# Well-conditioned collocation schemes and new triangular spectral-element methods

Villena Samson, Michael Daniel

2014

Villena Samson, M. D. (2014). Well-conditioned collocation schemes and new triangular spectral-element methods. Doctoral thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/60760

https://doi.org/10.32657/10356/60760

# Well-Conditioned Collocation Schemes and New Triangular Spectral-Element Methods

**Michael Daniel Villena Samson**

Division of Mathematical Sciences

School of Physical and Mathematical Sciences

Nanyang Technological University

Singapore

A thesis submitted for the degree of

*Doctor of Philosophy*

May 2014

# Acknowledgements

Primarily, I would like to share my gratitude to my supervisor Prof. Li-Lian Wang, for his guidance through these years of study, and for providing me the privilege of working in these interesting research areas. Without his steady supervision, unfailing encouragement and invaluable wisdom during the course of this candidature, this thesis could not have reached its present form.

I am grateful to Nanyang Technological University for awarding me the Research Scholarship. I have benefited much from the robust, productive and multifaceted environment of the School of Physical and Mathematical Sciences, where I could develop further as a researcher. In particular, I appreciate the management and leadership of Prof. San Ling, Prof. Yeow Meng Chee, Prof. Bernhard Schmidt and Prof. Chaoping Xing, and the help from its IT and Admin staff.

I would also like to thank Prof. Alfred Bruckstein, Prof. Frederique Oggier, Prof. Sinai Robins, Prof. Liangyi Zhao, Prof. Dmitri Pasechnik, Dr. M. Frederic Ezerman and Dr. Bo Wang for their thoughtful discussions and collaborations.

My gratitude extends to other members in our research group, including Jing Zhang, Ying Gu, Jie Chen, Yuping Duan, Juan Shi, Min Wan, Yu Wang and Xiaodan Zhao, for their friendship and help.

Last, but not least, I thank my family for their love and support.

# Contents

# List of Tables

# List of Figures

# Abstract

In the first portion of this thesis, a new well-conditioned collocation method for solving differential equations based on Birkhoff interpolation is presented. The collocation schemes on interior points using the interpolation basis functions produce linear systems that do not use differentiation matrices and have coefficient matrices with condition numbers independent of the number of points. The method is extended to different differentiation orders, computational domains and dimensionalities, noting corresponding implementation issues.

In the latter portion of this thesis, a new triangular spectral-element method using a recently introduced rectangle-triangle map is presented. This map induces a logarithmic singularity, removed by a fast, stable and accurate numerical algorithm; thus, triangular elements are as efficiently handled as quadrilateral elements. Optimal estimates of approximation by the new modal and nodal bases on a triangle are obtained. Efficient and accurate implementations on one triangle and on an unstructured triangulation of a polygon are demonstrated.

# Notation

## Common notation

| | |
|---|---|
| $\delta_{ij}$ | Kronecker delta |
| $O(f(x))$ | upper-bound O-notation, i.e. $g(x) \in O(f(x))$ means $g(x) \leq cf(x)$ for some positive constant $c$ |
| $\vec{a}$ | vector |
| $(a_1, \ldots, a_N)$ | row vector |
| $(a_1, \ldots, a_N)^t$ | column vector |
| $\boldsymbol{A} = [a_{ij}]_{1 \leq j \leq N}^{1 \leq i \leq M}$ | $M \times N$ matrix |
| $\mathrm{diag}(d_1, \ldots, d_N)$ | $N \times N$ diagonal matrix with diagonal $d_1, \ldots, d_N$ |
| $\boldsymbol{I}_M$ | $M \times M$ identity matrix, $\boldsymbol{I}_M = [\delta_{ij}]$ |

## Computational domains

| | |
|---|---|
| $I$ | standard interval $(-1, 1)$ |
| $\Omega$ | polygonal domain |

## Basis polynomials

| | |
|---|---|
| $L_k(x), \ell_k(x)$ | Lagrange interpolation basis polynomials |
| $\widehat{L}_k(x), \widehat{\ell}_k(x)$ | Lagrange interpolation basis functions |
| $B_k(x)$ | Birkhoff (-type) interpolation basis polynomials |
| $\widehat{B}_k(x)$ | Birkhoff (-type) interpolation basis functions |

## Orthogonal polynomial families

| | |
|---|---|
| $P_k^{\alpha,\beta}(x)$ | Jacobi polynomials of degree $k$ (associated weight: $\omega^{\alpha,\beta} = (1-x)^\alpha(1+x)^\beta$) |
| $G_k^\alpha(x)$ | normalized Gegenbauer polynomials of degree $k$, $G_k^\alpha(x) = P_k^{\alpha,\alpha}(x)/P_k^{\alpha,\alpha}(1)$ |
| $P_k(x) = G_k^0(x)$ | Legendre polynomials of the first kind of degree $k$ |
| $\widetilde{P}_k(x)$ | Legendre polynomials of the second kind of degree $k$ |
| $Q_k(x)$ | Legendre functions of the second kind of degree $k$ |
| $T_k(x) = G_k^{1/2}(x)$ | Chebyshev polynomials of degree $k$ |
| $\mathcal{L}_k(x)$ | Laguerre polynomials of degree $k$ |
| $\widehat{\mathcal{L}}_k(x)$ | Laguerre functions of degree $k$ |

## Function spaces

| | |
|---|---|
| $\mathbb{P}_N$ | function space of all polynomials of degree $N$ or less in one variable |
| $\mathbb{P}_N(\Omega)$ | function space of all polynomials of total degree $N$ or less on the domain |
| $\mathbb{Q}_N(\Omega)$ | function space of all polynomials of degree $N$ or less in each variable on the domain |
| $L^2_\omega(\Omega)$ | weighted function space of square-integrable functions on the domain |
| $H^1(\Omega)$ | function space of square-integrable functions with square-integrable partial derivatives on the domain |
| $H(\mathrm{div};\Omega)$ | function space of square-integrable functions with square-integrable div on the domain |

Chapter 1

# Introduction

This chapter elaborates on the background and the motivation of the topics to be studied in this thesis, and highlights the main contributions. For clarity, we address the issues separately for the two parts of the thesis.

## 1.1 Collocation schemes

The spectral collocation method, also called the *pseudospectral method* sometimes, is implemented in physical space and approximates derivative values by direct differentiation of the Lagrange interpolating polynomial at a set of Gauss-type points, generating *pseudospectral differentiation matrices* (PSDMs). Its fairly straightforward realization is akin to the high-order finite difference method (cf. [52, 109]). This marks its advantages over the spectral method using modal basis functions in dealing with variable coefficient and/or nonlinear problems (see various monographs on spectral methods [61, 63, 14, 19, 71, 101]). The ubiquity of the fast Fourier transform and the ease of determining Chebyshev-Gauss-type points and quadrature weights accounts for the popularity of pseudospectral methods amongst engineers. However, practitioners are plagued with the involved ill-conditioned linear systems, e.g., the condition number of the $p$th-order PSDM grows like $N^{2p}$. This longstanding drawback causes severe degradation

of expected spectral accuracy [110], while the accuracy of machine zero can be well observed from the well-conditioned spectral-Gakerkin method (see e.g., [99]). In practice, it becomes rather prohibitive to solve the linear system by a direct solver, or even by an iterative method, when the number of collocation points is large.

One significant attempt to circumvent this barrier is the use of suitable *preconditioners*. Preconditioners built on low-order finite-difference or finite-element approximations can be found in, e.g., [38, 39, 21, 75, 76, 18]. *Integration preconditioning* (IP) proposed by Coutsias, Hagstrom, Hesthaven, et al, [36, 35, 70] (with ideas from Clenshaw [29]) has proven to be efficient in alleviating the ill-conditioning. The key to its success resides in realizing that any orthogonal polynomial $p_n(x)$ can be sparsely represented by $\{p'_{n\pm k}(x) : k = 0, 1\}$, where the process is much more stable due to coefficients that decay as in $1/n$, opposite to the process from $p'_n(x)$ to $\{p_k(x) : 0 \leq k \leq n-1\}$, which has coefficients that grow as in $n$. We highlight that the IP in Hesthaven [70] led to a significant reduction of the condition number from $O(N^4)$ to $O(\sqrt{N})$ for second-order differential linear operators with Dirichlet boundary conditions (which were imposed by the penalty method [57]). Elbarbary [47] improved the IP in [70] through carefully manipulating the involved singular matrices and imposing the boundary conditions by some auxiliary equations. Similar ideas of inverting the PSDM for $\mathrm{d}/\mathrm{d}x - \mu$ were attempted by Loncaric [84]. We also remark that Funaro [56, 50, 51] proposed a superconsistent Chebyshev collocation method, where the solution was computed at usual Chebyshev points while the equation was collocated at new super-consistent points. This interesting approach led to better convergence.

Another remarkable approach is the *spectral integration method* proposed by Greengard [62] (also see [121]), which recasts the differential form into integral form, and then approximates the solution by orthogonal polynomials. This method was incorporated into the `chebop` system [43, 42]. A relevant approach by El-Gendi [46] is not based on reformulating the differential equations, but

uses the integrated Chebyshev polynomials as basis functions. Then the spectral integration matrix (SIM) is employed in place of PSDM to obtain much better-conditioned linear systems (see e.g., [87, 59, 88, 48] and the references therein).

In the first part of this thesis, we take a very different route to construct well-conditioned collocation methods. The essential idea is to associate the highest differential operator and underlying boundary conditions with a suitable *Birkhoff interpolation* (cf. [85, 104]) of the derivative values at interior collocation points, and the boundary data at endpoints.[1] This leads to the so-called *Birkhoff interpolation basis polynomials* with the following distinctive features:

- Under the new modal-type basis, the linear system of a usual collocation scheme on the interior collocation points is well-conditioned, where the matrix of the highest derivative is diagonal or identity and no differentiation matrices are involved. Moreover, the underlying boundary conditions are imposed exactly. This technique can be viewed as the collocation analogue of the well-conditioned spectral-Galerkin method (cf. [99, 100, 64]), where the matrix of the highest derivative in the Galerkin system is diagonal under certain modal basis functions.

- The new basis produces the *exact inverse* of PSDM of the highest derivative, with boundary conditions (also when only on interior collocation points, in the case of Dirichlet boundary conditions). This inspires us to introduce the concept of pseudospectral integration matrix (PSIM). The integral expression of the new basis offers a stable way to compute PSIM and the inverse of PSDM even for thousands of collocation points.

- PSIMs lead to optimal integration preconditioners for the usual collocation methods, and enables us to have insights into the IP in [70, 47]. Indeed, the preconditioning from Birkhoff interpolation is natural and optimal.

[1] It is noteworthy that our idea is also inspired by the special Birkhoff interpolation considered by Zhang [122] in the context of superconvergence of polynomial interpolation.

In Ch. 2, we describe the detailed construction of the new collocation method for one-dimensional second-order boundary-value problems (BVPs) with general boundary conditions. We also provide ample numerical comparisons and evidence to show the well-conditioning of this new approach, together with a proof for the simple case involving the Helmholtz operator $\mathrm{d}^2/\,\mathrm{d}x^2 - \mu$.

In Ch. 3, we consider various extensions of the methodology proposed in Ch. 2. Basically, we extend the new well-conditioned collocation methods for solving first-order initial value problems (IVPs, using the Birkhoff interpolation on Gauss-Radau points) and higher-order BVPs, as well as the time-dependent third-order and fifth-order Korteweg-de Vries equations. We also consider multi-dimensional cases by integrating it with matrix decomposition techniques. More importantly, the well-conditioned method is nontrivially extended to the Laguerre functions on the half-line. In fact, the eigenvalues of the second-order differentiation matrix on Laguerre-Gauss-Radau points have quite different behaviour from those of the Legendre/Chebyshev polynomials. This requires the introduction of a new differential operator to regroup some derivative terms.

## 1.2 DG-based triangular spectral-element methods

The spectral element method (SEM), originated from Patera [93], integrates the unparalleled accuracy of spectral methods with the geometric flexibility of finite elements, and also enjoys a high-level parallel computer architecture. Nowadays, it has become a pervasive numerical technique for simulating challenging problems in complex geometries [37, 20]. While the classical SEM on quadrilateral/hexahedral elements (QSEM) exhibits the advantages of using tensorial basis functions and naturally diagonal mass matrices, the need for high-order methods on unstructured meshes with robust adaptivity spawns the development of triangular/tetrahedral spectral elements (TSEM). In general, research efforts

along this line fall into three trends:

(i) nodal TSEM based on high-order polynomial interpolation on special interpolation points [26, 69, 108, 107];

(ii) modal TSEM based on the Koornwinder-Dubiner polynomials [78, 44, 74];

(iii) approximation by non-polynomial functions [103, 81, 24].

For the developments along (i), the question of how to construct "good" interpolation points for stable high-order polynomial interpolation on the triangle is still quite subtle and somehow open. The strict analogy of the Gauss-Lobatto integration rule on quadrilaterals/hexahedra does not exist on triangles [68], though a "relaxed" rule can be constructed in the sense of [120]. We refer to [92] for an up-to-date review and a very dedicated comparative study of various criteria for constructing workable interpolation points on the triangle. In general, such points have low degree of precision (i.e., exactness for integration of polynomials), and this motivates the use of a different set of points for integration (see [91]). However, this needs extra effort in interpolating the solution and extra input between two sets of points.

The developments along (ii) can be best attested to by the monograph [74] and the spectral-element package `NekTar` (http://www.nektar.info/). The analysis of this approach can be found in, e.g., [65, 98, 80, 27]. However, the main drawbacks of this approach lie in that the interpolation points are unfavorably clustered near one vertex of the triangle, and in the absence of a corresponding nodal basis, making it complicated to implement. To overcome the second difficulty, developments along (iii) have arisen: a fully-tensorial rational approximation on triangles was proposed in [103] for elliptic problems, and extended to the Navier-Stokes problem in [24]. However, this approach still builds on the collapsed Duffy's transform with clustered grids.

It is important to point out that Duffy's transform not only leads to undesirable distributions of interpolation/quadrature points, but also requires modifying the tensorial polynomial basis to meet the underlying consistency conditions (analogous to "pole conditions" in polar/spherical coordinates) induced by the singularity of the transform.

In the second part of this thesis, we aim to develop a new triangular spectral-element method on unstructured meshes and overcome some numerical difficulties of aforementioned TSEM. We base the new TSEM on the rectangle-triangle mapping introduced in [82]. This mapping pulls one side (at the midpoint) of the triangle to two sides of the rectangle (see Fig. 4.1 (a)), and results in relatively desirable distributions of the mapped LGL points (cf. Fig. 4.1 (c) vs. (d)) as interpolation/quadrature points on the triangle. Moreover, this mapping is one-to-one. Note that the TSEM in [82] also requires modifying the tensorial polynomial basis to meet the underlying consistency conditions induced by the singularity of the new transform. In the new approach, we find a way to remove the singularity and significantly improve the efficiency of implementation.

In Ch. 4, we describe the detailed implementation and error analysis of this new method on a triangle, which paves the way for the new TSEM on unstructured meshes to be discussed in Ch. 5. We address in Ch. 4 the following issues.

- We have some new insights into the originality and distinctive features of the new mapping.
- We demonstrate that the singularity of the mapping is of logarithmic type, which can be fully removed.
- We provide optimal error estimates for approximation by the associated basis functions.

Our approach brings about an important viewpoint that any triangular element can be mapped to the reference square via a composite of the rectangle-triangle mapping and an affine mapping and, with the successful removal of the singularity,

the triangular element can be treated as efficiently as a quadrilateral element, so one can handle more complex domains with more regular computational meshes.

One immediate benefit is that one can tile the triangular elements along the boundary of the complex domain while using quadrilaterals in the interior. Such a combination leads to much regular mesh which is important for high-order methods. However, the main difficulty in implementing element methods on an unstructured mesh arises from handling the coupling, or interdependence, of solutions among the mesh elements, in particular through the internal edges imposed by the mesh. The new TSEM in particular adds the complication of an induced singularity on the midpoint of one edge of each element, each of which can be treated as a hanging node of the mesh. This inspires us to build the TSEM upon the discontinuous Galerkin (DG) formulation.

DG methods, first developed in 1973 by Reed and Hill [95] for hyperbolic equations, and since applied to elliptic problems (see [5, 41, 6, 117, 2, 7, 97, 11, 10] for interior penalty methods, [8, 33, 9] for approaches based on convection-dominated problems, [16] for a unified approach and [4] for a review within a unified framework), enjoy a large degree of flexibility, non-conformity and locality. In particular, DG methods can handle hanging nodes in meshes, unlike other methods (e.g., continuous Galerkin methods, mixed methods), while providing a scheme to handle the coupling on the mesh.

In Ch. 5, we use the hybridized LDG method (LDG-H, cf. [77, 30, 31]) with the new TSEM to develop an unstructured TSEM, which we use to solve a model elliptic PDE. LDG-H has the following properties, which are desirable for use with the new TSEM:

- LDG-H makes use of auxillary functions, which renders the elliptic problem into a system of first-order differential equations. For those equations, the rectangle-triangle map (4.1)–(4.2) does not induce a singularity.

- LDG-H generates a global system whose degrees of freedom are only those

on the interior edges. Here, hanging nodes are handled by the computational mechanism used to generate the global system, and elements can be solved independently from the relevant portion of the global solution.

The resulting unstructured TSEM serves as a preliminary extension of the new TSEM to complex domains.

## 1.3 Main contributions and outline

The main content and contributions of this thesis are highlighted as follows.

- In Ch. 2, we present the new method for generating well-conditioned collocation schemes, for solving general-boundary second-order BVPs:

  (i) We find a natural way to construct well-conditioned collocation methods from the perspective of Birkhoff interpolation.

  (ii) The Birkhoff interpolation basis leads to the exact inverse of the differentiation matrix on interior collocation points, so the resulting preconditioner is optimal.

  (iii) The new basis can also be used a modal-type basis, under which the collocation system does not involve the differentiation matrix, leading to more stable collocation schemes for very large number of points.

  Using Legendre- and Chebyshev-Gauss-Lobatto spectral collocation points, we demonstrate that the linear systems of the new collocation schemes for the BVPs have coefficient matrices whose condition numbers are independent of the number of collocation points, and that the PSIM are constructed efficiently and in a stable manner.

- In Ch. 3, we construct, by a new differentiation operator, a novel Birkhoff-type interpolation—using the behavior of the eigenvalues of the second-order differentiation matrix on Laguerre-Gauss-Radau points—that is more

suitable, with the method in Ch. 2, for generating a well-conditioned collocation scheme to solve second-order IVPs on the half-line. We extend the method in Ch. 2 to generate well-conditioned collocation schemes to solve IVPs on Legendre- and Chebyshev-Gauss-Radau points, higher-order BVPs for stable simulations of time-dependent KdV equations, and second-order BVPs on Gegenbauer-Gauss-Lobatto points. Using the inverse matrix from Ch. 2 and matrix decomposition methods, we generate a collocation scheme to solve a model two-dimensional elliptic PDE.

- In Ch. 4, we present a new TSEM, using the rectangle-triangle mapping in [82], and new modal and nodal basis functions on the triangle with optimal $L^2$- and $H^1$-estimates for projection and interpolation errors. For the first time, no modifications are made on the modal and nodal basis functions on the reference square to generate the corresponding basis functions on the triangle under the map. We determine that the singularity arising from computations in the stiffness matrix is removable, and produce an algorithm that efficiently and accurately performs these computations in a stable manner. We show that the TSEM can be applied to any triangle, with analogous matrix computations that can be performed with the same efficiency.

- In Ch. 5, we seamlessly integrate the TSEM in Ch. 4 with the LDG-H formulation to generate a new TSEM for unstructured meshes. We demonstrate the efficiency of the construction of the global solver, incorporating the hanging nodes induced by the rectangle-triangle map, and the parallelizable construction of the local solvers. Through numerical results on a model elliptic PDE, we show our results improve similar implementations over TSEM using Duffy's transform.

Chapter 2

# Well-Conditioned Collocation Methods for Second-Order BVPs

As already mentioned in Ch. 1, the collocation method using PSDM suffers from severe ill-conditioning. Although some preconditioners have been proposed in [36, 35, 70, 47], they appear non-optimal; the purpose of this chapter is to introduce well-conditioned collocation methods. This chapter focuses on developing these well-conditioned collocation methods for second-order boundary value problems (BVPs), as the preconditioner texts above have been applied to such.

The essential idea is to develop pseudospectral integration matrices (PSIMs). PSIMs are well-equipped to be the core of developing efficient and stable well-conditioned collocation schemes: as in [34], evaluating Birkhoff interpolation basis polynomials $\{B_k\}$, which result from particular Birkhoff interpolation problems that incorporate boundary data from the differential equation, at the collocation points gives PSIMs; when using spectral collocation points, these values can be derived in an efficient and stable manner. PSIMs are then used in the collocation scheme corresponding to this basis (herein referred to as BCOL) for these differential equations.

We point out that Castabile and Longo [34] touched on the application of Birkhoff interpolation (see (2.26)) to second-order boundary value problems (BVPs),

but the focus of their work was largely on the analysis of interpolation and quadrature errors. Zhang [122] considered the Birkhoff interpolation (see (3.2)) in a very different context of superconvergence of polynomial interpolation. Collocation methods based on a special Birkhoff quadrature rule for Neumann problems were discussed in [49, 111] and for mixed-boundary problems in [112]. It is also noteworthy to point out recent interest in developing spectral solvers using modal basis functions (see e.g., [83, 23, 89, 66]). Lastly, the works [53, 105] discussed the inverse of the PSDM, but not from the same perspective as ours.

The rest of the chapter is outlined as follows: Preliminaries on pseudospectral method and orthogonal polynomials are given in Sec. 2.1. PSIM for second-order boundary value problems (BVPs) are demonstrated in Sec. 2.2, proving their properties and providing numerical results.

## 2.1 Preliminaries

We begin with some preliminaries for the subsequential algorithm and analysis.

### 2.1.1 Birkhoff interpolation

Let $\{x_j\}_{j=0}^N \subseteq [-1,1]$ be a set of distinct interpolation points, which are arranged in ascending order:

$$-1 \le x_0 < x_1 < \cdots < x_{N-1} < x_N \le 1. \tag{2.1}$$

Given $K+1$ data $\{y_j^m\}$ (with $K \ge N$), we consider the interpolation problem (cf. [85, 104]):

$$\begin{cases} \text{Find a polynomial } p_K \in \mathbb{P}_K \text{ such that} \\ p_K^{(m)}(x_j) = y_j^m \quad (K+1 \text{ equations}), \end{cases} \tag{2.2}$$

where $\mathbb{P}_K$ is the set of all algebraic polynomials of degree at most $K$, and the superscript $m$ indicates the order of specified derivative values.

We have Hermite interpolation if, for each $j$, the orders of derivatives in (2.2) form an unbroken sequence, $m = 0, 1, \ldots, m_j$. In this case, the interpolation polynomial $p_K$ uniquely exists and can be given by an explicit formula. In particular, Hermite interpolation with $m_j = 0$ for each $j$ is called *Lagrange interpolation*, and $K = N$, i.e. $p_N \in \mathbb{P}_N$. On the other hand, if some of the sequences are broken, we have *Birkhoff interpolation*.

The existence and uniqueness of the Birkhoff interpolation polynomial are not guaranteed. For example, for (2.2) with $K = N = 2$, and the given data $\{y_0^0, y_1^1, y_2^0\}$, the quadratic polynomial $p_2(x)$ does not exist when $x_1 = (x_0 + x_2)/2$. This happens to Legendre/Chebyshev-Gauss-Lobatto points, where $x_0 = -1, x_1 = 0$ and $x_2 = 1$. We refer to the monographs [85, 104] for comprehensive discussions of Birkhoff interpolation.

In this text, we will consider special Birkhoff interpolation problems at Gauss-type points, and some variants we will call *Birkhoff-type interpolation* that incorporate with mixed boundary data, for instance, $ap_K'(-1) + bp_K(-1) = y_0$ for constants $a, b$.

**Remark 2.1.** *Even when uniqueness and existence of the Birkhoff interpolation polynomial are assured, care must be taken when selecting the points $x_j$ when considering interpolation accuracy—Gauss-type points provide sufficient assurance of accuracy. An example follows.*

*Let the interpolation points be $x_0 < x_1 < x_2$, and $k_-, k_+ \geq 0$. Consider the interpolation problem:*

$$\begin{cases} \text{Find a polynomial } p \in \mathbb{P}_{k_-+k_+} \text{ such that, for } u \in C^{k_-+k_+}(x_0, x_2), \\ p^{(k)}(x_0) = u^{(k)}(x_0),\ 0 \leq k < k_-, \quad p^{(k)}(x_2) = u^{(k)}(x_2),\ 0 \leq k < k_+, \\ p^{(k_-+k_+)}(x_1) = u^{(k_-+k_+)}(x_1). \end{cases}$$

*This interpolation scheme is order regular [85], and its existence and uniqueness are assured for whatever choice of interpolation points.*

*However, for $x_0 = -1$, $x_2 = 1$, $k_- = k_+ = 1$, the quadratic interpolant is*

$$p(x) = u(-1)\left(\frac{1-x}{2}\right) + u''(x_1)\left(\frac{x^2-1}{2}\right) + u(1)\left(\frac{1+x}{2}\right). \tag{2.3}$$

*The resulting interpolation matrix is nearly singular whenever $x_1 \to \pm 1$, a possible source of round-off error in computing the interpolant. For Gauss-type points, $x_1 = 0$, as noted above.*

### 2.1.2 Pseudospectral differentiation matrix

The pseudospectral differentiation matrix (PSDM) is an essential building block for collocation methods. Let $\{x_j\}_{j=0}^N$ (with $x_0 = -1$ and $x_N = 1$), and let $\{L_j\}_{j=0}^N$ be the *Lagrange interpolation basis polynomials* such that $L_j \in \mathbb{P}_N$ and $L_j(x_i) = \delta_{ij}$, for $0 \le i, j \le N$, where $\delta_{ij}$ is 1 if $i = j$ and 0 otherwise. Recall that

$$L_j(x) = \frac{q(x)}{(x - x_j)q'(x_j)}, \tag{2.4}$$

where

$$q(x) = c\prod_{j=0}^{N}(x - x_j), \quad c \neq 0.$$

We have

$$p(x) = \sum_{j=0}^{N} p(x_j)L_j(x), \quad \forall\, p \in \mathbb{P}_N. \tag{2.5}$$

Denoting $d_{ij}^{(k)} := L_j^{(k)}(x_i)$, we introduce the matrices

$$\boldsymbol{D}^{(k)} = [d_{ij}^{(k)}]_{0\le i,j\le N}, \quad \boldsymbol{D}_{\text{in}}^{(k)} = [d_{ij}^{(k)}]_{0< i,j< N}, \quad k \ge 1. \tag{2.6}$$

Note that $\boldsymbol{D}_{\text{in}}^{(k)}$ is obtained by deleting the last and first rows and columns of $\boldsymbol{D}^{(k)}$, so it is associated with interior points. In particular, we denote $\boldsymbol{D} := \boldsymbol{D}^{(1)}$, and $\boldsymbol{D}_{\text{in}} := \boldsymbol{D}_{\text{in}}^{(1)}$. The matrix $\boldsymbol{D}^{(k)}$ is usually referred to as the $k$th order PSDM. We highlight the following property (see e.g., [101, Thm. 3.10]):

$$\boldsymbol{D}^{(k)} = \overbrace{\boldsymbol{D}\boldsymbol{D}\cdots\boldsymbol{D}}^{k \text{ copies}} = \boldsymbol{D}^k, \quad k \ge 1, \tag{2.7}$$

so the higher-order PSDM is a product of the first-order PSDM.

Set

$$\vec{p}^{(k)} := (p^{(k)}(x_0), \ldots, p^{(k)}(x_N))^t, \quad \vec{p} := \vec{p}^{(0)}. \tag{2.8}$$

By (2.5) and (2.7), the pseudospectral differentiation process is performed via

$$\boldsymbol{D}^{(k)}\vec{p} = \boldsymbol{D}^k\vec{p} = \vec{p}^{(k)}, \quad k \geq 1. \tag{2.9}$$

**Remark 2.2.** *Differentiation via* (2.9) *suffers from significant round-off errors for large $N$, due to the involvement of ill-conditioned operations* (cf. [114]).

The matrix $\boldsymbol{D}^{(k)}$ is singular ($\boldsymbol{D}^{(k)}\vec{1}^t = \vec{0}^t$, where $\vec{1} = (1, 1, \ldots, 1)$, so the rows of $\boldsymbol{D}^{(k)}$ are linearly dependent), while $\boldsymbol{D}^{(k)}_{\text{in}}$ is nonsingular. In addition, the condition numbers of $\boldsymbol{D}^{(k)}_{\text{in}}$ and $\boldsymbol{D}^{(k)} - \boldsymbol{I}_{N+1}$, where $\boldsymbol{I}_m$ is the $m \times m$ identity matrix, behave like $O(N^{2k})$.

### 2.1.3 Legendre and Chebyshev polynomials

We collect below some properties of Legendre and Chebyshev polynomials (see e.g., [106, 55, 101]) to be used throughout this text.

Let $P_k(x)$ be the Legendre polynomial of degree $k$. Legendre polynomials are mutually orthogonal:

$$\int_{-1}^{1} P_k(x)P_j(x)\,\mathrm{d}x = \gamma_k\delta_{kj} \tag{2.10}$$

with $\gamma_k = 1/(2k+1)$. There hold

$$P_k(x) = \frac{1}{2k+1}(P'_{k+1}(x) - P'_{k-1}(x)), \quad k \geq 1, \tag{2.11}$$

and

$$P_k(\pm 1) = (\pm 1)^k, \quad P'_k(\pm 1) = \frac{1}{2}(\pm 1)^{k-1}k(k+1). \tag{2.12}$$

The Legendre-Gauss-Lobatto (LGL) points are zeros of $(1-x^2)P'_N(x)$, and the corresponding quadrature weights are

$$\omega_j = \frac{2}{N(N+1)}\frac{1}{P_N^2(x_j)}, \quad 0 \leq j \leq N. \tag{2.13}$$

Then the LGL quadrature has the exactness

$$\int_{-1}^{1} \phi(x)\,\mathrm{d}x = \sum_{j=0}^{N} \phi(x_j)\omega_j, \quad \forall \phi \in \mathbb{P}_{2N-1}; \quad \sum_{j=0}^{N} [P_N(x_j)]^2 \omega_j = \frac{2}{N}. \tag{2.14}$$

The Chebyshev polynomials $T_k(x) = \cos(k\arccos(x))$ are mutually weighted-orthogonal

$$\int_{-1}^{1} \frac{T_k(x)T_j(x)}{\sqrt{1-x^2}}\,\mathrm{d}x = \gamma_k \delta_{kj} \tag{2.15}$$

with $\gamma_k = c_k\pi/2$, where $c_0 = 2$ and $c_k = 1$ for $k \geq 1$. We have

$$T_k(x) = \frac{1}{2(k+1)} T'_{k+1}(x) - \frac{1}{2(k-1)} T'_{k-1}(x), \quad k \geq 2, \tag{2.16}$$

and

$$T_k(\pm 1) = (\pm 1)^k, \quad T'_k(\pm 1) = (\pm 1)^{k-1} k^2. \tag{2.17}$$

The Chebyshev-Gauss-Lobatto (CGL) points and quadrature weights are

$$\begin{aligned} &x_j = -\cos(jh), \quad 0 \leq j \leq N; \\ &\omega_0 = \omega_N = \frac{h}{2}, \quad \omega_j = h,\ 0 < j < N; \quad h = \frac{\pi}{N}. \end{aligned} \tag{2.18}$$

Then the CGL quadrature has the exactness

$$\int_{-1}^{1} \frac{\phi(x)}{\sqrt{1-x^2}}\,\mathrm{d}x = \sum_{j=0}^{N} \phi(x_j)\omega_j, \quad \forall \phi \in \mathbb{P}_{2N-1}; \quad \sum_{j=0}^{N} [T_N(x_j)]^2 \omega_j = \pi. \tag{2.19}$$

### 2.1.4 Integration preconditioning

We briefly examine the essential idea of constructing integration preconditioners in [70, 47] (inspired by [36, 35]).

We consider for example the Legendre case. By (2.10) and (2.14),

$$L_j(x) = \sum_{k=0}^{N} \frac{\omega_j}{\tilde{\gamma}_k} P_k(x_j) P_k(x), \quad 0 \leq j \leq N, \tag{2.20}$$

where $\tilde{\gamma}_k = 2/(2k+1)$ for $0 \leq k < N$, and $\tilde{\gamma}_N = 2/N$. This follows from letting

$$L_j(x) = \sum_{k=0}^{N} \alpha_{jk} P_k(x),$$

where

$$\alpha_{jk} = \frac{1}{\tilde{\gamma}_k} \int_{-1}^{1} L_j(x) P_k(x)\, \mathrm{d}x.$$

Then

$$L_j''(x) = \sum_{k=2}^{N} \frac{\omega_j}{\tilde{\gamma}_k} P_k(x_j) P_k''(x). \tag{2.21}$$

The key observation in [70, 47] is that *the pseudospectral differentiation process actually involves the ill-conditioned transform:*

$$\mathrm{span}\{P_k'' : 2 \le k \le N\} := Q_2^N \longmapsto Q_0^{N-2} := \mathrm{span}\{P_k : 0 \le k \le N-2\}. \tag{2.22}$$

Indeed, we have (see [101, (3.176c)]):

$$P_k''(x) = \sum_{k+l \text{ even}}^{0 \le l \le k-2} (l + 1/2)(k(k+1) - l(l+1)) P_l(x), \tag{2.23}$$

so the transform matrix is dense and the coefficients grow like $k^2$.

However, *the inverse transform* $Q_0^{N-2} \mapsto Q_2^N$ *is sparse and stable*, thanks to the "compact" formula, derived from (2.11):

$$P_k(x) = \alpha_k P_{k-2}''(x) + \beta_k P_k''(x) + \alpha_{k+1} P_{k+2}''(x), \quad k \ge 2, \tag{2.24}$$

where the coefficients are

$$\alpha_k = \frac{1}{(2k-1)(2k+1)}, \quad \beta_k = -\frac{2}{(2k-1)(2k+3)}, \tag{2.25}$$

which decay like $k^{-2}$.

Based on (2.24), [70, 47] attempted to precondition the collocation system by the "inverse" of $\boldsymbol{D}^{(2)}$. However, since $\boldsymbol{D}^{(2)}$ is singular, there exist multiple ways to manipulate the involved singular matrices. The boundary conditions were imposed by the penalty method (cf. [57]) in [70], and by using auxiliary equations in [47]. Note that the condition number of the preconditioned system for, e.g., the operator $\mathrm{d}/\mathrm{d}x - \mu$ with Dirichlet boundary conditions, behaves like $O(\sqrt{N})$.

## 2.2 New collocation methods for second-order BVPs

Consider the Birkhoff interpolation problem on $-1 = x_0 < x_1 < \cdots < x_N = 1$ (cf. (2.2)):

$$\begin{cases} \text{Find } p \in \mathbb{P}_N \text{ such that for any } u \in C^2(I), \\ p(-1) = u(-1); \quad p''(x_j) = u''(x_j),\ 0 < j < N; \quad p(1) = u(1). \end{cases} \tag{2.26}$$

The Birkhoff interpolation polynomial $p$ of $u$ can be uniquely determined by

$$p(x) = u(-1)B_0(x) + \sum_{j=1}^{N-1} u''(x_j)B_j(x) + u(1)B_N(x), \quad x \in [-1, 1], \tag{2.27}$$

if one can find $\{B_j\}_{j=0}^N \subseteq \mathbb{P}_N$, such that

$$B_0(-1) = 1, \qquad B_0(1) = 0, \qquad B_0''(x_i) = 0, \qquad 0 < i < N; \tag{2.28}$$

$$B_j(-1) = 0, \qquad B_j(1) = 0, \qquad B_j''(x_i) = \delta_{ij}, \qquad 0 < i, j < N; \tag{2.29}$$

$$B_N(-1) = 0, \qquad B_N(1) = 1, \qquad B_N''(x_i) = 0, \qquad 0 < i < N. \tag{2.30}$$

We call $\{B_j\}_{j=0}^N$ the *Birkhoff interpolation basis polynomials* of (2.26), which are the counterpart of the Lagrange basis polynomials $\{L_j\}_{j=0}^N$ (2.4).

The basis $\{B_j\}_{j=0}^N$ can be uniquely expressed by the following formulas.

**Theorem 2.1.** *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=0}^N$ *defined in* (2.28)–(2.30) *are given by*

$$B_0(x) = \frac{1-x}{2}, \qquad B_N(x) = \frac{1+x}{2}; \tag{2.31}$$

$$\begin{aligned} B_j(x) &= \frac{1+x}{2}\int_{-1}^{1}(t-1)\ell_j(t)\,\mathrm{d}t + \int_{-1}^{x}(x-t)\ell_j(t)\,\mathrm{d}t \\ &= \frac{x-1}{2}\int_{-1}^{1}(1+t)\ell_j(t)\,\mathrm{d}t - \int_{x}^{1}(x-t)\ell_j(t)\,\mathrm{d}t, \quad 0 < j < N. \end{aligned} \tag{2.32}$$

*where* $\{\ell_j\}_{j=1}^{N-1}$ *are the Lagrange basis polynomials (of degree* $N-2$*) associated with* $N-1$ *interior points* $\{x_j\}_{j=1}^{N-1}$ (see (2.4)). *Moreover, we have*

$$\begin{aligned} B_0'(x) &= -B_N'(x) = -\frac{1}{2}; \\ B_j'(x) &= \frac{1}{2}\int_{-1}^{1}(t-1)\ell_j(t)\,\mathrm{d}t + \int_{-1}^{x}\ell_j(t)\,\mathrm{d}t \\ &= \frac{1}{2}\int_{-1}^{1}(1+t)\ell_j(t)\,\mathrm{d}t - \int_{x}^{1}\ell_j(t)\,\mathrm{d}t, \quad 0<j<N. \end{aligned} \tag{2.33}$$

**Proof**: One verifies readily from (2.28), (2.30) that $B_0$ and $B_N$ must be linear polynomials given by (2.31). Using (2.29) and the fact $B_j''(x), \ell_j(x) \in \mathbb{P}_{N-2}$, we find that $B_j''(x) = \ell_j(x)$, so solving this ordinary differential equation with boundary conditions $B_j(\pm 1) = 0$ leads to the expression in (2.32). Finally, (2.33) follows from (2.31)–(2.32). ■

**Remark 2.3.** *The new Birkhoff basis* $\{B_j\}_{j=0}^{N}$ *from* (2.31)–(2.32) *can be represented in terms of the Lagrange basis polynomials* $\{L_j\}_{j=0}^{N}$ *associated with* $\{x_j\}_{j=0}^{N}$. *We have*

$$\ell_j(x) = \frac{1-x_j^2}{1-x^2}L_j(x), \quad 0<j<N,$$

*and, by* (2.32),

$$\begin{aligned} B_j(x) &= (1-x_j^2)\left[-\frac{1+x}{2}\int_{-1}^{1}\frac{L_j(t)}{1+t}\,\mathrm{d}t + \int_{-1}^{x}\frac{x-t}{1-t^2}L_j(t)\,\mathrm{d}t\right] \\ &= (1-x_j^2)\left[\frac{x-1}{2}\int_{-1}^{1}\frac{L_j(t)}{1-t}\,\mathrm{d}t - \int_{x}^{1}\frac{x-t}{1-t^2}L_j(t)\,\mathrm{d}t\right], \end{aligned} \tag{2.34}$$

*for* $0<j<N$, *which is alternative to* (2.32).

Let $b_{ij}^{(k)} := B_j^{(k)}(x_i)$, and define the matrices

$$\boldsymbol{B}^{(k)} = [b_{ij}^{(k)}]_{0\le i,j\le N}, \quad \boldsymbol{B}_{\mathrm{in}}^{(k)} = [b_{ij}^{(k)}]_{0<i,j<N}, \ k \ge 0. \tag{2.35}$$

In particular, denote $b_{ij} := B_j(x_i)$, $\boldsymbol{B} = \boldsymbol{B}^{(0)}$ and $\boldsymbol{B}_{\mathrm{in}} = \boldsymbol{B}_{\mathrm{in}}^{(0)}$.

**Remark 2.4.** *Let* $u''(x) \equiv 1$ *in* (2.27)*: by the interpolant's uniqueness (cf.* (2.3)*)*

$$\sum_{k=1}^{N-1} B_i(x) = \frac{x^2-1}{2}. \tag{2.36}$$

We have the following analogue of (2.7), and this approach leads to the exact inverse of second-order PSDM associated with the interior interpolation points. The last assertion is indispensable for optimally preconditioning the collocation systems.

**Theorem 2.2.** *There hold*

$$\boldsymbol{B}^{(k)} = \boldsymbol{D}^{(k)}\boldsymbol{B} = \boldsymbol{D}^{k}\boldsymbol{B} = \boldsymbol{D}\boldsymbol{B}^{(k-1)}, \quad k \geq 1, \tag{2.37}$$

*and*

$$\boldsymbol{D}_{\rm in}^{(2)}\boldsymbol{B}_{\rm in} = \boldsymbol{I}_{N-1}, \quad \widetilde{\boldsymbol{D}}^{(2)}\boldsymbol{B} = \boldsymbol{I}_{N+1}, \tag{2.38}$$

*where $\boldsymbol{I}_M$ is an $M \times M$ identity matrix, and*

$$\begin{aligned} \widetilde{\boldsymbol{D}}^{(2)} \text{ is } \boldsymbol{D}^{(2)} &\text{ with the first row replaced by } \vec{e}_0 = (1, \vec{0}) \\ &\text{and the last row replaced by } \vec{e}_N = (\vec{0}, 1). \end{aligned} \tag{2.39}$$

**Proof**: We first prove (2.37). For any $\phi \in \mathbb{P}_N$, we write $\phi(x) = \sum\limits_{p=0}^{N} \phi(x_p) L_p(x)$, so we have

$$\phi^{(k)}(x) = \sum_{p=0}^{N} \phi(x_p) L_p^{(k)}(x), \quad k \geq 1.$$

Taking $\phi = B_j (\in \mathbb{P}_N)$ and $x = x_i$, we obtain

$$b_{ij}^{(k)} = \sum_{p=0}^{N} d_{ip}^{(k)} b_{pj}, \quad k \geq 1, \tag{2.40}$$

which implies $\boldsymbol{B}^{(k)} = \boldsymbol{D}^{(k)}\boldsymbol{B}$. The second equality follows from (2.7), and the last identity in (2.37) is due to the recursive relation $\boldsymbol{B}^{(k-1)} = \boldsymbol{D}^{k-1}\boldsymbol{B}$.

We now turn to the proof of (2.38). It is clear that by (2.30), $b_{0j} = b_{Nj} = 0$ for $0 < j < N$ and $b_{ij}^{(2)} = \delta_{ij}$ for $0 < i, j < N$. Taking $k = 2$ in (2.40) leads to

$$\delta_{ij} = \sum_{p=1}^{N-1} d_{ip}^{(2)} b_{pj}, \quad 0 < i, j < N.$$

This yields $\boldsymbol{D}_{\rm in}^{(2)}\boldsymbol{B}_{\rm in} = \boldsymbol{I}_{N-1}$, from which the second statement follows directly. ■

In view of Thm. 2.2, we call $\boldsymbol{B}$ and $\boldsymbol{B}^{(1)}$ the second-order and first-order *pseudospectral integration matrices* (PSIMs), respectively.

### 2.2.1 Computation of PSIM on Gauss-Lobatto points

Let $\{x_j\}_{j=0}^N$ be a set of GL points. Now, we present stable algorithms for computing the matrices $\boldsymbol{B}$ and $\boldsymbol{B}^{(1)}$. For convenience, we introduce the integral operators:

$$\partial_x^{-1}u(x)=\int_{-1}^{x}u(t)\,\mathrm{d}t;\quad \partial_x^{-m}u(x)=\partial_x^{-1}\left(\partial_x^{1-m}u(x)\right),\quad m\geq 2. \tag{2.41}$$

By (2.11)–(2.12) and (2.24)–(2.25),

$$\partial_x^{-1}P_k(x)=\frac{1}{2k+1}(P_{k+1}(x)-P_{k-1}(x)),\quad k\geq 1;\quad \partial_x^{-1}P_0(x)=1+x. \tag{2.42}$$

and, for $k\geq 2$,

$$\begin{aligned}\partial_x^{-2}P_k(x)&=\frac{P_{k+2}(x)}{(2k+1)(2k+3)}-\frac{2P_k(x)}{(2k-1)(2k+3)}+\frac{P_{k-2}(x)}{(2k-1)(2k+1)};\\ \partial_x^{-2}P_0(x)&=\frac{(1+x)^2}{2};\quad \partial_x^{-2}P_1(x)=\frac{(1+x)^2(x-2)}{6}.\end{aligned} \tag{2.43}$$

Similarly, we find from (2.16)–(2.17) that

$$\begin{aligned}\partial_x^{-1}T_k(x)&=\frac{T_{k+1}(x)}{2(k+1)}-\frac{T_{k-1}(x)}{2(k-1)}-\frac{(-1)^k}{k^2-1},\quad k\geq 2;\\ \partial_x^{-1}T_0(x)&=1+x;\quad \partial_x^{-1}T_1(x)=\frac{x^2-1}{2}.\end{aligned} \tag{2.44}$$

Using (2.44) recursively yields

$$\begin{aligned}\partial_x^{-2}T_k(x)&=\frac{T_{k+2}(x)}{4(k+1)(k+2)}-\frac{T_k(x)}{2(k^2-1)}+\frac{T_{k-2}(x)}{4(k-1)(k-2)}\\ &\quad-\frac{(-1)^k(1+x)}{k^2-1}-\frac{3(-1)^k}{(k^2-1)(k^2-4)},\quad k\geq 3;\\ \partial_x^{-2}T_0(x)&=\frac{(1+x)^2}{2};\quad \partial_x^{-2}T_1(x)=\frac{(1+x)^2(x-2)}{6};\\ \partial_x^{-2}T_2(x)&=\frac{x(1+x)^2(x-2)}{6}.\end{aligned} \tag{2.45}$$

**Remark 2.5.** *Observe that $\partial_x^{-m}P_k(\pm 1)=0$ for all $k\geq m$ with $m=1,2$, while $\partial_x^{-m}T_k(1)$ may not vanish. The integrated Legendre and/or Chebyshev polynomials are used to construct well-conditioned spectral-Galerkin methods, hp-element methods* (see [99, 100, 64], and [17] for a review)*, and spectral integral methods* (see e.g., [29, 46, 62]).

**Proposition 2.1** (**Birkhoff interpolation at LGL points**). *Let* $\{x_j, \omega_j\}_{j=0}^N$ *be the LGL points and weights given in* (2.13). *Then the Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in Thm. 2.1 can be computed by*

$$B_j(x) = (\beta_{1j} - \beta_{0j})\frac{1+x}{2} + \sum_{k=0}^{N-2} \beta_{kj} \frac{\partial_x^{-2} P_k(x)}{\gamma_k}, \tag{2.46}$$

*where* $\gamma_k = 2/(2k+1)$, $\partial_x^{-2} P_k(x)$ *is given in* (2.43), *and*

$$\beta_{kj} = \left( P_k(x_j) - \frac{1-(-1)^{N+k}}{2} P_{N-1}(x_j) - \frac{1+(-1)^{N+k}}{2} P_N(x_j) \right) \omega_j. \tag{2.47}$$

*Moreover, we have*

$$B_j'(x) = \frac{\beta_{1j} - \beta_{0j}}{2} + \sum_{k=0}^{N-2} \beta_{kj} \frac{\partial_x^{-1} P_k(x)}{\gamma_k}, \tag{2.48}$$

*where* $\partial_x^{-1} P_k(x)$ *is given in* (2.42).

**Proof**: Since $B_j'' \in \mathbb{P}_{N-2}$, we expand it in terms of Legendre polynomials:

$$B_j''(x) = \sum_{k=0}^{N-2} \beta_{kj} \frac{P_k(x)}{\gamma_k} = \frac{1}{2} \sum_{k=0}^{N-2} (2k+1) \beta_{kj} P_k(x), \tag{2.49}$$

where

$$\beta_{kj} = \int_{-1}^{1} B_j''(x) P_k(x)\, \mathrm{d}x.$$

For $0 < j < N$, using (2.14) and (2.30) leads to

$$\beta_{kj} = \int_{-1}^{1} B_j''(x) P_k(x)\, \mathrm{d}x = [(-1)^k B_j''(-1) + B_j''(1)]\omega_0 + P_k(x_j)\omega_j. \tag{2.50}$$

Notice that the last identity of (2.50) is valid for all $k \le N+1$. Taking $k = N-1, N$, we obtain from (2.10) that the resulting integrals vanish, so we have the linear system of $B_j''(\pm 1)$:

$$\begin{aligned} &[(-1)^{N-1} B_j''(-1) + B_j''(1)]\omega_0 + P_{N-1}(x_j)\omega_j = 0, \\ &[(-1)^N B_j''(-1) + B_j''(1)]\omega_0 + P_N(x_j)\omega_j = 0. \end{aligned}$$

Therefore, we solve it and find that

$$B_j''(\pm 1) = -(\pm 1)^N \frac{\omega_j}{2\omega_0}(P_N(x_j) \pm P_{N-1}(x_j)), \quad 0 < j < N. \tag{2.51}$$

Inserting (2.51) into (2.50) yields the expression for $\beta_{kj}$ in (2.47).

Next, it follows from (2.49) that

$$B_j(x) = \sum_{k=0}^{N-2} \beta_{kj} \frac{\partial_x^{-2} P_k(x)}{\gamma_k} + C_1 + C_2(x+1), \tag{2.52}$$

where $C_1$ and $C_2$ are constants to be determined by $B_j(\pm 1) = 0$. Observe from (2.43) that $\partial_x^{-2} P_k(-1) = 0$ for $k \geq 0$ and $\partial_x^{-2} P_k(1) = 0$ for $k \geq 2$. This implies $C_1 = 0$ and

$$2C_2 = -\frac{\beta_{0j}}{\gamma_0} \partial_x^{-2} P_0(1) - \frac{\beta_{1j}}{\gamma_1} \partial_x^{-2} P_1(1) = \beta_{1j} - \beta_{0j}.$$

Thus, (2.46) follows. Finally, differentiating (2.46) leads to (2.48). ■

**Remark 2.6.** *It can also be shown that* (2.46) *gives*

$$B_j(x) = -\sum_{k=1}^{N-1} \frac{\omega_j}{\gamma_k} \partial_x^{-1} P_k(x_j) \partial_x^{-1} P_k(x). \tag{2.53}$$

**Proposition 2.2** (**Birkhoff interpolation at CGL points**). *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in Thm. 2.1 at CGL points* (2.18)*, can be computed by*

$$B_j(x) = \sum_{k=0}^{N-2} \beta_{kj} \left( \partial_x^{-2} T_k(x) - \frac{1+x}{2} \partial_x^{-2} T_k(1) \right), \tag{2.54}$$

*where* $\partial_x^{-2} T_k(x)$ *is given in* (2.45)*, and*

$$\beta_{kj} = \frac{2}{c_k N} \left( T_k(x_j) - \frac{1-(-1)^{N+k}}{2} T_{N-1}(x_j) - \frac{1+(-1)^{N+k}}{2} T_N(x_j) \right). \tag{2.55}$$

*Moreover, we have*

$$B_j'(x) = \sum_{k=0}^{N-2} \beta_{kj} \left( \partial_x^{-1} T_k(x) - \frac{\partial_x^{-2} T_k(1)}{2} \right), \tag{2.56}$$

*where* $\partial_x^{-1} T_k(x)$ *is computed by* (2.44)*. Here,* $c_0 = 2$ *and* $c_k = 1$ *for* $k \geq 1$ *as in* (2.15).

We omit the proof of Prop. 2.2, since it is very similar to that of Prop. 2.1.

**Remark 2.7.** *The formulas for evaluating integrated Legendre and/or Chebyshev polynomials are sparse and the coefficients decay* (cf. (2.24)). *This allows for stable computation of PSIM even for thousands of collocation points.*

In Fig. 2.1, we plot the first six Birkhoff interpolation basis polynomials at the GL points $\{x_j\}_{j=0}^5$ for both the Legendre (left) and Chebyshev (right) cases.

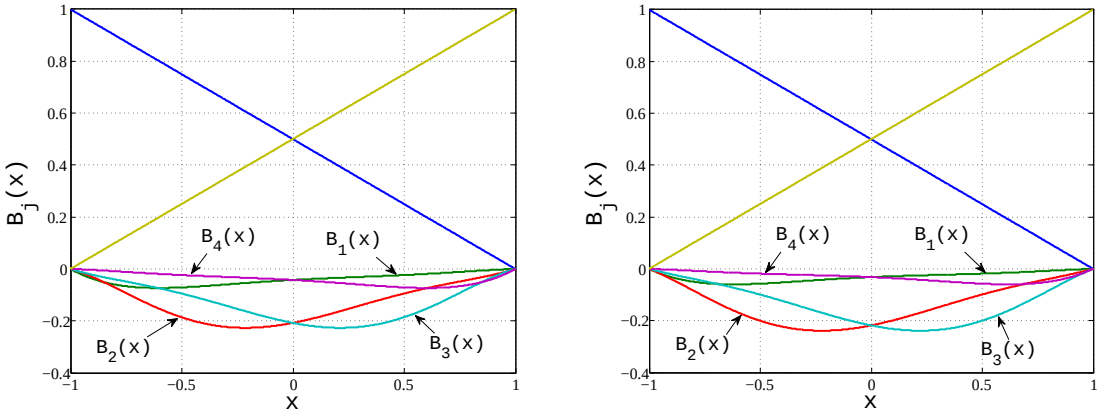



Figure 2.1: Plots of $\{B_j\}_{j=0}^5$. Left: Legendre; right: Chebyshev.

**Remark 2.8.** *Fig. 2.1 hints at a possible scaling problem regarding $B_j(x)$, $0 < j < N$, for large $N$, when used in an interpolation (cf. Rem. 2.1).*

*We consider Birkhoff interpolation on LGL points (Prop. 2.1), and note that, by* (2.53), *$B_j(x_i)\omega_i = B_i(x_j)\omega_j$. Thus, by* (2.36),

$$\max_I |B_j(x)| \geq \frac{1}{2}\left|\int_{-1}^{1} B_j(x)\,\mathrm{d}x\right| = \frac{1}{2}\left|\sum_{i=0}^{N} B_j(x_i)\omega_i\right| = \frac{\omega_j}{2}\left|\sum_{i=0}^{N} B_i(x_j)\right|$$
$$\geq \frac{\omega_j(1-x_j^2)}{4} \approx \frac{\pi}{4(N-1)}\sin^3\frac{j\pi + O(1)}{N-1}.$$

*The approximation in the last line follows from [101, eq. (3.301)].*

*So $\boldsymbol{B}_{\text{in}}$ may have only entries of order $O(N^{-1})$ or larger in the middle columns and $O(N^{-4})$ in the outer columns. $\boldsymbol{B}$ has $O(1)$ terms, so it is best to avoid using the full matrix for computations. Very large choices for $N$ give nearly singular $\boldsymbol{B}$ and $\boldsymbol{B}_{\text{in}}$, a possible source of round-off error in computing the interpolant.*

### 2.2.2 Collocation schemes

Consider the BVP

$$u''(x) + r(x)u'(x) + s(x)u(x) = f(x), \quad x \in I; \quad u(\pm 1) = u_{\pm}, \tag{2.57}$$

where the given functions $r, s, f \in C(I)$. Let $\{x_j\}_{j=0}^N$ be the set of Gauss-Lobatto points as in (2.26). Then the collocation scheme for (2.57) is to find $u_N \in \mathbb{P}_N$ such that

$$u_N''(x_i)+r(x_i)u_N'(x_i)+s(x_i)u_N(x_i) = f(x_i), \quad 0 < i < N; \quad u_N(\pm 1) = u_{\pm}. \tag{2.58}$$

As the Birkhoff interpolation polynomial of $u_N$ is itself, we have from (2.27) that

$$u_N(x) = u_- B_0(x) + u_+ B_N(x) + \sum_{j=1}^{N-1} u_N''(x_j) B_j(x). \tag{2.59}$$

Then the matrix form of (2.58) reads

$$\left(-\boldsymbol{I}_{N-1} + \boldsymbol{\Lambda}_r \boldsymbol{B}_{\text{in}}^{(1)} + \boldsymbol{\Lambda}_s \boldsymbol{B}_{\text{in}}\right) \vec{v} = \vec{f} - u_- \vec{v}_- - u_+ \vec{v}_+, \tag{2.60}$$

where $\boldsymbol{I}_{N-1}$ is the $(N-1) \times (N-1)$ identity matrix, and

$$\begin{aligned}
&\boldsymbol{\Lambda}_r = \operatorname{diag}(r(x_1), \ldots, r(x_{N-1})), \quad \boldsymbol{\Lambda}_s = \operatorname{diag}(s(x_1), \ldots, s(x_{N-1})),\\
&\vec{v} = (u_N''(x_1), \ldots, u_N''(x_{N-1}))^t, \qquad \vec{f} = (f(x_1), \ldots, f(x_{N-1}))^t,\\
&\vec{v}_{\pm} = \left(\pm\frac{r(x_1)}{2} + s(x_1)\frac{1 \pm x_1}{2}, \ldots, \pm\frac{r(x_{N-1})}{2} + s(x_{N-1})\frac{1 \pm x_{N-1}}{2}\right)^t.
\end{aligned}$$

It is seen that, under the new basis $\{B_j\}$, the matrix of the highest derivative is identity, and it also allows for exact imposition of boundary conditions.

In summary, we take the following steps to solve (2.58):

- Pre-compute $\boldsymbol{B}$ and $\boldsymbol{B}^{(1)}$ via the formulas in Prop. 2.1–2.2.
- Find $\vec{v}$ by solving the system (2.60).

- Recover $\vec{u} = (u_N(x_1), \ldots, u_N(x_{N-1}))^t$ from (2.59):

$$\vec{u} = \boldsymbol{B}_{\text{in}}\vec{v} + u_-\vec{b}_0 + u_+\vec{b}_N, \tag{2.61}$$

  where $\vec{b}_j = (B_j(x_1), \ldots, B_j(x_{N-1}))^t$ for $j = 0, N$.

**Remark 2.9.** *The unknowns under the new basis in* (2.59)–(2.60) *are the approximations to* $\{u''(x_j)\}$. *This situation is reminiscent of the spectral integration method* [62], *which is built upon the orthogonal polynomial expansion of* $u''(x)$. *Thus, the approach can be regarded as the collocation counterpart of the modal approach in* [62].

For comparison, we look at the usual collocation scheme (2.58) under the Lagrange basis. Write the solution of (2.58) as

$$u_N(x) = u_-L_0(x) + u_+L_N(x) + \sum_{j=1}^{N-1} u_N(x_j)L_j(x),$$

and insert it into (2.58), leading to the usual collocation system for (2.58):

$$\left(-\boldsymbol{D}_{\text{in}}^{(2)} + \boldsymbol{\Lambda}_r\boldsymbol{D}_{\text{in}}^{(1)} + \boldsymbol{\Lambda}_s\right)\vec{u} = \vec{f} + \vec{u}_B, \tag{2.62}$$

where $\vec{f}$ and $\vec{u}$ are as above, and $\vec{u}_B$ is the vector $\{u_-(d_{i0}^{(2)} - r(x_i)d_{i0}^{(1)}) + u_+(d_{iN}^{(2)} - r(x_i)d_{iN}^{(1)})\}_{i=1}^{N-1}$. It is known that the condition number of the coefficient matrix in (2.62) grows like $O(N^4)$.

Thanks to the property $\boldsymbol{B}_{\text{in}}\boldsymbol{D}_{\text{in}}^{(2)} = \boldsymbol{I}_{N-1}$ (see Thm. 2.2), the matrix $\boldsymbol{B}_{\text{in}}$ can be used to precondition the ill-conditioned system (2.62), leading to

$$\left(-\boldsymbol{I}_{N-1} + \boldsymbol{B}_{\text{in}}\boldsymbol{\Lambda}_r\boldsymbol{D}_{\text{in}}^{(1)} + \boldsymbol{B}_{\text{in}}\boldsymbol{\Lambda}_s\right)\boldsymbol{u} = \boldsymbol{B}_{\text{in}}(\vec{f} + \vec{u}_B). \tag{2.63}$$

**Remark 2.10.** *The right-hand side of* (2.63) *can be expanded:*

$$\boldsymbol{B}_{\text{in}}(\vec{f} + \vec{u}_B) = \boldsymbol{B}_{\text{in}}\vec{f} - u_-(\vec{b}_0 + \boldsymbol{B}_{\text{in}}\boldsymbol{\Lambda}_r\vec{d}_0^{(1)}) - u_+(\vec{b}_N + \boldsymbol{B}_{\text{in}}\boldsymbol{\Lambda}_r\vec{d}_N^{(1)}),$$

*where* $\vec{d}_j^{(k)} = (d_{1j}^{(k)}, d_{2j}^{(k)} \ldots, d_{N-1,j}^{(k)})^t$ *for* $j = 0, N$ *and* $k = 1$. *This improves the accuracy of the resulting computation.*

To illustrate, we compare the condition numbers of the linear systems between the Lagrange collocation (LCOL) scheme (2.62), the Birkhoff collocation (BCOL) scheme (2.60), the preconditioned LCOL (P-LCOL) scheme (2.63), and the preconditioned scheme from [47] (which improved that in [70]) (PLCOL), respectively. We also look at the number of iterations for solving the systems via BiCGSTAB in MatLab, and compare their convergence behavior.

We first consider the example

$$u''(x) - (1 + \sin x)u'(x) + e^x u(x) = f(x), \quad x \in (-1, 1); \quad u(\pm 1) = u_\pm, \quad (2.64)$$

with the exact solution $u(x) = e^{(x^2-1)/2}$. Observe from Table 2.1 that the condition numbers of the two new approaches are independent of $N$, and do not induce round-off errors. As already mentioned, the condition number of PLCOL in [47] grows like $O(\sqrt{N})$, and that of LCOL behaves like $O(N^4)$.

Table 2.1: Comparison of condition numbers, accuracy and iterations for (2.64).

| | LCOL (2.62) | | | PLCOL [47] | | | BCOL (2.60) | | | P-LCOL (2.63) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $N$ | Cond.# | Error | iters | Cond.# | Error | iters | Cond.# | Error | iters | Cond.# | Error | iters |
| Legendre | | | | | | | | | | | | |
| 64 | 3.97e+05 | 3.82e-14 | 286 | 80.1 | 1.44e-15 | 14 | 6.36 | 5.55e-16 | 10 | 2.86 | 1.67e-15 | 8 |
| 128 | 6.23e+06 | 4.42e-13 | 1251 | 156 | 2.66e-15 | 13 | 6.46 | 1.11e-15 | 10 | 2.86 | 2.44e-15 | 8 |
| 256 | 9.91e+07 | 3.95e-13 | 6988 | 308 | 2.33e-15 | 13 | 6.51 | 1.11e-15 | 11 | 2.86 | 2.55e-15 | 8 |
| 512 | 1.58e+09 | 1.02e-11 | 9457 | 612 | 3.77e-15 | 13 | 6.54 | 1.89e-15 | 11 | 2.86 | 4.77e-15 | 8 |
| 1024 | 2.52e+10 | 6.58e-12 | 9697 | 1220 | 1.03e-14 | 13 | 6.55 | 3.44e-15 | 11 | 2.86 | 1.15e-14 | 9 |
| Chebyshev | | | | | | | | | | | | |
| 64 | 7.23e+05 | 8.38e-14 | 285 | 79.6 | 1.67e-15 | 16 | 6.43 | 7.77e-16 | 10 | 2.86 | 1.44e-15 | 8 |
| 128 | 1.16e+07 | 2.87e-13 | 1304 | 156 | 3.44e-15 | 16 | 6.50 | 7.77e-16 | 10 | 2.86 | 4.22e-15 | 8 |
| 256 | 1.85e+08 | 9.74e-13 | 5868 | 308 | 8.44e-15 | 16 | 6.53 | 1.22e-15 | 11 | 2.86 | 6.55e-15 | 8 |
| 512 | 2.96e+09 | 4.51e-12 | 9987 | 611 | 4.22e-15 | 14 | 6.55 | 1.78e-15 | 11 | 2.86 | 3.44e-15 | 8 |
| 1024 | 4.73e+10 | 1.27e-11 | 9938 | 1219 | 5.22e-15 | 15 | 6.56 | 3.77e-15 | 11 | 2.86 | 6.00e-15 | 9 |

In Fig. 2.2, we depict the distribution of the eigenvalues (in magnitude) of the coefficient matrices of BCOL, PLCOL and P-LCOL for (2.64) with $N = 1024$

for both the Legendre and Chebyshev cases. Observe that almost all of them are concentrated around 1, though it appears nontrivial to show this rigorously (cf. Prop. 2.3).

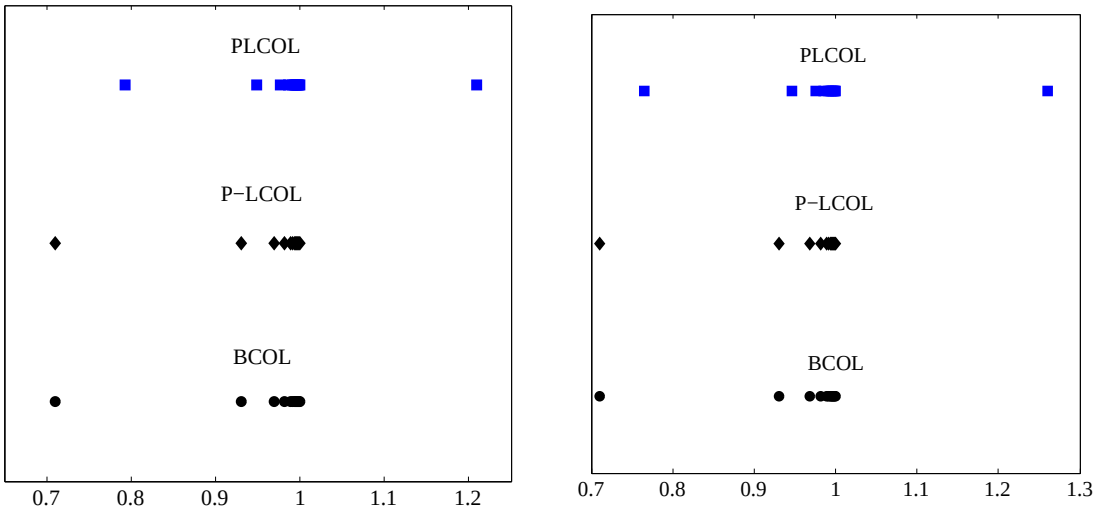



Figure 2.2: Distribution of magnitude of eigenvalues for the coefficient matrices of collocation schemes with $N = 1024$. Left: LGL; right: CGL.

We next consider (2.64) with $f \in C^1(\bar{I})$ and the exact solution $u \in C^3(\bar{I})$, given by

$$u(x) = \begin{cases} \cosh(x+1) - x^2/2 - x, & -1 \le x < 0, \\ \cosh(x+1) - \cosh(x) - x + 1, & 0 \le x \le 1. \end{cases}$$

Note that $u$ has Sobolev-regularity in $H^{4-\varepsilon}(I)$ with $\varepsilon > 0$. In Fig. 2.3, we graph the maximum point-wise errors for BCOL, LCOL and PLCOL, where the slope of the lines is approximately $-4$. We see that BCOL and PLCOL are free of round-off errors even for thousands of points, though the PLCOL system (in [47]) has a mildly-growing condition number.

Below, we have some insights into eigenvalues of the new collocation system for the operator: $\mathrm{d}^2/\,\mathrm{d}x^2 - \mu$ (i.e., Helmholtz (resp. modified Helmholtz) operator for $\mu < 0$ (resp. $\mu \ge 0$)) with Dirichlet boundary conditions.

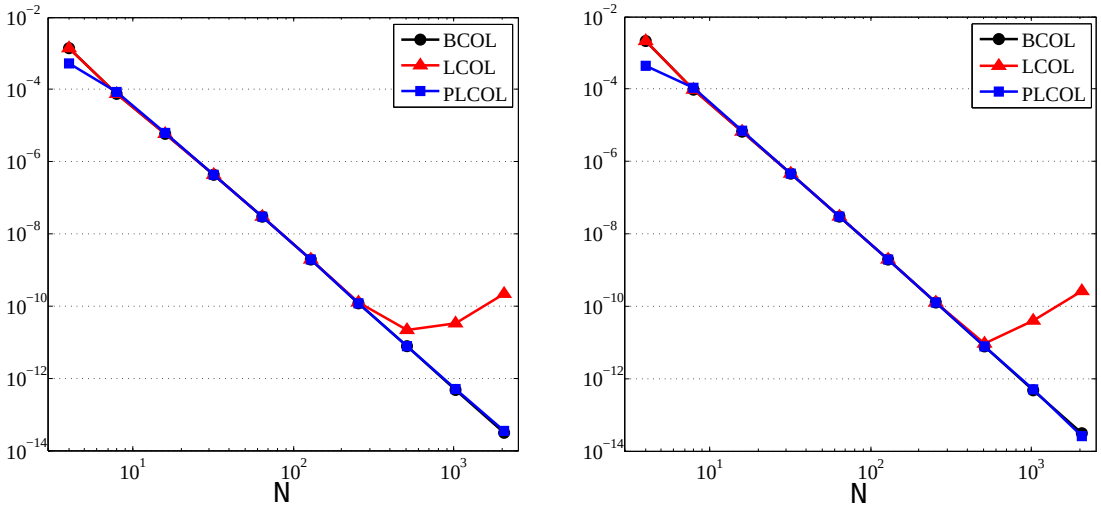



Figure 2.3: Comparison of maximum pointwise errors. Left: LGL; right: CGL.

**Proposition 2.3.** *In the LGL case, the eigenvalues of* $\boldsymbol{A} = \boldsymbol{I}_{N-1} - \mu \boldsymbol{B}_{\text{in}}$ *are all real and distinct, which are uniformly bounded. More precisely, for any eigenvalue* $\lambda$ *of* $\boldsymbol{A}$*, we have*

$$1+c_N\frac{4\mu\pi^2}{N^4} < \lambda < 1+\frac{4\mu}{\pi^2}, \ \textit{if } \mu \geq 0; \quad 1+\frac{4\mu}{\pi^2} < \lambda < 1+c_N\frac{4\mu\pi^2}{N^4}, \ \textit{if } \mu < 0, \tag{2.65}$$

*where* $c_N \approx 1$ *for large* $N$*.*

**Proof**: From [116, Thm. 7], we know that all eigenvalues of $\boldsymbol{D}_{\text{in}}^{(2)}$, denoted by $\{\lambda_{N,l}\}_{l=1}^{N-1}$, are real, distinct and negative, which we arrange as $\lambda_{N,N-1} < \cdots < \lambda_{N,1} < 0$. We diagonalize $\boldsymbol{D}_{\text{in}}^{(2)}$ as

$$\boldsymbol{D}_{\text{in}}^{(2)} = \boldsymbol{Q}\boldsymbol{\Lambda}_\lambda \boldsymbol{Q}^{-1},$$

where $\boldsymbol{Q}$ is formed by the eigenvectors and $\boldsymbol{\Lambda}_\lambda$ is the diagonal matrix of all eigenvalues. Since $\boldsymbol{B}_{\text{in}} = (\boldsymbol{D}_{\text{in}}^{(2)})^{-1}$ (cf. Thm. 2.2), we have

$$\boldsymbol{A} = \boldsymbol{Q}(\boldsymbol{I}_{N-1} - \mu\boldsymbol{\Lambda}_\lambda^{-1})\boldsymbol{Q}^{-1}.$$

Therefore, the eigenvalues of $\boldsymbol{A}$ are $\{1 - \mu\lambda_{N,l}^{-1}\}_{l=1}^{N-1}$, which are real and distinct. Then the bounds in (2.65) can be obtained from the properties: $-\lambda_{N,1} > \pi^2/4$

(see [116, p. 286, last line] and [13, Thm. 2.1]), and $-\lambda_{N,N-1} = c_N N^4/(4\pi^2)$ (see [116, Prop. 9]). ■

**Remark 2.11.** *We can obtain similar bounds for the CGL case by using the bounds for eigenvalues of* $\boldsymbol{D}_{\rm in}^{(2)}$ *in e.g.,* [115] *and* [19, Ch. 4.3]. *However, it appears open to conduct similar eigen-analysis for* (2.63) *and* (2.60) *with general variable coefficients.*

**Remark 2.12.** *As a consequence of* (2.65), *the condition number of* $\boldsymbol{A}$ *is independent of* $N$. *For example, it is uniformly bounded by* $1 + 4\mu/\pi^2$ *for* $\mu \geq 0$. *It is noteworthy that if* $\mu = -\omega^2$ *with* $\omega \gg 1$ (*i.e., Helmholtz equation with high wave-number*), *then the condition number behaves like* $O(\omega^2)$, *independent of* $N$.

### 2.2.3 Mixed boundary conditions

Consider the second-order BVP (2.57), equipped with mixed boundary conditions:

$$\mathcal{B}_-[u] := a_- u(-1) + b_- u'(-1) = c_-, \quad \mathcal{B}_+[u] := a_+ u(1) + b_+ u'(1) = c_+, \quad (2.66)$$

where $a_\pm, b_\pm$ and $u_\pm$ are given constants. We first assume that

$$d := 2a_+a_- - a_+b_- + a_-b_+ \neq 0, \quad (2.67)$$

which excludes Neumann boundary conditions (i.e., $a_- = a_+ = 0$) to be considered separately.

We associate (2.66) with the Birkhoff-type interpolation:

$$\begin{cases} \text{Find } p \in \mathbb{P}_N \text{ such that} \\ \mathcal{B}_-[p] = c_-, \quad p''(x_j) = c_j, 0 < j < N, \quad \mathcal{B}_+[p] = c_+, \end{cases} \quad (2.68)$$

where $\{c_\pm, c_j\}$ are given. As before, we look for the interpolation basis polynomials, still denoted by $\{B_j\}_{j=0}^N$, satisfying

$$\begin{aligned} &\mathcal{B}_-[B_0] = 1, \quad B_0''(x_i) = 0, \quad 0 < i < N, \quad \mathcal{B}_+[B_0] = 0; \\ &\mathcal{B}_-[B_j] = 0, \quad B_j''(x_i) = \delta_{ij}, \quad 0 < i < N, \quad \mathcal{B}_+[B_j] = 0, \quad 0 < j < N; \\ &\mathcal{B}_-[B_N] = 0, \quad B_N''(x_i) = 0, \quad 0 < i < N, \quad \mathcal{B}_+[B_N] = 1. \end{aligned} \quad (2.69)$$

Following the same lines as the proof of Thm. 2.1, we find that if $d \neq 0$,

$$B_0(x) = \frac{a_+}{d}(1-x) + \frac{b_+}{d}, \quad B_N(x) = \frac{a_-}{d}(1+x) - \frac{b_-}{d}, \tag{2.70}$$

and, for $0 < j < N$,

$$\begin{aligned} B_j(x) &= \int_{-1}^{x} (x-t)\ell_j(t)\,\mathrm{d}t - \frac{a_-(1+x)-b_-}{d}\int_{-1}^{1}(a_+(1-t)+b_+)\ell_j(t)\,\mathrm{d}t, \\ &= \int_{x}^{1} (t-x)\ell_j(t)\,\mathrm{d}t - \frac{a_+(1-x)+b_+}{d}\int_{-1}^{1}(a_-(1+t)-b_-)\ell_j(t)\,\mathrm{d}t. \end{aligned} \tag{2.71}$$

where $\{\ell_j\}$ are the Lagrange basis polynomials associated with the interior Gauss-Lobatto points as defined in Thm. 2.1. Thus, for any $u \in C^2(I)$, its interpolation polynomial is given by

$$p(x) = \big(\mathcal{B}_-[u]\big)B_0(x) + \sum_{j=1}^{N-1} u''(x_j)B_j(x) + \big(\mathcal{B}_+[u]\big)B_N(x), \quad x \in [-1,1]. \tag{2.72}$$

We can find formulas for computing $\{B_j\}_{j=1}^{N-1}$ on LGL and CGL points by using the same approach as in Prop. 2.1.

Armed with the new basis, we can impose mixed boundary conditions exactly, and the linear system resulting from the corresponding collocation scheme is well-conditioned. Here, we test the method on the second-order equation in (2.57) but with the mixed boundary conditions: $u(\pm 1) \pm u'(\pm 1) = u_\pm$.

In Table 2.2, we list the condition numbers of the usual collocation method (LCOL, where the boundary conditions are treated by the $\tau$-method), and the Birkhoff collocation method (BCOL, as in (2.60)) for both Legendre and Chebyshev cases. Once again, the new approach is well-conditioned.

The previous discussions exclude the Neumann boundary conditions, which need much care. Consider the Poisson equation:

$$u''(x) = f(x), \quad x \in I; \quad u'(\pm 1) = 0, \tag{2.73}$$

where $f$ is a continuous function such that $\int_{-1}^{1} f(x)\,\mathrm{d}x = 0$. Its solution is unique up to any additive constant. To ensure uniqueness, we supply (2.73) with an additional condition: $u(-1) = u_-$.

Table 2.2: Comparison of condition numbers.

| $N$ | $r \equiv 0$ and $s \equiv 1$ | | | | $r \equiv s \equiv -1$ | | | |
|---|---|---|---|---|---|---|---|---|
| | Chebyshev | | Legendre | | Chebyshev | | Legendre | |
| | BCOL | LCOL | BCOL | LCOL | BCOL | LCOL | BCOL | LCOL |
| 32 | 2.42 | 1.21e+05 | 2.45 | 6.66e+04 | 2.61 | 1.43e+05 | 2.61 | 7.87e+04 |
| 64 | 2.43 | 2.65e+06 | 2.45 | 1.41e+06 | 2.63 | 3.15e+06 | 2.63 | 1.68e+06 |
| 128 | 2.44 | 5.88e+07 | 2.45 | 3.09e+07 | 2.64 | 7.04e+07 | 2.64 | 3.70e+07 |
| 256 | 2.44 | 1.32e+09 | 2.45 | 6.88e+08 | 2.64 | 1.58e+09 | 2.64 | 8.26e+08 |
| 512 | 2.44 | 2.97e+10 | 2.44 | 1.54e+10 | 2.65 | 3.57e+10 | 2.65 | 1.86e+10 |
| 1024 | 2.44 | 6.71e+11 | 2.44 | 3.48e+11 | 2.65 | 8.08e+11 | 2.65 | 4.19e+11 |

Observe that the interpolation problem (2.68) is not well-posed if $\mathcal{B}_\pm[u]$ reduces to Neumann boundary conditions. Here, we consider the following special case of (2.2):

$$\begin{cases} \text{Find } p \in \mathbb{P}_{N+1} \text{ such that} \\ p(-1) = y_0^0, \quad p'(-1) = y_0^1, \quad p''(x_j) = y_j^2, \ 0 < j < N, \quad p'(1) = y_N^1, \end{cases} \tag{2.74}$$

where the data $\{y_j^m\}$ are given. However, this interpolation problem is only conditionally well-posed. For example, in the LGL and CGL cases, we have to assume that $N$ is odd.

As before, we look for basis polynomials, still denoted by $\{B_j\}_{j=0}^{N+1}$, such that for $0 < i < N$,

$$\begin{aligned}
&B_0(-1) = 0, && B_0'(-1) = 1, && B_0''(x_i) = 0, && B_0'(1) = 0; \\
&B_j(-1) = 0, && B_j'(-1) = 0, && B_j''(x_i) = \delta_{ij}, && B_j'(1) = 0, \qquad 0 < j < N; \\
&B_N(-1) = 0, && B_N'(-1) = 0, && B_N''(x_i) = 0, && B_N'(1) = 1; \\
&B_{N+1}(-1) = 1, && B_{N+1}'(-1) = 0, && B_{N+1}''(x_i) = 0, && B_{N+1}'(1) = 0.
\end{aligned}$$

Let $\ell_j(x)$ and $q_N(x)$ be as defined in (2.4) for $\{x_i\}_{i=1}^{N-1}$. Following the proof of

Thm. 2.1, we find that if $\int_{-1}^{1} q_N(t)\,\mathrm{d}t \neq 0$, we have

$$\begin{aligned} B_0(x) &= \frac{\displaystyle\int_x^1 (x-t)q_N(t)\,\mathrm{d}t}{\displaystyle\int_{-1}^1 q_N(t)\,\mathrm{d}t} + \frac{\displaystyle\int_{-1}^1 (t+1)q_N(t)\,\mathrm{d}t}{\displaystyle\int_{-1}^1 q_N(t)\,\mathrm{d}t} = 1 + x - B_N(x), \\ B_N(x) &= 1 + x - B_0(x) = \frac{\displaystyle\int_{-1}^x (x-t)q_N(t)\,\mathrm{d}t}{\displaystyle\int_{-1}^1 q_N(t)\,\mathrm{d}t}, \end{aligned} \tag{2.75}$$

$$B_{N+1}(x) \equiv 1,$$

and, for $0 < j < N$,

$$B_j(x) = \int_{-1}^x (x-t)\ell_j(t)\,\mathrm{d}t - \left(\frac{\displaystyle\int_{-1}^1 \ell_j(t)\,\mathrm{d}t}{\displaystyle\int_{-1}^1 q_N(t)\,\mathrm{d}t}\right)\int_{-1}^x (x-t)q_N(t)\,\mathrm{d}t. \tag{2.76}$$

**Remark 2.13.** *In the Legendre and Chebyshev cases, we have* $q_N(x) = P_N'(x)$ *or* $T_N'(x)$, *so by* (2.12) *and* (2.17),

$$\int_{-1}^1 q_N(t)\,\mathrm{d}t = \int_{-1}^1 P_N'(t)\,\mathrm{d}t = 1 - (-1)^N = \int_{-1}^1 T_N'(t)\,\mathrm{d}t,$$

*which is nonzero if and only if* $N$ *is odd.*

We plot in Fig. 2.4 the maximum point-wise errors of the usual collocation (LCOL) and Birkhoff collocation (BCOL) methods for (2.73), using the collocation schemes in Subsec. 2.2.2, appropriately modified, with the exact solution

$$u(x) = \cos(10x) - \cos(10).$$

Note that the condition numbers of systems obtained from BCOL are all 1. We see that BCOL outperforms LCOL as before.

**Remark 2.14.** *As remarked in* [47, Sec. 5.2], *how to implement the integration preconditioning technique therein for Neumann boundary conditions remains open.*

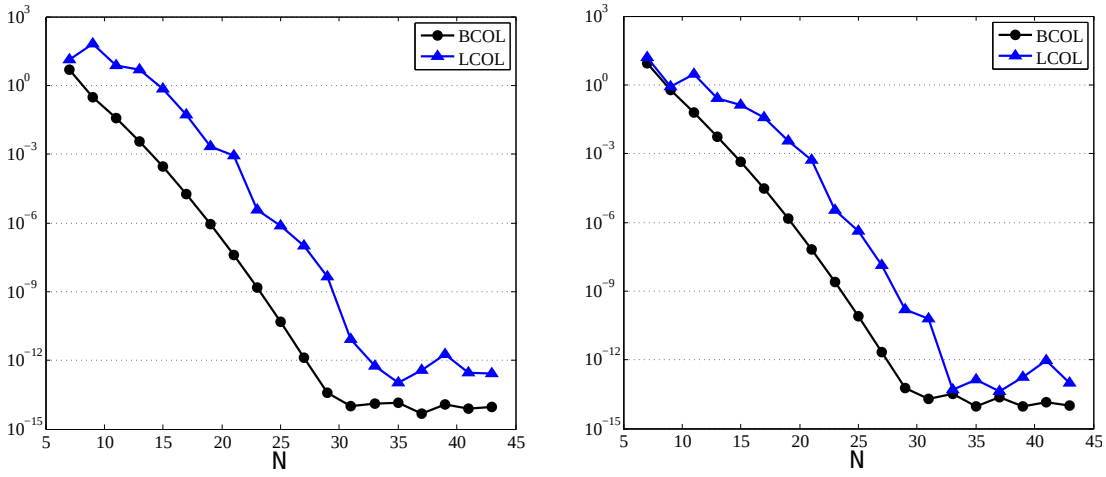



Figure 2.4: Comparison of maximum pointwise errors. Left: LGL; right: CGL.

## 2.3 Summary

We tackled the longstanding ill-conditioning issue of collocation/pseudospectral methods from a new perspective. Based on a suitable Birkhoff interpolation problem, we obtained dual-nature Birkhoff interpolation basis polynomials.

We have developed a collocation scheme under the new basis that is well-conditioned, in addition to an optimally-preconditioned collocation scheme under the usual Lagrange basis, for solving second-order BVPs on Legendre- and Chebyshev-Gauss-Lobatto points, with general boundary conditions. In both cases, the matrix corresponding to the highest derivative of the equation is diagonal or identity; the collocation scheme under the new basis could be viewed as the collocation analogue of the well-conditioned Galerkin method in [99].

In addition, we have shown that PSIMs are generated efficiently and in a stable manner, with consistent results through thousands of collocation points.

Chapter 3

# More About Well-Conditioned Collocation Methods

By associating a second-order BVP with a Birkhoff or Birkhoff-type interpolation problem, we showed in Ch. 2 that this led to the PSIM—an optimal integration preconditioner—and also produced a new basis, under which the collocation scheme was well-conditioned without involving differentiation matrices. This chapter is concerned with miscellaneous extensions of this methodology in various aspects and directions. More precisely,

- In Sec. 3.1, we apply a well-conditioned collocation method based on Birkhoff interpolation on Gauss-Radau points to initial value problems (IVPs), solving a first-order IVP with a highly-oscillatory exact solution.

- In Sec. 3.2, we apply a well-conditioned collocation method based on Birkhoff interpolation to BVPs of higher orders, for simulating time-dependent Korteweg-de Vries (KdV) equations with high-order differentiation in space. The concern is whether accurate and efficient collocation schemes generated by the Birkhoff interpolation basis compare well with the generalized

Lagrange interpolation collocation scheme (see (3.46)), and whether time-stepping methods will work with space-discretization handled by the Birkhoff interpolation method to efficiently produce a stable simulation.

- In Sec. 3.3, we apply (2.60)–(2.61) on general Gegenbauer-Gauss-Lobatto collocation points, allowing for flexibility in the selection of spectral collocation points.
- In Sec. 3.4, we develop a collocation method in two dimensions using PSIM from Ch. 2 and matrix decomposition techniques.
- In Sec. 3.5, we present a novel Birkhoff-type interpolation, indicated by a new differential operator, that, with the method in Ch. 2, produces a well-conditioned collocation method for problems on the half-line. Here, the method of Ch. 2, with typical Birkhoff interpolation, does not generate a well-conditioned collocation scheme, due to the behavior of the eigenvalues of the second-order PSDM associated with Laguerre functions.

In these cases, we highlight the robustness of the method in Ch. 2 in overcoming the ill-conditioning of the standard Lagrange interpolation collocation scheme.

## 3.1 IVPs

The purpose of this subsection is to extend the method of generating well-conditioned collocation schemes to solving initial-value problems (IVPs). An IVP differs from a BVP in that the boundary data that is given is the value of the solution function (and its derivatives) at a single initial point, such as for the first-order IVP

$$u'(x) + \gamma(x)u(x) = f(x), \quad x \in I = (-1, 1); \quad u(-1) = u_-, \tag{3.1}$$

where $\gamma$ and $f$ are given continuous functions on $I$, and $u_-$ is a given constant. Here, the collocation points used are (left) Gauss-Radau $-1 = x_0 < x_1 < \cdots < x_N < 1$, instead of Gauss-Lobatto, where $x_N = 1$.

We construct a Birkhoff interpolation basis on the Gauss-Radau points that can be used for a PSIM-based collocation method to solve (3.1). Consider the Birkhoff interpolation problem

$$\begin{cases} \text{Find } p \in \mathbb{P}_N \text{ such that for } u \in C^1(I), \\ p(-1) = u(-1), \quad p'(x_j) = u'(x_j),\ 1 \le j \le N. \end{cases} \tag{3.2}$$

One verifies readily that $p(x)$ can be uniquely expressed by

$$p(x) = u(-1)B_0(x) + \sum_{j=1}^{N} u'(x_j)B_j(x), \quad x \in [-1, 1], \tag{3.3}$$

if there exist $\{B_j\}_{j=0}^N \subseteq \mathbb{P}_N$ such that

$$\begin{aligned} &B_0(-1) = 1, \quad B_0'(x_i) = 0, \quad 1 \le i \le N; \\ &B_j(-1) = 0, \quad B_j'(x_i) = \delta_{ij},\ 1 \le i, j \le N. \end{aligned} \tag{3.4}$$

Like Thm. 2.1, we can derive

$$B_0(x) \equiv 1, \quad B_j(x) = \int_{-1}^{x} \ell_j(t)\,\mathrm{d}t,\ 1 \le j \le N, \tag{3.5}$$

where $\{\ell_j\}_{j=1}^N$ are the Lagrange basis polynomials (of degree $N-1$) associated with $N$ interior Gauss-Radau points $\{x_j\}_{j=1}^N$ (see (2.4)).

Let $\{L_j\}_{j=0}^N$ be the Lagrange basis polynomials associated with $\{x_j\}_{j=0}^N$, set $b_{ij} := B_j(x_i)$, and $d_{ij} := L_j'(x_i)$. Define

$$\begin{aligned} &\boldsymbol{B} = [b_{ij}]_{0 \le i,j \le N}, \quad \boldsymbol{B}_{\text{in}} = [b_{ij}]_{1 \le i,j \le N}, \\ &\boldsymbol{D} = [d_{ij}]_{0 \le i,j \le N}, \quad \boldsymbol{D}_{\text{in}} = [d_{ij}]_{1 \le i,j \le N}. \end{aligned} \tag{3.6}$$

Note that, for Ch. 2, the collocation points used are Gauss-Lobatto points, thus the points on the interior of $I$ are indexed $0 < j < N$, whereas, here, Gauss-Radau points are used for collocation points, thus the interior points are indexed $1 \le j \le N$.

Like (2.38), we have the following important properties.

**Theorem 3.1.** *There hold*

$$\boldsymbol{D}_{\rm in}\boldsymbol{B}_{\rm in} = \boldsymbol{I}_N, \quad \widetilde{\boldsymbol{D}}\boldsymbol{B} = \boldsymbol{I}_{N+1}, \tag{3.7}$$

*where* $\widetilde{\boldsymbol{D}}$ *is obtained by replacing the first row of* $\boldsymbol{D}$ *by* $\vec{e}_0 = (1, \vec{0})$.

**Proof**: For any $\phi \in \mathbb{P}_N$, we write

$$\phi(x) = \sum_{k=0}^{N} \phi(x_k) L_k(x), \quad \phi'(x) = \sum_{k=0}^{N} \phi(x_k) L_k'(x). \tag{3.8}$$

Taking $\phi = B_j$ and setting $x = x_i$ leads to

$$B_j'(x_i) = \sum_{k=0}^{N} B_j(x_k) L_k'(x_i) = \sum_{k=0}^{N} d_{ik} b_{kj}. \tag{3.9}$$

Thus, for $1 \le i, j \le N$, we obtain from $B_j'(x_i) = \delta_{ij}$ and $b_{0j} = 0$ that

$$\delta_{ij} = \sum_{k=1}^{N} d_{ik} b_{kj}, \quad 1 \le i, j \le N, \tag{3.10}$$

which implies $\boldsymbol{D}_{\rm in}\boldsymbol{B}_{\rm in} = \boldsymbol{I}_N$.

Notice that the first row of $\boldsymbol{B}$ is $\vec{e}_0 = (1, \vec{0})$ (cf. (3.4)), so we verify from (3.9)–(3.10) that $\widetilde{\boldsymbol{D}}\boldsymbol{B} = \boldsymbol{I}_{N+1}$. ■

### 3.1.1 Computation of PSIM on Gauss-Radau points

As with Prop. 2.1–2.2, we provide formulas to compute $\{B_j\}$ for Chebyshev- and Legendre-Gauss-Radau interpolation. To avoid repetition, we just give the derivation for the CGR case.

**Proposition 3.1 (Birkhoff interpolation at CGR points).** *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=0}^N$ *in* (3.4) *at* $\{x_j = -\cos(jh)\}_{j=0}^N$, $h = 2\pi/(2N+1)$, *CGR points, are computed by*

$$B_0(x) \equiv 1; \quad B_j(x) = \sum_{k=0}^{N-1} \beta_{kj} \partial_x^{-1} T_k(x), \ 1 \le j \le N, \tag{3.11}$$

*where* $\partial_x^{-1}T_k(x)$ *is defined in* (2.44)*, and*

$$\beta_{kj} = \frac{4}{c_k(2N+1)}(T_k(x_j) - (-1)^{N+k}T_N(x_j)), \tag{3.12}$$

*with* $c_0 = 2$ *and* $c_k = 1$ *for* $k \geq 1$.

**Proof**: Writing

$$B_j'(x) = \sum_{k=0}^{N-1} \beta_{kj} T_k(x),$$

we derive from (2.15) that

$$\beta_{kj} = \frac{2}{c_k\pi}\int_{-1}^{1} \frac{B_j'(x)T_k(x)}{\sqrt{1-x^2}}\,\mathrm{d}x = \frac{2}{c_k\pi}\left(B_j'(-1)T_k(-1)\frac{h}{2} + T_k(x_j)h\right),$$

where we also used (3.4), the CGR quadrature weights $\omega_0 = h/2$, $\omega_j = h$, $1 \leq j \leq N$, and that the CGR quadrature has the exactness

$$\int_{-1}^{1} \frac{\phi(x)}{\sqrt{1-x^2}}\,\mathrm{d}x = \sum_{j=0}^{N} \phi(x_j)\omega_j, \quad \forall \phi \in \mathbb{P}_{2N}.$$

Taking $k = N$, we have from (2.15) and (2.17) that

$$\beta_{Nj} = 0, \quad B_j'(-1) = (-1)^{N+1}2T_N(x_j),\ 1 \leq j \leq N.$$

Thus (3.12) follows. Then direct integration leads to

$$B_j(x) = \sum_{k=0}^{N-1} \beta_{kj}\partial_x^{-1}T_k(x) + C.$$

Since $\partial_x^{-1}T_k(-1) = 0$, we find $C = 0$ from $B_j(-1) = 0$ in (3.4). ∎

We can derive the formulas for computing $\{B_j\}$ at LGR points in a very similar fashion.

**Proposition 3.2 (Birkhoff interpolation at LGR points).** *Let* $\{x_j, \omega_j\}_{j=0}^N$ *be the (left) LGR quadrature points, i.e. zeros of* $P_N(x) + P_{N+1}(x)$ *with* $x_0 = -1$*, and the associated quadrature weights, given by*

$$\omega_j = \frac{1}{(N+1)^2}\frac{1-x_j}{[P_N(x_j)]^2},\ 0 \leq j \leq N.$$

*Then the Birkhoff interpolation basis polynomials* $\{B_j\}_{j=0}^N$ *in* (3.4) *can be computed by*

$$B_0(x) \equiv 1; \quad B_j(x) = \sum_{k=0}^{N-1} \beta_{kj} \partial_x^{-1} P_k(x), \ 1 \le j \le N, \tag{3.13}$$

*where* $\partial_x^{-1} P_k(x)$ *is given in* (2.42), *and*

$$\beta_{kj} = \frac{2k+1}{2(N+1)^2} \frac{1-x_j}{[P_N(x_j)]^2} (P_k(x_j) - (-1)^{N+k} P_N(x_j)). \tag{3.14}$$

In Fig. 3.1, we plot the first six Birkhoff interpolation basis polynomials at the GR points $\{x_j\}_{j=0}^5$ for both the Legendre (left) and Chebyshev (right) cases.

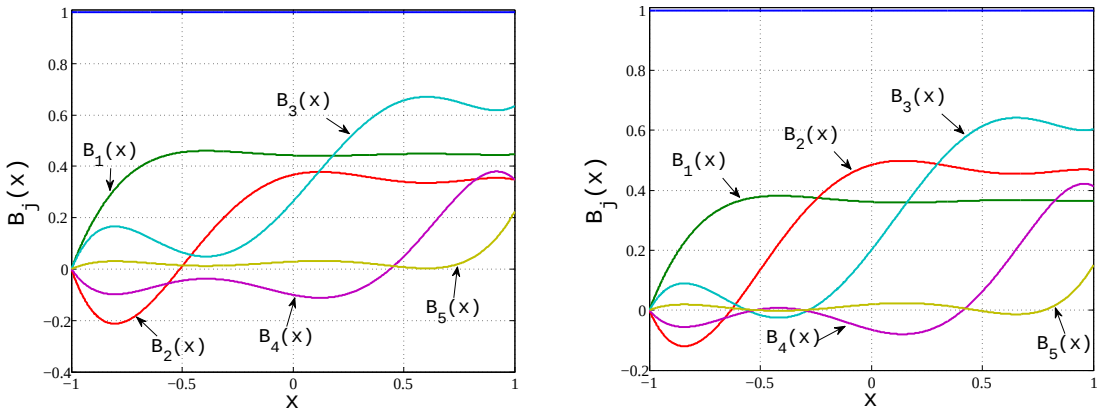



Figure 3.1: Plots of $\{B_j\}_{j=0}^5$. Left: Legendre; right: Chebyshev.

### 3.1.2 Collocation schemes

With the new basis at our disposal, we now apply it to solve first-order IVPs such as (3.1). The collocation scheme at Gauss-Radau points for (3.1) is to find $u_N \in \mathbb{P}_N$ such that

$$u_N'(x_j) + \gamma(x_j) u_N(x_j) = f(x_j), \quad 1 \le j \le N; \quad u_N(-1) = u_-. \tag{3.15}$$

The matrix form of (3.15) under the Lagrange interpolation basis $\{L_j\}_{j=0}^N$, reads

$$(\boldsymbol{D}_{\text{in}} + \boldsymbol{\Lambda}_N)\vec{u} = \vec{f} - u_- \vec{d}_0, \tag{3.16}$$

where $\boldsymbol{D}_{\text{in}}$ is defined in (3.6), and

$$\begin{aligned} &\vec{u} = (u_N(x_1), \ldots, u_N(x_N))^t, \quad \vec{f} = (f(x_1), \ldots, f(x_N))^t, \\ &\vec{d_0} = (L_0'(x_1), \ldots, L_0'(x_N))^t, \quad \boldsymbol{\Lambda}_N = \text{diag}(\gamma(x_1), \ldots, \gamma(x_N)). \end{aligned} \tag{3.17}$$

Under the new basis $\{B_j\}_{j=0}^N$, we find from (3.4) the matrix form:

$$(\boldsymbol{I}_N + \boldsymbol{\Lambda}_N \boldsymbol{B}_{\text{in}})\vec{v} = \vec{f} - u_- \bar{\gamma}, \tag{3.18}$$

where $\boldsymbol{B}_{\text{in}}$ is defined in (3.6), $\vec{f}$ is the same as in (3.17), and

$$\vec{v} = (u_N'(x_1), \ldots, u_N'(x_N))^t, \quad \bar{\gamma} = (\gamma(x_1), \ldots, \gamma(x_N))^t. \tag{3.19}$$

We intend to compare two collocation approaches under the Lagrange basis (LCOL) (3.16), and the new Birkhoff basis (BCOL) (3.18). The involved linear systems can be formed in the same manner as for the second-order BVPs in Subsec. 2.2.2. In Table 3.1, the condition numbers of LCOL and BCOL with $\gamma = 1, -\sin x$ and various $N$ are given. As what we have observed from previous section, the condition numbers of BCOL are independent of $N$, while those of LCOL grow like $N^2$.

Table 3.1: Comparison of the condition numbers.

| $N$ | $\gamma = 1$ | | | | $\gamma = -\sin x$ | | | |
|---|---|---|---|---|---|---|---|---|
| | Chebyshev | | Legendre | | Chebyshev | | Legendre | |
| | BCOL | LCOL | BCOL | LCOL | BCOL | LCOL | BCOL | LCOL |
| 128 | 2.35 | 5.65e+03 | 2.34 | 8.45e+03 | 2.77 | 1.35e+04 | 2.76 | 2.02e+04 |
| 256 | 2.35 | 2.25e+04 | 2.35 | 3.59e+04 | 2.77 | 5.38e+04 | 2.77 | 8.58e+04 |
| 512 | 2.35 | 8.98e+04 | 2.35 | 1.52e+05 | 2.77 | 2.15e+05 | 2.77 | 3.63e+05 |
| 1024 | 2.35 | 3.59e+05 | 2.35 | 6.40e+05 | 2.77 | 8.58e+05 | 2.77 | 1.53e+06 |

We next consider (3.1) with $\gamma(x) = -\sin x, f(x) = 20\sin(500x^2)$ and an oscillatory solution:

$$u(x) = 20\exp(-\cos(x)) \int_{-1}^{x} \exp(\cos(t))\sin(500t^2)\,\mathrm{d}t. \tag{3.20}$$

In Fig. 3.2 (left), we plot the exact solution (3.20) at 2000 evenly-spaced points against the numerical solution obtained by BCOL with $N = 640$. In Fig. 3.2 (right), we plot the maximum pointwise errors of LCOL and BCOL for the Chebyshev case. It indicates that, even for large $N$, BCOL is very stable.

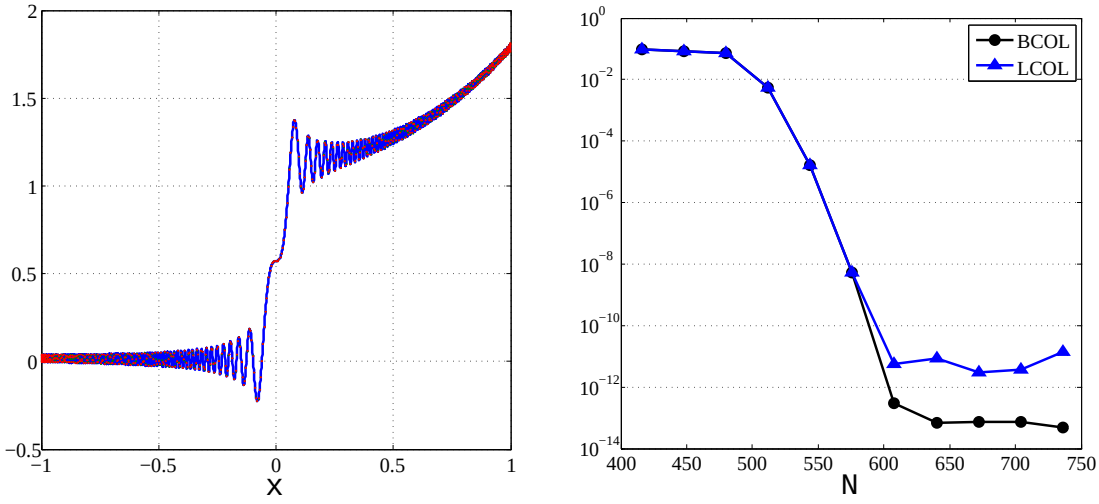



Figure 3.2: Left: exact solution versus numerical solution. Right: comparison of numerical errors (Chebyshev).

**Remark 3.1.** *It is straightforward to extend this construction to that for higher-order IVPs, such as, for* $k > 1$,

$$u^{(k)}(x) + \ l.o.t. \ = f(x), \quad x \in I; \quad u^{(i)}(-1) = u_i, \quad 0 \le i < k, \tag{3.21}$$

*where* $f$ *is a given continuous function on* $I$, *l.o.t. stands for lower-order terms, possibly with variable coefficients, and* $u_i$, $0 \le i < k$ *are given constants.*

*To determine the Birkhoff interpolation basis on Gauss-Radau points, we consider the Birkhoff interpolation problem*

$$\begin{cases} \text{Find } p \in \mathbb{P}_{N+k-1} \text{ such that for } u \in C^k(I), \\ p^{(i)}(-1) = u^{(i)}(-1), \ 0 \le i < k; \quad p^{(k)}(x_j) = u^{(k)}(x_j), \ 1 \le j \le N. \end{cases} \tag{3.22}$$

*The Birkhoff interpolation polynomial $p$ of $u$ can be uniquely determined by*

$$p(x) = \sum_{i=0}^{k-1} u^{(i)}(-1) B_{-i}(x) + \sum_{j=1}^{N} u^{(k)}(x_j) B_j(x), \quad x \in [-1, 1], \tag{3.23}$$

*if one can find $\{B_j\}_{j=1-k}^{N} \subseteq \mathbb{P}_{N+k-1}$, such that for $0 \le \ell < k$ and $1 \le i \le N$,*

$$\begin{aligned} &\text{for } -k < j \le 0: \ B_j^{(\ell)}(-1) = \delta_{\ell(-j)}, \ B_j'(x_i) = 0; \\ &\text{for } 1 \le j \le N: \quad B_j^{(\ell)}(-1) = 0, \qquad B_j'(x_i) = \delta_{ij}. \end{aligned} \tag{3.24}$$

*For CGR points, can be verified that*

$$B_{-i}(x) = \partial_x^{-i} T_0(x), \ 0 \le i < k, \quad B_j(x) = \sum_{m=0}^{N-1} \beta_{mj} \partial_x^{-k} T_m(x), \ 1 \le j \le N,$$

*where $\partial_x^{-k} T_m(x) = \partial_x^{-1}[\partial_x^{1-k} T_m(x)]$ as defined recursively in* (2.41) *and $\beta_{mj}$ is given in* (3.12)*. The collocation scheme and PSIM-based linear system produced are similar to* (3.15) *and* (3.18)*.*

### 3.1.3 Interpolation error estimates

We revisit the interpolation problem (3.2) at LGR points. For any $u \in C^1(I)$, we denote its Birkhoff interpolant by $\mathrm{I}_N^B u$, which satisfies the interpolation conditions:

$$\left(\mathrm{I}_N^B u\right)'(x_j) = u'(x_j), \quad 1 \le j \le N; \quad \left(\mathrm{I}_N^B u\right)(-1) = u(-1). \tag{3.25}$$

Let $\mathrm{I}_{N-1}^G$ be the Lagrange-Gauss interpolation operator at the interior LGR points $\{x_j\}_{j=1}^N$, such that, for any $v \in C(I)$, $\left(\mathrm{I}_{N-1}^G v\right)(x_j) = v(x_j)$, $1 \le j \le N$. Then we find from (3.25) that

$$\left(\mathrm{I}_N^B u\right)'(x) = \left(\mathrm{I}_{N-1}^G u'\right)(x) \in \mathbb{P}_{N-1}, \quad \forall\, x \in [-1, 1]. \tag{3.26}$$

Note that $\{x_j\}_{j=1}^N$ are zeros of the Jacobi polynomial $P_N^{(0,1)}(x)$. Define $L^2_\omega$-space with $\omega = 1 - x$ with the norm $\|\cdot\|_\omega$. Using (3.26) and [101, Thm. 3.41], we obtain

that, for $2 \le m \le N+1$,

$$\begin{aligned} \|(\mathrm{I}_N^B u - u)'\|_\omega &= \|\,\mathrm{I}_{N-1}^G u' - u'\|_\omega \\ &\le c\sqrt{\frac{(N-m+2)!}{N!}}(N+m)^{-\frac{m}{2}}\|(1-x^2)^{(m-1)/2}\partial_x^m u\|_\omega, \end{aligned} \tag{3.27}$$

where $c$ is a positive constant independent of $m$, $N$ and $u$. Note that if $m$ is fixed, the order of convergence is $O(N^{1-m})$.

**Remark 3.2.** *By* (3.25) *and* (3.26),

$$\left(\mathrm{I}_N^B u - u\right)(x) = \int_{-1}^{x} \left[\left(\mathrm{I}_{N-1}^G u'\right)(t) - u'(t)\right]\,\mathrm{d}t.$$

*We expect the convergence* $\|\,\mathrm{I}_N^B u - u\|_{L^2} = O(N^{-m})$. *However, it appears challenging to obtain this optimal order, though we observed such a convergence in computation.*

## 3.2 Higher-order BVPs

We illustrate, in this section, the existence and construction of a Birkhoff interpolation basis for PSIM-based collocation methods for higher-order BVPs. We also demonstrate that the solution of time-dependent third-order and fifth-order Korteweg-de Vries problems can be discretized in space in a stable manner by using the well-conditioned collocation method.

### 3.2.1 Third-order BVPs

Consider the following problem: for $x \in I = (-1, 1)$,

$$\begin{cases} -u'''(x) + r(x)u''(x) + s(x)u'(x) + t(x)u(x) = f(x); \\ u(\pm 1) = u_\pm, \quad u'(1) = u_1, \end{cases} \tag{3.28}$$

where $r$, $s$, $t$ and $f$ are given continuous functions on $I$, and $u_-$, $u_+$ and $u_1$ are given constants.

The collocation scheme at Gauss-Lobatto points $-1 = x_0 < x_1 < \cdots < x_N = 1$ for (3.28) is to find $u_N \in \mathbb{P}_{N+1}$ such that for $0 < j < N$,

$$\begin{cases} -u_N'''(x_j) + r(x_j)u_N''(x_j) + s(x_j)u_N'(x_j) + t(x_j)u_N(x_j) = f(x_j), \\ u_N(\pm 1) = u_\pm, \quad u_N'(1) = u_1. \end{cases} \tag{3.29}$$

Correspondingly, we consider the Birkhoff interpolation problem:

$$\begin{cases} \text{Find } p \in \mathbb{P}_{N+1} \text{ such that for } u \in C^3(I), \\ p(-1) = u(-1); \quad p'''(x_j) = u'''(x_j),\ 0 < j < N; \\ p(1) = u(1); \quad p'(1) = u'(1). \end{cases} \tag{3.30}$$

As with (2.27), the Birkhoff interpolation polynomial $p$ of $u$ can be uniquely determined, for $x \in [-1, 1]$, by

$$p(x) = u(-1)B_0(x) + \sum_{j=1}^{N-1} u'''(x_j)B_j(x) + u(1)B_N(x) + u'(1)B_{N+1}(x).$$

We find that

$$B_0(x) = \frac{(1-x)^2}{4}, \quad B_N(x) = \frac{(1+x)(3-x)}{4}, \quad B_{N+1}(x) = \frac{x^2-1}{2}, \tag{3.31}$$

and the "interior" basis elements $\{B_j\}_{j=1}^{N-1} \subseteq \mathbb{P}_{N+1}$ satisfy

$$B_j(-1) = 0, \quad B_j(1) = 0, \quad B_j'(1) = 0, \quad B_j'''(x_i) = \delta_{ij},\ 0 < i < N. \tag{3.32}$$

The following two propositions describe the formulas for (3.32) at LGL and CGL points. They are very stable in computation.

**Proposition 3.3 (Birkhoff interpolation at CGL points).** *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in* (3.32) *at CGL points* (2.18) *are*

$$B_j(x) = \sum_{k=0}^{N-2} \beta_{kj} \left( \partial_x^{-3} T_k(x) - [\partial_x^{-3} T_k(1)] B_N(x) - [\partial_x^{-2} T_k(1)] B_{N+1}(x) \right), \tag{3.33}$$

*where* $\partial_x^{-3} T_k(x) = \partial_x^{-1}[\partial_x^{-2} T_k(x)]$ *as given by* (2.41) *and* (2.45)*, and* $\beta_{kj}$ *given in* (2.55).

**Proof**: (3.33) gives

$$B_j'''(x_i) = \sum_{k=0}^{N-2} \beta_{kj} T_k(x_i) = \delta_{ij},$$

and $B_j(\pm 1) = B_j'(1) = 0$ from

$$F_k(x) = \partial_x^{-3} T_k(x) - [\partial_x^{-3} T_k(1)] B_N(x) - [\partial_x^{-2} T_k(1)] B_{N+1}(x),$$

and $F_k(\pm 1) = F_k'(1) = 0$. ∎

**Proposition 3.4** (**Birkhoff interpolation at LGL points**). *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in* (3.32) *at LGL points, i.e. roots of* $(1-x^2)P_N'(x)$, *are*

$$B_j(x) = \sum_{k=0}^{N-2} \frac{\beta_{kj}}{\gamma_k} \left( \partial_x^{-3} P_k(x) - [\partial_x^{-3} P_k(1)] B_N(x) - [\partial_x^{-2} P_k(1)] B_{N+1}(x) \right), \quad (3.34)$$

*where*

$$\gamma_k = \frac{2}{2k+1}, \quad \partial_x^{-3} P_k(x) = \partial_x^{-1} [\partial_x^{-2} P_k(x)],$$

*as given by* (2.41) *and* (2.43), *and* $\beta_{kj}$ *given in* (2.47).

**Proof**: (3.34) gives

$$B_j'''(x_i) = \sum_{k=0}^{N-2} \frac{\beta_{kj}}{\gamma_k} P_k(x_i) = \delta_{ij},$$

and $B_j(\pm 1) = B_j'(1) = 0$ from

$$F_k(x) = \partial_x^{-3} P_k(x) - [\partial_x^{-3} P_k(1)] B_N(x) - [\partial_x^{-2} P_k(1)] B_{N+1}(x),$$

and $F_k(\pm 1) = F_k'(1) = 0$. ∎

With these bases for LGL and CGL points, we can compute the PSIMs

$$\boldsymbol{B}^{(k)} = [B_j^{(k)}(x_i)]_{0 \le j \le N+1}^{0 \le i \le N}, \quad \boldsymbol{B}_{\text{in}}^{(k)} = [B_j^{(k)}(x_i)]_{0<i,j<N},$$

with $\boldsymbol{B} = \boldsymbol{B}^{(0)}$ and $\boldsymbol{B}_{\text{in}} = \boldsymbol{B}_{\text{in}}^{(0)}$, and apply them to solve (3.29).

Let $u_N$ be the solution of (3.29). Under the new basis,

$$u_N(x) = u_- B_0(x) + \sum_{j=1}^{N-1} v_j B_j(x) + u_+ B_N(x) + u_1 B_{N+1}(x),$$

where $v_j = u_N'''(x_j)$. We obtain the system

$$-v_i + \sum_{j=1}^{N-1} v_j \gamma_{ij} = f(x_i) - u_- \gamma_{i0} - u_+ \gamma_{iN} - u_1 \gamma_{i(N+1)}, \quad 0 < i < N,$$

where

$$\gamma_{ij} = r(x_i) B_j''(x_i) + s(x_i) B_j'(x_i) + t(x_i) B_j(x_i).$$

This, in matrix form, is

$$(-\boldsymbol{I}_{N-1} + \boldsymbol{\Lambda}_r \boldsymbol{B}_{\text{in}}^{(2)} + \boldsymbol{\Lambda}_s \boldsymbol{B}_{\text{in}}^{(1)} + \boldsymbol{\Lambda}_t \boldsymbol{B}_{\text{in}}) \vec{v} = \vec{f} - u_- \vec{w}_0 - u_+ \vec{w}_N - u_1 \vec{w}_{N+1}, \quad (3.35)$$

where $\boldsymbol{I}_{N-1}$ is the $(N-1) \times (N-1)$ identity matrix, and

$$\begin{aligned}
&\boldsymbol{\Lambda}_r = \text{diag}(r(x_1), \ldots, r(x_{N-1})), \quad && \boldsymbol{\Lambda}_s = \text{diag}(s(x_1), \ldots, s(x_{N-1})), \\
&\boldsymbol{\Lambda}_t = \text{diag}(t(x_1), \ldots, t(x_{N-1})), && \vec{f} = (f(x_1), \ldots, f(x_{N-1}))^t, \\
&\vec{w}_j = \boldsymbol{\Lambda}_r \vec{b}_j^{(2)} + \boldsymbol{\Lambda}_s \vec{b}_j^{(1)} + \boldsymbol{\Lambda}_t \vec{b}_j^{(0)}, && \vec{b}_j^{(k)} = (B_j^{(k)}(x_1), \ldots, B_j^{(k)}(x_{N-1}))^t, \\
&\vec{v} = (u_N'''(x_1), \ldots, u_N'''(x_{N-1}))^t.
\end{aligned}$$

Then we recover the approximation to $u$ by

$$\vec{u} = (u_N(-1), \ldots, u_N(1))^t = \boldsymbol{B}_{\text{in}} \vec{v} + u_- \vec{b}_0 + u_+ \vec{b}_N + u_1 \vec{b}_{N+1}. \quad (3.36)$$

Here, we just tabulate in Table 3.2 the condition numbers of the coefficient matrices in (3.35) for CGL points. In all cases, the condition numbers are independent of $N$.

Table 3.2: Condition numbers of (3.28) on CGL points.

| $N$ | $r \equiv s \equiv 0$, $t \equiv 1$ | $r \equiv 0$, $s \equiv t \equiv 1$ | $s \equiv 0$, $r \equiv t \equiv 1$ | $r \equiv s \equiv t \equiv 1$ |
|---|---|---|---|---|
| 128 | 1.16 | 1.56 | 2.22 | 1.80 |
| 256 | 1.16 | 1.56 | 2.22 | 1.80 |
| 512 | 1.16 | 1.56 | 2.23 | 1.80 |
| 1024 | 1.16 | 1.56 | 2.23 | 1.80 |

The solver (3.35)–(3.36) will be used for solving the third-order KdV equation in Subsec. 3.2.3. We see that it is stable and of spectral accuracy.

### 3.2.2 Fifth-order BVPs

We challenge the collocation method for an even higher-order BVP, which will also be used as a solver in space for the fifth-order KdV equation in Subsec. 3.2.3.

Consider the fifth-order problem:

$$\begin{cases} u^{(5)}(x) + a(x)u'(x) + b(x)u(x) = f(x), \quad x \in I = (-1,1); \\ u(\pm 1) = u'(\pm 1) = u''(1) = 0, \end{cases} \tag{3.37}$$

where $a$, $b$ and $f$ are given continuous functions on $I$.

The collocation scheme at Gauss-Lobatto points for (3.37) is to find $u_N \in \mathbb{P}_{N+3}$ such that

$$\begin{cases} u_N^{(5)}(x_j) + a(x_j)u_N'(x_j) + b(x_j)u_N(x_j) = f(x_j), \quad 0 < j < N; \\ u_N(\pm 1) = u_N'(\pm 1) = u_N''(1) = 0. \end{cases} \tag{3.38}$$

The associated Birkhoff interpolation is

$$\begin{cases} \text{Find } p \in \mathbb{P}_{N+3} \text{ such that for } u \in C^5(I), \\ p(-1) = u(-1); \;\; p'(-1) = u'(-1); \;\; p^{(5)}(x_j) = u^{(5)}(x_j), \; 0 < j < N; \\ p(1) = u(1); \quad p'(1) = u'(1); \quad p''(1) = u''(1). \end{cases} \tag{3.39}$$

The Birkhoff interpolation polynomial $p$ of $u$ can be uniquely determined by

$$\begin{aligned} p(x) = {} & u'(-1)B_{-1}(x) + u(-1)B_0(x) + \sum_{j=1}^{N-1} u^{(5)}(x_j)B_j(x) \\ & + u(1)B_N(x) + u'(1)B_{N+1}(x) + u''(1)B_{N+2}(x), \quad x \in [-1,1]. \end{aligned}$$

We find that

$$\begin{aligned} B_{-1}(x) &= -\frac{(x-1)^3(x+1)}{8}, & B_0(x) &= -\frac{(x-1)^3(3x+5)}{16}, \\ B_N(x) &= \frac{(x+1)^2(3x^2-10x+11)}{16}, & B_{N+1}(x) &= -\frac{(x-1)(x+1)^2(x-2)}{4}, \\ B_{N+2}(x) &= \frac{(x^2-1)^2}{8}, \end{aligned}$$

and the "interior" basis elements $\{B_j\}_{j=1}^{N-1} \subseteq \mathbb{P}_{N+3}$ satisfy

$$B_j(\pm 1) = 0, \quad B_j'(\pm 1) = 0, \quad B_j''(1) = 0, \quad B_j^{(5)}(x_i) = \delta_{ij},\ 0 < i < N. \tag{3.40}$$

The following two propositions describe the formulas for (3.40) at LGL and CGL points. Their computation is very stable.

**Proposition 3.5** (**Birkhoff interpolation at CGL points**). *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in the above basis at CGL points* (2.18) *are given by*

$$\begin{aligned} B_j(x) = \sum_{k=0}^{N-2} \beta_{kj} \big( & \partial_x^{-5} T_k(x) - [\partial_x^{-5} T_k(1)] B_N(x) \\ & - [\partial_x^{-4} T_k(1)] B_{N+1}(x) - [\partial_x^{-3} T_k(1)] B_{N+2}(x) \big), \end{aligned} \tag{3.41}$$

*where* $\partial_x^{-m} T_k(x) = \partial_x^{-1}[\partial_x^{1-m} T_k(x)]$ *as given by* (2.41) *and* (2.45), *and* $\beta_{kj}$ *given in* (2.55).

**Proof**: (3.41) gives

$$B_j^{(5)}(x_i) = \sum_{k=0}^{N-2} \beta_{kj} T_k(x_i) = \delta_{ij},$$

and $B_j(\pm 1) = B_j'(\pm 1) = B_j''(1) = 0$ from

$$\begin{aligned} F_k(x) = \partial_x^{-5} T_k(x) - [\partial_x^{-5} T_k(1)] B_N(x) - [\partial_x^{-4} T_k(1)] B_{N+1}(x) \\ - [\partial_x^{-3} T_k(1)] B_{N+2}(x), \end{aligned}$$

and $F_k(\pm 1) = F_k'(\pm 1) = F_k''(1) = 0$. ∎

**Proposition 3.6** (**Birkhoff interpolation at LGL points**). *The Birkhoff interpolation basis polynomials* $\{B_j\}_{j=1}^{N-1}$ *in the above basis at LGL points, i.e. roots of* $(1-x^2)P_N'(x)$, *are given by*

$$\begin{aligned} B_j(x) = \sum_{k=0}^{N-2} \frac{\beta_{kj}}{\gamma_k} \big( & \partial_x^{-5} P_k(x) - [\partial_x^{-5} P_k(1)] B_N(x) \\ & - [\partial_x^{-4} P_k(1)] B_{N+1}(x) - [\partial_x^{-3} P_k(1)] B_{N+2}(x) \big), \end{aligned} \tag{3.42}$$

*where*

$$\gamma_k = 2/(2k+1), \quad \partial_x^{-m} P_k(x) = \partial_x^{-1}[\partial_x^{1-m} P_k(x)],\ 3 \le m \le 5,$$

*as given by* (2.41) *and* (2.43)*, and* $\beta_{kj}$ *given in* (2.47)*.*

**Proof**: (3.42) gives

$$B_j^{(5)}(x_i) = \sum_{k=0}^{N-2} \frac{\beta_{kj}}{\gamma_k} P_k(x_i) = \delta_{ij},$$

and $B_j(\pm 1) = 0$, $B_j'(\pm 1) = B_j''(1) = 0$ from

$$\begin{aligned} F_k(x) = \partial_x^{-5} P_k(x) - [\partial_x^{-5} P_k(1)] B_N(x) - [\partial_x^{-4} P_k(1)] B_{N+1}(x) \\ - [\partial_x^{-3} P_k(1)] B_{N+2}(x), \end{aligned}$$

and $F_k(\pm 1) = F_k'(\pm 1) = F_k''(1) = 0$. ■

We test the new method (BCOL) on the problem (3.37) and compare results with two other collocation schemes (LCOL and SCOL), which we detail shortly, with a convergence comparison in Fig. 3.3 below.

### (i) BCOL scheme

The Birkhoff collocation method (BCOL) offers the numerical solution

$$u_N(x) = \sum_{j=1}^{N-1} v_j B_j(x),$$

where $v_j = u_N^{(5)}(x_j)$. Then (3.38) gives the system, solving for $\vec{v} = (v_1, \ldots, v_{N-1})^t$,

$$v_i + a(x_i) \sum_{j=1}^{N-1} v_j B_j'(x_i) + b(x_i) \sum_{j=1}^{N-1} v_j B_j(x_i) = f(x_i), \quad 0 < i < N. \tag{3.43}$$

With the bases (3.42) for LGL and (3.41) for CGL points, we can compute the pseudospectral integration matrices

$$\boldsymbol{B}_{\text{in}}^{(k)} = [B_j^{(k)}(x_i)]_{0<i,j<N}, \quad k = 0, 1,$$

with $\boldsymbol{B}_{\text{in}} = \boldsymbol{B}_{\text{in}}^{(0)}$. Thus, (3.43) in matrix form is

$$(-\boldsymbol{I}_{N-1} + \boldsymbol{\Lambda}_a \boldsymbol{B}_{\text{in}}^{(1)} + \boldsymbol{\Lambda}_b \boldsymbol{B}_{\text{in}})\vec{v} = \vec{f}. \tag{3.44}$$

where $\boldsymbol{I}_{N-1}$ is the $(N-1) \times (N-1)$ identity matrix, and

$$\begin{aligned} &\boldsymbol{\Lambda}_a = \text{diag}(a(x_1), \ldots, a(x_{N-1})), \quad && \boldsymbol{\Lambda}_b = \text{diag}(b(x_1), \ldots, b(x_{N-1})), \\ &\vec{f} = (f(x_1), \ldots, f(x_{N-1}))^t, && \vec{v} = (u_N'''(x_1), \ldots, u_N'''(x_{N-1}))^t. \end{aligned}$$

Then we recover the approximation to $u$ by

$$\vec{u} = (u_N(-1), \ldots, u_N(1))^t = \boldsymbol{B}_{\text{in}}\vec{v}. \tag{3.45}$$

**(ii) LCOL scheme**

The usual Lagrange collocation method (LCOL) offers the numerical solution

$$u_N(x) = \sum_{j=1}^{N-1} u_N(x_j) L_j(x) \in \mathbb{P}_N,$$

where

$$L_j(x) = \ell_j(x) \frac{(1-x^2)}{(1-x_j^2)},$$

with $\ell_j(x)$ defined as in (2.4), ($q_N$ given by Rem. 2.13). Then (3.38) gives the system, solving for $\vec{u} = (u_N(x_1), \ldots, u_N(x_{N-1}))^t$,

$$\begin{aligned} &\sum_{j=1}^{N-1} u_N(x_j) L_j'(\pm 1) = 0; \qquad \sum_{j=1}^{N-1} u_N(x_j) L_j''(1) = 0; \\ &\sum_{j=1}^{N-1} u_N(x_j) L_j^{(5)}(x_i) + a(x_i) \sum_{j=1}^{N-1} u_N(x_j) L_j'(x_i) + b(x_i) u_N(x_i) = f(x_i), \end{aligned}$$

for $0 < i < N$.

### (iii) SCOL scheme

The special collocation method (SCOL), as in [101, p. 218], is based on the (generalized Lagrange) interpolation problem:

$$\begin{cases} \text{Find } p \in \mathbb{P}_{N+3} \text{ such that} \\ p(y_j) = u(y_j),\ 0 < j < N; \quad p^{(k)}(\pm 1) = u^{(k)}(\pm 1),\ k = 0, 1; \\ p''(1) = u''(1), \end{cases} \tag{3.46}$$

where $\{y_j\}_{j=1}^{N-1}$ are zeros of the Jacobi polynomial $P_{N-1}^{3,2}(x)$, as suggested in [72]. It offers the numerical solution

$$u_N(x) = \sum_{j=1}^{N-1} s_j \tilde{L}_j(x),$$

where

$$\tilde{L}_j(x) = \frac{P_{N-1}^{3,2}(x)}{\partial_x P_{N-1}^{3,2}(x_j)(x - x_j)} \frac{(1-x)^3(1+x)^2}{(1-x_j)^3(1+x_j)^2}.$$

Then (3.38) gives the system, solving for $\vec{s} = (s_1, \ldots, s_{N-1})^t$,

$$\sum_{j=1}^{N-1} s_j \tilde{L}_j^{(5)}(y_i) + a(x_i) \sum_{j=1}^{N-1} s_j \tilde{L}_j'(y_i) + b(x_i) s_i = f(y_i), \quad 0 < i < N.$$

The collocation values are determined by

$$u_N(x_i) = \sum_{j=1}^{N-1} s_j \tilde{L}_j(x_i), \quad 0 < i < N.$$

Here we compare LCOL and BCOL at CGL points with SCOL, solving (3.38) with

$$a(x) = \sin(10x), \quad b(x) = x$$

and exact solution $u(x) = \sin^3(\pi x)$. We plot in Fig. 3.3 convergence behavior of three methods, which clearly indicates the new approach is well-conditioned and significantly superior to the other two.

The solver (3.44)–(3.45) will be used for solving the fifth-order KdV equation in the next subsection.

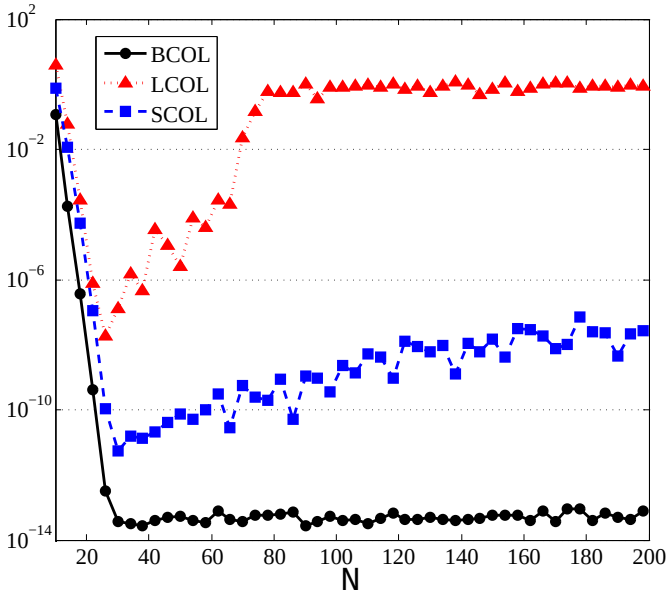



Figure 3.3: Comparison of three collocation schemes.

### 3.2.3 Third- and fifth-order KdV equations

We apply the well-conditioned collocation method in Subsec. 3.2.1 to solve the third-order KdV equation:

$$\partial_t u + u\partial_x u + \partial_x^3 u = 0, \quad x \in (-\infty, \infty),\ t > 0; \quad u(x,0) = u_0(x), \tag{3.47}$$

with the exact soliton solution (cf. [79])

$$u(x,t) = 12\kappa^2 \mathrm{sech}^2(\kappa(x - 4\kappa^2 t - x_0)), \tag{3.48}$$

where $\kappa$ and $x_0$ are constants.

Since the solution decays exponentially, we can approximate the initial value problems by imposing homogeneous boundary conditions over $x \in (-L, L)$ as long as the soliton wave does not reach the boundaries.

Let $\tau$ be the time step size, and $\{\xi_j = Lx_j\}_{j=0}^N$ with $\{x_j\}_{j=0}^N$ being CGL points. Then we adopt the Crank-Nicolson leap-frog scheme in time and the new

collocation method in space, that is, find $u_N^{k+1} \in \mathbb{P}_{N+1}$ such that for $0 < j < N$,

$$\begin{aligned} &\frac{u_N^{k+1}(\xi_j) - u_N^{k-1}(\xi_j)}{2\tau} + \partial_x^3 \left( \frac{u_N^{k+1} + u_N^{k-1}}{2} \right)(\xi_j) = -\partial_x u_N^k(\xi_j) u_N^k(\xi_j), \\ &u_N^k(\pm L) = \partial_x u_N^k(L) = 0, \quad k \geq 0. \end{aligned} \tag{3.49}$$

In the computation, we take $\kappa = 0.3$, $x_0 = -20$, $L = 50$ and $\tau = 0.001$. We depict in Fig. 3.4 (left) the numerical evolution of the solution (3.49) with $t \leq 50$ and $N = 160$. In Fig. 3.4 (right), we plot the maximum point-wise errors for various $N$ at $t = 1, 50$. We see the errors decay exponentially, and the scheme

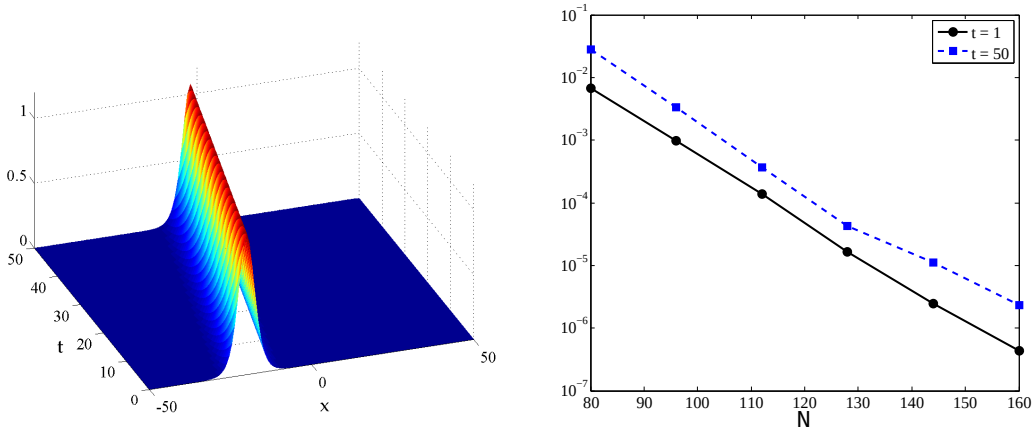



Figure 3.4: Left: time evolution of numerical solution (3.49) for $N = 160$. Right: maximum absolute error at interior collocation points at given $t$ for given $N$.

is stable. Indeed, the proposed collocation method produces a very accurate and stable solution, like the well-conditioned dual-Petrov-Galerkin method in [100].

We also apply the new method in space from Subsec. 3.2.2 to solve the fifth-order KdV equation:

$$\partial_t u + \gamma u \partial_x u + \nu \partial_x^3 u - \mu \partial_x^5 u = 0, \quad x \in (-\infty, \infty),\ t > 0; \quad u(x, 0) = u_0(x). \tag{3.50}$$

For $\gamma \neq 0$, and $\mu\nu > 0$, it has the exact soliton solution (cf. [90]):

$$u(x,t) = \eta_0 + \frac{105\nu^2}{169\mu\gamma} \operatorname{sech}^4 \left( \sqrt{\frac{\nu}{52\mu}} \left[ x - \left( \gamma\eta_0 + \frac{36\nu^2}{169\mu} \right) t - x_0 \right] \right), \tag{3.51}$$

where $\eta_0$ and $x_0$ are any constants.

We can approximate the IVPs by imposing homogeneous boundary conditions over $x \in (-L, L)$ as long as the soliton wave does not reach the boundaries.

Let $\tau$ be the time step size, and $\{\xi_j = Lx_j\}_{j=0}^N$ with $\{x_j\}_{j=0}^N$ being CGL points. We adopt the Crank-Nicolson leap-frog scheme in time and the new collocation method in space, that is, find $u_N^{k+1} \in \mathbb{P}_{N+3}$ such that for $0 < j < N$,

$$\begin{aligned} &\frac{u_N^{k+1}(\xi_j) - u_N^{k-1}(\xi_j)}{2\tau} + \nu\partial_x^3\left(\frac{u_N^{k+1} + u_N^{k-1}}{2}\right)(\xi_j) - \mu\partial_x^5\left(\frac{u_N^{k+1} + u_N^{k-1}}{2}\right)(\xi_j) \\ &= -\gamma\partial_x u_N^k(\xi_j)u_N^k(\xi_j), \quad u_N^k(\pm L) = \partial_x u_N^k(\pm L) = \partial_x^2 u_N^k(L) = 0, \quad k \geq 0. \end{aligned} \tag{3.52}$$

In Fig. 3.5, we depict the maximum pointwise errors at CGL points for (3.50)–(3.51) with $\mu = \gamma = 1$, $\nu = 1.1$, $\eta_0 = 0$, $x_0 = -10$, $L = 50$ and $\tau = 0.001$. It

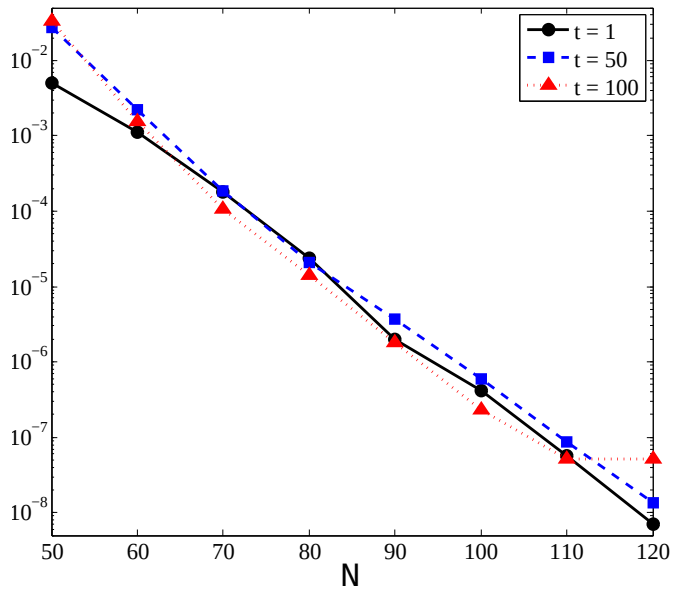



Figure 3.5: Maximum pointwise errors of the Crank-Nicolson-leap-frog and BCOL for fifth-order KdV equation.

indicates that the scheme is stable and accurate, which is comparable to the well-conditioned dual-Petrov-Galerkin scheme (cf. [101, Ch. 6]).

## 3.3 Birkhoff basis at Gegenbauer-Gauss-Lobatto points

For completeness, we provide the Birkhoff interpolation basis (2.31)–(2.32) on GGL points. We refer to e.g., [106, 55, 101] for details below.

**Proposition 3.7 (Birkhoff interpolation at GGL points).** *Let $G_n(x)$ be the normalized Gegenbauer polynomial of degree $n$ with parameter $\alpha$, i.e.,*

$$G_0(x) \equiv 1, \quad G_1(x) = x,$$
$$(2n+2\alpha+1)xG_n(x) = (n+2\alpha+1)G_{n+1}(x) + nG_{n-1}(x).$$

*Let $\{x_j, \omega_j\}_{j=0}^N$ be the GGL points, i.e. zeros of $(1-x^2)G_N'(x)$, with corresponding quadrature weights*

$$\omega_0 = \omega_N = \frac{2^{2\alpha+1}(\alpha+1)\Gamma^2(\alpha+1)\Gamma(N)}{\Gamma(N+2\alpha+2)}, \quad \omega_j = \frac{\omega_0}{(\alpha+1)[G_N(x_j)]^2}.$$

*Then the Birkhoff interpolation basis polynomials $\{B_j\}_{j=1}^{N-1}$ in Thm. 2.1 can be computed by*

$$B_j(x) = \sum_{k=0}^{N-2} \beta_{kj} \left( \partial_x^{(-2)} G_k(x) - \frac{\partial_x^{(-2)} G_k(1)}{2}(1+x) - \frac{\partial_x^{(-2)} G_k(-1)}{2}(1-x) \right),$$
$$\beta_{kj} = \left( G_k(x_j) - \frac{1-(-1)^{N+k}}{2} G_{N-1}(x_j) - \frac{1+(-1)^{N+k}}{2} G_N(x_j) \right) \frac{\omega_j}{\gamma_k},$$

*where*

$$\begin{aligned} \partial_x^{(-2)} G_k(x) = {} & \frac{(k+2\alpha+1)(k+2\alpha+2)}{(k+1)(k+2)(2k+2\alpha+1)(2k+2\alpha+3)} G_{k+2}(x) \\ & - \frac{2}{(2k+\alpha-1)(2k+2\alpha+3)} G_k(x) \\ & + \frac{k(k-1)}{(k+2\alpha)(k+2\alpha-1)[4(k+\alpha)^2-1]} G_{k-2}(x); \end{aligned} \tag{3.53}$$
$$\partial_x^{(-2)} G_0(x) = \frac{\alpha+1}{2\alpha+3} G_2(x);$$
$$\partial_x^{(-2)} G_1(x) = \frac{(\alpha+1)(2\alpha+1)G_3(x) - 6x}{3(2\alpha+1)(2\alpha+5)}.$$

*and*

$$\gamma_k = \int_{-1}^{1} [G_k(x)]^2 (1-x^2)^\alpha \,\mathrm{d}x = \frac{2^{2\alpha+1}\Gamma^2(\alpha+1)n!}{(2n+2\alpha+1)\Gamma(n+2\alpha+1)}. \tag{3.54}$$

**Proof**: Since $B_j'' \in \mathbb{P}_{N-2}$, we expand it in terms of Gegenbauer polynomials:

$$B_j''(x) = \sum_{k=0}^{N-2} \beta_{kj} G_k(x), \tag{3.55}$$

where

$$\beta_{kj} = \frac{1}{\gamma_k} \int_{-1}^{1} B_j''(x) G_k(x) (1-x^2)^\alpha \,\mathrm{d}x,$$

and $\gamma_k$ is given in (3.54). For $0 < j < N$, using the orthogonality of Gegenbauer polynomials in $L^2_{(1-x^2)^\alpha}$, the exactness of GGL quadrature (see, e.g. [101]), $G_k(\pm 1) = (\pm 1)^k$ and (2.29) leads to

$$\beta_{kj} = \frac{[(-1)^k B_j''(-1) + B_j''(1)]\omega_0 + G_k(x_j)\omega_j}{\gamma_k}. \tag{3.56}$$

Notice that (3.56) is valid for all $k \leq N+1$. Taking $k = N-1, N$, we obtain from (3.54) that the resulted integrals vanish, so we have the linear system of $B_j''(\pm 1)$:

$$\begin{aligned} &[(-1)^{N-1} B_j''(-1) + B_j''(1)]\omega_0 + G_{N-1}(x_j)\omega_j = 0, \\ &[(-1)^N B_j''(-1) + B_j''(1)]\omega_0 + G_N(x_j)\omega_j = 0. \end{aligned}$$

Therefore, we solve it and find that

$$B_j''(\pm 1) = -(\pm 1)^N \frac{\omega_j}{2\omega_0} (G_N(x_j) \pm G_{N-1}(x_j)), \quad 0 < j < N. \tag{3.57}$$

Inserting (3.57) into (3.56) yields the expression for $\beta_{kj}$ in Prop. 3.7.

Next, by

$$\begin{aligned} G_k(x) = &\frac{k+2\alpha+1}{(k+1)(2k+2\alpha+1)} G_{k+1}'(x) \\ &- \frac{k}{(k+2\alpha)(2k+2\alpha+1)} G_{k-1}'(x), \end{aligned} \tag{3.58}$$

we can define

$$\begin{aligned}\partial_x^{(-1)}G_k(x) &= \frac{(k+2\alpha)(k+2\alpha+1)G_{k+1}(x) - k(k+1)G_{k-1}(x)}{(k+1)(k+2\alpha)(2k+2\alpha+1)};\\ \partial_x^{(-1)}G_0(x) &= x = G_1(x).\end{aligned} \tag{3.59}$$

such that $\partial_x[\partial_x^{(-1)}G_k(x)] = G_k(x)$, and using (3.59) recursively yields, in (3.53), $\partial_x^2[\partial_x^{(-2)}G_k(x)] = G_k(x)$. Thus, it follows from (3.55) that

$$B_j(x) = \sum_{k=0}^{N-2} \beta_{kj}[\partial_x^{(-2)}G_k(x) + a_{kj}x + b_{kj}], \tag{3.60}$$

where $a_{kj}$ and $b_{kj}$ are constants to be determined by $B_j(\pm 1) = 0$. Verifying

$$\partial_x^{(-2)}G_k(\pm 1) - (\partial_x^{(-2)}G_k(1))B_N(\pm 1) - (\partial_x^{(-2)}G_k(-1))B_0(\pm 1) = 0,$$

the formula for $B_j(x)$ is confirmed. ■

**Remark 3.3.** *If $\alpha = 0$, $G_k = P_k$ and Prop. 3.7 gives the same basis as Prop. 2.1. If $\alpha = -1/2$, $G_k = T_k$ and Prop. 3.7 gives the same basis as Prop. 2.2.*

With this new basis, we can solve

$$u''(x) - u(x) = \frac{1+x}{2}, \quad x \in (-1,1); \quad u(\pm 1) = 0, \tag{3.61}$$

with exact solution

$$u(x) = \frac{\sinh(1+x)}{\sinh 2} - \frac{1+x}{2}. \tag{3.62}$$

The collocation scheme at GGL points for (3.61) is to find $u_N \in \mathbb{P}_N$ such that

$$u_N''(x_i) - u_N(x_i) = \frac{1+x_i}{2}, \quad 0 < i < N; \quad u_N(\pm 1) = 0. \tag{3.63}$$

We compare the new collocation scheme with the Lagrange collocation scheme.

For the usual Lagrange collocation method (LCOL), we can determine the solution by (3.63),

$$u_N(x) = \sum_{i=1}^{N-1} u_N(x_i)L_i(x),$$

to get the matrix equation

$$(\boldsymbol{D}_{\text{in}}^{(2)} - \boldsymbol{I}_{N-1})\vec{u} = \hat{f}, \tag{3.64}$$

where

$$\vec{u} = (u_N(x_1), \ldots, u_N(x_{N-1}))^t, \quad \hat{f} = (f(x_1), \ldots, f(x_{N-1}))^t.$$

For the Birkhoff collocation method (BCOL), we can determine the solution by (3.63),

$$u_N(x) = \sum_{i=1}^{N-1} u_N''(x_i) B_i(x),$$

to get the matrix equation

$$(\boldsymbol{I}_{N-1} - \boldsymbol{B}_{\text{in}})\vec{v} = \vec{f}, \tag{3.65}$$

where $\boldsymbol{B}_{\text{in}} = [B_j(x_i)]$ for $0 < i, j < N$, and

$$\vec{v} = (u_N''(x_1), \ldots, u_N''(x_{N-1}))^t, \quad \vec{f} = (f(x_1), \ldots, f(x_{N-1}))^t.$$

In Table 3.3, we tabulate the condition numbers of the coefficient matrices for different values of $\alpha$. As before, LCOL shows condition number growth like $O(N^4)$, while BCOL has condition number bounded by a constant.

Table 3.3: Comparison of condition numbers of coefficient matrices of LCOL (3.64) and BCOL (3.65), various $\alpha$.

| | $\alpha = -1/4$ | | $\alpha = 1/4$ | | $\alpha = 1/2$ | |
|---|---|---|---|---|---|---|
| $N$ | LCOL | BCOL | LCOL | BCOL | LCOL | BCOL |
| 64 | 1.73e+05 | 1.41 | 9.76e+04 | 1.41 | 7.53e+04 | 1.41 |
| 128 | 2.74e+06 | 1.41 | 1.52e+06 | 1.41 | 1.17e+06 | 1.41 |
| 256 | 4.36e+07 | 1.41 | 2.41e+07 | 1.41 | 1.84e+07 | 1.41 |
| 512 | 6.97e+08 | 1.41 | 3.83e+08 | 1.41 | 2.92e+08 | 1.41 |
| 1024 | 1.11e+10 | 1.41 | 6.11e+09 | 1.41 | 4.65e+09 | 1.41 |

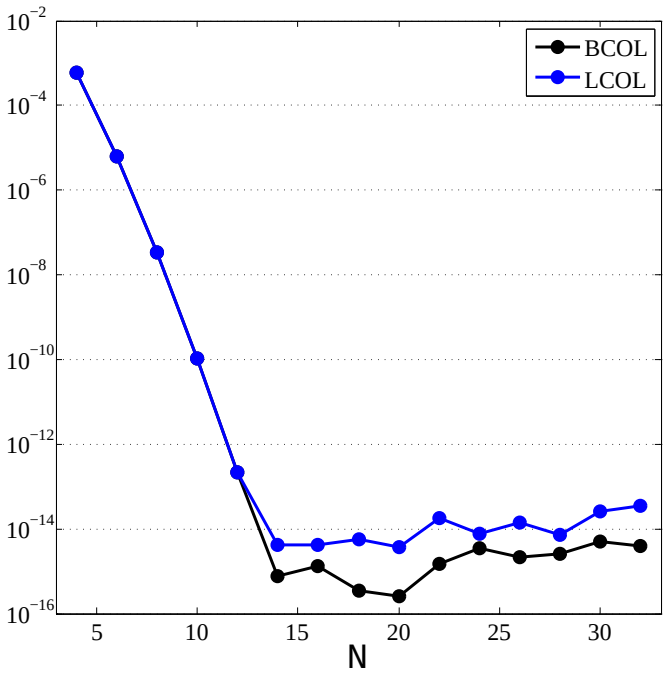



Figure 3.6: Comparison of maximum pointwise errors on GGL points, $\alpha = 2$.

In Fig. 3.6, we graph the maximum point-wise errors for both BCOL and LCOL on GGL points, with $\alpha = 2$.

Similar computations can be made for other boundary conditions. Extending these results to solutions of (3.28) or (3.37) for GGL points follows easily. Extending to (3.15) for GGR points, or any of these problems on even more general Jacobi-Gauss-type points follows from determining the formulas for orthogonality, anti-differentiation and quadrature (see e.g., [101]).

## 3.4 Multiple dimensions

In this section, we demonstrate the application of matrix decomposition (or diagonalization) technique (see [86]), which is also used with the spectral-Galerkin method [99, 102], to construct a tensorial matrix operator for partial differentiation.

We consider, as an example, the two-dimensional BVP:

$$\Delta u - \gamma u = f \quad \text{in } \Omega = (-1,1)^2; \quad u = 0 \quad \text{on } \partial\Omega, \tag{3.66}$$

where $\gamma \geq 0$ and $f \in C(\Omega)$. The collocation scheme is, on tensorial LGL points: find $u_N(x,y) \in \mathbb{Q}_N(\Omega) := \mathbb{P}_N^2$ such that

$$(\Delta u_N - \gamma u_N)(x_i, y_j) = f(x_i, y_j), \quad 0 < i, j < N; \quad u_N = 0 \quad \text{on } \partial\Omega, \tag{3.67}$$

where $\{x_i\}$ and $\{y_j\}$ are LGL points.

We illustrate the idea of matrix decomposition by using partial diagonalization (see [101, Sec. 8.1] and [12]). Write

$$u_N(x,y) = \sum_{k,l=1}^{N-1} u_{kl} B_k(x) B_l(y),$$

and obtain from (3.67) the system:

$$\boldsymbol{U}\boldsymbol{B}_{\text{in}}^t + \boldsymbol{B}_{\text{in}}\boldsymbol{U} - \gamma \boldsymbol{B}_{\text{in}}\boldsymbol{U}\boldsymbol{B}_{\text{in}}^t = \boldsymbol{F}, \tag{3.68}$$

where $\boldsymbol{U} = [u_{kl}]_{0<k,l<N}$ and $\boldsymbol{F} = [f_{kl}]_{0<k,l<N}$. We consider the generalized eigenproblem:

$$\boldsymbol{B}_{\text{in}}\vec{x} = \lambda(\boldsymbol{I}_{N-1} - \gamma \boldsymbol{B}_{\text{in}})\vec{x}. \tag{3.69}$$

We know from Prop. 2.3 and Rem. 2.11 that the eigenvalues are distinct. Let $\boldsymbol{\Lambda}$ be the diagonal matrix of the eigenvalues, and $\boldsymbol{E}$ be the matrix whose columns are the corresponding eigenvectors. Then we have

$$\boldsymbol{B}_{\text{in}}\boldsymbol{E} = (\boldsymbol{I}_{N-1} - \gamma \boldsymbol{B}_{\text{in}})\boldsymbol{E}\boldsymbol{\Lambda}.$$

We describe the partial diagonalization (see [101, Sec. 8.1]). Set $\boldsymbol{U} = \boldsymbol{E}\boldsymbol{V}$. Then (3.68) becomes

$$\boldsymbol{V}\boldsymbol{B}_{\text{in}}^t + \boldsymbol{\Lambda}\boldsymbol{V} = \boldsymbol{G} := \boldsymbol{E}^{-1}(\boldsymbol{I}_{N-1} - \gamma \boldsymbol{B}_{\text{in}})^{-1}\boldsymbol{F}. \tag{3.70}$$

Taking transpose of the above equation leads to

$$\boldsymbol{B}_{\text{in}}\boldsymbol{V}^t + \boldsymbol{V}^t\boldsymbol{\Lambda} = \boldsymbol{G}^t. \tag{3.71}$$

Let $\vec{v}_p$ be the transpose of $p$th row of $\boldsymbol{V}$, and likewise for $\vec{g}_p$. Then we solve the systems:

$$(\boldsymbol{B}_{\rm in} + \lambda_p \boldsymbol{I}_{N-1})\vec{v}_p = \vec{g}_p, \quad p = 1, 2, \ldots, N-1. \tag{3.72}$$

As shown in Sec. 2.2, the coefficient matrix of (3.72) is well-conditioned. Note that this process can be extended to three dimensions straightforwardly.

**Remark 3.4.** *It is seen that the extension to multiple dimensions essentially relies on solving the generalized eigen-problem* (3.69)*. We remark that* $\boldsymbol{B}_{\rm in}$ *has the same condition number as* $\boldsymbol{D}^{(2)}_{\rm in}$*. Nevertheless, this is common for tensor-based approaches including the spectral-Galerkin method* (see e.g., [101, 12]) *and other aforementioned integration preconditioning techniques* (see e.g., [62, 35, 36, 70, 47])*.*

As a numerical illustration, we consider (3.66) with $\gamma = 0$ and the exact solution,

$$u(x,y) = \begin{cases} (\sinh(x+1) - x - 1)\cos(\pi y/2)e^{xy}, & x < 0, \\ (\sinh(x+1) - \sinh(x) - 1 - (\sinh(2) - \sinh(1) - 1)x^3) & \\ \times \cos(\pi y/2)e^{xy}, & 0 \le x, \end{cases}$$

which is first-order differentiable in $x$, while smooth in $y$. We fix $N_y = 16$ (with negligible errors in $y$, and requiring to solve a generalized eigen-problem), and examine the accuracy for different $N_x$.

In Fig. 3.7, we plot the maximum pointwise errors against various $N_x$ of the new approach for more than one thousand points. We see that the new approach is stable with an expected rate of convergence. Moreover, it is comparable to the spectral-Galerkin method (SGAL) in [99], as seen from Fig. 3.7 (left), where the errors of two methods are indistinguishable.

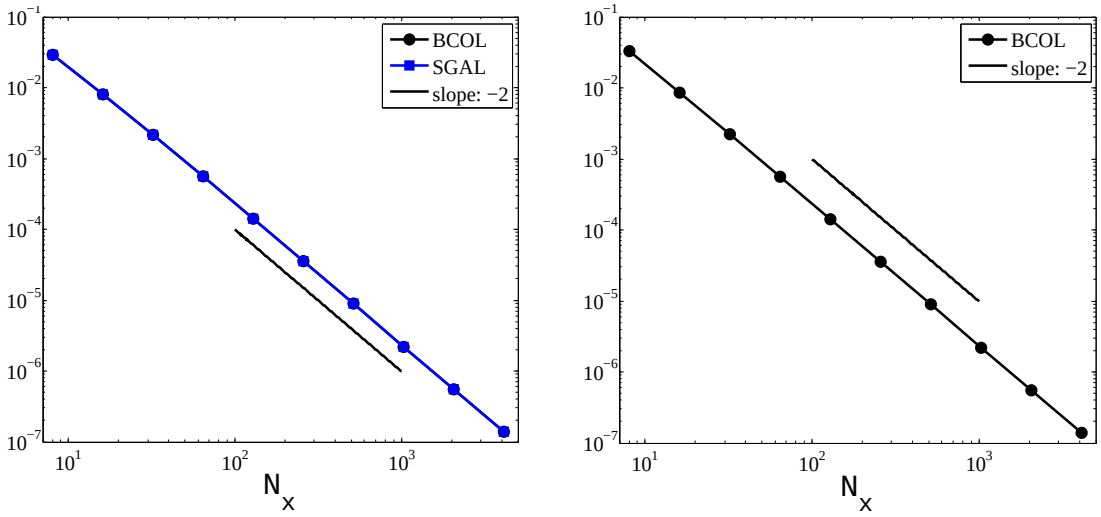



Figure 3.7: Maximum pointwise errors. Left: LGL; right: CGL.

## 3.5 Well-conditioned collocation methods on the half-line

Many problems are set in unbounded domains: we refer to [101, Ch. 7] for an overview of spectral methods on unbounded domains. The approach based on collocation on classical orthogonal systems on unbounded domains, e.g. Laguerre or Hermite polynomials or functions, is less addressed than the spectral-Galerkin method. In particular, the second-order differentiation matrix associated with Laguerre functions is ill-conditioned and its minimum eigenvalues shrink as $O(N^{-2})$ (see Table 3.4)—following the analysis in Prop. 2.3, this indicates that the direct application of the collocation method based on the natural Birkhoff interpolation will not result in a well-conditioned scheme.

We propose an alternative Birkhoff-type interpolation, making use of a different differentiation operator (3.81), to produce, with the method of Ch. 2, a well-conditioned collocation scheme.

To illustrate this modified method, in this section, we consider the following

half-line problem:

$$\begin{cases} -u''(x) + a(x)u'(x) + b(x)u(x) = f(x), \quad x \in (0, \infty), \\ u(0) = u_0, \quad \lim\limits_{x\to\infty} u(x) = 0, \end{cases} \tag{3.73}$$

where $a$, $b$ and $f$ are given continuous functions on the half-line, and $u_0$ is a given constant.

Let $\mathcal{L}_k(x)$ be the Laguerre polynomial of degree $k$ and $\widehat{\mathcal{L}}_k(x) = \mathcal{L}_k(x)e^{-x/2}$ be the Laguerre function of degree $k$. Laguerre polynomials are mutually weighted-orthonormal:

$$\int_0^\infty \mathcal{L}_m(x)\mathcal{L}_n(x)e^{-x}\,\mathrm{d}x = \delta_{mn}, \tag{3.74}$$

The Laguerre functions $\{\widehat{\mathcal{L}}_k(x)\}_{k=0}^N$ form a basis for the function space

$$\hat{\mathbb{P}}_N = \{p(x)\exp(-x/2), \quad p(x) \in \mathbb{P}_N\},$$

and they are mutually orthonormal:

$$\int_0^\infty \widehat{\mathcal{L}}_m(x)\widehat{\mathcal{L}}_n(x)\,\mathrm{d}x = \delta_{mn}. \tag{3.75}$$

The Laguerre-Gauss-Radau points $\{x_i\}_{i=0}^N$ are roots of $x\mathcal{L}'_{N+1}(x) = 0$, and the quadrature weights $\omega_j$ for Laguerre polynomials and $\widehat{\omega}_j$ for Laguerre functions are, for $0 \le j \le N$,

$$\omega_j = \frac{1}{(N+1)[\mathcal{L}_N(x_j)]^2}, \quad \widehat{\omega}_j = \frac{1}{(N+1)[\widehat{\mathcal{L}}_N(x_j)]^2} = \omega_j e^{x_j}. \tag{3.76}$$

Then the following Laguerre-Gauss-Radau quadratures are exact:

$$\int_0^\infty p(x)e^{-x}\,\mathrm{d}x = \sum_{j=0}^N p(x_j)\omega_j, \qquad \text{for all } \ p \in \mathbb{P}_{2N}, \tag{3.77}$$

$$\int_0^\infty \hat{p}(x)\,\mathrm{d}x = \sum_{j=0}^N \hat{p}(x_j)\widehat{\omega}_j, \qquad \text{for all } \ \hat{p}(x)e^x \in \mathbb{P}_{2N}. \tag{3.78}$$

**Remark 3.5.** *It must be noted here, as in* [101, Ch. 7]*, that $x_N$ grows like $O(N)$, while $x_1$ shrinks like $O(N^{-1})$. Thus, evaluating $\mathcal{L}_N(x_i)$ can cause resolution error, as $\mathcal{L}_N(x_1)$ is bounded by a constant, but $\mathcal{L}_N(x_N)$ grows as $O(N^N)$.*

Thus, the collocation scheme at $\{x_i\}_{i=0}^N$ for (3.73) is to find $u_N \in \hat{\mathbb{P}}_N$ such that for $1 \le i \le N$,

$$-u_N''(x_i) + a(x_i)u_N'(x_i) + b(x_i)u_N(x_i) = f(x_i); \quad u_N(0) = u_0. \tag{3.79}$$

### 3.5.1 PSIM on Laguerre-Gauss-Radau points

We look at the Lagrange interpolation basis on Laguerre-Gauss-Radau points associated with Laguerre functions, as properties of the PSDM will guide our selection of the Birkhoff interpolation problem from which we derive the new basis.

Define the Lagrange interpolation polynomial basis $\{L_i(x)\}_{i=0}^N$, by (3.74) and (3.77), and the Lagrange interpolation function basis $\{\hat{L}_i(x)\}_{i=0}^N \subset \hat{\mathbb{P}}_N$, by (3.75) and (3.78), over the Laguerre-Gauss-Radau points $\{x_i\}_{i=0}^N$

$$L_j(x) = \omega_j \sum_{k=0}^{N} \mathcal{L}_k(x_j)\mathcal{L}_k(x), \quad \hat{L}_j(x) = L_j(x)\exp\left(\frac{x_j - x}{2}\right). \tag{3.80}$$

Denote the PSDMs for the Laguerre-Gauss-Radau points $\{x_i\}_{i=0}^N$ by

$$\widehat{\boldsymbol{D}} := [\hat{L}_j'(x_i)]_{0\le i,j\le N}, \qquad \widehat{\boldsymbol{D}}^{(k)} := [\hat{L}_j^{(k)}(x_i)]_{0\le i,j\le N} = \widehat{\boldsymbol{D}}^k,$$
$$\widehat{\boldsymbol{D}}_{\text{in}}^{(k)} := [\hat{L}_j^{(k)}(x_i)]_{1\le i,j\le N}.$$

The eigenvalues of $\widehat{\boldsymbol{D}}_{\text{in}}^{(2)}$ are all real and negative. Table 3.4 shows that the largest eigenvalues grow like $O(N^2)$, and the smallest eigenvalues shrink like $O(N^{-2})$.

Thus, the condition numbers of both $\widehat{\boldsymbol{D}}_{\text{in}}^{(2)}$ and $[\widehat{\boldsymbol{D}}_{\text{in}}^{(2)}]^{-1}$ grow like $O(N^4)$.

If we let $\widetilde{\boldsymbol{D}}_{\text{in}}^{(2)} = \widehat{\boldsymbol{D}}_{\text{in}}^{(2)} - \mu\boldsymbol{I}_N$, the largest eigenvalue of $\widetilde{\boldsymbol{D}}_{\text{in}}^{(2)}$ grows like $O(N^2)$, but the smallest eigenvalue has an upper bound $-\mu$, the condition number of $[\widetilde{\boldsymbol{D}}_{\text{in}}^{(2)}]^{-1}$ will be independent of $N$. This is in contrast with the Legendre and Chebyshev cases (see Prop. 2.3 and Rem. 2.11), where the smallest eigenvalue in magnitude behaves like a constant. This is essential for the well-conditioned collocation schemes in Ch. 2–3.

Table 3.4: Condition number, maximum and minimum absolute value of eigenvalues of $\widehat{\boldsymbol{D}}_{\text{in}}^{(2)}$.

| $N$ | Cond. # | Max \|Eig\| | Min \|Eig\| |
|---|---|---|---|
| 32 | 2.08e+05 | 1.10e+02 | 5.67e−04 |
| 64 | 3.14e+06 | 4.28e+02 | 1.46e−04 |
| 128 | 4.87e+07 | 1.69e+03 | 3.71e−05 |
| 256 | 7.68e+08 | 6.69e+03 | 9.34e−06 |

The following details come from, e.g. [101, Ch. 7]: given the differential operator

$$\hat{\partial}_x = \partial_x + \frac{1}{2}, \tag{3.81}$$

if $\hat{p}(x) = p(x)e^{-x/2} \in \hat{\mathbb{P}}_N$, then $\hat{\partial}_x\hat{p}(x) = p'(x)e^{-x/2} \in \hat{\mathbb{P}}_{N-1}$. Also, we have,

$$\mathcal{L}_k(x) = \mathcal{L}_k'(x) - \mathcal{L}_{k+1}'(x), \quad \widehat{\mathcal{L}}_k(x) = \hat{\partial}_x\widehat{\mathcal{L}}_k(x) - \hat{\partial}_x\widehat{\mathcal{L}}_{k+1}(x),\ k \geq 0. \tag{3.82}$$

Thus, for $\mu = 1/4$,

$$\hat{p} \in \hat{\mathbb{P}}_N \quad \Rightarrow \quad \hat{p}'' - \frac{1}{4}\hat{p} = (\hat{\partial}_x^2 - \hat{\partial}_x)\hat{p} \in \hat{\mathbb{P}}_{N-1}. \tag{3.83}$$

With this in mind, we consider the Birkhoff-type interpolation problem:

$$\begin{cases} \text{Find } \hat{p} \in \hat{\mathbb{P}}_N \text{ such that for } u \in C^2(0,\infty), u \to 0 \text{ as } x \to \infty, \\ \hat{p}(0) = u(0); \\ (\hat{\partial}_x^2 - \hat{\partial}_x)\hat{p}(x_j) = \hat{p}''(x_j) - \frac{1}{4}\hat{p}(x_j) = u''(x_j) - \frac{1}{4}u(x_j),\ 1 \leq j \leq N. \end{cases} \tag{3.84}$$

The Birkhoff interpolation function $\hat{p}$ of $u$ in (3.84) can be uniquely determined:

$$\hat{p}(x) = u(0)\widehat{B}_0(x) + \sum_{j=1}^{N}\left(u''(x_j) - \frac{1}{4}u(x_j)\right)\widehat{B}_j(x), \quad x \geq 0,$$

if one can find $\{\widehat{B}_j\}_{j=0}^N \subseteq \hat{\mathbb{P}}_N$, such that

$$\begin{aligned} &\widehat{B}_0(0) = 1, \quad \widehat{B}_0''(x_i) - \frac{1}{4}\widehat{B}_0(x_i) = 0, \quad 1 \leq i \leq N; \\ &\widehat{B}_j(0) = 0, \quad \widehat{B}_j''(x_i) - \frac{1}{4}\widehat{B}_j(x_i) = \delta_{ij},\ 1 \leq i, j \leq N. \end{aligned} \tag{3.85}$$

The following proposition gives us the computation for the Birkhoff interpolation basis $\{\widehat{B}_j\}$ satisfying (3.85).

**Proposition 3.8.** *The Birkhoff interpolation basis functions* $\{\widehat{B}_j\}_{j=0}^N$ *defined in* (3.85) *for the Laguerre-Gauss-Radau points* $\{x_j\}_{j=0}^N$*, roots of* $x\mathcal{L}'_{N+1}(x)$*, are*

$$\begin{aligned}\widehat{B}_0(x) =& \ \exp\left(-\frac{x}{2}\right),\\ \widehat{B}_j(x) =& \ \widehat{\omega}_j \sum_{k=0}^{N-1}[\widehat{\mathcal{L}}_{k+1}(x_j) - \widehat{\mathcal{L}}_N(x_j)]\hat{\partial}_x^{-2}\widehat{\mathcal{L}}_k(x)\\ &- \widehat{\omega}_j[\widehat{\mathcal{L}}_0(x_j) - \widehat{\mathcal{L}}_N(x_j)]\hat{\partial}_x^{-1}\widehat{\mathcal{L}}_0(x),\end{aligned}$$

*for* $1 \le j \le N$*, where*

$$\hat{\partial}_x^{-1}\widehat{\mathcal{L}}_k(x) = \widehat{\mathcal{L}}_k(x) - \widehat{\mathcal{L}}_{k+1}(x), \quad \hat{\partial}_x^{-2}\widehat{\mathcal{L}}_k(x) = \hat{\partial}_x^{-1}[\hat{\partial}_x^{-1}\widehat{\mathcal{L}}_m(x)].$$

**Proof**: (3.84) is related to the following Birkhoff-type *polynomial* interpolation problem:

$$\begin{cases}\text{Find } p \in \mathbb{P}_N \text{ such that for } u \in C^2(0,\infty),\\ p(0) = u(0); \quad p''(x_j) - p'(x_j) = u''(x_j) - u'(x_j),\ 1 \le j \le N.\end{cases} \tag{3.86}$$

The Birkhoff interpolation polynomial $p$ of $u$ in (3.86) can be uniquely determined by

$$p(x) = u(0)B_0(x) + \sum_{j=1}^{N}(u''(x_j) - u'(x_j))B_j(x), \quad x \ge 0,$$

if one can find $\{B_j\}_{j=0}^N \subseteq \mathbb{P}_N$, such that

$$\begin{aligned}&B_0(0) = 1, \quad B_0''(x_i) - B_0'(x_i) = 0, \quad 1 \le i \le N;\\ &B_j(0) = 0, \quad B_j''(x_i) - B_j'(x_i) = \delta_{ij},\ 1 \le i, j \le N.\end{aligned}$$

We have $B_0(x) \equiv 1$. Let $B_j''(x) = \sum_{k=0}^{N-2}\beta_{kj}\mathcal{L}_k(x)$, $1 \le j \le N$. We define, for $k \ge 0$,

$$\partial_x^{-1}\mathcal{L}_k(x) = \mathcal{L}_k(x) - \mathcal{L}_{k+1}(x), \quad \partial_x^{-2}\mathcal{L}_k(x) = \partial_x^{-1}[\partial_x^{-1}\mathcal{L}_m(x)].$$

Then (3.82) gives $\partial_x^m[\partial_x^{-m}\mathcal{L}_k(x)] = \mathcal{L}_k(x)$, $m = 1, 2$—note that $\partial_x^{-m}\mathcal{L}_k(0) = 0$. Let $B_j'(x) = \partial_x^{-1}B_j''(x) - \beta_{(-1)j}$, and

$$\begin{aligned} B_j''(x) - B_j'(x) &= \sum_{k=0}^{N-2} \beta_{kj}[\mathcal{L}_k(x) - (\mathcal{L}_k(x) - \mathcal{L}_{k+1}(x))] + \beta_{(-1)j} \\ &= \sum_{k=0}^{N-1} \beta_{(k-1)j}\mathcal{L}_k(x). \end{aligned}$$

By (3.74) and (3.77),

$$\begin{aligned} \int_0^\infty [B_j''(x) - B_j'(x)]\mathcal{L}_m(x)e^{-x}\,\mathrm{d}x &= \begin{cases} \beta_{(m-1)j}, & \text{if } m < N, \\ 0, & \text{if } m = N \end{cases} \\ &= [B_j''(0) - B_j'(0)]\omega_0 + \mathcal{L}_m(x_j)\omega_j. \end{aligned}$$

Setting $m = N$ gives

$$B_j''(0) - B_j'(0) = -\frac{\omega_j}{\omega_0}\mathcal{L}_N(x_j) = -\frac{1}{\mathcal{L}_N(x_j)}.$$

Thus, for $-1 \le k \le N-2$, $\beta_{kj} = \omega_j[\mathcal{L}_{k+1}(x_j) - \mathcal{L}_N(x_j)]$ and

$$B_j'(x) = \omega_j \sum_{k=0}^{N-2} (\mathcal{L}_{k+1}(x_j) - \mathcal{L}_N(x_j))\partial_x^{-1}\mathcal{L}_k(x) - \omega_j(\mathcal{L}_0(x_j) - \mathcal{L}_N(x_j)).$$

The proposition follows from $B_j(x) = \partial_x^{-1}B_j'(x)$, letting

$$\widehat{B}_j(x) = B_j(x)\exp\left(\frac{x_j - x}{2}\right), \quad 0 \le j \le N,$$

(3.82) giving $\hat{\partial}_x^m[\hat{\partial}_x^{-m}\widehat{\mathcal{L}}_k(x)] = \widehat{\mathcal{L}}_k(x)$, $m = 1, 2$, (3.75) and (3.78). ■

If $\widehat{\boldsymbol{B}}_{\text{in}} = [\widehat{B}_j(x_i)]$ for $1 \le i, j \le N$, then $\boldsymbol{I}_N = (\widehat{\boldsymbol{D}}_{\text{in}}^{(2)} - \frac{1}{4}\boldsymbol{I}_N)\widehat{\boldsymbol{B}}_{\text{in}}$. Thus, with the basis (3.85), we can construct the PSIM for a well-conditioned collocation scheme to solve (3.79).

### 3.5.2 Collocation schemes

Birkhoff interpolation based collocation (BCOL) indicates the solution

$$u_N(x) = u_0\widehat{B}_0(x) + \sum_{j=1}^{N} v_j\widehat{B}_j(x),$$

where

$$v_j = u_N''(x_j) - \frac{1}{4}u_N(x_j).$$

Then (3.79) gives, for $1 \le i \le N$,

$$\begin{aligned} &- v_i + a(x_i)\sum_{j=1}^{N} v_j \widehat{B}_j'(x_i) + \left(b(x_i) - \frac{1}{4}\right)\sum_{j=1}^{N} v_j \widehat{B}_j(x_i) \\ =&f(x_i) - u_0\left[a(x_i)\widehat{B}_0'(x_i) + \left(b(x_i) - \frac{1}{4}\right)\widehat{B}_0(x_i)\right]. \end{aligned} \tag{3.87}$$

First, to solve (3.73) with $a(x) \equiv 0$ and $b(x) \equiv \gamma$ by collocation on Laguerre-Gauss-Radau points $\{x_j\}_{j=0}^{N}$, we consider the condition numbers of the coefficient matrices in (3.87), and list them in Table 3.5, for various $\gamma$. It is readily apparent that the condition number is bounded by $4\gamma$.

Table 3.5: Comparison of condition numbers of coefficient matrix for (3.87), various $\gamma$.

| $N$ | $\gamma = 1$ | $\gamma = 2$ | $\gamma = 3$ |
|---|---|---|---|
| 16 | 3.88 | 7.52 | 10.93 |
| 32 | 3.97 | 7.87 | 11.70 |
| 64 | 3.99 | 7.97 | 11.92 |
| 128 | 4.00 | 7.99 | 11.98 |
| 256 | 4.00 | 8.00 | 11.99 |

Next, we compare BCOL with the standard Lagrange collocation method. Lagrange interpolation-based collocation (LCOL) indicates the solution

$$u_N(x) = u_0\hat{L}_0(x) + \sum_{j=1}^{N} u_N(x_j)\hat{L}_j(x).$$

Then (3.79) gives, for $1 \le i \le N$,

$$\begin{aligned} &- \sum_{j=1}^{N} u_N(x_j)\hat{L}_j''(x_i) + a(x_i)\sum_{j=1}^{N} u_N(x_j)\hat{L}_j'(x_i) + b(x_i)u_N(x_i) \\ =&f(x_i) + u_0[\hat{L}_0(x_i) - a(x_i)\hat{L}_0'(x_i)]. \end{aligned} \tag{3.88}$$

The Lagrange (LCOL) (3.88) and Birkhoff-like (BCOL) (3.87) collocation schemes are used to solve (3.73) with

$$a(x) = \sqrt{x}, \quad b(x) = \log(1+x),$$

where the exact solution is

$$u(x) = (1+x)^{-9/2}.$$

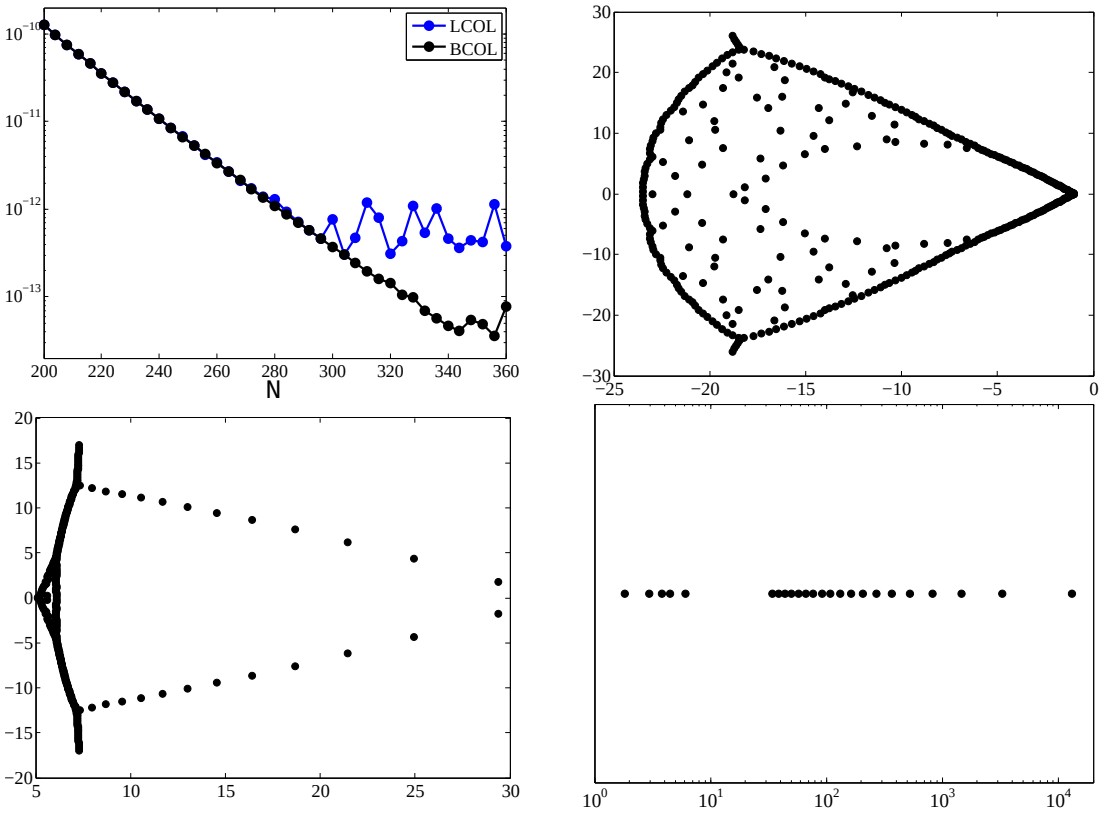



Figure 3.8: Upper left: accuracy graph comparing LCOL and BCOL; upper right: eigenvalues of BCOL, $N = 360$; lower row: eigenvalues of LCOL, $N = 360$, complex on left, real on right

The upper left graph of Fig. 3.8 shows $N$ versus the maximum absolute error of the computed solutions using LCOL and BCOL at the collocation points. The upper right graph of Fig. 3.8 is a scatter plot of the eigenvalues of the coefficient

matrix for BCOL at $N = 360$. The two lower graphs of Fig. 3.8 are scatter plots of the eigenvalues of the coefficient matrix for BCOL at $N = 360$, with complex eigenvalues on the left, and real eigenvalues on the right.

Thus, the new scheme is well-conditioned, and poses less run-off error than the LCOL scheme.

## 3.6 Summary

In this chapter, we extended the method of Ch. 2 to a number of problems different from second-order BVPs on Legendre- and Chebyshev-Gauss-Lobatto points. To summarize the results of the chapter:

- In Sec. 3.1, we demonstrated PSIM-based collocation schemes for IVPs are well-conditioned.

- In Sec. 3.2, we demonstrated that PSIM-based collocation schemes on BVPs of order higher than two are well-conditioned and can perform better than generalized Lagrange interpolation collocation schemes. We also showed that these collocation schemes could be used for space-discretization in simulations of KdV equations, with stability, efficiency and accuracy comparable to current methods.

- In Sec. 3.3, we demonstrated that the method in Ch. 2 can be applied to generate collocation schemes on Gegenbauer-Gauss-Lobatto points, with comparable results.

- In Sec. 3.4, we demonstrated a diagonalized PSIM-based collocation scheme in two-dimensions solving an elliptic PDE with Dirichlet boundary, and showed its spectral accuracy.

- In Sec. 3.5, we used eigenanalysis of the second-order PSDM on Laguerre-Gauss-Radau points associated with Laguerre functions to determine a more

suitable Birkhoff-type interpolation which, with the method of Ch. 2, produced a well-conditioned collocation scheme for the second-order IVP on the half-line. The eigenanalysis indicated that the Birkhoff interpolation analogous to those used in Ch. 2 would not produce a well-conditioned collocation scheme.

As shown in Sec. 3.5, applying the methodology of Ch. 2 requires proper eigenanalysis of the highest-order differentiation matrix to determine if a straightforward Birkhoff interpolation problem or if some alternative Birkhoff-type interpolation problem should be used to generate the interpolation basis that informs the PSIM, and will result in a well-conditioned collocation scheme.

Chapter 4

# A New TSEM: Implementation and Analysis on a Triangle

As indicated in Ch. 1, this chapter proposes a new triangular spectral-element method, based on the rectangle-triangle mapping (4.1)–(4.2) introduced in [82]. We emphasize two main features:

- The computational nodes on the reference triangle are the mapped tensorial LGL points on the reference square, which are much better distributed than the nodes mapped by Duffy's transform (4.4)–(4.5).
- There is no need to modify the tensorial basis functions on the reference element, as the singularity induced by the mapping can be removed.

On the first point, the mapping (4.1)–(4.2) improves on Duffy's transform by being one-to-one, mapping the singular point on the reference triangle to a vertex of the reference square, instead of an edge for the latter (see Fig. 4.1).

On the second point, the method detailed herein is an improvement over that in [82], which uses modified tensorial nodal functions on the reference square, incorporating the consistency condition (4.13).

The main difficulty lies in removing the logarithmic singularity produced by the mapping in the stiffness matrix analytically—this cannot be done under the

map (4.4)–(4.5). Details provided for generating the mass and stiffness matrices (cf. **Algorithm** on page 87) for implementing Galerkin formulation of elliptic problems demonstrate that their computation is stable, efficient and accurate.

The rest of the chapter is outlined as follows: A more detailed look on the map (4.1)–(4.2) is given in Sec. 4.1. Basis functions for the reference triangle and computations involving these for Galerkin methods are given in Sec. 4.2. Optimal error estimates for projection and interpolation are proven in Sec. 4.3. Numerical results are provided in Sec. 4.4.

## 4.1 The rectangle-triangle mapping

We collect in this section some properties of the rectangle-triangle mapping introduced in [82], and provide some insightful perspectives on this transform.

### 4.1.1 The map

Throughout the paper, we denote by

$$\triangle := \{(x,y) : 0 < x, y, x+y < 1\}, \quad \square := \{(\xi,\eta) : -1 < \xi, \eta < 1\},$$

the *reference triangle* and the *reference square*, respectively. The rectangle-triangle transform (cf. [82]) $T : \square \to \triangle$, takes the form

$$x = \frac{1}{8}(1+\xi)(3-\eta), \quad y = \frac{1}{8}(3-\xi)(1+\eta), \quad \forall\ (\xi,\eta) \in \square, \tag{4.1}$$

with the inversion $T^{-1} : \triangle \to \square$:

$$\xi = 1 + (x-y) - \chi, \quad \eta = 1 - (x-y) - \chi, \quad \forall\ (x,y) \in \triangle, \tag{4.2}$$

where

$$\chi = \sqrt{(x-y)^2 + 4(1-x-y)} = \frac{2-\xi-\eta}{2}. \tag{4.3}$$

$T$ maps the vertices $(-1,-1)$, $(1,-1)$ and $(-1,1)$ of the square $\square$ to the vertices $(0,0)$, $(1,0)$ and $(0,1)$ of the triangle $\triangle$, respectively, while the middle point

$(1/2, 1/2)$ of the hypotenuse is the image of the vertex $(1, 1)$ of $\square$. In other words, $T$ deforms two edges ($\xi = 1$ and $\eta = 1$) of $\square$ into the hypotenuse of $\triangle$.

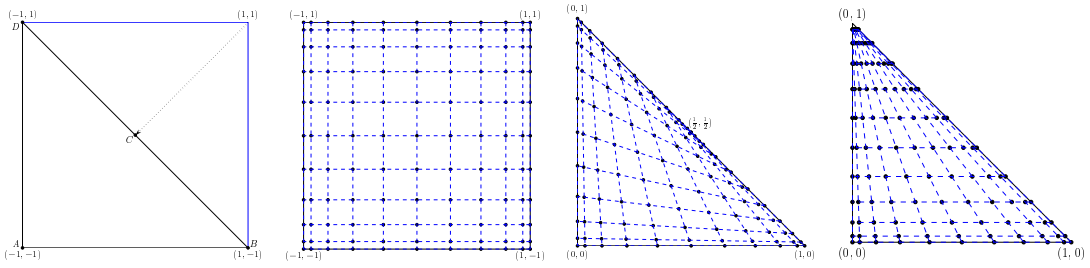
We reproduce in Fig. 4.1 the figure from the conference note [82] comparing the mapping (4.1)–(4.2) (Fig. 4.1 (a) illustrating the point mapped to the fourth vertex of $\square$) to Duffy's transform [45]

$$x = \frac{1}{4}(1+\xi)(1-\eta), \quad y = \frac{1}{2}(1+\eta), \quad \forall(\xi,\eta) \in \square, \tag{4.4}$$

with the inverse transform:

$$\xi = \frac{2x}{1-y} - 1, \quad \eta = 2y - 1, \quad \forall(x,y) \in \triangle. \tag{4.5}$$

showing a much more desirable distribution of the mapped LGL points (cf. Fig. 4.1 (c) vs. (d); LGL points shown in Fig. 4.1 (b)). Moreover, the mapping (4.1)–(4.2) is one-to-one, while Duffy's transform is one-to-many on the upper vertex of the triangle.



(a) (b) (c) (d)

Figure 4.1: (a) $\triangle \leftrightarrow \square$ mapping; (b) tensorial Legendre-Gauss-Lobatto (LGL) points on $\square$; (c) mapped LGL grids on $\triangle$; (d) mapped LGL grids on $\triangle$ using Duffy's transform.

Under $T$, we have

$$\frac{\partial x}{\partial \xi} = \frac{3-\eta}{8}, \quad \frac{\partial x}{\partial \eta} = -\frac{1+\xi}{8}, \quad \frac{\partial y}{\partial \xi} = -\frac{1+\eta}{8}, \quad \frac{\partial y}{\partial \eta} = \frac{3-\xi}{8}, \tag{4.6}$$

and the Jacobian is given by

$$J = \det\left(\frac{\partial(x,y)}{\partial(\xi,\eta)}\right) = \frac{2-\xi-\eta}{16} = \frac{\chi}{8} = \frac{\sqrt{(x-y)^2 + 4(1-x-y)}}{8}. \tag{4.7}$$

For convenience of presentation, we use the handy notation:

$$\widetilde{\nabla} = (\partial_\xi, \partial_\eta), \quad \widetilde{\nabla}^\perp = (-\partial_\eta, \partial_\xi), \quad \widetilde{\nabla}^\intercal = (1-\xi)\partial_\xi - (1-\eta)\partial_\eta, \tag{4.8}$$

where we put "$\widetilde{\ }$" to distinguish them from the differential operators in $(x, y)$. Given $u(x, y)$ on $\triangle$, we define the transformed function

$$\tilde{u}(\xi, \eta) = (u \circ T)(\xi, \eta) = u(x, y), \tag{4.9}$$

and likewise for $\tilde{v}$, etc. Then we have

$$(u, v)_\triangle = \iint_\triangle u(x, y)v(x, y)\,\mathrm{d}x\,\mathrm{d}y = \iint_\square \tilde{u}(\xi, \eta)\tilde{v}(\xi, \eta)J\,\mathrm{d}\xi\,\mathrm{d}\eta. \tag{4.10}$$

Moreover, one verifies that

$$\nabla u = \big(\partial_x u, \partial_y u\big) = \chi^{-1}(2(\widetilde{\nabla} \cdot \tilde{u}) + (\widetilde{\nabla}^\intercal \tilde{u}), 2(\widetilde{\nabla} \cdot \tilde{u}) - (\widetilde{\nabla}^\intercal \tilde{u})), \tag{4.11}$$

and

$$(\nabla u, \nabla v)_\triangle = \iint_\square (\widetilde{\nabla} \cdot \tilde{u})(\widetilde{\nabla} \cdot \tilde{v})\chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta + \frac{1}{4}\iint_\square (\widetilde{\nabla}^\intercal \tilde{u})(\widetilde{\nabla}^\intercal \tilde{v})\chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta. \tag{4.12}$$

We observe from (4.11)–(4.12) that, if $\nabla u$ is continuous at the middle point $(1/2, 1/2)$ of the hypotenuse of $\triangle$ (note: $(\widetilde{\nabla}^\intercal \tilde{u})|_{(1,1)} = 0$), there automatically holds:

$$(\widetilde{\nabla} \cdot \tilde{u})|_{(1,1)} = \left(\frac{\partial \tilde{u}}{\partial \xi} + \frac{\partial \tilde{u}}{\partial \eta}\right)\bigg|_{(1,1)} = 0, \tag{4.13}$$

which is referred to as the *consistency condition*, and can be viewed as an analogy of the pole condition in the polar/spherical coordinates. In general, we have to build the condition (4.13) in the approximation space so as to obtain high-order accuracy, which therefore results in the reduction of dimension and modification of the usual basis functions (cf. [82]).

One important goal is to demonstrate that this singularity can be removed, thanks to the observation:

$$\iint_\square \frac{1}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta = \frac{1}{2}\iint_\square \chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta = 4\ln 2, \tag{4.14}$$

which implies that, for any $f \in C(\overline{\square})$,

$$\left| \iint_{\square} \frac{f(\xi,\eta)}{2-\xi-\eta} \, \mathrm{d}\xi \, \mathrm{d}\eta \right| \leq 4M \ln 2, \tag{4.15}$$

where

$$M = \max_{\overline{\square}} |f(\xi,\eta)|.$$

In particular, the coordinate singularity can be eliminated, if $f$ is a polynomial on $\square$ (see Subsec. 4.2.2).

Now, we present other important features of $T$. Denote by

$$\mathbb{P}_N(\triangle) := \mathrm{span}\{x^i y^j : 0 \leq i+j \leq N\}, \quad \mathbb{Q}_N(\square) := (\mathbb{P}_N(I))^2. \tag{4.16}$$

That is: the polynomial space on $\triangle$ has polynomials of *total degree* at most $N$, whereas the polynomial space on $\square$ has degree at most $N$ *in both variables*.

The following property shows the correspondence between the two polynomial spaces.

**Proposition 4.1.** *We have*

(*i*) $\mathbb{P}_N(\triangle) \circ T \subset \mathbb{Q}_N(\square)$.

(*ii*) $\mathbb{Q}_N(\square) = (\mathbb{P}_N(\triangle) \circ T) \oplus \chi(\mathbb{P}_{N-1}(\triangle) \circ T)$.

*Here, $T$ is the rectangle-triangle transform defined by* (4.1)*, and $\chi$ is as defined in* (4.3).

**Proof**: We find from (4.1) that, for $0 \leq i+j \leq N$,

$$x^i y^j = \left(\frac{1+\xi}{2}\right)^i \left(\frac{3-\eta}{4}\right)^i \left(\frac{3-\xi}{4}\right)^j \left(\frac{1+\eta}{2}\right)^j \in \mathbb{Q}_N(\square).$$

This leads to the inclusion in *(*i*)*.

We see that, for $0 \leq i+j < N$,

$$x^i y^j \chi = \left(\frac{1+\xi}{2}\right)^i \left(\frac{3-\eta}{4}\right)^i \left(\frac{3-\xi}{4}\right)^j \left(\frac{1+\eta}{2}\right)^j \frac{2-\xi-\eta}{2} \in \mathbb{Q}_N(\square),$$

which implies $\chi(\mathbb{P}_{N-1}(\triangle)\circ T)\subset\mathbb{Q}_N(\square)$.

It remains to prove $\mathbb{Q}_N(\square)\subset(\mathbb{P}_N(\triangle)\circ T)\oplus\chi(\mathbb{P}_{N-1}(\triangle)\circ T)$, which we will show by induction. Firstly, by (4.2), it is true for $\xi$, $\eta$, so is $\xi\eta$, since $\xi\eta=5-4x-4y-2\chi$. Now, assume that it holds for $\xi^i\eta^j$ with $0\le i,j<N$. Then, for $0\le i,j\le N$, we find that

$$\xi^N\eta^j=\xi(\xi^{N-1}\eta^j),\quad \xi^i\eta^N=\eta(\xi^i\eta^{N-1}),\quad \xi^N\eta^N=(\xi\eta)(\xi^{N-1}\eta^{N-1})$$

are all of the form $(a+bx+cy+d\chi)(p(x,y)+q(x,y)\chi)$, where $a,b,c,d$ are constants, $p\in\mathbb{P}_{N-1}(\triangle)$ and $q\in\mathbb{P}_{N-2}(\triangle)$. It is apparent that

$$\begin{aligned}(a+bx+cy+d\chi)(p+q\chi)&=(a+bx+cy)p+dp\chi+(a+bx+cy)q\chi+dq\chi^2\\&\overset{(4.2)}{=}(a+bx+cy)p+d((x-y)^2+4(1-x-y))q+(dp+(a+bx+cy)q)\chi.\end{aligned}$$

Since $(a+bx+cy)p, d\chi^2q\in\mathbb{P}_N(\triangle)$ and $dp,(a+bx+cy)q\in\mathbb{P}_{N-1}(\triangle)$, we have

$$\xi^N\eta^j,\xi^i\eta^N,\xi^N\eta^N\in(\mathbb{P}_N(\triangle)\circ T)\oplus\chi(\mathbb{P}_{N-1}(\triangle)\circ T),$$

for all $0\le i,j\le N$. This completes the induction. ∎

In what follows, let $\omega>0$ be a generic weight function on $\Omega=\triangle$ or $\square$. The weighted Sobolev space $H^r_\omega(\Omega)$ with $r\ge0$ is defined as in Adams [1], and its norm and semi-norm are denoted by $\|\cdot\|_{r,\omega,\Omega}$ and $|\cdot|_{r,\omega,\Omega}$, respectively. In particular, if $r=0$, we denote the inner product and norm of $L^2_\omega(\Omega)$ by $(\cdot,\cdot)_{\omega,\Omega}$ and $\|\cdot\|_{\omega,\Omega}$, respectively. Moreover, if $\omega\equiv1$, we drop it from the notation.

**Proposition 4.2.** *For any $u\in H^1(\triangle)$, we have*

$$\begin{aligned}\frac{\sqrt{6}}{4}\|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}+\frac{1}{4}\|\widetilde{\nabla}^\perp\cdot\tilde{u}\|_{\chi,\square}&\le\|\nabla u\|_\triangle\\&\le\frac{\sqrt{5}}{2}\|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}+\frac{1}{2}\|\widetilde{\nabla}^\perp\cdot\tilde{u}\|_{\chi,\square},\end{aligned}\tag{4.17}$$

*where $\chi$ is defined in* (4.3)*, $\tilde{u}$ is defined in* (4.9) *and the differential operators are defined in* (4.8)*.*

**Proof**: By (4.12), we have

$$\|\nabla u\|_{\triangle}^2 = \|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}^2 + \frac{1}{4}\|\widetilde{\nabla}^{\intercal}\tilde{u}\|_{\chi^{-1},\square}^2.$$

Then, using the identity

$$\widetilde{\nabla}^{\intercal}\tilde{u} = (1-\xi)\partial_{\xi}\tilde{u} - (1-\eta)\partial_{\eta}\tilde{u} = \frac{1}{2}[2\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u}) - (\xi-\eta)(\widetilde{\nabla}\cdot\tilde{u})],$$

we obtain

$$\|\nabla u\|_{\triangle}^2 = \|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}^2 + \frac{1}{16}\|2\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u}) - (\xi-\eta)(\widetilde{\nabla}\cdot\tilde{u})\|_{\chi^{-1},\square}^2. \tag{4.18}$$

As $|\xi-\eta|\le 2$, we get

$$\frac{1}{16}\|2\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u}) - (\xi-\eta)(\widetilde{\nabla}\cdot\tilde{u})\|_{\chi^{-1},\square}^2 \le \frac{1}{4}\|\widetilde{\nabla}^{\perp}\cdot\tilde{u}\|_{\chi,\square}^2 + \frac{1}{4}\|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}^2.$$

Thus, the upper bound of (4.17) is a consequence of (4.18).

It is clear that

$$-4(\xi-\eta)\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u})(\widetilde{\nabla}\cdot\tilde{u}) \ge -[2\chi^2|\widetilde{\nabla}^{\perp}\cdot\tilde{u}|^2 + 2(\xi-\eta)^2|\widetilde{\nabla}\cdot\tilde{u}|^2].$$

Thus,

$$\begin{aligned}[2\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u}) - (\xi-\eta)(\widetilde{\nabla}\cdot\tilde{u})]^2 &\ge 2\chi^2|\widetilde{\nabla}^{\perp}\cdot\tilde{u}|^2 - (\xi-\eta)^2|\widetilde{\nabla}\cdot\tilde{u}|^2\\ &\ge 2\chi^2|\widetilde{\nabla}^{\perp}\cdot\tilde{u}|^2 - 4|\widetilde{\nabla}\cdot\tilde{u}|^2,\end{aligned}$$

which implies

$$\frac{1}{16}\|2\chi(\widetilde{\nabla}^{\perp}\cdot\tilde{u}) - (\xi-\eta)(\widetilde{\nabla}\cdot\tilde{u})\|_{\chi^{-1},\square}^2 \ge \frac{1}{8}\|\widetilde{\nabla}^{\perp}\cdot\tilde{u}\|_{\chi,\square}^2 - \frac{1}{4}\|\widetilde{\nabla}\cdot\tilde{u}\|_{\chi^{-1},\square}^2.$$

Therefore, the lower bound of (4.17) follows from (4.18), and the fundamental inequality: $A^2+B^2 \ge (A+B)^2/2$. ∎

**Remark 4.1.** *We find from Prop. 4.2 that, under the rectangle-triangle mapping* (4.1)*, the space* $H^1(\triangle)$ *is mapped to the weighted space on* $\square$*:*

$$\widetilde{H}_{\chi}^1(\square) := \{\tilde{u}\in L_{\chi}^2(\square) : \widetilde{\nabla}\cdot\tilde{u}\in L_{\chi}^2(\square)\,, \widetilde{\nabla}^{\perp}\cdot\tilde{u}\in L_{\chi^{-1}}^2(\square)\}, \tag{4.19}$$

*and vice verse.*

### 4.1.2 Some new perspectives and a comparison study

Next, we have some insights of the rectangle-triangle mapping (4.1) and compare it with Duffy's transform [45].

Firstly, the transform (4.1) is a special case of the general mapping $T_\theta : \square \mapsto \triangle$:

$$(x, y) = \left( \frac{1+\xi}{2} \frac{2-(1-\theta)(1+\eta)}{2}, \frac{1+\eta}{2} \frac{2-\theta(1+\xi)}{2} \right), \quad \forall (\xi, \eta) \in \square, \quad (4.20)$$

with $\theta = 1/2$. We see that this mapping pulls the hypotenuse of $\triangle$ into two edges of $\square$ at the point $(\theta, 1-\theta)$.

The limiting case with $\theta = 0$ reduces to Duffy's transform $T_0 = T_D$ (4.4)–(4.5). $T_D$ collapses one edge, $\eta = 1$, of $\square$ into the vertex $(0, 1)$ of $\triangle$. As the singular vertex corresponds to one edge, Duffy's transform is not a one-to-one mapping, as opposite to (4.1), or indeed (4.20) for $0 < \theta < 1$. This results in a large portion of mapped LGL points clustered near the singular vertex of $\triangle$ (see Fig. 4.1 (d)). The Jacobian of (4.4)–(4.5) is $J = (1-\eta)/8$, and we have

$$\nabla u = \left( \frac{4}{1-\eta} \partial_\xi \tilde{u}, \frac{2(1+\xi)}{1-\eta} \partial_\xi \tilde{u} + 2\partial_\eta \tilde{u} \right). \quad (4.21)$$

Different from (4.13), the corresponding consistency condition of Duffy's transform becomes $\partial_\xi \tilde{u}(\xi, 1) = 0$. In a distinct contrast with (4.14), the integral

$$\iint_\square \frac{1}{1-\eta} \, \mathrm{d}\xi \, \mathrm{d}\eta = \infty. \quad (4.22)$$

Consequently, the consistency condition has to be built in the approximation space, and much care has to be taken to deal with this singularity for Duffy's-transform-based methods in terms of implementation and analysis.

Secondly, the nature of the point singularity of (4.1) is reminiscent to that of the Gordon-Hall mapping [60], which maps the reference square to the unit disc via

$$x = \frac{\xi}{\sqrt{2}} \sqrt{2 - \eta^2}, \quad y = \frac{\eta}{\sqrt{2}} \sqrt{2 - \xi^2}, \quad \forall (\xi, \eta) \in \square,$$

and whose Jacobian is $(2-\xi^2-\eta^2)/\sqrt{(2-\xi^2)(2-\eta^2)}$. It is clear that this transform induces singularity at four vertices of the reference square (cf. Fig. 4.2). It is worthwhile to point out that the collocation scheme on the unit disc using this mapping was discussed in [67], and this mapping technique was further examined in [15].

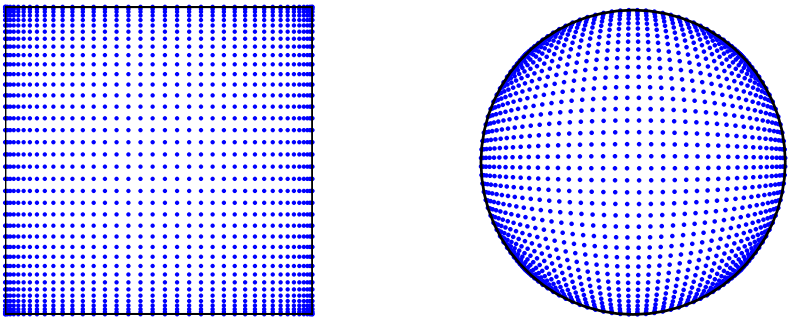


Figure 4.2: Left: tensorial Legendre-Gauss-Lobatto points on the square. Right: the corresponding mapped LGL points on the unit disc.

In addition, we find that the rectangle-triangle transform (4.1) can be derived from the symmetric mapping on $\square$:

$$\hat{x} = \xi + \eta, \quad \hat{y} = \xi\eta, \quad \forall\,(\xi,\eta) \in \square. \tag{4.23}$$

It transforms any symmetric polynomial in $(\xi,\eta)$ to a polynomial in $(\hat{x},\hat{y})$, so it is referred to as a symmetric mapping [113]. One verifies that the image of this mapping is the curvilinear triangle (see Fig. 4.3 (b)):[1]

$$\Omega = \{(\hat{x},\hat{y}) : 1-\hat{x}+\hat{y}, 1+\hat{x}+\hat{y}, \hat{x}^2-4\hat{y} > 0\}.$$

As the symmetric mapping (4.23), denoted by $\widehat{T} : \square \mapsto \Omega$, cannot distinguish the images of $(\xi,\eta)$ and $(\eta,\xi)$, it is not one-to-one. To amend this, one may restrict the domain of $\widehat{T}$ to the upper triangle, denoted by $\triangle_{\text{up}}$, in $\square$ (see Fig. 4.3 (a)), and interestingly, the square of maximum area contained in this subdomain is

[1]It is worthwhile to note that thanks to the symmetric mapping $\widehat{T} : \square \mapsto \Omega$, Xu [119] discovered the first example of multivariate Gauss quadrature.

one-to-one mapped to the triangle of maximum area included in the curvilinear triangle $\Omega$, that is,

$$\widehat{T} : \widehat{\square} := (-1, 0) \times (0, 1) \longmapsto \widehat{\triangle} := \{(\hat{x}, \hat{y}) : |\hat{x}| < 1 + \hat{y} < 1\}, \tag{4.24}$$

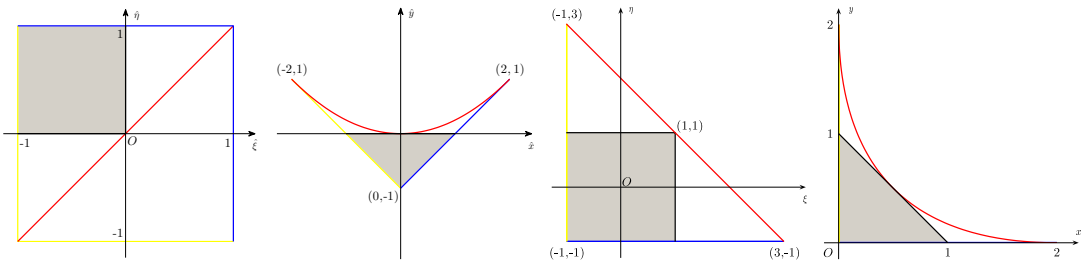
is a bijective mapping (see the shaded parts in Fig. 4.3 (a)–(b)). For clarity of presentation, we denote the coordinate of any point in $\widehat{\square}$ by $(\hat{\xi}, \hat{\eta})$. It is clear that the reference square $\square$ and $\widehat{\square}$ are connected by the affine mapping $F_1 : \square \mapsto \widehat{\square}$, of the form (see the shaded parts of Fig. 4.3 (a), (c)):

$$\hat{\xi} = \frac{\xi - 1}{2}, \quad \hat{\eta} = \frac{1 - \eta}{2}, \quad \forall\, (\xi, \eta) \in \square, \tag{4.25}$$

and the affine mapping $F_2 : \widehat{\triangle} \mapsto \triangle$ takes the form (see the shaded parts of Fig. 4.3 (b), (d)):

$$x = \frac{1}{2}(\hat{y} + \hat{x} + 1), \quad y = \frac{1}{2}(\hat{y} - \hat{x} + 1), \quad \forall\, (\hat{x}, \hat{y}) \in \widehat{\triangle}. \tag{4.26}$$

In summary, we have $\square \overset{F_1}{\longmapsto} \widehat{\square} \overset{\widehat{T}}{\longmapsto} \widehat{\triangle} \overset{F_2}{\longmapsto} \triangle$. Remarkably, this composite mapping is identical to the rectangle-triangle mapping (4.1), i.e., $T = F_1 \circ \widehat{T} \circ F_2$.




(a) (b) (c) (d)

Figure 4.3: (a) The reference square $\square$, the upper triangle $\triangle_{\text{up}} = \{(x, y) : -1 < x < y < 1\}$ and the square $\widehat{\square}$ (shaded); (b) The image $\Omega$ (resp. $\widehat{\triangle}$ (shaded)) of the symmetric mapping $\widehat{T}$, whose domain is $\triangle_{\text{up}}$ (resp. $\widehat{\square}$ (shaded)); (c) Domains obtained from $\widehat{\square}$ and the upper triangle $\triangle_{\text{up}}$ in (a) by the affine mapping $F_1$; (d) Domains obtained from $\widehat{\triangle}$ and $\Omega$ in (b) by the affine mapping $F_2$.

## 4.2 Basis functions and computation of the stiffness matrix

We introduce in this section the modal and nodal basis functions on triangles, and present a fast and accurate algorithm for computing the stiffness matrix, with a focus on how to deal with the singularity (cf. (4.14)–(4.15)).

Observe from Prop. 4.1 that for any polynomial $p$ on $\triangle$, its transformed counterpart $\tilde{p}$ is a polynomial on $\square$, but a general polynomial on $\square$ is mapped to a rational or an irrational function on $\triangle$.

In order to obtain orthogonal polynomials on both $\triangle$ and $\square$, Dubiner [44] introduced the polynomial basis using Duffy's transform (4.4):

$$\begin{aligned} D_{mn}(x,y) &= (1-y)^m P_m\left(\frac{2x}{1-y}-1\right) P_n^{2m+1,0}(2y-1) \\ &= P_m(\xi)\left(\frac{1-\eta}{2}\right)^m P_n^{2m+1,0}(\eta) = \widetilde{D}_{mn}(\xi,\eta). \end{aligned} \tag{4.27}$$

Here, $P_k^{\alpha,\beta}(\zeta)$, $\alpha,\beta > -1$, are the Jacobi polynomials (see, e.g. [106]), while, as previous, $P_k(\zeta)$ are the Legendre polynomials.

The so-defined $\{P_{mn}\}$ are orthogonal under the inner product (4.10). As the degree $m$ appears in the parameter of $P_n^{2m+1,0}$, this basis is not built upon a full tensor structure, which is called a warped tensor basis. Interested readers are referred to the monograph of Karniadakis and Sherwin [74] for a very extensive development of this notion. The error analysis of this approach can be found in [65, 19, 80, 27].

By dropping the requirement of being polynomials on $\triangle$, Shen, Wang and Li [103] introduced fully-tensorial rational basis functions based on Duffy's transform (4.4):

$$R_{mn}(x,y) = P_m\left(\frac{2x}{1-y}-1\right) P_n^{1,0}(2y-1) = P_m(\xi) P_n^{1,0}(\eta) = \tilde{R}_{mn}(\xi,\eta), \tag{4.28}$$

which are orthogonal with respect to $(\cdot,\cdot)_\triangle$. This allows the implementation to be as efficient as that on rectangles (see further development in [25]).

### 4.2.1 Modal basis

We define the space

$$Y_N(\triangle) = \mathbb{Q}_N(\square) \circ T^{-1} = (\mathbb{P}_N(I))^2 \circ T^{-1}, \tag{4.29}$$

which consists of the images of the tensor-product polynomials on $\square$ under the inverse mapping $T^{-1}$ defined in (4.2). As a direct consequence of Prop. 4.1 (*ii*), we have

$$Y_N(\triangle) = \mathbb{P}_N(\triangle) \oplus \chi\mathbb{P}_{N-1}(\triangle), \tag{4.30}$$

where $\chi$ is defined in (4.3) and $\mathbb{P}_N(\triangle)$ is defined in (4.16). This implies that $Y_N(\triangle)$ contains not only polynomials, but also special irrational functions: $\chi\phi$ for any $\phi \in \mathbb{P}_{N-1}(\triangle)$.

Define

$$\phi_0(\zeta) = \frac{1-\zeta}{2}, \quad \phi_k(\zeta) = \frac{1-\zeta^2}{4} P_{k-1}^{1,1}(\zeta),\ 0 < k < N, \quad \phi_N(\zeta) = \frac{1+\zeta}{2}. \tag{4.31}$$

It is clear that $\{\phi_k\}_{k=0}^N$ forms a basis of $\mathbb{P}_N(I)$, and we have

$$\mathbb{Q}_N(\square) = \text{span}\{\Phi_{kl} : \Phi_{kl}(\xi,\eta) = \phi_k(\xi)\phi_l(\eta), 0 \le k,l \le N\}. \tag{4.32}$$

(4.31) is a commonly used $C^0$-modal basis for QSEM, which enjoys a distinct separation of the interior and boundary modes (including vertex and edge modes). All interior modes are zero on the square boundary. The vertex modes have a unit magnitude at one vertex and are zero at all other vertices, and the edge modes only have magnitude along one edge and are zero at all other vertices and edges.

In view of (4.29) and (4.32), we obtain the modal basis for $Y_N(\triangle)$:

$$Y_N(\triangle) = \text{span}\{\Psi_{kl} : \Psi_{kl}(x,y) = \Phi_{kl} \circ T^{-1}, 0 \le k,l \le N\}. \tag{4.33}$$

### 4.2.2 Computation of the stiffness matrix

Though the singular integral of (4.15)-type has a finite value, some efforts are needed to compute such integrals in a fast and stable manner. Next, we devise an efficient algorithm for this purpose.

Recall that (see e.g., [106])

$$(1-\zeta^2)P_{k-1}^{1,1}(\zeta) = \frac{2k}{2k+1}(P_{k-1}(\zeta) - P_{k+1}(\zeta)), \tag{4.34}$$

$$\zeta P_k(\zeta) = \frac{k}{2k+1}P_{k-1}(\zeta) + \frac{k+1}{2k+1}P_{k+1}(\zeta). \tag{4.35}$$

Thus, we have, from (4.31), (4.34) and (2.11),

$$\phi_0'(\zeta) = -\frac{1}{2}P_0(\zeta) = -\phi_N'(\zeta), \quad \phi_k'(\zeta) = -\frac{k}{2}P_k(\zeta), 0 < k < N. \tag{4.36}$$

By (4.8), (4.11) and (4.32),

$$\begin{aligned}
\chi\partial_x\Psi_{kl} &= 2(\phi_k'(\xi)\phi_l(\eta) + \phi_k(\xi)\phi_l'(\eta)) \\
&\quad + [(1-\xi)\phi_k'(\xi)\phi_l(\eta) - (1-\eta)\phi_k(\xi)\phi_l'(\eta)], \\
\chi\partial_y\Psi_{kl} &= 2(\phi_k'(\xi)\phi_l(\eta) + \phi_k(\xi)\phi_l'(\eta)) \\
&\quad - [(1-\xi)\phi_k'(\xi)\phi_l(\eta) - (1-\eta)\phi_k(\xi)\phi_l'(\eta)].
\end{aligned} \tag{4.37}$$

Thanks to (4.34)–(4.36), $\chi\partial_x\Psi_{kl}$ and $\chi\partial_y\Psi_{kl}$ can be represented by a linear combination of $\{P_{k\pm i}(\xi)P_{l\pm j}(\eta)\}_{i,j=0,1}$. In view of this, we can evaluate the entries of the stiffness matrix by computing the integrals of the product of Legendre polynomials:

$$s_{kl}^{k'l'} := \iint_{\triangle} \nabla\Psi_{kl}\cdot\nabla\Psi_{k'l'}\,\mathrm{d}x\,\mathrm{d}y \leftrightarrow \iint_{\square} \frac{P_i(\xi)P_j(\eta)P_{i'}(\xi)P_{j'}(\eta)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta := a_{ij}^{i'j'}. \tag{4.38}$$

Using the fact that the product $P_mP_n$ can be represented by $\{P_p\}_{p=0}^{m+n}$:

$$P_m(\zeta)P_n(\zeta) = \sum_{p=0}^{m+n} c_p^{mn}P_p(\zeta), \tag{4.39}$$

where the expansion coefficient $\{c_p^{mn}\}$ can be found in, e.g., [73], we obtain

$$a_{ij}^{i'j'} = \sum_{p=0}^{i+i'}\sum_{q=0}^{j+j'} c_p^{ii'}c_q^{jj'}\hat{a}_{pq}, \tag{4.40}$$

where

$$\hat{a}_{pq} = \iint_{\square} \frac{P_p(\xi)P_q(\eta)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta.$$

Now, we describe how to compute $\{\hat{a}_{pq}\}$ in a fast and accurate manner. This essentially relies on the following recurrence relation.

**Lemma 4.1.** *We have*

$$\frac{\hat{a}_{p,q+1}-\hat{a}_{p,q-1}}{2q+1}=\frac{\hat{a}_{p+1,q}-\hat{a}_{p-1,q}}{2p+1},\quad \forall p,q\geq 1. \tag{4.41}$$

**Proof**: The statement is true for $p=q\geq 1$, since $\hat{a}_{p,p\pm 1}=\hat{a}_{p\pm 1,p}$. In view of the symmetry $\hat{a}_{pq}=\hat{a}_{qp}$, it suffices to show (4.41) holds for $p>q\geq 1$. We start with recalling the Legendre functions of the second kind (see, [106, eq. (4.61.4)]) are defined as

$$Q_n(x)=\frac{1}{2}\int_{-1}^{1}\frac{P_n(t)}{x-t}\,\mathrm{d}t,\quad n\geq 1;\quad Q_0(x)=\frac{1}{2}\ln\frac{x+1}{x-1},\quad \forall\, x>1, \tag{4.42}$$

with the important identity (see [106, eq. (4.62.1)]):

$$\begin{aligned} Q_n(x)&=\frac{1}{2}\left(\ln\frac{x+1}{x-1}\right)P_n(x)-\frac{1}{2}\int_{-1}^{1}\frac{P_n(x)-P_n(t)}{x-t}\,\mathrm{d}t\\ &=\frac{1}{2}\left(\ln\frac{x+1}{x-1}\right)P_n(x)-\widetilde{P}_{n-1}(x). \end{aligned} \tag{4.43}$$

Here, $\widetilde{P}_n$ is the Legendre polynomial of the second kind, satisfying

$$\begin{aligned} &\widetilde{P}_{-1}(x)\equiv 0,\quad \widetilde{P}_0(x)\equiv 1,\\ &\widetilde{P}_n(x)=\frac{2n+1}{n+1}x\widetilde{P}_{n-1}(x)-\frac{n}{n+1}\widetilde{P}_{n-2}(x),\quad n\geq 1, \end{aligned} \tag{4.44}$$

which follows from (4.43) and [106, eq. (4.62.13)] directly. From (4.42)–(4.44) and the orthogonality of Legendre polynomials (2.10), we find that, for $p>q\geq 1$,

$$\begin{aligned} \hat{a}_{pq}&=\int_{-1}^{1}\int_{-1}^{1}\frac{P_p(\xi)P_q(\eta)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta=2\int_{-1}^{1}Q_q(2-\xi)P_p(\xi)\,\mathrm{d}\xi\\ &=\int_{-1}^{1}\left[\left(\ln\frac{3-\xi}{1-\xi}\right)P_q(2-\xi)-2\widetilde{P}_{q-1}(2-\xi)\right]P_p(\xi)\,\mathrm{d}\xi\\ &=\int_{-1}^{1}\left(\ln\frac{3-\xi}{1-\xi}\right)P_q(2-\xi)P_p(\xi)\,\mathrm{d}\xi. \end{aligned} \tag{4.45}$$

Thus, we have, from (2.11) and integration by parts, that

$$\begin{aligned}
&\frac{\hat{a}_{p,q+1}-\hat{a}_{p,q-1}}{2q+1}\\
&\quad=\int_{-1}^{1}\left(\ln\frac{3-\xi}{1-\xi}\right)\frac{P_{q+1}(2-\xi)-P_{q-1}(2-\xi)}{2q+1}P_p(\xi)\,\mathrm{d}\xi\\
&\quad=\int_{-1}^{1}\left(\ln\frac{3-\xi}{1-\xi}\right)\frac{P_{q+1}(2-\xi)-P_{q-1}(2-\xi)}{2q+1}\left[\frac{P_{p+1}(\xi)-P_{p-1}(\xi)}{2p+1}\right]'\mathrm{d}\xi\\
&\quad=-\int_{-1}^{1}\left[\left(\ln\frac{3-\xi}{1-\xi}\right)\frac{P_{q+1}(2-\xi)-P_{q-1}(2-\xi)}{2q+1}\right]'\frac{P_{p+1}(\xi)-P_{p-1}(\xi)}{2p+1}\,\mathrm{d}\xi.
\end{aligned}$$

Working out the derivative, we obtain

$$\begin{aligned}
&\frac{\hat{a}_{p,q+1}-\hat{a}_{p,q-1}}{2q+1}\\
&\overset{(2.11)}{=}\int_{-1}^{1}\left[P_q(2-\xi)\ln\left(\frac{3-\xi}{1-\xi}\right)-\frac{P_{q+1}(2-\xi)-P_{q-1}(2-\xi)}{(q+1/2)(3-\xi)(1-\xi)}\right]\frac{P_{p+1}(\xi)-P_{p-1}(\xi)}{2p+1}\,\mathrm{d}\xi\\
&\overset{(4.45)}{=}\frac{\hat{a}_{p+1,q}-\hat{a}_{p-1,q}}{2p+1}-\int_{-1}^{1}\frac{P_{q+1}(2-\xi)-P_{q-1}(2-\xi)}{(q+1/2)(3-\xi)(1-\xi)}\frac{P_{p+1}(\xi)-P_{p-1}(\xi)}{2p+1}\,\mathrm{d}\xi\\
&\overset{(4.34)}{=}\frac{\hat{a}_{p+1,q}-\hat{a}_{p-1,q}}{2p+1}+\frac{1}{2pq}\int_{-1}^{1}P_{q-1}^{1,1}(2-\xi)P_{p-1}^{1,1}(\xi)(1-\xi^2)\,\mathrm{d}\xi=\frac{\hat{a}_{p+1,q}-\hat{a}_{p-1,q}}{2p+1},
\end{aligned}$$

where we used the fact $p>q$ and the orthogonality of Jacobi polynomials in the last step. ■

Equipped with (4.41), we are able to compute $\{\hat{a}_{pq}\}_{p\geq q}$ accurately and rapidly. We summarize the algorithm as follows.

---

**Algorithm for computing $\{\hat{a}_{pq}\}_{p,q=0}^{N}$**

1. Initialization

   (a) For $p=0,1,\ldots,2N$, compute $\hat{a}_{p0}$;

   (b) For $p=1,2,\ldots,2N-1$, compute $\hat{a}_{p1}$

2. For $q=2,3,\ldots,N$,

   For $p=q,\ldots,2N-q$,

$$\hat{a}_{pq}=\hat{a}_{p,q-2}+\frac{2q-1}{2p+1}(\hat{a}_{p+1,q-1}-\hat{a}_{p-1,q-1}), \tag{4.46}$$

   Endfor of $p,q$.

3. Set $\hat{a}_{pq} = \hat{a}_{qp}$ for all $0 \le p < q < N$.

---

We describe below the details for computing the initial values.

- We find from (4.45) that

$$\begin{aligned}\hat{a}_{p0} &= \int_{-1}^{1} P_p(\xi) \ln \frac{3-\xi}{1-\xi}\,\mathrm{d}\xi \\ &= \int_{-1}^{1} P_p(\xi) \ln \frac{3-\xi}{2}\,\mathrm{d}\xi + \int_{-1}^{1} P_p(\xi) \ln \frac{2}{1-\xi}\,\mathrm{d}\xi := \alpha_p + \beta_p. \end{aligned} \tag{4.47}$$

  It is clear that, by (2.11) and integration by parts,

$$\alpha_p = \int_{-1}^{1} P_p(\xi) \ln \frac{3-\xi}{2}\,\mathrm{d}\xi = \frac{1}{2p+1}\left(\int_{-1}^{1} \frac{P_{p+1}(\xi)}{3-\xi}\,\mathrm{d}\xi - \int_{-1}^{1} \frac{P_{p-1}(\xi)}{3-\xi}\,\mathrm{d}\xi\right).$$

  $\alpha_p$ decays exponentially with respect to $p$, and the use of a Legendre-Gauss quadrature leads to an exponentially accurate approximation, since the function $1/(3-\xi)$ is analytic within an ellipse (see [118]). We find from, e.g., [58] that

$$\beta_p = \int_{-1}^{1} P_p(\xi) \ln \frac{2}{1-\xi}\,\mathrm{d}\xi = \begin{cases} 2, & \text{if } p = 0, \\ \dfrac{2}{p(p+1)}, & \text{if } p \ge 1. \end{cases}$$

- Using (4.40), (4.35) and the orthogonality of Legendre polynomials, we find

$$\begin{aligned}\hat{a}_{p1} &= \iint_{\square} \frac{\eta P_p(\xi)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta \\ &= \iint_{\square} \frac{(2-\xi)P_p(\xi)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta - \iint_{\square} \frac{(2-\xi-\eta)P_p(\xi)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta \\ &= 2\iint_{\square} \frac{P_p(\xi)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta - \iint_{\square} \frac{\xi P_p(\xi)}{2-\xi-\eta}\,\mathrm{d}\xi\,\mathrm{d}\eta - \int_{-1}^{1}\left[\int_{-1}^{1} P_p(\xi)\,\mathrm{d}\xi\right]\mathrm{d}\eta \\ &= 2\hat{a}_{p0} - \frac{(p+1)\hat{a}_{p+1,0} + p\hat{a}_{p-1,0}}{2p+1}, \quad p \ge 1. \end{aligned} \tag{4.48}$$

- We see that with an accurate computation of the initial values $\{\hat{a}_{p0}\}$, marching by (4.48) and (4.46) is expected to be stable. In Fig. 4.4, we provide a schematic illustration of sweeping the stencils by the **Algorithm**.

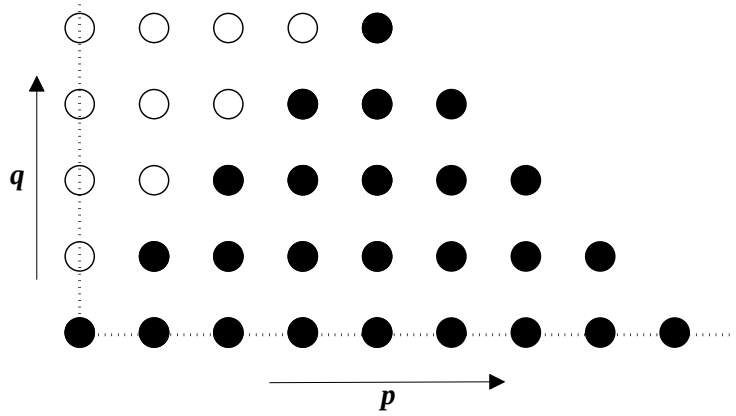



Figure 4.4: Diagram for computing $\{\hat{a}_{pq}\}_{p,q=0}^{N}$ with $N = 2$, where the stencils marked by "•" are marched via Steps 1–2 in the **Algorithm**, and those marked by "○" are obtained by the symmetric property in Step 3.

**Remark 4.2.** *We see that the rectangle-triangle mapping* (4.1) *essentially induces logarithmic singularity. Indeed, numerical quadrature of integrands involving a logarithmic weight function is of independent interest* (see e.g., [58]).

**Remark 4.3.** *As a quick note, the mass matrix under this basis is sparse. Indeed, by* (4.10)*,*

$$(u,v)_{\triangle} = \frac{1}{8}\iint_{\square} \tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta - \frac{1}{16}\iint_{\square} \xi\tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta - \frac{1}{16}\iint_{\square} \eta\tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta, \tag{4.49}$$

*so we claim this from* (4.35) *and the orthogonality of Legendre polynomials.*

**Remark 4.4.** *With an additional affine mapping, any triangular element* $\triangle_{\text{any}}$ *can be transformed to the reference square* $\square$*. It is important to point out that the stiffness and mass matrices on* $\triangle_{\text{any}}$ *can be precomputed in a similar fashion as above. To justify this, we consider a general triangle* $\triangle_{\text{any}}$ *with vertices* $V_i = (x_i, y_i)$*,* $i = 1, 2, 3$*. Then we have the mapping:*

$$\begin{aligned}(x,y) = (x_1,y_1)\frac{(1-\xi)(1-\eta)}{4} + (x_2,y_2)\frac{(1+\xi)(3-\eta)}{8} \\ + (x_3,y_3)\frac{(3-\xi)(1+\eta)}{8}, \quad \forall\,(\xi,\eta)\in\square.\end{aligned} \tag{4.50}$$

*A direct calculation leads to*

$$(u,v)_{\triangle_{\text{any}}} = \frac{F}{8}\iint_{\square}\tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta - \frac{F}{16}\iint_{\square}\xi\tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta - \frac{F}{16}\iint_{\square}\eta\tilde{u}\tilde{v}\,\mathrm{d}\xi\,\mathrm{d}\eta, \quad (4.51)$$

*and*

$$\begin{aligned}(\nabla u,\nabla v)_{\triangle_{\text{any}}} = {} & A\iint_{\square}(\widetilde{\nabla}\cdot\tilde{u})(\widetilde{\nabla}\cdot\tilde{v})\chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta \\ & - B\iint_{\square}\left[(\widetilde{\nabla}\cdot\tilde{u})(\widetilde{\nabla}^{\intercal}\tilde{v}) + (\widetilde{\nabla}^{\intercal}\tilde{u})(\widetilde{\nabla}\cdot\tilde{v})\right]\chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta \\ & + C\iint_{\square}(\widetilde{\nabla}^{\intercal}\tilde{u})(\widetilde{\nabla}^{\intercal}\tilde{v})\chi^{-1}\,\mathrm{d}\xi\,\mathrm{d}\eta,\end{aligned} \quad (4.52)$$

*where $\chi$ is defined in* (4.3)*, the differential operators are defined in* (4.8)*, and the constants are given by*

$$\begin{aligned}F &= (x_2-x_1)(y_3-y_1) - (x_3-x_1)(y_2-y_1) \neq 0,\\ A &= ((x_2-x_3)^2 + (y_2-y_3)^2)/2F,\\ B &= ((x_2-x_1)^2 + (y_2-y_1)^2 - (x_3-x_1)^2 - (y_3-y_1)^2)/4F,\\ C &= ((2x_1-x_2-x_3)^2 + (2y_1-y_2-y_3)^2)/8F.\end{aligned}$$

*In particular, if $\triangle_{\text{any}} = \triangle$,* (4.51) *and* (4.52) (*note: $B = 0$*) *reduce to* (4.10) *and* (4.12)*, respectively.*

*As with* (4.37)*, we find from* (4.34)–(4.36) *that $\widetilde{\nabla}\cdot\Phi_{kl}$ and $\widetilde{\nabla}^{\intercal}\Phi_{kl}$ can be expressed in terms of $\{P_{k\pm i}(\xi)P_{l\pm j}(\eta)\}_{i,j=0,1}$, so the mass matrix on $\triangle_{\text{any}}$ can be precomputed by the same algorithm described previously.*

### 4.2.3 Interpolation, quadrature and nodal basis

Through the general mapping (4.50), operations (e.g., interpolation, quadrature and numerical differentiations) on a triangular element can be performed on the reference square $\square$.

Hereafter, let $\{\zeta_j\}_{j=0}^N$ be the LGL points, i.e., the zeros of $(1-\zeta^2)P_N'(\zeta)$, and let $\{L_j\}_{j=0}^N$ be the associated Lagrange basis polynomials. Given $v \in C(\bar{I})$, the

one-dimensional polynomial interpolation of $v$ is (cf. (2.5))

$$(\mathrm{I}_N^\zeta v)(\zeta) = \sum_{j=0}^{N} v(\zeta_j) L_j(\zeta) \in \mathbb{P}_N, \quad \forall\, \zeta \in \bar{I}. \tag{4.53}$$

Recall that the LGL quadrature has the exactness (2.14), where $b = 2$.

Given any $u \in C(\bar{\triangle})$, we define the interpolant of $u$ by

$$(\mathrm{I\!I}_N u)(x,y) = (\mathrm{I}_N^\xi \mathrm{I}_N^\eta \tilde{u}) \circ T^{-1} = \left( \sum_{i,j=0}^{N} (u \circ T)(\xi_i, \eta_j) L_i(\xi) L_j(\eta) \right) \circ T^{-1}, \tag{4.54}$$

where $T$ and $T^{-1}$ are defined in (4.1) and (4.2) as before, and $\{\xi_k = \eta_k = \zeta_k\}_{k=0}^N$. Notice that $\mathrm{I\!I}_N u \in Y_N(\triangle)$.

We also extend the LGL quadrature to define the discrete inner product on $\triangle$ as

$$(u,v)_{N,\triangle} = \frac{1}{8} \sum_{i,j=0}^{N} \tilde{u}(\xi_i, \eta_j) \tilde{v}(\xi_i, \eta_j) \chi(\xi_i, \eta_j) \omega_i \omega_j, \tag{4.55}$$

where $\chi$ is defined in (4.3). As a consequence of (4.10), (2.14) and (4.29)–(4.30), there holds

$$(u,v)_{N,\triangle} = (u,v)_\triangle, \quad \forall\, u \cdot v \in Y_{2N-2}(\triangle), \tag{4.56}$$

which also holds for all $u \cdot v \in \mathbb{P}_{2N-2}(\triangle)$.

Since $\{L_k L_l\}_{k,l=0}^N$ forms the nodal basis for $\mathbb{Q}_N(\square)$, we can obtain the nodal basis for $Y_N(\triangle)$:

$$Y_N(\triangle) = \mathrm{span}\{\widehat{\Psi}_{kl} : \widehat{\Psi}_{kl}(x,y) = (L_k L_l) \circ T^{-1}, 0 \le k, l \le N\}. \tag{4.57}$$

In view of (4.49), the mass matrix under this nodal basis can be computed easily as usual by tensorial LGL quadrature. However, the direct evaluation of the stiffness matrix like (4.38) is prohibitive, as there is no recursive computation. In order to surmount this obstacle, we resort to the notion of "discrete transform" (cf. [101]). Like (4.37), we have

$$\chi \partial_x \widehat{\Psi}_{kl} = 2(L_k'(\xi) L_l(\eta) + L_k(\xi) L_l'(\eta)) + [(1-\xi) L_k'(\xi) L_l(\eta) - (1-\eta) L_k(\xi) L_l'(\eta)],$$
$$\chi \partial_y \widehat{\Psi}_{kl} = 2(L_k'(\xi) L_l(\eta) + L_k(\xi) L_l'(\eta)) - [(1-\xi) L_k'(\xi) L_l(\eta) - (1-\eta) L_k(\xi) L_l'(\eta)],$$

both in $\mathbb{Q}_N(\square)$. Via a two-dimensional discrete transform, $\{\chi\partial_x\widehat{\Psi}_{kl}, \chi\partial_y\widehat{\Psi}_{kl}\}_{k,l=0}^N$ is converted to $\{P_i(\xi)P_j(\eta)\}_{i,j=0}^N$. Then the evaluation boils down to finding $\{a_{ij}^{i'j'}\}$ in (4.38) as before.

## 4.3 Estimates of orthogonal projection and interpolation errors

The section is devoted to error estimates of orthogonal projections and interpolations on triangles. These results will be essential for understanding the approximability of the basis functions, and provide important tools for error analysis of the TSEM for PDEs.

### 4.3.1 Orthogonal projections

We start with considering the projection $\Pi_N : L^2(\triangle) \to Y_N(\triangle)$, defined by

$$(\Pi_N u - u, v)_\triangle = 0, \quad \forall\, v \in Y_N(\triangle). \tag{4.58}$$

**Theorem 4.2.** *For any $u \in H^r(\triangle)$ with $r \geq 0$, we have*

$$\|\Pi_N u - u\|_\triangle \leq cN^{-r}|u|_{r,\triangle}, \tag{4.59}$$

*where $c$ is a positive constant independent of $N$ and $u$.*

**Proof**: We have

$$\|\Pi_N u - u\|_\triangle \overset{(4.58)}{=} \inf_{\phi\in Y_N(\triangle)} \|\phi - u\|_\triangle \overset{(4.30)}{\leq} \|\psi - u\|_\triangle, \quad \forall\, \psi \in \mathbb{P}_N(\triangle). \tag{4.60}$$

Now, we take $\psi$ to be the best $L^2$-approximation in $\mathbb{P}_N(\triangle)$, denoted by $\pi_N u$. By [80, Thm. 3.3],

$$\begin{aligned}\|\pi_N u - u\|_\triangle &\leq cN^{-r}\left(\sum_{k_1+k_2+k_3=r} \|\partial_x^{k_1}\partial_y^{k_2}(\partial_y - \partial_x)^{k_3} u\|^2_{\omega^{k_1,k_2,k_3},\triangle}\right)^{1/2} \\ &\leq cN^{-r}|u|_{r,\triangle},\end{aligned} \tag{4.61}$$

where

$$\omega^{k_1,k_2,k_3} = x^{k_1+k_3} y^{k_2+k_3} (1-x-y)^{k_1+k_2}$$

is a Jacobi weight function on $\triangle$. Therefore, the estimate (4.59) follows from (4.60)–(4.61). ■

We now turn to the $H^1$-projection: $\Pi_N^1 : H^1(\triangle) \to Y_N(\triangle)$ such that

$$\big(\nabla(\Pi_N^1 u - u), \nabla v\big)_{\triangle} + \big(\Pi_N^1 u - u, v\big)_{\triangle} = 0, \quad \forall\, v \in Y_N(\triangle), \tag{4.62}$$

and the $H_0^1$-projection: $\Pi_N^{1,0} : H_0^1(\triangle) \to Y_N^0(\triangle) = Y_N(\triangle) \cap H_0^1(\triangle)$, defined by

$$\big(\nabla(\Pi_N^{1,0} u - u), \nabla v\big)_{\triangle} = 0, \quad \forall\, v \in Y_N^0(\triangle), \tag{4.63}$$

where $H_0^1(\triangle)$ is defined as usual, i.e., the subspace of $H^1(\triangle)$ with functions vanishing on the boundary of $\triangle$.

**Theorem 4.3.** *For any $u \in H_0^1(\triangle) \cap H^r(\triangle)$ with $r \geq 1$, we have*

$$\|\Pi_N^{1,0} u - u\|_{\mu,\triangle} \leq c N^{\mu - r} |u|_{r,\triangle}, \quad \mu = 0, 1, \tag{4.64}$$

*where $c$ is a positive constant independent of $u$ and $N$. It also holds for any $u \in H^r(\triangle)$ with $\Pi_N^1 u$ in place of $\Pi_N^{1,0} u$.*

**Proof**: Here, we only provide the proof for the projection $\Pi_N^{1,0}$, as the estimate for $\Pi_N^1$ can be obtained in a very similar fashion.

By the Poincaré inequality, we know that the semi-norm $|\cdot|_{1,\triangle}$ is a norm of $H_0^1(\triangle)$. Hence, by the definition (4.63),

$$\|u - \Pi_N^{1,0} u\|_{1,\triangle} \leq c|\phi - u|_{1,\triangle} \leq c\|\phi - u\|_{1,\triangle}, \quad \forall\, \phi \in Y_N(\triangle). \tag{4.65}$$

It is known from (4.30) that $\mathbb{P}_N(\triangle) \subset Y_N(\triangle)$, so we can take $\phi$ to be the orthogonal projection $\pi_N^{1,0} : H_0^1(\triangle) \to \mathbb{P}_N^0(\triangle) = \mathbb{P}_N(\triangle) \cap H_0^1(\triangle)$, defined by

$$\big(\nabla(\pi_N^{1,0} u - u), \nabla v\big)_{\triangle} = 0, \quad \forall\, v \in \mathbb{P}_N^0(\triangle). \tag{4.66}$$

We quote the estimate in [80, Thm. 3.4]:

$$\begin{aligned}\|\pi_N^{1,0}u-u\|_{1,\triangle} &\le cN^{1-r}\left(\sum_{k_1+k_2+k_3=r}\|\partial_x^{k_1}\partial_y^{k_2}(\partial_y-\partial_x)^{k_3}u\|^2_{\omega_+^{k_1,k_2,k_3},\triangle}\right)^{1/2} \\ &\le cN^{1-r}|u|_{r,\triangle},\end{aligned} \tag{4.67}$$

where

$$\omega_+^{k_1,k_2,k_3}=x^{\max(k_1+k_3-1,0)}y^{\max(k_2+k_3-1,0)}(1-x-y)^{\max(k_1+k_2-1,0)}.$$

Hence, the estimate (4.64) with $\mu=1$ follows from (4.65) and (4.67).

To show (4.64) with $\mu=0$, we use a duality argument as in [28], which we sketch below. Given $g\in L^2(\triangle)$, we consider the auxiliary problem: Find $u_g\in H_0^1(\triangle)$ such that

$$a(u_g,v):=(\nabla u_g,\nabla v)_\triangle=(g,v)_\triangle\,,\quad \forall\, v\in H_0^1(\triangle)\,. \tag{4.68}$$

By a standard argument, we can show that this problem has a unique solution with the regularity $\|u_g\|_{2,\triangle}\le c\|g\|_\triangle$.

Now, taking $v=u-\Pi_N^{1,0}u$ into (4.68), we find from (4.63) and (4.64) with $\mu=1$ that

$$\begin{aligned}\left|\left(g,u-\Pi_N^{1,0}u\right)_\triangle\right| &= \left|a\left(u_g,u-\Pi_N^{1,0}u\right)\right| = \left|a\left(u_g-\Pi_N^{1,0}u_g,u-\Pi_N^{1,0}u\right)\right| \\ &\le |u_g-\Pi_N^{1,0}u_g|_{1,\triangle}|u-\Pi_N^{1,0}u|_{1,\triangle} \\ &\le cN^{-r}|u_g|_{2,\triangle}|u|_{r,\triangle}\le cN^{-r}\|g\|_\triangle|u|_{r,\triangle}.\end{aligned}$$

Finally, we derive

$$\|u-\Pi_N^{1,0}u\|_\triangle=\sup_{0\ne g\in L^2(\triangle)}\frac{\left|\left(g,u-\Pi_N^{1,0}u\right)_\triangle\right|}{\|g\|_\triangle}\le cN^{-r}|u|_{r,\triangle}.$$

This completes the proof. ∎

**Remark 4.5.** *It is seen that, benefited from the fact that $\mathbb{P}_N(\triangle)\subset Y_N(\triangle)$, we are able to obtain optimal error estimates directly from available polynomial approximation results on triangles.*

### 4.3.2 Estimation of interpolation error

Now, we estimate the error of interpolation by (4.54) on $\triangle$. The estimate of one-dimensional LGL interpolation (cf. (4.53)) is useful for our analysis (see [101, Thm. 3.44]), that is, for any $v \in H^r(I)$ with $r \geq 1$, we have

$$\| \mathrm{I}_N^\zeta v - v\|_I \leq cN^{-r} \|(1-\zeta^2)^{(r-1)/2} v^{(r)}\|_I. \tag{4.69}$$

**Theorem 4.4.** *For any $u \in H^r(\triangle)$ with $r \geq 2$,*

$$\| \mathrm{I\!I}_N u - u\|_\triangle \leq cN^{-r} B_r(u), \tag{4.70}$$

*where*

$$B_r(u) = \begin{cases} |u|_{2,\triangle} + \|(\partial_y - \partial_x)^2 u\|_{J^{-1},\triangle} + \|\nabla \cdot u\|_{J^{-1},\triangle}, & \text{if } r = 2, \\ |u|_{r,\triangle} + |u|_{r-1,\triangle}, & \text{if } r \geq 3, \end{cases} \tag{4.71}$$

*$J$ is the Jacobian as defined in* (4.7)*, and $c$ is a constant independent of $u$ and $N$.*

**Proof**: To this end, let $\mathrm{I}_d$ be the identity operator. Using (4.10), (4.54) and (4.69), we obtain

$$\begin{aligned} \| \mathrm{I\!I}_N u - u\|_\triangle &\leq c \| \mathrm{I}_N^\xi \mathrm{I}_N^\eta \tilde{u} - \tilde{u}\|_\square \\ &\leq c \|(\mathrm{I}_N^\xi - \mathrm{I}_d)(\mathrm{I}_N^\eta - \mathrm{I}_d)\tilde{u} + (\mathrm{I}_N^\xi - \mathrm{I}_d)\tilde{u} + (\mathrm{I}_N^\eta - \mathrm{I}_d)\tilde{u}\|_\square \\ &\leq c \Big( \|(\mathrm{I}_N^\xi - \mathrm{I}_d)(\mathrm{I}_N^\eta - \mathrm{I}_d)\tilde{u}\|_\square + \|(\mathrm{I}_N^\xi - \mathrm{I}_d)\tilde{u}\|_\square + \|(\mathrm{I}_N^\eta - \mathrm{I}_d)\tilde{u}\|_\square \Big) \\ &\leq cN^{-1} \|(\mathrm{I}_N^\eta - \mathrm{I}_d)\partial_\xi \tilde{u}\|_\square + c \Big( \|(\mathrm{I}_N^\xi - \mathrm{I}_d)\tilde{u}\|_\square + \|(\mathrm{I}_N^\eta - \mathrm{I}_d)\tilde{u}\|_\square \Big) \\ &\leq cN^{-r} \Big( \|(1-\eta^2)^{(r-2)/2} \partial_\xi \partial_\eta^{r-1} \tilde{u}\|_\square + \|(1-\xi^2)^{(r-1)/2} \partial_\xi^r \tilde{u}\|_\square \\ &\qquad + \|(1-\eta^2)^{(r-1)/2} \partial_\eta^r \tilde{u}\|_\square \Big). \end{aligned}$$

It remains to transform the variables $(\xi, \eta)$ back to $(x, y)$ and obtain tight upper bounds of the right-hand side using norms of $u$ on $\triangle$. By (4.6),

$$\partial_\xi \tilde{u} = \frac{1-\eta}{4} \partial_y u - \frac{3-\eta}{8} (\partial_y - \partial_x) u = \frac{1-\eta}{4} \partial_x u - \frac{1+\eta}{8} (\partial_y - \partial_x) u, \tag{4.72}$$

$$\partial_\eta \tilde{u} = \frac{1-\xi}{4} \partial_x u + \frac{3-\xi}{8} (\partial_y - \partial_x) u = \frac{1-\xi}{4} \partial_y u + \frac{1+\xi}{8} (\partial_y - \partial_x) u. \tag{4.73}$$

Thus, we have

$$\partial_\xi^r \tilde{u} = \sum_{k=0}^{r} (-1)^k \binom{r}{k} \left(\frac{1+\eta}{8}\right)^k \left(\frac{1-\eta}{4}\right)^{r-k} \partial_x^{r-k} (\partial_y - \partial_x)^k u, \tag{4.74}$$

and

$$\begin{aligned} \|(1-\xi^2)^{(r-1)/2} \partial_\xi^r \tilde{u}\|_\square^2 &= \iint_\square |\partial_\xi^r \tilde{u}|^2 (1-\xi^2)^{r-1} \,\mathrm{d}\xi \,\mathrm{d}\eta \\ &\le c \sum_{k=0}^{r} \iint_\triangle |\partial_x^{r-k} (\partial_y - \partial_x)^k u|^2 \frac{Q(\xi,\eta;r,k)}{J} \,\mathrm{d}x \,\mathrm{d}y, \end{aligned}$$

where

$$Q(\xi,\eta;r,k) = \left(\frac{1+\eta}{8}\right)^{2k} \left(\frac{1-\eta}{4}\right)^{2r-2k} (1-\xi^2)^{r-1}.$$

One verifies readily from (4.1) that

$$\frac{1}{4}(1-\xi)(1-\eta) = 1-x-y, \tag{4.75}$$

$$\frac{1}{4}(1+\xi)(1-\eta) + \frac{1}{8}(1+\xi)(1+\eta) = x, \tag{4.76}$$

$$\frac{1}{4}(1-\xi)(1+\eta) + \frac{1}{8}(1+\xi)(1+\eta) = y. \tag{4.77}$$

Therefore, by (4.75)–(4.77), we derive that, for $1 < r < k$,

$$\begin{aligned} Q(\xi,\eta;r,k) &= \frac{1}{2^k} \left[(1+\xi)^k \left(\frac{1+\eta}{8}\right)^k\right] \left[(1-\xi)^k \left(\frac{1+\eta}{4}\right)^k\right] \\ &\quad \times \left(\frac{(1+\xi)(1-\eta)}{4}\right)^{r-k-1} \left(\frac{(1-\xi)(1-\eta)}{4}\right)^{r-k-1} \frac{(1-\eta)^2}{16} \\ &\le c x^k y^k x^{r-k-1} (1-x-y)^{r-k-1} J^2 \le c \varpi^{r-1,k,r-k-1} J, \end{aligned}$$

where we used the fact: $1-\eta \le 2-\xi-\eta = 16J$, and denoted by $\varpi^{\alpha,\beta,\gamma} = x^\alpha y^\beta (1-x-y)^\gamma$. Similarly, for $1 < r = k$,

$$\begin{aligned} Q(\xi,\eta;r,k) &= \frac{1}{2^r} \left(\frac{(1+\xi)(1+\eta)}{8}\right)^{r-1} \left(\frac{(1-\xi)(1+\eta)}{4}\right)^{r-2} \left(\frac{(1+\eta)}{4}\right)^2 (1-\xi) \\ &\le c x^{r-1} y^{r-2} J \le c \varpi^{r-1,r-2,0} J, \end{aligned}$$

where we used $1-\xi \le 2-\xi-\eta = 16J$. Consequently, we obtain for $r \ge 2$,

$$\begin{aligned}
&\|(1-\xi^2)^{(r-1)/2}\partial_\xi^r \tilde{u}\|_\square \\
&\le c\left(\sum_{k=0}^{r-1} \|\partial_x^{r-k}(\partial_y-\partial_x)^k u\|^2_{\varpi^{r-1,k,r-k-1},\triangle} + \|(\partial_y-\partial_x)^r u\|^2_{\varpi^{r-1,r-2,0},\triangle}\right)^{\frac{1}{2}} \\
&\le c(|u|_{r-1,\triangle} + |u|_{r,\triangle}).
\end{aligned} \tag{4.78}$$

By swapping $x \leftrightarrow y$ and $\xi \leftrightarrow \eta$, we get that for $r \ge 2$,

$$\|(1-\eta^2)^{(r-1)/2}\partial_\eta^r \tilde{u}\|_\square \le c(|u|_{r-1,\triangle} + |u|_{r,\triangle}). \tag{4.79}$$

We now turn to deal with the term $\|(1-\xi^2)^{(r-2)/2}\partial_\eta\partial_\xi^{r-1}\tilde{u}\|_\square$. By (4.73)–(4.74),

$$\begin{aligned}
&\partial_\eta\partial_\xi^{r-1}\tilde{u} \\
&= \partial_\eta\left[\sum_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\left(\frac{1+\eta}{8}\right)^k\left(\frac{1-\eta}{4}\right)^{r-k-1}\partial_x^{r-k-1}(\partial_y-\partial_x)^k u\right] \\
&= \sum_{k=0}^{r} W_1^{r,k}(\xi,\eta)\partial_x^{r-k}(\partial_y-\partial_x)^k u + \sum_{k=0}^{r-1} W_2^{r,k}(\xi,\eta)\partial_x^{r-k-1}(\partial_y-\partial_x)^k u,
\end{aligned} \tag{4.80}$$

where $W_1^{r,k}$ and $W_2^{r,k}$ are polynomials of $\xi$ and $\eta$. Thus, we have

$$\begin{aligned}
\|(1-\xi^2)^{(r-2)/2}\partial_\eta\partial_\xi^{r-1}\tilde{u}\|^2_\square \le\ & c\sum_{k=0}^{r}\iint_\triangle |\partial_x^{r-k}(\partial_y-\partial_x)^k u|^2\frac{(1-\xi^2)^{r-2}}{J}\,\mathrm{d}x\,\mathrm{d}y \\
&+ c\sum_{k=0}^{r-1}\iint_\triangle |\partial_x^{r-k-1}(\partial_y-\partial_x)^k u|^2\frac{(1-\xi^2)^{r-2}}{J}\,\mathrm{d}x\,\mathrm{d}y.
\end{aligned}$$

This implies that, for $r \ge 3$,

$$\|(1-\xi^2)^{(r-2)/2}\partial_\eta\partial_\xi^{r-1}\tilde{u}\|_\square \le c(|u|_{r-1,\triangle} + |u|_{r,\triangle}). \tag{4.81}$$

For $r = 2$, we obtain from a direct calculation that

$$\|\partial_\xi\partial_\eta\tilde{u}\|_\square \le |u|_{2,\triangle} + \frac{1}{256}\|(\partial_y-\partial_x)^2 u\|_{J^{-1},\triangle} + \frac{1}{64}\|\nabla\cdot u\|_{J^{-1},\triangle}. \tag{4.82}$$

A combination of (4.78)–(4.79) and (4.81)–(4.82) leads to the desired result. ∎

**Remark 4.6.** *Like* (4.78)*, we could obtain sharper estimates with semi-norms in the upper bound of* (4.70) *featured with the Jacobi-type weight functions* $\varpi^{\alpha,\beta,\gamma}$.

*Notice that, for* $r = 2$*, the semi-norms are weighted with* $J^{-1}$*, as we can not factor out* $1-\xi$ *or* $1-\eta$ *from* $W_1^{r,k}$ *and* $W_2^{r,k}$ *in* (4.80) *to eliminate* $J^{-1}$*. However, we point out that the value of* $\iint_{\triangle} J^{-1}\,\mathrm{d}x\,\mathrm{d}y$ *is finite.*

## 4.4 Numerical results and remarks

In this section, we just provide some numerical results to demonstrate the high accuracy of the proposed algorithm for model elliptic problems on $\triangle$. We also intend to compare it with the standard tensor-product spectral approximations on rectangles to assess the performance of our approach.

Consider the elliptic equation:

$$-\Delta u+\gamma u = f \quad \text{in } \triangle; \quad u|_{\Gamma_1} = 0; \quad \frac{\partial u}{\partial \nu}\Big|_{\Gamma_2} = g, \tag{4.83}$$

where the constant $\gamma \geq 0$, $\Gamma_1$ is the edges $x = 0$ and $y = 0$, $\Gamma_2$ is the hypotenuse of $\triangle$, and $\nu$ is the unit vector outer normal to $\Gamma_2$.

### 4.4.1 The scheme and its convergence

A weak formulation of (4.83) is to find $u \in H^1_{\Gamma_1}(\triangle) := \{u \in H^1(\triangle) : u|_{\Gamma_1} = 0\}$ such that

$$\mathcal{B}(u,v) := (\nabla u, \nabla v)_{\triangle} + \gamma\,(u,v)_{\triangle} = (f,v)_{\triangle} + \gamma\,\langle g, v\rangle_{\Gamma_2}, \quad \forall\, v \in H^1_{\Gamma_1}(\triangle), \tag{4.84}$$

where $\langle\cdot,\cdot\rangle_{\Gamma_2}$ is the inner product of $L^2(\Gamma_2)$. It follows from a standard argument that if $f \in L^2(\triangle)$ and $g \in L^2(\Gamma_2)$, the problem (4.84) admits a unique solution in $H^1(\triangle)\,\Gamma_1$.

The spectral-Galerkin approximation of (4.84) is to find $u_N \in Y_N^{\Gamma_1}(\triangle) := Y_N(\triangle) \cap H^1_{\Gamma_1}(\triangle)$ such that for any $v_N \in Y_N^{\Gamma_1}(\triangle)$,

$$\mathcal{B}_N(u_N, v_N) := (\nabla u_N, \nabla v_N)_{\triangle} + \gamma\,(u_N, v_N)_{\triangle} = (\mathbb{I}_N\, f, v_N)_{\triangle} + \langle g, v_N\rangle_{N,\Gamma_2}, \tag{4.85}$$

where $\mathbb{I}_N$ is the interpolation operator as defined in (4.55), and the discrete inner product $\langle g, v_N\rangle_{N,\Gamma_2}$ can be defined on the quadrature rule:

$$\begin{aligned}\int_{\Gamma_2} g\,\mathrm{d}\gamma &= \frac{\sqrt{2}}{2}\left[\int_{-1}^{1}\tilde{g}(\xi,1)\,\mathrm{d}\xi - \int_{-1}^{1}\tilde{g}(1,\eta)\,\mathrm{d}\eta\right]\\ &\sim \frac{1}{\sqrt{2}}\left[\sum_{j=0}^{N}(\tilde{g}(\zeta_j,1)-\tilde{g}(1,\zeta_j))\omega_j\right],\end{aligned} \tag{4.86}$$

where $\{\zeta_j,\omega_j\}$ are the LGL interpolation points and weights, as before. More precisely, we define

$$\langle g, v_N\rangle_{N,\Gamma_2} = \frac{1}{\sqrt{2}}\sum_{j=0}^{N}\tilde{g}(\zeta_j,1)\tilde{v}_N(\zeta_j,1)\omega_j - \frac{1}{\sqrt{2}}\sum_{j=0}^{N}\tilde{g}(1,\zeta_j)\tilde{v}_N(1,\zeta_j)\omega_j, \tag{4.87}$$

where $\tilde{g} = g\circ T$ and $\tilde{v}_N = v_N\circ T$, as in (4.9).

**Remark 4.7.** *Here, we purposely impose the Neumann boundary condition on the hypotenuse of $\triangle$, so that the basis functions associated with this "singular" edge are involved in the computation.*

*We reiterate that a distinct difference with the scheme in* [82, eq. (25)] *lies in that the consistency condition* (4.13) *is not needed to be built in the approximation space, which significantly facilitates the implementation. Note that the approaches based on Duffy's transform also need to modify the basis functions to meet the corresponding consistency condition* (see e.g., [103, 24]).

To analyze the convergence of (4.85), it is essential to study the approximability of the orthogonal projection $\Pi_N^{1,\Gamma_1} : H^1_{\Gamma_1}(\triangle)\to Y_N^{\Gamma_1}(\triangle)$, such that

$$\Big(\nabla(\Pi_N^{1,\Gamma_1}u-u),\nabla\phi\Big)_{\triangle} = 0,\quad \forall\,\phi\in Y_N^{\Gamma_1}(\triangle).$$

Following the lines of the proof of Thm. 4.3, we find that (4.64) holds with $\Pi_N^{1,\Gamma_1}$ and $H^1_{\Gamma_1}(\triangle)$ in place of $\Pi_N^{1,0}$ and $H^1_0(\triangle)$, respectively.

Another ingredient for the analysis is to estimate the error between the continuous and discrete inner products on $\Gamma_2$. Using [101, Lem. 4.8] leads to

$$\begin{aligned}\Big|\langle g, v_N\rangle_{N,\Gamma_2} - \langle g, v_N\rangle_{\Gamma_2}\Big| \le cN^{-t}\big(&\|(1-\xi^2)^{(t-1)/2}\partial_\xi^t\tilde{g}(\cdot,1)\|_I\|\tilde{v}_N(\cdot,1)\|_I\\ &+\|(1-\eta^2)^{(t-1)/2}\partial_\eta^t\tilde{g}(1,\cdot)\|_I\|\tilde{v}_N(1,\cdot)\|_I\big).\end{aligned}$$

Then we obtain, from (4.72)–(4.73) and a derivation similar to the proof of Thm. 4.4, the following estimate:

$$\begin{aligned}\left|\langle g, v_N\rangle_{N,\Gamma_2} - \langle g, v_N\rangle_{\Gamma_2}\right| &\le cN^{-t}\|(xy)^{(t-1)/2}(\partial_y - \partial_x)^t g\|_{\Gamma_2}\|v_N\|_{\Gamma_2} \\ &\le cN^{-t}|g|_{t,\Gamma_2}\|v_N\|_{\Gamma_2}, \quad t \ge 1.\end{aligned} \tag{4.88}$$

With the above preparations, we derive the convergence result for the scheme (4.85) from Thm. 4.3–4.4, the estimate (4.88) and a standard argument for error estimate of spectral approximation of elliptic problems.

**Theorem 4.5.** *Let $u$ and $u_N$ be the solutions of (4.84) and (4.85), respectively. If $u \in H^1_{\Gamma_1}(\triangle) \cap H^r(\triangle)$, $f \in H^s(\triangle)$ and $g \in H^t(\Gamma_2)$ with $r \ge 1$, $s \ge 2$ and $t \ge 1$, then we have*

$$\|u - u_N\|_{\mu,\triangle} \le c(N^{\mu-r}|u|_{r,\triangle} + N^{-s}B_s(f) + N^{-t}|g|_{t,\Gamma_2}),$$

*where $\mu = 0, 1$, $B_s(f)$ is defined in (4.71), and $c$ is a positive constant independent of $N$, $u$, $f$ and $g$.*

### 4.4.2 Numerical results

We first intend to show the typical spectral accuracy of the proposed method, so we particularly test it on (4.83) (with $\gamma = 1$) with the exact solution:

$$u(x, y) = e^{x+y-1}\sin\left(3xy\left(y - \frac{\sqrt{3}x}{2} + \frac{\sqrt{3}}{4}\right)\right), \quad \forall\,(x, y) \in \triangle. \tag{4.89}$$

For comparison, we also consider the standard tensor polynomial approximation of (4.83) on a square $S = (0, 1/\sqrt{2})^2$ (note: it has the same area as $\triangle$) under a similar setting, i.e., Neumann data on two edges $x = 1/\sqrt{2}$ and $y = 1/\sqrt{2}$, and homogeneous Dirichet data on the other two edges. We take the exact solution: for all $(x, y) \in S$,

$$u(x, y) = \exp\left(-\left(\frac{1}{\sqrt{2}} - x\right)\left(\frac{1}{\sqrt{2}} - y\right)\right)\sin\left(3xy\left(y - \frac{\sqrt{3}x}{2} + \frac{\sqrt{3}}{4}\right)\right). \tag{4.90}$$

In Fig. 4.5, we plot the numerical errors of two methods, from which we observe that they share a very similar convergence behavior and the errors decay like $O(e^{-cN})$. For a fixed $N$, the accuracy of approximation on $S$ seems to be slightly better than expected. We refer to [20, Fig. 2.17] for a similar comparison of

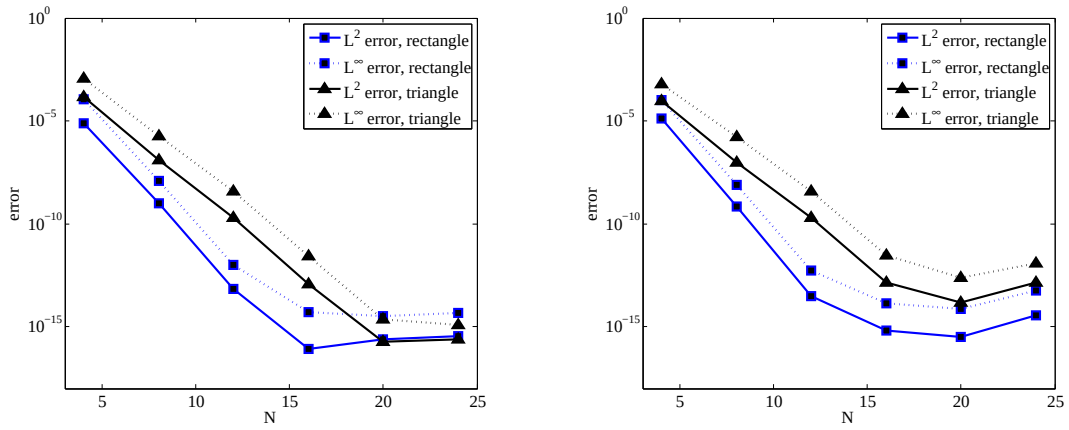



Figure 4.5: Numerical errors of (4.85) vs. tensorial polynomial approximation on the square $S$. Left: $L^2$- and $L^\infty$-errors using modal basis. Right: $L^2$- and $L^\infty$-errors using nodal basis.

polynomial approximations on triangles [44] and rectangles. Indeed, the accuracy is comparable to existing means in [74, 103, 82].

In the second test, we choose the exact solution of (4.83) with finite regularity:

$$u(x,y) = (1-x-y)^{\frac{5}{2}}(e^{xy}-1), \quad \forall\,(x,y) \in \triangle, \tag{4.91}$$

which belongs to $H^{3-\epsilon}(\triangle)$ (for small $\epsilon > 0$). The counterpart on the square $S$ takes the form:

$$u(x,y) = \left(\frac{1}{\sqrt{2}} - x\right)^{\frac{5}{2}} \left(\frac{1}{\sqrt{2}} - y\right)^{\frac{5}{2}} (e^{xy}-1), \quad \forall\,(x,y) \in S. \tag{4.92}$$

We depict in Fig. 4.6 the numerical errors of two approaches in log-log scale, where the slopes of the lines are all roughly $-3$, as predicted by theoretical results (cf. Thm. 4.5).

Finally, we compare our new approach with the method in [82] (where the explicit consistency condition (4.13) was built in the approximation space). One

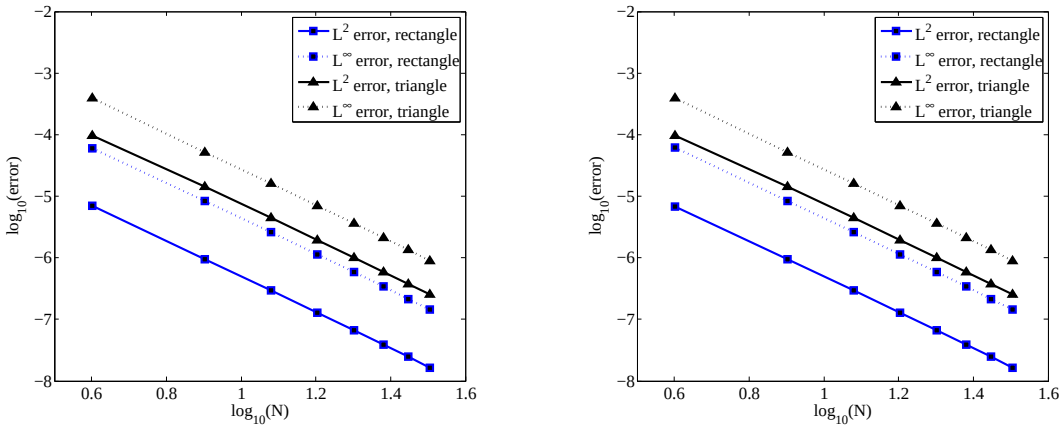



Figure 4.6: Numerical errors of (4.85) vs. tensorial polynomial approximation on the square $S$ with finite-regualrity solutions. Left: $L^2$- and $L^\infty$-errors using modal basis. Right: $L^2$- and $L^\infty$-errors using nodal basis.

Table 4.1: Comparison between the approach in [82] and the new method.

| $N$ | Approach in [82] | | Approach in Sec. 4.4 | |
|---|---|---|---|---|
| | $L^2$-error | $L^\infty$-error | $L^2$-error | $L^\infty$-error |
| 15 | 2.866e−06 | 1.018e−05 | 2.349e−06 | 8.281e−06 |
| 30 | 3.410e−07 | 1.203e−06 | 3.087e−07 | 1.091e−06 |
| 45 | 9.940e−08 | 3.513e−07 | 9.299e−08 | 3.283e−07 |

can see from Table 4.1 that both approaches enjoy a similar convergence behavior. We reiterate that the new method does not require to modify the basis function, so with a pre-computation of the stiffness matrix, the triangular element can be treated as efficiently as the quadrilateral element.

## 4.5 Summary

We have developed a TSEM using the rectangle-triangle map (4.1)–(4.2) that is able to use mapped LGL points on the reference square in a more desirable distribution than that of Duffy's transform (4.4)–(4.5), and a modal or nodal

basis on each triangular element, producing optimal $L^2$- and $H^1$-estimates for projection and interpolation errors, efficiently treating each triangular element computationally as a rectangular element.

To handle the logarithmically-singular weighted inner product that arises from the stiffness matrix of variational formulations of elliptic problems we wish to solve using this TSEM, we use an efficient, stable and accurate spectral quadrature that analytically removes the singularity. By composition with an affine transformation, we generalized the transform to arbitrary triangles. We have also demonstrated that the scheme produces numerical results improving on the current methods.

Chapter 5

# An Unstructured TSEM with DG Implementation

We detailed the implementation and analysis of a spectral-element method on a triangle based on a new rectangle-triangle mapping in Ch. 4, and showed that the singularity induced by the map can be removed analytically. We would like to utilize this TSEM as the basis of a SEM on arbitrary domains: this chapter outlines a preliminary extension, illustrating the TSEM on an arbitrary polygonal domain.

For SEM, since a rectangular element can use a tensorial basis, it is more convenient to decompose the domain over a tensorial grid, such that the interior elements are rectangular, and only the elements along the boundary of the domain can be triangular. In this case, any of the finite-element methods—namely, continuous Galerkin methods, mixed element methods and discontinuous Galerkin methods—can be employed, as the singular point induced by the map can be placed on the edge of triangular elements that lie on the boundary of the domain. To differentiate the methods, and to highlight the difficulties induced by the new TSEM, for this chapter, we outline an unstructured TSEM: that is, a TSEM on an unstructured triangulation of the domain.

Extending element methods to unstructured meshes has the primary concern

of how to handle information on common (interior) edges (and faces, for higher dimensions) of neighboring elements of the mesh, often called *coupling*. When the elements are handled by the new TSEM, continuity of functions between elements—such as enforced by continuous Galerkin methods on the numerical solution, or enforced on the normal of the dual variable by mixed element methods (see e.g., [94, 3])—poses a difficult requirement: if the common edge between two triangular elements is mapped to two edges of the reference square $\square$ by (4.1) on one element, enforced continuity on that edge computationally constrains that element. Thus, we employ discontinuous Galerkin (DG) methods for our new unstructured TSEM.

DG methods, like mixed element methods, introduce a dual variable $\vec{q}_h$ that can be treated as independent of the primal variable $u_h$, and the values of both primal and dual variable on the boundary of the element, $\tilde{u}_h$ and $\tilde{q}_h$, are derived from $u_h$ and $\vec{q}_h$ and each other, thereby allowing the solutions $u_h$ and $\vec{q}_h$ to be discontinuous on the common edges, by instead constraining $\tilde{u}_h$ and $\tilde{q}_h$, with the coupling sometimes weakly enforced by penalty constraints. By this, DG methods are inherently easier to implement on more complex triangulations, such as meshes with nonconformal edges and hanging nodes, which characterizes the singular point induced by (4.1), or those on the edges of the collocation-based TSEM in [54]: computational constraints induced by the geometry only affect computations on $\tilde{u}_h$ and $\tilde{q}_h$. The monographs [96, 40, 32] provide a survey of the selections of $\tilde{u}_h$ and $\tilde{q}_h$ for the variety of DG methods, as well as applications for which these methods are employed. For elliptic problems, see [4] and references therein.

For ease and efficiency of computation, we implement the hybridized local discontinuous Galerkin method (LDG-H) [22, 31, 30], for the following reasons:

- LDG-H, among hybridized DG methods, has been extensively studied.
- LDG-H defines both $\lambda = \tilde{u}_h$ and the value of $\tilde{q}_h \cdot \vec{n}_K$ as single-valued along each element edge.

- LDG-H uses the same element function space for $u_h$ and each component of $\vec{q}_h$, which simplifies per-element precomputation.
- LDG-H has a small linear system to solve for the global constraints $\lambda$, typical of hybridized DG methods.

We refer to the details in [77], which uses Duffy's transform (4.4)–(4.5), and implement LDG-H using the new TSEM on the elements. In particular, handling the singular point on the TSEM is of importance to the implementation.

The rest of the chapter is outlined as follows: We use DG formulation on a model elliptic problem in Sec. 5.1. We apply the LDG-H scheme to this formulation in Sec. 5.2. We perform per-element computations using the new TSEM in Sec. 5.3. The global system in the hybrid variable is computed in Sec. 5.4. Finally, we present some numerical results to illustrate the efficacy of the unstructured TSEM in Sec. 5.5.

## 5.1 DG formulation

Consider the model problem

$$-\Delta u + u = f, \quad \text{in } \Omega; \quad u = 0 \quad \text{on } \partial\Omega, \tag{5.1}$$

where $\Omega$ is a polygonal domain, as in [77, Sec. 6].

We employ the DG method to solve (5.1). The *flux formulation* of (5.1) introduces an auxillary flux variable $\vec{q} = \nabla u$ such that (5.1) is reformulated in mixed form: find $(u, \vec{q})$ such that

$$-\nabla \cdot \vec{q} + u = f \quad \text{in } \Omega; \quad \vec{q} - \nabla u = 0 \quad \text{in } \Omega; \quad u = 0 \quad \text{on } \partial\Omega. \tag{5.2}$$

We now introduce the general setting for the DG formulation [40]. Let $\mathcal{T}_h$ be a triangulation of $\Omega$ and let $K \in \mathcal{T}_h$ be a non-overlapping element within the triangulation such that, if $K_i \neq K_j$, then $K_i \cap K_j = \emptyset$, for $1 \leq i, j \leq |\mathcal{T}_h|$. Let $\partial K$ denote the boundary of $K$ and $\partial K^i$ denote an individual edge of $K$, $i = 1, 2, 3$.

We then denote by $\Gamma$ the set of edges on the boundaries $\partial K$ of all the elements $K \in \mathcal{T}_h$. $\mathcal{T}_h$ is herein assumed to be *conformal* (vertices of any element cannot be in the interior of an edge of another element) with parameter $h = \max_{K\in\mathcal{T}_h} h_K$, where $h_K = \mathrm{diam}K$. We say that $e^\ell \in \Gamma$ is an interior edge of $\mathcal{T}_h$ if there are two elements $K_i, K_j \in \mathcal{T}_h$ such that $e^\ell = \partial K_i \cap \partial K_j$ and the length of $e^\ell$ is nonzero. For consistency, we use subscripts to indicate element indices, e.g. $K_i \in \mathcal{T}_h$, and superscripts to indicate edges, e.g. $e^\ell \in \Gamma$.

Let $\sigma$ be the index map $\sigma(i,j) = \ell$ if and only if $\partial K_i^j = e^\ell$. Note that, for $1 \le \ell \le |\Gamma|$, there is one pair $(i,j)$ such that $\sigma(i,j) = \ell$ if $e^\ell \subset \partial\Omega$; otherwise, there are two such pairs.

Our choice of element solver, the new TSEM, influences the definition of the finite-dimensional spaces associated with $\mathcal{T}_h$. Recall, from Ch. 4, Rem. 4.4 indicates that the methods of the previous chapter can be applied to an arbitrary triangle $K$ and, similar to (4.29), we have

$$Y_N(K) = \mathbb{Q}_N(\square) \circ T_K^{-1} = (\mathbb{P}_N(I))^2 \circ T_K^{-1}, \tag{5.3}$$

where $T_K$ is given by (4.50), noting which edge of $K$ receives the singular point—in practice, we choose the longest edge of $K$. Then,

$$\begin{aligned}
V_h &:= \{v \in L^2(\Omega) \ : \ v|_{K_i} \in Y_{N_i}(K_i) \text{ for } 1 \le i \le |\mathcal{T}_h|\},\\
\vec{\Sigma}_h &:= \{\vec{\tau} \in [L^2(\Omega)]^2 \ : \ \vec{\tau}|_{K_i} \in [Y_{N_i}(K_i)]^2 \text{ for } 1 \le i \le |\mathcal{T}_h|\},\\
\mathcal{M}_h^\circ &:= \{\mu \in L^2(\Gamma) \ : \ \mu|_{e^\ell} \in \mathbb{P}_{N^\ell}(e^\ell) \text{ for } 1 \le \ell \le |\Gamma|,\ \mu = 0 \text{ on } \partial\Omega\},
\end{aligned}$$

where $N_i$, $1 \le i \le |\mathcal{T}_h|$ and $N^\ell$, $1 \le \ell \le |\Gamma|$ are given orders for the respective elements $K_i$ and edges $e^\ell$.

With these definitions, we formulate the problem of the DG method: find

$(u_h, \vec{q}_h) \in V_h \times \vec{\Sigma}_h$ such that for all $(v, \vec{w}) \in V_h \times \vec{\Sigma}_h$,

$$\begin{aligned}
&\sum_{i=1}^{|\mathcal{T}_h|} \iint_{K_i} (\nabla v \cdot \vec{q}_h)\, \mathrm{d}\vec{x} - \sum_{i=1}^{|\mathcal{T}_h|} \int_{\partial K_i} v(\vec{n}_i \cdot \tilde{q}_h)\, \mathrm{d}s + \sum_{i=1}^{|\mathcal{T}_h|} \iint_{K_i} u_h v\, \mathrm{d}\vec{x} = \sum_{i=1}^{|\mathcal{T}_h|} \iint_{K_i} f v\, \mathrm{d}\vec{x}, \\
&\sum_{i=1}^{|\mathcal{T}_h|} \iint_{K_i} (\vec{w} \cdot \vec{q}_h)\, \mathrm{d}\vec{x} = -\sum_{i=1}^{|\mathcal{T}_h|} \iint_{K_i} (\nabla \cdot \vec{w}) u_h\, \mathrm{d}\vec{x} + \sum_{i=1}^{|\mathcal{T}_h|} \int_{\partial K_i} (\vec{w} \cdot \vec{n}_i) \tilde{u}_h\, \mathrm{d}s,
\end{aligned} \tag{5.4}$$

where $\vec{n}_i$ denotes the unit normal vector to $\partial K_i$, $1 \leq i \leq |\mathcal{T}_h|$, and $(\tilde{u}_h, \tilde{q}_h)$ correspond to the values of $(u_h, \vec{q}_h)$ on $\Gamma$ and are defined, per element, in terms of $u_h$ and $\vec{q}_h$ by the DG method used.

## 5.2 LDG-H scheme

For the LDG-H scheme, we use the decomposition of the domain to solve (5.2) on each element $K \in \mathcal{T}_h$, with the aim of using the new TSEM on $K$.

First, assume that $\lambda$, defined by

$$\lambda = \tilde{u}_h \in \mathcal{M}_h^\circ, \tag{5.5}$$

is given. For $1 \leq i \leq |\mathcal{T}_h|$, if we denote by $(u_i, \vec{q}_i) = (u_h, \vec{q}_h)|_{K_i} \in Y_{N_i}(K_i) \times [Y_{N_i}(K_i)]^2$, then by (5.4), for all $(v, \vec{w}) \in Y_{N_i}(K_i) \times [Y_{N_i}(K_i)]^2$,

$$\begin{aligned}
&\iint_{K_i} (\nabla v \cdot \vec{q}_i)\, \mathrm{d}\vec{x} - \int_{\partial K_i} v(\vec{n}_i \cdot \tilde{q}_i)\, \mathrm{d}s + \iint_{K_i} u_i v\, \mathrm{d}\vec{x} = \iint_{K_i} f v\, \mathrm{d}\vec{x}, \\
&\iint_{K_i} (\vec{w} \cdot \vec{q}_i)\, \mathrm{d}\vec{x} = -\iint_{K_i} (\nabla \cdot \vec{w}) u_i\, \mathrm{d}\vec{x} + \int_{\partial K_i} (\vec{w} \cdot \vec{n}_i) \lambda\, \mathrm{d}s,
\end{aligned} \tag{5.6}$$

which follows from enforcing (5.4) element-wise and where, by definition of the LDG-H scheme,

$$\tilde{q}_i = \vec{q}_i - \tau(u_i - \lambda)\vec{n}_i \text{ on } \partial K_i, \tag{5.7}$$

for some positive function $\tau \in \mathcal{M}_h^\circ$ of order $O(1)$. Here, $\tau$ is piecewise constant on $\Gamma$, and its value is usually given on an edge of an element $\partial K_i^j$ as $\tau_i^j$. This

definition of $(\tilde{u}_h = \lambda, \tilde{q}_h)$ contrasts with how general DG schemes define the traces as functions of $u_h$ and $\vec{q}_h$.

It remains to determine $\lambda$, which acts as a Lagrange multiplier for the following continuity condition (cf. [30]): for every $\mu \in \mathcal{M}_h^\circ$,

$$\sum_{i=1}^{|\mathcal{T}_h|} \int_{\partial K_i} \mu(\tilde{q}_i \cdot \vec{n}_i)\,\mathrm{d}s = 0. \tag{5.8}$$

Thus, by (5.7) and (5.8), for $e^\ell = \partial K_i^j = \partial K_{i'}^{j'}$, $i \neq i'$, for every $\mu \in \mathcal{M}_h^\circ$ with support only on $e^\ell$,

$$\int_{e^\ell} \mu(\vec{q}_i \cdot \vec{n}_i + \vec{q}_{i'} \cdot \vec{n}_{i'})\,\mathrm{d}s + (\tau_i^j + \tau_{i'}^{j'}) \int_{e^\ell} \mu\lambda\,\mathrm{d}s - \int_{e^\ell} \mu(\tau_i^j u_i + \tau_{i'}^{j'} u_{i'})\,\mathrm{d}s = 0. \tag{5.9}$$

The equations (5.6) are called the *local solvers* on $K_i$, while the equation (5.9) is called the *transmission condition* on $e^\ell$. Collectively, for $1 \leq i \leq |\mathcal{T}_h|$ and $1 \leq \ell \leq |\Gamma|$, (5.6) and (5.9) describe the LDG-H scheme, wherein the solution $(u_h, \vec{q}_h, \lambda) \in V_h \times \vec{\Sigma}_h \times \mathcal{M}_h^\circ$ is to be determined.

The next two sections detail the matrix implementation of (5.6) and (5.9) for computation.

## 5.3 Local implementation with new TSEM

In this section, we derive all the computations that perform the numerical integrations in (5.6) and (5.9). Let $\{\zeta_j, \omega_j\}_{j=0}^N$ be the LGL points and weights, i.e. $\zeta_j$ are roots of $(1-\zeta^2)P_N'(\zeta)$, and $\omega_j$ are given by (2.13).

### 5.3.1 Element integrals

First, to handle integrals on the triangles $K$, as in (5.6): On each triangular subdomain $K \in \mathcal{T}_h$, we use the TSEM in Ch. 4 which transforms $K$ to $\square$ and employs modified tensorial functions to perform accurate numerical integration by tensorial quadrature, as in Rem. 4.4.

Given that $K$ has vertices $\vec{p}_1 = (x_1, y_1)$, $\vec{p}_2 = (x_2, y_2)$ and $\vec{p}_3 = (x_3, y_3)$ in counterclockwise order, the points $(\xi, \eta) \in \square$ are mapped to the points of $K$ by the one-to-one onto transform $T_K : \square \to K$, (4.50). Note

$$T_K(-1,-1) = \vec{p}_1,\ T_K(1,-1) = \vec{p}_2,\ T_K(-1,1) = \vec{p}_3, T_K(1,1) = \frac{1}{2}(\vec{p}_2 + \vec{p}_3).$$

Let $\vec{d} = (x - x_1, y - y_1)$, $\vec{d}_1 = (x_3 - x_2, y_3 - y_2)$, $\vec{d}_i = (x_i - x_1, y_i - y_1)$ for $i = 2, 3$, and

$$\begin{aligned}
F &= \vec{d}_2 \cdot \vec{d}_3^{\perp} = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1) \neq 0,\\
\bar{x} &= \vec{d} \cdot \vec{d}_3^{\perp}/F = F^{-1}[(x - x_1)(y_3 - y_1) - (y - y_1)(x_3 - x_1)],\\
\bar{y} &= \vec{d}_2 \cdot \vec{d}^{\perp}/F = F^{-1}[(y - y_1)(x_2 - x_1) - (x - x_1)(y_2 - y_1)],\\
\chi &= (2 - \xi - \eta)/2 = \sqrt{(\bar{x} - \bar{y})^2 + 4(1 - \bar{x} - \bar{y})}.
\end{aligned}$$

Then transform (4.50) sends $\tilde{u}(\xi, \eta) \mapsto u(x, y)$ and carries the tensorial polynomial space $\mathbb{Q}_N(\square)$ one-to-one onto $Y_N(K) = \mathbb{P}_N(K) \oplus \chi\mathbb{P}_{N-1}(K)$.

It can be shown that

$$\nabla u = \chi^{-1}[2(\vec{d}_2 - \vec{d}_3)^{\perp}(\widetilde{\nabla} \cdot \tilde{u}) + (\vec{d}_2 + \vec{d}_3)^{\perp}\widetilde{\nabla}^{\intercal}\tilde{u}], \quad J = \left|\frac{\partial(x, y)}{\partial(\xi, \eta)}\right| = \frac{F\chi}{8},$$

where $\widetilde{\nabla}\cdot$ and $\widetilde{\nabla}^{\intercal}\cdot$ are given by (4.8), so, we have

$$\begin{aligned}
&\iint_K uv \,\mathrm{d}\vec{x} = \frac{F}{8}\iint_{\square} \tilde{u}\tilde{v}\chi \,\mathrm{d}\xi \,\mathrm{d}\eta,\\
&\iint_K (\partial_x u)v \,\mathrm{d}\vec{x} = \frac{y_2 - y_3}{4}\iint_{\square} (\widetilde{\nabla} \cdot \tilde{u})\tilde{v} \,\mathrm{d}\xi \,\mathrm{d}\eta + \frac{y_2 + y_3 - 2y_1}{8}\iint_{\square} (\widetilde{\nabla}^{\intercal}\tilde{u})\tilde{v} \,\mathrm{d}\xi \,\mathrm{d}\eta,\\
&\iint_K (\partial_y u)v \,\mathrm{d}\vec{x} = \frac{x_3 - x_2}{4}\iint_{\square} (\widetilde{\nabla} \cdot \tilde{u})\tilde{v} \,\mathrm{d}\xi \,\mathrm{d}\eta + \frac{2x_1 - x_2 - x_3}{8}\iint_{\square} (\widetilde{\nabla}^{\intercal}\tilde{u})\tilde{v} \,\mathrm{d}\xi \,\mathrm{d}\eta.
\end{aligned}$$

Note here that, if $u, v \in Y_N(K)$, then $\tilde{u}, \tilde{v}, \widetilde{\nabla} \cdot \tilde{u}, \widetilde{\nabla}^{\intercal}\tilde{u} \in \mathbb{Q}_N(\square)$. Thus, the first equation can be handled by Legendre polynomial orthogonality on $\square$, as in (4.49), while the other integrals on $\square$ are on elements of $\mathbb{Q}_{2N-1}(\square)$ on $\square$, and LGL quadrature can be used on the transformed functions, with the expected accuracy: for all $\tilde{p} \in \mathbb{Q}_{N-1}(\square), \tilde{p}' \in \mathbb{Q}_N(\square)$,

$$\iint_{\square} \tilde{p}\tilde{p}' \,\mathrm{d}\xi \,\mathrm{d}\eta = (\tilde{p}, \tilde{p}')_{N,\square} := \sum_{i=0}^{N}\sum_{j=0}^{N} \tilde{p}(\zeta_i, \zeta_j)\tilde{p}'(\zeta_i, \zeta_j)\omega_i\omega_j. \tag{5.10}$$

**Remark 5.1.** *Note that, unlike in Ch. 4, the weight $\chi^{-1}$ never appears in any of the above integrals.*

### 5.3.2 Trace integrals

Next, to handle integrals on edges $e^\ell \in \Gamma$: all segments (edges and intervals) we map to $I$, where we can perform one-dimensional LGL quadrature.

For $\mu, \lambda \in \mathbb{P}_{N^\ell}(e^\ell)$, as in (5.9):

$$\int_{e^\ell} \mu\lambda \,\mathrm{d}s = \frac{|e^\ell|}{2} \left\langle \hat{\mu}, \hat{\lambda} \right\rangle_{N^\ell, I} := \frac{|e^\ell|}{2} \sum_{k=0}^{N^\ell} \hat{\mu}(\zeta_k)\hat{\lambda}(\zeta_k)\omega_k, \tag{5.11}$$

where $\hat{\mu}(\zeta) = \mu(s(x,y))$ and $(x,y) \in e^\ell \mapsto \zeta \in I$.

For edges of elements $K \in \mathcal{T}_h$, the Jacobian is merely a factor $|\vec{d}_i|/2$, for $i \in \{1,2,3\}$. Assign the numbers to the boundary edges of $\partial K$ such that $\partial K^j$ is opposite $\vec{p}_j$. The new TSEM makes us give special consideration for $\partial K^1$: the transformation (4.50) maps this edge to two edges of $\square$, $\xi = 1$ and $\eta = 1$.

Thus, for $u, v \in Y_N(K)$, as in (5.6):

$$\int_{\partial K^1} uv \,\mathrm{d}s = \frac{|\vec{d}_1|}{4} \sum_{k=0}^{N} \tilde{u}(\zeta_k, 1)\tilde{v}(\zeta_k, 1)\omega_k + \frac{|\vec{d}_1|}{4} \sum_{k=0}^{N} \tilde{u}(1, \zeta_k)\tilde{v}(1, \zeta_k)\omega_k,$$

$$\int_{\partial K^2} uv \,\mathrm{d}s = \frac{|\vec{d}_3|}{2} \sum_{k=0}^{N} \tilde{u}(-1, \zeta_k)\tilde{v}(-1, \zeta_k)\omega_k,$$

$$\int_{\partial K^3} uv \,\mathrm{d}s = \frac{|\vec{d}_2|}{2} \sum_{k=0}^{N} \tilde{u}(\zeta_k, -1)\tilde{v}(\zeta_k, -1)\omega_k.$$

What remains is to consider the "nonhomogenous" integrals in (5.9) and the last integral of (5.6):

$$\int_e \mu v \,\mathrm{d}s, \quad \int_e \mu(\vec{w} \cdot \vec{n}) \,\mathrm{d}s,$$

where $\mu \in \mathbb{P}_{N^\ell}(e^\ell)$ and $v, \vec{w} \cdot \vec{n} \in Y_N(K)$. We pause our detailing of the implementation to consider the impact of the *locality* of the LDG-H method.

Hybridized DG methods enjoy a high degree of independence in-between subdomains, cf. (5.6)—values of $u_h$ and $\vec{q}_h$ in the subdomains are decoupled and the only globally-coupled variable is $\lambda$, as described by (5.9).

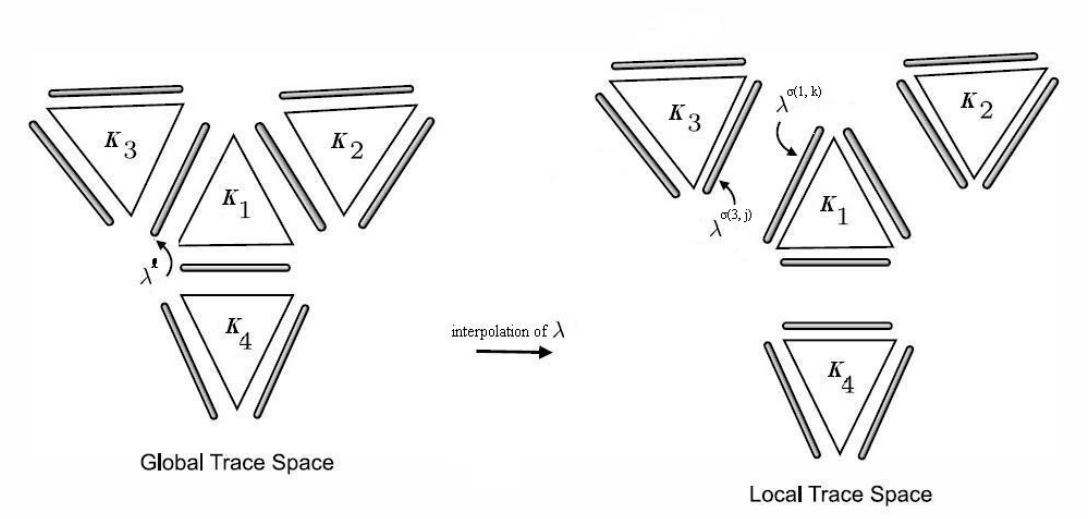



Figure 5.1: [77, Fig. 4] illustrating the locality of (5.6) and (5.9).

[77, Fig. 4] (reproduced and relabeled in Fig. 5.1) illustrates the mechanism involved when solving (5.6) and (5.9): on left, (5.9) allows us to determine the values of $\lambda$ ($\lambda^\ell$ indicated); on right, these $\lambda$ values are copied locally to each element $K_i$ ($\lambda^{\sigma(3,j)}$ and $\lambda^{\sigma(1,k)}$ indicated, where $\ell = \sigma(3,j) = \sigma(1,k)$) to solve for $u_i$ and $\vec{q}_i$ using (5.6).

This feature is a crucial property of hybridized DG methods that is known to help in handling nonconformal triangulations, in particular in the handling of what are called *hanging nodes* (see e.g., [30, Fig. 2.1])..

Therefore, the following considerations must be taken into account when implementing "nonhomogenous" integrals on the trace:

- The orientation of the edge for $\mu$ and $v$ must be consistent: the transform (4.50) determines that, for $\tilde{v}$, the edges that include $\vec{p}_1$ must start there, i.e., $\vec{p} = \vec{p}_1 \mapsto \zeta = -1$, whereas $\partial K^1$ can be set to start from either. That means that the map $(x, y) \mapsto \zeta$ must be consistent for $\tilde{v}|_{e^\ell}$ and $\hat{\mu}$.

- $N$ and $N^\ell$ are possibly not equal: we upsample and let $\hat{N} = \max\{N, N^\ell\}$.

Returning to the implementation, we consider the integral on the edge $e^\ell =$

$\partial K^1$, starting $e^\ell$ on $\vec{p}_i$: this is the edge where the hanging node of the element is located. We rely on $e' \subset e^\ell$ implying $\mu|_{e'} \in \mathbb{P}_{N^\ell}(e')$—this is a treatment of the hanging node induced by the transformation (4.50), which takes advantage of the fact that this node is always found in the middle of $e^\ell$, and that the triangulation is conformal. Thus,

$$\int_{\partial K^1} \mu v \,\mathrm{d}s = \frac{|\vec{d}_1|}{4} \sum_{k=0}^{\hat{N}} \left[ \tilde{v}(\zeta_k, 1)\hat{\mu}\left((-1)^i \frac{1-\zeta_k}{2}\right) + \tilde{v}(1, \zeta_k)\hat{\mu}\left((-1)^i \frac{\zeta_k - 1}{2}\right)\right] \omega_k.$$

As computed, the hanging node is handled locally, i.e. the local copy of $\mu$ used on $K$ is interpolated to determine the value of $\hat{\mu}$ on $\xi = 1$ and $\eta = 1$ on $\square$, to minimize the size of the global system (cf. [30, p. 1333], for treatment of hanging nodes in hybridized schemes).

On the other edges on $\partial K$:

$$\int_{\partial K^2} \mu v \,\mathrm{d}s = \frac{|\vec{d}_3|}{2} \sum_{k=0}^{\hat{N}} \tilde{v}(-1, \zeta_k)\hat{\mu}(\zeta_k)\omega_k,$$
$$\int_{\partial K^3} \mu v \,\mathrm{d}s = \frac{|\vec{d}_2|}{2} \sum_{k=0}^{\hat{N}} \tilde{v}(\zeta_k, -1)\hat{\mu}(\zeta_k)\omega_k.$$

## 5.4 Global system

In this section, we collect the computations in the previous section into linear systems: we follow the development in [77], choosing to use modal basis functions $\{\hat{\phi}_k\}$ (4.31) on $I$.

Note here that

$$\mathrm{span}\{\hat{\phi}_k\}_{k=0}^{N} = \mathbb{P}_N(I), \quad P_0(\zeta) = \hat{\phi}_N(\zeta) + \hat{\phi}_0(\zeta), \quad P_1(\zeta) = \hat{\phi}_N(\zeta) - \hat{\phi}_0(\zeta).$$

Also note that $\hat{\phi}_k(-1) = \delta_{0k}$ and $\hat{\phi}_k(1) = \delta_{Nk}$ for $0 \leq k \leq N$.

Consider then functions $\mathrm{span}\{\widetilde{\Phi}_{mn}\}_{m,n=0}^{N} = \mathbb{Q}_N(\square)$ (4.32).

We then define the coefficient vectors associated with $u_h$, $\vec{q}_h$ and $\lambda$: for $K_i \in$

$\mathcal{T}_h$, let $\vec{q}_i(x,y) = (q_{i,1}(x,y), q_{i,2}(x,y))$ and

$$u_i(x,y) = \tilde{u}_i(\xi,\eta) = \sum_{m=0}^{N_i}\sum_{n=0}^{N_i} \hat{u}_{mn,i}\widetilde{\Phi}_{mn}(\xi,\eta),$$
$$q_{i,j}(x,y) = \tilde{q}_{i,j}(\xi,\eta) = \sum_{m=0}^{N_i}\sum_{n=0}^{N_i} \hat{q}_{mn,i,j}\widetilde{\Phi}_{mn}(\xi,\eta),\ j=1,2,$$

yielding

$$\vec{u}_i = \{\hat{u}_{mn,i}\}_{m,n=0}^{N_i}, \quad \vec{q}_{i,j} = \{\hat{q}_{mn,i,j}\}_{m,n=0}^{N_i},\ j=1,2.$$

We likewise define $\vec{f}_i$ from $\mathbb{I}_h f$: for edge $e^\ell \subset \Gamma$, let

$$\lambda^\ell(s(x,y)) = \hat{\lambda}^\ell(\zeta) = \sum_{k=0}^{N^\ell} \hat{\lambda}_k^\ell \hat{\phi}_k(\zeta),$$

yielding

$$\vec{\lambda}^\ell = \{\hat{\lambda}_k^\ell\}_{k=0}^{N^\ell}.$$

At the local level ($K_i$, edges $\partial K_i^j$, $j=1,2,3$ and $e^\ell$), we can construct inner products versus test functions, based on computations in Sec. 5.3, as thus: if $v \in V_h$ and $\mu \in \mathcal{M}_h^\circ$ are test functions,

$$\iint_{K_i} u_h v \,\mathrm{d}\vec{x} = (\vec{v}_i)^t \boldsymbol{M}_i \vec{u}_i, \qquad \int_{\partial K_i} \tau u_h v \,\mathrm{d}s = \sum_{j=1}^{3} \tau_i^j (\vec{v}_i)^t \boldsymbol{E}_i^j \vec{u}_i,$$
$$\iint_{K_i} (\partial_x q_1) v \,\mathrm{d}\vec{x} = (\vec{v}_i)^t \boldsymbol{D}_{i,1} \vec{q}_{i,1}, \quad \int_{\partial K_i} \tau \lambda v \,\mathrm{d}s = \sum_{j=1}^{3} \tau_i^j (\vec{v}_i)^t \boldsymbol{F}_i^{\sigma(i,j)} \vec{\lambda}^{\sigma(i,j)},$$
$$\iint_{K_i} (\partial_y q_2) v \,\mathrm{d}\vec{x} = (\vec{v}_i)^t \boldsymbol{D}_{i,2} \vec{q}_{i,2}, \quad \int_{e^\ell} \lambda \mu \,\mathrm{d}s = (\vec{\mu}^\ell)^t \boldsymbol{G}^\ell \vec{\lambda}^\ell.$$

From the selection of the basis functions, these matrices are all sparse.

**Remark 5.2.** *Considerations made for nonhomogenous trace integrals (orientation, resizing, interpolation from $\lambda^\ell$ to $\lambda^{\sigma(i,j)}$) can be made at the operator level, each representable by matrix operations. In* [77]*, this combination of operations is referred to as "edge spreading", denoted by $\mathcal{A}_{\mathrm{HDG}}$—this is integrated, and only, in the matrix $\boldsymbol{F}_i^{\sigma(i,j)}$.*

Thus, if $\vec{n}_i^j = (n_{i,1}^j, n_{i,2}^j)$ is the unit outer normal vector to $K_i$ on $\partial K_i^j$ and $\tau_i^j$ is the value of $\tau$ on $\partial K_i^j$, $j = 1, 2, 3$, the local solvers (cf. (5.6)) read as:

$$\begin{aligned} \sum_{j=1}^{3} \tau_i^j \boldsymbol{E}_i^j \vec{u}_i + \boldsymbol{M}_i \vec{u}_i - \sum_{m=1}^{2} \boldsymbol{D}_{i,m} \vec{q}_{i,m} &= \boldsymbol{M}_i \vec{f}_i + \sum_{j=1}^{3} \tau_i^j \boldsymbol{F}_i^{\sigma(i,j)} \vec{\lambda}^{\sigma(i,j)}, \\ (\boldsymbol{D}_{i,m})^t \vec{u}_i + \boldsymbol{M}_i \vec{q}_{i,m} &= \sum_{j=1}^{3} n_{i,m}^j \boldsymbol{F}_i^{\sigma(i,j)} \vec{\lambda}^{\sigma(i,j)}, \quad m = 1, 2; \end{aligned} \tag{5.12}$$

and the transmission condition (cf. (5.9)) on $\ell = \sigma(i,j) = \sigma(i',j')$ reads as:

$$\begin{aligned} (\tau_i^j + \tau_{i'}^{j'}) \boldsymbol{G}^\ell \vec{\lambda}^\ell = \tau_i^j (\boldsymbol{F}_i^\ell)^t \vec{u}_i - \sum_{m=1}^{2} n_{i,m}^j (\boldsymbol{F}_i^\ell)^t \vec{q}_{i,m} \\ + \tau_{i'}^{j'} (\boldsymbol{F}_{i'}^\ell)^t \vec{u}_{i'} - \sum_{m=1}^{2} n_{i',m}^{j'} (\boldsymbol{F}_{i'}^\ell)^t \vec{q}_{i',m}. \end{aligned} \tag{5.13}$$

As in [77], it is possible to eliminate $\vec{u}$ and $\vec{q}_m$, $m = 1, 2$, from the global system. Solving (5.12) for $u_i$ gives

$$\vec{u}_i = \boldsymbol{Z}_i^{-1} \left( \boldsymbol{M}_i \vec{f}_i + \sum_{j=1}^{3} \left[ \left( \sum_{m=1}^{2} n_{i,m}^j \boldsymbol{D}_{i,m} \right) \boldsymbol{M}_i^{-1} + \tau_i^j \boldsymbol{I}_i \right] \boldsymbol{F}_i^{\sigma(i,j)} \vec{\lambda}^{\sigma(i,j)} \right), \tag{5.14}$$

where $\boldsymbol{I}_i$ is the identity matrix and

$$\boldsymbol{Z}_i = \sum_{j=1}^{3} \tau_i^j \boldsymbol{E}_i^j + \sum_{m=1}^{2} \boldsymbol{D}_{i,m} \boldsymbol{M}_i^{-1} (\boldsymbol{D}_{i,m})^t + \boldsymbol{M}_i.$$

With

$$\vec{q}_{i,m} = \boldsymbol{M}_i^{-1} \left( \sum_{j=1}^{3} n_{i,m}^j \boldsymbol{F}_i^{\sigma(i,j)} \vec{\lambda}^{\sigma(i,j)} - (\boldsymbol{D}_{i,m})^t \vec{u}_i \right), \quad m = 1, 2, \tag{5.15}$$

(5.13) gives us the equation in $\vec{\lambda}^\ell$ dependent only on $\vec{\lambda}^{\ell'}$ that shares an element with $\vec{\lambda}^\ell$. We then obtain

$$\boldsymbol{K} \vec{\lambda} = \boldsymbol{L}, \tag{5.16}$$

a system from which $\vec{\lambda}$ (and thus $\vec{\lambda}^\ell$ for all edges $e^\ell$—remember that $\lambda|_{\partial\Omega} = 0$) can be determined. From $\vec{\lambda}$, $\vec{u}_i$ can be determined by (5.14) (and $\vec{q}_{i,m}$s can be determined by (5.15), if needed), per element $K_i$.

Obtaining (5.16) requires some inversion of matrices (see (5.14)–(5.15)), and implementation details follow [77] closely.

**Remark 5.3.** *$\boldsymbol{K}$ is a square, block-sparse matrix (blocks are of size $N^\ell \times N^{\ell'}$, $1 \le \ell, \ell' \le |\Gamma|$, only five blocks per block-row of $\boldsymbol{K}$ are nonzero, corresponding to edges that share a triangle with a given edge) and thus is the size of the square of the total degrees of freedom of $\vec{\lambda}$, $\sum_{\ell=1}^{|\Gamma|} N^\ell$.*

*On grids with triangles and quadrilaterals, local solvers for QSEM are more straightforwrd constructions, and edge solvers are the same—the difference lies in the number of nonzero blocks per block row of the global system* (5.16)*, which can vary from six blocks, for edges between quadrilaterals and triangles, to seven blocks, for edges between quadrilaterals.*

**Remark 5.4.** *If available, parallel computation can be used to solve the system, with one processor handling the global system (`GLOB`) and each local solver handled independently (each by a single processor `LOC(K)` for maximum parallelization). One such parallel algorithm is, as follows:*

1. *`GLOB` receives data for the mesh (possibly including $\tau$, $N_i$, $1 \le i \le |\mathcal{T}_h|$ and $N^\ell$, $1 \le \ell \le |\Gamma|$) and for $f$.*

2. *`GLOB` determines indices for elements $K_i$ and edges $e^\ell$, and the indexing function $\sigma$, which takes $O(|\mathcal{T}_h||\Gamma|)$ operations, noting edge orientation.*

3. *For each element $K_i$, `GLOB` gives `LOC(`$K_i$`)` $N_i$, $f_i$ ($O(N_i^2)$ degrees of freedom), $\tau_i^j$, $n_i^j$ and $N^{\sigma(i,j)}$, $1 \le j \le 3$.*

4. *Matrices are constructed in parallel.*

   - `LOC(`$K_i$`)`

     (a) `LOC(`$K_i$`)` *generates $\boldsymbol{M}_i$, $\boldsymbol{D}_{i,m}$, $m = 1, 2$, $\boldsymbol{E}_i^j$, $\boldsymbol{F}_i^{\sigma(i,j)}$, $1 \le j \le 3$.*

     (b) `LOC(`$K_i$`)` *determines $\boldsymbol{Z}_i$, a dense matrix requiring the inversion of $\boldsymbol{M}_i$. Inverting $\boldsymbol{Z}_i$ requires $O(N_i^6)$ operations.*

(c) `LOC(`$K_i$`)` *determines, for* $m = 1, 2$, $1 \le j, j' \le 3$,

$$\begin{aligned}
\boldsymbol{V}_i^j &= \left[ \left( \sum_{m=1}^{2} n_{i,m}^j \boldsymbol{D}_{i,m} \right) \boldsymbol{M}_i^{-1} + \tau_i^j \boldsymbol{I}_i \right] \boldsymbol{F}_i^{\sigma(i,j)}, \\
\boldsymbol{U}_i^j &= \boldsymbol{Z}_i^{-1} \boldsymbol{V}_i^j, \qquad \tilde{f}_i = \boldsymbol{Z}_i^{-1} \boldsymbol{M}_i \vec{f}_i, \qquad \vec{l}_i^{j'} = \left(\boldsymbol{V}_i^{j'}\right)^t \tilde{f}_i, \\
\boldsymbol{Q}_{i,m}^j &= \boldsymbol{M}_i^{-1} \left( n_{i,m}^j \boldsymbol{F}_i^{\sigma(i,j)} - (\boldsymbol{D}_{i,m})^t \boldsymbol{U}_i^j \right), \\
\boldsymbol{K}_i^{j,j'} &= \left(\boldsymbol{F}_i^{\sigma(i,j')}\right)^t \left( \tau_i^{j'} \boldsymbol{U}_i^j - \sum_{m=1}^{2} n_{i,m}^{j'} \boldsymbol{Q}_{i,m}^j \right).
\end{aligned}$$

*Most of the computational cost is concentrated here: operations involving* $\boldsymbol{Z}_i^{-1}$ *require* $O(N_i^6)$ *operations.*

(d) `LOC(`$K_i$`)` *gives* $\boldsymbol{K}_i^{j,j'}$ *and* $\vec{l}_i^{j'}$, $1 \le j, j' \le 3$, *to* `GLOB`.

- `GLOB`

(a) `GLOB` *generates* $\boldsymbol{G}^\ell$ *for every internal edge* $e^\ell$.

(b) `GLOB` *initializes* $\boldsymbol{K}$ *as a block diagonal matrix with* $(\tau_i^j + \tau_{i'}^{j'})\boldsymbol{G}^\ell$*s in its diagonal, where* $\ell = \sigma(i,j) = \sigma(i',j')$, *and* $\boldsymbol{L}$ *as a zero vector.*

(c) *When* $\boldsymbol{K}_i^{j,j'}$ *and* $\vec{l}_i^{j'}$, $1 \le j, j' \le 3$, *are received from* `LOC(`$K_i$`)`, `GLOB` *queues these, adding* $\vec{l}_i^{j'}$ *to the sections of* $\boldsymbol{L}$ *corresponding to* $\sigma(i,j')$ *and subtracting* $\boldsymbol{K}_i^{j,j'}$ *from appropriate sections of* $\boldsymbol{K}$.

5. *When all* `LOC(`$K$`)` *have sent the computed matrices,* `GLOB` *solves* (5.16) *for* $\vec{\lambda}$ *in* $O([\sum_\ell N^\ell]^3)$ *operations.*

6. *For each element* $K_i$, `GLOB` *gives* `LOC(`$K_i$`)` $\vec{\lambda}^{\sigma(i,j)}$, $1 \le j \le 3$.

7. `LOC(`$K_i$`)` *determines*

$$\vec{u}_i = \tilde{f}_i + \sum_{j=1}^{3} \boldsymbol{U}_i^j \vec{\lambda}^{\sigma(i,j)}.$$

*This takes* $O(N_i^4)$ *operations. If needed,* $\vec{q}_{i,m}$, $m = 1, 2$ *are determined.*

8. `LOC(`$K_i$`)` *gives* `GLOB` $\vec{u}_i$ *(and* $\vec{q}_{i,m}$, $m = 1, 2$*) for domain-level processing.*

**Remark 5.5.** *Iterative systems that generate* $\vec{\lambda}$ *from* $\vec{u}$ *and* $\vec{q}_m$*s by the transmission condition* (5.9) *then* $\vec{u}$ *and* $\vec{q}_m$*s from* $\vec{\lambda}$ *by the local solvers* (5.6) *avoid the use of matrix inversions in constructing* (5.16)*. However, these loops do not seem to converge, even when initialized with slight perturbations of either set of exact solutions, for even simple domain decompositions, e.g. two triangles with a common edge.*

## 5.5 Numerical results

In this section, we solve (5.1), with $\Omega = [0,1]^2$, numerically by (5.16) for $\lambda$ then by (5.12) for $(u_i, \vec{q}_i)$, $1 \leq i \leq |\mathcal{T}_h|$, to get $(u_h, \vec{q}_h)$. All errors are computed on per-element ($K \in \mathcal{T}_h$) average, either determining the $L^2$ error $\|u - u_h\|_K$ or the $H^1$ error

$$\sqrt{\|u - u_h\|_K^2 + \|\nabla u - \vec{q}_h\|_K^2}.$$

First, the domain is triangulated into a coarse unstructured mesh, denoted $5 \times 5$, a medium mesh, denoted $15 \times 15$ and a fine mesh, denoted $25 \times 25$. Each mesh is named after the uniform quadrilateral mesh used as a basis of comparison in [77, Sec. 6], and are shown, in that order, in Fig. 5.2.

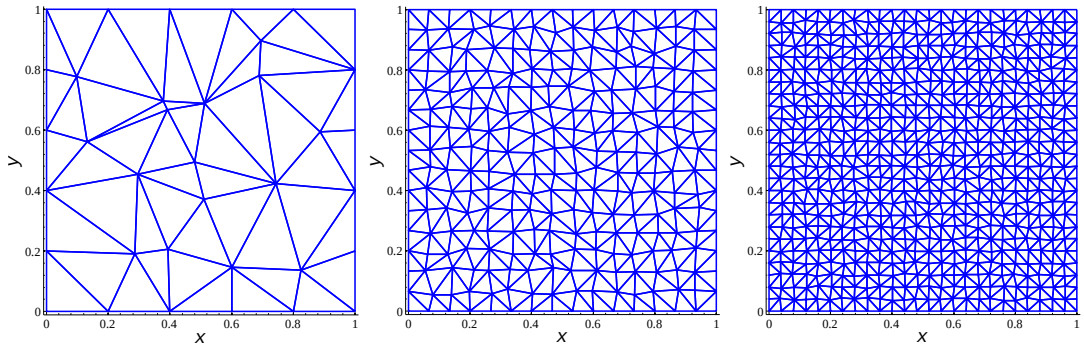



Figure 5.2: Unstructured triangulated meshes.

We solve (5.1) with the highly-oscillating exact solution

$$u(x, y) = \sin(10\pi x)\cos(10\pi y).$$

Results herein are to be compared with the triangular results for $H^1$ error, given in solid lines of [77, Fig. 6(b)] ($\tau = 1000$, to compare between LDG-H and CG), shown in the top row of Fig. 5.3, and the triangular results for $L^2$ error, given in solid lines in [77, Fig. 9(a)] ($\tau = 1$, whose points correspond to meshes $M \times M$, where $M = 5, 10, \ldots, 45$), shown in the bottom row of Fig. 5.3. The new results given here try to mimic those given in [77], with $\tau = 1, 1000$ for $H^1$ error, and $\tau = 1$ and $M = 5, 10, \ldots, 45$ (with $h = 1/M$ and shown in reverse order) for $L^2$ error.

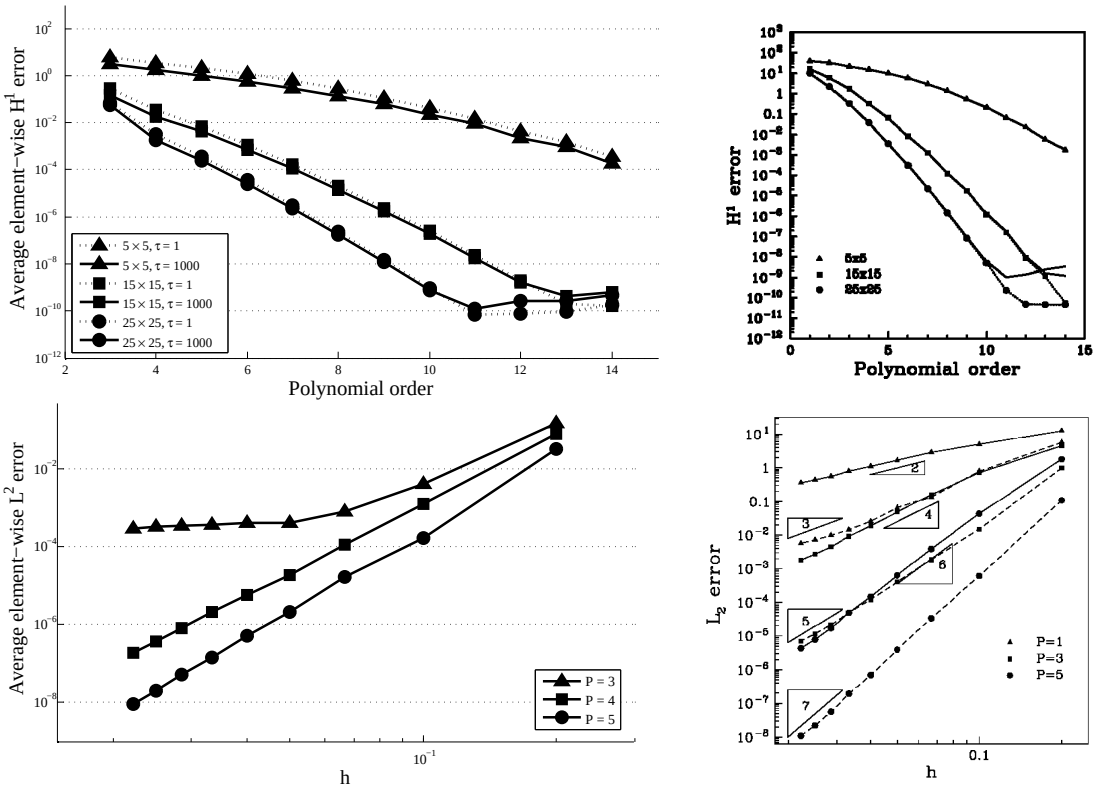



Figure 5.3: Average error per element: top, $H^1$ error; bottom: $L^2$ error; left: new results; right, published results [77].

In general, results generated improve the results in [77] for the triangular case. Indeed, the results align better with the "better case" CG results in the first set, and the "better case" post-processed results in the second set.

Next, the same exact solution as above is used, while the square domain is triangulated into two meshes of varying coarseness, denoted $5 \times 5$ (shown with circles) and $10 \times 10$ (shown with squares), same as above, either maintaining the regular underlying mesh (shown in black) or perturbing slightly on internal vertices (shown in red). The results herein are given with $\tau = 1$ (solid lines) and $\tau = 1000$ (dotted lines), with polynomial order $N = 3, 6, \ldots, 24$, shown in Fig. 5.4.

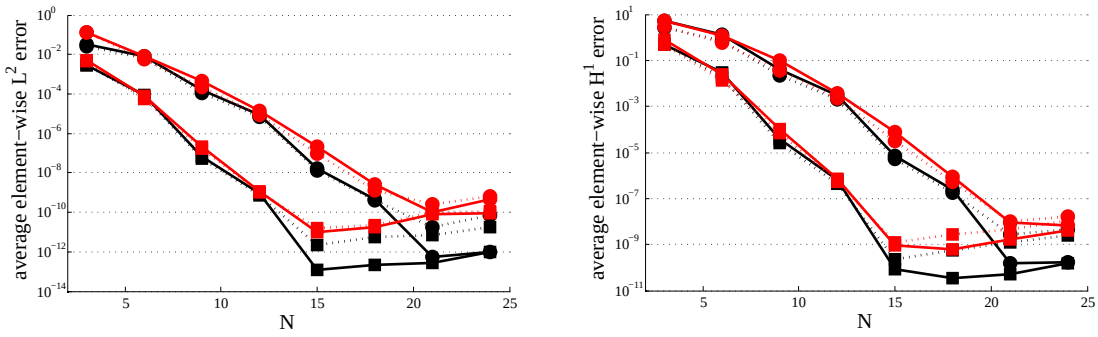



Figure 5.4: Average error per element: left, $L^2$ error; right, $H^1$ error.

Here, it is noticable that the algorithm significantly loses some accuracy on the irregular mesh, compared to the regular mesh, seen here for $N \geq 15$—this is expected, since the regular grid allows elements, their traces and the unit normals to align, even through the rectangle-triangle map, which will not be the case in the irregular grid, a source of additional error. Also, the $L^2$ error seems to minimize to about $10^{-13}$ at best (here at $N = 15$) while the $H^1$ error (which combines the $L^2$ errors on $u$ and $\vec{q}$) seems to minimize to about $10^{-10}$ at best (here at $N \in \{15, 18, 21\}$). These optimal levels are not improved upon by refining the mesh further, though, in those cases, they could be attained at lower values of $N$. Note further that $\tau = 1000$ does not give better results than $\tau = 1$, and this is more noticable when $N \geq 15$, i.e. $\tau = 1000$ on the regular mesh performs only slightly better than either value of $\tau$ on the perturbed mesh.

For the final two tests, the square domain is split into just two right triangles (here, the algorithm is also changed slightly so that one right triangle has its

singular point on the hypotenuse, and the other triangle has its singular point on a leg).

First, with the low-oscillation exact solution with slight highly-oscillating perturbation

$$u(x, y) = \sin(2\pi x)\cos(2\pi y) + \epsilon \cos(20\pi x)\sin(20\pi y),$$

the top row of Fig. 5.5 shows the results for $\epsilon = 0$ shown in black, $\epsilon = 10^{-8}$ shown in blue and $\epsilon = 10^{-4}$ shown in red. $N = 3, 6 \ldots, 21$ again indicates polynomial order.

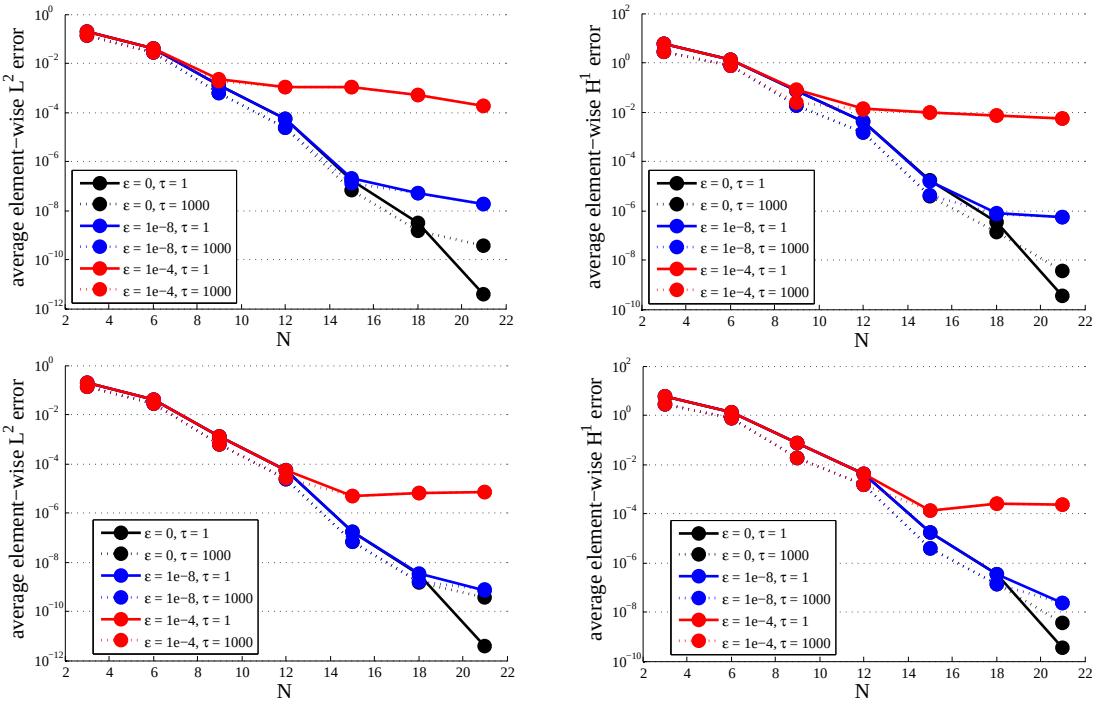



Figure 5.5: Average error per element: left, $L^2$ error; right, $H^1$ error; top, perturbed exact solution; bottom, perturbed $\lambda_h$ values.

Finally, if, instead, the exact solution is unperturbed

$$u(x, y) = \sin(2\pi x)\cos(2\pi y),$$

and the perturbation

$$\epsilon \cos(20\pi x)\sin(20\pi y),$$

on the diagonal is introduced after computing the values on the diagonal ($\lambda_h$) but before applying the local solvers, the bottom row of Fig. 5.5 shows the results for $\epsilon = 0$ shown in black, $\epsilon = 10^{-8}$ shown in blue and $\epsilon = 10^{-4}$ shown in red. Note that the unperturbed graphs are the same (top and bottom).

It is evident that perturbing the exact solution (which similarly perturbs values of $f$ and $g$) has a larger effect on the accuracy than perturbing $\lambda_h$—however, it is evident the perturbation does not significantly affect the computed solutions below a threshhold value for $N$, threshhold values differing between the two perturbations. Also, past their threshhold values, neither perturbation is affected by changing the value of $\tau$, indicating that the perturbations have a larger effect on the accuracy than does the value of $\tau$, but lesser effect than the value of $N$.

## 5.6 Summary

We presented an unstructured TSEM using the rectangle-triangle map (4.1)–(4.2) with LDG-H implementation, efficiently and independently assembling each of the local element solvers, which are used in precomputing blocks of the global system and in solving the local systems based on the global solution.

Notably, the construction of the global system from the local element solvers incorporates the specific nature of the hanging nodes induced by the rectangle-triangle map, maintaining the efficiency of the computation.

We demonstrated that the scheme produces numerical results improving on current methods using Duffy's transform on the local solvers.

Chapter 6

# Conclusion and Future Works

In this chapter, we summarize the contributions of the thesis and explore some relavant topics for further investigation.

## 6.1 Conclusions

In the first part of the thesis, we presented a new method for generating well-conditioned collocation schemes using Birkhoff or Birkhoff-type interpolation. For the schemes generated by the method in the text, the coefficient matrices of the linear systems had condition numbers independent of the number of collocation points used.

- In Ch. 2, the method produced well-conditioned collocation schemes (either by using the new modal-type Birkhoff interpolation basis, where the scheme involves no differentiation matrix, or by optimally preconditioning the collocation schemes using the Lagrange interpolation basis) for second-order BVPs with general boundary conditions on Legendre- and Chebyshev-Gauss-Lobatto points. PSIM were proved to be inverses of PSDM, and construction of the PSIM was done in an efficient and stable manner. We also proved the condition-number property for the Helmholtz equation with

Dirichlet boundary using eigenanalysis of the second-order differentiation matrix on the interior points.

- In Ch. 3, for second-order IVPs on the half-line on Laguerre-Gauss-Radau points, we determined a novel Birkhoff-type interpolation (indicated by the eigenvalues of the second-order differentiation matrix associated with Laguerre functions, which led to a new differentation operator) which, when used with the method, produced a well-conditioned collocation scheme—using the method with the natural Birkhoff interpolation would not produce a well-conditioned collocation scheme. The method generated well-conditioned collocation schemes for IVPs on Gauss-Radau points, higher-order BVPs on Gauss-Lobatto points for space-discretization of stable simulations of time-dependent KdV equations and second-order BVPs on Gegenbauer-Gauss-Lobatto points. We used the PSIM from Ch. 2 and matrix decomposition to determine a two-dimensional collocation scheme to solve a model second-order elliptic PDE on the square.

In the second part of the thesis, we presented a new TSEM based on a recent rectangle-triangle map (4.1)–(4.2), which produces a more desirable distribution of mapped LGL points on the triangle than the Duffy's transform.

- In Ch. 4, we presented new modal and nodal basis functions on the triangle used in the TSEM, and provided optimal $L^2$- and $H^1$-estimates for projection and interpolation errors. No modifications were needed on the modal and nodal basis functions on the reference square to generate the corresponding basis functions on the triangle under the new map. We demonstrated the efficient, stable and accurate computation of mass and stiffness matrices, as the logarithmic singularity (4.13) was analytically removed. Composing (4.1)–(4.2) with an affine transformation led to a similarly efficient TSEM on arbitrary triangles. We also show that the new TSEM on

one triangle compares favorably to the TSEM using the Duffy's transform and the TSEM using the mapping (4.1)–(4.2) from [82].

- In Ch. 5, the new TSEM was integrated into the LDG-H formulation to produce a TSEM on an unstructured mesh. The hanging node produced by (4.1)–(4.2) on each triangular element was efficiently handled by the global system and the local solvers were computed in parallel. We demonstrated through numerical results the accuracy of the resulting scheme, comparing favorably to the LDG-H-TSEM using the Duffy's transform.

## 6.2 Future works

Research for the method in the first part, which produces well-conditioned collocation schemes, three directions are worthy of further investigation.

- Investigate the notion for well-conditioned polynomial-based collocation methods for other situations, e.g., the spline collocation, radial basis functions and some non-polynomial bases. In fact, as shown in *L. L. Wang, J. Zhang, and Z. Zhang, On hp-convergence of PSWFs and a new well-conditioned prolate-collocation scheme, arXiv:1310.3457*, some properties essential for such a construction might be lacking, so new ideas are needed.

- As commented in Rem. 3.4, the extension of the well-conditioned collocation approach to multiple dimensions requires the matrix decomposition techniques, that is, to solve generalized eigen-problems like other tensorial methods. One idea to explore for time-dependent nonlinear problems is to use the alternating direction implicit (ADI) method in time.

- As addressed in Rem. 3.2, it is challenging to obtain the optimal error estimates for the Birkhoff interpolations. However, these results are essential for the error analysis of the new collocation approaches. This will be another direction of our future works.

The new TSEM on unstructured meshes based on the DG formulation in Ch. 5 is worthy of deep investigation. Further development can be taken in the following directions:

- Apply the TSEM to more challenging problems such as the Stokes equations and the Navier-Stokes equations.
- Develop a three-dimensional unstructured tetrahedral TSEM. This has the advantages that the new mapping leads to much better distribution of the points (see Ch. 4), and under DG formulation, the singular Jacobian does not cause trouble (see Ch. 5). However, it becomes much more involved for both implementation and analysis.
- Prove global convergence of the method in Ch. 5. The approach in [31] can be used, but the proof does not follow immediately, due primarily to difficulty in proving the uniqueness of the map from the traces on $K$ to $Y_N(K)$.

# Bibliography


[1] R. A. Adams. *Sobolov Spaces*, volume 65 of *Pure and Applied Mathematics*. Academic Press, New York, 1975.

[2] D. N. Arnold. An interior penalty finite element method with discontinuous elements. *SIAM J. Numer. Anal.*, 19(4):742–760, 1982.

[3] D. N. Arnold. Mixed finite element methods for elliptic problems. *Comput. Meth. Appl. Mech. Engrg*, 82(1-3):281–300, 1990.

[4] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779, 2002.

[5] I. M. Babuška and M. Zlámal. Nonconforming elements in the finite element method with penalty. *SIAM J. Numer. Anal.*, 10(5):863–875, 1973.

[6] G. A. Baker. Finite element methods for elliptic equations using nonconforming elements. *Math. Comp.*, 31(137):45–59, 1977.

[7] G. A. Baker, W. N. Jureidini, and O. A. Karakashian. Piecewise solenoidal vector fields and the Stokes problem. *SIAM J. Numer. Anal.*, 27(6):1466–1485, 1990.

[8] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *J. Comput. Phys.*, 131(2):267–279, 1997.

[9] C. E. Baumann and J. T. Oden. A discontinuous *hp* finite element method for convection-diffusion problems. *Comput. Methods Appl. Mech. Engrg*, 175(3–4):311–341, 1999.

[10] R. Becker, P. Hansbo, and R. Stenberg. A finite element method for domain decomposition with non-matching grids. *Math. Model. Numer. Anal.*, 37(2):209–226, 2003.

[11] C. Bernardi, Y. Maday, and A. T. Patera. Domain decomposition by the mortar element method. In *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*, volume 384 of *NATO ASI Series, Advanced Science Institutes Series C: Mathematical and Physical Sciences*, pages 269–286. Kluwer Academic Publishers, 1993.

[12] B. Bialecki, G. Fairweather, and A. Karageorghis. Matrix decomposition algorithms for elliptic boundary value problems: a survey. *Numer. Algorithms*, 56(2):253–295, 2011.

[13] T. Z. Boulmezaoud and J. M. Urquiza. On the eigenvalues of the spectral second order differentiation operator and application to the boundary observability of the wave equation. *J. Sci. Comput.*, 31(3):307–345, 2007.

[14] J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover Publications Inc., 2001.

[15] J. P. Boyd and F. Yu. Comparing seven spectral methods for interpolation and for solving the Poisson equation in a disk: Zernike polynomials, Logan-Shepp ridge polynomials, Chebyshev-Fourier series, cylindrical Robert functions, Bessel-Fourier expansions, square-to-disk conformal mapping and radial basis functions. *J. Comput. Phys.*, 230(4):1408–1438, 2011.

[16] F. Brezzi, G. Manzini, D. Marini, P. Pietra, and A. Russo. Discontinuous finite elements for diffusion problems. In *Atti del Convegno in onore di F. Brioschi (Milano 1997)*, pages 197–217. Istituto Lombardo, Accademia di Scienze e Lettere, Milan, Italy, 1999.

[17] C. Canuto. High-order methods for PDEs: recent advances and new perspectives. In *ICIAM 07—6th International Congress on Industrial and Applied Mathematics*, pages 57–87. Eur. Math. Soc., Zürich, 2009.

[18] C. Canuto, P. Gervasio, and A. Quarteroni. Finite-element preconditioning of G-NI spectral methods. *SIAM J. Sci. Comput.*, 31(6):4422–4451, 2009/2010.

[19] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods*. Scientific Computation. Springer-Verlag, Berlin, 2006. Fundamentals in single domains.

[20] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral Methods*. Scientific Computation. Springer, Berlin, 2007. Evolution to complex geometries and applications to fluid dynamics.

[21] C. Canuto and A. Quarteroni. Preconditioned minimal residual methods for Chebyshev spectral calculations. *J. Comput. Phys.*, 60(2):315–337, 1985.

[22] P. Castillo, B. Cockburn, I. Perugia, and D. Schötzau. An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. *SIAM J. Numer. Anal.*, 38(5):1676–1706, 2000.

[23] F. Chen and J. Shen. Efficient spectral-Galerkin methods for systems of coupled second-order equations and their applications. *J. Comput. Phys.*, 231(15):5016–5028, 2012.

[24] L. Chen, J. Shen, and C. Xu. A triangular spectral method for the Stokes equations. *Numer. Math.: Theory, Methods Appl.*, 4(2):158–179, 2011.

[25] L. Chen, J. Shen, and C. Xu. A unstructured nodal spectral-element method for the Navier-Stokes equations. *Comm. Comput. Phys.*, 12(1):315–336, 2012.

[26] Q. Chen and I. M. Babuška. Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Comp. Meth. Appl. Math. Eng.*, 128(2):405–417, 1995.

[27] A. Chernov. Optimal convergence estimates for the trace of the polynomial $L^2$-projection operator on a simplex. *Math. Comp.*, 81(278):765–787, 2011.

[28] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, 1978.

[29] C. W. Clenshaw. The numerical solution of linear differential equations in Chebyshev series. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 53, pages 134–149. Cambridge Univ. Press, 1957.

[30] B. Cockburn, J. Gopalakrishnan, and R. Lazarov. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM J. Numer. Anal.*, 47(2):1319–1365, 2009.

[31] B. Cockburn, J. Gopalakrishnan, and F.-J. Sayas. A projection-based error analysis of HDG methods. *Math. Comp.*, 79(271):1351–1367, 2010.

[32] B. Cockburn, G. E. Karniadakis, and C. W. Shu. *Discontinuous Galerkin Methods: Theory, Computation and Applications*, volume 11 of *Lecture Notes in Computational Sciences and Engineering*. Springer-Verlag Berlin Heidelberg, 2000.

[33] B. Cockburn and C. W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.*, 35(6):2440–2463, 1998.

[34] F. A. Costabile and E. Longo. A Birkhoff interpolation problem and application. *Calcolo*, 47(1):49–63, 2010.

[35] E. A. Coutsias, T. Hagstrom, J. S. Hesthaven, and D. Torres. Integration preconditioners for differential operators in spectral $\tau$-methods. In *Proceedings of the Third International Conference on Spectral and High Order Methods, Houston, TX*, pages 21–38, 1996.

[36] E. A. Coutsias, T. Hagstrom, and D. Torres. An efficient spectral method for ordinary differential equations with rational function coefficients. *Math. Comp.*, 65(214):611–635, 1996.

[37] M. O. Deville, P. F. Fischer, and E. H. Mund. *High-order methods for incompressible fluid flow*, volume 9 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2002.

[38] M. O. Deville and E. H. Mund. Chebyshev pseudospectral solution of second-order elliptic equations with finite element preconditioning. *J. Comput. Phys.*, 60(3):517–533, 1985.

[39] M. O. Deville and E. H. Mund. Finite element preconditioning for pseudospectral solutions of elliptic problems. *SIAM J. Sci. Stat. Comput.*, 11(2):311–342, 1990.

[40] D. A. Di Pietro and A. Ern. *Mathematical Aspects of Disontinuous Galerkin Methods*. Springer-Verlag Berlin Heidelberg, 2012.

[41] J. Douglas, Jr. and T. Dupont. *Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods*, volume 58 of *Lecture Notes in Physics*. Springer-Verlag, Berlin, 1976.

[42] T. A. Driscoll. Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations. *J. Comput. Phys.*, 229(17):5980–5998, 2010.

[43] T. A. Driscoll, F. Bornemann, and L. N. Trefethen. The Chebop system for automatic solution of differential equations. *BIT*, 48(4):701–723, 2008.

[44] M. Dubiner. Spectral methods on triangles and other domains. *J. Sci. Comput.*, 6(4):345–390, 1991.

[45] M. G. Duffy. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM J. Numer. Anal.*, 19(6):1260–1262, 1982.

[46] S. E. El-Gendi. Chebyshev solution of differential, integral and integro-differential equations. *Comput. J.*, 12(3):282–287, 1969.

[47] M. E. Elbarbary. Integration preconditioning matrix for ultraspherical pseudospectral operators. *SIAM J. Sci. Comput.*, 28(3):1186–1201, 2006.

[48] K. T. Elgindy and K. A. Smith-Miles. Solving boundary value problems, integral, and integro-differential equations using Gegenbauer integration matrices. *J. Comput. Appl. Math.*, 237(1):307–325, 2013.

[49] A. Ezzirani and A. Guessab. A fast algorithm for Gaussian type quadrature formulae with mixed boundary conditions and some lumped mass spectral approximations. *Math. Comp.*, 68(225):217–248, 1999.

[50] L. Fatone, D. Funaro, and V. Scannavini. Finite-difference preconditioners for superconsistent pseudospectral approximations. *Math. Model. Numer. Anal.*, 41(6):1021–1039, 2007.

[51] L. Fatone, D. Funaro, and G. J. Yoon. A convergence analysis for the superconsistent Chebyshev method. *Appl. Numer. Math.*, 58(1):88–100, 2008.

[52] B. Fornberg. *A Practical Guide to Pseudospectral Methods*. Cambridge University Press, 1996.

[53] D. Funaro. Computing the inverse of the Chebyshev collocation derivative. *SIAM J. Sci. Stat. Comput.*, 9(6):1050–1057, 1988.

[54] D. Funaro. Pseudospectral approximation of a PDE defined on a triangle. *Appl. Math. Comput.*, 42(2):121–138, 1991.

[55] D. Funaro. *Polynomial Approximation of Differential Equations*, volume 8 of *Lecture Notes in Physics New Series m: Monographs*. Springer-Verlag, 1992.

[56] D. Funaro. A superconsistent Chebyshev collocation method for second-order differential operators. *Numer. Algorithms*, 28(1–4):151–157, 2001.

[57] D. Funaro and D. Gottlieb. A new method of imposing boundary conditions in pseudospectral approximations of hyperbolic equations. *Math. Comp.*, 51(184):599–613, 1988.

[58] W. Gautschi. Gauss quadrature routines for two classes of logarithmic weight functions. *Numer. Algorithms*, 55(2–3):265–277, 2010.

[59] F. Ghoreishi and S. M. Hosseini. The Tau method and a new preconditioner. *J. Comput. Appl. Math.*, 163(2):351–379, 2004.

[60] W. J. Gordon and C. A. Hall. Construction of curvilinear co-ordinate systems and applications to mesh generation. *Internat. J. Numer. Methods Engrg.*, 7(4):461–477, 1973.

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

[62] L. Greengard. Spectral integration and two-point boundary value problems. *SIAM J. Numer. Anal.*, 28(4):1071–1080, 1991.

[63] B. Y. Guo. *Spectral Methods and Their Applications*. World Scientific Publishing Co. Inc., River Edge, NJ, 1998.

[64] B. Y. Guo, J. Shen, and L. L. Wang. Optimal spectral-Galerkin methods using generalized Jacobi polynomials. *J. Sci. Comput.*, 27(1–3):305–322, 2006.

[65] B. Y. Guo and L. L. Wang. Error analysis of spectral method on a triangle. *Adv. Comput. Math.*, 26(4):473–496, 2007.

[66] B. Y. Guo and Z. Q. Wang. A collocation method for generalized nonlinear Klein-Gordon equation. *Adv. Comput. Math.*, DOI: 10.1007/s10444-013-9312-5, 2013.

[67] W. Heinrichs. Spectral collocation schemes on the unit disc. *J. Comput. Phys.*, 199(1):55–86, 2004.

[68] B. T. Helenbrook. On the existence of explicit *hp*-finite element methods using Gauss-Lobatto integration on the triangle. *SIAM J. Numer. Anal.*, 47(2):1304–1318, 2009.

[69] J. S. Hesthaven. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM J. Numer. Anal.*, 35(2):655–676, 1998.

[70] J. S. Hesthaven. Integration preconditioning of pseudospectral operators. I. Basic linear operators. *SIAM J. Numer. Anal.*, 35(4):1571–1593, 1998.

[71] J. S. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral Methods for Time-Dependent Problems*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge, 2007.

[72] W. Z. Huang and D. M. Sloan. The pseudospectral method for third-order differential equations. *SIAM J. Numer. Anal.*, 29(6):1626–1647, 1992.

[73] E. A. Hylleraas. Linearization of products of Jacobi polynomials. *Math. Scand.*, 10:189–200, 1962.

[74] G. E. Karniadakis and S. J. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, second edition, 2005.

[75] S. D. Kim and S. V. Parter. Preconditioning Chebyshev spectral collocation method for elliptic partial differential equations. *SIAM J. Numer. Anal.*, 33(6):2375–2400, 1996.

[76] S. D. Kim and S. V. Parter. Preconditioning Chebyshev spectral collocation by finite difference operators. *SIAM J. Numer. Anal.*, 34(3):939–958, 1997.

[77] R. M. Kirby, S. J. Sherwin, and B. Cockburn. To CG or to HDG: a comparative study. *J. Sci. Comput.*, 51(1):183–212, 2012.

[78] T. Koornwinder. Two-variable analogues of the classical orthogonal polynomials. In *Theory and application of special functions (Proc. Advanced Sem., Math. Res. Center, Univ. Wisconsin, Madison, Wis., 1975)*, pages 435–495. Academic Press, New York, 1975.

[79] D. J. Korteweg and G. de Vries. On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *Philos. Mag.*, 39(240):422–443, 1895.

[80] H. Li and J. Shen. Optimal error estimates in Jacobi-weighted Sobolev spaces for polynomial approximations on the triangle. *Math. Comp.*, 79(271):1621–1646, 2010.

[81] H. Li and L. L. Wang. A spectral method on tetrahedra using rational basis functions. *Int. J. Numer. Anal. Model.*, 7(2):330–355, 2010.

[82] Y. Li, L. L. Wang, H. Li, and H. Ma. A new spectral method on triangles. In *Spectral and High Order Methods for Partial Differential Equations:*

*Selected papers from the ICOSAHOM '09 conference, June 22-26, Trondheim, Norway*, volume 76 of *Lecture Notes in Computational Sciences and Engineering*, pages 237–246. Springer, 2011.

[83] P. W. Livermore. Galerkin orthogonal polynomials. *J. Comput. Phys.*, 229(6):2046–2060, 2010.

[84] J. Loncaric. The pseudo-inverse of the derivative operator in polynomial spectral methods. *ICASE Report*, National Aeronautics and Space Administration, Langley Research Center, Hampton, Va., 1997.

[85] G. G. Lorentz, K. Jetter, and S. D. Riemenschneider. *Birkhoff Interpolation*, volume 19 of *Encyclopedia of Mathematics and its Applications*. Addison-Wesley Publishing Co., Reading, Mass., 1983.

[86] R. E. Lynch, J. R. Rice, and D. H. Thomas. Direct solution of partial differential equations by tensor product methods. *Numer. Math.*, 6(1):185–199, 1964.

[87] B. Mihaila and I. Mihaila. Numerical approximations using Chebyshev polynomial expansions: El-Gendi's method revisited. *J. Phys. A*, 35(3):731–746, 2002.

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

[89] S. Olver and A. Townsend. A fast and well-conditioned spectral method. *SIAM Review*, 55(3):462–489, 2013.

[90] E. J. Parkes, Z. Zhu, B. R. Duffy, and H. C. Huang. Sech-polynomial travelling solitary-wave solutions of odd-order generalized KdV equations. *Phys. Lett. A*, 248(2–4):219–224, 1998.

[91] R. Pasquetti and F. Rapetti. Spectral element methods on unstructured meshes: comparisons and recent advances. *J. Sci. Comput.*, 27(1–3):377–387, 2006.

[92] R. Pasquetti and F. Rapetti. Spectral element methods on unstructured meshes: which interpolation points? *Numer. Algorithms*, 55(2–3):349–366, 2010.

[93] A. T. Patera. A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J. Comput. Phys.*, 54(3):468–488, 1984.

[94] P. A. Raviart and J. M. Thomas. A mixed finite element method for 2-nd order elliptic problems. In Ilio Galligani and Enrico Magenes, editors, *Mathematical Aspects of Finite Element Methods*, volume 606 of *Lecture Notes in Mathematics*, pages 292–315. Springer-Verlag, Berlin Heidelberg, 1977.

[95] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, University of California, Los Alamos Scientific Laboratory, Los Alamos, New Mexico, 1973.

[96] B. Rivière. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation.* SIAM, 2008.

[97] T. Rusten, P. S. Vassilevski, and R. Winther. Interior penalty preconditioners for mixed finite element approximations of elliptic problems. *Math. Comp.*, 65(214):447–466, 1996.

[98] C. Schwab. *p- and hp-Finite Element Methods: Theory and Applications in Solid and Fluid Mechanics.* Numerical Mathematics and Scientific Computation. Oxford Science Publications, 1998.

[99] J. Shen. Efficient spectral-Galerkin method I. Direct solvers for second- and fourth-order equations by using Legendre polynomials. *SIAM J. Sci. Comput.*, 15(6):1489–1505, 1994.

[100] J. Shen. A new dual-Petrov-Galerkin method for third and higher odd-order differential equations: Application to the KDV equation. *SIAM J. Numer. Anal.*, 41(5):1595–1619, 2003.

[101] J. Shen, T. Tang, and L. L. Wang. *Spectral Methods: Algorithms, Analysis and Applications*, volume 41 of *Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, 2011.

[102] J. Shen and L. L. Wang. Fourierization of the Legendre-Galerkin method and a new space-time spectral method. *Appl. Numer. Math.*, 57(5–7):710–720, 2007.

[103] J. Shen, L. L. Wang, and H. Li. A triangular spectral element method using fully tensorial rational basis functions. *SIAM J. Numer. Anal.*, 47(3):1619–1650, 2009.

[104] Y. G. Shi. *Theory of Birkhoff Interpolation*. Nova Science Pub Incorporated, 2003.

[105] D. M. Sloan. On the norms of inverses of pseudospectral differentiation matrices. *SIAM J. Numer. Anal.*, 42(1):30–48, 2004.

[106] G. Szegö. *Orthogonal Polynomials*. AMS Colloquium Publications, fourth edition, 1975.

[107] M. A. Taylor, B. A. Wingate, and L. P. Bos. A cardinal function algorithm for computing multivariate quadrature points. *SIAM J. Numer. Anal.*, 45(1):193–205, 2007.

[108] M. A. Taylor, B. A. Wingate, and R. E. Vincent. An algorithm for computing Fekete points in the triangle. *SIAM J. Numer. Anal.*, 38(5):1707–1720, 2000.

[109] L. N. Trefethen. *Spectral Methods in MATLAB*, volume 10 of *Software, Environments, and Tools*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[110] L. N. Trefethen and M. R. Trummer. An instability phenomenon in spectral methods. *SIAM J. Numer. Anal.*, 24(5):1008–1023, 1987.

[111] L. L. Wang and B. Y. Guo. Interpolation approximations based on Gauss-Lobatto-Legendre-Birkhoff quadrature. *J. Approx. Theory*, 161(1):142–173, 2009.

[112] Z. Q. Wang and L. L. Wang. A collocation method with exact imposition of mixed boundary conditions. *J. Sci. Comput.*, 42(2):291–317, 2010.

[113] H. Weber. *Lehrbuch der Algebra*. Erster Band, Braunschweig, 1912.

[114] J. A. C. Weideman and S. C. Reddy. A MATLAB differentiation matrix suite. *ACM TOMS*, 26(4):465–519, 2000.

[115] J. A. C. Weideman and L. N. Trefethen. The eigenvalues of second-order spectral differentiation matrices. *SIAM J. Numer. Anal.*, 25(6):1279–1298, 1988.

[116] B. D. Welfert. On the eigenvalues of second-order pseudospectral differentiation operators. *Comput. Methods Appl. Mech. Engrg.*, 116(1–4):281–292, 1994.

[117] M. F. Wheeler. An elliptic collocation-finite element method with interior penalties. *SIAM J. Numer. Anal.*, 15(1):152–161, 1978.

[118] Z. Xie, L. L. Wang, and X. D. Zhao. On exponential convergence of Gegenbauer interpolation and spectral differentiation. *Math. Comp.*, 82(282):1017–1036, 2013.

[119] Y. Xu. *Common Zeros of Polynomials in Several Variables and Higher Dimensional Quadrature*. Chapman & Hall / CRC, 1994.

[120] Y. Xu. On Gauss-Lobatto integration on the triangle. *SIAM J. Numer. Anal.*, 49(2):541–548, 2011.

[121] A. Zebib. A Chebyshev method for the solution of boundary value problems. *J. Comput. Phys.*, 53(3):443–455, 1984.

[122] Z. M. Zhang. Superconvergence points of polynomial spectral interpolation. *SIAM J. Numer. Anal.*, 50(6):2966–2985, 2012.

# List of Publications

## Paper published in referred Journals

1. M. D. Samson, H. Li, and L. L. Wang. A new triangular spectral element method I: implementation and analysis on a triangle. *Numer. Algorithms*, 64(3):519–547, 2013.

## Paper to appear

2. L. L. Wang, M. D. Samson, and X. D. Zhao. A well-conditioned collocation method using a pseudospectral integration matrix. Accepted to *SIAM J. Sci. Comput.*, 2014.

## Papers in preparation

3. M. D. Samson and L. L. Wang. Well-conditioned collocation methods for unbounded domains. In preparation.

4. M. D. Samson and L. L. Wang. A new TSEM II: implementation and analysis on unstructured meshes based DG. In preparation.

## Online publication

5. M. D. Samson and M. F. Ezerman. Factoring permutation matrices into a product of tridiagonal matrices. arXiV:1107.3467, 2010.