SIAM J. SCI. COMPUT. © 2020 Society for Industrial and Applied Mathematics
Vol. 42, No. 3, pp. B608–B629

PRESSURE ROBUST WEAK GALERKIN FINITE ELEMENT METHODS FOR STOKES PROBLEMS[*]

LIN MU[†]

Abstract. In this paper, we develop a pressure robust weak Galerkin finite element scheme for Stokes equations on polygonal mesh. The major idea for achieving a pressure-independent energy-error estimate is to use a divergence preserving velocity reconstruction operator in the discretization of the right-hand side body force. Our scheme only modifies the body force assembling but remains the same stiffness matrix for Stokes simulation. The optimal convergence results for velocity and pressure have been established in this paper. Finally, numerical examples based on triangular, rectangular, and polygonal meshes are presented for validating the theoretical conclusions.

Key words. weak Galerkin, finite element, Stokes equation, pressure robust, polygonal meshes, divergence preserving

AMS subject classifications. 65N15, 65N30

DOI. 10.1137/19M1266320

1. Introduction.

In this paper, we consider the following viscosity-dependent Stokes equations: Find velocity $\mathbf{u} : \Omega \to \mathbb{R}^2$ and pressure $p : \Omega \to \mathbb{R}$ such that

$$-\nu\Delta\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega, \tag{1.1}$$

$$\nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega, \tag{1.2}$$

$$\mathbf{u} = 0 \quad \text{on } \partial\Omega, \tag{1.3}$$

where $\nu > 0$ is a constant viscosity parameter, and $\mathbf{f} \in [L^2(\Omega)]^2$ is a given vector field.

The Stokes equation plays an important role in modeling many natural phenomena, including ocean currents, water flow, and air flow. However, the classical finite element methods for incompressible Stokes problems are not pressure-robust. Their velocity error is pressure dependent, which is shown as follows:

$$\|\nabla(\mathbf{u}-\mathbf{u}_h)\| \le C \inf_{\mathbf{w}\in V_h} \|\nabla(\mathbf{u}-\mathbf{w})\| + \frac{1}{\nu}\inf_{q\in Q_h}\|p-q\|.$$

Here V_h and Q_h denote the finite element spaces for velocity and pressure. This velocity error shows the locking phenomenon, as small parameters in $\nu \ll 1$ will produce large error in velocity. This may result in a bad velocity simulation in cases when the irrotational part of the body force is dominant.

Several divergence-free elements have been developed for two-dimensional problems [11, 29] and three-dimensional problems [35]. The idea to enrich the H(div; Ω)-conforming elements locally with divergence-free rational shape-functions has been proposed in [13, 14]. These additional functions will help to enforce the continuity and achieve the divergence-free property. Cockburn [9] and Wang [33] modified the variational formulation and introduced the tangential penalty and thus obtained discontinuous Galerkin divergence-free schemes. In addition, Chen proposed a divergence-free MAC scheme on triangular grids by introducing a discrete dual curl operator

[*]Submitted to the journal's Computational Methods in Science and Engineering section June 5, 2019; accepted for publication (in revised form) February 13, 2020; published electronically May 12, 2020.
https://doi.org/10.1137/19M1266320

[†]Department of Mathematics, University of Georgia, Athens, GA 30602 (linmu@uga.edu).

[6]. There are other approaches for dealing with such difficulties by adding grad-div stabilization [25, 26, 15]. An alternative method to get pressure independent error estimate was introduced recently in [12, 19, 20, 22] by divergence preserving velocity reconstructions on the simplicial meshes.

In [19], the problem is analyzed and a reconstruction is presented to retrieve pressure robustness. This reconstruction maps discrete divergence-free test functions onto exactly divergence-free test functions applied only on the right-hand side. This approach was performed for several elements using discontinuous pressure [16, 20, 19, 21, 3] and continuous pressure elements [17, 18]. The advantage of this approach shows a pressure independent velocity error estimate

$$\|\nabla(\mathbf{u}-\mathbf{u}_h)\| \leq C \inf_{\mathbf{w}\in V_h} \|\nabla(\mathbf{u}-\mathbf{w})\|.$$

Recently, the polygonal/polyhedral finite element methods (including all the finite element methods that can be applied to the polyhedral grids) were proposed for dealing with meshing complexity. The flexibility in handling polygonal meshes and meshes with hanging nodes not only weakens the complexity in meshing for the irregular domain but also increases the efficiency in mesh refinement and coarsening for adaptive finite element methods. Because of the features of polygonal finite element methods, there are increasing demands for extending existing numerical schemes to polygonal meshes. Virtual element methods [1, 2], mimetic finite difference methods [23], hybrid high order (HHO) methods [27], hybridized discontinuous Galerkin (HDG) methods [10], weak Galerkin (WG) finite element methods [30], generalized barycentric coordinates methods [4], etc., have been developed on these types of meshes for various types of partial differential equations. The bridging and connections in HHO, HDG, and WG methods are discussed in [8, 7, 34]. Within the framework of WG finite element methods, discrete weak gradient, divergence, or curl are proposed for discontinuous basis functions via integration by parts based on unknowns in element interiors and on element interfaces.

In this paper, we develop a numerical discretization of pressure-robust schemes for viscosity dependent Stokes equations on polygonal meshes. Our methods are based on the WG finite element methods and extend the velocity reconstruction operator and pressure robust scheme to the polygonal elements. Inspired by [28, 4], the key ideas of the new scheme for achieving pressure-independent error in velocity are the use of a divergence preserving velocity reconstruction operator in the discretization of the source term and the generalized barycentric coordinates on polygonal mesh. Compared to the existing WG scheme for the Stokes equation, our scheme only modifies the body force assembling but remains the same stiffness matrix, and thus achieves divergence preserving by minimal effort. To the best of our knowledge, in the pressure robust community, few of the existing divergence preserving methods are based on polygonal elements [5]. Our project develops a pressure robust numerical scheme on the polygonal mesh, and the techniques designed in our project can also be integrated into other polygonal finite element methods.

This paper is organized as follows. In the remainder of this section, we present the weak form of the model problem. A brief review of basis functions on the polygonal meshes will be presented in section 2. The discretization is developed in section 3. Our main results regarding error estimates are stated in section 4. Section 5 presents several numerical examples for Stokes equations that compare the new pressure-robust scheme with the existing WG scheme. Finally, this paper is summarized with concluding remarks in section 6.

The weak formulation of the problem reads: Find $(\mathbf{u}, p) \in [H_0^1(\Omega)]^2 \times L_0^2(\Omega)$ such that

$$\nu(\nabla \mathbf{u}, \nabla \mathbf{v}) - (\nabla \cdot \mathbf{v}, p) = (\mathbf{f}, \mathbf{v}) \quad \forall \mathbf{v} \in [H_0^1(\Omega)]^2,$$
$$(\nabla \cdot \mathbf{u}, q) = 0 \quad \forall q \in L_0^2(\Omega).$$

Here $L_0^2(\Omega) := \{q \in L^2(\Omega) | \int_\Omega q = 0\}$.

Let D be any open bounded domain with Lipschitz continuous boundary in $\mathbb{R}^2$. We use the standard definitions for the Sobolev space $H^s(D)$ and the associated inner product $(\cdot,\cdot)_{s,D}$, norm $\|\cdot\|_{s,D}$, and seminorm $|\cdot|_{s,D}$ for any $s \geq 0$. The space $H^0(D)$ coincides with $L^2(D)$, for which the norm and the inner product are denoted by $\|\cdot\|_D$ and $(\cdot,\cdot)_D$, respectively. When $D = \Omega$, we drop the subscript D in the norm and inner product notation. The space $H(\text{div}; D)$ is defined as the set of vector-valued functions on D as follows:

$$H(\text{div}; D) = \{\mathbf{v} : \mathbf{v} \in [L^2(D)]^2, \nabla \cdot \mathbf{v} \in L^2(D)\}.$$

In what follows, we adopt the notation $A \lesssim B$ for an inequality $A \leq CB$, where C is a positive constant which is independent of mesh size h but may take different values at different locations.

2. Basis functions on polygonal mesh.

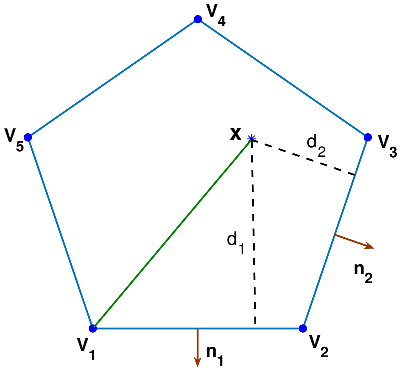

FIG. 1. *Illustration of Wachspress coordinates on a pentagon.*

2.1. Wachspress coordinates.

In this section, we shall briefly review the Wachspress coordinates on a polygonal cell. Let T be a polygon (as shown in Figure 1) with n vertices $\mathbf{v}_i$ $(1 \leq i \leq n)$ that are arranged counterclockwise, and let $\mathbf{n}_i$ $(1 \leq i \leq n)$ be the outward unit normal vector on edge e_i that connects vertices $\mathbf{v}_i$ and $\mathbf{v}_{i+1}$. Here the modulus n convention is adopted for indexing. Let $\mathbf{x} \in T^0$ be the interior of T. Its distance to edge e_i and a scaled normal vector are defined as

$$d_i = (\mathbf{v}_i - \mathbf{x}) \cdot \mathbf{n}_i, \quad 1 \leq i \leq n, \tilde{\mathbf{n}}_i = \frac{1}{d_i}\mathbf{n}_i, \quad 1 \leq i \leq n. \tag{2.1}$$

Then the Wachspress coordinates are defined as

$$\lambda_i(\mathbf{x}) = \omega_i(\mathbf{x})/W(\mathbf{x}), \quad 1 \leq i \leq n, \tag{2.2}$$

where $\omega_i(\mathbf{x}) = \det(\tilde{\mathbf{n}}_i, \tilde{\mathbf{n}}_{i+1}),\ W(\mathbf{x}) = \sum_{i=1}^n \omega_i(\mathbf{x})$.

The Wachspress coordinates are usually rational functions. One example of calculation on a quadrilateral can be found in [24]. Here we plot a set of bases on the pentagon in Figure 2.

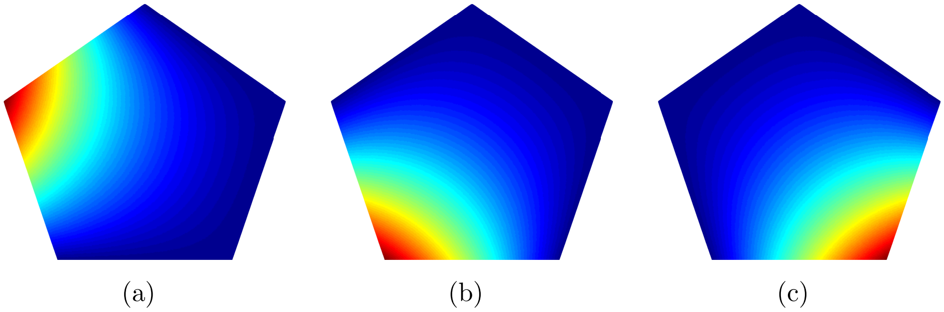

FIG. 2. *Illustration of Wachspress coordinate functions for the pentagon shown in Figure* 1 *for* (a) λ_5, (b) λ_1, *and* (c) λ_2.

Finally, a counterclockwise 90^o rotation of the gradient is defined as

$$\text{curl}(\lambda_i) = \begin{bmatrix} -\partial_y \lambda_i \\ \partial_x \lambda_i \end{bmatrix}, \quad 1 \le i \le n. \tag{2.3}$$

2.2. H(div)-space on polygon ($\mathbb{CW}_0$). The minimal degree H(div) conforming finite element space is constructed in [4] on general convex polygons. In this section, we shall give a brief review for such space. Assume T is a convex polygon. Choose an arbitrary point $\mathbf{x}_*$ inside this polygon T. Denote $|T|$, $|e_i|$ as the area and the length of edge e_i for $1 \le i \le n$. Let $|T_i|$ be the area of the triangle formed by $\mathbf{x}_*$, $\mathbf{v}_i$, $\mathbf{v}_{i+1}$.

Then for each $1 \le i \le n$, define $\Theta_i \in \mathbb{CW}_0$ by

$$\Theta_i = c_{i,0}(\mathbf{x} - \mathbf{x}_*) + \sum_{k=1}^{n} c_{i,k}\text{curl}\lambda_k, \tag{2.4}$$

where $c_{i,0} = \frac{|e_i|}{2|T|}$, $c_{i,k} = -\frac{1}{n}\sum_{l=1}^{n-1} l b_{i,k+l}$, and $b_{i,l} = \delta_{i,l}|e_l| - |e_i|\frac{|T_l|}{|T|}$. Here $\delta_{i,l}$ is the Kronecker symbol.

Furthermore, for these basis functions, we have the following properties:

$$\Theta_i|_{e_j} \cdot \mathbf{n}_j = \delta_{i,j}, \quad \nabla \cdot \Theta_i = 2c_{i,0} \quad \forall 1 \le i,j \le n. \tag{2.5}$$

The pointwise normal flux on element boundaries and the pointwise divergence on elements are both constants (shown in Figure 3). It is noted that when a polygon degenerates to a triangle or rectangle, these $\mathbb{CW}_0$ elements become the standard Raviart–Thomas space RT_0 or $\text{RT}_{[0]}$.

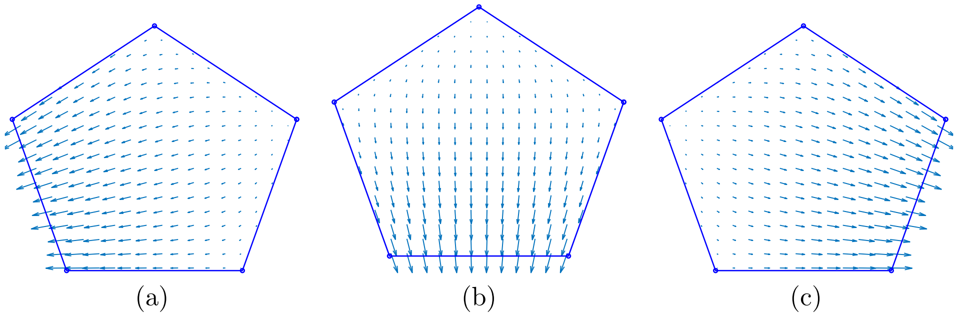

FIG. 3. *Illustration of* $\mathbb{CW}_0$ *bases with respect to Figure* 1*:* (a) Θ_5*,* (b) Θ_1*, and* (c) Θ_2*.*

Now let us define the interpolation operator. For any $\mathbf{q} \in H(\mathrm{div}, T) \cap [L^p(T)]^2$ (with $p > 2$), define $\Pi_T\mathbf{q} \in \mathbb{CW}_0(T)$ by

$$\Pi_T\mathbf{q}\cdot\mathbf{n}|_{e_j} = \frac{1}{|e_j|}\int_{e_j} \mathbf{q}\cdot\mathbf{n}ds,$$

where $\mathbf{n}$ denotes the outer normal on edge e_j for the element T. According to the definition, it is clear that

$$\Pi_T\mathbf{q} = \sum_{i=1}^{n} a_i\Theta_i, \text{ where } a_i = \frac{1}{|e_i|}\int_{e_i}\mathbf{q}\cdot\mathbf{n}ds.$$

Denote $\mathcal{Q}_T$ the $[L^2(T)]^2$-projection onto the $[P_0(T)]^2$; clearly we have $\Pi_T\mathcal{Q}_T\mathbf{q} = \mathcal{Q}_T\mathbf{q}$.

LEMMA 2.1. *For* $1 \le i \le n$*, one has* [4]

$$\|\Theta_i\|_T \le C(n)|e_i|. \tag{2.6}$$

For $\mathbf{q} \in [H^1(T)]^2$*, one has*

$$\|\mathbf{q} - \Pi_T\mathbf{q}\|_T \le C(n)h_T\|\mathbf{q}\|_{1,T}. \tag{2.7}$$

Here $C(n)$ *is a generic positive constant depending only on* n*.*

3. Finite element space and numerical scheme.

3.1. Finite element space.

Let $\mathcal{T}_h$ be a partition of the domain Ω with mesh size h that consists of arbitrary convex polygons. Assume that partition $\mathcal{T}_h$ is shape regular (as defined in [31, 24]). Denote by $\mathcal{E}_h$ the set of all edges in $\mathcal{T}_h$, and let $\mathcal{E}_h^0 = \mathcal{E}_h\backslash\partial\Omega$ be the set of all interior edges.

We define a weak Galerkin finite element space for the velocity variable as follows:

$$V_h = \{\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} : \mathbf{v}_0 \in [P_1(T)]^2, \mathbf{v}_b \in [P_0(e)]^2, e \subset \partial T\}.$$

We emphasize that there is only a *single* value $\mathbf{v}_b$ defined on each edge $e \in \mathcal{E}_h$. For the pressure variable, we define the following finite element space:

$$W_h = \{q : q \in L_0^2(\Omega), q|_T \in P_0(T)\}.$$

Denote by V_h^0 the subspace of V_h consisting of discrete weak functions with vanishing boundary value; i.e.,

$$V_h^0 = \{\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} \in V_h, \mathbf{v}_b = 0 \text{ on } \partial\Omega\}.$$

The discrete weak gradient (∇_w) and weak divergence ($\nabla_w \cdot$) for space V_h can be computed on each element T:

$$(\nabla_w \mathbf{v}, \psi)_T = \langle \mathbf{v}_b \cdot \mathbf{n}, \psi \rangle_{\partial T} \ \forall \psi \in [P_0(T)]^{2\times 2}, \tag{3.1}$$

$$(\nabla_w \cdot \mathbf{v}, \phi)_T = \langle \mathbf{v}_b \cdot \mathbf{n}, \phi \rangle_{\partial T} \ \forall \phi \in P_0(T). \tag{3.2}$$

The usual L^2 inner product can be written locally on each element as follows:

$$(\nabla_w \mathbf{v}, \nabla_w \mathbf{w}) = \sum_{T \in \mathcal{T}_h} (\nabla_w \mathbf{v}, \nabla_w \mathbf{w})_T,$$

$$(\nabla_w \cdot \mathbf{v}, q) = \sum_{T \in \mathcal{T}_h} (\nabla_w \cdot \mathbf{v}, q)_T.$$

3.2. Divergence-preserving velocity reconstruction. Let $T \in \mathcal{T}_h$, and let $\mathbb{CW}_0(T)$ denote the lowest order Raviart–Thomas polynomial space on T constructed in (2.4). We define the operator $\mathbb{R}_{\mathbb{T}} : V_h \to \mathbb{CW}_0(T)$ such that, for all $\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} \in V_h$,

$$\sum_{e \in \partial T} \int_e \mathbb{R}_{\mathbb{T}}(\mathbf{v}) \cdot \mathbf{n} q ds = \sum_{e \in \partial T} \int_e \mathbf{v}_b \cdot \mathbf{n} q ds \quad \forall q \in \mathbb{P}^0(e), e \in \partial T. \tag{3.3}$$

Also we can represent $\mathbb{R}_{\mathbb{T}}(\mathbf{v})$ by the basis $\Theta_j \in \mathbb{CW}_0(T)$ $(j = 1, \ldots, n)$,

$$\mathbb{R}_{\mathbb{T}}(\mathbf{v}) = \sum_{i=1}^{n} a_i \Theta_i, \text{ where } a_i = \frac{1}{|e_i|} \int_{e_i} \mathbf{v}_b \cdot \mathbf{n} ds.$$

3.3. Numerical scheme. We are now ready to describe a WG finite element scheme for the Stokes equations (1.1)–(1.3). To this end, we first introduce three bilinear forms as follows:

$$s(\mathbf{v}, \mathbf{w}) = \sum_{T \in \mathcal{T}_h} h_T^{-1} \langle \mathbf{Q}_b \mathbf{v}_0 - \mathbf{v}_b, \mathbf{Q}_b \mathbf{w}_0 - \mathbf{w}_b \rangle_{\partial T},$$

$$a(\mathbf{v}, \mathbf{w}) = (\nu \nabla_w \mathbf{v}, \nabla_w \mathbf{w}) + \nu s(\mathbf{v}, \mathbf{w}),$$

$$b(\mathbf{v}, q) = (\nabla_w \cdot \mathbf{v}, q).$$

Here $\mathbf{Q}_b$ denotes the L^2-projection to the edge $e \subset \partial T$. Next, we shall develop our new discretization as Algorithm 3.1. The standard scheme presented as Algorithm 3.2 [32] will be also cited for comparison.

Algorithm 3.1 WG algorithm.

The WG finite element: A numerical approximation for (1.1)–(1.3) is to find $\mathbf{u}_h = \{\mathbf{u}_0, \mathbf{u}_b\} \in V_h^0$ and $p_h \in W_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}) - b(\mathbf{v}, p_h) = (\mathbf{f}, \mathbb{R}_{\mathbb{T}}(\mathbf{v})), \tag{3.4}$$

$$b(\mathbf{u}_h, q) = 0 \tag{3.5}$$

for all $\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} \in V_h^0$ and $q \in W_h$. Here $\mathbb{R}_{\mathbb{T}}(\mathbf{v})$ is defined in (3.3).

Remark 3.1. Algorithms 3.1 and 3.2 have the same stiffness matrix but only differ at the right-hand side.

Algorithm 3.2 WG algorithm [32].

The WG finite element: A numerical approximation for (1.1)–(1.3) is to find $\mathbf{u}_h = \{\mathbf{u}_0, \mathbf{u}_b\} \in V_h^0$ and $p_h \in W_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}) - b(\mathbf{v}, p_h) = (\mathbf{f}, \mathbf{v}_0), \tag{3.6}$$

$$b(\mathbf{u}_h, q) = 0 \tag{3.7}$$

for all $\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} \in V_h^0$ and $q \in W_h$.

3.4. Preliminary results. We introduce the following discrete norms on V_h:

$$|\!|\!|\mathbf{v}|\!|\!|^2 := \sum_{T\in\mathcal{T}_h} \|\nabla_w \mathbf{v}\|_T^2 + \sum_{T\in\mathcal{T}_h} h_T^{-1}\|\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b\|_{\partial T}^2. \tag{3.8}$$

As shown in [32], the defined $|\!|\!|\cdot|\!|\!|$-norm is a norm in V_h^0. We cite the following lemmas.

LEMMA 3.2. *For any* $\mathbf{v}, \mathbf{w} \in V_h^0$, *we have* [32]

$$|a(\mathbf{v}, \mathbf{w})| \le \nu|\!|\!|\mathbf{v}|\!|\!||\!|\!|\mathbf{w}|\!|\!|, \tag{3.9}$$

$$a(\mathbf{v}, \mathbf{v}) = \nu|\!|\!|\mathbf{v}|\!|\!|^2. \tag{3.10}$$

The L^2-projection in the finite element space V_h is given by $\mathbf{Q}_h\mathbf{v} = \{\mathbf{Q}_0\mathbf{v}, \mathbf{Q}_b\mathbf{v}\}$ for $\mathbf{v} \in [H^1(\Omega)]^2$. Similarly, denote the L^2-projection into finite element space W_h as $\Pi_h q$ for $q \in L^2(\Omega)$.

LEMMA 3.3. *The projection operators* $\mathbf{Q}_h$, $\mathbb{Q}_h$, *and* Π_h *satisfy the following commutative properties* [32]*:*

$$\nabla_w(\mathbf{Q}_h\mathbf{v}) = \mathbb{Q}_h(\nabla\mathbf{v}) \quad \forall \mathbf{v} \in [H^1(\Omega)]^2, \tag{3.11}$$

$$\nabla_w \cdot (\mathbf{Q}_h\mathbf{v}) = \Pi_h(\nabla \cdot \mathbf{v}) \quad \forall \mathbf{v} \in H(div;\Omega). \tag{3.12}$$

LEMMA 3.4. *There exists a positive constant* β *independent of* h *such that* [32]

$$\sup_{\mathbf{v}\in V_h^0} \frac{b(\mathbf{v}, \rho)}{|\!|\!|\mathbf{v}|\!|\!|} \ge \beta\|\rho\| \tag{3.13}$$

for all $\rho \in W_h$.

Thus, we have the following theorem.

THEOREM 3.5. *The WG finite element Algorithms* 3.1 *and* 3.2 *admit one and only one solution.*

LEMMA 3.6. *Let* $\mathcal{T}_h$ *be a finite element partition of* Ω *satisfying the shape regularity assumptions, and let* $\mathbf{w} \in [H^{r+1}(\Omega)]^d$ *and* $\rho \in H^r(\Omega)$ *with* $1 \le r \le k$. *Then, for* $0 \le s \le 1$, *we have* [32]

$$\sum_{T\in\mathcal{T}_h} h_T^{2s}\|\mathbf{w} - \mathbf{Q}_0\mathbf{w}\|_{T,s}^2 \lesssim h^{2(r+1)}\|\mathbf{w}\|_{r+1}^2, \tag{3.14}$$

$$\sum_{T\in\mathcal{T}_h} h_T^{2s}\|\nabla\mathbf{w} - \mathbb{Q}_h(\nabla\mathbf{w})\|_{T,s}^2 \lesssim h^{2(r)}\|\mathbf{w}\|_{r+1}^2, \tag{3.15}$$

$$\sum_{T\in\mathcal{T}_h} h_T^{2s}\|\rho - \Pi_h\rho\|_{T,s}^2 \lesssim h^{2(r)}\|\mathbf{w}\|_{r}^2. \tag{3.16}$$

LEMMA 3.7. *For any* $\mathbf{v} = \{\mathbf{v}_0, \mathbf{v}_b\} \in V_h$, *we have* [32]

$$\sum_{T\in\mathcal{T}_h} \|\nabla \mathbf{v}_0\|_T^2 \lesssim |\!|\!|\mathbf{v}|\!|\!|^2. \tag{3.17}$$

LEMMA 3.8. *The operator* $\mathbb{R}_{\mathbb{T}}$ *is divergence-preserving, i.e., for all* $\mathbf{v} \in V_h$, *the following holds:*

$$\operatorname{div}(\mathbb{R}_{\mathbb{T}}(\mathbf{v})) = \nabla_w \cdot \mathbf{v}, \tag{3.18}$$

and $\mathbb{R}_{\mathbb{T}}(\mathbf{v})|_e \cdot \mathbf{n}$ *only depends on* $\mathbf{v}_b|_e \cdot \mathbf{n}$. *Further, for all* $\mathbf{v} \in V_h$, *the following bound holds:*

$$\left(\sum_{T\in\mathcal{T}_h} \|\mathbb{R}_{\mathbb{T}}(\mathbf{v}) - \mathbf{v}_0\|_T^2\right)^{1/2} \lesssim h|\!|\!|\mathbf{v}|\!|\!|. \tag{3.19}$$

Proof. For all $q \in P_0(T)$, by using integration by parts, $\nabla q|_T = \mathbf{0}$ for $q \in P_0(T)$, the definition of $\mathbb{R}_{\mathbb{T}}(\mathbf{v})$ as (3.3), and the definition of $\nabla_w \cdot \mathbf{v}$ in (3.2), we arrive at

$$\begin{aligned}(\operatorname{div}(\mathbb{R}_{\mathbb{T}}(\mathbf{v})), q)_T &= -(\mathbb{R}_{\mathbb{T}}(\mathbf{v}), \nabla q)_T + \langle \mathbb{R}_{\mathbb{T}}(\mathbf{v}) \cdot \mathbf{n}, q\rangle_{\partial T}\\ &= \langle \mathbf{v}_b \cdot \mathbf{n}, q\rangle_{\partial T} = (\nabla_w \cdot \mathbf{v}, q)_T,\end{aligned}$$

which thus proves (3.18).

By the triangle inequality, definition of $\mathbb{\Pi}_T$, estimates in (2.7), Cauchy–Schwarz inequality, and (2.6), it is obtained that

$$\begin{aligned}\|\mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v})\|_T^2 &= \|\mathbf{v}_0 - \mathbb{\Pi}_T(\mathbf{v}_0)\|_T^2 + \|\mathbb{\Pi}_T(\mathbf{v}_0) - \mathbb{R}_{\mathbb{T}}(\mathbf{v})\|_T^2\\ &\le Ch_T^2\|\mathbf{v}_0\|_{1,T}^2 + \left\|\sum_{j=1}^n \left(\frac{1}{|e_j|}\int_{e_j} (\mathbf{v}_0 - \mathbf{v}_b)\cdot \mathbf{n} ds\right)\Theta_j\right\|_T^2\\ &= Ch_T^2\|\mathbf{v}_0\|_{1,T}^2 + \left\|\sum_{j=1}^n \left(\frac{1}{|e_j|}\int_{e_j} (\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b)\cdot \mathbf{n} ds\right)\Theta_j\right\|_T^2\\ &\lesssim h_T^2\|\mathbf{v}_0\|_{1,T}^2 + \sum_{j=1}^n \frac{1}{|e_j|}\left|\int_{e_j} (\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b)\cdot \mathbf{n} ds\right|^2 \|\Theta_j\|_{0,T}^2\\ &\lesssim h_T^2\|\mathbf{v}_0\|_{1,T}^2 + \sum_{j=1}^n |e_j| \int_{e_j} |(\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b)\cdot \mathbf{n}|^2\, ds\\ &\lesssim Ch_T^2\|\mathbf{v}_0\|_{1,T}^2 + \sum_{j=1}^n h_T|e_j| \left(h_T^{-1}\int_{e_j} |\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b|^2\, ds\right).\end{aligned}$$

Thus, by taking summation of T, (3.17), and definition of $|\!|\!|\cdot|\!|\!|$-norm, it is obtained that

$$\begin{aligned}\sum_{T\in\mathcal{T}_h} \|\mathbb{R}_{\mathbb{T}}(\mathbf{v}) - \mathbf{v}_0\|_T^2 &\lesssim \sum_{T\in\mathcal{T}_h} h_T^2\|\nabla \mathbf{v}_0\|_T^2 + \sum_{T\in\mathcal{T}_h} h_T^2\big(h_T^{-1}\|\mathbf{Q}_b\mathbf{v}_0 - \mathbf{v}_b\|_{0,\partial T}^2\big)\\ &\lesssim h^2|\!|\!|\mathbf{v}|\!|\!|^2,\end{aligned}$$

which then completes the proof. ∎

4. Main results.

In this section, we shall present the main results concerning the convergence analysis of scheme (3.4)–(3.5).

4.1. Error equation.

Denote by $\mathbf{u}$ and p the exact solution of (1.1)–(1.3). Denote by $\mathbf{e}_h$ and ϵ_h the corresponding error given by

$$\mathbf{e}_h = \{\mathbf{e}_0, \mathbf{e}_b\} = \{\mathbf{Q}_0\mathbf{u} - \mathbf{u}_0, \mathbf{Q}_b\mathbf{u} - \mathbf{u}_b\}, \quad \epsilon_h = \Pi_h p - p_h. \tag{4.1}$$

Lemma 4.1. *Let $(\mathbf{w};\rho) \in [H^1(\Omega)]^2 \times L^2(\Omega)$ be sufficiently smooth, and satisfy the following equation:*

$$-\nu\Delta\mathbf{w} + \nabla\rho = \eta \tag{4.2}$$

in the domain Ω. Let $Q_h\mathbf{w} = \{Q_0\mathbf{w}, Q_b\mathbf{w}\}$, and let $\Pi_h\rho$ be the L^2-projection of $(\mathbf{w};\rho)$ into the finite element space $V_h \times W_h$. Then, the following equation holds:

$$(\nu\nabla_w(\mathbf{Q}_h\mathbf{w}), \nabla_w\mathbf{v}) - (\nabla_w \cdot \mathbf{v}, \Pi_h\rho) = (\eta, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) + \nu\ell_w(\mathbf{v}) + D_w(\mathbf{v}) \tag{4.3}$$

for all $\mathbf{v} \in V_h^0$, where $\ell_{\mathbf{w}}$ and $D_{\mathbf{w}}$ are defined by

$$\ell_{\mathbf{w}}(\mathbf{v}) = \sum_{T\in\mathcal{T}_h} \langle \mathbf{v}_0 - \mathbf{v}_b, \nabla\mathbf{w}\cdot\mathbf{n} - \mathbb{Q}_h(\nabla\mathbf{w})\cdot\mathbf{n}\rangle_{\partial T}, \tag{4.4}$$

$$D_{\mathbf{w}}(\mathbf{v}) = \sum_{T\in\mathcal{T}_h} (\nu\Delta\mathbf{w}, \mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v}))_T. \tag{4.5}$$

Proof. First, by integration by parts, (3.11), and definition of ∇_w, it follows that

$$\begin{aligned}
(\nabla_w(\mathbf{Q}_h\mathbf{w}), \nabla_w\mathbf{v})_T &= (\mathbb{Q}_h(\nabla\mathbf{w}), \nabla_w\mathbf{v})_T \\
&= -(\mathbf{v}_0, \nabla\cdot\mathbb{Q}_h(\nabla\mathbf{w}))_T + \langle \mathbf{v}_b, \mathbb{Q}_h(\nabla\mathbf{w})\cdot\mathbf{n}\rangle_{\partial T} \\
&= (\nabla\mathbf{v}_0, \mathbb{Q}_h(\nabla\mathbf{w}))_T - \langle \mathbf{v}_0 - \mathbf{v}_b, \mathbb{Q}_h(\nabla\mathbf{w})\cdot\mathbf{n}\rangle_{\partial T} \\
&= (\nabla\mathbf{w}, \nabla\mathbf{v}_0)_T - \langle \mathbf{v}_0 - \mathbf{v}_b, \mathbb{Q}_h(\nabla\mathbf{w})\cdot\mathbf{n}\rangle_{\partial T}. \qquad (4.6)
\end{aligned}$$

Test (4.2) by $\mathbb{R}_{\mathbb{T}}(\mathbf{v})$ to obtain

$$\begin{aligned}
(\eta, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) &= -(\nu\Delta\mathbf{w}, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) + (\nabla\rho, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) \\
&= -(\nu\Delta\mathbf{w}, \mathbf{v}_0) + (\nu\Delta\mathbf{w}, \mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v})) + (\nabla\rho, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) \\
&= \nu I_1 + I_2 + I_3. \qquad (4.7)
\end{aligned}$$

It follows from the integration by parts, definition of $\mathbb{Q}_h$, $\sum_{T\in\mathcal{T}_h}\langle \mathbf{v}_b, \nabla\mathbf{w}\cdot\mathbf{n}\rangle_{\partial T} = 0$, and definition of ∇_w that

$$\begin{aligned}
I_1 = -(\Delta\mathbf{w}, \mathbf{v}_0) &= \sum_{T\in\mathcal{T}_h} (\nabla\mathbf{w}, \nabla\mathbf{v}_0)_T - \sum_{T\in\mathcal{T}_h} \langle \nabla\mathbf{w}\cdot\mathbf{n}, \mathbf{v}_0\rangle_{\partial T} \\
&= (\nabla_w\mathbf{Q}_h\mathbf{w}, \nabla_w\mathbf{v}) + \sum_{T\in\mathcal{T}_h} \langle \mathbf{v}_0 - \mathbf{v}_b, \big(\mathbb{Q}_h\nabla\mathbf{w} - \nabla\mathbf{w}\big)\cdot\mathbf{n}\rangle_{\partial T}.
\end{aligned}$$

For the second term, we have

$$I_2 = \sum_{T\in\mathcal{T}_h} (\nu\Delta\mathbf{w}, \mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v}))_T := D_{\mathbf{w}}(\mathbf{v}).$$

By integration by parts, $\mathbb{R}_{\mathbb{T}}(\mathbf{v})$ has a continuous normal component at mesh interfaces, and ρ is continuous across interfaces and the property of $\nabla\cdot\mathbb{R}_{\mathbb{T}}(\mathbf{v})$. It follows that

$$\begin{aligned} I_3 = (\nabla\rho, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) &= \sum_{T\in\mathcal{T}_h} -(\rho, \nabla\cdot\mathbb{R}_{\mathbb{T}}(\mathbf{v}))_T + \langle\rho, \mathbb{R}_{\mathbb{T}}(\mathbf{v})\cdot\mathbf{n}\rangle_{\partial T} \\ &= \sum_{T\in\mathcal{T}_h} -(Q_h\rho, \nabla\cdot\mathbb{R}_{\mathbb{T}}(\mathbf{v}))_T = -(Q_h\rho, \nabla_w\cdot\mathbf{v}). \end{aligned}$$

Here we have used (3.18) in the last step. By substituting all terms I_1, I_2, I_3 into (4.7), we complete the proof. □

The following is a result on the error equation for the WG finite element scheme.

LEMMA 4.2. *Let $\mathbf{e}_h$ and ϵ_h be the errors of the WG finite element solutions arising from (3.4)–(3.5). Then we have*

$$a(\mathbf{e}_h, \mathbf{v}) - b(\mathbf{v}, \epsilon_h) = \psi_{\mathbf{u}}(\mathbf{v}), \tag{4.8}$$

$$b(\mathbf{e}_h, q) = 0 \tag{4.9}$$

for all $\mathbf{v}\in V_h^0$ and $q\in W_h$, where $\psi_{\mathbf{u}}(\mathbf{v}) = \nu\ell_{\mathbf{u}}(\mathbf{v}) + D_{\mathbf{u}}(\mathbf{v}) + \nu s(\mathbf{Q}_h\mathbf{u}, \mathbf{v})$ is a linear functional defined on V_h^0.

Proof. Since $\mathbf{u}$ and p satisfy (4.2) with $\eta = \mathbf{f}$, then from (4.3) we have

$$(\nu\nabla_w(\mathbf{Q}_h\mathbf{u}), \nabla_w\mathbf{v}) - (\nabla_w\cdot\mathbf{v}, \Pi_h p) = (\mathbf{f}, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) + \nu\ell_{\mathbf{u}}(\mathbf{v}) + D_{\mathbf{u}}(\mathbf{v}).$$

Adding $\nu s(Q_h\mathbf{u}, \mathbf{v})$ to both sides of the above equation gives

$$a(\mathbf{Q}_h\mathbf{u}, \mathbf{v}) - b(\mathbf{v}, \Pi_h p) = (\mathbf{f}, \mathbb{R}_{\mathbb{T}}(\mathbf{v})) + \nu\ell_{\mathbf{u}}(\mathbf{v}) + D_{\mathbf{u}}(\mathbf{v}) + \nu s(\mathbf{Q}_h\mathbf{u}, \mathbf{v}).$$

The difference of the above equation and (3.4) gives the following equation:

$$a(\mathbf{e}_h, \mathbf{v}) - b(\mathbf{v}, \epsilon_h) = \nu\ell_{\mathbf{u}}(\mathbf{v}) + D_{\mathbf{u}}(\mathbf{v}) + \nu s(\mathbf{Q}_h, \mathbf{v}) \tag{4.10}$$

for all $\mathbf{v}\in V_h^0$, which thus completes (4.8).

Then, we test (4.9) by $q\in W_h$ and use Lemma 3.3 to obtain

$$0 = (\nabla\cdot\mathbf{u}, q) = (\nabla_w\cdot\mathbf{Q}_h\mathbf{u}, q). \tag{4.11}$$

The difference between the above equation and (3.5) yields the following:

$$b(\mathbf{e}_h, q) = 0$$

for all $q\in W_h$. Thus, we complete the proof. □

LEMMA 4.3. *Assuming the finite element partition $\mathcal{T}_h$ is shape regular and $\mathbf{w}\in[H^2(\Omega)]^2$, the following estimates hold true for $\mathbf{v}\in V_h$:*

$$|s(\mathbf{Q}_h\mathbf{w}, \mathbf{v})| \lesssim h\|\mathbf{w}\|_2 |||\mathbf{v}|||, \tag{4.12}$$

$$|\ell_{\mathbf{w}}(\mathbf{v})| \lesssim h\|\mathbf{w}\|_2 |||\mathbf{v}|||, \tag{4.13}$$

$$|D_{\mathbf{w}}(\mathbf{v})| \lesssim \nu h\|\mathbf{w}\|_2 |||\mathbf{v}|||. \tag{4.14}$$

Proof. Equations (4.12) and (4.13) can be found in [32]. By the Cauchy–Schwarz inequality and estimate (3.19), this implies that

$$\begin{aligned} D_{\mathbf{w}}(\mathbf{v}) &= (\nu\Delta\mathbf{u}, \mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v})) \\ &\le \nu\|\mathbf{u}\|_2\Bigg(\sum_{T\in\mathcal{T}_h}\|\mathbf{v}_0 - \mathbb{R}_{\mathbb{T}}(\mathbf{v})\|_T^2\Bigg)^{1/2} \\ &\lesssim \nu h\|\mathbf{u}\|_2 |||\mathbf{v}|||. \end{aligned}$$

Thus, we complete the proof. □

4.2. Error estimates. In this section, we shall establish optimal order error estimates for the velocity approximation $\mathbf{u}_h$ in a $|||\cdot|||$-norm and L^2-norm, and for the pressure approximation p_h in the standard L^2-norm.

THEOREM 4.4. *Let* $(\mathbf{u}, p) \in [H_0^1(\Omega)]^2 \times L_0^2(\Omega)$, *and let* $(\mathbf{u}_h, p_h) \in V_h^0 \times Q_h$ *denote the unique solutions of* (1.1)–(1.3) *and* (3.4)–(3.5), *respectively, and assume the additional regularity* $\mathbf{u} \in [H^2(\Omega)]^2$. *Then the following estimates hold:*

$$|||\mathbf{u}_h - \mathbf{Q}_h\mathbf{u}||| \lesssim h\|\mathbf{u}\|_2, \tag{4.15}$$

$$\|p_h - \Pi_h p\| \lesssim \nu h\|\mathbf{u}\|_2. \tag{4.16}$$

Proof. By letting $\mathbf{v} = \mathbf{e}_h$ in (4.8) and $q = \epsilon_h$ in (4.9) and adding the two resulting equations, plugging in (4.12)–(4.14), we obtain

$$\begin{aligned} \nu|||\mathbf{e}_h|||^2 &= \psi_{\mathbf{u}}(\mathbf{e}_h) \\ &= \nu\ell_{\mathbf{u}}(\mathbf{e}_h) + D_{\mathbf{u}}(\mathbf{e}_h) + \nu s(\mathbf{Q}_h\mathbf{u}, \mathbf{e}_h) \\ &\lesssim \nu h\|\mathbf{u}\|_2|||\mathbf{e}_h|||, \end{aligned}$$

which gives (4.15). Error equation (4.8) implies

$$b(\mathbf{v}, \epsilon_h) = a(\mathbf{e}_h, \mathbf{v}) - \psi_{\mathbf{u}}(\mathbf{v}).$$

By combining (3.9), (4.15), the definition of $\psi_{\mathbf{u}}(\mathbf{v})$, and the estimates in (4.12)–(4.14), we arrive at

$$|b(\mathbf{v}, \epsilon_h)| \le C\nu h\big(\|\mathbf{u}\|_2\big)|||\mathbf{v}|||,$$

and then together with the inf-sup condition,

$$\|\epsilon_h\| \lesssim \nu h\|\mathbf{u}\|_2.$$

The above completes the proof. □

Remark 4.5. Furthermore, if under the additional regularity assumption $p \in H^1(\Omega)$, we have the following bound for the pressure error:

$$\|p_h - p\| \lesssim \nu h\|\mathbf{u}\|_2 + h\|p\|_1. \tag{4.17}$$

Remark 4.6. The error analysis results for Algorithm 3.2 show the following bound if $\mathbf{u} \in [H^2(\Omega)]^2$, $p \in H^1(\Omega)$:

$$|||\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h||| \lesssim h\bigg(\|\mathbf{u}\|_2 + \frac{1}{\nu}\|p\|_1\bigg). \tag{4.18}$$

The term $\nu^{-1}\|p\|_1$ on the right-hand side grows unboundedly as $\nu \to 0^+$, and therefore only breaks the accuracy of the velocity approximation. We refer to [32] for the proof.

5. Numerical results. In this section, several numerical examples in two dimensions are presented to validate the theoretical results. We shall consider triangular grids, rectangular grids, and polygonal grids. When the exact solutions have the regularity required by Theorem 4.4, we expect the following:

- If $(\mathbf{u}_h; p_h)$ are solutions from Algorithm 3.1 (denoted as WG 1), then
  - $|\!|\!|\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h|\!|\!| \le Ch\|\mathbf{u}\|_2$,
  - $\|\mathbf{Q}_0\mathbf{u} - \mathbf{u}_0\| \le Ch^2\|\mathbf{u}\|_2$,
  - $\|\Pi_h p - p_h\| \le C\nu h\|\mathbf{u}\|_2$,
  - $\|p - p_h\| \le Ch(\nu\|\mathbf{u}\|_2 + \|p\|_1)$.
- If $(\mathbf{u}_h; p_h)$ are solutions from Algorithm 3.2 (denoted as WG 2), then
  - $|\!|\!|\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h|\!|\!| \le Ch(\|\mathbf{u}\|_2 + \nu^{-1}\|p\|_1)$,
  - $\|\mathbf{Q}_0\mathbf{u} - \mathbf{u}_0\| \le Ch^2(\|\mathbf{u}\|_2 + \nu^{-1}\|p\|_1)$,
  - $\|\Pi p_h - p_h\| \le Ch(\nu\|\mathbf{u}\|_2 + \|p\|_1)$,
  - $\|p - p_h\| \le Ch(\nu\|\mathbf{u}\|_2 + \|p\|_1)$.

Furthermore, the discrete norm for divergence is defined as follows:

$$\|\nabla\cdot\mathbf{u}_0\| := \left(\sum_{T\in\mathcal{T}_h}\int_T |\nabla\cdot\mathbf{u}_0|^2\right)^{1/2}.$$

5.1. Triangular mesh.

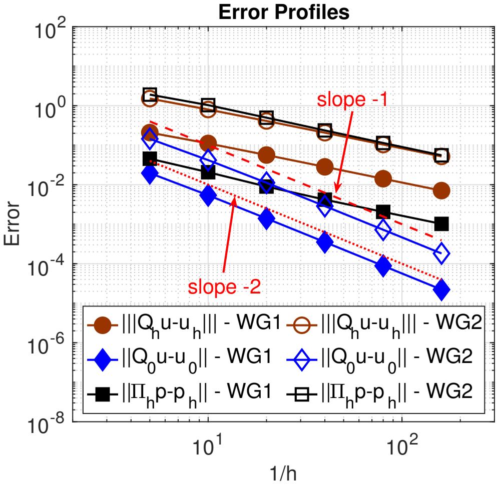

FIG. 4. *Example* 5.1.1*: Error profiles and convergence results on triangular mesh.*

5.1.1. Accuracy test. Let $\Omega = (0,1)^2$, $\nu = 1$, and let the exact solutions be chosen as follows:

$$\mathbf{u} = \begin{pmatrix} 10x^2(x-1)^2y(y-1)(2y-1) \\ -10x(x-1)(2x-1)y^2(y-1)^2 \end{pmatrix}, \quad p = 10(2x-1)(2y-1). \tag{5.1}$$

In this test, we shall compare the accuracy and convergence rates for Algorithms 1 and 2.

Error profiles and convergence results are plotted in Figure 4. When $\nu = 1$, we expect the convergence rate as $\mathcal{O}(h)$ for $|\!|\!|\cdot|\!|\!|$-error in velocity and for L^2-error in pressure, and $\mathcal{O}(h^2)$ for L^2-error of velocity, respectively, which are all validated in these numerical experiments. Also, the accuracy of the velocity variable from Algorithm 1 is ten times better than it was from Algorithm 2. The improvement in the accuracy of the pressure is more than 30 times.

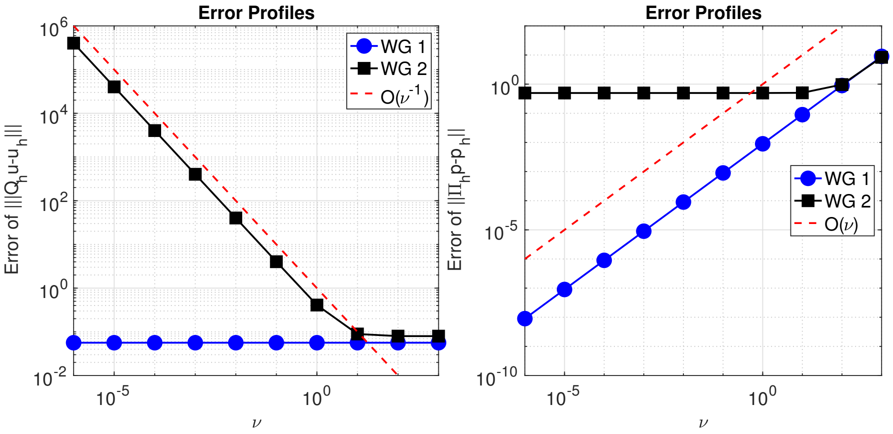

FIG. 5. *Example* 5.1.2*: Error profiles and convergence results on triangular mesh.*

5.1.2. Pressure-robustness with respect to ν. In this test, we choose the exact solution the same way as in (5.1) and perform WG simulation.

Let the simulation be performed on the grid with $h = 1/20$, and choose $\nu = 10^3, 10^2, 10^1, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$. The right-hand side source varies according to (1.1)–(1.3). In Figure 5, numerical results for the velocity error and pressure error are presented. The velocity error deteriorates for $\nu \to 0$ and is asymptotically proportional to $1/\nu$ (when $\nu \leq 1$) as predicted by the theory for Algorithm 3.2. This observation shows that Algorithm 3.2 is not pressure robust. In contrast, the numerical simulation from Algorithm 3.1 is pressure robust. The error in velocity from Algorithm 3.1 retains a constant as $\nu \to 0$, which validates the independence of the pressure variable. As opposed to the velocity error, the discrete pressure error for Algorithm 3.1 behaves in a robust manner for $\nu \to 0$. Furthermore, the new algorithm delivers not only a discrete pressure error which is robust, but even converges to 0 for $\nu \to 0$. Therefore, the new algorithm improves both velocity and pressure simultaneously.

5.1.3. No flow. The velocity field and pressure are chosen as

$$\mathbf{u}(\mathbf{x}) = \begin{pmatrix} u_1(\mathbf{x}) \\ u_2(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad p(\mathbf{x}) = -\frac{\mathrm{R}a}{2} y^2 + \mathrm{R}a\, y - \frac{\mathrm{R}a}{3}, \tag{5.2}$$

with $\mathrm{R}a := 1000$.

In this test, the velocity field is zero. However, we shall show Algorithm 3.2 will deliver a nonzero velocity. By modifying the right-hand side assembling as Algorithm 3.1, the numerical solution will be improved and then produce a good match with exact solutions.

Figure 6 shows the results of simulation from both algorithms. As we can see from the comparison, the performance of Algorithm 3.1 is significantly better than that of Algorithm 3.2. In fact, the approximation produces a velocity field with magnitude around 10^{-17}, which is almost zero.

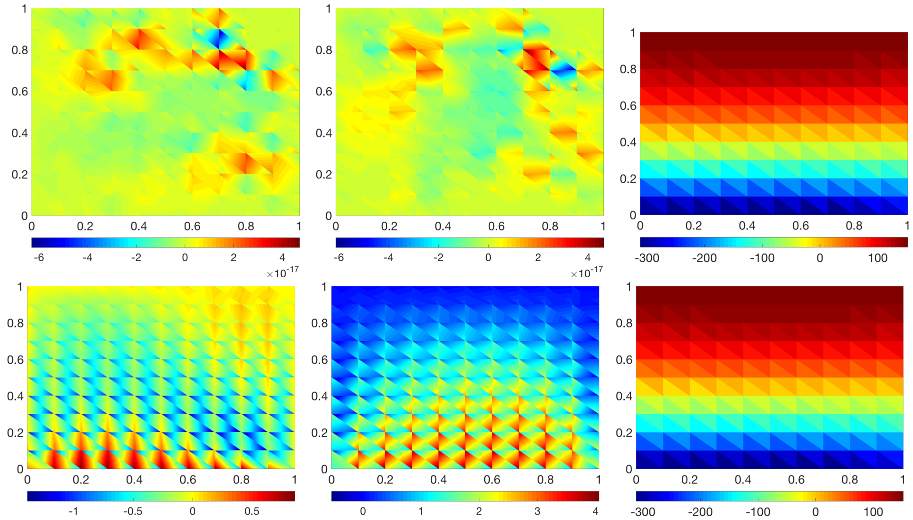

FIG. 6. *Example* 5.1.3*: Velocity approximation on the mesh with $h = 1/10$. Top: Numerical solution of u_1 (left), u_2 (middle), and p (right) from WG Algorithm* 1*. Bottom: Numerical solution of u_1 (left), u_2 (middle), and p (right) from WG Algorithm* 2*. The colormap bar denotes the magnitude of the solutions.*

The errors in the $\|\!|\cdot|\!\|$-norm for velocity are plotted in Figure 7 (left). One can observe the good match of numerical solutions from Algorithm 3.1 with zero field. Though Algorithm 3.2 cannot produce a zero velocity field, the error of velocity in this discrete H^1-norm retains the optimal rate in convergence as $\mathcal{O}(h)$. From the middle image in Figure 7, we find the performance of simulation in the pressure variable. It is noted that $\|\Pi_h p - p_h\| \approx 0$ for WG Algorithm 3.1. This result matches our analysis in (4.16). While the error between the numerical solution and exact solution for pressure converges at the order $\mathcal{O}(h)$ for Algorithms 3.1 and 3.2, one can also observe the improvement in accuracy for pressure by using Algorithm 3.1. The discrete norm for measuring the divergence has also been collected and reported in Figure 7 (right). It can be detected that the error is almost constant during mesh refinement, and its value is almost zero for Algorithm 3.1. The error from Algorithm 3.2 converges to zero at the second order.

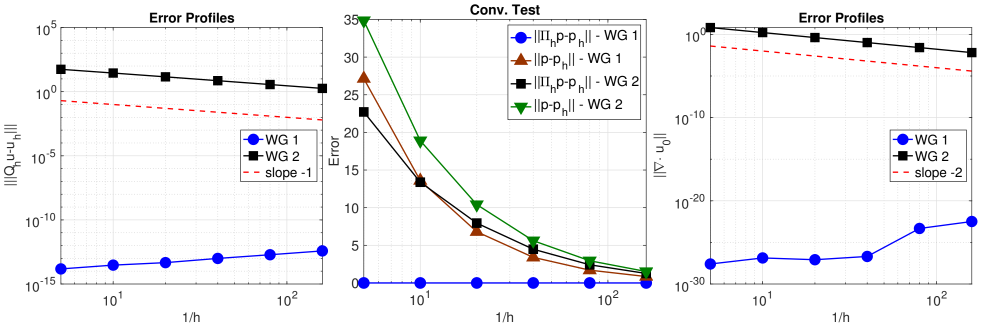

FIG. 7. *Example* 5.1.3*: Error profiles and convergence test: Results for velocity (left). Results for pressure (middle). Results for divergence (right).*

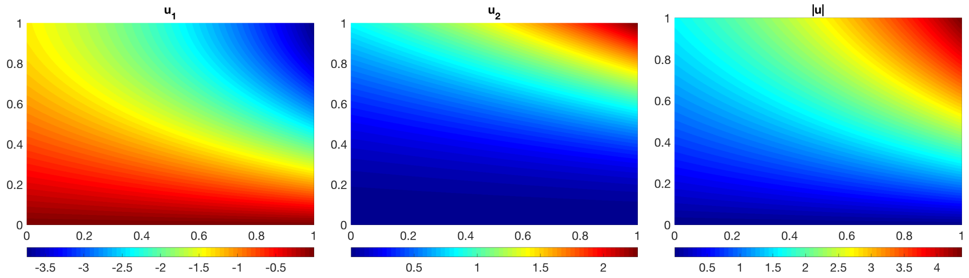

FIG. 8. *Example* 5.2.1*: Numerical solutions* $\mathbf{u}_h = (u_1, u_2)^\top$ *from WG algorithm* 3.1 *for* $\nu =$ 1e − 6*. Left: numerical solution of* u_1*. Middle: numerical solution of* u_2*. Right: numerical solution of magnitude* $|\mathbf{u}| := (u_1^2 + u_2^2)^{1/2}$*. The colormap bar shows the magnitude of solutions.*

5.2. Rectangular mesh. In this section, we shall present our numerical experiment based on the rectangular mesh.

5.2.1. Accuracy test. In the first example, let $\Omega = (0, 1)^2$, and choose the proper right-hand side function such that the following functions satisfy system (1.1)–(1.3):

$$\mathbf{u} = \begin{pmatrix} -e^x(y\cos y + \sin y) \\ e^x y \sin y \end{pmatrix}, \quad p = 2e^x \sin y. \tag{5.3}$$

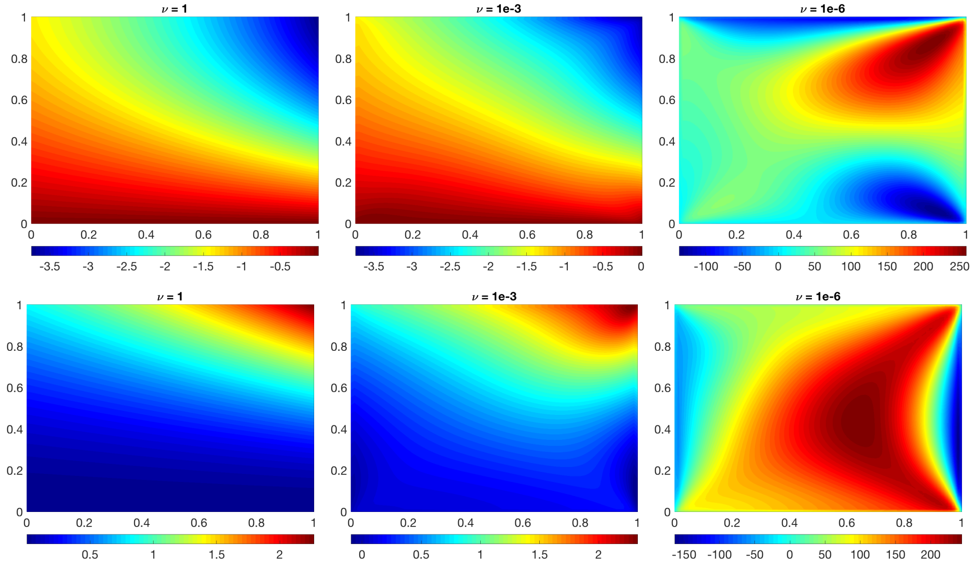

FIG. 9. *Example* 5.2.1*: Numerical solutions* $\mathbf{u}_h = (u_1, u_2)^\top$ *from WG algorithm* 3.2 *for different values in* ν*. Top: numerical solution of* u_1*. Bottom: numerical solution of* u_2*. The colormap bar shows the magnitude of each component.*

The external source $\mathbf{f}$ is chosen such that the above functions $\mathbf{u}$ and p are solutions to the system (1.1)–(1.3). Since $-\Delta\mathbf{u} = (-2e^x \sin y, -2e^x \cos y)^\top$, it is easy to check that when $\nu = 1$, we have a homogeneous equation with $\mathbf{f} = 0$. When $\nu \neq 0$, $\mathbf{f}$ is a nonzero function. We perform two WG algorithms on the rectangular mesh with $h = 1/80$. The numerical simulations are illustrated in Figures 8 and 9. As we can see from Figure 9, when ν is close to 1, Algorithm 3.2 will deliver a good numerical solution (the exact solution in the velocity field is similar to the left column in Figure 9). However, $\nu = 1e-6$ will break the simulation and produce incorrect results. But for Algorithm 3.1, in all cases, we can observe good approximation results. We only plot the numerical solutions of the velocity field in the case when $\nu = 1e-6$ in Figure 8.

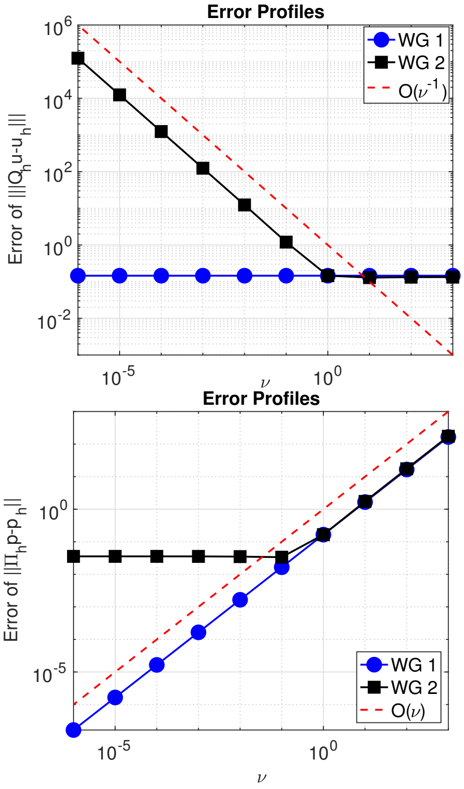

FIG. 10. *Example* 5.2.2*: Error profiles with respect to* ν *on rectangular mesh with* $h = 1/20$. *Top: convergence results in velocity. Bottom: convergence results in pressure.*

5.2.2. Robustness test. To clearly see the consequences of pressure-robustness in the developed scheme, Figure 10 shows the errors for different values of $\nu = 10^j$ ($j = -6, \ldots, 3$) for the WG Algorithms 3.1 and 3.2 on the rectangular mesh with $h = 1/20$. We shall take the same exact solutions as (5.3). There are several observations to make:

- For $\nu > 1$ the irrotational part in the right-hand side $\mathbf{f}$ is not larger than the divergence-free part. In this situation both methods deliver similar errors.
- For $\nu = 1$ the right-hand side $\mathbf{f} = 0$, and thus Algorithm 3.1 is exactly the same as Algorithm 3.2, so we expect the same computational results for this case.
- For $\nu < 1$ the irrotational part in the right-hand side $\mathbf{f}$ begins to dominate and so does the pressure-dependent term in the a priori error estimate for Algorithm 3.2. As predicted by these estimates, the errors of Algorithm 3.2 deteriorate and scale with $1/\nu$. Algorithm 3.1 does not see the dependency

on ν due to the divergence-free test functions in the right-hand side.

- The velocity error of Algorithm 3.2 depends on ν for $\nu < 1$, which is shown as an order $\mathcal{O}(\nu^{-1})$. The error corresponding to pressure, measured both by $\|\Pi_h p - p_h\|$ and $\|p - p_h\|$, remains as a constant for $\nu < 1$ and increases for $\nu > 1$ at the order $\mathcal{O}(\nu)$.
- The velocity error of Algorithm 3.1 is independent of ν, which is shown almost as a constant for varying values in ν. The pressure error, measured by $\|p - p_h\|$, increases for large $\nu > 1$ at $\mathcal{O}(\nu)$. This agrees with the error estimate.

5.3. Polygonal mesh. In this section, we shall perform WG simulations on the polygonal mesh. The meshes we considered have been plotted in Figure 11. The next level of meshes are generated to ensure the number of cells is almost double that of the previous level.

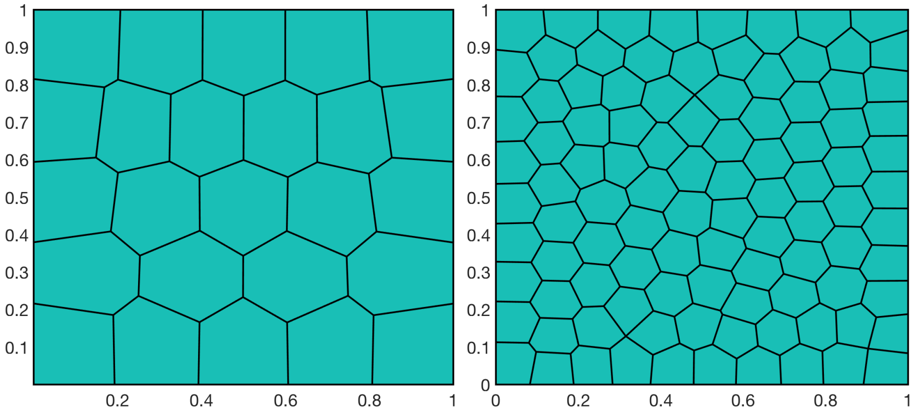

FIG. 11. *Illustration of polygonal Mesh: Mesh Level* 1 *(left); Mesh Level* 2 *(right).*

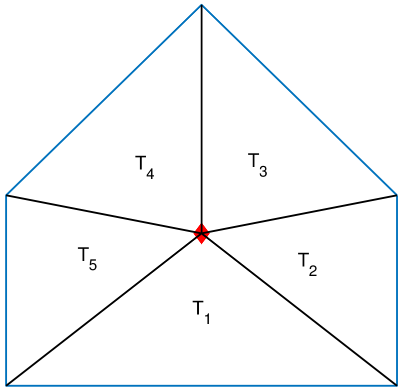

FIG. 12. *Example* 5.3.1*: Illustration of numerical integration rules on a polygonal cell.*

5.3.1. Impact of quadrature rules. In the polygonal simulation, since the basis $\Theta_i \in \mathbb{CW}_0(T)$ may be nonpolynomial functions, the right-hand side assembling $\sum_T \int_T \mathbf{f}\cdot\mathbb{R}_{\mathbb{T}}(\mathbf{v})dT$ usually cannot be computed exactly. This leads to some quadrature error that pollutes the pressure-robustness property. Therefore, the error, though

Table 1

Example 5.3.1*: Error profiles and convergence results for WG Algorithm* 3.1.

Mesh	$\|\mathbf{e}_h\|$	Rate	$\|\mathbf{e}_0\|$	Rate	$\|\epsilon_h\|$	Rate	$\|p - p_h\|$	Rate
4 Points Gaussian quadrature								
1	2.90E-02		1.40E-03		4.62E-02		3.80E+01	
2	7.47E-03	1.96	1.73E-04	3.02	1.76E-02	1.39	2.05E+01	0.89
3	1.88E-03	1.99	2.14E-05	3.01	6.38E-03	1.46	1.06E+01	0.95
4	4.71E-04	2.00	2.66E-06	3.01	2.28E-03	1.48	5.40E+00	0.97
5	1.18E-04	2.00	3.31E-07	3.01	8.12E-04	1.49	2.72E+00	0.99
9 Points Gaussian quadrature								
1	5.61E-04		2.76E-05		8.52E-04		3.80E+01	
2	1.44E-04	1.97	3.37E-06	3.04	3.24E-04	1.40	2.05E+01	0.89
3	3.60E-05	2.00	4.12E-07	3.03	1.17E-04	1.46	1.06E+01	0.95
4	8.97E-06	2.00	5.08E-08	3.02	4.19E-05	1.48	5.40E+00	0.97
5	2.24E-06	2.00	6.30E-09	3.01	1.49E-05	1.49	2.72E+00	0.99
13 Points Gaussian quadrature								
1	7.71E-05		3.78E-06		1.12E-04		3.80E+01	
2	1.97E-05	1.97	4.61E-07	3.04	4.31E-05	1.37	2.05E+01	0.89
3	4.94E-06	2.00	5.65E-08	3.03	1.58E-05	1.45	1.06E+01	0.95
4	1.23E-06	2.00	6.98E-09	3.02	5.65E-06	1.48	5.40E+00	0.97
5	3.08E-07	2.00	8.66E-10	3.01	2.01E-06	1.49	2.72E+00	0.99

theoretically pressure-independent, shows some dependence on pressure that can be reduced by increasing the number of points in quadrature rules.

For a polygonal cell (assume it is convex), we can split the cell into several small triangles by connecting its centroid with all vertexes. Then on each small triangle, the usual Gaussian quadrature rules can be directly applied. Figure 12 plots one example of splitting a polygonal cell into five small triangles.

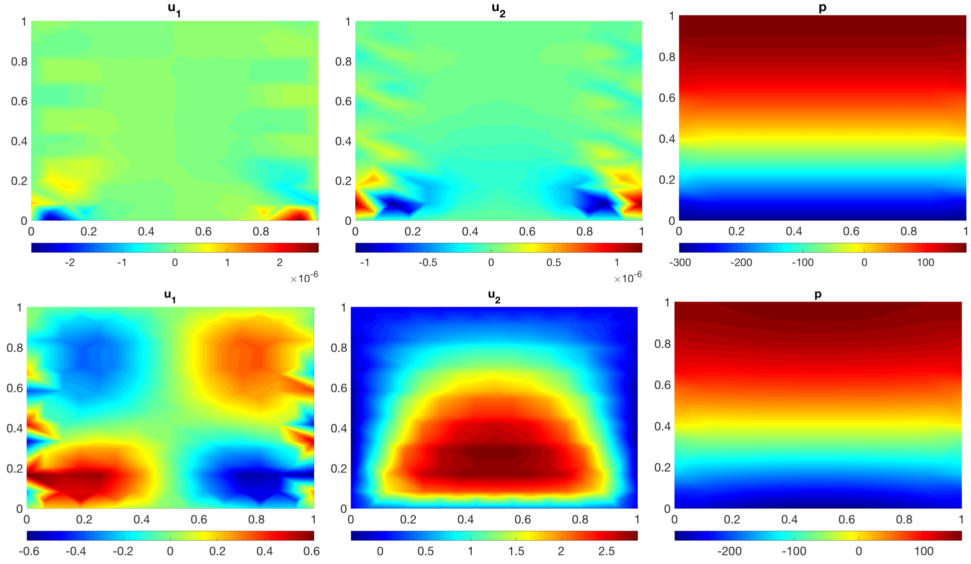

Fig. 13. *Example* 5.3.2*: Numerical solutions on Mesh Level* 1*. Top: Numerical solution* u_1 *(left),* u_2 *(middle),* p *(right) from Algorithm* 3.1*. Bottom: Numerical solution* u_1 *(left),* u_2 *(middle),* p *(right) from Algorithm* 3.2*. The colormap bar shows the magnitude of each component for* u_1*,* u_2*, and* p*.*

TABLE 2
Example 5.3.1*: Error profiles and convergence results for WG Algorithm* 3.2.

Mesh	$\|\|\|\mathbf{e}_h\|\|\|$	Rate	$\|\mathbf{e}_0\|$	Rate	$\|\epsilon_h\|$	Rate	$\|p-p_h\|$	Rate
4 Points Gaussian quadrature								
1	6.76E+01		8.98E+00		3.15E+01		4.94E+01	
2	3.86E+01	0.81	2.83E+00	1.67	1.32E+01	1.26	2.44E+01	1.02
3	2.06E+01	0.91	7.81E-01	1.86	4.60E+00	1.52	1.16E+01	1.07
4	1.06E+01	0.96	2.04E-01	1.94	1.47E+00	1.64	5.59E+00	1.05
5	5.36E+00	0.98	5.18E-02	1.97	4.50E-01	1.71	2.76E+00	1.02
9 Points Gaussian quadrature								
1	6.76E+01		8.98E+00		3.15E+01		4.94E+01	
2	3.86E+01	0.81	2.83E+00	1.67	1.32E+01	1.26	2.44E+01	1.02
3	2.06E+01	0.91	7.81E-01	1.86	4.60E+00	1.52	1.16E+01	1.07
4	1.06E+01	0.96	2.04E-01	1.94	1.47E+00	1.64	5.59E+00	1.05
5	5.36E+00	0.98	5.18E-02	1.97	4.50E-01	1.71	2.76E+00	1.02
13 Points Gaussian quadrature								
1	6.76E+01		8.98E+00		3.15E+01		4.94E+01	
2	3.86E+01	0.81	2.83E+00	1.67	1.32E+01	1.26	2.44E+01	1.02
3	2.06E+01	0.91	7.81E-01	1.86	4.60E+00	1.52	1.16E+01	1.07
4	1.06E+01	0.96	2.04E-01	1.94	1.47E+00	1.64	5.59E+00	1.05
5	5.36E+00	0.98	5.18E-02	1.97	4.50E-01	1.71	2.76E+00	1.02

The exact solutions are taken as (5.2); error profiles are reported in Tables 1 and 2 for Gaussian quadrature rules of 4, 9, 13 points in Algorithms 3.1 and 3.2. This test has zero velocity field, and the exact integration methods should produce zero numerical solutions in velocity. Thus, for the exact integration, one can expect the zero errors between numerical and exact solutions. However, due to the error in numerical integration methods, we cannot achieve such values in the simulation. One can observe that with better quadrature rules, the errors are significantly reduced in Algorithm 3.1 on the polygonal meshes. Also, super convergence rates are obtained from this table. However, there is almost no improvement for Algorithm 3.2 by increasing quadrature points. Further, the errors of velocity measured in the $|||\cdot|||$-norm and L^2-norm converge at the orders $\mathcal{O}(h^3)$ and $\mathcal{O}(h^2)$ for Algorithm 3.1; at the orders $\mathcal{O}(h^2)$ and $\mathcal{O}(h)$ for Algorithm 3.2. The error of pressure measured by the L^2-norm is dominated by the projection error $\|p-\Pi_h p\|$, and thus is at order $\mathcal{O}(h)$ for both algorithms.

5.3.2. Accuracy test. Now, we plot the numerical solutions for Algorithms 3.1 and 3.2 on Mesh Level 1. The exact solutions are taken as (5.2). We can clearly detect the improvement of Algorithm 3.1 in Figure 13.

5.3.3. Robustness on ν. In this section, we will report the robustness results on the polygonal mesh. The exact solutions are chosen as (5.3). The numerical experiments are carried out for $\nu = 10^{-6}, \ldots, 10^3$, and the results are summarized in Figure 14 for Mesh Level 2. Again, we can obtain similar conclusions as section 5.2.2.

6. Concluding remarks. In this paper, we have developed a new WG scheme for solving Stokes equation on polygonal mesh. The new method modifies the right-hand side assembling by projecting the test function onto the H(div)-conforming space. This new method shows the features of divergence preserving and robustness on viscosity variable ν. Also, the numerical performance can be significantly improved by using such techniques.

The results in this manuscript can be applied to solve Navier–Stokes equations with high Reynolds numbers where inertia terms play a role. In our ongoing project,

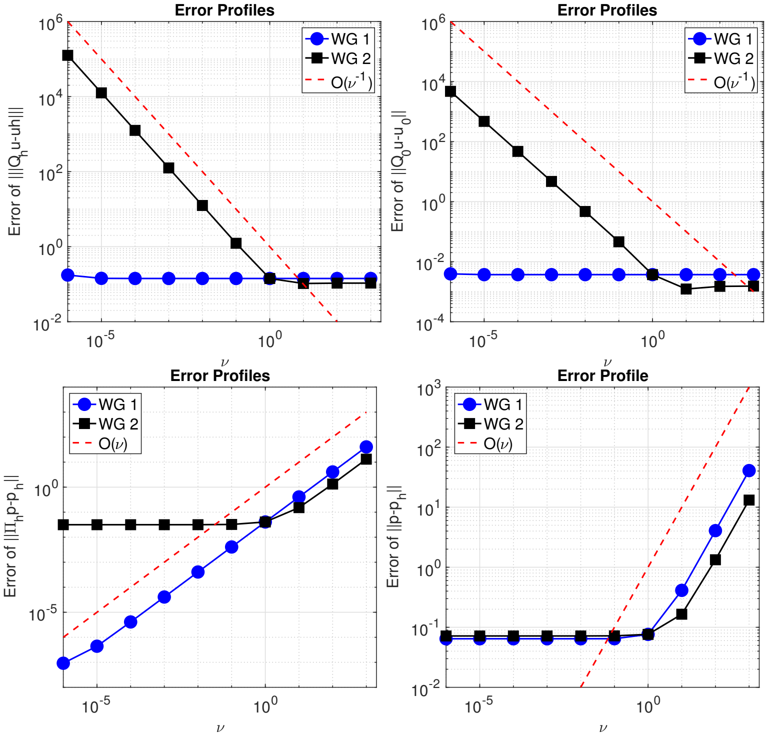

FIG. 14. *Example* 5.3.3*: Error profiles with respect to ν on polygonal mesh Level* 2*. Top: performance of velocity. Bottom: performance of pressure.*

the proposed scheme has been used to solve Stokes equations and thus provide an initial guess of the Newton and Picard iteration for the related nonlinear problem. A better initial guess guarantees better convergence for the nonlinear solver. Another application for the proposed scheme is to develop the uniform numerical scheme for the Brinkman equation. Because of the viscosity independence feature, this algorithm enables the accurate simulation both in the Stokes dominated regions with significant viscosity and the Darcy dominated regions with small values in viscosity.

Another extension includes the implementation of three-dimensional polyhedral meshes. For the three-dimensional element, the lowest order H(div) element is constructed in [4]. Our algorithm can be directly applied to the polyhedral element on a three-dimensional $\mathbb{CW}_0$ basis. However, a good algorithm for generating nested polyhedra meshes requires further investigation. We shall leave these topics as future areas of research.

Acknowledgments. We thank Dr. Alexander Linke, Dr. Yanqiu Wang, and Dr. Xiu Ye for their insights and helpful discussions. Also, we thank the anonymous reviewers for their constructive comments and suggestions.

REFERENCES

[1] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. Marini, and A. Russo, *Basic principles of virtual element methods*, Math. Models Methods Appl. Sci., 23 (2013), pp. 199–214.

[2] L. Beirão da Veiga, F. Brezzi, and L. Marini, *Virtual elements for linear elasticity problems*, SIAM J. Numer. Anal., 51 (2013), pp. 794–812, https://doi.org/10.1137/120874746.

[3] C. Brennecke, A. Linke, C. Merdon, and J. Schöberl, *Optimal and pressure-independent L^2 velocity error estimates for a modified Crouzeix-Raviart Stokes element with BDM reconstructions*, J. Comput. Math., 33 (2015), pp. 191–208.

[4] W. Chen and Y. Wang, *Minimal Degree H(curl) and H(div) conforming finite elements on polytopal meshes*, Math. Comp., 86 (2017), pp. 2053–2087.

[5] L. Chen and F. Wang, *A divergence free weak virtual element method for the Stokes problem on polytopal meshes*, J. Sci. Comput., 78 (2019), pp. 864–886.

[6] L. Chen, M. Wang, and L. Zhong, *Convergence analysis of triangular MAC schmes for two dimensional Stokes equations*, J. Sci. Comput., 63 (2015), pp. 716–744.

[7] B. Cockburn, (2018). *The Weak Galerkin Methods are Rewritings of the Hybridizable Discontinuous Galerkin Methods*, preprint, https://doi.org/10.1137/abs/1812.08146, 2018.

[8] B. Cockburn, D. A. Di Pietro, and A. Ern, *Bridging the hybrid high-order and hybridizable discontinuous Galerkin methods*, ESAIM Math. Model. Numer. Anal., 50 (2016), pp. 635–650.

[9] B. Cockburn, G. Kanschat, and D. Schötzau, *A note on discontinuous Galerkin divergence-free solutions of the Navier-Stokes equations*, J. Sci. Comput., 31 (2007), pp. 61–73.

[10] B. Cockburn, D. Di Pietro, and A. Ern, *Bridging the hybrid high-order and hybridizable discontinuous Galerkin methods*, ESAIM Math. Model. Numer. Anal., 50 (2016), pp. 635–650.

[11] R. Falk and M. Neilan, *Stokes complexes and the construction of stable finite elements with pointwise mass conservation*, SIAM J. Numer. Anal., 51 (2013), pp. 1308–1326, https://doi.org/10.1137/120888132.

[12] N. Gauger, A. Linke, and P. Schroeder, *On high-order pressure-robust space discretisations, their advantages for incompressible high Reynolds number generalised Beltrami flows and beyond*, SMAI J. Comput. Math., 5 (2019), pp. 89–129.

[13] J. Guzmán and M. Neilan, *Conforming and divergence free Stokes elements on general triangular meshes*, Math. Comput., 83 (2014), pp 15–36.

[14] J. Guzmán and M. Neilan, *Conforming and divergence-free Stokes elements in three dimensions*, IMA J. Numer. Anal., 34 (2014), pp. 1489–1508.

[15] E. Jenkins, V. John A. Linke, and L. Rebholz, *On the parameter choice in grad-div stabilization for the Stokes equations*, Adv. Compute. Math., 40 (2014), pp. 491–516.

[16] V. John, A. Linke, C. Merdon, M. Neilan, and L. Rebholz, *On the divergence constraint in mixed finite element methods for incompressible flows*, SIAM Rev., 59 (2017), pp. 492–544, https://doi.org/10.1137/15M1047696.

[17] P. Lederer, *Pressure-robust Discretizations for Navier-Stokes Equations: Divergence-free Reconstruction for Taylor-Hood Elements and High Order Hybrid Discontinuous Galerkin Methods*, Master's thesis, Vienna Technical University, Vienna, Austria, 2016.

[18] P. Lederer and J. Schöberl, *Polynomial robust stability analysis for h(div)-conforming finite elements for the Stokes equations*, IMA J. Numer. Anal., 38 (2018), pp. 1832–1860.

[19] A. Linke, *A divergence-free velocity reconstruction for incompressible flows*, C. R. Math. Acad. Sci. Paris, 350 (2012), pp. 837–840.

[20] A. Linke, *On the role of the Helmholtz decomposition in mixed methods for incompressible flows and a new variational crime*, Comput. Methods Appl. Mech. Eng., 268 (2014), pp. 782–800.

[21] A. Linke, G. Matthies, and L. Tobiska, *Robust arbitrary order mixed finite element methods for the incompressible Stokes equations with pressure independent velocity errors*, ESAIM Math. Model. Numer. Anal., 50 (2016), pp. 289–309.

[22] A. Linke and C. Merdon, *Pressure-robustness and discrete Helmholtz projectors in mixed finite element methods for the incompressible Navier-Stokes equations*, Comput. Methods Appl. Mech. Eng., 311 (2016), pp. 304–326.

[23] K. Lipnikov, G. Manzini, and M. Shashkov, *Mimetic finite difference method*, J. Comput. Phys., 257 (2014), pp. 1163–1227.

[24] J. Liu, S. Tavener, and Z. Wang, *Lowest-order weak Galerkin finite element method for Darcy flow on convex polygonal meshes*, SIAM J. Sci. Comput., 40 (2018), pp. B1229–B1252, https://doi.org/10.1137/M1145677.

[25] M. Olshanskii and A. Olshanskii, *Grad-div stabilization for Stokes equations*, Math. Comp., 73 (2004), pp. 1699–1718.

[26] M. Olshanskii, G. Lube, T. Heister, and J. Löwe, *Grad-div stabilization and subgrid pressure models for the incompressible Navier-Stokes equations*, Comput. Methods Appl. Mech. Engrg., 198 (2009), pp. 3975–3988.

[27] D. Di Pietro and A. Ern, *A hybrid high-order locking-free method for linear elasticity on general meshes*, Comput. Methods Appl. Mech. Engrg., 283 (2015), pp. 1–21.

[28] Di Pietro, A. Ern, A. Linke, and F. Schieweck, *A discontinuous skeletal method for the viscosity-dependent Stokes problem*, Comput. Methods Appl. Mech. Engrg., 306 (2016), pp. 175–195.

[29] L. Scott and M. Vogelius, *Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials*, ESAIM Math. Model. Numer. Anal., 19 (1985), pp. 111–143.

[30] J. Wang and X. Ye, *A weak Galerkin finite element method for second-order elliptic problems*, J. Comput. Appl. Math., 241 (2013), pp. 103–115.

[31] J. Wang and X. Ye, *A weak Galerkin mixed finite element method for second-order elliptic problems*, Math. Comp., 83 (2014), pp. 2101–2126.

[32] J. Wang and X. Ye, *A weak Galerkin finite element method for the Stokes equations*, Adv. Comput. Math., 42 (2016), pp. 155–174.

[33] J. Wang and X. Ye, *New finite element methods in computational fluid dynamics by H(div) elements*, SIAM J. Numer. Anal., 45 (2007), pp. 1269–1286, https://doi.org/10.1137/060649227.

[34] J. Wang and Y. Xiu, *The Basics of Weak Galerkin Finite Element Methods*, preprint, https://doi.org/10.1137/abs/1901.10035, 2019.

[35] S. Zhang, *Divergence-free finite elements on tetrahedral grids for $k \geq 6$*, Math. Comput., 80 (2011), pp. 669–695.