Journal of Computational and Applied Mathematics 290 (2015) 159–173

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Spectral integration of linear boundary value problems

Divakar Viswanath

Department of Mathematics, University of Michigan, United States

ARTICLE INFO

Article history:
Received 16 August 2014
Received in revised form 27 April 2015

Keywords:
Chebyshev series
Direct numerical simulation of turbulence
Spectral integration

ABSTRACT

Spectral integration was deployed by Orszag and co-workers (1977, 1980, 1981) to obtain stable and efficient solvers for the incompressible Navier–Stokes equation in rectangular geometries. Since then several variations of spectral integration have appeared in the literature. In this article, we derive yet more versions of spectral integration. These new versions of spectral integration rely exclusively on banded matrices as opposed to banded matrices bordered with dense rows. In addition, we derive a factored form of spectral integration which relies only on bi- and tri-diagonal matrices. Key properties, such as the accuracy of spectral integration even when Green's functions are not resolved by the underlying grid and the accuracy of spectral integration in spite of ill-conditioning of underlying linear systems are investigated. Timed comparisons show that reducing spectral integration to bi- and tri-diagonal systems leads to significant speed-ups.

1. Introduction

One of the earliest methods for solving the incompressible Navier–Stokes equation was proposed in a pioneering paper by Orszag [1]. In that paper, Orszag tackled the problem of numerically integrating wall-bounded shear flows using Chebyshev series expansions. The Chebyshev polynomial is defined by $T_n(y) = \cos(n \arccos y)$ for $-1 \leq y \leq 1$. If $u(y) = \alpha_0 T_0/2 + \sum_{j=1}^{\infty} \alpha_j T_j$ is the Chebyshev series of $u(y)$, we denote the Chebyshev coefficient α_n by $\mathcal{T}_n(u)$. The points $y_j = \cos(j\pi/M)$, $j = 0, \ldots, M$, are the Chebyshev grid points. The discrete cosine transform may be used to pass back and forth between the physical domain function values $u(y_j)$, $0 \leq j \leq M$, and the coefficients in the Chebyshev expansion $\alpha_0 T_0/2 + \sum_{j=1}^{M-1} \alpha_j T_j + \alpha_M T_M/2$, if $\alpha_j = 0$ for $j > M$.

The method proposed by Orszag in [1] is certainly complete. However, it is much too expensive. It does not appear to have been implemented and therefore its effectiveness cannot be gaged. Nevertheless, Orszag and co-workers [2,3] derived an effective algorithm for the integration of the incompressible Navier–Stokes equations in rectangular geometries that has been widely used along with a few other popular modifications for more than two decades [4].

The method of spectral integration was introduced by Gottlieb and Orszag as a reformulation of the tau-equations [5, p. 119]. It forms the basis of widely used methods for the solution of the incompressible Navier–Stokes equations [4,2]. Below and throughout this paper, D denotes d/dy. The Chebyshev tau equations for a boundary value problem such as

$$\left(D^2 - a^2\right) u = f(y), \quad u(\pm 1) = 0, \tag{1.1}$$

are obtained by expanding the solution u in a truncated Chebyshev series and equating the Chebyshev coefficients of $T_0, \ldots, T_{M-2}$ in the expansion of $\left(D^2 - a^2\right) u$ to those of f, and enforcing the boundary conditions to get two more equations. As Gottlieb and Orszag noted the tau equations are dense and not well-conditioned. Their method of rewriting gives a tridiagonal system bordered by dense rows corresponding to the boundary conditions.

E-mail address: divakar@umich.edu.

http://dx.doi.org/10.1016/j.cam.2015.04.043
0377-0427/© 2015 Elsevier B.V. All rights reserved.

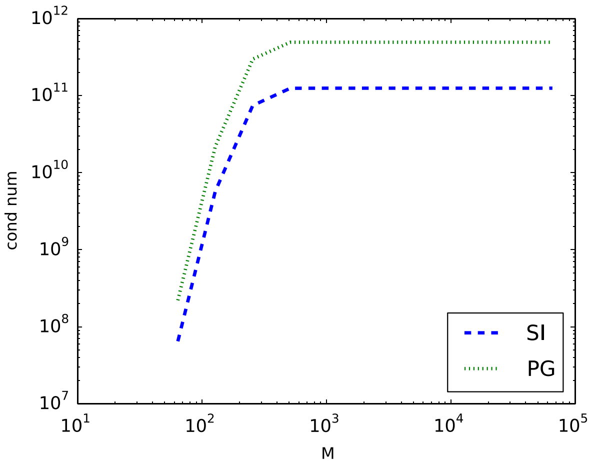

Fig. 1.1. Plot of the infinite-norm condition number vs the number of grid point M for two methods, spectral integration (SI) and Petrov–Galerkin (PG). The methods are applied to the 4th order problem $\left(D^2-\alpha^2\right)\left(D^2-\beta^2\right)u=f$ with $\alpha=10^3$ and $\beta=10^4$. The condition number indeed converges to a constant as M increases but the constant is very large.

In Section 2, we derive a variety of spectral integration methods. All the methods of Section 2 work with purely banded matrices and no bordering rows. Later in this introduction, and in Section 4, we argue that eliminating bordering by dense rows leads to a more efficient solver. The main reason for greater efficiency is that bi- and tri-diagonal solvers are included in the LAPACK library for which highly optimized implementations, such as Intel MKL, are available. Optimized implementations, such as Intel MKL, are continually updated to keep up with changes in computer architecture. A hand coded implementation, which would be required for banded matrices bordered by dense rows, is unlikely to be as well optimized and even more unlikely to stay up-to-date with changes in computer architecture.

A property brought to light by Greengard [6] is that condition numbers of spectral integration matrices, corresponding to boundary value problems such as (1.1), are bounded in the limit $M\to\infty$. As noted by Rokhlin [7], any integral formulation has this property because the integral operators that are discretized are compact. In contrast, the tau equations discretize (1.1) in its differential form and therefore suffer from ill-conditioning. In particular, their condition number goes to ∞ as $M\to\infty$. This property of spectral integration has been noted by other authors as well and spectral integration has been deemed to be well-conditioned [5,8].

Although this may be a useful property, it is by itself inadequate to understand the robustness of spectral integration as applied to the Navier–Stokes equations in the turbulent regime. Fig. 1.1 depicts a scenario, typical of the incompressible Navier–Stokes equations in rectangular geometries [9], where the constant the condition number converges to as M increases is greater than 10^{11}. These matrices cannot be considered well-conditioned. The figure shows condition numbers (computed using the dgbcon and dpbcon routines in LAPACK) for a version of spectral integration derived in Section 2 and for the Petrov–Galerkin method of Shen [10–12], which uses Legendre polynomials. Plots of condition numbers would look the same for any version of spectral integration. Even though the condition number implies a loss of 11 digits of accuracy, we show in Section 4 that such systems are solved with almost machine precision. In Section 3, we give a partial explanation of this phenomenon. Contrary to what the condition numbers suggest the Petrov–Galerkin method is the most accurate.

Iterative methods have been championed for the solution of linear systems that arise after the discretization of integral formulations [7]. In this instance, iterative methods would be of little use because the constants the condition numbers converge to are so large. It is not enough for a method to be $\mathcal{O}(M)$. The constant in front of the M can make a big difference. In Section 4, we find that the speed-up between even highly optimized implementations can approach and exceed a factor of 2.

A number of numerical examples are included in Section 4. The example in Section 4.1 shows that the forms of spectral integration derived in Section 2 match the accuracy of earlier computations [6,13]. One of the forms of spectral integration derived in Section 2 allows for piecewise Chebyshev grids. The example in Section 4.2 shows that this method reduces the number of grid points from 1024 in an earlier computation [14] to only 161, while reducing the relative error from 10^{-4} to 10^{-10}.

Piecewise Legendre grids, which are analogous to piecewise Chebyshev grids, have been considered by Diamessis et al. [15] in the context of stratified flow. The patching conditions which occurs between subdomains are handled by Diamessis et al. using a penalty term. In our method they are handled explicitly.

In Section 4.3, we give a timed comparison between the two different versions of spectral integration in Section 2 and the Petrov–Galerkin method [10]. All our implementations use highly optimized library functions for solving linear systems and for computing the discrete cosine transform. Even so, spectral integration relying on pentadiagonal systems is found to consume 50% more time than spectral integration using tridiagonal systems. The reason is that solving two tridiagonal systems using the optimized MKL library is much cheaper than solving one pentadiagonal system. Some of the issues that arise in such optimized implementations are discussed. Explicit comparison to spectral integration with dense bordered rows is not given because optimized solvers in that setting are not available. Without an optimized implementation, the solvers are unlikely to come even within a factor of 2.

Our implementation of the Petrov–Galerkin method [10] relies on the fast Legendre transform derived and implemented by Alpert and Rokhlin [7].[1] Although the Legendre transform is fast, it cannot match the speed of MKL's discrete cosine transform. As a result the Petrov–Galerkin method is slower by about a factor of 3. However, the Petrov–Galerkin method is notably more accurate (by 2 digits) in resolving boundary layers.

The spectral integration solvers are deployed on a multi-core platform. The resulting speed-up with 16 cores is greater than a factor of 10 and consistent with scaling of bandwidth to memory.

This paper is entirely concerned with linear boundary value problems. Certain additional issues such as the occurrence of spurious eigenvalues arise for linear eigenvalue problems [16,17], although their formulation is analogous to that of boundary value problems.

2. Varieties of spectral integration

In this section, the interval of the boundary value problem is taken to be $-1 \leq y \leq 1$ and the solution u or one of its derivatives is expanded as follows:

$$\frac{\alpha_0}{2} + \alpha_1 T_1 + \cdots + \alpha_{M-1} T_{M-1} + 0.T_M. \tag{2.1}$$

The Chebyshev points $y_j = \cos(j\pi/M)$ with $j = 0 \ldots M$ are $M + 1$ in number including the endpoints ± 1, but the last coefficient in the Chebyshev series is suppressed for convenience as indicated in (2.1).

In each of the methods of this section including the factored form of spectral integration, the boundary conditions are enforced by adding suitable multiples of homogeneous solutions (as in the method of undetermined coefficients). However, the homogeneous solutions are computed in a special way. As explained in Section 3.1, such a roundabout calculation is essential for ensuring accuracy.

All the methods we derive rely upon the solution of banded systems as opposed to banded systems bordered by a few dense rows. The main advantage is that banded solvers are a part of the LAPACK library and therefore highly optimized implementations are available. Even within an optimized implementation, tridiagonal solvers are faster than general banded solvers. One of the methods we derive reduces the solution of linear boundary value problems to bi- and tri-diagonal systems, assuming a numerical accurate factorization of the linear differential operator. In Section 4, we find this method to be significantly faster than a solver that relies on pentadiagonal matrices.

The methods we discuss assume constant coefficients. There are other forms of spectral integration that apply when the coefficients are low order polynomials [14,8].

2.1. First and second order spectral integration

Suppose a first order boundary value problem is given in the form

$$(D - a)u = f. \tag{2.2}$$

The exact boundary condition is unimportant for much of the method. To begin with we assume the integral condition $\mathcal{T}_0(u) = 0$ or equivalently $\alpha_0 = 0$. Suppose that the Chebyshev coefficients of f are given by $f_j = \mathcal{T}_j(f)$.

The indefinite integral of (2.2) gives

$$u - a \int u + A = \int f, \tag{2.3}$$

where A is an undetermined constant. The integral $\int T_n\, dy$ is $T_{n+1}/2(n+1) - T_{n-1}/2(n-1)$ if $n > 1$, $T_2/4$ if $n = 1$, and T_1 if $n = 0$. Therefore, the coefficient of T_n on the right hand side of (2.3) is

$$\mathcal{T}_n\left(\int f\right) = \frac{f_{n-1} - f_{n+1}}{2n} \quad \text{for } n = 1, 2, \ldots, M-1.$$

The $n = M - 1$ case assumes $f_M = 0$. Similarly, the coefficient of T_n in the expansion of the left hand side of (2.3) is

$$\mathcal{T}_n\left(u - a\int u + A\right) = \begin{cases} \alpha_n - a\left(\dfrac{\alpha_{n-1} - \alpha_{n+1}}{2n}\right) & 2 \leq n \leq M-2 \\ \alpha_n - a\left(\dfrac{-\alpha_{n+1}}{2n}\right) & n = 1 \\ \alpha_n - a\left(\dfrac{\alpha_{n-1}}{2n}\right) & n = M-1 \end{cases}$$

[1] I thank Dr. Bradley Alpert for sending me the Fortran source for the fast Legendre transform.

for $n = 1, \dots, M-1$. The coefficients with $n = 1$ and $n = M-1$ are obtained from the more general expression $\alpha_n - a(\alpha_{n-1} - \alpha_{n+1})/2n$ by setting $\alpha_0 = 0$ and $\alpha_M = 0$, respectively. Equating coefficients for $n = 1, \dots, M-1$ we have $M-1$ equations for the $M-1$ unknowns $\alpha_1, \dots, \alpha_{M-1}$. If we did not set $\alpha_M = 0$, another equation with $n = M$ may be used, but that equation has a different form from the equations for $1 \le n \le M-1$. Setting $\alpha_M = 0$ saves us from a little inconvenience. This tridiagonal linear system is solved to compute a particular solution u^p satisfying $(D-a)u^p = f$ and the integral condition $\mathcal{T}_0(u^p) = 0$.

A homogeneous solution $\bar{u}^1$ satisfying $(D-a)\bar{u}^1 = 0$ and $\mathcal{T}_0(\bar{u}^1) = 1$ is found as follows. We set $\bar{u} = 1/2 + u^*$ so that u^* satisfies $\mathcal{T}_0(u^*) = 0$ and $(D-a)u^* = a/2$. Thus u^* is the particular solution of (2.2) satisfying $\mathcal{T}_0(u^*) = 0$ if $f \equiv -a/2$ and it may be found using the method described for computing particular solutions. The same linear tridiagonal system is solved for computing u^* and u^p but with different right hand sides.

The solution u is expressed as $u^p + C\bar{u}^1$ and the constant C is found using the boundary condition on u.

Now we consider the second order problem

$$\left(D^2 + bD + c\right) u = f. \tag{2.4}$$

Integrating twice assuming b, c to be constant, we have

$$u + b\int u + c\iint u + A + By = \iint f.$$

To find a particular solution, we assume the integral conditions $\mathcal{T}_0(u) = \mathcal{T}_1(u) = 0$ or equivalently $\alpha_0 = \alpha_1 = 0$. By standard formulas for $\int T_n$ and $\iint T_n$, the coefficient of T_n of the right hand side is

$$\mathcal{T}_n\left(\iint f\right) = \frac{f_{n-2}}{4n(n-1)} - \frac{f_n}{2(n^2-1)} + \frac{f_{n+2}}{4n(n+1)}$$

for $n = 2, \dots, M-1$ (here $f_M = f_{M+1} = 0$ is assumed). The coefficient of the left hand side is

$$\begin{aligned}\mathcal{T}_n\left(u + b\int u + c\iint u + A + By\right) = \alpha_{n-2}\left(\frac{c}{4n(n-1)}\right) + \alpha_{n-1}\left(\frac{b}{2n}\right) + \alpha_n\left(1 - \frac{c}{2(n^2-1)}\right)\\ + \alpha_{n+1}\left(\frac{-b}{2n}\right) + \alpha_{n+2}\left(\frac{c}{4n(n+1)}\right)\end{aligned}$$

for $n = 2, \dots, M-1$. The validity of the equations for $n = 2, 3$ relies on the boundary conditions $\alpha_1 = \alpha_2 = 0$. The equations for $n = M-2, M-1$ assume $\alpha_M = \alpha_{M+1} = 0$. The coefficients for $n = 2, \dots, M-1$ on the left and right hand sides are equated to solve for the unknowns $\alpha_2, \dots, \alpha_{M-1}$. The particular solution u^p obtained in this manner satisfies $(D^2 + bD + c)u^p = f$ and the integral conditions $\mathcal{T}_0(u^p) = \mathcal{T}_1(u^p) = 0$.

The first homogeneous solution satisfies $(D^2 + bD + c)\bar{u}^1 = 0$ and the integral conditions $\mathcal{T}_0(\bar{u}^1) = 1, \mathcal{T}_1(\bar{u}^1) = 0$. To find it, we set $\bar{u}^1 = 1/2 + u^*$. Then u^* satisfies the inhomogeneous equation (2.4) with $f \equiv -c/2$ and the integral conditions $\mathcal{T}_0(u^*) = \mathcal{T}_1(u^*) = 0$. The solution u^* is computed using the same pentadiagonal system used for u^p but with a different right hand side.

The second homogeneous solution $\bar{u}^2$ satisfies the integral conditions $\mathcal{T}_0(\bar{u}^2) = 0, \mathcal{T}_1(\bar{u}^2) = 1$. If we set $\bar{u}^2 = T_1 + u^*$, u^* satisfies the inhomogeneous equation with $f \equiv -(b + cT_1)$. It is found by solving the same pentadiagonal system.

The solution u of (2.4) is expressed as $u^p + C\bar{u}^1 + D\bar{u}^2$ and the constants C and D are determined using the boundary conditions on u.

2.2. Spectral integration of rth order

Define the operator L as $Lu = u^{(r)} + a_{r-1}u^{(r-1)} + \cdots + a_1u^{(1)} + a_0u$. Consider the inhomogeneous equation $Lu = f$. A particular solution satisfying the integral conditions

$$\mathcal{T}_0(u) = \cdots = \mathcal{T}_{r-1}(u) = 0$$

or equivalently $\alpha_0 = \cdots = \alpha_{r-1} = 0$ may be found as follows. Assuming constant coefficients, the inhomogeneous equation is written in an integral form as

$$u + a_{r-1}\int u + \cdots + a_0\int^r u + \sum_{j=0}^{r-1} A_j y^j = \int^r f.$$

Using formulas for $\int^j T_n$ we may express the coefficients of $T_r, \dots, T_{M-1}$ in terms of $\alpha_r, \dots, \alpha_{M-1}$. These coefficients are equated to the coefficients of $\int^r f$ and solved for $\alpha_r, \dots, \alpha_{M-1}$ to find the particular solution u^p. The linear system has $2r+1$ diagonals.

To find the jth homogeneous solution for $j = 1, \dots, r$, we first set $\bar{u}^j = T_{j-1} + u^*$. The jth homogeneous solution satisfies the conditions $\mathcal{T}_k(\bar{u}^j) = 0$, if $0 \le k \le r-1$ and $k \ne j-1$, and $\mathcal{T}_{j-1}(\bar{u}^j) = 1$. The function u^* satisfies $Lu^* = -LT_j$ and the first r coefficients in its Chebyshev series are zero. It can be found in the same manner as the particular solution.

The solution of the linear boundary value problem is expressed as $u^p + \sum_{j=1}^{r} C_j \bar{u}^j$. The boundary conditions satisfied by u are used to determine the constants C_j.

Formulas for $\int^j T_n$ can be derived but get complicated. The $2r+1$ diagonal system can be difficult to set up correctly in programs. For the difficulties that arise for $r = 4$, see [18]. Spectral integration of rth order will, however, prove quite useful in the discussion of cancellation errors in Section 3.1.

2.3. Greengard form of spectral integration

The formulation of the Greengard form of spectral integration given here is based on banded matrices without any dense bordered rows. In general, the matrix systems have $2r+1$ diagonals if the problem is of order r. We assume L to be the operator defined in Section 2.2.

The Greengard form begins by assuming a Chebyshev series for $u^{(r)}$. A similar method was proposed earlier by Zebib [19]. We first find a particular solution of $Lu = f$ subject to the integral conditions

$$\mathcal{T}_0(u) = \mathcal{T}_0(u^{(1)}) = \cdots = \mathcal{T}_0(u^{(r-1)}) = 0.$$

The integral conditions are different this time. The integral conditions given earlier ensure that if u satisfies the integral conditions and we know the Chebyshev series of u, then we can produce the Chebyshev series of $\int_s u$ for $s = 0, \ldots, r-1$. The integral conditions here ensure that if we know the Chebyshev series of $u^{(r)}$, then we can produce the Chebyshev series of $u^{(s)}$ for $s = 0, \ldots, r-1$ without ambiguity. When these conditions are used, the Chebyshev series of $u^{(s)}$ determines the Chebyshev series of $u^{(s-1)}$ for $s = r, \ldots, 1$ with no ambiguity. Normally, there is an undetermined constant of integration when the series of $u^{(s)}$ is integrated. But here the constant disappears because the mean mode of $u^{(s-1)}$ is specified to be zero. Thus the Chebyshev series of Lu is determined unambiguously by the Chebyshev series of $u^{(r)}$. Coefficients in the Chebyshev series of Lu and f are equated and solved for $\mathcal{T}_j\left(u^{(r)}\right)$ for $j = 0, \ldots, M$.

To find homogeneous solutions $\bar{u}$, we expand $\bar{u}^{(r)}$ in a Chebyshev series and take the integral conditions to be such that exactly one of $\mathcal{T}_0(\bar{u}), \ldots, \mathcal{T}_0(\bar{u}^{(r-1)})$ is one and the others are all zero. It is harder to find homogeneous solutions here than in the rth order spectral integration method described in Section 2.2. One has to find polynomials p_k of degree k for $k = 0, 1, \ldots, r-1$ such that $\mathcal{T}_0(p_k^{(k)}) = 1$ but $\mathcal{T}_0(p_k^{(d)}) = 0$ for $d = 0, \ldots, k-1$. However, this form of spectral integration generalizes more easily to linear differential equations with polynomial coefficients [14,8].

2.4. Factored form of spectral integration

A linear operator L with constant and real coefficients can be factorized as

$$L = (D - a_1) \ldots (D - a_m)(D^2 + b_1 D + c_1) \ldots (D^2 + b_n D + c_n)$$

where the coefficients are all real. We assume $m + n \geq 2$ and derive a method for solving $Lu = f$ subject to boundary conditions that exploits this factorization of L. This method relies on spectral integration of orders one and two described in Section 2.1. The presentation of the method may appear more complicated but its implementation is much simpler than the methods of Sections 2.2 and 2.3.

It is well-known that the numerical factorization of a polynomial can be quite inaccurate [20]. This method is applicable to only those situations where an accurate numerical factorization is available. It is applicable if the operator is known in the factored form to begin with. It may also be applied if the coefficients of the operator L are known exactly. If the coefficients are known exactly, a symbolic computing package supporting extended precision arithmetic may be used to obtain a factorization accurate to double precision, even if the factorization is ill-conditioned.

A particular solution is found by solving the following equations subject to integral conditions on their solutions:

$$\begin{aligned}
&(D - a_1)u^p_{m+2n-1} = f \qquad \mathcal{T}_0(u^p_{m+2n-1}) = 0\\
&(D - a_2)u^p_{m+2n-2} = u^p_{m+2n-1} \qquad \mathcal{T}_0(u^p_{m+2n-2}) = 0\\
&\vdots\\
&(D - a_m)u^p_{2n} = u^p_{2n+1} \qquad \mathcal{T}_0(u^p_{2n}) = 0\\
&(D^2 + b_1 D + c_1)u^p_{2n-2} = u^p_{2n} \qquad \mathcal{T}_0(u^p_{2n-2}) = \mathcal{T}_1(u^p_{2n-2}) = 0\\
&\vdots\\
&(D^2 + b_n D + c_n)u^p_0 = u^p_2 \qquad \mathcal{T}_0(u^p_0) = \mathcal{T}_1(u^p_0) = 0.
\end{aligned}$$

This list of equations is solved from first to last. Each equation is solved using one of the two methods described in Section 2.1. The subscripts on u, as in u^p_{2n}, indicate the number of "derivatives" in the function relative to u^p_0 which satisfies $Lu^p_0 = f$ and is therefore a particular solution.

If $m \geq 1$, the homogeneous solution $\bar{u}^h$ with $h = 1$ is found as follows. To begin with we solve the homogeneous problem

$$(D - a_1)\bar{u}^h_{n+2m-1} = 0 \qquad \mathcal{T}_0(\bar{u}^h_{n+2m-1}) = 1$$

as described in Section 2.1. Thereafter, the inhomogeneous problems

$$(D - a_j)\bar{u}^h_{n+2m-j} = \bar{u}^h_{n+2m-j+1} \qquad \mathcal{T}_0(\bar{u}^h_{n+2m-j}) = 0 \tag{2.5}$$

are solved in the order $j = 2, \ldots, m$ followed by the solution of

$$(D^2 + b_k D + c_k)\bar{u}^h_{2n-2k} = \bar{u}^h_{2n-2k+2} \qquad \mathcal{T}_0(\bar{u}^h_{2n-2k}) = \mathcal{T}_1(\bar{u}^h_{2n-2k}) = 0 \tag{2.6}$$

in the order $k = 1, \ldots, n$. The last solution to be found is $\bar{u}^h = \bar{u}^h_0$ and it satisfies $L\bar{u}^h = 0$. The inhomogeneous equations (2.5) and (2.6) are solved as described in Section 2.1.

More generally, the homogeneous solution $\bar{u}^h$ with $1 \leq h \leq m$ is solved beginning with the homogeneous problem

$$(D^2 - a_h)\bar{u}^h_{m+2n-h} = 0 \qquad \mathcal{T}_0(\bar{u}^h_{m+2n-h}) = 1$$

followed by the solution of (2.5) with $j = h + 1, \ldots, m$ and (2.6) with $k = 1, \ldots, n$. As before, $\bar{u}^h_0$ is the last solution to be found and $\bar{u}^h = \bar{u}^h_0$.

If $h = m + 2i - 1$ with $1 \leq i \leq n$, the homogeneous problem solved at the beginning is

$$(D^2 + b_i D + c_i)\bar{u}^h_{2n-2i} = 0 \qquad \mathcal{T}_0(\bar{u}^h_{2n-2i}) = 1, \qquad \mathcal{T}_1(\bar{u}^h_{2n-2i}) = 0.$$

This is followed by the solution of (2.6) with $k = i + 1, \ldots, n$. As before, $\bar{u}^h_0$ is the last solution to be found and $\bar{u}^h = \bar{u}^h_0$. On the other hand, if $h = m + 2i$ with $1 \leq i \leq n$, the homogeneous problem solved at the beginning is

$$(D^2 + b_i D + c_i)\bar{u}^h_{2n-2i} = 0 \qquad \mathcal{T}_0(\bar{u}^h_{2n-2i}) = 0, \qquad \mathcal{T}_1(\bar{u}^h_{2n-2i}) = 1.$$

This is followed by the solution of (2.6) with $k = i + 1, \ldots, n$. As before, $\bar{u}^h_0$ is the last solution to be found and $\bar{u}^h = \bar{u}^h_0$.

By using the methods of Section 2.1 repeatedly, we end up with a particular solution u^p and homogeneous solutions $\bar{u}^1, \ldots, \bar{u}^{m+2n}$. The solution of the boundary value problem $Lu = f$ is expressed as

$$u = u^p + \sum_{j=1}^{m+2n} C_j \bar{u}^j.$$

The constants C_j are found to fit the boundary conditions on u.

There are two ways to find C_j. In the first method, the particular solution u^p and the homogeneous solutions $\bar{u}^h$ are obtained in physical space as numerical values at the $M + 1$ points on the Chebyshev grid. Boundary conditions such as $u(1) = A$ or $u''(-1) = B$ are expressed using a linear combinations of function values at the grid point. A boundary condition such as $u(1) = A$ simply specifies the function value at a single grid point. A boundary condition such as $u''(-1)$ is interpreted as specifying that a certain linear combination of function values, the linear combination being determined by a single row of a spectral differentiation matrix, must have a specified value. This is the easier method for implementation and the one we have implemented. If the number M is not too large, this method will be adequate. If M is very large, then errors will creep in through the boundaries.

The second technique uses the intermediate objects created when the particular solution and the homogeneous solutions are found. We illustrate the technique using an example. Suppose the boundary value problem is

$$(D - a_1)(D - a_2)(D - a_3)(D - a_4)u = f$$

subject to $u(\pm 1) = u'(\pm 1) = 0$. If $u = u^p + \sum_{j=1}^{4} C_j \bar{u}^j$, the conditions on $u(\pm 1)$ give two equations for the C_j after evaluation at ±1. We may rewrite the other boundary conditions as $(D - a_4)u = 0$ at ±1. If we now note that

$$(D - a_4)u = u^p_1 + \sum_{j=1}^{3} C_j \bar{u}^j_1$$

we get two more equations for the C_j by evaluating at ±1. In the sum above, the $j = 4$ term does not appear. That is because the homogeneous solution $\bar{u}^4$ satisfies $(D - a_4)\bar{u}^4 = 0$.

2.5. Spectral integration with piecewise Chebyshev grid

To generalize spectral integration to piecewise Chebyshev grids, we consider the operator $(D-a)(D-b)$ over the interval $-1 \leq y \leq 1$ and the boundary value problem corresponding to $(D-a)(D-b)u = f$. As earlier in this section, the boundary conditions enter only at the end and much of the method is independent of the specific form of the boundary conditions. The generalization to operators of the form $(D - a_1) \ldots (D - a_m)$ will be obvious.

Let $[-1, 1] = \mathcal{I}_1 \cup \cdots \cup \mathcal{I}_n$, where $\mathcal{I}_i$ are intervals with disjoint interiors and with the right end point of $\mathcal{I}_i$ equal to the left end point of $\mathcal{I}_{i+1}$. Thus $\mathcal{I}_1, \ldots, \mathcal{I}_n$ are disjoint intervals arranged in order. Let w_i denote the width of the interval $\mathcal{I}_i$.

We use a linear change of variables $\mathcal{I}_i \to [-1, 1]$ and rewrite the given differential equation as

$$\left(D - \frac{aw_i}{2}\right)\left(D - \frac{bw_i}{2}\right) u = \frac{w_i^2}{2} f \tag{2.7}$$

after the change of variables. In (2.7) it is assumed that u and f have been shifted from $\mathcal{I}_i$ to $[-1, 1]$ although that is not indicated explicitly by the notation.

We define u_i as

$$u_i = u^p + \alpha_{\mathcal{I}_i}\bar{u}^1 + \beta_{\mathcal{I}_i}\bar{u}^2, \tag{2.8}$$

where u^p is the particular solution and $\bar{u}^1$, $\bar{u}^2$ are the homogeneous solutions of (2.7), computed as described in Section 3.

For $i = 1, \ldots, n$, the coefficients $\alpha_{\mathcal{I}_i}$ and $\beta_{\mathcal{I}_i}$ comprise $2n$ unknown variables in total. We will solve for these unknowns using the two boundary conditions and continuity conditions between intervals. The boundary conditions give two equations such as

$$u_1(-1) = \text{left value}$$
$$u_n(1) = \text{right value}.$$

For $i = 1, \ldots, n-1$, the continuity conditions are

$$u_i(1) = u_{i+1}(-1)$$
$$\frac{((D - w_i b/2)u_i)\,(1)}{w_i} = \frac{((D - w_{i+1}b/2)u_{i+1})\,(-1)}{w_{i+1}}.$$

The second continuity condition requires the derivatives to be continuous while accounting for the shifting and scaling of intervals of width w_i and w_{i+1} to $[-1, 1]$. The function $(D - w_i b/2)u_i$ is available through the intermediate quantities generated by the method of Section 3. In particular, we have,

$$(D - w_i b/2)u^p = u_1^p, \qquad (D - w_i b/2)\bar{u}^1 = \bar{u}_1^1, \qquad (D - w_i b/2)\bar{u}^2 = 0$$

in interval $\mathcal{I}_i$. Once we solve for $\alpha_{\mathcal{I}_i}$ and $\beta_{\mathcal{I}_i}$ for $i = 1, \ldots, n$, we may use (2.8) to form u_i. The matrix system for computing the coefficients $\alpha_{\mathcal{I}_i}$ and $\beta_{\mathcal{I}_i}$ is banded. The solution u is obtained by shifting the u_i from $[-1, 1]$ back to $\mathcal{I}_i$.

3. Properties of spectral integration

In the first part of this section, we explain that the peculiar way in which homogeneous solutions are computed in all the methods of Section 2 ensure cancellation of errors. It is important to note that the cancellation error that occurs is the cancellation of discretization errors. All our computations show this to be quite robust. If versions of spectral integration that use matrices bordered by dense rows are employed, the same cancellation will take place implicitly during the solution of the linear systems.

3.1. Cancellation of intermediate errors

The solution of the linear boundary value problem $(D^2 - a^2)u = -(\pi^2 + a^2)\sin \pi y$ with boundary conditions $u(\pm 1) = 0$ is $u = \sin \pi y$. The solution can be represented with machine precision on a Chebyshev grid that uses slightly more than 20 points. If $a = 10^6$ it will take a Chebyshev grid with $M > 2 \times 10^4$ points to resolve Green's function at the boundaries. Spectral integration can solve this boundary value problem using a Chebyshev grid with 20 or 30 points even if $a = 10^6$. In this section, we illustrate and explain this property.

A smooth particular solution of $(D^2 - a^2)u = f$ for $y \in [-1, 1]$ is given by the integral $\int_{-1}^{1} \exp(-a|t - y|) f(t)\, dt$. This integral is amenable to Fourier analysis [21]. However, it does not satisfy the sort of boundary conditions in terms of Chebyshev coefficients that we impose in Section 2 and is therefore not the same as the particular solution that arises in Section 2.1.

Fig. 3.1 shows a homogeneous solution, the particular solution, and the computed answer when the method of Section 2.2 is applied to solve the boundary value problem

$$\left(D^2 - \alpha^2\right)\left(D^2 - \beta^2\right) u = f,$$

with

$$f = -8\pi^4 \cos 2\pi y - 2(\alpha^2 + \beta^2)\pi^2 \cos 2\pi y + \alpha^2\beta^2 \sin^2 \pi y. \tag{3.1}$$

The exact solution is $u = \sin^2 \pi y$. The parameters employed were $\alpha = 10^3$ and $\beta = 10^6$. Since M was only 32, there was no chance that the grid could resolve a boundary layer of thickness about 10^{-6} at $y = \pm 1$ that occurs in the particular solution u_p. Accordingly, the figure shows both u_p and one of the homogeneous solutions h_3 to be highly inaccurate. Yet the computed solution for u has 14/15 digits of accuracy.

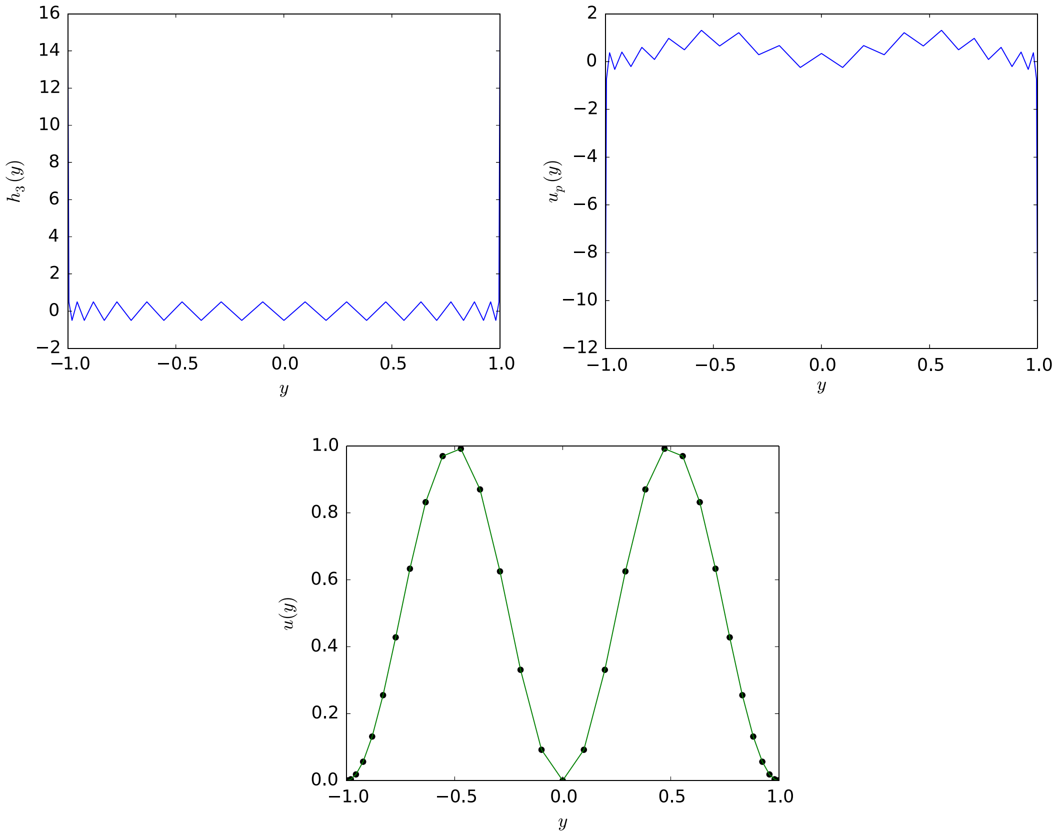

Fig. 3.1. The top two plots show that the homogeneous solution $h_3(y)$ and the particular solution $u_p(y)$ are highly inaccurate. Yet when the homogeneous solutions are combined with u_p, the solution $u(y) = \sin^2 \pi y$ is computed with 14/15 digits of accuracy. The lower plot shows agreement of the computed solution with the exact solution (black markers).

If x is the solution of the matrix system $Ax = b_1 + b_2$ and x_1, x_2 are the solutions of $Ax_1 = b_1, Ax_2 = b_2$, then $x = x_1 + x_2$ if A is nonsingular. In machine arithmetic and in the presence of rounding errors, this linear superposition property will be true only approximately. This section deals with discretization errors and not rounding errors. Therefore we will assume this linear superposition property.

Suppose that $Lu = f$ is the given equation. With given boundary conditions on u, this equation is assumed to have a solution that is well-resolved using $M + 1$ Chebyshev points. In all forms of spectral integration, a particular solution satisfying $Lu^p = f$ is found using some other global conditions on u^p. Typically it will take many more points than M to resolve u^p. Thus the computed u^p will be inaccurate. However, the approximation to u obtained by combining u^p with the homogeneous solutions will retain its accuracy for reasons we will now explain.

The explanation takes its simplest form for rth order spectral integration described in Section 2.2 and it is with that method that we begin. Suppose L is a linear differential operator with constant coefficients and order r as in Section 2.2. We begin by denoting the *computed* solution of

$$Lu = LT_j$$

with integral conditions $\mathcal{T}_0(u) = \cdots = \mathcal{T}_{r-1}(u) = 0$ by U_j for $j = 0, \ldots, r-1$. Thus the Chebyshev series of U_j is obtained by solving a banded system with $2r + 1$ diagonals and a right hand side that corresponds to the Chebyshev series of LT_j integrated r times. The U_j will be typically quite inaccurate. We will show that the U_j occur in the particular solution and the homogeneous solutions in such a way that they cancel when an approximation to the solution of $Lu = f$ with the given boundary conditions is computed.

Let u_E be the solution of $Lu_E = f$ which satisfies the given boundary conditions and is accurate to machine precision with a Chebyshev series of M terms. We rewrite u_E as

$$u_E = \frac{\alpha_0}{2} + \alpha_1 T_1 + \cdots + \alpha_{r-1} T_{r-1} + u_R$$

where $\mathcal{T}_j(u_R) = 0$ for $j < r$. We may rewrite f as

$$f = \frac{\alpha_0}{2} LT_0 + \cdots + \alpha_{r-1} LT_{r-1} + f_R$$

where $f_R = Lu_R$.

The particular solution of $Lu = f_R$ which is computed by rth order spectral integration is $u^p = u_R$. This is because u_R satisfies the integral boundary conditions, the first r of its Chebyshev coefficients being zero, as well as $Lu = f_R$ and can be represented to machine precision using a Chebyshev series of M terms. By linear superposition, the particular solution of $Lu = f$ satisfying integral boundary conditions that is computed is given by

$$u^p = \frac{\alpha_0}{2}U_0 + \alpha_1 U_1 + \cdots + \alpha_{r-1}U_{r-1} + u_R. \tag{3.2}$$

Homogeneous solutions of $Lu = 0$ are computed such that $\mathcal{T}_j(u) = 1$ but with the other $r-1$ Chebyshev coefficients among the first r coefficients being zero. This homogeneous solution is represented as $u = T_j + u^*$ and u^* is computed as the particular solution of $Lu^* = -LT_j$, whose first r coefficients are zero. Therefore the computed homogeneous solutions are

$$\bar{u}^1 = 1/2 - U_0/2, \qquad \bar{u}^2 = T_1 - U_1, \ldots, \bar{u}^r = T_{r-1} - U_{r-1}. \tag{3.3}$$

By observing (3.2) and (3.3), we recognize that

$$u_E = u^p + \frac{\alpha_0}{2}\bar{u}^1 + \alpha_1\bar{u}^2 + \cdots + \alpha_{r-1}\bar{u}^r.$$

In this linear combination of the particular solution with the homogeneous solutions, the coefficients are such that the inaccurate U_j cancel exactly and the solution u_E satisfies the given boundary conditions. If the equations that are solved to determine the linear combination of homogeneous solutions with the particular solution are reasonably well-conditioned, which we may expect because these are typically very small linear systems, the computed solution will produce u_E very accurately.

The argument above implies the following statement: *Let L be a linear differential operator with constant coefficients and of order r. Suppose that the linear boundary value $Lu = f$, with boundary conditions imposed possibly at the endpoints ±1, has a unique solution whose Chebyshev series $\sum' \alpha_i T_i(y)$ satisfies $\alpha_i = 0$ for $i \geq M$. In the absence of rounding errors, the method of* Section 2.2 *produces the solution u exactly*.

Cancellation of discretization errors is a property of many numerical methods. For example, in higher-order Runge–Kutta methods discretization errors in intermediate stages are canceled off in the final stage.

The explanations for the factored form of spectral integration and the Zebib–Greengard version are more complicated. We will give the explanation for the problem $(D-a)(D-b)u = f$. The given boundary conditions are assumed to be $u(\pm 1) = 0$. We assume as before that u_E is the approximate solution whose Chebyshev series has M terms and which is accurate to machine precision.

Suppose U_2 is the solution of $(D-b)u = 1$ satisfying $\mathcal{T}_0(U_2) = 0$ computed as explained in Section 2.1 using a Chebyshev series with M terms. Similarly, let U_1' be the computed solution of $(D-a)u = 1$ satisfying $\mathcal{T}_0(U_1') = 0$, and let U_1 be the particular solution of $(D-b)u = U_1'$ satisfying $\mathcal{T}_0(U_1) = 0$ and computed using a Chebyshev series with M terms only. For reasons given above, U_1 and U_2 are typically very inaccurate.

As before, we will split u_E but the split is more complicated this time. We write

$$u_E = \frac{\alpha_0}{2} + (\alpha_1 - \gamma)T_1 + u_R,$$

where γ is chosen such that

$$u_R = \gamma T_1 + \alpha_2 T_2 + \cdots + \alpha_{M-1}T_{M-1}$$

satisfies $(D-b)u_R = 0$. By applying $(D-a)(D-b)$ to u_E, f can be split as

$$f = \frac{ab\alpha_0}{2} - (a+b)(\alpha_1 - \gamma) + ab(\alpha_1 - \gamma)T_1 + (D-a)(D-b)u_R.$$

The computed particular solution that corresponds to $(D-a)(D-b)u = ab\alpha_0/2 - (a+b)(\alpha_1-\gamma)$ is $(ab\alpha_0/2 - (a+b)(\alpha_1-\gamma))U_1$. The particular solution of

$$(D-a)(D-b)u = ab(\alpha_1 - \gamma)T_1 \tag{3.4}$$

is obtained by solving

$$\begin{aligned}(D-a)v &= ab(\alpha_1-\gamma)T_1 = (D-a)\left(-b(\alpha_1-\gamma)T_1\right) + b(\alpha_1-\gamma)\\ (D-b)u &= v.\end{aligned}$$

Because of the way the right hand side of the $(D-a)v$ equation is rewritten, the particular solution of (3.4) may be taken to be computed as the particular solution of

$$(D-b)u = -b(\alpha_1-\gamma)T_1 + b(\alpha_1-\gamma)U_1' = (D-b)(\alpha_1-\gamma)T_1 - (\alpha_1-\gamma) + b(\alpha_1-\gamma)U_1'.$$

From the form of the right hand side, we infer that the particular solution of (3.4) is computed to be

$$(\alpha_1-\gamma)T_1 - (\alpha_1-\gamma)U_2 + b(\alpha_1-\gamma)U_1.$$

Table 3.1
Table of errors and condition numbers in the solution of $\left(D^2 - a^2\right) u = f$ with $a = 10^6$ and $f = -(\pi^2 + a^2) \sin \pi y$. The error is the infinite norm error in the computed u. The last two columns give the standard condition number of the spectral integration matrix and Bauer's spectral radius.

M	Error	Cond	Bauer
16	5.5e−16	3.8e2	1.9e1
32	1.6e−15	5.3e3	7.3e1
128	2.9e−15	1.2e6	1.1e3
1024	1.1e−13	4.8e9	7.0e4
4096	2.5e−13	1.5e11	2.5e5

Because of the way f was split,

$$\begin{aligned} u^p &= (ab\alpha_0/2 - (a+b)(\alpha_1 - \gamma))\, U_1 + (\alpha_1 - \gamma)T_1 - (\alpha_1 - \gamma)U_2 + b(\alpha_1 - \gamma)U_1 + u_R \\ &= a\,(b\alpha_0/2 - \alpha_1 + \gamma)\, U_1 - (\alpha_1 - \gamma)U_2 + (\alpha_1 - \gamma)T_1 + u_R \end{aligned} \tag{3.5}$$

is the particular solution of $(D - a)(D - b)u = f$ computed by the factored form of spectral integration.

The homogeneous solutions computed by the factored form of spectral integration are

$$\begin{aligned} \bar{u}^1 &= \frac{aU_1}{2} + \frac{U_2}{2} \\ \bar{u}^2 &= \frac{1}{2} + \frac{bU_2}{2}. \end{aligned} \tag{3.6}$$

By observing (3.5) and (3.6), we find that

$$u_E = u^p - 2\,(b\alpha_0/2 - \alpha_1 + \gamma)\,\bar{u}^1 + \alpha_0 \bar{u}^2.$$

We may argue as before that even though $u^p, \bar{u}^1, \bar{u}^2$ are inaccurate, the factored form of spectral integration solves $(D - a)(D - b)u = f$ with boundary conditions $u(\pm 1) = 0$ accurately.

3.2. *Condition numbers and accuracy*

Table 3.1 shows the errors in the solution of the linear system $\left(D^2 - a^2\right) u = f$ with $a = 10^6$ and $f = -(\pi^2 + a^2) \sin \pi y$. The errors are of the order of machine precision when $M = 16$ or $M = 32$ and grow only very slowly as M is increased. The version of spectral integration employed here was that of Section 2.2. However, the results are similar for the versions in Section 2.3 or 2.4.

Table 3.1 also shows that the 2-norm condition number of the spectral integration matrix is increasing rapidly. It does converge to a limit as $M \to \infty$ [14,6,7], but the limit is approximately $a^2 = 10^{12}$ (see the last columns of Tables 2 and 3 of [14] for another similar example). The 2-norm condition number here has nothing to do with the accuracy of the computed answer and the fact that it converges to a limit as $M \to \infty$ is of no consequence.

A more pertinent quantity is Bauer's spectral radius. It is known that

$$\min \kappa_\infty\,(D_1 A D_2) = \rho\left(|A|\,|A^{-1}|\right)$$

where the minimum is taken over all non-singular diagonal matrices D_1 and D_2, and $\rho(\cdot)$ is the spectral radius [22] [20, p. 127]. Bauer's spectral radius accounts for both row and column scaling. From Table 3.1, this quantity seems to converge approximately to a and not a^2 in the limit $M \to \infty$. The Green function corresponding to $\left(D^2 - a^2\right) u = f$ has a scale proportional to $1/a$ (see [9]). Therefore, even with row and column scaling we cannot expect a better condition number than $1/a$.

Although more pertinent, Bauer's spectral radius too fails to explain the accuracy of computed solution for large M in Table 3.1. The explanation appears to be that because the spectral integration matrix is banded and the Chebyshev coefficients of $u = \sin \pi y$ decay rapidly, it is as if only a section of the matrix corresponding to the lower coefficients is really active. Correspondingly, it may be noted that the singular vectors corresponding to the largest singular values are strongly localized within the lowest Chebyshev coefficients.

The situation in Table 3.1 is one extreme. The other extreme is shown in Table 3.2. The solution of the fourth order problem in the latter table develops boundary layers of size 10^{-6}. The 2-norm condition number for the linear systems is 10^{24} and is again totally irrelevant to the observed accuracy. In this latter table, we never get accuracy close to machine precision. The observed accuracy implies a loss of at least 6 digits. Because the solution develops boundary layers, the assumption that only the lowest few Chebyshev modes are active is no longer valid. The solution is of poor quality for $M = 1024$ because the grid fails to resolve the boundary layers.

The situation in turbulence simulations is in between the two scenarios. Turbulent solutions will not develop boundary layers or internal layers as thin as $\mathcal{O}\,(1/a)$. Thus we may summarize the discussion by saying that the unscaled condition numbers are of no relevance, that Bauer's spectral radius is more pertinent, and even that quantity may be unduly pessimistic.

Table 3.2
Infinite norm errors in the solution of $(D^2-a^2)(D^2-b^2)u = a^2b^2$ with $u(\pm 1) = u'(\pm 1) = 0$. The two errors correspond to spectral integration using the factorizations $(D-a)(D+a)(D-b)(D+b)$ and $(D^2-a^2)(D^2-b^2)$, respectively.

a	b	M	Error1	Error2
1e+06	2e+06	1 024	0.863351	0.863351
1e+06	2e+06	8 192	2.14342e−07	2.14697e−07
1e+06	2e+06	16 384	1.11927e−09	8.68444e−10
1e+06	2e+06	131 072	2.62727e−08	3.47769e−08

Table 4.1
Solution of $D^2 - aD = 0$ with $u(-1) = 1$, $u(1) = 2$, and $a = 10^6$ using a grid with three intervals, which are discretized using $M1+1$, $M2+1$, and $M3+1$ Chebyshev points, respectively. Nodes 1 and 4 are located at ±1. Node 2 is outside the boundary layer in the first two rows. The error is in the infinity norm.

$M1$	$M2$	$M3$	Node 2	Node 3	Error
16	1024	32	0.5	0.99999	5.80845e−06
16	4096	32	0.5	0.99999	4.07361e−11
32	128	32	0.999	0.99999	4.49718e−11
32	64	32	0.9999	0.99999	4.33247e−11
32	32	32	0.99995	0.99999	4.66069e−11

For an illustration of the points made so far in this section, we turn to a MATLAB boundary value solver that uses an integral formulation [23]. This MATLAB implementation does not use Chebyshev series but works exclusively in the physical domain using quadrature rules to discretize integral operators. While any of the spectral integration methods of Section 2 applied to $\left(D^2 - 10^{12}\right) u = -\left(\pi^2 + 10^{12}\right) \sin \pi y$, $u(\pm 1) = 0$ can find the solution $u = \sin \pi y$ with machine precision using $M = 32$, the physical space MATLAB implementation fails to do so. The solution computed in this MATLAB implementation lose more than 10 digits to numerical error. Here we see one advantage of working using Chebyshev coefficients instead of in physical space. The MATLAB implementation can handle the problem $\left(D^2 - a^2\right)\left(D^2 - b^2\right) u = f$, if a and b are both $\mathcal{O}(1)$ which is the simplest scenario.

As a point of comparison, we mention that while MATLAB [23] took 1.5523 billion cycles for a single solve of that system on a single core of 2.67 GHz Intel Xeon 5650, a somewhat casual C/C++ implementation of the method of Section 2.4 can do the same in 108,600 cycles. The more carefully optimized implementation used in Section 4.3 takes fewer than 50,000 cycles. Thus the C/C++ speed-up is at least 15,000.

4. Numerical examples and timed comparisons

In this section, we give numerical examples that illustrate the properties of spectral integration. The last part of this section includes a timed comparison with the Petrov–Galerkin method [10].

4.1. An example with a boundary layer

Table 4.1 summarizes spectral integration of piecewise Chebyshev grids applied to solve $(D^2 - aD)u = 0$ with $u(-1) = 1$, $u(1) = 2$, and $a = 10^6$. This problem develops a boundary layer at $y = 1$; see Fig. 4.1. It is evident from the table, that the intervals must be chosen carefully. The table shows that attempts to get an accurate solution with fewer than a thousand grid points and just a single interval properly contained in the boundary layer did not work. The last row of Table 4.1 reports a solution with $M_1 = M_2 = M_3 = 32$ and an error of 4.7×10^{-11}. In that computation, two intervals are contained inside the boundary layer. If a single Chebyshev grid is used, $M = 8192$ is needed to get more than ten digits of accuracy.

The example of Table 4.1 coincides with Example 3 of [13]. Table 7 of [13] reports an error of 2.33×10^{-11} using 20 intervals and $M = 16$ for each interval (the total number of grid points is 321). The last computation in Table 4.1 has slightly higher error but uses fewer grid points.

4.2. An example with an internal layer

The second example we consider is $\epsilon u'' + yu' = 0$ with boundary conditions $u(\pm 1) = \pm 1$ and $\epsilon = 10^{-12}$. This example too uses the piecewise Chebyshev grid. The exact solution of this boundary value problem is given by

$$u(y) = -1 + \frac{2\int_{-1/\sqrt{2\epsilon}}^{y/\sqrt{2\epsilon}} \mathrm{e}^{-t^2}\, dt}{\int_{-1/\sqrt{2\epsilon}}^{1/\sqrt{2\epsilon}} \mathrm{e}^{-t^2}\, dt}.$$

The solution has an internal layer at $y = 0$ of width approximately $\epsilon^{-1/2}$ or 10^{-6}. In Fig. 4.2, we show the spy plot of a matrix corresponding to division of $[-1, 1]$ into five sub-intervals as well as the transition region of the solution.

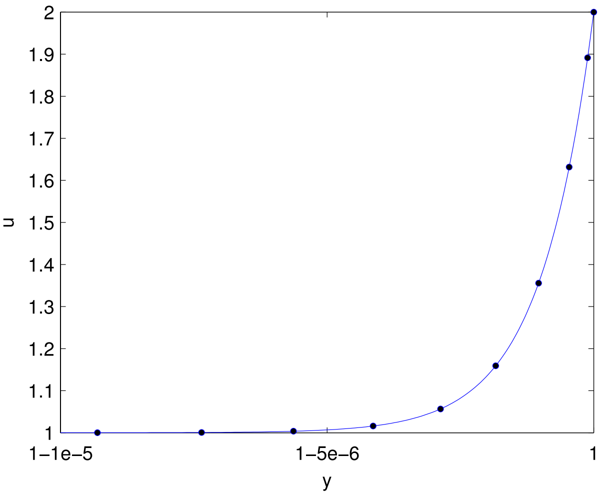

Fig. 4.1. Solution of $(D^2 - aD)u = 0$ with $a = 10^6$ and $u(1) = 2$ and $u(-1) = 1$. The figure shows the very thin boundary layer near $y = 1$. The computed solution (solid line) is in excellent agreement with the exact solution (filled markers).

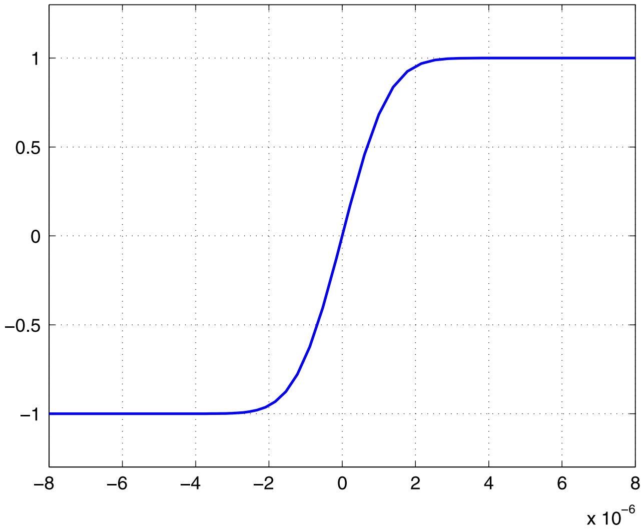

Fig. 4.2. Solution of $u'' + a\,y\,u' = 0$ with $u(-1) = -1,\ u(1) = 1$, and $a = 10^{12}$ using differentiation matrices and a piecewise Chebyshev grid. The plot shows the transition region of the solution.

Table 4.2
Solution of $\epsilon\, u'' + y\, u' = 0$ with $u(-1) = -1,\ u(1) = 1$, and $\epsilon = 10^{-12}$ is shown in Table 4.2. This table gives the overshoot beyond $[-1, 1]$ of the solution computed using 6 nodes and 5 intervals. Nodes 1, 2, 3, 5, and 6 are fixed at -1, $-8\epsilon^{1/2}$, $-3\epsilon^{1/2}$, $8\epsilon^{1/2}$, and 1, respectively. The number of Chebyshev points in each interval is $m + 1$.

m	Node 4	Overshoot
32	$5\sqrt{\epsilon}$	3.7e−15
32	$3\sqrt{\epsilon}$	1.2e−08
32	$7\sqrt{\epsilon}$	8.6e−09
24	$5\sqrt{\epsilon}$	1.8e−08

This second example occurs near the end of [14], where it is reported that mapped Chebyshev points with $M = 1024$ compute the solution with an overshoot of 3×10^{-4}. From Table 4.2, we see that the overshoot is reduced to the order of machine precision using only 161 grid points. The overshoot is seen to be highly sensitive to the location of the nodes. The solution plotted in Fig. 4.2(b) corresponds to the top row of the table. The solution appears to have around 10 digits of accuracy. Thus a more accurate solution is found using a fraction of the grid points.

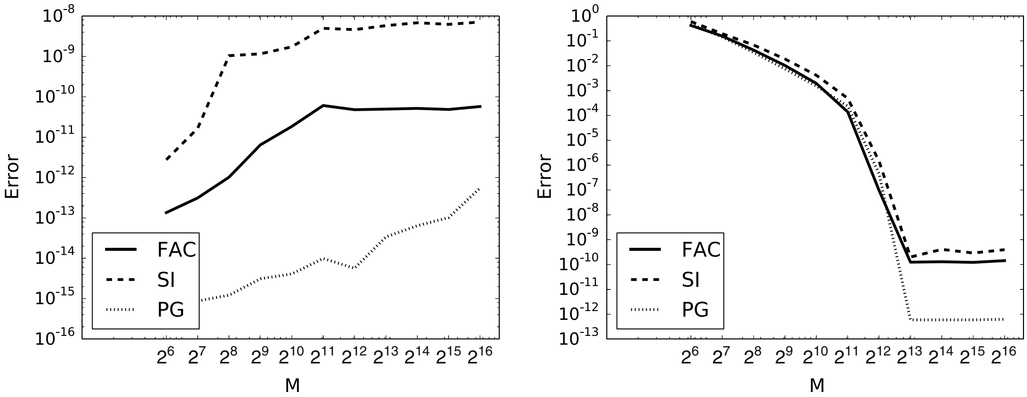

Fig. 4.3. The absolute error in three different methods for solving a 4th order boundary value problem: factored form of spectral integration from Section 2.4 (FAC), the spectral integration method of Section 2.2 (SI), and the Petrov–Galerkin method of Shen (PG). For the plot on the left, the exact solution is $\sin^2 \pi y$. The solution corresponding to the plot on the right develops thin boundary layers of size 10^{-6} at $y = \pm 1$. The errors shown are infinity norm errors.

4.3. Comparison with the Petrov–Galerkin method

We compare three different methods. Two of them are from Section 2 and a third is due to Shen [10]. The method of Shen is implemented using the fast Legendre transform of Alpert and Rokhlin [24]. The comparison is sufficient to bring out all the main points related to accuracy and efficiency that arise between different formulations of spectral integration.

The two problems considered in Fig. 4.3 both solve $(D^2 - \alpha^2)(D^2 - \beta^2)u = f$ with boundary conditions $u(\pm 1) = u'(\pm 1) = 0$. In both plots $\alpha = 10^3$ and $\beta = 10^6$. In the plot on the left, f is chosen as in (3.1) so that the exact solution is $u = \sin^2 \pi y$. The Petrov–Galerkin method has 2 more digits of accuracy than the factored form of spectral integration. The error in all three methods increases slowly as the grid is over-resolved. The errors were measured using grid sizes $M = 2^6, \ldots, 2^{16}$ as shown in the plots. The fast Legendre transform implementation requires M to be a power of 2 greater than 64.

In the plot on the right, $f = \alpha^{-2}\beta^{-2}$. The solution is $\mathcal{O}(1)$ in the middle of the domain but has thin boundary layers at the endpoints. Here too the Petrov–Galerkin method is seen to be more accurate by about 2 digits.

The impressive accuracy of the Petrov–Galerkin method could be because it uses linear combinations of Legendre polynomials which satisfy the boundary conditions exactly. In contrast, all versions of spectral integration rely on solving linear systems to enforce boundary conditions. Shen [11] has derived a spectral Galerkin method that uses Chebyshev polynomials and which similarly enforces boundary conditions by choosing an appropriate basis.

As a prelude to timing comparisons we describe the implementation of the three methods. The factored form of spectral integration was implemented using a second order solver. Linear systems were solved using the LAPACK routine `dgttrs` after decoupling even and odd modes. The method of Section 2.2 leads to pentadiagonal systems after the even and odd modes are decoupled. These systems were solved using the LAPACK routine `dgbtrs`. The LAPACK routines as well as the discrete cosine transforms were obtained by linking against the highly optimized Intel MKL library. For the Petrov–Galerkin method the symmetric positive define linear systems that arise after decoupling into even and odd modes were solved using `dpbtrs`. The fast Legendre transform is from [24], which is the best choice to the best of our knowledge.

All the computations were performed on a 16-core Intel Xeon E5-2660 machine clocked at 2.2 GHz. Timing measurements used the hardware timestamp counter accessed using the `RDTSC` instruction. The timestamp counter, properly used, provides accuracy at the level of processor cycles. All the timing measurements will be reported in cycles. The cycle is a unit of time intrinsic to the computer and is the right choice when different algorithms or implementations are compared on the same machine.

The size of the caches on modern computers is so large that a million grid point computation can fit comfortably in cache. Since our interest is in boundary value problems that arise as a part of large computations, such as turbulence simulations, which do not fit into cache, we made sure to eliminate cache effects. This was done by lining up a large number of problems in a single array and applying the solvers to them in succession, instead of repeatedly solving a problem that resides in the same location in memory. At least 16 GB of memory was accessed by each timing figure reported in a plot.

The plots in Fig. 4.4 report the timing figures as cycles per grid point. The cost of the boundary value solvers grows approximately linearly with the grid size M. Therefore cycles per grid point is a natural measure. The plot on the left is for single core runs. It shows that the spectral integration method of Section 2.2 uses nearly 75 more cycles per grid point than the factored form of spectral integration. The Petrov–Galerkin method uses more than 3 times as many cycles.

The principle reason the factored form of spectral integration is faster is because a single tridiagonal solve takes only around $37n$ cycles for a problem of size n. In contrast, a single pentadiagonal solve takes $120n$ cycles. Therefore using two tridiagonal solves in place of one pentadiagonal solve leads to an advantage of about 40 cycles per grid point. This computation clarifies the advantage of the factored form of spectral integration.

Writing optimized solvers requires consideration of register usage, cache locality, memory bandwidth, and other architectural features. Some of the principles that arise in such implementations are discussed in [25]. If a solver for banded

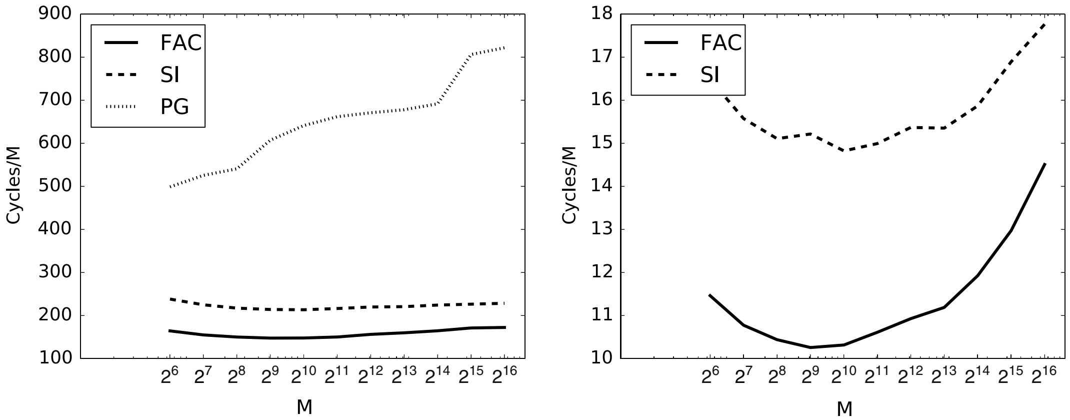

Fig. 4.4. Timing figures for three different boundary value solvers. FAC, SI, and PG are as in Fig. 4.3. The plot on the left is from a single core run. In the plot on the right, all 16 cores are used in parallel.

matrices with a few dense rows is implemented by hand, in may be as efficient as the factored form of spectral integration derived in Section 2.4 in terms of operation counts. However, producing an implementation that is as efficient as Intel MKL is a formidable problem. In addition, it is nearly impossible to keep up with constant changes in architecture. For a discussion of the solution of linear systems with a few dense rows using QR factorization, see [26].

For a more precise discussion, we note that the VADDPD and VMULPD 256-bit AVX2 instructions have latencies of 3 and 5 cycles, respectively, and a throughput equal to 1 cycle. Latency is the time the instruction spends in the execution pipeline and throughput is the inverse of the maximum rate at which instructions can be issued. In contrast, the latencies of the VDIVPD and VSQRTPD instructions are 33–55 cycles and 21–45 cycles, respectively. The throughputs are 44 and 28 cycles, respectively (Intel Architectures Optimization Reference Manual, 2014). These figures make it obvious that great care must be taken to favor some instructions over others.

Going back to Fig. 4.4, we explain why the Petrov–Galerkin method is slower by about a factor of 3. A discrete cosine transform with $n = 1024$ takes approximately $11n$ cycles. In contrast, a Legendre transform with $n = 1024$ takes $167n$ cycles. The additional cost of two Legendre transforms is responsible for most of the extra cycles consumed by the Petrov–Galerkin method.

The right side plot in Fig. 4.4 shows excellent speed-ups when all 16 cores solve boundary value problems in parallel. We omitted the Petrov–Galerkin solver for two reasons. Firstly, the fast Legendre transform program we used is not verified to be thread safe. Secondly, its memory requirements mean that a very different implementation must be produced for use in a multi-threaded environment. With $n = 1024$, the fast Legendre transform requires space of 0.6 GB to store pre-computed data. An effective multi-threaded implementation would share the precomputed data between threads.

The observed speed-up is greater than 10 but less than 16, which is the number of cores. Memory bandwidth does not scale linearly with the number of cores and it is theoretically impossible for a memory-limited program to achieve a speed-up equal to the number of cores. When such speed-ups are reported, it is usually a sign that the computation does not go out of cache.

5. Conclusions

In this article, we derived many different versions of spectral integration for solving linear boundary value problems. The treatment of boundary conditions is uncoupled from finding a particular solution in each of these versions. As a result, the various forms of spectral integration can be combined in many ways.

We showed that although the condition number of the spectral integration matrices converges to a constant as the grid size increases, the constant can be quite large. In Section 4.3, we gave timed comparisons and demonstrated that reducing boundary value solvers to the solution of tridiagonal matrices enables efficient implementation.

Acknowledgments

Many thanks to Bradley Alpert for sending me the Fortran source of his fast Legendre transform. The code for all of Section 4, which includes the Legendre transform, is posted at https://github.com/divakarvi/15-03-spectral. Thanks to both the reviewers for their detailed and valuable reports. Thanks to Fabian Waleffe, Hans Johnston, and Benson Muite for comments. This research was partially supported by NSF grants DMS-1115277 and SCREMS-1026317.

References

[1] S.A. Orszag, Galerkin approximations to flows within slabs, spheres, and cylinders, Phys. Rev. Lett. 26 (18) (1971) 1100–1103.
[2] S.A. Orszag, L.C. Kells, Transition to turbulence in plane Poiseuille and plane Couette flow, J. Fluid Mech. 96 (1980) 159–205.

[3] S.A. Orszag, A.T. Patera, Subcritical transition to turbulence in planar shear flows, in: R.E. Meyer (Ed.), Transition and Turbulence, Academic Press, 1981, pp. 127–146.

[4] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer-Verlag, 1993.

[5] D. Gottlieb, S.A. Orszag, Numerical Analysis of Spectral Methods: Theory and Applications, Society for Industrial and Applied Mathematics, 1977.

[6] L. Greengard, Spectral integration and two-point boundary value problems, SIAM J. Numer. Anal. 28 (1991) 1071–1080.

[7] V. Rokhlin, Solution of acoustic scattering problems by means of second kind integral equations, Wave Motion 5 (1983) 257–272.

[8] S. Olver, A. Townsend, A fast and well-conditioned spectral method, SIAM Rev. 55 (2013) 462–489.

[9] D. Viswanath, I. Tobasco, Navier–Stokes solver using Green's functions I: Channel flow and plane Couette flow, J. Comput. Phys. 251 (2013) 414–431.

[10] J. Shen, Efficient spectral-Galerkin method I: Direct solvers of second- and fourth-order equations using Legendre polynomials, SIAM J. Sci. Comput. 15 (1993) 1498–1505.

[11] J. Shen, Efficient spectral-Galerkin method II: Direct solvers of second- and fourth-order equations using Chebyshev polynomials, SIAM J. Sci. Comput. 16 (1995) 74–87.

[12] J. Shen, Efficient spectral-Galerkin methods IV: Spherial geometries, SIAM J. Sci. Comput. 20 (1999) 1438–1455.

[13] L. Greengard, V. Rokhlin, On the numerical solution of two-point boundary value problems, Comm. Pure Appl. Math. 44 (1991) 419–452.

[14] E.A. Coutsias, T. Hagstrom, D. Torres, An efficient spectral method for ordinary differential equations with rational function coefficients, Math. Comp. 65 (1996) 611–636.

[15] P.J. Diamessis, J.A. Domaradzki, J.S. Hesthaven, A spectral multidomain penalty method model for the simulation of high Reynolds number localized incompressible stratified turbulence, J. Comput. Phys. 205 (2005) 298–322.

[16] M. Charalambides, F. Waleffe, Spectrum of the Jacobi tau operator for the second derivative operator, SIAM J. Numer. Anal. 46 (2008) 280–294.

[17] M. Charalambides, F. Waleffe, Gegenbauer tau methods with and without spurious eigenvalues, SIAM J. Numer. Anal. 47 (2008) 48–68.

[18] B.K. Muite, A numerical comparison of Chebyshev methods for solving fourth order semilinear initial boundary value problems, J. Comput. Appl. Math. 234 (2) (2010) 317–342.

[19] A. Zebib, A Chebyshev method for the solution of boundary value problems, J. Comput. Phys. 53 (1984) 443–455.

[20] N.J. Higham, Accuracy and Stability of Numerical Algorithms, second ed., SIAM, Philadelphia, 2002.

[21] J.P. Boyd, Hyperasymptotics and the linear boundary layer problem: why asymptotic series diverge? SIAM Rev. 47 (2005) 553–575.

[22] F.L. Bauer, Optimally scaled matrices, Numer. Math. 5 (1963) 73–87.

[23] T.A. Driscoll, Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations, J. Comput. Phys. 229 (2010) 5980–5998.

[24] B.K. Alpert, V. Rokhlin, A fast algorithm for the evaluation of Legendre expansions, SIAM J. Sci. Stat. Comput. 12 (1991) 158–179.

[25] B. Sadiq, D. Viswanath, Finite difference weights, spectral differentiation, and superconvergence, Math. Comp. 83 (2014) 2403–2427.

[26] S. Olver, A. Townsend, A practical framework for infinite-dimensional linear algebra, 2014. arxiv.org, arxiv:1409.5529v1.