Applied Numerical Mathematics 102 (2016) 17–30

Contents lists available at ScienceDirect

Applied Numerical Mathematics

www.elsevier.com/locate/apnum

The Chebyshev points of the first kind

Kuan Xu

Mathematical Institute, University of Oxford, Oxford OX2 6GG, UK

ARTICLE INFO

Article history:
Received 12 April 2015
Received in revised form 31 August 2015
Accepted 23 December 2015
Available online 11 January 2016

Keywords:
Chebyshev points
Chebyshev nodes
Approximation theory
Spectral methods

ABSTRACT

In the last thirty years, the Chebyshev points of the first kind have not been given as much attention for numerical applications as the second-kind ones. This survey summarizes theorems and algorithms for first-kind Chebyshev points with references to the existing literature. Benefits from using the first-kind Chebyshev points in various contexts are discussed.

1. Introduction

The Chebyshev polynomials of the first kind, $T_n(x) = \cos(n \arccos x)$, were introduced by Pafnuty Chebyshev in a paper on hinge mechanisms in 1853 [11]. The zeros and the extrema of these polynomials were investigated in 1859 in a paper by Chebyshev on best approximation [12]. The zeros of Chebyshev polynomials are called *Chebyshev points of the first kind*, *Chebyshev nodes*, or, more formally, *Chebyshev–Gauss points*; they are given by

$$x_k = \cos\theta_k, \theta_k = \frac{(2k+1)\pi}{2n}, \qquad k = 0, \ldots, n-1. \tag{1}$$

The extrema, given by

$$y_k = \cos\phi_k, \phi_k = \frac{k\pi}{n-1}, \qquad k = 0, 1, \ldots, n-1, \tag{2}$$

are called the *Chebyshev points of the second kind*, or *Chebyshev extreme points*, or *Chebyshev–Lobatto points*. Both sets of points are the projections onto the real axis of equally spaced points on the upper half of the unit circle that, if extended with a uniform spacing to the lower half of the unit circle, are symmetric about the real axis. The difference is that the first-kind Chebyshev grid excludes the boundary points ± 1, while they are present in the second-kind grid. It is not hard to see that polynomial interpolation at either kind of Chebyshev points is equivalent to trigonometric interpolation of an even function at evenly-spaced points on unit circle using a cosine series. For a graphical illustration of these points, see [28, p. 90] or [56, p. 171]. Both kinds of points have been useful in many areas of numerical analysis and scientific computing, such as function approximation and spectral methods.

E-mail address: kuan.xu@maths.ox.ac.uk.

http://dx.doi.org/10.1016/j.apnum.2015.12.002
0168-9274/

Table 1
References and Chebfun commands for fundamental operations on a Chebyshev grid.

	First kind `chebpts(n,1)`	Second kind `chebpts(n,2)`
Nested sampling 	 `chebtech1.refine`	Battles & Trefethen [2] `chebtech2.refine`
Aliasing formula	Fox & Parker [26, p. 67] Hayes [33, p. 21] Mason & Handscomb [44, p. 153] `chebtech1.alias`	Fox & Parker [26, p. 67] Hayes [33, p. 21] Trefethen [64, pp. 26–27] `chebtech2.alias`
Interpolation formulae	Fox & Parker [26, p. 32]	Fox & Parker [26, p. 32]
Conversion between values and coefficients via FFT	 `chebtech1.coeffs2vals`	Trefethen [62, p. 78] `chebtech2.coeffs2vals`
Cardinal function	Boyd [6, pp. 105, 571]	Boyd [6, pp. 105, 570]
Barycentric weights	Henrici [34, p. 249] `[∼,∼,v]=chebpts(n,1)`	M. Riesz [52] Salzer [55] `[∼,∼,v]=chebpts(n,2)`
Differentiation matrix	 `diffmat(n,'chebkind1')`	Gottlieb et al. [30, p. 15] Weideman & Reddy [66] `diffmat(n,'chebkind2')`
Quadrature rule and algorithms for weights	Fejér [24] Waldvogel [65] `[x,w]=chebpts(n,1)`	Clenshaw & Curtis [14] von Winckel [67] Waldvogel [65] `[x,w]=chebpts(n,2)`
Lebesgue constants	Ehlich & Zeller [20] McCabe [45] Rivlin [54] Günttner [32] Brutman [9] `lebesgue`	Bernstein [3] Erdős [23] McCabe [45] Brutman [8] `lebesgue`
Chebfun	Since 2014	Since 2004

If we look at popular textbooks in numerical analysis, it is interesting to note that a majority of the classic approximation textbooks introduce only the first-kind Chebyshev points but do not discuss them in depth [13,42,50,61], and the second-kind Chebyshev points are totally omitted. The new title by Trefethen [64] works with the second-kind points entirely and little attention is paid to the first-kind points. For spectral methods, the second-kind points are treated in all the classic monographs [6,10,25,35,48,57,62] with only some of them discussing the first-kind points as well [6,10,35,57]. If we look at these classics chronologically, we see that the second-kind points have been increasingly adopted and have gained dominant status in the last three decades.

In this survey, we summarize results for the first-kind Chebyshev points which pertain to function approximation and spectral methods and we offer a collection of pointers to relevant references. These results, though they can be found in the literature, are scattered over many papers, not all of which are well-known (see Table 1). Unlike the Chebyshev points of the second kind, which have been well surveyed in the spectral methods books mentioned above, it seems that no review of first-kind Chebyshev points has appeared to date. The aim of this paper is to fill this gap in the literature and also to highlight a few occasions where the use of the first-kind Chebyshev points may lead to particularly easy and efficient computations in function approximation and solution of differential equations. We do not claim that methods based on other sets of points, e.g. the Chebyshev points of the second kind or the Legendre points, are inapplicable or ineffective; however, in some circumstances, the first-kind Chebyshev points may be more convenient.

Our writing this survey is motivated in part by a desire to share some of the practical experience we gained with first-kind Chebyshev grids during our work on the new version 5 release of Chebfun [19]. Chebfun is an open-source software system for numerical computing with functions based on piecewise polynomial interpolation using Chebyshev grids. Traditionally, Chebfun has used second-kind grids for all of its computations. Version 5 is the first release that supports the first-kind grids as well (see Table 1).

2. Fundamentals

In this section, we review various fundamental results for first-kind Chebyshev points.

Fig. 1. Circles show Chebyshev points of the first kind for $n = 5$ and crosses for $n = 15$.

2.1. Chebyshev points of the first kind

Though they are most simply defined using the cosine function as in (1), in practice, the first-kind points are better computed using the sine-based formula

$$x_k = \sin\frac{(n-2k-1)\pi}{2n}, \qquad k = 0, \ldots, n-1. \tag{3}$$

One advantage of (3) over (1) is that (3) maintains exact symmetry of x_k about the origin in floating-point arithmetic, while (1) does not. Formula (3) also has an advantage of better relative accuracy, for example, in the construction of spectral differentiation matrices; see e.g. [16,66].

Just like their second-kind siblings, the Chebyshev points of the first kind are distributed in $[-1, 1]$ with the density $\dfrac{n}{\pi\sqrt{1-x^2}}$, which implies that the spacing between adjacent points is $O(n^{-2})$ near the endpoints. This property immunizes interpolants in first-kind Chebyshev grids from the famous *Runge phenomenon*.

2.2. Nested sampling on a Chebyshev grid of the first kind

It is well known that a Chebyshev grid of the second kind is nested in the second-kind Chebyshev grid with twice the number of points. Chebyshev grids of the first kind enjoy a similar nestedness. The difference is that we need to have the number of points tripled to see the nestedness, instead of doubled.

Consider the grid given by (1) and another Chebyshev grid of the first kind which triples the number of points:

$$x_l = \cos\frac{(2l+1)\pi}{6n}, \qquad l = 0, \ldots, 3n-1. \tag{4}$$

It is easy to see in Fig. 1 that the k-th point in (1) is the $(3k+1)$-th point in (4).

If we are approximating a function, this nestedness means that we do not have to re-evaluate the function at the nested points when we sample it on a grid three times finer than the current one. Tripling the number of the points and doing the nested sampling can save us one third of the function evaluations, each time we need to refine the sampling grid. In practice, whether or not these savings are worthwhile depends on programming language, computer, etc. and could be marginal, as the induced overhead may outweigh the gain.

2.3. Approximation by Chebyshev series on first-kind grids — interpolation, truncation, and aliasing

Suppose $f(x)$ is a Lipschitz continuous function on $[-1, 1]$ with the Chebyshev series

$$f(x) = \sum_{j=0}^{\infty} a_j T_j(x), \tag{5}$$

with the truncation or projection to degree $n-1$ being

$$\tilde{p}_{n-1}(x) = \sum_{j=0}^{n-1} a_j T_j(x),$$

and the polynomial interpolation in the n-point Chebyshev grid of the first kind being

$$p_{n-1}(x) = \sum_{j=0}^{n-1} c_j T_j(x). \tag{6}$$

We have the following theorems, which parallel similar results given in [64, Theorem 4.1] for Chebyshev points of the second kind. The first theorem gives the aliasing property of Chebyshev polynomials on a Chebyshev grid of the first kind.

Theorem 1. *For any $n \geq 1$ and $0 \leq m \leq n-1$, the following polynomials take the same values on the n-point Chebyshev grid of the first kind:*

$$T_m,\ -T_{2n-m},\ -T_{2n+m},\ T_{4n-m},\ T_{4n+m},\ -T_{6n-m},\ -T_{6n+m},\ \ldots.$$

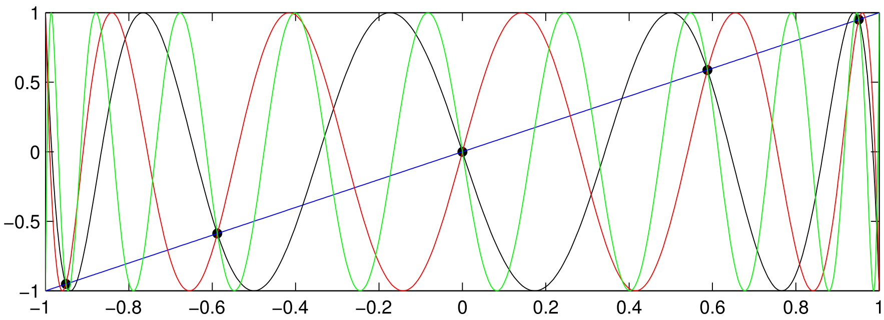

Fig. 2. Aliasing of T_1, $-T_9$, $-T_{11}$, and T_{19} on the 5-point Chebyshev grid of the first kind.

Proof. This is clear if we notice that

$$T_m(x_k) = \cos\left(\frac{(2k+1)m\pi}{2n}\right) = (-1)^r \cos\left[(2rn \pm m)\left(\frac{\pi(2k+1)}{2n}\right)\right] = (-1)^r T_{2rn\pm m}(x_k). \qquad \square$$

The second theorem identifies all Chebyshev polynomials T_j that take the same values (up to sign) as the m-th Chebyshev polynomial T_m on the n-point Chebyshev grid of the first kind.

Theorem 2. *For any $j \geq 0$, T_j takes the same values on an n-point Chebyshev grid of the first kind as $(-1)^p T_m$ with*

$$m = |(j+n-1)\,(\mathrm{mod}\ 2n) - (n-1)|\,, \tag{7}$$

a number in the range $0 \leq m \leq n-1$. Here, $p = \lfloor \frac{n+j}{2n} \rfloor$, where $\lfloor\cdot\rfloor$ denotes the floor function.

Proof. Suppose first that $0 \leq j\,(\mathrm{mod}\ 2n) \leq n$. Then $n-1 \leq (j+n-1)(\mathrm{mod}\ 2n) \leq 2n-1$, so (7) reduces to $m = j\,(\mathrm{mod}\ 2n)$, with $0 \leq m \leq n$, and it is shown by Theorem 1 that this implies that $(-1)^p T_j$ and T_m take the same values on the grid. On the other hand, suppose that $n+1 \leq j\,(\mathrm{mod}\ 2n) \leq 2n-1$. Then $0 \leq (j+n-1)(\mathrm{mod}\ 2n) \leq n-2$, so the absolute value becomes a negation and (7) reduces to $m = 2n - j\,(\mathrm{mod}\ 2n)$, with $1 \leq m \leq n-1$. Again Theorem 1 implies that $(-1)^p T_j$ and T_m take the same values on the grid. $\square$

Let us demonstrate the theorems above by taking a 5-point Chebyshev grid of the first kind and plotting T_1, $-T_9$, $-T_{11}$, and T_{19}. We can see in Fig. 2 that T_1, $-T_9$, $-T_{11}$, and T_{19} have identical values on the grid.

From the last two theorems, we give the following aliasing formula with proof omitted.

Corollary 1. *(See [26, p. 67] and [44, p. 153].) The coefficients of the interpolant p_{n-1} on an n-point Chebyshev grid of the first kind and the coefficients of the infinite Chebyshev series in (5) are related by*

$$c_j = a_j - (a_{2n-j} + a_{2n+j}) + (a_{4n-j} + a_{4n+j}) - \ldots \tag{8}$$

for $1 \leq j \leq n-1$, and

$$c_0 = a_0 - a_{2n} + a_{4n} - a_{6n} + a_{8n} - \ldots. \tag{9}$$

Remark 1. On an n-point Chebyshev grid of the first kind, T_{ln} for $l = 1, 3, 5, \ldots$ are "blind spots" of aliasing, as they are not present in (8) and (9). This happens because the values of T_n on the n-point first-kind Chebyshev grid are zeros (since these n points are by definition the zeros of T_n) and the zeros of $T_n, T_{3n}, T_{5n}, \ldots$ are nested.

The above aliasing formulae show how we can obtain the coefficients c_j of the degree n Chebyshev interpolant (6) by reassigning the coefficients a_j of the infinite series (5) to the corresponding aliases of degree 0 through $n-1$, i.e. $T_0, T_1, \ldots, T_{n-1}$. A matrix interpretation of Corollary 1 can be found in [68].

2.4. Discrete orthogonality on Chebyshev grid of the first kind

Chebyshev polynomials T_j are orthogonal on $[-1, 1]$ with respect to the weight function $w(x) = 1/\sqrt{1-x^2}$; see e.g. [53, p. 30]. That is,

$$\langle T_i, T_j \rangle = \begin{cases} 0 & i \neq j \\ \pi & i = j = 0 \\ \pi/2 & i = j > 0, \end{cases}$$

where the inner product is given by $\langle f, g \rangle = \int_{-1}^{1} w(x)\overline{f(x)}g(x)dx$ with the bar over $f(x)$ denoting complex conjugation.

The orthogonality also holds in a discrete sense on a Chebyshev grid of either kind. On an n-point first-kind grid, the discrete inner product leads to the discrete orthogonality as follows.

Theorem 3. *The Chebyshev polynomials* $\{T_j\}$ $j = 0, \ldots, n-1$ *satisfy discrete orthogonality on the n-point Chebyshev grid of the first kind. That is,*

$$\langle T_i, T_j \rangle = \begin{cases} 0 & i \neq j \text{ and } i, j \leq n-1 \\ n & i = j = 0 \\ n/2 & 0 < i = j \leq n-1 \end{cases},$$

where the discrete inner product is defined as

$$\langle u, v \rangle = \sum_{k=0}^{n-1} u(x_k) v(x_k)$$

with x_k given by (1).

Proof. See section 4.6.1 of [44]. □

We shall now show that this discrete orthogonality relation immediately leads to an explicit expression for the Chebyshev coefficients c_j in (6) in terms of function values at the n-point Chebyshev grid of the first kind by linking the conversion between Chebyshev coefficients c_j and function values to the discrete cosine transform (DCT).

2.5. Transform between function values and coefficients

Just as with Chebyshev points of the second kind, transforming from functions values on a first-kind grid to Chebyshev coefficients c_j in (6) can be accomplished by variants of the discrete cosine transform.

Values to coefficients. For $n \geq 1$, let $f_k = f(x_k)$ be the values of $f(x)$ on the first-kind Chebyshev grid of n points and let c_k be the coefficients in (6), and define the vectors

$$\mathbf{f} = (f_0, f_1, \ldots, f_{n-1})^T, \qquad \text{and} \qquad \mathbf{c} = (c_0, c_1, \ldots, c_{n-1})^T,$$

where the superscript T denotes transpose. Then c_j is related to f_k via [26, p. 30]

$$c_j = \frac{b}{n} \sum_{k=0}^{n-1} f_k T_j(x_k) \qquad j = 0, 1, \ldots, n-1, \tag{10}$$

where $b = 1$ for $j = 0$ and $b = 2$ otherwise. If $\mathbf{q} = (q_0, q_1, \ldots, q_{n-1})^T$, the *discrete cosine transform of type II* maps $\mathbf{q}$ to a vector $\mathbf{r} = (r_0, r_1, \ldots, r_{n-1})^T$ via

$$r_j = \sum_{k=0}^{n-1} q_k \cos\left[\frac{\pi j}{n}\left(k + \frac{1}{2}\right)\right], \qquad j = 0, 1, \ldots, n-1. \tag{11}$$

Writing this in matrix form as $\mathbf{r} = \mathbf{C}\mathbf{q}$, where $\mathbf{C}$ is defined by (11), we find that the Chebyshev coefficients (10) satisfy

$$\mathbf{c} = \frac{2}{n}\mathbf{C}\mathbf{f},$$

with the first entry of $\mathbf{c}$, i.e. c_0, further halved.

Coefficients to values. The function values can be evaluated in terms of the Chebyshev coefficients:

$$f_k = \sum_{j=0}^{n-1} c_j T_j(x_k) = \mathrm{Re} \sum_{j=0}^{n-1} c_j \exp\left(-\frac{j\pi(2k+1)i}{2n}\right).$$

The *inverse discrete cosine transform (IDCT) of type II* maps a vector $\mathbf{r}$ to another vector $\mathbf{q}$

$$q_j = \sum_{k=0}^{n-1} r_k \cos\left[\frac{\pi k}{n}\left(j + \frac{1}{2}\right)\right], \qquad j = 0, 1, \ldots, n-1; \tag{12}$$

or equivalently in matrix form $\mathbf{q} = \tilde{\mathbf{C}}\mathbf{r}$, where $\tilde{\mathbf{C}}$ is defined by (12), the function values on the n-point Chebyshev grid can be evaluated by

$$\mathbf{f} = \tilde{\mathbf{C}}\mathbf{c}.$$

In the absence of a specialized DCT code, these transforms can be implemented using the fast Fourier transform; see e.g. [43].

2.6. Cardinal functions

Given a set of n distinct points, $t_0, t_1, \ldots, t_{n-1}$, the k-th *cardinal function* or *Lagrange polynomial* defined by

$$\ell_k(x) = \prod_{\substack{j=0 \\ j \neq k}}^{n-1} \frac{x - t_j}{t_k - t_j}$$

is the unique polynomial of degree $n-1$ that vanishes at all points except t_k, at which it equals 1. If the points $t_0, t_1, \ldots, t_{n-1}$ are chosen to be the n-point Chebyshev grid of the first kind, the k-th cardinal function can be written neatly in terms of the Chebyshev polynomial $T_n(x)$ as

$$\ell_k(x) = \frac{T_n(x)}{(x - x_k) T_n'(x_k)}. \tag{13}$$

It follows that the linear combination

$$p(x) = \sum_{k=0}^{n-1} \ell_k(x) f(x_k), \tag{14}$$

is the unique polynomial of degree $n-1$ interpolating a function $f(x)$ in the given first-kind grid. Evaluating (14) at a single value of x requires $O(n^2)$ operations; however, this can be reduced to $O(n)$ by using the barycentric formula.

2.7. Barycentric interpolation formula

The *barycentric interpolation formula* has been widely adopted for polynomial interpolation since the publication of the survey paper by Berrut and Trefethen [4].

The formula comes in two canonical forms. When applied to the first-kind Chebyshev points (1), the first form of the formula reads

$$p_n(x) = \ell(x) \sum_{k=0}^{n-1} \frac{w_k^I}{x - x_k} f(x_k), \tag{15}$$

where $\ell(x)$ is the node polynomial $\ell(x) = (x - x_0)(x - x_1) \cdots (x - x_{n-1}) = T_n(x)/2^{n-1}$ and the barycentric weights are given by

$$w_k^I = (-1)^k \frac{2^{n-1}}{n} \sin\theta_k, \tag{16}$$

where θ_k is defined in (1).

Rewriting (15) in rational form, the *second (true) form of the barycentric interpolation formula* is given by

$$p_n(x) = \frac{\displaystyle\sum_{k=0}^{n-1} \frac{w_k^I}{x - x_k} f(x_k)}{\displaystyle\sum_{k=0}^{n-1} \frac{w_k^I}{x - x_k}} \tag{17}$$

with the same weights $\{w_k^I\}$ given by (16). However, removing the factors in (17) that are independent of the index k, we can obtain a simplified set of weights, which are scale-invariant:

$$w_k^{II} = (-1)^k \sin\theta_k.$$

Formula (17) continues to be valid with the simpler weights $\{w_k^{II}\}$.

2.8. Spectral differentiation matrix

To approximate the derivative of a function $f(x)$, we can differentiate its Chebyshev interpolant (6) directly, which yields an approximation to the derivative in the form of a Chebyshev series. An alternative is to apply a differentiation matrix to a vector of function values sampled at chosen points. Such matrices are fundamental building blocks in pseudospectral methods for differential equations. Denoting the values of $f(x)$ and its derivative on the n-point Chebyshev grid of the first kind by $\mathbf{f}$ and $\mathbf{u}$ respectively, we can relate $\mathbf{f}$ and $\mathbf{u}$ by the *spectral differentiation matrix* D_n:

$$\mathbf{u} = D_n \mathbf{f}$$

with the entries of D_n given by

$$D_{ij} = \begin{cases} \dfrac{x_i}{2(1-x_i^2)} & i = j \\ \dfrac{T_n'(x_i)}{(x_i - x_j)T_n'(x_j)} & i \neq j, \end{cases}$$

where x_j are the Chebyshev points given by (1). The entry D_{ij} can be obtained by differentiating the cardinal function $\ell_j(x)$ given by (13) and then evaluating at x_i. Like its counterpart for differentiation on a Chebyshev grid of the second kind, application of D_n suffers from numerical instability in the presence of rounding errors [7]. To mitigate the amplification of rounding errors, Baltensperger and Berrut [1] suggested that the above formula for the diagonal entries D_{ii} be replaced by the negated sum of all off-diagonal entries, that is,

$$D_{ii} = -\sum_{\substack{j=0 \\ j\neq i}}^{n} \frac{T_n'(x_i)}{(x_i - x_j)T_n'(x_j)}.$$

For many years, spectral differentiation matrices have been understood to be square. Recently, Driscoll and Hale [18] have introduced the concept of *rectangular spectral collocation*, in which the first-kind Chebyshev points play an important role. We defer the discussion of *rectangular spectral collocation* and *rectangular differentiation matrices* to section 3.3.

2.9. Quadrature weights on the Chebyshev grid of the first kind

Clenshaw–Curtis quadrature, based on sampling the integrand on a Chebyshev grid of the second kind, has comparable performance to Gauss quadrature but is easier to implement [63]. Analogously, this is also true of *Fejér's first rule*, which uses the values of the integrand on the Chebyshev grid of the first kind. Suppose we wish to compute

$$I = \int_{-1}^{1} w(x) f(x) dx, \tag{18}$$

where the integrand $f(x)$ is a continuous function on $[-1, 1]$ and the weight function $w(x)$ is positive and continuous on $(-1, 1)$. Fejér's first rule on the n-point Chebyshev grid of the first kind approximates I by

$$I_n = \sum_{k=0}^{n-1} w_k f(x_k), \tag{19}$$

where x_k are the first-kind Chebyshev points given by (1) and w_k are quadrature weights to be determined. Since Fejér quadrature is *polynomial interpolatory* [15, p. 84], that is, defined by integrating a polynomial interpolant of the integrand, we can derive the formula for the weights w_k as follows. On the one hand, we can start from (18) by substituting $f(x)$ by its interpolant given in (6) and swapping the summation and the integral to get

$$I_n = \sum_{j=0}^{n-1} c_j m_j, \tag{20}$$

where

$$m_j = \int_{-1}^{1} w(x) T_j(x) dx$$

is the j-th *moment* of the weight function $w(x)$. Collecting the Chebyshev coefficients in a column vector $\mathbf{c}$ as in section 2.5, we can write (20) as

$$I_n = \mathbf{m}^T \mathbf{c},$$

where $\mathbf{m}$ is a column vector with j-th entry m_j. Recognizing (10) as a linear transformation which we denote by

$$\mathbf{c} = \mathbf{T}\mathbf{f},$$

where $\mathbf{T}$ is the matrix that maps the function values at the Chebyshev grid of the first kind to the Chebyshev coefficients, we can write the quadrature rule as

$$I_n = \mathbf{f}^T (\mathbf{m}^T \mathbf{T})^T. \tag{21}$$

On the other hand, (19) can be written in vector notation as

$$I_n = \mathbf{f}^T \mathbf{w}, \tag{22}$$

where $\mathbf{w}$ is a column vector with the j-th entry being w_j. Comparing (21) and (22), both of which are valid for an arbitrary function, we have

$$\mathbf{w} = \mathbf{T}^T \mathbf{m}.$$

If we notice that $\mathbf{T}^T$ is the transform matrix from coefficients to function values discussed in section 2.5, up to a constant, the weights $\mathbf{w}$ can be computed in $O(n \log n)$ operations once the moments $\mathbf{m}$ are available. Sommariva [59] discusses the details of computing the weights $\mathbf{w}$ via FFT when the weight function $w(x)$ is of a general form. Here we look at a few special cases.

2.9.1. *Fejér's first rule with the Legendre weight function*

When the weight function in (18) is identically 1, the moments are as simple as

$$m_j = \int_{-1}^{1} T_j(x)dx = \begin{cases} 0 & j \text{ is odd} \\ \dfrac{2}{1-j^2} & j \text{ is even} \end{cases}$$

and the quadrature weights w_k for $k = 0, \ldots, n-1$ can be explicitly written as (see e.g. [15, p. 85]):

$$w_k = \frac{2}{n}\left(1 - 2\sum_{j=1}^{\lfloor n/2 \rfloor} \frac{\cos(2j\theta_k)}{4j^2 - 1}\right).$$

Here $\theta_k = (2k+1)\pi/2n, k = 0, \ldots, n-1$ are the arguments of the cosine function in (1).

Remark 2. Another common way to implement Fejér's first rule, which is slightly more straightforward, is to first figure out the Chebyshev coefficients $\mathbf{c}$ following the method in section 2.5 and then simply calculate the inner product of $\mathbf{c}$ and the moment vector $\mathbf{m}$, as in (20).

2.9.2. *Fejér's first rule with the Jacobi weight function*

When $w(x)$ is the Jacobi weight function, $w(x) = (1+x)^\alpha (1-x)^\beta$, the quadrature weights have no explicit form unless α and β have certain special values; however, we can compute the Jacobi moment vector $\mathbf{m}$ of dimension n with $m_j = \int_{-1}^{1}(1+x)^\alpha(1-x)^\beta T_j(x)dx$ cheaply in $O(n)$ operations. For $\alpha, \beta > -1$, the Jacobi moments can be expressed in terms of hypergeometric functions [49]:

$$m_j(\alpha, \beta) = 2^{\alpha+\beta+1} B(\alpha+1, \beta+1) s_j,$$

where $B(\cdot)$ is the beta function and s_j is a generalized hypergeometric function

$$s_j = {}_3F_2\left(\begin{matrix} j, -j, \beta+1 \\ \frac{1}{2}, \alpha+\beta+2 \end{matrix}; 1\right).$$

Each s_j can be calculated recursively using a three-term recurrence formula which is derived by using Sister Celine's technique [51]:

$$(\alpha+\beta+j+2)s_{j+1} + 2(\beta-\alpha)s_j + (\alpha+\beta-j+2)s_{j-1} = 0 \tag{23}$$

with initial values

$$s_0 = 1, \qquad s_1 = \frac{\alpha-\beta}{\alpha+\beta+2}.$$

Note that when $\alpha \neq \beta$ and the smaller one is a half-integer in $[-\frac{1}{2}, \infty)$, the naive forward recursion of (23) is not numerically stable. In such a situation, a more sophisticated method for the solution of (23) needs to be used, for example, the approach presented in [46].

When α and β are identical, the numbers m_j are Gegenbauer moments, which have an explicit formula [37]:

$$m_j = \begin{cases} 0 & j \text{ odd}, \\ \dfrac{\sqrt{\pi}\,\Gamma(\alpha+1)}{\Gamma(\alpha+\frac{3}{2})} \displaystyle\prod_{k=1}^{j/2} \frac{k-\frac{3}{2}-\alpha}{k+\alpha+\frac{1}{2}} & j \text{ even}. \end{cases}$$

Remark 3. When $\alpha = \beta = -1/2$, the weights w_k become π/n identically, matching the weights of the Gauss–Chebyshev quadrature rule exactly [44, section 8.3].

2.10. Lebesgue constants

Given a grid of interpolation points in $[-1, 1]$, the *Lebesgue constant* for the grid is the norm of the linear operator that maps data sampled at the grid points to the corresponding polynomial interpolant [50,64]. (Here, the data are thought of as elements of a vector in C^n and the interpolant as belonging to the space of continuous functions on $[-1, 1]$, both measured using their respective infinity-norms.) It can be shown that the Lebesgue constant is the maximum value of the *Lebesgue function* $\lambda_{n-1}(x) = \sum_{j=0}^{n-1} |\ell_j(x)|$ over the domain, i.e.,

$$\Lambda_{n-1} = \sup_{x\in[-1,1]} \lambda_{n-1}(x).$$

The Lebesgue constant enables us to bound the infinity-norm difference between the polynomial interpolant and the best infinity-norm polynomial approximation of the same degree.

Theorem 4. *For an $(n+1)$-point Chebyshev grid of the first kind, the Lebesgue constant Λ_n has the following properties:*

(1) For $n = 0, 1, 2, \ldots$,

$$a_0 + \frac{2}{\pi}\log(n+1) < \Lambda_n \leq 1 + \frac{2}{\pi}\log(n+1), \tag{24}$$

where $a_0 = 0.9625\ldots$.

(2)

$$\Lambda_n \sim \frac{2}{\pi}\log(n+1) + a_0 + \sum_{k=1}^{\infty} \frac{a_k}{(n+1)^{2k}}, \tag{25}$$

where a_0 is given above and

$$a_k = \frac{(-1)^{k+1}(2^{2k-1}-1)^2\pi^{2k-1}B_{2k}^2}{4^{k-1}k(2k)!} \quad \text{for } k = 1, 2, \ldots$$

and B_k are the Bernoulli numbers. If finitely many terms are taken in (25), the truncation error has the same sign as the first neglected term and is less than it in absolute value.

(3) The Lebesgue constant Λ_{n-1} for degree $n-1$ interpolation on a Chebyshev grid of the first kind bounds Λ_n^*, the Lebesgue constant for degree n interpolation on a Chebyshev grid of the second kind, by

$$\Lambda_{n-1} = \Lambda_n^* + s_n,$$

where $s_n = 0$ for odd n and $\frac{\pi}{8(2n)^2} < s_n < \frac{2\sqrt{2}-2}{(2n)^2}$ for even n.

Proof. The bounds in (24) are given by Rivlin [54] by sharpening the results obtained by a series of authors. The asymptotic expansion (25) was first established by Günttner [32]. The relation between the Lebesgue constant for the first-kind Chebyshev grid and that of the second kind was found by Ehlich and Zeller [20] and McCabe and Phillips [45]. □

Here are the first 9 Lebesgue constants for the Chebyshev grids of the first and the second kind, which demonstrate part (3) of Theorem 4.

degree	first kind	second kind
1	1.414213562373095	1.000000000000000
2	1.666666666666667	1.250000000000000
3	1.847759065022574	1.666666666666667
4	1.988854381999833	1.798761803322554
5	2.104397682646484	1.988854381999832
6	2.202214555205530	2.082555385939792
7	2.287016068458566	2.202214555205529
8	2.361856787767076	2.274730766233740
9	2.428829482376075	2.361856787767072

The fact that $\Lambda_n > \Lambda_{n-1}$ due to the monotonic increase of Λ_n and part (3) of Theorem 4 implies that a Chebyshev grid of the first kind has a larger Lebesgue constant than the second-kind grid of the same size. But the difference is by no means significant, as suggested by the magnitude of s_n in part (3) of Theorem 4.

3. Applications of Chebyshev points of the first kind

The fundamentals of the last section show that the Chebyshev points of the first kind do not differ much from the second-kind ones, which suggests that similar numerical results should be expected if they are used for function approximation or solution of differential equations. In this section, we discuss certain benefits we can gain from using the Chebyshev points of the first kind.

3.1. Approximation

A distinctive property of the first-kind Chebyshev grid is the exclusion of the endpoints -1 and 1, which sometimes makes things easier when function approximations are constructed.

3.1.1. Construction of piecewise smooth functions

Robust algorithms for edge detection have made automatic construction of piecewise smooth function possible [47]. For instance, let us consider the construction of a two-piece representation for the sign function in $[-1, 1]$:

$$\operatorname{sgn}(x) = \begin{cases} -1 & \text{if } x < 0, \\ 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0. \end{cases}$$

An automatic constructor can locate the discontinuity at 0 using edge detection algorithms. Then the whole domain is split into two subdomains and an approximation is sought over each subdomain. One way to construct approximations with adequate resolution could be to sample the sign function in each subdomain on grids of increasing size until the Chebyshev coefficients decay to machine precision. However, if we sample, for example, the right subdomain on a second-kind Chebyshev grid mapped to $[0, 1]$, the vector of function values passed to the interpolant constructor is $\mathbf{f} = (0, 1, 1, \ldots, 1)^T$ where the first entry 0 is due to $\operatorname{sgn}(0) = 0$. The Chebyshev coefficients corresponding to this vector converge slowly, as the function values suggest non-smoothness.

One workaround is to discard the sampling value at the origin and then extrapolate the function value there under the assumption that the function value at the endpoints along with other sample values are from a smooth piece, and this is what Chebfun does when working in second-kind mode. The extrapolation makes use of the usual barycentric interpolation formula. Such a strategy works but at the cost of extra computation. On the other hand, it is not easy to predict when this is needed and therefore the use of this method may make the algorithm less easy to automate.

In contrast, if the sampling is done on first-kind Chebyshev grids, the cumbersome process just described is avoided, and we never need to worry about boundary issues.

3.1.2. Approximation of functions in unbounded domains

Methods for the approximation of functions defined on unbounded domains generally fall into two categories. Methods in the first category use basis functions intrinsic to an unbounded domain. These include sinc-related methods [60] which use sinc functions (also known as Whittaker cardinal functions) as the building blocks, and spectral methods based on Hermite functions or Laguerre functions which rely on the corresponding orthonormal system defined on an unbounded domain. These methods have certain limitations. For example, they are not efficient in approximating functions that decay algebraically or more slowly as the convergence cannot be geometric or exponential but only algebraic. Moreover, the sinc-related methods cannot be sped up by FFTs and, in general, do not give exact solutions to classic eigenvalue problems. For a more detailed discussion, see [6, p. 346].

The methods in the second category construct the approximation to a function defined on an unbounded domain by mapping the original function to a finite interval, for example $[-1, 1]$, and then seeking an approximation to the mapped function on a finite domain. Experiments show that mapping methods work well for functions that decay algebraically at

Table 2
Domain mappings for approximating functions defined on an unbounded domain. The variable y is defined on the unbounded domain, while x lives on $[-1,1]$. The forward map takes $[-1,1]$ to one of the three unbounded domains, and the inverse does the opposite.

Domain D	$(-\infty,\infty)$	$(-\infty,b]$	$[a,\infty)$
Forward map $[-1,1]\to D$	$y=\dfrac{5x}{1-x^2}$	$y=\dfrac{15(x-1)}{x+1}+b$	$y=\dfrac{15(x+1)}{1-x}+a$
Inverse map $D\to[-1,1]$	$x=\dfrac{2y}{5+\sqrt{25+4y^2}}$	$x=\dfrac{15+y-b}{15-y+b}$	$x=\dfrac{-15+y-a}{15+y-a}$

infinity. In practice, the sampling of the function is usually done by first mapping the interpolation points to the unbounded domain and then evaluating the function on the mapped grid. The maps recommended by Grosch and Orszag [31] and Boyd [5,6] are summarized in Table 2, subject to a slight change. If the second-kind Chebyshev grid is used, we have to treat the endpoints ±1 with extra care as they are mapped to $\pm\infty$, that is, `±Inf` in floating point arithmetic.

However, if Chebyshev grids of the first kind are used, the mapped sampling points are free of this issue and there is no need to add an extra check for the mapped endpoints.

3.2. Solution of integral equations

Chebyshev points of the first kind play an important role in the numerical solution of certain integral or integro-differential equations, which involve integral operators of the form

$$u \mapsto \int w(y)K(x,y)u(y)dy.$$

acting on $u(y)$. Here the weight function is $w(y)=1/\sqrt{1-y^2}$ and the kernel $K(x,y)$ is singular.

For instance, the singular integral equation of the first kind

$$⨍_{-1}^{1}\frac{u(t)dt}{t-s}+\lambda\int_{-1}^{1}u(t)K(s,t)dt=f(s), \qquad s\in(-1,1) \tag{26}$$

is of particular importance in certain engineering problems, such as elasticity and aerodynamics; see e.g. [29,27,17]. The Hilbert-type integral is interpreted as Cauchy principal value integral, indicated by $⨍$, and $K(s,t)$ is a Fredholm kernel [40, p. 72]. Equation (26) can be reduced to a Fredholm integral equation of the second kind by setting $u(t)=(1-t^2)^{-1/2}v(t)$; see e.g. [21,22], that is,

$$⨍_{-1}^{1}\frac{v(t)dt}{\sqrt{1-t^2}(t-s)}+\lambda\int_{-1}^{1}\frac{v(t)K(s,t)}{\sqrt{1-t^2}}dt=f(s). \tag{27}$$

A widely adopted approach for solving (27) is to use Chebyshev points of the first kind as the quadrature points for the evaluation of both the integrals and Chebyshev points of the second kind as the collocation points. The benefits are twofold. On the one hand, if the quadrature abscissae are the first-kind Chebyshev grid, the quadrature weights are simple and explicitly known (see Remark 3). On the other hand, by using different sets of points for quadrature and collocation, we do not have to compute the limit of the integrand in the Hilbert-type integral when the quadrature points and the collocation points coincide. Therefore, the discrete system we solve for $v(t)$ is

$$\frac{\pi}{n}\sum_{k=0}^{n-1}\left(\frac{1}{x_k-y_j}+\lambda K(y_j,x_k)\right)v(x_k)=f(y_j),$$

where x_k are the first-kind Chebyshev grid given by (1) while y_j are the second-kind points given by (2).

A similar strategy of using the Chebyshev points of the first kind can also be applied to other singular integral equations, such as integral equation of Symm's type; see e.g. [58,36,41].

3.3. Solution of differential equations

Spectral collocation methods approximate solutions of differential equations by polynomial interpolants that satisfy the given equation at a set of carefully chosen points, the *collocation points*. Chebyshev points of either kind are among the most natural choices for spectral collocation methods. As an example, consider the two-point boundary value problem

$$u_{xx}=f(x), \quad -1<x<1, \quad u(-1)=0 \text{ and } u(1)=0. \tag{28}$$

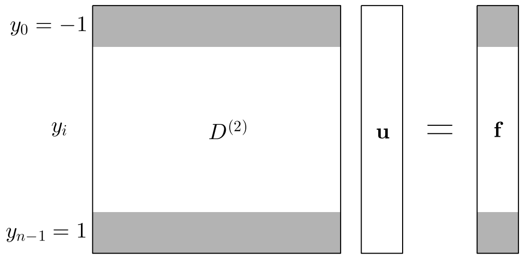

Fig. 3. Traditionally, boundary conditions are imposed by row replacement. The top and the bottom rows in grey are replaced by rows corresponding to the boundary conditions.

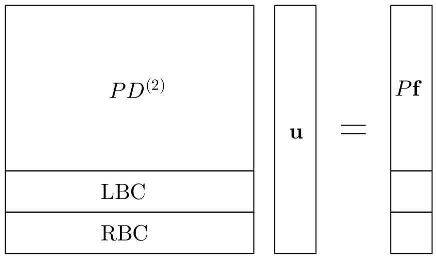

Fig. 4. In the alternative approach introduced by Driscoll and Hale, boundary conditions are appended to a rectangular differentiation matrix, where the second-order differential operator $D^{(2)}$ is rectangularized by the barycentric interpolation matrix P and the right hand side $\mathbf{f}$ is mapped to the same grid too.

When the second-kind Chebyshev grid given by (2) is adopted as is common, (28) is discretized as

$$D^{(2)}\mathbf{u} = \mathbf{f}, \tag{29}$$

where $\mathbf{f}$ is the column vector whose j-th entry is the value of $f(x)$ at y_j, while $\mathbf{u}$ is the discretized version of u. The matrix $D^{(2)}$ is the second-order differentiation matrix defined on a second-kind Chebyshev grid; see e.g. [62, Chapter 6]. To impose the boundary conditions at ± 1, the first and the last rows of (29) are replaced by the boundary conditions, as shown in Fig. 3. This serves two purposes. On the one hand, the second order differential equation in (28) governs only the interior of the domain. Therefore the first and the last row which correspond to the boundary points should not be included. On the other hand, the matrix on the left-hand side of (29) is singular and hence needs to be regularized by adding rows with "independent" information. Enforcing the boundary conditions using these kinds of row replacements seem reasonable for such a simple example.

In more general situations, how to decide which rows to remove is not always so obvious. For example, if the boundary condition at 1 is changed to a more complicated boundary condition like $u(-1) + u(1) = c$ or a side condition such as $\int_{-1}^{1} u(x)dx = c$ with c a given constant, the row replacement approach often does not work. To circumvent this, an ingenious method of discretization was suggested by Driscoll and Hale [18], as they observed that a differentiation matrix should be rectangular instead of square and boundary information can be further added. Specifically, an order d differentiation matrix should have dimension $(n - d) \times n$. This makes good sense as each differentiation reduces the degree of a polynomial interpolant by one and a polynomial interpolant of degree $(n - 1)$ can be uniquely defined by a set of n values. For the present example, the second-order rectangular differentiation matrix on the left-hand side of (29) should map from an n-point Chebyshev grid of either kind to an $(n - 2)$-point Chebyshev grid of the first kind. Here the choice of the first-kind Chebyshev grid is important, since the collocation grid should not include the boundary points. This rectangular differentiation matrix can be formed by pre-multiplying the standard square differentiation matrix with an $(n - 2) \times n$ barycentric interpolation matrix P which maps from an n-point grid to an $(n - 2)$-point one using barycentric interpolation formulae. Likewise, we map the values of $f(x)$ on the right hand side to the $(n - 2)$-point first-kind grid as well, using the same barycentric interpolation matrix or by simply sampling at the $(n - 2)$-point Chebyshev grid of the first kind. Finally, the system is "squared up" by appending the boundary conditions, as shown in Fig. 4.

For details of the rectangular spectral collocation method and the explicit construction of rectangular differentiation matrices, see [18] and [68]. It also worth noting that a similar idea was exploited by Kopriva et al. for the numerical simulation of compressible flows [39,38] using so-called *staggered grids*.

4. Conclusion

We have reviewed the fundamental properties of the Chebyshev points of the first kind, which show a clear similarity to those of the more widely used Chebyshev points of the second kind. We have also discussed some examples in function approximation and integral and differential equations for which benefits can be gained by using the first-kind Chebyshev points. We hope to have made clear that certain properties of the first-kind Chebyshev points, e.g., exclusion of the boundary points and their interlacing with the second-kind Chebyshev points, can sometimes make them more convenient than their second-kind counterparts.

Acknowledgements

This work was supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement 291068. The views expressed in this article are not those of the ERC or the European Commission, and the European Union is not liable for any use that may be made of the information contained here. The author is grateful to Nick Trefethen and Anthony Austin for their encouragement and insightful comments. The author also thanks the anonymous referees, whose comments and suggestions led him to greatly improve the manuscript.

References

[1] R. Baltensperger, J.P. Berrut, The errors in calculating the pseudospectral differentiation matrices for Čebyšev–Gauss–Lobatto points, Comput. Math. Appl. 37 (1999) 41–48.
[2] Z. Battles, L.N. Trefethen, An extension of MATLAB to continuous functions and operators, SIAM J. Sci. Comput. 25 (2004) 1743–1770.
[3] S. Bernstein, Sur l'ordre de la meilleure approximation des fonctions continues par des polynomes de degré donné, Mém. Cl. Sci., Acad. R. Belg., Coll. 8 (1912) 1–104.
[4] J.P. Berrut, L.N. Trefethen, Barycentric Lagrange interpolation, SIAM Rev. 46 (2004) 501–517.
[5] J.P. Boyd, The optimization of convergence for Chebyshev polynomial methods in an unbounded domain, J. Comput. Phys. 45 (1982) 43–79.
[6] J.P. Boyd, Chebyshev and Fourier Spectral Methods, second edition, Dover Publications, Inc., Mineola, NY, 2001.
[7] K.S. Breuer, R.M. Everson, On the errors incurred calculating derivatives using Chebyshev polynomials, J. Comput. Phys. 99 (1992) 56–67.
[8] L. Brutman, On the Lebesgue function for polynomial interpolation, SIAM J. Numer. Anal. 15 (1978) 694–704.
[9] L. Brutman, Lebesgue functions for polynomial interpolation — a survey, Ann. Numer. Math. 4 (1997) 111–127.
[10] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods: Fundamentals in Single Domains, Springer-Verlag, Berlin, 2006.
[11] P. Chebyshev, Théorie des mécanismes connus sous le nom de parallélogrammes, Mém. Acad. Imp. Sci. St.-Pétersbg. 7 (1853) 539–568.
[12] P. Chebyshev, Sur les questions de minima qui se rattachent a la représentation approximative des fonctions, Mém. Acad. Imp. Sci. St.-Pétersbg. 7 (1859) 199–291.
[13] E.W. Cheney, Introduction to Approximation Theory, AMS Chelsea Publishing, Providence, RI, 1998.
[14] C.W. Clenshaw, A.R. Curtis, A method for numerical integration on an automatic computer, Numer. Math. 2 (1960) 197–205.
[15] P.J. Davis, P. Rabinowitz, Methods of Numerical Integration, Dover Publications, Inc., Mineola, NY, 2007.
[16] W.S. Don, A. Solomonoff, Accuracy and speed in computing the Chebyshev collocation derivative, SIAM J. Sci. Comput. 16 (1995) 1253–1268.
[17] M.A. Driscoll, R.P. Srivastav, Rational function approximations in the numerical solution of Cauchy-type singular integral equations, Comput. Math. Appl. 11 (1985) 949–965.
[18] T.A. Driscoll, N. Hale, Rectangular spectral collocation, IMA J. Numer. Anal. (2015), http://dx.doi.org/10.1093/imanum/dru062.
[19] T.A. Driscoll, N. Hale, L.N. Trefethen, Chebfun Guide, Pafnuty Publications, Oxford, 2014.
[20] H. Ehlich, K. Zeller, Auswertung der Normen von Interpolationsoperatoren, Math. Ann. 164 (1966) 105–112.
[21] F. Erdogan, G.D. Gupta, On the numerical solution of singular integral equations, Q. Appl. Math. 29 (1972) 525–534.
[22] F. Erdogan, G.D. Gupta, T.S. Cook, Numerical solution of singular integral equations, in: Mechanics of Fracture, vol. 1, Noordhoff, Leiden, 1973, pp. 368–425.
[23] P. Erdős, Problems and results on the theory of interpolation. II, Acta Math. Acad. Sci. Hung. 12 (1961) 235–244.
[24] L. Fejér, Über Interpolation, Nachr. Ges. Wiss. Gött., Math.-Phys. Kl. (1916) 66–91.
[25] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge University Press, Cambridge, 1996.
[26] L. Fox, I.B. Parker, Chebyshev Polynomials in Numerical Analysis, Oxford University Press, London–New York–Toronto, 1968.
[27] J.A. Fromme, M.A. Golberg, Numerical solution of a class of integral equations arising in two-dimensional aerodynamics, in: Solution Methods for Integral Equations, Plenum, New York–London, 1979, pp. 109–146.
[28] W. Gautschi, Numerical Analysis, An Introduction, Birkhäuser Boston, Inc., Boston, MA, 1997.
[29] G.M.L. Gladwell, A.H. England, Orthogonal polynomial solutions to some mixed boundary-value problems in elasticity theory, Q. J. Mech. Appl. Math. 30 (1977) 175–185.
[30] D. Gottlieb, M.Y. Hussaini, S.A. Orszag, Theory and applications of spectral methods, in: Spectral Methods for Partial Differential Equations, Hampton, Va., 1982, SIAM, Philadelphia, PA, 1984, pp. 1–54.
[31] C.E. Grosch, S.A. Orszag, Numerical solution of problems in unbounded regions: coordinate transforms, J. Comput. Phys. 25 (1977) 273–295.
[32] R. Günttner, Evaluation of Lebesgue constants, SIAM J. Numer. Anal. 17 (1980) 512–520.
[33] J.G. Hayes (Ed.), Numerical Approximation to Functions and Data, The Athlone Press, London, 1970.
[34] P. Henrici, Essentials of Numerical Analysis with Pocket Calculator Demonstrations, John Wiley & Sons, Inc., New York, 1982.
[35] J.S. Hesthaven, S. Gottlieb, D. Gottlieb, Spectral Methods for Time-Dependent Problems, Cambridge University Press, Cambridge, 2007.
[36] D.M. Hough, J. Levesley, S.N. Chandler-Wilde, Numerical conformal mapping via Chebyshev weighted solutions of Symm's integral equation, J. Comput. Appl. Math. 46 (1993) 29–48.
[37] D.B. Hunter, G. Nikolov, Gaussian quadrature of Chebyshev polynomials, J. Comput. Appl. Math. 94 (1998) 123–131.
[38] D.A. Kopriva, A staggered-grid multidomain spectral method for the compressible Navier–Stokes equations, J. Comput. Phys. 143 (1998) 125–158.
[39] D.A. Kopriva, J.H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, J. Comput. Phys. 125 (1996) 244–261.
[40] R. Kress, Linear Integral Equations, third edition, Applied Mathematical Sciences, vol. 82, Springer, New York, 2014.
[41] J. Levesley, D.M. Hough, S.N. Chandler-Wilde, A Chebyshev collocation method for solving Symm's integral equation for conformal mapping: a partial error analysis, IMA J. Numer. Anal. 14 (1994) 57–79.

[42] G.G. Lorentz, Approximation of Functions, second edition, Chelsea Publishing Co., New York, 1986.
[43] J. Makhoul, A fast cosine transform in one and two dimensions, IEEE Trans. Acoust. Speech Signal Process. 28 (1980) 27–34.
[44] J.C. Mason, D.C. Handscomb, Chebyshev Polynomials, Chapman & Hall/CRC, Boca Raton, FL, 2003.
[45] J.H. McCabe, G.M. Phillips, On a certain class of Lebesgue constants, BIT Numer. Math. 13 (1973) 434–442.
[46] J. Oliver, The numerical solution of linear recurrence relations, Numer. Math. 11 (1968) 349–360.
[47] R. Pachón, R.B. Platte, L.N. Trefethen, Piecewise-smooth chebfuns, IMA J. Numer. Anal. 30 (2010) 898–916.
[48] R. Peyret, Spectral Methods for Incompressible Viscous Flow, Springer-Verlag, New York, 2002.
[49] R. Piessens, M. Branders, The evaluation and application of some modified moments, BIT Numer. Math. 13 (1973) 443–450.
[50] M.J.D. Powell, Approximation Theory and Methods, Cambridge University Press, Cambridge–New York, 1981.
[51] E.D. Rainville, Special Functions, The Macmillan Company, New York, 1960.
[52] M. Riesz, Über einen Satz des Herrn Serge Bernstein, Acta Math. 40 (1916) 337–347.
[53] T.J. Rivlin, The Chebyshev Polynomials, John Wiley & Sons, New York–London–Sydney, 1974.
[54] T.J. Rivlin, The Lebesgue constants for polynomial interpolation, in: Functional Analysis and its Applications, Internat. Conf., Eleventh Anniversary of Matscience dedicated to Alladi Ramakrishnan, Madras, 1973, Springer, Berlin, 1974, pp. 422–437.
[55] H.E. Salzer, Lagrangian interpolation at the Chebyshev points $X_{n,\nu} \equiv \cos(\nu\pi/n)$, $\nu = 0(1)n$; some unnoted advantages, Comput. J. 15 (1972) 156–159.
[56] H.R. Schwarz, Numerical Analysis: A Comprehensive Introduction, John Wiley & Sons, Chichester, 1989.
[57] J. Shen, T. Tang, L.L. Wang, Spectral Methods: Algorithms, Analysis and Applications, Springer, Heidelberg, 2011.
[58] I.H. Sloan, E.P. Stephan, Collocation with Chebyshev polynomials for Symm's integral equation on an interval, J. Aust. Math. Soc. Ser. B, Appl. Math 34 (1992) 199–211.
[59] A. Sommariva, Fast construction of Fejér and Clenshaw–Curtis rules for general weight functions, Comput. Math. Appl. 65 (2013) 682–693.
[60] F. Stenger, Numerical Methods Based on Sinc and Analytic Functions, Springer-Verlag, New York, 1993.
[61] A.F. Timan, Theory of Approximation of Functions of a Real Variable, Dover Publications, Inc., New York, 1994.
[62] L.N. Trefethen, Spectral Methods in MATLAB, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.
[63] L.N. Trefethen, Is Gauss quadrature better than Clenshaw–Curtis?, SIAM Rev. 50 (2008) 67–87.
[64] L.N. Trefethen, Approximation Theory and Approximation Practice, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2013.
[65] J. Waldvogel, Fast construction of the Fejér and Clenshaw–Curtis quadrature rules, BIT Numer. Math. 46 (2006) 195–202.
[66] J.A.C. Weideman, S.C. Reddy, A MATLAB differentiation matrix suite, ACM Trans. Math. Softw. 26 (2000) 465–519.
[67] G. von Winckel, Codes for fast Clenshaw–Curtis quadrature, http://www.mathworks.co.uk/matlabcentral/fileexchange/6911-fast-clenshaw-curtis-quadrature, 2005.
[68] K. Xu, N. Hale, Explicit construction of rectangular differentiation matrices, IMA J. Numer. Anal. (2015), http://dx.doi.org/10.1093/imanum/drv013.