



OPEN

A posteriori error approximation in discontinuous Galerkin method on polygonal meshes in elliptic problems


Jan Jaśkowiec  & Jerzy Pamin

The paper presents a posteriori error approximation concept based on residuals in the two-dimensional discontinuous Galerkin (DG) method. The approach is relatively simple and effective in application, and it takes advantage of some unique properties of the DG method. The error function is constructed in an enriched approximation space, utilizing the hierarchical nature of the basis functions. Among many versions of the DG method, the most popular one is based on the interior penalty approach. However, in this paper a DG method with finite difference (DGFD) is utilized, where the continuity of the approximate solution is enforced by finite difference conditions applied on the mesh skeleton. In the DG methods arbitrarily shaped finite elements can be used, so in this paper the meshes with polygonal finite elements are considered, including quadrilateral and triangular elements. Some benchmark examples are presented, in which Poisson's and linear elasticity problems are considered. The examples use various mesh densities and approximation orders to evaluate the errors. The error estimation maps, generated for the discussed tests, indicate a good correlation with the exact errors. In the last example, the error approximation concept is applied for an adaptive hp mesh refinement.

This paper presents an error approximation concept for a version of the discontinuous Galerkin (DG) method called the discontinuous Galerkin with finite difference (DGFD) method, introduced in the first author's previous papers¹⁻⁵. The main characteristic of the DGFD method is a stabilization parameter w , interpreted as a small distance from the mesh skeleton. In the DGFD method, like in any other version of the DG method, arbitrary global approximation can be constructed on the entire domain with arbitrary basis functions in elements. In this paper Chebyshev polynomials are employed as the approximation basis since their construction does not require nodes and, as such, they do not depend on the shape of finite elements. In consequence, arbitrary polygonal finite elements can be used in the mesh. The Chebyshev polynomials can be constructed in a recursive manner, which helps one to avoid significant truncation errors when constructing higher-order polynomials. These basis function properties are utilized for error approximation.

Error estimation is a common problem in all computational methods. It is used to assess the quality of the approximate solution. When it is applied for mesh refinement, error estimation makes it possible to obtain a better solution with minimum computational costs. In any automatic mesh adaptation, the procedure of error estimation is the most important component. Therefore, this paper is focused on error estimation techniques in the DGFD method to be applied in an automatic hp mesh refinement.

The DG method has a long history that goes back to the early seventies of the previous century⁶. A constant development has been observed since that time⁷⁻⁹. In general, the DG method can be treated as a particular version of the finite element method. In the DG method the domain is discretized using a finite element mesh, and the basis functions are constructed within the finite elements. The main difference between those two methods is the construction of the global approximation space. In the standard FEM, the global approximation space is continuous a-priori, while in the DG method the global approximation space is not continuous, so some techniques have to be employed to make the final approximate solution continuous. The lack of such continuity in the DG method seems to be its drawback, but on the other hand, it provides great flexibility in applying the basis functions and the shapes of the finite elements. It is natural for the DG methods to use polygonal/polyhedral finite elements. Moreover, the orders of approximation in the finite elements can be selected independently for each

Chair for Computational Engineering, Faculty of Civil Engineering, Cracow University of Technology, Warszawska 24, 31-155 Cracow, Poland.  email: jan.jaskowiec@pk.edu.pl

element. It enables an efficient *hp*-adaptation procedure where the finite element division or modification of the approximation order can be performed in a single element without any problem. In contrast, the *hp*-adaptation in the FEM affects the neighboring elements. In general, the DG method can be seen as a particular version of FEM, which gives new opportunities in the numerical analysis, but we have to deal with the discontinuities.

Since in the DG methods, unlike in the FEM, the approximation functions passing from one finite element to another one are not continuous, integration along the mesh skeleton (inter-element borders) is needed. Depending on the version of the DG method used, the continuity of the final solution is enforced by different conditions related to the mesh skeleton. In the interior penalty DG (IPDG) method, the most popular in literature^{10–12}, such continuity is obtained with the use of the Nitsche method¹³ which utilizes numerical fluxes applied to the skeleton and an additional penalty-like term, called stabilization parameter. In the local DG (LDG) methods^{14–16}, the Bassi-Rebay DG¹⁷ or the Baumann-Oden DG¹⁸, two numerical fluxes are used for the primary and secondary fields. This is due to the ultra-weak problem formulation, in which a second-order problem is written with two first-order equations. These numerical fluxes utilize additional parameters that must be evaluated. The ultra-weak formulation is also applied in the hybridized DG (HDG) method^{19,20}. Hybridization is the process of removing continuity constraints of finite element spaces without altering the solution. The number of unknowns in this method is strongly reduced by the hybridization. The HDG method has been applied to various engineering problems like second order elliptic problems²¹, incompressible Navier–Stokes equations²², space-time fractional advection-dispersion equations²³, or the flow in petroleum reservoirs²⁴. An alternative approach is adopted in the discontinuous Petrov-Galerkin (DGP) method^{25,26}, or the DG method with Lagrange multipliers (DGLM)²⁷, where the numerical fluxes are treated as additional unknowns. Extensive overviews of some of the DG methods are presented in.douglasp2002,Kirby2005.

Error estimation has been discussed in a number of scientific articles referring to various computational methods, for instance the finite element^{30–33} or meshless methods^{34,35}. Recently, the a posteriori error estimation and mesh adaptation in the DG methods have gained increasing attention. Some techniques for error estimation in hyperbolic or elliptic problems have also been proposed. The error can be estimated using a dual problem formulation³⁶, where orthogonal polynomials, defined on triangular elements³⁷, are used to evaluate the error in each element. The modified version of the DG method exhibits superconvergence properties that can be adopted for error estimation^{38,39}. Radau polynomials have been applied in a local steady state problem in each element to approximate the error⁴⁰. In another DG application an enriched polynomial basis is utilized in each tetrahedral element for error estimation⁴¹. A global estimator in the L_2 norm is proposed⁴² for elliptic problems. A residual-based error estimator and a material force concept have been used to estimate the error on quadtree meshes⁴³. A posteriori error estimation in terms of the energy norm has also been considered in the interior penalty DG method to solve an elliptic boundary-value problem⁴⁴. An eigenvalue problem has been analyzed⁴⁵ with the DG method, in the context of an auxiliary subspace technique applied for error estimation⁴⁶. Special attention should be paid to the works of M. Baccouch concerning error estimation in the DG methods, as they have made a significant contribution to computational analysis using the DG methods^{47–61}. His articles focus mainly on the LDG methods for one-dimensional or two-dimensional domains discretised with Cartesian grids. Another approach is proposed in⁶² where the error in the elements is based on the oscillations in the right-hand side function when projected to the approximation space.

In this paper an original technique to approximate the error in the DGFD method is proposed. In this method, the values of the trial function gradients on the mesh skeleton and on the Dirichlet boundary are approximated using finite difference formulas. In consequence, the non-zero values in the residuum functions have their source in the difference between the approximated values on the mesh skeleton and the ones obtained in the approximate solution. The error function can be constructed in the same function space as the approximate solution, but to obtain a better error approximation an enriched space is recommended. The approximation in the finite elements is based on the Chebyshev hierarchical polynomials. The element stiffness matrix also has a hierarchical structure. It enables the *p* upgrade of the approximation in the finite elements and in consequence the upgraded global stiffness matrix can be constructed and factorized in an effective way. This property is utilized in the error approximation solver, which can be obtained with a minimum effort.

The error evaluation method presented in this paper is illustrated by some benchmark examples, where the exact error function is compared with the approximate one. In these examples demanding problems are analyzed where the exact solutions have localized large gradients. Moreover, various types of meshes are considered with different approximation orders in the elements. Each time the error is approximated with a high accuracy. In the last example, an *hp* refinement is performed based on the approximated error.

This paper is focused on two-dimensional (2D) boundary value problems. However, in the authors' opinion, the proposed method of error approximation can be directly applied to three-dimensional (3D) problems with polyhedra meshes. In this paper, the error approximation concept is applied to two types of continuous elliptic problems: Poisson's problem and linear elasticity. As a direct continuation of the research, the methods of error approximation can be applied to other types of problems, where the exact solutions have discontinuous fields, for example, for domains with cracks or hyperbolic partial differential equations.

This paper is organized as follows. In “[Problem formulation](#)” Section two elliptic problems are defined and analyzed, namely the Poisson's and elasticity problem. Their weak forms are derived for the DGFD method in “[Weak formulations](#)” Section. Some questions concerning the approximation in the DGFD method are discussed in “[Approximation in DG method](#)” Section. The stability parameter plays a crucial role in the error approximation, so the discussion on its influence on the approximate solution is presented in “[Discussion on the stability parameter](#)” Section. “[Error estimation method](#)” Section presents the proposed method of error approximation, which is subsequently illustrated by four examples provided in “[Examples](#)” Section. The paper ends with brief conclusions.

Problem formulation

This paper utilizes two types of two-dimensional linear elliptic boundary value problems to present a new error approximation method. The problems are defined on domain Ω with outer boundary Γ . The first one is the scalar Poisson's problem with the following boundary conditions

$$\begin{aligned} -\Delta u &= f & \text{in } \Omega \\ u &= \hat{u} & \text{on } \Gamma^D, \quad \nabla u \cdot \mathbf{n} = \hat{h} & \text{on } \Gamma^N \end{aligned} \tag{1}$$

where Γ^D and Γ^N are the parts of the outer boundary with Dirichlet and Neumann boundary conditions, respectively, and $\Gamma^D \cup \Gamma^N = \Gamma, \Gamma^D \cap \Gamma^N = \emptyset$.

The second problem is the two-dimensional plane strain elasticity problem which can be expressed in the following way

$$\begin{aligned} \text{div } \boldsymbol{\sigma} + \mathbf{b} &= \mathbf{0} & \text{in } \Omega \\ \mathbf{u} &= \hat{\mathbf{u}} & \text{on } \Gamma^D, \quad \boldsymbol{\sigma} \cdot \mathbf{n} = \hat{\mathbf{t}} & \text{on } \Gamma^N \end{aligned} \tag{2}$$

where $\boldsymbol{\sigma}$ is the stress tensor, \mathbf{b} is the body force vector, $\hat{\mathbf{t}}$ is the prescribed traction force vector, \mathbf{u} is the displacement vector and $\hat{\mathbf{u}}$ is the prescribed displacement vector.

Equation (2) is supplemented by Hooke's law

$$\boldsymbol{\sigma} = \mathbf{E} : \boldsymbol{\varepsilon} \quad \text{in } \Omega \tag{3}$$

where $\boldsymbol{\varepsilon}$ is the strain tensor, \mathbf{E} is the fourth order Hooke's tensor, in which for the two-dimensional plane strain state the following non-zero terms are present

$$\begin{aligned} (\mathbf{E})_{1111} = (\mathbf{E})_{2222} &= \lambda + 2\mu, & (\mathbf{E})_{1122} = (\mathbf{E})_{2211} &= \lambda \\ (\mathbf{E})_{1212} = (\mathbf{E})_{1221} &= (\mathbf{E})_{2112} = (\mathbf{E})_{2121} &= \mu \end{aligned} \tag{4}$$

and λ, μ are standard elasticity constants.

Small strains are assumed in this paper, therefore the Cauchy strain tensor is used

$$\boldsymbol{\varepsilon} = \frac{1}{2} \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right] \quad \text{in } \Omega \tag{5}$$

In the DGF method the discontinuities of the global approximation across the mesh skeleton Γ^d need to be considered in the problem formulation. Local coordinates associated with the mesh skeleton are specified. They are based on two unit vectors (\mathbf{n}, \mathbf{s}) , where \mathbf{n} is the unit vector normal to Γ^d and \mathbf{s} is tangent to Γ^d . Fig. 1a presents the skeleton local coordinates, and Fig. 1b shows a graphical illustration of a stability parameter w . The parameter w is related only to the mesh, not to the considered problem. Therefore, different types of boundary problems can be solved using the same mesh with a similar value of the stability parameter⁶³. The numerical fluxes on the mesh skeleton are evaluated using finite difference formulas in the small neighborhood of the mesh skeleton which is defined by parameter w . In the DGF method, w is not only a penalty-like parameter. It indicates points in the vicinity of the mesh skeleton, see Fig. 1b, where the trial function or its derivatives should be calculated to evaluate the numerical fluxes. It has been shown in other papers, for example^{3-5,63}, that the DGF method is stable and not sensitive to fluctuations of w . A wider discussion of parameter w is presented in "Discussion on the stability parameter" Section.

In the discontinuous Galerkin method, the discontinuity and mean-value operators appear, which are defined with the help of the vector \mathbf{n} normal to the mesh skeleton. Their definitions for an auxiliary function g are as follows:

$$\begin{aligned} [[g]]_\epsilon(\mathbf{x}) &= g(\mathbf{x} + \epsilon \mathbf{n}^d) - g(\mathbf{x} - \epsilon \mathbf{n}^d) \\ [[g]](\mathbf{x}) &= \lim_{\epsilon \rightarrow 0} [[g]]_\epsilon(\mathbf{x}) \end{aligned} \quad \text{for } \mathbf{x} \in \Gamma^d \tag{6a}$$

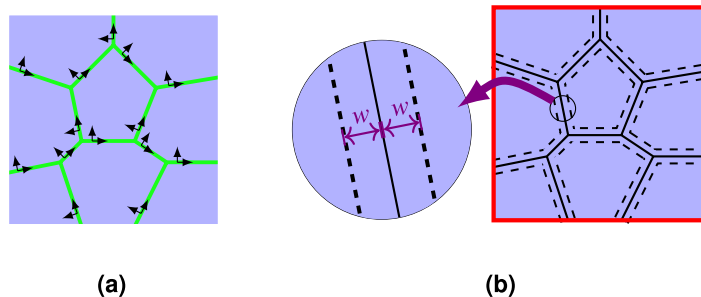


Figure 1. Mesh skeleton: local coordinates (a), interpretation of stability parameter w (b).

$$\begin{aligned} \langle g \rangle_\epsilon(\mathbf{x}) &= \frac{1}{2} (g(\mathbf{x} + \epsilon \mathbf{n}^d) + g(\mathbf{x} - \epsilon \mathbf{n}^d)) \\ \langle g \rangle(\mathbf{x}) &= \lim_{\epsilon \rightarrow 0} \langle g \rangle_\epsilon(\mathbf{x}) \end{aligned} \quad \text{for } \mathbf{x} \in \Gamma^d \tag{6b}$$

Weak formulations

Following the procedures given in^{2,5} the problems presented in Eqs. (1) and (2) can be rewritten in weak forms. In both cases of the weak forms the test and trial functions belong to the same broken Sobolev space S , defined as follows

$$S = H^1(\Omega, \Omega_h) = \left\{ g \in L^2(\Omega) : g|_{\Omega^e} \in H^1(\Omega^e) \forall \Omega^e \in \Omega_h \right\} \tag{7}$$

where Ω_h represents the set of disjoint finite elements covering the whole domain Ω , and Ω^e is a single finite element. In the space S a subset $\bar{S} \subset S$ can be distinguished that consists of functions which are continuous on the mesh skeleton and meet the boundary values on the Dirichlet boundary

$$\bar{S} = \left\{ g \in S : [[g]] = 0 \text{ on } \Gamma^d ; g = \hat{u} \text{ on } \Gamma^D \right\} \tag{8}$$

In this section the weak form of Poisson’s problem is first presented, and then, avoiding certain repetitions, the weak form of the elasticity problem is shown. The weak form of Poisson’s problem in Eq. (1) is expressed as follows: Find $u \in \bar{S}$ such that

$$b(v, u) = l(v) \quad \forall v \in S \tag{9}$$

where the bilinear and linear forms are defined as

$$b(v, u) = \int_{\Omega} \nabla v \cdot \nabla u \, d\Omega + \int_{\Gamma^d} [[v]] \mathbf{n} \cdot \nabla u \, d\Gamma - \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u \, d\Gamma \tag{10a}$$

$$l(v) = \int_{\Omega} v f \, d\Omega + \int_{\Gamma^N} v \hat{h} \, d\Gamma \tag{10b}$$

It is assumed in Eq. (9) that $\langle \nabla u \rangle \cdot \mathbf{n} = 0$ on Γ^d , which is a standard assumption in the computational methods such as FEM or other versions of the DG method. In this weak form, the test function belongs to the space of discontinuous functions, which is inconvenient when we want to find the approximate solution, since we want the trial function to belong to \bar{S} . Therefore, this formulation must be modified so that both the test and trial functions belong to the same space, however, the trial functions are forced to belong to set \bar{S} when the stability parameter w tends to zero. The modified weak form of Poisson’s problem is now expressed as follows: Find $u \in S$ such that:

$$b_w(v, u) = l(v) + l_w(v; \hat{u}) \quad \forall v \in S \quad \text{and for } w \rightarrow 0 \tag{11}$$

where

$$\begin{aligned} b_w(v, u) &= \int_{\Omega} \nabla v \cdot \nabla u \, d\Omega + \frac{3}{4} \int_{\Gamma^d} \frac{1}{w} [[v]][[u]]_w \, d\Gamma - \frac{1}{2} \int_{\Gamma^d} [[v]] \mathbf{n} \cdot \langle \nabla u \rangle_w \, d\Gamma \\ &+ 3 \int_{\Gamma^D} \frac{1}{w} v u(\mathbf{x}_{2w}) \, d\Gamma + 4 \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u(\mathbf{x}_w) \, d\Gamma + \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u(\mathbf{x}_{2w}) \, d\Gamma \end{aligned} \tag{12a}$$

$$l_w(v; \hat{u}) = \int_{\Gamma^D} \frac{3}{w} v \hat{u} \, d\Gamma \tag{12b}$$

where $\mathbf{x}_w = \mathbf{x} - w \mathbf{n}$, $\mathbf{x}_{2w} = \mathbf{x} - 2w \mathbf{n}$ and the value of the trial function at point \mathbf{x}_w on the Dirichlet boundary means the value of this function at distance w from the boundary in the normal direction: $u(\mathbf{x}_w) = u(\mathbf{x} - \mathbf{n}w)$ for $\mathbf{x} \in \Gamma^D$. Now, we present the lemma showing that the formulations in Eqs. (9) and (11) are equivalent.

Lemma 1 *If there is a function $u \in S$ that satisfies Eq. (11), then the function belongs to \bar{S} and satisfies Eq. (9).*

Proof Equation (11) has to be satisfied for every value of w , and, in particular for an arbitrarily small value of the parameter. This means that every finite difference relation in this equation has to tend to a specific derivative for w going to zero. Therefore, on the mesh skeleton we have

$$\frac{[[u]]_w}{2w} \xrightarrow{w \rightarrow 0} \mathbf{n} \cdot \nabla u \text{ on } \Gamma^d \Rightarrow [[u]] = 0 \text{ on } \Gamma^d \tag{13}$$

The finite difference on the mesh skeleton has to be finite for an arbitrarily small value of w , which indicates that the trial function has to be continuous on the mesh skeleton. On the Dirichlet boundary, we have the following finite difference relation

$$\frac{\hat{u} - u(\mathbf{x}_w)}{2w} \xrightarrow{w \rightarrow 0} \mathbf{n} \cdot \nabla u \text{ on } \Gamma^D \Rightarrow u = \hat{u} \text{ on } \Gamma^D \tag{14}$$

In this case the finite difference on the Dirichlet boundary is also finite for every value of w , which indicates that the trial function has to meet the boundary values on the Dirichlet boundary. Additionally, it is obvious that the mean values of the derivative on the mesh skeleton tend to the derivative on the mesh skeleton, and a similar situation occurs with the derivative on the Dirichlet boundary

$$\begin{aligned} \mathbf{n} \cdot \langle \nabla u \rangle_w &\xrightarrow{w \rightarrow 0} \mathbf{n} \cdot \nabla u \text{ on } \Gamma^d \\ \mathbf{n} \cdot \nabla u(\mathbf{x}_w) &\xrightarrow{w \rightarrow 0} \mathbf{n} \cdot \nabla u \text{ on } \Gamma^D \end{aligned} \tag{15}$$

Equation (11) is now rewritten with some small modifications

$$\begin{aligned} \int_{\Omega} \nabla v \cdot \nabla u \, d\Omega + \frac{3}{2} \int_{\Gamma^d} \frac{[[u]]_w}{2w} \, d\Gamma - \frac{1}{2} \int_{\Gamma^d} [[v]] \mathbf{n} \cdot \langle \nabla u \rangle_w \, d\Gamma \\ - 6 \int_{\Gamma^D} v \frac{\hat{u} - u(\mathbf{x}_{2w})}{2w} \, d\Gamma + 4 \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u(\mathbf{x}_w) \, d\Gamma + \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u(\mathbf{x}_{2w}) \, d\Gamma = \int_{\Omega} v f \, d\Omega + \int_{\Gamma^N} v \hat{h} \, d\Gamma \end{aligned} \tag{16}$$

When w goes to zero, then the finite difference relations in the second and fourth integrals in Eq. (16) tend to the derivatives as shown in Eqs. (13) and (14), and, using relations (15), this results in the following equation

$$\begin{aligned} \int_{\Omega} \nabla v \cdot \nabla u \, d\Omega + \frac{3}{2} \int_{\Gamma^d} [[v]] \mathbf{n} \cdot \nabla u \, d\Gamma - \frac{1}{2} \int_{\Gamma^d} [[v]] \mathbf{n} \cdot \nabla u \, d\Gamma \\ - 6 \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u \, d\Gamma + 4 \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u \, d\Gamma + \int_{\Gamma^D} v \mathbf{n} \cdot \nabla u \, d\Gamma = \int_{\Omega} v f \, d\Omega + \int_{\Gamma^N} v \hat{h} \, d\Gamma \end{aligned} \tag{17}$$

in which $u \in \bar{S}$. After a few easy algebraic operations in Eq. (17), the Eq. (9) is retrieved. □

To obtain a numerical solution, Eq. (11) has to be rewritten in a discrete version, in which the infinite functional space S is substituted by the finite space $S^p \subset S$. The subspace S^p comprises a complete set of polynomial basis functions in each element up to order p . In computations parameter w cannot be arbitrarily small, the parameter is just small in the numerical sense. The Poisson's problem shown in Eq. (11) is expressed in the discrete form as follows: Find $u^p \in S^p$ in Ω such that

$$b_w(v^p, u^p) = l(v^p) + l_w(v^p; \hat{u}) \quad \forall v^p \in S^p \text{ and for small value of } w \tag{18}$$

The boundary value problem of elasticity in Eq. (2) also has its discrete form obtained in the same way as for the Poisson's problem. It was derived in^{4,5} and reads Find $\mathbf{u}^p \in S^p$ in Ω such that

$$b_w^v(\mathbf{v}^p, \mathbf{u}^p) = l^v(\mathbf{v}^p) + l_w^v(\mathbf{v}^p; \hat{\mathbf{u}}) \quad \forall \mathbf{v}^p \in S^p \text{ and for small value of } w \tag{19}$$

The elasticity problem is a vector problem, so the definitions of the bilinear and linear forms are more complex:

$$\begin{aligned} b_w^v(\mathbf{v}, \mathbf{u}) = \int_{\Gamma^D} \mathbf{v} \cdot \mathbf{E}_1^b \cdot \mathbf{u} \, d\Gamma - \int_{\Gamma^D} \mathbf{v} \cdot \mathbf{E}_2^b : \nabla \mathbf{u} \, d\Gamma + \int_{\Gamma^D} \mathbf{v} \cdot \mathbf{E}_3^b : \nabla \mathbf{u}(\mathbf{x}_w) \, d\Gamma \\ + \int_{\Gamma^D} \mathbf{v} \cdot \mathbf{E}_4^b : \nabla \mathbf{u}(\mathbf{x}_{2w}) \, d\Gamma \\ + \int_{\Gamma^d} [[\mathbf{v}]] \cdot \mathbf{E}_1^d \cdot [[\mathbf{u}]]_w \, d\Gamma - \int_{\Gamma^d} [[\mathbf{v}]] \cdot \mathbf{E}_2^d : \langle \nabla \mathbf{u} \rangle_w \, d\Gamma \\ + \int_{\Gamma^d} [[\mathbf{v}]] \cdot \mathbf{E}_3^d : \langle \nabla \mathbf{u} \rangle_w \, d\Gamma + \int_{\Omega} \nabla \mathbf{v} : \mathbf{E} : \nabla \mathbf{u} \, d\Omega \end{aligned} \tag{20}$$

$$l^v(\mathbf{v}) = \int_{\Omega} \mathbf{v} \cdot \mathbf{b} \, d\Omega + \int_{\Gamma^N} \mathbf{v} \cdot \hat{\mathbf{t}} \, d\Gamma, \quad l_w^v(\mathbf{v}; \hat{\mathbf{u}}) = \int_{\Gamma^D} \mathbf{v} \cdot \mathbf{E}_1^b \cdot \hat{\mathbf{u}} \, d\Gamma \tag{21}$$

The tensors \mathbf{E}_i^d and \mathbf{E}_i^b , related to the mesh skeleton and outer boundary, respectively, are defined as follows

$$\mathbf{E}_1^d = \frac{3}{4w} \left((\lambda + 2\mu) \mathbf{I} - \lambda \mathbf{s} \otimes \mathbf{s} \right) \tag{22a}$$

$$\mathbf{E}_2^d = \frac{1}{2} \left((\lambda + 2\mu) (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + 2\mu (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) \right) \tag{22b}$$

$$\mathbf{E}_3^d = \lambda (\mathbf{n} \otimes \mathbf{I} - \mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + 2\mu (\mathbf{s} \otimes \mathbf{n} \otimes \mathbf{s}) \tag{22c}$$

$$\mathbf{E}_1^b = \frac{3}{w} \left((\lambda + 2\mu) \mathbf{I} - \lambda \mathbf{s} \otimes \mathbf{s} \right) \tag{22d}$$

$$\mathbf{E}_2^b = \lambda (\mathbf{n} \otimes \mathbf{s} \otimes \mathbf{s}) + 2\mu (\mathbf{s} \otimes \mathbf{n} \otimes \mathbf{s}) \tag{22e}$$

$$\mathbf{E}_3^b = 4 \left((\lambda + 2\mu) (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + 2\mu (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) \right) \tag{22f}$$

$$\mathbf{E}_4^b = (\lambda + 2\mu) (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + 2\mu (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) \tag{22g}$$

Approximation in DG method

In the DG methods, the basis functions in the elements can be chosen as any sort of polynomial basis functions. In particular, they can just be monomials, Chebyshev, Legendre, or any other sort of polynomials, provided that they are complete up to order p . In certain situations, the basis functions can be enriched by another sort of functions, see³. In this paper, the approximation basis is chosen to be the Chebyshev polynomials, which are constructed using the recursive form as follows⁶⁴

$$T_0(\xi) = 1, \quad T_1(\xi) = \xi, \quad T_{n+1}(\xi) = 2 \cdot \xi \cdot T_n(\xi) - T_{n-1}(\xi) \tag{23}$$

Orthogonality is a well-known property of the Chebyshev polynomials. However, they are not purely orthogonal, but orthogonal with the weight $\frac{1}{\sqrt{1-\xi^2}}$ in the interval $[-1, 1]$. The Chebyshev polynomials are more convenient for approximation due to their recursive definition, making it possible to calculate very high order polynomials without significant truncation errors. The basis functions are constructed in each element using the local coordinates starting at the center of the element

$$x^e = \frac{x - x_m^e}{0.5 h_x^e}, \quad y^e = \frac{y - y_m^e}{0.5 h_y^e} \tag{24}$$

where the point (x_m^e, y_m^e) is the centre of gravity of the e -th element cell and h_x^e, h_y^e are the characteristic dimensions of the e -th cell in the x and y directions, respectively. The 2D basis functions are derived from the cross product of 1D polynomials:

$$\begin{aligned} p = 1 : \quad \mathbf{b}^e &= \mathbf{b}_1^e = [1 \quad T_1(x^e) \quad T_1(y^e)] \\ p = 2 : \quad \mathbf{b}^e &= \mathbf{b}_2^e = [\mathbf{b}_1^e \quad T_1(x^e)T_1(y^e) \quad T_2(x^e) \quad T_2(y^e)] \\ p = 3 : \quad \mathbf{b}^e &= \mathbf{b}_3^e = [\mathbf{b}_2^e \quad T_3(x^e) \quad T_2(x^e)T_1(y^e) \quad T_1(x^e)T_2(y^e) \quad T_3(y^e)] \end{aligned} \tag{25}$$

and so on. The global approximation, for the entire domain Ω , is constructed with the help of the global approximation matrix Ψ^p and the vector of degrees of freedom \mathbf{h} . The suffix p in the symbol of the approximation matrix means that the polynomials of order up to p are used in the elements to compute the matrix. For example, the approximations for the scalar field u and the vector field \mathbf{u} are as follows

$$u^p = \Psi^p \mathbf{h} \quad \text{in } \Omega \quad \text{or} \quad \mathbf{u}^p = \Psi^p \mathbf{h} \quad \text{in } \Omega \tag{26}$$

The matrix Ψ^p is a one-row matrix for the scalar function u in the Poisson's problem given in Eq. (1) and two-row matrix for the 2D elasticity problem in Eq. (2).

In the standard finite element method, the approximation is constructed with the help of the so-called shape functions, whose definitions depend on the shape of the finite element. In the DG methods, the basis functions do not depend on the shapes of the elements. The element matrices in the DG methods have the hierarchic structure as illustrated in Fig. 2. This means that the stiffness matrix of order p keeps inside all the stiffness matrices of the lower orders. On the other hand, when we have the stiffness matrix of order p , it is relatively easy to extend it to the matrix of order $p + 1$.

When the approximation in Eq. (26) is applied to the weak formulation of the problem in Eq. (11) or (18), we obtain the discrete problem in an algebraic form

$$\mathbf{K}^p \mathbf{u} = \mathbf{f}^p \tag{27}$$

where \mathbf{K}^p and \mathbf{f}^p are defined depending on the problem under consideration

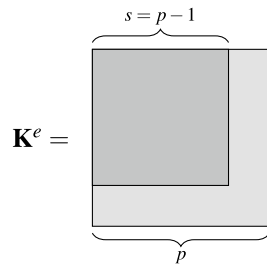


Figure 2. Hierarchical structure of element stiffness matrix of order p .

$$\mathbf{K}^p = b_w(\Psi^p, \Psi^p), \quad \mathbf{f}^p = l(\Psi^p) + l_w(\Psi^p; \hat{\mathbf{u}}) \quad (28a)$$

$$\mathbf{K}^p = b_w^v(\Psi^p, \Psi^p), \quad \mathbf{f}^p = l^v(\Psi^p) + l_w^v(\Psi^p; \hat{\mathbf{u}}) \quad (28b)$$

The DGF method has been programmed in the Matlab environment which offers great flexibility and many up-to-date mathematical tools. Particularly, the subroutines for linear solvers are available in this environment. In this paper, the 'linsolve' function has been applied for solving algebraic equations. The main problem in Eq. (27) is solved in the standard way.

Discussion on the stability parameter

In Eqs. (18) and (19) the scalar stabilisation parameter w plays an important role. It should cause the final discrete solution to be continuous and the Dirichlet boundary conditions to be met. The value of the parameter should be small enough for the compatibility conditions to be satisfied, but not too small to avoid numerical instabilities.

The w parameter measures the small distance from the mesh skeleton, so its value depends on the size of the adjacent elements. On the other hand, its value is independent of the considered problem, but has to be adjusted to the mesh, so it is possible to use a single stabilization parameter for coupled problems⁶³. The parameter cannot be arbitrarily small since numerical instabilities can occur if it is too small. In consequence, the solutions obtained this way are not strictly continuous and the Dirichlet boundary conditions are not exactly met. In other words, the value of the parameter should be chosen small enough to obtain good quality solutions, but not too small to avoid truncation errors that can spoil the solution.

The w parameter should be small, however, it turns out that the final solution is not sensitive to its value. This means that a relatively large change in the value causes tiny changes in the approximate solution. This indicates that the DGF method is consistent and well-defined. In the DGF method, elements of various sizes and orders can be freely combined in the mesh, so the method is especially eligible for mesh adaptation. The stabilization parameter depends only on the mesh, i.e. its value depends only on the size of elements that adhere to the skeleton segment, $w = \gamma \cdot h$, where h indicates the element size and γ is a scaling parameter that is constant for the whole mesh skeleton.

In order to illustrate the influence of the stabilization parameter on the final solution in the DGF method, the Poisson's problem, see Eq. (1), is solved in the square domain $\Omega = [-1, 1] \times [-1, 1]$. The right-hand side function and the boundary conditions are taken from the exact solution, which is assumed to be the following exponential function

$$u(x, y) = \exp(-\alpha \cdot (x^2 + (y - \beta x)^2)) \quad (29)$$

with the parameters $\alpha = 10$ and $\beta = 2$. The function with its both derivatives is depicted in Fig. 3.

The problem has been solved with various values of scaling parameter γ for quadrilateral and polygonal meshes. The meshes and maps of errors are shown in Figs. 4 and 5, respectively. In the quadrilateral mesh, the largest element is over one thousand times larger than the smallest element, and the two elements are adjacent to each other. The polynomial orders in the two meshes have been randomly generated: $p \in [3, 20]$ for the quadrilateral mesh and $p \in [3, 8]$ for the polygonal mesh. It can be seen in the two examples that even though the value of the stabilization parameter is changed by five orders of magnitude, only slight differences in the errors can be observed. Thus, the stability parameter in the DGF method can be used to combine finite elements of different sizes as well as different orders.

Error estimation method

In the proposed method of error approximation, the original boundary value problem is considered in a residuum form. The weak form of an error function is constructed, which is used to obtain an approximate error function. For the sake of clarity, the method is first presented for Poisson's problem and then extended to the elasticity problem. The derivation begins from defining an exact error function as the difference between the exact solution u and its approximation obtained by solving Eq. (18)

$$e = u - u^p \quad (30)$$

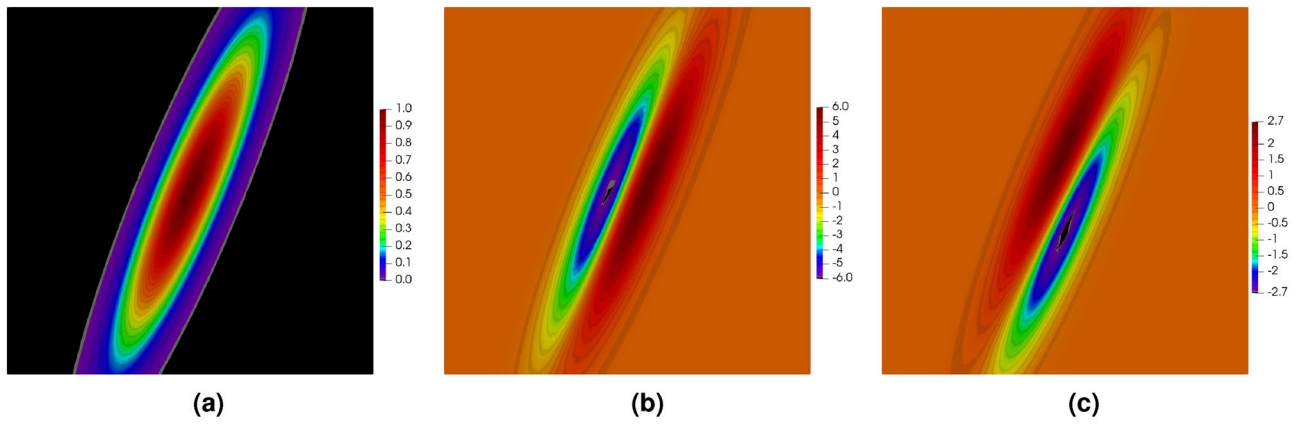


Figure 3. Function defined in Eq. (29) (a) and its derivatives in x direction (b) and y direction (c).

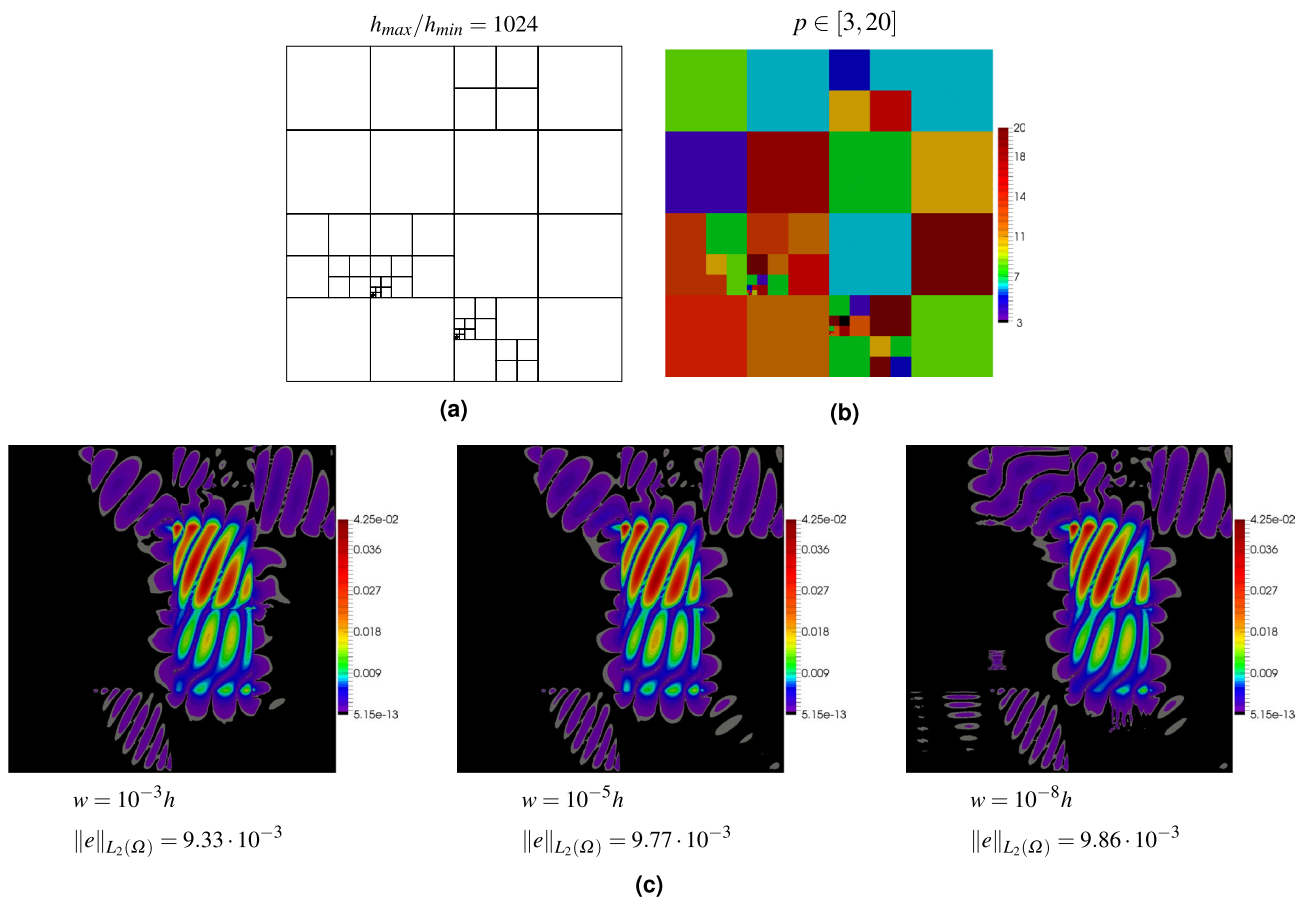


Figure 4. Numerical test of stability parameter w on the mesh with square elements: mesh structure (a), orders of elements (b), maps of errors for various values of w (c).

The boundary value problem for the error function is constructed by applying the Laplace operator to both sides of Eq. (30) and substituting the right-hand side from Eq. (1)

$$-\Delta e = f - (-\Delta u^p) \quad \text{in } \Omega \tag{31}$$

Equation (31) must be completed by boundary conditions, which are as follows

$$e = \hat{e} = \hat{u} - u^p \quad \text{on } \Gamma^D, \quad \nabla e \cdot \mathbf{n} = \hat{h} - \nabla u^p \cdot \mathbf{n} \quad \text{on } \Gamma^N \tag{32}$$

It should be noted that \hat{e} is not zero because in the DGF method the Dirichlet boundary conditions are not strictly met. Thus, there are small differences between the prescribed values and the approximate solution on the

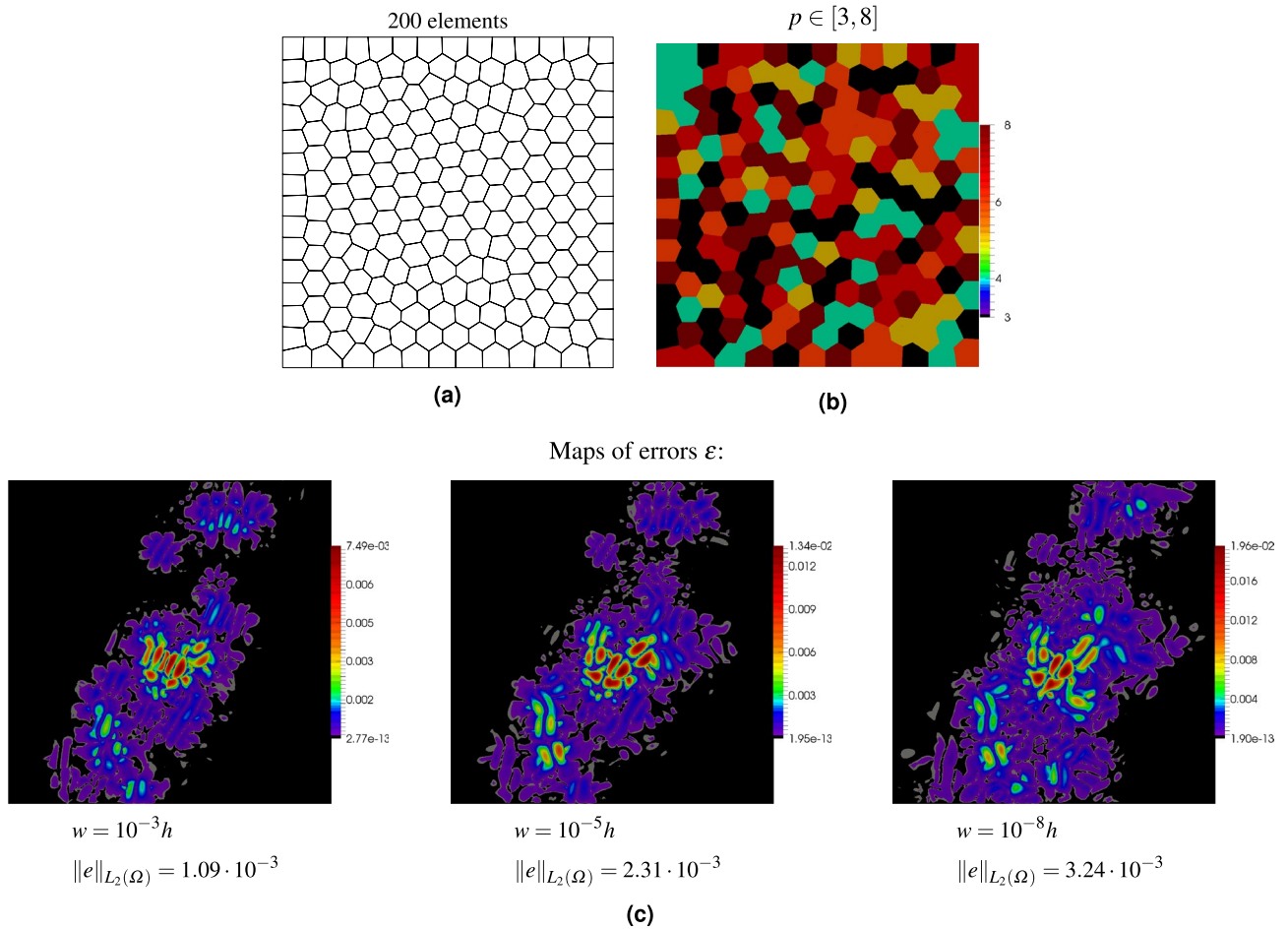


Figure 5. Numerical test of stability parameter w on the mesh with polygonal elements: mesh structure (a), orders of elements (b), maps of errors for various values of w (c).

Γ^D boundary part, which are utilized to approximate the solution error. In the next step, both sides of Eq. (31) are converted into a weak form to obtain the following problem: For given function u^p find function $e \in S$ such that

$$b_w(v, e) = l(v) + l_w(v; \hat{e}) - b(v, u^p) \quad \forall v \in S \text{ and for } w \rightarrow 0 \tag{33}$$

The test as well as the trial functions in Eq. (33) belong to the infinite space, that is $v, e \in S$, while the approximate solution belongs to the discrete space $u^p \in S^p$. We want to find an approximation of error e , and so the problem defined in Eq. (33) is rewritten in the finite space as follows: For given function $u^p \in S^p$ find approximation $e^{\bar{p}} \in S^{\bar{p}}$ such that

$$b_w(v^{\bar{p}}, e^{\bar{p}}) = l(v^{\bar{p}}) + l_w(v^{\bar{p}}; \hat{e}) - b(v^{\bar{p}}, u^p) \quad \forall v^{\bar{p}} \in S^{\bar{p}} \text{ and for small value of } w \tag{34}$$

The symbol \bar{p} indicates the order of approximation space, in which the error function is approximated. It is possible that \bar{p} is the same as p , but much better results are obtained when $\bar{p} > p$. In this paper, it is assumed that $\bar{p} = p + 1$. As it is noticed in “Approximation in DG method” Section, the enriched space in the DGF method is quite easy to construct due to the hierarchical form of the basis functions.

A similar error approximation equation to the one presented in Eq. (34) can be generated for the elasticity problem. In that case the final problem of error approximation reads: For given \mathbf{u}^p find $\mathbf{e}^{\bar{p}} \in S^{\bar{p}}$ in Ω such that

$$b_w^v(\mathbf{v}^{\bar{p}}, \mathbf{e}^{\bar{p}}) = l^v(\mathbf{v}^{\bar{p}}) + l_w^v(\mathbf{v}^{\bar{p}}; \hat{\mathbf{e}}) - b^v(\mathbf{v}^{\bar{p}}, \mathbf{u}^p) \quad \forall \mathbf{v}^{\bar{p}} \in S^{\bar{p}} \text{ and for small value of } w \tag{35a}$$

where

$$b^v(\mathbf{v}, \mathbf{u}) = \int_{\Omega} \nabla \mathbf{v} : \mathbf{E} : \nabla \mathbf{u} \, d\Omega + \int_{\Gamma^d} [[\mathbf{v}]] \otimes \mathbf{n} : \mathbf{E} : \langle \nabla \mathbf{u} \rangle \, d\Gamma - \int_{\Gamma^D} \mathbf{v} \otimes \mathbf{n} : \mathbf{E} : \nabla \mathbf{u} \, d\Gamma \tag{35b}$$

The error function $e^{\bar{p}}$ is approximated in the same manner as in Eq. (26), and using Eqs. (34) or (35a) the following system of algebraic equations is obtained

$$e^{\bar{p}} = \Psi^{\bar{p}} \mathbf{E} \quad \text{in } \Omega \quad \rightarrow \quad \mathbf{K}^{\bar{p}} \mathbf{E} = \mathbf{r}^{\bar{p}} \tag{36}$$

The approximation in the elements is constructed in the hierarchic way, as shown in “Approximation in DG method” Section and so, and therefore the system of algebraic equations in (36) is arranged in the following way

$$\mathbf{K}^{\bar{p}} = \begin{bmatrix} \mathbf{K}^p & \bar{\mathbf{K}} \\ \bar{\mathbf{K}}^T & \bar{\mathbf{K}} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \end{bmatrix}, \quad \mathbf{r}^{\bar{p}} = \begin{bmatrix} \mathbf{r}_1^{\bar{p}} \\ \mathbf{r}_2^{\bar{p}} \end{bmatrix} \tag{37}$$

The matrix \mathbf{K}^p has already been used to solve the Eq. (27), so it is already factorized. This is utilized in the error approximation

$$\left(\bar{\mathbf{K}} - \bar{\mathbf{K}}^T (\mathbf{K}^p)^{-1} \bar{\mathbf{K}} \right) \mathbf{E}_2 = \bar{\mathbf{r}}_2^{\bar{p}} - \bar{\mathbf{K}}^T (\mathbf{K}^p)^{-1} \mathbf{r}_1^{\bar{p}} \tag{38a}$$

$$\mathbf{E}_1 = (\mathbf{K}^p)^{-1} \mathbf{r}_1^{\bar{p}} - (\mathbf{K}^p)^{-1} \bar{\mathbf{K}} \mathbf{E}_2 \tag{38b}$$

The solution scheme shown in Eq. (38) requires the solution of the relatively small algebraic problem to get vector \mathbf{E}_2 and then vector \mathbf{E}_1 is obtained. The reduced system of algebraic Eq. (38a) is connected with the enriched part of the approximation space, which is much smaller in comparison to the main problem and therefore it is especially effective for high-order approximation.

Examples

This section presents four benchmark examples, illustrating the performance of the developed error estimation method. Poisson’s problem is considered in the first two examples (“Exponential example for Poisson’s problem, Hyperbolic example for Poisson’s problem” Sections). The third example focuses on an elasticity problem (“Elasticity problem with singularity” Section). In the fourth example the Poisson’s problem is considered once again and an hp mesh adaptation is performed using the error estimation approach to identify the elements for the refinement. Error estimates are compared in each case with the exact error for various mesh densities and approximation orders. In order to present the quality of the error estimation the following efficiency index is defined for the approximate error in the whole domain and in a single finite element

$$\eta^{\bar{p}} = \frac{\ln \|e^{\bar{p}}\|_{L^2(\Omega)}}{\ln \|e\|_{L^2(\Omega)}}, \quad \eta_e^{\bar{p}} = \frac{\ln \|e^{\bar{p}}\|_{L^2(\Omega^e)}}{\ln \|e\|_{L^2(\Omega^e)}} \tag{39}$$

Exponential example for Poisson’s problem. In this section, a benchmark Poisson’s problem is discussed whose exact solution is known a priori. It is the exponential function shown in Eq. (29) with $\alpha = 10$ and $\beta = 2$, depicted in Fig. 6a. It can be observed that the function is non-zero in the region close to the domain center, thus, the error concentrations are expected to appear there. Three polygonal meshes shown in Fig. 6b–d, comprising 50, 250, and 1000 elements, respectively, are used in this example. Additionally, quadrilateral and triangular meshes (not illustrated), consisting of 49/256/1024 and 50/242/1058 elements, respectively, are used in the calculations.

The calculations have been performed for three approximation orders $p = 3, 5, 8$, for each type of mesh. The results are shown in Tables 1, 2 and 3 where the global error measure $\|e\|_{L^2(\Omega)}$ is compared with the approximate global error $\|e^{\bar{p}}\|_{L^2(\Omega)}$ with $\bar{p} = p + 1$ and the efficiency indices are also provided in these tables. It is shown that the value of $\|e^{\bar{p}}\|_{L^2(\Omega)}$ gets closer and closer to the exact value with the increase of mesh density and approximation order. It can be observed in Tables 1, 2 and 3 that the best results have been obtained for the polygonal meshes, while the least accurate results are for the triangular mesh. The rate of convergence for all kinds of meshes is the same, however, the error estimates for the triangular meshes show some irregularity, especially for higher order approximation. In the triangular elements, the vertex inner angles are acute, which may lead to some perturbations when the continuity of the approximate solution is enforced. It is the reason why the error estimation for the triangular elements does not show a stable performance, especially for high-order approximation. The

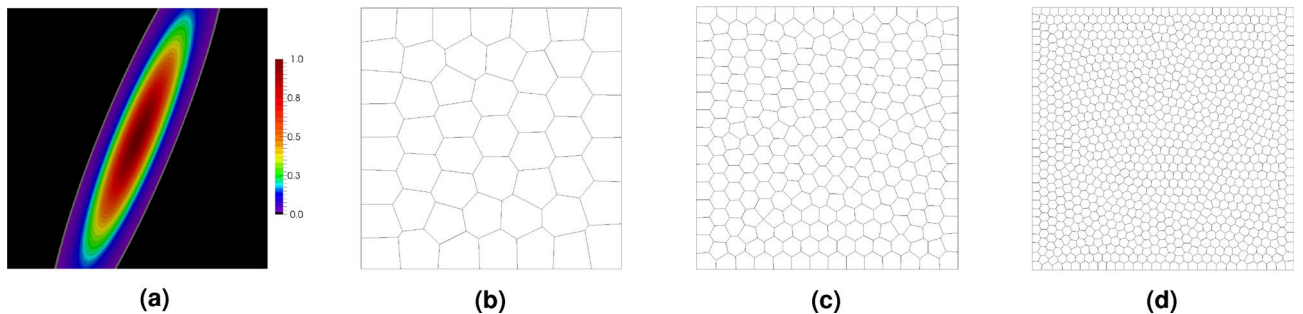


Figure 6. Example 1: exact solution (a), polygonal meshes with various numbers of elements: mp50 (b), mp250 (c) and mp1000 (d).

	p	$\ e\ _{L^2(\Omega)}$	$\ e^{\tilde{p}}\ _{L^2(\Omega)}$	$\eta^{\tilde{p}}$
mp50	3	7.09×10^{-2}	6.76×10^{-2}	1.02
	5	4.25×10^{-3}	4.06×10^{-3}	1.01
	8	7.98×10^{-5}	7.40×10^{-5}	1.01
mp250	3	3.48×10^{-3}	3.40×10^{-3}	1.00
	5	4.56×10^{-5}	4.31×10^{-5}	1.01
	8	9.83×10^{-8}	9.32×10^{-8}	1.00
mp1000	3	1.29×10^{-4}	1.23×10^{-4}	1.01
	5	7.45×10^{-7}	7.13×10^{-7}	1.00
	8	1.92×10^{-10}	1.83×10^{-10}	1.00

Table 1. Example 1: exact and approximate global errors for polygonal meshes.

	p	$\ e\ _{L^2(\Omega)}$	$\ e^{\tilde{p}}\ _{L^2(\Omega)}$	$\eta^{\tilde{p}}$
mqt49	3	5.04×10^{-2}	4.48×10^{-2}	1.04
	5	7.38×10^{-3}	6.93×10^{-3}	1.01
	8	1.60×10^{-4}	1.42×10^{-4}	1.01
mqt256	3	4.22×10^{-3}	4.12×10^{-3}	1.00
	5	7.75×10^{-5}	7.53×10^{-5}	1.00
	8	1.94×10^{-7}	1.87×10^{-7}	1.00
mqt1024	3	3.84×10^{-4}	3.80×10^{-4}	1.00
	5	1.41×10^{-6}	1.35×10^{-6}	1.00
	8	4.20×10^{-10}	3.94×10^{-10}	1.00

Table 2. Example 1: exact and approximate global errors for quadrilateral meshes.

	p	$\ e\ _{L^2(\Omega)}$	$\ e^{\tilde{p}}\ _{L^2(\Omega)}$	$\eta^{\tilde{p}}$
mtr50	3	6.48×10^{-2}	5.66×10^{-2}	1.05
	5	1.45×10^{-2}	1.45×10^{-2}	1.00
	8	1.17×10^{-3}	5.73×10^{-4}	1.11
mtr242	3	4.59×10^{-3}	4.24×10^{-3}	1.01
	5	2.50×10^{-4}	2.40×10^{-4}	1.00
	8	1.98×10^{-6}	1.77×10^{-6}	1.01
mtr1058	3	2.72×10^{-4}	2.69×10^{-4}	1.00
	5	3.55×10^{-6}	3.52×10^{-6}	1.00
	8	1.34×10^{-9}	1.27×10^{-9}	1.00

Table 3. Example 1: exact and approximate global errors for triangular meshes.

polygonal finite elements, i.e. quadrilateral, pentagonal, and hexagonal elements, usually give better results in comparison to the triangular ones, which has been confirmed in this example.

For a deeper analysis, the results have been presented in the form of error maps for three meshes: mp250, msq256, and mtr242 as well as for three values of $p = 3, 5, 8$. The results are shown in Figs. 7, 8 and 9 in the form of maps of exact and approximate errors for the three meshes and the three approximation orders. As it is seen in the figures, the exact error is quite well recovered by the approximate error $e^{\tilde{p}}$. In each case, the error distributions are reconstructed with a relatively high accuracy using the proposed method.

Tab. 1 presents the global efficiency indexes for the polygonal meshes. For a deeper analysis of the error, the maps of the efficiency indexes over the fifth order elements for three meshes m50, m25 and m1000 are shown in Fig. 10. Comparing the maps of efficiency indexes with the error maps in Fig. 7 it can be concluded that the efficiency indexes are very close to one for the elements located at the places of error concentration. At the places where the error level is very small the element efficiency index tilts from unity. This means that the error approximation technique works well at the places with error concentration, however, at the places where the level of the error is very small its approximation may differ from the exact error values.

So far, the error approximation methods discussed in this paper have been applied only to meshes with uniform approximation orders. However, the method is also suitable for unstructured meshes with non-uniform

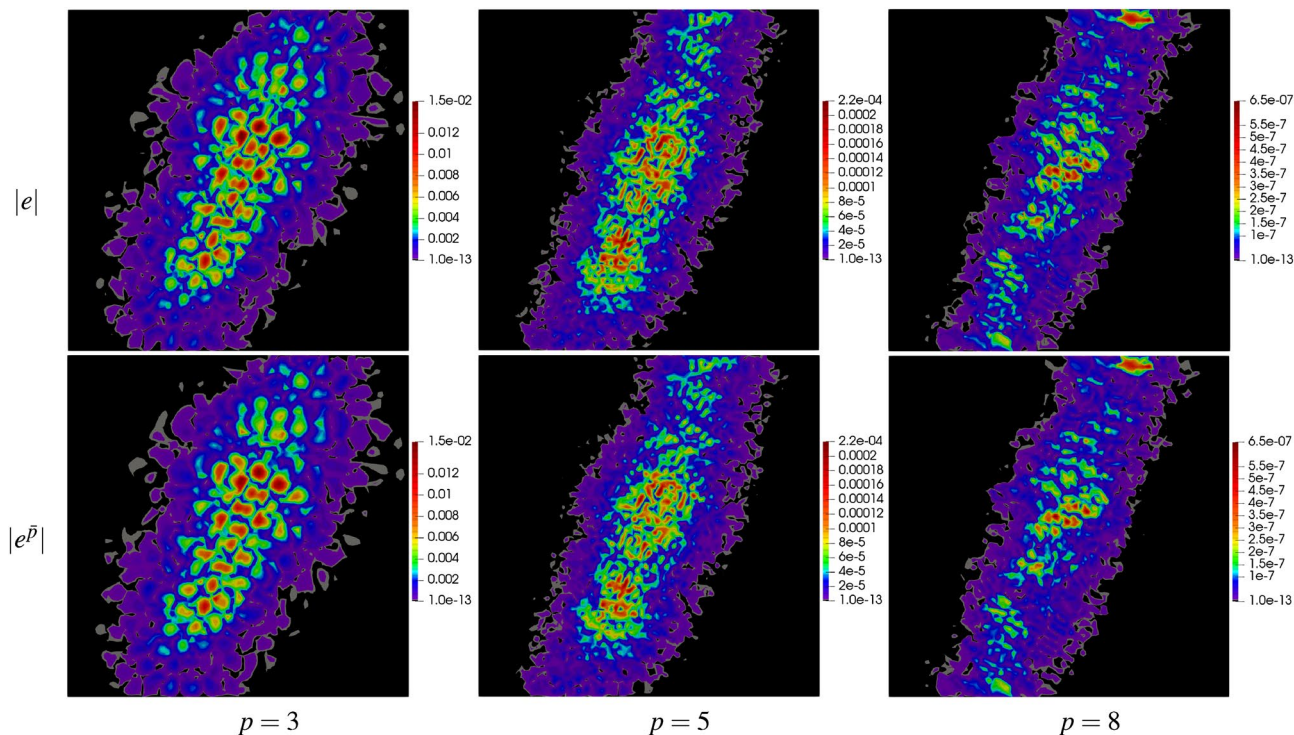


Figure 7. Example 1: maps of exact error $|e|$ and approximate error $|e^{\bar{p}}|$ on polygonal mesh with 250 elements for $p = 3, 5, 8$.

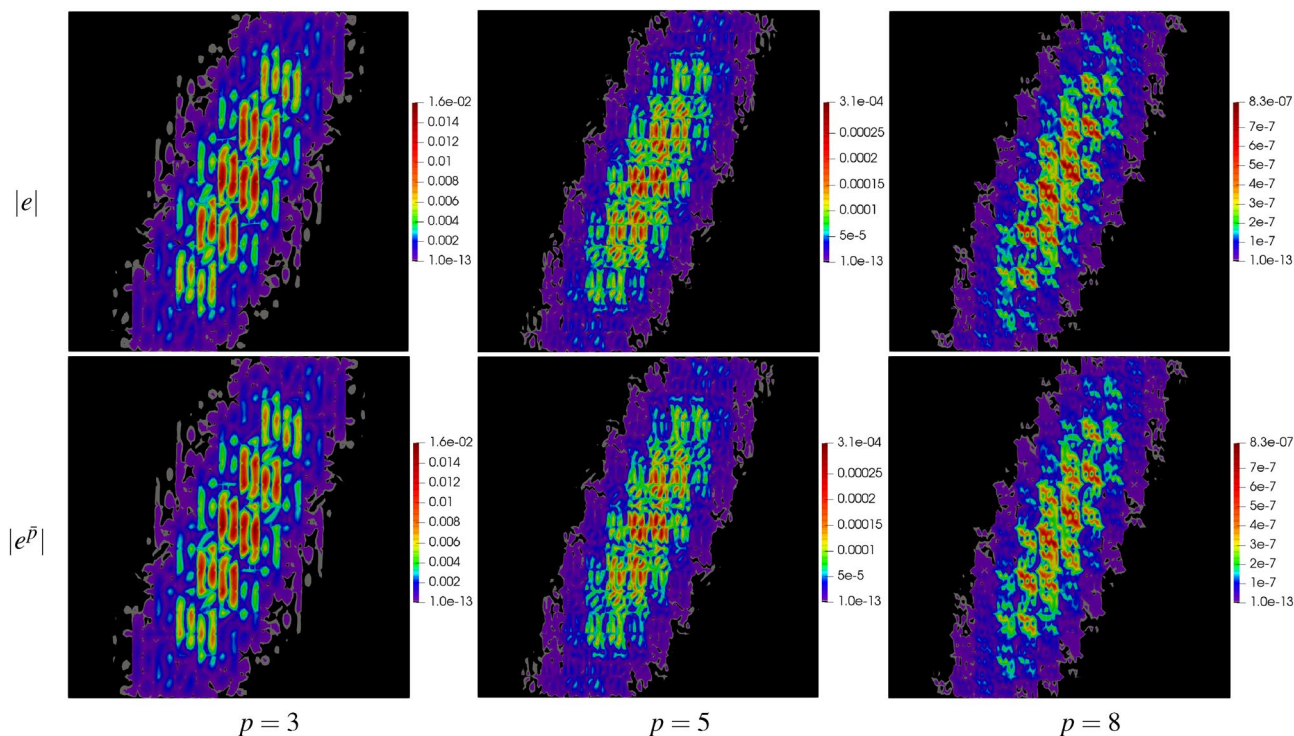


Figure 8. Example 1: maps of exact error $|e|$ and approximate error $|e^{\bar{p}}|$ on quadrilateral mesh with 256 elements for $p = 3, 5, 8$.

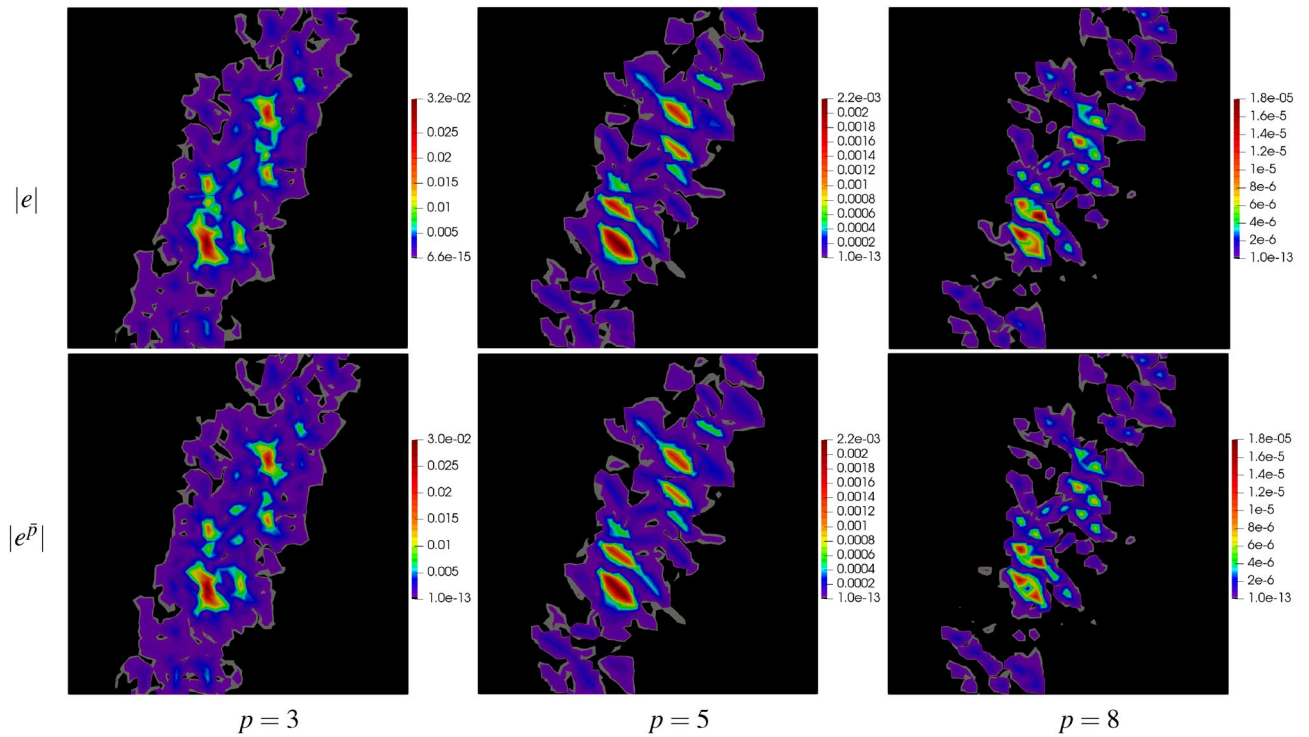


Figure 9. Example 1: maps of exact error $|e|$ and approximate error $|e^{\bar{p}}|$ on triangular mesh with 242 elements for $p = 3, 5, 8$.

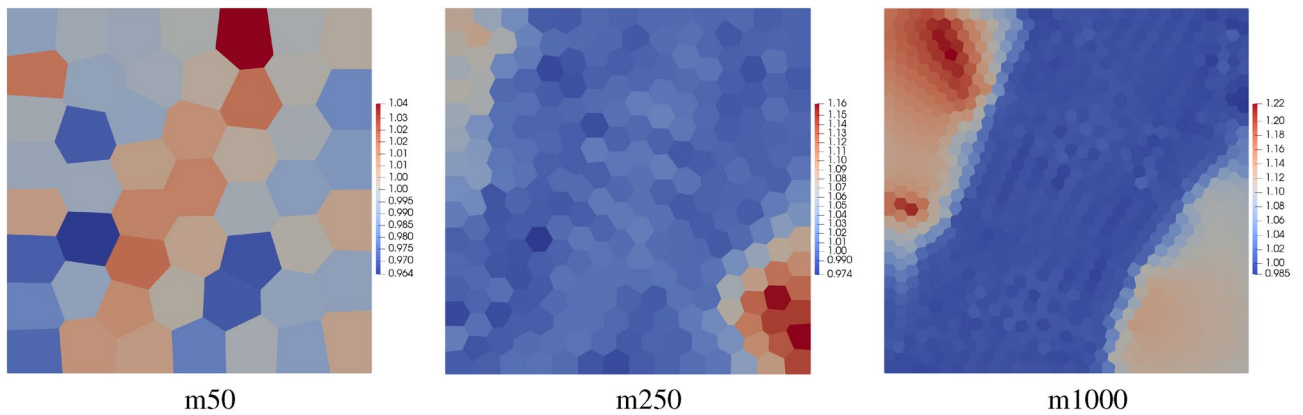


Figure 10. Example 1: maps of element error efficiency indexes for three meshes m50, m250 and m1000 with $p = 5$.

elements orders. The Poisson’s exponential benchmark has also been solved using a quadrilateral, randomly refined mesh where the orders of the elements have been chosen randomly from the range $p = 2$ to $p = 10$. The mesh structure and the map of the orders of the elements are depicted in Fig. 11. The analyzed error approximation technique is able to cope with such a strongly heterogeneous mesh because the error can be approximated with a relatively good accuracy as shown in Fig. 12.

Hyperbolic example for Poisson’s problem. The Poisson’s problem defined in the square domain $[-1, 1] \times [-1, 1]$ is also the focus of this section. This time the exact solution is a function which is hyperbolic in the y direction and trigonometrical in the x direction

$$u(x, y) = \tanh(50y) + \sin(x) + 2 \tag{40}$$

Figure 13 shows maps of the exact solution given in Eq. (40) and of its both derivatives. The function exhibits a strong change in the y direction for all points close to line $y = 0$. The non-zero values of the derivative $u_{,y}$ appear only in the narrow band along the x axis, while the derivative $u_{,x}$ is quite smooth across the whole domain. Due to large gradients, errors are expected to occur in the regions close to line $y = 0$.

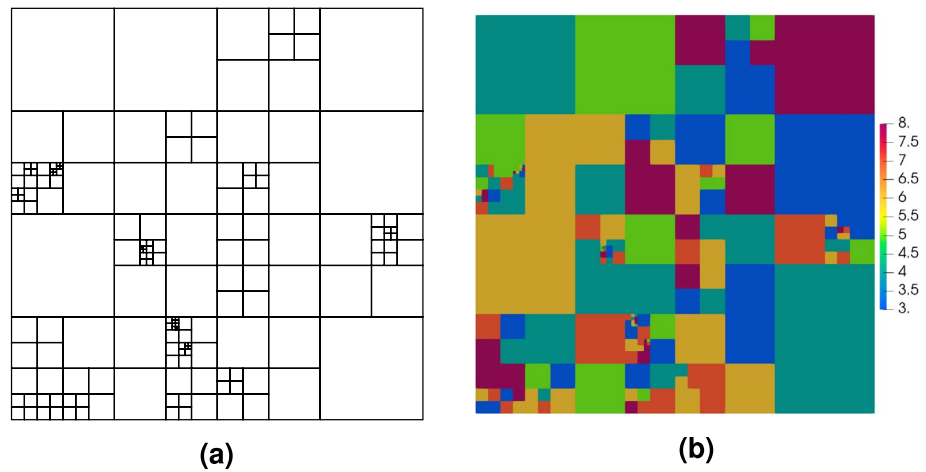


Figure 11. Example 1: randomly refined quadrilateral mesh (a) and map of randomly selected element orders (b).

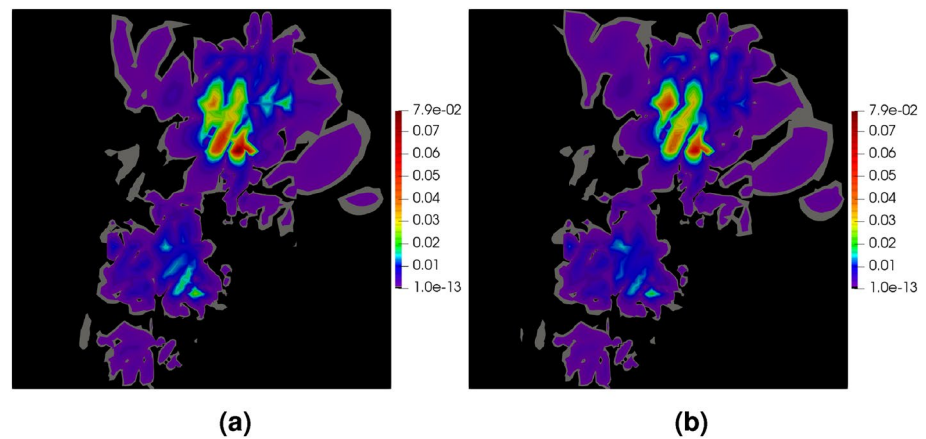


Figure 12. Example 1: map of exact error $|e|$ for the randomly generated quadrilateral mesh (a) and its approximation $|e^P|$ (b).

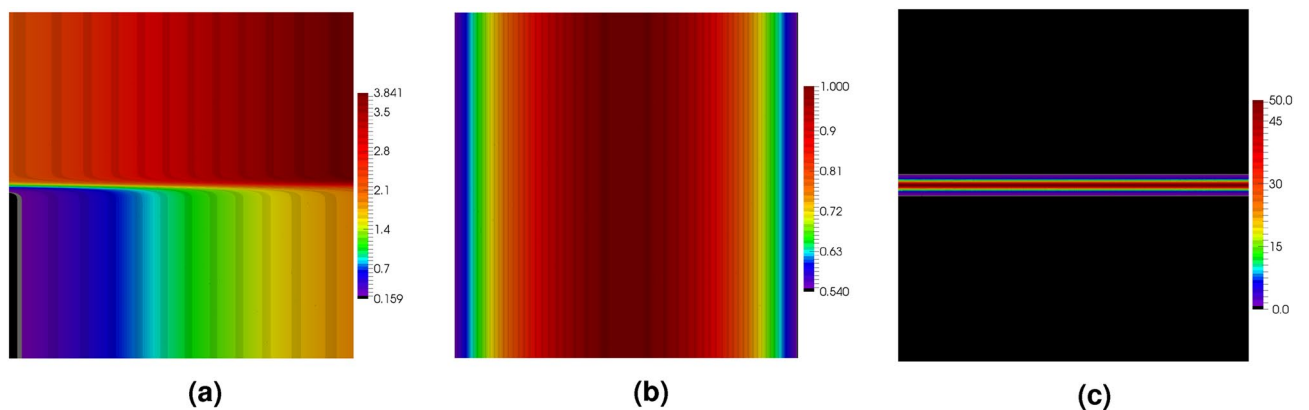


Figure 13. Example 2: exact solution u (a) and its derivatives u_x (b) and u_y (c).

The domain has been discretized with the polygonal mesh mp1000, see Fig. 6d. Uniform approximation orders $p = 3, 5, 8$ have been applied to all elements in the mesh. The global errors, their approximations and efficiency indices are shown in Table 4. The approximate global errors, once again, are very close to the exact errors. The maps of errors and their approximations are presented in Fig. 14, showing accurate concentrations of errors produced by the new error approximation technique.

Elasticity problem with singularity. In this example, a benchmark elasticity problem, defined in Eq. (2), is analyzed. In the example⁶⁵, the exact solution is written in terms of polar coordinates

$$\begin{aligned}
 u_r(r, \theta) &= \frac{1}{2\mu} r^\alpha [-(\alpha + 1) \cos((\alpha + 1)\theta) + (C_2 - \alpha - 1)C_1 \cos((\alpha - 1)\theta)] \\
 u_\theta(r, \theta) &= \frac{1}{2\mu} r^\alpha [(\alpha + 1) \sin((\alpha + 1)\theta) + (C_2 + \alpha - 1)C_1 \sin((\alpha - 1)\theta)]
 \end{aligned}
 \tag{41}$$

where $C_1 = -\cos((\alpha + 1)\omega) / \cos((\alpha - 1)\omega)$, $C_2 = 2(\lambda + 2\mu) / (\lambda + \mu)$, $\omega = 3\pi/4$ and the critical exponent α is the positive solution of the equation $\alpha \sin(2\omega) + \sin(2\alpha\omega) = 0$, so that $\alpha \approx 0.544483737$.

The solution of Eq. (41) is singular at the origin since the stresses at this point tend to infinity. That is why in the approximate solution the error concentration is expected to appear around this point. The benchmark was originally solved on the so-called rotated L-shaped domain, see⁶⁵. In this paper, the standard L-shaped domain with vertices at points: $(-1, -1), (1, -1), (1, 0), (0, 0), (0, 1), (-1, 1)$ is considered. To obtain the solution in such a domain, both the global coordinates and the displacements need to be appropriately transformed, as shown below. The polar coordinates are constructed using Cartesian coordinates ξ and η

$$[\theta, r] = \text{cart2pol}(\xi, \eta)
 \tag{42}$$

where the auxiliary coordinates (ξ, η) come from the following transformation of the global coordinates

	p	$\ e\ _{L^2(\Omega)}$	$\ e^{\bar{p}}\ _{L^2(\Omega)}$	$\eta^{\bar{p}}$
mp1000	3	3.08×10^{-2}	2.66×10^{-2}	1.04
	5	4.27×10^{-3}	3.61×10^{-3}	1.03
	8	3.29×10^{-4}	2.69×10^{-4}	1.02

Table 4. Example 2: exact and approximate global errors for polygonal meshes.

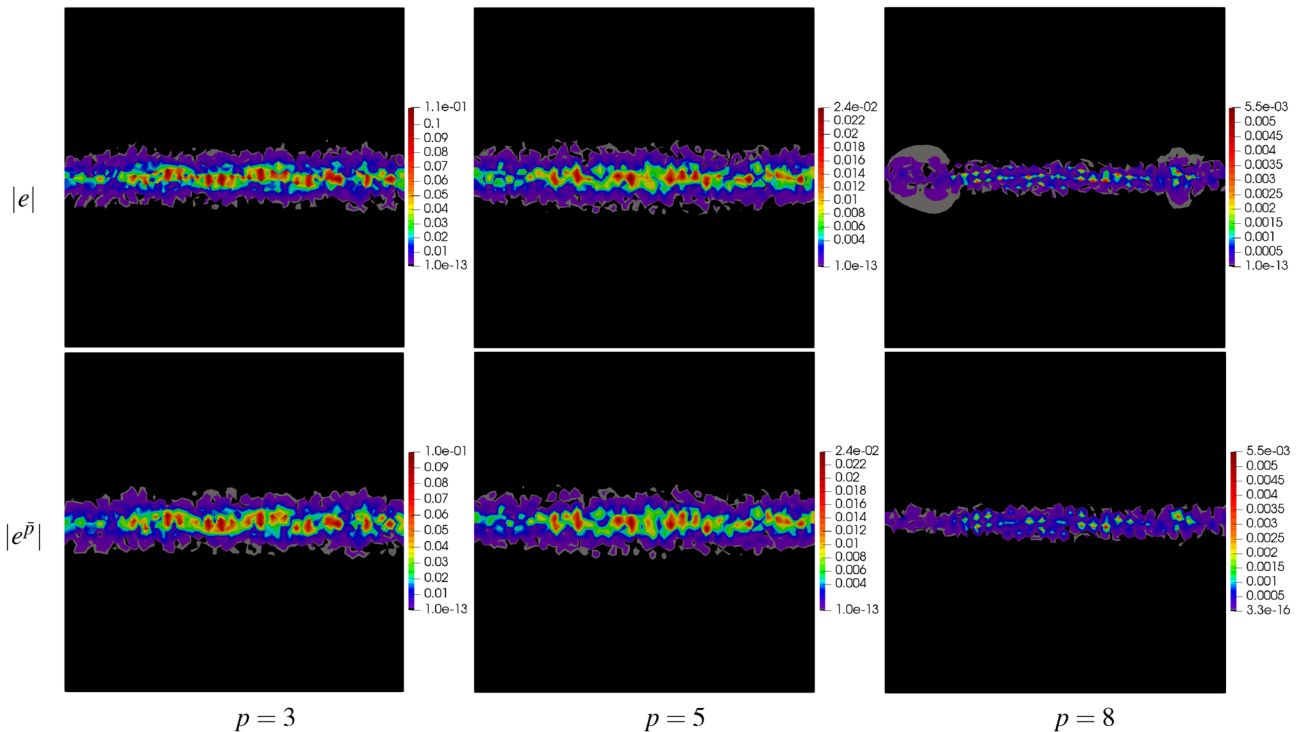


Figure 14. Example 2: maps of exact error $|e|$ and approximate error $|e^{\bar{p}}|$ on polygonal mesh with 1000 elements for $p = 3, 5, 8$.

$$\begin{aligned} \phi &= -3\pi/4 \\ \xi &= \cos(\phi)x + \sin(\phi)y \\ \eta &= -\sin(\phi)x + \cos(\phi)y \end{aligned} \tag{43}$$

When the displacements are calculated in the polar coordinates, the displacements in the global coordinates are obtained with the same transformation angle ϕ

$$\begin{aligned} u_x(x, y) &= \cos(\phi)u_r - \sin(\phi)u_\theta \\ u_y(x, y) &= \sin(\phi)u_r + \cos(\phi)u_\theta \end{aligned} \tag{44}$$

Fig. 15 presents the maps of the exact solution of the considered problem as well as the polygonal mesh comprising 200 elements. The calculations have been performed for Young’s modulus $E = 1$ and $\nu = 0.3$. The unified approximation order for all the elements has been chosen as $p = 5$.

The maps of errors for the polygonal mesh, provided in Fig. 16a,b, enable the comparison of the exact and approximated errors. In this example, the error is concentrated, as expected, in the vicinity of the origin. The error approximation procedure is able to locate the place with the error concentrations and correctly recover the level of the error. The global efficiency index in this example is $\eta^p = 0.5$, which can be caused by the error concentration in the very narrow region in the vicinity of the origin. For a deeper analysis of the situation the maps of the element efficiency indexes are presented in Fig. 16c. It can be noticed that in a large part of the domain the element efficiency indexes are close to one in most of the elements except those located in two regions where the indexes are about 0.7, however, the regions are not located near the origin. This example confirms that the procedure for error approximation performs well also for the elasticity problem.

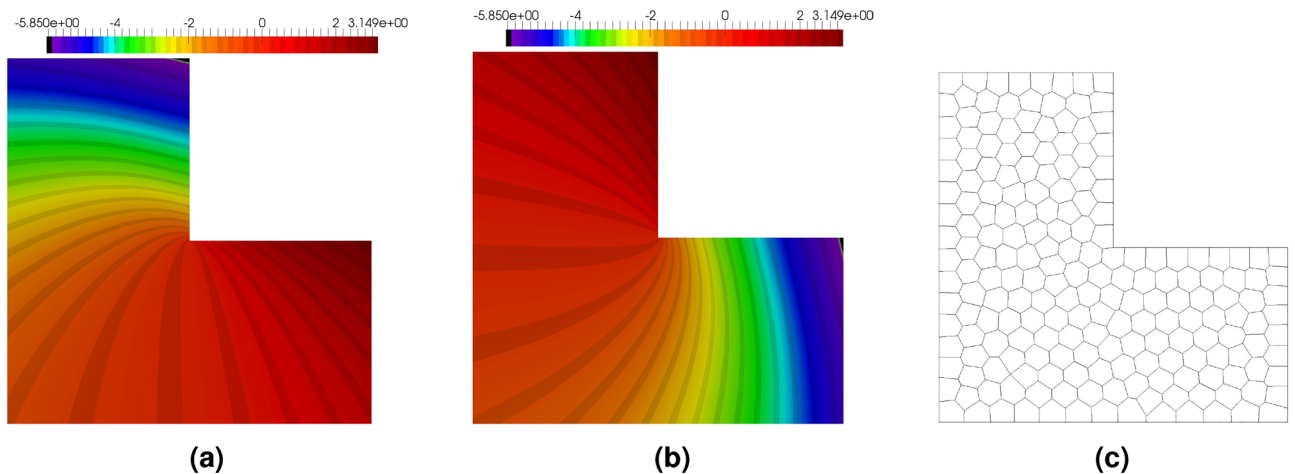


Figure 15. Example 3: component u_x of exact solution (a), component u_y of exact solution (b), polygonal mesh with 200 elements, lm_{200} (c).

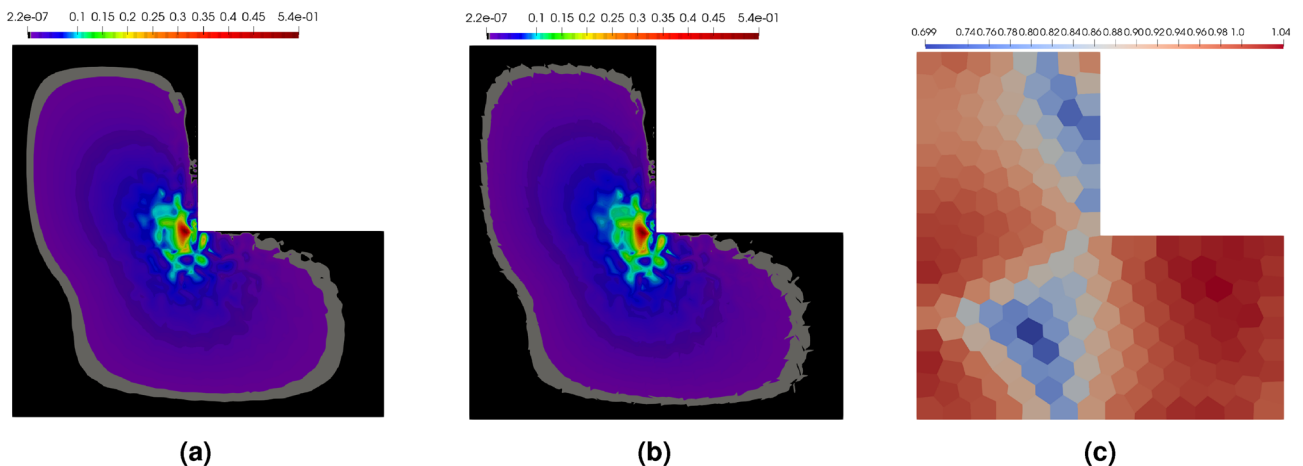


Figure 16. Example 3: exact error map $|e|$ (a), the error approximation $|e^p|$ (b), and the map of the efficiency indexes in elements (c).

Poisson's problem with hp mesh adaptation. In this example, we solve the benchmark Poisson's problem whose exact solution in the domain $[-1, 1] \times [-1, 1]$ is the same as shown in Eq. (29), but now the values of the parameters are $\alpha = 200$ and $\beta = 6$. This causes the value of function $u(x, y)$ to be almost zero in the whole domain except the vicinity of a narrow band going through the domain, see Fig. 17.

In order to obtain an accurate solution by means of the discrete method the mesh should be refined along the narrow band. In this example, an automatic *hp* mesh refinement is applied using the a posteriori error approximation presented in this paper. In this example quadrilateral elements are used, since they enable flexible refinement in the DGF method, as shown in the example in “Exponential example for Poisson's problem” Section. In each step of the refinement, the problem is solved on the current mesh with the error approximation. Then, the error level for each element is calculated.

Various scenarios for the combined *h* and *p* mesh refinement can be applied. The main role of the automatic mesh adaptation is to minimize the global error and make the error uniformly distributed in the domain. The mesh adaptation is performed in refinement steps, in which the error is estimated after the problem solution on the current mesh. It is important to prepare a proper refinement scenario to reach the final mesh with a relatively small number of refinement steps. In this application the error is estimated for every finite element, then the elements with maximum and minimum errors are identified. In every adaptation step, two error parameters are set: the first is the 0.2 and 0.8 combination of the maximum and minimum errors in elements, respectively, and the second is the 0.05 and 0.95 combination of the same errors. The two error parameters are applied in the *hp* mesh refinement. The finite elements in which the error is higher than the second error parameter are *p* refined by increasing the approximation order by two and they are divided into four elements. Moreover, their neighbors are *p* refined by one and *h* refined by dividing them into four elements. The unrefined elements with errors greater than the first error parameter are divided into four elements. The whole procedure starts with a very coarse mesh, 3×3 with second-order finite elements. In the first three steps, only the *h* refinement is applied to locate the places with higher errors. Afterward, the combined *hp* refinement is applied as described above. Many *hp* schemes can be applied to reach a similar final refined mesh. However, in this case, the *p* and *h* refinement are applied in every step of the refinement procedure to limit the number of iterations. Using this approach we obtained the final refined mesh in relatively few steps. It should be emphasized that we got the refined mesh that consists of elements with a wide spectrum of orders, from $p = 2$ up to $p = 13$. In the refinement procedure, the large and small finite elements as well as the elements with low and high order of approximation can be neighbors. It shows that the properties of the DGF method allow for very flexible mesh refinement. The refinement of the *p* or *h* type can be applied only to a single element in the DGF method without the need to interfere with the neighboring elements. In this example, the ability of the DGF method to perform effectively combined mesh refinement is presented, and automated adaptivity is based on the effective error estimation in the elements.

Selected meshes obtained during the refinement procedure are shown in Fig. 18. It is seen that the mesh is mainly refined along the band where the errors are concentrated, so that the global error can be reduced with a relatively small number of degrees of freedom (*#dofs*). In the first stage of the process, only the *h*-refinement is performed in order to identify the places with error concentration. In Fig. 19a the convergence diagram for the refinement is depicted. This is visible in the quite flat first part of the convergence diagram. After this stage, further refinement is practically concentrated in the vicinity of the specific band, which results in the steep second part of the diagram. The value of the efficiency index η^p during the refinement procedure is shown in Fig. 19b. For the first five steps of the refinement the efficiency index is underestimated, however in the further steps of the refinement procedure the index values are close to one. The initial perturbations in the efficiency index are caused by the coarse mesh structure and the fact that the error is concentrated in the narrow band in the domain. For the refined mesh the value of the global error is correctly estimated. In the fifth step the efficiency index is about -8, where the real error was 1.5×10^0 and the approximated error was 4.2×10^{-2} . In Fig. 20 the maps of the element efficiency indexes are presented for the fifth, ninth and thirteenth adaptation steps. It can be noticed

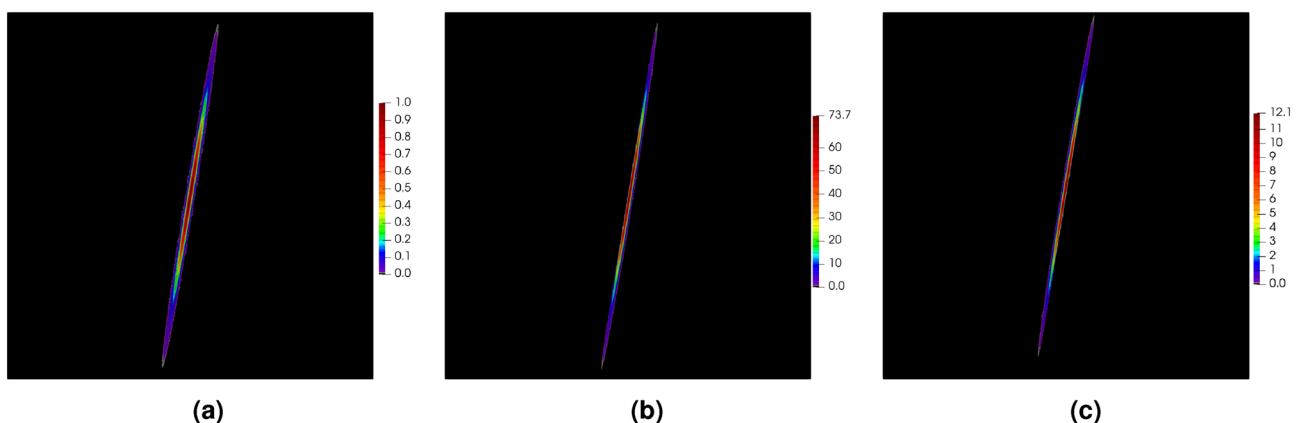


Figure 17. Example 4: exact solution $u(x, y)$ of Poisson's problem in domain $[-1, 1] \times [-1, 1]$ (a), its derivatives in *x* direction (b) and in *y* direction (c).

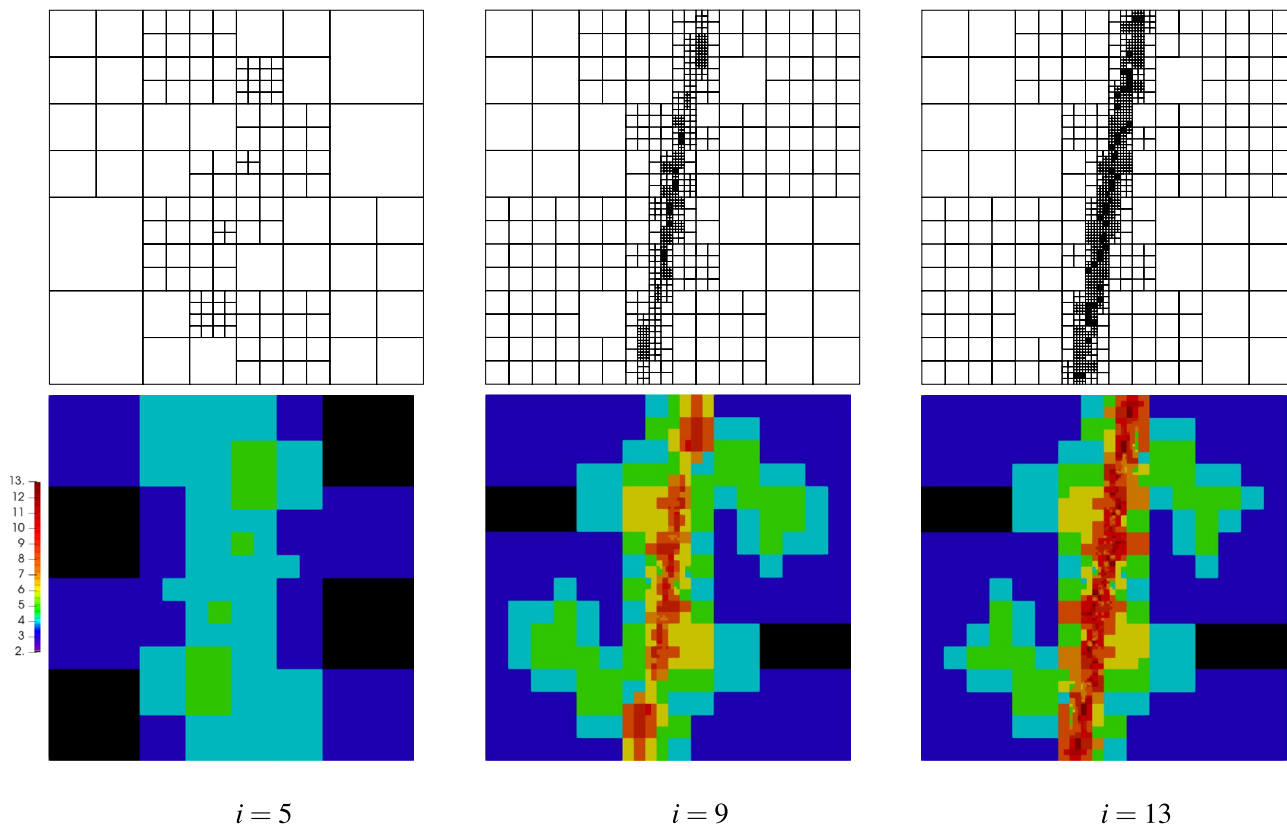


Figure 18. Example 4: meshes with maps of elements orders in selected steps of automatic hp -refinement.

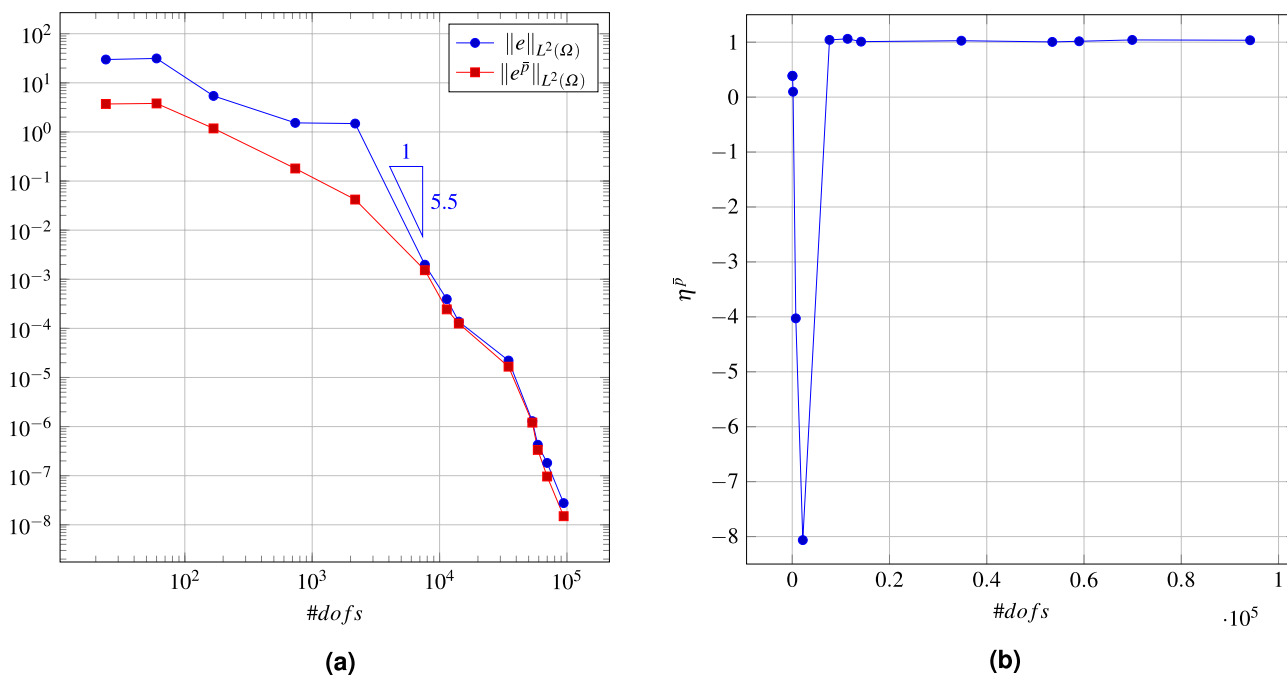


Figure 19. Example 4: convergence diagram in hp mesh refinement procedure (a), the efficiency indexes calculated in the refinement procedure (b).

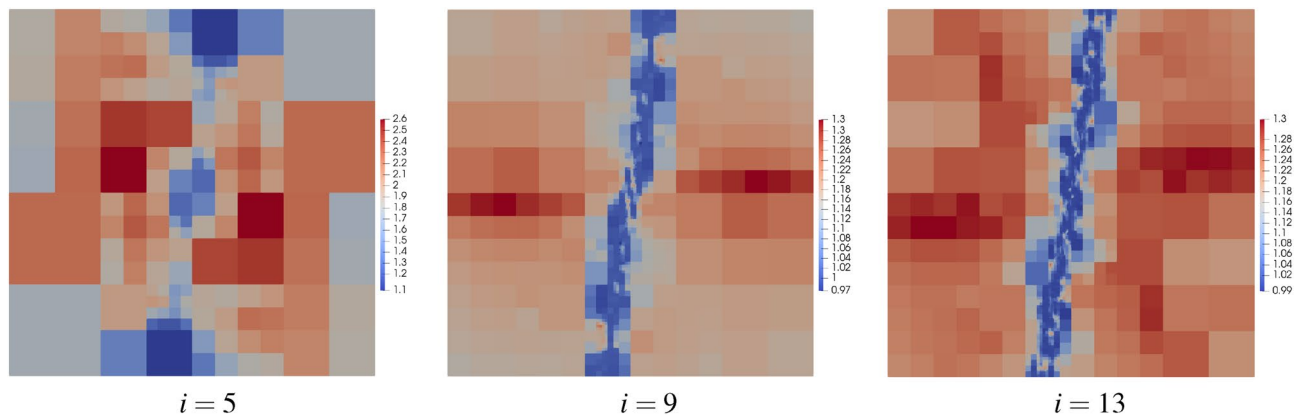


Figure 20. Example 4: maps of element efficiency indexes in selected steps of automatic hp -refinement.

that in each case the indexes are close to one near the band with error concentration and the element indexes with higher values than one are generally at places with small error values.

Conclusions

The paper presents the method for error approximation in the DGF method for elliptic problems analyzed with various polygonal meshes. Some unique properties of the DGF method are utilized to obtain an effective error approximation in the approximate solution. As a result, a new technique for error approximation is proposed.

Chebyshev polynomials are applied as the basis functions in the DGF method. Their definition is recursive, thus they are hierarchical, which enables an efficient generation of the matrices for higher-order approximation space. The hierarchical formation of the stiffness matrix is utilized to get the error approximation developed in the upgraded approximation space.

The stability parameter w plays an important role in the DGF method, since it is used for evaluating the so-called numerical fluxes on the mesh skeleton and for enforcing the continuity of the final solution. Generally, the stability parameter should be small for the compatibility of the final solution. However, the DGF method is not very sensitive to the variations of this parameter, i.e. it can be changed significantly with a very small influence on the final solution. This property is utilized in this paper for error approximation techniques.

The method of error approximation has been illustrated with four benchmark examples, in which two kinds of two-dimensional elliptic problems, Poisson's and elasticity problem, have been analyzed. It has been shown that the error approximation method on the polygonal meshes is able to locate places with error concentrations. In the last example, the method has been applied with success for an automatic hp mesh refinement. The method of error approximation can recover the exact error measure of high quality for polygonal meshes, including quadrilateral and triangular ones.

So far the error approximation method for the DGF method has been applied in 2D elliptic problems. In the further research the method will be applied to other problems like linear and non-linear Navier–Stokes problem, or diffusion-convection transport in a compressible or incompressible medium.

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 21 November 2022; Accepted: 21 June 2023

Published online: 04 July 2023

References

1. Jaśkowiec, J. Discontinuous Galerkin method on reference domain. *Comput. Assist. Methods Eng. Sci.* **22**, 177–204 (2015).
2. Jaśkowiec, J. The discontinuous Galerkin method with higher degree finite difference compatibility conditions and arbitrary local and global basis functions. *Comput. Assist. Methods Eng. Sci.* **23**, 109–132 (2017).
3. Jaśkowiec, J., Pluciński, P. & Stankiewicz, A. Discontinuous Galerkin method with arbitrary polygonal finite elements. *Fin. Elem. Anal. Des.* **20**, 1–16 (2016).
4. Jaśkowiec, J. Very high-order discontinuous Galerkin method in elliptic problems. *Comput. Mech.* **62**, 1–21. <https://doi.org/10.1007/s0046> (2017).
5. Jaśkowiec, J. Application of discontinuous Galerkin method to mechanical 2D problem with arbitrary polygonal and high-order finite elements. *Comput. Methods Appl. Mech. Eng.* **323**, 389–415 (2017).
6. Reed, W. & Hill, T. *Triangular Mesh Methods for the Neutron Transport Equation*. Report LA-UR-73-479 (Los Alamos Scientific Laboratory, 1973).
7. Cockburn, B. B., Karniadakis, G. & Shu, C.-W. (eds.) *Discontinuous Galerkin methods: theory, computation, and applications*. Lecture notes in computational science and engineering (Springer, 2000).
8. Di Pietro, D. & Ern, A. *Mathematical Aspects of Discontinuous Galerkin Methods*. *Mathématiques et Applications* (Springer, 2011).
9. Hesthaven, J. S. & Warburton, T. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications* 1st edn. (Springer Publishing Company, Incorporated, 2007).
10. Mu, L., Wang, J., Wang, Y. & Ye, X. Interior penalty discontinuous Galerkin method on very general polygonal and polyhedral meshes. *J. Comput. Appl. Math.* **255**, 432–440. <https://doi.org/10.1016/j.cam.2013.06.003> (2014).

11. Heimann, F., Engwer, C., Ippisch, O. & Bastian, P. An unfitted interior penalty discontinuous Galerkin method for incompressible Navier–Stokes two-phase flow. *Int. J. Numer. Methods Fluids* **71**, 269–293. <https://doi.org/10.1002/flid.3653> (2013).
12. Rivière, B. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*. Frontiers in Applied Mathematics (Society for Industrial and Applied Mathematics, 2008).
13. Nitsche, J. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* **36**, 9–15. <https://doi.org/10.1007/BF02995904> (1971).
14. Cockburn, B. & Shu, C.-W. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.* **35**, 2440–2463. <https://doi.org/10.1137/S0036142997316712> (1998).
15. Castillo, P. A review of the local Discontinuous Galerkin (LDG) method applied to elliptic problems. *Appl. Numer. Math.* **56**, 1307–1313. <https://doi.org/10.1016/j.apnum.2006.03.016> (2006).
16. Rupp, A. & Knabner, P. Convergence order estimates of the local discontinuous Galerkin method for instationary Darcy flow. *Numer. Methods Part. Differ. Equs.* **33**, 1374–1394. <https://doi.org/10.1002/num.22150> (2017).
17. Bassi, F. & Rebay, S. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J. Comput. Phys.* **131**, 267–279. <https://doi.org/10.1006/jcph.1996.5572> (1997).
18. Baumann, C. E. & Oden, J. T. A discontinuous *hp* finite element method for convection-diffusion problems. *Comput. Methods Appl. Mech. Eng.* **175**, 311–341. [https://doi.org/10.1016/S0045-7825\(98\)00359-4](https://doi.org/10.1016/S0045-7825(98)00359-4) (1999).
19. Cockburn, B. & Gopalakrishnan, J. A characterization of hybridized mixed methods for second order elliptic problems. *SIAM J. Numer. Anal.* **42**, 283–301. <https://doi.org/10.1137/S0036142902417893> (2004).
20. Samii, A., Panda, N., Michoski, C. & Dawson, C. A hybridized discontinuous Galerkin method for the nonlinear Korteweg-de Vries equation. *J. Sci. Comput.* **68**, 191–212 (2016).
21. Cockburn, B., Gopalakrishnan, J. & Lazarov, R. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM J. Numer. Anal.* **47**, 1319–1365 (2009).
22. Nguyen, N. C., Peraire, J. & Cockburn, B. An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. *J. Comput. Phys.* **230**, 1147–1170 (2011).
23. Zhao, J., Zhao, W. & Xu, Y. Hybridizable discontinuous Galerkin methods for space-time fractional advection-dispersion equations. *Appl. Math. Comput.* **442**, 127745 (2023).
24. Troesch, N. & Higdon, J. A fully-implicit hybridized discontinuous Galerkin spectral element method for two phase flow in petroleum reservoirs. *J. Comput. Phys.* **474**, 111824 (2023).
25. Demkowicz, L. F. & Gopalakrishnan, J. *An Overview of the Discontinuous Petrov Galerkin Method* 149–180 (Springer International Publishing, 2014).
26. Keith, B. *et al.* An ultraweak DPG method for viscoelastic fluids. *J. Non-Newton. Fluid Mech.* **247**, 107–122. <https://doi.org/10.1016/j.jnnfm.2017.06.006> (2017).
27. Borker, R., Farhat, C. & Tezaur, R. A discontinuous Galerkin method with Lagrange multipliers for spatially-dependent advection–diffusion problems. *Comput. Methods Appl. Mech. Eng.* **327**, 93–117. <https://doi.org/10.1016/j.cma.2017.08.024> (2017).
28. Arnold, D. N., Brezzi, F., Cockburn, B. & Marini, L. D. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.* **39**, 1749–1779. <https://doi.org/10.1137/S0036142901384162> (2002).
29. Kirby, R. M. & Karniadakis, G. E. Selecting the numerical flux in discontinuous Galerkin methods for diffusion problems. *J. Sci. Comput.* **22**, 385–411 (2005).
30. Segeth, K. A review of some a posteriori error estimates for adaptive finite element methods. *Math. Comput. Simul.* **80**, 1589–1600. <https://doi.org/10.1016/j.matcom.2008.12.019> (2010).
31. Zhang, W., Nie, Y. & Gu, Y. Adaptive finite element analysis of elliptic problems based on bubble-type local mesh generation. *J. Comput. Appl. Math.* **280**, 42–58. <https://doi.org/10.1016/j.cam.2014.11.049> (2015).
32. Zboinski, G. Adaptive *hpq* finite element methods for the analysis of 3D-based models of complex structures. Part 2. A posteriori error estimation. *Comput. Methods Appl. Mech. Eng.* **267**, 531–565. <https://doi.org/10.1016/j.cma.2013.08.018> (2013).
33. Kamenski, L. A study on using hierarchical basis error estimates in anisotropic mesh adaptation for the finite element method. *Eng. Comput.* **28**, 451–460. <https://doi.org/10.1007/s00366-011-0240-z> (2012).
34. Jaworska, I. & Orkisz, J. Estimation of a posteriori computational error by the higher order multipoint meshless FDM. *Comput. Informa.* **36**, 1447–1466 (2017).
35. Amani, J., Bagherzadeh, A. S. & Rabczuk, T. Error estimate and adaptive refinement in mixed discrete least squares meshless method. *Math. Probl. Eng.* <https://doi.org/10.1155/2014/721240> (2014).
36. Hartmann, R. & Houston, P. Adaptive discontinuous Galerkin finite element methods for nonlinear hyperbolic conservation laws. *SIAM J. Sci. Comput.* **24**, 979–1004. <https://doi.org/10.1137/S1064827501389084> (2003).
37. Krivodonova, L. & Flaherty, J. E. Error estimation for discontinuous Galerkin solutions of two-dimensional hyperbolic problems. *Adv. Comput. Math.* **19**, 57–71. <https://doi.org/10.1023/A:1022894504834> (2003).
38. Adjerid, S. & Baccouch, M. The discontinuous Galerkin method for two-dimensional hyperbolic problems. Part I: Superconvergence error analysis. *J. Sci. Comput.* **33**, 75–113. <https://doi.org/10.1007/s10915-007-9144-x> (2007).
39. Adjerid, S. & Baccouch, M. The discontinuous Galerkin method for two-dimensional hyperbolic problems Part II: A Posteriori error estimation. *J. Sci. Comput.* **38**, 15–49. <https://doi.org/10.1007/s10915-008-9222-8> (2009).
40. Adjerid, S. & Baccouch, M. Asymptotically exact a posteriori error estimates for a one-dimensional linear hyperbolic problem. *Appl. Numer. Math.* **60**, 903–914. <https://doi.org/10.1016/j.apnum.2010.04.014> (2010).
41. Adjerid, S. & Mechali, I. A superconvergent discontinuous Galerkin method for hyperbolic problems on tetrahedral meshes. *J. Sci. Comput.* **58**, 203–248 (2014).
42. Rivière, B. & Wheeler, M. A posteriori error estimates for a discontinuous Galerkin method applied to elliptic problems. Log number: R74. *Comput. Math. Appl.* **46**, 141–163. [https://doi.org/10.1016/S0898-1221\(03\)90086-1](https://doi.org/10.1016/S0898-1221(03)90086-1) (2003).
43. Tabarraei, A. & Sukumar, N. Adaptive computations using material forces and residual-based error estimators on quadtree meshes. *Comput. Methods Appl. Mech. Eng.* **196**, 2657–2680. <https://doi.org/10.1016/j.cma.2007.01.016> (2007).
44. Houston, P., Schötzau, D. & Wihler, T. P. Energy norm a posteriori error estimation of *hp*-adaptive discontinuous Galerkin methods for elliptic problems. *Math. Mod. Methods Appl. Sci.* **17**, 33–62. <https://doi.org/10.1142/S0218202507001826> (2007).
45. Giani, S., Grubisic, L., Hakula, H. & Ovall, J. S. An a posteriori estimator of eigenvalue/eigenvector error for penalty-type discontinuous Galerkin methods. *Appl. Math. Comput.* **319**, 562–574. <https://doi.org/10.1016/j.amc.2017.07.007> (2018).
46. Hakula, H., Neilan, M. & Ovall, J. S. A posteriori estimates using auxiliary subspace techniques. *J. Sci. Comput.* **72**, 97–127. <https://doi.org/10.1007/s10915-016-0352-0> (2017).
47. Baccouch, M. Superconvergence of the local discontinuous Galerkin method for the sine-Gordon equation in one space dimension. *J. Comput. Appl. Math.* **333**, 292–313. <https://doi.org/10.1016/j.cam.2017.11.007> (2018).
48. Baccouch, M. A superconvergent local discontinuous Galerkin method for nonlinear two-point boundary-value problems. *Numer. Algorithms* <https://doi.org/10.1007/s11075-017-0456-0> (2017).
49. Baccouch, M. A recovery-based error estimator for the discontinuous Galerkin method for transient linear hyperbolic conservation laws on Cartesian grids. *Int. J. Comput. Methods* **14**, 43. <https://doi.org/10.1142/S0219876217500621> (2017).
50. Baccouch, M. A posteriori error estimates and adaptivity for the discontinuous Galerkin solutions of nonlinear second-order initial-value problems. *Appl. Numer. Math.* **121**, 18–37. <https://doi.org/10.1016/j.apnum.2017.06.001> (2017).

51. Baccouch, M. A posteriori error estimator based on derivative recovery for the discontinuous Galerkin method for nonlinear hyperbolic conservation laws on Cartesian grids. *Numer. Methods Part. Differ. Equs.* **33**, 1224–1265. <https://doi.org/10.1002/num.22141> (2017).
52. Baccouch, M. Superconvergence of the discontinuous Galerkin method for nonlinear second-order initial-value problems for ordinary differential equations. *Appl. Numer. Math.* **115**, 160–179. <https://doi.org/10.1016/j.apnum.2017.01.007> (2017).
53. Baccouch, M. A posteriori local discontinuous Galerkin error estimates for the one-dimensional sine-Gordon equation. *Int. J. Comput. Math.* **95**, 815–844 (2018).
54. Baccouch, M. Superconvergence of the local discontinuous Galerkin method for the sine-Gordon equation on Cartesian grids. *Appl. Numer. Math.* **113**, 124–155. <https://doi.org/10.1016/j.apnum.2016.11.007> (2017).
55. Baccouch, M. An optimal a posteriori error estimates of the local discontinuous Galerkin method for the second-order wave equation in one space dimension. *Int. J. Numer. Anal. Model.* **14**, 355–380 (2017).
56. Baccouch, M. A posteriori error analysis of the discontinuous Galerkin method for two-dimensional linear hyperbolic conservation laws on Cartesian grids. *J. Sci. Comput.* **68**, 945–974. <https://doi.org/10.1007/s10915-016-0166-0> (2016).
57. Baccouch, M. Optimal a posteriori error estimates of the local discontinuous Galerkin method for convection-diffusion problems in one space dimension. *J. Comput. Math.* **34**, 511–531. <https://doi.org/10.4208/jcm.1603-m2015-0317> (2016).
58. Baccouch, M. Analysis of a posteriori error estimates of the discontinuous Galerkin method for nonlinear ordinary differential equations. *Appl. Numer. Math.* **106**, 129–153. <https://doi.org/10.1016/j.apnum.2016.03.008> (2016).
59. Baccouch, M. Recovery-based error estimator for the discontinuous Galerkin method for nonlinear scalar conservation laws in one space dimension. *J. Sci. Comput.* **66**, 459–476. <https://doi.org/10.1007/s10915-015-0030-7> (2016).
60. Baccouch, M. & Temimi, H. Analysis of optimal error estimates and superconvergence of the discontinuous Galerkin method for convection-diffusion problems in one space dimension. *Int. J. Numer. Anal. Model.* **13**, 403–434 (2016).
61. Baccouch, M. & Adjerid, S. Discontinuous Galerkin error estimation for hyperbolic problems on unstructured triangular meshes. *Comput. Methods Appl. Mech. Eng.* **200**, 162–177. <https://doi.org/10.1016/j.cma.2010.08.002> (2011).
62. Cui, J., Gao, F., Sun, Z. & Zhu, P. A posteriori error estimate for discontinuous Galerkin finite element method on polytopal mesh. *Numer. Methods Part. Differ. Equs.* **36**, 601–616. <https://doi.org/10.1002/num.22443> (2020).
63. Jaśkowiec, J. & Pluciński, P. Discontinuous Galerkin method in numerical simulation of two-dimensional thermoelasticity problem with single stabilization parameter. *Adv. Eng. Softw.* **122**, 62–80 (2018).
64. Doman, B. *The Classical Orthogonal Polynomials* (World Scientific Publishing CoWorld Scientific Publishing Co. Pte. Ltd., 2015).
65. Carstensen, C. & Gedicke, J. Robust residual-based a posteriori Arnold–Winther mixed finite element analysis in elasticity. *Comput. Methods Appl. Mech. Eng.* **300**, 245–264. <https://doi.org/10.1016/j.cma.2015.10.001> (2016).

Author contributions

J.J.: proposed the method for the error approximation—prepared the programs in Matlab for the calculations—prepared the examples presented in the article—proposed the article structure - wrote the main manuscript text, prepared revisions of the article J.P.: gave consultations on the error approximation concept—analyzed and verified the error approximation method—discussed the calculation results—reviewed and corrected the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.J.

Reprints and permissions information is available at www.nature.com/reprints.

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



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023