Isogeometric Analysis on Multiple Patches for Aerospace Applications

Thorben Konstantin Rusch

Geboren am 25. August 1995 in Ulm

6. September 2018

Bachelorarbeit Mathematik

Betreuer: Prof. Dr. Marc Alexander Schweitzer

Zweitgutachter: Dr. Martin Siggel

Institut für Numerische Simulation

Mathematisch-Naturwissenschaftliche Fakultät der

Rheinischen Friedrich-Wilhelms-Universität Bonn

und

Deutsches zentrum für Luft- und Raumfahrt

Abstract

Computer-driven simulations in the context of aerospace research are of paramount importance. These simulations are done by solving partial differential equations approximately with numerical methods. One famous and recent approach in this context is the isogeometric analysis (IGA) which is able to simulate directly on smooth geometries. Since in most real-world applications the geometries are given as CAD files, the isogeometric approach has to be extended to more complicated geometries, i.e. NURBS-based multiple patches. The contributions of this thesis are that we derive an isogeometric method on multiple patches. Since this a restricted variational problem for elliptic equations, we solve it with the generalized Lagrange multipliers method. Furthermore, we give an estimation for the integration complexity of IGA and compare it to the integration complexity of the classical finite element method (FEM). We demonstrate that the a-priori error of IGA on single patches is the same as encountered in the a-priori error analysis of FEM. In addition, we show that the condition number of IGA increases exponentially with increasing polynomial degree and that it is asymptotically of the same order as the condition number of FEM under h-refinement. Finally, we simulate a potential flow around a NACA airfoil in two dimensions using the isogeometric method on multiple patches. This provides information about the problems and challenges in the application of IGA in modern aerospace science.

Zusammenfassung

Computerbasierte Simulationen sind in der Luft- und Raumfahrtforschung von großer Bedeutung. Hierzu werden partielle Differentialgleichungen mit numerischen Methoden gelöst. Ein neuer Ansatz in diesem Zusammenhang ist die isogeometrische Analysis (IGA), welche direkt auf glatte Geometrien angewendet werden kann, ohne diese vorher approximieren zu müssen. Da die Geometrien in den meisten Anwendungen als CAD Dateien vorliegen, verallgemeinern wir die IGA auf sogenannten Multiple Patches. Die Beiträge dieser Arbeit umfassen, dass wir zunächst eine isogeometrische Methode auf Multiple Patches herleiten. Da dies ein restringiertes Variationsproblem für elliptische Gleichungen darstellt, lösen wir es mit der Lagrange Methode. Zusätzlich geben wir eine Abschätzung der Integrationskomplexität von IGA an und vergleichen diese mit der Integrationskomplexität der klassischen Finite Elemente Methode (FEM). Wir zeigen, dass der a-priori-Fehler von IGA auf Single Patches derselbe ist wie bei der a-priori Fehleranalyse von FEM. Darüber hinaus zeigen wir, dass die Kondition von IGA exponentiell bei zunehmendem Polynomgrad steigt und dass IGA unter h-Verfeinerung asymptotisch dieselbe Kondition hat wie die FEM. Zum Schluss simulieren wir mittels der IGA auf Multiple Patches eine Potentialströmung um ein zweidimensionales NACA Flügelprofil. Dies gibt uns Aufschluss über die Probleme und Herausforderungen, die bei der Anwendung von IGA in der modernen Luft- und Raumfahrtwissenschaft vorliegen.

Acknowledgements

I would like to thank my supervisor Prof. Dr. Marc Alexander Schweitzer for choosing the topic of this thesis as well as for his support and guidance.

I am very grateful for the good advice, support and patience of Dr. Martin Siggel who is co-supervisor of this thesis as well as my supervisor at the German Aerospace Center.

Last but not least, I would like to express my thanks to the staff of the High-Performance Computing Institute of the German Aerospace Center for the helpful discussions about the topic. Especially, I would like to thank my colleagues Dr. Margrit Klitz and Philipp Knechtges for their kind and very helpful support.

Contents

List of Figures

List of Tables

1 Introduction

In modern aerospace research computer-based techniques for analyzing materials and shapes are of major interest. A key example is the analysis of aerodynamic or aeroelastic properties of an aircraft. This can be done by using mathematical models derived from physical laws and behaviors. These models are usually formulated in terms of partial differential equations such as the Navier-Stokes equations that describe the motion of viscous fluid substances. Since most equations cannot be solved directly using standard analysis, the numerical analysis and scientific computing provide techniques for solving them approximately.
In this context there exist a variety of different approaches such as the finite element method (FEM), the finite difference method or the finite volume method. Among these the standard FEM is the most comparable to the isogeometric analysis (IGA) that we employ in this thesis.
IGA is a recent approach and was invented by Hughes et al. [2005]. The basic idea of isogeometric analysis is to use the description that already forms the underlying domain for solving partial differential equations approximately. This is a major advantage since IGA is able to simulate directly on CAD files (Computer Aided Design). These files are mostly based on non-uniform rational B-splines (NURBS). As an example, Figure 1 shows a smooth representation of an airplane geometry given as a NURBS-based CAD file. In contrast to classical FEM, IGA does not require a lengthy mesh creation process to approximate the geometry. This is a highly desirable feature since mesh generation is very expensive in most simulations. Furthermore, the choice of higher-order shape functions can be achieved by simply elevating the degree of the NURBS basis functions.
The contributions of this thesis are:

- we derive an isogeometric method on NURBS-based multiple patches by solving a restricted variational problem using the Lagrange multiplier method
- we provide an estimation for the integration complexity of IGA on single patches and compare it to the integration complexity of FEM
- we analyze the convergence behavior and condition numbers of the single patch method as well as of the multiple patch method and compare them to the convergence behavior and condition numbers of FEM
- we solve a prototype CFD (Computational Fluid Dynamics) problem on a two dimensional NACA airfoil geometry in order to validate the applicability of IGA to classical aerospace problems

The remainder of thesis is structured as follows. Section 2 introduces B-splines and NURBS together with some standard geometric algorithms. These algorithms serve as the refinement techniques for the isogeometric method. After that, we present the basic isogeometric approach on simplified NURBS surfaces, i.e. single patches. Additionally, we provide an a-priori error estimation for elliptic problems for isogeometric analysis on single patches, which we then compare to the a-priori error estimation of the standard FEM. Finally, we present a method for doing isogeometric simulations on multiple patches. The underlying coupling process of the various patches is motivated by solving a restricted variational problem. We handle these restrictions by a generalization of the Lagrange multiplier method on general Banach spaces. Section 3 presents numerical experiments on a single patch as well as on multiple patches. We illustrate the convergence behavior

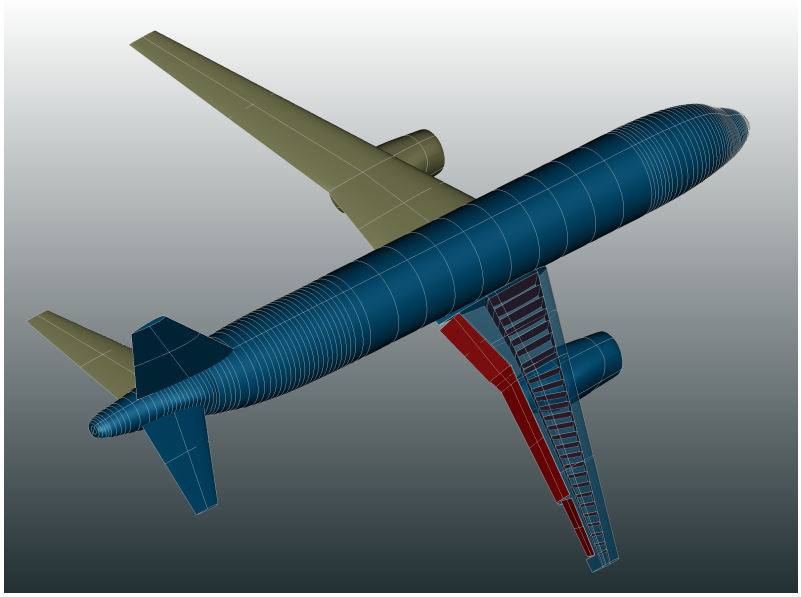

Figure 1: CAD example of an airplane

in the L^2 norm and in the H^1 semi-norm under h- and p-refinement in both, the single patch and the multiple patch setup. Furthermore, we analyze the condition numbers of the left-hand side matrices arising in the isogeometric discretization for both setups. Finally, we present a flow simulation of an incompressible laminar fluid around a NACA airfoil in two dimensions in the multiple patch setup, which is a prototype of a typical aerospace problem. We conclude this thesis in Section 4 by a discussion of the presented IGA and an outlook of possible future research topics.

2 Methods

2.1 Non-uniform rational B-splines

NURBS basis functions are the corner stones of IGA as well as represent most CAD files. In order to define NURBS, we first introduce B-splines and their basic properties. The following geometric principles are mostly based on the book of Piegl and Tiller [2012].

2.1.1 B-splines

In order to define B-spline basis functions we first have to define a knot vector.

Definition 2.1. *A non-decreasing sequence* $\Xi = \{\xi_1, \dots, \xi_{n+p+1}\}$ *where* $\xi_i \in \mathbb{R}$ *for all* $i = 1, \dots, n+p+1$ *is called a knot vector and its elements are called knots.*

The natural numbers n and p will refer in the following always to the amount of basis functions and their degree. If the knots are equidistant, the knot vector is called uniform, otherwise, it is called non-uniform. If a knot ξ_i appears in the sequence k times, it is called a multiple knot of multiplicity k. The half-open interval $[\xi_i, \xi_{i+1})$ is called the i-th knot span. We define the interior knots of the knot vector as the set $\{\xi_{p+1}, \dots, \xi_{n+1}\}$. The B-spline basis functions can now be defined by the method of De Boor [1972]:

Definition 2.2. *Let* $\Xi = \{\xi_1, \dots, \xi_{n+p+1}\}$ *be a knot vector. B-splines of degree* p *are defined recursively by*

$$N_{i,0}(\xi) = \begin{cases} 1, & \text{if } \xi_i \le \xi \le \xi_{i+1} \\ 0, & \text{else} \end{cases}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi).$$

The following list of the most important properties of B-spline basis functions determine their many desirable geometric characteristics:

1. Piecewise polynomial: $N_{i,p}$ is a p-th degree polynomial function in the knot span $[\xi_i, \xi_{i+1})$, where i is in $\{1, \dots, n+p\}$.
2. Continuity: If ξ_i has a multiplicity of k then
$$N_{i,p} \in C^{p-k}.$$
3. Partition of unity:
$$\sum_{i=1}^{n} N_{i,p}(\xi) = 1 \quad \forall \xi \in [\xi_{p+1}, \xi_{n+1}]$$
4. Non-negativity:
$$\text{For all } i, p : N_{i,p}(\xi) \ge 0 \quad \forall \xi \in [\xi_1, \xi_{n+p+1}]$$
5. Local-support:
$$N_{i,p}(\xi) = 0 \text{ if } \xi \notin [\xi_i, \xi_{i+p+1})$$
6. In a given knot span $[\xi_j, \xi_{j+1})$ only the $p+1$ B-splines $N_{j-p,p}, \dots, N_{j,p}$ are nonzero.

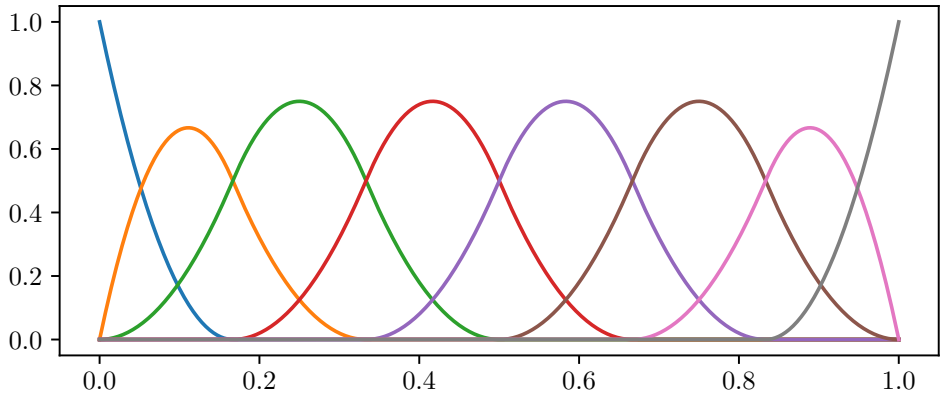

Figure 2: Quadratic B-splines with equidistant interior knots

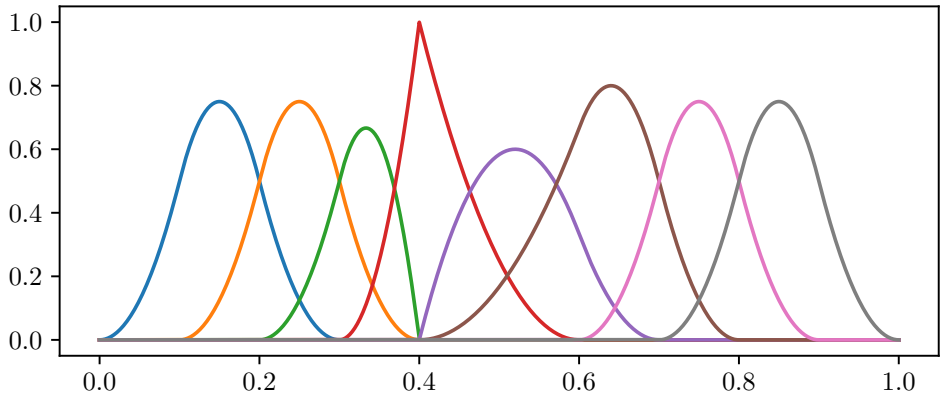

Figure 3: Quadratic B-splines with a double multiplicity at knot $\xi_5 = 0.4$

The derivative of a B-spline basis function is given by

$$\frac{\partial}{\partial \xi} N_{i,p}(\xi) = \frac{p}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi).$$

Figure 2 illustrates quadratic B-spline basis functions based on the knot vector $\Xi = \{0, 0, 0, 0.1, \ldots, 0.9, 1, 1, 1\}$ with uniform interior knots. In comparison, Figure 3 shows also quadratic B-spline basis functions, but based on a non-uniform knot vector with a double multiplicity at $\xi_5 = 0.4$. The reduced $C^{2-2} = C^0$ continuity of the function $N_{5,2}$ can be seen very well at $\xi = 0.4$.

2.1.2 B-spline curves and surfaces

B-spline basis functions are used to define geometries such as curves or surfaces. A B-spline curve can be represented by a linear combination of the basis functions with some vectors in $\mathbb{R}^d$ called the control points.

Definition 2.3. *Let $N_{1,p}, \ldots, N_{n,p}$ be n B-spline basis functions of degree p based on the*

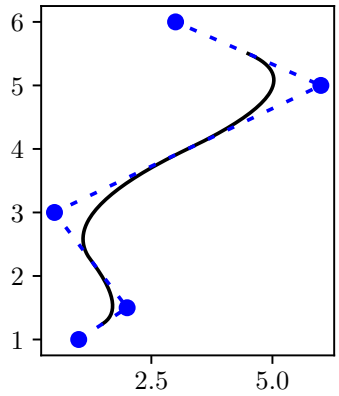

Figure 4: Open curve

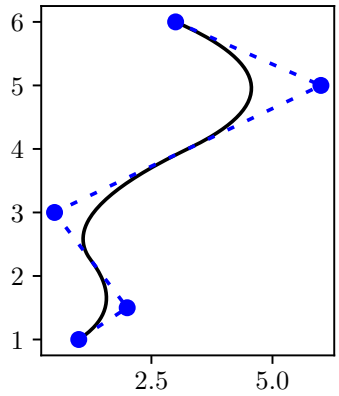

Figure 5: Clamped curve

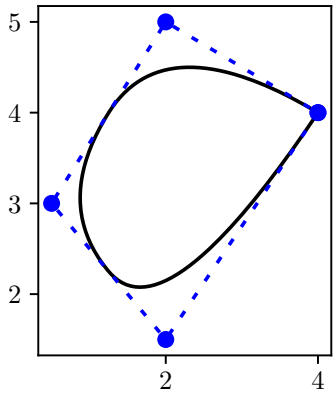

Figure 6: Closed curve

knot vector $\Xi = \{\xi_1, ..., \xi_{n+p+1}\}$. *A B-spline curve of degree* p *in* $\mathbb{R}^d$ *is given by*

$$F(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) P_i \quad \xi \in [\xi_{p+1}, \xi_{n+1}],$$

where $P_i \in \mathbb{R}^d$ *are the control points.*

Since B-spline basis functions are piecewise polynomial functions, the B-spline curves are also piecewise polynomials. If the first and last knot of a B-spline curve have a multiplicity of $p+1$, the corresponding B-spline curve is called a clamped curve. These curves have the characteristic of interpolating the first and last control point. Additionally, a curve whose beginning and end define the same point in $\mathbb{R}^n$ are called closed curves. Moreover, if the knot vector has no special structure or shape, the curve is called an open curve. These curves do not necessarily need to interpolate any of their control points.
Figure 4 shows a common open B-spline curve. In contrast, Figure 5 illustrates the curve of Figure 4, based on a rearranged knot vector with $p+1$ multiplicities at its first and last entries. This results in a clamped curve which interpolates the first and the last control point. Additionally, Figure 6 shows a closed B-spline curve based on a knot vector with $p+1$ multiplicities at its first and last entries. All of them are illustrated with their control net which visualizes the order of the control points.
B-spline surfaces can be constructed by using a tensoric basis of B-splines in two directions.

Definition 2.4. *Let* $\{N_{1,p}, \ldots, N_{n,p}\}$, $\{M_{1,q}, \ldots, M_{m,q}\}$ *be two sets of B-spline basis functions based on the two knot vectors* $\Xi = \{\xi_1, ..., \xi_{n+p+1}\}$, $\Theta = \{\theta_1, \ldots, \theta_{m+q+1}\}$. *A B-spline surface in* $\mathbb{R}^d$ *of degree* p, q *is defined by*

$$F(\xi, \theta) = \sum_{i=1}^{n} \sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\theta) P_{ij} \quad (\xi, \theta) \in [\xi_{p+1}, \xi_{n+1}] \times [\theta_{q+1}, \theta_{m+1}],$$

where the $P_{ij} \in \mathbb{R}^d$ *are again the control points.*

We define $\hat{\Omega} := [\xi_{p+1}, \xi_{n+1}] \times [\theta_{q+1}, \theta_{m+1}]$ as the parametric space of the surface. In the following, the parameter space will always be the unit square, i.e. $\hat{\Omega} = [0,1]^2$. We call the image of F the physical space and will denote it as $\mathrm{Im}(F) = \Omega$. Additionally, we call F the

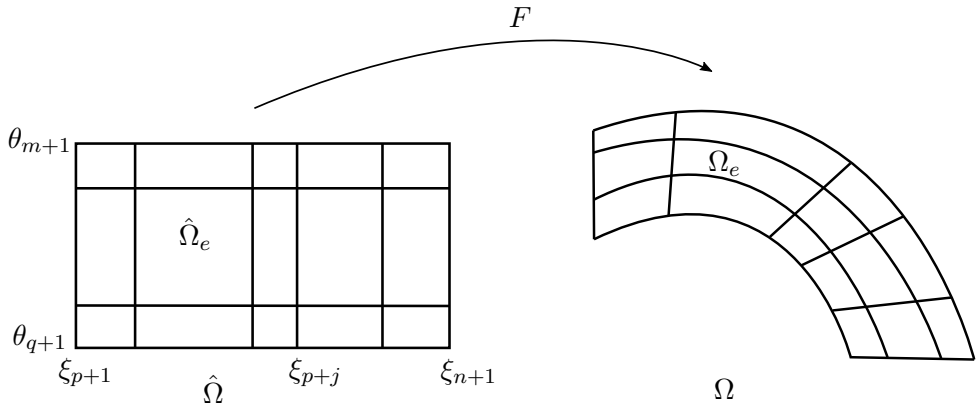

Figure 7: Geometrical mapping of a B-spline surface

geometrical map. We assume that $F : \hat{\Omega} \longrightarrow \Omega$ is a diffeomorphism from the parameter space to the physical one. This means that the geometries are not self-overlapping and thus the inverse F^{-1} exists. Elements in the physical space are defined by its partition, which is generated by mapping the knots of the parameter space to the physical space with F. We denote them as Ω_e and the corresponding elements in the parameter space as $\hat{\Omega}_e = F^{-1}(\Omega_e)$. Together with the fifth property of B-splines, it follows that only $p+1$ B-spline basis functions are non-zero on each parametric element. These definitions can be used analogously for B-spline curves. Similar to B-spline curves, B-spline surfaces can be open, clamped and closed. Figure 7 illustrates the parameter and the physical space of a B-spline surface. Especially the description of the elements Ω_e in the physical space by the geometrical mapping F can be seen.

2.1.3 Refinement techniques

In order to use the B-spline basis functions as a basis for an FE approach, we should be able to refine them without altering the underlying geometry or its parameterization. That means we will change the basis functions as well as the underlying knot vector under the constraint that the geometrical mapping F will remain unchanged.

h-refinement

A way to increase the amount of basis functions without changing the underlying geometry and its degree is the knot insertion approach. We will use an algorithm, which was first presented by Boehm [1980]. This method is able to insert a single knot into the knot vector and thus either increases the multiplicity of an already existing knot or inserts a completely new entry in the knot vector. There are also methods that are able to insert many knots at once which is called knot refinement. In this thesis we will focus on the knot insertion method and when it will be necessary to insert more than one knot, we will insert them sequentially.
Let $\Xi = \{\xi_1, \ldots, \xi_{n+p+1}\}$ be a knot vector of a curve and $\bar{\xi} \in [\xi_k, \xi_{k+1})$ with $p+1 \leq k \leq n$. The knot insertion method yields a new knot vector $\bar{\Xi} = \{\xi_1, \ldots, \xi_k, \bar{\xi}, \xi_{k+1}, \ldots, \xi_{n+p+1}\}$ with $\Xi \subset \bar{\Xi}$. The new $n+1$ basis functions are defined similar to the n basis functions by the recursion formula. The new $n+1$ control points $\bar{\mathcal{P}} = [\bar{P}_1, \ldots, \bar{P}_{n+1}]^\top$ are convex combinations of the already known control points $\mathcal{P} = [P_1, \ldots, P_n]^\top$:

$$\bar{P}_i = \alpha_i P_i + (1 - \alpha_i) P_{i-1},$$

where

$$\alpha_i = \begin{cases} 1, & \text{if } i \leq k - p \\ \dfrac{\bar{\xi} - \xi_i}{\xi_{i+p} - \xi_i}, & \text{if } k - p + 1 \leq i \leq k \\ 0, & \text{if } i \geq k + 1. \end{cases}$$

From this equation it follows that only p new control points have to be computed. Note that the inserted knot $\bar{\xi}$ cannot be chosen as an already existing knot with multiplicity of $p+1$. Hence, if we have an open knot vector we are not able to increase the multiplicity of the first or the last knot.

p-refinement

Another way to enrich the basis functions without changing the underlying geometry or its parameterization is to increase their degree. We will only consider the approach suggested by Piegl and Tiller [1994] although there exist several other methods in this context. This method is based on the Bézier decomposition of B-spline geometries. Originally a Bézier curve is defined as a linear combination of Bernstein polynomials

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad i \leq n, t \in [0,1]$$

and control points. One can show that a Bézier curve is equivalent to a B-spline curve with no interior knots and interpolating control points at the beginning and end. This follows directly from the conversion algorithm between Bézier and B-spline curves, which can be found in Section 5.10 of Prautzsch et al. [2013]. We decompose a B-spline curve in Bézier segments by increasing the multiplicities of the interior knots of the knot vector to p. This can be done using the knot insertion method described in the h-refinement section. As a consequence, this leads to interpolating control points at every interior knot which forms the decomposition. Because of the mentioned equivalence, the segments are Bézier curves. We can elevate the degree of Bézier curves by increasing the multiplicities of the first as well as of the last knot. The new control points can then be computed by a matrix computation as described by Farouki and Rajan [1988]. The degree elevated segments can be joined together to form a B-spline curve again. The unnecessary interior knots that were inserted to create the Bézier segments can be removed with a knot removal algorithm which is basically the inversion of the knot insertion method. One well-known approach was suggested by Tiller [1992].

k-refinement

One disadvantage of p-refinement is that the continuity of a B-spline basis function at the corresponding knot remains the same after elevating the degree. Thus, if we first insert a distinct knot $\bar{\xi}$, the corresponding B-spline basis function has a continuity of C^{p-1} at $\bar{\xi}$. After elevating the degree to q the basis function still has $p-1$ times continuous derivatives at $\bar{\xi}$. On the other hand, if we first begin by elevating the degree to q and inserting a unique knot afterwards, the corresponding basis has $q-1$ continuous derivatives at the inserted knot. This procedure is called k-refinement.

2.1.4 NURBS curves and surfaces

Although we can represent a lot of curves and surfaces with B-splines, there are some shapes that cannot be covered by them. For example, it is not possible to represent rotational symmetric geometries exactly. This fact was proven for a simple circle curve by Piegl and Tiller [1989]. This is one major reason most CAD files are based on the so-called non-uniform rational B-splines (NURBS), which are weighted and rational B-spline basis functions. These functions are able to create perfect circle-like shapes without error.

Definition 2.5. *Let $N_{1,p}, \dots, N_{n,p}$ be n B-spline basis functions of degree p based on the knot vector $\Xi = \{\xi_1, ..., \xi_{n+p+1}\}$. The NURBS basis functions of degree p are defined by*

$$R_i^p(\xi) = \frac{N_{i,p}(\xi) w_i}{\sum_{j=1}^{n} N_{j,p}(\xi) w_j},$$

where $w = [w_1, \dots, w_n]$ with $w_i \in \mathbb{R}$ are called weights. For better readability, we abbreviate the numerator and denominator

$$R_i^p(\xi) = \frac{A_i(\xi)}{w(\xi)},$$

where $A_i(\xi) = N_{i,p}(\xi) w_i$ and $w(\xi) = \sum_{j=1}^{n} N_{j,p}(\xi) w_j$. In the context of NURBS surfaces we define the bivariate NURBS basis functions of degree p, q by

$$R_{i,j}^{p,q}(\xi, \theta) = \frac{N_{i,p}(\xi) M_{j,q}(\theta) w_{ij}}{\sum_{k=1}^{n} \sum_{l=1}^{m} N_{k,p}(\xi) M_{l,q}(\theta) w_{kl}},$$

where again the $w_{i,j}$ are weights and the $N_{i,p}, M_{j,q}$ are B-spline basis functions of degree p, and q respectively with $i = 1, \dots, n$ and $j = 1, \dots, m$.

One can show that most of the B-spline properties also hold for NURBS basis functions. Together with the quotient rule we can obtain the derivatives of NURBS basis functions by

$$\frac{\partial}{\partial \xi} R_i^p(\xi) = \frac{\frac{\partial}{\partial \xi} A_i(\xi) w(\xi) - A_i(\xi) \frac{\partial}{\partial \xi} w(\xi)}{w(\xi)^2}$$

In addition, we can also use the B-spline definitions for NURBS curves and surfaces. Similar to B-spline geometries we can construct NURBS curves and surfaces as linear combinations of these basis functions with control points.

Definition 2.6. *For R_i^p and $R_{i,j}^{p,q}$ as in Definition 2.5 and control points P_i,$P_{ij} \in \mathbb{R}^d$, a NURBS curve of degree p is defined by*

$$F(\xi) = \sum_{i=1}^{n} R_i^p(\xi) P_i \quad \xi \in \hat{\Omega},$$

a NURBS surface of degree p, q by

$$F(\xi, \theta) = \sum_{i=1,j=1}^{n,m} R_{i,j}^{p,q}(\xi, \theta) P_{ij} \quad (\xi, \theta) \in \hat{\Omega},$$

with $i = 1, \dots, n$ and $j = 1, \dots, m$.

A bivariate NURBS geometry is called a single patch if it is completely parameterized by only one geometrical mapping. If more than one mapping is needed for defining the whole geometry, it is called a multiple patch. In general, CAD files consist of multiple patches. The h, p, k-refinement techniques of the B-spline section can also be applied to NURBS files by representing them as B-spline geometries. That means a NURBS curve

$$F(\xi) = \sum_{i=1}^{n} R_i^p(\xi) P_i,$$

with $P_i \in \mathbb{R}^d$ can be represented as a projection of a higher-dimensional B-spline curve

$$\bar{F}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \bar{P}_i,$$

with $\bar{P}_i = (w_i P_i^1, \ldots, w_i P_i^d, w_i)^\top \in \mathbb{R}^{d+1}$. This geometry can then be refined by one of the described techniques. After that the geometry is projected back onto the d dimensional space by using the last dimension of the control points as the weights again.

2.2 IGA on a single patch

The basic idea of isogeometric analysis is to use the same basis functions underlying the geometrical description for discretizing the domain as well as for spanning the finite dimensional solution space. We will consider the elliptic Poisson equation with homogeneous Dirichlet boundary condition as a reference problem:
Let $\Omega \in \mathbb{R}^n$ be an open domain with the boundary $\partial\Omega$. Find $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\triangle u &= f \text{ in } \Omega \\ u &= 0 \text{ on } \partial\Omega, \end{aligned} \tag{2.1}$$

where $f : \Omega \to \mathbb{R}$. For simplicity, we first assume that Ω is a Jordan domain, that $\partial\Omega$ is a C^∞ manifold and that $u, f \in C^\infty$.

2.2.1 Variational problem and weak formulation

For the weak formulation of the Poisson problem, the first equation of (2.1) is multiplied with a test function $v \in \mathcal{V} \coloneqq \{\, u \in H^1(\Omega) \mid u\big|_{\partial\Omega} = 0 \,\}$, where H^1 is the Sobolev space $W^{1,2}$. Both sides of the resulting equation are then integrated over Ω

$$-\int_\Omega \triangle u v \,\mathrm{d}\Omega = \int_\Omega f v \,\mathrm{d}\Omega.$$

Using Green's lemma we can obtain

$$\underbrace{\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}\Omega}_{a(u,v)} - \underbrace{\int_{\partial\Omega} v \frac{\partial u}{\partial \nu} \,\mathrm{d}S}_{0} = \underbrace{\int_\Omega f v \,\mathrm{d}\Omega}_{L(v)}. \tag{2.2}$$

The weak formulation of the Poisson equation now weakens the previously made C^∞ constraint:
Find $u \in \mathcal{V}$ such that

$$a(u, v) = L(v) \quad \forall v \in \mathcal{V}. \tag{2.3}$$

Interestingly, the unique solution of the weak formulation also solves the following variational problem:

$$J(v) = \frac{1}{2}a(v,v) - L(v) \longrightarrow \min_{v\in\mathcal{V}} \tag{2.4}$$

This relation is proven in a more general way in the following theorem.

Theorem 2.7 (Riesz representation theorem)**.** *Let $\mathbb{V}$ be a Hilbert space with inner product $a(\cdot,\cdot) : \mathbb{V}\times\mathbb{V} \to \mathbb{R}$ and the induced norm $\|v\|_{\mathbb{V}} = a(v,v)^{\frac{1}{2}}$. Then for every continuous linear functional $L \in \mathbb{V}^*$ there exist a unique $u \in \mathbb{V}$ such that*

$$a(u,v) = L(v) \quad \forall v \in \mathbb{V}.$$

Moreover, u is the unique solution of the variational problem

$$J(v) = \frac{1}{2}a(v,v) - L(v) \longrightarrow \min_{v\in\mathbb{V}}$$

Proof. The following proof can be found in Chapter 4 of Alt [1992].
Due to the continuity of L there exists a c such that

$$|L(v)| \le c\|v\|_{\mathbb{V}},$$

and therefore

$$J(v) \ge \frac{1}{2}\|v\|_{\mathbb{V}}^2 - c\|v\|_{\mathbb{V}} = \frac{1}{2}(\|v\|_{\mathbb{V}} - c)^2 - \frac{1}{2}c^2 \ge -\frac{1}{2}c^2.$$

Hence, the functional J is bounded below and there exists an infimum

$$d = \inf_{v\in\mathbb{V}} J(v).$$

Let $\{v_k\}_{k\in\mathbb{N}}$ be a sequence in $\mathbb{V}$ with $J(v_k) \xrightarrow{\lim\limits_{k\to\infty}} d$. With the parallelogram law in Hilbert spaces follows

$$\|v_k - v_l\|_{\mathbb{V}}^2 + \|v_k + v_l\|_{\mathbb{V}}^2 = 2\|v_k\|_{\mathbb{V}}^2 + 2\|v_l\|_{\mathbb{V}}^2$$

Finally, we have

$$\begin{aligned}
&\|v_k - v_l\|_{\mathbb{V}}^2 \\
&= 2\|v_k\|_{\mathbb{V}}^2 + \|v_l\|_{\mathbb{V}}^2 - 4\left\|\frac{v_k+v_l}{2}\right\|_{\mathbb{V}}^2 - 4L(v_k) - 4L(v_l) + 8L\left(\frac{v_k+v_l}{2}\right) \\
&= 4J(v_k) + 4J(v_l) - 8J\left(\frac{v_k+v_l}{2}\right) \\
&\le 4J(v_k) + 4J(v_l) - 8d \longrightarrow 0
\end{aligned}$$

for $k,l \to \infty$. Therefore $\{v_k\}_{k\in\mathbb{N}}$ is a Cauchy sequence which has a limit u in $\mathbb{V}$ due the completeness of Hilbert spaces. With the continuity of J follows that $J(u) = d$ and u is the solution of the variational problem.
We will now prove that every solution of the variational problem is also a solution to the equation. We can reformulate

$$\begin{aligned}
\Phi(\epsilon) = J(u+\epsilon v) &= \frac{1}{2}a(u+\epsilon v, u+\epsilon v) - L(u+\epsilon v) \\
&= \frac{1}{2}a(u,u) + \epsilon a(u,v) + \frac{\epsilon^2}{2}a(v,v) - L(u) - \epsilon L(v)
\end{aligned}$$

for every $v \in \mathbb{V}$. Finally the necessary criterion for a minimum yields the condition

$$0 = \Phi'(0) = a(u, v) - L(v) \text{ for every } v \in \mathbb{V}$$

Lastly we show the uniqueness of the solution. Let u_1, u_2 be two solutions of the equation. Their difference is

$$a(u_1 - u_2, v) = 0 \quad \forall v \in \mathbb{V}$$

which only holds true for every $v \in \mathbb{V}$ if $u_1 - u_2 = 0$ and therefore $u_1 = u_2$. □

This theorem can be applied to the weak formulation (2.3) if the map $a(\cdot, \cdot)$ from Equation (2.2) forms an inner product on $\mathcal{V}$ fulfilling the Hilbert space properties. Due to the Poincaré inequality there exists a constant c such that

$$\|v\|_{L^2} \leq c\|\nabla v\|_{L^2} \quad \forall v \in \mathcal{V}.$$

Applying this to the H^1 norm we get for $v \in \mathcal{V} \subset H^1$:

$$\begin{aligned}\|v\|_{\mathcal{V}} = \|\nabla v\|_{L^2(\Omega)} &\leq \left(\|v\|^2_{L^2(\Omega)} + \|\nabla v\|^2_{L^2(\Omega)}\right)^{1/2} = \|v\|_{H^1(\Omega)} \\ &\leq \left(c\|\nabla v\|^2_{L^2(\Omega)} + \|\nabla v\|^2_{L^2(\Omega)}\right)^{1/2} \leq C\|\nabla v\|_{L^2(\Omega)} \leq Ca(v, v)^{1/2} = C\|v\|_{\mathcal{V}}.\end{aligned}$$

From the equivalence of the norms on $\mathcal{V}$ it follows immediately that $a(\cdot, \cdot)$ is an inner product on $\mathcal{V}$ such that it is a complete metric space under the norm $\|\cdot\|_{\mathcal{V}} = a(\cdot, \cdot)^{1/2}$. Hence, the representation theorem can be used to obtain the existence and uniqueness of the solution for the weak formulation. The existence and uniqueness can also be shown without the representation theorem by using the Lions-Lax-Milgram theorem. However, the variational derivation was chosen as a preparation for the subsequent motivation of the multiple patch method.

2.2.2 Galerkin approach

In order to solve the elliptic equation (2.1) approximately, we restrict the related variational problem to a finite dimensional subspace $\mathcal{V}^h \subset \mathcal{V}$:

$$J(v) = \frac{1}{2}a(v, v) - L(v) \longrightarrow \min_{v \in \mathcal{V}^h}. \tag{2.5}$$

This problem has the same property which was proven in the infinite dimensional case. That means every solution $u^h \in \mathcal{V}^h$ of (2.5) satisfies

$$a(u^h, v) = L(v) \quad \forall v \in \mathcal{V}^h. \tag{2.6}$$

Let $\{\phi_1, \phi_2, ..., \phi_N\}$ be a basis of the finite dimensional subspace $\mathcal{V}^h$. Then

$$a(u^h, \phi_i) = L(\phi_i) \quad \forall i = 1, ..., N$$

is equivalent to (2.6). By supposing

$$u^h = \sum_{j=1}^{N} u_j \phi_j \quad \text{for some } u_j \in \mathbb{R},$$

in combination with the bilinearity of $a(\cdot,\cdot)$, we get

$$\sum_{j=1}^{N} a(\phi_j, \phi_i) u_j = L(\phi_i) \quad \forall i = 1, ..., N.$$

We can rewrite this in matrix notation as

$$Ku = f,$$

where

$$K_{i,j} = a(\phi_j, \phi_i), \quad f_i = L(\phi_i).$$

In most cases we are interested in non-homogeneous Dirichlet boundary conditions, i.e. solving

$$\begin{aligned} -\triangle u &= f \text{ in } \Omega \\ u &= g \text{ on } \partial\Omega, \end{aligned}$$

where $f\colon\ \Omega \longrightarrow \mathbb{R}$ and $g\colon\ \partial\Omega \longrightarrow \mathbb{R}$. Hence, we search for a solution $\bar{u}^h$ in the space $\mathcal{S}^h \subset \mathcal{S} \coloneqq \{\, u \in H^1(\Omega) \mid u\big|_{\partial\Omega} = g \,\}$ such that the weak formulation (2.6) is fulfilled. It can be shown that given a function $g^h \in \mathcal{S}^h$, then for every $\bar{u}^h \in \mathcal{S}^h$ there exists a unique $u^h \in \mathcal{V}^h$ such that $\bar{u}^h = u^h + g^h$. Therefore, given $g^h \in \mathcal{S}^h$ we can search for $\bar{u}^h = u^h + g^h$ such that

$$a(\bar{u}^h, v^h) = L(v^h) \quad \forall v^h \in \mathcal{V}^h.$$

Recalling the bilinearity of $a(\cdot,\cdot)$ the equation can be rewritten as

$$a(u^h, v^h) = L(v^h) - a(v^h, g^h). \tag{2.7}$$

We call the function g^h the lifting and point to the fact that such a function does not necessarily exist in every $\mathcal{S}^h$. However, we will limit ourselves to the case where the existence of those functions can be ensured.

2.2.3 Matrix equation

Consider a single NURBS patch Ω with the parameterization given by the geometrical mapping

$$F(\xi) = \sum_{A' \in A} \hat{R}_{A'}(\xi) P_{A'},$$

where A' is a two dimensional multiindex in the case of NURBS surfaces. In order to solve partial differential equations like (2.1) directly on Ω the isogeometric approach is used. That means the NURBS basis functions $\{R_{A'}\} = \{\hat{R}_{A'} \circ F^{-1}\}$ of the surface Ω are used to span the subspace $\mathcal{V}^h$. Hence, the finite dimensional subspace is the push-back $\mathcal{V}^h = \operatorname{span}(\{R_{A'}\}) = \operatorname{span}(\{\hat{R}_{A'} \circ F^{-1}\})$.
Applying the Galerkin method to $\mathcal{V}^h$ yields the following system of equations:

$$\sum_{B' \in A} a(R_{A'}, R_{B'}) u_{B'} = L(R_{A'}) \quad \forall A' \in A,$$

where $u_{B'} \in \mathbb{R}$ are called the control variables. This leads to a matrix K on the left-hand side and a vector f on the right-hand side which we call the global stiffness matrix and the

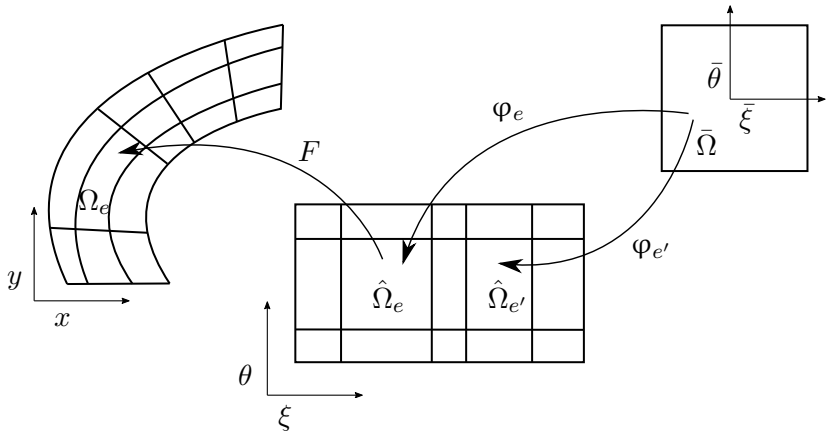

Figure 8: Transformation scheme for surfaces

global force vector, respectively. One advantage of NURBS geometries is that a function f integrated over Ω can be rearranged to

$$\int_{\Omega} f(x)\,\mathrm{d}\Omega = \sum_{e} \int_{\Omega_e} f(x)\,\mathrm{d}\Omega, \tag{2.8}$$

where Ω_e denotes an element on Ω. Hence, instead of computing the global stiffness matrix and the global force vector we can compute the local $K^e_{A',B'}$ and $f^e_{A'}$ by

$$\begin{aligned} K^e_{A',B'} &= \int_{\Omega_e} \nabla R_{A'} \cdot \nabla R_{B'}\,\mathrm{d}\Omega \\ f^e_{A'} &= \int_{\Omega_e} f(x) R_{A'}\,\mathrm{d}\Omega \end{aligned} \tag{2.9}$$

and assemble them to the global one afterwards. The advantage of this approach is that $R_{A'}$ equals to zero for most NURBS basis functions on an element Ω_e. By using the transformation theorem twice we have

$$\begin{aligned} \int_{\Omega_e} f(x)\,\mathrm{d}\Omega &= \int_{\hat{\Omega}_e} f \circ F\ |det(DF)|\,\mathrm{d}\Omega \\ &= \int_{\bar{\Omega}} f \circ F \circ \varphi_e\ |det(DF(\varphi_e))|\ |det(D\varphi_e)|\,\mathrm{d}\Omega \end{aligned} \tag{2.10}$$

for every $f \in \Omega_e$. $\bar{\Omega}$ denotes the so-called parent domain of every parametric element $\hat{\Omega}_e$, and $\varphi_e : \bar{\Omega} \to \hat{\Omega}_e$ is the diffeomorphism used for the transformation. For the sake of simplicity we choose $\bar{\Omega} = [-1, 1]^{\dim(\hat{\Omega})}$.
Finally, the last thing needed for computing (2.9) is $\nabla R_{A'}$. Using the chain rule we obtain

$$\nabla R_{A'} = (DF^{-1})^{\top} \nabla \hat{R}_{A'}.$$

To summarize,

$$K^e_{A',B'} = \int_{\bar{\Omega}} ((DF^{-1})^\top \nabla \hat{R}_{A'}) \circ \varphi_e \cdot ((DF^{-1})^\top \nabla \hat{R}_{B'}) \circ \varphi_e \; |det(DF(\varphi_e))| \; |det(D\varphi_e)| \, \mathrm{d}\Omega \tag{2.11}$$

$$f^e_{A'} = \int_{\bar{\Omega}} f(F(\varphi_e)) \hat{R}_{A'}(\varphi_e) \; |det(DF(\varphi_e))| |det(D\varphi_e)| \, \mathrm{d}\Omega. \tag{2.12}$$

With (2.8)

$$K_{A',B'} = \sum_e K^e_{A',B'}$$
$$f_{A'} = \sum_e f^e_{A'}.$$

These integrals can be computed using numerical quadrature rules like the Gauss-Legendre method. In order to determine the required amount of quadrature points in each parametric direction on every element for an exact integration, the following simplification of the integral (2.11) has to be analyzed:

$$\int_{\bar{\Omega}} ((DF^{-1})^\top \nabla \hat{R}) \cdot ((DF^{-1})^\top \nabla \hat{R}) |det(DF)| \, \mathrm{d}\Omega.$$

The basis functions $\hat{R}$ and the geometrical mapping F as well as its inverse F^{-1} are assumed to be piecewise polynomials of degree p and q, respectively.
Hence, the differentiated basis functions $\nabla \hat{R}$ have a degree of $p-1$ and the differentiated geometrical mapping DF (and its differentiated inverse $(DF^{-1})^\top$) a degree of $q-1$. The determinant $|det(DF)|$ has therefore a degree of $d(q-1)$, where d is the dimension of the geometry. Multiplying the polynomials leads to a polynomial integrand of degree $2(p-1)+(d+2)(q-1)$. There exists a theorem regarding the exactness of the Gauss quadrature rule on polynomials which states that n quadrature points are sufficient for integrating polynomials exactly up to an degree of $2n-1$.
Therefore, by assuming that $q = p$ and $d = 2$ a total of $3p-2$ points is needed in each parametric direction to integrate these functions exactly. Hence, 49 quadrature points on each element are required for a cubic two dimensional geometry.
Hughes et al. [2010] argue that for most geometries the terms depending on F can be ignored. This can be done since they change slowly compared to the polynomial numerator of the NURBS basis functions as they are piecewise smooth functions on the initial coarse mesh where the geometry is exactly represented. This yields a required amount of p quadrature points in each direction, which is the same as in standard FEM. For a detailed proof regarding the FEM integration complexity we refer to Ern and Guermond [2013]. However, the is only the best case integration complexity of IGA and in general it may be much higher.
Finally, in order to fulfill also non-homogeneous Dirichlet boundary conditions like $u = g$ on $\partial\Omega$, the basis functions $\{R_{\hat{A}'}\}_{\hat{A}' \in \bar{A}}$ with $\hat{A} \subset A$ that are not zero on the boundary $\partial\Omega$ are chosen to represent g^h by

$$g^h = \sum_{\hat{A}' \in \hat{A}} R_{\hat{A}'} g_{\hat{A}'},$$

where $g_{\hat{A}'} \in \mathbb{R}$. Applying this to the formulation in (2.7) leads to the changed matrix equation

$$\sum_{B' \in A \setminus \hat{A}} a(R_{A'}, R_{B'}) u_{B'} = L(R_{A'}) - a(R_{A'}, g^h) \quad \forall A' \in A \setminus \hat{A}.$$

Solving this equation results in the degrees of freedom u for the inner basis functions. Hence the whole isogeometric solution together with the lifting g^h is given by

$$u^h(x) = \sum_{A' \in A \setminus \hat{A}} R_{A'}(x) u_{A'} + \sum_{\hat{A}' \in \hat{A}} R_{\hat{A}'} g_{\hat{A}'}.$$

2.2.4 A priori error

The a-priori error estimation gives an upper bound of the error $e_h = u - u_h$ before the approximated solution u_h is computed. The aim of this section is to get an error estimation for the isogeometric approach on elliptical problems similar to the FEM case.
For finite element polynomials the well-known a-priori error estimation is given by: Let $u \in \mathcal{V}$ be the exact solution of the weak formulation (2.3) and u^h its finite element approximation in $\mathcal{V}^h$ of degree p, then

$$\|u - u^h\|_{H^k(\Omega)} \leq C h^{p+1-k} |u|_{H^{p+1}(\Omega)} \quad \forall k = 0, 1, \ldots,$$

where $\|\cdot\|_{H^k(\Omega)}$ and $|\cdot|_{H^{p+1}(\Omega)}$ are the norms as well as the semi-norms, respectively, of the Sobolev spaces $H^k(\Omega)$ and $H^{p+1}(\Omega)$. A proof of this error estimation can be found in Ern and Guermond [2013] for example. The number h is a characteristic size of the mesh elements, which is often selected as the diameter. The constant C is independent of the solution u or of h. The rate of convergence is $p + 1 - k$, which means that successive bisection of the elements decreases the error by $(1/2)^{p+1-k}$ in the k-th Sobolev norm.
A similar but much more technical estimation for isogeometric analysis was done by Bazilevs et al. [2006]. Hence, one of the major difficulties is that NURBS have a large support, i.e. the degree +1 elements in each direction, which leads to the problem that the optimal values for the control variables cannot be determined by looking at each element individually. Another difficulty is the fact that the continuity of NURBS can vary on the element boundaries. This is handled by introducing so-called bent Sobolev spaces in which the continuity varies throughout the domain. The main result is the following theorem.

Theorem 2.8. *Let $k, l \in \mathbb{N}$ with $1 \leq k \leq l \leq p + 1$ and $u \in H^l(\Omega)$; then*

$$\sum_e |u - \Pi_k u|^2_{H^k(\Omega_e)} \leq C \sum_e h_e^{2(l-k)} \sum_{i=0}^{l} \|\nabla F\|^{2(i-l)}_{L^\infty(F^{-1}(\Omega_e))} |u|^2_{H^i(\Omega_e)}$$

The projection operator Π_k from $H^k(\Omega)$ to the finite dimensional solution space $\mathcal{V}^h$ defines the optimal interpolate

$$|u - \Pi_k u|_{H^k(\Omega_e)} \leq |u - v|_{H^k(\Omega_e)} \quad \forall v \in \mathcal{V}^h.$$

The constant C depends only on the shape of Ω and the degree p, but not on h. In the refinement techniques section we saw that the geometrical mapping F remains unchanged during h-refinement and thus the gradient ∇F does also. By assuming the isogeometric analysis solution u^h relates to the optimal interpolate $\Pi_k u$, the theorem proves the same convergence behavior as in the classical finite element approach. Hence, $u - u^h$ converges in the L^2 norm by the rate of $p + 1$ and in the H^1 semi-norm by the rate of p.

2.3 IGA on multiple patches

In order to generalize the isogeometric method on more complicated geometries like NURBS-based multiple patches we can use standard domain decomposition methods. We restrict ourselves to well-suited geometries, i.e. matching multiple patches with non-crossing boundaries. Under this restriction we use the Lagrange multiplier method. This section is mostly based on the publication of Magoulès and Roux [2006]. Consider an open and bounded domain Ω with boundary $\partial\Omega$, which is split into a set of m non-overlapping sub-domains Ω_i. The interface between the sub-domains is defined by

$$\Gamma = \bigcup_{ij} \Gamma_{ij}$$

with $\Gamma_{ij} = \partial\Omega_i \cap \partial\Omega_j$ and $i \neq j$. The boundary $\partial\Omega_i$ of every sub-domain is split into two parts, the external boundary

$$\Gamma_i^{\text{ext}} = \partial\Omega_i \cap \partial\Omega$$

and the interface of the sub-domains

$$\Gamma_i = \partial\Omega_i \cap \Gamma = \bigcup_{j} \Gamma_{ij}$$

We define the restriction to Ω_i of a scalar field u defined on Ω as u_i. Analogously, the restriction to Γ_i of a field u defined on Γ is denoted by u_i.

2.3.1 Lagrange multiplier method

We consider again the reference problem (2.1) with the related variational problem

$$J(v) = \frac{1}{2}\int_\Omega \nabla v \nabla v \,\mathrm{d}\Omega - \int_\Omega f v \,\mathrm{d}\Omega \longrightarrow \min_{v\in\mathcal{V}}, \tag{2.13}$$

where $\mathcal{V} \coloneqq \{\, v \in H^1(\Omega) \mid v\big|_{\partial\Omega} = 0 \,\}$. The global formulation can be expressed as a local one, where the functions u_i minimize the sum of the local energy functionals subjected to some interface boundary constraints

$$\sum_i J_i(v_i) = \sum_i \left(\frac{1}{2}\int_{\Omega_i} \nabla v_i \nabla v_i \,\mathrm{d}\Omega - \int_{\Omega_i} f_i v_i \,\mathrm{d}\Omega \right) \longrightarrow \min_{v\in\prod_i \mathcal{V}_i} \tag{2.14}$$

$$\text{subject to: } v_i - v_j = 0 \text{ on } \Gamma_{ij} \;\; \forall ij, \tag{2.15}$$

where $\mathcal{V}_i \coloneqq \{\, v_i \in H^1(\Omega_i) \mid v_i\big|_{\Gamma_i^{\text{ext}}} = 0 \,\}$. This can be done since the functions $u_i \in \mathcal{V}_i$ satisfying the continuity constraint (2.15) are the restriction in Ω_i of a function $u \in \mathcal{V}$, and Equation (2.13) is equal to (2.14).
In order to write the continuity constraint (2.15) in a variational form, the so-called Lions-Magenes space $H_{00}^{1/2}$ has to be introduced, which was first presented by Lions and Magenes [1968]. Let Ω be a domain with boundary $\partial\Omega$ and $\Gamma \subset \Omega$ is a subset. $H_{00}^{1/2}(\Gamma)$ is defined as the space of the restriction on Γ of the trace on $\partial\Omega$ of functions belonging to $H^1(\Omega)$ with zero value on $\partial\Omega \setminus \Gamma$. This can be written mathematically as $H_{00}^{1/2}(\Gamma) \coloneqq \{\, v\big|_\Gamma \mid v \in H^1(\Omega), \text{ with } v\big|_{\partial\Omega\setminus\Gamma} = 0 \,\}$. Hence, the functions u_i and u_j of the continuity condition (2.15) belong to $H_{00}^{1/2}(\Gamma_{ij})$

The restricted optimization problem (2.14),(2.15) can be solved using the Lagrange method, which leads to the equation

$$\lambda_{ij}(v_i - v_j) = \lambda_{ji}(v_j - v_i) = 0 \quad \forall \lambda_{ij} \in H^{-1/2}(\Gamma_{ij}), \tag{2.16}$$

where $\lambda_{ij} = -\lambda_{ji}$ are the Lagrange multipliers and $H^{-1/2}(\Gamma_{ij})$ is the dual space of $H_{00}^{1/2}(\Gamma_{ij})$. Typically, Equation (2.16) is tested only in the subspace L^2, due to representation reasons. Hence, together with Theorem 2.7, which proves the equivalence of the dual pair and the inner product of Hilbert spaces, follows

$$\int_{\Gamma_{ij}} (v_i - v_j)\lambda_{ij} \,\mathrm{d}S = \int_{\Gamma_{ji}} (v_j - v_i)\lambda_{ji} \,\mathrm{d}S = 0 \quad \forall \lambda_{ij} \in L^2(\Gamma_{ij}).$$

Brezzi and Fortin [1991] have shown that finding the solution of the restricted variational problem (2.4) is equivalent to finding the saddle point of the Lagrangian $\mathcal{L} : \prod_i \mathcal{V}_i \times L^2(\Gamma) \longrightarrow \mathbb{R}$ with

$$\mathcal{L}(v, \lambda) = \sum_i \left(\frac{1}{2} \int_{\Omega_i} \nabla v_i \nabla v_i \,\mathrm{d}\Omega - \int_{\Omega_i} f_i v_i \,\mathrm{d}\Omega \right) + \sum_i \left(\int_{\Gamma_i} v_i \lambda_i \,\mathrm{d}S \right). \tag{2.17}$$

The following hybrid system is the corresponding equation system derived from this saddle point problem:

$$\begin{aligned} &\int_{\Omega_i} \nabla u_i \nabla v_i \,\mathrm{d}\Omega = \int_{\Omega_i} f_i v_i \,\mathrm{d}\Omega + \int_{\Gamma_i} v_i \lambda_i \,\mathrm{d}S \quad \forall v_i \in \mathcal{V}_i \\ &\int_{\Gamma_{ij}} (u_i - u_j)\mu_{ij} \,\mathrm{d}S = 0 \quad \forall ij \; \forall \mu_{ij} \in L^2(\Gamma_{ij}). \end{aligned}$$

The existence and uniqueness of the solution of the hybrid system can be shown by analyzing the so-called Ladyženskaja-Babuška-Brezzi condition, which will be introduced subsequently in its discrete form, and the coercivity of this problem.

2.3.2 Discretized Lagrangian method

The Galerkin discretization of the previously described restricted variational problem (2.14),(2.15) leads to the discrete saddle point problem:
Find $(u^h, \lambda^h) \in \prod_i \mathcal{V}_i^h \times L^h$ such that

$$\mathcal{L}(u^h, \hat{\lambda}^h) \leq \mathcal{L}(u^h, \lambda^h) \leq \mathcal{L}(\hat{u}^h, \hat{\lambda}^h) \quad \forall (\hat{u}^h, \hat{\lambda}^h) \in \prod_i \mathcal{V}_i^h \times L^h,$$

where $\mathcal{V}_i^h \subset \mathcal{V}_i$ and $L^h \subset L^2(\Gamma)$ are finite dimensional subspaces. Similar to the infinite dimensional problem there exists a unique saddle point. However, the existence of this solution can be shown directly because of the special form of this problem. In general, the well-known discrete Ladyženskaja-Babuška-Brezzi condition (LBB condition) has to be fulfilled. This is a particular instance of the so-called discrete inf-sup condition which is necessary and sufficient for the well-posedness of discrete saddle point problems arising from discretization via Galerkin methods. Applying this to our saddle point problem the condition claims that there exists a constant $\alpha > 0$ such that

$$\inf_{\mu \in L^h} \sup_{v \in \prod_i \mathcal{V}_i^h} \frac{\frac{1}{2} \sum_{ij} \left[\int_{\Gamma_{ij}} (u_i^h - u_j^h)\mu_{ij} \,\mathrm{d}S \right]}{\|v\|_{\prod_i \mathcal{V}_i} \|\mu\|_{L^2(\Gamma)}} \geq \alpha.$$

Magoulès and Roux [2006] have shown that this condition is always satisfied in this problem setup. However, in other cases the standard theory on mixed and hybrid formulation recommend the choice of the discrete spaces such that the discrete Ladyženskaja-Babuška-Brezzi condition is uniformly fulfilled and that the constant α is independent upon the diameter h, in order to ensure the consistency of the discretization.
After ensuring the freedom of choice at constructing the finite dimensional subspaces for the Galerkin method, they can be represented as the span of the following basis functions $\mathcal{V}_i^h = \text{span}(\{\phi_1^i, ..., \phi_{N_i}^i\})$ and $L^h = \text{span}(\cup_{ij}\{\phi_1^{\lambda_{ij}}, ..., \phi_{N_{ij}^\lambda}^{\lambda_{ij}}\})$. Similar to the continuous problem, the hybrid system of equations can be solved instead of the discrete saddle point problem:

$$\int_{\Omega_i} \nabla u_i^h \nabla v_i \,\mathrm{d}\Omega = \int_{\Omega_i} f_i v_i \,\mathrm{d}\Omega + \int_{\Gamma_i} v_i \lambda_i^h \,\mathrm{d}S \quad \forall v_i \in \mathcal{V}_i^h$$
$$\int_{\Gamma_{ij}} (u_i^h - u_j^h)\mu_{ij}^h \,\mathrm{d}S = 0 \quad \forall ij \; \forall \mu_{ij}^h \in L_{ij}^h,$$

where $u_i^h \in \mathcal{V}_i^h$, $\lambda^h \in L^h$ and $L_{ij}^h = \text{span}(\{\phi_1^{\lambda_{ij}}, ..., \phi_{N_{ij}^\lambda}^{\lambda_{ij}}\})$.
The approach for the representation of the elements

$$u_i^h = \sum_{k=1}^{N_i} u_k^i \phi_k^i$$
$$\lambda_{ij}^h = \sum_{l=1}^{N_{ij}^\lambda} \lambda_l^{ij} \phi_l^{\lambda_{ij}}$$

yields the following matrix equation

$$\left[\begin{array}{ccc|c} K_1 & & & \\ & \ddots & & C \\ & & K_n & \\ \hline & C^\top & & 0 \end{array}\right] \begin{bmatrix} u_1 \\ \vdots \\ u_n \\ \lambda \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ 0 \end{bmatrix},$$

where K_i and f_i are the stiffness matrices and force vectors, respectively, for each single patch Ω_i. The degrees of freedom for each patch are represented by $u_i = (u_1^i, \ldots, u_{N_i}^i)^\top$. Additionally, the Lagrangian degrees of freedom are represented by λ.
The whole system of linear equations can be written shortly as

$$K_\lambda u_\lambda = f_\lambda.$$

The matrix C consists of the coupling block matrices

$$C_{ij} = \left[\int_{\Gamma_{ij}} \phi_k^i \phi_l^{\lambda_{ij}} \,\mathrm{d}S\right]_{k,l=1,1}^{N_i, N_{ij}^\lambda}$$

which are placed in the $i-$th block-row and the same columns as the degrees of freedom of λ_{ij}^h appear in the vector λ.
For assembling the coupling matrices C_{ij}, a total of $p+1$ quadrature points on each curve element is needed under the assumption of piecewise polynomial basis functions R with

degree p. The required amount of quadrature points results from the multiplicative form of the integrand, i.e.

$$\int_{\Gamma_{ij}} R_A R_B \, \mathrm{d}S.$$

Therefore, a polynomial of degree $2p$ has to be computed on each curve element. This leads to a required amount of $p+1$ quadrature points, which is even one quadrature point more than for assembling the isogeometric stiffness matrix.

2.3.3 Matrix equation

The stiffness matrices and force vectors can be constructed similar to the single patch case. The coupling matrices C_{ij}, however, require some more technical effort. In most real-world applications the adjacent patches of a multiple patch CAD file are not parameterized equally. In order to compute the integrals corresponding to the various coupling matrices, a method has to be derived that projects points from a given NURBS geometry back to its parameterization. In other words, this approach has to compute the inverse geometrical mapping F^{-1}. There exists a variety of methods for handling these kinds of problems. For differentiable problems, however, the most common approach is the Newton method. There are two major problems using the Newton method in this context. The first is a general problem of this method as the starting points have to be in a sufficient neighborhood of the solution. This condition determines whether the method converges or not and can be handled by a simple uniform pre-sampling of the search space. The sampling point with minimal deviation to the right-hand side under the evaluation with F can then serve as a starting point. The second is specific for B-spline or NURBS projection methods. In general, the Newton method provides a sequence converging to a critical point (the point with zero gradient). Therefore, this sequence can have elements outside the search space. For example if the parameter space of a B-spline curve is the standard interval $[0, 1]$ and the sequence limit is the point 0, some elements of the newton sequence can have negative values. This would cause errors in the computation of the next sequence element as it uses the value of the old element evaluated with the function F, which is not defined outside a special range. Therefore, in order to remain in the parameter space, the values outside have to be clamped into the valid domain. We write this algorithm as a pseudo code in

Algorithm 2.1 Inverse geometrical mapping

Computes the parameter vector ξ for a given point p on a NURBS geometry

Require: geometrical mapping F, point $p \in \mathbb{R}^d$ on geometry, $x^{(0)} \in \mathbb{R}^d$, $\epsilon > 0$

1: Set $g(x) = \|F(x) - p\|_2^2$
2: $k \leftarrow 0$
3: **while** $\left\|\nabla g(x^{(k)})\right\| > \epsilon$ **do**
4: Solve

$$\mathrm{H}_g\left(x^{(k)}\right) q^{(k)} = -\nabla g\left(x^{(k)}\right)$$

5: $q^{(k)} \leftarrow \max\{\xi_{p+1}, \min\{q^{(k)}, \xi_{n+1}\}\}$ ▷ element-wise
6: $x^{(k+1)} \leftarrow x^{(k)} + q^{(k)}$
7: $k \leftarrow k + 1$
8: **end while**
return $\xi \leftarrow x^{(k)}$

Even though the algorithm can be applied to curves as well as to surfaces, only the curve case is needed for assembling the coupling matrices. The integrals

$$\int_{\Gamma_{ij}} \phi_k^i \phi_l^{\lambda_{ij}} \, \mathrm{d}S$$

from the coupling matrices C_{ij} can be computed using the Algorithm 2.1 with the standard Gauss-Legendre quadrature rule. First, the quadrature points are mapped to the coupling boundary Γ_{ij}. Their parameterization derived from the two different geometrical mappings are then computed with the previous described Algorithm 2.1. In fact, the finite dimensional subspace L_{ij}^h would be the span of the NURBS basis functions of the isoparametric curve of one of the two patches. Therefore, only one inverse mapping has to be computed.
That means the basis functions $\phi_k^{\lambda_{ij}}$ of L_{ij}^h are either elements of the basis of $\mathcal{V}_i^h$ or of $\mathcal{V}_j^h$.

3 Results

In the following we will consider elliptic problems with varying Dirichlet boundary conditions. We begin by solving a problem on a single patch, followed by solving problems in the multiple patch setup. In the end we will solve a prototype example in the area of CFD (Computational Fluid Dynamics).
The implementation is done in the programming language Python. The vectorization of the code is done using NumPy, which is a numerical linear algebra package for implementing optimized and efficient code developed by Walt et al. [2011]. The spline routines are based mostly on a DLR explicit developed spline library. This library includes vectorized implementations of the most important B-spline and NURBS algorithms, such as the De Boor algorithm of Section 2.1.1 for fast and numerically stable evaluation of the B-splines, functions to compute the B-spline basis matrices, knot insertion or degree elevation. The figures are generated by using Matplotlib, which is a python package for generating plots, developed by Hunter [2007].

3.1 Single patch

In a first experiment we consider the first quarter of the standard annulus $\Omega \coloneqq \{\, (x,y) \in \mathbb{R}^2 \mid x^2+y^2 \in [1,4],\ x,y \geq 0 \,\}$ as a NURBS-based single patch. This can be constructed by a $(2,1)$-degree patch with the following parameters:

$$\begin{aligned} P &= \{(1,0),(1,1),(0,1),(2,0),(2,2),(0,2)\} \\ \Xi &= \{0,0,0,1,1,1\} \\ \Theta &= \{0,0,1,1\} \\ w &= \{1,\frac{1}{\sqrt{2}},1,1,\frac{1}{\sqrt{2}},1\} \end{aligned}$$

The physical space as well as the control net is illustrated in Figure 9.
Again, we consider the homogeneous Poisson problem with a predefined function f:
Find $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\triangle u &= f \quad \text{in} \quad \Omega \\ u &= 0 \quad \text{on} \quad \partial\Omega \end{aligned} \tag{3.1}$$

with

$$f(x,y) = \frac{8-9\cdot\sqrt{x^2+y^2}}{x^2+y^2} \cdot \sin\left(2\cdot\arctan\left(\frac{y}{x}\right)\right).$$

The analytic solution of (3.1) is given by

$$u(x,y) = (x^2+y^2-3\sqrt{x^2+y^2}+2)\cdot\sin\left(2\cdot\arctan\left(\frac{y}{x}\right)\right), \tag{3.2}$$

which is visualized in Figure 10 with the contours ranging from 0 to -0.3.
We perform the integration by using the standard Gauss-Legendre quadrature rule with p quadrature points in each direction on every element, where p is the degree of the NURBS basis functions. Since the isogeometric stiffness matrix is symmetric and positive-definite, the system of linear equation is solved by the iterative conjugate gradient method (CG).

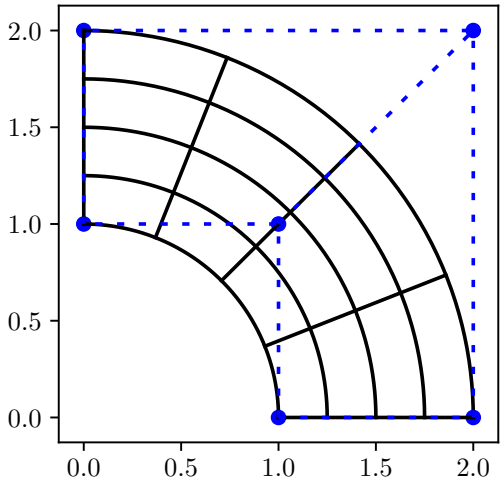

Figure 9: NURBS representation of Ω

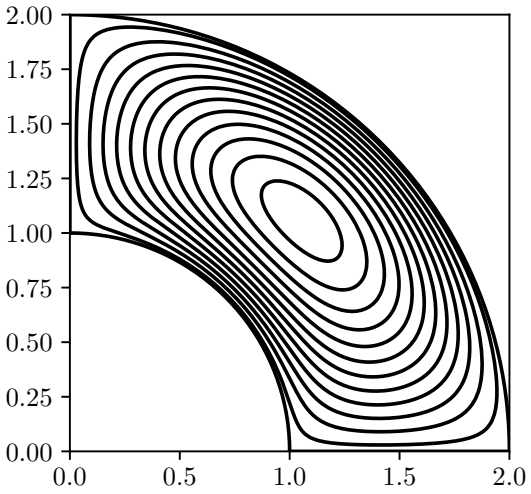

Figure 10: Analytical solution (3.2) on Ω

In order to analyze the convergence behavior of the isogeometric approach on single patches we compute the L^2 norm, as well as, the H^1 semi-norm of the differences to the closed-form solution. The norms are defined continuously by

$$\|u - u^h\|_{L^2(\Omega)} = \left(\int_\Omega (u - u^h)^2 \, \mathrm{d}\Omega \right)^{1/2}$$
$$|u - u^h|_{H^1(\Omega)} = \left(\int_\Omega \left(\nabla (u - u^h) \right)^2 \mathrm{d}\Omega \right)^{1/2}.$$

For simplicity, we calculate these integrals with the trapezoidal rule, which in fact only results in discrete norms. However, in the following we will use the discrete norm and the continuous norm equivalently. The quadratic single patch was generated by applying the degree elevation or p-refinement approach to the linear parametric direction of the surface Ω. The characteristic size of the elements h is chosen to be the square root of the surface area of the elements. The a-priori error estimation of Section 2.2.4 claims that a NURBS-based single patch with degree p should have a rate of convergence of $p+1$ in the L^2 norm and a rate of p in the H^1 semi-norm. In fact, the two convergence plots in Figure 11 and Figure 12 propagate these convergence rates. In particular, the error of the linear NURBS geometry in the L^2 norm has a rate of convergence of 2 and in the H^1 semi-norm of 1. Additionally, the error of the quadratic patch has a rate of convergence of 3 in the L^2 norm and in the H^1 semi-norm a rate of 2. The sufficiency of p quadrature points in each parametric direction, mentioned in Section 2.2.3 for almost affine geometries is also supported by these experiments as they have shown the expected rate of convergence in the several norms by using p quadrature points but have had no convergence behavior at all by using less than this amount of quadrature points.
Additionally, the condition $\kappa(K)$ of the stiffness matrix K is of major importance for the performance of iterative solvers, especially for the used CG method. The condition is defined as $\kappa(K) = \|K\|\|K^{-1}\|$ with an arbitrary matrix norm $\|\cdot\|$. In our case, we use the spectral norm $\|K\|_2 = \max_{\|x\|_2=1} \|Kx\|_2$. Then, the condition number can be rewritten for

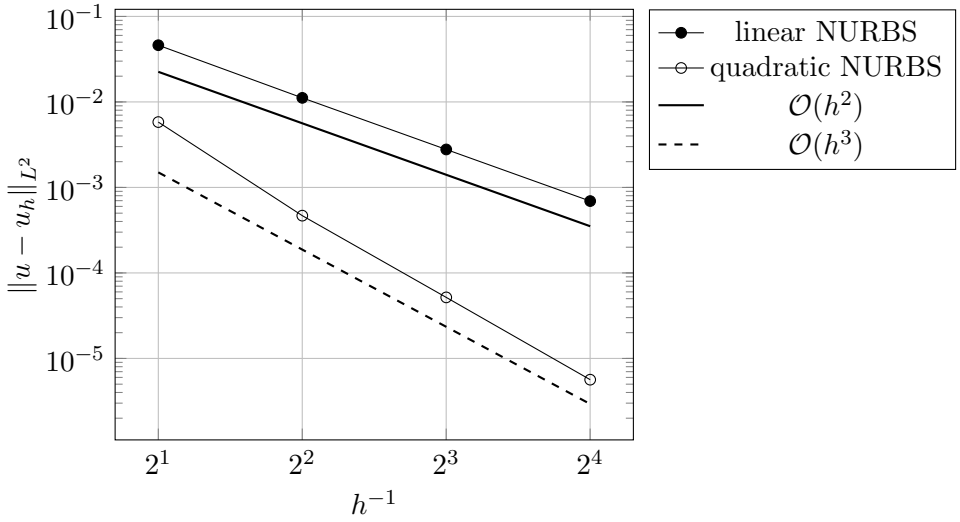

Figure 11: Convergence in the discrete L^2 norm for different degrees

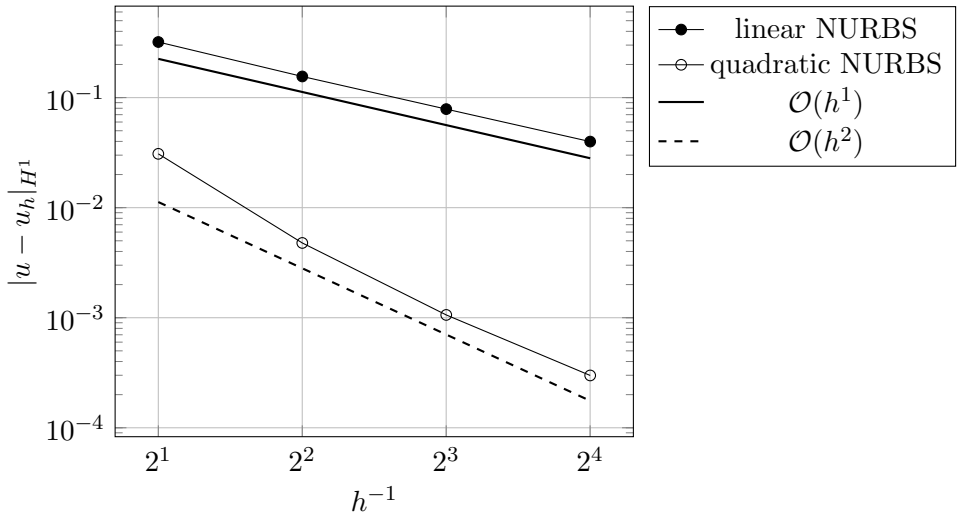

Figure 12: Convergence in the discrete H^1 semi-norm for different degrees

normal matrices as

$$\kappa_2(K) = \left| \frac{\lambda_{\max}(K)}{\lambda_{\min}(K)} \right|,$$

where $\lambda_{\max}(K)$ and $\lambda_{\min}(K)$ are the maximal and minimal eigenvalues of K, respectively. A matrix with a high condition number is called ill-conditioned, while we call it well-conditioned otherwise. In addition, the performance of the iterative solver decreases with increasing condition numbers.

Table 1 shows the condition number of the stiffness matrix for the reference problem. Note that the geometry is only h-refined in one parametric direction which leads to smaller condition numbers. Hence, the subsequent explained behavior can be seen better. The condition numbers are generated for h-refined geometries, where h^{-1} ranges from 2 to 128 for different degrees p. In classical FEM, the condition number of the stiffness matrix is of

Table 1: Condition number of the stiffness matrix

p \ h^{-1}	2	4	8	16	32	64	128
2	2.92	2.00	1.83	6.30	24.73	99.47	401.04
3	26.59	20.07	18.55	21.57	53.23	212.26	849.36
4	290.52	204.83	221.44	299.27	341.65	566.64	2265.88

order h^{-2}. In isogeometric analysis, for higher p the condition number does not appear to be of order h^{-2} for coarse mesh-sizes. Gahalaut and Tomar [2012] show that this is due to the stability of B-splines. The condition number of B-splines heavily depends on polynomial degree, and scales as $(p2^p)^2$. This factor dominates the factor h^{-2} for coarse meshes. However, Table 1 shows that the condition number is asymptotically of order h^{-2}. This can be seen for $p = 2$ starting in $h^{-1} = 16$, for $p = 3$ and $p = 4$ starting at $h^{-1} = 32$ and $h^{-1} = 64$, respectively.

3.2 Multiple patch

The same problem as known from the single patch section will also serve as a reference problem in the multiple patch setup. In contrast to the single patch section, the geometry $\Omega := \{ (x, y) \in \mathbb{R}^2 \mid x^2 + y^2 \in [1, 4],\ x, y \geq 0 \}$ is constructed as a multiple patch. This can be done by first constructing the NURBS single patch of the previous section. After that it is split into two matching patches by inserting a knot of multiplicity p in one parametric direction. This leads to an interpolating control point at the same parametric value which can be then used for constructing two matching patches.

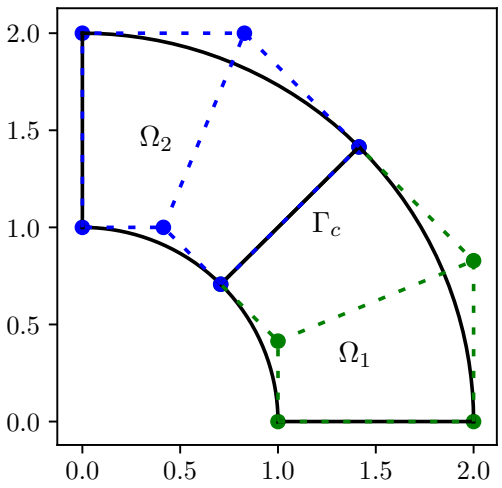

Figure 13: Decomposed geometry Ω

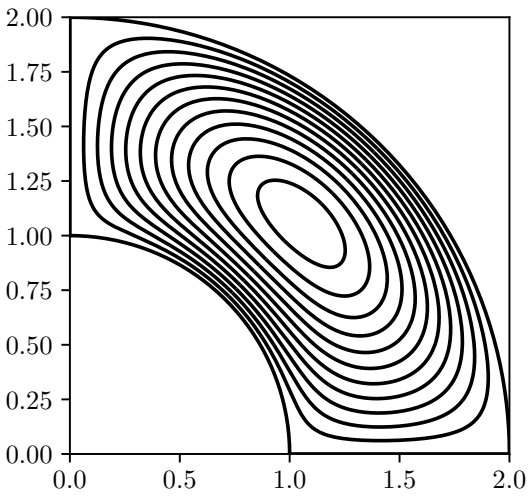

Figure 14: IGA solution on decomposed Ω

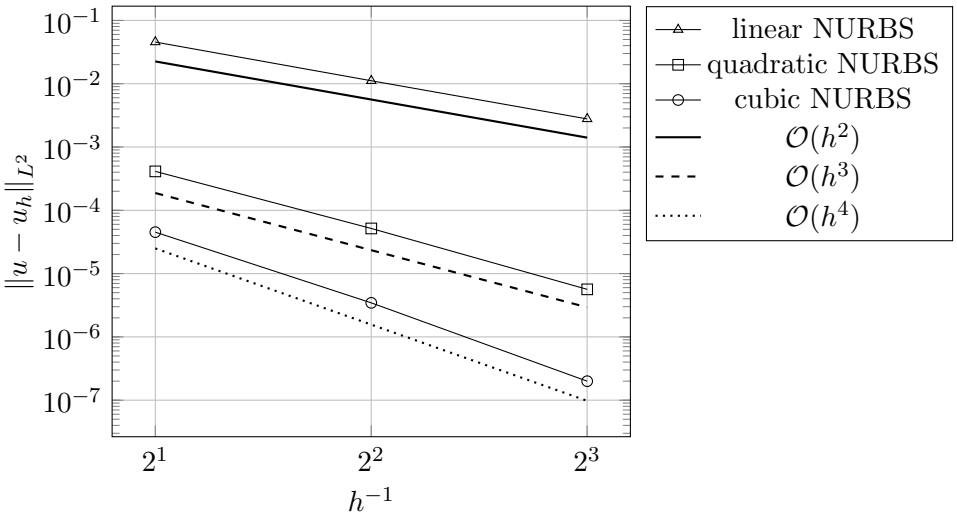

Figure 15: Convergence in the discrete L^2 norm of multiple patch IGA

Figure 13 illustrates the decomposition of the quarter annulus Ω into two patches Ω_1 and Ω_2 as well as their coupling boundary Γ_c. Hence, a knot with a double multiplicity is inserted in the quadratic parametric direction. For constructing the coupling matrices C_{12} and C_{21} a total of $p+1$ quadrature points on each curve element is required as explained in Section 2.3.2.

Since the linear problem corresponding to the multiple patch setup is a saddle point problem, the underlying left-hand side matrix is indefinite. This leads to a restriction in the choice of the solver. In this case, the minres method was chosen. For a detailed analysis of this method in a saddle point problem context, we refer to Benzi et al. [2005].

Figure 14 shows the isogeometric solution of the multiple patch problem. The patches are both quadratic but have different parameterizations. The patch Ω_1 has 4 elements whereas the second patch Ω_2 has a total of 16 elements. We can see that the difference of the two solutions u_1^h and u_2^h is almost zero at the coupling boundary. In particular, the average difference is $1e^{-16}$. The rates of convergence are also analyzed for the multiple patch problem. Figure 15 shows the convergence in the discrete L^2 norm for different degrees ranging from $p = 1$ to $p = 3$. Both patches are equally h-refined ranging from 2^2 to 8^2 elements on each patch. It turns out that this multiple patch problem can be solved preserving the same rate of convergence as in the single patch setup. That means the isogeometric solution u^h converges with a rate of $p+1$ in the L^2 norm for a NURBS geometry of degree p.

Again, the condition number of the left-hand side matrix has a major impact on the convergence behavior of the iterative solver. The corresponding matrix of this problem can be reformulated using standard Gauss elimination:

$$\left[\begin{array}{cc|c} K_1 & & C_{12} \\ & K_2 & C_{21} \\ \hline C_{12}^\top & C_{21}^\top & 0 \end{array}\right] \Leftrightarrow \left[\begin{array}{cc|c} K_1 & & C_{12} \\ & K_2 & C_{21} \\ \hline & 0 & -C_{12}^\top K_1^{-1} C_{12} - C_{21}^\top K_2^{-1} C_{21} \end{array}\right]. \tag{3.3}$$

Since the condition number κ_2 depends on the biggest and smallest eigenvalues of a matrix, the condition number of the multiple patch IGA matrix can only be equal or worse,

Table 2: Condition number of the multiple patch IGA matrix

p \ h^{-1}	2	4	8	16	32
2	81.07	372.90	1698.95	7717.41	34523.49
3	529.13	1786.16	6811.00	28829.57	125683.36

compared to the condition numbers of the several stiffness matrices. This follows directly from (3.3) as the eigenvalues of K_1 and K_2 are also eigenvalues of the whole matrix. Hence, the condition number of the multiple patch matrix should scale at least as the condition number of the stiffness matrix in the single patch setup under h-refinement.
Table 2 supports this statement asymptotically since the condition number scales more than h^{-2} for a sufficient h-refinement. This behavior can bee seen for the quadratic case starting at $h^{-1} = 4$ as well as for the cubic case starting at $h^{-1} = 8$.

3.3 Aerospace application

The reference problem for the aerospace application is a potential flow simulation around a two dimensional airfoil. The underlying geometry is a B-spline approximation of a NACA (National Advisory Committee for Aeronautics) 4 digit airfoil profile. The 4 digits designate the camber, position of the maximum camber and thickness of the airfoil. For example the NACA MPXX airfoil number implies the following specifications:

- M is the maximum camber divided by 100.
- P is the position of the maximum camber divided by 10.
- XX is the thickness divided by 100.

In the following the NACA 6412 airfoil is used as a reference. It is generated with the method described by Ladson et al. [1996]. This approach provides a set of discretized points of a 2 dimensional airfoil profile. These points are then approximated with a basic B-spline approximation algorithm which can be found in De Boor [1978]. In order to get a suitable geometry for isogeometric simulations, a rectangle is build around the resulting airfoil B-spline curve. After that, a multiple patch is generated by combining the curves of the rectangle and curves of the decomposed airfoil profile to produce the surfaces. By setting the weights to 1 we obtain the desired multiple patch for the potential flow simulation. This surface is visualized in Figure 16, where Ω_i denote the various patches.
The interfaces of the patches previously denoted by Γ_i are now named Γ_{C_i} in Figure 16 and are called coupling boundaries. Additionally, we can summarize the boundaries of Figure 16 into three different types. First, the already mentioned coupling boundaries with their union denoted by Γ_C. Second, the also known Dirichlet boundaries Γ_{D_i} summarized by Γ_D. And lastly the so-called Neumann boundary Γ_N, where $\Gamma_N = \cup_i \Gamma_{N_i}$.
In classical fluid dynamics of a potential flow, the velocity v is the gradient of the potential u. Therefore, we can write $v = \nabla u$. With standard vector calculus we know that the curl of the potential flow equals zero due to the fact that curls of gradients of C^2 functions are zero. Hence, v describes an irrotational flow. Furthermore, we assume the potential flow to be incompressible. This behavior can be found in fluids with a low Mach number,

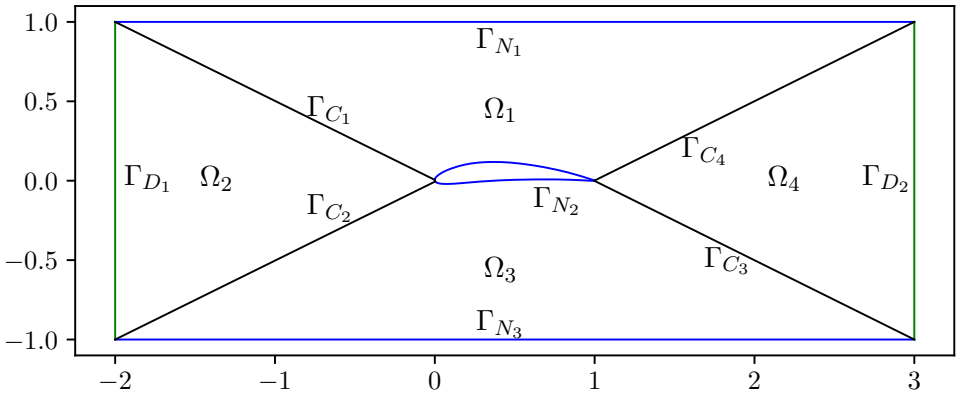

Figure 16: Initial multiple patch of NACA airfoil

for example. This means the velocity field v has zero divergence $\text{div}(v) = \nabla \cdot v = 0$. This forces the potential u to satisfy the Laplace equation $\Delta u = \nabla \cdot \nabla u = 0$.
Therefore, we can simulate the potential by solving the following strong formulation:
Find $u : \Omega \longrightarrow \mathbb{R}$ such that

$$\begin{aligned} -\triangle u &= 0 \quad \text{in } \Omega = \cup_i \Omega_i \\ u &= g \quad \text{on } \Gamma_D \\ \frac{\partial u}{\partial \nu} &= 0 \quad \text{on } \Gamma_N, \end{aligned} \tag{3.4}$$

where $g : \Gamma_D \longrightarrow \mathbb{R}$ and with continuity on the coupling boundary Γ_C in the same manner we assumed in the multiple patch section.
Previously, we showed that the variational formulation of this problem has a unique solution subject to the existence of the Lagrangian multipliers. This follows directly from the fact that the condition $\frac{\partial u}{\partial \nu} = 0$ on Γ_N has no effect on the weak formulation. We assume to have a uniform onflow. Hence, the function g defined on the Dirichlet boundary can be written as

$$g(x) = \begin{cases} c_1, & \text{if } x \in \Gamma_{D_1} \\ c_2, & \text{if } x \in \Gamma_{D_2} \end{cases},$$

where c_1 and c_2 are positive constants. As a reference we chose c_1 to be 0 and c_2 to be 0.2. The Neumann boundary conditions can be interpreted as a condition for the contour lines of the solution that forces them to be normal to the Neumann boundaries. Figure 17 visualizes the isogeometric solution to the specific Laplace problem (3.4) by plotting the contour lines of the solution together with the NACA airfoil. The solution was generated with a refinement of 8 times 8 elements on each patch. In this case the mean error of $u_i - u_j$ on each coupling boundary is limited by 1e-16.
Particularly, the effect of the Neumann condition is illustrated well as the contour lines cross the Neumann boundaries perpendicularly.
The potential flow can be computed by calculating the gradient of the isogeometric approximation of the velocity potential. Recalling the representation of the isogeometric solution from the matrix equation section yields the following equation: Let u_h^i be the solution on the single patch Ω_i, then

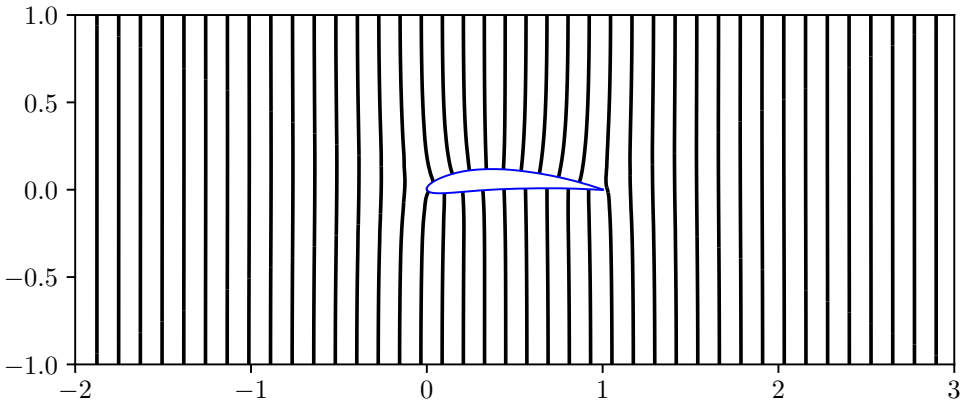

Figure 17: Potential lines

$$\begin{aligned} u_h^i(x) &= \sum_{A' \in A \setminus \hat{A}} R_{A'}^i(x) u_{A'}^i + \sum_{\hat{A}' \in \hat{A}} R_{\hat{A}'}^i g_{\hat{A}'}^i \\ &= \sum_{A' \in A} R_{A'}^i(x) \tilde{u}_{A'}^i, \end{aligned}$$

where $\tilde{u}^i$ denotes the combined vector of $u_{A'}^i$ and $g_{\hat{A}'}^i$. Additionally, $R_{A'}^i$ denote the corresponding NURBS basis functions of patch Ω_i. Together with the chain we have

$$\nabla u_h^i(x) = \sum_{A' \in A} (DF^{-1})^\top \nabla \hat{R}_{A'}^i \tilde{u}_{A'}^i$$

In summary, the whole potential flow derived from the reference multiple patch problem can be visualized by plotting $\nabla u_h^1(x), \ldots, \nabla u_h^4(x)$ together. Figure 18 illustrates the streamlines derived from the vector field given by the potential flow. These streamlines can be generated by the following method: First, a discrete and equidistant solution grid in the physical space of the potential flow has to be generated. In a second step, using a variant of Algorithm 2.1, the grid points have to be back projected to the parametric

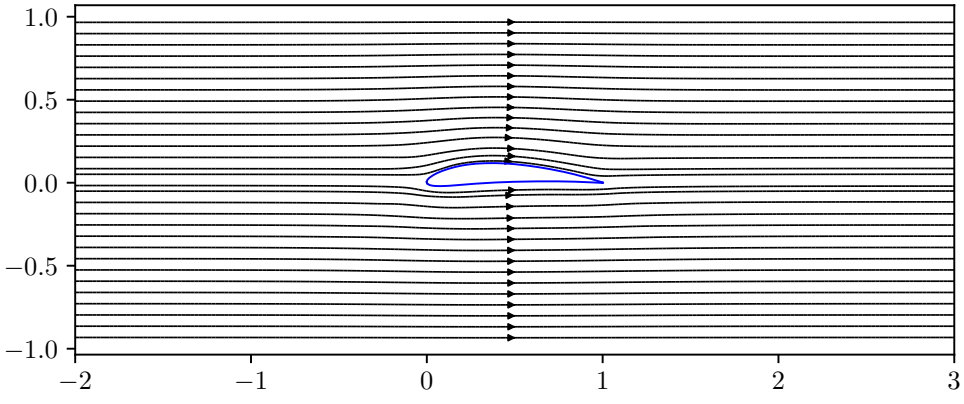

Figure 18: Potential flow simulation

space to evaluate the gradients.
After generating the equidistant grid on the physical space, the streamlines can be computed by using the streamline method of the matplotlib package. These streamlines are computed by solving the following ordinary differential equation:

$$\mathrm{d}x = \nabla u(x,t), \tag{3.5}$$

where $x(t) = (x_1(t), x_2(t))^\top$ is the resulting parameterized path line. The ODE can be solved by standard Runge-Kutta methods. Figure 18 illustrates the potential flow as streamlines generated with the previous described matplotlib method. It shows the motion of a non-rotational and incompressible fluid around an airfoil.

4 Discussion and conclusion

In this thesis we presented isogeometric methods for solving partial differential equations on single patches as well as on multiple patches. The single patch case is the standard approach, which was first suggested by Hughes et al. [2005]. We explained in detail how to generate the matrix equation by using standard NURBS and B-spline algorithms. Furthermore, we provided an a-priori error estimation for the h-refined geometry which we then compared to the standard finite element a-priori error.
Our results showed that the rate of convergence for a single patch problem is the same as in the FEM. One major advantage of the isogeometric approach compared to finite element methods is the non-necessity of meshing the geometries initially. Additionally, the h-refinement approach is almost trivial as it turns out to be equivalent to a simple knot insertion. This is in contrast to the standard FEM where local mesh refinement can require major computing efforts. On the other hand assembling the isogeometric stiffness matrix requires a high integration effort and has only in the best case (depending on the underlying geometry) the same integration complexity as the classical FEM. In particular, the complexity is comparable if and only if the isogeometric mesh is almost affine. However, this would be an uninteresting case for isogeometric analysis, since this approach has its advantages over FEM only for more complicated geometries. Additionally, the stiffness matrix in IGA is much denser than the ones in classical FEM since NURBS and B-spline basis functions have a large support. In addition, the condition number of the stiffness matrix in the isogeometric context increases exponentially with increasing polynomial degree. In summary, one has to decide whether to use the isogeometric approach or the standard FEM depending on the underlying problem. More precisely, the benefit of the non-necessity of the initial meshing process has to be balanced against a worse integration complexity for assembling the stiffness matrix as well as against a worse performance of the iterative solver especially when using higher degrees.
In common real-world applications however, the geometries are given as CAD files. Most of them are more complicated than standard single patches. Hence, the main focus of this thesis was on multiple patches in order to show the applicability of the method to real-world problems. The coupling of the several patches of a multiple patch geometry was done using Lagrange multipliers. However, there exists a variety of other methods for coupling sub-domains in order to solve partial differential equations globally on compounded geometries. The decision for using the Lagrange method was motivated by its exactness, when compared to standard penalty methods, and its simplicity, when compared to some augmented or Nitsche-type approaches. For an analysis of the latter in an isogeometric context, we refer to Apostolatos et al. [2014]. One major benefit of the IGA on multiple patches is its applicability to more complicated and common CAD files. Another big advantage is the possibility of local refinement. This is not possible on single patches as NURBS or B-spline surfaces have a tensoric structure, which allows only global refinements. This can be very useful, for example if the rough behavior of the equation is known a-priori. The geometry could be decomposed to a multiple patch geometry such that the parts of most interest form a single patch. The presented method then allows the local h-refinement of these parts. One disadvantage of the presented multiple patch IGA, however, is that it results in a saddle point problem and therefore in a indefinite left-hand side matrix. This leads to a restriction in the choice of the iterative solver, since most methods are only applicable to positive definite matrices. Another disadvantage of multiple patch IGA is that the integration complexity for the coupling matrices is very high, in particular even higher than for assembling the stiffness matrix. Last but not least, the disadvantages

regarding the stiffness matrix of the single patch setup are also valid in the multiple patch case.
The usability of isogeometric analysis in the context of modern aerospace research was also addressed in this thesis. One major application is the analysis of aerodynamic properties of an aircraft which is based on the simulation of fluid dynamics. Since this would lead to solving a hyperbolic equation on a three dimensional geometry, the presented problem of solving an elliptic equation on a two dimensional domain is only a first but very important step to tackle real-world CFD tasks. Note that the issues appearing in our prototype example will appear also in more complicated simulations. One major problem is that only well-suited geometries can be handled. For example, if a CAD file is a non-matching multiple patch, the method cannot be used without manipulating and rearranging the geometry before. This results in major pre-processing efforts. Hence, it takes a lot of work and effort to apply IGA robustly to complicated geometries. Together with the previously described disadvantages on general geometries, this makes IGA less interesting for applications to real-world problems.
Although we investigated many interesting aspects of isogeometric analysis in this thesis, there are various possibilities for future research. First, we could parallelize the presented isogeometric method on multiple patches. Magoulès and Roux [2006] have shown that this method is indeed very well suited for parallel implementation on distributed memory MIMD (multiple instruction, multiple data) machines with message passing programming environment, where each patch is allocated to one processor. Then, each processor can assemble the local stiffness matrix as well as the local right-hand side associated with its patch. The non-local operations are related to the assembly of the coupling matrices, which can be treated by lists exchanging the local solutions in order to compute the jumps on the interfaces. These lists are similar to the standard description of boundary conditions in FEM implementations. Hence, the presented method could be implemented efficiently on supercomputers. Second, we could develop an isogeometric method that is able to handle non-matching patches. The coupling of these non-matching patches results in projection methods as there exists no coupling boundary anymore. And lastly, we could use another geometrical basis for the isogeometric methods such as T-splines, which were invented by Sederberg et al. [2003]. The major benefit of T-spline geometries is that they can be refined locally, which makes them very interesting for IGA. Their basic behavior in the context of IGA has already been analyzed by Bazilevs et al. [2010].
In summary, the advantage of not generating a mesh initially is compensated by the higher integration complexity and the higher efforts for the iterative solver. Additionally, using higher degrees leads to a bad conditioning of the stiffness matrix. This leads to a restriction in choosing a suitable and robust iterative solver for large problem sizes. The indefiniteness of the left-hand side matrix arising in the multiple patch setup together with the even higher integration complexity for assembling the coupling matrices present a challenge for applying IGA to more complicated problems in modern aerospace research.

References

H.W. Alt. *Linear functional analysis*. Springer, 1992.

Andreas Apostolatos, Robert Schmidt, Roland Wüchner, and Kai-Uwe Bletzinger. A Nitsche-type formulation and comparison of the most common domain decomposition methods in isogeometric analysis. *International Journal for Numerical Methods in Engineering*, 97(7):473–504, 2014.

Yuri Bazilevs, L Beirao da Veiga, J Austin Cottrell, Thomas JR Hughes, and Giancarlo Sangalli. Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07):1031–1090, 2006.

Yuri Bazilevs, Victor M Calo, John A Cottrell, John A Evans, Thomas Jr R Hughes, S Lipton, Michael A Scott, and Thomas W Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):229–263, 2010.

Michele Benzi, Gene H Golub, and Jörg Liesen. Numerical solution of saddle point problems. *Acta numerica*, 14:1–137, 2005.

Wolfgang Boehm. Inserting new knots into B-spline curves. *Computer-Aided Design*, 12 (4):199–201, 1980. DOI 10.1016/0010-4485(80)90154-2.

Franco Brezzi and Michel Fortin. *Mixed and hybrid finite element methods*. Springer-Verlag New York, Inc., 1991.

Carl De Boor. On calculating with B-splines. *Journal of Approximation theory*, 6(1):50–62, 1972. DOI 10.1016/0021-9045(72)90080-9.

Carl De Boor. *A practical guide to splines*, volume 27. Springer-Verlag New York, 1978.

Alexandre Ern and Jean-Luc Guermond. *Theory and practice of finite elements*, volume 159. Springer Science & Business Media, 2013.

Rida T Farouki and V T Rajan. Algorithms for polynomials in bernstein form. *Computer Aided Geometric Design*, 5(1):1–26, 1988.

Krishan P.S. Gahalaut and Satyendra Tomar. Condition number estimates for matrices arising in the isogeometric discretizations. *RICAM report*, 23(2012):1–38, 2012.

Thomas JR Hughes, John A Cottrell, and Yuri Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39-41):4135–4195, 2005.

Thomas JR Hughes, Alessandro Reali, and Giancarlo Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer methods in applied mechanics and engineering*, 199(5-8):301–313, 2010.

J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

Charles L Ladson, Cuyler W Brooks Jr, Acquilla S Hill, and Darrell W Sproles. Computer program to obtain ordinates for NACA airfoils. 1996.

Jacques-Louis Lions and Enrico Magenes. Problèmes aux limites non homogènes et applications. volume i. 1968.

Frédéric Magoulès and François-Xavier Roux. Lagrangian formulation of domain decomposition methods: A unified theory. *Applied Mathematical Modelling*, 30(7):593–615, 2006.

Les Piegl and Wayne Tiller. A menagerie of rational B-spline circles. *IEEE Computer Graphics and Applications*, 9(5):48–56, 1989.

Les Piegl and Wayne Tiller. Software-engineering approach to degree elevation of B-spline curves. *Computer-Aided Design*, 26(1):17–28, 1994.

Les Piegl and Wayne Tiller. *The NURBS book*. Springer Science & Business Media, 2012.

Hartmut Prautzsch, Wolfgang Boehm, and Marco Paluszny. *Bézier and B-spline techniques*. Springer Science & Business Media, 2013.

Thomas W Sederberg, Jianmin Zheng, Almaz Bakenov, and Ahmad Nasri. T-splines and T-NURCCs. In *ACM transactions on graphics (TOG)*, volume 22, pages 477–484. ACM, 2003.

Wayne Tiller. Knot-removal algorithms for NURBS curves and surfaces. *Computer-Aided Design*, 24(8):445–453, 1992.

Stéfan van der Walt, S. Chris Colbert, and Gael Varoquaux. The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011. DOI 10.1109/MCSE.2011.37.