is bounded below and above by \( \mu > 0 \) and \( \lambda \) respectively for all \( x \in S(x_0) \).

Then both

\[ \frac{\| d_{i+1} \|}{\| d_i \|} \quad \text{and} \quad \beta_i = \frac{\| g_{i+1} \|^2}{\| g_i \|^2} \]

are bounded above for all \( i \). (Proof in the Appendix).
It was shown in Corollary (4.2.1.1) that the conjugate gradient Algorithm (B) is stable, i.e.

\[ f(x_{i+1}) < f(x_i) \quad i = 0, 1, 2, \ldots \] (4.2.1.4)

Therefore if \( f(x) \) is bounded below then this implies that the sequence \( \{f(x_i)\} \) converges downward to a limit. This by itself does not imply the convergence of either \( \{\|g_i\|\} \) or \( \{x_i\} \).

However if an inequality of the form

\[ \|g_i\| > c \|d_i\|, \quad c > 0 \quad i = 0, 1, 2, \ldots \] (4.2.1.5)

holds in addition to (4.2.1.4) then it can be shown that

\[ \lim_{i \to \infty} \|g_i\| = 0 . \] (4.2.1.6)

If \( f(x) \) is also strictly convex, it can further be shown that

\[ \lim_{i \to \infty} x_i = z, \quad g(z) = 0 . \] (4.2.1.7)

The inequality (4.2.1.5) holds for such methods as the method of steepest descent and the conjugate gradient Algorithm (B). The main theorems of the convergence of the conjugate gradient Algorithm (B) follow next.

**Theorem (4.2.1.1):**

Let \( f \) be a vector function of \( C^2 \) class on an open convex set \( D \), containing the bounded set \( S(x_0) \) and let the spectrum of \( G(x) \) be bounded below and above by \( \mu > 0 \) and \( \lambda \) respectively for all \( x \in S(x_0) \). Let \( z \) be the unique minimizer of \( f(x) \). Then the sequence \( \{x_k\} \) defined by Algorithm (B) is well defined, lies
in \( S(x_0) \) and satisfy

\[
\lim_{k \to \infty} x_k = z
\]

where \( z \) is the unique critical point of \( f \) in \( S(x_0) \).

The proof of this Theorem follows the same lines as the proof in Ortega and Rheinboldt (1970), pp. 509-510.

If the convexity assumptions on \( f(x) \) of the previous theorem are removed a weaker version of Theorem (4.2.I.1) can be proved.

**Theorem (4.2.I.2):**

Let \( f \) be of class \( C^1 \) on an open convex set \( D \) containing \( S(x_0) \). If the set \( S(x_0) \) is bounded and \( g \) is uniformly Lipschitz continuous on \( S(x_0) \). Then

\[
\inf_i \| g_i \| = 0.
\]

**Proof:** Suppose that

\[
0 < \rho = \inf_i \| g_i \|
\]

and let

\[
M = \max \left\{ \| g(x) \| : x \in S(x_0) \right\}.
\]

Then from Lemma (4.2.I.5):

\[
\| d_i \| \leq (q+1) M \| g_i \| /\rho. \tag{4.2.I.8}
\]

Let \( L \) be the Lipschitz constant of \( g \) so that:

\[
\| g(x) - g(y) \| \leq L \| x - y \| \quad \forall x, y \in S(x_0). \tag{4.2.I.9}
\]
Define
\[ \delta_i = -\frac{T g_i d_i}{2L \|d_i\|^2} \]
then by (4.2.1.8):
\[ \delta_i = \frac{\|g_i\|^2}{2L \|d_i\|^2} \geq \frac{\rho^2}{2L(q+1)^2 M} \]  \hspace{1cm} (4.2.1.10)

Now, for 0 \leq \alpha \leq \delta , from (4.2.1.9):
\[ \|g_i - g(x_i + \alpha d_i)\| \leq \alpha \|d_i\| \leq -\frac{g_i T d_i}{2 \|d_i\|} \]  \hspace{1cm} (4.2.1.11)

Hence
\[ |( - g_i^T d_i) - (- g_i^T (x_i + \alpha d_i) d_i)| \leq -\frac{g_i^T d_i}{2} \]  \hspace{1cm} (4.2.1.12)

From (4.2.1.12):
\[ - g_i^T (x_i + \alpha d_i) d_i \geq -\frac{g_i^T d_i}{2} \]  \hspace{1cm} (4.2.1.13)

for 0 \leq \alpha \leq \delta .

Because of the relation:
\[ f(x_i) - f(x_i + \alpha d_i) > -\frac{g_i^T d_i \delta_i}{2} \]  \hspace{1cm} (4.2.1.14)

(Proof in the Appendix), and from (4.2.1.10):
\[ f(x_i) - f(x_i + \alpha d_i) > \frac{\rho^4}{4L(q+1)M^2} \]  \hspace{1cm} (4.2.1.15)

Therefore:
\[ f(x_{i+1}) < f(x_i) - \frac{\rho^4}{4L(q+1)M^2} \quad \forall \ i = 0, 1, 2, \ldots \]
which contradicts the fact that \( f(x) \) is bounded below. Hence

\[
\inf_i \|g_i\| = 0.
\]

The rate of convergence of the conjugate gradient Algorithm (B) can be shown in a similar way to that in Cohen's (1972) paper that it is \( n \)-step rational for a large class of functions. The main convergence theorem of the above mentioned paper is the following.

**Theorem (4.2.I.3):**

Let \( f \) be a real valued vector function of \( C^3 \) class and the spectrum of the Hessian matrix \( G(x) \) of \( f(x) \) be bounded below and above by \( \mu > 0 \) and \( \lambda \) respectively. Then for the sequence \( \{x_i\} \) formed by Algorithm (B) there exists a constant \( C \) such that:

\[
\lim_{i \to \infty} \sup_{n} \frac{|x_{i+n} - x^*|}{|x_i - x^*|^2} \leq C < \infty
\]

where \( x^* \) is the minimizer of the function \( f(x) \).

The convergence results of this section show that Algorithm (B) produces a sequence of points \( \{x_i\} \) which is well defined and converges to the unique minimizer \( x^* \) of a function \( f(x) \). Furthermore, the algorithm possesses the property of stability.

The rate of convergence of the same algorithm is the same as any other conjugate gradient gradient algorithm with restarts. This rate is given by Theorem (4.2.I.3).

Next, some computational results follow and comparison with the standard algorithms will establish the performance of the new model.
4.2.11 Numerical Experiments.

Several standard test functions were minimized to compare the extended conjugate gradient method using the updating formula (4.2.2) with the one using the model proposed in Boland et al. (1979) and the standard algorithm with Fletcher-Reeves (Method (I), Method (II), Method (III), Table 1) and Polak-Ribière updates (Method (Ia), Method (IIa), Method (IIIa), Table 2).

The same line search was employed in each of the three methods used. This was the cubic interpolation described in Fletcher and Reeves (1964).

The algorithm was terminated if the difference between two successive function values was reduced below 0.1E-10 and the norm of the difference between two successive points was reduced below 0.1E-05.

The computational results which are shown in Table (1) were performed in double precision on an interactive PRIME computer, using the same pilot program written in Fortran.

Table (1) gives the number of iterations after which the iterative procedure is reset, Index of Computational Labour (I.C.L.), i.e. number of function evaluations plus n times the number of gradient evaluations, as well as the error flag.

Table (2) gives the index of computational labour and the number of iterations needed for the algorithm to converge.

The error flag is set to zero if the algorithm converges to the optimal value and is set to one if the algorithm fails to converge in the time allowed. No column flag has been included in the second table as all the runs were successful.
<table>
<thead>
<tr>
<th>Problem</th>
<th>Reset</th>
<th>METHOD I</th>
<th>METHOD II</th>
<th>METHOD III</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>I.C.L.</td>
<td>Flag</td>
<td>I.C.L.</td>
</tr>
<tr>
<td>1</td>
<td>N+1</td>
<td>243</td>
<td>0</td>
<td>174</td>
</tr>
<tr>
<td>2</td>
<td>N+1</td>
<td>120</td>
<td>0</td>
<td>123</td>
</tr>
<tr>
<td>3</td>
<td>N+1</td>
<td>204</td>
<td>0</td>
<td>153</td>
</tr>
<tr>
<td>4</td>
<td>N+1</td>
<td>440</td>
<td>0</td>
<td>365</td>
</tr>
<tr>
<td>5</td>
<td>N+1</td>
<td>3360</td>
<td>0</td>
<td>3528</td>
</tr>
<tr>
<td>6</td>
<td>N-1</td>
<td>3402</td>
<td>0</td>
<td>3129</td>
</tr>
<tr>
<td>6</td>
<td>N+1</td>
<td>119595</td>
<td>1</td>
<td>114972</td>
</tr>
<tr>
<td>6</td>
<td>N-1</td>
<td>119795</td>
<td>1</td>
<td>114570</td>
</tr>
<tr>
<td>6</td>
<td>N/4</td>
<td>91254</td>
<td>0</td>
<td>93666</td>
</tr>
<tr>
<td>7</td>
<td>N+1</td>
<td>199698</td>
<td>1</td>
<td>195287</td>
</tr>
<tr>
<td>7</td>
<td>N/4</td>
<td>240199</td>
<td>1</td>
<td>232179</td>
</tr>
<tr>
<td>7</td>
<td>N/10</td>
<td>145563</td>
<td>0</td>
<td>182455</td>
</tr>
<tr>
<td>8</td>
<td>N+1</td>
<td>830</td>
<td>0</td>
<td>510</td>
</tr>
<tr>
<td>9</td>
<td>N+1</td>
<td>29727</td>
<td>0</td>
<td>18468</td>
</tr>
<tr>
<td>9</td>
<td>N-1</td>
<td>26649</td>
<td>0</td>
<td>19602</td>
</tr>
<tr>
<td>10</td>
<td>N/4</td>
<td>46632</td>
<td>0</td>
<td>108339</td>
</tr>
<tr>
<td>10</td>
<td>N+1</td>
<td>100902</td>
<td>0</td>
<td>119796</td>
</tr>
<tr>
<td>11</td>
<td>N+1</td>
<td>1615</td>
<td>0</td>
<td>1995</td>
</tr>
<tr>
<td>11</td>
<td>N-1</td>
<td>1465</td>
<td>0</td>
<td>1430</td>
</tr>
<tr>
<td>12</td>
<td>N+1</td>
<td>2301</td>
<td>0</td>
<td>1352</td>
</tr>
<tr>
<td>12</td>
<td>N-1</td>
<td>2340</td>
<td>0</td>
<td>1768</td>
</tr>
<tr>
<td>12</td>
<td>N/4</td>
<td>2899</td>
<td>0</td>
<td>4589</td>
</tr>
<tr>
<td>13</td>
<td>N+1</td>
<td>3906</td>
<td>0</td>
<td>4011</td>
</tr>
<tr>
<td>13</td>
<td>N/4</td>
<td>7329</td>
<td>0</td>
<td>7455</td>
</tr>
<tr>
<td>14</td>
<td>N+1</td>
<td>36640</td>
<td>0</td>
<td>36207</td>
</tr>
<tr>
<td>14</td>
<td>N/4</td>
<td>19683</td>
<td>0</td>
<td>21627</td>
</tr>
<tr>
<td>14</td>
<td>N-1</td>
<td>35640</td>
<td>0</td>
<td>40662</td>
</tr>
<tr>
<td>15</td>
<td>N-1</td>
<td>104118</td>
<td>0</td>
<td>160599</td>
</tr>
<tr>
<td>15</td>
<td>N/4</td>
<td>82008</td>
<td>0</td>
<td>108942</td>
</tr>
<tr>
<td>15</td>
<td>N+1</td>
<td>104721</td>
<td>0</td>
<td>160800</td>
</tr>
<tr>
<td>Problem</td>
<td>Reset</td>
<td>METHOD (Ia)</td>
<td>METHOD (IIa)</td>
<td>METHOD (IIIa)</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>-------------</td>
<td>--------------</td>
<td>---------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>I.C.L.</td>
<td>IT</td>
<td>I.C.L.</td>
</tr>
<tr>
<td>4</td>
<td>N+1</td>
<td>1010</td>
<td>97</td>
<td>840</td>
</tr>
<tr>
<td>5</td>
<td>N+1</td>
<td>5040</td>
<td>92</td>
<td>6720</td>
</tr>
<tr>
<td>5</td>
<td>N-1</td>
<td>5124</td>
<td>90</td>
<td>5544</td>
</tr>
<tr>
<td>6</td>
<td>N+1</td>
<td>143514</td>
<td>209</td>
<td>137082</td>
</tr>
<tr>
<td>6</td>
<td>N-1</td>
<td>142710</td>
<td>207</td>
<td>137500</td>
</tr>
<tr>
<td>8</td>
<td>N+1</td>
<td>565</td>
<td>51</td>
<td>380</td>
</tr>
<tr>
<td>9</td>
<td>N+1</td>
<td>27761</td>
<td>117</td>
<td>19521</td>
</tr>
<tr>
<td>9</td>
<td>N-1</td>
<td>22113</td>
<td>111</td>
<td>19926</td>
</tr>
<tr>
<td>10</td>
<td>N+1</td>
<td>100500</td>
<td>205</td>
<td>96681</td>
</tr>
<tr>
<td>10</td>
<td>N/4</td>
<td>36180</td>
<td>56</td>
<td>37185</td>
</tr>
<tr>
<td>10</td>
<td>N/10</td>
<td>30753</td>
<td>44</td>
<td>36783</td>
</tr>
<tr>
<td>10</td>
<td>√N=14</td>
<td>36381</td>
<td>57</td>
<td>33366</td>
</tr>
<tr>
<td>10a</td>
<td>N+1</td>
<td>371326</td>
<td>404</td>
<td>238595</td>
</tr>
<tr>
<td>10a</td>
<td>√N=20</td>
<td>73383</td>
<td>46</td>
<td>74987</td>
</tr>
<tr>
<td>10a</td>
<td>√N/2</td>
<td>79799</td>
<td>53</td>
<td>70977</td>
</tr>
<tr>
<td>10a</td>
<td>N/10</td>
<td>95839</td>
<td>67</td>
<td>87418</td>
</tr>
<tr>
<td>11</td>
<td>N+1</td>
<td>665</td>
<td>64</td>
<td>570</td>
</tr>
<tr>
<td>12</td>
<td>N+1</td>
<td>1287</td>
<td>43</td>
<td>1053</td>
</tr>
<tr>
<td>12</td>
<td>N-1</td>
<td>1404</td>
<td>49</td>
<td>1287</td>
</tr>
<tr>
<td>12</td>
<td>N/4</td>
<td>2652</td>
<td>99</td>
<td>1586</td>
</tr>
<tr>
<td>13</td>
<td>N+1</td>
<td>3192</td>
<td>67</td>
<td>1428</td>
</tr>
<tr>
<td>13</td>
<td>N-1</td>
<td>2961</td>
<td>61</td>
<td>1386</td>
</tr>
<tr>
<td>13</td>
<td>N/4</td>
<td>4221</td>
<td>94</td>
<td>3318</td>
</tr>
<tr>
<td>14</td>
<td>N+1</td>
<td>30051</td>
<td>166</td>
<td>16686</td>
</tr>
<tr>
<td>14</td>
<td>N-1</td>
<td>21708</td>
<td>113</td>
<td>16524</td>
</tr>
<tr>
<td>14</td>
<td>N/4</td>
<td>12879</td>
<td>57</td>
<td>10611</td>
</tr>
<tr>
<td>15</td>
<td>N+1</td>
<td>91053</td>
<td>197</td>
<td>92661</td>
</tr>
<tr>
<td>15</td>
<td>N-1</td>
<td>91053</td>
<td>197</td>
<td>91857</td>
</tr>
<tr>
<td>15</td>
<td>N/4</td>
<td>40803</td>
<td>72</td>
<td>52863</td>
</tr>
<tr>
<td>15</td>
<td>√N=20</td>
<td>36381</td>
<td>57</td>
<td>37185</td>
</tr>
<tr>
<td>15a</td>
<td>N/10</td>
<td>102255</td>
<td>86</td>
<td>109874</td>
</tr>
<tr>
<td>15a</td>
<td>√N</td>
<td>66967</td>
<td>42</td>
<td>70576</td>
</tr>
<tr>
<td>15a</td>
<td>√N/2</td>
<td>83007</td>
<td>54</td>
<td>73784</td>
</tr>
</tbody>
</table>
4.2.111 Conclusions.

The relative effectiveness of a conjugate gradient algorithm which incorporated the proposed scheme was tested against the performance of the standard conjugate gradient algorithm and the one proposed in Boland et al. (1979). The numerical work was carried out on several standard test functions of both small and large dimensions. (For details of these test functions, see Appendix).

The comparison of the three algorithms was made according to the following criteria:

(i) Success in obtaining an optimal solution for problems of different dimensionality.

(ii) Number of function and gradient evaluations.

(iii) Computer time to termination.

The numerical results indicate that all three formulae perform about the same on small dimensional problems. As the dimensionality of the problem increases, the proposed formula shows a superior performance over the other two methods.

In several of the high dimensional problems, the Fletcher-Reeves method and the one proposed in Boland et al. (1979) failed to converge to the optimal solution within the time allowed. Different restarts were tried for problems of moderate and large dimensionalities. It appears that for large dimensional problems using the usual restart (every N or (N+1) iterations, where N is the number of variables) gives less satisfactory results than using an earlier restart, (i.e. every N/4 iterations).
Using the proposed model with the Polak-Ribière update was 31% more efficient than it was with the Fletcher-Reeves update. Here the percentage is obtained from the ratio of the sum of I.C.L. of the successful runs from Table 1 to the sum of I.C.L. for the corresponding runs from Table 2.

Similarly using the successful runs of Table 1 only, the new model with Fletcher-Reeves update is 8% better than the standard and 17% better than the model of Boland et al. (1979). Finally, from Table 2 only, the new method with Polak Ribière update is 19% better than the standard model and 27% better than the model of Boland et al. (1979).

The computational work which is presented in this section, was performed using accurate line searches. The effects of the nonlinear scaling in presence of inaccurate line searches is investigated in the next section.
4.3 THE CONJUGATE GRADIENT METHOD WITH INACCURATE LINE SEARCH.

4.3.I Convergence of the New Model - Development of the Algorithm.

So far as the analysis has been taken in Section 4.2, it appears that the function $f(x)$ must be minimized precisely along the direction $d$ at each iteration in order to guarantee convergence. For quadratic functions perfect line searches do not require essentially more computational effort than imperfect line searches, since $a_k$ has to be set equal to:

$$a_k = -\frac{d_k^T s_k}{d_k^T Q d_k} \quad (4.3.I.1)$$

and this is substituted in the equation:

$$x_{k+1} = x_k + a_k d_k .$$

For the method of conjugate gradients, Dixon (1975) has shown how the minimum of a quadratic function can be obtained in at most $n$ iterations without explicit knowledge of the matrix $Q$, which would be required if (4.2.2) is to be applied.

Perry (1978) suggested a conjugate gradient method which satisfies the perfect line search conditions even for non-quadratic functions with inaccurate line searches. (The new equation is a linear combination of two orthogonal vectors one of which is the negative gradient. Perry's procedure is reduced to the Polak-Ribiière method whenever the line search is perfectly accurate).

Since exact minimization along the direction $d$ is computationally impossible, it is important to know that convergence will still be maintained under some approximations. A reasonable approximation
would be to stop the linear minimization whenever

\[ |g^T_{i+1} d_i| < \varepsilon |g^T_i d_i| \]

or

\[ |g^T_{i+1} d_i| < \varepsilon \|g_i\|^2 \]

(4.3.1.2)

where \( \varepsilon \) is a positive number less than one. These stopping criteria were proposed by Nazareth (1977) where his algorithm was tested for values of \( \varepsilon \) between 0.1 and 0.95. Nazareth's algorithm maintained finite termination when applied to quadratic functions and required \( O(n) \) storage, where \( n \) was the dimensionality of the problem.

Some convergence properties of the conjugate gradient Algorithm (B) with the stopping criteria (4.3.1.2) for the line search are now given.

**Theorem (4.3.1.1):**

Let \( f \) be a real valued vector function of class \( C^2 \) on an open convex set \( D \) containing \( S(x_o) \). Assume that \( S(x_o) \) is bounded and the spectrum of the Hessian matrix \( G(x) \) is bounded below and above by \( \mu > 0 \) and \( \lambda \) respectively for all \( x \in S(x_o) \). Then Algorithm (B) with the modification in the line search given by (4.3.1.2) is stable.

**Proof:**

Let \( x^*_i = x_i + \alpha^*_i d_i \) be the minimum point along the direction \( d_i \) and \( x_{i+1} = x_i + \alpha_i d_i \). Then

\[ g^T_{i+1} d_i = 0 \]  

(4.3.1.3)
and by the mean-value theorem:

\[ f(x_{i+1}) - f(x_{i+1}^*) = \frac{1}{2} \left( \alpha_{i+1} - \alpha_i^* \right)^2 d_i^T G(z) d_i \leq \frac{1}{2} \lambda (\alpha_{i+1} - \alpha_i^*)^2 \|d_i\|^2 \]

(4.3.I.4)

where \( z \in (x_{i+1}, x_{i+1}^*) \). Similarly follows the inequality:

\[ f(x_i) - f(x_i^*) \geq \frac{1}{2} \mu \alpha_i^2 \|d_i\|^2 \]

(4.3.I.5)

But

\[ \alpha_i^* \geq \frac{1}{\lambda} \frac{g_i^T d_i}{\|d_i\|^2} \]  

(4.3.I.6)

(Proof in the Appendix, Lemma A1), therefore:

\[ f(x_i) - f(x_i^*) \geq \frac{1}{2} \frac{\mu}{\lambda} \left( -g_i^T d_i \right)^2 \|d_i\|^2 \]

(4.3.I.7)

Now

\[ g_{i+1}^* = g_{i+1} + (\alpha_i^* - \alpha_i) \tilde{c}(w) \quad \text{for some } w \in (x_{i+1}, x_{i+1}^*) \]

Then

\[ |\alpha_i^* - \alpha_i| = \frac{|-g_{i+1}^T d_i|}{\|d_i\|^2 G(w) d_i} \leq \frac{\varepsilon}{\mu \|d_i\|^2} \|g_{i+1}^*\| \|d_i\|} \]

(4.3.I.8)

By assumption:

\[ |\alpha_i^* - \alpha_i| \leq \frac{\varepsilon^2 |g_i^T d_i|}{\mu \|d_i\|^2} \]

or

\[ |\alpha_i^* - \alpha_i| \|d_i\|^2 \leq \frac{\varepsilon^2}{\mu} \frac{|g_i^T d_i|^2}{\|d_i\|^2} \]

(4.3.I.9)

Substituting (4.3.I.9) into (4.3.I.7):
If \( \varepsilon \leq \frac{1}{\lambda} (\lambda)^{-\frac{1}{2}} \) then from (4.3.1.4) and (4.3.1.10):

\[
\frac{\theta_i}{\theta_{i+1}} = \frac{\nabla f(x_i) \cdot d_i}{\nabla f(x_{i+1}) \cdot d_i} > 1
\]

or

\[
f(x_i) > f(x_{i+1})
\]

The first inequality of (4.3.1.2) can be considered as a measure of the error in the solution of the one-dimensional problem of the i-th iteration. If

\[
\theta_i = \frac{T_{i+1} d_i}{T_i d_i}
\]

then (4.3.1.11) represents the ratio of the derivative of the function in the direction \( d_i \) at the approximate solution \( x_{i+1} \) to the derivative in the direction \( d_i \) at the starting point \( x_i \). In a similar way as in Lenard (1976) it can be shown that Algorithm (B) with inaccurate line searches exhibits n-step rational convergence when Fletcher-Reeves or Polak-Ribière's updates are used.

4.3.II Numerical Results.

Different standard test functions were tried in various dimensions in order to examine the effectiveness of the three models. The first model (ST) corresponds to the standard method proposed by Fletcher and Reeves (1964), the second and the third models ((PO) and (RA)) to the ones proposed in Boland et al. (1979) and Tassopoulos and Storey (to appear) respectively.
The performance of the three models was tested on a variety of updating formulae for the search directions and line search accuracy parameters. The tables included at the end of this work contain the results of the numerical trials.

The line search routine used was a cubic interpolation using function values and gradients. The programme used was an adapted routine from the NAG library. This routine proceeds as follows. From a starting point \( x_k \) (where \( k \) is the iteration number) a search is conducted along a direction \( d_k \) for a "better" point \( x_{k+1} \). This point will have the form:

\[
x_{k+1} = x_k + \lambda d_k
\]

for some scalar \( \lambda \). Then the function used in our line search will be a function of \( \lambda \), \( f(x_k + \lambda d_k) \) with derivative

\[
\frac{d}{d\lambda} (f(x_k + \lambda d_k)) = g^T(x_k + \lambda d_k)d_k.
\]

If an exact line search is performed then \( \lambda \) is required to minimize \( f(x_k + \lambda d_k) \). If \( \lambda^* \) is the minimizing value of \( \lambda \) then:

\[
\frac{d}{d\lambda} f(x_k + \lambda^* d_k) = g^T(x_k + \lambda^* d_k)d_k = 0
\]

i.e. if the line searches were exact the directional derivative

\[
g^T(x_k + \lambda^* d_k)d_k
\]

at the new point \( x_{k+1} = x_k + \lambda^* d_k \) would be zero.

However, in the relaxed line search used the scalar \( \lambda \) is selected in such a way that at the new point \( x_{k+1} = x_k + \lambda^* d_k \) the
directional derivative has been reduced by a certain amount but not necessarily to zero, i.e. \( \lambda^* \) is chosen to satisfy the relation:

\[
\left| \frac{df}{d\lambda} (x_k + \lambda d_k) \right|_{\lambda=\lambda^*} < P \left| \frac{df}{d\lambda} (x_k + \lambda d_k) \right|_{\lambda=0}
\]  \hspace{1cm} (4.3.III.1)

where \( P \) is a specified scalar in the interval \((0,1)\). The above relation can also be expressed as

\[
\left| g_k^T d_k \right| < P \left| g_k^T d_k \right| .
\]  \hspace{1cm} (4.3.III.2)

The algorithms were tested for the following five values of \( P \): 0.1, 0.3, 0.5, 0.7 and 0.9. To ensure that progress had been made towards \( x^* \) (i.e. the minimizer of \( f(x) \)) an additional condition was imposed. This is that \( \lambda = \lambda^* \) is chosen to satisfy the condition:

\[
f(x_k + \lambda^* d_k) < f(x_k) .
\]  \hspace{1cm} (4.3.III.3)

Therefore, to sum up, \( f(x_k + \lambda^* d_k) \) is a function of \( \lambda \) and the value \( \lambda = \lambda^* \) is accepted if conditions (4.3.III.1) and (4.3.III.3) hold. Finally, the line search is terminated if either of the following conditions holds:

(i) The Function Subroutine has been called nine times. The number nine has been chosen from computational experience.

(ii) The step length does not increase by more than \( 0.5 \times 10^{-12} \) at any particular stage in the line search.

The initial step length at an iterate \( x_k \) is taken to be the one suggested by Fletcher and Reeves (1964), i.e.
\[ \lambda = 2(F(x_k) - F(x_{k-1}))/g_k^T d_k. \]

For the initial step length from the starting point \( x_1 \) a value was chosen for each individual function usually to gain the largest initial decrease in the function value.

The convergence criteria used were a test on the gradient norm and a test on the progress which had been made from the iterate \( x_k \) to the iterate \( x_{k+1} \), i.e.

\[ \|g_k\|^2_E < 5.0 \times 10^{-12} \quad (4.3.II.4) \]

\[ \|x_k - x_{k+1}\|^E \leq 1.0 \times 10^{-6} \quad (4.3.II.5) \]

The criterion (4.3.II.4) was initially used alone but the combination of the two criteria (4.3.II.4) and (4.3.II.5) proved to be more satisfactory for all of the functions used.

4.3.III Conclusions.

The first three tables that follow contain the numerical results for the three models used in the same algorithm. The test functions used are given in the Appendix.

For each value of the line search parameter \( P \) and each different updating procedure the number of times that the function subroutine was called is given. When the test failed to converge to the right point an "F" is marked in the appropriate place. Failures were caused because of premature termination which can be altered by allowing the function subroutine to be called more than nine times.
(see for example Table 3), or by adding the error-vector term every \( n^* \) iterations, where \( n^* \leq n-2 \) (preferably \( \sqrt{n} \)), instead of adding it every \( n \) iterations.

For each individual function the initial step from the starting point is listed at the head of the table. The fourth table includes the numerical results of the standard method when the error-vector term is not added every \( n \) iterations but every \( n^* \) iterations where \( n^* \leq n-2 \). The different values taken for \( n^* \) showed that for functions of average to large dimensionality adding the error-vector at every \( n^* = (n/2, \sqrt{n}, n-2) \) iterations is beneficial and improves the efficiency of the method.

The optimum value of \( n^* \) depends rather on the dimensionality and on the function itself. The numerical results of the fourth table showed however, that the value \( n^* = \sqrt{n} \) gave the best results for the functions tested. From the same table it is clear that as the line search becomes more accurate (i.e. \( P = 0.1 \) or 0.3) a higher amount of computational labour is required for the algorithm to converge.

The last table contains the percentage improvement, measured in terms of function subroutine calls, of the model proposed in Tassopoulos and Storey (to appear in JOTA) for each individual updating procedure. The percentages were taken as the ratio of the sums of the successful runs of one model to the sum of the corresponding successful runs of the other models. A careful examination of the Tables 1, 2 and 3 shows that the first update gave better results for functions (1) and (3) and the second update was more efficient for function (2) for all the models used. As the dimensionality of the problem
increases the two models (RA) and (PO) behave more satisfactorily than the standard model. Finally, it can be seen from the last table, that using the proposed model there was a considerable improvement in almost all the updating procedures used.

Further theoretical work on the model proposed in this work and its relationships with the conjugate gradient algorithms are currently being carried out.
<table>
<thead>
<tr>
<th>P</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ST</td>
<td>RA</td>
<td>PO</td>
<td>ST</td>
</tr>
<tr>
<td>2</td>
<td>112</td>
<td>100</td>
<td>141</td>
<td>112</td>
</tr>
<tr>
<td>4</td>
<td>238</td>
<td>256</td>
<td>328</td>
<td>342</td>
</tr>
<tr>
<td>0.1</td>
<td>8</td>
<td>194</td>
<td>200</td>
<td>117</td>
</tr>
<tr>
<td>16</td>
<td>234</td>
<td>141</td>
<td>123</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>398</td>
<td>176</td>
<td>123</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>72</td>
<td>162</td>
<td>75</td>
</tr>
<tr>
<td>4</td>
<td>254</td>
<td>236</td>
<td>275</td>
<td>270</td>
</tr>
<tr>
<td>0.3</td>
<td>8</td>
<td>116</td>
<td>116</td>
<td>219</td>
</tr>
<tr>
<td>16</td>
<td>270</td>
<td>323</td>
<td>217</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>398</td>
<td>176</td>
<td>123</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td>55</td>
<td>64</td>
<td>119</td>
<td>55</td>
</tr>
<tr>
<td>4</td>
<td>247</td>
<td>211</td>
<td>211</td>
<td>283</td>
</tr>
<tr>
<td>0.5</td>
<td>8</td>
<td>116</td>
<td>116</td>
<td>229</td>
</tr>
<tr>
<td>16</td>
<td>209</td>
<td>252</td>
<td>260</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>261</td>
<td>253</td>
<td>288</td>
<td>404</td>
</tr>
<tr>
<td>2</td>
<td>46</td>
<td>54</td>
<td>125</td>
<td>46</td>
</tr>
<tr>
<td>4</td>
<td>228</td>
<td>183</td>
<td>205</td>
<td>228</td>
</tr>
<tr>
<td>0.7</td>
<td>8</td>
<td>116</td>
<td>116</td>
<td>229</td>
</tr>
<tr>
<td>16</td>
<td>209</td>
<td>161</td>
<td>307</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>292</td>
<td>146</td>
<td>293</td>
<td>F</td>
</tr>
<tr>
<td>2</td>
<td>46</td>
<td>54</td>
<td>94</td>
<td>46</td>
</tr>
<tr>
<td>4</td>
<td>228</td>
<td>183</td>
<td>205</td>
<td>228</td>
</tr>
<tr>
<td>0.9</td>
<td>8</td>
<td>116</td>
<td>116</td>
<td>229</td>
</tr>
<tr>
<td>16</td>
<td>209</td>
<td>161</td>
<td>233</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>292</td>
<td>171</td>
<td>134</td>
<td>F</td>
</tr>
<tr>
<td></td>
<td>I</td>
<td>II</td>
<td>III</td>
<td>IV</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td>ST</td>
<td>RA</td>
<td>PO</td>
<td>ST</td>
</tr>
<tr>
<td>0.1</td>
<td>4</td>
<td>342</td>
<td>308</td>
<td>172</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>126</td>
<td>112</td>
<td>112</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>113</td>
<td>112</td>
<td>112</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>113</td>
<td>112</td>
<td>112</td>
</tr>
<tr>
<td>0.3</td>
<td>4</td>
<td>297</td>
<td>263</td>
<td>291</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>271</td>
<td>158</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>153</td>
<td>153</td>
<td>173</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>153</td>
<td>153</td>
<td>153</td>
</tr>
<tr>
<td>0.5</td>
<td>4</td>
<td>221</td>
<td>158</td>
<td>158</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>323</td>
<td>188</td>
<td>237</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>252</td>
<td>210</td>
<td>216</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>330</td>
<td>211</td>
<td>283</td>
</tr>
<tr>
<td>0.7</td>
<td>4</td>
<td>252</td>
<td>225</td>
<td>253</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>182</td>
<td>273</td>
<td>178</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>315</td>
<td>273</td>
<td>273</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>286</td>
<td>314</td>
<td>224</td>
</tr>
<tr>
<td>0.9</td>
<td>4</td>
<td>216</td>
<td>162</td>
<td>207</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>223</td>
<td>242</td>
<td>245</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>294</td>
<td>276</td>
<td>282</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>328</td>
<td>282</td>
<td>260</td>
</tr>
</tbody>
</table>
Table (3)

Wood

(The numbers in brackets are the results of allowing the function subroutine to be called more than nine times.)

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th></th>
<th>II</th>
<th></th>
<th>III</th>
<th></th>
<th>IV</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ST</td>
<td>RA</td>
<td>PO</td>
<td>ST</td>
<td>RA</td>
<td>PO</td>
<td>ST</td>
<td>RA</td>
</tr>
<tr>
<td>4</td>
<td>749</td>
<td>1031</td>
<td>1033</td>
<td>764</td>
<td>781</td>
<td>1053</td>
<td>774</td>
<td>781</td>
</tr>
<tr>
<td>8</td>
<td>1660</td>
<td>1634</td>
<td>1924</td>
<td>F</td>
<td>1681</td>
<td>2287</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>0.1</td>
<td>16</td>
<td>F</td>
<td>1986</td>
<td>1913</td>
<td>F</td>
<td>(3532)</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>20</td>
<td>F (3354)</td>
<td>1448</td>
<td>F</td>
<td>2033</td>
<td>F</td>
<td>2125</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>40</td>
<td>F</td>
<td>2619</td>
<td>1994</td>
<td>2304</td>
<td>F</td>
<td>F</td>
<td>2444</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>122</td>
<td>64</td>
<td>96</td>
<td>101</td>
<td>109</td>
<td>101</td>
<td>131</td>
<td>144</td>
</tr>
<tr>
<td>8</td>
<td>226</td>
<td>193</td>
<td>462</td>
<td>398</td>
<td>317</td>
<td>155</td>
<td>333</td>
<td>329</td>
</tr>
<tr>
<td>0.3</td>
<td>16</td>
<td>1124</td>
<td>413</td>
<td>418</td>
<td>F</td>
<td>F</td>
<td>359</td>
<td>301</td>
</tr>
<tr>
<td>20</td>
<td>221</td>
<td>381</td>
<td>511</td>
<td>F</td>
<td>F</td>
<td>391</td>
<td>F</td>
<td>303</td>
</tr>
<tr>
<td>40</td>
<td>221</td>
<td>480</td>
<td>383</td>
<td>F</td>
<td>F</td>
<td>380</td>
<td>F</td>
<td>366</td>
</tr>
<tr>
<td>4</td>
<td>115</td>
<td>59</td>
<td>98</td>
<td>60</td>
<td>135</td>
<td>80</td>
<td>97</td>
<td>92</td>
</tr>
<tr>
<td>8</td>
<td>138</td>
<td>138</td>
<td>646</td>
<td>286</td>
<td>305</td>
<td>372</td>
<td>405</td>
<td>215</td>
</tr>
<tr>
<td>0.5</td>
<td>16</td>
<td>671(1393)</td>
<td>262</td>
<td>159</td>
<td>F</td>
<td>374</td>
<td>297</td>
<td>320(640)</td>
</tr>
<tr>
<td>20</td>
<td>249(993)</td>
<td>336</td>
<td>159</td>
<td>F</td>
<td>400</td>
<td>456</td>
<td>549(806)</td>
<td>455</td>
</tr>
<tr>
<td>40</td>
<td>249</td>
<td>336</td>
<td>159</td>
<td>F</td>
<td>548</td>
<td>492</td>
<td>459</td>
<td>287</td>
</tr>
<tr>
<td>4</td>
<td>63</td>
<td>62</td>
<td>108</td>
<td>68</td>
<td>77</td>
<td>89</td>
<td>105</td>
<td>69</td>
</tr>
<tr>
<td>8</td>
<td>320</td>
<td>305</td>
<td>295</td>
<td>284</td>
<td>191</td>
<td>164</td>
<td>241</td>
<td>214</td>
</tr>
<tr>
<td>0.7</td>
<td>16</td>
<td>585(1711)</td>
<td>822</td>
<td>374</td>
<td>F</td>
<td>F</td>
<td>649</td>
<td>471(766)</td>
</tr>
<tr>
<td>20</td>
<td>873(1446)</td>
<td>1136</td>
<td>513</td>
<td>F</td>
<td>F</td>
<td>538</td>
<td>339(808)</td>
<td>487</td>
</tr>
<tr>
<td>40</td>
<td>F</td>
<td>767</td>
<td>F</td>
<td>F</td>
<td>630</td>
<td>523</td>
<td>320</td>
<td>442</td>
</tr>
<tr>
<td>4</td>
<td>61</td>
<td>82</td>
<td>86</td>
<td>66</td>
<td>75</td>
<td>77</td>
<td>90</td>
<td>65</td>
</tr>
<tr>
<td>8</td>
<td>979</td>
<td>490</td>
<td>196</td>
<td>476</td>
<td>394</td>
<td>164</td>
<td>505</td>
<td>166</td>
</tr>
<tr>
<td>0.9</td>
<td>16</td>
<td>610(1444)</td>
<td>1251</td>
<td>624</td>
<td>F</td>
<td>(1237)</td>
<td>933</td>
<td>541</td>
</tr>
<tr>
<td>20</td>
<td>F</td>
<td>669</td>
<td>F</td>
<td>(1256)</td>
<td>402</td>
<td>316</td>
<td>346(1320)</td>
<td>425</td>
</tr>
<tr>
<td>40</td>
<td>F</td>
<td>564</td>
<td>F</td>
<td>F</td>
<td>462</td>
<td>479</td>
<td>488</td>
<td>448</td>
</tr>
</tbody>
</table>

124
TABLE (4)

WOOD'S FUNCTION.

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>√N</td>
<td>N-2</td>
<td>N/2</td>
<td>√N</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>1818</td>
</tr>
<tr>
<td>16</td>
<td>1910</td>
<td>F</td>
<td>1595</td>
<td>1868</td>
</tr>
<tr>
<td>20</td>
<td>1715</td>
<td>F</td>
<td>F</td>
<td>1970</td>
</tr>
<tr>
<td>40</td>
<td>2292</td>
<td>F</td>
<td>F</td>
<td>1945</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>314</td>
</tr>
<tr>
<td>16</td>
<td>325</td>
<td>221</td>
<td>316</td>
<td>599</td>
</tr>
<tr>
<td>20</td>
<td>390</td>
<td>221</td>
<td>221</td>
<td>518</td>
</tr>
<tr>
<td>40</td>
<td>507</td>
<td>221</td>
<td>221</td>
<td>523</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>251</td>
</tr>
<tr>
<td>16</td>
<td>263</td>
<td>671</td>
<td>138</td>
<td>149</td>
</tr>
<tr>
<td>20</td>
<td>252</td>
<td>249</td>
<td>249</td>
<td>149</td>
</tr>
<tr>
<td>40</td>
<td>217</td>
<td>249</td>
<td>249</td>
<td>158</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>181</td>
</tr>
<tr>
<td>16</td>
<td>118</td>
<td>585</td>
<td>526</td>
<td>148</td>
</tr>
<tr>
<td>20</td>
<td>118</td>
<td>873</td>
<td>873</td>
<td>148</td>
</tr>
<tr>
<td>40</td>
<td>118</td>
<td>F</td>
<td>F</td>
<td>157</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td>170</td>
</tr>
<tr>
<td>16</td>
<td>226</td>
<td>F</td>
<td>F</td>
<td>278</td>
</tr>
<tr>
<td>20</td>
<td>232</td>
<td>F</td>
<td>F</td>
<td>275</td>
</tr>
<tr>
<td>40</td>
<td>261</td>
<td>F</td>
<td>F</td>
<td>280</td>
</tr>
</tbody>
</table>
TABLE (5)

Percentages of improvement of the standard method adding the error-vector term every $\sqrt{N}$ iterations over the other two restarts (N-2, N/2).

<table>
<thead>
<tr>
<th></th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-2</td>
<td>33.4</td>
<td>24.2</td>
<td>27.4</td>
<td>1.6</td>
</tr>
<tr>
<td>N/2</td>
<td>21.6</td>
<td>24.2</td>
<td>17.9</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Average percentages of improvement:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>N-2</td>
<td>21.6</td>
</tr>
<tr>
<td>N/2</td>
<td>16.2</td>
</tr>
</tbody>
</table>

Percentages of improvement of model (4.2.2) over other models.

<table>
<thead>
<tr>
<th></th>
<th>I (H/S)</th>
<th>II (P/R)</th>
<th>III (F/R)</th>
<th>IV (DIXON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST</td>
<td>ROS</td>
<td>13.5</td>
<td>2.5</td>
<td>6.6</td>
</tr>
<tr>
<td></td>
<td>POW</td>
<td>12.6</td>
<td>12.0</td>
<td>19.0</td>
</tr>
<tr>
<td></td>
<td>WOOD</td>
<td>-2.5</td>
<td>4.1</td>
<td>23.1</td>
</tr>
<tr>
<td>POL</td>
<td>ROS</td>
<td>19.1</td>
<td>29.2</td>
<td>7.7</td>
</tr>
<tr>
<td></td>
<td>POW</td>
<td>2.8</td>
<td>4.4</td>
<td>-3.5</td>
</tr>
<tr>
<td></td>
<td>WOOD</td>
<td>-6.3</td>
<td>15.8</td>
<td>15.4</td>
</tr>
</tbody>
</table>

Average percentages:

<table>
<thead>
<tr>
<th></th>
<th>ROS</th>
<th>POW</th>
<th>WOOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST</td>
<td>6.95</td>
<td>13.7</td>
<td>6.6</td>
</tr>
<tr>
<td>POL</td>
<td>16.5</td>
<td>-1.3</td>
<td>6.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ST</th>
<th>ROS</th>
<th>9.0 %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>POW</td>
<td>9.0 %</td>
</tr>
<tr>
<td></td>
<td>WOOD</td>
<td>9.0 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>POL</th>
<th>ROS</th>
<th>7.1 %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>POW</td>
<td>7.1 %</td>
</tr>
<tr>
<td></td>
<td>WOOD</td>
<td>7.1 %</td>
</tr>
</tbody>
</table>
APPENDIX

START

SET INITIAL DIRECTIONS

\[ D(I) = - G(I) \]

FIND NEXT DIRECTION AND POINT

\[ D(I+1) = - \hat{G}(I+1) + \hat{B}(I) D(I) \]
\[ X(I+1) = X(I) + \lambda(I) D(I) \text{ (SEE ALGORITHM (II))} \]

ARE THE STOPPING CRITERIA SATISFIED

STOP

N

(AT THIS POINT THE MODIFICATION OF DIXON'S ALGORITHM IS INTRODUCED)

\[ (I+1) \neq N* \]

Y

ADD ACCUMULATED ERROR VECTOR
The test functions used are the following:

**Test function 1**
ROS(2): Rosenbrock,

\[ f(x) = 100(x_2 - x_1^2)^2 + (1-x_1)^2 \]

starting approximation \((x_1, x_2) \equiv (-1.2, 1.0)\)

**Test function 2**
ROS(2): White and HoIst,

\[ f(x) = 100(x_2 - x_1^3)^2 + (1-x_1)^2 \]

starting approximation \((x_1, x_2) \equiv (-1.2, 1.0)\)

**Test function 3**
ROS(2)

\[ f(x) = 100(x_2 - x_1^2)^3 + (1-x_1)^8 \]

starting approximation \((x_1, x_2) \equiv (-1.2, 1.0)\)

**Test function 4**
WOOD(4): C.F. Wood,

\[ f(x) = 100(x_2 - x_1^2)^2 + (1-x_1)^2 + 90(x_4 - x_3^2)^2 + (1-x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1) \]

starting approximation \((x_1, x_2, x_3, x_4) \equiv (-3, -1, -3, -1)\)
Test function 5

WOOD(20)

\[ f(x) = \frac{N}{4} \sum_{i=1}^{100} \left[ (x_{4i-2} - x_{4i-3})^2 \right] + (1-x_{4i-3})^2 + 90(x_{4i} - x_{4i-1})^2 + \\
+ (1-x_{4i-1})^2 + 10.1\left( (x_{4i-2} - 1)^2 + (x_{4i} - 1)^2 \right) + 19.8(x_{4i-2} - 1)(x_{4i} - 1), \quad N=20 \]

starting approximation: \((x_1, \ldots, x_{20}) \equiv (-3, \ldots, -1)\)

Test function 6

WOOD(200)

Test function 7

WOOD(400)

Test function 8

POWELL(4): Powell,

\[ f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4 \]

starting approximation: \((x_1, x_2, x_3, x_4) \equiv (3, -1, 0, 1)\)

Test function 9

POWELL(80)

\[ f(x) = \frac{N}{4} \sum_{i=1}^{100} \left[ (x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + \\
+ (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \right], \quad N=80 \]

starting approximation = \((x_1, x_2, \ldots, x_{80}) \equiv (3, -1, \ldots, 1)\)

Test function 10

POWELL(200)

Test function 10a

POWELL(400)
**Test function 11**

MIE(4): attributed to Miele by Cornwell,

\[ f(x) = (\exp(x_1) - x_2)^2 + 100(x_2 - x_3)^6 + \tan^4(x_3 - x_4) + x_1^8 + (x_4 - 1)^2 \]

starting approximation: \((x_1, x_2, x_3, x_4) \approx (1, 2, 2, 2)\)

**Test function 12**

MIE(12)

\[ f(x) = \frac{N/4}{i=1} \left[ \exp(x_{4i-3}) - x_{4i-2} \right]^2 + 100(x_{4i-2} - x_{4i-1})^6 + \]
\[ + \left[ \tan(x_{4i-1} - x_{4i}) \right]^4 + x_{4i-3}^8 + (x_{4i} - 1)^2, \quad N=12 \]

starting approximation: \((x_1, x_2, \ldots, x_{12}) \approx (1, 2, \ldots, 2)\)

**Test function 13**

MIE(20)

**Test function 14**

MIE(80)

**Test function 15**

MIE(200)

**Test function 15a**

MIE(400)
Proof of Lemma (4.2.I.1): Since $f$ is assumed to be convex for any $x, y \in S(x_0)$ and $0 \leq t \leq 1:$

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y).$$

But $f(x) \leq f(x_0)$ and $f(y) \leq f(x_0),$ then

$$f\left[tx + (1-t)y\right] \leq tf(x_0) + (1-t)f(x_0) = f(x_0)$$

which implies $(tx + (1-t)y) \in S(x_0).$

Hence $S(x_0)$ is convex.

Proof of Lemma (4.2.I.2): The closure of the set is a direct consequence of the continuity of the function $f(x).$ Therefore, to show that $S(x_0)$ is compact it has to be proved that it is bounded. Let $d$ be any direction through $x_0$ such that:

$||d|| = 1.$

The function of one variable is considered:

$$\phi(a) = f(x_0 + ad). \quad (A.1)$$

Then $\phi(a)$ is convex since

$$\phi''(a) = p^T G(x_0 + ap)p \succeq \mu > 0.$$

Next another function of one variable is considered:

$$r(a) = \phi(a) - \phi(0) - ap^T g(x_0) - \frac{1}{2} a^2 \mu. \quad (A.2)$$

Then it can be shown that $r(a)$ is also convex.

Furthermore $r(0) = r'(0) = 0.$
Hence, from (A.2):

\[ \phi(a) \geq \phi(0) + a^T g(x_0) + \frac{1}{2} a^2 \mu \]

and \( \phi(a) > \phi(0) = f(x_o) \) if \( |a| > 2 p^T g_o \). (A.3)

By Schwartz's inequality:

\[ |p^T g_o| \leq \|p\| \|g_o\| \]  (A.4)

Hence, for any \( x \in \mathbb{R}^n \) (A.3) and (A.4) imply that:

\[ f(x) > f(x_o) \) if \( \|x - x_o\| > \frac{2}{\mu} \|g_o\| \]

Therefore the set of points \( x \) satisfying the condition

\[ f(x) \leq f(x_o) \]

is bounded.

**Proof of Lemma (4.2.I.6):** By the mean value theorem, there exists a \( z_i \in (0, a_i) \) such that:

\[ g_{i+1} = g_i + G(x_i + z_i d_i)(x_{i+1} - x_i) = g_i + a_i G(x_i + d_i z_i)d_i \]

Hence

\[ \|g_{i+1}\| \leq \|g_{i+1} - g_i\| + \|g_i\| \leq a_i \|G(x_i + d_i z_i)d_i\| + \|g_i\| \]

\[ \leq a_i \lambda \|d_i\| + \|g_i\| \]

But \( \|d_i\| = \|g_i + \beta_{i-1} \rho_{i-1} d_{i-1}\| \geq \|g_i\| \)
and by Lemma (4.2.1.4):

\[ a_i \leq \frac{1}{\mu} \frac{\|g_i\|^2}{\|d_i\|^2} \leq \frac{1}{\mu} \frac{\|g_i\|}{\|d_i\|} \]

Therefore

\[ \|g_{i+1}\| \leq \left(1 + \frac{\lambda}{\mu}\right) \|g_i\| \]

The previous inequality can be rewritten as:

\[ \frac{\|g_{i+1}\|}{\|g_i\|} \leq \left(1 + \frac{\lambda}{\mu}\right) \]

and hence

\[ \beta_i \leq \left(1 + \frac{\lambda}{\mu}\right)^2 \]

From the definition of the directions \(d_i\):

\[ d_{i+1} = -g_{i+1} + \rho_i \beta_i d_i \]

and so

\[ \|d_{i+1}\|^2 = \|g_{i+1}\|^2 + \rho_i^2 \beta_i^2 \|d_i\|^2 \leq \|g_{i+1}\|^2 + \left(1 + \frac{\lambda}{\mu}\right)^4 \rho_i^2 \|d_i\|^2 \leq \left(1 + \frac{\lambda}{\mu}\right)^2 \|g_i\|^2 + \left(1 + \frac{\lambda}{\mu}\right)^4 \|d_i\|^2 \]

Therefore

\[ \|d_{i+1}\|^2 \leq \left(1 + \frac{\lambda}{\mu}\right)^2 \|d_i\|^2 + \left(1 + \frac{\lambda}{\mu}\right)^4 \|d_i\|^2 \]

and so

\[ \|d_{i+1}\|^2 \leq \left(\left(1 + \frac{\lambda}{\mu}\right)^2 + \left(1 + \frac{\lambda}{\mu}\right)^4\right) \|d_i\|^2 \]
Proof of Lemma (4.2.I.7): Let $\delta = x_i - x_j$ and $\gamma = g_i - g_j$.

Let $t$ be a scalar such that $0 \leq t \leq 1$.

Then

$$\frac{d}{dt} g(x_j + t\delta) = G(x_j + t\delta) \delta$$

therefore

$$\gamma = \int_0^1 G(x_j + t\delta) \delta \, dt.$$ 

Hence

$$||\gamma|| \leq \int_0^1 ||G(x_j + t\delta)|| \, dt \leq \int_0^1 ||G(x_j + t\delta)|| \, dt$$

and so

$$||\gamma|| \leq \max_{0 \leq t \leq 1} ||G(x_j + t\delta)|| = \lambda ||\delta||.$$

The lower bound is obtained by integrating the identity:

$$d\frac{dt}{(g(x_j + t\delta))^T \delta} = \delta^T G(x_j + t\delta) \delta$$

and obtain:

$$\int_0^1 \delta^T G(x_j + t\delta) \delta \, dt = \delta^T \gamma \gamma^T \delta \geq \mu ||\delta||^2.$$

Hence the last inequality and the Schwartz's inequality give:

$$\frac{1}{\mu} \geq \frac{||\delta||}{\gamma^T \delta} \geq \frac{||\delta||}{||\gamma||} \square.$$

Proof of the inequality (4.2.I.14): From the mean value theorem:

$$f(x_i + \alpha d_i) = f(x_i) + \alpha g_i d_i + \frac{1}{2} \alpha^2 d_i^T G(z) d_i$$

where $z \in (x_i, x_i + \alpha d_i)$. 
Hence by Lemma (4.2.1.5):

\[ f(x_i) - f(x_i + \alpha d_i) \geq \alpha \|g_i\|^2 - \frac{1}{2} \alpha^2 \lambda \|d_i\|^2 \geq \]

\[ \frac{1}{2} \alpha^2 \lambda (q+1) (1 + \frac{\lambda}{\mu})^q \|g_i\|^2. \]

Let \( \delta_i = \frac{1}{\lambda (q+1) (1 + \frac{\lambda}{\mu})^q} \)

and the previous inequality gives:

\[ f(x_i) - f(x_i + \alpha d_i) \geq f(x_i) - f(x_i + \delta_i d_i) \geq \frac{\delta_i}{2} \|g_i\|^2 = \]

\[ -\frac{g_i^T d_i \delta_i}{2}. \quad \square \]
CHAPTER V

VARIABLE METRIC METHODS BASED ON A

NON-QUADRATIC MODEL
5.1 INTRODUCTION.

This chapter is concerned with another sophisticated gradient method, the Variable Metric Method, used for locating a local unconstrained minimizer $x^*$ of a continuously differentiable function $f(x)$. The first Variable Metric Method was developed by Davidon in 1959. Davidon's method was not widely publicized, but it constituted a considerable advance over the current alternatives. A modified version of Davidon's method was published by Fletcher and Powell in 1963. This method is known as the Davidon-Fletcher-Powell, (DFP) method. A one-parameter class of updates was proposed by Broyden (1970). When the parameter $\omega$ in Broyden's class equals 1 the BFGS method results and the DFP with choice $\omega = 0$ (see Algorithm (I)) BFGS has replaced the DFP as the most successful V.M.M.

The BFGS method generates a sequence of points $\{x_k\}$, $k = 1, 2, \ldots$ as approximations to a local minimizer of a differentiable function $f(x), \mathbb{R}^n \to \mathbb{R}$, according to the following algorithm.

Algorithm (I): If an initial vector $x_0 \in \mathbb{R}^n$ and an initial $n \times n$ matrix $H_0 = I$ (or any symmetric positive definite matrix) are given then:

For $k = 0, 1, 2, \ldots$

If $\|g_k\| = \|\nabla f(x_k)\| < \varepsilon$, (where $\varepsilon$ is a prescribed small positive number) then stop.

Else, set $d_k = -H_k g_k$

Find $a_k > 0$ which minimizes $f(x_k + ad_k)$ with respect to $a$

* Named after Broyden, Fletcher, Goldfarb and Shanno.
and set:

\[ p_k = a_k d_k \]

\[ x_{k+1} = x_k + p_k \]

\[ q_k = g_{k+1} - g_k \]

\[ H_{k+1} = H_k - \frac{H_k q_k q_k^T H_k}{q_k^T H_k q_k} + \frac{p_k p_k^T}{q_k^T q_k} + \frac{\omega (q_k^T H_k q_k) r_k r_k^T}{q_k^T q_k} \]

where

\[ r_k = \frac{p_k}{q_k^T p_k} - \frac{H_k q_k}{q_k^T q_k} \]

Remark 1: The DFP and BFGS updates can also be written as the choices \( \omega = 0, 1 \) respectively, in the above class of updates.

Remark 2: Note that although \( H_0 \) was chosen to be symmetric, the subsequent matrices \( H_k \) in some V.M.M. (e.g. in Broyden's 1965 method) are not necessarily symmetric. (See Avriel, (1976), p.347). Since \( H_k \) is supposed to approximate the inverse Hessian - a symmetric matrix - it is reasonable to use an updating formula such that all matrices \( H_k \) are symmetric.

For Algorithm (I), it is clear that the trial matrix \( H_k \) is improved on the basis of the actual relations between changes in \( x_k \) and changes in the gradient. Associated with the positive definite matrix \( H_k \) is the norm defined by

\[ \|x_k\|_{H_k} = \sqrt{x_k^T H_k^{-1} x_k} \]
for points $x_k, z$ in $\mathbb{R}^n$. Thus $H_k$ induces the metric

$$d(x_k, z) = \|x_k - z\|_{H_k}.$$  

Davidon called his method a Variable Metric Method to reflect the fact that $H_k$ is changed after each iteration. The change in $H_k$ at each iteration affects the direction of "steepest descent" from a given point $x$, because this direction depends upon how the distance between two points $x_k$ and $z$ in $\mathbb{R}^n$ is measured. In general there is no reason to assume that a unit of distance along the $x_i$ axis is equal to a unit of distance along the $x_j$ axis, for $i \neq j$. The definition of distance, that is a metric, implies a particular system of weighting these units. Since the distance between $x_k$ and $z$ is defined by $\|x_k - z\|_{H_k}$, then the set of all points $z$ at a distance $\mu$ from $x_k$ is given by the ellipsoid

$$\|z - x_k\|_{H_k} = \mu,$$

that is,

$$\left\{z \mid (z - x_k)^T H_k^{-1} (z - x_k) = \mu^2 \right\}.$$  

The direction of "steepest descent" in the neighbourhood bounded by the ellipsoid may be defined as the direction from $x_k$ to that point on the ellipsoid for which the value of the function $f(x_k)$ is smallest. It can be seen that as $\mu$ tends to zero, this direction approaches a limit which is the direction of the vector

$$d_k = -H_k g(x_k).$$

Therefore this direction is called the direction of steepest descent from $x_k$ relative to $H_k$. If $H_k = I$, then
\[ \| x_k \|_{H_k} = \sqrt{x_k^T H_k x_k}, \]

is the Euclidean norm and \( d_k = -g(x_k) \) is called simply the direction of steepest descent from \( x_k \). This is the most common usage of the term "steepest descent". In particular, it is the direction used in the classical method of steepest descent described by A. Cauchy in 1847.

The BFGS method may be applied to a general differentiable function, but proof that the sequence of points generated by this method will always converge to a local minimum of the function, if one exists, can be given only for a restricted class of functions.

The following definition of Variable Metric Methods is broader than the definition which is given in some references. (See Avriel, (1976), p.322).

**Definition 5.1.1.**

A Variable Metric Method is an iterative minimization method which consists of the following iteration.

Given the point \( x \) and the matrix \( H \), let \( d = -Hg \), where \( g \) is the gradient of the function \( f(x) \) at \( x \). The next point \( x^* = x + ad \) is computed, where \( a \) is chosen to minimize \( f(x + ad) \) with respect to \( a \) and \( H \) is updated to \( H^* = H + C \), where \( C \) is a given correction matrix. Different Variable Metric Methods are obtained from different correction matrices.

The restriction of the matrices \( H \) to be positive definite forms a subclass of the variable metric algorithms.

If the difference between the gradient of function \( f(x) \) at
two successive points is denoted by \( q_k \) and the difference between the two successive points is denoted by \( p_k \), i.e.

\[
p_k = x_{k+1} - x_k \quad \text{and} \quad q_k = g(x_{k+1}) - g(x_k)
\]

then the approximation \( H_k \) to the inverse of the Hessian can be selected to satisfy the quasi-Newton equation:

\[
p_k = H_k q_k \quad \text{(5.1.1)}
\]

Variable Metric Methods in which \( H_k \) satisfies the relation (5.1.1) are called Quasi-Newton methods.

5.2 PROPERTIES OF THE BFGS METHOD USING A NEW MODEL.

Suppose that \( x \in \mathbb{R}^n \), \( Q \) is a \((n \times n)\) positive definite matrix and \( m(x) = \frac{1}{2} x^T Q x \). If \( F(x) \) is a nonlinear scaling of the quadratic \( m(x) \) of the form:

\[
\begin{cases}
F(x) = \frac{\varepsilon u(x)}{m(x) + 1}, & \varepsilon > 0
\end{cases}
\]

then it is clear that \( F(x) \) belongs to the class of rational functions. Since \( \varepsilon > 0 \) the first derivative of \( F(x) \) with respect to \( m(x) \):

\[
\frac{dF(m(x))}{dm(x)} = \frac{\varepsilon}{(1+m(x))^2}
\]

is positive.

The difference of the gradient of \( F(x) \) at two different points \( x_i \) and \( x_{i+1} \) can be expressed as follows:
\[ \nabla F(x_{i+1}) - \nabla F(x_i) = \frac{\varepsilon}{(1+m(x_{i+1}))^2} \nabla m(x_{i+1}) - \frac{\varepsilon}{(1+m(x_i))^2} \nabla m(x_i) = \]
\[
= \left(\frac{1+m(x_{i+1})}{1+m(x_i)}\right)^2 \left[ \frac{\varepsilon(1+m(x_i))^2}{(1+m(x_{i+1}))^4} \nabla m(x_{i+1}) - \frac{\varepsilon}{(1+m(x_i))^2} \nabla m(x_i) \right] =
\]
\[
= \phi_i \left[ \rho_i \nabla m(x_{i+1}) - \sigma_{i+1} \nabla m(x_i) \right]
\]

where
\[
\begin{align*}
\phi_i &= \left(\frac{1+m(x_{i+1})}{1+m(x_i)}\right)^2, \\
\rho_i &= \frac{\varepsilon}{(1+m(x_i))^2}, \\
\sigma_{i+1} &= \frac{\varepsilon(1+m(x_i))^2}{(1+m(x_{i+1}))^4}
\end{align*}
\]

Since \( \varepsilon > 0 \), it is clear that \( \phi_i, \rho_i, \sigma_{i+1} \) are positive scalars. It is shown in Appendix A that the quantities \( \phi_i, \rho_i \) and \( \sigma_{i+1} \) can be evaluated with already available quantities.

The following Algorithm includes Algorithm (I). Let \( f(x) \) be a differentiable real valued function, \( x \in \mathbb{R}^n \) and \( g(x) \) the gradient of \( f(x) \) at the point \( x \).

Algorithm (II): Begin with any positive definite matrix \( H_0 \), a starting point \( x_0 \) and index \( i = 0 \).

Step 1: Set \( d_i = -H_i g_i \).

Step 2: Minimize \( f(x_i + a d_i) \) with respect to \( a > 0 \) to obtain \( a_i \).

Set \( p_i = a_i d_i \), \( x_{i+1} = x_i + p_i \), and compute \( g_{i+1} \).
and \[ q_i = (\rho_i g_{i+1} - \sigma_{i+1} g_i) \phi_i, \]

where \( \phi_i, \rho_i, \sigma_{i+1} \) are given by (5.2.2). Select \( a_i \) accurate enough so \( p_i^T q_i > 0 \) holds.

**Step 3:** Set \( H_{i+1} = H_i + C_i \) \hspace{1cm} (II.3)

where the correction matrix \( C_i \) is selected in such a way that
\( H_{i+1} \) is a positive definite matrix. (In the following sections it is assumed that \( C_i \) is the BFGS update unless otherwise stated).

**Step 4:** Set \( i = i+1 \) and return to Step 1.

If Algorithm (II) is used with exact line search, then
\[ g_{i+1}^T d_i = 0 \quad \text{and} \quad d_i = -H_i g_i. \]
Thus
\[ p_i^T q_i = p_i^T (g_{i+1} - g_i) = -p_i^T g_i = a_i g_i^T H_i g_i. \]
So by the positive definiteness of \( H_i \), \( p_i^T q_i > 0 \). However, when the function \( f(x) \) is continuous and bounded below the condition of exact line-search is not necessary for \( p_i^T q_i \) to be positive, provided the steps taken are large enough.

For progress to be guaranteed at each step of the algorithm, the property of "stability" is required. This property was defined in the previous chapter.
The property of stability is not sufficient for convergence because the sequence of function values at the points generated by a stable method may not be bounded below.

**Corollary 5.2.1:** The BFGS method is stable if and only if $H_k$ (as defined in Algorithm (I)) is positive definite for $k = 1, 2, \ldots$.

The proof of Corollary 5.2.1. can be found in Appendix A.

**Remark:** If the BFGS method is stable and applied to the function (5.2.1) then $\phi_i \in [0, 1)$.

The following Theorem is concerned with the rate of decrease of a function of the form (5.2.1) when Algorithm (II) is applied to such a function.

**Theorem 5.2.1:**

For a positive definite quadratic function

$$q(x) = \frac{1}{2} (x - x^*)^T Q(x - x^*)$$

(5.2.3)

and for any starting point $x_0$, Algorithm (II) converges to the unique minimum $x^*$ of $F(x)$ given by (5.2.1). Furthermore, at every step $i$ the inequality:

$$F(x_{i+1}) \leq F(x_i) - A$$

(5.2.4)

holds, where $A > 0$, is a function of the condition number of the matrix

$$P_i = Q^{\frac{1}{2}} H_i Q^{\frac{1}{2}}.$$
Proof:
Let $k(P_i)$ denote the condition number of any matrix $P_i$.
For the sake of simplicity the subscript $(i+1)$ is replaced by (')
and the subscript $i$ is omitted. So

$$x' = x - aH\nabla F(x).$$

Let

$$g(x) = Q(x - x^*)$$

Then

$$\nabla F(q(x)) = \frac{\varepsilon}{(1 + q(x))^2} Q(x - x^*) = \sigma g(x).$$

So

$$F(x') = F(x - aH\nabla F(q(x))) =$$

$$= \frac{\varepsilon \left[ \frac{1}{2} (x - x^*)^T Q(x - x^*) - a(x - x^*)^T QHg\sigma \right.}{\left. \varepsilon + \frac{2}{\sigma^2} a^2 g^T H Q H g \right]}. \quad (5.2.5)$$

It can be seen that if (5.2.5) is considered as a function of $a$,
it is minimized at the point

$$a = \frac{T H g}{\sigma g^T H Q H g}. \quad (5.2.6)$$

If equation (5.2.6) is substituted into equation (5.2.5)
then:

$$F(x') = \frac{\varepsilon q(x) - \varepsilon h}{q(x) - h + 1}. \quad (5.2.7)$$

where

$$h = \frac{(g^T H g)^2}{2(g^T H Q H g)}. \quad (5.2.8)$$

The quadratic function $q(x)$ can be expressed as:

$$q(x) = \frac{1}{2} (x - x^*)^T Q Q^{-1} Q(x - x^*) = \frac{1}{2} g^T Q^{-1} g.$$
Therefore from equation (5.2.7):

\[ F(x') - F(x) = \frac{-ch}{(q(x)+1)(q(x)+1-h)} \]  (5.2.9)

By defining \( k = H^T Q H^T \) and \( s = H^T g(x) \) the quantities \( q(x) + 1 \) and \( q(x) - h + 1 \) can be expressed as:

\[ h = \frac{(s^T s)^2}{2(s^T ks)} \quad q - h = \frac{1}{2} (s^T k s - 1) - \frac{(s^T s)^2}{2(s^T ks)} = \]

\[ = \frac{(s^T k s - 1)}{2} \left[ 1 - \frac{(s^T s)^2}{(s^T k s)(s^T k s)} \right] \]

\[ q + 1 = 1 + \frac{s^T k s - 1}{2} \]

\[ F(x') - F(x) = \left[ \frac{-\varepsilon}{1 - \frac{(s^T s)^2}{(s^T k s)(s^T k s)(s^T k s)}} \right] \left[ 1 + \frac{s^T k s - 1}{2} \right] \frac{(s^T s)^2}{(s^T k s)(s^T k s)} = \]

\[ = (-W) \left[ \frac{(s^T s)^2}{(s^T k s)(s^T k s)} \right]^2 \]  (5.2.10)

By the Kantorovich inequality (Luenberger (1972), p.151) equation (5.2.10) can be written as:

\[ F(x') - F(x) = (-W) \frac{(s^T s)^2}{(s^T k s)(s^T k s)} \Rightarrow \]

\[ \Rightarrow F(x) - F(x') = W \frac{(s^T s)^2}{(s^T k s)(s^T k s)} \Rightarrow W \frac{4Mm}{(M+m)^2} \]  (5.2.11)
where \( M \) and \( m \) are the largest and the smallest eigenvalues of \( k \).

However, the matrices \( k \) and \( P \) are similar because \( k \) can be written as:

\[
k = (Q^\dagger H^\dagger)^{-1} P(Q^\dagger H^\dagger).
\]

Therefore \( M \) and \( m \) are also the largest and smallest eigenvalues of the matrix \( P \). Since for a symmetric matrix the condition number is the ratio of the largest to the smallest eigenvalue

\[
k(P) = \frac{M}{m}
\]

then the inequality (5.2.11) can be rewritten as:

\[
F(x) - F(x') \geq W \frac{4Mm}{(M+m)^2} = W \frac{4k(P)}{(1+k(P))^2} = A.
\]

Therefore

\[
F(x') \leq F(x) - A.
\] (5.2.12)

Inequality (5.2.12) is inequality (5.2.4) after the subscripts are restored.

An important property of Algorithm (II) is given by the following Proposition.

**Proposition 5.1:** Let the correction matrix \( C_i \) in Algorithm (II) be defined by:

\[
C_i = -\frac{H_i q^T_i H_i}{q_i^T H_i q_i} + \frac{P_i P_i^T}{\phi_i P_i^T q_i} + \omega \phi_i (q_i^T H_i q_i) r_i r_i^T
\] (5.2.13)

where \( r_i \) is given in Algorithm (I) and \( q_i, p_i, \phi_i \) are defined in
Algorithm (II).

If $H_i$ is a positive definite matrix then $H_{i+1}$ is also positive definite.

Proof:

For every $x \in \mathbb{R}^n$, $x \neq 0$, equations (5.2.13), (II.3) give:

$$
 x^T H_{i+1} x = x^T H_i x - \frac{(q_i^T H_i x)^2}{q_i^T H_i q_i} + \frac{(p_i^T x)^2}{\phi_i p_i^T q_i}.
$$

(5.2.14)

If $a = H_i^T x$ and $b = H_i^T q_i$, equation (5.2.14) can be rewritten as:

$$
 x^T H_{i+1} x = \left[ a^T a - \frac{(b^T a)^2}{b^T b} \right] + \frac{(p_i^T x)^2}{p_i^T q_i \phi_i}.
$$

(5.2.15)

The term in the square brackets is non-negative by the Cauchy-Schwartz inequality. The last quantity on the right hand side of (5.2.15) is non-negative by the assumptions of the proposition.

It will next be proved that all the terms in the right hand side of equation (5.2.15) cannot vanish simultaneously. The terms in the square brackets vanish if and only if $a$ and $b$ are collinear. This implies that $x$ and $q_i$ are collinear, e.g. $x = w q_i$. In that case

$$
 x^T p_i = w q_i^T p_i = w p_i^T q_i \neq 0.
$$

So the last term in equation (5.2.15) is clearly non-zero. Therefore

$$
 x^T H_{i+1} x > 0
$$

for all non-zero $x$. 

The property described in Proposition 5.1 depends on the algorithm, only for the fact that $p_t q_i$ has to be positive. It can otherwise be attributed as a property of the updating formula (5.2.13). If an algorithm with an exact line-search is used, $p_t q_i > 0$ is implied by the positive definiteness of the matrix $H_i$.

Remark: It should be noted that the scalars $\phi_i$, $\sigma_i$, $\rho_i$ given by (5.2.2) can be computed by already known quantities of Algorithm (II). (Appendix).

From the previous Proposition 5.1, it can be seen that in Algorithm (II), $H_i$ is positive definite for $i = 1, 2, \ldots$. The proof is based on the fact that $H_o$ is positive definite and $p_t q_i > 0$ for $i = 1, 2, \ldots$ and follows by induction.

5.3 THE CONDITIONING OF THE VARIABLE METRIC METHODS USING THE PROPOSED UPDATES.

In the previous paragraph it was seen that VMM (Variable Metric Methods) are algorithms for minimizing a differentiable function $f(x)$ over $\mathbb{R}^n$ given that the gradients $g(x)$ are available for all $x \in \mathbb{R}^n$. The iteration which these methods are based on is:

$$x_{k+1} = x_k - a_k H_k g_k$$

(5.3.1)

where $H_k$ is a $n \times n$ matrix approximating the inverse Hessian $G^{-1}(x_k)$ and the scalar $a_k$ satisfies prescribed descent conditions. The $\theta$-class of updates was introduced firstly by Broyden (1967) and later rewritten in the following form by Fletcher (1970).
\[ H_{i+1} = (1-\theta) H_{DFP}^{i+1} + \theta H_{BFGS}^{i+1} \] (5.3.2)

where \( \theta \in [0, 1] \) and \( i = 0, 1, 2, \ldots \)

The most important members of this \( \theta \)-class are the DFP-updates:

\[ H_{DFP}^{i+1} = H_i - \frac{H_i q_i q_i^T H_i}{q_i^T H_i q_i} + \frac{p_i p_i^T}{p_i^T q_i} \quad i = 0, 1, 2, \ldots \]

where \( p_i = x_{i+1} - x_i \) and \( q_i = g_{i+1} - g_i \)

which corresponds to \( \theta = 0 \) in the \( \theta \)-class and the BFGS updates:

\[ H_{BFGS}^{i+1} = \left( I - \frac{q_i q_i^T}{p_i^T q_i} \right) H_i \left( I - \frac{q_i q_i^T}{p_i^T q_i} \right) + \frac{q_i q_i^T}{p_i^T q_i} \]

which corresponds to \( \theta = 1 \). These two updates are sometimes called complementary updates. In this paragraph the modified form of DFP-update, given by (5.2.13) is considered. The following theorem establishes a bound on the condition number of the matrix \( H_{i+1} \) given by

\[ H_{i+1} = H_i - \frac{H_i q_i q_i^T H_i}{q_i^T H_i q_i} + \frac{p_i p_i^T}{p_i^T q_i} \quad i = 0, 1, 2, \ldots \] - (5.3.3)

**Theorem 5.3.1:**

Let \( H_{i+1} \) be given by (5.3.3). For the sake of brevity the notation \( H \) will be used instead of \( H_{i+1} \) and the subscript \( i \) will be dropped. If \( p = x - x \) and \( q = g(x) - g(x) \) then the following quantities are defined:
\[
\begin{align*}
\begin{cases}
b = p^T q, & a = q^T H q, & c = p^T H^{-1} p
\end{cases}
\end{align*}
\]

In terms of the above notation, equation (5.3.3) can be rewritten as:
\[
\bar{H} = H - \frac{H q q^T H}{a} + \frac{p p^T}{\phi b}.
\]

Let \( k(\cdot) \) denote the condition number of the matrix \( \cdot \) with respect to the spectral norm (i.e. \( k(A) = \|A\| \|A^{-1}\| \)). Then if the matrices \( H \) and \( \bar{H} \) are positive definite,
\[
k(\bar{H}) \leq k(H) \left\{ \frac{2 - V + \sqrt{V - \mu}}{2 - V - \sqrt{V - \mu}} \right\} \tag{5.3.4}
\]
where \( V = 1 + \frac{c}{\phi b} \) and \( \mu = -\frac{4b}{\phi a} \).

**Proof:**

For all \( x \in \mathbb{R}^n, x \neq 0 \), equation (5.3.3) gives:
\[
x^T \bar{H} x = x^T H x W(\phi, x) \tag{5.3.5}
\]
where
\[
W(\phi, x) = 1 - \frac{(x^T H q)^2}{(q^T H q)(x^T H x)} + \frac{(p^T x)^2}{\phi (p^T q)(x^T H x)}.
\]

If \( C(\phi) = \frac{\max x W(\phi, x)}{\min x W(\phi, x)} \),

then since \( H \) and \( \bar{H} \) are both positive definite,
\[
k(\bar{H}) = \|\bar{H}\| \|\bar{H}^{-1}\| \leq \frac{\max x \frac{T H x}{x^T H x}}{\min x \frac{T H x}{x^T H x}} C(\phi) = k(H) C(\phi). \tag{5.3.6}
\]
The algebra that follows is simplified by the introduction of the following notation:

\[
\begin{align*}
L = H^{\frac{1}{2}}, & \quad y = \frac{Lx}{(x^T H x)^{\frac{1}{2}}}, & \quad v = L^{-1} p, & \quad u = Lq \end{align*}
\]

Using the above notation \(W(\phi, x)\) can be rewritten as:

\[
W(\phi, x) = 1 - \frac{(u^T y)^2}{a} + \frac{(v^T y)^2}{\phi v^T u} \overset{d e f}{=} M(\phi, y). \quad (5.3.7)
\]

Consequently \(C(\phi) = \max_{\|y\| = 1} \frac{M(\phi, y)}{\min_{\|y\| = 1} M(\phi, y)}\).

The necessary conditions for an extremum of \(M(\phi, y)\) to exist over the hypersphere \(\|y\| = 1\) are:

\[
\begin{align*}
\forall y \quad M(\phi, y) + 2\lambda y &= 0 \\
y^T y &= 1
\end{align*}
\]

If the gradient of \(M(\phi, y)\) with respect to \(y\) is expressed analytically then:

\[
\begin{align*}
-\frac{2u^T y u}{a} + \frac{2v^T y v}{\phi v^T u} + 2\lambda y &= 0 \quad (5.3.8) \\
y^T y &= 1 \quad (5.3.9)
\end{align*}
\]

If the inner products of \((5.3.8)\) are taken with \(u, y\) and \(v\) then:

\[
\begin{align*}
nonumber -\frac{(u^T y)(u^T u)}{a} + \frac{(v^T y)(v^T u)}{\phi v^T u} + \lambda (y^T u) &= 0 \\
-\frac{(u^T y)(u^T y)}{a} + \frac{(v^T y)(v^T y)}{\phi v^T u} + \lambda y^T y &= 0 \\
-\frac{(u^T y)(u^T v)}{a} + \frac{(v^T y)(v^T v)}{\phi v^T u} + \lambda y^T v &= 0
\end{align*}
\]
If the first and third equations above are divided by $u^T y$:

\[
\begin{align*}
-\frac{u^T u}{a} + \frac{v^T y}{u^T y} + \lambda &= 0 \\
-\frac{u^T v}{a} + \frac{v^T v}{\phi^T u} + \frac{v^T}{u^T y} + \lambda \frac{v^T y}{u^T y} &= 0
\end{align*}
\]

A new parameter $d = \frac{v^T y}{u^T y}$ may be introduced and the last equations can be rewritten as:

\[
\begin{align*}
-\frac{u^T u}{a} + \frac{d}{\phi} + \lambda &= 0 \\
-\frac{u^T v}{a} + \frac{v^T v}{\phi^T u} d + \lambda d &= 0
\end{align*}
\] (5.3.10)

Solving equations (5.3.10) for $d$ and $\lambda$ results in:

\[
\begin{align*}
-\frac{u^T v}{a} + \frac{d}{\phi} \left( \frac{v^T v}{\phi^T u} \right) + \frac{d}{\phi} \left( \frac{u^T u}{a} - \frac{d^2}{\phi} \right) &= 0
\end{align*}
\]

or

\[
d^2 \left( \frac{1}{\phi} \right) - d \left( \frac{u^T u}{a} + \frac{v^T v}{\phi^T u} \right) + \frac{u^T v}{a} = 0 .
\] (5.3.11)

The roots of (5.3.11) are real numbers because

\[
\left( \frac{u^T u}{a} + \frac{v^T v}{\phi^T u} \right)^2 - \frac{4u^T v}{a\phi} > 0 .
\] (5.3.12)

The left hand side of the previous relation can be rewritten as:
\[
\left(1 - \frac{c}{\phi b}\right)^2 + \frac{4c}{\phi b} - \frac{4b}{\phi^2} = 0
\]

However \(\phi, a\), are positive numbers and if an exact line-search is used by the algorithm, then \(b > 0\). Therefore the roots of (5.3.11) are real and distinct.

If
\[
v = \frac{u^T u}{a} + \frac{v^T v}{\phi a} u^T u \quad \text{and} \quad \mu = 4u^T v
\]
then
\[
d = \phi \left\{ \frac{v \pm \sqrt{v^2 + \mu}}{2} \right\}.
\]

The second equation of (I) is substituted in (5.3.7) and results in:
\[
M(\phi, y) = 1 - \lambda.
\]

From the equation
\[
\frac{u^T u}{a} + \frac{d}{\phi} + \lambda = 0
\]
\(\lambda\) is obtained:
\[
\lambda = 1 - \frac{d}{\phi}.
\]

In view of equation (5.3.13) the above equation gives:
\[
\lambda = \frac{2 - v \pm \sqrt{v^2 + \mu}}{2}.
\]

Therefore,
\[
\begin{align*}
\max_{\|y\| = 1} M(\phi, y) &= 2 - v + \sqrt{v^2 + \mu} \\
\min_{\|y\| = 1} M(\phi, y) &= 2 - v - \sqrt{v^2 + \mu}
\end{align*}
\]
The relation (5.3.4) follows directly from (5.3.7), (5.3.6) and (5.3.15).

From a numerical stability point of view, it is desirable to employ an updating formula that will minimize the condition number of $H_k$ at each iteration. However, an explicit expression for this condition number is not usually available. It seems reasonable to minimize its upper bound by proper selection of $\phi$. The following definition will be used in the next theorem (Oren and Spedicato (1976)).

**Definition 5.3.1:**
Let $H_i$ be a positive definite $(n \times n)$ matrix, $i = 0, 1, 2, \ldots$ and $p_i, q_i \in \mathbb{R}^n$ satisfy $p_i^T q_i > 0$. Then $H_{i+1}$ given by (5.3.3) is said to be optimally conditioned if $\phi$ is such that the right hand side of (5.3.4) is minimized.

**Theorem 5.3.2:**
Let $H, \overline{H}, p, q, \phi$ be defined as in Theorem 5.3.1. Then $\overline{H}$ is optimally conditioned if and only if $\phi$ is given by the expression:

$$\phi = \frac{c}{b} \quad (5.3.16)$$

and $ac \geq b^2$.

**Proof:**
Clearly $\overline{H}$ is optimally conditioned if $\phi$ minimizes the right hand side of (5.3.4). By the definition of $\nu$ and $\mu$ and of $a, b, c, u$ and $v$, then:
\[
\begin{align*}
\left\{ \nu = 1 - \frac{c}{b\phi} \quad \text{and} \quad \mu = \frac{4b}{a\phi} \right\}.
\end{align*}
\] (5.3.17)

If \(\frac{1}{\phi}\) is substituted by \(\phi'\), the expression (5.3.17) can be rewritten as:
\[
\left\{ \nu = 1 - \frac{c}{b} \phi' \quad \text{and} \quad \mu = \frac{4b}{a} \phi' \right\}.
\] (5.3.18)

For the sake of simplicity the dash (') is omitted. Therefore, the right hand side of (5.3.4) can be rewritten as:
\[
2 - \frac{\nu^{2} + \mu}{2 - \sqrt{\nu^{2} + \mu}} = k(H)
\]

\[
= k(H) \, G(\phi)
\]

where
\[
G(\phi) = \frac{2 - \nu + \sqrt{\nu^{2} + \mu}}{2 - \nu - \sqrt{\nu^{2} + \mu}}.
\] (5.3.19)

By calling \(A(\phi) = 2 - \nu\) and \(B(\phi) = \sqrt{\nu^{2} + \mu}\), equation (5.3.19) can be rewritten as:
\[
G(\phi) = \frac{A(\phi) + B(\phi)}{A(\phi) - B(\phi)}.
\] (5.3.20)

Clearly \(\overline{H}\) is optimally conditioned if and only if \(\phi\) minimizes \(G(\phi)\) defined by (5.3.20). Then \(G(\phi)\) will have a stationary point with respect to \(\phi\) if
\[
\frac{dG(\phi)}{d\phi} = 0.
\] (5.3.21)

Equation (5.3.21) can be written as:
\[
\frac{d}{d\phi} \left( \frac{A(\phi) + B(\phi)}{A(\phi) - B(\phi)} \right) = \frac{2}{(A(\phi) - B(\phi))^2} \left( A(\phi) \frac{dB(\phi)}{d\phi} - B(\phi) \frac{dA(\phi)}{d\phi} \right) = 0
\]

\[
\Rightarrow A(\phi) \frac{dB(\phi)}{d\phi} = B(\phi) \frac{dA(\phi)}{d\phi} \quad (A(\phi) \neq B(\phi))
\]

\[
A(\phi) = 3 - \frac{c}{b} \Rightarrow \frac{dA(\phi)}{d\phi} = \frac{c}{b}
\]

\[
B(\phi) = \sqrt{1 - \frac{c}{b}}^2 - \frac{4b\phi}{a} \Rightarrow \frac{dB(\phi)}{d\phi} = \frac{1}{2B(\phi)} \left\{ 2 \left[ 1 - \frac{c\phi}{b} \right] \left[ - \frac{c}{b} \right] - \frac{4b}{a} \right\}
\]

Then

\[
A(\phi) \frac{dB(\phi)}{d\phi} - B(\phi) \frac{dA(\phi)}{d\phi}
\]

equals:

\[
= \frac{1}{2B} \left\{ \left[ 1 + \frac{c}{b} \phi \right] \left[ \left( 2 - \left( \frac{c}{b} \phi \right) \right) \left( - \frac{c}{b} \right) \right] - \frac{4b}{a} \right\} + \frac{c}{b} \cdot B =
\]

\[
= \frac{1}{2B} \left\{ 2 \left( \frac{c}{b} \phi \left[ \frac{c^2}{b^2} - 1 \right] \right) - \frac{2c}{b} \left[ \left( 1 - \frac{c}{b} \phi \right)^2 - \frac{4b\phi}{a} \right] \right\}
\]

\[
- \frac{4b}{a} \left\{ 1 + \frac{c}{b} \phi \right\}
\]

\[
= \frac{1}{B} \left\{ \phi \left[ 2 \frac{c^2}{b^2} + \frac{2c}{a} \right] - \frac{2b}{a} - \frac{2c}{b} \right\} = 0
\]

\[
\Rightarrow \phi = \left[ \frac{c}{b} + \frac{b}{a} \right] / c \left( \frac{c}{b^2} + \frac{1}{a} \right) = \frac{b}{c}
\]

\[
(5.3.22)
\]

Since \( \phi = \frac{1}{\phi} \) then \( \phi = \frac{c}{b} \).

By examining the second derivative of \( G(\phi) \) at the point \( \phi \),
given by (5.3.22), it is clear that (5.3.22) corresponds to a minimum of $G(\phi)$.

The case $A(\phi) = B(\phi)$ is now considered.

From the definition of $A(\phi)$ and $B(\phi)$ is implied that:

$$2 - \nu = \sqrt{\mu^2 + \mu}$$

$$1 + \frac{c_\phi}{b} = \sqrt{\left(1 - \frac{c_\phi}{b}\right)^2 - \frac{4b\phi}{a}}$$

$$\left(1 + \frac{c_\phi}{b}\right)^2 = \left(1 - \frac{c_\phi}{b}\right)^2 - \frac{4b\phi}{a}$$

$$1 + \left(\frac{c_\phi}{b}\right)^2 + \frac{2c_\phi}{b} =$$

$$= 1 + \left(\frac{c_\phi}{b}\right)^2 - \frac{2c_\phi}{b} - \frac{4b\phi}{a}$$

$$a = -b^2.$$

This last relation contradicts the assumption of the Theorem.

5.4 THE EIGENVALUES STRUCTURE OF THE UPDATING FORMULA.

In this section the properties of the updating formula for $H_i$, defined by equation (5.3.3) are considered. Considering Algorithm (II), it is well known that Variable Metric Methods construct approximations $H_i$ to the inverse Hessian. As a measure to characterize the approximation $H_i$ to the inverse Hessian $G^{-1}$ the matrix $R_i = G^1 H_i^{-1} G^1$ may be used. (Broyden (1970)). It is
clear that the approximation is perfect if \( R_i = I \). In this respect the eigenvalue structure of \( R_i \) is very important.

**Lemma 5.4.1:**

If \( H_{i+1} \) is given by equation (5.3.3) the following holds

\[ p_i^T q_i > 0, \text{ and } q_i = Q p_i \] where \( Q \) is a positive definite symmetric matrix and if

\[ R_i = Q^\frac{1}{2} H_i Q^\frac{1}{2} \text{ and } z = Q^\frac{1}{2} p_i \]

then

\[ R_{i+1} = Q^\frac{1}{2} \left\{ H_{i+1} \right\} Q^\frac{1}{2} = R_i - \frac{R_i z_i^T z_i R_i}{z_i^T R_i z_i} + \frac{z_i z_i^T}{\phi_i z_i^T z_i} \]  \hspace{1cm} (5.4.1)

for \( i = 0, 1, 2, \ldots \).

**Proof:**

If the left and right hand sides of equation (5.3.3) are multiplied by \( Q^\frac{1}{2} \) and the relation \( q_i = Q p_i \) is used, then:

\[ Q^\frac{1}{2} H_{i+1} Q^\frac{1}{2} = Q^\frac{1}{2} H_i Q^\frac{1}{2} - \frac{Q^\frac{1}{2} H_i Q p_i p_i^T Q H_i Q^\frac{1}{2}}{p_i^T Q H_i Q p_i} + \]

\[ + \frac{Q^\frac{1}{2} p_i p_i^T Q^\frac{1}{2}}{\phi_i p_i^T Q p_i} \]

The right hand side of the above equation is expressed in terms of \( R_i \) and \( z_i \) as:

\[ R_i = \frac{R_i z_i^T z_i R_i}{z_i^T R_i z_i} + \frac{z_i z_i^T}{\phi_i z_i^T z_i} \]
which is the right hand side of equation (5.4.1).

The results of Lemma 1 can be applied directly in the following Proposition.

**Proposition 5.4.1:** Let $R_{i+1}$ be given by equation (5.4.1), for some positive definite matrix $R_i$ and a non-zero vector $z$, and let the eigenvalues of $R_{i+1}$ be denoted by:

$$
\mu_{i+1}^{1}(\theta_i) \leq \mu_{i+1}^{2}(\theta_i) \leq \ldots \leq \mu_{i+1}^{n}(\theta_i)
$$

where $\theta_i = \frac{1}{\phi_i}$. Then for $\theta_i \in (0, 1]$ there holds:

$$
\mu_i^k(0) < \mu_i^k(\theta_i) \leq \mu_i^k(1) \quad \text{for } k = 1, 2, \ldots, n
$$

and $i = 0, 1, 2, \ldots$.

**Proof:**

By using equation (5.4.1):

$$
R_{i+1}(\theta_i) = R_{i+1}(0) + \theta_i \frac{z_i z_i^T}{z_i^T z_i}
$$

(5.4.2)

$$
R_{i+1}(1) = R_{i+1}(0) + \frac{z_i z_i^T}{z_i^T z_i}
$$

(5.4.3)

Equation (5.4.3) is then subtracted from (5.4.2):

$$
R_{i+1}(1) = R_{i+1}(\theta_i) + (1 - \theta_i) \frac{z_i z_i^T}{z_i^T z_i}
$$

Using the interlocking eigenvalues theorem (Appendix A) and the assumptions of the Proposition ($(1 - \theta_i) > 0$ and $\theta_i > 0$)
there holds
\[
\mu_i^1(0) < \mu_i^1(\theta_i) < \mu_i^2(0) < \mu_i^2(\theta_i) < \ldots < \mu_i^n(0) < \mu_i^n(\theta_i)
\]
\[
\mu_i^1(\theta_i) < \mu_i^1(1) < \mu_i^2(\theta_i) < \mu_i^2(1) < \ldots < \mu_i^n(\theta_i) < \mu_i^n(1)
\]
Hence the relation
\[
\mu_i^k(0) < \mu_i^k(\theta_i) < \mu_i^k(1), \quad k = 1, 2, \ldots, n \quad i = 0, 1, 2, \ldots
\]
follows directly from the last two relations.

By modifying Corollary 6 of the paper by Oren & Luenberger (1974) the following theorem is implied:

**Theorem 5.4.1:**

Let \( R_{i+1}(\theta_i) \), \( \mu_i(\theta_i) \), \( \theta_i \) be as in Proposition (5.4.1) and denote the condition number of a matrix by \( k(\cdot) \) as usual. Then for \( \theta_i \in [0, 1] \) there are three possible cases:

i) If \( \lambda_i > 1 \) then \( \frac{1}{\lambda_i} \geq k(R_{i+1}(\theta_i)) \geq \frac{1}{\lambda_{n-1}} \)

ii) If \( \lambda_n \leq 1 \) then \( \frac{1}{\lambda_i} \geq k(R_{i+1}(\theta_i)) \geq \frac{1}{\lambda_2} \)

iii) If \( \lambda_n \geq 1 \geq \lambda_i \) then \( k(R_i) \geq k(R_{i+1}(\theta_i)) \)

where \( \lambda_i, \quad i = 1, 2, \ldots, n \) are the eigenvalues of \( R_i \).

The Proof of Theorem 4.1 is omitted as it is a straightforward modification of Corollary 6 of Oren & Luenberger (1974).
COMPARISON OF THE CONDITION NUMBERS.

In previous sections of the fifth chapter different updates of the symmetric and positive definite inverse of the Hessian matrix $B_{i+1}$ have been studied. These updates satisfy the following conditions:

(i) $B_{i+1} p_i = q_i$ for $p_i, q_i$ non-zero n-vectors,

(ii) $B_{i+1}$ is symmetric and positive definite,

(iii) $(B_{i+1} - B_i)z = 0$ for all $z \in D_i$, where $D_i$ is a $i$-dimensional subspace of $\mathbb{R}^n$, $1 \leq i \leq n-2$, i.e.

$$D_i = \left\{ p_1, p_2, \ldots, p_{i-1} \right\}$$

assuming $\{p_1, \ldots, p_{n-2}\}$ are linearly independent.

The third condition was used by Davidon (1975) to enable past Quasi-Newton equations $B_{i+1} p_j = q_j$, $j < i$ to be retained for quadratic problems. This leads to finite termination for quadratic problems without using line searches. Among the matrices $B_{i+1}$ satisfying the above conditions are the updates:

$$B_{i+1} = \left( I - \frac{q_i p_i^T}{q_i p_i} \right) B_i \left( I - \frac{q_i p_i^T}{q_i p_i} \right) + \frac{q_i q_i^T}{q_i p_i} +$$

$$+ \theta \left[ B_i - \frac{B_i p_i p_i^T B_i}{p_i B_i p_i} + \frac{p_i p_i^T}{q_i p_i} \right]$$

(5.5.1)

where $\theta$ lies in the interval $[0, 1]$ and $p_i, q_i$ are defined by Algorithm (II).

The updates of $B_{i+1}$ which correspond to the two values of $\theta$, $\theta = 0$ and $\theta = 1$ are called the DFP and BFGS updates respectively.
One way to select an update satisfying the conditions (i) and (ii) is to choose the $B_{i+1}$ which minimizes

$$\|B_{i+1} - B_1\| \quad \text{or} \quad \|H_{i+1} - H_1\|$$

subject to conditions (i) and (ii), where $H^{-1}_i = B_i$.

The condition number of a non-singular matrix $A$, denoted by $k(A)$, is defined as

$$\|A\|_2 \|A^{-1}\|_2$$

If $A$ is symmetric and positive definite then $k(A)$ is the ratio of the largest to the smallest eigenvalues of $A$. It is well known that the difficulty in solving a linear system $Ax = b$ is proportional to $k(A)$. Since in a Quasi-Newton method, $B_{i+1}(\theta)$ could have been used in solving a linear system

$$B_{i+1}(\theta) d_i = -g_{i+1}$$

where $d_i$ is the $i$-th direction and $g_{i+1}$ is the gradient at the $(i+1)$th point, it is reasonable to try to keep the condition number of $B_{i+1}(\theta)$ bounded at each iteration. (If $H_{i+1}(\theta)$ is used instead of $B_{i+1}(\theta)$ then $H_{i+1}(\theta)$ is used to solve the linear system $d_i = -H_{i+1}(\theta) g_{i+1}$).

In this section a bound on the condition number of $B_{i+1}(\theta)$ is derived. The condition numbers of the BFGS class of updates with $\phi_i = 1$ are also compared with the condition numbers of the BFGS class of updates given by (5.5.1) and with $\phi_i \in [0, 1]$. The approach is different from that of Davidon (1975) and Oren & Spedicato (1976) in that the eigenvalues of
are calculated analytically. This gives information about the curves \( k(\theta) \).

Notice that since

\[
 k(H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{-\frac{1}{2}}) = k(B_i^{-\frac{1}{2}} B_{i+1}(\theta) B_i^{-\frac{1}{2}})
\]

for

\[
 H_{i+1}(\theta) = B_{i+1}(\theta)
\]

then the analysis of conditioning is independent of whether \( B_i \) or \( H_i \) is being updated.

If \( B_{i+1} \) is given by (5.5.1) then \( H_{i+1} \) can be expressed as:

\[
 H_{i+1} = H_i - \frac{H_i q_i q_i^T H_i}{a} + \frac{p_i p_i^T}{b} + \theta a \left( \frac{p_i}{b} - \frac{H_i q_i}{a} \right) \left( \frac{p_i}{b} - \frac{H_i q_i}{a} \right)^T
\]

where

\[
 a = q_i^T H_i q_i
\]

and

\[
 b = q_i^T p_i
\]

If the following product is defined:

\[
 A(\theta) = H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{-\frac{1}{2}}
\]

then

\[
 A(\theta) = I + A_1(\theta)
\]

where \( A_1(\theta) \) is a matrix of rank two belonging to the subspace

\[
 \begin{bmatrix} H_i^{-\frac{1}{2}} q_i, & H_i^{-\frac{1}{2}} p_i \end{bmatrix}
\]

Therefore \( A(\theta) \) has \( n-2 \) unit eigenvalues corresponding to the \( n-2 \) dimensional eigenspace orthogonal to

* For the sake of simplicity \( \phi_i \) is omitted from the numerical calculations which follow.
and two eigenvectors \( \omega_1 \) and \( \omega_2 \). Since

\[
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\]

and

\[
\begin{bmatrix}
H_{1}^{i} H_{1}^{i+1}(\theta) H_{1}^{i}
\end{bmatrix}
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix} = H_{1}^{i}
\]

(5.5.3)

and

\[
\begin{bmatrix}
H_{1}^{i} H_{1}^{i+1}(\theta) H_{1}^{i}
\end{bmatrix}
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix} = H_{1}^{i} p_{i} - \left(\frac{(H_{1}^{i} q_{i})^T (H_{1}^{i} p_{i})}{a}\right)
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\]

\[
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\]

it is clear that

\[
H_{1}^{i} H_{1}^{i+1}(\theta) H_{1}^{i}
\]

maps the subspace

\[
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\]

into itself and hence possesses two eigenvectors in

\[
\begin{bmatrix}
H_{1}^{i} q_{i}, H_{1}^{i} p_{i}
\end{bmatrix}
\]

Since the matrix \( H_{1}^{i} H_{1}^{i+1}(\theta) \) is equal to

\[
H_{1}^{i} H_{1}^{i+1}(\theta) H_{1}^{i} H_{1}^{i}
\]

and therefore is similar to

\[
H_{1}^{i} H_{1}^{i+1}(\theta) H_{1}^{i}
\]
then
\[
\det \left( \begin{bmatrix} H_i^{-1} & H_{i+1}(	heta) \\ \end{bmatrix} \right) = \det \left( \begin{bmatrix} H_i^{-1} & H_{i+1}(	heta) \end{bmatrix} H_i^{-1} \right) = \lambda_1 \lambda_2 = \\
= \frac{b}{a} + \theta \left( \frac{c}{b} - \frac{b}{a} \right). \tag{5.5.4}
\]

From the relationship (5.5.3) it is obvious that \( H_i \frac{1}{2} q_i \) cannot be an eigenvector of
\[
H_i^{-\frac{1}{2}} H_{i+1}(	heta) H_i^{-\frac{1}{2}}
\]
which means that if
\[
a_1 H_i^{\frac{1}{2}} q_i + a_2 H_i^{-\frac{1}{2}} p_i
\]
is an eigenvector then \( a_2 \) cannot be zero. It is then sufficient to assume that \( a_2 = 1 \) in seeking the eigenvectors \( w_1 \) and \( w_2 \) in the subspace
\[
\left[ H_i^{\frac{1}{2}} q_i, H_i^{-\frac{1}{2}} p_i \right].
\]

If \( H_i^{\frac{1}{2}} q_i \) and \( H_i^{-\frac{1}{2}} p_i \) are linearly independent then \( \lambda_1 \neq \lambda_2 \) and if \( H_i^{\frac{1}{2}} q_i \) and \( H_i^{-\frac{1}{2}} p_i \) are linearly dependent then \( H_{i+1}(\theta) \) is independent of \( \theta \).

Let
\[
w_1 = H_i^{-\frac{1}{2}} p_i - \alpha H_i^{\frac{1}{2}} q_i
\]
and
\[
w_2 = H_i^{-\frac{1}{2}} p_i - \beta H_i^{\frac{1}{2}} q_i
\]
then since
\[
H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{-\frac{1}{2}}
\]
is symmetric, eigenvectors corresponding to different eigenvalues are orthogonal which implies that \( w_1^T w_2 = 0 \), i.e.

\[
\left( H_i^{-\frac{1}{2}} p_i - \alpha H_i^{\frac{1}{2}} q_i \right) \left( H_i^{-\frac{1}{2}} p_i - \beta H_i^{\frac{1}{2}} q_i \right) = \alpha \beta a - b (\alpha + \beta) + c = 0. \tag{5.5.4a}
\]

But
\[
H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{\frac{1}{2}} \omega_1 = \lambda_1 \omega_1 = \lambda_1 \left( H_i^{-\frac{1}{2}} p_i - \alpha H_i^{\frac{1}{2}} q_i \right)
\]
and
\[
H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{\frac{1}{2}} \omega_1 = H_i^{-\frac{1}{2}} p_i - \alpha H_i^{\frac{1}{2}} q_i - \frac{b - \alpha a}{a} H_i^{\frac{1}{2}} q_i +
\]
\[
+ \frac{c - \alpha b H_i^{-\frac{1}{2}} p_i}{b} + \theta a \left( \frac{H_i^{-\frac{1}{2}} p_i}{b} - \frac{H_i^{\frac{1}{2}} q_i}{a} \right) \left( ac - b^2 \right) =
\]
\[
= \left( 1 - \frac{c}{b} - \alpha + \theta \left( \frac{ac - b^2}{b^2} \right) \right) H_i^{-\frac{1}{2}} p_i - \left( \frac{b}{a} + \theta \left( \frac{c}{b} - \frac{b}{a} \right) \right) H_i^{\frac{1}{2}} q_i.
\tag{5.5.5}
\]

Solving equation (5.5.4) and (5.5.5) with respect to \( \lambda_1 \lambda_2 \) and \( \theta \) gives:
\[
H_i^{-\frac{1}{2}} H_{i+1}(\theta) H_i^{\frac{1}{2}} \omega_1 = \left( -\frac{c}{b} - \alpha + \frac{a}{b} \lambda_1 \lambda_2 \right) H_i^{-\frac{1}{2}} p_i - \lambda_1 \lambda_2 H_i^{\frac{1}{2}} q_i
\]
which by (5.5.4) and (5.5.4a) implies:
\[
\alpha = \lambda_2 \quad \text{and} \quad \beta = \lambda_1.
\]

From equations (5.5.4) and (5.5.4a) the two eigenvalues \( \lambda_1 \) and \( \lambda_2 \) can be expressed as:
\[
\lambda_1 \lambda_2 = \frac{b}{a} + \theta \left( \frac{c}{b} - \frac{b}{a} \right)
\]
and
\[
a \lambda_1 \lambda_2 - b (\lambda_1 + \lambda_2) + c = 0.
\]

From the last two equations, it is clear that \( \lambda_1 \) and \( \lambda_2 \) are the
roots of the equation:
\[ \lambda^2(ab^2) - \lambda(ab^2 + \theta(ac-b^2)a + cb) + b(b^2 + \theta(ac-b^2)) = 0 \]  \hspace{1cm} (5.5.5a)

Therefore, if the following quantities are defined:
\[ f_1(\theta) = \frac{b^2 + cb + \theta(ac-b^2)}{2b^2} \]
and
\[ f_2(\theta) = \frac{b^2 + \theta(ac-b^2)}{ab} \]
then
\[ \lambda_{1,2} = f_1(\theta) \pm \left\{ f_1^2(\theta) - f_2(\theta) \right\}^{\frac{1}{2}}. \]

It can be seen that \( H_{i+1}(\theta) \) is positive definite if and only if \( b > 0 \).

A comparison follows of the condition numbers of the DFP and BFGS families with the ones of the corresponding DFP and BFGS families which are defined by (5.5.6) if \( p_i \) and \( q_i \) are given by the relations
\[
\begin{align*}
p_i &= x_{i+1} - x_i \quad \text{and} \quad q_i = \phi_i \left( p_i g_{i+1} - \sigma_{i+1} g_i \right) \\
\hat{H}_{i+1} &= H_i - \frac{H_i q_i q_i^T}{a} + \frac{1}{\phi_i} \frac{p_i p_i^T}{b} + \\
&\quad + \theta a \phi_i^2 \left( \frac{p_i}{b} \frac{1}{\phi_i} - \frac{H_i q_i 1}{a} \frac{1}{\phi_i} \right)^T \left( \frac{p_i}{b} \frac{1}{\phi_i} - \frac{H_i q_i 1}{a} \frac{1}{\phi_i} \right). \hspace{1cm} (5.5.6)
\end{align*}
\]
(It is clear that \( \hat{H}_{\text{DFP}} \) corresponds to \( \theta = 0 \) and \( \hat{H}_{\text{BFGS}} \) corresponds to \( \theta = 1 \)).
Theorem 5.5.1: *

Given any symmetric and positive definite matrix $H$, define the condition number of the matrix $A$, $k(A)$ as before and quantities $a$, $b$, $c$ as:

$$a = q^T H q, \quad b = p^T q, \quad c = p^T H^{-1} p$$

then

$$k(H_{DFP}) \neq k(H_{DFP}')$$

if and only if

$$\left[ \sqrt{\phi} > 1, \frac{c}{\sqrt{\phi}} > b \right] \quad \text{and} \quad \left[ \sqrt{\phi} < 1, \frac{c}{\sqrt{\phi}} < b < a \right].$$

Proof:

The condition numbers $k_1 = k(H_{DFP})$ and $k_2 = k(H_{DFP}')$ are explicitly expressed in terms of $a$, $b$ and $c$, using the relations:

$$\begin{cases} \frac{b}{a} = \lambda_1 \lambda_2 \quad \text{and} \quad a \lambda_1 \lambda_2 - b (\lambda_1 + \lambda_2) + c = 0 \end{cases} \quad (5.5.7)$$

Therefore

$$\lambda_{1,2} = \frac{a(b+c) \pm \sqrt{a^2 (b+c)^2 - 4ab}}{2ab} \quad (5.5.8)$$

If the transformation:

$$a = a^{\phi^2} \quad \text{and} \quad b = b^{\phi}$$

is made in relations (5.5.8) then:

$$\hat{\lambda}_{1,2} = \frac{a(b^{\phi}+c) \pm \sqrt{a^2 (c+b^{\phi})^2 - 4ab^{3}}} {2ab^{\phi}} \quad (5.5.9)$$

From the expressions (5.5.8) and (5.5.9) it is clear that:

* Theorems 5.5.1, 5.5.2 consider the change in the condition numbers when $\hat{a}$, $\hat{b}$ $\alpha$ $a, b$. In the numerical work the actual condition numbers for the two models are used for switching.
\[(\lambda_1 - 1)(\lambda_2 - 1) \leq 0\]

and

\[(\hat{\lambda}_1 - 1)(\hat{\lambda}_2 - 1) \leq 0\]

which implies that:

\[\lambda_1 \geq 1 \geq \lambda_2\]

and

\[\hat{\lambda}_1 \geq 1 \geq \hat{\lambda}_2\]

From the last two relations the condition numbers \(k_1\) and \(k_2\) can be expressed as:

\[
k_1 = \frac{(a(c+b) + \sqrt{a^2(b+c)^2 - 4ab})^2}{4ab^3}
\]

\[
k_2 = \frac{(a(c+b\phi) + \sqrt{a^2(b\phi+c)^2 - 4ab\phi})^2}{4ab^3\phi}
\]

If the difference is taken:

\[
k_1 - k_2 = \frac{1}{4ab^3}\left\{ \left[ a(c+b) + \sqrt{D} \right]^2 - \left[ \frac{a(c+b\phi) + \sqrt{E}}{\sqrt{\phi}} \right]^2 \right\} = (5.5.10)
\]

where \(D = a^2(b+c)^2 - 4ab^3\) and \(E = a^2(b\phi+c)^2 - 4ab^3\phi\).

Then

\[
k_1 - k_2 = P \left\{ ac\left( 1 - \frac{1}{\sqrt{\phi}} \right) + ab\left( 1 - \sqrt{\phi} \right) + \sqrt{D} - \sqrt{\frac{E}{\phi}} \right\} = \frac{1}{4ab^3}\left\{ a(c+b) + \sqrt{D} + \frac{a(c+b\phi) + \sqrt{E}}{\sqrt{\phi}} \right\} = (5.5.11)
\]

where

\[
P = \frac{1}{4ab^3}\left\{ a(c+b) + \sqrt{D} + \frac{a(c+b\phi) + \sqrt{E}}{\sqrt{\phi}} \right\} > 0
\]
\[
\sqrt{D} - \sqrt{\frac{E}{\phi}} = \sqrt{\frac{a^2}{(b+c)^2 - 4ab}} - \sqrt{\frac{a^2(b\phi+c)^2}{\phi} - 4ab^3} = \]

\[
a^2 \left[ (b+c)^2 - \left(\frac{b\phi+c}{\sqrt{\phi}}\right)^2 \right] = \frac{P_1}{\sqrt{D} + \sqrt{\frac{E}{\phi}}} = P_1 \left[ \left(1 - \phi\right) \left(b - \frac{c}{\phi}\right) \right]
\]

where \( P_1 = \frac{a^2}{\sqrt{D} + \sqrt{\frac{E}{\phi}}} \).

If expression (5.5.12) is used in (5.5.11) then:

\[
k_1 - k_2 = P \left\{ \frac{a}{\sqrt{\phi}} \left( \sqrt{\phi} - 1 \right) \left( c - b\sqrt{\phi} \right) + P_1 \left(1 - \phi\right) \left(b^2 - \frac{c^2}{\phi}\right) \right\} = \]

\[
= P \left\{ \frac{a}{\sqrt{\phi}} \left(1 - \sqrt{\phi}\right) \left(b\sqrt{\phi} - c\right) + \frac{P_1}{\phi} \left(1 - \phi\right) \left(b^2\phi - c^2\right) \right\} = \]

\[
= P \left[ (1 - \sqrt{\phi}) (b\sqrt{\phi} - c) \right] \left\{ \frac{a}{\sqrt{\phi}} + \frac{P_1}{\phi} \left(1 + \sqrt{\phi}\right) (b\sqrt{\phi} + c) \right\} \quad (5.5.13)
\]

From relation (5.5.13) it is obvious that if

(i) \( \phi < 1 \) then \( k_1 \geq k_2 \) if and only if \( b \geq \frac{c}{\sqrt{\phi}} \) \quad (5.5.14)

and

(ii) \( \phi > 1 \) then \( k_1 \geq k_2 \) if and only if \( b < \frac{c}{\sqrt{\phi}} \) . \quad (5.5.15)

From relation (5.5.14) and from the inequality \( b^2 < ac \) it is implied that:

\[
\frac{c}{\sqrt{\phi}} \leq b \leq a .
\]
The next Theorem describes the condition which must hold for the condition number of the $H_{BFGS}$ update to be greater than that of the $\hat{H}_{BFGS}$ update.

**Theorem 5.5.2:**

If the condition number of a matrix $A$ is defined as in Theorem 5.5.1, then $k(H_{BFGS}) > k(\hat{H}_{BFGS})$ if and only if:

(i) $\phi < 1$ then $\frac{b}{\sqrt{\phi}} < a$ implies $k(H_{BFGS}) < k(H_{BFGS})$

(ii) $\phi > 1$ then $\frac{b}{\sqrt{\phi}} > a$ implies $k(H_{BFGS}) < k(H_{BFGS})$.

**Proof:**

The $H_{BFGS}$ update corresponds to $\theta = 1$ and from (5.5.5a) the $\lambda_{1,2}$ are the roots of the equation

$$b^2 \lambda^2 - c(a+b)\lambda + bc = 0.$$  $$
\left[\lambda_1, \lambda_2 = \frac{c}{b} \text{ and } \lambda_1 + \lambda_2 = \frac{(a+b)c}{b^2}\right].$$  

Therefore,

$$\lambda_{1,2} = \frac{c(a+b) \pm \sqrt{c^2(a+b)^2 - 4b^3c}}{2b^2}.$$  

From (5.5.6) the corresponding $\hat{\lambda}_{1,2}$ are given by the expression:

$$\hat{\lambda}_{1,2} = \frac{c(a+b) \pm \sqrt{c^2(a+b)^2 - 4b^3c}}{2b^2 \phi} \quad (5.5.16)$$  

Again $\lambda_1 > 1 > \lambda_2$ and $\hat{\lambda}_1 > 1 > \hat{\lambda}_2$ and if the following expressions are defined:
\[
f_1 = \frac{(b+a)c}{2b^2}, \quad f_2 = \frac{c}{b}
\]

\[
f_1(\phi) = \frac{(b+a\phi)c}{2b^2\phi}, \quad f_2(\phi) = \frac{c}{b\phi}
\]

then:

\[
k_1 = k(H_{\text{BFGS}}) = \frac{f_2}{(f_1 - (f_2^2 - f_2^3)^{1/2})^2}
\]

and

\[
k_2 = k(\hat{H}_{\text{BFGS}}) = \frac{f_2(\phi)}{(f_1(\phi) - (f_2(\phi) - f_2(\phi))^3)^{1/2}}
\]

Following similar methods as in Theorem 5.5.1, it can be seen that:

(i) if \( \phi \leq 1 \) then \( k_1 \geq k_2 \) if and only if \( \frac{b}{\sqrt{\phi}} \leq a \)

(ii) if \( \phi > 1 \) then \( k_1 \geq k_2 \) if and only if \( a < \frac{b}{\sqrt{\phi}} \)

It can be noted that

\[
a = q^T H q = (\Delta g)^T H \Delta g \quad \text{and} \quad b = (\Delta g)^T (\Delta x) = (\Delta g)^T \kappa g H g
\]

and

\[
c = (\Delta x)^T H^{-1}(\Delta x) = \kappa^2 g^T H g
\]

therefore all three quantities \( a, b \) and \( c \) are readily available in every iteration.

If similar techniques to the ones used in Theorem (5.5.1) and Theorem (5.5.2) are used, then it can be shown that

\( k(H_{\text{DFP}}) \geq k(H_{\text{BFGS}}) \) if and only if \( c \geq a \). This provides the user with a switching algorithm which chooses either the DFP or BFGS updates depending on whether \( c \) is less than \( a \) or not. This result
however, does not explain conclusively why the BFGS formula performs better than the DFP one. Computational work which has been carried out by Shanno and Phua (1978) and Van der Hoek (1980) has shown an efficient performance of the classical BFGS algorithm. This algorithm is competitive even with sophisticated Quasi-Newton algorithms such as those based on factorizations or projections of search directions.

5.6 **LOCAL CONVERGENCE RESULTS.**

A local convergence analysis for the BFGS family is presented in this section where the update $H$ is given by (5.3.3) and the new point is given by:

$$
\bar{x} = x - Hg = x - B^{-1}g
$$

The basic idea behind the proof is that under certain reasonable conditions on $x_0$, $B_0$ and $f$ (where $x_0$ and $B_0$ are the initial approximations to the minimizer $x^*$ and the second derivative matrix $G(x^*)$ respectively) the errors in the sequence of approximations $\{B_k\}$ to $G(x^*)$ can be shown to be of bounded deterioration. That means these errors, while not ensured to decrease, can increase only in a controlled way (Broyden, Dennis & Moré (1973)).

**Some Auxiliary Results.**

Let $\mathbb{R}^n$ denote the real $n$-dimensional linear space of column vectors and $L(\mathbb{R}^n)$ the linear space of real matrices of order $n$. 
The following minimization problem is considered:

\[
\text{Minimize } f(x) \quad x \in D \subset \mathbb{R}^n
\]

where \( x^* \) denotes the minimum point of the function \( f(x) \).

The unconstrained minimization algorithm which is considered calculates the next point \( x_{k+1} \) from the previous one \( x_k \) as:

\[
x_{k+1} = x_k - H_k g_k \quad k = 0, 1, 2, \ldots
\]

with the update matrices \( H_k \) generated by Broyden's family from the \( \theta \)-class, where \( \theta \in [0, 1] \):

\[
H_{k+1} = \theta H_{DFP} + (1-\theta) H_{BFGS} =
\]

\[
= \theta \left( H_k - \frac{H_k q_k q_k^T H_k}{q_k^T H_k q_k} + \frac{p_k p_k^T}{q_k^T p_k} \phi_k \right) +
\]

\[
+ (1-\theta) \left( I - \frac{p_k p_k^T}{q_k^T p_k} \right) H_k \left( I - \frac{p_k p_k^T}{q_k^T p_k} \right) + \frac{p_k p_k^T}{q_k^T p_k} \phi_k \right)
\]

where

\[
p_k = x_{k+1} - x_k \quad q_k = \frac{1}{\phi_k} \left[ g(x_{k+1}) - g(x_k) \right]
\]

\[
\rho_k, \sigma_{k+1} \text{ as defined in (5.2.2) and } \phi_k = \left( \frac{1 + f(x_k)}{1 + f(x_{k+1})} \right)^2.
\]

If the following scalars are defined:

\[
a = q_k^T H_k q_k, \quad b = q_k^T p_k \quad \text{and} \quad c = p_k^T \phi_k^{-1} p_k
\]

then

\[
H_{DFP,k+1} = H_k - \frac{H_k q_k q_k^T H_k}{a} + \frac{p_k p_k^T}{b} \phi_k
\]
\[ H_{\text{BFGS},k+1} = \left( I - \frac{p_k q_k^T}{b} \right) H_k \left( I - \frac{p_k q_k^T}{b} \right)^T + \frac{p_k p_k^T}{b} \phi_k . \quad (5.6.3) \]

The following results will be needed in the convergence analysis.

**Lemma 5.6.1:**

Assume that \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is twice differentiable in the open convex set \( D \) and suppose that for some \( x^* \) in \( D \) and \( p > 0, \ k > 0, \ g(x^*) = 0, \) the following inequality holds:

\[ \| G(x) - G(x^*) \| \leq k \| x - x^* \|^p . \quad (5.6.4) \]

Then, for every \( u \) and \( v \) in \( D \):

\[ \| g(u) - g(v) - G(x^*)(u-v) \| \leq k \max \left( \| u - x^* \|^p, \| v - x^* \|^p \right) \| u - v \| \quad (5.6.5) \]

where \( g(x) = \nabla f(x) \) is the gradient vector of \( f \) at \( x \). Moreover, if \( G(x^*) \) is invertible, there is an \( \varepsilon > 0 \) and a \( \rho > 0 \) such that:

\[ \max \left( \| u - x^* \|, \| v - x^* \| \right) \leq \varepsilon \]

implies \( u \) and \( v \) belong to \( D \) and

\[ \frac{1}{\rho} \| v - u \| \leq \| g(v) - g(u) \| \leq \rho \| v - u \| \quad (5.6.6) \]

Inequality (5.6.5) follows from a standard result (Ortega & Rheinboldt, 1970, p.70) while (5.6.6) is a consequence of the continuity and non-singularity of \( G(x) \) at \( x^* \). It should be noted that the result quoted from Ortega & Rheinboldt (1970) assumes that \( g(x) \) is Gateaux differentiable on \( D \), but by using the mean value
Theorem, it is easy to show that Lemma 5.6.1 (with a larger constant \( k \) in (5.6.5)) holds if \( g(x) \) differentiable means that the second derivatives matrix \( G(x) \) exists. The following Lemma is needed next:

**Lemma 5.6.2:**

Let \( x \) and \( y \) be any two vectors in \( \mathbb{R}^n \) and \( A \) be an \( n \times n \) real symmetric positive definite matrix. The following inequality is true:

\[
(x^T y)^2 \leq (x^T Ax)(y^T A^{-1} y)
\]

**Proof:**

Since \( A \) is symmetric and positive definite there exists a symmetric and positive definite matrix \( B \) such that \( A = B^2 \).

Therefore,

\[
(x^T y)^2 = (x^T BB^{-1} y)^2 = \left( (Bx)^T (B^{-1} y) \right)^2 \leq (x^T BBx)(y^T B^{-1} B^{-1} y) = (x^T Ax)(y^T A^{-1} y)
\]

(by the Cauchy-Schwartz inequality).

An immediate application of this lemma gives the relation:

\[
(q^T p)^2 = b^2 \leq ac = (q^T Hq)(p^T H^{-1} p)
\]

The main convergence Theorem of this section follows.

**Theorem 5.6.1:**

Let \( f : \mathbb{R}^n \to \mathbb{R} \) be twice differentiable on the open convex set \( D \) and assume that for some \( x^* \) in \( D \) and \( p > 0 \) inequality (5.6.4)
holds. \( g(x^*) = 0 \) and \( G(x^*) \) is non-singular). Then for every \( r \in (0, 1) \) and \( \varepsilon > 0 \), there exists \( \varepsilon(r) > 0 \), \( \delta(r) > 0 \) such that if
\[
\|x_o - x^*\| < \varepsilon \text{ and } \|B_o - G(x^*)\| < \delta
\]
then the sequence
\[
x_{k+1} = x_k - B_k^{-1} g_k = x_k - H_k g_k,
\]
as described in the previous sections, is well defined and converges to \( x^* \). Furthermore,
\[
\|x_{k+1} - x^*\| \leq r \|x_k - x^*\| \quad \text{for all } k \geq 0
\]
and the sequences \( \{\|B_k\|\} \) and \( \{\|H_k\|\} \) are uniformly bounded.

Proof:
To simplify the notation, \( G(x^*) \) will be denoted by \( Q \) in this proof. It is also assumed that
\[
\|Q^{-1}\| \leq m \quad \text{and} \quad \|Q\| \leq M.
\]
Let an arbitrary \( r \in (0, 1) \) and \( \varepsilon_0 > 0 \) be given, and assume that \( \varepsilon = \varepsilon(r) \) and \( \delta = \delta(r) \) are chosen in such a way that the following inequality holds:
\[
m(1+r)(\varepsilon^p + 2\delta) \leq r. \quad (5.6.7)
\]
Suppose that
\[
\|B_o - Q\| < \delta < 2\delta \text{ and } \|x_o - x^*\| < \varepsilon.
\]
Inequality (5.6.7) yields:
2m(1+r)\delta \leq r, \text{ therefore } 2m\delta < 1 \quad (5.6.8)

From the Banach Perturbation Lemma (Ortega & Rheinboldt, 1970 p.45), it follows that:

\[ \|B^{-1}_0\| = \|B^{-1}_0\| \leq m(1+r) \]

Lemma (5.6.1) implies that:

\[ \|x_1 - x^*\| \leq \|B^{-1}_0\| \left[ \|g_0 - g^* - Q(x_0 - x^*)\| + \|Q - B_0\| \|x_0 - x^*\| \right] \]

\[ \leq m(1+r) \left( e^p + 2\delta \right) \|x_0 - x^*\| \quad (5.6.9) \]

From (5.6.7) and (5.6.9) it follows that:

\[ \|x_1 - x^*\| \leq \|x_0 - x^*\| \leq \|x_0 - x^*\| + r \epsilon < \epsilon \]

and thus \( x_1 \in D \).

The proof of the theorem will be completed with the following inductive argument. Assume

\[ \|B_k - Q\| \leq 2\delta \]

(which implies that \( \|B_k\| \leq M + 2\delta \)) and

\[ \|x_{k+1} - x^*\| \leq r \|x_k - x^*\| \quad \text{for } k = 0, 1, \ldots, n-1. \]

Then using the Banach Perturbation Lemma once more:

\[ \|B_k\| = \|B^{-1}_k\| \leq m(1+r) \]

and by letting \( P = B_{k+1} B^{-1}_k \):
If the BFGS update is considered from the $\theta$-class ($\theta = 1$) then:

$$B_{k+1} = B_k - \frac{B_k P_k B_k^T}{P_k T B_k P_k} + \frac{P_k T}{q_k P_k} \phi_k,$$

and the eigenvalues of $B_{k+1}^{-1}$ are $\hat{\lambda}_1$ and $\hat{\lambda}_2$ and one (n-2) times.

Since

$$B_{k+1}^{-1} B_k^{-1} \text{ and } B_k^{-1} B_{k+1}^{-1}$$

are similar therefore they have identical eigenvalues.

Using the inequality:

$$\|AB\|_F < \min \left\{ \|A\|_F, \|B\|_F, \|A\|_F, \|B\|_F \right\}$$

then

$$\|B_{k+1}^{-1} B_k^{-1} - I\|_F < \left\| B_k^{-1} \left( B_k B_{k+1} B_k^{-1} - I \right) B_k^{-1} \right\|_F \leq$$

$$\leq \|B_k^{-1}\| \|B_k^{-1}\| \|B_k^{-1} B_{k+1} B_k^{-1} - I\|_F.$$

* ($\hat{\lambda}_1, \hat{\lambda}_2$ given by 5.5.16)
Since $B_k^{-\frac{1}{2}} B_{k+1} B_k^{-\frac{1}{2}}$ is a symmetric matrix all its eigenvalues are real and

$$\| B_k^{\frac{1}{2}} B_{k+1} B_k^{\frac{1}{2}} - I \|_F \leq \left\{ \text{trace} \left( B_k^{-\frac{1}{2}} B_{k+1} B_k^{-\frac{1}{2}} - I \right) \right\}^{\frac{1}{2}} = \left\{ \sum_{i=1}^{n} (\lambda_i - 1)^2 \right\}^{\frac{1}{2}}.$$  

From the fact that $(n-2)$ $\lambda_i$'s are one, then follows:

$$\| B_k^{\frac{1}{2}} B_{k+1} B_k^{\frac{1}{2}} - I \|_F \leq \frac{a\phi_k}{b} \|egin{bmatrix} (M+2) m (1+r) \end{bmatrix}^{\frac{1}{2}} \right\| \quad (5.6.10)$$

Since $a = q_k^T H_k q_k$, $b = q_k^T p_k$, $(5.6.10)$ becomes:

$$\left\{ \frac{\|H_k\|}{\varepsilon_0} \frac{\|q_k\|^2}{\phi_k} \right\} \|egin{bmatrix} q_k^T H_k q_k \phi_k \\ q_k^T p_k \end{bmatrix} \right\| \leq \left\{ \frac{\|q_k^T H_k q_k \phi_k\|}{\|q_k^T p_k\|} \right\} W \leq \left\{ \frac{\|q_k^T H_k q_k \phi_k\|}{\|q_k^T p_k\|} \right\} W \quad (5.6.11)$$

Therefore for the case where the BFGS update is used:

$$\| B_k^{\frac{1}{2}} B_{k+1} B_k^{\frac{1}{2}} - I \|_F \leq \frac{4Wm(1+r)\rho^3 \varepsilon^2}{\varepsilon_0} r^{2k} \phi_k \quad a_1$$

Since it has been assumed that $\|B_k - Q\| \leq 2\delta$ then:

$$\| B_k^{\frac{1}{2}} B_{k+1} B_k^{\frac{1}{2}} - I \|_F \leq \frac{4Wm(1+r)\rho^3 \varepsilon^2}{\varepsilon_0} r^{2k} \phi_k \leq M a_1 r^{2k} \phi_k \leq M a_1 r^{2k} +$$

$$+ (a_1 r^{2k} + 1)( \|B_k - Q\| \right) \quad (5.6.12)$$
Inequality (5.6.12) can be rewritten as:

\[ \|B_{k+1} - Q\| - \|B_k - Q\| \leq (2\delta + M) a_1 r^{2k} . \]

Summing both sides from \( k = 0 \) to \( n-1 \) the previous inequality implies:

\[ \|B_n - Q\| - \|B_0 - Q\| \leq \frac{(2\delta + M) a_1}{(1+r)^2} \]  \hspace{1cm} (5.6.13)

If \( \delta = \delta(r) \) is selected so the inequality

\[ \frac{(2\delta + M) a_1}{(1-r)^2} \leq \delta \]  \hspace{1cm} (5.6.14)

holds then (5.6.13) implies that

\[ \|B_n - Q\| \leq 2\delta . \]

From the Banach Perturbation Lemma and from Lemma (5.6.1) the relation \( S \)

\[ \|B_n^{-1}\| = \|H_n\| \leq m(1+r) \]

and

\[ \|x_{n+1} - x^*\| \leq \|H_n\| \left\{ \|g_n - g^* - Q(x_n - x^*)\| + \|B_n - Q\| \|x_n - x^*\| \right\} \leq \]

\[ \leq m(1+r) \left( \varepsilon^p + 2\delta \right) \|x_n - x^*\| \leq r \|x_n - x^*\| \] (from (5.6.7)).

From the proof of the theorem the following remarks can be deduced:

(1) Only the property \( \phi_k \in [0, 1] \) was used in the proof of Theorem (5.6.1). This means that there is still considerable freedom in selecting the update \( H \) from the \( \theta \)-class of BFGS algorithm.
Notice the $B_k$'s and the $H_k$'s of the DFP's update are of "bounded deterioration".

(2) It can be seen from the above proof that if instead of $b \geq \varepsilon_o$ the condition $b \geq r^k \varepsilon_o$ is imposed then the convergence result is still valid. The only modification which should be made is that the (5.6.14) inequality should read

$$\frac{(2\delta + M) a_1}{(1-r)} < \delta.$$  

5.7 **COMPUTATIONAL EXPERIMENTS.**

Several standard test functions were used to test the ability of the updating formulae discussed in the previous chapter to solve badly scaled problems and problems of high dimensionality. A description of the structure of the performed experiments and the selection of the appropriate test problems are given in this section.

The numerical experiments were performed on a PRIME interactive computer using a pilot program written in Fortran (see Kuester, and Mize, 1973). The main program calls a subroutine for the line search along each direction. The line search is the bracketing process followed by cubic interpolation used in Fletcher and Reeves (1963). The test problems used are given in the Appendix. Most of them are taken from the literature. The new problems are generated by varying parameters which influence the dimensionality of the test functions.

Although a variety of different stopping criteria were tested it was found that the same results were obtained for all the
algorithms and functions that were used. Consequently the results quoted are based on the stopping condition:

\[ \| e_k \| < 10^{-6} \]

in conformance with the stopping criteria used in the NAG subroutine E04DDF. The cubic line search also stops if the distance of two successively generated points along the search direction is less than a preset parameter.

The performance indicators employed were the number of function and gradient evaluations (Index of Computational Labour, ICL) and the number of iterations required to solve a test problem. The main indicator used was the ICL. The CPU-time was not considered as an indicator because of the inaccuracy of the multiprogramming facilities.

Since the test problems examined vary considerably in complexity, two different sets of test problems were considered. The first consists of functions with low dimensionality \( n = 2, 4, 8 \) and the second includes functions of high dimensionality \( n = 16, 24, 48 \). The two different sets of test problems were considered to bring out any possible advantages of the new model over the standard one. A non-standard test problem, problem 5, see Rosenbrock and Storey (1966) was also tried to show the behaviour of the algorithms on more realistic problems.

The first table included at the end of this section, gives the numerical results for both sets of the test problems considered. The first and second columns of the first table give the problem and the dimensionality used. The next columns give
the number of iterations and the ICL required for termination for the new model, the standard, the EØ4DDF NAG subroutine and the hybrid algorithm respectively.

The second table presents the percentage of more(+) or fewer (-) function and gradient evaluations required in comparison with the standard BFGS formula. Finally in the same table the comparison of the overall performance of the proposed model is given for the two sets of test problems used.

The third table presents the performance of the algorithms on the problems of highest dimensionality possible on the PRIME computer using V.M.M. methods. The table shows that some methods failed to converge in the prescribed time (P3 stream).

The first four figures which follow the three tables show how the efficiency of the new model increases with the dimensionality. The two models behave the same at two dimensions but the difference becomes bigger as the dimension of the problem increases.

The next four figures give an image of the convergence of three different models (new, quadratic and hybrid model) and a NAG subroutine, when applied to high dimensionality problems.
<table>
<thead>
<tr>
<th>DIM</th>
<th>NEW MODEL</th>
<th>STANDARD</th>
<th>EO4DDF</th>
<th>HYBRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>24</td>
<td>213</td>
<td>21</td>
<td>192</td>
</tr>
<tr>
<td>4</td>
<td>33</td>
<td>485</td>
<td>33</td>
<td>475</td>
</tr>
<tr>
<td>8</td>
<td>39</td>
<td>954</td>
<td>37</td>
<td>891</td>
</tr>
<tr>
<td>16</td>
<td>57</td>
<td>2499</td>
<td>57</td>
<td>2516</td>
</tr>
<tr>
<td>24</td>
<td>69</td>
<td>4100</td>
<td>73</td>
<td>4525</td>
</tr>
<tr>
<td>48</td>
<td>101</td>
<td>11417</td>
<td>107</td>
<td>12495</td>
</tr>
<tr>
<td>4</td>
<td>26</td>
<td>410</td>
<td>30</td>
<td>440</td>
</tr>
<tr>
<td>8</td>
<td>34</td>
<td>765</td>
<td>39</td>
<td>900</td>
</tr>
<tr>
<td>16</td>
<td>37</td>
<td>1462</td>
<td>55</td>
<td>2261</td>
</tr>
<tr>
<td>24</td>
<td>49</td>
<td>2650</td>
<td>98</td>
<td>6200</td>
</tr>
<tr>
<td>48</td>
<td>78</td>
<td>7987</td>
<td>110</td>
<td>11417</td>
</tr>
<tr>
<td>4</td>
<td>30</td>
<td>565</td>
<td>28</td>
<td>505</td>
</tr>
<tr>
<td>8</td>
<td>33</td>
<td>972</td>
<td>42</td>
<td>1278</td>
</tr>
<tr>
<td>16</td>
<td>34</td>
<td>1700</td>
<td>54</td>
<td>2907</td>
</tr>
<tr>
<td>24</td>
<td>38</td>
<td>2825</td>
<td>62</td>
<td>4950</td>
</tr>
<tr>
<td>48</td>
<td>60</td>
<td>9751</td>
<td>87</td>
<td>13671</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>295</td>
<td>18</td>
<td>275</td>
</tr>
<tr>
<td>8</td>
<td>23</td>
<td>549</td>
<td>26</td>
<td>729</td>
</tr>
<tr>
<td>16</td>
<td>34</td>
<td>1343</td>
<td>35</td>
<td>1445</td>
</tr>
<tr>
<td>24</td>
<td>41</td>
<td>2250</td>
<td>44</td>
<td>2575</td>
</tr>
<tr>
<td>48</td>
<td>65</td>
<td>6762</td>
<td>68</td>
<td>7380</td>
</tr>
<tr>
<td>2</td>
<td>22</td>
<td>219</td>
<td>20</td>
<td>225</td>
</tr>
<tr>
<td>10</td>
<td>45</td>
<td>1474</td>
<td>45</td>
<td>1628</td>
</tr>
<tr>
<td>20</td>
<td>58</td>
<td>2982</td>
<td>55</td>
<td>3171</td>
</tr>
<tr>
<td>40</td>
<td>68</td>
<td>6068</td>
<td>81</td>
<td>6970</td>
</tr>
</tbody>
</table>

**ST. POINT**

**KINETIC PROBLEM:**

1. \((1073, .5)\)  18  165  27  243  159  22  151
2. \((873, .5)\)  15  159  13  159  129  20  143
3. \((773, .5)\)  14  159  17  198  135  23  155
### TABLE 2

<table>
<thead>
<tr>
<th>PROBLEM</th>
<th>DIM</th>
<th>PROP.MOD - EO4DDF</th>
<th>PROP.MOD - ST.MOD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
<td>+28.2</td>
<td>+9.8</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>+16.6</td>
<td>+2.1</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>+0.1</td>
<td>+6.6</td>
</tr>
<tr>
<td>ROSENBROCK</td>
<td>16</td>
<td>-25.8</td>
<td>-1.0</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>-29.0</td>
<td>-9.4</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>-28.0</td>
<td>-8.6</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>-37.8</td>
<td>-6.8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>-47.2</td>
<td>-15.0</td>
</tr>
<tr>
<td>WOOD</td>
<td>16</td>
<td>-76.8</td>
<td>-35.3</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>-79.9</td>
<td>-57.2</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>-71.9</td>
<td>-30.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>+56.6</td>
<td>+10.6</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>+20.4</td>
<td>-23.9</td>
</tr>
<tr>
<td>MIELE &amp; CANTRELL</td>
<td>16</td>
<td>+58.1</td>
<td>-41.5</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>+0.9</td>
<td>-42.9</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>+35.2</td>
<td>-28.7</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>+32.2</td>
<td>+6.8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>+13.1</td>
<td>-24.7</td>
</tr>
<tr>
<td>POWELL</td>
<td>16</td>
<td>+15.2</td>
<td>-7.1</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>-23.1</td>
<td>-12.6</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>-21.6</td>
<td>-8.0</td>
</tr>
<tr>
<td>KINETIC</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(st.point): (1073, .5)</td>
<td>+3.6</td>
<td>-32.1</td>
<td></td>
</tr>
<tr>
<td>(st.point): (873, .5)</td>
<td>+18.9</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>(st.point): (773, .5)</td>
<td>+15.1</td>
<td>-19.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>+27.3</td>
<td>-2.6</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>-37.9</td>
<td>-9.4</td>
</tr>
<tr>
<td>CUBIC</td>
<td>20</td>
<td>-60.5</td>
<td>-5.9</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>-52.1</td>
<td>-12.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>PR.MOD. -</th>
<th>PR.MOD. -</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST.MOD.</td>
<td>EO4DDF</td>
<td></td>
</tr>
<tr>
<td>Lower Dimensionality Set (Dimensions: 2,4,8):</td>
<td>-20.1</td>
<td>+9.9</td>
</tr>
<tr>
<td>Higher Dimensionality Set (Dimensions: 16,24,48):</td>
<td>-24.3</td>
<td>-20.5</td>
</tr>
</tbody>
</table>
### TABLE 3

**DIMENSION : 100**

<table>
<thead>
<tr>
<th></th>
<th>NEW MODEL</th>
<th>STANDARD</th>
<th>EO4DDF</th>
<th>HYBRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROS.</td>
<td>119</td>
<td>29290</td>
<td>165 39289*</td>
<td>75548*</td>
</tr>
<tr>
<td>MIELE</td>
<td>111</td>
<td>24543</td>
<td>115 29391</td>
<td>131099</td>
</tr>
<tr>
<td>WOOD</td>
<td>147</td>
<td>30401</td>
<td>158 32421*</td>
<td>134636</td>
</tr>
<tr>
<td>POWELL</td>
<td>118</td>
<td>25048</td>
<td>119 25250</td>
<td>62527</td>
</tr>
<tr>
<td>CUBIC</td>
<td>152</td>
<td>32017</td>
<td>140 29290</td>
<td>71003</td>
</tr>
</tbody>
</table>

* : Failed to converge in the prescribed time.
CUBIC
N = 10

The graph depicts the behavior of a cubic function with a logarithmic scale on the x-axis and the number of iterations on the y-axis.

Key points:
- The function shows a rapid decrease in the number of iterations as the logarithm of the difference between function values increases.
- The graph is labeled 'CUBIC N = 10', indicating the type of function and its parameters.

Mathematical expression:
\[ f(x) - f(x_{i-1}) \]
CUBIC
N = 20

|f(x_i) - f(x_{i-1})|^{\log 10}
CUBIC
$N = 40$
WOOD

$N = 48$

New model $\circ$
Standard model $\circ$
EO4DDF $\times$
Hybrid model $+$

$\frac{\text{Funct. Eval.} \times 10^3}{30}$

$|f(x_i) - f(x_{i-1})|$
\[ |f(x_i) - f(x_{i-1})| \]
ROSENROCK

N = 48

\[ |f(x_i) - f(x_{i-1})| \]
POWELL

N = 48

$|f(x_1) - f(x_{1-1})|$
5.8 DISCUSSION.

In this chapter a new model for the Variable Metric Method is proposed and tested using the BFGS updating formula. Several standard test functions and a more realistic highly nonlinear function were used. The BFGS updating formula was preferred to the DFP because of its better computational performance. (See, e.g. Van der Hoek (1980)).

The motivation for the new model is the same as that reported in previous papers, (Storey (1978) and Tassopoulos and Storey (to appear)). Finite termination can be proved for Variable Metric Methods using the proposed model when applied to functions of the form of (5.2.1), in a similar way as in Tassopoulos and Storey (to appear).

A switching algorithm was introduced which chooses one of the two models depending on their condition numbers.

Since the switching algorithm chooses the update with the smallest condition number, it should be better than using only the standard or the new model update. The numerical results show some considerable improvement when the switching algorithm is used especially when the dimensionality of the problem increases. Therefore some improvement results from the hybrid method used.

The final conclusion that can be drawn from the numerical results is that the proposed model behaves as efficiently as the currently used model in general and is more satisfactory for problems of higher dimensionality.
APPENDIX.

1. ROSENBROCKS Function (Generalised form)

\[ F(x) = \sum_{i=1}^{N/2} 100 \left( x_{2i}^2 - x_{2i-1}^2 \right)^2 + (1-x_{2i-1})^2 \]

starting point: \((-1.2, 1, \ldots)\), \(N = 2, 4, 8, 16, 24, 48\).

2. WOODS Function (Generalised form)

\[ F(x) = \sum_{i=1}^{N/4} 100 \left( x_{4i-2}^2 - x_{4i-3}^2 \right)^2 + (1-x_{4i-3})^2 + 90(x_{4i}^2 - x_{4i-1}^2)^2 + (1-x_{4i-1})^2 + 10.1 \left( x_{4i-2} - x_{4i-1} \right)^2 + (x_{4i-1})^2 + 19.8 \left( x_{4i-2} - x_{4i-1} \right)^2 (x_{4i-1}) \]

starting point: \((-3, -1, -3, -1, \ldots)\), \(N = 4, 8, 16, 24, 48\).

3. MIELE & CANTRELL (Generalised form)

\[ F(x) = \sum_{i=1}^{N/4} \left[ \exp(x_{4i-3} - x_{4i-2}) \right]^2 + 100(x_{4i-2} - x_{4i-1})^6 + \left[ \tan(x_{4i-1} - x_{4i}) \right]^4 + x_{4i-3}^8 + (x_{4i-1})^2 \]

starting point: \((1, 2, 2, 2, \ldots)\), \(N = 4, 8, 16, 24, 48\).

4. POWELL (Generalised form)

\[ F(x) = \sum_{i=1}^{N/4} \left[ (x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^5 \right] \]

starting point: \((3, -1, 0, 1, \ldots)\), \(N = 4, 8, 16, 24, 48\).
5. KINETIC (Rosenbrock and Storey (1966) p.102).

\[
F(x) = \frac{k_4 k_1 \exp(-k_5 x_1)}{k-k_4} \left( \frac{1-\exp\left(-(k_4-k_5)x_1\right)}{(k_4-k_5)} \right)
\]

where \( k_i = C_i \exp\left( -\frac{E_i}{R \left( \frac{1}{x_2} - \frac{1}{638} \right)} \right) \),

\( R = 1.9865 \),

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_i )</td>
<td>1.02</td>
<td>0.93</td>
<td>0.386</td>
<td>3.28</td>
<td>0.084</td>
</tr>
<tr>
<td>( E_i )</td>
<td>16,000</td>
<td>14,000</td>
<td>15,000</td>
<td>10,000</td>
<td>15,000</td>
</tr>
</tbody>
</table>

and \( k = k_1 + k_2 + k_3 \)

starting points: \((1073, 0.5)\)
\((873, 0.5)\)
\((773, 0.5)\)

Proof of Remark (1):

The assumption of stability of the BFGS method implies the inequality:

\[
F(x_k) > F(x_{k+1}) \quad \text{(A.1)}
\]

and using the expression (5.2.1) the above inequality can be rewritten as:
\[
\frac{m(x_k)}{1 + m(x_k)} > \frac{m(x_{k+1})}{1 + m(x_{k+1})}. \quad (A.2)
\]

The following cases are considered:

(i) \( m(x_k) < -1 \) then \( m(x_k) + 1 < 0 \)

\[\quad \frac{m(x_k)}{1 + m(x_k)} > 0 . \] Therefore \( m(x_k) > m(x_{k+1}) \).

(ii) \(-1 < m(x_k) < 0\) then \( m(x_k) + 1 > 0 \).

So: \( m(x_k) > m(x_{k+1}) \).

(iii) \( m(x_k) > 0 \) then \( m(x_k) > m(x_{k+1}) \).

Using the above results and the formula which gives \( \phi_i \) ((5.2.2))

it can easily be implied that

\[ 1 + m(x_{k+1}) < 1 + m(x_k) \]

or

\[ \phi_k = \left( \frac{1 + m(x_{k+1})}{1 + m(x_k)} \right)^2 < 1 . \]
**Proof of Corollary 5.2.1:**

The step from \( x_k \) to \( x_{k+1} \) is in the direction \( d_k = -H_k g_k \). The stepsize is chosen to minimize \( f \) in that direction, that is to minimize \( f(x_k + ad_k) \) with respect to \( a \). Hence

\[
\left. \frac{d \ f(x_k + ad_k)}{da} \right|_{a=a_k} = 0.
\]

This implies \( d_k^T g_{k+1} = 0 \). For a sufficiently small step \( a > 0 \) and for a function \( f \) having continuous second partial derivatives, the first order terms in the Taylor series give:

\[
f(x_k + ad_k) = f(x_k) + ad_k^T g_k.
\]

Since \( a > 0 \), the previous relation implies that:

\[
f(x_k + ad_k) < f(x_k) \quad \text{if and only if} \quad d_k^T g_k < 0.
\]

The direction \( d_k \) is downhill if and only if

\[- g_k^T H_k g_k < 0.
\]

Therefore if \( H_k \) is a positive definite matrix and \( g_k \neq 0 \) there exists a \( a_k > 0 \) such that:

\[
f(x_{k+1}) = f(x_k + a_k d_k) < f(x_k).
\]

Evaluating the quantities \( \{\phi_i, \rho_i, \sigma_{i+1}\} \) with already known quantities.

If \( F(x) \) is given by (5.2.1) and \( \phi_i, \rho_i, \sigma_{i+1} \) are given by (5.2.2) then:

\[
m(x) = \frac{F(x)}{\varepsilon - F(x)} \quad \text{(A.1)}
\]
\[
\phi_i = \frac{F(x_{i+1})}{1 + \frac{F(x_i)}{\varepsilon - F(x_i)}} = \frac{\varepsilon - F(x_i)}{\varepsilon - F(x_i+1)} \quad (A.2)
\]

\[
\sigma_{i+1} = \frac{\varepsilon}{(1 + m(x_{i+1}))^2} = \frac{(\varepsilon - F(x_{i+1}))^2}{\varepsilon} \quad (A.3)
\]

\[
\rho_i = \frac{\sigma_{i+1}}{\phi_i} \quad (A.4)
\]

Using similar algebraic manipulations as in the Appendix of Chapter IV, the scalar \(\varepsilon\) can be calculated and from equations (A.2), (A.3) and (A.4) the respective values of \(\phi_i, \sigma_{i+1}, \rho_i\) can be computed.

**Interlocking Eigenvalues Theorem (due to Loewner):**

Let the symmetric \((n \times n)\) matrix \(A\) have eigenvalues

\[
\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n.
\]

Let \(a\) be any vector in \(\mathbb{R}^n\) and denote the eigenvalues of the matrix \(A + aa^T\) by

\[
\mu_1 \leq \mu_2 \leq \ldots \leq \mu_n.
\]

Then

\[
\lambda_1 \leq \mu_1 \leq \lambda_2 \leq \mu_2 \leq \ldots \leq \mu_n.
\]

Proof of the conditions of Theorem (5.5.2).

\[ k_1 - k_2 = \frac{c(a+b) + \sqrt{D}}{c(a+b) - \sqrt{D}} - \frac{c(a+b) + \sqrt{E}}{c(a+b) - \sqrt{E}} \]

\[ = \frac{1}{4b^3c} \left\{ \left( c(a+b) + \sqrt{D} \right)^2 - \left( \frac{c(a+b) + \sqrt{E}}{\sqrt{\phi}} \right)^2 \right\} \]

\[ = \frac{1}{4b^3c} \left\{ \left( c(a+b) + \sqrt{D} \right)^2 + c \left( \frac{a\sqrt{\phi} + b}{\sqrt{\phi}} \right) \left( c(a+b) + \sqrt{D} - c \left( \frac{a\sqrt{\phi} + b}{\sqrt{\phi}} \right) - \sqrt{E} \right) \right\} \]

\[ = \frac{1}{4b^3c} \left\{ p_c \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) + \sqrt{D} - \frac{\sqrt{E}}{\sqrt{\phi}} \right\} \]

\[ \text{(I)} \]

(i) \[ \sqrt{D} - \frac{\sqrt{E}}{\sqrt{\phi}} = \sqrt{c^2(a+b)^2 - 4b^3c} - \sqrt{\frac{c^2(a\sqrt{\phi} + b)^2 - 4b^3c}{\phi}} \]

\[ = \frac{c^2(a+b)^2 - 4b^3c - c^2 \left( \frac{a\sqrt{\phi} + b}{\sqrt{\phi}} \right)^2 + 4b^3c}{\sqrt{c^2(a+b)^2 - 4b^3c} + \sqrt{c^2(a\sqrt{\phi} + b)^2 - 4b^3c}} \]

\[ = \frac{1}{P_1} \left\{ p^2 \left( a + b + a\sqrt{\phi} + b \sqrt{\phi} \right) \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) \right\} \]

\[ = \frac{1}{P_2} \left\{ p \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) \right\} \]
Therefore (I) becomes:

\[
\frac{1}{4b^3} c \left[ p \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) + p_2 \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) \right] =
\]

\[
= \left( a - \frac{b}{\sqrt{\phi}} \right) \left( 1 - \sqrt{\phi} \right) \left( \frac{p}{4b^3} + p_2 \right)
\]

(i) \( \phi < 1 \) then \( (1 - \sqrt{\phi}) > 0 \)

so \( a > \frac{b}{\sqrt{\phi}} \).

(ii) \( \phi > 1 \) then \( (1 - \sqrt{\phi}) < 0 \)

so \( a < \frac{b}{\sqrt{\phi}} \).
CHAPTER VI

CONSTRAINED MINIMIZATION TECHNIQUE

USING A NEW VARIABLE METRIC UPDATE
6.1 **INTRODUCTION.**

The nonlinear programming problem to be considered in this chapter is defined in the first chapter as problem (P2) but for convenience is restated here:

\[
\begin{align*}
\text{minimize} & \quad f(x), \quad x \in \mathbb{R}^n \\
\text{subject to} & \quad c_i(x) = 0, \quad i = 1, \ldots, m' \\
& \quad c_i(x) \geq 0, \quad i = m'+1, \ldots, m \\
\end{align*}
\]

(P2)

It is clear that problem (P2) will be an unconstrained optimization problem if \( m = 0 \). For unconstrained optimization problems, Variable Metric Methods generate sequences \( \{x_k\} \), \( \{B_k\} \) which are respectively approximations to the minimizer of the function \( f(x) \) and the Hessian evaluated at this point. At the \( k \)-th step the point \( x_{k+1} \) is obtained from \( x_k \) and \( B_k \) by setting

\[
x_{k+1} = x_k + t_k B_k^{-1} \nabla f(x_k)
\]

where \( t_k \) is a real number determined by some stepsize procedure and \( B_{k+1} \) is chosen to satisfy the relation:

\[
B_{k+1} (x_{k+1} - x_k) = \nabla f(x_{k+1}) - \nabla f(x_k)
\]

which may be regarded as a first divided difference of \( \nabla f \), (Broyden, 1973).

The borrowing of ideas which work well in one method for use in another has led to a blurring of the usual distinction between the various classes of methods. In fact, all the methods are, in some sense, variants of Newton's method on an appropriate equivalent problem.
Therefore, in developing methods for solving constrained optimization problems by reducing them to a series of unconstrained ones, gives us a selection over a wide range of successful methods used in the unconstrained case. It is of great importance to extend successful unconstrained optimization methods to the constrained case. The extension of Variable Metric Methods to the constrained case has been studied by Goldfarb (1969), by employing the concept of the projected-gradient method. Other different approaches to the extension of the related Variable Metric Methods have been investigated by Biggs (1975), Garcia and Mangasarian (1976), Han (1976) and more recently by M.J.D. Powell (1977).

The algorithm proposed by Powell (1977) starts with a point \( x_0 \) and a positive definite matrix \( E_0 \) (usually taken as the unit matrix). It proceeds by solving a linearly constrained quadratic problem and replacing the point \( x_0 \) by an improved point \( x' \). A more detailed description of the method follows in the next section.

This chapter is an attempt to incorporate an updating formula for \( E_k \), based on a nonquadratic model, into the algorithm proposed by Powell (1977) and compare it with the algorithm using the standard updating formula. The nonquadratic model used is a nonlinear scaling of a quadratic function and it is an extension of the idea used in previous chapters for unconstrained optimization. Numerical experience with the proposed update is included and it can be inferred that the algorithm is competitive with those currently available.
6.2 **STATEMENT OF THE ALGORITHM.**

In this section an algorithm is presented for finding a minimizer of the problem (P2). The algorithm constructs a sequence of points \( \{x_k\} \) starting from \( x_0 \), which is an estimate of the solution point \( x^* \) of (P2). This sequence is formed by solving a sequence of linearly constrained quadratic programs which contain an estimate \( E_k \) of the Hessian of the Lagrangian of the problem (P2). For the sake of simplicity the subscripts \( k \) will be dropped and the next points will be represented by barred \( x \)'s.

To be more precise the vector \( p \) which minimizes the function:

\[
Q(x, E) = (\nabla f(x))^T p + \frac{1}{2} p^T E p
\]

is calculated. It is then used as a search direction in the space of the variables, current \( x \) being replaced by the vector:

\[
\bar{x} = x + \alpha p
\]

where \( \alpha \) is a positive scalar whose value depends on the function of one variable:

\[
\bar{L}_1(\alpha) = \bar{L}(x + \alpha p) . \tag{6.2.1}
\]

Then the matrix \( E \) is revised using an update based on a nonquadratic model. This is the BFGS formula used in Algorithm II of Chapter V. A brief description of the rationale for the use of quadratic subproblems is now given.

Denote the Lagrangian function:

\[
L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \text{ by } L(x, \lambda) = f(x) + \lambda^T c(x)
\]
where \( x \in \mathbb{R}^n, \lambda \in \mathbb{R}^m \) and \( f : \mathbb{R}^n \to \mathbb{R} \) is the objective function of the problem (P2) and \( c : \mathbb{R}^n \to \mathbb{R}^m \) is the constrained function of the same problem. It is a well known fact (see Fiacco & McCormick (1968)) that if there exists \((\bar{x}, \bar{\lambda})\) such that:

\[
L(\bar{x}, \lambda) \leq L(x, \lambda) \leq L(x, \bar{\lambda}) \quad \lambda \geq 0
\]  

(6.2.2)

then \( \bar{x} \) solves problem (P2). It turns out that if

\[
E_i = V^2 L(x_i, \lambda_i)
\]

the point \((x_{i+1}, \lambda_{i+1})\) generated by an algorithm employing quadratic subproblems, satisfies (6.2.2) for a quadratic approximation of the Lagrangian around the point \((x_i, \lambda_i)\). If the quadratic approximation to the Lagrangian is considered:

\[
\hat{L}_i(x, \lambda) = L(x_i, \lambda_i) + \nabla f(x_i) + \lambda_i^T c(x_i) + c^T(x_i)(\lambda - \lambda_i)
\]

\[
+ \frac{1}{2} \begin{bmatrix}
V^2 f(x_i) + \lambda_i^T c(x_i), Vc(x_i)
\end{bmatrix}
\]

\[
\begin{bmatrix}
\nabla^T c(x_i)
\end{bmatrix}
\]

This expression can be rewritten as:

\[
\hat{L}_i(x, \lambda) = f(x_i) + \lambda_i^T c(x_i) + (\nabla f(x_i) + \lambda_i Vc(x_i))^T (x - x_i) +
\]

\[
+ c^T(x_i)(\lambda - \lambda_i) + \frac{1}{2} (x - x_i)^T V^2 f(x_i) + \lambda_i^T c(x_i) (x - x_i) +
\]

\[
+ (\lambda - \lambda_i)^T Vc(x_i) (x - x_i),
\]

and after simplifying terms:
It can be observed that $L_i(x, \lambda)$ is precisely the Lagrangian function for the subproblem:

$$\min_{x \in N_i} \left\{ (\nabla f(x_i))^T (x-x_i) + \frac{1}{2} (x-x_i)^T E_i (x-x_i) \right\} = (Q(x,E))$$

and

$$N_i = \left\{ x \mid c(x_i) + (\nabla c(x_i))^T (x-x_i) \leq 0 \right\}$$

with $E_i = \nabla^2 L(x_i, \lambda_i)$.

Hence by the Kuhn-Tucker theory a saddle point of $L_i(x, \lambda)$ solves $Q(x,E)$ and if $E_i$ is positive definite, each solution of $Q(x,E)$ is also a saddle point of $L_i(x, \lambda)$. This relation provides a motivation for generating $\{E_i\}$ which approach

$$\nabla^2 f(x_i) + \lambda_i^T c(x_i)$$

and hence the algorithm tends to saddlepoints of the quadratic approximation to the Lagrangian. The following definition is now introduced.

**Definition (6.2.1):**

A vector $z = (x + \bar{p}, \bar{\lambda})$ in $\mathbb{R}^{n+m}$ is called a $z$-solution of $Q(x,E)$ if $(\bar{p}, \bar{\lambda})$ is a Kuhn-Tucker pair of $Q(x,E)$.

The algorithm can now be stated:
Algorithm (6.2):

Step 1: Start with an estimate $z_0 = (x_0, \lambda_0)$ of a Kuhn-Tucker pair of problem (P2) and an estimate $E_0$ of the Hessian of the Lagrangian, a positive definite matrix (usually the identity matrix).

Step 2: Set $k = 0$.

Step 3: Find a $z$-solution of $Q(x_k, E_k)$ and call this $z$-solution:

$$z_{k+1} = (x_{k+1}, \lambda_{k+1}).$$

If there are more than one such $z$-solutions, then choose the closest one to $z_k$, using any arbitrary but fixed norm in $\mathbb{R}^{n+m}$.

Step 4: If $z_{k+1} = (x_{k+1}, \lambda_{k+1})$ satisfies a prescribed convergence criterion, stop; otherwise go to Step 5.

Step 5: Update $E_{k+1}$ using the BFGS updating scheme, based on a non-quadratic model and used also in Algorithm (II) of the previous chapter for the updating of $E_{k+1}$ then set $k = k+1$ and go to Step 3.

It can be easily observed that Algorithm (P2) is an iterative process of a similar type to a Variable Metric algorithm. As in a Variable Metric algorithm a matrix is updated in each iteration to generate the next point. To ensure convergence the updated matrices are required to be good estimates of the Hessian of the Lagrangian of the problem (P2). In the degenerate case ($m = 0$) the Algorithm (6.2) is just a Variable Metric algorithm, therefore it can be considered as a very natural extension of Variable Metric algorithms to solve constrained optimization problems. By this...
extension a wide variety of updating schemes in the unconstrained optimization literature become available for solving constrained optimization problems. More significantly this extension offers a unified treatment of constrained and unconstrained optimization problems by a Variable Metric approach.

6.3 THE UPDATING OF $E_k$.

A further point of particular interest to be treated in this section, is the incorporation of the non-quadratic based Variable Metric update, discussed in previous chapters in the framework of recursive quadratic programming. In the extension to the constrained case, a Variable Metric method retains the important property that the definition of the search direction $p_k$ depends only on a positive definite matrix $E_k$ of dimension $n$ and on function values and first derivatives that are calculated at the starting point of the iteration $x_k$.

The technique is to let $p_k$ be the vector $p$ that minimizes the function $Q(x,E)$ subject to the linear constraints:

$$
\begin{align*}
   c_i(x_k) + p^T \nabla c_i(x_k) &= 0, \quad i = 1, \ldots, m \\
   c_i(x_k) + p^T \nabla c_i(x_k) &\geq 0, \quad i = m+1, \ldots, m
\end{align*}
$$

which are approximations to the conditions of the problem (P2) at the point $x = x_k + p$, (Garcia and Palomares (1976)). Therefore the calculation of $p_k$ is a convex programming problem.

The description of a Variable Metric algorithm for the constrained case will be completed by specifying a method that
generates the sequence of matrices $E_k$ and a procedure which fixes the step-lengths $\alpha$ in equation (6.3.1). Most of the available methods for generating the positive definite matrices are based on formulae that are used in the unconstrained case.

In the unconstrained case it is a sensible strategy to force every matrix $E_k$ to be positive definite because it ensures that each search direction $p_k$ is downhill. However, the situation is less clear when constraints are present. As is pointed out in Powell (1977), in order to obtain superlinear convergence first derivatives as well as function values of $f(x)$ and $c_i(x)$, $i = 1, \ldots, m$, are required and second derivative information of the Lagrangian function

$$L(x, \lambda) = f(x) - \sum_{i=1}^{m} \lambda_i c_i(x)$$

i.e.

$$E(x, \lambda) = \nabla^2 L(x, \lambda) = G(x) - \sum_{i=1}^{m} \lambda_i \nabla^2 c_i(x). \quad (6.3.1)$$

The second derivative matrix $G(x)$ of the function $f(x)$, is helpful only in the case where the constraints are linear. However, when the constraints are curved then $G(x)$ is helpful only if the second derivatives $\nabla^2 c_i(x)$, $\{i = 1, \ldots, m\}$ are also available, (Powell, (1978)). The second derivative matrix $E(x, \lambda)$ which is needed in the calculations contains the matrices $G(x)$ and $\nabla^2 c_i(x)$, $\{i = 1, \ldots, m\}$ in a combination given in expression (6.3.1). Therefore the second derivatives matrix of the Lagrangian function is more relevant than the matrix $G(x)$ of the objective function $f(x)$. Therefore, the updating of $E_k$ depends also on
estimates of the parameter $\lambda$ which is taken as the vector of the Lagrangian parameters at the solution of the $Q(x, E)$ problem.

As is usual in Variable Metric methods on each iteration $E_{k+1}$ is calculated by using the previous matrix $E_k$ and a correction depending on the differences:

$$p_k = x_{k+1} - x_k$$

and

$$q_k = \phi_k (p_k \nabla_x L(x, \lambda) - \sigma_{k+1} \nabla_x L(x, \lambda))$$

where $\phi_k, p_k, \sigma_k$ are given by the relation (5.2.2). Then the matrix $E_{k+1}$ is defined by the formula

$$E_{k+1} = E_k - \frac{E_k p_k p_k^T E_k}{q_k q_k^T} + \frac{q_k q_k^T}{p_k^T E_k p_k}$$

In the calculations given in the fifth section of the present chapter, the vector $q_k$ is replaced by the vector:

$$n_k = \theta_k q_k + (1-\theta_k) E_k p_k$$

(see Powell (1977)).

The scalar $\theta_k$ is set to a value from the interval $[0, 1]$ that maintains positive definiteness in the same way as in Powell (1977).

In the same paper, it is shown that this method of revising $E_k$ can give superlinear convergence even when the second derivative matrix of the Lagrangian function is indefinite.
6.4 LOCAL CONVERGENCE RESULTS.

This section will be devoted to the establishment of local convergence theorems for the (P2) algorithm. The Algorithm (6.2) proposed in the previous section requires calculations with function values and gradients, hence \( f(x) \) and \( c(x) \) should be at least differentiable. However, the convergence proofs which follow require, amongst other things, twice differentiability of \( f(x) \) and \( c(x) \).

Starting with an initial and possibly infeasible guess \( x_0 \) of the solution and an initial estimate \( \lambda_0 \) of the Kuhn-Tucker multipliers, the Algorithm (6.2) constructs a sequence \( \{(x_i, \lambda_i)\} \) which under suitable conditions converges to a Kuhn-Tucker point \( (\bar{x}, \bar{\lambda}) \) of (P2).

The convergence which is established is a local one and the main conditions required for it are closeness of the starting point to a point \( (\bar{x}, \bar{\lambda}) \) satisfying the second order sufficiency conditions (Fiacco and McCormick, 1968), the linear independence of the gradients of the active constraints and the positivity of the multipliers associated with them.

The quadratic subproblems contained in the Algorithm (6.2) involve a matrix which is an estimate of the Hessian of the Lagrangian of problem (P2). The definition of a \( p \)-th order convergence of a sequence follows. Note that \( \| \cdot \| \) will denote any norm defined on \( \mathbb{R}^n \) throughout this work.

**Definition (6.4.1) (Convergence Rate):**

Let \( \{x_k\} \) be a convergent sequence in \( \mathbb{R}^n \) and \( \bar{x} \) be its limit.
The sequence \( \{x_k\} \) is said to converge to \( \bar{x} \) with order "Q-superlinear" if and only if either \( x_k = \bar{x} \) for all sufficiently large \( k \) or \( x_k \neq \bar{x} \) for \( k \geq k_0 \) and
\[
\lim_{k \to \infty} \frac{||x_{k+1} - \bar{x}||}{||x_k - \bar{x}||} = 0 . \tag{6.4.1}
\]

Almost all theorems on local convergence and on the superlinear rate of convergence developed in Han (1976) apply for Algorithm (6.2) as well. It will now be shown that the update used in Chapter V for the unconstrained minimization problem (P1) satisfies the hypothesis of these results as well. The update considered in Algorithm (6.2) is now restated with the barred quantities denoting the \((k+1)\)st variables:

\[
\overline{E} = E - \frac{Ep \ p^T E}{p^T Ep} + \frac{q \ q^T}{p^T q} \tag{6.4.2}
\]

and
\[
q = \phi(\rho \nabla_x L(\overline{x}, \overline{\lambda}) - \sigma \nabla_x L(x, \lambda)) = \gamma_1 \nabla_x L(\overline{x}, \overline{\lambda}) - \gamma_2 \nabla_x L(x, \lambda) \tag{6.4.3}
\]
\[
p = \overline{x} - x .
\]

The quantities \( \phi, \rho, \sigma \) are defined in (5.2.2).
Next the following lemma will be established prior to stating the key local convergence result. This is similar to Lemma (3.3) of Han (1976) but with the appropriate changes made for the update used in Algorithm (6.2).

Lemma 6.4.1:

If \( f \) and \( c_i \) (\( i = 1, \ldots, m \)) have Lipschitz continuous, \( 2^{nd} \) derivatives with constants \( k_0 \) and \( k_i \) (\( i = 1, \ldots, m \)), then there exists a neighbourhood \( N(x^*, \epsilon) \) and two positive constants \( \hat{k}_1, \hat{k}_2 \), such that for any \( x_1, x_2 \in N(x^*, \epsilon) \) and any \( \lambda \) in \( \mathbb{R}^m \) the following holds:

\[
\begin{align*}
&\| \nabla f(x_1) + \nabla c(x_1) \lambda - \nabla f(x_2) - \nabla c(x_2) \lambda - \nabla^2 L(x^*, \lambda^*) (x_1 - x_2) \| \\
\leq &\left( \hat{k}_1 \max\left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} + \hat{k}_2 \| \lambda - \lambda^* \| \right) \| x_1 - x_2 \| .
\end{align*}
\]

Proof:

By the assumptions there exists a neighbourhood \( N(x^*, \epsilon) \) and positive numbers \( k_i \) (\( i = 0, 1, \ldots, m \)) such that for any \( x \in N(x^*, \epsilon) \):

\[
\| \nabla^2 f(x) - \nabla^2 f(x^*) \| \leq k_0 \| x - x^* \| \\
\text{and} \quad \| \nabla^2 c_i(x) - \nabla^2 c_i(x^*) \| \leq k_i \| x - x^* \| \quad (i = 1, \ldots, m).
\]

Let \( \bar{k}_i = \| \nabla^2 c_i(x^*) \| + k_i \epsilon \), \( i = 1, \ldots, m \),

then for any \( x \) in \( N(x^*) \), the following holds:
If \( x_1 \) and \( x_2 \) belong in \( N(x^*, \varepsilon) \) and \( \gamma = \max \left( \gamma_1, \gamma_2 \right) \), then by the mean value theorem, (Ortega & Rheinboldt (1970)), it follows that:

\[
\| \gamma_1 \nabla f(x_1) - \gamma_2 \nabla f(x_2) - \nabla^2 f(x^*)(x_1 - x_2) \| \leq \varepsilon
\]

\[
\| \gamma \left\{ \nabla f(x_1) - \nabla f(x_2) - \frac{\nabla^2 f(x^*) (x_1 - x_2)}{\gamma} \right\} \| \leq \varepsilon
\]

\[
\leq k_0 \sup_{0 \leq \xi \leq 1} \| \nabla^2 f(\xi x_1 + (1-\xi)x_2) - \nabla^2 f(x^*) \| \frac{\| x_1 - x_2 \|}{\gamma}
\]

\[
= k_0 \max \left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} \| x_1 - x_2 \|.
\]

Similarly it follows that for any \( i = 1, \ldots, m \)

\[
\| \gamma_1 \nabla c_i(x_1) - \gamma_2 \nabla c_i(x_2) - \nabla^2 c_i(x^*)(x_1 - x_2) \| \leq \varepsilon
\]

\[
\leq k_i \max \left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} \| x_1 - x_2 \|.
\]

Hence

\[
\| \lambda_1 \gamma_1 \nabla c_i(x_1) - \lambda_2 \gamma_2 \nabla c_i(x_2) - \lambda_i^* \nabla^2 c_i(x^*)(x_1 - x_2) \| \leq \varepsilon
\]
\[ \varepsilon \| \lambda_i - \lambda_i^* \| \| \gamma_1 \nabla c_i(x_1) - \gamma_2 \nabla c_i(x_2) \| + \]

\[ + \| \lambda_i^* \| (\| \gamma_1 \nabla c_i(x_1) - \gamma_2 \nabla c_i(x) - \nabla^2 c_i(x^*)(x_1-x_2) \| \leq \varepsilon \]

\[ \leq \gamma |\lambda_i - \lambda_i^*| \sup_{0 \leq \xi \leq 1} \| \nabla^2 c_i(\xi x_1 + (1-\xi)x_2) \| \frac{\| x_1 - x_2 \|}{\gamma} + \]

\[ + \gamma |\lambda_i^*| k_i \max \left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} \frac{\| x_1 - x_2 \|}{\gamma} \leq \varepsilon \]

\[ \leq (\text{using relation (6.4.4)}) \]

\[ \left\{ \bar{k}_1 |\lambda_i - \lambda_i^*| + k_i |\lambda_i^*| \max \left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} \right\} \| x_1 - x_2 \| . \]

Let

\[ \hat{k}_1 = k_0 + \sum_{i=1}^m |\lambda_i^*| k_i \quad \text{and} \quad \hat{k}_2 = \| (\bar{k}_1, \bar{k}_2, \ldots, \bar{k}_m) \| \]

then

\[ \| \nabla f(x_1) + \nabla c(x_1) - \nabla f(x_2) - \nabla c(x_2) \| \leq \varepsilon \]

\[ \leq \| \nabla f(x_1) - \nabla f(x_2) - \nabla^2 f(x^*)(x_1-x_2) \| + \sum_{i=1}^m |\lambda_i \nabla c_i(x_1) - \lambda_i \nabla c_i(x_2) - \lambda_i^* \nabla^2 c_i(x^*)(x_1-x_2) \| \leq \varepsilon \]

\[ \leq \left\{ \bar{k}_1 \max \left\{ \| x_1 - x^* \|, \| x_2 - x^* \| \right\} + \hat{k}_2 \| \lambda - \lambda^* \| \right\} \| x_1 - x_2 \|. \]
This completes the proof of the Lemma. The following theorem provides a sufficient condition to guarantee Q-superlinear convergence of Algorithm (6.2).

**Theorem (6.4.1):**

Let $z^* = (x^*, \lambda^*)$ be a Kuhn-Tucker pair of problem (P2) satisfying the Jacobian uniqueness condition and $f, c_i, (i = 1, \ldots, m)$ be Lipschitz continuous functions belonging to the class $C^2$.

For the Algorithm (6.2) with any $c_k$ such that $c_k^T p_k \neq 0$,

$$\frac{\|Mc_k - M^{-1} p_k\|}{\|M^{-1} p_k\|} \leq \mu \max \left\{ \|z_k - z^*\|, \|z_{k+1} - z^*\| \right\}$$

for a constant $\mu$ and for any fixed non-singular symmetric matrix $M$ (e.g. $M = \left[ \nabla^2 L(z^*)^{-1} \right]$).

If $z_o$ and $E_o$ are sufficiently close to $z^*$ and $\nabla^2 L(z^*)$ respectively, then the sequence $\{z_k\}$ generated by the Algorithm (6.2) is well defined and converges Q-superlinearly to $z^*$.

The proof of this theorem follows the same way as theorem (5.2) of Han (1976). The only change made is the use of the previous quoted Lemma (6.4.2) instead of Lemma (3.3) of Han (1976).

**6.5 COMPUTATIONAL EXPERIENCE.**

An experimental computer program has been written which implements Algorithm (P2) generally enough for the new model to be incorporated. The program following the lines of Powell (1977) is intended to test the validity and efficiency of the new algorithm based on a non-quadratic model and compare it with the algorithm
using the standard model. Storage requirements and execution
time of the program, can be greatly improved but this will affect both models equally.

Although the performance of the program on test problems has been quite satisfactory, full implementation of more sophisticated strategies (i.e. a better quadratic programming subroutine) would further improve its performance.

A relatively small number of numerical results for a few typical problems will be given. Most of the test problems selected are well known examples. Only those problems for which analytic derivatives could be conveniently supplied were selected. This requirement eliminated mainly problems having discontinuities in functions or derivatives.

The computational experiments were performed on a PRIME computer using double precision arithmetic. For a comparison with results reported by other papers, the computational results should be based on generally applicable preferably machine independent performance indicators.

The performance indicator used was the number of function evaluations needed to solve the test problems. The iteration numbers are also included but these do not give a good impression of the computational effort required for the solution of the problems.
The execution of an algorithm requires the computation of the value of both the objective and the constrained functions as well as their gradients at intermediate points. Instead of presenting all function evaluations separately, the number of equivalent objective function evaluations is given. This means that all constrained function evaluations are to be converted into objective function evaluations. For this the general assumption is made that the cost of the objective function evaluation is the same as the cost of a constrained function evaluation.

In the following table the figures in parenthesis give the iteration numbers. In order to compare the efficiency of the algorithms the same termination criteria were applied. Then convergence to the same point \( x^* \) means that both algorithms locate this point with the same accuracy. The algorithms converged if both criteria were satisfied:

\[
\text{(CR.1)} : \quad \| x_k - x_{k-1} \| \leq \varepsilon_1 \left( \| x_k \| + 1 \right) \quad \text{and}
\]
\[
\text{(CR.2)} : \quad | c_i(x_k) | \leq \varepsilon_2 \left( \| x_k \| + 1 \right)
\]

for all currently violated constraints and for some prescribed parameters \( \varepsilon_1 \) and \( \varepsilon_2 > 0 \). The numerical values for \( \varepsilon_1 \) and \( \varepsilon_2 \) used were:

\[
\varepsilon_1 = \varepsilon_2 = 10^{-5}
\]
# Table 6.5

<table>
<thead>
<tr>
<th>Test problem</th>
<th>Rational</th>
<th>Quadratic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(13) 172</td>
<td>(18) 194</td>
</tr>
<tr>
<td>2</td>
<td>(9) 72</td>
<td>(10) 83</td>
</tr>
<tr>
<td>3 $x_o$: (1)</td>
<td>(22) 284</td>
<td>(31) 365</td>
</tr>
<tr>
<td>3 $x_o$: (2)</td>
<td>(19) 253</td>
<td>(28) 324</td>
</tr>
<tr>
<td>3 $x_o$: (3)</td>
<td>(26) 310</td>
<td>(33) 387</td>
</tr>
<tr>
<td>3 $x_o$: (4)</td>
<td>(28) 320</td>
<td>(36) 395</td>
</tr>
<tr>
<td>4</td>
<td>(15) 183</td>
<td>(21) 270</td>
</tr>
</tbody>
</table>
6.6 **DISCUSSION.**

The results summarized in Table 6.5 are an indication of the behaviour of the two algorithms. The quadratic-based algorithm seems to have more difficulties in solving the test problems than the proposed one. This was expected as Variable Metric algorithm based on the same non-quadratic model behaved better in general than the standard one for the unconstrained problem (P1).

In this chapter a unified treatment of unconstrained and constrained optimization by Variable Metric algorithms is presented. This unification opens a very interesting field of future research. Over the last decade not only the Variable Metric algorithms but the conjugate gradient algorithms have been developed extensively and examined in depth. It is reasonable to expect that this algorithm may also work when some other efficient updating schemes are employed in this way.

There are two conjectures which ought to be mentioned:

1. As proposed by Stewart (1967) in the unconstrained case, Algorithm (6.2) could be modified by the use of difference approximations of derivatives. The modified algorithms could be efficient especially in case derivatives are not readily available.

2. Under some suitable conditions the elegant results of Dixon (1972) could also be true in the constrained case, i.e. some family of algorithms could produce an identical sequence of points.
APPENDIX

TP1: \( N = 3, \ M = 1 \) (Non-linear)

\[ x_0 = (1, 1, 1), \quad f(x_0) = -1 \]
\[ x^* = (4.000, 2.828, 2.000), \quad f(x^*) = -22.627 \]

minimize: \( f(x) = -x_1 x_2 x_3 \)

subject to: \( x_i \geq 0 \) for \( i = 1, 2, 3 \)

\[ 48 - x_1^2 - 2x_2^2 - 4x_3^2 \geq 0 \]

TP2: \( N = 3, \ M = 1 \) (Linear)

\[ x_0 = (10, 10, 10), \quad f(x_0) = -1000 \]
\[ x^* = (24, 12, 12), \quad f(x^*) = -3456 \]

minimize: \( f(x) = -x_1 x_2 x_3 \)

subject to: \( x_1 + 2x_2 + 2x_3 - 72 = 0 \)

\( x_i \geq 0 \) for \( i = 1, 2, 3 \)

TP3: \( N = 5, \ M = 3 \)

(1) \[ x_0 = (-1.7, 1.6, 1.8, -0.8, -0.8) \]

(2) \[ x_0 = (-1.7, 1.6, 1.8, -0.76, -0.76) \]
(3) \( x_0 = (-2, 2, 2, -1, -1) \)

(4) \( x_0 = (-2, 1.5, 2, -1, -1) \)

\( x^* = (-1.71714, 1.59571, 1.82725, -0.76364, -0.76364) \)

\( f(x^*) = 0.05395 \)

**minimize** : \( f(x) = \exp(x_1x_2x_3x_4x_5) \)

**subject to** : 
\[ \begin{align*}
  x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 &= 0 \\
  x_2x_3 - 5x_4x_5 &= 0 \\
  x_1^3 + x_2^3 + 1 &= 0
\end{align*} \]

**TP4** : \( N = 5, \quad M = 3 \)

\( x_0 = (78.62, 33.44, 31.07, 44.18, 35.32) \)

\( x^* = (46.51, 33, 29.65, 58.37, 40.62) \)

\( f(x^*) = -31596.7 \)

**minimize** : 
\[ f(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1^2 - 40792.141 + (x_1 - 78)^2 + (x_4 - 45)^2 \]

**subject to** : 
\[ \begin{align*}
  x_1x_4 - 0.0022053x_3x_5 &= 92 \\
  9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 &= 20 \\
  x_2 - 33 &= 0
\end{align*} \]
CHAPTER VII

SUMMARY AND CONCLUSIONS
In this thesis various non-quadratic models are used as a basis for different methods for unconstrained and constrained optimization. These models have been analyzed, implemented and tested to some extent.

A rational one-dimensional model proposed by Moufti (1977) has been analyzed and studied from the convergence point of view, in the second chapter. The average rate of convergence of this model is shown to be superlinear.

A straightforward extension of the one-dimensional model has been tried in Chapter 3. The multidimensional model is analyzed and implemented in Winfield's algorithm. The various disadvantages of using such a model were discussed and alternative models were introduced. Winfield's algorithm proceeded by finding a new point at each iteration and including it in the current grid. The new grid consists of the points of the previous one plus this new point. Therefore, the next model does not differ much from the previous one. Hence the grid size becomes smaller and the total number of function evaluations increases. A possible way of improving the performance of the algorithm is to allow the algorithm to discard more than one point at each iteration. This will have, as a result, a better representation of the objective function in the area of the new point.

Sophisticated gradient techniques like the conjugate gradient and the Variable Metric Method have been studied using as models non-quadratic functions. Numerical tests were carried out on low and high dimensionality problems and comparisons were made amongst different non-quadratic and quadratic models.
Conjugate gradient methods were tested on functions of four hundred dimensions. Oversizing of the computer storage capacity limited the tests of Variable Metric Methods to functions of one hundred variables.

The general conclusion that can be drawn from the tests on the gradient methods, is that the non-quadratic models performed, in general, as well as the standard model used so far and showed a superiority on problems of high dimensionality.

The ideas of a Variable Metric Method using a non-quadratic model, are applied in the sixth chapter to constrained optimization problems, following the lines of a recent paper by Powell (1977).

The effects of this thesis, it is hoped, will fall into two main categories. By the use of analysis and by pointing out the desired properties of the non-quadratic models in various optimization techniques, a better understanding should be gained on the use of such models. Optimization techniques are now at a point where models, other than quadratic, are sought in both the one-dimensional and the multi-dimensional cases.

Secondly, by analyzing different aspects of the main unconstrained optimization techniques, more understanding was gained and new algorithms were introduced.

A number of computational experiments have been performed to substantiate the theoretical results. The comparisons made were as critical as possible. The numerical experiments carried out in the fourth chapter give a good impression of the effects of varying degrees of accuracy in the line search in terms of the overall performance of the conjugate gradient algorithm.
Finally some comments are made on the major unfinished tasks which are evident from this thesis. Most important are more extensive computer testing of the proposed algorithms on high dimensionality problems and the investigation of the effect of numerical derivatives for the various algorithms proposed in this thesis.

Further work could also be directed towards new non-quadratic models with different basic properties than rational models (e.g. positive definiteness).
REFERENCES


TAMIR , A., "Line search techniques based on interpolating polynomials using function values only". Management Science, 22, No.5, (1976).


