JOURNAL OF COMPUTATIONAL PHYSICS 62, 248-262 (1986)

Continuous Orthonormalization for Boundary Value Problems

GUNTER H. MEYER*

School of Mathematics, Georgia Institute of Technology, Atlanta, Georgia 30332

Received July 16, 1984; revised March 1, 1985

Continuous orthonormalization describes an initial value method for linear 2-point boundary value problems which provides an orthogonal basis for the solution space at all points of the interval. In this paper the equations of continuous orthonormalization are derived with elementary projection arguments to provide geometric insight and motivate some modifications of an earlier algorithm. The method is then applied to some oscillatory and stiff boundary value problems to demonstrate that it is simple to use, problem independent, and as adaptive as the initial value code which is used to integrate the equations of continuous orthonormalization. © 1986 Academic Press, Inc

1. INTRODUCTION

A continuous orthonormalization method for 2-point boundary value problems was described by Bakhvalov [1], suggested by Drury [2] for the Orr–Sommerfeld equation, and finally implemented and tested in this setting by Davey [3]. The method shows impressive results when applied to such notoriously stiff boundary value problems as the Orr–Sommerfeld eigenvalue problem for plane Poiseuille flow at Reynolds numbers up to 10^9. The results are all the more impressive because a simple fixed-step Runge–Kutta integrator was used in [3] to solve the nonlinear initial value problems on which the new method is based.

It is the purpose of this paper to demonstrate that the continuous orthonormalization method is a robust solution method for stiff as well as oscillatory linear boundary value problems. It is easy to apply, requires no decision about orthonormalization points, uses the full adaptivity built into current initial value codes, and has moderate storage requirements. Like invariant imbedding and shooting [4], the method is particularly useful for problems with nonlinear or free boundary conditions and for eigenvalue problems where a representation of the solution on the boundary is required before the complete solution can be computed. Unlike invariant imbedding and shooting, the continuous orthonormalization method does

* This research was carried out while the author was a Visiting Scientist at the Mc Donnell Douglas Research Laboratories in St. Louis, Missouri. It was supported through the McDonnell Douglas Independent Research and Development program and NSF Grant MCS 8302548.

0021-9991/86 $3.00

not have to contend with unbounded or unstable initial value problems and is therefore easier to use. These advantages must be balanced against the necessity to integrate larger systems of differential equations than those occurring in competing initial value methods for boundary value problems.

It is also the purpose of this paper to provide some geometric insight into the method. In particular, we shall give a proof of the key spanning property of the computed orthonormal set based on elementary projection arguments similar to those of [1] rather than the less familiar wedge product used in [3], explain why the hints on implementation given in [3] are essential for success of the method, and provide a new algorithm for the reverse integration required by the method.

The power of continuous orthonormalization is demonstrated by solving three 2-point boundary value problems involving boundary layers, rapid oscillations, turning points, and Orr–Sommerfeld eigenvalue calculations. The method proved equal to all these problems without special fine-tuning.

2. Continuous Orthonormalization

Our aim is to derive the equations of continuous orthonormalization for an inner product space to motivate geometrically the original algorithm and its modifications. Consider the linear system

$$y' = A(t)\,y + r(t) \tag{2.1}$$

where $y \in C_n$, $A(t)$ is an $n \times n$ continuous complex matrix, and $r(t)$ is a continuous vector in C_n. The system (2.1) is augmented by separated boundary conditions of the form

$$B_0\,y(0) = \alpha_0, \qquad g[y(1)] = 0 \tag{2.2}$$

where B_0 is a $k \times n$ complex matrix of rank k and $g[y(1)] = 0$ represents the remaining $n-k$ boundary conditions. We shall assume that the vector y has been ordered such that the first k columns of B_0 are linearly independent. If

$$B_0 = (B_1\ B_2)$$

where B_1 is a nonsingular $k \times k$ matrix, and the $n \times n$ matrix C is defined as

$$C = \begin{pmatrix} B_1^{-1} & -B_1^{-1}B_2 \\ 0 & I \end{pmatrix},$$

then for $z(t) = C^{-1}y(t)$ we obtain the boundary value problem

$$z' = C^{-1}A(t)\,Cz,$$

$$z(0)_i = \alpha_i, \qquad i = 1,\dots, k,$$

$$g[Cz(1)] = 0.$$

Hence, without loss of generality we shall assume that B_0 is of the form $(I\,0)$ where I is the $k \times k$ identity. If both boundary conditions are linear, then the end point should be chosen as the starting point for the orthonormalization method for which k is larger since initial value problems for $n(n-k)$ equations will have to be integrated.

As is common in the method of superposition, the solution of (2.1, 2) may be expressed as

$$y(t) = y_p(t) + U(t)c \tag{2.3}$$

where y_p is any solution of the initial value problem

$$\begin{aligned} y' &= A(t)\,y + r(t), \\ B_0\,y(0) &= \alpha \end{aligned}$$

and U is an $n \times (n-k)$ matrix with linearly independent columns which satisfies

$$\begin{aligned} U' &= A(t)\,U, \\ B_0\,U(0) &= 0. \end{aligned} \tag{2.4}$$

The columns of U span a $n-k \equiv p$-dimensional subspace $S(t)$ of C_n. The unknown vector c is determined by substituting the representation (2.3) into $g[y(1)] = 0$ and solving for c (if possible).

In nonstiff boundary value problems, linearly independent initial conditions for $U(t)$ retain numerical linear independence on integration which, as a rule, helps in determining c. In stiff boundary value problems, even orthogonal initial conditions can lose numerical linear independence so that $g[y(1)] = 0$ may become ill-conditioned or unsolvable. The discrete orthonormalization method is designed to provide almost orthogonal columns for U by periodically redefining initial conditions for U through a Gram–Schmidt process (see, e.g., [5]). The continuous method, on the other hand, maintains orthogonality at all times. This method may be motivated as follows:

On the complex vector space C_n, a convenient inner product denoted by $\langle\ ,\ \rangle$ is imposed. (In the sample calculations the standard l_2 inner product is used; however, the structure of (2.1) or the boundary conditions may dictate a better choice.)

Let $\{u_i(0)\}$ denote p orthonormal initial conditions for the homogeneous problem (2.4) with respect to this inner product. Then the solution $\{u_i(t)\}$ remains linearly independent and spans the solution space $S(t)$ of (2.4).

Let $\{y_i(t)\}$ be p orthonormal vectors which also span $S(t)$ and which coincide with $\{u_i(0)\}$ at $t=0$. Then $u_i(t)$ can be written as

$$u_i(t) = \sum_{j=1}^{p} \langle u_i(t), y_j(t)\rangle\, y_j(t). \tag{2.5}$$

Since the y_i are assumed to be orthonormal, we require that

$$\langle y_i, y_j \rangle = \delta_{ij} \qquad \text{and} \qquad \langle y_i, y_j \rangle' = 0.$$

Each u_i satisfies (2.4) which may be expanded as

$$u_i' = \sum_{j=1}^{p} \langle u_i' y_j \rangle y_j + \langle u_i, y_j' \rangle y_j + \langle u_i, y_j \rangle y_j' = \sum_{j=1}^{p} \langle u_i, y_j \rangle A y_j.$$

Substitution of (2.5) for u_i leads to

$$\sum_{j=1}^{p} \langle u_i, y_j \rangle y_j' = \sum_{j=1}^{p} \langle u_i, y_j \rangle A y_j - \sum_{j=1}^{p} \langle u_i, y_j \rangle \sum_{k=1}^{p} \langle A y_j, y_k \rangle y_k$$
$$- \sum_{j=1}^{p} \langle u_i, y_j \rangle \sum_{k=1}^{p} \langle y_j, y_k' \rangle y_k.$$

As $S(t)$ changes with t, it is possible to update it continuously rather than recompute it completely [3]; i.e.,

$$S(t + \Delta t) = S(t) \oplus \Delta S(\Delta t);$$

$\Delta S(\Delta t)$ denotes the orthogonal complement of $S(t)$ in $S(t + \Delta t)$ so that $S(t) \perp \Delta S$. Then any vector $y_i(t + \Delta t)$ in $S(t + \Delta t)$ can be decomposed as

$$y_i(t + \Delta t) = y_i(t) + \Delta y_i$$

where $\Delta y_i \perp S(t)$. Differentiability of $y_i(t)$ now implies that

$$\langle y_i'(t), y_j(t) \rangle = 0 \qquad 1 \leqslant i, j \leqslant p. \tag{2.6}$$

If we impose condition (2.6) rather than $\langle y_i, y_j \rangle' = 0$, then the above equation is satisfied if the y_j are chosen such that

$$y_j' = A y_j + \sum_{k=1}^{p} g_{jk} y_k, \tag{2.7}$$

$$y_j(0) = u_j(0)$$

with

$$g_{jk} = -\langle A y_j, y_k \rangle. \tag{2.8}$$

Alternatively, we can make the Ansatz (2.7) and compute g_{jk} such that $\langle y_l, y_j' \rangle = 0$ which leads to the linear system

$$\langle A y_j, y_l \rangle + \sum_{k=1}^{p} g_{jk} \langle y_k, y_l \rangle = 0 \qquad 1 \leqslant l \leqslant p \tag{2.9}$$

for the unknown g_{jk}. The Eqs. (2.7), (2.9) describe the continuous orthonormalization method. We note that if the l_2 inner product is chosen then the Eqs. (2.7)–(2.9) can be rewritten compactly in matrix form as

$$y' = (I - YY^+)AY$$

where Y^+ denotes the generalized inverse $(Y^*Y)^{-1}Y^*$ and Y^* the conjugate transpose of the complex $n \times p$ matrix Y.

We summarize and make precise the above heuristic derivation in the following key theorem which was proved in [3] with wedge products.

THEOREM 1. *Let $\{u_i(t)\}$ be p linearly independent solutions of* (2.4) *which are initially orthonormal. Then the nonlinear pn-dimensional complex system*

$$y_i' = Ay_i + \sum_{j=1}^{p} g_{ij} y_j \qquad i = 1,..., p, \tag{2.10}$$
$$y_i(0) = u_i(0)$$

with $\{g_{ij}\}$ computed from (2.9) *has a unique solution $\{y_i(t)\}_{i=1}^{p}$. The set $\{y_i(t)\}$ is orthonormal and*

$$\operatorname{span}\{y_i(t)\} = \operatorname{span}\{u_i(t)\}.$$

Proof. The pn-dimensional system (2.10) has a polynomial right side and hence has a unique local solution. Moreover, it follows from the defining relations for g_{ij} that $\langle y_i', y_j \rangle = 0$ for all i and j so that $\langle y_i, y_j \rangle' = 0$. Hence the set $\{y_i(t)\}$ remains orthonormal and exists globally. Consider now the residual

$$r_i = u_i - \sum_{j=1}^{p} \langle u_i, y_j \rangle y_j.$$

Differentiation shows that

$$r_i' = u_i' - \sum_{j=1}^{p} \langle u_i', y_j \rangle y_j - \sum_{j=1}^{p} \langle u_i, y_j' \rangle y_j - \sum_{j=1}^{p} \langle u_i, y_j \rangle y_j'.$$

We note from the definition of r_i and the orthonormal property of $\{y_j\}$ that

$$\langle u_i, y_j' \rangle = \langle r_i, y_j' \rangle$$

and

$$\langle Au_i, y_j \rangle = \langle Ar_i, y_j \rangle + \sum_{k=1}^{p} \langle u_i, y_k \rangle \langle Ay_k, y_j \rangle.$$

Substitution into the expression for r_i leads to

$$r_i' = Ar_i - \sum_{j=1}^{p} \{\langle Ar_i, y_j\rangle - \langle r_i, y_j'\rangle\} y_j$$
$$- \sum_{j,k=1}^{p} \{\langle u_i, y_k\rangle \langle Ay_k, y_j\rangle y_j + g_{jk}\langle u_i, y_j\rangle y_k\}.$$

Since $\langle Ay_k, y_j\rangle = -g_{kj}$, the last two sums cancel. Thus

$$r_i' = Ar_i - \sum_{j=1}^{p} \{\langle Ar_i, y_j\rangle - \langle r_i, y_j'\rangle\} y_j,$$
$$r_i(0) = 0.$$

This linear homogeneous differential equation in the components of r_i has only the zero solution so that $u_i \in \operatorname{span}\{y_j\}$ for $i = 1,..., p$.

In solving the Orr–Sommerfeld eigenvalue problem, Davey [3] found that "the temptation of substituting $-\langle Ay_j, y_k\rangle$ for g_{jk} is to be sorely resisted" because the algorithm proved unstable. Instead, g_{jk} is to be computed from the system (2.9) without imposing a priori any orthogonality. This instability is, in fact, mathematical rather than numerical, because not every orthonormal solution to (2.7), (2.8) is asymptotically stable. For ease of argument, let A be a constant Hermitian matrix, and $\{\hat{y}_k\}_{k=1}^{p}$ be an orthonormal set of eigenvectors with associated eigenvalues $\{\lambda_k\}$. Then the unique solution of (2.7),(2.8) with initial condition $y_i(0) = \hat{y}_i$ can be written as $y_i(t) = \phi_i(t)\,\hat{y}_i$, where

$$\phi_i' = \lambda_i\phi_i - \lambda_i\phi_i^3, \tag{2.11}$$
$$\phi_i(0) = 1.$$

Clearly, $\phi_i(t) \equiv 1$ is the unique solution of (2.11), but it is not always asymptotically stable. For example, if $\lambda_i < 0$ and $\phi_i(0) < 1$ then $\lim_{t\to\infty} \phi_i(t) = 0$, while for $\phi_i(0) > 1$, $\lim_{t\to\infty} \phi_i(t) = \infty$. If $\lambda_i > 0$ then $\phi_i(t) \equiv 1$ is asymptotically stable.

If g_{ik} is computed from (2.9) then

$$\lambda_i\phi_i\phi_k\delta_{ik} + \sum_{j=1}^{p} g_{ij}\phi_j\phi_k\delta_{jk} = 0$$

yields $g_{ij} = -\lambda_i\delta_{ij}$. The equation corresponding to (2.11) is now

$$\phi_i' = \lambda_i\phi_i - \lambda_i\phi_i \equiv 0,$$
$$\phi_i(0) = 1$$

whose solution is asymptotically stable. A general stability theorem for (2.7), (2.9) follows from the observation that $Y^*(t)\,Y(t) = Y^*(0)\,Y(0)$ provided only that

$Y^*(0)\,Y(0)$ is invertible [1]. In particular since $Y^*(0)\,Y(0)=I$ it follows that $Y^*(t)\,Y(t)$ numerically will remain close to the identity matrix at all times. Hence a reasonable simplification of the method can be based on the splitting $Y^*Y=D(I-B)$ and the approximation

$$(Y^*Y)^{-1}=\left(\sum_{k=0}^{m} B^k\right)D^{-1}$$

where DB contains all off-diagonal elements of Y^*Y. Numerical results indicate that $m=0$ already provides a stable method.

Let us next consider the calculation of the particular integral $y_p(t)$. Again for computational stability, it is advantageous to find a particular integral which is orthogonal to the subspace $\operatorname{span}\{y_i(t)\}$. Let $u_p(t)$ be any particular integral of (2.1) which also satisfies the boundary condition

$$B_0\,y_p(0)=\alpha_0.$$

In view of Theorem 1 we can write

$$y_p(t)=\sum_{i=1}^{p}\langle y_p(t), y_i(t)\rangle\, y_i(t)+z_p(t) \tag{2.12}$$

where $z_p(t)$ belongs to the orthogonal complement of $\operatorname{span}\{y_i(t)\}$. Differentiation shows that

$$z_p'(t)=y_p'(t)-\sum_{i=1}^{p}\{\langle y_p', y_i\rangle\, y_i+\langle y_p, y_i'\rangle\, y_i+\langle y_p, y_i\rangle\, y_i'\}.$$

If the differential equations are substituted for y_p' and y_j' and y_p is replaced with the expression (2.12), then it follows immediately that

$$z_p'=Az_p+r+\sum_{i=1}^{p} g_{pi}\,y_i \tag{2.13}$$

$$B_0 z_p(0)=\alpha_0$$

where $\{g_{pi}\}$ are computed by using orthogonality and the defining relations (2.8) and (2.12). Alternatively, we can use the representation (2.13) and the requirement that $\langle z_p, y_j\rangle'\equiv 0$. We find the linear system

$$\langle Az_p, y_k\rangle+\langle r, y_k\rangle+\sum_{j=1}^{p} g_{pj}\langle y_j, y_k\rangle$$

$$+\langle z_p, Ay_k\rangle+\left\langle z_p, \sum_{j=1}^{p} g_{kj}\,y_j\right\rangle=0 \qquad k=1,\ldots, p.$$

The relation for g_{pk} suggested in [3] requires only that $\langle z'_p, y_j \rangle = 0$ which does not assure orthogonality of z_p to span$\{y_i\}$ but does provide a minimum norm particular solution [1]. We note again that for the l_2 inner product Eq. (2.13) can be written in matrix form as

$$z'_p = (I - YY^+)(Az_p + y) - Y(Y^*Y)^{-1} Y'^* z.$$

Let us summarize the algorithm. For the inhomogeneous boundary value problem (2.1), (2.2) we integrate simultaneously the nonlinear $n(p+1)$-dimensional system

$$y'_j = Ay_j + \sum_{k=1}^{p} g_{jk} y_k, \tag{2.14}$$

$$z'_p = Az_p + r + \sum_{k=1}^{p} g_{pk} y_k$$

where $\{y_j(0)\}$ is an orthonormal set satisfying

$$B_0 y_j(0) = 0, \qquad j = 1, \ldots, p,$$

and $z_p(0)$ is any vector in the orthonormal complement of span$\{y_j(0)\}$ which satisfies $B_0 z_p(0) = \alpha_0$. Then by Theorem 1 the solution of (2.1), (2.2) at any point $t \in [0, 1]$ can be expressed as

$$y(t) = z_p(t) + \sum_{i=1}^{p} \lambda_i(t)\, y_i(t). \tag{2.15}$$

The integration of (2.14) is referred to as the forward sweep. The complete solution $y(t)$ is found during the reverse sweep to be discussed next.

As before we assume that the boundary condition

$$g\left(z_p(1) + \sum_{i=1}^{p} \lambda_i(1)\, y_i(1)\right) = 0 \tag{2.16}$$

has at least one solution $\{\lambda_i(1)\}$. This solution determines an initial condition for the solution y of (2.1), (2.2) at $t = 1$. However, for stiff problems it is not advisable to integrate an initial value problem for (2.1) over the whole interval. Instead, it is suggested in [3] to solve the differential equation satisfied by $\{\lambda_i(t)\}$ which on differentiation is obtained as

$$\lambda'_i = -g_{pi} - \sum_{j=1}^{p} g_{ji} \lambda_j \tag{2.17}$$

with $\lambda_i(1)$ determined from (2.16).

The coefficients g_{ij} depend on the orthonormal set $\{y_i(t)\}$. To avoid storage and interpolation, Davey [3] suggests simultaneous integration of (2.7), (2.9), and (2.17) with periodic adjustment of the initial conditions for (2.7) with the results from the forward sweep to improve stability. The calculation of the Orr–Sommerfeld eigenfunction for a Reynolds number of $R = 10^9$ proves the soundness of this approach.

A somewhat different method is suggested here; it requires only the integration of the n original differential equations (2.1) rather than the $(n+1)p$ equations (2.7), (2.17). Moreover, no interpolation is necessary, but the demand on storage increases.

We save the orthogonal set $\{y_i(t), z_p(t)\}$ during the forward sweep at an a priori chosen number of mesh points $\{t_k\}_{k=0}^{M}$ with $t_0 = 0$ and $t_M = 1$. Let v_k denote the solution of (2.1) over the interval $[t_{k-1}, t_k]$. It is obtained over successive subintervals from

$$\begin{aligned} v_M(1) &= y(1), \\ v_k'(t) &= A(t)v_k + r(t),\ t \in [t_{k-1}, t_k], \\ v_k(t_k) &= \sum_{i=1}^{p} \langle v_{k+1}(t_k) - z(t_k), y_i(t_k)\rangle\, y_i(t_k) + z(t_k). \end{aligned} \tag{2.18}$$

In other words, the computed solution of the initial value problem is periodically projected into the subspace where it is supposed to be. These interior points serve the same role as the interior points during multiple shooting, but unlike in the shooting method they do not contribute to the complexity of the algorithm since only the simple projection (2.18) is required. The examples of the following section illustrate the efficiency and stability of the method.

3. Numerical Examples

In the following sample problems all ordinary differential equations have been integrated with the IMSL Runge–Kutta routine DVERK with a default error tolerance of 10^{-6}. Unless otherwise noted, the projection points $\{t_k\}_{k=0}^{M}$ are evenly distributed over the interval of integration.

Example 1. The simple 2-point boundary value problem

$$\begin{gathered} u'' + \gamma u = 10\,000 \cos 80t, \\ u(0) = \alpha, \qquad u(1) = \beta \end{gathered} \tag{3.1}$$

is chosen to examine the behavior of the method for large $|\gamma|$. Continuous orthonormalization is applied to the equivalent first-order system

$$v' = Av + r$$
$$v = \begin{pmatrix} u \\ u' \end{pmatrix}, \qquad A = \begin{pmatrix} 0 & 1 \\ -\gamma & 0 \end{pmatrix}, \qquad r = \begin{pmatrix} 0 \\ 10\,000 \cos 80t \end{pmatrix}. \tag{3.2}$$

The solution v is expressed as

$$v(t) = \lambda(t)\, y(t) + z_p(t)$$

where

$$y'(t) = Ay + g_{11}\, y,$$
$$y(0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad g_{11} = -\langle Ay, y\rangle / \langle y, y\rangle$$

and

$$z_p'(t) = Az_p + r + g_{p1}\, y,$$
$$z_p(0) = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}, \qquad g_{p1} = -[\langle Az_p, y\rangle + \langle z_p, Ay\rangle + \langle r, y\rangle$$
$$+ g_{11}\langle z_p, y\rangle] / \langle y, y\rangle.$$

The boundary condition at $t = 1$ requires that $\lambda(1)$ be determined such that $[\lambda(1)\, y(1) + z_p(1)]_1 = \beta$. Then (3.2) is integrated backward subject to

$$v(1) = \begin{pmatrix} \beta \\ [\lambda(1)\, y(1) + z_p(1)]_2 \end{pmatrix}.$$

The computed solution is projected at the mesh points according to (2.18).

Case (1) $\gamma = 10\,000$. The solution to the problem is

$$u = c_1 \cos 100t + c_2 \sin 100t + y_p(t)$$

where

$$y_p(t) = \frac{10\,000}{\gamma - 6400} \cos 80t$$

and where c_1 and c_2 are determined from the assumed boundary conditions $u(0) = 1$ and $u(1) = 0$. This case is not a difficult problem for superposition since the fundamental solutions remain well behaved over the interval, and γ is not near an eigenvalue of the differential operator. Continuous orthonormalization also reproduces the analytic solutions well. With 100 mesh points the absolute error is observed to be less than 10^{-2}. It improves somewhat when the reverse integration

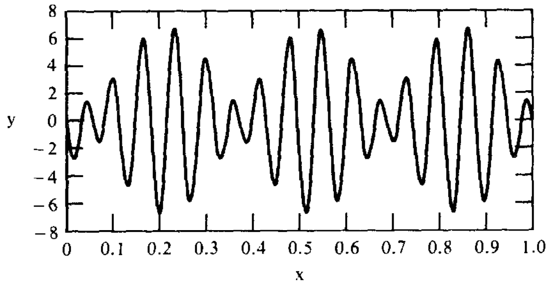

FIG. 1. Oscillatory solution of problem (3.1): $\gamma = 10^4$, $M = 200$, observed maximum absolute error: 0.012; Cyber 176 execution time: 0.77 s.

is carried out without projection, which is an indication of the stability of the initial value problem for (3.1), and also a hint that increasing the number of projection points need not always improve the solution. In fact, for $M = 1000$ the maximum error increased to 0.024. A plot of the computed and analytic solution is shown in Fig. 1. In highly oscillatory problems the vectors y and z_p rotate rapidly and are difficult to maintain at right angles with DVERK. Nonetheless, continuous orthonormalization solves the problem effectively.

If the far boundary condition had been given as $u(s) = f(s)$, $u'(s) = g(s)$ at an unknown location $t = s$, then on integrating y and z_p and by eliminating $\lambda(t)$ from the representation $v(t) = z_p(t) + \lambda(t)\, y(t)$, one can evaluate the expression

$$\phi(t) \equiv y(t)_1 z_p(t)_2 + (f(t) - z_p(t)_1)\, y(t)_2 - g(t)\, y(t)_1.$$

Any root s of $\phi(t) = 0$ defines a permissible free boundary with corresponding initial data $u(s) = f(s)$, $u'(s) = g(s)$. Thus continuous orthonormalization can be used like invariant imbedding for free boundary problems; however, it is easier to apply here since the standard invariant imbedding method is based on the Riccati transformation $u(t) = R(t)\, u'(t) + w(t)$, where $R(t) = \frac{1}{100} \tan 100t$ is periodically unbounded.

Case (2) $y = -10\,000$. The solution to the problem is

$$u = c_1 e^{100t} + c_2 e^{-100t} + y_p(t)$$

where $y_p(t)$ is the particular integral given above. The boundary conditions are again chosen as $u(0) = 1$, $u(1) = 0$. Continuous orthonormalization performs reliably. For $M = 100$ the absolute error is $O(10^{-6})$ at the mesh points which decreases to $O(10^{-9})$ for $M = 1000$. A plot of the computed solution is shown in Fig. 2. Reverse integration without projection gives a final value of $u(0) = 1.121 \cdot 10^{35}$. Hence superposition and shooting fail, while the Riccati equation in invariant imbedding has the solution $R(t) = \frac{1}{100} \tanh 100t$ which changes rapidly near 0. Orthonormalization, on the other hand, produces a nearly constant vector $y(t)$ in spite of the boundary layer near $t = 0$.

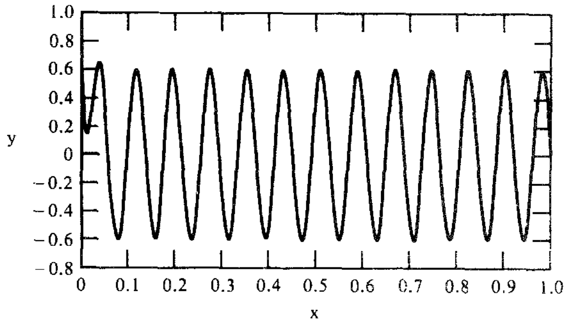

FIG. 2. Boundary layer solution of problem (3.1): $\gamma = -10^4$, $M = 200$: observed maximum absolute error: $9.7 \cdot 10^{-7}$; Cyber 176 execution time: 0.27 s.

EXAMPLE 2. The problem

$$\varepsilon u'' + tu' = -\varepsilon\pi^2 \cos(\pi t) - (\pi t) \sin(\pi t), \qquad (3.3)$$

$$u(-1) = -2, \qquad u(1) = 0$$

is used in [6] to illustrate the performance of the adaptive collocation code COLSYS for 2-point boundary value problems. The solution of the problem is $u(t) = \cos(\pi t) + \operatorname{erf}(t/\sqrt{2\varepsilon})/\operatorname{erf}(1\sqrt{2\varepsilon})$ which shows a turning point and sharp transition layer near $t = 0$. COLSYS solves this problem for $\varepsilon = 10^{-6}$ within an absolute error of $O(10^{-6})$. Initial value, finite element, and finite difference codes are expected to fail for this problem [6]. Continuous orthonormalization as implemented in the research code found no difficulty in rapidly computing the representation

$$\begin{pmatrix} u(1) \\ u'(1) \end{pmatrix} = z_p(1) + \lambda(1)\, y(1). \qquad (3.4)$$

However, the reverse integration with DVERK is stable only if $\Delta t = O(\varepsilon^{-1})$ near $t = 1$. This reverse integration would be expensive and was not attempted. The

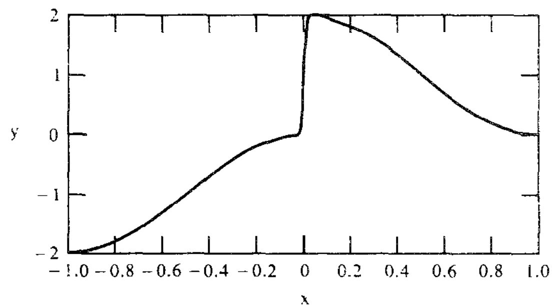

FIG. 3. Solution of problem (3.3): $\varepsilon = 10^{-4}$, $M = 2000$: observed maximum absolute error: $2.7 \cdot 10^{-4}$; CDC Cyber 176 execution time: 2.7 s.

results shown in Fig. 3 hold for $\varepsilon = 10^{-4}$ and $M = 2000$. The maximum absolute error was $3 \cdot 10^{-4}$ which decreased to 3.10^{-6} for $M = 5000$. The same number of mesh points uniformly distributed over the transformed interval $[y_0, y_1]$, where $y = (x+1)^4$ allowed the solution of (3.3) for $\varepsilon = 10^{-5}$ with an accuracy of $4 \cdot 10^{-4}$. However, such a transformation is problem specific and runs counter to the attempt to use continuous orthonormalization without special fine-tuning.

It appears to be straightforward to combine the theoretical error bounds for the Runge–Kutta method with the measured projection difference at the mesh point to give an a posteriori error bound for the method of continuous orthonormalization. However, such a bound may be very pessimistic. For example, when a solution of (3.3) is attempted with $M = 5000$ and $\varepsilon = 10^{-5}$ then the reverse integration produces errors of $O(10^5)$ on $(0, 1)$. Nonetheless, as the integration proceeds the answer suddenly jumps to the correct value and maintains an absolute error of $O(10^{-6})$ on $(-1, 0)$. Thus, integration and projection errors need not accumulate but may in fact cancel if the system has the proper structure.

A final comment on Example 2. If the boundary condition $u(1) = 0$ is replaced by a nonlinear condition $g(u(1), u'(1)) = 0$, then the representation (3.4) reduces to a single scalar equation in the unknown $\lambda(1)$. Thus, continuous orthonormalization may be used as a preprocessor for more sophisticated but time consuming codes such as COLSYS whenever a nonlinearity is introduced through the boundary data only.

EXAMPLE 3. As a final illustration, let us briefly discuss the Orr–Sommerfeld eigenvalue calculation carried out by Davey for plane Poiseuille flow with high Reynolds numbers [3]. The problem may be written as

$$y' = A(x)\, y \tag{3.5}$$

where

$$A(x) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \alpha^2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a & 0 & b & 0 \end{pmatrix},$$

$a = 2i\alpha R$, $b = \alpha^2 + i\alpha R(1 - x^2 - c)$, and c is the unknown complex eigenvalue. For the most unstable flow the boundary conditions

$$y(0)_2 = y(0)_4 = 0$$
$$y(1)_1 = y(1)_2 = 0$$

apply. Highly accurate eigenvalues for a Reynolds number of $R = 10^4$ have been published [7] which serve as a starting point for a continuation in R.

The implementation of continuous orthonormalization assembled the eighth-order system

$$y_i' = Ay_i + \sum_{j=1}^{2} g_{ij} y_i,$$

$$y_1(1) = \vec{e}_3, \qquad y_2(1) = \vec{e}_4$$

in complex form and used DVERK to integrate the real and imaginary parts. A solution of the eigenvalue problem results if c is found such that the two equations

$$[\lambda_1 y_1(0) + \lambda_2 y_2(0)]_2 = 0$$

$$[\lambda_1 y_1(0) + \lambda_2 y_2(0)]_4 = 0$$

allow a nontrivial solution (λ_1, λ_2). When the determinant of the coefficient matrix is denoted by $F(c)$, then the nonlinear complex equation

$$F(c) = 0$$

must be solved. Davey employed Newton's method whereas here a discrete Newton's method is used. For a Reynolds number of $R = 10^6$ and a wave number of $\alpha = 1$ our implementation of the eigenvalue iteration and the calculation of the eigenfunction by integration of (3.5) with periodic projection at 1000 points exactly reproduces the tabulated answer of [3]. To give an impression of the severity of his problem we have shown in Fig. 4 the imaginary part of $y(x)_4$ near the boundary $x = 1$. The other components of $y(x)$ are better behaved. It also was noted that the

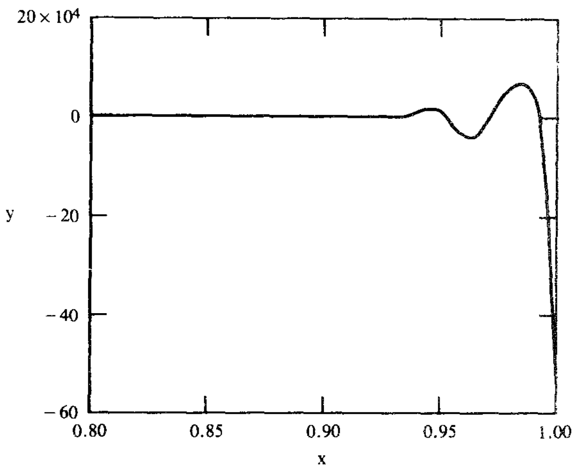

FIG. 4. Imaginary part of the component $y(x)_4$ for problem (3.5): Reynolds number: $R = 10^6$; Wave number: $\alpha = 1$. Initial guess for the discrete Newton method: $c = 0.066 - i0.013$. Convergence in nine iterations to $c = 0.066592523 - i\,0.013983266$, $M = 2000$. Cyber 176 execution time: 120 s.

significant digits of the eigenvalue and eigenvector remained unaffected when in the linear system (2.9) the off-diagonal terms $\langle y_1, y_2 \rangle$ were set to zero as suggested by the above stability analysis. No attempt was made here to compute with higher Reynolds numbers (see [3] for an eigenvalue and eigenvector calculation for $R = 10^9$).

All the above examples indicate that continuous orthonormalization merits attention as a simple robust solver for n-dimensional first-order linear systems, provided the number p of missing boundary conditions at the distinguished end point is sufficiently small that the resulting $n(p+1)$-dimensional initial value problem can be integrated economically. The use of a Runge–Kutta routine for continuous orthonormalization appears sensible since the initial value problem for the forward sweep is not stiff, while frequent projections efficiently overcome stiffness during the reverse sweep.

For both sweeps inner products need to be computed repeatedly. An effective computer code can use basic linear algebra subroutines to assmble the right side of (2.7) and (2.18). Partial vectorization of the code is thus possible without much effort.

Finally, the use of continuous orthonormalization for multipoint problems and its incorporation into a Newton iteration for nonlinear boundary value problems remain to be studied.

References

1. N. S. Bakhvalov, "Numerical Methods," MIR, Moscow, 1977. [Russian ed., 1973]
2. L. O'C. Drury, Numerical solution of Orr–Sommerfeld-type equations, *J. Comput. Phys.* **37** (1980), 133–139.
3. A. Davey, An automatic orthonormalization method for solving stiff boundary-value problems, *J. Comput. Phys.* **51** (1983), 343–356.
4. G. H. Meyer, "Initial Value Methods for Boundary Value Problems," Academic Press, New York, 1973.
5. M. R. Scott and H. A. Watts, Superposition, orthonormalization, quasilinearization and two-point boundary value problems, *in* "Codes for Boundary-Value Problems in Ordinary Differential Equations" (B. Childs *et al.*, Eds.), Lecture Notes in Computer Science Vol. 76, Springer-Verlag, Berlin, 1979.
6. U. Ascher, J. Christiansen, and R. D. Russell, COLSYS—A Collocation Code for Boundary-Value Problems, *in* "Codes for Boundary-Value Problems in Ordinary Differential Equations" (B. Childs *et al.*, Eds.), Lecture Notes in Computer Science Vol. 76, Springer-Verlag, Berlin, 1979.
7. S. A. Orszag, Accurate solution of the Orr–Sommerfeld stability equation, *Fluid Mech.* **50** (1971), 689–703.