where n is a positive integer. Equation (16) agrees with the results obtained by solving Eq. (15) using separation of variables.

As a second example, consider the transverse vibration of a simply supported bar that is given by Eq. (1) with $L_1=0$ and $L_2=a(\partial^4/\partial x^4)$. The domain $D(L)$ becomes the set of all real functions $u(x,t)$ such that u and its derivatives up through order four are continuous in $L^2(0,1)$ and satisfy the boundary conditions $u(0)=u(1)=u_{xx}(0)=u_{xx}(1)=0$. If a control force of the form

$$f(x,t)=-bu_t(x,t)+c\frac{\partial^2}{\partial x^2}u(x,t)$$

is used then $L_3=-bI$ and $L_4=c(\partial^2/\partial x^2)$. Again, each operator is self-adjoint and has a compact resolvent. A short calculation shows that Eq. (6) is satisfied and the parameters b and c can be chosen as control parameters to shape the response without using modal truncation.

Conclusion

Necessary conditions for a given nonconservative structure and feedback control system to be both externally and internally decoupled have been presented. These conditions allow the method of independent modal-space control to be applied to a larger class of structures.

Acknowledgment

This work was performed under Air Force Office of Scientific Research Grant AFOSR820242.

References

[1]Meirovitch, L. and Baruh, H., "Control of Self-Adjoint Distributed Parameter Systems," *Journal of Guidance, Control and Dynamics*, Vol. 5, Jan.-Feb. 1982, pp. 60-66.

[2]Inman, D. J. and Andry, A. N. Jr., "The Nature of the Temporal Solutions of Damped Distributed Systems with Classical Normal Modes," *Journal of Applied Mechanics*, Vol. 49, No. 4, pp. 867-870.

[3]Kato, T., *Perturbation Theory for Linear Operators*, Springer-Verlag, New York, 1966, p. 169.

[4]Caughey, T.K. and O'Kelly, M. E. J., "Classical Normal Modes in Damped Linear Systems," *Journal of Applied Mechanics*, Vol. 32, 1965, pp. 583-588.

Robust Nonlinear Least Squares Estimation Using the Chow-Yorke Homotopy Method

J.P. Dunyak,* J.L. Junkins,†
and L.T. Watson‡
Virginia Polytechnic Institute and State University
Blacksburg, Virginia

Introduction

HOMOTOPY and continuation methods have recently found widespread adoption as approaches for developing exceptionally robust algorithms for many multidimensional problems in nonlinear solid and fluid mechanics and optimal control.[1-8] To a lesser extent, these ideas have been applied to nonlinear estimation and system identification problems; Refs. 9 and 10 are recent examples. It is important to remark that merely adopting a homotopy or a continuation approach guarantees neither efficiency nor robustness of the resulting algorithm. A marginal increase in the domain of numerical convergence in exchange for a large decrease in computational efficiency is a dubious justification for a homotopy or continuation method in comparison to, say, a Gauss-Newton method.

In the present paper, we demonstrate a recently developed homotopy algorithm,[3] and find an order of magnitude increase in the domain of convergence in comparison to the embedded Gauss-Newton algorithm of Kirszenblat and Chetrit.[10] Since Kirszenblat and Chetrit's algorithm was originally shown to be considerably more robust than the classical Gauss-Newton algorithm, we establish that the present Chow-Yorke algorithm is robust, indeed.

The Nonlinear Least Squares Problem

We consider the problem of determining the best estimate of a parameter vector $x=(x_1,x_2\ldots x_n)^T$ which results in a given nonlinear function

$$y=f(t,x) \tag{1}$$

being a best fit of a given set of measured y-values

$$\{t_1,\tilde{y}_1;t_2,\tilde{y}_2;\ldots;t_m,\tilde{y}_m\}\,, \qquad m>n \tag{2}$$

where the m measurement times t_j are assumed perfectly known. Adopting the simple least squares penalty function, we seek the estimate of x which minimizes

$$J(x)=\sum_{j=1}^{m}[\tilde{y}_j-f(t_j,x)]^2 \tag{3}$$

The classical Gauss-Newton algorithm[11] for obtaining successive corrections $\Delta x(i)$ to a sequence of trial vectors $x(i)$ is based upon taking $x=x(i)+\Delta x(i)$ and linearizing $f(t_j,x)$ about $x(i)$; upon substituting this linear approximation for $f(t_j,x)$ into Eq. (3), the resulting quadratic (in Δx) approximation of J can be minimized with respect to Δx to obtain the normal equations

$$\Delta x(i)=(A^TA)^{-1}A^T\Delta y\,; \qquad i=0,1,2,\ldots \tag{4}$$

where

$$[A]=\begin{bmatrix} \left.\frac{\partial f(t_1,x)}{\partial x_1}\right|_{x(i)} & \cdots & \left.\frac{\partial f(t_1,x)}{\partial x_n}\right|_{x(i)} \\ \vdots & & \vdots \\ \left.\frac{\partial f(t_m,x)}{\partial x_1}\right|_{x(i)} & \cdots & \left.\frac{\partial f(t_m,x)}{\partial x_n}\right|_{x(i)} \end{bmatrix} \tag{5}$$

is the locally evaluated Jacobian matrix and

$$\Delta y=\{[\tilde{y}_1-f(t_1,x(i))],\ \ldots[\tilde{y}_m-f(t_m-f(t_m,x(i))]\}^T \tag{6}$$

is the residual vector.

The Continuation Method of Kirszenblat and Chetrit

Motivated by the desire to enlarge the domain of convergence of the Gauss-Newton algorithm [successive corrections using Eq. (4)], Kirszenblat and Chetrit[10] imbedded the Gauss-Newton algorithm into a continuation

Received July 6, 1983; revision received Aug. 29, 1983.

*Graduate Research Assistant, Department of Engineering Science and Mechanics.

†Professor of Engineering Science and Mechanics. Associate Fellow AIAA.

‡Associate Professor of Computer Science.

process. They introduced the one-parameter family of penalty functions

$$J(x,\lambda) = \lambda J(x) + (1-\lambda)\sum_{j=1}^{m} [a_j(\lambda) - f(t_j,x)]^2 \tag{7}$$

where

$$a_j(\lambda) = \lambda \bar{y}_j + (1-\lambda) f(t_j, x(0)) \tag{8}$$

Careful inspection of Eqs. (7) and (8) reveals that

$$J(x,1) \equiv J(x) \tag{9}$$

and

$$J(x,0) = 0, \quad \text{at} \quad x = x(0) \tag{10}$$

Thus, sweeping λ from zero to one defines a family of neighboring least squares problems; if a converged solution at $\lambda = 1$ can be found, then we have a minimum of the original penalty function. The algorithm of Kirszenblat and Chetrit involved introducing a sequence of λ-values and for each λ_k linearizing the local behavior of $f(t,x)$; an algorithm identical in structure to Eq. (4) is obtained (see Ref. 10 for the details). In essence, the classical Gauss-Newton algorithm has been imbedded into a process defining a one-parameter family of neighboring least squares problems. By sweeping λ at small increments, we can remain close to converged neighboring solutions which can hopefully be used to obtain good starting iterates. Thus, one can structure the algorithm to make the linearization implicit in Eq. (4) more nearly satisfied for each of a sequence of problems; as is evident below, this usually results in an increased domain of convergence.

A number of difficulties may be encountered, however, in applications of this continuation approach. The Jacobian matrix implicit in this quadratic approximation of $J(x+\Delta x,\lambda)$ from Eq. (7) is not always well-conditioned. Also, turning points and bifuration points can be encountered for some λ-values less than one. The numerical results presented below support the conclusion that our implementation[13] of the Chow-Yorke algorithm[12] is vastly superior vis-a-vis the resulting domain of convergence.

The Chow-Yorke Homotopy Algorithm

The Chow-Yorke algorithm is a scheme for solving nonlinear systems of equations based on homotopy maps of the form

$$F(x,\lambda,a) = 0 \tag{11}$$

where x, F, a are n-vectors and $\lambda \epsilon [0,1]$ is a scalar continuation or homotopy parameter. Under fairly weak assumptions on F, for almost all (fixed) vectors a, the zero set of Eq. (11) is a smooth curve γ in the $(n+1)$-dimensional (x,λ) space. The smooth curve γ can be further parametrized by the arc length s along the $(n+1)$ dimensional space curve γ; thus

$$x = x(s) \quad \text{and} \quad \lambda = \lambda(s), \qquad 0 \le \lambda \le 1 \tag{12}$$

Equation (11) then becomes

$$F(x(s),\lambda(s),a) = 0 \tag{13}$$

Considering both x and λ to be dependent functions of s has many theoretical and computational advantages. Whereas x is often not a single-valued function of λ, x is (with probability one) a single-valued function of s. We now develop a set of simultaneous differential equations whose numerical solution gives $x(s)$, $\lambda(s)$. Taking the s-derivative of Eq. (13) gives the homogeneous equation

$$[D]\begin{bmatrix} \dfrac{dx}{ds} \\ \\ \dfrac{d\lambda}{ds} \end{bmatrix} = 0 \tag{14}$$

and, since ds is the differential arc length, we have

$$\left\{\frac{dx}{ds}\right\}^T\left\{\frac{dx}{ds}\right\} + \left(\frac{d\lambda}{ds}\right)^2 = 1 \tag{15}$$

where the $n \times (n+1)$ Jacobian matrix is

$$D(x(s),\lambda(s)) = \begin{bmatrix} \dfrac{\partial F_1}{\partial x_1} & \cdots & \dfrac{\partial F_1}{\partial x_n} & \dfrac{\partial F_1}{\partial \lambda} \\ \vdots & & \vdots & \vdots \\ \dfrac{\partial F_n}{\partial x_1} & \cdots & \dfrac{\partial F_n}{\partial x_n} & \dfrac{\partial F_n}{\partial \lambda} \end{bmatrix}$$

The initial conditions are

$$x(0) = x_o, \qquad \lambda(0) = 0 \tag{16}$$

Equations (14) and (15) determine the derivative vector $(dz/ds) = (dx/ds, d\lambda/ds)^T$ only implicitly, but as is shown in Ref. 3, the solution (for D of rank n) is unique if one imposes a continuity assumption (the current derivative (dz/ds) must always make an acute angle with the previous value of this vector). In Ref. 3, a novel solution process is given which uses

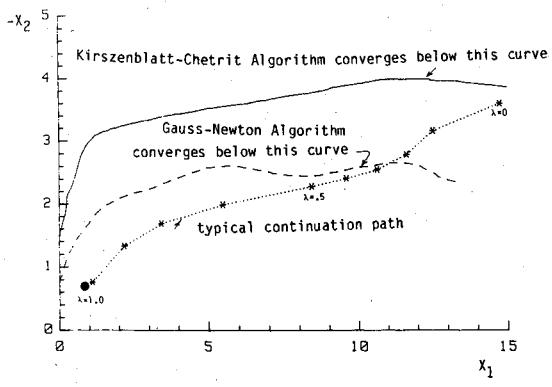

Fig. 1 Gauss-Newton and Kirszenblat-Chetrit convergence domains.

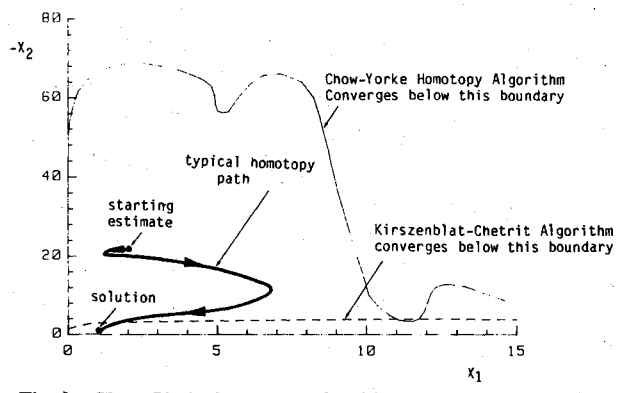

Fig. 2 Chow-Yorke homotopy algorithm convergence domain.

a robust Householder matrix reduction method[14] to calculate $dz/ds = (d\lambda/ds)^T$ on each step of a variable step, variable order Adams PECE differential equation solver. The Adams PECE differential equation solver is implemented in L.F. Shampine's subroutine STEP.[15]

For the nonlinear least squares problem, we introduce the following homotopy:

$$F(x(s),\lambda(s),a) = (1-\lambda)(x-a) + \lambda \begin{Bmatrix} \frac{\partial J}{\partial x_1}\Big|_{x(s)} \\ \vdots \\ \frac{\partial J}{\partial x_n}\Big|_{x(s)} \end{Bmatrix} = 0 \tag{17}$$

It is evident that $\lambda = 0$ has the trivial root $x(0) = a$, and a solution x (1) at $\lambda = 1$ requires that the gradient of the original residual penalty function, Eq. (3), vanish.

In the example below, we consider the domain of numerical convergence resulting from applying the three algorithms:

1) The classical Gauss-Newton algorithm.[11]

2) The continuation algorithm of Kirszenblat and Chetrit.[10]

3) The Chow-Yorke homotopy algorithm,[13] which follows the zeroes of Eq. (17) from $\lambda = 0$ to the condition $\lambda = 1$.

The example we use is the nonlinear least squares example introduced by Kirszenblat and Chetrit.[10] Other examples using the Chow-Yorke algorithm are given in Ref. 16.

Numerical Solutions of Kirszenblat and Chetrit's Problem

Following Ref. 10, we consider the function of four parameters and time:

$$y = x_1 e^{x_2 t} \cos(x_3 t + x_4) \tag{18}$$

Data was simulated by taking

$$x = [x_1 x_2 x_3 x_4]^T = [1.00 \quad -0.70 \quad 2.00 \quad 0.00] \tag{19}$$

Thirty time samples were taken, starting at $t = 0$ and using a uniform increment $\Delta t = 0.2$ s between successive measurements. Both perfectly calculated y-values and y-values corrupted by zero mean Gaussian random numbers ($\sigma = 0.02$) were used as measurements. The addition of this small random noise had negligible impact upon any of our major results; we therefore will not discuss further these noisy data solutions.

Following the pattern of Kirszenblat and Chetrit, we vary the starting iterates for the four elements of x away from their true values (which obviously minimize Eq. (3) with a minimum value of zero), to study the domain of numerical convergence of their algorithm. Since graphical presentations are difficult in a four-dimensional space, we will discuss here a subspace of starting iterates for which x_3 and x_4 are initially assigned their true values, but starting iterates for x_1 and x_2 are varied over a broad region. After the first correction, of course, all four elements of x are typically displaced from their true values until convergence is achieved. Using the subspace of x_1, x_2 over which convergence is achieved as a graphical means to study convergence is introduced in Ref. 10; we follow this pattern so that homogeneous convergence comparisons can be made.

Figure 1 displays a portion of the (x_1, x_2) starting iterates over which successful numerical convergence ensues, using i) the classical Gauss-Newton algorithm, and ii) the continuation algorithm of Kirszenblat and Chetrit. A typical Kirszenblat-Chetrit continuation path is also shown in Fig. 1. The Gauss-Newton algorithm converges (in three to six iterations) below the dashed curve, whereas Kirszenblat and Chetrit's algorithm converges below the solid line if 10 or more continuation steps are taken. Thus, Kirszenblat and Chetrit's algorithm does significantly expand the domain of convergence of the Gauss-Newton algorithm for this example. The convergence domain of Kirszenblat and Chetrit's continuation process does not increase significantly, however, even if 50 continuation steps are taken. The failure to converge is a consequence of turning points and all elements of the Jacobian matrix tending to zero for large negative values of x_2 (due to the $e^{x_2 t}$ terms). The results we computed in Fig. 1 are essentially identical to those originally reported by Kirszenblat and Chetrit.[10]

Figure 2 displays the domain of convergence and a typical homotopy path for the Chow-Yorke homotopy algorithm. Notice that the region of convergene is over an order of magnitude larger in the x_2 direction than the Kirszenblat-Chetrit region of Fig. 1. Considering arc length as the independent variable avoids the numerical difficulty at turning points if λ is chosen as the independent variable; this is an inherent limitation of the Kirszenblat-Chetrit approach.

Concluding Remarks

The Chow-Yorke homotopy algorithm was found to be much more robust than the Kirszenblat-Chetrit continuation method, when applied to the Kirszenblat-Chetrit example nonlinear least squares problem. The Chow-Yorke algorithm is broadly applicable for achieving very large numerical convergence domains. However, the homotopy methods as a group (and the Chow-Yorke algorithm in particular) are relatively expensive vis-a-vis computer run time compared to, for example, a successful solution using a Gauss-Newton algorithm. Of course, an expensive solution is vastly preferred to no solution at all. Robustness measures "stability with respect to initial ignorance"; a high degree of such stability is most important when seeking iterative solutions of problems having both high dimension and nonlinearity.

References

[1] Kamat, M.P., Watson, L.T., and Venkayya, V.B., "A Quasi-Newton Versus a Homotopy Method for Nonlinear Structural Analysis," *Computers and Structures,* in press.

[2] Kamat, M.P., Watson, L.T., and Junkins, J.L., "A Robust and Efficient Hybrid Method for Finding Multiple Equilibrium Solutions," Paper No. 83-107, *Third International Symposium on Numerical Methods in Engineering,* Paris, March 1983.

[3] Watson, L.T., "A Globally Convergent Algorithm for Computing Fixed Points of C^2 Maps," *Applied Mathematical Computations,* Vol. 5, 1979, pp. 297-311.

[4] Junkins, J.L. and Turner, J.D., "Optimal Continous Torque Attitude Maneuvers," *Journal of Guidance and Control,* Vol. 3, May 1980, pp. 210-217.

[5] Turner, J.D. and Junkins, J.L., "Optimal Large Angle Rotational Maneuvers for Flexible Spacecraft," *Journal of Guidance and Control,* Vol. 3, Nov. 1980, pp. 578-585.

[6] Turner, J.D., Chun, H.M., and Junkins, J.L., "Optimal Large-Angle Maneuvers with Vibration Suppression," *Large Space Structures Symposium,* Williamsburg, Va., May 13, 1982, NASA Conference Publication 2258.

[7] Wasserstrom, E., "Solving Boundary Value Problems by Imbedding," *Journal of Assoc. Comp. Mech.,* Vol. 18, 1971, pp. 594-602.

[8] Roberts, S.M., and Shipman, J.S., *Two-Point Boundary-Value Problems: Shooting Methods,* Elsevier, New York, 1972, p. 153.

[9] Wasserstrom, E., "Numerical Solutions by the Continuation Method," *SIAM Review,* Vol. 15, Jan. 1973.

[10] Kirszenblat, A. and Chetrit, M., "Least Squares Nonlinear Parameter Estimation by the Iterative Continuation Methods," *AIAA Journal,* Vol. 12, 1974, pp. 1751-1752.

[11] Junkins, J.L., *Optimal Estimation of Dynamical Systems,* Sijthoff-Noordhoff, Leyden, The Netherlands, 1978, p. 33.

[12]Chow, S.N., Mallet-Paret, J., and Yorke, J.A., "Finding Zeros of Maps: Homotopy Methods that are Constructive with Probability One," *Math. Com.*, Vol. 32, 1978, pp. 887-889.
[13]Watson, L.T., and Fenner, D., "Chow-Yorke Algorithm for Fixed Points or Zeros of C^2 Maps," *ACM Trans. Math. Software*, Vol. 6, 1980, pp. 252-260.
[14]Businger, P. and Golub, G.H., "Linear Least Squares Solutions by Householder Transformations" *Numer. Math.*, Vol. 7, 1965, pp. 269-276.
[15]Shampine, L.F., and Gordon, M.K., *Computer Solution of Ordinary Differential Equations: The Initial Value Problem*, W.H. Freeman, San Francisco, 1975.
[16]Watson, L.T., "Engineering Applications of the Chow-Yorke Algorithm," *Applied Mathematical Computations*, Vol. 9, 1981, pp. 111-133.

Rapid Poisson Series Evaluation

William H. Harr* and Bernard Kaufman†
U.S. Naval Research Laboratory, Washington, D.C.

Introduction

WITH the advent of machine automated algebraic manipulation programs, many problems which were formerly intractable because of the extreme amount of algebra required have now generated new interest. With many of these problems, a new complication has arisen in that it often occurs that series development in analytic or semianalytic methods becomes very large in the total number of terms required, which limits the practical application of these theories in a useful computer program. This paper outlines a method of optimizing the evaluation of large Poisson series in terms of speed, number of unique factors, and program storage.

Poisson Series Compression

A Poisson series can be represented symbolically in the form:

$$\sum_{i=1}^{n} a_i b_i c_i \tag{1}$$

where a_i are the numeric coefficients (M/N), b_i are the algebraic coefficients $(a^j b^k..)$, and c_i are the trigonometric terms $(\sin/\cos(A\alpha+B\beta..))$. If there are x unique numeric coefficients, y unique algebraic coefficients, and z unique trigonometric terms, then the maximum series length possible is $y\times z$. Two terms of the series with the same algebraic and trigonometric factors, but different numeric coefficients, are assumed to be combined by the distributive principle to form a single term. The maximum number of unique numeric coefficients (x) is then also $y\times z$.

Few series are composed of terms in which all three factors are unique to each term. Therefore, representing the series in Poisson form is duplicating information. A Poisson series of the form of Eq. (1) can be rewritten as

$$\sum_{i=1}^{n} d_i c_i, \quad \text{where} \quad d_i = a_i b_i \tag{2}$$

or as

$$\sum_{i=1}^{n} d_i b_i, \quad \text{where} \quad d_i = a_i c_i \tag{3}$$

If m is the total number of unique d_i, then this step will save $(n-m)$ multiplications. For the case $d_i = b_i c_i$, m will always equal n; that is, no work will be saved. The choice of which manner in which to combine factors will affect the final degree of optimization, but is dependent upon the particular series being optimized.

For series where each factor of each term is not unique in the entire series, the distributive principle can be used to sum all of the separate multipliers of the factor. For a series in the form of Eq. 2, in most cases each d_i is not unique to the ith term, i.e., $d_i = d_j$, for some $i \neq j$. In addition, the same is possibly true for the c_i. We can then rewrite the series in either of two forms:

$$\sum_{i=1}^{n'} e_i c_i, \quad \text{where} \quad e_i = \sum_{j=1}^{k} d_j \tag{4}$$

or as

$$\sum_{i=1}^{n'} d_i e_i, \quad \text{where} \quad e_i = \sum_{j=1}^{k} c_j \tag{5}$$

where n' is the new series length and is less than n. Assuming that the form $e_i c_i$ is used, the c_i factors are now unique to the series. This is not necessarily true for the e_i, however, so the distributive principle can be applied again. The series now have the form:

$$\sum_{i=1}^{n''} e_i f_i, \quad \text{where} \quad f_i = \sum_{j=1}^{\ell} c_j \tag{6}$$

and n'' is the series length and is less than n'.

The series are now composed of terms consisting of two factors, each factor being unique. The order in which the factors are summed (in this example, e_i c_i, and then e_i f_i, as opposed to d_i e_i and then f_i e_i) will affect the degree of series compression. Which order to use will depend upon the particular series being optimized. Normally the order of compression would be chosen to minimize the number of operations (not necessarily minimizing the series length). Figure 1 shows the possible orders of series compression. Note that the n's and n''s are not necessarily equal.

The sequence of evaluation of the series at this point is as follows: the numeric coefficients (a_i) are stored. At each point of evaluation, the algebraic and trigonometric terms $(b_i$ and $c_i)$ are calculated. The array of multiplicand pairs (d_i) are then calculated. The two series of sums $(e_i$ and $f_i)$ are

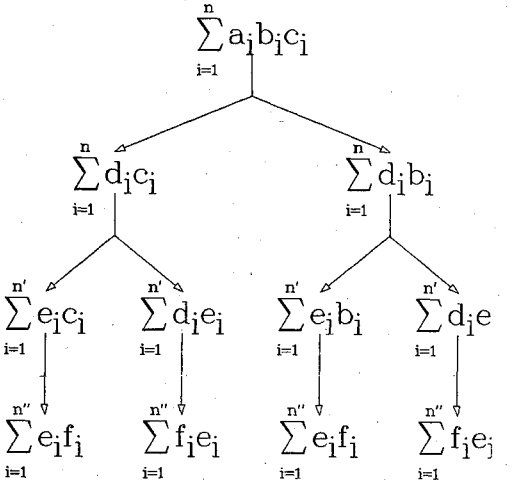

Fig. 1 Possible orders of series compression.

Presented as Paper 82-1432 at the AIAA/AAS Astrodynamics Conference, San Diego, Calif., Aug. 9-11, 1982; received Aug. 25, 1982; revision received Nov. 4, 1983. This paper is declared a work of the U.S. Government and therefore is in the public domain.

*Research Physicist, Aerospace Systems Division. Member AIAA.
†Supervisory Aerospace Engineer, Aerospace Systems Division. Associate Fellow AIAA.