J. Inst. Maths Applics (1970) **6**, 222–231

The Convergence of a Class of Double-rank Minimization Algorithms
2. The New Algorithm

C. G. BROYDEN

Computing Centre, University of Essex,
Wivenhoe Park, Colchester, Essex

[Received 18 August 1969 and in revised form 30 September 1969]

This paper presents a new minimization algorithm and discusses theoretically some of its properties when applied to quadratic functions. Results of comparative testing for a set of non-quadratic functions are described and reasons for the observed experimental behaviour are suggested.

1. Introduction

IN THE FIRST PART of this paper (Broyden, 1970) we examined a class of algorithms for minimizing the function $F(\mathbf{x})$ where it was assumed that the gradient of $F(\mathbf{x})$, denoted by $\mathbf{f}(\mathbf{x})$, is available as an explicit expression. In these algorithms the function $F(\mathbf{x})$ is minimized at the ith iteration in the direction $\mathbf{p}_i$, where $\mathbf{p}_i$ is given by

$$\mathbf{p}_i = -\mathbf{H}_i\mathbf{f}_i \tag{1.1}$$

and where $\mathbf{f}_i$ is the value of the gradient at $\mathbf{x}_i$. $\mathbf{H}_i$ is a symmetric and preferably positive definite matrix. The matrix $\mathbf{H}_i$ is updated at each iteration, and the updating equation involved an arbitrary parameter denoted in Part 1 by β_i. This updating equation is somewhat complicated, but its properties may be analysed more simply if $F(\mathbf{x})$, the function to be minimized, is quadratic. If then

$$F(\mathbf{x}) = \tfrac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x} + c, \tag{1.2}$$

where $\mathbf{A}$ is an nth order positive definite matrix, $\mathbf{b}$ an nth order vector and c a constant, we can define a matrix $\mathbf{K}_i$ by

$$\mathbf{K}_i = \mathbf{B}\mathbf{H}_i\mathbf{B} \tag{1.3}$$

where $\mathbf{B}$ is the positive definite matrix that satisfies

$$\mathbf{B}^2 = \mathbf{A}. \tag{1.4}$$

Now we are concerned in this paper only with changes that occur during a single iteration so we may, in order to simplify notation, omit the subscript i and replace the subscript $i+1$ by 1. With this convention it was shown in Part 1 that $\mathbf{K}$ is updated according to the equation

$$\mathbf{K}_1 = \mathbf{K} + [\mathbf{q} \quad \mathbf{q}_1]\begin{bmatrix}\omega & \xi\\ \xi & \eta\end{bmatrix}\begin{bmatrix}\mathbf{q}^T\\ \mathbf{q}_1^T\end{bmatrix} \tag{1.5a}$$

where

$$\omega = 1 - \mathbf{q}^T\mathbf{K}\mathbf{q}, \tag{1.5b}$$

$$\xi = -\mathbf{q}_1^T\mathbf{K}\mathbf{q} \tag{1.5c}$$

and where $\mathbf{q}$ and $\mathbf{q}_1$ are uniquely determined orthonormal vectors. The parameter η is essentially arbitrary in that it depends upon β. It was suggested in Part 1 that a suitable choice for η would be zero since if it were negative, or large and positive, the matrix $\mathbf{K}_1$ and hence $\mathbf{H}_1$ might become needlessly badly conditioned. It was noted moreover that choosing η in this way gives rise to a new algorithm.

Of the two algorithms in this class already published, that due to Davidon (1959) and modified by Fletcher & Powell (1963) is obtained by putting β equal to zero and it was shown in Part 1 that this led, in general, to negative values of η. We thus expect the sequence of matrices $\{\mathbf{H}_i\}$ obtained by that algorithm to exhibit a tendency to singularity and this tendency has been noted by, among others, Broyden (1967) and Pearson (1969). In a more recent algorithm, due to Greenstadt (1967), if $\mathbf{H}$ is positive definite the values of η are even more negative than those occurring in the DFP algorithm. One result of this is that for this algorithm the matrices $\mathbf{H}$ cannot, unlike those for the DFP algorithm, be proved to be positive definite and this has serious implications when considering numerical stability.

In this paper we show theoretically that the new algorithm is stable and we prove that it is the only member of the class considered for which a certain matrix error norm is reduced strictly monotonically when minimizing quadratic functions. We discuss the effect of rounding and of poor conditioning of $\mathbf{H}$ on the attainable accuracy of the final solution and conclude by presenting the results of a numerical survey in which the performance of the new algorithm for a variety of test problem is compared with that of the DFP algorithm.

2. The New Algorithm

The new algorithm is obtained by setting η equal to zero and it then follows from Part 1, equations (6.5) and (6.10a), that in order to achieve this β must be chosen to satisfy

$$\beta t\mathbf{z}^T\mathbf{K}\mathbf{z} = 1. \tag{2.1}$$

This equation becomes, from Part 1, equations (2.5), (3.1), (3.4) and (3.5),

$$\beta = -1/(t\mathbf{p}^T\mathbf{f}) \tag{2.2}$$

which is equivalent, from equations (2.8) and (3.2b) of Part 1, to

$$\beta = 1/(t\mathbf{p}^T\mathbf{y}). \tag{2.3}$$

If this value of β is substituted into the general matrix updating equation (Part 1, equation (3.2)) we obtain the updating equation of the new algorithm in a form suited to computation,

$$\mathbf{H}_1 = \mathbf{H} + \frac{1}{\mathbf{p}^T\mathbf{y}}(\mathbf{p}\rho\mathbf{p}^T - \mathbf{p}\mathbf{y}^T\mathbf{H} - \mathbf{H}\mathbf{y}\mathbf{p}^T) \tag{2.4a}$$

where

$$\rho = t + \frac{\mathbf{y}^T\mathbf{H}\mathbf{y}}{\mathbf{p}^T\mathbf{y}}. \tag{2.4b}$$

We now prove that the new algorithm is stable in the sense of Broyden (1967).

THEOREM 1. *If t is chosen to minimize $F(\mathbf{x})$, $\mathbf{H}$ is positive definite and $\mathbf{H}_1$ is given by equation* (2.4) *then $\mathbf{H}_1$ is positive definite.*

Proof. Since β is given by equation (2.2) it follows from equation (1.1) that

$$\beta = 1/(t\mathbf{f}^T\mathbf{H}\mathbf{f}). \tag{2.5}$$

But since $\mathbf{H}$ is assumed to be positive definite both $\mathbf{f}^T\mathbf{H}\mathbf{f}$ and t are positive (for proof of the latter assertion see, e.g. Broyden, 1967). Thus $\beta > 0$ and the proof follows from Broyden (1967), Theorem 8.

It follows from this theorem that the matrix update of the new algorithm is always well-defined, and hence suitable for automatic computation. Since the algorithm in addition minimizes a quadratic function in at most n iterations it possesses in theory all the properties that made the DFP algorithm so successful. Our final theorem shows that the new algorithm possesses a property not possessed by the DFP algorithm, and although it is strictly relevant only to quadratic functions it is likely, since it involves matrix error norms, to have a bearing on the more general case. Expressed crudely the final theorem states that of all the double rank algorithms considered in Part 1, the new algorithm approximates most closely to Newton's method. We first prove a lemma.

LEMMA 1. If $\mathbf{M} = [\mathbf{m}_{ij}]$ is a 2×2 matrix, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2]$ and $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2]$ where $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{y}_1$ and $\mathbf{y}_2$ are nth order vectors, then

$$\mathrm{Tr}(\mathbf{XMY}^T) = \mathbf{m}_{11}\mathbf{y}_1^T\mathbf{x}_1 + \mathbf{m}_{12}\mathbf{y}_2^T\mathbf{x}_1 + \mathbf{m}_{21}\mathbf{y}_1^T\mathbf{x}_2 + \mathbf{m}_{22}\mathbf{y}_2^T\mathbf{x}_2.$$

Proof. On expanding the matrix product we obtain

$$\mathrm{Tr}(\mathbf{XMY}^T) = \mathrm{Tr}(\mathbf{x}_1\mathbf{m}_{11}\mathbf{y}_1^T + \mathbf{x}_1\mathbf{m}_{12}\mathbf{y}_2^T + \mathbf{x}_2\mathbf{m}_{21}\mathbf{y}_1^T + \mathbf{x}_2\mathbf{m}_{22}\mathbf{y}_2^T)$$

and the lemma follows from the identities

$$\mathrm{Tr}(\mathbf{uv}^T) \equiv \mathbf{v}^T\mathbf{u} \tag{2.6}$$

and

$$\mathrm{Tr}(\mathbf{A}+\mathbf{B}) \equiv \mathrm{Tr}(\mathbf{A}) + \mathrm{Tr}(\mathbf{B}). \tag{2.7}$$

THEOREM 2. *Let the generalized algorithm as defined in Part* 1 *be applied to the quadratic function* (1.2) *and let* $\mathbf{E}$ *be given by*

$$\mathbf{E} = \mathbf{K} - \mathbf{I}, \tag{2.8}$$

where $\mathbf{K}$ *is defined by equation* (1.3) *and* (1.4). *Then at least one of the following two possibilities occurs:*

(a) $$\mathbf{e}_1 = 0, \tag{2.9a}$$

where $\mathbf{e}$ *is the difference between the solution and the* i*th approximation, and*

(b) $$\|\mathbf{E}_1\|^2 < \|\mathbf{E}\|^2 + \eta(\eta + 2\mathbf{q}_1^T\mathbf{E}\mathbf{q}_1), \tag{2.9b}$$

where $\|.\|$ denotes the Euclidean matrix norm.

Proof. Equations (1.5) and (2.8) give

$$\mathbf{E}_1 = \mathbf{E} - \mathbf{QMQ}^T \tag{2.10}$$

where

$$\mathbf{Q} = [\mathbf{q}, \mathbf{q}_1] \tag{2.11a}$$

and

$$\mathbf{M} = [\mathbf{m}_{ij}] = \begin{bmatrix} \mathbf{q}^T\mathbf{E}\mathbf{q} & \mathbf{q}^T\mathbf{E}\mathbf{q}_1 \\ \mathbf{q}_1^T\mathbf{E}\mathbf{q} & -\eta \end{bmatrix}. \tag{2.11b}$$

Now $\|\mathbf{E}_1\|^2 = \mathrm{Tr}(\mathbf{E}_1^T\mathbf{E}_1)$ and it then follows from symmetry and the orthonormality of $\mathbf{q}$ and $\mathbf{q}_1$ that

$$\|\mathbf{E}_1\|^2 = \mathrm{Tr}(\mathbf{E}^2 - \mathbf{EQMQ}^T - \mathbf{QMQ}^T\mathbf{E} + \mathbf{QM}^2\mathbf{Q}^T). \tag{2.12}$$

Applying the lemma to equation (2.12) then gives, since $\mathbf{M}$ is symmetric,

$$\|\mathbf{E}_1\|^2 = \|\mathbf{E}\|^2 - 2(\mathbf{m}_{11}\mathbf{q}^T\mathbf{E}\mathbf{q} + \mathbf{m}_{12}\mathbf{q}_1^T\mathbf{E}\mathbf{q} + \mathbf{m}_{21}\mathbf{q}^T\mathbf{E}\mathbf{q} + \mathbf{m}_{22}\mathbf{q}_1^T\mathbf{E}\mathbf{q}_1) + \mathbf{m}_{11}^2 + 2\mathbf{m}_{12}^2 + \mathbf{m}_{22}^2$$

which reduces from equation (2.11b) to

$$\|\mathbf{E}_1\|^2 = \|\mathbf{E}\|^2 - (\mathbf{q}^T\mathbf{E}\mathbf{q})^2 - 2(\mathbf{q}^T\mathbf{E}\mathbf{q}_1)^2 + \eta(\eta + 2\mathbf{q}_1\mathbf{E}\mathbf{q}_1). \tag{2.13}$$

We now consider the two possibilities:

(1) $$\mathbf{Kq} = \mathbf{q}\lambda \tag{2.14}$$

where λ is some scalar, non-zero since $\mathbf{K}$ is non-singular. This implies, from the definition of $\mathbf{q}$, that

$$\mathbf{K}^2\mathbf{z} = \mathbf{Kz}\lambda. \tag{2.15}$$

Now Part 1, equation (3.6), states that

$$\mathbf{z}_1 = \mathbf{z} - \mathbf{Kz}t$$

but it follows from Part 1, equation (3.8) and equation (2.15) (above) that $t = \lambda^{-1}$, so that

$$\mathbf{z}_1 = \mathbf{z} - \mathbf{Kz}\lambda^{-1}. \tag{2.16}$$

But since $\mathbf{K}$ is non-singular equation (2.15) may be pre-multiplied by $\mathbf{K}^{-1}$ and this gives, from equation (2.16), $\mathbf{z}_1 = 0$. The first alternative of the theorem thus occurs.

(2) $$\mathbf{Kq} \neq \mathbf{q}\lambda. \tag{2.17}$$

We prove that, in this case, $\mathbf{q}^T\mathbf{E}\mathbf{q}_1 \neq 0$. It was shown in Part 1 (equation (4.4)) that

$$\mathbf{K}_{i+1}\mathbf{z}_{i+1} = \mathbf{K}_i\mathbf{z}_i\phi_i - \mathbf{K}_i^2\mathbf{z}_i\psi_i,$$

where ϕ_i and ψ_i are scalars, and it was moreover established that $\psi_i \neq 0$.

From the definition of $\mathbf{q}_i$ and the subscript convention this equation may then be written

$$\mathbf{Kq} = \mathbf{q}\alpha + \mathbf{q}_1\beta \tag{2.18}$$

and it follows immediately from inequality (2.17) that $\beta \neq 0$. Pre-multiplication of equation (2.18) by $\mathbf{q}_1^T$ gives, from equation (2.8),

$$\mathbf{q}_1^T\mathbf{E}\mathbf{q} = \beta$$

so that $\mathbf{q}_1^T\mathbf{E}\mathbf{q} \neq 0$. The theorem then follows immediately from equation (2.13) and the symmetry of $\mathbf{E}$.

Corollary. In the new algorithm at least one of the following two possibilities occurs:

(a) $\mathbf{e}_1 = 0$,

(b) $\|\mathbf{E}_1\| < \|\mathbf{E}\|$.

Proof. Put $\eta = 0$ in inequality (2.9b).

The relevance of this theorem and its corollary is due to the fact that for quadratic functions the norms of the successive errors $\mathbf{e}_i$ are bounded in terms of the spectral norm of $\mathbf{E}$ (Part 1, Theorem 8), and although for quadratic functions the vector errors are independent of the successive values of β_i, for non-quadratic functions it may well be advisable to choose these parameters with a view to the greatest possible reduction

of the vector error norms. This may hopefully be achieved if the matrix error norms $\|\mathbf{E}_i\|$ are kept as small as possible, and though the spectral norms may increase when using the new algorithm the Euclidean norms are reduced strictly monotonically. It is of course possible that, for some problems, algorithms for which $\eta \neq 0$ may reduce the matrix error norms by a greater amount than the new algorithm, and we might expect the new algorithm to be inferior in these cases. On the other hand, if $\eta \neq 0$, it is possible that a very large increase in the matrix error norm might occur and in this case we would expect the use of the new algorithm to result in a substantial improvement. Since reducing the matrix error norm makes the iteration matrix look more like the inverse Jacobian we are justified to some extent in claiming that of the general class of double-rank methods, the new one is that which most resembles Newton's method. We would thus expect the performance of the new algorithm to reflect that of Newton's method, so that it might perform comparatively badly if the Jacobian matrix is singular at the solution. This did, indeed, occur in the one example attempted of this type of problem.

3. Numerical Examples

One of the principal initial difficulties in carrying out a programme of comparative testing was finding problems sufficiently difficult to reveal any significant differences between the performance of the new algorithm and the DFP algorithm. The tests commenced with four standard problems, namely Rosenbrock's, Helical Valley, Powell in four variables (for all these see Fletcher & Powell, 1963) and a problem

TABLE 1

Problem	n	N_0	ε	m(NM)	M(DFP)	s(NM)	s(DFP)
1	2	$2{\cdot}3_{10^2}$	10^{-6}	19	23	188	239
2	3	$1{\cdot}9_{10^3}$	10^{-6}	21	21	167	167
3	4	$3{\cdot}6_{10^3}$	10^{-6}	26	18	231	182
4	2	$1{\cdot}3_{10^2}$	10^{-6}	15	14	152	157

attributed to Beale (Shah, Buehler & Kempthorne, 1964). We denote these as Problems 1, 2, 3, and 4 respectively. Each problem was deemed to be solved when $\|\mathbf{f}\| < \varepsilon$, where ε is some arbitrary tolerance, and the results of the calculations are summarized in Table 1. In the tables we denote the number of independent variables by n, the total number of gradient evaluations by m and the total number of function evaluations by s. This latter figure represents the amount of labour involved in linear minimization, a process that was carried out using a well-tried quadratic interpolation algorithm to be described more fully by Fielding (to appear). The initial value of $\|\mathbf{f}\|$ is denoted by N_0 and the final value, as stated, is less than ε. We choose to consider $\|\mathbf{f}\|$ rather than the function value since the former is zero at the solution whereas in general no such statement may be made about the latter.

A glance atTable 1 reveals no significant differences between the methods except for

Problem 3, where the new method is markedly inferior. In this problem the Jacobian is singular at the solution and the remarks of the previous section apply.

Problems 5–10 were cases of the trigonometric function of Fletcher & Powell (1963). The maximum number of independent variables, 45, was dictated by the size of the available computer and it is seen from Table 2 that for these problems there is nothing to choose between the two methods.

TABLE 2

Problem	n	N_0	ε	m(NM)	m(DFP)	s(NM)	s(DFP)
5	5	$4{\cdot}89_{10^8}$	$2{\cdot}24_{10^{-5}}$	15	15	86	83
6	10	$1{\cdot}86_{10^9}$	$3{\cdot}16_{10^{-5}}$	21	21	151	145
7	20	$1{\cdot}95_{10^{10}}$	$4{\cdot}47_{10^{-5}}$	29	29	217	225
8	30	$3{\cdot}21_{10^{10}}$	$5{\cdot}48_{10^{-5}}$	46	46	350	350
9	40	$2{\cdot}62_{10^{11}}$	$6{\cdot}32_{10^{-5}}$	53	54	382	403
10	45	$1{\cdot}63_{10^{11}}$	$6{\cdot}71_{10^{-5}}$	63	63	480	499

The remaining tests were attempts to fit data by a sum of exponentials, it being thought that this would combine maximum scope for testing with minimum extra programming. If (x_i, y_i), $i = 1, 2, \ldots, p$ represents p data points we define a function S by

$$S = \sum_{i=1}^{p} \sigma_i^2 \tag{3.1}$$

where

$$\sigma_i = y_i - \sum_{j=1}^{q} \alpha_j e^{-\beta_j x_i}. \tag{3.2}$$

Explicit expressions for $\partial S/\partial \alpha_j$ and $\partial S/\partial \beta_j$ were then found and these formed the elements of the vector $\mathbf{f}$, of order $2q$. The q values of α_j and q values of β_j formed the vector of independent variables with respect to which S is minimized. Data for these tests was obtained in three ways. For Problems 11 and 12 the values of y_i were the sum of three exponentials evaluated at 13 values of x_i but for problems 13 and 14 a freehand curve of faintly exponential character was drawn, and 17 points taken from it. The values of q for Problems 11 to 14 were 1, 3, 1 and 2 respectively, the choice of three for Problem 12 being governed by the knowledge that the data did in fact represent three exponentials. The data for Problem 15 was provided by a user, who required a sum of six exponentials fitting to 54 data points. The values of y_i were obtained experimentally. The results of the five exponential problems are summarized in Table 3.

The behaviour of the DFP algorithm in these examples was extremely interesting. It appeared to get reasonably close to the solution in only a few more iterations than required by the new algorithm, and it then proceeded to "mark time" for perhaps twenty iterations or so. A characteristic example was Problem 12. After 33 iterations $\|\mathbf{f}\|$ had been reduced to approximately 10^{-3}, and it then hovered around this value

until iteration 60 when it was reduced to about 10^{-4}. Subsequent iterations then reduced $\|\mathbf{f}\|$ steadily until at the 65th iteration it fell below 10^{-6} and the program terminated. This behaviour, which was repeated in a more complex manner in Problem 14, was in marked contrast to the new algorithm which converged extremely rapidly when close to the solution.

TABLE 3

Problem	n	N_0	ε	m(NM)	m(DFP)	s(NM)	s(DFP)
11	2	$9{\cdot}9_{10^1}$	10^{-6}	9	9	90	90
12	6	$5{\cdot}7_{10^1}$	10^{-6}	33	65	404	886
13	2	$7{\cdot}9_{10^3}$	10^{-6}	11	—	148	—
14	4	$2{\cdot}1$	10^{-6}	20	61	267	912
15	12	$3{\cdot}1_{10^3}$	10^{-4}	56	more than 150	783	not recorded

One more feature of Table 3 calls for comment, the fact that the DFP algorithm did not solve Problems 13 and 15. This was due in the latter case to exceeding an arbitrary limit on the number of iterations and in the former to a value of β_j becoming negative. This resulted, for one of the larger values of x_i, in exponential overflow when evaluating σ_i.

4. Discussion

It was noted in the previous section that the performance of the new algorithm was substantially the same as that of the DFP algorithm in the initial stages of the solution of a problem, but that the characteristics of the algorithms during the final stages were markedly different. That this behaviour is not unreasonable may be inferred from a consideration of the values of β for the two algorithms. We have, in fact,

$$\beta = 0 \tag{4.1}$$

for the DFP algorithm and

$$\beta = 1/(\mathbf{f}^T\mathbf{H}\mathbf{f}t) \tag{4.2}$$

for the new one. At the beginning of the iteration, when the gradients are usually large, equation (4.2) implies that provided t is not too small then β may well approach zero, so that the two algorithms become effectively identical. In practice it has been found that initially β has usually been very close to zero (values of 10^{-4} have been recorded), as for example in Problem 9, where the values of $\mathbf{x}_i$ obtained by both algorithms were identical (to four significant figures) for the first five or so iterations.

On the other hand, as the solution is approached, β for the new method becomes extremely large (a value of 10^4 has been monitored) and the maximum discrepancy between the two methods occurs. The other eventuality that could give rise to a large value of β is severe ill-conditioning of the Hessian of $F(\mathbf{x})$. If this occurred in a region where $F(\mathbf{x})$ was strongly non-quadratic it would be possible, despite a large value of $\|\mathbf{f}\|$, for both $\|\mathbf{H}\mathbf{f}\|$ and t to be small so that in this case also the DFP algorithm would

differ markedly from the new one. Since, as was shown in Part 1, this would cause the DFP algorithm to yield a new value of $\mathbf{H}$ that would be much more badly conditioned than that given by the new algorithm, and since the occurrence of a near singular $\mathbf{H}$ would explain the observed poor performance of the DFP algorithm for badly conditioned problems, we regard the discrepancy between the two algorithms in this case as highly encouraging.

We turn now to the effect of rounding error in computing the gradient. Suppose that $\mathbf{f}$ is the true value of the gradient but the computed value is $\mathbf{f}+\Delta\mathbf{f}$. We then have, ignoring the possibility of further error,

$$\mathbf{p} = -\mathbf{H}\mathbf{f} \tag{4.3}$$

and

$$\mathbf{p}+\Delta\mathbf{p} = -\mathbf{H}(\mathbf{f}+\Delta\mathbf{f}) \tag{4.4}$$

where $\mathbf{p}$ is the correct step vector, and $\mathbf{p}+\Delta\mathbf{p}$ is the computed one. Thus, from equation (4.3) and (4.4),

$$\Delta\mathbf{p} = -\mathbf{H}\Delta\mathbf{f}. \tag{4.5}$$

Equations (4.3) and (4.5) now yield

$$\|\mathbf{f}\| \leqslant \|\mathbf{H}^{-1}\|\,\|\mathbf{p}\|$$

and

$$\|\Delta\mathbf{p}\| \leqslant \|\mathbf{H}\|\,\|\Delta\mathbf{f}\|$$

so that

$$\|\Delta\mathbf{p}\|/\|\mathbf{p}\| \leqslant k(\mathbf{H})\|\Delta\mathbf{f}\|/\|\mathbf{f}\|. \tag{4.6}$$

Thus the relative error in the computed step vector is bounded by the product of the relative error in the computed gradient multiplied by the condition number of $\mathbf{H}$. Since $\|\Delta\mathbf{f}\|/\|\mathbf{f}\|$ becomes very large as the solution is approached equation (4.5) gives a rough practical guide as to the attainable accuracy. The quantity $\|\Delta\mathbf{f}\|$ may be assessed roughly from the knowledge of the details of the computation of $\mathbf{f}$ and the word-length of the computer, and $\mathbf{f}$ of course is known. It remains therefore to estimate the condition number of $\mathbf{H}$, and a lower bound of this may be determined very simply. Consider the function ϕ defined by

$$\phi = \frac{\|\mathbf{p}\|\,\|\mathbf{f}\|}{|\mathbf{p}^T\mathbf{f}|}$$

which since $\mathbf{p} = -\mathbf{H}\mathbf{f}$, may be written

$$\phi = \frac{\|\mathbf{H}\mathbf{f}\|\,\|\mathbf{f}\|}{|\mathbf{f}^T\mathbf{H}\mathbf{f}|}.$$

Now since $\mathbf{H}$ is symmetric and positive definite its spectral norm is equal to its largest eigenvalue, say $\lambda_{\max}$. Thus

$$\|\mathbf{H}\mathbf{f}\|\,\|\mathbf{f}\| \leqslant \lambda_{\max}\|\mathbf{f}\|^2.$$

But $\mathbf{f}^T\mathbf{H}\mathbf{f} \geqslant \lambda_{\min}\|\mathbf{f}\|^2$, where $\lambda_{\min}$ is the smallest eigenvalue of $\mathbf{H}$, and combining these inequalities gives

$$\phi \leqslant \lambda_{\max}/\lambda_{\min}. \tag{4.7}$$

But $\lambda_{\max}/\lambda_{\min}$ is the condition number of $\mathbf{H}$, so that a lower bound of the condition number may readily be computed. The use of this bound, coupled with the inequality (4.6), is sufficient to give an idea of the attainable accuracy in a given case. It would be

of use, for instance, in deciding whether failure to converge was due to some peculiarity of the function or to an unreasonably small value of ε.

We do not propose to discuss the technical details of the programs here beyond saying that apart from the matrix updating routines the programs for both DFP and the new algorithm were identical. It is hoped that an ALGOL procedure embodying the new algorithm will be published in due course (Fielding, to appear) and it seems more appropriate to defer publication of the relevant computational details until then.

5. Conclusions

The experimental results quoted in Section 3 above support the view that the DFP algorithm exhibits a tendency to "mark time" for certain problems. The success of the new algorithm in overcoming this tendency in the cases tried suggests that the explanation of this behaviour given in Part 1, namely that in the DFP algorithm the matrices $\mathbf{H}_i$ are predisposed towards singularity, is probably correct. It is also clear that the strategy of choosing β to eliminate this tendency appears to have been largely successful.

Of the 15 cases attempted, in only one was the DFP algorithm significantly superior to the new algorithm and this case was special in that the Jacobian at the solution was singular. For all other cases the number of iterations required was either comparable, or substantially favoured the new algorithm. Indeed the DFP algorithm required more than three times as many iterations than the new algorithm to solve Problem 14, and this ratio could have been exceeded in Problem 15 where the DFP test was terminated after 150 iterations. The choice of ε in Problem 15 was perhaps a little unfortunate for the DFP algorithm since the value of $\|\mathbf{f}\|$ achieved by this algorithm hovered at only slightly above ε for some 85 iterations. This example did, though, confirm the apparent inability of the DFP algorithm to administer the *coup de grâce* to a difficult problem.

Further perusal of the tables shows that not only is the new algorithm on the whole superior to the DFP algorithm in terms of number of iterations but it is also, if the number of evaluations of $F(\mathbf{x})$ per iteration is taken to be the yardstick, slightly better in terms of work done during each iteration. Indeed for Problem 4 the number of iterations for the new method is one more than required by the DFP method but the total number of evaluations of $F(\mathbf{x})$ is five fewer, and taking an average over the 13 problems for which full results are available we see that the mean ratio of function to gradient evaluations is 8·55 for the new method and 9·88 for the DFP method. We therefore take as a measure of the relative effectiveness of the two algorithms the total number of gradient evaluations and on this criterion the new algorithm would be regarded as better than the DFP algorithm for Problems 1, 12, 13, 14, and 15 and comparable for Problems 2 and 4–11.

It must be borne in mind that the above results represent only a limited amount of numerical experience applied to a restricted set of problems, and to this extent will not necessarily reflect the overall merit of the two algorithms. They in no way constitute a case for modifying existing programs using the DFP algorithm, simple though this may be. Nor are they substantiated by any form of convergence proof apart from those based upon the erroneous hypothesis that the function to be minimized is

quadratic, and to this extent they are unsatisfactory and incomplete. They do, though, indicate that the new algorithm may repay further consideration, especially in those cases where the problems are known to be difficult or where convergence using existing methods has been less than rapid.

The author is extremely grateful to Mr K. Fielding, of the University of Essex, for carrying out the comparative testing of the algorithms, and to Mr W. Temple of the same University for providing one of the test problems.

References

Broyden, C. G. 1967 *Maths Comput.* **21,** 368.
Broyden, C. G. 1970 *J. Inst. Maths Applics* **6,** 76.
Davidon, W. C. 1959 *A.E.C. Research and Development Report ANL*-5990.
Fletcher, R. & Powell, M. J. D. 1963 *Comput. J.* **6,** 163.
Greenstadt, J. 1967 I.B.M. Scientfic Center Report No. 320–2901.
Pearson, J. D. 1969 *Comput. J.* **12,** 171.
Shah, B. V., Buehler, R. J. & Kempthorne, O. 1964 *J. Soc. ind. appl. Math.* **12,** 74.