A new approach to variable metric algorithms

R. Fletcher

Atomic Energy Research Establishment, Harwell, Didcot, Berkshire

An approach to variable metric algorithms has been investigated in which the linear search subproblem no longer becomes necessary. The property of quadratic termination has been replaced by one of monotonic convergence of the eigenvalues of the approximating matrix to the inverse hessian. A convex class of updating formulae which possess this property has been established, and a strategy has been indicated for choosing a member of the class so as to keep the approximation away from both singularity and unboundedness. A FORTRAN program has been tested extensively with encouraging results.

(Received October 1969)

1. Motivation

This paper deals with the problem of minimising a function $F(\boldsymbol{x})$ of n variables $\boldsymbol{x}^T = (x_1, x_2, \ldots, x_n)$ assuming that the gradient vector $\nabla_x F = \boldsymbol{g}(\boldsymbol{x})$ is available explicitly, but that the hessian G is not $(G_{ij} = (\partial^2 F/\partial x_i \partial x_j))$. Superscript T is used to denote transposition. A type of method which has achieved considerable success in solving this problem is the variable metric method (VMM) due to Davidon (1959), and simplified by Fletcher and Powell (1963). The main feature of the VMM is that an approximation H to G^{-1} is kept, and is updated at each iteration using the formula

$$H^* = H + \frac{\boldsymbol{\delta}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} - \frac{H\boldsymbol{\gamma}\boldsymbol{\gamma}^T H}{\boldsymbol{\gamma}^T H \boldsymbol{\gamma}} \tag{1}$$

where $\boldsymbol{\delta} = \boldsymbol{x}^* - \boldsymbol{x}$ and $\boldsymbol{\gamma} = \boldsymbol{g}^* - \boldsymbol{g}$ are the changes in $\boldsymbol{x}$ and $\boldsymbol{g}$ made on that iteration, and superscript * denotes values appropriate to the next iteration. The correction $\boldsymbol{\delta}$ is taken as a multiple α of a 'direction of search' $\boldsymbol{s} = -H\boldsymbol{g}$ chosen by analogy with Newton's method, so that

$$\boldsymbol{\delta} = \alpha \boldsymbol{s} = -\alpha H \boldsymbol{g}. \tag{2}$$

The multiple α is taken as the value of λ which minimises $F(\boldsymbol{x} + \lambda \boldsymbol{s})$, that is the function is minimised locally along the direction of search. The method has a number of important properties, for instance if the approximating matrix H is initially chosen to be positive definite, then this property is retained by subsequent approximations. Also if the function to be minimised is a positive definite quadratic form, then the iteration terminates in at most n iterations. Moreover Powell (AERE report, to be published) has recently produced a convergence proof for a more general class of functions.

The algorithm has, however, some inconvenient features. The main one is the need to solve the subproblem of finding α at each iteration (the 'linear search'). This is usually done by evaluating the function and gradient for a number of different values of λ and interpolating according to some strategy, until a sufficiently accurate minimum is obtained. Thus a considerable extra computing effort is required, above that for calculating $\boldsymbol{\gamma}$ and updating H. (Computing effort is most readily measured by the number of times F and $\boldsymbol{g}$ have to be evaluated.) A further disadvantage is that the linear search is hazardous to program because of the many special circumstances which can arise (for instance the minimum may not exist at all). This can lead at worst to undetected program errors; at best to a proliferation of different programs for implementing the VMM, giving rise to incompatibility in results. The linear search can also often be a disadvantage when constraints are present, because then the minimum along the line may not be feasible, even though no constraints limit the position of the ultimate solution. In this context, the flexibility of being able to generate directions of search, other than by $\boldsymbol{s} = -H\boldsymbol{g}$, would also be convenient.

It is important therefore to consider whether the linear search subproblem can be dispensed with. The importance of the linear search is that it furnishes a property which enables finite termination to be proved for quadratic functions. The first point to examine therefore is whether this termination can be proved for variable metric algorithms not requiring linear searches, and based upon updating formulae other than (1). Now quadratic termination can be proved by requiring that the successive matrices H satisfy a 'hereditary property' when the function is quadratic: that is not only must H^* satisfy $H^*\boldsymbol{\gamma} = \boldsymbol{\delta}$ (a natural property because $G^{-1}\boldsymbol{\gamma} = \boldsymbol{\delta}$), but also $H^*\hat{\boldsymbol{\gamma}} = \hat{\boldsymbol{\delta}}$ where $\hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{\delta}}$ are a pair of differences from an earlier iteration. It is quite easy to show that there is only one formula for which hereditary properties can be proved without relying upon linear searches, and for which the correction is of rank 2 in the space of $\boldsymbol{\delta}$ and $H\boldsymbol{\gamma}$. This formula is

$$H^* = H + \frac{(\boldsymbol{\delta} - H\boldsymbol{\gamma})(\boldsymbol{\delta} - H\boldsymbol{\gamma})^T}{\boldsymbol{\gamma}^T(\boldsymbol{\delta} - H\boldsymbol{\gamma})} \tag{3}$$

in which the correction has degenerated to be of rank 1, and which has attracted a lot of attention in recent years. The formula was discovered by a number of workers, a list of references being given by Powell (1969). Although the formula does remove the need to solve the linear

search subproblem, it unfortunately introduces a number of unpleasant side effects. One is that the H matrices no longer remain positive definite so that a reduction in F cannot be guaranteed in a basic algorithm. Another is that the correction is unbounded, even for quadratic functions. A particularly nasty property is that if the correction $\boldsymbol{\delta} = -H\boldsymbol{g}$ happens by chance to minimise F along $\boldsymbol{\delta}$ (surely a good choice of $\boldsymbol{\delta}$ to have made) then H^* is automatically singular (or undetermined). Thus many additions to the basic algorithm are required if the updating formula (3) is used (see Powell (1969) for instance). Even then no significant advantage in efficiency has yet been reported for this method. Thus algorithms based directly on (3) will not be considered further in this paper.

This discussion illustrates the dilemma which faces numerical analysts in attempting to improve on the VMM, in that some seemingly desirable feature has to go in order to make progress. When looking for a simple effective algorithm it is my view that retention of positive definiteness in H is important, as it ensures a reduction in the function value on each iteration. The approach that is considered therefore is that of abandoning the property of quadratic termination in order to be able to replace the linear search process by a more simple, and hopefully more efficient, way of reducing the function at each iteration. This cannot be done with impunity because quadratic termination is a feature which guarantees the fast ultimate convergence of an algorithm. It would seem desirable to retain some guarantee that the H matrices tend to G^{-1} so that ultimate convergence for quadratic functions can be proved. A suitable property has been defined in Section 6 (where matters concerning theoretical developments have been collected) as Property 1, which requires that for quadratic functions the eigenvalues of H must tend monotonically to those of G^{-1} in a certain sense. In this section an important theorem is then proved (Theorem 1) showing that the updating formula (1) of the VMM satisfies Property 1, and thus becomes a candidate for use in the sort of algorithm which is envisaged.

The abandonment of linear searches also requires that something is done to force a sufficiently large decrease in F at each iteration to guarantee ultimate convergence. However to assure the efficiency of an algorithm without linear searches, it is necessary that only one evaluation of F and $\boldsymbol{g}$ be made on each iteration, except on rare occasions. The change ΔF in F on an iteration would be expected by Taylor's series to be approximately $\boldsymbol{g}^T\boldsymbol{\delta}$ when $\boldsymbol{\delta}$ is small, but much less than $\boldsymbol{g}^T\boldsymbol{\delta}$ in absolute value when the position of the minimum along a line is overestimated. The change in F relative to $\boldsymbol{g}^T\boldsymbol{\delta}$ cannot become arbitrarily small if $\Delta F/\boldsymbol{g}^T\boldsymbol{\delta} \geqslant \mu$ where $0 < \mu \ll 1$ is a preassigned small quantity. In fact if corrections are determined by $\boldsymbol{\delta} = -\lambda H\boldsymbol{g}$, then trying values of $\lambda = 1, w, w^2, w^3, \ldots (0 < w < 1)$ will eventually produce a $\boldsymbol{\delta}$ which satisfies this test. In practice choices of $\mu = 0{\cdot}0001$ and $w = 0{\cdot}1$ have been made. Goldstein and Price (1967) show that if $H \to G^{-1}$ when converging to the minimum of a non-quadratic function, then eventually the choice $\lambda = 1$ will always be taken, and convergence will be superlinear. Although $H \to G^{-1}$ in their sense is not proved, it is expected that this behaviour will occur in most cases and so the efficiency of the method in eventually requiring only one evaluation of F and $\boldsymbol{g}$ per iteration is justified. This has been amply borne out in practical cases. To ensure convergence to a solution it is also requisite that the step length λ does not tend to zero, a sufficient condition for which is shown by Goldstein and Price to be that $\Delta F/\boldsymbol{g}^T\boldsymbol{\delta} \leqslant 1-\mu$. However the way in which λ is chosen, and the relative magnitudes of w and μ ensure that this condition holds except in pathological cases. Although convergence for any general class of non-quadratic functions is not proved, it will be noted that convergence for quadratic functions is an immediate consequence of Property 1 and the way in which λ is chosen.

Another property which is important is the retention of strict positive definiteness by the approximating matrices. Although this follows from Property 1 when the functions are quadratic, for non-quadratic functions it is necessary to update over an interval $\boldsymbol{\delta}$ for which $\boldsymbol{\delta}^T\boldsymbol{\gamma} > 0$. In fact, although $\boldsymbol{\delta} = -\lambda H\boldsymbol{g}$ where $\lambda = 1$ may not satisfy this condition, a $\boldsymbol{\delta}$ which does can be found by looking at values of λ larger than 1, so long as the function is bounded below.

None of the conditions which are imposed in the two previous paragraphs are very restrictive, and they are simple to program. An algorithm based on these ideas was implemented using the original VMM formula (1) to effect the updating of H, on account of it possessing Property 1. Although the results of limited tests were very encouraging, one example was found in which failure occurred because the H matrix became singular. This tendency is reported by others in isolated cases with the original VMM, and it seems that removal of the linear search does not remove this tendency and in all probability exacerbates it. It is necessary therefore that if these ideas are to be used, a more satisfactory means of updating H must be found, still possessing Property 1, but without the tendency to singularity exhibited by (1) alone.

2. A new formula

At about this time in the research, the possibility of generating a new updating formula was realised, based upon a very simple idea. The updating formula (1) is one which forces the relationship $H^*\boldsymbol{\gamma} = \boldsymbol{\delta}$ to hold. If Γ is defined as H^{-1}, then Γ and Γ^* corresponding to the H and H^* of (1) would be related by

$$\Gamma^* = \left(I - \frac{\boldsymbol{\gamma}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}}\right)\Gamma\left(I - \frac{\boldsymbol{\delta}\boldsymbol{\gamma}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}}\right) + \frac{\boldsymbol{\gamma}\boldsymbol{\gamma}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} \tag{4}$$

a formula which naturally forces $\Gamma^*\boldsymbol{\delta} = \boldsymbol{\gamma}$. Thus a way is obtained of forcing a mapping of $\boldsymbol{\delta}$ into $\boldsymbol{\gamma}$. Now by carrying out the simple interchange $\boldsymbol{\delta} \leftrightarrow \boldsymbol{\gamma}$ in (4), a formula would be obtained which mapped $\boldsymbol{\gamma}$ into $\boldsymbol{\delta}$, and this could be used as a formula to update H thus

$$\begin{aligned} H^* &= \left(I - \frac{\boldsymbol{\delta}\boldsymbol{\gamma}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}}\right)H\left(I - \frac{\boldsymbol{\gamma}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}}\right) + \frac{\boldsymbol{\delta}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} \qquad (5) \\ &= H - \frac{\boldsymbol{\delta}\boldsymbol{\gamma}^T H}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} - \frac{H\boldsymbol{\gamma}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} + \left(1 + \frac{\boldsymbol{\gamma}^T H\boldsymbol{\gamma}}{\boldsymbol{\delta}^T\boldsymbol{\gamma}}\right)\frac{\boldsymbol{\delta}\boldsymbol{\delta}^T}{\boldsymbol{\delta}^T\boldsymbol{\gamma}} \end{aligned}$$

A new formula is thus obtained in a way which could also be used in other circumstances. A particularly nice property is that if H is updated by (5), then the corresponding updating formula for Γ is obtained by per-

forming the interchange $\gamma \leftrightarrow \delta$ in (1), to give

$$\Gamma^* = \Gamma + \frac{\gamma\gamma^T}{\delta^T\gamma} - \frac{\Gamma\delta\delta^T\Gamma}{\delta^T\Gamma\delta} \tag{6}$$

Thus the formulae (1), (4) and (5), (6) may be considered as *dual* in this sense. Another important result, proved in Theorem 2 (Section 6), is that the new formula possesses Property 1, showing that it can be used to minimise quadratic functions in the sort of algorithm discussed in the previous section. The same requirement ($\gamma^T\delta > 0$) will also guarantee positive definiteness in the general case (see Section 3).

Of course the new formula could be used in a conventional way with linear searches. In this case it is also possible to prove quadratic termination. The reason that I do not do this is that I have heard very recently from C. G. Broyden in a private communication, that he has also come across this formula in a different way, has also proved quadratic termination and other similar properties, has compared it numerically against the original algorithm, and is publishing his results concurrently (Broyden (1970, Parts I, II)). This paper will only be concerned with the properties of the formula in an algorithm not requiring linear searches.

3. A convex class of formulae

An obvious extension of the two formulae (1) and (5) is to generate other formulae by taking any linear combination of the right-hand sides of (1) and (5) such that the coefficients sum to unity. Denoting the H^* in (1) and (5) by H_0^* and H_1^* respectively, then a matrix H_ϕ^* can be determined from

$$H_\phi^* = (1 - \phi)\, H_0^* + \phi H_1^* \tag{7}$$

Substituting for H_0^* and H_1^* gives a whole class of formulae generated by the single parameter ϕ. The corrections to H all lie in the space spanned by δ and $H\gamma$ and are therefore of rank 2. In fact analysis shows that this class of formulae is related directly to that class identified by Broyden (1967), based on a parameter β, through the relationship $\phi = \beta\delta^T\gamma$.

An important new result is that (7) can be rearranged as

$$H_\phi^* = H_0^* + \phi vv^T \tag{8}$$

where $$v = (\gamma^T H\gamma)^{1/2}\left\{\frac{\delta}{\delta^T\gamma} - \frac{H\gamma}{\gamma^T H\gamma}\right\}.$$

This shows that the difference between any two formulae in the class is of rank one, and also that the difference is orthogonal to γ. In particular the rank one property enables a number of useful results to be obtained by invoking Lemma 1 of section 6.

In the present context, most interest lies in the possession or otherwise of Property 1 by these formulae. The important Theorem 3 shows that any formula for which $\phi \in [0, 1]$ will possess this property. Furthermore Theorem 4 shows that however close ϕ is taken outside the range [0, 1], for example $\phi = -\epsilon$ or $\phi = 1 + \epsilon$ for small $\epsilon > 0$, then a quadratic function exists for which Property 1 does not hold, and for which H can diverge from G^{-1}. Thus Property 1 can only be guaranteed if the parameter ϕ is a convex combination of 0 and 1, and so the importance of the *convex class of formulae*, given by (7) with $\phi \in [0, 1]$, is established. Of course (1) and (5) are the extreme points in this class.

Another useful result obtained directly from (8) and Lemma 1, is that if the eigenvalues λ, λ' of any two matrices H_ϕ^* and $H_{\phi'}^*$ are arranged in decreasing order, then $\phi > \phi'$ implies $\lambda \geqslant \lambda'$ for any corresponding pair of eigenvalues. One result which follows from the discussion in Section (1) about the VMM formula, and from this result, is that the matrices H_ϕ^* retain positive definiteness if $\phi \geqslant 0$. Another is that the determinant, the L_2 norm, and the Frobenius norm of H_ϕ^* are all greater than or equal to the corresponding quantities for $H_{\phi'}^*$.

In particular H_1^* is 'less singular' than H_0^*, indicating that use of H_1^* in a variable metric algorithm might counteract the tendency to singularity mentioned in Section 1. However it may well be that use of H_1^* alone might cause H to tend to become unbounded. The flexibility given by a whole convex class of formulae possessing Property 1 is obviously important, indicating that algorithms which choose ϕ in a systematic way at each iteration to counter both singularity and unboundedness might be determined. For instance one possibility would be to choose ϕ so that the product of the traces $\mathrm{tr}(H_\phi^*)\mathrm{tr}(\Gamma_\phi^*)$ is minimised, this being an upper bound on the condition number $||H_\phi^*||_2||\Gamma_\phi^*||_2$ of H^*. Although appealing, one reason for not using this might be that the successive updating of $\mathrm{tr}(\Gamma)$ would lead to excessive accumulation of round-off error. With this in mind, the next section presents some analysis leading to another attractive and even more simple means of choosing an appropriate ϕ.

4. The rank one formula

The rank one formula (3) is also a member of the general class of formulae (7) because the rank one correction is in the space spanned by δ and $H\gamma$. In fact it can be identified as the formula $H_{\hat{\phi}}$ where $\hat{\phi} = \delta^T\gamma/(\delta^T\gamma - \gamma^T H\gamma)$. A significant observation which arises from this is that if H is positive definite, and $\delta^T\gamma$ is restricted to be strictly positive (as in section 1), then $\hat{\phi}$ *cannot* lie in the range [0, 1]. Thus the rank one formula cannot be a member of the convex class of the previous section. In fact if $\delta^T\gamma > \gamma^T H\gamma$, then $\hat{\phi} > 1$; and if $\delta^T\gamma < \gamma^T H\gamma$ then $\hat{\phi} < 0$. The rank one formula plays an interesting role in that it is a formula which does not restrict the eigenvalues of H^* in any way, and thus permits an approximation to any G^{-1} as closely as possible. The position of $\hat{\phi}$ in the range $(-\infty, \infty)$ could therefore be taken as an indication of which member of the convex class to use. That is if $\hat{\phi} > 1$, then use of H_1^* is indicated, and if $\hat{\phi} < 0$ then H_0^*. This test (i.e. whether $\delta^T\gamma \geqslant \gamma^T H\gamma$ or not) has been used with some success as described in the next section. However more detailed examination of it, when the function is quadratic, shows an even more simple interpretation. On replacing δ by $G^{-1}\gamma$ then the relation becomes $\gamma^T G^{-1}\gamma \geqslant \gamma^T H\gamma$. If true the indication is that H is 'smaller' than G^{-1} in this sense, so that the 'larger' formula H_1^* is used. If however $\gamma^T G^{-1}\gamma < \gamma^T H\gamma$ then H is 'larger' than G^{-1} and so the 'smaller' formula H_0^* is used. In fact if equality holds then no indication is given of which formula to use; in this case it was decided to use H_1^* on account of the importance of avoiding singularity in H.

Before leaving this section another result will be noted. This is that if the formula for updating the inverse Γ of the matrix H in (3) is considered, and if the interchanges $\Gamma \leftrightarrow H$ and $\boldsymbol{\delta} \leftrightarrow \boldsymbol{\gamma}$ are performed, then the original formula (3) is restored. Thus the rank one formula is *self dual* in the sense of Section 2.

5. An algorithm

An algorithm based upon choosing corrections as in Section 1, and using either updating formula (1) or (5) depending upon the test of Section 4, has been written and tested extensively. Two other additions have been made which do not materially affect the complication of the program, but contribute to efficiency. One is that if $\lambda = 1$ is not successful in reducing F sufficiently, then a λ determined by cubic interpolation would be calculated, and the largest of this value and 0·1 would be used. Thus the possibility of obtaining a local minimum along a line is retained when it is necessary for other reasons to use more than one evaluation of F and $\boldsymbol{g}$ in an iteration. This feature also helps to inject an element of independence into the successive $\boldsymbol{\delta}$ and thus to improve the adequacy of the approximation to H.

Another detail is included because the algorithm can be inefficient if H is chosen initially to be much greater than the local G^{-1}. Then any $\boldsymbol{\delta}$ which reduces F would be considerably less than $-H\boldsymbol{g}$ and a considerable number of extra function evaluations would be required at each iteration. This only occurs at up to and including the nth iteration, after which a step of $-H\boldsymbol{g}$ is almost always successful. One possibility which was not adopted, mainly because of the problems of guaranteeing positive definiteness, is to set up H by differences. In practice a running step length λ has been kept, derived from the value used on the previous iteration, and used to generate an initial $\boldsymbol{\delta}$ of $-\lambda H\boldsymbol{g}$. However the program reverts to the basic algorithm of Section 1 after the nth step. In a similar vein, the program uses a lower bound $\hat{F}$ on the least value of F to estimate a suitable step length on the basis of a quadratic function passing through the current point. This step length, if less than that above, is used in preference to it.

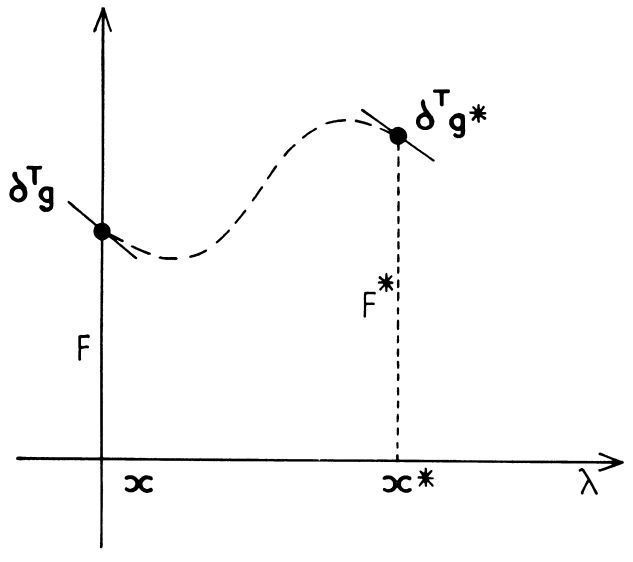

Fig. 1

The algorithm is terminated when $|\boldsymbol{\delta}| < \boldsymbol{\varepsilon}$ a tolerance vector, which was 0·00005 in all tests. There are also two error exits, which might occur either by the tolerance being set too small relative to the effect of round-off, or by incorrect programming of gradients. One is an exit when $\boldsymbol{\delta}^T\boldsymbol{g} \geqslant 0$; the other an exit when for some $\boldsymbol{\delta}$, $F^* > F$ and $\boldsymbol{\delta}^T\boldsymbol{g}^* < 0$. This latter situation is illustrated in **Fig. 1** and can be caused either by an unusual variation of the function on an early iteration or by rounding error in the region of the solution. To cater for both cases, an error exit is only taken in this situation if the iteration count is also greater than n.

A FORTRAN subroutine has been written embodying all these ideas. The testing has been done on two types of function; problems in relatively few variables which might prove difficult to minimise (including Rosenbrock's (1960) parabolic valley, Powell's (1962) function of 4 variables, Wood's function of 4 variables (Colville, 1968) and Fletcher's, (1965) Chebyquad functions for $n = 2$, 4, 6 and 8); also possibly more simple problems in larger numbers of variables, in particular Fletcher and Powell's (1963) trigonometric functions for $n = 2(2)10$, 20, 30, 40 and 60. The results of these tests are given in **Table 1**. In all cases the problems have been solved satisfactorily, excepting possibly that the full accuracy asked for has not always been obtained, a feature

Table 1

Comparative testing of the new algorithm

PROBLEM n	NEW ALGORITHM		FLEPOMIN	
	ITERATIONS	EVALUATIONS	ITERATIONS	EVALUATIONS
Parabolic Valley				
2	39	47	22	64
Powell's function				
4	42[a]	43[a]	21[a]	64[a]
Wood's function				
4	122[b]	136[b]	45	154
Chebyquad				
2	7	8	2	6
4	10	13	6	22
6	22	27	10	29
8	21	23	18	50
Trigonometric functions				
2	8	9	4	15
4	13	19	10	30
6	13	15	10	38
8	14	15	11	40
10	17[c]	18[c]	17[c]	58[c]
20	50	51	32	93
30	71[a]	75[a]	41[a]	132[a]
40	99[a]	102[a]	58[a]	160[a]
60	133[a]	149[a]	85[a]	245[a]

a Full accuracy not obtained—see text.
b Double length—see text.
c FLEPOMIN and new algorithm find a different solution.

discussed later. Another test was to consider the failing case of the version of algorithm using only formula (1) as described in Section 1. The quantities x, g and H (nearly singular) were taken from an iteration near to where the algorithm stopped, and the new algorithm was started with this data. The new algorithm was completely successful in rescuing the situation and causing convergence to the correct solution.

Comparative testing of the new algorithm has been done against a FORTRAN version of the ALGOL **procedure** FLEPOMIN (Wells, 1965; Fletcher, 1966), in which a few small errors have been corrected, and which is an implementation of the original VMM. The results are again given in Table 1 and it is immediately obvious that the new algorithm gives uniformly better performance than FLEPOMIN, as measured by the number of evaluations of F and g. One particularly significant feature is that the improvement is most noticeable as n becomes large, a most encouraging result. Although the number of iterations of the new algorithm is about 50% higher, the average of little more than one function evaluation per iteration tips the overall scale considerably in favour of the new algorithm.

The only cases in which the requested accuracy has not been realised have been common to both the new algorithm and FLEPOMIN and are not considered serious. On Powell's function, although an error exit due to round-off is taken, this is caused by the slow rate of linear convergence to the solution which occurs on account of the singular Hessian there. On the trigonometric functions it is due solely to the effect of round-off error which becomes serious for $n \geqslant 30$. This is not surprising however in that the calculations have been carried out on an IBM 360/65 with only 6–7 significant decimal digits. A different fault arises with Wood's function. Here there is a region remote from the solution which is nearly a stationary point and which causes small steps to be taken for a number of iterations. With the new algorithm one of these steps was sufficiently small as to cause the ill-definition of F and g typified by Fig. 1. In fact both algorithms were taking steps of magnitude about 0·00005 in this region. When the new algorithm was run in double length with slightly smaller tolerance, then no trouble was obtained. However one particularly encouraging feature of the single length run was that the eigenvalues of the H matrix showed no tendency to singularity as the stationary point was neared, and is a result which gives confidence in the way in which the matrix is updated.

Another observation which also gives rise to confidence in the new algorithm occurred as the program was being developed, and small changes in strategy or different values of parameters (μ, w, etc.) were tried. The experience with the new algorithm has been that such changes have caused very little variation in the total run times. This is taken as a very encouraging feature—firstly in that it suggests that the algorithm is robust, and secondly in that it seems likely that the improvements shown in Table 1 are representative of what can be obtained in practice and are not just caused by a freak choice of adjustable parameters.

To sum up then, a variable metric algorithm has been developed in which the linear search sub-problem is no longer necessary. The property of quadratic termination, whose relevance for general functions has always been questionable, has been replaced by a property in which the approximating matrix H has to tend to the inverse hessian G^{-1} in a certain sense. A convex class of formulae for updating H, which has this property, has been established; and a strategy has been described which indicates how to choose a member of the class. A FORTRAN program has been tested extensively and the results indicate that the improvements which had been hoped for have been realised.

6. Some theorems

Lemma 1: *If symmetric matrices A and A^* differ by a matrix of rank* 1, *then their eigenvalues* $\boldsymbol{\lambda}$ *and* $\boldsymbol{\lambda}^*$ *interpolate each other in a weak sense.*

In particular, if $A^* = A + \sigma \boldsymbol{v}\boldsymbol{v}^T$ where $\sigma = \pm 1$, and if $\lambda_1 \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_n$, and if $\lambda_1^* \geqslant \lambda_2^* \geqslant \ldots \geqslant \lambda_n^*$, then

(i) if $\sigma = +1$, $\lambda_1^* \geqslant \lambda_1 \geqslant \lambda_2^* \geqslant \lambda_2 \geqslant \ldots \geqslant \lambda_n^* \geqslant \lambda_n$

(ii) if $\sigma = -1$, $\lambda_1 \geqslant \lambda_1^* \geqslant \lambda_2 \geqslant \lambda_2^* \geqslant \ldots \geqslant \lambda_n \geqslant \lambda_n^*$

Proof: See Wilkinson (1965), pp. 94–98.

Property 1: A measure of the difference between the approximating matrix H and any inverse hessian G^{-1} is given by the difference between the $K = G^{1/2}HG^{1/2}$ and the unit matrix (see also Broyden, 1970). This is equivalent to transforming the problem so that the inverse hessian in question is a unit matrix, and K is the approximation to it.

Definition: An updating formula for positive definite matrices possesses Property 1 *if, for a quadratic function with G strictly positive definite, the eigenvalues of K (arranged in order) tend monotonically to* 1 *for any sequence of vectors* $\boldsymbol{\delta}$ *(but not necessarily strictly monotonically.)*

Property 1 implies that the L_2 norms of both K and K^{-1} tend monotonically to 1. It also implies that both the L_2 and Frobenius norms of both $K - I$ and $K^{-1} - I$ tend monotonically to zero. Note however that although convergence to these limits is likely, only non-divergence is actually implied by Property 1.

Reference to the positive definiteness of H and H^* has already been made. This implies the positive definiteness of K and K^* if G is non singular, and vice versa.

Theorem 1: *The formula* (1) *possesses Property* 1.

Proof: Making the substitution $z = G^{1/2}\boldsymbol{\delta}$ and using $\boldsymbol{\gamma} = G\boldsymbol{\delta} = G^{1/2}z$, then (1) can be rearranged as

$$K_0^* = K - \frac{Kzz^TK}{z^TKz} + \frac{zz^T}{z^Tz}$$

Now $\bar{K} = K - Kzz^TK/z^TKz$ has one zero eigenvalue (eigenvector z) and by Lemma 1, its other eigenvalues interpolate those of K in the weak sense. Addition of the term zz^T/z^Tz changes the zero eigenvalue of $\bar{K}$ to a unit eigenvalue and leaves the others unchanged. Now each non-zero eigenvalue of $\bar{K}$ (and hence of K^*) lies in a closed interval of two eigenvalues of K and so can be paired up with one which is more remote from 1 but is the same side of 1. The remaining eigenvalue of K can be paired with the unit eigenvalue of K^*. This pairing up ensures the monotonic tendency of eigenvalues to 1 and so proves the theorem. The situation is illustrated in **Fig. 2**.

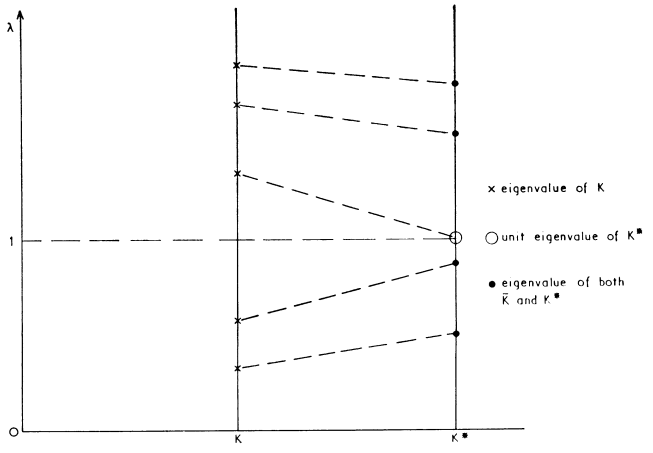

Fig. 2

Theorem 2: *The formula* (5) *possesses Property* 1.
Proof: If H_1^* is obtained from (5), then its inverse is obtained from (6) by virtue of the duality. Define $M = G^{-1/2}\Gamma G^{-1/2}$, whence M is updated by

$$M_1^* = M - \frac{Mzz^TM}{z^TMz} + \frac{zz^T}{z^Tz}$$

Thus the formula for changing M satisfies the equivalent of Property 1 from Theorem 2. But M is the inverse of K, so the eigenvalues of K are the inverse of those of M. Thus, because M and K are both positive definite, the updating formula (5) satisfies Property 1.
Theorem 3: *The updating formula* (7) *with* $\phi \in [0, 1]$ *satisfies Property* 1.

Proof: By Lemma 1 and by virtue of (8), each eigenvalue of K_ϕ^* cannot lie below the corresponding eigenvalue of K_0^* nor above the corresponding eigenvalue of K_1^*. The theorem follows from Theorems 1 and 2.
Theorem 4: *The updating formulae with* $\phi = -\epsilon$ *and* $\phi = 1 + \epsilon$, *where* $0 < \epsilon \ll 1$ *is a given small number, do not satisfy Property* 1.
Proof: Consider a strictly positive definite quadratic function for which

$$G = \begin{pmatrix} 1+\epsilon & \sqrt{\epsilon} \\ \sqrt{\epsilon} & \epsilon \end{pmatrix}$$

If H is taken as the unit matrix, then $K = G$. The eigenvalues of K are η and $1 + 2\epsilon - \eta$ where η is a strictly positive term of order ϵ^2. If $\boldsymbol{\delta}$ is chosen so that $z^T = (0, 1)$ then it can be verified on substitution that

$$K_{-\epsilon}^* = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad K_{1+\epsilon}^* = \begin{pmatrix} 1+2\epsilon & 0 \\ 0 & 1 \end{pmatrix}$$

Thus in both cases an eigenvalue has diverged from 1. For $K_{-\epsilon}^*$ the smallest eigenvalue has gone from η to zero, and for $K_{1+\epsilon}^*$, the largest eigenvalue has gone from $1 + 2\epsilon - \eta$ to $1 + 2\epsilon$. Thus a counter example has been produced and the theorem is proved.

7. Acknowledgement

I would particularly like to acknowledge the help of Mr. M. J. D. Powell who drew my attention to both Lemma 1 and to the equation (4), and whose advice has proved most valuable in many ways.

References

Broyden, C. G. (1967). Quasi-Newton methods and their application to function minimisation, *Maths. Comput.*, Vol. 21, p. 368.
Broyden, C. G. (1970). The convergence of a class of double-rank minimisation algorithms Parts I and II, *Jour. Inst. Maths. and its Applics.* (to be published).
Colville, A. R. (1968). A comparative study on nonlinear programming codes, IBM tech. rep. No. 320–2949.
Davidon, W. C. (1959). Variable Metric Method for minimization, Argonne Nat. Lab. report ANL–5990 Rev.
Fletcher, R. (1965). Function minimization without evaluating derivatives—A review, *Comp. J.*, Vol. 8, p. 33.
Fletcher, R. (1966). Certification of Algorithm 251, *CACM*, Vol. 9, p. 686.
Fletcher, R., and Powell, M. J. D. (1963). A rapidly convergent descent method for minimization, *Comp. J.*, Vol. 6, p. 163.
Goldstein, A. A., and Price, J. F. An effective algorithm for minimization, *Num. Math.*, Vol. 10, p. 184.
Powell, M. J. D. (1962). An iterative method for finding stationary values of a function of several variables, *Comp. J.*, Vol. 5, p. 147.
Powell, M. J. D. (1964). An efficient method of finding the minimum of a function of several variables without calculating derivatives, *Comp. J.*, Vol. 7, p. 155.
Powell, M. J. D. (1969). Rank one methods for unconstrained optimization, AERE Report T.P.372, to appear in J. Abadie (*ed.*), North Holland (1970).
Rosenbrock, H. H. (1960). An automatic method for finding the greatest or least value of a function, *Comp. J.*, Vol. 3, p. 175.
Wells, M. (1965). Algorithm 251, Function minimisation, *CACM*, Vol. 8, p. 169.
Wilkinson, J. H. (1965). *The algebraic eigenvalue problem*, O.U.P., London.