DR.J.K.REID, N.A.R.
B.3.9.

CODED FOR 1,2,3,4,

T.P. 515

AN ALGORITHM FOR MINIMIZATION USING EXACT SECOND DERIVATIVES

by

M. D. Hebden

Theoretical Physics Division,
U.K.A.E.A. Research Group,
Atomic Energy Research Establishment,
HARWELL.

March, 1973.

HL.73/983

T.P. 515

AN ALGORITHM FOR MINIMIZATION USING EXACT SECOND DERIVATIVES

by

M. D. Hebden

ABSTRACT

A review of the methods currently available for the minimization of a function whose first and second derivatives can be calculated shows either that the method requires the eigensolution of the hessian, or with one exception that a simple example can be found which causes the method to fail. In this paper one of the successful methods that requires the eigensolution is modified so that at each iteration the solution of a number (approximately two) of systems of linear equations is required, instead of the eigenvalue calculation.

Theoretical Physics Division,
A.E.R.E. Harwell, Didcot, Berks.

March, 1973

1. Introduction

The problem considered is the minimization of a function $f(\underline{x})$ of n variables $\underline{x}$ when the function has a form that allows the gradient $\underline{g}$ and the hessian G to be evaluated

i.e. $$g_i = \frac{\partial f}{\partial x_i} \qquad i=1,2,\ldots,n$$

$$G_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j} \qquad i,j=1,2,\ldots,n \ .$$

The basic approach to such an optimization problem is to use Newton's method which generates the iterates

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - [G(\underline{x}^{(k)})]^{-1} \, \underline{g}(\underline{x}^{(k)}) \ . \qquad (1.1)$$

In one variable this reduces to Newton-Raphson iteration to find a zero of $g(\underline{x})$. More generally $\underline{x}^{(k+1)}$ is a stationary point of the quadratic approximation

$$f(\underline{x}) \approx f(\underline{x}^{(k)}) + (\underline{x}-\underline{x}^{(k)}) \, \underline{g}(\underline{x}^{(k)}) + \tfrac{1}{2}(\underline{x}-\underline{x}^{(k)})^T \, G(\underline{x}-\underline{x}^{(k)}) \ . \qquad (1.2)$$

There are two major difficulties when using this iteration, one occurs because $\underline{x}^{(k+1)}$ may lie outside the region where the quadratic approximation is accurate, in which case it may be that $f(\underline{x}^{(k+1)}) \geq f(\underline{x}^{(k)})$. The other occurs when G is not positive definite in which case, even though the quadratic approximation is accurate, the move to a stationary point may be in a direction in which f increases.

To counteract these difficulties it is necessary to generate a downhill direction $\underline{s}$ (i.e. $\underline{g}^T\underline{s} < 0$), whereupon taking a sufficiently small step in this direction ensures that the value of the objective function is reduced at each iteration.

In section 2 various published strategies are described, together with their drawbacks. Subsequently the paper is devoted to the derivation of an

algorithm that overcomes many of the objections to other methods. Some numerical evidence is presented to demonstrate the success of the algorithm.

2. Algorithms

Five algorithms presently available are those due to Goldfeld, Quandt and Trotter (1966), Greenstadt (1967), Fiaccio and McCormick (1968), Matthews and Davies (1971), and Gill and Murray (1972). As the algorithm described in this paper is a development of that due to Goldfeld et al., particular attention is given to that algorithm.

In the GQT algorithm a multiple of the identity matrix is added to G, the multiple being chosen to satisfy the conditions

i) $(G+\lambda I)$ is positive definite

ii) $||\underline{\delta}||_2 = ||(G+\lambda I)^{-1}\ \underline{g}||_2 \leq d$. (2.1)

Where d, a restriction on the step size, is updated after each iteration with the intention that the next step will be restricted to a region in which the quadratic approximation is expected to be applicable. The method generates the iterates

$$\underline{\delta} = (G+\lambda I)^{-1}\underline{g}$$

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} - \underline{\delta} \ . \qquad (2.2)$$

If at any time $f(\underline{x}^{(k+1)}) - f(\underline{x}^{(k)})$ is less than a small multiple ρ of the reduction predicted by the quadratic approximation (1.2), d is reduced, $x^{(k+1)}$ is discarded and the iteration repeated. A value of ρ used in practice is 0.0001. (It should be noted that G,$\underline{g}$ and $\underline{\delta}$ are all functions of $\underline{x}^{(k)}$ but the superscript is omitted).

The optimal value of λ is the smallest non-negative number that is consistent with conditions (2.1). Therefore the following method for calculating λ is suitable when the eigenvalues and eigenvectors of G are known. Denoting the eigenvalues by $\mu_1,\mu_2,\ldots,\mu_n$ ($\mu_i \leq \mu_j$ if $i \leq j$), and the eigenvectors by $\underline{e}_1,\underline{e}_2,\ldots,\underline{e}_n$, we have

$$(G+\lambda I)^{-1} = \sum_{i=1}^{n} \frac{1}{\mu_i+\lambda} \underline{e}_i \underline{e}_i^T , \qquad (2.3)$$

$$\underline{\delta} = \sum_{i=1}^{n} \frac{\underline{e}_i^T \underline{g}}{\mu_i+\lambda} \underline{e}_i \qquad (2.4)$$

and

$$||\underline{\delta}||_2^2 = \sum_{i=1}^{n} \left(\frac{\underline{e}_i^T \underline{g}}{\mu_i+\lambda} \right)^2 . \qquad (2.5)$$

A useful estimate of λ is

$$\lambda_o = \max \left(\max_i \left(\frac{|\underline{e}_i^T \underline{g}|}{d} - \mu_i \right), 0 \right) \qquad (2.6)$$

and it can be shown that λ_o is never an over-estimate of λ. The estimate can be refined by applying Newton-Raphson iteration to the equation

$$\sum_{i=1}^{n} \left(\frac{\underline{e}_i^T \underline{g}}{\mu_i+\lambda} \right)^2 - d^2 = 0. \qquad (2.7)$$

This method is satisfactory except when simultaneously μ_1 is negative and $\underline{e}_1^T \underline{g}$ is zero, corresponding to zero gradient and negative curvature in the direction $\underline{e}_1$. In this case a consequence of (2.6) is that $(G+\lambda_o I)$ may be singular, however $\underline{\delta}(\lambda_o)$ can be calculated by ignoring the first component of $\underline{\delta}$. If now $||\underline{\delta}(\lambda_o)||_2 > d$ the normal refinement can take place, otherwise a satisfactory solution is to take a step of norm d in the direction of $\underline{e}_1$.

The remaining problem, that of updating d, is dealt with extensively in section 8.

A study of equation (2.4) shows that an attribute of the method is that the step $\underline{\delta}$ is biased towards directions in which the gradient is large, but as well as this the addition of λ weights the step in favour of the

directions with negative curvature. In fact Goldfeld et al.(1966) prove that the step $-\underline{\delta}(\lambda)$ minimizes the quadratic

$$Q(\underline{x}+\underline{\delta}) = f(\underline{x}) + \underline{\delta}^T\underline{g} + \tfrac{1}{2}\underline{\delta}^T G\underline{\delta} \qquad (2.8)$$

subject to the restriction $||\underline{\delta}||_2 \leq ||\underline{\delta}(\lambda)||_2$.

In an alternative method due to Greenstadt a positive semi-definite approximation to G is found by replacing each eigenvalue by its modulus. Thus

$$G^* = \sum_{i=1}^{n} |\lambda_i| \, \underline{e}_i \underline{e}_i^T$$

A search direction $\underline{s} = -[G^*]^{-1}\underline{g}$ is calculated and $\underline{x}^{(k+1)}$ is at the minimum of f along the line $\underline{x}^{(k)} + \alpha\underline{s}$. The method breaks down if any eigenvalue is zero, and the method is unable to distinguish between positive and negative curvature so that no progress can be made away from a saddle point. The point is demonstrated in the example below. Minimize

$$f(\underline{x}) = x_1^2 - x_2^2 + \tfrac{1}{2} x_2^4 \qquad (2.9)$$

starting from (1,0). The initial search direction is

$$\underline{s} = \begin{pmatrix} 2 & 0 \\ 0 & |-2| \end{pmatrix}^{-1} \begin{pmatrix} 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

A linear search terminates at the point (0,0), and the next search direction is

$$\underline{s} = \begin{pmatrix} 2 & 0 \\ 0 & |-2| \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

The method due to Fiaccio and McCormick avoids the eigenproblem, and that of negative curvature, by using the factorization

$$G = LDL^T$$

where L is lower triangular with unit diagonal and S is diagonal. If G is positive definite the search direction is $\underline{s} = -G^{-1}\underline{g}$. If, however, D contains negative elements or zeros the alternative (and satisfactory) strategy of identifying a direction of negative curvature is adopted. When G is not positive definite the factorization may be unstable, and the effect of instability on the algorithm has not been investigated. Moreover there are matrices for which an LDL^T factorization does not exist. Again the point is demonstrated by an example, namely to minimize

$$f(\underline{x}) = (x_1^4 - 3)^2 + x_2^4 + (x_1 - 3^{\frac{1}{4}})\ x_2$$

starting from (0,0). It will be seen that at this point the hessian is

$$G = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

for which no factorization of the form LDL^T can be found.

Matthews and Davies overcome the factorization problem above by attempting the factorization

$$G = LU \ .$$

With L unit lower triangular and U upper triangular. To ensure that the factorization can be completed the diagonal elements of U are modified as they are used as pivots. If U_{ii} is negative it is replaced by $|U_{ii}|$, and if it is zero it is replaced by unity. Consequently the factors L* and U* of an unknown positive definite matrix G* are generated. G* is then used to generate the search direction

$$\underline{s} = -\,[G^*]^{-1}\ \underline{g} \ .$$

Unfortunately this factorization is potentially unstable, and in addition, as with the Greenstadt algorithm, no progress can be made away from

a saddle point. In fact for the example (2.9) the behaviour of the algorithms is identical.

Recently Gill and Murray (and Gill, Murray and Picken (1972)) have published an algorithm in the style of that of Matthews and Davies. The essential difference is that pivots are increased in size by an amount that is sufficient to ensure a stable factorization. Eventually the factorization

$$G + E = LDL^T$$

is found where(G+E) is"sufficiently positive definite". If $||\underline{g}|| \neq 0$ the search direction is $(G+E)^{-1}\underline{g}$, otherwise D,E and L are used to generate a direction of negative curvature.

The method is sound, but rather different in approach to that of Goldfeld et al. The tendency of the Gill and Murray algorithm is to replace negative curvature by positive curvature and so weight the step in favour of directions of small absolute curvature.

3. A new algorithm

Excepting the GQT algorithm simple problems have been found which cannot be solved by the above algorithms. The remainder of the paper is devoted to an attempt to modify the technique used by GQT so that the need for the eigensolution of the hessian is eliminated.

The problem is to solve for λ the equation

$$||(G+\lambda I)^{-1}\underline{g}||_2 = d \qquad (3.1)$$

subject to $(G+\lambda I)$ being positive definite; taking into account any special cases that arise.

Section 4 is devoted to the development of an iterative scheme for the refinement of λ which is applicable if $(G+\lambda I)$ is positive definite. Section 5 deals with the possibility that $(G+\lambda I)$ may be indefinite, and in section 6 an upper bound for λ is derived. In section 7, this work is put

together to yield a comprehensive strategy for the determination of λ which experience shows to require an average of only two matrix factorizations. Section 8 covers the adjustment of the step size d, whilst in section 9 some numerical results are presented.

4. An iterative scheme for λ

Initially, any special difficulties are neglected, and an iterative scheme for the refinement of λ is developed. It is assumed that a value of λ has been found such that $(G+\lambda I)$ is positive definite but $||(G+\lambda I)^{-1}\underline{g}||_2 \neq d$. The scheme consists of identifying the derivative, with respect to λ of $||\underline{\delta}(\lambda)||_2$; fitting a rational function with constant numerator and linear denominator to $||\underline{\delta}(\lambda)||_2$ and its gradient, and determining the change $\delta\lambda$ that makes the rational function equal to d.

$$\underline{\delta}(\lambda) = (G+\lambda I)^{-1}\underline{g} \tag{4.1}$$

$$\therefore \quad ||\underline{\delta}(\lambda)||_2 = [\underline{g}^T(G+\lambda I)^{-2}\underline{g}]^{\frac{1}{2}} \tag{4.2}$$

$$\therefore \quad \frac{\partial}{\partial\lambda}||\underline{\delta}(\lambda)||_2 = \frac{1}{2||\underline{\delta}(\lambda)||_2}(-2\,\underline{g}^T(G+\lambda I)^{-3}\underline{g}). \tag{4.3}$$

Thus by solving the two systems of equations (with the same coefficient matrix)

$$(G+\lambda I)\underline{\delta} = \underline{g} \tag{4.4}$$

and

$$(G+\lambda I)\underline{\gamma} = \underline{\delta} \tag{4.5}$$

we have

$$||\underline{\delta}(\lambda)||_2 = (\underline{\delta}^T\underline{\delta})^{\frac{1}{2}} \tag{4.6}$$

$$\frac{\partial}{\partial\lambda}||\underline{\delta}(\lambda)||_2 = -(\underline{\delta}^T\underline{\gamma})/||\underline{\delta}(\lambda)||_2 \,. \tag{4.7}$$

It would be straightforward to solve equation (3.1) by Newton-Raphson iteration, but as $\underline{\delta}$ is a linear combination of the eigenvectors of G, the coefficients being $\underline{e}_i^T\underline{g}/(\mu_i+\lambda)$, it seems more reasonable to use the

approximation

$$||\underline{\delta}(\lambda)||_2 = \frac{a}{b+\lambda} \quad . \tag{4.8}$$

Then

$$\frac{\partial}{\partial\lambda} ||\underline{\delta}(\lambda)||_2 = \frac{-a}{(b+\lambda)^2} \quad . \tag{4.9}$$

Equating terms leads to the expression

$$\frac{1}{b+\lambda} = -\frac{\partial}{\partial\lambda} ||\underline{\delta}(\lambda)||_2 / ||\underline{\delta}(\lambda)||_2 \quad . \tag{4.10}$$

It now remains to find $\delta\lambda$ such that

$$\frac{a}{b+\lambda+\delta\lambda} = d \tag{4.11}$$

i.e.

$$\begin{aligned} \delta\lambda &= \frac{a - d(b+\lambda)}{d} \\ &= -\frac{||\underline{\delta}(\lambda)|| - d}{d} \; \frac{-||\underline{\delta}(\lambda)||}{\frac{\partial}{\partial\lambda} ||\delta(\lambda)||} \\ &= \left(\frac{||\underline{\delta}(\lambda)||}{d} - 1 \right) \frac{\underline{\delta}^T\underline{\delta}}{\underline{\delta}^T\underline{\gamma}} \quad . \end{aligned} \tag{4.12}$$

This provides the basis of an iterative scheme to find λ by repeatedly solving systems of linear equations. Convergence of the scheme is not guaranteed so it is necessary to use it in conjunction with a bracket on the root.

An additional complication is that the matrix $(G+\lambda I)$ may not be positive definite. It is necessary to identify such cases and to be able to determine an improved value for λ.

5. The procedure when $(G+\lambda I)$ may be indefinite

Usually the matrix G is positive definite and the iteration described in section 3 can be used from the initial estimate $\lambda=0$. Otherwise it is necessary to find a value for λ such that $(G+\lambda I)$ is positive definite. In addition sometimes it happens that an iterate is produced for which $(G+\lambda I)$ is not positive definite.

To allow for an indefinite matrix in an efficient manner an attempt is made to solve the system

$$(G+\lambda I)\underline{\delta} = \underline{g} \qquad (5.1)$$

by first obtaining the factorization

$$(G+\lambda I) = LDL^T . \qquad (5.2)$$

If at any stage a negative diagonal element is generated the purpose of the calculation changes to that of finding a value μ such that $(G+\lambda I+\mu I)$ is positive definite.

Interchanges are made to ensure that the largest diagonal element is chosen as pivot, but for the purposes of analysis it can be assumed that no interchanges are necessary.

It is apparent that if G has a negative diagonal element it cannot be positive definite, so a lower bound on λ is set as

$$\max_i \ (0,-G_{ii}) .$$

Thus the possibility that $(G+\lambda I)$ has some negative diagonal elememts is eliminated.

The factorization starts in the usual way by finding d_1 and $\underline{\ell}_1$ (the first column of L) such that

$$(G+\lambda I) = (\underline{\ell}_1,\underline{0},\ldots,\underline{0}) \begin{pmatrix} d_1 & 0 & & \\ & \ddots & & \\ & & \ddots & \\ 0 & & & \end{pmatrix} (\underline{\ell}_1,\underline{0},\ldots,\underline{0})^T + G_1 \tag{5.3}$$

where the first row and columns of G_1 are zero, and where the first component of $\underline{\ell}_1$ is unity. It follows that

$$\begin{aligned} d_1 &= G_{11}+\lambda \\ L_{i1} &= G_{i1}/d_1 \qquad\qquad i\neq 1. \end{aligned} \tag{5.4}$$

Then

$$(G_1)_{ij} = G_{ij}+\lambda\delta_{ij} - d_1 L_{i1} L_{j1} \qquad i,j\neq 1. \tag{5.5}$$

Notably the diagonal elements of G_1 are

$$\begin{aligned} (G_1)_{ii} &= ((G_{ii}+\lambda) - d_1 L_{i1}^2) \qquad\qquad i\neq 1 \\ &= ((G_{11}+\lambda)(G_{ii}+\lambda) - G_{1i}^2)/(G_{11}+\lambda) \qquad i\neq 1 . \end{aligned} \tag{5.6}$$

If any of these elements are negative this stage of the factorization is abandoned and a value κ_1 found such that

$$(G_{11}+\lambda+\kappa_1)(G_{ii}+\lambda+\kappa_1) \geq G_{1i}^2 . \qquad i\neq 1 \tag{5.7}$$

$\kappa_1 I$ is added to $(G+\lambda I)$ so that the matrix being factorized is now $(G+\lambda I+\kappa_1 I)$. Thus the matrix G_1 from the new factorization has all its diagonal elements greater than or equal to zero.

The only possible cause of trouble is that all the diagonal elements of G are zero (implying that the pivot is zero), but tests on the signs of $(G_1)_{ii}$ are accomplished without dividing by the pivot, so even in this case the factorization proceeds without difficulty.

After the elimination of the first row and column of G, the factorization continues in a similar manner eliminating the second row and column of G_1, except that if it is necessary to add a constant κ_2 to the

diagonal elements of G_1 it is not added to $(G_1)_{11}$.

Ultimately the process yields the factorization

$$(G+\lambda I+\Lambda) = LDL^T \tag{5.8}$$

where Λ is a diagonal matrix whose elements are

$$\Lambda_{ii} = \sum_{j=1}^{i} \kappa_j \, . \tag{5.9}$$

Now the κ_i have been chosen to keep the elements of D non-negative, so that $(G+\lambda I+\Lambda)$ is positive semi-definite. Thus if μ is set to Λ_{nn}, $(G+\lambda I+\mu I)$ is positive semi-definite.

An additional feature is that if $\Lambda \neq 0$, then at least one diagonal element of D is zero. So $(G+\lambda I+\Lambda)$ is singular and a zero eigenvector $\underline{\eta}$ can be found by solving $L^T\underline{\eta}=\underline{\gamma}$, where γ_i is equal to unity if D_{ii} is zero, and to zero otherwise. The vector $\underline{\eta}$ is stored for use when the correct value for λ cannot easily be found, as will be described in section 6.

This factorization is insufficient on two counts. One is that if $(G+\lambda I)$ is singular with no negative eigenvalues, μ is zero and it is not obvious how to proceed. The other is the problem of determining a zero numerically.

Both of these are overcome by supposing that a diagonal element is negative only if its modulus is less than $-\varepsilon_1 = -\varepsilon|G_{ii}+\lambda|$, where ε is a measure of the relative accuracy of the machine. The modification would allow a pivot to be negative, and to avoid this any pivot less than ε_1 is set to ε_1, or if ε_1 is zero the pivot is set to ε. In setting the pivot to ε the implicit assumption is made that the diagonal elements of G are of order unity. Ideally some measure of the scale in the problem should be incorporated, for example the pivot could be set to $\varepsilon||G||$. This extension is not followed as it would require additional computation, and

only defers the difficulty as it may be that $||G|| = 0$.

If μ is non-zero this amendment has no effect, otherwise $-\lambda$ is close to the smallest eigenvalue of G and the system of equations solved is

$$(G+\lambda I+E)\ \underline{\delta} = \underline{g}$$

where E contains the small terms added to $(G+\lambda I)$ to make it non-singular. If now $||\underline{\delta}||_2 > d$ the iterative refinement (4.2) can be used, whereas smaller values of $||\underline{\delta}||_2$ imply that the correct value is λ is known.

6. An upper bound on λ

A useful technique when using an iterative method to find a root of a single non linear equation is to bracket the root and then ensure that the iterates lie within the bracket but not arbitrarily close to either end point. In this way for any required accuracy the finite termination of the iterates can be guaranteed.

In section 5 it was shown that a value μ can be found such that $(G+\lambda I+\mu I)$ is positive semi-definite. This is now used to find a value λ_{max} such that

$$||\underline{\delta}(\lambda_{max})||_2 \leq d. \qquad (6.1)$$

Because $(G+\lambda I+\mu I)$ is positive semi-definite, the eigenvalues of $(G+\lambda I+\mu I+\xi I)$ are bounded below by ξ, and those of the inverse matrix are bounded above by ξ^{-1}

$$\begin{aligned} \therefore ||\underline{\delta}(\lambda+\mu+\xi)||_2 &= ||(G+\lambda I+\mu I+\xi I)^{-1}\ \underline{g}||_2 \\ &\leq ||(G+\lambda I+\mu I+\xi I)^{-1}||_2 ||\underline{g}||_2 \\ &\leq \xi^{-1}||\underline{g}||_2 \\ &\leq d \quad \text{provided } \xi \geq ||\underline{g}||_2/d\ . \end{aligned}$$

It follows that we can obtain inequality (6.1) by setting

$$\lambda_{max} = \lambda+\mu + ||\underline{g}||_2/d\ . \qquad (6.2)$$

7. The determination of λ

Here it is shown how the ideas of the previous sections are put together to form a coherent scheme for the rapid determination of a suitable value for λ.

The successful operation of the minimization routine requires that where possible the value $\lambda=0$ is chosen, and so unless G has some negative elements on the diagonal, zero is the first value of λ that is tried.

The LDL^T factorization of G is attempted, and if G is positive definite the step $\underline{\delta}=G^{-1}\underline{g}$ is generated and $||\underline{\delta}||_2$ compared with d. $||\underline{\delta}|| < d$ allows a Newton step to be taken, otherwise $\underline{\gamma}=G^{-1}\underline{\delta}$ is determined, and hence (as described in sections 4 and 6) we find an improved value of λ and an upper bound λ_{max}. Also a lower bound, $\lambda_{min}=0$ is known. Subsequently the iterative scheme of section 4 is applied with the restriction that no iterate may lie within $(\lambda_{max}-\lambda_{min})/10$ of either end point.

If G is not positive definite a value μ is found such that $(G+\mu I)$ is positive semi-definite. The iteration starts from $\lambda=\mu$ with the bracket $\lambda_{min}=0$, $\lambda_{max}=\mu + ||\underline{g}||_2/d$. A complication in this case is that an iterate λ may be generated such that $(G+\lambda I)$ is not positive definite. Here, as happens when $\lambda=0$, a value μ is found such that $(G+\lambda I+\mu I)$ is positive semi-definite and $\lambda+\mu$ is taken as the next iterate, with the restriction $(\lambda+\mu) \leq \frac{1}{2}(\lambda_{max}+\lambda_{min})$.

Should G have a negative diagonal element the initial value of λ is $\max(-G_{ii})$, and the subsequent iterations are as above.

To avoid the excessive refinement of $\underline{\delta}$ the iteration terminates when $\underline{\delta}$ satisfies $0.9d \leq ||\underline{\delta}||_2 \leq 1.1d$. There remains the problem that the interval of suitable values for λ may be small or even non-existent. The cause of such difficulty is that μ_1 (the smallest eigenvalue) is not positive and $\underline{g}^T\underline{e}_1$ (the component of the gradient in the direction of its eigenvector) is small or zero.

Such problems are countered by storing a vector $\underline{\eta}_{min}$, which is $(G+\lambda_{min}I)^{-1}\underline{g}$ when $(G+\lambda_{min}I)$ is positive definite, and otherwise it is an eigenvector associated with a zero eigenvalue of $(G+\lambda_{min}I+\Lambda)$. If the difference between λ_{min} and λ_{max} becomes less than one tenth of λ_{max} the step $\underline{\delta}=\alpha\,\underline{\eta}_{min}$ is taken, where α is chosen so that $||\underline{\delta}||_2=d$ and $\underline{\delta}^T\underline{g} \leq 0$.

In this way the step chosen is usually in a downhill direction, and when it is not it can cause a reduction in $f(\underline{x})$ due to negative curvature.

8. The alteration of d

After each iteration an assessment is made of the success of using a step of norm d. The aim being to amend d so that the step taken is restricted to a region in which the quadratic approximation

$$f(\underline{x}+\underline{\delta}) \approx f(\underline{x}) + \underline{\delta}^T\underline{g} + \tfrac{1}{2}\,\underline{\delta}^T G\underline{\delta} \qquad (8.1)$$

is adequate for reducing the objective function.

Making the assumption that for some K

$$f(\underline{x}-\underline{\delta}) = f(\underline{x}) - \underline{\delta}^T\underline{g} + \tfrac{1}{2}\,\underline{\delta}^T G\underline{\delta} + K||\underline{\delta}||^3 \qquad (8.2)$$

$$|K|\;\;||\underline{\delta}||^3 = \left|\left(\underline{\delta}^T\underline{g}-\tfrac{1}{2}\underline{\delta}^T G\underline{\delta}\right) - (f(\underline{x})-f(\underline{x}-\underline{\delta}))\right|$$
$$= |\text{ pred } - \text{ ared }| \qquad (8.3)$$

(pred and ared are the predicted reduction and the actual reduction of the function).

This allows the error in the quadratic approximation when taking a step $\alpha\underline{\delta}$ to be estimated as

$$\alpha^3\,|K|\;\;||\underline{\delta}||^3 = \alpha^3|\text{pred}-\text{ared}|. \qquad (8.4)$$

In an attempt to ensure that the error in the quadratic approximation does not dominate the predicted reduction, a value for α is sought such that

$$\alpha^3|k| \quad ||\underline{\delta}||^3 = (1/\gamma)\ \text{pred}(\alpha\underline{\delta}) \qquad (8.5)$$

where $\text{pred}(\alpha\underline{\delta})$ is the predicted reduction from a step, bounded by $||\alpha\underline{\delta}||_2$, in the direction of $\underline{\delta}$ and γ some constant not less than unity. That is

$$\begin{aligned} \text{pred}(\alpha\underline{\delta}) &= \alpha\underline{g}^T\underline{\delta} - \tfrac{1}{2}\alpha^2\underline{\delta}^T G\underline{\delta} && (\alpha \leq \underline{g}^T\underline{\delta}/\underline{\delta}^T G\underline{\delta}) \\ &= \tfrac{1}{2}(\underline{g}^T\underline{\delta})^2/\underline{\delta}^T G\underline{\delta} && (\alpha > \underline{g}^T\underline{\delta}/\underline{\delta}^T G\underline{\delta}). \end{aligned} \qquad (8.6)$$

Surprisingly the behaviour of the algorithm is insensitive to the value of γ (some results are given in Table 8.2). Further, the implementation of such a scheme is no improvment over the simple set of rules:

$$\left.\begin{array}{ll} \text{if} \quad \left|\dfrac{\text{ared}}{\text{pred}} - 1\right| < 0.025 & d^{(k+1)} = 4\,d^{(k)} \\ \text{if} \quad \text{ared/pred} \geq 0.75 & d^{(k+1)} = 2\,d^{(k)} \\ \text{if } 0.25 < \text{ared/pred} < 0.75 & d^{(k+1)} = d^{(k)}\,. \end{array}\right\} \qquad (8.7)$$

In fact similar rules are recommended by Goldfeld, Quandt and Trotter.

If the actual reduction is less than one quarter of the predicted reduction, the following idea similar to that used by Fletcher (1971) is used to decrease d. Cubic interpolation to the function, gradient, and second derivative at $\underline{x}^{(k)}$ and to the function at $\underline{x}^{(k+1)}$ leads to the prediction that the minimum with respect to α of $f(\underline{x}^{(k)} - \alpha\underline{\delta})$ is at

$$\alpha = \frac{-\underline{\delta}^T G\underline{\delta} + \sqrt{(\underline{\delta}^T G\underline{\delta})^2 + 12\,\underline{g}^T\underline{\delta}(\text{pred-ared})}}{6(\text{pred-ared})}\,. \qquad (8.8)$$

To ensure a reasonable decrease, and yet avoid an excessive decrease, α is set to the value (8.8) unless it is outside the interval $0.1<\alpha<0.5$ in which case we use either $\alpha = 0.1$ or $\alpha = 0.5$, whichever is closer to the value (8.8). Then $d^{(k+1)}$ is taken as $\alpha d^{(k)}$.

A point worth noting is that for a normal step when

$$(G+\lambda I)\underline{\delta} = \underline{g}$$

we have

$$\underline{\delta}^T G\underline{\delta} = \underline{\delta}^T\underline{g} - \lambda\underline{\delta}^T\underline{\delta}.$$

Therefore the direct calculation of $\underline{\delta}^T G\underline{\delta}$ required for the adjustment of d can be avoided.

9. Numerical Results

The algorithm has been tried successfully on a number of problems, including some that do not yield to the algorithms mentioned in section 2. The results for one test function are given below.

To show that the algorithm is competitive with others, the result of applying it to Wood's function (Colville, 1968)

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

is shown in Table 9.1. There the results of other authors are also given.

Table 9.2 demonstrates the effect of varying the method by which d is updated. Results are shown for three values of γ (γ is defined in (8.5)) and for the more simple updating method. The table shows the total number of iterations and function values required to minimize Wood's function from the nine different starting points used by Matthews and Davies. To provide a setting for these results, the figures for the Greenstadt algorithm and the Matthews and Davies algorithm are also shown.

10. Conclusions

The examples of section 2 demonstrate that the only satisfactory algorithms for optimization with exact second derivatives are those of Goldfeld et al. and Gill and Murray. The former, however, requires the eigensolution of the Hessian at each iteration, and whilst this is of no

Table 9.1

A comparison of methods for minimizing Wood's function with the starting point (-3,-1,-3,-1)

	Function Evaluations	Gradients	Iterations
Goldfeld, Quandt and Trotter	47	40	40
Greenstadt	61	29	29
Fiaccio and McCormick	unknown		24
Matthews and Davies	61	27	27
Gill and Murray*	58	54	36
Proposed Method**	45	40	40

* result using recommended value $\eta = 0.5$

**requires 66 matrix factorizations.

Table 9.2

A comparison of the cumulative cost of minimizing Wood's function from nine different starting points

	Function Evaluations	Iterations
Greenstadt	501	199
Matthews and Davies	487	189
Proposed Method:		
$\gamma^{-1} = 1$	311	276
$\gamma^{-1} = 0.75$	330	273
$\gamma^{-1} = 0.5$	311	274
Simple form	308	270*

* requires 445 matrix factorizations

consequence on small problems it certainly is for large problems. The drawback is amplified if, as is likely, the method is applied to functions whose Hessian is sparse. This is because for a method that requires an eigensolution, little saving can be made either in time or in storage, whereas for a method relying on the solution of systems of equations, considerable savings can be made in both of these areas.

A direct comparison with the Gill and Murray method is difficult because one must weigh together function, gradient and hessian evaluations, and matrix factorizations. The figures in Table 9.1 clearly demonstrate that the proposed method is competitive with the alternatives, including those that are less reliable.

Acknowledgements

I should like to thank M.J.D. Powell and R. Fletcher for a number of helpful suggestions made by them during the preparation of this report.

References

Colville, A.R. (1968). "A comparative study of non-linear programming codes". IBM Tech. Rep. No. 320-2949.

Fiaccio, A.V. and McCormick, G.P. (1968). "Non-linear programming: sequential unconstrained minimization techniques". Wiley.

Fletcher, R. (1971). "A modified Marquardt subroutine for non-linear least squares". A.E.R.E. R.6799.

Gill, P.E. and Murray, W. (1972). "Two methods for the solution of linearly constrained and unconstrained optimization problems". NPL Report NAC 25.

Gill, P.E., Murray, W. and Picken, S.M. (1972). "The implementation of two modified Newton algorithms for unconstrained optimization". NPL Report NAC 24.

Goldfeld, S.M., Quandt, R.E. and Trotter, H.F. (1966). "Maximization by quadratic hill climbing". Econometrica 34, pp.541-551.

Greenstadt, J.L. (1967). "On the relative efficiency of gradient methods". Math. Comp. 21, pp.360-367.

Matthews, A. and Davies, D. (1971). "A comparison of modified Newton methods for unconstrained optimization". Comp. J. 14, pp.293-294.