SIAM J. SCI. COMPUT.
Vol. 17, No. 2, pp. 430–438, March 1996

© 1996 Society for Industrial and Applied Mathematics
008

BOUNDARY LAYER RESOLVING PSEUDOSPECTRAL METHODS FOR SINGULAR PERTURBATION PROBLEMS*

TAO TANG† AND MANFRED R. TRUMMER‡

Abstract. Pseudospectral methods are investigated for singularly perturbed boundary value problems for ordinary differential equations (ODEs) which possess boundary layers. It is well known that if the boundary layer is very small then a very large number of spectral collocation points is required to obtain accurate solutions. We introduce here a new effective procedure based on coordinate stretching and the Chebyshev pseudospectral method to resolve the boundary layers. Stable and accurate results are obtained for very thin boundary layers with a fairly small number of spectral collocation points.

Key words. spectral methods, singular perturbation, boundary layer

AMS subject classification. 65N35

1. Introduction.

We consider the pseudospectral (PS) method for the singular perturbation boundary value problem (BVP) given by

$$(1)\quad \epsilon u''(x) + p(x)u'(x) + q(x)u(x) = f(x), \quad x \in (-1, 1), \quad u(-1) = \alpha, \ u(1) = \beta,$$

where $\epsilon > 0$ denotes a fixed (small) constant. In many applications, (1) possesses boundary layers, i.e., regions of rapid change in the solution near the endpoints, with widths $o(1)$ as $\epsilon \to 0$. It has been found that PS methods are attractive in solving this problem (see, e.g., [4]). By clustering the gridpoints toward the boundaries, for example, as in the Chebyshev method ($x_j = \cos \frac{\pi j}{N}$, $j = 0, 1, \ldots, N$), PS methods are more efficient than finite difference methods in resolving the boundary layers. However, in performance they still lag behind collocation methods with adaptive mesh selection (e.g., COLSYS [1]).

Although PS methods are remarkably accurate in exact arithmetic, there are a number of difficulties associated with its use. Especially with very small parameter ϵ in (1), large N is required to obtain accurate solutions (see, e.g., [11]). In addition, ill conditioning of the corresponding differentiation matrices with increasing N frequently causes degradation of the observed precision. Furthermore, as clarified in recent studies by Trefethen and Trefethen and Trummer [14, 15] the time step restrictions due to this ill conditioning can be more severe than those predicted by the standard stability theory, if such methods were to be applied to a time-dependent problem. Therefore, there has been considerable interest in recent years in developing well-conditioned spectral methods (see, e.g., [5–7]).

If $\epsilon \ll 1$ (e.g., $\epsilon < 10^{-6}$) and the problem possesses a boundary layer of width $O(\epsilon)$, high accuracy cannot be expected no matter how stable the spectral method is (see, e.g., [5, 11]). In the Chebyshev PS method, the spacing between the collocation points near the boundary is $O(N^{-2})$. For good resolution of the numerical solution at least one of the collocation points ought to lie in the boundary layer, which implies that $N = O(\epsilon^{-1/2})$. If $\epsilon = 10^{-8}$ then about 10^4 collocation points should be used, which is not practical in most calculations.

The Chebyshev spectral method and the finite difference method with coordinate stretching [8, 12] are two potentially useful methods for resolving the boundary layers. However,

*Received by the editors July 13, 1992; accepted for publication (in revised form) October 19, 1994.

†Department of Mathematics, Simon Fraser University, Burnaby, British Columbia V5A 1S6, Canada (tangtao@cs.sfu.ca). The research of this author was supported by Natural Science and Engineering Research Council of Canada grant OGP0105545.

‡Department of Mathematics, Simon Fraser University, Burnaby, British Columbia V5A 1S6, Canada (trummer@sfu.ca). The research of this author was supported by Natural Science and Engineering Research Council of Canada grant OGP0036901.

neither method works well if $\epsilon \ll 1$, since in this case N has to be very large. To avoid this difficulty we combine the two methods (with a new coordinate stretching technique) to solve (1). The idea is simple: first, the problem is replaced by an equivalent one using a transformation of the computational domain; second, the transformed problem is solved with the standard Chebyshev PS method. After the transformation more collocation points lie in the boundary layer than before, and there are collocation points in the layer even for fairly small values of N.

2. Transformations. As mentioned in §1, at least one of the collocation points should lie in a small neighborhood of $x = \pm 1$ in order to resolve the boundary layers. Therefore, we introduce a sequence of variable transformations so that there are some collocation points within distance ϵ from the boundaries ± 1 even for $\epsilon \ll 1$ and $N = O(10)$. These transformations are iterated SINE functions $x = g_m(y)$, $m = 0, 1, \ldots$, where

$$g_0(y) := y, \qquad g_m(y) = \sin\left(\frac{\pi}{2} g_{m-1}(y)\right), \quad m \geq 1. \tag{2}$$

The theorem below characterizes these transformations based on the relative spacing of the transformed Chebyshev–Gauss–Lobatto nodes.

THEOREM 2.1. *The following two statements hold for any integer $m \geq 0$.* (a) *The map g_m is one-to-one and $g_m([-1, 1]) = [-1, 1]$.* (b) *If $y_j = \cos(\frac{j\pi}{N})$, $j = 0, \ldots, N$, then*

$$g_m(y_0) - g_m(y_1) = g_m(y_{N-1}) - g_m(y_N) = \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{m+1}} \left(1 + O(N^{-2})\right).$$

Proof. For (a) We need to show that $g_m'(y) \neq 0$ for $y \in (-1, 1)$, $|g_m(y)| \leq 1$ and $g_m(\pm 1) = \pm 1$, which can be proved by induction (see also (6)). To establish (b), we note that $g_0(y_0) - g_0(y_1) = \frac{\pi^2}{2N^2}\left(1 + O(N^{-2})\right)$. Assuming that

$$g_k(y_0) - g_k(y_1) = \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{k+1}} \left(1 + O(N^{-2})\right)$$

and noting that $g_k(y_0) = g_{k+1}(y_0) = 1$, we obtain

$$\begin{aligned} g_{k+1}(y_0) - g_{k+1}(y_1) &= 1 - \sin\left(\frac{\pi}{2} g_k(y_1)\right) \\ &= 1 - \sin\left(\frac{\pi}{2}\left(1 - \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{k+1}} \left(1 + O(N^{-2})\right)\right)\right) \\ &= 1 - \cos\left(\frac{4}{\pi}\left(\frac{\pi^2}{4N}\right)^{2^{k+1}} \left(1 + O(N^{-2})\right)\right) = \frac{8}{\pi^2}\left(\frac{\pi^2}{4N}\right)^{2^{k+2}} \left(1 + O(N^{-2})\right). \end{aligned}$$

Since $g_m(y_N) = -g_m(y_0)$ and $g_m(y_{N-1}) = -g_m(y_1)$, the proof of (b) is hereby complete. □

From Theorem 2.1 it can be expected that the transformations (2) together with the Chebyshev PS method can deal with extremely small boundary layers with a fairly small number of collocation points. For $m = 1, 2$, and 3 (which correspond to one, two, and three SINE transformations), the distance between each boundary point and its nearest interior point is $O(N^{-4})$, $O(N^{-8})$, and $O(N^{-16})$, respectively. Therefore, even for very small ϵ such as $\epsilon = 10^{-12}$, at least one collocation point lies in the boundary layer even for moderate values of N, if two or three SINE transformations are used.

3. The transformed equations. We transform the singularly perturbed linear BVP (1) via the variable transformation $x \mapsto y(x)$ (or $x = x(y)$) into the new BVP

$$\epsilon v''(y) + P(y)v'(y) + Q(y)v(y) = F(y), \tag{3}$$

where v is the transplant of u, $v(y) = u(x(y))$. The transformed coefficients are

$$P(y) := \frac{p(x)}{y'(x)} + \epsilon \frac{y''(x)}{y'(x)^2}, \tag{4}$$

$$Q(y) := \frac{q(x)}{y'(x)^2}, \qquad F(y) := \frac{f(x)}{y'(x)^2}, \tag{5}$$

where again $x = x(y)$. It is clear from (3)–(5) that for any variable transformation $x \mapsto y(x)$ the two quantities $1/y'(x)$ and $y''(x)/[y'(x)]^2$ are of interest and should be easy to calculate.

We now consider the transformation $x = x(y) := g_m(y)$ of §2. In this case, the computation of $1/y'(x)$ is straightforward. Differentiating the recursion (2) we obtain

$$g_0'(y) = 1, \quad g_m'(y) = \frac{\pi}{2} \cos\left(\frac{\pi}{2} g_{m-1}(y)\right) g_{m-1}'(y), \quad m \geq 1. \tag{6}$$

Since $y'(x) = 1/g_m'(y)$, we have

$$\frac{1}{y'(x)} = \prod_{k=0}^{m-1} \left(\frac{\pi}{2} \cos\left(\frac{\pi}{2} g_k(y)\right)\right), \quad m \geq 1. \tag{7}$$

Now we define the functions $h_m(x)$, mapping $[-1, 1]$ onto itself, recursively via

$$h_0(x) := x, \qquad h_m(x) := \frac{2}{\pi} \arcsin\left(h_{m-1}(x)\right), \quad m \geq 1. \tag{8}$$

LEMMA 3.1. *$h_m = g_m^{-1}$ for $m = 0, 1, \ldots$.*

Proof. The case $m = 0$ is trivial. For $m \geq 1$, we let $z = h_m(g_m(y))$. It can be shown by induction that for $k = 0, \ldots, m$,

$$g_k(z) = h_{m-k}(g_m(y)). \tag{9}$$

For $k = m$ we therefore obtain

$$g_m(z) = h_0(g_m(y)) = g_m(y),$$

and, since g_m is injective, it follows $y = z$; i.e., $y = h_m(g_m(y))$. □

We now proceed to find a recursion for the quantity $h_m''(x)/[h_m'(x)]^2$. From (8) we obtain

$$\sin\left(\frac{\pi}{2} h_m(x)\right) = h_{m-1}(x), \qquad m \geq 1. \tag{10}$$

Differentiating (10) twice with respect to x yields

$$\frac{\pi}{2} \cos\left(\frac{\pi}{2} h_m(x)\right) h_m'(x) = h_{m-1}'(x), \tag{11}$$

$$-\left(\frac{\pi}{2}\right)^2 \sin\left(\frac{\pi}{2} h_m(x)\right) \left(h_m'(x)\right)^2 + \left(\frac{\pi}{2}\right) \cos\left(\frac{\pi}{2} h_m(x)\right) h_m''(x) = h_{m-1}''(x). \tag{12}$$

Finally, using (11) and (12) we obtain the recursion

$$\frac{h_m''(x)}{\left(h_m'(x)\right)^2} = \frac{\pi}{2} \tan\left(\frac{\pi}{2} h_m(x)\right) + \frac{\pi}{2} \cos\left(\frac{\pi}{2} h_m(x)\right) \frac{h_{m-1}''(x)}{\left(h_{m-1}'(x)\right)^2}. \tag{13}$$

Note that $h_0'(x) \equiv 1$ and $h_0''(x) \equiv 0$. Since $y(x) = h_m(x)$, the quantity $y''(x)/[y'(x)]^2$ can be computed easily using (13).

TABLE 1
Maximum errors for Example 1 ("–" *indicates the error is greater than* 10).

		$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$
$\epsilon = 10^{-3}$	$m = 0$	4.39(00)	3.02(−01)	1.60(−04)	6.84(−14)	2.36(−13)
	$m = 1$	1.50(−01)	4.27(−04)	2.22(−11)	9.30(−14)	2.13(−13)
	$m = 2$	2.20(−02)	3.91(−04)	1.74(−09)	5.57(−12)	9.02(−11)
$\epsilon = 10^{-6}$	$m = 1$	–	8.37(00)	2.60(00)	1.14(−01)	2.32(−05)
	$m = 2$	4.77(00)	2.11(−01)	7.50(−03)	6.82(−07)	1.10(−10)
	$m = 3$	1.01(00)	1.73(−01)	2.50(−03)	2.59(−07)	1.08(−10)
$\epsilon = 10^{-9}$	$m = 1$	–	–	–	–	–
	$m = 2$	6.56(−01)	3.20(−01)	3.03(−01)	9.00(−02)	6.25(−5)
	$m = 3$	2.66(00)	9.11(−01)	2.33(−02)	3.06(−04)	1.08(−07)

4. Examples. We denote by Q_N the space of polynomials of degree $\leq N$. We collocate (3) at the Chebyshev–Gauss–Lobatto nodes $y_j = \cos\frac{j\pi}{N}$, $j = 1, \ldots, N-1$, leading to the PS method for (3) as follows: find $v_N \in Q_N$ such that

$$\epsilon v_N''(y_j) + P(y_j)v_N'(y_j) + Q(y_j)v_N(y_j) = F(y_j), \quad j = 1, \ldots, N-1; \tag{14}$$

$$v_N(-1) = \alpha, \quad v_N(1) = \beta. \tag{15}$$

To solve (14) and (15), we have to solve a matrix equation of the form $AV = b$, where $A \in \mathbf{R}^{(N-1)\times(N-1)}$ and $V, b \in \mathbf{R}^{N-1}$, with $V = (V_1, \ldots, V_{N-1})^T$. The $V_j = v_N(y_j)$ are approximations of $v(y_j)$. The matrix equation is solved in MATLAB, which uses the standard LINPACK routines.

Example 1. Our first example has variable coefficients and the solution develops two boundary layers of width $O(\epsilon)$ near $x = \pm 1$. The equation is

$$\epsilon u''(x) - xu'(x) - u(x) = f(x) = \left(\frac{x+1}{\epsilon} - 1\right)e^{\frac{x+1}{\epsilon}} - 2\left(\frac{x-1}{\epsilon} + 1\right)e^{\frac{x-1}{\epsilon}}, \tag{16}$$

where f is chosen such that the function

$$u(x) = e^{-\frac{(x+1)}{\epsilon}} + 2e^{\frac{x-1}{\epsilon}} \tag{17}$$

is an exact solution of the differential equation. The boundary conditions are $u(-1) = 1$ and $u(+1) = 2$. Note that function (17) will satisfy these boundary conditions to machine precision (machine epsilon equals $2.22 * 10^{-16}$ in double precision) for all values of $\epsilon \leq 0.05$.

This is a difficult problem since high resolution is needed to avoid oscillations in the middle of the interval. The Chebyshev PS method *without* transformation fails to resolve the solution satisfactorily for $\epsilon = 10^{-4}$, even with $N = 256$ (the maximum error, defined by $\max_j\{|v(y_j) - V_j|\}$, is approximately equal to 0.13 in this case, compared with errors of approximately 2×10^{-12} for $m = 1$ and $m = 2$). Table 1 contains the results of our experiments for $\epsilon = 10^{-3}$, $\epsilon = 10^{-6}$, and $\epsilon = 10^{-9}$.

Figure 1 shows the plot of the solution for $\epsilon = 10^{-9}$, $N = 256$, and $m = 3$, and Figure 2 shows the corresponding error. It may not come as a surprise to find the major portion of the error located in the middle of the interval since we have a coarser grid spacing there. However, it is interesting to note that in this case the strategy of moving points out of the region of large errors actually helps in the solution process. This indicates that a strategy for adaptive gridding will have to be rather sophisticated, as it would appear natural to move more points *into* the region exhibiting large errors.

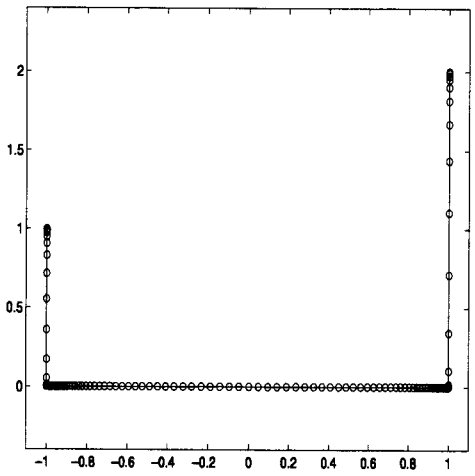

FIG. 1. *Numerical solution of Example* 1 *for* $\epsilon = 10^{-9}$, $N = 256$, *and* $m = 3$.

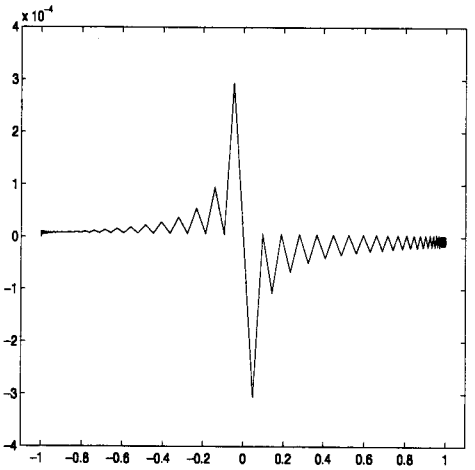

FIG. 2. *Error of Example* 1 *for* $\epsilon = 10^{-9}$, $N = 256$, *and* $m = 3$.

Example 2. Our second example is a nonlinear one; namely the stationary Burgers equation

$$\epsilon u''(x) + u(x)u'(x) = 0, \qquad x \in [-1, 1], \tag{18}$$

with boundary conditions chosen such that the function

$$u(x) = \tanh\left(\frac{x+1}{2\epsilon}\right) \tag{19}$$

is an exact solution. This function is 0 at the left boundary, and extremely close to 1 for most of the interval, with a boundary layer of width $O(\epsilon)$ at $x = -1$. The transformed equation with new variable $y = y(x)$ is simply

$$\epsilon v''(y) + \left[\frac{1}{y'(x)}v(y) + \epsilon\frac{y''(x)}{y'(x)^2}\right]v'(y) = 0, \qquad y \in [-1, 1]. \tag{20}$$

The solution is computed by Newton's method with $v \equiv 1$ as an initial guess; for small values of the parameter ϵ a continuation procedure for ϵ to obtain better initial guesses is advisable, and at times essential. Newton's method converges quickly and often in monotone (convergence problems during the first few iterations appear to indicate insufficient resolution of the discretization). Table 2 lists the results for $\epsilon = 10^{-3}$, $\epsilon = 10^{-6}$, and $\epsilon = 10^{-9}$.

A similar procedure has been applied to obtain numerical solutions for

$$\epsilon u''(x) + \frac{1}{2}u(x)u'(x) - \frac{1}{4}u(x) = 0, \qquad x \in [-1, 1], \tag{21}$$

with boundary conditions $u(-1) = u(1) = \frac{1}{2}$. This problem has the same type of nonlinearity as the stationary Burgers equation; it has been studied in detail in [9], and has been solved with COLSYS [2, pp. 382–383]. We find that for $\epsilon = 10^{-4}$ and $N = 64$ the method without transformation is developing oscillations near the boundary layer, whereas the approximation obtained with one SINE transformation ($m = 1$) easily resolves the boundary layer. Our results appear to be more accurate than the ones obtained with COLSYS for a comparable number of collocation points. In fact, with $m = 2$ we have no problem in resolving the boundary layer with $N = 128$ for ϵ as small as $\epsilon = 10^{-8}$ (see [13] for more details).

TABLE 2
Maximum errors for Example 2 ("⋆" indicates an error > 1 or convergence difficulties in the Newton process).

		$N = 32$	$N = 64$	$N = 128$	$N = 256$
$\epsilon = 10^{-3}$	$m = 0$	⋆	1.8144(−02)	3.6293(−04)	3.3776(−07)
	$m = 1$	3.5818(−02)	4.5561(−04)	1.3573(−07)	3.4528(−14)
	$m = 2$	1.6063(−02)	3.6709(−04)	6.3134(−08)	4.9238(−14)
$\epsilon = 10^{-6}$	$m = 1$	⋆	⋆	4.3554(−02)	1.3762(−03)
	$m = 2$	⋆	2.5004(−02)	2.4636(−03)	9.7656(−07)
	$m = 3$	⋆	3.7734(−02)	7.1848(−04)	2.3793(−07)
$\epsilon = 10^{-9}$	$m = 1$	⋆	⋆	⋆	⋆
	$m = 2$	⋆	⋆	⋆	6.1103(−03)
	$m = 3$	⋆	⋆	6.7784(−03)	1.0602(−04)

5. Conditioning. Some recent work on spectral methods for BVPs is concerned with improving the condition numbers of the matrices for which linear systems have to be solved (e.g., [3, 5, 7]). Since the second-order Chebyshev differentiation matrix has a condition number $O(N^4)$, the corresponding linear systems quickly become very ill conditioned, even for moderate values of N. Interestingly enough, these large condition numbers do not appear to affect the accuracy in the solutions nearly as badly as one would expect. This was first observed by Berrut [3], who transformed the BVP to the circle and solved it with the much better conditioned Fourier spectral method, without seeing any improvement in the accuracy of solutions. However, the large condition numbers would be important in time stepping (so in this sense the PDE case is more difficult than the ODE case), or, if one were to solve the linear systems by iterative methods.

We would like to give a heuristic argument why our solutions are surprisingly accurate (we get close to machine precision, even in cases where the condition number of the linear system is approximately 10^8). Denoting the n-by-n matrix ($n = N - 1$) of our linear system by A, we compute the singular value decomposition $A = W\Sigma V^T$. Σ is a diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ on its diagonal. The singular vectors $v_1, v_2, \ldots,$ are the columns of V. Both V and W are orthogonal n-by-n matrices. It is easy to see that the maximum magnification of roundoff errors in the right-hand side occurs when the exact solution u of the system is a multiple of the first singular vector v_1 and the perturbation δu is entirely in the direction of the last singular vector v_n. Figure 3 shows plots of four of the singular vectors[1] for the matrix A of Example 1, with $\epsilon = 10^{-2}$. Singular vectors belonging to large singular values are highly oscillatory, whereas singular vectors associated with small singular values are smooth (here, v_j has $n + 1 - j$ local extrema). This is not surprising, as A is a discretization of a differential operator, and therefore the statement above holds not only for Example 1. The exact solution has a substantial smooth component and roundoff errors cannot be expected to produce a completely smooth perturbation to the exact solution—on the contrary, a nonsmooth perturbation is much more likely to emerge. Thus, the actual amplification of the roundoff error is much smaller than the worst-case scenario of an amplification by $\text{cond}(A) = \sigma_1/\sigma_n$.

The condition numbers of the matrices generated by our repeated SINE transformations exhibit the same growth rates with N as the matrices for the original problem. The conditioning problem is largely unaffected by the transformation.

[1]The vectors are plotted against a stretched version of the interval (0, 1) to make the oscillations near the boundaries more visible.

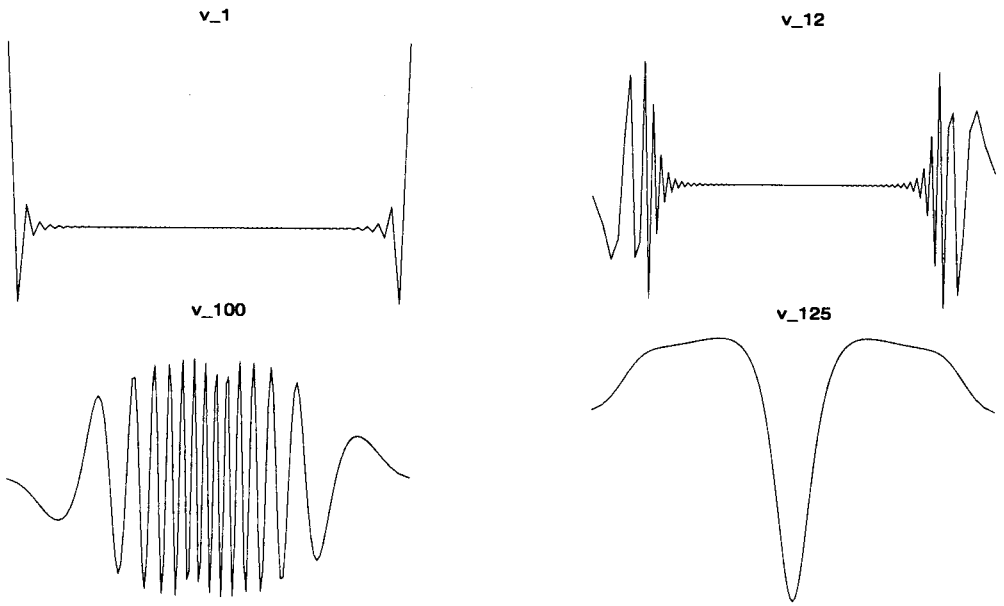

FIG. 3. *Singular vectors* v_1, v_{12}, v_{100}, *and* v_{125} *of A of Example* 1 *for* $\epsilon = 10^{-2}$, *plotted against a stretched interval* (0, 1).

Denoting again the Chebyshev–Gauss–Lobatto nodes by $y_j = \cos \frac{\pi j}{N}$, $j = 0, \ldots, N$, the first-order Chebyshev differentiation matrix D is given by

$$D_{kj} = \frac{c_k}{c_j} \frac{(-1)^{j+k}}{y_k - y_j}, \quad k \neq j, \tag{22}$$

$$D_{kk} = -\frac{y_k}{2(1 - y_k^2)}, \quad k \neq 0, N, \tag{23}$$

$$D_{00} = -D_{NN} = \frac{2N^2 + 1}{6}, \tag{24}$$

where $c_k = 1$, except for $c_0 = c_N = 2$. It has been observed [4] that for large N the direct implementation of (22)–(24) suffers from cancellation, causing errors in the elements of the matrix D. Thus, it is advisable to replace (22) and (23) using trigonometric identities by the formulas

$$D_{kj} = \frac{c_k}{c_j} \frac{(-1)^{j+k}}{\sin((k + j)\pi/(2N)) \sin((k - j)\pi/(2N))}, \quad k \neq j, \tag{25}$$

$$D_{kk} = -\frac{y_k}{2 \sin^2(k\pi/N)}, \quad k \neq 0, N. \tag{26}$$

Finally, to avoid computing the SINE of arguments larger than $\pi/2$ in absolute value, one can take advantage of the symmetry property

$$D_{N-k,N-j} = D_{kj}. \tag{27}$$

Thus the most accurate method of computing D is using formulas (25)–(26) to find the *upper left triangle* of D (i.e., compute D_{kj} with $k + j \leq N$), and then use relation (27) for the other elements. This also appears to be more efficient (at least in our MATLAB implementation).

It should be noted that the effect of a more accurate matrix D cannot always be felt. To be noticeable, N has to be quite large, and the approximate solution must be extremely accurate. For Example 1 (see §4) with $\epsilon = 10^{-2}$, $N = 128$, and $m = 2$, the maximum error is $1.29 * 10^{-14}$, if the more accurate D is used, whereas the error with D computed by formulas (22)–(24) is $1.22 * 10^{-13}$.

6. Conclusions. Very thin boundary layers still must have one or more collocation points within the boundary layer. This results in extremely fine discretizations if the relative spacing of the gridpoints remains unchanged. Although the Chebyshev PS methods are more efficient than finite difference methods in resolving boundary layers, for $\epsilon \ll 1$ they still may need extremely large N to produce reasonable results, as discussed in §1. A much better approach for resolving the boundary layer is to use a mapping. However, a single mapping such as that of [8] is often not sufficient when $\epsilon \ll 1$.

To obtain good resolution for boundary layer problems, at least one of the grid points should lie in the boundary layer no matter how small the boundary layer is. The iterated SINE transformations introduced in §2 provide a very useful coordinate stretching technique to achieve this goal. Theoretically, as indicated in Theorem 2.1, these particular transformations together with the Chebyshev PS method can deal effectively with very small boundary layers using only a fairly small number of collocation points. Even for very small ϵ such as $\epsilon = 10^{-9}$, two or three SINE transformations with $N \approx 100$ are found to be sufficient to resolve the boundary layer, while most of the previously reported finite difference or spectral calculations cannot handle the case when ϵ is as small as 10^{-9}.

Section 3 of this paper gives a practical procedure for implementing the transformations. The transformation technique is also successful for nonlinear BVPs whose solutions have boundary layers. To date the most reliable methods for solving two-point BVPs are based on the collocation method with adaptive mesh selection (e.g., COLSYS [1, 2]). However, for boundary layer problems the present method is a serious competitor, in particular when spectral accuracy is a desirable feature.

The ill conditioning of the linear systems to be solved does not appear to be a serious problem as our experiments and the heuristic argument in §5 indicate. However, care must be taken if one uses these matrices in explicit time stepping in the time-dependent case, or, in the ODE case, if iterative methods are employed to solve the linear system.

Many practical problems possess boundary layers. For example, viscous flows have boundary layers next to solid surfaces where the tangential velocity is reduced to zero. The use of the finite difference method or the Chebyshev PS method is expensive for high Reynolds number flows. The numerical technique introduced in this work can be applied to solve more practical problems (see, e.g., [10]).

REFERENCES

[1] U. M. ASCHER, J. CHRISTIANSEN, AND R. D. RUSSELL, *A collocation solver for mixed order systems of boundary value problems*, Math. Comp., 33 (1979), pp. 659–679.

[2] U. M. ASCHER, R. M. M. MATTHEIJ, AND R. D. RUSSELL, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, Prentice–Hall, Englewood Cliff, NJ, 1988.

[3] J. P. BERRUT, *A pseudospectral Čebyšev method with preliminary transform to the circle: Ordinary differential equations*, Technical rep. 252, Math. Institut, Techn. Universität München, Germany, 1990; private communication, 1989.

[4] C. CANUTO, M. Y. HUSSAINI, A. QUARTERONI, AND T. A. ZANG, *Spectral Methods in Fluid Dynamics*, Series of Computational Physics, Springer–Verlag, Heidelberg, Berlin, New York, 1988.

[5] H. EISEN AND W. HEINRICHS, *A new method of stabilization for singular perturbation problems with spectral methods*, SIAM J. Numer. Anal., 29 (1992), pp. 107–122.

[6] B. FORNBERG, *An improved pseudospectral method for initial-boundary value problems*, J. Comput. Phys., 91 (1990), pp. 381–397.

[7] L. GREENGARD, *Spectral integration and two-point boundary value problem*, SIAM J. Numer. Anal., 28 (1991), pp. 1071–1080.

[8] E. KALNAY DE RIVAS, *On the use of nonlinear grids in finite-difference equations*, J. Comput. Phys., 10 (1972), pp. 202–210.

[9] J. KEVORKIAN AND J. COLE, *Perturbation Methods in Applied Mathematics*, Springer–Verlag, Berlin, 1981.

[10] W. LIU AND J. SHEN, *A new efficient spectral Galerkin method for singular perturbation problems*, preprint, 1994 (submitted).

[11] Y. Y. LIU, *The pseudospectral Chebyshev method for two-point boundary value problems*, master's thesis, Dept of Mathematics and Statistics, Simon Fraser University, Burnaby, B.C., Canada, 1992.

[12] S. A. ORSZAG AND M. ISRAELI, *Numerical simulation of viscous incompressible flows*, Ann. Rev. Fluid Mech., 6 (1974), pp. 281–318.

[13] T. TANG AND M. R. TRUMMER, *Boundary layer resolving pseudospectral method for singular perturbation problems*, Research report 93-06, Dept of Mathematics and Statistics, Simon Fraser University, B.C., Canada, 1993.

[14] L. N. TREFETHEN, *Lax-stability vs. eigenvalue stability of spectral methods*, in Numerical Methods for Fluid Dynamics III, K. W. Morton and M. J. Baines, eds., Clarendon Press, Oxford, 1988.

[15] L. N. TREFETHEN AND M. R. TRUMMER, *An instability phenomenon in spectral methods*, SIAM J. Numer. Anal., 24 (1987), pp. 1008–1023.