SIAM J. NUMER. ANAL.
Vol. 18, No. 6, December 1981

0036-1429/81/1806-0013$01.00/0

A CONTINUATION METHOD FOR NONLINEAR REGRESSION*

NOËL DE VILLIERS† AND DAVID GLASSER†

Abstract. Many iterative regression methods only perform well near a solution.

One may construct a modified problem with a free parameter (k), which has a known solution for a particular k. By suitably deforming this problem through k-space one may, always remaining close to an intermediate solution, eventually solve the original problem.

The continuation method was used to perform three nonlinear least squares regressions where the conventional Newton methods failed to converge. In particular, the third problem was associated with experimental kinetic data and differential equations which needed to be integrated numerically. No satisfactory answers were obtainable from the Gauss–Newton method, while the continuation (parameter variation) method showed its robustness by converging for a large range of initial values.

1. Introduction. The problem of the estimation of the "best" constants in the fitting of a model to experimental data is an old one, but one which has not really been solved entirely satisfactorily.

The most recent methods and philosophy for attacking and solving these problems are fully described in Bard (1974), the appropriate chapters in Seinfeld and Lapidus (1974) and an article by Dennis (1977).

New methods have been suggested by Cresswell and Paterson (1976) and Stewart and Sorenson (1976). These are associated with the methods of weighted residuals and collocation.

All of the more general methods essentially require the definition of some objective function and the subsequent choice of parameters to minimize this function. The choice of the "best" or most appropriate objective function is itself a subject of much discussion and will not be pursued further in this paper. Objective functions which are often used are least squares and maximum likelihood.

Once we have chosen the objective function we have essentially to solve an optimization problem, but one in which some of the structure is already defined. Thus, all the iterative nonlinear optimization methods devised are in principle available for the numerical solution of this problem. However, because of the structure of the problem certain methods have found favor. In particular, Bard (1970) has found the Gauss–Newton method to be somewhat more efficient than its competitors. This is the basic method that will be used in this paper, although the general approach will be valid for any of the iterative methods. Dennis (1977) discusses convergence of the Gauss–Newton algorithm, together with limitations of this method and modifications to overcome these difficulties.

To return to the iterative methods for the solution of nonlinear parameter estimation problems, the main difficulty in their use appears to be the problem of convergence.

The convergence of most of the methods is very good within some region near the solution. That is, if we have a "good" initial set of parameters we can expect an easy passage home. If, however, our initial set is "poor" there may be great difficulty in converging to a solution; in fact the method may not converge at all. It is to this problem that the method in this paper addresses itself.

These drawbacks can be overcome by a method which greatly enlarges the region of "good initial guesses" by ensuring that any chosen set of initial values is close to a solution of the nonlinear estimation problem. This may be achieved by modifying

* Received by the editors January 12, 1978, and in final revised form March 4, 1981.

† Chemical Engineering Department, University of the Witwatersrand, Johannesburg, South Africa.

the problem through the introduction of an independent linear parameter $k \in [0, 1]$ into the objective function in such a way that when $k = 0$ the initially guessed starting values yield a zero modified objective function (thus constituting a solution to the modified problem), and when $k = 1$ the modified objective function becomes the required objective function. By starting from a known solution and systematically developing solutions for intermediate values of $k \in [0, 1]$, very good starting values for final iterations at $k = 1$ are obtained. This approach may be called a continuation method or a parameter variation method. (See Deist and Sefor (1967), Glasser (1969) and Glasser and de Villiers (1974).)

In addition to good starting values for an iterative algorithm, several other advantages are also gained. Since the modified problem is solved for a sequence of $k \in [0, 1]$, any instability or failure to provide an intermediate solution may be immediately detected and rectified. If the stepsize Δk in k is sufficiently small, the original nonlinear problem is reduced to the solution of a sequence of intermediate linear problems which may be solved without iteration. Finally, the way in which the parameter k is introduced is very general, allowing the method to be tailored to special characteristics of the problem.

For the purpose of illustrating this philosophy a weighted least squares objective function has been chosen and the optimizing procedures used are the standard Newton–Raphson or Gauss–Newton methods.

2. The standard nonlinear least squares problem. It is required to approximate a given set of data (x_i, y_i), $i = 1, \cdots, n$, where x is the independent variable, by a nonlinear approximating function containing r unknown parameters b_j, $j = 1, \cdots, r$. Let b denote the r-vector b_j, $j = 1, \cdots, r$, and let the nonlinear approximating function be $f(x, b)$. The residual at a given point (x_i, y_i) is

$$F_i(b) = f(x_i, b) - y_i, \qquad i = 1, \cdots, n. \tag{1}$$

The n-vector $F_i(b)$, $i = 1, \cdots, n$ is denoted by $F(b)$. The nonlinear least squares problem is to find a value of b which is a critical point of the sum-of-squares function

$$S(b) = \tfrac{1}{2}\|F(b)\|_2^2 = \tfrac{1}{2}F(b)^T F(b), \tag{2}$$

where $F : R^r \to R^n$ is twice differentiable and $n \geqq r$.

There is a whole battery of Newton, quasi-Newton, conjugate gradient and derivative-free methods available to perform unconstrained minimization of the function $S(b)$ of (2). Discussion here will be focussed on Newton-type methods.

Let $J = J(b) = F'(b)$ be the Jacobian of $F(b)$. The gradient of $S(b)$ is

$$\nabla S(b) = J^T F(b) \tag{3}$$

and the Hessian of $S(b)$ is given by

$$\nabla^2 S(b) = J^T J + \sum_{i=1}^{n} F_i(b) \nabla^2 F_i(b), \tag{4}$$

where $\nabla^2 F_i(b)$ is the Hessian of the ith component of $F(b)$. From Brodlie (1977), a Newton–Raphson step on b is given by

$$b_{l+1} = b_l - [\nabla^2 S(b_l)]^{-1} \nabla S(b_l), \qquad l = 1, 2, \cdots \tag{5}$$

$$= b_l - \left[J^T J + \sum_{i=1}^{n} F_i(b_l) \nabla^2 F_i(b_l) \right]^{-1} J^T F(b_l). \tag{6}$$

Provided that b_l is near to the solution and $\nabla^2 S(b)$ is positive definite, the iteration of (5) will converge to a solution and will display a quadratic rate of convergence.

When b is close to a critical point, the residuals F_i, $i = 1, \cdots, n$ are small (provided the model is "good"). If these residuals, which are coefficients of $\nabla^2 F_i(b)$, are sufficiently small so as to make the term $\sum_{i=1}^{n} F_i(b)\nabla^2 F_i(b)$ small compared to the term $J^T J$, then from (4) the following approximation may be made:

$$\nabla^2 S(b) \doteqdot J^T J. \tag{7}$$

Substituting from (7) into (5) yields the following step:

$$b_{l+1} = b_l - [J^T J]^{-1} J^T F(b_l), \qquad l = 1, \cdots. \tag{8}$$

The method described in (8) is known as the Gauss–Newton method. The Gauss–Newton step requires evaluation of J, while the Newton–Raphson step requires evaluation of J in addition to evaluation of the Hessian of each component F_i of F. In the domain where it works, the Gauss–Newton method is therefore computationally much more efficient than the Newton–Raphson method. Dennis (1977) discusses the validity of this approximation.

A problem frequently encountered in practice is the actual or near rank-deficiency of J and the fact that $\nabla^2 S(b)$ is not positive definite, either in its exact form of (4) or in its approximate form of (7). One way of countering this problem which has proved successful in practice has been to incorporate the Levenberg–Marquardt modification into the Newton–Raphson and Gauss–Newton algorithms. This requires that $\nabla^2 S(b_l)$ be replaced by the augmented matrix $\nabla^2 S(b_l) + \mu_l I$, where μ_l is chosen so that the augmented matrix is positive definite.

To ensure that the algorithm converges, a damping factor α_l may be incorporated into the minimization scheme and is chosen in such a way that $S(b_{l+1}) \leqq S(b_l)$. A damped Newton–Raphson step with the Levenberg–Marquardt modification is given by

$$b_{l+1} = b_l - \alpha_l [\nabla^2 S(b_l) + \mu_l I]^{-1} J(b_l)^T F(b_l),$$

while the corresponding damped Gauss–Newton step is

$$b_{l+1} = b_l - \alpha_l [J(b_l)^T J(b_l) + \mu_l I]^{-1} J(b_l)^T F(b_l).$$

The Newton-type methods described above have been extensively applied to the solution of nonlinear least squares problems.

3. Formulation of a continuation method to solve the nonlinear least squares problem. It is well known that many iterative methods only perform well near a solution. When not near a solution, these methods may converge slowly or may sometimes even diverge. One way of overcoming this problem is to use a continuation method to ensure nearness to a solution at all stages of the computation.

A continuation method employs a parameter to generate a family of problems which has the property that if a solution to one member of the family is known, the solution to the next member of the family may be predicted. If this predicted solution for the next member is within the region of convergence for an iterative method, then the iterative method will quickly and efficiently find the actual solution for that member of the family. If the continuation parameter is now chosen in such a way that the solution to the first member of the family is known, and if the last member of the family is the actual problem to be solved, then the continuation method provides an orderly way of developing from a known solution of a simplified synthetic problem

through to the desired solution of the problem originally posed. In this way, nearness to the solution is ensured throughout the algorithm. For this method to work it is of course necessary that solutions to all members of the family do exist.

A continuation method for the nonlinear least squares problem of § 2 may be constructed in the following manner. Choose an independent linear parameter $k \in [0, 1]$ as the continuation parameter. Introduce k into the sum-of-squared errors objective function in such a way that when $k = 0$ the resulting modified objective function is minimized by an initial guess b^0 which now becomes a known solution to the first member of the family of problems generated by parameter k, and when $k = 1$ the modified objective function becomes (2), which is the desired sum of squared errors to be minimized and corresponds to the last member of the family. This may be achieved by replacing the residual vector $F(b)$ by the modified residual vector

$$F^k(b) = F(b) + (k-1)F(b^0). \tag{9}$$

The corresponding modified objective function is

$$S^k(b) = \tfrac{1}{2}\|F^k(b)\|_2^2 = \tfrac{1}{2}F^k(b)^T F^k(b), \tag{10}$$

where $F^k(b): R^r \to R^n$ is twice differentiable and $n \geqq r$. When $k = 0$ and $b = b^0$, then $F^0(b^0) = \mathbf{0}$ and consequently $S^0(b^0) = 0$. The modified objective function is thus minimized and the initial guess b^0 is the critical point of the first member of the family.

Starting from the initial guess b^0, the family of problems is generated by selecting a sequence $0 = k_0 < k_1 < k_2 < \cdots < k_m = 1$ of continuation parameters k. It is required to determine the sequence $b^1, b^2, \cdots, b^m$, where b^l is the critical point of $S^{k_l}(b)$, and b^m is the solution to the problem originally posed.

For any $k_l \in [0, 1]$, a Newton–Raphson or Gauss–Newton iterative method may be used to find the critical point of $S^{k_l}(b)$. The form of these iterations follows directly from § 2. Note that $F^{k_l}(b) = F'(b) = J$. The gradient vector of $S^k(b)$ is given by

$$\nabla S^k(b) = J^T F^k(b)$$

and the Hessian of $S^k(b)$ is given by

$$\nabla^2 S^k(b) = J^T J + \sum_{i=1}^{n} F_i^k(b) \nabla^2 F_i(b),$$

where $\nabla^2 F_i(b) = \nabla^2 F_i^k(b)$ is the Hessian of the ith component of the modified residual vector $F^k(b)$. The Newton–Raphson step on b at $k = k_l$ is given by

$$b_{l+1} = b_l - \left[J^T J + \sum_{i=1}^{n} F_i^{k_l}(b_l) \nabla^2 F_i(b_l)\right]^{-1} J^T F^{k_l}(b_l), \qquad l = 1, 2, \cdots. \tag{11}$$

The corresponding Gauss–Newton step on b at $k = k_l$ is given by

$$b_{l+1} = b_l - [J^T J]^{-1} J^T F^{k_l}(b_l), \qquad l = 1, 2, \cdots. \tag{12}$$

If the value of b at which iteration commences is within the region of convergence for the method being used, and if the matrices within the square brackets in (11) and (12) are positive definite, then for $k = k_l$ iteration of these steps should converge to a critical point b^l. Note that the Levenberg–Marquardt modification may be applied to the two steps of (11) and (12).

It is now necessary to establish a criterion to determine when the intermediate problems and the problem originally posed have converged to their critical points $b^1, \cdots, b^{m-1}$ and b^m respectively. For the embedded least squares problem described

above, the main property of interest of the sequence $b^0, b^1, \cdots, b^{m-1}$ is that it enables the continuation path to be traced. Therefore these intermediate critical points of S^{k_l} need not be determined to great accuracy. Only the final critical point b^m needs to be determined to the accuracy required for the original problem. Criteria which may be used to determine convergence to b^j include tests on: the reduction of the objective function S^{k_l}, a norm or reduction of a norm of the gradient ∇S^{k_l}, or on the size of the correction step $\Delta b_l = b_{l+1} - b_l$. The following simple test proved to be suitable for the examples tested in § 4:

Accept b as a critical point b^j of S^{k_l} when

$$\|\nabla S^{k_j}(b^j)\|_2 \leqq \xi_1 \quad \text{for } k \in (0, 1), \qquad \|\nabla S^{k_m}(b^m)\|_2 \leqq \xi_2 \quad \text{at } k = 1. \tag{13}$$

In the examples of § 4 the convergence tests of (13) were used with values of $\xi_1 = 10^{-2}$ and $\xi_2 = 10^{-6}$. Dennis (1977) reports a very reliable test based upon a geometric interpretation of the structure of $\nabla S^{k_l}(b^j)$.

Once the critical point b^j for $k = k_j$ has been determined, it may be used together with the critical points for preceding values of the continuation parameter sequence to predict a starting point for an iteration at $k = k_{j+1}$. This prediction is then corrected by a Newton-type step described above. There are a number of ways in which this prediction may be achieved, depending upon

(1) whether the stepsize $\Delta k_j = k_{j+1} - k_j$ is constant or variable, and
(2) whether the vector $\partial b^j/\partial k$ is known at each point k_j on the continuation path.

While it is simpler to choose a uniform step size $\Delta k_j = 1/m$, reasons of computational efficiency make it attractive to implement an algorithm with a variable or adaptive stepsize. For a variable stepsize in k, prediction using Lagrangian polynomials may be used when only the critical points b^j are known, and Hermite extrapolation may be used when the critical points b^j and their corresponding derivatives $\partial b^j/\partial k$ with respect to the continuation parameter are known.

In order to investigate the value of applying a continuation method to the nonlinear least squares problem, the simpler case of a uniform stepsize $\Delta k = 1/m$ was chosen. It will be shown that for the continuation method suggested (as indeed for the variable stepsize case), the derivatives $\partial b^j/\partial k$ can be obtained with minimal additional computation. Therefore prediction in the examples of § 4 will be based upon a uniform stepsize in k and will use the derivatives $\partial b^j/\partial k$.

At $k = k_j$, evaluation of $\partial b^j/\partial k$ may be performed in the following manner. It is required that $S^k(b)$ be a minimum for all $k \in [0, 1]$, i.e.,

$$\nabla S^k(b) = 0 \quad \text{for all } k \in [0, 1].$$

Hence,

$$\frac{\partial}{\partial k}\nabla S^k(b) = J^T\left[\frac{\partial F^k(b)}{\partial k} + J\frac{\partial b}{\partial k}\right] + \sum_{i=1}^{n} F_i^k(b)\nabla^2 F_i(b)\frac{\partial b}{\partial k} = 0.$$

Substituting $\partial F^k(b)/\partial k = F(b^0)$ and solving for $\partial b^j/\partial k$ at b^j yields

$$\frac{\partial b^j}{\partial k} = -\left[J^T J + \sum_{i=1}^{n} F_i^{k_j}(b^j)\nabla^2 F_i(b^j)\right]^{-1} J^T F(b^0). \tag{14}$$

Comparison between (11) and (14) shows that the terms enclosed in the square brackets are identical and have already been evaluated on the most recent step to determine b^j, enabling $\partial b^j/\partial k$ to be obtained with minimal additional computation.

When the Gauss–Newton approximation is valid, (14) may be reduced to

$$\frac{\partial b^j}{\partial k} = -[J^T J]^{-1} J^T F(b^0). \tag{15}$$

At $k = 0$ the standard Euler predictor formula using b^0 and $\partial b^0/\partial k$ may be used to predict a starting value of b for k_1. For $k \geqq k_1$, the following predictor formula of Haselgrove (1961) has been found to work very well in practice and to be stable:

$$b^{j+1} = b^j + \left(1.5\frac{\partial b^j}{\partial k} - 0.5\frac{\partial b^{j-1}}{\partial k}\right)\Delta k, \qquad j = 1, 2, \cdots, \tag{16}$$

For $k \geqq k_3$ the Adams–Moulton predictor formula may be used (Scharmack (1968)).

The important remaining question to be considered is how to choose the stepsize Δk_j in k. For a small stepsize the prediction of b for the next value of k_j may be sufficiently accurate to ensure that a single Newton–Raphson or Gauss–Newton correction will satisfy the convergence test of (13), so that iteration is not necessary. If a larger stepsize is chosen, the prediction for b will not be as good as before (but still within the region of convergence of the iterative method) and it may be necessary to iterate on b in order to satisfy (13). Computing time saved by choosing a larger stepsize in k is then expended on iterating on b to ensure convergence of the intermediate problem. If the stepsize is too large, prediction for b may be outside the region of convergence of the iterative method and the algorithm will break down. Additional factors to be considered in the selection of stepsize are the number of unknowns being solved for, the complexity of the nonlinearity of the problem, the extent of the deformation introduced by the continuation parameter and the accuracy of the initial guess b^0 with respect to the original problem (which will determine how much deformation will take place during development of a solution). For most nonlinear problems, where less than about a dozen unknowns are to be solved for, the typical number of uniform steps in $k \in [0, 1]$ required for a satisfactory solution has been found to be between 10 and 40. Example 4.3 of § 4 is considered to be a more difficult problem, and for some very poor initial guesses a greater number of steps were required to obtain a solution.

The most efficient solution to the stepsize selection problem is to have a variable or adaptive stepsize which will adjust itself to satisfy the convergence requirements of the intermediate problems on the continuation path. Two approaches to the problem of achieving stepsize control for continuation with Newton's method are described by Rheinboldt (1975) and Deuflhard (1976). Convergence is also discussed in some detail. While the use of a constant stepsize has been adopted in this paper in order to demonstrate some of the advantages of applying a continuation method to the nonlinear least squares problem, the most fertile area for further work lies in applying dynamic stepsize control to the continuation algorithm to solve this problem.

The method as presented is dependent on the fact that solutions for Δb and $\partial b^j/\partial k$ do exist at all points along a continuation path. There are several complicating conditions which may either require special procedures or may cause the method to fail. For example the Jacobian of S^{k_j} for Gauss–Newton steps or the Hessian of S^{k_j} for Newton–Raphson steps may be rank deficient. The continuation path may also not be well behaved and may terminate prematurely or may split into several branches.

4. Examples of the application of the continuation method to the solution of the nonlinear least squares problem. Three examples of the application of the continuation method, described in § 3, to the solution of the nonlinear least squares problem are

discussed. The first two examples approximate nonlinear functions to synthetically generated data while the third example describes an actual problem.

In these examples comparisons were made between the performance of conventional and continuation methods using Newton–Raphson and Gauss–Newton iterative procedures. These iterative procedures were damped, and they incorporated line searching. Also included was a device to ensure that the objective function would progressively decrease in value with each iteration. The term "conventional method" is used to describe a method that does not incorporate continuation.

Example 4.1. The approximating function

$$f(x, b) = b_1 b_2^x \sin(b_3 x + b_4) \tag{17}$$

was used to test the continuation method. The data to which (17) was fitted, y_i, $i = 1, \cdots, 24$, was generated by choosing generating vector $b_{(g)} = [60.137, 1.371, 3.112, 1.761]$ and evaluating (17) for $b_{(g)}$ at 24 equispaced values of $x \in [0, 2.3]$. The generating vector $b_{(g)}$ is therefore a solution to the nonlinear least squares approximation problem. (Note that it is possible for there to be more than one solution.) Four poor starting vectors b^0 were selected and used to compare the performances of the conventional Newton–Raphson and Gauss–Newton methods with the performance of these two methods incorporating continuation. The initial vectors b^0 chosen are given in Table 1.

Since this is a synthesized problem, convergence to a solution occurs when the final solution vector b^m is equal to $b_{(g)}$ and the sum of squared errors is zero. Nonnull entries in Table 1 denote that convergence has been achieved; null entries indicate that convergence was not attained. In order to compare the relative amounts of computation required to obtain a solution for the different algorithms, the term "equivalent evaluations" is used. Each evaluation of the objective function counts as one equivalent evaluation. Evaluation of each of the r elements of ∇S accounts for an equivalent evaluation, as do each of the $r + r(r-1)/2$ elements of $\nabla^2 S$ (since $\nabla^2 S$ is symmetric). By making the assumption that the same amount of computation is required for each of these equivalent evaluations, the total number of equivalent evaluations required for each algorithm to converge may be summed and used as an approximate measure of the amount of computation required by that algorithm. Note that this is only a crude comparison since in general the elements of $\nabla^2 S$ require more computation than is required for S or ∇S.

For the four poor starting vectors of Table 1, the conventional Newton–Raphson and Gauss–Newton algorithms did not converge to a solution. The Newton–Raphson algorithm used with a continuation method did converge for the four cases listed, and the Gauss–Newton method used with continuation converged for all but the fourth initial vector b^0. The region of convergence of the Newton–Raphson and Gauss–Newton methods has therefore been enlarged, with respect to the initial vectors b^0, through use of the continuation method. In each case where an algorithm did not converge, the failure was caused by a prediction of $b_2 < 0$ which is not defined.

As would be expected, for initial vectors b^0 where continuation algorithms using the Newton–Raphson and Gauss–Newton iterative methods converged, less computation was required for the Gauss–Newton based algorithm than was required for the Newton–Raphson based algorithm.

An interesting feature of the continuation method is illustrated by the third case in Table 1, for the Newton–Raphson method with continuation. In the light of previous experience, 10 and 40 steps in $k \in [0, 1]$ were selected. For 10 steps in k, 33 evaluations of the Hessian were required, which indicates that the Newton–Raphson method was

TABLE 1.
Comparison between methods of solution for Example 4.1

Starting vector b^0	Conventional method		Continuation method							
	Newton–Raphson	Gauss–Newton	Newton–Raphson				Gauss–Newton			
			Steps in k	Hessian evaluations	Objective function evaluations	Equivalent evaluations	Steps in k	Jacobian evaluations	Objective function evaluations	Equivalent evaluations
[1.0, 8.0, 4.0, 4.412]	—	—	20	60	205	1,045	20	93	300	672
[1.0, 8.0, 8.0, 1.0]	—	—	40	70	258	1,238	40	102	350	758
[1.0, 8.0, 1.0, 4.412]	—	—	10	33	114	576	10	36	119	263
			40	44	175	791	40	85	296	636
[1.0, 8.0, 4.0, 1.0]	—	—	20	33	121	583		—		

required to iterate in order to find the critical point b^l at some or all of the points k_j on the continuation path. For 40 steps in k, 44 evaluations of the Hessian were required, signifying that the Newton–Raphson method did not need to iterate to determine b^l at intermediate points k_j on the continuation path. (One Newton–Raphson step was required to satisfy $\|\nabla S^{k_j}(b^l)\|_2 < 10^{-2}$ at each of the 41 points k_j, $j = 0, \cdots, 40$, and three iterations were required at $k = 1$ in order to satisfy the more stringent condition $\|\nabla S^{k_m}(b^m)\|_2 < 10^{-6}$. For the large difference in the number of steps in k (40 compared with 10), the amount of computation required to attain a solution was comparable (791 and 576 equivalent evaluations respectively). The objective of work on variable or adaptive stepsize continuation methods is to determine a compromise between computation requirements to iterate at intermediate points k_j on the continuation path for stepsizes which are large, and computation requirements to determine critical points for an excessive number of these intermediate points k_j.

This example also highlights an additional advantage gained by using a continuation method to ensure that b remains close to an intermediate solution for all $k \in [0, 1]$. For a poor b^0 the conventional method generates a large change Δb in b and it is easy for b_2 to become negative. Since b_2^x is not defined for $b_2 < 0$ the algorithm breaks down at this point. By ensuring that changes in b are kept small the continuation method provides a docile and well behaved algorithm.

Example 4.2. The continuation method was tested on a rather more difficult nonlinear approximating function

$$f(x, b) = b_1 b_2^x (\tanh b_3 x + \sin b_4 x) \cos (x e^{b_5}). \tag{18}$$

As before the data y_i, $i = 1, \cdots, 16$, to which (18) was to be fitted was generated by choosing $b_{(g)} = [53.81, 1.27, 3.012, 2.13, 0.507]$ and evaluating (18) for $b_{(g)}$ at 16 equispaced values of $x \in [0, 1.5]$. (Note that the values of the independent variable need not be equispaced.) As explained in Example 4.1 the vector $b_{(g)}$ is a solution to this synthetic problem.

This problem is more difficult than the previous problem. In order to economize on computation it would therefore be advantageous to be able to use an algorithm in which the stepsize in k is smaller near $k = 0$ in order to rapidly establish a trace of the continuation path, with the stepsize becoming larger near $k = 1$, where the continuation path is well established and more accurate predictions can be made. Deuflhard (1976) describes a way in which such a variable stepsize algorithm may be implemented. A much simpler though less flexible way in which the nonuniform stepsize may be effectively simulated, while retaining a uniform stepsize algorithm, is to introduce k in such a way that the changes in S due to an increment Δk in k are much smaller near $k = 0$ than they are near $k = 1$. One way in which this can be done is to further modify the residual vector of (9) to the following,

$$F^k(b) = F(b) + (k^q - 1)F(b^0), \tag{19}$$

where q is a positive integer. This is equivalent to using the objective function of (10) with a stepsize of $k_j = (j/m)^q$, $j = 0, \cdots, m$. The value of q should not be too large as the value of the objective function becomes "steep" with respect to k as k approaches unity. In practice, values of q up to 4 have been found to work successfully.

Comparison between the results obtained using the conventional and continuation implementations of the Newton–Raphson and Gauss–Newton iterative methods for six poor initial vectors b^0 is shown in Table 2.

The conventional Newton–Raphson algorithm converges for only the first of the initial vectors of Table 2, while the conventional Gauss–Newton algorithm fails to

TABLE 2.
Comparison between methods of solution for Example 4.2

Starting vector b^0	Conventional method				Continuation method									
	Newton–Raphson			Gauss–Newton	Newton–Raphson				Gauss–Newton					
	Hessian evaluations	Objective function evaluations	Equivalent evaluations		q	Steps in k	Hessian evaluations	Objective function evaluations	Equivalent evaluations	q	Steps in k	Jacobian evaluations	Objective function evaluations	Equivalent evaluations
[45.0, 2.0, 2.5, 1.5, 0.9]	15	50	350	—	3	20	30	111	711	3	20	22	87	197
					3	40	40	161	961	3	40	41	164	369
[42.0, 0.8, 1.4, 1.8, 1.0]		—		—	1	40	49	194	1,174	1	40	33	142	307
[45.0, 2.0, 2.1, 2.0, 0.9]		—		—	3	20	24	93	573	3	20	22	87	197
[45.0, 2.5, 1.7, 1.0, 1.0]		—		—	3	40	55	206	1,306	3	40	73	260	625
					3	20	55	191	1,291	3	20	48	165	405
[35.0, 2.5, 1.7, 1.0, 1.0]		—		—	3	20	57	203	1,343	3	20	45	156	381
					3	40	53	201	1,261	3	40	69	248	593
[42.0, 0.8, 1.8, 3.15, 1.0]		—		—	3	10	42	137	977			—		
					3	50	52	208	1,248			—		

converge for any of the listed initial vectors b^0. The Newton–Raphson method with continuation converges for all six of the given initial vectors. For the Gauss–Newton algorithm with continuation, convergence was attained for all the initial vectors listed except the last. For the given initial vectors continuation, therefore, does greatly enlarge the region of convergence of the Newton–Raphson and Gauss–Newton iterative methods. For the first initial vector of Table 2, the results confirm the expectation that when both the conventional and continuation algorithms converge for a given initial vector, the conventional algorithm is computationally more efficient. However, the continuation method proves superior for poor initial vectors in that it often yields a solution when the corresponding conventional algorithms fail to do so. In the long run, use of the continuation method may also prove more efficient if it results in trying fewer initial guesses before convergence is attained. Note that as for Example 4.1, the amount of computation required for the different stepsizes shown does not vary widely for a given initial vector, indicating that 20 to 40 steps in $k \in [0, 1]$ is a reasonable choice for this example.

Example 4.3. This example compares the application of the conventional and continuation Gauss–Newton methods for the solution of a difficult practical problem using actual measured data. The problem arises from work done by D. Williams (1976) on the estimation of liquid-phase kinetics. Williams used temperature as the measured variable to estimate the heat of reaction and rate constants for saponification of glycol diacetate by aqueous sodium hydroxide.

The behavior of the system is described by the following set of nonlinear differential equations:

$$\begin{aligned} \dot{z}_1 &= -A_1 z_1 z_2, \\ \dot{z}_2 &= -A_1 z_1 z_2 - A_2 z_2 z_3, \\ \dot{z}_3 &= A_1 z_1 z_2 - A_2 z_2 z_3, \\ C_p \dot{z}_4 &= b_3 A_1 z_1 z_2 + b_4 A_2 z_2 z_3. \end{aligned} \tag{20}$$

Let z denote the state vector z_i, $i = 1, \cdots, 4$. The system (20) may be expressed in vector form as

$$\dot{z} = \phi(z, b, t),$$

where ϕ is a 4-vector and t is the time.

Temperature rise is denoted by z_4. The heats of reaction are b_3 and b_4 cal/g mol, the temperature dependent rate constants are $A_1 = \exp(b_1 - b_5/z_4)$ and $A_2 = \exp(b_2 - b_6/z_4)$ liters/g mol min. and C_p is the total heat capacity of the reactor contents. The vector of approximation parameters b is a 6-vector. Eight experimental time versus temperature-rise curves were recorded from eight runs at various initial concentrations and temperatures. The problem is to determine b to simultaneously approximate, in a least squares sense, the family of eight temperature-rise curves by the component $z_4(t)$ of the solution of (20).

Two problems require resolution before approximation can commence. The first problem is to decide upon suitable initial conditions for z for use in the numerical integration of (20). The method used is described in Appendix A. The second problem arises from the tendency of the approximating solution $z_4(t)$ to fit more closely runs containing a larger number of experimental points or runs in which the temperature rise was higher. This problem is overcome by weighting the terms in each squared

residual of the objective function for run p, $p = 1, \cdots, 8$ by the factor $1/\omega_p$ where

$\omega_p = (\text{number of measurements in run } p) \times (\text{maximum temperature rise in run } p)$

Suppose n_p is the number of measurements in run p. Let t_{ip}, $i = 1, \cdots, n_p$, $p = 1, \cdots, 8$ be the time of the ith measurement of run p and let u_{ip} be the weighted measured temperature rise at t_{ip} for run p. Let $z_4^*(t_{ip}) = z_4(t_{ip})/\omega_p$ be the weighted value of component z_4 of the solution $z(t)$ of (20).

Let F_{ip} be the ith residual of run p, i.e.,

$$F_{ip}(b) = z_4^*(t_{ip}) - u_{ip}, \qquad i = 1, \cdots, n_p. \tag{21}$$

Let the vector $F_p = [F_{1p}, F_{2p}, \cdots, F_{n_p p}]$, $p = 1, \cdots, 8$ be the vector of residuals for all points of measurement of run p. The least squares objective function is

$$S(b) = \frac{1}{2} \sum_{p=1}^{8} F_p^T F_p.$$

Let $J_p = F_p'$, $p = 1, \cdots, 8$. To minimize $S(b)$, the Newton–Raphson step is given by the analogue of (6).

We may make the same assumptions as those following (6) to arrive at the analogue of (8), the Gauss–Newton method. The terms J_p in the analogue of (8) were obtained by solving the additional differential equations

$$\frac{\partial \dot{z}}{\partial b} = \left(\frac{\partial z}{\partial b}\right)\left(\frac{\partial \phi}{\partial z}\right) + \left(\frac{\partial \phi}{\partial b}\right), \tag{22}$$

simultaneously with equations (20) for each run p. The initial conditions for (22) were obtained from the initial conditions of (20) as described in Appendix A.

Applying the Gauss–Newton method analogous to (8) requires very good starting values to obtain convergence. For example, a starting vector $b^0 = [19.0, 40.7, 1{,}335.0, 8{,}668.0, 4{,}800.0, 10{,}850.0]$ converged to a solution $b = [19.7323, 47.3471, 13{,}990.2870, 8{,}743.2315, 4{,}938.0251, 12{,}709.3880]$, while a starting vector $b^0 = [17.1, 36.63, 12{,}195.0, 7{,}802.1, 4{,}320.0, 9{,}765.0]$, whose components are 10% less than those of the successful b^0 used above, failed to converge. Many local minima were also encountered. A contributory factor in the difficulties may have been the neglect of the second derivative terms in (6) to arrive at the Gauss–Newton method.

A continuation method was then used to solve this problem. The residuals were defined as in (21), and the following modified residual vectors were used

$$F_p^k(b) = F_p(b) + (k^q - 1)F_p(b^0), \qquad p = 1, \cdots, 8. \tag{23}$$

The objective function is

$$S^k(b) = \frac{1}{2} \sum_{p=1}^{8} (F_p^k)^T F_p^k. \tag{24}$$

Proceeding as before, the Newton–Raphson step is given by the analogue of (11).

An important characteristic of the continuation method is that it remains close to a solution for all $k \in [0, 1]$. Hence, if the stepsize Δk_j is sufficiently small the modified residuals $F_{ip}^k(b)$ of (23) will in general be smaller than the residuals $F_{ip}(b)$ of (21). There is therefore stronger justification for neglecting the terms $\sum_{i=1}^{n_p} F_{ip}^k \nabla^2 F_{ip}$ (in comparison to $J_p^T J_p$) in the analogue of (11) than there was for neglecting these second

derivative terms in the analogue of (6). The analogue of (11) will then reduce to the following Gauss–Newton step:

$$b_{l+1} = b_l - \left[\sum_{p=1}^{8} J_p^T J_p\right]^{-1} \sum_{p=1}^{8} J_p^T F_p^k(b_l), \qquad l = 1, 2, \cdots. \tag{25}$$

The equations to solve for $\partial b/\partial k$ may be derived in a similar manner to that described in § 3, by developing the equation $(\partial/\partial k)(\partial S^k(b)/\partial b) = 0$. Using the same argument as above to justify neglecting the terms containing $\nabla^2 F_{ip}$ (in comparison to $J_p^T J_p$), results in the following equation for $\partial b/\partial k$:

$$\frac{\partial b}{\partial k} = -qk^{q-1}\left[\sum_{p=1}^{8} J_p^T J_p\right]^{-1} \sum_{p=1}^{8} J_p^T F_p(b^0). \tag{26}$$

Note that the term $[\sum_{p=1}^{8} J_p^T J_p]^{-1}$, once evaluated in (25) to determine the critical point b^l at k_j, may be used again in (26) to solve for $\partial b^l/\partial k$ at k_j, resulting in a saving of computation.

For the eight diverse "poor" initial vectors b^0 listed in Table 3, the continuation method converged to a solution $b^m = [24.13, 15.31, 11{,}922.20, 13{,}368.40, 6{,}108.85, 3{,}694.55]$, which yielded a weighted sum of squared errors of 0.0008. No local minima were encountered. The fact that eight rather different initial vectors for the continuation method all converged to the same solution, which is different from the solution obtained by the conventional method and yields a smaller value of the objective function, suggests that the conventional method may have yielded a local minimum when using the starting vector $b^0 = [19.0, 40.70, 13{,}550.0, 8{,}668.0, 4{,}800.0, 10{,}850.0]$.

TABLE 3.
Comparison between methods of solution for Example 4.3

Starting vector b^0	Conventional Gauss–Newton method	Continuation Gauss–Newton method		
		Result	Order of method (q)	No. of steps in k-space
[1.9, 4.0, 1000.0, 500.0, 1085.0, 480.0]	No convergence	Convergence to solution	4	40
[−3.0, −1.5, 1000.0, 1000.0, −414.0, −750.0]	No convergence	Convergence to solution	4	60
[1.9, 4.0, 1355.0, 867.0, 480.0, 1085.0]	No convergence	Convergence to solution	2	100
[1.9, 4.0, 2000.0, 1000.0, 1085.0, 480.0]	No convergence	Convergence to solution	2	500
[3.0, −1.5, 8000.0, 8000.0, −414.0, −750.0]	No convergence	Convergence to solution	2	40
[4.75, 10.17, 3388.0, 2167.0, 1200.0, 2713.0]	No convergence	Convergence to solution	1	40
[9.5, 20.3, 6776.0, 4334.0, 2400.0, 5426.0]	No convergence	Convergence to solution	1	20
[1.9, 4.0, 8000.0, 8000.0, 1085.0, 480.0]	No convergence	Convergence to solution	2	100

Using each of the eight initial vectors b^0 listed in Table 3, the conventional Gauss–Newton method failed to converge to a solution. This clearly demonstrates the ability of the continuation method to increase the region of convergence of the Gauss–Newton method with respect to initial vectors b^0, for this problem. Note that for the continuation method, there was no iteration at intermediate values of k_j, $j = 0, \cdots, m$ so that the number of steps in k space also represents the number of evaluations of the term $[\sum_{p=1}^{8} J_p^T J_p]^{-1}$.

Using the solution vector $b^m = [24.12, 15.31, 11{,}922.20, 13{,}368.40, 6{,}108.35, 3{,}694.55]$, (20) was solved for each of the eight runs and the solution $z_4(t)$ for each run was plotted (in Williams (1976)) against the corresponding measured temperature-rise. A good fit was obtained. Factors affecting the closeness of fit of the family of curves to the measured data include the fact that some runs contained more observations than other runs and the effects of inaccuracies arising from estimation of the initial values of the state variable z in (20). The stronger justification for neglecting the second derivative terms $\nabla^2 F_{ip}$ in the continuation method contributes to the superior convergence of this method over the conventional method.

5. Discussion. Many iterative regression methods only perform well near a solution. Further from the solution they often exhibit a slow rate of convergence and may even diverge. The continuation method proves an effective way of ensuring that a computing algorithm remains close to a solution for a wide range of initial vectors. The region of convergence of the algorithm is effectively increased.

This closeness to a solution also ensures good numerical behavior of the method since the arguments of nonlinear functions such as exponentials do not become excessively large. For instance, in Example 4.3 the numerical integration of (20) would become unstable if the exponential terms were to become too large due to an unfortunate Gauss–Newton prediction.

Example 4.3 also demonstrates the value of the continuation method in solving a practical problem. The method attained its design objective of greatly enlarging the region of convergence of initial vectors, and strengthened the justification for neglecting the second derivative terms in deriving the Gauss–Newton step, which provides a more accurate algorithm which results in better convergence with fewer local minima.

In exchange for the effective increase in the region of convergence for initial vectors is usually an increase in the amount of computation when starting from a given initial vector. The relative amounts of computation for conventional and continuation methods vary from case to case. For initial vectors where both methods converge, the continuation method invariably requires more computation. However, in cases where the continuation method succeeds when the conventional method fails, the conventional algorithm may require more initial guesses before convergence is attained, resulting in more overall computer and researcher time than would be required using the continuation method. The fact that the continuation method succeeds in many cases where the conventional method fails is a strong recommendation for use of the continuation method. The continuation method proves useful for practical problems such as Example 4.3 where it may be difficult to determine beforehand, from a knowledge of the problem, what a suitable initial guess might be.

The purpose of this paper is to demonstrate the value of applying continuation methods to the solution of nonlinear least squares problems. Newton-type methods were used to solve the local minimization problems. There are more efficient minimization methods, for example derivative free methods that only require evaluation of the objective function and not Jacobians or Hessians. Since these methods also have

regions of convergence, continuation would probably similarly assist them to effectively increase their respective regions of convergence.

In order to make the continuation method more attractive, it would be desirable to reduce the amount of computation required by the method. The most fruitful area of research for achieving this goal would be in providing a variable stepsize strategy. This strategy should balance dynamically the contradictory needs of a sufficiently small stepsize so that no iteration is required in the intermediate problems, with the desire to minimize the total number of steps required for the continuation method. Deuflhard (1976) has demonstrated a stepsize control for use with the damped Newton method using multiple shooting techniques to solve boundary value problems on nonlinear differential equations.

Appendix A. A continuation method for nonlinear regression. The experiment performed by Williams (1976) consisted of injecting a small amount of one reactant in a concentrated form into a dilute solution of the other. The reaction rate was large so that the time taken for the initial mixing was not a negligible part of the experiment. Furthermore, there was a heat of dilution associated with the mixing of the substances. As a result, the concentrations z_1, z_2, z_3 were known at the time of injection, but not the initial temperature z_4. At a later time, when the initial disturbances had died down, the value of the temperature was known but not the concentrations. However, as the temperature change was not large within a single run, the following procedure was adopted:

The first three equations were integrated for a time period up to the point when the disturbances had died out, using an average value of temperature. These values with the measured value of the temperature were now used as the "initial" conditions for (20). Exactly the same procedure was used on the influence equations since using this procedure the "initial" conditions were of course functions of the parameters b. This effect needed to be taken into account in order to ensure an accurate calculation of the influence coefficients obtained from (22).

REFERENCES

Y. BARD (1974), *Non-linear Parameter Estimation*, Academic Press, New York.

———, (1970), *Comparison of gradient methods for the solution of nonlinear parameter estimation problems*, this Journal, 7, pp. 157–186.

K. W. BRODLIE (1977), *Unconstrained minimization*, in The State of the Art in Numerical Analysis, D. A. H. Jacobs, ed., Academic Press, London, pp. 229–268.

D. L. CRESWELL AND W. R. PATERSON (1976), *Weighted residual methods for parameter estimation* in Chemical Reaction Engineering, Dechema Frankfurt Am Main, I, pp. 21–30.

F. H. DEIST AND L. SEFOR (1966), *Solutions of systems of non-linear equations by parameter variation*, Comput. J., 10, pp. 78–82.

J. E. DENNIS (1977), *Non-linear least squares and equations*, in The State of the Art in Numerical Analysis, D. A. H. Jacobs, ed., Academic Press, London, pp. 269–312.

P. DEUFLHARD (1976), *A stepsize control for continuation methods with special application to multiple shooting techniques*, Rep. Technische Universität München, Institut für Mathematik, TUM-MATH-7627.

D. GLASSER (1969), *Numerical solution of two-point boundary value problems on total differential equations*, this Journal, 6, pp. 591–597.

D. GLASSER AND N. DE VILLIERS (1974), *Parameter variation for the solution of two-point boundary value problems and applications in the calculus of variations*, J. Optim. Theory Appl., 13, pp. 164–178.

C. B. HASELGROVE (1961), *Numerical solution of two-point boundary value problems on tdtal differential equations*, Comput. J., 4, pp. 255–259.

W. C. RHEINBOLDT (1975), *An adaptive continuation process for solving systems of non-linear equations*, University of Maryland, Computer Science Dept., Tech. Rep. TR-393.

D. K. SCHARMACK (1968), *Trajectory optimization problems*, in Advances in Control Systems, Theory and Applications, vol. 5, C. T. Leondes, ed., Academic Press, New York.

J. H. SEINFELD AND L. LAPIDUS (1974), *Mathematical Methods in Chemical Engineering, vol.* III, *Process Modelling, Estimation, and Identification*, Prentice-Hall, Englewood Cliffs, NJ.

W. E. STEWART AND J. P. SORENSON (1976), *Sensitivity and regression of multicomponent reaction models* in Chemical Reaction Engineering, I, Dechema Frankfurt Am Main, pp. 12–20.

D. A. F. WILLIAMS (1976), *The use of temperature as a measured variable in the estimation of liquid-phase kinetics*, M.Sc. Dissertation, University of the Witwatersrand, Johannesburg, South Africa.