Universität des Saarlandes

Fachrichtung Mathematik

Preprint Nr. 404

Recovery-based Error Estimators for the VEM and BEM-based FEM

Daniel Seibel and Steffen Weißer

Saarbrücken 2020

Fachrichtung Mathematik
Universität des Saarlandes

Preprint No. 404
submitted: January 23, 2020

Recovery-based Error Estimators for the VEM and BEM-based FEM

Daniel Seibel

Saarland University
Department of Mathematics
P.O. Box 15 11 50
66041 Saarbrücken
Germany
seibel@num.uni-sb.de

Steffen Weißer

Saarland University
Department of Mathematics
P.O. Box 15 11 50
66041 Saarbrücken
Germany
weisser@num.uni-sb.de

Edited by
FR Mathematik
Universität des Saarlandes
Postfach 15 11 50
66041 Saarbrücken
Germany

Fax: + 49 681 302 4443
e-Mail: preprint@math.uni-sb.de
WWW: https://www.math.uni-sb.de/

RECOVERY-BASED ERROR ESTIMATORS FOR THE VEM AND BEM-BASED FEM

DANIEL SEIBEL AND STEFFEN WEISSER

Abstract. In this article, we propose new gradient recovery schemes for the Virtual Element Method (VEM) and Boundary Element Method based Finite Element Method (BEM-based FEM). Supporting general polytopal meshes, the VEM and BEM-based FEM are highly flexible and effective tools for the numerical solution of boundary value problems in two and three dimensions. We post-process the gradient of the finite element approximation via local averaging to obtain a continuous approximation, which is called recovered gradient. For the BEM-based FEM and under certain requirements on the mesh, we demonstrate superconvergence of the recovered gradient, which means that it converges to the exact gradient at a higher rate than the untreated gradient. Moreover, we propose a simple yet very efficient a posteriori error estimator, which measures the difference between the unprocessed and recovered gradient for error estimation. As the VEM and BEM-based FEM are specifically suited for adaptive refinement, the resulting adaptive algorithms perform very well in numerical examples.

1. Introduction

Gradient or stress recovery has a long tradition in the context of Finite Element Methods (FEM) [1–5]. One of its primary fields of application is found in computational elasticity, where the stresses, fluxes and strains are often more important than the displacements. In this context, the FEM with piece-wise linear elements yields discontinuous results for the gradient, which may be inconvenient for applications like visualisation. Therefore, it is common practice to post-process the gradient of the finite element solution to obtain a continuous approximation of the gradient. This form of post-processing is usually called "gradient recovery", while the term "recovered gradient" refers to the post-processed gradient. Remarkably, experiments show that the increase of regularity of the latter is accompanied by an increase in accuracy as well. Under certain conditions on the underlying mesh and on the regularity of the solution, it is even found that the recovered gradient converges to the exact solution at a higher rate than the untreated gradient [6, 7]. This phenomenon is well known as gradient superconvergence or superconvergent recovery. In the past decades, it has been extensively studied and several different post-processing strategies have been proposed, for example the L^2-recovery [8] or the popular patch recovery by Zienkiewicz and Zhu [9], which follows a least-squares approach to determine the recovered gradient.

Besides plain gradient reconstruction, the second main application is found in the field of a posteriori error estimation [4]. The approach is rather intuitive: based

2010 *Mathematics Subject Classification.* 65N30,65N38,65N50.

Key words and phrases. BEM-based FEM, VEM, gradient recovery, superconvergence, a posteriori error estimators, polygonal meshes.

on the observation that the post-processed gradient is more accurate than the unprocessed gradient, one measures the distance between these two approximations. In other words, if ∇u is the exact gradient, ∇u_h the gradient of the finite element solution and $\mathcal{G}u_h$ its reconstruction, the error is approximated by

$$\int_\Omega |\nabla u - \nabla u_h|^2 \, dx \approx \int_\Omega |\mathcal{G}u_h - \nabla u_h|^2 \, dx \,.$$

Such recovery-based error estimators are quite appealing because of several aspects. First of all, the post-processing, and hence the estimator, is typically designed to be cheap, fast and easy to implement. Secondly, there is a great variety of different recovery strategies available, which highlights the flexibility of the approach. Last but not least, the performance of the estimator is remarkably good, as the estimates are very accurate in most cases.

Yet, the application of gradient recovery is almost limited to standard FEM on triangular or quadrilateral discretisations. In [10], Guo, Xie and Zhao propose a Zienkiewicz-Zhu-type recovery scheme for Virtual Element Method (VEM), which is a new FEM-like method for the numerical solution of partial differential equations [11, 12]. The VEM belongs to the family of Galerkin methods based on polytopal grids and, as such, features a great flexibility with handling complex geometries and allows for easy adaptive refinement and coarsening. For instance, the VEM can be easily adapted for anisotropic discretisations [13, 14]. Despite its newness, it has already been applied to a wide range of problems, ranging from scattering problems governed by the Helmholtz equation [15] to structural mechanics [16] and plate bending problems [17] as well Stokes problems [18]. Besides the VEM, the discontinuous Galerkin method [19], hybridised discontinuous Galerkin method [20], the mimetic finite difference method [21, 22] and the Boundary Element Method based Finite Element Method (BEM-based FEM) [23] are prominent examples for numerical methods on polytopal grids. The latter has been introduced in [24] and has been studied, in particular, for adaptive FEM strategies involving residual [25, 26] and goal-oriented error estimators [27]. Other fields of application of the BEM-based FEM include, but are not limited to, FETI methods for large scale problems [28], convection dominated problems [29], anisotropic discretisations [30] and Nyström-based formulations [31].

In this paper, we focus on the VEM and BEM-based FEM. Both methods define the shape functions similarly as solutions of local boundary value problems on the elements of the mesh. The distinctive feature of the VEM is that these virtual element functions are not computed at all, but, instead, only their polynomial part is determined by suitable projection operators. Certainly, this is also the case for the virtual element solution, such that direct point-wise evaluations are not possible. In contrast, the BEM-based FEM actually computes the solution of the local boundary values problems by means of boundary integral equations [32]. Thus, the finite element approximation is known explicitly and can be evaluated anywhere in the domain.

In this work, we formulate gradient recovery schemes by averaging for the lowest order BEM-based FEM and VEM. In Section 3, we show that for the BEM-based FEM the centroids of regular k-gons are points of extraordinary accuracy, which we use to construct superconvergent recovered gradients in Section 4. Thereafter, we derive an a posteriori error estimator based on this recovery scheme in Section 5.

In the final Section 6, we see how the averaging technique needs to be modified to work with the VEM.

2. Preliminaries

Let $\Omega \subset \mathbb{R}^2$ be a bounded polygonal domain that admits a finite decomposition $\mathcal{T}_h$ into open non-overlapping polygonal elements E with maximal diameter $h > 0$ such that

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

The boundary ∂E of each element E is assumed to be not self-intersecting. Moreover, edges are always located between two nodes of the mesh and are either part of the boundary $\Gamma = \partial\Omega$ or are shared by elements. Besides, hanging nodes and non-convex elements are explicitly allowed in polygonal meshes. As usual, we require the meshes to fulfil some regularity assumptions in order to prove approximation properties later on. We call the mesh $\mathcal{T}_h$ shape-regular, if there exists some constant $\rho > 0$ such that

- every $E \in \mathcal{T}_h$ is star-shaped with respect to a ball of radius ρh_E, where h_E is the diameter of E, and
- for each $E \in \mathcal{T}_h$, the length h_e of every edge e of E satisfies $h_e > \rho h_E$.

Moreover, we denote by x_i, $i = 1, \dots, N$, the set of interior nodes of $\mathcal{T}_h$.
We use the common notation for function spaces encountered in finite element analysis. We denote by $L^2(\Omega)$ the space of square-integrable functions v equipped with the norm

$$\|v\|_{L^2(\Omega)} = \left(\int_\Omega |v|^2 \, dx \right)^{1/2}.$$

Moreover, we call a tuple $\underline{\alpha} = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$ multi-index with absolute value $|\underline{\alpha}| = \alpha_1 + \cdots + \alpha_d$. In this context, we denote multivariate monomials by

$$x^{\underline{\alpha}} = x_1^{\alpha_1} \dots x_d^{\alpha_d}$$

and by $\partial^{\underline{\alpha}}$ the differential operator

$$\partial^{\underline{\alpha}} = \left(\frac{\partial}{\partial x_1} \right)^{\alpha_1} \cdots \left(\frac{\partial}{\partial x_d} \right)^{\alpha_d}.$$

We consider for $k \in \mathbb{N}$ the Sobolev norm

$$\|v\|_{H^k(\Omega)} = \left(\sum_{|\underline{\alpha}| \le k} \|\partial^{\underline{\alpha}} v\|_{L^2(\Omega)}^2 \right)^{1/2},$$

and the corresponding Sobolev space

$$H^k(\Omega) = \left\{ v \in L^2(\Omega) \mid \|v\|_{H^k(\Omega)} < \infty \right\}.$$

In addition, we denote by

$$|v|_{H^k(\Omega)} = \left(\sum_{|\underline{\alpha}| = k} \|\partial^{\underline{\alpha}} v\|_{L^2(\Omega)}^2 \right)^{1/2}$$

the $H^k(\Omega)$-semi-norm.

For simplicity, we consider Poisson's equation with Dirichlet conditions $g \in L^2(\Gamma)$ and right-hand side $f \in L^2(\Omega)$, i.e.,

$$\begin{aligned} -\Delta u &= f, \quad \text{in } \Omega, \\ u &= g, \quad \text{on } \Gamma. \end{aligned} \tag{1}$$

It should be noted that the concepts presented here can be easily extended to more general partial differential equations and boundary conditions. The Galerkin formulation is posed in the space

$$V = \left\{ v \in H^1(\Omega) \mid v_{|\Gamma} = g \right\}.$$

and reads: find $u \in V$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, dx = \int_\Omega f v \, dx, \quad \forall v \in V, \tag{2}$$

where $\cdot$ denotes the euclidean inner product on $\mathbb{R}^2$. We abbriviate the left-hand side by $a(\cdot,\cdot)$. Due to the Lax–Milgram lemma, there exists a unique solution to (2). For later purposes, it is convenient to split the bilinear form $a(\cdot,\cdot)$ into the sum of its elemental contributions

$$a(u,v) = \sum_{E \in \mathcal{T}_h} a^E(u,v), \text{ where } a^E(u,v) = \int_E \nabla u \cdot \nabla v \, dx \,.$$

In the following, we introduce the shape functions of lowest order used by both methods. Let $\mathcal{P}_p(E)$ be the spaces of polynomials of degree p on the element E and

$$\mathcal{P}_p^{\mathrm{pw}}(\partial E) = \left\{ g \in C^0(\partial E) \mid g_{|e} \in \mathcal{P}_p(e) \text{ for every edge } e \text{ of } E \right\}$$

the space of piece-wise polynomials of degree p on ∂E. We define the local space of shape functions by

$$V_h^E = \left\{ v_h \in H^1(E) \mid \left\{ \begin{matrix} -\Delta v_h = 0 \quad \text{on } E, \\ (v_h)_{|e} \in \mathcal{P}_1^{\mathrm{pw}}(\partial E) \end{matrix} \right\} \right\}.$$

Thus, the shape functions are harmonic in the element E and piece-wise linear on its boundary ∂E. Moreover, we have $V_h^E \subset C(\overline{E})$. Consequently, the global finite element space is given by

$$V_h = \{ v_h \in V \mid (v_h)_{|E} \in V_h^E \ \forall E \in \mathcal{T}_h \} \cap C^0(\overline{\Omega})$$

and we choose the degrees of freedom as nodal values

$$\mathcal{N}_i(v_h) = v_h(x_i), \quad i = 1, \dots N,$$

to identify the shape functions in a unique way. The Galerkin approximation to (2) reads: find $u_h \in V_h$ such that

$$\int_\Omega \nabla u_h \cdot \nabla v_h \, dx = \int_\Omega f v_h \, dx, \quad \forall v_h \in V_h. \tag{3}$$

Here is where the two methods depart. The BEM-based FEM uses the theory of boundary integral operators to reformulate (3), whereas the VEM reduces the problem to the polynomial component $\mathcal{P}_1(E)$ of V_h^E. Nonetheless, in both cases, we end up with a discrete formulation of the form: find $u_h \in V_h$ such that

$$a_h(u_h, v_h) = b_h(v_h), \quad \forall v_h \in V_h, \tag{4}$$

where the discrete bilinear and linear form are constructed element-wisely

$$a_h(u_h, v_h) = \sum_{E \in \mathcal{T}_h} a_h^E(u_h, v_h), \quad b_h(v_h) = \sum_{E \in \mathcal{T}_h} b_h^E(v_h).$$

Firstly, we consider the discrete bilinear form of the BEM-based FEM. Since u_h is harmonic in each E, integration by parts yields

$$\int_E \nabla u_h \cdot \nabla v_h \, dx = -\int_E \Delta u_h v_h \, dx + \int_{\partial E} (\nabla u_h \cdot n) \, v_h \, dS = \int_{\partial E} (\nabla u_h \cdot n) \, v_h \, dS,$$

where n is the outer unit normal field to ∂E. The normal derivative $\nabla u_h \cdot n$ is obtained by means of the Steklov–Poincaré operator $\mathbf{S}^E$, see Example 1 for details. Therefore, the discrete bilinear form is given by

$$a_h^E(u_h, v_h) = \int_{\partial E} \left(\mathbf{S}^E (u_h)_{|\partial E} \right) v_h \, dS$$

and the right-hand side

$$b_h^E(v_h) = \int_E f v_h \, dS$$

is evaluated via numerical quadrature.
In comparison, the VEM introduces the H_0^1-orthogonal projection $\Pi_q^\nabla : V_h^E \to \mathcal{P}_q(E)$ for $q \in \mathbb{N}_0$ defined by the relation

$$\int_E \nabla m \cdot \nabla (\Pi_q^\nabla u_h) \, dx = \int_E \nabla m \cdot \nabla u_h \, dx, \quad \forall m \in \mathcal{P}_q(E),$$

and an additional constraint to fix the constant part [33].
After choosing a suitable stabilising term $S^E(\cdot, \cdot)$, see [34], we construct the discrete bilinear forms as follows:

$$a_h^E(u_h, v_h) = \int_E \nabla (\Pi_1^\nabla u_h) \cdot \nabla (\Pi_1^\nabla v_h) \, dx + S^E(u_h, v_h)$$

for $u_h, v_h \in V_h^E$. In essence, the discrete bilinear form only approximates the polynomial part of the left-hand side of (3) and the stabilising term guarantees the existence and uniqueness of the finite element solution. Furthermore, the right-hand side is treated similarly, i.e.,

$$b_h^E(v_h) = \int_E f \, \Pi_1^\nabla v_h \, dx \, .$$

It is important to note that the VEM is specifically designed in such a way that the projections are computable from just the degrees of freedoms [11].
Finally, we can rewrite the discrete formulation (4) as a system of linear equations by introducing the Lagrangian basis

$$\{\varphi_i\}_{i=1}^N \text{ with } \mathcal{N}_i(\varphi_j) = \begin{Bmatrix} 1, & i = j, \\ 0, & i \neq j, \end{Bmatrix} \quad \text{for } i, j = 1, \ldots, N.$$

Then, the ansatz

$$u_h = \sum_{i=1}^N c_i \varphi_i, \quad c_i \in \mathbb{R},$$

leads to

$$Ac = b,$$

where

$$A_{ij} = a_h(\varphi_i, \varphi_j), \quad b_i = b_h(\varphi_i), \quad i, j = 1, \dots, N.$$

This symmetric system is then solved efficiently with the conjugated gradient method to obtain the unknown coefficients c_i.

Remark 1. The BEM-based FEM and the VEM are not restricted to first order only, but admit natural generalisations with high order approximation spaces. For the BEM-based FEM and general polynomial degree $p \geq 2$, the respective local space of shape functions is defined by

$$V_h^E = \left\{ v_h \in H^1(E) \mid \begin{Bmatrix} -\Delta v_h \in \mathcal{P}_{p-2}(E), \\ (v_h)_{|\partial E} \in \mathcal{P}_p^{\mathrm{pw}}(\partial E) \end{Bmatrix} \right\}. \tag{5}$$

Here, we consider two types of basis functions: those that are harmonic inside E and piece-wise polynomial on ∂E and those that have polynomial Laplacian inside E and vanish on ∂E. Thus, we can specify the basis functions of V_h^E by choosing appropriate basis functions for the polynomial spaces $\mathcal{P}_{p-2}(E)$ and $\mathcal{P}_p^{\mathrm{pw}}(\partial E)$. In other words, the degrees of freedom of the finite element space are defined with respect to these polynomial trace spaces. As such, the shape functions are not known explicitly, but are accessible by means of higher-order BEM. In comparison, in the VEM only the polynomial part of finite element space is of interest and the degrees of freedom are chosen to be either point values or polynomial moments on elements and edges. In view of more general elliptic problems, the original space (5) is not applicable, since the L^2-orthogonal projection $\Pi_p : V_h^E \to \mathcal{P}_p(E)$ used in more general settings is not computable from the degrees of freedom of (5). Instead, one uses a slightly different space, where $-\Delta v_h \in \mathcal{P}_P(E)$ is allowed but under additional constraints on the higher polynomial moments, such that Π_p becomes computable. More details on the construction can be found in [35, 36].

Example 1. We consider as an example the harmonic BEM-based FEM of order p. The local space V_h^E simplifies to

$$V_h^E = \left\{ v_h \in H^1(E) \mid \begin{Bmatrix} -\Delta v_h = 0 \quad \text{on } E, \\ (v_h)_{|\partial E} \in \mathcal{P}_p^{\mathrm{pw}}(\partial E) \end{Bmatrix} \right\},$$

such that element bubble functions do not occur. The representation formula allows us to express each $v_h \in V_h^E$ by its Dirichlet trace $\gamma_0 v_h = (v_h)_{|\partial E}$ and Neumann trace $\gamma_1 v_h = (\nabla v_h \cdot n)_{|\partial E}$ with the help of the fundamental solution

$$u^*(y, x) = -\frac{1}{2\pi} \log(|y - x|), \quad x \neq y.$$

of the Laplacian. For $x \in E$ we obtain

$$v_h(x) = \int_{\partial E} \gamma_0 u^*(y, x)\, \gamma_1 v_h(y)\, dS(y) - \int_{\partial E} \gamma_{1,y} u^*(y, x)\, \gamma_0 v_h(y)\, dS(y).$$

By applying the trace operator γ_0 to the equation above, we obtain the following boundary integral equation for the unknown Neumann trace $\gamma_1 v_h$

$$\mathbf{V}^E \gamma_1 v_h = \frac{1}{2} \gamma_0 v_h + \mathbf{K}^E \gamma_0 v_h,$$

where the operators are defined by

$$\left(\mathbf{V}^E t\right)(x) = \int_{\partial E} \gamma_0 u^*(y,x)\, t(y)\, dS(y), \quad \left(\mathbf{K}^E g\right)(x) = \int_{\partial E} \gamma_{1,y} u^*(y,x)\, g(y)\, dS(y),$$

for $x \in \partial E$ and sufficiently smooth t and g. An equivalent representation is

$$\gamma_1 v_h = \mathbf{S}^E \gamma_0 v_h = \left(\mathbf{V}^E\right)^{-1} \left(\frac{1}{2}\mathbf{I}^E + \mathbf{K}^E\right) \gamma_0 v_h$$

with the Steklov–Poincaré operator $\mathbf{S}^E$. Details on the properties of the operators and solvability of the equation can be found in [37, 38]. For the numerical realisation, we search for an approximate solution $t_h \approx \gamma_1 v_h$ in

$$\mathcal{P}_{p-1}^{\mathrm{pw,d}}(\partial E) = \left\{ t \in L^2(E) \mid t_{|e} \in \mathcal{P}_{p-1}(e) \text{ for every edge } e \text{ of } E \right\},$$

which contains discontinuous functions compared to $\mathcal{P}_{p-1}^{\mathrm{pw}}(\partial E)$. Following [39], we select suitable bases for the trace spaces $\mathcal{P}_p^{\mathrm{pw}}(\partial E)$ and $\mathcal{P}_{p-1}^{\mathrm{pw,d}}(\partial E)$, where the former corresponds to a basis of the space of shape functions V_h^E. In this way, we obtain global basis functions $\varphi_i^{(p)}$, $i = 1, \ldots, N$, that are identified by their element-wise Dirichlet traces. Then, the approximate solution $u_h = \sum_{i=1}^N c_i\, \varphi_i^{(p)}$ of the underlying Laplace problem is found analogously to the first order case by solving the associated system of linear equations.

3. Superconvergent Points

Originally, the term "gradient superconvergence" describes the phenomenon that for certain types of elements there exist points at which the gradient of the finite element solution converges to the true solution at a higher rate than that encountered globally. Such points of extraordinary accuracy are known as stress points and were first discovered by Barlow [1]. Subsequently, Strang and Fix [2] proposed a strategy to locate these stress points. They argue that the leading term in the error of the finite element method corresponds to the problem of approximating polynomials of degree k by the finite element basis of degree $k-1$. To be more precise, the stress points can be identified by the property that the gradient of the true solution, which is now a polynomial of degree $k-1$, coincides with the gradient of its approximation, which is a piece-wise polynomial of degree $k-2$. For Poisson‘s equation in one dimension, it turns out that the Gauss-Legendre points on each element fulfil this condition and are superconvergent [7].

Turning to higher dimensions, the situation is much more complicated, since the existence and location of superconvergent points relies on the element geometry. In two dimensions, the stress points for bilinear basis functions on squares are simply the centroids and tensor products of Gauss-Legendre points for higher order basis functions, in accordance to the one-dimensional case [**MR3292660**, **MR533879**]. However, replacing squares by more general quadrilaterals or triangles already leads to completely different results. For instance, linear basis functions on triangles admit no stress points at all, while quadratics are superconvergent at the edge midpoints [2].

Thus, most of the theory is devoted to regular meshes which consist of identical elements, i.e., either squares or right-angled triangles of the same size. To the best of our knowledge, there are no results on superconvergent points for the FEM on polygonal elements.

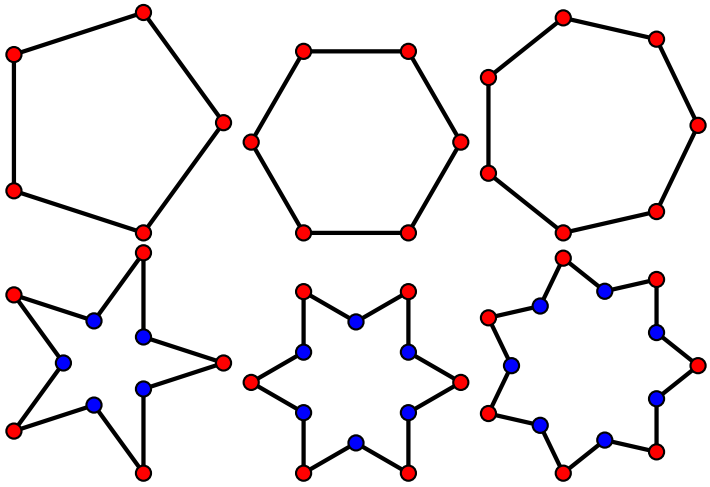

FIGURE 1. Regular convex and star k-gons having k and $2k$ vertices respectively with $k = 5, 6, 7$.

In the following, we extend the strategy by Strang and Fix to the BEM-based FEM on regular polygonal elements. To this end, let $\Omega \subset \mathbb{R}^2$ be a regular convex or star k-gon as depicted in Figure 1 and $p \geq 1$. We interpolate a harmonic polynomial $m_p \in \mathcal{P}_{p+1}(\Omega)$ with the shape functions $\varphi_i^{(p)}$ of order p from Example 1. Then, m_p is the unique solution to the boundary value problem

$$\begin{aligned} -\Delta u &= 0 \quad && \text{in } \Omega, \\ u &= m_p && \text{on } \Gamma = \partial\Omega. \end{aligned}$$

Since the $\varphi_i^{(p)}$ are piece-wise polyomials of degree p on Γ, the trace of m cannot be approximated exactly, so the interpolant $\tilde{m}_p$ is not exact. In fact, $\tilde{m}_p$ does not need to be a polynomial anymore. We are interested in the point-wise error between the gradients of m_p and $\tilde{m}_p$. It seems reasonable to choose the centroid x_c of Ω as a test point. Since $\tilde{m}_p$ is only given implicitly, we use the BEM to evaluate the interpolant via the representation formula.

In our example, Ω is always centered at the origin with diameter less than one and we choose

$$\begin{aligned} m_1 &= (x-1)^2 - (y+0.05)^2, \\ m_2 &= (x-0.1)^3 - 3(y-0.1)^2(x-0.1), \\ m_3 &= (x-6)^4 + (y+0.05)^4 - 6(x-6)^2(y+0.05)^2 \end{aligned}$$

as harmonic test polynomials, but it is worth to note that the qualitative results do not depend on the particular choice.

In Tables 1 and 2 the relative error

$$\frac{|\nabla m_p(x_c) - \nabla \tilde{m}_p(x_c)|}{|\nabla m_p(x_c)|}, \quad p = 1, 2, 3,$$

measured in the Euclidean norm at the centroid x_c of Ω is listed, where Ω is either a convex or a star k-gon. Above all, we see that superconvergence is clearly present in almost all cases. Indeed, the error is less than 10^{-9} on average, so we can certainly assume that the error induced by the BEM dominates. Furthermore, we observe that in the case of triangles and quadrilaterals the results presented here agree with those

k	$p=1$	$p=2$	$p=3$
3	$4.990 \cdot 10^{-2}$	$3.070 \cdot 10^{-8}$	$8.690 \cdot 10^{-8}$
4	$3.040 \cdot 10^{-10}$	$1.080 \cdot 10^{-3}$	$3.900 \cdot 10^{-11}$
5	$4.530 \cdot 10^{-11}$	$5.300 \cdot 10^{-11}$	$1.960 \cdot 10^{-8}$
6	$1.410 \cdot 10^{-11}$	$1.600 \cdot 10^{-11}$	$5.860 \cdot 10^{-12}$
7	$6.340 \cdot 10^{-12}$	$6.980 \cdot 10^{-12}$	$3.160 \cdot 10^{-12}$
8	$3.510 \cdot 10^{-12}$	$3.790 \cdot 10^{-12}$	$1.980 \cdot 10^{-12}$
9	$2.220 \cdot 10^{-12}$	$2.370 \cdot 10^{-12}$	$1.360 \cdot 10^{-12}$
10	$1.540 \cdot 10^{-12}$	$1.620 \cdot 10^{-12}$	$1.000 \cdot 10^{-12}$

TABLE 1. Relative error in the centroids of regular convex k-gons.

k	$p=1$	$p=2$	$p=3$
5	$9.310 \cdot 10^{-7}$	$1.130 \cdot 10^{-6}$	$6.710 \cdot 10^{-7}$
6	$9.980 \cdot 10^{-9}$	$6.330 \cdot 10^{-9}$	$1.660 \cdot 10^{-9}$
7	$8.530 \cdot 10^{-10}$	$2.750 \cdot 10^{-10}$	$6.730 \cdot 10^{-11}$
8	$1.740 \cdot 10^{-10}$	$2.170 \cdot 10^{-11}$	$1.620 \cdot 10^{-11}$
9	$5.600 \cdot 10^{-11}$	$7.290 \cdot 10^{-13}$	$7.470 \cdot 10^{-12}$
10	$2.370 \cdot 10^{-11}$	$2.750 \cdot 10^{-12}$	$4.260 \cdot 10^{-12}$
11	$1.190 \cdot 10^{-11}$	$2.350 \cdot 10^{-12}$	$2.720 \cdot 10^{-12}$
12	$6.730 \cdot 10^{-12}$	$1.780 \cdot 10^{-12}$	$1.830 \cdot 10^{-12}$
13	$4.150 \cdot 10^{-12}$	$1.330 \cdot 10^{-12}$	$1.310 \cdot 10^{-12}$
14	$2.720 \cdot 10^{-12}$	$1.010 \cdot 10^{-12}$	$9.730 \cdot 10^{-13}$

TABLE 2. Relative error in the centroids of regular star k-gons.

of the standard FEM [**MR3292660**]. This is not surprising for $p = 1$, because the first order shape functions of the BEM-based FEM coincide with linear or bilinear shape functions on triangles and quadrilaterals respectively.

4. Superconvergent Gradient Recovery

The primary use of superconvergence lies in the reconstruction of the gradient of the finite element solution ∇u_h in order to obtain a more accurate approximation of the true gradient ∇u. That is to say, we apply a post-processing technique to u_h that incorporates the superconvergent points to construct a recovered gradient $\mathcal{G}u_h$, with the intention that the latter is superconvergent not only at certain points but throughout subdomains or even the whole domain. Besides that the post-processing procedure has to preserve superconvergence, it should also be relatively fast in comparison to the actual finite element code, otherwise we could just use a finer mesh or higher order basis functions instead. Therefore, the reconstruction step should be limited to use only local information of the finite element solution.
To give an illustration, we consider the case of bilinear finite elements on a regular mesh made of identical squares of size h. Then, the finite element solution u_h is bilinear on each element E, while its gradient ∇u_h is piece-wise linear. The main idea is to reuse the bilinear basis for the recovered gradient $\mathcal{G}u_h$ in such a way that it interpolates the gradient ∇u_h at the superconvergent points. Since shape functions

are identified by their degrees of freedom, which are nodal values in this case, we need to specify those of $\mathcal{G}u_h$. To this end, we sample ∇u_h at the centroids of the elements and assign to each degree of freedom the average of these values on the corresponding patch, see Figure 2. As simple calculation shows that this is indeed the interpolant we are looking for. Moreover, one can prove that the recovered gradient is actually superconvergent, cf. [4, 7]. Also note that $\mathcal{G}u_h$, in contrast to ∇u_h, is continuous across edges. Thus, the reconstruction of the gradient can be interpreted as a form of gradient smoothing.

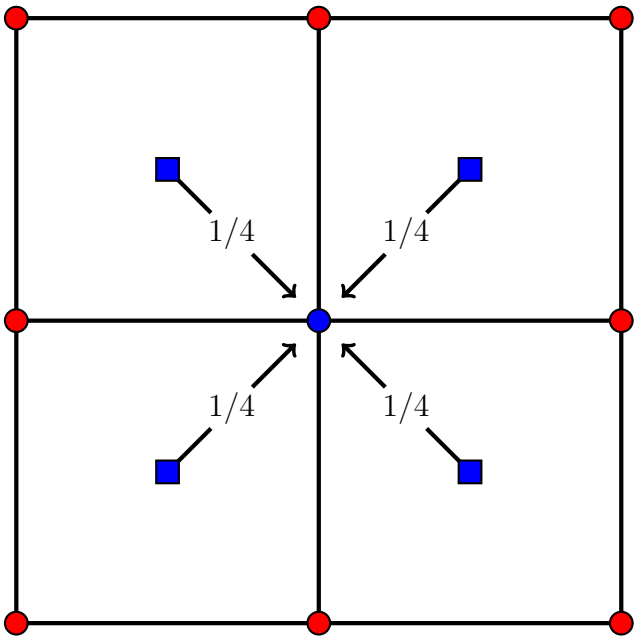

FIGURE 2. We assign to a node the average value of the gradient ∇u_h at the adjacent centroids.

With this in mind, we return to the BEM-based FEM and try to replicate the averaging technique for regular hexagonal meshes. We consider the recovery operator

$$\mathcal{G} : V_h \to V_h^2,$$

which maps a finite element function to its recovered gradient and is defined by the degrees of freedom in each component, i.e.,

$$\mathcal{N}_i\left(\mathcal{G}u_h\right) = \left(\mathcal{G}u_h\right)\left(x_i\right) \in \mathbb{R}^2, \quad i = 1, \ldots, N.$$

Denote by

$$E(x_i) = \left\{E \in \mathcal{T}_h \mid x_i \in \overline{E}\right\}$$

the patch of elements which share the node x_i and by $x_c(E)$ the centroid of the element E. We choose the degrees of freedoms of $\mathcal{G}u_h$ to be the average

$$\mathcal{N}_i\left(\mathcal{G}u_h\right) = \frac{1}{\#E(x_i)} \sum_{E \in E(x_i)} \nabla u_h\left(x_c(E)\right), \quad i = 1, \ldots N, \tag{6}$$

of the values of ∇u_h at the nearby centroids. As before, the recovery process is fairly simple and localised and practically does not increase numerical costs.

Moreover, we observe superconvergence of the recovered gradient in numerical experiments for the BEM-based FEM. To demonstrate this, we consider two Laplace problems with Dirichlet conditions on the square $\Omega = (-1,1)^2$,

$$\begin{aligned} -\Delta u &= 0 \quad \text{in } \Omega, \\ u &= g \quad \text{on } \Gamma = \partial\Omega, \end{aligned}$$

where g is either

$$\begin{aligned} g^{(1)} &= x^2 - y^2 + 4xy, \\ g^{(2)} &= \exp(2\pi(x-0.3))\,\cos(2\pi(y-0.3)). \end{aligned}$$

The exact solutions $u^{(1)}$ and $u^{(2)}$ are just the extensions of the Dirichlet data to the whole domain.

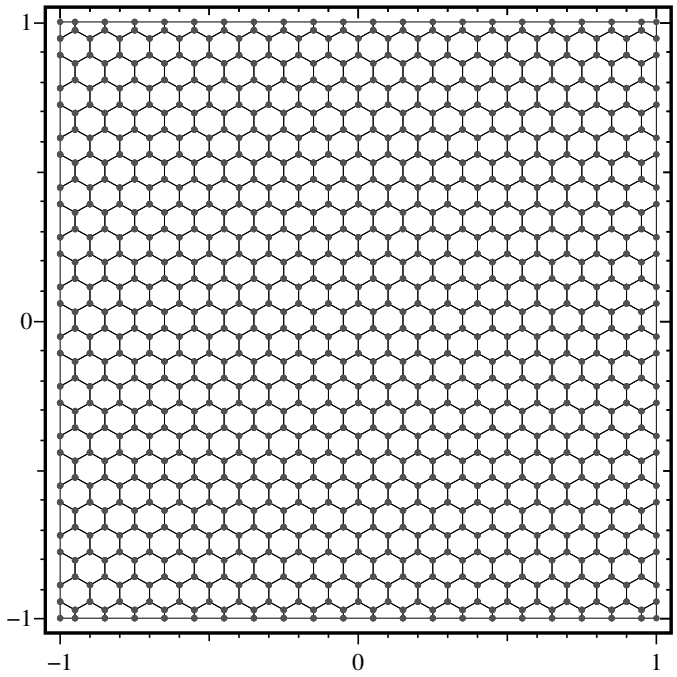

FIGURE 3. One of the meshes used in the example. Note that the elements at the boundary are clipped hexagons.

For both problems, we use meshes made of regular hexagons as depicted in Figure 3 and compare the error in the L^2-norm between the exact gradient $\nabla u^{(i)}$ and the approximate gradient $\nabla u_h^{(i)}$ as well as the recovered gradient $\mathcal{G}u_h^{(i)}$, i.e.,

$$\left\|\nabla u^{(i)} - \nabla u_h^{(i)}\right\|_{L^2(\Omega)} \text{ and } \left\|\nabla u^{(i)} - \mathcal{G}u_h^{(i)}\right\|_{L^2(\Omega)}, \quad i = 1, 2.$$

Since $u^{(1)}$ is a harmonic quadratic polynomial, the gradient sampling at the centroids is exact according to Section 3. The purpose of this first example is to check, whether the reconstruction $\mathcal{G}u_h^{(1)}$ from these values is exact throughout the domain. Thus, we interpolate the exact solution $u^{(1)}$ by the shape functions in V_h and apply the averaging technique (6) to the interpolation $u_h^{(1)}$. In Figure 4, the element-wise error between the reconstructed gradient and the exact gradient is visualised for the polynomial example. We see that the error is almost zero away from the

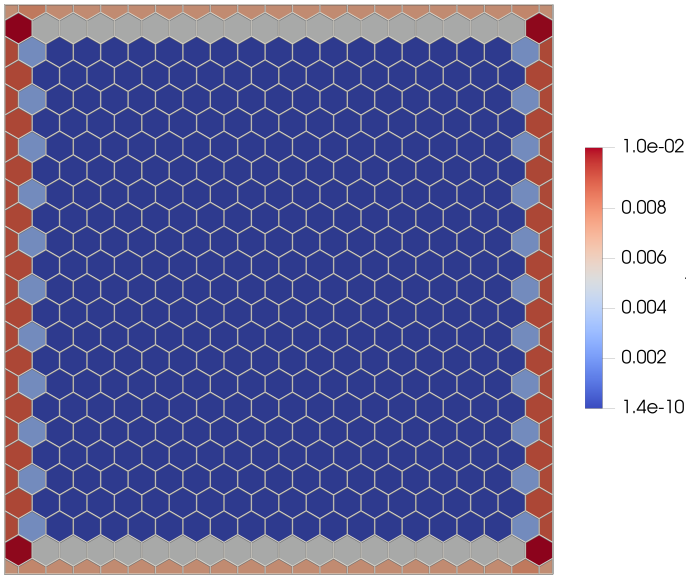

FIGURE 4. The element-wise error $\|\nabla u^{(1)} - \mathcal{G}u_h^{(1)}\|_{L^2(E)}$ is illustrated. We see that the reconstructed gradient coincides with the exact gradient in the interior of the square.

boundary, which hints that superconvergence is present in the interior of Ω. Hence, we conclude that the averaged gradient $\mathcal{G}u_h^{(1)}$ interpolates $\nabla u^{(1)}$ at the centroids x_c and is therefore exact, since $\nabla u^{(1)}$ is a linear polynomial in each component. Yet, the quality of the reconstruction at the boundary is poor, since a patch consists of only one or two elements there. Enlarging these patches by may be the solution to this problem [9].

Moving to the second example, we solve the actual boundary value problem on a sequence of hexagonal meshes with decreasing mesh size h and compute the reconstructed gradient like in (6) on each level. The results against the number of degrees of freedom N are depicted in Figure 5. In contrast to the previous example, we certainly cannot expect that the gradient is reconstructed exactly. Nevertheless, we do observe superconvergence, but now in form of an increase in the order of convergence, i.e., from $\mathcal{O}(N^{-1/2})$ for $\nabla u_h^{(2)}$ to approximately $\mathcal{O}(N^{-3/4})$ for $\mathcal{G}u_h^{(2)}$. In terms of the mesh size h, this translates to an increase from linear $\mathcal{O}(h)$ to superlinear convergence $\mathcal{O}(h^{3/2})$.

5. Recovery-based error estimator

Another signifcant application of gradient recovery lies in a posteriori error estimation with the aim of adaptive mesh refinement techniques. As usual, the goal is to find an estimate η^2 for the unknown error

$$|u - u_h|^2_{H^1(\Omega)} = \|\nabla u - \nabla u_h\|^2_{L^2(\Omega)}$$

measured in the energy norm. As the true gradient ∇u is not available, one common strategy is to replace it with a suitable approximation and use this as an estimate for the error. The choice of a recovered gradient $\mathcal{G}u_h$ obtained by post-processing of the

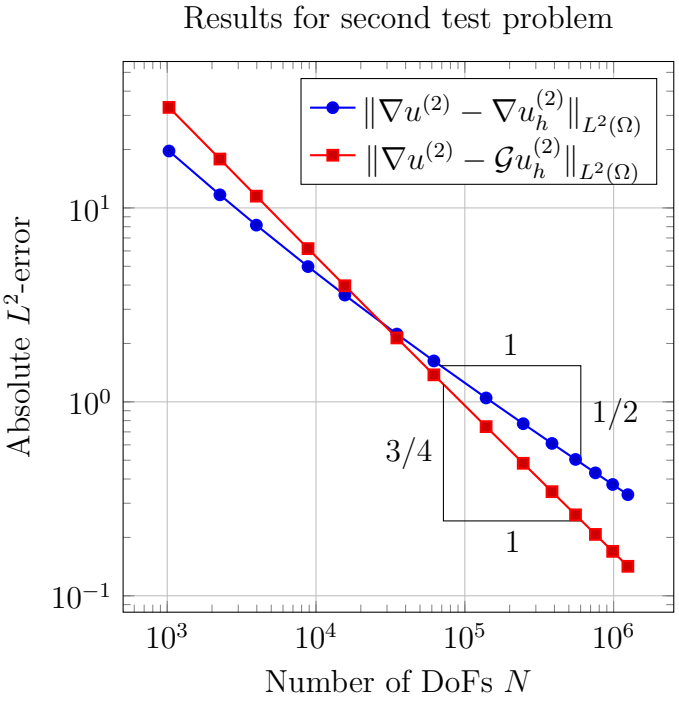

FIGURE 5. The error in the gradient between the exact solution and the approximate and recovered one respectively against the number of degrees of freedom in a loglog plot.

approximate gradient ∇u_h leads to the so called "recovery-based error estimator" [4]

$$\eta^2 = \|\mathcal{G} u_h - \nabla u_h\|^2_{L^2(\Omega)} .$$

The motivation behind such estimators is easy to comprehend: in most practical cases, the accuracy of the recovered gradient $\mathcal{G} u_h$ is superior to that of the untreated gradient ∇u_h and, furthermore, the post-processing is designed to be cheap and simple to implement. It is important to note that the post-processed gradient does not need to be superconvergent to produce a reliable estimator [40]. Therefore, we consider arbitrary polygonal meshes that are shape-regular from now on.
Thus, we have to modify our averaging scheme to take general meshes into account. We suggest to weight the elemental contributions according to the element's volume, so that the degree of freedom corresponding to the node x_i is given by

$$\mathcal{N}_i(\mathcal{G} u_h) = \frac{1}{|E(x_i)|} \sum_{E \in E(x_i)} |E| \, \nabla u_h(x_c(E)) , \quad i = 1, \dots, N. \tag{7}$$

In this way, we are consistent with the special case (6) and make sure that the larger the element, the more it contributes to the degrees of freedom in its nodes.

Algorithm 1. We follow the basic concept of an adaptive FEM algorithm of the form

(1) *Solve*: We solve problem (1) on the current mesh level and compute the gradient ∇u_h.
(2) *Recovery*: Then, we apply the post-processing by averaging (7) and obtain the recovered gradient $\mathcal{G} u_h$.

(3) *Estimate*: Subsequently, we estimate the error on each element E by

$$\eta_E^2 = \|\mathcal{G}u_h - \nabla u_h\|_{L^2(E)}^2$$

and form the sum

$$\eta^2 = \sum_E \eta_E^2.$$

(4) *Mark*: Afterwards, we mark the elements for refinement. Here, we apply the Dörfler marking strategy [41], i.e., we mark the subset $\mathcal{F} \subset \mathcal{T}_h$ of elements with largest estimated error such that

$$\left(\sum_{E\in\mathcal{F}} \eta_E^2\right)^{1/2} > \theta\,\eta$$

holds for a fixed parameter $\theta > 0$ under the condition that $\mathcal{F}$ is small as possible. We choose $\theta = 0.5$ for all experiments.

(5) *Refine*: Finally, we refine the marked elements and start again. Here, we use the bisection algorithm introduced in [25].

We stop the algorithm if the maximum mesh level is reached or η is sufficiently small.

In the following, we compare this recovery-based algorithm with a residual-based algorithm for the BEM-based FEM [26], which does not perform any post-processing but instead computes the residual

$$\eta_E^2 = \sum_{e\subset\partial E\setminus\Gamma} h_e \left\| \left[\!\left[\frac{\partial u_h}{\partial n} \right]\!\right] \right\|_{L^2(e)}^2, \tag{8}$$

in step (3) of Algorithm 1. The sum is taken over all edges e of the element E which are not part of boundary and the summand

$$\left[\!\left[\frac{\partial u_h}{\partial n} \right]\!\right] = (\nabla u_h)_{|E} \cdot n + (\nabla u_h)_{|(\Omega\setminus\overline{E})} \cdot n.$$

is defined as the jump of the normal derivative across the boundary of E. Note that n denotes the outer normal to E.

We consider the Laplace problem on the L-shaped domain $\Omega = (-1,1)^2 \setminus [0,1]^2$ with Dirichlet conditions

$$g(r,\varphi) = r^{2/3}\sin(2(\varphi - \pi/2)/3) \text{ on } \Gamma,$$

given in polar coordinates (r,φ). The solution u is simply the extension of g to Ω. This example serves as a popular test for adaptive algorithms, since $u \notin H^2(\Omega)$ and the rate of convergence of the error

$$|u - u_h|_{H^1(\Omega)} = \|\nabla u - \nabla u_h\|_{L^2(\Omega)}$$

is limited to $\mathcal{O}(N^{-1/3})$ for uniform refinement, i.e., refining every element in each step. In comparison, adaptive algorithms should recover the optimal convergence rate of $\mathcal{O}(N^{-1/2})$ despite the singularity.

We use the same initial Voronoi mesh, which is depicted in Figure 6, and observe intensive refinement around the origin for the residual- and recovery-based estimator. Hence, we conclude that the singularity is detected correctly regardless of the error estimator in use. Furthermore, we see in Figure 7 that both algorithms produce very similar results and succeed in achieving the convergence rate of $\mathcal{O}(N^{-1/2})$.

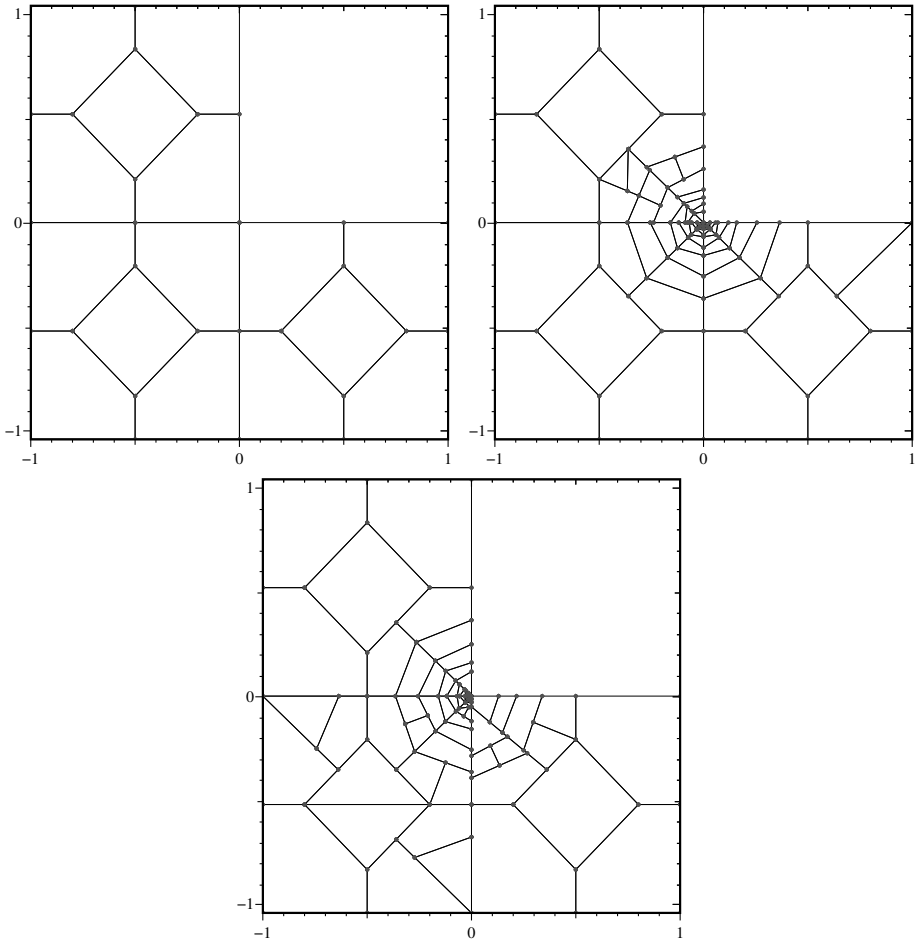

FIGURE 6. The initial Voronoi mesh (left) and after 15 refinement steps with the recovery-based estimator (right) and residual-based estimator (below).

In addition, we study the efficiency of the estimators given by

$$\Psi = \frac{\eta}{|u - u_h|_{H^1(\Omega)}},$$

which measures the quality of the estimated error. An efficiency equal to one indicates that the estimator reproduces the true error exactly, which is particularly useful in practice. The efficiency of our two estimators in this example is illustrated in Figure 8. Overall, we see that the recovery-based estimator operates nearly optimally and also outperforms the residual-based one. Taking the numerical work into consideration, we conclude that the recovery-based error estimator is our preferred choice for this particular example.

6. Gradient Recovery for VEM

In this final section, we present an analogue of the recovery-based estimator (7) for the VEM. Once more, we consider Poisson‘s problem (1) with exact solution $u \in V = H^1(\Omega)$ and virtual element solution $u_h \in V_h$.

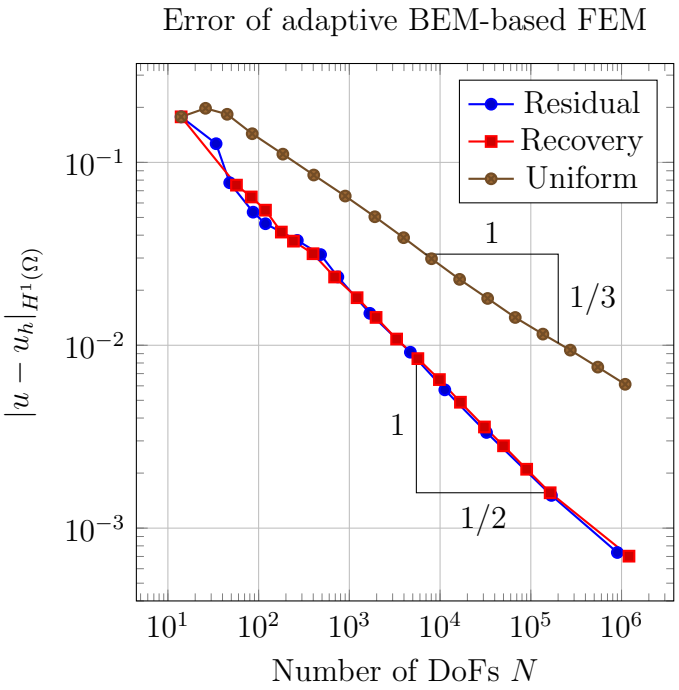

FIGURE 7. The error in the energy norm at every fourth step plotted against the number of degrees of freedom in logarithmic scaling. We see that both estimators produce equally good results.

Moving from the BEM-based FEM to the VEM, we encounter one difficulty: without the BEM framework, we lose the ability to evaluate the shape functions and its derivatives directly. As a consequence, the error in the H^1-norm is not accessible even if u is known and the recovery scheme fails since it involves function evaluations at the centroids. Therefore, we rely on the projection operator Π_1^∇ to guarantee computability and aim at estimating the error

$$\left|u - \Pi_1^\nabla u_h\right|_{H^1(\Omega)},$$

of the projection $\Pi_1^\nabla u_h$ instead of the actual error $|u - u_h|_{H^1(\Omega)}$.
Now, the derivation of the recovery process is straightforward. Since the projected gradient $\Pi_1^\nabla \nabla u_h$ is piece-wise constant, the post-processed gradient $\mathcal{G}u_h \in V_h^2$ is given by

$$\mathcal{N}_i(\mathcal{G}u_h) = \frac{1}{|E(x_i)|} \sum_{E \in E(x_i)} |E| \left(\nabla(\Pi_1^\nabla u_h)\right)(x_c(E)), \quad i = 1, \dots N. \tag{9}$$

In other words, we average the projection and interpolate with virtual element functions in each component afterwards, which yields a smoothed version of the gradient. Note that the averaging can also be interpreted as some form of L^2-projection on the patch.
Now the recovered gradient $\mathcal{G}u_h$ is a virtual element function in each component and we apply the projection Π_1^∇ to each component to retain computability. Thus, the recovery-based error estimator for the VEM has the representation

$$\eta^2 = \sum_E \eta_E^2 \quad \text{with } \eta_E^2 = \left\|\Pi_1^\nabla(\mathcal{G}u_h) - \nabla\left(\Pi_1^\nabla u_h\right)\right\|_{L^2(E)}^2.$$

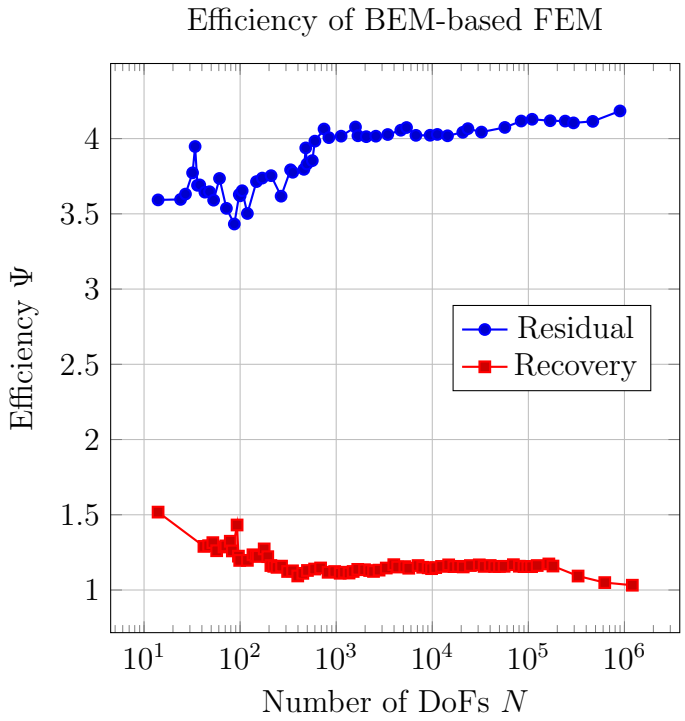

FIGURE 8. The efficiency of the two estimators against the number of degrees of freedom. While the recovery-based estimator yields almost optimal results close to 1, the residual-based one is overestimating the error about $3-4$ times its actual size.

Next, we check the performance of this estimator as part of the adaptive Algorithm 1. To this end, we use the same setup from the previous section for the adaptive test problem with the re-entrant corner. Likewise, we compare this estimator with the residual-based one [42] of the form

$$\eta_E^2 = \sum_{e \subset \partial E \setminus \Gamma} h_e \left\| \left[\!\left[\left(\nabla \left(\Pi_1^\nabla u_h \right) \right) \cdot n \right]\!\right] \right\|_{L^2(e)}^2 + S^E(u_h, u_h).$$

Based on Figure 9, we find that the error is of order $\mathcal{O}(N^{-1/2})$ for both strategies with the recovery-based one leading to marginally better results. In essence, the results for the VEM agree with the results for the BEM-based FEM.
Since we now estimate the projected error, the efficiency is given by

$$\Psi = \frac{\eta}{|u - \Pi_1^\nabla u_h|_{H^1(\Omega)}}.$$

Judging by Figure 10, the efficiency of the recovery-based estimator is again nearly optimal and superior to that of the residual-based approach.
Overall, both estimators perform very well in this particular example. Due to the fact that the recovery-based scheme achieves slightly better results and estimates the error more accurately, we prefer it over the residual-based estimator.

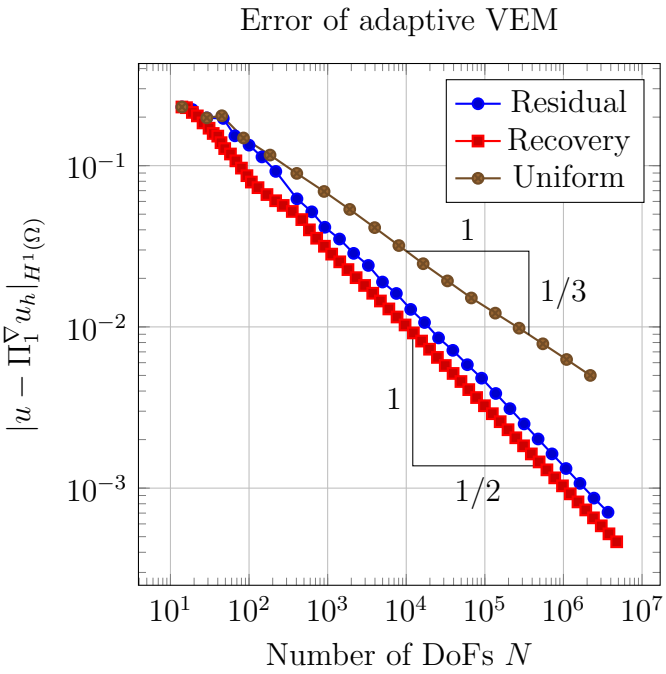

FIGURE 9. The error between the gradient of the exact solution and the projected gradient of the VEM solution plotted against the number of degrees of freedom in logarithmic scaling. We see that the recovery-based estimator produces slightly better results.

7. Conclusion

In this paper, we have studied different aspects of gradient recovery, namely superconvergent points, superconvergent gradient recovery and recovery-based error estimators. For the BEM-based FEM, we have found superconvergent points for regular k-gons and designed a post-processing by averaging which preserves superconvergence for the recovered gradient $\mathcal{G}u_h$. Notably, we observe that the rate of convergence improves from $\mathcal{O}(N^{-1/2})$ for the unprocessed gradient to $\mathcal{O}(N^{-3/4})$ for the recovered gradient. Furthermore, we have formulated recovery-based error estimators for the BEM-based FEM and VEM which are compatible with arbitrary polygonal meshes. These estimators feature a nearly optimal efficiency and good performance, so they present an attractive alternative to residual-based estimators.

References

[1] J. Barlow. "Optimal stress locations in finite element models". In: *Int. J. Numer. Meth. Eng.* 10.2 (1976), pp. 243–251.

[2] G. Strang and G. J. Fix. *An analysis of the finite element method*. Prentice-Hall Series in Automatic Computation. Prentice-Hall, Inc., Englewood Cliffs, N. J., 1973, pp. xiv+306.

[3] M. Zlámal. "Superconvergence and reduced integration in the finite element method". In: *Math. Comp.* 32.143 (1978), pp. 663–685.

[4] M. Ainsworth and J. T. Oden. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000, pp. xx+240.

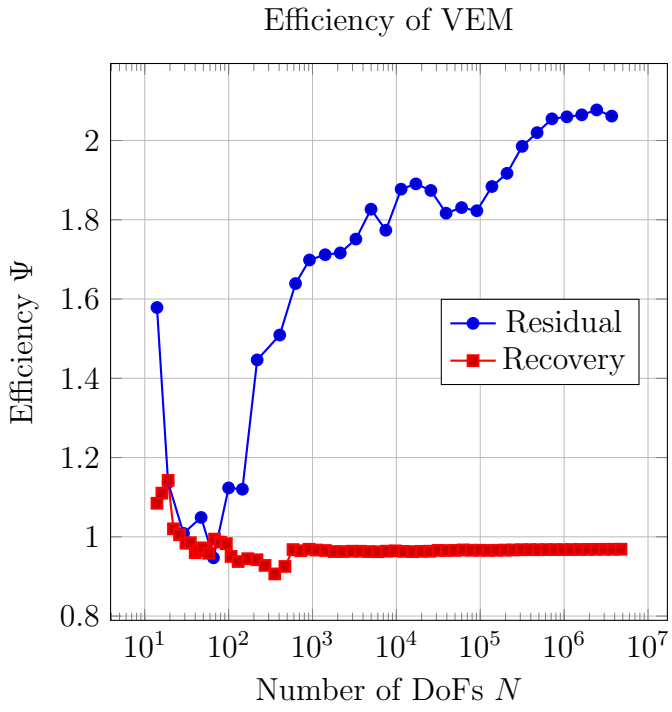

FIGURE 10. The efficiency of the two estimators against the number of degrees of freedom. We see that the recovery-based estimator yields good results close to 1, while the efficiency of the residual-based one varies from 0.8 to 2.2.

[5] O. C. Zienkiewicz and J. Z. Zhu. "A simple error estimator and adaptive procedure for practical engineering analysis". In: *Internat. J. Numer. Methods Engrg.* 24.2 (1987), pp. 337–357.

[6] M. Zlámal. *Some superconvergence results in the finite element method.* Vol. 606. Lecture Notes in Mathematics. Springer, Berlin, 1977, pp. 353–362.

[7] L. B. Wahlbin. *Superconvergence in Galerkin Finite Element Methods.* Vol. 1605. Lecture Notes in Mathematics. Springer, Berlin, 1995, pp. xii+166.

[8] J. T. Oden and H. J. Brauchli. "On the calculation of consistent stress distributions in finite element approximations". In: *Int. J. Numer. Meth. Eng.* 3.3 (1971), pp. 317–325.

[9] O. C. Zienkiewicz and J. Z. Zhu. "The superconvergent patch recovery and a posteriori error estimates. I. The recovery technique". In: *Internat. J. Numer. Methods Engrg.* 33.7 (1992), pp. 1331–1364.

[10] H. Guo, C. Xie, and R. Zhao. "Superconvergent gradient recovery for virtual element methods". In: *Math. Models Methods Appl. Sci.* 29.11 (2019), pp. 2007–2031.

[11] L. Beirão da Veiga et al. "Basic principles of virtual element methods". In: *Math. Models Methods Appl. Sci.* 23.1 (2013), pp. 199–214.

[12] L. Beirão da Veiga et al. "The hitchhiker's guide to the virtual element method". In: *Math. Models Methods Appl. Sci.* 24.8 (2014), pp. 1541–1573.

[13] P. F. Antonietti et al. "The Virtual Element Method on Anisotropic Polygonal Discretizations". In: *Numerical Mathematics and Advanced Applications* -

ENUMATH 2017. Ed. by F. A. Radu et al. Vol. 126. Lect. Notes Comput. Sci. Eng. Springer International Publishing, 2018.

[14] P. Antonietti et al. "Anisotropic a posteriori error estimate for the Virtual Element Method". In: *ArXiv* abs/2001.00381 (2020).

[15] L. Mascotto, I. Perugia, and A. Pichler. "A nonconforming Trefftz virtual element method for the Helmholtz problem: Numerical aspects". In: *Comput. Methods Appl. Mech. Engrg.* 347 (2019), pp. 445–476.

[16] L. Beirão da Veiga, C. Lovadina, and D. Mora. "A virtual element method for elastic and inelastic problems on polytope meshes". In: *Comput. Methods Appl. Mech. Engrg.* 295 (2015), pp. 327–346.

[17] F. Brezzi and L. D. Marini. "Virtual element methods for plate bending problems". In: *Comput. Methods Appl. Mech. Engrg.* 253 (2013), pp. 455–462.

[18] A. Cangiani, V. Gyrya, and G. Manzini. "The nonconforming virtual element method for the Stokes equations". In: *SIAM J. Numer. Anal.* 54.6 (2016), pp. 3411–3435.

[19] P. F. Antonietti et al. "Review of discontinuous Galerkin finite element methods for partial differential equations on complicated domains". In: *Building bridges: connections and challenges in modern approaches to numerical partial differential equations*. Vol. 114. Lect. Notes Comput. Sci. Eng. Springer, [Cham], 2016, pp. 279–308.

[20] B. Cockburn, J. Gopalakrishnan, and R. Lazarov. "Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems". In: *SIAM J. Numer. Anal.* 47.2 (2009), pp. 1319–1365.

[21] F. Brezzi, A. Buffa, and K. Lipnikov. "Mimetic finite differences for elliptic problems". In: *M2AN Math. Model. Numer. Anal.* 43.2 (2009), pp. 277–295.

[22] K. Lipnikov, G. Manzini, and M. Shashkov. "Mimetic finite difference method". In: *J. Comput. Phys.* 257.part B (2014), pp. 1163–1227.

[23] S. Rjasanow and S. Weißer. "Higher order BEM-based FEM on polygonal meshes". In: *SIAM J. Numer. Anal.* 50.5 (2012), pp. 2357–2378.

[24] 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*. Vol. 70. Lect. Notes Comput. Sci. Eng. Springer, Berlin, 2009, pp. 315–322.

[25] S. Weißer. "Residual error estimate for BEM-based FEM on polygonal meshes". In: *Numer. Math.* 118.4 (2011), pp. 765–788.

[26] S. Weißer. "Residual based Error Estimate and Quasi-Interpolation on Polygonal Meshes for High Order BEM-based FEM". In: *Comput. Math. Appl.* 73.2 (2017), pp. 187–202.

[27] S. Weißer and T. Wick. "The dual-weighted residual estimator realized on polygonal meshes". In: *Comput. Methods Appl. Math.* (2017). DOI: 10.1515/cmam-2017-0046.

[28] C. Hofreither, U. Langer, and C. Pechstein. "FETI solvers for non-standard finite element equations based on boundary integral operators". In: *Domain decomposition methods in science and engineering XXI*. Vol. 98. Lect. Notes Comput. Sci. Eng. Springer, Cham, 2014, pp. 729–737.

[29] C. Hofreither, U. Langer, and S. Weißer. "Convection-adapted BEM-based FEM". In: *ZAMM Z. Angew. Math. Mech.* 96.12 (2016), pp. 1467–1481.

[30] S. Weißer. “Anisotropic polygonal and polyhedral discretizations in finite element analysis”. In: *ESAIM Math. Model. Num.* (2018).
[31] A. Anand, J. S. Ovall, and S. Weißer. “A Nyström-based finite element method on polygonal elements”. In: *Comput. Math. Appl.* 75.11 (2018), pp. 3971–3986.
[32] O. Steinbach. *Numerical approximation methods for elliptic boundary value problems*. Finite and boundary elements, Translated from the 2003 German original. Springer, New York, 2008, pp. xii+386.
[33] B. Ahmad et al. “Equivalent projectors for virtual element methods”. In: *Comput. Math. Appl.* 66.3 (2013), pp. 376–391.
[34] L. Beirão da Veiga, C. Lovadina, and A. Russo. “Stability analysis for the virtual element method”. In: *Math. Models Methods Appl. Sci.* 27.13 (2017), pp. 2557–2594.
[35] A. Cangiani, G. Manzini, and O. J. Sutton. “Conforming and nonconforming virtual element methods for elliptic problems”. In: *IMA J. Numer. Anal.* 37.3 (2017), pp. 1317–1354.
[36] L. Beirão da Veiga et al. “Virtual element method for general second-order elliptic problems on polygonal meshes”. In: *Math. Models Methods Appl. Sci.* 26.4 (2016), pp. 729–750.
[37] M. Costabel. “Boundary integral operators on Lipschitz domains: elementary results”. In: *SIAM J. Math. Anal.* 19.3 (1988), pp. 613–626.
[38] S. Sauter and C. Schwab. *Boundary element methods*. Vol. 39. Springer Series in Computational Mathematics. Translated and expanded from the 2004 German original. Springer-Verlag, Berlin, 2011, pp. xviii+561.
[39] S. Weißer. “Arbitrary order Trefftz-like basis functions on polygonal meshes and realization in BEM-based FEM”. In: *Comput. Math. Appl.* 67.7 (2014), pp. 1390–1406.
[40] I. Babuška et al. “η%-superconvergence in the interior of locally refined meshes of quadrilaterals: superconvergence of the gradient in finite element solutions of Laplace’s and Poisson’s equations”. In: *Appl. Numer. Math.* 16.1-2 (1994), pp. 3–49.
[41] W. Dörfler. “A convergent adaptive algorithm for Poisson’s equation”. In: *SIAM J. Numer. Anal.* 33.3 (1996), pp. 1106–1124.
[42] A. Cangiani et al. “A posteriori error estimates for the virtual element method”. In: *Numer. Math.* 137.4 (2017), pp. 857–893.

Daniel Seibel, Department of Mathematics, Saarland University, 66041 Saarbrücken, Germany
Email address: seibel@num.uni-sb.de

Steffen Weisser, Department of Mathematics, Saarland University, 66041 Saarbrücken, Germany
Email address: weisser@num.uni-sb.de