Johann Radon Institute for
Computational and Applied Mathematics
Austrian Academy of Sciences (ÖAW)

ÖAW AUSTRIAN ACADEMY OF SCIENCES

RICAM
JOHANN·RADON·INSTITUTE
FOR COMPUTATIONAL AND APPLIED MATHEMATICS

Guaranteed and sharp a posteriori error estimates in isogeometric analysis

S.K. Kleiss, S. Tomar

GUARANTEED AND SHARP A POSTERIORI ERROR ESTIMATES IN ISOGEOMETRIC ANALYSIS

STEFAN K. KLEISS AND SATYENDRA K. TOMAR

ABSTRACT. We present functional-type a posteriori error estimates in isogeometric analysis. These estimates, derived on functional grounds, provide guaranteed and sharp upper bounds of the exact error in the energy norm. Moreover, since these estimates do not contain any unknown/generic constants, they are fully computable, and thus provide quantitative information on the error. By exploiting the properties of non-uniform rational B-splines, we present efficient computation of these error estimates. The numerical realization and the quality of the computed error distribution are addressed. The potential and the limitations of the proposed approach are illustrated using several computational examples.

1. INTRODUCTION

The geometry representations in finite element methods (FEM) and computer aided design (CAD) have been developed independent of each other, and are optimized for the purposes within their respective fields. As a consequence, the representations are different from each other, and a transfer of geometry information from CAD to FEM programmes (and vice versa) requires a transformation of geometry data. These transformations are, in general, not only costly, but also prone to approximation errors, and may require manual input.

Isogeometric analysis (IGA), introduced by Hughes et al. [22], see also [11], aims at closing this gap between FEM and CAD. The key observation is that it is a widespread standard in CAD to use geometry representations based on non-uniform rational B-splines (NURBS), and that these NURBS basis functions have properties which make them suitable as basis functions for FEM. Instead of transforming the geometry data to a conventional FEM representation, the original geometry description is used directly, and the underlying NURBS functions are used as basis for the discrete solution. This way, the geometry is represented *exactly* in the sense that the geometry obtained from CAD is not changed. Thus, the need for data transformation is eliminated, and furthermore, the exact representation from the coarsest mesh is preserved throughout the refinement process. IGA has been thoroughly studied and analyzed (see, e.g., [3, 7, 12, 23, 37]), and its potential has been shown by successful applications to a wide range of problems (see, e.g., [5, 6, 10, 17, 28]).

As mentioned above, the most widely used spline representations in CAD are based on NURBS. The straightforward definition of NURBS basis functions leads to a tensor-product structure of the basis functions, and thus of the discretization. Since naive mesh refinement in a tensor-product setting has global effects, the development of local refinement strategies for isogeometric analysis is a subject of current active research. Such local refinement techniques include, for example, T-splines [4, 27, 34, 35, 36], truncated hierarchical B-splines (THB-splines) [21], polynomial splines over hierarchical T-meshes (PHT-splines) [13, 39], and locally-refineable splines (LR-splines) [14].

The issue of adaptive, local refinement is closely linked to the question of efficient a posteriori error estimation (see, e.g., [1, 33] for a general overview on error estimators). In the light of adaptive refinement, an error estimator has to identify the areas where further refinement is needed due to the local error being significantly larger than in the rest of the domain. Hence, an accurate indication of the error distribution is essential. Another important objective in computing a posteriori error estimates is to address the *quality assurance*, i.e., to quantify the error in the computed solution with certain degree of *guarantee*. However, a posteriori error estimation in isogeometric analysis is still in an infancy stage. To the best of the authors' knowledge, the only published results are [15, 24, 38, 39, 40, 41]. In [15, 38], the authors used a posteriori error estimates based on hierarchical bases [2]. Its reliability and efficiency is subjected to the saturation assumption on the (enlarged) underlying space and the constants in the strengthened Cauchy inequality. As the authors remarked, the first assumption is critical and its validity

depends on the considered example. Moreover, an accurate estimation of constants in the strengthened Cauchy inequality requires the solution of generalized minimum eigenvalue problem. As noted in [24, Page 41], this approach delivers *less than satisfactory* results. In [24, 39, 40, 41], the authors used the residual-based a posteriori error estimates, which require the computation of constants in Clement-type interpolation operators. Such constants are mesh (element) dependent, often generic/unknown or incomputable for general element shape; and the global constant often over-estimates the local constants, and thus the exact error. This fact has been explicitly stated by the authors in [24, Pages 42-43] and in [39, Remark 1]. Furthermore, Zienkiewicz-Zhu type a posteriori error estimates are based on post-processing of approximate solutions, and depend on the superconvergence properties of the underlying basis. To the best of authors' knowledge, superconvergence properties for B-splines (NURBS) functions are not yet known. Summarily, in general situations, the reliability and efficiency of these methods often depend on undetermined constants, which is not suitable for quality assurance purposes.

In this paper, we present *functional-type a posteriori error estimates* for isogeometric discretizations. These error estimates, which were introduced in [30, 31, 32] and have been studied for various fields (see [33] and the references therein), provide guaranteed, sharp and fully computable bounds (without any generic undetermined constants). These estimates are derived on purely functional grounds (based on integral identities or functional analysis) and are thus applicable to any conforming approximation in the respective space. For elliptic problems with the weak solution $u \in H_0^1(\Omega)$, these error bounds involve computing an auxiliary function $y \in H(\Omega, \mathrm{div})$. In order to get a sharp estimate, this function y is computed by solving a *global* problem. This could be perceived as a drawback when compared to error estimation techniques which rely on local computations and are thus apparently cheaper. However, as briefly explained above, our emphasis is not only on adaptivity, but also on *quantifying the error in the computed solution* (and thus guaranteeing the quality of the computed solution). Therefore, the associated cost should be weighed against the stated objectives. To the best of authors' knowledge, there is no other, particularly cheaper, method available which can fulfil these objectives in general situations. In this paper, we will elaborate how such estimates can be computed efficiently by a proper set-up of the global problem.

Two aspects motivate the application of functional-type error estimates in IGA. Firstly, unlike the standard Lagrange basis functions, NURBS basis functions of degree p are, in general, globally C^{p-1}-continuous. Hence, NURBS basis functions of degree $p \geq 2$ are, in general, at least C^1-continuous, and therefore, their gradients are automatically in $H(\Omega, \mathrm{div})$. Thereby, we avoid constructing complicated functions in $H(\Omega, \mathrm{div})$, in particular for higher degrees (see, e.g., [8, 9, 18]). Secondly, since the considered problem is solved in an isogeometric setting, an efficient implementation of NURBS basis functions is readily available, which can be used to construct the above mentioned function y. Hence, applying the technique of functional-type a posteriori error estimation in a setting that relies only on the use of already available NURBS basis functions is greatly appealing.

The remainder of this paper is organized as follows. In Section 2, we define the model problem, and recall the definition and some important properties of B-spline and NURBS basis functions. In Section 3, we first recall functional-type a posteriori error estimates and known implementation issues. Then, we derive a quality criterion and the local error indicator. In Section 4, we discuss a cost-efficient realization of the proposed error estimator using an illustrative numerical example. Further numerical examples are presented in Section 5, and finally, conclusions are drawn in Section 6.

2. PRELIMINARIES

In order to fix notation and to provide an overview, we define the model problem and recall the definition and some aspects of isogeometric analysis in this section.

2.1. **Model Problem.** Let $\Omega \subset \mathbb{R}^2$ be an open, bounded and connected Lipschitz domain with boundary $\partial\Omega$. We shall consider the following model problem:

Find the scalar function $u : \overline{\Omega} \to \mathbb{R}$ such that

$$(1) \qquad \begin{aligned} -\operatorname{div}(A\nabla u) &= f && \text{in } \Omega, \\ u &= u_D && \text{on } \Gamma_D = \partial\Omega, \end{aligned}$$

where A, f and u_D are given data. We assume that A is a symmetric positive definite matrix and has a positive inverse A^{-1}, and that there exist constants $c_1, c_2 > 0$ such that

$$(2) \qquad c_1|\xi|^2 \leq A\xi \cdot \xi \leq c_2|\xi|^2, \quad \forall \xi \in \mathbb{R}^2.$$

Then, the norms

$$(3) \qquad \|v\|_A^2 = \int_\Omega Av \cdot v \, dx, \quad \|v\|_{\bar{A}}^2 = \int_\Omega A^{-1}v \cdot v \, dx,$$

are equivalent to the L^2-norm $\|v\|^2 = \int_\Omega v \cdot v \, dx$. The weak form of problem (1) can be written as follows:

Find $u \in V_g$, such that

$$(4) \qquad a(u, v) = f(v), \quad \forall v \in V_0,$$

where $V_0 \subset H^1(\Omega)$ contains the functions which vanish on Γ_D, and $V_g \subset H^1(\Omega)$ contains the functions satisfying the Dirichlet boundary conditions $u = u_D$ on Γ_D. We assume that the problem data A, f and u_D are given such that the bilinear form $a(\cdot, \cdot)$ is bounded, symmetric and positive definite, and that $f(\cdot)$ is a bounded linear functional. The energy norm of a function v is given by $\|\nabla v\|_A = \sqrt{a(v, v)}$. Note that we have considered the Dirichlet problem only for the sake of simplicity. Functional-type error estimates can be easily generalized to problems with mixed boundary conditions, see, e.g., [26, 33].

We discretize the problem (4) in the standard way by choosing a finite-dimensional space $V_h \subset V_g$ and looking for a *discrete solution* $u_h \in V_h$. This leads to a linear system of equations of the form

$$(5) \qquad \underline{K}_h \underline{u}_h = \underline{f}_h,$$

where $\underline{K}_h$ is the stiffness matrix induced by the bilinear form $a(\cdot, \cdot)$, $\underline{f}_h$ is the load vector, and $\underline{u}_h$ is the coefficient vector of the discrete solution u_h.

2.2. **B-Splines, NURBS and Isogeometric Analysis.** We briefly recall the definition of B-spline basis functions and NURBS mappings. We only provide the basic definitions and properties relevant for the scope of this paper. For detailed discussions of NURBS basis functions, geometry mappings and their properties, we refer to, e.g., [11, 12, 22, 29] and the references therein. The following standard definitions and statements can also be found there.

Let p be a non-negative *degree* and let $s = (s_1, \ldots, s_m)$ be a *knot vector* with $s_i \leq s_{i+1}$ for all i. We consider only *open knot vectors*, i.e., knot vectors s where the multiplicity of a knot is at most p, except for the first and last knot which have multiplicity $p + 1$. For simplicity, we assume that $s_1 = 0$ and $s_m = 1$, which can be easily achieved by a suitable scaling. The $n = m - p - 1$ univariate *B-spline basis functions* $B_{i,p}^s : (0, 1) \to \mathbb{R}$, $i = 1, \ldots, n$, are defined recursively as follows:

$$B_{i,0}^s(\xi) = \begin{cases} 1 & \text{for} \quad s_i \leq \xi < s_{i+1} \\ 0 & \text{else} \end{cases}$$

$$B_{i,p}^s(\xi) = \frac{\xi - s_i}{s_{i+p} - s_i} B_{i,p-1}^s(\xi) + \frac{s_{i+p+1} - \xi}{s_{i+p+1} - s_{i+1}} B_{i+1,p-1}^s(\xi).$$

Whenever a zero denominator appears in the definition above, the corresponding function $B_{i,p}^s$ is zero, and the whole term is considered to be zero. For open knot vectors, the first and last basis function are interpolatory at the first and the last knot, respectively. The derivatives of B-spline basis functions are given by the following formula:

$$\partial_\xi B_{i,p}^s(\xi) = \frac{p}{s_{i+p} - s_i} B_{i,p-1}^s(\xi) - \frac{p}{s_{i+p+1} - s_{i+1}} B_{i+1,p-1}^s(\xi).$$

B-spline basis functions of degree p are, in general, globally C^{p-1}-continuous. In the presence of multiple knots, the continuity reduces according to the multiplicity, i.e., if a knot appears k times, the continuity of a B-spline basis function of degree p at that knot is C^{p-k}.

Let $\{B_{i,p}^s\}_{i=1}^{n_1}$ and $\{B_{j,q}^t\}_{j=1}^{n_2}$ be two families of B-spline basis functions defined by the degrees p and q, and the open knot vectors

$$s = (s_1, \ldots, s_{n_1+p+1}), \ t = (t_1, \ldots, t_{n_2+q+1}),$$

respectively. We denote the set of all double-indices (i, j) by

$$\mathcal{I}_R = \{(i, j) : \ i \in \{1, \ldots, n_1\}, j \in \{1, \ldots, n_2\}\}.$$

3

Let $w_{(i,j)}$, $(i,j) \in \mathcal{I}_R$, be positive *weights*. The *bivariate NURBS basis functions* $R_{(i,j)}(\xi_1, \xi_2)$, $(i,j) \in \mathcal{I}_R$ are defined as follows:

$$R_{(i,j)}(\xi_1, \xi_2) = \frac{w_{(i,j)} B_{i,p}^s(\xi_1) B_{j,q}^t(\xi_2)}{\sum_{(k,\ell) \in \mathcal{I}_R} w_{(k,\ell)} B_{k,p}^s(\xi_1) B_{\ell,q}^t(\xi_2)}.$$

The continuity of the B-spline basis functions is inherited by the NURBS basis functions. Note that B-splines can be seen as a special case of NURBS with all weights being equal to one. Hence, we will not distinguish between these two and we will only use the term *NURBS* in the remainder of the paper.

The set of functions

$$\hat{V}_h = \text{span}\{R_{(i,j)}, \ (i,j) \in \mathcal{I}_R\},$$

associated with the *parameter domain* $\hat{\Omega} = (0,1)^2$, is uniquely determined by the degrees p and q, the knot vectors s and t, and the weights w. Given the set of functions $\hat{V}_h$ and a *control net* of *control points* $P_{(i,j)} \in \mathbb{R}^2$, where $(i,j) \in \mathcal{I}_R$, the two-dimensional *NURBS-surface* $G : \hat{\Omega} \to \Omega$ is defined by

$$(6) \qquad G(\xi_1, \xi_2) = \sum_{(i,j) \in \mathcal{I}_R} R_{(i,j)}(\xi_1, \xi_2) \, P_{(i,j)}.$$

We refer to $\Omega = G(\hat{\Omega})$ as the *physical domain*. We assume that the geometry mapping is continuous and bijective (i.e., not self-penetrating), which are natural assumptions for CAD-applications.

In isogeometric analysis, the isoparametric principle is applied by using the same basis functions for the discrete solution u_h which are used for representing the geometry. For detailed discussion, we refer the reader to, e.g., [11, 12, 22]. The discrete solution u_h on the physical domain Ω is represented as follows:

$$(7) \qquad u_h(x) = \sum_{(i,j) \in \mathcal{I}_R} u_{(i,j)} \left(R_{(i,j)} \circ G^{-1} \right)(x),$$

where $u_{(i,j)} \in \mathbb{R}$ are real-valued coefficients which form the coefficient vector $\underline{u}_h$. The discrete functions space is thus defined by

$$V_h = \text{span}\{R_{(i,j)} \circ G^{-1}, \ (i,j) \in \mathcal{I}_R\}.$$

The initial mesh, and thereby the basis functions on this initial mesh, are assumed to be given via the geometry representation of the computational domain, i.e., the initial discretization is already determined by the problem domain. The exact representation of the geometry on the initial (coarsest) level is preserved in the process of mesh refinement.

As mentioned in the introduction, the straightforward definition of NURBS basis functions, leads to a tensor-product structure of the discretization, which is the focus of this paper. Nevertheless, the error estimator presented herein is also applicable to local refinement techniques (e.g., T-splines, THB-splines, PHT-splines, LR-splines, see Section 1) since it is derived purely on functional grounds.

3. Functional-type a Posteriori Error Estimates

In the first two parts of this section, we will discuss the well-known theoretical upper bound for the error in the energy norm (see, e.g., [30, 31, 32, 33]), and we recall how to minimize this upper bound in order to get a sharp error estimate (see, e.g., [25, 26]). Thereafter, in Section 3.3, we will derive a quality criterion from the discussed theory. We will comment on the realization in the isogeometric context in Section 4.

3.1. **Guaranteed Upper Bound for the Error.** The starting point for the proposed method is the following main result, which gives an upper bound for the error in the energy norm. It can be found, e.g., in [31, 32, 33].

Theorem 3.1. *Let C_Ω be the constant in the Friedrich's type inequality $\|v\| \le C_\Omega \|\nabla v\|_A$, $\forall v \in V_0$. Let u be the exact solution of the problem (4), and let $u_h \in V_h$ be an approximate solution. Then, the following estimate holds:*

$$(8) \qquad \|\nabla u - \nabla u_h\|_A \le \|A \nabla u_h - y\|_{\bar{A}} + C_\Omega \|\operatorname{div} y + f\|,$$

where y is an arbitrary vector-valued function in $H(\Omega, \operatorname{div})$, and the norms are as defined in (3).

The constant C_Ω depends only on the domain Ω and the coefficient matrix A (but not on the underlying mesh), see, e.g., [26, 33]. Note that C_Ω can be computed either numerically or, if one can find a domain $\Omega_\square \supset \Omega$, where $\Omega_\square$ is a square domain with side-length ℓ, then $C_\Omega \leq c_2 \frac{\ell}{\pi\sqrt{d}}$, where d is the dimension and c_2 is the constant in (2).

Note that, if we choose y via the (unknown) exact solution $y = A\nabla u$, both sides of (8) coincide. Hence, the estimate is sharp in the sense that, for any fixed u_h, we can find a function y such that the upper bound is as close to the exact error as desired. The estimate given in Theorem 3.1 is a guaranteed and fully computable upper bound for any conforming approximation $u_h \in V_g$.

In the following, we describe some approaches to construct y and discuss their relative merits.

3.1.1. *Post-processing of u_h.* It is possible to obtain *good* error indicators by constructing function y by some post-processing of the discrete solution u_h, see [26, 33] and the references therein. Consider, for example, $A = I$ and that $u_h \in V_h$ has been computed with NURBS basis functions of degree $p \geq 2$. Then, since $u_h \in C^{p-1}$, we have $\nabla u_h \in (C^{p-2})^2 \subset H(\Omega, \mathrm{div})$. Choosing $y = \nabla u_h$ will thus result in

$$(9) \qquad \|\nabla u - \nabla u_h\| \leq C_\Omega \|\Delta u_h + f\|.$$

To show the efficiency of the estimator (9), we present an illustrative numerical example. This example, referred to as *Example 1* in the remainder, is chosen due to a smoothly varying function (without any large gradients) in both directions.

Example 1 (Sinus Function on the Unit Square): In this numerical example, the computational domain is the unit square $\Omega = (0, 1)^2$ and u_h is a piecewise quadratic function in both directions, i.e., $p = q = 2$. The coefficient matrix is constant, $A = I$, and the exact solution is given by

$$u = \sin(6\pi x)\sin(3\pi y).$$

The right-hand-side f and the (homogeneous) boundary conditions u_D are determined by the prescribed exact solution u.

Once we have calculated $\eta_Q := \|\Delta u_h + f\|_Q$ for each cell Q of the mesh, we can compare the local errors and choose a criterion for selecting cells which will be marked for further refinement. Typically, one chooses a threshold Θ and marks all cells Q for refinement, where the local error is above this threshold. There are several possibilities for determining Θ, e.g., the bulk-criterion proposed in [16]. For simplicity, we choose a percentage ψ and mark a cell Q for refinement, if

$$(10) \qquad \eta_Q > \Theta, \text{ where } \Theta = (100 - \psi)\text{-percentile of } \{\eta_Q\}_Q.$$

The α-percentile of a set $\mathcal{A} = \{a_1, \ldots, a_\nu\}$ denotes the value $\bar{a}$ below which α percent of all values a_i fall. For example, if we choose $\psi = 20\%$ in (10), then Θ is chosen such that $n_Q > \Theta$ holds for 20% of all cells Q.

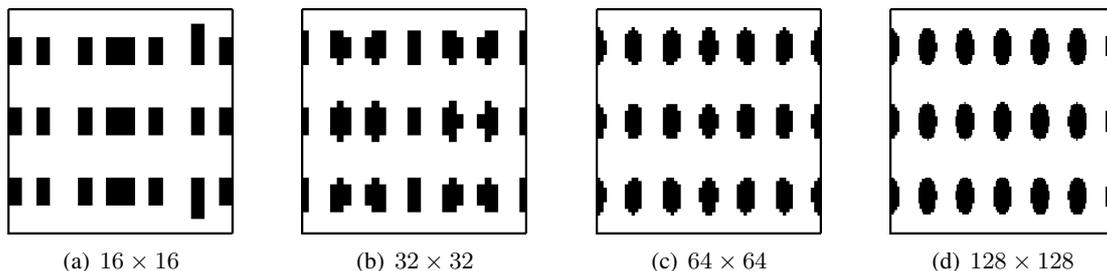(a) 16×16 (b) 32×32 (c) 64×64 (d) 128×128

FIGURE 1. Cells marked by exact error with $\psi = 20\%$ in Example 1.

In Figure 1, we present the cells marked for refinement by the exact error. The cells marked for refinement by the majorant given in (9) are presented in Figure 2. We see that starting from the mesh 32×32, the majorant is able to nicely capture the refinement pattern of exact error. However, from a closer look at the convergence of the exact error and the majorant, see Figure 3, we find that though such an estimate is a guaranteed upper bound and very cheap to compute, it over-estimates the exact error, and its convergence is slower than the exact error (due to a lack of proper scaling, different operators acting on u_h on both sides).

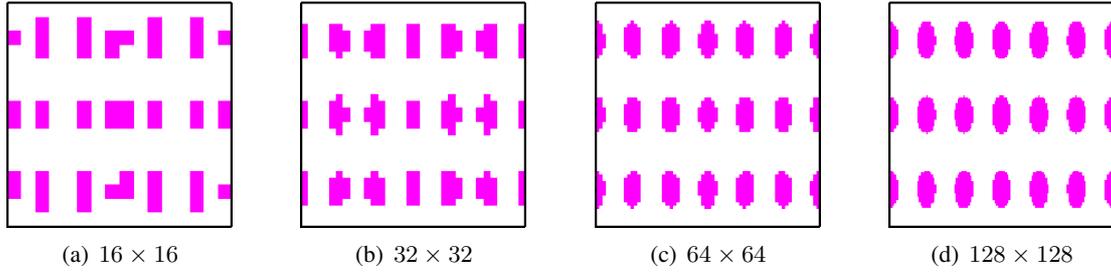| (a) 16×16 | (b) 32×32 | (c) 64×64 | (d) 128×128 |

FIGURE 2. Cells marked by error estimator with $\psi = 20\%$ in Example 1, $y_h = \nabla u_h$.
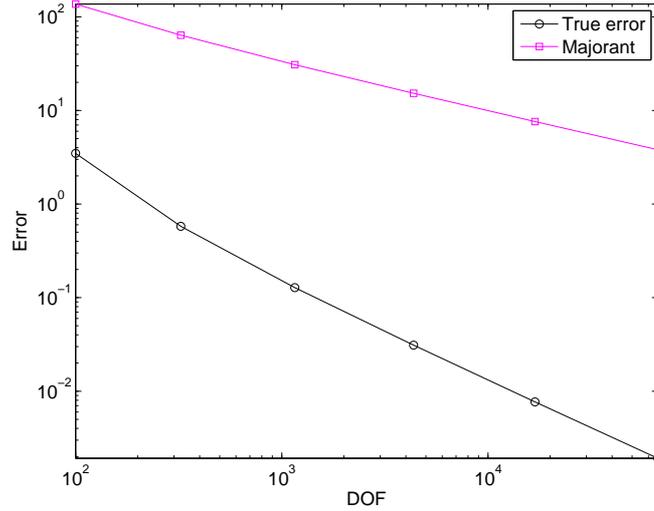

FIGURE 3. Convergence of exact error and the majorant (9) for Example 1.

3.1.2. *Patch-wise interpolation.* We now discuss a quasi interpolation (patch-wise) approach. Considering the 1D case first, we fix a certain degree of freedom (DOF) u_i, and set up a local problem that involves only those DOF/basis functions which have an influence on u_i. For univariate B-splines (or NURBS), the support of a basis function $B_{i,p}^s$ extends through $p+1$ knot spans (namely the interval (s_i, s_{i+p+1})). The total number of basis functions whose support intersect the support of $B_{i,p}^s$ is $2p+1$, i.e. the functions with the indices $k \in \{i-p, \ldots, i+p\}$. Considering only these basis functions, we obtain a local system matrix which we denote by $\underline{L}_h^i$. This matrix is of size $(2p+1) \times (2p+1)$, and has a bandwidth of $2p+1$. For example, if $p = 3$, the matrix $\underline{L}_h^i$ is a 7×7-matrix, and its structure is given by

$$
\begin{pmatrix}
* & * & * & * & 0 & 0 & 0 \\
* & * & * & * & * & 0 & 0 \\
* & * & * & * & * & * & 0 \\
* & * & * & * & * & * & * \\
0 & * & * & * & * & * & * \\
0 & 0 & * & * & * & * & * \\
0 & 0 & 0 & * & * & * & *
\end{pmatrix}.
$$

For each DOF u_i, such a system (with appropriate right hand side) is needed to be solved. The same principle applies for bivariate basis functions. Assume that the polynomial degrees are same in both directions and are denoted by p. Due to the tensor-product structure, the support of a bivariate NURBS basis function $R_{(i,j)}$ intersects the support of a total of $(2p+1)^2$ basis functions. Hence, the size of the local system matrix $\underline{L}_h^{(i,j)}$ is $(2p+1)^2 \times (2p+1)^2$. Furthermore, if the basis functions are vector-valued with d components, the local system matrix is of size $d(2p+1)^2 \times d(2p+1)^2$. For example, for vector-valued functions of degree 3 with two components, the local system size is 98×98. The solution of the corresponding system will give two coefficients which can be used for both the components associated

with the double-index (i, j). Unfortunately, this approach is neither a cheap one (to compute y) nor does it result in desired efficiency indices in the proximity of 1, and therefore, we do not present its results.

3.1.3. *Global minimization.* In order to obtain a sharp estimate (and not just an indicator), therefore, one has to find a function y which minimizes the right-hand-side of (8). For minimizing the estimate (8) numerically, we first rewrite the estimate in the following form

(11) $\qquad \|\nabla u - \nabla u_h\|_A^2 \leq (1 + \beta)\|A\nabla u_h - y\|_{\bar{A}}^2 + (1 + \frac{1}{\beta})C_\Omega^2\|\operatorname{div} y + f\|^2 =: M_\oplus^2(y, \beta),$

where $\beta > 0$ is a free parameter [26, 33]. Note that the upper bound in (11) holds true for *any* fixed $y \in H(\Omega, \operatorname{div})$ and $\beta > 0$. Hereinafter, for simplicity, we will refer to $M_\oplus^2(y, \beta)$ as the *majorant*. Introducing

(12) $\qquad \begin{aligned} a_1 &= 1 + \beta, & a_2 &= (1 + \frac{1}{\beta})C_\Omega^2, \\ B_1 &= \|A\nabla u_h - y\|_{\bar{A}}^2, & B_2 &= \|\operatorname{div} y + f\|^2, \end{aligned}$

we can briefly write the majorant as

(13) $$M_\oplus^2(y, \beta) = a_1 B_1 + a_2 B_2.$$

The *efficiency index*, defined by

(14) $$I_{\text{eff}} = \frac{M_\oplus(y, \beta)}{\|\nabla u - \nabla u_h\|_A},$$

indicates how close the calculated majorant is to the exact error. The closer I_{eff} is to 1, the better the estimate. Therefore, obtaining a *sharp* estimate requires to find $y \in H(\Omega, \operatorname{div})$ and $\beta > 0$ as solutions to the global minimization problem

(15) $$\min_{y \in H(\Omega, \operatorname{div}),\ \beta > 0} M_\oplus^2(y, \beta).$$

The technique for finding such minimizing parameters y and β will be discussed in Sections 3.2 and 4.2. Before proceeding further, we give the following Lemma 3.3, which can be found in [33, Prop. 3.10]. It provides an analytical result on the sharpness of the bound $M_\oplus^2(y, \beta)$. For later reference, we also sketch the proof.

Definition 3.2. *A sequence of finite-dimensional subspaces $\{Y_j\}_{j=1}^\infty$ of a Banach-space Y is called* limit dense *in Y, if for any $\varepsilon > 0$ and any $v \in Y$, there exists an index j_ε, such that $\inf_{p_k \in Y_k} \|p_k - v\|_Y < \varepsilon$ for all $k > j_\varepsilon$.*

Lemma 3.3. *Let the spaces $\{Y_j\}_{j=1}^\infty$ be limit dense in $H(\Omega, \operatorname{div})$. Then*

$$\lim_{j \to \infty} \inf_{y_j \in Y_j, \beta > 0} M_\oplus^2(y_j, \beta) = \|\nabla u - \nabla u_h\|_A^2.$$

Proof. Recall that the $H(\Omega, \operatorname{div})$-norm $\|\cdot\|_{\operatorname{div}}$ is defined by $\|v\|_{\operatorname{div}}^2 = \|v\|^2 + \|\operatorname{div} v\|^2$. Let $\varepsilon > 0$ be arbitrarily small, but fixed. Let j_ε be the index such that, for all $k > j_\varepsilon$, there exists a $p_k \in Y_k$ with $\|A\nabla u - p_k\|_{\operatorname{div}} < \varepsilon$. Then,

(16) $\qquad \inf_{y_j \in Y_j, \beta > 0} M_\oplus^2(y_j, \beta) \leq M_\oplus^2(p_k, \varepsilon) = (1 + \varepsilon)\|A\nabla u_h - p_k\|_{\bar{A}}^2 + (1 + \frac{1}{\varepsilon})C_\Omega^2\|f + \operatorname{div} p_k\|^2.$

Since $\|Av\|_{\bar{A}} = \|v\|_A$, we can write

$$\begin{aligned} \|A\nabla u_h - p_k\|_{\bar{A}} &\leq \|A\nabla u_h - A\nabla u\|_{\bar{A}} + \|A\nabla u - p_k\|_{\bar{A}} \\ &= \|\nabla u_h - \nabla u\|_A + \|A\nabla u - p_k\|_{\bar{A}}. \end{aligned}$$

The norm $\|\cdot\|_{\bar{A}}$ is equivalent to the L^2-norm, so there exists a constant c_A, such that the second term in the right-hand side can be bounded by

$$\|A\nabla u - p_k\|_{\bar{A}} \leq c_A\|A\nabla u - p_k\| \leq c_A\|A\nabla u - p_k\|_{\operatorname{div}} \leq c_A\varepsilon.$$

Hence, we obtain the following estimate for the first term in (16):

(17) $$\|A\nabla u_h - p_k\|_{\bar{A}} \leq \|\nabla u - \nabla u_h\|_A + \mathcal{O}(\varepsilon).$$

Since $f = -\operatorname{div} A\nabla u$, we can bound the second term in (16) as follows:

(18) $$\|\operatorname{div} p_k + f\| = \|\operatorname{div} p_k - \operatorname{div} A\nabla u\| \leq \|p_k - A\nabla u\|_{\operatorname{div}} \leq \varepsilon.$$

With (17) and (18), we can rewrite (16) as

$$M_\oplus^2(p_k, \varepsilon) \leq (1 + \varepsilon)(\|\nabla u - \nabla u_h\|_A^2 + \mathcal{O}(\varepsilon)) + (1 + \tfrac{1}{\varepsilon})C_\Omega^2\varepsilon^2 = \|\nabla u - \nabla u_h\|_A^2 + \mathcal{O}(\varepsilon).$$

Hence, the bound $M_\oplus^2(p_k, \varepsilon) \to \|\nabla u - \nabla u_h\|_A^2$ as $\varepsilon \to 0$. □

3.2. **Steps Involved in Minimizing** $M_\oplus^2(y, \beta)$. As mentioned above, we need to find parameters y and β which minimize the majorant. To do this, we apply an interleaved iteration process in which we alternately fix one of the variables and minimize with respect to the other. This process, which we summarize in the following, has been described, e.g., in [25, 26].

Step 1: Minimization with respect to y: Assume that $\beta > 0$ is given and fixed, either by an initial guess or as a result of Step 2 below. We view the majorant $M_\oplus^2(y)$ as a quadratic function of y and calculate its Gateaux-derivative $M_\oplus^2(y)'$ with respect to y in direction $\tilde{y}$. Setting $M_\oplus^2(y)' = 0$, we obtain

$$(19) \qquad a_1 \int_\Omega A^{-1}y \cdot \tilde{y} \, dx + a_2 \int_\Omega \operatorname{div} y \, \operatorname{div} \tilde{y} \, dx \;=\; a_1 \int_\Omega \nabla u_h \cdot \tilde{y} \, dx - a_2 \int_\Omega f \, \operatorname{div} \tilde{y} \, dx,$$

where $a_1 = 1 + \beta$ and $a_2 = (1 + \tfrac{1}{\beta})C_\Omega^2$, as defined in (12). In order to solve (19), we choose a finite-dimensional subspace $Y_h \subset H(\Omega, \operatorname{div})$ and search for a solution $y_h \in Y_h$. Testing in all directions $\tilde{y} \in Y_h$ leads to a linear system of equations which we write as

$$(20) \qquad \underline{L}_h \underline{y}_h = \underline{r}_h.$$

Here, $\underline{L}_h$ and $\underline{r}_h$ are the matrix and the vector induced by the left hand side and the right hand side of equation (19), respectively. By solving (20), we obtain the coefficient vector $\underline{y}_h$ for the discrete function y_h minimizing $M_\oplus^2(y)$ in $Y_h \subset H(\Omega, \operatorname{div})$. Note that this process requires non-negligible cost as we need to assemble $\underline{L}_h$ and $\underline{r}_h$ and solve the system (20).

Step 2: Minimization with respect to β: Assume that y_h is given from Step 1. By direct calculation, we see that $M_\oplus^2(\beta)$ is minimized with respect to β by setting

$$(21) \qquad \beta \;=\; C_\Omega \sqrt{\frac{B_2}{B_1}},$$

where B_1 and B_2 are as defined in (12). Note that the evaluation of B_1 and B_2 (and thus β) requires only the evaluation of integrals, and thus involves negligible cost.

Steps 1 and 2 are repeated iteratively. We will refer to one loop of applying Step 1 and Step 2 as one *interleaved iteration*. Once we have computed minimizers y_h and β, the computation of the majorant $M_\oplus^2(y_h, \beta)$ is straight-forward as it requires only the evaluation of the integrals.

Note that the matrix $\underline{L}_h$ can be written as

$$(22) \qquad \underline{L}_h = a_1 \underline{L}_h^1 + a_2 \underline{L}_h^2,$$

where $\underline{L}_h^1$ and $\underline{L}_h^2$ correspond to the terms $\int_\Omega A^{-1}y \cdot \tilde{y} \, dx$ and $\int_\Omega \operatorname{div} y \, \operatorname{div} \tilde{y} \, dx$ in (19), respectively. Since the matrices $\underline{L}_h^1$ and $\underline{L}_h^2$ in (22) do not change in the interleaved iteration process, they need to be assembled only once. Analogously to (22), we can write $\underline{r}_h$ as

$$(23) \qquad \underline{r}_h = a_1 \underline{r}_h^1 - a_2 \underline{r}_h^2,$$

where $\underline{r}_h^1$ and $\underline{r}_h^2$ correspond to the terms $\int_\Omega \nabla u_h \cdot \tilde{y}_h \, dx$ and $\int_\Omega f \, \operatorname{div} \tilde{y} \, dx$ in (19), respectively. The terms $\underline{r}_h^1$ and $\underline{r}_h^2$ also need to be assembled only once since they also do not change in the interleaved iteration process. The full matrix $\underline{L}_h$ and vector $\underline{r}_h$, however, do change in each iteration, because of the change in β and y_h. Based on past numerical studies, see, e.g., [25, 26], and the results presented in Sections 4 and 5, it has been found that for linear problems, one or two such interleaved iterations are enough for obtaining a sufficiently accurate result.

To recapitulate, we summarize the steps for computing the majorant in Algorithm 1.

Remark 3.4. *Note that the space $H(\Omega, \operatorname{div})$, where the auxiliary quantity y is sought, is a global space, and for a general complicated problem, it is not immediately clear how to locally compute y without global effect. That being said, a local version of our estimator can be devised for specific problems and data (like equilibration of flux approach), however, that will restrict its generality, which is not very appealing to us. Therefore, in the remainder of the paper, we will focus on computing the majorant from the global minimization problem.*

Algorithm 1 Computation of the majorant $M_\oplus$

Require: u_h, f, C_Ω, Y_h
Ensure: $M_\oplus$
  $\beta :=$ initial guess
  Assemble and store $\underline{L}_h^1, \underline{L}_h^2, \underline{r}_h^1, \underline{r}_h^2$
  while convergence is not achieved or maximum number of interleaved iterations is not reached **do**
      $\underline{L}_h := (1+\beta)\underline{L}_h^1 + (1+\frac{1}{\beta})C_\Omega^2\underline{L}_h^2$
      $\underline{r}_h := (1+\beta)\underline{r}_h^1 - (1+\frac{1}{\beta})C_\Omega^2\underline{r}_h^2$
      Solve $\underline{L}_h\underline{y}_h = \underline{r}_h$ for $\underline{y}_h$
      $B_1 := \|A\nabla u_h - y_h\|_{\bar{A}}^2$
      $B_2 := \|\operatorname{div} y_h + f\|^2$
      $\beta := C_\Omega\sqrt{B_2/B_1}$
  end while
  $M_\oplus(y, \beta) := \sqrt{(1+\beta)B_1 + (1+\frac{1}{\beta})C_\Omega^2 B_2}$

3.3. **Quality Indicator and Local Error Indicator.** So far, we have defined the majorant and discussed how we minimize (numerically) the majorant over Y_h. Another important question, especially in the light of adaptive, local refinement, is whether a calculated majorant does correctly capture the error distribution. From the proof of Lemma 3.3, we recall the following observation:

$$(24) \qquad a_1 B_1 \to \|\nabla u - \nabla u_h\|_A^2 \text{ and } a_2 B_2 \to 0, \text{ as } y_h \in H(\Omega, \operatorname{div}) \to A\nabla u.$$

From this, we deduce the following quality indicator.

Proposition 3.5. *The distribution of the exact error is captured correctly, if*

$$(25) \qquad a_1 B_1 > C_\oplus \, a_2 B_2$$

with some constant $C_\oplus > 1$.

This criterion is easy to check, since the terms appearing in (25) are evaluated in the process of minimizing $M_\oplus^2(y, \beta)$. It was found from extensive numerical studies (see examples presented in Sections 4 and 5) that an accurate distribution of the error is obtained for $C_\oplus \geq 5$.

Remark 3.6. *For the choice of $C_\oplus \geq 5$, we have $a_2 B_2 < a_1 B_1/5$, and therefore, $\|\nabla u - \nabla u_h\|_A \leq \sqrt{1.2\, a_1 B_1}$. One can see from all the tables in Sections 4 and 5, that whenever this criterion is satisfied, we have $I_{eff} \leq 1.2$ (the ratio of $\sqrt{a_1 B_1}/\|\nabla u - \nabla u_h\|_A$ appears to be of the same magnitude as $\sqrt{1 + 1/C_\oplus}$. Note that this criterion does not require $a_2 B_2$ to be close to zero, but just less than $a_1 B_1/5$. Since these approximations (of the original problem and the auxiliary problem in $H(\Omega, \operatorname{div})$) are monotonically convergent, the approximation at any level will only improve at the next refinement level, and this is why the results get better for any further refinement. Clearly, all the terms are fully computable, and thus, usable in an algorithm.*

We define the local error indicator η_Q on a cell Q as the restriction of the first component of the majorant to the cell Q, i.e., by

$$(26) \qquad \eta_Q^2(y_h) = \int_Q (\nabla u_h - A^{-1}y_h)(A\nabla u_h - y_h) \, dx.$$

The factor $(1+\beta)$ is omitted, since this scalar factor is the same for all cells of the domain. As remarked in the observation (24), the first component will converge to the exact error, thus providing a good indicator for the error distribution. A more detailed discussion of this indicator can be found in [33, Sec. 3.6.4]. For refinement based on η_Q, we again use the criterion (10).

4. Efficiency and Computational Cost of the Proposed Estimator in the Isogeometric Context

We now discuss the efficiency and the computational cost of the proposed estimator based on the global minimization steps presented in Section 3.2. We again consider *Example 1* from Section 3.1. All the computations for this example and the examples presented in Section 5 are performed in MATLAB®,

and the linear systems (5) and (20) are solved using the in-built direct solver. One can, however, also use efficient iterative solvers, see, e.g., [19, 20] for (5). The right-hand-side f and the boundary conditions u_D are determined by the prescribed exact solution u.

We study the efficiency of the majorant based on *straight forward* computational procedure, as discussed in Section 4.1, and based on *cost-efficient* procedure, as discussed in Section 4.2, which coarsens the mesh and increases the polynomial degree simultaneously. This alternative cost-efficient procedure will then be used in Section 5 for further numerical examples. In all the numerical results of Example 1 in this Section, the initial guess for β is 0.01.

In the tables, we indicate the mesh-size by the number of interior knot spans of the knot vectors s and t, respectively. By this, we mean the number of knot spans without counting the vanishing knot spans at the beginning and the end of the open knot vectors. For example, if

$$
\begin{aligned}
s &= (0, 0, 0, 0.25, 0.5, 0.75, 1, 1, 1) \\
t &= (0, 0, 0, 0, 0.5, 0.5, 1, 1, 1, 1),
\end{aligned}
$$

then the mesh-size is 4×3, since the empty knot span $(0.5, 0.5)$ in t is also counted as an interior knot span.

The computed efficiency indices I_{eff} (see (14)) are presented in tables. In order to check the quality criterion discussed in Section 3.3, we present the values of $a_1 B_1$ and $a_2 B_2$ and see whether the inequality (25) is fulfilled or not. To indicate the quality of the error distribution captured by the majorant, we plot which cells are marked for refinement based on the exact local error and the criterion (10) (plotted in black), and compare this to the refinement marking based on the criterion (10) applied to the computed error estimate (plotted in magenta).

4.1. Straightforward Procedure.
Analogously to $\hat{V}_h$ and V_h, we choose a function space $\hat{Y}_h$ on the parameter domain and we define the function space Y_h by the push-forward

$$
Y_h = \hat{Y}_h \circ G^{-1}.
$$

Example 1 (Straightforward Procedure): For our first choice for $\hat{Y}_h$, we use the same mesh as for $\hat{V}_h$, and we choose

$$
\hat{Y}_h = \mathcal{S}_h^{p+1,p} \otimes \mathcal{S}_h^{p,p+1}, \tag{27}
$$

where $\mathcal{S}_h^{p,q}$ denotes the space of NURBS functions of degree p and C^{p-1}-continuity in the first coordinate, and degree q and C^{q-1}-continuity in the second coordinate. The parameter h indicates the characteristic cell size of the mesh for V_h.

We consider the same setting as presented in Example 1 in Section 3.1. In Table 1, we present the computed efficiency indices obtained with this choice of Y_h, which show that upper bound approaches 1 (representing exact error) as the mesh is refined. The dashed line in Table 1 indicates that the criterion (25) is fulfilled with $C_\oplus = 5$ (actually 4.94) starting from the mesh 64×64.

The cells marked by the error estimator are shown in Figure 4. When comparing these plots to those presented in Figure 1, we see that the error distribution is captured accurately starting from the mesh 32×32.

A careful look at the numerical tests, however, reveals that the computation of the error estimate with the straight forward approach is costlier (about 4.5 times the cost of assembling and solving the original problem). This is not surprising, since, when N_u denotes the number of DOFs of u_h, the number of DOFs of y_h, which is vector-valued, is asymptotically $2N_u$. This results in higher assembly time and the solution time for the linear system (where a direct solver is used).

In the next section, we discuss cost-efficient approaches for computing y_h.

4.2. Alternative Cost-Efficient Procedure.
Recall that the cost of Step 1 of the algorithm presented in Section 3.2 depends on the choice of $Y_h \subset H(\Omega, \text{div})$. As shown in Lemma 3.3, we can make the estimate as sharp as we desire by choosing a suitably large space Y_h. However, the larger Y_h is chosen, the more costly setting up and solving the system (20) becomes. Clearly, it is highly desirable to keep the cost for error estimation below the cost for solving the original problem.

As discussed above, choosing $\hat{Y}_h$ as in (27) does not result in a cost-efficient method. Apart from the fact that y_h is vector-valued while u_h is scalar, another aspect contributes to the high cost for the

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
8×8	3.43	2.62e+01	1.17e+02
16×16	1.92	6.07e-01	6.19e-01
32×32	1.41	2.29e-02	9.71e-03
64×64	1.20	1.15e-03	2.33e-04
128×128	1.10	6.51e-05	6.54e-06
256×256	1.05	3.87e-06	1.95e-07
512×512	1.03	2.36e-07	5.94e-09

TABLE 1. Efficiency index and components of the majorant in Example 1, Y_h as in (27).

(a) 16×16 (b) 32×32 (c) 64×64 (d) 128×128
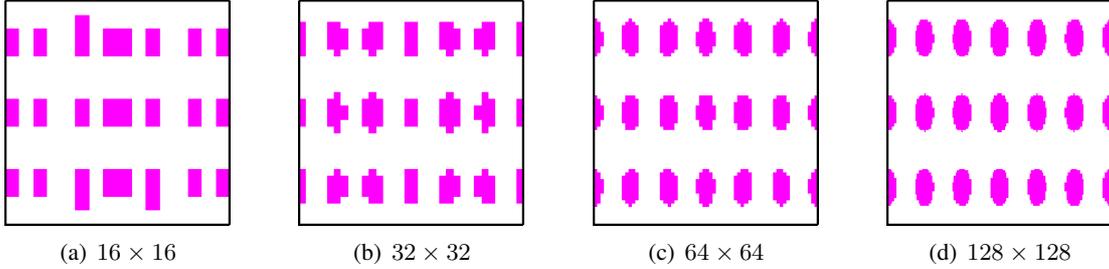
FIGURE 4. Cells marked by error estimator with $\psi = 20\%$ in Example 1, $\hat{Y}_h$ as in (27).

procedure presented in Section 4.1. Recall that, by choosing $\hat{Y}_h$ as in (27), we have

$$y_1 \in \mathcal{S}_h^{p+1,p},$$
$$y_2 \in \mathcal{S}_h^{p,p+1},$$

i.e., the components of y_h are in different spline spaces. Hence, we have to compute different basis functions for y_1 and y_2 (note that this can be a costly procedure for higher polynomial degrees). Furthermore, when assembling, for example, the matrix $\underline{L}_h^1$, we need to compute integrals over products of basis functions of the form

$$\int_\Omega R_i R_j \, dx.$$

With $\hat{Y}_h$ as in (27), the product $R_i R_j$ of basis functions of y_1 is different to the product of basis functions of y_2, hence, the integrals have to be evaluated independently for y_1 and y_2.

Example 1 (Case 1): In the light of these observations, and since $(C^{p-2})^d \subset H(\Omega, \text{div})$, $\forall p \geq 2$, we study the following alternative choice for $\hat{Y}_h$.

(28) $$\hat{Y}_h = \mathcal{S}_h^{p+1,p+1} \otimes \mathcal{S}_h^{p+1,p+1}.$$

We refer to this setting as *Case 1* in the remainder of the paper. With this choice, y_1 and y_2 are contained in the same spline spaces. Hence, the basis functions need to be computed only once, and any computed function values can be used for both components of y_h.

The computed efficiency indices are presented in Table 2, which show that we obtain even better (i.e., sharper) upper bounds for the exact error with $\hat{Y}_h$ as in (28) than with the choice (27). When we compare the plots of the cells marked by the error estimator in Figure 5 to the plots in Figure 1, we see that the error distribution is again captured accurately starting from the mesh 32×32. The dashed line in Table 2 indicates that the criterion (25) is fulfilled with $C_\oplus = 5$ starting from the mesh 64×64.

With this approach of Case 1, the total time needed for computing the majorant is reduced from a factor of about 4.5 to a factor of approximately 3 compared to the time for assembling and solving the original problem. Nevertheless, a factor of 3 in the timings is still not very appealing, and demands further reduction in the cost.

Remark 4.1. *Note that the use of equal degree polynomials for both the components of $\hat{Y}_h$ is only possible because of extra regularity readily available from NURBS basis functions. A counter-part is not possible in FEM case simply because the derivatives of FEM basis functions (with C^0 regularity) is*

11

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
8×8	2.77	8.08e+01	1.24e+01
16×16	1.71	5.75e-01	3.96e-01
32×32	1.32	2.14e-02	7.05e-03
64×64	1.16	1.11e-03	1.78e-04
128×128	1.08	6.39e-05	5.08e-06
256×256	1.04	3.83e-06	1.53e-07
512×512	1.02	2.35e-07	4.69e-09

TABLE 2. Efficiency index and components of the majorant in Example 1, Case 1.

(a) 16×16 (b) 32×32 (c) 64×64 (d) 128×128
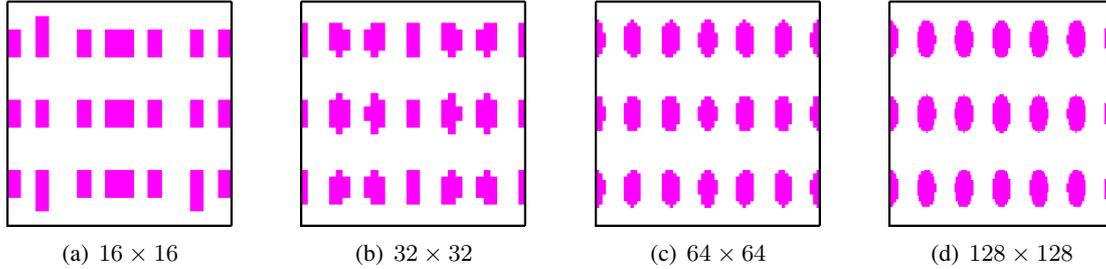
FIGURE 5. Cells marked by error estimator with $\psi = 20\%$ in Example 1, Case 1.

only in L^2, and hence, one can not avoid using proper subspaces of $H(\Omega, \mathrm{div})$, e.g., Raviart-Thomas space (with unequal degree polynomials in both the dimensions for both the components). It is further important to note from a close inspection of Tables 1 and 2 that the results from equal degree components of vector-valued quantity outperformed the results from unequal degree case.

In order to further reduce the computational cost, we reduce the number of DOFs of y_h by coarsening the mesh by a factor K in each dimension. The number of DOFs of y_h is thus reduced to $2N_u/K^2$ (asymptotically). The larger K is chosen, the greater the reduction of DOFs will be. At the same time, if the coarsening is done too aggressively, sharp features might not be detected properly on coarse meshes. We counter the reduction in accuracy due to mesh-coarsening by increasing the polynomial degree of y_h by some positive integer k, i.e., we choose

$$(29) \qquad \hat{Y}_h = \mathcal{S}_{Kh}^{p+k,p+k} \otimes \mathcal{S}_{Kh}^{p+k,p+k}.$$

Note that, if desired, one could also choose different factors K_1 and K_2 and different degree increases k_1 and k_2 for the first and second component, respectively.

Remark 4.2. *With the choices of $\hat{Y}_h$ as in (29), we take advantage of the following specific property of univariate NURBS basis functions. For C^{p-1} regularity, increasing the polynomial degree by k only adds a total of k additional basis functions. In other words, the global smoothness can be increased at the cost of only a few additional DOFs. Coarsening the mesh by a factor K, however, will also reduce the number of DOFs by the same factor K (asymptotically).*

Moreover, as we will see from the three cases of Example 1, asymptotically we get better efficiency indices with higher degree p and coarser meshes as compared to lower degree p and finer meshes. This phenomenon is similar to the p finite element discretization for problems with smooth solutions, where increasing the polynomial degree for a fixed mesh size h is much more advantageous than decreasing the mesh size h for a fixed (low) polynomial degree. Nevertheless, such a low cost construction for higher degree p is not possible in FEM discretizations.

Note that Case 1 discussed above fits into this framework, since Case 1 corresponds to the choice $K = k = 1$.

Example 1 (Case 2): For the next setting, we apply moderate mesh-coarsening by choosing

$$K = k = 2 \text{ (i.e., } \hat{Y}_h = \mathcal{S}_{2h}^{p+2,p+2} \otimes \mathcal{S}_{2h}^{p+2,p+2}).$$

This setting will be referred to as *Case 2* in the remainder of the paper. The computed efficiency indices along with the magnitudes of the terms a_1B_1 and a_2B_2 for Case 2 are presented in Table 3, and the marked cells are plotted in Figure 6. The dashed line indicates that criterion (25) is fulfilled with $C_\oplus = 5$, and that a good upper bound of the error is computed and the correct error distribution is captured on meshes starting from 64×64. On coarse meshes, however, the efficiency index is larger than in Case 1, which is due to the boundary effects. The observed timings show that this approach still costs roughly as much as solving the original problem.

mesh-size	I_{eff}	a_1B_1	a_2B_2
8×8	14.19	1.59e+03	8.53e+02
16×16	8.49	1.97e+01	4.32e+00
32×32	1.82	3.05e-02	2.41e-02
64×64	1.16	1.12e-03	1.76e-04
128×128	1.04	6.14e-05	2.24e-06
256×256	1.01	3.72e-06	3.32e-08
512×512	1.00	2.31e-07	5.13e-10

TABLE 3. Efficiency index and components of the majorant in Example 1, Case 2.

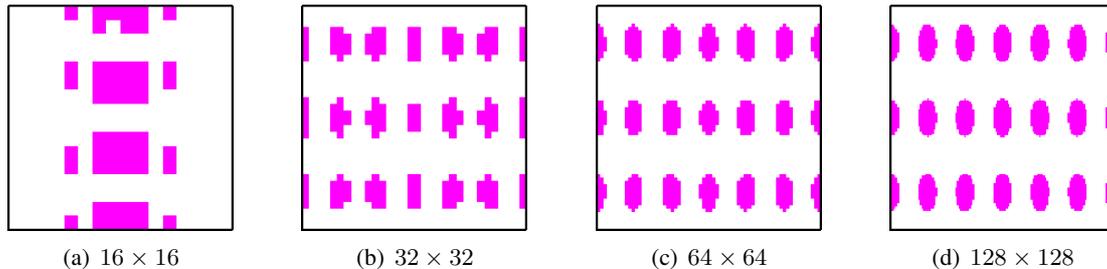(a) 16×16 (b) 32×32 (c) 64×64 (d) 128×128

FIGURE 6. Cells marked by error estimator with $\psi = 20\%$ in Example 1, Case 2.

Example 1 (Case 3): To further improve the timings, we coarsen the mesh more aggressively by a factor of 4 and, at the same time, increase the polynomial degree of y_h by 4, as compared to u_h, i.e.,

$$K = k = 4 \text{ (i.e., } \hat{Y}_h = \mathcal{S}_{4h}^{p+4,p+4} \otimes \mathcal{S}_{4h}^{p+4,p+4}).$$

We refer to this setting as *Case 3* in the remainder of the paper. This aggressive coarsening notably affects the efficiency index on coarse meshes, see Table 4. On fine meshes, however, the efficiency indices are close to 1 in all presented cases. The number of DOFs of y_h in Case 3 is only $N_u/8$ (asymptotically). The observed timings show that this setting results in a method which can be performed significantly faster (at almost half of the cost) than solving the original problem.

Remark 4.3. *In all Cases for Example 1, criterion (25) is fulfilled with $C_\oplus = 5$ on meshes of size 64×64 and finer. This is indicated by the dashed lines in Tables 1-4, and is clear from Figures 4-7. Therefore, Example 1 and the examples discussed in Section 5 show that $C_\oplus = 5$ is a good choice for checking criterion (25) numerically, even though this choice may be conservative in some cases.*

We now comment on the interleaved iterations. The results in the Tables 1-4 were obtained by applying only two interleaved iterations, as described in Section 3.2. As mentioned there, a sufficiently accurate result can be obtained already after the first such iteration. To illustrate this, we present the efficiency indices for Case 3 in Table 5, which were obtained after one, two, and four interleaved iterations, respectively. The efficiency index does vary notably on the coarser meshes, but since all of these values greatly overestimate the exact error, they do not correctly capture the error distribution. On meshes, where the criterion (25) is fulfilled with $C_\oplus = 5$, and thus the error distribution is correctly recovered, the differences due to more interleaved iterations are insignificant.

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
8×8	11.28	5.38e+02	1.01e+03
16×16	36.43	2.83e+02	1.60e+02
32×32	12.63	2.04e+00	5.81e-01
64×64	1.17	1.13e-03	1.88e-04
128×128	1.01	5.98e-05	3.79e-07
256×256	1.00	3.70e-06	1.24e-09
512×512	1.00	2.31e-07	5.32e-12

TABLE 4. Efficiency index and components of the majorant in Example 1, Case 3.

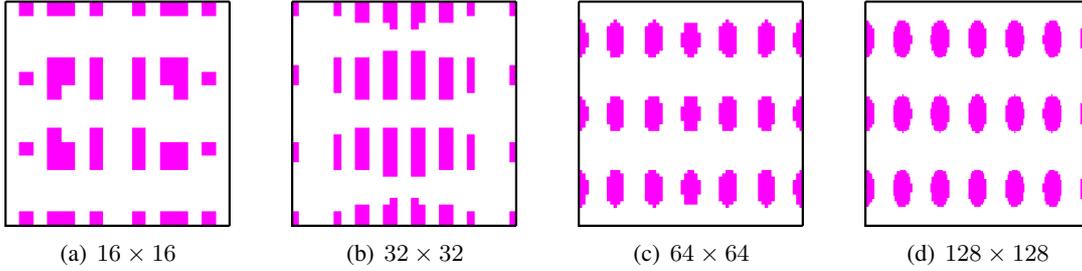| (a) 16×16	(b) 32×32	(c) 64×64	(d) 128×128

FIGURE 7. Cells marked by error estimator with $\psi = 20\%$ in Example 1, Case 3.

mesh-size	interleaved iterations		
	1	2	4
8×8	11.84	11.28	11.25
16×16	80.31	36.43	33.78
32×32	17.36	12.63	10.11
64×64	1.20	1.17	1.17
128×128	1.01	1.01	1.01
256×256	1.00	1.00	1.00
512×512	1.00	1.00	1.00

TABLE 5. Comparison of I_{eff} for different numbers of interleaved iterations, Example 1, Case 3.

Remark 4.4. *The observations discussed above illustrate that one has to balance the sharpness of the majorant on the one hand, and the required computational effort on the one hand. Note that in typical practical applications, the exact solution (and thus the sharpness of the majorant) is not known. Therefore, to address the balancing between sharpness and required computational effort, we propose the following strategy. If the mesh is coarse and the total computational cost for the error estimate is moderate, we apply no (or only moderate) coarsening. When the original mesh is fine (problem size being large), we coarsen the mesh aggressively, and thereby, profit from the fast computation of the estimate. While exercising this strategy it is important to enforce the criterion (25) with $C_\oplus \geq 5$.*

5. NUMERICAL EXAMPLES

In this section, we present further numerical examples which illustrate the potential of the proposed a posteriori error estimator. We will discuss the following three settings that were also discussed in Section 4. *Case 1:* $K = k = 1$, *Case 2:* $K = k = 2$, and *Case 3:* $K = k = 4$. As in Example 1, the initial guess for β is 0.01.

As discussed in Section 2.2, the parameter domain in all presented examples is the unit square $\hat{\Omega} = (0,1)^2$. The mesh-sizes in the two coordinate directions, which will be presented in the tables, are determined by the respective initial meshes, which in turn, are determined by the geometry mappings. The figures plotted in black represent the computations based on the exact error, and the figures plotted in magenta represent the computations based on the majorant. The data presented in the tables is as described in the beginning of Section 4.

We first consider an example with reduced regularity C^{p-m}, $m > 1$.

Example 2: Sinus Function on the Unit Square with $p = q = 4$. In this example, we consider the same exact solution and the same physical domain as in Example 1, i.e.,

$$u = \sin(6\pi x)\sin(3\pi y), \quad \Omega = (0,1)^2.$$

However, we now use B-splines of degree $p = q = 4$ to represent Ω, and we add a triple knot at the coordinates $x = 0.5$ and $y = 0.5$. The initial knot vectors are thus given by

$$s = t = (0,0,0,0,0,0.5,0.5,0.5,1,1,1,1,1),$$

and the geometry mapping is only C^1-continuous at the coordinate 0.5.

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
Case 1			
18×18	1.84	1.04e-03	9.00e-04
34×34	1.40	1.78e-06	7.23e-07
66×66	1.20	5.09e-09	1.00e-09
130×130	1.10	1.77e-11	1.74e-12
258×258	1.05	6.61e-14	3.25e-15
Case 2			
18×18	15.43	7.95e-02	5.75e-02
34×34	6.04	1.14e-05	3.53e-05
66×66	1.76	7.52e-09	5.69e-09
130×130	1.16	1.87e-11	3.01e-12
258×258	1.04	6.54e-14	2.49e-15
Case 3			
18×18	132.77	7.38e+00	2.76e+00
34×34	148.41	1.86e-02	9.53e-03
66×66	6.42	5.49e-08	1.21e-07
130×130	1.13	1.83e-11	2.39e-12
258×258	1.01	6.34e-14	3.78e-16

TABLE 6. Efficiency index and components of the majorant in Example 2.

(a) 18×18. (b) 34×34. (c) 66×66. (d) 130×130.
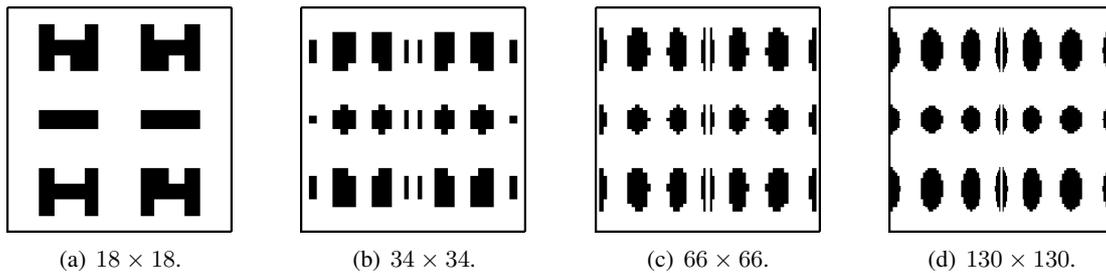
FIGURE 8. Cells marked by exact error with $\psi = 20\%$ in Example 2.

(a) 18×18, Case 1. (b) 34×34, Case 1. (c) 66×66, Case 2. (d) 130×130, Case 3.
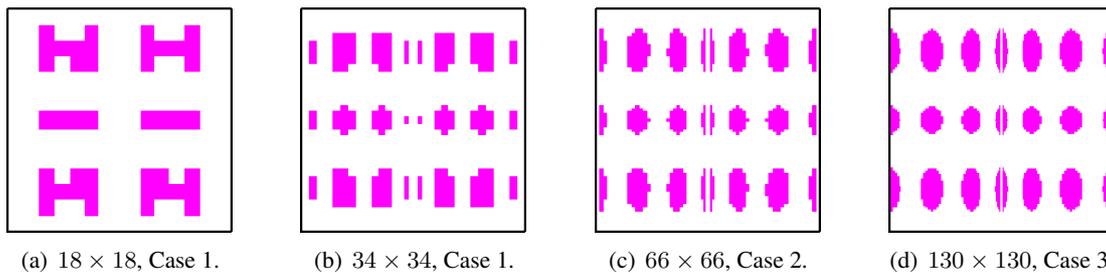
FIGURE 9. Cells marked by error estimator with $\psi = 20\%$ in Example 2.

The computed efficiency indices are presented in Table 6. The dashed lines, which correspond to criterion (25) being fulfilled with $C_\oplus = 5$, again show that more aggressive mesh-coarsening requires a finer initial mesh. By this criterion, we get a good quality of the estimate and the indicated error distribution starting from the mesh 66×66 in Case 1, and from 130×130 in Cases 2 and 3.

We present the cells marked for refinement by the exact error in Figure 8, and the cells marked by the error estimator in Figure 9. Figure 9(a) shows that the error distribution is already captured on the mesh 18×18 in Case 1. In Case 2, we obtain a good indication of the error distribution from the mesh 66×66, i.e., before criterion (25) with $C_\oplus = 5$ is fulfilled. Once the error distribution is captured correctly on a certain mesh, it is also captured on all finer meshes (as in Example 1). Hence, we do not show all plots for all meshes and cases, but only the first meshes, on which the error distribution is captured correctly. Also, as in Example 1, the estimate could be computed in approximately the same time as the original problem in Case 2, and in approximately half the time in Case 3.

In the next example, we consider a domain with a curved boundary (requiring a NURBS mapping for exact representation) and a problem whose solution has sharp peaks.
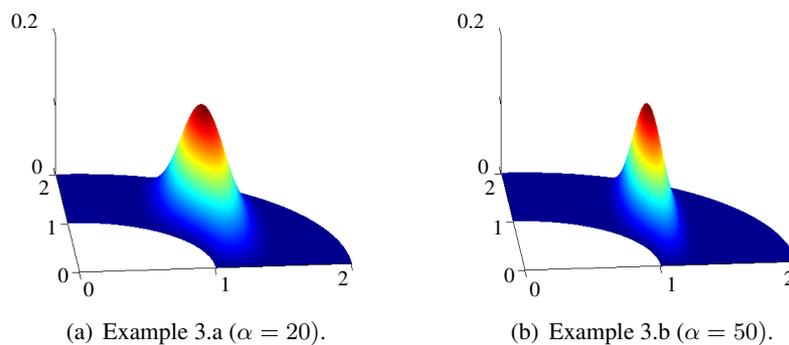

(a) Example 3.a ($\alpha = 20$).

(b) Example 3.b ($\alpha = 50$).

FIGURE 10. Exact solutions u on Ω, Example 3.

Example 3: Quarter Annulus. The domain Ω in this example represents a quarter annulus. In polar coordinates, Ω is defined by $(r, \phi) \in (1, 2) \times (0, \frac{\pi}{2})$. Note that the circular parts of the domain boundary are represented exactly by the NURBS geometry mapping of degree 2, i.e., we have $p = q = 2$. We set $A = I$, and we prescribe the exact solution

$$u = (r - 1)(r - 2)\phi(\phi - \tfrac{\pi}{2})e^{-\alpha(r\cos\phi - 1)^2}.$$

We test our method with two values of α, namely,

$$\text{Example 3.a:} \quad \alpha = 20, \qquad \text{Example 3.b:} \quad \alpha = 50.$$

In both examples, this function has zero Dirichlet boundary values and a peak at $x = 1$, the sharpness of which is determined by the value of α. The exact solutions are depicted in Figure 10.

In Tables 7 and 8, the efficiency index I_{eff} and the magnitudes of $a_1 B_1$ and $a_2 B_2$ are presented for both the cases of α. The dashed lines indicate the mesh-size after which criterion (25) with $C_\oplus = 5$ is fulfilled. The distribution of the marked cells is depicted in Figures 11 and 12. As before, we observe that the error distribution is represented correctly if the criterion (25) is fulfilled with $C_\oplus = 5$. When comparing Tables 7 and 8, as well as Figures 11 and 12, we notice that the more aggressive the mesh coarsening, and the sharper the peak, the more refinements are needed before criterion (25) is fulfilled and the error distribution is captured correctly.

In the next example, we test the proposed estimator in a basic adaptive refinement scheme.

Example 4: Adaptive Refinement. The exact solution for this example is given by

$$u = (x^2 - x)(y^2 - y)e^{-100|(x,y)-(0.8,0.05)|^2 - 100|(x,y)-(0.8,0.95)|^2}.$$

The computational domain is again the unit square $\Omega = (0, 1)^2$, and is represented by B-splines of degree $p = q = 2$. The function u, which is illustrated in Figure 13, has zero Dirichlet boundary values and has two peaks at the coordinates $(0.8, 0.05)$ and $(0.8, 0.95)$. In this example, we test a very basic adaptive

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
Case 1			
16×8	1.83	9.98e-04	3.59e-04
32×16	1.29	2.08e-05	6.51e-06
64×32	1.13	1.04e-06	1.44e-07
128×64	1.07	5.95e-08	4.00e-09
256×128	1.03	3.58e-09	1.20e-10
512×256	1.02	2.20e-10	3.67e-12
Case 2			
16×8	13.99	4.44e-02	3.51e-02
32×16	4.17	2.00e-04	8.43e-05
64×32	1.31	1.20e-06	3.66e-07
128×64	1.06	5.91e-08	3.36e-09
256×128	1.01	3.51e-09	4.60e-11
512×256	1.00	2.17e-10	6.96e-13
Case 3			
16×8	24.87	1.09e-01	1.42e-01
32×16	56.02	2.92e-02	2.22e-02
64×32	10.42	7.81e-05	2.16e-05
128×64	1.11	6.21e-08	6.61e-09
256×128	1.00	3.49e-09	1.02e-11
512×256	1.00	2.17e-10	3.27e-14

TABLE 7. Efficiency index and components of the majorant in Example 3.a ($\alpha = 20$).

(a) Exact,
mesh 32×16.

(b) Exact,
mesh 64×32.

(c) Exact,
mesh 128×64.

(d) Case 1,
mesh 32×16.

(e) Case 2,
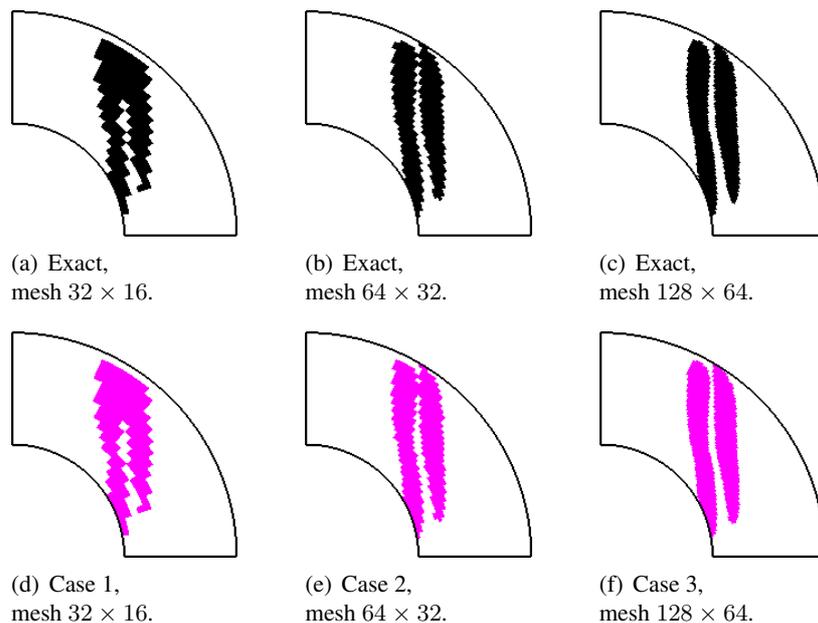mesh 64×32.

(f) Case 3,
mesh 128×64.

FIGURE 11. Marked cells with $\psi = 20\%$ in Example 3.a ($\alpha = 20$).

refinement procedure using tensor-product B-splines. The discussion of isogeometric local refinement schemes is out of the scope of this paper (see Section 1 for an overview on local refinement methods).

We apply adaptive refinement based on a marking with $\psi = 25\%$, starting on an initial mesh 16×16. On the first four steps, we apply Case 1, then Case 2 on the next three steps, and thereafter Case 3. The efficiency indices and the applied cases are shown in Table 9. In Figure 15, the meshes and the marked cells are shown for steps 4, 7, and 9. Clearly, the correct areas of the domain are identified and marked for refinement. Since the solution of the problem is sufficiently regular, the error plots in Figure 14 show that the adaptive refinement converges with the same rate as the uniform refinement, but with a better

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
Case 1			
16×8	3.02	2.94e-02	1.78e-02
32×16	1.92	3.57e-04	1.83e-04
64×32	1.34	9.15e-06	3.22e-06
128×64	1.16	4.67e-07	7.56e-08
256×128	1.08	2.67e-08	2.12e-09
512×256	1.04	1.60e-09	6.32e-11
Case 2			
16×8	13.84	3.45e-01	6.49e-01
32×16	16.76	2.58e-02	1.53e-02
64×32	3.16	4.10e-05	2.80e-05
128×64	1.25	5.04e-07	1.24e-07
256×128	1.05	2.61e-08	1.33e-09
512×256	1.01	1.56e-09	1.89e-11
Case 3			
16×8	17.20	4.24e-01	1.11e+00
32×16	76.95	3.24e-01	5.41e-01
64×32	83.72	3.02e-02	1.83e-02
128×64	4.19	4.64e-06	2.44e-06
256×128	1.04	2.59e-08	1.02e-09
512×256	1.00	1.55e-09	2.22e-12

TABLE 8. Efficiency index and components of the majorant in Example 3.b ($\alpha = 50$).

(a) Exact, mesh 64×32.

(b) Exact, mesh 128×64.

(c) Exact, mesh 256×128.

(d) Case 1, mesh 64×32.

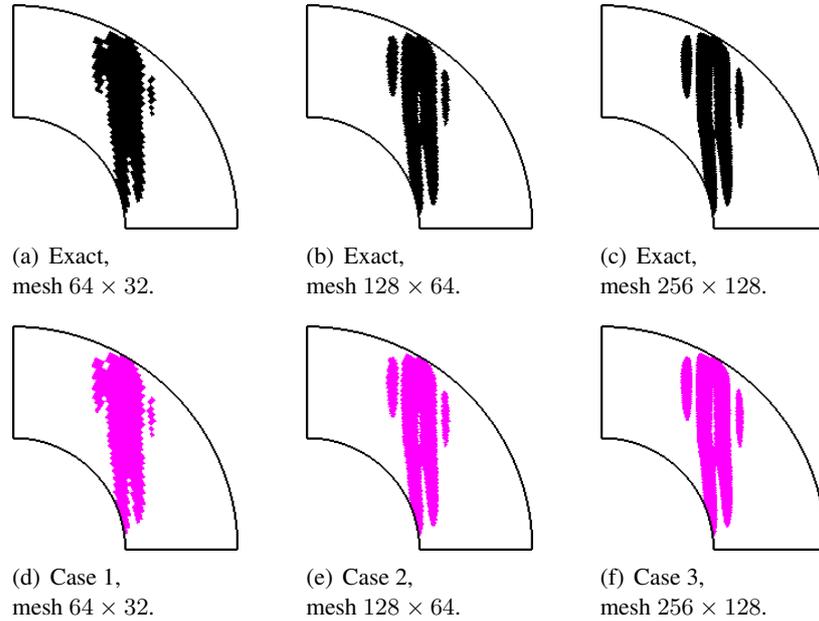(e) Case 2, mesh 128×64.

(f) Case 3, mesh 256×128.

FIGURE 12. Marked cells with $\psi = 20\%$ in Example 3.b ($\alpha = 50$).

constant. However, due to the tensor-product structure of the mesh, many superfluous DOFs are inserted outside of the marked areas, which worsens the rate of convergence for given total DOFs.

Next, we consider a classical example for a posteriori error estimation and adaptive refinement studies.

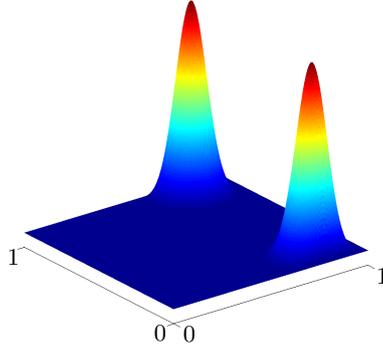**Example 5: L-shaped Domain.** We consider the Laplace equation

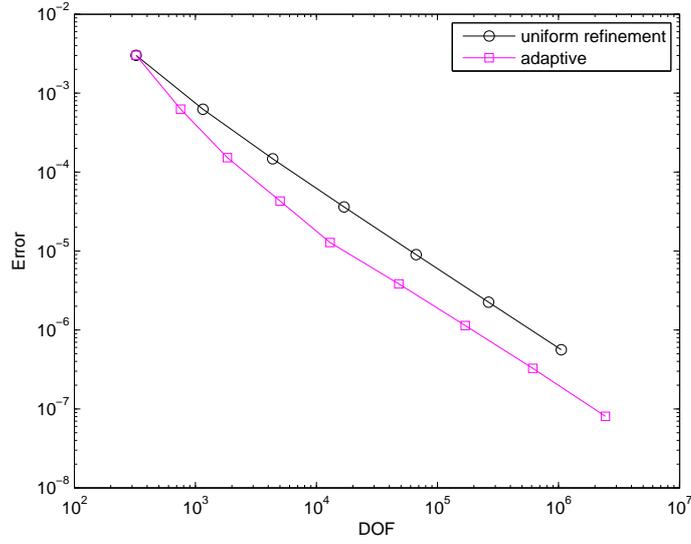$$(30) \qquad\qquad\qquad \Delta u = 0$$

FIGURE 13. Exact solution, Example 4.

FIGURE 14. Error convergence, Example 4.

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$	Case
16×16	3.77	9.39e-05	3.49e-05	1
25×26	2.06	8.62e-07	8.11e-07	1
38×44	1.69	4.30e-08	2.35e-08	1
64×74	1.47	2.79e-09	1.19e-09	1
92×136	2.82	8.19e-10	4.87e-10	2
184×256	1.30	2.05e-11	4.55e-12	2
341×492	1.11	1.45e-12	1.47e-13	2
652×934	1.84	2.55e-13	1.07e-13	3
1304×1868	1.09	7.40e-15	3.63e-16	3

TABLE 9. Efficiency index, components of the majorant and applied cases in Example 4.

with Dirichlet boundary conditions on the L-shaped domain $\Omega = (-1, 1)^2 \setminus [0, 1]^2$. In this example, we use a bilinear geometry mapping, i.e., $p = q = 1$. The function

$$u(r, \phi) = r^{\frac{2}{3}} \sin((2\phi - \pi)/3)$$

solves (30) and is used to prescribe Dirichlet boundary conditions. The solution has a singularity at the re-entrant corner at $(0, 0)$. We compare uniform refinement and adaptive refinement in the tensor-product

(a) Mesh 4.

(b) Cells marked by exact error on mesh 4.

(c) Cells marked by estimator on mesh 4.

(d) Mesh 7.

(e) Cells marked by exact error on mesh 7.

(f) Cells marked by estimator on mesh 7.

(g) Mesh 9.

(h) Cells marked by exact error on mesh 9.
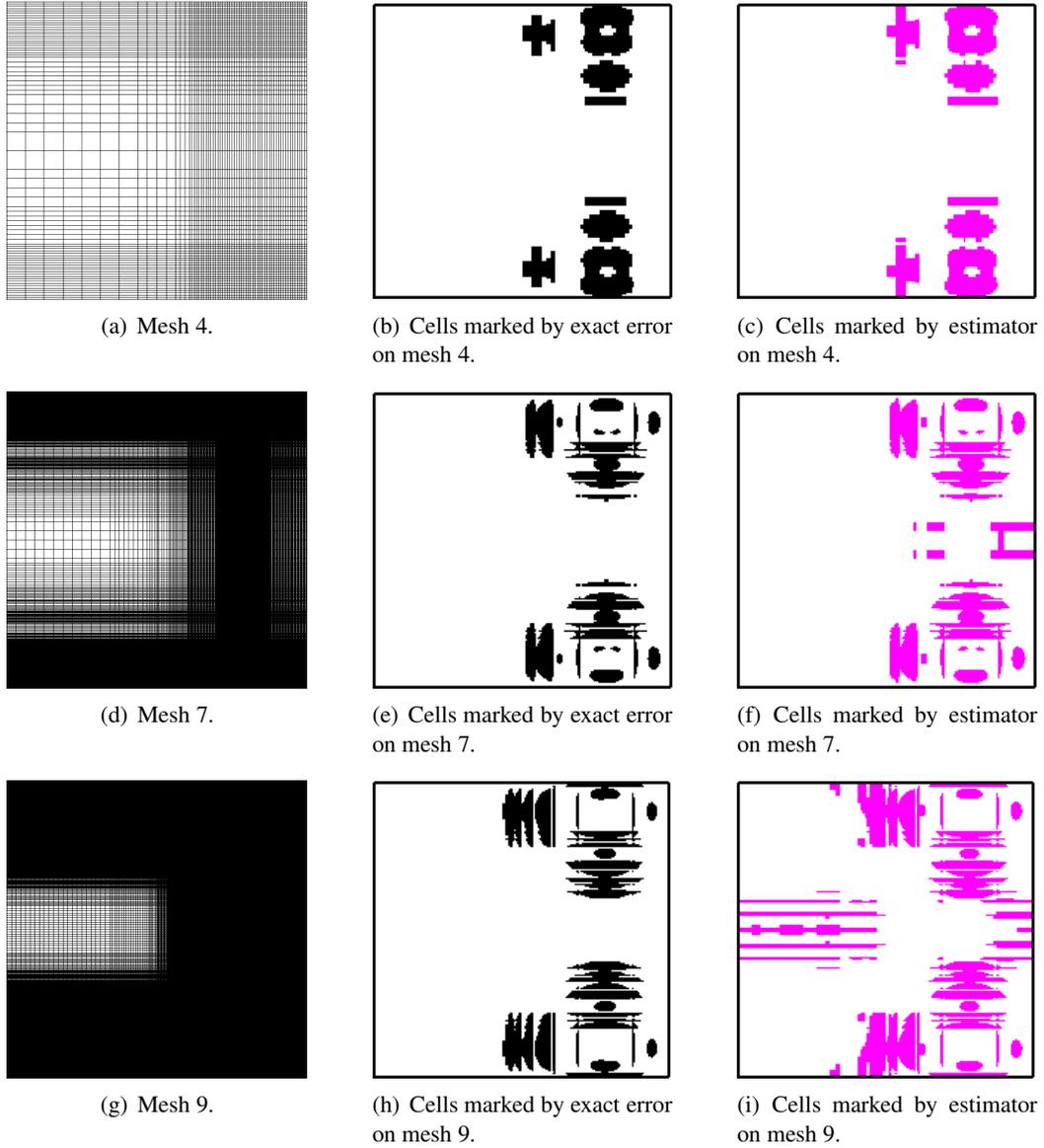
(i) Cells marked by estimator on mesh 9.

FIGURE 15. Meshes and marked cells in Example 4, $\psi = 25\%$.

setting. In this example we set $\psi = 10\%$, and to avoid the pollution near the singularity, we only use Case 1 in the majorant computations.

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
16×8	1.1785	5.67e-02	1.71e-03
32×16	1.1401	3.44e-02	8.98e-04
64×32	1.1116	2.09e-02	4.72e-04
128×64	1.0898	1.28e-02	2.49e-04
256×128	1.0729	7.87e-03	1.32e-04
512×256	1.0593	4.86e-03	7.01e-05
1024×512	1.0485	3.01e-03	3.73e-05

TABLE 10. Efficiency index and components of the majorant in Example 5, uniform refinement.

The magnitudes of the components $a_1 B_1$ and $a_2 B_2$, which are presented in Table 10 for uniform refinement, and in Table 11 for adaptive refinement, show that the criterion (25) with $C_{\oplus} = 5$ is fulfilled on all the considered meshes.

mesh-size	I_{eff}	$a_1 B_1$	$a_2 B_2$
16×8	1.1785	5.67e-02	1.71e-03
22×11	1.1839	2.68e-02	8.66e-04
30×16	1.1749	1.37e-02	4.32e-04
39×23	1.1622	7.22e-03	2.24e-04
55×37	1.1635	3.52e-03	1.10e-04
87×60	1.1634	1.75e-03	5.41e-05
133×101	1.1525	9.25e-04	2.69e-05

TABLE 11. Efficiency index and components of the majorant in Example 5, adaptive refinement with $\psi = 10\%$.

The error plots presented in Figure 16 show the expected faster convergence on the adaptively refined mesh, even though we are only using tensor-product splines. In Figure 17, meshes and marked cells are shown for steps 2 and 6, again indicating that the error indicator correctly identifies the corner singularity.
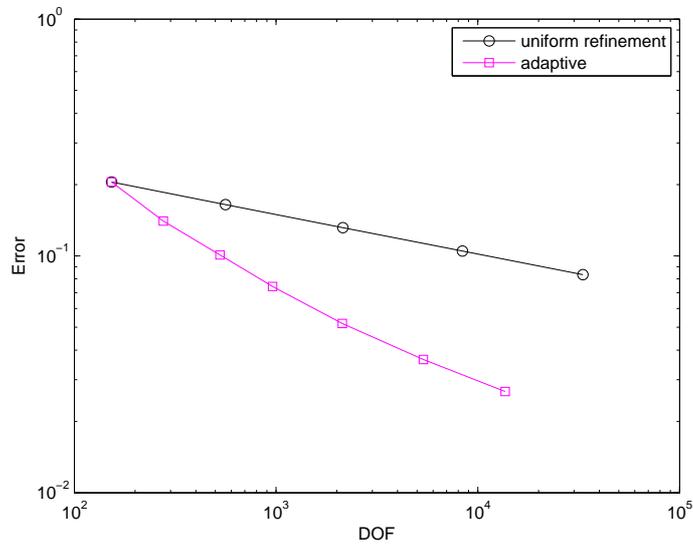

FIGURE 16. Error convergence, Example 5.

In our final example, we consider an advection dominated advection diffusion equation to see the performance of the estimator for sharp boundary layers.

Example 6: Advection Dominated Advection Diffusion Equation. We consider the advection diffusion equation with Dirichlet boundary conditions on the unit square $\Omega = (0, 1)^2$, with $p = q = 2$, i.e.,

$$-\kappa \Delta u + b \cdot \nabla u = 0 \qquad \text{in } \Omega,$$
$$u = u_D \qquad \text{on } \partial\Omega,$$

where

$$\kappa = 10^{-6}, \qquad b = (\cos \tfrac{\pi}{3}, \sin \tfrac{\pi}{3})^T, \qquad u_D = \begin{cases} 1, & \text{if } y = 0 \\ 0, & \text{else} \end{cases}.$$

We use the standard streamline upwind Petrov-Galerkin (SUPG) scheme for the stabilization. The stabilization parameter τ is set to $\tau(Q) = h_b(Q)/2|b|$, where $h_b(Q)$ is the diameter of the cell Q in direction of the flow b, and $|b|$ is the magnitude of the vector b. For advection diffusion problems, we have to adapt the majorant. Since the principle method is the same, we refer the reader to [33, Section 4.3.1] for a detailed discussion. In this special case, where $A = \kappa I$ with $\kappa \ll |b|$, and with constant velocity vector b, the majorant $M^2_{\oplus,\text{adv}}$ for the advection diffusion problem is given by

$$M^2_{\oplus,\text{adv}} = (1 + \beta)\|A\nabla u_h - y\|^2_{\bar{A}} + (1 + \tfrac{1}{\beta})C^2_\Omega\|\operatorname{div} y + f - b \cdot \nabla u_h\|^2.$$

(a) Mesh after 2 refinements.

(b) Cells marked by exact error on mesh 2.

(c) Cells marked by estimator on mesh 2.

(d) Mesh after 6 refinements.

(e) Cells marked by exact error on mesh 6.
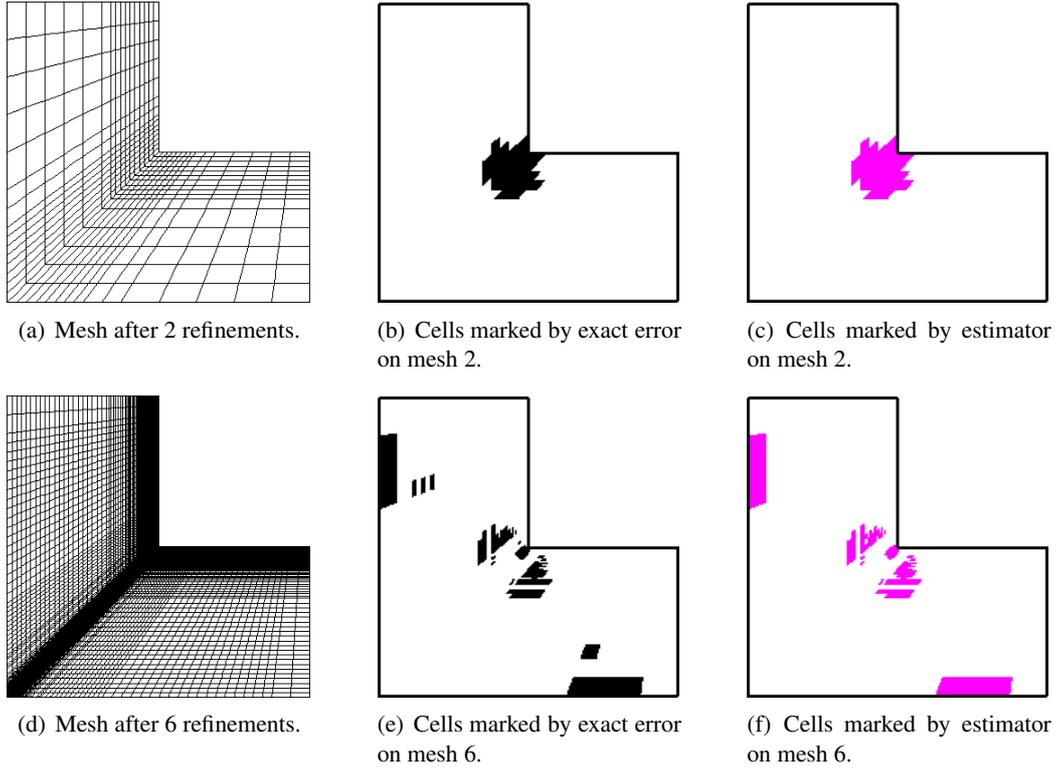
(f) Cells marked by estimator on mesh 6.

FIGURE 17. Meshes and marked cells in Example 5, $\psi = 10\%$.

The strong advection and the discontinuous boundary conditions result in sharp layers. In Figure 18(a), the expected positions of the layers are indicated by dashed lines.

mesh-size	$a_1 B_1$	$a_2 B_2$
Case 1		
16×16	1.98e-07	3.18e-10
64×64	6.45e-07	1.15e-09
256×256	2.28e-06	4.33e-09
Case 2		
16×16	1.83e-06	9.66e-10
64×64	6.50e-06	3.65e-09
256×256	1.86e-05	1.24e-08
Case 3		
16×16	3.24e-06	1.29e-09
64×64	2.07e-05	6.52e-09
256×256	6.86e-05	2.38e-08

TABLE 12. Comparison of terms $a_1 B_1$ and $a_2 B_2$ in Example 6.

The magnitudes of $a_1 B_1$ and $a_2 B_2$ presented in Table 12 indicate that the criterion (25) with $C_\oplus = 5$ is fulfilled on all the considered meshes. The distribution of the marked cells presented in Figures 18(b) and 18(c) provides the visual indication that the expected layers are accurately detected by the error estimator.

Note that, unlike the previous examples, for this example, assembling and solving the system for the estimator is faster than for the original problem not only in Case 3 (less than $1/2$ of the original cost), but also in Case 2 (about $2/3$ of the original cost). This is due to the SUPG stabilization which is costlier than computing the additional term $b \cdot \nabla u_h$ in the majorant $M_{\oplus,\text{adv}}^2$.
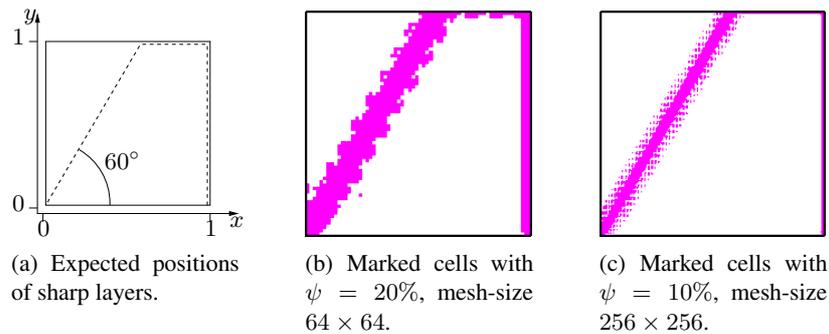
(a) Expected positions of sharp layers.

(b) Marked cells with $\psi = 20\%$, mesh-size 64×64.

(c) Marked cells with $\psi = 10\%$, mesh-size 256×256.

FIGURE 18. Expected layers and marked cells in Example 6, Case 3.

6. CONCLUSION

We have proposed a method for cost-efficient computation of guaranteed and sharp a posteriori error estimates in IGA. This method relies only on the use of NURBS basis functions, without the need for constructing complicated basis functions of $H(\Omega, \text{div})$. We have discussed different settings which allow the user to balance the sharpness of the bound and accurate error distribution on the one hand, and the required computational cost of the error estimator on the other hand (see Remark 4.4). For the presented settings, we have derived a quality criterion, which is easy to check numerically and which indicates whether the computed estimate is sharp or not (see Remark 4.3). Two properties of NURBS basis functions are exploited. Firstly, the basis functions are, in general, automatically in $H(\Omega, \text{div})$ due to their high smoothness. Without this property, we could not use NURBS of equal degree for both components as basis functions for the minimizing function y_h. Secondly, increasing the polynomial degree of NURBS basis functions adds only few DOFs. This fact is necessary for keeping the computational cost of the majorant as low as possible (see Remark 4.2). It is important to note that none of these properties are possible in FEM discretizations based on C^0 basis functions.

Apart from the topical interest of a posteriori error estimation and adaptivity, the presented method should also be of interest in parametrization of computational domain [40, 41]. Finally, in this paper, we have only considered tensor-product NURBS discretizations. While the extension of this method to locally refined isogeometric discretizations and also to three dimensions is, in theory, straightforward, the actual performance and efficiency of the error estimator on such methods and meshes is the subject of further studies.

ACKNOWLEDGEMENTS

REFERENCES

[1] M. Ainsworth and J.T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Wiley Interscience, 2000.

[2] R.E. Bank and R.K. Smith. A posteriori error estimates based on hierarchical bases. *SIAM J. Numer. Anal.*, 30(4):921–935, 1993.

[3] Y. Bazilevs, L. Beirão da Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli. Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Math. Models Methods Appl. Sci.*, 16(7):1031–1090, 2006.

[4] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg. Isogeometric analysis using T-splines. *Comput. Methods Appl. Mech. Engrg.*, 199(5-8):229–263, 2010.

[5] Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modelling for large eddy simulation of incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 197(1-4):173–201, 2007.

[6] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, and Y. Zhang. Isogeometric fluid-structure interaction: theory, algorithms, and computations. *Comput. Mech.*, 43:3–37, 2008.

[7] L. Beirão da Veiga, A. Buffa, J. Rivas, and G. Sangalli. Some estimates for h- p- k- refinement in isogeometric analysis. *Numer. Math.*, 118:271–305, 2011.

[8] A. Buffa, C. de Falco, and G. Sangalli. IsoGeometric Analysis: Stable elements for the 2D Stokes equation. *Int. J. Numer. Meth. Fluids*, 65(11-12):1407–1422, 2011.

[9] A. Buffa, J. Rivas, G. Sangalli, and R. Vázquez. Isogeometric discrete differential forms in three dimensions. *SIAM J. Numer. Anal.*, 49(2):818–844, 2011.

[10] A. Buffa, G. Sangalli, and R. Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Comput. Methods Appl. Mech. Engrg.*, 199(17-20):1143–1152, 2010.

[11] J. Cottrell, T.J.R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley, Chichester, 2009.

[12] J.A. Cottrell, T.J.R. Hughes, and A. Reali. Studies of refinement and continuity in isogeometric structural analysis. *Comput. Methods Appl. Mech. Engrg.*, 196:4160–4183, 2007.

[13] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, and Y. Feng. Polynomial splines over hierarchical T-meshes. *Graph. Models*, 70:76–86, 2008.

[14] T. Dokken, T. Lyche, and K.F. Pettersen. Polynomial splines over locally refined box-partitions *Comput. Aided Geom. Des.*, 30(3):331–356, 2013.

[15] M.R. Dörfel, B. Jüttler, and B. Simeon. Adaptive isogeometric analysis by local h-refinement with T-splines. *Comput. Methods Appl. Mech. Engrg.*, 199(5-8):264–275, 2010.

[16] W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33(3):1106–1124, 1996.

[17] T. Elguedj, Y. Bazilevs, V.M. Calo, and T.J.R. Hughes. $\bar{B}$ and $\bar{F}$ projection methods for nearly incompressible linear and nonlinear elasticity and plasticity using higher-order NURBS elements. *Comput. Methods Appl. Mech. Engrg.*, 197(33–40):2732–2762, 2008.

[18] J.A. Evans and T.J.R. Hughes. Isogeometric Divergence-conforming B-splines for the Darcy-Stokes-Brinkman equations. *Math. Mod. Meth. Appl. S.*, 23(04):671–741, 2013.

[19] K.P.S. Gahalaut, J.K. Kraus, and S.K. Tomar. Multigrid methods for isogeometric discretization. *Comput. Methods Appl. Mech. Engrg.*, 253(1):413–425, 2013.

[20] K.P.S. Gahalaut, S.K. Tomar and J.K. Kraus. Algebraic multilevel preconditioning in isogeometric analysis: Construction and numerical studies. *Comput. Methods Appl. Mech. Engrg.*, 266:40–56, 2013.

[21] C. Giannelli, B. Jüttler, and H. Speleers. THB-splines: The truncated basis for hierarchical splines. *Comput. Aided Geom. Des.*, 29(7):485–498, 2012.

[22] T.J.R. Hughes, J. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Engrg.*, 194(39-41):4135–4195, 2005.

[23] T.J.R. Hughes, A. Reali, and G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Comput. Methods Appl. Mech. Engrg.*, 199(5-8):301–313, 2010.

[24] K.A. Johannessen. An adaptive isogeometric finite element analysis. Master's thesis, Norwegian University of Science and Technology, 2009.

[25] J.K. Kraus and S.K. Tomar. Algebraic multilevel iteration method for lowest order Raviart-Thomas space and applications. *Int. J. Numer. Meth. Engrg.*, 86(10):1175–1196, 2011.

[26] R. Lazarov, S.I. Repin, and S.K. Tomar. Functional a posteriori error estimates for discontinuous Galerkin approximations of elliptic problems. *Numer. Methods Partial Differ. Equ.*, 25(4):952–971, 2009.

[27] X. Li, J. Zheng, T.W. Sederberg, T.J.R. Hughes, and M.A. Scott. On linear independence of T-spline blending functions. *Comput. Aided Geom. Des.*, 29(1):63–76, 2012.

[28] P.N. Nielsen, A.R. Gersborg, J. Gravesen, and N.L. Pedersen. Discretizations in isogeometric analysis of Navier-Stokes flow. *Comput. Methods Appl. Mech. Engrg.*, 200(45-46):3242–3253, 2011.

[29] L. Piegl and W. Tiller. *The NURBS book*. Monographs in Visual Communications. Springer Berlin Heidelberg, 2. edition, 1997.

[30] S.I. Repin. A posteriori error estimation for nonlinear variational problems by duality theory. *Journal of Mathematical Sciences*, 99:927–935, 2000. See also: *Zapiski Nauchnych Seminarov POMI*, 243:201–214, 1997.

[31] S.I. Repin. A posteriori error estimates for approximate solutions to variational problems with strongly convex functionals. *Journal of Mathematical Sciences*, 97:4311–4328, 1999.

[32] S.I. Repin. A posteriori error estimation for variational problems with uniformly convex functionals. *Math. Comput.*, 69(230):481–500, 2000.

[33] S.I. Repin. *A Posteriori Estimates for Partial Differential Equations*. Walter de Gruyter, Berlin, Germany, 2008.

[34] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sederberg, and T.J.R. Hughes. Isogeometric finite element data structures based on Bézier extraction of T-splines. *Int. J. Numer. Meth. Engng.*, 88:126–156, 2011.

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

[36] T.W. Sederberg, D.L. Cardon, G.T. Finnigan, N.S. North, J. Zheng, and T. Lyche. T-spline simplification and local refinement. *ACM Trans. Graph.*, 23(3):276–283, 2004.

[37] T. Takacs and B. Jüttler. Existence of stiffness matrix integrals for singularly parametrized domains in isogeometric analysis. *Comput. Methods Appl. Mech. Engrg.*, 200:3568–3582, 2011.

[38] A.-V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon. A hierarchical approach to adaptive local refinement in isogeometric analysis. *Comput. Methods Appl. Mech. Engrg.*, 200(49-52):3554–3567, 2011.

[39] P. Wang, J. Xu, J. Deng, and F. Chen. Adaptive isogeometric analysis using rational PHT-splines. *Computer-Aided Design*, 43(11):1438–1448, 2011.

[40] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo. Parameterization of computational domain in isogeometric analysis: Methods and comparison. *Comput. Methods Appl. Mech. Engrg.*, 200(23-24):2021–2031, 2011.

[41] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo. Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis. *Computer-Aided Design*, 45:812–821, 2013.

Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenbergerstrasse 69, 4040 Linz, Austria

E-mail address, Corresponding author: `satyendra.tomar@ricam.oeaw.ac.at`