A Framework for Shape Optimization in the Context of Isogeometric Analysis

D. Fußeder[a,*], B. Simeon[a], A.-V. Vuong[a]

[a] *Department of Mathematics, Technische Universität Kaiserslautern, Paul-Ehrlich-Straße 31, 67663 Kaiserslautern, Germany*

Abstract

We develop a framework for shape optimization problems under state equation constraints where both state and control are discretized by B-splines or NURBS. In other words, we use isogeometric analysis (IGA) for solving the partial differential equation and a nodal approach to change domains where control points take the place of nodes and where thus a quite general class of functions for representing optimal shapes and their boundaries becomes available. The minimization problem is solved by a gradient descent method where the shape gradient will be defined in isogeometric terms. This gradient is obtained following two schemes, *optimize first–discretize then* and, reversely, *discretize first–optimize then*. We show that for isogeometric analysis, the two schemes yield the same discrete system. Moreover, we also formulate shape optimization with respect to NURBS in the optimize first ansatz which amounts to finding optimal control points and weights simultaneously. Numerical tests illustrate the theory.

Keywords: isogeometric analysis, shape optimization, adjoint approach, weight optimization, NURBS

1. Introduction

Isogeometric analysis (IGA) combines the fundamental idea of the finite element method (FEM) with spline techniques from computer aided geometric design for a common description of the domain and the Galerkin projection [1]. IGA aims to overcome the bottleneck of converting design-suitable descriptions to FEM-suitable models, and it holds particularly great promise in the field of shape optimization where the frequent conversion between geometry description and computational mesh is cumbersome and error-prone.

In shape optimization, a nodal approach to find the optimal shape, i.e., using a piecewise linear interpolation of the domain's boundary, is mostly avoided because of regularity issues. One common way out is to parameterize the boundary by B-splines

*Corresponding author. Phone: +49 631 205 5318

Email addresses: fusseder@mathematik.uni-kl.de (D. Fußeder), simeon@mathematik.uni-kl.de (B. Simeon), vuong@mathematik.uni-kl.de (A.-V. Vuong)

[2, 3], and obviously, the approximation or exact representation of a domain by means of B-splines and NURBS is a better choice than using a space of polygons.

Therefor, the combination of shape optimization with isogeometric analysis seems very favorable because all occurring approximation spaces can be covered by one common description, namely B-splines or NURBS, and there is the benefit of arbitrary regularity of boundary interpolation. However, in IGA not only the boundary is parameterized in this way but also the inside, leading to new options in shape optimization methods. The combination of IGA and shape optimization has already been investigated in a number of papers such as [4, 5] with application to electromagnetism and [6, 7, 8, 9] with application to solid mechanics and also shells [10].

It is the objective of this paper to introduce a general framework that clarifies certain aspects and sheds new light on the solution of shape optimization problems by means of IGA. In particular, we discuss the two different approaches *discretize first–optimize then* vs. *optimize first–discretize then* with gradient-based shape optimization and show that the order of optimization and discretization commutes for shape optimization in IGA. Though being a common statement in optimal control theory, this equivalence of the two approaches has certain restrictions and specific consequences.

Optimization with partial differential equations as state constraints is an active research area with interconnections to functional analysis and various other fields. Reaching out to a broader audience in the engineering community, our exposition here tries to compromise between a rigorous mathematical treatment and a more informal discussion of the subject that highlights the main ideas and concepts. Throughout the paper, we assume a given cost functional

$$J(u,\Omega,\Gamma) := \int_\Omega j_1(u,x)\,\mathrm{d}x + \int_\Gamma j_2(u,s)\,\mathrm{d}s \tag{1}$$

for domains $\Omega \subset \mathbb{R}^d$, $d = 2,3$, with moving boundary $\Gamma \subset \partial\Omega$, and a general shape optimization problem

$$\min J(u,\Omega,\Gamma) \quad \text{s.t. } E(u,\Omega) = 0 \tag{2}$$

with a state equation

$$E(u,\Omega) = 0 \tag{3}$$

representing a second order linear elliptic partial differential equation with solution $u := u(\Omega)$.

At an optimal shape Ω^*, formal differentiation of (2) w.r.t. Ω yields the necessary optimality condition

$$\mathrm{d}_\Omega\, J(u^*,\Omega^*,\Gamma^*) = 0. \tag{4}$$

The crucial point in shape calculus is how to define the shape derivative d_Ω. More specifically, the main problem here is that domains are sets and as such the space of admissible shapes has no vector space nor topological structure. Hence, adding domains as well as speaking of distances between them makes no sense - let alone making statements about convergence and differentiability. Shape calculus methods such as the method of perturbation of identity, or speed method, overcome these deficiencies by providing both structures. As one of the basic references in this field, we refer to [11], and a rigorous mathematical treatment is provided by [12]. The *discretize first* point of view is treated in [2] whereas the *optimize first* approach in the form of Lagrange multipliers can be

found in [13], among others. Except for [9, 8], the *discretize first* ansatz is so far used in IGA.

There also are several other angles from which shape gradients can be viewed, for instance [14] from a Riemannian perspective which might be even more natural to the isogeometric setting than perturbation of identity. We also want to point to [15, 16] for more shape optimization problems and in particular for topology optimization, which we do not consider in this work.

In the following, we will first introduce IGA in **Section 2** and form the space $\mathcal{G}$ for admissible shapes. We claim that shape calculus in IGA is tailored towards these parameterizations in $\mathcal{G}$, but even though, it is just a special case of the method of perturbation of identity. We will review the latter briefly in **Section 3.1** before specializing it to shape calculus in IGA in **Section 3.2**. **Section 4** utilizes this theory to show that in IGA *discretize first-optimize then* is the same as *optimize first-discretize then*. Results in **Section 5** illustrate the capability of shape optimization with IGA using the *optimize first–discretize then* ansatz.

2. Isogeometric Analysis

In this section, we briefly outline the basics of IGA and provide the necessary notation and mathematical foundation.

2.1. Preliminaries

We say that u satisfies a linear elliptic equation denoted by $E(u, \Omega) = 0$ if

$$u \in \mathcal{V} \colon a(u, v) = l(v) \quad \forall v \in \mathcal{V}, \tag{5}$$

with bilinear form

$$a(u, v) := \int_\Omega \left(\sum_{i,j=1}^{n} \sum_{k,\ell=1}^{m} a_{ik,j\ell} \partial_i u_k \partial_j v_\ell + \sum_{k,\ell=1}^{m} b_{k\ell} u_k v_\ell + c \right) \mathrm{d}\Omega \tag{6}$$

and linear form

$$l(v) := \int_\Omega f \cdot v \, \mathrm{d}\Omega + \int_{\Gamma_N} g \cdot v \, \mathrm{d}\Gamma. \tag{7}$$

Here, the coefficients $a_{ik,j\ell}$, $b_{k\ell}$, and c are in $L^\infty(\Omega)$, and we consider Neumann boundary conditions on Γ_N while homogeneous Dirichlet boundary conditions hold on Γ_D, with $\partial\Omega = \Gamma \cup \Gamma_D \cup \Gamma_N$. The function space $\mathcal{V}$ is the Sobolev space $\mathcal{V} := H^1_{\Gamma_D}(\Omega)^m := \{v = (v_1, \ldots, v_m) \in H^1(\Omega)^m \colon v|_{\Gamma_D} = 0\}$. Of course, we assume sufficient regularity of the domain, i.e., Ω is polygonal and convex or a Lipschitz domain or has a C^2-boundary. Finally, let $f \in L^2(\Omega)^m$ and $g \in L^2(\Gamma_N)^m$.

The general definition (6) of the bilinear form includes Poisson's equation $\Delta u = f$ where $m = 1$ and linear elasticity $\operatorname{div} \sigma(u) = f$ where $m = 2$ or 3 as important special cases, and we note that the dimension m of the solution field u is, in general, not equal to the dimension d of the domain Ω.

Let G denote a diffeomorphism from the unit cube $\hat{\Omega} := (0,1)^d$ to $\Omega \subset \mathbb{R}^d$ with bounded derivatives, i.e., it is an element of

$$\begin{aligned} \mathcal{G} := \{ G \in C^k(\mathbb{R}^d) \text{ such that } G\colon \hat{\Omega} \to \Omega \text{ is diffeomorphic and} \\ \exists c,\, C \in \mathbb{R} : \quad 0 < c \leq |\det J_G| \leq C \} \end{aligned} \tag{8}$$

with $k > 0$ and with Jacobian $J_G := (\partial G_i / \partial \hat{x}_j)$. Then, $u \mapsto \hat{u} \circ G^{-1}$ transforms $W^{s,p}(\hat{\Omega})$ boundedly onto $W^{s,p}(\Omega)$ and has a bounded inverse, c.f. [17]. This result also holds for Sobolev spaces that include boundary conditions.

As a consequence for $\phi \in L^1(\Omega)$, the change of variables

$$\int_\Omega \phi(x)\,\mathrm{d}x = \int_{\hat{\Omega}} (\phi \circ G)(\hat{x}) |\det J_G(\hat{x})|\,\mathrm{d}\hat{\Omega} \tag{9}$$

holds, and in particular for $\hat{u} := u \circ G \in \hat{\mathcal{V}} := H^1_{\hat{\Gamma}_D}(\hat{\Omega})^m$ a change of basis yields equivalent formulations for the state equation $E(u, \Omega) = 0$:

$$u \in \mathcal{V}\colon a(u,v) = l(v) \quad \forall v \in \mathcal{V} \tag{10}$$

$$\Leftrightarrow \quad \hat{u} \in \hat{\mathcal{V}}\colon \hat{a}_G(\hat{u},\hat{v}) = \hat{l}_G(\hat{v}) \quad \forall \hat{v} \in \hat{\mathcal{V}}. \tag{11}$$

The notation $a(u,v)$ versus $\hat{a}_G(\hat{u},\hat{v})$ emphasizes that in the variational form (10), the function spaces depend on Ω and thus on G in contrast to (11) where the dependency is moved to the operators in the bilinear and linear form that are given by

$$\hat{a}_G(\hat{u},\hat{v}) := \int_{\hat{\Omega}} \left(\sum_{i,j=1}^{n} \sum_{k,\ell=1}^{m} a_{ik,j\ell} \circ G (D\hat{u} DG^{-1})_{k,i} (D\hat{v} DG^{-1})_{\ell,j} + \right. \tag{12}$$

$$\left. \sum_{k,\ell=1}^{m} b_{k\ell} \circ G \hat{u}_k \hat{v}_\ell + c \circ G \right) |\det J_G|\,\mathrm{d}\Omega, \tag{13}$$

$$\hat{l}_G(\hat{v}) := \int_{\hat{\Omega}} f \circ G \cdot \hat{v} |\det J_G|\,\mathrm{d}\hat{\Omega} + \int_{\hat{\Gamma}_N} g \circ G \cdot \hat{v} |J_G^{-T} \hat{n}| |\det J_G|\,\mathrm{d}\hat{\Gamma} \tag{14}$$

with outer normal $\hat{n}$ to the boundary $\hat{\Gamma}_N = G^{-1}(\Gamma_N)$ in the parameter domain. The latter is a standard unit vector since $\hat{\Omega}$ is a unit cube by definition.

Indeed, the chain rule yields for any composite function $\hat{u} = u \circ G$

$$D\hat{u} = \big((Du) \circ G\big) DG = ((Du) \circ G) J_G. \tag{15}$$

with differential operator D and Jacobian J_G.

We illustrate this in the following example: Let $E(u, \Omega) = 0$ denote Poisson's equation with variational form

$$u \in H^1_0(\Omega)\colon \int_\Omega \nabla u \nabla v\,\mathrm{d}x = \int_\Omega f v\,\mathrm{d}x \quad \forall v \in H^1_0(\Omega). \tag{16}$$

Because of a change of variables, the chain rule and bounded derivatives of G equation, (16) is equivalent to $\hat{u} \in H^1_0(\hat{\Omega})$:

$$\int_{\hat{\Omega}} \nabla \hat{u} J_G^{-1} J_G^{-\mathrm{T}} \nabla \hat{v} |\det J_G|\,\mathrm{d}\hat{x} = \int_{\hat{\Omega}} f \circ G \hat{v} |\det J_G|\,\mathrm{d}\hat{x} \quad \forall \hat{v} \in H^1_0(\hat{\Omega}). \tag{17}$$

The next step is to specify the transformation map G.

2.2. Domains in Isogeometric Analysis

In IGA, a domain $\Omega \subset \mathbb{R}^d$ is represented by a geometry map

$$G\colon \hat{\Omega} \to \Omega, \quad \hat{x} \mapsto \sum_{i=1}^{n} \sum_{k=1}^{d} N_i(\hat{x}) X_{i,k} e_k \tag{18}$$

on the unit cube $\hat{\Omega}$ consisting of n B-splines or NURBS $N_i\colon \hat{\Omega} \to \mathbb{R}$ with control points $X_i := (X_{i,k})_{k=1,\dots,d} \in \mathbb{R}^d$ and standard unit vectors $e_k \mathbb{R}^d$

More specifically, G is generated as follows: Let $\mathcal{B}(\Sigma, p) := \{N_{i,p}\colon i = 1, \dots, m_\Sigma\}$ denote a basis of univariate B-splines or NURBS of order p, which are uniquely determined by a knot vector $\Sigma := (\hat{s}_1, \dots, \hat{s}_{m_\Sigma+p+1})$ with $\hat{s}_i \in [0,1]$. Let $\mathcal{S}(\Sigma, p) := \operatorname{span} \mathcal{B}(\Sigma, p)$ define the spline space generated by this basis. For d knot vectors $\Sigma_{i=1,\dots,d}$ and corresponding spline spaces $\mathcal{S}_i := \mathcal{S}(\Sigma_i, p_i)$ we form the tensor-product space [18]

$$\mathcal{S} := \bigotimes_{i=1}^{d} \mathcal{S}_i \qquad \text{with } n \text{ basis functions } N_i(\hat{x}) = N_{i_1}(\hat{x}_1) \cdots N_{i_d}(\hat{x}_d), \tag{19}$$

$\hat{x} \in [0,1]^d$, $\dim \mathcal{S} = n = \prod_{i=1}^{d} m_{\Sigma_i}$. We restrict the space of admissible domains to a subspace $\mathcal{G}_h := \mathcal{S}^d \cap \mathcal{G}$ of all diffeomorphisms with bounded derivatives, where $\mathcal{S}^d := \operatorname{span}\{N_i e_k\colon N_i \in \mathcal{B},\, k = 1, \dots, d\}$ is a direct sum and where the basis of $\mathcal{S}$ is given by $\mathcal{B} := \{N_i = \prod_{j=1}^{d} N_{i_j}(\hat{x}_j)\colon N_{i_j} \in \mathcal{B}(\Sigma_j, p_j)\}$.

The continuity of $\mathcal{S}$ determines the smoothness of parameterizations in $\mathcal{G}_h$. Specifically, it holds $G \in C^k\big((0,1)^d, \mathbb{R}^d\big)$ where $k \geq \min(p_i) \geq 1$ in each component.

In the following, we assume that the moving boundary Γ has a preimage

$$\hat{\Gamma} = \{\hat{x} \in \partial\hat{\Omega}\colon\, G(\hat{x}) \in \Gamma\}. \tag{20}$$

Furthermore, we require for $\hat{\Gamma}$ that either $(0,1)^{d-1} \times \{\hat{s}\} \subset \hat{\Gamma}$ or $(0,1)^{d-1} \times \{\hat{s}\} \cap \hat{\Gamma} = \emptyset$ for $\hat{s} = 0$ or 1. The same must apply to Γ_D and Γ_N. This means that for each boundary segment Γ, Γ_N and Γ_D, there is a B-Spline or NURBS parameterization over a union of sides of the parameter domain in $\mathcal{S}^{d-1}$. We illustrate this for a 2-dimensional example in Figure 1.

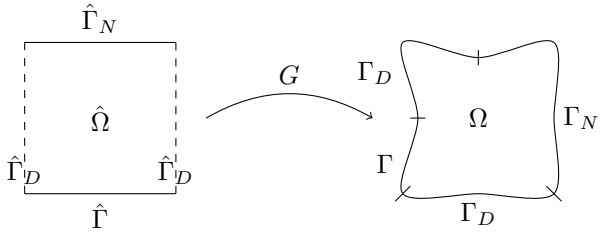


Figure 1: We assume that all boundary segments Γ, Γ_N and Γ_D can be mapped by a geometry function G as images of the union of codimension-1 manifolds of the form $[0,1]^{d-1} \times \{0\}$ and $[0,1]^{d-1} \times \{1\}$.

Remark 1. For the definition of B-splines we refer to various sources like [18, 19]. NURBS are treated in the monograph [20], and the above statements are valid for both. However, for the optimization part we need to be more specific in the NURBS case. NURBS are in principle weighted B-splines. Given a univariate B-Spline basis $\tilde{\mathcal{B}}(\Sigma, p) := \{N_{i,p} \colon i = 1, \dots, m_\Sigma\}$ and a fixed weight vector $W = (\omega_i)_{i=1,\dots,m_\Sigma}$, $\omega_i > 0$, let $w(\hat{x}) := \sum_{i=1}^{m_\Sigma} N_{i,p}(\hat{x})\omega_i$ denote a weight function from $\hat{\Omega} \to \mathbb{R}$ from which we obtain a NURBS basis $\mathcal{B}(W, \Sigma, p) := \{\frac{\omega_i}{w} N_{i,p} \colon N_{i,p} \in \tilde{\mathcal{B}}\}$. If there is no danger of confusion, we drop the index W in $\mathcal{B}(W, \Sigma, p) = \mathcal{B}(\Sigma, p)$.

2.3. Galerkin Projection in Isogeometric Analysis

Now, the fundamental idea in IGA is to use a finite dimensional subspace

$$\mathcal{V}_h := \{N \circ G^{-1} \colon N \in \mathcal{S}\}^m \cap \mathcal{V} = \mathcal{S}^m \circ G^{-1} \cap H^1_{\Gamma_D}(\Omega)^m \tag{21}$$

for the discrete Galerkin formulation $u_h \in \mathcal{V}_h \colon a(u_h, v_h) = l(v_h) \quad \forall v_h \in \mathcal{V}_h$. Practically, IGA uses equation (11) and solves for $\hat{u}_h \in \hat{\mathcal{V}}_h := \mathcal{S}^m \cap \hat{\mathcal{V}}$:

$$\hat{a}_G(\hat{u}_h, \hat{v}) = \hat{l}_G(\hat{v}) \quad \forall \hat{v} \in \hat{\mathcal{V}}_h. \tag{22}$$

As above, the hat notation $\hat{}$ indicates that a function is defined over the parameter space $\hat{\Omega} = (0, 1)^d$ or that a variable is from $\hat{\Omega}$.

Remark 2. In fact we could use any $G \in \mathcal{G}$ and are not restricted by theory to use a parameterization from a spline space, let alone the same one as the test functions. But particularly, given two different spline spaces $\mathcal{S}_1$ and $\mathcal{S}_2$, we could choose $\mathcal{G}_h = \mathcal{S}_1^d \cap \mathcal{G}$ and $\hat{\mathcal{V}}_h = \mathcal{S}_2^m \cap H^1_{\Gamma_D}(\hat{\Omega})^m$ and (22) still yields a legitimate solution $u_h = \hat{u}_h \circ G^{-1}$, with $\hat{u}_h \in \mathcal{V}_h$.

3. Shape Calculus

To provide a gradient w.r.t. domains in IGA we use the classical results in shape calculus and just apply integration by substitution. In this way, we can readily draw on known results from, e.g., [12, 13].

3.1. Shape Sensitivities in General

In principle, it is not the description of domains that matters but how changes on them affect the cost functional. Exemplary, the method of perturbation of identity uses a vector field $h \in X$ to change a domain

$$F_t := id + th, \quad F_t \colon \Omega \to \mathbb{R}^d \tag{23}$$

to obtain a perturbed domain and boundary segment

$$\Omega_t = F_t(\Omega), \quad \Gamma_t = F_t(\Gamma). \tag{24}$$

This method thus assigns the vector space structure of X to the space of admissible domains. We postpone the choice of X for the moment and remark that it is possible to

extend the notion of Gateaux- and Fréchet derivatives to sets (domains) by defining the derivative in direction of $h \in X$ as

$$J'(u,\Omega,\Gamma;h) := \lim_{t\to 0^+} \frac{1}{t}\big(J(u_t,\Omega_t,\Gamma_t) - J(u,\Omega,\Gamma)\big) \tag{25}$$

where u_t solves $E(u_t,\Omega_t) = 0$. In [12], the class of admissible perturbations is either chosen as Lipschitz-continuous functions $h \in X = C^{1,1}(\mathbb{R}^d)$ or $X = W^{k,\infty}(\mathbb{R}^d)$ or $X = C^k(\mathbb{R}^d)$. In all cases, the limit in (25) exists and J is Fréchet differentiable.

The limit $J'(u,\Omega,\Gamma;h)$ is also called Eulerian derivative. We write down the explicit formulas for boundary and domain integrals, cf. [13].

Lemma 1. *Let the shape functional $J := J(\Omega,\Gamma) = \int_\Omega j_1(x)\,\mathrm{d}x + \int_\Gamma j_2(s)\,\mathrm{d}s$ depend on Ω and Γ only as limits of integration. Then definition* (25) *gives*

$$J'(\Omega,\Gamma;h) = \int_\Omega \operatorname{div}(j_1 h)\,\mathrm{d}x + \int_\Gamma \nabla j_2 \cdot h + j_2(\operatorname{div} h - J_h n)\cdot n\,\mathrm{d}s \tag{26}$$

$$= \int_\Gamma j_1 h\cdot n\,\mathrm{d}s + \int_\Gamma \left(\frac{\partial j_2}{\partial n} + j_2\kappa\right) h\cdot n\,\mathrm{d}s \tag{27}$$

with the curvature of Γ denoted by κ. We refer to (26) *as the domain and to* (27) *as the Hadamard representation of sensitivities.*

In general, $J'\colon X \to L(X,\mathbb{R})$ denotes the Fréchet-derivative of a linear functional $J\colon X \to \mathbb{R}$ on a Banach space X and $J'(x;\delta x) =$ its directional derivative. Moreover, for multiple variables $(x,y) \in X$, the "partial derivatives" are defined by $J'_x(x,y;\delta x) = \lim_{t\to 0} \frac{1}{t}\big(J(x+t\delta x,y) - J(x,y)\big)$.

3.2. Shape Sensitivities in Isogeometric Analysis

In IGA shape calculus is tailored towards the representation of geometries by NURBS and B-splines. In particular, the perturbation of identity in (23) is pulled back to the parameter domain via

$$G_t := F_t \circ G = G + th \circ G \tag{28}$$

with a given parameterization of Ω by the geometry function $G \in \mathcal{G}$. Further, we choose the vector field h due to the isogeometric paradigm as $h = \theta \circ G^{-1}$ with $\theta \in \mathcal{S}^d$. Since G^{-1} is continuous by definition, $\mathcal{S}^d \subset C^1\big((0,1)^d,\mathbb{R}^d\big)$, and since Fréchet derivatives are linear and continuous w.r.t. h, it holds

$$J'(u,\Omega,\Gamma;\theta) := \lim_{t\to 0^+} \frac{1}{t}\big(J(u_t,\Omega_t,\Gamma_t) - J(u,\Omega,\Gamma)\big) \tag{29}$$

where $\Omega_t = (G + t\theta)(\hat{\Omega})$, is well-defined. Figure 2 shows the relationship of the perturbation of identity and IGA. We then apply the transformation formula to the equations in Lemma 1 with the isogeometric variation $h = \theta \circ G^{-1}$.

We can reformulate this result in a manner that directly applies to IGA.

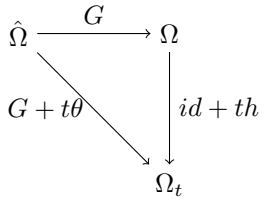


Figure 2: Perturbation of identity with IGA.

Lemma 2. *For a shape functional J as in Lemma 1, we obtain the isogeometric shape sensitivities in direction $\theta \in \mathcal{S}^d$ as*

$$J'(G, \hat{\Gamma}; \theta) = \int_{\hat{\Omega}} \left(\nabla j_1 \circ G \cdot \theta + j_1 \circ G \operatorname{tr}(J_G^{-1} J_\theta)\right) |\det J_G| \, \mathrm{d}\hat{x} \tag{30}$$

$$+ \int_{\hat{\Gamma}} \Big(\nabla j_2 \circ G \cdot \theta |J_G^{-T} \hat{n}| + j_2 \circ G \operatorname{tr}(J_G^{-1} J_\theta) \tag{31}$$

$$- j_2 \circ G \frac{\hat{n}^T J_G^{-1} J_G^{-T} J_\theta^T J_G^{-T} \hat{n}}{|J_G^{-T} \hat{n}|} \Big) |\det J_G| \, \mathrm{d}\hat{s}. \tag{32}$$

Proof. One considers (26) and applies the transformation rule where we note that $\operatorname{div}(j_1 h) = \nabla j_1 \cdot h + j_1 \operatorname{div}(h)$, $\operatorname{div}(h) \circ G = \operatorname{tr}(J_G^{-1} D\theta)$ and $n \circ G = \frac{J_G^{-T} \hat{n}}{|J_G^{-T} \hat{n}|}$.

Analogously, we can also transform equation (27) to obtain a Hadamard representation of shape sensitivities.

3.3. Shape Sensitivities for Optimal Weights

In [7], shape optimization simultaneously over control points and weights has been performed for an example in linear elasticity, based on the discretize-first ansatz. Together with the framework of Section 3.2, we now extend weight optimization to general elliptic PDEs for the optimize-first scheme.

In Section 2 we introduced B-spline and NURBS spaces. For the latter we assumed a fixed vector of weights W, see Remark 1. In shape optimization, however, weights might be another instrument of fine-tuning. But, if W is not fixed, a much larger space than $\mathcal{S}$ of (19) is available in order to search for the best shape. Different to the notation of Remark 1, we have free weights. We observe that the weight function $w := \sum_{i=1}^{m_\Sigma} N_{i,p} \omega_i$ is a positive combination of B-splines

$$w \in \mathcal{S}_+(\Sigma, p) := \operatorname{pos} \mathcal{B}(\Sigma, p) \subset \mathcal{S}(\Sigma, p), \tag{33}$$

and a set of NURBS with free weights is given by

$$\mathcal{N}(\Sigma, p) := \mathcal{S}(\Sigma, p) \times \mathcal{S}_+(\Sigma, p). \tag{34}$$

Unfortunately, it is not directly obvious how $\mathcal{N}(\Sigma, p)$ turns into a vector space because any free weight NURBS function $(s, w) := \frac{s}{w}$ in $\mathcal{N}(\Sigma, p)$ is non-linear w.r.t. the weight functions w. This also reflects that the rational functions $\frac{1}{x-a}$ and $\frac{1}{x-b}$ are linearly

independent for $a \neq b$. In order to restore linearity we resort to homogeneous coordinates and a perspective map, see [20]. A homogeneous coordinate vector $\tilde{X}$ in $\mathbb{R}^{d+1}$ with $\tilde{X} := (X^w, X_{d+1})$ and $X^w \in \mathbb{R}^d$ is projected to $\mathbb{R}^d$ by

$$H(\tilde{X}) = \begin{cases} X^w / X_{d+1}, & X_{d+1} \neq 0 \\ X^w / |X_{d+1}|, & \text{else.} \end{cases}$$

Given a B-Spline space $\mathcal{S}^d$ as in Section 2.2 a rational space $\mathcal{N}$ of free weights is obtained by the direct sum $\mathcal{S}^d \oplus \mathcal{S}$ and the perspective map H. More specifically, the vector valued function $(G^w, w) \in \mathcal{S}^{d+1}$, with $G^w \in \mathcal{S}^d$, is mapped from $\mathcal{S}^{d+1} \to \mathcal{S}^d$, which yields a rational d-manifold

$$(G^w, w) \mapsto \begin{cases} G := G^w / w & \text{if } w \neq 0, \\ G := G^w / \|G^w\| & \text{if } w = 0. \end{cases} \tag{35}$$

Finally, $\mathcal{S}^d \oplus \mathcal{S}$ has a basis $\{N_i e_k \colon N_i \in \mathcal{S},\, i = 1, \dots, m,\, k = 1, \dots, d+1\}$ and is a linear space which is isomorphic to $\mathbb{R}^{md} \times \mathbb{R}^d$. Then, the desired space $\mathcal{N}$ is obtained as $\mathcal{N} = \{H(\tilde{G}) \colon \tilde{G} = (G^w, w) \in \mathcal{S}^d \oplus \mathcal{S}\}$. We use this in the following sensitivity formula.

Lemma 3. *For a shape functional J as in Lemma 1, we obtain the isogeometric shape sensitivities in direction $\tilde{\theta} \in \mathcal{N}$ for $G = H \circ \tilde{G}$ and $\theta = H \circ \tilde{\theta}$ as*

$$\begin{aligned} &J'(G, \hat{\Gamma}; \theta) \\ &\quad = \int_{\hat{\Omega}} \left(\nabla j_1 \circ G \cdot DH \circ \tilde{G} \tilde{\theta} + j_1 \circ G \operatorname{tr}(J_G^{-1} \dot{D}) \right) |\det J_G| \, \mathrm{d}\hat{x} && (36) \\ &\quad + \int_{\hat{\Gamma}} \Bigg(\nabla j_2 \circ G \cdot DH \circ \tilde{G} \tilde{\theta} |J_G^{-T} \hat{n}| + j_2 \circ G |J_G^{-T} \hat{n}| \operatorname{tr}(J_G^{-1} \dot{D}) + && (37) \\ &\qquad - j_2 \circ G \frac{J_G^{-T} \dot{D}^T J_G^{-T}}{|J_G^{-T} \hat{n}|} \Bigg) |\det J_G| \, \mathrm{d}\hat{x} && (38) \end{aligned}$$

with $\dot{D} := \mathrm{d}_t \, D(H \circ \tilde{G}_t)|_{t=0}$.

Proof. Since we use homogeneous coordinates for $\tilde{G}$ and the projective map H, this implies that in the substitution of variables in the shape functional we have

$$\begin{aligned} J(G, \hat{\Gamma}) = &\int_{\hat{\Omega}} j_1 \circ (H \circ \tilde{G}) |\det(D(H \circ \tilde{G}))| \, \mathrm{d}\hat{x} \\ &+ \int_{\hat{\Gamma}} j_2 \circ (H \circ \tilde{G}) |D(H \circ \tilde{G})^{-1} \hat{n}| |\det(D(H \circ \tilde{G}))| \, \mathrm{d}\hat{x}. \end{aligned}$$

For the limit $\mathrm{d}_t \, J(G_t, \hat{\Gamma})|_{t=0}$ with $G_t = H \circ (\tilde{G} + t\tilde{\theta})$ we need the transformations

$$J_G = DG = D(H \circ \tilde{G}) = DH \circ \tilde{G} D\tilde{G} \text{ and} \tag{39}$$

$$DH \circ \tilde{G} = \frac{1}{w} \begin{pmatrix} I_d, & -G \end{pmatrix}, \tag{40}$$

with identity matrix $I_d \in \mathbb{R}^{d\times d}$ and assuming positive weights. Therefor it holds for $\tilde{G}_t := \tilde{G} + t\tilde{\theta}$

$$\dot{D} = \mathrm{d}_t\, D(H\circ\tilde{G}_t)|_{t=0} = D^2H\circ\tilde{G}\odot\tilde{\theta}D\tilde{G} + DH\circ\tilde{G}D\tilde{\theta}, \tag{41}$$

where $D^2H(x)\odot v := (\mathrm{d}_{x_1} DHv, \ldots, \mathrm{d}_{x_{d+1}} DHv) \in \mathbb{R}^{d\times d+1}$ for x and $v \in \mathbb{R}^{d+1}$. With this notation we get explicitly

$$\begin{aligned} &D^2H\circ\tilde{G}\odot\tilde{\theta} = \\ &= \frac{1}{w^2}\left((0_d,-e_1)\tilde{\theta},\ldots,(0_d,-e_d)\tilde{\theta},(-I_d,2G)\tilde{\theta}\right) &(42)\\ &= \begin{cases} \frac{1}{w^2}\left(0_d, \quad -\theta^w\right) & \text{, if } \tilde{\theta}_{d+1}=0 \\ \frac{-\tilde{\theta}_{d+1}}{w^2}\left(I_d, \frac{1}{\tilde{\theta}_{d+1}}\theta^w - 2G\right) & \text{, if } \tilde{\theta}_{d+1}\neq 0. \end{cases} &(43)\end{aligned}$$

From the implicit function theorem we obtain

$$\begin{aligned} 0 &= \mathrm{d}_t\, I_d = \mathrm{d}_t\big(D(H\circ\tilde{G}_t)^{-1}D(H\circ\tilde{G}_t)\big)|_{t=0} \\ &= \mathrm{d}_t\big(D(H\circ\tilde{G}_t)^{-1}\big)|_{t=0}D(H\circ\tilde{G}) + D(H\circ\tilde{G})^{-1}\dot{D}. \end{aligned}$$

Hence,

$$\mathrm{d}_t\big(D(H\circ\tilde{G}_t)^{-1}\big)|_{t=0} = -J_G^{-1}\dot{D}J_G^{-1},$$

which completes the proof.

Shape sensitivities allow to check the necessary optimality condition of equation (4), but moreover are useful to find an optimal shape by means of gradient based optimization methods, as is studied in the next section.

4. Shape Optimization

Relying on the information of the shape sensitivities, gradient-based optimization methods can be employed to find an optimal shape. Basically, a sequence of domains $\{\Omega_k\}_{k=0,1,\ldots}$ is generated by an update rule

$$\Omega_{k+1} = (id + ts_k)(\Omega_k) \tag{44}$$

where s_k is a descent direction, typically $s_k = -\nabla_{\Omega_k} J(\Omega_k)$.

To solve a shape optimization problem we can either (i) discretize both control and state variables, then optimize the resulting finite-dimensional problem. Or (ii) we set up the necessary optimality conditions for the infinite-dimensional problem and then discretize control and state. In the gradient descent method, the update rule of equation (44) employs a descent direction resp. gradient from a real vector space for scheme (i) while in case (ii) the descent direction is an element of a function space.

In the following, we formulate first the *discretize first–optimize then* scheme (i) with IGA and thereafter the contrary case (ii) *optimize first–then discretize*. Subsequently, we show that both methods eventually lead to the same discrete system.

In IGA, we discretize state and control, i.e., the admissible domains, by B-splines and NURBS, respectively by the spaces $\hat{\mathcal{V}}_h$ and $\mathcal{G}_h$. For IGA again it is therefor better to use the transported objective

$$J(u,\Omega,\Gamma) = J(\hat{u},G,\hat{\Gamma}) := \\ = \int_{\hat{\Omega}} j_1(\hat{u},G)|\det J_G|\,\mathrm{d}\hat{x} + \int_{\hat{\Gamma}} j_2(\hat{u},G)|J_G^{-T}\hat{n}||\det J_G|\,\mathrm{d}\hat{s}. \tag{45}$$

Note that in case the functions j_1 and j_2 contain a differential operator D applied to u, we additionally have to consider the transformation $Du \circ G = D(u \circ G)DG^{-1}$, as is the case for the bilinear and linear forms in equation (12)–(14).

4.1. Discretize First–Optimize Then

Spline spaces are isomorphic to real vector spaces, which means that $\mathcal{G} \cong \mathbb{R}^{nd}$ and $\mathcal{V}_h \cong \mathbb{R}^{nm}$ in our setting. So, discretizing the control by B-splines or NURBS means that any $\Omega = G(\hat{\Omega})$ can be expressed as $\Omega(X) \in \mathcal{G}_h$ where $X \in \mathbb{R}^{nd}$. In other words, any domain is represented by a vector of control points, corresponding to the classically called "design variables".

The Galerkin discretization yields a discrete state $u_h = \sum q_i N_i \circ G^{-1}$ where the coefficients q_i are given by the linear equation $Kq = F$ derived from the projected weak form (22). In terms of X and q and with the discrete cost function $J_h(q,X) := J(\hat{u}_h, \Omega(X), \Gamma(X))$ we formulate the discrete isogeometric optimization problem

$$\min J_h(q,X) \quad \text{subject to } Kq = F. \tag{46}$$

For an optimal pair (q,X) the necessary first order optimality conditions are

1. State equation: $Kq = F$,
2. Stationary point: $\nabla_X J_h(q,X) = 0$.

Because q depends on X implicitly, we have to take the derivative $\partial_X q$ for the stationary point condition into account. For this purpose we introduce the index pair $\alpha = (i,k)$ where $i = 1,\dots,n$ runs over the control points and $k = 1,\dots,d$ over their components. By $\mathrm{d}_\alpha := \mathrm{d}_{X_\alpha}$ and $\partial_\alpha := \partial_{X_\alpha}$ we denote the derivatives with respect to component k of control point X_i. Then by the chain rule

$$\mathrm{d}_\alpha J_h(q,X) = \partial_\alpha J_h + \partial_q J_h \cdot \partial_\alpha q \tag{47}$$

where the shape derivative of q respectively u_h is given by

$$K\partial_\alpha q = \partial_\alpha F - \partial_\alpha K q. \tag{48}$$

We introduce the adjoint p as the solution of $K^{\mathrm{T}}p = \partial_q J_h$ and obtain

$$\mathrm{d}_\alpha J_h(q,X) = \partial_\alpha J_h + p^{\mathrm{T}}(\partial_\alpha F - \partial_\alpha K q). \tag{49}$$

Summing up, we have the necessary optimality conditions for an optimal pair (q,X): For all tuples $\alpha = (i,k)$

$$Kq = F, \tag{50}$$

$$Kp = \partial_q J_h, \tag{51}$$

$$\mathrm{d}_\alpha J_h(q,X) = \partial_\alpha J_h + p^{\mathrm{T}}(\partial_\alpha F - \partial_\alpha K q) = 0. \tag{52}$$

4.2. Optimize First–Discretize Then

In this case, we couple cost functional and state equation in the Lagrangian

$$\mathcal{L}(\Omega, y, v) := J(y, \Omega, \Gamma) + l(v) - a(y, v), \tag{53}$$

or in the isogeometric semi-discrete version, where $G \in \mathcal{G}_h$,

$$\mathcal{L}(G, \hat{y}, \hat{v}) := J(\hat{y}, G, \hat{\Gamma}) + l_G(\hat{v}) - a_G(\hat{y}, \hat{v}) \tag{54}$$

where all variables G, $\hat{y}$ and $\hat{v}$ are independent.

Under the assumption that the state u and thus $\hat{u} = u \circ G$, is shape differentiable, the necessary optimality conditions are given by the Karush-Kuhn-Tucker system $\nabla\mathcal{L} = 0$, i.e., at an optimum $(\hat{u}, \hat{z}, G)$ it holds

$$\mathcal{L}'_{\hat{v}}(G, \hat{u}, \hat{z}; \delta\hat{v}) = 0 \quad \forall \delta\hat{v} \in \hat{\mathcal{V}}, \tag{55}$$

$$\mathcal{L}'_{\hat{y}}(\Omega, \hat{u}, \hat{z}; \delta\hat{u}) = 0 \quad \forall \delta\hat{u} \in \hat{\mathcal{V}}, \tag{56}$$

$$\mathcal{L}'_{G}(G, \hat{u}, \hat{z}; \theta) = 0 \quad \forall \theta \in \mathcal{S}^d. \tag{57}$$

Explicitly, the KKT system in IGA is given by the next lemma.

Lemma 4. *The first order necessary optimality system of equations* (55)–(57) *in isogeometric shape optimization translates to*

$$a_G(\hat{u}, \delta\hat{v}) = l_G(\delta\hat{v}) \qquad \forall \delta\hat{v} \in \hat{\mathcal{V}} \qquad \textit{state equation,} \tag{58}$$

$$J'_{\hat{y}}(\hat{u}, G, \hat{\Gamma}; \delta\hat{u}) = a_G(\delta\hat{u}, \hat{z}) \quad \forall \delta\hat{u} \in \hat{\mathcal{V}} \qquad \textit{adjoint equation,} \tag{59}$$

with $J'_{\hat{y}}(\hat{u}, G, \hat{\Gamma}; \delta\hat{u}) = \mathrm{d}_t\, J(\hat{u} + t\delta\hat{u}, G, \hat{\Gamma})|_{t=0}$. *The shape gradient is formed by directional derivatives*

$$\mathcal{L}'_G(G, \hat{y}, \hat{v}; \theta) = J'_G(\hat{y}, G, \hat{\Gamma}; \theta) + l'_G(\hat{v}; \theta) - a'_G(\hat{y}, \hat{v}; \theta) \tag{60}$$

with $J'_G(\hat{y}, G, \hat{\Gamma}; \theta)$ *given by expression* (30) *in Lemma 2,*

$$l'_G(\hat{v}; \theta) = \mathrm{d}_t\, l_{G+t\theta}(\hat{v})|_{t=0} \text{ and } a'_G(\hat{y}, \hat{v}; \theta) = \mathrm{d}_t\, a_{G+t\theta}(\hat{y}, \hat{v})|_{t=0}.$$

This semi-discrete systems turns fully discrete by projecting $\hat{\mathcal{V}}$ onto $\hat{\mathcal{V}}_h$.

4.3. Comparison of the Two Discretization Schemes

The two approaches above are reflected in IGA by the works [8, 9] for the optimize first and [6, 7, 4, 10] for the discretize first ansatz. Both schemes yield shape sensitivities also inside the domain, in contrast to a nodal approach in (isoparametric) FEM where typically only gradients on the boundary are available.

Since we claim the fully discrete systems are equal for both approaches, what makes the difference? First of all, from a theoretical point of view to show well-posedness of (shape) optimization problems answering for existence and uniqueness of and convergence to optimal solutions we need the continuous case, i.e., the *optimize first* part. From a practical angle, some problems require a different ansatz space for the adjoint in (56) than $\hat{\mathcal{V}}$, see [21], which is in favor of *optimize first*.

Whichever ansatz one prefers, eventually it yields the same discrete optimality system in IGA. For finite dimensional Banach spaces the directional derivative of j_i in direction $h_\alpha = N_i \circ G^{-1} e_k$ is just the partial derivative $\partial_\alpha j_i \circ G^{-1}$. That is why for Galerkin methods it does not matter if we use a *discretize first–optimize then* or *optimize first–discretize then* ansatz. We summarize this in the following theorem.

Theorem 5. *Assume that $G \in \mathcal{G}_h$ with $\Omega = G(\hat{\Omega})$ and that the discretization space for the adjoint in equation* (56) *is $\hat{\mathcal{V}}_h$. Then, the partial derivatives w.r.t. shape obtained in the* discretize first-optimize then *and those from the* optimize first–discretize then *ansatz are equal in IGA.*

Proof. Equation (58) yields the discrete state equation $Kq = F$ of (50) and equation (59) the discrete adjoint $K^T p = \partial_q J_h$ in (51).

Consider a shape functional without implicit dependency on the domain

$$\begin{aligned} J(\Omega) &= \int_\Omega \phi(x)\,\mathrm{d}x, \qquad \phi\colon \Omega \to \mathbb{R}, \\ &= \int_{\hat{\Omega}} \phi \circ G |\det DG|\,\mathrm{d}\hat{x}. \end{aligned} \tag{61}$$

In IGA we set $\partial_{X_\alpha} := \partial_\alpha$ for an arbitrary control point X_α for some $\alpha = (i,k)$, $i = 1,\dots,n$ and $k = 1,\dots,m$. Then,

$$\partial_\alpha J_h(X) = \int_{\hat{\Omega}} \Big((\nabla\phi) \circ G \cdot \partial_\alpha G + \phi \circ G \mathrm{tr}(J_G^{-1} \partial_\alpha J_G) \Big) |\det J_G|\,\mathrm{d}\hat{x} \tag{62}$$

from the chain rule on the transported shape function in (61). On the other hand, with $h = N_\alpha \circ G^{-1}$ and Lemma 2 we have

$$J'(G; N_\alpha) = \int_{\hat{\Omega}} \big((\nabla\phi) \circ G \cdot N_\alpha + \phi \circ G \mathrm{tr}(J_G^{-1} DN_\alpha) \big) |\det J_G|\,\mathrm{d}\hat{x}. \tag{63}$$

From direct calculation it can be seen that $\partial_\alpha G = \partial_\alpha \sum X_{\alpha_i} N_{\alpha_i} = N_\alpha$ and $\partial_\alpha J_G = DN_\alpha$, hence $J'(G; N_\alpha) = \partial_{X_\alpha} J_h(X)$.

Because in $l_G(\hat{v})$ as well as $a_G(\hat{u}, \hat{v})$ functions $\hat{u}$, $\hat{v}$ are independent of G, the same arguments as for equations (62) and (63) apply component wise:

$$\partial_\alpha F_{\alpha_i} = l'_G(N_{\alpha_i}; N_\alpha)$$

and

$$\partial_\alpha K_{\alpha_i,\alpha_j} = a'_G(N_{\alpha_i}, N_{\alpha_j}; N_\alpha).$$

Therefor

$$\begin{aligned} \mathrm{d}_\alpha J_h(X) &= \partial_\alpha J_h + p^{\mathrm{T}}(\partial_\alpha F - \partial_\alpha K q), \\ J'(\Omega; N_\alpha) &= \partial_\alpha J_h + \sum_j p_j \partial_\alpha F_j - \sum_{i,j} q_i \partial_\alpha K_{ij} p_j, \end{aligned}$$

and the discrete systems are equal.

4.4. Different Analysis and Optimization Models

Consider Remark 2 where we assumed two spline spaces $\mathcal{S}_1$, $\mathcal{S}_2$: one for the geometry representation and one for the projection space. Since optimization is very costly for each design variable we want to keep the number of design variables low. However, the interpolation error of the state equation directly plays a role in evaluating the objective function J and also in the accuracy of its gradient, which in turn influences the optimization. As an intuitive example, suppose we solve the Poisson equation on the unit square with homogeneous Dirichlet boundary conditions with only two linear basis functions per direction, then $u_h = 0$, which leaves not much scope for the optimizer. Hence, a good approximation of the state is vital. For detailed studies on the influence of the error $u - u_h$, see for instance [22, 23].

Both schemes, *optimize first–discretize then* and *discretize first–optimize then* allow for this scenario of two spline spaces. In the Karush-Kuhn-Tucker system (58)–(60) the variations $\delta\hat{v}$ and $\delta\hat{u}$ are taken from $\hat{\mathcal{V}} := \mathcal{S}_2^m \cap H_0^1(\hat{\Omega}^m$, whereas the for domain perturbations one selects $\theta \in \mathcal{S}_1^d$. In the discretize first system (50)–(52) the shape derivatives ∂_{X_α} refer to control points for $G \in \mathcal{S}_1^d$.

4.5. Extending the Shape Gradient to the Interior of the Domain

Depending on the formulation of the directional derivative in Lemma 1 we have either a domain or a boundary representation. Furthermore in the spirit of saving, the choice of variations $h = \theta \circ G^{-1}$ can be restricted to the boundary of interest, taking only

$$\theta \in \{N \in \mathcal{S}^d \colon N \neq 0 \text{ on } \hat{\Gamma}\}$$

instead of $\theta \in \mathcal{S}^d$. In the case of boundary representations for the shape gradient or the restricted space of variations, the extension of the shape sensitivities to the interior of

(a) initial parameterization with moving boundary $\hat{\Gamma}$

(b) overlapping parameterization after moving only the boundary

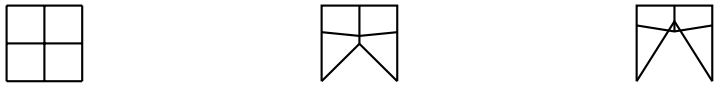
Figure 3: Moving only the boundary can cause irregular parameterizations

the domain can be achieved by several methods. The following incomplete list gives an overview of velocity based measures as have been applied in IGA or FEM:

- minimizing the Winslow functional in electromagnetic shape optimization, which aims for an evenly spread determinant of the Jacobian of G [5],

- solving a Poisson problem to distribute the control points in the interior, where the displacement of the control points on the boundary serves as inhomogeneous Dirichlet boundary conditions,
- solving a linear elasticity problem with the displacement of control points on the boundary as Neumann boundary condition [6, 7, 8],
- Analysis-aware IGA meshes [24], which also optimize the magnitude of the determinant of the Jacobian of G.

However, be aware of mesh tangling or mesh racing [25], which may happen in the Poisson and linear elasticity methods above if the step size in an optimization loop is too large and leads to non-convex domains. For instance in the toy example shown in Figure 4, the Poisson moving method yields irregular parameterizations for some large step sizes.



(a) $t = 0$ (b) $t = 0.5$ (c) $t = 0.8$

Figure 4: A minimal example to show that moving the inner control points by solving a Poisson equation may yield irregular parameterizations if changes to a domain are too large and non-convex. We use the knot vector $(0, 0, 0.5, 1, 1)$ for both directions for B-splines of degree $p = 1$ and variable second control point $(0.5, t)^T$ with $0 \leq t < 1$. The initial configuration is given at $t = 0$. For $t = 0.5$ we have an admissible step size, $t = 0.8$ is already too large.

Solving an additional optimization problem to obtain a good parameterization can be quite expensive, too. Furthermore, for analysis-aware meshes a good a posteriori error estimator is required.

An alternative is to use a relative positioning of inner control points, which is quite easy to realize due to the tensor product space structure: In $d = 2$ dimensions, let the boundaries of Ω be given by functions γ_N, γ_S, γ_W and γ_E which are B-spline or NURBS parameterizations. For instance $\gamma_S = G(\hat{x}_1, 0)$, $\gamma_N = G(\hat{x}_1, 1)$, $\hat{x}_1 \in (0, 1)$, and so forth.

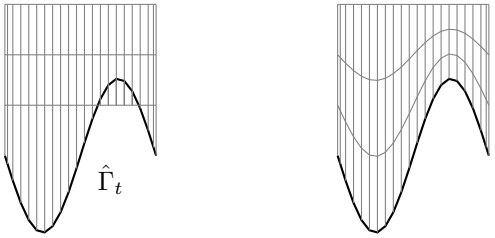
Then

$$T(\hat{x}_1, \hat{x}_2) = id + \begin{pmatrix} \gamma_W(\hat{x}_2)(1 - \hat{x}_1) - \hat{x}_1(1 - \gamma_E(\hat{x}_2)) \\ \gamma_S(\hat{x}_1)(1 - \hat{x}_2) - \hat{x}_2(1 - \gamma_N(\hat{x}_1)) \end{pmatrix} \tag{64}$$

transforms $\hat{\Omega}$ to Ω. In particular, we get equidistantly spaced control points by mapping the intersecting grid points of $\hat{\Omega}$ by means of T to Ω. An example is shown in Figure 5. The method works if control points only move up and down as in the minimal example in Figure 4. However, it works only to some extent if control points move diagonally as in Figure 6.

5. Numerical Results

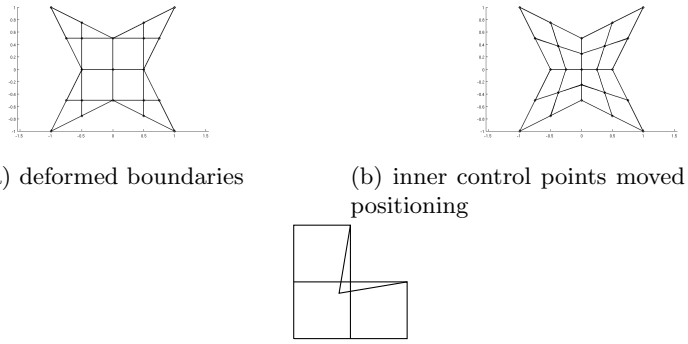
In the following, we study two examples to illustrate the above theoretical framework. Example 1 is a purely geometric problem with a length constraint instead of a PDE constraint and meant as introduction to shape calculus as well as B-Spline and NURBS



(a) overlapping parameterization after moving only the boundary

(b) parameterization by relative positioning of inner control points

Figure 5: Adjusting inner control points by relative positioning to avoid mesh tangling: instead of moving only the boundary, use transformation T of (64) to obtain an equidistant parameterization



(a) deformed boundaries

(b) inner control points moved by relative positioning

(c) relative positioning fails for diagonally non-convex domains

Figure 6: An example where relative positioning of inner control points works in non-convex domains and one where it does not

based geometry representation. Although simple, it highlights the benefits from using B-Splines and NURBS but also exposes the pitfalls when it comes to extending the gradient to the interior of the domain.

As second example, we investigate the classical compliance minimization, which has also been treated in [6, 7, 8]. Here, we show the deep correspondence of the models in design, simulation, and optimization in IGA.

5.1. Example with Geometric Constraint

We want to maximize the area of a domain Ω such that the length of its perimeter stays invariant. We choose this example to demonstrate the combination of shape optimization with NURBS and B-splines. It also serves as simple case for an IGA shape gradient derivation.

Example 1 (Area Maximization). For $\Omega \subset \mathbb{R}^2$ we consider the shape optimization problem

$$\min J(\Omega) := -\int_\Omega \mathrm{d}\Omega, \text{ s.t. } \int_{\partial\Omega} \mathrm{d}\Gamma = P_0. \tag{65}$$

The optimal shape then is a circular domain with radius $r = \frac{P_0}{2\pi}$.

Suppose we have a geometry function G in a B-Spline or NURBS space with fixed weights $\mathcal{S}^2 = \mathrm{span}\{N_{i,k} = N_i e_k,\, i = 1, \dots, n,\, k = 1, 2\}$. Then $G = \sum_{i=1}^{n} \sum_{k=1}^{2} X_{i,k} N_i e_k = \sum_{\alpha=(i,k)} X_\alpha N_\alpha$ with control points X_i. The transported symmetric problem reads

$$\min J(G) := -\int_{(0,1)^2} |\det J_G| \,\mathrm{d}\hat{\Omega} \tag{66}$$

for the objective, and for the length constraint $C(G) = 0$,

$$C(G) := \int_0^1 |\,\mathrm{d}_{\hat{s}}\, G(\hat{s}, 0)| + |\,\mathrm{d}_{\hat{s}}\, G(\hat{s}, 1)| + |\,\mathrm{d}_{\hat{s}}\, G(0, \hat{s})| + |\,\mathrm{d}_{\hat{s}}\, G(1, \hat{s})|\,\mathrm{d}\hat{s} - P_0, \tag{67}$$

because $J_G^{-T} \hat{n} \det J_G = \mathrm{d}_{\hat{s}}\, G(\gamma(\hat{s}))$, $\gamma(\hat{s}) \in \{(0, \hat{s}), (1, \hat{s}), (\hat{s}, 0), (\hat{s}, 1)\}$.

Sensitivities. Lemma 2 yields the shape derivatives for $\theta = N_\alpha \in \mathcal{S}^2$

$$J'(G; N_\alpha) = -\int_{(0,1)^d} \mathrm{tr}(J_G^{-1} J_{N_\alpha}) |\det J_G| \,\mathrm{d}\hat{\Omega}, \tag{68}$$

$$\begin{aligned} C'(G; N_\alpha) = \int_0^1 & \frac{\mathrm{d}_{\hat{s}}\, G(\hat{s}, 0) \cdot \mathrm{d}_{\hat{s}}\, N_\alpha(\hat{s}, 0)}{|\,\mathrm{d}_{\hat{s}}\, G(\hat{s}, 0)|} + \frac{\mathrm{d}_{\hat{s}}\, G(\hat{s}, 1) \cdot \mathrm{d}_{\hat{s}}\, N_\alpha(\hat{s}, 1)}{|\,\mathrm{d}_{\hat{s}}\, G(\hat{s}, 1)|} + \\ & + \frac{\mathrm{d}_{\hat{s}}\, G(0, \hat{s}) \cdot \mathrm{d}_{\hat{s}}\, N_\alpha(0, \hat{s})}{|\,\mathrm{d}_{\hat{s}}\, G(0, \hat{s})|} + \frac{\mathrm{d}_{\hat{s}}\, G(1, \hat{s}) \cdot \mathrm{d}_{\hat{s}}\, N_\alpha(1, \hat{s})}{|\,\mathrm{d}_{\hat{s}}\, G(1, \hat{s})|}\,\mathrm{d}\hat{s}. \end{aligned} \tag{69}$$

Results. We performed the optimization with the SQP algorithm of MATLAB's constrained minimization function `fmincon`, which uses the shape gradient information to update a quasi-Newton approximation of the Hessian of the Lagrangian function L of the cost functional and the constraint on the control. For this example, $L(\lambda, G) = J(G) + \lambda C(G)$ with Lagrange multiplier λ. We examined both, the influence of the degree p of our B-spline basis and of the discretization parameter h corresponding to a refinement by knot insertion.

From the convergence plot in Figure 7 we learn that a higher degree p of our B-spline basis speeds up convergence. Intuitively this is what one expects, since the approximation power of B-spline curves to sufficiently smooth functions f goes like $\max(Qf - f) \leq Ch^{p+1}$ for the B-spline interpolation Qf of f, see [19, Jackson type estimate].

We also tested the simultaneous optimization of control points and weights in this example. For $p = 2$ and knot vectors $(0, 0, 0, 1, 1, 1)$ in both directions, there is a NURBS representation for a disk. However, we observed that `fmincon` finds only almost optimal control points and weights within an absolute error of $J(\Omega^*) - J(\Omega) = 4.6 \cdot 10^{-6}$ in the

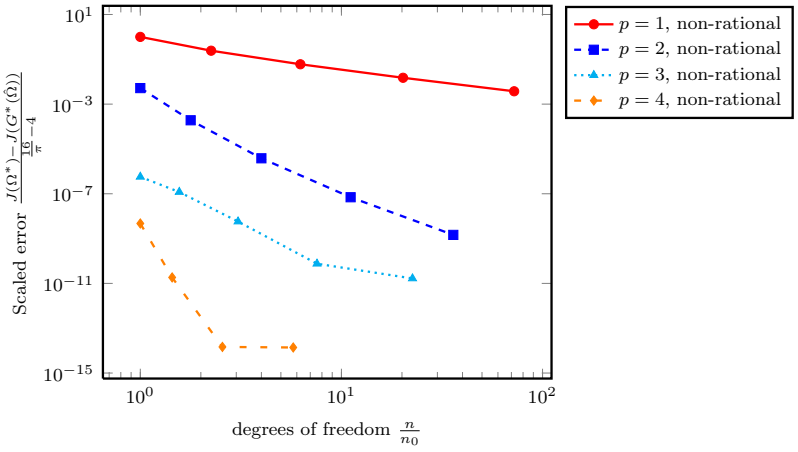
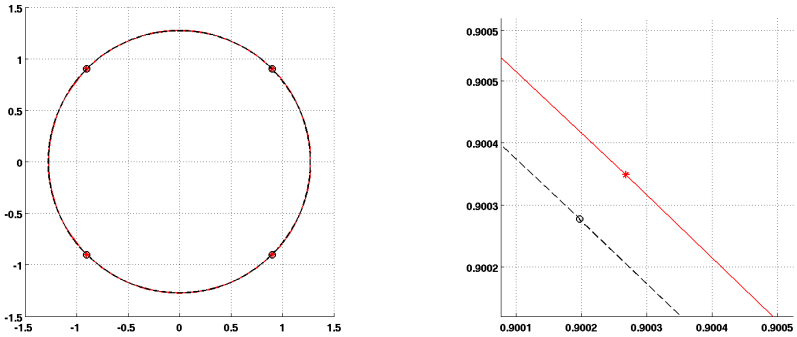


Figure 7: convergence plot for Example 1 where h-refinement corresponds to an increasing number of degrees of freedom n and Ω^* is the known optimal domain and $G^*(\hat{\Omega})$ the numerical optimum by the isogeometric method with *optimize first* scheme. For a better comparison we scale the error by the starting error which is the area of the optimal disk minus area of initial square. Moreover, for each p we also norm the degrees of freedom n by their starting number of degrees of freedom n_0.



(a) **exact** and **numerical** optimum (b) upper right control point

Figure 8: Optimization of control points and weights of Example 1: Optically, a difference can be detected between the **exact disk** and **numerical optimal shape** after zooming in, for instance at the upper right control point

objective, and then deteriorates, Figure 8. One reason for this behavior could stem from the quadrature errors and the rational terms due to the projective map, which leaves room for future work.

We have double-checked the shape gradients of J and C for both, optimization with and without weights, with the MATLAB central finite difference test for gradients where $J'(G, N_i e_k) \approx \frac{1}{2\epsilon}\big(J(G + \epsilon N_i e_k) - J(G - \epsilon N_i e_k)\big)$. These values agree up to the default

tolerance in `fmincon` which means a relative error of less than 10^{-6} in each component of the gradient, but typically we even observed a relative error of less than 10^{-13}. From

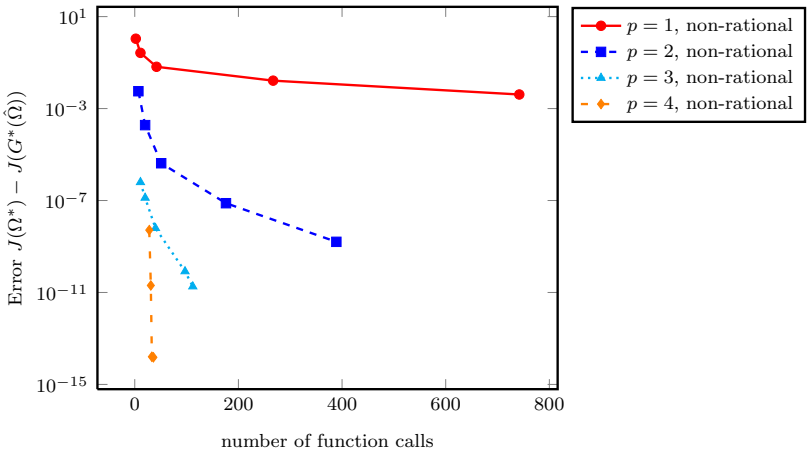


Figure 9: error in terms of computational costs for Example 1, where the costs are expressed as number of function calls by the MATLAB optimization method. The number of function calls is closely related to the number of iterations of the optimizer.

Figure 9 we see that for a higher polynomial degree p, the SQP optimizer in `fmincon` needs significantly less iterations or number of function calls respectively.

Choosing an initial B-spline representation such as in Figure 10a will yield an irregular parameterization for some step sizes, as indicated in Figure 10b. Therefor, to use this configuration for shape optimization, in Example 1 one of the moving mesh algorithms has to be applied. Applying for instance the relative positioning (64), again convergence of the optimization algorithm can be observed, Figure 10c.

5.2. Compliance Minimization in Linear Elasticity

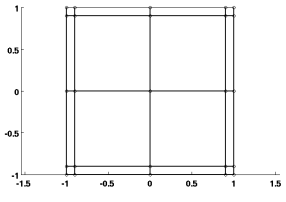
One of the classical shape optimization problems is minimizing the compliance in linear elasticity. In the planar setting, we select $d = m = 2$ and seek for $u \in H^1_{\Gamma_D}(\Omega)^2$ such that

$$\begin{cases} -\operatorname{div}\sigma(u) &= 0 & \text{in } \Omega, \\ u &= 0 & \text{on } \Gamma_D, \\ \sigma(u)\cdot n &= g & \text{on } \Gamma_N, \\ \sigma(u)\cdot n &= 0 & \text{on } \Gamma, \end{cases} \tag{70}$$

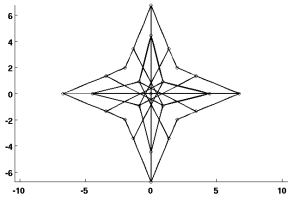
with strain $\varepsilon(u) = \frac{1}{2}(\nabla u + \nabla u^T)$ and stress $\sigma(u) = 2\mu\varepsilon(u) + \lambda(\nabla\cdot u)I_2$.

A specific problem is the plate with circular hole: Given Ω a plate with hole, one tries to find the shape of the hole such that the deformation through external forces, the compliance, is minimized, hence the stiffness of the plate increases. This problem has already been been treated by means of IGA in [7, 6] for the *discretize first* ansatz and in [8] also for the *optimize first* method.

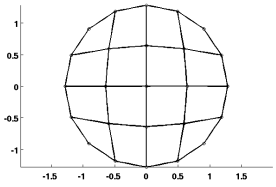
Example 2. We seek to minimize the compliance $\min J(\Omega, u) := \int_{\Gamma_N} g\cdot u\,\mathrm{d}\Gamma$ where u solves (70) under an additional volume constraint on the control, $\int_\Omega \mathrm{d}\Omega = V_0 = \text{const.}$



(a) Initial parameterization



(b) mesh tangling



(c) numerical optimal domain with relative positioning

Figure 10: Example 1 with an arbitrary initial prameterization leads to mesh tangling in some iterations of the SQP optimizer, and therefor to no convergence. However, convergence can be achieved if the moving mesh method (64) is applied.

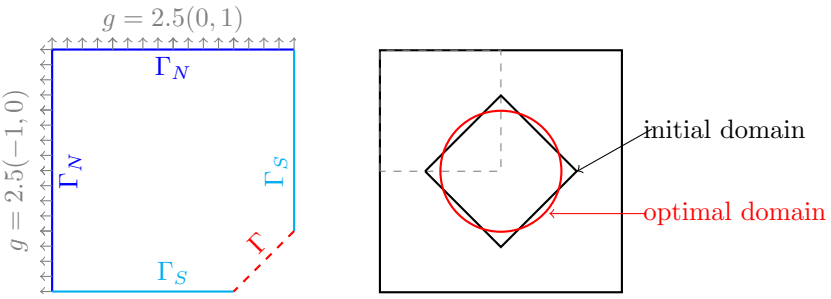
Because the problem is symmetric, we use only a quarter of the plate with symmetric boundary conditions on Γ_S. Existence of optimal shapes for this example has been shown, for instance, in [2]. Typically, one expects a circular hole as optimal solution from calculations in [26, p. 88 ff]. The problem set-up of [6, 7, 8] is displayed in Figure 11.

The shape gradient is given by directional derivatives

$$J'(\Omega; h) = \int_{\Gamma_N} \Big[2\mu |\varepsilon(u)|^2 + \lambda |\operatorname{div} u|^2 \Big] h \cdot n \, \mathrm{d}\Gamma \tag{71}$$

since the compliance minimization is self-adjoint. In this example, the mesh movement of inner control points is realized by means of the linear elasticity operator instead of a Laplacian or relative positioning. Since we have calculated the stiffness matrix for solving the linear elasticity equation anyway, we can recycle it for the mesh movement: the shape gradient yields the right hand side, and also different boundary conditions must be applied, but the bulk of expense from assembling the stiffness matrix has been paid before.

Results. In 4.4 we argued that it is possible to use two representations, one for solving the state equation and one for optimization. However, here we claim that it is even necessary:



(a) boundary conditions (b) initial and optimal shapes

Figure 11: Configuration of state equation of Example 2, as well as initial and optimal shapes

for the compliance, an error in the state u directly influences the cost functional. Suppose u is the exact solution on a domain Ω and u_h its numerical approximation and $e = u - u_h$ the error. Then, setting $u_h = u - e$ the influence of the error with respect to the state in the objective reads

$$J(\Omega, u_h) = \int_{\Gamma_N} g \cdot u \, \mathrm{d}x - \int_{\Gamma_N} g \cdot e \, \mathrm{d}x, \tag{72}$$

which also has an impact on the shape gradient of J. See Figure 12 for a behavior of the compliance for the optimal domain under h-refinement. In Figure 13 we show the initial

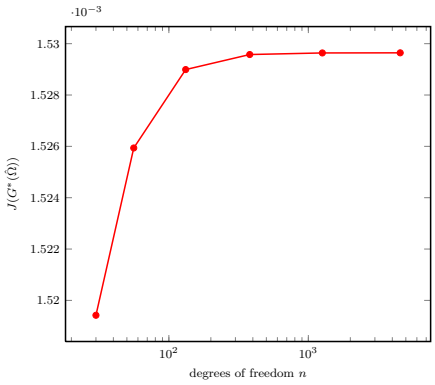
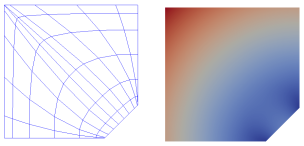


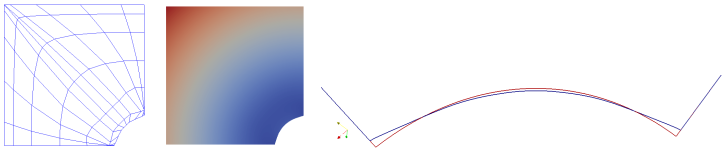
Figure 12: Influence of the numerical error in the PDE on the objective function for Example 2: as the error in the state $u - u_h$ decreases, the compliance, too, is approximated better.

parameterization and results for an optimization run, where we use the same mesh for analysis and optimization. Although the optimizer converges towards a minimum there still is a certain difference to the circular hole as is illustrated in Figure 13c. As long as the mesh for solving the state equation is not fine enough, the optimizer will not be able to further enhance the optimal shape. On the other hand, starting the optimization with

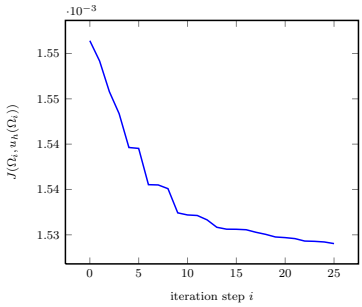
an already refined geometry is much too costly. Compare also the successive refinement strategy in [9].



(a) Initial parameterization and stress analysis for Example 2



(b) Final parameterization and stress analysis for Example 2 after optimization, as well as deviation from the exact optimal shape–a circular hole



(c) Convergence history of optimizer

Figure 13: Optimization run for Example 2

6. Conclusion

We have provided a theoretical framework for shape optimization with isogeometric analysis on an *optimize first* basis, in particular for NURBS geometries, which comes down to finding optimal control points and weights. To do so, one has to work with a representation by homogeneous coordinates. To obtain these derivatives we basically applied the change of variables formula to the existing shape calculus since this quickly enabled us to reuse the main statements. However, from an IGA point of view defining the shape derivatives as Frechét derivatives in the Banach space $\mathcal{G}$ without perturbations of identity but variations of $G \in \mathcal{G}$ might be more elegant. From our perspective, this seems attractive but also more elaborate: basic results, for instance the right metric such that a sequence of domains $\Omega_k \to \Omega^*$ converges to a domain Ω^* if $G_k \to G^*$, have to be re-checked. Such an approach would offer also new options like estimating the error $\Omega_h^* - \Omega^*$ in a suitable norm with the whole approximation theory of B-splines at hand.

From the geometry representation in IGA sensitivity information is also available inside the domain and not on the boundary only, which makes one expect that through this

information a re-meshing of the domain in each iteration step can be avoided. However, we have seen that for large deformations of a shape on the boundary, irregular parameterizations can occur independent of the sensitivity information of the interior basis functions resp. inner control points. This has to be treated by extending the gradient of the boundary to the interior. Although we sketched some possibilities, there appears to be only the choice between expensive methods or risk of failure, i.e., tangled meshes.

Another discussion of trade offs is lead by the observation that the geometry representation for the analysis and for the optimization can be separated, though stemming from the same initial B-spline or NURBS model. For instance it then is possible to use NURBS for domain representation (and optimization) and B-splines for the analysis. On the one hand, in case of two parameterizations we loose some of the tight connections of design and analysis in IGA. On the other hand, to get good results from simulation we need a fine mesh which in IGA implies a fine parameterization. Typically such fine representations of domains are not necessary for shape optimization and only drive the costs of evaluating shape sensitivities and finding an optimal shape. This indicates that two parameterizations are favorable, one for analysis and one for optimization. Of course, this immediately calls for an error estimator that directs the refinement for the state as for the control discretization, combined in a goal-oriented error estimator like [27].

To conclude, there are of course open questions but most of all there is a huge potential of isogeometric analysis for shape optimization.

7. Acknowledgements

We thank Louis Blanchard and Régis Duvigneau from INRIA Sophia Antipolis as well as Utz Wever from Siemens AG, Corporate Technology, for many helpful discussions on the subject.

This work was supported by the European Union within the Project 284981 "TERRIFIC" (7th Framework Program).

References

[1] T. J. R. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (39) (2005) 4135–4195.

[2] J. Haslinger, R. A. E. Mäkinen, Introduction to Shape Optimization: Theory, Approximation, and Computation, Advances in Desgin and Control, siam, 2003.

[3] V. Braibant, C. Fleury, Shape Optimal Design - A CAD-Oriented Formulation, Engineering with Computers 1 (1986) 193–204.

[4] D. Nguyen, Isogeometric Analysis and Shape Optimization in Electromagnetism, Ph.D. thesis, Technical University of Denmark (Feb. 2012).

[5] D. Nguyen, A. Evgrafov, J. Gravesen, Isogeometric shape optimization for electromagnetic scattering problems, Progress in Electromagnetics Research B 45 (2012) 117–146.

[6] W. Wall, M. Frenzel, C. Cyron, Isogeometric structural shape optimization, Computer Methods in Applied Mechanics and Engineering 197 (2008) 2976–2988.

[7] X. Qian, Full analytical sensitivities in NURBS based isogeometric shape optimization, Computer Methods in Applied Mechanics and Engineering 199 (2010) 2059–2071.

[8] L. Blanchard, R. Duvigneau, A.-V. Vuong, B. Simeon, Shape Gradient for Isogeometric Structural Design, Journal of Optimization Theory and Applications (2013) 1–7.

[9] P. Bornemann, F. Cirak, A subdivision-based implementation of the hierarchical b-spline finite element method, Computer Methods in Applied Mechanics and Engineering 253 (2013) 584–598.

[10] J. Kiendl, R. Schmidt, R. Wüchner, K.-U. Bletzinger, Isogeometric shape optimization of shells using semi-analytical sensitivity analysis and sensitivity weighting, Computer Methods in Applied Mechanics and Engineering 274 (2014) 148–167.
[11] O. Pironneau, Optimal Shape Design for Elliptic Systems, Springer, 1983.
[12] J. Sokolowski, J.-P. Zolésio, Introduction to Shape Optimization. Shape Sensitivity Analysis, Vol. 16 of Springer Series in Computational Mathematics, Springer, 1992.
[13] M. C. Delfour, J.-P. Zolésio, Shapes and Geometries. Metrics, Analysis, Differential Calculus, and Optimization, 2nd Edition, Advances in Desgin and Control, siam, 2011.
[14] V. H. Schulz, A Riemannian view on shape optimization, Foundations of Computational Mathematics 14 (3) (2014) 483–501.
[15] G. Allaire, F. Jouve, A level-set method for vibration and multiple loads structural optimization, Computer Methods in Applied Mechanics and Engineering 194 (30–33) (2005) 3269–3290.
[16] M. P. Bendsoe, O. Sigmund, Topology optimization: theory, methods and applications, Springer, 2003.
[17] R. Adams, J. Fournier, Sobolev Spaces, 2nd Edition, Academic Press, 2003.
[18] L. L. Schumaker, Spline Functions: Basic Theory, Wiley & Sons Ltd., 1981.
[19] C. De Boor, A practical guide to splines, Vol. 27, Springer-Verlag New York, 2001.
[20] L. Piegl, W. Tiller, The NURBS book, Springer, 1995.
[21] M. Hinze, R. Pinnau, M. Ulbrich, S. Ulbrich, Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications, Vol. 23 of Mathematical Modelling: Theory and Applications, Springer, 2009.
[22] B. Kiniger, B. Vexler, A priori error estimates for finite element discretizations of a shape optimization problem, ESAIM: Mathematical Modelling and Numerical Analysis 47 (06) (2013) 1733–1763.
[23] K. Eppler, H. Harbrecht, R. Schneider, On convergence in elliptic shape optimization, SIAM Journal on Control and Optimization 46 (1) (2007) 61–83.
[24] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo, Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis, Computer-Aided Design.
[25] C. J. Budd, W. Huang, R. D. Russel, Adaptivity with moving grids, Acta Numerica (2009) 1–131.
[26] S. Timoshenko, J. Goodier, Theory of elasticity, McGraw-Hill, 1951.
[27] R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, Acta Numerica 2001 10 (2001) 1–102.