Spiral—A new algorithm for non-linear parameter estimation using least squares

A. Jones

Shell Research Ltd., Thornton Research Centre, P.O. Box 1, Chester CH1 3SH

A new algorithm for the estimation of parameters in non-linear models is described and its performance is compared with that of other algorithms for minimising sums of squares. The new algorithm is shown to be a significant improvement and has performed well for practical as well as bench-mark problems. Further improvements are under investigation.

(Received November 1968)

Introduction

This paper describes a new algorithm for the estimation of parameters in non-linear models. It has been in use now for about a year and has proved very successful in fitting chemical kinetic (and other) models to experimental data.

The algorithm was devised while the author was investigating the relative merits of the two most successful algorithms, due to Marquardt (1963) and Powell (1965), for minimising a sum of squares that have appeared in the literature so far. The main disadvantage of Marquardt's algorithm is that it requires a matrix inversion to generate each search point, while Powell's method suffers from the disadvantage that a lot of time is spent in minimising along a selected direction in the parameter space.

The author believes that Newton-steps (i.e. a move to a new point in the parameter space as soon as a reduced sum of squares has been found) are preferable to line minimisation for parameter estimation and that the effort required to calculate gradients (analytic or estimated) is worthwhile. The present algorithm is based on these ideas and searches roughly the same area as does Marquardt's but the search points are generated by vector addition rather than by matrix inversion.

The algorithm is described below and its performance is compared with that of Marquardt's and Powell's algorithms over a fairly wide range of problems. The superiority of the new algorithm, measured in terms of function evaluations, is general and is quite spectacular for some of the bench-mark problems that have appeared in the literature recently.

A flexible computer program has been written in FORTRAN V for the Univac 1108 which includes a number of features that have proved useful to the statistician in fitting and discriminating between physical models. The statistical information printed on convergence includes the eigenvalues and eigenvectors of the parameter correlation matrix together with the actual sum of squares of the end points of the axes of any number of confidence ellipsoids. The comparison of these sums of squares with that predicted for the contours by linear theory can provide useful information about the degree of non-linearity of the model. Beale's (1960) measure of non-linearity was included in an early version of the program but in common with Guttman and Meeter (1965) the author has not found it to be a particularly useful measure of non-linearity and it has now been dropped. Powell's (1967) subroutine has been built into the program package and may be specified for the minimisation stage of the calculation in place of the new algorithm. This was done in order to facilitate the present comparison of the two algorithms by using the same subroutines and data packs. Further details are given in the section on convergence criteria (Appendix 1).

Existing methods of minimising a sum of squares

Good general accounts of the methods for solving non-linear least-squares problems have been given by Draper and Smith (1966) and by Powell in Walsh (1966). For completeness and to establish notation a brief account of the basic ideas is given here.

Let the mathematical model be represented by

$$\eta = f(\boldsymbol{\beta}, \boldsymbol{x})$$

where η is the experimentally observed variable, $\boldsymbol{\beta}$ is a vector of unknown parameters and $\boldsymbol{x}$ is a vector of independent experimental variables. The problem is to find an estimate $\boldsymbol{b}$ of $\boldsymbol{\beta}$ for which the sum of squares

$$\phi = \sum_{i=1}^{n} [y_i - f_i]^2$$

is a minimum, where y_i is the ith observation of η and $f_i = f(\boldsymbol{b}, \boldsymbol{x}_i)$, the $\boldsymbol{x}_i$ being the corresponding values of the independent variables $\boldsymbol{x}$. The ordinary least-squares method may be applied if the model is expanded in a Taylor Series about the current estimate $\boldsymbol{b}_0$, only first-order terms being retained. This leads to an improved estimate $\boldsymbol{b}\ (= \boldsymbol{b}_0 + \boldsymbol{t})$ for the linearised model, where $\boldsymbol{t}$ is the solution of

$$A\boldsymbol{t} = \boldsymbol{g}$$

where

$$A = P'P,$$

$$\boldsymbol{g} = P'(\boldsymbol{y} - \boldsymbol{f}_0),$$

P being the $(n \times k)$ matrix with elements $\left(\frac{\partial f_i}{\partial b_j}\right)_{\boldsymbol{b}=\boldsymbol{b}_0}$.

If the sum of squares at the new point $\boldsymbol{b}$ is smaller than the sum of squares at the point $\boldsymbol{b}_0$ then repeated application of this procedure will lead to a solution of the problem. If not, then the Taylor Series method

may not converge. However, at points away from the minimum, it is always possible to reduce the sum of squares by taking a sufficiently small step in the direction of steepest descent, $\boldsymbol{d}(= -\boldsymbol{g})$. Methods based entirely on the direction of steepest descent have not been successful since they take a very large number of iterations.

Marquardt's algorithm is based on the idea that the best direction for finding a reduced sum of squares lies in a direction $\boldsymbol{\delta}$ lying between $\boldsymbol{t}$ and $\boldsymbol{d}$. He finds this direction by solving the equations

$$(A + \lambda I)\boldsymbol{\delta} = \boldsymbol{g},$$

i.e. he adds λ to the diagonal elements of A. (For this to be meaningful the system must be scaled so that the matrix A has ones on the main diagonal.) When $\lambda = 0$, $\boldsymbol{\delta}$ is the Taylor Series direction $\boldsymbol{t}$, and as λ increases $\boldsymbol{\delta}$ swings towards the steepest descent direction $\boldsymbol{d}$. Within an iteration, λ is increased until a reduction in the sum of squares is obtained; between iterations, i.e. as the minimum is approached, λ is reduced in order to ensure second-order convergence. The re-inversion of the $(A + \lambda I)$ matrix each time λ is changed may be avoided and the inverse computed by a matrix multiplication (Appendix 2). A new version of Marquardt's (1966) program incorporating this idea was written by the author and was used in the present comparison.

Powell's algorithm uses estimates of the derivatives and is based on line minimisation along the Taylor Series direction. Having found the minimum along this line one of the axes of the parameter space is replaced by this direction and a new Taylor Series direction is computed. The main idea in Powell's algorithm is the method of computing the inverse of the matrix A in the transformed parameter space which avoids extra function evaluations for the calculation of derivatives. Full details are given in Powell's (1965) paper.

The Spiral algorithm

The basic idea behind the Spiral algorithm is that a reduced sum of squares can always be found in the plane defined by the Taylor Series point and the line of steepest descent at the base point. In **Fig. 1**, which is drawn in this plane, O is the base point, T is the Taylor Series point and OD is the direction of steepest descent, the point D being chosen so that the distances OT and OD are equal.

Since the sum of squares must decrease initially along OD and since the Taylor Series approximation predicts a reduced sum of squares at the point T then it is reasonable to suppose that reduced values of the sum of squares can be found over part of the area OTD. Overall strategy demands that the base point for the next iteration be as far away as possible from O but that the number of evaluations of the least squares surface be kept to a minimum.

With these points in mind the next base point will be taken at the first point found that shows a reduction in the sum of squares. Clearly the first point to be investigated must be the Taylor Series point, T. If this is not successful then the validity of the linear approximation to the model at O does not range as far as T. Hence the sum of squares 'valley' must curve in one of the two directions shown as hatched curves in Fig. 1. Since the overall strategy tends to give base points on the outside shoulder of the valley it is reasonable to assume that the 'valley' is moving away from the line OT. To try and intercept the valley the spiral OTS is searched; this curve moves out from T at an angle β into the area OTD and moves back into O tangentially to OD. The most suitable equation for this spiral (expressed in polar co-ordinates with O as origin) has been found to be

$$r = r_0(1 - \theta \cos \beta - (1 - \gamma \cos \beta)(\theta/\gamma)^2)$$

where r is the distance OS and r_0 is the distance OT.

The sequence of points, S, on the spiral to be investigated is computed from a sequence of points, L, generated on the line TD such that L divides TD in the ratio $\mu : (1 - \mu)$. The successive values of μ are computed from the recurrence relation

$$\mu_{n+1} = 2\mu_n/(1 + \mu_n),$$

which has been chosen to ensure that the points become closer together as they approach D.

If the coordinates of L are (ξ, θ), then, by elementary trigonometry, they are given by the relations

$$\tan \theta = \frac{\mu \sin \gamma}{1 - \mu + \mu \sin \gamma},$$

$$\xi = \frac{r_0 \mu \sin \gamma}{\sin \theta}.$$

Hence the coordinates of $\boldsymbol{s}$ in the parameter space, referred to O as origin, are given in terms of $\boldsymbol{t}$, the coordinates of T, and $\boldsymbol{d}$, the coordinates of D, by the relation

$$\boldsymbol{s} = \frac{r}{\xi} \cdot \{\mu \boldsymbol{d} + (1 - \mu)\boldsymbol{t}\}.$$

This equation is the basis for the Spiral algorithm and shows the main advantage of the present algorithm over Marquardt's; the successive search points in Marquardt's algorithms were generated by a matrix inversion whereas here they are generated by a weighted sum of two vectors. A block flow diagram of the algorithm is given in **Fig. 2** (with the nomenclature explained in **Table 1**) and the main features are explained below.*

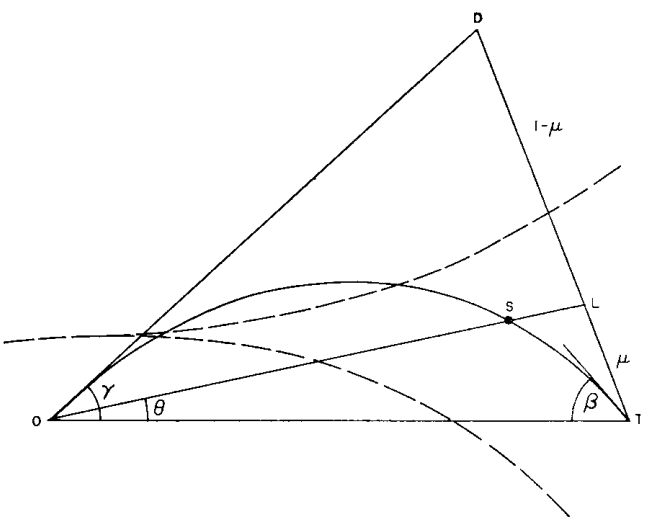

Fig. 1

* In the present version of the program the arbitrary angle β has been assigned the value $\gamma/2$ and the μ series has been started with $\mu_1 = 0{\cdot}1$.

Since the spiral is searched only at discrete points it is possible to jump over from one side of the valley to the other without finding a sum of squares smaller than that at the base point. To overcome this difficulty an interpolation is performed wherever three consecutive sums of squares along the spiral are convex downwards, the sum of squares being regarded as a function of μ. Since spurious local minima can occur there is no point in refining this interpolation and so, if the sum of squares at the interpolated value of μ is not lower than at the base point, the search along the spiral is continued until a preset maximum value of μ has been reached.

At this stage the Taylor Series point is replaced by the point midway along the line OT and another spiral is searched from this point. However, if the sum of squares at this mid-point is lower than that at the original Taylor Series it is worth interpolating along the Taylor Series direction in order to cater for the possibility that the sum-of-squares 'valley' crosses the line OT. If the value of the sum of squares at this interpolated point is lower than at the mid-point, a new spiral is searched starting from the interpolated point.

The program allows, as standard, four spirals to be searched. If a reduced value of the sum of squares has not been reached along any of these spirals then the direction of steepest descent itself is searched. The only occasions that this has happened with this technique have been for badly defined problems, i.e. problems for which very high correlations exists between the parameters. This situation is usually due to redundant parameters in the model but can be caused by poor starting values, particularly when estimated derivatives are being used and the model is in the form of a system of differential equations.

Test problems

The test problems which were used fall into three groups, bench-mark problems that have been repeatedly cited in the literature on optimisation techniques, chemical kinetic problems that have appeared in the literature on model fitting and some of the model fitting problems that the author was concerned with while this new algorithm was being developed. A brief description of each of the problems is given below. For those problems which arise from simultaneous non-linear equations the sum of squares to be minimised is given and for the parameter estimation problems the model to be fitted is given.

Rosenbrock's parabolic valley

(1) The function to be minimised is

$$\phi = 100(b_2 - b_1^2)^2 + (1 - b_1)^2$$

starting from the point $(-1{\cdot}2,\ 1{\cdot}0)$. This function (Rosenbrock, 1960) is difficult to minimise since the contours of equal ϕ follow a steep-sided banana-shaped valley following the curve $b_2 = b_1^2$.

(2) The ϕ-function is the same as in problem (1) but the starting point $(-1{\cdot}2,\ 1{\cdot}0)$ has been reflected in the parabola $b_2 = b_1^2$ to give a new starting point $(-0{\cdot}86,\ 1{\cdot}14)$. This problem has been included to investigate the effect on convergence of starting on the inside shoulder of the valley rather than the outside shoulder.

Fletcher and Powell's helical valley

(3) The function to be minimised is

$$\phi = 100[(b_3 - 10\theta)^2 + (r - 1)^2] + b_3^2,$$

where $b_1 = r\cos 2\pi\theta$ and $b_2 = r\sin 2\pi\theta$, starting from the point $(-1,\ 0,\ 0)$ (Fletcher and Powell, 1963). The ϕ surface follows a steep sided helical valley in the b_3 direction.

Powell's quartic function

(4) This is a four-parameter problem with the function to be minimised defined by

$$\phi = (b_1 + 10b_2)^2 + 5(b_3 - b_4)^2 + (b_2 - 2b_3)^4 + 10(b_1 - b_4)^4$$

with the starting point $(3,\ -1,\ 0,\ 1)$. This function is due to Powell (1962) and exhibits certain difficulties at the solution point $(0,\ 0,\ 0,\ 0)$.

(5) The function to be minimised is the same as in problem (4) but the starting point is chosen to be

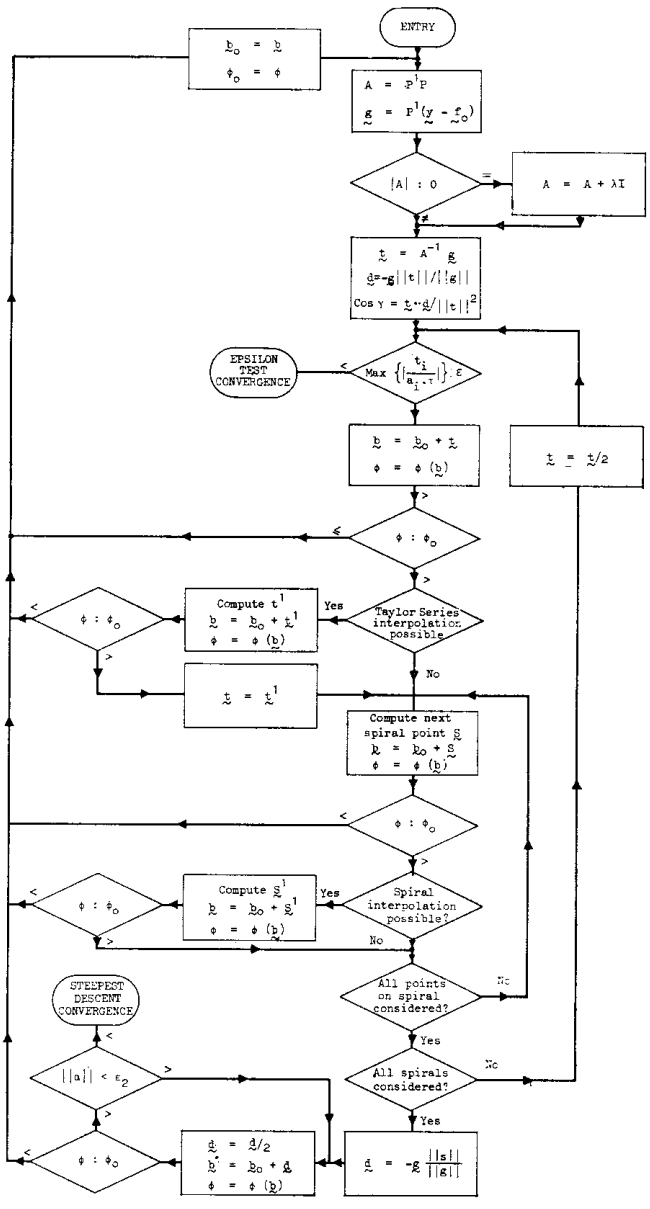

Fig. 2

Table 1

Nomenclature for Fig. 2

P	$(n \times k)$ matrix of elements $\left\{\frac{\partial f_j}{\partial b_j}\right\}$ $i = 1, ..., n, j = 1, ..., k$
$\boldsymbol{y}$	experimental data
$\boldsymbol{f}_0$	predicted values of the model at $\boldsymbol{x}_0$
λ	Marquardt's parameter
$\boldsymbol{t}$	Taylor Series increments
$\boldsymbol{d}$	steepest descent increments
γ	angle between Taylor Series and steepest descent directions
τ, ϵ	convergence parameters
$\boldsymbol{b}$	search point
ϕ	value of sum of squares at search point
$\boldsymbol{b}_0$	base point
ϕ_0	value of sum of squares at base point
$\boldsymbol{t}'$	interpolated Taylor Series increments
$\boldsymbol{s}$	spiral increments
$\boldsymbol{s}'$	spiral interpolated increments
ϵ_2	steepest descent convergence criterion

(3, 2, 1, 3) where the matrix of the linearised normal equations is singular.

The above problems have been considered by many authors in their evaluation of new general optimisation techniques. For ease of reference **Table 2** gives a list of the authors and shows which of problems (1), (3) and (4) they cite.

Zangwill's problem

(6) The function to be minimised is

$$\phi = (b_1 - b_2 + b_3)^2 + (-b_1 + b_2 + b_3)^2 + (b_1 + b_2 - b_3)^2$$

from the starting point $(\frac{1}{2}, 1, \frac{1}{2})$. This function was reported by Zangwill (1967) as a counter-example to Powell's (1964) conjugate gradient algorithm.

(7) This is the same function as in problem (6) but with the starting point (100, −1, 2·5). This is Zangwill's (1967) second counter-example to Powell's (1964) algorithm.

Box, Hunter and Roth's problem

(8) This problem is based on the model reported by Box and Hunter (1965) for catalytic dehydration of *n*-hexyl alcohol. This model is

$$y = \frac{b_3 b_1 x_1}{1 + b_1 x_1 + b_2 x_2}$$

where y is the rate of reaction, x_1, x_2 are the partial pressures of alcohol and olefin, b_1, b_2 are the equilibrium constants, and b_3 is the effective reaction rate constant.

Consecutive reactions

(9) The two-stage unimolecular decomposition of a species A, through B to a product C may be represented by the reaction scheme

$$A \xrightarrow{b_1} B \xrightarrow{b_2} C$$

where b_1 and b_2 are the reaction rate constants. By solving the differential equations representing this reaction scheme, the concentration, y, of B at any time, t, may be expressed in the form

$$y = \frac{b_1}{b_1 - b_2}(e^{-b_2 t} - e^{-b_1 t})$$

This problem was originally considered by Box and Coutie (1956) and was used as an example by Booth and Peterson (1960).

Morse potential function

(10) The interaction, μ, between the two atoms of a diatomic molecule is a function of their separation, r, measured from the equilibrium separation, and may be represented as a potential function (see Pauling and Wilson, p. 272)

$$\mu = b_1[1 - \exp(-b_2 r)]^2$$

Table 2

Citations of bench-mark problems

AUTHOR	PROBLEM		
	ROSENBROCK'S PARABOLIC VALLEY	HELICAL VALLEY	QUARTIC FUNCTION
Powell (1964)	√		√
Nelder and Mead	√	√	√
Fletcher (1965)	√	√	√
Fletcher and Powell	√	√	√
Powell (1965)	√		
Powell (1962)	√		√
Fletcher and Reeves	√	√	
Stewart	√	√	√

where b_1 is the dissociation energy and b_2 is a curve-fitting parameter.

Flame chemistry

(11) The hydrogen atom concentration, y, in a hydrogen/oxygen flame with an added metal catalyst may be described as a functon of time, t, by the differential equation

$$\frac{d}{dt}\left(\frac{1}{y}\right) = A + b_7 M_0(y^2 + By + C)$$

with $y = y_0$ at $t = 0$, where b_7 is an unknown rate constant, M_0 is a known amount of added metal and A, B, C are known constants. In order to estimate b_7 accurately six experimental runs were considered at the same time and so six values $b_1, b_2, \ldots, b_6$ of the initial hydrogen atom concentration, y_0, have to be determined. Thus seven constants are being determined simultaneously in this problem; however, the first six of these are statistically independent of one another. The chemical aspects of this work will be reported by Cotton and Jenkins (1968).

Hydrogen recombination

(12) The relationship between hydrogen atom concentration, y, and distance, x, in a constant velocity flow tube at room temperature may be described by the differential equation

$$\frac{d^2y}{dx^2} + \frac{A dy}{dx} + b_1 y + b_2 y^2 = 0$$

with $y = b_3 \quad x = 0,$

$y = 0 \quad x = \infty$

The chemical aspects of this work will be reported by Bennett and Blackmore (1968).

These last two problems are interesting in that the predicted values are given as the (numerical) solution of differential equations. The initial value problem in case (11) was solved using a fifth order Runge-Kutta process due to Scraton (1964) and the two-point boundary value problem in case (12) was solved using a modified form of the finite difference scheme due to Allen (1966).

Viscosity-temperature

(13) The viscosity, η, of a mineral oil may be predicted from its temperature, T by the relationship

$$n = b_1 + b_2/(T - b_3)^{b_4}$$

Attempts to estimate all four parameters in this model led to difficulties with all the algorithms because of very high correlations between the parameters. So for the purposes of this work the parameter b_4 was held fixed at a given value, thus reducing the size of this problem to three parameters.

Performance of algorithms

The criterion used to judge the performance of the algorithms was the number of evaluations of the predicted values at a given point in the parameter space. This is equivalent to the number of function evaluations, an evaluation of predicted values for the purpose of estimating derivatives being interpreted as a function evaluation. **Table 3** summarises the performance of the three algorithms on this basis. The figures for the Marquardt and Spiral algorithms are for estimated derivatives; the corresponding figures for analytical derivatives, in those cases where differentiation was convenient, were virtually identical. (Incidentally, this shows that writing extra coding for the evaluation of analytic derivatives is only worthwhile when the derivatives are markedly easier to compute than the original function.)

The convergence criterion used was the same for all three programs and is discussed in Appendix 1. In order to show the relative position that each algorithm had reached at convergence the minimum sum of squares for each run is given in **Table 4**.

The only problem which produced significantly different minimum sums of squares was problem (13). However, there were no significant differences between the least squares estimates of the parameters obtained from the three algorithms and it may be safely said that the correct answers were obtained for all the problems.

From Table 3 the new algorithm comes out clearly better than either 'Powell' or 'Marquardt', with, perhaps, 'Marquardt' being marginally better than 'Powell'. A comparison of the results shown here with those quoted by the authors listed in Table 2, for general optimisation algorithms, shows quite clearly the benefit to be obtained by using special techniques for minimising sums of squares. This comparison does not take into account the overhead involved in the three algorithms and in order to attempt the measurement of this overhead the execution times were recorded for each computer run. However, these were so short that they proved completely unreliable.* Problem (12) took about one minute, problem (11) took about four seconds, while

Table 3

Performance of the Powell, Marquardt and Spiral algorithms on various test problems given in terms of function evaluations

PROBLEM	MARQUARDT	POWELL	SPIRAL
1	92	143	17
2	72	103	27
3	49	319	39
4	98	81	66
5	103	112	76
6	61	27	9
7	21	52	13
8	67	97	31
9	27	12	13
10	22	25	13
11	—	23	33
12	—	32	29
13	106	48	21

NOTE: Problems 11 and 12 were not attempted with Marquardt's algorithm.

* The computer execution time was measured as the difference between the times at the end and at the beginning of execution as indicated by the internal time-of-day clock.

the other runs took less than 1·5 seconds actual computing time on the Univac 1108. Computer times as short as this are significantly affected by the internal and external interrupt handling which is going on all the time on a multi-access computer. This means that, for the sort of problem considered here, there is very little difference between the algorithms as far as computer time is concerned. Hence the extrapolation to larger more realistic problems must be made on the basis of the number of function evaluations required for convergence.

Discussion

From a comparison of the current work with results obtained on similar problems by general methods (e.g. the algorithm FLEPOMIN, Wells (1965), Fletcher (1966)) it is clear that there is considerable benefit to be obtained by using special methods for minimising functions that are sums of squares. (It is also worth noting that these special methods are also applicable to solving non-linear equations.) The other comparison which should be made is that with the simplex method due originally to Spendley *et al.* (1962) (see also Nelder and Mead (1965) and Box (1965).) The author has been able to compare some of the test problems considered here with the new simplex-type algorithm with quadratic fitting due to Spendley (1968). This algorithm looks quite promising and so far appears to be comparable to the Marquardt and Powell algorithms as far as the number of function evaluations is concerned.

The author is currently considering two possible improvements to the Spiral algorithm.

The first improvement will be an attempt to overcome the problem of 'ridges' in the least-squares surface. This might be handled by an extrapolation along the Taylor Series direction coupled with an attempt to follow the curvature of the ridge by evaluating the direction of steepest descent at the Taylor Series point.

The second improvement is concerned with the geometry of the spiral when the Taylor Series point is unsuccessful. At present the spiral is generated inside an isosceles triangle, i.e. the distance along the steepest descent direction is taken as being equal to the distance of the Taylor Series point from the base point. It might be preferable to relate the distance along the steepest descent direction to the ratio of the smallest to the largest eigenvector, possibly after these have been projected on the Taylor Series—steepest descent plane.

Acknowledgements

The author would like to thank his colleagues in the Mathematics Division at Thornton for many stimulating discussions in the development of this work, in particular, Mrs. J. M. Ferguson, who helped with the programming and organised all the computer runs. The author would also like to thank Mr. W. Spendley and Mr. M. J. D. Powell for their co-operation in providing copies of their computer programs.

Appendix 1

Convergence criteria and estimated derivatives

The Marquardt and Spiral programs will compute either analytic or estimated derivatives. The estimated derivatives are computed from the following formulae

$$\frac{\partial f_i}{\partial b_j} = \frac{f(b_j + e_j) - f(b_j)}{e_j}$$

where $e_j = \Delta b_j$ if $|\Delta b_j| > B$

and $e_j = \Delta$ if $|\Delta b_j| < B$;

where Δ, B have the values 5×10^{-5} and 10^{-10}.

For both the Marquardt and Spiral algorithms the convergence criterion was based on the relative change in the parameters predicted by the Taylor Series point. Specifically convergence was assumed when

$$\text{Max} \left| \frac{t_i}{b_i + \tau} \right| < \epsilon$$

where b_i are the current parameters estimates and t_i are

Table 4

Minimum sum of squares attained by the Powell, Marquardt and Spiral algorithms on the test problems

PROBLEM	MARQUARDT	POWELL	SPIRAL
1	$0{\cdot}22 \times 10^{-13}$	$0{\cdot}61 \times 10^{-12}$	0
2	$0{\cdot}22 \times 10^{-13}$	$0{\cdot}30 \times 10^{-10}$	0
3	$0{\cdot}22 \times 10^{-13}$	$0{\cdot}23 \times 10^{-13}$	$0{\cdot}22 \times 10^{-13}$
4	$0{\cdot}19 \times 10^{-14}$	$0{\cdot}61 \times 10^{-15}$	$0{\cdot}56 \times 10^{-12}$
5	$0{\cdot}24 \times 10^{-14}$	$0{\cdot}10 \times 10^{-12}$	$0{\cdot}10 \times 10^{-12}$
6	$0{\cdot}26 \times 10^{-16}$	$0{\cdot}47 \times 10^{-14}$	$0{\cdot}81 \times 10^{-14}$
7	$0{\cdot}25 \times 10^{-18}$	$0{\cdot}51 \times 10^{-5}$	$0{\cdot}40 \times 10^{-17}$
8	$0{\cdot}435527 \times 10^{-4}$	$0{\cdot}435559 \times 10^{-4}$	$0{\cdot}435527 \times 10^{-4}$
9	$0{\cdot}738599 \times 10^{-2}$	$0{\cdot}738689 \times 10^{-2}$	$0{\cdot}738599 \times 10^{-2}$
10	$0{\cdot}180065 \times 10^{8}$	$0{\cdot}180065 \times 10^{8}$	$0{\cdot}180065 \times 10^{8}$
11	—	0·14416384	0·14415643
12	—	0·0241627	0·0241627
13	0·218589	0·212564	0·206477

the corresponding Taylor Series increments. (τ is included to cope with the possibility that a parameter has a zero value at the minimum.) The values of τ, ϵ were 0·001 and 0·0001 for all the computer runs on the problems reported.

Powell's subroutine uses an absolute accuracy vector e both for estimating derivatives and for convergence. In order to ensure that the convergence criteria were the same as for the other two algorithms the elements of the e vector were computed as in the derivative increment formulae above but with Δ taken as 0·0001, i.e. the same as ϵ. The parameter ESCALE, which the Powell subroutine requires in order to limit the step-size of the search (normally a parameter b_i will not be changed by more than ESCALE $\times$ b_i in a single step), was always set to 100·0. Early attempts to use the Powell subroutine were abortive but no difficulties were encountered after this scheme for computing the e vector and ESCALE was introduced.

Appendix 2

A modification to Marquardt's algorithm

Marquardt's algorithm is based on the idea that the optimum direction of search lies between the Taylor direction $\boldsymbol{\delta}_t$, and the steepest descent direction, $\boldsymbol{\delta}_g$. He finds this direction, $\boldsymbol{\delta}$ by solving the equations

$$(A + \lambda I)\boldsymbol{\delta} = \boldsymbol{g}$$

i.e. he adds λ to the diagonal elements of A. (For this to be meaningful the system must be scaled so that the diagonal elements of A are all unity; Levenberg (1944) multiplies the diagonal elements of A by a constant greater than unity.) When $\lambda = 0$, $\boldsymbol{\delta}$ is the Taylor Series direction and as λ increases to infinity $\boldsymbol{\delta}$ swings towards the steepest descent direction $\boldsymbol{\delta}_g$. Marquardt's algorithm contains two further ideas. Within an iteration, λ is increased until a reduced sum of squares is obtained.* Between iterations λ is successively reduced so that as the minimum is reached δ tends to move closer to the Taylor Series direction. This ensures that the overall algorithm has second-order convergence near the minimum.

One of the disadvantages of Marquardt's program is that he inverts the $(A + \lambda I)$ matrix each time λ is altered. The reinversions within an iteration, i.e. when A is unchanged, may be avoided as follows.

Since A is positive definite and symmetric then

$$TAT' = \text{diag}\{\mu_j\}$$

where T is an orthogonal matrix of eigenvectors and the μ_j are positive real eigenvalues. The inverse of A is given by

$$A^{-1} = T' \,\text{diag}\left\{\frac{1}{\mu_j}\right\} T$$

and the inverse of $(A + \lambda I)$ is given by

$$(A + \lambda I)^{-1} = T' \,\text{diag}\left\{\frac{1}{\lambda + \mu_j}\right\} T$$

So that once the eigenvalues and eigenvectors of A are known the inverse of $(A + \lambda I)$ may be computed for any value of λ by a matrix multiplication.

This analysis throws some light on the reason why Marquardt's algorithm is more successful than the straight Taylor Series method. The smallest eigenvalue value of $(A + \lambda I)$ is $\mu_j + \lambda$, where μ_j is the smallest eigenvalue of A, so that $A + \lambda I$ cannot be singular for $\lambda > 0$.

In addition to replacing matrix inversions by multiplications the calculation of these eigenvalues and eigenvectors provides very useful information about the shape of the least-squares surface. Further details of this are given by Booth and Peterson (1960).

* This is always possible, as Marquardt (1963) shows in his paper.

References

ALLEN, B. T. (1966). A new method of solving second order differential equations when the first derivative is present, *The Computer Journal*, Vol. 8, p. 392.

BEALE, E. M. L. (1960). Confidence regions in non-linear estimation, *Journal of the Royal Statistical Society, Series B*, Vol. 22, p. 41.

BENNETT, J. E., and BLACKMORE, D. R. (1968). The measurement of the rate of recombination of hydrogen atoms at room temperature using E.S.R. spectroscopy. (To be published.)

BOOTH, G. W., and PETERSON, T. I. (1960). Non-linear estimation, IBM Mathematics and Applications Reports M and A 3, New York.

BOX, M. J. (1965). A new method of constrained optimization and a comparison with other techniques, *The Computer Journal*, Vol. 8, p. 42.

BOX, G. E. P., and COUTIE, G. A. (1956). Applications of digital computers in the exploration of functional relationships, *Proceedings of the Institute of Electrical Engineers*, Vol. 103B, Suppl. No. 1, p. 100.

BOX, G. E. P., and HUNTER, W. G. (1965). The experimental study of physical mechanisms, *Technometrics*, Vol. 7, p. 23.

COTTON, D. C., and JENKINS, D. R. (1968). The catalysis of radical recombination reactions in flames with metal additives. (To be published.)

DRAPER, N. R., and SMITH, H. (1966). *Applied regression analysis*, J. Wiley and Sons, New York.

FLETCHER, R. (1965). Function minimization without evaluating derivatives—a review, *The Computer Journal*, Vol. 8, p. 33.

FLETCHER, R., and POWELL, M. J. D. (1963). A rapidly convergent descent method for minimization, *The Computer Journal*, Vol. 6, p. 163.

FLETCHER, R., and REEVES, C. M. (1964). Function minimization by conjugate gradients, *The Computer Journal*, Vol. 7, p. 149.

FLETCHER, R. (1966). Function minimization, certification of Algorithm 251, *CACM*, Vol. 9, p. 686.

GUTTMAN, I., and MEETER, D. A. (1965). On Beale's measure of non-linearity, *Technometrics*, Vol. 7, p. 623.

LEVENBERG, K. (1944). A method for the solution of certain non-linear problems in least squares, *Quarterly Journal of Applied Mathematics*, Vol. 2, p. 164.

MARQUARDT, D. W. (1963). An algorithm for least-squares estimation of non-linear parameters, *Journal of the Society for Industrial and Applied Mathematics*, Vol. 2, p. 431.

MARQUARDT, D. W. (1966). Least squares estimation of non-linear parameters, Share Program Library No. 3094.
NELDER, J. A., and MEAD, R. (1965). A simplex method for function minimization, *The Computer Journal*, Vol. 7, p. 308.
PAULING, L., and WILSON, E. B. (1935). *Introduction to quantum mechanics*, Students Version, McGraw-Hill, Kogakusha, Tokyo.
POWELL, M. J. D. (1962). An iterative method for finding stationary values of a function of several variables, *The Computer Journal*, Vol. 5, p. 147.
POWELL, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives, *The Computer Journal*, Vol. 7, p. 155.
POWELL, M. J. D. (1965). A method for minimizing a sum of squares of non-linear functions without calculating derivatives, *The Computer Journal*, Vol. 7, p. 303.
POWELL, M. J. D. (1967). Unpublished program specifications.
ROSENBROCK, H. H. (1960). An automatic method for finding the greatest or least value of a function, *The Computer Journal*, Vol. 3, p. 175.
SCRATON, R. E. (1964). Estimation of the truncation error in Runge-Kutta and allied processes, *The Computer Journal*, Vol. 7, p. 246.
SPENDLEY, W., HEXT, C. R., and HIMSWORTH, F. C. (1962). Sequential application of the simplex designs in optimization and evolutionary operations, *Technometrics*, Vol. 4, p. 441.
SPENDLEY, W. (1968). Non-linear least square fitting using a modified simplex minimization technique, Paper presented to the Joint I.M.A.-B.C.S. Conference on Optimization at Keele University. (Proceedings to be published.)
STEWART, III, G. W. (1967). A modification of Davidon's minimization method to accept difference approximations of derivatives, *JACM*, Vol. 14, p. 72.
WALSH, J. (Ed.) (1966). *Numerical analysis: an introduction*, Academic Press, London.
WELLS, M. (1965). Function minimization-FLEPOMIN, Algorithm 251, *CACM*, Vol. 8, p. 169.
ZANGWILL, W. I. (1967). Minimizing a function without calculating derivatives, *The Computer Journal*, Vol. 10, p. 293.

Book Review

Computational Problems in Abstract Algebra, Edited by John Leech, 1970; 401 pages. (*Pergamon Press Ltd.*, £7 $18.50.)

This work is the proceedings of a conference at Oxford in 1967, to study the uses of computers in algebra. It is a well-known paradox that one of the fields where computers have had the least impact is pure mathematics, but there are signs that this situation is slowly changing, partly no doubt due to conferences such as this one. The main problem is to sort out the questions that are both manageable by computers *and* useful to the mathematician. This volume contains 35 articles ranging from research papers with faintly computational flavour to detailed programmes for answering specific questions using computers.

One of the crucial ways in which computers help the algebraist is in the study of finite simple groups. It so happens that the simple groups that were being found then were just of the right size to make this possible. An article by Marshall Hall, Jr. describes the search for simple groups of order less than 10^6, and actually constructs Janko's group of order 604,800 for the first time. There are several papers on the problem of constructing character tables, enumerating cosets and computing other data for groups, with varying amounts of stress on the computational side. Some authors, e.g. H. Jürgensen, describing multiplication in groups with given presentations, give detailed programmes illustrated by flow charts, but most authors are content to leave the problem in a form in which it can be passed on to the programmer and then report the outcome of the calculation.

There are several articles on other algebraic systems such as semigroups and projective configurations. The most striking are the papers on Jordan algebras by J. L. Paige and C. M. Glennie, which describe the search for identities in special Jordan algebras. Earlier work by Glennie has shown that 'special' identities of degree 8 exist, and he describes here how the computer was used to show that 8 is the least such degree; perhaps such methods can be used to help one get a basis for all the identities.

An interesting feature are the examples of what N. S. Mendelsohn calls 'man-machine interaction'. When a routine computation, e.g. deriving the consequences of a set of defining relations, is put on a computer, a pattern may emerge that was not apparent before. Some pertinent cases are discussed by D. E. Knuth and P. B. Bendix in their article on word problems and universal algebra.

Finally there are articles that do not specifically invoke computers. In particular there is J. H. Conway's splendid piece on enumerating knots, describing a notation which compresses six years work (in the 1890's) into one afternoon. This paper, and the one by H. F. Trotter make it clear that the use of computers can greatly increase the scope of these methods.

To sum up, here is a varied collection of papers showing both the scope and limitations of computers in algebra; they will clearly help the algebraist to decide when to call in a programmer, what he will have to tell him and what answer he can expect.

P. M. COHN (London)