Practical isogeometric shape optimization: parametrization by means of regularization

Limkilde, Asger; Evgrafov, Anton; Gravesen, Jens; Mantzaflaris, A

Published in:
Journal of Computational Design and Engineering

DOI (link to publication from Publisher):
10.1093/jcde/qwaa093

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Limkilde, A., Evgrafov, A., Gravesen, J., & Mantzaflaris, A. (2021). Practical isogeometric shape optimization: parametrization by means of regularization. *Journal of Computational Design and Engineering*, *8*(2), 547-558. Article qwaa093. https://doi.org/10.1093/jcde/qwaa093

OXFORD CDE

Journal of Computational Design and Engineering, 2021, 8(2), 547–558

doi: 10.1093/jcde/qwaa093
Journal homepage: www.jcde.org

RESEARCH ARTICLE

Practical isogeometric shape optimization: parametrization by means of regularization

A. Limkilde [1], A. Evgrafov[2,*], J. Gravesen[1] and A. Mantzaflaris[3]

[1]Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kgs Lyngby, Denmark; [2]Department of Mathematical Sciences, Aalborg University, 9210 Aalborg, Denmark and [3]Inria Sophia Antipolis, Université Côte d'Azur, 06902 Sophia Antipolis cedex, France

*Corresponding author. E-mail: asgl@dtu.dk

Abstract

Shape optimization based on isogeometric analysis (IGA) has gained popularity in recent years. Performing shape optimization directly over parameters defining the computer-aided design (CAD) geometry, such as the control points of a spline parametrization, opens up the prospect of seamless integration of a shape optimization step into the CAD workflow. One of the challenges when using IGA for shape optimization is that of maintaining a valid geometry parametrization of the interior of the domain during an optimization process, as the shape of the boundary is altered by an optimization algorithm. Existing methods impose constraints on the Jacobian of the parametrization, to guarantee that the parametrization remains valid. The number of such validity constraints quickly becomes intractably large, especially when 3D shape optimization problems are considered. An alternative, and arguably simpler, approach is to formulate the isogeometric shape optimization problem in terms of both the boundary and the interior control points. To ensure a geometric parametrization of sufficient quality, a regularization term, such as the Winslow functional, is added to the objective function of the shape optimization problem. We illustrate the performance of these methods on the optimal design problem of electromagnetic reflectors and compare their performance. Both methods are implemented for multipatch geometries, using the IGA library G+Smo and the optimization library Ipopt. We find that the second approach performs comparably to a state-of-the-art method with respect to both the quality of the found solutions and computational time, while its performance in our experience is more robust for coarse discretizations.

Keywords: isogeometric analysis; shape optimization; parametrizations

1 Introduction

Isogeometric analysis (IGA) introduced in Hughes, Cottrell, and Bazilevs (2005) is a Galerkin method that uses splines to approximate both the geometric domain and the solutions to partial differential equations (PDEs). Splines are commonly used in computer-aided design (CAD) and IGA is an attempt to bridge the gap between simulation and design (Cottrell, Hughes, & Bazilevs, 2009). This makes it beneficial for shape optimization as the optimization can be performed directly over parameters defining the CAD geometry, e.g. the control points of a spline parametrization, and it opens up the prospect of seamless integration of a shape optimization step into the CAD workflow.

One of the key challenges when using IGA in general is that one needs a parametrization of the interior of the physical domain, on which the PDE is posed (Martin, Cohen, & Kirby, 2008; Gravesen, Evgrafov, Nguyen, & Nørtoft, 2012; Hinz, Möller, & Vuik, 2018; Shamanskiy, Gfrerer, Hinz, & Simeon, 2020). This parametrization is used to pull back the weak form of the PDE to the parameter domain where the basis splines

Received: 9 July 2020; **Revised:** 26 November 2020; **Accepted:** 28 November 2020

(B-splines) are defined. The choice of parametrization can affect the accuracy of the resulting IGA discretization (Xu, Mourrain, Duvigneau, & Galligo, 2010; Gravesen *et al.*, 2012) and, at the very least, the parametrization should be valid (a bijective map), i.e., its Jacobian determinant should be nonzero. One approach to constructing a valid parametrization in 2D is to search for the one which has a harmonic inverse. In Hinz, Möller, and Vuik (2018), this property is reformulated as a nonlinear PDE and the parametrization is found by solving this PDE. In Hinz, Jaeschke, Möller, and Vuik (2020), this PDE-based parametrization technique is used for a gradient-based shape optimization algorithm with IGA. In Gravesen *et al.* (2012), Nguyen, Evgrafov, and Gravesen (2012), and Speleers and Manni (2015), the same property is attained by minimizing the Winslow functional (Winslow, 1966). The method can be made more flexible by the use of adaptive splines (Falini, Špeh, & Jüttler, 2015), which allow to enrich the feasible region near to complex boundaries. In the recent works (Sajavičius, Jüttler, & Špeh, 2019; Shamanskiy *et al.*, 2020), the approach of parametrizing a complex domain by deforming a given template is explored. In Pan and Chen (2019), the focus is on producing parametrizations with low rank with respect to the coefficient tensor.

When using IGA for shape optimization, the challenge of finding a valid parametrization is even more important, since the shape of the physical domain changes during an optimization process. This means that a valid parametrization needs to be maintained during this process.

To guarantee that the parametrization remains valid during the optimization process, shape optimizations methods based on IGA often rely on constraints on the Jacobian of the parametrization (Nguyen, Evgrafov, & Gravesen, 2012; Gangl, Langer, Mantzaflaris, & Schneckenleitner, 2020). These can be enforced either by using injectivity cones or by using the spline coefficients of the Jacobian determinant (Xu, Mourrain, Duvigneau, & Galligo, 2011). However, the number of constraints needed quickly becomes very large, especially in 3D. Furthermore, when using the coefficients of the Jacobian determinant for the constraints, as we will do in this work, it may be necessary to expand the Jacobian determinant on a finer spline space, which increases the number of constraints even further.

In this work, we will compare an existing approach to IGA shape optimization, relying upon reparametrizing the domain, with a simple approach to maintaining a valid parametrization without the use of explicit validity constraints. Namely, one lets the positions of all the control points that define the parametrization and the shape of the domain enter the formulation as independent optimization variables and adds a regularization term to drive the optimization toward a design with a valid parametrization. Such an approach has been considered in the context of shape optimization in mechanics in Scherer, Denzer, and Steinmann (2010), and has, to the best of our knowledge, only been considered very briefly in the context of shape optimization with IGA in Friederich, Scherer, and Steinmann (2011). It remains a question whether this approach performs comparably to state-of-the-art methods, and the aim of the work is to investigate exactly this question.

Different frameworks for shape optimization with IGA exist in the literature. For example, in López, Anitescu, and Rabczuk (2020) a tetrahedral mesh is used to represent the interior of the computational domain, while still representing the boundary using splines. The authors rely on external mesh generator software, while we in this work aim to avoid mesh generation by instead maintaining a parametrization of the interior. In Ghasemi, Park, and Rabczuk (2017), the shape is represented implicitly as the level set of a function. The PDE is then posed on a design domain, which remains constant during the optimization, and the level set function enters the formulation by the ersatz material approach. This framework has the advantage that it allows for changes in the topology of the shape; however, at the same time the final shape is represented as a level set and therefore postprocessing is required to represent this shape using splines, which is necessary for importing the shape into CAD software. In our work, the optimization is performed directly on the spline representation and thus the result can be readily imported into CAD software after the optimization. Another possible approach is to use the isogeometric representation of geometry combined with the boundary element method (IGABEM; Liu, Chen, Zhao, & Chen, 2017; Chen *et al.*, 2019). In IGABEM, the PDE is reduced to an integral formulation on the boundary of the domain, and therefore maintaining a parametrization of the interior of the domain is avoided altogether. However, in this case, one has to deal with the standard complications of BEM, in particular the fact that the system matrices are dense, nonsymmetric, and costly to compute. Additionally, the Greens function for the considered PDE has to be known, which is not always the case.

In this work, we illustrate that the simple regularization-based approach is able to handle complicated geometries, by comparing its performance to a shape optimization approach based on using a linearization of the Winslow minimization problem as a parametrization strategy, and employing locally refined splines to represent the Jacobian determinant. The method closely resembles the one in Nguyen, Evgrafov, and Gravesen (2012). The main difference is that we use truncated hierarchical basis splines (THB-splines; Giannelli, Jüttler, & Speleers, 2012), which possess the partition of unity property, to refine the spline space in which the determinant is expanded locally. This reduces the number of constraints needed as compared with tensor product global refinement.

We will apply the two methods and compare their performance on the shape optimization problem of designing electromagnetic reflectors. In this problem, we have two metallic reflectors in a dielectric medium and search for a shape that maximizes the electrical energy close to a chosen point. The same problem has been studied with topology optimization in Aage, Mortensen, and Sigmund (2010), Wadbro and Engström (2015), and Christiansen, Vester-Petersen, Madsen, and Sigmund (2019), and with IGA in Nguyen, Evgrafov, and Gravesen (2012).

The methods are implemented for multipatch geometries, using the IGA library G+Smo (https://github.com/gismo) and the optimization library Ipopt (https://github.com/coin-or/Ipopt). A benchmark study on the performance of different optimization algorithms in the context of structural optimization can be found in Rojas-Labanda and Stolpe (2015). For a review of implementation aspects of IGA in general, we refer the reader to Nguyen, Anitescu, Bordas, and Rabczuk (2015) and for details of the G+Smo library we refer the reader to Jüttler, Langer, Mantzaflaris, Moore, and Zulehner (2014) and Mantzaflaris (2020). The code used for this work can be found at https://github.com/gismo/shapeopt.

The paper is organized as follows: In Section 2, we outline the relevant notation, and in Sections 3 and 4, we describe the two methods we are going to compare. In Section 5, we apply these methods to the aforementioned shape optimization problem, and discuss and compare the performance of the two approaches. We end the paper with some discussion and conclusions. Some of the more technical details are presented in the appendices.

2 Preliminaries and Notation

Let us consider the following PDE-constrained shape optimization problem:

$$\max_{\Omega \in \mathcal{O}_{ad}} E(\Omega, u), \tag{1a}$$

$$\text{s.t. } a_\Omega(u, v) = \ell_\Omega(v) \qquad \text{for all } v \in V, \tag{1b}$$

where $\mathcal{O}_{ad}$ is a set of admissible shapes, E is the objective, and (1b) is the governing PDE in the weak form.

Within the IGA framework, both Ω and u will be approximated numerically using splines. Namely, we have $\Omega = G(]0, 1[^d)$, with the *parametrization*

$$G(\boldsymbol{\xi}) = \sum_{i=1}^{N^g} \mathbf{c}_i R_i^g(\boldsymbol{\xi}), \tag{2}$$

where $\mathbf{c}_i \in \mathbb{R}^d$ are the control points, N^g is the number of control points, and R_i^g are the basis (B) splines. In this work, unless specifically stated otherwise, we will use tensor product B-splines. The superscript g indicates that the B-splines R_i^g (i.e., their degrees and knotvectors) are specific to the geometry representation. Within the shape optimization framework, it will sometimes be necessary to distinguish between *boundary* and *inner control points*. We will therefore introduce the notation

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}^\partial \\ \mathbf{c}^i \end{bmatrix}, \tag{3}$$

where $\mathbf{c}^\partial$ are the boundary control points and $\mathbf{c}^i$ are the inner control points. The Jacobian $J = \frac{\partial G}{\partial \boldsymbol{\xi}}$ will also play an important role in the forthcoming development.

Similarly to (2), we approximate the state of our system as a pulled back spline

$$u_h = \sum_{i=1}^{N} u_i R_i \circ G^{-1}, \tag{4}$$

where R_i^g, $i = 1, \ldots, N$ are B-splines. The expansion coefficients u_i, $i = 1, \ldots, N$ will be found by solving a system of linear algebraic equations

$$K_\mathbf{c}\, \boldsymbol{u} = f_\mathbf{c}.$$

As standard in the Galerkin approach, the elements of the stiffness matrix $K_\mathbf{c}$ and the load vector $f_\mathbf{c}$ are computed as $K_{\mathbf{c},i,j} = a_{G(]0,1[^d)}(R_j \circ G^{-1}, R_i \circ G^{-1})$ and $f_{\mathbf{c},i} = \ell_{G(]0,1[^d)}(R_i \circ G^{-1})$. Note that the dependence of $K_\mathbf{c}$ and $f_\mathbf{c}$ on the control points is encapsulated in (2) and (4).

Already at this point, the importance of the geometry parametrization should be apparent. Indeed, at the very least it should be an invertible map, which is used to pull back the weak form of the PDE defined on the physical domain Ω into the parameter domain $]0, 1[^d$. In particular, for all $\boldsymbol{\xi} \in]0, 1[^d$ it is necessary that $\det(J(\boldsymbol{\xi})) > 0$ (It is equivalent to require that $\det(J(\boldsymbol{\xi})) < 0$; however, in this work we will use the constraint $\det(J(\boldsymbol{\xi})) > 0$.). A sufficient condition, which guarantees the validity of the parametrization, is discussed in Appendix A.

3 Boundary-Driven Approach to IGA Shape Optimization

In this section, we will consider one possible approach to IGA shape optimization, which follows the ideas developed in Gravesen *et al.* (2012) and Nguyen, Evgrafov, and Gravesen (2012). Within this framework, we formulate the optimization problem in terms of boundary control points $\mathbf{c}^\partial$. The collection of interior control points $\mathbf{c}^i$ for the geometry parametrization is treated as an implicit function of $\mathbf{c}^\partial$; see Appendix B. Additionally, the parametrization validity constraints $\det(J) > 0$, or a sufficient condition for these (cf. Appendix A), have to be explicitly included into the problem formulation.

To compute domain parametrizations of high quality, we rely upon minimizing the Winslow functional (Section B.2). However, to avoid solving a nonlinear optimization problem at each shape optimization iteration, we construct a quadratic approximation to the Winslow functional around a reference parametrization, and update the reference parametrization when it becomes necessary to do so. Specifically, given a reference parametrization G_0 defined by the control points $\mathbf{c}_0$, to find a new parametrization, we consider the quadratic programming problem

$$\min_{\Delta \mathbf{c}^i} \frac{1}{2} \Delta \mathbf{c}^T H(\mathbf{c}_0) \Delta \mathbf{c} + \nabla W(\mathbf{c}_0)^T \Delta \mathbf{c} + W(\mathbf{c}_0), \tag{5}$$

where W is the Winslow functional and H is its Hessian. The minimizer of this problem can be found by solving a linear system

$$H(\mathbf{c}_0) \Delta \mathbf{c} = -\nabla W(\mathbf{c}_0).$$

Using (3), this can be restated as

$$H_{\mathbf{c}^i, \mathbf{c}^i} \Delta \mathbf{c}^i = -\nabla W_{\mathbf{c}^i} - H_{\mathbf{c}^i, \mathbf{c}^\partial} \Delta \mathbf{c}^\partial, \tag{6}$$

where $[H_{\mathbf{c}^i, \mathbf{c}^i}]_{ij} = \partial^2 W / \partial \mathbf{c}^i{}_i \partial \mathbf{c}^i{}_j$, $[H_{\mathbf{c}^i, \mathbf{c}^\partial}]_{ij} = \partial^2 W / \partial \mathbf{c}^i{}_i \partial \mathbf{c}^\partial{}_j$, and $[\nabla W_{\mathbf{c}^i}]_i = \partial W / \partial \mathbf{c}^i{}_i$. The new parametrization is then defined by the control points given by $\mathbf{c} = \mathbf{c}_0 + \Delta \mathbf{c}$.

With this in mind, to approximate (1) numerically, we solve a sequence of subproblems

$$\max_{\Delta \mathbf{c}^\partial} \; E(\mathbf{c}, \boldsymbol{u}), \tag{7a}$$

$$\text{s.t. } K_\mathbf{c}\, \boldsymbol{u} = f_\mathbf{c}, \tag{7b}$$

$$H_{\mathbf{c}^i, \mathbf{c}^i} \Delta \mathbf{c}^i = -\nabla W_{\mathbf{c}^i} - H_{\mathbf{c}^i, \mathbf{c}^\partial} \Delta \mathbf{c}^\partial, \tag{7c}$$

$$\mathbf{c} = \mathbf{c}_0 + \Delta \mathbf{c}, \tag{7d}$$

$$\boldsymbol{d} \geq \varepsilon, \tag{7e}$$

$$\mathbf{c}^\partial{}_L \leq \mathbf{c}^\partial \leq \mathbf{c}^\partial{}_U, \tag{7f}$$

where $\mathbf{c}_0$ is the reference parametrization, $\Delta \mathbf{c} = (\Delta \mathbf{c}^\partial, \Delta \mathbf{c}^i)$, and (7e) is the sufficient condition for the validity of the parametrization discussed in Appendix A. Each time we solve the subproblem (7), we update the reference parametrization. In the model problem considered in this work, we saw no further progress after 5–10 reparametrizations.

Reference parametrizations can be computed as follows. We minimize the Winslow functional as described in Appendix B, and check if the sufficient condition $\boldsymbol{d} > 0$ is violated. If it is, then this condition is too strict and should be relaxed. To facilitate this, we refine the spline space $\mathcal{S}_{\det}$ where we compute expansion coefficients $\boldsymbol{d}$ of $\det J$. To reduce the number of constraints resulting from such refinement steps, we utilize local refinement. Specifically, we use truncated hierarchical B-splines (THB-splines) as basis functions. Note that it is important to use the truncated version of hierarchical splines, since the partition of unity property (cf. Giannelli, Jüttler, & Speleers, 2012) implies that the spline control polygon converges locally to function values. Note that other locally refinable splines with this property are available, e.g. polynomial splines over hierarchical T-meshes (Deng *et al.*, 2008) or locally refinable (Johannessen, Kvamsdal, &

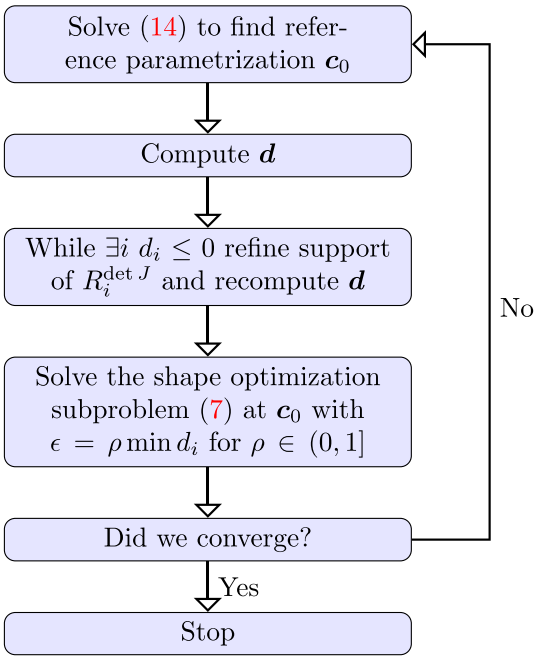


Figure 1: Flowchart of the optimization algorithm.

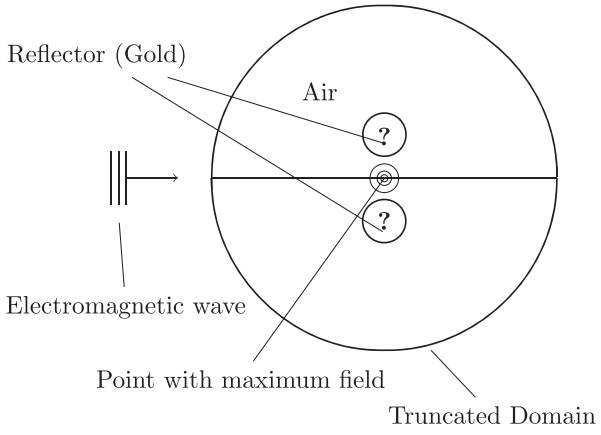


Figure 2: Sketch of the shape optimization problem. The goal is to find a shape of the reflector that maximizes the field close to a point.

Dokken, 2014). For a comparison of these methodologies, we refer the reader to Li, Chen, Kang, and Deng (2016).

The refinement strategy we employ is as follows. For all indices i that have negative spline expansion coefficient $d_i \leq 0$ of $\det J$, we refine the support of the associated basis function $R_i^{\det}$. This is repeated until $d_i > 0$ for all $i = 1, \ldots, N_{\det}$ (In practice, we terminate this procedure either if $\boldsymbol{d} > 0$ or when a maximum level of refinement (7 in our numerical experiments) is attained. The latter termination criterion has not been observed in our experiments.). In the subproblem (7), we then put $\varepsilon = \rho \cdot \min_i d_i$ with $\rho = 0.25$; see (A.1).

The full optimization loop is illustrated in Fig. 1. Note that we have chosen to carry out the spline space refinement described above only when the reference parametrization, and therefore also the subproblem (7), is updated. This allows us to keep the number of constraints constant when solving (7) numerically and therefore employ off-the-shelf optimization software. The initial guess for the nonlinear Winslow optimization problem is generated using Coons' patches; see Appendix B.

4 Regularization-Driven Approach to IGA Shape Optimization

In this section, we discuss an alternative approach to shape optimization using IGA, which does not involve explicit constraints on $\det J$. The positions of the inner control points $\boldsymbol{c}^i$ enter this formulation as independent optimization variables, in the same way as $\boldsymbol{c}^b$. Consequently, we do not need to explicitly compute a domain parametrization, as this will be part of the outcome of the optimization process.

To this end, we add the Winslow functional W as a regularization term to the objective function. Its role is to penalize configurations of control points that result in poor parametrizations. This idea has been used previously in the context of shape optimization in mechanics (Scherer, Denzer, & Steinmann, 2010; Friederich, Scherer, & Steinmann, 2011). Thus, for a regularization parameter $\tau > 0$, we consider the optimization problem

$$\min_{\mathbf{c}} \quad \tau W(\mathbf{c}) - E(\mathbf{c}, \mathbf{u}), \tag{8a}$$

$$\text{s.t.} \quad K_{\mathbf{c}}\, \mathbf{u} = f_{\mathbf{c}}, \tag{8b}$$

$$\mathbf{c}_L \leq \mathbf{c} \leq \mathbf{c}_U. \tag{8c}$$

We put $W(\mathbf{c}) = \infty$ if $\det J \leq 0$ at one of the quadrature points used for the integration when calculating $W(\mathbf{c})$. In this way, when the optimization algorithm enforces, e.g. the standard sufficient decrease condition, such a choice ensures that the chosen step will always have positive determinant at the quadrature points. This does not guarantee that it is positive everywhere, but it means that the numerics will not collapse due to a division by zero.

The regularization parameter τ needs to be tuned for the specific problem at hand. If it is too large, the minimization will find a design with a small value of the Winslow functional but disregarding the objective $E(\mathbf{c}, \mathbf{u})$. If it is too small, the optimization will find positions of the control points that have a low objective $E(\mathbf{c}, \mathbf{u})$, but with a poor parametrization, which might give a large discretization error of the discretized PDE. The appropriate values of τ would lead to a compromise between these two extreme situations. One simple strategy for choosing such a value is to solve a sequence of problems (8) for decreasing values of τ (We should note that the literature on regularization is quite extensive; see e.g. Hansen (1994) and references therein, and this topic is somewhat beyond the scope of this work.).

The two points above constitute the main drawback of this method. Namely that, in contrast to a constraint-based method, we cannot guarantee that the parametrization is valid in between the quadrature points. Additionally, one has to find a suitable value of the regularization parameter τ, which is problem dependent.

5 Case Study: Optimization of Electromagnetic Reflectors

In this section, we will consider a 2D shape optimization problem originating from the field of electromagnetism. Our goal is to design a reflector that concentrates electrical energy in a desired area. This problem will serve as a model problem for comparing the two optimization approaches outlined in Sections 3 and 4.

Table 1: Physical parameters.

f	μ_r	μ_r^s	σ	ϵ_0	μ_0	$\epsilon_{r,\text{gold}}$
$4 \cdot 10^{14}$ [Hz]	1.0	1.0	10^6 [S/m]	$(\mu_0 c^2)^{-1}$	$4\pi 10^{-7}$	$-20.199 + j1.381$

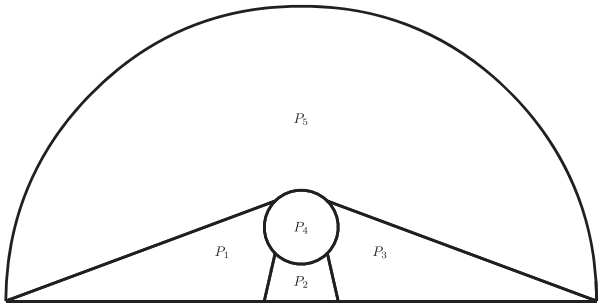


Figure 3: Patch layout.

5.1 Physical model

We consider a 2D scattering problem where a plane wave with frequency f travels in a dielectric (air) and is scattered by two symmetric metallic (gold) reflectors, as depicted in Fig. 2. Let ϵ_{cr} and μ_r denote the complex permittivity and permeability of the medium. Using the first-order absorbing boundary condition (Jin, 1993) at the boundary Γ_t of the truncated domain, the electromagnetic field $\hat{u}$ should satisfy the following PDE:

$$\nabla \cdot \left(\frac{1}{\epsilon_{cr}} \nabla \hat{u}\right) + k_0^2 \mu_r \hat{u} = 0 \qquad \text{in } \Omega, \tag{9a}$$

$$\frac{\partial(\hat{u} - u^i)}{\partial n} + \left(jk_0 + \frac{1}{2r_t}\right)(\hat{u} - u^i) = 0 \qquad \text{on } \Gamma_t. \tag{9b}$$

In the equations above, $k_0 = 2\pi\sqrt{\epsilon_0 \mu_0}$ is the wave number and ϵ_0 and μ_0 refer to the permittivity and permeability, respectively, of free space. The imaginary unit is denoted by j, the radius of the truncated domain is given by r_t, and u^i is the incident plane wave, given by

$$u^i(x, y) = e^{-jk_0\sqrt{\epsilon_{cr}\mu_r}x}.$$

The objective function of the shape optimization will be given by

$$E(\mathbf{c}, u) = \int_\Omega \delta |\hat{u}|^2 dx,$$

where δ is a Gaussian bell function

$$\delta(x, y) = e^{-(x^2+y^2)/(2\alpha^2)},$$

with $\alpha = 0.1$. Thus, we aim to focus the incoming energy in the vicinity of the origin (0, 0). The physical parameters that we use are given in Table 1. The complex permittivity of the reflector is calculated as $\epsilon_{cr}^s = \epsilon_{r,\text{gold}} - j(\sigma/\omega\epsilon_0)$.

The weak statement of the PDE (9) is to find $\hat{u} \in H^1(\Omega)$ such that for all test functions $\hat{v} \in H^1(\Omega)$ the following equality holds:

$$\int_\Omega \frac{1}{\epsilon_{cr}} \hat{\nabla}\hat{u} \cdot \hat{\nabla}\hat{v} dx + k_0^2 \int_\Omega \mu_r \hat{u}\hat{v} dx + \left(jk_0 + \frac{1}{2r_t}\right) \int_{\Gamma_t} \frac{1}{\epsilon_{cr}} \hat{u}\hat{v} ds$$
$$= \frac{1}{\epsilon_{cr}} \int_{\Gamma_t} \left(\frac{\partial u^i}{\partial n} + \left(jk_0 + \frac{1}{2r_t}\right) u^i\right) \hat{v} ds. \tag{10}$$

Due to the symmetry, we only consider the upper half of the geometry shown in Fig. 2. To accommodate the change of material parameters between the metallic reflector and the surrounding dielectric medium, we will split the domain into five patches, one for the reflector and the other four for the surrounding air. The layout is shown in Fig. 3 (For automatic generation of patch layouts, the interested reader is referred to Zheng, Pan, and Chen (2019).). Each patch is parametrized using splines of degree $p = 2$ as described in Section 2. We will use strong patch coupling to enforce C^0 continuity at the patch interfaces, while noting that other alternatives exist, e.g. weak coupling (Hu, Chouly, Hu, Cheng, & Bordas, 2018) and strong C^1 coupling (Chan, Anitescu, & Rabczuk, 2019).

The shape of the reflector is represented via the patch interfaces, which consist of four spline curves with C^0 continuity at the four corners. Such a description, for example, allows shapes that are only piecewise smooth such as the classical bowtie antenna (Compton *et al.*, 1987).

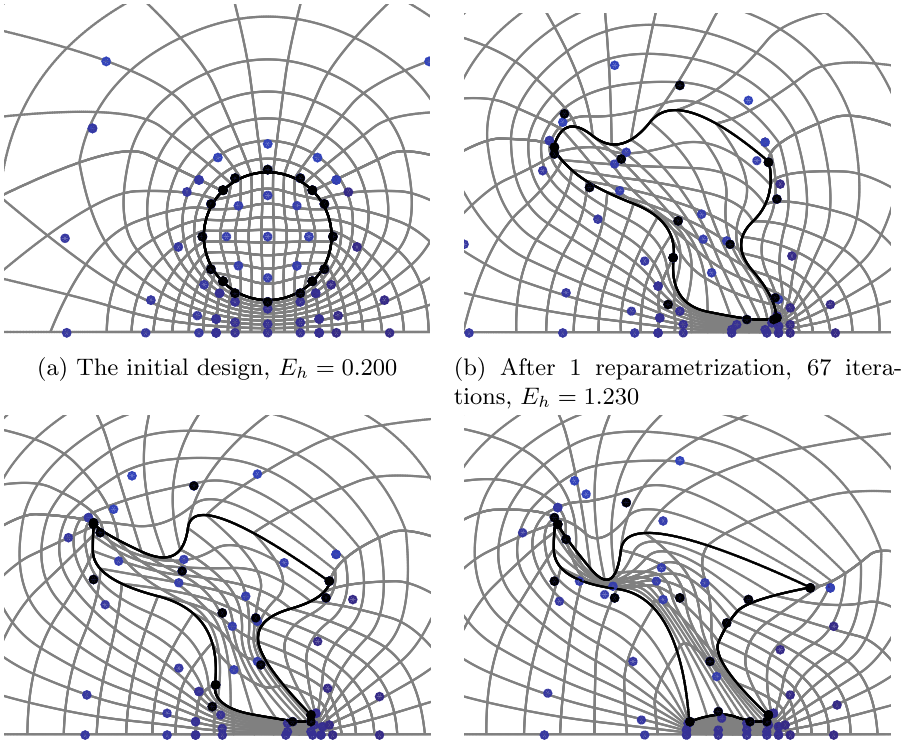
Using the parametrizations, we can pull back the equation (10) to the parameter domain and apply the Galerkin method to it, which ultimately results in a system of linear algebraic equations; see Appendix C for details. To evaluate the integrals involved, we use element-wise Gauss–Legendre quadrature, owing to their immediate availability in the G+Smo library, while noting that more efficient alternatives exist (Bartoň & Calo, 2017; Calabro, Sangalli, & Tani, 2017).

5.2 Results with boundary-driven approach

In this section, we apply the method described in Section 3 to our model problem. We start with an initial design where the reflector has the shape of a circle. We consider two different spline spaces in which to approximate the PDE (9), namely where the knotvectors used for representing the geometry are refined uniformly either 3 or 4 times (Note that while we here use tensor product spline spaces for the approximation of the solution to the PDE, adaptivemethods that use local refinement exist; see e.g. Giannelli, Jüttler, and Speleers (2012).). Both spline spaces have degree $p = 2$, and the numbers of degrees of freedoms are $N_{\text{coarse}} = 2548$ and $N_{\text{fine}} = 9300$, respectively. We will refer to these as the coarse and fine meshes.

We will use a tolerance $\texttt{tol} = 10^{-3}$ when solving the subproblems (7) and a fixed number of reparametrizations, namely 10 when using the coarse mesh and 5 when using the fine mesh. We observed that using more reparametrizations did not lead to significant improvements in the design. In our experiments, the results with this method are sensitive to the number of quadrature points used when calculating the Winslow functional. To produce the results presented here, we use 12 quadrature points per knot interval, to avoid underintegration (In the IGA formulation (C3), we also integrate nonpolynomials. We use three quadrature points for the mass matrix M and seven quadrature points for the stiffness matrix K.).

In our implementation, we use the interior point solver Ipopt to solve the subproblems (7). As we solve a sequence of subproblems, a warm start is available, namely the minimizer from the previous subproblem; so an interior point algorithm might not be the best choice of optimization algorithm (John &



(a) The initial design, $E_h = 0.200$

(b) After 1 reparametrization, 67 iterations, $E_h = 1.230$

(c) After 2 reparametrizations, 143 iterations, $E_h = 1.484$

(d) After 10 reparametrizations, 579 iterations, $E_h = 1.803$

Figure 4: The designs at different stages of the optimization process, when using validity constraints and the coarse mesh. The reflector is outlined with a black line, and the control points of this boundary are colored black. The grey lines are parameter lines mapped with the geometry map, to illustrate the parametrization.

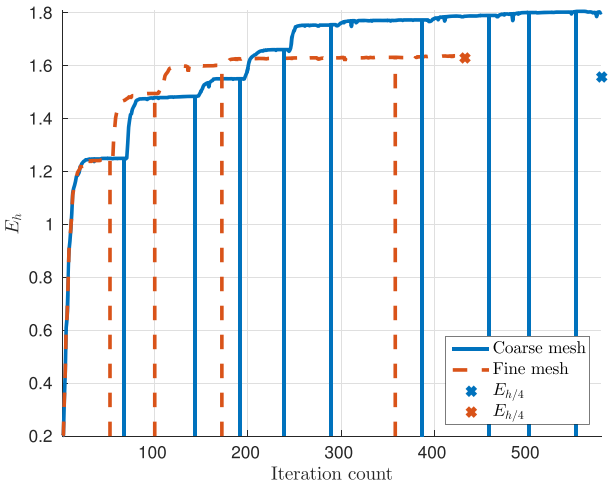
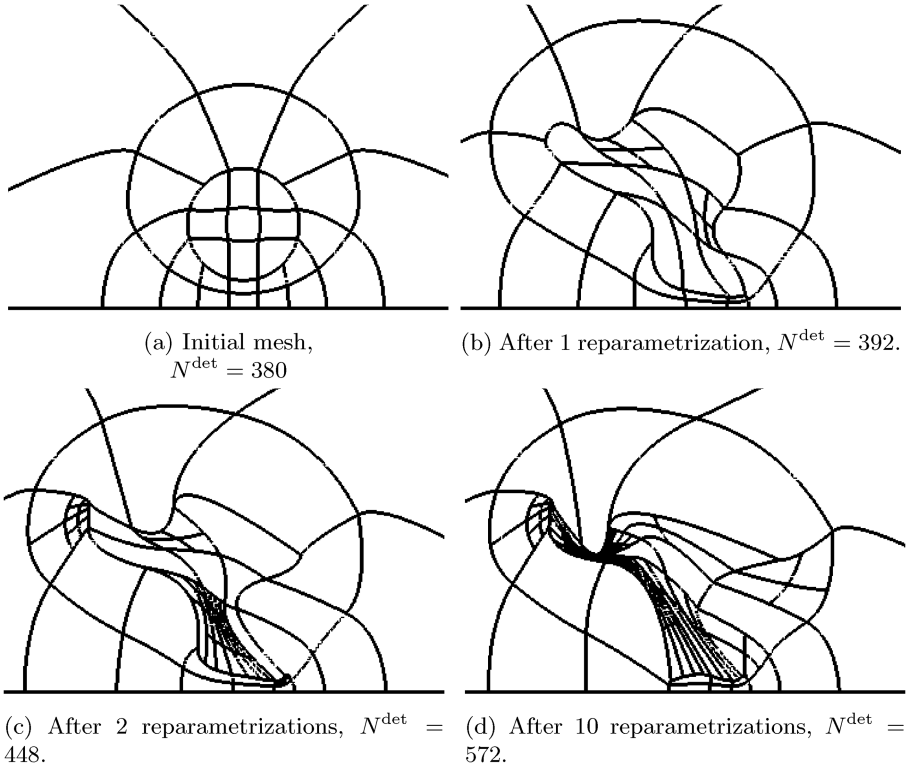


Figure 5: The objective function during optimization process for the coarse and fine meshes. The vertical lines indicate where the parametrization is updated. $E_{h/4}$ is the objective calculated on a refined mesh.

Yıldırım, 2008). However, with the following parameter tuning, we found the method to work well with Ipopt. Namely, after solving one of the subproblems (7) some of the design bounds (7f) will be active; however, since Ipopt is an interior point algorithm, the starting point for the subsequent subproblem will be pushed away from the boundary as controlled by the parameter `bound_push`. We found that this parameter needs to be lower than the default value since the constraints on $\det J$ are quite sensitive, and a relatively small perturbation of the control points might violate these constraints, which is undesirable. The default value is 0.1, but in the experiments, we set it to 10^{-5} instead.

Another key parameter is the barrier parameter `mu_init`. Specifically, we use the `monotone` strategy, where the barrier parameter is monotonically decreased as the optimization algorithm progresses. However, if this parameter is too large in the beginning of the algorithm we found that it will push the design toward configurations with large $\det J$. To remedy this, instead of the default value 0.1 we use 10^{-4}. For more information about the optimization algorithm implemented in Ipopt and its parameters, see Wächter and Biegler (2006).

In Fig. 4, the design at different stages of the optimization is presented, when using the coarse mesh. We observe that the design becomes increasingly hard to parametrize as the objective increases each time we change the reference parametrization. In Fig. 5, the objective is plotted against the number of iterations. We see that the objective function increases after the reference parametrization is changed, but relatively quickly reaches a plateau. We already use a fairly large tolerance of 10^{-3} for the stopping criterion when solving the subproblems (7); however, this behavior indicates that it might help to relax the stopping criterion even further in these subproblems to improve the overall efficiency of the method. However, to allow for a fair comparison between the two methods, we do not investigate this further and use the same tolerance for both methods. The final objective, after 10 reparametrizations when using the coarse mesh, is $E_h = 1.803$. However, if we calculate the objective with a mesh that is refined uniformly twice, we get $E_{h/4} = 1.556$, i.e., a 16% difference. This means that there is actually a significant



(a) Initial mesh, $N^{\det} = 380$

(b) After 1 reparametrization, $N^{\det} = 392$.

(c) After 2 reparametrizations, $N^{\det} = 448$.

(d) After 10 reparametrizations, $N^{\det} = 572$.

Figure 6: The mesh (knot lines), used for representing $\det J$ during the subproblems (7) for the constraints (A.1). The number of constraints is given by $N^{\det}$, and the designs are obtained using the coarse mesh.

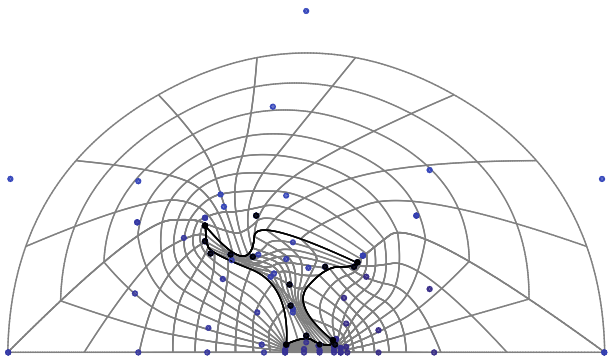


Figure 7: The final design using validity constraints and the fine mesh, $E_{h/2} = 1.638$.

discretization error at this refinement level, which leads to an artificially large objective value.

As described in Section 3, we use local refinement to adapt the constraints on $\det J$ to the current design. This is done by using local refinements in the areas where the spline coefficients are nonpositive. The resulting meshes are plotted in Fig. 6 to illustrate where the refinement is needed. We see that local refinement is utilized primarily inside the reflector and near the reflector–air interface. We see that the number of constraints increases during the optimization process, but by no more than a factor of 2. If we were to use uniform refinement, the number of constraints would increase by more than a factor of 9.

The final design, after five reparametrizations, when using the fine mesh, is shown in Fig. 7. We see that it is similar to the design obtained using the coarse mesh. The final objective here is $E_h = 1.638$. After the mesh is refined uniformly, the electrical energy is $E_{h/4} = 1.628$, which is only a 0.6% difference. The objective function during the optimization process is plotted in Fig. 5. We see that the algorithm converges faster when using the fine mesh as no progress was observed after five reparametrizations.

5.3 Results with regularization-driven approach

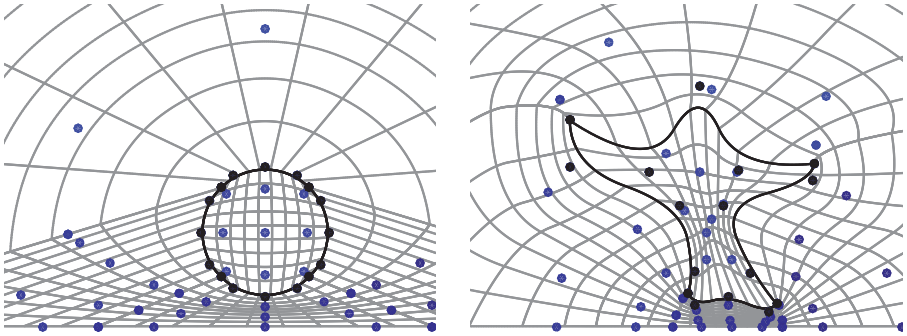
In this section, we will present the results obtained with the method described in Section 4. With this method, we perform the optimization with all control points as optimization variables while using the Winslow functional as a regularization term. Again, we use the interior point solver Ipopt with a tolerance of $\texttt{tol} = 10^{-3}$ for solving the problem (8). The regularization parameter is set to $\tau = \frac{1}{8}$.

One can compare the design evolution shown in Fig. 8 with those obtained previously; see Fig. 4. The designs obtained using the regularization approach seem to have more regular parametrizations compared to those in Fig. 4.

The final objective is $E_h = 1.684$. If we calculate the electrical energy for this design on a twice refined mesh, we get $E_{h/4} = 1.546$, i.e., a difference of 9% . This is less than the 16% we observed when using the boundary-driven method. This increase in accuracy might be due to the parametrization being of higher quality.

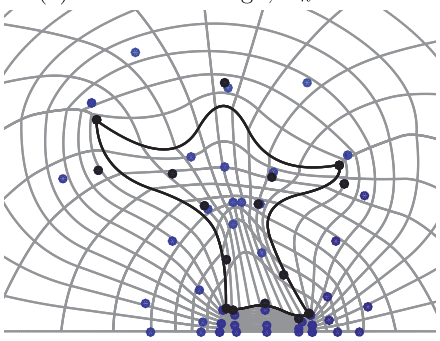
In Fig. 9, we plot the electric energy E_h, the regularization term τW, and the objective function $\tau W - E_h$.

When using the method with the fine mesh, we get the final design shown in Fig. 10. We see that the shape of the reflector is very similar to the one shown in Fig. 8d. The main difference is that when using the fine mesh, the parametrization is more regular, since the error in the discretization of the PDE is

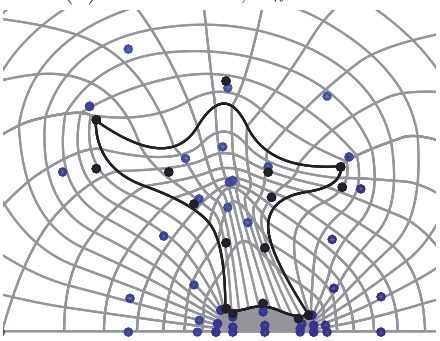


(a) The initial design, $E_h = 0.202$

(b) 67 iterations, $E_h = 1.461$



(c) 143 iterations, $E_h = 1.593$



(d) The final design after 432 iterations , $E_h = 1.683$

Figure 8: The designs at different stages of the optimization process when using the regularization approach. The reflector is outlined with a black line, and the control points that control this boundary are colored black. The grey lines are parameter lines mapped with the geometry map, to illustrate the parametrization.

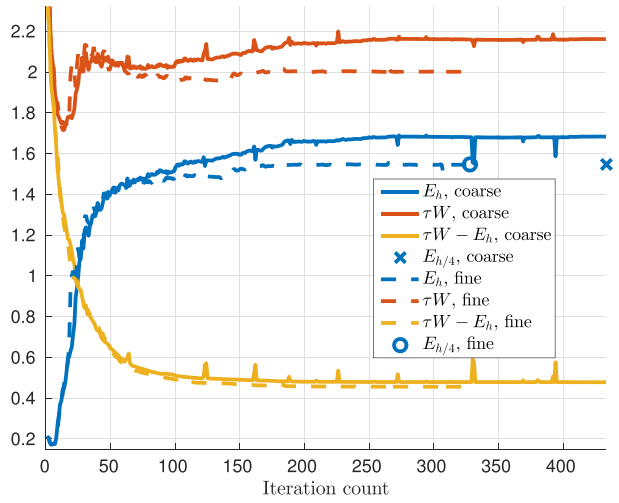


Figure 9: The objective function during the optimization process when using the regularization-based approach, with the fine and coarse mesh.

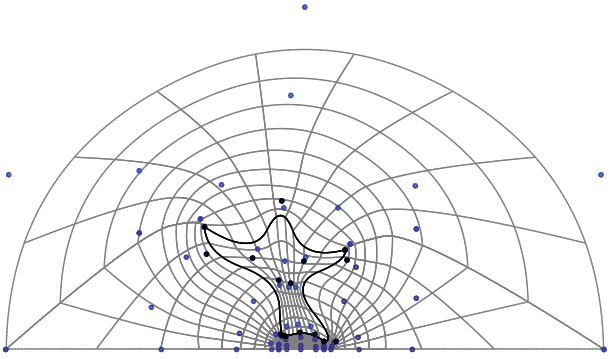


Figure 10: The final design when using the regularization approach and the fine mesh, $E_{h/2} = 1.545$.

smaller, and therefore the optimization cannot exploit it to the same extent. This is especially notable at the bottom of the reflector where the inner control points were moved away from the point of interest when using the coarse mesh, as seen in Fig. 8d. The final objective is $E_h = 1.545$, and when evaluating it on a refined mesh, we get the same result $E_{h/4} = 1.545$ with the difference at the fifth digit. The objective during the optimization process is plotted in Fig. 9. The behavior is similar for the two meshes; however, the tolerance $\texttt{tol} = 10^{-3}$ is reached with fewer iterations when using the fine mesh.

5.4 Comparison and discussion

In Table 2, we summarize the performance of the two methods. We report the objective after a fixed number of iterations, in this case, after 100 iterations, the final objective computed on 3 different refinement levels and the average running time per iteration.

We observe that the average execution time per iteration is the same order of magnitude for the two methods. The main portion of the running times is spent on solving the state equation and computing the gradient of the objective function. The difference in the running time that we observed between the two methods might be due to a different number of function evaluations per iteration needed for trial steps of the algorithm. If we were to consider a larger problem, e.g. in 3D, the large number

Table 2: Comparison of the two methods from Sections 3 and 4 on two different meshes. $E_{\text{iter}=100}$ is the objective after 100 iterations. E_h is the objective computed on the mesh used in the optimization. $E_{h/2}$ and $E_{h/4}$ are the objectives computed after refining the mesh uniformly, once and twice. The execution time was measured on a 64-bit HP EliteBook 840 G4 with an Intel(R) Core(TM) i7-7500U CPU, with a clock rate of 2.70 GHz.

Method	$E_{\text{iter}=100}$	E_h	$E_{h/2}$	$E_{h/4}$	$\frac{E_h - E_{h/4}}{E_{h/4}}$	Avg time per iteration
Linearizations coarse mesh	1.482	1.803	1.589	1.556	16%	9.68 second
Regularization coarse mesh	1.529	1.684	1.561	1.546	9%	11.54 second
Linearizations fine mesh	1.495	1.638	1.629	1.628	0.6%	53.5 second
Regularization fine mesh	1.482	1.545	1.545	1.545	0.003%	71.0 second

of validity constraints would likely lead to an increase in running time for the boundary-driven approach. For the boundary-driven approach, the computation of reference parametrizations by minimizing the Winslow functional only accounts for 2.4% and 0.25% of the total running time for the coarse and the fine grid, respectively.

Regarding the quality of the designs we find, we note that shape optimization problems are prone to having many local optima, so it could be that the two methods find two different local optima. Therefore, it can be futile to directly compare objective values. That being said, we observe that the boundary-driven approach happens to find solutions with slightly higher (better) objective value. On the other hand, the regularization-based approach seems to estimate the objective value more accurately on coarser meshes, since our results were more reliable using this method, probably due to the better quality of the parametrizations it produced.

Also, observe that with the regularization-based approach, we only need the objective E_h, the Winslow functional, and their first-order derivatives. In addition, we can solve a single optimization problem with design bounds as the only constraints. This means that the method is significantly easier to implement.

5.5 Conclusion

We described and compared two methods for shape optimization on spline-based representations. One uses validity constraints to enforce the validity of the geometry parametrization. The other uses a regularization term, and thus avoids both the validity constraints and the need for an explicit parametrization strategy altogether. We demonstrated how this simple approach performed comparably to the more complicated approach in terms of the final design, while requiring similar running times for the 2D problem we considered. The regularization-based approach seems to produce more reliable results, and it is, in addition, much simpler to implement, since we only need the objective, the Winslow functional, and their first-order derivatives.

These results are encouraging, and we plan to use the regularization-based approach for 3D problems, where we expect that the efficiency advantages of the regularization approach will be more prominent, since the number of validity constraints for 3D parametrizations grows quickly. For obtaining competent overall running times, one can couple the approach with low-rank approximation techniques, which have been recently introduced in IGA (Mantzaflaris, Juettler, Khoromskij, & Langer, 2017; Scholz, Mantzaflaris, & Juettler, 2018) and have proved efficient in the frame of PDE-constrained optimization (Bünger, Dolgov, & Stoll, 2020).

Conflict of interest statement

None declared

References

Aage, N., Mortensen, N., & Sigmund, O. (2010). Topology optimization of metallic devices for microwave applications. *International Journal for Numerical Methods in Engineering*, *83*(2), 228–248.

Bartoň, M., & Calo, V. M. (2017). Gauss–Galerkin quadrature rules for quadratic and cubic spline spaces and their application to isogeometric analysis. *Computer-Aided Design*, *82*, 57–67.

Bünger, A., Dolgov, S., & Stoll, M. (2020). A low-rank tensor method for PDE-constrained optimization with isogeometric analysis. *SIAM Journal on Scientific Computing*, *42*(1), A140–A161.

Calabro, F., Sangalli, G., & Tani, M. (2017). Fast formation of isogeometric Galerkin matrices by weighted quadrature. *Computer Methods in Applied Mechanics and Engineering*, *316*, 606–622.

Chan, C. L., Anitescu, C., & Rabczuk, T. (2019). Strong multipatch C1-coupling for isogeometric analysis on 2D and 3D domains. *Computer Methods in Applied Mechanics and Engineering*, *357*, 112599.

Chen, L., Lian, H., Liu, Z., Chen, H., Atroshchenko, E., & Bordas, S. (2019). Structural shape optimization of three dimensional acoustic problems with isogeometric boundary element methods. *Computer Methods in Applied Mechanics and Engineering*, *355*, 926–951.

Christiansen, R. E., Vester-Petersen, J., Madsen, S. P., & Sigmund, O. (2019). A non-linear material interpolation for design of metallic nano-particles using topology optimization. *Computer Methods in Applied Mechanics and Engineering*, *343*, 23–39.

Compton, R., McPhedran, R., Popovic, Z., Rebeiz, G., Tong, P., & Rutledge, D. (1987). Bow-tie antennas on a dielectric half-space: Theory and experiment. *IEEE Transactions on Antennas and Propagation*, *35*(6), 622–631.

Cottrell, J. A., Hughes, T. J., & Bazilevs, Y. (2009). *Isogeometric analysis: Toward integration of CAD and FEA*. John Wiley & Sons.

Deng, J., Chen, F., Li, X., Hu, C., Tong, W., Yang, Z., & Feng, Y. (2008). Polynomial splines over hierarchical T-meshes. *Graphical Models*, *70*(4), 76–86.

Falini, A., Špeh, J., & Jüttler, B. (2015). Planar domain parameterization with THB-splines. *Computer Aided Geometric Design*, *35–36*, 95–108.

Farin, G., & Hansford, D. (1999). Discrete Coons patches. *Computer Aided Geometric Design*, *16*(7), 691–700.

Friederich, J., Scherer, M., & Steinmann, P. (2011). Isogeometric structural shape optimization using a fictitious energy regularization. *PAMM*, *11*(1), 709–710.

Gangl, P., Langer, U., Mantzaflaris, A., & Schneckenleitner, R. (2020). Isogeometric simulation and shape optimization with applications to electrical machines. In *Scientific computing in electrical engineering*(pp. 35–43). Springer International Publishing.

Ghasemi, H., Park, H. S., & Rabczuk, T. (2017). A level-set based IGA formulation for topology optimization of flexoelectric

materials. *Computer Methods in Applied Mechanics and Engineering*, *313*, 239–258.
Giannelli, C., Jüttler, B., & Speleers, H. (2012). THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, *29*(7), 485–498.
Gravesen, J., Evgrafov, A., Nguyen, D.-M., & Nørtoft, P. (2012). Planar parametrization in isogeometric analysis. In *International Conference on Mathematical Methods for Curves and Surfaces*(pp. 189–212). Springer.
Hansen, P. C. (1994). Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems. *Numerical Algorithms*, *6*(1), 1–35.
Hinz, J., Möller, M., & Vuik, C. (2018). Elliptic grid generation techniques in the framework of isogeometric analysis applications. *Computer Aided Geometric Design*, *65*, 48–75.
Hinz, J., Jaeschke, A., Möller, M., & Vuik, C. (2020). The role of PDE-based parameterization techniques in gradient-based IGA shape optimization applications. preprint arXiv:2001.10921.
Hu, Q., Chouly, F., Hu, P., Cheng, G., & Bordas, S. P. (2018). Skew-symmetric Nitsche's formulation in isogeometric analysis: Dirichlet and symmetry conditions, patch coupling and frictionless contact. *Computer Methods in Applied Mechanics and Engineering*, *341*, 188–220.
Hughes, T., Cottrell, J., & Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, *194*(39–41), 4135–4195.
Jin, J.-M. (1993). *The finite element method in electromagnetics*. Wiley.
Johannessen, K. A., Kvamsdal, T., & Dokken, T. (2014). Isogeometric analysis using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, *269*, 471–514.
John, E., & Yıldırım, E. A. (2008). Implementation of warm-start strategies in interior-point methods for linear programming in fixed dimension. *Computational Optimization and Applications*, *41*(2), 151–183.
Jüttler, B., Langer, U., Mantzaflaris, A., Moore, S. E., & Zulehner, W. (2014). Geometry+simulation modules: Implementing isogeometric analysis. *PAMM*, *14*(1), 961–962.
Li, X., Chen, F., Kang, H., & Deng, J. (2016). A survey on the local refinable splines. *Science China Mathematics*, *59*(4), 617–644.
Liu, C., Chen, L., Zhao, W., & Chen, H. (2017). Shape optimization of sound barrier using an isogeometric fast multipole boundary element method in two dimensions. *Engineering Analysis with Boundary Elements*, *85*, 142–157.
López, J., Anitescu, C., & Rabczuk, T. (2020). CAD-compatible structural shape optimization with a movable Bézier tetrahedral mesh. *Computer Methods in Applied Mechanics and Engineering*, *367*, 113066.
Mantzaflaris, A. (2020). An overview of geometry plus simulation modules. In *Mathematical aspects of computer and information sciences*(pp. 453–456). Springer International Publishing.
Mantzaflaris, A., Juettler, B., Khoromskij, B. N., & Langer, U. (2017). Low rank tensor methods in Galerkin-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, *316*, 1062–1085, Special Issue on Isogeometric Analysis: Progress and Challenges.
Martin, T., Cohen, E., & Kirby, M. (2008). Volumetric parameterization and trivariate B-spline fitting using harmonic functions. In *Proceedings of the 2008 ACM Symposium on Solid and Physical Modeling*(pp. 269–280).
Nguyen, D., Evgrafov, A., & Gravesen, J. (2012). Isogeometric shape optimization for electromagnetic scattering problems. *Progress in Electromagnetics Research B*, *45*, 117–146.
Nguyen, V. P., Anitescu, C., Bordas, S. P., & Rabczuk, T. (2015). Isogeometric analysis: An overview and computer implementation aspects. *Mathematics and Computers in Simulation*, *117*, 89–116.
Pan, M., & Chen, F. (2019). Low-rank parameterization of volumetric domains for isogeometric analysis. *Computer-Aided Design*, *114*, 82–90.
Rojas-Labanda, S., & Stolpe, M. (2015). Benchmarking optimization solvers for structural topology optimization. *Structural and Multidisciplinary Optimization*, *52*(3), 527–547.
Sajavičius, S., Jüttler, B., & Špeh, J. (2019). *Template mapping using adaptive splines and optimization of the parameterization*(pp. 217–238). Springer International Publishing.
Scherer, M., Denzer, R., & Steinmann, P. (2010). A fictitious energy approach for shape optimization. *International Journal for Numerical Methods in Engineering*, *82*(3), 269–302.
Scholz, F., Mantzaflaris, A., & Juettler, B. (2018). Partial tensor decomposition for decoupling isogeometric Galerkin discretizations. *Computer Methods in Applied Mechanics and Engineering*, *336*, 485–506.
Shamanskiy, A., Gfrerer, M. H., Hinz, J., & Simeon, B. (2020). Isogeometric parametrization inspired by large elastic deformation. *Computer Methods in Applied Mechanics and Engineering*, *363*, 112920.
Speleers, H., & Manni, C. (2015). Optimizing domain parameterization in isogeometric analysis based on Powell–Sabin splines. *Journal of Computational and Applied Mathematics*, *289*, 68–86.
Wächter, A., & Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, *106*(1), 25–57.
Wadbro, E., & Engström, C. (2015). Topology and shape optimization of plasmonic nano-antennas. *Computer Methods in Applied Mechanics and Engineering*, *293*, 155–169.
Winslow, A. M. (1966). Numerical solution of the quasilinear Poisson equation in a nonuniform triangle mesh. *Journal of Computational Physics*, *1*(2), 149–172.
Xu, G., Mourrain, B., Duvigneau, R., & Galligo, A. (2010). Optimal analysis-aware parameterization of computational domain in isogeometric analysis. In *International Conference on Geometric Modeling and Processing*(pp. 236–254). Springer.
Xu, G., Mourrain, B., Duvigneau, R., & Galligo, A. (2011). Parameterization of computational domain in isogeometric analysis: Methods and comparison. *Computer Methods in Applied Mechanics and Engineering*, *200*(23–24), 2021–2031.
Xu, G., Mourrain, B., Duvigneau, R., & Galligo, A. (2013). Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method. *Journal of Computational Physics*, *252*, 275–289.
Zheng, Y., Pan, M., & Chen, F. (2019). Boundary correspondence of planar domains for isogeometric analysis based on optimal mass transport. *Computer-Aided Design*, *114*, 28–36.

Appendix 1: A Sufficient Condition for A Valid Parametrization

As pointwise constraints $\det(J) > 0$ are, generally speaking, difficult to deal with, we utilize the spline nature of the geometry parametrization (2). Namely, when the geometry map $G \in C^k$ is a spline of degree p, then $\det J \in C^{k-1}$ is a spline with degree $d \cdot p - 1$. This means that we can construct a spline space $\mathcal{S}_{\det}$ that contains $\det J$ (Specifically, the smallest spline space containing $\det J$ can be obtained by increasing themultiplicity of each knot in geometrymap knotvector by $(d-1) \cdot p$ to

account for the reduction in the differentiability and increase of the degree.). We can therefore find the expansion coefficients $\boldsymbol{d}$ of $\det J$ with respect to B-splines in this space (This could be done using either interpolation or L^2 projection. In either approach, one has to solve a linear system to obtain the expansion coefficients. The linear system matrix needs to be inverted only once for a given spline basis and, moreover, the matrix has a Kronecker/separable structure (cf. Mantzaflaris et al., 2017); therefore, the solution can be obtained extremely fast.).

Now we can use this expansion to derive a sufficient condition to replace the pointwise parametrization validity constraints $\det(J) > 0$, by requiring that

$$\boldsymbol{d} \geq \varepsilon, \tag{A.1}$$

where ε is a small positive algorithmic parameter. This condition guarantees that $\det J > 0$ since B-splines are nonnegative and form a partition of unity. This is not a necessary condition, meaning that we might have $d_i \leq 0$ for some i even though $\det J(\boldsymbol{\xi}) > 0$, $\forall \boldsymbol{\xi} \in]0,1[^d$. However, if the spline space $\mathcal{S}_{\det}$ is refined, then the spline expansion coefficients will move closer to values of the spline. So if ε is small enough and $\det J > 0$ then the constraint is likely to be satisfied for a sufficiently refined spline space $\mathcal{S}_{\det}$.

Appendix 2: Domain Parametrization Techniques

In this section, we review some techniques for finding a parametrization of the interior given the boundary. In IGA, this comes down to finding the position of the inner control points given boundary control points. We do not aim to give a thorough review of all the techniques that are available, as this is out of the scope of this work. We will only introduce the methods that are related, or directly used, by the two shape optimization methods considered in this work.

B.1 Coons patch and spring method

Two simple methods for constructing a grid of control points are the Coons patch (Farin & Hansford, 1999) and the spring method (Gravesen *et al.*, 2012). They both produce inner control points that depend linearly on the boundary control points; however, they only produce valid parametrizations for geometries that are not too complicated. In this work, we will use these methods for finding an initial guess for the optimization-based approach that is described below.

B.2 Optimization-based techniques

A more complex, and in general nonlinear, class of parametrization methods consists of optimization-based methods. Here, the geometry map G is chosen such that it minimizes a quality metric $w(\boldsymbol{\xi})$

$$\min_G \int_{]0,1[^d} w(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}, \tag{B.1}$$

$$\text{s.t. } G|_{\partial[0,1]^d} = \gamma, \tag{B.2}$$

where γ is a given boundary curve.

There are several different quality metrics to choose from (cf. Gravesen *et al.*, 2012; Xu, Mourrain, Duvigneau, & Galligo, 2013). In this work, we will consider the Winslow functional, which is given in terms of the Jacobian matrix J as

$$W = \int_{]0,1[^d} w(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}, \tag{B.3}$$

with

$$w = \frac{\operatorname{tr}(J^T J)}{\det J}. \tag{B.4}$$

In 2D, the Winslow functional has the nice property that its minimizer has a harmonic inverse (Gravesen *et al.*, 2012). This guarantees that the minimizer of (B.1) is unique and bijective, i.e., $\det J \neq 0$. It should be noted that this minimizer is not necessarily a spline, so looking for a spline parametrization on the form (2) by minimizing (B.1) with the Winslow functional will only guarantee a valid parametrization if the spline space used for the parametrization has high enough resolution.

Within the shape optimization context, we also need first- and second-order partial derivatives of w. The derivative with respect to a parameter α is given by

$$\frac{\partial w}{\partial \alpha} = 2(\det J)^{-1} \operatorname{tr}\left(J^T \frac{\partial J}{\partial \alpha}\right) - \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \alpha}\right) \frac{\operatorname{tr}(J^T J)}{\det J}, \tag{B.5}$$

where we used the relation $\frac{\partial}{\partial \alpha} \det J = \det J \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \alpha}\right)$. The second-order derivative is given by

$$\begin{aligned}
\frac{\partial^2 w}{\partial \alpha \partial \beta} &= 2(\det J)^{-1} \operatorname{tr}\left(\frac{\partial J}{\partial \alpha}^T \frac{\partial J}{\partial \beta}\right) \\
&\quad -2(\det J)^{-1} \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \alpha}\right) \operatorname{tr}\left(J^T \frac{\partial J}{\partial \beta}\right) \\
&\quad -2(\det J)^{-1} \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \beta}\right) \operatorname{tr}\left(J^T \frac{\partial J}{\partial \alpha}\right) \\
&\quad + \frac{\operatorname{tr}(J^T J)}{\det J} \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \alpha}\right) \operatorname{tr}\left(J^{-1} \frac{\partial J}{\partial \beta}\right) \\
&\quad + \frac{\operatorname{tr}(J^T J)}{\det J} \operatorname{tr}\left(\frac{\partial J}{\partial \beta}^T J^{-1} \frac{\partial J}{\partial \alpha} J^{-1}\right),
\end{aligned} \tag{B.6}$$

using the fact that $\frac{\partial}{\partial \alpha} J^{-1} = -J^{-1} \frac{\partial J}{\partial \alpha} J^{-1}$ and assuming that J depends linearly on α, which is the case when α is a coordinate of a control point. The calculation of $\frac{\partial w}{\partial \alpha}$ and $\frac{\partial^2 w}{\partial \alpha \partial \beta}$ can be implemented as an assembly of linear and bilinear forms within IGA framework. In G+Smo, this can, for instance, be accomplished using the `gsExprEvaluator` class that is typically employed for isogeometric stiffness matrix assembly purposes.

Appendix 3: IGA Discretization Details

Using the spline parametrizations of patch geometries, we pull back the weak form (10) to the parameter domain, which results in the following equation:

$$\begin{aligned}
&\int_{]0,1[^2} \frac{1}{\epsilon_{cr}} J^{-T} \nabla u \cdot J^{-T} \nabla v \, |\det J| \, \mathrm{d}\boldsymbol{\xi} - k_0^2 \int_{]0,1[^2} \mu_r u v \, |\det J| \, \mathrm{d}\boldsymbol{\xi} \\
&\quad + \left(j k_0 + \frac{1}{2 r_t}\right) \int_{G^{-1}(\Gamma_t)} \frac{1}{\epsilon_{cr}} u v \left|\frac{\partial G}{\partial t}\right| \mathrm{d}t \\
&= \frac{1}{\epsilon_{cr}} \int_{G^{-1}(\Gamma_t)} \frac{1}{\epsilon_{cr}} \left(\frac{\partial u^i}{\partial n} \circ G + \left(j k_0 + \frac{1}{2 r_t}\right) u^i \circ G\right) v \left|\frac{\partial G}{\partial t}\right| \mathrm{d}t,
\end{aligned} \tag{C.1}$$

where we have $u = \hat{u} \circ G$, $v = \hat{v} \circ G$, and t is the parameter on the boundary. After applying the Galerkin method to (C.1), we arrive at the linear system of linear algebraic equations

$$A\boldsymbol{u} = (K + M + T)\boldsymbol{u} = \boldsymbol{f}, \tag{C.2}$$

where $\boldsymbol{u} = (u_1, \ldots, u_N)^T$ and K, M, T, and f are given by

$$K_{kl} = \int_{[0,1]^2} \frac{1}{\epsilon_{cr}} J^{-T} \nabla R_k \cdot J^{-T} \nabla R_l \, |\det J| \, \mathrm{d}\boldsymbol{\xi}, \tag{C.3a}$$

$$M_{kl} = -k_0^2 \int_{[0,1]^2} \mu_r R_k R_l \left|\det J\right| \, d\xi, \tag{C.3b}$$

$$T_{kl} = \left(jk_0 + \frac{1}{2r_t}\right) \int_{G^{-1}(\Gamma_t)} \frac{1}{\epsilon_{cr}} R_k R_l \left|\frac{\partial G}{\partial t}\right| \, dt, \tag{C.3c}$$

$$f_l = \int_{G^{-1}(\Gamma_t)} \frac{1}{\epsilon_{cr}} \left(\frac{\partial u^i}{\partial n} \circ G + \left(jk_0 + \frac{1}{2r_t}\right) u^i \circ G\right) e_l \left|\frac{\partial G}{\partial t}\right| \, d\xi. \tag{C.3d}$$

Note that the values of ε_{cr} and μ_r are set to the properties of gold in patch 4 and for air in the other patches. Owing to the restriction of the IGA library we are utilizing, we further reformulate the system of complex algebraic equations (C.2) as

$$\begin{bmatrix} \Re(A) & -\Im(A) \\ -\Im(A) & -\Re(A) \end{bmatrix} \begin{bmatrix} \Re(\boldsymbol{u}) \\ \Im(\boldsymbol{u}) \end{bmatrix} = \begin{bmatrix} \Re(\boldsymbol{f}) \\ -\Im(\boldsymbol{f}) \end{bmatrix}.$$