Structural and Multidisciplinary Optimization (2025) 68:9
https://doi.org/10.1007/s00158-024-03921-9

RESEARCH PAPER

Adaptive immersed isogeometric level-set topology optimization

Mathias R. Schmidt[1] · Lise Noël[2] · Nils Wunsch[3] · Keenan Doble[3] · John A. Evans[3] · Kurt Maute[3]

Received: 11 July 2024 / Revised: 14 October 2024 / Accepted: 24 October 2024 / Published online: 27 December 2024

Abstract

This paper presents for the first time an adaptive immersed approach for level-set topology optimization using higher-order truncated hierarchical B-spline discretizations for design and state variable fields. Boundaries and interfaces are represented implicitly by the iso-contour of one or multiple level-set functions. An immersed finite element method, the eXtended IsoGeometric Analysis, is used to predict the physical response. The proposed optimization framework affords different adaptively refined higher-order B-spline discretizations for individual design and state variable fields. The increased continuity of higher-order B-spline discretizations together with local refinement enables direct control over the accuracy of the representation of each field while simultaneously reducing computational cost compared to uniformly refined discretizations. A flexible mesh adaptation strategy enables local refinement based on geometric measures or physics-based error indicators. These adaptive discretization and analysis approaches are integrated into gradient-based optimization schemes, evaluating the design sensitivities using the adjoint method. Numerical studies illustrate the features of the proposed framework with static, linear elastic, multi-material, two- and three-dimensional problems. The examples provide insight into the effect of refining the design variable field on the optimization result and the convergence rate of the optimization process. Using coarse higher-order B-spline discretizations for level-set fields promotes the development of smooth designs and suppresses the emergence of small features. Moreover, adaptive mesh refinement for state variable fields results in a reduction of overall computational cost. Higher-order B-spline discretizations are especially interesting when evaluating gradients of state variable fields due to their higher inter-element continuity.

Keywords Topology optimization · Level-set method · Extended isogeometric analysis · Adaptive mesh refinement

1 Introduction

In general, Topology Optimization (TO) minimizes an objective function while simultaneously satisfying a set of constraints. Both the objective and the constraints are calculated from a set of performance measures. In its early days, TO results were mainly considered conceptual design ideas due to rough geometric representations. In contrast, modern TO methods aim to produce optimization designs that are closer to the final manufacturable design, which includes an accurate description of the geometry and evaluation of the physical response.

With an increased interest in TO, a variety of TO approaches have emerged. Most notable for continuum-mechanical problems are density-based and level-set-based TO approaches. Density-based TO methods are popular due to their simplicity and easy implementation, see Bendsøe and Sigmund (2004). By introducing fictitious design densities to interpolate material properties, this approach allows for the efficient use of gradient-based optimization algorithms, while promoting close to material-void designs. Extensions of density-based TO to multi-material applications have been presented by Bendsøe and Sigmund (2004) and Hvejsel and Lund (2011). Since their initial development for linear

Responsible Editor: Helder C. Rodrigues

✉ Mathias R. Schmidt
schmidt43@llnl.gov

[1] Computational Engineering Division, Lawrence Livermore National Laboratory, 7000 East Ave, Livermore, CA 94550, USA

[2] Department of Precision and Microsystems Engineering, Faculty of Mechanical Engineering, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands

[3] Aerospace Mechanics Research Center, Department of Aerospace Engineering Sciences, University of Colorado Boulder, 3775 Discovery Dr, Boulder, CO 80309-0429, USA

elastic problems, density-based methods have been applied to a variety of physics, see Eschenauer and Olhoff (2001); Sigmund and Maute (2013); Deaton and Grandhi (2014). A major drawback of density-based methods is the existence of areas with intermediate densities at the material interface, both during the optimization process and in the final design. These intermediate regions impede the crisp definition of the material interface and limit the accuracy of the physical response. While density projection methods, see Lazarov and Sigmund (2011), and fine discretizations improve the interface representation, the evaluation of interface phenomena remains challenging. To address these shortcomings of density-based TO approaches, level-set-based TO (LS-TO) approaches have been developed. The level-set method defines the geometric interfaces implicitly through the iso-contour of a level-set function (LSF) as introduced by Osher (1988), and provides a crisp description of the geometry. The level-set method was successfully applied to TO, see Wang et al. (2003); Allaire et al. (2004); Van Dijk et al. (2013). This work focuses on LS-TO, where the parameters of the discretized LSFs are treated as design variables and advanced by a nonlinear programming method.

TO results rely on the accuracy of the evaluation of the physical response, as both the objective and the constraints are calculated from a set of physics-based performance measures. Many TO applications employ numerical analysis methods to approximate the physical response. In this work, we focus on the finite element method. Considering geometry representation, three types of analysis models are typically used for LS-TO: (i) Models using conforming, body-fitted discretizations aligned with material interfaces, see Allaire et al. (2014); Christiansen et al. (2014); Schmidt et al. (2024). This approach requires repeated generation of high-quality conformal meshes as the geometry of the design changes throughout the optimization process. While a reduction in computational cost may be achieved by limiting these re-meshing operations to areas in the vicinity of the moving interfaces, the accuracy of the finite element prediction strongly depends on the mesh quality. (ii) Fictitious material approaches, also called Ersatz material methods, operate on fixed meshes and represent geometry via spatially varying material properties. These properties are defined in terms of the LSF, see Wang et al. (2003); Allaire et al. (2004). While there are different material interpolation schemes, all of them inevitably face the same challenge as density-based TO methods, namely that the fictitious material approach creates non-physical models at the material interface, lowering the accuracy of the finite element analysis. (iii) Immersed Finite Element Methods (IFEMs) embed the geometry of the physical domain into a computational domain with a much simpler geometry, see Babuška and Melenk (1997); Sethian and Wiegmann (2000); Peskin (2002); Mittal and Iaccarino (2005). Similar to methods using a conforming discretization, IFEMs preserve the geometric accuracy afforded by LSFs and facilitate a high-fidelity prediction of the physical response due to efficient enforcement of boundary and interface conditions, see Moës et al. (2006). Moreover, immersed methods do not require the repeated generation of high-quality body-fitted meshes. IFEMs have been incorporated into LS-TO by, for example, Duysinx et al. (2006); Burman et al. (2019); Van Dijk et al. (2013). This paper utilizes an immersed analysis model, a modification of the eXtended Finite Element Method (XFEM).

Most IFEMs rely on Lagrange polynomials for the interpolation of the state variable fields due to their simplicity and easy implementation. However, higher-order Lagrange polynomials are limited to a C^0 inter-element continuity. With the development of and advances in IsoGeometric Analysis (IGA) by Hughes et al. (2005), B-spline basis functions have been presented as an advantageous alternative to Lagrange basis functions. Smooth, higher-order basis functions in particular, such as quadratic and cubic B-splines, often result in a more accurate computation of the physical response per degree of freedom (DOF) than approaches relying on a C^0-continuous basis, see Hughes et al. (2008); Evans et al. (2009); Hughes et al. (2014). B-spline basis functions have been used for density-based TO approaches to discretize both the design and state variable fields, see Qian (2013); Wang et al. (2018); Gao et al. (2020). Moreover, IGA has been successfully employed in LS-TO by Cai et al. (2014); Wang and Benson (2016); Jahangiry and Tavakkoli (2017) for the discretization of design and state variable fields. These works exclusively use fictitious material approaches to interpolate the material properties. This paper proposes an LS-TO framework that builds upon the eXtended IsoGeometric Analysis (XIGA) approach, see Noël et al. (2022) and Schmidt et al. (2023). The XIGA is a modification of the XFEM and employs B-spline basis functions, whereas the XFEM uses standard Lagrange polynomials. This work highlights that integrating XIGA into LS-TO enables a crisp representation of geometry and an accurate and efficient prediction of the physical performance.

Both design and state variable fields typically benefit from a finer discretization, as this improves the resolution of geometry and the accuracy of the finite element prediction. This gain in accuracy typically results in higher computational costs as the number of design variables and finite element DOFs increases. Locally refined discretizations provide a compromise between accuracy and computational cost, as shown for TO applications first by Maute and Ramm (1995); Ramm et al. (1998). In the past two decades, adaptive TO has been studied extensively. Focusing on density-based TO problems, Costa and Alves (2003); Stainko (2006); Wang et al. (2010); Nana et al. (2016); Nguyen-Xuan (2017) presented various adaptive refinement approaches where the material interface was chosen as the refinement criteria. Adaptive refinement approaches were employed for LS-TO by Noël et al. (2020), also refining the discretization along the material interfaces. Bruggi and Verani (2011) and de Troya and Tortorelli (2020) used error estimators as refinement criteria to improve the accuracy of the finite element predictions. This paper introduces local mesh adaptation strategies for XIGA LS-TO to increase both the accuracy of the geometry representation and finite element analysis.

To improve the computational efficiency of TO, the idea of exploiting separate discretizations for state and design variables was first explored by Maute and Ramm (1995); Guest and Smith (2010). Using a coarser mesh for the design variable field discretization, the number of design variables is reduced and a faster convergence of the optimization problem has been observed. To further this idea, Wang et al. (2013, 2014) locally refined the state and design variable fields independently based on geometric and error estimator criteria. In a previous study, Noël et al. (2020) investigated adaptively refined B-spline bases to approximate the design and state variable fields. In their work, both fields were discretized on a mesh with the same adaptive refinement. While the state variable field was limited to linear interpolations, a comprehensive comparison of higher-order interpolations for the design variable field was performed. This comparison showed that the larger support of higher-order B-spline discretizations leads to slightly smoother designs. However, this previous work was limited to spatial discretizations where both design and state variable fields are refined equally. This restriction inherently couples the accuracy of the geometry representation and finite element prediction and prevents the independent refinement of design and state variable fields.

This paper presents a discretization framework that overcomes the above restriction by allowing for different adaptive refinements of the design and state variable fields. Local refinement of the individual fields is achieved through hierarchically refined B-splines and selecting field-specific polynomial orders. The individual discretizations are adapted based on field-specific criteria to control the accuracy of each field and to reduce the computational cost. The numerical quadrature of the weak form of the governing equations is performed on a single XIGA integration mesh. This XIGA integration mesh is generated by tessellating the most refined background mesh in a given mesh region to accurately integrate the governing equations for any of the discretizations employed. A more detailed construction of the approximation spaces can be found in Schmidt et al. (2023).

Many LS-TO approaches rely on an initial seeding of the design. This often results in a bias of the final design with respect to the initial seeding, as shown by Van Dijk et al. (2013). To mitigate this bias, we adopt the density seeding approach presented by Barrera et al. (2020). Moreover, we use the regularization approach from Geiss et al. (2019) to mitigate oscillations in the design variable field. Rigid body motions of topologically disconnected material sub-phases are prevented by adopting the concept of selective springs, introduced by Villanueva and Maute (2017). The optimization problem is solved with mathematical programming techniques, in particular, the Globally Convergent Method of Moving Asymptotes (GCMMA), see Svanberg (2002). The sensitivity analysis is performed with the adjoint method.

The features of the proposed LS-TO framework are investigated with numerical examples of static, linear elastic structural problems. These studies provide insight into the interplay between uniform and adaptive refinement of design and state variable fields, considering both geometry and physics-based refinement indicators.

The remainder of this paper is organized as follows: Sect. 2 provides an overview of the adaptive optimization strategy and a definition of the different meshes used in the XIGA. Section 3 expands on the XIGA model. Section 4 summarizes the fundamentals of hierarchical B-splines and their integration into the XIGA model. Section 5 focuses on the creation of hierarchical meshes and the re-meshing strategy. Sections 6 outline the variational forms of the governing equation and stabilization terms used in the XIGA formulation. Section 7 focuses on the formulation of optimization problems, level-set regularization techniques, and the hole-seeding approach. Numerical examples are presented in Sect. 8. Section 9 summarizes the work and presents conclusions.

2 Overview

This work presents an LS-TO framework with locally refined B-spline discretizations of design and state variable fields. The geometry and material composition are implicitly described by one or multiple LSFs. The discretized design variable field is utilized to parameterize a scalar-valued LSF where the discretization coefficients serve as design variables. Additionally, LSFs may be defined by analytical functions. The response is predicted by an immersed analysis method, the XIGA, on a non-conforming mesh. Adaptive mesh refinement is utilized to meet accuracy requirements while simultaneously reducing computational cost, using different discretizations for design and state variable fields. This section provides an overview of the optimization strategy and the featured mesh adaptation. Figure 1 illustrates the main steps in the optimization process: (1) Optimization from an initial design until adaptation is required, (2) the mesh adaptation, and (3) the restart of the optimization from the previous design with the new adaptively refined design variable field.

The discretization and analysis approaches used in this study differentiate between three types of meshes:

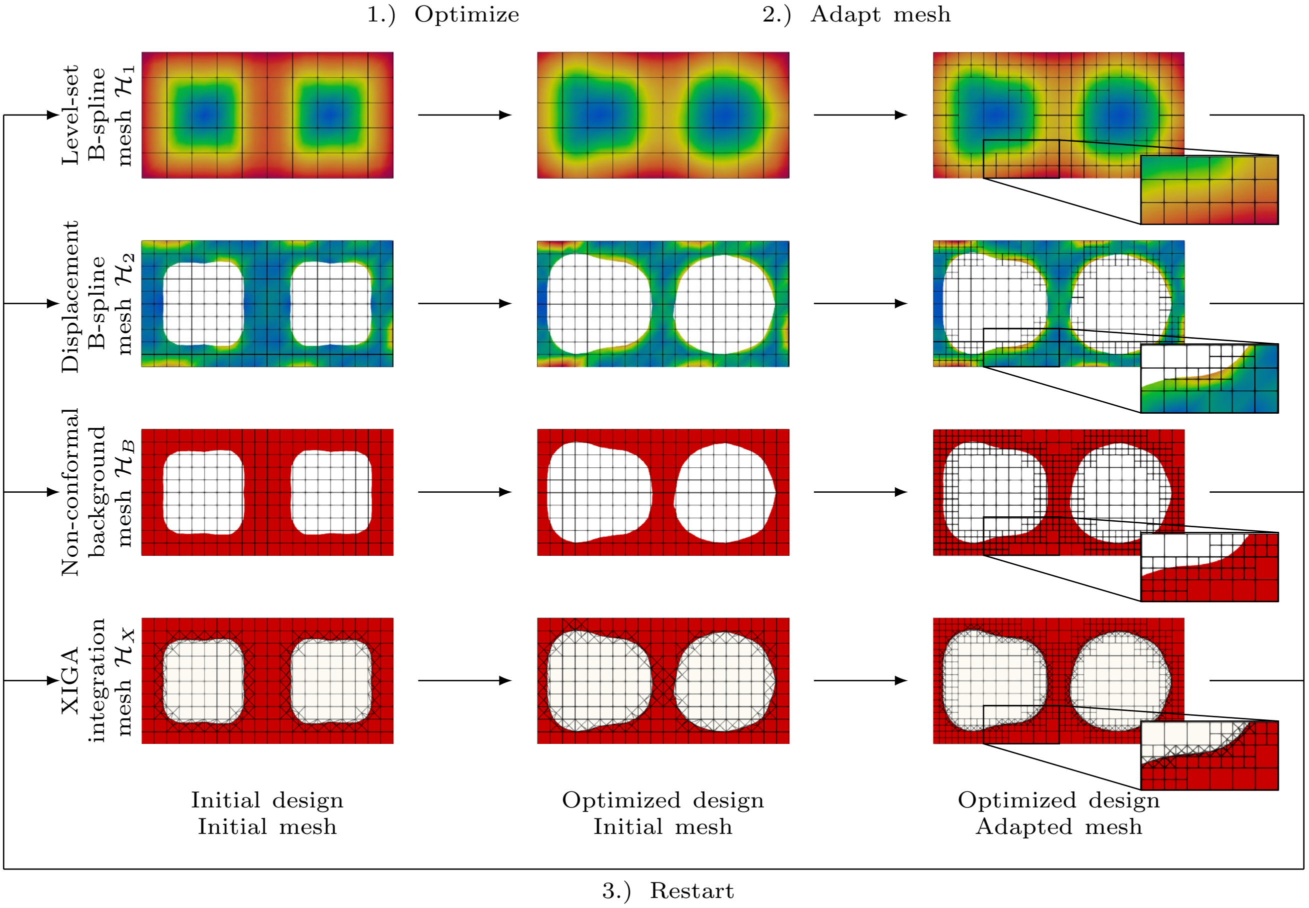

1.) Optimize
2.) Adapt mesh
Level-set B-spline mesh $\mathcal{H}_1$
Displacement B-spline mesh $\mathcal{H}_2$
Non-conformal background mesh $\mathcal{H}_B$
XIGA integration mesh $\mathcal{H}_X$
Initial design
Initial mesh
Optimized design
Initial mesh
Optimized design
Adapted mesh
3.) Restart

Fig. 1 Overview of the optimization strategy with mesh adaptation. The optimization strategy consists of three steps: perform optimization for a given number of iterations, adapt meshes based on certain criteria, and restart the optimization using the adapted meshes. The illustrated optimization problem presents four meshes: the level-set and displacement B-spline meshes $\mathcal{H}_1$ and $\mathcal{H}_2$, the non-conformal background mesh $\mathcal{H}_B$, and the XIGA integration mesh $\mathcal{H}_X$. The first and second row display the level-set and von Mises stress fields, respectively

Hierarchically refined B-spline meshes $\mathcal{H}_I$, a non-conformal background mesh $\mathcal{H}_B$, and an XIGA integration mesh $\mathcal{H}_X$. The design and state variable fields are discretized on hierarchically refined B-spline meshes $\mathcal{H}_I$, where $I = 1, \ldots, N_\mathrm{I}$ is the index of the N_I B-spline meshes. Each field may be discretized by a different B-spline mesh that is unique in spatial refinement and interpolation order. The XIGA uses a single non-conformal background mesh $\mathcal{H}_B$ to immerse the geometry and to create the XIGA integration mesh $\mathcal{H}_X$. The XIGA integration mesh $\mathcal{H}_X$ is used to integrate the weak form of the governing equations based on the B-spline discretizations $\mathcal{H}_I$ of the state variable fields. The non-conformal background mesh $\mathcal{H}_B$ must be of sufficient polynomial order and refinement to represent all B-spline discretizations used to interpolate the state variable fields. An additional refinement of the non-conformal background mesh $\mathcal{H}_B$ allows for a more precise representation of curved geometries on the XIGA integration mesh $\mathcal{H}_X$ without increasing the number of design variables or finite element DOFs.

Detailed information on the construction of the non-conformal background mesh $\mathcal{H}_B$ and the XIGA integration mesh $\mathcal{H}_X$ is provided in Sects. 3.3. The construction of the B-spline meshes $\mathcal{H}_I$ is detailed in Sect. 4. We utilize Lagrange extraction, as presented in Sect. 4.3, to compute nodal field representations on the non-conforming mesh $\mathcal{H}_B$ for all design and state variable fields discretized on the B-spline discretizations $\mathcal{H}_I$. Details on the construction of all differently refined meshes are given in Sect. 5. This novel capability of utilizing different individual B-spline discretizations for LS-XIGA TO is a primary focus of this work.

The initial meshes can be uniformly or locally refined. Mesh adaptation is initiated either when the material interface moves into a less refined region of the non-conformal background mesh $\mathcal{H}_B$, or after reaching a given number of iterations following the previous mesh adaptation. Although not explored in this work, the proposed framework allows for other strategies for initiating mesh adaptation, such as the geometry- or physics-based error exceeding a threshold. In the adaptation step, all meshes are adapted based on individual refinement indicator fields, which determine the level of mesh refinement or coarsening. This work utilizes physics-based and geometric refinement and coarsening indicators.

Meshes presented in Fig. 1 are individually adapted based on two refinement indicator fields, specifically the proximity to the zero iso-contour of the LSF and areas with large spatial gradients in the state variable field, here identified by large von Mises stresses. To demonstrate the ability to adapt meshes individually, the level-set B-spline mesh $\mathcal{H}_1$ is refined in a band around the zero iso-contour of the LSF. In contrast, the B-spline discretization $\mathcal{H}_2$ of the state variable field is only refined in areas of large von Mises stresses. The non-conforming background mesh $\mathcal{H}_B$ utilizes both refinement indicator fields to refine the mesh. The mesh adaptation step concludes by mapping the design variable field to the new design variable field discretization. The adaptive refinement of the meshes $\mathcal{H}_B$ and $\mathcal{H}_I$ is based on a structured tensor grid, which allows for a simple implementation of this robust and high-performing analysis framework, especially when compared to methods that involve adaptive refinement of body-fitted meshes. Nevertheless, each optimization step necessitates rebuilding the XIGA integration mesh $\mathcal{H}_X$ and the corresponding algebraic systems.

To continue the process, the optimization is restarted from the previous design. The adapted meshes are used for the construction of the new XIGA integration mesh $\mathcal{H}_X$ and the B-spline meshes $\mathcal{H}_I$ for design and state variable fields. It should be noted that the adaptation of the design variable field discretization may result in a change in the number of design variables.

3 Level-set XIGA

This section outlines the basic concepts of the LS-TO approach and the immersed finite element method, the XIGA, used in this work.

3.1 Geometry description

In this work, the geometry and the material composition are described by one or multiple level-set fields. The level-set method was first introduced by Osher (1988) to implicitly track an evolving geometry. The level-set method uses the iso-contour of a LSF $\phi(\mathbf{x})$ to represent the interface $\Gamma^{\pm}$ between two phases Ω^{+} and Ω^{-} as follows:

$$\begin{aligned} &\phi(\mathbf{x}) > \phi_t, \ \forall \ \mathbf{x} \in \Omega^{+} \\ &\phi(\mathbf{x}) < \phi_t, \ \forall \ \mathbf{x} \in \Omega^{-} \ , \\ &\phi(\mathbf{x}) = \phi_t, \ \forall \ \mathbf{x} \in \Gamma^{\pm} \end{aligned} \tag{1}$$

where ϕ_t is the iso-contour threshold, with $\phi_t = 0.0$ being the most common choice. For multi-material optimization problems, this work employs multiple LSFs $\phi_i(\mathbf{x})$ with $i = 1, \ldots, N_\text{l}$, where N_l is the total number of LSFs. We define a phase Ω^P as the set of all spatial points $\mathbf{x}$ which have the same combination of LSF signs. A unique phase index $P(\mathbf{x})$ is calculated through

$$P(\mathbf{x}) = \sum_{i=1}^{N_\text{l}} 2^{i-1} H_i(\mathbf{x}), \tag{2}$$

where $H_i(\mathbf{x})$ is an indicator function operating on the LSF $\phi_i(\mathbf{x})$ and is defined as follows:

$$H_i(\mathbf{x}) = \begin{cases} 0, \ \phi_i(\mathbf{x}) \leq \phi_t \\ 1, \ \phi_i(\mathbf{x}) > \phi_t. \end{cases} \tag{3}$$

This approach allows for the definition of 2^{N_l} phases such that $\Omega^1 \cup \cdots \cup \Omega^{(2^{N_\text{l}})} = \Omega$. One or multiple phases can be assigned a unique material Ω^m or can be void, with $m = 1, \ldots, N_\text{m}$, where N_m is the total number of materials.

3.2 Level-set design variables

The LSFs used in this work are defined by an analytic function or discretized on a mesh using B-spline basis functions $B_k(\mathbf{x})$:

$$\phi_i^h(\mathbf{x}) = \sum_k B_k(\mathbf{x}) \phi_i^k, \tag{4}$$

where ϕ_i^k are the coefficients associated with the discretized LSF ϕ_i^h.

In this work, the coefficients ϕ_i^k are defined as explicit functions of the level-set design variables $\mathbf{s} := \{ s \in \mathbb{R}^{N_s} | \phi_\text{low} \leq s_j \leq \phi_\text{up}, j = 1, \ldots, N_s \}$, where N_s is the total number of B-spline coefficients for this discretization; see also Van Dijk et al. (2013); Sigmund and Maute (2013). The design variables are constrained by an upper and lower bound, ϕ_up and ϕ_low, respectively.

Many TO problems employ a convolutional or PDE-based filter to improve the numerical stability and the convergence of the optimization problem, see Bourdin (2001); Burger (2003); Gournay (2006) and Lazarov and Sigmund (2011). This work utilizes higher-order B-spline discretizations. The increased smoothness and larger support of higher-order B-spline basis functions provide an equivalent effect to filtering and thus an additional filter is not required.

Equally to convolutional and PDE-based filter approaches, a coarse, higher-order B-spline discretization does not necessarily suppress small features, see Villanueva and Maute (2014) and Coffin and Maute (2016). However, it promotes smoother shapes and bulkier features, see Noël et al. (2020). We demonstrate the effect of higher-order B-spline basis functions in combination with a coarser mesh in detail in our first example, see Sect. 8.1.

Contrary to classical approaches wherein the LSF is updated by solving the Hamilton-Jacobi equation, here the coefficients of the level-set field are the design variables that are updated by mathematical programming techniques.

3.3 The extended isogeometric analysis

In this work, we use the XIGA to discretize the state variable fields on the non-conforming background mesh. The XIGA is a modification of the XFEM and utilizes the advantages of B-spline basis functions, whereas the XFEM uses standard Lagrange polynomials. The XFEM was first developed for crack modeling by Belytschko and Black (1999). Since then, its applications have been extended to material modeling, see Belytschko et al. (2009), and design optimization, see Van Dijk et al. (2013), among others. Basis functions are defined on a background mesh to construct a non-conforming finite element approximation space. We follow the basic concepts of a generalized Heaviside enrichment strategy, introduced by Terada et al. (2003) and later adopted for TO by Makhija and Maute (2014). The non-conforming finite element approximation space with a total of N_b basis functions is locally enriched to avoid artificial coupling caused by the same basis function interpolating in topologically disconnected material domains Ω^m, see Andreasen et al. (2020).

This enrichment strategy for a single basis function is illustrated in Fig. 2. The problem consists of two materials, Ω^1 and Ω^2. The support of the basis function is depicted by a dashed red line. The basis function is three times enriched as it interpolates into three topologically disconnected regions of the two materials. This approach has no limitation with regard to the number of materials or the spatial complexity of the intersection configurations. For the definition of the enriched basis, an indicator function $\psi_j^l(\mathbf{x})$, defined as the two-element set $\{0, 1\}$, is introduced. This function $\psi_j^l(\mathbf{x})$ is used to select the active enrichment level l for the basis function j at the spatial coordinate $\mathbf{x}$. The component of a vector-valued state variable field $\mathbf{u}^h(\mathbf{x})$ is approximated by the enriched finite element space as follows:

$$u_i^h(\mathbf{x}) = \sum_{j=1}^{N_b} \left(\sum_{l=1}^{L_j} \psi_j^l(\mathbf{x}) B_j(\mathbf{x}) c_{i,j}^l \right), \tag{5}$$

where L_j is the maximum number of enrichment levels for basis function j. The B-spline basis functions B_j are evaluated at the spatial coordinate $\mathbf{x}$, and $c_{i,j}^l$ are the enriched basis function coefficients for the basis function j, the enrichment level l, and the vector component i. The indicator function ensures satisfaction of the Partition of Unity (PU) principle, see Babuška and Melenk (1997), as only one set of enriched basis coefficients is used to interpolate the solution.

The XIGA is suited to model intra-element discontinuities of state variables within a non-conforming background element. In this work, boundary or interface conditions are enforced weakly using Nitsche's method, see Nitsche (1971) and Burman (2012). The accuracy of our analysis model has been demonstrated in Noël et al. (2022) and Schmidt et al. (2023), which suggest that the employed XIGA method achieves the same level of accuracy as traditional body-fitted FEM analysis.

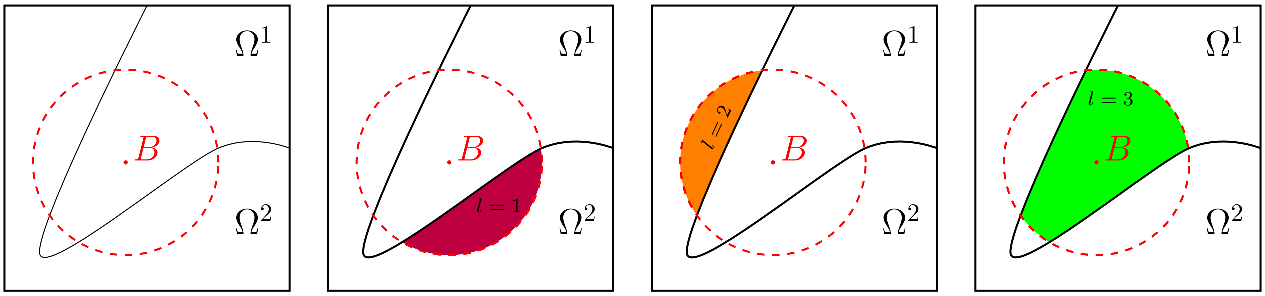


Fig. 2 Basis function enrichment for the two material problem, Ω^1 and Ω^2. The support of the basis function B is delineated by a dashed red line. The basis function is three times enriched based on the three topologically disconnected regions of the two phases into which it is interpolating

3.3.1 XIGA integration mesh

The XIGA decomposes background elements intersected by the zero iso-contour of the LSFs into material subdomains and identifies the material interfaces between them. The weak form of the governing equations is integrated over these subdomains and interfaces. In this work, to initiate the decomposition process, the quadrilateral or hexahedral background elements of the mesh $\mathcal{H}_B$, which are intersected by geometric interfaces, are first subdivided into four triangular integration elements in 2D and intersected hexahedral background elements are subdivided into 24 tetrahedral integration elements in 3D. Sequentially processing all LSFs, the resulting triangles and tetrahedrons that are also intersected by an LSF are decomposed into smaller triangles and tetrahedrons with their edges and faces conforming to the geometric interfaces.

To increase the robustness of the geometry decomposition within an intersected background element, we assume that edges are only intersected once and only if the level-set values of the endpoints $[I, J]$ are on opposite sides of an interface, i.e., $(\tilde{\phi}_i^h(I) - \phi_t)(\tilde{\phi}_i^h(J) - \phi_t) < 0$.

The intersection location is computed by finding the root of the given LSF along the edge. For this, a multilinear interpolation $\tilde{\phi}_i^h$ on the background element is used. Constructing vertices at these root points and connecting them with planar edges and faces forms the piecewise linear approximation of the interfaces.

For coarse meshes $\mathcal{H}_B$, the geometric approximation error for the interface can be significant, as is illustrated in Fig. 3. On the left, an LSF is discretized on a 2×3 mesh utilizing a quadratic polynomial order. The iso-contour depicted in black represents the geometry. The figure in the middle illustrates the resulting XIGA integration mesh $\mathcal{H}_X$ constructed based on a 2×3 nonconformal background mesh $\mathcal{H}_B$. This XIGA integration mesh $\mathcal{H}_X$ poorly describes the geometry defined by the level-set iso-contour. The geometric representation of the XIGA integration mesh $\mathcal{H}_X$ is improved through additional refinement of the non-conforming background mesh $\mathcal{H}_B$, as illustrated by the picture on the right.

In this work, individual state variable fields may be discretized by their own, individual approximation space. Therefore, each approximation space must be enriched separately. The numerical integration of the governing equations, however, is performed on a single XIGA integration mesh $\mathcal{H}_X$. This XIGA integration mesh $\mathcal{H}_X$ holds a set of enriched extraction operators for every enriched discretization available.

4 Hierarchical B-splines

In this work, we discretize the design and state variable fields by truncated hierarchical B-spline (THB) bases. In this section, the basic concepts of B-spline basis functions in one and multiple dimensions are outlined first. Next, we summarize the concept of Lagrange extraction used to link the smooth spline basis functions to the Lagrange basis of the non-conformal background mesh $\mathcal{H}_B$. Additional details on the THB-spline implementation, Lagrange extraction, and various convergence studies can be found in Schmidt et al. (2023).

4.1 B-spline basis functions

Starting from a knot vector $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$, for which $\xi \in \mathbb{R}$ and $\xi_1 \leq \xi_2 \leq \cdots \leq \xi_{n+p+1}$, univariate B-spline basis functions $N_{i,p}(\xi)$ of degree p are constructed recursively starting from the piecewise constant basis function:

$$N_{i,0}(\xi) = \begin{cases} 1, & \text{if } \xi_i \leq \xi \leq \xi_{i+1}, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

and using the Cox-de Boor recursion formula in de Boor (1972) for higher degrees, $p > 0$:

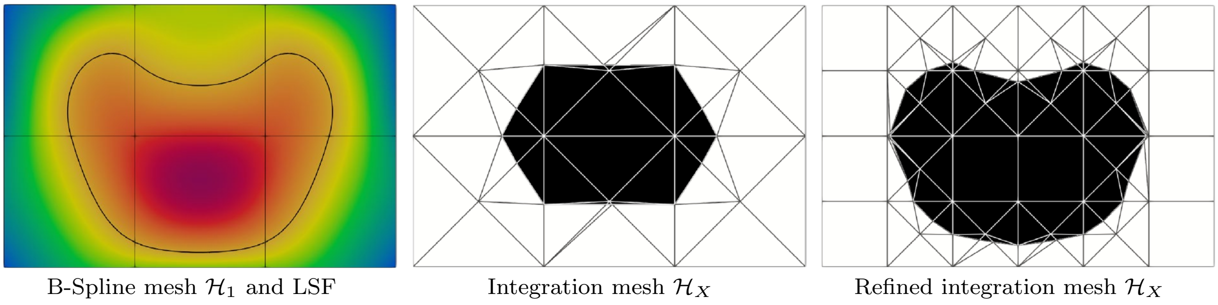

B-Spline mesh $\mathcal{H}_1$ and LSF
Integration mesh $\mathcal{H}_X$
Refined integration mesh $\mathcal{H}_X$

Fig. 3 Illustration of a quadratic level-set field discretized on a 2×3 mesh $\mathcal{H}_1$ (left). The resulting XIGA integration meshes $\mathcal{H}_X$ for a 2×3 and 4×6 non-conforming background mesh $\mathcal{H}_B$ are presented in the middle and right picture, respectively

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{7}$$

The corresponding B-spline basis is C^{p-1} continuous at every knot in the interior of the domain for $\xi_i < \xi_{i+1}$, while it is C^{∞} continuous in between the knots. A knot span is defined as the half-open interval $[\xi_i, \xi_{i+1})$. Within this context, an element is defined as a non-empty knot span. Equivalently, elements are precisely the non-overlapping subdomains over which the basis is C^{∞}.

Tensor-product B-spline basis functions $B_p(\boldsymbol{\xi})$ of spatial dimension d are obtained by applying the tensor product to univariate B-spline basis functions of each parametric direction $m = 1, \ldots, d$. The d knot vectors are defined as $\varXi^m = \{\xi_1^m, \xi_2^m, \ldots, \xi_{n_m+p+1}^m\}$ with n_m being the number of basis functions in the parametric direction m:

$$B_{\mathbf{i},p}(\boldsymbol{\xi}) = \prod_{m=1}^{d} N_{i_m,p}^{m}(\xi^m), \tag{8}$$

where the position in the tensor-product structure is given by the index $\mathbf{i} = \{i_1, \ldots, i_d\}$. A B-spline space $\mathcal{V}$ is defined as the span of B-spline basis functions.

4.2 Truncated hierarchical B-splines

This work utilizes THB-spline bases to discretize state and design variable fields due to their strong stability and PU property, see Giannelli et al. (2012). The THB-spline bases are constructed based on a hierarchical mesh of depth n defined as a sequence of subdomains $\varOmega^l$ (see Fig. 4):

$$\varOmega^{n-1} \subseteq \varOmega^{n-2} \subseteq \cdots \subseteq \varOmega^0 = \varOmega, \tag{9}$$

where each subdomain $\varOmega^l$ is a refined sub-region of $\varOmega^{l-1}$. Consequently, $\varOmega$ is equal to the union of all the subdomains $\varOmega^l$. Moreover, it relies on the hierarchical refinement property of B-splines which enables the representation of a basis function B^l, part of $\mathcal{B}^l$ and defined on the domain $\varOmega^l$ in terms of the finer basis of level $l+1$:

$$B^l = \sum_{B^{l+1} \in \mathcal{B}^{l+1}} c_{B^{l+1}}^{l+1}\left(B^l\right) B^{l+1}, \tag{10}$$

where $c_{B^{l+1}}^{l+1}$ is the coefficient associated with a basis function B^{l+1}.

To build THB-spline bases, a sequence of tensor-product B-spline spaces is introduced where each B-spline space $\mathcal{V}^l$ has a corresponding basis $\mathcal{B}^l$:

$$\mathcal{V}^0 \subset \mathcal{V}^1 \subset \mathcal{V}^2 \subset \mathcal{V}^3 \subset \cdots \subset \mathcal{V}^n. \tag{11}$$

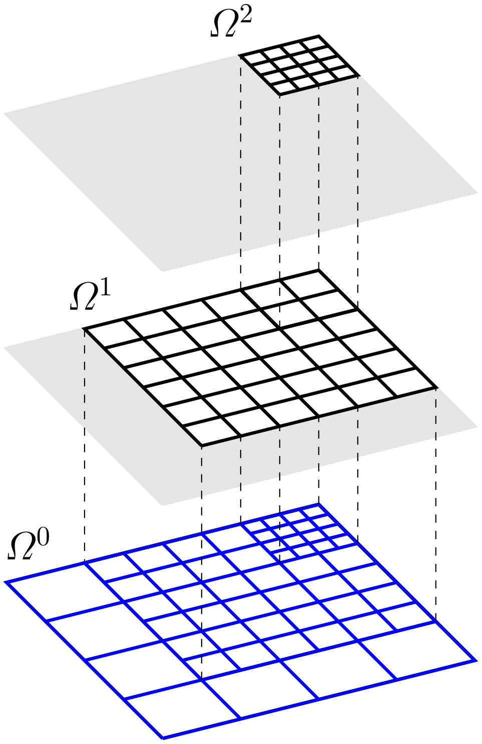


Fig. 4 Hierarchically refined mesh

The THB-spline basis $\mathcal{T}$ is constructed recursively based on the sequence of truncated B-spline bases $\mathcal{B}^l$ that span the domains $\varOmega^l$. In an initial step, the basis functions defined on the coarsest level, $l = 0$, are collected and assigned to $\mathcal{T}^0$. The THB-spline basis $\mathcal{T}^{l+1}$ is constructed by taking the union of the truncated basis functions $\operatorname{trunc}^{l+1}(B)$ of $\mathcal{T}^l$ and all basis functions B in $\mathcal{B}^{l+1}$ whose support lies in $\varOmega^{l+1}$. The recursive algorithm of Garau and Vázquez (2018) reads

$$\begin{cases} \mathcal{T}^0 & =: \mathcal{B}^0 \\ \mathcal{T}^{l+1} & =: \{\operatorname{trunc}^{l+1}(B) \mid B \text{ in } \mathcal{T}^l \wedge \operatorname{supp}(B)\} \\ & \quad \cup \{B \in \mathcal{B}^{l+1} \mid \operatorname{supp}(B) \subseteq \varOmega^{l+1}\}, \\ & \quad \text{for } l = 0, \ldots, n-2, \end{cases} \tag{12}$$

where the truncation operation of basis function B^l is defined as follows (Giannelli et al. (2012); Garau and Vázquez (2018)):

$$\begin{aligned} \operatorname{trunc}^{l+1}(B^l) &= \sum_{\substack{B^{l+1} \in \mathcal{B}^{l+1}, \\ \operatorname{supp}(B^{l+1}) \nsubseteq \varOmega^{l+1}}} c_{B^{l+1}}^{l+1}\left(B^l\right) B^{l+1}, \\ &= B^l - \sum_{\operatorname{supp}(B^{l+1}) \subseteq \varOmega^{l+1}} c_{B^{l+1}}^{l+1}\left(B^l\right) B^{l+1}. \end{aligned} \tag{13}$$

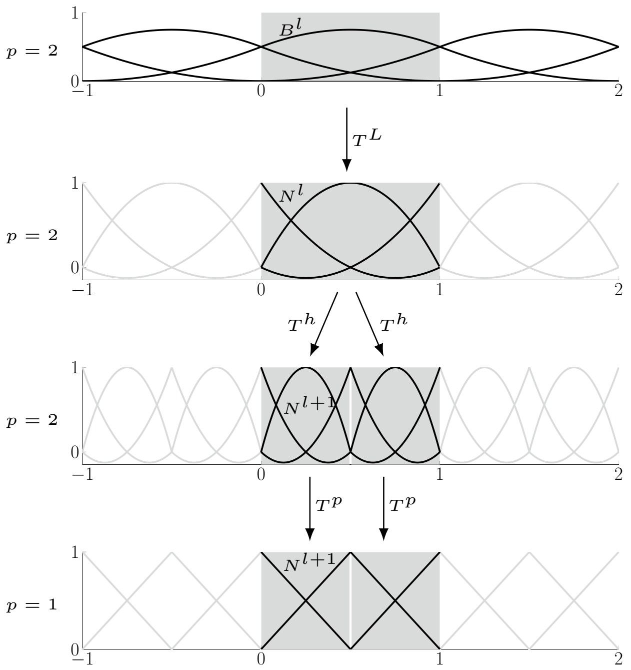


Fig. 5 Illustration of the extraction process of a coarse quadratic B-spline basis onto a more refined, linear Lagrange basis for a single highlighted B-spline element

4.3 Lagrange extraction

This work allows for differently refined THB-spline discretizations for individual design and state variable fields. These discretizations do not conform with material interfaces or external boundaries. To simplify the formulation and evaluation of the weak form of the governing equations, numerical integration is performed on a unique XIGA integration mesh $\mathcal{H}_X$ which, as discussed in Subsection 3.3.1, is generated by tessellation of the background mesh $\mathcal{H}_B$. The non-conforming background mesh $\mathcal{H}_B$ may be more refined and may use basis functions of a different polynomial order than the B-spline discretizations $\mathcal{H}_I$. The extraction process of interpolating the B-spline basis associated with one of the meshes $\mathcal{H}_I$ on the basis of the mesh $\mathcal{H}_B$ is shown in Fig. 5.

Following the work of Schillinger et al. (2016); D'Angella et al. (2018) and Schmidt et al. (2023), "extraction" is used to link the B-spline bases to the C^0 Lagrange bases[1]. This allows for the use of a standard Lagrange finite element implementation to compute the elemental contributions to Jacobian and residual, while a B-spline discretization $\mathcal{H}_I$ is used to approximate the physical response. The THB-spline basis functions B_k defined on a background element of the mesh $\mathcal{H}_I$ interpolated using the Lagrange basis functions defined on an equally or more refined element of the mesh $\mathcal{H}_B$:

$$B_k(\xi) = \sum_i T_{ik}\hat{N}_i^L(\xi). \tag{14}$$

The extraction operator T_{ik} employed in this work takes the following form:

$$T_{ik} = \sum_j T^L_{jk} \sum_l T^h_{lj} T^p_{il}, \tag{15}$$

where T^L, T^h , and T^p are a Lagrange extraction operator, a h-refinement extraction operator, and a p-coarsening extraction operator, respectively. These operators are described in detail as follows.

The Lagrange extraction operator T^L enables an element-wise representation of THB-spline basis functions in terms of Lagrange shape functions as

$$B_k(\xi) = \sum_j B_k(\xi_j) N^L_j(\xi) = \sum_j T^L_{jk} N^L_j(\xi), \tag{16}$$

where B_k is the k^{th} THB-spline basis function, N_j^L are the Lagrange shape functions over the element, ξ_j are the locations at which the Lagrange shape functions N_j^L are interpolatory, and $T^L_{jk} = B_k(\xi_j)$.

Similarly, the h-refinement extraction operator T^h allows for the representation of Lagrange shape functions over each element in terms of Lagrange shape functions over a more refined element as

$$N^L_j(\xi) = \sum_l N^L_j(\hat{\xi}_l)\hat{N}^L_l(\hat{\xi}_l) = \sum_l T^h_{lj}\hat{N}^L_l(\hat{\xi}_l), \tag{17}$$

where $\hat{N}_l^L$ are the Lagrange shape functions over the more refined element, $\hat{\xi}_l$ are the locations at which the Lagrange shape functions $\hat{N}_l^L$ are interpolatory, and $T^h_{lj} = N^L_j(\hat{\xi}_l)$.

Lastly, the p-coarsening extraction operator T^p enables the representation of Lagrange shape functions of order p over each element in terms of Lagrange shape functions of order lower than p as

$$N^L_l(\xi) = \sum_i N^L_l(\hat{\xi}_i)\hat{N}^L_i(\hat{\xi}_i) \approx \sum_i T^p_{il}\hat{N}^L_i(\hat{\xi}_i), \tag{18}$$

where $\hat{N}_i^L$ are the lower-order Lagrange shape functions, $\hat{\xi}_i$ are the locations at which the Lagrange shape functions $\hat{N}_i^L$ are interpolatory, and $T^p_{il} = N^L_l(\hat{\xi}_i)$. Note that in this case the Lagrange extraction operation is not exact, i.e., the space spanned by the Lagrange basis function is not the space spanned by the B-spline bases. When using this operator

[1] The space spanned by the B-Spline basis of polinomial order $p_{\text{B-spline}}$ defined on a given mesh is a subspace of the space spanned by the Lagrange basis of order $p_{\text{Lagrange}} \geq p_{\text{B-spline}}$ on the same mesh. The Lagrange extraction is exact, i.e., the extracted field expressed in the Lagrange basis retains a C^{p-1} continuity by construction.

in a projection, one must guarantee a sufficiently refined Lagrange mesh compared to the B-spline discretization for the resulting linear system to be of full rank.

Our framework limits the background element refinement to a factor of two, as presented in Subsection 5.2.1. This consequently limits the number of h-refinement extraction matrices to four in 2D and eight in 3D. These matrices can be precomputed and efficiently selected exploiting the polytree data structure presented in the following section.

5 Hierarchical mesh

TO problems often lead to small geometric features with large surface curvatures and areas with large spatial gradients of the state variables. To locally resolve the associated design and state variable fields while preventing a significant increase in computational cost, this work utilizes individual, locally refined B-spline discretizations $\mathcal{H}_I$. We adopt the adaptive mesh generation tool presented in Schmidt et al. (2023). In this section, we discuss local refinement strategies employed in this work.

5.1 Adaptively refined meshes

This work utilizes locally refined meshes for the construction of B-spline discretizations $\mathcal{H}_I$ and the non-conformal background mesh $\mathcal{H}_B$. All refined meshes are based on the same, initial tensor grid with a refinement level of $l^{m=0}$. Higher refinement levels $l^{m+1} > l^m$ are created recursively by locally subdividing elements of refinement level l^m into four equal-sized rectangles in 2D and eight equal-sized hexahedrons in 3D, respectively. This subdivision can be expressed via a polytree data structure, that is a quadtree in two and an octree in three dimensions. More implementation details for a memory and computationally efficient data structure can be found in Schmidt et al. (2023). Considering an element of level l^m, the next coarser element of level l^{m-1} is referred to as its parent, and the refined elements of level l^{m+1} as its children. The polytree data structure enables an efficient and unique identification of parents and children. This is particularly useful when computing the extraction operators and for the adaptation of the meshes outlined below.

5.2 Local mesh adaptation strategies

Algorithm 1 Marking of elements for refinement

```
for Meshes $\mathcal{H}_I$ do
    for Elements $\mathcal{H}_B^e$ do
        Get RO for element $\mathcal{H}_B^e$
        $\mathcal{H}_I^e = \mathcal{H}_B^e$
        while $\mathcal{H}_I^e$ does not exist in $\mathcal{H}_I$ do
            $\mathcal{H}_I^e$ = parent of $\mathcal{H}_I^e$
        end while
        if RO is refine and $l = l_{\max}^{\mathcal{H}_I}$ then
            Set RO to hold
        end if
        if RO is refine then
            Mark $\mathcal{H}_I^e$ for refinement
        else if RO is hold then
            if $l > l^0$ then
                Mark parent of $\mathcal{H}_I^e$ for refinement
            else
                Do nothing
            end if
        else if RO is coarsen then
            if $l > l^1$ then
                Mark parent of parent of $\mathcal{H}_I^e$ for refinement
            else
                Do nothing
            end if
        else if RO is drop then
            Do nothing
        end if
    end for
    for Max levels of refinement for $\mathcal{H}_I$ do
        Apply Algorithm 2
    end for
end for
```

Algorithm 2 Perform mesh refinement

```
Collect marked $\mathcal{H}_I^e$ into queue for refinement
while Changes in refinement queue do
    for All $\mathcal{H}_I^e$ in queue for refinement do
        Apply Algorithm 3 Refinement buffer
        Collect marked $\mathcal{H}_I^e$ in queue for refinement
    end for
end while
Refine all marked elements
```

As the design evolves during the TO process, the discretizations of state and design variable fields are repeatedly adapted, i.e., locally refined. Refinement criteria such as the proximity to material interfaces and physics-based error estimators are considered to determine regions where local refinement or coarsening is desired. Refinement is performed for the non-conforming background mesh $\mathcal{H}_B$ and the B-spline meshes $\mathcal{H}_I$ with a corresponding mesh adaptation strategy. The refinement criteria in this work are defined by either nodal or elemental, proximity to material interfaces and physics-based estimator fields. All

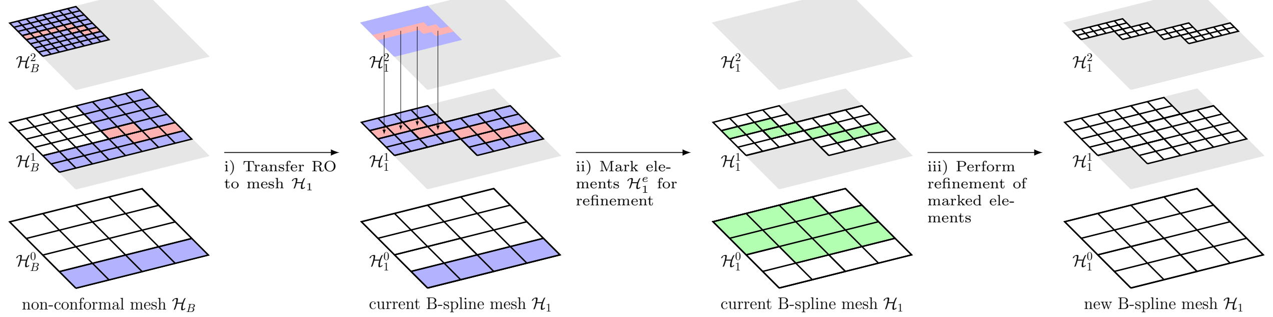

$\mathcal{H}_B^2$
$\mathcal{H}_B^1$
$\mathcal{H}_B^0$
non-conformal mesh $\mathcal{H}_B$
i) Transfer RO to mesh $\mathcal{H}_1$
$\mathcal{H}_1^2$
$\mathcal{H}_1^1$
$\mathcal{H}_1^0$
current B-spline mesh $\mathcal{H}_1$
ii) Mark elements $\mathcal{H}_1^e$ for refinement
$\mathcal{H}_1^2$
$\mathcal{H}_1^1$
$\mathcal{H}_1^0$
current B-spline mesh $\mathcal{H}_1$
iii) Perform refinement of marked elements
$\mathcal{H}_1^2$
$\mathcal{H}_1^1$
$\mathcal{H}_1^0$
new B-spline mesh $\mathcal{H}_1$

Fig. 6 Illustration of the mesh adaptation strategy. Non-conformal background mesh $\mathcal{H}_B$ with *refine* RO (red) and *hold* RO (blue). Transfer of ROs to the current B-spline mesh $\mathcal{H}_1$. Subsequently, elements of B-spline mesh $\mathcal{H}_1$ are marked for refinement based on the RO (green). A new B-spline mesh $\mathcal{H}_1$ is generated

fields are represented on the non-conformal background mesh $\mathcal{H}_B$ and are referred to as refinement indicator fields.

For each refinement indicator field, a Refinement Operation (RO) is assigned to each element $\mathcal{H}_B^e$ of the current non-conformal background mesh $\mathcal{H}_B$ based on the specific refinement criteria. ROs are either *refine*, *hold*, *coarsen*, or *drop* and are used to adapt each mesh $\mathcal{H}_B$ and $\mathcal{H}_I$ individually. The mesh adaptation is then facilitated in a three-step process. i) Find the first existing parent element $\mathcal{H}_I^e$ with respect to the non-conformal background mesh element $\mathcal{H}_B^e$, as the non-conformal background mesh $\mathcal{H}_B$ may be more refined than the B-spline mesh $\mathcal{H}_I$. The RO is then applied to this element $\mathcal{H}_I^e$. This search is facilitated by the polytree data structure mentioned earlier. ii) Elements of the polytree of the current mesh $\mathcal{H}_I$ are marked for refinement based on the ROs of the elements $\mathcal{H}_I^e$. This marking procedure is explained in detail below. iii) A new instantiation of mesh $\mathcal{H}_I$ is created by refining all marked elements starting at the coarsest level $l^{m=0}$ and recursively creating their children $l^{m+1} > l^m$ until all marked elements are refined. Note that this process generates a new mesh $\mathcal{H}_I$ which may not include previous elements if they are not created by the new refinement information. This effectively results in a coarsening effect after the mesh adaptation. The mesh generation process is described in more detail in Schmidt et al. (2023).

The ROs are summarized in Algorithm 1. Based on the type of RO for element $\mathcal{H}_I^e$, one of the following marking procedures is executed. A *refine* RO marks the element $\mathcal{H}_I^e$ for refinement. A *hold* RO marks the parent of the element $\mathcal{H}_I^e$ for refinement, and a *coarsen* RO marks the parent's parent for refinement. In addition, all parents of marked elements are automatically marked for refinement. Examples of the *refine* and *hold* ROs are presented in Fig. 6. A *drop* RO does not trigger any marking for refinement. The presented framework does not allow for coarsening of elements on the coarsest refinement level l^0. Moreover, each mesh $\mathcal{H}_I$ may have an assigned maximal $l_{\max}^{\mathcal{H}_I}$ and minimal refinement level $l_{\min}^{\mathcal{H}_I}$. These maximal and minimal refinement levels may be adjusted throughout the optimization process.

5.2.1 Mesh regularity requirements

Locally refined B-spline meshes $\mathcal{H}_I$ are utilized to construct THB-spline bases, see Sect. 4. This requires the consideration of additional mesh regularity requirements. The size difference between adjacent elements in a refined mesh is limited to a factor of four in 2D and eight in 3D. Furthermore, all neighbor elements inside a so-called buffer zone of an element on level l must be of level $l-1$ or higher. The buffer range d_{buffer} for a particular element is calculated by multiplying the element size with a buffer parameter b_{buffer}. When creating a B-spline basis, the width of the buffer zone must be greater than or equal to the support size of the interpolation functions. In this work, each state variable field is interpolated with an individual interpolation order p. To satisfy the mesh regularity requirement for all interpolation functions, the buffer parameter must be chosen such that $b_{\text{buffer}} \geq p_{\max}$, where $p_{\max}$ is the maximal polynomial degree of all used B-spline bases. The mesh refinement procedure for enforcing a buffer zone is summarized in Algorithm 3. The algorithm is applied to each element in $\mathcal{H}_B$ and $\mathcal{H}_I$ initially marked for refinement in Step 2 of Algorithm 2 and starts by determining its parent element. The refinement status of the parent's neighbors, i.e., elements within the buffer range of the considered parent, is checked. If these neighbors are not marked for refinement, the distance $d_{\max}$ between the considered parent element and its neighbors is calculated. If the distance $d_{\max}$ is smaller than the buffer range d_{buffer}, the neighbor elements are marked for refinement. The algorithm is then applied recursively to all newly marked neighbor elements until no further elements are marked for refinement.

Algorithm 3 Applying mesh regularity requirements

```
$\mathcal{H}^e_I$ marked for refinement in Algorithm 2
Get parent
Get parent's neighbors in half buffer range $d_{\text{buffer}}/2$
if Neighbor exists and not marked for refinement then
    Calculate distance $d_{\max}$ between parent and neighbor
    if $d_{\max} < d_{\text{buffer}}$ then
        Mark neighbor for refinement
        Apply refinement buffer algorithm 3 for neighbor
    end if
end if
```

6 Structural analysis

The LS-TO approach presented in this work is applicable to a broad range of physical systems. We focus here on structural topology optimization problems. This section presents the variational form of the linear elastic governing equations. The state variable field, i.e., the displacement field, is used to compute optimization performance measures such as total strain energy and maximal stress. This section presents the computation of the smooth stress field in this XIGA TO framework. We will recall the selective structural springs approach for material-void TO problems to suppress rigid body motion of topologically disconnected regions that may appear throughout the optimization process. Further, we outline the definition of the physics-based error estimator which is used for the adaptation of the hierarchical mesh.

6.1 Variational form of governing equations

Following a Galerkin approach, we define the solution spaces $\mathcal{U} = \mathcal{U}^1 \times \mathcal{U}^2 \times \ldots \times \mathcal{U}^{N_m}$ and test spaces $\mathcal{V} = \mathcal{V}^1 \times \mathcal{V}^2 \times \ldots \times \mathcal{V}^{N_m}$ where the spaces $\mathcal{U}^m$ and $\mathcal{V}^m$ are Hilbert spaces over the domain Ω^m:

$$\begin{aligned} \mathcal{U}^m_h &= \{\boldsymbol{u}^m_h \in \left(H^1(\Omega^m)\right)^d\}, \\ \mathcal{V}^m_h &= \{\boldsymbol{v}^m_h \in \left(H^1(\Omega^m)\right)^d\}, \end{aligned} \tag{19}$$

where $\boldsymbol{u}^m$ is the displacement field and $\boldsymbol{v}^m$ the test function for material domains $m = 1, \ldots, N_m$.

The weak form of the total residual with stabilization and boundary terms is decomposed into the following five contributions:

$$\mathcal{R} = \mathcal{R}^{\text{U}} + \mathcal{R}^{\text{N}} + \mathcal{R}^{\text{G}} + \mathcal{R}^{\text{S}} = \mathbf{0}. \tag{20}$$

The weak form of the linear elastic governing equations $\mathcal{R}^{\text{U}}$ is defined as

$$\begin{aligned} \mathcal{R}^{\text{U}} = &\sum_m \int_{\Omega^m} \boldsymbol{\varepsilon}(\boldsymbol{v}^m) : \boldsymbol{\sigma}(\boldsymbol{u}^m)\, d\Omega \\ &- \sum_m \int_{\Gamma^m_N} \boldsymbol{v}^m \cdot \mathbf{t}_N \, d\Gamma. \end{aligned} \tag{21}$$

The Cauchy stress tensor is denoted by $\boldsymbol{\sigma}(\boldsymbol{u}^m) = \mathbf{D}^m \boldsymbol{\varepsilon}^m$ with $\mathbf{D}^m$ being the constitutive tensor, and $\boldsymbol{\varepsilon}^m$ the elastic strain tensor of material m. The elastic infinitesimal strain $\boldsymbol{\varepsilon}^m$ is computed by $\boldsymbol{\varepsilon}^m = \frac{1}{2}\left(\nabla(\boldsymbol{u}^m) + \nabla(\boldsymbol{u}^m)^T\right)$. Traction forces, $\mathbf{t}_N$, are applied on the Neumann boundary, Γ^m_N.

The residual is augmented with boundary and stabilization terms. To weakly enforce Dirichlet boundary and interface conditions, the residual in Eq. (21) is augmented with Nitsche's method, as introduced in Nitsche (1971) and Embar et al. (2010):

$$\begin{aligned} \mathcal{R}^{\text{N}} = \sum_{\Gamma^{m,n}} \Bigg(&- \int_{\Gamma^{m,n}} [\![\boldsymbol{v}]\!] \cdot \{\boldsymbol{\sigma}(\boldsymbol{u}) \cdot \mathbf{n}_\Gamma\} d\Gamma \\ &- \int_{\Gamma^{m,n}} \{\boldsymbol{\sigma}(\boldsymbol{v}) \cdot \mathbf{n}_\Gamma\} \cdot [\![\boldsymbol{u}]\!]\, d\Gamma \\ &+ \gamma^\Gamma \int_{\Gamma^{m,n}} [\![\boldsymbol{v}]\!] \cdot [\![\boldsymbol{u}]\!]\, d\Gamma \Bigg), \end{aligned} \tag{22}$$

where the jump operator is defined as $[\![\cdot]\!] = (\cdot)^m - (\cdot)^n$ and the averaging operator is defined as $\{\cdot\} = \gamma^m(\cdot)^m + \gamma^n(\cdot)^n$. The parameters γ^Γ, γ^m , and γ^n control the accuracy of enforcing the boundary and interface conditions.

For interface conditions, continuity of the displacement fields and balance of tractions must be satisfied at all interfaces $\Gamma^{m,n} = \Omega^m \cap \Omega^n \neq \emptyset$. We follow the work of Annavarapu et al. (2012) and define these constants as

$$\begin{aligned} \gamma^\Gamma &= 2c^\Gamma \frac{\text{meas}\,(\Gamma^{m,n})}{\text{meas}\,(\Omega^m)/E^m + \text{meas}\,(\Omega^n)/E^n}, \\ \gamma^m &= \frac{\text{meas}\,(\Omega^m)/E^m}{\text{meas}\,(\Omega^m)/E^m + \text{meas}\,(\Omega^n)/E^n}, \end{aligned} \tag{23}$$

where E^m is the Young's modulus of material m and the operator meas$(\cdot)$ measures the volume or area of the respective elemental subdomains or interface sections. The user-defined penalty parameter c^Γ determines how strongly the condition is enforced.

For Dirichlet boundary conditions, $(\cdot)^n = (\cdot)_D$ is the value imposed on the Dirichlet boundary Γ^m_D and the average stress simplifies to the stress in the material m, i.e., $\gamma^m = 1$ and $\gamma^n = 0$. We define $\gamma^\Gamma = c^\Gamma E^m / h$, where h is the edge length of the intersected elements. The weak enforcement of Dirichlet and interface conditions with Nitsche's method does not impact the optimal convergence rate of the analysis, see Noël et al. (2022) and Schmidt et al. (2023).

Face-oriented ghost stabilization is used to mitigate numerical instabilities caused by basis functions with small support. This may occur when an interface moves close to the boundary of the support of an interpolating basis function. While such behavior may result in ill-conditioning of the linear system, it may also result in imprecise spatial gradients, see de Prenter et al. (2017, 2019). We follow the work of Burman and Hansbo (2014) and augment the residual contribution of intersected elements with the following face-oriented ghost stabilization term, penalizing the jumps in spatial gradients across element faces:

$$\mathcal{R}^{\mathrm{G}} = \sum_{m} \sum_{F \in \mathcal{F}^m_{\mathrm{cut}}} \int_F \gamma^G \, E^m h^{\tilde{k}} \, [\![\nabla^p \boldsymbol{v}^m \cdot \mathbf{n}_F]\!][\![\nabla^p \boldsymbol{u}^m \cdot \mathbf{n}_F]\!] d\Gamma, \tag{24}$$

where F is an element face in the set $\mathcal{F}^m_{cut}$, which collects all faces between any two adjacent B-spline elements where at least one of the two elements is intersected. The parameter $\tilde{k}$ is defined as $\tilde{k} = 2(p-1)+1$. The outward normal of the face F is denoted by $\mathbf{n}_F$. The penalty parameter γ^G allows for control over the influence of the ghost penalty term. Due to the application of C^{p-1} continuous THB-spline bases, only jumps in gradients of order p must be penalized. Further details on the penalty term can be found in Burman and Hansbo (2014) and Noël et al. (2022) where a detailed study on the choice of γ^G is conducted. Based on this work, we choose a penalty parameter $\gamma^G = 0.001$.

6.2 Suppression of rigid body modes

Solid-void LS-TO problems may create topologically detached material regions at any point throughout the optimization process. Rigid body motions of such regions result in a singular linear system. These regions must be stabilized to prevent a rank deficient linear system. We adopt the selective structural springs approach presented by Villanueva and Maute (2014, 2017). This approach attaches distributed springs only to topologically detached material regions. These regions are identified by solving an auxiliary diffusion-convection problem for the indicator field θ, which is projected via a Heaviside function to zero or one, i.e., $\bar{\theta} = 0$ in regions connected to the mechanical support and $\bar{\theta} = 1$ in detached material regions. The contribution of this rigid body mode suppression approach to the elastic residual in Eq. (20) reads

$$\mathcal{R}^{\mathrm{S}} = \sum_{m} \int_{\Omega^m} \boldsymbol{v}^m \, \boldsymbol{u}^m \, r_s \, \frac{E^m}{h^2} \, \bar{\theta} \, d\Omega, \tag{25}$$

where r_s is the spring stiffness and is chosen to be $r_s = 10^{-4}$. The projected indicator field $\bar{\theta}$ allows for the stabilization of detached material regions while effectively deactivating the residual contribution for non-detached material regions.

6.3 XIGA-informed stress projection

Adaptively refined discretizations are of particular interest in stress-based TO due to the accuracy requirements of the stress computation. Especially in areas with large displacement gradients, adaptive refinement leads to a more accurate prediction of the displacement gradients. The calculation of accurate gradient-based fields, such as stresses, is often challenging. A post-processing step, such as stress smoothing, is typically required for C^0 continuous basis functions which include the linear B-spline basis functions studied in this work. Moreover, even C^{p-1} continuous bases might result in inaccurate gradients along the material boundary and interface due to basis functions with a small support. To compute smooth stresses, we adopt the global stress projection approach presented by Sharma and Maute (2018). The fundamental idea is to project an elemental stress measure predicted by the displacement field via the constitutive equations onto a scalar stress field. The face-oriented ghost stabilization approach outlined in the previous subsection provides additional stabilization for the stress field and prevents overestimation of stresses due to the penalization of jumps in stresses across element faces, see Sharma and Maute (2018). The scalar stress field measure is denoted by τ^m and the corresponding test function by χ^m. The von Mises stress is an example of a frequently used stress measure. The weak form augmented by a face-oriented ghost stabilization reads

$$\begin{aligned} 0 = \mathcal{R}^{\tau} + \mathcal{R}^{\mathrm{G}_\tau} = \\ \sum_{m} \Bigg(\int_{\Omega^m} \chi^m (\tau^m - \mathcal{C}(\boldsymbol{u}^m)) \, d\Omega \, + \\ \sum_{F \in \mathcal{F}_{\mathrm{cut}}} \int_F \gamma^{\tau} \, h^{\tilde{k}} \, [\![\nabla^p \chi^m \cdot \mathbf{n}_F]\!][\![\nabla^p \tau^m \cdot \mathbf{n}_F]\!] \, d\Gamma \Bigg), \end{aligned} \tag{26}$$

where the scalar stress measure $\mathcal{C}$ is a function of the components of the Cauchy stress tensor.

6.3.1 A priori error estimator

Error estimators are used to identify areas with inaccurate gradients which cause inaccurate stress predictions. For simplicity, we adopt an *a priori* error estimator approach inspired by Zienkiewicz and Zhu (1992) and compute an elemental error indicator field. This field identifies regions with large displacement curvatures by calculating the difference between the smoothed stress τ^m and the displacement-based stresses $\mathcal{C}(\boldsymbol{u}^m)$. The elemental least squares error is calculated as follows:

$$F_{ele} = \frac{\int_{\Omega^e} (\mathcal{C}(\boldsymbol{u}^m) - \tau^m)^2 \, d\Omega}{\int_{\Omega^e} \, d\Omega}. \tag{27}$$

We will use this error estimator as a refinement indicator to adaptively refine or coarsen a discretization. Without loss of generality, a variety of other error estimators may be used for the adaptation of the mesh, see for example Verfürth (1994).

7 Explicit level-set topology optimization

This section presents the utilized optimization formulation. Additionally, we recall the level-set regularization and hole-seeding strategies employed in this work.

7.1 Optimization problem formulation

We study multi-objective optimization problems that include minimal compliance designs subject to a mass constraint and mass minimization subject to a stress constraint. These optimization problems are formulated as

$$\begin{aligned} \min_{\boldsymbol{s}} \quad & \mathcal{Z}(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s})) + c_{\mathcal{P}_p} \frac{\mathcal{P}_p(\boldsymbol{s})}{\mathcal{P}_p^0} + c_{\mathcal{P}_\theta} \frac{\mathcal{P}_\theta(\boldsymbol{s})}{\mathcal{P}_\theta^0} \\ s.t. \quad & g(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s})) \leq 0, \end{aligned} \tag{28}$$

where $\mathcal{Z}(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s}))$ is the weighted sum of the compliance $\mathcal{S}$ and mass $\mathcal{M}$ evaluated over the solid domain and defined as

$$\mathcal{Z}(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s})) = c_{\mathcal{S}} \frac{\mathcal{S}(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s}))}{\mathcal{S}^0} + c_{\mathcal{M}} \frac{\mathcal{M}(\boldsymbol{s})}{\mathcal{M}^0}. \tag{29}$$

The initial strain energy and mass are denoted by $\mathcal{S}^0$ and $\mathcal{M}^0$, respectively. The compliance and mass objective contributions are weighted with the factors $c_{\mathcal{S}}$ and $c_{\mathcal{M}}$. To reduce spatial oscillations of the material interface, we add a perimeter contribution $\mathcal{P}_p$ to the objective. Furthermore, the objective is augmented by a regularization term $\mathcal{P}_\theta$. The purpose of the latter term will be explained in Sect. 7.2. The latter two contributions to the objective are normalized by their respective reference values $\mathcal{P}_p^0$ and $\mathcal{P}_\theta^0$. The weighting factors $c_{\mathcal{P}_p}$ and $c_{\mathcal{P}_\theta}$ control the perimeter penalty and regularization contributions. In Makhija and Maute (2014) and Geiss et al. (2019), it was shown that choosing these weights in the range of 10^{-2} provides a good compromise, i.e., a well-posed optimization problem with a desired, smooth interface geometry and a well-regularized level-set field is obtained, while the influence on the mechanical objective remains minimal.

In this work, we consider constraints on the mass $g_{\mathcal{M}}$ and on the stress g_τ. The mass constraint $g_{\mathcal{M}}$ is formulated as

$$g_{\mathcal{M}}(\boldsymbol{s}) = \frac{\int_\Omega d\Omega}{\mathcal{M}^0} - 1. \tag{30}$$

The stress constraint g_τ is formulated as

$$g_\tau(\boldsymbol{s}, \boldsymbol{u}(\boldsymbol{s})) = \int_\Omega \max((\frac{\tau}{\tau_{\text{ref}}} - 1),\ 0)^q \, d\Omega, \tag{31}$$

where τ_{ref} is the allowed von Mises stress. We choose an exponent $q = 2$ as it provides a continuous smooth gradient. Note that we do not impose any restrictions on the feature size to allow for the formation of small geometric features.

The design sensitivities are computed by the adjoint approach, see Sharma et al. (2017) and Noël et al. (2020).

7.2 Level-set regularization

The LS-TO approach may suffer from spurious oscillations of the LSF, see Geiss et al. (2019). We adopt the regularization scheme presented by Geiss et al. (2019) to create a smooth LSF and to control the spatial gradients at the material interface, improving the stability and convergence of the optimization problem. This regularization scheme aims to minimize the difference between the LSF and a globally defined target distance field $\tilde{\phi}$. The spatial gradient of the target field, projected onto the interface normal, is uniform along the material interface, and the target field assumes an upper or lower bound value away from the material interface. The target field, $\tilde{\phi}$, is constructed from a distance field whose computation is based on the heat method of Crane et al. (2013) and truncated via a sigmoid function.

Convergence of the LSF to the target field is promoted by adding a regularization contribution $\mathcal{P}_\theta$ to the objective, penalizing the squared difference between the LSF ϕ and the target field $\tilde{\phi}$, as well as their gradients:

$$\mathcal{P}_\theta = \frac{\int_\Omega w_\phi \left(\phi - \tilde{\phi}\right)^2 d\Omega}{\int_\Omega \phi_{\text{Bnd}}^2 d\Omega} + \frac{\int_\Omega w_{\nabla\phi} |\nabla\phi - \nabla\tilde{\phi}|^2 d\Omega}{\int_\Omega d\Omega}, \tag{32}$$

where ϕ_{Bnd} is the level-set bound value. Following the work of Barrera et al. (2020), the weights w_ϕ and $w_{\nabla\phi}$ allow for a customized regularization in the vicinity and away from the interface.

7.3 Hole-seeding via the density method

Classic LS-TO relies exclusively on shape sensitivities along the material interfaces to evolve the design. This allows for material domains to merge, split, or vanish. However, nucleation of new material or void regions within another material is not possible. As a result, the optimized designs are often highly dependent on the initial guess, see Van Dijk et al. (2013). Density-based TO approaches do not experience this limitation but may produce results with intermediate, non-physical densities.

In this work, we adopt the single-field hole-seeding approach presented by Barrera et al. (2020). This approach utilizes a level-set field to describe the material-void interface and a density field that defines the distribution of material properties within the material domain. The LSF and density fields are coupled such that the LSF creates holes if the density drops below a threshold.

The single-field approach defines a nodal volume fraction, or fictitious density field $\rho(\mathbf{x})$ as an explicit function of the LSF $\phi(\mathbf{x})$ within the material domain, Ω^+:

$$\rho(\mathbf{x}) = \begin{cases} \frac{\phi(\mathbf{x})-\phi_t}{\phi_{up}-\phi_t}, & \forall\, \mathbf{x} \in \Omega^+ \\ \text{not defined}, & \forall\, \mathbf{x} \in \Omega^-, \end{cases} \tag{33}$$

where ϕ_{up} is the upper bound of the LSF. Our physical model is only defined in the material domain, i.e., areas that are void are omitted in the analysis. Consequentially, we do not define the density for level-set values below the threshold ϕ_t as these regions are omitted in the analysis. The material properties are interpolated as functions of ρ.

Starting from a design domain entirely filled with material, Barrera et al. (2020) show that this method creates holes in the first TO iterations. To eliminate intermediate densities in the final design, the fictitious density is gradually shifted to a value of 1.0. as follows:

$$\tilde{\rho}(\mathbf{x}) = \phi_{sh} + (1 - \phi_{sh})\rho(\mathbf{x}), \tag{34}$$

where the shift parameter $\phi_{sh} = \phi_{sh}(D_{\text{it}})$. This parameter is defined in terms of the optimization iteration D_{it}:

$$\phi_{sh} = \begin{cases} 0.0, \ \forall\, D_{\text{it}} < D_{\text{it}_{\text{Start}}} \\ \hat{\phi}, \quad \forall\, D_{\text{it}_{\text{Start}}} \le D_{\text{it}} \le D_{\text{it}_{\text{End}}} \\ 1.0, \ \forall\, D_{\text{it}} > D_{\text{it}_{\text{End}}}, \end{cases} \tag{35}$$

with

$$\hat{\phi} = \phi^0_{sh} + (1 - \phi^0_{sh})(\frac{D_{\text{it}} - D_{it_{\text{Start}}}}{D_{\text{it}_{\text{End}}} - D_{\text{it}_{\text{Start}}}})^2, \tag{36}$$

where $\phi^0_{sh} = 10^{-3}$ is the initial lowest fictitious density value. The density shift is performed in the optimization iteration interval $[D_{\text{it}_{\text{Start}}}, D_{\text{it}_{\text{End}}}]$. As the shifted density increases to 1.0, the material properties in the entire material domain are the ones of the bulk material, and the optimization problem formulation corresponds to the one of pure LS-TO.

In this work, we interpolate the material properties by the Solid Isotropic Material with Penalization (SIMP) method, see Bendsøe and Sigmund (2004). The material density $\rho^m(\mathbf{x})$ is interpolated by

$$\rho^m(\mathbf{x}) = \rho^m_0\ \tilde{\rho}(\mathbf{x}) \tag{37}$$

and the Young's modulus $E^m(\mathbf{x})$ by

$$E^m(\mathbf{x}) = E^m_0\ \tilde{\rho}(\mathbf{x})^\beta, \tag{38}$$

where the density and the Young's modulus of the material m are denoted by ρ^m_0 and E^m_0, respectively. The SIMP exponent is denoted by β.

8 Numerical examples

In this section, we study the proposed TO framework with numerical problems that use multiple, different adaptively refined discretizations of design and state variable fields. Our examples include 2D and 3D compliance minimization problems with a mass constraint as well as mass minimization problems with a stress constraint. For all problems, a static linear elastic response is considered. All numerical examples require the solution of four linear systems per optimization iteration. More specifically, we solve the structural problem determining the physical response, two auxiliary diffusion problems for the regularization penalization term in Eq. (32), and one auxiliary diffusion problem for the selective springs, see Eq. (25). Additionally, the second example utilizes a global stress projection as presented in Sect. 6.3.

The optimization problems are solved by GCMMA from Svanberg (2002) and the required sensitivity analysis is performed following the adjoint approach. The parameters for the initial, lower, and upper asymptote adaptation in GCMMA are set to 0.05, 0.7, and 1.2, respectively. The GCMMA maximal step size is set to 0.02. The optimization problem is considered converged if the absolute change of the objective function relative to the mean of the objective function in the five previous optimization steps drops below 10^{-4} and the constraint is satisfied.

The systems of discretized governing equations are solved by a direct solver, PARDISO, for 2D problems, see Kourounis et al. (2018). A GMRES algorithm is utilized for 3D problems preconditioned by an ILU(2), see Heroux (2007). A relative drop of 10^{-10} in the linear residual is used as the convergence criterion for the iterative solver.

We use the following simple strategy to adapt the discretization in the course of the optimization process. The mesh is initially adapted after the first 20 optimization iterations and subsequently every 50 iterations after a previous mesh adaptation step. Additionally, mesh adaptation may be initiated to guarantee a uniform refinement level for all intersected elements of the non-conformal background mesh $\mathcal{H}_B$. Alternative strategies will be considered in future work. After each mesh adaptation step, the optimization algorithm is restarted from the previous design. Furthermore, the GCMMA is restarted with uniform lower and upper asymptotes for all variables which may alter the evolution of the design.

In the first example, we study the influence of refining the design variable field discretization. This study is performed for uniformly and adaptively refined meshes. The second example demonstrates the capabilities of our framework to exploit differently refined discretizations for each field. This work does not enforce any form of feature size constraint as they may narrow the design space and distort the results.

To quantify the computational gain using adaptively refined discretizations, we define the following measures: The efficiency factor E_{xiga} is defined as the ratio of the total number of displacement DOFs in the uniform XIGA model divided by the total number of displacement DOFs in the XIGA model for an adaptive discretization. This measure is indicative of the computational gains when adaptively refining the state variable field discretization and is defined as follows:

$$E_{\text{xiga}} = \frac{\sum_{k=1}^{N_{Opt}} \left(\#\text{DOFs}(u)^k_{\text{uniform}}\right)^{n_s}}{\sum_{k=1}^{N_{Opt}} \left(\#\text{DOFs}(u)^k_{\text{adaptive}}\right)^{n_s}}, \tag{39}$$

where $\#\text{DOFs}(u)^k$ is the number of displacement DOFs in the linear system per optimization iteration k. The exponent n_s is used to relate the number of DOFs to the computational effort in terms of floating-point operations or wall-clock time, see Woźniak et al. (2014). We use $n_s = 1$ for two-dimensional and $n_s = \frac{4}{3}$ for three-dimensional problems.

The peak resource requirement R_{xiga} is defined as the ratio of the maximum number of displacement DOFs in a linear system for a uniform state variable field discretization over the maximal number of displacement DOFs for the linear system of the adaptive discretization. This ratio quantifies the computational gain with respect to a given set of limited computational resources and is defined as

$$R_{\text{xiga}} = \frac{\max\left(\#\text{DOFs}_{\text{uniform}}\right)^{n_s}}{\max\left(\#\text{DOFs}_{\text{adaptive}}\right)^{n_s}}. \tag{40}$$

In addition, we monitor the ratio of the runtime of non-adaptive over adaptive processes. We perform a simple wall-clock time comparison T_{xiga} which takes into account the time spent in the forward and sensitivity analyses and which is defined as follows:

$$T_{\text{xiga}} = \frac{\sum_{k=1}^{N_{Opt}} t^k_{\text{uniform}}}{\sum_{k=1}^{N_{Opt}} t^k_{\text{adaptive}}}. \tag{41}$$

Table 1 Parameter list for design optimization problem

Parameter	Value
Initial strain energy $\mathcal{S}^0$	0.003
Allowable mass $\mathcal{M}^0$	1.0
Initial perimeter value $\mathcal{P}_P^0$	8.0
Initial regularization value $\mathcal{P}_\theta^0$	4.0
Strain energy weighting factor $c_{\mathcal{S}}$	1.0
Mass weighting factor $c_{\mathcal{M}}$	0.0
Perimeter weighting factor $c_{\mathcal{P}_P}$	0.01
Regularization weighting factor $c_{\mathcal{P}_\theta}$	0.03

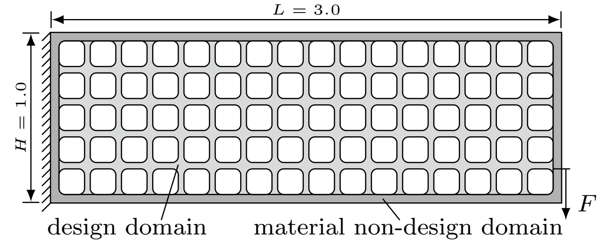


Fig. 7 Support structure within the prescribed frame with applied boundary conditions and load

While the runtime depends to some extent on the implementation of the framework, the runtime measure in Eq. (41) allows for a comparison of the relative computational cost. To compute a meaningful runtime measure, all computations must be run on the same hardware, the same number of parallel processors, and using the same domain decomposition.

8.1 Design of support structure within prescribed frame

The first example considers the design of the internal structure within a prescribed outer frame, as illustrated in Fig. 7. We optimize the structure for minimum compliance with a mass constraint of 33% of the total mass, where total mass is computed as if the entire domain is material. Additionally, regularization and perimeter penalties are added to the objective function. The initial problem parameters and weighting factors of the optimization formulation are summarized in Table 1 in self-consistent units.

The outer frame has a length $L = 3.0$ and a height $H = 1.0$. The frame is approximated using the signed-distance function of a hyperellipse with a semi-major and semi-minor axis of $r_a = 1.45$ and $r_b = 0.45$, respectively, and a hyperellipse exponent of 24.0. The entire left face is clamped, i.e., all displacement components are prescribed to zero. A distributed load of magnitude $\boldsymbol{F} = -1.0\,\boldsymbol{e}_{\text{y}}$ is applied at the right surface for y < 0.2. The frame and the solid material of the design domain are described by a linear

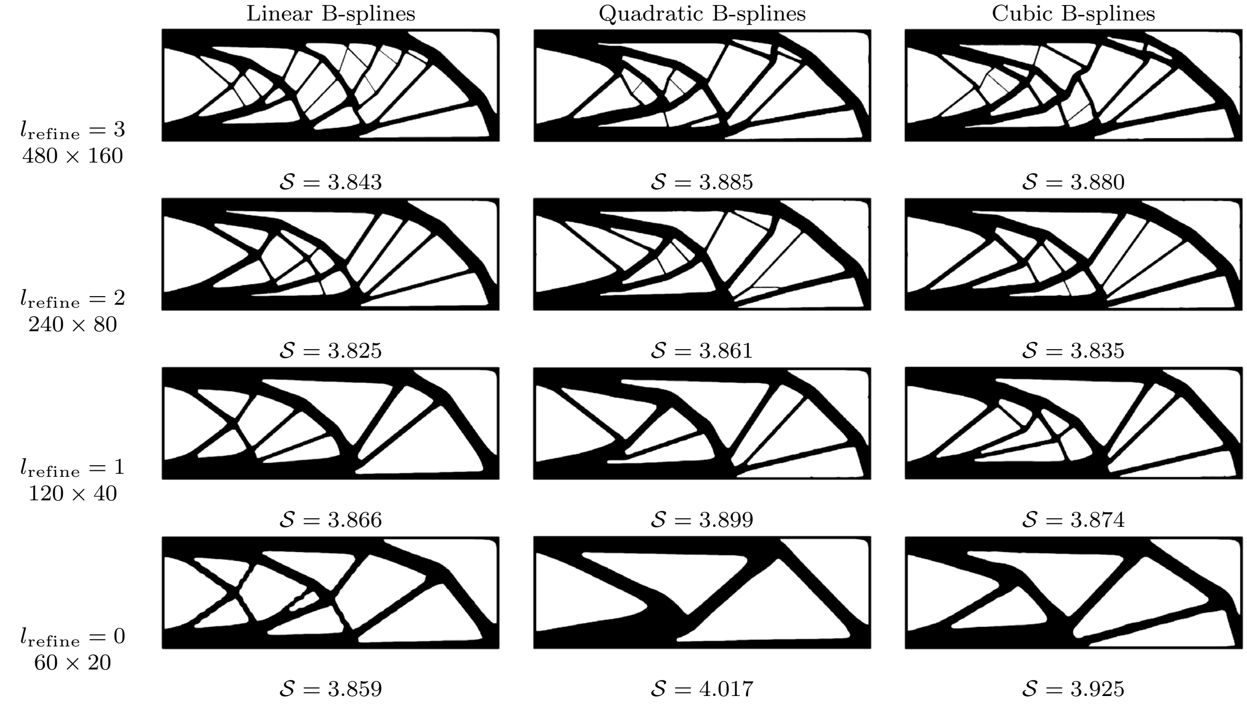


Fig. 8 Two-dimensional optimization results using uniformly refined meshes. Level-set field discretization: linear (left), quadratic (middle), and cubic (right)

elastic material model with Young's modulus $E = 10^4$ and Poisson ratio $\nu = 0.3$. All values are given in consistent units.

A 60×20 mesh is chosen as the initial coarse mesh. The compliance problem is evaluated for uniformly and adaptively refined discretizations. The state variable field is evaluated on a three-times uniformly or adaptively refined linear B-spline discretization. We present optimization results for linear state variable field discretizations and for a linear, quadratic, and cubic design variable field, each refined up to three times. The upper and lower bounds of the design variable field are set to $\pm 4h$, where h is the edge length of the most refined elements of the mesh used for the design variable field.

The design domain is initially seeded with 13×4 holes of radius 0.1 such that the mass constraint is satisfied at the beginning of the optimization process, see Fig. 7.

8.1.1 Uniform refinement

In this section, we study the influence of the design variable field discretization with uniform refinement. The state variable field is discretized on a three times refined mesh with 480×160 elements. We present results for a zero-, one-, two-, and three-times uniformly refined design variable field. The designs generated for a linear, quadratic, and cubic design variable field discretization are presented in Fig. 8. The solid phase is depicted in black. The convergence plots of the objective function are presented in Fig. 9.

While all designs present a similar strain energy value, the results show that refining the design variable discretization allows for a higher geometric complexity, including thinner structural members. In contrast, a coarser design variable field discretization leads to a smoother design. While these results are expected because an increase in the number of design variables increases the overall design freedom, it also highlights the ability of our framework to directly tune and limit the design space. In addition, the convergence rate of the optimization problem directly depends on the design variable field discretization level. A coarse discretization converges faster to an optimized design than a fine discretization. This increased convergence rate is a result of the smaller design space and the larger support of corresponding basis functions.

It should be noted that, while designs with a linear level-set discretization show the same behavior with respect to mesh refinement, they present non-smooth trusses and wavy material surfaces. The non-smooth nature of the trusses is especially pronounced when exploiting coarser design variable field discretizations. The LSFs of the final design for the non-refined linear and quadratic level-set discretizations are presented in Fig. 10. The C^0 inter-element continuity of the linear B-spline discretization does not allow for a smooth

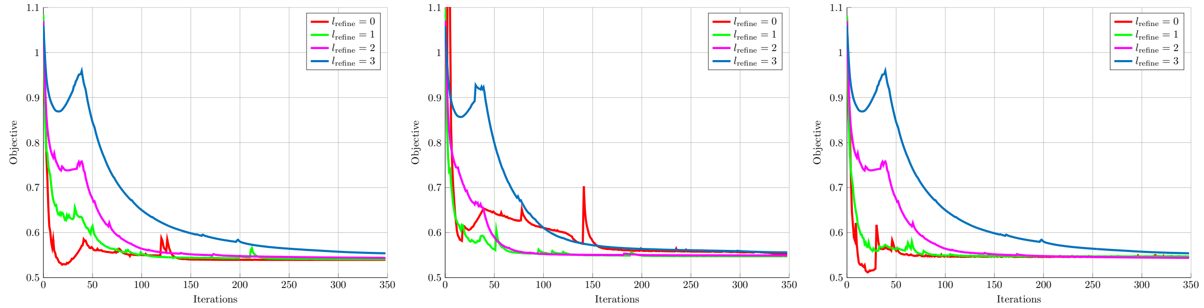


Fig. 9 Convergence of the objective for different uniform level-set refinement levels. Linear (left), quadratic (middle), and cubic (right) level-set interpolation

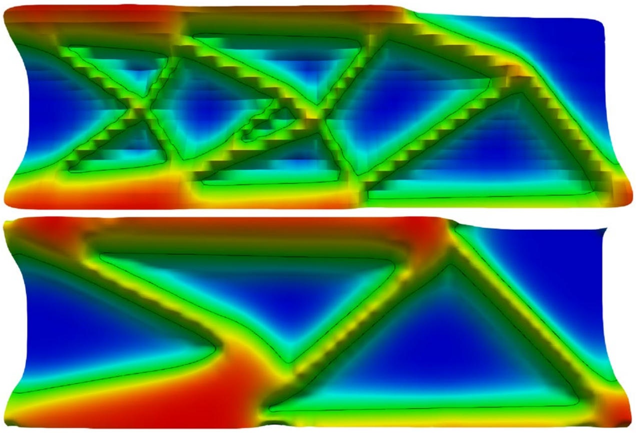


Fig. 10 Level-set field. Linear (top) and quadratic (bottom)

design. Higher-order B-spline discretizations, in contrast, allow for smooth first-order gradients which in turn result in smoother designs. Consequentially, we recommend using at least quadratic B-spline discretizations for the design variable field.

8.1.2 Adaptive refinement

Next, we study the influence of using adaptive state and design variable field discretizations. We apply the remeshing algorithm presented in Sect. 5.2. The zero isocontour of the LSF is used as the refinement criterion, see Refinement Criterion Algorithm (RCA) 4. The state variable field discretization and the non-conformal background mesh are always adaptively refined to a maximal refinement level $l = 3$ around the interface. To widen the refined zone around the material interface, all direct neighbor elements to the material interface are refined as well. This reduces the need for mesh adaptation triggered by the interface moving into a less refined region.

Algorithm 4 Isocontour of level-set-based RCA

```
for All elements $\mathcal{H}^e_B$ do
    if $\phi_{\max} > 0$ and $\phi_{\min} < 0$ then
        RO is *refine*
    else
        RO is *drop*
    end if
end for
```

The study is performed for design variable field discretizations, which are up to three times adaptively refined. We perform this study for linear, quadratic, and cubic hierarchical B-spline discretizations. The optimized designs are presented in Fig. 11, where the columns correspond to the polynomial order of the design variable field discretization and the rows correspond to the maximum allowed refinement level l_{refine}. For each layout, the solid is depicted in black, highlighting the design, while the refined mesh is presented in the void. The convergence plots of the objective function are shown in Fig. 12. The gains in computational efficiency achieved with the adaptive strategy are summarized in Table 2.

The adaptive approach yields similar results compared to the ones obtained for a uniform discretization. There is no significant decrease in the level of detail or complexity of the resultant geometries. Small differences can be explained through the restart of the optimization algorithm after adapting the mesh. The objective converges to a similar value for the uniform and adaptive cases. The convergence rate of the optimization problem depends on the mesh refinement level as the rate increases with a coarser mesh.

When comparing the computational efficiency of the uniform to adaptively refined state variable field discretizations, as shown in Table 2, a small reduction in computational cost is observed. The efficiency factor E_{xiga} shows that mesh adaptation leads to an overall reduction in computational cost. Bulkier designs, like those obtained

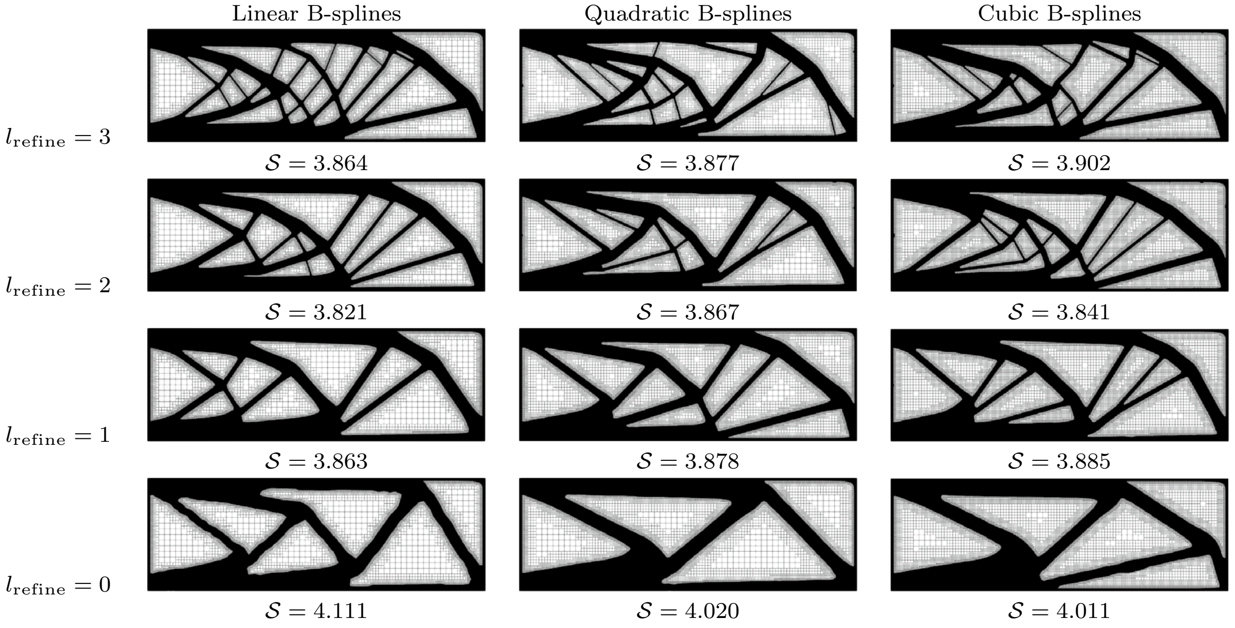


Fig. 11 Two-dimensional optimization results using adaptively refined meshes. Level-set field discretization: linear (left), quadratic (middle), and cubic (right)

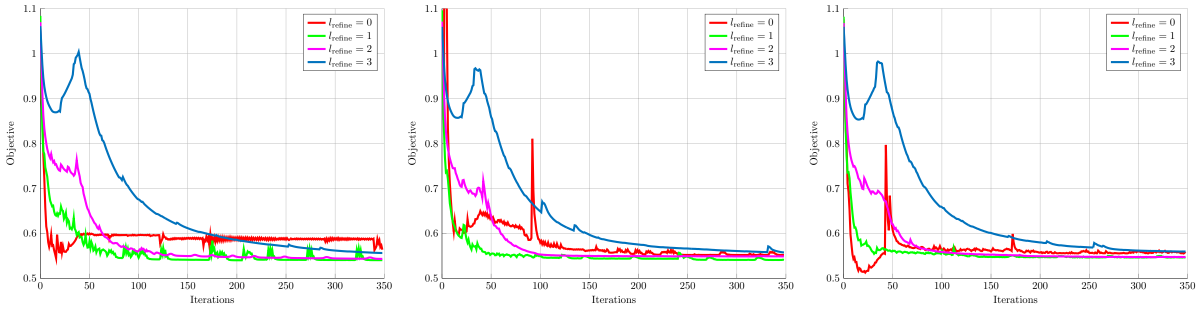


Fig. 12 Convergence of the objective for different adaptively refined level-set refinement levels. Linear (left), quadratic (middle), and cubic (right) level-set field discretizations

Table 2 Performance in terms of computational cost for designs in Fig. 11

	linear				quadratic				cubic			
l_{refine}	zero	one	two	three	zero	one	two	three	zero	one	two	three
E_{xiga}	1.65	1.54	1.42	1.37	1.71	1.49	1.43	1.35	1.66	1.45	1.38	1.35
R_{xiga}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
T_{xiga}	1.39	1.39	1.14	1.36	1.57	1.34	1.16	1.03	1.62	1.31	1.17	1.25

with a coarse design variable field discretization, especially benefit from adaptive refinement. This is expected as the material interface serves as the refinement criteria, whereas the coarser state variable field discretization is found in the interior of the material domain. The initial design is discretized over a uniform discretization which also results in the largest linear system. Consequently, the peak resource requirement R_{xiga} is found to be 1.0 for all optimization designs. The runtime ratios present a computational gain of up to 50% when using adaptively refined discretizations.

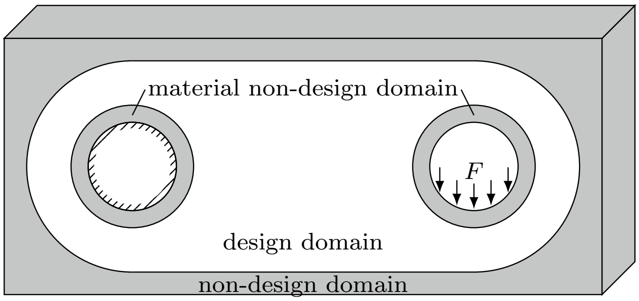


Fig. 13 Connector bracket with applied boundary conditions and load

8.2 Connector bracket with predefined features

This second example studies the ability of the presented optimization framework to independently and adaptively refine the discretizations of both design and state variable fields, each using different refinement criteria. This example considers the multi-material connector bracket with predefined features as shown in Fig. 13 in two and three dimensions. The optimization problem considers mass and compliance minimization subject to a stress constraint. Additionally, level-set regularization and perimeter penalties are added to the objective function. The initial two- and three-dimensional problem parameters and weighting factors of the optimization formulation are summarized in Table 3 in self-consistent units.

The size of the computational domain is 7 × 3 × 1.2. The design domain is fully immersed into the computational domain and consists of two semi-cylinders of radius $r_a = 1.237$ with their origin at coordinates (1.5, 1.5, 0.0) and (5.5, 1.5, 0.0), connected by two lines. Two non-design domains with prescribed material are defined by two hollow cylinders with an inner and outer radius of $r_b = 0.517$ and $r_c = 0.7173$, respectively. The displacement along the inner surface of the left cylinder is prescribed to zero. A uniform distributed load $\boldsymbol{F} = -1.0\, \boldsymbol{e}_{\mathrm{y}}$ is applied to the right cylinder's bottom half of the inner surface. The response of the cylinders and the connecting design is described by a linear elastic material model with a Young's modulus $E = 10^5$ and a Poisson ratio $\nu = 0.3$.

Table 3 Optimization parameters for the connector bracket problem

Parameter	2D setup	3D setup
Initial mass $\mathcal{M}^0$	1.0	1.0
Initial strain energy $\mathcal{S}^0$	9.0^{-4}	9.0^{-4}
Initial perimeter value $\mathcal{P}_P^0$	42.0	62.0
Initial regularization value $\mathcal{P}_\theta^0$	40	20
Strain energy weighting factor $c_{\mathcal{S}}$	1.0	0.5
Mass weighting factor $c_{\mathcal{M}}$	1.0	1.0
Perimeter weighting factor $c_{\mathcal{P}_P}$	0.3	0.1
Regularization weighting factor $c_{\mathcal{P}_\theta}$	0.2	0.05
Allowable stress τ_{ref}	8.0	10.0

8.2.1 Two-dimensional setup

First, we consider a simplified two-dimensional configuration as it allows for a more detailed visualization of the results. The domain is discretized with 70 × 30 elements on the coarsest mesh level which results in an element edge length of $h = 0.1$. The upper and lower level-set bounds are chosen to be $\pm 3h$. To nucleate holes, the density hole-seeding approach presented in Sect. 7.3 is employed. The initial LSF value is $\phi(\mathbf{x}) = 0.1$. A SIMP exponent of $\beta = 3.0$ is used to calculate the material property for the material in the design domain. The density shift is performed between optimization iterations 20 and 70. The design variable field is approximated with quadratic B-splines. We compare optimized designs and computational performances for linear, quadratic, and cubic displacement and stress field discretizations. Additionally, we highlight the reduction in computational cost compared to uniformly refined discretizations.

This example utilizes B-spline mesh $\mathcal{H}_1$ for the displacement and stress field discretizations and B-spline mesh $\mathcal{H}_2$ for the design variable and auxiliary diffusion field discretizations. B-spline mesh $\mathcal{H}_2$ is based on a quadratic polynomial order.

At the start of the optimization process, the meshes $\mathcal{H}_B$ and $\mathcal{H}_1$ are uniformly refined twice, while the mesh $\mathcal{H}_2$ is uniformly refined once. The first adaptive re-meshing is initialized after 20 optimization iterations. The meshes $\mathcal{H}_B$, $\mathcal{H}_1$, and $\mathcal{H}_2$ are adapted in every re-meshing step. In this example, we utilize two RCAs, 5 and 6, for the

Table 4 Parameter list for RCA

Parameter	Value
$l_{\mathrm{max}}^{\mathcal{H}_B}$	3
$l_{\mathrm{max}}^{\mathcal{H}_1}$	3
$l_{\mathrm{max}}^{\mathcal{H}_2}$	2
ϕ_t^{low}	− 0.9
ϕ_t^{up}	1.1
F_{ref}	0.1
ν_{refine}	1.0
ν_{hold}	0.3

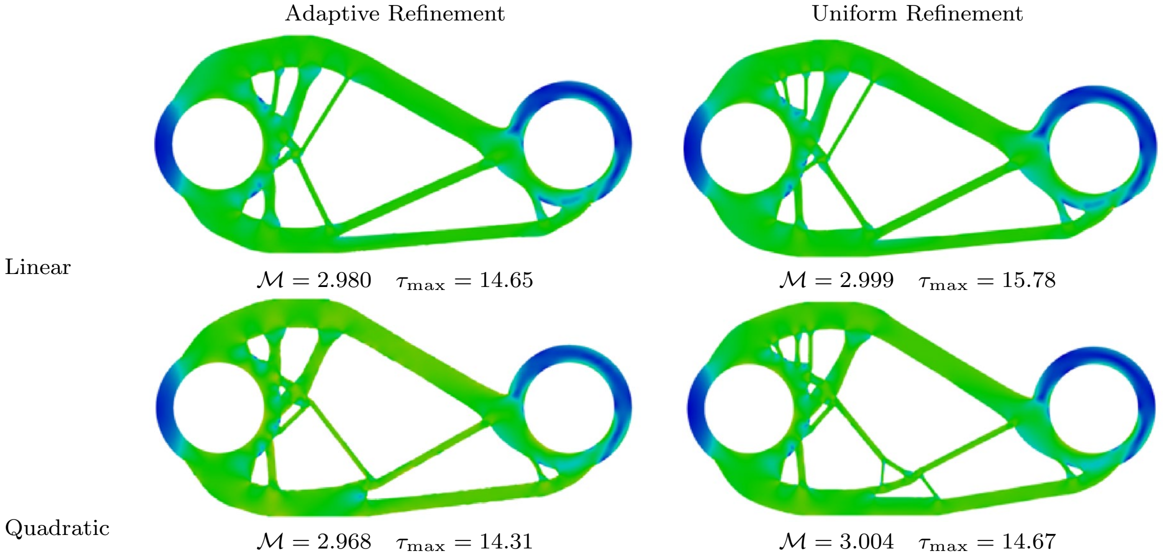


Fig. 14 Optimization results for a linear (top) and a quadratic (bottom) state variable field discretization; adaptively refined THB-spline discretization (left); uniform B-spline discretization (right)

computation of the RO. RCA 5 is evaluated recursively for all geometries, e.g., domain boundary, circular non-design domain, and the LSF. RCA 6 uses the elemental Zienkiewicz-Zhu inspired *a priori* error indicator field for refinement and coarsening. The non-conformal background mesh $\mathcal{H}_B$ is adapted based on both RCA 5 and 6. B-spline mesh $\mathcal{H}_1$ is adapted solely utilizing RCA 6, while B-spline mesh $\mathcal{H}_2$ is adapted by evaluating RCA 5. The parameters for refinement are given in Table 4.

An optimized design is generated on a uniform B-spline and a Lagrange discretization to establish a reference solution. In both cases, we start with the same initial uniform discretization as in the adaptive case. After 20 optimization iterations, one uniform level of refinement is added to all discretizations. This guarantees that the maximal refinement level of all discretizations for both the adaptive and the uniform case are identical.

Algorithm 5 Level-set-based RCA

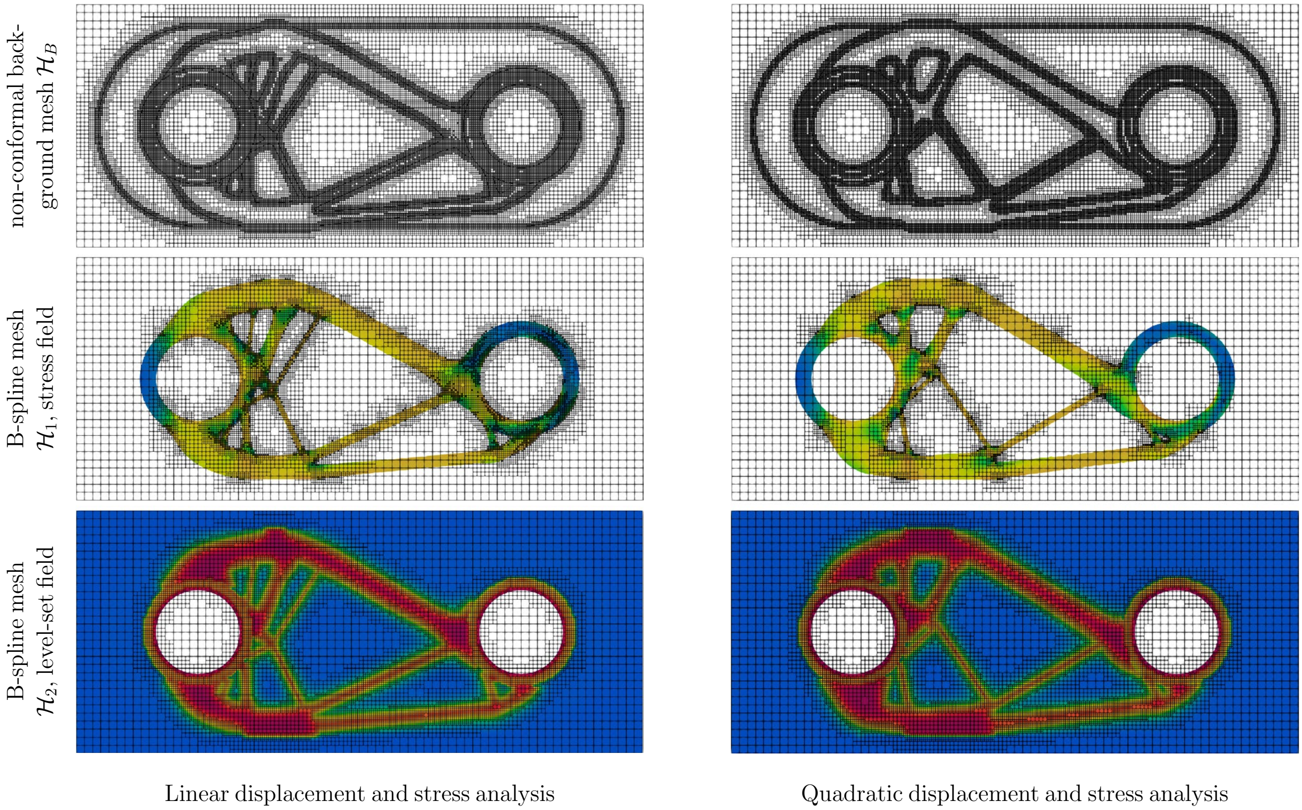
```
for All elements $\mathcal{H}_B^e$ do
    for All geometries $\phi$ do
        if $\phi_t^{\text{low}} \leq \phi \leq \phi_t^{\text{up}}$ then
            RO is refine
        else
            RO is drop
        end if
    end for
end for
```

Algorithm 6 Von Mises stress-based RCA

```
for All elements $\mathcal{H}_B^e$ do
    if $\frac{F_{\text{ele}}}{F_{\text{ref}}} \geq v_{\text{refine}}$ then
        RO is refine
    else if $\frac{F_{\text{ele}}}{F_{\text{ref}}} \geq v_{\text{hold}}$ then
        RO is hold
    else
        RO is coarsen
    end if
end for
```

The optimized designs for a linear and a quadratic state variable field discretization are presented in Fig. 14, along with the design performance in terms of the mass $\mathcal{M}$ and the maximum von Mises stress $\tau_{\max}$. The final design for an adaptively refined discretization is presented on the left, while the uniform B-spline discretization is presented on the right. For both cases, a qualitatively similar design is obtained and the maximum observed stress as well as the final mass does not show a significant difference.

In Fig. 15, we highlight optimized designs for a linear and a quadratic discretization of the displacement and stress fields, as well as the final adapted meshes $\mathcal{H}_B$, $\mathcal{H}_1$, and $\mathcal{H}_2$. The non-conformal background mesh $\mathcal{H}_B$ is refined at all material interfaces which allows for a precise representation of all geometries and is sufficiently refined to represent all approximating B-spline discretizations. The B-spline mesh $\mathcal{H}_1$ used for the discretization of the displacement and the global stress fields presents adaptive refinement in areas with large spatial gradients of the stress. This behavior is


non-conformal back-ground mesh $\mathcal{H}_B$
B-spline mesh $\mathcal{H}_1$, stress field
B-spline mesh $\mathcal{H}_2$, level-set field

Linear displacement and stress analysis

Quadratic displacement and stress analysis

Fig. 15 Optimized designs and corresponding meshes $\mathcal{H}_B$, $\mathcal{H}_1$, and $\mathcal{H}_2$ for a linear and a quadratic displacement discretization

Table 5 Performance in terms of computational cost for designs in Fig. 14

	linear		quadratic		cubic	
	THB-splines	Lagrange	THB-splines	Lagrange	THB-splines	Lagrange
E_{xiga}	2.91	2.91	5.23	18.75	4.68	–
R_{xiga}	2.22	2.22	2.54	9.45	2.52	–
T_{xiga}	1.35	1.35	1.47	1.81	1.45	–

expected as the Zienkiewizc-Zhu inspired elemental error estimator from Eq. (27) presents the largest errors in areas where the second spatial derivatives of the displacements are large. Comparing the linear and quadratic state variable field discretizations, we find that the quadratic mesh $\mathcal{H}_1$ is less refined overall. This observation aligns with our expectation due to the higher continuity of the quadratic B-spline discretization, which enables a more accurate calculation of stresses. In addition, we observe spots of locally refined regions at the material interface. This refinement results from a large elemental error caused by numerical instabilities due to basis functions with small support, as outlined in Sect. 6.1. This indicates that the ghost stabilization may not be sufficient. Locally adjusting the magnitude of the ghost stabilization penalty might help to further mitigate this issue.

The level-set discretization presents an adaptive refinement around the zero iso-contour of the LSF up to a maximal refinement level of $l_{\max}^{\mathcal{H}_2} = 2$. This choice was made based on the results in Sect. 8.1.2, as this refinement level is considered to be a good compromise to allow for a large amount of structural detail while also improving the convergence rate of the optimization problem.

Table 5 presents a comparison of the adaptively refined state variable fields approximated by THB-spline against uniformly refined B-spline and Lagrange discretizations for linear, quadratic, and cubic polynomial orders. Using an adaptively refined linear THB-spline discretization results in a factor 2.9 reduction in computational cost expressed through the efficiency factor E_{xiga} compared to a uniformly refined THB-spline or Lagrange discretization.[2] The further reduction

[2] The uniformly refined linear THB-spline and Lagrange bases are equivalent.

Table 6 Parameter list for RCA

Parameter	Value
$l_{\max}^{\mathcal{H}_B}$	2
$l_{\max}^{\mathcal{H}_1}$	2
$l_{\max}^{\mathcal{H}_2}$	1
ϕ_t^{low}	−0.9
ϕ_t^{up}	1.1
F_{ref}	1.0
v_{refine}	0.7
v_{hold}	0.2

in computational cost when using quadratic and cubic state variable discretizations confirms the expectation that higher inter-element continuity enables accurate calculation of stresses, which in turn, results in a coarser analysis discretization. Moreover, a significant reduction in computational cost, i.e., E_{xiga} larger than 10, is observed when comparing the higher-order B-Spline discretization to a uniform Lagrange discretization. This is expected as the higher-order B-spline space is a proper subspace of the Lagrange space of the same order defined over the same mesh, as B-spline basis functions exhibit a higher continuity than Lagrange basis functions. As a consequence, the higher-order B-spline space has fewer DOFs than the Lagrange space. Furthermore, we recall that the higher continuity only applies to regions inside of a material domain defined by the XIGA model as described in Sect. 3.3. In combination with the larger support of higher-order B-spline basis functions, this leads to a diminishing reduction in computational cost for fine geometric features.

In contrast to the previous example, the peak resource requirement R_{xiga} also shows a significant reduction in computational cost. In comparison to a classic Lagrange discretization, a factor of 9.0 is observed which may have a significant impact on the choice of solvers that may be used to solve the linear system.

8.2.2 Three-dimensional setup

This section extends the previous 2D configuration to 3D. The displacement field is approximated with a linear polynomial order. The optimization is initialized with uniformly, once refined meshes $\mathcal{H}_B$ and $\mathcal{H}_1$. The parameters for RCAs 5 and 6 are given in Table 6.

The optimization results generated with adaptively refined discretizations are presented in Fig. 16, along with vertical cross-sections of the design to visualize the internal material layout. The optimized design together with the non-conformal background mesh $\mathcal{H}_B$ and B-spline mesh $\mathcal{H}_1$ used

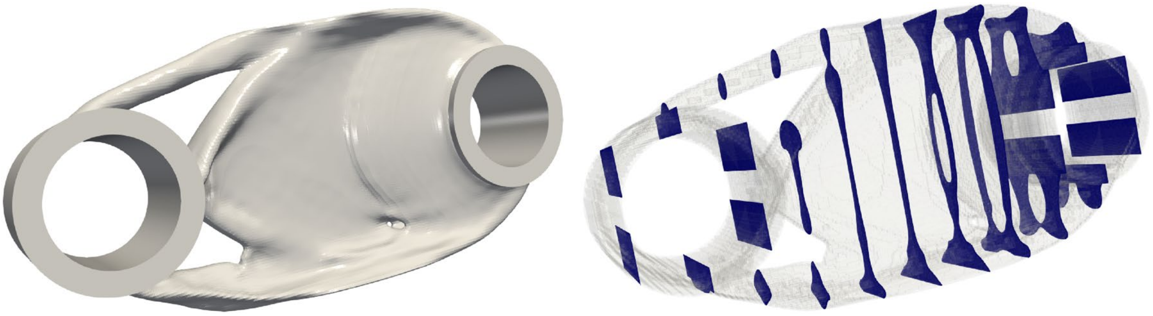


Fig. 16 Three-dimensional optimization result (left); final design and cross-sections (left)

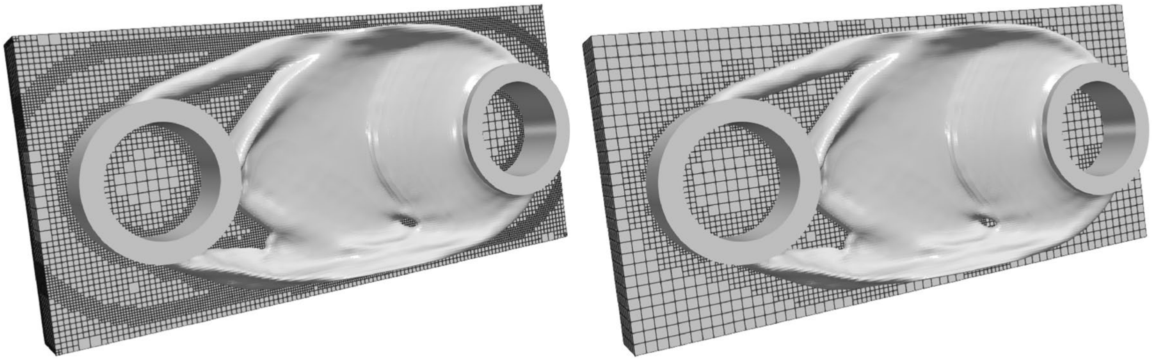


Fig. 17 Optimized design with non-conformal background mesh $\mathcal{H}_B$ (left) and structural analysis B-spline mesh $\mathcal{H}_1$ (right)

Table 7 Performance in terms of computational cost for designs in Fig. 16

	linear THB-splines
E_{xiga}	1.61
R_{xiga}	2.88

for the evaluation of the displacement field are presented in Fig. 17. The optimized design exhibits a noticeable similarity to the two-dimensional design except for the development of shear webs usually observed when solving optimization problems on fine, uniform meshes. The results show that the presented optimization strategy is applicable to three-dimensional problems.

Given the limitation in hardware resources, a direct comparison of the computational cost of adaptive versus uniform discretizations was not possible in this case. To estimate the cost of using uniform discretizations, we assume an identical evolution of the design during the optimization process, irrespective of using adaptive or uniform discretizations. Based on this assumption, a lower estimate can be calculated for the size of the linear systems considering the volume in each optimization iteration. The efficiency factors comparing the adaptive results with a uniform estimate are given in Table 7. This example shows that, in the three-dimensional setting, a reduction of computation cost with respect to the size of the linear system for both the efficiency factor E_{xiga} of 1.6 and peak resource requirement R_{xiga} of 2.8 is achieved.

9 Conclusion

This paper presents an adaptive discretization framework for level-set-based topology optimization. The geometry of the design is described implicitly by a level-set field. An immersed method, the XIGA, predicts the structural response and the design criteria. Design optimization problems often present different local resolution requirements for geometric and state variable fields. To address this problem, a hierarchical meshing strategy is presented that enables the construction of multiple, differently adaptively refined approximation spaces, utilizing truncated hierarchical B-spline discretizations. The numerical integration of the governing equation is performed on a single XIGA integration mesh of sufficient refinement and order to represent accurately all underlying approximation spaces. This XIGA integration mesh is constructed based on a non-conformal Lagrange background mesh. Lagrange extraction links the B-spline discretizations to the non-conformal background mesh.

The user-defined refinements considered in this work are performed based on geometric and physics-based error indicators. Adaptive re-meshing, including an increase of the maximum and a decrease of the minimum refinement level, is performed after a given number of iterations or when the material interface moves into a less refined region of the XIGA integration mesh. A simple mesh adaptation strategy is chosen to demonstrate the functionality of the presented framework but may need further research for practical applications. To allow for sufficient freedom in the design, the numerical examples either utilize an initial hole-seeding approach or nucleate holes based on a combined level-set/density approach. The proposed optimization strategy is applied to 2D and 3D structural problems. Differently adaptively refined meshes are constructed for the non-conformal background mesh, the level-set field discretization, the evaluation of the structural response, and auxiliary diffusion fields used for regularization and stabilization. Optimization results are computed for uniformly and adaptively refined discretizations up to a cubic polynomial order. Comparing these results provides insight into the influence of the discretization on the final design and the associated computational cost.

Numerical results suggest that separating the level-set and state variable field discretizations enables direct control over the accuracy and cost of the individual approximations. A coarser level-set field discretization promotes smoother designs with larger features and overall bulkier designs. Moreover, a higher-order B-spline discretization promotes a smooth level-set field due to a higher inter-element continuity. A coarser, higher-order level-set field discretization eliminates the need for a convolutional filter while providing similar functionality with respect to controlling the smoothness of the design. Similarly, the convergence rate of the optimization problem directly benefits from a coarser level-set field discretization due to the larger support of each design variable. Adapting the state variable field based on physics-based error estimators allows for a significant reduction in computational cost while meeting accuracy requirements on the prediction of the physical response, such as displacement and stress fields.

The adaptive discretization for design and state variable fields yields similar design performances when compared to a uniform, fine discretization. In addition, a reduction in computational cost is observed, achieved through a decreased size of the linear system in the response prediction. This reduction in computational cost is especially apparent when comparing the adaptively refined higher-order THB-spline discretization for state variable fields with a classic Lagrange discretization of the same polynomial order.

Possible future work may focus on extending the adaptive optimization framework to goal-oriented refinement strategies, a more sophisticated approach to control the geometric error, and advanced strategies for triggering mesh adaptation. Furthermore, the proposed framework could be applied to optimization problems that require predicting different phenomena, such as fluid flow or conjugate heat transfer. These problems may benefit from mesh adaptivity to resolve boundary layers.

Acknowledgements The first, second, fifth, and sixth authors acknowledge the support for this work from the Defense Advanced Research Projects Agency (DARPA) under the TRADES program (agreement HR0011-17-2-0022). The third, fifth, and sixth authors acknowledge the support from the National Science Foundation (NSF) under Grant 2104106. The first author acknowledges the partial auspice of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-JRNL-842737). K. Doble was supported by SANDIA through the contract PO 2120843. K. Maute was also partially supported by the AFOSR grant FA9550-20-1-0306. The opinions and conclusions presented in this paper are those of the authors and do not necessarily reflect the views of the sponsoring organization.

Funding This work was funded by Defense Advanced Research Projects Agency (DARPA) under the TRADES program (agreement HR0011-17-2-0022). The first author acknowledges the partial auspice of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-JRNL-842737). K. Doble was supported by SANDIA through the contract PO 2120843. K. Maute was also partially supported by the AFOSR grant FA9550-20-1-0306.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no Conflict of interest.

Replication of results The numerical studies presented in this paper used the open-source code MORIS, available at www.github.com/kkmaute/moris. Upon request, the authors will provide the input decks for the optimization problems presented in the paper.

References

Allaire G, Jouve F, Toader AM (2004) Structural optimization using sensitivity analysis and a level-set method. J Comput Phys. https://doi.org/10.1016/j.jcp.2003.09.032

Allaire G, Dapogny C, Frey P (2014) Shape optimization with a level set based mesh evolution method. Comput Methods Appl Mech Eng 282:22–53. https://doi.org/10.1016/j.cma.2014.08.028

Andreasen CS, Elingaard MO, Aage N (2020) Level set topology and shape optimization by density methods using cut elements with length scale control. Struct Multidiscip Optim 62(8):685–707. https://doi.org/10.1007/s00158-020-02527-1

Annavarapu C, Hautefeuille M, Dolbow JE (2012) A robust Nitsche's formulation for interface problems. Comput Methods Appl Mech Eng 225–228:44–54. https://doi.org/10.1016/j.cma.2012.03.008

Babuška I, Melenk JM (1997) The partition of unity method. Int J Numer Meth Eng 40:727–758

Barrera JL, Geiss MJ, Maute K (2020) Hole seeding in level set topology optimization via density fields. Struct Multidiscip Optim 61:1319–1343. https://doi.org/10.1007/s00158-019-02480-8

Bendsøe MP, Sigmund O (2004) Typology optimization, 2nd edn. Springer, New York

Belytschko T, Black T (1999) Elastic crack growth in finite elements with minimal remashing. Int J Numer Meth Eng 45:601–620

Belytschko T, Gracie R, Ventura G (2009) A review of extended/generalized finite element methods for material modeling. Modell Simul Mater Sci Eng. https://doi.org/10.1088/0965-0393/17/4/043001

Bourdin B (2001) Filters in topology optimization. Int J Numer Meth Eng 50:2143–2158. https://doi.org/10.1002/nme.116

Bruggi M, Verani M (2011) A fully adaptive topology optimization algorithm with goal-oriented error control. Comput Struct 89:1481–1493. https://doi.org/10.1016/j.compstruc.2011.05.003

Burger M (2003) A framework for the construction of level set methods for shape optimization and reconstruction. Interfaces Free Bound 5:301–329

Burman E (2012) A penalty-free nonsymmetric nitsche-type method for the weak imposition of boundary conditions. SIAM J Numer Anal 50:1959–1981. https://doi.org/10.1137/10081784X

Burman E, Hansbo P (2014) Fictitious domain methods using cut elements: Iii. A stabilized nitsche method for stokes' problem. ESAIM Math Model Num Anal 48:859–874

Burman E, Elfverson D, Hansbo P, Larson MG, Larsson K (2019) Cut topology optimization for linear elasticity with coupling to parametric nondesign domain regions. Comput Methods Appl Mech Eng 350:462–479. https://doi.org/10.1016/j.cma.2019.03.016

Cai S, Zhang W, Zhu J, Gao T (2014) Stress constrained shape and topology optimization with fixed mesh: a b-spline finite cell method combined with level set function. Comput Methods Appl Mech Eng 278:361–387. https://doi.org/10.1016/j.cma.2014.06.007

Christiansen AN, Nobel-Jørgensen M, Aage N, Sigmund O, Bærentzen JA (2014) Topology optimization using an explicit interface representation. Struct Multidiscip Optim 49:387–399. https://doi.org/10.1007/s00158-013-0983-9

Coffin P, Maute K (2016) A level-set method for steady-state and transient natural convection problems. Struct Multidiscip Optim 53:1047–1067. https://doi.org/10.1007/s00158-015-1377-y

Costa JCA, Alves MK (2003) Layout optimization with h-adaptivity of structures. Int J Numer Meth Eng 58:83–102. https://doi.org/10.1002/nme.759

Crane K, Weischedel C, Wardetzky M (2013) Geodesics in heat. ACM Trans Gr 32:1–11. https://doi.org/10.1145/2516971.2516977

D'Angella D, Kollmannsberger S, Rank E, Reali A (2018) Multilevel bézier extraction for hierarchical local refinement of isogeometric analysis. Comput Methods Appl Mech Eng 328:147–174. https://doi.org/10.1016/j.cma.2017.08.017

de Boor C (1972) On calculating with b-splines. J Approx Theory 6:50–62

de Prenter F, Verhoosel CV, van Zwieten GJ, van Brummelen EH (2017) Condition number analysis and preconditioning of the finite cell method. Comput Methods Appl Mech Eng 316:297–327. https://doi.org/10.1016/j.cma.2016.07.006

de Prenter F, Verhoosel CV, van Brummelen EH (2019) Preconditioning immersed isogeometric finite element methods with application to flow problems. Comput Methods Appl Mech Eng 348:604–631. https://doi.org/10.1016/j.cma.2019.01.030

de Troya MAS, Tortorelli DA (2020) Three-dimensional adaptive mesh refinement in stress-constrained topology optimization. Struct Multidiscip Optim 62:2467–2479. https://doi.org/10.1007/s00158-020-02618-z

Deaton JD, Grandhi RV (2000) A survey of structural and multidisciplinary continuum topology optimization: post. Struct Multidiscip Optim 49:1–38. https://doi.org/10.1007/s00158-013-0956-z

Duysinx P, Miegroet L, Jacobs T, Fleury C (2006) Generalized shape optimization using x-fem and level set methods. IUTAM symposium on topological design optimization of structures, machines and materials. pp. 23-32

Embar A, Dolbow J, Harari I (2010) Imposing dirichlet boundary conditions with nitsche's method and spline-based finite elements. Int J Numer Meth Eng 83(8):877–898. https://doi.org/10.1002/nme.2863

Eschenauer HA, Olhoff N (2001) Topology optimization of continuum structures: a review. Appl Mech Rev 54:331–390. https://doi.org/10.1115/1.1388075

Evans JA, Bazilevs Y, Babuška I, Hughes TJ (2009) n-Widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method. Comput Methods Appl Mech Eng 198:1726–1741. https://doi.org/10.1016/j.cma.2009.01.021

Gao J, Xiao M, Zhang Y, Gao L (2020) A comprehensive review of isogeometric topology optimization: methods, applications and prospects. Chin J Mech Eng (Engl Ed). https://doi.org/10.1186/s10033-020-00503-w

Garau EM, Vázquez R (2018) Algorithms for the implementation of adaptive isogeometric methods using hierarchical b-splines. Appl Numer Math 123(1):58–87. https://doi.org/10.1016/j.apnum.2017.08.006

Geiss MJ, Barrera JL, Boddeti N, Maute K (2019) A regularization scheme for explicit level-set xfem topology optimization. Front Mech Eng 14:153–170. https://doi.org/10.1007/s11465-019-0533-2

Giannelli C, Jüttler B, Speleers H (2012) Thb-splines: The truncated basis for hierarchical splines. Computer Aided Geometric Design 29:485–498. https://doi.org/10.1016/j.cagd.2012.03.025

Gournay FD (2006) Velocity extension for the level-set method and multiple eigenvalues in shape optimization. SIAM J Control Optim 45:343–367. https://doi.org/10.1137/050624108

Guest JK, Smith GLC (2010) Reducing dimensionality in topology optimization using adaptive design variable field. J Numer Meth Eng. https://doi.org/10.1002/nme

Heroux M A (2007) Aztecoo tm user guide a

Hughes TJ, Cottrell JA, Bazilevs Y (2005) Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. Comput Methods Appl Mech Eng 194:4135–4195. https://doi.org/10.1016/j.cma.2004.10.008

Hughes TJ, Reali A, Sangalli G (2008) Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: comparison of p-method finite elements with k-method nurbs. Comput Methods Appl Mech Eng 197(9):4104–4124. https://doi.org/10.1016/j.cma.2008.04.006

Hughes TJ, Evans JA, Reali A (2014) Finite element and nurbs approximations of eigenvalue, boundary-value, and initial-value problems. Comput Methods Appl Mech Eng 272(4):290–320. https://doi.org/10.1016/j.cma.2013.11.012

Hvejsel CF, Lund E (2011) Material interpolation schemes for unified topology and multi-material optimization. Struct Multidiscip Optim 43(6):811–825. https://doi.org/10.1007/s00158-011-0625-z

Jahangiry HA, Tavakkoli SM (2017) An isogeometrical approach to structural level set topology optimization. Comput Methods Appl Mech Eng 319:240–257. https://doi.org/10.1016/j.cma.2017.02.005

Kourounis D, Fuchs A, Schenk O (2018) Toward the next generation of multiperiod optimal power flow solvers. IEEE Trans Power Syst 33:4005–4014. https://doi.org/10.1109/TPWRS.2017.2789187

Lazarov BS, Sigmund O (2011) Filters in topology optimization based on helmholtz-type differential equations. Int J Numer Meth Eng 86(5):765–781. https://doi.org/10.1002/nme.3072

Makhija D, Maute K (2014) Numerical instabilities in level set topology optimization with the extended finite element method. Struct Multidisc Optim 49:185–197. https://doi.org/10.1007/s00158-013-0982-x

Maute K, Ramm E (1995) Adaptive topology optimization. Structural Optimization 10:100–112. https://doi.org/10.1007/BF01743537

Mittal R, Iaccarino G (2005) Immersed boundary methods. Annu Rev Fluid Mech 37:239–261. https://doi.org/10.1146/annurev.fluid.37.061903.175743

Moës N, Béchet E, Tourbier M (2006) Imposing dirichlet boundary conditions in the extended finite element method. Int J Numer Meth Eng 67:1641–1669. https://doi.org/10.1002/nme.1675

Nana A, Cuillière JC, Francois V (2016) Towards adaptive topology optimization. Adv Eng Softw 100:290–307. https://doi.org/10.1016/j.advengsoft.2016.08.005

Nguyen-Xuan H (2017) A polytree-based adaptive polygonal finite element method for topology optimization. Int J Numer Meth Eng 110:972–1000. https://doi.org/10.1002/nme.5448

Nitsche J (1971) Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. Abh Math Semin Univ Hambg 36:9–15. https://doi.org/10.1007/BF02995904

Noël L, Schmidt M, Messe C, Evans JA, Maute K (2020) Adaptive level set topology optimization using hierarchical b-splines. Struct Multidiscip Optim 62:1669–1699. https://doi.org/10.1007/s00158-020-02584-6

Noël L, Schmidt M, Doble K, Evans JA, Maute K (2022) Xiga: an extended isogeometric analysis approach for multi-material problems. Comput Mech. https://doi.org/10.1007/s00466-022-02200-y

Osher S (1988) Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. J Comput Phys 49:12–49

Peskin CS (2002) The immersed boundary method. Acta Numer 11:479–517. https://doi.org/10.1017/S0962492902000077

Qian X (2013) Topology optimization in b-spline space. Comput Methods Appl Mech Eng 265:15–35. https://doi.org/10.1016/j.cma.2013.06.001

Ramm E, Maute K, Schwarz S (1998) Adaptive topology and shape optimization. *Comput Mech*. pp 1–22

Schillinger D, Ruthala PK, Nguyen, (2016) Lagrange extraction and projection for nurbs basis functions: a direct link between isogeometric and standard nodal finite element formulations. International. https://doi.org/10.1002/nme

Schmidt M, Noël L, Doble K, Evans JA, Maute K (2023) Extended isogeometric analysis of multi-material and multi-physics problems using hierarchical b-splines. Comput Mech 71(6):1179–1203. https://doi.org/10.1007/s00466-023-02306-x

Schmidt MR, Barrera JL, Mittal K, Swartz KE, Tortorelli DA (2024) Level-set topology optimization with PDE generated conformal meshes. Struct Multidiscip Optim. https://doi.org/10.1007/s00158-024-03870-3

Sethian JA, Wiegmann A (2000) Structural boundary design via level set and immersed interface methods. J Comput Phys 163(9):489–528. https://doi.org/10.1006/jcph.2000.6581

Sharma A, Maute K (2018) Stress-based topology optimization using spatial gradient stabilized XFEM. Struct Multidiscip Optim 57:17–38. https://doi.org/10.1007/s00158-017-1833-y

Sharma A, Villanueva H, Maute K (2017) On shape sensitivities with heaviside-enriched XFEM. Struct Multidiscip Optim 55(2):385–408. https://doi.org/10.1007/s00158-016-1640-x

Sigmund O, Maute K (2013) Topology optimization approaches a comparative review. Struct Multidisc Optim 48:1031–1055. https://doi.org/10.1007/s00158-013-0978-6

Stainko R (2006) An adaptive multilevel approach to the minimal compliance problem in topology optimization. Commun Numer Methods Eng 22:109–118. https://doi.org/10.1002/cnm.800

Svanberg K (2002) A class of globally convergent optimization methods based on conservative convex separable approximations.

SIAM J Optim 12:555–573. https://doi.org/10.1137/S1052623499362822

Terada K, Asai M, Yamagishi M (2003) Finite cover method for linear and non-linear analyses of heterogeneous solids. Int J Numer Meth Eng 58:1321–1346. https://doi.org/10.1002/nme.820

Van Dijk NP, Maute K, Langelaar M, Keulen FV (2013) Level-set methods for structural topology optimization: a review. Struct Multidisc Optim 48:437–472. https://doi.org/10.1007/s00158-013-0912-y

Verfürth R (1994) A posteriori error estimates for nonlinear problems. finite element discretizations of elliptic equations. Math Comput 62:445–475

Villanueva CH, Maute K (2014) Density and level set-XFEM schemes for topology optimization of 3-d structures. Comput Mech 54:133–150. https://doi.org/10.1007/s00466-014-1027-z

Villanueva CH, Maute K (2017) Cutfem topology optimization of 3d laminar incompressible flow problems. Comput Methods Appl Mech Eng 320:444–473. https://doi.org/10.1016/j.cma.2017.03.007

Wang Y, Benson DJ (2016) Isogeometric analysis for parameterized LSM-based structural topology optimization. Comput Mech 57:19–35. https://doi.org/10.1007/s00466-015-1219-1

Wang MY, Wang X, Guo D (2003) A level set method for structural topology optimization. Comput Methods Appl Mech Eng 192:227–246

Wang S, de Sturler E, Paulino G H (2010) Dynamic adaptive mesh refinement for topology optimization. arXiv:1009.4975

Wang Y, Kang Z, He Q (2014) Adaptive topology optimization with independent error control for separated displacement and density fields. Comput Struct 135:50–61. https://doi.org/10.1016/j.compstruc.2014.01.008

Wang YQ, He JJ, Luo Z, Kang Z (2013) An adaptive method for high-resolution topology design. Acta Mechanica Sinica/Lixue Xuebao 29:840–850. https://doi.org/10.1007/s10409-013-0084-4

Wang Y, Wang Z, Xia Z, Poh LH (2018) Structural design optimization using isogeometric analysis: a comprehensive review. CMES Comput Model Eng Sci 117:455–507. https://doi.org/10.31614/cmes.2018.04603

Woźniak M, Kuźnik K, Paszyńnski M, Calo VM, Pardo D (2014) Computational cost estimates for parallel shared memory isogeometric multi-frontal solvers. Comput Math Appl 67:1864–1883. https://doi.org/10.1016/j.camwa.2014.03.017

Zienkiewicz OC, Zhu JZ (1992) The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity. Int J Numer Meth Eng 33:1365–1382. https://doi.org/10.1002/nme.1620330703

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.