JOURNAL OF COMPUTATIONAL PHYSICS 51, 343–356 (1983)

An Automatic Orthonormalization Method for Solving Stiff Boundary-Value Problems

A. DAVEY

School of Mathematics, University of Newcastle upon Tyne, England

Received July 30, 1982

A new initial-value method is described, based on a remark by Drury, for solving stiff linear differential two-point eigenvalue and boundary-value problems. The method is extremely reliable, it is especially suitable for high-order differential systems, and it is capable of accommodating realms of stiffness which other methods cannot reach. The key idea behind the method is to decompose the stiff differential operator into *two non-stiff* operators, one of which is nonlinear. The nonlinear one is specially chosen so that it advances an orthonormal frame, indeed the method is essentially a kind of *automatic* orthonormalization; the second is auxiliary but it is needed to determine the required function. The usefulness of the method is demonstrated by calculating some *eigenfunctions* for an Orr–Sommerfeld problem when the Reynolds number is as large as 10^9.

1. INTRODUCTION

Although linear differential problems are usually easier to solve *analytically* than nonlinear problems, a somewhat curious feature of trying to obtain a *numerical* solution on a computer, which will necessarily have a *finite* word-length, is that it is often more difficult to solve linear problems. The reason for this is that basic solutions of linear differential systems are usually exponential in character and if some solutions have widely differing exponents they will grow or decay very rapidly relative to each other as the independent variable changes. Thus, because of the finite word-length, it is often impossible to form sensible numerical combinations of the basic solutions. Such problems are said to be *stiff*.

Throughout this paper we shall be concerned with the numerical solution of stiff linear two-point boundary-value problems for systems of ordinary differential equations by the use of explicit shooting methods. The hallmark of a stiff boundary-value problem is when the number of basic solutions which dominate as the integration proceeds is less than the number of unknown initial conditions. It is well understood that if one wishes to solve a stiff boundary-value problem by using an initial-value shooting method then one has to resort to a special technique such as orthonormalization [5, 1], or the Riccati method [7], or the compound matrix method [4, 6].

How nice it would be, however, if one could simply use the standard (superposition) shooting method to solve a stiff problem. Or, alternatively, how nice it

0021-9991/83 $3.00

would be if one could start with a set of orthonormal solutions, satisfying the known initial conditions, and integrate away so that these solutions remain orthonormal. Of course, we know that we cannot do this if we integrate the given differential system, but maybe we can integrate instead a slightly different differential system whose solutions remain orthonormal and *span the required subspace*, that is, the same subspace as spanned by the solutions of the original differential system. This can indeed be done and the above idea is due to Drury [3]; it should be noted, however, that although the mathematical formulation of the idea as expressed in his paper is formally correct, he did not use his idea to publish any numerical calculations. This is a dangerous situation in the field of numerical analysis because the usefulness of a method is dominated by its stability characteristics.

In Section 2 of this paper we explain how to develop Drury's idea to solve stiff linear homogeneous problems and in Section 3 we indicate how to solve inhomogeneous problems. Then, in Section 4, we explain in detail how to use the method to obtain eigenfunctions of the Orr–Sommerfeld equation for plane Poiseuille flow when the Reynolds number is very large. We also discuss briefly some comparisons between using the new method and using orthonormalization.

2. Description of the Method for the Homogeneous Problem

The general linear homogeneous problem of order n may be written as

$$\mathbf{y}' = \mathbf{A}\mathbf{y}, \tag{1}$$

where $\mathbf{y}$ is a complex n-vector, $\mathbf{A}$ is a complex $n \times n$ matrix of coefficients and a' denotes differentiation with respect to the independent variable x. Since (1) is linear and homogeneous it is an eigenvalue problem, at least one of the components of $\mathbf{A}$ will be a function of the unknown eigenvalue c, say. We suppose that the homogeneous boundary conditions are separated so that q boundary conditions are known at one end of the range of integration and p at the other end; $p + q = n$. For convenience of discussion we also suppose that the independent variable x may be defined so that the range of integration is $0 \leqslant x \leqslant 1$ with $q \geqslant p$ boundary conditions given at $x = 0$, the integration being from $x = 0$ to $x = 1$. Thus, there will be p *unknown* initial conditions and so all the solutions of (1) which satisfy the *known* initial conditions will lie in a subspace V of dimension p. Without any loss of generality we may assume that $\mathbf{y}$ has already been suitably redefined by a non-singular transformation so that the known initial conditions on $\mathbf{y}$ at $x = 0$ simply become that the first q components of $\mathbf{y}$ are zero. We do not need to mention until later the other p boundary conditions on $\mathbf{y}$ at $x = 1$.

Let $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_p$ be a set of p vectors which are orthonormal at $x = 0$ and such that each of them satisfy the known initial conditions, for example, we may choose $\mathbf{y}_i$ $(1 \leqslant i \leqslant p)$, so that at $x = 0$ every component of $\mathbf{y}_i$ is zero except the $(q + i)$th component which we may choose to be one. Given an estimate for the unknown eigenvalue c now calculate the $\mathbf{y}_i$ by integrating simultaneously from $x = 0$ to $x = 1$, *not* the stiff equations $\mathbf{y}_i' = \mathbf{A}\mathbf{y}_i$, but instead the non-stiff equations

$$\mathbf{y}_i' = \mathbf{A}\mathbf{y}_i + g_{ij}\mathbf{y}_j, \tag{2}$$

where $1 \leqslant i,\ j \leqslant p$ and the summation convention applies to j. In (2) the *scalar* quantities g_{ij} are specially chosen so that the $\mathbf{y}_i$ remain orthonormal, i.e., so that

$$\mathbf{y}_k^\dagger \mathbf{y}_i = \delta_{ki} \qquad (1 \leqslant k \leqslant p), \tag{3}$$

where a † denotes the complex conjugate transpose and δ is the Kronecker delta.

The conditions which the g_{ij} must satisfy may be found by differentiating (3) and using (2), the solution is not unique and we select the particular one which also satisfies the stronger condition

$$\mathbf{y}_k^\dagger \mathbf{y}_i' = 0, \tag{4}$$

so that the g_{ij} are determined by

$$\mathbf{y}_k^\dagger \mathbf{A}\mathbf{y}_i + g_{ij}\mathbf{y}_k^\dagger \mathbf{y}_j = 0. \tag{5}$$

Given p and $\mathbf{A}$ we may solve the simultaneous linear algebraic equations (5) for g_{ij} in terms of the $\mathbf{y}_i$, substitute in (2) and then integrate (2) to evaluate the $\mathbf{y}_i$ for $0 \leqslant x \leqslant 1$. (For high-order differential systems it may be more appropriate to define the g_{ij} so that they are triangular, whence they may be determined more easily. Note that although they are only scalars they depend upon x and the $\mathbf{y}_i$ in a very nonlinear way.)

As Drury mentions, note that in view of (4) we only allow the $\mathbf{y}_i$ to increment perpendicular to the subspace W which they span. As the integration proceeds the $\mathbf{y}_i$ remain orthonormal and only rotate as much as is necessary for them to stay in W. Now the whole essence of the method lies in the fact that W is the same subspace as V! This is not immediately obvious and so we now indicate how to prove (Drury: private communication) that this is indeed the case. Let $\mathbf{y}$ be the required solution of (1), lying in V, and let the $\mathbf{y}_i$ be solutions of (2), (4), and (5) lying in W. Now consider the derivative of the wedge product $\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y}$; we have

$$\begin{aligned}(\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y})' = {} & (\mathbf{A}\mathbf{y}_1 + g_{1j}\mathbf{y}_j) \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} \\ & + \mathbf{y}_1 \wedge (\mathbf{A}\mathbf{y}_2 + g_{2j}\mathbf{y}_j) \wedge \mathbf{y}_3 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} \\ & \qquad \vdots \\ & + \mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_{p-1} \wedge (\mathbf{A}\mathbf{y}_p + g_{pj}\mathbf{y}_j) \wedge \mathbf{y} \\ & + \mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge (\mathbf{A}\mathbf{y}), \end{aligned} \tag{6}$$

$$\begin{aligned}\therefore (\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y})' = {} & \mathbf{A}\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} \\ & + \mathbf{y}_1 \wedge \mathbf{A}\mathbf{y}_2 \wedge \mathbf{y}_3 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} \\ & \qquad \vdots \\ & + \mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{A}\mathbf{y}_p \wedge \mathbf{y} + \mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{A}\mathbf{y} \\ & + (g_{11} + g_{22} + \cdots + g_{pp})\,\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y}. \end{aligned} \tag{7}$$

However, all the terms on the right-hand side of (7), excluding the last term, constitute a linear mapping from $\Lambda^{p+1}\mathbb{C}^n$ to itself and hence (7) may be written

$$(\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y})' = L(\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y}) + (g_{11} + g_{22} + \cdots + g_{pp})(\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y}), \quad (8)$$

where L is a linear map from $\Lambda^{p+1}\mathbb{C}^n$ to itself.

When $x = 0$ then $\mathbf{y}$ must be a linear combination of the $\mathbf{y}_i$ since the $\mathbf{y}_i$ span V there and thus

$$\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} = \mathbf{0}, \qquad \text{when} \quad x = 0. \quad (9)$$

It follows from (8), and from repeated differentiation of (8), that all the derivatives of the left-hand side of (9) are also zero at $x = 0$. By appealing to analyticity we have therefore that

$$\mathbf{y}_1 \wedge \mathbf{y}_2 \wedge \cdots \wedge \mathbf{y}_p \wedge \mathbf{y} = \mathbf{0}, \qquad \forall\, x, \quad (10)$$

and so *throughout the range of integration* $\mathbf{y}$ *must be a linear combination of the* $\mathbf{y}_i$ and hence W is the same as V. So for each value of x in the range $0 \leqslant x \leqslant 1$ the $\mathbf{y}_i$ form an orthonormal base of V and so in this sense the method is an *automatic* orthonormalization.

To determine the unknown eigenvalue c we proceed in the usual manner: since $\mathbf{y}$ is a linear combination of the $\mathbf{y}_i$ there must exist scalars λ_i such that

$$\mathbf{y} = \lambda_1 \mathbf{y}_1 + \lambda_2 \mathbf{y}_2 + \cdots + \lambda_p \mathbf{y}_p, \quad (11)$$

and the boundary conditions at $x = 1$ will be of the form

$$\mathbf{B}\mathbf{y} = \mathbf{0}, \quad (12)$$

where $\mathbf{B}$ is a complex $p \times n$ matrix. From (11) and (12) it follows that c may be found by using a standard iteration procedure, such as Newton–Raphson or Muller's method, to make the determinant $|d_{ij}|$ zero, to some suitably preassigned tolerance level, where

$$d_{ij} = b_{ik}\, y_{kj}, \qquad 1 \leqslant i, j \leqslant p, \quad 1 \leqslant k \leqslant n, \quad (13)$$

and y_{kj} denotes the kth component of $\mathbf{y}_j$. This iteration will be numerically stable because the $\mathbf{y}_i$ are orthonormal, from the numerical point of view what we are doing is akin to the use of the standard (superposition) shooting method for a non-stiff problem.

Having found the eigenvalue c, as described above, we may now proceed to determine the eigenfunction $\mathbf{y}$ as follows: first, at the final stage of the iteration to find the eigenvalue c, during the last integration from $x = 0$ to $x = 1$ we carefully

store the values of the $\mathbf{y}_i$ *every so often.*[1] Second, we determine the scalars λ_i in (11) by a backward integration from $x = 1$ to $x = 0$ of the differential equations satisfied by the λ_i. To obtain these, differentiate (11), then using (1) and (2) yields

$$\lambda_i' \mathbf{y}_i + \lambda_i g_{ij} \mathbf{y}_j = \mathbf{0}. \tag{14}$$

However, since the $\mathbf{y}_i$ are linearly independent we may equate to zero the coefficient of each $\mathbf{y}_i$ separately in (14); hence

$$\lambda_i' = -g_{ji} \lambda_j. \tag{15}$$

So the eigenfunction $\mathbf{y}$ is found by simultaneously integrating (2) and (15) from $x = 1$ to $x = 0$, and then using (11). The initial values for the λ_i at $x = 1$ are obtained by solving the equation $\mathbf{D}\boldsymbol{\lambda} = \mathbf{0}$.

We have used the above process to determine the eigenfunction for many different stiff problems and never encountered any numerical instability except for the fact that the equations for the $\mathbf{y}_i$ are numerically unstable during the *backward* integration and so their values should be reset every so often to the stored values. Alternatively, the values of the $\mathbf{y}_i$ can be stored at every mesh point on the final forward integration and then (2) need not be integrated backwards; however, this means that more storage space will be needed and it is not so elegant. We discuss in detail the evaluation of an eigenfunction for a very difficult stiff problem in Section 4.

3. The Inhomogeneous Problem

In this section we just mention briefly the way in which the content of the previous section needs to be amended to solve the general linear inhomogeneous problem

$$\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{r}, \tag{16}$$

where $\mathbf{r}$ is a complex n-vector and where the boundary conditions may also be inhomogeneous. Thus we now have a strict two-point boundary-value problem rather than an eigenvalue problem. In order to solve a linear inhomogeneous problem most methods obtain the solution by forming an appropriate combination of the solutions of the associated homogeneous problem with a particular integral and the method of this paper is no exception to this general rule.

We assume that $\mathbf{y}$ has been suitably defined so that the known initial conditions on $\mathbf{y}$ at $x = 0$ are that the first q components of $\mathbf{y}$ are given there hence, say,

$$y_l = \gamma_l, \qquad 1 \leqslant l \leqslant q, \quad \text{at } x = 0, \tag{17}$$

[1] Here, and later, where we use the phrase *every so often* we mean after every tenth or so integration step; the idea is to obtain a suitable balance between the accuracy required and minimization of storage space needed by the computer program.

where y_l is the lth component of $\mathbf{y}$ and the γ_l are known. Now calculate $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_p$ for the associated homogeneous problem $\mathbf{y}' = \mathbf{A}\mathbf{y}$ and zero boundary conditions in the same way as described in Section 2, and *simultaneously* integrate also the additional equation

$$\mathbf{y}'_{p+1} = \mathbf{A}\mathbf{y}_{p+1} + \mathbf{r} + g_{p+1j}\mathbf{y}_j. \tag{18}$$

storing the values of the $\mathbf{y}_i$ and $\mathbf{y}_{p+1}$ every so often. The initial condition at $x = 0$ for $\mathbf{y}_{p+1}$ is given by

$$y_{lp+1} = \gamma_l, \qquad 1 \leqslant l \leqslant q; \qquad y_{kp+1} = 0, \qquad q+1 \leqslant k \leqslant n, \tag{19}$$

and the scalars g_{p+1j} are specially chosen so that $\mathbf{y}_{p+1}$ remains orthogonal to the $\mathbf{y}_i$—this is what physicists sometimes call *re*orthonormalization—that is, we choose them via

$$g_{p+1j}\mathbf{y}_i^\dagger \mathbf{y}_j = -\mathbf{y}_i^\dagger \mathbf{A}\mathbf{y}_{p+1} - \mathbf{y}_i^\dagger \mathbf{r}. \tag{20}$$

The required solution of (16) is now of the form

$$\mathbf{y} = \mathbf{y}_{p+1} + \lambda_i \mathbf{y}_i. \tag{21}$$

and the boundary conditions at $x = 1$ will be of the form

$$\mathbf{B}\mathbf{y} = \boldsymbol{\delta}, \tag{22}$$

where $\mathbf{B}$ is a complex $p \times n$ matrix and $\boldsymbol{\delta}$ is a complex p-vector. Substituting (21) in (22) we see that the initial values of the λ_i at $x = 1$ are given by solving the linear algebraic equations

$$\lambda_i \mathbf{B}\mathbf{y}_i = \boldsymbol{\delta} - \mathbf{B}\mathbf{y}_{p+1}. \tag{23}$$

As in the case of the homogeneous problem we determine the λ_i for $x < 1$ by integrating backwards from $x = 1$ to $x = 0$ the differential equations which they satisfy. By differentiating (21) and using (2), (16), and (18) we find that these equations are

$$\lambda'_i = g_{p+1i} - g_{ji}\lambda_j. \tag{24}$$

Thus, the required solution $\mathbf{y}$ of (16) is found by simultaneously integrating (2), (18), and (24) from $x = 1$ to $x = 0$, and then using (21). During this integration the $\mathbf{y}_i$ and $\mathbf{y}_{p+1}$ should be reset every so often to their previously stored values—see the last paragraph of section 2.

4. A Numerical Example: The Orr–Sommerfeld Equation

To illustrate the power of the method of this paper we will use it to calculate the eigenvalue c, and in particular the associated eigen*function*, for the most unstable mode in the classical linear stability problem of plane Poiseuille flow, when the wavenumber $\alpha = 1$ and the Reynolds number R is very large. For this problem the characteristic values of the differential operator are of order ± 1 and $\pm R^{1/2}$ so that when R is very large then the problem is very stiff indeed.

The differential equation for this problem is the Orr–Sommerfeld equation

$$\{D^2 - \alpha^2 - i\alpha R(1 - x^2 - c)\}\{D^2 - \alpha^2\}\varphi - 2i\alpha R\varphi = 0, \tag{25}$$

and the appropriate boundary conditions for the most unstable mode are

$$\varphi' = \varphi''' = 0, \qquad \text{when} \quad x = 0, \tag{26}$$

and

$$\varphi = \varphi' = 0, \qquad \text{when} \quad x = 1. \tag{27}$$

In the above both D and a' denote differentiation with respect to x. Our ultimate aim will be to calculate the eigenfunction when $R = 10^9$ since, so far as we are aware, no one[2] has managed to do this yet by any method; also this particular value has a close association with asymptotic theory.

In order to formulate the problem in the notation of Section 2 we define

$$\mathbf{y} \equiv (\varphi, \varphi', \varphi'' - \alpha^2\varphi, \varphi''' - \alpha^2\varphi')^T, \tag{28}$$

so that (25) becomes

$$\mathbf{y}' = \mathbf{A}\mathbf{y}, \tag{29}$$

where

$$\mathbf{A} \equiv \begin{pmatrix} 0 & 1 & 0 & 0 \\ \alpha^2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a & 0 & b & 0 \end{pmatrix}, \tag{30}$$

with

$$a = 2i\alpha R, \qquad b = \alpha^2 + i\alpha R(1 - x^2 - c). \tag{31}$$

For this particular problem it transpires that it is better to commence the integration at the wall where $x = 1$ rather than at $x = 0$ because the region where the eigenfunction varies most rapidly is near the wall. So we begin by making an initial

[2] A referee's report which arrived too late to affect the content of this paper indicates that Scott and Watts have also performed this calculation using their orthonormalization code.

estimate for the unknown eigenvalue c and integrating from $x = 1$ to $x = 0$ the eighth-order system

$$\begin{aligned} \mathbf{y}_1' &= \mathbf{A}\mathbf{y}_1 + g_{11}\mathbf{y}_1 + g_{12}\mathbf{y}_2, \\ \mathbf{y}_2' &= \mathbf{A}\mathbf{y}_2 + g_{21}\mathbf{y}_1 + g_{22}\mathbf{y}_2, \end{aligned} \tag{32}$$

where, solving (5),

$$\begin{aligned} dg_{11} &= (\mathbf{y}_1^\dagger \mathbf{y}_2)(\mathbf{y}_2^\dagger \mathbf{A}\mathbf{y}_1) - (\mathbf{y}_2^\dagger \mathbf{y}_2)(\mathbf{y}_1^\dagger \mathbf{A}\mathbf{y}_1), \\ dg_{12} &= (\mathbf{y}_2^\dagger \mathbf{y}_1)(\mathbf{y}_1^\dagger \mathbf{A}\mathbf{y}_1) - (\mathbf{y}_1^\dagger \mathbf{y}_1)(\mathbf{y}_2^\dagger \mathbf{A}\mathbf{y}_1), \\ dg_{21} &= (\mathbf{y}_1^\dagger \mathbf{y}_2)(\mathbf{y}_2^\dagger \mathbf{A}\mathbf{y}_2) - (\mathbf{y}_2^\dagger \mathbf{y}_2)(\mathbf{y}_1^\dagger \mathbf{A}\mathbf{y}_2), \\ dg_{22} &= (\mathbf{y}_2^\dagger \mathbf{y}_1)(\mathbf{y}_1^\dagger \mathbf{A}\mathbf{y}_2) - (\mathbf{y}_1^\dagger \mathbf{y}_1)(\mathbf{y}_2^\dagger \mathbf{A}\mathbf{y}_2), \\ d &= (\mathbf{y}_1^\dagger \mathbf{y}_1)(\mathbf{y}_2^\dagger \mathbf{y}_2) - (\mathbf{y}_2^\dagger \mathbf{y}_1)(\mathbf{y}_1^\dagger \mathbf{y}_2). \end{aligned} \tag{33}$$

The initial conditions for $\mathbf{y}_1, \mathbf{y}_2$ are given by

$$\mathbf{y}_1 = (0, 0, 1, 0)^T, \qquad \mathbf{y}_2 = (0, 0, 0, 1)^T, \qquad \text{at} \quad x = 1. \tag{34}$$

This integration determines the values of $\mathbf{y}_1$ and $\mathbf{y}_2$ at $x = 0$ given the initial estimate of the eigenvalue c.

Now the required eigenfunction $\mathbf{y}$ will be a linear combination of $\mathbf{y}_1, \mathbf{y}_2$ so that

$$\mathbf{y} = \lambda_1 \mathbf{y}_1 + \lambda_2 \mathbf{y}_2, \tag{35}$$

and the boundary conditions (26) tell us that we must have

$$\begin{pmatrix} y_{21} & y_{22} \\ y_{41} & y_{42} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \mathbf{0}, \qquad \text{at} \quad x = 0. \tag{36}$$

Thus, we found the eigenvalue c by using Newton–Raphson iteration until the determinant

$$\Delta = y_{21} y_{42} - y_{22} y_{41}, \tag{37}$$

was sufficiently small, the iteration was terminated when successive iterates differed by less than a pre-assigned tolerance level, usually about 10^{-9}. As a check, and to ensure that we felt confident that we had calculated c correctly to eight decimal places, the whole of the above procedure was repeated for the *adjoint* differential problem.

It is appropriate at this stage to mention a very important point concerning the numerical stability of the method. We know that the method keeps $\mathbf{y}_1, \mathbf{y}_2$ orthonormal so that $\mathbf{y}_1^\dagger \mathbf{y}_1 = \mathbf{y}_2^\dagger \mathbf{y}_2 = 1$ and $\mathbf{y}_1^\dagger \mathbf{y}_2 = \mathbf{y}_2^\dagger \mathbf{y}_1 = 0$ so there is a *temptation* to replace the rather cumbersome Eq. (33) by $g_{11} = -\mathbf{y}_1^\dagger \mathbf{A}\mathbf{y}_1$, $g_{12} = -\mathbf{y}_2^\dagger \mathbf{A}\mathbf{y}_1, \ldots, d = 1$. This simplification is, however, to be sorely resisted as it is not numerically stable. It

is for this reaon that the *mathematical* formulation of the method proposed by Drury [3] (see his equation (22)) is not *numerically* successful, as we discovered in the early stages of our work. The full formulation (33) is essential to ensure that the method is numerically stable.

Having found the eigenvalue c we solve (36) to obtain a solution for λ_1, λ_2 when $x = 0$; since the problem is homogeneous there is an arbitrary scale factor. Next, we determine λ_1, λ_2 for all values of x by a "backward" integration from $x = 0$ to $x = 1$ of the differential equations satisfied by λ_1, λ_2. These are obtained by differentiating (35), using (1), (32), and equating coefficients of $\mathbf{y}_1, \mathbf{y}_2$ separately to yield (see (15))

$$\begin{aligned} \lambda_1' &= -g_{11}\lambda_1 - g_{21}\lambda_2, \\ \lambda_2' &= -g_{12}\lambda_1 - g_{22}\lambda_2. \end{aligned} \tag{38}$$

Equations (32) for $\mathbf{y}_1, \mathbf{y}_2$ are also integrated backwards at the same time as (38), every so often resetting $\mathbf{y}_1, \mathbf{y}_2$ to previously stored values, and then (35) is used to construct the required eigenfunction $\mathbf{y}$.

We concentrated our attention upon finding the eigenvalue c and the eigenfunction $\mathbf{y} \equiv (\varphi, \varphi', \varphi'' - \alpha^2\varphi, \varphi''' - \alpha^2\varphi')^T$ when $\alpha = 1$ and $\log_{10} R = 5(1)9$; the case $R = 10^9$ seemed to be a good final one to consider since the associated asymptotic theory depends heavily on $R^{-1/3} = 0.001$. We used double-precision Fortran (56 mantissa bits) on an IBM 370/168.

As regards the eigenvalue c we also determined the *smallest* number of integration steps of equal length which we can use to obtain c correct to four significant figures and we compare this with the corresponding number using the orthonormalization method in Table I.

When R is very large the number of steps needed by both methods is proportional to $R^{1/2}$ because the main restriction is that the Runge–Kutta integration routine which we used must be convergent. It is clear from Table I that when $R \geqslant 10^7$, so that the characteristic values of the differential operator are *very* widely separated then the method of this paper requires approximately twice as many integration steps

TABLE I

The Number of Integration Steps of Equal Length Required by the Orthonormalization Method (ONIZ) and by the Present Method (PM) to Calculate the Eigenvalue c, Correct to Four Significant Figures, of the Orr–Sommerfeld Problem (25)–(27) for Plane Poiseuille Flow

$\log_{10} R$	c	ONIZ	PM
5[a]	0.14592479–0.01504204i	325	400
6	0.06659252–0.01398327i	600	700
7	0.03064130–0.00726049i	1200	2300
8	0.01417134–0.00351239i	3700	7300
9	0.00656630–0.00166002i	12000	24000

[a] For this entry c was calculated correct to five decimal places instead of four significant figures.

as orthonormalization—this suggests that $\mathbf{y}_1, \mathbf{y}_2$ probably have a tanh rather than an exponential character. The number of iterations needed was usually three or four, and so about the same as when the orthonormalization method is used.

When $R \leqslant 10^6$ so that the characteristic values are not too widely separated then the step length is not so severely restricted and, as Table I indicates, the present method requires scarcely any more steps than the orthonormalization method. Of course really both methods should be used with a sophisticated variable-step integration routine such as RKF45, even if this were done however the *ratio* of the number of steps needed by the two methods would be little changed.

We mentioned earlier that the real power of the method is exemplified by its ability to calculate the eigenfunction $\mathbf{y} \equiv (\varphi, \varphi', \varphi'' - \alpha^2\varphi,\ \varphi''' - \alpha^2\varphi')^T$ for very large values of the parameter R which other methods have not been able to reach. Also, finding the eigenfunction by the orthonormalization method is a laborious accounting task which needs very careful treatment. So we used our method to calculated φ and its derivatives and we encountered no obstacles whatsoever except that for the most difficult case, $R = 10^9$, it was essential to use a variable mesh to obtain φ and φ' correct to about 1 part in 5000, a margin which is not discernible when plotted.

Figure 1 shows the real and imaginary parts of $\varphi = \varphi_r + i\varphi_i$ when $R = 10^6$, as expected we see that the most rapid variation takes place in the vicinity of the so-called critical layer near the wall $x = 1$; note that the top of the peak in φ_i is at about $x = 0.97$. In Table II we give values of φ and φ' for some selected values of x; the normalization used is $\varphi(0) = 1$.

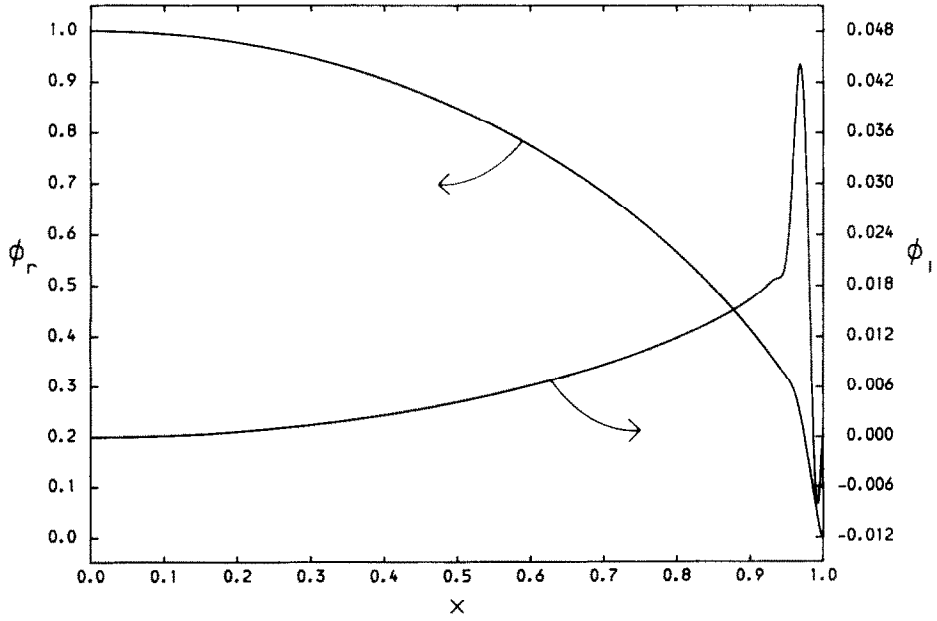

FIG. 1. The real and imaginary parts of the eigenfunction $\varphi = \varphi_r + i\varphi_i$ when $\alpha = 1$ and $R = 10^6$.

TABLE II

Selected Values of the Eigenfunction
and Its Derivative when $\alpha = 1$ and $R = 10^6$

x	φ_r	φ_i	φ_r'	φ_i'
0	1	0	0	0
0.10	0.994275	0.000161	−0.114770	0.003220
0.20	0.976935	0.000646	−0.232886	0.006506
0.30	0.947467	0.001467	−0.357953	0.009931
0.40	0.904973	0.002640	−0.494180	0.013588
0.50	0.848082	0.004196	−0.646944	0.017610
0.60	0.774783	0.006182	−0.823916	0.022231
0.70	0.682080	0.008679	−1.037860	0.027974
0.80	0.565125	0.011864	−1.315809	0.036512
0.90	0.414021	0.016380	−1.752106	0.059504
0.91	0.396177	0.017002	−1.817852	0.065280
0.92	0.377635	0.017708	−1.893744	0.077689
0.93	0.358173	0.018515	−2.002780	0.071782
0.94	0.338027	0.018894	−1.968373	0.026988
0.95	0.319256	0.021543	−1.910555	0.711309
0.96	0.292490	0.034694	−3.942176	1.677579
0.97	0.235171	0.043870	−7.376132	−0.451300
0.98	0.152351	0.022455	−8.894877	−3.397737
0.99	0.059840	−0.006417	−9.229558	−1.299547
1	0	0	0	0

When $R = 10^9$ the region of rapid variation is so narrow that it is barely sensible to show a plot of φ over the whole range $0 \leqslant x \leqslant 1$, we do this in Fig. 2, however, just to indicate to the reader what a difficult calculation this case is. In Fig. 3 we present a more sensible plot of the real and imaginary parts of φ over the range $0.95 \leqslant x \leqslant 1$; note that the top of the peak in φ_i is now at about $x = 0.997$, this value together with the corresponding value of 0.97 for the $R = 10^6$ case implies that the critical-layer region extends all the way to the wall so that it includes the wall-layer region. In Table III we give values of φ and φ' for some selected values of x; the normalization used is again $\varphi(0) = 1$.

Since the value $R = 10^9$ is so large the shape of φ shown in Fig. 3 should be very close to the asymptotic solution for φ of the Orr–Sommerfeld equation and it would be very interesting to be able to make a comparison. Although a lot of work has been done on the asymptotic form of the eigenvalue relation for the Orr–Sommerfeld equation, see, for example [2], much less work has been done on the asymptotic form of the eigenfunction and we do not know where any such eigenfunctions have been plotted. Moreover, most of the asymptotic work has been done for α, R-values which lie on, or very close to, the neutral stability curve, whereas when $\alpha = 1$ then the solution is relatively highly damped, being well away from the neutral stability curve. Consequently, we have not been able to make an asymptotic comparison with Fig. 3, despite the fact that we would very much have liked to do so.

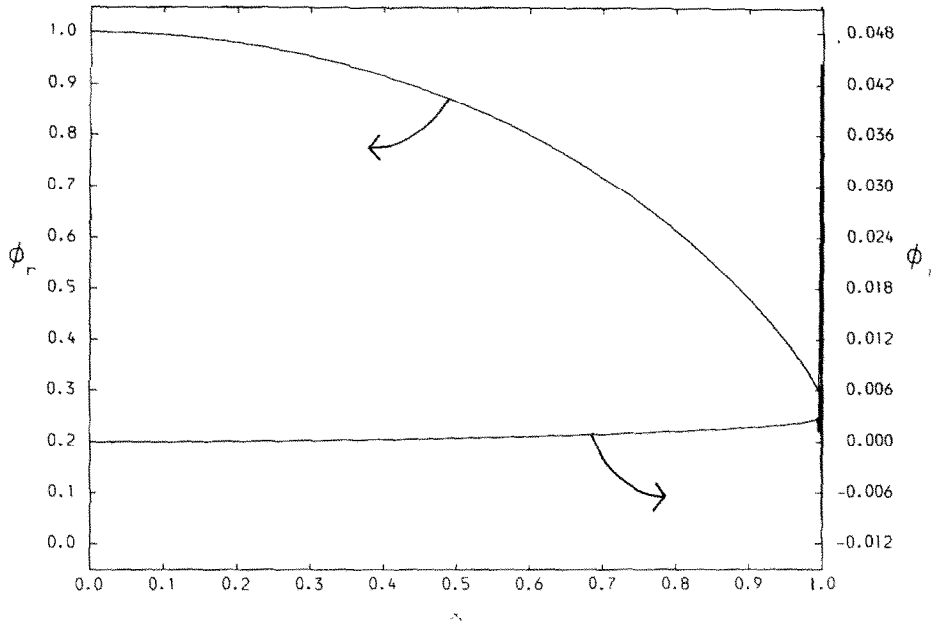

FIG. 2. The real and imaginary parts of the eigenfunction $\varphi = \varphi_r + i\varphi_i$ when $\alpha = 1$ and $R = 10^9$ plotted over the *whole* range $0 \leqslant x \leqslant 1$.

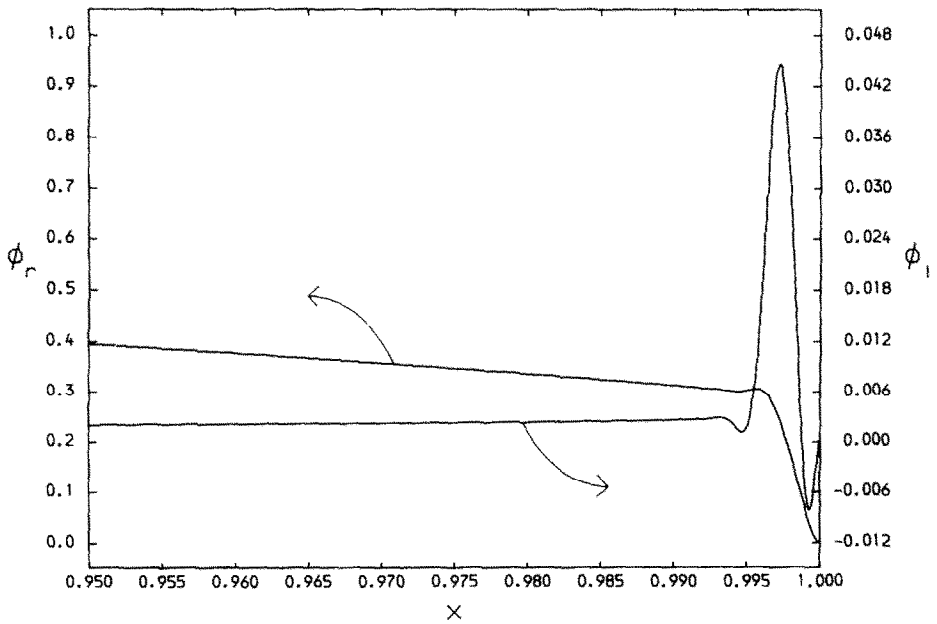

FIG. 3. The real and imaginary parts of the eigenfunction $\varphi = \varphi_r + i\varphi_i$ when $\alpha = 1$ and $R = 10^9$ plotted over the *partial* range $0.95 \leqslant x \leqslant 1$.

TABLE III

Selected Values of the Eigenfunction and Its Derivative when $\alpha = 1$ and $R = 10^9$

x	φ_r	φ_i	φ_r'	φ_i'
0	1	0	0	0
0.90	0.479061	0.001682	−1.5314	0.0055
0.91	0.463504	0.001738	−1.5807	0.0058
0.92	0.447432	0.001798	−1.6346	0.0062
0.93	0.430795	0.001862	−1.6940	0.0067
0.94	0.413527	0.001933	−1.7607	0.0074
0.95	0.395547	0.002012	−1.8374	0.0084
0.96	0.376731	0.002102	−1.9288	0.0099
0.97	0.356894	0.002212	−2.0438	0.0124
0.98	0.335708	0.002360	−2.2045	0.0179
0.99	0.312404	0.002616	−2.4963	0.0403
0.991	0.309883	0.002658	−2.5457	0.0416
0.992	0.307325	0.002710	−2.5717	0.0877
0.993	0.304604	0.002864	−3.0084	0.1040
0.994	0.301606	0.002171	−2.1074	−1.7852
0.995	0.302993	0.001767	4.9551	4.4797
0.996	0.302941	0.018100	−13.6548	27.8864
0.997	0.261255	0.043265	−70.8207	11.3602
0.998	0.171740	0.029031	−101.3322	−36.2021
0.999	0.066749	−0.005572	−103.6835	−17.9406
1	0	0	0	0

5. Concluding Remarks

We have described a new shooting method for solving *stiff* linear eigenvalue and boundary-value problems which solely utilizes well understood techniques for solving similar *non-stiff* problems. The essence of the method is to use the standard shooting method to integrate *not* the given differential system but, instead, a carefully chosen associated nonlinear differential system. The beauty of the method is that it advances an orthonormal frame which spans the space of all those solutions which satisfy the known initial conditions and so no orthonormalization is required.

Other shooting methods for stiff linear problems have serious disadvantages: with the orthonormalization method laborious accounting is needed, especially to reconstruct the required function; with the Riccati method the frequent occurrence of singularities causes technical problems and loss of accuracy; with the compound matrix method for a differential system of order $2n$ one has to integrate $\binom{2n}{n}$ equations, and so use of this method is restricted to low-order differential systems. The present method suffers none of these disadvantages, it is very reliable, it only has to integrate as many differential equations as the orthonormalization method, and we

have used it on a very stiff Orr–Sommerfeld problem to calculate eigenfunctions which no other method has been able to produce. The sole disadvantage of the method is that it is somewhat slower than the orthonormalization method for cases with which the latter can cope—mainly because the g_{ij} terms produce longer right-hand sides to integrate.

Throughout all our numerical work using the method of this paper on many different problems we have never experienced any numerical instability, essentially of course because everything is so nicely nearly orthogonal. In a later paper we shall explain how to develop the ideas contained herein to obtain the solution of stiff nonlinear initial-value problems by solving instead a small finite sequence of non-stiff problems.

ACKNOWLEDGMENT

I would like to express my deep gratitude to Dr. L. O'C. Drury for a most helpful and stimulating correspondence during the course of this work.

REFERENCES

1. S. D. CONTE, *SIAM Rev.* **8** (1966), 309.
2. P. G. DRAZIN AND W. H. REID, "Hydrodynamic Stability," Cambridge Univ. Press, Cambridge, 1981.
3. L. O'C. DRURY, *J. Comput. Phys.* **37** (1980), 133.
4. F. GILBERT AND G. E. BACKUS, *Geophysics* **31** (1966), 326.
5. S. K. GODUNOV, *Usp. Mat. Nauk.* **16** (1961), 171.
6. B. S. NG AND W. H. REID, *J. Comput. Phys.* **30** (1979), 125.
7. M. R. SCOTT, *J. Comput. Phys.* **12** (1973), 334.