Condition Numbers of Bases in The Finite Element Exterior Calculus

by

Sveinn Sandvik Svendsen

Master thesis for the degree of
Master of Science
in Applied mathematics, mechanics and numerical physics
(Computational science)

Department of Mathematics
Faculty of Mathematics and Natural Sciences
University of Oslo

February 2010

Acknowledgements

First of all I would like to express my thanks to my supervisor, Ragnar Winther. Your immediate appeal and open attitude towards a young man who wants to be a researcher is invaluable. Thank you for being forever calm, helpful and available, and for giving me nudges in the right direction when I was stuck. I couldn't have wished for a wiser or more confident supervisor.

I would also like to direct gratitude towards all the other mathematics teachers up through the years, especially Ole Kristian Eivindsson and Tom Lindstrøm for demonstrating without a doubt, that mathematics can be and is fun.

I would like to thank all my friends for their love and friendship, and for giving me joy in my everyday. To all my board gaming friends (I am after all a very promiscuous board gamer), thank you for providing me with recreation and casual mental exercise. Without it would surely have gone mad. To all people at B1002: Thank you for giving me a home away from home. My motivation to go to my reading room at Blindern has persisted because of the wonderful and interesting people there.

I would like to thank the creators of all tools that I've used for researching and writing my thesis. LyX, the LaTeX-WYSIWYG editor, whose program enabled me to quickly discover new functionality when needing it, instead of spending time browsing the web for it or creating it myself, leaving me even more time to focus on the subject matter, and also for letting me see the formulas I'm writing as I write them.

Thank you to all the student organisations at the University of Oslo. They all contribute towards a better and more interesting university. A special thank you to all of my elected representatives who are working for the students: The University of Oslo is a better place to study because of your efforts.

Thanks to Steffen Sjursen and Kjetil Matias Holte for reading through and commenting on this thesis in its final stages, and thanks to all my fellow students who've listened to me rant about differential forms and barycentric coordinates and inspired me with their comments.

Most of all, I would like to thank all my family, for providing me with love and caring in good times and bad times. They have supported me in countless ways , and for that I am truly grateful.

Contents

List of Tables

Nomenclature

$\mathbb{A}$ $= \{\mathbb{A}_{ij}\}_{i,j=1}^{N} := \{a(\phi_i, \phi_j)\}_{i,j=1}^{N}$ is the *stiffness matrix* of a over ϕ_{ii}, page 14

a.e. almost everywhere (outside a set of measure 0), page 12

$\mathrm{Alt}^k(W)$ the space of alternating k-forms over W, page 33

B_j^T $= \binom{r}{j} \left(\lambda^T\right)^j$ is the Bernstein basis, page 23

$\mathbb{C}^{n,n}$ is the set of all complex-valued $n \times n$ matrices., page 25

$\mathrm{cond}(A)$ $= \|A\| \cdot \left\|A^{-1}\right\|$ the condition number of a matrix $A \in \mathbb{C}^{n,n}$, page 25

$\mathrm{cond}(a)$ the condition number of the bilinear form a, page 26

d is the exterior derivative $\mathrm{d}u = \sum_{i=1}^{n} \frac{\partial}{\partial x_i} u \wedge \mathrm{d}x_i$, page 37

$\Delta_k(\mathcal{T})$ the collection of all the k-subsimplices of $\mathcal{T}$, page 17

$\Delta_k(T)$ the collection of k-subsimplices of T, page 17

$\Delta(T)$ the *collection of all subsimplices of* T, page 17

$\mathrm{d}\lambda_i^T$ are the *barycentric alternating forms* related to the simplex T, page 35

$\mathrm{d}\lambda_i$ are the barycentric differential forms of the reference simplex (i.e. $\mathrm{d}\lambda_i^{T_0}$), page 35

$\mathrm{d}x_\sigma$ the orthonormal basis of alternating k-forms over $\mathbb{R}^n$, page 34

e_σ $= (e_{\sigma(j)}, \ldots, e_{\sigma(k)})$ a sequence of orthonormal vectors in $\mathbb{R}^n$, page 35

F Affine transformation $F : \mathbb{R}^n \to \mathbb{R}^n$, page 23

F Affine transformation $F : \mathbb{R}^n \to \mathbb{R}^n$, page 42

FEM The Finite Element Method, page 11

G is the Gram matrix $G\left(\{\phi_i\}_{i=1}^{N}\right) := \left\{(\phi_i, \phi_j)_V\right\}_{i,j=1}^{N}$, page 26

h $= \max_{T \in \mathcal{T}} h_T$ where $h_T := \frac{\mathrm{diam}(T)}{2}$, page 17

$H^l(\Omega)$ the space of l-times-differentiable functions in $L^2(\Omega)$. , page 12

$H\Lambda^k(\Omega)$ the space of d-differentiable differential k-forms, page 37

$[\![j]\!]$ $:= \{\, i \in \mathbb{N}_0 |\ j_i \neq 0\}$ is the support of j, page 17

L^* the pullback of L, page 42

$L_2(\Omega)$ $= \left\{f : \Omega \to \mathbb{R} \,\middle|\, \int_\Omega f^2 \mathrm{d}x < \infty\right\}$, page 11

L_* the push-forward of L, page 42

$L^2\Lambda^k(\Omega)$ the L^2-space of differential k-forms on Ω, page 36

$\Lambda^k(\Omega)$ the space of differential k-forms on Ω, page 36

λ_i^T the *barycentric coordinates* of a simplex T, page 22

λ_i are the barycentric coordinates of T_0, page 22

$[\![\sigma]\!]$ the image of σ, page 17

$\mathsf{lsum}(i, j, n)$ the `lsum` function, page 29

$\{M_\alpha\}_\alpha$ a matrix indexed by α, page 49

$\mathbb{N}_0^{m:n}$ The set of multiindices $(j_m, \ldots, j_n)$, $j_i \in \mathbb{N}_0$, page 17

$\mathcal{N}_\mathcal{T}$ the degrees of freedom of $\mathcal{T}$, page 19

$\mathcal{N}_T$ the degrees of freedom associated with a simplex T, page 19

$\omega_f(\mathcal{T})$ all T that share f as a subsimplex., page 17

$\{\phi_i\}_{i=1}^N$ usually a generic basis for V_h, page 14
ϕ_σ the Whitney forms, page 41
$\mathcal{P}_r^-\Lambda^k(T)$ the space of intermediary polynomials over T, page 39
$\dim \mathcal{P}_r\Lambda^k(T)$ the space of polynomial differential k-forms over the domain T, page 39
$\mathcal{P}_r(T)$ the space of rth degree polynomials over T, page 18
$C\mathcal{P}_r(\mathcal{T})$ continuous piecewise polynomials over $\mathcal{T}$, page 18
S_k is the group of all permutations on k elements., page 33
$\Sigma[i:k; j:n]$ Increasing indices $\sigma : \{i, \ldots, k\} \to \{j, \ldots, n\}$, page 16
T the convex hull between $n+1$ points $\{p_i \in \mathbb{R}^n\}_{i=0}^n$, page 17
T_0 $= [0, e_1, \ldots, e_n]$ is called the reference simplex., page 17
$\mathcal{T}$ simplicial mesh, a collection of adjacent simplices T_i in Ω, page 17
$\mathrm{Tr}_\Gamma u$ function shows the limit of u towards $\partial\Gamma$, page 12
V space where the variational problem (2.3.1) is defined, page 14
V_h finite-dimensional subspace of V, page 14
v_σ $= (v_{\sigma(j)}, \ldots, v_{\sigma(k)})$ is a $[\![\sigma]\!]$-tuple of vectors., page 35
$\wedge$ Wedge product of alternating forms, page 34

CHAPTER 1

Introduction

1.1. Motivation

The Finite Element Method (FEM) has its motivation from solving partial differential equations (PDEs) with large accuracy. We often have the necessity to solve PDEs which can't be solved analytically, i.e. when its domain (the space on which the PDE is solved) is of irregular shape . The FEM offers a discretization technique that is analytically appealing, and at the same time it is time efficient and carries good precition estimates. The wonder of the FEM is that it generates a computer-solvable problem which is analogous in formulation to its theoretical counterpart.

Its generality can be described amongst other examples by the program pack FeNICS[1] or the calculation tool *Fluent*. These libraries contain tools for dividing domains into *polyhedral meshes* and constructing the relevant equations for calculating approximate solutions to PDEs over the domains. In this thesis we will only focus on the use of *simplicial meshes* because of their friendliness towards our formulation, even though *hypercubical meshes* might seem more intuitive and esthetically appealing to some. This thesis will not argue against them, but we will see that simplices are quite sufficient to develop our theory.

Necessary for understanding the FEM is a limited knowledge of partial differential equations, functional analysis (and consequently linear algebra), because the method's grounding in theory, some of which we will repeat here.

The reason that the Finite Element Exterior Calculus (FEEC) is so interesting is because it generalizes the notion of affine equivalence so it's not only valid for H^1-spaces, but also for $H(\mathrm{div})$- and $H(\mathrm{curl})$-spaces. Affine equivalence is an important tool in most finite element computations, as it increases the efficiency of the calculations by a huge factor.

The reason for developing the FEEC is that the FEM is considered slow but precise, so many people doing simulations with limited computing methods often use FEM only on parts of their domain Ω. They then leave the rest of the domain to some time-efficient method with more constraints or assumptions, for instance a Finite Difference Method. The view of FEM as time-consuming is supported by its slow calculation time (especially when working on $H(\mathrm{div})$- and $H(\mathrm{curl})$-spaces).

The FEEC was first summarised in the survey article by Arnold et.al. [**2**], drawing upon many works to give a complete framework for treating PDE of differential forms. Some articles [**3**] have been published on this subject, but so far this is a fairly new area.

[1] www.fenics.org

1.2. What's in this thesis?

In Chapter 2 we introduce the FEM and what kind of problems we focus on. We develop a framework for approximating solutions to certain types of variational problems over the function space V. In our case $V = H^1(\Omega)$ of functions over Ω whose derivatives are square integrable. We then limit the space V to a finite subspace V_h which in our case is the space $C\mathcal{P}_r(\mathcal{T})$ of piecewise polynomial and continuous functions over the simplicial meshing $\mathcal{T}$ of Ω. In Chapter 3 we try to find which basis which gives the highest level of accuracy when solving the limited variational problem. The goal is numerical stability and accuracy when approximating the variational problems with an element in V_h.

In Chapter 4 we introduce the FEEC (from [**2**]), a generalisation from our scalar variational problems in Chapter 2, using differential (k-)forms. We thus define a new version of V, $H\Lambda^k(\Omega)$ containing the k-forms whose exterior derivative has only components in $L^2(\Omega)$. We expand our study of variational problems to this space. The motivation behind these is again an efficient and numerically accurate and stable framework for PDEs formulated with antisymmetric tensors.

In Chapter 5 we compare different bases for our two new versions of V_h. One of them is $H\mathcal{P}_r\Lambda^k(\mathcal{T})$, the space of piecewise polynomial differential k-forms up to degree r which are in $H\Lambda^k(\Omega)$. The other is $H\mathcal{P}_r^-\Lambda^k(\mathcal{T})$ (originally introduced in [**1**]), a subspace of the former where some of the homogeneous rth-degree polynomials are removed, to ensure that we cover all kinds of PDE whose solutions exist in $H\Lambda^k(\Omega)$. We compare bases for these and see which provides us with the greatest numerical accuracy when approximating variational problems on V_h.

This thesis' main focus is the conditioning of the obtained discrete systems from Chapters 2 and 4, in other words we will consider the conditioning of the stiffness matrix relative to different bases for V_h. However, we don't approach it directly, but prove that it can be limited by the condition number of the Gram matrix of each individual basis (sometimes called the weight matrix). In Chapters 3 and 5 we study how to calculate the elements in each Gram matrix, so that we might estimate its condition numbers by computer calculations. These results for the space $H\Lambda^k(\Omega)$ are the main goal of this thesis, and for the impatient they can be found in Tables 1 on page 52 to 10 on page 54.

CHAPTER 2

The Finite Element Method

In this section we describe the Finite Element Method (FEM).[1] The FEM is a Galerkin method (explained in 2.3) for approximating solutions to Partial Differential Equations (PDE) and integral equations with the aid of piecewise smooth functions on polyhedral meshes, using the tools of functional analysis.[2] This thesis focuses on the PDE side of the FEM, and every time we say "FEM" there is no intended reference to solving integral equations.

In Section 2.1 we detail what spaces we are working with, and in Section 2.2 what kind of problem we want to solve and how we formulate it. In Section 2.3 we give a quick overview of Galerkin's method (a class of methods for solving our problem) and what motivates us in using it. Section 2.4 explains which of these problems we choose and introduces the bases which are the object of study in this thesis.

2.1. Weak derivatives, function spaces

Before we can consider our method, we need to define the concepts of weak derivatives and Sobolev spaces. (We will assume some knowledge about topology, measure theory and functional analysis – **[14, 15, 5]** are good sources.) In our case, we will be working in a subspace of the $L_2(\Omega)$ Hilbert space, the space of square integrable functions over $\Omega \subset\subset \mathbb{R}^n$,[3]

$$L_2(\Omega) := \left\{ f : \Omega \to \mathbb{R} \,\middle|\, \int_\Omega f^2 \mathrm{d}x < \infty \right\}.$$

This is a normed vector space of functions with the norm and inner product

$$\|u\|_{L_2(\Omega)} := \left(\int_\Omega (u)^2 \, \mathrm{d}x \right)^{\frac{1}{2}}, \quad (u, v) := \int_\Omega uv \mathrm{d}x.$$

When the integral is over another domain $\Gamma \subseteq \Omega$ will write

$$(u, v)_{L^2(\Gamma)} := \int_\Gamma uv \mathrm{d}x.$$

Since we will be working with PDE, we must also be able to differentiate our functions, and we must restrict the space $L_2(\Omega)$ of integrable functions to the subspace $H^1(\Omega)$ which has weak (i.e. integrable) derivatives:

---

[1] Main sources Finite Element Method: **[6, 7]**

[2] Main source for Partial Differential Equations: **[10]**; Main sources for functional analysis: **[15, 9, 12]**

[3] $\subset\subset$ means that Ω is compactly embedded in $\mathbb{R}^n$

DEFINITION 2.1. Taking $C_0^1(\Omega)$ to be the set of all once-differentiable continuous functions v over Ω with $v|_{\partial\Omega} = 0$, A *weak partial derivative of* u is a function $\xi_i \in L_2(\Omega)$ such that

$$\forall v \in C_0^1(\Omega) : (\xi_i, v) = -\left(u, \frac{\partial v}{\partial x_i}\right) \tag{2.1.1}$$

or written in integral form,

$$\forall v \in C_0^1(\Omega) : \int_\Omega \xi_i v \mathrm{d}x = -\int_\Omega u \frac{\partial v}{\partial x_i} \mathrm{d}x.$$

In other words, we require that integration by parts will work on $u\frac{\partial v}{\partial x_i}$ and give $\xi_i v$ and vice-versa. We will write ξ_i as $\frac{\partial u}{\partial x_i}$ or $\frac{\partial}{\partial x_i}(u)$, but be aware that this function is only unique almost everywhere (also written *a.e.*, this means outside a set of measure 0). The *space of once-differentiable functions* is

$$H^1(\Omega) := \left\{ f \in L_2(\Omega) \,\middle|\, \forall i \leq n, \exists \frac{\partial f}{\partial x_i} \in L_2(\Omega) \right\},$$

which is an example of a *Sobolev space*. More general Sobolev spaces are $H^l(\Omega)$ of l times-differentiable functions. Since the FEM uses piecewise continuous functions inside this space, we will see what restrictions being in $H^1(\Omega)$ imposes on such functions. But first we will define piecewise continuous functions:

DEFINITION 2.2. Assume we have a domain $\Omega = \bigcup_i \Omega_i$ which is a union of disjoint (compact) sets. Then, a *piecewise continuous function* u has the properties that $u|_{\Omega_1} = u_1$ and $u|_{\Omega_2} = u_2$ are continuous functions, $u_i \in C(\Omega_i)$.

Note that $C(\Omega_i) \subset H^1(\Omega_i)$ since Ω_i is compact.

We are going to work with piecewise continuous functions in this space, and since they are in $H^1(\Omega)$ they have this property:

THEOREM 2.3. *Let Ω be a domain that can be partitioned into the disjoint domains Ω_1 and Ω_2 whose boundaries are C^1 a.e.. Any piecewise continuous function in $H^1(\Omega)$ is continuous.*

PROOF. Let $u \in H^1(\Omega)$ be continuous on Ω_1 and Ω_2. Assuming $v \in C_0^1(\Omega)$ (C^1-functions that are 0 on $\partial\Omega$):

$$\int_\Omega u \frac{\partial v}{\partial x_i} \mathrm{d}x = \int_{\Omega_1} u \frac{\partial v}{\partial x_i} \mathrm{d}x + \int_{\Omega_2} u \frac{\partial v}{\partial x_i} \mathrm{d}x.$$

Here we do an integration by parts (where $\mathrm{Tr}_\Omega(v)(x) = v(x)$ on $\partial\Omega_1 \cup \partial\Omega_2$ because it is continuous) and get

$$= -\int_{\Omega_1} \frac{\partial u}{\partial x_i} v \mathrm{d}x + \int_{\partial\Omega_1} \mathrm{Tr}_{\Omega_1}(u) v n_i^{\Omega_1} \mathrm{d}x - \int_{\Omega_2} \frac{\partial u}{\partial x_i} v \mathrm{d}x + \int_{\partial\Omega_2} \mathrm{Tr}_{\Omega_2}(u) v n_i^{\Omega_2} \mathrm{d}x$$

where $n_i^{\Omega_1}$ is the ith component of the unit normal on $\partial\Omega_1$. The $\mathrm{Tr}_\Gamma u$ (Trace) function shows the limit towards $\partial\Gamma$ (see [**10**]). Let $B = \partial\Omega_1 \cap \partial\Omega_2$. Since $v = 0$ on $\partial\Omega$ and $n_i^{\Omega_1} = -n_i^{\Omega_2}$ on B,

$$\int_\Omega u \frac{\partial v}{\partial x_i} \mathrm{d}x = -\int_\Omega \frac{\partial u}{\partial x_i} v \mathrm{d}x + \int_B v n_i^{\Omega_1} \left(\mathrm{Tr}_{\Omega_1}(u) - \mathrm{Tr}_{\Omega_2}(u)\right) \mathrm{d}x.$$

This does not coincide with our definition of the weak derivative in (2.1.1) unless $\mathrm{Tr}_{\Omega_1}(u) - \mathrm{Tr}_{\Omega_2}(u) = 0$. Since $u \in H^1(\Omega)$ was chosen arbitrarily, we can draw

the conclusion that a piecewise continuous function in $H^1(\Omega)$ must be continuous everywhere on Ω. □

2.2. Variational problems

Variational problems are an abstract way of interpreting possible measurements or states of a system as vectors. A variational problem (sometimes referred to as "weak" or "integral" formulation in certain applications) is usually formulated like this: Find $u \in V$ that satisfies

$$\forall v \in V : a(u,v) = l(v) \tag{2.2.1}$$

for scalar functions $a : V \times V \to \mathbb{R}$ and $l : V \to \mathbb{R}$. In our case this is a formulation where V is a Hilbert space with norm $\|\cdot\|_V$, $a : V \times V \to \mathbb{R}$ is a symmetric ($\forall u, v : a(u,v) = a(v,u)$), bounded ($\forall u, v : |a(u,v)| \leq \hat{C}\|u\|_V\|v\|_V$) and bilinear (linear in both arguments) form, and l is a linear bounded functional. In the case that $\forall v \in V : a(v,v) \geq C\|v\|_V^2$, a is also called *coercive* and according to The Lax-Milgram Lemma in [**12**, p. 57] (2.2.1) is has a unique solution. We will persist in using such a and l because of the certainty of a unique solution.

Example 2.4. A Weak formulation of a PDE

We let $V = H_0^1(\Omega)$ (functions that are 0 on $\partial\Omega$, integrable, once-differentiable), $l(v) := \int_\Omega fv\mathrm{d}x$ and $a(u,v) := \int_\Omega \left(\sum_{i,j=1}^n a_{ij}(x)\frac{\partial u}{\partial x_i}\frac{\partial v}{\partial x_j} + c(x)uv\right)\mathrm{d}x$ where $\forall x : a_{ij}(x)$ is symmetric and positive definite and $\forall x : c(x) \geq 0$. (2.2.1) becomes

$$\forall v \in H_0^1(\Omega) \int_\Omega \left(\sum_{i,j=1}^n a_{ij}(x)\frac{\partial u}{\partial x_i}\frac{\partial v}{\partial x_j} + c(x)uv\right)\mathrm{d}x = \int_\Omega fv\mathrm{d}x. \tag{2.2.2}$$

If we add the condition that u is twice differentiable (and $v = 0$ on $\partial\Omega$), this may be (through integration by parts)

$$\forall v \in H_0^1(\Omega) \int_\Omega \left(-\sum_{i,j=1}^n a_{ij}(x)\frac{\partial^2 u}{\partial x_i \partial x_j} + c(x)u\right)v\mathrm{d}x = \int_\Omega fv\mathrm{d}x$$

which corresponds to a strong formulation of the PDE (where one tries to find $u \in C^2(\Omega)$)

$$\begin{aligned} -\sum_{i,j=1}^n a_{ij}(x)\frac{\partial^2 u}{\partial x_i \partial x_j} + c(x)u &= f \text{ on } \Omega \\ u &= 0 \text{ on } \partial\Omega. \end{aligned} \tag{2.2.3}$$

This equation is classified as an *elliptic* in [**6**] (i.e.$\forall x : a_{ij}(x)$ is symmetric and positive definite). A more general case can be seen in [**6, 10**].

The weak formulation has the added benefit of looking at u, v and f (and some of their derivatives) as *integrable* instead of having to be continuous functions of x on Ω. The variational formulation clearly shows the possible application of the Lax-Milgram lemma [**12**, p. 57] which proves existence of a unique solution for certain elliptic PDE (including the example here). To prove this claim, we have to

show that a is linear, bounded, and coercive. The bilinear form a is clearly linear. It is bounded by the Cauchy-Schwartz inequality,

$$\int_\Omega \left(\sum_{i,j=1}^{n} a_{ij}(x) \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_j} \right) \mathrm{d}x \leq \max_{i,j} \left(|a_{ij}(x)| \right) \sum_{i,j=1}^{n} \int_\Omega \left(\left| \frac{\partial u}{\partial x_i} \right| \left| \frac{\partial v}{\partial x_j} \right| \right) \mathrm{d}x \tag{2.2.4}$$

$$\overset{\text{C–S}}{\leq} \max_{i,j} \left(|a_{ij}(x)| \right) \sqrt{\int_\Omega \left| \frac{\partial u}{\partial x_i} \right|^2 \mathrm{d}x} \sqrt{\int_\Omega \left| \frac{\partial v}{\partial x_j} \right|^2 \mathrm{d}x.} = Cn^2 \left\| Dv \right\| \left\| Du \right\|$$

According to the Poincaré inequality this is bounded by

$$\leq Cn^2 \left\| v \right\| \left\| u \right\|.$$

As for its coercivity, using the fact of a_{ij}'s positive definiteness ($\forall \xi \in \mathbb{R}^n : \sum_{ij} \xi_i a_{ij} \xi_j \geq \hat{C} \|\xi\|^2$, for some $\hat{C} > 0$) we have the following inequalities:

$$a(v,v) = \int_\Omega \left(\sum_{i,j=1}^{n} a_{ij}(x) \frac{\partial v}{\partial x_i} \frac{\partial v}{\partial x_j} + c(x) v^2 \right) \mathrm{d}x \geq \int_\Omega \left(\hat{C} \left| Dv \right|^2 + c(x) v^2 \right) \mathrm{d}x$$

$$\geq \hat{C} \int_\Omega \left(\left| Dv \right|^2 \right) \mathrm{d}x$$

whose square root is a norm on $H_0^1(\Omega)$ as a consequence of the Poincaré inequality. This proves the coercivity of a, and hence (2.2.2) has a unique solution according to the Lax-Milgram Lemma.

2.3. Galerkin's method for variational problems

Galerkin's method for approximating solutions to variational problems is used both for proofs and numerical approximation, the latter of which is our focus. The basics of (the generalised) Galerkin's method as used in this thesis are:

(1) Start out with a variational problem: Find $u \in V$ (V is a vector space)

$$\text{find } u \in V \text{ s.t. } \forall v \in V : a(u,v) = l(v), \tag{2.3.1}$$

for example (2.2.1) (V Hilbert space; a linear, bounded, coercive; l linear, bounded).

(2) Choose a finite-dimensional subspace V_h of the Hilbert space V.

(3) Restrict the problem in (1) to the subspace V_h,

$$\text{find } u \in V_h \text{ s.t. } \forall v \in V_h : a(u,v) = l(v) \tag{2.3.2}$$

and

(4) solve (2.3.2) (if possible). This might be done (as was Galerkin's proposal in [**11**]) by choosing a basis $\mathcal{F} = \{\phi_i\}_{i=1}^N$ for V_h, thus converting (2.3.2) to an equation system that turns it into

$$\text{find } U \in \mathbb{R}^N \text{ s.t. } \forall j \leq N : \sum_{i=1}^{N} U_i a(\phi_i, \phi_j) = l(\phi_j). \tag{2.3.3}$$

$\mathbb{A} := \{\mathbb{A}_{ij}\}_{i,j=1}^N := \{a(\phi_i, \phi_j)\}_{i,j=1}^N$ is called the *stiffness matrix of a over the basis* $\{\phi_i\}_{i=1}^N$.

It should be noted that Galerkin's method in its most general form (a nonlinear and asymmetric) doesn't necessarily converge to any solution of the variational problem, but in the following case it does: According to Céa's Lemma [**6**, p. 55], if a is linear, bounded and coercive and l is bounded and linear, u is the (unique) solution of (2.3.1) and u_h is the approximated solution in V_h, then Céa's inequality

$$\|u - u_h\|_V \leq C \inf_{v \in V_h} \|u - v\|_V \tag{2.3.4}$$

tells us that

(1) u_h is the element of V_h closest to u, and
(2) As V_h approaches V, u_h will approach u in V.

THEOREM 2.5. *A well-known fact is that the variational problem* (2.3.1) *is equivalent to the minimisation problem*

$$u := \min_{v \in V} M(v) \text{ where } M(v) := \frac{1}{2}a(v,v) - l(v). \tag{2.3.5}$$

PROOF. **The function u solves (2.3.1). $\Rightarrow$ The function u gives the minimum of (2.3.5):** Take a $u \in V$ that solves **(2.3.1)** . Then $\forall v \in V$

$$M(u+v) = \frac{1}{2}\left(a(u,u) + a(v,v)\right) + a(u,v) - l(v) - l(u).$$

Since a is positive definite/coercive and $a(u,v) = l(v)$,

$$M(u+v) = \frac{1}{2}\left(a(u,u) + a(v,v)\right) - l(u) \geq \frac{1}{2}a(u,u) - l(u) = M(u).$$

The function u gives a minimum of (2.3.5). $\Rightarrow$ The function u solves (2.3.1): Let u be the minimum of **(2.3.5)**, and let $\epsilon \in \mathbb{R}$. Take any $v \in V$, and define

$$\mu(\epsilon) := M(u + \epsilon v)$$

which by the linearity of a and l is

$$\mu(\epsilon) = \frac{1}{2}\left(a(u,u) + \epsilon^2 a(v,v)\right) + \epsilon a(u,v) - \epsilon l(v) - l(u).$$

Since μ is a real, continuous function (because of the linearity and boundedness of M), μ has at least a weak derivative

$$\mu'(\epsilon) = \epsilon a(v,v) + a(u,v) - l(v)$$

$\mu(0) \leq \mu(\epsilon) \forall \epsilon \in \mathbb{R}$, $\mu'(0) = 0$ and thus

$$a(u,v) - l(v) = 0.$$

Since $v \in V$ was arbitrary, this holds for all $v \in V$. □

2.4. Constructing V_h

Our goal is to solve a variational problem of the type in (2.2.1). Now we want to construct a numerical method that approximates the solution of Example 2.4. Our solutions will be in $H^1(\Omega)$, and we have to choose our subspace within it, and consequently the functions in the subspace will be continuous (by Theorem 2.3). Since our subspace is not uniquely defined yet we can add useful restrictions to have more control of its contents. What motivates further choice of subspace is that it

- Converges towards our solution when refining the parameters,

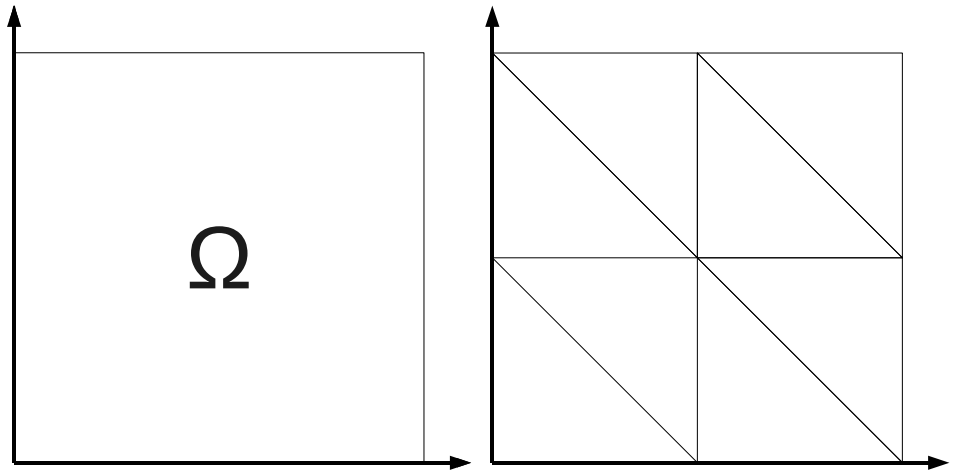

FIGURE 2.4.1. Our domain the unit square $\Omega = [0,1]^2$

- Generates a sparse stiffness matrix, and
- Uses a minimal amount of operations to do so.

In short we want to create a version of the problem that is well adapted to quick and precise solving on a computer. The rest of this section describes how the choice of subspace and its basis satisfies these motivations.

2.4.1. The Mesh.

Having some sort of refinement of the domain to include more data points is a usual method of increasing precision in computational science. We're going to divide our domain into mesh of subdomains with a piecewise C^1 border and a maximum diameter of $2h$. To begin with, we have something to assume about the regularity of the domain Ω: It has to have a piecewise C^1 border, meaning that its border can be covered by a $\gamma : [0,1] \to \mathbb{R}^n$ for $t \in [0,1]$, where $\gamma(0) = \gamma(1)$, and $\frac{\mathrm{d}\gamma}{\mathrm{d}t}(t)$ exists almost everywhere. We will then proceed to partition Ω into disjoint subdomains Ω_i, $\bigcup_i \Omega_i = \Omega$ with the same regularity property.

EXAMPLE 2.6. For instance we have the unit square which can be divided into triangles. In three dimensions we can have the unit cube divided into tetrahedra.

Since the domain Ω is a subset of $\mathbb{R}^n$, we do not only work in two or three dimensions. To help this, we can generalise these two- and three-dimensional triangles and tetrahedra to the n-dimensional notion of *simplices* (using increasing indices):

DEFINITION 2.7. Increasing indices

An *increasing index* is a $\sigma : \{i, \ldots, k\} \to \{j, \ldots, n\}$ which adheres to the following rule:

$$l < m \Rightarrow \sigma(l) < \sigma(m).$$

We will write it with the notation $\sigma \in \Sigma[i:k, j:n]$. The collection of all such increasing indices is written as Σ. The image of an increasing multiindex is written $[\![\sigma]\!]$.

DEFINITION 2.8. Simplices and simplicial meshes

A *simplex* T in n dimensions (an *n-simplex*) is the convex hull between $n+1$ different points $\{p_i \in \mathbb{R}^n\}_{i=0}^n$. Notation: $T = [p_0, \ldots, p_n]$. $T_0 = [0, e_1, \ldots, e_n]$ is called the reference simplex.

A *k-subsimplex* f of $T = [p_0, \ldots, p_n]$ is the convex hull of $\left\{p_{\sigma(i)} \in \mathbb{R}^n\right\}_{j=0}^k$ for some $\sigma \in \Sigma(0:k; 0:n)$, denoted $f_\sigma := \left[p_{\sigma(0)}, \ldots, p_{\sigma(k)}\right]$. The collection of k-subsimplices of T is denoted $\Delta_k(T)$, and the *collection of all subsimplices of* T is denoted $\Delta(T) = \bigcup_{k=0}^n \Delta_k(T)$.

A *simplical mesh* $\mathcal{T}$ of a domain Ω is a collection of disjoint simplices T_i such that

(1) $\bigcup_i T_i = \Omega$, and
(2) All intersections of two simplices $f_{ij} = T_i \cap T_j$ must be either $f_{ij} = \emptyset$ or $f \in \Delta(T_1), \Delta(T_2)$.

The collection of all the k-subsimplices of $\mathcal{T}$ is denoted $\Delta_k(\mathcal{T})$. The set of all simplices $T \in \mathcal{T}$ that share the subsimplex $f \in \Delta_k(\mathcal{T})$ $(T \cap f \neq 0)$ is denoted $\omega_f(\mathcal{T})$.

It can also be noted that these meshes can be *refined* with respect to the parameter h. For a simplex T we have the parameter $h_T := \frac{\operatorname{diam}(T)}{2}$, and for the mesh, $h := \max_{T \in \mathcal{T}} h_T$. This is a measure of the coarseness of the mesh $\mathcal{T}$. For simplicial meshes, we can divide the mesh into more simplices by bisecting them thus decreasing the coarseness h, and this is what is meant when writing V_h for the subspace.

2.4.2. Shape functions (polynomials).

Shape functions are piecewise functions over our mesh $\mathcal{T}$, continuous on each $T \in \mathcal{T}$. Among these are the functions that make up our subspace V_h of $H^1(\Omega)$. The natural choices for a basis on T are a trigonometric basis (e.g. Fourier series) or a polynomial basis (e.g. Taylor series), since these are easy to differentiate and integrate. Building upon work done in **[2, 3, 8, 16]**, the objects of study in this thesis are polynomials, therefore we abandon trigonometric series at this point. Before we go on with polynomials, a short definition of multiindex notation is necessary:

DEFINITION 2.9. *Multiindex notation*

A *multi-index* j is an $(n+1-m)$-tuple $(j_m, \ldots, j_n)$, $j_i \in \mathbb{N}_0$ which describes the respective degrees of a monomial over $\mathbb{R}^n$:

$$x^j := x_1^{j_1} \cdots x_n^{j_n}$$

$|j| := \sum_i j_i$ is *the degree of* j, $[\![j]\!] := \{\, i \in \mathbb{N}_0 |\ j_i \neq 0\}$ *is the support of* j, and *the set of multi-indices* is written $\mathbb{N}_0^{m:n}$.

DEFINITION 2.10. Polynomial function spaces

Given a domain $T \subset \mathbb{R}^n$, the *space of rth degree polynomials over* T is denoted

$$\mathcal{P}_r(T) := \left\{ p : T \to \mathbb{R} \;\middle|\; \forall i \in \mathbb{N}_0^{1:n}, |i| \leq r : \exists a_i \in \mathbb{R} : \forall x \in T : p(x) = \sum_{|i| \leq r} a_i x^i \right\}$$

where $\mathbb{N}_0^{1:n}$ is the space of natural number-valued n-tuples. More compactly written:

$$\mathcal{P}_r(T) := \left\{ \sum_{|i| \leq r} a_i x^i \;\middle|\; a_i \in \mathbb{R} \right\}$$

The direct sum of these spaces (assuming $p \in \mathcal{P}_r(T)$ is zero for $x \notin T$)

$$\bigoplus_{T \in \mathcal{T}} \mathcal{P}_r(T) \tag{2.4.1}$$

will give a space of discontinuous functions. This does not satisfy Theorem 2.3, and we must therefore restrict this space a bit more.

2.4.3. Continuity.

We still need to restrict the space from (2.4.1) to a proper subspace of $H^1(\Omega)$. Theorem 2.3 implies that if $u \in \bigoplus_{T \in \mathcal{T}} \mathcal{P}_r(T)$ and $u \in H^1(\Omega)$, then $u \in C(\Omega)$ (i.e. u is continuous). Hence we need to ensure that our subspace contains only continuous functions. The name for such a space is a *conforming* finite element space, where conforming implies that V_h is a subspace of $H^1(\Omega)$. We let $V_h := H\mathcal{P}_r(\mathcal{T})$ as defined here:

DEFINITION 2.11. Continuous piecewise polynomial function spaces

The *continuous piecewise polynomial functions* are

$$H\mathcal{P}_r(\mathcal{T}) := \{ u \in C(\Omega) \,|\, u|_T \in \mathcal{P}_r(T) \},$$

meaning piecewise polynomials that are continuous on the edges between the simplices, required by Theorem 2.3.

According to [**2**, p. 60-61] this space is well-defined and at any $f \in \Delta_k(\mathcal{T})$ the trace $\mathrm{Tr}_f(p)$ for $p \in \mathcal{P}_r(\mathcal{T})$ is single-valued. Thus we are certain that $\mathcal{P}_r(\mathcal{T})$ is a nondegenerate subspace of $H^1(\Omega)$, i.e. $\dim(\mathcal{P}_r(\mathcal{T})) > 0$.

2.4.4. Basis of local support.

We want to generate a basis $\{\phi_i\}_{i=1}^{\dim H\mathcal{P}_r(\mathcal{T})}$ for $H\mathcal{P}_r(\mathcal{T})$ that has *local support*, i.e. that $\mathrm{supp}(\phi_i)$ is covered by a small subset $\varpi_i(\mathcal{T})$ of $\mathcal{T}$, not overlapping with $\mathrm{supp}(\phi_j)$ for too many $j \neq i$.[4] The reason for this is that we are working with evaluating integrals over Ω of the form

$$I_{ij} = \int_\Omega f(x) \mathsf{der}_1(\phi_i) \mathsf{der}_2(\phi_j) \mathrm{d}x$$

where each der_m can be either the identity or $\frac{\partial}{\partial x_i}$ and $f(x) \in C(\Omega)$ is an arbitrary function. We can then restrict the integral to

$$\int_{\mathrm{supp}(\phi_i) \cap \mathrm{supp}(\phi_j)} f(x) \mathrm{der}_1(\phi_i) \mathrm{der}_2(\phi_j) \mathrm{d}x = \sum_{T \in (\varpi_i(\mathcal{T}) \cap \varpi_j(\mathcal{T}))} \int_T f(x) \mathrm{der}_1(\phi_i) \mathrm{der}_2(\phi_j),$$

[4] $\mathrm{supp}(u) := \{ x \in \Omega \,|\, u(x) \neq 0 \}$

which is zero when $\varpi_i(\mathcal{T}) \cap \varpi_j(\mathcal{T}) = \emptyset$ regardless of the choice of the der_m. This makes the matrix $\{I_{ij}\}_{i,j}$ sparse when the $\mathrm{supp}(\phi_i)$ are many and far apart in Ω, and our stiffness matrix $\mathbb{A}_{ij} = a(\phi_i, \phi_j)$ which is based on a weighted sum of I_{ij}-matrices, will have the same sparseness property.

2.4.5. Degrees of Freedom.

To define a basis in $H\mathcal{P}_r(\mathcal{T})$ which has local support, we have to take a detour through the dual space $H\mathcal{P}_r(\mathcal{T})^*$. In this subsection we define and give a few examples of different spaces of the degrees of freedom of $H\mathcal{P}_r(\mathcal{T})$. First we need to establish what exactly the dual space is:

DEFINITION 2.12. Given a vector space X, the *dual space* X^* is the space of bounded linear functionals over X.

FACT 2.13. *By* [**15**, Th. 5.1] *if X is finite dimensional with basis $\{x_i\}_i$, then X^* has a basis $\{f_i\}_i$ such that $f_i(x_j) = \delta_{ij}$. In particular* $\dim X^* = \dim X$.

Since $\dim H\mathcal{P}_r(\mathcal{T}) < \infty$, then $\dim H\mathcal{P}_r(\mathcal{T}) = \dim H\mathcal{P}_r(\mathcal{T})^*$ by Fact 2.13 $H\mathcal{P}_r(\mathcal{T})$ and $H\mathcal{P}_r(\mathcal{T})^*$ are isomorphic as vector spaces. We can also reverse the process of Fact 2.13 by choosing a basis for $H\mathcal{P}_r(\mathcal{T})^*$ that induces a basis on $H\mathcal{P}_r(\mathcal{T})$ with desirable properties such as local support. We call this basis the *degrees of freedom* or *nodes:*

DEFINITION 2.14. The *degrees of freedom* (also called *nodes*) $\mathcal{N}_\mathcal{T} = \{n_i\}_{i=1}^{\dim H\mathcal{P}_r(\mathcal{T})}$ of $\mathcal{T}$ are linearly independent elements of $H\mathcal{P}_r(\mathcal{T})^*$ that uniquely determine any function in $H\mathcal{P}_r(\mathcal{T})$ ($u = v \in H\mathcal{P}_r(\mathcal{T})$ if $\forall \varphi \in H\mathcal{P}_r(\mathcal{T})^* \phi(u - w) = 0$). The degrees of freedom associated with a simplex T are denoted $\mathcal{N}_T = \left\{n_i^T\right\}_{i=1}^{\dim \mathcal{P}_r(T)}$. They are linearly independent elements of $\mathcal{P}_r(T)^*$ that together can be used to uniquely determine any $u \in \mathcal{P}_r(T)$.

The degrees of freedom need to be constructed as integral evaluations, a certain number restricted to certain subsimplices of T. This is because when elements T_1, T_2 are linked together on the subsimplex $f = T_1 \cap T_2$ we need $\mathcal{P}_r(T_1)|_f = \mathcal{P}_r(T_2)|_f$. This can be done by choosing $\mathcal{P}_r(T_1)^*$ and $\mathcal{P}_r(T_2)^*$ such that $\mathcal{P}_r(T_1)^*|_f = \mathcal{P}_r(T_2)^*|_f$.

Given a domain $\Omega = [0,1]^2$ (as in Figure 2.4.1 on page 17) with a triangular mesh $\mathcal{T}$, examples of such conformity-enforcing degrees of freedom on an element [5] are:

EXAMPLE 2.15. Linear elements (see Figure 2.4.2 on page 21)

In the case of a linear element where $\mathcal{P}_1(T)$ is the function space, we know that $\dim \mathcal{P}_1(T) = \dim T + 1$, which equals the number of vertices of T. Hence it makes sense to let $\mathcal{N}_T$ consist of evaluations at the ith vertex point, i.e. $\mathcal{N} \ni n_i(u) := \int_{\{x_i\}} u dx = u(x_i)$. The vertices of all $T \in \mathcal{T}$ are what connects them, and we have continuity of $H\mathcal{P}(\mathcal{T})$ at the subsimplices. The linear element is usually the starting point for all conforming families of elements over $H^1(\Omega)$, and it is a simple version of the two in Examples 2.16 and 2.17.

EXAMPLE 2.16. Point evaluation in a triangle (see Figure 2.4.3 on page 22 and 2.4.4) This is a basis for $\mathcal{P}_r(T)^*$, where the nodes are point evaluations uniformly

[5] a collection $(T, \mathcal{F}, \mathcal{N})$ of T, and shape functions $\mathcal{F}$ and a basis $\mathcal{N}$ for the dual space of span$\mathcal{F}$

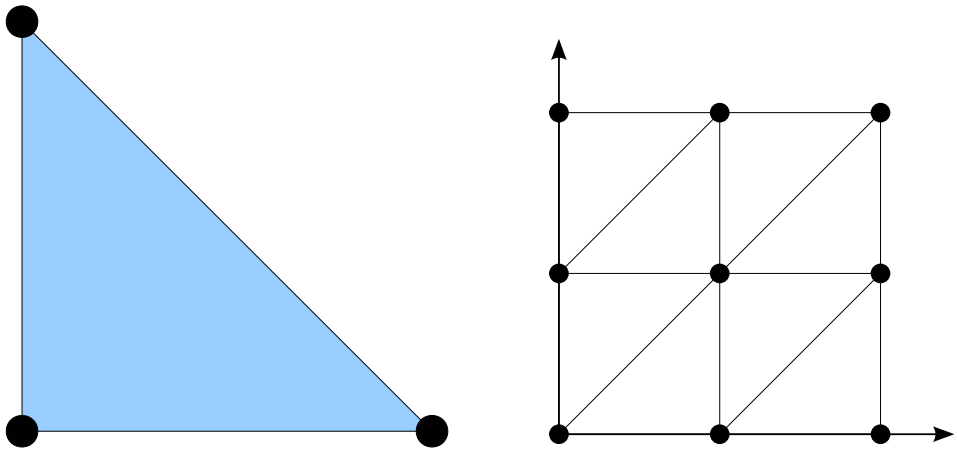

FIGURE 2.4.2. On the left a single linear element in two dimensions, with point evaluation. When several simplices are connected together their degrees of freedom are shared as in the right figure.

distributed throughout the triangle in the fashion of triangular numbers, which in fact are $\binom{2+r}{2} = \dim \mathcal{P}_r(T)$.

These elements are constructed so that an appropriate number of their nodes will coincide with (be linearly dependent of) the nodes of neighboring elements on the edges and in the vertices. Observe that the evaluations on the edges (including vertices) are enough to determine the polynomial degree uniquely for that edge. Three evaluations for second degree polynomials, and four for third degree polynomials. In other words,

$$\bigcup_{T\in\mathcal{T}} \mathcal{N}_T = \mathcal{N}_{\mathcal{T}}.$$

In fact, $\dim \mathcal{P}_r(\mathbb{R}^n) = \binom{n+r}{n}$ which are the n-simplicial numbers in Pascal's triangle. This makes it easy to place the elements of $\mathcal{N}_T$ uniformly throughout any T and its subsimplices.

EXAMPLE 2.17. Weighted integrals on the subsimplices

We can replace each node from Example 2.16 (evaluation at $\{x_i \in T\}_{i=1}^{\dim \mathcal{P}_r(T)}$) with (linearly independent) integrals on the smallest subsimplex containing x_i,

$$\bigcap_{f\in\Delta(T), x_i\in f} f.$$

This we do to preserve the number of nodes on the edges of T, so that the linking to neighbouring T (similar to the linking in Figure 2.4.4 on the following page) is preserved.

Letting ψ_f^i for $i \leq \dim \mathcal{P}_r(f) = \binom{\dim f+r}{\dim f}$ be a basis for $\mathcal{P}_r(f)$ for all $f \in \Delta(\mathcal{T})$, we can construct a basis for the dual space:

$$\mathcal{D}_r(T) = \left\{ \int_f \mathrm{Tr}(u)\psi_f^i \mathrm{d}x \,\middle|\, f \in \Delta(T),\, 1 \leq i \leq \binom{r-1}{\dim f} \right\}. \tag{2.4.2}$$

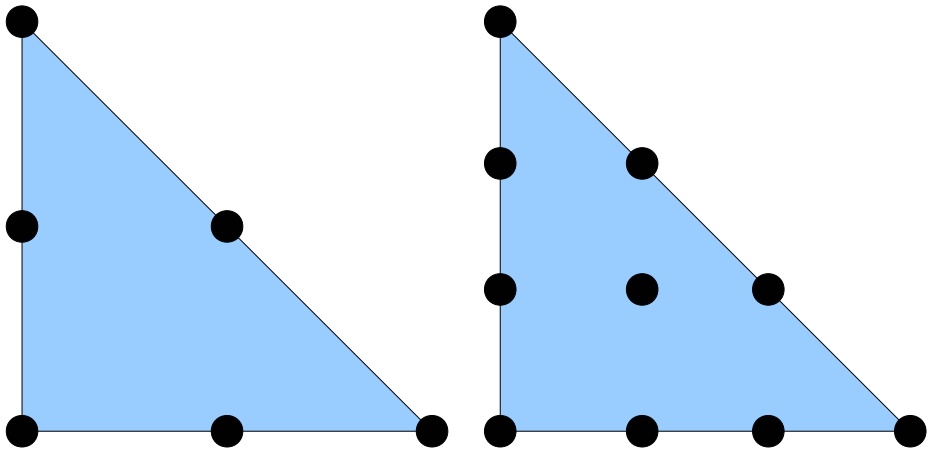

FIGURE 2.4.3. Point evaluation on a triangle: A quadratic element (left) and a qubic element (right).

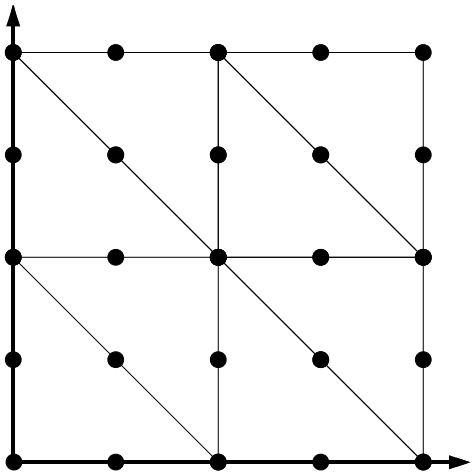

FIGURE 2.4.4. A quadratic mesh.

For the entire mesh this is

$$\mathcal{D}_r(\mathcal{T}) = \left\{ \int_f \mathrm{Tr}(u)\psi_f^i \mathrm{d}x \,\middle|\, f \in \Delta(\mathcal{T}),\, 1 \leq i \leq \binom{r-1}{\dim f} \right\}. \tag{2.4.3}$$

The reader should note that, as in Example 2.16 the number of degrees of freedom associated to each subsimplex (including its subsimplices) is $\binom{\dim f + r}{\dim f}$.

These examples show us that there are several possible ways of constructing a basis for $\mathcal{P}_r(T)^*$. For our purpose of obtaining a basis for $H\mathcal{P}_r(\mathcal{T})$ with local support we now have the appropriate tools.

2.4.6. Construction of a basis. We want a basis $\{\phi_i\}_i$ for $H\mathcal{P}_r(\mathcal{T})$, which has local support, i.e. that it has support on few $T \in \mathcal{T}$. To construct this basis, we will use the a basis for any degrees of freedom from the last example, called $\mathcal{N}_T$. We will then generate a *nodal basis* $\{\phi_i\}_i$ defined by $\forall i, j : n_i(\phi_j) = \delta_{ij}$. Let us explore what this means:

In the case of linear elements, the $n_i \in \mathcal{N}(T)$ represents point evaluations in the vertices of $\mathcal{T}$. We will then have $\{\phi_i\}_i$ that are piecewise linear, and since they are 0 in all but one vertex $\{t_i\} \in \Delta_0(\mathcal{T})$, each ϕ_i will have support only on the simplices surrounding t_i, $\omega_{\{t_i\}}(\mathcal{T})$. Hence we have local support. The nodal basis for these vertices are the barycentric coordinates:

DEFINITION 2.18. Barycentric coordinates

The *barycentric coordinates* $\lambda_i^T(x)$ of a simplex T are here interpreted as a function of $x \in \mathbb{R}^n$ where the $\lambda_i^T(x)$ are defined to be s.t.

$$x = \sum_{i=0}^{n} \lambda_i^T(x) p_i$$

where $\{p_i\}_{i=0}^n$ are the vertices of T.

The barycentric coordinates of the reference simplex, $\lambda_i^{T_0}(x)$ are written without T_0:

$$\lambda_i(x) := \lambda_i^{T_0}(x) = \begin{cases} x_i & \text{when } i \geq 1 \\ 1 - \sum_{j=1}^n x_j & \text{when } i = 0 \end{cases}.$$

It is worth noting that $\sum_{i=0}^n \lambda_i^T = 1$.

For the two other examples of degrees of freedom, it will suffice to say that there exists a nodal basis $\{\phi_i\}_i$ where $\forall i, j : n_i(\phi_j) = \delta_{ij}$. In the case of the point evaluation nodes on T, let $n_i(\phi_i) = \phi_i(x_i) = 1$, and

$$f = \bigcap_{g \in \Delta(\mathcal{T}),\, x_i \in f} g.$$

We see that for all $T \notin \omega_f(\mathcal{T})$, $\forall j : n_j^T(\phi_i) = 0$, and thus $\text{supp}\phi_i = \omega_f(\mathcal{T})$. The case of integral evaluations along subsimplices produces nodal basis functions with $\text{supp}\phi_i = \omega_f(\mathcal{T})$ along a similar argument.

EXAMPLE 2.19. Now we can describe the nodes of Example 2.16 in more detail, because they are distributed as follows: Let $\lambda = (\lambda_0, \lambda_1, \lambda_2)$ be the barycentric coordinates of $T \subset \mathbb{R}^n$. Then the nodes n_i are point evaluation at barycentric coordinates

$$\left\{ \left(\frac{i_0}{r}, \frac{i_1}{r}, \frac{i_2}{r} \right) \middle| \; i \in \mathbb{N}_0^{0:2}, |i| = r \right\}$$

The barycentric basis, which will be in our focus in this thesis, has the same property as these nodal bases that it has $\text{supp}\phi_i = \omega_f(\mathcal{T})$ for some $f \in \Delta(\mathcal{T})$:

DEFINITION 2.20. Barycentric (monomial) basis function

The polynomial bases of $\mathcal{P}_r(T)$ can also be represented by monomials of barycentric coordinates $\left(\lambda^T\right)^i = \prod_{j=0}^n \left(\lambda_j^T\right)^{i_j}$:

$$\left\{ \left(\lambda^T\right)^i \middle| \; |i| = r \right\}$$

where $r = |i| = \sum_j i_j \; i \in \mathbb{N}_r^{0:n}$.

It is often common to substitute the barycentric basis with the Bernstein basis:

DEFINITION 2.21. Bernstein basis
The *Bernstein basis* B_j^T is based upon the barycentric basis $\left(\lambda^T\right)^j$ such that

$$B_j^T = \binom{r}{j}\left(\lambda^T\right)^j$$

where $r = |j|$ and $\binom{r}{j} := \frac{r!}{j_0!\cdots j_n!(r-|j|)!}$ is the *multinomial coefficient*.

This basis is also called the *normalised barycentric basis* because every $\int_{T_0} B_j^{T_0}\mathrm{d}x = 1$ (proved in [**8**, p. 140].) It is then a scaling of the barycentric basis, and we will see in Chapter 3 that The Bernstein basis is very well conditioned compared to the barycentric basis.

2.4.7. Convergence of solution. In essence, we have convergence of solution from the fact that $\mathcal{P}(T) := \sum_{r=1}^{\infty}\mathcal{P}_r(T)$ is dense in $H^1(\Omega)$ and that $\mathcal{P}(T)$ is dense in $H^1(T)$. But knowing the rate at which it converges can be much harder, and can only be obtained for solutions of variational problems in certain subsets of $H^1(\Omega)$, namely $H^t(\Omega)$ for $t \geq 2$.

FACT 2.22. [**6**, Th. 6.4] *Assuming $\mathcal{T}$ is shape-regular*[6] *the convergence of the solution is certain with the rate*

$$\|u - \Pi_h u\|_{L_2(\Omega)} \leq ch^t\left(\sum_{|i|\leq t}\left\|D^i u\right\|_{L_2(\Omega)}^2\right)^{\frac{1}{2}}.$$

This requires that $u \in H^t(\Omega)$, that h is half the largest diameter of all $T \in \mathcal{T}$ and that Π_h is interpolation by a piecewise polynomial of degree $r = t - 1 \geq 1$.

Actually the theorem is stated for polynomial interpolation, but according to Cea's Lemma, $\|u - u_h\| \leq \|u - \Pi_h u\|$, so we can be sure that our numerical solution u of our variational problem converges just as well.

This makes h a very good parameter for ensuring convergence of the solution. Technically, we can also refine the polynomial degree of $H\mathcal{P}_r(\mathcal{T})$, and because polynomials are dense in $C(\Omega)$, which is dense in $H^1(\Omega)$, this will also create convergence. But this is the subject of another method, the *hp-FEM*, so here we have no estimate for the error depending on r.

2.4.8. Time-efficient calculations. Our evaluating of $\mathbb{A}_{ij} = a(\phi_i, \phi_j)$ on a computer requires a process of differentiation and integration with a symbolic engine (e.g. maple). In general, this kind of calculation is very cumbersome for a computer compared to regular numerical operations, and motivates us to try to cut down on the use of these symbolic integrations. Our tool for this is called *affine equivalence*, which tells us that we only need to perform one standardised symbolic calculation per stiffness matrix, instead of doing one per $T \in \mathcal{T}$.

DEFINITION 2.23. Affine equivalence
Two elements $(T_1, \mathcal{F}_1, \mathcal{N}_1)$ and $(T_2, \mathcal{F}_2, \mathcal{N}_2)$ are *affine equivalent* if there exists an affine injective transformation $F : \mathbb{R}^n \to \mathbb{R}^n$ such that the images of F, F^* and F_* are

[6] [**6**, p. 61]: $\exists\kappa > 0 : \forall T \in \mathcal{T}$: the inscribed circle of has a radius $\geq h_T/\kappa$ where h_T is the length of the longest line of T.

(1) $F(T_1) = T_2$
(2) $F^*(\mathcal{F}_2) = \mathcal{F}_1$
(3) $F_*(\mathcal{N}_1) = \mathcal{N}_2$

where F^*, $F^*(f) := f \circ F$ for $f \in \mathcal{F}_2$, is called the *pullback function* and F_*, $F_*(N) := N \circ F^*$ for $N \in \mathcal{N}_1$, is called the *push-forward function*.

Affine equivalence is really vital to efficiency when doing calculations with the FEM, since the number of symbolic calculations when doing the calculation of the Stiffness matrix shrinks by a factor of $|\mathcal{T}|$. The reason is simply that the chain rule for derivation acts on scalar functions such that

$$D_x(u \circ F(x)) = D_x u(F(x)) = \det\left(D_x F(x)\right) D_{F(x)} u(F(x)).$$

The stiffness matrix $\mathbb{A}_{ij} = a(\phi_i, \phi_j)$ for the basis $\{\phi_i\}_{i=1}^N$ of $H\mathcal{P}_r(\mathcal{T})$ has to be calculated for every simplex $T \in \mathcal{T}$, with a local matrix $\mathbb{A}_{ij}^T = a(\phi_i^T, \phi_j^T)_T$ (the integral restricted to T) from the local bases $\left\{\phi_i^T\right\}_{i=1}^M \subset \{\phi_i\}_i^N$ (those that have support on T).

EXAMPLE 2.24. Let's take a simplified version of the equation component (from Subsection 2.4.4)

$$I_{ij}^{T_1} = \int_{T_1} \mathsf{der}_1\left(\phi_i^{T_2} \circ F\right) \mathsf{der}_2\left(\phi_j^{T_2} \circ F\right) \mathrm{d}x$$

$$= \int_{T_1} \left(\mathsf{der}_1(F) \cdot D_{F(x)}\left(\phi_i^{T_2} \circ F\right)\right)\left(\mathsf{der}_2(F) \cdot D_{F(x)}\left(\phi_j^{T_2} \circ F\right)\right) \mathrm{d}x.$$

Changing the domain of the integral gives us

$$= \int_{T_2} \left(\mathsf{der}_1(F) \cdot D\phi_i^{T_2}\right)\left(\mathsf{der}_2(F) \cdot D\phi_j^{T_2}\right) \mathrm{d}x.$$

Switching to component notation, we get

$$= \sum_{m,l} \int_{T_2} \left(\mathsf{der}_1(F)\right)_m \left(D\phi_i^{T_2}\right)_m \left(\mathsf{der}_2(F)\right)_l \left(D\phi_j^{T_2}\right)_l \mathrm{d}x$$

$$= \sum_{m,l} \left(\mathsf{der}_1(F)\right)_m \left(\mathsf{der}_2(F)\right)_l \int_{T_2} \left(D\phi_i^{T_2}\right)_m \left(D\phi_j^{T_2}\right)_l \mathrm{d}x = \sum_{m,l} \mathbf{F}_{ml} \mathbf{D}_{ml}^{T_2}.$$

so we need only calculate $D\phi_i^{T_2}$ for one $T_2 \in \mathcal{T}$ in order to evaluate the integral I_{ij}^T for all other $T \in \mathcal{T}$. This calculation requires only that we compute integrals for one element to find $\mathbb{A}^{T_i}$ for one i, and the rest can be calculated with simple linear algebra operations. Since analytically computing integrals is much more time-consuming than computing the determinant of an affine transformation, the computing time is decreased significantly.

2.4.9. In conclusion. We have now chosen $H\mathcal{P}_r(\mathcal{T})$ as our subspace V_h. It has basis functions with local support which generates a sparse stiffness matrix, its functions are continuous, a solution in V_h converges towards the exact solution in V, and we don't require many symbolic calculations in the process. In later chapters (3, 5) we will only consider the bases on a single T, because the calculations we are trying to optimize (stiffness matrix calculation) is done element by element.

CHAPTER 3

Condition numbers

In this chapter we will describe the importance of condition numbers, how we calculate them, and their relevance to FEM solutions of PDE. [**18**] explains condition numbers quite adequately:

> In the numerical analysis, the *condition number* associated with a problem is a measure of that problem's amenability to digital computation, that is, how numerically well-conditioned the problem is. A problem with a low condition number is said to be *well-conditioned*, while a problem with a high condition number is said to be *ill-conditioned*.

In other words, a condition number is an abstract measure of how well a computer-based solution method for any problem performs.

In our case we're dealing with linear algebra equation systems of the kind $Ax = b$ like (2.3.3). We want to find out what consequences truncation errors on b have on the solution x. The following result from [**13**, p. 155-159] gives us a *condition number of a matrix* to work with:

FACT 3.1. *Suppose* $A \in \mathbb{C}^{n,n}$ *is nonsingular,*[1] $b, e \in \mathbb{C}^n$, $b \neq 0$ *and* $Ax = b$, $Ay = b + e$. *Then*

$$\frac{1}{\operatorname{cond}(A)} \frac{\|e\|}{\|b\|} \leq \frac{\|y - x\|}{\|x\|} \leq \operatorname{cond}(A) \frac{\|e\|}{\|b\|}, \quad \operatorname{cond}(A) := \|A\| \cdot \left\|A^{-1}\right\|. \tag{3.0.4}$$

What this means is that by having a low condition number, we can limit the relative error of the solution of $Ax = b$ by a factor of $\operatorname{cond}(A)$.

Condition numbers are quite a useful tool, so let us investigate a bit further what this expression $\|A\| \cdot \left\|A^{-1}\right\|$ *actually* means in our case when A is symmetric: $\|A\|$ is the norm of A , defined as $\sup_{x \in \mathbb{C}^n} \frac{|x^t A x|}{|x|^2}$, which is the magnitude of the largest eigenvalue $\left|\mu^A_{\max}\right|$ of the matrix A. Its smallest eigenvalue satisfies $\left|\mu^A_{\min}\right| = \frac{1}{\left|\mu^{A^{-1}}_{\max}\right|}$ where $\mu^{A^{-1}}_{\max}$ is the largest (in magnitude) eigenvalue of A^{-1}. Since $\left|\mu^A_{\max}\right| \geq \left|\mu^A_{\min}\right|$, the expression from (3.0.4) then becomes

$$\|A\| \cdot \left\|A^{-1}\right\| = \frac{\left|\mu^A_{\max}\right|}{\left|\mu^A_{\min}\right|} \geq 1.$$

Then we know that the lowest possible condition number we can obtain is 1, which is equivalent to A being a unitary matrix.

[1] $\mathbb{C}^{n,n}$ denotes all complex-valued $n \times n$ matrices.

3.1. Relation to variational forms' condition numbers

In this section we will describe the relation between the condition numbers of the stiffness matrix $\mathbb{A} = \{a(\phi_i, \phi_j)\}_{ij}$, the bilinear (and bounded, coercive, and symmetric) form restricted to the subspace V_h, $a_h : V_h \times V_h \to \mathbb{R}$, $(u, v) \mapsto a(u, v)$ (the same as $a|_{V_h \times V_h}$, thus we will write only a where they are interchangeable) and the basis $\{\phi_i\}_i$.

Now, the equation system we're regarding is based on the stiffness matrix which in itself consists of a basis $\{\phi_i\}_{i=1}^N$ and a bilinear form a. We now expand the concept of condition number to these two mathematical objects.

DEFINITION 3.2. Condition numbers of bases

If we have a basis $\{\phi_i\}_{i=1}^N$ for a N-dimensional Hilbert space within $L^2(\Omega)$ (with same norm and inner product) where $N < \infty$, then

$$\operatorname{cond}\left(\{\phi_i\}_{i=1}^N\right) := \frac{\sup_{c\in\mathbb{R}^N} \frac{\left\|\sum_i c_i\phi_i\right\|_{L^2(\Omega)}}{|c|_{l^2}}}{\inf_{c\in\mathbb{R}^N} \frac{\left\|\sum_i c_i\phi_i\right\|_{L^2(\Omega)}}{|c|_{l^2}}} \tag{3.1.1}$$

where $|c|_{l^2}$ is the Euclidean norm of c. We see that the expression

$$\left\|\sum_i c_i\phi_i\right\|_{L^2(\Omega)} = \left(\sum_i c_i\phi_i, \sum_j c_j\phi_j\right)_{L^2(\Omega)}^{\frac{1}{2}} = \left(c^t G c\right)^{\frac{1}{2}}$$

where G is the *Gram matrix* (We will write only G when only it is obvious which basis is used.)

$$G\left(\{\phi_i\}_{i=1}^N\right) := \left\{(\phi_i, \phi_j)_V\right\}_{i,j=1}^N.$$

We then get that $\sup_{c\in\mathbb{R}^N} \frac{\left\|\sum_i c_i\phi_i\right\|_{L^2(\Omega)}}{|c|_{l^2}} = \sup_{c\in\mathbb{R}^N} \frac{\left(c^t G c\right)^{\frac{1}{2}}}{|c|_{l^2}}$ is the square root of highest eigenvalue of G by magnitude. Similarly, $\inf_{c\in\mathbb{R}^N} \frac{\left\|\sum_i c_i\phi_i\right\|_{L^2(\Omega)}}{|c|_{l^2}}$ is the square root of the lowest, and as a consequence (3.1.1)$= \sqrt{\frac{\mu^G_{\max}}{\mu^G_{\min}}}$. From now on we will work on the Gram matrix G, since has some nice properties, such as linearity and relation to the following condition number (see Theorem 3.4):

DEFINITION 3.3. Condition number of a bilinear form

Suppose $a : V \times V \to V$ is a symmetric, bounded, coercive bilinear form on the Hilbert space V , then

$$\operatorname{cond}(a) := \lambda^a_{\max}/\lambda^a_{\min}$$

is the *condition number* of a where $\lambda^a_{\max} := \sup_{x\in V_h} \frac{a(x,x)}{\|x\|^2}$, $\lambda^a_{\min} := \inf_{x\in V_h} \frac{a(x,x)}{\|x\|^2}$ are respectively the absolute of the highest and lowest eigenvalues of a.[2]

Having defined these two condition numbers, we can state that the most important part of the following theorem is the inequality

$$\operatorname{cond}(\mathbb{A}) \le \operatorname{cond}(a_h) \operatorname{cond}\left(G\left(\{\phi_i\}_{i=1}^{\dim V_h}\right)\right).$$

[2]Being the highest and lowest values of λ for which the eigenvalue problem $\forall v \in V : a(u, v) = \lambda(u, v)$ has a solution.

As we will see, $\mathbb{A} := a(\phi_i, \phi_j)$ here is dependent both on choice of problem (a), subspace V_h and a basis ϕ_i. The restricted bilinear form a_h is only dependent on choice of a and V_h, while G on the other hand only depends on $\{\phi_i\}_{i=1}^{\dim V_h}$. We can't do much about our bilinear form a, because it represents the problem we want to solve, but the choice of bases $\{\phi_i\}_{i=1}^{\dim V_h}$ are quite many!

In that way we may condition $\mathbb{A}$ through improvement of G, and this will work for any problem a which can be restriced to V_h. This is our main motivation for exploring the condition numbers of different polynomial bases, comparing them to each other to see what benefits us the most.

THEOREM 3.4. *Given the variational problem*

$$\forall v \in V_h : a(u, v) = (f, v)$$

as in (2.3.2) *and a finite basis* $\mathcal{F} = \{\phi_i\}_{i=1}^{\dim V_h}$ *for* V_h*, define the stiffness matrix* $\mathbb{A}$ *by* $\mathbb{A}_{ij} := a(\phi_i, \phi_j)$*. Then*

1. $\operatorname{cond}(\mathbb{A}) \leq \operatorname{cond}(a_h)\operatorname{cond}(G)$*, and*
2. *If the* ϕ_i *are orthonormal, then* $\operatorname{cond}(a_h) = \operatorname{cond}(\mathbb{A})$*, and*
3. *There exists a basis* ϕ_i *such that* $\operatorname{cond}(\mathbb{A}) = 1$.

PROOF. To prove Item 1, we must show that $\mathbb{A}U \leq \lambda_{\max}^{a_h}\lambda_{\max}^{G}U$ and $\mathbb{A}U \geq \lambda_{\min}^{a_h}\lambda_{\min}^{G}U$. Suppose that $u \in V_h$ satisifies

$$\forall v \in V_h : a(u, v) = \lambda(u, v), \qquad \lambda \in \mathbb{R}$$

i.e. solves the eigenvalue problem for the eigenvalue λ of a_h in V_h. Using the basis $\mathcal{F}$ for V_h, we get

$$\forall j \in \mathbb{N}_0^{1:\dim V_h} : a\left(\sum_i U_i\phi_i, \phi_j\right) = \lambda\left(\sum_i U_i\phi_i, \phi_j\right)$$

$$\forall j \in \mathbb{N}^{1:\dim V_h} : \sum_i U_i a(\phi_i, \phi_j) = \lambda\sum_i U_i(\phi_i, \phi_j) \tag{3.1.2}$$

We can write this as

$$\mathbb{A}U = \lambda GU$$

where $\mathbb{A}_{ij} = a(\phi_i, \phi_j)$ and $G_{ij} = (\phi_i, \phi_j)$. Suppose that $\lambda_{\max}^{a_h}$ is the maximal and $\lambda_{\min}^{a_h}$ is the minimal eigenvalue of a, and that $\lambda_{\max}^{G}$ is the maximal and $\lambda_{\min}^{G}$ is the minimal eigenvalue of G. We conclude from the spectral theorem and a's property as coercive (only positive eigenvalues) that

$$\|\mathbb{A}U\| \leq \lambda_{\max}^{a_h}\|GU\| \leq \lambda_{\max}^{a_h}\lambda_{\max}^{G}\|U\|, \text{ and}$$
$$\|\mathbb{A}U\| \geq \lambda_{\min}^{a_h}\|GU\| \geq \lambda_{\min}^{a_h}\lambda_{\min}^{G}\|U\|.$$

Consequently

$$\operatorname{cond}(\mathbb{A}) = \frac{\lambda_{\max}^{\mathbb{A}}}{\lambda_{\min}^{\mathbb{A}}} \leq \frac{\lambda_{\max}^{a_h}\lambda_{\max}^{G}}{\lambda_{\min}^{a_h}\lambda_{\min}^{G}} = \operatorname{cond}(a)\operatorname{cond}(G). \tag{3.1.3}$$

For Item 2 we know that a is a bilinear form on V_h. Since V_h is finite it has an orthonormal basis $\{e_i\}_i$, thus we can define the matrix

$$\hat{\mathbb{A}}_{ij} := a(e_i, e_j)$$

which we will show has the same condition number as a_h. Define

$$W(a_h) := \left\{ \frac{a(x,x)}{\|x\|^2} \right\}_{x \in V_h} = \{a(x,x)\}_{\|x\|=1}$$

and since $\|x\|^2 = 1 \Leftrightarrow \sum_i c_i^2 = 1$, [3]

$$= \left\{ a\left(\sum_i c_i e_i, \sum_j c_j e_j \right) \right\}_{\sum_i c_i^2 = 1} = \left\{ \sum_{i,j} c_i c_j a(e_i, e_j) \right\}_{\sum_i c_i^2 = 1}$$

$$= \left\{ \sum_{i,j} c_i c_j \hat{\mathbb{A}}_{ij} \right\}_{\sum_i c_i^2 = 1} = \left\{ c \hat{\mathbb{A}} c^t \right\}_{\|c\|=1}$$

which is the numerical range of both a and $\hat{\mathbb{A}}$ over the unit circle $\|x\| = 1$, so $\sup(W(a_h))/\inf(W(a_h))$ is both the condition number of A_h and $\hat{\mathbb{A}}$.

Proving Item 3 we assume the orthonormal basis $\{e_i\}_{i=1}^N$ of the previous item, and we will show that there exists a basis $\{\phi_i\}_{i=1}^N$ with $\phi_i = \sum_l d_{il} e_l$ such that $\mathrm{cond}(\mathbb{A}) = 1$ where $\mathbb{A}_{ij} := a(\phi_i, \phi_j)$. We know that $\hat{\mathbb{A}} = a(e_i, e_j)$ is symmetric and positive definite since a is symmetric and coercive. This in turn implies a singular value decomposition $\hat{\mathbb{A}} = B \Sigma B^t$ for a unitary matrix B and diagonal matrix Σ. Having a look at

$$\mathbb{A} = \{a(\phi_i, \phi_j)\}_{i,j} = \left\{ \sum_{l,m} d_{il} a(e_l, e_m)_{V_h} d_{jm} \right\}_{i,j} = d \hat{\mathbb{A}} d^t$$

we realise that f we choose the ϕ_i such that $d = \Sigma^{-\frac{1}{2}} B^{-1}$, we get that $\mathbb{A} = I$, and thus $\mathrm{cond}(\mathbb{A}) = 1$. Thus there is no limit to how well-conditioned a basis can be. □

Having proved this, we go on to observe specific condition numbers for the Gram matrix of different bases.

3.2. The condition number of the Bernstein basis

In this section we consider , the Bernstein basis for $\mathcal{P}_r(T_0)$, i.e. r-th degree polynomials on the reference simplex T_0, as defined in Definition 2.21. Because its condition number has already been examined and exactly defined in **[16]**, we need not prove anything about it, but we will illustrate and validate our approach towards calculating the Gram matrix $\left\{ (\phi_i, \phi_j)_{L_2(T_0)} \right\}_{i,j}$ for certain given bases $\{\phi_i\}_i$ of V_h. Since this basis is normal ($\int_{T_0} B_j^{T_0} \mathrm{d}x = 1$), it will hopefully have a very evenly valued Gram matrix with a low condition number.

THEOREM 3.5. *The Bernstein basis Gram matrix has the form*

$$\int_T B_i B_j \mathrm{d}x = \frac{|i|!|j|!}{\underline{i}_0! \underline{j}_0!} \mathsf{lsum}(i,j,n) \tag{3.2.1}$$

[3] $\|x\|^2 = 1 \Leftrightarrow \left(\sum_i c_i e_i, \sum_j c_j e_j \right) = 1 \Leftrightarrow \sum_{i,j} c_i c_j (e_i, e_j) = 1 \Leftrightarrow \sum_{i,j} c_i c_j \Leftrightarrow \sum_i c_i^2 = 1$

where $i, j \in \mathbb{N}_0^{0:n}$, $|i| = |j| = r$ *and*

$$\mathsf{lsum}(i,j,n) := \sum_{|l|\leq i_0} \frac{(-1)^{|l|}\left(\underline{i}_0 + l + \underline{j}_0\right)!}{(i_0 - |l|)!l!(|\underline{i}_0 + l| + |j| + n)!}.$$

Proof. We know from [**16**, Lem. 1,(3)] that

$$\int_T x^i B_j \mathrm{d}x = \frac{(i+\underline{j}_0)!}{\underline{j}_0} \frac{|j|!}{(|i|+|j|+n)!}. \tag{3.2.2}$$

We can then compute the Gram matrix of the basis $\{B_i\}_{|i|=r}$ as follows:

$$\int_T B_i B_j \mathrm{d}x = \int_T \binom{|i|}{i}\left(1 - \sum_i x_i\right)^{i_0} x^{\underline{i}_0} B_j \mathrm{d}x = \sum_{|l|\leq i_0} \binom{i_0}{l}(-1)^{|l|}\binom{|i|}{i}\int x^{\underline{i}_0 + l} B_j \mathrm{d}x$$
$$= \binom{|i|}{i}\sum_{|l|\leq i_0}(-1)^l \binom{i_0}{l}\int x^{\underline{i}_0+l} B_j \mathrm{d}x.$$

We then apply (3.2.2) and get

$$= \frac{|i|!}{i!}\sum_{|l|\leq i_0}(-1)^{|l|}\frac{i_0!}{l!(i_0-|l|)!}\frac{(\underline{i}_0 + l + \underline{j}_0)!}{\underline{j}_0!}\frac{|j|!}{(|\underline{i}_0 + l| + |j| + n)!}.$$

Moving all factors independent of l out of the sum, we get

$$\int_T B_i B_j \mathrm{d}x = \frac{|i|!|j|!}{\underline{i}_0!\underline{j}_0!}\sum_{|l|\leq i_0}(-1)^{|l|}\frac{\left(\underline{i}_0 + l + \underline{j}_0\right)!}{l!(i_0-|l|)!(|\underline{i}_0 + l| + |j| + n)!}.$$

□

We can use this information to calculate the condition number of the matrix $\left\{(B_i, B_j)_{L_2(\Omega)}\right\}_{|i|,|j|=r}$ for $i, j \in \mathbb{N}_0^{0:n}$. For later, we'll use the term `lsum` to express the factor

$$\texttt{lsum}(i,j,n) := \sum_{|l|\leq i_0}(-1)^{|l|}\frac{\left(\underline{i}_0 + l + \underline{j}_0\right)!}{l!(i_0-|l|)!(|\underline{i}_0 + l| + |j| + n)!}.$$

This very well illustrates our method for obtaining the values of the different Gram matrices, and we've made a control program BernsteinCondsControl.m which checks out that we get same result as the following consequence of [**16**, Th. 3]:

Fact 3.6. *The Condition number of the matrix* $\left\{(B_i, B_j)_{L_2(\Omega)}\right\}_{|i|,|j|=r}$ *where* $|i| = |j| = r$ *is*

$$\binom{2r+n}{r}.$$

The lower and upper bounds for this expression are

$$\exp\left[\frac{-n(n-1)}{8r}\right]\frac{2^{r+\frac{n}{2}}}{\left(\left(r+n+\frac{1}{2}\right)\pi\right)^{\frac{1}{4}}} \leq \sqrt{\binom{2r+n}{r}} \leq \exp\left[\frac{-n(n-1)}{8(r+n)}\right]\frac{2^{r+\frac{n}{2}}}{\left((r+n)\,\pi\right)^{\frac{1}{4}}}.$$

The condition numbers for $n, r \leq 7$ *are shown in Table 1 on page 33.*

The upper and lower bounds imply that $\sqrt{\binom{2r+n}{r}}$ increases exponentially in r, but not exponentially in n. As we can see in the table, the expression increases linearly in n for $r = 1$, along triangle numbers $10, 15, 21, 28, \ldots$ when $r = 2$, and polynomially after that.

3.3. The barycentric basis

We will proceed to consider the barycentric basis defined in Subsection 2.4.6, and how to calculate its condition number, to compare it with the condition numbers of the Bernstein basis. As opposed to the Bernstein basis, this basis is not normalised, leaving $\int_{T_0} \lambda^i \mathrm{d}x = \frac{1}{\binom{|i|}{i}}$, which will cause a great deal more variation in the magnitude of the elements of the Gram matrix $\int \lambda^i \lambda^j \mathrm{d}x$

COROLLARY 3.7. *Barycentric Gram matrix*
Let $\left\{\lambda^i\right\}_{|i|=r}$ be the barycentric basis of $\mathcal{P}_r(T_0)$ as defined in Definition Definition 2.20. We then know that the Gram matrix for this basis has the form

$$\int \lambda^i \lambda^j \mathrm{d}x = i_0! j_0! \mathsf{lsum}(i, j, n) \tag{3.3.1}$$

with lsum *defined as in Theorem 3.5.*

PROOF. We know from Definition Definition 2.21 that

$$B_i B_j = \binom{|i|}{i} \lambda^i \binom{|j|}{j} \lambda^j.$$

Thus

$$\int \lambda^i \lambda^j \mathrm{d}x = \frac{1}{\binom{|i|}{i}\binom{|j|}{j}} \int_T B_i B_j \mathrm{d}x$$

which together with Theorem 3.5 proves the corollary. □

To find approximate solutions of the condition numbers of the barycentric basis, we have written a few programs, which can be found in Section A.2. The results of these calculations can be seen in Table 2 on page 33. We can there draw the conclusion that the barycentric basis is (for $n, r \leq 7$) worse conditioned than the Bernstein basis. Thus we can conclude that the normalisation coefficient $\binom{|i|}{i}$ in front of λ^i is well justified for all the barycentric polynomials.

3.4. The Subsimplex nodal bases

In this section we will look at the condition number for the nodal bases of the subsimplex nodes defined in [**2, 3**]:

$$\mathcal{D}_r^{\mathrm{bar}}(T_0) = \left\{ \int_f \mathrm{Tr}(u) \left(\lambda^f\right)^i \mathrm{d}x \,\middle|\, f \in \Delta(T),\, i \in \mathbb{N}_0^{0:n}, |i| = r - \dim f - 1 \right\} \tag{3.4.1}$$

$$\mathcal{D}_r^{\mathrm{ber}}(T_0) = \left\{ \int_f \mathrm{Tr}(u) \binom{|i|}{i} \left(\lambda^f\right)^i \mathrm{d}x \,\middle|\, f \in \Delta(T),\, i \in \mathbb{N}_0^{0:n}, |i| = r - \dim f - 1 \right\} \tag{3.4.2}$$

These bases for degrees of freedom for general simplices T coincide when they are joined

Both these sets have the same property: Let's take two adjacent simplices in a mesh, $T_1, T_2 \in \mathcal{T}$, adjacent meaning that $\partial T_1 \cap \partial T_2 \neq \emptyset$. According to the

definition of the mesh (Definition 2.8), their intersection $\partial T_1 \cap \partial T_2$ will be a common subsimplex $f \in \Delta(T_1) \cap \Delta(T_2)$. On this subsimplex, the elements of $\mathcal{D}(T_1)$ and $\mathcal{D}(T_2)$ restricted to f will coincide,

$$\int_f \mathrm{Tr}(u) \binom{|i|}{i} \left(\lambda^f\right)^i, \, |i| = r - \dim f - 1$$

and thus be identical. If $f = f_\sigma$ it also uniquely determines any λ^j where $|j| = r$ and $[\![j]\!] = [\![\sigma]\!]$, because

$$\lambda^j = \lambda^m \lambda^{\sum_{l \in [\![\sigma]\!]} e_l}$$

where $|m| = r - (\dim f + 1)$.

We will explain how to calculate the condition numbers of the Gram matrices of these subsimplex bases. We will do this numerically based on the Gram matrix G_ϕ of a general basis $\{\phi_i\}_{i=1}^N$ for $\mathcal{P}_r(T)$.

Given a basis $\{\phi_i\}_{i=1}^N$ where $N := \dim \mathcal{P}_r(T_0)$, we will calculate the nodal basis $\{\psi_i\}_{i=1}^N$ by finding the matrix C in $\forall i : \psi_i = \sum_{j=1}^N C_{ij}\phi_j$. Knowing that we wish the statement $\forall i, j : \delta_{ij} = n_i(\psi_j)$ to be true, we see that

$$n_i(\psi_j) = n_i(\sum_l C_{jl}\phi_l) = \sum_l C_{jl} n_i(\phi_l) = \delta_{ij}.$$

This implies that our matrix c satisifies $CJ = I$ where $J_{ji} = n_i(\psi_j)$, and consequently $C = J^{-1}$. Using this knowledge, we set out to determine $G_\psi := \{(\psi_i, \psi_j)\}_{i,j}$ and its condition number. Assuming that $G_\phi := \{(\phi_i, \phi_j)\}_{i,j}$, we see that

$$G_\psi = (\psi_i, \psi_j) = \left(\sum_l c_{il}\phi_l, \sum_m c_{jm}\phi_m\right) = \sum_{l,m} c_{il} \left(\phi_l, \phi_m\right) c_{jm} = C G_\phi C^t.$$

This way, letting $\{\phi_i\}_{i=1}^N$ be the barycentric basis, we have written our programs so that they calculate $CG_\phi C^t$ and subsequently its condition number. The results of the programs (seen in Section A.3) are shown in Table 3 and 4 on the next page. As we can clearly see, these bases are significantly worse conditioned compared to the Bernstein and barycentric basis.

3.5. Conclusion

The optimal basis out of these four bases is by far the Bernstein basis, although we have been unable to prove any general estimate of how their condition numbers develop for $n > 7$ or $r > 7$. We can safely say that the Bernstein basis is the optimal basis (out of these) to use for lower-dimensional PDE.

It is apparent that the to nodal bases differ very little in their condition numbers, and are both ill-conditioned. This migh have to do with their coefficients, and in the future, one might try to scale these nodal bases differently to achieve better condition numbers.

$n\downarrow, r\rightarrow$	1	2	3	4	5	6	7
1	3.0000	10.0000	35.0000	$1.26 \cdot 10^2$	$4.62 \cdot 10^2$	$1.72 \cdot 10^3$	$6.43 \cdot 10^3$
2	4.0000	15.0000	56.0000	$2.10 \cdot 10^2$	$7.92 \cdot 10^2$	$3.00 \cdot 10^3$	$1.14 \cdot 10^4$
3	5.0000	21.0000	84.0000	$3.30 \cdot 10^2$	$1.29 \cdot 10^3$	$5.00 \cdot 10^3$	$1.94 \cdot 10^4$
4	6.0000	28.0000	$1.20 \cdot 10^2$	$4.95 \cdot 10^2$	$2.00 \cdot 10^3$	$8.01 \cdot 10^3$	$3.18 \cdot 10^4$
5	7.0000	36.0000	$1.65 \cdot 10^2$	$7.15 \cdot 10^2$	$3.00 \cdot 10^3$	$1.24 \cdot 10^4$	$5.04 \cdot 10^4$
6	8.0000	45.0000	$2.20 \cdot 10^2$	$1.00 \cdot 10^3$	$4.37 \cdot 10^3$	$1.86 \cdot 10^4$	$7.75 \cdot 10^4$
7	9.0000	55.0000	$2.86 \cdot 10^2$	$1.36 \cdot 10^3$	$6.19 \cdot 10^3$	$2.71 \cdot 10^4$	$1.16 \cdot 10^5$

TABLE 1. The condition numbers of the Bernstein bases for $n \leq 7, r \leq 7$.

$n\downarrow, r\rightarrow$	1	2	3	4	5	6	7
1	3.0000	23.5576	$1.82 \cdot 10^2$	$1.89 \cdot 10^3$	$2.16 \cdot 10^4$	$2.67 \cdot 10^5$	$3.42 \cdot 10^6$
2	4.0000	33.3921	$6.80 \cdot 10^2$	$9.28 \cdot 10^3$	$1.80 \cdot 10^5$	$5.03 \cdot 10^6$	$1.11 \cdot 10^8$
3	5.0000	47.2211	$8.32 \cdot 10^2$	$3.53 \cdot 10^4$	$7.99 \cdot 10^5$	$2.77 \cdot 10^7$	$1.05 \cdot 10^9$
4	6.0000	57.0363	$1.01 \cdot 10^3$	$3.94 \cdot 10^4$	$3.04 \cdot 10^6$	$1.03 \cdot 10^8$	$4.88 \cdot 10^9$
5	7.0000	67.8786	$1.22 \cdot 10^3$	$4.39 \cdot 10^4$	$3.27 \cdot 10^6$	$3.94 \cdot 10^8$	$1.85 \cdot 10^{10}$
6	8.0000	79.7427	$1.37 \cdot 10^3$	$4.88 \cdot 10^4$	$3.50 \cdot 10^6$	$4.16 \cdot 10^8$	$7.11 \cdot 10^{10}$
7	9.0000	92.6248	$1.53 \cdot 10^3$	$5.42 \cdot 10^4$	$3.74 \cdot 10^6$	$4.38 \cdot 10^8$	$7.44 \cdot 10^{10}$

TABLE 2. The condition numbers of the barycentric bases for $n \leq 7, r \leq 7$.

$n\downarrow, r\rightarrow$	1	2	3	4	5	6	7
1	3.0000	$1.08 \cdot 10^3$	$1.58 \cdot 10^5$	$1.75 \cdot 10^7$	$1.13 \cdot 10^9$	$5.81 \cdot 10^{10}$	$2.49 \cdot 10^{12}$
2	4.0000	$5.25 \cdot 10^3$	$1.49 \cdot 10^6$	$7.74 \cdot 10^8$	$2.05 \cdot 10^{11}$	$3.91 \cdot 10^{13}$	$6.46 \cdot 10^{15}$
3	5.0000	$1.18 \cdot 10^4$	$1.40 \cdot 10^7$	$1.45 \cdot 10^{10}$	$1.19 \cdot 10^{13}$	$6.14 \cdot 10^{15}$	$3.05 \cdot 10^{18}$
4	6.0000	$2.03 \cdot 10^4$	$5.27 \cdot 10^7$	$1.18 \cdot 10^{11}$	$2.65 \cdot 10^{14}$	$2.25 \cdot 10^{17}$	$6.75 \cdot 10^{20}$
5	7.0000	$3.04 \cdot 10^4$	$1.31 \cdot 10^8$	$5.54 \cdot 10^{11}$	$3.03 \cdot 10^{15}$	$3.54 \cdot 10^{18}$	$7.08 \cdot 10^{21}$
6	8.0000	$4.22 \cdot 10^4$	$2.62 \cdot 10^8$	$1.79 \cdot 10^{12}$	$1.78 \cdot 10^{16}$	$1.40 \cdot 10^{20}$	$2.18 \cdot 10^{22}$
7	9.0000	$5.55 \cdot 10^4$	$4.56 \cdot 10^8$	$4.55 \cdot 10^{12}$	$4.14 \cdot 10^{16}$	$3.14 \cdot 10^{20}$	$1.31 \cdot 10^{23}$

TABLE 3. The condition numbers of the subsimplex barycentric-weighted nodal bases for $n \leq 7, r \leq 7$.

$n\downarrow, r\rightarrow$	1	2	3	4	5	6	7
1	3.0000	$1.08 \cdot 10^3$	$1.58 \cdot 10^5$	$2.60 \cdot 10^7$	$2.79 \cdot 10^9$	$2.34 \cdot 10^{11}$	$1.57 \cdot 10^{13}$
2	4.0000	$5.25 \cdot 10^3$	$1.52 \cdot 10^6$	$8.36 \cdot 10^8$	$4.20 \cdot 10^{11}$	$1.48 \cdot 10^{14}$	$4.79 \cdot 10^{16}$
3	5.0000	$1.18 \cdot 10^4$	$1.45 \cdot 10^7$	$1.50 \cdot 10^{10}$	$1.39 \cdot 10^{13}$	$1.38 \cdot 10^{16}$	$1.27 \cdot 10^{19}$
4	6.0000	$2.03 \cdot 10^4$	$5.42 \cdot 10^7$	$1.25 \cdot 10^{11}$	$2.76 \cdot 10^{14}$	$3.88 \cdot 10^{17}$	$1.56 \cdot 10^{21}$
5	7.0000	$3.04 \cdot 10^4$	$1.34 \cdot 10^8$	$5.90 \cdot 10^{11}$	$3.02 \cdot 10^{15}$	$5.13 \cdot 10^{18}$	$1.10 \cdot 10^{22}$
6	8.0000	$4.22 \cdot 10^4$	$2.67 \cdot 10^8$	$1.91 \cdot 10^{12}$	$1.15 \cdot 10^{16}$	$3.83 \cdot 10^{19}$	$2.96 \cdot 10^{22}$
7	9.0000	$5.55 \cdot 10^4$	$4.65 \cdot 10^8$	$4.84 \cdot 10^{12}$	$2.83 \cdot 10^{16}$	$6.84 \cdot 10^{20}$	$4.91 \cdot 10^{23}$

TABLE 4. The condition numbers of the subsimplex Bernstein-weighted nodal bases for $n \leq 7, r \leq 7$.

CHAPTER 4

FEM with Differential forms (Finite Element Exterior Calculus)

Differential forms are a useful, unifying tool for formulating curl- or div-based PDE and higher-dimensional antisymmetric tensor field PDE, like electromagnetic particle systems. In this chapter we will explain what alternating forms and differential forms are, and give some useful examples of spaces of these. We'll also explain how to apply the FEM to PDE in spaces of differential forms, using the Finite Element Exterior Calculus as presented in [**2**]. The notation in this chapter aspires to be parallell to [**2, 3**], but tries to correct some possible perceived ambiguity of the symbol $\mathcal{P}$ by furthering the use of $H\mathcal{P}$ from Chapter 2 when reffering to a piecewise polynomial space over a mesh.

More accurately: In Section 4.1 we describe alternating forms, so that we can describe our function spaces in Section 4.2. In Section 4.3 we see what new kind of problem we want to solve and how we formulate it, and Section 4.4 details the framework of approximation.

4.1. Alternating forms

Alternating forms are one of the two building blocks (the other being L^2-functions) of differential forms. Thus we need to know what alternating forms are before we can embark on a voyage through differential forms into the FEM and FEEC.

DEFINITION 4.1. Alternating forms

Given a n-dimensional vector space W over $\mathbb{R}$,[1] $0 \leq k \leq n$, an *alternating k-form* is a multilinear[2] function $\omega : W^k \to \mathbb{R}$ that alternates when exchanging two arguments:

$$\omega(v_1, \ldots, \underset{\uparrow}{v_i}, \ldots, \underset{\uparrow}{v_j}, \ldots, v_k) = -\omega(v_1, \ldots, \underset{\uparrow}{v_j}, \ldots, \underset{\uparrow}{v_i}, \ldots, v_k)$$

Generalized, this means that for all permutations $\sigma \in S_k$,[3]

$$\omega(v_1, \ldots, v_k) = (\operatorname{sign}\sigma)\omega(v_{\sigma(1)}, \ldots, v_{\sigma(k)})$$

We define the space $\operatorname{Alt}^k(W)$ to be the space of such alternating k-forms (k-aric alternating forms) over the space W. We write $\operatorname{Alt}(W)$ for $\bigcup_{k=0}^{n} \operatorname{Alt}^k(W)$.

---

[1] In our case (for the following sections), $W = \mathbb{R}^n$.

[2] Linear in each argument.

[3] S_k is the group of all permutations on k elements, and every $\sigma \in S_k$ is thus a unique injective function $\sigma : \{1, \ldots, k\} \to \{1, \ldots, k\}$. $\operatorname{sign}\sigma := (-1)^m$ where m is the number of transpositions (swapping of two positions) that σ can be split up into.

These alternating forms are a generalisation of $\overbrace{n \times \cdots \times n}^{k \text{ times}}$ antisymmetric tensors (k-dimensional matrices), where the arguments $v_1, \ldots, v_k$ symbolise the indices of the matrix. Observe that any underlying coordinate system for W is not mentioned in this abstraction, therefore differential forms are often referred to as "independent of coordinates" or something similar. We will define our alternating forms with the aid of bases (and thus coordinates) for $\mathrm{Alt}^k(W)$. It is important to remember that we don't always need to use specific bases. This is an important tool when working with proofs on an abstract level (which will not be done here). The space $\mathrm{Alt}^k(W)$ also has an inner product

DEFINITION 4.2. Inner product of alternating k-forms

$$(\omega, \eta)_{\mathrm{Alt}^k(W)} := \sum_{\sigma \in \Sigma[1:k;1:n]} \omega(b_{\sigma(1)}, \ldots, b_{\sigma(k)}) \eta(b_{\sigma(1)}, \ldots, b_{\sigma(k)}) \tag{4.1.1}$$

where $\omega, \eta \in \mathrm{Alt}^k(W)$, and $\{b_i\}_{i=1}^n$ is any orthonormal basis for W.

Accompanying the alternating forms is the wedge product, a generalisation of the cross, dot and scalar product of vectors to general alternating forms. Applying it to two alternating forms will produce a third one:

DEFINITION 4.3. Wedge product

Given $\omega \in \mathrm{Alt}^k(W)$ and $\nu \in \mathrm{Alt}^l(W)$, the *wedge product* or *exterior product* $\wedge : \mathrm{Alt}^k(W) \times \mathrm{Alt}^l(W) \to \mathrm{Alt}^{k+l}(W)$ is defined as

$$\omega \wedge \nu\left(v_1, \ldots, v_{k+l}\right) := \sum_{\sigma \in \mathcal{S}_{k,l}} \operatorname{sign}(\sigma) \omega\left(v_{\sigma(1)}, \ldots, v_{\sigma(k)}\right) \nu\left(v_{\sigma(k+1)}, \ldots, v_{\sigma(k+l)}\right).$$

Here $\mathcal{S}_{k,l}$ is the space of permutations $\sigma \in S_{k+l}$ that are (k,l)-increasing, meaning that $\forall i, j \geq k+1$ and $\forall i, j \leq k$: $\quad i < j \Rightarrow \sigma(i) < \sigma(j)$.

In fact, according to [**4**, Prop. 4.1.2] we have a basis for every $\mathrm{Alt}^k(W)$:

FACT 4.4. *Basis for* $\mathrm{Alt}^k(W)$

Given a vector space W over $\mathbb{R}$ with dimension $n < \infty$, there exists a basis $\mathrm{d}y_i$ for W^* such that

$$\left\{\mathrm{d}y_{\sigma(1)} \wedge \cdots \wedge \mathrm{d}y_{\sigma(k)}\right\}_{\sigma \in \Sigma(1:k,1:n)}$$

is a basis for $\mathrm{Alt}^k(W)$.

This results in an algebra of forms with such calculations as $\mathrm{d}x \wedge \mathrm{d}y(v_1, v_2) = \mathrm{d}x(v_1)\mathrm{d}y(v_2) - \mathrm{d}y(v_1)\mathrm{d}x(v_2)$.

DEFINITION 4.5. An orthonormal basis for $\mathrm{Alt}^k(W)$

Supposing that W has an orthonormal basis $\{e_i\}_{i=1}^n$, then $\{\mathrm{d}x_i\}_{i=1}^n$ is a basis for the dual space of W, i.e. $\mathrm{Alt}^0(W)$. Similarly, the wedgings of such forms form a basis (an orthonormal one) for $\mathrm{Alt}^k(W)$:

$$\operatorname{span}\left\{\mathrm{d}x_{\sigma(1)} \wedge \cdots \wedge \mathrm{d}x_{\sigma(k)} | \sigma \in \Sigma(1:k;1:n)\right\} = \mathrm{Alt}^k(W).$$

We will use this basis as a way to express an alternating form by components,

$$\omega = \sum_{\sigma\in\Sigma(1:k,1:n)} \omega_\sigma \mathrm{d}x_{\sigma(1)} \wedge \cdots \wedge \mathrm{d}x_{\sigma(k)}$$

where $\omega_\sigma \in \mathbb{R}$ to better show the parallell to antisymmetric matrices. There is a reason for the notation $\mathrm{d}x$ which will become apparent in Definition 4.8. We also want to be able to write expressions as above in a more concise way, hence we introduce:

Definition 4.6. Increasing multi-indices

With *increasing multi-indices* we apply the brevity of multiindices to increasing indices. Letting $\sigma \in \Sigma(j:k, m:n)$, a series of (comma-separated) vectors in $\mathbb{R}^n$ is written

$$v_\sigma := (v_{\sigma(j)}, \ldots, v_{\sigma(k)}).$$

If σ is the identity ($\forall i : \sigma(i) = i$), we just write v. A subset of an orthonormal basis $\{e_i\}_{i\in[\![\sigma]\!]} \subseteq \{e_i\}_{i=1}^n$ for $\mathbb{R}^n$ is thus written $e_\sigma := (e_{\sigma(j)}, \ldots, e_{\sigma(k)})$.

When we have an alternating form generated from a dual basis $\mathrm{d}y_i$ of $\mathbb{R}^n$ we will write

$$\mathrm{d}y_\sigma := \mathrm{d}y_{\sigma(j)} \wedge \cdots \wedge \mathrm{d}y_{\sigma(k)}.$$

Consequently, we get the notation

$$\mathrm{d}y_\sigma(v) = \mathrm{d}y_{\sigma(j)} \wedge \cdots \wedge \mathrm{d}y_{\sigma(k)}(v_j, \ldots, v_k).$$

Lastly, we have the notation $\underline{\sigma_i}$ which is used to denote

$$\mathrm{d}y_{\underline{\sigma_i}} := \mathrm{d}y_{\sigma(j)} \wedge \cdots \wedge \widehat{\mathrm{d}y_{\sigma(i)}} \wedge \cdots \wedge \mathrm{d}y_{\sigma(k)}$$

i.e. the $\nu \in \Sigma(j:k-1, m:n)$ where $[\![\nu]\!] = [\![\sigma]\!] \setminus \{\sigma(i)\}$

This is all used for brevity of notation, and we will try not to over-use it to avoid causing unintended difficulties to the reader.

It is important to remember that an alternating form is originally expressed without reference to a specific set of coordinates, so they might be expressed with any basis, such as the one below, which we will use later on:

Definition 4.7. Barycentric alternating forms

The alternating forms $\mathrm{d}\lambda_i^T$ related to a particular simplex $T = [t_0, \ldots, t_n]$ in W are defined as the dual of the gradients of the barycentric coordinates. Given $v \in W$,

$$\mathrm{d}\lambda_i^T(v) := \sum_{i=1}^n \frac{\partial \lambda_i^T}{\partial x_i} v_i = D\lambda_i^T \cdot v$$

so $\mathrm{d}\lambda_i^T$ is the dual of the vector $D\lambda_i^T = t_i - t_0$.

Similarly to barycentric coordinates in Definition 2.18, $\mathrm{d}\lambda_i$ are the barycentric differential forms corresponding to the reference simplex (i.e. $\mathrm{d}\lambda_i^{T_0}$).

Henceforth, we will dispense with W and write $\mathbb{R}^n$ wherever W would otherwise appear.

4.2. Differential forms, function spaces

4.2.1. Differential forms. *Differential k-forms* are functions $u : \Omega \to \mathrm{Alt}^k(\mathbb{R}^n)$, where $\Omega \subseteq \mathbb{R}^n$.[4] This is parallell to tensor fields of antisymmetric tensors, and the space of differential k-forms on Ω is written $\Lambda^k(\Omega)$. We write $\Lambda(\Omega)$ for the collection of all differential forms over Ω, $\bigcup_{k=0}^n \Lambda^k(\Omega)$.

In this section we want to expand the notions in Section 2.1 from scalar function spaces to function spaces of differential forms. First, we need to simplify our notation: According to Fact 4.4 we can then write the complete evaluation of u as a sum of components u_σ

$$u(x)(v_1, \dots, v_k) = \sum_{\sigma \in \Sigma(1:k, 1:n)} u_\sigma(x) \mathrm{d}x_{\sigma(1)} \wedge \cdots \wedge \mathrm{d}x_{\sigma(k)}(v_1, \dots, v_k).$$

With shorter notation, as in Definition 4.6, this becomes

$$u(x)(v) = \sum_{\sigma \in \Sigma(1:k, 1:n)} u_\sigma(x) \mathrm{d}x_\sigma(v)$$

for all $v_1, \dots, v_k \in \mathbb{R}^n$, $v = (v_1, \dots, v_k)$ and $x \in \Omega$.

4.2.2. L^2-spaces of differential forms. We now want to establish an L^2-space of differential k-forms on Ω, called $L^2\Lambda^k(\Omega)$. Now, integrals of k-forms are a rather complex matter, involving integration over a k-dimensional subset of Ω, as you will see in the next subsection. We will first define the inner product of $L^2\Lambda^k(\Omega)$ as

$$(u, v)_{L^2\Lambda^k(\Omega)} := \sum_{\sigma, \nu \in \Sigma(1:k, 1:n)} (u_\sigma, v_\nu) \left(\mathrm{d}x_\sigma, \mathrm{d}x_\nu\right)_{\mathrm{Alt}^k(\mathbb{R}^n)}.$$

Originally defined as $\sum_{\sigma,\kappa \in \Sigma(1:k,1:n)} \left(\int_\Omega u(x)(e_\sigma) v(x)(e_\sigma) \mathrm{d}x\right)$, in the formulation above one can clearly see that this is an analogue to the inner product of vectors, $\int u \cdot v \mathrm{d}x$. The inner product of u by itself is then the square sum of its components. We can then define the space of square integrable functions as

$$L^2\Lambda^k(\Omega) = \left\{ u \in \Lambda^k(\Omega) \,\middle|\, (u, u)_{L^2\Lambda^k(\Omega)} < \infty \right\}.$$

Let $L^2\Lambda(\Omega) = \bigcup_{k=0}^n L^2\Lambda^k(\Omega)$.

4.2.3. Integration of differential forms. The integral of a k-form $u = \sum_{\sigma \in \Sigma(1:k,1:n)} u_\sigma \mathrm{d}x_\sigma$ over Ω is defined w.r.t. a k-dimensional subset of Ω:

$$\int_f u$$

It is usually determined by a mapping to the reference simplex, including the Jacobian and such, as in [**4**, 4.4]. If the $\mathrm{d}x_i$ are orthogonal, we're in $\mathbb{R}^n$ and $f \subset \mathbb{R}^n$ is a k-simplex, the definition of will suffice to say that

$$\int_f \left(u_\sigma \mathrm{d}x_{\sigma(1)} \wedge \cdots \wedge \mathrm{d}x_{\sigma(k)}\right) := \int_f u_\sigma \mathrm{d}x_{\sigma(1)} \cdots \mathrm{d}x_{\sigma(k)}.$$

I.e. the integral of u_σ over f as seen from the σ-subplane of $\mathbb{R}^n$.

This way, integration of a differential form is quite simply and elegantly performed.

[4] In our case, $\Omega \subset\subset \mathbb{R}^n$ is a domain with polyhedral boundary, so that it can be deconstructed into a mesh.

4.2.4. Derivation on $\Lambda(\Omega)$. First of all, we further the use of the weak derivative, as defined in Section 2.1. Letting $u \in L^2\Lambda^k(\Omega)$, its weak derivative with respect to $x_i \in \mathbb{R}^n$ is just the weak derivatives of its components u_σ,

$$\frac{\partial u}{\partial x_i} = \sum_{\sigma \in S_k} \frac{\partial u_\sigma}{\partial x_i} \mathrm{d}x_\sigma.$$

Our intention for using differential forms is to define operators that differentiate only certain components of differential k-forms and produce new differential $(k+1)$-forms, for example $\mathrm{div} = D\cdot$ or $\mathrm{curl} = D\times$ in three dimensions. For this use, we have the *exterior derivative*:

DEFINITION 4.8. Exterior derivative
Given a differential form $u \in L^2\Lambda^k(\Omega)$, we have

$$\mathrm{d}u := \sum_{i=1}^{n} \frac{\partial}{\partial x_i} u \wedge \mathrm{d}x_i \tag{4.2.1}$$

which is a $k+1$-form.

The *space of* d*-differentiable differential k-forms* is

$$H\Lambda^k(\Omega) := \left\{ u \in L^2\Lambda^k(\Omega) \,\middle|\, \mathrm{d}u \in L^2\Lambda^{k+1}(\Omega) \right\}.$$

We will write $H\Lambda(\Omega)$ for $\bigcup_{k=0}^{n} H\Lambda^k(\Omega)$.

After this definition, the reason for writing of the basis for $\mathrm{Alt}^1(\Omega)$ as $\mathrm{d}x_i$ becomes apparent by letting $u = x_i \in \mathbb{R}^n$ in (4.2.1), making $\mathrm{d}x_i(v) = e_i \cdot v$ for $v \in \mathbb{R}^n$. We see

One can also see that $\mathrm{d}|_{\Lambda^0}$ has all the same properties as D, and also $\mathrm{d}|_{\Lambda^{n-1}}$ is similar to $(D\cdot) = \mathrm{div}$. In two and three dimensions, $\mathrm{d}|_{\Lambda^1}$ is similar to curl. Thus, when differentiating with d, we are only concerned that the components can be differentiated,

$$\frac{\partial u_\sigma}{\partial x_i} \in L^2\Lambda^k(\Omega),$$

where $i \notin [\![\sigma]\!]$ for $u = \sum_\sigma u_\sigma \mathrm{d}x_\sigma$. As a consequence of this and Theorem 2.3, every component u_σ must be continuous along the ith axis for all $i \notin [\![\sigma]\!]$.

4.3. Variational problems formulated with differential forms

We already have the basic idea of variational formulations from Section 2.2:

> We have a Hilbert space V, the bilinear form a and the functional l, all with the same properties as before. Find u such that

$$\forall v \in V : a(u, v) = l(v). \tag{4.3.1}$$

Assuming that $V \subseteq \Lambda^k$ is a normed vector space of differential forms, for example $L^2\Lambda^k(\Omega)$, we can easily apply Galerkin's method from Section 2.3 since the variational problem has the same form. There is perhaps a need to give an example showing that differential forms also can be used to formulate variational problems:

EXAMPLE 4.9. Example of PDE formulated with differential forms
Letting $u, v \in H_0^1\Lambda^k(\Omega)$ (u, v are 0 on $\partial\Omega$), we can define the bilinear form

$$a(u, v) := (\mathrm{d}u, \mathrm{d}v)_{L^2\Lambda^{k+1}(\Omega)} + (u, v)_{L^2\Lambda^{k-1}(\Omega)}$$

and the linear functional

$$l(v) := (f, v)_{L^2\Lambda^k(\Omega)}$$

where $f \in H\Lambda^k(\Omega)$. Given these definitions we have a weakly formulated PDE,

$$\forall v \in H^1\Lambda^k(\Omega) : (\mathrm{d}u, \mathrm{d}v)_{L^2\Lambda^{k+1}(\Omega)} + (u, v)_{L^2\Lambda^k(\Omega)} = (f, v)_{L^2\Lambda^k(\Omega)} . \tag{4.3.2}$$

Showing that this is a parallell to a PDE requires that we do an integration by parts, which works on differential forms according to [**2**, p. 16]. We will then get

$$\forall v \in \Lambda^k(\Omega) : (\delta\mathrm{d}u, v)_{L^2\Lambda^k(\Omega)} + (u, v)_{L^2\Lambda^k(\Omega)} = (f, v)_{L^2\Lambda^k(\Omega)}$$

where δ is the formal adjoint to d in the inner product of $L^2\Lambda(\Omega)$.[5] We of course have to assume that $\delta\mathrm{d}u \in L^2\Lambda^k(\Omega)$ to make this integration by parts work. Dispersing with the inner products, we can see that this is a weak formulation of the PDE:

Given $f \in \Lambda^k(\Omega)$, find $u \in \Lambda^k(\Omega)$ such that

$$(\delta\mathrm{d} + I)\, u = f.$$

where I is the identity.

Since a is bilinear, symmetric, coercive ($(\mathrm{d}u, \mathrm{d}u) + (u, u) \geq (u, u)$) we need to show that it is bounded:

$$(\mathrm{d}u, \mathrm{d}u)_{L^2\Lambda^{k+1}(\Omega)} \leq C \left\|\mathrm{d}u\right\|_{L^2\Lambda^{k+1}(\Omega)} \left\|\mathrm{d}v\right\|_{L^2\Lambda^{k+1}(\Omega)}$$

which is $\leq C\hat{C} \left\|u\right\|_{L^2\Lambda^k(\Omega)} \left\|v\right\|_{L^2\Lambda^k(\Omega)}$ by the Poincaré inequality. Because of this, we can apply Theorem 3.4 to conclude that we also in this case can focus on improving the condition number of the bases of $V_h \subset L^2\Lambda^k(\Omega)$. This goes for all problems with the same properties for the bilinear form a.

4.4. Constructing a new V_h

In this section, we will construct a subspace V_h of $V = H\Lambda^k(\Omega)$ so that we can come up with an approximate solution for solvable versions of (4.3.1) on spaces of differential forms. We will construct it so that when $V = H\Lambda^0(\Omega)$ it will coincide with the case for scalar equations in Section 2.4. We still have the same requirements, that the subspace V_h

- Converges towards our solution when refining the parameters,
- Generates a sparse stiffness matrix, and
- Uses a minimal amount of operations to do so.

Since our domain Ω has not changed any since the scalar case, the mesh definition of $\mathcal{T}$ will remain the same as Definition 2.8.

4.4.1. Shape functions and continuity. Due to our change in function spaces, from $H^1(\Omega)$ to $H\Lambda(\Omega)$, we must define our shape functions slightly differently. They will still be piecewise functions over our mesh $\mathcal{T}$, and they will be continuous just on "orthogonal" components ($H\Lambda(\Omega)$). Again we choose polynomials as our shape functions.

[5] See more on *the coderivative operator* δ in [**2**, p. 18].

DEFINITION 4.10. Polynomial differential forms

Let $r \geq 1$, $0 \leq k \leq n$ be integers, then the first space of polynomial differential k-forms over the domain $T \subset\subset \mathbb{R}^n$ is defined as

$$\mathcal{P}_r\Lambda^k(T) := \left\{ u \in L^2\Lambda^k(T) \,\middle|\, u = \sum_\sigma u_\sigma \mathrm{d}x_\sigma,\, \forall \sigma \in \Sigma(1:k, 1:n) : u_\sigma \in \mathcal{P}_r(T) \right\}.$$

Note that $\dim \mathcal{P}_r\Lambda^k(T) = \dim \mathcal{P}_r(T) \dim \Lambda^k(\mathbb{R}^n) = \binom{n+r}{n}\binom{n}{k} = \binom{r+k}{r}\binom{n+r}{n-k}$ according to [**2**, (3.1)].

As in Subsection 2.4.2 we cannot really take the direct sum $\bigoplus_{T\in\mathcal{T}} \mathcal{P}_r\Lambda^k(T)$ to get a subspace of $H\Lambda^k(\Omega)$, as this would violate the continuity condition of Theorem 2.3. Hence we need to restrict this space properly so that we get continuity of the right components. Letting $\mathcal{T}$ be a mesh over Ω, we define

$$H\mathcal{P}_r\Lambda^k(\mathcal{T}) := \left\{ u \in H\Lambda^k(\Omega) \,\middle|\, \forall T \in \mathcal{T} : u|_T \in \mathcal{P}_r\Lambda^k(T) \right\}.$$

Unfortunately, according to [**1**] such spaces aren't sufficent to produce numerically stable methods. For instance, when $u \in H(\mathrm{div}, \Omega; \mathbb{R}^3)$ a polynomial structure which is entirely in $H^1(\Omega)$ will according to [**1**, 3] produce an unstable solution with oscillations. The article instead introduces a space of intermediate polynomials which allows for the same kind of discontinuity as exists in $H(\mathrm{div}, \Omega; \mathbb{R}^3)$, which gives numerical stability, and gives a general space, $\mathcal{P}_r^-\Lambda^k(T)$ of these polynomials. This space is constructed with the aid of the following operator:

DEFINITION 4.11. The Koszul operator

The *Koszul operator* $\kappa : \Lambda^k(\Omega) \to \Lambda^{k-1}(\Omega)$ is defined by

$$\kappa(\omega)(x)(v_1, \ldots, v_k) := \omega(x)(x, v_1, \ldots, v_{k-1}).$$

What this will do with a given basis $\mathrm{d}y_\sigma \in \mathrm{Alt}^k(\mathbb{R}^n)$ is

$$\kappa(\mathrm{d}y_\sigma) = \sum_{i=1}^{k} (-1)^i y_{\sigma(i)} \mathrm{d}y_{\underline{\sigma}_i}.$$

We can see that as a consequence, when applying the Koszul operator to a differential form u, it adds a polynomial degree in orthogonal directions. Thus if we apply it to the space of homogeneous polynomial k-forms , $\mathcal{H}_r\Lambda^k(\mathbb{R}^n)$ we get a set of $r+1$-degree homogeneous polynomial k-forms where the degree of orthogonal polynomials is at least 1. When we add this together with $\mathcal{P}_r\Lambda^k(\mathbb{R}^n)$ we get the following space:

DEFINITION 4.12. The intermediary polynomials $\mathcal{P}_r^-$

We must also define a second polynomial space that will fit in the $H\Lambda(\Omega)$ space. We therefore define the space of intermediary polynomials over $T \subset\subset \mathbb{R}^n$ as

$$\mathcal{P}_r^-\Lambda^k(T) := \left\{ u \in \mathcal{P}_r\Lambda^k(T) \,\middle|\, \kappa u \in \mathcal{P}_r\Lambda^{k-1}(T) \right\}.$$

We can also express it as the direct sum

$$\mathcal{P}_r^-\Lambda^k(\Omega) = \mathcal{P}_{r-1}\Lambda^k(\Omega) + \kappa\mathcal{H}_{r-1}\Lambda^k(\Omega).$$

where $\mathcal{H}_r\Lambda^k(\Omega)$ is the space of all homogeneous polynomial k-forms like $x^i \mathrm{d}x_\sigma$ where $|i| = r$ and $\sigma \in \Sigma(1:k, 1:n)$.

According to[**2**, (3.15)] $\dim \mathcal{P}_r^- \Lambda^k(\Omega) = \binom{r+k-1}{k}\binom{n+r}{n-k}$. We may also define the space of piecewise intermediary polynomials over the mesh $\mathcal{T}$ as

$$H\mathcal{P}_r^- \Lambda^k(\mathcal{T}) := \left\{ u \in H\Lambda^k(\Omega) \,\middle|\, \forall T \in \mathcal{T} : u|_T \in \mathcal{P}_r^- \Lambda^k(T) \right\}.$$

[**2**, p. 60-61] proves that this space is well-defined, and at any subsimplex $f \in \Delta_j(\mathcal{T})$, the trace $\mathrm{Tr}_f(p)$ for is single-valued for $k \leq j \leq n-1$ for $p \in H\mathcal{P}_r\Lambda^k(\mathcal{T})$ or $p \in H\mathcal{P}_r^-\Lambda^k(\mathcal{T})$. We need this result when introducing the degrees of freedom in the next subsection.

4.4.2. Degrees of Freedom. Within these spaces $H\mathcal{P}_r^-\Lambda^k(\mathcal{T})$ and $H\mathcal{P}_r\Lambda^k(\mathcal{T})$ we want to create bases that have local support in the sense presented in Subsection 2.4.4. This can be done with great success through the use of Degrees of Freedom based upon local evaluations, i.e. integrals over only part of the domain Ω. This can be a tool to let us construct very local nodal bases, giving us a very sparse stiffness matrix. In [**2**, (5.1) p. 59] the dual space of $\mathcal{P}_r\Lambda^k(T)$ is defined by functionals of the form

$$\int_f \mathrm{Tr}_f u \wedge v \mathrm{d}x,\ v \in \mathcal{P}^-_{r-j+k}\Lambda^{j-k}(f),\ f \in \Delta_j(T)$$

The dual space of $\mathcal{P}_r^-\Lambda^k(T)$ in turn may be spanned by the functionals

$$\int_f \mathrm{Tr}_f u \wedge v \mathrm{d}x,\ v \in \mathcal{P}_{r-j+k-1}\Lambda^{j-k}(f),\ f \in \Delta_j(T).$$

Using these sets, [**2**, 5.1] tells us that they uniquely respectively determine any u in $H\mathcal{P}_r\Lambda^k(\mathcal{T})$ or $H\mathcal{P}_r^-\Lambda^k(\mathcal{T})$. For a more practical approach, one needs to define the bases of these spaces. The basis for $H\mathcal{P}_r\Lambda^k(\mathcal{T})^*$ we define as

$$\mathcal{D}_r\Lambda^k(T) = \left\{ \int_f \mathrm{Tr}(u)\psi_i^f \mathrm{d}x \,\middle|\, f \in \Delta(T),\ i \leq \dim \mathcal{P}^-_{r-j+k}\Lambda^{j-k}(f) \right\} \tag{4.4.1}$$

where $\left\{\psi_i^f\right\}_i$ is a basis for $\mathcal{P}^-_{r-j+k}\Lambda^{j-k}(f)$ and $j = \dim f$. We extend this to the entire mesh to get the basis

$$\mathcal{D}_r\Lambda^k(\mathcal{T}) = \left\{ \int_f \mathrm{Tr}(u)\psi_i^f \mathrm{d}x \,\middle|\, f \in \Delta(\mathcal{T}),\ i \leq \dim \mathcal{P}^-_{r-j+k}\Lambda^{j-k}(f) \right\}. \tag{4.4.2}$$

We can construct a similar basis $\mathcal{D}_r^-\Lambda^k(\mathcal{T})$ for $H\mathcal{P}_r^-\Lambda^k(\mathcal{T})^*$:

$$\mathcal{D}_r^-\Lambda^k(T) = \left\{ \int_f \mathrm{Tr}(u)\xi_i^f \mathrm{d}x \,\middle|\, f \in \Delta(T),\ i \leq \dim \mathcal{P}_{r-j+k-1}\Lambda^{j-k}(f) \right\} \tag{4.4.3}$$

where $\left\{\xi_i^f\right\}_i$ is a basis for $\mathcal{P}_{r-j+k-1}\Lambda^{j-k}(f)$. We extend this to the entire mesh to get the basis

$$\mathcal{D}_r^-\Lambda^k(\mathcal{T}) = \left\{ \int_f \mathrm{Tr}(u)\xi_i^f \mathrm{d}x \,\middle|\, f \in \Delta(\mathcal{T}),\ i \leq \dim \mathcal{P}_{r-j+k-1}\Lambda^{j-k}(f) \right\}. \tag{4.4.4}$$

4.4.3. Constructing a basis. First of all, we have the nodal bases of $\mathcal{D}_r\Lambda^k(\mathcal{T})$ and $\mathcal{D}_r^-\Lambda^k(\mathcal{T})$ which are both based on integrals along certain $f \in \Delta(\mathcal{T})$. We then have a basis ϕ_i where $\forall n_i \in \mathcal{D}_r\Lambda^k(\mathcal{T}) : n_i(\phi_j) = \delta_{ij}$. This again causes the support of ϕ_i to be $\mathrm{supp}\phi_i = \bigcup \omega_f(\mathcal{T})$ for the single $f \in \Delta(\mathcal{T})$ to which n_i is associated. Thus we are able to produce local support, which in turn will generate a sparse matrix as long as the bilinear form a is based on integrals over Ω.

We will now proceed to define a basis which has a similar property (that $\text{supp}\phi_i = \bigcup \omega_f(\mathcal{T})$ for some f). This will be a generalisation of the barycentric monomial basis (see Definition 2.20).

DEFINITION 4.13. Barycentric basis for differential forms
Let $\lambda^T = \left(\lambda_0^T, \dots, \lambda_n^T\right)$ be the barycentric coordinate function for the n-simplex T, and let $\{\mathrm{d}x_\sigma\}_{\sigma\in\Sigma(1:k,1:n)}$ be a basis for $\mathrm{Alt}^k(T)$ then

$$\left\{ \left(\lambda^T\right)^i \mathrm{d}\lambda_\sigma \middle| \; i \in \mathbb{N}_r^{0:n}, |i| = r, \sigma \in \Sigma(1:k, 1:n) \right\} \tag{4.4.5}$$

which on the reference simplex T_0 is $\lambda^i \mathrm{d}x_\sigma$. This is a basis for $\mathcal{P}_r\Lambda^k(T)$.

For the space $\mathcal{P}_r^-\Lambda^k(T)$ we need to be more careful in our approach. Actually, we need the help of a basis for $\mathcal{P}_1^-\Lambda^k(T)$, the Whitney forms (named after Hassler Whitney who introduced them in [**17**, p. 228-229]):

DEFINITION 4.14. Whitney forms

On the same assumptions as the previous Definition, we establish that the Whitney k-forms are defined as

$$\left\{ \phi_\sigma := \sum_{i=0}^{k} \lambda_{\sigma(i)} \mathrm{d}\lambda_{\sigma(0)}^T \wedge \cdots \wedge \widehat{\mathrm{d}\lambda_{\sigma(i)}^T} \wedge \cdots \wedge \mathrm{d}\lambda_{\sigma(k)}^T \middle| \; \sigma \in \Sigma(0:k, 0:n) \right\}. \tag{4.4.6}$$

We can immediately see that this is the same as

$$\left\{ \kappa \mathrm{d}\lambda_\sigma^T \middle| \; \sigma \in \Sigma(0:k, 0:n) \right\},$$

and that it in that way spans $\mathcal{P}_1^-\Lambda^k(T) = \kappa\mathcal{P}_0\Lambda^{k+1}(T)$ because $\{\mathrm{d}\lambda_\sigma\}_{\sigma\in\Sigma(1:k+1,1:n)}$ spans $\mathrm{Alt}^{k+1}(\mathbb{R}^n) = \mathcal{P}_0\Lambda^{k+1}(T)$.

DEFINITION 4.15. Reduced barycentric polynomial forms
On the same assumptions as before,

$$\left\{ \left(\lambda^T\right)^i \phi_\sigma \middle| \; i \in \mathbb{N}_r^{0:n}, |i| = r, \sigma \in \Sigma(1:k, 1:n) \right\} \tag{4.4.7}$$

This spans $\mathcal{P}_r^-\Lambda^k(T)$ according to [**2**].

THEOREM 4.16. *Any $u \in \mathcal{P}_1^-\Lambda^k(T)$ is uniquely determined by $\mathcal{D}_1^-\Lambda^k(T)$.*

PROOF. Since each of ϕ_σ can be asssociated with a simplex f_σ. Since every

$$\phi_\sigma = \sum_{i=0}^{k} \lambda_{\sigma(i)} \mathrm{d}\lambda_{\sigma(0)}^T \wedge \cdots \wedge \widehat{\mathrm{d}\lambda_{\sigma(i)}^T} \wedge \cdots \wedge \mathrm{d}\lambda_{\sigma(k)}^T$$

is non-zero on any k-subsimplex f_ν because $\lambda_{\sigma(i)}^T > 0$ on the interior of f_σ, we can conclude that $\phi_\sigma|_{f_\sigma} > 0$ and thus that

$$\int_{f_\sigma} \phi_\sigma.$$

We can then assume that if $u \in \mathcal{P}_1^-\Lambda^k(T)$ and $\forall n \in \mathcal{D}_1^-\Lambda^k(T) : n(u) = 0$, $u = 0$ because$\mathcal{D}_1^-\Lambda^k(T)$ spans $\mathcal{P}_1^-\Lambda^k(T)^*$. □

The Whitney forms together with the $\mathcal{D}_1^-\Lambda^k(T)$ form a class of affine equivalent elements, which is proved in [**2**, p. 57].

4.4.4. Convergence of the method. In essence, we have convergence of solution from the fact that $\mathcal{P}\Lambda^k(T) := \sum_{r=1}^{\infty} \mathcal{P}_r\Lambda^k(T) = \sum_{r=1}^{\infty} \mathcal{P}_r^-\Lambda^k(T)$ is dense in $H\Lambda^k(T)$ and thus $H\mathcal{P}\Lambda^k(\mathcal{T})$ is dense in $H\Lambda^k(\Omega)$. The result from 2.4.7 can be applied here: Getting an estimate for the rate of convergence supposes a higher degree of differentiability of our $u \in H\Lambda^k(\Omega)$. Thus, we must suppose that the components of u, u_σ are in $H^t(\Omega)$, then the following general convergence estimate applies:

FACT 4.17. [**6**, Th. 6.4] *Assuming $\mathcal{T}$ is shape-regular*[6] *the convergence of the solution is certain with the rate*

$$(\|u - u_h| \leq) \quad \|u - \Pi_h u\|_{L_2(\Omega)} \leq ch^t \left(\sum_{|i| \leq t} \left\|D^i u\right\|^2_{L_2(\Omega)} \right)^{\frac{1}{2}}.$$

This requires that $u \in H^t(\Omega)$ and h is half the largest diameter of all $T \in \mathcal{T}$ and Π_h is interpolation by a piecewise polynomial of degree $r = t - 1 \geq 1$.

Thus for a $u_h \in H\mathcal{P}_r(\mathcal{T})$, we must suppose that it is weakly differentiable $r+1$ times, which cannot always be the case, but this is the best convergence estimate that we have for general polynomial interpolation. h is then the refining parameter for ensuring convergence, even though one might also be able to infer convergence by increasing the polynomial degree.

4.4.5. Affine equivalence. As previously mentioned, affine equivalence is our tool for reducing the number of symbolic integrations needed to be done by a factor of $|\mathcal{T}|$. Because we only need to calculate the stiffness matrix symbolically for basis functions associated to one $T \in \mathcal{T}$, we can apply this calculation to the rest of the basis functions using properties of affine transformation. More details can be found in [**7**, p. 82 etc.]. In this subsection we use the term *finite element* to denote a triple $(T, \mathcal{F}, \mathcal{N})$ consisting of a simplex T, a set of basis functions $\mathcal{F}$ spanning $V_h|_T$, and a basis $\mathcal{N}$ for$\left(V_h|_T\right)^*$. We now recount the requirements for affine equivalence.

DEFINITION 4.18. Affine equivalence

Two elements $(T_1, \mathcal{F}_1, \mathcal{N}_1)$ and $(T_2, \mathcal{F}_2, \mathcal{N}_2)$ are *affine equivalent* if there exists an affine injective transformation $L : \mathbb{R}^n \to \mathbb{R}^n$ such that the images of L, L^* and L_* are

(1) $L(T_1) = T_2$
(2) $L^*(\mathcal{F}_2) = \mathcal{F}_1$
(3) $L_*(\mathcal{N}_1) = \mathcal{N}_2$

where L^*, $L^*(f) := f \circ L$ for $f \in \mathcal{F}_2$, is called the *pullback of* L and L_*, $L_*(N) := N \circ L^*$ for $N \in \mathcal{N}_1$, is called the *push-forward of* L.

Now, the problem with affine equivalence in many finite elements over vector (and tensor) fields has been that their pullback operator only affected the functions in span$\mathcal{F}$ directly, while these were relying on an underlying coordinate system. We will describe how this problem is solved with the change from general asymmetric tensor fields to differential forms with this "parable" on vector fields:

[6][**6**, p. 61]: $\exists \kappa > 0 : \forall T \in \mathcal{T}$: the inscribed circle of has a radius $\geq h_T/\kappa$ where h_T is the length of the longest line of T.

Assuming $u : T \to \mathbb{R}^M$ for some simplex T, the degrees of freedom $N \in \mathcal{N}_T$ have the property

$$N(u) = \int_f u \cdot p \mathrm{d}x$$

on $u \in \mathrm{span}\mathcal{F}$ for some normal or tangential vector p. According to Definition 4.18 one usually defines the push-forward function L_* as

$$L_*(N)(u) = N\left(L^*(u)\right) = \int_f L^*(u) \cdot p \mathrm{d}x.$$

The problem with this definition is that it ignores the vector p, which then causes elements to be affine inequivalent because the corresponding degree of freedom for another element might look like

$$\int_g v \cdot q \mathrm{d}x$$

where q does not have to be equal to p, and therefore affine equivalence is not the case in most cases when $v = L^*(u)$ and $L(g) = f$

$$\int_f L^*(u) \cdot p \mathrm{d}x = \int_g v \cdot p \mathrm{d}x \neq \int_g v \cdot q \mathrm{d}x.$$

This is solved in formulations with differential forms, because they are inextricably linked to their integrands and relative directions, and [**2**, pp. 10,16] defines the pullback function L^* as

$$L^*(u(x)(v)) := u(L(x))\left(DL\left(v_1\right), \ldots, DL\left(v_n\right)\right)$$

where the DL is the Jacobian for L. Let one of our $N^T \in \mathcal{N}_T$ be defined as

$$N^T(u) := \int_{f_\sigma^T} u \wedge \nu, \quad \nu^T \in \mathcal{P}^-_{r-\dim f+k}\Lambda^{\dim f-k}(T)$$

for *any* simplex $T \subset \mathbb{R}^n$. For u in span$\mathcal{F}_S$ (assuming the pullback from Definition 4.18, Item 2 works) we then have

$$L_*\left(N^T\right)(u) = \int_{f_\sigma^T} L^*(u) \wedge \nu^T = \int_{f_\sigma^S} u(x)(v) \wedge \nu^S = N^S(u)$$

where $DL(v) := DL\left(v_1\right), \ldots, DL\left(v_n\right)$. we get the corresponding degree of freedom N^S for the simplex S, which provides us with affine equivalence.

This sketch of a general proof of affine equivalence is not sufficient to prove that the classes of bases for $\mathcal{P}_r\Lambda^k(T)$ and $\mathcal{P}_r^-\Lambda^k(T)$ coupled with their degrees of freedom $\mathcal{D}_r\Lambda^k(T)$ and $\mathcal{D}_r^-\Lambda^k(T)$. We will therefore prove the affine equivalence of the Whitney forms under the degrees of freedom $\mathcal{D}_1^-\Lambda^k(T)$

THEOREM 4.19. *The Whitney forms for two simplices T and S are affine equivalent w.r.t. the degrees of freedom $\mathcal{D}_1^-\Lambda^k(T)$.*

PROOF. To prove affine equivalence we need to establish the following facts from Definition 4.18:

(1) $L(T) = L(S)$
(2) $L^*(\mathcal{F}_S) = \mathcal{F}_T$
(3) $L_*(\mathcal{D}_1^-\Lambda^k(T)) = \mathcal{D}_1^-\Lambda^k(S)$

Item 1 defines the affine map $L : \mathbb{R}^n \to \mathbb{R}^n$, and proving it is thus trivial.

Proving Item 2, we take a Whitney k-form $\phi_\sigma^S \in \mathcal{F}_S$. We then take its pullback,

$$L^*\left(\phi_\sigma^S\right) = L^*\left(\sum_{i=0}^{k}(-1)^i\lambda_{\sigma(i)}^S \mathrm{d}\lambda_{\sigma(1)}^S \wedge \cdots \wedge \widehat{\mathrm{d}\lambda_{\sigma(i)}^S} \wedge \cdots \wedge \mathrm{d}\lambda_{\sigma(k)}^S(v)\right)$$

$$= \sum_{i=0}^{k}(-1)^i L\left(\lambda_{\sigma(i)}^S\right)\mathrm{d}\lambda_{\sigma(1)}^S \wedge \cdots \wedge \widehat{\mathrm{d}\lambda_{\sigma(i)}^S} \wedge \cdots \wedge \mathrm{d}\lambda_{\sigma(k)}^S(DL(v))$$

$$= \sum_{i=0}^{k}(-1)^i L^*\left(\lambda_{\sigma(i)}^S\right)\left(\mathrm{d}\lambda_{\sigma(1)}^S \circ DL\right) \wedge \cdots \wedge \widehat{\left(\mathrm{d}\lambda_{\sigma(i)}^S \circ DL\right)} \wedge \cdots \wedge \left(\mathrm{d}\lambda_{\sigma(k)}^S \circ DL\right)(v).$$

To prove that this equals ϕ_σ^T we only require $L^*\left(\lambda_{\sigma(i)}^S\right) = \lambda_{\sigma(i)}^S\left(L(x)\right) = \lambda_{\sigma(i)}^T(x)$ and $\mathrm{d}\lambda_{\sigma(j)}^S \circ DL = \mathrm{d}\lambda_{\sigma(j)}^T$ which can be easily checked.

Proving Item 3 is fortunately then quite simple: Let $N_\sigma^T \in \mathcal{D}_1^-(T)$ and $N_\sigma^S \in \mathcal{D}_1^-(S)$ be corresponding degrees of freedom. Then,

$$N_\sigma^T(u) := \int_{f_\sigma^T} \mathrm{Tr}(u \wedge 1) = \int_{f_\sigma^T} u.$$

We need not take the trace, since u is polynomial and therefore continuous. Applying the push-forward operator to N_σ^T gives us the following:

$$L_*\left(N_\sigma^T\right)(u) = \int_{f_\sigma^T} L^*(u) = \int_{f_\sigma^S} u(x),$$

the last equality according to [**2**, p. 16]. This proves that $L_*(\mathcal{D}_1^-(T)) = \mathcal{D}_1^-(S)$ □

4.4.6. Conclusion. We now have to possible choices of finite element spaces, $H\mathcal{P}_r(\mathcal{T})$ and $H\mathcal{P}_r^-(\mathcal{T})$ with their associated degrees of freedom. They are still able to converge towards a theoretical solution of our variational problem, under the same conditions as in Chapter 2. They also have a set of local bases based on barycentric coordinates of $T \in \mathcal{T}$, which create a stiffness matrix which is quite sparse. Since we have affine equivalence for our spaces (of course with respect to the degrees of freedom), $|\mathcal{T}|$ symbolic calculations can be skipped per stiffness matrix calculation, and thus our algorithm is time-efficient.

We will now repeat the process of exploring the condition numbers for our new bases for $H\mathcal{P}_r(\mathcal{T})$ and $H\mathcal{P}_r^-(\mathcal{T})$, but restricting ourselves to calculations on the reference simplex because of their affine equivalence.

We have now chosen $H\mathcal{P}_r(\mathcal{T})$ as our subspace V_h. It has basis functions with local support which generates a sparse stiffness matrix, its functions are continuous, a solution in V_h converges towards the exact solution in V, and we don't require many symbolic calculations in the process. In later chapters (3, 5) we will only consider the bases on a single T, because the calculations we are trying to optimize (stiffness matrix calculation) is done element by element.

CHAPTER 5

Condition numbers of bases in $\mathcal{P}_r\Lambda^k(T_0)$ and $\mathcal{P}_r^-\Lambda^k(T_0)$

Earlier on, in Chapter 3 we calculated the condition numbers of different bases for polynomial scalar functions on the reference simplex, $\mathcal{P}_r(T_0)$. Most of them were not analytical results as in [**16**], but approximations made on computer. It is not necessarily easy (maybe not even possible) to calculate them analytically, but doing so provides us with exact knowledge of the condition numbers for higher-dimensional PDE with any polynomial degree. For instance one might easily create system with higher dimensions than what's been calculated here in electromagnetics.

Nevertheless, the results we have found are useful, and tells us what the numerical stability of the stiffness matrix is in many cases of variational formulations of PDE. In the last chapter we described a generalisation of vector fields called differential forms. We also defined two spaces of polynomial differential forms and two sets of bases for these. Here we will calculate the Gram matrices of these bases analytically, then proceed to calculate the condition numbers of these matrices by computer.

The reader will probably observe that the tables now are now more numerous and have a triangular appearance. This is because our spaces no longer rely only on the two integers (n, r) as $\mathcal{P}_r(T_0)$, but on the three integers (n, r, k), where k varies between 0 and n giving spaces $\mathcal{P}_r\Lambda^k(T_0)$.

5.1. Some calculations of alternating forms

Before we dive into the bases, we notice that the bases from Subsection 4.4.3 look like this:

$$\lambda^i \mathrm{d}\lambda_\sigma, \quad \lambda^i\phi_\sigma = \lambda^i \sum_{l=0}^{k} (-1)\lambda_{\sigma(l)}\mathrm{d}\lambda_{\underline{\sigma}_l}. \tag{5.1.1}$$

Seeing that for either set of basis functions u, v, their Gram matrices will look like

$$\left(\left(\lambda^T\right)^\alpha \mathrm{d}\lambda_\sigma^T, \left(\lambda^T\right)^\beta \mathrm{d}\lambda_\pi^T\right)_{L^2\Lambda^k(T)} = \left(\left(\lambda^T\right)^\alpha, \left(\lambda^T\right)^\beta\right)_{L^2(T)} \left(\mathrm{d}\lambda_\sigma^T, \mathrm{d}\lambda_\pi^T\right)_{\mathrm{Alt}^k(\mathbb{R}^n)}$$

and

$$\left(\lambda^s\phi_\alpha^T, \lambda^t\phi_\beta^T\right)_{L^2\Lambda^k(T)} = \sum_{i,j=0}^{k} (-1)^{i+j} \left(\left(\lambda^T\right)^{s+e_{\alpha(i)}}, \left(\lambda^T\right)^{t+e_{\beta(j)}}\right)_{L^2(T)} \left(\mathrm{d}\lambda_{\underline{\alpha}_i}^T, \mathrm{d}\lambda_{\underline{\beta}_j}^T\right)_{\mathrm{Alt}^k(\mathbb{R}^n)}.$$

We immediately notice that there is an unknown factor $\left(\mathrm{d}\lambda_\sigma^T, \mathrm{d}\lambda_\pi^T\right)_{\mathrm{Alt}^k}$ in both of these expressions. In order to calculate them, we must first find out what $\left(\mathrm{d}\lambda_\sigma^T, \mathrm{d}\lambda_\pi^T\right)_{\mathrm{Alt}^k}$ means, which will be done in this section for $T = T_0$, i.e. the reference simplex.

The inner product of $\mathrm{Alt}^k(\mathbb{R}^n)$ is defined in Definition 4.2 as

$$(\mathrm{d}\lambda_\mu, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k(\mathbb{R}^n)} = \sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(b_\sigma)\mathrm{d}\lambda_\pi(b_\sigma)$$

where $\{b_i\}_{i=1}^n$ is any orthonormal basis for $\mathbb{R}^n$. We choose $b_i = e_i$ (the unit vectors in $\mathbb{R}^n$). The unknown term here then looks like $\mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma)$ (calculated in Corollary 5.5), with factors

$$\mathrm{d}\lambda_\pi(e_\sigma) = \sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=1}^k \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)})$$

(calculated in Lemma 5.4) that have factors $\mathrm{d}\lambda_i(e_j)$ (calculated in Lemma 5.1), and this is where we begin.

LEMMA 5.1. *Evaluating a barycentric alternating form*

Let $\{e_i\}_{i=1}^n$ be the standard orthonormal basis for $\mathbb{R}^n$, and $\mathrm{d}\lambda_i$ be the barycentric alternating 1-forms of the reference simplex defined in Definition 4.7. Then

$$\mathrm{d}\lambda_i(e_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } 0 \neq i \neq j \\ -1 & \text{if } i = 0 \end{cases}.$$

PROOF. We know from Definition 4.7 that $\mathrm{d}\lambda_i(e_j) = D\lambda_i \cdot e_j$. Since $D\lambda_0 = D(1 - \sum_i x_i) = -\sum_i e_i$ and $D\lambda_i = e_i$ for $1 \leq i \leq n$ we have the result. □

DEFINITION 5.2. The sets of separate indices

Given $\mu, \rho \in \Sigma[0:k;0:n]$, we define the *set of separate indices* for ρ with respect to μ as

$$\mathrm{si}_\nu^\rho := \rho^{-1}\left([\![\rho]\!] \setminus [\![\mu]\!]\right) \tag{5.1.2}$$

The set gives the indices i of all the $\rho(i)$ which are not in $[\![\mu]\!]$.

COROLLARY 5.3. *For all* $\mu, \rho \in \Sigma[0:k;0:n]$, $|\mathrm{si}_\nu^\rho| = |\mathrm{si}_\rho^\nu|$.

LEMMA 5.4. *Evaluating the alternating form part of Whitney forms on the reference simplex*

$$\mathrm{d}\lambda_\mu(e_\sigma) = \begin{cases} 1 & \text{if } \mu = \sigma \\ (-1)^m & \text{if } |\mathrm{si}_\mu^\sigma| = 1, \mu(1) = 0 \text{ where } m \in \mathrm{si}_\mu^\sigma \\ 0 & \text{if } \mathrm{si}_\mu^\sigma := \sigma^{-1}\left([\![\sigma]\!] \setminus [\![\mu]\!]\right) \text{ has more than } \delta_{0\mu(1)} \text{ elements} \end{cases}$$

where $\mu \in \Sigma[1:k;0:n]$ *describes a barycentric alternating form, and* $\sigma \in \Sigma[1:k;1:n]$ *describes a selection of* k *orthonormal coordinates* $\left\{e_{\sigma(i)}\right\}_{i=1}^k \subseteq \{e_i\}_{i=1}^n$ *as in Definition 4.2.*

PROOF. We have to calculate $\mathrm{d}\lambda_{\mu(1)} \wedge \cdots \wedge \mathrm{d}\lambda_{\mu(k)}(e_{\sigma(1)}, \ldots, e_{\sigma(k)})$. We know that $0 \in [\![\mu]\!] \overset{\mu\in\Sigma}{\Leftrightarrow} \mu(1) = 0$ for all increasing μ, so we will write the latter equivalent instead of the former.

Case 1. Suppose that $\mu(1) = 0$, then

$$\mathrm{d}\lambda_\mu(e_\sigma) = \sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=1}^{k} \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)})$$

and by Lemma 5.1

$$= \sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=1}^{k} \delta_{\mu(i)\nu\circ\sigma(i)} = \sum_{\nu\in S_k} (\operatorname{sign}\nu)\, \delta_{\mu,\sigma\circ\nu}$$

and since μ and σ are both increasing indices,

$$= \sum_{\nu\in S_k} (\operatorname{sign}\nu)\, \delta_{\mu\sigma}\delta_{\mathrm{id},\nu} = \delta_{\mu\sigma}.$$

Case 2. Let $0 \in [\![\mu]\!]$,

$$\mathrm{d}\lambda_\mu(e_\sigma) = \sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=1}^{k} \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)})$$

we have a factor including $\mathrm{d}\lambda_0 = \mathrm{d}\lambda_{\mu(1)}$ (since μ is increasing). Thus,

$$\mathrm{d}\lambda_\mu(e_\sigma) = \sum_{\nu\in S_k} (\operatorname{sign}\nu)\, \mathrm{d}\lambda_0(e_{\sigma\circ\nu(1)}) \prod_{i=2}^{k} \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)})$$

$$= -\sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=2}^{k} \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)}) = -\sum_{\nu\in S_k} (\operatorname{sign}\nu) \prod_{i=2}^{k} \delta_{\mu(i)\sigma\circ\nu(i)}$$

$$= -\sum_{\nu\in S_k} (\operatorname{sign}\nu)\, \delta_{\underline{\mu}_1 \underline{\sigma\circ\nu}_1} \tag{5.1.3}$$

where $\underline{\mu}_j : \{1,\dots,k\}\backslash\{j\} \to \{1,\dots,n\}$ means μ as an increasing sequence, but truncating/overlooking the jth argument. For this to be nonzero, $[\![\mu]\!] \cap [\![\sigma]\!]$ can have no more than one element, or that the si-set

$$\mathrm{si}_\mu^\sigma = \sigma^{-1}\left([\![\mu]\!] \cap [\![\sigma]\!]\right)$$

must contain only one element. Since $\underline{\mu}_1$ is increasing, $\underline{\sigma\circ\nu}_1$ must also be increasing for the terms in (5.1.3) to be nonzero, which makes all but one term vanish:

$$= -(\operatorname{sign}\nu)\, \delta_{\underline{\mu}_1 \underline{\sigma\circ\nu}_1} \tag{5.1.4}$$

That way, this term, which yields nonzero results is the one where ν shifts one argument to the first position. Thus we can utilize the element $m \in \mathrm{si}_\mu^\sigma$ which tells us how many transpositions ν must contain. This must be $(m-1)$ for $\underline{\sigma\circ\nu}_1$to be increasing and equal to $\underline{\mu}_1$, and thus

$$-(\operatorname{sign}\nu)\, \delta_{\underline{\mu}_1 \underline{\sigma\circ\nu}_1} = -(-1)^{m-1} = (-1)^m,$$

concluding the second case.

Case 3. Assume $\left|\mathrm{si}_\mu^\sigma\right| > \delta_{0\mu(1)}$. In either case, $\forall\nu \in S_n : \exists m \in \mathrm{si}_\mu^\sigma : \exists i : m = \nu(i)$ making $\mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)}) = 0$ for some i, $\Rightarrow \prod_{i=1}^{k} \mathrm{d}\lambda_{\mu(i)}(e_{\sigma\circ\nu(i)}) = 0$ for all $\nu \in S_k$.

□

We build upon this lemma to find the product of two such terms:

COROLLARY 5.5. *A product of two* $\mathrm{d}\lambda_\mu(e_\sigma)$*-terms,*

$$\mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = \begin{cases} 1 & \text{if } \mu = \pi = \sigma \\ 0 & \text{if either } \left|\mathrm{si}_\mu^\sigma\right| > \delta_{0\mu(0)} \text{ or } |\mathrm{si}_\pi^\sigma| > \delta_{0\pi(0)} \\ (-1)^m & \text{if } \mu(1) = 0, \pi = \sigma, |\mathrm{si}_\mu^\sigma| = 1 \text{ and } \exists! m \in \mathrm{si}_\mu^\sigma \\ (-1)^p & \text{if } \pi(1) = 0, \mu = \sigma, |\mathrm{si}_\pi^\sigma| = 1 \text{ and } \exists! p \in \mathrm{si}_\pi^\sigma \\ (-1)^{m+p} & \text{if } \mu(1) = \pi(1) = 0, |\mathrm{si}_\mu^\sigma| = |\mathrm{si}_\pi^\sigma| = 1 \text{ and } \exists! m \in \mathrm{si}_\mu^\sigma, p \in \mathrm{si}_\pi^\sigma \end{cases}$$

We may then finally conclude on the inner product of two barycentric alternating forms.

LEMMA 5.6. *The inner product of two barycentric alternating forms in the reference simplex*

$$(\mathrm{d}\lambda_\mu, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k} = \sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = \tag{5.1.5}$$

$$= \begin{cases} n-k+1 & \text{if } \mu(1) = 0 = \pi(1) \text{ and } \mu = \pi \\ 1 & \text{if } \mu = \pi \text{ and both } \mu(1), \pi(1) \neq 0 \\ 0 & \text{if either } \delta_{0\mu(0)}, \delta_{0\pi(0)} < \left|\mathrm{si}_\mu^\pi\right| \\ (-1)^m & \text{if } \mu(1) = 0 \neq \pi(1) \text{ and } \left|\mathrm{si}_\mu^\pi\right| = 1 \text{ where } m \in \mathrm{si}_\mu^\pi \\ (-1)^p & \text{if } \mu(1) \neq 0 = \pi(1) \text{ and } |\mathrm{si}_\pi^\mu| = 1 \text{ where } p \in \mathrm{si}_\pi^\mu \\ (-1)^{m+p} & \text{if } \mu(1) = 0 = \pi(1) \text{ and } \left|\mathrm{si}_\mu^\pi\right| = |\mathrm{si}_\pi^\mu| = 1 \text{ where } m \in \mathrm{si}_\mu^\pi \text{ and } p \in \mathrm{si}_\pi^\mu \end{cases}$$

PROOF. We will prove each case above separately. The details of increasing sequences are usually skipped. When referring to "the Corollary", we refer to the Corollary Corollary 5.5.

Case 1. If $\mu = \pi$ and $\mu(1) = 0 = \pi(1)$, then

$$\sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = \sum_{\sigma\in X} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma)$$

where $X = \{\sigma \in \Sigma(1:k;1:n) : [\![\mu]\!] \setminus \{0\} \subset [\![\sigma]\!]\}$ is the collection of all σ with $|\mathrm{si}_\mu^\sigma| = 1$. Thus, for some constants $i(\sigma)$,

$$\sum_{\sigma\in X} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = \sum_{\sigma\in X} \left(\mathrm{d}\lambda_\mu(e_\sigma)\right)^2 = \sum_{\sigma\in X} (-1)^{2i(\sigma)} = \sum_{\sigma\in X} 1 = |X| = n-k+1$$

Case 2. If $\mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma)$ will in the case $\mu = \pi$ and both $\mu(0), \pi(0) \neq 0$ be nonzero except when $\sigma = \mu = \pi$ thus by the Corollary $\sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = 1$.

Case 3. If either $\delta_{0\mu(0)}, \delta_{0\pi(0)} < \left|\mathrm{si}_\mu^\pi\right|$, then for all σ either $\left|\mathrm{si}_\mu^\sigma\right| > \delta_{0\mu(0)}$ or $|\mathrm{si}_\pi^\sigma| > \delta_{0\pi(0)}$, thus by the Corollary $\sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = 0$.

Case 4. For $\mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma)$ to be nonzero (by the first case of the Corollary the term is zero if either $\left|\mathrm{si}_\mu^\sigma\right| > \delta_{0\mu(0)}$ or $|\mathrm{si}_\pi^\sigma| > \delta_{0\pi(0)}$) it is necessary that $\sigma = \pi$ by the Corollary. This is only one term, and thus

$$\sum_{\sigma\in\Sigma[1:k;1:n]} \mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma) = \mathrm{d}\lambda_\mu(e_\pi)\mathrm{d}\lambda_\pi(e_\pi)$$

which equals $(-1)^m$ where $m \in \mathrm{si}_\mu^\pi$ according to the third case of the Corollary.

Case 5. This case is similar to the last one, switching π and μ.

Case 6. By the Corollary, the only nonzero $\mathrm{d}\lambda_\mu(e_\sigma)\mathrm{d}\lambda_\pi(e_\sigma)$ is yielded by a σ satisfying $[\![\sigma]\!] = [\![\pi]\!] \cup [\![\mu]\!] \setminus \{0\}$ or $\left|\mathrm{si}_\mu^\sigma\right| = |\mathrm{si}_\pi^\sigma| = 1$. Then the content of these separate index sets define $m \in \mathrm{si}_\mu^\sigma$ and $p \in \mathrm{si}_\pi^\sigma$ in the final term $(-1)^{m+p}$, and these are the same as $m \in \mathrm{si}_\mu^\pi$ and $p \in \mathrm{si}_\pi^\mu$.

□

We have now proven how $(\mathrm{d}\lambda_\mu, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k(\mathbb{R}^n)}$ can be calculated, all factors of the inner products of our polynomial bases in (5.1.1) have now been identified. Thus we can proceed to calculate their Gram matrices

5.2. Condition numbers for the basis of $\mathcal{P}_r\Lambda^k(T_0)$

In this section we show that the condition number of our bases for $\mathcal{P}_r\Lambda^k(T_0)$, is independent of k because $\mathrm{d}\lambda_\sigma^{T_0} = \mathrm{d}x_\sigma$. The bases for a general $\mathcal{P}_r\Lambda^k(T)$ are

$$\left\{ \left(\lambda^T\right)^\alpha \mathrm{d}\lambda_\sigma^T \,\middle|\, \alpha \in \mathbb{N}_0^{0:n},\ |\alpha| = r,\ \sigma \in \Sigma(1:k; 1:n) \right\} \tag{5.2.1}$$

and the Bernstein-weighted basis

$$\left\{ \binom{|\alpha|}{\alpha} \left(\lambda^T\right)^\alpha \mathrm{d}\lambda_\sigma^T \,\middle|\, \alpha \in \mathbb{N}_0^{0:n},\ |\alpha| = r,\ \sigma \in \Sigma(1:k; 1:n) \right\}. \tag{5.2.2}$$

We will now consider how to calculate their condition numbers for T_0:

THEOREM 5.7. *The condition number for the basis of $H\mathcal{P}_r\Lambda^k(T_0)$*

$$\mathrm{cond}G\left(\left\{(\lambda)^\alpha \mathrm{d}\lambda_\sigma^T\right\}_{\alpha,\sigma}\right) = \mathrm{cond}G\left(\left\{(\lambda)^\alpha\right\}_\alpha\right) \mathrm{cond}\left(\left\{(\mathrm{d}\lambda_\sigma, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k}\right\}_{\sigma,\pi}\right) \tag{5.2.3}$$

where $\alpha \in \mathbb{N}_0^{0:n}$, $|\alpha| = r$ and $\sigma \in \Sigma(1:k; 1:n)$.

PROOF. The Gram matrix for these is simply described as

$$\left((\lambda)^\alpha \mathrm{d}\lambda_\sigma, (\lambda)^\beta \mathrm{d}\lambda_\pi\right)_{L^2\Lambda^k}$$

which can be written as

$$= \left((\lambda)^\alpha, (\lambda)^\beta\right) (\mathrm{d}\lambda_\sigma, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k}.$$

This is the Kronecker product of two matrices

$$\left\{ \left((\lambda)^\alpha, (\lambda)^\beta\right)_{L^2\Lambda^k} (\mathrm{d}\lambda_\sigma, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k} \right\}_{\alpha,\beta,\sigma,\pi} = \{L_{\alpha\beta} R_{\sigma\pi}\}_{\alpha,\beta,\sigma,\pi} = L \otimes R$$

which according to **[19]** has eigenvalues $\{y_i z_j\}_{i,j}$ given eigenvalues $\{y_i\}_i$ of L and $\{z_j\}_j$ of R. □

Proving this for the Bernstein-weighted basis in (5.2.2) is done by adding coefficients, and will not change the proof or conclusion.

Seeing that the condition number of the bases is directly reliant on their scalar counterparts and the Gram matrix of the alternating forms, we will get the following result.

COROLLARY 5.8. *The Bernstein and barycentric basis of $H\mathcal{P}_r\Lambda^k(T_0)$ have the same condition numbers as their scalar counterparts in Chapter 3.*

PROOF. Since $\left(\mathrm{d}\lambda_\sigma^{T_0}, \mathrm{d}\lambda_\pi^{T_0}\right)_{\mathrm{Alt}^k} = \delta_{\sigma\pi}$ (Kronecker delta) for $\sigma, \pi \in \Sigma(1:k; 1:n)$ (proven in Lemma 5.6 in Section 5.1), $\left\{(\mathrm{d}\lambda_\sigma, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k}\right\}_{\sigma,\pi} = I$. We then see that according to Theorem 5.7,

$$\mathrm{cond}G\left(\left\{\lambda^\alpha \mathrm{d}\lambda_\sigma\right\}_{|\alpha|=r,\sigma\in\Sigma}\right) = \mathrm{cond}G\left(\left\{\lambda^\alpha\right\}_{|\alpha|=r}\right)\mathrm{cond}\left(\left\{(\mathrm{d}\lambda_\sigma, \mathrm{d}\lambda_\pi)_{\mathrm{Alt}^k}\right\}_{\sigma,\pi}\right).$$

Since and $\mathrm{cond}I = 1$, the corollary is proved. □

We then only need to refer to Table 1 and 2 on page 33 to see the condition numbers of our bases for $\mathcal{P}_r\Lambda^k(T_0)$, drawing the conclusion that the Bernstein coefficients still improve the condition number quite a lot.

5.3. Condition numbers for the basis of $\mathcal{P}_r^-\Lambda^k(T_0)$

In this section we will calculate the condition numbers for the bases of the form $\lambda^i\phi_\sigma$ or $\binom{|i|}{i}\lambda^i\phi_\sigma$ for $\mathcal{P}_r^-\Lambda^k(T_0)$: The standard *barycentric basis*

$$\left\{\left(\lambda^T\right)^i\phi_\sigma \,\middle|\, i \in \mathbb{N}_r^{0:n}, |i| = r, \sigma \in \Sigma(1:k, 1:n)\right\}$$

and the *Bernstein-weighted basis*

$$\left\{\binom{|i|}{i}\left(\lambda^T\right)^i\phi_\sigma \,\middle|\, i \in \mathbb{N}_r^{0:n}, |i| = r, \sigma \in \Sigma(1:k, 1:n)\right\}.$$

Since it was easy to do, we've also calculated the condition number of the set of functions of the form

$$\sum_{l=0}^{k}\binom{|i+e_{\sigma(l)}|}{i+e_{\sigma(l)}}\left(\lambda^T\right)^{i+e_{\sigma(l)}}\mathrm{d}\lambda_{\underline{\sigma}_l}. \tag{5.3.1}$$

We have not proved that they are a basis of the space $\mathcal{P}_r^-\Lambda^k(T)$ nor that they have any similar, but the comparison with the two bases' condition numbers might justify further investigation. Programs for calculating their tables are found in A.5.

THEOREM 5.9. *The inner product of two $\mathcal{P}_r^-$-forms from $\mathcal{D}_r^-\Lambda^k(T_0)$ on the reference simplex is*

(5.3.2)

$$\left(\lambda^s\phi_\alpha^T, \lambda^t\phi_\beta^T\right)_{L^2\Lambda^k} = \sum_{i,j=0}^{k}(-1)^{i+j}\left(s_0+\delta_{0\alpha(i)}\right)!\left(t_0+\delta_{0\beta(j)}\right)!\mathtt{lsum}(s+e_{\alpha(i)}, j+e_{\beta(j)}, n)\left(\mathrm{d}\lambda_{\underline{\alpha}_i}, \mathrm{d}\lambda_{\underline{\beta}_j}\right)_{\mathrm{Alt}^k}$$

PROOF. We have the basis

$$\begin{aligned}\left(\lambda^s\phi_\alpha^T, \lambda^t\phi_\beta^T\right)_{H\Lambda^k} &= \sum_{i,j=0}^{k}(-1)^{i+j}\left(\lambda^s\lambda_{\alpha(i)}\mathrm{d}\lambda_{\underline{\alpha}_i}, \lambda^t\lambda_{\beta(j)}\mathrm{d}\lambda_{\underline{\beta}_j}\right)_{H\Lambda}\\ &= \sum_{i,j=0}^{k}(-1)^{i+j}\left(\lambda^{s+e_{\alpha(i)}}, \lambda^{t+e_{\beta(j)}}\right)_H\left(\mathrm{d}\lambda_{\underline{\alpha}_i}, \mathrm{d}\lambda_{\underline{\beta}_j}\right)_{\mathrm{Alt}}\end{aligned}$$

According to Corollary 3.7 we know that

$$\left(\lambda^{s+e_{\alpha(i)}}, \lambda^{t+e_{\beta(j)}}\right)_{L^2} = (s_0+\delta_{0\alpha(i)})!(t_0+\delta_{0\beta(j)})!\mathtt{lsum}(s+e_{\alpha(i)}, j+e_{\beta(j)}, n)$$

$$= (s_0+\delta_{0\alpha(i)})!(t_0+\delta_{0\beta(j)})!\sum_{|l|\le s_0+\delta_{0\alpha(i)}}(-1)^{|l|}\frac{\left(\underline{s+e_{\alpha(i)}}_0 + l + \underline{j+e_{\beta(j)}}_0\right)!}{l!(s_0+\delta_{0\alpha(i)}-|l|)!\left(\left|\underline{s+e_{\alpha(i)}}_0\right| + |l| + \left|j+e_{\beta(j)}\right| + n\right)!}.$$

□

Again the proof is similar for the Bernstein-weighted basis.

To compute the condition numbers of the gram matrix from (5.3.2), we've written the programs in Section A.4. The results of the programs can be seen in the Tables 1 through 10. As can be clearly seen from these results, the Bernstein-weighted basis has in general lower condition numbers than the barycentric basis, except for the (n, r, k)- values

$$(4,2,0),\ (5,2,0),\ (1,2,1),\ (2,2,2),\ (3,2,3),\ (4,2,4),\ (5,2,5)$$

which are all highlighted in Table 2.

We can also see from Table 1 and Table 6 on page 53 that the Whitney k-forms for $k > 0$ all have lower condition numbers than those for $k = 0$. For general $r \geq 0$, we can the condition numbers reach a peak when k is far from 0 and n, and decreases when going to 0 or n. There is not enough data to pinpoint the exact position of this peak for either set of bases.

We include the results of the potential basis from (5.3.1) in Chapter B, just to illustrate that this potential basis would be worse conditioned than the Bernstein-weighted basis, and therefore not a good candidate to replace it.

Considering normalising coefficients for the basis functions would probably produce good candidates for replacing the Bernstein-weighted basis as the best conditioned basis for $H\mathcal{P}_r^-(\mathcal{T})$, and is therefore a good place to start future work.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	3.0000	1.0000	–	–	–	–
2	4.0000	2.5000	1.0000	–	–	–
3	5.0000	3.0000	2.3333	1.0000	–	–
4	6.0000	3.5000	2.6667	2.2500	1.0000	–
5	7.0000	4.0000	3.0000	2.5000	2.2000	1.0000

TABLE 1. Condition numbers of the barycentric basis for $n, k \leq 5$, $r = 1$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	23.5576	**2.5000**	–	–	–	–
2	33.3921	38.5591	**3.3333**	–	–	–
3	47.2211	59.2133	53.5537	**4.2000**	–	–
4	**57.0363**	83.8786	83.1270	70.6842	**5.0909**	–
5	**67.8786**	$1.03 \cdot 10^2$	$1.13 \cdot 10^2$	$1.11 \cdot 10^2$	89.9042	**6.0000**

TABLE 2. Condition numbers of the barycentric basis for $n, k \leq 5$, $r = 2$. Highlighted numbers indicate where the barycentric basis has a lower condition number than the corresponding Bernstein-weighted basis Gram matrix.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$1.82\cdot 10^2$	24.8505	–	–	–	–
2	$6.80\cdot 10^2$	$3.30\cdot 10^2$	33.8306	–	–	–
3	$8.32\cdot 10^2$	$9.95\cdot 10^2$	$4.40\cdot 10^2$	46.3571	–	–
4	$1.01\cdot 10^3$	$1.30\cdot 10^3$	$1.28\cdot 10^3$	$5.81\cdot 10^2$	54.5223	–
5	$1.22\cdot 10^3$	$1.64\cdot 10^3$	$1.69\cdot 10^3$	$1.63\cdot 10^3$	$7.69\cdot 10^2$	63.6753

TABLE 3. Condition numbers of the barycentric basis for $n, k \leq 5$, $r = 3$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$1.89\cdot 10^3$	$1.97\cdot 10^2$	–	–	–	–
2	$9.28\cdot 10^3$	$7.05\cdot 10^3$	$9.05\cdot 10^2$	–	–	–
3	$3.53\cdot 10^4$	$2.07\cdot 10^4$	$9.35\cdot 10^3$	$1.06\cdot 10^3$	–	–
4	$3.94\cdot 10^4$	$4.37\cdot 10^4$	$2.90\cdot 10^4$	$1.21\cdot 10^4$	$1.24\cdot 10^3$	–
5	$4.39\cdot 10^4$	$5.07\cdot 10^4$	$5.02\cdot 10^4$	$3.54\cdot 10^4$	$1.52\cdot 10^4$	$1.45\cdot 10^3$

TABLE 4. Condition numbers of the barycentric basis for $n, k \leq 5$, $r = 4$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$2.16\cdot 10^4$	$2.18\cdot 10^3$	–	–	–	–
2	$1.80\cdot 10^5$	$1.46\cdot 10^5$	$1.29\cdot 10^4$	–	–	–
3	$7.99\cdot 10^5$	$8.23\cdot 10^5$	$4.72\cdot 10^5$	$5.79\cdot 10^4$	–	–
4	$3.04\cdot 10^6$	$1.93\cdot 10^6$	$1.30\cdot 10^6$	$5.88\cdot 10^5$	$6.30\cdot 10^4$	–
5	$3.27\cdot 10^6$	$3.44\cdot 10^6$	$2.37\cdot 10^6$	$1.71\cdot 10^6$	$7.17\cdot 10^5$	$6.84\cdot 10^4$

TABLE 5. Condition numbers of the barycentric basis for $n, k \leq 5$, $r = 5$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	3.0000	1.0000	–	–	–	–
2	4.0000	2.5000	1.0000	–	–	–
3	5.0000	3.0000	2.3333	1.0000	–	–
4	6.0000	3.5000	2.6667	2.2500	1.0000	–
5	7.0000	4.0000	3.0000	2.5000	2.2000	1.0000

TABLE 6. Condition numbers of the Bernstein-weighted basis for $n, k \leq 5$, $r = 1$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	11.1352	3.4000	–	–	–	–
2	24.6261	13.8667	4.3333	–	–	–
3	42.2740	28.7752	16.9266	5.3077	–	–
4	64.0201	43.5169	42.1325	24.6175	6.2941	–
5	89.8291	58.8286	63.2408	60.5373	34.8626	7.2857

TABLE 7. Condition numbers of the Bernstein-weighted basis for $n, k \leq 5$, $r = 2$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	49.1979	18.6722	–	–	–	–
2	$1.17\cdot 10^2$	96.3658	19.3228	–	–	–
3	$2.24\cdot 10^2$	$3.11\cdot 10^2$	$1.67\cdot 10^2$	25.3280	–	–
4	$3.82\cdot 10^2$	$5.68\cdot 10^2$	$5.49\cdot 10^2$	$3.00\cdot 10^2$	32.0291	–
5	$6.00\cdot 10^2$	$9.17\cdot 10^2$	$9.10\cdot 10^2$	$9.10\cdot 10^2$	$4.96\cdot 10^2$	39.9775

TABLE 8. Condition numbers of the Bernstein-weighted basis for $n, k \leq 5$, $r = 3$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$2.44\cdot 10^2$	94.9730	–	–	–	–
2	$6.10\cdot 10^2$	$5.58\cdot 10^2$	$1.21\cdot 10^2$	–	–	–
3	$1.25\cdot 10^3$	$2.10\cdot 10^3$	$1.11\cdot 10^3$	$1.23\cdot 10^2$	–	–
4	$2.26\cdot 10^3$	$4.16\cdot 10^3$	$4.10\cdot 10^3$	$2.20\cdot 10^3$	$1.56\cdot 10^2$	–
5	$3.77\cdot 10^3$	$7.32\cdot 10^3$	$7.35\cdot 10^3$	$7.43\cdot 10^3$	$3.99\cdot 10^3$	$2.02\cdot 10^2$

TABLE 9. Condition numbers of the Bernstein-weighted basis for $n, k \leq 5$, $r = 4$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$1.23\cdot 10^3$	$4.78\cdot 10^2$	–	–	–	–
2	$3.12\cdot 10^3$	$3.08\cdot 10^3$	$5.50\cdot 10^2$	–	–	–
3	$6.53\cdot 10^3$	$1.22\cdot 10^4$	$6.34\cdot 10^3$	$7.63\cdot 10^2$	–	–
4	$1.22\cdot 10^4$	$2.60\cdot 10^4$	$2.51\cdot 10^4$	$1.33\cdot 10^4$	$7.92\cdot 10^2$	–
5	$2.12\cdot 10^4$	$4.90\cdot 10^4$	$4.83\cdot 10^4$	$4.82\cdot 10^4$	$2.56\cdot 10^4$	$9.95\cdot 10^2$

TABLE 10. Condition numbers of the Bernstein-weighted basis for $n, k \leq 5$, $r = 5$

Bibliography

[1] Douglas N. Arnold, Richard S. Falk, and Ragnar Winther. Differential complexes and stability of finite element methods. i. the de rham complex. *IMA volumes in Math. and appl.*, 142:23–46, 2006.

[2] Douglas N. Arnold, Richard S. Falk, and Ragnar Winther. Finite element exterior calculus, homological techniques, and applications. *Acta Numerica*, 15:1–155, 2006.

[3] Douglas N. Arnold, Richard S. Falk, and Ragnar Winther. Geometric decompositions and local bases for spaces of finite element differential forms. *Computer methods in appl. mech. and eng.*, 198:1660–1672, 2009.

[4] Dennis Barden and Charles Thomas. *An Introduction To Differential Manifolds*. Imperial College Press, 2003.

[5] Robert G. Bartle. *The elements of integration and Lebesgue measure*. Bartle, 1995.

[6] Dietrich Braess. *Finite elements : theory, fast solvers, and applications in solid mechanics*. Cambridge : Cambridge University Press, 2001.

[7] Susanne C. Brenner and L. Ridgway Scott. *The Mathematical Theory of Finite Element Methods*. Springer, third edition, 2008.

[8] C. de Boor. B-form basics. In G. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, pages 131–148. SIAM, Philadelphia, 1987.

[9] Eidelman. *Functional Analysis*. AMS, 2004.

[10] Lawrence C. Evans. *Partial Differential Equations*. AMS, 1998.

[11] B. G. Galerkin. On electrical circuits for the approximate solution of the laplace equation. *Vestnik Inzhenyerom*, pages 897–908, 1915.

[12] Peter D. Lax. *Functional Analysis*. Wiley-Interscience, 2002.

[13] Tom Lyche. Lecture notes for inf-mat 4350, 2009. 2009.

[14] Sidney A. Morris. *Topology Without Tears*. October 2007.

[15] Bryan P. Rynne and Martin A. Youngson. *Linear Functional Analysis*. Springer, 2008.

[16] Karl Scherer Tom Lyche. On the p-norm condition number of the multivariate triangular bernstein basis. *Journal of Computational and Applied Mathematics*, 119:259–273, 2000.

[17] Hassler Whitney. *Geometric Integration Theory*. Princeton University Press, 1957.

[18] Wikipedia. Condition number — wikipedia, the free encyclopedia, 2009. [Online; accessed 21-July-2009].

[19] Wikipedia. Kronecker product — wikipedia, the free encyclopedia, 2009. [Online; accessed 23-April-2009].

APPENDIX A

Source code

A.1. General programs

Listing A.1

```
%% MULTINOMIAL FUNCTION mnom(x,y)
% gives you x!/(y!*(x-|y|)!), which is the multinomial coefficient,
% binomial if y is 1x1.
%
% If given a vector in x and matrix in y, it will take the multinomial of
% the columns and return a vector.
function B = mnom(x,y)
%try
    B=round(factorial(x)./(prod(factorial(y),1).*factorial(x-sum(y,1))));
%catch
%    B=0;
%end
end
```

Listing A.2

```
function a = fact(b)
a = prod(factorial(b),1);
endfunction
```

Listing A.3

```
function X=generateIndices(s,n) %s is spatial dimention, n polynomial degree
X = zeros(s,mnom(s+n,n));
csum = zeros(1,mnom(s+n,n));
for row = 1:s
    for degsum = 0:n
        b = csum==degsum;
        if sum(b)
            d = 0:n-degsum;
            num = mnom(s-row+n-d-degsum,s-row);
            cnum = [0, cumsum(num)];
            p = zeros(1,max(cnum));
            repetitions = sum(b)/max(cnum);
            for deg = 0:n-degsum
                p( cnum(deg+1) + 1 : cnum(deg+2) ) = deg;
            end
            p_orig = p;
                for i = 2:repetitions
                    p = [p p_orig];
                end
```

```
            X(row,b)=p;
        end;
    end
    csum = csum + X(row,:);
end
end
```

Listing A.4

```
%% homogeneousIndices(n,r)
% generates the indices of barycentric homogenous polynomials of degree r
% in R^n - meaning that there are n+1 indices.
function I=homogeneousIndices(n,r) %n er romdimensjon, r er polynomgrad
J=generateIndices(n,r);
Io=r-sum(J,1);
I=[J;Io];
end
```

Listing A.5

```
function S = lsum(i,j)
s=length(i);

%establishing arguments for the vector space P^r(R^n):
r=sum(i);
n=s-1;

% separating the power of the barycentric coordinate relating to origo
i_=i(1:n);
i0=i(s);

j_=j(1:n);
j0=j(s);


%generating summation indexes
l = generateIndices(n,i0);
%alpha0 = i0-sum(l,1);

%pre-generating sum
l_i_j = l;
for k=1:n
    l_i_j(k,:) = l_i_j(k,:) +i_(k) + j_(k);
end;
%creating the terms depending on l
numerator = fact(l_i_j).*((-1).^(sum(l,1)));
denominator = fact(l) .* factorial(i0-sum(l,1)).*fact(r+n+sum(l,1)+sum(i_,1));
S = sum(numerator./denominator);
end
```

A.2. Barycentric basis programs

Listing A.6

```
%% barycentricInnerProduct(i,j)
% has the peculiar ability to return the analytical result of the inner
```

```
% product (lambda^i,lambda^j), where i and j are multiindices of at least
% dimension 2.
%
% Properties:
% -symmetric
% usage:
% <not final yet>

function R=barycentricInnerProduct(i,j)
l=length(j);
if l~=length(i)
    error('The lenghts of i and j are different!')
end
if l==1
    R = 1;
else
% summing up and calculating final inner product
R=factorial(i(length(i)))*factorial(j(length(j)))*lsum(i,j);
end

end
```

Listing A.7

```
%% GramBarycentric(n,r)
% Generates the gram matrix for the r-th degree polynomial basis over an
% n-simplex.
function G = GramBarycentric(n,r)
I = homogeneousIndices(n,r);
Isize = size(I);
S=Isize(2);
G=zeros(S);
for i=1:S;
    for j=1:S;
        %T(i,j)=sum(I(:,i).*I(:,j));
        G(i,j)=barycentricInnerProduct(I(:,i),I(:,j));
    end
end
%cond(T);
end
```

Listing A.8

```
function g=BarycentricConds(N,R)
sprintf('initialising...\n')
try
    load('matrices/baryc_conds_matrix.mat','g')
catch
    g=zeros(N,R);
end

for n=1:N
    for r=1:R
        [n,r]
        g(n,r) = cond(GramBarycentric(n,r));
```

```
    end
end

save('matrices/baryc_conds_matrix.mat','g')
end
```

Listing A.9

```
%% BernsteinInnerProduct(i,j)
% has the peculiar ability to return the analytical result of the inner
% product (lambda^i,lambda^j), where i and j are multiindices of at least
% dimension 2.
%
% Properties:
% -symmetric
% usage:
% R=

function R=BernsteinInnerProduct(i,j)
s=length(j);
if s~=length(i)
    error('The lenghts of i and j are different!')
elseif s==1
R = 1;
else
% summing up and calculating final inner product
R=factorial(sum(i))*factorial(sum(j))/fact(i(1:s-1))/fact(j(1:s-1))*lsum(i,j);
end

end
```

Listing A.10

```
%% Gram(n,r)
% Generates the gram matrix for the r-th degree polynomial basis over an
% n-simplex.
function G = GramBernstein(n,r)
I = homogeneousIndices(n,r);
Isize = size(I);
S=Isize(2);
G=zeros(S);
for i=1:S;
    for j=1:S;
        %T(i,j)=sum(I(:,i).*I(:,j));
        G(i,j)=BernsteinInnerProduct(I(:,i),I(:,j));
    end
end
%cond(T);
end
```

Listing A.11

```
function g=BernsteinCondsControl(N,R)
sprintf('initialising...\n')
try
    load('matrices/berns_conds_control_matrix.mat','g')
```

```
catch
    g=zeros(N,R);
end

for n=1:N
    for r=1:R
        [n,r]
        g(n,r) = cond(GramBernstein(n,r));
    end
end

save('matrices/berns_conds_control_matrix.mat','g')

end
```

LISTING A.12

```
function g=BernsteinConds(N,R)
try
    load('matrices/berns_conds_matrix.mat','g')
catch
    g=zeros(N,R);
end

for n=1:N
    for r=1:R
%       [n,r]
        g(n,r) = (mnom(2*r+n,r));
    end
end

save('matrices/berns_conds_matrix','g')

end
```

A.3. Nodal scalar bases

LISTING A.13

```
%% barycentricSubsimplexFunctional(i,j)
% returns the value of the inner product of lambda^i-reduced (see
% definitions) with lambda^j on the subsimplex where lambda^i is strictly
% positive. sum(i) must be equal to sum(j).
%
% Does the
function R=barycentricSubsimplexFunctional(i,j)
i_corners = (i~=0);
i_complementCorners = (i==0);
i_red = i-i_corners;
n = sum(i_corners);
if sum(i_complementCorners.*j) ~= 0
    R = 0;
else
    jnew = i_corners.*j;
    R =barycentricInnerProduct(i_red(i_corners),jnew(i_corners));
```

```
    if (i(length(i))==0)
        R=sqrt(n)*R;
    end
    %sqrt(n) is the determinant of the jacobian to the linear
    %transformation to an outer simplex.
end
end
```

Listing A.14

```
%% barycentricSSmatrix(n,r)
% Generates the matrix of standard K-nodes of
% for the r-th degree polynomial basis over an
% n-simplex.
function G = barycentricSSmatrix(n,r)
I = homogeneousIndices(n,r);
Isize = size(I);
S=Isize(2);
G=zeros(S);
for i=1:S;
    for j=1:S;
        %T(i,j)=sum(I(:,i).*I(:,j));
        G(i,j)=barycentricSubsimplexFunctional(I(:,i),I(:,j));
    end
end
%cond(T);
end
```

Listing A.15

```
%% GramNodal(n,r)
% Generates the gram matrix for the r-th degree _sub-nodal_ polynomial basis
% over an n-simplex.
function G = GramNodal(n,r)
C=inv(standardSSmatrix(n,r));
D=GramBarycentric(n,r);
G=C*D*C';
end
```

Listing A.16

```
%% GramsNodalConds(n,r)
% Generates the condition numbers for the gram matrices of the nodal basis
% of Aof degree r in an n-simplex
function g = NodalConds(n,r)
sprintf('initialising...\n')
try
    load('matrices/nodal_conds_matrix.mat','g')
catch
    g=zeros(n,r);
end

for N=1:n;
    for R=1:r;
    [N R]
        g(N,R)=cond(GramNodal(N,R));
```

```
    end
end
save('matrices/nodal_conds_matrix.mat','g')

end
```

Listing A.17

```
%% BernsteinSubsimplexFunctional(i,j)
% returns the value of the inner product of B_{i_red} on the subsimplex f
% (|i_red| = r-dim(f)-1) with B_j. This is done  on the subsimplex where i
% (the original) is strictly positive, then i is manipulated to an i_red, which
%
function R = BernsteinSubsimplexFunctional(i,j)
%defining the reduced index i_red out of i
i_corners = (i~=0);
i_complementCorners = (i==0);
i_reduced = i-i_corners;
i_red = i_reduced(i_corners);

n = sum(i_corners); %dimension of f
% B_j vanishes on f if it has a factor lambda_m where m is not an index of a corner in f, it .
if sum(i_complementCorners.*j) ~= 0
    R = 0;
else
    jnew = i_corners.*j;
    R =mnom(sum(i_red),i_red)*barycentricInnerProduct(i_red,jnew(i_corners));
    if (i(length(i))==0)
        R=sqrt(n)*R;
    end
    %sqrt(n) is the determinant of the jacobian to the linear
    %transformation to an outer subsimplex.
end
end
```

Listing A.18

```
%% standardSSmatrix(n,r)
% Generates the matrix of standard K-nodes of
% for the r-th degree polynomial basis over an
% n-simplex.
function G = BernsteinSSmatrix(n,r)
I = homogeneousIndices(n,r);
Isize = size(I);
S=Isize(2);
G=zeros(S);
for i=1:S;
    for j=1:S;
        %T(i,j)=sum(I(:,i).*I(:,j));
        G(i,j)=BernsteinSubsimplexFunctional(I(:,i),I(:,j));
    end
end
%cond(T);
end
```

LISTING A.19

```
%% GramNodal(n,r)
% Generates the gram matrix for the r-th degree _sub-nodal_ polynomial basis
% over an n-simplex.
function G = GramBernsteinNodal(n,r)
C=inv(BernsteinSSmatrix(n,r));
D=GramBarycentric(n,r);
G=C*D*C';
end
```

LISTING A.20

```
%% GramsNodalConds(n,r)
% Generates the condition numbers for the gram matrices of the nodal basis
% of Aof degree r in an n-simplex
function g = BernsteinNodalConds(n,r)
sprintf('initialising...\n')
try
    load('matrices/berns_nodal_conds_matrix.mat','g')
catch
    g=zeros(n,r);
end

for N=1:n;
    for R=1:r;
    [N R]
        g(N,R)=cond(GramBernsteinNodal(N,R));
    end
end
save('matrices/berns_nodal_conds_matrix.mat','g')

end
```

A.4. Bases for $\mathcal{P}_r^-\Lambda^k(T_0)$

LISTING A.21

```
function P = AltInner(a, b, n, k)
P=0; %3
if k==0
P=1;
elseif and(a(1)==0,b(1)==0)
    if a==b
        P = n - k + 1;%1
    else
    m = indicesOfUniques( a , b );
    p = indicesOfUniques( b , a );
        if length(m)==1
            P =(-1)^(m+p) ;%6
        end
    end
elseif or(a(1)==0,b(1)==0) ,
    s = [];
    if a(1)==0
        s = indicesOfUniques( b , a );
```

```
    else
        s = indicesOfUniques( a , b );
    end
    if length(s)==1
        P = (-1)^s(1); %4+5
    end
elseif a==b
    P = 1; %2
end
end

function s = indicesOfUniques( a , b )
s = [];
for i=1:length(a)
    if prod(a(i)~=b)
        s = [s,i];
    end
end
end
```

Listing A.22

```
function p = P_innerProduct(i,a,j,b,n,k)
kplus = k+1;
p=0;
for l=1:kplus
    for m=1:kplus
%[l,m]
%a([1:l-1,l+1:kplus])
%b([1:m-1,m+1:kplus])
        s = AltInner( a([1:l-1,l+1:kplus]),b([1:m-1,m+1:kplus]),n , k);
        if s ~= 0
            i_ = i;
            i_(a(l)) = i_(a(l)) + 1;
            j_ = j;
            j_(b(m)) = j_(b(m)) + 1;
            s = s * ((-1)^(l+m)) * barycentricInnerProduct(i_,j_);
            p = p + s;
        end
        %p=p+s;

    end
end
end
```

Listing A.23

```
function G = Gram_P(n,r,k)
I = homogeneousIndices(n,r-1);
Isize = size(I);
S=Isize(2);
indices = combnk( 1 : n+1 , k+1 );
Ssize=size(indices);
SI=Ssize(1);
```

```
G=zeros(mnom(r+k-1,k)*mnom(n+r,n-k));

c=1;
d=1;

for i=1:S
    for a = 1:SI
        if sum( I(1:min(indices(a,:))-1 , i)) == 0
            for j=1:S
                for b = 1:SI
                  if sum( I(1:min(indices(b,:))-1 , j)) == 0
%                       [I(:,i),I(:,j)]
%                       [indices(a,:);indices(b,:)]
                    G(c,d) = ...
                        P_innerProduct(I(:,i), indices(a,:),I(:,j),indices(b,:),n,k);
                    d = d+1;
                  end
                end
            end
            c = c+1;
            d = 1;
        end
    end
end
%cond(T);
end
```

Listing A.24

```
function G = PConds(N,R,K)
for n=N
    for r=R
        for k=K(K<=n)
            [n,r,k]
            G(n,r,k+1) = cond(Gram_P(n,r,k));
            try
                load('P_conds_matrix.mat','g');
            end
            g(n,r,k+1) = G(n,r,k+1);
            try
                save('P_conds_matrix.mat','g');
            end
        end
    end
end
end
```

Listing A.25

```
function p = P_Bernstein_2_InnerProduct(i,a,j,b,n,k)
kplus = k+1;
p=0;
for l=1:kplus
    for m=1:kplus
        s = AltInner( a([1:l-1,l+1:kplus]),b([1:m-1,m+1:kplus]),n , k);
```

```
        if s ~= 0
            i_ = i;
            i_(a(l)) = i_(a(l)) + 1;
            j_ = j;
            j_(b(m)) = j_(b(m)) + 1;
            s = s * ((-1)^(l+m)) *(sum(i_)*sum(j_)/(i_(a(l))*j_(b(m)))) * BernsteinInnerProduct(
            p = p + s;
        end
    end
end
end
```

Listing A.26

```
function G = Gram_PBernstein_2(n,r,k)
I = homogeneousIndices(n,r-1);
Isize = size(I);
S=Isize(2);
indices = combnk( 1 : n+1 , k+1 );
Ssize=size(indices);
SI=Ssize(1);

G=zeros(mnom(r+k-1,k)*mnom(n+r,n-k));

c=1;
d=1;

for i=1:S
    for a = 1:SI
        if  sum( I(1:min(indices(a,:))-1 , i)) == 0
            for j=1:S
                for b = 1:SI
                  if sum( I(1:min(indices(b,:))-1 , j)) == 0
%                       [I(:,i),I(:,j)]
%                       [indices(a,:);indices(b,:)]
                    G(c,d) = ...
                        P_Bernstein_2_InnerProduct(I(:,i), indices(a,:),I(:,j),indices(b,:),n,k)
                    d = d+1;
                  end
                end
            end
            c = c+1;
            d = 1;
        end
    end
end
end
```

Listing A.27

```
function G = PBernstein_2_Conds(N,R,K)
for n=N
    for r=R
        for k=K(K<=n)
            [n,r,k]
```

```
            G(n,r,k+1) = cond(Gram_PBernstein_2(n,r,k));
            try
                load('P_Bernstein_2_conds_matrix.mat','g');
            end
            g(n,r,k+1) = G(n,r,k+1);
            try
                save('P_Bernstein_2_conds_matrix.mat','g');
            end
        end
    end
end
end
```

A.5. Experimental programs

LISTING A.28

```
function p = P_BernsteinInnerProduct(i,a,j,b,n,k)
kplus = k+1;
p=0;
for l=1:kplus
    for m=1:kplus
        s = AltInner( a([1:l-1,l+1:kplus]),b([1:m-1,m+1:kplus]),n , k);
        if s ~= 0
                i_ = i;
            i_(a(l)) = i_(a(l)) + 1;
                   j_ = j;
            j_(b(m)) = j_(b(m)) + 1;
            s = s * ((-1)^(l+m)) * BernsteinInnerProduct(i_,j_);
                p = p + s;
            end
    end
end
end
```

LISTING A.29

```
function G = Gram_PBernstein(n,r,k)
I = homogeneousIndices(n,r-1);
Isize = size(I);
S=Isize(2);
G=zeros(S);
indices = combnk( 1 : n+1 , k+1 );
Ssize=size(indices);
SI=Ssize(1);
G=zeros(mnom(r+k-1,k)*mnom(n+r,n-k));

c=1;
d=1;

for i=1:S
    for a = 1:SI
        if  sum( I(1:min(indices(a,:))-1 , i)) == 0
            for j=1:S
                for b = 1:SI
```

```
                    if sum( I(1:min(indices(b,:))-1 , j)) == 0
%                       [I(:,i),I(:,j)]
%                       [indices(a,:);indices(b,:)]
                      G(c,d) = ...
                          P_BernsteinInnerProduct(I(:,i), indices(a,:),I(:,j),indices(b,:),n,k);
                      d = d+1;
                    end
                end
            end
            c = c+1;
            d = 1;
        end
    end
 end
% for i=1:S;
%     for j=1:S;
%       for a = 1:SI
%           for b = 1:SI
%           P_BernsteinInnerProduct(I(:,i), indices(a,:),I(:,j),indices(b,:),n,k);
%           end
%       end
%     end
% end
% %cond(T);
end
```

Listing A.30

```
function G = PBernsteinConds(N,R,K)
for n=N
    for r=R
        for k=K(K<=n)
            [n,r,k]
            G(n,r,k+1) = cond(Gram_PBernstein(n,r,k));
            try
                load('P_Bernstein_conds_matrix.mat','g');
            end
            g(n,r,k+1) = G(n,r,k+1);
            try
                save('P_Bernstein_conds_matrix.mat','g');
            end
        end
    end
end
end
```

APPENDIX B

Results of the programs from A.5

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	3.0000	1.0000	–	–	–	–
2	4.0000	2.5000	1.0000	–	–	–
3	5.0000	3.0000	2.3333	1.0000	–	–
4	6.0000	3.5000	2.6667	2.2500	1.0000	–
5	7.0000	4.0000	3.0000	2.5000	2.2000	1.0000

TABLE 1. Results for the experimental basis candidate (5.3.1) for $n, k \leq 5$, $r = 1$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	10.0000	4.0000	–	–	–	–
2	15.0000	19.8015	5.0000	–	–	–
3	21.0000	35.2737	27.7186	6.0000	–	–
4	28.0000	52.6050	52.8141	37.7591	7.0000	–
5	36.0000	68.6702	75.5037	74.5922	49.9311	8.0000

TABLE 2. Results for the experimental basis candidate (5.3.1) for $n, k \leq 5$, $r = 2$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	35.0000	15.1441	–	–	–	–
2	56.0000	97.0139	21.1229	–	–	–
3	84.0000	$1.85 \cdot 10^2$	$1.52 \cdot 10^2$	28.1290	–	–
4	$1.20 \cdot 10^2$	$2.84 \cdot 10^2$	$3.03 \cdot 10^2$	$2.24 \cdot 10^2$	36.1377	–
5	$1.65 \cdot 10^2$	$4.07 \cdot 10^2$	$4.40 \cdot 10^2$	$4.65 \cdot 10^2$	$3.19 \cdot 10^2$	45.1409

TABLE 3. Results for the experimental basis candidate (5.3.1)for $n, k \leq 5$, $r = 3$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$2.44 \cdot 10^2$	94.9730	–	–	–	–
2	$6.10 \cdot 10^2$	$5.58 \cdot 10^2$	$1.21 \cdot 10^2$	–	–	–
3	$1.25 \cdot 10^3$	$2.10 \cdot 10^3$	$1.11 \cdot 10^3$	$1.23 \cdot 10^2$	–	–
4	$2.26 \cdot 10^3$	$4.16 \cdot 10^3$	$4.10 \cdot 10^3$	$2.20 \cdot 10^3$	$1.56 \cdot 10^2$	–
5	$3.77 \cdot 10^3$	$7.32 \cdot 10^3$	$7.35 \cdot 10^3$	$7.43 \cdot 10^3$	$3.99 \cdot 10^3$	$2.02 \cdot 10^2$

TABLE 4. Results for the experimental basis candidate (5.3.1)for $n, k \leq 5$, $r = 4$.

$n\downarrow, k\rightarrow$	0	1	2	3	4	5
1	$4.62 \cdot 10^2$	$2.12 \cdot 10^2$	–	–	–	–
2	$7.92 \cdot 10^2$	$1.73 \cdot 10^3$	$3.37 \cdot 10^2$	–	–	–
3	$1.29 \cdot 10^3$	$3.51 \cdot 10^3$	$3.09 \cdot 10^3$	$5.06 \cdot 10^2$	–	–
4	$2.00 \cdot 10^3$	$6.02 \cdot 10^3$	$6.39 \cdot 10^3$	$5.07 \cdot 10^3$	$7.30 \cdot 10^2$	–
5	$3.00 \cdot 10^3$	$9.69 \cdot 10^3$	$1.03 \cdot 10^4$	$1.08 \cdot 10^4$	$7.94 \cdot 10^3$	$1.02 \cdot 10^3$

TABLE 5. Results for the experimental basis candidate (5.3.1)for $n, k \leq 5$, $r = 5$