Journal of Computational and Applied Mathematics 362 (2019) 383–399

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

An a posteriori error estimator for the weak Galerkin least-squares finite-element method[☆]

James H. Adler[a], Xiaozhe Hu[a], Lin Mu[b], Xiu Ye[c,*]

[a] *Department of Mathematics, Tufts University, Medford, MA 02155, USA*
[b] *Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831, USA*
[c] *Department of Mathematics, University of Arkansas at Little Rock, Little Rock, AR, USA*

ARTICLE INFO

Article history:
Received 8 February 2018
Received in revised form 27 August 2018

Keywords:
Weak Galerkin
Finite-element methods
Least-squares finite-element methods
Second-order elliptic problems

ABSTRACT

In this paper, we derive an *a posteriori* error estimator for the weak Galerkin least-squares (WG-LS) method applied to the reaction–diffusion equation. We show that this estimator is both reliable and efficient, allowing it to be used for adaptive refinement. Due to the flexibility of the WG-LS discretization, we are able to design a simple and straightforward refinement scheme that is applicable to any shape regular polygonal mesh. Finally, we present numerical experiments that confirm the effectiveness of the estimator, and demonstrate the robustness and efficiency of the proposed adaptive WG-LS approach.

1. Introduction

The weak Galerkin (WG) method [1–3] takes a system of equations (usually first-order partial differential equations (PDEs)) and approximates the differential operators by a weakly-defined derivative in the distribution sense. A WG formulation is derived by replacing the usual derivatives with these weak ones and, as a result, allows for a variety of problems to be solved on general and non-conforming meshes. WG methods provide robust and stable discretizations for various problems, including Stokes' equations [4,5], the Brinkman equations [6], locking-free schemes for linear elasticity [7], and poroelasticity [8], to name a few. Recently, in [9], a least-squares version of the WG method was developed, which uses discontinuous approximating functions on mixed finite-element partitions consisting of arbitrary polygon/polyhedron shapes.

The first-order system least-squares finite-element approach [10,11], on the other hand, takes a system of first-order PDEs and minimizes the residuals in a least-squares sense to obtain their weak formulations. Advantages of the least-squares finite-element method are that the resulting linear systems are symmetric and positive definite and, thus, iterative solution techniques such as multigrid have successfully been applied to solve them (e.g. [12,10,11,13–21]). Another advantage of the least-squares approach is the natural sharp and reliable a posterior error estimate that arises. To illustrate this, consider a first-order system of (linear) PDEs,

$$\mathcal{L}\vec{u} = \vec{f},$$

☆ The work of J. H. Adler and X. Hu was supported in part by National Science Foundation Grant DMS-1620063. The research of L. Mu is based upon work supported in part by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program under award number ERKJE45; and by the Laboratory Directed Research and Development program at the Oak Ridge National Laboratory, which is operated by UT-Battelle, LLC., for the U.S. Department of Energy under Contract DE-AC05-00OR22725. The work of X. Ye was supported in part by National Science Foundation Grant DMS-1620016.
* Corresponding author.
E-mail addresses: James.Adler@tufts.edu (J.H. Adler), Xiaozhe.Hu@tufts.edu (X. Hu), mul1@ornl.gov (L. Mu), xxye@ualr.edu (X. Ye).

https://doi.org/10.1016/j.cam.2018.09.049
0377-0427/

and define the least-squares functional,

$$\mathcal{G}(\vec{u}; \vec{f}) = \|\mathcal{L}\vec{u} - \vec{f}\|.$$

In order to show well-posedness, this functional is proven to be elliptic with respect to the functional space of interest (most often H^1),

$$c_0\|\vec{u}\|_{\mathcal{V}} \leq \mathcal{G}(\vec{u}; \vec{0}) \leq c_1\|\vec{u}\|_{\mathcal{V}} \qquad \forall\ \vec{u} \in \mathcal{V},$$

including at the discrete level. As a result, the functional itself is the a posteriori error estimator and can be utilized in an adaptive refinement scheme.

Adaptive refinement schemes have been used extensively to approximate solutions of PDEs containing local features [22–28]. The approach is simple,

$$\texttt{SOLVE} \rightarrow \texttt{ESTIMATE} \rightarrow \texttt{MARK} \rightarrow \texttt{REFINE}, \tag{1}$$

where the a posteriori error estimate is used to `Estimate` the error in the solution. Then, a certain percentage of the elements that contain the highest amount of approximate error are `Mark`ed for refinement. Other approaches can be used that optimize the marking portion of the algorithm taking into account both the sharpness and reliability of the estimator [29].

While the WG least-squares (WG-LS) approach can increase the flexibility of a least-squares method by making it applicable to a wide range of PDEs that are discretized on non-conforming or non-standard meshes, the natural a posteriori error estimator might be lost, since the discretization is no longer a direct result of a minimization problem. However, the goal of this paper is to show that such an a posteriori error estimator is obtained, allowing us to design an adaptive WG-LS method.

Consider the model problem satisfying

$$-\nabla\cdot(A\nabla u) + cu = f, \quad \text{in } \Omega, \tag{2}$$

$$u = g, \quad \text{on } \partial\Omega, \tag{3}$$

where $c \geq 0$ is a function on Ω, which is a polytopal domain in $\mathbb{R}^d$ (polygonal or polyhedral domain for $d = 2,3$), ∇u denotes the gradient of the function u, and the tensor A is uniformly symmetric positive-definite. This PDE (2)–(3) has a wide range of applications. For example, Darcy's Law models the flow and transport in porous media, and A is then the permeability of the medium. The solution of the model problem (2) and (3) may have large derivatives or singularities due to the diffusion tensor A, the domain Ω, and/or the right-hand side f, which makes it difficult to approximate using uniform meshes. An extremely small global mesh size is usually required in order to resolve the local singularities and, hence, standard finite-element methods cannot achieve optimal convergence on these meshes. On the other hand, adaptive schemes refine locally near the singularities allowing for more efficiency of the overall method.

In [30], a reliable and efficient a posteriori error estimate for the *standard* WG finite-element method applied to the model problem (2) and (3) was derived based on a residual-type a posteriori error estimator. In this work, we propose a similar but different a posteriori error estimator for the WG-LS problem applied to (2)–(3), and prove that it gives both a lower and an upper bound on the true errors. This implies that the error estimator is reliable and efficient. Then, we design an adaptive WG-LS algorithm based on the a posteriori error estimator. Due to the nature of WG, this adaptive algorithm works on general polygonal meshes and, thus, adaptive refinement is done in a straightforward way. We note that adaptive algorithms on polygonal meshes can also be done based on discontinuous Galerkin finite-element methods, see [31]. Using our approach with a WG scheme, in comparison however, there is no need for the handling of hanging nodes.

This paper is organized as follows. In Section 2, we give a brief background on the WG-LS approach applied to the model problem above. In Section 3, we define and prove the a posteriori error estimate for the model problem, and then use this to describe an adaptive algorithm based on WG-LS in Section 4. We give several examples showing the performance of adaptive refinement with the WG-LS approach in Section 5 and, finally, a brief discussion of the results is given in Section 6.

2. The Weak Galerkin Least-Squares method

First, we recall the WG-LS method developed in [9] for solving the model problem (2)–(3). The WG-LS method follows the WG principles, but applies a least-squares finite-element approach to the first-order mixed form of (2)–(3):

$$\mathbf{q} + A\nabla u = 0, \ \text{in } \Omega \tag{4}$$

$$\nabla\cdot\mathbf{q} + cu = f, \ \text{in } \Omega \tag{5}$$

$$u = g, \ \text{on } \partial\Omega. \tag{6}$$

Let $\mathcal{T}_h$ be a partition of a domain Ω consisting of polygons in two dimensions or polyhedra in three dimensions, satisfying a set of conditions specified in [3]. Denote by $\mathcal{E}_h$ the set of all edges or flat faces in $\mathcal{T}_h$, and let $\mathcal{E}_h^0 = \mathcal{E}_h\backslash\partial\Omega$ be the set of all interior edges or flat faces. Let Γ_h be the subset of $\mathcal{E}_h$ of all edges on Γ. For every element $T \in \mathcal{T}_h$, we denote by h_T its diameter and mesh size $h = \max_{T\in\mathcal{T}_h} h_T$ for $\mathcal{T}_h$. For convenience, we introduce a set of normal directions on $\mathcal{E}_h$ as follows,

$$\mathcal{D}_h = \{\mathbf{n}_e :\ \mathbf{n}_e \text{ is unit and normal to } e,\ e \in \mathcal{E}_h\}. \tag{7}$$

Next, we introduce two WG finite-element spaces, V_h and Σ_h, for u and $\mathbf{q}$, respectively,

$$V_h = \{v = \{v_0, v_b\} : \ v_0|_T \in P_{k+1}(T), v_b|_{\partial T} \in P_{k+1}(\partial T)\}$$
$$\Sigma_h = \{\boldsymbol{\sigma} = \{\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_b\} : \ \boldsymbol{\sigma}_0|_T \in [P_k(T)]^d, \boldsymbol{\sigma}_b|_e = \sigma_b \mathbf{n}_e, \sigma_b|_e \in P_k(e), \ e \in \partial T\}.$$

In addition, we define $V_h^0 \subset V_h$ as $V_h^0 = \{v \in V_h : \ v_b = 0 \text{ on } \partial\Omega\}$. For $\boldsymbol{\sigma} = \{\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_b\} \in \Sigma_h$, the weak divergence $\nabla_w \cdot \boldsymbol{\sigma} \in P_k(T)$ is defined on each element T,

$$(\nabla_w \cdot \boldsymbol{\sigma}, v)_T = -(\boldsymbol{\sigma}_0, \nabla v)_T + \langle \boldsymbol{\sigma}_b \cdot \mathbf{n}, v\rangle_{\partial T}, \qquad \forall v \in P_k(T). \tag{8}$$

where $(\cdot,\cdot)_T$ denotes the L^2 inner product defined on element T and $\langle\cdot,\cdot\rangle$ denotes the duality pair defined on ∂T. For $v = \{v_0, v_b\} \in V_h$, the weak gradient $\nabla_w v \in [P_k(T)]^d$ is defined on each element T by

$$(\nabla_w v, \tau)_T = -(v_0, \ \nabla \cdot \tau)_T + \langle v_b, \ \tau \cdot \mathbf{n}\rangle_{\partial T}, \qquad \forall \tau \in [P_k(T)]^d. \tag{9}$$

With these definitions, then, the WG-LS scheme, for the model problem (2)–(3), seeks $u_h = \{u_0, u_b\} \in V_h$ and $\mathbf{q}_h = \{\mathbf{q}_0, \mathbf{q}_b\} \in \Sigma_h$, such that $u_b = Q_b g$ (the L^2 projection from $L^2(e)$ to $P_{k+1}(e)$ of g on $\partial\Omega$) and,

$$a(u_h, \mathbf{q}_h; v, \boldsymbol{\sigma}) = (f, \ \nabla_w \cdot \boldsymbol{\sigma} + cv_0) \quad \forall (v, \boldsymbol{\sigma}) \in V_h^0 \times \Sigma_h. \tag{10}$$

Here, $a(w, \boldsymbol{\tau}; v, \boldsymbol{\sigma}) = \sum_{T\in\mathcal{T}_h} a_T(w, \boldsymbol{\tau}; v, \boldsymbol{\sigma})$ with

$$\begin{aligned}
a_T(w, \boldsymbol{\tau}; v, \boldsymbol{\sigma}) &= (\nabla_w \cdot \boldsymbol{\tau} + cw_0, \ \nabla_w \cdot \boldsymbol{\sigma} + cv_0)_T \\
&\quad + (\boldsymbol{\tau}_0 + A\nabla_w w, \ \boldsymbol{\sigma}_0 + A\nabla_w v)_T + s_{1,T}(w, v) + s_{2,T}(\boldsymbol{\tau}, \boldsymbol{\sigma}), \\
s_{1,T}(w, v) &= h_T^{-1}\langle w_0 - w_b, \ v_0 - v_b\rangle_{\partial T}, \\
s_{2,T}(\boldsymbol{\tau}, \boldsymbol{\sigma}) &= h_T\langle(\boldsymbol{\tau}_0 - \boldsymbol{\tau}_b)\cdot\mathbf{n}, \ (\boldsymbol{\sigma}_0 - \boldsymbol{\sigma}_b)\cdot\mathbf{n}\rangle_{\partial T}.
\end{aligned}$$

Optimal error estimates for the WG-LS scheme (10) are then derived, considering the following norms in V_h and Σ_h, respectively,

$$|||v|||_V^2 = \sum_{T\in\mathcal{T}_h} \|\nabla_w v\|_T^2 + s_1(v, v), \quad |||\boldsymbol{\sigma}|||_\Sigma^2 = \sum_{T\in\mathcal{T}_h} \|\nabla_w \cdot \boldsymbol{\sigma}\|_T^2 + \|\boldsymbol{\sigma}_0\|^2 + s_2(\boldsymbol{\sigma}, \boldsymbol{\sigma}),$$

where $\|v\|_T^2 := \|(v, v)_T\|$ and $\|v\|^2 := (v, v), (v, v)$ is the L^2 inner product defined on Ω. The following theorem summarizes the results. For more details on the method, we refer the reader to [9].

Theorem 1 ([9]). *First, define $Q_h u = \{Q_0 u, Q_b u\} \in V_h$ with Q_0 and Q_b being the L^2 projections from $L^2(T)$ to $P_{k+1}(T)$ and from $L^2(e)$ to $P_{k+1}(e)$, respectively. Then, define $\mathbf{Q}_h\mathbf{q} = \{\mathbf{Q}_0\mathbf{q}, (\mathbf{Q}_b\mathbf{q}\cdot\mathbf{n}_e)\mathbf{n}_e\} \in \Sigma_h$ with $\mathbf{Q}_0$ and $\mathbf{Q}_b$ being the L^2 projections from $[L^2(T)]^d$ to $[P_k(T)]^d$ and from $[L^2(e)]^d$ to $[P_k(e)]^d$, respectively. Let $(u_h, \mathbf{q}_h) \in V_h \times \Sigma_h$ be the WG-LS finite-element solution of the problem (4)–(6) arising from (10). Assume the exact solution $u \in H^{k+2}(\Omega)$ and $\mathbf{q} \in [H^{k+1}(\Omega)]^d$, then there exists a constant C such that*

$$|||u_h - Q_h u|||_V + |||\mathbf{q}_h - \mathbf{Q}_h\mathbf{q}|||_\Sigma \le Ch^{k+1}(\|u\|_{k+2} + \|\mathbf{q}\|_{k+1}). \tag{11}$$

The error estimate, (11), gives an a priori error estimate for the WG-LS scheme (10). Note that such an estimate only holds when the solution is sufficiently smooth ($u \in H^{k+2}(\Omega)$) and, therefore, cannot be applied for problems with singular solutions. In this paper, we derive a residual-type *a posteriori* error estimate for (10) without assuming such high smoothness on the solution.

3. A posteriori error estimate

Let $u_h = \{u_0, u_b\}$ and $\mathbf{q}_h = \{\mathbf{q}_0, \mathbf{q}_b\}$ be the solution of the WG-LS scheme (10) for the model problem (4)–(6). For simplicity, assume that the coefficient A is the identity and that the function c is bounded away from zero, $c \ge \xi > 0$, for some constant, ξ. The analysis below can easily be extended to more general cases of coefficients with simple and straightforward modifications.

We obtain an a posteriori error estimator by using a residual-based approach. On each $T \in \mathcal{T}_h$, define a local estimator, η_T,

$$\eta_T^2 = \|\nabla_w \cdot \mathbf{q}_h + cu_0 - f\|_T^2 + \|\mathbf{q}_0 + \nabla_w u_h\|_T^2 + s_{1,T}(u_h, u_h) + s_{2,T}(\mathbf{q}_h, \mathbf{q}_h). \tag{12}$$

Note that the first two terms correspond to the residuals of (10) on T and the last two terms are the stabilization terms which appear in (10). Based on η_T, we define a global a posteriori error estimator, η, as follows:

$$\eta^2 = \sum_{T\in\mathcal{T}_h} \eta_T^2. \tag{13}$$

In the rest of this section, we show that the error estimator defined above is both efficient and reliable, i.e., it provides both lower and upper bounds for the true error.

For the sake of simplicity, we introduce the following notation for measuring the true errors,

$$\|u-u_h\|_h^2=\sum_{T\in\mathcal{T}_h}\|u-u_h\|_{h,T}^2 \quad \text{and} \quad \|\mathbf{q}-\mathbf{q}_h\|_h^2=\sum_{T\in\mathcal{T}_h}\|\mathbf{q}-\mathbf{q}_h\|_{h,T}^2,$$

where

$$\begin{aligned}\|u-u_h\|_{h,T}^2 &= \|\nabla u-\nabla_w u_h\|_T^2+\|u-u_0\|_T^2+s_{1,T}(u_h,u_h),\\ \|\mathbf{q}-\mathbf{q}_h\|_{h,T}^2 &= \|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}\|_T^2+s_{2,T}(\mathbf{q}_h,\mathbf{q}_h).\end{aligned}$$

In addition, the following trace inequality (see [3] for details) plays an important role in the analysis. For any function $\varphi\in H^1(T)$,

$$\|\varphi\|_e^2\le C\left(h_T^{-1}\|\varphi\|_T^2+h_T\|\nabla\varphi\|_T^2\right), \tag{14}$$

where $\|v\|_e^2:=(v,v)_e$ with $(\cdot,\cdot)_e$ being the L^2 inner product defined on edge e.

Before proving the efficiency and reliability of the a posterior error estimator, we introduce a few lemmas to help with the analysis. The first relates the errors in the standard derivatives to the errors in the weak derivatives.

Lemma 3.1. *Let* $(u,\mathbf{q})$ *and* $(u_h,\mathbf{q}_h)$ *be the solutions of* (4)–(6) *and* (10) *respectively, then*

$$\|\nabla u-\nabla u_0\|_T \le \|\nabla u-\nabla_w u_h\|_T+s_{1,T}(u_h,u_h)^{\frac12}, \tag{15}$$

$$\|\nabla\cdot\mathbf{q}-\nabla\cdot\mathbf{q}_0\|_T \le \|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T+s_{2,T}(\mathbf{q}_h,\mathbf{q}_h)^{\frac12}. \tag{16}$$

Proof. By the triangle inequality,

$$\|\nabla u-\nabla u_0\|_T\le\|\nabla u-\nabla_w u_h\|_T+\|\nabla_w u_h-\nabla u_0\|_T. \tag{17}$$

Then, the second term expands to

$$\|\nabla_w u_h-\nabla u_0\|_T^2=(\nabla_w u_h,\nabla_w u_h)_T-2(\nabla_w u_h,\nabla u_0)_T+(\nabla u_0,\nabla u_0)_T. \tag{18}$$

Using the definition of the weak gradient, (9),

$$\begin{aligned}(\nabla_w u_h,\nabla_w u_h)_T &= -(u_0,\nabla\cdot\nabla_w u_h)_T+\langle u_b,\nabla_w u_h\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\nabla u_0,\nabla_w u_h)_T-\langle u_0-u_b,\nabla_w u_h\cdot\mathbf{n}\rangle_{\partial T},\end{aligned}$$

and

$$\begin{aligned}(\nabla u_0,\nabla u_0)_T &= -(u_0,\nabla\cdot\nabla u_0)_T+\langle u_0,\nabla u_0\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\nabla u_0,\nabla_w u_h)_T+\langle u_0-u_b,\nabla u_0\cdot\mathbf{n}\rangle_{\partial T}.\end{aligned}$$

Substituting the two equations above back into (18) and using (14) and the inverse inequality,

$$\|\nabla_w u_h-\nabla u_0\|_T^2=\langle u_0-u_b,(\nabla u_0-\nabla_w u_h)\cdot\mathbf{n}\rangle_{\partial T}\le s_{1,T}(u_h,u_h)^{\frac12}\|\nabla_w u_h-\nabla u_0\|_T,$$

which implies $\|\nabla_w u_h-\nabla u_0\|_T\le s_{1,T}(u_h,u_h)^{1/2}$. Together with (17), (15) follows directly. Inequality (16) can be derived in the same fashion. □

Lemma 3.2. *Let* u *and* $\mathbf{q}$ *be the solution of* (4)–(6) *and* $u_h=\{u_0,u_b\}$ *and* $\mathbf{q}_h=\{\mathbf{q}_0,\mathbf{q}_b\}$ *be the solutions of* (10). *Then, the following equality holds,*

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\mathbf{q}-\mathbf{q}_0,\nabla u-\nabla_w u_h)_T &= \sum_{T\in\mathcal{T}_h}((\nabla_w\cdot\mathbf{q}_h+cu_0-f,u-u_0)_T\\ &\quad +c(u-u_0,u-u_0)_T)\\ &\quad +\ell_1(u,\mathbf{q}_h)-\ell_2(u_h,\mathbf{q})-\ell_3(u_h,\mathbf{q}_h),\end{aligned} \tag{19}$$

where

$$\begin{aligned}\ell_1(u,\mathbf{q}_h) &= \sum_{T\in\mathcal{T}_h}\langle Q_0u-u,(\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_{\partial T},\\ \ell_2(u_h,\mathbf{q}) &= \sum_{T\in\mathcal{T}_h}\langle u_0-u_b,(\mathbf{q}-\mathbf{Q}_0\mathbf{q})\cdot\mathbf{n}\rangle_{\partial T},\\ \ell_3(u_h,\mathbf{q}_h) &= \sum_{T\in\mathcal{T}_h}\langle u_0-u_b,(\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_{\partial T}.\end{aligned}$$

Proof. On each element T,

$$(\mathbf{q}-\mathbf{q}_0, \nabla u - \nabla_w u_h)_T = (\mathbf{q}-\mathbf{q}_0, \nabla u)_T - (\mathbf{q}-\mathbf{q}_0, \nabla_w u_h)_T. \tag{20}$$

Using integration by parts and (8),

$$\begin{aligned}(\nabla\cdot\mathbf{q}_0, Q_0 u)_T &= -(\mathbf{q}_0, \nabla Q_0 u)_T + \langle Q_0 u, \mathbf{q}_0\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\nabla_w\cdot\mathbf{q}_h, u)_T + \langle Q_0 u, (\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_{\partial T}.\end{aligned} \tag{21}$$

Then, using integration by parts, (21), and the fact that

$$\sum_{T\in\mathcal{T}_h}\langle u, \mathbf{q}_b\cdot\mathbf{n}\rangle_{\partial T} = \sum_{T\in\mathcal{T}_h}\langle u, \mathbf{q}\cdot\mathbf{n}\rangle_{\partial T} = 0,$$

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\mathbf{q}-\mathbf{q}_0, \nabla u)_T &= \sum_{T\in\mathcal{T}_h}\left(-(\nabla\cdot(\mathbf{q}-\mathbf{q}_0), u)_T + \langle u, (\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}\right)\\ &= \sum_{T\in\mathcal{T}_h}\left(-(\nabla\cdot\mathbf{q}, u)_T + (\nabla\cdot\mathbf{q}_0, u)_T + \langle u, (\mathbf{q}_b-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}\right)\\ &= -\sum_{T\in\mathcal{T}_h}(\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h, u)_T + \ell_1(u,\mathbf{q}_h).\end{aligned} \tag{22}$$

Next, using (9), (8), and integration by parts,

$$\begin{aligned}(\mathbf{Q}_0\mathbf{q}, \nabla_w u_h)_T &= -(u_0, \nabla\cdot\mathbf{Q}_0\mathbf{q})_T + \langle u_b, \mathbf{Q}_0\mathbf{q}\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\nabla u_0, \mathbf{q})_T - \langle u_0-u_b, \mathbf{Q}_0\mathbf{q}\cdot\mathbf{n}\rangle_{\partial T}\\ &= -(\nabla\cdot\mathbf{q}, u_0)_T + \langle u_0-u_b, (\mathbf{q}-\mathbf{Q}_0\mathbf{q})\cdot\mathbf{n}\rangle_{\partial T},\end{aligned} \tag{23}$$

and

$$\begin{aligned}(\mathbf{q}_0, \nabla_w u_h)_T &= -(u_0, \nabla\cdot\mathbf{q}_0)_T + \langle u_b, \mathbf{q}_0\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\nabla u_0, \mathbf{q}_0)_T - \langle u_0-u_b, \mathbf{q}_0\cdot\mathbf{n}\rangle_{\partial T}\\ &= -(\nabla_w\cdot\mathbf{q}_h, u_0)_T - \langle u_0-u_b, (\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_{\partial T}.\end{aligned} \tag{24}$$

Combining (23) and (24), we get

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\mathbf{q}-\mathbf{q}_0, \nabla_w u_h)_T &= \sum_{T\in\mathcal{T}_h}\left((\mathbf{q}, \nabla_w u_h)_T - (\mathbf{q}_0, \nabla_w u_h)_T\right)\\ &= -\sum_{T\in\mathcal{T}_h}(\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h, u_0)_T + \ell_2(u_h,\mathbf{q}) + \ell_3(u_h,\mathbf{q}_h).\end{aligned} \tag{25}$$

Finally, substituting (22) and (25) back into (20), we arrive at,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\mathbf{q}-\mathbf{q}_0, \nabla u-\nabla_w u_h)_T &= \sum_{T\in\mathcal{T}_h}\left((\mathbf{q}-\mathbf{q}_0, \nabla u)_T - (\mathbf{q}-\mathbf{q}_0, \nabla_w u_h)_T\right)\\ &= -\sum_{T\in\mathcal{T}_h}(\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h, u-u_0)_T + \ell_1(u,\mathbf{q}_h)\\ &\quad - \ell_2(u_h,\mathbf{q}) - \ell_3(u_h,\mathbf{q}_h).\end{aligned} \tag{26}$$

Then, (19) is obtained directly from (5), which completes the proof. □

Lemma 3.3. *Let* $(u,\mathbf{q})$ *and* $(u_h,\mathbf{q}_h)$ *be the solutions of* (4)–(6) *and* (10)*, respectively. For* $0<\epsilon<1$*, there exists constant* C *such that,*

$$|\ell_1(u,\mathbf{q}_h)| \le C\epsilon^{-1}\eta^2 + \epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2, \tag{27}$$

$$|\ell_2(u_h,\mathbf{q})| \le C\epsilon^{-1}\eta^2 + \epsilon\sum_{T\in\mathcal{T}_h}\left(\|\mathbf{q}-\mathbf{q}_0\|_T^2 + \|(\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h)\|_T^2\right), \tag{28}$$

$$|\ell_3(u_h,\mathbf{q}_h)| \le C\eta^2. \tag{29}$$

Proof. By the trace inequality (14), (15), and the Cauchy–Schwarz inequality, (27) is derived as follows,

$$\begin{aligned}|\ell_1(u,\mathbf{q}_h)| &= |\sum_{T\in\mathcal{T}_h}\langle Q_0(u-u_0)+u_0-u,(\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_{\partial T}|\\ &\le \sum_{T\in\mathcal{T}_h}\left(h^{-1}\|Q_0(u-u_0)-(u-u_0)\|_T^2+h|\nabla Q_0(u-u_0)-(u-u_0)|_T^2\right)^{1/2}s_{2,T}(\mathbf{q}_h,\mathbf{q}_h)^{1/2}\\ &\le C\sum_{T\in\mathcal{T}_h}\|\nabla(u-u_0)\|_T\, s_{2,T}(\mathbf{q}_h,\mathbf{q}_h)^{\frac{1}{2}}\\ &\le C\sum_{T\in\mathcal{T}_h}\left(\|\nabla u-\nabla_w u_h\|_T^2+s_{1,T}(u_h,u_h)\right)s_{2,T}(\mathbf{q}_h,\mathbf{q}_h)^{\frac{1}{2}}\\ &\le C\epsilon^{-1}\eta^2+\epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2.\end{aligned}$$

Considering ℓ_2, note that,

$$\begin{aligned}|\ell_2(u_h,\mathbf{q})| &= |\sum_{T\in\mathcal{T}_h}\langle u_0-u_b,(\mathbf{q}-\mathbf{q}_0+\mathbf{q}_0-\mathbf{Q}_0\mathbf{q})\cdot\mathbf{n}\rangle_{\partial T}|\\ &\le \sum_{T\in\mathcal{T}_h}|\langle u_0-u_b,(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}|+|\langle u_0-u_b,\mathbf{Q}_0(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}|.\end{aligned}\tag{30}$$

Using the Cauchy–Schwarz inequality, the inverse inequality, and the trace inequality, we estimate the two terms on the left-hand side as follows,

$$\begin{aligned}|\langle u_0-u_b,(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}| &\le \|u_0-u_b\|_{H^{\frac{1}{2}}(\partial T)}\|(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\|_{H^{-\frac{1}{2}}(\partial T)}\\ &\le Cs_{1,T}(u_h,u_h)^{\frac{1}{2}}\left(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot(\mathbf{q}-\mathbf{q}_0)\|_T^2\right)^{\frac{1}{2}},\end{aligned}$$

and

$$\begin{aligned}|\langle u_0-u_b,\mathbf{Q}_0(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\rangle_{\partial T}| &\le h_T^{-\frac{1}{2}}\|u_0-u_b\|_{\partial T}\,h_T\|\mathbf{Q}_0(\mathbf{q}-\mathbf{q}_0)\cdot\mathbf{n}\|_{\partial T}\\ &\le s_{1,T}(u_h,u_h)^{\frac{1}{2}}h_T\|\mathbf{Q}_0(\mathbf{q}-\mathbf{q}_0)\|_{\partial T}\\ &\le Cs_{1,T}(u_h,u_h)^{\frac{1}{2}}\|\mathbf{Q}_0(\mathbf{q}-\mathbf{q}_0)\|_T\\ &\le Cs_{1,T}(u_h,u_h)^{\frac{1}{2}}\|\mathbf{q}-\mathbf{q}_0\|_T.\end{aligned}$$

Then, substituting the above two inequalities back into (30), we obtain,

$$\begin{aligned}|\ell_2(u_h,\mathbf{q})| &\le C\sum_{T\in\mathcal{T}_h}\left(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot(\mathbf{q}-\mathbf{q}_0)\|_T^2\right)^{\frac{1}{2}}s_{1,T}(u_h,u_h)^{\frac{1}{2}}\\ &\le C\epsilon^{-1}\eta^2+\epsilon\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|(\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h)\|_T^2).\end{aligned}$$

Finally, ℓ_3 is bounded as follows,

$$\begin{aligned}|\ell_3(u_h,\mathbf{q}_h)| &\le \sum_{T\in\mathcal{T}_h}|\langle u_0-u_b,(\mathbf{q}_0-\mathbf{q}_b)\cdot\mathbf{n}\rangle_T|\\ &\le \sum_{T\in\mathcal{T}_h}s_{1,T}(u_h,u_h)^{\frac{1}{2}}s_{2,T}(\mathbf{q}_h,\mathbf{q}_h)^{\frac{1}{2}}\le\eta^2.\quad\square\end{aligned}$$

Now we are ready to present the main results of the paper. First, we show that the a posteriori error estimator, (12), is reliable.

Theorem 2. *Let u and $\mathbf{q}$ be the solution of (4)–(6) and $u_h=\{u_0,u_b\}$ and $\mathbf{q}_h=\{\mathbf{q}_0,\mathbf{q}_b\}$ be the WG-LS solution of (10). There exists a positive constant C such that,*

$$\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2\le C\sum_{T\in\mathcal{T}_h}\eta_T^2=C\eta^2.\tag{31}$$

Proof. Using (19) and (4), we have

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2 &= \sum_{T\in\mathcal{T}_h}[(\nabla u-\nabla_w u_h+\mathbf{q}-\mathbf{q}_0,\nabla u-\nabla_w u_h)_T\\ &\quad -(\mathbf{q}-\mathbf{q}_0,\nabla u-\nabla_w u_h)_T]\\ &= -\sum_{T\in\mathcal{T}_h}[(\nabla_w u_h+\mathbf{q}_0,\nabla u-\nabla_w u_h)_T\\ &\quad +(\nabla_w\cdot\mathbf{q}_h+cu_0-f,u-u_0)_T+c(u-u_0,u-u_0)_T]\\ &\quad -\ell_1(u,\mathbf{q}_h)+\ell_2(u_h,\mathbf{q})+\ell_3(u_h,\mathbf{q}_h).\end{aligned}$$

This implies,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}\left[\|\nabla u-\nabla_w u_h\|_T^2+c\|u-u_0\|_T^2\right] &= -\sum_{T\in\mathcal{T}_h}[(\nabla_w u_h+\mathbf{q}_0,\nabla u-\nabla_w u_h)_T\\ &\quad +(\nabla_w\cdot\mathbf{q}_h+cu_0-f,u-u_0)_T]\\ &\quad -\ell_1(u,\mathbf{q}_h)+\ell_2(u_h,\mathbf{q})+\ell_3(u_h,\mathbf{q}_h).\end{aligned}$$

Using (27)–(29) and the Cauchy–Schwarz inequality,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\|\nabla u-\nabla_w u_h\|_T^2+c\|u-u_0\|_T^2) &\le C\epsilon^{-1}\sum_{T\in\mathcal{T}_h}\eta_T^2\\ &\quad +2\epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2+\epsilon\sum_{T\in\mathcal{T}_h}c\|u-u_0\|_T^2\\ &\quad +\epsilon\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2).\end{aligned}\tag{32}$$

Similarly,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}\|\mathbf{q}-\mathbf{q}_0\|_T^2 &= \sum_{T\in\mathcal{T}_h}((\mathbf{q}-\mathbf{q}_0+\nabla u-\nabla_w u_h,\mathbf{q}-\mathbf{q}_0)_T-(\nabla u-\nabla_w u_h,\mathbf{q}-\mathbf{q}_0)_T\\ &= -\sum_{T\in\mathcal{T}_h}((\mathbf{q}_0+\nabla_w u_h,\mathbf{q}-\mathbf{q}_0)_T)+(\nabla_w\cdot\mathbf{q}_h+cu_0-f,u-u_0)_T\\ &\quad -c(u-u_0,u-u_0)_T)-\ell_1(u,\mathbf{q}_h)+\ell_2(u_h,\mathbf{q})+\ell_3(u_h,\mathbf{q}_h),\end{aligned}$$

which implies,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+c\|u-u_0\|_T^2) &\le C\epsilon^{-1}\sum_{T\in\mathcal{T}_h}\eta_T^2\\ &\quad +\epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2+\epsilon\sum_{T\in\mathcal{T}_h}c\|u-u_0\|_T^2\\ &\quad +2\epsilon\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2).\end{aligned}\tag{33}$$

In addition, from (32) and (33), we have,

$$\begin{aligned}\sum_{T\in\mathcal{T}_h}\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2 &\le 2\sum_{T\in\mathcal{T}_h}(\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h+c(u-u_0)\|_T^2+c\|u-u_0\|_T^2)\\ &= 2\sum_{T\in\mathcal{T}_h}(\|\nabla_w\cdot\mathbf{q}_h+cu_0-f\|_T^2+c\|u-u_0\|_T^2)\\ &\le C\epsilon^{-1}\sum_{T\in\mathcal{T}_h}\eta_T^2+3\epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2+2\epsilon\sum_{T\in\mathcal{T}_h}c\|u-u_0\|_T^2\\ &\quad +3\epsilon\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2).\end{aligned}\tag{34}$$

Using (32)–(34),

$$
\begin{aligned}
&\sum_{T\in\mathcal{T}_h}\left(\|\nabla u-\nabla_w u_h\|_T^2+c\|u-u_0\|_T^2+\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2\right)\\
&\le C\epsilon^{-1}\sum_{T\in\mathcal{T}_h}\eta_T^2+6\epsilon\sum_{T\in\mathcal{T}_h}\|\nabla u-\nabla_w u_h\|_T^2+4\epsilon\sum_{T\in\mathcal{T}_h}c\|u-u_0\|_T^2\\
&\quad+6\epsilon\sum_{T\in\mathcal{T}_h}(\|\mathbf{q}-\mathbf{q}_0\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}_h\|_T^2).
\end{aligned}
$$

By choosing $\epsilon = 1/12$, we obtain (31) directly, which completes the proof. □

Finally, we show that the a posteriori error estimator is also efficient locally.

Theorem 3. *Let u and $\mathbf{q}$ be the solution of (4)–(6) and $u_h = \{u_0, u_b\}$ and $\mathbf{q}_h = \{\mathbf{q}_0, \mathbf{q}_b\}$ be the solution of (10). There exists a positive constant C such that*

$$\eta_T^2 \le C(\|u-u_h\|_{h,T}^2+\|\mathbf{q}-\mathbf{q}_h\|_{h,T}^2). \tag{35}$$

Proof. Using (5),

$$
\begin{aligned}
\|\nabla_w\cdot\mathbf{q}_h+cu_0-f\|_T &= \|\nabla_w\cdot\mathbf{q}_h+cu_0-\nabla\cdot\mathbf{q}-cu\|_T\\
&\le \|\nabla_w\cdot\mathbf{q}_h-\nabla\cdot\mathbf{q}\|_T+c\|u-u_0\|_T\\
&\le \|u-u_h\|_{h,T}+\|\mathbf{q}-\mathbf{q}_h\|_{h,T}.
\end{aligned}
$$

On the other hand, by (4),

$$
\begin{aligned}
\|\mathbf{q}_0+\nabla_w u_h\|_T &= \|\mathbf{q}_0+\nabla_w u_h-\mathbf{q}+\nabla u\|_T\\
&\le \|\mathbf{q}-\mathbf{q}_0\|_T+\|\nabla u-\nabla_w u_h\|_T\\
&\le \|u-u_h\|_{h,T}+\|\mathbf{q}-\mathbf{q}_h\|_{h,T}.
\end{aligned}
$$

Combining the two estimates above and using the definition of η_T, (12), (35) follows directly. □

Summing over all the element T, we obtain the following global lower bound for the error estimator.

Theorem 4. *Let u and $\mathbf{q}$ be the solution of (4)–(6) and $u_h = \{u_0, u_b\}$ and $\mathbf{q}_h = \{\mathbf{q}_0, \mathbf{q}_b\}$ be the solution of (10). There exists a positive constant C such that*

$$\eta^2 \le C(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2). \tag{36}$$

Remark 3.1. For general symmetric positive definite A, we need to slightly modify the definition of the local error estimator as follows,

$$\eta_T^2 = \|\nabla_w\cdot\mathbf{q}_h+cu_0-f\|_T^2+\|A^{-\frac{1}{2}}\mathbf{q}_0+A^{\frac{1}{2}}\nabla_w u_h\|_T^2+s_{1,T}(u_h,u_h)+s_{2,T}(\mathbf{q}_h,\mathbf{q}_h). \tag{37}$$

where $A^{\frac{1}{2}}$ is the square root of A and $A^{-\frac{1}{2}}$ is its inverse. Moreover, the definition of $\|u-u_h\|_{h,T}^2$ and $\|\mathbf{q}-\mathbf{q}_h\|_{h,T}^2$ is modified as follows,

$$
\begin{aligned}
\|u-u_h\|_{h,T}^2 &= \|A^{\frac{1}{2}}(\nabla u-\nabla_w u_h)\|_T^2+\|u-u_0\|_T^2+s_{1,T}(u_h,u_h),\\
\|\mathbf{q}-\mathbf{q}_h\|_{h,T}^2 &= \|A^{-\frac{1}{2}}(\mathbf{q}-\mathbf{q}_0)\|_T^2+\|\nabla\cdot\mathbf{q}-\nabla_w\cdot\mathbf{q}\|_T^2+s_{2,T}(\mathbf{q}_h,\mathbf{q}_h).
\end{aligned}
$$

Then the analysis can be carried out in a similar manner with a slight modification and the reliability and efficiency of the a posteriori error estimator still hold. The main difference is that the constant C in the reliability bound (31) depends on $\max\{\lambda_{\max}(A), \lambda_{\min}^{-1}(A)\}$, where $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ are the extreme eigenvalues of A, and the constant C in the efficiency bound (35) is independent of A.

Theorems 2, 3, and 4, and more specifically estimates (31), (35), and (36), show that the a posteriori error estimator is both efficient and reliable. We now use this fact to develop an adaptive refinement algorithm.

4. The adaptive WG-LS method

Following [32], for the adaptive approximation of (4)–(6), we consider a loop of the form (1). In particular, assume that an initial mesh $\mathcal{T}_0$, a parameter $\theta \in (0, 1]$, and a targeted tolerance ε are given. Let u_ℓ and $\mathbf{q}_\ell$ denote the approximate solutions to (10) on the ℓth level adaptive mesh, $\mathcal{T}_\ell$, and let η_T^ℓ and η_ℓ denote the local and global a posteriori error estimator on mesh $\mathcal{T}_\ell$, respectively. Then, the adaptive algorithm is summarized in Algorithm 1.

Algorithm 1 Adaptive WG-LS Method

$[u_J, \mathbf{q}_J] = \texttt{AFEM}(\mathcal{T}_0, \theta, \varepsilon)$

Set $\ell = 0$
loop
  $[u_\ell, \mathbf{q}_\ell] = \texttt{SOLVE}(\mathcal{T}_\ell)$
  $[\{\eta_T^\ell\}_{T\in\mathcal{T}_\ell}, \eta^\ell] = \texttt{ESTIMATE}(u_\ell, \mathbf{q}_\ell \mathcal{T}_\ell)$
  if $\eta^\ell \le \varepsilon$ **then**
    $J = \ell$ and $u_J := u_\ell,\ \mathbf{q}_J = \mathbf{q}_\ell$.
    return
  end if
  $\mathcal{M}_\ell = \texttt{MARK}(\mathcal{T}_\ell, \{\eta_T^\ell\}_{T\in\mathcal{T}_\ell}, \eta^\ell, \theta)$
  $\mathcal{T}_{\ell+1} = \texttt{REFINE}(\mathcal{T}_\ell, \mathcal{M}_\ell)$
  $\ell = \ell + 1$
end loop

As shown, the adaptive algorithm involves four different modules, i.e., SOLVE, ESTIMATE, MARK, and REFINE. In the next few sections, we introduce and define these four modules used in the adaptive loop.

4.1. *SOLVE* Module

The SOLVE module takes the current adaptive grid as input and outputs the corresponding WG-LS approximation to the problem. This stage may involve a variety of other numerical techniques needed to discretize the problem and to solve the linear systems. The detailed module with our specific approach is shown in Algorithm 2.

Algorithm 2 SOLVE module

$[u_\ell, \mathbf{q}_\ell] = \texttt{SOLVE}(\mathcal{T}_\ell)$

On current grid, $\mathcal{T}_\ell$, assemble the linear system of equations corresponding to the WG-LS scheme (10).
Solve the linear system for u_ℓ and $\mathbf{q}_\ell$ (in this work, MATLAB's built-in "\" function is used, i.e., a sparse direct solver).

4.2. *ESTIMATE* Module

Given an adaptive mesh and the WG-LS approximations, u_ℓ and $\mathbf{q}_\ell$, the ESTIMATE module computes the a posteriori error estimator, η_T^ℓ, on each element $T \in \mathcal{T}_\ell$. Here, we use the local estimator (12) and the algorithm is shown in Algorithm 3.

Algorithm 3 ESTIMATE Module

$[\{\eta_T^\ell\}_{T\in\mathcal{T}_\ell}, \eta^\ell] = \texttt{ESTIMATE}(u_\ell, \mathbf{q}_\ell, \mathcal{T}_\ell)$

Compute η_T^ℓ on each element $T \in \mathcal{T}_\ell$ using (12).
Compute η^ℓ based on (13).

4.3. *MARK* Module

The MARK module selects elements $T \in \mathcal{T}_\ell$ whose local error indicator η_T^ℓ is relatively large and which need to be refined. In this work, we use the Döflers marking strategy [32], and this module is shown in Algorithm 4.

Algorithm 4 MARK Module

$\mathcal{M}_\ell = \texttt{MARK}(\mathcal{T}_\ell, \{\eta_T^\ell\}_{T\in\mathcal{T}_\ell}, \eta^\ell, \theta)$

Choose a subset $\mathcal{M}_\ell \subset \mathcal{T}_\ell$ such that

$$\eta^\ell(\mathcal{M}_\ell) \le \theta\eta^\ell, \tag{38}$$

where $\eta^\ell(\mathcal{M}_\ell) := \left(\sum_{T\in\mathcal{M}_\ell} \eta_T^\ell\right)^{\frac{1}{2}}$.

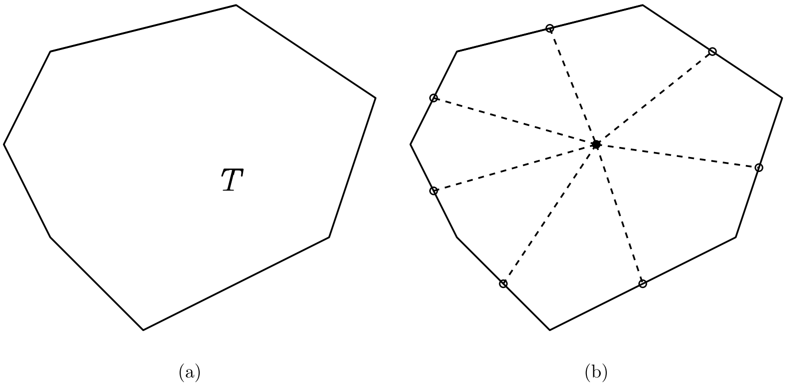


Fig. 1. Refinement of polygonal mesh: (a) Original polygon; (b) After refinement.

Here, we require the parameter $\theta \in (0, 1]$. While the choice of $\mathcal{M}_\ell$ is not unique, in practice, we choose the size of the subset $\mathcal{M}_\ell$ to be as small as possible. Therefore, we use a greedy approach, where we rank the elements, T, according to the error indicators, η_T^ℓ, from largest to smallest. Then, we choose the elements with the largest error, so that condition (38) is satisfied using the minimal number of elements.

4.4. REFINE Module

The final module, REFINE, refines the marked elements $T \in \mathcal{T}_\ell$ and outputs a newly refined mesh $\mathcal{T}_{\ell+1}$. Traditionally, for simplicial meshes (triangles in 2D and tetrahedrons in 3D), sophisticated refinement procedures, such as newest vertex bisection, need to be used in order to keep the shape regularity of the elements and eliminate any hanging nodes that are introduced during refinement. Those refinement algorithms are usually quite complicated and not easy to implement (see [32] for details). However, one of the advantages of the WG-LS scheme is that it allows for the usage of more general polygonal meshes, where hanging nodes are not an issue. This gives more freedom and flexibility on how to perform the refinement of an element without worrying about the conformity of the meshes. Therefore, for this work, we use a straightforward approach to refine a polygonal element. More precisely, on one element, T, the refinement is performed by connecting the barycenter of the polygon and the middle point of each edge on the boundary. See Fig. 1. Because hanging nodes are allowed, no further steps are needed to eliminate them. Therefore, the union of the new elements obtained from refining $T \in \mathcal{M}_\ell$, along with the elements $T \in \mathcal{T}_\ell \backslash \mathcal{M}_\ell$ gives the new adaptive mesh $\mathcal{T}_{\ell+1}$. This is summarized in Algorithm 5.

Algorithm 5 REFINE Module

$\mathcal{T}_{\ell+1} = \texttt{REFINE}(\mathcal{T}_\ell, \mathcal{M}_\ell)$
1: **for** $T \in \mathcal{M}_\ell$ **do**
2: refine T as shown in Figure 1.
3: **end for**
4: Combine all new elements and subset $\mathcal{T}_\ell \backslash \mathcal{M}_l$ to construct the new mesh $\mathcal{T}_{\ell+1}$.

5. Numerical examples

Next, we perform several numerical tests to validate the theoretical results using the overall adaptive WG-LS algorithm, Algorithm 1, based on the a posteriori error estimator (13). The tests are implemented in MATLAB based on iFEM [33] and the polygonal meshes used in the experiments are produced by PolyMesher [34].

5.1. *Efficiency and reliability of a posteriori error estimator η*

In this subsection, we show the effectiveness of the proposed a posteriori error estimator η by verifying its efficiency and reliability, i.e., Theorems 2 and 3. More precisely, we define the *effective index* as follows

$$\text{eff-index} = \frac{\eta}{\left(\|u - u_h\|_h^2 + \|\mathbf{q} - \mathbf{q}_h\|_h^2\right)^{\frac{1}{2}}}, \tag{39}$$

which is the ratio between the a posteriori error estimator and the true error. According to the reliability result (31) and the efficiency result (35), we expect the effective index to be bounded from both above and below. In fact, as is shown later,

Table 1
Efficiency and reliability tests for Example 5.1.

h	$\left(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2\right)^{1/2}$	Rate	η	Rate	Eff-index
Triangular mesh					
1/4	8.3451e−02		7.5531e−02		0.91
1/8	3.9910e−02	1.06	3.7073e−02	1.03	0.93
1/16	1.9702e−02	1.02	1.8436e−02	1.01	0.94
1/32	9.8179e−03	1.00	9.2051e−03	1.00	0.94
1/64	4.9048e−03	1.00	4.6009e−03	1.00	0.94
1/128	2.4519e−03	1.00	2.3002e−03	1.00	0.94
Rectangular mesh					
1/4	9.5636E−02		9.2179E−02		0.96
1/8	4.7251E−02	1.02	4.6711E−02	0.98	0.99
1/16	2.3599E−02	1.00	2.3528E−02	0.99	1.00
1/32	1.1800E−02	1.00	1.1791E−02	1.00	1.00
1/64	5.9001E−03	1.00	5.8990E−03	1.00	1.00
1/128	2.9501E−03	1.00	2.9500E−03	1.00	1.00
Polygonal mesh					
1/4	8.9887E−02		8.4270E−02		0.94
1/8	4.3345E−02	1.05	4.1622E−02	1.02	0.96
1/16	2.1899E−02	0.98	2.1418E−02	0.96	0.98
1/32	1.1078E−02	0.98	1.0901E−02	0.97	0.98
1/64	5.5544E−03	1.00	5.4883E−03	0.99	0.99
1/128	2.7958E−03	0.99	2.7667E−03	0.99	0.99

the effective index of the proposed a posteriori error estimator is very close to 1, demonstrating that η provides a very good estimate of the true error. In order to compute the effective index, we need to know the exact solution of u and $\mathbf{q}$.

Example 5.1. Consider the model problem (4)–(6) with $A = I$, where I is the identity matrix, $c = 2$, and $\Omega = (0, 1)\times(0, 1)$. Choose the following analytical solution,

$$u = x(1-x)y(1-y).$$

Homogeneous Dirichlet boundary conditions are used ($g = 0$) and the right-hand side function f is computed accordingly.

The efficiency and reliability tests are performed on uniform triangular, rectangular, and polygonal meshes with different mesh size h. Here, we use linear elements, i.e., $k = 1$. The true error and the a posteriori error estimator are reported in Table 1 together with the effective index defined as in (39).

From Table 1, we see that both the true error and the a posteriori error estimator converge with the optimal rate, which is 1 for $k = 1$. Not only do they have the same convergence rate, but they are almost the same for all test cases. Similarly, the effective index is close to 1 for all cases as well. This demonstrates that the a posteriori error estimator, η, provides an efficient and reliable estimate of the true error $\left(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2\right)^{1/2}$. This also implies that the constants involved in the efficiency estimate (35) and reliability estimate (31) are both close to 1. Finally, we note that the a posteriori error estimator is also robust with respect to different types of meshes. This property implies that the a posteriori error estimator and the adaptive algorithm based on it can be effectively applied to meshes which consist of different types of elements.

5.2. Adaptive algorithm on general polygonal meshes

Next, we use the adaptive algorithm (Algorithm 1) to solve the model problem (4)–(6) on general polygonal meshes and demonstrate the robustness of the proposed a posteriori error estimator (13), as well as the overall adaptive method. We choose $\theta = 0.2$ in the `MARK` module for all test problems.

Example 5.2. Consider the model problem (4)–(6) with $A = I$, where I is the identity matrix, and $c = 0$ on an L-shaped domain with a reentrant corner, namely $\Omega = (-1, 1)^2\backslash[0, 1)\times(-1, 0]$. The exact solution is $u(r, \phi) = r^{2/3}\sin(\frac{2}{3}\phi)$ in polar coordinates. The right-hand side $f = 0$ and Dirichlet boundary conditions, $u = g$ are used on $\partial\Omega$.

We first solve Example 5.2 by Algorithm 1 with the initial mesh $\mathcal{T}_0$ being a rectangular mesh as shown in Fig. 2(a). Although we start with a uniform rectangular mesh, after applying the refinement procedure discussed in Section 4.4, adaptive meshes $\mathcal{T}_\ell$, $\ell \geq 1$, become general polygonal meshes with possible hanging nodes. The final adaptive meshes with different polynomial degrees ($k = 1$ and $k = 2$) are shown in Fig. 2, and we see that they are general polygonal meshes. More importantly, for both cases, the adaptive meshes are refined mainly around the corner where the singularity occurs. In addition, the refinements are more focused around the corner singularities when $k = 2$. This is expected since away from the corner, the solution is smooth and high-order schemes provide a better approximation. In Fig. 3, we plot the true error

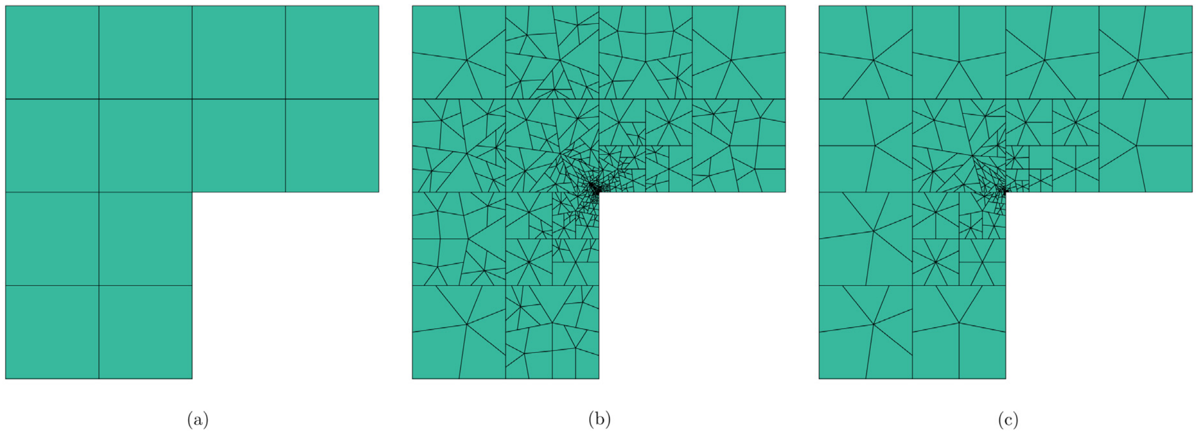


Fig. 2. Example 5.2: (a) Initial rectangular mesh; (b) Final mesh with linear elements, $k = 1$; (c) Final mesh with quadratic elements, $k = 2$.

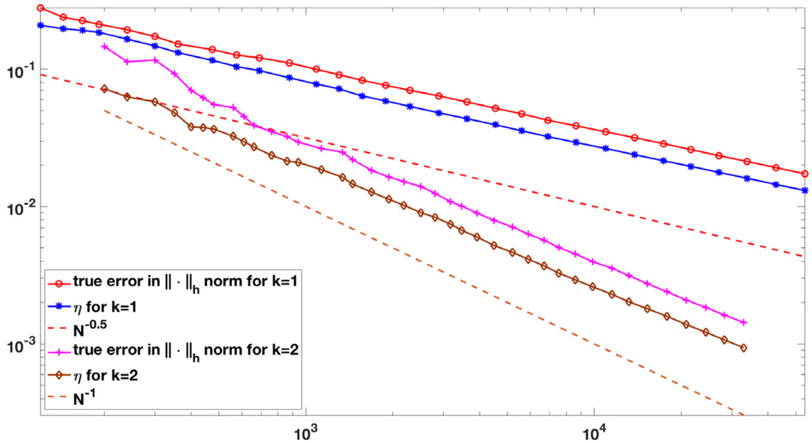


Fig. 3. Convergence tests for the L-shaped problem (Example 5.2) using a rectangular initial mesh.

and error estimator η for $k = 1$ and $k = 2$, respectively. As shown, the error estimator is quite close to the true error. From Table 2, we can see that the effective index is about $0.5 - 0.9$, which means that the error estimator is efficient and reliable on adaptive general polygonal meshes. Additionally, both converge with the optimal rate, i.e., $N^{-0.5}$ for $k = 1$ and N^{-1} for $k = 2$, where N is the total number of degrees of freedom. This shows the optimality of the adaptive algorithm.

We next repeat the same test with the initial mesh $\mathcal{T}_0$ being a polygonal mesh as shown in Fig. 4(a). Final adaptive meshes for $k = 1$ and $k = 2$ are shown in Fig. 4(b) and (c), respectively. Similarly, the meshes are correctly refined near the corner where the singularity is, showing the effectiveness of the adaptive Algorithm 1 for general polygonal meshes. Moreover, the refinements are more focused when $k = 2$ as expected. The convergence results are plotted in Fig. 5. Again, the true error and the error estimator are quite close. In fact, the effective index is about $0.7 - -0.9$ as shown in Table 3. Therefore, we conclude that the adaptive algorithm converges with optimal order for both $k = 1$ and $k = 2$ on general polygonal mesh as well.

In the final test, we consider a problem with an interior layer, which is not resolved on the initial meshes.

Example 5.3. Consider the model problem (4)–(6) with $A = I$, where I is the identity matrix, and $c = 0$ on a square domain $\Omega = (0, 1) \times (0, 1)$. The exact solution is

$$u = 16x(1 - x)y(1 - y)\arctan(25x - 100y + 50),$$

which has a sharp interior layer as shown in Fig. 6. Homogeneous Dirichlet boundary conditions are used ($g = 0$) and the right-hand side function f is computed accordingly.

We again test with two different initial meshes: a rectangular mesh and a polygonal mesh. As shown in Figs. 6, 7 and 9, both meshes are locally refined near the interior layer as expected. This shows that the a posteriori error estimator, η, successfully detects the singularities along the interior layer so that our adaptive algorithm can accurately refine along that

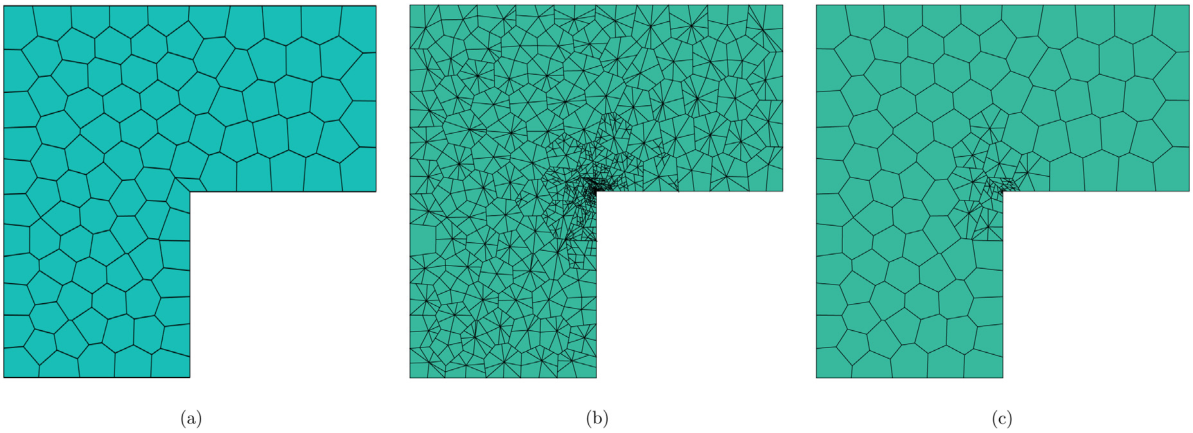


Fig. 4. Example 5.2: (a) Initial polygonal mesh; (b) Final mesh with linear elements, $k = 1$; (c) Final mesh with quadratic elements, $k = 2$.

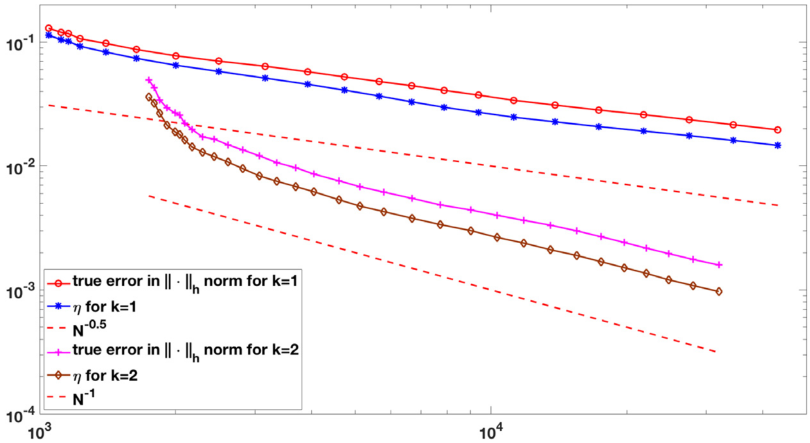


Fig. 5. Convergence tests for the L-shaped problem (Example 5.2) using a polygonal initial mesh.

Table 2
Efficiency and reliability tests for Example 5.2 with rectangular initial mesh (shown as Fig. 2(a)).

Refined steps	DOFs	$(\|u-u_h\|_h^2 + \|\mathbf{q}-\mathbf{q}_h\|_h^2)^{1/2}$	η	Eff-index
Results for $k = 1$				
1	120	2.8016e−01	2.0852e−01	7.4430e−01
5	240	1.9363e−01	1.6534e−01	8.5388e−01
10	690	1.2097e−01	9.7574e−02	8.0660e−01
15	1893	7.6209e−02	5.8731e−02	7.7065e−01
20	5595	4.7324e−02	3.5583e−02	7.5191e−01
Results for $k = 2$				
1	200	1.4622e−01	7.2139e−02	4.9336e−01
5	400	7.0034e−02	3.7995e−02	5.4251e−01
10	660	3.8928e−02	2.7145e−02	6.9731e−01
15	1340	2.4894e−02	1.6331e−02	6.5601e−01
20	2505	1.3985e−02	9.0276e−03	6.4552e−01

layer. As expected, such refinement is more focused with less pollution when $k = 2$. Again this is because the solution is quite smooth away from the singularity. In addition, we plot the true error and the error estimator in Figs. 8 and 10 for the different initial meshes, respectively. In all cases, the true error and the error estimator do not converge well at the beginning of the adaptive algorithm. This is because the mesh is still not locally refined enough to capture the interior layer completely. Once the mesh is refined to capture the layer, the optimal convergence rate is achieved for all cases. From Tables 4 and 5, we can see that the effective index is about $0.6 - -0.9$ which shows that the error estimator approximates the true error quite

Table 3
Efficiency and reliability tests for Example 5.2 with polygonal initial mesh (shown as Fig. 4(a)).

Refined steps	DOFs	$(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2)^{1/2}$	η	Eff-index
Results for $k=1$				
1	1 047	1.2916e−01	1.1339e−01	8.7792e−01
5	1 401	9.7526e−02	8.3008e−02	8.5113e−01
10	3 924	5.7645e−02	4.5632e−02	7.9161e−01
15	9 384	3.7333e−02	2.7122e−02	7.2648e−01
20	27 459	2.3641e−02	1.7583e−02	7.4376e−01
Results for $k=2$				
1	1 745	4.9363e−02	3.6032e−02	7.2995e−01
5	1 995	2.6775e−02	1.8850e−02	7.0402e−01
10	2 435	1.6496e−02	1.1911e−02	7.2205e−01
15	3 695	9.6811e−03	6.8209e−03	7.0456e−01
20	6 680	5.4837e−03	3.7892e−03	6.9099e−01

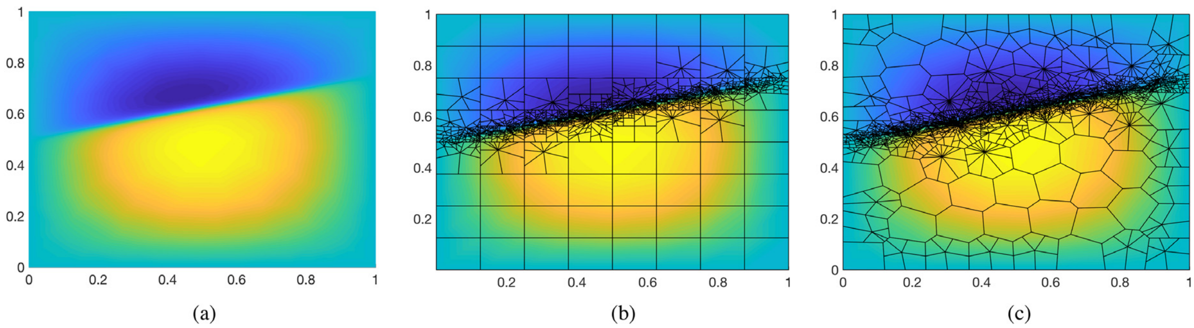


Fig. 6. Example 5.3: (a) Exact solution; (b) Numerical solution on final mesh shown as Fig. 7(c); (c) Numerical solution on final mesh shown as Fig. 9(c).

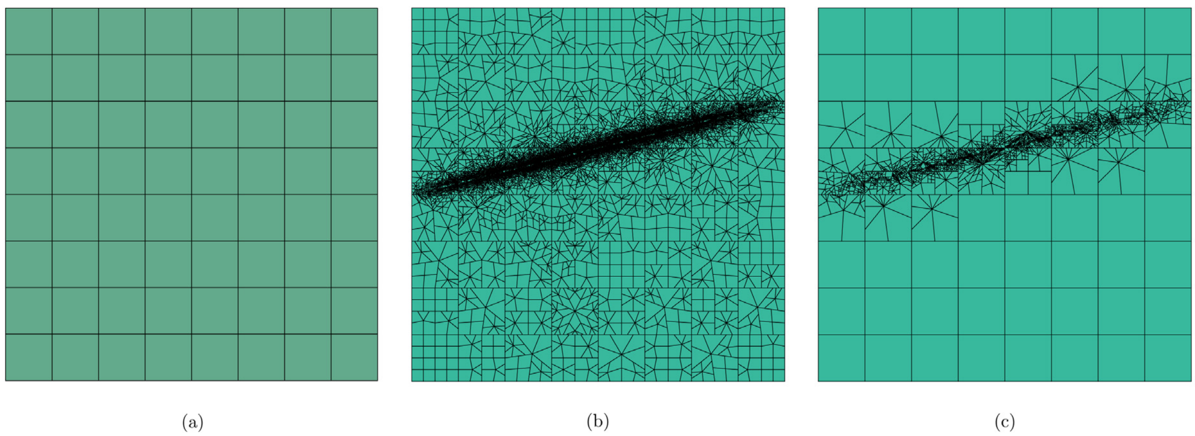


Fig. 7. Example 5.3: (a) Initial rectangular mesh; (b) Final mesh with linear elements, $k=1$; (c) Final mesh with quadratic elements, $k=2$.

well. Overall, the adaptive algorithm based on the proposed a posteriori error estimator is effective on general polygonal meshes.

6. Discussion

In [9], the WG-LS method was proposed to solve mixed formulations of Poisson's equation on general polygonal meshes. Here, in order to resolve the singularities that usually appear in practical applications, we derive an a posteriori error estimator for the WG-LS method and theoretically prove that it is reliable and efficient. An adaptive algorithm is also developed based on the proposed error indicator and a simple refinement scheme. This adaptive WG-LS method is easily implemented and the numerical experiments presented here demonstrate its robustness. The a posteriori error estimator accurately captures singularities as well as interior layers, and the overall adaptive algorithm achieves optimal convergence in practice.

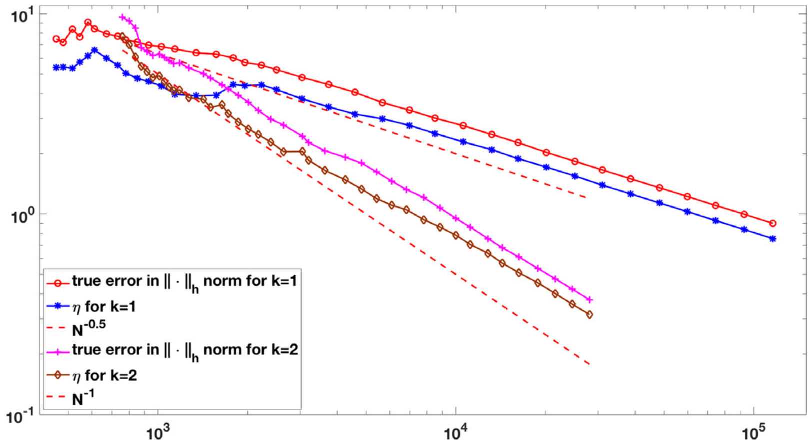
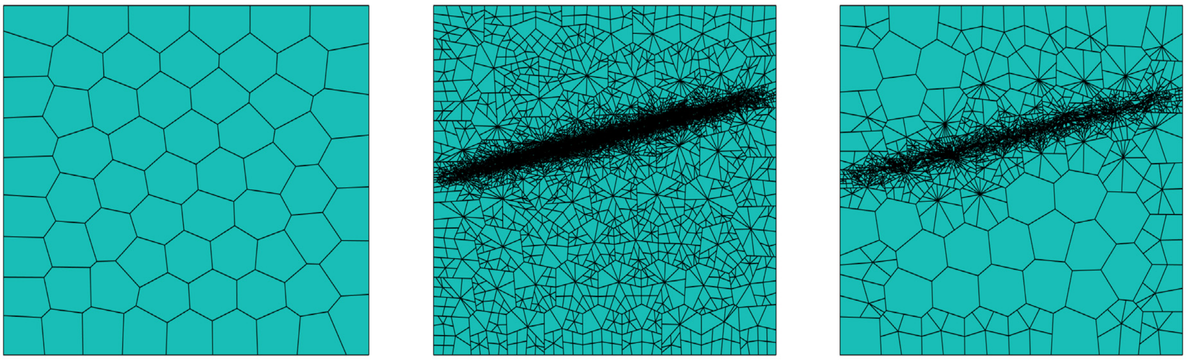


Fig. 8. Convergence tests for Example 5.3 using a rectangular initial mesh.



(a) (b) (c)

Fig. 9. Example 5.3: (a) Initial polygonal mesh; (b) Final mesh with linear elements, $k = 1$; (c) Final mesh with quadratic elements, $k = 2$.

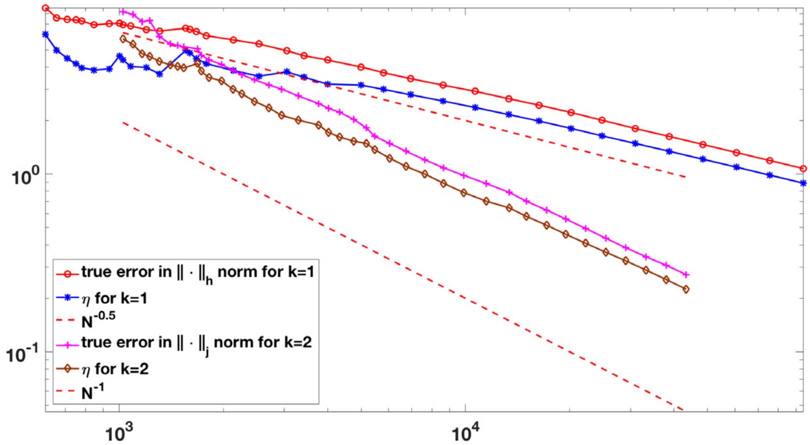


Fig. 10. Convergence tests for Example 5.3 using a polygonal initial mesh.

Future work involves extending the adaptive WG-LS method to other applications. In addition, based on the reliability and efficiency results, we will theoretically prove the optimal linear convergence and computational complexity of the adaptive algorithm using the framework proposed in [32]. Finally, in order to achieve the full potential of the adaptive WG-LS method,

Table 4
Efficiency and reliability tests for Example 5.3 with rectangular initial mesh (shown as Fig. 7(a)).

Refined steps	DOFs	$(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2)^{1/2}$	η	Eff-index
Results for $k=1$				
1	456	7.4742e+00	5.4008e+00	7.2259e−01
5	582	9.0813e+00	6.1551e+00	6.7777e−01
10	852	7.2369e+00	4.7588e+00	6.5757e−01
15	1 569	6.2796e+00	3.9191e+00	6.2410e−01
20	3 048	4.8116e+00	3.7600e+00	7.8145e−01
25	8 505	3.0131e+00	2.5276e+00	8.3886e−01
30	25 128	1.8329e+00	1.5482e+00	8.4467e−01
35	74 445	1.1041e+00	9.2828e−01	8.4075e−01
Results for $k=2$				
1	760	9.5896e+00	7.6956e+00	8.0250e−01
5	920	6.4620e+00	5.1345e+00	7.9458e−01
10	1 130	5.6455e+00	4.1229e+00	7.3031e−01
15	1 640	4.4214e+00	3.5152e+00	7.9504e−01
20	2 390	2.9776e+00	2.2892e+00	7.6882e−01
25	4 250	1.9212e+00	1.4834e+00	7.7215e−01
30	7 805	1.2091e+00	9.3808e−01	7.7587e−01
35	14 300	6.7927e−01	5.7008e−01	8.3926e−01

Table 5
Efficiency and reliability tests for Example 5.3 with polygonal initial mesh (shown as Fig. 9(a)).

Refined steps	DOFs	$\left(\|u-u_h\|_h^2+\|\mathbf{q}-\mathbf{q}_h\|_h^2\right)^{1/2}$	η	Eff-index
Results for $k=1$				
1	612	8.6299e+00	6.1078e+00	7.0774e−01
5	780	7.2732e+00	3.9663e+00	5.4533e−01
10	1 080	6.8099e+00	4.0300e+00	5.9178e−01
15	1 671	6.3390e+00	4.4649e+00	7.0436e−01
20	3 420	4.6317e+00	3.5092e+00	7.5766e−01
25	8 619	3.1636e+00	2.5765e+00	8.1444e−01
30	24 789	2.0097e+00	1.6421e+00	8.1712e−01
35	75 624	1.1920e+00	9.8636e−01	8.2747e−01
Results for $k=2$				
1	1 025	8.2012e+00	5.7751e+00	7.0417e−01
5	1 305	5.9464e+00	4.3151e+00	7.2566e−01
10	1 725	4.6714e+00	3.7978e+00	8.1298e−01
15	2 460	3.3975e+00	2.5637e+00	7.5458e−01
20	4 025	2.3416e+00	1.7140e+00	7.3201e−01
25	6 045	1.4879e+00	1.2278e+00	8.2517e−01
30	11 490	8.8482e−01	7.0348e−01	7.9506e−01
35	22 225	4.9469e−01	4.0818e−01	8.2512e−01

we will design optimal multigrid solvers based on the adaptive polygonal meshes and ultimately solve the resulting linear system of equations efficiently.

References

[1] L. Mu, J. Wang, X. Ye, A Weak Galerkin finite element method with polynomial reduction, J. Comput. Appl. Math. 285 (2015) 45–58.
[2] J. Wang, X. Ye, A weak Galerkin finite element method for second order elliptic problems, J. Comput. Appl. Math. 241 (2013) 103–115.
[3] J. Wang, X. Ye, A weak Galerkin mixed finite element method for second-order elliptic problems, Math. Comp. 83 (2014) 2101–2126.
[4] J. Wang, X. Ye, A weak Galerkin finite element method for the Stokes equations, Adv. Comput. Math. 42 (2016) 155–174.
[5] L. Mu, X. Wang, X. Ye, A modified weak Galerkin finite element method for the Stokes equations, J. Comput. Appl. Math. 275 (2015) 79–90.
[6] L. Mu, J. Wang, X. Ye, A stable numerical algorithm for the Brinkman equations by weak Galerkin finite element methods, J. Comput. Phys. 273 (2014) 327–342.
[7] C. Wang, J. Wang, R. Wang, R. Zhang, A locking-free weak Galerkin finite element method for elasticity problems in the primal formulation, J. Comput. Appl. Math..
[8] X. Hu, L. Mu, X. Ye, Weak Galerkin method for the Biot's consolidation model, Comput. Math. Appl. 75 (6) (2018) 2017–2030.
[9] L. Mu, J. Wang, X. Ye, A Least-Squares-based weak Galerkin finite element method for second order elliptic equations, SIAM J. Sci. Comput. 39 (4) (2017) A1531–A1557.
[10] Z. Cai, R. Lazarov, T.A. Manteuffel, S.F. McCormick, First-order system least squares for second-order partial differential equations: Part I, SIAM J. Numer. Anal. 31 (6) (1994) 1785–1799.
[11] Z. Cai, T.A. Manteuffel, S.F. McCormick, First-order system least squares for second-order partial differential equations. II, SIAM J. Numer. Anal. 34 (2) (1997) 425–454.

[12] J. Bramble, R. Lazarov, J. Pasciak, A least-squares approach based on a discrete minus one inner product for first order systems, Math. Comp. 66 (1997) 935–955.
[13] B. n. Jiang, L. Povinelli, Optimal least-squares finite element method for elliptic problems, Comput. Methods Appl. Mech. Engrg. 102 (1993) 199–212.
[14] A. Pehlivanov, G. Carey, R. Lazarov, Least-squares mixed finite elements for second-order elliptic problems, SIAM J. Numer. Anal. 31 (1994) 1368–1377.
[15] P. Bochev, M. Gunzburger, Accuracy of the least squares methods for the navier stokes equations, Comput. & Fluids 22 (1993) 549–563.
[16] P. Bochev, M. Gunzburger, Analysis of least-squares finite element methods for the Stokes equations, Math. Comp. 63 (1994) 479–506.
[17] Z. Cai, X. Ye, A least-squares finite element approximation to the compressible stokes equations, Numer. Methods Partial Differential Equations 16 (2000) 62–70.
[18] C.-L. Chang, An error estimate of the least squares finite element methods for the Stokes problem in three dimensions, Math. Comp. 63 (1994) 41–50.
[19] B. n. Jiang, L. Povinelli, Least-squares finite element method for fluid dynamics, Comput. Methods Appl. Mech. Engrg. 81 (1990) 13–37.
[20] Z. Cai, X. Ye, H. Zhang, Least-squares finite element approximations for the Reissner–Mindlin Plate, Numer. Linear Algebra Appl. 6 (1999) 479–496.
[21] T. Manteuffel, S. McCormick, J. Schmidt, C. Westphal, First-order system least squares for geometrically nonlinear elasticity, SIAM J. Numer. Anal. 44 (2006) 2057–2081.
[22] W. Gui, I. Babuška, The h, p, and hp Versions of the finite element method in 1 Dimension, Parts I, II, III, Numer. Math. 49 (1986) 577–683.
[23] U. Rüede, Mathematical and Computational Techniques for Multilevel Adaptive Methods, in: Frontiers in Applied Mathematics, vol. 13, SIAM, Philadephia, PA, 1993.
[24] R. Verfürth, A Review of a Posteriori Error Estimation and Adaptive Mesh-refinement Techniques, Wiley-Teubner, 1995.
[25] W. Dörfler, A convergent adaptive algorithm for Poisson's equation, SIAM J. Numer. Anal. 33 (1996) 1106–1124.
[26] R. Bank, M. Holst, A new paradigm for parallel adaptive meshing algorithms, SIAM Rev. 45 (2003) 291–323.
[27] P. Morin, R. Nochetto, K. Siebert, Data oscillation and convergence of adaptive FEM, SIAM J. Numer. Anal. 38 (2000) 466–488.
[28] P. Morin, R. Nochetto, K. Siebert, Convergence of adaptive finite element methods, SIAM Rev. 44 (2002) 631–658.
[29] J. Adler, T. Manteuffel, S. McCormick, J. Nolting, J. Ruge, L. Tang, Efficiency based adaptive local refinement for first-order system least-squares formulations, SIAM J. Sci. Comput. 33 (1) (2011) 1–24.
[30] L. Chen, J. Wang, X. Ye, A posteriori error estimates for weak Galerkin finite element methods for second order elliptic problems, J. Sci. Comput. 59 (2) (2014) 496–511.
[31] O. Karakashian, F. Pascal, A posteriori error estimates for a discontinuous Galerkin approximation of second-order elliptic problems, SIAM J. Numer. Anal. 41 (6) (2003) 2374–2399.
[32] R. Nochetto, K. Siebert, A. Veeser, Theory of adaptive finite element methods: an introduction, in: Multiscale, Nonlinear and Adaptive Approximation, Springer, Berlin, 2009, pp. 409–542.
[33] L. Chen, iFEM: an integrated finite element methods package in MATLAB, University of California at Irvine.
[34] T. Cameron, P. Glaucio, P.I.M. Anderson, PolyMesher: a general-purpose mesh generator for polygonal elements written in Matlab, Struct. Multidiscip. Optim. 45 (3) (2012) 309–328.