Research Article

Steffen Weißer* and Thomas Wick

The Dual-Weighted Residual Estimator Realized on Polygonal Meshes

https://doi.org/10.1515/cmam-2017-0046
Received April 25, 2017; revised September 1, 2017; accepted October 3, 2017

Abstract: In this work, we realize goal-oriented error estimation using the dual-weighted residual method on general polygonal meshes. Such meshes are of current interest in various applications thanks to their great flexibility. Specifically the discrete problems are treated on BEM-based FEM. Our dual-weighted residual estimator is derived for two localization procedures. Firstly, a classical (strong) localization. Secondly, a weak form is adopted in which localization is achieved with the help of a partition-of-unity. The dual (i.e., adjoint) solution is obtained via a local higher-order approximation using a single element. Our algorithmic developments are substantiated with the help of several numerical tests.

Keywords: BEM-Based FEM, Polygonal Finite Elements, Goal-Oriented A Posteriori Error Estimation, Dual-Weighted Residual Estimator, Partition-of-Unity

MSC 2010: 65N30, 65N38, 65N50

1 Introduction

In this work, we consider goal-oriented local mesh adaptivity and a posteriori error estimation employing BEM-based FEM (BEM stands for boundary element method and FEM abbreviates finite element method) on polygonal meshes. In recent years, the attractions and the applications of such general meshes have been of emerging interest, because of the high flexibility in meshing complex structures and compound objects. Specifically, in mesh adaptivity the use of polygonal (2D) and polyhedral (3D) discretizations is very promising. When refining elements locally, there is no need for post-processing in order to maintain the mesh admissibility and there is no need to handle hanging nodes explicitly. Both scenarios are included naturally in the framework of polygonal and polyhedral meshes and the resultant flexibility.

So far, there is only a small number of publications exploiting the advantageous properties of general meshes for adaptivity. The BEM-based FEM has originally been proposed in [13] and has shown its flexibility and applicability on adaptively refined polygonal meshes in [42, 45, 46]. These articles concentrate on residual-based a-posteriori error estimates. Beside of the BEM-based FEM, we mention the Virtual Element Method (VEM), which has been analyzed for adaptive polygonal and polyhedral meshes in two recent publications [9, 14] treating a residual-based estimator and which has demonstrated its practicability in [18], for example. Other classes of methods applicable on general meshes are polygonal finite element methods, see [25] and the references therein, and hybrid high-order techniques [30]. On the other hand, there are also non-conforming discretization strategies on general meshes. A posteriori error estimates for the discontinu-

***Corresponding author: Steffen Weißer,** Applied Mathematics, Saarland University, Campus, 66041 Saarbrücken, Germany, e-mail: weisser@num.uni-sb.de. http://orcid.org/0000-0001-8507-9413
Thomas Wick, Centre de Mathématiques Appliquées, École Polytechnique, Université Paris-Saclay, 91128 Palaiseau, France, e-mail: thomas.wick@polytechnique.edu

ous Galerkin method are given in, e.g., [21, 24]. Furthermore, there is one publication for the Weak Galerkin Method [12], which is, however, limited to simplicial meshes. For the mimetic discretization technique there are also only few references, which are limited to low order methods, see the recent work [3].

In contrast to the previously mentioned error estimators studied for general meshes, the dual-weighted residual (DWR) method allows for estimating the error $u - u_h$ between the exact solution $u \in V$ (for a function space V) of the PDE and its Galerkin solution $u_h \in V_h \subset V$ in general (error) functionals $J : V \to \mathbb{R}$. These functionals can be norms but also more general expressions, like point-values, (local) averages or technical expressions like (in the case of fluid dynamics) lift- or drag-coefficients. Error estimators based on the DWR method always consist of residual evaluations, that are weighted by (local) adjoint sensitivity measures. These sensitivities are the solution to adjoint problems that measure the influence of the error functional J.

The DWR technique goes back to [6, 7] and is motivated by [17]. Important further developments in the early stages have been accomplished in [1, 2, 4, 8, 19, 28, 29, 31]. To date, goal-oriented DWR techniques have been applied successfully to elasto-plasticity [32] and contact problems [37, 40] as well as optimal control [5], fluid-structure interaction [33, 48], phase-field fracture [47], and isogeometric analysis [22] to name a few.

Most of these previous studies have in common that we either need the strong formulation [7] for the error localization or a special weak form with patched meshes [8]. In [34], a partition-of-unity (PU) localization technique based on the weak form has been introduced that is straightforward to employ and easy to implement. As in [8], partial integration back to the strong operator is not necessary. Therefore, neither face terms need to be evaluated nor the (costly) computation of second-order differential operators. Rather, solution information about neighboring cells is gathered by employing a PU (using a lowest-order discretization) leading to a nodal-based error indicator representation. In essence, such a weak localization of the error estimator works with the localized weak form of the PDE (which is available in any case), while replacing the test functions by adjoint weights. The key advantage of such weakly localized DWR techniques lies in its application to PDE systems and multiphysics problems in which many differential operators need to be evaluated and possibly additional face terms would arise on internal interfaces as for instance in fluid-structure interaction [33]. For a first application of the PU-DWR technique to phase-field fracture where two PDEs are coupled resulting in a nonlinear PDE system, we refer to [47]. We finally notice that a PU for strongly localized DWR error estimation has been suggested previously in [23].

Alongside with the realization of the DWR estimator on general meshes and using the two previously explained localization methods (namely classical and PU) for comparison reasons, we also propose a modification of the computation of the dual solution. Often, a brute force approach is to compute the dual solution with a globally higher-order discretization [7, Section 5], which is an expensive operation. Another option is to work with the same degree as for the primal solution. Here, the usual procedure is based on patched meshes (i.e., agglomeration of neighboring cells) [8, 34]. However, the usage of polygonal meshes allows us to work with the same degree as for the primal solution, but a locally higher-order approximation can be obtained without patched meshes.

Our previous descriptions can be summarized in short novelties as follows:

- Goal-oriented DWR error estimation for BEM-based FEM on polygonal meshes.
- Employing an element-based PU-based localization that allows to work with the weak equations inside the error estimator. These developments include comparisons with the classical strong localization.
- Local post-processing for the dual solution on single polygons which does not need the agglomeration (i.e., patched meshes) of neighboring elements.

For simplicity, we study in this work the Poisson equation as model problem. Although this equation has been employed in numerous papers, we believe that we need to establish our concepts first for this problem before we consider more complicated equations and practical applications in future work.

The outline of this paper is as follows: In Section 2, the model problem is given and the basics of the dual-weighted residual method are recapitulated. Next, in Section 3, the key ideas of BEM-based FEM are explained. Then in Section 4, our specific realization of the dual-weighted residual method on polygonal meshes is explained in great detail. Several numerical examples are provided in Section 5. These tests substantiate our developments. We finish our work with the conclusions in Section 6.

2 The Dual-Weighted Residual Method for Error Estimation

By $\Omega \subset \mathbb{R}^2$ we denote a bounded domain with polygonal boundary $\Gamma = \partial\Omega$. The boundary $\Gamma = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ is split into a Dirichlet and Neumann boundary Γ_D and Γ_N, respectively, and we assume that $|\Gamma_D| > 0$. Let $\omega \subset \Omega$ be a two-dimensional subdomain or a one-dimensional manifold; then we denote by $(\cdot\,,\cdot)_\omega$ the L^2-inner product and by $\|\cdot\|_\omega$ the corresponding L^2-norm over ω. For shorter notation we skip the index if $\omega = \Omega$. By $H^k(\omega)$ we denote the Sobolev space of Lebesgue functions with square integrable weak derivatives up to degree k and its norm is denoted by $\|\cdot\|_{H^k(\omega)}$. In particular, by $V := H^1_D(\Omega)$ we denote the space of $H^1(\Omega)$ functions with trace zero on the Dirichlet boundary Γ_D.

2.1 The Model Problem

The diffusion problem is defined as

$$\begin{cases} -\mathrm{div}(\nabla u) = f & \text{in } \Omega, \\ u = 0 & \text{on } \Gamma_D, \\ \partial_n u = g & \text{on } \Gamma_N, \end{cases}$$

where n denotes the outer unit normal vector to $\partial\Omega$. The corresponding weak problem on the continuous level reads: Find $u \in H^1_D(\Omega)$ such that

$$(\nabla u, \nabla\varphi) = (f, \varphi) + (g, \varphi)_{\Gamma_N} \quad \text{for all } \varphi \in V := H^1_D(\Omega). \tag{2.1}$$

The unknown solution $u \in V$ is computed numerically; namely in a finite-dimensional function space V_h. To this end, we derive the discrete weak formulation: Find $u_h \in V_h$ such that

$$(\nabla u_h, \nabla\varphi_h) = (f, \varphi_h) + (g, \varphi_h)_{\Gamma_N} \quad \text{for all } \varphi_h \in V_h. \tag{2.2}$$

Details on the concrete form of the discrete trial and test functions are provided in Section 3.

2.2 DWR for the Linear Diffusion Problems and Linear Goal Functionals

2.2.1 Derivation

We now describe the DWR approach for certain quantities of interest, i.e., goal functionals $J(u)$. In this work, we restrict our focus to linear functionals, but the general theory [7] does in particular hold for nonlinear situations as well. Such goal functionals can be of academic nature such as mean values or technical values such as drag or lift in fluid dynamics. For instance, mean values, line integrations of derivatives (related to stress values in elasticity) or point values are defined as

$$J(u) = \int_\Omega u\,dx, \qquad J(u) = \int_\Gamma \partial_n u\,dx, \qquad J(u) = u(x^*), \quad x^* \in \Omega.$$

The last functional is delicate from a theoretical point of view and is usually not defined on $H^1_D(\Omega)$. Detailed discussions can be found in [15, 16].

However, u is unknown and approximated by a discrete function u_h. Thus the key question is whether we can bound the error

$$J(u) - J(u_h).$$

To address this question, we assign a (linear) dual problem: Find $z \in V$ such that

$$a(v, z) = J(v) \quad \text{for all } v \in V. \tag{2.3}$$

Specifically, the dual bilinear form, corresponding to (2.1), is given by

$$a(v, z) = (\nabla v, \nabla z).$$

The boundary conditions are of homogeneous Dirichlet and Neumann type, where the Dirichlet conditions are built into V. The derivation of the dual (or better "adjoint") problem follows the Lagrangian formalism that is well known in optimization. The original motivation is provided in detail in [7].

Existence and uniqueness of this adjoint solution follow by standard arguments. As usual, the regularity of $z \in V$ depends on the regularity of the functional J. For example, for $J \in H^{-1}(\Omega)$, it holds $z \in H^1(\Omega)$. Given a more regular functional like the normalized L^2-error $J(\varphi) = \|e\|^{-1}(e, \varphi)$ with $e := u - u_h$, it holds $J \in L^2(\Omega)^*$, and consequently $z \in H^2(\Omega)$ on suitable domains (convex polygonal or smooth boundary with C^2-parametrization).

Inserting into (2.3) as special test function $v := u - u_h$ yields

$$a(u - u_h, z) = J(u - u_h),$$

and therefore we have a representation for the error in the goal functional.

In order to derive an error estimator, we use Galerkin orthogonality and insert the test function v_h:

$$a(u - u_h, z - v_h) = J(u - u_h).$$

Since v_h is an arbitrary discrete test function, we can, for instance, use a projection $v_h := i_h z$ (where $i_h : V \to V_h$) and obtain the following:

Proposition 2.1. *For the Galerkin approximation of the above bilinear form, we have the a posteriori error identity*

$$J(u - u_h) = a(u - u_h, z - i_h z). \tag{2.4}$$

Since z is an unknown itself, we cannot yet simply evaluate the error estimator because z is only analytically known in very special cases. Consequently, in order to obtain a computable error representation, z is approximated through a discrete function z_h^* that is (as the primal problem itself) obtained from solving a discretized version of (2.3). We then obtain:

Proposition 2.2. *Let z_h^* be the discrete dual function. For the Galerkin approximation of the above bilinear form, we have the a posteriori error representation*

$$J(u - u_h) \approx a(u - u_h, z_h^* - i_h z_h^*).$$

The straightforward choice of $z_h^* = z_h \in V_h$ as solution of

$$a(v_h, z_h) = J(v_h) \quad \text{for all } v_h \in V_h$$

is not applicable. Since $z_h - i_h z_h \in V_h$ and due to the Galerkin orthogonality, this choice yields

$$J(u - u_h) \approx a(u - u_h, z_h - i_h z_h) \equiv 0.$$

To overcome this point, we have different possibilities to get an approximation $z_h^* \notin V_h$ as explained in the next section.

In order to obtain an error estimator, the right-hand side of (2.4) is either estimated or approximated by some $\eta(u_h, z)$. The quality of this error estimator with respect to the true error is measured in terms of the effectivity index I_{eff} with

$$I_{\text{eff}}(u_h, z) = \left| \frac{\eta(u_h, z)}{J(u - u_h)} \right| \to 1 \quad (h \to 0).$$

In many applications, the asymptotic sharpness 1 cannot be achieved, but it should be emphasized that even overestimations of a factor 2 or 4 still yield a significant reduction of the computational cost in order to obtain a desired accuracy for the goal functional $J(u)$.

Remark 2.1. The above error estimator $\eta(u_h, z)$ is the basis for the derivation of a posteriori error estimates. In order to use this formulation for mesh refinement, we need to localize the error contributions on each element (or alternatively per degree of freedom). For localizing, the classical strategy and a variational localization using a partition-of-unity are explained in Section 4.3.1 and Section 4.3.2.

2.2.2 Approximation of the Dual Weights

For the evaluation of the error form (2.4), we must calculate approximations of the interpolation errors $z - i_h z$. This approximation is the critical part in the DWR framework that limits strict reliability [27]. A remedy is only given by spending sufficient effort on the estimation of these weights on fine meshes [7, 10] or an additional control of the approximation error in $z - i_h z$ (see [27]).

As just mentioned, it is well known that the discrete approximation of $z - i_h z$ must be finer than the trial space for the primal variable as the residual is orthogonal on V_h. Different methods have been suggested in the literature:

- Global higher-order approximation: higher FE solution on the same mesh or using the same FE degree for primal and dual, but the dual is computed on a finer mesh. Both variants are quite expensive [7].
- Local higher-order approximation using a patch-mesh structure [7], which is a cheaper alternative, but still needs an agglomeration of elements around the element of interest.

In this work, we propose a third variant:

- A local higher-order approximation using a single element. Here only the current element is used for the approximation of the dual problem while using the same polynomial degree for both the primal and the dual problem.

A detailed explanation of this third variant is provided in Section 4.2.2.

3 BEM-Based FEM Discretization

In this section we review some properties of polygonal meshes and the BEM-based FEM. Since we are aiming to solve the diffusion equation, we face the discretization of

$$H_D^1(\Omega) = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D\}.$$

The BEM-based FEM is a finite element method employing implicitly defined test and trial functions such that the method is applicable on polygonal meshes. These functions are treated by means of boundary integral formulations locally. Basically, what changes in comparison to FEM procedures is the computation of the local stiffness matrices on each element employing a local BEM solver. In the following, we review the approach described in [44]. For generalizations to 3D and more general equations see, e.g., [20, 36].

3.1 Polygonal Meshes

We decompose the domain Ω into a family of meshes $\mathcal{K}_h$ containing non-overlapping, polygonal elements $K \in \mathcal{K}_h$ such that

$$\overline{\Omega} = \bigcup_{K \in \mathcal{K}_h} \overline{K}.$$

The mesh is called regular if the following hold:

- all elements $K \in \mathcal{K}_h$ are star-shaped with respect to a circle of radius r_K and midpoint x_K, see Figure 1,
- the ratio of the element diameter h_K and r_K is uniformly bounded, i.e., $\frac{h_K}{r_K} < \sigma_{\mathcal{K}}$,
- the element diameter can be uniformly bounded by a constant $c_{\mathcal{K}}$ times the smallest length of its edges, i.e., $h_K < c_{\mathcal{K}} h_E$ for $E \in \mathcal{E}(K)$.

Here, $\mathcal{E}(K)$ denotes the set of edges of K. Analogously, we denote by $\mathcal{N}(K)$ the set of nodes of K. The midpoint x_K is placed such that r_K is maximized. If it is not unique, an arbitrary but fixed one is chosen. Furthermore, we denote by $\mathcal{N}_h$ and $\mathcal{E}_h$ the sets of all nodes and edges in the mesh. Each edge is located between two nodes $E = \overline{x_{E,1} x_{E,2}}$ that are the only nodes on the edge. We assume without loss of generality that $h_K < 1$, $K \in \mathcal{K}_h$. This is always achievable by scaling the domain.

In this context of regular polygonal meshes we especially allow non-convex elements and hanging nodes appear naturally as ordinary nodes on straight parts of the element boundary ∂K, see Figure 1. This property

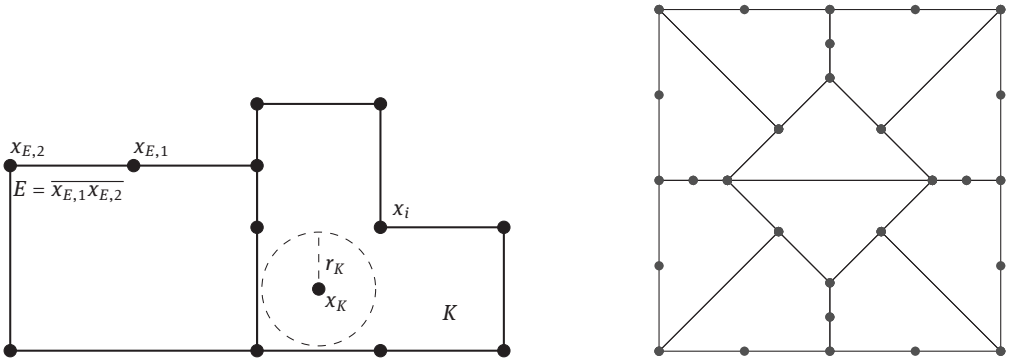

Figure 1: Two admissible elements (left) and polygonal mesh with naturally appearing hanging nodes (right), the nodes are marked with dots.

makes polygonal meshes very flexible and useful in adaptive meshing strategies. Local mesh refinement and coarsening can be easily obtained without additional local post-processing to keep the mesh admissible. Such operations are usually applied in high performance computing to achieve accurate results or in time-dependent problems to track singularities. On polygonal meshes these operations come almost for free.

3.2 Approximation Space

Obviously, it is not possible to use piecewise polynomials over the polygonal mesh to define a conforming approximation space $V_h^k \subset H_D^1(\Omega)$. In contrast, V_h^k is defined implicitly over each polygonal element. The index $k \geq 1$ gives the order of approximation. In the case of the diffusion equation, the BEM-based FEM makes use of

$$V_h^k = \{v \in H_D^1(\Omega) : \Delta v|_K \in \mathcal{P}^{k-2}(K),\ K \in \mathcal{K}_h \text{ and } v|_E \in \mathcal{P}^k(E),\ E \in \mathcal{E}_h\}.$$

Here, $\mathcal{P}^p(\cdot)$ denotes the space of polynomials of degree up to p over the elements and edges, respectively, where by definition $\mathcal{P}^{-1}(\cdot) = \{0\}$. In [44], it has been shown that the discrete variational formulation (2.2) together with this approximation space yields optimal rates of convergence for the error in the H^1- and L^2-norm on a sequence of regular, uniformly refined, polygonal meshes. More precisely, it holds

$$\|u - u_h\|_{H^\ell(\Omega)} \leq c\, h^{k+1-\ell}\, |u|_{H^{k+1}(\Omega)} \quad \text{for } u \in H^{k+1}(\Omega) \text{ and } \ell = 0, 1,$$

where $h = \max\{h_K : K \in \mathcal{K}_h\}$ and the constant c only depends on the mesh regularity. For $\ell = 0$ some additional regularity of the dual problem is assumed.

The basis set of V_h^k is defined element-by-element and glued continuously to obtain the globally conforming approximation space of $H^1(\Omega)$. In this regard, we consider the restriction of V_h^k onto a single element $K \in \mathcal{K}_h$ and denote it by $V_h^k(K)$. Let $\mathcal{P}_{\text{pw}}^k(\partial K) = \mathcal{P}_{\text{pw,d}}^k(\partial K) \cap C^0(\partial K)$ and

$$\mathcal{P}_{\text{pw,d}}^k(\partial K) = \{p \in L^2(\partial K) : p|_E \in \mathcal{P}^k(E), E \in \mathcal{E}(K)\}$$

be the spaces of piecewise polynomials over the boundary of the element K, which are continuous and discontinuous at the nodes, respectively. Each function $\psi \in V_h^k(K)$ fulfils locally a boundary value problem

$$\begin{cases} -\Delta\psi = p_K & \text{in } K, \\ \psi = p_{\partial K} & \text{on } \partial K, \end{cases} \tag{3.1}$$

with $p_K \in \mathcal{P}^{k-2}(K)$ and $p_{\partial K} \in \mathcal{P}_{\text{pw}}^k(\partial K)$. Since (3.1) has a unique solution, $\psi \in V_h^k(K)$ is given uniquely by the data p_K and $p_{\partial K}$. The space $V_h^k(K)$ is split into the direct sum $V_{h,H}^k(K) \oplus V_{h,B}^k(K)$ of two types of functions: the harmonic functions $V_{h,H}^k$ with $-\Delta\psi = 0$ in K and the element bubble functions $V_{h,B}^k$ with $\psi = 0$ on ∂K. In order to construct the basis functions of $V_h^k(K)$, we choose a standard basis set of $\mathcal{P}^{k-2}(K)$ as well as of $\mathcal{P}_{\text{pw}}^k(\partial K)$.

For each function in these sets an element bubble or a harmonic basis function ψ of $V_{h,B}^k(K)$ and $V_{h,H}^k(K)$ is obtained by (3.1) with the corresponding data. Finally, we obtain by gluing the harmonic basis function continuously the global representation

$$V_h^k = V_{h,H}^k \oplus V_{h,B}^k \quad \text{with} \quad V_{h,B}^k = \bigoplus_{K\in\mathcal{K}_h} V_{h,B}^k(K). \tag{3.2}$$

For a more detailed description of this procedure and a study of interpolation and approximation properties, we refer to [44]. In particular, we utilize the interpolation operator $\mathfrak{I}_h^k : C(\overline{\Omega}) \cap H^1(\Omega) \to V_h^k$ defined in [44] for the realization of the dual-weighted residual estimator later on. In the practical implementation the local problems (3.1) for the basis functions are realized by means of boundary element methods, see Section 3.4.

Since V_h^1 and especially the first order basis function are of particular interest in the remainder of this paper, we state an explicit definition. For each node $x_i \in \mathcal{N}_h$ in the mesh, we define the piecewise harmonic function ψ_{x_i} as unique solution of

$$\begin{cases} -\Delta\psi_{x_i} = 0 \quad \text{in } K \quad \text{for all } K \in \mathcal{K}_h, \\ \psi_{x_i}(x) = \begin{cases} 1 & \text{for } x = x_i, \\ 0 & \text{for } x \in \mathcal{N}_h \setminus \{x_i\}, \end{cases} \\ \psi_{x_i} \text{ is linear on each edge of the mesh.} \end{cases} \tag{3.3}$$

These functions fulfil the Dirac delta property in the nodes of the mesh.

3.3 Variational Formulation

In this subsection, we discuss an important consequence of the decomposition (3.2) of the approximation space V_h^k for the discrete weak formulation (2.2) of the primal problem. These considerations also apply to the discrete version of the dual problem (2.3). The approximation $u_h \in V_h^k$ is sought in the decomposed form $u_h = u_{h,H} + u_{h,B} \in V_{h,H}^k \oplus V_{h,B}^k$, where $u_{h,B} = \sum_{K\in\mathcal{K}_h} u_{h,B,K}$ is split further into its element contributions $u_{h,B,K} \in V_{h,B}^k(K)$. If we insert this representation into (2.2), the discrete weak formulation decouples. For $\psi \in V_{h,H}^k$ and $\varphi \in V_{h,B}^k$, the restriction $\psi|_K$ onto an element $K \in \mathcal{K}_h$ is harmonic and $\varphi \in H_0^1(K)$. Consequently, it is $(\nabla\psi, \nabla\varphi)_K = 0$ and we obtain $u_{h,H} \in V_{h,H}^k$ as unique solution of

$$(\nabla u_{h,H}, \nabla\varphi) = (f, \varphi) + (g, \varphi)_{\Gamma_N} \quad \text{for all } \varphi \in V_{h,H}^k. \tag{3.4}$$

Furthermore, it is easily seen that $u_{h,B} \in V_{h,B}^k$ is given uniquely by element-wise subproblems. On each $K \in \mathcal{K}_h$, $u_{h,B,K}$ fulfils

$$(\nabla u_{h,B,K}, \nabla\varphi)_K = (f, \varphi)_K \quad \text{for all } \varphi \in V_{h,B}^k(K). \tag{3.5}$$

Consequently, $u_{h,B}$ is determined by local projections of the right-hand side f into the discrete spaces $V_{h,B}^k(K)$, whereas $u_{h,H}$ is obtained by a global system of linear equations.

3.4 Boundary Element Method (BEM)

Let us consider the boundary value problem (3.1), and denote by $\gamma_0^K\psi \in H^{1/2}(\partial K)$ the usual trace of ψ on ∂K. Furthermore, let n_K be the outer unit normal vector of ∂K and denote by $\gamma_1^K\psi \in H^{-1/2}(\partial K)$ the Neumann trace of ψ on ∂K. This trace is given for sufficiently regular ψ as

$$\gamma_1^K\psi(x) = \lim_{K\ni\tilde{x}\to x} n_K(x) \cdot \nabla\psi(\tilde{x}) \quad \text{for } x \in \partial K.$$

Here, $H^{1/2}(\partial K)$ is the Sobolev–Slobodeckii space on the manifold ∂K and $H^{-1/2}(\partial K)$ its dual space. Thus, γ_1^K and ∂_{n_K} coincide for sufficiently regular functions. In the following, we assume that $p_K = 0$ in (3.1). Since $p_K \in \mathcal{P}^{k-2}(K)$, this assumption is always achievable by homogenization of the right-hand side with a polynomial, see, e.g., [21]. In this case, it is known that the solution of (3.1) can be expressed in the interior

of K with the help of the representation formula

$$\psi(x) = \int_{\partial K} U^*(x,y)\gamma_1^K\psi(y)\,ds_y - \int_{\partial K} \gamma_{1,y}^K U^*(x,y)\gamma_0^K\psi(y)\,ds_y \quad \text{for } x \in K, \tag{3.6}$$

where $\gamma_{1,y}^K$ denotes the Neumann trace with respect to y and $U^*(x,y) = -\frac{1}{2\pi}\ln(|x-y|)$, $x, y \in \mathbb{R}^2$, is the so-called fundamental solution of minus the Laplacian. A corresponding representation for $\nabla\psi$ can be obtained by differentiating (3.6). The Neumann trace, appearing in the first boundary integral, is unknown in general. Therefore, we approximate $\gamma_1^K\psi$ by $\widetilde{\gamma_1^K\psi} \in \mathcal{P}^k_{\text{pw,d}}(\partial K)$ using a Galerkin projection for a boundary integral equation connecting the Dirichlet and Neumann trace. The utilized formulation of the boundary element method reads:

$$\text{Find } \widetilde{\gamma_1^K\psi} \in \mathcal{P}^k_{\text{pw,d}}(\partial K) \text{ such that } (\mathbf{V}_K\widetilde{\gamma_1^K\psi}, q)_{\partial K} = \left(\left(\frac{1}{2}\mathbf{I} + \mathbf{K}_K\right)p_{\partial K}, q\right)_{\partial K} \quad \text{for all } q \in \mathcal{P}^{k-1}_{\text{pw,d}}(\partial K). \tag{3.7}$$

This formulation is uniquely solvable and involves boundary integral operators: the single layer potential $\mathbf{V}_K : H^{-1/2}(\partial K) \to H^{1/2}(\partial K)$ and the double layer potential $\mathbf{K}_K : H^{1/2}(\partial K) \to H^{1/2}(\partial K)$. Thus, the task to find an approximation of the Neumann trace is reduced to a one-dimensional problem on the boundary ∂K. The approximated trace can be used afterwards in the representation formula (3.6) in order to obtain an approximate evaluation of ψ. In many applications it is even sufficient to know the Dirichlet and the approximated Neumann data on the boundary of K. For more details on the concepts presented in this section we refer to the literature on boundary integral equations and boundary element methods, see, e.g., [26, 39].

3.5 Comments on the Implementation

In this subsection, we explain the main steps required for the implementation of the BEM-based FEM. This implementation constitutes the basis of our adaptive algorithm, which is developed in the next section. The variational formulation (3.4) and the local projections (3.5) are treated in the usual finite element framework:

- The global matrix of the system of linear equations in (3.4) is assembled element-wise by the successive adding up of local stiffness matrices.
- In order to avoid volume integrals involving the implicitly defined basis functions, we apply Green's first identity locally such that $(\nabla u_{h,H}, \nabla\varphi)_K = (\gamma_1^K u_{h,H}, \gamma_0^K\varphi)_{\partial K}$. The term $(\nabla u_{h,B,K}, \nabla\varphi)_K$ is similarly reduced to the boundary.
- The data terms (f, φ) and $(g, \varphi)_{\Gamma_N}$ are treated by means of numerical quadrature over each element and edge, respectively. For the first integral each element is split into triangles and a seven point quadrature formula is applied, where the evaluation of φ is realized with the help of the representation formula (3.6).
- The local stiffness matrix can be approximated by replacing the first term in $(\gamma_1^K u_{h,H}, \gamma_0^K\varphi)_{\partial K}$ by the BEM approximation $\widetilde{\gamma_1^K u_{h,H}}$. Alternatively, it turns out that the local stiffness matrices can be approximated very well with the symmetric discretization of the Steklov–Poincaré operator

  $$\mathbf{S}_K = \mathbf{D}_K + \left(\frac{1}{2}\mathbf{I}_K + \mathbf{K}_K'\right)\mathbf{V}_K^{-1}\left(\frac{1}{2}\mathbf{I}_K + \mathbf{K}_K\right),$$

  which is expressed by boundary integral operators. Here, $\mathbf{I}_K$ is the identity and $\mathbf{D}_K$ is the hypersingular integral operator.
- The global system of linear equations is solved iteratively with the help of the conjugate gradient (CG) method. For the tests in Section 5, we simply work with the CG method without any preconditioning.

In order to treat the locally implicitly defined trial functions and to compute the local stiffness matrices a boundary element method is applied:

- The application of the potentials $\mathbf{V}_K$ and $\mathbf{K}_K$ as well as the boundary integrals in (3.6) to piecewise polynomial data can be evaluated analytically.
- In the implementation, the double integrals over the boundary ∂K in (3.7), are realized by an advanced adaptive integration scheme, which is based on local subdivision of edges and shifting of Gaussian points according to the singularities of the integral kernels.

- The BEM is applied on the naturally given discretization of ∂K, i.e., no further discretization of the edges in the element boundary is applied. Therefore, and since the number of nodes per element is bounded, the resulting boundary element matrices are small.
- The resulting system of linear equations from (3.7) is solved with LAPACK.
- Boundary element matrices are set up once per element and used for all basis functions throughout the computations.
- The set up of boundary element matrices for different elements is independent and can be fully parallelized. Even the computation of the single entries of the matrices can be parallelized.
  More details on the specific implementation for the BEM-based FEM can be found in [35].

4 Realization of the Dual-Weighted Residual Method

In this section, we explain the realization of the dual-weighted residual method for goal-oriented error estimation using as discretization the previously reviewed BEM-based FEM. We first introduce special meshes and then recall various strategies to discretize the primal and dual problems. As novelty, we introduce an element-based post-processing of the dual solution. Finally, two error representations are recapitulated: using the classical method with strong forms of the differential operator, and secondly, using a PU for the variational form.

4.1 Special Meshes

The local post-processing of the dual solution might be done on coarsened meshes obtained by agglomerating polygonal elements in a classical way. However, on these general meshes we might alternatively use novel kinds of hierarchies. In the later described post-processing of the dual solution, we exploit this possibility. Therefore, we do not allow general polygonal meshes $\mathcal{K}_h$ as described in Section 3.1. There, we restrict ourselves to regular meshes $\mathcal{K}_h$ with polygonal elements having an even number of nodes such that every second node lies on an straight part of the boundary of the element. Furthermore, we assume that by removing these nodes from the mesh we obtain a coarsened polygonal mesh $\mathcal{K}_{2h}$ which is still regular. In Figure 2, we visualize such meshes $\mathcal{K}_h$ in the middle column and their corresponding coarsened meshes $\mathcal{K}_{2h}$ in the left column. Using these meshes, we define the approximation spaces V_h^k and V_{2h}^k, respectively.

The condition on the node count for $\mathcal{K}_h$ is not a real restriction. We can always introduce some additional nodes in the mesh to ensure the requirements. This is also done when we refine some given meshes. The middle column of Figure 2 shows a sequence of uniform refined meshes which are used in later numerical experiments in Section 5. In the refinement procedure each element in the mesh $\mathcal{K}_h$ is split as described in [42]. This yields a mesh which does not fulfil the requirement on the node count for each element in general, see Figure 2 right. However, we can ensure the required structure of the mesh by introducing some additional nodes. This can be observed by comparing the refined, but inappropriate mesh, in the right column of Figure 2 with the next mesh in the sequence depicted in the middle column.

We further point out that the refinement algorithm given in [42] does not preserve the mesh regularity discussed in Section 3.1 in general. In the case of convex elements, the applied bisection yields two new convex elements and their aspect ratio of h_K and r_K stays bounded. The edge lengths, however, might degenerate with respect to the element diameter. This can be avoided by an additional adjustment of the splitting.

4.2 Approximation of the Primal and Dual Solution

The primal and dual problems are treated by means of BEM-based FEM as described in Section 3. For this reason the domain is meshed into polygonal elements fulfilling the regularity assumption and, in most experiments, the requirement on the node count as described above. The primal variable u is approximated by $u_h \in V_h^k$, which is given by the decoupled weak formulation (3.4) and (3.5).

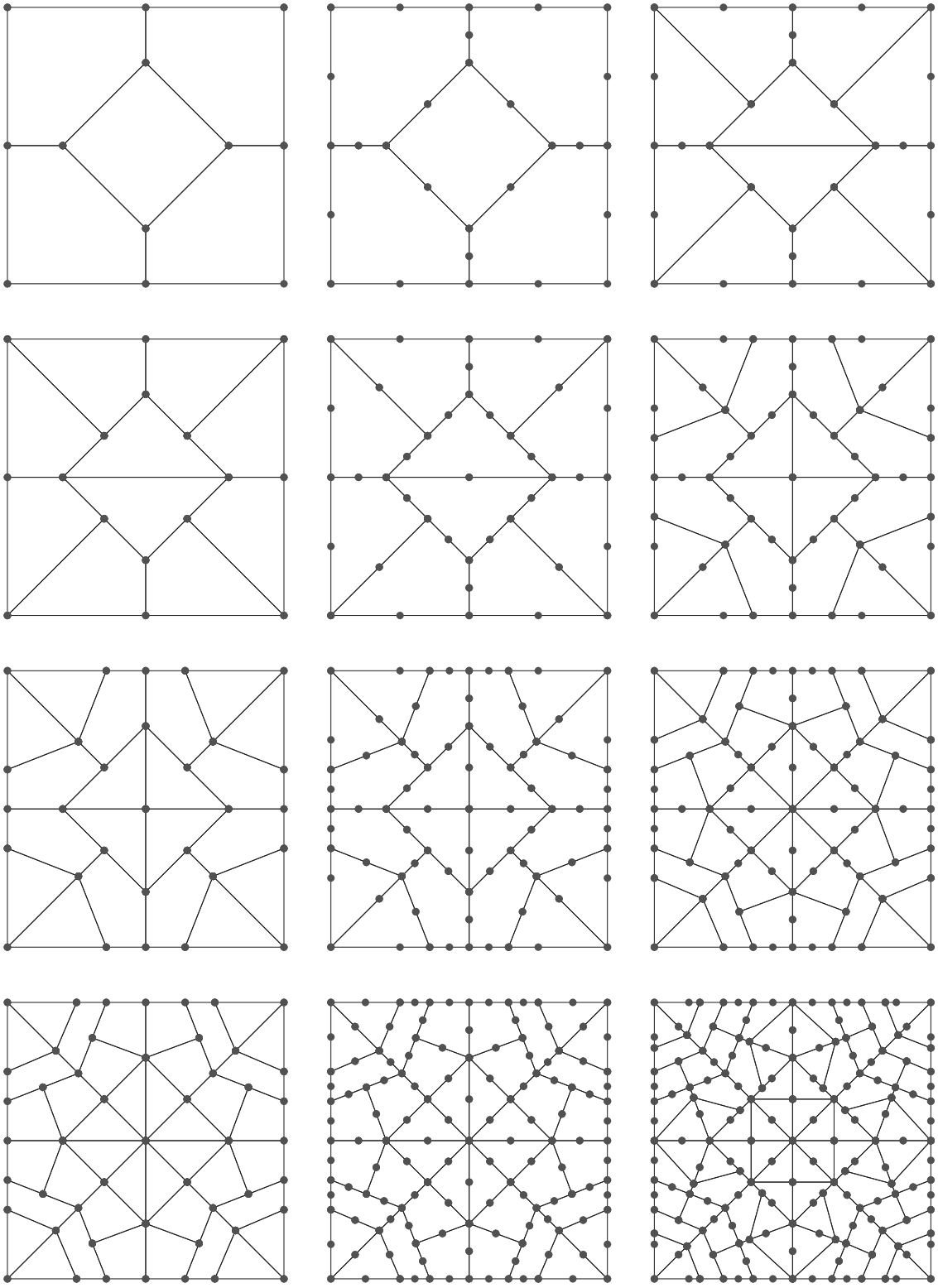

Figure 2: Each row corresponds to the mesh in one FEM simulation, the middle column corresponds to the actual mesh $\mathcal{K}_h$, the left column shows the mesh $\mathcal{K}_{2h}$ after coarsening and the right column shows the mesh after refinement before the nodes are added to ensure the condition on the node count.

In this paper we focus on two strategies for the approximation of the dual solution z. Either we use globally a higher-order for the approximation, which is, however, practically expensive, or we apply a local post-processing of $z_h \in V_h^k$. The local post-processing is especially attractive for the approximation space of the BEM-based FEM, since there is no need for local agglomerations of elements as we see in the next subsections. The reason to consider both strategies is a more reliable verification of our algorithms and programming code rather than just testing one single technique.

4.2.1 Dual Solution with Globally Higher-Order Discretization

A brute force strategy to obtain an approximation of the dual solution, which is suited for error estimation, is to solve the discrete variational formulation with higher accuracy. To track the approximation order, we write $u_h = u_h^{(k)} \in V_h^k$ for the approximation of the primal solution. The dual solution can be approximated by $z_h^{(k+1)} \in V_h^{k+1}$ on the same mesh. The choice $z_h^* = z_h^{(k+1)}$ is applicable for the error representation, cf. (2.4). Here, we do not need the restriction on the node count for the mesh $\mathcal{K}_h$. As we already mentioned, this strategy is computationally expensive in practical applications. However, it serves as a good starting point to verify the performance of the dual-weighted residual method on polygonal meshes.

4.2.2 Dual Solution Exploiting Local Post-Processing

A more convenient and efficient strategy is to approximate the dual solution $z_h = z_h^{(k)} \in V_h^k$ on the same mesh with the same approximation order as the primal solution. Afterwards z_h^* is chosen as a post-processed version of z_h on a coarsened mesh with higher approximation order. This strategy is well discussed in the literature for simplicial meshes, see [34] and the references therein (but has in fact already been introduced in the early studies [7]). The key point is, how the meshes, and especially the coarse meshes, are chosen. Since polygonal meshes are very flexible and inexpensive for coarsening and refining, they are well suited for this task. It is possible to just agglomerate two or more neighboring elements to construct a coarsened mesh and to proceed in a classical way for the local post-processing.

In the following we describe, however, a slightly different strategy that does not need the agglomeration of elements and is applicable on single elements. We use the meshes $\mathcal{K}_h$ and $\mathcal{K}_{2h}$ discussed in Section 4.1 fulfilling the requirement on the node count. The approach relies on two key ingredients: the hierarchy of the discretization of the element boundaries ∂K in these two meshes and the decoupling of the dual problem analogously to (3.4) and (3.5) for the primal problem.

Let $z_h \in V_h^k$ be the approximation of the dual problem over the mesh $\mathcal{K}_h$. We construct $z_h^* \in V_{2h}^{k+1}$ as locally post-processed function over the mesh $\mathcal{K}_{2h}$. We write the mapping $z_h = z_h^{(k)} \mapsto z_h^* \in V_{2h}^{k+1}$ also in operator notation with $\mathfrak{P}_{2h}^{k+1} : V_h^k \to V_{2h}^{k+1}$ such that $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$. It is sufficient to define the post-processing on a single element $K \in \mathcal{K}_h$, since it directly generalizes to the entire mesh. The element $K = K_h \in \mathcal{K}_h$ has a corresponding element $K_{2h} \in \mathcal{K}_{2h}$, which is obtained by skipping every second node on the boundary ∂K_h. Thus, the shapes of these elements coincide and they only differ in the number of nodes on the boundary. Consequently, ∂K_h can be interpreted as a refinement of ∂K_{2h}, or in other words, ∂K_h and ∂K_{2h} are one-dimensional patched meshes of the element boundary. Therefore, it is $\mathcal{P}_{\mathrm{pw}}^k(\partial K_{2h}) \subset \mathcal{P}_{\mathrm{pw}}^k(\partial K_h)$. In terms of the approximation space we have

$$V_{2h}^k(K_h) := V_h^k(K_{2h}) \subset V_h^k(K_h).$$

Since it is clear from the approximation space which element is meant, we skip the index h and $2h$ again.

Suppose that we would approximate the dual problem globally in V_{2h}^{k+1}. Then the weak formulation decouples into a global system of linear equations in order to compute the expansion coefficients of the harmonic basis functions and into a projection of the error functional into the space of element bubble functions. We similarly proceed with the post-processing. Exploiting the hierarchy of the boundary, we construct

$$z_h^* = z_{h,H}^* + z_{h,B}^* \in V_{2h}^{k+1}(K) = V_{2h,H}^{k+1}(K) \oplus V_{2h,B}^{k+1}(K)$$

from $z_h = z_{h,H} + z_{h,B} \in V_h^k(K)$ in the following way: We set

$$z_{h,H}^* \in V_{2h,H}^{k+1}(K) \text{ as interpolation of } z_{h,H} \in V_{h,H}^k(K) \tag{4.1}$$

and

$$z_{h,B}^* \in V_{2h,B}^{k+1}(K) \text{ as solution of } (\nabla z_{h,B}^*, \nabla\varphi)_K = J(\varphi) \text{ for all } \varphi \in V_{2h,B}^{k+1}(K). \tag{4.2}$$

The interpolation in (4.1) is equivalent to an interpolation of a function in $\mathcal{P}_{\text{pw}}^k(\partial K_h)$ by a function in $\mathcal{P}_{\text{pw}}^{k+1}(\partial K_{2h})$. Thus, a standard point-wise interpolation procedure is applied. The definition of $z_{h,B}^*$ is exactly the projection of the error functional into the space of element bubble functions. Both operations are local over a single element and are thus suited for a computationally inexpensive post-processing.

Remark 4.1. The first idea might be to use the interpolation operator $\mathfrak{I}_h^k$ mentioned in Section 3.2 and to set $z_h^* = \mathfrak{I}_{2h}^{k+1} z_h^{(k)}$. But, this strategy does not work. The interpolation affecting the harmonic basis functions yields the same results as described above. But, the transition from the lower order element bubble functions to the higher-order ones is not well suited. Since there is no agglomeration of elements and the process is kept on a single element, there is no additional information in the interpolation using higher-order element bubble functions. This is reflected by the fact that $V_{h,B}^k(K) = V_{2h,B}^k(K)$. The choice (4.2) overcomes this deficit and includes the required information for the element bubble functions by exploiting the dual problem.

4.3 The Localized Error Estimators

In this subsection, we discuss the localization of the error representation derived in Section 2 on polygonal meshes. The representation involves the adjoint sensitivity measure $z - i_h z$ with $i_h : V \to V_h^k$. Since the dual solution is not known in general, it is approximated (as usually done in the literature, e.g., [7]) in the numerical tests as discussed in Section 4.2. In the realization, we replace z in the error estimates by z_h^*. The operator i_h is realized in the following with the help of the interpolation operator $\mathfrak{I}_h^k$, which is mentioned in Section 3.2 and defined in [44].

4.3.1 The Classical Way of Localization

In the classical way, the error identity in Proposition 2.1 is realized by using the concrete problem, followed by integration by parts on every mesh element $K \in \mathcal{K}_h$, yielding

$$J(u - u_h) = \sum_{K\in\mathcal{K}_h} \left\{ (f + \Delta u_h, z - i_h z)_K - \int_{\partial K} \partial_n u_h \, (z - i_h z)\, ds \right\} + \int_{\Gamma_N} g\,(z - i_h z)\, ds.$$

Following the usual procedure for residual-based error estimators [41, 42], we combine each two boundary integrals over element edges to a normal jump and proceed with Cauchy–Schwarz to get

Proposition 4.1. *For the BEM-based FEM approximation of the Poisson equation, we have the a posteriori error estimate based on the classical localization*

$$|J(u - u_h)| \le \eta^{\text{CL}} := \sum_{K\in\mathcal{K}_h} \eta_K^{\text{CL}} \tag{4.3}$$

with

$$\eta_K^{\text{CL}} = \|f + \Delta u_h\|_K \, \|z - i_h z\|_K + \sum_{E\in\mathcal{E}(K)} \|R\|_E \, \|(z - i_h z)\|_E, \tag{4.4}$$

where R is the so-called edge residual, which is defined over each edge of the mesh as

$$R = \begin{cases} 0 & \text{on } \Gamma_D, \\ g - \partial_n u_h & \text{on } \Gamma_N, \\ -\frac{1}{2}[\partial_n u_h] & \text{else}. \end{cases}$$

By $[\partial_n u_h]$ we denote the jump of the u_h derivative in normal direction to the edge.

According to the definition of the trial space we have $\Delta u_h \in \mathcal{P}^{k-2}(K)$ in each $K \in \mathcal{K}_h$. Since most of the basis functions are harmonic, Δu_h is directly obtained by the expansion coefficients of the element bubble functions. The term $\partial_n u_h$ has already been approximated in $\mathcal{P}^{k-1}_{\text{pw,d}}(\partial K)$ by means of BEM.

In the literature (e.g., [2, 4, 6, 7, 34]), the local error indicator (4.4) is usually estimated in order to separate it into two parts such that $\eta_K^{\text{CL}} \leq \rho_K(u_h)\omega_K(z)$. The first part $\rho_K(u_h)$ contains the residual with the discrete solution u_h and the problem data and the second part $\omega_K(z)$ contains the adjoint sensitivity measure $z - i_h z$. The separation is obtained by further applications of the Cauchy–Schwarz inequality and reads in our notation

$$|J(u-u_h)| \leq \sum_{K\in\mathcal{K}_h} \underbrace{\left(\|f+\Delta u_h\|_K + h_K^{-1/2}\|R\|_{\partial K}\right)}_{=:\,\rho_K(u_h)} \underbrace{\left(\|z-i_h z\|_K + h_K^{1/2}\|z-i_h z\|_{\partial K}\right)}_{=:\,\omega_K(z)}. \tag{4.5}$$

In order to incorporate the polygonal structure of the elements and in particular the different numbers and lengths of their edges, we propose to split the L^2-norms over the boundaries of the elements. This refined manipulation yields

$$|J(u-u_h)| \leq \sum_{K\in\mathcal{K}_h} \underbrace{\left(\|f+\Delta u_h\|_K^2 + \sum_{E\in\mathcal{E}(K)} h_E^{-1}\|R\|_E^2\right)^{1/2}}_{=:\,\rho_K(u_h)} \cdot \underbrace{\left(\|z-i_h z\|_K^2 + \sum_{E\in\mathcal{E}(K)} h_E\|z-i_h z\|_E^2\right)^{1/2}}_{=:\,\omega_K(z)}. \tag{4.6}$$

The powers of h_K and h_E in (4.5) and (4.6) are chosen in such a way that the volume and boundary terms contribute in the right proportion. This weighting of the norms implicitly makes use of $h_E \sim h_K$, which is guaranteed by the regularity of the polygonal meshes. For triangular and quadrilateral meshes the terms h_E and h_K only differ by a small multiplicative factor. (For quadrilaterals it is $h_K = \sqrt{2}h_E$.) In polygonal meshes, however, the ratio $\frac{h_K}{h_E} < c_{\mathcal{K}}$ can be large and it might even blow up in the numerical tests, if the regularity is not enforced. Due to these reasons, it seems to be natural to weight directly the volume term $\|f+\Delta u_h\|_K$ with $\|z-i_h z\|_K$ and the edge term $\|R\|_E$ with $\|(z-i_h z)\|_E$ which gives rise to Proposition 4.1.

Remark 4.2. Similar estimators can be derived based on the adjoint form, where the residuals of the adjoint equation $\rho_K^*(u_h, z_h)$ are weighted with primal interpolation errors $\omega_K(u)$. A mixed estimator is then composed by using the primal and dual sub-estimators as explained in [7]. For linear problems (linear PDEs and linear goal functionals) all estimators, primal, dual, and mixed, are equivalent. Corresponding computational comparisons have been undertaken in [34] yielding the same numbers for goal functionals and their errors, but different meshes.

4.3.2 A Variational Error Estimator with PU Localization

We use a new localization approach [34] based on the variational formulation. Localization is simply based on introducing a partition-of-unity (PU) $\sum \psi_i \equiv 1$ into the global error representation Proposition 2.1:

$$J(u-u_h) = \sum_{i=1}^{N} \eta_i^{\text{PU}}, \tag{4.7}$$

where

$$\eta_i^{\text{PU}} = \{a(u-u_h, (z-i_h z)\psi_i)\}, \tag{4.8}$$

and more specifically for the diffusion problem

$$\eta_i^{\text{PU}} = \{(f, (z-i_h z)\psi_i) + (g, (z-i_h z)\psi_i)_{\Gamma_N} - (\nabla u_h, \nabla((z-i_h z)\psi_i))\}.$$

Thus, it remains to define a PU over polygonal meshes. In the case of triangular or quadrilateral meshes, the (bi-)linear basis functions are usually utilized, which are associated with nodes. The same is possible for polygonal meshes and the corresponding nodal basis functions, cf. (3.3). Let $K \in \mathcal{K}_h$ and consider

$$\psi = \sum_{x_i \in \mathcal{N}(K)} \psi_{x_i}.$$

Obviously, it is $\Delta\psi = 0$ and $\psi|_{\partial K} \equiv 1$. Therefore, ψ is the solution of a Dirichlet problem for the Laplace equation and because of the unique solvability we have $\psi \equiv 1$ on K. Consequently, the nodal basis functions fulfils the partition-of-unity property and we can use them in (4.7).

The corresponding error indicators η_i^{PU} are node-wise contributions of the error and one might usually proceed in one of the two ways:

- In a node-wise fashion: if a node x_i is picked for refinement, all elements touching this node will be refined. This procedure automatically leads to a patch-refined structure of the mesh.
- Alternatively, one could also first assemble element wise for each $K \in \mathcal{K}_h$ indicators by summing up all indicators belonging to nodes of this element and then carry out adaptivity in the usual element-wise way.

However, in this paper, we propose a third possibility:

- An element-wise partition-of-unity $\sum_{K\in\mathcal{K}_h} \chi_K$ is applied in order to obtain directly an element-wise indicator.

In the following we explain the last idea in more detail. For this reason, let $\sigma(x_i) = |\{K \in \mathcal{K}_h : x_i \in \mathcal{N}(K)\}|$ be the number of neighboring elements to the node $x_i \in \mathcal{N}_h$. Then we can write

$$1 \equiv \sum_{x_i\in\mathcal{N}_h} \psi_{x_i} = \sum_{K\in\mathcal{K}_h} \sum_{x_i\in\mathcal{N}(K)} \frac{1}{\sigma(x_i)} \psi_{x_i} = \sum_{K\in\mathcal{K}_h} \chi_K \quad \text{on } \Omega,$$

and thus obtain a new partition-of-unity employing the element-wise functions

$$\chi_K = \sum_{x_i\in\mathcal{N}(K)} \frac{1}{\sigma(x_i)} \psi_{x_i}. \tag{4.9}$$

The support of χ_K is local and covers the neighboring elements of K, namely

$$\operatorname{supp}\chi_K = \overline{\{x \in K' : K' \in \mathcal{K}_h,\ \overline{K} \cap \overline{K'} \neq \emptyset\}}.$$

Consequently, when we refer from now on to the PU-based localization technique, we mean the following error representation:

Proposition 4.2. *For the BEM-based FEM approximation of the Poisson equation, we have the element-wise PU-DWR a posteriori error representation and estimate*

$$J(u-u_h) = \eta^{\mathrm{PU}} := \sum_{K\in\mathcal{K}_h} \eta_K^{\mathrm{PU}} \quad \textit{and} \quad |J(u-u_h)| \le \eta_{\mathrm{abs}}^{\mathrm{PU}} := \sum_{K\in\mathcal{K}_h} |\eta_K^{\mathrm{PU}}|,$$

respectively, with

$$\eta_K^{\mathrm{PU}} = (f, (z - i_h z)\chi_K) + (g, (z - i_h z)\chi_K)_{\Gamma_N} - (\nabla u_h, \nabla((z - i_h z)\chi_K)).$$

We finish this section by three comments on the practical realization:

Remark 4.3. Due to the identity in (4.9), the element-wise indicators η_K^{PU} can be interpreted as weighted combinations of node-wise indicators η_i^{PU} defined by (4.8) and exploiting the nodal basis functions ψ_{x_i}.

Remark 4.4. Even if high-order approximations are used for the primal and dual problems, the PU can be realized using a lowest order method, e.g., of linear type.

Remark 4.5. We finally emphasize that the weak error estimator without PU can be used for mesh refinement but yields a significant overestimation of the error. That is of course equivalent to saying that in the classical form the integration by parts is not carried out. A theoretical justification to apply integration by parts can be found in [11].

4.4 Mesh Adaptation Algorithm

Let an error tolerance TOL be given. Mesh adaption is realized using extracted local error indicators from Proposition 4.1 or Proposition 4.2, respectively. This information is used to adapt the mesh using the

following strategy:
(1) **Solve:** Compute the primal solution u_h and the (higher-order) dual solution z_h^* on the present mesh $\mathcal{K}_h$.
(2) **Estimate:**
- Determine the indicator η_K for each element K.
- Compute the sum of all indicators $\eta := \sum_K \eta_K$ and $\eta_{\text{abs}} := \sum_K |\eta_K|$. (Note that $\eta_{\text{abs}}^{\text{CL}} = \eta^{\text{CL}}$ but $\eta_{\text{abs}}^{\text{PU}} \neq \eta^{\text{PU}}$.)
- Check if the stopping criterion is satisfied, i.e., $|\eta| \leq \text{TOL}$, then accept u_h within the tolerance TOL. Otherwise, proceed to the following step.

(3) **Mark** all elements K that have values $|\eta_K|$ above the average $\frac{\alpha}{N}\eta_{\text{abs}}$ (where N denotes the total number of elements of the mesh $\mathcal{K}_h$ and $\alpha \approx 1$).
(4) **Refine** the mesh by splitting the marked elements.

5 Numerical Tests

In this section, we substantiate our developments with several different numerical tests and various goal functionals. In the first example, we consider the standard Poisson problem with a regular goal functional. The second example considers a norm-based goal functional. In the third example we study adaptivity in detail. In all examples, we compare the classical and PU localization techniques. Moreover, we compare as previously mentioned different ways to approximate the dual solution. The programming code is based on C, is self-developed and an extension of [43].

In analyzing our results, we notice that the tables and graphs are given with respect to the number of degrees of freedom (DoF) in the following. This highlights the fact that the considered sequences of meshes may have the same shapes of elements, but have different numbers of degrees of freedom. This behavior is due to the mesh requirement for the local post-processing involving additional nodes on the boundaries of the elements. The degrees of freedom are also the usual criterion for adaptive refined meshes. Furthermore, we point out that not all theoretical assumptions on the mesh regularity in Section 3.1 are enforced in the tests. During the refinement, the edge lengths may degenerate with respect to the element diameter. If not otherwise stated, all appearing volume integrals are treated by numerical quadrature over polygonal elements as described in Section 3.5.

5.1 Problem 1: Verification in Terms of a Domain Goal Functional

Let $\Omega = (0, 1)^2$. We consider the boundary value problem

$$\begin{cases} -\Delta u = 1 & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

on two uniform sequences of meshes depicted in Figure 2 (left and middle columns). With a little abuse of notation we denote the sequence of meshes by $\mathcal{K}_{2h}$ and $\mathcal{K}_h$ for the left and middle column in Figure 2, respectively. The goal functional is chosen as $J(v) = \int_\Omega v\, dx$ such that the dual and primal problems coincide. The regularity of the solutions is only limited by the corners of the domain and it is $u, z \in H^{3-\varepsilon}(\Omega)$ for arbitrary small $\varepsilon > 0$. Furthermore, we use the reference value $J(u) \approx 0.03514425375 \pm 10^{-10}$ taken from [34] for the convergence analysis.

In the first experiment, we compare the different representations of the classical localization technique given in Section 4.3.1. Here we detect a significant difference depending on the partition into residual terms and dual weights of the classical estimator. The primal solution is approximated in V_h^1 and the dual solution is treated by globally higher-order, i.e., $z_h^* = z_h^{(2)}$. For this choice, we do not need the requirement on the node count for the meshes. Therefore, we perform the computations on the mesh sequence $\mathcal{K}_{2h}$ of the unite square Ω. The effectivity index I_{eff} is presented in Table 1. For comparisons, we also provide results computed on a sequence of structured meshes with rectangular elements. We observe that the effectivity index is indeed

closer to one for the sharpened estimates. Therefore, we only apply (4.3) in the following experiments for the classical localization. Furthermore, the comparison with structured meshes indicate that the polygonal shapes of the elements do not influence the effectivity on these uniform refined meshes.

Next, we compare the effectivity index for the PU-based and the classical localization with (4.3). The problems are approximated with $k = 1, 2$. In Table 2, we show I_{eff} for the choice $z_h^* = z_h^{(k+1)}$ on a sequence of meshes $\mathcal{K}_{2h}$. The effectivity index for the PU localization is close to one whereas the classical localization lacks on effectivity for the first order approximation $k = 1$. For $k = 2$ the effectivity $I_{\text{eff}}^{\text{CL}}$ is improved.

Furthermore, in Table 3, we applied the local post-processing of $z_h^{(k)}$ in order to construct $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$ and therefore the computations are done on the sequence of meshes $\mathcal{K}_h$, which fulfil the condition on the node count. Although the elements have the same shapes in the sequences of meshes, the number of degrees of freedom is larger in $\mathcal{K}_h$ than in $\mathcal{K}_{2h}$. Both localization strategies show good effectivity in Table 3. Due to the local post-processing instead of the globally higher-order approximation for the dual solution, the com-

Polygonal-Meshes					Quad-Meshes		
DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{CL}}$ (4.5)	$I_{\text{eff}}^{\text{CL}}$ (4.6)	DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)
4	$5.52 \cdot 10^{-3}$	3.01	6.30	3.31	9	$2.56 \cdot 10^{-3}$	2.91
8	$3.48 \cdot 10^{-3}$	2.21	6.56	4.03	49	$6.51 \cdot 10^{-4}$	2.93
13	$4.30 \cdot 10^{-3}$	1.74	4.83	2.59	121	$2.90 \cdot 10^{-4}$	2.93
25	$2.33 \cdot 10^{-3}$	2.02	5.42	2.98	225	$1.64 \cdot 10^{-4}$	2.92
57	$1.32 \cdot 10^{-3}$	1.99	5.89	3.37	361	$1.05 \cdot 10^{-4}$	2.92
129	$5.36 \cdot 10^{-4}$	2.29	6.47	3.64	529	$7.27 \cdot 10^{-5}$	2.92
289	$2.63 \cdot 10^{-4}$	2.34	6.73	3.84	729	$5.35 \cdot 10^{-5}$	2.92
620	$1.17 \cdot 10^{-4}$	2.65	7.45	4.28	961	$4.09 \cdot 10^{-5}$	2.92
1,297	$5.67 \cdot 10^{-5}$	2.66	7.48	4.24	1,225	$3.23 \cdot 10^{-5}$	2.92

Table 1: Problem 1 approximated with $u_h \in V_h^1$, and dual solution treated by globally higher-order, i.e., $z_h^* = z_h^{(2)}$. Comparison of effectivity for different representations of the classic localization on a mesh sequence $\mathcal{K}_{2h}$ and on structured meshes.

DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$	DoF	$J(u - u_h^{(2)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$
8	$3.48 \cdot 10^{-3}$	2.21	1.16	35	$1.76 \cdot 10^{-4}$	2.09	1.30
13	$4.30 \cdot 10^{-3}$	1.74	0.99	65	$7.36 \cdot 10^{-5}$	1.59	1.24
25	$2.33 \cdot 10^{-3}$	2.02	1.00	129	$1.41 \cdot 10^{-5}$	1.65	1.30
57	$1.32 \cdot 10^{-3}$	1.99	1.01	273	$4.00 \cdot 10^{-6}$	1.60	1.27
129	$5.36 \cdot 10^{-4}$	2.29	1.03	577	$7.80 \cdot 10^{-7}$	1.68	1.34
289	$2.63 \cdot 10^{-4}$	2.34	1.04	1,217	$1.93 \cdot 10^{-7}$	1.66	1.34
620	$1.17 \cdot 10^{-4}$	2.65	1.07	2,519	$3.63 \cdot 10^{-8}$	1.80	1.41
1,297	$5.67 \cdot 10^{-5}$	2.66	1.07	5,153	$4.95 \cdot 10^{-9}$	3.62	2.90

Table 2: Problem 1 approximated with $u_h \in V_h^k$, $k = 1, 2$, and dual solution treated by globally higher-order, i.e., $z_h^* = z_h^{(k+1)}$. Comparison of effectivity for PU localization and classical localization with (4.3) on mesh sequence $\mathcal{K}_{2h}$.

DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$	DoF	$J(u - u_h^{(2)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$
25	$3.41 \cdot 10^{-3}$	1.40	0.92	69	$1.21 \cdot 10^{-5}$	1.20	0.57
45	$1.76 \cdot 10^{-3}$	1.96	0.96	129	$1.07 \cdot 10^{-5}$	1.27	0.86
89	$9.17 \cdot 10^{-4}$	2.05	0.95	257	$8.69 \cdot 10^{-7}$	1.19	0.68
193	$4.63 \cdot 10^{-4}$	2.36	0.96	545	$6.76 \cdot 10^{-7}$	1.50	1.08
465	$2.31 \cdot 10^{-4}$	1.99	0.93	1,249	$4.36 \cdot 10^{-8}$	1.34	0.79
953	$1.14 \cdot 10^{-4}$	2.10	0.95	2,545	$2.65 \cdot 10^{-8}$	1.56	1.14
2,069	$5.66 \cdot 10^{-5}$	2.12	0.95	5,417	$1.50 \cdot 10^{-9}$	2.01	1.39
4,269	$2.83 \cdot 10^{-5}$	2.09	0.96	11,097	$< 10^{-9}$	–	–

Table 3: Problem 1 approximated with $u_h \in V_h^k$, $k = 1, 2$ and dual solution treated by local post-processing, i.e., $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$. Comparison of effectivity for classical with (4.3) and PU localization on mesh sequence $\mathcal{K}_h$.

putational cost is significantly reduced compared to the experiments for Table 2. We finally remark that for obtaining errors of similar order in the case of $k = 2$, the meshes in Table 2 are one times more refined in comparison to the method presented in Table 3. However, as just explained, the mesh itself is coarser but the number of degrees of freedom is higher on the other hand when using the local post-processing of $z_h^{(k)}$.

5.2 Problem 2: A Norm-Based Goal Functional

In our second example, let again $\Omega = (0, 1)^2$. We consider the boundary value problem

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

where f is chosen such that $u(x_1, x_2) = \sin(\pi x_1)\sin(\pi x_2)$ is the analytical solution. As in the previous problem, we compare the different localization techniques and the two choices of z_h^*. The computations are done solely on the sequence of meshes fulfilling the node count condition, which is depicted in Figure 2 in the middle column. The error functional is chosen as

$$J(v) = \frac{(u - u_h, v)}{\|u - u_h\|^2}$$

such that $J(u - u_h) = \|u - u_h\|$. Our results of the effectivity indices are shown in Tables 4 and 5. All indices are close to one and behave similar to those of the previous Problem 1. Note that I_{eff}^{PU} is hardly effected by the different approximations of the dual solution and also the classical localization shows comparable effectivity. Consequently, the computationally less expensive post-processing is to favor over the higher-order approximation of the dual solution in practical applications.

DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$	DoF	$J(u - u_h^{(2)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$
25	$3.80 \cdot 10^{-2}$	1.76	0.92	69	$5.64 \cdot 10^{-3}$	1.38	0.95
45	$2.10 \cdot 10^{-2}$	1.99	0.98	129	$2.90 \cdot 10^{-3}$	1.26	0.95
89	$1.05 \cdot 10^{-2}$	1.91	0.81	257	$8.48 \cdot 10^{-4}$	1.27	0.97
193	$5.34 \cdot 10^{-3}$	2.05	0.83	545	$3.57 \cdot 10^{-4}$	1.29	0.96
465	$2.59 \cdot 10^{-3}$	1.98	0.82	1,249	$1.17 \cdot 10^{-4}$	1.46	0.93
953	$1.35 \cdot 10^{-3}$	2.06	0.83	2,545	$4.04 \cdot 10^{-5}$	1.35	0.98
2,069	$6.75 \cdot 10^{-4}$	2.11	0.82	5,417	$1.59 \cdot 10^{-5}$	1.37	0.99
4,269	$3.38 \cdot 10^{-4}$	2.04	0.84	11,097	$5.26 \cdot 10^{-6}$	1.36	1.05

Table 4: Problem 2 approximated with $u_h \in V_h^k$, $k = 1, 2$ and dual solution treated by globally higher-order, i.e., $z_h^* = z_h^{(k+1)}$. Comparison of effectivity for PU localization and classical localization with (4.3) on mesh sequence $\mathcal{K}_h$.

DoF	$J(u - u_h^{(1)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$	DoF	$J(u - u_h^{(2)})$	$I_{\text{eff}}^{\text{CL}}$ (4.3)	$I_{\text{eff}}^{\text{PU}}$
25	$3.80 \cdot 10^{-2}$	1.65	0.83	69	$5.64 \cdot 10^{-3}$	1.26	0.95
45	$2.10 \cdot 10^{-2}$	1.79	0.86	129	$2.90 \cdot 10^{-3}$	1.28	0.96
89	$1.05 \cdot 10^{-2}$	2.29	0.84	257	$8.48 \cdot 10^{-4}$	1.28	0.97
193	$5.34 \cdot 10^{-3}$	2.16	0.80	545	$3.57 \cdot 10^{-4}$	1.30	0.95
465	$2.59 \cdot 10^{-3}$	2.24	0.82	1,249	$1.17 \cdot 10^{-4}$	1.29	0.89
953	$1.35 \cdot 10^{-3}$	2.20	0.82	2,545	$4.04 \cdot 10^{-5}$	1.32	0.97
2,069	$6.75 \cdot 10^{-4}$	2.25	0.82	5,417	$1.59 \cdot 10^{-5}$	1.33	0.97
4,269	$3.38 \cdot 10^{-4}$	2.19	0.82	11,097	$5.26 \cdot 10^{-6}$	1.34	1.01

Table 5: Problem 2 approximated with $u_h \in V_h^k$, $k = 1, 2$, and dual solution treated by local post-processing, i.e., $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$. Comparison of effectivity for classical with (4.3) and PU localization on mesh sequence $\mathcal{K}_h$.

5.3 Problem 3: Adaptivity

In this final example, let $\Omega = (-1, 1) \times (-1, 1) \setminus [0, 1] \times [-1, 0]$ be an L-shaped domain and its boundary is split into $\Gamma_D = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 \in [0, 1],\ x_2 = 0 \text{ or } x_1 = 0,\ x_2 \in [-1, 0]\}$ and $\Gamma_N = \partial\Omega \setminus \Gamma_D$. We consider the boundary value problem

$$\begin{cases} -\Delta u = 0 & \text{in } \Omega, \\ u = 0 & \text{on } \Gamma_D, \\ \partial_n u = g & \text{on } \Gamma_N, \end{cases}$$

where g is chosen with the help of polar coordinates (r, ϕ) such that

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

is the exact solution. This is a classical problem for mesh adaptivity, since the gradient of the solution inherits a singularity at the reentrant corner in the origin of the coordinate system. It holds $u \in H^{5/3}(\Omega)$. The considered goal functional is a point evaluation

$$J(v) = v(x^*),$$

where x^* is chosen as the upper right node inside the domain, which is adjacent to six elements of the initial mesh, see Figure 3 (left). We apply the adaptive strategy and compare the resulting meshes for the different localization techniques and approximations of the dual solution.

In Figure 3, we display the initial mesh and the adaptively refined meshes for $k = 2$ after ten refinement steps for the classical and the PU localization. A zoom-in highlighting the resulting shapes of adaptively refined elements is provided in Figure 4. The dual problem is treated by a globally higher-order discretization, i.e., $z_h^* = z_h^{(k+1)}$. This experiment has been carried out on sequences of meshes, which do not fulfil the condition on the node count. The elements in the initial mesh are triangles. The adaptive process, however, produces naturally polygonal elements during the local refinements. These refinements are located in the expected regions.

The resulting meshes for the experiments with local post-processing for the dual solution, that is, for $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$, are visualized in Figure 5. As before, As before, a zoom-in highlighting the resulting shapes of adaptively refined elements is provided in Figure 6. This strategy is carried out on a sequence of meshes fulfilling the condition on the node count, and thus, the triangular elements in the initial mesh are actually degenerated hexagons. The refinement pattern is similar to the one in Figure 3. But we observe that there are less refinements far from the singularity and the point x^* after ten steps. Due to the additional nodes on the boundary of the elements, there are more degrees of freedom per element. Consequently, the approximation over the degenerated hexagonal elements (with triangular shape) is more accurate compared to the corresponding triangular elements in Figure 3.

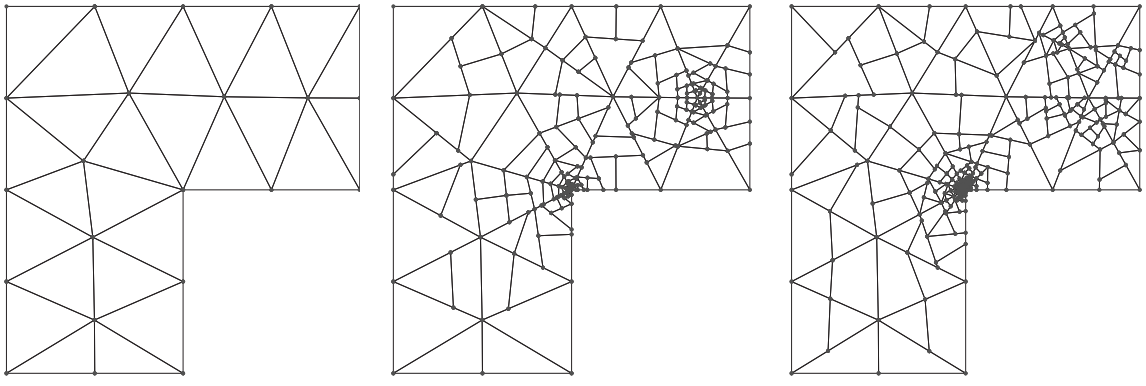

Figure 3: Initial mesh of the L-shaped domain in Problem 3 with triangular elements (left) and adaptive meshes for $k = 2$ after ten refinements for classical (middle) and PU (right) localization, where the dual problem is treated by globally higher-order, i.e., $z_h^* = z_h^{(k+1)}$.

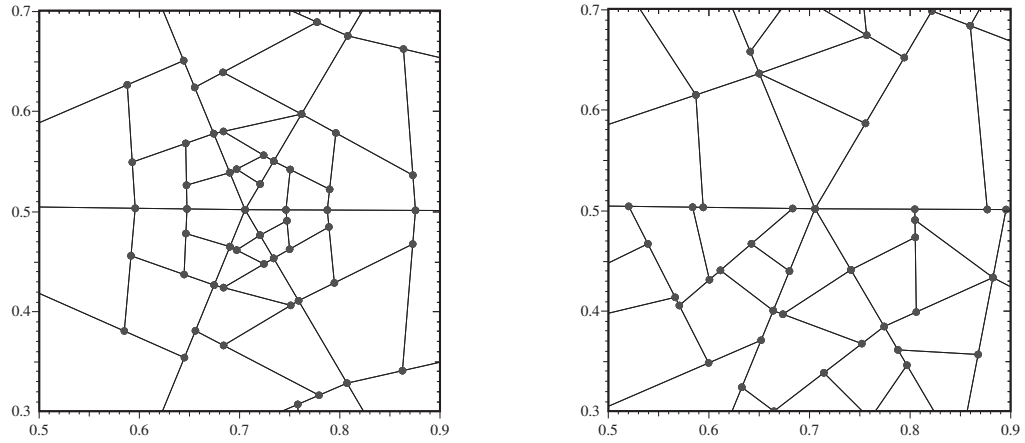

Figure 4: Zoom into L-shaped domain in Problem 3 with adaptive meshes for $k = 2$ after ten refinements for classical (left) and PU (right) localization, where the dual problem is treated by globally higher-order, i.e., $z_h^* = z_h^{(k+1)}$.

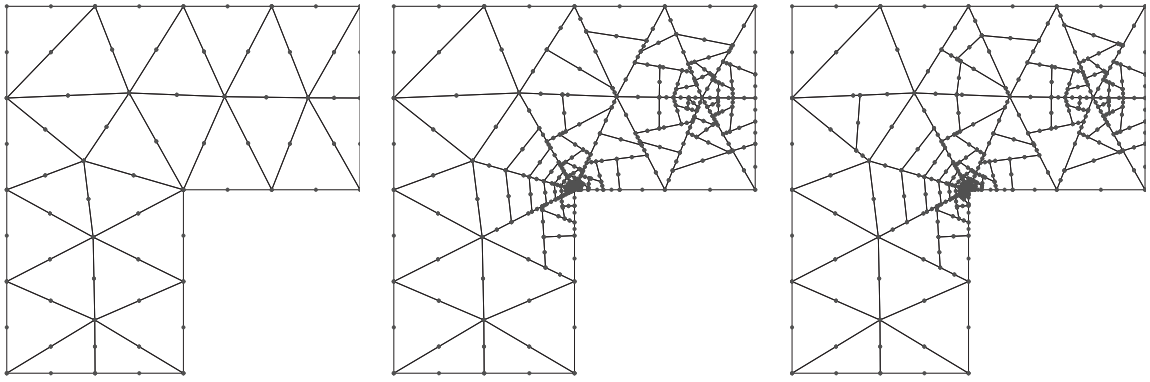

Figure 5: Initial mesh of the L-shaped domain in Problem 3 with triangular elements (left), which are actually degenerated hexagons, and adaptive meshes for $k = 2$ after ten refinements for classical (middle) and PU (right) localization, where the dual problem is treated by local post-processing, i.e., $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$.

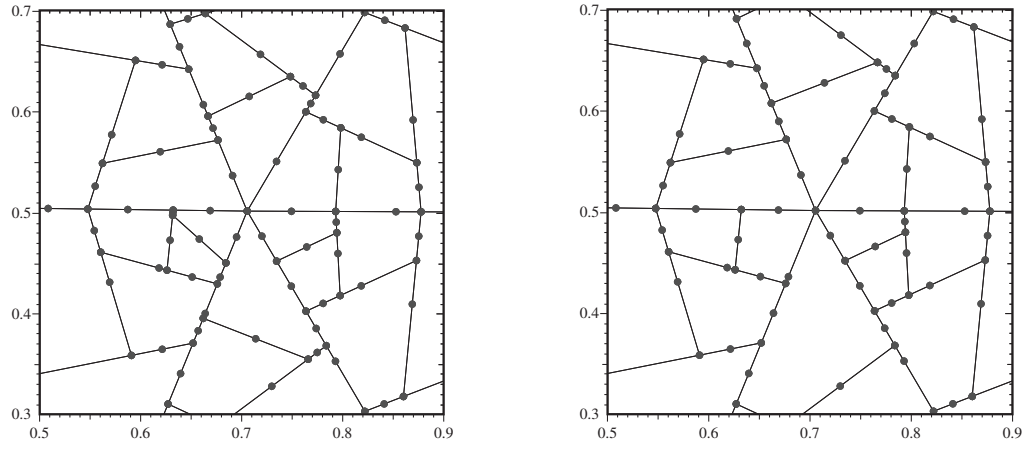

Figure 6: Zoom into L-shaped domain in Problem 3 with adaptive meshes for $k = 2$ after ten refinements for classical (left) and PU (right) localization, where the dual problem is treated by local post-processing, i.e., $z_h^* = \mathfrak{P}_{2h}^{k+1} z_h^{(k)}$.

In order to study convergence, we plot the absolute values of the errors and the estimators with respect to the number of degrees of freedom on a logarithmic scale. The abbreviation $e = u - u_h$ is used in the key of the plots. If we run the computations on a sequence of uniform refined meshes, the convergence slows down due to the singularity located at the reentrant corner. The tests are performed on a uniform sequence $\mathcal{K}_{2h}$, which does not fulfil the condition on the node count, and on a uniform sequence $\mathcal{K}_h$, which fulfils the condition. The initial meshes are visualized in Figure 3 and 5, respectively. The corresponding convergence graphs are given in Figure 7 for $k = 1, 2$. In these graphs, the error estimator η^{PU} is given additionally, which clearly reflects the behavior of the true error $J(e)$.

Next, we apply the adaptive refinement strategy discussed in Section 4.4. The following computations are run on meshes fulfilling the condition on the node count only. We have performed 25 adaptive refinement steps for the different localization techniques and the two choices of z_h^*. Since $f \equiv 0$ in this test, we directly obtain from (3.5) that $u_{h,B} \equiv 0$ and thus $u_h = u_{h,H} \in V_{h,H}^k$. Consequently, we can reduce the volume integral in η^{PU} to the boundaries of the elements. Let $K' \in \mathcal{K}_h$ with $\overline{K'} \cap \overline{K} \neq \emptyset$; then it is

$$(\nabla u_h, \nabla((z - i_h z)\chi_K))_{K'} = (\gamma_1^{K'} u_h, \gamma_0^{K'}((z - i_h z)\chi_K))_{\partial K'}$$

according to Green's first identity. This reformulation improved the accuracy of the numerical results. The convergence graphs are given in Figure 8 for the PU localization and in Figure 9 for the classical localization stated in Proposition 4.1. In contrast to the uniform refinement strategy, we recover higher convergence rates, which are not limited by the regularity of the primal solution. Both localization techniques show comparable performance in Figure 8 and 9, respectively. The PU localization, however, has a better effectivity while less computational effort is spent for the dual problem. Furthermore, we point out that the convergence is actually faster than expected. Indeed for finite elements, L^∞-regularity results for irregular meshes have been established in [38] (with further references to regular meshes cited therein). Specifically, assuming enough regularity, for $k = 2$ we would expect a behavior like $\mathcal{O}(\text{DoF}^{-3/2})$. For $k = 1$ we would expect $\mathcal{O}(\text{DoF}^{-1})$ including a logarithm term [38]. However, in our computations, we observe for $k = 2$ a behavior like $\mathcal{O}(\text{DoF}^{-3})$. For $k = 1$ the error $J(e)$ seems to converge with $\mathcal{O}(\text{DoF}^{-2})$ rather than with $\mathcal{O}(\text{DoF}^{-1})$ indicated by the estimators η^{PU} and η^{CL}. These effects might be caused by the special meshes, which include additional nodes in order to fulfil the condition on the node count during the refinement. Furthermore, the implementation allows edge degeneration, that is excluded in the current theory of most polygonal discretization techniques, but which might be beneficial. These observations will be analyzed in future research.

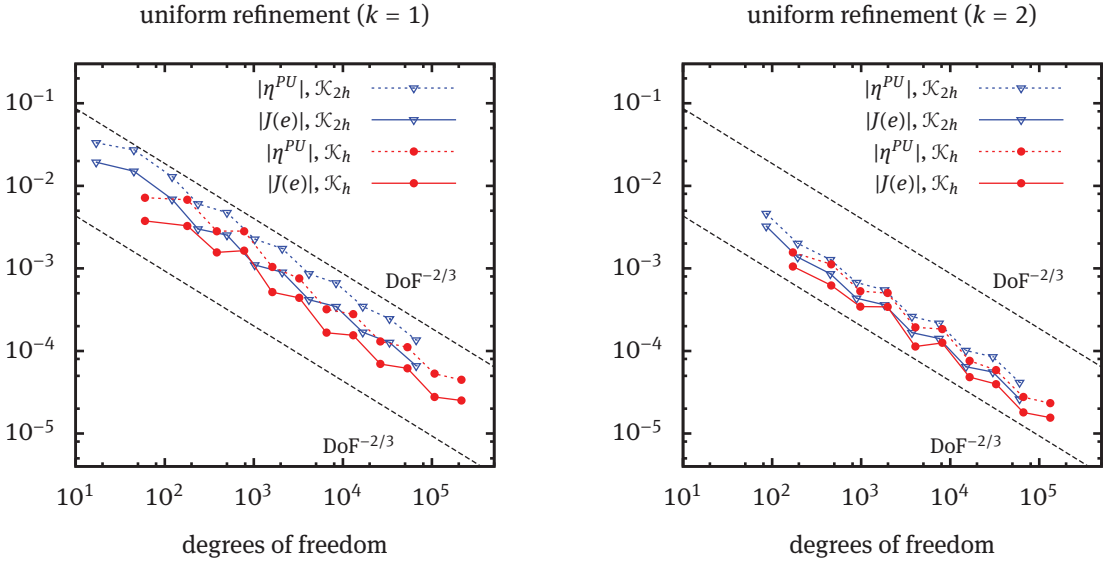

Figure 7: Convergence of uniform refinement strategy with respect to the number of degrees of freedom for Problem 3 with PU localization and $z_h^* = z_h^{(k+1)}$.

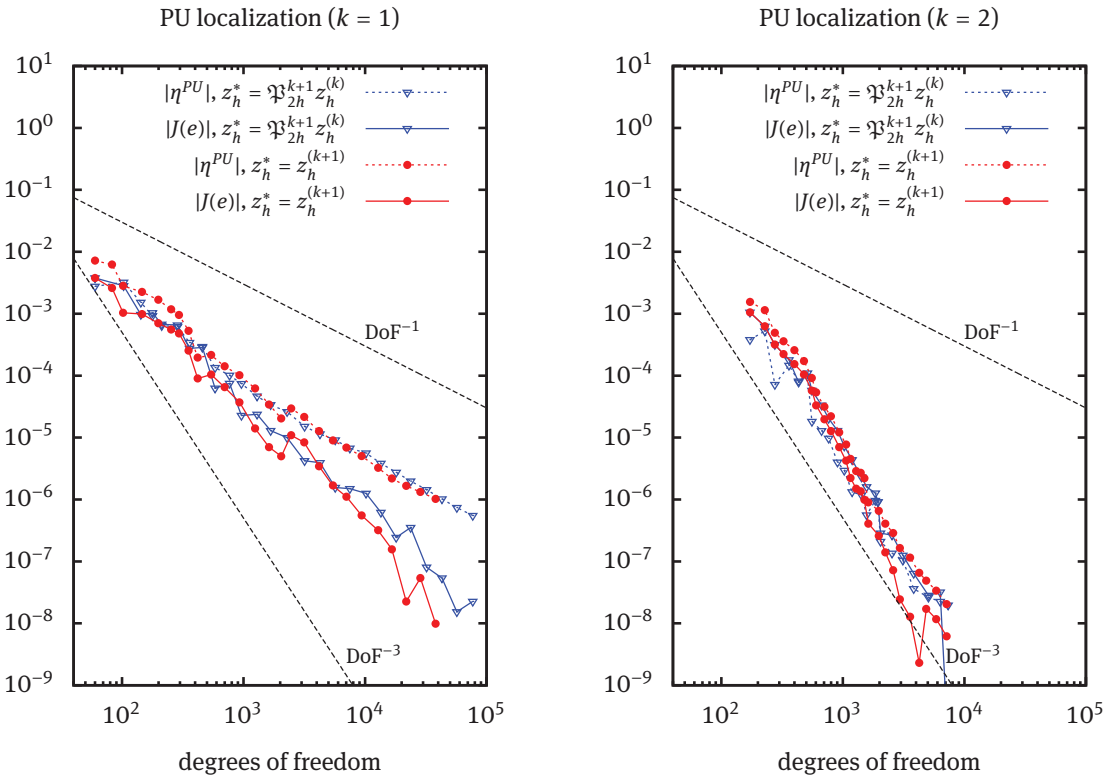

Figure 8: Convergence of adaptive refinement strategy with respect to the number of degrees of freedom for Problem 3 with PU localization.

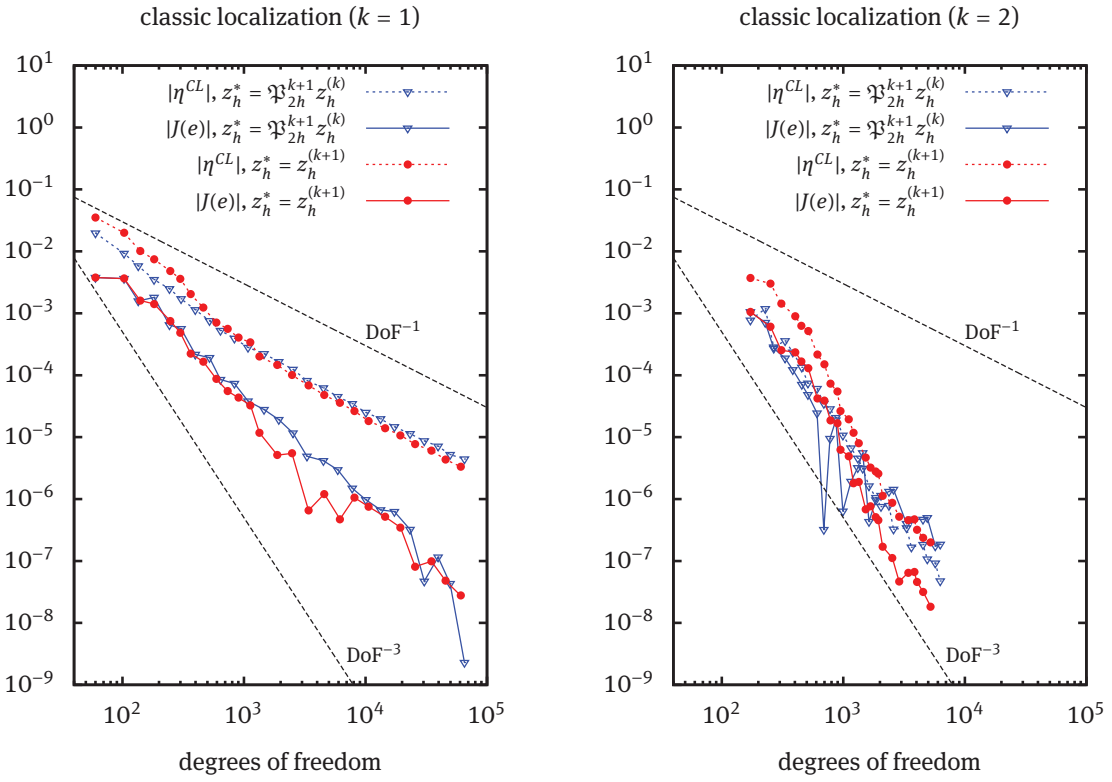

Figure 9: Convergence of adaptive refinement strategy with respect to the number of degrees of freedom for Problem 3 with classic localization.

6 Conclusion

In this work we realized a variational form of the dual-weighted residual estimator on general polygonal meshes. Specifically, the discretization is based on a BEM-based FEM technique that allows for general meshes and flexible discretizations. Moreover, the crucial part of the error estimator is an appropriate localization for which we used a classical technique and a variational technique. For solving the dual problem, we proposed a new technique that neither needs higher-order global approximations nor local higher-order approximations on patched meshes, but only one element. This results in an element-based post-processing method. In the numerical examples the basic properties of our suggested methodology are demonstrated. Here, we focused on linear diffusion equations since they are well understood and they allow for a detailed computational analysis of our proposed dual-weighted residual estimators. In these examples, we considered different boundary conditions and various goal functionals. Furthermore, linear as well as second-order discretizations have been adopted and our single element approximation of the dual solution has been compared to a globally higher-order approximation. As demonstrated in all these numerical tests and our observations, the method has a potential to be extended for complex problems that require special and adapted meshes while measuring goal functionals with a desired accuracy. Such studies will be future perspectives.

References

[1] M. Ainsworth and J. T. Oden, A posteriori error estimation in finite element analysis, *Comput. Methods Appl. Mech. Engrg.* **142** (1997), no. 1–2, 1–88.

[2] M. Ainsworth and J. T. Oden, *A Posteriori Error Estimation in Finite Element Analysis*, Pure Appl. Math. (New York), John Wiley & Sons, New York, 2000.

[3] P. F. Antonietti, L. B. da Veiga, C. Lovadina and M. Verani, Hierarchical a posteriori error estimators for the mimetic discretization of elliptic problems, *SIAM J. Numer. Anal.* **51** (2013), no. 1, 654–675.

[4] W. Bangerth and R. Rannacher, *Adaptive Finite Element Methods for Differential Equations*, Lect. Math. ETH Zürich, Birkhäuser, Basel, 2003.

[5] R. Becker, H. Kapp and R. Rannacher, Adaptive finite element methods for optimal control of partial differential equations: Basic concepts, *SIAM J. Optim. Control* **39** (2000), 113–132.

[6] R. Becker and R. Rannacher, Weighted a posteriori error control in FE methods, in: *ENUMATH'97—Proceedings of the 2nd European Conference on Numerical Mathematics and Advanced Applications* (Heidelberg 1997), World Scientific, Singapore (1995), 18–22.

[7] R. Becker and R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, *Acta Numer.* **10** (2001), 1–102.

[8] M. Braack and A. Ern, A posteriori control of modeling errors and discretization errors, *Multiscale Model. Simul.* **1** (2003), no. 2, 221–238.

[9] A. Cangiani, E. H. Georgoulis, T. Prayer and O. J. Sutton, A posteriori error estimates for the Virtual Element Method, *Numer. Math.* (2017), DOI 10.1007/s00211-017-0891-9.

[10] C. Carstensen, Estimation of higher sobolev norm from lower order approximation, *SIAM J. Numer. Anal.* **42** (2004), 2136–2147.

[11] C. Carstensen and R. Verfürth, Edge residuals dominate a posteriori error estimates for low order finite element methods, *SIAM J. Numer. Anal.* **36** (1999), no. 5, 1571–1587.

[12] L. Chen, J. Wang and X. Ye, A posteriori error estimates for weak Galerkin finite element methods for second order elliptic problems, *J. Sci. Comput.* **59** (2014), no. 2, 496–511.

[13] D. Copeland, U. Langer and D. Pusch, From the boundary element domain decomposition methods to local Trefftz finite element methods on polyhedral meshes, in: *Domain Decomposition Methods in Science and Engineering XVIII*, Lect. Notes Comput. Sci. Eng. 70, Springer, Berlin (2009), 315–322.

[14] L. B. da Veiga and G. Manzini, Residual a posteriori error estimation for the virtual element method for elliptic problems, *ESAIM Math. Model. Numer. Anal.* **49** (2015), no. 2, 577–599.

[15] B. Endtmayer, *Adaptive mesh refinement for multiple goal functionals*, Master's thesis, Johannes Kepler University Linz, Institute of Computational Mathematics, 2017.

[16] B. Endtmayer and T. Wick, A partition-of-unity dual-weighted residual approach for multi-objective goal functional error estimation applied to elliptic problems, *Comput. Methods Appl. Math.* **17** (2017), no. 4, 575–599.

[17] K. Eriksson, D. Estep, P. Hansbo and C. Johnson, Introduction to adaptive methods for differential equations, in: *Acta Numerica 1995*, Cambridge University Press, Cambridge (1995), 105–158.
[18] A. L. Gain, C. Talischi and G. H. Paulino, On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes, *Comput. Methods Appl. Mech. Engrg.* **282** (2014), 132–160.
[19] M. B. Giles and E. Süli, Adjoint methods for PDEs: A posteriori error analysis and postprocessing by duality, *Acta Numerica* **11** (2002), 145–236.
[20] C. Hofreither, U. Langer and S. Weißer, Convection adapted BEM-based FEM, *ZAMM Z. Angew. Math. Mech.* **96** (2016), no. 12, 1467–1481.
[21] O. A. Karakashian and F. Pascal, A posteriori error estimates for a discontinuous Galerkin approximation of second-order elliptic problems, *SIAM J. Numer. Anal.* **41** (2003), no. 6, 2374–2399.
[22] G. Kuru, C. V. Verhoosel, K. G. van der Zee and E. H. van Brummelen, Goal-adaptive isogeometric analysis with hierarchical splines, *Comput. Methods Appl. Mech. Engrg.* **270** (2014), 270–292.
[23] D. Kuzmin and S. Korotov, Goal-oriented a posteriori error estimates for transport problems, *Math. Comput. Simulation* **80** (2010), no. 8, 1674–1683.
[24] R. Lazarov, S. Repin and S. Tomar, Functional a posteriori error estimates for discontinuous Galerkin approximations of elliptic problems, *Numer. Methods Partial Differential Equations* **25** (2009), no. 4, 952–971.
[25] G. Manzini, A. Russo and N. Sukumar, New perspectives on polygonal and polyhedral finite element methods, *Math. Models Methods Appl. Sci.* **24** (2014), no. 8, 1665–1699.
[26] W. C. H. McLean, *Strongly Elliptic Systems and Boundary Integral Equations*, Cambridge University Press, Cambridge, 2000.
[27] R. H. Nochetto, A. Veeser and M. Verani, A safeguarded dual weighted residual method, *IMA J. Numer. Anal.* **29** (2009), no. 1, 126–140.
[28] J. T. Oden and S. Prudhomme, On goal-oriented error estimation for elliptic problems: Application to the control of pointwise errors, *Comput. Methods Appl. Mech. Engrg.* **176** (1999), 313–331.
[29] J. Peraire and A. T. Patera, Bounds for linear-functional outputs of coercive partial differential equations: Local indicators and adaptive refinement, in: *Advances in Adaptive Computational Methods in Mechanics*, Elsevier, Amsterdam (1998), 199–215.
[30] D. A. D. Pietro and A. Ern, A hybrid high-order locking-free method for linear elasticity on general meshes, *Comput. Methods Appl. Mech. Engrg.* **283** (2015), 1–21.
[31] R. Rannacher and F.-T. Suttmeier, A feed-back approach to error control in finite element methods: Application to linear elasticity, *Comput. Mech.* **19** (1997), no. 5, 434–446.
[32] R. Rannacher and F.-T. Suttmeier, A posteriori error estimation and mesh adaptation for finite element models in elasto-plasticity, *Comput. Methods Appl. Mech. Engrg.* **176** (1999), no. 1–4, 333–361.
[33] T. Richter, Goal-oriented error estimation for fluid-structure interaction problems, *Comput. Methods Appl. Mech. Engrg.* **223–224** (2012), 38–42.
[34] T. Richter and T. Wick, Variational localizations of the dual weighted residual estimator, *J. Comput. Appl. Math.* **279** (2015), 192–208.
[35] S. Rjasanow and S. Weißer, Higher order BEM-based FEM on polygonal meshes, *SIAM J. Numer. Anal.* **50** (2012), no. 5, 2357–2378.
[36] S. Rjasanow and S. Weißer, FEM with Trefftz trial functions on polyhedral elements, *J. Comput. Appl. Math.* **263** (2014), 202–217.
[37] A. Schroeder and A. Rademacher, Goal-oriented error control in adaptive mixed FEM for Signorini's problem, *Comput. Methods Appl. Mech. Engrg.* **200** (2011), no. 1–4, 345–355.
[38] R. Scott, Optimal L^∞ estimates for the finite element method on irregular meshes, *Math. Comp.* **30** (1976), no. 136, 681–697.
[39] O. Steinbach, *Numerical Approximation Methods for Elliptic Boundary Value Problems: Finite and Boundary Elements*, Springer, New York, 2007.
[40] F.-T. Suttmeier, *Numerical solution of Variational Inequalities by Adaptive Finite Elements*, Vieweg+Teubner, Wiesbaden, 2008.
[41] R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Wiley/Teubner, New York/Stuttgart, 1996.
[42] S. Weißer, Residual error estimate for BEM-based FEM on polygonal meshes, *Numer. Math.* **118** (2011), no. 4, 765–788.
[43] S. Weißer, *Finite Element Methods with local Trefftz trial functions*, Ph.D. thesis, Universität des Saarlandes, Saarbrücken, 2012.
[44] S. Weißer, Arbitrary order Trefftz-like basis functions on polygonal meshes and realization in BEM-based FEM, *Comput. Math. Appl.* **67** (2014), no. 7, 1390–1406.
[45] S. Weißer, Residual based error estimate for higher order Trefftz-like trial functions on adaptively refined polygonal meshes, in: *Numerical Mathematics and Advanced Applications—ENUMATH 2013*, Lect. Notes Comput. Sci. Eng. 103, Springer, Cham (2015), 233–241.

[46] S. Weißer, Residual based error estimate and quasi-interpolation on polygonal meshes for high order BEM-based FEM, *Comput. Math. Appl.* **73** (2017), no. 2, 187–202.
[47] T. Wick, Goal functional evaluations for phase-field fracture using PU-based DWR mesh adaptivity, *Comput. Mech.* **57** (2016), no. 6, 1017–1035.
[48] K. Zee, E. Brummelen, I. Akkerman and R. Borst, Goal-oriented error estimation and adaptivity for fluid-structure interaction using exact linearized adjoints, *Comput. Methods Appl. Mech. Engrg.* **200** (2011), 2738–2757.