Applications of Special Functions in High Order Finite Element Methods

Von der Fakultät für Mathematik und Physik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
Dr. rer. nat.

genehmigte Dissertation von

M.Sc. Tim Sebastian Haubold

2023

Referent: Prof. Dr. Sven Beuchler, Gottfried Wilhelm Leibniz Universität Hannover
Korreferentin: Assoc. Prof. Dr. Veronika Pillwein, Johannes Kepler Universität Linz
Korreferent: Prof. Dr. Christoph Lehrenfeld, Georg-August-Universität Göttingen
Tag der Promotion: 06.10.2023

Acknowledgements

Firstly, I am grateful to Prof. Dr. Sven Beuchler for providing many fruitful discussions and giving me the opportunity to venture forth in many new mathematical directions. I also want to thank him for always finding time to hear me out. A lot of this work would not be possible without Prof. Dr. Veronika Pillwein, who inspired many new insights and gave me the opportunity to work with the `Mathematica` software `RISCErgoSum`.
I am also grateful to the `Ngsolve`-team, which influenced many of my design decisions. Especially Prof. Dr. Joachim Schöberl, who also provided the very interesting topic of biorthogonal functions.
I had a big pleasure of working with Robin, who not only proofread the biggest parts of this thesis, but also made every day at the Ifam a fun experience. Thanks go also to Sebastian for many fruitful discussions and comments.
Many thanks go also to my Mother, my Brother and my friends, who supported me in many different ways. Lastly, I am grateful to my girlfriend Maris, who helped to keep me motivated and gave me heavy support even in daring pandemic times.
Thank you all.

Abstract

In this thesis, we optimize different parts of high order finite element methods by application of special functions and symbolic computation. In high order finite element methods, orthogonal polynomials like the Jacobi polynomials are deeply rooted. A broad classical theory of these polynomials is known. Moreover, with modern computer algebra software we can extend this knowledge even further. Here, we apply this knowledge and software for different special functions to derive new recursive relations of local matrix entries. This massively optimizes the assembly time of local high order finite element matrices. Furthermore, the introduced algorithm is in optimal complexity. Moreover, we derive new high order dual functions, which result in fast interpolation operators. Lastly, efficient recursive algorithms for hanging node constraint matrices provided by this new dual functions are given.
Keywords: High-order FEM, Special Functions, Symbolic Computation

Contents

List of Figures

Nomenclature

Special Functions

$\Gamma(\cdot)$	Gamma function
$B(\cdot,\cdot)$	Beta function
$L_n(x)$	Legendre polynomial
$\widehat{L}_n(x)$	Integrated Legendre polynomial
$P_n^{(\alpha,\beta)}(x)$	Jacobi polynomial
$\widehat{P}_n^{\alpha}(x)$	Integrated Jacobi polynomial
$(a)_n$	Pochhammer symbol
${}_2F_1\left(\begin{matrix}a,b\\c\end{matrix};x\right)$	Gaussian hypergeometric series
${}_pF_q\left(\begin{matrix}a_1,\dots,a_p\\c_1,\dots,c_q\end{matrix};x\right)$	Generalized hypergeometric series
$F^{p_1,p_2,p_3}_{q_1,q_2,q_3}$	Kampé de Fériet series

Functional Analysis

Ω	Polygonal Lipschitz domain
∇	Gradient operator
$\mathbb{V}$	Modified gradient operator
curl	Scalar curl operator in $\mathbb{R}^2$ or vectorial curl operator in $\mathbb{R}^3$
Curl	Vectorial curl operator in $\mathbb{R}^2$
div	Divergence operator
$L^2(\Omega)$	Space of square integrable functions
$H^1(\Omega)$, $H(\mathrm{curl},\Omega)$, $H(\mathrm{div},\Omega)$	Sobolev spaces on Ω
$\|\cdot\|_{H^1}$, $\|\cdot\|_{H(\mathrm{div})}$, $\|\cdot\|_{H(\mathrm{curl})}$	Induced norms
$\langle\cdot,\cdot\rangle$	L^2 inner product
$a(\cdot,\cdot)$	Arbitrary (elliptic) bilinear form

Finite Element Method

$\mathcal{N}$	Nédélec function
$\mathcal{RT}$	Raviart-Thomas function
u	H^1-conforming basis function
v	$H(\text{curl})$-conforming basis function
w	$H(\text{div})$-conforming basis function
$\square, \triangle$	Reference square or triangle
$\blacksquare, \blacktriangle$	Reference hexahedron or tetrahedron

1. Introduction

The finite element method is broadly used in computer aided engineering, and has a number of very different applications. It is very well known that applications like Navier-Stokes equations, Maxwell's equations or even elasticity can be handled. The classical mathematical formulation is given by Ciarlet [Cia78]. Today, other classical mathematical literature can be found, see e.g. [BS07, Bra13, GR94, EG21]. First work in the late 70s [SM78] indicated that with a rising number of polynomial functions and their polynomial degree the convergence of finite element methods greatly improves, if the solution is sufficiently locally smooth. Since then, high order finite element methods have been established as one of the standard algorithms. Classical high order literature includes [SB91, Sch98, Mel02, Dem06, DKP+08, vSD04, KS13]. Free scientific software which implements high order methods are for example `NGSolve`[Sch14], `deal.II`[ABF+22, ABD+21], `MFEM`[mfe, AAB+21], `Nektar++`[CMC+15] or `Hermes` [VŠZ07].

It has been shown that high order finite element functions based on orthogonal polynomials exhibit beneficial properties. E.g. they can be assembled very efficiently, see [Ors80]. Furthermore, they have better element condition numbers compared to nodal basis functions, e.g. based on Lagrangian polynomials, see [Ors80] and also [BGP89, MP96].

Therefore, the first part of this thesis focuses on special functions and orthogonal polynomials. The main role throughout the thesis will be played by the Jacobi polynomial $P_n^{(\alpha,\beta)}(x)$. In many classical books, e.g. [Sze67, Rai71, Chi11, AAR99] it is shown that these polynomials fulfil certain relations to other Jacobi polynomials, so-called contiguous relations or recurrence relations. Classically, those relations are derived with help of a series representation of the Jacobi polynomials. To be precise, the Jacobi polynomials can be represented as a so-called *hypergeometric series*, which will play a crucial role in section 2.2. From a theoretical point of view, products or integrals of Jacobi polynomials have always been a highly researched topic. For example, the topic of linearization of the product of two Jacobi polynomials has been discussed in [Hyl62, Gas70, Rah81]. Here the integral over three Jacobi polynomials is needed. More generally speaking, the integrals over orthogonal polynomials were needed in different settings throughout the last two centuries, which lead to the first tabulated systems by Bateman[1], see [BE55]. Similar tabulated projects include e.g. [MOS66] and [AS65]. Today, such knowledge can be found online on sites like [DLM] or [WF].

The first systematic techniques for the analysis of hypergeometric series, including the derivation of contiguous relations by application of differential operators, can be found in the work of Gauss [GSB+66]. Generalizations to more general hypergeometric series have been done either by a brute force approach, for examples of such an approach see the monograph by Rainville [Rai71], or systematical, see e.g. Wilson [Wil78]. Today, such relations can be computed by computer algebra. Sister Celine Fasenmyer [Fas47, Fas49] provided the first algorithmic groundwork for the computation of such recursive relations. Later on, Wilf and Zeilberger [WZ90] rigorously proved the correctness of this algorithm and extended it further. Today such algorithms are a standard

[1]Finished after his death under the editorship of Arthur Erdélyi

part of computer algebra system like `Mathematica` [Inc], `Maple` [Map] or `Sage` [S+YY].
We will derive new recursive relations throughout this thesis with the help of the `Mathematica` packages `Guess` by Manuel Kauers [Kau09] and `HolonomicFunctions` by Christoph Koutschan [Kou10]. While `Guess` tries to guess relations between the provided (exact) data, `HolonomicFunctions` uses a telescoping technique to find annihilating formulations to prove such relations. Similar guessing algorithms can be found e.g. in [SZ94, Kra01]. The holonomic systems approach goes back to Zeilberger [Zei90a], but see also [Sab93, Chy00].
Furthermore, we applied `Mathematica` heavily in the background to double-check more tedious computations.
After the rigorous introduction to (multivariate) hypergeometric series, we will change our focus to high order finite element methods. In general, we have a problem of the form:

Problem 1.0.1
Find $u \in \mathbb{V}$ such that:

$$a(u, v) = F(v) \quad \text{for all } v \in \mathbb{V}$$

Here $\mathbb{V}$ is some Sobolev space, $a(\cdot, \cdot)$ some bilinear form and $F(\cdot)$ some linear form. Usual choices for $\mathbb{V}$ include H^1, $H(\text{curl})$ and $H(\text{div})$. We will mainly focus on the first two, but similar results can be extended to the case of $H(\text{div})$. The methodology of describing those spaces with relations to each other is given by the so-called De-Rham complex, which we will discuss in chapter 3, see also [Mon03]. For the discretization, we need to apply different discrete De-Rham complexes and different specially chosen basis. The idea of this special chosen elements goes back to Raviart and Thomas [RT77] in the case of divergence conforming elements in $2D$, and in $3D$ to Nédélec [Né86], see also [GR86]. The origin of the curl conforming basis goes back to Nédélec. Those functions were derived in two steps or two families of functions, see [Né80, Né86]. They are often also called Whitney elements, compare [Whi57]. Similar constructions were also done by Mur [Mur92].
In a high order setting it has been shown in [BGP89] and [MP96], that the choice of basis functions has a big influence on the condition number of the resulting system matrix. We follow the classical ansatz of dividing all element functions into vertex, edge, face and interior functions, see e.g. [SB91].
A high order L^2-ansatz on triangles goes back to Dubiner[2][Dub91] and were generalized by Karniadakis and Sherwin [SK95] to an H^1 setting in $2D$ and $3D$, see also [KS13]. One of the popular choices for the high order discretizations of $H(\text{curl})$ and $H(\text{div})$, was given by Sabine Zaglmayr in her dissertation [Zag06]. We will base our construction on this and the follow-up works, [BPZ13b, BPZ12]. Some of this work also goes back to [AC01]. Alternative sets of high order curl or divergence functions can be found in [Dem06, DKP+08, vSD04, FKDN15].
Moreover, in this thesis we will derive a new assembly routine for high order element matrices, by application of the results in section 2.2. With this we present an efficient element matrix assembly algorithm based on orthogonal polynomials in optimal complexity $\mathcal{O}(p^d)$. The state-of-the-art algorithm, the so-called *sum- factorization*, [Ors80, MGS99, EM05], achieves only the complexity $\mathcal{O}(p^{d+1})$. On the other hand sum factorization techniques are applicable to all reasonable material functions. Experiments in [VSK10] seems to indicate, that the break even point of sum-factorization techniques and classical numerical quadratures for certain differential opera-

[2] Although the orthogonality of those polynomials were already proven in [Pro57] unknown to Dubiner, see also [Koo75, DX14].

tors can be rather large for high order finite element method[3].
Alternatives to the orthogonal polynomials, like *Bernstein*-polynomials, can lead to assembly routines in optimal complexity but at the price of higher condition numbers, see [AAD11]. Moreover, some interesting techniques like recursive computation of inverse mass matrix blocks exist for Bernstein polynomials, see [Kir17].
At last, we derive biorthogonal functions for H^1 and $H(\text{curl})$ functions. Those biorthogonal functions will speed up interpolation operations, e.g. for starting values in time dependent systems. But they could also be applied as projection operators in a preconditioner setting. Here again, knowledge of Jacobi polynomials takes a major role in the computation. An interesting application is given in section 6.2, where we compute constraint matrices for hanging nodes. Although the results have been published in [HPB22], we give a small extension to $3D$.
The topic of efficient numerical solver and/or preconditioners will not be discussed here, since it is a whole topic in itself. Summarizing the thesis is organized as follows:

Chapter 2 The first part 2.1 is an introduction to special functions like $\Gamma(\cdot)$ and $B(\cdot,\cdot)$ functions, as well as hypergeometric series. The main actors of this thesis, the Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$, are introduced. A small first result is given at the end of section 2.1.
After introduction of a multivariate generalization, one of the main results, theorem 2.2.9, is introduced and proven in section 2.2. These introductions and results have been published in [BHP23].
Some new results regarding the linearization of Jacobi polynomial are given in section 2.3.

Chapter 3 This chapter only recalls the basics of finite element methods. In more detail, we introduce the variational framework. Moreover, since we are not only interested in the Sobolev space H^1 but in the spaces $H(\text{curl})$ and $H(\text{div})$ as well, all spaces and their commuting diagram, the De-Rham complex, is introduced. Finally, we end this chapter with a small collection of classical results for the convergence rates of higher order finite elements.

Chapter 4 In this chapter, a collection of basis functions for different variational spaces and reference elements is laid out. We will focus on the basis functions in $H^1, H(\text{curl})$ and $H(\text{div})$ for quadrilateral, hexahedral, triangle and tetrahedron. Except for some small adjustments in edge or face functions, new results are first presented in section 4.2.8. Here we modified existing basis functions for $H(\text{curl})$, such that those functions are in the so-called Nédélec space of *first kind*.

Chapter 5 One of the main topics of this thesis is discussed here: The assembly of high order finite elements in optimal complexity. The standard method and the so called *sum-factorization* are introduced. We applied theorem 2.2.9 to derive new algorithms for the computation of elements matrices in optimal complexity $\mathcal{O}(p^d)$. Two small numerical experiments are presented at the end of the chapter.

Chapter 6 In the first part of this chapter we derive dual or biorthogonal functions for functions in H^1 and $H(\text{curl})$ in $2D$ and in $3D$. Furthermore we scale those functions, such that the resulting diagonal matrix is constant. In section 6.2 we derive effective recursive relations for the computation of constraint matrix for irregular meshes with hanging nodes. Section 6.2

[3]Which is no problem for pure spectral methods.

has been published in [HPB22], except for some $3D$ generalization at the end of the chapter. Throughout this section, we will readily apply different already introduced methods.

Appendix A The appendix contains additional results, with regard to multivariate hypergeometric functions, sparsity results and some further properties of Jacobi polynomials.

The main contribution in this thesis will be as follows:

Recursive relations of multivariate hypergeometric series New relations will thoroughly be investigated and proven.

Nédélec functions of first type We will modify the basis functions from [BPZ13b], such that those functions fit back in the setting of Nédélec's first and second family.

Assembly routine We will introduce an element assembly routine in optimal complexity

Biorthogonal functions We will derive biorthogonal functions in $2D$ and $3D$ for H^1 and $H(\text{curl})$ on different reference elements.

Efficient computation of hanging nodes We will consider efficient algorithms for the computation of constraint matrices for hanging nodes, and also for hanging edges and hanging faces as well.

2. Special functions and orthogonal polynomials

The aim of this chapter is the thorough introduction of special functions and orthogonal polynomials. Our main interest lies in the properties of Jacobi polynomials and of the integrals of their products. Since some classical orthogonal polynomials like the Jacobi polynomials can be written as so-called hypergeometric series, we will start with the introduction of these series.
Section 2.1 and section 2.2 are an extended version of previously published results, see [BHP23].

2.1. Hypergeometric series

The gamma function $\Gamma(\cdot)$ is defined by Eulers integral, i.e.

$$\Gamma(z) := \int_0^\infty e^{-t} t^{z-1}\,\mathrm{d}t, \quad \mathrm{Re}(z) > 0.$$

It is analytic if z is finite, and its roots are 0 and $-k$ for $k \in \mathbb{N}$. If $z \in \mathbb{N}$ we can write $\Gamma(z) = (z-1)!$. Using the following form of the Euler integral, we define the beta function as

$$B(x, y) := \int_0^1 t^{x-1}(1-t)^{y-1}\,\mathrm{d}t. \tag{2.1}$$

It is connected to the Gamma function by the following relation

$$B(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}.$$

Furthermore, we define the Pochhammer symbol (or rising factorial) as

$$(a)_n = a(a+1)(a+2)\dots(a+n-1) = \frac{\Gamma(a+n)}{\Gamma(a)} = \frac{(a+n-1)!}{(a-1)!},$$

with $(a)_0 = 1$. We denote the (Gaussian) *hypergeometric function* for arbitrary parameters a, b, c where $c \notin -\mathbb{N}_0$ by

$${}_2F_1\begin{pmatrix} a, b \\ c \end{pmatrix}; z\Big) = \sum_{n=0}^\infty \frac{(a)_n (b)_n z^n}{(c)_n n!}. \tag{2.2}$$

Such a series converges on the circle $|z| < 1$ and it is terminating for all z if either a or b is zero or a negative integer. On $|z| = 1$ the condition $\mathrm{Re}(c - a - b) > 0$ is sufficient for absolute convergence. Often the question arises if such a series is summable for $z = 1$. Example of such a summation theorem was given by Gauss, see theorem 2.1.1. Proofs can be found e.g. in [Chi11, AAR99] or [Rai71].

Theorem 2.1.1 (Gaussian summation theorem)
If $\mathrm{Re}(c-a-b) > 0$ and if $c \notin -\mathbb{N}_0$, then

$$ {}_2F_1\left(\begin{matrix} a, b \\ c \end{matrix}; z\right) := \frac{\Gamma(c)\Gamma(c-a-b)}{\Gamma(c-a)\Gamma(c-b)}. $$

Contiguous relations

Since (2.2) contains three parameters a, b and c we can try to connect the series to its neighbours, where parameters are raised or lowered by 1. Those relations are called *contiguous relations*. We will follow [Rai71] closely and present the basic concept of Gauß' proof for such relations, since it will be applied to more complex functions later on. For the sake of brevity we introduce the following notation

$$ \begin{aligned} F &:= {}_2F_1\left(\begin{matrix} a, b \\ c \end{matrix}; z\right) \\ F(a+) &:= {}_2F_1\left(\begin{matrix} a+1, b \\ c \end{matrix}; z\right) \\ F(a-) &:= {}_2F_1\left(\begin{matrix} a-1, b \\ c \end{matrix}; z\right) \end{aligned} $$

and analogously for $F(b+), F(b-), F(c+)$ and $F(c-)$. Since all parameters are defined by Pochhammer symbols, they are connected to their neighbours by simple arguments, e.g.

$$ (a+1)_n = \frac{(a+n)!}{(a)!} = \frac{a+n}{a}\frac{(a+n-1)!}{(a-1)!} = \frac{a+n}{a}(a)_n. $$

If we set

$$ \delta_n = \frac{(a)_n (b)_n z^n}{(c)_n n!}, $$

we can then write

$$ \begin{aligned} F(a+) &= \sum_{n=0}^{\infty} \frac{a+n}{a}\delta_n, \quad & F(a-) &= \sum_{n=0}^{\infty} \frac{a-1}{a-1+n}\delta_n, \\ F(b+) &= \sum_{n=0}^{\infty} \frac{b+n}{b}\delta_n, \quad & F(b-) &= \sum_{n=0}^{\infty} \frac{b-1}{b-1+n}\delta_n, \\ F(c+) &= \sum_{n=0}^{\infty} \frac{c}{c+n}\delta_n, \quad & F(c-) &= \sum_{n=0}^{\infty} \frac{c-1+n}{c-1}\delta_n. \end{aligned} $$

Since all relations depend closely on the sum index n we need to interact with z^n. Thus we introduce the Euler differential operator $\theta = z\frac{\partial}{\partial z}$. Applied to z^n it yields $\theta z^n = n z^n$. If we add a to the operator and scale everything by a we get

$$ (\theta + a)F = \sum_{n=0}^{\infty} (a+n)\delta_n = a\sum_{n=0}^{\infty} \frac{(a+n)}{a}\delta_n = aF(a+). \tag{2.3} $$

Similarly we can derive the relations

$$(\theta + b)F = bF(b+) \tag{2.4}$$
$$(\theta + c - 1)F = (c-1)F(c-).$$

By directly applying θ to F and some simple combinatorical arguments we get the relation

$$(1-z)\theta F = (a+b-c)zF + \frac{(c-a)(c-b)}{c} zF(c+).$$

Now all equations have in common that the left-hand site depends on θ. By subtracting (2.4) from (2.3), we get the contiguous relation

$$(a-b)F = aF(a+) - bF(b+).$$

This is the first of 15 contiguous relations for the Gaussian hypergeometric series, although all can be derived by some linear combinations as shown.

Generalized hypergeometric function

The Gaussian hypergeometric series is only depending on the parameters a, b and c. Additional parameters generalize this function.

Definition 2.1.1 (Generalized hypergeometric function)
Let $b_i \notin -\mathbb{N}_0$, then the function

$${}_pF_q\left(\begin{matrix} a_1, a_2, \ldots, a_p \\ b_1, b_2, \ldots, b_q \end{matrix}; z\right) := \sum_{n=0}^{\infty} \frac{\prod_{i=1}^{p}(a_i)_n}{\prod_{j=1}^{q}(b_j)_n} \cdot \frac{z^n}{n!}$$

is called generalized hypergeometric function.

The series terminates if $-a_i \in \mathbb{N}_0$ for some $i \in \{1, \ldots, p\}$. The function converges if either $p \leq q$ and z is finite or if $p = q+1$ and $|z| \leq 1$. It diverges if $p > q+1$, except if it terminates as mentioned before.

Definition 2.1.2 (Balanced hypergeometric series)
A generalized hypergeometric series is called *Saalschützian* (or balanced) if

$$\sum_{j=0}^{q} b_j - \sum_{i=0}^{p} a_i = 1.$$

It is called s-balanced, if

$$\sum_{j=0}^{q} b_j - \sum_{i=0}^{p} a_i = s.$$

A Saalschützian hypergeometric ${}_3F_2$ is summable by the Pfaff-Saalschütz theorem, see e.g.

[AAR99, Rai71] or [Sla66] for the proof.

Theorem 2.1.2 (Pfaff-Saalschütz theorem)
If $n \in \mathbb{N}$

$$ {}_3F_2\left(\begin{matrix} -n, a, b \\ c, 1-c+a+b-n \end{matrix}; 1\right) = \frac{(c-a)_n (c-b)_n}{(c)_n (c-a-b)_n}. $$

This theorem has been extended in [RR11] and [KR12]. Summation theorems for higher p and q exist, e.g. Dougalls summation [Dou06], but further assumptions on a_i and b_i are needed.
See e.g. [Bai64, Sla66] or [AAR99] for more summations and information.

2.1.1. Jacobi Polynomials

The *Jacobi polynomials* are defined for $\alpha, \beta > -1$ by the hypergeometric ${}_2F_1$ series

$$ P_n^{(\alpha,\beta)}(x) = \frac{(1+\alpha)_n}{n!} {}_2F_1\left(\begin{matrix} -n, n+\alpha+\beta+1 \\ 1+\alpha \end{matrix}; \frac{1-x}{2}\right). \tag{2.5} $$

For $\alpha = \beta = 0$ they are the *Legendre polynomials* and for $\alpha = \beta$ they are a scaled version of the *Gegenbauer polynomials*, which include the *Chebychev polynomials* for $\alpha = \beta = \frac{1}{2}$. Due to the terminating property of hypergeometric series in (2.5), $P_n^{(\alpha,\beta)}$ is a polynomial of degree n and $P_n^{(\alpha,\beta)}(1) = \frac{(\alpha+1)_n}{n!}$.
Moreover the property

$$ P_n^{(\alpha,\beta)}(x) = (-1)^n P_n^{(\beta,\alpha)}(-x) \tag{2.6} $$

yields in the representation

$$ P_n^{(\alpha,\beta)}(x) = \frac{(-1)^n(1+\beta)_n}{n!} {}_2F_1\left(\begin{matrix} -n, n+\alpha+\beta+1 \\ 1+\beta \end{matrix}; \frac{1+x}{2}\right). \tag{2.7} $$

Alternatively they can be defined by the Rodrigues formula

$$ P_n^{(\alpha,\beta)}(x) = \frac{(-1)^n(1-x)^{-\alpha}(1+x)^{-\beta}}{2^n n!} \frac{\mathrm{d}^n}{\mathrm{d}x^n}\left[(1-x)^{n+\alpha}(1+x)^{n+\beta}\right], $$

see e.g. [Sze67, Rai71, Chi11, AS65] or [AAR99]. All definitions define the same polynomials. The Jacobi polynomials are orthogonal with respect to the weight function $w^{(\alpha,\beta)}(x) = (1-x)^\alpha(1+x)^\beta$.
Thus

$$ \int_{-1}^{1} P_n^{(\alpha,\beta)}(x) P_m^{(\alpha,\beta)}(x) w^{(\alpha,\beta)}(x)\,\mathrm{d}x = \begin{cases} \frac{2^{\alpha+\beta+1}}{2n+\alpha+\beta+1} \frac{\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{n!\,\Gamma(n+\alpha+\beta+1)} & n = m, \\ 0 & n \neq m \end{cases} $$

holds.
Since Jacobi polynomials are orthogonal polynomials on a real interval with a positive weight function, they are bound to have a three term recursion by Favard's Theorem see e.g. [AAR99].

For the Jacobi polynomials the three term recursion is given by

$$\begin{aligned}2n(\alpha+\beta+n)(\alpha+\beta+2n-2)P_n^{(\alpha,\beta)}(x) = \\ (\alpha+\beta+2n-1)\left[\alpha^2-\beta^2+x(\alpha+\beta-2n)(\alpha+\beta+2n-2)\right]P_{n-1}^{(\alpha,\beta)}(x) \\ -2(\alpha+n-1)(\beta+n-1)(\alpha+\beta+2n)P_{n-2}^{(\alpha,\beta)}(x).\end{aligned} \tag{2.8}$$

Furthermore since $P_n^{(\alpha,\beta)}(x)$ is described by a Gaussian hypergeometric function, we can apply all contiguous relations to derive relations between Jacobi polynomials of different degree and/or different parameters. The following lemma is a collection of relations between Jacobi polynomials as they can be found in [Rai71, Chapter 16].

Lemma 2.1.3
For $\alpha, \beta > -1$ and $n \in \mathbb{N}$ holds

$$\begin{aligned}(\alpha+\beta+n)P_n^{(\alpha,\beta)}(x) &= (\beta+n)P_n^{(\alpha,\beta-1)}(x) + (\alpha+n)P_n^{(\alpha-1,\beta)}(x) \\ \frac{1}{2}(2+\alpha+\beta+2n)(x-1)P_n^{(\alpha+1,\beta)}(x) &= (n+1)P_{n+1}^{(\alpha,\beta)}(x) - (1+\alpha+n)P_n^{(\alpha,\beta)}(x) \\ \frac{1}{2}(2+\alpha+\beta+2n)(x+1)P_n^{(\alpha,\beta+1)}(x) &= (n+1)P_{n+1}^{(\alpha,\beta)}(x) + (1+\beta+n)P_n^{(\alpha,\beta)}(x) \\ (\alpha+\beta+2n)P_n^{(\alpha,\beta-1)}(x) &= (\alpha+\beta+n)P_n^{(\alpha,\beta)}(x) + (\alpha+n)P_{n-1}^{(\alpha,\beta)}(x) \\ (\alpha+\beta+2n)P_n^{(\alpha-1,\beta)}(x) &= (\alpha+\beta+n)P_n^{(\alpha,\beta)}(x) - (\beta+n)P_{n-1}^{(\alpha,\beta)}(x) \\ 2P_n^{(\alpha,\beta)}(x) &= (1+x)P_n^{(\alpha,\beta+1)}(x) + (1-x)P_n^{(\alpha+1,\beta)}(x) \\ P_{n-1}^{(\alpha,\beta)}(x) &= P_n^{(\alpha,\beta-1)}(x) - P_n^{(\alpha-1,\beta)}(x).\end{aligned}$$

The k-th derivative of Jacobi polynomials can be written as

$$\frac{\mathrm{d}^k}{\mathrm{d}x^k}P_n^{(\alpha,\beta)}(x) = 2^{-k}(1+\alpha+\beta+n)_k P_{n-k}^{(\alpha+k,\beta+k)}(x). \tag{2.9}$$

In the application of Jacobi polynomials in high order finite element methods it is often useful to define integrated Jacobi polynomials as follows

$$\widehat{P}_n^{(\alpha,0)}(x) = \int_{-1}^{x} P_{n-1}^{(\alpha,0)}(t)\,\mathrm{d}t.$$

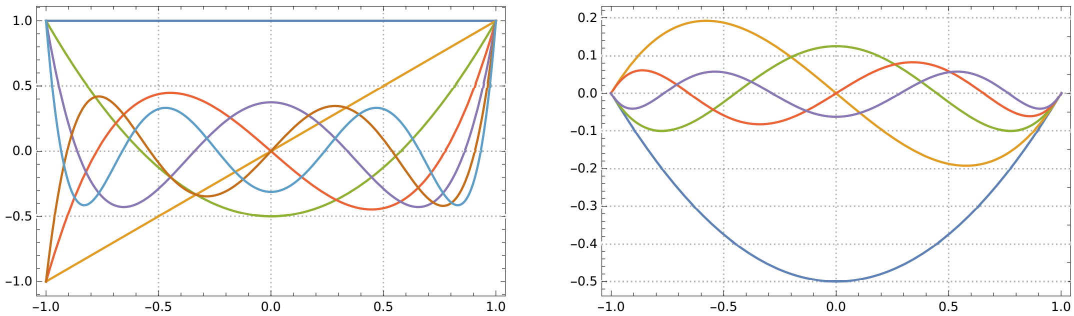
Integrating (2.9) with $k = 1$ yields the following relation between Jacobi and integrated Jacobi polynomials

$$\widehat{P}_n^{(\alpha,0)}(x) = \frac{2}{n+\alpha-1}P_n^{(\alpha-1,-1)}(x)$$

and for the integrated Legendre polynomials

$$\widehat{L}_n(x) = \widehat{P}_n^{(0,0)}(x) = \frac{2}{n-1}P_n^{(-1,-1)}(x).$$

Although Jacobi polynomials in their hypergeometric form are not clearly defined for negative parameters α and β, they can be generalized as can be seen in [Sze67], see also [GSW09] or [Xu17]. One needs to be careful, otherwise a reduction in the polynomial degree due to the Pochhammer



(a) Legendre polynomials $p = 0, \dots, 6$.

(b) Integrated Legendre polynomials $p = 2, \dots, 6$.

Figure 2.1.: Legendre polynomials and integrated Legendre polynomials.

symbols can happen for certain polynomial degrees. Following [Sze67], we thus write

$$\widehat{P}_n^{(\alpha,0)}(x) = \frac{2}{n+\alpha-1} P_n^{(\alpha-1,-1)}(x) = \frac{1+x}{n} P_{n-1}^{(\alpha-1,1)}(x) \tag{2.10}$$

and for the integrated Legendre polynomials

$$\widehat{L}_n(x) = \frac{2}{n-1} P_n^{(-1,-1)}(x) = \frac{x^2-1}{2(n-1)} P_{n-2}^{(1,1)}(x). \tag{2.11}$$

The Legendre polynomials and their integrated variant are depicted in Figure 2.1.

Remark 1
The so called *generalized* Jacobi polynomials, see e.g. [GSW09], are thus identical to the integrated Jacobi polynomial except for a scaling, which depends on the polynomial degree n and the parameter α.

2.1.2. Integral product of Jacobi polynomials

As we see later on, we are interested in efficiently evaluating integrals of the form

$$I_{n,m} := \int_1^1 (1-x)^\mu \, (1+x)^\nu \, P_n^{(\alpha,\beta)}(x) P_m^{(\rho,\delta)}(x) \, \mathrm{d}x. \tag{2.12}$$

Usually this is done by a numerical quadrature, but due to the hypergeometric basis, we can derive recurrence formulas between neighbours of $I_{n,m}$ with respect to n and m.
Such recursion formulas can be calculated by symbolic software. Examples of such software-packages are *SumCracker* [Kau05, Kau06] or *Guess* [Kau09], both written in *Mathematica* [Inc]. Guessing is a well known technique, where the aim is to give a plausible hypothesis about relations of an infinite sequence, see e.g. [HR11, Kau13] or [SZ94]. Let us show the application of the package `Guess` on an example.

Example 2.1.1
Let $\beta = \delta = \nu = 0$. Then the integral (2.12) reduces to

$$I_{n,m} := \int_{1}^{1} (1-x)^{\mu}\, P_n^{(\alpha,0)}(x) P_m^{(\rho,0)}(x)\, \mathrm{d}x. \tag{2.13}$$

To connect $I_{n,m}$ of different degrees n, m in a recurrence formula, we will integrate $I_{n,m}$ exactly for different n, m, α, ρ, μ. `Guess` takes this multivariate table as input and returns a possible hypothesis for such recurrence formulas. The following is a coding example for such a task

```
In[1]:= data = ParallelTable[Integrate[(1 − x)^μ JacobiP[n, α, 0, x] JacobiP[m, ρ, 0, x], {x, −1, 1}], {n, 2, 10},
        {α, 0, 10}, {m, 2, 10}, {ρ, 0, 10}, {μ, 0, 4}]

In[2]:= GuessMultRE[data, Flatten[Table[F[n + a1, α, m + a2, ρ, μ], {a1, 0, 1}, {a2, 0, 1}]], {n, α, m, ρ, μ}, 1,
        StartPoint → {2, 0, 2, 0, 0}]

Out[2]= {−(1 + m + n + α − μ + ρ)F[n, α, m, ρ, μ] + (1 + m − n − α + μ)F[n, α, 1 + m, ρ, μ] − (−1 + m − n − μ +
        ρ)F[1 + n, α, m, ρ, μ] + (3 + m + n + μ)F[1 + n, α, 1 + m, ρ, μ]}
```

Note that we specified the structure of the recursion in the second option of the command `GuessMultRE`. We allowed a shift by one in both directions, which corresponds with a shift in the polynomial order n and m. The third option specifies the order of the coefficients in the exact table `data`. With the next option we give the polynomial order for the coefficients, i.e. here it is 1, which means that we search for a linear recurrence relation. The last option just specifies the starting number of the variables, i.e. we start with $n, m = 2$ and $\alpha, \rho, \mu = 0$.

The example yields the following conjecture

Conjecture 1
Let $I_{n,m}$ be as in (2.13), then the following recurrence relation holds

$$\begin{aligned}(3+m+n+\mu)I_{n+1,m+1} = (-1+m-n-\mu+\rho)I_{n+1,m} + (-1-m+n+\alpha-\mu)I_{n,m+1} \\ + (1+m+n+\alpha-\mu+\rho)I_{n,m}\end{aligned} \tag{2.14}$$

At this stage relation (2.14) is only a conjecture, since we only *guessed* this relation. A proof of this conjecture can be done by symbolic computation as well. One could use the package `HolonomicFunctions` [Kou10] to show that $I_{n,m}$ is holonomic[1] and find the annihilating operator, which is just equation (2.14). Alternatively, one could use the package `SumCracker` [Kau05] to proof this conjecture.
Instead of going into more details of the symbolic proofs, we will instead focus on an analytic proof. Since $I_{n,m}$ is not feasible in the form (2.13), we start by rewriting the equation in terms of hypergeometric functions. Due to (2.5) we can write (2.13) as

$$I_{n,m} = \frac{2^{\mu}(\alpha+1)_n(\rho+1)_m}{n!\, m!} \sum_{l=0}^{n}\sum_{r=0}^{m} \frac{(-n)_l(n+\alpha+1)_l(-m)_r(m+\rho+1)_r}{(\alpha+1)_l(\rho+1)_r l!\, r!} \int_{-1}^{1} \left(\frac{1-x}{2}\right)^{\mu+r+l} \mathrm{d}x. \tag{2.15}$$

[1] An univariate infinite sequence $(a_n)_{n=0}^{\infty}$ is called holonomic if there exist an $r \in \mathbb{N}$ and univariate polynomials $p_0, \ldots, p_r$, s.t. $p_0(n)a_n + p_1(n)a_{n+1} + \cdots + p_r(n)a_{n+r} = 0$. In the multivariate case this is extended by a definition by the power series.

The integral term in (2.15) simplifies by using the definition of the beta integral (2.1) to

$$\int_{-1}^{1} \left(\frac{1-x}{2}\right)^{\mu+r+l} \mathrm{d}x = \frac{2}{\mu+r+l+1}.$$

Now we can rewrite the right-hand side in terms of Pochhammer symbols, i.e.

$$\frac{2}{\mu+r+l+1} = 2\frac{(\mu+r+1)_l}{(\mu+r+2)_l}\frac{\Gamma(\mu+r+1)}{\Gamma(\mu+r+2)}.$$

To use the Pfaff-Saalschütz theorem, we separate both sums again

$$I_{n,m} = \frac{2^{\mu+1}(\alpha+1)_n(\rho+1)_m}{n!\,m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{\Gamma(\mu+r+1)}{\Gamma(\mu+r+2)}\sum_{l=0}^{n}\frac{(-n)_l(n+\alpha+1)_l(\mu+r+1)_l}{(\alpha+1)_l(\mu+r+2)_l l!}. \quad (2.16)$$

The innermost sum is now summable by the Pfaff-Saalschütz theorem with

$$a = n+\alpha+1,$$
$$b = \mu+r+1,$$
$$c = \mu+r+2.$$

It follows with $1-c+(a+b)-n = 1-(\mu+r+2)+(n+\alpha+1)+(\mu+r+1)-n = \alpha+1$, that

$$\sum_{l=0}^{n}\frac{(-n)_l(n+\alpha+1)_l(\mu+r+1)_l}{(\alpha+1)_l(\mu+r+2)_l l!} = \frac{(\mu+r+2-n-\alpha-1)_n(1)_n}{(\mu+r+2)_n(-n-\alpha)_n}$$

Application to (2.16) yields

$$I_{n,m} = \frac{2^{\mu+1}(\alpha+1)_n(\rho+1)_m}{n!\,m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{\Gamma(\mu+r+1)}{\Gamma(\mu+r+2)}\frac{(\mu+r+1-n-\alpha)_n(1)_n}{(\mu+r+2)_n(-\alpha-n)_n}.$$

We rewrite the negative Pochhammer symbols by the combinatorical argument

$$(a-n)_n = (-1)^n(1-a)_n$$

and

$$(a-n)_N = \frac{(1-a)_n(a)_N}{(1-a-N)_n}, \quad (a+n)_N = \frac{(a)_N(a+N)_n}{(a)_n} \quad (2.17)$$

with $(1)_n = n!$, see e.g. [Sla66, p. 239 Appendix I].

In detail, we rewrite the sum by

$$
\begin{aligned}
I_{n,m} &= \frac{2^{\mu+1}(\alpha+1)_n(\rho+1)_m}{n!\,m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{\Gamma(\mu+r+1)}{\Gamma(\mu+r+2)}\frac{(-1)^n(\alpha-(\mu+r))_n(1)_n}{(\mu+r+2)_n(-1)^n(1+\alpha)_n},\\
&= \frac{2^{\mu+1}(\rho+1)_m}{m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{\Gamma(\mu+r+1)}{\Gamma(\mu+r+2)}\frac{(\alpha-(\mu+r))_n}{(\mu+r+2)_n}\\
&= \frac{2^{\mu+1}(\rho+1)_m}{m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{(\mu+1)_r\Gamma(\mu+1)}{(\mu+2)_r\Gamma(\mu+2)}\frac{(\alpha-(\mu+r))_n}{(\mu+r+2)_n}\\
&\overset{(2.17)}{=} \frac{2^{\mu+1}(\rho+1)_m}{m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{(\mu+1)_r\Gamma(\mu+1)}{(\mu+2)_r\Gamma(\mu+2)}\frac{(1-\alpha+\mu)_r(\alpha-\mu)_n}{(1-\alpha+\mu-n)_r}\frac{(\mu+2)_r}{(\mu+2)_n(\mu+2+n)_r}\\
&= \frac{2^{\mu+1}(\rho+1)_m(\alpha-\mu)_n}{(\mu+1)_{n+1}m!}\sum_{r=0}^{m}\frac{(-m)_r(m+\rho+1)_r}{(\rho+1)_r r!}\frac{(1-\alpha+\mu)_r}{(1-\alpha+\mu-n)_r}\frac{(\mu+1)_r}{(\mu+2+n)_r}\\
&= \frac{2^{\mu+1}(\rho+1)_m(\alpha-\mu)_n}{(\mu+1)_{n+1}m!}\,{}_4F_3\left(\begin{matrix}-m,m+\rho+1,1-\alpha+\mu,\mu+1\\ \rho+1,1-\alpha+\mu-n,\mu+2+n\end{matrix};1\right)
\end{aligned}
$$

Although the ${}_4F_3$ is Saalschützian, the summation theorem in [KR12] is not applicable. Those summation theorems need a special form of the coefficients, which is not given in this case. Classical results by Whipple's transformation followed by Dougalls summation [Bai64], only works in special cases like $n = m$.

But the summability is not needed. Here, we can derive relations with a brute force method: We compare contiguous relations and solve for the unknown coefficients. Due to the guessed recursion, we know a priori that we need relation of the form

$$
c_0 I_{n+1,m+1} + c_1 I_{n,m+1} + c_2 I_{n+1,m} + c_3 I_{n,m} = 0. \tag{2.18}
$$

Thus, we start by deriving the relations of the ${}_4F_3$. We introduce the following notation

$$
{}_4F_3\left(\begin{matrix}-m,m+\rho+1,1-\alpha+\mu,\mu+1\\ \rho+1,1-\alpha+\mu-n,\mu+2+n\end{matrix};1\right) = \sum_{k=0}^{\infty}\underbrace{\frac{(-m)_k(m+\rho+1)_k(1-\alpha+\mu)_k(1)_k}{(\rho+1)_k(1-\alpha+\mu-n)_k(\mu+2+n)_k}\frac{1}{k!}}_{=:\Phi_k}.
$$

If we raise one of the parameters we can give the change of the summand in terms of Φ_k. The change of $n \to n+1$ after equating the Pochhammer symbols is as follows

$$
{}_4F_3\left(\begin{matrix}-m,m+\rho+1,1-\alpha+\mu,\mu+1\\ \rho+1,-\alpha+\mu-n,\mu+3+n\end{matrix};1\right) = \sum_{k=0}^{\infty}\frac{(k-\alpha+\mu-n)(2+n+\mu)}{(-\alpha+\mu-n)(2+n+\mu+k)}\Phi_k
$$

and for $m \to m+1$

$$
{}_4F_3\left(\begin{matrix}-(m+1),m+\rho+2,1-\alpha+\mu,\mu+1\\ \rho+1,1-\alpha-n,\mu+2+n\end{matrix};1\right) = \sum_{k=0}^{\infty}\frac{(-m-1)(m+\rho+1+k)}{(k-m-1)(m+\rho+1)}\Phi_k.
$$

With an analogue formulation for $n \to n+1, m \to m+1$, we insert those relations into (2.18) such that

$$0 = \sum_{k=0}^{\infty} \left(c_0 \frac{(m+\rho+1+k)(k-\alpha+\mu-n)}{(k-m-1)(2+n+\mu+k)} + c_1 \frac{(k-\alpha+\mu-n)}{(2+n+\mu+k)} + c_2 \frac{(m+\rho+1+k)}{(k-m-1)} + c_3 \right) \Phi_k$$

holds. Note that we included all terms which are independent on k into the coefficients c_i. Next we expand the fractions in the same denominator and search for the root of all summands. Thus,

$$\begin{aligned} 0 = c_0(m+\rho+1+k)(k-\alpha+\mu-n) + c_1(k-\alpha+\mu-n)(k-m-1) \\ + c_2(m+\rho+1+k)(2+n+\mu+k) + c_3(k-m-1)(2+n+\mu+k). \end{aligned}$$

Since this is a polynomial in k, all monomials $1, k, k^2$ need to be zero. Thus, we get three equations which need to be fulfilled, namely

$$\begin{aligned} &k^2(c_0+c_1+c_2+c_3) &&= 0 \\ &k\left(c_0(m+\rho-\alpha+\mu-n+1) - c_1(m+\alpha+n-\mu+1) + c_2(n+\mu+m+\rho+3) + c_3(n+\mu-m+1)\right) &&= 0 \\ &\left(c_0(m+\rho+1)(-\alpha+\mu-n) + c_1(\alpha-\mu+n)(m+1) + c_2(m+\rho+1)(2+n+\mu) - c_3(m+1)(2+n+\mu)\right) &&= 0. \end{aligned}$$

This leads to the underdetermined system

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ (m+\rho-\alpha+\mu-n+1) & -(m+\alpha+n-\mu+1) & (n+\mu+m+\rho+3) & (n+\mu-m+1) \\ (m+\rho+1)(-\alpha+\mu-n) & (\alpha-\mu+n)(m+1) & (m+\rho+1)(2+n+\mu) & -(m+1)(2+n+\mu) \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

where we can choose $c_0 = 1$ and transform the right-hand side as follows

$$\begin{pmatrix} 1 & 1 & 1 \\ -(m+\alpha+n-\mu+1) & (n+\mu+m+\rho+3) & (n+\mu-m+1) \\ (\alpha-\mu+n)(m+1) & (m+\rho+1)(2+n+\mu) & -(m+1)(2+n+\mu) \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} -1 \\ -(m+\rho-\alpha+\mu-n+1) \\ -(m+\rho+1)(-\alpha+\mu-n) \end{pmatrix}$$

The solution of this system is

$$\begin{aligned} c_1 &= \frac{m+n+\rho-\mu-1}{3+m+n+\mu}, \\ c_2 &= \frac{n+\alpha-m-\mu-1}{3+m+n+\mu}, \\ c_3 &= -\frac{1+m+n+\alpha+\rho-\mu}{3+m+n+\mu}. \end{aligned}$$

If we rescale everything including c_0 by $(3+m+n+\mu)$ we have proven conjecture 1.
A generalization with arbitrary coefficients ν, β, δ is not possible due to the loss of summability. Since the general Euler integral

$$\int_{-1}^{1} (1-x)^{\mu+l+r}(1+x)^{\nu}\,\mathrm{d}x = B(\mu+l+r+1, \nu+1) = \frac{(\mu+r+1)_l}{(\mu+\nu+r+1)_l} \frac{\Gamma(\mu+r+1)\Gamma(\nu+1)}{\Gamma(\mu+\nu+r+2)},$$

we see that the inner summand is $(\nu-\beta+1)$-balanced. Thus, the inner sum is not summable in all generality, since there is no such summing theorem. If $\beta = \delta = \nu$ and $\alpha = \rho = \mu$ the series is summable and one is able to prove orthogonality of the Jacobi polynomials, as can be seen in [AB95].

2.2. Kampé de Fériet Series

It is well known that the concept of hypergeometric series can be extended to the multivariate case. Examples of such series include the Appell series[App25], an extension of the Gaussian ${}_2F_1$, the Kampé de Fériet Series [AK26], or the Humber series [Hum26]. A collection of such and further series can be found, e.g. in [EMOT54, Ext76, SK85].

Definition 2.2.1 (Kampé de Fériet series)
For $p_1, p_2, p_3, q_1, q_2, q_3 \in \mathbb{N}$ and coefficients $(a_1, \dots a_{q_1}), (b_1, \dots b_{p_1}), \dots$ arbitrarily the series

$$F^{p_1;p_2;p_3}_{q_1;q_2;q_3} = \sum_{n,m=0}^{\infty} \frac{\prod_{i=1}^{p_1}(k_i)_{n+m} \prod_{i=1}^{p_2}(a_i)_n \prod_{i=1}^{p_3}(b_i)_m}{\prod_{i=1}^{q_1}(l_i)_{n+m} \prod_{i=1}^{q_2}(c_i)_n \prod_{i=1}^{q_3}(d_i)_m} \frac{x^n y^m}{n!\, m!}$$

is called Kampé de Fériet series.

A Kampé de Fériet series diverges if one of the l_i, c_i, d_i is zero or a negative integer. It is terminating if one of the k_i, a_i, b_i is zero or a negative integer. More information regarding convergence can be found e.g. in [Ext76].
The notation was introduced by Burchnall and Chaundy [BC40, BC41]. Although (2.12) is not summable for arbitrary coefficients, we can still represent it in terms of a Kampé de Fériet series

$$\begin{aligned} F \coloneqq I_{n,m} = 2^{\mu+\nu+1} & \frac{(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\, m!} \\ & F^{1;2;2}_{1;1;1}\left(\begin{matrix} \mu+1 & ; & -n & n+\alpha+\beta+1 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+1 & & ; & \rho+1 & \end{matrix} ;1;1\right). \end{aligned} \tag{2.19}$$

Here F converges due to the negative coefficients $-m$ and $-n$. We use the simple notation of contiguous relations as we have seen in the ${}_2F_1$ case. We denote a shift of parameters by unity as follows,

$$\begin{aligned} F(\alpha+) = & 2^{\mu+\nu+1} \frac{(\alpha+2)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\, m!} \\ & F^{1;2;2}_{1;1;1}\left(\begin{matrix} \mu+1 & ; & -n & n+\alpha+\beta+2 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+2 & & ; & \rho+1 & \end{matrix} ;1;1\right) \\ F(\nu+) = & 2^{\mu+\nu+2} \frac{(\alpha+1)_n(\rho+1)_m B(\nu+2,\mu+1)}{n!\, m!} \\ & F^{1;2;2}_{1;1;1}\left(\begin{matrix} \mu+1 & ; & -n & n+\alpha+\beta+1 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+3 & ; & \alpha+1 & & ; & \rho+1 & \end{matrix} ;1;1\right) \end{aligned}$$

and so on.

Recurrence Relations

We can now directly derive recurrence relations for (2.19) by using the recurrence relations of lemma 2.1.3.

Corollary 2.2.1
Let $\alpha, \beta, \mu, \nu > -1$ and $n \in \mathbb{N}$, then holds

$$\begin{aligned}
(\alpha+\beta+n)F &= (\beta+n)F(\beta-) + (\alpha+n)F(\alpha-), \\
-\frac{1}{2}(2+\alpha+\beta+2n)F(\alpha+,\mu+) &= (n+1)F(n+) - (1+\alpha+n)F, && (2.20)\\
\frac{1}{2}(2+\alpha+\beta+2n)F(\beta+,\nu+) &= (n+1)F(n+) + (1+\beta+n)F, && (2.21)\\
(\alpha+\beta+2n)F(\beta-) &= (\alpha+\beta+n)F + (\alpha+n)F(n-), && (2.22)\\
(\alpha+\beta+2n)F(\alpha-) &= (\alpha+\beta+n)F - (\beta+n)F(n-), && (2.23)\\
2F &= F(\nu+,\beta+) + F(\mu+,\alpha+), \\
F(n-) &= F(\beta-) - F(\alpha-).
\end{aligned}$$

Furthermore there are also 7 analogue recurrence formulas in m, ρ and δ, where

$$(\rho+\delta+2m)F(\delta-) = (\rho+\delta+m)F + (\rho+m)F(m-)$$

is one of them.
Moreover, we can derive more recurrence relations by linear combinations of the above.

Corollary 2.2.2
The following recurrence formula holds,

$$\begin{aligned}
0 = &-2(1+\alpha+n)F - (1+\beta+n)F(\alpha+) + (2+\alpha+\beta+2n)F(\alpha+,\mu+) \\
&-(\alpha+\beta+1)F(n+) + (2+\alpha+\beta+n)F(n+,\alpha+)
\end{aligned} \quad (2.24)$$

Proof. The equation is just a linear combination of (2.20) and (2.23). Start by transforming (2.20) and (2.23) with $\alpha \to \alpha+1, n \to n+1$,

$$\begin{aligned}
0 &= (2+\alpha+\beta+2n)F(\alpha+,\mu+) - 2(n+1)F(n+) - 2(1+\alpha+n)F, \\
0 &= -(\alpha+\beta+2n+3)F(n+) - (\beta+n+1)F(\alpha+) + (\alpha+\beta+n+2)F(n+,\alpha+).
\end{aligned}$$

Adding these equations lead to (2.24). □

One important, but rather trivial relation is given by

$$2F = F(\nu+) + F(\mu+), \quad (2.25)$$

which follows from $2P_n^{(\alpha,\beta)}(x) = (1+x)P_n^{(\alpha,\beta)}(x) + (1-x)P_n^{(\alpha,\beta)}(x)$.
The following relations are proven in appendix A.1 by using a generalized form of (2.19). Set $x = y = 1$ in lemma A.1.2 and A.1.3, then the following corollary holds.

Corollary 2.2.3
Let $\alpha, \beta, \rho, \delta, \mu, \nu > -1$ and $m, n \in \mathbb{N}$, then holds

$$(n+m+\mu+\nu+4)F(n+,m+,\nu+) = (\alpha+n+1)F(m+,\beta+,\nu+) + (\rho+m+1)F(n+,\delta+,\nu+) + 2(\nu+1)F(n+,m+), \tag{2.26}$$

$$(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F = (n+\alpha+\beta+1)F(\beta+) + (m+\rho+\delta+1)F(\delta+) + 2\nu F(\nu-). \tag{2.27}$$

Since the weights in the Jacobi polynomials are interchangeable, see (2.6), the following two relations can be derived as well.

Corollary 2.2.4
Let $\alpha, \beta, \rho, \delta, \mu, \nu > -1$ and $m, n \in \mathbb{N}$, then holds

$$(n+m+\mu+\nu+4)F(n+,m+,\mu+) = -(\beta+n+1)F(m+,\alpha+,\mu+) - (\delta+m+1)F(n+,\rho+,\mu+) + 2(\mu+1)F(n+,m+), \tag{2.28}$$

$$(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F = (n+\alpha+\beta+1)F(\alpha+) + (m+\rho+\delta+1)F(\rho+) + 2\mu F(\mu-). \tag{2.29}$$

Proof. See lemmas A.1.5 and A.1.6 and set $x = y = 1$. □

5 - point recurrence relation There are some known starlike recurrence relations, see [PPSS06]. Those can be derived in this context as follows.

Corollary 2.2.5
Let $\alpha, \beta, \rho, \delta, \mu, \nu > -1$ and $m, n \in \mathbb{N}$, then holds the mixed recurrence relations

$$\begin{aligned}&(2m+\rho+\delta+1)\left((n+1)F(n+,\alpha-) - (\alpha+n)F(\alpha-)\right)\\ &\quad = (2n+\alpha+\beta+1)\left((m+1)F(m+,\rho-) - (\rho+m)F(\rho-)\right)\end{aligned} \tag{2.30}$$

and

$$\begin{aligned}&(2m+\rho+\delta+1)\left((n+1)F(n+,\beta-) - (\beta+n)F(\beta-)\right)\\ &\quad = (2n+\alpha+\beta+1)\left((m+1)F(m+,\delta-) - (\delta+m)F(\delta-)\right)\end{aligned} \tag{2.31}$$

Proof. Take (2.20) and replace α by $\alpha - 1$ to derive

$$F(\mu+) = \frac{-2}{2n+\alpha+\beta+1}\left((n+1)F(n+,\alpha-) - (\alpha+n)F(\alpha-)\right).$$

Also replace ρ by $\rho - 1$ in the analog to (2.20) w.r.t. m, ρ, δ to derive

$$F(\mu+) = \frac{-2}{2m+\rho+\delta+1}\left((m+1)F(m+,\rho-) - (\rho+m)F(\rho-)\right).$$

Setting both right-hand sides equal yields (2.30). The second mixed relation follows analogously from the recursion formula (2.21). □

The mixed relations (2.30) and (2.31) yield some 5-point recurrence relations with support $(m,n),(m-1,n),(m+1,n),(m,n-1),(m,n+1)$, see also [PPSS06].

Theorem 2.2.6

$$\begin{aligned}
&(2m+\rho+\delta)_3\big[(n+1)\left((n+\alpha+\beta+1)(2n+\alpha+\beta)F(n+)+(n+\alpha+1)F\right)\\
&\quad+(\beta+n)\left((n+\alpha+\beta)F+(n+\alpha)F(n-)\right)\big]\\
=&(2n+\alpha+\beta)_3\big[(m+1)\left((m+\rho+\delta+1)(2m+\rho+\delta)F(m+)+(m+\rho+1)F\right)\\
&\quad+(m+\delta)\left((m+\rho+\delta)F+(m+\delta)F(m-)\right)\big],
\end{aligned} \tag{2.32}$$

$$\begin{aligned}
&(2m+\rho+\delta)_3\big[(n+1)\left((n+\alpha+\beta+1)(2n+\alpha+\beta)F(n+)-(n+\beta+1)F\right)\\
&\quad-(\alpha+n)\left((n+\alpha+\beta)F-(n+\beta)F(n-)\right)\big]\\
=&(2n+\alpha+\beta)_3\big[(m+1)\left((m+\rho+\delta+1)(2m+\rho+\delta)F(m+)-(m+\delta+1)F\right)\\
&\quad+(m+\rho)\left((m+\rho+\delta)F+(m+\rho)F(m-)\right)\big],
\end{aligned} \tag{2.33}$$

$$\begin{aligned}
&(2m+\rho+\delta)_3\big[(n+1)\left(2(n+\alpha+\beta+1)(2n+\alpha+\beta)F(n+)+(\alpha-\beta)F\right)\\
&\quad+\left((\beta-\alpha)(n+\alpha+\beta)F+(n+\alpha)(n+\beta)F(n-)\right)\big]\\
=&(2n+\alpha+\beta)_3\big[(m+1)\left(2(m+\rho+\delta+1)(2m+\rho+\delta)F(m+)+(\rho-\delta)F\right)\\
&\quad+\left((\delta-\rho)(m+\rho+\delta)F+(m+\rho)(m+\delta)F(m-)\right)\big]
\end{aligned} \tag{2.34}$$

Proof. Take the first mixed relation (2.30) and replace all series by (2.23), this yields the first equation. The second equation follows by using (2.31) with (2.22). Lastly, the equation (2.34) can be derived by linear combination of (2.32) and (2.33) □

Alternatively one can use the 3-term recursion (2.8) to prove the same result as in [PPSS06].

Recurrence relation Multiple recurrence relations similar to (2.14) can be proven.

Lemma 2.2.7
Let $F = I_{n,m}$, where $I_{n,m}$ is as in (2.19). Then the following recurrence relation holds

$$\begin{aligned}
&(n+\alpha+\beta+1)(m+\rho+\delta+1)\left((n+m+\mu+\nu+4)F(n+,m+,\nu+)-2(\nu+1)F(n+,m+)\right)\\
&\quad=(\alpha+n+1)(m+\rho+\delta+1)\left((n+\alpha+\beta-m-\mu-\nu-2)F(m+,\nu+)+2(\nu+1)F(m+)\right)\\
&\qquad+(\rho+m+1)(n+\alpha+\beta+1)((-n+m+\rho+\delta-\mu-\nu-2)F(n+,\nu+)+2(\nu+1)F(n+))\\
&\qquad+(\rho+m+1)(\alpha+n+1)((n+\alpha+\beta+m+\rho+\delta-\mu-\nu)F(\nu+)+2(\nu+1)F).
\end{aligned} \tag{2.35}$$

Proof. Start with equation (2.26)

$$(n+m+\mu+\nu+4)F(n+,m+,\nu+)-2(\nu+1)F(n+,m+)=(\alpha+n+1)F(m+,\beta+,\nu+)+(\rho+m+1)F(n+,\delta+,\nu+).$$

Replace both terms of the RHS by using shifted versions of equation (2.27), i.e.

$$\begin{aligned}
&\frac{(n+\alpha+1)}{(n+\alpha+\beta+1)}(n+\alpha+\beta+1)F(m+,\beta+,\nu+)\\
=&\frac{(n+\alpha+1)}{(n+\alpha+\beta+1)}[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(m+,\nu+)\\
&-(m+\rho+\delta+2)F(m+,\delta+,\nu+)+2(\nu+1)F(m+)],
\end{aligned}\tag{2.36}$$

and

$$\begin{aligned}
&\frac{(m+\rho+1)}{(m+\rho+\delta+1)}(m+\rho+\delta+1)F(n+,\delta+,\nu+)\\
=&\frac{(m+\rho+1)}{(m+\rho+\delta+1)}[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(n+,\nu+)\\
&-(n+\alpha+\beta+2)F(n+,\beta+,\nu+)+2(\nu+1)F(n+)].
\end{aligned}\tag{2.37}$$

Moreover use the shifted relation (2.22) for the middle part of the last two equations, i.e.

$$(m+\rho+\delta+2)F(m+,\delta+,\nu+)=(2m+\rho+\delta+3)F(m+,\nu+)-(m+\rho+1)F(\delta+,\nu+)\tag{2.38}$$

$$(n+\alpha+\beta+2)F(n+,\beta+,\nu+)=(2n+\alpha+\beta+3)F(n+,\nu+)-(n+\alpha+1)F(\beta+,\nu+).\tag{2.39}$$

Lastly replace the remaining terms by a shifted version of (2.27),

$$\begin{aligned}
&\frac{(n+\alpha+1)}{(n+\alpha+\beta+1)}(m+\rho+1)F(\delta+,\nu+)+\frac{(m+\rho+1)}{(m+\rho+\delta+1)}(n+\alpha+1)F(\beta+,\nu+)\\
=&\frac{(n+\alpha+1)(m+\rho+1)}{(n+\alpha+\beta+1)(m+\rho+\delta+1)}\left((n+\alpha+\beta+1)F(\beta+,\nu+)+(m+\rho+\delta+1)F(\delta+,\nu+)\right)\\
=&\frac{(n+\alpha+1)(m+\rho+1)}{(n+\alpha+\beta+1)(m+\rho+\delta+1)}\left((n+\alpha+\beta+m+\rho+\delta-\mu-\nu)F(\nu+)+2(\nu+1)F\right)
\end{aligned}$$

The claim follows from combining the above with the remaining terms of (2.36), (2.37), (2.38) and (2.39). □

Since α and β or ρ and δ are interchangeable, the following lemma can be proven by using (2.28) and (2.29) instead of (2.26) and (2.27).

Lemma 2.2.8
Let $F = I_{n,m}$, where $I_{n,m}$ is as in (2.19). Then the following recurrence relation holds

$$\begin{aligned}
&(n+\alpha+\beta+1)(m+\rho+\delta+1)\left((n+m+\mu+\nu+4)F(n+,m+,\mu+)-2(\mu+1)F(n+,m+)\right)\\
&=-(\beta+n+1)(m+\rho+\delta+1)\left((n+\alpha+\beta-m-\mu-\nu-2)F(m+,\mu+)+2(\mu+1)F(m+)\right)\\
&\quad-(\delta+m+1)(n+\alpha+\beta+1)\left((-n+m+\rho+\delta-\mu-\nu-2)F(n+,\mu+)+2(\mu+1)F(n+)\right)\\
&\quad+(\delta+m+1)(\beta+n+1)\left((n+\alpha+\beta+m+\rho+\delta-\mu-\nu)F(\mu+)+2(\mu+1)F\right).
\end{aligned}\tag{2.40}$$

Both of these recursion formulas have the drawback, that terms with $\nu+1$ or $\mu+1$ vanish only for $\nu=-1$ or $\mu=-1$, which correspond to the special cases $(1+x)^0$ or $(1-x)^0$. If the steps of the proof are slightly adjusted, a recursion formula, which reduces to a smaller form for more cases, can be proven. The following theorem is the main result of this chapter.

Theorem 2.2.9 (Main recurrence theorem)
Let $F = I_{n,m}$, where $I_{n,m}$ is as in (2.19). Then the following recurrence relation holds

$$\begin{aligned}(n{+}1)(m+1)\left((n+m+\mu+\nu+4)F(n+,m+,\nu+)-2(\nu+1-\beta-\delta)F(n+,m+)\right)\\ =(n+\beta+1)(m+1)\left((n+\alpha+\beta-m-\mu-\nu-2)F(m+,\nu+)+2(\nu+1-\beta-\delta)F(m+)\right)\\ +(n+1)(m+\delta+1)\left((-n+m+\rho+\delta-\mu-\nu-2)F(n+,\nu+)+2(\nu+1-\beta-\delta)F(n+)\right)\\ +(n+\beta+1)(m+\delta+1)\left((n+\alpha+\beta+m+\rho+\delta-\mu-\nu)F(\nu+)+2(\nu+1-\beta-\delta)F\right).\end{aligned} \tag{2.41}$$

Proof. Again start with recursion (2.26), i.e.

$$\begin{aligned}&(n+m+\mu+\nu+4)F(n+,m+,\nu+)-2(\nu+1)F(n+,m+)\\ &=(\alpha+n+1)F(m+,\beta+,\nu+)+(\rho+m+1)F(n+,\delta+,\nu+).\end{aligned}$$

Now add $2(\beta+\delta)F(n+,m+)$ to both sides and multiply by the factor $(n+1)(m+1)$ on both sides. Thus,

$$\begin{aligned}(n+1)(m+1)\left((n+m+\mu+\nu+4)F(n+,m+,\nu+)-2(\nu+1-\beta-\delta)F(n+,m+)\right)\\ =&(n+1)(m+1)\big((\alpha+n+1)F(m+,\beta+,\nu+)\\ &+(\rho+m+1)F(n+,\delta+,\nu+)+2(\beta+\delta)F(n+,m+)\big)\\ =&\,\mathrm{RHS}.\end{aligned}$$

Instead of multiplying with 1, as in the proof for (2.35), we will add a 0 to expand the RHS. Hence,

$$\begin{aligned}\mathrm{RHS}=&(n+1)(m+1)\Big((n+\alpha+\beta+1)F(m+,\beta+,\nu+)+(m+\rho+\delta+1)F(n+,\delta+,\nu+)\\ &-(\beta F(m+,\beta+,\nu+)+\delta F(n+,\delta+,\nu+))+2(\beta+\delta)F(n+,m+)\Big)\\ =&(n+\beta+1)(m+1)(n+\alpha+\beta+1)F(m+,\beta+,\nu+)+(n+1)(m+\delta+1)(m+\rho+\delta+1)F(n+,\delta+,\nu+)\\ &-\beta(m+1)(n+\alpha+\beta+1)F(m+,\beta+,\nu+)-\delta(n+1)(m+\rho+\delta+1)F(n+,\delta+,\nu+)\\ &-\beta(n+1)(m+1)F(m+,\beta+,\nu+)-\delta(n+1)(m+1)F(n+,\delta+,\nu+)\\ &+2(n+1)(m+1)(\beta+\delta)F(n+,m+).\end{aligned}$$

After adding up the additional terms, recurrence relation (2.21) can be used. This gives

$$\begin{aligned}\mathrm{RHS}=&(n+\beta+1)(m+1)(n+\alpha+\beta+1)F(m+,\beta+,\nu+)+(n+1)(m+\delta+1)(m+\rho+\delta+1)F(n+,\delta+,\nu+)\\ &-2\beta(m+1)\left((n+1)F(n+,m+)+(n+\beta+1)F(m+)\right)\\ &-2\delta(n+1)((m+1)F(n+,m+)+(m+\delta+1)F(n+))+2(n+1)(m+1)(\beta+\delta)F(n+,m+1)\\ =&(n+\beta+1)(m+1)(n+\alpha+\beta+1)F(m+,\beta+,\nu+)+(n+1)(m+\delta+1)(m+\rho+\delta+1)F(n+,\delta,\nu+)\\ &-2\beta(m+1)(n+\beta+1)F(m+)-2\delta(n+1)(m+\delta+1)F(n+).\end{aligned}$$

Now use the mixed relation (2.27)

$$
\begin{aligned}
\text{RHS} =&(n+\beta+1)(m+1)\\
&[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(m+,\nu+)+2(\nu+1)F(m+)+(m+\rho+1)F(\delta+,\nu+)]\\
&+(n+1)(m+\delta+1)\\
&[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(n+,\nu+)+2(\nu+1)F(n+)+(n+\alpha+1)F(\beta+,\nu+)]\\
&-2\beta(m+1)(n+\beta+1)F(m+)-2\delta(n+1)(m+\delta+1)F(n+)\\
=&(n+\beta+1)(m+1)\\
&[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(m+,\nu+)+2(\nu+1-\beta-\delta)F(m+)+(m+\rho+1)F(\delta+,\nu+)]\\
&+(n+1)(m+\delta+1)\\
&[(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F(n+,\nu+)+2(\nu+1-\beta-\delta)F(n+)+(n+\alpha+1)F(\beta+,\nu+)]\\
&+2\delta(n+\beta+1)(m+1)F(m+)+2\beta(n+1)(m+\delta+1)F(n+).
\end{aligned}
$$

Consider only a part of the RHS to shorten the notation. Begin by transforming $F(n+)$ or $F(m+)$ back to the form $F(\beta+,\nu+)$ or $F(\delta+,\nu+)$ by equation (2.21), i.e.

$$
\begin{aligned}
&(n+\beta+1)(m+1)(m+\rho+1)F(\delta+,\nu+)+(n+1)(m+\delta+1)(n+\alpha+1)F(\beta+,\nu+)\\
&\quad+2\delta(n+\beta+1)(m+1)F(m+)+2\beta(n+1)(m+\delta+1)F(n+)\\
&\quad=(n+\beta+1)(m+1)(m+\rho+1)F(\delta+,\nu+)+(n+1)(m+\delta+1)(n+\alpha+1)F(\beta+,\nu+)\\
&\quad+\delta(n+\beta+1)(2m+\rho+\delta+2)F(\delta+,\nu+)+\beta(m+\delta+1)(2n+\alpha+\beta+2)F(\beta+,\nu+)\\
&\quad+2\delta(n+\beta+1)(m+\delta+1)F+2\beta(n+\beta+1)(m+\delta+1)F\\
&\quad=(m+1)(n+\beta+1)(m+\rho+\delta+1)F(\delta+,\nu+)+(n+1)(m+\delta+1)(n+\alpha+\beta+1)F(\beta+,\nu+)\\
&\quad+\delta(n+\beta+1)(m+\rho+\delta+1)F(\delta+,\nu+)+\beta(m+\delta+1)(n+\alpha+\beta+1)F(\beta+,\nu+)\\
&\quad+2\beta(n+\beta+1)(m+\delta+1)F\\
&\quad=(m+\delta+1)(n+\beta+1)(m+\rho+\delta+1)F(\delta+,\nu+)+(n+\beta+1)(m+\delta+1)(n+\alpha+\beta+1)F(\beta+,\nu+)\\
&\quad+2\beta(n+\beta+1)(m+\delta+1)F.
\end{aligned}
$$

In the last step use again (2.27), then the claim follows. □

Setting $\nu=-1$, $\beta=\delta=0$ and $\alpha=\rho$ in (2.41) annihilates the coefficient $(\nu+1-\beta-\delta)$ and thus reduces (2.41) to

$$
\begin{aligned}
(n+1)(m+1)\left((n+m+3)F(n+,m+)\right)=&(n+1)(m+1)\left((n+\alpha-m-1)F(m+)\right)\\
&+(n+1)(m+1)\left((-n+m+\alpha-1)F(n+)\right)\\
&+(n+1)(m+1)\left((n+m+2\alpha+1)F\right),
\end{aligned}
$$

which is again conjecture 1.

Corollary 2.2.10
If $\nu + 1 = \beta + \delta$ equation (2.41) reduces to

$$\begin{aligned}(n+1)(m+1)\left((n+m+\mu+\nu+4)F(n+,m+,\nu+)\right)\\ =(n+\beta+1)(m+1)\left((n+\alpha+\beta-m-\mu-\nu-2)F(m+,\nu+)\right)\\ +(n+1)(m+\delta+1)\left((-n+m+\rho+\delta-\mu-\nu-2)F(n+,\nu+)\right)\\ +(n+\beta+1)(m+\delta+1)\left((n+\alpha+m+\rho-\mu-1)F(\nu+)\right).\end{aligned} \tag{2.42}$$

Remark 2
The case $\nu + 1 = \beta + \delta$ corresponds to the integrated Jacobi polynomials $\widehat{P}_n^{\alpha}(x) = \frac{1-x}{n} P_{n-1}^{(\alpha-1,1)}(x)$, where $\beta = \delta = 1, \nu = 1, n \geq 2, \alpha > 1$.

2.3. Recurrence relations for the integral of three Jacobi polynomials

An integral of the product of three Jacobi polynomials appear in different topics. In the numerical application, these kinds of integrals appear in the context of non-constant material functions, e.g. in section 5.4. They also appear in the convection term of the Navier-Stokes equations. Furthermore, from a theoretical viewpoint, those integrals are needed in the linearization of products of Jacobi polynomials.

Linearization of products of Jacobi polynomials

The problem of linearizing products of Jacobi polynomials has been studied intensively in the last century. The classical results e.g. by Gasper [Gas70], Hylleraas [Hyl62], Koornwinder [Koo78] and Rahman [Rah81] were extended e.g. in [Tch14] or [AE15]. The list of works regarding linearization of products of orthogonal polynomials is far from being complete.
Using symbolic packages, we are able to contribute some new recursive relations for the linearization coefficients, which helps to make the linearization more feasible. Those recurrence relations will find application in section 5.4 although it is not directly connected to the linearization problem. Another approach using symbolic computation can be found in [CK10], where the product of two Jacobi polynomials for a certain set of parameters is represented as Kampé de Fériet series.
The classical linearization problem as e.g. in [Gas70] reads as follows

Problem 2.3.1
Find the coefficients $g(k, m, n; \alpha, \beta)$ in the expansion formulation

$$P_n^{(\alpha,\beta)}(x)P_m^{(\alpha,\beta)}(x) = \sum_{k=|n-m|}^{n+m} g(k, m, n; \alpha, \beta)P_k^{(\alpha,\beta)}(x). \tag{2.43}$$

This problem is easily solved. Multiply each side by $(1-x)^{\alpha}(1+x)^{\beta}P_k^{(\alpha,\beta)}(x)$ and integrate from -1 to 1. Then the coefficients $g(k, m, n; \alpha, \beta)$ can each be computed by the integral of a product of three Jacobi polynomials. If we replace in (2.43) the Jacobi polynomials by their (at $x = 1$) normalized version $R_n^{(\alpha,\beta)}(x) = \frac{P_n^{(\alpha,\beta)}(x)}{P_n^{(\alpha,\beta)}(1)}$, the presentation simplifies, see e.g. [Rah81]. Thus,

$$g(k,m,n;\alpha,\beta) = \frac{2^{-(\alpha+\beta+1)}\Gamma(\alpha+\beta+1)(\alpha+1)_k(\alpha+\beta+1)_k(2k+\alpha+\beta+1)}{\Gamma(\alpha+1)\Gamma(\beta+1)k!\,(\beta+1)_k} F(k,m,n,\alpha,\beta),$$

where

$$F(k,m,n,\alpha,\beta) = \int_{-1}^{1} (1-x)^{\alpha}(1+x)^{\beta} R_n^{(\alpha,\beta)}(x) R_m^{(\alpha,\beta)}(x) R_k^{(\alpha,\beta)}(x)\,\mathrm{d}x.$$

We are now interested in finding some recurrence relations for F. As we have seen in the case of theorem 2.2.9 finding linear recurrence relations independent of α, β, works only for special cases. We again denote a parameter raised by unity as before, e.g. $F(n+)$. We find the following corollaries again by application of `Guess`.

Corollary 2.3.1
Let $\beta = 0$, then the following recurrence relation holds

$$\begin{aligned} c_0 F(\alpha+,n+,m+,k+) = {} & (k-m-n-2)F(n+,m+) - (k-m+n+2)F(n+,k+) \\ & + (\alpha+k+m-n)F(n+) - (k+m-n+2)F(m+,k+) \\ & + (k-m+n+\alpha)F(m+) - (k-m-n-\alpha)F(k+) + (k+m+n+2\alpha+2)F, \end{aligned}$$

where $c_0 = (4+\alpha+k+m+n)$.

A quadratic recurrence relation is given for the integrated Jacobi polynomials. We recall that $\widehat{P}_n^{(\alpha,0)}(x) = \frac{(1+x)}{n} P_{n-1}^{(\alpha-1,1)}(x)$. Then we search for $\widetilde{g}(k,m,n;\alpha,\beta)$ such that

$$\widehat{P}_n^{(\alpha,\beta)}(x)\widehat{P}_m^{(\alpha,\beta)}(x) = \sum_{k=|n-m|}^{n+m} \widetilde{g}(k,m,n;\alpha,\beta)\widehat{P}_k^{(\alpha,\beta)}(x).$$

Instead of multiplying both sides with an integrated Jacobi polynomial, we multiply both sides with the weighted Jacobi polynomial $(1-x)^{\alpha-1}P_{k-1}^{(\alpha-1,1)}$. Now we again have orthogonality on the right-hand side such that

$$\widetilde{g}(k,m,n;\alpha,\beta) = \left(\int_{-1}^{1}(1-x)^{\alpha-1}(1+x)P_{k-1}^{(\alpha-1,1)}(x)P_{k-1}^{(\alpha-1,1)}(x)\,\mathrm{d}x\right)^{-1}\widetilde{F}(k,m,n,\alpha)$$

where

$$\widetilde{F}(k,m,n,\alpha) = \int_{-1}^{1}(1-x)^{\alpha-1}(1+x)\widehat{P}_n^{(\alpha,0)}(x)\widehat{P}_m^{(\alpha,0)}(x)P_{k-1}^{(\alpha-1,1)}(x)\,\mathrm{d}x. \tag{2.44}$$

This method of multiplication with the right *dual* functions will be heavily applied in section 6.1.

Corollary 2.3.2
The following recurrence relation holds for $\widetilde{F}(k,m,n,\alpha)$ as in (2.44)

$$\begin{aligned} \widetilde{c_0}\widetilde{F}(n+,m+,k+) = {} & (\alpha+k)(k-m-n-2)\widetilde{F}(n+,m+) - (1+\alpha+k)(k-m+n+3)\widetilde{F}(n+,k+) \\ & + (\alpha+k)(k+m-n+\alpha-2)\widetilde{F}(n+) - (1+\alpha+k)(3+k+m-n)\widetilde{F}(m+,k+) \\ & + (\alpha+k)(k-m+n+\alpha-2)\widetilde{F}(m+) \\ & + (1+\alpha+k)(m+n-k+\alpha-3)\widetilde{F}(k+) + (\alpha+k)(m+n+k+2\alpha-2)\widetilde{F}, \end{aligned}$$

where $\widetilde{c_0} = (1+\alpha+k)(3+\alpha+k+m+n)$.

3. High order finite element method

3.1. Sobolev Spaces

Let Ω be a polynomial Lipschitz domain in $\mathbb{R}^d$. As usual we denote the Lebesgue-integral of f by

$$\int_\Omega f(x)\,\mathrm{d}x.$$

We define

$$L^2(\Omega) \coloneqq \{f : \|f\|_{L^2(\Omega)} < \infty\},$$

where

$$\|f\|_{L^2(\Omega)} \coloneqq \left(\int_\Omega |f(x)|^2\,\mathrm{d}x\right)^{\frac{1}{2}}.$$

We denote by $D^\alpha = \left(\frac{\partial}{\partial x_1}\right)^{\alpha_1} \dots \left(\frac{\partial}{\partial x_n}\right)^{\alpha_n}$ the weak multivariate derivative, where $|\alpha| = \sum_{i=1}^n \alpha_i$. Using the weak derivative we generalize the L^2-norm to the space which include derivatives.

Definition 3.1.1 (Sobolev space)
For $k \in \mathbb{N}$ the space

$$H^k(\Omega) \coloneqq \{f \in L^2(\Omega) : \|f\|_{H^k(\Omega)} < \infty\}$$

is called **Sobolev space** with the **Sobolev norm**

$$\|f\|_{H^k(\Omega)} \coloneqq \left(\sum_{|\alpha|\leq k} \|D^\alpha\|^2_{L^2(\Omega)}\,\mathrm{d}x\right)^{\frac{1}{2}}$$

Indeed it can be shown that the Sobolev norms are actual norms and that $H^k(\Omega)$ is a Hilbert space. One usually defines the corresponding semi-norm for technical reasons as follows

$$|f|_{H^k(\Omega)} = \left(\sum_{|\alpha|=k} \|D^\alpha f\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}},$$

where $f \in H^k(\Omega)$. Before we introduce the relevant function spaces, let us recall the relevant differential operators.
The gradient operator ∇ is defined for a scalar function $v : \Omega \to \mathbb{R}$ as

$$\nabla v = \left(\frac{\partial v}{\partial x_1}, \dots, \frac{\partial v}{\partial x_d}\right)^\top,$$

where $\frac{\partial}{\partial_{x_i}}$ denotes the weak derivatives.

For vector valued functions $v : \Omega \to \mathbb{R}^d$ we define the divergence as

$$\operatorname{div} v = \nabla \cdot v := \frac{\partial v_1}{\partial x_1} + \cdots + \frac{\partial v_d}{\partial x_d}.$$

Furthermore, we define the curl-operator, which has different definitions for $\mathbb{R}^2$ and $\mathbb{R}^3$. Moreover, it has 2 definitions for $\mathbb{R}^2$. Let $\nu : \Omega \to \mathbb{R}$ then the vector Curl maps a scalar onto a vector, i.e.

$$\operatorname{Curl} \nu := \left(\frac{\partial \nu}{\partial x_2}, -\frac{\partial \nu}{\partial x_1} \right)^\top .$$

The scalar curl operator maps vectorial functions $v : \Omega \to \mathbb{R}^2$ onto scalar functions, i.e.

$$\operatorname{curl} v = \nabla \times v := \frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2}.$$

Both definitions of the curl operator are just special cases of the curl operator in $\mathbb{R}^3$. It is defined for a vector $v : \Omega \to \mathbb{R}^3$ as

$$\operatorname{curl} v = \nabla \times v := \left(\frac{\partial v_3}{\partial x_2} - \frac{\partial v_2}{\partial x_3}, \frac{\partial v_1}{\partial x_3} - \frac{\partial v_3}{\partial x_1}, \frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2} \right)^\top ,$$

where we get the scalar two-dimensional curl operator by using the vector field $E = (e_1, e_2, 0)$ and the vector curl operator by using the vector field $E = (0, 0, e_3)$.
The relevant functional spaces are provided in the next definition.

Definition 3.1.2 (Important Spaces)
Let Ω be a non-empty subset in $\mathbb{R}^d$, where $d = 2, 3$.

$$\begin{aligned}
L^2(\Omega) &= \{u : \int_\Omega |u|^2 \, \mathrm{d}x \leq \infty\}, \\
H^1(\Omega) &= \{u \in L^2(\Omega) : \nabla u \in (L^2(\Omega))^d\}, \\
H(\operatorname{div}, \Omega) &= \{\vec{u} \in (L^2(\Omega))^d : \operatorname{div} \vec{u} \in L^2(\Omega)\}, \\
H(\operatorname{curl}, \Omega) &= \{\vec{u} \in (L^2(\Omega))^2 : \operatorname{curl} \vec{u} \in L^2(\Omega)\} && \text{if } \Omega \subset \mathbb{R}^2, \\
H(\operatorname{curl}, \Omega) &= \{\vec{u} \in (L^2(\Omega))^3 : \operatorname{curl} \vec{u} \in (L^2(\Omega))^3\} && \text{if } \Omega \subset \mathbb{R}^3,
\end{aligned}$$

and define the corresponding inner products as

$$\begin{aligned}
\langle u, v \rangle_0 &= \int_\Omega uv \, \mathrm{d}x, \\
\langle u, v \rangle_1 &= \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x + \langle u, v \rangle_0, \\
\langle \vec{u}, \vec{v} \rangle_{\operatorname{div}} &= \int_\Omega \operatorname{div} \vec{u} \cdot \operatorname{div} \vec{v} \, \mathrm{d}x + \int_\Omega \vec{u} \cdot \vec{v} \, \mathrm{d}x, \\
\langle \vec{u}, \vec{v} \rangle_{\operatorname{curl}} &= \int_\Omega \operatorname{curl} \vec{u} \cdot \operatorname{curl} \vec{v} \, \mathrm{d}x + \int_\Omega \vec{u} \cdot \vec{v} \, \mathrm{d}x.
\end{aligned}$$

We denote the induced norms by $\|\cdot\|_0$, $\|\cdot\|_1$, $\|\cdot\|_{\operatorname{div}}$ and $\|\cdot\|_{\operatorname{curl}}$ respectively.

Let Ω be a simple connected Lipschitz domain[1], then this function spaces are related. This relation can be written short in the *De-Rham* complex (3.1), which summarizes the relation between the differential operator.
It forms an exact sequence and reads as follows,

$$\mathbb{R} \overset{id}{\to} H^1(\Omega) \setminus \mathbb{R} \overset{\nabla}{\to} H(\mathrm{curl}, \Omega) \overset{curl}{\to} H(\mathrm{div}, \Omega) \overset{div}{\to} L^2(\Omega) \overset{0}{\to} \{0\}. \tag{3.1}$$

This relation can be easily seen. Assume that $p \in H^1(\Omega)$, then $\nabla p \in H(\mathrm{curl}, \Omega)$, since $\mathrm{curl}(\nabla p) = 0 \in \left(L^2(\Omega)\right)^3$ and $\nabla p \in \left(L^2(\Omega)\right)^3$ per definition. Analogously, if $u \in H(\mathrm{curl}, \Omega)$, then $\mathrm{curl}(u) \in H(\mathrm{div}, \Omega)$.
To incorporate boundary conditions, we need the trace of a function v on the boundary of the domain. If $v \in C^\infty(\overline{\Omega})$ it can be evaluated pointwise, but if v is an element of an arbitrary functional space, e.g. $L^2(\Omega)$, this is not reasonable, since v is not a single function any more. It is an equivalence class of functions, which may differ from each other on a subset of measure zero. By the Gauß and the trace theorem, we can extend the restriction of $v|_{\partial\Omega}$ uniquely to the whole domain by the so-called *trace operator.* For details, see e.g. [AV96, CDZ00, Mon03].
We define the space with essential boundary conditions as

$$H^1_0(\Omega) := \{u \in H^1(\Omega) : \mathrm{tr}_{\partial\Omega}(u) = 0\},$$

where $\mathrm{tr}|_{\partial\Omega}$ is the trace operator on the boundary of the domain. Partial boundary condition are denoted by

$$H^1_{0;\Gamma_D}(\Omega) := \{u \in H^1(\Omega) : \mathrm{tr}_{\Gamma_D}(u) = 0\}.$$

With different trace operators for $H(\mathrm{curl}, \Omega)$ and $H(\mathrm{div}, \Omega)$ we can define the spaces

$$\begin{aligned} H_0(\mathrm{div}, \Omega) &:= \{u \in H(\mathrm{div}, \Omega) : \mathrm{tr}^{(\mathrm{div})}_{\partial\Omega}(u) = 0\}, \\ &= \{u \in H(\mathrm{div}, \Omega) : u \cdot \vec{n} = 0\}, \end{aligned}$$

and

$$\begin{aligned} H_0(\mathrm{curl}, \Omega) &:= \{u \in H(\mathrm{curl}, \Omega) : \mathrm{tr}^{(\mathrm{curl})}_{\partial\Omega}(u) = 0\}, \\ &= \{u \in H(\mathrm{curl}, \Omega) : u \times \vec{n} = 0\}, \end{aligned}$$

where $\vec{n}$ denotes the outward normal vector. Similar relations to (3.1) can be achieved for the spaces with essential boundary conditions, i.e.

$$H^1_0(\Omega) \overset{\nabla}{\to} H_0(\mathrm{curl}, \Omega) \overset{curl}{\to} H_0(\mathrm{div}, \Omega) \overset{div}{\to} L^2(\Omega) \setminus \mathbb{R}. \tag{3.2}$$

Important properties of this operator are summarized in the following theorem:

Theorem 3.1.1 (Theorem[a] 7 of [Ces96])
The diagrams (3.1) and (3.2) have the property that the range of one operator is contained in the kernel of the one following it in the sequence. The range space of each operator is a closed subspace of the appropriate kernel with finite codimension.

[a] see also [Mon03]

[1] This assumption can be further relaxed, see e.g. [ABDG98, Mon03].

We refer the readers e.g. to [Mon03] for more details.
In the following, if it is obvious which domain is of interest we will neglect the domain for the sake of brevity, e.g. we just write H_0^1 or $H(\mathrm{div})$.

3.2. Variational Formulation

We follow [Dem06] and start with the geometry. Let Ω be a bounded Lipschitz domain in $\mathbb{R}^d, d = 2, 3$. We denote by $\Gamma_D, \Gamma_N, \Gamma_R$ subsets of the boundary $\partial\Omega$ corresponding to either imposed Dirichlet, Neumann or Robin boundary condition. An arbitrary boundary value problem in the classical formulation reads:

Problem 3.2.1
Find $u(x), x \in \bar{\Omega}$ such that

$$\begin{aligned} -\operatorname{div}(A(x)\nabla u(x)) + \vec{b}(x) \cdot \nabla u(x) + c(x)u(x) &= f(x) && \text{in } \Omega, \\ u &= u_D(x) && \text{on } \Gamma_D, \\ A(x)\nabla u(x) \cdot \vec{n} &= g(x) && \text{on } \Gamma_n, \\ A(x)\nabla u(x) \cdot \vec{n} - \beta(x)u(x) &= g(x) && \text{on } \Gamma_R. \end{aligned} \tag{3.3}$$

Here n is the exterior normal vector. The left-hand side of the first equation denotes a differential operator Lu, with coefficient functions $A(x) \in \mathbb{R}^{d\times d}, \vec{b}(x) \in \mathbb{R}^d$ and $c(x) \in \mathbb{R}$. Those are called *material functions* and for the most parts of this thesis are assumed to be constant or at least piecewise constant. We will discuss the non-constant variant in section 5.4. The functions $f(x), u_D(x), g(x)$ on the right-hand side are called *load data.* If the matrix $A(x)$ is symmetric positive definite for all $x \in \Omega$ the partial differential equation is called *elliptic.* Classical solutions, which means solution $u \in C^2(\Omega) \cap C^1(\bar{\Omega})$, exist if Ω is sufficiently smooth and further assumptions are fulfilled, see [Rud87]. Here, we introduce the variational formulation of such a boundary value problem. Later on, we will use finite element methods to approximate the corresponding weaker solution of the boundary value problem.
The variational form is derived by multiplication with a test function $v(x)$ and integration over the domain Ω, such that

$$-\int_\Omega \operatorname{div}(A(x)\nabla u(x))v(x) + \vec{b}(x) \cdot \nabla u(x)v(x) + c(x)u(x)v(x)\,\mathrm{d}x = \int_\Omega f(x)v(x)\,\mathrm{d}x.$$

We shift one of the differentiations onto the test function $v(x)$ by integration by parts and obtain

$$\begin{aligned} &\int_\Omega A(x)\nabla u(x)\nabla v(x) + \vec{b}(x) \cdot \nabla u(x)v(x) + c(x)u(x)v(x)\,\mathrm{d}x - \int_\Gamma a(x)(\nabla u \cdot n)v(x)\,\mathrm{d}s \\ &\quad = \int_\Omega f(x)v(x)\,\mathrm{d}x. \end{aligned}$$

Neumann and Robin boundary conditions can now be incorporated naturally in this form. On the Dirichlet part of the boundary we have no information on the derivative of the function, thus we only test with functions which are zero on Γ_D, i.e.

$$v(x) = 0 \quad x \in \Gamma_D.$$

Since we integrated over the domain Ω we need that $u(x)$ and $v(x)$ as well as their differentiation is square integrable. Thus, the natural choice for the functional space is $H^1_{0,\Gamma_D}(\Omega)$.

Model problems

Our boundary value problem in variational form reads:

Problem 3.2.2
Find $u \in H^1_{0,\Gamma_D}(\Omega)$ such that

$$\int_\Omega A\nabla u \cdot \nabla v + \vec{b} \cdot \nabla u v + c u v \,\mathrm{d}x + \int_{\Gamma_R} \beta u v \,\mathrm{d}s = \int_\Omega f v \,\mathrm{d}x + \int_{\Gamma_N \cup \Gamma_R} g v \,\mathrm{d}s \quad \forall v \in H^1_{0,\Gamma_D}(\Omega)$$

This problem is called *diffusion convection reaction equation* and models different phenomena in physic, chemistry or biology. For the ease of presentation, we consider homogeneous Dirichlet boundary conditions:

Problem 3.2.3
Find $u \in H^1_0(\Omega)$ such that

$$\int_\Omega A\nabla u \cdot \nabla v + \vec{b} \cdot \nabla u v + c u v \,\mathrm{d}x = \int_\Omega f v \quad \forall v \in H^1_0(\Omega)$$

Similar model problems for $H(\mathrm{div})$ and $H(\mathrm{curl})$ follow from applications e.g. in fluid dynamics, electromagnetics or elasticity. We state them here without derivation and refer to standard literature like [EGK08, GR86, BF91] or [Mon03].
The following $H(\mathrm{curl})$ problem arises in the context of time harmonic Maxwell's equations.

Problem 3.2.4
Find $\vec{u} \in H_0(\mathrm{curl})$ such that

$$\int_\Omega \mathrm{curl}\,\vec{u} \cdot \mathrm{curl}\,\vec{v} + \vec{u} \cdot \vec{v} \,\mathrm{d}x = \int_\Omega \vec{f} \cdot \vec{v} \,\mathrm{d}x \quad \forall \vec{v} \in H_0(\mathrm{curl}). \tag{3.4}$$

Moreover a similar homogeneous model problem for $H(\mathrm{div})$ is given by:

Problem 3.2.5
Find $\vec{u} \in H_0(\mathrm{div})$ such that

$$\int_\Omega \mathrm{div}\,\vec{u}\,\mathrm{div}\,\vec{v} + \vec{u} \cdot \vec{v} \,\mathrm{d}x = \int_\Omega \vec{f} \cdot \vec{v} \,\mathrm{d}x \quad \forall \vec{v} \in H_0(\mathrm{div}). \tag{3.5}$$

3.3. FEM

We will introduce the finite element method following Szabó and Babuška [SB91], see also the monograph by Schwab [Sch98]. The main difference to the classical definition by Ciarlet [Cia78]

lies in the derivation of shape function. While the former approach introduces those shape functions in higher dimensions by geometrical arguments, the latter introduces those with respect to the nodes of the finite elements.
As we have seen in the last section, we typically have a problem of the form:

Problem 3.3.1
Find $u \in X$ such that

$$a(u, v) = F(v) \quad \forall v \in Y.$$

Here X and Y denote some infinite dimensional function spaces, $a(\cdot, \cdot)$ is a bilinear or sesquilinear form and $F(\cdot)$ is a linear form, as e.g. in (3.3), (3.4) or (3.5). Since we work on computer architecture, we have limited resources and can't therefore work with an infinite number of linear independent functions. For the FEM approach (or Galerkin ansatz) we take a finite subspace $S \subset X(\Omega)$ of dimension N. The dimension N is called *number of degrees of freedom*. Since S is finite, we can find a finite number of basis functions $\phi_1, \dots, \phi_N$, which span S. We can now represent every function $\tilde{u} \in S$ uniquely by

$$\tilde{u} = \sum_{i=1}^{N} c_i \phi_i,$$

where c_i are real or complex numbers.

Definition 3.3.1 (Finite element approximation)
Let $\tilde{X} \subset X$ and $\tilde{Y} \subset Y$ have the finite dimension N. We call u_x a finite element approximation to problem 3.3.1, if

$$u_x \in \tilde{X}, \text{ s.t. } a(u_x, v) = F(v) \quad \forall v \in \tilde{Y}.$$

Here $\tilde{X}$ is called **trial space** and $\tilde{Y}$ is called **test space.** Elements of $\tilde{X}$ are called **trial functions** and elements of $\tilde{Y}$ are called **test functions.**

In a general finite element approach, usually the subspace S is constructed by splitting Ω in subdomains. A *mesh* $\mathcal{T}$ is a partition of Ω in non-overlapping subdomains. These subdomains are usually simple geometrical structures, e.g. intervals in $1D$, or quadrilaterals and triangles in $2D$. A mesh is called regular, if two elements T_i and T_j only share a vertex or an element edge. If the mesh consists only out of quadrilaterals, one considers irregular meshes for local refinement as well. This usually means that a vertex is not shared by another adjacent element, for more details see section 6.2.
In $3D$ a mesh consists either out of cubes or tetrahedron. An irregular mesh contains hanging nodes, edges, and faces. If a conforming regular mesh consists of cubes and tetrahedron, it consists of prisms and pyramids as well, see e.g. [FKDN15]. All definitions in $3D$ are defined analogously to the $2D$ case.
Let

$$P_k := \{p(x) | p(x) = \sum_{|\alpha| \leq k} a_\alpha \vec{x}^\alpha, a_\alpha \in \mathbb{R}\},$$

where P_k is called the space of polynomials of total degree k. Additionally we introduce

$$Q_k := \{p(x) | p(x) = \sum_{\alpha_1 \leq k} \dots \sum_{\alpha_i \leq k} a_\alpha x_1^{\alpha_1} \dots x_d^{\alpha_d}, a_\alpha \in \mathbb{R}\},$$

as the space of polynomials with maximal polynomial degree k. Then we define the subspace

$$\mathcal{S}^{p,l}(\Omega, \mathcal{T}) = \{u \in C^{l-1}(\Omega) \colon u|_{\Omega_j} \in P_p, \Omega_j \subset \mathcal{T}\},$$

on simplical mesh, and similar

$$\hat{\mathcal{S}}^{p,l}(\Omega, \mathcal{T}) = \{u \in C^{l-1}(\Omega) \colon u|_{\Omega_j} \in Q_p, \Omega_j \subset \mathcal{T}\},$$

on a quadrilateral or hexahedral mesh. In the following, we will omit the notational difference between these types of meshes.
Different basis for these subspaces can be chosen, depending on the dimension, the triangulation and the kind of boundary value problem. Different explicit choices are given in chapter 4.
By inserting the chosen basis functions into our bilinear form, we can derive a linear system. In most applications one chooses $\tilde{X} = \tilde{Y}$, i.e. then

$$F(\phi_j) = a(u, \phi_j) = \sum_{i=1}^{N} c_i a(\phi_i, \phi_j) = \left[K_{i,j}\right]_{i=1}^{n} \vec{c}, \quad \text{for all } \phi_j \in \mathcal{S}^{p,l}.$$

and equivalently

$$\vec{f} = K\vec{c}.$$

All the choices of $\mathcal{S}^{p,l}$ will be given as local basis functions on a standard reference domain. We will then map from an arbitrary domain to the standard domain, see the next section. To construct a globally continuous solution, we will need to reinforce continuity. This is done in the choice of our basis functions, depending on the function space. Alternatively, this could be done by methods like discontinuous Galerkin methods see e.g. [ABCM02, PE12, HW08] or mortar methods see e.g. [MMP88, Woh01].
After the local assembly, which will be discussed in section 5.3, we need a local to global operation. Meaning, that we need to assemble the local matrices into a global system matrix[2]. First we determine the connectivity of the basis functions at the interfaces depending on the respective functional space, then we assemble those connectivity relations into a Boolean array B. We assemble the global matrix by application of B. Basis functions without connectivity relations, i.e. so-called *interior* or *bubble* functions are usually eliminated before or directly after the assembly routine by static condensation, see e.g. [Guy65, KS13].

3.3.1. Transformations

In finite elements, we usually define so-called standard or *reference* elements and transform between those and an arbitrary finite element in our mesh. Since we work on an arbitrary setting, we still need to ensure that our transformed functions have a well-defined ∇, curl or div operator, see among other authors the monograph by Monk [Mon03]. Assume K is a bounded domain in $\mathbb{R}^3$ and $\hat{K}$ is our reference domain, also a bounded domain in $\mathbb{R}^3$. Let $F_K : \hat{K} \to K$ be a continu-

[2] Although there are also some matrix-free methods, see e.g. [MGS99, EM05, KS13]

ously one to one map.
If we apply F_K onto a function $\hat{u} \in H^1(\hat{K})$, we transform it to a scalar function u on K by

$$u \circ F_k = \hat{u}. \tag{3.6}$$

The gradient is then transformed by the chain rule as

$$\nabla u = (\mathrm{d}F_k)^{-\top} \hat{\nabla} \hat{u},$$

where $\hat{\nabla}$ denotes the gradient, w.r.t the coordinate system on $\hat{K}$, and $\mathrm{d}F_K$ is the Jacobian matrix. See [Cia78] for a proof.
On the other hand, vectorial functions need to be transformed more carefully. Take a function $\hat{v} \in H(\mathrm{curl}, \hat{K})$, which we want to transform to a function $v \in H(\mathrm{curl}, K)$.
Then $\hat{v}$ must be transformed by

$$v \circ F_K = (\mathrm{d}F_K)^{-\top} \hat{v}, \tag{3.7}$$

since ∇u and $\hat{\nabla}\hat{u}$, as in (3.6), are also in $H(\mathrm{curl}, K)$ or $H(\mathrm{curl}, \hat{K})$ respectively. We cite the following results from [Mon03], but these can also be found in the work of [Dub00] and [Coh02].

Lemma 3.3.1 (See [Mon03, Lemma, 3.57])
Let v and $\hat{v}$ be related by (3.7), where $F_k : \hat{K} \to K$ is a continuously differentiable, invertible and surjective mapping. Let $[\mathrm{curl}(v)]$ denote the 3×3 matrix with

$$[\mathrm{curl}(v)]_{i,j} = \frac{\partial v_i}{\partial x_j} - \frac{\partial v_j}{\partial x_i}.$$

Then

$$[\mathrm{curl}(v)] \circ F_K = (\mathrm{d}F_K)^{-\top} [\mathrm{curl}(v)] (\mathrm{d}F_K)^{-1}.$$

This directly leads to the following corollary

Corollary 3.3.2 (See [Mon03, Corollary, 3.58])
Under the conditions of lemma 3.3.1 and let $\hat{v} \in H(\mathrm{curl}, \hat{K})$. If v and $\hat{v}$ are related by (3.7), then $v \in H(\mathrm{curl}, K)$ and

$$\mathrm{curl}(v) \circ F_K = \frac{1}{\det(\mathrm{d}F_K)} \mathrm{d}F_K \hat{\nabla} \times \hat{v}. \tag{3.8}$$

Moreover we can state similar results for the divergence operator.
For a function $\hat{v} \in H(\mathrm{curl}, \hat{K})$ we have $\hat{\nabla} \times \hat{v} \in H(\mathrm{div}, \hat{K})$. Thus a function $\hat{w} \in H(\mathrm{div}, \hat{K})$ is transformed to $w \in H(\mathrm{div}, K)$ by

$$w \circ F_K = \frac{1}{\det(\mathrm{d}F_K)} \mathrm{d}F_K \hat{w}. \tag{3.9}$$

This is again a reasonable transformation. We again cite the following lemma from [Mon03].

Lemma 3.3.3 (See [Mon03, Lemma, 3.59])
If w and $\hat{w}$ are differentiable functions related by (3.9), where $F_K : \hat{K} \to K$ is a continuously differentiable, invertible and surjective mapping. Then

$$\nabla \cdot w = \frac{1}{\det(\mathrm{d}F_K)} \hat{\nabla} \cdot \hat{w}.$$

Thus, if $\hat{w} \in H(\mathrm{div}, \hat{K})$ then $w \in H(\mathrm{div}, K)$.

The reduction and application to $2D$ is straightforward. There is no problem for $\mathrm{Curl}(v) \in \mathbb{R}^2$, but the transformation (3.8) for $\mathrm{curl}(\vec{u}) = (\frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2})$ reduces to

$$\mathrm{curl}(v) = \frac{1}{\det(dF_k)} \mathrm{curl}(\hat{v}).$$

The proofs of all results above can be found, as mentioned, in [Mon03].

3.3.2. Discrete De-Rham complex

For the derivation of finite elements for the Sobolev spaces $H(\mathrm{curl})$ and $H(\mathrm{div})$, the De-Rham complex (3.1) is of upmost importance. But the derivation of the exact discrete spaces is not that easy. Based on the work by Nédélec [Né80, Né86] and the monograph by Monk [Mon03], we will discuss the discrete spaces for $H(\mathrm{curl})$ in more detail, since we will introduce some new tweaks to the high order $H(\mathrm{curl})$ basis functions in chapter 4. For the discrete space for $H(\mathrm{div})$, we refer the reader to standard literature, e.g. [GR86, Mon03, Dem06, EG21]. In $2D$ and in $3D$ this discretization of the $H(\mathrm{curl})$ basis functions leads to both families of the simplicial edge elements of Nédélec [Né80, Né86][3]. The discrete De-Rham complex is given in fig. 3.1, where we denoted by $U \subset H^1(\Omega), V \subset H(\mathrm{curl}, \Omega), W \subset H(\mathrm{div}, \Omega)$ and $Z = L^2(\Omega)$ the suitable subset of our Sobolev spaces. Furthermore, we denote by U_h, V_h, W_h, Z_h the finite element spaces and by π_h, r_h, w_h, P_h the respective interpolation operators.

$$\begin{array}{ccccccc}
H^1(\Omega) & \xrightarrow{\nabla} & H(\mathrm{curl}, \Omega) & \xrightarrow{\nabla\times} & H(\mathrm{div}, \Omega) & \xrightarrow{\nabla\cdot} & L^2(\Omega) \\
\cup & & \cup & & \cup & & \| \\
U & & V & & W & & Z \\
\downarrow \pi_h & & \downarrow r_h & & \downarrow w_h & & \downarrow P_h \\
U_h & \xrightarrow{\nabla} & V_h & \xrightarrow{\nabla\times} & W_h & \xrightarrow{\nabla\cdot} & Z_h
\end{array}$$

Figure 3.1.: Discrete De-Rham complex in $3D$

The commuting diagram fig. 3.1 can be read as follows: E.g. if we take a function $v \in H(\mathrm{curl})$ and apply the curl operator to it, the resulting functions is part of the Sobolev space $H(\mathrm{div})$, furthermore by application of the interpolating operator w_h this function is in the finite element space W_h. Moreover since this is a commuting diagram, we can first interpolate v onto the finite element

[3] For the hexahedral case, only the first family of Nédélecs functions are part of the discrete De-Rham complex, [Mon03, Dem06]

space V_h by r_h and then apply the curl operator to achieve the same finite element function, i.e.

$$\operatorname{curl}(r_h v) = w_h \operatorname{curl}(v).$$

We will not discuss the interpolating operators in any more detail and refer to [Mon03] and references therein.
The div-conforming finite element space W_h is given by Nédélec [Né80] as a three-dimensional extension of the Raviart-Thomas functions [RT77].
For the discrete space V_h we first introduce the following additional polynomial space

$$\tilde{P}_k := \{\tilde{p}(x) | \tilde{p}(x) = \sum_{|\alpha|=k} a_\alpha \vec{x}^\alpha, a_\alpha \in \mathbb{C}\},$$

where $\tilde{P}_k$ is called the space of homogeneous polynomials of total degree exactly k. Furthermore α denotes the usual multi-index.
Furthermore, we introduce the space of homogeneous vector polynomials as

$$S_k = \{\vec{p} \in (\tilde{P}_k)^3 | \vec{x} \cdot \vec{p} = 0\},$$

where the dimension of S_k is $\dim(S_k) = k(k+2)$. Then the Nédélec space of *first kind* is given by

$$R_k := (P_{k-1})^3 \oplus S_k \tag{3.10}$$

An important property of the space R_k is that functions $v \in R_k$ are transformed invariantly by corollary 3.3.2. Thus, we can define the finite element space as

$$V_h := \{v \in H(\operatorname{curl}, \Omega) | v_{|_K} \in R_k \, \forall K \in \mathcal{T}\}. \tag{3.11}$$

It has been shown by Nédélec [Né80, Né86] that this discretization leads to a suboptimal L^2 error estimate of $\mathcal{O}(h^k)$ instead of $\mathcal{O}(h^{k+1})$. This property is to be expected since the dimension of R_k is smaller than $(P_k)^3$. This motivates the choice

$$V_h^{(2)} := \{v \in H(\operatorname{curl}, \Omega) | v_{|_K} \in (P_k)^3 \, \forall K \in \mathcal{T}\}, \tag{3.12}$$

which is called Nédélec space of *second kind.* In Figure 3.2 we see the second discrete De-Rham

$$\begin{array}{ccccccc}
H^1(\Omega) & \xrightarrow{\nabla} & H(\operatorname{curl},\Omega) & \xrightarrow{\nabla\times} & H(\operatorname{div},\Omega) & \xrightarrow{\nabla\cdot} & L^2(\Omega) \\
\cup & & \cup & & \cup & & \| \\
U & & V & & W & & Z \\
\downarrow{\scriptstyle \pi_h} & & \downarrow{\scriptstyle r_h} & & \downarrow{\scriptstyle w_h} & & \downarrow{\scriptstyle P_h} \\
U_h^{(2)} & \xrightarrow{\nabla} & V_h^{(2)} & \xrightarrow{\nabla\times} & W_h^{(2)} & \xrightarrow{\nabla\cdot} & Z_h
\end{array}$$

Figure 3.2.: Second discrete De-Rham complex in $3D$

complex, which incorporates the discrete Nédélec space of second kind, where $W_h^{(2)}$ denotes a second divergence conforming family, which we will not discuss here, but can be found in [Mon03].

The appropriate scalar space is given by

$$U_h^{(2)} = \{p_h \in H^1(\Omega) | p_h|_K \in P_{k+1} \text{ for all K} \in \mathcal{T}\}.$$

The interpolation operators are denoted by the same symbols as before.
Theoretically we could use any suitable basis of $(P_k)^3$, but in the case of a tetrahedral mesh the polynomial degree of our choice would drop by three from the start of the De-Rham complex to the end, i.e.

$$\mathbb{R} \xrightarrow{\text{id}} P_{k+1} \xrightarrow{\nabla} (P_k)^3 \xrightarrow{\text{curl}} (P_{k-1})^3 \xrightarrow{\text{div}} P_{k-2} \rightarrow \{0\},$$

see [DKP+08]. To get the full polynomial space in $H(\text{curl})$ we need to start with polynomials P_{k+1} in our discrete sequence. When considering e.g. Maxwells equations, this drop in polynomial degree causes different approximation orders in the electric and the magnetic field.
On the other hand, we could also enrich the discrete $H(\text{curl})$ space by the decomposition (3.10). This would lead back to the first Nédélec space. To circumvent the missing degrees of freedom, we can use the following Helmholtz decomposition for the first Nédélec space, i.e.

$$(P_k)^3 = R_{k-1} \oplus \nabla \tilde{P}_{k+1}, \tag{3.13}$$

see [Mon03]. Thus, by adding the gradients, we have the right polynomial order. In other words, the Nédélec space of the second kind can be constructed by the first kind and by the addition of the gradients, without touching the discrete H^1 space. Since the standard error estimates, including the appropriate interpolation operators, are done for the standard Nédélec spaces, it is desirable to use this decomposition. Furthermore, the first space is transformational invariant. In chapter 4 we modify the construction by Sabine Zaglmayr [Zag06] to fit back into this framework. Keep in mind that the second Nédélec family on hexahedrons does not fit the exact De-Rham sequence and is thus not discussed here. That the exact sequence can't be ignored has been shown, e.g. in [BCDD06], where Nédélec's second hexahedron leads to non-physical eigenvalues.

3.3.3. hp-fem

We will summarize shortly the theoretical advantage of high order finite element methods, see e.g. [SB91, Sch98, Mel02]. For a practical guide including implementation, see e.g. [Dem06, DKP+08, KS13].
Consider the standard stationary heat equation on a polygonal domain $\Omega \in \mathbb{R}^2$:

Problem 3.3.2
Find $u \in H_0^1(\Omega)$ such that

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \quad \forall v \in H_0^1(\Omega),$$

where $f : \Omega \rightarrow \mathbb{R}$ is sufficiently smooth.

Let u_{ex} be the exact solution. We denote by λ some measure of smoothness of the exact solution, see e.g. [SB91, Sch98, Mel02] for details. If there are no singularities on the boundary of the solution domain or in the solution domain itself, the parameter $\lambda \rightarrow \infty$.
Let u_{fe} denote the finite element solution, then we measure the error of our approximation in the

energy norm, i.e.

$$\|e\|_{E(\Omega)} = \|u_{ex} - u_{fe}\|_{E(\Omega)},$$

where

$$\|u\|_{E(\Omega)} := \sqrt{\frac{1}{2}a(u,u)}.$$

We are usually interested in the convergence of our solution, which means under which conditions and in which manner, does

$$\|e\|_{E(\Omega)} \to 0?$$

In the usual (low order) finite elements, see e.g. [Cia78, Bra13, BS07], such convergences rates are given with respect to the element size and the polynomial order. But this assumes that we have a constant polynomial order on all elements. We will give the convergence rates with respect to the degrees of freedom, such that we can compare these rates with each other.
We distinguish 3 refinement strategies. The first one, the h-refinement, introduces more degrees of freedom by splitting all (or only some) elements in more elements. For triangular grids this can be done e.g. by either the *red-green* refinement or the *newest-vertex-bisection*, see [Bey98].
In this case, the convergence rate for our model problem is given by

$$\|e\|_{E(\Omega)} \leq \frac{k}{N^{\beta}}, \tag{3.14}$$

where k and β are some positive constants, and N is the number of degrees of freedom.
For a uniform refinement, i.e. a refinement of all elements, we can determine

$$\beta = \frac{1}{2}\min(p, \lambda). \tag{3.15}$$

This means the rate of refinement is bounded by the smoothness of the solution and the polynomial order of our elements. It is possible to construct a sequence of meshes such that we are able to drop the parameter λ from (3.15). For this, one needs to develop heavily refined elements towards any singularities, such that the error is almost the same on each element. For such a non-uniform refinement, we can write

$$\beta = \frac{1}{2}p.$$

This can be done e.g. by an error estimator adaptively, see e.g. [Bra13].
The second refinement strategy, the p-refinement, is to keep the mesh constant, and only refine the polynomial degree on each (or some) elements.
If there are no singularities inside the solution domain or the boundary thereof, then the convergence rate is of exponential form, i.e.

$$\|e\|_{E(\Omega)} \leq \frac{k}{\exp(\gamma N^{\theta})}, \tag{3.16}$$

where k, γ and θ are positive constants, $\theta \geq \frac{1}{2}$.
This exponential convergence rate estimate only holds as long as u_{ex} is analytic. If we consider singularities, the convergence rate becomes algebraic again.
In such cases β in (3.14) is

$$\beta = \frac{1}{2}\lambda,$$

if the singularity is in the domain or

$$\beta = \lambda,$$

if the singularity is on a boundary node, [SB91, Yos12].
The third refinement strategy[4], the hp-refinement, is a combination of both. Optimal meshes are refined to the singularities. The polynomial degree of the element is raised the further we are away from the singularity. The reasoning behind this is very simple. We have seen that we can drop the dependency of the smoothness if we refine heavily towards the singularities. Furthermore, with raising distance towards the singularity, we expect that the exact solution is smooth in these parts. See e.g. [GB86a, GB86b, SB91, Sch98, Mel02] for more details.
These relations are collected in table 3.1, where the categories are defined as follows:

- Category A: u_{ex} is analytic everywhere on the domain including the boundaries
- Category B: u_{ex} is analytic everywhere including the boundaries, except a finite number of points.
- Category C: Neither A nor B.

Category	h	p	hp
A	algebraic $\beta = p/2$	exponential $\theta \geq 1/2$	exponential $\theta \geq 1/2$
B	algebraic $\beta = \frac{1}{2}\min(p, \lambda)$	algebraic $\beta = \lambda$	exponential $\theta \geq 1/3$
C	algebraic $\beta > 0$	algebraic $\beta > 0$	algebraic[5]

Table 3.1.: Asymptotic rate of convergence in 2 dimensions for $H^1(\Omega)$, see [SB91, GB86c, Sch98] and also [Yos12]

Similar results can be achieved in $3D$, although now different types of singularities need to be investigated. We refer to [SA96] and references thereof. For the case of singular edges, see [SS18]. For extension to optimal control, see [WW16].
Optimal (or exponential) convergence rates have not been proved yet for the case of $H(\text{curl})$ or $H(\text{div})$ conforming elements, but have been seen in many numerical works, see e.g. [Dem06, DKP+08].
Since the convergence rates depend on the kind of singularity, it is not trivial to define those for $H(\text{curl})$ or $H(\text{div})$ conforming elements, and neither to predict them.
In $2D$ and in the case of $H(\text{curl})$ conforming elements, optimal p-interpolation errors can be found in [BH09]. See also [Né86] and [Mon03] for the classical interpolation error results by Nédélec.
The hp-refinement can be done in adaptive fashion as well. See e.g. [MW01] for a residual error estimator. For a survey of different hp-refinement schemes and their comparison, see [MM14].
In the next section we will discuss different basis functions for all kind of elements in $2D$ and $3D$, including their geometrical properties.

[4] Possible other refinement strategies, like the r-refinement, where mesh points are moved in each step, are not considered.
[5] Faster rates are possible, if one can use the structure of u_{ex}

4. A list of basis functions

Given a functional space, e.g. H^1, and a conforming discrete space W^p of order p, denote by $\mathcal{S}^{p,l}$ the basis of shape functions spanning the discrete space. It is hierarchical if $\mathcal{S}^{p,l} \subset \mathcal{S}^{p+1,l}$. We enrich a hierarchical space by adding more shape functions. For elements of different order p_l at least some polynomials will match. For the p-adaptivity, this is an important feature, where different polynomial orders on neighbouring elements can happen. In the following, we will summarize the compatibility ideas as collected in [FKDN15]. But note that most of this are to be found e.g. in more detail in [SB91, SK95, AC01, BS06, BP07, Zag06, BPZ12, BPZ13b] and computational details can e.g. be found in [KS13, Dem06, vSD04].
It is of upmost importance for a conforming discrete space that the orientation of edge functions on adjacent elements point in the same direction. Broadly speaking, this can be done by a transformation with −1, where global and local orientation does not match, see e.g. [SK95, BS06]. An alternative approach is to include the orientation directly into the shape functions, such functions are called *orientation embedded*, see [GD10, FKDN15].
In the following, we will omit the problem of orientation to not overcomplicate notation.
The notation in the following chapter is as follows: We denote by u a function in H^1, by v a function in $H(\text{curl})$ and by w a function in $H(\text{div})$. With the symbols $\square$, $\triangle$ we denote, that a function is based on a quadrilateral or a triangle. By $\blacksquare$, $\blacktriangle$, we denote a hexahedron or a tetrahedron, respectively. For example, the function $v_{ij}^{\triangle}$ is a $H(\text{curl})$ conforming basis function on a triangle.

4.1. Traces and compatibility

Our shape functions need to fulfil certain conditions for global continuity depending on the functional space. Therefore, each energy space has a different definition of the boundary *trace*. Those read on a polyhedral element:

- The H^1 trace is the functional value at the boundary of the element. It may take values at the vertices, edges and in $3D$ at the faces.
- The trace of $H(\text{curl})$ is the tangential component of the vector valued functions at the element boundary. Only edge and face functions need to be considered, since vertex functions have no concept of trace.
- Traces of the $H(\text{div})$ are the normal component of vector valued functions at the boundary. Although edge traces have a definition in $2D$, they don't exist in $3D$. In $3D$ only face traces are considered.

To keep notation simple, we can embed shape functions in a dimensional hierarchy. This means e.g. the trace of a $2D$ H^1-edge function is a $1D$ H^1-shape functions. Furthermore, the boundary trace needs to be continuous to the neighbouring element. Depending on the spatial dimension, we only consider the following functions in each dimension:

1. 1D: vertex and edge functions,
2. 2D: vertex, edge and face functions,
3. 3D: vertex, edge, face and interior functions.

To fulfil compatibility, shape functions need to have certain basic properties.

Shape functions in $1D$

Vertex functions: The vertex functions need to vanish on the unassociated vertex and take the value 1 at their respective vertex. To guarantee hierarchy in p, they are restricted to the linear case.
Edge functions: Edge functions are zero on both vertices and are polynomials of order p in the interior of the segment. They are also called *bubble* functions.

Shape function in $2D$

Shape function for H^1 in $2D$

Quadrilateral and triangle are the only two $2D$ elements which we discuss. Their boundary is decomposed in vertices and edges.
Vertex functions: Independent of the element type, the vertex functions vanish on all unassociated vertices and edges. The trace of a vertex function on an adjacent edge should again be the $1D$ vertex function. They have a bilinear decay on quadrilaterals and a linear decay for a triangle.
Edge functions: Edge functions are designed such that they vanish on all other edges and unconnected vertices. The trace of an edge function should again be a $1D$ edge function. Furthermore, they should be designed to have a linear decay for the quadrilateral element, though the triangle is more complicated.
Face functions: Face functions vanish on the boundary of the element and are thus naturally compatible in $2D$.

Shape function for $H(\mathrm{curl})$ in $2D$

As mentioned before $H(\mathrm{curl})$-function have no notion of vertex functions, thus only edge and face functions are relevant.
Edge functions: The edge functions must be constructed such that the trace of an edge function vanish on all other edges. Additionally, the trace of the edge function should be a $1D\,L^2$ edge function of order p. Each of the components should decay linearly on the quadrilateral or in the same order as in the H^1-case on the triangle.
Face functions: The face functions have a vanishing tangential trace on all edges.

Shape function for $H(\mathrm{div})$ in $2D$

The $H(\mathrm{div})$ basis functions are just the rotation of the corresponding $H(\mathrm{curl})$ functions.

Shape functions in $3D$

Shape functions for H^1 in $3D$

We will only discuss the hexahedron and the tetrahedron. For the prism and pyramid, see Nigam and Phillips [NP12] or Fuentes et al. [FKDN15]. The hexahedron and tetrahedron have been thoroughly investigated in [KS13, Zag06, BP07].
Vertex function: Again all vertex function vanish on all other unassociated vertices, edges, and faces. Additionally, the trace of a vertex function over an adjoined face, should be a $2D$ H^1 vertex function.
Edge function: Edge functions vanish on all other edges and disjointed faces. The trace of an edge function should be a $2D$ edge function.
Face function: Face functions vanish on all other faces, and the trace of the face function is again a $2D$ face function.
Interior function: Interior functions only takes values in the interior of the element and vanish on the boundary.

Shape functions for $H(\mathrm{curl})$ in $3D$

Edge functions: In $3D$ the tangential trace vanishes on all other edges and disjointed faces. On the associated edge the trace is a $2D$ $H(\mathrm{curl})$ functions, except for some orientation factor.
Face functions: The tangential trace of a face function vanishes on all other faces. The trace on the associated face is exactly a $2D$ function, except for some orientation.
Interior functions: Interior functions have a vanishing trace on all faces.

Shape functions for $H(\mathrm{div})$ in $3D$

Face functions: The normal trace of the face function vanishes on all other faces and the trace on the associated edge is a L^2 face function, except for some orientation.
Interior functions: The interior functions have a vanishing normal component on all faces.

4.2. Collection of shape functions

Instead of defining all those shape functions on an arbitrary element, one usually uses a pullback to a master element. This is usually done by an affine linear transformation, but in the case of $H(\mathrm{curl})$ or $H(\mathrm{div})$ functions one needs to apply a Piola transformation, see section 3.3.1.
In the following, we will collect shape functions on their respective master element. Usually these elements are defined by their barycentric coordinates, see e.g. [Zag06, Dem06, FKDN15]. But this complicates the application of the techniques of section 5.3, thus we will use the non-barycentric variant.
Furthermore, let ∇ denote the gradient as usual, i.e. for a function

$$f(\vec{x}) = \prod_{i=1}^{n} f_i(\vec{x}), \quad n \in \mathbb{N} \tag{4.1}$$

the gradient can be written out as

$$\nabla f = \sum_{k=1}^{n}(\prod_{\substack{i=1\\i\neq k}}^{n} f_i(\vec{x}))\nabla f_k(\vec{x}).$$

In the context of $H(\text{div})$ and $H(\text{curl})$ functions, we apply a slightly modified gradient operator.

Definition 4.2.1
Let $f : \mathbb{R}^d \to \mathbb{R}$ be the product of multiple functions, i.e. $f = \Pi_{i=1}^{n} f_i, n \in \mathbb{N}$. Then, we introduce

$$\mathbb{V}_l(f(\vec{x})) := \sum_{\substack{k=1\\k\neq l}}^{n}(\prod_{\substack{i=1\\i\neq k}}^{n} f_i(\vec{x}))\nabla f_k(\vec{x}) - (\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x}))\nabla f_l(\vec{x}).$$

In short, we defined a differential operator, which is just the gradient except for a sign change at the l-th part of the sum. E.g. let $f = f_1 \cdot f_2$, then $\mathbb{V}_2 f = (\nabla f_1) f_2 - f_1(\nabla f_2)$.

Corollary 4.2.1
For a function $f : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ as in (4.1) holds

$$\text{curl}(\mathbb{V}_l(f(\vec{x}))) = -2\,\text{Curl}(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x}))\nabla f_l(\vec{x})$$

Proof. We rewrite the equation as follows

$$\begin{aligned}
\text{curl}(\mathbb{V}_l(f(\vec{x}))) &= \text{curl}\left(\nabla f(\vec{x}) - 2(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x}))\nabla f_l(\vec{x})\right) = -2\,\text{curl}(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x})\nabla f_l(\vec{x})),\\
&= -2\left(\frac{\mathrm{d}}{\mathrm{d}x_2}\left(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_1}f_l(\vec{x})\right) - \frac{\mathrm{d}}{\mathrm{d}x_1}\left(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_2}f_l(\vec{x})\right)\right)\\
&= -2\left(\frac{\mathrm{d}}{\mathrm{d}x_2}\left(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x})\right)\frac{\mathrm{d}}{\mathrm{d}x_1}f_l(\vec{x}) - \frac{\mathrm{d}}{\mathrm{d}x_1}\left(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x})\right)\frac{\mathrm{d}}{\mathrm{d}x_2}f_l(\vec{x})\right)\\
&= -2\,\text{Curl}(\prod_{\substack{i=1\\i\neq l}}^{n} f_i(\vec{x}))\nabla f_l(\vec{x}).
\end{aligned}$$

□

A similar result can be given for $\mathbb{R}^3$.

Corollary 4.2.2
For a function $f(\vec{x}) : \Omega \subset \mathbb{R}^3 \to \mathbb{R}$ as in (4.1) holds

$$\operatorname{curl}(\mathbb{W}_l(f(\vec{x}))) = -2\operatorname{curl}\left(\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\nabla f_l(\vec{x})\right)$$

Proof. The proof follows as in the 2D-case. □

Remark 3
We can write corollary 4.2.2 out as

$$-2\operatorname{curl}\left(\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\nabla f_l(\vec{x})\right) = \begin{pmatrix} \frac{\mathrm{d}}{\mathrm{d}x_2}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_3}f_l(\vec{x}) - \frac{\mathrm{d}}{\mathrm{d}x_3}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_2}f_l(\vec{x}) \\ \frac{\mathrm{d}}{\mathrm{d}x_3}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_1}f_l(\vec{x}) - \frac{\mathrm{d}}{\mathrm{d}x_1}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_3}f_l(\vec{x}) \\ \frac{\mathrm{d}}{\mathrm{d}x_1}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_2}f_l(\vec{x}) - \frac{\mathrm{d}}{\mathrm{d}x_2}\prod_{\substack{i=1\\ i\neq l}}^{z} f_i(\vec{x})\frac{\mathrm{d}}{\mathrm{d}x_1}f_l(\vec{x}) \end{pmatrix}.$$

4.2.1. H^1 on a segment, quadrilateral, and hexahedron

As mentioned in the beginning of the last section, we define vertex function as functions with a linear decay towards the unassociated vertex, i.e. hat functions. For the edge functions, we choose integrated Legendre polynomials, since the derivatives are Legendre polynomials and the integrated Legendre polynomials are naturally zero at the vertices. See e.g. [SB91].

Table 4.2.1: H^1 element on the segment

Let the master element be $I = (-1, 1)$. We define the affine pair of coordinates by

$$\lambda_0(s) = \frac{1+s}{2}, \qquad \lambda_1(s) = \frac{1-s}{2}.$$

Vertex functions

$$u_1^v(x) = \lambda_0(x), \qquad u_2^v(x) = \lambda_1(x)$$

Edge functions

$$u_i^E(x) = \widehat{L}_i(x), \qquad 2 \le i \le p.$$

Remark 4
For the edge functions x could be replaced by an auxiliary function $\vec{\lambda}_{ab}(x)$ which involves the local ordering from the vertex with number a to the vertex with number b. This simplifies implementations routines, see [GD10, FKDN15] for details.

The sparsity pattern of the associated element mass and stiffness matrix can easily be determined. Recall that

$$(2i-1)\widehat{L}_i(x) = L_i(x) - L_{i-2}(x), \qquad \text{for } i \ge 2 \tag{4.2}$$

Then the following corollary is easily proven, see [SB91].

Corollary 4.2.3
Let $I = (0,1)$ and consider $\vec{u^e} = \left(u_2^e(x), \ldots, u_p^e(x)\right)^\top$. Then

$$M_{i,j}^e = \int_0^1 u_i^e(x) u_j^e(x)\,\mathrm{d}x = 0, \quad \text{if } i \neq j \text{ or } |i-j| \neq 2 \quad \forall i,j \geq 2$$

and

$$K_{i,j}^e = \int_0^1 \frac{\mathrm{d}}{\mathrm{d}x} u_i^e(x) \frac{\mathrm{d}}{\mathrm{d}x} u_j^e(x)\,\mathrm{d}x = 0, \quad \text{if } i \neq j \quad \forall i,j \geq 2.$$

The first part is shown by using (4.2) and the second equation is just the Legendre orthogonality. This construction can be extended to the quadrilateral by applying a tensor product structure. We now define face functions, see table table 4.2.2 for the collected list of functions.

Table 4.2.2: H^1 element on a quadrilateral

Let the master element be $\square = (-1,1)^2$. Define the affine pair:

$$\lambda_0(s) = \frac{1+s}{2}, \quad \lambda_1(s) = \frac{1-s}{2},$$

Vertex functions

$$u_{a,b}^v(x_1,x_2) = \lambda_a(x_1)\lambda_b(x_2) \quad \text{for } a = 0,1 \text{ and } b = 0,1.$$

Edge functions

For $i = 2, \ldots, p$

$$u_{i,c}^{\square,E}(x_1,x_2) = \lambda_c(x_a)\,\widehat{L}_i\,(x_b)\,, \quad \text{for } (a,b) = (1,2),(2,1) \quad c = 0,1$$

Face functions

For $2 \leq i,k \leq p$

$$u_{ik}^{\square}(x_1,x_2) = \widehat{L}_i(x_1)\widehat{L}_k(x_2),$$

To compute the sparsity pattern of the respective shape functions on the master element use again (4.2). We will only state the sparsity pattern of the face functions, but similar corollaries can easily be computed for the edge-edge and the edge-face blocks, see [SB91].

Corollary 4.2.4
Let $\square = (-1,1)^2$ be the master element. For the entries of the mass matrix w.r.t. the face functions, holds

$$M_{i,j} = \int_Q u^{\square}_{i,k}(x_1,x_2) u^{\square}_{j,l}(x_1,x_2)\,\mathrm{d}x_1\,\mathrm{d}x_2 = 0 \quad \text{if either } |i-j| \neq 0,2, \quad \text{or } |k-l| \neq 0,2,$$

for all $i,j,k,l \geq 2$.
Furthermore, let $C \in \mathbb{R}^{2\times 2}$. For the face entries of the element stiffness matrix

$$K_{i,k,j,l} = \int_Q (C\nabla u^{\square}_{i,k}(x_1,x_2)) \cdot \nabla u^{\square}_{j,l}(x_1,x_2)\,\mathrm{d}x_1\,\mathrm{d}x_2$$

holds

$$K_{i,k,j,l} = 0 \quad \text{if either } |i-j| \geq 0,2, \quad \text{or } |k-l| \geq 0,2.$$

We follow the same principle for $3D$ and define the shape functions by a tensorial product.

Table 4.2.3: H^1 element on a hexahedron

Let the master element be $\blacksquare = (-1,1)^3$. Define the affine pair

$$\lambda_0(s) = \frac{1+s}{2}, \quad \lambda_1(s) = \frac{1-s}{2}$$

Vertex functions:

For $i = 1,\dots,8$

$$u^v_i(x_1,x_2,x_3) = \lambda_\alpha(x_1)\lambda_\beta(x_2)\lambda_\gamma(x_3), \quad \alpha,\beta,\gamma = 0,1$$

Edge functions:

For $i = 1,\dots,p-1$

$$u^{\square,E}_{i,\alpha,\beta}(x_1,x_2,x_3) = \lambda_\beta(x_a)\widehat{L}_i(x_b)\lambda_\alpha(x_c), \qquad \alpha,\beta = 0,1, \quad (a,b,c) \in \{(1,2,3),(2,3,1),(3,1,2)\}$$

Face functions:

For $i,j = 2,\dots,p$

$$u^{\square}_{ij,\alpha}(x_1,x_2,x_3) = \lambda_\alpha(x_a)\widehat{L}_i(x_b)\widehat{L}_j(x_c), \qquad \alpha = 0,1, \quad (a,b,c) \in \{(1,2,3),(2,3,1),(3,1,2)\}$$

Interior functions:

For $i,j,k = 2,\dots,p$

$$u^{\blacksquare}_{i,j,k}(x_1,x_2,x_3) = \widehat{L}_i(x_1)\widehat{L}_j(x_2)\widehat{L}_k(x_3)$$

The derivation of the sparsity pattern is analogously to the previous cases.

4.2.2. $H(\mathrm{curl})$ and $H(\mathrm{div})$ on a quadrilateral

As we have seen in the last chapter, we have two different choices for the discretization of $H(\mathrm{curl})$ functions: Functions of Nédélec's first or second family. Since we didn't discuss the respective spaces for the quadrilateral, we will keep the explanation short.
There are two popular choices for hierarchical basis functions based on orthogonal polynomials. The first one is e.g. discussed in the dissertation of Sabine Zaglmayr [Zag06], and corresponds to the second family of Nédélec's functions. The second choice is the orientation embedded version, as presented e.g. by Fuentes et al. [FKDN15], this is purely based on the first family of Nédélec's functions. The advantage of the first choice is that part of the shape functions are curl- or divergence-free and the rest of the stiffness matrix results in an almost diagonal matrix. Since they are part of Nédélec's second family, one needs to be careful regarding the discrete De-Rham complex, see [Mon03, Dem06]. On the other hand, the advantage of the orientation embedded shape functions is the easy handling of local orientation and the dimensional hierarchy. We will focus on the first type of functions, which we will call *sparsity optimized*.
The choice of integrated Legendre polynomials or the gradient thereof goes back to [AC01].

Table 4.2.4: Sparsity optimized $H(\mathrm{curl})$ shape functions on the quadrilateral

Let $\square = (-1,1)^2$ be the master element and $u^v_{a,b}, u^E_{i,c}, u^\square_{ik}$ be the H^1 shape functions on the quadrilateral.
Define the auxiliary function

$$\vec{\mu}_{01}(s) = \lambda_1(s)\nabla\lambda_0(s) - \lambda_0(s)\nabla\lambda_1(s).$$

Edge functions

Lowest-order edge function:

$$v^{\mathcal{N}_0}_{a,b}(x_1,x_2) = \lambda_c(x_b)\vec{\mu}_{01}(x_a), \quad \text{for } (a,b) \in \{(1,2),(2,1)\}, \quad c = 0,1$$

Higher-order edge functions: For $2 \le i \le p$

$$v^{\square,E}_{i,c}(x_1,x_2) = \nabla u^{\square,E}_{i,c}(x_a,x_b) \quad \text{for } (a,b) \in \{(1,2),(2,1)\}, \quad c = 0,1$$

Face functions:

For $2 \le i,j \le p$

$$v^{\square,I}_{ik}(x_1,x_2) = \nabla u^{\square}_{ij}(x_1,x_2)$$
$$v^{\square,II}_{ik}(x_1,x_2) = \mathbb{\nabla}_2 u^{\square}_{ij}(x_1,x_2)$$

For $2 \le i \le p$

$$v^{\square,III}_{i}(x_1,x_2) = \widehat{L}_i(x_2)\nabla x_1, \quad v^{\square,III}_{i+p}(x_1,x_2) = \widehat{L}_i(x_1)\nabla x_2.$$

Note that $\mathbb{W}_2 u_{ik}^{\square} = \left(\nabla \widehat{L}_i(x_1)\right) \widehat{L}_i(x_2) - \widehat{L}_{i+1}(x_1) \left(\nabla \widehat{L}_i(x_2)\right)$.
The sparsity pattern of the blocks in the mass matrix are the same as for the stiffness matrix of the H^1 case on the quadrilateral.
On the other hand the respective stiffness matrix vanishes for $v_i^{\square,E}$ and $v_{ik}^{\square,I}$ since

$$\mathrm{curl}(\nabla(f(\vec{x})) = 0,$$

for any function $f(\vec{x})$ by definition of the differential operators. For the remaining functions v_{ik}^{II} and v_i^{III} the resulting stiffness matrix is a diagonal matrix, since

$$\mathrm{curl}(v_{ik}^{\square,II})(x_1, x_2) = -2L_{i-1}(x_1)L_{j-1}(x_2),$$

due to corollary 4.2.1 and analogously for v_i^{III}.
The $H(\mathrm{div})$ basis functions in $2D$ are just a 90 degree rotated version of the $H(\mathrm{curl})$ functions, i.e. $w_i = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} v_i$.
An alternative set of face-based functions can be given by

$$v_{ij}^{\square,I} = \begin{pmatrix} 0 \\ \widehat{L}_j(x_1)L_i(x_2) \end{pmatrix}$$
$$v_{ij}^{\square,II} = \begin{pmatrix} \widehat{L}_j(x_2)L_i(x_1) \\ 0 \end{pmatrix},$$

which is a special case of the orientation embedded variant by [FKDN15].
Note that the sparsity optimized functions can be interpreted as a linear combination of this alternative set. This fact will play a crucial role in the derivation of dual function in section 6.1.

4.2.3. $H(\mathrm{curl})$ and $H(\mathrm{div})$ on a hexahedron

As in the quadrilateral case, we state the sparsity optimized variant by Zaglmayr [Zag06]. Again, this corresponds to the second family of Nédélec's functions. Similar to the two-dimensional case, the sparsity optimized variant is a linear combination of an alternative simpler set. An important implementational detail is that all edge and face functions can be calculated by already implemented $2D$ functions.
Furthermore, a big advantage of the sparsity optimized variant is that part of the functions are again *curl-free*. We recall that we enriched our discrete $H(\mathrm{curl})$ space by the gradients, as presented by Nédélec. Those gradients are obviously curl-free. A similar behaviour is achieved for the second discrete divergence conforming $H(\mathrm{div})$ space of Nédélec. We enrich the space by the curl of functions from the $H(\mathrm{curl})$ space, those are naturally divergence-free.
This curl or divergence freedom massively reduces the needed degrees of freedom for the element sparsity matrix.

Table 4.2.5: Sparsity optimized $H(\text{curl})$ shape functions on the hexahedron

Let $\blacksquare = (-1,1)^3$ be the master element and $u^v_{a,b}, u^E_i, u^\square_{ik}$ and $u^\blacksquare_{ijk}$ be the H^1 shape functions on the hexahedron, see table 4.2.3.
Define the auxiliary function

$$\vec{\mu}_{01}(s) = \lambda_1(s)\nabla\lambda_0(s) - \lambda_0(s)\nabla\lambda_1(s).$$

Edge functions

Lowest-order edge function:

For $(a,b,c) = (1,2,3),(2,3,1),(3,1,2)$ and $d = 0,1,\ e = 0,1$

$$v^{\mathcal{N}_0}_{a,b} = \lambda_e(x_c)\lambda_d(x_b)\vec{\mu}_{01}(x_a)$$

Higher-order edge functions:

For $2 \leq i \leq p$

$$v^{\square,E}_{i,\alpha,\beta}(x_1,x_2,x_3) = \nabla u^{\square,E}_{i,\alpha,\beta}(x_a,x_b,x_c), \qquad \alpha,\beta = 0,1$$
$$(a,b,c) = \{(1,2,3),(2,3,1),(3,1,2)\}$$

Face functions:

For $2 \leq i,j \leq p$

$$\begin{aligned} v^{\square,I}_{ij,\alpha}(x_1,x_2,x_3) &= \nabla u^\square_{ij,\alpha}(x_a,x_b,x_c), && \text{for } \alpha = 0,1 \\ v^{\square,II}_{ij,\alpha}(x_1,x_2,x_3) &= \mathbb{W}_2 u^\square_{ij,\alpha}(x_a,x_b,x_c), && \text{and } (a,b,c) \in \{(1,2,3),(2,3,1),(3,1,2)\} \end{aligned}$$

For $2 \leq i \leq p$

$$v^{\square,III}_{i,\alpha}(x_1,x_2,x_3) = \widehat{L}_{i+1}(x_a)\lambda_\alpha(x_b)\nabla\vec{\mu}_{01}(x_c), \qquad \text{for } \alpha = 0,1$$
$$\text{and } (a,b,c) = \{(1,2,3),(2,3,1),(3,1,2)\}$$

Interior functions:

For $2 \leq i,j,k \leq p$

$$\begin{aligned} v^{\blacksquare,I}_{ijk}(x_1,x_2,x_3) &= \nabla u^\blacksquare_{ijk}(x_1,x_2,x_3) \\ v^{\blacksquare,II}_{ijk}(x_1,x_2,x_3) &= \mathbb{W}_2 u^\blacksquare_{ijk}(x_1,x_2,x_3) \\ v^{\blacksquare,III}_{ijk}(x_1,x_2,x_3) &= \mathbb{W}_3 u^\blacksquare_{ijk}(x_1,x_2,x_3) \end{aligned}$$

For $2 \leq i,j \leq p$

$$v^{\blacksquare,IV}_{ij}(x_1,x_2,x_3) = \widehat{L}_i(x_a)\widehat{L}_j(x_b)\nabla x_c, \qquad \text{for } (a,b,c) = \{(1,2,3),(2,3,1),(3,2,1)\}$$

For the case of $H(\text{div})$ on the hexahedron we apply similar arguments. Due to the De-Rham-complex, we derive $H(\text{div})$ basis functions by application of the $\text{curl}(\cdot)$ differential operator only to the non-curl-free $H(\text{curl})$ basis functions, since the gradients vanish under the curl operator.

As we have seen for the $H(\text{curl})$ basis functions, we need to fill up our polynomial space to gain the full order of the interpolation error[1]. This is described in the following: The type II basis functions of the $H(\text{curl})$ are given in the sparsity optimized case by

$$v_{ijk}^{\blacksquare,II} = (\nabla \widehat{L}_i(x))\widehat{L}_j(y)\widehat{L}_k(z) - \widehat{L}_i(x)(\nabla \widehat{L}_j(y))\widehat{L}_k(z) + \widehat{L}_i(x)\widehat{L}_j(y)(\nabla \widehat{L}_k(z))$$

If we apply the curl-operator we get

$$\begin{aligned}\text{curl}(v_{ijk}^{\blacksquare,II}) =& \text{curl}(L_{i-1}(x)\widehat{L}_j(y)\widehat{L}_k(z)e_x) - \text{curl}(\widehat{L}_i(x)L_{j-1}(y)\widehat{L}_k(z)e_y) + \text{curl}(\widehat{L}_i(x)\widehat{L}_j(x)L_{k-1}(z)e_z)\\ =& L_{i-1}(x)\widehat{L}_j(y)L_{k-1}(z)e_y - L_{i-1}(x)L_{j-1}(y)\widehat{L}_k(z)e_z + \widehat{L}_i(x)L_{j-1}(y)L_{k-1}(z)e_y\\ &- L_{i-1}(x)L_{j-1}(y)\widehat{L}_k(z)e_z + \widehat{L}_i(x)L_{j-1}(y)L_{k-1}(z)e_x - L_{i-1}(x)\widehat{L}_j(y)L_{k-1}(z)e_y\\ =& 2\,\widehat{L}_i(x)L_{j-1}(y)L_{k-1}(z)e_x - 2\,L_{i-1}(x)L_{j-1}\widehat{L}_k(z)e_z.\end{aligned}$$

Analogously we get the second type of divergence-free functions by

$$\text{curl}(v_{ijk}^{\square,III}) = 2\,L_{i-1}(x)\widehat{L}_j(y)L_{k-1}(z)e_y - 2\,L_{i-1}(x)L_{j-1}(y)\widehat{L}_k(z)e_z.$$

To complete the space we introduce a modified curl operator for a vector function $v : \Omega \to \mathbb{R}^3$, i.e.

$$\widetilde{\text{curl}}(v) := \left(\frac{\partial v_3}{\partial x_2} + \frac{\partial v_2}{\partial x_3}, \frac{\partial v_1}{\partial x_3} + \frac{\partial v_3}{\partial x_1}, \frac{\partial v_2}{\partial x_1} + \frac{\partial v_1}{\partial x_2}\right)^\top.$$

Applied to our *curl-free* functions $v_{ijk}^{\square,I}$ yields

$$\widetilde{\text{curl}}(v_{ijk}^{\square,I}) = \widehat{L}_i(x)L_{j-1}(y)L_{k-1}(z)\,e_x + L_{i-1}(x)\widehat{L}_j(y)L_{k-1}(z)\,e_y + L_{i-1}(x)L_{j-1}(y)\widehat{L}_k(z)\,e_z,$$

which are obviously linear independent of our introduced *divergence-free* functions. We fill the space with the remaining degrees of freedom by setting i, j or k equal to 1 and applying either the curl or the $\widetilde{\text{curl}}$ operator. This corresponds to Nédélec's second family of divergence conforming elements.

[1]For an analogous result to (3.13), see [Mon03]

Table 4.2.6: Sparsity optimized $H(\mathrm{div})$ shape functions on the hexahedron

Let $\blacksquare = (-1,1)^3$ be the master element and $v^{v}_{a,b}, v^{E}_{i}, v^{\square}_{ik}$ and $v^{\blacksquare}_{i,j,k}$ be the $H(\mathrm{curl})$ shape functions on the hexahedron, see table 4.2.5.
Define the auxiliary function

$$\vec{\mu}_{01}(s) = \lambda_1(s)\nabla\lambda_0(s) - \lambda_0(s)\nabla\lambda_1(s).$$

Face functions:

Lowest-order Raviart Thomas $\mathcal{RT}_0$ functions

$$w^{\mathcal{RT}_0}_{a,c}(x_1,x_2,x_3) = -(\nabla\lambda_c(x_a))x_a, \quad \text{for } a = 1,2,3 \text{ and } c = 0,1.$$

Higher-order face functions: For $2 \le i,j \le p$

$$w^{\square,I}_{ij,\alpha}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\square,II}_{ij,\alpha}(x_a,x_b,x_c)\right), \quad (a,b,c) = \{(1,2,3),(2,3,1),(3,1,2)\}$$

For $2 \le i \le p$

$$w^{\square,II}_{i,\alpha}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\square,III}_{i,\alpha}(x_a,x_b,x_c)\right), \qquad (a,b,c) = \{(1,2,3),(2,3,1),(3,1,2)\}$$
$$w^{\square,II}_{i+p,\alpha}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\square,III}_{i,\alpha}(x_a,x_b,x_c)\right), \qquad \alpha = 0,1$$

Interior functions:

Divergence-free functions:
For $2 \le i,j,k \le p$

$$w^{\blacksquare,Ia}_{ijk}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\blacksquare,II}_{ijk}(x_1,x_2,x_3)\right)$$
$$w^{\blacksquare,Ib}_{ijk}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\blacksquare,III}_{ijk}(x_1,x_2,x_3)\right)$$

For $2 \le i,j \le p$

$$w^{\blacksquare,Ic}_{ij}(x_1,x_2,x_3) = \mathrm{curl}\left(v^{\square,IV}_{ij}(x_a,x_b,x_c)\right), \quad \text{for } (a,b,c) = \{(1,2,3),(2,3,1),(3,2,1)\}$$

Non-divergence-free functions:
For $2 \le i,j,k \le p$

$$w^{\blacksquare,IIa}_{ijk}(x_1,x_2,x_3) = \widetilde{\mathrm{curl}}\left(v^{\blacksquare,I}_{ijk}(x_1,x_2,x_3)\right)$$

For $2 \le i,j \le p$

$$w^{\blacksquare,IIb}_{ij}(x_1,x_2,x_3) = \widetilde{\mathrm{curl}}\left(v^{\square,IV}_{ij}(x_a,x_b,x_c)\right), \quad \text{for } (a,b,c) = \{(1,2,3),(2,3,1),(3,2,1)\}$$

For $2 \le i \le p$

$$w^{\blacksquare,III}_{i}(x_1,x_2,x_3) = \widetilde{\mathrm{curl}}\left(v^{\square,III}_{i,0}(x_a,x_b,x_c)\right), \quad \text{for } (a,b,c) = \{(1,2,3),(2,3,1),(3,2,1)\}$$

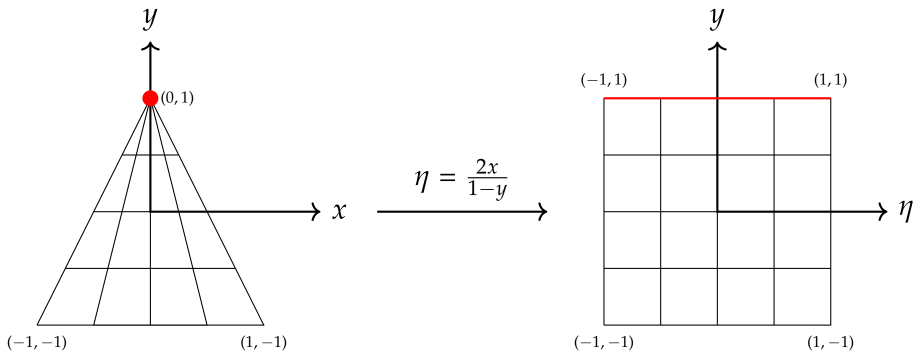


Figure 4.1.: Duffy transformation

4.2.4. H^1 on a triangle

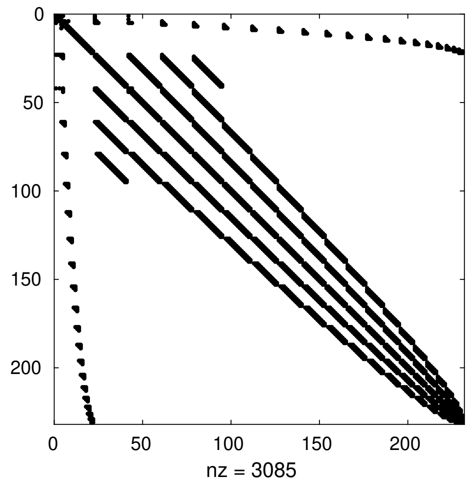
Although a triangle does not inhibit a natural tensorial structure, it has as a similar structure. Following Dubiner [Dub91] and Karniadakis and Sherwin [SK95] we map a triangle to a quadrilateral. This transformation is called Duffy-transformation [Duf82]. That this a good-natured transformation, see e.g. [EM05].

Consider the reference element $\triangle$ with vertices $(-1,-1),(1,-1)$ and $(0,1)$. Then the Duffy transformation is given by

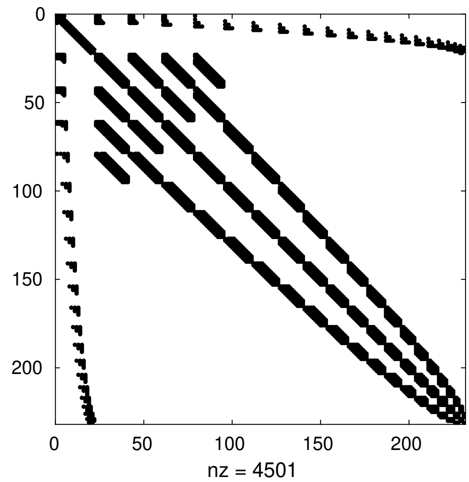
$$\begin{aligned} D_2 : \triangle &\to \square \\ (x,y) &\mapsto (\eta, y) \end{aligned} \tag{4.3}$$

where $\eta = \frac{2x}{1-y}$, see Figure 4.1.

High order L^2 basis functions on a triangle were found first by Dubiner [Dub91] by application of the Duffy transformation. Note that, this basis functions are identical to the classical multivariate orthogonal polynomials found by Proriol [Pro57]. A modification from L^2 to H^1 was found by Karniadakis and Sherwin [SK95]. A variant of this basis was given by Beuchler and Schöberl [BS06]. But it was later shown by Beuchler and Pillwein [BP08] that the basis by Karniadakis and Sherwin is optimal in the sense of sparsity. These functions are given in table 4.2.7.



(a) Mass matrix, $p = 20$



(b) Stiffness matrix, $p = 20$

Figure 4.2.: Sparsity pattern of local triangular element matrix

Table 4.2.7: H^1-basis on a triangle

Let $\triangle$ be the reference triangle with vertices $(-1,-1), (1,-1)$ and $(0,1)$ and edges E_1, E_2, E_3. The barycentric coordinates are then given by

$$\lambda_0(x,y) = \frac{1-2x-y}{4}, \quad \lambda_1(x,y) = \frac{1+2x-y}{4}, \quad \lambda_3(x,y) = \frac{1+y}{2}$$

Vertex functions

$$u_a^{\triangle,v}(x,y) = \lambda_a, \quad a = 0,1,2$$

Edge functions

For $2 \le i \le p$:

$$u_i^{\triangle,E1}(x,y) = \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i$$

$$u_i^{\triangle,E2}(x,y) = \frac{1}{2}\left(1-\frac{2x}{1-y}\right)\widehat{L}_i(y),$$

$$u_i^{\triangle,E3}(x,y) = \frac{1}{2}\left(1+\frac{2x}{1-y}\right)\widehat{L}_i(y)$$

Face functions

For $i \ge 2, j \ge 1$ and $i+j \le p$:

$$u_{ij}^{\triangle,I}(x,y) = \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i \widehat{P}_j^{2i}(y) \qquad (4.4)$$

Note that $\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{2x}{1-y}\right)^i$ is a polynomial of order i. Furthermore $u_i^{\triangle,E_2}$ and $u_i^{\triangle,E3}$ are also polynomials of order i, which can be shown by

$$\left(1 \pm \frac{2x}{1-y}\right)\widehat{L}_i(y) = \left(1 \pm \frac{2x}{1-y}\right)\frac{y-1^2}{2(i-1)}P_{i-2}^{(1,1)} = (y-1 \mp 2x)(1+y)P_{i-2}^{(1,1)}(y).$$

To show that all edges have the same trace insert $\eta = \frac{2x}{1-y} = \pm 1$ or $y = -1$.
An alternative set of edge functions can be given by using barycentric (or alternatively area) coordinates. For an edge $E_k = [\lambda_{e_1}, \lambda_{e_2}]$ those functions are given by

$$u_i^{\triangle,E_k} = \widehat{L}_i\left(\frac{\lambda_{e_2}-\lambda_{e_1}}{\lambda_{e_2}+\lambda_{e_1}}\right)(\lambda_{e_2}+\lambda_{e_1})^i, \qquad (4.5)$$

and are applied e.g. in [FKDN15] and [BPSZ12]. An example of such an implementation can be found in `Ngsolve` [Sch14]. Those edge functions have the advantage that they are simpler to implement, since any change in the edge orientation can be handled by swapping the order of the vertices.

The gradient of $u_{ij}^{\triangle,I}$ is given by

$$\nabla u_{ij}^{\triangle,I}(x,y) = \begin{pmatrix} L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}(y) \\ \frac{1}{2}L_{i-2}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}(y) + \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i} P_{j-1}^{(2i,0)}(y) \end{pmatrix}. \tag{4.6}$$

The first component is straight forward. The second component can be derived by a classical differential recurrence relation ([Rai71] eq. 87.6), i.e.

$$(x^2-1)\frac{\mathrm{d}}{\mathrm{d}x}L_{n-1}(x) = nxL_n(x) - nL_{n-1}(x), \tag{4.7}$$

and thus

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}y}\widehat{L}_j\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i} &= \frac{1}{2}\left(\frac{2x}{1-y}\right)L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1} - \frac{i}{2}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1} \\ &= \frac{1}{2}\eta L_{i-1}(\eta)\left(\frac{1-y}{2}\right)^{i-1} - \frac{\eta^2-1}{2(i-1)}\frac{\mathrm{d}}{\mathrm{d}\eta}L_{i-1}(\eta)\left(\frac{1-y}{2}\right)^{i-1} \\ &= \frac{1}{2}L_{i-2}(\eta)\left(\frac{1-y}{2}\right)^{i-1}, \end{aligned}$$

where $\eta = \left(\frac{2x}{1-y}\right)$, see also [BS06].
First sparsity results were given by Karniadakis and Sherwin in [SK95]. The complete sparsity pattern was proven by Beuchler and Schöberl [BS06], for a slightly suboptimal basis. The sparsity results for the optimal basis were given in [BP08], see also [BPSZ12]. This results can be proven similar to quadrilateral case, but more orthogonality condition need to be checked. Alternatively, one can do this by symbolic software as e.g. described in [BP07].
As in the quadrilateral case we denote by M^I the interior block of the mass matrix, i.e.

$$M_{ij,kl}^I = \left[\int_{\triangle} u_{ij}^{\triangle}(x,y)\cdot u_{kl}^{\triangle}(x,y)\mathrm{d}(x,y)\right]_{ij,kl}, \tag{4.8}$$

and by K^I the interior block of the stiffness matrix, i.e.

$$K_{ij,kl}^I = \left[\int_{\triangle} \nabla u_{ij}^{\triangle}(x,y)\cdot D\nabla u_{kl}^{\triangle}(x,y)\mathrm{d}(x,y)\right]_{ij,kl}, \quad D \in \mathbb{R}^{2\times 2} \text{ and constant.} \tag{4.9}$$

Lemma 4.2.5 (Sparsity pattern on the reference triangle)
Let $u_{ij}^{\triangle}(x,y)$ be defined as in (4.4) for $i \geq 2, j \geq 1, i+j \leq p$. The matrices M^I and K^I have $\mathcal{O}(p^2)$ non-zero entries. More precisely

$$M_{ij,kl}^I = 0 \text{ if } |i-k| > 2 \text{ or } |i+j-k-l| > 3$$

and

$$K_{ij,kl}^I = 0 \text{ if } |i-k| > 2 \text{ or } |i+j-k-l| > 1$$

The proof is based on [BS06] and is given in [BP08]. The sparsity pattern can be seen in Figure 4.2.

4.2.5. $H(\mathrm{curl})$ and $H(\mathrm{div})$ on a triangle

The ideas from the quadrilateral case carry on to the triangular case. We define our basis functions for the $H(\mathrm{curl})$ by the differential operators ∇ and $\mathbb{W}$, see [BPZ12] and [BPZ13b]. Keep in mind that the application of the operator $\mathbb{W}$ does not necessarily yield Nédélec functions of the first kind. A fix for this is presented at the end of the chapter.
As in the quadrilateral case, the sparsity optimized variant has a curl-free part given by the gradient. Again, the functions in $H(\mathrm{div})$ in $2D$ are just the rotated variants of the $H(\mathrm{curl})$ basis functions.

Table 4.2.8: $H(\mathrm{curl})$-basis on a triangle

On the reference triangle $\triangle$ with vertices $(-1,-1), (1,-1)$ and $(0,1)$, and edges E_1, E_2, E_3. Let

$$\lambda_1(x,y) = \frac{1-2x-4}{2},\ \lambda_2(x,y) = \frac{1+2x-y}{4},\ \text{ and } \lambda_3(x,y) = \frac{1+y}{2}$$

Edge functions

Lowest-order functions: For $(a,b) = \{(1,2),(2,3),(3,1)\}$ let

$$v_a^{\triangle,\mathcal{N}_0}(x,y) = \nabla(\lambda_b(x,y))\lambda_a(x,y) - \lambda_b(x,y)\nabla(\lambda_a(x,y))$$

Higher-order edge based functions: For $2 \le i \le p$ and $m = 1,2,3$

$$v_i^{\triangle,E_m}(x,y) = \nabla u_i^{\triangle,E_m}(x,y)$$

Face functions

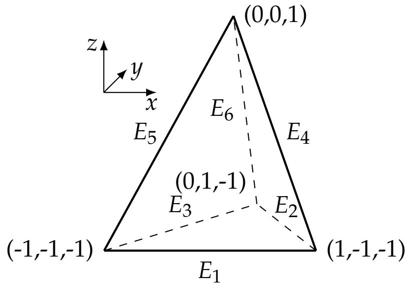
Let

$$f_i(x,y) := \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i$$
$$g_{ij}(y) := \widehat{P}_j^{2i}(y)$$

then for $2 \le i, 1 \le j$ and $i+j \le p$

$$v_{ij}^{\triangle,I}(x,y) = \nabla u_{ij}^{\triangle,I}(x,y) = (\nabla f_i(x,y))g_{ij}(y) + f_i(x,y)\nabla g_{ij}(y)$$
$$v_{ij}^{\triangle,II}(x,y) = \mathbb{W}_2 u_{ij}^{\triangle,I}(x,y) = (\nabla f_i(x,y))g_{ij}(y) - f_i(x,y)\nabla g_{ij}(y)$$
$$v_{1j}^{\triangle,III}(x,y) = v_1^{\triangle,\mathcal{N}_0}(x,y)\widehat{P}_j^3(y)$$

For the sparsity pattern of the $H(\mathrm{curl})$ element mass matrix we apply lemma 4.2.5. It holds that one of the main blocks of the $H(\mathrm{curl})$ mass matrix is the stiffness matrix of the H^1 case. Moreover, note that for the other blocks the change in sign of the differential operator $\mathbb{W}_2$, does not influence the sparsity pattern.
The stiffness matrix has only few non-zero blocks, since $\mathrm{curl}(\nabla(\cdot)) = 0$ per definition. For the curl



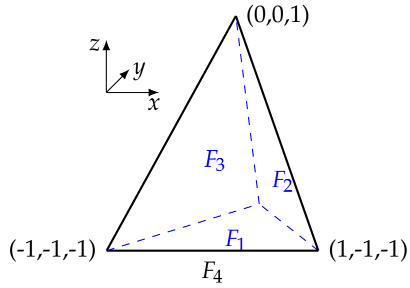


Figure 4.3.: Reference tetrahedron with edge notation on the left and face notation on the right

of type II functions follows with corollary 4.2.1

$$\begin{aligned}
\operatorname{curl}_2 \mathbb{\nabla} u_{ij}^{\triangle} &= -2\operatorname{curl}(g_i(x,y))\nabla h_{ij}(y) \\
&= -2 \begin{pmatrix} L_{i-2}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1} \\ -L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1} \end{pmatrix}^{\top} \begin{pmatrix} 0 \\ P_{j-1}^{(2i,0)}(y) \end{pmatrix} \\
&= 2L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1} P_{j-1}^{(2i,0)}(y).
\end{aligned} \tag{4.10}$$

Thus the remaining non-zero blocks yield a tridiagonal matrix. If we choose $\tilde{h}_j(y) = \widehat{P}_j^{2i-1}(y)$ instead of $h_j(y)$, we get a diagonal matrix. But this yields a worse sparsity pattern of the mass matrix and more importantly we need the choice of at least $2i$ as Jacobi index due to technical reasons in section 6.1.

4.2.6. H^1 on a tetrahedon

Let the reference element tetrahedron ▲ be defined by the vertices $(-1,-1,-1), (1,-1,-1), (0,1,-1)$ and $(0,0,1)$, see Figure 4.3. On the left-hand side we denoted the numeration of edges, while on the right-hand side the face numeration is denoted. Furthermore, integration on the tetrahedron is similar to the triangle. First transform the tetrahedron ▲ to the reference hexahedron ■, then use the tensorial structure to integrate in each coordinate dimension. The Duffy transformation on the tetrahedron, see Figure 4.4, is given by

$$\begin{aligned}
D_3 : \qquad & ▲ \to ■ \\
(x,y,z) &\mapsto (\eta,\chi,z),
\end{aligned} \tag{4.11}$$

where

$$\begin{aligned}
\eta &= \frac{4x}{1-2y-z} \\
\chi &= \frac{2y}{1-z}.
\end{aligned}$$

The functional determinant of the Jacobian of the Duffy transformation is

$$\det(\mathrm{d}D_3) = \frac{1-\chi}{2}\left(\frac{1-z}{2}\right)^2.$$

Again, this needs to be considered for the optimal Jacobi indices. The here used construction of

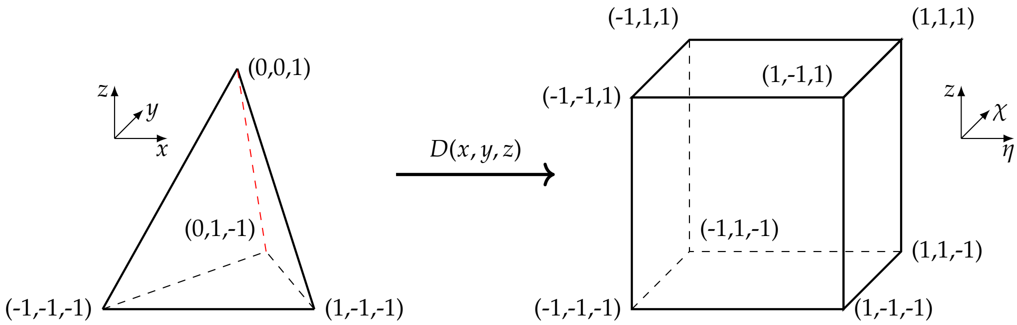


Figure 4.4.: Duffy transformation in $3D$

the H^1 conforming basis was given in [BP07], but see also [KS13]. Note that the definitions in the edge and face functions make sense, since the integrated Legendre polynomials removes the singularities. For example,

$$\left(1 - \frac{2y}{1-z}\right) \widehat{L}_i(z) = \left(1 - \frac{2y}{1-z}\right) \frac{1-z^2}{2(i-1)} P_{i-2}^{(1,1)}(z),$$

and so on.

By using the tensorial structure and properties of the orthogonal polynomial, one can again determine the sparsity pattern. This computation gets very tedious for the stiffness matrix, and is usually done by computer algebra, like `mathematica` [Inc]. The relevant results can be found in [BP07, BP08]. In Figure 4.5 the sparsity pattern of the tetrahedron can be found. Furthermore, in appendix A.2 we collected some partial sparsity results, which will be relevant in section 5.3.

Table 4.2.9: H^1-basis on a tetrahedron

Let $\blacktriangle$ be the reference tetrahedron with vertices $(-1,-1,-1)$, $(1,-1,-1)$, $(0,1,-1)$ and $(0,0,1)$. Denote by $E_1,\dots,E_6$ and $F_1,\dots,F_4$ the edges and the faces, respectively, see Figure 4.3. The barycentric coordinates are then given by

$$\lambda_{0/1}(x,y,z) = \frac{1 \pm 4x - 2y - z}{4}, \quad \lambda_2(x,y,z) = \frac{1+2y-z}{2}, \quad \lambda_3(x,y,z) = \frac{1+z}{2}$$

Vertex functions

$$u_a^{\blacktriangle,v}(x,y,z) = \lambda_a, \quad a = 0,1,2,3$$

Edge functions

For $2 \le i \le p$:

$$u_i^{\blacktriangle,E_1}(x,y,z) = \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i$$

$$u_i^{\blacktriangle,E_{2/3}}(x,y,z) = \frac{1}{2}\left(1 \pm \frac{4x}{1-2y-z}\right)\widehat{L}_i\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^i,$$

$$u_i^{\blacktriangle,E_{4/5}}(x,y,z) = \frac{1}{4}\left(1 \pm \frac{4x}{1-2y-z}\right)\left(1-\frac{2y}{1-z}\right)\widehat{L}_i(z)$$

$$u_i^{\blacktriangle,E_6}(x,y,z) = \frac{1}{2}\left(1+\frac{2y}{1-z}\right)\widehat{L}_i(z)$$

Face functions

For $i \ge 2, j \ge 1$ and $i+j \le p$:

$$u_{ij}^{\blacktriangle,F_1}(x,y,z) = \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i \widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^j$$

$$u_{ij}^{\blacktriangle,F_{2/3}}(x,y,z) = \frac{1}{4}\left(1 \pm \frac{4x}{1-2y-z}\right)\left(1-\frac{2y}{1-z}\right)\widehat{L}_i\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^i \widehat{P}_j^{2i}(z)$$

$$u_{ij}^{\blacktriangle,F_4}(x,y,z) = \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i \widehat{P}_j^{2i}(z)$$

Interior functions

For $i = j+k \le p$ and $i \le 2, j,k \ge 1$:

$$u_{ijk}^{\blacktriangle,I}(x,y,z) = \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i \widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^i \widehat{P}_k^{2i+2j}(z)$$

Again an alternative set of edge and face functions can be defined by the barycentric coordinates, see e.g. [BPSZ12, FKDN15]. Although they are simpler to implement than the given edge functions above, they do not exhibit the same kind of underlying orthogonal structure[2]. This kind of structure will again be relevant in section 5.3.

[2] Additionally, they have a slightly worse element condition number.

On the other hand, for the here presented version of edge and face functions, the inclusion of orientation needs to be done very carefully.

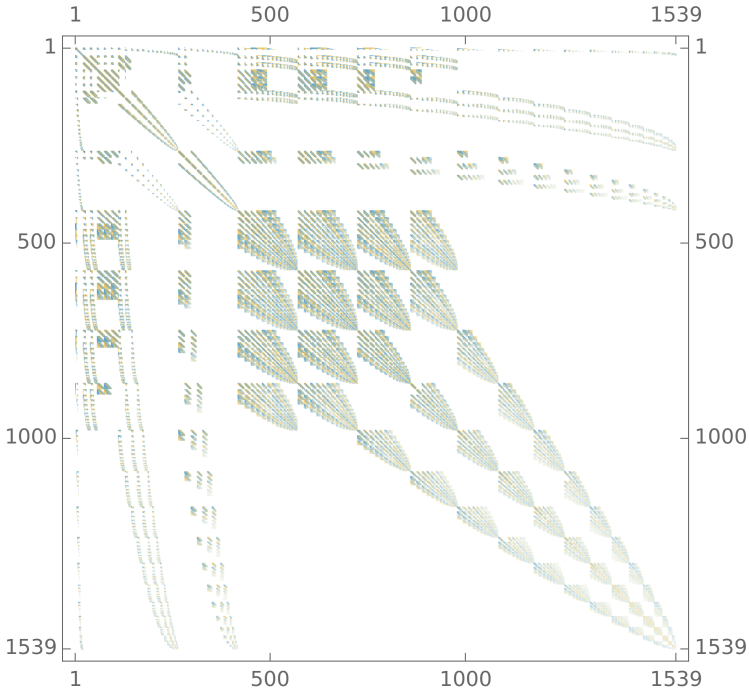


Figure 4.5.: Sparsity pattern of the local tetrahedral mass matrix on the reference element for $p = 19$

4.2.7. $H(\text{curl})$ and $H(\text{div})$ on a tetrahedron

The here used concept of the basis functions for $H(\text{curl})$ and $H(\text{div})$ is based on the work by Zaglmayr [Zag06], see also [BPZ12] and [BPZ13b]. The construction idea is the same as in the hexahedral case. We derive interior functions by application of the differential operators ∇, $\mathbb{W}_2$ and $\mathbb{W}_3$. Face and edge functions follow from the two-dimensional case. Again the operators $\mathbb{W}_2$ and $\mathbb{W}_3$ will not give us Nédélec functions of type I, but we will correct this later on.
The functions are given in table 4.2.10.

Table 4.2.10: $H(\text{curl})$-basis on a tetrahedron

Let ▲ be the reference tetrahedron with vertices $(-1,-1,-1)$, $(1,-1,-1)$, $(0,1,-1)$ and $(0,0,1)$. The barycentric coordinates are then given by

$$\lambda_{0,1}(x,y,z) = \frac{1 \pm 4x - 2y - z}{8}, \quad \lambda_2(x,y,z) = \frac{1+2y-z}{4}, \quad \lambda_3(x,y,z) = \frac{1+z}{2}$$

Edge functions

Let $E_m = [e_1, e_2]$ be the edge with vertices λ_{e_1} and λ_{e_2}.

The lowest order Nédélec functions:

$$v_1^{\blacktriangle,E_m} = \nabla\lambda_{e_1}\lambda_{e_2} - \lambda_{e_1}\nabla\lambda_{e_2}$$

Higher order edge functions:

For $2 \leq i \leq p$:

$$v_i^{\blacktriangle,E_m}(x,y,z) = \nabla u_i^{\blacktriangle,E_m}(x,y,z)$$

Face functions

Let $F_m = [s_1, s_2, s_3]$ be the face with vertices $\lambda_{s_1}, \lambda_{s_2}$ and λ_{s_3}.

Lowest order Nédélec functions:

For $1 \leq j \leq p-1$:

$$v_{1j}^{\blacktriangle,F_m} = (\nabla\lambda_{s_1}\lambda_{s_2} - \lambda_{s_1}\nabla\lambda_{s_2})\,\widehat{P}_j^1(\lambda_{s_3} - \lambda_{s_2} - \lambda_{s_1})$$

Gradient based face functions:

For $i \geq 2, j \geq 1$ and $i + j \leq p$:

$$v_{ij}^{\blacktriangle,F_m,I} = \nabla u_{ij}^{\blacktriangle,F_m}$$

Non-gradient based face functions:

For $i \geq 2, j \geq 1$ and $i + j \leq p$:

$$v_{ij}^{\blacktriangle,F_m,II}(x,y,z) = \mathbb{W}_2 u_{ij}^{\blacktriangle,F_m}(x,y,z)$$

Interior functions

Gradient based interior functions for $i = j + k \leq p$ and $i \leq 2, j, k \geq 1$:

$$v_{ijk}^{\blacktriangle,I}(x,y,z) = \nabla u_{ijk}^{\blacktriangle}(x,y,z)$$

Non-gradient based interior functions

$$\begin{aligned} v_{ijk}^{\blacktriangle,II}(x,y,z) &= \mathbb{W}_2 u_{ijk}^{\blacktriangle,C}(x,y,z) \\ v_{ijk}^{\blacktriangle,III}(x,y,z) &= \mathbb{W}_3 u_{ijk}^{\blacktriangle,C}(x,y,z) \\ v_{1jk}^{\blacktriangle,IV}(x,y,z) &= v_1^{\blacktriangle,E_1}\,\widehat{P}_j^3\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^j \widehat{P}_k^{2j+2}(z) \end{aligned} \tag{4.12}$$

For the $H(\text{div})$ conforming functions, we need to construct the divergence conforming Nédélec space [Né86, Mon03], also called the Raviart-Thomas-Nédélec space. Following the construction of the De-Rham complex, we first construct $H(\text{div})$ conforming functions by application of the $\widetilde{\text{curl}}$ operator and enrich the space with the curl of the non-curl-free $H(\text{curl})$ functions[3]. Those functions are given in table 4.2.11.

Table 4.2.11: $H(\text{div})$-basis on a tetrahedron

Let ▲ be the reference tetrahedron with vertices $(-1,-1,-1), (1,-1,-1), (0,1,-1)$ and $(0,0,1)$. The barycentric coordinates are then given by

$$\lambda_{0,1}(x,y,z) = \frac{1 \pm 4x - 2y - z}{8}, \quad \lambda_2(x,y,z) = \frac{1+2y-z}{4}, \quad \lambda_3(x,y,z) = \frac{1+z}{2}$$

Face functions

Let $F_m = [s_1, s_2, s_3]$ be the face with vertices $\lambda_{s_1}, \lambda_{s_2}$ and λ_{s_3}.

$$w_0^{▲,F_m}(x,y,z) = \lambda_{s_1}\nabla\lambda_{s_2} \times \nabla\lambda_{s_3} + \lambda_{s_2}\nabla\lambda_{s_3} \times \nabla\lambda_{s_1} + \lambda_{s_3}\nabla\lambda_{s_1} \times \nabla\lambda_{s_2}$$
$$w_{1j}^{▲,F_m}(x,y,z) = \text{curl}(v_{1j}^{▲,F_m}(x,y,z))$$
$$w_{ij}^{▲,F_m}(x,y,z) = \text{curl}(v_{ij}^{▲,F_m}(x,y,z))$$

Interior functions

Non-divergence-free functions:

$$w_{ijk}^{▲,I}(x,y,z) = \widetilde{\text{curl}}(v_{ijk}^{▲,I}(x,y,z))$$
$$w_{1jk}^{▲,I}(x,y,z) = \widetilde{\text{curl}}(v_{1jk}^{▲,IV}(x,y,z))$$
$$w_{10k}^{▲,I}(x,y,z) = 4w_{1j}^{▲,F_1}(x,y,z)\widehat{P}_k^3(z)$$

Divergence-free face functions:

$$w_{ijk}^{▲,II}(x,y,z) = \text{curl}(v_{ijk}^{▲,II}(x,y,z))$$
$$w_{ijk}^{▲,III}(x,y,z) = \text{curl}(v_{ijk}^{▲,III}(x,y,z))$$
$$w_{1jk}^{▲,IV}(x,y,z) = \text{curl}(v_{1jk}^{▲,IV}(x,y,z))$$

Sparsity results can be found in [BPZ13b] for the $H(\text{curl})$ conforming functions and in [BPZ12] for the divergence-conforming functions.

[3] This functions need to be modified as in the $H(\text{curl})$ case. This is part of future work and not described in this thesis, although similar techniques as in the next section apply.

4.2.8. New $H(\mathrm{curl})$ basis functions on a triangle and a tetrahedron

Since $v_{ij}^{\triangle}(x,y)$ are not Nedelec conforming basis functions, we need to modify those. We introduce the following auxiliary notation in $2D$

$$\begin{aligned} f_i(x,y) &:= \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right) \\ g_{ij}(x,y) &:= \widehat{P}_j^{2i}(y) \end{aligned}$$

and in $3D$

$$\begin{aligned} f_i(x,y,z) &:= \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i \\ g_{ij}(x,y,z) &:= \widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^j \\ h_{ijk}(x,y,z) &:= \widehat{P}_k^{2i+2j}(z) \end{aligned}$$

Recall the space

$$\tilde{P}_k = \{\text{homogeneous polynomials of total degree exactly } k \text{ in } \mathbb{R}^d\}.$$

In our constructions for the $H(\mathrm{curl})$ conforming functions, we followed Nédélec's construction. By using the $\mathbb{W}_2$ operator, we defined non-curl-free functions and enriched this space by the gradients. But those interior functions, defined by the operator $\mathbb{W}_2$, do not split into a homogeneous and a non-homogeneous part, as needed by the Nédélec space of first order.

For the triangle we redefine our type II functions as

$$v_{ij}^{\triangle,\mathcal{N}}(x,y) = (\nabla f_i(x,y))g_{ij}(x,y) - \frac{i}{j}f_i(x,y)\nabla g_{ij}(x,y). \tag{4.13}$$

Those are in the first Nédélec space R_k, as can be seen in the following lemma:

Lemma 4.2.6
Let $i+j \leq p$, with $i \geq 2, j \geq 1$, then

$$v_{ij}^{\triangle,\mathcal{N}}(x,y) \in R_{p-1},$$

with $R_k = (P_{k-1})^2 \oplus S_k$, where $S_k = \{p \in (\tilde{P}_k)^2 | \vec{x}\cdot p = 0\}$, where $\vec{x} = (x,y)^\top$.

Proof. First recall the following relation

$$(1+x)P_n^{(\alpha,\beta+1)}(x) + (1-x)P_n^{(\alpha+1,\beta)} = 2P_n^{(\alpha,\beta)}(x) \tag{4.14}$$

For the first component of $v_{ij}^{\triangle,\mathcal{N}}(x,y)$ holds

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}x}\left(\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i \widehat{P}_j^{2i}(y)\right) &= L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}(y)\\
&\overset{(2.10)}{=} L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\frac{(1+y)}{j}P_{j-1}^{(2i-1,1)}(y)\\
&\overset{(4.14)}{=} \frac{1}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\left(2P_{j-1}^{(2i-1,0)}(y)-(1-y)P_{j-1}^{(2i,0)}(y)\right)\\
&= \underbrace{\frac{y}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i,0)}(y)}_{\in P_{p-1}\text{ or }\in y\cdot P_{p-2}}\\
&\quad+\underbrace{\frac{1}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\left(2P_{j-1}^{(2i-1,0)}(y)-P_{j-1}^{(2i,0)}(y)\right)}_{\in P_{p-2}}.
\end{aligned}$$

The second component follows analogously.

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}y}\left(\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i\right)\widehat{P}_j^{2i}(y)-\frac{i}{j}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i\frac{\mathrm{d}}{\mathrm{d}y}\widehat{P}_j^{2i}(y)\\
&\quad=\frac{x}{1-y}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}(y)\\
&\quad-\frac{i}{2}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}(y)-\frac{i}{j}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i}P_{j-1}^{(2i,0)}(y)\\
&\overset{(4.14)}{=}\frac{x}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-2}\left(2P_{j-1}^{(2i-1,0)}(y)-(1-y)P_{j-1}^{(2i,0)}(y)\right)\\
&\quad-\frac{i}{2j}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}\left(2P_{j-1}^{(2i-1,0)}(y)-(1-y)P_{j-1}^{(2i,0)}(y)\right)-\frac{i}{j}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i}P_{j-1}^{(2i,0)}(y)\\
&\quad=-\frac{x}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i,0)}(y)\\
&\quad+\frac{2x}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i-1,0)}(y)-\frac{2}{j}\widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i-1,0)}(y)\\
&\quad=\underbrace{-\frac{x}{j}L_{i-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i,0)}(y)}_{\in P_{p-1}\text{ or }\in(-x)\cdot P_{p-2}}+\underbrace{\frac{2}{j}L_{i-2}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{i-1}P_{j-1}^{(2i-1,0)}(y)}_{\in P_{p-2}}
\end{aligned}$$

□

Alternatively one could use the formulation

$$\tilde{v}_{ij}^{\triangle,\mathcal{N}}(x,y)=j\,(\nabla f_i(x,y))g_{ij}(x,y)-i\,f_i(x,y)\nabla g_{ij}(x,y).$$

The same problem holds for the three-dimensional case on the tetrahedron. Instead of proving everything in one go, as for the $2D$ case, we apply a more constructive approach. The idea is very simple: We first investigate the influence of $\vec{x}\cdot$ on the gradient of our auxiliary functions f, g, h, and then recombine those, such that the highest polynomial order vanishes under multiplication with $\vec{x}$. Therefore, the following auxiliary lemmas are introduced.

Lemma 4.2.7
Let $\vec{x} = (x, y, z)^\top$, then

$$\vec{x} \cdot (\nabla f_i(x,y,z)) g_{ij}(x,y,z) h_{ijk}(x,y,z) = i\, u^{\blacktriangle}_{ijk}(x,y,z) + \underbrace{p(x,y,z)}_{\in P^{p-1}},$$

for $i + j + k \leq p$.

Proof. It is enough to show that

$$\vec{x} \cdot \nabla f_i = i \widehat{L}_i \left(\frac{4x}{1 - 2y - z} \right) (1 - 2y - z)^i + R_{i-1},$$

where $R_{i-1} \in P^{i-1}$. Let

$$\tilde{f}_i(x,y,z) = \begin{pmatrix} L_{i-1}(\eta) \\ \frac{1}{2} L_{i-2}(\eta) \\ \frac{1}{4} L_{i-2}(\eta) \end{pmatrix},$$

with $\nabla f_i(x,y,z) = \tilde{f}_i(x,y,z) \left(\frac{1-2y-z}{4}\right)^{i-1}$ and $\eta = \left(\frac{4x}{1-2y-z}\right)$.

$$\begin{aligned}
\vec{x} \cdot \tilde{f}_i(x,y,z) &= \left(x L_{i-1}(\eta) + \frac{y}{2} L_{i-2}(\eta) + \frac{z}{4} L_{i-2}(\eta) \right) \\
&= \frac{1}{4} \left(4x L_{i-1}(\eta) - (1 - 2y - z) L_{i-2}(\eta) + L_{i-2}(\eta) \right) \\
&= \frac{1 - 2y - z}{4} \left(\frac{4x}{1 - 2y - z} L_{i-1}(\eta) - L_{i-2}(\eta) + \frac{1}{1 - 2y - z} L_{i-2}(\eta) \right)
\end{aligned}$$

By application of the recursion

$$i \widehat{L}_i(x) = x L_{i-1}(x) - L_{i-2}(x),$$

see [BS06], we can simplify the last term to

$$\vec{x} \cdot \tilde{f}_i(x,y,z) = i \left(\frac{1 - 2y - z}{4} \right) \widehat{L}_i(\eta) + \frac{1}{4} L_{i-2}(\eta)$$

and thus

$$\vec{x} \cdot \nabla f_i(x,y,z) = i \underbrace{\left(\frac{1 - 2y - z}{4} \right)^i \widehat{L}_i \left(\frac{4x}{1 - 2y - z} \right)}_{f_i(x,y,z)} + \underbrace{\frac{1}{4} \left(\frac{1 - 2y - z}{4} \right)^{i-1} L_{i-2} \left(\frac{4x}{1 - 2y - z} \right)}_{\in P^{i-1}}$$

□

Lemma 4.2.8
Let $\vec{x} = (x, y, z)^\top$, then

$$\vec{x} \cdot f_i(x,y,z)(\nabla g_{ij}(x,y,z))h_{ijk}(x,y,z) = j\, u_{ijk}^{\blacktriangle}(x,y,z) + \underbrace{p(x,y,z)}_{\in P^{p-1}},$$

for $i + j + k \leq p$.

Proof. Let

$$\tilde{g}_{ij}(x,y,z) = \begin{pmatrix} 0 \\ P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) \\ \left(\frac{y}{1-z}\right) P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) - \frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) \end{pmatrix},$$

where $\nabla g_{ij}(x,y,z) = \tilde{g}_{ij}(x,y,z)\,(1-z)^{j-1}$. We write the scalar product as follows

$$\begin{aligned}
\vec{x} \cdot \tilde{g}_{ij}(x,y,z) &= yP_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) + \left(\frac{yz}{1-z}\right)P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) - z\frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) \\
&= \frac{y}{1-z}P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) - z\frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)
\end{aligned}$$

From [BP07] the relation

$$\eta P_{j-1}^{(\alpha,0)}(\eta) - j\widehat{P}_j^{\alpha}(\eta) = \frac{1}{2j+\alpha-2}\left(-\alpha P_{j-1}^{(\alpha,0)}(\eta) + (2j-2)P_{j-2}^{(\alpha,0)}(\eta)\right) \tag{4.15}$$

is known. Thus we apply the relation as follows

$$\begin{aligned}
\vec{x} \cdot \tilde{g}_{ij}(x,y,z) &= \frac{y}{1-z}P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) - z\frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) \\
&= \frac{y}{1-z}P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right) - \frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) + (1-z)\frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) \\
&= (1-z)\frac{j}{2}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right) + \tilde{R}_{j-1}(x,y,z).
\end{aligned}$$

After multiplication with $(1-z)^{j-1}$, it follows that

$$\vec{x} \cdot \nabla g_{ij}(x,y,z) = j\underbrace{\left(\frac{1-z}{2}\right)^j \widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)}_{g_{ij}(x,y,z)} + \underbrace{R_{j-1}(x,y,z)}_{\in P^{j-1}}.$$

□

Lemma 4.2.9
Let $\vec{x} = (x, y, z)^\top$, then

$$\vec{x} \cdot f_i(x,y,z) g_{ij}(x,y,z)(\nabla h_{ijk}(x,y,z)) = k\, u^{\blacktriangle}_{ijk}(x,y,z) + \underbrace{p(x,y,z)}_{\in P^{p-1}},$$

for $i + j + k \leq p$.

Proof. Consider

$$\nabla h_{ijk}(x,y,z) = \begin{pmatrix} 0 \\ 0 \\ P_{k-1}^{(2i+2j,0)}(z) \end{pmatrix}.$$

We can modify the scalar product by

$$\begin{aligned} \vec{x} \cdot \nabla h_{ijk}(x,y,z) &= z P_{k-1}^{(2i+2j,0)}(z) \\ &\overset{(4.15)}{=} k \underbrace{\widehat{P}_k^{2i+2j}(z)}_{h_{ijk}(x,y,z)} + \underbrace{R_{k-1}}_{\in P^{k-1}}. \end{aligned}$$

□

We apply the results of lemmas 4.2.7 to 4.2.9 to a general polynomial function of the type

$$\phi_{ijk}(x,y,z) = c_1 \nabla f g h + c_2 f \nabla g h + c_3 f g \nabla h, \quad \in \left(P^{i+j+k-1}\right)^3. \tag{4.16}$$

Thus the relation

$$\vec{x} \cdot \phi_{ijk}(x,y,z) = (c_1\, i + c_2\, j + c_3\, k) u^{\blacktriangle}_{ijk} + R_{i+j+k-1} \tag{4.17}$$

holds. If $(c_1\, i + c_2\, j + c_3\, k) = 0$ the polynomial $\phi_{ijk}(x,y,z)$ is a Nédélec function of first kind. One possible (non-unique) set of linear independent functions is given by,

$$\begin{aligned} v^{\blacktriangle,II,\mathcal{N}}_{ijk} &:= j\, \nabla(f_i) g_{ij} h_{ijk} - i\, f_i \nabla(g_{ij}) h_{ijk}, \\ v^{\blacktriangle,III,\mathcal{N}}_{ijk} &:= k\, \nabla(f_i) g_{ij} h_{ijk} - i\, f_i g_{ij} \nabla(h_{ijk}). \end{aligned} \tag{4.18}$$

We summarize our results in the following theorem.

Theorem 4.2.10
Let $i + j + k \leq p$ with $i \geq 2, j, k \geq 1$, then the functions $v^{\blacktriangle,I}_{ijk}(x,y,z), v^{\blacktriangle,IV}_{1jk}(x,y,z)$ as in (6.22) and $v^{\blacktriangle,I,\mathcal{N}}_{ijk}(x,y,z), v^{\blacktriangle,III,\mathcal{N}}_{ijk}(x,y,z)$ as in (4.18) are a basis of $H(\text{curl}, \blacktriangle)$. Furthermore $v^{\blacktriangle,I}_{ijk}(x,y,z)$ are Nédélec functions of second kind, while $v^{\blacktriangle,I,\mathcal{N}}_{ijk}(x,y,z), v^{\blacktriangle,III,\mathcal{N}}_{ijk}(x,y,z)$ and $v^{\blacktriangle,IV,\mathcal{N}}_{1jk}(x,y,z)$ are Nédélec functions of first kind.

In the case of $H(\text{div})$ conforming basis functions, it needs to be shown that those functions are part of the Raviart-Thomas-Nédélec space of the first kind. This is postponed to future work.

5. Computation of local finite element matrices

Entries of finite element matrices can be computed by application of high order multidimensional quadrature or cubature rules. This naive approach is highly cost intensive. Under the assumption of a tensorial or tensorial-like structure, the problem can be reduced to multiple one dimensional integrals.
Furthermore, the state-of-the-art method for the efficient computation of finite element entries is the so-called *sum factorization* method introduced by Orszag [Ors80] for spectral element methods. It was extended e.g. by [SK95, MGS99] and [EM05] for high order finite element methods, which leads to almost optimal complexity under the assumption of sparsity. In the context of iterative solvers, the sum factorization approach can be used to define a matrix-free matrix-vector product.
We will introduce a new approach which computes the finite element matrices in optimal complexity, under the assumption of piecewise constant material functions. This new approach is based on the recursive relations of theorem 2.2.9. Moreover, this approach computes the element matrices directly. A direct computation of the element matrices has the advantage, that it can be reapplied, e.g. in a non-linear finite element method. Furthermore, a p-refinement on the same cell can be achieved in optimal complexity as well. This is one of the few algorithms which achieve optimal complexity. An algorithm for a basis based on Bernstein polynomials can be found, e.g. in [AAD11, AF18]. This algorithm achieves optimal complexity even for smooth non-constant material functions, but has a rapidly increasing condition number.
For this chapter, we will solve all integrals on the cube. Thus, we first need to transform all functions on a tetrahedron by the Duffy transformation D_3 onto a cube. In the following, we consider different algorithms to set up the local mass and stiffness matrix M and K.

5.1. Standard algorithm

Let $\Phi = \{u_i | i = 1, \ldots, N\}$ be an arbitrary set of shape function, then the local mass and stiffness matrix are given by

$$M_{ij} := \left[\int_T u_i(\vec{x}) u_j(\vec{x}) \mathrm{d}\vec{x}\right],$$
$$K_{ij} := \left[\int_T (A \nabla u_i(\vec{x})) \cdot u_j(\vec{x}) \mathrm{d}\vec{x}\right],$$

where A is a constant matrix and T is any of the reference elements $\square, \triangle, \blacksquare$ or $\blacktriangle$. The standard algorithm for setting up the stiffness matrix is the application of $1D$ Gaussian quadrature rules in each direction. The Jacobian determinant $|\det(\mathrm{d}D_d)|$ of the Duffy transformation D_d (4.3) or (4.11) can be incorporated by application of a Gauss-Jacobi quadrature. The algorithm is stated in algorithm 1.
Assuming we have the same polynomial order p in each direction, we need p^d quadrature points. Since we have N^2 entries the resulting asymptotic complexity is at least $\mathcal{O}(N^2 p^d)$. For a triangle

Algorithm 1 Computation of the entries K_{ij}

1: Choose Gaussian quadrature rule for each direction: $GQ^{(i)} = \{(\eta_0^{(i)}, \omega_0^{(i)}), \dots, (\eta_{q_i}^{(i)}, \omega_{q_i}^{(i)})\}$,
2: **Set** $GQ = GQ^{(1)} \times \dots \times GQ^{(d)}$.
3: **Initialize** $K_{ij} = 0$,
4: **for all** $\eta, \omega \in GQ$ **do**
5: **Set**

$$K_{ij} \mathrel{+}= \omega \left(\tilde{\nabla}(u_i \circ F_k) \cdot A\tilde{\nabla}(u_j \circ F_k)\right)|_\eta \quad \text{for all } 1 \le i, j \le N$$

6: **end for**

we have $N = \frac{1}{2}(p-1)(p-2)$ and for a tetrahedron $N = \frac{1}{6}(p-1)(p-2)(p-3)$ shape functions, i.e. the asymptotic complexity on a triangle and tetrahedron is $\mathcal{O}(p^{3d})$. Under the assumption of a constant matrix A, we get sparsity for a chosen special basis, and such we only have to integrate the p^d non-zero entries. The complexity then reduces to $\mathcal{O}(p^{2d})$.

5.2. Sum factorization

The sum factorization is a reordering approach of the standard algorithm. Denote by $\Phi^{\triangle}$ the basis of the H^1 on the triangle and $\Phi^{\blacktriangle}$ on the tetrahedron. We follow Eibner and Melenk [EM05] to rewrite the gradient in tensorial-like structure as well.

Corollary 5.2.1 (Eibner/Melenk 2005)
For $d = 2, 3$ let $u^{\triangle}$ and $u^{\blacktriangle}$ be given as in tables 4.2.7 and 4.2.9. Then the entries of the stiffness matrix K can be computed as

$$K_{ij} = \int_{Q^d} (\tilde{\nabla} u_j \cdot \hat{C} \tilde{\nabla} u_i) |\det \mathrm{d}D_d| \mathrm{d}\Omega = \sum_{r,r'=1}^{d} \int_{Q^d} \tilde{\nabla}_{r'} u_j \hat{C}_{r'r} \tilde{\nabla}_r u_i |\det \mathrm{d}D_d| \mathrm{d}\Omega,$$

where

$$\tilde{\nabla} u_i = \begin{cases} \left[\frac{1}{(1-\eta_2)} \frac{\partial u}{\partial \eta_1}, \frac{\partial u}{\eta_2}\right]^\top & \text{for } d = 2 \\ \left[\frac{1}{(1-\eta_2)(1-\eta_3)} \frac{\partial u}{\partial \eta_1}, \frac{1}{(1-\eta_3)} \frac{\partial u}{\partial \eta_2}, \frac{\partial u}{\eta_3}\right]^\top & \text{for } d = 3 \end{cases}$$

is polynomial and

$$\hat{C} := M_d^{-1} (A \circ D_d) M_d^{-\top}, \quad M_2^{-1} := \begin{bmatrix} 2 & 2(1+\eta_1) \\ 0 & 1 \end{bmatrix}, \quad M_3^{-1} := \begin{bmatrix} 4 & 2(1+\eta_1) & 2(1+\eta_1) \\ 0 & 2 & (1+\eta_2) \\ 0 & 0 & 1 \end{bmatrix}.$$

We now have a tensorial structure for our gradients. Since our (transformed) shape functions and their gradients are of tensorial type structure, we can first compute auxiliary fields and then perform the summation in a more optimal order. Consider for example the mass matrix M on the triangle. The entry $M_{n,m}$, where $(n, m) = (n(i, j), m(i', j'))$ can be computed in the standard approach as

$$
\begin{aligned}
M_{n,m} &= \int_{-1}^{1} \widehat{L}_i(\eta_1)\widehat{L}_{i'}(\eta_1)\mathrm{d}\eta_1 \int_{-1}^{1} \left(\frac{1-\eta_2}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(\eta_2)\widehat{P}_{j'}^{2i'}(\eta_2)\mathrm{d}\eta_2 \\
&\approx \sum_{q_1=0}^{p} \omega_{q_1} \widehat{L}_i(\hat{\eta}_{p_1})\widehat{L}_{i'}(\hat{\eta}_{p_1}) \sum_{q_2=0}^{p} \omega_{q_2} \widehat{P}_j(\hat{\eta}_{q_2})\widehat{P}_{j'}(\hat{\eta}_{q_2}).
\end{aligned}
$$

On the other hand, by the sum factorization ansatz we first compute both sums for all indices and then get the entries by reading out the entries of the auxiliary fields. By this approach, we save us the repetitive computation of the first sum.
All entries of this auxiliary arrays for the tetrahedron can be computed in optimal complexity $\mathcal{O}(p^3)$, but only in suboptimal complexity $\mathcal{O}(p^3)$ in $2D$ for the triangle, see [BPZ13a].
By corollary 5.2.1, the tensorial structure extends to the gradients of the shape functions on simplices. For the tetrahedron we write

$$
u_{ijk}^{\blacktriangle,T} \circ D_3 = g_{T,i}(\eta_1) g_{T,i,j}(\eta_2) g_{T,i,j,k}(\eta_3), \tag{5.1}
$$

where T denotes the type of function, i.e. vertex, edge, face or interior. We use an analogous notation for the gradient as well. For the stiffness matrix this leads to algorithm 2 in $3D$, which has a complexity of $\mathcal{O}(p^{2d+1})$, see [MGS99, EM05] for a detailed analysis. The algorithm in $2D$ follows by the same ideas. See also [VSK10] for a runtime comparison for low and high order discretizations. Under the assumption that A is constant, we can use sparsity to reduce the numerical costs. Since we have p^d non-zero entries instead of p^{2d}, the complexity is only $\mathcal{O}(p^{d+1})$. Although this could be reduced to optimal complexity in $3D$, i.e. $\mathcal{O}(p^3)$, this asymptotic advantages are only observable for high polynomial orders due to high coefficients in the complexity, see [BPZ13a].
For an application in an iterative solver like a **CG**- or **GMRES**-method, the sum factorization can be used to calculate the matrix-vector product in $\mathcal{O}(p^{d+1})$ operations independent of the choice of material functions A. But it has been shown in [VSK10] that the break-even point of the matrix-free sum-factorization approach and an approach where the local matrix is computed is dependent on the differential operator and the chosen reference element. E.g. they have shown for the Helmholtz operator on a mesh, consisting of triangles, it is as high as $p = 27$, which may be a low polynomial degree for a spectral method, but is a high polynomial degree for a hp-FEM method.

Algorithm 2 Sum factorization for the entries K_{ij}

Choose quadrature rules

$$GQ^i = \{(\eta^{(i)}_{l_i}, \omega^{(i)}_{l_i})| l_i = 0, \dots, q_i\}$$

which incorporates $|\det(\mathrm{d}D_3)|$.

for all $1 \le r \le 3$ **and** types of shape functions T **do**

$$\tilde{\nabla}_r u^{\blacktriangle,T}_{ijk} = \tilde{g}^{(1)}_{T,r,i}(\eta_1)\tilde{g}^{(2)}_{T,r,i,j}(\eta_2)\tilde{g}^{(3)}_{T,r,i,j,k}(\eta_3)$$

end for

for all $1 \le r \le 3$, all types T **and** all (i, j, k) depending on T **do**

  Compute the auxiliary arrays:

$$\begin{aligned} G^{(1)}[T, r, i, l_1] &= \tilde{g}^{(1)}_{T,r,i}(\eta^{(1)}_{l_1}) \\ G^{(2)}[T, r, i, j, l_2] &= \tilde{g}^{(2)}_{T,r,i,j}(\eta^{(2)}_{l_2}) \\ G^{(3)}[T, r, i, j, k, l_3] &= \tilde{g}^{(3)}_{T,r,i,j,k}(\eta^{(3)}_{l_3}) \end{aligned}$$

end for

for $1 \le r, r' \le 3$, and $0 \le l_i \le q_i$ **do**

  Compute the auxiliary array

$$\hat{C}[r', r, l_1, l_2, l_3] = \hat{C}_{(r',r)}(\eta^{(1)}_{l_1}, \eta^{(2)}_{l_2}, \eta^{(3)}_{l_3})$$

end for

Initialize $K = 0$

for all $1 \le r, r' \le 3$ and all types T, T' **do**

  Compute :

$$\begin{aligned} H^{(1)}[i, i', l_3, l_2] &= \sum_{l_1=0}^{q_1} G^{(1)}[T, r, i, l_1]G^{(1)}[T', r', i', l_1]\hat{C}[]r', r, l_1, l_2, l_3]\omega^{(1)}_{l_1}, \\ H^{(2)}[i, i', j, j', l_3] &= \sum_{l_2=0}^{q_2} G^{(2)}[T, r, i, j, l_2]G^{(2)}[T', r', i', j', l_2]H^{(1)}[i, i', l_3, l_2]\omega^{(2)}_{l_2} \end{aligned}$$

  and

$$K[T, i, j, k][T', i', j', k'] \mathrel{+}= \sum_{l_3=0}^{q_3} G^{(3)}[T, r, i, j, k, l_3]G^{(3)}[T', r', i', j', k', l_3]H^{(2)}[i, i', j, j', l_3]\omega^{(3)}_{l_3}$$

end for

5.3. Recursion formulas

Let A be elementwise constant, then we can assume sparsity of our mass and stiffness matrices. Under this assumption, we derive a new ansatz for the computation of finite element matrices, which has optimal complexity. Let

$$\begin{aligned} I_{n,m}^{(a,b,\mu)} &= \int_{-1}^{1} (1-x)^{\mu} P_n^{(a,0)}(x) P_m^{(b,0)}(x)\,\mathrm{d}x, \\ J_{n,m}^{(a,b,\mu)} &= \int_{-1}^{1} (1-x)^{\mu} \widehat{P}_n^{a}(x) P_m^{(b,0)}(x)\,\mathrm{d}x, \\ K_{n,m}^{(a,b,\mu)} &= \int_{-1}^{1} (1-x)^{\mu} \widehat{P}_n^{a}(x) \widehat{P}_m^{b}(x)\,\mathrm{d}x, \end{aligned} \tag{5.2}$$

and

$$\vec{c}(n,m,\mu,\nu,\alpha,\beta,\rho,\delta) := \begin{pmatrix} n+m+\mu+\nu+4 \\ n+\alpha+\beta-m-\mu-\nu-2 \\ m+\rho+\delta-n-\mu-\nu-2 \\ n+m+\alpha+\beta+\rho+\delta-\mu-\nu \end{pmatrix}, \tag{5.3}$$

then the following corollaries are a direct consequence of theorem 2.2.9 and corollary 2.2.10.

Corollary 5.3.1
Let $n, m \geq 2$ and $\alpha, \beta, \mu > -1$. Then the recursive relation

$$c_1 I_{n+1,m+1}^{(a,b,\mu)} = c_2 I_{n,m+1}^{(a,b,\mu)} + c_3 I_{n+1,m}^{(a,b,\mu)} + c_4 I_{n,m}^{(a,b,\mu)},$$

holds, where $\nu = -1, \beta = 0, \delta = 0$, and

$$\vec{c} = \vec{c}(n,m,\mu,-1,\alpha,0,\rho,0).$$

Proof. Insert $\beta = \rho = 0$ and $\nu = -1$ in corollary 2.2.10. □

Similarly, we arrive at the following two corollaries.

Corollary 5.3.2
Let $n, m \geq 2$ and $\alpha, \beta, \mu > -1$. Then the recursive relation

$$c_1 J_{n+1,m+1}^{(a,b,\mu)} = c_2 J_{n+1,m}^{(a,b,\mu)} + c_3 J_{n,m+1}^{(a,b,\mu)} + c_4 J_{n,m}^{(a,b,\mu)} \tag{5.4}$$

holds, where $\nu = 0, \beta = 1, \delta = 0$, and

$$\vec{c} = \vec{c}(n,m,\mu,0,\alpha-1,1,\rho,0).$$

Proof. Since

$$\widehat{P}_i^{\alpha}(x) = \frac{1+x}{i} P_{i-1}^{(2i-1,1)}(x)$$

we apply corollary 2.2.10 with $\nu = 0, \beta = 1, \rho = 0$, but since β was raised, we need to lower α by one. □

Corollary 5.3.3

Let $n, m \geq 2$ and $\alpha, \beta, \mu > -1$. Then the recursive relation

$$c_1 K^{(a,b,\mu)}_{n+1,m+1} = c_2 K^{(a,b,\mu)}_{n+1,m} + c_3 K^{(a,b,\mu)}_{n,m+1} + c_4 K^{(a,b,\mu)}_{n,m} \tag{5.5}$$

holds, where $\nu = 1, \beta = 1, \delta = 1$, and

$$\vec{c} = \vec{c}(n, m, \mu, 1, \alpha - 1, 1, \rho - 1, 1).$$

Proof. We apply corollary 2.2.10 with $\nu = 1, \beta = \delta = 1$ and lower α and ρ by one. ☐

Corollary 5.3.4

For $I^{(a,b,\mu)}_{n,m}$, $J^{(a,b,\mu-1)}_{n,m}$ and $K^{(a,b,\mu-2)}_{n,m}$ the coefficients c_1, c_2, c_3, c_4 are identical.

Proof. Insert the according values in corollary 5.3.1, corollary 5.3.2 and corollary 5.3.3 ☐

The vector $\vec{c}$ is a function depending on the coefficients $n, m, \mu, \nu, \alpha, \beta, \rho$ and δ. As seen in the previous corollaries, if we replace a Jacobi polynomial $P^{(\alpha,\beta)}_n(x)$ by $\widehat{P}^{\alpha}_n(x)$, we need to do the replacement:

$$\begin{aligned} n &\to n - 1, \\ \nu &\to \nu + 1, \\ \alpha &\to \alpha - 1, \\ \beta &\to \beta + 1. \end{aligned}$$

Examples of the resulting coefficients can be found in example 5.3.1.

Example 5.3.1

Consider the integral

$$K^{(1)}_{n,m} = \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}^{2i}_n(y)\widehat{P}^{2i'}_m(y)$$

then the recursion is given by corollary 5.3.1 as

$$\begin{aligned} (n+m+i+i'+2)K^{(1)}_{n,m} =& (n-m+i-i'-3)K^{(1)}_{n-1,m} + (m-n+i'-i-3)K^{(1)}_{n,m-1} \\ &+ (n+m+i+i'-4)K^{(1)}_{n-1,m-1}. \end{aligned} \tag{5.6}$$

For the integral

$$K^{(2)}_{n,m} = \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+2} \widehat{P}^{2i}_n(y)\widehat{P}^{2i'}_m(y)$$

the recursion

$$\begin{aligned} (n+m+i+i'+3)K^{(1)}_{n,m} =& (n-m+i-i'-4)K^{(1)}_{n-1,m} + (m-n+i'-i-4)K^{(1)}_{n,m-1} \\ &+ (n+m+i+i'-5)K^{(1)}_{n-1,m-1} \end{aligned} \tag{5.7}$$

holds.

In the following, we derive an algorithm for the computation of the interior mass matrix for the H^1 basis on a tetrahedron. An application in $2D$ is straight forward and moreover an extension to edge and face functions can easily be done.
Consider an entry $M_{n(i,j,k),m(i',j',k')}$ of the mass matrix. Here $n(i,j,k), m(i',j',k')$ map (i,j,k) and (i',j',k') onto unique indices (n,m) of our mass matrix.
An entry is computed exactly after the Duffy transformation as

$$\begin{aligned} M_{n(i,j,k),m(i',j',k')} = &\int_{-1}^{1} \widehat{L}_i(\eta_1)\widehat{L}_{i'}(\eta_1)\,\mathrm{d}\eta_1 \int_{-1}^{1} \left(\frac{1-\eta_2}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(\eta_2)\widehat{P}_{j'}^{2i'}(\eta_2)\,\mathrm{d}\eta_2 \\ &\int_{-1}^{1} \left(\frac{1-\eta_3}{2}\right)^{i+j+i'+j'+2} \widehat{P}_k^{2i+2j}(\eta_3)\widehat{P}_{k'}^{2i'+2j'}(\eta_3)\,\mathrm{d}\eta_3. \end{aligned} \tag{5.8}$$

Since all 3 integrals are independent of each other, we can apply corollary 5.3.1, to each of those integrals. The first integral

$$G^{(1)}[i,i'] := \int_{-1}^{1} \widehat{L}_i(x)\widehat{L}_{i'}(x)\,\mathrm{d}x,$$

can be computed directly, see appendix A.3.1.
We use corollary 5.3.1 to compute the second integral

$$\tilde{G}^{(2)}[i,i',j,j'] := \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y \tag{5.9}$$

and the third integral

$$\tilde{G}^{3}[\beta,\beta',k,k'] := \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{\beta+\beta'+2} \widehat{P}_k^{2\beta}(z)\widehat{P}_{k'}^{2\beta'}(z)\,\mathrm{d}z, \tag{5.10}$$

where $\beta = i + j$. Those recursions need starting values, which can be computed by a Gaussian quadrature. This can be achieved by multiple low order Gauss-Jacobi quadratures or by one high order Gaussian quadrature. An alternative can be found in corollaries A.3.3 and A.3.4, where we can either compute the starting values directly or use small recurrent relations.
To optimally apply the sparsity pattern, we use helping functions. Due to corollary A.2.1 we define the functions

$$S_1^{(1)}[i,i',j] := i' - i + j + 2$$

and

$$S_2^{(1)}[i,i',j'] := i - i' + j' + 2.$$

Then for a fixed set of indices i, i', j', the functions $S_1^{(1)}$ gives the last index j, for which equation (5.9) is not zero. And analogously $S_2^{(1)}$ gives the last index of j' for the set i, i', j.
Accordingly, for the third integral (5.10) the index boundaries are given by

$$S_1^{(2)}[\beta,\beta',k] := \beta' - \beta + k + 3$$

and

$$S_2^{(2)}[\beta,\beta',k'] := \beta - \beta' + k' + 3.$$

Since $S_1^{(2)}$ and $S_2^{(2)}$ give the bigger index range, we will choose to only apply those for the recursions, to avoid one extra loop.

For the complete assembly routine we need to fulfil both the sparsity conditions $S^{(1)}$ and $S^{(2)}$, as well as $|i-i'| \leq 2$. We denote this set of indices (i,j,k,i',j',k') by $\mathfrak{S}$. Thus all entries can be computed by algorithm 3.

Algorithm 3 Recursive computation for the interior entries M_{nm}

for all $2 \leq i \leq p-2$ **and** i' s.t $|i-i'| \leq 2$ **do**
  Compute $G^{(1)}[i,i']$ by eq. (A.20) or eq. (A.21).
end for
for all $i,i',j,j' \in \mathbb{N}$ s.t. $j \leq S_1^{(2)}[i,i',j]$ **and** $j' \leq S_2^{(2)}[i,i',j']$ **do**
  Compute $G^{(2)}[i,i',1,j'], G^{(2)}[i,i',j,1], G^{(3)}[i,i',1,j']$ **and** $G^{(3)}[i,i',j,1]$ by corollary A.3.4
  Compute $\vec{c} = c(j-1,j'-1,i+i'+1,1,2i-1,1,2i'-1,1)$
  Compute $\vec{d} = c(j-1,j'-1,i+i'+2,1,2i-1,1,2i'-1,1)$
  Compute

$$G^{(2)}[i,i',j+1,j'+1] = \frac{1}{c_1}(c_2 G^{(2)}[i,i',j,j'+1] + c_3 G^{(2)}[i,i',j+1,j'] + c_4 G^{(2)}[i,i',j+1,j'+1])$$

  Compute

$$G^{(3)}[i,i',j+1,j'+1] = \frac{1}{d_1}(d_2 G^{(3)}[i,i',j,j'+1] + d_3 G^{(3)}[i,i',j+1,j'] + d_4 G^{(3)}[i,i',j+1,j'+1])$$

end for
for all $i,i',j,j',k,k' \in \mathfrak{S}$ **do**
  Compute $M_{n(i,j,k),m(i,j,k)} = G^{(1)}[i,i']G^{(2)}[i,i',j',j']G^{(3)}[i+j,i'+j',k,k']$
end for

Now onto the stiffness matrix K. An entry $K_{n(i,j,k),m(i',j',k')}$ of the stiffness matrix is given by

$$K_{n(i,j,k),m(i',j',k')} = \int_{\blacktriangle} \nabla u_{i,j,k}^{\top} \tilde{A} \nabla u_{i',j',k'} \mathrm{d}\vec{x}.$$

There are different choices how this integral can be split into smaller chunks. By application of corollary 5.2.1 we get a tensorial structure of the gradient, but since we compute all integrals exactly, we have to multiply our gradients with the matrix $\hat{C}$, where we lose the tensorial structure again.

An equivalent alternative is to work with the gradients directly. If we write those gradients out and sort for same (integrated) Jacobi polynomials, we can recursively compute all those integrals. As shown in [BP08] and [BPZ13a], we get 21 summands which needs to be computed, which are collected in table A.1 and table A.2. Those tables are written for reference purpose and readability, and should be optimized before implementation, e.g. similar terms can be excluded to reduce complexity.

We define the auxiliary arrays used in table A.1 and table A.2 as follows

$$
\begin{aligned}
I_1[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'-1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_2[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_3[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_4[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'} \widehat{P}_j^{2i}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_5[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_6[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} P_{j-1}^{(2i,0)}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_7[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+2} P_{j-1}^{(2i,0)}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y.
\end{aligned}
$$

and

$$
\begin{aligned}
L_1[i,i'] &:= \int_{-1}^{1} \widehat{L}_i(y)\widehat{L}_{i'}(y)\,\mathrm{d}y,\\
L_2[i,i'] &:= \int_{-1}^{1} L_{i-1}(y)\widehat{L}_{i'}(y)\,\mathrm{d}y,\\
L_3[i,i'] &:= \int_{-1}^{1} L_{i-1}(y)L_{i'-1}(y)\,\mathrm{d}y.
\end{aligned}
$$

By corollary 5.3.4, we know that replacing an integrated Jacobi polynomial $\widehat{P}_j^{\alpha}(y)$ by $(1-y)P_{j-1}^{(\alpha,0)}$ retains the same recursion formulas. Thus, the arrays I_1, I_4 and I_6 have the same recursion formula, and furthermore I_2, I_5, I_7 have the same recursion formula as well. The arrays $L_1[i,i'], L_2[i,i']$ and $L_3[i,i']$ can be computed by appendix A.3.1.
We denote the local sparsity pattern by $\hat{\mathfrak{S}}$, see [BP08, BPZ13a].

Remark 5
In algorithm 4 the lines 8-11 can be computed in $\mathcal{O}(p^2)$. Except for the computation of the starting values, the coefficients of the recursion are small, since they can be computed by a raise or reduction by unity.

Remark 6
If we use tables A.1 and A.2 in step 14 of algorithm 4 we have a lot of floating point operations for each of the $\mathcal{O}(p^3)$ entries. But consider e.g. table A.2 each term in the sum depends either on $I_2[i+j,i'+j',k,k']$, $I_5[i+j,i'+j',k,k']$, $I_5[i'+j',i+j,k',k]$ or $I_7[i+j,i'+j',k,k']$. If we group them in the optimal ordering, we reduce operations per set k,k'. For the set i,i',j,j' we can exclude, e.g. $L_3[i,i']$, to reduce the complexity even further.

Remark 7
In the triangular case, the stiffness matrix has the same recursive relation as the mass matrix, due to corollary 5.3.4. Let

$$
H[i,i',j,j'] = \int_{\triangle} u_{ij}^{\triangle}u_{i'j'}^{\triangle} + \nabla u_{ij}^{\triangle}\hat{C}\nabla u_{i'j'}^{\triangle}\,\mathrm{d}x. \tag{5.11}
$$

Algorithm 4 Recursive computation for the interior entries K_{nm}

for all $2 \leq i \leq p-2$ **and** i' s.t. $|i - i'| \leq 2$ **do**
    Compute $L_1[i, i']$ by eq. (A.20) and eq. (A.21)
    Compute $L_2[i, i']$ by eq. (A.22)
    Compute $L_3[i, i']$ by eq. (A.19)
end for
for all i, i', j, j' where $|i - i'| \leq 2, j \leq S_1^{1)}[i, i', j']$ **and** $j' \leq S_2^{(2)}[i, i', j]$ **do**
    for $q = 1, \ldots, 7$ **do**
        Compute $I_q[i, i', 1, j']$ and $I_q[i, i', j, 1]$
    end for
    Compute $\vec{c} = c(j-1, j'-1, i+i'-1, 1, 2i-1, 1, 2i'-1, 1)$
    Compute $\vec{d} = c(j-1, j'-1, i+i', 1, 2i-1, 1, 2i'-1, 1)$
    Compute $\vec{e} = c(j-1, j'-1, i+i'+1, 1, 2i-1, 1, 2i'-1, 1)$
    Compute $I_1[i, i', j+1, j'+1], I_4[i, i', j+1, j'+1], I_6[i, i', j+1, j'+1]$ by recursion with vector $\vec{c}$
    Compute $I_2[i, i', j+1, j'+1], I_5[i, i', j+1, j'+1], I_7[i, i', j+1, j'+1]$ by recursion with vector $\vec{d}$
    Compute $I_3[i, i', j+1, j'+1]$ by recursion with vector e
end for
for all $i, i', j, j', k, k' \in \hat{\mathfrak{S}}$ **do**
    Assemble $D_{r,r'} = \int_{\blacktriangle} (\mathrm{d}_{x_r} u_{ijk})(\mathrm{d}_{x_r} u_{i'j'k'}) \mathrm{d}\vec{x}$ by tables A.1 and A.2
    Compute $K_{n(ijk),m(i'j'k')} = \sum_{r,r'=1}^{3} A_{r,r'} D_{r,r'}$
end for

Then we state the following $2D$ algorithm:

Algorithm 5 Recursive computation for the interior matrix entries $M_{nm} + K_{nm}$ in $2D$

for all i, i', j, j' where $|i - i'| \leq 2, j \leq S_1^{1)}[i, i', j']$ **and** $j' \leq S_2^{(2)}[i, i', j]$ **do**
    Compute $H[i, i', 1, j']$ and $H[i, i', j, 1]$
    Compute $\vec{c} = \vec{c}(j-1, j'-1, i+i'+1, 1, 2i-1, 1, 2i'-1, 1)$
    Compute

$$H[i, i', j+1, j'+1] = \frac{1}{c_1}(c_2 H[i, i', j, j'+1] + c_3 H[i, i', j+1, j'] + c_4 H[i, i', j+1, j'+1])$$

end for

5.4. Extension to non-constant material functions

Up until this point, we assumed material functions to be piecewise constant, such that sparsity results and recursion formulas hold. In this section, an extension to the case with non-constant material functions will be discussed. We assume that the material function $\kappa(x) \in \mathbb{R}$, $x \in \mathbb{R}^d$ is at least elementwise continuous. Discontinuities at the element border are handled as usual.
We will present only the idea on how to extend the recursive approach to the mass matrix on the tetrahedron. Consider the mass matrix entry $M_{n(ijk),m(i',j',k')}$ under the influence of a non-constant

material function, given as

$$M_{n(i,j,k),m(i',j',k')} = \int_{\blacktriangle} \kappa(x,y,z) u^{\blacktriangle}_{ijk}(x,y,z) u^{\blacktriangle}_{i'j'k'}(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z. \tag{5.12}$$

Since an arbitrary $\kappa(x,y,z)$ will not be polynomial in general, we can not expect to find recursive relations. Furthermore, even if we would find recursive relations, we would need to compute those for each material function on each tetrahedron.

A more general approach is to approximate $\kappa(x,y,z)$ by some polynomial or rational functions, as

$$\kappa(x,y,z) = \sum_{i,j,k}^{q} \alpha_{ijk}\phi_{ijk}(x,y,z). \tag{5.13}$$

For the ease of presentation consider $\kappa(x,y,z) = \kappa(\eta_1,\eta_2,\eta_3)$ after the Duffy transformation. Since all recursive relations are based on the Beta integral (2.1) the best choice for our approximation functions ϕ are functions based on a linear combination of

$$(1-\eta_1)^{r_1}(1+\eta_1)^{r_2}(1-\eta_2)^{s_1}(1+\eta_2)^{s_2}(1-\eta_3)^{t_1}(1+\eta_3)^{t_2},$$

for some coefficients $r_1, r_2, s_1, s_2, t_1, t_2$. Those can be achieved e.g. by Bernstein [Lor86] or Jacobi polynomials. One possibility is to approximation $\kappa(x,y,z)$ by our shape functions, i.e. $u^{\blacktriangle}_{ijk}$. In section 6.1 we present a method to compute this approximation for the interior entries in optimal complexity. Inserting the approximation in (5.12) yields

$$M_{n(i,j,k),m(i',j',k')} = \sum_{r,s,t=0}^{q} \alpha_{rst} \int_{\blacktriangle} u^{\blacktriangle}_{rst}(x,y,z) u^{\blacktriangle}_{ijk}(x,y,z) u^{\blacktriangle}_{i'j'k'}(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z. \tag{5.14}$$

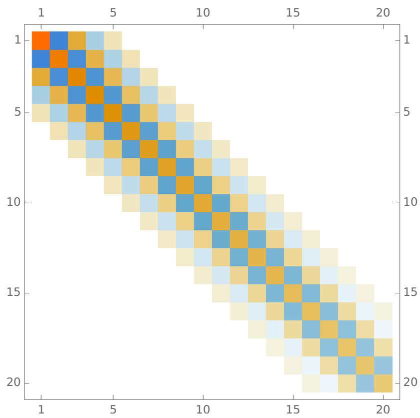
After the Duffy transformation, we need to compute

$$\begin{aligned} M_{n(i,j,k),m(i',j',k')} = \sum_{r,s,t=0}^{q} \alpha_{rst} \int_{-1}^{1} \widehat{L}_r(\eta_1)\widehat{L}_i(\eta_1)\widehat{L}_{i'}(\eta_1)\,\mathrm{d}\eta_1 \int_{-1}^{1} (1-\eta_2)^{r+i+i'+1} \widehat{P}^{2r}_s(\eta_2)\widehat{P}^{2i}_j(\eta_2)\widehat{P}^{2i'}_{j'}(\eta_2)\,\mathrm{d}\eta_2 \\ \int_{-1}^{1} (1-\eta_3)^{r+s+i+j+i'+j'+2} \widehat{P}^{2r+2s}_s(\eta_3)\widehat{P}^{2i+2j}_j(\eta_3)\widehat{P}^{2i'+2j'}_{j'}(\eta_3)\,\mathrm{d}\eta_3. \end{aligned} \tag{5.15}$$

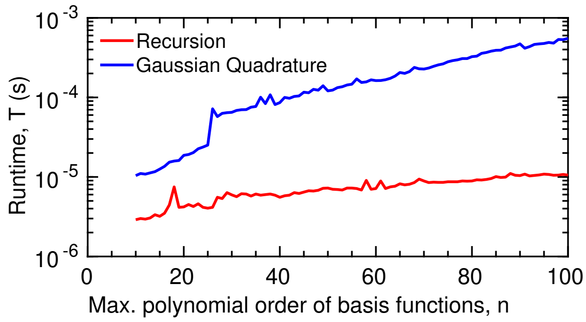
The integrals over η_2 and η_3 can be computed by (2.44). An alternative approach is to approximate the product $\kappa(x,y,z)u^{\blacktriangle}_{ijk}$ by our shape functions. Then we would only need to sum over our sparse mass matrix.

Nevertheless, the asymptotic complexity is $\mathcal{O}(q^d p^{2d})$, since we have dense element matrix with p^{2d} entries. We need to sum over our approximation, where the complexity depends on the quality of the approximation, which is $\mathcal{O}(q^d)$. If we choose the same polynomial degree as for our shape functions, the asymptotic complexity is $\mathcal{O}(p^{3d})$, as worse as the standard approach.

Alternatively one could compute new orthogonal polynomials, which includes $\kappa(x,y,z)$ in the weight functions, see e.g. [Gau82, Gau04]. On one hand, this approach would provide sparsity in the mass matrix again, but on the other hand, it is uncertain if recursive relations can be found. Those algorithms to construct orthogonal polynomial in dimensions $d > 1$ have only been introduced recently, see e.g. [LN23].



(a) Sparsity pattern of the example



(b) Comparing the runtime of Gaussian quadrature with recursion formulas

Figure 5.1.: $1D$ experiment of recursion formulas

5.5. Numerical experiments

For our numerical experiments, we assume that our material functions are constant. We compare the assembly time of one Gram matrix in $1D$ with the numerical quadrature, see the publication [BHP23]. Furthermore, we provide a small comparison between the sum factorization approach and the recursive approach for the $3D$ mass matrix.

5.5.1. 1D - Example

Let $\langle f(x), g(x)\rangle = \int_{-1}^{1} f(x)g(x)\left(\frac{1-x}{2}\right)^8 \mathrm{d}x$ be a weighted scalar product. Let $\phi_n(x) = P_n^{(4,0)}(x)$, then calculate the Gram matrix $G = \left[G_{i,j}\right]_{i,j}^{n_{max}^2} = \left[\langle \phi_i, \phi_j\rangle\right]_{i,j}^{n_{max}^2}$. The resulting sparsity pattern can be seen in Figure 5.1a. To compare the standard assembly routine with the recursive version, we assume now, that the quadrature points, weights and the basis functions are tabulated, i.e. we only need to perform the summation step of the standard assembly routine.

In Figure 5.1b the runtime of a Matlab implementation of both assembly routines are compared. We measured the assembly time of the Gram matrix for the total polynomial order $10 < n_{max} < 160$ for the integration of each non-zero value $\langle \phi_i, \phi_j\rangle$. As expected the recursive version is a lot faster than the Gaussian quadrature, even in $1D$.

5.5.2. Assembly time of the interior entries of the local $3D$ mass matrix

As seen above we split the computation of the local interior entries of $3D$ element mass matrix on the tetrahedron into the three auxiliary arrays,

$$
\begin{aligned}
G^{(1)}[i,i'] &= \int_{-1}^{1} \widehat{L}_i(x)\widehat{L}_{i'}(x) \\
G^{(2)}[i,i',j,j'] &= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}\,\mathrm{d}y \\
G^{(3)}[i,i',j,j'] &= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+2} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}\,\mathrm{d}y.
\end{aligned}
$$

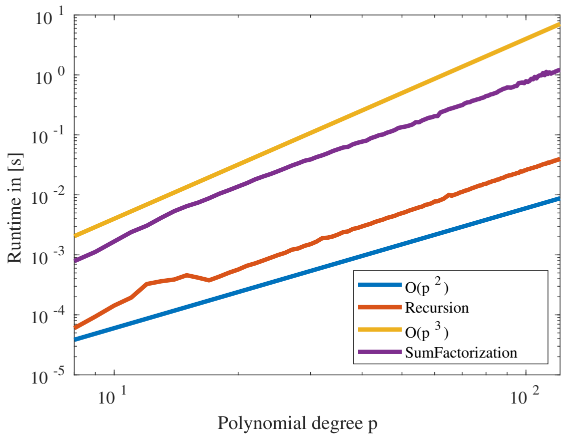


Figure 5.2.: Timings of the assembly of auxiliary arrays for $3D$ element mass matrix

As seen in [BPZ13a] we can compute the mass matrix in $3D$ in optimal complexity by a sum factorization approach. For this we first evaluate all relevant Jacobi polynomials at the quadrature points and then determine all non-zero entries by $G^{(1)}[i,i']G^{(2)}[i,i',j,j']G^{(3)}[i+j,i'+j',k,k']$. As such the only real difference between the sum factorization approach and the recursive approach, lies in the computation of the auxiliary arrays. In Figure 5.2, we see a direct comparison between the sum factorization[1] and the recursive approach as a small `Matlab` example.
Here we only compare the assembly time of the auxiliary arrays, we did not take the evaluation time of the Jacobi polynomials and the quadrature points into account. Moreover, we excluded the time for the assembly of $G^{(1)}[i,i']G^{(2)}[i,i',j,j']G^{(3)}[i+j,i'+j',k,k']$ as well.
The recursive approach computes the auxiliary arrays ca. 40 times faster than the sum factorization approach. Since the coefficients to compute the auxiliary arrays are constant, the recursive approach computes one auxiliary entry in optimal complexity $\mathcal{O}(1)$. On the other hand the quadrature approach computes those entries in complexity $\mathcal{O}(p)$.

Remark 8
Since the auxiliary arrays consists of $O(p^2)$ entries, the sum factorization approach is still in optimal complexity in $3D$. This is not the case for $2D$, where we still have $O(p^2)$ entries in the auxiliary arrays. See [BPZ13a].

Remark 9
The asymptotic behaviour of recursive and quadrature approach for the auxiliary arrays can only be observed for $p \gg 100$. For degrees smaller than that the number of non-zero entries is not yet in the asymptotic range. But note that the observed behaviour is not worse than $O(p^3)$, which is the asymptotic range given by the assembly of entries.

[1] All quadrature points and Jacobi polynomials were evaluated by `chebfun`[DHT14]

6. Algorithmic optimization for hp finite element methods

6.1. Dual functions

The aim of this chapter is to give new high order dual functions for H^1 and $H(\text{curl})$ functions of chapter 4.
Consider an arbitrary basis $\{\phi_i\}_i$ from chapter 4, then the respective dual functions $\{\psi_j\}_j \in L^2(\Omega)$ are given by the relation

$$\langle \phi_i, \psi_j \rangle = \delta_{i,j}.$$

Dual functions are used in defining interpolation operators, see e.g. [Mal09, Chap. 7], or transfer operators between finite elements spaces, see e.g. [WW98, WK01].
Consider for example some function $t(\vec{x})$, which we want to approximate by our basis functions ϕ_i, i.e.

$$t(\vec{x}) = \sum_{i=1}^{N} \alpha_i \phi_i,$$

where $N = \dim(\phi)$. This best approximation problem is solved by multiplication with test functions $v \in L^2(\Omega)$ and integration thereof. The following linear system needs to be solved:

$$\int_\Omega t(\vec{x}) v(\vec{x}) \mathrm{d}\vec{x} = \sum_{i=1}^{N} \int_\Omega \alpha_i \phi_i(\vec{x}) v(\vec{x}) \mathrm{d}\vec{x} \quad \forall v \in L^2.$$

The choice $v = \phi_i$ for all i would lead into a dense or almost dense system. On the other hand, the choice of biorthogonal functions leads to a diagonal system. Moreover, biorthogonal functions are well known from the theory of wavelets, see e.g. [Mal09]. There are also purely biorthogonal polynomial systems, see e.g. [DX14].
An algorithmic implementation for the H^1 dual functions can be found in `NGsolve` [Sch14].
A further application is the efficient calculation of the constraint matrices for non-conforming meshes, see e.g. section 6.2.
In $2D$ the problem in general reads:

Problem 6.1.1
Find $u_{hp} \in \mathbb{V}_{hp}$ such that

$$\begin{aligned}
u_{hp}(\lambda) &= u(\lambda) && \forall \text{ vertices } \lambda, \\
\int_E u_{hp} v &= \int_E uv && \forall v \in \mathcal{P}^{p-2} \text{ or } \mathcal{Q}^{p-2} \quad \forall \text{ edges } E, \\
\int_{Q/T} u_{hp} v &= \int_{Q/T} uv && \forall v \in \mathcal{P}^{p-3} \text{ or } \mathcal{Q}^{p-3} \quad \forall \text{ triangles/quadrilaterals } T.
\end{aligned}$$

A similar approach is valid in $3D$. We will start by introducing the H^1 dual functions for the quadrilateral and for the triangular case. In a next step, dual functions for $H(\text{curl})$ will be derived. We apply the already introduced notation, i.e. a function u denotes a H^1 basis function, $v^{Q,a}$ a $H(\text{curl})$ basis function of type $a = I, II, III$ on a reference element Q. Here i, j are the indices of the basis functions, while k, l are the indices of the dual functions in $2D$, or (i, j, k), and (l, m, n) respectively in $3D$. Furthermore p denotes either the total or the maximal polynomial degree, depending on the reference element.

6.1.1. Dual function in H^1

H^1 dual functions on the quadrilateral

Consider the master element $\square = (-1, 1)^2$. Recall that the face functions were defined as $u_{ij}^{\square}(x, y) = \widehat{L}_i(x)\widehat{L}_j(y)$, see table 4.2.2. To find the dual functions, we write the integrated Legendre polynomials as Jacobi polynomials by using the relation

$$\widehat{L}_i(x) = \frac{(x^2-1)}{2(i-1)} P_{i-2}^{(1,1)}(x), \tag{6.1}$$

see (2.11).
The dual function $\hat{b}_{kl}^{\square}(x, y) = b_k^{\square}(x)b_l^{\square}(y)$ has to satisfy the relation

$$\int_{\square} u_{ij}^{\square}(x, y)\hat{b}_{kl}^{\square}(x, y)\,\mathrm{d}x = \int_{-1}^{1} \widehat{L}_i(x)b_k^{\square}(x)\,\mathrm{d}x \int_{-1}^{1} \widehat{L}_j(y)b_l^{\square}(y)\,\mathrm{d}y = c_{ijkl}\delta_{i,k}\delta_{j,l}. \tag{6.2}$$

By inserting the Jacobi polynomials, we directly notice the orthogonality relations. Thus, one obtains

$$\tilde{c}\int_{-1}^{1}(x^2-1)P_{i-2}^{(1,1)}(x)b_k^{\square}(x)\,\mathrm{d}x \int_{-1}^{1}(y^2-1)P_{i-2}^{(1,1)}(y)b_l^{\square}(y)\,\mathrm{d}y = c_{kl}\delta_{i,k}\delta_{j,l}.$$

This motivates the choice

$$b_k^{\square}(x) = P_{k-2}^{(1,1)}(x) \text{ and } b_l^{\square}(y) = P_{l-2}^{(1,1)}(y). \tag{6.3}$$

The main driving factor is, that the integrated Jacobi (or Legendre) polynomials can be stated as Jacobi polynomials with an appropriate weight for the second Jacobi index. By normalizing b_{kl} with c_{kl}, the linear system of equation results in the identity matrix, i.e.

$$\frac{1}{c_{kl}}\int_Q u_{ij}^{\square}(x, y)\, b_{kl}(x, y)\,\mathrm{d}x\,\mathrm{d}y = \delta_{i,k}\delta_{j,l} = \frac{1}{c_{kl}}\int_Q f(x, y)\, b_{kl}(x, y)\,\mathrm{d}x\,\mathrm{d}y$$

H^1 dual function on the simplex

We now apply the same strategy to the simplicial case. (An extension from quadrilateral to hexahedron is straight forward). Recall that

$$u_{ij}^{\triangle} = \widehat{L}_i\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i \widehat{P}_j^{2i}(y)$$

on the triangle $\triangle$ with vertices $(-1,-1), (1,-1), (0,1)$, see table 4.2.7. Using

$$\widehat{P}_i^{\alpha}(x) = \frac{(1+x)}{n} P_{i-1}^{(\alpha-1,1)}(x)$$

and (6.1) $u_{ij}^{\triangle}(x,y)$ is rewritten as

$$u_{ij}^{\triangle}(x,y) = c\frac{1}{2}\left(\left(\frac{2x}{1-y}\right)^2 - 1\right) P_{i-2}^{(1,1)}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^i\left(\frac{1+y}{2}\right) P_{j-1}^{(2i-1,1)}(y),$$

with some known constant c. As before we search for $\hat{b}_{kl}^{\triangle}(x,y) = b_k^{\triangle}\left(\frac{2x}{1-y}\right) b_{kl}^{\triangle}(y) = b_k^{\triangle}(\eta)\, b_{kl}^{\triangle}(y)$, where $\eta = \frac{2x}{1-y}$. Using the Duffy transformation we write down the biorthogonality condition as

$$\int_{\triangle} u_{ij}^{\triangle}(x,y)\hat{b}_{kl}^{\triangle}(x,y)\,\mathrm{d}x = c\int_{-1}^{1}\left(\frac{\eta^2-1}{2}\right) P_{i-2}^{(1,1)}(\eta) b_k^{\triangle}(\eta)\mathrm{d}\eta \int_{-1}^{1}\left(\frac{1-y}{2}\right)^{i+1} P_{j-1}^{(2i-1,1)}(y) b_{kl}^{\triangle}(y)\,\mathrm{d}y. \tag{6.4}$$

Again this motivates the choice

$$b_k^{\triangle}(\eta) = P_{k-2}^{(1,1)}(\eta) \quad \text{and } b_{kl}^{\triangle}(y) = \left(\frac{1-y}{2}\right)^{k-2} P_{l-1}^{(2k-1,1)}(y).$$

Normalizing the dual functions means that the system matrix is again the identity matrix. We summarize in the following lemma:

Lemma 6.1.1 (H^1 dual functions on a triangle)
The face functions $u_{ij}^{\triangle}(x,y)$ as in (4.4) and

$$\hat{b}_{kl}^{\triangle}(x,y) = P_{k-2}^{(1,1)}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{k-2} P_{l-1}^{(2k-1,1)}(y) \quad \forall\, k \geq 2, i \geq 1 \tag{6.5}$$

are a biorthogonal system on $\triangle$, i.e.

$$\int_{\triangle} u_{ij}^{\triangle}(x,y)\hat{b}_{kl}^{\triangle}(x,y)\,\mathrm{d}x\,\mathrm{d}y = c\delta_{ik}\delta_{jl}$$

On the tetrahedron $\blacktriangle$ with vertices $(-1,-1,-1), (1,-1,-1), (0,1,-1)$ and $(0,0,1)$ the interior functions $u_{ijk}^{\blacktriangle}$, see table 4.2.9, and

$$b_{lnm}^{\blacktriangle}(x,y,z) = P_{l-2}^{(1,1)}\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^{(l-2)} P_n^{(2i-1,1)}\left(\frac{2z}{1-y}\right)^{(n-1)} P_m^{(2i+2j-1,1)}(z)$$

are biorthogonal, i.e.

$$\int_{\blacktriangle} u_{ijk}^{\blacktriangle}(x,y,z) b_{lnm}^{\blacktriangle}(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \tilde{c}\delta_{il}\delta_{jn}\delta_{km}. \tag{6.6}$$

Proof. The biorthogonality follows by inserting (6.5) in (6.4). Analogously, the biorthogonality for the tetrahedron follows by (6.6). □

6.1.2. Example of efficient approximation in H^1

Consider our reference triangle $\triangle$ with vertices $(-1,-1),(1,-)$ and $(0,1)$. Let $\phi \in L^2(\triangle)$ be the functions which we want to approximate, i.e.

$$\phi(x,y) = \sum_{s=1}^{ndof} \alpha_s u_s(x,y), \tag{6.7}$$

where $u_i(x,y)$ are the basis functions and $ndof$ is the number of degrees of freedom. Foremost $u_i(x,y)$ is given as the set of vertex, edge and face functions, see table 4.2.7. We can rewrite (6.7) as

$$\begin{aligned}\phi(x,y) = &\sum_{i=1}^{3} \alpha_i^v u_i^{\triangle,v}(x,y) + \sum_{i=2}^{p} \alpha_i^{E1} u_i^{\triangle,E1}(x,y) + \sum_{i=2}^{p} \alpha_i^{E2} u_i^{\triangle,E2}(x,y)\\ &+ \sum_{i=2}^{p} \alpha_i^{E3} u_i^{\triangle,E3}(x,y) + \sum_{i=2}^{p}\sum_{j=1}^{p-i} \alpha_{ij}^{I} u_{ij}^{\triangle}(x,y).\end{aligned} \tag{6.8}$$

Since $u_i^{\triangle,v}(x,y)$ are the only functions which have non-zero values at the vertices, we can directly determine

$$\begin{aligned}\alpha_1^v &= \phi(-1,-1),\\ \alpha_2^v &= \phi(1,-1),\\ \alpha_3^v &= \phi(0,1).\end{aligned}$$

Let $\tilde{\phi}(x,y) := \phi(x,y) - \sum_{i=1}^{3} \alpha_i^v u_i^{\triangle,v}(x,y)$. Our approximation problem (6.8) has been reduced to

$$\begin{aligned}\tilde{\phi}(x,y) = &\sum_{i=2}^{p} \alpha_i^{E1} u_i^{\triangle,E1}(x,y) + \sum_{i=2}^{p} \alpha_i^{E2} u_i^{\triangle,E2}(x,y)\\ &+ \sum_{i=2}^{p} \alpha_i^{E3} u_i^{\triangle,E3}(x,y) + \sum_{i=2}^{p}\sum_{j=1}^{p-i} \alpha_{ij}^{I} u_{ij}^{\triangle}(x,y).\end{aligned}$$

As mentioned at the beginning of this chapter, the usual approach would be to multiply with a set of test function and solve the remaining system. But due to the biorthogonal functions, we can reduce this problem to a diagonal system.
The steps are as follows: First multiply by a test function $v \in L^2(\triangle)$. Since only the functions u_i^{E1} have a non-vanishing trace on $E1$, integrate over $E1$. Thus we need to solve the following system

$$\int_{E1} \tilde{\phi}(x,y) v(x,y)\,\mathrm{d}x\,\mathrm{d}y = \sum_{i=2}^{p} \alpha_i^{E1} \int_{E1} u_i^{\triangle,E1}(x,y) v(x,y)\,\mathrm{d}x\,\mathrm{d}y \quad \forall v \in L^2(\triangle).$$

Since the trace of the edge functions is always $\widehat{L}_i(z)$, the best choice for the test functions is $\mathrm{tr}(v) = P_{j-2}^{(1,1)}(z)$. For example on $E1$ this reads as follows

$$\begin{aligned}\int_{E1} \tilde{\phi}(x,y) P_{j-2}^{(1,1)}(x)\,\mathrm{d}x\,\mathrm{d}y &= \sum_{i=2}^{p} \alpha_i \int_{-1}^{1} \widehat{L}_i(x) P_{j-2}^{(1,1)}(x)\,\mathrm{d}x\\ &= \alpha_i^{E1} c_{ij} \delta_{ij}.\end{aligned}$$

Thus, the entries α_i^{E1} can be directly computed by only solving the integral on the left-hand side. We can compute α_i^{E2} and α_i^{E3} analogously.
Now we only need to solve for the interior parts. Let

$$\widehat{\phi}(x,y) := \phi(x,y) - \left(\sum_{i=1}^{3} \alpha_i^v u_i^{\triangle,v}(x,y) + \sum_{i=2}^{p} \alpha_i^{E1} u_i^{\triangle,E1}(x,y) + \sum_{i=2}^{p} \alpha_i^{E2} u_i^{\triangle,E2}(x,y) + \sum_{i=2}^{p} \alpha_i^{E3} u_i^{\triangle,E3}(x,y) \right).$$

Here we again multiply by a test function $v \in L^2(\triangle)$, but this time integrate over $\triangle$, i.e.

$$\int_{\triangle} \widehat{\phi}(x,y)v(x,y) = \sum_{i=2}^{p} \sum_{j=1}^{p-i} \alpha_{ij}^{I} \int_{\triangle} u_{ij}^{\triangle}(x,y)v(x,y)\, \mathrm{d}x\, \mathrm{d}y \quad \forall v \in L^2(\triangle). \tag{6.9}$$

The best choice for v was given in (6.5), which results in the diagonal system

$$\int_{\triangle} \widehat{\phi}(x,y)v(x,y) = \alpha_{ij}^{I} c_{ij} \delta_{i,k} \delta_{j,l}. \tag{6.10}$$

The generalization to the tetrahedron is straight forward. First determine the coefficients w.r.t. the vertices, then determine the constraints w.r.t the edges. Next we need to determine all coefficients w.r.t. the faces, by integration over the individual faces. Lastly, solve for the coefficients w.r.t. the interior functions.
All biorthogonal functions on the edge and faces follow from the dimensional hierarchy, i.e. the trace of the faces on the respective face, is just the $2D$ face functions from table 4.2.7.

6.1.3. Dual functions in $H(\mathrm{curl})$

Finding dual functions for $H(\mathrm{curl})$ functions is more complicated. Not only are the shape functions vectorial, but they also appear in multiple types. Our goal is to find all dual functions which are orthogonal to the corresponding type of $H(\mathrm{curl})$ and additionally are zero for all other types.

Quadrilateral basis

Recall that the $H(\mathrm{curl})$ face shape functions on the quadrilateral $\square = (-1,1)^2$ written out are

$$\begin{aligned} v_{ij}^{\square,I}(x,y) &= \nabla \left(\widehat{L}_i(x)\widehat{L}_j(y) \right) = \begin{pmatrix} L_{i-1}(x)\widehat{L}_j(y) \\ \widehat{L}_i(x)L_{j-1}(y) \end{pmatrix}, \\ v_{ij}^{\square,II}(x,y) &= \mathbb{W} \left(\widehat{L}_i(x)\widehat{L}_j(y) \right) = \begin{pmatrix} L_{i-1}(x)\widehat{L}_j(y) \\ -\widehat{L}_i(x)L_{j-1}(y) \end{pmatrix}, \\ v_{i}^{\square,III}(x,y) &= \begin{pmatrix} \widehat{L}_i(y) \\ 0 \end{pmatrix}, \quad v_{i+p}^{\square,III}(x,y) = \begin{pmatrix} 0 \\ -\widehat{L}_i(x) \end{pmatrix}, \end{aligned} \tag{6.11}$$

for $2 \leq i, j \leq p$, see table 4.2.4. After linear combination we see, that we can also define the face functions as

$$\tilde{v}_{ij}^{\square,I}(x,y) = \nabla(\widehat{L}_i(x))\widehat{L}_j(y) = \begin{pmatrix} L_{i-1}(x)\widehat{L}_j(y) \\ 0 \end{pmatrix}, \quad \text{for } 1 \leq i \leq p, 2 \leq j \leq p, \tag{6.12}$$

$$\tilde{v}_{ij}^{\square,II}(x,y) = \widehat{L}_i(x)\nabla(\widehat{L}_j(y)) = \begin{pmatrix} 0 \\ \widehat{L}_i(x)L_{j-1}(y) \end{pmatrix}, \quad \text{for } 1 \leq j \leq p, 2 \leq i \leq p. \tag{6.13}$$

Note that the functions of type III are now a special case of the new type I and II for $i = 1$ and $j = 1$, respectively. By application of (6.1) we again find the biorthogonal functions.

Table 6.1.1: Dual functions on a quadrilateral

Define

$$\begin{aligned} \tilde{b}_{kl}^{\square,I}(x,y) &:= \begin{pmatrix} L_{k-1}(x)P_{l-2}^{(1,1)}(y) \\ 0 \end{pmatrix}, \quad \text{for } 1 \leq k \leq p, 2 \leq l \leq p, \\ \tilde{b}_{kl}^{\square,II}(x,y) &:= \begin{pmatrix} 0 \\ P_{k-2}^{(1,1)}(x)L_{l-1}(y) \end{pmatrix}, \quad \text{for } 1 \leq l \leq p, 2 \leq k \leq p. \end{aligned} \tag{6.14}$$

Corollary 6.1.2
For $\tilde{v}_{ij}^{\square,I}$ and $\tilde{v}_{ij}^{\square,II}$ as in (6.12) and (6.13) and the functions $\tilde{b}_{kl}^{\square,I}$ and $\tilde{b}_{kl}^{\square,II}$ as in (6.14) are biorthogonal, i.e.

$$\int_{\square} \tilde{v}_{ij}^{\square,\omega_1}(x,y)\tilde{b}_{kl}^{\square,\omega_2}(x,y) = c\delta_{ik}\delta_{jl}\delta_{\omega_1,\omega_2}$$

Proof. The proof follows as in lemma 6.1.1. □

If we apply the dual functions from corollary 6.1.2 to (6.11), it follows

$$\begin{aligned} \langle v_{ij}^{\square,I}, \tilde{b}_{kl}^{\square,I} \rangle &= c\,\delta_{ik}\delta_{jl} \\ \langle v_{ij}^{\square,I}, \tilde{b}_{kl}^{\square,II} \rangle &= d\,\delta_{ik}\delta_{jl} \\ \langle v_{ij}^{\square,II}, \tilde{b}_{kl}^{\square,I} \rangle &= c\,\delta_{ik}\delta_{jl} \\ \langle v_{ij}^{\square,II}, \tilde{b}_{kl}^{\square,II} \rangle &= -d\,\delta_{ik}\delta_{jl}, \end{aligned}$$

where $c \neq 0$ and $d \neq 0$ are constants depending on i, j, k, l. Since all blocks are diagonal this motivates the choice

$$b_{ij}^{\square,II}(x,y) = d^{-1}\,\tilde{b}_{ij}^{\square,II}(x,y) - c^{-1}\,\tilde{b}_{\square,ij}^{I}(x,y).$$

and

$$b_{ij}^{\square,I}(x,y) = c^{-1}\,\tilde{b}_{ij}^{\square,I}(x,y) - d^{-1}\,\tilde{b}_{ij}^{\square,II}(x,y).$$

These yield orthogonality since

$$\langle v_{ij}^{\square,I}, b_{kl}^{II} \rangle = d^{-1}\langle v_{ij}^{\square,I}, \tilde{b}_{kl}^{\square,II} \rangle - c^{-1}\langle v_{ij}^{\square,I}, \tilde{b}_{kl}^{\square,I} \rangle = d^{-1}\,d\,\delta_{ik}\delta_{jl} - c^{-1}\,c\,\delta_{ik}\delta_{jl} = 0 \quad \forall i, j, k, l \geq 2$$

and

$$\langle v_{ij}^{\Box,II}, b_{kl}^{\Box,I}\rangle = c^{-1}\langle v_{ij}^{\Box,II}, \tilde{b}_{kl}^{\Box,I}\rangle - d^{-1}\langle v_{ij}^{\Box,II}, \tilde{b}_{kl}^{\Box,II}\rangle = -c^{-1}\, c\, \delta_{ik}\delta_{jl} + d^{-1}\, d\, \delta_{ik}\delta_{jl} = 0 \quad \forall i,j,k,l \geq 2.$$

Orthogonality of v_i^{III} and v_{i+p}^{III} to b_{kl}^{I} and b_{kl}^{II} is trivial. We summarize in the following lemma.

Lemma 6.1.3
Let $v_{ij}^{\Box,I}, v_{ij}^{\Box,II}, v_i^{\Box,III}$ and $v_{i+p}^{\Box,III}$ be as in (6.11), and $\tilde{b}_{kl}^{\Box,I}$ and $\tilde{b}_{kl}^{\Box,II}$ as in table 6.1.1. Furthermore, let

$$\alpha_{ij} = \frac{1}{8}(i)(2i-1)(2j-1). \tag{6.15}$$

Then the functions

$$\begin{aligned}
b_{ij}^{\Box,I}(x,y) &= \alpha_{ij}\,\tilde{b}_{ij}^{\Box,I}(x,y) - \alpha_{ji}\,\tilde{b}_{ij}^{\Box,II}(x,y),\\
b_{ij}^{\Box,II}(x,y) &= \alpha_{ij}\,\tilde{b}_{ij}^{\Box,I}(x,y) + \alpha_{ji}\,\tilde{b}_{ij}^{\Box,II}(x,y),\\
b_{i}^{\Box,III}(x,y) &= \tilde{b}_{1l}^{\Box,I}(x,y), \quad b_{i+p}^{\Box,III}(x,y) = \tilde{b}_{1l}^{\Box,II}(x,y),
\end{aligned}$$

are biorthogonal to $v_{ij}^{\Box,I}, v_{ij}^{\Box,II}, v_i^{\Box,III}$ and $v_{i+p}^{\Box,III}$.

Proof. Biorthogonality was already shown above. For the coefficients in (6.15) apply the usual orthogonality results, i.e.

$$c = \langle v_{ij}^{I}, b_{ij}^{I}\rangle = \int_{-1}^{1} L_{i-1}(x)L_{i-1}(x)\,\mathrm{d}x \int_{-1}^{1} \frac{y^2-1}{2(j-1)} P_{j-2}^{(1,1)}(y)P_{j-2}^{(1,1)}(y)\,\mathrm{d}y,$$

where

$$\int_{-1}^{1}(L_{i-1}(x))^2\,\mathrm{d}x = \frac{2}{2i-1}$$

and

$$\int_{-1}^{1} \frac{y^2-1}{2(j-1)}(P_{j-2}^{(1,1)}(y))^2\,\mathrm{d}y = \frac{-1}{2(j-1)}\frac{2^3}{2j-1}\frac{\Gamma(j)\Gamma(j)}{\Gamma(j+1)(j-2)!} = \frac{-4}{(j-1)(2j-1)}\frac{(j-1)}{j} = \frac{-4}{j(2j-1)}.$$

Choose $\alpha_{ji} = c^{-1}$. The coefficient α_{ij} is computed analogously. □

Triangular case

The triangular case is more complicated. Recall that the basis function of $H(\mathrm{curl})$ on the reference triangle with vertices $(-1,-1), (1,-1)$ and $(0,1)$ are derived as

$$\begin{aligned}
v_{ij}^{\triangle,I}(x,y) &= \nabla(u_{ij}^{\triangle}(x,y)) = \nabla(f_i(x,y))g_{ij}(y) + f_i(x,y)\nabla(g_{ij}(y))\\
v_{ij}^{\triangle,II}(x,y) &= \mathbb{W}_2(u_{ij}^{\triangle}(x,y)) = \nabla(f_i(x,y))g_{ij}(y) - f_i(x,y)\nabla(g_{ij}(y))\\
v_{1j}^{\triangle,III}(x,y) &= \nabla(f_1(x,y))\widehat{P}_j^3(y),
\end{aligned} \tag{6.16}$$

where $f_i(x,y) = \widehat{L}_i(\frac{2x}{1-y})\left(\frac{1-y}{2}\right)^i$ and $g_{ij}(y) = \widehat{P}_j^{2i}(y)$, see table 4.2.8.
The gradients of the auxiliary functions $f_i(x,y)$ and $g_{ij}(y)$ can be calculated as

$$\begin{aligned}
\nabla(f_i(x,y)) &= \left(\frac{1-y}{2}\right)^{(i-1)} \begin{pmatrix} L_{i-1}\left(\frac{2x}{1-y}\right) \\ \frac{1}{2}L_{i-2}\left(\frac{2x}{1-y}\right) \end{pmatrix} && \text{for } i \geq 2, \\
\nabla(g_{ij}(y)) &= \begin{pmatrix} 0 \\ P_{j-1}^{(2i,0)}(y) \end{pmatrix} && \text{for } i \geq 2, j \geq 1, \\
\nabla(f_1(x,y)) &= \frac{1-y}{4} \begin{pmatrix} 1 \\ \frac{1}{2}\frac{2x}{(1-y)} \end{pmatrix},
\end{aligned} \tag{6.17}$$

where we simplified the first gradient by using the recursive relation (4.7), see also [BS06].
We follow the ansatz as described for the quadrilateral case. First split $v_{ij}^{\triangle,I/II/III}$ in the functions

$$\begin{aligned}
\tilde{v}_{ij}^{\triangle,I}(x,y) &= \nabla(f_i(x,y))g_{ij}(y) && \text{for } i \geq 1, j \geq 2, \\
\tilde{v}_{ij}^{\triangle,II}(x,y) &= f_i(x,y)\nabla(g_{ij}(y)) && \text{for } i,j \geq 2.
\end{aligned} \tag{6.18}$$

The functions $\tilde{v}^{\triangle,I}(x,y)$ and $\tilde{v}^{\triangle,II}(x,y)$ are also a basis of the space $H_0(\text{curl})$. Next we derive the biorthogonal vectorial functions for those $\tilde{v}_{ij}^{\triangle,I}$ and $\tilde{v}_{ij}^{\triangle,II}$, and then solve the original problem by linear combination, as in the quadrilateral case.
Here the main idea of the construction is that we first find vectorial functions which are orthogonal to either $\nabla(g_{ij}(x,y))$ or $\nabla(f_i(x,y))$ and then biorthogonalise those to the respective other basis functions.
It is clear that we have the following structure of the orthogonal vectors:

$$\begin{aligned}
B_{kl}(x,y) &= \begin{pmatrix} b(x,y) \\ 0 \end{pmatrix} \\
C_{kl}(x,y) &= \begin{pmatrix} c_1(x,y) \\ c_2(x,y) \end{pmatrix}
\end{aligned} \tag{6.19}$$

In the following we use the notation $\eta = \frac{2x}{1-y}$ and write all functions in dependence of (η, y), e.g. write $a(x,y)$ as $a(\eta,y)$. The first problem which needs to be solved then reads:

Problem 6.1.2
Find polynomials $B_{kl}(\eta,y)$ such that

$$\langle \tilde{v}_{ij}^{\triangle,II}(\eta,y), B_{kl}(\eta,y)\rangle = 0 \text{ and } \langle \tilde{v}^{\triangle,I}, B_{kl}(\eta,y)\rangle = d_{ijkl}^{(1)}\delta_{ik}\delta_{jl}.$$

Since the first component of $\tilde{v}^{\triangle,II}(\eta,y)$ is zero, $B_{kl}(\eta,y)$ as in (6.19) naturally fulfills the first condition. Furthermore, we can assume a tensorial-like structure, i.e.

$$B_{kl}(\eta,y) = \begin{pmatrix} b_k^{(1)}(\eta)b_{kl}^{(2)}(y) \\ 0 \end{pmatrix}.$$

Now $b_1(z)$ and $b_2(y)$ only needs to fulfil the relationship

$$\int_{-1}^{1} L_{i-1}(\eta)\, b_k^{(1)}(\eta)\mathrm{d}\eta \int_{-1}^{1} \left(\frac{1-y}{2}\right)^i \frac{(1+y)}{2j} P_{j-1}^{(2i-1,1)}(y)\, b_{kl}^{(2)}(y)\,\mathrm{d}y = d_{ijkl}^{(1)}\delta_{ik}\delta_{jl},$$

where we applied the Duffy transformation. This motivates the choice $b_k^{(1)}(\eta) = L_{k-1}(\eta)$ and $b_{kl}^{(2)}(y) = \left(\frac{1-y}{2}\right)^{k-1} P_{l-1}^{(2k-1,1)}(y)$.
For the second type of dual functions we need to solve the following problem:

Problem 6.1.3
Find polynomials $C_{kl}(\eta, y)$ such that,

$$\langle \tilde{v}_{ij}^{\triangle,I}(\eta, y), C_{kl}(\eta, y)\rangle = 0 \text{ and } \langle \tilde{v}_{ij}^{\triangle,II}(\eta, y), C_{kl}(\eta, y)\rangle = d_{ijkl}^{(2)}\delta_{ik}\delta_{jl}. \tag{6.20}$$

Here we will need the help of the following small lemma.

Lemma 6.1.4
For $1 \leq i \leq k$, the relation

$$\int_{-1}^{1} L_i(x)P_k^{(1,1)}(x)\,\mathrm{d}x = \begin{cases} \frac{4}{2+k} & \text{if } k \geq i \text{ and } (k-i) \bmod 2 = 0 \\ 0 & \text{else} \end{cases}$$

holds.

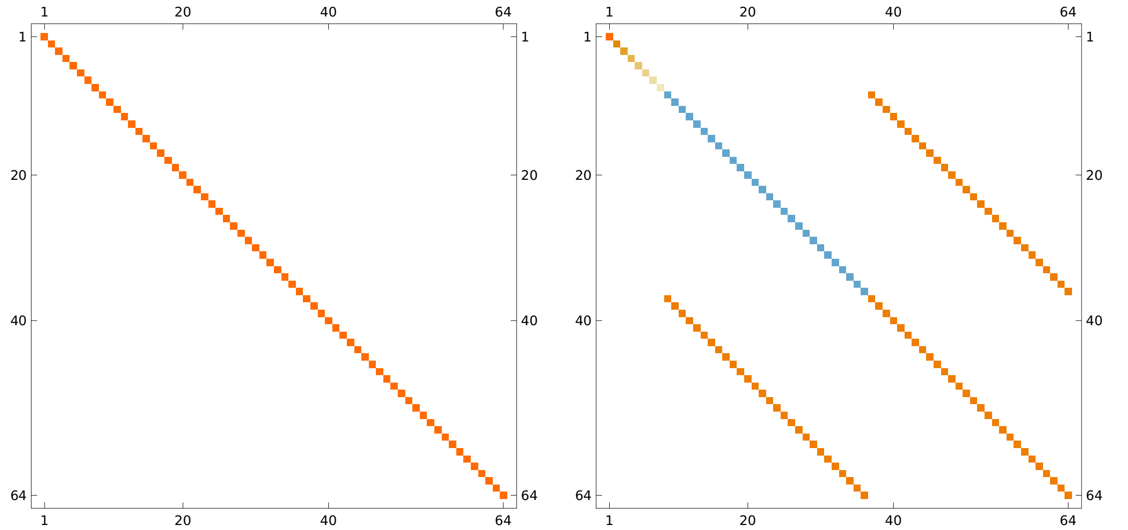
Proof. A classical result of Jacobi polynomials states that $P_n^{(\alpha,\alpha)}(x)$ is even if n is even and it is odd if n is odd. Thus the relation is trivial if k and i have a different parity.
In the following, assume i and k have the same parity. Let $I_{ik} := \int_{-1}^{1} L_i(x)P_k^{(1,1)}(x)\,\mathrm{d}x$. If $i > k$ it follows that I_{ik} is zero, due to the orthogonality condition of $L_i(x)$. Now assume $k \geq i$. By partial integration it follows

$$\begin{aligned} I_{ik} = \int_{-1}^{1} L_i(x)P_k^{(1,1)}(x)\,\mathrm{d}x &= \int_{-1}^{1} L_i(x)\frac{2}{2+k}\frac{\mathrm{d}}{\mathrm{d}x}L_{k+1}(x)\,\mathrm{d}x \\ &= \frac{2}{2+k}\left[L_i(x)L_{k+1}(x)\right]\big|_{-1}^{1} - \int_{-1}^{1}\left(\frac{\mathrm{d}}{\mathrm{d}x}L_i(x)\right)L_{k+1}(x)\,\mathrm{d}x \\ &= \frac{4}{2+k}, \end{aligned}$$

where the last integral vanishes due to the orthogonality condition of $L_{k+1}(x)$ and $[L_i(x)L_{k+1}(x)]|_{-1}^{1} = 2$ due to the odd parity. □

For problem 6.1.3 we start with the biorthogonality condition. We again assume a tensorial-like structure, i.e.

$$C_{kl}(\eta, y) = c_{kl}^{(3)}(y)\begin{pmatrix} c_k^{(1)}(\eta) \\ c_k^{(2)}(\eta) \end{pmatrix}.$$



(a) L^2-scalar product of B_{kl}, C_{kl} and auxiliary functions.

(b) L^2-scalar product of B_{kl}, C_{kl} and $H(\text{curl})$ basis functions.

Figure 6.1.: Biorthogonal sparsity pattern for $p = 6$

The condition is

$$\langle \tilde{v}_{ij}^{\triangle,II}(\eta, y), C_{kl}(\eta, y)\rangle = \int_{-1}^{1} \widehat{L}_i(\eta) c_k^{(2)}(\eta)\, \mathrm{d}\eta \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+1} P_{j-1}^{(2i,0)}(y) c_{kl}^{(3)}(y)\, \mathrm{d}y = d_{ijkl}^{(2)} \delta_{ik}\delta_{jl}.$$

This leads to the choice $c_k^{(2)}(\eta) = \kappa P_{k-2}^{(1,1)}(\eta)$ and $c_{kl}^{(3)}(y) = \left(\frac{1-y}{2}\right)^{k-1} P_{l-1}^{(2k,0)}(y)$, where κ is some constant. The orthogonality condition

$$\langle \tilde{v}_{ij}^{\triangle,I}, C_{kl}\rangle = 0$$

is satisfied, if

$$\langle \nabla f_i(\eta, y), C_{kl}(\eta)\rangle = 0. \tag{6.21}$$

Since both components of ∇f_i depend on $\left(\frac{1-y}{2}\right)^{i-1} \widehat{P}_j^{2i,0}(y)$, the orthogonality relation reduces to

$$\int_{-1}^{1} L_{i-1}(\eta) c_k^{(1)}(\eta)\, \mathrm{d}\eta + \int_{-1}^{1} \frac{1}{2} L_{i-2}(\eta) c_k^{(2)}(\eta)\, \mathrm{d}\eta = 0.$$

Due to lemma 6.1.4 this condition is fulfilled if $c_k^{(1)}(\eta) = (2 + k - 1) P_{k-1}^{(1,1)}(\eta)$ and $c_k^{(2)}(\eta) = -2(2 + k - 2) P_{k-2}^{(1,1)}(\eta)$. Now we have found a biorthogonal basis for $\tilde{v}^{\triangle,I}$ and $\tilde{v}^{\triangle,II}$, which results in a diagonal matrix which can be seen in Figure 6.1a.

Table 6.1.2: Dual functions on a triangle

Define

$$B_{kl}(x,y) := \begin{pmatrix} L_{k-1}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{k-1} P_{l-1}^{2k-1,1}(y) \\ 0 \end{pmatrix}, \qquad \text{for } 1 \le k, l$$

$$C_{kl}(x,y) := \begin{pmatrix} (2+k-1)P_{k-1}^{(1,1)}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{k-1} P_{l-1}^{(2k,0)}(y) \\ -2(2+k-2)P_{k-2}^{(1,1)}\left(\frac{2x}{1-y}\right)\left(\frac{1-y}{2}\right)^{k-1} P_{l-1}^{(2k,0)}(y) \end{pmatrix}, \qquad \text{for } 2 \le k, 1 \le l.$$

With this choice, we have proven the following corollary.

Corollary 6.1.5
Let $\tilde{v}_{ij}^{\triangle,I}, \tilde{v}_{ij}^{\triangle,II}$ be defined by (6.18) and the functions B_{kl} and C_{kl} be defined as in table 6.1.2. Then these functions are biorthogonal.

Obviously those are not biorthogonal to the basis $v_{ij}^{\triangle,I}, v_{ij}^{\triangle,II}$ and $v_{1j}^{\triangle,III}$, which can be seen in Figure 6.1b. Thus, we derive the biorthogonal functions by linear combination, as in the quadrilateral case.

Lemma 6.1.6
Let the coefficients α_1, α_2 and α_3 be given by

$$\begin{aligned} \alpha_1 &= \frac{1}{8}(2i-1)(2j+2i-1)(j+2i-1), \\ \alpha_2 &= \frac{1}{16}(2i-1)(2j+2i-1), \\ \alpha_3 &= \frac{1}{16}(2j+2)(j+2). \end{aligned}$$

Then for $2 \le i, k \le p, 1 \le j, l \le p$, and $i+j, k+l \le p$, the functions $v_{ij}^{\triangle,I}, v_{ij}^{\triangle,II}$ and $v_{1j}^{\triangle,III}$ as in (6.16) are biorthogonal to

$$\begin{aligned} b_{kl}^{\triangle,I}(x,y) &= -\frac{1}{2}(\alpha_1 B_{kl}(x,y) + \alpha_2 C_{kl}(x,y)), \\ b_{kl}^{\triangle,II}(x,y) &= \frac{1}{2}(\alpha_1 B_{kl}(x,y) - \alpha_2 C_{kl}(x,y)), \\ b_{1l}^{\triangle,III}(x,y) &= \alpha_3 B_{1l}(x,y), \end{aligned}$$

where B_{kl} and C_{kl} are given in table 6.1.2.

Proof. Since we have shown biorthogonality of $\tilde{v}_{ij}^{\triangle,I}, \tilde{v}_{ij}^{\triangle,II}$ to B_{kl}, C_{kl} in corollary 6.1.5, we get

$$\begin{aligned} \langle v_{ij}^{\triangle,I}, B_{kl}\rangle &= \langle \tilde{v}_{ij}^{\triangle,I}, B_{kl}\rangle + \langle \tilde{v}_{ij}^{\triangle,II}, B_{kl}\rangle = \langle \tilde{v}_{ij}^{\triangle,I}, B_{kl}\rangle = c\delta_{ik}\delta_{jl} \\ \langle v_{ij}^{\triangle,I}, C_{kl}\rangle &= \langle \tilde{v}_{ij}^{\triangle,I}, C_{kl}\rangle + \langle \tilde{v}_{ij}^{\triangle,II}, C_{kl}\rangle = \langle \tilde{v}_{ij}^{\triangle,II}, C_{kl}\rangle = \tilde{c}\delta_{ik}\delta_{jl}. \end{aligned}$$

This motivates the choice $b_{kl}^{\triangle,I} = \alpha_1 B_{kl} + \alpha_2 C_{kl}$ with some coefficients α_1, α_2 depending on i, j, k, l.

The parameters α_1, α_2 are chosen such that

$$0 \neq \langle v_{ij}^{\triangle,I}, b_{kl}^{\triangle,I} \rangle = \alpha_1 \langle \tilde{v}_{ij}^{\triangle,I}, B_{kl} \rangle + \alpha_2 \langle \tilde{v}_{ij}^{\triangle,II}, C_{kl} \rangle,$$
$$0 = \langle v_{ij}^{\triangle,II}, b_{kl}^{\triangle,I} \rangle = \alpha_1 \langle \tilde{v}_{ij}^{\triangle,I}, B_{kl} \rangle - \alpha_2 \langle \tilde{v}_{ij}^{\triangle,II}, C_{kl} \rangle.$$

We only need to solve this for $(i, j) = (k, l)$, thus

$$\begin{aligned}
\langle \tilde{v}_{ij}^{\triangle,I}, B_{ij} \rangle &= \int_{-1}^{1} (L_{i-1}(\eta))^2 \,\mathrm{d}\eta \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{2i-1} \frac{(1+y)}{j} (P_{j-1}^{(2i-1,1)}(y))^2 \,\mathrm{d}y \\
&\overset{(6.1.3)}{=} \frac{2}{2i-1} \left(\frac{1-y}{2}\right)^{2i-1} \frac{1+y}{j} (P_{j-1}^{(2i-1,1)}(y))^2 \,\mathrm{d}y \\
&= \frac{2}{2i-1} \frac{1}{2^{2i-1} j} \frac{2^{2i+1}}{2j+2i-1} \frac{\Gamma(j+2i-1)\Gamma(j+1)}{\Gamma(j+2i)(j-1)!} \\
&= \frac{8}{(2i-1)(2j+2i-1)(j+2i-1)}.
\end{aligned}$$

This implies $\alpha_1 = \frac{1}{8}(2i-1)(2j-2i-1)(j+2i-1)$. Analogously

$$\begin{aligned}
\langle \tilde{v}_{ij}^{\triangle,II}, C_{ij} \rangle &= -2i \int_{-1}^{1} \left(\frac{\eta^2-1}{2(i-1)}\right) (P_{i-2}^{(1,1)}(\eta))^2 \,\mathrm{d}\eta \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{2i} P_{j-1}^{(2i,0)}(y) \,\mathrm{d}y \\
&= \frac{8}{(2i-1)} \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{2i} P_{j-1}^{(2i,0)}(y) \,\mathrm{d}y \\
&= \frac{8}{(2i-1)} \frac{2}{(2j+2i-1)} \frac{\Gamma(j+2i)\Gamma(j)}{\Gamma(j+2i)(j-1)!} \\
&= \frac{16}{(2i-1)(2j+2i-1)},
\end{aligned}$$

implies $\alpha_2 = \frac{1}{16}(2i-1)(2j+2i-1)$. The only thing remaining to show is that B_{kl}, C_{kl} are orthogonal to $v_{1,j}^{\triangle,III}$ and that $b_{1l}^{\triangle,III}$ is orthogonal to $\tilde{v}_{ij}^{\triangle,I}$ and $\tilde{v}_{ij}^{\triangle,II}$. Indeed,

$$\langle v_{1j}^{\triangle,III}, B_{kl} \rangle = \int_{-1}^{1} 1 \cdot L_{k-1}(\eta) \,\mathrm{d}\eta \int_{-1}^{1} \frac{1}{2} \left(\frac{1-y}{2}\right)^{k} P_{l-1}^{(2k-1,1)}(y) = 0,$$

due to the orthogonality of $L_{k-1}(\eta)$ for all $k \geq 2$. For the orthogonality of

$$\langle v_{1j}^{\triangle,III}, C_{kl} \rangle$$

we can apply lemma 6.1.4.
On the other hand, orthogonality of $b_{1l}^{\triangle,III}$ follows from corollary 6.1.5. □

Since all basis functions are properly scaled, the next corollary follows directly.

Corollary 6.1.7
Let $i,k \geq 2$ and $j,l \geq 1$. Furthermore let $v_{ij}^{\triangle,I}, v_{ij}^{\triangle,II}, v_{1j}^{\triangle,III}$ be the basis of the $H(\text{curl})$ face functions and let $b_{kl}^{\triangle,I}, b_{kl}^{\triangle,II}, b_{1l}^{\triangle,III}$ be the corresponding normalized dual face functions, then holds for the entries of the element Matrix G that

$$G_{ij,kl} = \langle v_{ij}^{\triangle,T_1}(x,y), b_{kl}^{\triangle,T_2}(x,y)\rangle = \delta_{i,k}\delta_{j,l}\delta_{T_1,T_2}.$$

As we have seen before, the functions based on the operator $\mathbb{W}$ are not functions of Nédélec's space of the first kind. But $v^{\triangle,\mathcal{N}}$ as in (4.13) are just a small recombination, and thus it is possible to reconstruct our dual functions to those modified basis functions. The same arguments as above lead to the following corollary:

Corollary 6.1.8
Let the coefficients $\alpha_1, \alpha_2, \alpha_3$ and c_{ij} be given by

$$\begin{aligned}
\alpha_1 &= \frac{1}{8}(2i-1)(2j+2i-1)(j+2i-1),\\
\alpha_2 &= \frac{1}{16}(2i-1)(2j+2i-1),\\
\alpha_3 &= \frac{1}{16}(2j+2)(j+2),\\
c_{ij} &= \frac{i}{j}.
\end{aligned}$$

Then for $2 \leq i,k \leq p, 1 \leq j,l \leq p$, and $i+j, k+l \leq p$, the functions $v_{ij}^{\triangle,I}, v_{1j}^{\triangle,III}$ as in (6.16) and $v^{\triangle,\mathcal{N}}$ as in (4.13) are biorthogonal to

$$\begin{aligned}
b_{kl}^{\triangle,I}(x,y) &= \frac{1}{1+c_{ji}}(\alpha_1 B_{kl}(x,y) + c_{ji}\alpha_2 C_{kl}(x,y)),\\
b_{kl}^{\triangle,II}(x,y) &= \frac{1}{1+c_{ij}}(\alpha_1 B_{kl}(x,y) - \alpha_2 C_{kl}(x,y)),\\
b_{1l}^{\triangle,III}(x,y) &= \alpha_3 B_{1l}(x,y),
\end{aligned}$$

where B_{kl} and C_{kl} are given in table 6.1.2.

Tetrahedral case

Our ansatz is the same as in the triangular case:
We split the basis functions in a simpler basis, find the orthogonal basis to this simpler basis and build the right dual basis by linear combination. Recall the basis of $H(\text{curl})$ on the tetrahedral with vertices $(-1,-1,-1), (1,-1,-1), (0,1,-1)$ and $(0,0,1)$ is given by

$$\begin{aligned}
v_{ijk}^{I} &= \nabla(f_i g_{ij} h_{ijk}) = \nabla(f_i) g_{ij} h_{ijk} + f_i \nabla(g_{ij}) h_{ijk} + f_i g_{ij} \nabla(h_{ijk}),\\
v_{ijk}^{II} &= \mathbb{W}_2(f_i g_{ij} h_{ijk}) = \nabla(f_i) g_{ij} h_{ijk} - f_i \nabla(g_{ij}) h_{ijk} + f_i g_{ij} \nabla(h_{ijk}),\\
v_{ijk}^{III} &= \mathbb{W}_3(f_i g_{ij} h_{ijk}) = \nabla(f_i) g_{ij} h_{ijk} + f_i \nabla(g_{ij}) h_{ijk} - f_i g_{ij} \nabla(h_{ijk}),\\
v_{1jk}^{IV} &= v_{[1,2]}^{\mathcal{N}_0} g_{ij} h_{ijk},
\end{aligned} \tag{6.22}$$

for $i \geq 2; j,k \geq 1; i+j+k \leq p$, where

$$\begin{aligned} f_i &= \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-2y-z}{4}\right)^i, \\ g_{ij} &= \widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^j, \\ h_{ijk} &= \widehat{P}_k^{2i+2j}(z) \end{aligned} \tag{6.23}$$

and $v_{[1,2]}^{\mathcal{N}_0}$ is the lowest order Nédélec function of first kind, based on the edge from vertex 1 to 2, see table 4.2.10. With the substitutions $\eta = \frac{4x}{1-2y-z}$ and $\chi = \frac{2y}{1-z}$, the gradients of the auxiliary functions are

$$\begin{aligned} \nabla f_i &= \begin{pmatrix} L_{i-1}(\eta) \\ \frac{1}{2}L_{i-2}(\eta) \\ \frac{1}{4}L_{i-2}(\eta) \end{pmatrix} \left(\frac{1-\chi}{2}\right)^{i-1} \left(\frac{1-z}{2}\right)^{i-1}, \\ \nabla g_{ij} &= \begin{pmatrix} 0 \\ P_{j-1}^{(2i,0)}(\chi) \\ \frac{\chi}{2}P_{j-1}^{(2i,0)}(\chi) - \frac{j}{2}\widehat{P}_j^{2i}(\chi) \end{pmatrix} \left(\frac{1-z}{2}\right)^{j-1} \text{ and} \\ \nabla h_{ijk} &= \begin{pmatrix} 0 \\ 0 \\ P_{k-1}^{(2i+2j,0)}(z) \end{pmatrix}. \end{aligned}$$

We now derive three biorthogonal vectors with respect to $(\nabla f_i)\, g_{ij}\, h_{ijk}$, $f_i\, (\nabla g_{ij})\, h_{ijk}$ and $f_i\, g_{ij}\, (\nabla h_{ijk})$. Similar to the triangular case, we have the following conditions on the dual functions B_{ijk}, C_{ijk} and D_{ijk} :

Problem 6.1.4
Find $B_{lmn}, C_{lmn}, D_{lmn}$ such that

$$\begin{aligned} f_i\,(\nabla g_{ij})\,h_{ijk} \perp B_{lmn} \perp f_i\,g_{ij}\,(\nabla h_{ijk}) &\text{ and } \langle B_{lmn}, (\nabla f_i)\,g_{ij}\,h_{ijk}\rangle = r_{ijklmn}\delta_{il}\delta_{jm}\delta_{kn}, \\ (\nabla f_i)\,g_{ij}\,h_{ijk} \perp C_{lmn} \perp f_i\,g_{ij}\,(\nabla h_{ijk}) &\text{ and } \langle C_{lmn}, f_i\,(\nabla g_{ij})\,h_{ijk}\rangle = s_{ijklmn}\delta_{il}\delta_{jm}\delta_{kn}, \\ (\nabla f_i)\,g_{ij}\,h_{ijk} \perp D_{lmn} \perp f_i\,(\nabla g_{ij})\,h_{ijk} &\text{ and } \langle D_{lmn}, f_i\,g_{ij}\,(\nabla h_{ijk})\rangle = t_{ijklmn}\delta_{il}\delta_{jm}\delta_{kn}. \end{aligned}$$

Since the basis vectors build a lower triangular system, we know that we need to find an upper triangular system. Thus,

$$\begin{aligned} B_{lmn} &= \begin{pmatrix} b_{lmn}(\eta,\chi,z) \\ 0 \\ 0 \end{pmatrix}, \\ C_{lmn} &= \begin{pmatrix} c_{lmn}^{(1)}(\eta,\chi,z) \\ c_{lmn}^{(2)}(\eta,\chi,z) \\ 0 \end{pmatrix}, \\ D_{lmn} &= \begin{pmatrix} d_{lmn}^{(1)}(\eta,\chi,z) \\ d_{lmn}^{(2)}(\eta,\chi,z) \\ d_{lmn}^{(3)}(\eta,\chi,z) \end{pmatrix}. \end{aligned}$$

The construction of B_{lmn} is trivial and the construction of C_{lmn} follows from the $2D$ case. Thus,

$$B_{lmn} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} L_{l-1}(\eta) \left(\frac{1-\chi}{2}\right)^{l-1} P_{m-1}^{(2l-1,1)}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P_{n-1}^{(2l+2m-1,1)}(z) \text{ and}$$

$$C_{lmn} = \begin{pmatrix} (2+l-1)P_{l-1}^{(1,1)}(\eta) \\ -2(2+l-2)P_{l-2}^{(1,1)}(\eta) \\ 0 \end{pmatrix} \left(\frac{1-\chi}{2}\right)^{l-1} P_{m-1}^{(2l,0)}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P_{n-1}^{(2l+2m-1,1)}(z),$$

where the exponents of $\left(\frac{1-\chi}{2}\right)$ and $\left(\frac{1-z}{2}\right)$ are determined with respect to the functional determinant of the Duffy trick, i.e. $\left(\frac{1-\chi}{2}\right)\left(\frac{1-z}{2}\right)^2$.
To derive D_{lmn} we go step by step. It follows immediately that

$$d_{lmn}^{(3)} = P_{l-2}^{(1,1)}(\eta) \left(\frac{1-\chi}{2}\right)^{l-2} P_{m-1}^{(2l-1,1)}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P_{n-1}^{(2l+2m,0)}(z), \tag{6.24}$$

due to $\langle D_{lmn}, f_i\, g_{ij}\, (\nabla h_{ijk})\rangle = s_{ijklmn}\delta_{il}\delta_{jm}\delta_{kn}$.
Next we derive $d_{lmn}^{(2)}$ by demanding

$$0 = \langle f_i(\nabla g_{ij})h_{ijk}, D_{lmn}\rangle. \tag{6.25}$$

A rather obvious choice for $d_{lmn}^{(2)}$ is $d_{lmn}^{(2)} = P_{l-2}^{(1,1)}(\eta)\, \tilde{d}^{(2)}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P_{n-1}^{(2l+2m,0)}(z)$. This choice reduces the condition (6.25) to

$$0 = \int_{-1}^{1} \left(\frac{1-\chi}{2}\right)^{i+1} P_{j-1}^{(2i,0)}(\chi)\, \tilde{d}^{(2)}(\chi) + \left(\frac{1-\chi}{2}\right)^{2i-1} \left(\frac{\chi}{2} P_{j-1}^{(2i,0)}(\chi) - \frac{j}{2}\widehat{P}_j^{2i}(\chi)\right) P_{m-1}^{(2i-1,1)}(\chi)\, \mathrm{d}\chi, \tag{6.26}$$

since the condition (6.25) is trivial for $i \neq l$ and $k \neq n$ with this choice of $d_{ijk}^{(2)}$.
On the right part of the integral the combination $\widehat{P}_j^{2i}(\chi)P_{m-1}^{(2i-1,1)}(\chi)$ already fulfils the orthogonality relation. On the other hand, for the product of the two different Jacobi polynomials, namely $P_{j-1}^{(2i,0)}(\chi)$ and $P_{m-1}^{(2i-1,1)}(\chi)$, we can't apply standard orthogonality results. We eliminate this mixed part by linear combination. Therefore, we choose

$$\tilde{d}^{(2)} = -\frac{\chi}{2}\left(\frac{1-\chi}{2}\right)^{l-2} P_{m-1}^{(2l-1,1)}(\chi) + \hat{d}^{(2)},$$

such that the mixed products cancel each other out. Those linear combinations result in the further reduced condition

$$0 = \int_{-1}^{1} \left(\frac{1-\chi}{2}\right)^{i+1} P_{j-1}^{(2i,0)}(\chi)\, \hat{d}^{(2)}(\chi) - \frac{j}{2}\left(\frac{1-\chi}{2}\right)^{2i-1} \widehat{P}_j^{2i}(\chi) P_{m-1}^{(2i-1,1)}(\chi)\, \mathrm{d}\chi. \tag{6.27}$$

The last part of the integral in (6.27) only appears if $m = j$. We can achieve the same for the first part of the integral, if we choose

$$\hat{d}^{(2)} = c\left(\frac{1-\chi}{2}\right)^{l-1} P_{m-1}^{(2l,0)}(\chi).$$

Since both instances in (6.27) are integrals over Jacobi polynomials with matching indices, order and weights, we can determine the constant c directly. It holds that

$$\int_{-1}^{1}\left(\frac{1-\chi}{2}\right)^{2i}\left(P_{j-1}^{(2i,0)}(\chi)\right)^2 \,\mathrm{d}\chi = \frac{1}{2j+2i-1}$$
$$\int_{-1}^{1}\left(\frac{1-\chi}{2}\right)^{2i-1}\left(\frac{1+\chi}{2}\right)\left(P_{j-1}^{(2i-1,1)}(\chi)\right)^2 \,\mathrm{d}\chi = \frac{j}{(2j+2i-1)(2i+j-1)}.$$

Collecting everything

$$d_{lmn}^{(2)} = P_{l-2}^{(1,1)}(\eta)\left(\frac{1-\chi}{2}\right)^{l-2}\left(-\frac{\chi}{2}P_{m-1}^{(2l-1,1)}(\chi)+\frac{m}{(2l+m-1)}\frac{(1-\chi)}{2}P_{m-1}^{(2l,0)}(\chi)\right)\left(\frac{1-z}{2}\right)^{l+m-2}P_{n-1}^{(2l+2m,0)}(z).$$

Now we need to determine $d_{lmn}^{(1)}$, by

$$0 = \langle(\nabla f_i)g_{ij}h_{ijk}, D_{lmn}\rangle.$$

Inserting $d_{lmn}^{(2)}$ and $d_{lmn}^{(3)}$ yields the following condition after some simplification

$$\begin{aligned}0 = &\int_{(-1,1)^3} L_{i-1}(\eta)\left(\frac{1-\chi}{2}\right)^{i-1}\widehat{P}_j^{2i}(\chi)\left(\frac{1-z}{2}\right)^{i+j+1}\widehat{P}_k^{2i+2j}(z)d_{lmn}^{(1)}(\eta,\chi,z)\,\mathrm{d}\eta\,\mathrm{d}\chi\,\mathrm{d}z\\ &+\int_{(-1,1)^3} L_{i-2}(\eta)P_{l-2}^{(1,1)}(\eta)\left(\frac{1-\chi}{2}\right)^{i+l-1}\widehat{P}_j^{2i}(\chi)\left[P_{m-1}^{(2l-1,1)}(\chi)+\frac{m}{2l+m-1}P_{m-1}^{(2l,0)}(\chi)\right]\\ &\left(\frac{1-z}{2}\right)^{i+j+l+m-1}\widehat{P}_k^{2i+2j}(z)P_{n-1}^{(2l+2m-1,1)}(z)\,\mathrm{d}z\,\mathrm{d}\chi\,\mathrm{d}\eta.\end{aligned}$$

If we choose $d_{lmn}^{(1)}(\eta,\chi,z) = \frac{2+l-1}{2(2+l-2)}P_{l-1}^{(1,1)}(\eta)\hat{d}^{(1)}(\chi)\left(\frac{1-z}{2}\right)^{l+m-2}P_{n-1}^{(2l+2m,0)}(z)$, we can factor all terms depending on η and z out. Thus, we only need to determine $\hat{d}^{(1)}(\chi)$, by the condition

$$0 = \int_{-1}^{1}\left(\frac{1-\chi}{2}\right)^{i-1}\widehat{P}_j^{2i}(\chi)\,\hat{d}^{(1)}(\chi)+\left(\frac{1-\chi}{2}\right)^{i+l-1}\widehat{P}_j^{2i}(\chi)\left[P_{m-1}^{(2l-1,1)}(\chi)+\frac{m}{2l+m-1}P_{m-1}^{(2l,0)}(\chi)\right]\mathrm{d}\chi.$$

It is obvious, that we can choose

$$d_{lmn}^{(1)} = \frac{-(2+l-1)}{2(2+l-2)}P_{l-1}^{(1,1)}(\eta)\left(\frac{1-\chi}{2}\right)^{l-1}\left[P_{m-1}^{(2l-1,1)}(\chi)+\frac{m}{2l+m-1}P_{m-1}^{(2l,0)}(\chi)\right]\left(\frac{1-z}{2}\right)^{l+m-2}P_{n-1}^{(2l+2m,0)}(z).$$

We summarize in the following table.

Table 6.1.3: Dual basis on the tetrahedron

Let $l \geq 2, m, n \geq 1$ then define

$$\tilde{b}^{\blacktriangle,I}_{lmn} := \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} L_{l-1}(\eta) \left(\frac{1-\chi}{2}\right)^{l-1} P^{(2l-1,1)}_{m-1}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P^{(2l+2m-1,1)}_{n-1}(z)$$

$$\tilde{b}^{\blacktriangle,II}_{lmn} := \begin{pmatrix} (l+1)P^{(1,1)}_{l-1}(\eta) \\ -2lP^{(1,1)}_{l-2}(\eta) \\ 0 \end{pmatrix} \left(\frac{1-\chi}{2}\right)^{l-1} P^{(2l,0)}_{m-1}(\chi) \left(\frac{1-z}{2}\right)^{l+m-2} P^{(2l+2m-1,1)}_{n-1}(z),$$

$$\tilde{b}^{\blacktriangle,III}_{lmn} := \begin{pmatrix} \frac{-(l+1)}{2l} P^{(1,1)}_{l-1}(\eta) \left(\frac{1-\chi}{2}\right)^{l-1} \left[P^{(2l-1,1)}_{m-1}(\chi) + \frac{m}{2l+m-1} P^{(2l,0)}_{m-1}(\chi)\right] \\ P^{(1,1)}_{l-2}(\eta) \left(\frac{1-\chi}{2}\right)^{l-2} \left[-\frac{\chi}{2} P^{(2l-1,1)}_{m-1}(\chi) + \frac{m}{2l+m-1} \frac{1-\chi}{2} P^{(2l,0)}_{m-1}(\chi)\right] \\ P^{(1,1)}_{l-2}(\eta) \left(\frac{1-\chi}{2}\right)^{l-2} P^{2l-1,1}_{m-1}(\chi) \end{pmatrix} \left(\frac{1-z}{2}\right)^{l+m-2} P^{(2l+2m,0)}_{n-1}(z),$$

where $\eta = \frac{4x}{1-2y-z}$ and $\chi = \frac{1-2y-z}{4}$.

Thus the following lemma has been shown.

Lemma 6.1.9
Let f_i, g_{ij} and h_{ijk} be defined as in (6.23). Then, the functions

$$\begin{aligned} \tilde{v}^{\blacktriangle,I}_{ijk} &= (\nabla f_i) g_{ij} h_{ijk}, \\ \tilde{v}^{\blacktriangle,II}_{ijk} &= f_i (\nabla g_{ij}) h_{ijk}, \\ \tilde{v}^{\blacktriangle,III}_{ijk} &= f_i g_{ij} (\nabla h_{ijk}) \end{aligned}$$

are biorthogonal to $\tilde{b}^{\blacktriangle,I}_{lmn}, \tilde{b}^{\blacktriangle,II}_{lmn}$ and $\tilde{b}^{\blacktriangle,III}_{lmn}$ defined by table 6.1.3, i.e.

$$\langle \tilde{v}^{\blacktriangle,\omega_1}_{ijk}, \tilde{b}^{\blacktriangle,\omega_2}_{lmn} \rangle = c^{\omega_1}_{ijk} \delta_{il} \delta_{jm} \delta_{kn} \delta_{\omega_1,\omega_2}, \quad \omega_1, \omega_2 \in \{I, II, III\}.$$

As before, we transfer this to the original interior basis functions.

Lemma 6.1.10
Let $\tilde{b}^{\blacktriangle,I}_{lmn}$, $\tilde{b}^{\blacktriangle,I}_{lmn}$ and $\tilde{b}^{\blacktriangle,I}_{lmn}$ be defined by table 6.1.3.
For $i \geq 2, j \geq 1, k \geq 1$ the functions $v^{\blacktriangle,I}_{ijk}, v^{\blacktriangle,II}_{ijk}, v^{\blacktriangle,III}_{ijk}$ and $v^{\blacktriangle,IV}_{1jk}$ are biorthogonal to

$$
\begin{aligned}
b^{\blacktriangle,I}_{lmn} &= \frac{1}{2}\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + \frac{1}{2}\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn},\\
b^{\blacktriangle,II}_{lmn} &= \frac{1}{2}\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} - \frac{1}{2}\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn},\\
b^{\blacktriangle,III}_{lmn} &= \frac{1}{2}\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} - \frac{1}{2}\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn},\\
b^{\blacktriangle,IV}_{1mn} &= \alpha^{(4)}_{lmn}\tilde{b}^{\blacktriangle,I}_{1mn},
\end{aligned}
$$

where

$$
\begin{aligned}
\alpha^{(1)}_{lmn} &= \frac{1}{2^7}(2l-1)(2m+2l-1)(m+2l-1)(2n+2l+2m-1)(n+2l+2m-1)\\
\alpha^{(2)}_{lmn} &= \frac{1}{2^6}(2l-1)(2m+2l-1)(2n+2l+2m-1)(n+2l+2m-1)\\
\alpha^{(3)}_{lmn} &= \frac{-1}{2^5}l(2l-1)(2m+2l-1)(m+2l-1)(2n+2l+2m-1)\\
\alpha^{(4)}_{lmn} &= \frac{1}{2^5}(2m+2)(m+2)(n+2m+2)(2n+2m+2)
\end{aligned}
$$

Proof. To show the coefficients $\alpha^{(1)}_{lmn}, \alpha^{(2)}_{lmn}$ and $\alpha^{(3)}_{lmn}$ we need to calculate the coefficients $c^{I}_{ijk}, c^{II}_{ijk}, c^{III}_{ijk}$ in lemma 6.1.9, since

$$
\langle v^{\blacktriangle,I}_{ijk}, \tilde{b}^{\blacktriangle,I}_{lmn}\rangle = \langle \tilde{v}^{\blacktriangle,I}_{ijk}, \tilde{b}^{\blacktriangle,I}_{lmn}\rangle + \langle \tilde{v}^{\blacktriangle,II}_{ijk}, \tilde{b}^{\blacktriangle,I}_{lmn}\rangle + \langle \tilde{v}^{\blacktriangle,III}_{ijk}, \tilde{b}^{\blacktriangle,I}_{lmn}\rangle \quad = \langle \tilde{v}^{\blacktriangle,I}_{ijk}, \tilde{b}^{\blacktriangle,I}_{lmn}\rangle = c^{I}_{ijk}\delta_{il}\delta_{jm}\delta_{kn}. \tag{6.28}
$$

Due to the already shown biorthogonality we can assume that

$$
\begin{aligned}
b^{\blacktriangle,I}_{lmn} &= \alpha^{(1,I)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} + \alpha^{(2,I)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + \alpha^{(3,I)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\\
b^{\blacktriangle,II}_{lmn} &= \alpha^{(1,II)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} + \alpha^{(2,II)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + \alpha^{(3,II)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\\
b^{\blacktriangle,III}_{lmn} &= \alpha^{(1,III)}_{1}\tilde{b}^{\blacktriangle,I}_{lmn} + \alpha^{(2,III)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + \alpha^{(3,III)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}.
\end{aligned}
$$

This combined with (6.28) yields 9 equations. If we choose $\alpha^{\omega}_1 = \beta^{\omega}_1(c^{I}_{ijk})^{-1}$ and so on, we get the following system which needs to be solved

$$
\begin{pmatrix} \beta^I_1 & \beta^I_2 & \beta^I_3 \\ \beta^{II}_1 & \beta^{II}_2 & \beta^{II}_3 \\ \beta^{III}_1 & \beta^{III}_2 & \beta^{III}_3 \end{pmatrix}
\begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix} = I_3.
$$

The solution is given by

$$
\begin{pmatrix} \beta^I_1 & \beta^I_2 & \beta^I_3 \\ \beta^{II}_1 & \beta^{II}_2 & \beta^{II}_3 \\ \beta^{III}_1 & \beta^{III}_2 & \beta^{III}_3 \end{pmatrix}
= \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & 0 \\ \frac{1}{2} & 0 & -\frac{1}{2} \end{pmatrix}.
$$

The coefficients c^{ω}_{ijk} can be computed analogously to the triangular case be using the exact values of the integrals over Jacobi polynomials. Finally, one obtains

$$\langle \tilde{v}^I_{ijk}, \tilde{b}^I_{ijk} \rangle = \frac{2^7}{(2i-1)} \frac{1}{(2j+2i-1)(j+2i-1)(2k+2i+2j-1)(k+2i+2j-1)},$$
$$\langle \tilde{v}^{II}_{ijk}, \tilde{b}^{II}_{ijk} \rangle = \frac{2^6}{(2i-1)} \frac{1}{(2j+2i-1)(2k+2i+2j-1)(k+2i+2j-1)},$$
$$\langle \tilde{v}^{III}_{ijk}, \tilde{b}^{III}_{ijk} \rangle = \frac{2^6}{(2i-1)} \frac{1}{i\,(2i+2j-1)(j+2i-1)(2k+2i+2j-1)},$$

by using the orthogonality relations of the Jacobi polynomials. It remains to show, that to

$$v^{IV}_{1jk} = \begin{pmatrix} 1 \\ \frac{\eta}{2} \\ \frac{\eta}{4} \end{pmatrix} \left(\frac{1-\chi}{2}\right) \widehat{P}^1_j(\chi) \left(\frac{1-z}{2}\right)^{j+1} \widehat{P}^{2j+3}_k(z)$$

the dual shape functions are naturally orthogonal. For B_{ijk} orthogonality follows since the first component of v^{IV}_{1jk} is independent of $\eta = \frac{4x}{1-2y-z}$.
For C_{ijk} we apply the relations

$$\int_{-1}^{1} (i+1) P^{(1,1)}_{i-1}(x)\,\mathrm{d}x = 2\int_{-1}^{1} \frac{\mathrm{d}}{\mathrm{d}x} L_i(x)\,\mathrm{d}x = 2\,[L_i(x)]_{-1}^{1} = 2(1-(-1)^i)$$

and

$$\int_{-1}^{1} \frac{x}{2}(2i) P^{(1,1)}_{i-2}(x)\,\mathrm{d}x = 2\int_{-1}^{1} x \frac{\mathrm{d}}{\mathrm{d}x} L_{i-1}(x)\,\mathrm{d}x = 2\,[xL_i(x)]_{-1}^{1} - \underbrace{\int_{-1}^{1} L_{-1}(x)}_{=0}\,\mathrm{d}x = 2(1-(-1)^i)$$

to see that the scalar product $\langle v^{IV}_{1jk}, C_{lmn} \rangle = 0$, for all i, j, k, l, m, n. For D_{ijk} the same relation is applied, thus $\langle v^{IV}_{1jk}, D_{lmn} \rangle = 0$, for all i, j, k, l, m, n.
On the other hand, the dual function to v^{IV}_{1jk} is easily found to be

$$B_{1jk} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} P^{(2,1)}_{j-1}(y) \left(\frac{1-z}{2}\right)^{j-1} P^{(2j+2,1)}_{k-1}(z).$$

It is obviously orthogonal to ∇g_{ij} and ∇h_{ijk}, furthermore it is orthogonal to ∇f_i since it is independent of η. □

We conclude with some remarks.

Remark 10
It is usually possible to modify the index of the integrated Jacobi polynomials to modify the sparsity pattern and condition number of the element matrices. But in the context of polynomial dual functions the index $(2i)$ and $(2i+2j)$ are minimal, otherwise the dual functions will become rational with singularities for low polynomial degrees for the interior $H(\mathrm{curl})$ shape functions.

Remark 11
The expression of the coefficients $\alpha^{(1)}_{lmn}, \alpha^{(2)}_{lmn}$ and $\alpha^{(3)}_{lmn}$ can be significantly reduced by dividing each with

$$(2l{+}2m{-}1)(2l{-}1)(2n{-}2l{-}2m{-}1).$$

In this case one needs to compute the element matrix corresponding to biorthogonal system by numerical quadrature or similar methods.

As we have seen previously, we needed to modify part of our $H(\text{curl})$ conforming functions, such that they are in Nédélec's first space. This results are summarized in the following lemma:

Lemma 6.1.11
Let $\tilde{b}^{\blacktriangle,I}_{lmn}, \tilde{b}^{\blacktriangle,I}_{lmn}$ and $\tilde{b}^{\blacktriangle,I}_{lmn}$ be defined by table 6.1.3.
For $i \geq 2, j \geq 1, k \geq 1$ the functions $v^{\blacktriangle,I}_{ijk}, v^{\blacktriangle,II,\mathcal{N}}_{ijk}, v^{\blacktriangle,III,\mathcal{N}}_{ijk}, v^{\blacktriangle,IV}_{1jk}$ are biorthogonal to

$$\begin{aligned}
b^{\blacktriangle,I}_{lmn} &= 1/(l+m+n)\left(l\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} + m\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + n\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\right),\\
b^{\blacktriangle,II}_{lmn} &= -1/(mn)\left(l\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} - (l+n)\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + n\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\right),\\
b^{\blacktriangle,III}_{lmn} &= -1/(mn)\left(l\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} + m\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} - (l+m)\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\right),\\
b^{\blacktriangle,IV}_{1mn} &= \alpha^{(4)}_{lmn}\tilde{b}^{\blacktriangle,I}_{1mn},
\end{aligned}$$

where

$$\begin{aligned}
\alpha^{(1)}_{lmn} &= \frac{1}{2^7}(2l-1)(2m+2l-1)(m+2l-1)(2n+2l+2m-1)(n+2l+2m-1)\\
\alpha^{(2)}_{lmn} &= \frac{1}{2^6}(2l-1)(2m+2l-1)(2n+2l+2m-1)(n+2l+2m-1)\\
\alpha^{(3)}_{lmn} &= \frac{-1}{2^5}l(2l-1)(2m+2l-1)(m+2l-1)(2n+2l+2m-1)\\
\alpha^{(4)}_{lmn} &= \frac{1}{2^5}(2m+2)(m+2)(n+2m+2)(2n+2m+2)
\end{aligned}$$

Proof. Recall that the inner product between the auxiliary functions and biorthogonal auxiliary functions fulfil an orthogonality condition, e.g.

$$\begin{aligned}
\alpha^{(1)}_{ijk}\,\langle(\nabla f_i)g_{ij}h_{ijk}, \tilde{b}^{\blacktriangle,I}_{ijk}\rangle &= 1\\
\alpha^{(1)}_{ijk}\,\langle(\nabla f_i)g_{ij}h_{ijk}, \tilde{b}^{\blacktriangle,II}_{ijk}\rangle &= 0,
\end{aligned}$$

where $\alpha^{l}_{ijk}, l = 1,\ldots,4$ are the normalization constants.
Due to the weighted construction of $v^{\blacktriangle,II,\mathcal{N}}_{ijk}$ and $v^{\blacktriangle,III,\mathcal{N}}_{ijk}$, the biorthogonal functions will be weighted as well.
We apply the ansatz

$$b^{\blacktriangle,I}_{lmn} = \left(c_1\alpha^{(1)}_{lmn}\tilde{b}^{\blacktriangle,I}_{lmn} + c_2\alpha^{(2)}_{lmn}\tilde{b}^{\blacktriangle,II}_{lmn} + c_3\alpha^{(3)}_{lmn}\tilde{b}^{\blacktriangle,III}_{lmn}\right).$$

Taking the inner product between this ansatz and our functions $v^{\blacktriangle,I}_{ijk}, v^{\blacktriangle,II,\mathcal{N}}_{ijk}$, and $v^{\blacktriangle,III,\mathcal{N}}_{ijk}$ results

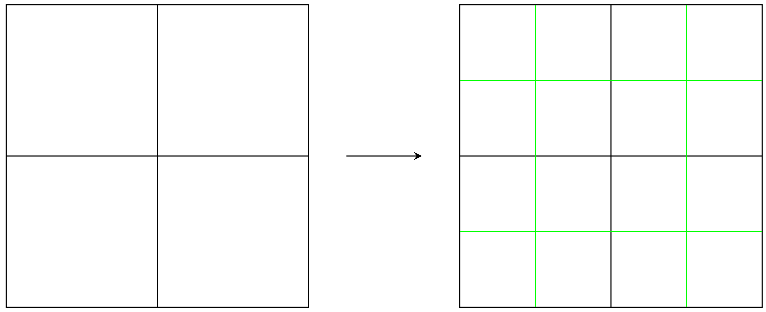


Figure 6.2.: Uniform refinement of quadrilateral grid

in the linear system

$$\begin{pmatrix} 1 & 1 & 1 \\ j & -i & 0 \\ k & 0 & -i \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$

The solution of this linear system is $\vec{c} = \frac{1}{i+j+k}(i,j,k)^\top$. The other biorthogonal functions follow analogously. □

6.2. Efficient computation of coefficients for hanging nodes

The results of this section have been published in [HPB22] and are extended shortly for the $3D$ case. We return to the discrete problem.

Problem 6.2.1
Let Ω be a polygonal Lipschitz domain, find $u_h \in \mathbb{V}_h(\Omega) \subset \mathbb{V}(\Omega)$ such that

$$a(u_h, v_h) = F(v_h) \text{ for all } v_h \in \mathbb{V}_h(\Omega). \tag{6.29}$$

The numerical solution can be obtained by a uniform refinement of all elements, see Figure 6.2 for an example of such a refinement on a quadrilateral grid.
But usually a uniform refinement is not optimal, since we generate many unnecessary degrees of freedom. A local mesh refinement can reduce this overhead. An application can have singularities due to non-smooth boundaries and/or coefficients. In such a case, the local mesh refinement can be based on a-priori information, see e.g. [KM03]. Alternatively, one can use a-posteriori information. See e.g. [CFPP14] or in the case of *hp*-refinement [MW01]. For simplicial mesh a local mesh refinement can be achieved consistently, by the so called red-green refinement or the newest vertex bisection, see [Bey98]. For quadrilateral or hexahedral triangulation, this usually not possible. In general, the application of a local refinement yields in vertices of a quadrilateral or a hexahedral, which are not a vertex of a neighbouring element. This is depicted in Figure 6.3. We restrict our self to the case where we have only one hanging node per edge, a so-called *hanging node of level one*. A generalization to hanging nodes of higher level, i.e. more hanging nodes per element interface, can be done, see e.g. [SDD10, KSA14, DSZK16, DPCF⁺20].
An efficient implementation for the case of $H(\text{curl})$ can be found in [KWB23].
After a local refinement, we have more degrees of freedom on the finer edge. Those are constraint by the values of the degrees of freedom on the coarser grid. For the ease of presentation, we

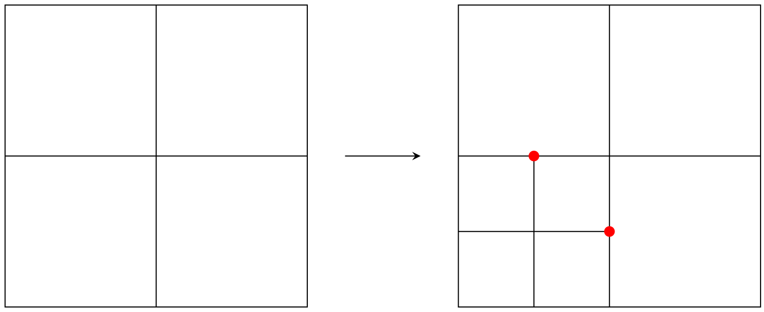


Figure 6.3.: Local refinement of a quadrilateral grid, the red dote denotes a hanging node

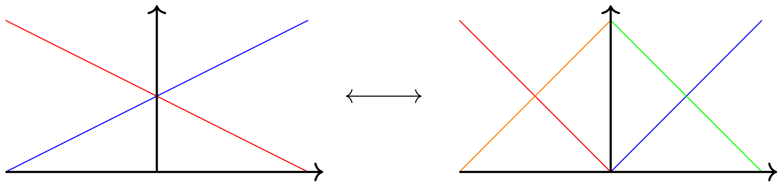


Figure 6.4.: Hat functions on $[-1,1]$ and $[-1,0] \cup [0,1]$

assume constant material function over the whole domain Ω and omit those. In the following we will present the setting only on the coarser interval $I = [-1,1]$ and the two finer intervals $I^l = [-1.0]$ and $I^r = [0,1]$. As example consider the Lagrangian polynomials of order $p = 1$,

$$
\begin{aligned}
\ell_0(x) &= \frac{1-x}{2}, & \ell_1 &= \frac{1+x}{2} \\
\ell_0^l(x) &= \ell_0(2x+1), & \ell_1^l(x) &= \ell_1(2x+1) \\
\ell_0^r(x) &= \ell_1(2x-1), & \ell_1^r(x) &= \ell_0(2x-1).
\end{aligned}
$$

Here $\ell_0(x)$ and $\ell_1(x)$ are given on I, while $I_{0/1}^l$ and $I_{0/1}^r$ are given on I^l or I^r. Note that the functions on I^r are reversed. In Figure 6.4 these relations are depicted. If we evaluate the functions at the midpoint, we can directly compute the coefficients for the linear functions, i.e.

$$
\begin{aligned}
\ell_0(x) &= \ell_0^l(x) + \frac{1}{2}\left(\ell_1^l(x) + \ell_1^r(x)\right), \\
\ell_1(x) &= \ell_0^r(x) + \frac{1}{2}\left(\ell_1^l(x) + \ell_1^r(x)\right).
\end{aligned}
$$

We can represent this as a matrix relation by

$$
B = \begin{pmatrix} 1 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 1 & \frac{1}{2} \end{pmatrix} \leftrightarrow \begin{pmatrix} \ell_0(x) \\ \ell_1(x) \end{pmatrix} = B \begin{pmatrix} \ell_0^l(x) \\ \ell_1^l(x) \\ \ell_0^r(x) \\ \ell_1^r(x) \end{pmatrix}
$$

For orders $p > 1$ we introduced basis functions based on integrated Legendre polynomials in section 4.2.1. In this case, we would need to evaluate the integrated Legendre polynomial on more points in our interval. This is a costly procedure, and does not take into account the orthogonality relations of the underlying Legendre polynomials. We thus introduce a new ansatz to determine B in a more efficient way.

In a first step, we consider $\widehat{L}_0(x) = -1$ and $\widehat{L}_1(x) = x$, instead of the usual hat functions. Then the three term recursion

$$i\,\widehat{L}_i(x) = (2i-3)x\widehat{L}_{i-1}(x) - (i-3)\widehat{L}_{i-2}(x) \text{ for } i \geq 2, \tag{6.30}$$

holds. Denote by $\widehat{L}_i^l(x)$ and $\widehat{L}_i^r(x)$ the i-th integrated Legendre polynomial on I^l and I^r, where

$$\widehat{L}_i^l(x) = \begin{cases} \widehat{L}_i(2x+1) & x \in [-1,0] \\ 0 & \text{else} \end{cases}, \text{ and } \widehat{L}_i^r(x) = \begin{cases} \widehat{L}_i(2x-1) & x \in [0,1] \\ 0 & \text{else} \end{cases} \text{ for } i \geq 2.$$

Formally written out, we search for the matrix $B = (B^l, B^r)$, such that

$$\begin{pmatrix} \widehat{L}_0(x) \\ \vdots \\ \widehat{L}_n(x) \end{pmatrix} = \begin{pmatrix} B^l & B^r \end{pmatrix} \begin{pmatrix} \widehat{L}_0^l(x) \\ \vdots \\ \widehat{L}_n^l(x) \\ \widehat{L}_0^r(x) \\ \vdots \\ \widehat{L}_n^r(x) \end{pmatrix}$$

For $i \geq 2$ there is no contribution from $\widehat{L}_0^k, k = l, r$, since $\widehat{L}_i(\pm 1) = 0$. By the three term recursion (6.30) we use $L_0^k(x) = -1$ and $L_1^k(x) = 2x \pm 1, k = l, r$. The linear hat functions can then be easily computed. For this choice we can compute the coefficients $B_{i,j}^k$ for

$$\widehat{L}_i(x) = \sum_j^i B_{i,j}^l \widehat{L}_j^l(x) + \sum_j^i B_{i,j}^r \widehat{L}_j^r(x), \tag{6.31}$$

by multiplication with a test function v_j and integration over either I^l or I^r. This method is called *constrained approximation* and was introduced in [DORH89], see also [vSD04]. Since

$$\widehat{L}_i(x) = \frac{1-x^2}{2(i-1)} P_{i-2}^{(1,1)}(x),$$

it is efficient to choose

$$\begin{aligned} v_j^l(x) &= P_{j-2}^{(1,1)}(2x+1), \\ v_j^r(x) &= P_{j-2}^{(1,1)}(2x-1), \end{aligned} \tag{6.32}$$

as our test functions.
This is the biorthogonal choice which we have seen in section 6.1. Thus, it follows

$$\begin{aligned} \int_{I^l} \widehat{L}_i(x) v_n^l(x)\,\mathrm{d}x &= \int_{I^l} \sum_j^i B_{i,j}^l \widehat{L}_j^l(x) v_j^l(x)\,\mathrm{d}x \\ &= \sum_j^i \int_{I^l} B_{i,j}^l \widehat{L}_j^l(x) v_n^l(x)\,\mathrm{d}x \\ &= B_{i,n}^l \int_{I^l} \widehat{L}_n^l(x) v_n^l(x)\,\mathrm{d}x, \end{aligned}$$

where we used that $\widehat{L}_j^r(x)$ has no support on I^l. The coefficients $B_{i,n}^l$ follow now by solving the integrals. Alternatively, we can collect those entries from a coefficient comparison. From this data, we can compute linear recursions for $B^l_{i,j}$ (and $B^r_{i,j}$), by automated guessing, as we have seen in section 2.1.2. Again, we apply the `Mathematica` software `Guess` [Kau09]. For $i, j \geq 0$ the following relation

$$2j(2j+1)(j-i-1)B^l_{i+1,j} + i(2j-1)(i-j-1)B^l_{i,j+1} + (2j+1)(2j-i)(i+j-1)B^l_{i,j} = 0, \qquad (6.33)$$

is obtained. Note that the leading coefficient does not have a pole in the range of the indices. Furthermore, $B^l_{i,j} = 0$ if $j > i$. A second recursion reads

$$2B^l_{i+1,j+1} = B^l_{i+1,j+2} - 2B^l_{i+2,j+1} + B^l_{i+1,j} - 2B^l_{i,j+1} \qquad (6.34)$$

which has the advantage of constant coefficients. Lastly we mention, that, for any fixed j, we have the relation

$$(i-j)(i+1)(i+j-1)B^l_{i,j} + (j-1)j(2i+1)B^l_{i+1,j} + (i-j+2)i(i+j+1)B^l_{i+2,j} = 0.$$

For the multivariate guessing we again used the command `GuessMultRE` from the package `Guess`, which allows specifying different structure sets and polynomial degrees for the recurrence, see section 2.1.2. For the univariate recurrence, we employed the `GuessMinRE` command, which not only guesses the recurrence, but also guesses what "minimal" could be for the given data.

Example 6.2.2
Below, we show a short coding example, where we applied a coefficient comparison.

```
In[3]:= L[0] := −1; L[1] := x; L[n_Integer] := L[n] = Factor[((2n − 3)xL[n − 1] − (n − 3)L[n − 2])/n]];
In[4]:= L1[0] := −1; L1[1] := 2x + 1;
        L1[n_Integer] := L1[n] = Factor[((2n − 3)(2x + 1)L1[n − 1] − (n − 3)L1[n − 2])/n]];
In[5]:= sys = Table[Thread[CoefficientList[L[i] − Sum[B[i, j]L1[j], {j, 0, i}], x] == 0], i, 0, 30];
In[6]:= sol = Flatten[Solve[sys]];
In[7]:= data = Table[B[i, j], {i, 0, 30}, {j, 0, 30}]/.sol/.{B[i_, j_] → 0};
In[8]:= GuessMultRE[data, Flatten[Table[B[i + ii, j + jj], {ii, 0, 2}, {jj, 0, 2}]], {i, j}, 0]

Out[8]= {B[i, 1 + j] − 1/2B[1 + i, j] + B[1 + i, 1 + j] − 1/2B[1 + i, 2 + j] + B[2 + i, 1 + j]}

In[9]:= GuessMultRE[data, Flatten[Table[B[i + ii, j + jj], {ii, 0, 1}, {jj, 0, 1}]], {i, j}, 2]

Out[9]= {−(1/2)(−1 + i + j)(1 + 2j)B[i, j] + 1/2(−1 + i − j)(−1 + 2j)B[i, 1 + j] − 1/2(1 + i − j)(1 + 2j)B[1 + i, j] +
        1/2(1 + i + j)(−1 + 2j)B[1 + i, 1 + j]}
```

For the application of the integral relations, we can resolve this directly, by using algorithms for deriving and proving identities among holonomic functions. By application of the holonomic system approach [Zei90b, Chy98, Kou10], the guessed relations, can be proven.

Example 6.2.3
We can compute

$$B^l_{i,j} = \int_{I^l} \frac{1-x^2}{2(i-1)} P^{(1,1)}_{i-2}(x) P^{(1,1)}_{i-2}(2x+1)\,\mathrm{d}x \Big/ \int_{I^l} \frac{1-x^2}{2(i-1)} P^{(1,1)}_{i-2}(2x+1) P^{(1,1)}_{i-2}(2x+1)\,\mathrm{d}x$$

exactly by standard `Mathematica` commands. Then the input for the package `HolonomicFuntions`, looks as follows:

```
In[10]:= ann = Annihilator[Integrate[JacobiP[i − 2, 1, 1, x]JacobiP[j − 2, 1, 1, 2x + 1], {x, −1, 0}]/
         Integrate[JacobiP[i − 2, 1, 1, 2x + 1]JacobiP[j − 2, 1, 1, 2x + 1], {x, −1, 0}],
         {S[i], S[j]}, Inhomogeneous → True]
```

Out[10]= $\{\{2(1+i-j)j(1+2j)S_i + i(-1+i-j)(-1+2j)S_j - (i-2j)(-1+i+j)(1+2j),$
$(-2+i-j)j(1+i+j)(-1+2j)S_j^2 - (i-i^2+3j(1+j))(-3+4j(1+j)S_j + (i-j)(1+j)(-1+i+j)(3+2j))\},$
$\ldots$somethingmessy$\ldots\}$

Here, S_n (or $S[n]$ in the input) denotes the forward shift in n. The messy part contains the Legendre polynomials evaluated at $x = \pm 1$. `Mathematica` fully reduces those to zero with its `FullSimplify` command under the assumption that i, j are integer. The first part of the results ann, the annihilator, is just recurrence (6.33). The second operator in the annihilator corresponds to the univariate recurrence. Larger order recurrences such as (6.34), can be verified automatically using the `OreReduce` command. Next, we show the numerical algorithms of these results. In order to compute the connection coefficients efficiently from any of these relations, we need a sufficient set of initial values. Firstly, recall that naturally $B^{(k)}_{i,j} = 0$ if $j > i$. For the upper left block, we have

$$B^l_{0,0} = 1, \quad B^l_{1,0} = B^l_{1,1} = \frac{1}{2},$$

and for the first column

$$B^l_{2i,0} = \frac{(-1)^i}{2} \frac{\left(-\frac{1}{2}\right)_i}{i!}, \quad \text{and} \quad B^l_{2i+1,0} = 0 \quad \text{for } i \geq 1$$

where $(a)_n = a(a+1)\cdots(a+n-1)$ denotes the Pochhammer symbol (or rising factorial). Note that we have the rather obvious relation

$$B^l_{2i,0} = -\frac{2i+1}{2i+2} B^l_{2(i-1),0}.$$

Furthermore, we have for the diagonal that $B^l_{i,i} = 2^{-i}$. Finally, even and odd (integrated) Legendre polynomials are symmetric and antisymmetric, respectively. This gives $B^r_{i,j} = (-1)^{i+j} B^l_{i,j}$. Summarizing, the matrix entries can be computed by algorithm 6.
In context of FEM, the basis functions $\ell_0(x) = \frac{1+x}{2}$ and $\ell_1(x) = \frac{1-x}{2}$ are preferred instead of $\widehat{L}_0(x), \widehat{L}_1(x)$. Then, some of the steps have to be modified. Finally, one obtains the algorithm 7 with the recursion formula (6.34).

Algorithm 6 Computation of the entries $B_{i,j}^{(k)}$

Initialize $B = 0$,
Set $B_{0,0}^l = 1, B_{1,0}^l = B_{1,1}^l = \frac{1}{2}, B_{2,0}^l = B_{2,0}^r = \frac{1}{4}$,
for $i = 2$ **to** $\lfloor p/2 \rfloor$ **do**
  Set $B_{2i,0}^l = -\frac{2i+1}{2i+2} B_{2(i-1),0}^l$ **and** $B_{2i,0}^r = B_{2i,0}^l$
end for
for $j = 1$ **to** p **do**
  for $i = j$ **to** p **do**
    Compute $B_{i,j}^l$ by (6.33)
    Set $B_{i,j}^r = (-1)^{i+j} B_{i,j}^l$
  end for
end for

Algorithm 7 Computation of the entries $\tilde{B}_{i,j}^{(k)}$

Initialize $\tilde{B} = 0$,
Set $\tilde{B}_{0,0}^l = 1, \tilde{B}_{1,0}^l = \tilde{B}_{1,1}^l = \frac{1}{2}, \tilde{B}_{2,0}^l = \tilde{B}_{2,0}^r = \frac{1}{4}, \tilde{B}_{2,2}^l = \tilde{B}_{2,2}^r = \frac{3}{4}, \tilde{B}_{3,2}^l = \frac{3}{8}, \tilde{B}_{3,2}^r = -\frac{3}{8}$,
for $i = 2$ **to** $\lfloor p/2 \rfloor$ **do**
  Set $\tilde{B}_{2i,1}^l = -\frac{2i+1}{2i+2} \tilde{B}_{2(i-1),0}^l$ **and** $\tilde{B}_{2i,1}^r = \tilde{B}_{2i,1}^l$
  $\tilde{B}_{(2i,2)}^l = -\frac{2i+3}{2i+2} \tilde{B}_{2(i-1),2}^l$ **and** $\tilde{B}_{2i,2}^r = \tilde{B}_{2i,2}^l$
  if $2i + 1 \leq p$ **then**
    $\tilde{B}_{(2i+1,2)}^l = -\frac{2i+3}{2i+4} \tilde{B}_{2i,2}^l$ **and** $\tilde{B}_{2i+1,2}^r = -\tilde{B}_{2i+1,2}^l$
  end if
end for
for $i = 2$ **to** p **do**
  for $j = 1$ **to** $i - 1$ **do**
    Compute $\tilde{B}_{i,j}^l$ by (6.34)
    Set $\tilde{B}_{i,j}^r = (-1)^{i+j} \tilde{B}_{i,j}^l$
  end for
end for

Extension to $H(\text{curl})$

We consider the higher order edge functions of the space $H(\text{curl})$ in $2D$. For example, on the quadrilateral $\square = [-1,1]^2$, the right and the left edge functions are given by

$$u_{1j}^{\square,E_1}(x,y) = \nabla\left(\frac{1+x}{2}\widehat{L}_j(y)\right) = \begin{pmatrix} \frac{1}{2}\widehat{L}_j(y) \\ \frac{1+x}{2} L_{j-1}(y) \end{pmatrix}, \tag{6.35}$$

$$u_{1j}^{\square,E_3}(x,y) = \nabla\left(\frac{1-x}{2}\widehat{L}_j(y)\right) = \begin{pmatrix} \frac{1}{2}\widehat{L}_j(y) \\ \frac{1-x}{2} L_{j-1}(y) \end{pmatrix}. \tag{6.36}$$

In Figure 6.5 we depicted the case, where we have an unrefined element $Q^l = [-1,1]^2$ and two refined elements $Q^t = [1,2] \times [0,1]$, $Q^b = [1,2] \times [-1,0]$. On Q^l our right edge functions are given by $u_{1j}^{\square,E_1}(x,y)$ as in (6.35), while the left edge functions on Q^t and Q^b are given by

$$u_{1j}^{Q^t,E_3}(x,y) := u_{1j}^{\square,E_3}(2x-3, 2y-1), \quad u_{1j}^{Q^b,E_3}(x,y) := u_{1j}^{\square,E_3}(2x-3, 2y+1). \tag{6.37}$$

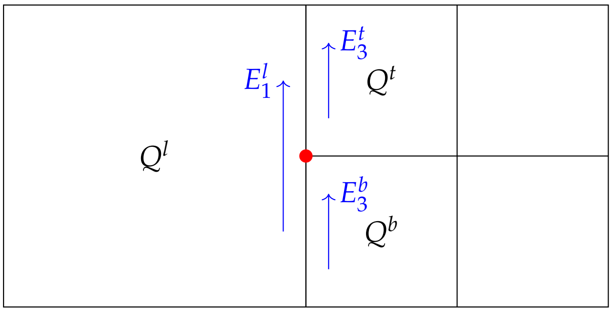


Figure 6.5.: Hanging node (red) in $2D$ with orientation of edge functions (blue)

From the construction of our basis function in chapter 4, we recall that we only need continuity in the tangential trace, depicted in blue in Figure 6.5. At $x = 1$ this results in the conditions

$$\begin{aligned} \operatorname{tr} u_{1,j}^{Q_l,E_1}(x,y) \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} &= L_{j-1}(y) \\ \operatorname{tr} u_{1,j}^{Q_t,E_1}(x,y) \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} &= L_{j-1}(2y+1) \\ \operatorname{tr} u_{1,j}^{Q_b,E_1}(x,y) \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} &= L_{j-1}(2y-1) \end{aligned} \tag{6.38}$$

For $j \geq 2$ we thus need the constraint matrix $B = (B^b, B^t)$ such that

$$L_i(y) = \sum_j^i B_{ij}^b L_{j-1}(2y-1) + \sum_j^i B_{ij}^t L_{j-1}(2y+1). \tag{6.39}$$

If we take the derivative of (6.31), we see that we have the same constraint matrix as before (except for a constant factor of 2).

Continuous constrained approximation in $3D$

In $3D$ we need to differ between hanging nodes and hanging edges. The former can be handled directly as before. For the hanging edge problem, we need to handle the continuity over the interfaces, i.e. edges and faces.

As e.g. discussed in [vSD04] there are different settings which may appear. In [vSD04] the following four cases were declared as essential:

- Two quadrilateral faces constrained by one quadrilateral face,
- four quadrilateral faces constrained by one quadrilateral face,
- four triangular faces constrained by one triangular face,
- two triangular faces constrained by one quadrilateral face.

The last situation is needed, if one wants to construct a joint mesh between hexahedrons and tetrahedrons without pyramidal or prismatic elements.

In the following, we simplify the problem by assuming that no orientation problems are present at the interface, see e.g. [KWB23, vSD04]. Furthermore, we omit the transformation from an arbitrary element to the reference elements. We will discuss only the first and the fourth case.

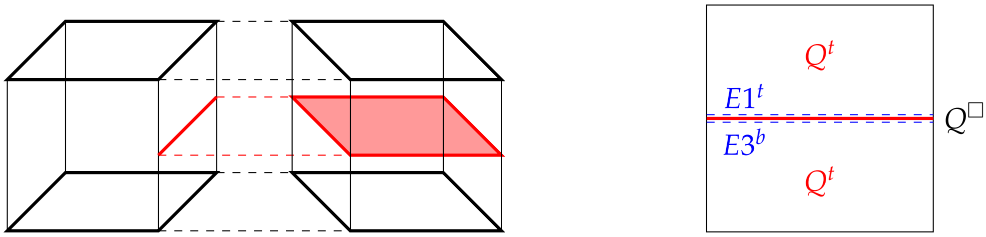


Figure 6.6.: Hanging edge in $3D$ and reduction to the $2D$ trace

The other two cases can be handled analogously, although the hanging node in case two needs to be handled with the techniques from the previous sections.

Firstly, we need to recall the dimensional hierarchy as mentioned in chapter 4. If we take the trace of our face functions, we get the $2D$ edge functions. Depicted in Figure 6.6 we have on the left side the case of a hanging edge in the three-dimensional setting, and on the right side, we see the reduction to the traces. We denoted the relevant edges in blue.

Without loss of generality assume $z = 1$, i.e. we are interested in the face $\square = [-1,1] \times [-1,1] \times \{1\}$. The trace on this face function is thus just $u_{ij}^{\square}(x,y)$.

Furthermore, let $B^E[i,j,m,l]$ be the array of constraints with respect to the edge functions, $B^t[i,j,m,l]$ and $B^b[i,j,m,l]$ the arrays with respect to the top or bottom face functions, respectively.

Our problem then reads:

Find $B = (B^E, B^b, B^t)$, such that

$$
\begin{aligned}
u_{ij}^{\square}(x,y) = &\sum_{k=2}^{i} B^E[i,j,k,1]u_{k1}^{\square,E}(x,y) + \sum_{k=2}^{i}\sum_{l=2}^{j} B^b[i,j,k,l]u_{kl}^{\square}(x,2y+1) \\
&+ \sum_{k=2}^{i}\sum_{l=2}^{j} B^t[i,j,k,1]u_{kl}^{\square}(x,2y-1),
\end{aligned}
\tag{6.40}
$$

where

$$
u_{k1}^{\square,E}(x,y) = \begin{cases} u^{\square,E_1^t}(x,y) & (x,y) \in Q^t \\ u^{\square,E_3^b}(x,y) & (x,y) \in Q^b \end{cases}.
$$

The first part includes the influence of the edge functions, while the other two include the influence of the face functions. The edge part is of utmost importance, since all face functions of Q^t and Q^b vanish on the edge. Thus, only the edge functions can approximate values on the edge E. Moreover there is no influence of the vertex functions, since they would take values at the boundary of $Q^{\square}$ and thus does not vanish on the boundary.

We solve problem (6.40) in two steps, first solve for the constraint coefficients w.r.t. the edge E. Then solve for the constraints of the face functions.

As seen before in the approximation case, first multiply with a test function $v \in L^2(\square)$, then integrate over the edge E. The problem reduces to

$$
\int_E u_{ij}^{\square}(x,y)v(x,y)\,\mathrm{d}x\,\mathrm{d}y = \int_E \sum_{k=2}^{i} B^E[i,j,k,1]u_{k1}^{\square,E}(x,y)v(x,y)\,\mathrm{d}x\,\mathrm{d}y
$$

Since the edge functions are continuous, the trace of the edge functions are equal on the edge. So, without loss of generality, we can replace $u^{\square,E}(x,y)$ by $u^{\square,E_1^t}(x,y)$ in the edge integral. Furthermore, by dimensional hierarchy the trace of the edge functions is given by $\widehat{L}_i(x)$. Together with the biorthogonal choice $P_{i-2}^{(1,1)}(x)$, we get the relation

$$\int_{E1^t} u_{ij}^{\square}(x,y)P_{k-2}^{(1,1)}(x)\,\mathrm{d}x\,\mathrm{d}y = c_{ik}B^E[i,j,k,1]\delta ik.$$

Now to the whole approximation problem. We rewrite (6.40) as follows:

$$\begin{aligned} u_{ij}^{\square}(x,y) - \sum_{k=2} B^E[i,j,k,1]u_{k1}^{\square,E}(x,y) \\ = \sum_{k=2}^{i}\sum_{l=2}^{j} B^b[i,j,k,l]u_{kl}^{\square}(x,2y+1) + \sum_{k=2}^{i}\sum_{l=2}^{j} B^t[i,j,k,l]u_{kl}^{\square}(x,2y-1). \end{aligned}$$

We can then write the equation out as

$$\begin{aligned} \widehat{L}_i(x)\widehat{L}_j(y) - \sum_{k=2} B^E[i,j,k,1]\widehat{L}_k(x)\left(\mathbb{1}_{[-1,1]\times[0,1]}(1-y) + \mathbb{1}_{[-1,1]\times[-1,0]}(1+y)\right) \\ = \sum_{k=2}^{i}\sum_{l=2}^{j} B^b[i,j,k,l]\widehat{L}_k(x)\widehat{L}_l(2y+1) + \sum_{k=2}^{i}\sum_{l=2}^{j} B^t[i,j,k,l]\widehat{L}_k(x)\widehat{L}_l(2y-1). \end{aligned}$$

By multiplication with the biorthogonal function $P_{k-2}^{(1,1)}(x)$ and integration over $x \in [-1,1]$, we see that the relation reduces to

$$\widehat{L}_j(y) - \phi(y) = \sum_{l=0}^{j} B^b[i,j,i,l]\widehat{L}_j(2y+1) + \sum_{l=0}^{j} B^t[i,j,i,l]\widehat{L}_l(2y-1),$$

where

$$\phi(y) \coloneqq \sum_{k=2} B^E[i,j,k,l]\left(\mathbb{1}_{[-1,1]\times[0,1]}(1-y) + \mathbb{1}_{[-1,1]\times[-1,0]}(1+y)\right).$$

The coefficients

$$\int_{-1}^{1} \widehat{L}_j(y)P_{k-2}^{(1,1)}(2y-1)\,\mathrm{d}y$$

are the same as before, and can thus be computed as before. For the edge part holds

$$\int_{-1}^{1} \phi(y)P_{k-2}^{(1,1)}(2y-1)\,\mathrm{d}y = \int_{0}^{1} (1-y)P_{k-2}^{(1,1)}(2y-1)\,\mathrm{d}y,$$

which can be efficiently computed by the Mellin transform, see lemma A.3.2.

Case of two triangular faces constrained by a quadrilateral

In the last case, we heavily applied simplifications due to the tensorial structure. Consider here the reference domain $\widehat{\square} = [0,1]^2$, due to a simpler notation. Furthermore, let

$$\begin{aligned} \triangle_b &\coloneqq \{(x,y) \in \mathbb{R} | 0 \le y \le 1 \text{ and } 0 \le x \le 1-y\}, \\ \triangle_t &\coloneqq \{(x,y) \in \mathbb{R} | 0 \le y \le 1 \text{ and } 1-y \le x \le 1\}. \end{aligned}$$

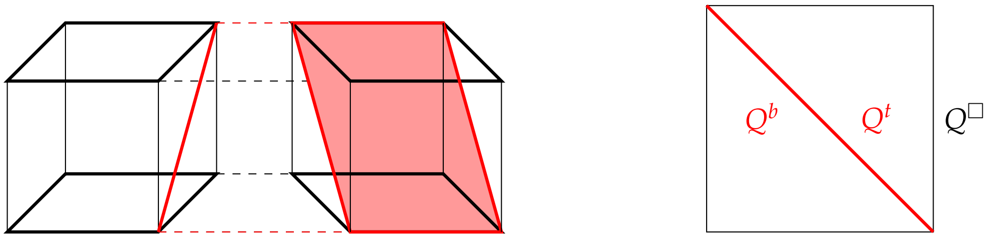


Figure 6.7.: Hanging edge for a non-conforming mesh and the reduction to the 2D trace

Denote by $u_{ij}^{\widehat{\square}}$ the face functions on $\widehat{\square}$ and $u_{ij}^{\triangle_b}, u_{ij}^{\triangle_t}$ the face functions on $\triangle_b$ or $\triangle_r$, respectively. We now search for $B = (B^E, B^b, B^t)$, s.t.

$$u_{ij}^{\widehat{\square}}(x,y) = \sum_{l=2}^{i} B^E[i,j,1,l] u_l^{\triangle,E}(x,y) + \sum_{k=2}^{i}\sum_{l=1}^{j} B^b[i,j,k,l] u_{kl}^{\triangle_b}(x,y) + \sum_{k=2}^{i}\sum_{l=1}^{j} B^t[i,j,k,l] u_{kl}^{\triangle_t}(x,y). \tag{6.41}$$

The coefficients B^E can again be determined by the integration of the edge biorthogonal function over the edge E. For $k \geq 2, l \geq 1$ we can determine $B^b[i,j,k,l]$ by multiplication with the biorthogonal function (6.5) (on $\triangle_b$) and integration over $\triangle_b$. Then with (6.41), we need to compute

$$\begin{aligned} G^{(1)}_{i,j,k,l} - G^{(2)}_{i,j,k,l} \\ = B^b[i,j,k,l] \int_{\triangle_b} \widehat{L}_k\left(\frac{2x}{1-y} - 1\right) P^{(1,1)}_{k-2}\left(\frac{2x}{1-y} - 1\right) (1-y)^{2i-1} \widehat{P}_l^{2i}(2y-1) P^{(2i-1,1)}_{l-1}(2y-1), \end{aligned}$$

where

$$\begin{aligned} G^{(1)}_{i,j,k,l} &:= \int_{\triangle_b} \widehat{L}_i(2x-1)\widehat{L}_j(2y-1) P^{(1,1)}_{k-2}\left(\frac{2x}{1-y} - 1\right)(1-y)^{k-2} P^{(2k-1,1)}_{l-1}(2y-1)\, \mathrm{d}x\, \mathrm{d}y \\ G^{(2)}_{i,j,k,l} &:= \sum_{l=2} \int_{\triangle_b} B^E[i,j,1,l] u_l^{E2}(x,y) P^{(1,1)}_{k-2}\left(\frac{2x}{1-y} - 1\right)(1-y)^{k-2} P^{(2k-1,1)}_{l-1}(2y-1)\, \mathrm{d}x\, \mathrm{d}y \end{aligned}$$

The integral on the right-hand side can be directly computed and normalized. The left-hand side needs to be computed, but this can be done efficiently by recursive relation. If we plug the exact integral $G^{(1)}_{i,j,k,l}$ on the left-hand side into `Guess`, we get the following recursive relation,

$$\begin{aligned} 0 = &(-1+i+j+k+l)(3+2k+2l)G^{(1)}_{i,1+k,j,l} + (-3+i+j-k-l)(1+2k+2l)G^{(1)}_{i,1+k,j,1+l} \\ &+ (-1+i-j+k+l)(3+2k+2l)G^{(1)}_{i,1+k,1+j,l} + (-3+i-j-k-l)(1+2k+2l)G^{(1)}_{i,1+k,1+j,1+l} \\ &+ (1+i-j-k-l)(3+2k+2l)G^{(1)}_{1+i,1+k,j,l} + (3+i-j+k+l)(1+2k+2l)G^{(1)}_{1+i,1+k,j,1+l} \\ &+ (1+i+j-k-l)(3+2k+2l)G^{(1)}_{1+i,1+k,1+j,l} + (3+i+j+k+l)(1+2k+2l)G^{(1)}_{1+i,1+k,1+j,1+l}. \end{aligned}$$

For the starting values, one could apply the results from appendix A.3.2. Also, the integral $G^{(2)}_{i,j,k,l}$ can be computed by the Mellin transformation lemma A.3.2.
Due to symmetry, the same relations also hold for the second set of coefficients. Thus, we are able to *glue* two triangles for arbitrary polynomial order. Further extension to the mentioned two other possible face configuration are straight forward. Furthermore, an extension to $H(\mathrm{div})$ and $H(\mathrm{curl})$ should be possible, but we postpone this discussion to future work.

7. Conclusion

7.1. Discussion

In this thesis we have seen that the structure of special functions, especially of orthogonal polynomials, can be used to optimize multiple problems in high order finite element methods (or spectral methods). We have shown that all interior entries of mass and stiffness matrix can be described by a Kampé de Fériet series. This series has contiguous relations which can be detected by `Guess` and implemented in an efficient recursive algorithm to set up the local matrices, which also applies for edge and face functions. Moreover, this structure also helps in the case of constraint matrices for hanging nodes (or edges). This efficiency of the recursion algorithm has been shown in two very simple numerical experiments, which were enough to show the advantage of our method. Even in $1D$ we drastically reduce the local assembly time. But we have to take this with a grain of salt. At the moment this method only applies for elementwise constant material functions on a polygonal domain and an extension is not straight forward.
Another connected topic which we discussed were the derivation of biorthogonal polynomials by application of connections between integrated Jacobi polynomials and Jacobi polynomials. We were able to derive biorthogonal functions for the interior part of our element matrices, even for vectorial functions in $H(\mathrm{curl})$. An important fact was, that integrated Jacobi polynomials can be viewed as weighted Jacobi polynomials and as such we can use the orthogonality relations. We gave a brief extension to edge and face functions by explicit application of the dimensional hierarchy. A purely biorthogonal basis for all functions including vertex, edge, and face functions is not possible, since these functions contain lower order parts, which are neither an orthogonal polynomial nor do they match an appropriate weight index.
In the last part, we gave an example where all our introduced methods were applied at the same time. For constraint matrices in a non-conforming mesh, we applied first the biorthogonal functions to derive the exact formulation of those matrices. Then by application of the symbolic software `Guess` and some smaller results on Jacobi polynomials, we were able to state a recursive method in optimal complexity.

7.2. Outlook

We end this thesis with possible future works:

Special Functions: First not published tests have shown that it may be possible to find similar recursive relations for other classes of orthogonal polynomials. Since Jacobi polynomials are only a small part of the Askey table, see e.g. [KLS10], it should be possible to find similar relations for other discrete or continuous polynomials. An example would be the discrete or continuous Hahn polynomials. From a numerical point of view, such a relation for the Hahn polynomials could be interesting in the case of visualization. Numerical solutions are usually plotted or visualized by spline functions, which can be represented by Bernstein polynomials. It has been shown, e.g. in [LW06], that two-variate Jacobi and Bernstein polynomials can be represented by each other with the help of the Hahn polynomials.

Basis functions: We have seen that we lose sparsity in the case of curved domains or non-constant material functions. A possible way, to circumvent this, would be to derive new orthogonal polynomials. In this case, we include the isoparametric map or the material function into the weight functions of the orthogonal polynomials. A similar approach was done in [SO20] for a spectral element method. Furthermore, one could try to compute new multivariate Sobolev orthogonal polynomials, see [MX15], which are orthogonal with respect to the bilinear form.

Recursion relations: Recursive relations of the integral over three Jacobi polynomials could be applied in a nonlinear setting like the Navier-Stokes equation. This could optimize some high order linearization techniques, see e.g. [GR86, GR94, Bra13].

Biorthogonal Functions: Many classical biorthogonal techniques, like error estimates and preconditioners, from the theory of biorthogonal wavelets could be extended to the case of our FEM basis functions. See e.g. [BSS04] for reference.

Constraint matrices: We have seen that we can glue elements together. The respective constraint matrices can be computed in optimal complexity. Some open topics like constrained faces for vectorial functions, e.g. in $H(\mathrm{curl})$ are still open.

A. Appendix

A.1. Derivation of mixed recurrence relation

Generalization

The generalization of (2.19) is the following the Kampé de Fériet series where x and y are not equal one anymore, i.e.

$$F := F^{1;2;2}_{1;1;1}\left(\begin{matrix} \mu+1 & ; & -n \quad n+\alpha+\beta+1 & ; & -m \quad m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+1 & ; & \rho+1 \end{matrix} ; x ; y\right). \tag{A.1}$$

Furthermore we omit the indices of F to keep the notation as simple as possible.

Recurrence formula

One can extend the proofs for the contiguous relations of a hypergeometric series to the general case of the Kampé de Fériet series using the techniques in Rainville [Rai71, Ch.4]. Let Z be a generalization of F to arbitrary coefficients, i.e.

$$Z = \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \underbrace{\frac{(a)_n(b)_n(f)_m(g)_m(d)_{n+m}x^n y^m}{n!\,m!\,(c)_n(h)_m(e)_{n+m}}}_{=:\tau_n\tau_m\tau_{n+m}}$$

also a general Kampé de Fériet series. It has similar contiguous functions to the six contiguous functions of the Gaussian hypergeometric series, namely

$$\begin{aligned} Z(a+) &= \sum_{n=0}^{\infty}\frac{a+n}{a}\tau_n\tau_m\tau_{n+m}, & Z(a-) &= \sum_{n=0}^{\infty}\frac{a-1}{a-1+n}\tau_n\tau_m\tau_{n+m}, \\ Z(b+) &= \sum_{n=0}^{\infty}\frac{b+n}{b}\tau_n\tau_m\tau_{n+m}, & Z(b-) &= \sum_{n=0}^{\infty}\frac{b-1}{b-1+n}\tau_n\tau_m\tau_{n+m}, \\ Z(c+) &= \sum_{n=0}^{\infty}\frac{c}{c+n}\tau_n\tau_m\tau_{n+m}, & Z(c-) &= \sum_{n=0}^{\infty}\frac{c-1+n}{c-1}\tau_n\tau_m\tau_{n+m}, \end{aligned}$$

obviously one can find 6 similar functions for f, g and h. In the following, omit $(f)_m, (g)_m$ and $(h)_m$ to simplify the notation.
Use the differential operator $\theta_x = x\frac{\partial}{\partial x}$. This leads to

$$(\theta_x + a)Z = \sum_{n=0}^{\infty}(a+n)\tau_n\tau_m\tau_{n+m}$$

and hence

$$\begin{aligned}(\theta_x + a)Z &= aZ(a+),\\ (\theta_x + c - 1)Z &= (c-1)Z(c-),\end{aligned} \tag{A.2}$$

and so on. Following one of the calculations in [Rai71], one can derive

$$\begin{aligned}\theta_x Z(a-) &= \sum_{n=1}^{\infty}\sum_{m=0}^{\infty} \frac{(a-1)_n (b)_n (d)_{n+m} \tau_m x^n y^m}{(c)_n (n-1)!\,(e)_{n+m}}\\ &= x\sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \frac{(a-1)_{n+1} (b)_{n+1} (d)_{n+m+1} \tau_m x^n y^m}{(c)_{n+1} n!\,(e)_{n+m+1}}\\ &= x(a-1)\frac{d}{e}\sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \frac{(b+n)(d+n+m)}{(c+n)(e+n+m)}\tau_{n+m}\tau_n\tau_m\\ &= x(a-1)\frac{d}{e}Z(d+,e+) - \frac{(a-1)(c-b)d}{ce}xZ(c+,d+,e+),\end{aligned}$$

since $\frac{b+n}{c+n} = 1 - \frac{c-b}{c+n}\frac{c}{c}$. Now replace $\theta_x Z(a-1)$ by (A.2) with a replaced by $(a-1)$. Thus,

$$\frac{e}{d}Z = \frac{e}{d}Z(a-) + xZ(d+,e+) - \frac{(c-b)}{c}xZ(c+,d+,e+),$$

since a and b are interchangeable, there holds

$$\frac{e}{d}Z = \frac{e}{d}Z(b-) + xZ(d+,e+) - \frac{(c-a)}{c}xZ(c+,d+,e+).$$

Subtracting the last two equations from each other yields

$$0 = e(Z(a-) - Z(b-) + c^{-1}d(b-a)xZ(c+,d+,e+)$$

and if we set b to $b+1$

$$0 = e(Z(a-,b+) - Z) + c^{-1}d(b+1-a)xZ(b+,c+,d+,e+).$$

Setting $a = -n, b = n+\alpha+\beta+1, c = \alpha+1$ and $x = 1$ (and the respective values for f, g, h) leads after simplification to recursion formula (2.20).

Proof of mixed relations

Moreover, the recurrence relations, which we have seen in section 2.2, can be derived for the more general case (A.1). All of the following recursion hold for $x = 1$ and $y = 1$, which follows just from the recurrence formula of the Jacobi polynomials, as we have seen.
To prove the mixed relations (2.26) and (2.27) consider

$$\begin{aligned}F :=& \frac{2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!}\\ & F^{1;2;2}_{1;1;1}\left(\begin{matrix}\mu+1 & ; & -n \quad n+\alpha+\beta+1 & ; & -m \quad m+\rho+\delta+1\\ \mu+\nu+2 & ; & \alpha+1 & ; & \rho+1\end{matrix}\,;x;y\right).\end{aligned} \tag{A.3}$$

Denote the contiguous functions as usual, i.e.

$$\begin{aligned}
F(\alpha+) =& \frac{2^{\mu+\nu+1}(\alpha+2)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} \\
& F\left(\begin{matrix} \mu+1 & ; & -n & n+\alpha+\beta+2 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+2 & & ; & \rho+1 & \end{matrix} ;x;y\right) \\
F(\nu+) =& \frac{2^{\mu+\nu+2}(\alpha+1)_n(\rho+1)_m B(\nu+2,\mu+1)}{n!\,m!} \\
& F\left(\begin{matrix} \mu+1 & ; & -n & n+\alpha+\beta+1 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+1 & & ; & \rho+1 & \end{matrix} ;x;y\right) \\
& \dots
\end{aligned}$$

Lemma A.1.1
Let $\theta_x = x\frac{\partial}{\partial x}$ and $\theta_y = y\frac{\partial}{\partial y}$, then the following differential equations hold

$$(\theta_x - n)F = -(n+\alpha)F(n-,\beta+) \tag{A.4}$$

$$(\theta_x + n + \alpha + \beta + 1)F = (n+\alpha+\beta+1)F(\beta+) \tag{A.5}$$

$$(\theta_y - m)F = -(m+\rho)F(m-,\delta+) \tag{A.6}$$

$$(\theta_y + m + \rho + \delta + 1)F = (m+\rho+\delta+1)F(\delta+) \tag{A.7}$$

$$(\theta_x + \theta_y + \mu + \nu + 1)F = 2\nu F(\nu-) \tag{A.8}$$

Proof. For F as in (A.3)

$$\begin{aligned}
(\theta_x - n)F =& \frac{2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} \\
& \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} \frac{(\mu+1)_{k+l}(-n)_k(n+\alpha+\beta+1)_k(-m)_l(m+\rho+\delta+1)_l}{(\mu+\nu+2)_{k+l}(\alpha+1)_k(\rho+1)_l}(\theta_x - n)\frac{x^k y^l}{k!\,l!}, \\
=& \frac{2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} \\
& \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} \frac{(\mu+1)_{k+l}(-n)_k(n+\alpha+\beta+1)_k(-m)_l(m+\rho+\delta+1)_l}{(\mu+\nu+2)_{k+l}(\alpha+1)_k(\rho+1)_l}(k-n)\frac{x^{k-1} y^l}{k!\,l!}.
\end{aligned}$$

As usual, replace $(k-n)(-n)_k$ by $(-n)(-n+1)_k$, thus

$$\begin{aligned}
=& \frac{(-n)2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} \\
& \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} \frac{(\mu+1)_{k+l}(-n+1)_k(n+\alpha+\beta+1)_k(-m)_l(m+\rho+\delta+1)_l}{(\mu+\nu+2)_{k+l}(\alpha+1)_k(\rho+1)_l}\frac{x^k y^l}{k!\,l!} \\
=& \frac{2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} \\
& F^{1;2;2}_{1;1;1}\left(\begin{matrix} \mu+1 & ; & -n-1 & (n-1)+\alpha+\beta+2 & ; & -m & m+\rho+\delta+1 \\ \mu+\nu+2 & ; & \alpha+1 & & ; & \rho+1 & \end{matrix} ;x;y\right).
\end{aligned}$$

Since we changed n to $n-1$ in the series, we need to equate $n+\alpha+\beta+1$ as well. This is done by raising β by one. Furthermore, we need to change the parts in prefactor, to accommodate $n-1$

as well. Since

$$\frac{-n2^{\mu+\nu+1}(\alpha+1)_n(\rho+1)_m B(\nu+1,\mu+1)}{n!\,m!} = -\frac{2^{\mu+\nu+1}(\rho+1)_m B(\nu+1,\mu+1)(\alpha+n)(\alpha+1)_{n-1}}{m!\,(n-1)!}, \tag{A.9}$$

the equation (A.4) follows. Relation (A.6) follows analogue to it. Relation (A.5) or (A.7) follow directly by applying the differential operators, since the prefactor doesn't need to be changed. For the last relation (A.8) the differential operator $(\theta_x+\theta_y+\mu+\nu+1)$ applied to $x^k y^l$ yields $(k+l+\mu+\nu+1)$. This reduces $(\nu+\mu+2)_{k+l}$ in the denominator to $(\nu+\mu+2)_{k+l-1}$. Therefore, we multiply the series with $\frac{\nu+\mu+1}{\nu+\mu+1}$, such that the part in the denominator becomes $(\nu+\mu+1)_{k+l}$. Since it is only a change in the denominator, it is rather a change in ν than in μ. The rest follows using a property of the Beta-function, i.e.

$$B(x+1,y) = B(x,y)\frac{x}{x+y}.$$

□

Subtracting (A.4) and (A.6) from (A.8) proves (2.26).

Lemma A.1.2
For F as in (A.3) the following contiguous recurrence relation holds,

$$\begin{aligned}(n+m+\mu+\nu+4)F(n+,m+,\nu+) &= (n+\alpha+1)F(m+,\beta+,\nu+)+(m+\rho+1)F(n+,\delta+,\nu+)\\ &\quad +2(\nu+1)F(\nu+).\end{aligned}$$

Similar (2.27) can be proven. Take (A.5) and (A.7) and subtract (A.8)

Lemma A.1.3
For F as in (A.3) the following contiguous recurrence relation holds,

$$(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F = (n+\alpha+\beta+1)F(\beta+)+(m+\rho+\delta+1)F(\delta+)+2\nu F(\nu-).$$

The easiest way of deriving (2.28) and (2.29) is to introduce another formulation for F. Recall that Jacobi polynomials can be expressed as

$$P_n^{(\alpha,\beta)}(x) = \frac{(-1)^n(\beta+1)_n}{n!}\,{}_2F_1\left(\begin{matrix}-n,\, n+\alpha+\beta+1\\ \beta+1\end{matrix};\frac{1+x}{2}\right)$$

as well. Using this expression in the derivation of the Kampé de Fériet series yields the analogue form

$$\begin{aligned}\widetilde{F} :=& \frac{(-1)^{n+m}2^{\mu+\nu+1}(\beta+1)_n(\delta+1)_m B(\nu+1,\mu+1)}{m!\,n!}\\ & F^{1;2;2}_{1;1;1}\left(\begin{matrix}\nu+1 & ; & -n & n+\alpha+\beta+1 & ; & -m & m+\rho+\delta+1\\ \mu+\nu+2 & ; & \beta+1 & & ; & \delta+1 & \end{matrix};x;y\right).\end{aligned} \tag{A.10}$$

Then the differential equations can be derived in the same way as seen before. Thus,

Lemma A.1.4

$$(\theta_x - n)\widetilde{F} = (n+\beta)\widetilde{F}(n-,\alpha+), \tag{A.11}$$
$$(\theta_x + n+\alpha+\beta+1)\widetilde{F} = (n+\alpha+\beta+1)\widetilde{F}(\alpha+), \tag{A.12}$$
$$(\theta_y - m)\widetilde{F} = (m+\delta)\widetilde{F}(m-,\rho+), \tag{A.13}$$
$$(\theta_y + m+\rho+\delta+1)\widetilde{F} = (m+\rho+\delta+1)\widetilde{F}(\rho+), \tag{A.14}$$
$$(\theta_x + \theta_y + \mu+\nu+1)\widetilde{F} = 2\mu\widetilde{F}(\mu-). \tag{A.15}$$

Now (2.28) follows by subtracting (A.11) and (A.13) from (A.15).

Lemma A.1.5
For F as in (A.3) or (A.10) the following contiguous recurrence relation holds,

$$\begin{aligned}(n+m+\mu+\nu+4)F(n+,m+,\mu+) = &-(n+\beta+1)F(m+,\alpha+,\mu+)\\ &-(m+\delta+1)F(n+,\rho+,\mu+)+2(\mu+1)F(\mu+).\end{aligned}$$

We derive analogously

Lemma A.1.6
For F as in (A.3) or (A.10) the following contiguous recurrence relation holds,

$$(n+\alpha+\beta+m+\rho+\delta-\mu-\nu+1)F = (n+\alpha+\beta+1)F(\alpha+)+(m+\rho+\delta+1)F(\rho+)+2\mu F(\mu-).$$

Remark 12
Following Burchnall and Chaundy [BC40], [BC41] one can easily compute an expansion base, i.e. $F = \sum_{i=0}^{\infty} A_i(x)A_i(y)$, for F.

A.2. Sparsity results for integrals of Jacobi polynomials

In this section we collect some basic sparsity results for integrals over Jacobi and integrated Jacobi polynomials.

Corollary A.2.1
Let $\alpha,\beta,\mu > -1$; $k,l > 1$ and $\alpha,\beta \leq \mu$, then

$$\int_{-1}^{1}(1-x)^{\mu}\widehat{P}_k^{\mu-\alpha}(x)\widehat{P}_l^{\mu-\beta}(x)\,\mathrm{d}x = \begin{cases}0 & \text{if } k > (\alpha+l+2) \text{ or } l > (\beta+k+2)\\ c(\alpha,\beta,\mu,k,l) & \text{else.}\end{cases} \tag{A.16}$$

Proof. With
$$\widehat{P}_n^{\alpha}(x) = \frac{(1+x)}{n}P_{n-1}^{(\alpha-1,1)}(x)$$

follows

$$\int_{-1}^{1}(1-x)^{\mu}(1+x)^{2}P_{k-1}^{(\mu-\alpha-1,1)}(x)P_{l-1}^{(\mu-\beta-1,1)}(x)\,\mathrm{d}x.$$

By the orthogonality relation

$$\int_{-1}^{1}(1-x)^{\alpha}(1+x)^{\beta}q(x)P_{n}^{(\alpha,\beta)}(x)\,\mathrm{d}x=0,\quad \text{if } q\in P_m,\ \text{where } m<n$$

follows the statement. □

Analogously, we get the following corollary.

Corollary A.2.2
Let $\alpha,\beta,\mu>-1$; $k,l>1$ and $\alpha,\beta\leq\mu$, then

$$\int_{-1}^{1}(1-x)^{\mu}\widehat{P}_{k}^{\mu-\alpha}(x)P_{l-1}^{(\mu-\beta,0)}(x)\,\mathrm{d}x=\begin{cases}0 & \text{if } k>(\alpha+l+1)\text{ or } l>(\beta+k+2)\\ c(\alpha,\beta,\mu,k,l) & \text{else.}\end{cases}\tag{A.17}$$

Furthermore a direct consequence of the Jacobi orthogonality relation is given in corollary A.2.3.

Corollary A.2.3
Let $\alpha,\beta,\mu>-1$; $k,l>1$ and $\alpha,\beta\leq\mu$, then

$$\int_{-1}^{1}(1-x)^{\mu}P_{k-1}^{(\mu-\alpha,0)}(x)P_{l-1}^{(\mu-\beta,0)}(x)\,\mathrm{d}x=\begin{cases}0 & \text{if } k>(\alpha+l)\text{ or } l>(\beta+k)\\ c(\alpha,\beta,\mu,k,l) & \text{else.}\end{cases}\tag{A.18}$$

A.3. Further properties of integrals of Jacobi polynomials

A.3.1. Legendre polynomials

For the Legendre polynomials hold

$$\int_{-1}^{1}L_n(x)L_m(x)\,\mathrm{d}x=\frac{2}{2n+1}\delta_{n,m}.\tag{A.19}$$

The integral

$$\int_{-1}^{1}\widehat{L}_n(x)\widehat{L}_m(x)\,\mathrm{d}x$$

can be computed exactly, by the relation $(2n-1)\widehat{L}_n(x)=L_n(x)-L_{n-2}(x)$ and the orthogonality relations of the Legendre polynomials. It follows

$$\int_{-1}^{1}\widehat{L}_n(x)\widehat{L}_n(x)\,\mathrm{d}x=\frac{1}{(2n-1)^2}\left(\frac{2}{2n+1}+\frac{2}{2n-3}\right)=\frac{4}{(2n+1)(2n-1)(2n-3)}\tag{A.20}$$

and thus the relation

$$\int_{-1}^{1} \widehat{L}_{n+1}(x)\widehat{L}_{n+1}(x)\,\mathrm{d}x = \frac{(2n-3)}{(2n+3)} \int_{-1}^{1} \widehat{L}_n(x)\widehat{L}_n(x)\,\mathrm{d}x.$$

For the entry $(i, i-2)$ we can compute the exact value as follows

$$\int_{-1}^{1} \widehat{L}_n(x)\widehat{L}_{n-2}(x)\,\mathrm{d}x = \int_{-1}^{1} \frac{-1}{(2n-1)(2n-5)} L_{n-2}(x)L_{n-2}(x)\,\mathrm{d}x = \frac{-2}{(2n-1)(2n-3)(2n-5)} \tag{A.21}$$

and thus

$$\int_{-1}^{1} \widehat{L}_{n+1}(x)\widehat{L}_{n-1}(x)\,\mathrm{d}x = \frac{(2n-5)}{(2n+1)} \int_{-1}^{1} \widehat{L}_n(x)\widehat{L}_{n-2}(x)\,\mathrm{d}x.$$

Analogously, we can derive that

$$\begin{aligned} \int_{-1}^{1} \widehat{L}_n(x)L_m(x)\,\mathrm{d}x &= \frac{1}{2n-1}\left(\int_{-1}^{1} L_n(x)L_m(x)\,\mathrm{d}x - \int_{-1}^{1} L_{n-2}(x)L_m(x)\,\mathrm{d}x\right) \\ &= \frac{\delta_{n,m}}{(2n-1)(2n+1)} - \frac{\delta_{n-2,m}}{(2n-1)(2n-3)}. \end{aligned} \tag{A.22}$$

A.3.2. Jacobi polynomials

Some of the integrals over low order (integrated) Jacobi polynomials can be computed directly, e.g.

$$\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_1^{2i}(y)\widehat{P}_1^{2i}(y)\,\mathrm{d}y = \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \left(\frac{1+y}{2}\right)^{2} \mathrm{d}y = B(i+i'+2,3).$$

The Beta function fulfils the natural recursive relations

$$B(i+i'+2,3) = \frac{i+i'+1}{i+i'+4}B(i+i'+1,3), \text{ and } B(i+i'+2,3) = \frac{2}{i+i'+4}B(i+i'+2,2) \tag{A.23}$$

Furthermore the following recursive relations can be found with the symbolic software `Guess` [Kau09]. Let

$$I_j^{i,i'} = \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)\widehat{P}_1^{2i'}(y)\,\mathrm{d}y. \tag{A.24}$$

Corollary A.3.1
For $j > 1$ and $i, i' > 0$, the recursive relation

$$(i+i'+j+5)I_j^{i+1,i'+1} = (i+i'+2)I_j^{i,i'} + (i-i'+j-5)I_j^{i,i'+1} + (i'-i+2)I_j^{i+1,i'}, \tag{A.25}$$

holds.

Moreover, most starting values can be computed exactly. An integral of the form

$$\int_{-1}^{1} (1-y)^z(1+y)^\beta P_j^{(\alpha,\beta)}(y)\,\mathrm{d}y$$

can be interpreted as a Mellin-transform of the Jacobi-polynom $P_j^{(\alpha,\beta)}(y)$, see [DLM, 18.17vii].

Lemma A.3.2 (Mellin transformation)
Let $\alpha, \beta > -1$ and $j > 0$, then

$$\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y)\, \mathrm{d}y = \frac{\Gamma(z)\Gamma(1+\beta+j)(1+\alpha-z)_j}{j!\,\Gamma(1+\beta+z+j)} \tag{A.26}$$

Proof. As stated in the literature, replace the Jacobi polynomial by (2.7), apply the representation of the Beta integral (2.1) and use the Pfaff-Saalschütz theorem 2.1.2. □

In general, we can't use lemma A.3.2 on our (integrated) Jacobi polynomials, since the exponents β_1 of the weight $(1+y)^{\beta_1}$ and the indices β_2 of $P_j^{(\alpha,\beta_2)}(y)$ differ. By application of the same techniques, we can derive a similar formulation.

Corollary A.3.3
Let $\alpha, \beta > -1$ and $j > 0$, then

$$\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \widehat{P}_j^{\alpha}(y)\widehat{P}_1^{\rho}(y)\, \mathrm{d}y$$
$$= \frac{16}{(j+\alpha-z-1)} \frac{\Gamma(z+1)}{\Gamma(z+j+3)} \left(\frac{1}{2}(j+\alpha)(j+1)(\alpha-z-1)_{j-1} - (z+1)(\alpha-z-2)_{j-1}\right)$$

Proof. We begin by rewriting the integrated Jacobi polynomials as Jacobi polynomials, i.e.

$$\widehat{P}_j^{\alpha}(y) = \frac{1+y}{j} P_{j-1}^{(2i-1,1)}(y), \text{ and } \widehat{P}_1^{\rho}(y) = (1+y).$$

Now we apply the same steps, as for the proof of the Mellin-transformation.

$$\begin{aligned}
\frac{4}{j}\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{2} P_{j-1}^{(\alpha-1,1)}(y)\, \mathrm{d}y &= \frac{4(\alpha)_{j-1}}{j!} \sum_{l=0}^{\infty} \frac{(-(j-1))_l (j+\alpha)_l}{(\alpha)_l l!} \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{l+z} \left(\frac{1+y}{2}\right)^{2} \mathrm{d}y \\
&= \frac{8(\alpha)_{j-1}}{j!} \sum_{l=0}^{\infty} \frac{(-(j-1))_l (j+\alpha)_l}{(\alpha)_l l!} \frac{\Gamma(l+z+1)\Gamma(3)}{\Gamma(l+z+4)} \\
&= \frac{16(\alpha)_{j-1}}{j!} \frac{\Gamma(z+1)}{\Gamma(z+4)} {}_3F_2\left(\begin{matrix} -(j-1), j+\alpha, z+1 \\ \alpha, z+4 \end{matrix}; 1\right).
\end{aligned} \tag{A.27}$$

The hypergeometric series is not *Saalschützian* and theorem 2.1.2 can not be applied. On the other hand, it fulfils the contiguous relation

$$(b-a){}_3F_2\left(\begin{matrix} a, b, d \\ c, e \end{matrix}; 1\right) + a\,{}_3F_2\left(\begin{matrix} a+1, b, d \\ c, e \end{matrix}; 1\right) - b\,{}_3F_2\left(\begin{matrix} a, b+1, d \\ c, e \end{matrix}; 1\right) = 0,$$

see [AAR99, Chap. 3.7]. By this relation, we can replace our ${}_3F_2$ series by two *Saalschützian* series,

$$
{}_3F_2\left(\begin{matrix}-(j-1), j+\alpha, z+1\\ \alpha, z+4\end{matrix};1\right) =
$$
$$
\frac{1}{j+\alpha-z-1}\left((j+\alpha){}_3F_2\left(\begin{matrix}-(j-1), j+\alpha+1, z+1\\ \alpha, z+4\end{matrix};1\right)-(z+1){}_3F_2\left(\begin{matrix}-(j-1), j+\alpha, z+2\\ \alpha, z+4\end{matrix};1\right)\right).
$$

Since both series are balanced we can apply theorem 2.1.2 on both series, thus

$$
\begin{aligned}
&{}_3F_2\left(\begin{matrix}-(j-1), j+\alpha, z+1\\ \alpha, z+4\end{matrix};1\right)\\
&=\frac{1}{j+\alpha-z-1}\left(\frac{(j+\alpha)(-(j+1))_{j-1}(\alpha-z-1)_{j-1}}{(\alpha)_{j-1}(-j-z-2)_{j-1}}-\frac{(z+1)(-j)_{j-1}(\alpha-z-2)_{j-1}}{(\alpha)_{j-1}(-j-z-2)_{j-1}}\right)\\
&=\frac{\Gamma(j+1)}{j+\alpha-z-1}\left(\frac{\frac{1}{2}(j+1)(j+\alpha)(\alpha-z-1)_{j-1}}{(\alpha)_{j-1}(z+4)_{j-1}}-\frac{(z+1)(\alpha-z-2)_{j-1}}{(\alpha)_{j-1}(z+4)_{j-1}}\right)\\
&=\frac{\Gamma(j+1)}{(j+\alpha-z-1)(\alpha)_{j-1}(z+4)_{j-1}}\left(\frac{1}{2}(j+\alpha)(j+1)(\alpha-z-1)_{j-1}-(z+1)(\alpha-z-2)_{j-1}\right).
\end{aligned}
$$

Inserted in (A.27) gives

$$
\begin{aligned}
&\frac{16(\alpha)_{j-1}}{j!}\frac{\Gamma(z+1)}{\Gamma(z+4)}\frac{\Gamma(j+1)}{(j+\alpha-z-1)(\alpha)_{j-1}(z+4)_{j-1}}\left(\frac{1}{2}(j+\alpha)(j+1)(\alpha-z-1)_{j-1}-(z+1)(\alpha-z-2)_{j-1}\right)\\
&=\frac{16}{(j+\alpha-z-1)}\frac{\Gamma(z+1)}{\Gamma(z+j+3)}\left(\frac{1}{2}(j+\alpha)(j+1)(\alpha-z-1)_{j-1}-(z+1)(\alpha-z-2)_{j-1}\right).
\end{aligned}
$$

□

Here, the factor $(j+\alpha-z-1)$ contains a singularity. A better choice is given by a similar formulation.

Corollary A.3.4
Let $\alpha, \beta > -1$ and $j > 0$, then

$$
\begin{aligned}
&\int_{-1}^{1}\left(\frac{1-y}{2}\right)^{z}\widehat{P}_j^{\alpha}(y)\widehat{P}_1^{\rho}(y)\,\mathrm{d}y\\
&=16\frac{(\alpha-z-1)_{j-2}}{(z+1)_{j+2}}\left((\alpha-z-2)-\frac{1}{2}(1-j)(j+\alpha)\right)
\end{aligned}
$$

Proof. The proof follows the same steps as in the previous corollary. But we raise the hypergeometric series by the relation

$$
de\left({}_3F_2\left(\begin{matrix}a, b, c\\ d, e\end{matrix};z\right)-{}_3F_2\left(\begin{matrix}a+1, b, c\\ d, e\end{matrix};z\right)\right)+zbc\,{}_3F_2\left(\begin{matrix}a+1, b+1, c+1\\ d+1, e+1\end{matrix};z\right)=0, \tag{A.28}
$$

see [WF, 07.27.17.0012.01]. The rest follows by combinatorical arguments. □

Although the exact values of corollary A.3.4 contain Pochhammer symbols, the application to the

starting values by recurrence relation is straight forward. Since $z = \alpha + c$, where c is a constant, the terms $(\alpha - z - 1)_{j-1}$ and $(\alpha - z - 1)_{j-1}$ remain the same along the translation $\alpha + d, z + d$, where $d \in \mathbb{Z}$. On the other hand j is usually a small number, and such the computation of the Pochhammer symbols are not that costly. The fraction $\frac{\Gamma(z+1)}{\Gamma(z+j+3)}$ has only $j+2$ terms and could be evaluated directly or by a simple recurrence relation.
For the gradient or derivatives of our functions, we need starting values of the form

$$\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y) P_1^{(\rho,0)}(y)\, \mathrm{d}y,$$

where β is either 0 or 1 for a mixed case between integrated Jacobi polynomials and Jacobi polynomials.

Corollary A.3.5
Let $\alpha, \beta > -1$ and $j > 0$, then

$$\begin{aligned} &\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y) P_1^{(\rho,0)}\, \mathrm{d}y \\ &= \frac{(\rho+1)}{2} \frac{\Gamma(z)\Gamma(1+\beta+j)(1+\alpha-z)_j}{j!\,\Gamma(1+\beta+z+j)} - \frac{(\rho+2)}{2} \frac{\Gamma(z+1)\Gamma(1+\beta+j)(\alpha-z)_j}{j!\,\Gamma(2+\beta+z+j)} \end{aligned} \tag{A.29}$$

Proof. Here we replace the low order Jacobi polynomial by

$$P_1^{(\rho,0)}(y) = \frac{(\rho+1)}{2} \left(1 + \frac{(-1)(\rho+2)}{(\rho+1)} \left(\frac{1-y}{2}\right)\right).$$

Thus we can apply lemma A.3.2 two times, i.e.

$$\begin{aligned} &\int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y) P_1^{(\rho,0)}\, \mathrm{d}y \\ &= \frac{(\rho+1)}{2} \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y)\, \mathrm{d}y - \frac{(\rho+2)}{2} \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{z+1} \left(\frac{1+y}{2}\right)^{\beta} P_j^{(\alpha,\beta)}(y)\, \mathrm{d}y \\ &= \frac{(\rho+1)}{2} \frac{\Gamma(z)\Gamma(1+\beta+j)(1+\alpha-z)_j}{j!\,\Gamma(1+\beta+z+j)} - \frac{(\rho+2)}{2} \frac{\Gamma(z+1)\Gamma(1+\beta+j)(\alpha-z)_j}{j!\,\Gamma(2+\beta+z+j)} \end{aligned}$$

□

A.4. Products of partial derivatives

In the following chapter, we will list all combinations of partial derivatives under the inner product. To simplify readability we define the following auxiliary arrays,

$$
\begin{aligned}
I_1[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'-1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_2[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_3[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)\widehat{P}_{j'}^{2i'}(y)\,\mathrm{d}y,\\
I_4[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'} \widehat{P}_j^{2i}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_5[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} \widehat{P}_j^{2i}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_6[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+1} P_{j-1}^{(2i,0)}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,\\
I_7[i,i',j,j'] &:= \int_{-1}^{1} \left(\frac{1-y}{2}\right)^{i+i'+2} P_{j-1}^{(2i,0)}(y)P_{j'-1}^{(2i',0)}(y)\,\mathrm{d}y,
\end{aligned}
$$

and the auxiliary arrays with respect to the Legendre polynomials,

$$
\begin{aligned}
L_1[i,i'] &:= \int_{-1}^{1} \widehat{L}_i(y)\widehat{L}_{i'}(y)\,\mathrm{d}y,\\
L_2[i,i'] &:= \int_{-1}^{1} L_{i-1}(y)\widehat{L}_{i'}(y)\,\mathrm{d}y,\\
L_3[i,i'] &:= \int_{-1}^{1} L_{i-1}(y)L_{i'-1}(y)\,\mathrm{d}y.
\end{aligned}
$$

To simplify notation even further, we write $\beta = i + j$ and $\beta = i' + j'$, then the H^1 shape function on a tetrahedron is given by

$$
u_{ijk}^{\blacktriangle} = \widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta}\widehat{P}_k^{2\beta}(z),
$$

we also apply this notation to the partial derivatives, which are then given by

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}x} u_{ijk}^{\blacktriangle}(x,y,z) =& L_{i-1}\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
\frac{\mathrm{d}}{\mathrm{d}y} u_{ijk}^{\blacktriangle}(x,y,z) =& \frac{1}{2}L_{i-2}\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
&+\widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i}P_{j-1}^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
\frac{\mathrm{d}}{\mathrm{d}z} u_{ijk}^{\blacktriangle}(x,y,z) =& \frac{1}{4}L_{i-2}\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i-1}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
&-\frac{i}{2j+2i-2}\widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i}P_{j-1}^{(2i,0)}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
&+\frac{j-1}{2j+2i-2}\widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i}P_{j-2}^{(2i,0)}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta-1}\widehat{P}_k^{2\beta}(z)\\
&+\widehat{L}_i\left(\frac{4x}{1-2y-z}\right)\left(\frac{1-y}{2}\right)^{i}\widehat{P}_j^{2i}\left(\frac{2y}{1-z}\right)\left(\frac{1-z}{2}\right)^{\beta}P_{k-1}^{(2\beta,0)}(z).
\end{aligned}$$

After application of the Duffy transformation $D_3(x,y,z) = \left(\frac{4x}{1-2y-z}, \frac{2y}{1-z}, z\right) =: (\eta_1, \eta_2, \eta_3)$, e.g. we can write the integrals over the partial x-derivative as follows,

$$\begin{aligned}
&\int_{\blacktriangle} \frac{\mathrm{d}}{\mathrm{d}x} u_{ijk}^{\blacktriangle}(x,y,z)\frac{\mathrm{d}}{\mathrm{d}x} u_{i'j'k'}^{\blacktriangle}(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\\
&= \int_{-1}^{1} L_{i-1}(\eta_1)L_{i'-1}(\eta_1)\,\mathrm{d}\eta_1 \int_{-1}^{1}\left(\frac{1-\eta_2}{2}\right)^{i+i'-1}\widehat{P}_j^{2i}(\eta_2)\widehat{P}_{j'}^{2i'}(\eta_2)\,\mathrm{d}\eta_2\int_{-1}^{1}\left(\frac{1-\eta_2}{2}\right)^{\beta+\beta'}\widehat{P}_k^{2\beta}(\eta_3)\widehat{P}_{k'}^{2\beta'}(\eta_3)\,\mathrm{d}\eta_3\\
&= L_3[i,i']I_1[i,i',j,j']I_2[k,k',i+j,i'+j'].
\end{aligned}$$

The integrals over the rest of possible combinations are collected in the tables A.1 and A.2. It is important to note, that we only need to compute the 7 auxiliary arrays $I_1, \dots, I_7$, where we apply the sparsity results of appendix A.2. The auxiliary arrays L_1, L_2 and L_3 are computed by the results in appendix A.3.1 and appendix A.3.2.

Table A.1.: Integral of the product of the partial derivatives of $u^{\blacktriangle}_{ijk}$

dx × dx			$L_3[i,i']$	$I_1[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
dx × dy		$\frac{1}{2}$	$L_3[i,i'-1]$	$I_1[i,i',j,j']$	$I_2[k,k',i+j,i'+j']$
	$+$		$L_2[i,i']$	$I_4[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
dx × dz		$\frac{1}{4}$	$L_3[i,i'-1]$	$I_1[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$-$	$\frac{i'}{2i'+2j'-2}$	$L_2[i,i']$	$I_4[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{j'-1}{2i'+2j'-2}$	$L_2[i,i']$	$I_4[i,i',j,j'-1]$	$I_2[i+j,i'+j',k,k']$
	$+$		$L_2[i,i']$	$I_2[i,i',j,j']$	$I_5[i+j,i'+j',k,k']$
dy × dy		$\frac{1}{4}$	$L_3[i-1,i'-1]$	$I_1[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{1}{2}$	$L_2[i-1,i']$	$I_4[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{1}{2}$	$L_2[i'-1,i]$	$I_4[i',i,j',j]$	$I_2[i+j,i'+j',k,k']$
	$+$		$L_1[i,i']$	$I_6[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
dy × dz		$\frac{1}{8}$	$I_3[i-1,i'-1]$	$I_1[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$-$	$\frac{i'}{2(2i'+2j'-2)}$	$L_2[i-1,i']$	$I_4[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{j'-1}{2(2i'+2j'-2)}$	$L_2[i-1,i']$	$I_4[i,i',j,j'-1]$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{1}{2}$	$L_2[i-1,i']$	$I_2[i,i',j,j']$	$I_5[i+j,i'+j',k,k']$
	$+$	$\frac{1}{4}$	$L_2[i'-1,i']$	$I_4[i',i,j',j,]$	$I_2[i+j,i'+j',k,k']$
	$-$	$\frac{i'}{2i'+2j'-2}$	$L_3[i,i']$	$I_6[i,i',j,j']$	$I_2[i+j,i'+j',k,k']$
	$+$	$\frac{j'-1}{2i'+2j'-2}$	$L_3[i,i']$	$I_6[i,i',j,j'-1]$	$I_2[i+j,i'+j',k,k']$
	$+$		$L_3[i,i']$	$I_5[i',i,j',j]$	$I_5[i+j,i'+j',k,k']$

Table A.2.: Integral of the product of the partial z-derivatives of $u^{\blacktriangle}_{ijk}$

dz × dz		$\frac{1}{16}$	$L_3[i-1, i'-1]$	$I_1[i, i', j, j']$	$I_2[i+j, i'+j', k, k']$
	$-$	$\frac{i'}{4(2i'+2j'-2)}$	$L_2[i-1, i']$	$I_4[i, i', j, j']$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{j'-1}{4(2i'+2j'-2)}$	$L_2[i-1, i']$	$I_4[i, i', j, j'-1]$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{1}{4}$	$L_2[i-1, i']$	$I_2[i, i', j, j']$	$I_5[i+j, i'+j', k, k']$
	$-$	$\frac{i}{4(2i+2j-2)}$	$L_2[i'-1, i]$	$I_4[i', i, j', j]$	$I_2[i+j, i'+j', k, k']$
	$-$	$\frac{i'}{4(2i'+2j'-2)} \frac{j-1}{4(2i+2j-2)}$	$L_3[i, i']$	$I_6[i, i', j, j']$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{j'-1}{4(2i'+2j'-2)} \frac{j-1}{4(2i+2j-2)}$	$L_3[i, i']$	$I_6[i, i', j, j'-1]$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{j-1}{4(2i+2j-2)}$	$L_3[i, i']$	$I_5[i, i', j, j']$	$I_2[i+j, i'+j', k, k']$
	$-$	$\frac{j-1}{4(2i+2j-2)}$	$L_2[i'-1, i]$	$I_4[i', i, j', j-1]$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{i'}{4(2i'+2j'-2)} \frac{i}{4(2i+2j-2)}$	$L_3[i, i']$	$I_6[i, i', j-1, j']$	$I_2[i+j, i'+j', k, k']$
	$-$	$\frac{j'-1}{4(2i'+2j'-2)} \frac{i}{4(2i+2j-2)}$	$L_3[i, i']$	$I_6[i, i', j-1, j'-1]$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{j-1}{4(2i+2j-2)}$	$L_3[i, i']$	$I_5[i', i, j', j-1]$	$I_2[i+j, i'+j', k, k']$
	$+$	$\frac{1}{4}$	$L_2[i', i-1]$	$I_2[i, i', j, j']$	$I_5[i'+j', i+j, k', k]$
	$-$	$\frac{i'}{4(2i'+2j'-2)}$	$L_3[i, i']$	$I_5[i, i', j, j']$	$I_5[i'+j', i+j, k', k]$
	$+$	$\frac{j'-1}{4(2i'+2j'-2)}$	$L_3[i, i']$	$I_5[i, i', j, j']$	$I_5[i'+j', i+j, k', k]$
	$+$		$L_3[i, i']$	$I_3[i, i', j, j']$	$I_7[i+j, i'+j', k, k']$

Bibliography

[AAB+21] R. Anderson, J. Andrej, A. Barker, J. Bramwell, J.-S. Camier, J. Cerveny, V. Dobrev, Y. Dudouit, A. Fisher, Tz. Kolev, W. Pazner, M. Stowell, V. Tomov, I. Akkerman, J. Dahm, D. Medina, and S. Zampini, MFEM: A modular finite element methods library, Computers & Mathematics with Applications **81** (2021), 42–74.

[AAD11] Mark Ainsworth, Gaelle Andriamaro, and Oleg Davydov, Bernstein-Bézier finite elements of arbitrary order and optimal assembly procedures, SIAM J. Sci. Comput. **33** (2011), no. 6, 3087–3109. MR 2862005

[AAR99] George E. Andrews, Richard Askey, and Ranjan Roy, Special functions, Encyclopedia of Mathematics and its Applications, vol. 71, Cambridge University Press, Cambridge, 1999. MR 1688958

[AB95] Manmohan S. Arora and S. D. Bajpai, A new proof of the orthogonality of Jacobi polynomials, Demonstratio Math. **28** (1995), no. 1, 177–180. MR 1330093

[ABCM02] Douglas N. Arnold, Franco Brezzi, Bernardo Cockburn, and L. Donatella Marini, Unified analysis of discontinuous galerkin methods for elliptic problems, SIAM Journal on Numerical Analysis **39** (2002), no. 5, 1749–1779.

[ABD+21] Daniel Arndt, Wolfgang Bangerth, Denis Davydov, Timo Heister, Luca Heltai, Martin Kronbichler, Matthias Maier, Jean-Paul Pelteret, Bruno Turcksin, and David Wells, The deal.II finite element library: Design, features, and insights, Computers & Mathematics with Applications **81** (2021), 407–422.

[ABDG98] C. Amrouche, C. Bernardi, M. Dauge, and V. Girault, Vector potentials in three-dimensional non-smooth domains, Mathematical Methods in the Applied Sciences **21** (1998), no. 9, 823–864.

[ABF+22] Daniel Arndt, Wolfgang Bangerth, Marco Feder, Marc Fehling, Rene Gassmöller, Timo Heister, Luca Heltai, Martin Kronbichler, Matthias Maier, Peter Munch, Jean-Paul Pelteret, Simon Sticko, Bruno Turcksin, and David Wells, The `deal.II` library, version 9.4, Journal of Numerical Mathematics **30** (2022), no. 3, 231–246.

[AC01] Mark Ainsworth and Joe Coyle, Hierarchic hp-edge element families for maxwell's equations on hybrid quadrilateral/triangular meshes, Computer Methods in Applied Mechanics and Engineering **190** (2001), no. 49, 6709–6733.

[AE15] W.M. Abd-Elhameed, New product and linearization formulae of jacobi polynomials of certain parameters, Integral Transforms and Special Functions **26** (2015), no. 8, 586–599.

[AF18] Mark Ainsworth and Guosheng Fu, Bernstein-Bézier bases for tetrahedral finite elements, Comput. Methods Appl. Mech. Engrg. **340** (2018), 178–201. MR 3845208

[AK26] P. Appell and J. Kampé de Fériet, Fonctions hypergéométriques et hypersphériques. polynomes d'hermite., VII + 434 p. Paris, Gauthier-Villars (1926)., 1926.

[App25] Paul Appell, Sur les fonctions hypergéométriques de plusieurs variables, les polynômes d'Hermite et autres fonctions sphériques dans l'hyperespace, Mémorial des sciences mathématiques, no. 3, Gauthier-Villars, 1925 (fr).

[AS65] M. Abramowitz and I.A. Stegun, Handbook of mathematical functions: With formulas, graphs, and mathematical tables, Applied mathematics series, Dover Publications, 1965.

[AV96] Ana Alonso and Alberto Valli, Some remarks on the characterization of the space of tangential traces of $H(\mathrm{rot};\Omega)$ and the construction of an extension operator, Manuscripta Math. **89** (1996), no. 2, 159–178. MR 1371994

[Bai64] W. N. Bailey, Generalized hypergeometric series, Cambridge Tracts in Mathematics and Mathematical Physics, No. 32, Stechert-Hafner, Inc., New York, 1964. MR 0185155

[BC40] J. L. Burchnall and T. W. Chaundy, Expansions of Appell's double hypergeometric functions, Quart. J. Math., Oxford Ser. **11** (1940), 249–270. MR 0003885

[BC41] ———, Expansions of Appell's double hypergeometric functions. II, Quart. J. Math., Oxford Ser. **12** (1941), 112–128. MR 0005208

[BCDD06] Daniele Boffi, Martin Costabel, Monique Dauge, and Leszek Demkowicz, Discrete compactness for the hp version of rectangular edge finite elements, SIAM J. Numer. Anal. **44** (2006), no. 3, 979–1004. MR 2231852

[BE55] Harry Bateman and Arthur Erdélyi, Higher transcendental functions, California Institute of technology. Bateman Manuscript project, McGraw-Hill, New York, NY, 1955.

[Bey98] Jürgen Bey, Finite-Volumen- und Mehrgitter-Verfahren für elliptische Randwertprobleme, Advances in Numerical Mathematics, B. G. Teubner, Stuttgart, 1998. MR 1649911

[BF91] F. Brezzi and M. Fortin, Mixed and hybrid finite element methods, Springer series in computational mathematics, Springer-Verlag, 1991.

[BGP89] I. Babuška, M. Griebel, and J. Pitkäranta, The problem of selecting the shape functions for ap-type finite element, International Journal for Numerical Methods in Engineering **28** (1989), no. 8, 1891–1908.

[BH09] Alexei Bespalov and Norbert Heuer, Optimal error estimation for H(curl)-conforming p-interpolation in two dimensions, SIAM J. Numer. Anal. **47** (2009), no. 5, 3977–3989. MR 2576528

[BHP23] Sven Beuchler, Tim Haubold, and Veronika Pillwein, Recursion formulas for integrated products of jacobi polynomials, Constructive Approximation (2023).

[BP07] S. Beuchler and V. Pillwein, Sparse shape functions for tetrahedral p-FEM using integrated Jacobi polynomials, Computing **80** (2007), no. 4, 345–375. MR 2336235

[BP08] ______, Completions to sparse shape functions for triangular and tetrahedral p-FEM, Domain decomposition methods in science and engineering XVII, Lect. Notes Comput. Sci. Eng., vol. 60, Springer, Berlin, 2008, pp. 435–442. MR 2436111

[BPSZ12] S. Beuchler, V. Pillwein, J. Schöberl, and S. Zaglmayr, Sparsity optimized high order finite element functions on simplices, Numerical and symbolic scientific computing, Texts Monogr. Symbol. Comput., SpringerWienNewYork, Vienna, 2012, pp. 21–44. MR 3060506

[BPZ12] S. Beuchler, V. Pillwein, and S. Zaglmayr, Sparsity optimized high order finite element functions for h(div) on simplices, Numerische Mathematik **122** (2012), no. 2, 197–225.

[BPZ13a] ______, Fast summation techniques for sparse shape functions in tetrahedral hp-FEM, Domain decomposition methods in science and engineering XX, Lect. Notes Comput. Sci. Eng., vol. 91, Springer, Heidelberg, 2013, pp. 511–518. MR 3243028

[BPZ13b] ______, Sparsity optimized high order finite element functions for h(curl) on tetrahedra, Advances in Applied Mathematics **50** (2013), no. 5, 749–769.

[Bra13] D. Braess, Finite elemente: Theorie, schnelle löser und anwendungen in der elastizitätstheorie, Masterclass, Springer Berlin Heidelberg, 2013.

[BS06] S. Beuchler and J. Schöberl, New shape functions for triangular p-FEM using integrated Jacobi polynomials, Numer. Math. **103** (2006), no. 3, 339–366. MR 2221053

[BS07] S. Brenner and R. Scott, The mathematical theory of finite element methods, Texts in Applied Mathematics, Springer New York, 2007.

[BSS04] S. Beuchler, R. Schneider, and C. Schwab, Multiresolution weighted norm equivalences and applications, Numerische Mathematik **98** (2004), no. 1, 67–97.

[CDZ00] Zhiming Chen, Qiang Du, and Jun Zou, Finite element methods with matching and nonmatching meshes for Maxwell equations with discontinuous coefficients, SIAM J. Numer. Anal. **37** (2000), no. 5, 1542–1570. MR 1759906

[Ces96] Michel Cessenat, Mathematical methods in electromagnetism, Series on Advances in Mathematics for Applied Sciences, vol. 41, World Scientific Publishing Co., Inc., River Edge, NJ, 1996, Linear theory and applications. MR 1409140

[CFPP14] C. Carstensen, M. Feischl, M. Page, and D. Praetorius, Axioms of adaptivity, Comput. Math. Appl. **67** (2014), no. 6, 1195–1253. MR 3170325

[Chi11] T.S. Chihara, An introduction to orthogonal polynomials, Dover Books on Mathematics, Dover Publications, 2011.

[Chy98] F. Chyzak, Gröbner bases, symbolic summation and symbolic integration, Gröbner bases and applications (Linz, 1998), London Math. Soc. Lecture Note Ser., vol. 251, Cambridge Univ. Press, Cambridge, 1998, pp. 32–60. MR MR1699813 (2001a:68115)

[Chy00] Frédéric Chyzak, An extension of zeilberger's fast algorithm to general holonomic functions, Discrete Mathematics **217** (2000), no. 1-3, 115–134.

[Cia78] P.G. Ciarlet, The finite element method for elliptic problems, ISSN, Elsevier Science, 1978.

[CK10] Hamza Chaggara and Wolfram Koepf, On linearization coefficients of jacobi polynomials, Applied Mathematics Letters **23** (2010), no. 5, 609–614.

[CMC+15] C.D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. De Grazia, S. Yakovlev, J.-E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R.M. Kirby, and S.J. Sherwin, Nektar++: An open-source spectral/hp element framework, Computer Physics Communications **192** (2015), 205–219.

[Coh02] Gary C. Cohen, Higher-order numerical methods for transient wave equations, Scientific Computation, Springer-Verlag, Berlin, 2002, With a foreword by R. Glowinski. MR 1870851

[Dem06] L. Demkowicz, Computing with hp-adaptive finite elements: Volume 1 one and two dimensional elliptic and maxwell problems, Chapman & Hall/CRC Applied Mathematics & Nonlinear Science, CRC Press, 2006.

[DHT14] T. A Driscoll, N. Hale, and L. N. Trefethen, Chebfun guide, Pafnuty Publications, 2014.

[DKP+08] Leszek Demkowicz, Jason Kurtz, David Pardo, Maciej Paszyński, Waldemar Rachowicz, and Adam Zdunek, Computing with *hp*-adaptive finite elements. Vol. 2, Chapman & Hall/CRC Applied Mathematics and Nonlinear Science Series, Chapman & Hall/CRC, Boca Raton, FL, 2008, Frontiers: three dimensional elliptic and Maxwell problems with applications. MR 2406401

[DLM] *n*ist digital library of mathematical functions, `https://dlmf.nist.gov/`, Release 1.1.9 of 2023-03-15, "F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds.".

[DORH89] L. Demkowicz, J.T. Oden, W. Rachowicz, and O. Hardy, Toward a universal adaptive finite element strategy, part 1. constrained approximation and data structure, Computer Methods in Applied Mechanics and Engineering **77** (1989), no. 1-2, 79–112.

[Dou06] John Dougall, On Vandermonde's Theorem, and some more general Expansions, Proceedings of the Edinburgh Mathematical Society **25** (1906), 114–132.

[DPCF+20] Vincent Darrigrand, David Pardo, Théophile Chaumont-Frelet, Ignacio Gómez-Revuelto, and Luis Emilio Garcia-Castillo, A painless automatic *hp*-adaptive strategy for elliptic problems, Finite Elem. Anal. Des. **178** (2020), 103424, 22. MR 4134064

[DSZK16] Paolo Di Stolfo, Andreas Schröder, Nils Zander, and Stefan Kollmannsberger, An easy treatment of hanging nodes in hp-finite elements, Finite Elements in Analysis and Design **121** (2016), 101–117.

[Dub91] Moshe Dubiner, Spectral methods on triangles and other domains, J. Sci. Comput. **6** (1991), no. 4, 345–390. MR 1154903

[Dub00] F. Dubois, Lecture notes: Du tourbillon au champ de vitesse, 2000.

[Duf82] Michael G. Duffy, Quadrature over a pyramid or cube of integrands with a singularity at a vertex, SIAM J. Numer. Anal. **19** (1982), no. 6, 1260–1262. MR 679664

[DX14] Charles F. Dunkl and Yuan Xu, Orthogonal polynomials of several variables, second ed., Encyclopedia of Mathematics and its Applications, vol. 155, Cambridge University Press, Cambridge, 2014. MR 3289583

[EG21] Alexandre Ern and Jean-Luc Guermond, Finite elements I—Approximation and interpolation, Texts in Applied Mathematics, vol. 72, Springer, Cham, [2021] ©2021. MR 4242224

[EGK08] C. Eck, H. Garcke, and P. Knabner, Mathematische modellierung, Springer-Lehrbuch, Springer Berlin Heidelberg, 2008.

[EM05] Tino Eibner and Jens Melenk, Fast algorithms for setting up the stiffness matrix in hp-fem: a comparison, Preprintreihe des Chemnitzer SFB 393, 05-08 (2005), 1–39.

[EMOT54] A. Erdélyi, W. Magnus, F. Oberhettinger, and F. G. Tricomi, Tables of integral transforms. Vol. II, McGraw-Hill Book Company, Inc., New York-Toronto-London, 1954, Based, in part, on notes left by Harry Bateman. MR 0065685

[Ext76] Harold Exton, Multiple hypergeometric functions and applications, Ellis Horwood Ltd., Chichester; Halsted Press [John Wiley & Sons, Inc.], New York-London-Sydney, 1976, Foreword by L. J. Slater, Mathematics & its Applications. MR 0422713

[Fas47] Sister Mary Celine Fasenmyer, Some generalized hypergeometric polynomials, Bulletin of the American Mathematical Society **53** (1947), no. 8, 806–812.

[Fas49] ———, A note on pure recurrence relations, The American Mathematical Monthly **56** (1949), no. 1P1, 14–17.

[FKDN15] Federico Fuentes, Brendan Keith, Leszek Demkowicz, and Sriram Nagaraj, Orientation embedded high order shape functions for the exact sequence elements of all shapes, Comput. Math. Appl. **70** (2015), no. 4, 353–458. MR 3372037

[Gas70] George Gasper, Linearization of the product of jacobi polynomials. i, Canadian Journal of Mathematics **22** (1970), no. 1, 171–175.

[Gau82] Walter Gautschi, On generating orthogonal polynomials, SIAM J. Sci. Statist. Comput. **3** (1982), no. 3, 289–317. MR 667829

[Gau04] ———, Orthogonal polynomials: computation and approximation, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, 2004, Oxford Science Publications. MR 2061539

[GB86a] W. Gui and I. Babuška, The h, p and h-p versions of the finite element method in 1 dimension. I. The error analysis of the p-version, Numer. Math. **49** (1986), no. 6, 577–612. MR 861522

[GB86b] ______, The h, p and h-p versions of the finite element method in 1 dimension. II. The error analysis of the h- and h-p versions, Numer. Math. **49** (1986), no. 6, 613–657. MR 861523

[GB86c] Guo B. and Babuška I., The h-p version of the finite element method, Computational Mechanics **1** (1986), no. 1, 21–41.

[GD10] P. Gatto and L. Demkowicz, Construction of h1-conforming hierarchical shape functions for elements of all shapes and transfinite interpolation, Finite Elements in Analysis and Design **46** (2010), no. 6, 474–486, The Twenty-First Annual Robert J. Melosh Competition.

[GR86] Vivette Girault and Pierre-Arnaud Raviart, Finite element methods for navier-stokes equations - theory and algorithms, Springer Series in Computational Mathematics, vol. 5, Springer, 1986.

[GR94] Christian Großmann and Hans-Görg Roos, Numerik partieller differentialgleichungen, Vieweg+Teubner Verlag, 1994.

[GSB+66] C.F. Gauss, E. Schering, M. Brendel, L. Schlesinger, W.F. Kaestner, B.G. Teubner, J. Springer, R. Dedekind, F.A. Perthes, and Gesellschaft der Wissenschaften zu Göttingen, Carl friedrich gauss werke: Bd. analysis (various texts, in latin and german, orig. publ. between 1799-1851, or found in the "nachlass"; annotated by e.j. schering). 1866 [i.e. 1868, Carl Friedrich Gauss Werke, Gedruckt in der Dieterichschen Universitäts-Druckerei W. Fr. Kaestner, 1866.

[GSW09] Ben-Yu Guo, Jie Shen, and Li-Lian Wang, Generalized Jacobi polynomials/functions and their applications, Appl. Numer. Math. **59** (2009), no. 5, 1011–1028. MR 2495135

[Guy65] Robert J. Guyan, Reduction of stiffness and mass matrices, AIAA Journal **3** (1965), no. 2, 380–380.

[HPB19] Tim Haubold, Veronika Pillwein, and Sven Beuchler, Symbolic evaluation of hp-fem element matrices, PAMM **19** (2019), e201900446.

[HPB21] Tim Haubold, Veronika Pillwein, and Sven Beuchler, Recursion relations for hp-fem element matrices on quadrilaterals, PAMM **21** (2021), no. 1, e202100200.

[HPB22] Tim Haubold, Veronika Pillwein, and Sven Beuchler, Recurrences for quadrilateral high-order finite elements, Math. Comput. Sci. **16** (2022), no. 4, Paper No. 32, 17. MR 4522812

[HR11] Waldemar Hebisch and Martin Rubey, Extended rate, more GFUN, Journal of Symbolic Computation **46** (2011), no. 8, 889–903.

[Hum26] Pierre Humbert, Fonctions de Lamé et fonctions de Mathieu, Mémorial des sciences mathématiques, no. 10, Gauthier-Villars, 1926 (fr).

[HW08] Jan S. Hesthaven and Tim Warburton, Nodal discontinuous galerkin methods, Springer New York, 2008.

[Hyl62] Egil A. Hylleraas, Linearization of products of jacobi polynomials., MATHEMATICA SCANDINAVICA **10** (1962), 189–200.

[Inc] Wolfram Research, Inc., Mathematica, Version 13.1, Champaign, IL, 2022.

[Kau05] M. Kauers, Algorithms for Nonlinear Higher Order Difference Equations, Diss, RISC-Linz, Johannes Kepler University, Linz, 10 2005.

[Kau06] ———, SumCracker: A package for manipulating symbolic sums and related objects, Journal of Symbolic Computation **41** (2006), no. 9, 1039–1057.

[Kau09] ———, Guessing handbook, Tech. Report 09-07, Research Institute for Symbolic Computation (RISC), 2009.

[Kau13] ———, The Holonomic Toolkit, pp. 119–144, Springer Vienna, Vienna, 2013.

[Kir17] Robert C. Kirby, Fast inversion of the simplicial Bernstein mass matrix, Numer. Math. **135** (2017), no. 1, 73–95. MR 3592107

[KLS10] Roelof Koekoek, Peter A. Lesky, and René F. Swarttouw, Hypergeometric orthogonal polynomials and their q-analogues, Springer Berlin Heidelberg, 2010.

[KM03] B. N. Khoromskij and J. M. Melenk, Boundary concentrated finite element methods, SIAM J. Numer. Anal. **41** (2003), no. 1, 1–36. MR 1974490

[Koo75] Tom Koornwinder, Two-variable analogues of the classical orthogonal polynomials, Theory and application of special functions (Proc. Advanced Sem., Math. Res. Center, Univ. Wisconsin, Madison, Wis., 1975), 1975, pp. 435–495. Math. Res. Center, Univ. Wisconsin, Publ. No. 35. MR 0402146

[Koo78] ———, Positivity proofs for linearization and connection coefficients of orthogonal polynomials satisfying an addition formula, Journal of the London Mathematical Society **s2-18** (1978), no. 1, 101–114.

[Kou10] Christoph Koutschan, A fast approach to creative telescoping, Math. Comput. Sci. **4** (2010), no. 2-3, 259–266. MR 2775992

[KR12] Yong Sup Kim and Arjun Kumar Rathie, A new proof of Saalschütz's theorem for the series ${}_3F_2(1)$ and its contiguous results with applications, Commun. Korean Math. Soc. **27** (2012), no. 1, 129–135. MR 2919018

[Kra01] C. Krattenthaler, Advanced determinant calculus, The Andrews Festschrift, Springer Berlin Heidelberg, 2001, pp. 349–426.

[KS13] George Em Karniadakis and Spencer J. Sherwin, Spectral/hp element methods for CFD, second ed., Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2013. MR 3088105

[KSA14] Pavel Kus, Pavel Solin, and David Andrs, Arbitrary-level hanging nodes for adaptive hp-fem approximations in 3d, Journal of Computational and Applied Mathematics **270** (2014), 121–133, Fourth International Conference on Finite Element Methods in Engineering and Sciences (FEMTEC 2013).

[KWB23] Sebastian Kinnewig, Thomas Wick, and Sven Beuchler, Algorithmic realization of the solution to the sign conflict problem for hanging nodes on hp-hexahedral nédélec elements, 2023.

[LN23] Zexin Liu and Akil Narayan, A stieltjes algorithm for generating multivariate orthogonal polynomials, SIAM Journal on Scientific Computing **45** (2023), no. 3, A1125–A1147.

[Lor86] G. G. Lorentz, Bernstein polynomials, second ed., Chelsea Publishing Co., New York, 1986. MR 864976

[LW06] Stanisław Lewanowicz and Paweł Woźny, Connections between two-variable bernstein and jacobi polynomials on the triangle, Journal of Computational and Applied Mathematics **197** (2006), no. 2, 520–533.

[Mal09] Stéphane Mallat, A wavelet tour of signal processing, Elsevier, 2009.

[Map] Maplesoft, a division of Waterloo Maple Inc.., Maple, Waterloo, Ontario.

[Mel02] J.M. Melenk, Hp-finite element methods for singular perturbations, Hp-finite Element Methods for Singular Perturbations, no. Nr. 1796, Springer, 2002.

[mfe] MFEM: Modular finite element methods [Software], `mfem.org`.

[MGS99] J.M. Melenk, K Gerdes, and C Schwab, Fully discrete hp-finite elements: Fast quadrature, Computer Methods in Applied Mechanics and Engineering **190** (1999), 4339–4364.

[MM14] William F. Mitchell and Marjorie A. McClain, A comparison of hp-adaptive strategies for elliptic partial differential equations, ACM Trans. Math. Softw. **41** (2014), no. 1, 1–39.

[MMP88] Yvon Maday, Catherine Mavriplis, and Anthony T. Patera, Nonconforming mortar element methods: Application to spectral discretizations, 1988.

[Mon03] P. Monk, Finite element methods for maxwell's equations, Numerical Mathematics and Scie, Clarendon Press, 2003.

[MOS66] W. Magnus, F. Oberhettinger, and R.P. Soni, Formulas and theorems for the special functions of mathematical physics, Die Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen, Springer-Verlag, 1966.

[MP96] Jean-François Maitre and Olivier Pourquier, Condition number and diagonal preconditioning: comparison of the p-version and the spectral element methods, Numer. Math. **74** (1996), no. 1, 69–84. MR 1400216

[Mur92] G. Mur, The finite-element modeling of three-dimensional time-domain electromagnetic fields in strongly inhomogeneous media, IEEE Transactions on Magnetics **28** (1992), no. 2, 1130–1133.

[MW01] J. M. Melenk and B. I. Wohlmuth, On residual-based a posteriori error estimation in *hp*-FEM, Adv. Comput. Math. **15** (2001), no. 1-4, 311–331 (2002), A posteriori error estimation and adaptive computational methods. MR 1887738

[MX15] Francisco Marcellán and Yuan Xu, On sobolev orthogonal polynomials, Expositiones Mathematicae **33** (2015), no. 3, 308–352.

[NP12] Nilima Nigam and Joel Phillips, High-order conforming finite elements on pyramids, IMA Journal of Numerical Analysis **32** (2012), no. 2, 448–483.

[Né80] J.-C. Nédélec, Mixed finite elements in $\mathbf{R}^3$, Numer. Math. **35** (1980), no. 3, 315–341. MR 592160

[Né86] ______, A new family of mixed finite elements in $\mathbf{R}^3$, Numer. Math. **50** (1986), no. 1, 57–81. MR 864305

[Ors80] Steven A. Orszag, Spectral methods for problems in complex geometries, J. Comput. Phys. **37** (1980), no. 1, 70–92. MR 584322

[PE12] Daniele Antonio Di Pietro and Alexandre Ern, Mathematical aspects of discontinuous galerkin methods, Springer Berlin Heidelberg, 2012.

[PPSS06] V. Pillwein, Peter Paule, Carsten Schneider, and Joachim Schöberl, Hypergeometric Summation Techniques for High Order Finite Elements, PAMM **6** (2006), 689–690.

[Pro57] Joseph Proriol, Sur une famille de polynomes à deux variables orthogonaux dans un triangle, C. R. Acad. Sci. Paris **245** (1957), 2459–2461. MR 95994

[Rah81] Mizan Rahman, A non-negative representation of the linearization coefficients of the product of jacobi polynomials, Canadian Journal of Mathematics **33** (1981), no. 4, 915–928.

[Rai71] Earl D. Rainville, Special functions, first ed., Chelsea Publishing Co., Bronx, N.Y., 1971. MR 0393590

[RR11] Medhat A. Rakha and Arjun K. Rathie, Extensions of Euler type II transformation and Saalschütz's theorem, Bull. Korean Math. Soc. **48** (2011), no. 1, 151–156. MR 2778504

[RT77] P.-A. Raviart and J. M. Thomas, A mixed finite element method for 2nd order elliptic problems, Mathematical aspects of finite element methods (Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975), Lecture Notes in Math., Vol. 606, Springer, Berlin, 1977, pp. 292–315. MR 0483555

[Rud87] W. Rudin, Real and complex analysis, Mathematics series, McGraw-Hill, 1987.

[SA96] B.A. Szabo and R.L. Actis, Finite element analysis in professional practice, Computer Methods in Applied Mechanics and Engineering **133** (1996), no. 3-4, 209–228.

[Sab93] Claude Sabbah, Systemes holonomes d'équations aux g-différences, D-Modules and Microlocal Geometry, DE GRUYTER, 1993.

[SB91] B. Szabó and I. Babuška, Finite element analysis, A Wiley-Interscience Publication, John Wiley & Sons, Inc., New York, 1991. MR 1164869

[Sch98] Ch. Schwab, *p*- and *hp*-finite element methods, Numerical Mathematics and Scientific Computation, The Clarendon Press, Oxford University Press, New York, 1998, Theory and applications in solid and fluid mechanics. MR 1695813

[Sch14] Joachim Schöberl, C++11 implementation of finite elements in ngsolve, 09 2014.

[SDD10] Pavel Solin, Lenka Dubcova, and Ivo Dolezel, Adaptive *hp*-FEM with arbitrary-level hanging nodes for Maxwell's equations, Adv. Appl. Math. Mech. **2** (2010), no. 4, 518–532. MR 2658044

[SK85] H. M. Srivastava and Per W. Karlsson, Multiple Gaussian hypergeometric series, Ellis Horwood Series: Mathematics and its Applications, Ellis Horwood Ltd., Chichester, 1985. MR 834385 (87f:33015)

[SK95] Spencer J. Sherwin and George Em. Karniadakis, A new triangular and tetrahedral basis for high-order (*hp*) finite element methods, Internat. J. Numer. Methods Engrg. **38** (1995), no. 22, 3775–3802. MR 1362003

[Sla66] Lucy Joan Slater, Generalized hypergeometric functions, Cambridge University Press, Cambridge, 1966. MR 0201688

[SM78] B. A. Szabo and A. K. Mehta, p-convergent finite element approximations in fracture mechanics, International Journal for Numerical Methods in Engineering **12** (1978), no. 3, 551–560.

[SO20] Ben Snowball and Sheehan Olver, Sparse spectral and *p*-finite element methods for partial differential equations on disk slices and trapeziums, Stud. Appl. Math. **145** (2020), no. 1, 3–35. MR 4130677

[SS18] Dominik Schötzau and Christoph Schwab, Exponential convergence of *hp*-FEM for elliptic problems in polyhedra: mixed boundary conditions and anisotropic polynomial degrees, Found. Comput. Math. **18** (2018), no. 3, 595–660. MR 3807357

[S+YY] W. A. Stein et al., Sage Mathematics Software (Version x.y.z), The Sage Development Team, YYYY, `http://www.sagemath.org`.

[SZ94] Bruno Salvy and Paul Zimmermann, Gfun: A maple package for the manipulation of generating and holonomic functions in one variable, ACM Trans. Math. Softw. **20** (1994), no. 2, 163–177.

[Sze67] Gábor Szegő, Orthogonal polynomials, third ed., American Mathematical Society, Providence, R.I., 1967, American Mathematical Society Colloquium Publications, Vol. 23. MR 0310533

[Tch14] Daniel Duviol Tcheutia, On connection, linearization and duplication coefficients of classical orthogonal polynomials, Ph.D. thesis, Universität, Kassel, Fachbereich Mathematik und Naturwissenschaften, 07 2014.

[vSD04] Pavel Šolín, Karel Segeth, and Ivo Doležel, Higher-order finite element methods, Studies in Advanced Mathematics, Chapman & Hall/CRC, Boca Raton, FL, 2004, With 1 CD-ROM (Windows, Macintosh, UNIX and LINUX). MR 2000261

[VSK10] Peter E. J. Vos, Spencer J. Sherwin, and Robert M. Kirby, From h to p efficiently: implementing finite and spectral/hp element methods to achieve optimal performance for low- and high-order discretisations, J. Comput. Phys. **229** (2010), no. 13, 5161–5181. MR 2643647

[VŠZ07] Tomáš Vejchodský, Pavel Šolín, and Martin Zítka, Modular hp-FEM system HERMES and its application to maxwell's equations, Mathematics and Computers in Simulation **76** (2007), no. 1-3, 223–228.

[WF] Wolfram research, `https://functions.wolfram.com/`.

[Whi57] Hassler Whitney, Geometric integration theory, Princeton University Press, 1957.

[Wil78] James Arthur Wilson, HYPERGEOMETRIC SERIES RECURRENCE RELATIONS AND SOME NEW ORTHOGONAL FUNCTIONS, ProQuest LLC, Ann Arbor, MI, 1978, Thesis (Ph.D.)–The University of Wisconsin - Madison. MR 2628179

[WK01] Barbara I. Wohlmuth and Rolf H. Krause, A multigrid method based on the unconstrained product space for mortar finite element discretizations, SIAM Journal on Numerical Analysis **39** (2001), no. 1, 192–213.

[Woh01] Barbara I. Wohlmuth, A mortar finite element method using dual spaces for the lagrange multiplier, SIAM Journal on Numerical Analysis **38** (2001), no. 3, 989–1012.

[WW98] Christian Wieners and Barbara I. Wohlmuth, The coupling of mixed and conforming finite element discretizations, Domain decomposition methods, 10 (Boulder, CO, 1997), Contemp. Math., vol. 218, Amer. Math. Soc., Providence, RI, 1998, pp. 547–554. MR 1649655

[WW16] D. Wachsmuth and J.-E. Wurst, Optimal control of interface problems with hp-finite elements, Numerical Functional Analysis and Optimization **37** (2016), no. 3, 363–390.

[WZ90] Herbert S. Wilf and Doron Zeilberger, Rational functions certify combinatorial identities, Journal of the American Mathematical Society **3** (1990), no. 1, 147–158.

[Xu17] Yuan Xu, Approximation and orthogonality in Sobolev spaces on a triangle, Constr. Approx. **46** (2017), no. 2, 349–434. MR 3691233

[Yos12] Zohar Yosibash, Singularities in elliptic boundary value problems and elasticity and their connection with failure initiation, Interdisciplinary Applied Mathematics, vol. 37, Springer, New York, 2012. MR 2867994

[Zag06] Sabine Zaglmayr, High order finite elements for electromagnetic field computation, Ph.D. thesis, Johannes Kepler University Linz, 07 2006.

[Zei90a] Doron Zeilberger, A holonomic systems approach to special functions identities, Journal of Computational and Applied Mathematics **32** (1990), no. 3, 321–368.

[Zei90b] ———, A holonomic systems approach to special functions identities, J. Comput. Appl. Math. **32** (1990), no. 3, 321–368.

Curriculum Vitae

Scientific vita

2010 - 2017 Study of Technomathematics at the University of Bremen, Bachelor and Master of Science.
Master thesis: *Hydrostatic reconstructed Force scheme and application to the non-unique Riemann solutions for the shallow water equations*. Supervisor: Prof. Dr. Alfred Schmidt and Dr. Ee Han.

2017 - 2023 Doctoral and scientific researcher at the Institute for Applied Mathematics, Leibniz University Hannover. Supervisor: Prof. Dr. Sven Beuchler.

Publications

1. Tim Haubold, Veronika Pillwein, and Sven Beuchler, Symbolic evaluation of hp-fem element matrices, PAMM **19** (2019), e201900446
2. Tim Haubold, Veronika Pillwein, and Sven Beuchler, Recursion relations for hp-fem element matrices on quadrilaterals, PAMM **21** (2021), no. 1, e202100200
3. Tim Haubold, Veronika Pillwein, and Sven Beuchler, Recurrences for quadrilateral high-order finite elements, Math. Comput. Sci. **16** (2022), no. 4, Paper No. 32, 17. MR 4522812
4. Sven Beuchler, Tim Haubold, and Veronika Pillwein, Recursion formulas for integrated products of jacobi polynomials, Constructive Approximation (2023)