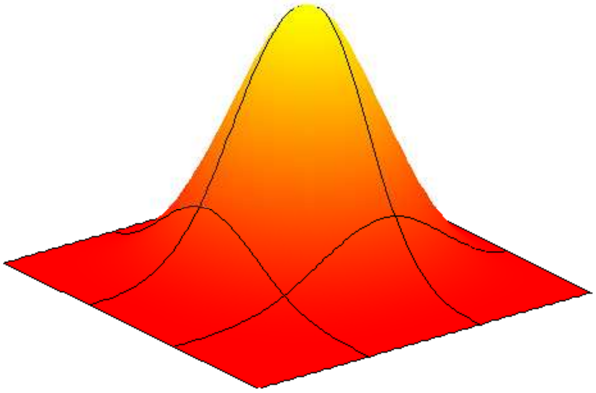
Mimetic Discretizations with B-Splines

On the Construction of a Discrete Hodge Star Operator

Master of Science Thesis



GIJS L. KOOIJ

Faculty of Aerospace Engineering

Delft University of Technology

Mimetic Discretizations with B-Splines

On the Construction of a Discrete Hodge Star Operator

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering at Delft University of Technology

GIJS L. KOOIJ

February 21, 2013

Faculty of Aerospace Engineering · Delft University of Technology

Gijs L. Kooij
Faculty of Aerospace Engineering
Delft University of Technology

Supervisor
M.I. Gerritsma
Faculty of Aerospace Engineering
Delft University of Technology

Cover illustration
Bivariate cubic B-spline. The first and second derivatives are continuous across the edges of the local polynomial surfaces, creating a seemingly smooth global surface.

© 2013 Gijs L. Kooij
This thesis is made available under the terms of the Creative Commons Attribution 3.0 license, found at http://creativecommons.org/licenses/by/3.0/.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Mimetic Discretizations with B-Splines"** by **Gijs L. Kooij** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: March 11, 2013

Professor:
Prof.dr.ir.drs. H. Bijl

Supervisor:
Dr.ir. M.I. Gerritsma

Reader:
Dr. S.J. Hulshoff

Reader:
Ir. W.R.M. van Hoydonck

Abstract

This thesis introduces a higher-order numerical method for elliptic boundary value problems. The discretization method belongs to the class of mimetic discretizations, which translate as many of properties of the continuous problem to the discrete system, aiming to improve accuracy and reliability. The novelty lies in the application of B-splines as basis functions in a dual grid approach. B-splines or basis splines are piecewise polynomials with a certain degree of continuity between the polynomial pieces. Therefore, splines offer an attractive compromise between piecewise linear functions, commonly seen in finite element analysis, and the Lagrange polynomials from spectral element methods.

Inner-oriented quantities, such as line integrals, are placed on the primal grid and outer-oriented quantities, such as fluxes, on the dual grid. The discrete Hodge operator is a mapping between the two grids and plays a key role in the discretization of the Laplacian. The definition of an explicit Hodge operator is not necessary, but has several advantages: material laws and Neumann boundary conditions can be imposed strongly, and problems can be posed naturally in a mixed formulation.

The discretization method is applied to Poisson's equation as a prototype for elliptic boundary value problems. The convergence rates with respect to grid refinement are found to be $k-1$ for odd order splines, but k for even order splines. The convergence rates of $k-1$ agree with theoretical estimates established for conventional spline collocation methods.

Keywords: Mimetic discretizations, B-splines, discrete Hodge operator, dual grid.

Preface

This report concludes my thesis project that lasted for approximately one year. I have to acknowledge that my thesis is an infinitesimal contribution to the grand body of knowledge; yet its completion gives me a great sense of satisfaction and fulfillment. During this time I was introduced to the mathematical disciplines of differential geometry and algebraic topology, the two pillars of mimetic discretizations. This new kind of mathematics throwed new light on the classical multivariable calculus taught to me as a university freshman. What struck me was the elegance of differential geometry: why do engineers and scientists alike continue to fiddle with the quirky del operator?

I kindly take this opportunity to express my gratitude to all the persons that supported me in my work. First of all, I would like to thank my supervisor Marc Gerritsma for his guidance and wisdom throughout the project.

Secondly, I would like to thank René Hiemstra for his early-on advice and kick-starting me in the world of splines and differential forms.

Thirdly, I would like to thank my parents for allowing me to pursue my personal interests, even with little notion of what I was doing.

Finally, I would like to thank all my fellow "dungeon" dwellers with whom I shared so many months downstairs. Their company definitely meant a more pleasurable and stimulating working environment.

Gijs Kooij
Delft, The Netherlands

Contents

Preface i

Contents i

List of figures v

List of tables vii

Nomenclature ix

1 Introduction 1
- 1.1 Literature survey . . . 2
- 1.2 Report outline . . . 3

2 Fundamentals of splines 5
- 2.1 Introduction . . . 5
- 2.2 Knots and B-splines . . . 7
- 2.3 Control points . . . 11
- 2.4 Differentiation . . . 13
- 2.5 Integration . . . 15
- 2.6 M-splines . . . 15
- 2.7 Interpolation . . . 15
- 2.8 Histopolation . . . 16
- 2.9 Basis functions . . . 19
  - 2.9.1 Nodal basis functions . . . 19
  - 2.9.2 Edge basis functions . . . 20

3 Mimetic discretizations 23
- 3.1 Introduction . . . 23
  - 3.1.1 Differential forms . . . 23
  - 3.1.2 Algebraic topology . . . 28
- 3.2 Basic operations . . . 29
  - 3.2.1 Reconstruction . . . 29
  - 3.2.2 Reduction . . . 31
- 3.3 Natural operations . . . 33
  - 3.3.1 Inner product . . . 33
  - 3.3.2 Exterior product . . . 33
- 3.4 Discrete operators . . . 33
  - 3.4.1 Exterior derivative . . . 34

3.4.2 Hodge $\star$ operator . . . 34
3.4.3 Laplace-de Rham operator . . . 39

4 Numerical experiments 41
4.1 Projection error . . . 41
4.2 1D Poisson's equation . . . 48
4.2.1 Dual grid . . . 48
4.2.2 Basis functions . . . 51
4.2.3 Discretization . . . 51
4.2.4 Error analysis . . . 52
4.2.5 Conclusions . . . 56
4.3 2D Poisson's equation . . . 57
4.3.1 Problem formulation . . . 57
4.3.2 Discrete differential forms . . . 57
4.3.3 Computational grid . . . 59
4.3.4 Error analysis . . . 60
4.3.5 Conclusions . . . 62

5 Conclusions 63

6 Future work 65

Bibliography 67

Appendix A 71

List of Figures

1.1 Staggered B-spline bases for velocity and pressure 4
1.2 Dual grid with periodic B-splines 4

2.1 Runge's phenomenon 6
2.2 Recurrence relation between B-splines 8
2.3 Cubic spline and its basis splines. 10
2.4 Open splines 12
2.5 Derivatives of a cubic spline 14
2.6 Nodal basis functions 20
2.7 Edge basis functions 21

3.1 Dual grid with staggered knots 37
3.2 Dual grid of different order 37
3.3 Dual grid closed by the boundary 38

4.1 Grid convergence $\|\epsilon_1\|_{L^2}$ 43
4.2 Grid convergence $\|\epsilon_2\|_{L^2}$ 43
4.3 Grid convergence $\|\epsilon_{tot}\|_{L^2}$ 44
4.4 Local errors in the double projection $\widetilde{\pi} \star \pi\alpha$ 47
4.5 Positive orientations of the vertices and edges 49
4.6 Positive orientations of the vertices and edges 49
4.7 One-dimensional grid A 50
4.8 One-dimensional grid B 50
4.9 Error $\|\epsilon\|_{L^2}$, grid A 54
4.10 Error $\|\epsilon\|_{L^2}$, grid B 54
4.11 Right-hand side function Poisson's equation in $\mathbb{R}^2$ 58
4.12 Two-dimensional grid 60
4.13 Grid convergence $\|\epsilon\|_{L^2}$, Poisson's equation in $\mathbb{R}^2$ 61

A.1 Local errors in the double projection $\widetilde{\pi} \star \pi\alpha$ 71

List of Tables

4.1 Error $\|\epsilon_1\|_{L^2}$ in a double projection . . . 45
4.2 Error $\|\epsilon_2\|_{L^2}$ in a double projection . . . 45
4.3 Error $\|\epsilon_{tot}\|_{L^2}$ in a double projection . . . 45
4.4 Convergence rates ϵ_1 . . . 46
4.5 Convergence rates ϵ_2 . . . 46
4.6 Convergence rates ϵ_{tot} . . . 46
4.7 Error $\|\epsilon\|_{L^2}$, 1D Poisson's equation, grid A . . . 55
4.8 Error $\|\epsilon\|_{L^2}$, 1D Poisson's equation, grid B . . . 55
4.9 Convergence rates $\|\epsilon\|_{L^2}$, 1D Poisson's equation, grid A . . . 55
4.10 Convergence rates $\|\epsilon\|_{L^2}$, 1D Poisson's equation, grid B . . . 55
4.11 Error $\|\epsilon\|_{L^2}$, 2D Poisson's equation . . . 62
4.12 Convergence rates $\|\epsilon\|_{L^2}$, 2D Poisson's equation . . . 62

Nomenclature

$B_{j,k,t}(x)$	B-spline of order k with knots $\mathbf{t}$
C^p	Set of p-cochains
C_p	Set of p-chains
d	Exterior derivative
$\widehat{d\alpha}$	Burke's notation of twisted differential forms
$e_j(x)$	Edge basis function
g	Metric tensor
$h_j(x)$	Nodal basis function
k	Polynomial order (p+1)
$M_{j,k,t}(x)$	M-spline of order k with knots $\mathbf{t}$
p	Polynomial degree
$\mathbf{t}$	Knot sequence
v^i	Contravariant component of a vector $\mathbf{v}$
v_i	Covariant component of of a vector $\mathbf{v}$
δ	Coboundary operator
∂	Boundary operator
Δ	Laplace-Beltrami operator
Λ^p	Space of p-forms
$\widetilde{\Lambda}^p$	Space of twisted p-forms
τ	Interpolation points, e.g. Greville sites
$\mathcal{F}$	Projection on the space of spline differential forms
$\mathcal{I}$	Reconstruction operator
$\mathcal{M}_{k,t}$	Function space spanned by $M_{j,k,t}(x)$
$\mathcal{R}$	De Rham map
$\bar{\mathcal{R}}$	Reduction map to spline coefficients
$\mathcal{S}_{k,t}$	Function space spanned by $B_{j,k,t}(x)$
$\mathbb{R}^n$	n-dimensional Euclidean space
$\wedge$	Exterior product
$\star$	Hodge operator

Chapter 1

Introduction

Partial differential equations (PDEs) describe the laws of continuum physics. The first step in solving the PDEs numerically, is the discretization of the continuum equations. The PDE is translated to a discrete system of algebraic equations, expressed in a finite set of degrees of freedom. An important goal in numerical analysis is to minimize the information lost in translation. The discretization has to be stable, accurate and physically consistent. Finite differences (FD), finite volumes (FV) and finite elements (FE) methods are all capable of approximating the PDE to an arbitrary extent, but may violate some of the conservation laws, leading to unphysical divergence or rotation in vector fields for example.

The issue with conservation is caused by the fact that the numerical solution merely approximates the vector calculus identities. Of course, the numerical solution cannot be expected to exactly comply with every vector identity. Otherwise, one would obtain the exact solution of the problem, and the exact solution is not an element of the typical function space in which one seeks the numerical solution. Nonetheless, discrete models do aim to *mimic* the behaviour of the continuous problem. For instance, FV methods reformulate the PDE into integral equations for individual cells using the discrete divergence theorem. The resulting solution does not exactly satisfy the divergence theorem within cells, but the divergence is, cellwise, contained and controlled by the discrete formulation of the divergence theorem; integral quantities, such as mass, are at least conserved.

In 1988, Hyman and Scovel [31] introduced a far more general discretization approach to mimic the vector calculus identities. The idea is to discretize the general Stokes' theorem, formulated in differential forms [14, 22, 23], and to employ the duality with algebraic topology. Discretizations that adhere to this paradigm are consequently called *mimetic* or *compatible* discretizations. Differential forms replace the classic scalar and vector fields, and their manifolds are the geometric objects on which the problem is studied. The Hodge duals of ordinary differential forms, with an internal orientation, are twisted differential forms, with an external orientation. The duality between the ordinary and twisted forms is naturally discretized by means of a dual grid. The primal grid houses the inner-oriented quantities, and the dual grid the outer-oriented ones. Nevertheless, the use of a dual grid can be circumvented, as demonstrated by Bochev & Gunzburger [8] and Hiptmair [27, 28]. The elimination of the dual grid will however obscure the geometric structure of the physical theory, including the constitutive relations between ordinary and twisted differential forms.

Mimetic discretizations are easily connected to the concept of expansion methods, i.e. the numerical solution is expanded into basis functions. The function space of the solution will be the span of those basis functions. There is a plethora of possible basis functions available. One of the simpler basis functions are the piecewise linear ("hat") functions, which is the classic choice in finite element analysis. Piecewise linear functions feature local support, and in general their implementation is straightforward. Nevertheless, a significant disadvantage are the relatively low resolving capabilities; refining the mesh is not always feasible due to constraints of the computer hardware. Higher-order polynomials, such as Lagrange or Legendre polynomials, are efficient alternatives, but tend to be inflexible because of their global support. B-splines offer an interesting compromise between the two. B-splines are piecewise polynomials of arbitrary degree and combine the flexibility of piecewise linear functions with the resolving capabilities of higher-order polynomials. At present, B-splines have been successfully been applied in mimetic discretizations in which the dual grid is eliminated. The goal of this thesis is to explore the possibilities of dual grids with B-splines. The focus lies therefore on the construction of a discrete Hodge operator. The discretization method is tested for (linear) elliptic boundary value problems. The main question is:

> *How is a discrete Hodge operator constructed when using a dual grid in mimetic discretizations of B-splines?*

Now the global direction of the work is set clear, the following section gives an overview of literature on the subject to support some of the previous statements and to give more details on the current state of relevant research. The second section lastly explains the organization and structure of the report.

1.1 Literature survey

B-splines are attractive basis functions because of their arbitrarily high degree of continuity over the computational domain. For example, turbulent flow computations benefit from increased continuity in the solution space, as observed by Akkerman et al. [1]. The benefits of the high continuity is recognized by Botella [10, 11] as well. The high resolving capabilities of B-splines are confirmed, yielding "spectral-like" solutions of the incompressible Navier-Stokes equations. Botella uses a collocation method in which the residual is set to zero at the collocation points. To prevent spurious modes in the solution the bases for velocity and pressure are staggered, see Fig. 1.1. The B-spline base of the pressure is of one order lower, because no boundary conditions are available for the pressure. The staggered B-spline base could be seen as a precursor for a dual grid in mimetic discretizations.

In many computer-aided design (CAD) packages B-splines or NURBS (non-uniform rational B-splines [33]) are used for generating smooth curves and surfaces. Hughes et al. [17, 30] introduced the concept of isogeometric analysis (IGA), integrating CAD with finite element analysis (FEA). The NURBS functions not only represent the geometry, but also span the solution space. This approach is originally found in isoparametric finite element methods [12]. The workhorse in IGA is the Bubnov-Galerkin method, which in itself does not abide by the conservation laws. Therefore, Buffa et al. [13] developed a mimetic isogeometric method, with applications to electromagnetism. In principle, the method is an extension of the variational methods described by Arnold et al. [2, 3] and Hiptmair [29] to the isogeometric concept. Hiemstra [25] pioneered another mimetic method for IGA with application to elliptic boundary value problems. The method corresponds to the discretization techniques proposed by Bochev & Hyman [9], in which the mass matrix fullfills the role of the discrete Hodge star operator. The methods by Hiemstra and Buffa are similar to Galerkin methods in eliminating the dual grid.

Back & Sonnendrücker [5] use B-spline bases for mimetic discretizations of Maxwell's equations. The dual grid is not eliminated, but realized by staggering the knots, somewhat similar to the staggered grid of Botella. The B-splines are periodic and therefore only appropriate for periodic boundary conditions. The resulting primal and dual grid are illustrated in Fig. 1.2. The periodic staggered base is the only example found in the literature in which B-splines and dual grids are combined in mimetic discretizations.

1.2 Report outline

The main content of this thesis is presented in a trilogy of chapters. First, Chapter 2 introduces the elementary theory of B-splines. The chapter serves as on overview of useful properties and definitions, without much mathematical rigour. Important mathematical tools, such as interpolation and differentiation, are explained and used as references in later chapters. In the end of the chapter, the construction of basis functions with splines for mimetic methods is discussed.

Chapter 3 covers the theoretical framework for mimetic discretizations with B-splines. Possible dual grids for B-splines are discussed, and the discrete operations relevant for mimetic discretizations of elliptic boundary value problems. The dual grid is used to discretize the Hodge ($\star$) operator, which plays an essential role in the construction of a discrete Laplace operator.

Results of numerical computations are presented in Chapter 4. Poisson's equation is solved both in one dimension and two dimensions. The discretizations are validated by comparison by means of a manufactured solution. The main aspect is the convergence of the numerical solution to the exact solution with respect to mesh refinement. The report is then finalized by the conclusions and suggestions for future work.

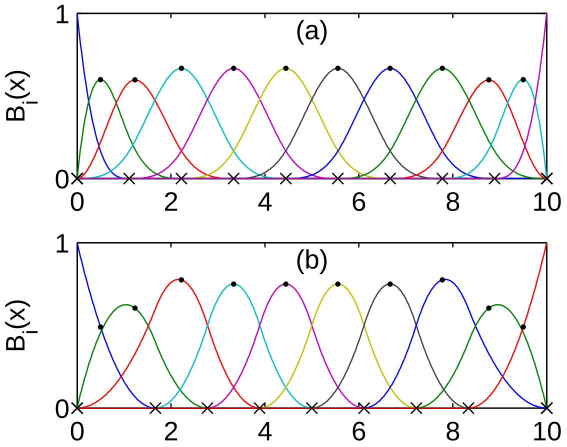


Figure 1.1: Staggered B-spline bases for (a) the velocity, with $k = 4$, and (b) the pressure, with $k = 3$, on a uniform knot sequence ($\times$). The collocation points ($\bullet$) are the maximum of the velocity B-splines. [10]

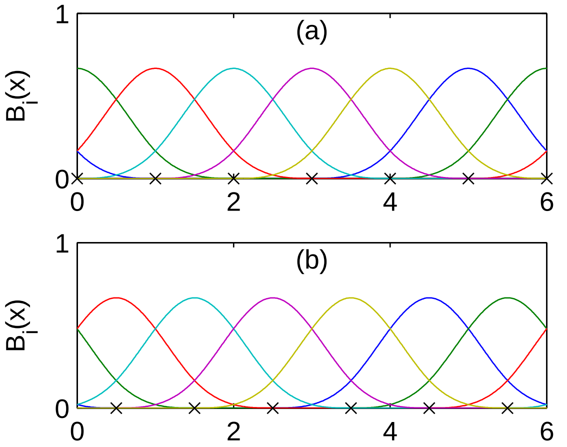


Figure 1.2: Periodic B-splines of $k = 4$ on (a) the primal grid, and (b) the dual grid on a staggered uniform knot sequence ($\times$).

Chapter 2

Fundamentals of splines

In this chapter the notion of splines, and B-splines in particular, is introduced. The goal is mainly to explain the tools used later on in Chapter 3 and to equip newcomers with the appropriate vocabulary dealing with splines. The only prerequisite is knowledge of undergraduate-level calculus and linear algebra. The core of this chapter draws extensively upon the classic handbook written by de Boor [21], which is highly recommended to anyone planning to work with B-splines. Towards the end of the chapter, the focus shifts to constructing basis functions for numerical solutions of differential equations.

2.1 Introduction

A common problem in numerical analysis is the interpolation of a given set of data points, obtained by either sampling or experiments. To approximate the unknown function a high-order polynomial can be fitted through the data points. The resulting interpolant is however susceptible to oscillatory artifacts, known as Runge's phenomenon. Recourse can be found in using piecewise polynomials. The idea for improving accuracy is to increase the number of polynomial pieces, rather than the polynomial order. An example of interpolating equidistant data points is given in Fig. 2.1, where Runge's phenomenon manifests in the polynomial, but is strongly mitigated in the piecewise polynomial.

In 1946, Schoenberg [35] introduced the term *splines* referring to piecewise polynomials as interpolants. The name originates from the aerospace and maritime industry, where splines are the flexible rulers used by draftsmen to produce the smooth curves of a hull or a wing. The splines are fixated at specified coordinates, which bends their shapes to the desired curve. Their mathematical counterparts generate similar shapes when fitted through a set of interpolation points.

Later in 1966, Curry & Schoenberg [19] published a fundamental paper on the mathematical construction of spline functions. In general, a spline is proven to be a linear combination of fundamental splines. Therefore, these fundamental splines are being referred to as *basis* splines or just B-splines. The recognition of B-splines as linear bases paved the way for numerical computations with splines. Unfortunately, the original algorithm for computing B-splines was ill-conditioned and prone to round-off errors. B-splines really matured in 1972 when de Boor [20] developed an algorithm for numerically stable evaluations of B-splines. In the same year, an identical algorithm was found independently by Cox [18]. That is why one commonly refers to the Cox-de Boor algorithm.

Over the years B-splines became a popular choice for interpolating data. Furthermore B-splines are succesfully applied to other problems encountered in numerical analysis, such as the smoothing of noisy data and solving differential equations.

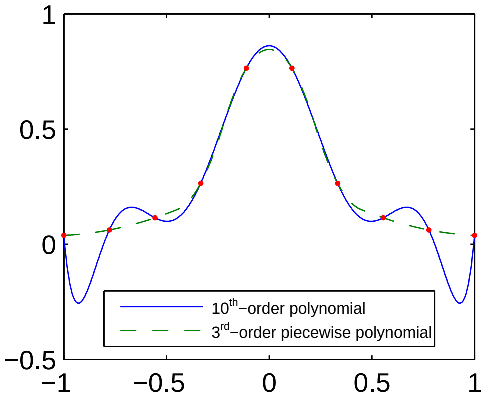


Figure 2.1: Runge's phenomenon in the interpolation of equidistant data points

2.2 Knots and B-splines

The most important splines are the basis splines or B-splines, as they form the basis for all spline functions or piecewise polynomials. The spline functions are defined as linear combinations of B-splines:

$$S(x) = \sum_j a_j B_{j,k}(x), \tag{2.1}$$

where a_j is the j^{th} basis coefficient and $B_{j,k}$ is the j^{th} basis spline of order k. The evaluation of the basis splines starts with defining an increasing sequence of real numbers: the knot vector or knot sequence $\mathbf{t} = \{t_0, ..., t_n\}$. The first-order basis splines are defined as piecewise constant functions on intervals prescribed by the knot sequence $\mathbf{t}$:

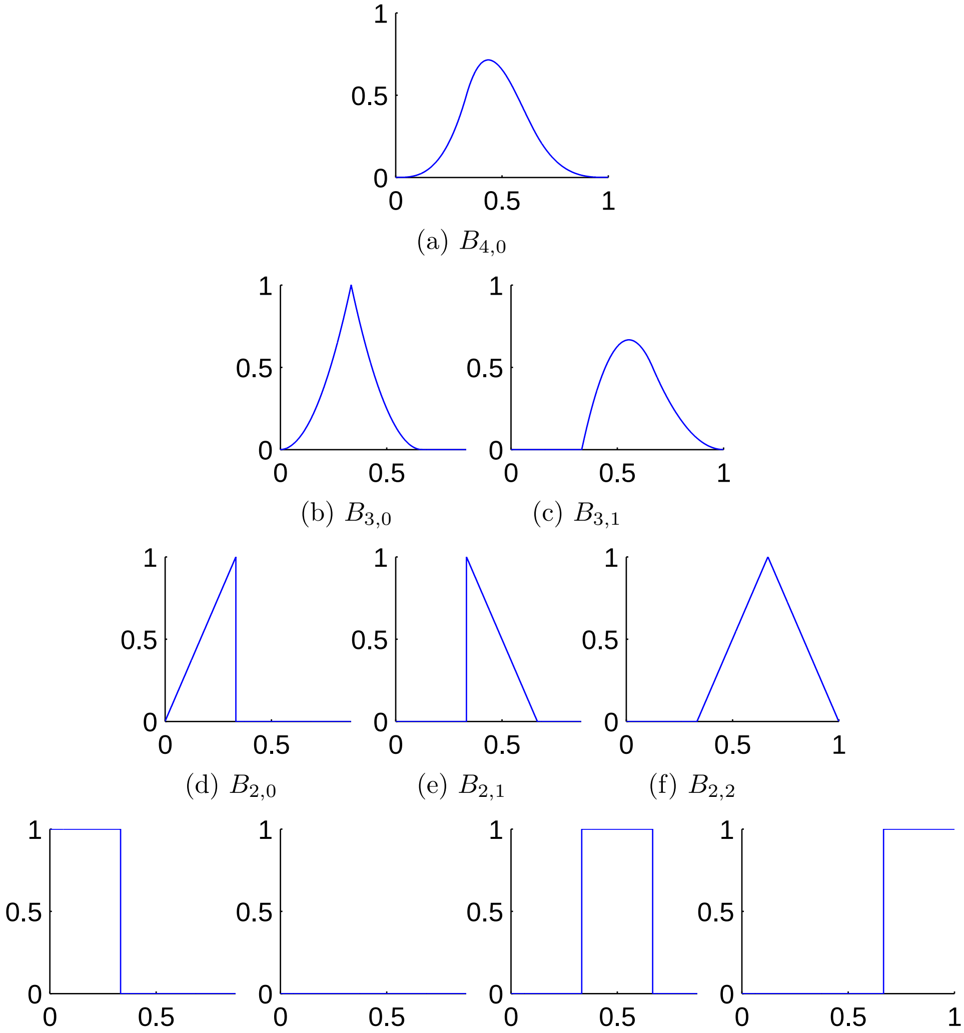
$$B_{i,1} := \begin{cases} 1, & \text{if } t_i \leq x < t_{i+1} \\ 0, & \text{otherwise} \end{cases} . \tag{2.2}$$

Higher-order B-splines are then evaluated in a stable way using Cox-de Boor recurrence relation:

$$B_{j,k} := \frac{x - t_j}{t_{j+k-1} - t_j} B_{j,k-1} + \left(1 - \frac{x - t_j}{t_{j+k-1} - t_j}\right) B_{j+1,k-1}. \tag{2.3}$$

Polynomials are typically classified by their degree p. With regard to B-splines, one commonly refers to the order k, where simply $k = p + 1$. The order k and the knot vector together constitute the B-splines.

Knots are not necessarily uniquely defined and can be repeated up to k times. The multiplicity of the knots prescribes the exact order of continuity in derivatives. An example with duplicate knots is shown in Fig. 2.2, where the recurrence relation constructs a “pyramid” of basis splines. Each time the recurrence relation is applied a new layer of higher-order basis splines is obtained. The maximum order is reached when only one basis spline remains. The multiplicity $t_1 = t_2$ means that $[t_1, t_2) = \emptyset$, in other words $\forall x : B_{1,1}(x) = 0$. This condition propagates in higher-order basis splines as a reduction in continuity.



(a) $B_{4,0}$

(b) $B_{3,0}$ (c) $B_{3,1}$

(d) $B_{2,0}$ (e) $B_{2,1}$ (f) $B_{2,2}$

(g) $B_{1,0}$ (h) $B_{1,1}$ (i) $B_{1,2}$ (j) $B_{1,3}$

Figure 2.2: Pyramid of B-splines generated by the recurrence relation, defined on $\mathbf{t} = \{0, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, 1\}$.

The order k and the knot sequence $\mathbf{t}$ define the B-splines, and thus span the function space of the spline $\mathcal{S}_{k,t}$. In this space spline functions are formed by the linear combination of B-splines. Let the formal definition of the space of spline functions $\mathcal{S}_{k,t}$ be:

$$\mathcal{S}_{k,t} := \operatorname{span}(B_{k,t}) = \left\{ \sum_j a_j B_{j,k,t} : a_j \in \mathbb{R} \right\}. \tag{2.4}$$

The polynomial pieces that comprise the spline are joined smoothly at the knots, in the sense that derivatives are continuous up to a certain order. For example, a cubic spline has continuous second order derivatives. In general the continuity over the knots is C^{k-2}. The polynomial pieces between knot intervals are infinitely smooth (C^{∞}) like regular polynomials.

Lemma 1. *Suppose the knot sequence* $\mathbf{t}$ *consists of unique knots. Then* $\mathcal{S}_{k,t} \subset C^{k-2}$. *A function* $f \in \mathcal{S}_{k,t}$ *has continuous derivatives of order* $(k-2)$ *and lower.*

Increasing the multiplicity of a knot reduces the local continuity of the spline over that particular knot. The continuity of the spline over the global domain is then equal to the lowest degree of continuity occuring in the domain.

Lemma 2. *Suppose a knot* $t \in \mathbf{t}$ *has multiplicity* m. *Then the continuity of* $f \in \mathcal{S}_{k,t}$ *over* t *is* C^{k-m-1}.

A spline based on a knot sequence of unique knots, always equals zero at the boundaries of the domain. To improve the flexibility of the spline, the local continuity of spline over the boundary knots must be decreased to C^{-1}. Hence the knot multiplicity m must be equal to k, recalling that the local continuity is C^{k-m-1}. The corresponding knot sequence is called *open* and is generally defined as:

$$\mathbf{t} = [\underbrace{t_0, ..., t_0}_{k}, t_1, ..., t_{n-1}, \underbrace{t_n, ..., t_n}_{k}]. \tag{2.5}$$

As the knots are multiplied by k at the boundaries only, the global continuity of the spline is not compromised. In practice most B-splines are defined on open knot sequences. An example of an open cubic spline and its basis splines is shown in Fig. 2.3.

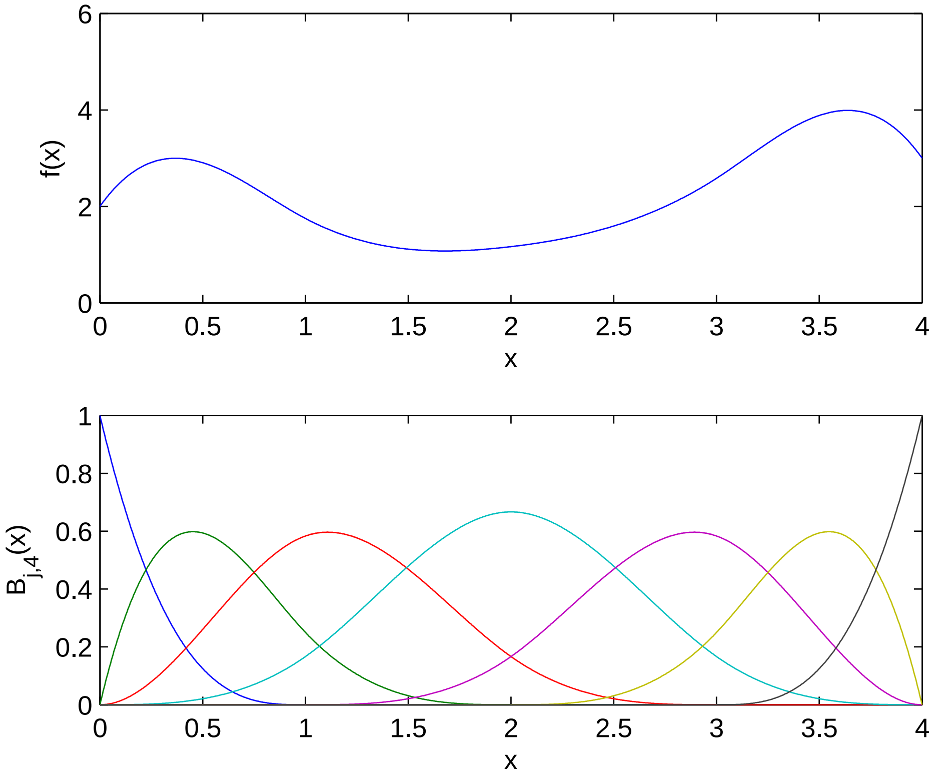


Figure 2.3: A cubic spline and its basis functions. With $\mathbf{t} = \{0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4\}$ and $\mathbf{a} = \{2, 4, 1, 1, 2, 5, 3\}$

2.3 Control points

The basis coefficients shape the B-spline function $f \in \mathcal{S}_{k,t}$, but do not represent function values, as the basis functions of order $k \geq 2$ in general do not satisfy the interpolation condition $B_j(x_i) = f(x_i)$. Although the coefficients do not represent function values, they can be visualised in the form of control points. These control points give a reasonable approximation of the spline function and illustrate how the coefficients "control" the spline. Relevant to control points are the knot averages, called the Greville sites:

$$\tau_{j,k} = \frac{t_{j+1} + ... + t_{j+k-1}}{k-1}. \tag{2.6}$$

The Greville sites have the property:

$$x = \sum_j \tau_{j,k} B_{j,k}(x).$$

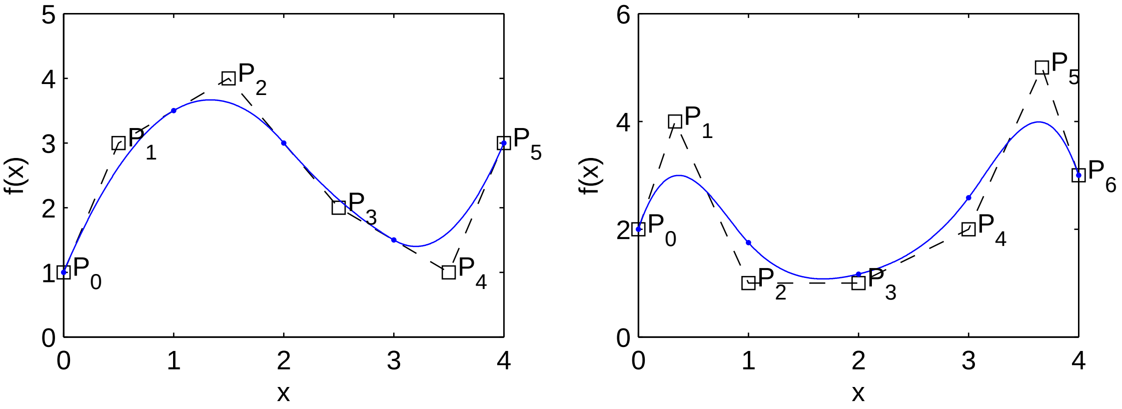
The coordinates of the control points P_j are then defined as:

$$P_j := (\tau_j, a_j) \in \mathbb{R}^2. \tag{2.7}$$

At last, the control polygon is obtained by linearly interpolating the control points. The control polygon, consisting of broken lines, is a coarse approximation to a spline. An important feature of B-splines is the convex hull property. The spline is bounded to the convex hull of the control polygon, that is the function can not overshoot the control polygon. For $x \in [t_i, t_{i+1}]$ this means the "nearby" coefficients mark the limits:

$$\min\{a_{i+1-k}, ..., a_i\} \leq f(x) \leq \max\{a_{i+1-k}, ..., a_i\}.$$

Example For example a quadratic and a cubic B-spline function with their control polygons are shown in Fig 2.4. Both splines are defined on an open knot sequence. Hence the first and the last basis coefficients prescribe the boundary values of the spline. For these points the basis coefficients *do* represent a function value. The corresponding control points coincide with the function itself.



(a) A quadratic spline and its control polygon. With $\mathbf{t} = \{0, 0, 0, 1, 2, 3, 4, 4, 4\}$ and $\mathbf{a} = \{1, 3, 4, 2, 1, 3\}$

(b) A cubic spline and its control polygon. With $\mathbf{t} = \{0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4\}$ and $\mathbf{a} = \{2, 4, 1, 1, 2, 5, 3\}$

Figure 2.4: B-spline functions on open knot sequences. The position of the knots is indicated by (●).

2.4 Differentiation

The linearity of B-spline functions provide simple means for exact differentiation. Consider the B-spline function $f \in \mathcal{S}_{k,t}$:

$$f(x) = \sum_{j=0}^{m-k} a_j B_{j,k,t}(x).$$

Here the basis functions $B_{j,k,t}$ are defined on an arbitrary knot sequence:

$$\mathbf{t} = \{t_0, ..., t_m\}.$$

Using the same knot sequence, the function $g \in \mathcal{S}_{k-1,t}$ is:

$$g(x) = \sum_{j=0}^{m-k+1} b_j B_{j,k-1,t}(x).$$

Differentiating $f(x)$ with respect to x gives however:

$$\frac{df}{dx} = \sum_{j=1}^{m-k} \frac{k-1}{t_{j+k-1} - t_j}(a_{j+1} - a_j) B_{j,k-1,t}(x). \tag{2.8}$$

So the first and last basis functions of $\mathcal{S}_{k-1,t}$ vanish in the derivative. Therefore the basis functions $B_{j,k-1}$ can be defined on a new knot sequence, called the reduced knot sequence:

$$\mathbf{t}' = \{t_1, ..., t_{m-1}\},$$

where the first and the last knot have been removed from $\mathbf{t}$. Hence the derivative of $f(x)$ can be expressed as a spline function in $\mathcal{S}_{k-1,t'}$:

$$\frac{df}{dx} = \sum_{j=0}^{m-k-1} b_j B_{j,k-1,t'}(x), \tag{2.9}$$

where the basis coefficients are found from $\mathbf{t}$:

$$b_j = \frac{k-1}{t_{j+k} - t_{j+1}}(a_{j+1} - a_j) \quad : j \in [0, ..., m-k-1]. \tag{2.10}$$

As t_0 and t_m remain unused, an equivalent expression would be:

$$b_j = \frac{k-1}{t'_{j+k-1} - t'_j}(a_{j+1} - a_j) \quad : j \in [0, ..., m-k-1]. \tag{2.11}$$

An example of a cubic B-spline and its derivatives is shown in Fig. 2.5. The double knot at $x = 1$ reduces the continuity of the spline to C^1, causing a discontinuity in the second derivative. The double knot gives a singular point in the third derivative.

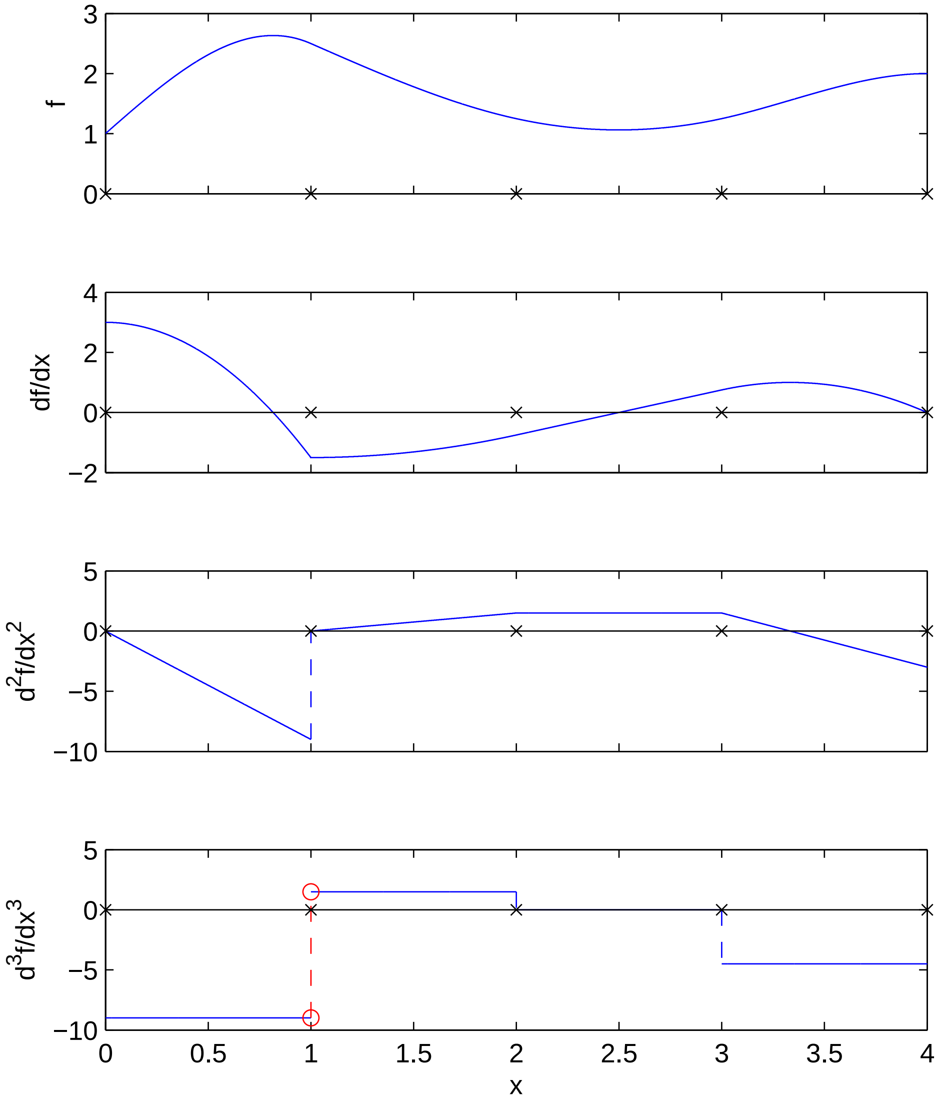


Figure 2.5: A cubic B-spline function $f(x)$ with $\mathbf{t} = \{0, 0, 0, 0, 1, 1, 2, 3, 4, 4, 4, 4\}$ and $\mathbf{a} = \{1, 2, 3, 2, 1, 1, 2, 2\}$, and its derivatives. Note the influence of the double knot at $x = 1$ on the smoothness of the derivatives.

2.5 Integration

The indefinite integral of a B-spline is:

$$\int_{-\infty}^{\infty} B_{j,k}(x)dx = \frac{t_{j+k} - t_j}{k}. \tag{2.12}$$

The indefinite integral of a B-spline function $f \in \mathcal{S}_{k,t}$ by using the linearity of the integration operator:

$$\int_{-\infty}^{\infty} \sum_j a_j B_{j,k}(x)dx = \sum_j a_j \int_{-\infty}^{\infty} B_{j,k}(x)dx = \sum_j a_j \frac{t_{j+k} - t_j}{k}. \tag{2.13}$$

2.6 M-splines

The B-splines introduced by Curry & Schoenberg [19] in 1966 have a different scaling than the contemporary B-splines. The original B-splines are commonly referred to as Curry-Schoenberg B-splines, or more fashionably M-splines, as the basis functions are denoted by $M_{k,t}$. Let the space of M-splines, $\mathcal{M}_{k,t}$, then be defined as:

$$\mathcal{M}_{k,t} := \text{span}(M_{k,t}) = \left\{ \sum_j a_j M_{j,k,t} : a_j \in \mathbb{R} \right\}. \tag{2.14}$$

M-splines differ from B-splines in the normalisation of the basis splines. M-splines are normalised namely such that:

$$\int_{-\infty}^{\infty} M_{j,k}(x)dx = 1. \tag{2.15}$$

Combined with Eq. (2.12) this leads to the definition:

$$M_{j,k}(x) := \frac{k}{t_{j+k} - t_j} B_{j,k}(x). \tag{2.16}$$

M-splines are simply scaled B-splines, making their numerical evaluation straightforward. The notion of M-splines allows a new expression for B-spline derivatives:

$$\frac{df}{dx} = \frac{d}{dx} \sum_{j=0}^{n} a_j B_{j,k,t} = \sum_{j=0}^{n-1} (a_{j+1} - a_j) M_{j,k-1,t'}. \tag{2.17}$$

In the context of numerical analysis, this relation proves to be of great value in the discrete formulations of Stokes' theorem discussed in Section 3.2.

2.7 Interpolation

Suppose we want to approximate a scalar function $f(x)$ by interpolation at specified nodes. The approximation function $f_h(x)$ is assumed to be spline function:

$$f_h(x) = \sum_{i=1}^{n} a_i B_i(x), \tag{2.18}$$

where the basis function B_i are defined on a knot sequence $\mathbf{t}$. The interpolation points are denoted by $\tau = [\tau_1, ..., \tau_m]$. If the interpolation points are to be chosen freely, the Greville points, Eq. (2.6), are recommended. At those nodes the approximation function equals the exact function:

$$f(\tau_i) = f_h(\tau_i) = \sum_{i=1}^{n} a_i B_i(\tau_i). \tag{2.19}$$

Combining the conditions for each interpolation point gives a linear system of equations:

$$\begin{pmatrix} f(\tau_1) \\ \vdots \\ f(\tau_m) \end{pmatrix} = \begin{pmatrix} B_0(\tau_1) & \cdots & B_n(\tau_1) \\ \vdots & \ddots & \vdots \\ B_0(\tau_m) & \cdots & B_n(\tau_m) \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}, \tag{2.20}$$

which defines the collocation matrix B:

$$B_{ij} := \begin{pmatrix} B_1(\tau_1) & \cdots & B_n(\tau_1) \\ \vdots & \ddots & \vdots \\ B_1(\tau_m) & \cdots & B_n(\tau_m) \end{pmatrix}. \tag{2.21}$$

For an invertible system m must be equal n, in other words the number of nodes must equal the number of basis functions. Furthermore, each basis function must have at least one node lying in its span to prevent empty columns of B. The conditions for a solution to exist are formally stated by Schoenberg's theorem, which requires $B_j(\tau_j) \neq 0$. Due to the local support of the basis functions B is a banded matrix and can be solved efficiently. If $m = n$, then the linear system can be solved for unknown coefficients a_i:

$$\mathbf{a} = B^{-1}\mathbf{f}. \tag{2.22}$$

Alternatively, the linear system of equations can be solved using Gaussian elimination.

2.8 Histopolation

Interpolation approximates a function by preserving function values at the points of interpolation. Histopolation is a different approximation method and preserves integrals of the function over the intervals of histopolation: the total integral of the function is preserved. The approximation function f_h is a linear combination of m M-splines:

$$f_h(x) = \sum_{j=1}^{m} a_j M_j(x).$$

The advantage of using M-splines instead of B-splines, is the simple evaluation of integrals (Eq. (2.25)). Also, the relation between B- and M-splines comes to good use in the discrete formulation of the generalized Stokes' theorem in Section 3.2. The functional domain is subdivided into n intervals L_i on which integrals are preserved. The histopolation condition for an interval is:

$$\int_{L_i} f(x)dx = \int_{L_i} \sum_{j=1}^{m} a_j M_j(x)dx = \sum_{j=1}^{m} a_j \int_{L_i} M_j(x)dx. \tag{2.23}$$

The histopolation conditions on the intervals poses a linear system of equations:

$$\begin{pmatrix} \int_{L_1} f(x)dx \\ \vdots \\ \int_{L_n} f(x)dx \end{pmatrix} = \begin{pmatrix} \int_{L_1} M_1(x)dx & \cdots & \int_{L_1} M_m(x)dx \\ \vdots & \ddots & \vdots \\ \int_{L_n} M_1(x)dx & \cdots & \int_{L_n} M_m(x)dx \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix}. \tag{2.24}$$

In this case, the collocation matrix M is an $n \times m$ matrix with entries:

$$M_{ij} = \int_{L_i} M_j(x)dx.$$

Lemma 3. *The integrals on $[a, b]$ of the M-splines of order k are calculated by evaluating the B-splines of order $(k + 1)$ at the interval boundaries:*

$$\int_a^b M_{j,k}(x)dx = \sum_{j=1}^{i} [B_{i,k+1}(a) - B_{i,k+1}(b)]. \tag{2.25}$$

Proof. The fundamental theorem of calculus states that for any function $F = f'$ on $[a, b]$:

$$\int_a^b f(x)dx = F(b) - F(a). \tag{2.26}$$

Substitution of a B-spline of order $(k + 1)$ gives:

$$\int_a^b \frac{d}{dx} B_{j,k+1}(x)dx = B_{j,k+1}(b) - B_{j,k+1}(a). \tag{2.27}$$

According to Eq. (2.17), the derivative of a B-spline is:

$$\frac{d}{dx} B_{j,k+1}(x) = M_{j-1,k}(x) - M_{j,k}(x). \tag{2.28}$$

Substituting of (2.28) in (2.27) results in:

$$\int_a^b [M_{j-1,k}(x) - M_{j,k}(x)]dx = B_{j,k+1}(b) - B_{j,k+1}(a).$$

Rearranging the terms gives a recursive relation for the definite integral:

$$\int_a^b M_{j,k}(x)dx = B_{j,k+1}(a) - B_{j,k+1}(b) - \int_a^b M_{j-1,k}(x)dx. \tag{2.29}$$

An explicit expression is obtained by summation of individual terms:

$$\begin{aligned} \int_a^b M_1(x)dx &= B_{1,k+1}(a) - B_{1,k+1}(b), \\ \int_a^b M_2(x)dx &= B_{2,k+1}(a) - B_{2,k+1}(b) - [B_{1,k+1}(a) - B_{1,k+1}(b)], \\ \ldots &= \ldots \\ \int_a^b M_{j,k}(x)dx &= \sum_{i=1}^{j} [B_{i,k+1}(a) - B_{i,k+1}(b)]. \end{aligned}$$

□

The histopolation problem can only be solved under the right conditions and is directly related to an interpolation problem.

Theorem 1. *Consider the histopolant* $f_h(x) = \sum_{j=1}^{m} a_j M_{j,k}(x)$ *on* $\mathbf{t}_k$ *with an antiderivative* $F(x) = \sum_{j=1}^{m+1} b_j B_{j,k+1}(x)$ *on* $\mathbf{t}_{k+1}$ *and the interval boundaries of integration are* $\{\tau_1, ..., \tau_{m+1}\}$*. The histopolation problem is solvable if* $F(\tau_i)$ *satisfies the Schoenberg-Whitney theorem, i.e.* $B_{j,k+1}(\tau_j) \neq 0$*.*

Proof. The histopolant is constructed from m M-splines and the histopolation is to be satisfied on m intervals defined by τ: the collocation matrix is an $m \times m$ matrix. Then, the system of histopolation equations, $M\mathbf{a} = \mathbf{f}$, is solvable when the unique solution to the homogeneous system, $M\mathbf{a} = 0$, is the trivial solution $\mathbf{a} = \mathbf{0}$. The histopolation condition for the homogeneous problem is:

$$\int_{\tau_i}^{\tau_{i+1}} f_h(x)dx = F(\tau_{i+1}) - F(\tau_i) = 0. \tag{2.30}$$

In other words, the antiderivative must satisfy:

$$F(\tau_i) = F(\tau_{i+1}). \tag{2.31}$$

An integration constant appears in the indefinite integral of $f_h(x)$:

$$\int f_h(x)dx = F(x) + C.$$

Considering the arbitrary constant, the condition (2.31) imposed on the antiderivative is equivalent to:

$$F(\tau_i) = F(\tau_{i+1}) = 0.$$

The antiderivative must be the solution of the interpolation problem:

$$F(\tau_i) = \sum_{j=1}^{m+1} b_j B_{j,k+1}(\tau_i) = 0,$$

which gives the linear system $B\mathbf{b} = \mathbf{0}$. If the interpolation points are chosen such that $B_{j,k+1}(\tau_j) \neq 0$, then the Schoenberg-Whitney theorem applies. As a consequence, the only solution to $B\mathbf{b} = \mathbf{0}$ is the trivial solution $\mathbf{b} = 0$, which induces:

$$a_j = b_{j+1} - b_j = 0.$$

Thus, $M\mathbf{a} = \mathbf{0}$ has indeed the only solution $\mathbf{a} = 0$. Therefore, the nonhomogeneous histopolation problem has a solution of the form $\mathbf{a} = M^{-1}\mathbf{f}$.

□

2.9 Basis functions

Basis splines can be deployed as basis functions for numerical solutions of differential equations. In general, the numerical solution u_h is a linear combination of basis functions $\Phi_j(x)$, in other words: $u_h(x) = \sum a_j \Phi_j(x)$. In the case of basis splines, the basis coefficients are generally not equal to physical quantities: their values represent neither function values nor function integrals. The spline function must then be evaluated to obtain any physical quantities. In mimetic spectral element methods [24, 32] however, nodal and edge basis functions are used, of which the basis coefficients respectively represent function values and function integrals. Comparable nodal and edge basis functions can be constructed as spline functions, i.e. those basis functions are linear combinations of basis splines themselves.

2.9.1 Nodal basis functions

If a function $f_h(x)$ is expanded into nodal basis functions $h_i(x)$, then the basis coefficients equal function values at the interpolation points. Prime examples of nodal basis functions are piecewise-linear polynomials and Lagrange polynomials. Nodal basis functions are defined by the interpolation condition:

$$h_i(\tau_j) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases} \tag{2.32}$$

The nodal basis function $h_i(x)$ is constructed from m B-splines:

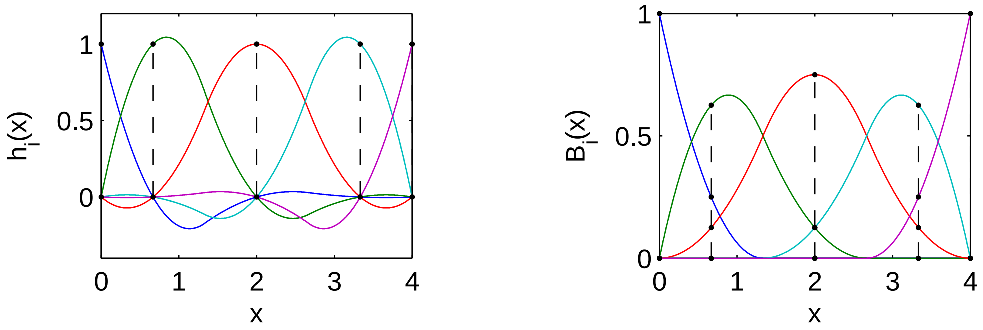
$$h_i(x) = \sum_{i=1}^{m} a_j B_j(x). \tag{2.33}$$

To obtain the coefficient a_j, the interpolation problem posed by Eq. (2.32) must solved. For m B-splines, there must be m interpolation points τ_j, such as the Greville sites. The problem is formulated as a linear system of equations (see Section 2.7 on interpolating functions with B-splines):

$$\begin{pmatrix} h_i(\tau_1) \\ \vdots \\ h_i(\tau_m) \end{pmatrix} = \begin{pmatrix} B_1(\tau_1) & \cdots & B_m(\tau_1) \\ \vdots & \ddots & \vdots \\ B_1(\tau_m) & \cdots & B_m(\tau_m) \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix}. \tag{2.34}$$

The basis coefficients of $h_i(x)$ are given by the solution of the form $\mathbf{a} = B^{-1}\mathbf{h}$. Each basis function is associated to a node, or interpolation point. The condition (2.32) imposed on all nodal basis functions, gives an $m \times m$ identity matrix: $I_m = (\delta_{ij})_{i,j=1}^{m}$. The complete set of nodal basis functions can therefore be obtained efficiently from the solution $B^{-1}I_m$, or simply calculating the inverse B^{-1}.

Example. B-splines of order $k = 2$ are piecewise-linear polynomials, and thus nodal basis functions. For higher orders this no longer the case. An example is shown in Fig. 2.6, where nodal basis functions are built from quadratic B-splines. The resulting nodal basis functions are supported by the entire knot span, whereas B-splines have local knot spans as support.



(a) Quadratic nodal basis functions.

(b) Quadratic B-splines.

Figure 2.6: Nodal basis functions $h_i(x)$ and B-splines $B_i(x)$ of order $k = 3$. The knot sequence is $\mathbf{t} = [0, 0, 0, 1, 2, 3, 4, 4, 4]$ and the interpolation points ($\bullet$) are Greville sites $\tau = [0, \frac{2}{3}, 2, 3\frac{1}{3}, 4]$.

2.9.2 Edge basis functions

If a function $f_h(x)$ is expanded into the edge basis functions $e_i(x)$, then the basis coefficients equal function integrals over histopolation intervals. Their construction is similar to that of nodal basis functions. The edge basis functions are defined by the histopolation condition:

$$\int_{L_j} e_i(x)dx = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases} \tag{2.35}$$

The nodal basis function $e_i(x)$ is constructed from m M-splines:

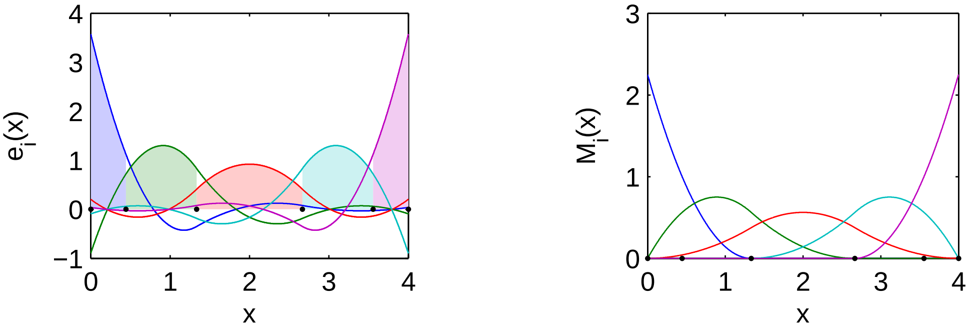
$$e_i(x) = \sum_{i=1}^{m} a_j M_j(x). \tag{2.36}$$

To obtain the coefficient a_j, the histopolation problem posed by Eq. (2.35) must solved. For m B-splines, there must be m histopolation intervals L_j, which must comply with Theorem 1. The problem is formulated as a linear system of equations (see Section 2.8 on histopolating functions with M-splines):

$$\begin{pmatrix} \int_{L_1} e_i(x)dx \\ \vdots \\ \int_{L_m} e_i(x)dx \end{pmatrix} = \begin{pmatrix} \int_{L_1} M_1(x)dx & \cdots & \int_{L_1} M_m(x)dx \\ \vdots & \ddots & \vdots \\ \int_{L_m} M_1(x)dx & \cdots & \int_{L_m} M_m(x)dx \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix}. \tag{2.37}$$

The basis coefficients of $e_i(x)$ are given by the solution of the form $\mathbf{a} = M^{-1}\mathbf{e}$. Each basis function is associated to an edge, or histopolation interval. The condition (2.35) imposed on all edge basis functions, gives an $m \times m$ identity matrix: $I_m = (\delta_{ij})_{i,j=1}^{m}$. The complete set of edge basis functions can therefore be obtained efficiently from the solution $M^{-1}I_m$, or simply calculating the inverse M^{-1}.

Example. In Fig. 2.7 edge basis functions are shown, which are obtained from quadratic M-splines. Their support is the entire knot span, as is the case for nodal basis functions. The integrals of the edge basis functions evaluate to 1 over the intervals associated to their own, and to 0 over the intervals of others.



(a) Quadratic edge basis functions.

(b) Quadratic M-splines.

Figure 2.7: Edge basis functions $e_i(x)$ and M-splines $M_i(x)$ of order $k = 3$. The knot sequence is $\mathbf{t} = [0, 0, 0, 1, 2, 3, 4, 4, 4]$ and histopolation intervals are given by the Greville sites (●) for a knot sequence of one order higher: $\tau = [0, \frac{4}{9}, 1\frac{1}{3}, 2\frac{2}{3}, 3\frac{5}{9}, 4]$. The unit integrals of each edge basis function are coloured.

Alternatively, edge basis functions can be obtained from a given set of nodal basis functions of one order higher [24]:

$$e_i(x) = -\sum_{j=1}^{i} \frac{dh_j}{dx}. \tag{2.38}$$

The mimetic spectral element method deploys nodal and edge basis functions. In Chapter 3.2.2, they are proven not to be an absolute necessity for mimetic methods. B- and M-splines are equally suited as basis functions, although their basis coefficient do not represent physical quantities. In both cases, the numerical solution remains unchanged, varying only in its expression in basis functions.

Chapter 3

Mimetic discretizations

In Chapter 2 splines are introduced as basis functions for numerical solutions of partial differential equations. This chapter presents a framework for mimetic discretizations using splines. First, the mathematical background of mimetic discretizations is introduced, covering the elementary operators in differential geometry and algebraic topology. Then, the discrete counterparts of the analytic operators are presented. The key operator is the Hodge $\star$ operator, which is necessary for discretizing the Laplacian. Applications to Poisson's equation and results follow in Chapter 4.

3.1 Introduction

In mimetic discretizations the connection between the generalized Stokes' theorem and algebraic topology is employed in order to obtain conservative solutions. Stokes' theorem is formulated in the language of differential geometry, which generalizes vector calculus. This section gives a brief overview of differential forms and algebraic topology upon which mimetic discretizations are founded.

3.1.1 Differential forms

Differential geometry generalizes vector calculus in a way that is independent of coordinate systems. Scalar and vector fields are represented by differential forms. A general p-form can be integrated over p-dimensional geometric objects, known as manifolds. This subsection contains a brief overview of ordinary and twisted differential forms, and their connection by the Hodge duality. All the mathematical definitions presented are extracted from the self-containing introductory books on differential geometry by Flanders [22] and Frankel [23], and in particular by Burke [14] for his elaborate treatment of twisted forms.

Ordinary differential forms

Dual vector space
A vector space E spanned by the tangent vectors $\frac{\partial}{\partial x^j}$ has a dual space spanned by linear functionals dx^i, if the manifold is equipped with a metric. The components of a vector are *contravariant* with respect to a change of basis, whereas those of a dual vector are *covariant*. The covariant components of the dual to contravariant vector $\mathbf{v} \in E$ follow

from:

$$v_j = \sum_i v^i g_{ij}. \tag{3.1}$$

where g is the metric tensor. Contravariant components have superscript indices, and covariant ones have subscript indices. The components of the metric tensor are defined by the inner products of the basis vectors:

$$g_{ij} := \left\langle \frac{\boldsymbol{\partial}}{\boldsymbol{\partial} x^i}, \frac{\boldsymbol{\partial}}{\boldsymbol{\partial} x^j} \right\rangle. \tag{3.2}$$

In an n-dimensional orthonormal basis the metric tensor simply evaluates to the identity matrix I_n. The covariant and contravariant components are equal then, $v_i = v^i$. A general representation of covariant vector, or *covector*, fields in $\mathbb{R}^3$ is:

$$\alpha = \alpha_1(x,y,z)dx + \alpha_2(x,y,z)dy + \alpha_3(x,y,z)dz, \tag{3.3}$$

of which the components are functions of the spatial coordinates x, y and z. Differential 1-forms are defined as covariant vector fields. The space of 1-forms is denoted by Λ^1, and for generalized p-forms by Λ^p.

Exterior product

The exterior product or wedge product, denoted by $\wedge$, is the product between two differential forms α^p and β^q, resulting in γ^{p+q}:

$$\wedge : \quad \Lambda^p \wedge \Lambda^q \mapsto \Lambda^{p+q}. \tag{3.4}$$

The wedge product is skew symmetric and associative:

$$\alpha^p \wedge \beta^q = (-1)^{pq} \beta^q \wedge \alpha^p, \tag{3.5a}$$

$$(\alpha \wedge \beta) \wedge \gamma = \alpha \wedge (\beta \wedge \gamma). \tag{3.5b}$$

It is also bilinear:

$$(a_1\alpha_1 + a_2\alpha_2) \wedge \beta = a_1(\alpha_1 \wedge \beta) + a_2(\alpha_2 \wedge \beta), \tag{3.5c}$$

$$\alpha \wedge (b_1\beta_1 + b_2\beta_2) = b_1(\alpha \wedge \beta_1) + b_2(\alpha \wedge \beta_2). \tag{3.5d}$$

Exterior derivative

The exterior derivative, denoted by d, maps a p-form into a $(p+1)$-form:

$$d : \quad \Lambda^p \mapsto \Lambda^{p+1}, \tag{3.6}$$

satisfying the conditions:

$$d(\alpha + \beta) = d\alpha + d\beta, \tag{3.7a}$$

$$d\alpha^0 = \sum_i \frac{\partial \alpha^0}{\partial x_i} dx_i, \tag{3.7b}$$

$$d(\alpha^p \wedge \beta^q) = d\alpha^p \wedge \beta^q + (-1)^p \alpha^p \wedge d\beta^q, \tag{3.7c}$$

$$d(d\alpha) = 0, \quad \forall \alpha. \tag{3.7d}$$

The exterior derivative connects the spaces of p-forms in a sequence called the De Rham complex, for example in $\mathbb{R}^3$:

$$\mathbb{R} \hookrightarrow \Lambda^0(\Omega) \xmapsto{d} \Lambda^1(\Omega) \xmapsto{d} \Lambda^2(\Omega) \xmapsto{d} \Lambda^3(\Omega) \longmapsto 0. \tag{3.8}$$

Twisted differential forms

Descriptions of ordinary differential forms are independent from the orientation of the vector space, whereas twisted differential forms do *change sign with a change of orientation.* For example, the magnetic field **B** circulates about a current-carrying wire. The direction of circulation is in the sense of a right-hand screw when the common right-handed orientation is positively defined, or in the sense of a left-hand screw, when the left-handed orientation is positively defined; both orientations are equally valid choices. The ambiguity in orientation is removed by prescribing an orientation Ω to an ordinary form α, so twisted forms are written as (α, Ω). The explicit notation of twisted forms is often neglected by tacitly presuming a right-handed orientation, as most engineers and physicists are accustomed by this kind of orientation. In many texts on differential geometry twisted forms are overlooked for that reason. A notable exception is the work of Burke [14, 15, 16], which provides a more thorough discussion of twisted forms.

The orientation of an n-dimensional space can be specified by n factors on the monomial $\Omega = d\alpha$. The orientation $d\alpha$ is written in curly brackets and has the equivalence property:

$$\{d\alpha\} \equiv \{k d\alpha\}, \quad k > 0, \tag{3.9}$$

and the equality:

$$\{-d\alpha\} = -\{d\alpha\}. \tag{3.10}$$

The orientation $\{d\alpha\}$ is paired to a differential form by multiplication. The $\wedge$ symbol for the exterior product is omitted for brevity.

Example. In Euclidean two-space the twisted 2-form is:

$$\{dx\, dy\}\, dx\, dy.$$

And the twisted 1-forms are:

$$\{dx\, dy\}\, dx$$

$$\{dx\, dy\}\, dy$$

Here, the right-handed orientation $\{dx\, dy\}$ of $\mathbb{R}^2$ is employed.

Ordinary forms have an inner orientation, whereas twisted forms have an *transverse* or *outer* orientation with respect to the manifold. For instance, a twisted 1-form represents the flux across a line in $\mathbb{R}^2$, and the circulation about a line in $\mathbb{R}^3$. The transverse orientation $\{\beta\}$ of the twisted form α follows from the relation:

$$\{\beta \wedge \alpha\} = \{\Omega\}\,. \tag{3.11}$$

Example. The orientation of $\{dx\, dy\}dy$ is:

$$\{\beta \wedge dy\} = \{dx\, dy\} \to \{\beta\} = \{dx\}\,,$$

which is the flux in the *plus* x-direction. However, the orientation of $\{dx\, dy\}dx$ is:

$$\{\beta \wedge dx\} = \{dx\, dy\} \to \{\beta\} = \{-dy\}\,,$$

which is the flux in the *minus* y-direction. The counterintuitive minus sign could be resolved by assigning the orientation $\{dy\, dx\}dx$, such that a positive flux is in the positive y-direction.

Writing the orientations of the coordinate system in brackets clutters the notation of twisted forms. Burke proposed a rather elegant notation as an alternative, following these rules:

1. denote all of the missing basis forms by writing them in with a hat over them;
2. arrange the factors, changing the sign if necessary, so that they are in the same order as the factors in the orientation;
3. since the orientation information is now redundant, leave it out;
4. use the hat notation only for twisted forms;
5. the twisted n-form has nothing left out, write it with a hat over the space in front of it.

Example. The alternative notation for twisted 1-forms is:

$$\{dx\,dy\}\,dy = \{dx\,dy\}\,\widehat{dx}\,dy = \widehat{dx}\,dy,$$

$$\{dy\,dx\}\,dx = \{dy\,dx\}\,\widehat{dy}\,dx = \widehat{dy}\,dx,$$

and for the twisted 2-form:

$$\{dx\,dy\}dx\,dy = \{dx\,dy\}\widehat{\ }dx\,dy = \widehat{\ }dx\,dy.$$

Example. In Euclidean three-space the twisted 2-forms can be written:

$$\begin{aligned}\widehat{dx}\,dy\,dz &= \widehat{dx}\,dz\,dy,\\ \widehat{dy}\,dx\,dz &= \widehat{dy}\,dz\,dx,\\ \widehat{dz}\,dx\,dy &= \widehat{dz}\,dy\,dx,\end{aligned}$$

which represent the fluxes in x-, y- and z-directions respectively.

Thus, the order of the explicit factors contains no essential information. Their redundancy calls for the final rule of the shorthand notation:

6. leave out the explicit factors.

Example. The twisted forms in the ultimate shorthand become:

$$\begin{aligned}\widehat{dx}\,dy &= \widehat{dx},\\ \widehat{dy}\,dx &= \widehat{dy},\\ \widehat{\ }dx\,dy &= \widehat{1}.\end{aligned}$$

Also for twisted forms there is a De Rham complex:

$$\mathbb{R} \hookrightarrow \widetilde{\Lambda}^0(\Omega) \xmapsto{d} \widetilde{\Lambda}^1(\Omega) \xmapsto{d} \widetilde{\Lambda}^2(\Omega) \xmapsto{d} \widetilde{\Lambda}^3(\Omega) \longmapsto 0, \tag{3.12}$$

where $\widetilde{\Lambda}^p$ is the space of twisted p-forms.

Hodge duality

An ordinary p-form is the Hodge dual of a twisted $(n-p)$-form and in the same way a twisted p-form is the Hodge dual of an ordinary $(n-p)$-form. In general, the Hodge-$\star$ operator maps a p-form to an $(n-p)$-form:

$$\star: \quad \Lambda^p \mapsto \Lambda^{n-p}. \tag{3.13}$$

The $\star$ operator consequently connects the ordinary De Rham complex with the twisted one.

$$\begin{tikzcd}
\mathbb{R} \arrow[r, hook] & \Lambda^0 \arrow[r, mapsto, "d"] \arrow[d, leftrightarrow, "\star"] & \Lambda^1 \arrow[r, mapsto, "d"] \arrow[d, leftrightarrow, "\star"] & \Lambda^2 \arrow[r, mapsto, "d"] \arrow[d, leftrightarrow, "\star"] & \Lambda^3 \arrow[r, mapsto] \arrow[d, leftrightarrow, "\star"] & 0 \\
0 & \widetilde{\Lambda}^3 \arrow[l, mapsto] & \widetilde{\Lambda}^2 \arrow[l, mapsto, "d"'] & \widetilde{\Lambda}^1 \arrow[l, mapsto, "d"'] & \widetilde{\Lambda}^0 \arrow[l, mapsto, "d"'] & \mathbb{R} \arrow[l, hook']
\end{tikzcd}$$

The relation between Hodge duals is:

$$\alpha \wedge \star\beta = (\alpha,\beta)\omega_n = (\alpha,\beta)\star 1 \quad \forall \alpha,\beta \in \Lambda^k. \tag{3.14}$$

which defines the local inner product (α,β). Here the unit volume form is defined as $\omega_n := \star 1$. The definition of the local inner product leads to a global inner product defined as:

$$(\alpha,\beta)_\Omega = \int_\Omega (\alpha,\beta)\omega_n = \int_\Omega \alpha \wedge \star\beta. \tag{3.15}$$

Often, the twistedness of the volume form is being ignored by choosing an arbitrary orientation:

$$\omega_n := \sqrt{|\det(g_{ij})|}\, dx^1 \wedge ... \wedge dx^n. \tag{3.16}$$

With this definition, the orientation of the volume form is automatically imposed on the twisted Hodge duals. The effects become clear in the next two examples.

Example. In Euclidean three-space the Hodge duals of the basis forms are:

$$\begin{gathered}
\star 1 = dx\,dy\,dz \\
\star dx = dy\,dz, \quad \star dy = dz\,dx, \quad \star dz = dx\,dy, \\
\star dx\,dy = dz, \quad \star dy\,dz = dx, \quad \star dz\,dx = dy, \\
\star dx\,dy\,dz = 1.
\end{gathered}$$

Example. In Euclidean two-space the Hodge duals of the basis forms are:

$$\begin{gathered}
\star 1 = dx\,dy, \\
\star dx = dy, \quad \star dy = -dx, \\
\star dx\,dy = 1.
\end{gathered}$$

In $\mathbb{R}^2$, the peculiarity occurs of having $\star dy = -dx$, i.e. a positive flux is directed in the negative y-direction. Here, the Burke notation of twisted forms comes to good use, when defining the volume form actually as a twisted form $\omega_n = \{dx\,dy\}dx\,dy = \{dy\,dx\}dy\,dx = \widehat{1}$.

Example. In Euclidean two-space the Hodge duals of the basis forms are:

$$\star 1 = \{dx\,dy\}dx\,dy = \widehat{1},$$
$$\star dx = \{dx\,dy\}dy = \widehat{dx}, \quad \star dy = \{dy\,dx\}dx = \widehat{dy},$$
$$\star dx\,dy = \{dx\,dy\}1 = \widehat{dx}\,\widehat{dy}.$$

Using the exterior derative and the $\star$ operator, the Laplacian differential operator of vector calculus is generalized by the Laplace-de Rham operator in $\mathbb{R}^n$:

$$\Delta := \star\, d \star d + (-)^n d \star d\, \star\,. \tag{3.17}$$

3.1.2 Algebraic topology

Algebraic topology is the discrete counterpart of differential geometry, without involving any notion of metric, see [22, 24, 32]. The analogues of p-forms are p-cochains, which serve as the degrees of freedom of the numerical solution. Like p-forms are associated to p-dimensional manifolds, p-cochains are placed on p-dimensional grid elements. In three-space, the computational grid is comprised of vertices, edges, faces and cells. The main concern is the topology of the grid: what elements are connected to what elements and what orientation do they have. By structuring the connection and orientation the conservation laws can be formulated topologically, e.g. the fluxes through faces are summed to give the net flux through the total surface of the cell, which is equal to the volume integral of the divergence.

Chains and cochains

The manifold is represented by a cell complex, consisting of p-cells denoted by s_p^i. In three-space, the complex consists of 0-cells (vertices), 1-cells (edges), 2-cells (faces) and 3-cells (volumes). The cell complex can be constructed by simplices (triangles or tetrahedrons). Multidimensional splines are however defined by tensor products. Therefore the cell complex is built from p-cubes. A p-chain is linear combination of p-cells:

$$c_p = \sum_i w_i s_p^i, \tag{3.18}$$

where p-chain c_p is an element of a set C_p and w_i are the weights of the cells. The values of the weights are $[+1, -1, 0]$, indicating if a cell is part of a chain and how it is orientated. The dual basis of the chains follows from the relation:

$$s^i(s_j) = \delta^i_j. \tag{3.19}$$

Cochains are dual to chains, like covectors are dual to vectors, and defined as a linear combination of the dual basis:

$$a = \sum_i a_i s_i^p. \tag{3.20}$$

A set of cochains is denoted by C^p, so $a \in C^p$. Covectors are linear functionals on vectors, leading to a duality between p-forms and p-vectors. Similarly, p-cochains are linear functionals on p-chains:

$$a(c_p) = \langle a, c_p \rangle = \sum_i a_i w_i. \tag{3.21}$$

The cochains are the discrete representations of differential forms. When the numerical solution is composed of basis functions, the cochains form the basis coefficients. In other words, the cochains serve as the degrees of freedom of the algebraic system of equations. The direct relation between cochains and differential forms is discussed in detail in Section 3.2.

Boundary operator
The boundary operator ∂ returns the boundary of a p-chain, which is a $(p-1)$-chain:

$$\partial : \quad C_p \mapsto C_{p-1}. \tag{3.22}$$

The boundary operator satisfies $\partial\partial = 0$, in other words a boundary has no boundary by itself. The cell complex must have a sequence, such that:

$$0 \longmapsto C_3 \overset{\partial}{\longmapsto} C_2 \overset{\partial}{\longmapsto} C_1 \overset{\partial}{\longmapsto} C_0 \longmapsto 0. \tag{3.23}$$

Coboundary operator
The adjoint of the boundary operator ∂ is the coboundary operator $\delta : C^p \mapsto C^{(p+1)}$:

$$\langle c^p, \partial c_p \rangle = \langle \delta c^p, c_{p+1} \rangle, \tag{3.24}$$

where $\langle \cdot, \cdot \rangle$ denotes a duality pairing. The coboundary operator is the discrete counterpart of the exterior derivative, and similarily satisfies $\delta\delta = 0$. Also, the coboundary operator forms a discrete De Rham complex:

$$\mathbb{R} \hookrightarrow C^0 \overset{\delta}{\longmapsto} C^1 \overset{\delta}{\longmapsto} C^2 \overset{\delta}{\longmapsto} C^3 \longmapsto 0. \tag{3.25}$$

3.2 Basic operations

The basic operations are reconstruction and reduction of differential forms. Discrete differential forms are reconstructed from the cochains, which are the degrees of freedom in the discrete system. Conversely, the reduction operation maps forms to cochains.

3.2.1 Reconstruction

The reconstruction operator $\mathcal{I} : \bar{C}^p \mapsto \Lambda_h^p$ reconstructs differential forms using basis splines. The cochains are then the basis coefficients of the spline function, which is emphasised by the barred notation of $\bar{C}^p$ (see Subsection 3.2.2). The reconstructed p-form is called a discrete form, because it can be expressed in a finite set of DOFs. The space of the discrete forms is denoted by Λ_h^p. The general expansion of differential forms in basis functions is:

$$\mathcal{I}a = \sum_j a_j \Phi_j(\mathbf{x}) d\alpha, \tag{3.26}$$

where $a \in \bar{C}^p$ and Φ_j is the j-th basis function.

Example. In $\mathbb{R}^1$, 0-forms, or scalar functions, are expanded in B-splines:

$$\mathcal{I}a = \sum_j a_j B_j(x).$$

One-forms, or volume-forms, are expanded in M-splines:

$$\mathcal{I}b = \sum_j b_j M_j(x)dx.$$

The use of B-splines and M-splines preserves the duality of d and δ.

Lemma 4. *The reconstruction operator $\mathcal{I}$ has the commuting property: $d\mathcal{I} = \mathcal{I}\delta$.*

Proof.

$$d\mathcal{I}a = d\left(\sum_j a_j B_j(x)\right) = \sum_j (a_{j+1} - a_j)M_j(x)dx = \mathcal{I}\delta a$$

The mapping $\mathcal{I}$ is hence called a *conforming* mimetic reconstruction operator. □

By similar proof, the reconstruction operator extends to higher dimensions, while retaining the conforming property. The commutivity of $\mathcal{I}$ is illustrated in the following diagram.

$$\begin{tikzcd}
\bar{C}^p \arrow[r, "\delta"] \arrow[d, "\mathcal{I}"] & \bar{C}^{p+1} \arrow[d, "\mathcal{I}"] \\
\Lambda_h^p \arrow[r, "d"] & \Lambda_h^{p+1}
\end{tikzcd}$$

Example. In $\mathbb{R}^3$, the basis forms of 0-forms are:

$$\Phi_{i,j,k}(x, y, z) = B_i(x)B_j(y)B_k(z).$$

For 1-forms:

$$\begin{aligned}
\Phi^x_{i,j,k}(x, y, z)dx &= M_i(x)B_j(y)B_k(z)dx, \\
\Phi^y_{i,j,k}(x, y, z)dy &= B_i(x)M_j(y)B_k(z)dy, \\
\Phi^z_{i,j,k}(x, y, z)dz &= B_i(x)B_j(y)M_k(z)dz.
\end{aligned}$$

For 2-forms:

$$\begin{aligned}
\Phi^{xy}_{i,j,k}(x, y, z)dx\,dy &= M_i(x)M_j(y)B_k(z)dx\,dy, \\
\Phi^{yz}_{i,j,k}(x, y, z)dy\,dz &= B_i(x)M_j(y)M_k(z)dy\,dz, \\
\Phi^{zx}_{i,j,k}(x, y, z)dz\,dx &= M_i(x)B_j(y)M_k(z)dz\,dx.
\end{aligned}$$

And, for 3-forms:

$$\Phi_{i,j,k}(x, y, z)dx\,dy\,dz = M_i(x)M_j(y)M_k(z)dx\,dy\,dz.$$

3.2.2 Reduction

The common choice for the reduction operator is the De Rham map $\Lambda^k \mapsto C^k$:

$$\langle \mathcal{R}\alpha, c\rangle = \int_c \alpha \tag{3.27}$$

where $c \in C_p$ is a p-chain and $\alpha \in \Lambda^p$ is a p-form. In other words, the operator $\mathcal{R}$ is simply equivalent to integration of the p-form over the p-chain. In the case of 0-forms the integration operation reduces to evaluating the 0-form at points. An important feature of the De Rham map is the commuting property: $\mathcal{R}d = \delta\mathcal{R}$.

Lemma 5. *The De Rham map has the commuting property $\mathcal{R}d = \delta\mathcal{R}$.*

Proof. Bochev [9] gives the proof by combining Stokes' theorem with the duality of the boundary operator ∂ and the coboundary operator δ:

$$\langle \mathcal{R}d\alpha, c\rangle = \int_c d\alpha = \int_{\partial c} \alpha = \langle \mathcal{R}\alpha, \partial c\rangle = \langle \delta\mathcal{R}\alpha, c\rangle. \tag{3.28}$$

□

The commutative property is illustrated in the following diagram.

$$\begin{tikzcd}
\Lambda^p \arrow[r, "d"] \arrow[d, "\mathcal{R}"] & \Lambda^{p+1} \arrow[d, "\mathcal{R}"] \\
C^p \arrow[r, "\delta"] & C^{p+1}
\end{tikzcd}$$

Basis coefficients of B- or M-splines generally do not correspond to cochains according to the De Rham map: the coefficients do not represent function values or integrals. To deploy B- and M-splines as basis functions directly, an additional mapping from De Rham cochains to spline cochains is required. Spline cochains are denoted by $\bar{C}^p$ to distinguish them from De Rham cochains C^p. Hiemstra et al. [26] defined the linear map $\mathcal{F} : \bar{C}^p \mapsto C^p$ as:

$$\mathcal{F} := \mathcal{R} \circ \mathcal{I}, \tag{3.29}$$

using the definitions of Eqs. (3.26) and (3.27).

Lemma 6. *The map $\mathcal{F}$ commutes with respect to the coboundary operator δ, so it has the commutative property $\delta\mathcal{F} = \mathcal{F}\delta$.*

Proof. Combining the commuting properties $\mathcal{I}\delta = d\mathcal{I}$ and $\mathcal{R}d = \delta\mathcal{R}$ gives the relation:

$$\mathcal{R}\mathcal{I}\delta = \mathcal{R}d\mathcal{I} = \delta\mathcal{R}\mathcal{I},$$

which is equal to:

$$\mathcal{F}\delta = \delta\mathcal{F}.$$

□

The commutative property is illustrated in the following diagram.

$$\begin{tikzcd}
\bar{C}^p \arrow[r, "d"] \arrow[d, "\mathcal{F}"] & \bar{C}^{p+1} \arrow[d, "\mathcal{F}"] \\
C^p \arrow[r, "\delta"] & C^{p+1}
\end{tikzcd}$$

The constructed bridge between De Rham cochains and spline cochains, leads to the definition of an alternative reduction operator $\bar{\mathcal{R}} : \Lambda^p \mapsto \bar{C}^p$ is defined to be as:

$$\bar{\mathcal{R}}\alpha := \mathcal{F}^{-1}\mathcal{R}\alpha.$$

Here, the invertibility of $\mathcal{F}$ is an *essential* requirement.

Lemma 7. *Similar to $\mathcal{R}$, the spline reduction operator $\bar{\mathcal{R}}$ has the commuting property: $\bar{\mathcal{R}}d = \delta\bar{\mathcal{R}}$.*

Proof. Combining Stokes' theorem and the duality of δ and ∂ gives:

$$\langle \mathcal{F}\bar{\mathcal{R}}d\alpha, c\rangle = \int_c d\alpha = \int_{\partial c} \alpha = \langle \mathcal{F}\bar{\mathcal{R}}\alpha, \partial c\rangle = \langle \delta\mathcal{F}\bar{\mathcal{R}}\alpha, c\rangle.$$

Subsequently, substituting $\delta\mathcal{F} = \mathcal{F}\delta$ gives:

$$\langle \mathcal{F}\bar{\mathcal{R}}d\alpha, c\rangle = \langle \mathcal{F}\delta\bar{\mathcal{R}}\alpha, c\rangle.$$

□

The reduction operator $\bar{\mathcal{R}}$ is paramount to the use of splines, because it pairs to the reconstruction operator $\mathcal{I}$ in terms of consistency. The operator $\mathcal{I}$ has the consistency property, so that $\mathcal{I}$ is the right inverse of $\bar{\mathcal{R}}$:

$$\bar{\mathcal{R}}\mathcal{I} = \mathcal{F}^{-1}\mathcal{R}\mathcal{I} = \mathcal{F}^{-1}\mathcal{F} = id.$$

Additionally, the operator $\mathcal{I}$ has the approximation property:

$$\mathcal{I}\bar{\mathcal{R}} = id + \mathcal{O}(h^s),$$

as in general the projected p-form is not an element of the particular spline space. The projection of a form on the space of discrete forms is defined as:

$$\pi := \mathcal{I} \circ \mathcal{F}^{-1} \circ \mathcal{R}. \tag{3.30}$$

The commutative diagrams can be assembled to form the big picture. Note that this diagram can be established in similar fashion for basis functions other than basis splines, as long as the reconstruction $\mathcal{I}$ is conforming and the mapping $\mathcal{F}$ is invertible.

$$\begin{tikzcd}
\Lambda^p \arrow[r, "d"] \arrow[d, "\mathcal{R}"] \arrow[ddd, bend right=60, "\pi"'] & \Lambda^{p+1} \arrow[d, "\mathcal{R}"] \arrow[ddd, bend left=60, "\pi"] \\
C^p \arrow[r, "\delta"] \arrow[d, "\mathcal{F}^{-1}"] & C^{p+1} \arrow[d, "\mathcal{F}^{-1}"] \\
\bar{C}^p \arrow[r, "\delta"] \arrow[d, "\mathcal{I}"] & \bar{C}^{p+1} \arrow[d, "\mathcal{I}"] \\
\Lambda_h^p \arrow[r, "d"] & \Lambda_h^{p+1}
\end{tikzcd}$$

Algebraic realisations The linear map $\mathcal{F} = \mathcal{R} \circ \mathcal{I}$ can be realised as an invertible matrix. For 0-forms $\mathcal{F}$ is:

$$\mathcal{F}a = \mathcal{R}\mathcal{I}a = \sum_j \sum_j a_j B_j(x_j) = B\mathbf{a} \tag{3.31}$$

where B is the collocation matrix of a B-spline. The operator $\bar{\mathcal{R}}$ is equivalent to interpolating data points $\mathbf{x}$ with B-splines:

$$\bar{\mathcal{R}}\alpha = \mathcal{F}^{-1}\mathcal{R}\alpha = B^{-1}\alpha(\mathbf{x}) \tag{3.32}$$

For 1-forms $\mathcal{F}$ is:

$$\mathcal{F}a = \bar{\mathcal{R}}\mathcal{I}a = \int_{L_j} \sum_j \sum_j a_j M_j(x)dx = M\mathbf{a} \tag{3.33}$$

where $M_{ij} = \int_{L_i} M_j dx$. The operator $\hat{\mathcal{R}}$ is equivalent to an integral-preserving interpolation (i.e. histopolation):

$$\bar{\mathcal{R}}\alpha = \mathcal{F}^{-1}\mathcal{R}\alpha = [M^{-1}]\left\{\int_{L_j} \alpha\right\} \tag{3.34}$$

3.3 Natural operations

Unlike the exterior derivative, the inner and exterior product on cochains do not have exact discrete representations. Their actions on cochains are defined naturally with the basic operations: reduction and reconstruction.

3.3.1 Inner product

The inner product of two differential forms is induced by the metric, see Eq. (3.14). Similary, the local inner product of two cochains is defined as follows:

$$(a, b) := (\mathcal{I}a, \mathcal{I}b), \quad \forall a, b \in C^p. \tag{3.35}$$

Subsequently, the inner product in a global sense, as in Eq. (3.15), follows from the local definition:

$$(a, b)_\Omega := (\mathcal{I}a, \mathcal{I}b)_\Omega, \quad \forall a, b \in C^p. \tag{3.36}$$

3.3.2 Exterior product

The exterior product or wedge product on cochains, $\wedge : C^p \wedge C^q \mapsto C^{p+q}$, follows naturally from the definition of the wedge product for differential forms:

$$a \wedge b = \mathcal{R}(\mathcal{I}a \wedge \mathcal{I}b) \quad \forall a \in C^p, \forall b \in C^q. \tag{3.37}$$

3.4 Discrete operators

In the introduction the Laplace-De Rham operator is expressed in terms of the exterior derivative and the Hodge $\star$ operator. To solve Laplace equations numerically, the Laplace-De Rham operator is discretized using the discrete exterior derivative and the discrete $\star$ operator, which act on cochains.

3.4.1 Exterior derivative

The exterior derivative acts on differential forms, whereas the discrete counterpart of the exterior derivative acts on the cochains. Their connection lies in the generalized Stokes' theorem:

$$\int_{\Omega} d\alpha = \int_{\partial\Omega} \alpha. \tag{3.38}$$

Thus, the exterior derivative d is the adjoint of the boundary operator ∂. Also, the coboundary operator δ is the adjoint of ∂. The coboundary operator δ acts as the discrete exterior derivative. This property becomes apparent in Eq. (3.28). The discrete derivative is realized as a matrix containing the values $+1, -1, 0$, called the indicidence matrix D. The sign indicates the orientations of $(p-1)$-chains with respect to the boundary of p-chains.

3.4.2 Hodge $\star$ operator

Unlike the exterior derivative, the Hodge $\star$ operator has no exact discrete counterpart. A discretization technique is necessary for obtaining a discrete Hodge ($\star_h$) operator, mapping inner-oriented quantities to outer-oriented quantities, and vice versa. Bochev & Hyman [9] distinguish two general approaches for constructing a discrete $\star_h$ operator. Depending on the choice of approach, the discrete $\star_h$ does not inherit some of the properties of the analytic $\star$ operator. In general, the $\star_h$ operator includes a form of approximation and connects the discrete De Rham complexes of inner- and outer-oriented cochains.

$$\begin{tikzcd}
\mathbb{R} \arrow[r, hook] & C^0 \arrow[r, mapsto, "\delta"] \arrow[d, leftrightarrow, "\star_h"] & C^1 \arrow[r, mapsto, "\delta"] \arrow[d, leftrightarrow, "\star_h"] & C^2 \arrow[r, mapsto, "\delta"] \arrow[d, leftrightarrow, "\star_h"] & C^3 \arrow[r, mapsto] \arrow[d, leftrightarrow, "\star_h"] & 0 \\
0 & \widetilde{C}^3 \arrow[l, mapsto] & \widetilde{C}^2 \arrow[l, mapsto, "\delta"'] & \widetilde{C}^1 \arrow[l, mapsto, "\delta"'] & \widetilde{C}^0 \arrow[l, mapsto, "\delta"'] & \mathbb{R} \arrow[l, hook']
\end{tikzcd}$$

Natural $\star_h$ operator. The *natural* $\star_h$ operator follows from the reduction $\mathcal{R}$ and reconstruction $\mathcal{I}$ operators. The definition of the natural $\star_h$ operator is, mapping cochains C^p to cochains C^{n-p} in $\mathcal{R}^n$:

$$\star_h := \mathcal{R} \star \mathcal{I}, \tag{3.39}$$

which is proposed by Tarhasaari et al. [37] as well. If basis splines are applied as basis functions for $\mathcal{I}$, then the appropriate reduction operator $\bar{\mathcal{R}}$ (see Subsection 3.2.2) and cochains must be used. The natural $\star_h$ operator is compatible with reduction and reconstruction by definition, but not with the inner product and wedge product.

> **Example.** In $\mathbb{R}^1$, a twisted 0-cochain is the Hodge dual to a 1-cochain: $\widetilde{b} = \star_h a$, where $\widetilde{b} \in \widetilde{C}^0, a \in C^1$. The vertices of the dual grid denoted by $\widetilde{\tau}$. The reconstructions are:
>
> $$\mathcal{I}a = \sum_j a_j M_j(x) dx,$$
> $$\mathcal{I}b = \sum_j b_j \widetilde{B}_j(x).$$
>
> The $\star_h$ is then realized as:
>
> $$\mathbf{b} = \star_h a = \mathcal{R} \star \mathcal{I} a = B^{-1} M \mathbf{a},$$

> where $B = B_j(\widetilde{\tau}_i)$ and $M = M_j(\widetilde{\tau}_i)$. The discrete Hodge operator can be written as the linear system: $B\widetilde{b} = Ma$, without inverting the collocation matrix B. The $\star_h$ operator can also be defined as a mapping from twisted 0-cochains to 1-cochains and involves a histopolation problem instead of an interpolation problem. Alternatively, this $\star_h : \widetilde{C}^0 \mapsto C^1$ operator can be approximated by the inverse of the demonstrated $\star_h : C^1 \mapsto \widetilde{C}^0$.

Derived $\star_h$ operator. The *derived* discrete $\star_h$ operator does not follow directly from the reduction and reconstruction operators, but from the inner product and wedge product. The definition of the Hodge dual is:

$$\int_\Omega \alpha \wedge \star\beta = (\alpha, \beta)_\Omega, \tag{3.40}$$

where $\alpha, \beta \in \Lambda^k$. The derived $\star_h$ operator is hence defined as:

$$\int_\Omega a \wedge \star_h b = (a, b)_\Omega, \tag{3.41}$$

where $a, b \in C^k$. The derived $\star_h$ operator is however incompatible with the reduction and reconstruction, as opposed to the natural $\star_h$ operator.

Dual grid

The $\star_h$ operator directly relates inner- and outer-oriented cochains, representing the Hodge duality of ordinary and twisted differential forms. This explicit relation necessitates the implementation of a dual grid to facilitate the Hodge duals of primal quantities. The adjectives primal and dual here do not imply a specific orientation of the grid. The inner-oriented grid could be declared as either the primal or the dual grid for instance. The vertices of the grids are the interpolation points. The Greville sites from Eq. (2.6) are the preferred interpolation points, because of they yield accurate solutions, are simple to calculate and have correspondence to control points. There are two general possibilities for constructing a dual basis with basis splines.

Staggered knots. Back & Sonnendrücker [5] introduced a mimetic method with periodic splines. The dual grid is obtained by staggering the knots: they are positioned halfway between the knots of the primal grid. The use of periodic splines limits the method to problems with periodic boundary conditions. This approach can however be extended to non-periodic splines, i.e. on closed knot sequences. An example with quadratic splines is shown in Fig. 3.1. At boundaries of the dual grid two additional points must be added to close the spline. Boundary conditions must be imposed at these points to prevent additional degrees of freedom. Thus, every line element (the intervals between interpolation points) is associated to one interpolation point of the dual grid, excluding the boundary points. The one-to-one mapping is required for transferring the invertibility property of the analytic $\star$ operator to its discrete counterpart.

Increased order. A different dual grid is proposed by Hiemstra [25], where the locations of the breakpoints remain unchanged. The knot multiplicity at the ends is increased, and so is the order of the B-splines. Due to the difference in order, the interpolation points of the dual grid are nevertheless staggered with respect to the primal grid, establishing a one-to-one relation between the grids. An example is shown in Fig. 3.2 with a cubic primal grid and a quadratic dual grid. Again, the boundary conditions must be imposed on the boundary points added to the dual grid.

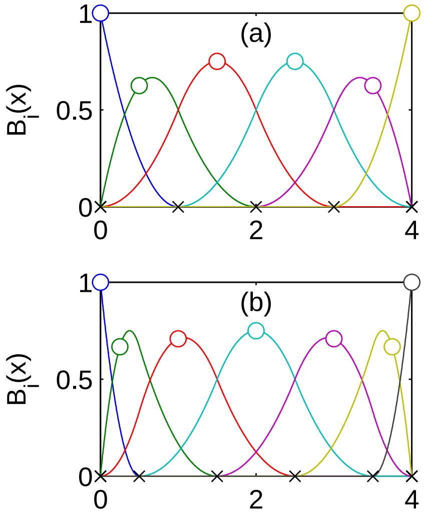


Figure 3.1: The primal grid (a) is defined by the uniform knot vector $\mathbf{t} = [0, 0, 0, 1, 2, 3, 4, 4, 4]$, and the dual grid (b) by $\mathbf{t} = [0, 0, 0, \frac{1}{2}, 1\frac{1}{2}, 2\frac{1}{2}, 3\frac{1}{2}, 4, 4, 4]$. The positioning of the knots is marked by ($\times$) and the interpolation points by ($\circ$), which are the Greville sites.

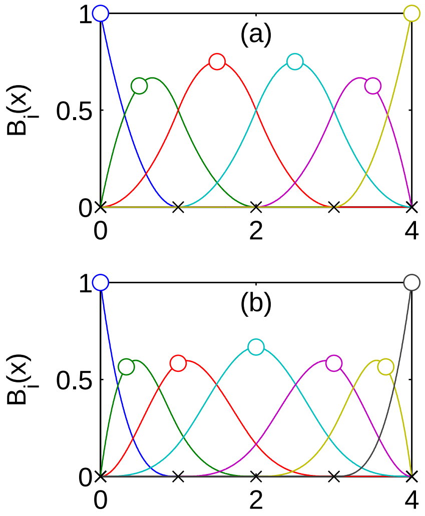


Figure 3.2: The primal grid (a) is defined by the uniform knot vector $\mathbf{t} = [0, 0, 0, 1, 2, 3, 4, 4, 4]$, and the dual grid (b) by $\mathbf{t} = [0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4]$. The positioning of the knots is marked by ($\times$) and the interpolation points by ($\circ$), which are the Greville sites.

Higher dimensions. Spline functions in higher dimensions are defined by tensor products of one-dimensional splines. Two- and three-dimensional grids are thus created by the tensor products of one-dimensional knot vectors and interpolation points. The vertices of the grid cells are the interpolation points, and subsequently define the line, surface and volume elements, presuming quadrilateral elements. An example of a two-dimensional primal and dual grid is shown in Fig. 3.3. The formal dual grid, with the right number of DOFs, is not a closed cell complex, which cannot be implemented using closed spline functions. Therefore a boundary is added to the dual grid, as shown in the one-dimensional dual grids. The boundary conditions must then be imposed on the dual grid, so the number of actual DOFs remains unchanged. Dirichlet boundary conditions are realized on the boundary vertices of the dual grid, and Neumann conditions on the line elements of the boundary.

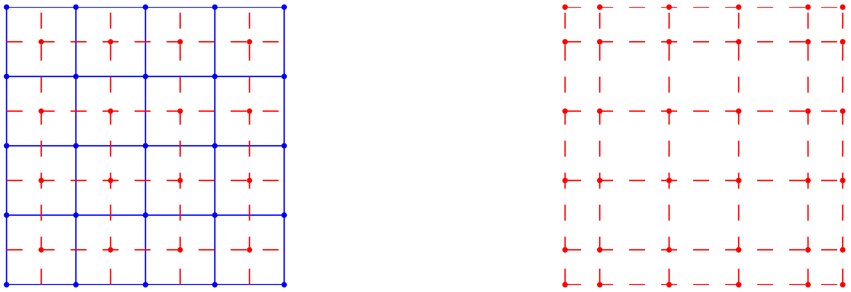

(a) Primal (blue) and dual (red) grid.
(b) Closed boundary of the dual grid.

Figure 3.3: Two-dimensional primal and dual grids defined by the tensor-product of interpolation points. Note that the vertices in the far corners of the dual grid are inevitable due to the tensor-product definition.

3.4.3 Laplace-de Rham operator

The Laplace-de Rham operator acting on cochains, $\Delta_h : C^p \mapsto C^p$, arises from the coboundary operator and the discrete Hodge $\star$ operator:

$$\Delta_h = \star_h \, \delta \star_h \delta + (-1)^n \delta \star_h \delta \star_h \tag{3.42}$$

> **Example.** The coboundary operator is represented by the incidence matrix D and the discrete Hodge $\star$ operator by the matrix H. The Laplacian on the 0-form α is:
>
> $$\Delta\alpha = \star\, d \star d\alpha = \beta,$$
>
> where β is a nonhomogeneous right-hand side. Discretizing the differential equation gives:
>
> $$H_2 D_2 H_1 D_1 \mathbf{a} = \mathbf{b},$$
>
> where a and b are 0-cochains such that $b = \mathcal{R}\beta$ and the numerical solution $\mathcal{I}a = \alpha_h$. The matrix $K = H_2 D_2 H_1 D_1$ is named the *stiffness matrix* in the terminology of finite-element analysis. Then, the linear system of equations becomes:
>
> $$K\mathbf{a} = \mathbf{b}.$$
>
> **Example.** Alternatively, the equation in the previous example could be formulated differently by bringing the $\star$ operator to the right-hand side:
>
> $$\Delta\alpha = d \star d\alpha = \star\beta.$$
>
> Discretizing the differential equation gives:
>
> $$D_2 H_1 D_1 \mathbf{a} = \mathbf{b},$$
>
> where a is the 0-cochain approximating the analytic solution, $\mathcal{I}a = \alpha_h$, and b is the n-cochain such that $b = \mathcal{R}(\star\beta)$. The second $\star$ operator is thus not discretized, but treated analytically.

Chapter 4

Numerical experiments

In Chapter 2 splines are introduced as basis functions for numerical analysis. In Chapter 3 a framework for mimetic discretizations with splines is presented. Within this framework, the discrete Hodge ($\star$) operator plays a key role in the discretization of the Laplace operator. The discrete Hodge operator requires the definition of a dual grid, as there is an explicit relation between Hodge duals. In the first section the projection error associated to the discrete Hodge operator is investigated. Then to assess the accuracy of the discretization method, Poisson's equation is solved in one dimension in Section 4.2 and subsequently in two dimensions in Section 4.3. The goal is to assess the convergence of the numerical solution to the analytic solution with respect to global mesh refinement.

4.1 Projection error

In Chapter 3 the projection operator π is introduced. In this section the projection error is analysed, before advancing to numerical solutions of (partial) differential equations. The projection of a differential form on the space of discrete differential forms, spanned by the basis splines, is:

$$\pi = \mathcal{I} \circ \mathcal{R}, \tag{4.1}$$

where $\mathcal{R}$ is not the De Rham map, but the reduction operator mapping forms to spline coefficients, omitting the bar notation $\bar{\mathcal{R}}$. The projection on the dual grid is denoted by $\widetilde{\pi}$, where the tilde denotes operation defined on the dual grid. The projection operator contains a consistency error by not reconstructing the differential form exactly:

$$\mathcal{I}\mathcal{R} = id + O(h^k), \tag{4.2}$$

where h is a measurement of grid spacing and k is the polynomial order. The approximation error of the projection operator approaches zero asymptotically when the grid h is refined: eventually the error norm becomes zero for the hypothetical value $h = 0$. The order of convergence, of the projection, with respect to refinement of h is the polynomial order k. The natural $\star_h$ operator contains an approximation error as well, which is related to the projection operator. The natural discrete $\star$ operator is defined as:

$$\star_h = \widetilde{\mathcal{R}} \star \mathcal{I}. \tag{4.3}$$

The $\star_h$ operator converges asymptotically to the analytic $\star$ operator when the grid is refined. The $\star_h$ operator is related to a projection followed by a second projection on the dual grid in the following way:

$$\widetilde{\pi} \star \pi = \widetilde{\mathcal{I}}\widetilde{\mathcal{R}} \star \mathcal{I}\mathcal{R} = \widetilde{\mathcal{I}}(\widetilde{\mathcal{R}} \star \mathcal{I})\mathcal{R} = \widetilde{\mathcal{I}} \star_h \mathcal{R}. \tag{4.4}$$

The relation between projection operator and the $\star_h$ operator gives an estimate of the order of grid convergence for the $\star_h$ operator. The approximation error of the double projection of a differential form $\alpha \in \Lambda^p$ is therefore decomposed into two errors:

$$\epsilon_{tot} = \star\alpha - \widetilde{\pi} \star \pi\alpha = \underbrace{(\star\alpha - \star\pi\alpha)}_{\epsilon_1} + \underbrace{(\star\pi\alpha - \widetilde{\pi} \star \pi\alpha)}_{\epsilon_2} = \epsilon_1 + \epsilon_2. \tag{4.5}$$

The convergence properties of the two errors depend on the order of the polynomial space on which the form is projected.

Example. Let $\alpha \in \Lambda^1$ in $\mathbb{R}$ be projected on the space of M-splines of order $k-1$, whereas $\beta \in \Lambda^0$ is projected on the space of B-splines of order k. The first error is approximately:

$$\epsilon_1 = \star\alpha - \star\pi\alpha = \star\alpha - \star(id + O(h^{k-1}))\alpha = O(h^{k-1}).$$

The second error is estimated to be:

$$\epsilon_2 = \star\pi\alpha - \widetilde{\pi} \star \pi\alpha = \star(id + O(h^{k-1}))\alpha - (id + O(h^k)) \star (id + O(h^{k-1}))\alpha = O(h^{k-1}).$$

Thus, the total error ϵ_{tot} is expected to be in the order of $O(h^{k-1})$.

Grid convergence study

To verify the error estimates given in the previous example, a one-dimensional test case is analysed. Consider the 1-form $\alpha \in \Lambda^1$:

$$\alpha = \sin(2\pi x)\, dx, \quad \forall x \in [0,1]. \tag{4.6}$$

The spaces of the spline functions are defined by the polynomial order k and the knot sequence $\mathbf{t}$. Zero-forms are projected on B-splines of order k, whereas one-forms are projected on M-splines of order $k-1$. The knot sequence of the primal grid is defined as the open and uniform sequence:

$$\mathbf{t} = [\underbrace{t_0, ..., t_0}_{k}, t_1, ..., t_{n-1}, \underbrace{t_n, ..., t_n}_{k}]. \tag{4.7}$$

Here, h is the uniform spacing between the breakpoints. The knot sequence of the dual grid results from the staggering the breakpoints of the primal grid:

$$\widetilde{\mathbf{t}} = [\underbrace{t_0, ..., t_0}_{k}, \frac{t_1 + t_2}{2}, ..., \frac{t_{n-2} + t_{n-1}}{2}, \underbrace{t_n, ..., t_n}_{k}]. \tag{4.8}$$

The M-splines are consequently defined on the reduced knot sequences $\mathbf{t}'$ and $\widetilde{\mathbf{t}}'$. The points of interpolation for the primal and dual grid are the Greville sites, defined by Eq. (2.6). Results of the errors, measured in the L^2-norm, versus the grid size h are plotted in Figs. 4.1, 4.2 and 4.3. The data points of the plots are listed in Tables 4.1, 4.2 and 4.3. The average slope of the logarithmic curves are shown in Tables 4.4, 4.5 and 4.6.

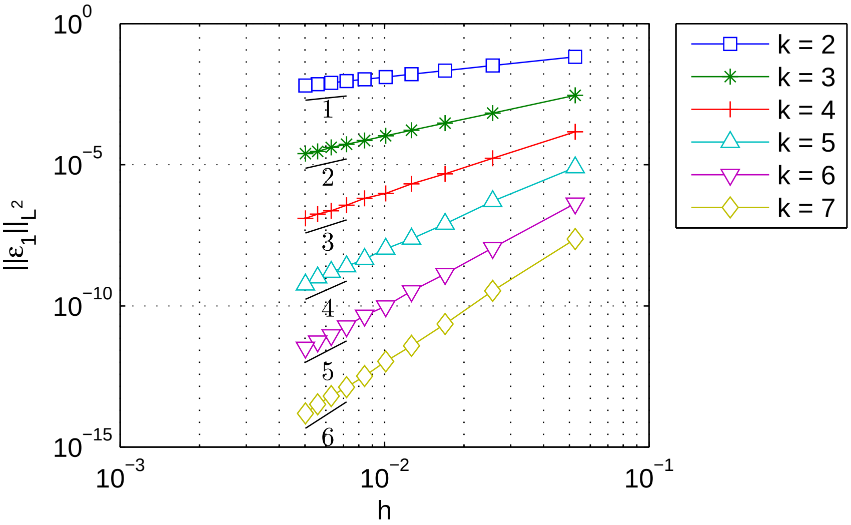


Figure 4.1: Error norm $\|\epsilon_1\|_{L^2}$ versus uniform knot distance h.

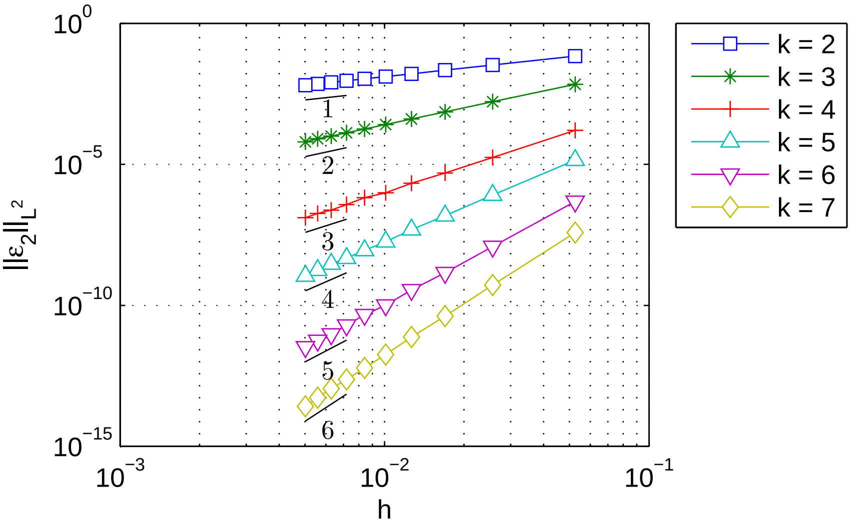


Figure 4.2: Error norm $\|\epsilon_2\|_{L^2}$ versus uniform knot distance h.

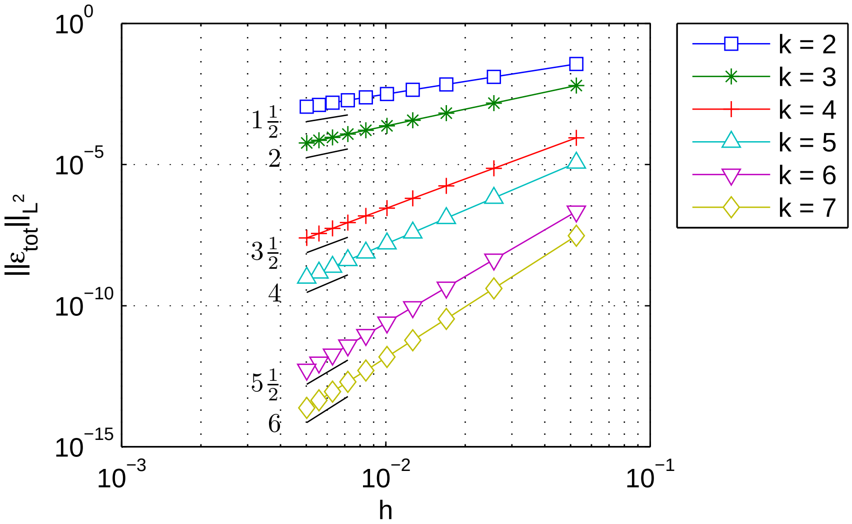


Figure 4.3: Error norm $\|\epsilon_{tot}\|_{L^2}$ versus uniform knot distance h.

Table 4.1: Logarithmic data points of uniform knot distance h and error norm $\|\epsilon_1\|_{L^2}$.

	$\log(\|\epsilon_1\|_{L^2})$					
$\log(h)$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
-2.9444	-2.6974	-5.8372	-8.8209	-11.7296	-14.6665	-17.5652
-3.6636	-3.4151	-7.2846	-10.9868	-14.4800	-18.2979	-21.8010
-4.0775	-3.8289	-8.1143	-12.2496	-16.3403	-20.4174	-24.4966
-4.3694	-4.1207	-8.6993	-13.0725	-17.5578	-21.8111	-26.2831
-4.5951	-4.3463	-9.1455	-13.8474	-18.3700	-23.0514	-27.5626
-4.7791	-4.5303	-9.5246	-14.2513	-19.1978	-23.8224	-28.7545
-4.9345	-4.6856	-9.8365	-14.8032	-19.8040	-24.6818	-29.6616
-5.0689	-4.8201	-10.0979	-15.2613	-20.2510	-25.4039	-30.3841
-5.1874	-4.9386	-10.4230	-15.5303	-20.7033	-25.9207	-31.0566
-5.2933	-5.0445	-10.5926	-15.8716	-21.2908	-26.4309	-31.8108

Table 4.2: Logarithmic data points of uniform knot distance h and error norm $\|\epsilon_2\|_{L^2}$.

	$\log(\|\epsilon_2\|_{L^2})$					
$\log(h)$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
-2.9444	-2.6648	-4.9624	-8.7347	-11.1689	-14.5381	-17.0742
-3.6636	-3.3931	-6.3926	-10.9358	-14.0069	-18.2240	-21.3566
-4.0775	-3.8133	-7.2199	-12.2126	-15.7033	-20.3589	-23.9030
-4.3694	-4.1088	-7.8036	-13.0471	-16.8374	-21.7713	-25.6203
-4.5951	-4.3367	-8.2548	-13.8256	-17.8199	-23.0184	-27.0531
-4.7791	-4.5222	-8.6228	-14.2356	-18.5343	-23.8010	-28.1416
-4.9345	-4.6782	-8.9335	-14.7859	-19.1651	-24.6538	-29.0826
-5.0689	-4.8140	-9.1945	-15.2492	-19.6423	-25.3813	-29.8212
-5.1874	-4.9331	-9.4245	-15.5139	-20.1534	-25.8954	-30.5688
-5.2933	-5.0395	-9.6625	-15.8648	-20.6275	-26.4288	-31.2725

Table 4.3: Logarithmic data points of uniform knot distance h and error norm $\|\epsilon_{tot}\|_{L^2}$.

	$\log(\|\epsilon_{tot}\|_{L^2})$					
$\log(h)$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
-2.9444	-3.2934	-5.0602	-9.3352	-11.3660	-15.3382	-17.3106
-3.6636	-4.3473	-6.4851	-11.8058	-14.2300	-19.2631	-21.6155
-4.0775	-4.9663	-7.3116	-13.2481	-15.8843	-21.5433	-24.0995
-4.3694	-5.4044	-7.8947	-14.2673	-17.0517	-23.1510	-25.8515
-4.5951	-5.7434	-8.3466	-15.0564	-17.9553	-24.3961	-27.2069
-4.7791	-6.0198	-8.7141	-15.6998	-18.6909	-25.4092	-28.3105
-4.9345	-6.2533	-9.0243	-16.2434	-19.3115	-26.2654	-29.2391
-5.0689	-6.4553	-9.2934	-16.7135	-19.8496	-27.0044	-30.0488
-5.1874	-6.6332	-9.5307	-17.1278	-20.3241	-27.6593	-30.7486
-5.2933	-6.7924	-9.7426	-17.4991	-20.7478	-28.2415	-31.3658

Table 4.4: Logarithmic slope (m) for $\|\epsilon_1\|_{L^2}$ averaged over data intervals and the mean absolute deviation (MAD).

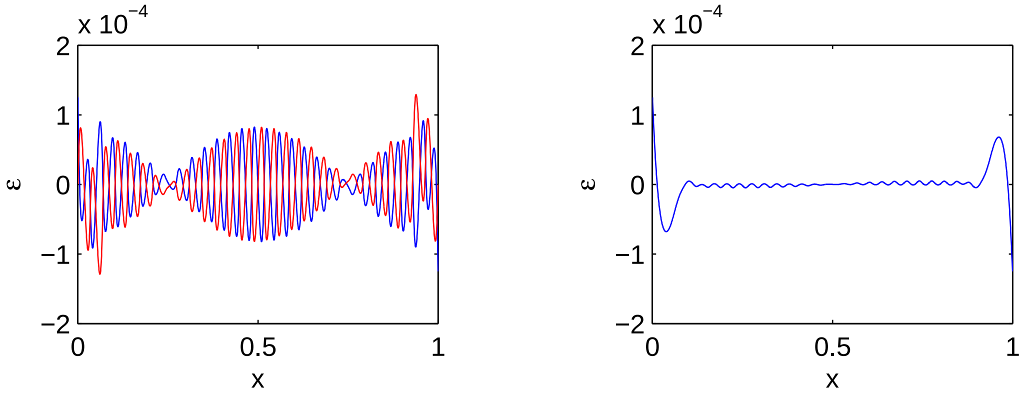
	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
m	0.9996	2.0395	2.9959	4.1310	4.9680	6.0755
MAD	0.0004	0.1609	0.3786	0.4859	0.3843	0.4285

Table 4.5: Logarithmic slope (m) for $\|\epsilon_2\|_{L^2}$ averaged over data intervals and the mean absolute deviation (MAD).

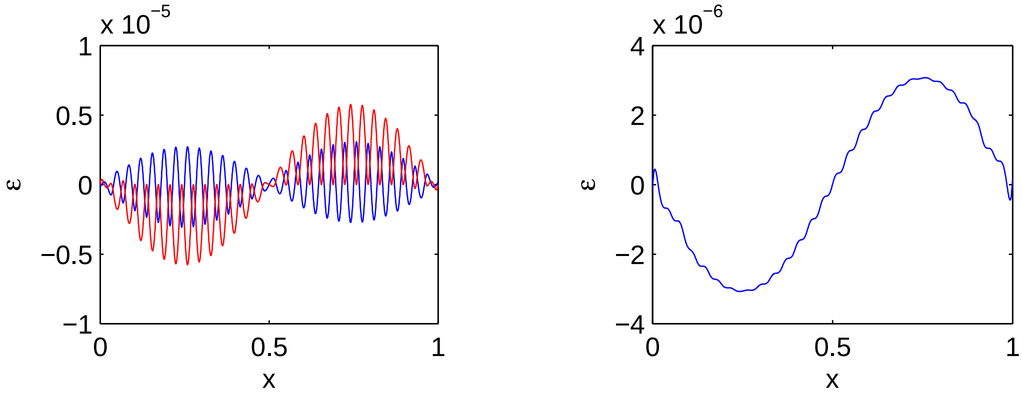
	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
m	1.0092	2.0129	3.0241	4.0628	5.0197	6.0844
MAD	0.0030	0.0521	0.3892	0.2199	0.3617	0.2480

Table 4.6: Logarithmic slope (m) for $\|\epsilon_{tot}\|_{L^2}$ averaged over data intervals and the mean absolute deviation (MAD).

	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
m	1.4974	1.9974	3.4892	3.9980	5.5032	5.9696
MAD	0.0075	0.0039	0.0131	0.0047	0.0128	0.0457



(a) ϵ_1 (blue) and ϵ_2 (red), $k = 4$ (b) ϵ_{tot}, $k = 4$



(c) ϵ_1 (blue) and ϵ_2 (red), $k = 5$ (d) ϵ_{tot}, $k = 5$

Figure 4.4: Local errors in the double projection $\widetilde{\pi} \star \pi\alpha$ for $k = 4$ and $k = 5$, with grid spacing $h = 0.0345$.

The orders of convergence for ϵ_1 and ϵ_2 are approximately $k - 1$, as expected. The total error ϵ_{tot} for the odd orders (3,5,7) has order $k - 1$ as well. For the even orders (2,4,6) however, the order is approximately $k - \frac{1}{2}$, which is higher than initially expected. The cause of this convergence behaviour is exemplified in Fig. 4.4 for an even order $k = 4$ and an odd order $k = 5$. In the case of even orders the second error counterbalances the first one, which evens the total error beneficially. In the case of odd orders the two errors are not balanced. This phenomenon explains why higher-than-expected orders of convergence are obtained with even orders, although the exact mechanism behind this feature remains unexplained. In Appendix A, an additional example of an asymmetric function is included, which exhibits the same phenomenon. The convergence behaviour is therefore expected for smooth functions in general, and not only for even and odd functions, such as sines and cosines.

4.2 1D Poisson's equation

In this section the Poisson's equation in one-dimensional space is solved numerically the mimetic discretization method presented in Chapter 3. Poisson's equation is a second-order differential equation:

$$\nabla \cdot \nabla \phi = \Delta \phi = f. \tag{4.9}$$

For $f = 0$, the equation becomes Laplace's equation. In one-dimensional space, Poisson's equation is written in terms of the coordinate x:

$$\frac{d^2}{dx^2}\phi(x) = f(x), \quad \forall x \in \Omega. \tag{4.10}$$

The accompanying boundary conditions are Dirichlet boundary conditions:

$$\phi(x) = 0, \quad \forall x \in \partial\Omega. \tag{4.11}$$

The problem is solved on the interval:

$$\Omega = [0, 1]. \tag{4.12}$$

Poisson's equation written in terms of differential forms is:

$$\star d \star d\phi = f. \tag{4.13}$$

Introducting the additional variables u, q and v, Poisson's equation could be written as a combination of two metric relations and two topological relations (the conservation laws):

$$\begin{cases} u = d\phi, \quad \text{(gradient theorem,)} & \text{(4.14a)} \\ q = \star u, & \text{(4.14b)} \\ v = dq, \quad \text{(divergence theorem,)} & \text{(4.14c)} \\ f = \star v. & \text{(4.14d)} \end{cases}$$

The gradient theorem is discretised on the inner-oriented primal grid and the divergence theorem on the outer-oriented dual grid. The discrete Hodge operators serve as the bridges between the variables on the primal and the dual grid.

4.2.1 Dual grid

The discrete Hodge $\star$ operator is constructed by its natural definition in Eq. (3.39). The explicit formulation of the discrete $\star$ operator requires a dual grid to allocate the Hodge duals. Two options for the dual grid, named A and B, are described in Subsection 3.4.2. Poisson's equation is solved with both grids to compare their results.

Grid A. The dual grid is constructed by staggering the knots, while maintaining the same polynomial order. An example is shown in Fig. 4.7.

Grid B. The dual grid is constructed by reducing the order, while maintaining the positions of the knots. An example is shown in Fig. 4.8.

Vertices and edges. The vertices, or 0-cells, of the grid are the interpolation points of the spline functions. The Greville sites, Eq. (2.6), are used as the interpolation points. The vertices of the primal and dual grid are denoted by τ_j and $\tilde{\tau}_j$ respectively. The vertices form the boundaries of the edges, or 1-cells. Following the work of Toshniwal [38], the positive orientation of the 0-cells is defined to a sink and the positive orientation of the 1-cells is defined to be from left to right. The positive orientations of the primal grid are illustrated in Fig. 4.5. For the dual grid, the 0-cells are defined from left to right and the positive orientation of the 1-cells is a source (indicative of a net outflux). The positive orientations of the dual grid are illustrated in Fig. 4.6.

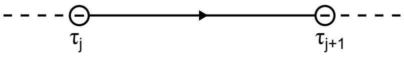


Figure 4.5: Positive orientations of the vertices and edges

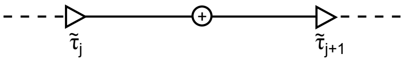


Figure 4.6: Positive orientations of the vertices and edges

The 0- and 1-cells are associated to 0- and 1-cochains, which are the DOFs of the discrete differential forms. The inner-oriented 0-cochains are related to point values and 1-cochains to line integrals. Their exact relation is given by the Stokes' theorem in the form of the gradient theorem:

$$\phi(\tau_{j+1}) - \phi(\tau_j) = \int_{\tau_j}^{\tau_{j+1}} d\phi = \int_{\tau_j}^{\tau_{j+1}} u,$$

and the divergence theorem:

$$q(\tilde{\tau}_{j+1}) - q(\tilde{\tau}_j) = \int_{\tilde{\tau}_j}^{\tilde{\tau}_{j+1}} dq = \int_{\tilde{\tau}_j}^{\tilde{\tau}_{j+1}} v.$$

The line integrals are obtained by differencing function values at points and therefore they are path-independent; Stokes' theorem is a topological equation which does not involve the metric of the space. In one-dimensional space the gradient and divergence theorem seem much alike. The interpretation of the cochains however does depend on the grid orientation. The primal grid houses the inner-oriented 0-cochains $\bar{\phi}$ (sinks) and 1-cochains $\bar{u}$ (line integrals). The dual grid houses the outer-oriented 0-cochains $\bar{q}$ (fluxes) and 1-cochains $\bar{v}$ (net outfluxes). The discrete representation of Stokes' theorem is the coboundary operator, for example the discrete gradient operator:

$$\delta\bar{\phi} = \begin{bmatrix} -1 & +1 & 0 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 \\ 0 & 0 & -1 & +1 & 0 \\ 0 & 0 & 0 & -1 & +1 \end{bmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{pmatrix},$$

where the coboundary operator returns four line integrals for a grid consisting of five vertices. The incidence matrix is denoted by D and represents the coboundary operator.

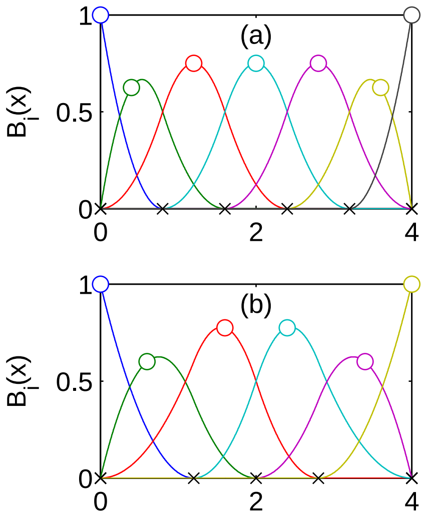


Figure 4.7: The primal grid (a) is defined by the uniform knot vector $\mathbf{t} = [0,0,0,\frac{4}{5},1\frac{3}{5},2\frac{2}{5},3\frac{1}{5},4,4,4]$, and the dual grid (b) by $\widetilde{\mathbf{t}} = [0,0,0,1\frac{1}{5},2,2\frac{4}{5},4,4,4]$. The knots are indicated by ($\times$) and the nodes by ($\circ$).

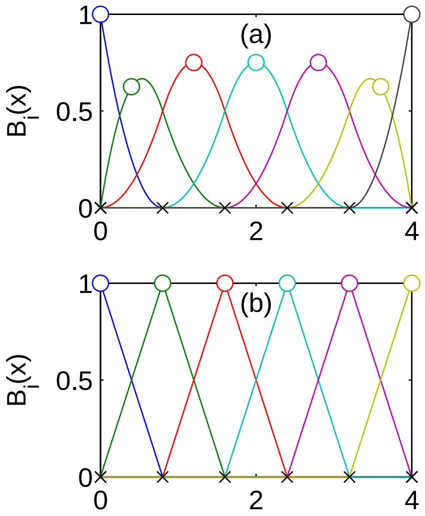


Figure 4.8: The primal grid (a) is defined by the uniform knot vector $\mathbf{t} = [0,0,0,\frac{4}{5},1\frac{3}{5},2\frac{2}{5},3\frac{1}{5},4,4,4]$, and the dual grid (b) by $\widetilde{\mathbf{t}} = [0,0,\frac{4}{5},1\frac{3}{5},2\frac{2}{5},3\frac{1}{5},4,4]$. The knots are indicated by ($\times$) and the nodes by ($\circ$).

4.2.2 Basis functions

The knot sequences of the primal and dual grid define the basis functions of the primal and dual quantities. The discrete ordinary differential forms are defined on the primal grid:

$$\phi_h = \sum_{j=1}^{m} \phi_j B_{j,k}(x), \tag{4.15}$$

$$u_h = d\phi_h = \sum_{j=1}^{m-1} (\phi_{j+1} - \phi_j) M_{j,k-1}(x) dx, \tag{4.16}$$

where the spline functions $B_{j,k}(x)$ are defined on the knot sequence $\mathbf{t}$ and $M_{j,k-1}(x)$ on the reduced knot sequence $\mathbf{t}'$. Similarly, the discrete twisted forms are defined on the dual grid A:

$$q_h = \sum_{j=1}^{m-1} q_j \widetilde{B}_{j,k}(x), \tag{4.17}$$

$$v_h = dq_h = \sum_{j=1}^{m-2} (q_{j+1} - q_j) \widetilde{M}_{j,k-1}(x) dx, \tag{4.18}$$

where the spline functions $\widetilde{B}_{j,k}(x)$ are defined on the knot sequence $\widetilde{\mathbf{t}}$ and $\widetilde{M}_{j,k-1}(x)$ on the reduced knot sequence $\widetilde{\mathbf{t}}'$. On dual grid B, the basis splines are of one order lower:

$$q_h = \sum_{j=1}^{m-1} q_j \widetilde{B}_{j,k-1}(x), \tag{4.19}$$

$$v_h = dq_h = \sum_{j=1}^{m-2} (q_{j+1} - q_j) \widetilde{M}_{j,k-2}(x) dx. \tag{4.20}$$

Note that for grid B, two knot sequences are identical, $\mathbf{t}' = \widetilde{\mathbf{t}}$, and that M-splines are B-splines scaled according to Eq. (2.16). The $\star_h$ operator, mapping ordinary 1-forms to twisted 0-forms, simply scales the splines appropriately and contains no approximation error by itself. For grid A, the two knot sequences are defined differently: $\mathbf{t}' \neq \widetilde{\mathbf{t}}$ and as a consequence the $\star_h$ operator contains a form of approximation. For both grids, the $\star_h$ operator between ordinary 0-forms and twisted 1-forms is an approximation, because in that case: $\mathbf{t} \neq \widetilde{\mathbf{t}}'$.

4.2.3 Discretization

The system of equations (4.14) is discretized with the discrete counterpart of the Hodge $\star$ operator and with the coboundary operator:

$$\begin{cases} \bar{u} = \delta\bar{\phi}, \quad \text{(discrete gradient theorem,)} & \text{(4.21a)} \\ \bar{q} = \star_h \bar{u}, & \text{(4.21b)} \\ \bar{v} = \delta\bar{q}, \quad \text{(discrete divergence theorem,)} & \text{(4.21c)} \\ \bar{f} = \star_h \bar{v}. & \text{(4.21d)} \end{cases}$$

The right-hand side function f is approximated with a spline interpolation:

$$\bar{f} = \bar{\mathcal{R}}(f) = B^{-1}\mathcal{R}(f). \tag{4.22}$$

The reduction operator $\bar{\mathcal{R}}$ maps the function f to the B-spline coefficients $\bar{f}$, i.e. the coefficients that solve the interpolation problem. The compact notation of the discrete Poisson equation is:

$$\star_h \delta \star_h \delta \bar{\phi} = \bar{\mathcal{R}}(f). \tag{4.23}$$

Let the coboundary operator be represented by the incidence matrix D and the $\star_h$ operator by the matrix H, then discrete equations give the linear system of equations:

$$\left[H^{(0,\tilde{1})} D^{(\tilde{1},\tilde{0})} H^{(\tilde{0},1)} D^{(1,0)} \right] \left\{ \bar{\phi} \right\} = \left\{ \bar{\mathcal{R}}(f) \right\}. \tag{4.24}$$

The superscripts of the matrices specify the cochains on which the operator acts, e.g. $H^{(\tilde{0},1)}$ acts on 1-cochains and maps to twisted 0-cochains. The natural $\star_h$ operator, given by Eq. (3.39), consists of a reduction and a reconstruction matrix:

$$H^{(\tilde{0},1)} = \widetilde{B}^{-1} M, \tag{4.25}$$

$$H^{(0,\tilde{1})} = B^{-1} \widetilde{M}. \tag{4.26}$$

The entries of the reconstruction matrices are defined as follows:

$$M_{ij} = M_j(\widetilde{\tau}_i), \tag{4.27}$$

$$\widetilde{M}_{ij} = \widetilde{M}_j(\tau_i), \tag{4.28}$$

and the entries of the reduction matrices as:

$$\widetilde{B}_{ij} = \widetilde{B}_j(\widetilde{\tau}_i), \tag{4.29}$$

$$B_{ij} = B_j(\tau_i). \tag{4.30}$$

The Dirichlet boundary conditions are imposed strongly by inserting $\phi_1 = 0$ and $\phi_m = 0$. The spline functions are defined on an open knot sequence where the first and last basis coefficients equal the boundary values. Non-zero boundary conditions $\phi(\tau_1)$ and $\phi(\tau_m)$ could then be imposed by the straightforward insertion of $\phi_1 = \phi(\tau_1)$ and $\phi_m = \phi(\tau_m)$.

4.2.4 Error analysis

The numerical method is validated by means of a manufactured solution. The analytic solution to Poisson's equation is chosen to be:

$$\phi(x) = e^x \sin(2\pi x). \tag{4.31}$$

Taking the derivative of $\phi(x)$ twice with respect to x yields the required right-hand side of Poisson's equation:

$$f(x) = \frac{d^2\phi}{dx^2} = e^x \left(\left(1 - 4\pi^2\right) \sin\left(2\pi x\right) + 4\pi \cos\left(2\pi x\right) \right). \tag{4.32}$$

The local error of the numerical solution $\phi_h(x)$ is:

$$\epsilon(x) = \phi(x) - \phi_h(x). \tag{4.33}$$

The global error is measured in the L^2-norm of the local error:

$$\|\epsilon\|_{L^2} = \left(\int_\Omega |\epsilon(x)|^2 \, dx \right)^{1/2} . \tag{4.34}$$

The numerical solution converges to the analytic solution when the uniform knot distance $h = t_{j+1} - t_j$ is refined. The discretization error is presumed to decrease algebraically when refining the mesh:

$$\|\epsilon\|_{L^2} = O(h^m). \tag{4.35}$$

The order of convergence m can be derived numerically from the slopes of the logarithmic curves, plotting the error $\|\epsilon\|_{L^2}$ against the mesh size h.

Grid A. In Fig. 4.9 the error is plotted against h. The data points of the logarithmic curves are listed in Table 4.7. The average slopes and mean absolute deviation from the average slope are given in Table 4.9. For the even orders $k = 2, 4, 6$, the orders of convergence are optimal. For the odd orders $k = 3, 5, 7$ however, the orders of convergence are perceived to be suboptimal $k - 1$.

Grid B. In Fig. 4.10 the error is plotted against h. The data points of the logarithmic curves are listed in Table 4.8. The average slopes and mean absolute deviation from the average slope are given in Table 4.10. Note that data for $k = 2$ is missing, because the minimum required order is $k = 3$ for second-order problems. The requirement is due to the fact that the order of the M-splines on the dual grid is $k - 2$. For the even orders $k = 4, 6$, the orders of convergence are suboptimal $k - 2$. For the odd orders $k = 3, 5, 7$ as well, the orders of convergence are seen to be suboptimal $k - 1$.

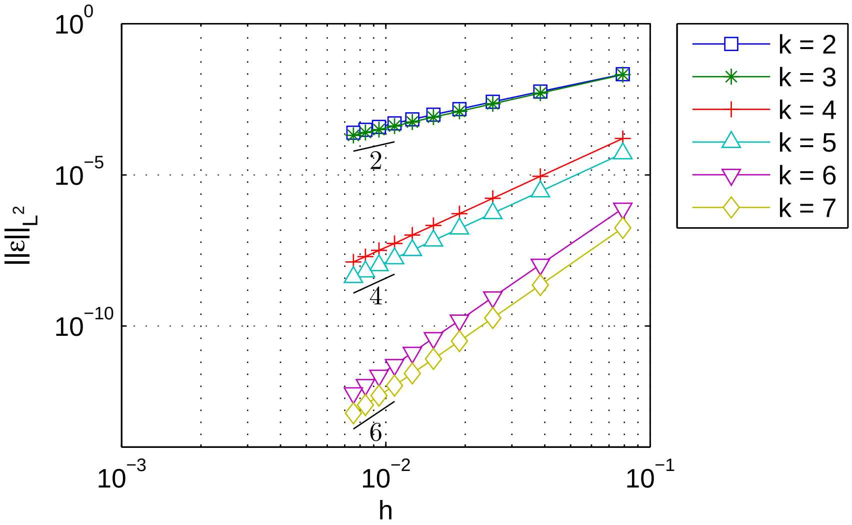


Figure 4.9: Error $\|\epsilon\|_{L^2}$ versus uniform knot distance h, obtained with grid A.

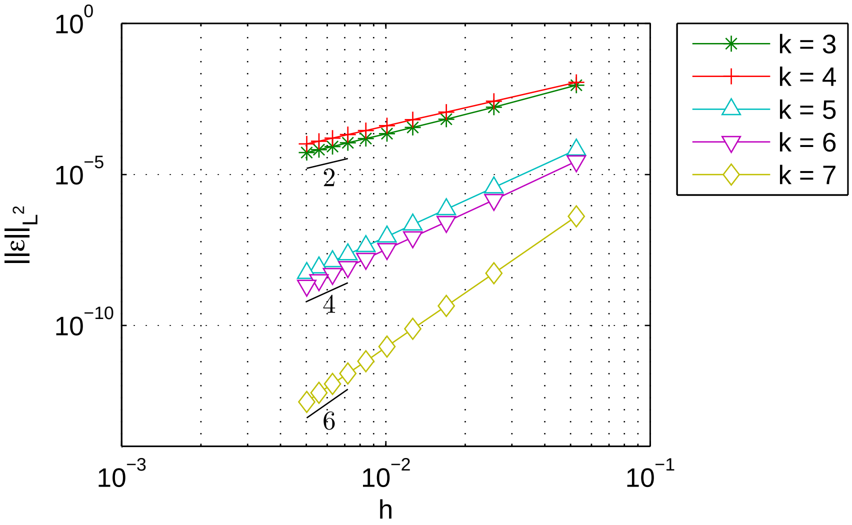


Figure 4.10: Error $\|\epsilon\|_{L^2}$ versus uniform knot distance h, obtained with grid B.

Table 4.7: Logarithmic data points of uniform knot distance h and error norm $\|\epsilon\|_{L^2}$, obtained with grid A.

	$\log(\|\epsilon\|_{L^2})$					
$\log(h)$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
-2.9444	-3.2934	-5.0602	-9.3352	-11.3660	-15.3382	-17.3106
-3.6636	-4.3473	-6.4851	-11.8058	-14.2300	-19.2631	-21.6155
-4.0775	-4.9663	-7.3116	-13.2481	-15.8843	-21.5433	-24.0995
-4.3694	-5.4044	-7.8947	-14.2673	-17.0517	-23.1510	-25.8515
-4.5951	-5.7434	-8.3466	-15.0564	-17.9553	-24.3961	-27.2069
-4.7791	-6.0198	-8.7141	-15.6998	-18.6909	-25.4092	-28.3105
-4.9345	-6.2533	-9.0243	-16.2434	-19.3115	-26.2654	-29.2391
-5.0689	-6.4553	-9.2934	-16.7135	-19.8496	-27.0044	-30.0488
-5.1874	-6.6332	-9.5307	-17.1278	-20.3241	-27.6593	-30.7486
-5.2933	-6.7924	-9.7426	-17.4991	-20.7478	-28.2415	-31.3658

Table 4.8: Logarithmic data points of uniform knot distance h and error norm $\|\epsilon\|_{L^2}$, obtained with grid B.

	$\log(\|\epsilon\|_{L^2})$				
$\log(h)$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
-2.9444	-4.7058	-4.4884	-9.6150	-10.5046	-14.7001
-3.6636	-6.3818	-5.9250	-12.5067	-13.4549	-19.0352
-4.0775	-7.2941	-6.7526	-14.1658	-15.1238	-21.5271
-4.3694	-7.9194	-7.3364	-15.3349	-16.2960	-23.2828
-4.5951	-8.3949	-7.7877	-16.2377	-17.2007	-24.6382
-4.7791	-8.7785	-8.1557	-16.9741	-17.9378	-25.7454
-4.9345	-9.0992	-8.4664	-17.5959	-18.5599	-26.6811
-5.0689	-9.3765	-8.7352	-18.1336	-19.0980	-27.4727
-5.1874	-9.6200	-8.9722	-18.6077	-19.5722	-28.1599
-5.2933	-9.8366	-9.1840	-19.0314	-19.9961	-28.8602

Table 4.9: Logarithmic slopes (m) averaged over data intervals and the mean absolute deviation (MAD), obtained with grid A.

	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
m	1.9241	1.9769	3.9898	4.0074	5.9958	5.9792
MAD	0.0340	0.0121	0.0460	0.0078	0.0293	0.0949

Table 4.10: Logarithmic slopes (m) averaged over data intervals and the mean absolute deviation (MAD), obtained with grid B.

	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
m	2.1217	1.9996	4.0045	4.0196	6.0455
MAD	0.0692	0.0005	0.0046	0.0211	0.1259

4.2.5 Conclusions

A comparison of the L^2-errors in Tables 4.7 and 4.8 indicates that grid A yields more accurate solutions than grid B for the analysed orders with the only exception of $k = 3$. The results given Tables 4.9 and 4.10 point out that the convergence orders of grid A are greater than or equal to the orders of grid B. The estimated orders of convergence with respect to h-refinement suggest the conclusion that:

	m	
k	Grid A	Grid B
2	k	-
3	$k-1$	$k-1$
4	k	$k-2$
5	$k-1$	$k-1$
6	k	$k-2$
7	$k-1$	$k-1$

The mimetic discretization method introduced in this work bears much resemblance with spline collocation methods, such as described by Russell & Shampine [34]. The principle of collocation methods is to interpolate the derivatives with splines, e.g. $\phi''_h(\tau) = f(\tau)$ in Poisson's equation, from which the solution ϕ_h is derived from ϕ''_h. The errors are estimated to be $O(h^{k-s})$, where the s^{th} derivative interpolates the right-hand side function. For grid A, the right-hand side function is interpolated by M-splines of order $k-1$, so the error is $O(h^{k-1})$. For grid b, the right-hand side function is interpolated by M-splines of order $k-2$, so the error is $O(h^{k-2})$.

The results of the numerical computations agree partially with the estimates given by Russell & Shampine. For grid A, the orders of convergence are found to be $k-1$ for $k = 3, 5, 7$, but k for $k = 2, 4, 6$. For grid B, the orders of convergence are found to be $k-2$ for $k = 4, 6$, but k for $k = 3, 5, 7$. The difference between odd and even order splines is demonstrated already in Section 4.1. The higher-than-expected orders of convergence are witnessed as well in collocation methods by Auricchio et al. [4], Beirão da Veiga [7] and Botella [10, 11]. The results found in literature lack adequate explanation however. The results presented here suggest that interpolating the righhand side function of Poisson's equation with odd order splines lead to superconvergent solutions. The probable explanation lies in a beneficial superposition of numerical errors, as seen in Fig. 4.4.

4.3 2D Poisson's equation

In the previous section Poisson's equation is solved in one dimension. Grid convergence tests have demonstrated that staggering knots of the dual grid (grid A) is superior to reducing the order of the dual grid (grid B). Therefore the approach of staggering knots of the dual grid with respect to the primal grid is applied to a two-dimensional problem. The orders of convergence with respect to mesh refinement are examined again.

4.3.1 Problem formulation

Poisson's equation is solved on the unit square domain $\Omega = [0,1] \times [0,1]$ in Euclidean two-space. Poisson's equation in Cartesian coordinates including Dirichlet boundary conditions is:

$$\begin{cases} \star d \star d\phi(x,y) = f(x,y), & \forall(x,y) \in \Omega, \qquad (4.36a) \\ \phi(x,y) = 0, & \forall(x,y) \in \partial\Omega. \qquad (4.36b) \end{cases}$$

By introducing the variables u, q and v, Poisson's equation can be separated into two topological relations and two metric relations:

$$\begin{cases} u = d\phi, & \text{(gradient theorem,)} \qquad (4.37a) \\ q = \star u, & \qquad (4.37b) \\ v = dq, & \text{(divergence theorem,)} \qquad (4.37c) \\ f = \star v. & \qquad (4.37d) \end{cases}$$

The numerical solution ϕ_h is compared to a manufactured analytic solution in order to assess the accuracy of the numerical solution. The manufactured analytic solution is:

$$\phi(x,y) = e^{x+y}\sin(2\pi x)\sin(2\pi y). \tag{4.38}$$

The right-hand side function corresponding to the manufactured solution is:

$$\begin{aligned} f(x,y) &= \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} \\ &= (2 - 8\pi^2)e^{x+y}\sin(2\pi x)\sin(2\pi y) + 4\pi e^{x+y}\sin\left(2\pi(x+y)\right). \end{aligned} \tag{4.39}$$

The right-hand side function $f(x,y)$ is illustrated in Fig. 4.11. The function is nonperiodic and has two local minima and two local maxima located in its domain.

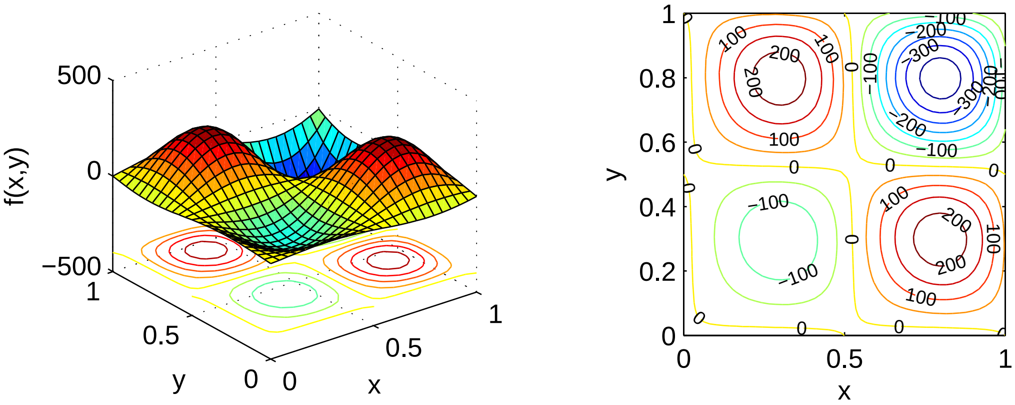
4.3.2 Discrete differential forms

Two-dimensional spline functions are defined as tensor products of one-dimensional spline functions. The space of one-dimensional spline functions is determined by the knot sequence $\mathbf{t}$ and polynomial order k. The B-splines for ordinary differential forms are defined on the uniform knot sequence:

$$\mathbf{t} = [\underbrace{t_0, ..., t_0}_{k}, t_1, ..., t_{n-1}, \underbrace{t_n, ..., t_n}_{k}]. \tag{4.40}$$

The uniform spacing between the breakpoints is denoted by h and is indicative for the mesh size. The B-splines for twisted forms, the Hodge duals of ordinary forms, are defined on a staggered knot sequence:

$$\widetilde{\mathbf{t}} = [\underbrace{t_0, ..., t_0}_{k}, \frac{t_1 + t_2}{2}, ..., \frac{t_{n-2} + t_{n-1}}{2}, \underbrace{t_n, ..., t_n}_{k}]. \tag{4.41}$$



(a) Surface plot with contour lines (b) Contour plot

Figure 4.11: Right-hand side function $f(x, y)$ of Poisson's equation in $\mathbb{R}^2$.

This dual grid corresponds to "grid A" in the one-dimensional test case, in which it produces the most accurate results. The breakpoints of $\widetilde{\mathbf{t}}$ are staggered with respect to $\mathbf{t}$ in order to preserve the one-to-one relation between Hodge duals. Naturally, the M-splines of order $k-1$ are defined on the reduced knot sequences $\mathbf{t}'$ and $\widetilde{\mathbf{t}}'$. The two-dimensional splines are the tensor products of the one-dimensional spline functions of x and y. The knot sequences in x and y direction are not necessarily identical and can be varied independently from one another.

When the univariate basis splines are defined, the basis forms of the ordinary and the twisted differential forms are obtained by the tensor products. For brevity the basis splines are written in a compact format: $B_i(x) = B_{i,k,t}(x)$, $\widetilde{B}_i(x) = B_{i,k,\widetilde{t}}(x)$, $M_i(x) = M_{i,k-1,t'}$ and $\widetilde{M}_i(x) = M_{i,k-1,\widetilde{t}'}$.

Ordinary forms The discrete 0-forms have m^2 basis forms:

$$\phi_h = \sum_{i=1}^{m} \sum_{j=1}^{m} \phi_j B_i(x) B_j(y), \tag{4.42}$$

whereas the 1-forms have $2m^2 - 2m$ basis forms:

$$u_h = \sum_{i=1}^{m-1} \sum_{j=1}^{m} u^x_{ij} M_i(x) B_j(y) dx + \sum_{j=1}^{m} \sum_{j=1}^{m-1} u^y_{ij} B_i(x) M_j(y) dy. \tag{4.43}$$

Twisted forms Twisted forms are notated in Burke's notation described in Subsection 3.1.1. The Hodge duals of 0-forms, which are twisted 2-forms, have $m^2 - 4m + 4$ basis forms:

$$v_h = \sum_{i=1}^{m-2} \sum_{j=1}^{m-2} v_{ij} \widetilde{M}_i(x) \widetilde{M}_j(y) \widehat{1}, \tag{4.44}$$

whereas twisted 1-forms have $2m^2 - 6m + 4$ basis forms:

$$q_h = \sum_{i=1}^{m-1} \sum_{j=1}^{m-2} q^x_{ij} \widetilde{B}_i(x) \widetilde{M}_j(y) \widehat{dx} + \sum_{j=1}^{m-2} \sum_{j=1}^{m-1} q^y_{ij} \widetilde{M}_i(x) \widetilde{B}_j(y) \widehat{dy}. \tag{4.45}$$

To achieve a one-to-one relation between Hodge duals, the number of DOFs must be equal in both bases. Initially, the basis of the ordinary forms involves more basis coefficients than the basis of twisted forms. The surplus of basis coefficients is remedied by the imposition of boundary conditions on the ordinary forms, which in this case are Dirichlet boundary conditions. The number of DOFs of the 0-forms then reduce to $m^2-4(m-1) = m^2-4+4$ and of the 1-forms to $(2m^2-2m)-4(m-1) = 2m^2-6m+4$. Now the number of DOFs agrees with the number of DOFs of the Hodge duals.

4.3.3 Computational grid

The vertices, or 0-cells, of the computational grid are the interpolation points of the basis splines. For univariate spline functions the interpolation points are the Greville sites τ, calculated for the specified knot sequence $\mathbf{t}$. The interpolation points of the two-variate splines can then be defined as the Kronecker product of the univariate interpolation points, which gives the vertices:

$$c_{(0)}^{i,j} = (\tau_i, \tau_j), \quad i, j \in \{1, \dots, m\}. \tag{4.46}$$

The vertices form the boundaries of the faces, the 1-cells of the cell complex, e.g.:

$$c_{(1)}^{i,j} = (c_{(0)}^{i,j}, c_{(0)}^{i+1,j}). \tag{4.47}$$

The faces again are the boundaries of the volumes, the 2-cells, resulting in a quadrilateral grid:

$$c_{(2)}^{i,j} = (c_{(0)}^{i,j}, c_{(0)}^{i+1,j}, c_{(0)}^{i+1,j+1}, c_{(0)}^{i,j+1}). \tag{4.48}$$

The dual grid is defined in an identical way in which the Greville sites $\widetilde{\tau}$ are calculated for $\widetilde{t}$. The construction of an inner- and outer-oriented cell complices gives rise to incidence matrices that represent the coboundary operators, i.e. the discrete formulations of the gradient theorem and the divergence theorem, see Subsection 3.1.2. On the primal grid the gradient operator is discretized by the coboundary operator $\delta : C^0 \mapsto C^1$:

$$\delta = D^{(1,0)}. \tag{4.49}$$

On the dual grid the divergence operator is discretized by the coboundary operator $\delta : \widetilde{C}^1 \mapsto \widetilde{C}^2$:

$$\delta = \widetilde{D}^{(2,1)}. \tag{4.50}$$

Poisson's equation is discretized by the coboundary operator δ and the discrete Hodge operator $\star_h$:

$$\star_h \, \delta \star_h \, \delta\phi = \bar{\mathcal{R}}(f). \tag{4.51}$$

An example of a two-dimensional primal and dual grid is given in Fig. 4.12. The positive orientations are chosen to be from left to right, and from down to up. To complete the discretization of the Laplace-operator, the Hodge $\star$ operator has to be discretized as well. The discrete $\star_h$ operator relates the inner-oriented 1-cochains (line integrals) to the outer-oriented 1-cochains (fluxes), and the inner-oriented 0-cochains (point values) to the outer-oriented 2-cochains (volume integrals). As in the one-dimensional test case, the $\star_h$ is construced according to its natural definition:

$$\star_h = \bar{\mathcal{R}} \star \mathcal{I}. \tag{4.52}$$

The $\star_h$ operator is realized as a pair of matrices M and W, where M is the reduction and W is the reconstruction of the differential form. For example, M is collocation matrix for the 0-form, then W evaluates the dual 2-form at the interpolation points. In general, the Hodge duality is discretized as:

$$\star_h a = b \rightarrow Ma = Wb, \tag{4.53}$$

where $a \in C^p$ and $b \in C^{(n-p)}$. The $\star_h$ operator can be represented by the matrix:

$$H^{(p,n-p)} = M^{-1}W. \tag{4.54}$$

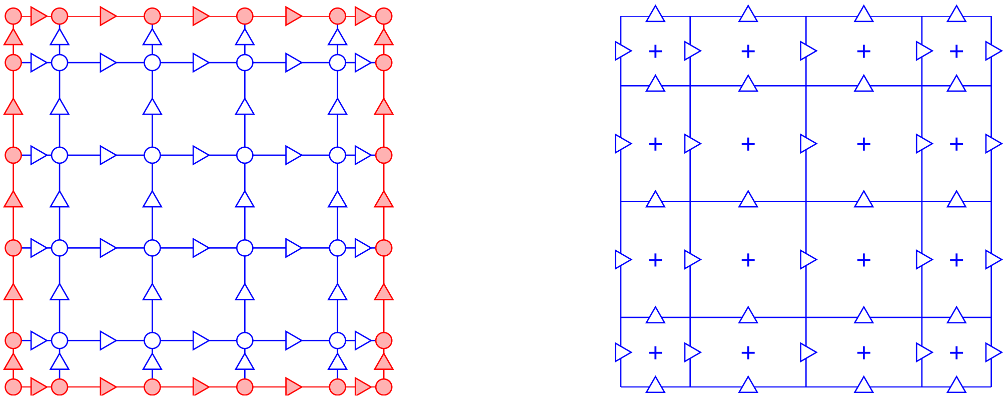
The discretization of Poisson's equation, given in (4.36), is given in terms of the coboundary operator δ and the discrete Hodge operator $\star_h$:

$$\star_h \, \delta \star_h \, \delta\bar{\phi} = \bar{\mathcal{R}}(f), \tag{4.55}$$

where the right-hand side function is approximated by a bivariate spline interpolation: $\bar{\mathcal{R}}(f)$ returns the B-spline coefficients that solve the interpolation problem. In matrix representations, the discrete system becomes:

$$\left[H^{(0,2)}\widetilde{D}^{(2,1)}\widetilde{H}^{(1,1)}D^{(1,0)}\right]\{\bar{\phi}\} = \{\bar{\mathcal{R}}(f)\}, \tag{4.56}$$

which can be solved when the boundary conditions are imposed on $\bar{\phi}$. Dirichlet boundary conditions are imposed strongly by inserting values at the boundary nodes. Alternatively, Neumann boundary conditions could be imposed by inserting flux values $\bar{v}$ at the boundary edges. Then, the problem should be posed in a mixed formulation, such that $\bar{v}$ becomes an additional variable in the linear system of equations.



(a) Inner-oriented primal grid with point values ϕ_{ij} (∘) and line integrals u_{ij} (▷). Boundary conditions are marked red.

(b) Outer-oriented dual grid with fluxes q_{ij} (▷) and net outfluxes v_{ij} (+).

Figure 4.12: Primal and dual grid on which Poisson's equation is solved, indicating the positive orientation of quantities. The gradient theorem is discretized on the primal grid and the divergence theorem on the dual grid.

4.3.4 Error analysis

The numerical solution is compared to the manufactured solution to assess the accuracy of the discretization method. The local error in the numerical solution $\phi_h(x, y)$ is defined as:

$$\epsilon(x,y) = \phi(x,y) - \phi_h(x,y). \tag{4.57}$$

The global error is measured in the L^2-norm:

$$\|\epsilon\|_{L^2} = \left(\int_\Omega |\epsilon(x,y)|^2 \, d\Omega \right)^{1/2}. \tag{4.58}$$

Grid convergence tests are performed for polynomial orders in the range from 2 to 7. The results for the global error are plotted in Fig. 4.13. The logarithmic data is listed in Table 4.11. From the slopes of the logarithmic curves the orders of convergence are derived. The average slope over each data interval is given in Table 4.12. For the polynomial orders $k = 3, 5, 7$ convergence orders of $k - 1$ are determined. For the polynomial orders $k = 2, 4, 6$ higher orders of approximately k are attained. For $k = 4$ the convergence order is even slightly higher than k.

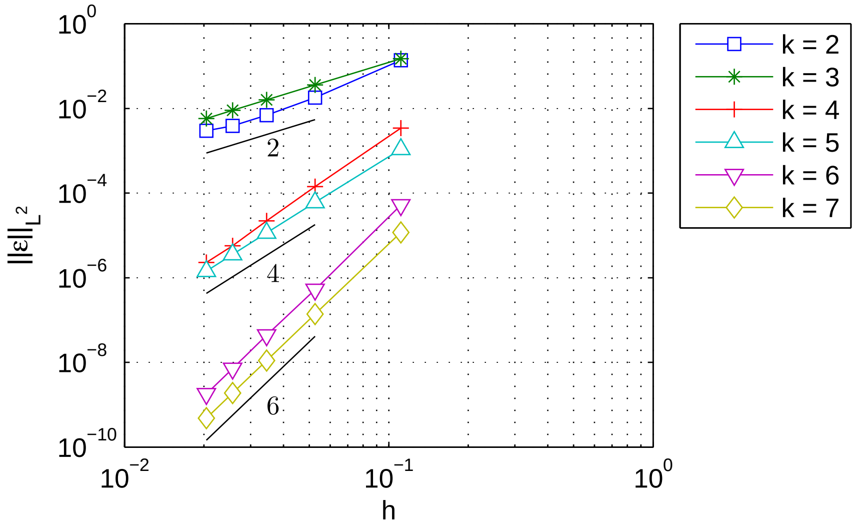


Figure 4.13: Error in the numerical solution of Poisson's equation in 2D, measured in the L^2-norm, versus knot spacing h.

Table 4.11: Logarithmic data points of knot spacing h and error norm $\|\epsilon\|_{L^2}$, plotted in Fig. 4.13.

	$\log(\|\epsilon\|_{L^2})$					
$\log(h)$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
-2.1972	-1.9863	-1.8995	-5.6754	-6.8236	-9.8586	-11.3494
-2.9444	-4.0128	-3.3309	-8.8554	-9.7183	-14.4614	-15.7959
-3.3673	-4.9723	-4.1389	-10.7248	-11.3776	-16.9395	-18.3214
-3.6636	-5.5540	-4.7085	-12.0728	-12.5502	-18.7513	-20.0859
-3.8918	-5.8217	-5.1546	-12.9825	-13.4592	-20.1312	-21.4508

Table 4.12: Logarithmic slopes (m) averaged over data intervals and the mean absolute deviation (MAD).

	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$
m	2.0294	1.9259	4.3030	3.9346	6.0453	5.9646
MAD	0.4612	0.0142	0.1824	0.0355	0.0925	0.0113

4.3.5 Conclusions

The results found in the two-dimensional test case agree with those found in the one-dimensional test case presented in Subsection 4.2. For spline orders $k = 3, 5, 7$ the estimated convergence order $k - 1$ is found. Again, for $k = 2, 4, 6$ higher-than-expected convergence of order k is witnessed. The superconvergent behaviour is conjectured to be caused by a fortunate superposition of numerical errors, as seen previously in Fig. 4.4. An anomalous result is encountered with $k = 4$ where the average order of convergence is $m = 4.3030$. Also, the results for even orders k exhibit relatively larger deviations than for odd k, see Table 4.12. These signs cast doubt on whether a solid theoretical estimate for the superconvergence phenomenon can be made.

Chapter 5

Conclusions

In this thesis a mimetic discretization method deploying B-splines as basis functions is introduced and succesfully applied to Poisson's equations in one and in two dimensions. Differential operators of the continuum problem, that is div, grad and curl, are represented by discrete analogues satisfying vector calculus identities of the continuous operators ($\mathrm{div} \circ \mathrm{curl} = 0$, $\mathrm{curl} \circ \mathrm{grad} = 0$). With the aforementioned discrete operators, the discretization method is applicable to a wide variety of elliptic partial differential equations (PDEs), producing locally conservative solutions of arbitrarily high orders.

In the first chapter a link is made between the nodal and edge basis functions deployed in the mimetic spectral element method (MSEM), of which the basis coefficients represent function values or integrals. In the MSEM, nodal basis functions are Lagrange polynomials and edge basis functions are found from the derivatives of those Lagrange polynomials. Comparable nodal and edge basis functions can be constructed by solving an interpolation or histopolation problem with basis splines. Histopolation is here understood as an integral-matching approximation of functions. Solutions of the histopolation problem are proven to exist under the conditions stated in Theorem 1 in Section 2.8. The construction of nodal and edge basis functions is not required, as basis splines can be used directly as basis functions in mimetic discretizations. Nodal and edge basis functions can be replaced respectively by B- and M-splines.

The discrete Hodge operator ($\star_h$) plays a key role in the discretization of the Laplace-Beltrami operator, required for numerical solutions of Poisson's equation. The discretization process could be simplified when the $\star_h$ operator is effectively realized by the mass matrix [8, 9, 27, 28], which eliminates the need for a dual grid. The construction of an explicit $\star_h$ operator brings several advantages on the other hand. Firstly, Neumann boundary conditions and material laws can be imposed strongly. Secondly, the problem is expressed naturally in a mixed formulation by introducing a dual quantity, which could be the quantity of interest in fact.

The Hodge operator is discretized according to its natural definition (3.39), based on the reduction and the reconstruction operators. Different field quantities are then stored on different grid elements, similar to staggered grids in finite volume methods. Inner-oriented quantities, e.g. line integrals, are housed on the primal grid and outer-oriented quantities, e.g. fluxes, on the dual grid. The $\star_h$ operator provides an invertible map between the two complices. The formal adjoint of a closed cell complex is an open complex, which is not compatible with spline functions. The subsequent solution is to close the open complex by boundary elements for which boundary conditions are available. Dirichlet boundary conditions are imposed on the boundary vertices augmenting an open primal grid and Neumann boundary conditions on the boundary faces augmenting an open dual grid.

The construction of an adjoint grid allows some freedom. One option is to stagger the knots of the dual grid with respect to the primal grid; the knots correspond to the “joints” of the polynomial pieces that comprise the spline. Another option is to reduce the polynomial order of the splines on the dual grid. The results of numerical tests in Section 4.2 indicate that the former provides faster-converging and more accurate solutions. Staggering the knots has the implication though that the dual grid does not adhere to the isogeometric paradigm: in isogeometric analysis the knots are retrieved from the geometry defined in the computer-aided design program.

In Chapter 4, the discretization method is applied to Poisson’s equation on one- and two-dimensional Cartesian meshes. The two-dimensional spline functions are defined as tensor products of one-dimensional splines and could be extended similarly to three dimensions. The error of the numerical solution is measured in the L^2-norm to analyse the convergence to the analytic solution when refining the knot spacings, i.e. increasing the number of polynomial pieces of the spline. In the one-dimensional test case, the grid convergence, using a dual grid with staggered knots, is of order k for even order ($k = 2, 4, 6, \dots$) splines and $k - 1$ for odd order ($k = 3, 5, 7, \dots$) splines, where k is the polynomial order. Using a dual grid with reduced order, the grid convergence is of order $k - 2$ for even order splines and $k - 1$ for odd order splines. Considering the accuracy of the obtained solutions, the dual grid with staggered knots is indubitably the best option.

In the two-dimensional test case, the dual grid is defined with staggered knots only. A grid convergence study supports the results of the one-dimensional test case. The grid convergence is of order k for even order splines and $k - 1$ for odd order splines. The alternating orders are however not supported by solid theory. Similar convergence properties are witnessed by others [4, 7, 10, 11] as well, but are not explained adequately and require more in-depth research. Error estimates found in literature [34] for conventional spline collocation methods predict orders of $k-1$ for first-order problems. Poisson’s equation in the mixed formulation is a combination of two first-order problems solved by collocation: one involves the gradient operator on the primal grid and one the divergence operator on the dual grid. A general convergence order of $k - 1$ can then be expected. Higher orders of k might be explained by a fortunate superposition of several error terms, as observed in Section 4.1.

All by all, B-splines can be deployed as basis functions in mimetic discretizations of elliptic PDEs. The numerical solution is of arbitrary order and locally conservative. The grid convergence is generally of order $k - 1$, one order lower than the polynomial order. For even order splines however, the convergence is of order k, but is not supported by theory. The dual grid can be constructed by staggering the knots with respect to the primal grid.

Chapter 6

Future work

The application of splines in numerical analysis is growing rapidly in popularity with the advent of isogeometric analysis. Mimetic discretizations is another relatively young topic that is expected to mature in the years to come. The potential of both research areas is recognised by the scientific community, publishing more and more articles each year. This thesis is by no means comprehensive and inclusive, leaving many possibilities unexplored and questions unanswered. This chapter includes a number of pointers to direct possible future work that builds upon the research presented in this thesis.

Adjoint coboundary operator

In this thesis the Laplace-Beltrami operator is discretized with the discrete Hodge $\star$ operator and the coboundary operator δ:

$$\Delta_h = \delta \star_h \delta \star_h + \star_h \delta \star_h \delta. \tag{6.1}$$

where the $\star_h$ operator is obtained according to the its natural definition (3.39). Another common approach [8, 9, 27, 28] is to derive an adjoint coboundary operator δ^* from mass matrices M:

$$\Delta_h = \delta^*\delta + \delta\delta^* = M_p^{-1} D_p^T M_{p+1} D_p + D_{p-1} M_{p-1}^{-1} D_{p-1}^T M_p. \tag{6.2}$$

There is no need to define a dual grid and the exact adjointness between differential operators is preserved:

$$(\delta^* a, b)_\Omega = (a, \delta b)_\Omega, \quad a \in C^p, b \in C^{p-1}. \tag{6.3}$$

The adjoint coboundary can also be discretized using the reduction operator $\mathcal{R}$ and reconstruction operator $\mathcal{I}$:

$$\delta^* = (-1)^p \mathcal{R} \star d \star \mathcal{I}, \tag{6.4}$$

which is the natural definition of the adjoint coboundary operator and closely related to collocation methods. Neither this discretization or Eq. (6.1) will give an exact adjoint with respect to the inner product:

$$(\delta^* a, b)_\Omega = (a, \delta b)_\Omega + O(h^s), \quad a \in C^p, b \in C^{p-1}. \tag{6.5}$$

The approximation might be justified by expected gains in computational efficiency. Collocation methods and alike are known to be low-cost alternatives to Galerkin methods, which require numerical quadrature to evaluate the inner products.

Derived Hodge $\star$ operator

An alternative to the natural $\star$ operator is the derived definition:

$$(a, b)_\Omega = \int_\Omega a \wedge \star b, \quad a, b \in C^p. \tag{6.6}$$

Evaluation of the inner product and wedge product for the basis functions gives the algebraic system:

$$M_p \mathbf{b} = W_{n-p}\{\star_h b\}, \tag{6.7}$$

in terms of the mass matrix M and the *wedge* or *pairing* matrix W. This discretization is regarded as a Galerkin method: the discrete form is projected orthogonally on the dual grid. The error, measured in the L^2-norm, is then minimised and is therefore indubitably smaller than for the natural definition of the $\star_h$ operator: the error is expected to be $O(h^k)$ for even *and* odd k. Note that this distinction does not exist in the mimetic spectral element method (MSEM): the spaces of discrete forms and their Hodge duals are identical, allowing a lossless map to the dual grid. It is however questionable if the accuracy of the derived $\star_h$ operator outweighs the computational cost of its implementation. For example, the discrete scalar Laplacian requires the evaluation of two mass matrices and two pairing matrices, necessitating highly-demanding numerical quadrature.

Local refinement

Much of the current research activities in isogeometric analysis concentrate on local refinement of splines, which is an essential capability for analysing realistic problems. The computer-aided design software provides geometries that are generally too coarse for accurate analysis. The tensor product nature of multivariate B-splines or NURBS however obstruct the local adaptation of the mesh. Sederberg [6, 36] introduced a generalisation of NURBS, called T-splines for allowing T-junctions in the meshes. With certain restrictions, T-splines are linearly independent and form a partition of unity. These properties make T-splines interesting candidates for locally adaptable basis functions in mimetic discretizations. Yet another approach to local refinement are hierarchical B-splines which is a multilevel refinement, leading to a hierarchy between the degrees of freedom.

Practical applications

The discretization method proposed in this thesis is directly applicable to realistic linear diffusion problems. For instance one can think of heat conduction, electromagnetism, Stokes flow, Darcy flow, and more. To allow more general geometries, a pullback operator must be defined to perform integration on curvilinear domains. The tensor product formulation of bivariate spline functions allows a natural extension to three dimensions. Also, more complex geometries can be achieved by connecting multiple *patches* or *superelements* of splines.

Bibliography

[1] I. Akkerman, Y. Bazilevs, V.M. Calo, T.J.R. Hughes, and S. Hulshoff. The role of continuity in residual-based variational multiscale modeling of turbulence. *Computational Mechanics*, 41:371–378, 2008.

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

[3] D. N. Arnold, R. S. Falk, and R. Winther. Finite Element Exterior Calculus: From Hodge Theory to Numerical Stability. *Bulletin (New Series) of the American Mathematical Society*, 47(2):281–354, 2010.

[4] F. Auricchio, L. Beirão da Veiga, T.J.R. Hughes, A. Reali, and G. Sangalli. Isogeometric collocation methods. *Mathematical Models and Methods in Applied Sciences*, 20(11), 2010.

[5] A. Back and E. Sonnendrücker. Spline discrete differential forms. *ESAIM: Proceedings*, 35:197–202, April 2012.

[6] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg. Isogeometric analysis using T-splines. *Comput. Methods Appl. Mech. Engrg.*, 199:229–263, 2010.

[7] L. Beirão da Veiga, C. Lovadina, and A. Reali. Avoiding shear locking for the Timoshenko beam problem via isogeometric collocation methods. *Comput. Methods Appl. Mech. Engrg.*, 241–244, 2012.

[8] P.B. Bochev and M. Gunzburger. Compatible discretizations of second-order elliptic problems. *Journal of Mathematical Sciences*, 136:3691–3705, 2006.

[9] P.B. Bochev and J.M. Hyman. Principles of mimetic discretizations of differential operators. In *Compatible Spatial Discretization*, volume 142 of *The IMA Volumes in mathematics and its applications*, pages 89–120. Springer, 2005.

[10] O. Botella. A velocity-pressure Navier-Stokes solver using a B-spline collocation method. In *Annual Research Briefs*, pages 403–421. Center for Turbulence Research, 1999.

[11] O. Botella. On a collocation B-spline method for the solution of the Navier-Stokes equations. *Computers & Fluids*, 31:397–420, 2002.

[12] S.C. Brenner and L.R. Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer, third edition, 2008.

[13] A. Buffa, J. Rivas, G. Sangalli, and R. Vázquez. Isogeometric discrete differential forms in three dimensions. *SIAM Journal on Numerical Analysis*, 49:818–844, 2011.

[14] W.L. Burke. *Applied differential geometry.* Cambridge University Press, 1985.

[15] W.L. Burke. Div, grad, curl are dead. Version 2.0, October 1995.

[16] W.L. Burke. Twisted forms: Twisted differential forms as they should be. Version 7.1, March 1995.

[17] J.A. Cottrell, T.J.R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA.* John Wiley & Sons Ltd, 2009.

[18] M.G. Cox. The Numerical Evaluation of B-Splines. *Journal of the Institute of Mathematics and its Applications*, 10:134–149, 1972.

[19] H.B. Curry and I.J. Schoenberg. On Pólya frequency functions IV: The fundamental spline functions and their limits. *Journal d'Analyse Mathématique*, 17:71–107, 1966.

[20] C. de Boor. On Calculating with B-Splines. *Journal of Approximation Theory*, 6:50–62, 1972.

[21] C. de Boor. *A Practical Guide to Splines*, volume 27 of *Applied Mathematical Sciences*. Springer, 2001.

[22] H. Flanders. *Differential Forms with Applications to the Physical Sciences.* Dover Publications, December 1989.

[23] T. Frankel. *The Geometry of Physics: An Introduction.* Cambridge University Press, 2nd edition, November 2003.

[24] M. Gerritsma. An Introduction to a Compatible Spectral Discretization Method. *Mechanics of Advanced Materials and Structures*, 19:48–67, 2012.

[25] R. Hiemstra. Isogeometric mimetic methods. Master's thesis, Delft University of Technology, 2011.

[26] R.R. Hiemstra, R.H.M. Huijsmans, and M.I. Gerritsma. High order gradient, curl and divergence conforming spaces, with an application to NURBS-based IsoGeometric Analysis. Eprint arXiv:1209.1793, submitted to Journal of Computational Physics, September 2012.

[27] R. Hiptmair. Discrete Hodge operators. *Numerische Mathematik*, 90:265–289, 2001.

[28] R. Hiptmair. Discrete Hodge-Operators: An Algebraic Perspective. *Progress In Electromagnetics Research*, 32:247–269, 2001.

[29] R. Hiptmair. Finite element in computational electromagnetism. *Acta Numerica*, 11:237–339, 2002.

[30] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computational Methods in Mechanics and Engineering*, 194:4134–4195, 2005.

[31] J.M. Hyman and J.C. Scovel. Deriving mimetic difference approximations to differential operators using algebraic topology. Los Alamos National Laboratory, unpublished report, 1988.

[32] J.J. Kreeft, A. Palha, and M.I. Gerritsma. Mimetic framework on curvilinear quadrilaterals of arbitrary order. At arXiv:1111.4304v1, 18 November 2011.

[33] L. Piegl and W. Tiller. *The NURBS book*. Springer-Verlag, 1995.

[34] R.D. Russell and L.F. Shampine. A Collocation Method for Boundary Value Problems. *Numer. Math.*, 19:1–28, 1972.

[35] I.J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions. *Quarterly of Applied Mathematics*, 47:45–99 and 112–141, 1946.

[36] M.A. Scott, X. Li, T.W. Sederberg, and T.J.R. Hughes. Local refinement of analysis-suitable T-splines. *Comput. Methods Appl. Mech. Engrg.*, 213–216:206–222, 2012.

[37] T. Tarhasaari, L. Kettunen, and A. Bossavit. Some realizations of a discrete Hodge operator: A reinterpretation of finite element techniques. *IEEE Transactions on Magnetics*, 35(3):1494–1497, May 1999.

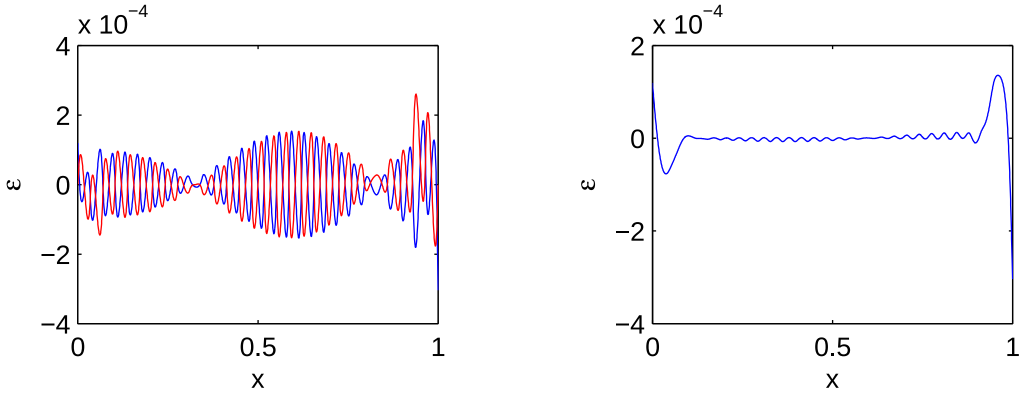
[38] D. Toshniwal. A geometic approach towards momentum conservation. Master's thesis, Delft University of Technology, 2012.

Appendix A

In Section 4.1, the projection error is examined for a sine function specifically. Similar observations can made for more general functions, such as the asymmetric function:

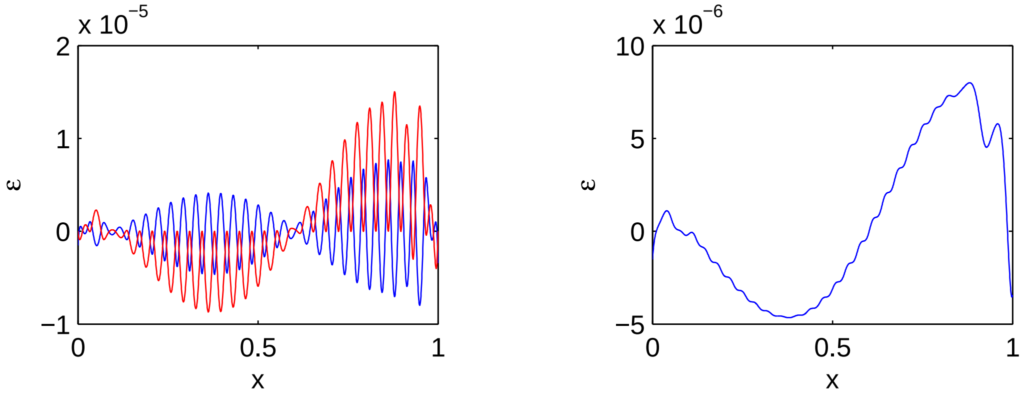
$$\alpha(x) = e^x \sin(2\pi x),$$

where the sine function is multiplied by an exponential function. The local projection errors are shown in Fig. A.1. Again, the second projection error counterbalances the first projection error for $k = 4$, which benefits the grid convergence of the total error. Similar results are expected for C^∞ functions in general.



(a) ϵ_1 (blue) and ϵ_2 (red), $k = 4$

(b) ϵ_{tot}, $k = 4$



(c) ϵ_1 (blue) and ϵ_2 (red), $k = 5$

(d) ϵ_{tot}, $k = 5$

Figure A.1: Local errors in the double projection $\widetilde{\pi} \star \pi\alpha$ for $k = 4$ and $k = 5$, with grid spacing $h = 0.0345$.