A GENERIC AND STRICTLY BANDED SPECTRAL PETROV-GALERKIN METHOD FOR DIFFERENTIAL EQUATIONS WITH POLYNOMIAL COEFFICIENTS

MIKAEL MORTENSEN*

Abstract. In this paper we describe a generic spectral Petrov-Galerkin method that is sparse and strictly banded for any linear ordinary differential equation with polynomial coefficients. The method applies to all subdivisions of Jacobi polynomials (e.g., Chebyshev and Legendre), utilises well-known recurrence relations of orthogonal polynomials and leads to almost exactly the same discretized system of equations as the integration preconditioners (IP) [Coutsias et al., Math Comp, 65 (1996), 611–635], if this method was redesigned to make use of trial functions that satisfy a given problems boundary conditions. A link between the new Petrov-Galerkin method and IP is revealed through a new recursion relation for Jacobi polynomials. Because of the strictly banded nature of all coefficient matrices, the new method extends easily and efficiently to multiple dimensions though the use of tensor product methods.

1. Introduction. Spectral methods are widespread in most branches of natural sciences, with several books dedicated entirely to the subject [3, 35, 16, 5, 22, 19]. Spectral methods are favoured by scientists aiming at ultimate accuracy in as few degrees of freedom as possible, making them particularly successful in fields such as meteorologi, turbulence, hydrodynamic stability, geophysical flows, stochastic differential equations and uncertainty quantifications. Common for these fields is that physical processes can be studied with high precision in simple Cartesian product domains, which is a requirement since global spectral methods can be difficult, or impossible, to apply to irregular domains. This disadvantage can sometimes be overcome by mapping a complex domain into a regular through an explicit, smooth [30], or a Gordon-Hall mapping [15], or, alternatively, by embedding the complex domain into a larger regular domain [18]. Still, the possibility to study physical processes with extreme accuracy in very few degrees of freedom has always been attractive to scientists, and it has recently inspired the development of several spectral software frameworks [10, 28, 4, 26, 25].

We will in this paper be interested in the global spectral methods that are referred to as spectral Galerkin, and more specifically spectral Petrov-Galerkin methods. These methods solve equations in spectral space, as opposed to collocation (or pseudospectral) methods [13] that solve equations in physical space. The Galerkin methods have a clean and elegant design, using variational principles and function spaces with built-in boundary conditions, that can be easily analysed. The Galerkin method is also easily implemented and automated, evidenced, e.g., by the large number of generic finite element software frameworks that have emerged in later years [1]. The Tau [24] method is quite closely related to spectral Galerkin, also solving equations in spectral space, but with a primary focus towards finding spectral differentiation matrices in the orthogonal basis. The Tau and Galerkin methods differ the most in how the boundary conditions are specified. The Tau method enforces boundary conditions by modifying rows of the coefficient matrix, whereas Galerkin builds homogeneous boundary conditions into the basis functions, and adds non-homogeneous boundary conditions through additional lifting functions [2]. The Galerkin approach has the advantage that the coefficient matrices remain strictly banded regardless of boundary condition, whereas the Tau-matrices become *almost-banded*, see, e.g., [27].

We will in this work limit ourselves to global spectral methods that make use of or-

*Department of Mathematics, University of Oslo, Norway (mikaem@math.uio.no).

thogonal Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$ as basis functions. This includes both Chebyshev, Legendre and all ultraspherical polynomials, but excludes Fourier exponentials and Hermite/Laguerre polynomials. Many efficient global methods for the Jacobi polynomials have already been described, and most of these methods are based on ideas that date back to Clenshaw [6] and Orzag [29], taking advantage of several recurrence relations that exists for orthogonal polynomials. Of particular importance is the recurrence relation

$$Q_n^{(\alpha)}(x) = b_{n-1,n}\frac{d}{dx}Q_{n-1}^{(\alpha)}(x) + b_{n+1,n}\frac{d}{dx}Q_{n+1}^{(\alpha)}(x), \tag{1.1}$$

for the ultraspherical polynomial $Q_n^{(\alpha)}(x) \sim P_n^{(\alpha,\alpha)}(x)$, where $B = (b_{ij})$ is a matrix operator. The best global methods that are known to take advantage of recursions like (1.1), are probably those based on integration reformulation [17, 36, 11], the related integration preconditioner [8, 7, 23] or the ultraspherical approach [27, 4, 14]. With integration reformulation the n'th order differential equation is first integrated n times, before an integral version of (1.1) is used on all lower order terms in the equation, leading to a banded linear system of equations (see, e.g., [36]). With integration preconditioners the matrix B of the recurrence (1.1) is used explicitly as a preconditioner on the otherwise poorly conditioned and full algebraic equations that are assembled for the Tau-method. The integration preconditioners also make use of other recurrence relations and obtain sparse systems off equations for linear differential equations with rational functions as coefficients, see [7]. The ultraspherical approach [27] makes use of recurrence (1.1), but somewhat camouflaged into a correlation between Chebyshev polynomials of first $T_k(x)$ and second $U_k(x)$ kind, see relation (2.8) given in [27]. Numerous other methods (e.g., [17, 14, 12]) rely on the same recurrence relations, in one form or another. However, there has to the author's best knowledge never been described a generic spectral Galerkin, or Petrov-Galerkin, method for Jacobi polynomials that take systematic advantage of recurrence relations, and that can match the sparsity of for example the integration preconditioners for variable coefficient equations. For specific equations, boundary conditions and bases there are of course exceptions. Shen has suggested a sparse and efficient method with compact Legendre polynomials [31], whereas Guo et al. [20] obtain sparse and efficient methods using generalized Jacobians. Elbarbary [12] describe a sparse Petrov-Galerkin method with Chebyshev polynomials for constant coefficient second-order equations subject to either Dirichlet or Neumann boundary conditions.

In his seminal paper series on efficient direct solvers for the spectral-Galerkin method [31, 32, 33, 34] Shen notes that *it is surprising that virtually no effort has been made to construct appropriate bases for the spectral Galerkin method.* His recommended approach is to use the most compact combinations of orthogonal basis functions that satisfy a given problems boundary conditions, for both the identical test and trial spaces. However, this does not always lead to sparse matrices, and in general it leads to algebraic problems that require tailored solvers for efficiency. In this paper we intend to show that a k'th order linear differential equation with polynomial coefficients simply can use a basis with the very specific test function

$$\phi_n^{(k)}(x) = (1-x^2)^k \frac{d^k}{dx^k}P_{n+k}^{(\alpha,\beta)}(x), \quad n \geq 0, k > 0, \tag{1.2}$$

regardless of boundary conditions for the problem under investigation. We will then automatically get a sparse and strictly banded Petrov-Galerkin method in the natural $L^2_{\omega^{(\alpha,\beta)}}$

space, as long as the trial function is chosen compactly (basis recombination) from the natural basis $\{P_n^{(\alpha,\beta)}\}$. Furthermore, through a new recurrence relation we will show that such a Petrov-Galerkin method will have a lot in common with the integration proconditioners [8, 7], or the quasi-inverse [23] approach if restricted to constant coefficient equations and Chebyshev polynomials.

This paper is outlined as follows: in Sec. 2 we present necessary theory for Jacobi and ultraspherical polynomials and a new recursion relation. In Sec. 3 we consider spectral differentiation in the frequency space and show how a test function like (1.2) can simplify the description considerably through a Petrov-Galerkin formulation. In Sec. 4 we present the new and sparse Petrov-Galerkin method for linear differential equations with either constant or polynomial coefficients. The extension to multiple dimensions is described briefly, and some numerical examples are presented. Conclusions are drawn in Sec. 5

2. Preliminaries. In this section we introduce some necessary identities and recurrence relations for Jacobi polynomials. We will mainly be interested in ultraspherical polynomials, like Legendre or Chebyshev, but the main results are applicable for any Jacobi basis, which is why we find it natural to start here.

2.1. Jacobi polynomials. The Jacobi polynomials, $P_n^{(\alpha,\beta)}(x)$, are found as eigensolutions to the Sturm-Lioville problem in the domain $x \in [-1, 1]$. The first two polynomials are

$$P_0^{(\alpha,\beta)} = 1, \quad P_1^{(\alpha,\beta)} = \frac{1}{2}(\alpha+\beta+2)x + \frac{1}{2}(\alpha-\beta), \tag{2.1}$$

and the remaining can be found through the recurrence relation

$$xP_n^{(\alpha,\beta)} = a_{n-1,n}^{(\alpha,\beta)} P_{n-1}^{(\alpha,\beta)} + a_{n,n}^{(\alpha,\beta)} P_n^{(\alpha,\beta)} + a_{n+1,n}^{(\alpha,\beta)} P_{n+1}^{(\alpha,\beta)}, \tag{2.2}$$

where

$$\begin{aligned} a_{n-1,n}^{(\alpha,\beta)} &= \frac{2(n+\alpha)(n+\beta)}{(2n+\alpha+\beta+1)(2n+\alpha+\beta)}, \\ a_{n,n}^{(\alpha,\beta)} &= -\frac{\alpha^2-\beta^2}{(2n+\alpha+\beta+2)(2n+\alpha+\beta)}, \\ a_{n+1,n}^{(\alpha,\beta)} &= \frac{2(n+1)(n+\alpha+\beta+1)}{(2n+\alpha+\beta+2)(2n+\alpha+\beta+1)}. \end{aligned} \tag{2.3}$$

The two parameters α and β are both real numbers > -1, and the boundary values of the Jacobi polynomials can be found as

$$P_n^{(\alpha,\beta)}(1) = \binom{n+\alpha}{n}, \quad P_n^{(\alpha,\beta)}(-1) = (-1)^n \binom{n+\beta}{n}. \tag{2.4}$$

On matrix form we will write

$$x\boldsymbol{P} = A^T \boldsymbol{P}, \tag{2.5}$$

for the infinite-dimensional matrix operator $A^{(\alpha,\beta)} = (a_{mn}^{(\alpha,\beta)})_{m,n=0}^{\infty}$ and the column vector $\boldsymbol{P}^{(\alpha,\beta)} = (P_0^{(\alpha,\beta)}, P_1^{(\alpha,\beta)}, \ldots)^T$. Note that if the parameters are simply (α, β), and it is

possible to avoid confusion, then we will simply omit the superscript from the matrix and vector operators, like in Eq. (2.5).

If we multiply (2.5) by x^{q-1}, for integer $q > 0$, and then use the original (2.5) $q-1$ times on the right hand side, we get a nested recursion

$$x^q \boldsymbol{P} = (A^q)^T \boldsymbol{P}, \tag{2.6}$$

where $A^q = (a^{(q)}_{mn})_{m,n=0}^{\infty}$ is the q'th matrix power of A.

For integer $k > 0$ the k'th derivative of $P_n^{(\alpha,\beta)}$ with respect to x is known to be [35]

$$\partial^k P_n^{(\alpha,\beta)} = \psi_n^{(k,\alpha,\beta)} P_{n-k}^{(\alpha+k,\beta+k)}, \quad n \geq k, \tag{2.7}$$

where ∂^k is conveniently used to represent the ordinary derivative $\frac{d^k}{dx^k}$, and

$$\psi_n^{(k,\alpha,\beta)} = \frac{(n+\alpha+\beta+1)_k}{2^k}, \tag{2.8}$$

using the Pochhammer symbol $(\alpha)_k = \Gamma(\alpha+k)/\Gamma(\alpha)$.

The Jacobi polynomials also satisfy a recurrence relation of the form

$$P_n^{(\alpha,\beta)} = b_{n-1,n}^{(\alpha,\beta)} \partial P_{n-1}^{(\alpha,\beta)} + b_{n,n}^{(\alpha,\beta)} \partial P_n^{(\alpha,\beta)} + b_{n+1,n}^{(\alpha,\beta)} \partial P_{n+1}^{(\alpha,\beta)}, \tag{2.9}$$

or

$$\boldsymbol{P} = B^T \partial \boldsymbol{P}, \tag{2.10}$$

where the matrix operator $B = (b_{mn}^{(\alpha,\beta)})_{m,n=0}^{\infty}$, $\partial \boldsymbol{P} = (0, \partial P_1^{(\alpha,\beta)}, \partial P_2^{(\alpha,\beta)}, \ldots)^T$ and

$$b_{n-1,n}^{(\alpha,\beta)} = -\frac{a_{n-1,n}^{(\alpha,\beta)}}{n+\alpha+\beta}, \quad b_{n,n}^{(\alpha,\beta)} = -\frac{2a_{n,n}^{(\alpha,\beta)}}{\alpha+\beta}, \quad b_{n+1,n}^{(\alpha,\beta)} = \frac{a_{n+1,n}^{(\alpha,\beta)}}{n+1}. \tag{2.11}$$

Note that negative indices into the matrix components are here and throughout treated by setting the component to zero.

Higher order derivatives satisfy (see Eq. (11) [7])

$$\partial^{k-l} \boldsymbol{P} = (B^l)^T \partial^k \boldsymbol{P}, \tag{2.12}$$

where $0 < l \leq k$, and the first k items of the vector $\partial^k \boldsymbol{P}$ are 0. Note that the bandwidth of the matrix B^l is $\leq 1 + 2l$, see [7].

The Jacobi polynomials $\boldsymbol{P}_N^{(\alpha,\beta)} = (P_0^{(\alpha,\beta)}, P_1^{(\alpha,\beta)}, \ldots, P_N^{(\alpha,\beta)})^T$ form an orthogonal basis in $L^2_{\omega^{(\alpha,\beta)}}[-1,1]$ for P_N, which is the set of polynomials of degree less than or equal to N. The weight $\omega^{(\alpha,\beta)} = (1-x)^\alpha (1+x)^\beta$, and we have

$$\begin{aligned} \left(P_n^{(\alpha,\beta)}, P_m^{(\alpha,\beta)}\right)_{\omega^{(\alpha,\beta)}} &= \int_{-1}^{1} P_n^{(\alpha,\beta)} P_m^{(\alpha,\beta)} \omega^{(\alpha,\beta)} dx, \\ &= h_m^{(\alpha,\beta)} \delta_{mn}, \end{aligned} \tag{2.13}$$

where δ_{mn} is the Kronecker delta-function and

$$h_n^{(\alpha,\beta)} = \frac{2^{\alpha+\beta+1}}{(2n+\alpha+\beta+1)n!} \frac{\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{\Gamma(n+\alpha+\beta+1)}. \tag{2.14}$$

Using (2.7) in (2.13) we find that the k'th derivatives of the Jacobi polynomials are orthogonal with respect to $\omega^{(\alpha+k,\beta+k)}$

$$\left(\partial^k P_{n+k}^{(\alpha,\beta)}, \partial^k P_{m+k}^{(\alpha,\beta)}\right)_{\omega^{(\alpha+k,\beta+k)}} = h_{m+k}^{(k,\alpha,\beta)} \delta_{m+k,n+k}, \quad \text{for} \quad m,n \geq 0, \tag{2.15}$$

where

$$h_n^{(k,\alpha,\beta)} = h_{n-k}^{(\alpha+k,\beta+k)} \left(\psi_n^{(k,\alpha,\beta)}\right)^2. \tag{2.16}$$

This result is derived also in [35], and it is the key to the sparse spectral Galerkin methods discussed in the current paper. Note that $h_n^{(k,\alpha,\beta)} = 0$ for $n < k$, and for simplicity we use $h_n^{(\alpha,\beta)} = h_n^{(0,\alpha,\beta)}$. On matrix form we will use the diagonal matrix operators $H = \text{diag}(h_0^{(\alpha,\beta)}, h_1^{(\alpha,\beta)}, \ldots)$ and $H^{(k)} = \text{diag}(h_0^{(k,\alpha,\beta)}, h_1^{(k,\alpha,\beta)}, \ldots)$, where the first k rows and columns of the matrix $H^{(k)}$ are 0.

Since $\partial^k \boldsymbol{P}$ are orthogonal polynomials they will also satisfy a three-term recurrence relation like (2.5), when multiplied by x. The relation is easily obtained by inserting for (2.7) in (2.5)

$$x\partial^k P_n^{(\alpha,\beta)} = \sum_{m=n-1}^{n+1} \underline{a}_{mn}^{(k,\alpha,\beta)} \partial^k P_m^{(\alpha,\beta)}, \tag{2.17}$$

where the nonzero components of the tri-diagonal matrix operator $\underline{a}_{mn}^{(k,\alpha,\beta)}$ are

$$\underline{a}_{m+k,n+k}^{(k,\alpha,\beta)} = (\psi_{m+k}^{(k,\alpha,\beta)})^{-1} a_{mn}^{(\alpha+k,\beta+k)} \psi_{n+k}^{(k,\alpha,\beta)}, \quad \forall\, m,n \geq 0. \tag{2.18}$$

On matrix form we get

$$x\partial^k \boldsymbol{P} = \underline{A}^T \partial^k \boldsymbol{P}, \tag{2.19}$$

where $\underline{A}^{(k,\alpha,\beta)} = (\underline{a}_{mn}^{(k,\alpha,\beta)})_{m,n=0}^{\infty}$ has both the first k columns and rows equal to 0. Multiplying (2.19) by x^{q-1}, for integer $q > 0$, and recursively using (2.19) on the right hand side leads to

$$x^q \partial^k P_n^{(\alpha,\beta)} = \sum_{m=n-q}^{n+q} \underline{a}_{mn}^{(k,q,\alpha,\beta)} \partial^k P_m^{(\alpha,\beta)}, \tag{2.20}$$

where $\underline{a}_{mn}^{(k,q,\alpha,\beta)}$ is a component of the q'th matrix power of $\underline{A}^{(k,\alpha,\beta)}$. Note that $A^{(\alpha,\beta)} = \underline{A}^{(0,\alpha,\beta)}$.

Using the recursion relations above together with the orthogonality (2.15) we can obtain three important inner products in $L^2_{\omega^{(\alpha+k,\beta+k)}}[-1,1]$ for the Jacobi polynomials

$$\left(\partial^{k-l} P_n, \partial^k P_m\right)_{\omega^{(\alpha+k,\beta+k)}} = h_m^{(k)}\, b_{mn}^{(l)}, \tag{2.21}$$

$$\left(\partial^k P_n, x^q \partial^k P_m\right)_{\omega^{(\alpha+k,\beta+k)}} = h_m^{(k)}\, \underline{a}_{mn}^{(k,q)}, \tag{2.22}$$

$$\left(\partial^{k-l} P_n, x^q \partial^k P_m\right)_{\omega^{(\alpha+k,\beta+k)}} = \sum_{s=m-q}^{m+q} h_m^{(k)}\, \underline{a}_{ms}^{(k,q)} b_{sn}^{(l)}, \tag{2.23}$$

where the (α, β) superscript has been dropped for simplicity. Also, we have used the transpose equality

$$h_n^{(k)} \underline{a}_{nm}^{(k,q)} = h_m^{(k)} \underline{a}_{mn}^{(k,q)}, \tag{2.24}$$

which follows since $\left(\partial^k P_n, x^q \partial^k P_m\right)_{\omega^{(\alpha+k,\beta+k)}} = \left(x^q \partial^k P_n, \partial^k P_m\right)_{\omega^{(\alpha+k,\beta+k)}}$. Equation (2.21) follows by inserting for $\partial^{k-l} P_n$ on the left hand side using (2.12), and then forming the right hand side using (2.15). Equation (2.22) follows by combining (2.20) and (2.15), whereas (2.23) follows by using (2.20) and (2.21). The bandwidth of (2.21) is $1+2l$, of (2.22) $1+2q$, and of (2.23) it is $1+2(q+l)$. Note that Eq. (2.23) is a generic form that simplifies to Eq. (2.22) for $l=0$, Eq. (2.21) for $q=0$, Eq. (2.15) for $l=q=0$, and (2.13) for $k=l=q=0$. The matrix components on the right hand side are then simplified by using that the zeroth matrix power equals the identity matrix.

Finally, we introduce in Lemma (2.1) a new recursion relation that will be heavily utilized in this paper.[1]

LEMMA 2.1. *The Jacobi polynomials satisfy the recursion relation*

$$(1-x^2)^k \partial^k \boldsymbol{P} = (C^{(k)})^T \boldsymbol{P}, \tag{2.25}$$

where $(C^{(k)})^T = H^{(k)} B^k H^{-1}$ *for integer* $k > 0$.

Proof. We first write the relation on index form as

$$(1-x^2)^k \partial^k P_m = \sum_{s=m-k}^{m+k} h_m^{(k)} b_{ms}^{(k)} h_s^{-1} P_s, \quad m \geq 0. \tag{2.26}$$

The relation is obviously true for $0 \leq m < k$ since both sides are then zero. It is also obvious that both sides of the equation have the same polynomial order $m+k$ for any given $m \geq k$. Hence we can write the left hand side as an expansion in Jacobi polynomials

$$(1-x^2)^k \partial^k P_m = \sum_{s=0}^{m+k} c_{sm}^{(k)} P_s, \quad m \geq k, \tag{2.27}$$

for some matrix $C^{(k)}$ with upper bandwidth k. In order to show that $c_{sm}^{(k)} = h_m^{(k)} b_{ms}^{(k)} h_s^{-1}$ we take the weighted inner product of (2.27) with P_n, for $n \geq 0$

$$\left(P_n, (1-x^2)^k \partial^k P_m\right)_{\omega^{(\alpha,\beta)}} = \sum_{s=0}^{m+k} \left(P_n, c_{sm}^{(k)} P_s\right)_{\omega^{(\alpha,\beta)}}.$$

The inner product on the left is given by (2.21) with $k=l$, and the right hand side can be simplified using (2.15)

$$h_m^{(k)}\, b_{mn}^{(k)} = \sum_{s=0}^{m+k} c_{sm}^{(k)} h_s \delta_{ns}.$$

A final step sets necessarily $s=n$ and hence $c_{nm}^{(k)} = h_m^{(k)}\, b_{mn}^{(k)} h_n^{-1}$ for all $m \geq k$ and $n > 0$, which concludes the proof. ∎

[1] The relation in Lemma 2.1 is well-known for $k=1$, but Lemma 2.1 for $k>1$ is not simply a recursive (or nested) version of this relation.

2.2. Special instances of the Jacobi polynomials. The Jacobi polynomials are commonly used with specific combinations of the parameters α and β, and often with different standardizations. For simplicity we will use the following form for a special orthogonal polynomial

$$Q_n^{(\alpha,\beta)}(x) = g_n^{(\alpha,\beta)} P_n^{(\alpha,\beta)}(x), \tag{2.28}$$

where $g_n^{(\alpha,\beta)}$ is a scaling function. The boundary values of Q_n will depend on g_n, and the basis $\{Q_n\}$ is obviously orthogonal with weight $\omega^{(\alpha,\beta)}$, which will normally be abbreviated as simply ω if it is possible to avoid confusion. We will also normally drop the (α,β) superscript on the special polynomials.

The orthogonal polynomials $\boldsymbol{Q} = (Q_0, Q_1, \ldots)^T$ need to take the function g_n into account when forming the recursion relations from Sec. 2.1. For example, for (2.5) we get

$$x\boldsymbol{Q} = A^T\boldsymbol{Q}, \tag{2.29}$$

where the matrix operator A with components $a_{mn} = (g_m^{(\alpha,\beta)})^{-1} a_{mn}^{(\alpha,\beta)} g_n^{(\alpha,\beta)}$ now has been defined to include the scaling function. All the other recursion relations and inner product equalities in Sec. 2.1, like (2.10), (2.12), (2.20), (2.25), (2.15), (2.21), (2.22) and (2.23) are used exactly as they stand simply by replacing the components P_m with Q_m and using scaled matrices and normalization factor $h_m^{(k)} = (g_m^{(\alpha,\beta)})^2 h_m^{(k,\alpha,\beta)}$. All the matrix operators that belong to a specific family $\{Q_n\}$ are in what follows written without the (α,β) superscript, whereas the Jacobi operators maintain theirs. Hence $A = (a_{mn})$ and $B = (b_{mn})$ will refer to the specific operators for a basis family that include g_n.

The basis functions $\boldsymbol{Q}_N = (Q_0, Q_1, \ldots, Q_N)^T$ form a discrete function space $\mathrm{Q}_N = \mathrm{span}\{Q_n\}_{n=0}^N$, and a function $u(x) \in \mathrm{Q}_N$ will be approximated as the truncated

$$u(x) = \sum_{n=0}^{N} \hat{u}_n Q_n(x), \tag{2.30}$$

where $\hat{u}_n = (u, Q_n)_w / h_n$ for $n = 0, 1, \ldots, N$.

2.3. Ultraspherical polynomials. The ultraspherical polynomials are defined as Jacobi polynomials with only one parameter $\alpha = \beta$ (see, e.g., Sec. 4.2.3 of [22])

$$Q_n^{(\alpha)}(x) = g_n^{(\alpha)} P^{(\alpha,\alpha)}(x), \quad \alpha > -1, \tag{2.31}$$

and normally (see [22]) the scaling factor in use is $g_n^{(\alpha)} = \frac{(2\alpha+1)_n}{(\alpha+1)_n}$. However, the regular ultraspherical polynomials have boundary values that make them slightly awkward to use with spectral Galerkin methods, and we will here follow Doha [9] and scale the ultraspherical polynomials as

$$Q_n^{(\alpha)}(x) = \frac{\Gamma(n+1)}{(\alpha+1)_n} P_n^{(\alpha,\alpha)}(x), \tag{2.32}$$

where the scaling factor corresponds to $g_n^{(\alpha)} = 1/P_n^{(\alpha,\alpha)}(1)$, such that

$$Q_n^{(\alpha)}(\pm 1) = (\pm 1)^n. \tag{2.33}$$

Legendre and Chebyshev polynomials of the first kind are ultraspherical polynomials with the scaling used in (2.32) and $\alpha = 0$ and $-1/2$, respectively. Chebyshev polynomials

Family	Ultraspherical $Q_n^{(\alpha)}$	Legendre $L_n = Q^{(0)}$	Cheb. 1st $T_n = Q_n^{(-1/2)}$	Cheb. 2nd $U_n = (n+1)Q_n^{(1/2)}$
g_n	$\frac{\Gamma(n+1)}{(\alpha+1)_n}$	1	$\frac{\Gamma(n+1)}{(1/2)_n}$	$\frac{\Gamma(n+2)}{(3/2)_n}$
$a_{n-1,n}$	$\frac{n}{2n+2\alpha+1}$	$\frac{n}{2n+1}$	$\frac{1}{2}$	$\frac{1}{2}$
$a_{n+1,n}$	$\frac{(n+2\alpha+1)}{(2n+2\alpha+1)}$	$\frac{n+1}{2n+1}$	$\frac{c_n}{2}$	$\frac{1}{2}$
$b_{n-1,n}$	$-\frac{n}{(n+2\alpha)(2n+2\alpha+1)}$	$-\frac{1}{2n+1}$	$-\frac{1}{2(n-1)}$	$-\frac{1}{2(n+1)}$
$b_{n+1,n}$	$\frac{(n+2\alpha+1}{2n+2\alpha+1)(n+1)}$	$\frac{1}{2n+1}$	$\frac{c_n}{2(n+1)}$	$\frac{1}{2(n+1)}$
$h_n^{(k)}$	$\frac{2^{2\alpha+1}(n!)^2\Gamma^2(\alpha+1)(n+2\alpha+k)!}{(2n+2\alpha+1)(n-k)!\Gamma^2(n+2\alpha+1)}$	$\frac{2(n+k)!}{(n-k)!(2n+1)}$	$\frac{c_{n+k}\pi n\Gamma(n+k)}{2(n-k)!}$	$\frac{\pi\Gamma(n+k+2)}{2(n+1)(n-k)!}$

TABLE 1

Recursion matrices and normalization factors for ultraspherical polynomials. The Pochhammer symbol is represented as $(a)_n = \Gamma(a+n)/\Gamma(a)$.

of the second kind are defined with $\alpha = 1/2$ and a slightly different scaling $g_n^{(1/2)} = (n+1)/P_n^{(1/2,1/2)}(1)$, such that $U_n(x) = (n+1)Q_n^{(1/2)}(x)$. A summary of the recursion matrices and normalization factors for these important families of ultraspherical polynomials is given in Table 1.

3. Spectral differentiation. For $u(x) \in \mathrm{P}_N$ and $u'(x) \in \mathrm{P}_{N-1}$ we have the expansions

$$u(x) = \sum_{n=0}^{N} \hat{u}_n Q_n(x) \in \mathrm{P}_N, \quad \text{and} \quad u'(x) = \sum_{n=0}^{N} \hat{u}_n^{(1)} Q_n(x) \in \mathrm{P}_{N-1}, \tag{3.1}$$

with $\hat{u}_N^{(1)} = 0$. The process of finding $\hat{\boldsymbol{u}}^{(\mathbf{1})} = \{\hat{u}_n^{(1)}\}_{n=0}^N \in \mathbb{R}^{N+1}$ in terms of $\hat{\boldsymbol{u}} = \{\hat{u}_n\}_{n=0}^N \in \mathbb{R}^{N+1}$ is usually termed spectral differentiation in the frequency space. In this section we will use spectral differentiation to introduce the idea of the new Petrov-Galerkin method, which we arrive at in Sec. 3.4.

3.1. Recursive approach. The most common approach for spectral differentiation in the frequency space is to assume

$$\sum_{n=0}^{N} \hat{u}_n^{(1)} Q_n = \sum_{n=0}^{N} \hat{u}_n \partial Q_n, \tag{3.2}$$

and then invoke (2.10) on the left hand side to get

$$\sum_{n=0}^{N} \sum_{s=n-1}^{n+1} \hat{u}_n^{(1)} b_{sn} \partial Q_s = \sum_{n=0}^{N} \hat{u}_n \partial Q_n. \tag{3.3}$$

The method is now usually described through equating coefficients, but we can also take the $L^2_{\omega^{\alpha+1,\beta+1}}[-1,1]$ inner product of (3.3) with $\{\partial Q_m\}_{m=1}^N$ and use orthogonality (2.15) to obtain

$$\sum_{n=0}^{N} b_{mn} \hat{u}_n^{(1)} = \hat{u}_m, \quad \text{for } m = 1, 2, \ldots, N. \tag{3.4}$$

This linear system of equations is not square. However, we can use $\hat{u}_N^{(1)} = 0$ and then solve (3.4) with back substitution (see, e.g., Ch. (3.2.6) of [35]) such that $\hat{u}_{N-1}^{(1)} = \hat{u}_N / b_{N,N-1}$ and

$$\hat{u}_n^{(1)} = \frac{1}{b_{n+1,n}} \left(\hat{u}_{n+1} - b_{n+1,n+1} \hat{u}_{n+1}^{(1)} - b_{n+1,n+2} \hat{u}_{n+2}^{(1)} \right), \quad \text{for } n = N-2, N-3, \ldots, 0. \tag{3.5}$$

Note that the first row of the singular matrix $B \in \mathbb{R}^{N+1 \times N+1}$ is never being used.

3.2. A Galerkin approach. The most obvious Galerkin method for finding $\hat{\boldsymbol{u}}^{(1)}$ is to take the $L^2_\omega[-1,1]$ inner product of (3.2) by $\{Q_m\}_{m=0}^N$

$$\sum_{n=0}^{N} (Q_n, Q_m)_\omega \hat{u}_n^{(1)} = \sum_{n=0}^{N} (\partial Q_n, Q_m)_\omega \hat{u}_n, \quad \forall\, m = 0, 1, \ldots N, \tag{3.6}$$

using orthogonality (2.13) on the left hand side and inverting

$$\hat{u}_m^{(1)} = \frac{1}{h_m} \sum_{n=0}^{N} (\partial Q_n, Q_m)_\omega \hat{u}_n, \quad \forall\, m = 0, 1, \ldots, N, \tag{3.7}$$

which automatically finds also $\hat{u}_N^{(1)} = 0$. The outcome is the same as with the recursive approach, but the differentiation matrix $d_{mn}^{(1)} = (\partial Q_n, Q_m)_\omega$ is badly conditioned, upper triangular and full, and the matrix vector product is costly unless the structure of the matrix is accounted for. On matrix form we can write

$$\hat{\boldsymbol{u}}^{(1)} = \underline{D}^{(1)} \hat{\boldsymbol{u}}, \tag{3.8}$$

where $\underline{d}_{mn}^{(1)} = (\partial Q_n, h_m^{-1} Q_m)_\omega$, $\underline{D}^{(1)} = (\underline{d}_{mn}^{(1)})_{m,n=0}^N \in \mathbb{R}^{N+1 \times N+1}$ is the spectral differentiation matrix, and $D^{(1)} = (d_{mn}^{(1)})_{n,m=0}^N \in \mathbb{R}^{N+1 \times N+1}$.

3.3. The Integration Preconditioner (IP) approach. The IP approach [8, 7] is to invoke the recursion (2.10) directly on (3.8), which is achieved by multiplying (3.8) from the left by the square matrix $B_{[1]} \in \mathbb{R}^{N+1 \times N+1}$ to get

$$B_{[1]} \hat{\boldsymbol{u}}^{(1)} = I_{[1]} \hat{\boldsymbol{u}}, \tag{3.9}$$

where $B_{[1]} \underline{D}^{(1)} = I_{[1]}$, I is the identity matrix, and the square bracket notation on $B_{[k]}$ and $I_{[k]}$ (which we get from [8]) is used to indicate that the first k rows of the matrix are set to zero. If the first row of (3.9) is ignored, the matrix equation can be solved with back substitution for the first N components of $\hat{\boldsymbol{u}}^{(1)}$, and the solution algorithm becomes identical to Eq. (3.5). Like for B, the fact that $B_{[1]}$ is singular demands special attention and complicates the description of the method, see [8, 7, 21].

From a Galerkin perspective, we can get a further understanding of the IP method if we first rewrite (3.6) by dividing each row of both sides by the normalization factor h_m

$$\sum_{n=0}^{N} (Q_n, h_m^{-1} Q_m)_\omega \hat{u}_n^{(1)} = \sum_{n=0}^{N} (\partial Q_n, h_m^{-1} Q_m)_\omega \hat{u}_n, \tag{3.10}$$

and then apply the preconditioner $B_{[1]}$ from the left. The action of the preconditioner is then simply to invoke (2.10) and to replace a scaled test function $\tilde{Q}_m = h_m^{-1} Q_m$ by component m of the matrix vector product $B_{[1]}\tilde{\boldsymbol{Q}}_N$. With notation $\rho_{N,m} = (B_{[1]}\tilde{\boldsymbol{Q}}_N)_m$, we get that

$$\sum_{n=0}^{N} (Q_n, \rho_{N,m})_\omega \hat{u}_n^{(1)} = \sum_{n=0}^{N} (\partial Q_n, \rho_{N,m})_\omega \hat{u}_n, \tag{3.11}$$

where the matrix on the right hand side is the diagonal $I_{[1]}$ and the matrix on the left is $(B_{[1]})_{mn} = (Q_n, \rho_{N,m})_\omega$. As such the IP method can be interpreted as a variational method that is using the test function basis $\{\rho_{N,m}\}_{m=1}^{N}$ for the N unknowns $\{\hat{u}_n\}_{n=0}^{N-1}$ of the trial basis $\{Q_n\}_{n=0}^{N-1}$. Again, the mapping of indices $1, 2, \ldots, N$ for the rows of B to indices $0, 1, \ldots, N-1$ of $\hat{\boldsymbol{u}}^{(\mathbf{1})}$ requires special attention.

3.4. A new Petrov Galerkin approach. We will now consider an alternative Petrov-Galerkin approach for finding specifically $\hat{\boldsymbol{u}}^{(\mathbf{1})}$, and in general $\hat{\boldsymbol{u}}^{(\boldsymbol{k})}$, from $\hat{\boldsymbol{u}}$. To set the stage we consider the specific version of the orthogonality equation (2.15) after dividing both sides by the normalization factor, shifting the m-index to nonzero values and rearranging

$$\left(\partial^k Q_n, \frac{(1-x^2)^k}{h_{m+k}^{(k)}} \partial^k Q_{m+k} \right)_{\omega^{(\alpha,\beta)}} = \delta_{m+k,n}, \quad \text{for} \quad m, n \geq 0. \tag{3.12}$$

From (3.12) we realise that a test function defined as

$$\phi_m^{(k)} = \frac{(1-x^2)^k}{h_{m+k}^{(k)}} \partial^k Q_{m+k} \quad \in \mathrm{P}_{m+2k}, \quad \text{for} \quad m \geq 0, \tag{3.13}$$

would, for the corresponding (the same parameters α, β) orthogonal trial basis $\{Q_n\}$, lead to a k'th order differentiation matrix with one single constant upper diagonal

$$(\partial^k Q_n, \phi_m^{(k)})_{\omega^{(\alpha,\beta)}} = \delta_{m+k,n}. \tag{3.14}$$

This will now be utilized for finding $\hat{\boldsymbol{u}}^{(\boldsymbol{k})}$ from $\hat{\boldsymbol{u}}$.

For simplicity we will first consider $\hat{\boldsymbol{u}}^{(\mathbf{1})}$, and start by multiplying (3.2) with the test function $\phi_m^{(1)}$ and the weight $\omega^{(\alpha,\beta)}$, and then integrate over the domain to obtain

$$\frac{1}{h_{m+1}^{(1)}} \sum_{n=0}^{N} (Q_n, \partial Q_{m+1})_{\omega^{(\alpha+1,\beta+1)}} \hat{u}_n^{(1)} = \sum_{n=0}^{N} (\partial Q_n, \phi_m^{(1)})_{\omega^{(\alpha,\beta)}} \hat{u}_n. \tag{3.15}$$

In order for this to be a square and well defined system of equations, we let $m = 0, 1, \ldots, N$. The right hand side matrix is already known from (3.14). Furthermore, the inner product matrix on the left $(Q_n, \partial Q_{m+1})_{\omega^{(\alpha+1,\beta+1)}} = h_{m+1}^{(1)} b_{m+1,n}$, which we get from (2.21) (using $k = l = 1$). The common factor $h_{m+1}^{(1)}$ falls out and Eq. (3.15) becomes

$$\sum_{n=0}^{N} b_{m+1,n} \hat{u}_n^{(1)} = \sum_{n=0}^{N} \delta_{m+1,n} \hat{u}_n. \tag{3.16}$$

On matrix form we get

$$B_{(1)} \hat{\boldsymbol{u}}^{(\mathbf{1})} = I_{(1)} \hat{\boldsymbol{u}}, \tag{3.17}$$

where the upper shift matrix $I_{(1)} = (\delta_{m+1,n})_{m,n=0}^{N} \in \mathbb{R}^{N+1\times N+1}$, and the mass matrix $B_{(1)} = (b_{m+1,n})_{m,n=0}^{N} \in \mathbb{R}^{N+1\times N+1}$ is the upper triangular

$$B_{(1)} = \begin{bmatrix} b_{1,0} & b_{1,1} & b_{1,2} & 0 & 0 & \cdots & 0 \\ 0 & b_{2,1} & b_{2,2} & b_{2,3} & 0 & \cdots & 0 \\ 0 & 0 & b_{3,2} & b_{3,3} & b_{3,4} & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & b_{N-1,N-2} & b_{N-1,N-1} & b_{N-1,N} \\ 0 & 0 & \cdots & \cdots & 0 & b_{N,N-1} & b_{N,N} \\ 0 & 0 & \cdots & \cdots & \cdots & 0 & b_{N+1,N} \end{bmatrix}. \tag{3.18}$$

Remark 3.1. For index shifted matrices, like $B_{(1)}$ and $I_{(1)}$, we will in this paper use a special subscript notation with parenthesis to indicate the shift: $A_{(p,q)} = (a_{m+p,n+q})_{m,n=0}^{\infty}$. If only the row is shifted, we will use only one number: $A_{(p)} = (a_{m+p,n})_{m,n=0}^{\infty}$. The notation applies both to infinite-dimensional matrix operators and finite-dimensional matrices.

We note that the mass matrix $B_{(1)}$ is just a shifted version of B, where row 0 has been excluded. Hence we are back at more or less exactly the same solution as obtained with explicit use of an integration preconditioner, which was also found to be exactly the same as the recursive approach. Apparently, nothing has been gained, and nothing has been lost. However, the Petrov-Galerkin approach suggested here has a slight advantage. The mass matrix $B_{(1)}$ is invertible and the condition $\hat{u}_N^{(1)} = 0$ is naturally part of the solution. Another advantage is that we can describe this Petrov Galerkin approach very compactly: with $u_N \in \mathrm{Q}_N$ find $u \in \mathrm{Q}_N$ such that

$$(u - u_N', v)_\omega = 0, \quad \forall\, v \in \mathrm{span}\{\phi_n^{(1)}\}_{n=0}^{N}, \tag{3.19}$$

and the recursive solution from back substitution then falls out naturally, with no tweaking or mapping of indices whatsoever. Regarding the function space used for the test function in (3.19), we note that $\phi_n^{(1)}(x)$ satisfies homogeneous Dirichlet boundary conditions on both sides of the domain. An appropriate test space is thus $\mathrm{V}_N^{(1)} = \{v \in \mathrm{P}_N \,|\, v(\pm 1) = 0\}$ of dimension $N-1$, which is spanned by the basis $\{\phi_n^{(1)}\}_{n=0}^{N-2}$. To get the correct dimension, the space used in (3.19) is thus $\mathrm{V}_{N+2}^{(1)} = \mathrm{span}\{\phi_n^{(1)}\}_{n=0}^{N}$.

We can proceed exactly the same way for higher order spectral differentiation. For second order differentiation we use the basis function $\phi_m^{(2)}$, which satisfies the four conditions $\phi_m^{(2)}(\pm 1) = \partial\phi_m^{(2)}(\pm 1) = 0$. An appropriate space for $\phi_m^{(2)}$ is thus the biharmonic $\mathrm{V}_N^{(2)} = \{v \in \mathrm{P}_N | v(\pm 1) = v'(\pm 1) = 0\}$ of dimension $N-3$. In general, the basis function $\phi_m^{(k)}$ will satisfy $2k$ boundary conditions, and for k'th order an appropriate space is thus

$$\mathrm{V}_N^{(k)} = \{v \in \mathrm{P}_N \,|\, \partial^n v(\pm 1) = 0, \forall\, n = 0, 1, \ldots, k-1\}, \tag{3.20}$$

of dimension $N - 2k + 1$. For k'th order spectral differentiation the problem becomes: with $u_N \in \mathrm{Q}_N$ find $u \in \mathrm{Q}_N$ such that

$$(u - \partial^k u_N, v)_{\omega^{(\alpha,\beta)}} = 0, \quad \forall\, v \in \mathrm{V}_{N+2k}^{(k)} = \mathrm{span}\{\phi_n^{(k)}\}_{n=0}^{N}. \tag{3.21}$$

Following the same approach as above for $k = 1$ we get that

$$\frac{1}{h^{(k)}_{m+k}} \sum_{n=0}^{N} (Q_n, \partial^k Q_{m+k})_{\omega^{(\alpha+k,\beta+k)}} \hat{u}^{(k)}_n = \sum_{n=0}^{N} \delta_{m+k,n} \hat{u}_n, \tag{3.22}$$

or

$$B^{(k)}_{(k)} \hat{\boldsymbol{u}}^{(\boldsymbol{k})} = I_{(k)} \hat{\boldsymbol{u}}. \tag{3.23}$$

Here $I_{(k)} \in \mathbb{R}^{N+1 \times N+1}$ and the mass matrix defined as $B^{(l)}_{(k)} = (b^{(l)}_{m+k,n})^N_{m,n=0} \in \mathbb{R}^{N+1 \times N+1}$ with $k = l$ (which we get from (2.21)), is upper triangular, and not to be confused with the l'th matrix power of $B_{(k)}$. Similarly, the IP method for k'th order spectral differentiation is

$$B^k_{[k]} \hat{\boldsymbol{u}}^{(\boldsymbol{k})} = I_{[k]} \hat{\boldsymbol{u}}, \tag{3.24}$$

which is basically the same linear algebra system as (3.23). In fact, we can get back to (3.23), except from the last k rows that will be zero, by multiplying (3.24) from the left by $I_{(k)}$.

Remark 3.2. With the new recursion (2.25) and index shifting it is evident that we can also write the new test function $\phi^{(k)}_m$ as

$$\phi^{(k)}_m = (B^k \tilde{\boldsymbol{Q}})_{m+k} = (B^{(k)}_{(k)} \tilde{\boldsymbol{Q}})_m \quad \in \mathrm{P}_{m+2k} \quad \text{for} \quad m \geq 0, \tag{3.25}$$

which highlights a link to the other methods described in this section, through the recursive matrix operator B.

There is a minor technical difference between the Petrov-Galerkin method described in this section and the IP method, even though they here lead to exactly the same result. For finite-dimensional systems IP corresponds to using a test function $\rho^{(k)}_{N,m} = (B^k \tilde{\boldsymbol{Q}}_N)_m \in \mathrm{P}_{\min(m+k,N)}$, with the finite matrix B^k and vector $\tilde{\boldsymbol{Q}}_N$, instead of the non-truncated $\phi^{(k)}_m = (B^k \tilde{\boldsymbol{Q}})_m \in \mathrm{P}_{m+k}$. This leads for IP to *test functions that are not all in the same space*, and there will be a difference from the PG method in the k highest wavenumbers. However, since spectral differentiation and thus (3.21) assumes $u_N \in \mathrm{Q}_N$, orthogonality makes all terms involving Q_{N+k} for $k > 0$ disappear and this minor detail is thus not significant for spectral differentiation, where $\{\hat{u}^{(k)}_n\}^N_{n=N-k+1} = 0$. However, the difference will matter for regular differential equations, as we will see in Sec. 4.

3.5. Spectral differentiation with ultraspherical polynomials. For ultraspherical polynomials the basis function used for $V^{(k)}_{N+2k}$ is

$$\phi^{(k,\alpha)}_n = \frac{(1-x^2)^k \partial^k Q^{(\alpha)}_{n+k}}{h^{(k)}_{n+k}} = (B^k \tilde{\boldsymbol{Q}}^{(\alpha)})_{n+k}. \tag{3.26}$$

For ultraspherical polynomials defined as (2.32) it can be shown that

$$\frac{b_{n+1,n}}{h_n} = -\frac{b_{n+1,n+2}}{h_{n+2}} = -\frac{\Gamma(n+2\alpha+2)}{2^{2\alpha+1}\Gamma^2(\alpha+1)\Gamma(n+2)}, \tag{3.27}$$

Family	$\phi_n^{(1)}$	$\phi_n^{(2)}$
Legendre	$\frac{1}{2}(L_n - L_{n+2})$	$\frac{1}{2(2n+3)}\left(L_n - \frac{2(2n+5)}{2n+7}L_{n+2} + \frac{2n+3}{2n+7}L_{n+4}\right)$
Cheb. 1st	$\frac{1}{\pi(n+1)}(T_n - T_{n+2})$	$\frac{1}{2\pi(n+1)(n+2)}\left(T_n - \frac{2(n+2)}{n+3}T_{n+2} + \frac{n+1}{n+3}T_{n+4}\right)$
Cheb. 2nd	$\frac{1}{\pi}\left(\frac{U_n}{n+1} - \frac{U_{n+2}}{n+3}\right)$	$\frac{1}{2\pi(n+1)(n+2)}\left(U_n - \frac{2(n+1)}{n+4}U_{n+2} + \frac{(n+1)(n+2)}{(n+3)(n+4)}U_{n+4}\right)$

TABLE 2
Basis functions $\phi_n^{(1)}$ and $\phi_n^{(2)}$ on expanded form for Legendre and Chebyshev polynomials.

and the basis functions for $k = 1$ and 2 can be written as

$$\phi_n^{(1)} = \frac{b_{n+1,n}}{h_n}\left(Q_n^{(\alpha)} - Q_{n+2}^{(\alpha)}\right), \tag{3.28}$$

$$\phi_n^{(2)} = \frac{b_{n+2,n}^{(2)}}{h_n}\left(Q_n^{(\alpha)} - (1+c_n)Q_{n+2}^{(\alpha)} + c_n Q_{n+4}^{(\alpha)}\right), \tag{3.29}$$

where $c_n = \frac{2n+2\alpha+3}{2n+2\alpha+7}$. For the Legendre and Chebyshev families the basis functions of lowest order are given in Table 2. It is interesting to note that, for both Legendre and Chebyshev of the first kind, $\phi_n^{(1)}$ and $\phi_n^{(2)}$ correspond to scaled versions of the well known Dirichlet and biharmonic basis functions of Shen [31, 32]. Also, the basis function (3.28) is a scaled version of the function used by Doha [9].

4. Sparse methods for differential equations. For spectral Galerkin methods the trial functions are constructed as linear combinations of orthogonal polynomials, in order to satisfy a given problems boundary conditions. Since we now know that the test function $\phi_n^{(k)}$ turns the k'th order differentiation matrix $(\partial^k Q_n, \phi_m^{(k)})_\omega$ into a matrix with one single upper diagonal, this means that $\phi_m^{(k)}$ will make the differentiation matrix of any spectral Galerkin problem sparse and banded. Because of (2.23) we also know that $\phi_n^{(k)}$ will make any lower order differentiation matrix with polynomial coefficients $(\partial^{k-l} Q_n, x^q \phi_m^{(k)})_\omega$ sparse and banded.

4.1. The new sparse method. We consider a linear differential equation of the form

$$\sum_{l=0}^{k} p_l(x)\partial^{k-l} u = f, \quad \text{for } x \in [-1, 1], \tag{4.1}$$

subject to the k homogeneous boundary conditions

$$\mathcal{T}^{(l)} u = 0, \quad l = 0, 1, \ldots, k-1. \tag{4.2}$$

The coefficients $\{p_l(x)\}_{l=0}^k$ are polynomials of different degree, and there are k boundary conditions (Dirichlet, Neumann, etc), specified at either side of the domain. If required, inhomogeneous boundary conditions can easily be incorporated using a lifting technique [2], which does not affect any of the coefficient matrices derived below.

We choose a trial space $\mathrm{S}_N^{(k)} = \{v \in \mathrm{P}_N \,|\, \mathcal{T}^{(l)} v = 0, l = 0, 1, \ldots, k-1\}$ of dimension $M + 1 = N - k + 1$. The new Petrov-Galerkin method is to find $u \in \mathrm{S}_N^{(k)}$ such that

$$\sum_{l=0}^{k} \left(p_l \partial^{k-l} u, v\right)_\omega = (f, v)_\omega, \quad \forall\, v \in \mathrm{V}_{N+k}^{(k)}, \tag{4.3}$$

using basis $\boldsymbol{\phi}^{(k)} = \{\phi_m^{(k)}\}_{m=0}^M$ for the test space $\mathrm{V}_{N+k}^{(k)}$ (see (3.20)). A basis for $\mathrm{S}_N^{(k)}$ is $\boldsymbol{\psi} = \{\psi_n\}_{n=0}^M$, where ψ_n is constructed from a small number of neighbouring orthogonal basis functions. For simplicity we will write the trial functions as

$$\boldsymbol{\psi} = K\boldsymbol{Q}_N, \quad \text{and thus} \quad u(x) = \sum_{n=0}^{M} \hat{u}_n (K\boldsymbol{Q}_N)_n \in \mathrm{P}_N, \tag{4.4}$$

where $K = (\kappa_{ij}) \in \mathbb{R}^{M+1\times N+1}$ is a strictly banded stencil matrix, normally with lower bandwidth 0 and upper k. The stencil matrix is used in order to derive one set of algebraic equations to be used for different problems and trial functions, satisfying different boundary conditions.

Remark 4.1. An uncommon feature of (4.3) is that there is a discrepancy in polynomial order between the m'th trial function $\psi_m \in \mathrm{P}_{m+k}$ and the test function $\phi_m^{(k)} \in \mathrm{P}_{m+2k}$, and we search for a solution in P_N, using the slightly larger test space P_{N+k}. The feature stems from the different number of boundary conditions used in test and trial spaces.

Remark 4.2. For the basis $\boldsymbol{\phi}^{(k)}$ the matrix $K_s = B_{(k)}^{(k)} H^{-1} \in \mathbb{R}^{M+1\times N+k+1}$ can be interpreted as a stencil matrix, since

$$\boldsymbol{\phi}^{(k)} = K_s \boldsymbol{Q}_{N+k}. \tag{4.5}$$

4.1.1. Constant coefficients. Assume that all the coefficients p_l are constant, and that $p_0 = 1$. Insert for test and trial functions in the bilinear part of (4.3)

$$(\partial^{k-l} u, v)_\omega = \sum_{n=0}^{M}\sum_{s=0}^{N} (\partial^{k-l} Q_s, \phi_m^{(k)})_\omega \kappa_{ns} \hat{u}_n, \quad \text{for } m = 0, 1, \ldots, M, \tag{4.6}$$

and use the inner products from Eqs. (2.21) and (2.13). Equation (4.3) on matrix form thus becomes

$$\left(I_{(k)} + \sum_{l=1}^{k} p_l B_{(k)}^{(l)} \right) K^T \hat{\boldsymbol{u}} = \tilde{\boldsymbol{f}}, \tag{4.7}$$

where the matrices $I_{(k)}$ and $B_{(k)}^{(l)}$ are of shape $\mathbb{R}^{M+1\times N+1}$. The right hand side $\tilde{\boldsymbol{f}} = \{(f, \phi_m^{(k)})_\omega\}_{m=0}^M \in \mathbb{R}^{M+1}$. However, using (3.25) we can also write

$$\tilde{\boldsymbol{f}} = B_{(k)}^{(k)} \hat{\boldsymbol{f}}, \tag{4.8}$$

where $\hat{\boldsymbol{f}} = \{\hat{f}_m\}_{m=0}^{N+k}$, $\hat{f}_m = (f, \tilde{Q}_m)_\omega$ and $B_{(k)}^{(k)} \in \mathbb{R}^{M+1\times N+k+1}$, which is of different shape from $B_{(k)}^{(k)}$ on the left hand side of (4.7). Simply restricting the right hand side as $f \in \mathrm{P}_N$ (like the trial function), we get $\hat{\boldsymbol{f}} = \{\hat{f}_m\}_{m=0}^N$ and can use the same $B_{(k)}^{(k)} \in \mathbb{R}^{M+1\times N+1}$ on the right hand side as on the left. With this restriction we get for Chebyshev polynomials and constant coefficients more or less the same method that is defined as the "quasi-inverse" method with Galerkin trial functions by Julien and Watson [23]. But here generalised to all Jacobi polynomials and wrapped up in a Petrov-Galerkin formulation.

4.1.2. Variable coefficients. Assume now that $p_l(x) = x^q$, with integer $q > 0$, such that we need to compute $(\partial^{k-l}u, x^q v)_\omega$ for some $l \leq k$. Inserting for test and trial function we get

$$(\partial^{k-l}u, x^q v)_\omega = \sum_{n=0}^{M}\sum_{s=0}^{N}(\partial^{k-l}Q_s, x^q\phi_m^{(k)})_\omega \kappa_{ns}\hat{u}_n, \quad \text{for } m = 0, 1, \ldots, M, \tag{4.9}$$

where the inner product matrix is computed using (2.23) as

$$(\partial^{k-l}Q_s, x^q\phi_m^{(k)})_\omega = \sum_{p=m-q}^{m+q} \underline{a}^{(k,q)}_{m+k,p+k} b^{(l)}_{p+k,s}. \tag{4.10}$$

On matrix form we get

$$(\partial^{k-l}u, x^q v)_\omega = L^{(k,q,l)}K^T\hat{\boldsymbol{u}}, \tag{4.11}$$

where the matrix $L^{(k,q,l)} = \underline{A}^{(k,q)}_{(k,k)}B^{(l)}_{(k)} \in \mathbb{R}^{M+1\times N+1}$, $\underline{A}^{(k,q)}_{(k,k)} \in \mathbb{R}^{M+1\times N+1}$ and $B^{(l)}_{(k)} \in \mathbb{R}^{N+1\times N+1}$. Note that $L^{(k,q,l)}$ has bandwidth $1+2(l+q)$, with the lower bandwidth $q+l-k$ and upper bandwidth $q+l+k$. The bandwidth $1+2(q+l)$ was given also in Theorem 2.1 of [7], and it follows since both $\underline{A}$ and B are tri-diagonal matrices and there is a total of $q+l$ matrix powers. The shift of the bandwidth from the centre is special for the current method and due to the row-shifted $B^{(l)}_{(k)}$. For the Galerkin method there is also an additional problem dependent bandwidth in (4.11) due to the stencil matrix K.

Remark 4.3. For any special orthogonal basis described as (2.28), the matrix $L^{(k,q,l)}$ can be computed explicitly simply from the Jacobi matrix and vector components $a^{(\alpha,\beta)}_{mn}, b^{(\alpha,\beta)}_{mn}, h^{(k,\alpha,\beta)}_m$ and $\psi^{(k,\alpha,\beta)}_m$ given in Sec. 2.1, and the scaling functions $g^{(\alpha,\beta)}_m$.

Remark 4.4. Any equation that can be written as Eq. (4.1) leads to an algebraic problem where the coefficient matrix is a sum of the strictly banded matrices $L^{(k,q,l)}$. This includes also constant coefficient matrices, since $\underline{A}^{(k,0)}_{(k,k)} = I$. On this form Eq. (4.7) becomes

$$\sum_{l=0}^{k} p_l L^{(k,0,l)}K^T\hat{\boldsymbol{u}} = \tilde{\boldsymbol{f}}, \tag{4.12}$$

where $L^{(k,0,0)} = I_{(k)}$ and $p_0 = 1$.

4.1.3. The linear form and numerical implementations. For a pure spectral Petrov-Galerkin method the inner product integrals in (4.3) will be computed exactly, leading to analytical coefficient matrices that are sums of $L^{(k,q,l)}$ matrices. A discrete inner product in a space with $N+1$ quadrature points is represented as $(\cdot,\cdot)_{N,\omega}$ and (4.3) thus becomes

$$\sum_{l=0}^{k} \left(p_l\partial^{k-l}u, v\right)_{N+k,\omega} = (f, v)_{N+k,\omega}, \tag{4.13}$$

using the $N+k+1$ quadrature points of the test space $\mathrm{V}^{(k)}_{N+k}$. Since Gaussian quadrature of order $N+k$ is exact for all polynomial integrands $\in \mathrm{P}_{2(N+k)+1}$, the constant coefficient

$L^{(k,0,l)}$ will be exact also for a numerical approach. However, since the integrand of the variable coefficient $L^{(k,q,l)}$ are polynomials of order less than or equal to $2N+k+q$, a numerical approach using $N+k+1$ quadrature points will only be exact for $q \leq 2k+1-l$. Naturally, if q exceeds this limit we can simply increase the number of quadrature points correspondingly.

A numerical approximation of the right hand side of (4.8) will use quadrature for the integral and interpolation of $f(x)$

$$\tilde{\boldsymbol{f}} = B^{(k)}_{(k)} \overline{\boldsymbol{f}}, \tag{4.14}$$

where the interpolation coefficients $\overline{\boldsymbol{f}} = \{\overline{f}_{m,N+k}\}_{m=0}^{N+k} \in \mathbb{R}^{N+k+1}$ are defined as $\overline{f}_{m,N+k} = (I_{N+k} f, \tilde{Q}_m)_{N+k,\omega}$, and the interpolation operator is defined such that $I_{N+k} f(x_j) = f(x_j)$ for all quadrature points $\{x_j\}_{j=0}^{N+k}$.

4.1.4. Alternative new sparse method. With a minor modification we can reformulate the method presented in Eq. (4.3), such that the m'th test and trial functions both have the same polynomial order $m+k$, and test and trial spaces both are in P_N. In order to achieve this we can simply pull k polynomial orders from the test function and put them into the weight. For even order equations a Petrov-Galerkin formulation may then be to find $u \in \mathrm{S}_N^{(k)}$ such that

$$\sum_{l=0}^{k} \left(p_l \partial^{k-l} u, v\right)_{\omega^{(\alpha+k/2,\beta+k/2)}} = (f, v)_{\omega^{(\alpha+k/2,\beta+k/2)}}, \quad \forall\, v \in \mathrm{V}_N^{(k/2)}. \tag{4.15}$$

In order to remain completely identical to (4.3) the additional weight will now have to be removed from the test function, and we should use

$$\overline{\phi}_m^{(k,\alpha,\beta)} = \frac{\phi_m^{(k,\alpha,\beta)}}{(1-x^2)^{k/2}} = \frac{(1-x^2)^{k/2} \partial^k Q_{m+k}^{(\alpha,\beta)}}{h_{m+k}^{(k,\alpha,\beta)}}, \tag{4.16}$$

and $\mathrm{V}_N^{(k/2)} = \mathrm{span}\{\overline{\phi}_m^{(k,\alpha,\beta)}\}_{m=0}^{M}$. Note that we have simply shuffled the $(1-x^2)^{k/2}$ term around and the variational form (4.15) is still identical to (4.3) and we get exactly the same matrices $L^{(k,q,l)}$ as in Secs. 4.1.1 and 4.1.2. The only difference will be manifested in numerical implementations of $(f,v)_{N,\omega^{(\alpha+k/2,\beta+k/2)}}$, that will naturally make use of $N+1$ quadrature points instead of $N+k+1$. Furthermore, since $\{Q_m^{(\alpha+k/2,\beta+k/2)}\}$ are orthogonal with weight $\omega^{(\alpha+k/2,\beta+k/2)}$, it is more natural to rewrite the test functions $\overline{\phi}_m^{(k,\alpha,\beta)}$ using (2.7) as

$$\overline{\phi}_m^{(k,\alpha,\beta)} = \gamma_m^{(k,\alpha,\beta)} \phi_m^{(k/2,\alpha+k/2,\beta+k/2)}, \tag{4.17}$$

where the scaling function

$$\gamma_m^{(k,\alpha,\beta)} = \frac{\psi_{m+k}^{(k/2,\alpha,\beta)} g_{m+k}^{(\alpha,\beta)} h_{m+k/2}^{(k/2,\alpha+k/2,\beta+k/2)}}{g_{m+k/2}^{(\alpha+k/2,\beta+k/2)} h_{m+k}^{(k,\alpha,\beta)}}. \tag{4.18}$$

Naturally, a numerical implementation of $(f,v)_{N,\omega^{(\alpha+k/2,\beta+k/2)}}$ should use the quadrature points of Jacobi polynomials with parameters $(\alpha+k/2, \beta+k/2)$ instead of (α,β), which is another discrepancy from the original method.

Remark 4.5. Equation (4.15) should only be considered for even order equations. For odd equations we can still modify the test function and weight, but would have to treat α and β separately, and depart from ultraspherical polynomials.

Remark 4.6. For $k = 2$ and $\alpha = \beta = -1/2$ this method corresponds to using Chebyshev polynomials of the first kind for the trial functions and Chebyshev polynomials of the second kind for the test functions, with inner products in $L^2_{\omega^{1/2}}[-1, 1]$ and $\gamma_m^{(2,-1/2,-1/2)} = \frac{1}{m+2}$. A similar approach is used, e.g., by Olver and Townsend [27] and Burns et al. [4].

4.2. Multiple dimensions. The methods described in Secs. 4.1 are all strictly banded and easily extended to multiple dimensions through the use of tensor product methods. Let us for illustration consider the two-dimensional Poisson's equation in Cartesian coordinates

$$\nabla^2 u(x, y) = f(x, y), \tag{4.19}$$

for any type of boundary conditions on the domain $\Omega = [-1, 1]^2$. For the trial function we choose the tensor product space $\mathcal{S} = S_N^{(2)}(x) \otimes S_N^{(2)}(y)$, with basis $\{\psi_m(x)\psi_n(y) \,|\, m, n = 0, 1, \dots M\}$, where $\psi_m(x) = (K_x \boldsymbol{Q}_N)_m$ and $\psi_n(y) = (K_y \boldsymbol{Q}_N)_n$. Here $K_x \in \mathbb{R}^{M+1\times N+1}$ and $K_y \in \mathbb{R}^{M+1\times N+1}$ are stencil matrices determined by the problems boundary conditions in the x and y-directions, respectively. The test space is chosen as $\mathcal{V} = V_{N+2}^{(2)} \otimes V_{N+2}^{(2)} = \text{span}\{\phi_m^{(2)}(x)\phi_n^{(2)}(y) \,|\, m, n = 0, 1, \dots M\}$ and we attempt to find $u \in \mathcal{S}$ such that

$$(\nabla^2 u, v)_\omega = (f, v)_\omega \quad \forall\, v \in \mathcal{V}, \tag{4.20}$$

where the weight $\omega = \omega(x)\omega(y)$ is the product of the weights in the x and y directions. The expansion for the solution is now

$$u(x, y) = \sum_{i=0}^{M}\sum_{j=0}^{M} \hat{u}_{ij}\psi_i(x)\psi_j(y) \in \mathcal{S}, \tag{4.21}$$

with expansion coefficients $\hat{U} = (\hat{u}_{ij}) \in \mathbb{R}^{M+1\times M+1}$. Inserting for test and trial functions it is easy to show that Poisson's equation (4.20) on algebraic form becomes

$$L_x^{(0,0)}\hat{U}{L_y^{(0,2)}}^T + L_x^{(0,2)}\hat{U}{L_y^{(0,0)}}^T = \tilde{F}, \tag{4.22}$$

where $(\tilde{F})_{ij} = (f, \phi_i^{(2)}\phi_j^{(2)})_\omega$ and $L_s^{(q,l)} = L^{(2,q,l)}K_s^T$ for $s \in (x, y)$. We now use the row-major vectorization, or *vec*[2], operation on (4.22) to arrive at

$$\left(L_x^{(0,0)} \otimes L_y^{(0,2)} + L_x^{(0,2)} \otimes L_y^{(0,0)}\right) \text{vec}(\hat{U}) = \left(L_x^{(0,2)} \otimes L_y^{(0,2)}\right) \text{vec}(\tilde{F}), \tag{4.23}$$

where $\otimes$ here represent a tensor product, or Kronecker product, of matrices, $\text{vec}(\hat{U}) \in \mathbb{R}^{(M+1)^2}$ is the column vector obtained by flattening the row-major two-dimensional $\hat{U}$, i.e., $\text{vec}(\hat{U}) = (\hat{u}_{00}, \dots, \hat{u}_{0M}, \hat{u}_{10}, \dots \hat{u}_{1M}, \dots, \dots \hat{u}_{M0}, \dots, \hat{u}_{MM})^T$ and the Kronecker product matrices are all of shape $\mathbb{R}^{(M+1)^2\times(M+1)^2}$.

The Kronecker product method is easily automated, also for higher dimensions, and sparse and strictly banded matrices $L_s^{(q,l)}$ lead to sparse and strictly banded Kronecker product matrices. For a Dirichlet problem using ultraspherical polynomials and $K_s = (\delta_{mn} - \delta_{m+2,n})_{m=0,n=0}^{M,N} \in \mathbb{R}^{M+1\times N+1}$ for both $s \in (x, y)$, the coefficient matrix on the left of (4.23) will have 12 nonzero diagonals.

[2] i.e., $\text{vec}(AUB^T) = (A \otimes B)\text{vec}(U)$ for matrices A, U, B of appropriate shape.

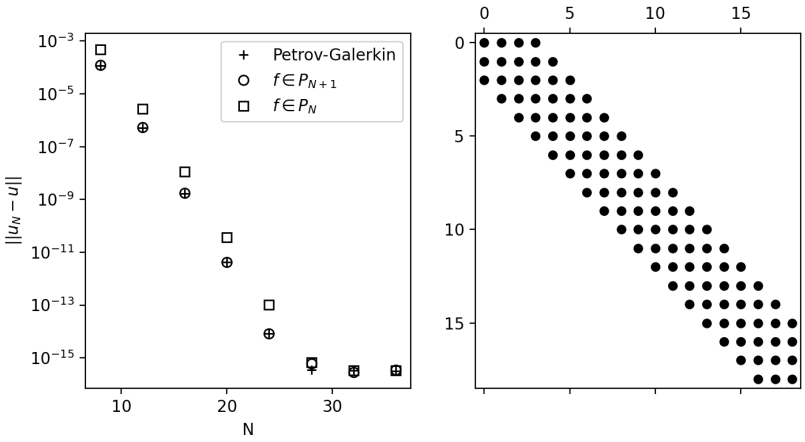

FIG. 1. *Left: The $L^2[-1,1]$ error norm for the solution of Eq. (4.25) using $u(x) = \exp(-0.25x^4)(x+1)$ and three different computations of $\tilde{\boldsymbol{f}}$. Right: Sparsity pattern of the coefficient matrix.*

4.3. Numerical examples.

4.3.1. First order problem. We consider the first order problem

$$u'(x) + \frac{1}{x^2+1}u(x) = s(x), \quad u(-1) = 0, \quad x \in [-1,1], \tag{4.24}$$

using ultraspherical polynomials. For this first order problem we use the trial space $\mathrm{S}_N^{(1)} = \{v \in \mathrm{P}_N | v(-1) = 0\}$ with basis function $\psi_n = Q_n^{(\alpha)} + Q_{n+1}^{(\alpha)}$, corresponding to a stencil matrix $K = (\delta_{mn} + \delta_{m+1,n})_{m=0,n=0}^{N-1,N} \in \mathbb{R}^{N \times N+1}$.

Next, we multiply through with x^2+1 to get only polynomial coefficients, and attempt to find $u \in \mathrm{S}_N^{(1)}$ such that

$$((x^2+1)u', v)_\omega + (u, v)_\omega = (f, v)_\omega, \quad \forall v \in \mathrm{V}_{N+1}^{(1)} = \mathrm{span}\{\phi_n^{(1)}\}_{n=0}^{N-1}, \tag{4.25}$$

where $f(x) = (x^2+1)s(x)$. Inserting for test and trial functions we get

$$(L^{(1,2,0)} + L^{(1,0,0)} + L^{(1,0,1)})K^T \hat{\boldsymbol{u}} = \tilde{\boldsymbol{f}}, \tag{4.26}$$

where the coefficient matrix on the left has 6 nonzero diagonals, with lower bandwidth 2 and upper 3, see Fig. 1. We compute the right hand side both exactly and numerically with either $f(x) \in \mathrm{P}_{N+1}$ or $f(x) \in \mathrm{P}_N$. The latter is computed merely as a curiosity, because it corresponds closely to using the IP method with a Galerkin trial function (see [23]). Note that if $u \in \mathrm{P}_N$, then, due to the polynomial coefficient, the right hand side $f(x)$ will be a polynomial $\in \mathrm{P}_{N+1}$. The larger test space of the current method thus has an advantage here. This is evident in the left hand panel of Figure 1, which shows the $L^2[-1,1]$ error norm $||u_N - u|| = (\int_{-1}^{1}(u_N - u)^2 dx)^{1/2}$ using the manufactured solution $u(x) = \exp(-0.25x^4)(x+1)$ and Chebyshev polynomials of the first kind. We see that for this problem one additional coefficient for $f(x)$ leads to approximately one number extra in accuracy until machine precision is reached.

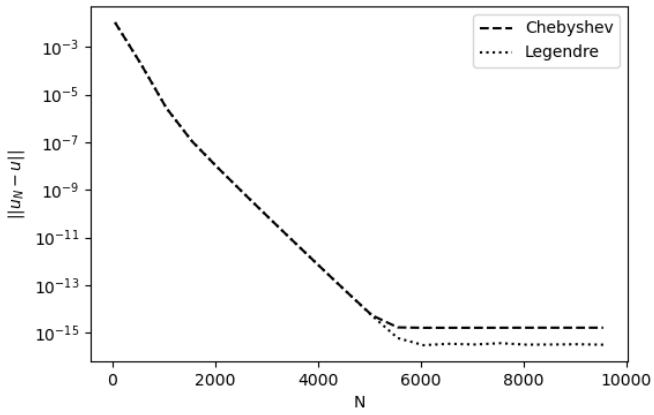

FIG. 2. *The $L^2[-1,1]$ error norm for the solution of Eq. (4.25) using manufactured solution $u(x) = \exp(-\frac{1}{\sqrt{a}}\left(\tan^{-1}(\sqrt{a}x) - \tan^{-1}(\sqrt{a})\right))$ and Legendre (dotted) and Chebyshev (dashed) basis functions.*

The current example is used in slightly different form by Olver and Townsend [27]

$$u'(x) + \frac{1}{ax^2+1}u(x) = 0, \quad u(-1) = 1, \quad x \in [-1,1], \tag{4.27}$$

with the analytical solution $u(x) = \exp(-\frac{1}{\sqrt{a}}\left(\tan^{-1}(\sqrt{a}x) - \tan^{-1}(\sqrt{a})\right))$ and $a = 5 \times 10^4$. We can solve this problem as described above, but need to add one (constant) basis function $\psi_N = Q_0^{(\alpha)} = 1$ to the trial basis and look for the solution

$$u(x) = \sum_{n=0}^{N} \hat{u}_n \psi_n(x). \tag{4.28}$$

We immediately get that $\hat{u}_N = u(-1) = 1$, and solve for the remaining coefficients using the same matrices as before (4.26), only scaled appropriately by a. The right hand side vector $\tilde{f}_n = 0$ for $n = 1, 2, \ldots, N-1$. However, due to the boundary basis and $(u,v)_\omega$ we get the following nonzero term on the right: $\tilde{f}_0 = -(\psi_N, \phi_0^{(1)})_\omega \hat{u}_N = -1$. We solve the problem using both Chebyshev and Legendre polynomials, and the L^2 error norm is shown in Figure 2. Not surprisingly, the problem is resolved to machine precision using approximately 5000 degrees of freedom, which was obtained also by Olver and Townsend.

Remark 4.7. It can be shown with direct computation that the condition number of the coefficient matrix for this problem grows as $\mathcal{O}(N)$ for any ultraspherical polynomials scaled as (2.32).

4.3.2. Second order problem. We consider the Helmholtz problem

$$u''(x) - \mu u(x) = f(x), \quad u(\pm 1) = 0, x \in [-1,1], \tag{4.29}$$

where the constant coefficient $\mu \geq 0$. For this problem we can use the Dirichlet trial space $S_N^{(2)} = \mathrm{V}_N^{(1)}$, with basis $\{\psi_n\}_{n=0}^{N-2}$, and $\psi_n = Q_n^{(\alpha)} - Q_{n+2}^{(\alpha)}$, corresponding to a stencil matrix

$K = (\delta_{mn} - \delta_{m+2,n})_{m=0,n=0}^{N-2,N} \in \mathbb{R}^{N-1\times N+1}$. The Petrov-Galerkin problem is formulated as: find $u \in \mathrm{V}_N^{(1)}$ such that

$$(u'', v)_\omega - \mu(u, v)_\omega = (f, v)_\omega, \quad \forall\, v \in \mathrm{V}_{N+2}^{(2)} = \mathrm{span}\{\phi_m^{(2)}\}_{m=0}^{N-2}. \tag{4.30}$$

Using Eq. (4.7) with $k = 2$, $p_1 = 0$ and $p_2 = -\mu$, we obtain

$$(I_{(2)} - \mu B_{(2)}^{(2)}) K^T \hat{\boldsymbol{u}} = \tilde{\boldsymbol{f}}, \tag{4.31}$$

where the coefficient matrix consists of 4 nonzero diagonals. This sparsity matches the best that has been reported for the Helmholtz problem with Chebyshev polynomials, see [23, 12].

Remark 4.8. Restricted to Chebyshev polynomials, and up to different scaling of the basis functions, this method corresponds to the Petrov-Galerkin method described by Elbarbary [12].

An alternative formulation for this problem according to Sec. 4.1.4 is to find $u \in \mathrm{V}_N^{(1)}$ such that

$$(u'', v)_{\omega^{(\alpha+1)}} - \mu(u, v)_{\omega^{(\alpha+1)}} = (f, v)_{\omega^{(\alpha+1)}}, \quad \forall\, v \in \mathrm{V}_N^{(1)} = \mathrm{span}\{\overline{\phi}_m^{(2,\alpha)}\}_{m=0}^{N-2}. \tag{4.32}$$

This is actually a regular Galerkin method (not Petrov-Galerkin), since the trial and test spaces are the same, and it leads to exactly the same left hand side of the algebraic problem (4.31) as before. The right hand side will differ only for a numerical implementation. For $\alpha = -1/2$ this corresponds to using the trial function $\psi_n = T_n - T_{n+2}$ and test function $\overline{\phi}_m^{(2,-1/2)} = \frac{1}{m+2}\phi_m^{(1,1/2)} = \frac{1}{\pi(m+2)}(\frac{U_m}{m+1} - \frac{U_{m+2}}{m+3})$.

Another second order problem is the Airy differential equation

$$\epsilon u'' - xu = 0, \quad u(-1) = \mathrm{Ai}\left(-\sqrt[3]{\tfrac{1}{\epsilon}}\right), u(1) = \mathrm{Ai}\left(\sqrt[3]{\tfrac{1}{\epsilon}}\right), \tag{4.33}$$

which has the Airy function $u(x) = \mathrm{Ai}\left(\sqrt[3]{\frac{1}{\epsilon}}x\right)$ as solution. Again we follow Olver and Townsend [27] and choose $\epsilon = 10^{-9}$ such that the solution becomes highly oscillatory. Because of the boundary conditions we also need to add two basis functions to the homogeneous trial basis, and use $\psi_{N-1} = \frac{1}{2}(Q_0^{(\alpha)} + Q_0^{(\alpha)})$ and $\psi_N = \frac{1}{2}(Q_0^{(\alpha)} - Q_0^{(\alpha)})$. The linear algebra problem to solve becomes

$$(\epsilon I_{(2)} - L^{(2,1,2)}) K^T \hat{\boldsymbol{u}} = \tilde{\boldsymbol{f}}, \tag{4.34}$$

where $\tilde{f}_j = 0$ for $j = 2, \ldots, N-2$ and, due to the boundary functions, $\hat{u}_{N-1} = u(-1)$, $\hat{u}_N = u(1)$, $\tilde{f}_0 = \sum_{j=N-1}^N (\psi_j, x\phi_0^{(2)})_\omega \hat{u}_j$ and $\tilde{f}_1 = \sum_{j=N-1}^N (\psi_j, x\phi_1^{(2)})_\omega \hat{u}_j$. Figure 3 shows the Airy function and the $L^2[-1,1]$ error using Chebyshev polynomials for basis. The results are similar to Olver and Townsend, and robust for large N due to good conditioning of the matrix.[3] A notable difference from the almost banded matrix obtained by Olver and Townsend is that the coefficient matrix here is strictly banded with 7 nonzero diagonals.

Remark 4.9. Like in the previous section it can be shown with direct computation that the coefficient matrices have condition numbers that are scaling as $\mathcal{O}(N)$ (for large N) for any ultraspherical basis with scaling as (2.32).

[3] In fact, we obtain an $L^2[-1,1]$ error of 1.7×10^{-14} for the overresolved $N = 10^6$.

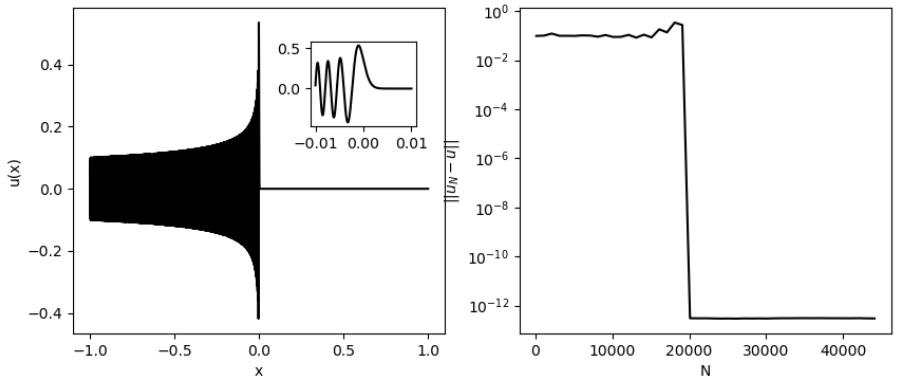

FIG. 3. *The Airy function* $u(x) = Ai\left(\sqrt[3]{\frac{1}{\epsilon}}x\right)$ *with* $\epsilon = 10^{-9}$ *on the left, with inset figure zoomed in on the region* $x \in [-0.01, 0.01]$. *The figure on the right shows the* $L^2[-1,1]$ *error norm using Chebyshev polynomials.*

4.3.3. A comment on sparsity. The test function $\phi_m^{(k)}$ guarantees a sparse and strictly banded differentiation matrix of any order lower than or equal to k. This is a generic sparse approach that applies to all orthogonal polynomials in the Jacobi family, but there is no guarantee that this is the best, or most sparse, solution. Consider, for example, the second order problem in Sec. 4.3.2 with $\mu = 0$. If we reformulate this as a Galerkin problem, using the same space for both test and trial functions, we can find $u \in \mathrm{V}_N^{(1)} = \mathrm{span}\{Q_m - Q_{m+2}\}_{m=0}^{N-2}$ such that

$$(u'', v)_\omega = (f, v)_\omega, \quad \forall\, v \in \mathrm{V}_N^{(1)} = \mathrm{span}\{\phi_m^{(1)}\}_{m=0}^{N-2}. \tag{4.35}$$

Note that we here use $\phi_m^{(1)}$ for the test space instead of $\phi_m^{(2)}$. For a Legendre basis we now obtain a diagonal stiffness matrix (see [31])

$$(\psi_n'', \phi_m^{(1)}) = (m+1)(m+2)\delta_{mn}. \tag{4.36}$$

Since a diagonal stiffness matrix is better than the bi-diagonal $I_{(2)}K^T$ that we found in Sec. 4.3.2, this is clearly a better approach for Legendre polynomials. However, for a Chebyshev basis of the first kind, or any other ultraspherical basis using (2.32), the corresponding stiffness matrix will be upper triangular and full (see, e.g., [32]). We note that the Legendre basis is probably the only ultraspherical basis that can achieve a diagonal stiffness matrix for (4.35), because a trial basis $\{\phi_n\}$ (e.g., with $\phi_n = Q_n - Q_{n+2}$) requires $\{\phi_n''\}$ to be orthogonal to a Dirichlet basis. For Legendre $L_n'' - L_{n+2}'' = (2n+3)L_{n+1}'$, which is orthogonal to the Dirichlet basis $\{(1-x^2)L_{n+1}'\}$. This good fortune stems from the fact that $b_{n+1,n} = -b_{n-1,n}$ (such that $Q_n = b_{n-1,n}(Q_{n-1}' - Q_{n+1}')$), which can only obtained for $\alpha = 1/2$ with the scaling used in (2.32).[4]

[4]The Chebyshev polynomials of second kind have $b_{n+1,n} = -b_{n-1,n}$, but use different scaling such that $\{U_n - U_{n+2}\}$ is not a Dirichlet basis.

5. Conclusions. We have described a generic global spectral Petrov-Galerkin method for linear ordinary differential equations with polynomial coefficients. The method leads to highly sparse and strictly banded matrices, and is as such easy to implement using off-the-shelf linear algebra softwares for banded matrices. Like most efficient methods that have been described for orthogonal polynomials, the method relies on recursion relations of Jacobi polynomials. The method is very easy to describe, because all it takes for a k'th order linear differential equation is the use of a specific test function

$$\phi_n^{(k)} \sim (1-x^2)^k \partial^k Q_{n+k}, \tag{5.1}$$

where $\partial^k = \frac{d^k}{dx^k}$, along with trial functions composed as compact combinations of the specialized Jacobi polynomials Q_n, satisfying a given problems boundary conditions. For efficient implementations we have also described a new recursion relation for Jacobi polynomials

$$(1-x^2)^k \partial^k \boldsymbol{Q} = (C^{(k)})^T \boldsymbol{Q}, \quad k > 0, \tag{5.2}$$

where $\boldsymbol{Q}^{(\alpha,\beta)} = (Q_0^{(\alpha,\beta)}, Q_1^{(\alpha,\beta)}, \ldots)^T$, and the matrix $C^{(k)}$, which has bandwidth $1+2k$, is easily computed from well-known, explicit Jacobi recursion operators.

The Petrov-Galerkin method leads naturally to coefficient matrices that consist of banded stencil matrices and index shifted versions of the matrices obtained by the integration preconditioner (IP) method. However, there is no explicit need for preconditioners (or "quasi-inverse" matrices) in the description of the method, only test and trial functions and naturally assembled coefficient matrices. We have described the generic coefficient matrix for an equation with polynomial coefficients, i.e., $(\partial^{k-l} Q_n, x^q \phi_m^{(k)})_\omega$ for integer k, l and q and indices m and n, through an explicit expression, which is easily computed simply from already well-known recursion matrix operators for Jacobi polynomials.

Since all coefficient matrices are strictly banded, the extension of the method to multiple dimensions is trivial through the use of Kronecker product methods. This is an advantage over tau-based methods, where the coefficient matrices are almost-banded, using full rows to implement boundary conditions. For the current method homogeneous boundary conditions are built into the trial functions, and inhomogeneous conditions can be added through lifting functions that do not interfere at all with the derived, strictly banded, coefficient matrices.

The Petrov-Galerkin method described in this paper has already been implemented in the open source global spectral Galerkin software framework Shenfun [26] for any ultraspherical or Jacobi basis. It can be used for any linear differential equations with polynomial coefficients, with any composition of Dirichlet and Neumann boundary conditions. For higher dimensions it is using tensor product methods.

Code availability. For reproducibility, the examples in this paper have all been computed with Shenfun (https://github.com/spectralDNS/shenfun), version 4.0.1. The public repository https://github.com/spectralDNS/PG-paper-2022 contains code used to create all figures in the paper.

Acknowledgements. I acknowledge support from the 4DSpace Strategic Research Initiative at the University of Oslo.

REFERENCES

[1] *List of finite element software packages*, https://en.wikipedia.org/wiki/List_of_finite_element_software_packages.

693 [2] F. Auteri and L. Quartapelle, *Galerkin–Legendre Spectral Method for the 3D Helmholtz Equation*,
694 Journal of Computational Physics, 161 (2000), pp. 454–483, https://doi.org/10.1006/jcph.2000.
695 6504.
696 [3] J. P. Boyd, *Chebyshev and Fourier Spectral Methods*, Dover publications, second ed., 2001.
697 [4] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, *Dedalus: A flexible frame-*
698 *work for numerical simulations with spectral methods*, Phys. Rev. Research, 2 (2020), p. 023068,
699 https://doi.org/10.1103/PhysRevResearch.2.023068.
700 [5] C. Canuto, M. Hussaini, A. Quarteroni, and J. Thomas A., *Spectral Methods in Fluid Dynamics*,
701 Scientific Computation, Springer Berlin Heidelberg, 2012.
702 [6] C. W. Clenshaw, *The numerical solution of linear differential equations in Chebyshev series*, Math-
703 ematical Proceedings of the Cambridge Philosophical Society, 53 (1957), p. 134–149, https:
704 //doi.org/10.1017/S0305004100032072.
705 [7] E. A. Coutsias, T. Hagstrom, J. Hesthaven, and D. Torres, *Integration preconditioners for differ-*
706 *ential operators in spectral tau-methods*, in ICOSAHOM-95. Proceedings, A. Ilin and L. Ridgway
707 Scott, eds., Houston Journal of Mathematics, Houston Journal of Mathematics, 1996, pp. 21–38.
708 3rd International Conference on Spectral and High Order Methods, ICOSAHOM '95.
709 [8] E. A. Coutsias, T. Hagstrom, and D. Torres, *An Efficient Spectral Method for Ordinary Differ-*
710 *ential Equations with Rational Function Coefficients*, Mathematics of Computation, 65 (1996),
711 pp. 611–635, www.jstor.org/stable/2153604.
712 [9] E. H. Doha and W. M. Abd-Elhameed, *Efficient Spectral-Galerkin Algorithms for Direct Solution*
713 *of Second-Order Equations Using Ultraspherical Polynomials*, SIAM Journal on Scientific Com-
714 puting, 24 (2002), pp. 548–571, https://doi.org/10.1137/S1064827500378933.
715 [10] T. A. Driscoll, N. Hale, and L. N. Trefethen, *Chebfun guide*, 2014.
716 [11] K. Du, *On Well-Conditioned Spectral Collocation and Spectral Methods by the Integral Reformulation*,
717 SIAM Journal on Scientific Computing, 38 (2016), pp. A3247–A3263, https://doi.org/10.1137/
718 15M1046629.
719 [12] E. M. E. Elbarbary, *Efficient Chebyshev–Petrov–Galerkin Method for Solving Second-Order*
720 *Equations*, Journal of Scientific Computing, 34 (2008), pp. 113–126, https://doi.org/10.1007/
721 s10915-007-9161-9.
722 [13] B. Fornberg and D. M. Sloan, *A review of pseudospectral methods for solving partial differential*
723 *equations*, Acta Numerica, 3 (1994), p. 203–267, https://doi.org/10.1017/S0962492900002440.
724 [14] F. Ghoreishi and S. Hosseini, *The Tau method and a new preconditioner*, Journal of Computational
725 and Applied Mathematics, 163 (2004), pp. 351–379, https://doi.org/10.1016/j.cam.2003.04.001.
726 [15] W. J. Gordon and L. C. Thiel, *Transfinite mappings and their application to grid generation*,
727 Applied Mathematics and Computation, 10-11 (1982), pp. 171 – 233, https://doi.org/10.1016/
728 0096-3003(82)90191-6.
729 [16] D. Gottlieb and S. A. Orszag, *Numerical analysis of spectral methods: theory and applications*,
730 SIAM, 1977.
731 [17] L. Greengard, *Spectral Integration and Two-Point Boundary Value Problems*, SIAM Journal on
732 Numerical Analysis, 28 (1991), pp. 1071–1080, https://doi.org/10.1137/0728057.
733 [18] Y. Gu and J. Shen, *An Efficient Spectral Method for Elliptic PDEs in Complex Domains with Circular*
734 *Embedding*, SIAM Journal on Scientific Computing, 43 (2021), pp. A309–A329, https://doi.org/
735 10.1137/20M1345153.
736 [19] B. Guo, *Spectral methods and their applications*, World Scientific, 1998.
737 [20] B.-Y. Guo, J. Shen, and L.-L. Wang, *Generalized jacobi polynomials/functions and their ap-*
738 *plications*, Applied Numerical Mathematics, 59 (2009), pp. 1011–1028, https://doi.org/https:
739 //doi.org/10.1016/j.apnum.2008.04.003.
740 [21] J. S. Hesthaven, *Integration Preconditioning of Pseudospectral Operators. I. Basic Linear Opera-*
741 *tors*, SIAM Journal on Numerical Analysis, 35 (1998), pp. 1571–1593, https://doi.org/10.1137/
742 S0036142997319182.
743 [22] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb, *Spectral Methods for Time-Dependent Prob-*
744 *lems*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University
745 Press, 2007, https://doi.org/10.1017/CBO9780511618352.
746 [23] K. Julien and M. Watson, *Efficient multi-dimensional solution of PDEs using Chebyshev spectral*
747 *methods*, Journal of Computational Physics, 228 (2009), pp. 1480–1503, https://doi.org/10.1016/
748 j.jcp.2008.10.043.
749 [24] C. Lanczos, *Trigonometric interpolation of empirical and analytical functions*, Journal of Mathemat-
750 ics and Physics, 17 (1938), p. 123–199, https://doi.org/10.1002/sapm1938171123.
751 [25] B. Miquel, *Coral: a parallel spectral solver for fluid dynamics and partial differential equations*,
752 Journal of Open Source Software, 6 (2021), p. 2978, https://doi.org/10.21105/joss.02978, https:

//doi.org/10.21105/joss.02978.
[26] M. Mortensen, *Shenfun: High performance spectral Galerkin computing platform*, Journal of Open Source Software, 3 (2018), p. 1071, https://doi.org/10.21105/joss.01071.
[27] S. Olver and A. Townsend, *A Fast and Well-Conditioned Spectral Method*, SIAM Review, 55 (2013), pp. 462–489, https://doi.org/10.1137/120865458.
[28] S. Olver and A. Townsend, *A practical framework for infinite-dimensional linear algebra*, in Proceedings of the 1st Workshop for High Performance Technical Computing in Dynamic Languages – HPTCDL '14, IEEE, 2014.
[29] S. A. Orszag, *Accurate solution of the Orr–Sommerfeld stability equation*, Journal of Fluid Mechanics, 50 (1971), p. 689–703, https://doi.org/10.1017/S0022112071002842.
[30] S. A. Orszag, *Spectral methods for problems in complex geometries*, Journal of Computational Physics, 37 (1980), pp. 70–92, https://doi.org/10.1016/0021-9991(80)90005-4.
[31] J. Shen, *Efficient Spectral-Galerkin Method I. Direct Solvers of Second- and Fourth-Order Equations Using Legendre Polynomials*, SIAM Journal on Scientific Computing, 15 (1994), pp. 1489–1505, https://doi.org/10.1137/0915089.
[32] J. Shen, *Efficient Spectral-Galerkin Method II. Direct Solvers of Second- and Fourth-Order Equations Using Chebyshev Polynomials*, SIAM Journal on Scientific Computing, 16 (1995), pp. 74–87, https://doi.org/10.1137/0916006.
[33] J. Shen, *Efficient Spectral-Galerkin Methods III: Polar and Cylindrical Geometries*, SIAM Journal on Scientific Computing, 18 (1997), pp. 1583–1604, https://doi.org/10.1137/S1064827595295301.
[34] J. Shen, *Efficient Spectral-Galerkin Methods IV. Spherical Geometries*, SIAM Journal on Scientific Computing, 20 (1999), pp. 1438–1455, https://doi.org/10.1137/S1064827597317028.
[35] J. Shen, T. Tang, and L.-L. Wang, *Spectral Methods - Algorithms, Analysis and Applications*, Springer-Verlag Berlin Heidelberg, 2011.
[36] D. Viswanath, *Spectral integration of linear boundary value problems*, Journal of Computational and Applied Mathematics, 290 (2015), pp. 159–173, https://doi.org/10.1016/j.cam.2015.04.043.