Solution of systems of non-linear equations by parameter variation

By F. H. Deist and L. Sefor*

This paper describes a novel method for solving systems of non-linear equations. Conceptually the procedure may be described as follows. The equations are modified to a form that can be handled analytically by introducing a set of parameters. These simple equations are then varied to their original form whilst simultaneously tracing the roots. Formal background, computational details and applications are considered.

1. Introduction

The problem of finding the solutions of non-linear equations is an important one occurring frequently in diverse fields of study. The approach discussed in this paper was originally developed with the aim of solving systems of transcendental equations where it was successfully employed. It has been indicated to the authors that the idea underlying the method was first described by Davidenko (1953).

The basis of the method is discussed in Section 2. Section 3 deals with uniqueness properties and contains some remarks on the selection of suitable parameters. Variations of the method are considered in Section 4. Based on these a simple computational technique is proposed. Practical details of the procedure along with numerical examples are presented in Section 5.

2. Basis of method

Let

$$F(X) = \{f_r(x_p)\} = 0, \quad (r, p = 1, 2, \ldots n) \tag{2.1}$$

be the system of non-linear equations to be solved. We introduce a set of parameters $a_s (s = 1, 2 \ldots k; k$ arbitrary) into equations (2.1), with the property that there is a particular set of values—a_s^f say—for which $F(X) \equiv F(X, a_s^f)$. As the a_s take on different values, the zeros of

$$F(X, a_s) = 0 \tag{2.2}$$

will move in X-space.

A suitable choice of the parameters in conjunction with a particular set of values—a_s^0 say—will allow one to find analytically m solutions of (2.2), $X_j^0 (j = 1, 2 \ldots m)$ say.

Starting with these parameter values and one of the m initial vectors X_j^0, the system of equations (2.2) is changed to its original form by a continuous variation of the a_s to their final values a_s^f. Simultaneous tracing of X, such that (2.2) is always satisfied during the deformation, will—subject to conditions discussed below—yield a unique zero of (2.1). Repeating this procedure for all X_j^0 will determine m zeros of the original set of equations.

The above deformation may formally be described as follows: Differentiating (2.2) w.r.t. one of the parameters a_s, one has:

$$G \cdot \frac{\partial}{\partial a_s}(X) + \frac{\partial}{\partial a_s}(F) = 0 \tag{2.3}$$

where

$$G = \left\{\frac{\partial f_r}{\partial x_p}\right\} \quad (r, p = 1, 2 \ldots n).$$

Solving for $\dfrac{\partial}{\partial a_s}(X)$:

$$\frac{\partial}{\partial a_s}(X) = -G^{-1} \cdot \frac{\partial}{\partial a_s}(F), \quad (s = 1, 2 \ldots k). \tag{2.4}$$

It is seen that the task of tracing X from X_j^0 to X_j^f amounts to the solution of k systems of first-order differential equations in the interval $a_s^0 < a_s < a_s^f$ subject to initial conditions X_j^0 at a_s^0.

Provided that certain conditions, to be discussed in the next section, are satisfied, each $X_j^0 (j = 1, 2 \ldots m)$ will result in a unique distinct solution of the differential equations, which is independent of the path of integration in a_s-space.

3. Remarks on uniqueness of solutions and selection of parameters

To establish the uniqueness conditions referred to in Section 2, (2.3) is differentiated w.r.t. a_t, say. One obtains:

$$\frac{\partial f_r}{\partial x_p} \cdot \frac{\partial^2 x_p}{\partial a_t \partial a_s} = -\left(\frac{\partial^2 f_r}{\partial x_p \partial x_q} \cdot \frac{\partial x_p}{\partial a_s} \cdot \frac{\partial x_q}{\partial a_t} + \frac{\partial^2 f_r}{\partial a_t \partial x_p} \cdot \frac{\partial x_p}{\partial a_s} + \frac{\partial^2 f_r}{\partial x_p \partial a_s} \cdot \frac{\partial x_p}{\partial a_t} + \frac{\partial^2 f_r}{\partial a_t \partial a_s}\right). \tag{3.1}$$

Restricting interest to functions f_r, which are twice differentiable w.r.t. the x_p as well as the parameters a_s, it is seen from (3.1) that, if the m solution sheets in (X, a_s)-space, generated by varying the path of integration in a_s-space and identified by the m distinct initial vectors $X_j^0 (j = 1, 2 \ldots m)$, are free of singularities, i.e. $|G| \neq 0$, m unique solutions that are independent of the path of integration, will result. Extending the picture

* *Department of Electrical Engineering, University of the Witwatersrand, Johannesburg, South Africa.*

to solution sheets that intersect (i.e. some of the $\frac{\partial x_p}{\partial a_s}$ are of the form $\frac{0}{0}$) and/or contain singularities that leave the sheets single-valued within the region of interest, m starting vectors will still result in m unique solutions, if nonsingular paths of integration connecting the m starting and corresponding end points can be found.

From what has been said, it is evident that in general there can be no formal way of introducing the parameters. The success of the method depends entirely on the ingenuity of the user. However, when dealing with a practical system, it is frequently feasible to find special case solutions, when certain system parameters take on particular values. If it is further known that the system depends continuously on these parameters, they will constitute the natural choice for the method. (See Section 5 (*b*) (ii).)

4. Remarks on computational procedures

When implementing the method on a digital machine, we require algorithms, which will follow the zeros as the system of equations undergoes stepped deformations.

Differential equation approach

The first approach is of course suggested by equations (2.4) and puts at our disposal the host of algorithms which have been devised for the solution of systems of first-order differential equations. The Jacobian matrix must be available.

Repetitive local iteration procedures

A distinct alternative approach utilizes any of the available iteration methods for the solution of equations in tracing a zero from the rth to the $(r+1)$th deformed system of equations. Clearly the deformation step size must be suited to the radius of convergence of the technique employed, which also determines the order of derivatives required.

Mixed method

Assessing these two methods with a view to further alternatives, it is evident that the differential equation approach does not utilize the information available in the form of equations (2.2), which must be satisfied everywhere along the path of integration. The requirements on the step size can be relaxed, if the integration is followed by or intermixed with local iteration runs. These do recognize equations (2.2) and eliminate the errors introduced by the preceding integration steps.

The difficulty is to strike an efficient compromise between step size and the number of integration steps between iteration runs. The two basic methods mark the extremes of possible combinations.

To overcome this dilemma, one would like an algorithm which embodies the properties of integration and local iteration methods. With regard to the latter, it is evident that most available techniques are unnecessarily elaborate. All we require is a method, which converges rapidly in the close vicinity of a zero.

If we restrict ourselves to functions whose Jacobian matrix is available, the obvious choice is the generalized Newton–Raphson method, defined as follows:

$$\Delta X^i = -(G^i)^{-1}.F^i \tag{4.1a}$$

$$X^{i+1} = X^i + \Delta X^i. \tag{4.1b}$$

The quantity $S_i = \sqrt{[(\Delta X^i)^T \Delta X^i]}$ can serve as a measure of convergence. (i is the iteration index.)

Formally the above algorithm corresponds to the second basic approach. However, we shall show that the first step in every iteration series is like an integration step of the differential equation approach.

Consider equations (2.4) subjected to an integration step w.r.t. one of the parameters, a_s say.

Then

$$\Delta X \doteqdot -G^{-1}.\left(\Delta a_s.\frac{\partial}{\partial a_s}(F)\right). \tag{4.2}$$

Let this correspond to the step from the rth to the $(r+1)$th zero. Now, $F = 0$ at the rth zero. When evaluated at the same point after deforming the system to its $(r+1)$th state, let $F = F^0$. Then by definition of $\frac{\partial}{\partial a_s}(F)$.

$$\Delta a_s.\frac{\partial}{\partial a_s}(F) \doteqdot F^0 - 0 = F^0.$$

Hence (4.2) takes the form

$$\Delta X \doteqdot -G^{-1}.F \tag{4.3}$$

where F is evaluated at the rth zero in the $(r+1)$th equations.

A numerical integration procedure evaluating G in the same manner, could be employed and would of course amount to proper integration in the limit as $\Delta a_s \to 0$. On the other hand, comparing (4.3) and (4.1*a*) we see that the former is equivalent to the first step of the iteration process defined by the latter. This technique has been used on a variety of problems and seems to constitute the best compromise for the mixed method.

5. Computational procedure

(*a*) Programming details

In all examples the computation was carried out on an IBM 1620 Model II computer using FORTRAN II. Both the differential equation approach and the mixed method have been employed. For the former, we found it convenient to use the Runge-Kutta method (Ralston and Wilf, 1960), because the parameter increment could be easily modified to ensure a reasonable rate of convergence. However, the latter was found to be more efficient in running time and programming effort for all cases tested, so, in **Fig. 1**, we present a flow chart of the mixed method program.

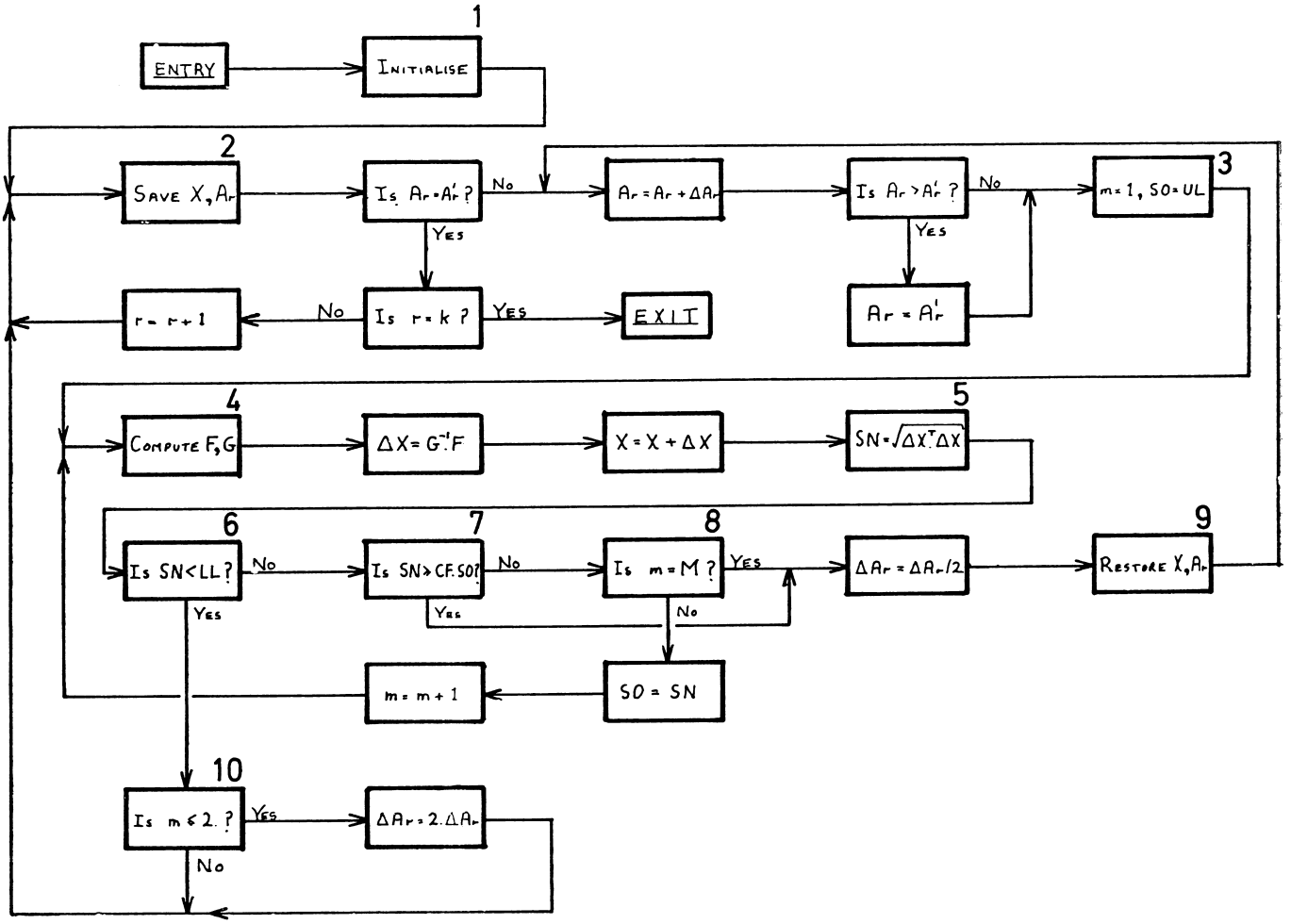

Fig. 1. Flow chart for mixed method

Description of flow chart

Box 1

The following quantities are defined for use in the program:

A_r	Parameter vector, initialized to starting values	$r = 1, 2 \ldots K$
ΔA_r	Parameter increment vector, initialized to starting values	
$A^1{}_r$	Final value of parameter vector	
X_i	Unknown vector, initialized to starting value	$i = 1, 2 \ldots N$
ΔX_i	Unknown increment vector	
M	Maximum number of iterations allowed before adjusting ΔA_r	
r	Parameter count set equal to one	

Box 2

Values of X and A_r are saved for recalculation with new ΔA_r, if convergence rate is not satisfactory. (See Box 9.)

Note that in the absence of a suitable criterion we have quite arbitrarily chosen to make a full change in A_r before altering A_{r+1}. Not having attempted any other scheme we cannot comment on relative merits.

Box 3

The iteration counter m is set to one and the quantity SO is set to an arbitrary upper limit UL to ensure that the initial ΔX does not run away.

Box 4

The function vector F and Jacobian matrix G are evaluated using the current values of X and A. An emergency exit is provided for in the event of G becoming singular.

Box 5

The quantity SN which is the absolute distance from the zero of the current system is computed.

Box 6

If the absolute distance from the zero is less than an

arbitrary lower limit, control is transferred to Box 10. If not, control is transferred to Box 7.

Box 7

A test is made for satisfactory convergence where CF is an arbitrary convergence factor $0 < CF < 1$. If convergence is satisfactory, control is transferred to Box 8. If not, the parameter increment ΔA_r is halved and control is transferred to Box 9.

Box 8

The iteration count m is tested to ensure that the limit M has not been exceeded. If $m = M$, ΔA_r is halved and control transferred to Box 9. If $m < M$, SO is set equal to SN, the iteration count is incremented and control is transferred to Box 4.

Box 9

The saved values of X and A_r are restored and the computation proceeds with a new, smaller ΔA_r.

Box 10

If the iteration count is one or two it can reasonably be expected that the system is well behaved and the increment ΔA_r is doubled. This ensures a rapid deformation to the original set of equations. Control is transferred to Box 2.

The un-numbered boxes are self evident in the function they perform.

(*b*) Numerical examples

(i) *Polynomial equations*

Consider the equations

$$\left.\begin{aligned} x_1^2 + x_2^2 + x_3^2 &= 5 \\ x_1 + x_2 &= 1 \\ x_1 + x_3 &= 3. \end{aligned}\right\} \qquad (5.1)$$

There are two sets of solutions.

$$X_1 = \begin{bmatrix} 1\frac{2}{3} \\ -\frac{2}{3} \\ 1\frac{1}{3} \end{bmatrix} \quad \text{and} \quad X_2 = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}.$$

Table 1

Solution of polynomial equations

STARTING VALUES	TOTAL NUMBER OF ITERATION STEPS	FINAL VALUES
$\sqrt{5}$ $1 - \sqrt{5}$ $3 - \sqrt{5}$	19	1·666667 −0·666667 1·333333
$-\sqrt{5}$ $1 + \sqrt{5}$ $3 + \sqrt{5}$	101	1·000000 $-0{\cdot}8 \times 10^{-8}$ 2·000000

We introduced parameter a in the following way:

$$\left.\begin{aligned} x_1^2 + ax_2^2 + ax_3^2 &= 5 \\ x_1 + x_2 &= 1 \\ x_1 + x_3 &= 3 \end{aligned}\right\} \qquad (5.2)$$

and applied the mixed method using $a_0 = 0$, so the starting vectors were

$$X_1^0 = \begin{bmatrix} \sqrt{5} \\ 1 - \sqrt{5} \\ 3 - \sqrt{5} \end{bmatrix} \quad \text{and} \quad X_2^0 = \begin{bmatrix} -\sqrt{5} \\ 1 + \sqrt{5} \\ 3 + \sqrt{5} \end{bmatrix}.$$

The resultant solutions of equation (5.1) are presented in **Table 1**.

(ii) *Transcendental equations*

Consider the following set of equations:

$$f_i = \sum_{j=1}^{n} F_{ij} = 0 \ (i = 1, \ldots n) \qquad (5.3)$$

where $F_{ij} = \cot \beta_i x_j$ for $i \neq j$

$= 0$ for $i = j$.

Table 2

Solution of transcendental equations

i	β_i 1/cm	MIXED METHOD		FLETCHER & POWELL	
		x_i cm	f_i	x_i cm	f_i
1	$0{\cdot}2249 \times 10^{-1}$	121·97	$0{\cdot}9 \times 10^{-6}$	121·95	$-0{\cdot}1 \times 10^{-3}$
2	$0{\cdot}2166 \times 10^{-1}$	114·32	$0{\cdot}7 \times 10^{-6}$	114·29	$-0{\cdot}2 \times 10^{-3}$
3	$0{\cdot}2083 \times 10^{-1}$	93·80	$0{\cdot}1 \times 10^{-5}$	93·78	$-0{\cdot}2 \times 10^{-3}$
4	$0{\cdot}2000 \times 10^{-1}$	62·32	$0{\cdot}7 \times 10^{-6}$	62·33	$-0{\cdot}2 \times 10^{-3}$
5	$0{\cdot}1918 \times 10^{-1}$	41·07	$0{\cdot}3 \times 10^{-6}$	41·10	$0{\cdot}7 \times 10^{-4}$
6	$0{\cdot}1835 \times 10^{-1}$	33·33	$0{\cdot}1 \times 10^{-6}$	30·36	$0{\cdot}2 \times 10^{-3}$
		No. of steps = 13 Time: 1 m 56 sec		No. of iterations = 14 Time: 12 m 51 sec	

These equations are used in the design of a VHF aerial feeder system. The x_j are the lengths associated with the coaxial line connectors and the β_i are constants dependent on the carrier frequency. Of the many techniques tried, the mixed method was found to be the most efficient. Equations (5.3) were solved for $n = 6$. Parameter a was introduced such that

$$\beta_i = \beta_0 + a(\Delta\beta_i) \quad a^0 = 0, \quad a^f = 1$$

where β_0 and $\Delta\beta_i$ are constants.

For comparison, the solutions furnished by the method of Fletcher and Powell (1963) are included. Both sets of results are shown in **Table 2**. The same starting vectors were used. The mixed method was about 6 times faster than Fletcher and Powell's method. This seems to indicate that a method which has access to each residual independently will be more efficient than one which minimizes the sum of squared residuals.

References

DAVIDENKO, D. F. (1953). "On a new method of numerical solution of systems of non-linear equations", *Mathematical Reviews*, Vol. 14, p. 906.

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

RALSTON, A., and WILF, S. (1960). *Mathematical Methods for Digital Computers*, New York: Wiley.

An iterative method for locating turning points

By P. Jarratt*

A method for calculating turning points is given which is shown to possess superlinear convergence. The iterative formula is applied to a numerical example and the problem of accelerating convergence is discussed.

1. Introduction

The problem of computing a value θ for which a function f has a turning point occurs frequently in scientific work and is usually solved by applying an appropriate root-finder to the derivative f'. In many cases of practical interest, however, an analytic form for f' is unavailable or difficult to obtain and alternative techniques must therefore be sought. One method which suggests itself is to compute new approximations to θ by the use of a polynomial which interpolates f. Thus let $x_i, x_{i-1}, \ldots, x_{i-n}$ be $n + 1$ approximations to a turning point θ of f and let $P_n(t)$ be the interpolatory polynomial of degree n such that

$$P_n(x_{i-j}) = f(x_{i-j}), \quad j = 0, 1, \ldots, n.$$

Define a new approximation to θ by

$$P_n'(x_{i+1}) = 0, \tag{1.1}$$

and then repeat the procedure for $x_{i+1}, x_i \ldots, x_{i-n+1}$, and so on. It is clear that this approach presents a number of problems. Firstly, since (1.1) is a polynomial of degree $n - 1$, a polynomial equation must be solved at each step of the iteration, and additionally x_{i+1} will not in general be uniquely specified. Some rule must therefore be formulated whereby x_{i+1} is chosen uniquely as one of the zeros of the polynomial. Secondly, it is not even certain that (1.1) has a real root in the region of θ. These objections can be met, however, if we restrict ourselves to a formulation in which (1.1) is linear in x_{i+1}, corresponding to interpolation by the quadratic $P_2(t)$. In this paper the properties of the corresponding iteration function are investigated and its behaviour is illustrated by a numerical example.

* *University of Bradford, Bradford, Yorks.*

2. Formulation

Following the previous discussion, we fit the quadratic

$$y = a + bt + ct^2 \tag{2.1}$$

to three points (x_{i-j}, f_{i-j}), $j = 0, 1, 2$, and then predict x_{i+1} by imposing the condition $y'_{i+1} = 0$. This leads to the system

$$\begin{aligned} f_{i-j} &= a + bx_{i-j} + cx_{i-j}^2, \quad j = 0, 1, 2 \\ 0 &= \quad b \quad + 2cx_{i+1}, \end{aligned} \tag{2.2}$$

and these four equations in the three parameters a, b, c, will be consistent provided that the determinantal condition

$$\begin{vmatrix} 2x_{i+1} & 1 & 0 & 0 \\ x_i^2 & x_i & 1 & f_i \\ x_{i-1}^2 & x_{i-1} & 1 & f_{i-1} \\ x_{i-2}^2 & x_{i-2} & 1 & f_{i-2} \end{vmatrix} = 0, \tag{2.3}$$

is satisfied. We now use (2.3) to examine the convergence of the method. First we define the errors in the