

Two-point boundary value problems, Green's functions, and product integration

G. D. McBain*

S. W. Armfield*

(Received 21 October 2004, revised 28 February 2005)

Abstract

When the Green's function for a two-point boundary value problem can be found, the solution for any forcing term reduces to a quadrature. Here we investigate using this as the basis for numerical schemes for boundary value problems and parabolic initial-boundary value problems. Application of Gaussian quadrature is disappointing, only converging slowly due to the discontinuous derivative of the Green's functions; however, rapid convergence is recovered by the use of product integration; that is, the prior accurate evaluation of Green's integral for a set of basis functions and the subsequent evaluation for arbitrary forcing functions by a matrix-vector multiplication. The method requires more operations than finite differencing for the same number of nodes, but converges far more rapidly. For small numbers

*School of Aerospace, Mechanical, & Mechatronic Engineering, The University of Sydney, AUSTRALIA. <mailto:geordie.mcbain@aeromech.usyd.edu.au> and <mailto:armfield@aeromech.usyd.edu.au>

See <http://anziamj.austms.org.au/V46/CTAC2004/Mcba> for this article, © Austral. Mathematical Soc. 2005. Published April 27, 2005. ISSN 1446-8735

of nodes, the performance is similar to that for orthogonal collocation; however, at high resolution, collocation accuracy deteriorates due to ill-conditioning whereas the present method is stable.

Contents

1	Introduction	C246
2	The method	C248
2.1	Evaluating Green's integral representation	C248
2.2	Product integration	C248
2.3	Application to Green's integral representation	C249
3	Examples	C251
3.1	Two-point boundary value problems	C251
3.1.1	A simple example	C251
3.1.2	An oscillatory problem	C251
3.2	Parabolic equations on a line segment	C254
4	Discussion and conclusion	C256
	References	C258

1 Introduction

We consider here the numerical solution of the two-point boundary value problem for ordinary differential equations. In particular, we are interested in problems where the ordinary differential operator and the boundary conditions are linear and fairly simple but the forcing function might be complicated and is perhaps only available through a discrete approximation, and moreover it is required to solve many problems with the same operator and

boundary conditions but different forcing functions. Such problems arise, for example, in the numerical solution of the Navier–Stokes equations for flow in an infinite channel [7].

As a simpler example,

$$\frac{d^2u}{dx^2} = -f, \quad u(\pm 1) = 0, \quad (1)$$

gives the steady temperature in an infinite slab with volumetric heat generation f and the sides held at a fixed temperature.

Green’s integral representation of the solution to a two-point boundary value problem with the associated boundary conditions is

$$u(x) = \int_{-1}^1 G(x, \xi) f(\xi) d\xi, \quad (2)$$

where $G(x, \xi)$ is the Green’s function. For example, the solution of (1) is given by (2) with

$$G(x, \xi) = \begin{cases} \frac{1}{2}(1-x)(1+\xi), & \xi \leq x, \\ \frac{1}{2}(1+x)(1-\xi), & x \leq \xi. \end{cases} \quad (3)$$

Green’s functions for arbitrary linear ordinary differential operators can be found by the method of variation of parameters; although it becomes increasingly difficult as the order increases. The Green’s functions for the two-point boundary value problems arising from the Navier–Stokes equation are given in [9]. We deal here only with boundary value problems for which the Green’s function has been found.

2 The method

2.1 Evaluating Green's integral representation

It may seem that once the Green's function is known, the two-point boundary value problem is solved; however, the integral (2) is still to be evaluated. If the forcing function f is only known through a discrete approximation—say by its n values at a predetermined set of abscissae—this can present difficulties. In particular, for a p th order differential equation, the Green's function has a discontinuity in its $(p-1)$ -th derivative along the line $\xi = x$, as exemplified by (3) for $p = 2$. This implies that a high-order quadrature rule such as Gauss's will only produce p th order accuracy; for the slab example (1), this was identified by Atkinson [2], who found that Gaussian quadrature actually produced poorer results than Simpson's rule.

Since the evaluation of Green's integral representation by a quadrature rule turns the integration into a matrix–vector multiplication, it requires $\mathcal{O}(n^2)$ floating-point operations for n abscissae. Thus, this approach is actually inferior to the simple application of second-order finite differences to the differential equation and the use of Thomas's algorithm to solve the resulting tridiagonal system. There is, however, a remedy, proposed by Atkinson [2] (but not, to our knowledge, implemented): the application of *product integration* to (2).

2.2 Product integration

A quadrature rule typically approximates an integral $\int_a^b f(\xi) d\xi$ by a weighted sum of the ordinates at certain abscissae

$$\sum_{k=1}^N w_k f(\xi_k). \quad (4)$$

Product integration [4, p. 87] approximates an integral like

$$\int_a^b \Phi(\xi) f(\xi) d\xi \quad (5)$$

by a rule of the same form (4), but where the weights w_k now depend on the factor $\Phi(\xi)$. Such a procedure is worthwhile when either many functions f must be integrated with the same factor Φ , or Φ lacks smoothness, or, as in the present case, both. If the function f is represented by an expansion

$$f(x) \doteq \sum_{j=1}^n \tilde{f}_j \phi_j(x), \quad (6)$$

the integral (5) becomes

$$\int_a^b \Phi(\xi) f(\xi) d\xi \doteq \sum_{j=1}^n \tilde{f}_j \int_a^b \Phi(\xi) \phi_j(\xi) d\xi. \quad (7)$$

2.3 Application to Green's integral representation

Thus Green's integral representation (2) is approximated as

$$u(x) \doteq \sum_{j=1}^N \tilde{f}_j \int_{-1}^1 G(x, \xi) \phi_j(\xi) d\xi, \quad (8)$$

and if we are content with the values of u at a certain set of nodes,

$$u_i \equiv u(x_i) \doteq \sum_{j=1}^n g_{ij} \tilde{f}_j, \quad (9)$$

where the *modified moments of the Green's function* are

$$g_{ij} \equiv \int_{-1}^1 G(x_i, \xi) \phi_j(\xi) d\xi. \quad (10)$$

We assume that the coefficients can be obtained by a linear transformation

$$\tilde{f}_j = \sum_{k=1}^n t_{jk} f(x_k) \equiv \sum_{k=1}^n t_{jk} f_k; \tag{11}$$

here we choose Chebyshev polynomials $\phi_j(x) = T_{j-1}(x)$ and

$$x_k = \cos \frac{(2k-1)\pi}{2n}, \tag{12}$$

so that (11) is just the discrete cosine transform. This could be computed with the Fast Fourier Transform, but it is more convenient to combine the two sets of coefficients with a matrix–matrix multiplication:

$$u_i = \sum_{j=1}^n g_{ij} \sum_{k=1}^n t_{jk} f_k = \sum_{k=1}^n \left(\sum_{j=1}^n g_{ij} t_{jk} \right) f_k = \sum_{k=1}^n \gamma_{ik} f_k. \tag{13}$$

Thus, provided we can evaluate the γ_{ij} (once and for all for a given Green’s function and n), the solution is accomplished in $\mathcal{O}(n^2)$ operations. If the expansion (6) is rapidly convergent (true for Chebyshev polynomials) and the modified moments (10) can be computed accurately, we should be able to achieve a highly accurate method for solving the two-point boundary value problem. This is the topic of the present paper.

Accurate evaluation of (10) is accomplished by splitting the integrals at the point $\xi = x_i$; that is, apply a quadrature rule to the two integrals in

$$g_{ij} = \int_{-1}^{x_i} G(x_i, \xi) \phi_j(\xi) \, d\xi + \int_{x_i}^1 G(x_i, \xi) \phi_j(\xi) \, d\xi. \tag{14}$$

In simple cases, it may be possible to evaluate the moments analytically, but in general numerical quadrature (say with n nodes in each subintegral) should suffice; here we use Fejér’s first rule [6]. Note that the device of splitting the integral (2) at the cusp $\xi = x$ is not so useful, since if f is known through a set of its nodal values, the subsets of these nodes lying in each subdomain are unlikely to lead to good quadrature rules.

3 Examples

3.1 Two-point boundary value problems

3.1.1 A simple example

As a first example, we take (1) with

$$f = \left(\frac{c}{2} \sec \frac{cx}{4} \right)^2, \quad (15)$$

where c is the root of $c = \sqrt{2} \cos \frac{c}{4}$ near 0.1336. This problem has the exact solution [8]

$$u = 2 \log \left(c \sec \frac{cx}{4} \right) - \log 2. \quad (16)$$

Direct application of a quadrature rule to the representation (2) with Green's function (3) is only slowly convergent, as expected. The results are plotted in Figure 1 for Gaussian quadrature and the midpoint rule; the two rules perform similarly, with linear convergence. This is, of course, worse than the quadratic convergence of central finite differences (also Figure 1).

Results are also plotted for orthogonal collocation [10, 11] and the present method. Both converge rapidly, achieving machine accuracy with $n \gtrsim 12$ abscissae. For finer meshes, the present method is stable with respect to rounding errors, but the collocation accuracy slowly deteriorates.

3.1.2 An oscillatory problem

A feature of most two-point boundary value problems encountered in practice is that a certain nodal density must be provided for the solution to be resolved; only with additional nodes is asymptotic convergence attained. This

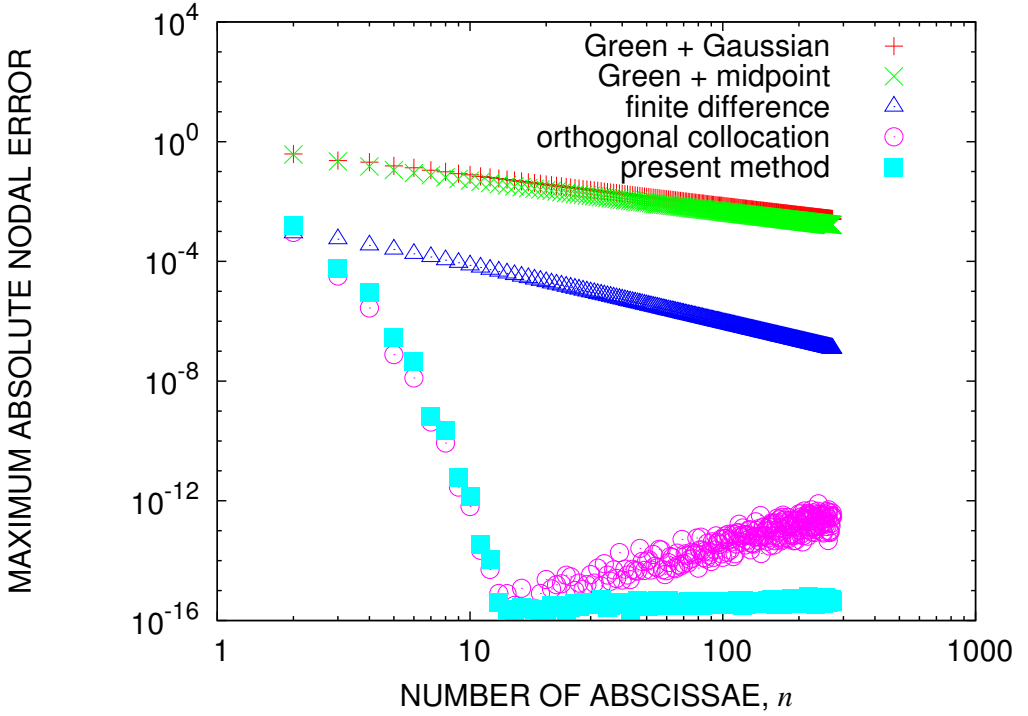


FIGURE 1: Convergence comparison for the two-point boundary value problem (1) with f given by (15).

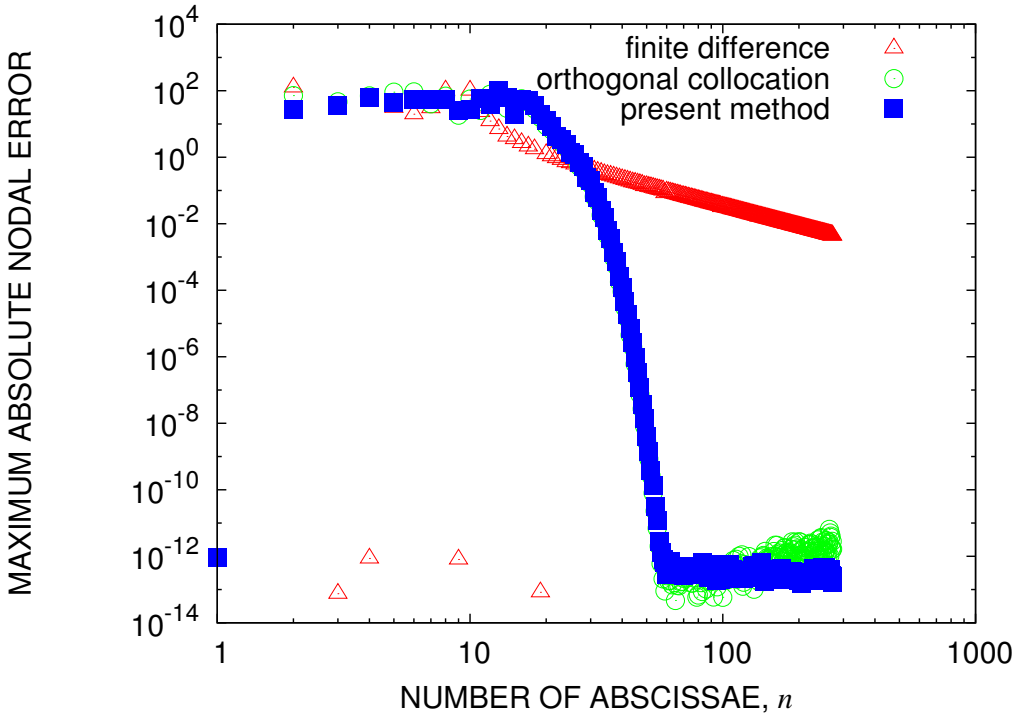


FIGURE 2: Convergence for the oscillatory problem (17).

is exemplified by problem (1) with the oscillatory forcing function

$$f = m^2 \pi^2 \sin m\pi x \quad (17)$$

for large integer m , say $m = 10$; the exact solution is $u = \sin m\pi x$. The results for three schemes, shown in Figure 2, are similar to the simple example, with the addition of an ‘unresolved’ regime for $n < 20$ for all schemes.

3.2 Parabolic equations on a line segment

Quasilinear parabolic partial differential equations on a line segment can be solved as a sequence of linear two-point boundary value problems of the type considered in this paper by using a mixed explicit–implicit time discretization; here we use second-order Adams–Bashforth/Crank–Nicolson, a simple choice popular in both high- [5, p. 119] and low-order [1] contexts. For example, the heat equation

$$\frac{\partial u}{\partial t} - \nu \frac{\partial^2 u}{\partial x^2} = \varphi, \quad u(\pm 1, t) = 0, \quad u(x, 0) = u_0(x), \quad (18)$$

becomes

$$\left(1 - \frac{\nu \Delta t}{2} \frac{d^2}{dx^2}\right) u_k(x) = f_k, \quad u_k(\pm 1) = 0, \quad (19)$$

where

$$f_k \equiv \left(1 + \frac{\nu \Delta t}{2} \frac{d^2}{dx^2}\right) u_{k-1} + \frac{\Delta t}{2} \{3\varphi_{k-1} - \varphi_{k-2}\}. \quad (20)$$

The Green’s function for (19) is

$$G(x, \xi) = \begin{cases} \frac{\sinh \lambda(1-x) \sinh \lambda(1+\xi)}{\lambda \sinh 2\lambda}, & \xi \leq x, \\ \frac{\sinh \lambda(1+x) \sinh \lambda(1-\xi)}{\lambda \sinh 2\lambda}, & x \leq \xi, \end{cases} \quad (21)$$

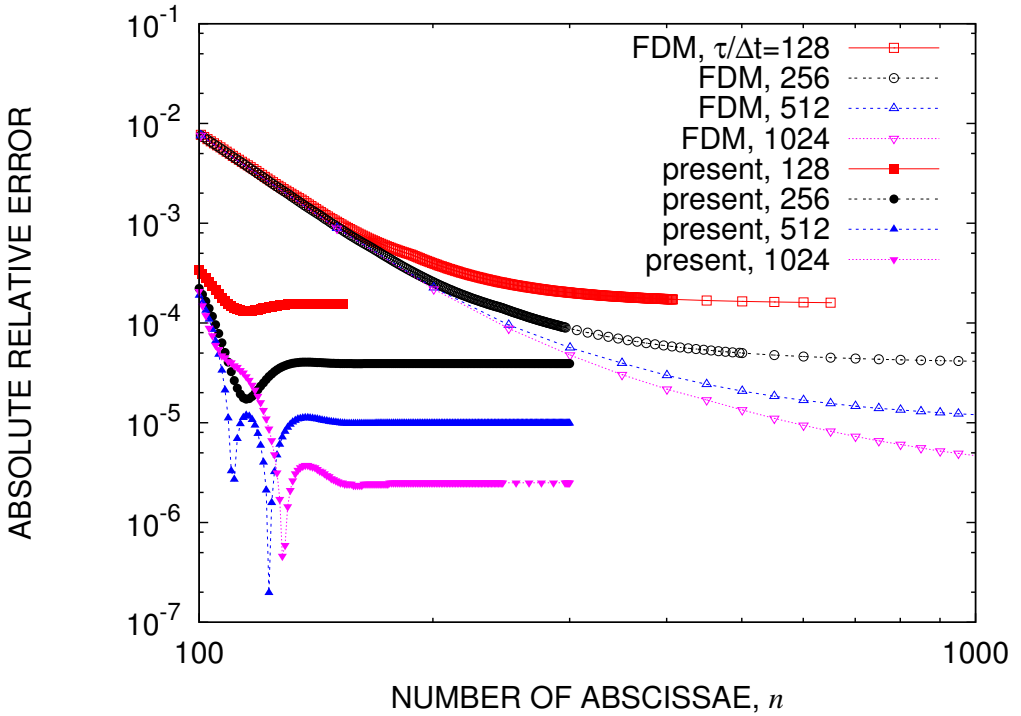


FIGURE 3: Convergence of finite difference and present methods for the Burgers problem. Time steps were fractions of $\tau = 1.6037/\pi$, as indicated.

$$= \begin{cases} \frac{e^{-\lambda(x-\xi)} - e^{-\lambda(2+x+\xi)} - e^{-\lambda(2-x-\xi)} - e^{-\lambda(4-x+\xi)}}{2\lambda(1 - e^{-4\lambda})}, & \xi \leq x, \\ \frac{e^{-\lambda(\xi-x)} - e^{-\lambda(2+\xi+x)} - e^{-\lambda(2-\xi-x)} - e^{-\lambda(4-\xi+x)}}{2\lambda(1 - e^{-4\lambda})}, & x \leq \xi, \end{cases} \quad (22)$$

where $\lambda^2 = 2/\nu\Delta t$. The latter form (22) is less prone to intermediate overflow errors for large λ ; that is, small time steps.

As an example, we treat Burgers equation; that is, (18) with

$$\varphi = -u \frac{\partial u}{\partial x}; \quad (23)$$

and the initial condition $u_0(x) = -\sin \pi x$. This has been studied in detail [3], and is useful because the exact solution is known. For small values of the viscosity ν , the initial peak and trough drift toward $x = 0$, forming a near-discontinuity before viscosity takes over and smooths the solution toward $u = 0$. In the case study [3], $\nu = 1/100\pi$, and the gradient at $x = 0$ reaches a minimum of -152.00516 at $t = 1.6037/\pi$. This quantity is a sensitive test for the accuracy of a parabolic scheme, and here we use it to compare the present method with a second-order finite difference scheme on a uniform grid (Figure 3). In both cases we exploit the odd-symmetry of the solution. And in the present method, the right-hand side derivatives are computed with pseudo-spectral differentiation matrices [11].

The present method converges rapidly to a level determined by the time-discretization error (which is evidently second-order, as expected). The finite difference scheme also appears to be doing this, but much more slowly.

4 Discussion and conclusion

The computational time required is often a more important measure of the cost of a computation than the number of abscissae. Here there is some competition between sparse low-order direct methods and the present dense high-order Green's function method. The present method requires a dense matrix-vector multiplication, and therefore $\mathcal{O}(n^2)$ floating point operations, whereas a central difference discretization results in a positive definite symmetric tridiagonal system than can be solved in $\mathcal{O}(n)$ operations. The question is whether the more rapid convergence of the spectral method can offset its more rapid growth in computational demands. The answer will depend on

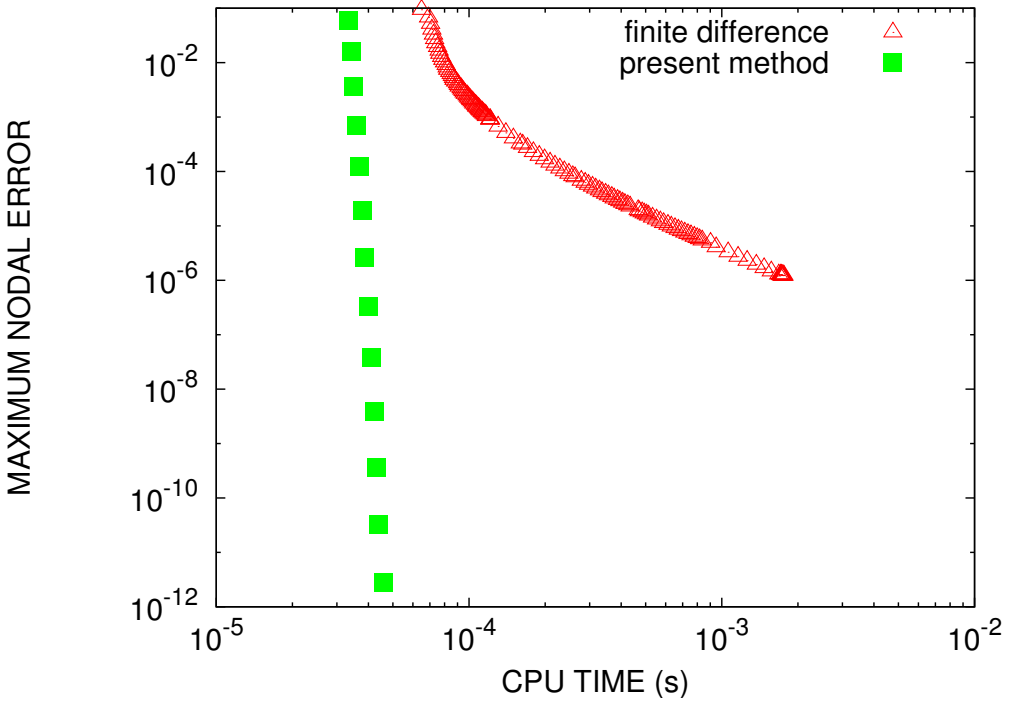


FIGURE 4: Computational time required for the oscillatory problem (17).

the problem and the accuracy required; here we plotted the error against computer time (excluding the time for initialization) for the oscillatory boundary value problem (Figure 4). The results show that for this problem the present method is faster than the finite difference method for any level of accuracy, with the advantage increasing for finer tolerances.

For a problem which requires many more nodes for resolution, one may struggle to provide sufficient nodes for the asymptotic regime to be entered. In this case, n will be limiting, and the finite difference scheme might be preferable. This could arise if the forcing terms are nearly discontinuous [7].

Also, the initialization of a finite difference scheme is trivial, whereas the present method is quite involved. If only a few problems are to be solved with the same linear operator, the present method is unlikely to be competitive with finite differences or orthogonal collocation.

References

- [1] S. Armfield and R. Street. Modified fractional-step methods for the Navier–Stokes equations. *ANZIAM J.*, 45(E):C364–C377, 2004.
<http://anziamj.austms.org.au/V45/CTAC2003/Armfield> C254
- [2] K. Atkinson. An automatic program for linear Fredholm integral equations of the second kind. *ACM Trans. Math. Software*, 2:154–171, 1976. <http://doi.acm.org/10.1145/355681.355686> C248
- [3] C. Basdevant, M. Deville, P. Haldenwang, J. M. Lacroix, J. Onazzoni, R. Peyret, P. Orlandi, and A. T. Patera. Spectral and finite difference solution of the Burgers equation. *Comput. & Fluids*, 14:23–41, 1986.
[http://dx.doi.org/10.1016/0045-7930\(86\)90036-8](http://dx.doi.org/10.1016/0045-7930(86)90036-8) C256

- [4] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic, New York, 2nd edition, 1984.
<http://www.ams.org/msnmain?fn=130&pg1=MR&s1=760629> C249
- [5] M. O. Deville, P. F. Fischer, and E. H. Mund. *High-Order Methods for Incompressible Fluid Flow*. Cambridge University Press, 2002.
<http://www.cambridge.org/uk/catalogue/catalogue.asp?isbn=0521453097> C254
- [6] L. Fejér. Mechanische Quadraturen mit positiven Cotesschen Zahlen. *Math. Z.*, 37:287–309, 1933. C250
- [7] R. P. Garg, J. H. Ferziger, and S. G. Monismith. Hybrid spectral finite difference simulations of stratified turbulent flows on distributed memory architectures. *Intl J. Numer. Methods Fluids*, 24:1129–1158, 1997. <http://www3.interscience.wiley.com/cgi-bin/abstract/61002258/> C247, C258
- [8] S. Kumar and I. H. Sloan. A new collocation-type method for Hammerstein integral equations. *Math. Comp.*, 48:585–593, 1987.
<http://www.ams.org/msnmain?fn=130&pg1=MR&s1=878692> C251
- [9] G. D. McBain. Poloidal–toroidal decomposition of doubly periodic vector fields. Part 2. The Stokes equations, 19 Mar. 2004. Submitted for publication in ANZIAM J. C247
- [10] J. Villadsen and M. L. Michelsen. *Solution of Differential Equation Models by Polynomial Approximation*. Prentice–Hall, 1978. C251
- [11] J. A. C. Weideman and S. C. Reddy. A MATLAB differentiation matrix suite. *ACM Trans. Math. Software*, 26:465–519, 2000.
<http://doi.acm.org/10.1145/365723.365727> C251, C256