Journal of Scientific Computing (2021) 89:26
https://doi.org/10.1007/s10915-021-01634-5

Development of Pressure-Robust Discontinuous Galerkin Finite Element Methods for the Stokes Problem

Lin Mu[1] · Xiu Ye[2] · Shangyou Zhang[3]

Received: 12 May 2021 / Revised: 10 August 2021 / Accepted: 25 August 2021 /
Published online: 12 September 2021

Abstract

Pressure-robustness is an essential demand for the incompressible fluid simulation. In this paper, we develop the enhanced discontinuous Galerkin (DG) finite element methods for solving Stokes equations in the primary velocity-pressure formulation to achieve pressure-robustness. The velocity reconstruction operator has been designed for discontinuous functions and utilized to modify the external source assembling. The new schemes stay almost the same as that in the existing DG schemes but only differ for the source terms. The conforming discontinuous Galerkin and symmetric interior penalty DG have been employed to demonstrate the enhancement. Optimal-order error estimates are established for the corresponding numerical approximation in various norms. Several numerical experiments are performed to validate our theoretical conclusions.

Keywords Discontinuous Galerkin · Finite element methods · Stokes equations · Pressure-robustness · Optimal rate in convergence

Mathematics Subject Classification 65N15 · 65N30 · 76D07 · 35B45 · 35J50

The research of the second author was supported in part by National Science Foundation Grant DMS-1620016.

✉ Lin Mu
linmu@uga.edu

Xiu Ye
xxye@ualr.edu

Shangyou Zhang
szhang@udel.edu

[1] Department of Mathematics, University of Georgia, Athens, GA 30605, USA

[2] Department of Mathematics, University of Arkansas at Little Rock, Little Rock, AR 72204, USA

[3] Department of Mathematical Sciences, University of Delaware, Newark, DE 19716, USA

1 Introduction

In this paper, we consider the Stokes problem: find the velocity $\mathbf{u}$ and the pressure p such that

$$-\nu\Delta\mathbf{u}+\nabla p=\mathbf{f} \quad \text{in } \Omega, \tag{1}$$

$$\nabla\cdot\mathbf{u}=0 \quad \text{in } \Omega, \tag{2}$$

$$\mathbf{u}=0 \quad \text{on } \partial\Omega, \tag{3}$$

where Ω is a polygonal or polyhedral domain in $\mathbb{R}^d$ $(d=2,3)$. Here ν is the kinematic viscosity and $\mathbf{f}$ is the external body force. The Stokes equation has been widely used in realistic applications. In order to derive a stable finite element numerical scheme, the finite element pairs for velocity and pressure need to satisfy the inf-sup condition. However the classical stable finite element methods for incompressible Stokes problem are usually not pressure-robust, i.e., their velocity error is pressure dependent, for example [8,11,15,24], shown as follows,

$$\|\nabla(\mathbf{u}-\mathbf{u}_h)\|\leq C\inf_{\mathbf{w}\in\mathbf{V}_h}\|\nabla(\mathbf{u}-\mathbf{w})\|+\frac{1}{\nu}\inf_{q\in W_h}\|p-q\|.$$

Here $\mathbf{V}_h$ and W_h are the stable finite element spaces to approximate velocity and the pressure. We observe that a pressure dependent term that can be relatively large in the case of small viscosity ν. This may cause the bad simulations in many cases: as $\nu\ll 1$, one may observe large errors in velocity; bad approximation or singularity in pressure may affect the velocity simulation due to the coupling of velocity and pressure. To reduce these coupling effects, one would have to assume that the pressure is smooth enough and also employ higher scheme order, which may introduce extra difficulties in the simulations.

The concept of pressure-robustness has been proposed to remove the pressure dependency in velocity simulation. In this category, nowadays, many divergence-free elements have been developed for two dimensional problems [9,29] and three dimensional problems [35]. The enrichment of the H(div; Ω)-conforming elements with divergence-free rational shape-functions has been proposed in [12,13]. Cockburn [7] and Wang [30] proposed divergence free schemes, which introduced tangential penalty to modify the variational formulation. Chen proposed to introduce a discrete dual curl operator [5] to implement divergence free MAC scheme on triangular grids. There are other approaches such as adding grad-div stabilization [16,26,27] and etc. An alternative method to get pressure independent approximation is introduced recently by Linke [20,21] as employing a divergence preserving velocity reconstruction operator. a velocity reconstruction is presented in [20] to map discrete divergence free test functions onto exactly divergence free test functions which is applied only on the right hand side for the Stokes equations. A sequence of papers are investigating approaches for discontinuous pressure elements [2,17,20–22] and continuous pressure elements [18,19]. With the help of velocity reconstruction, the pressure-robust schemes leading to a pressure independent velocity error estimate [20,21,23–25]

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

Such schemes can be directly used in the existing inf-sup stable schemes [4,8] to achieve the robustness without compromising the computational accuracy.

In this paper, we focus on developing new pressure-robust numerical discretizations by modifying the existing discontinuous Galerkin scheme for viscosity dependent Stokes equations with the velocity reconstruction technique. The conforming discontinuous Galerkin

[32] (CDG) and symmetric interior penalty discontinuous Galerkin (IPDG) finite element methods will be employed to demonstrate the enhancement. Comparing to the existing CDG and IPDG schemes for Stokes equation, our scheme only modifies the body force assembling but remains the same stiffness matrix. The rigorous error analysis is shown for the desired pressure-robustness and convergence results. Several numerical experiments are carried out for demonstrating the advantages of the new schemes.

This paper is organized as follows. Brief review regarding basis functions and velocity reconstruction operator will be presented in Sect. 2. The discretization is developed in Sect. 3 and the well-posedness is discussed in the same section. Then the error equations are given in Sect. 4. Our main results regarding error estimates are stated in Sect. 5. Section 6 presents several numerical examples for Stokes equations. Finally, this paper is summarized with concluding remarks in Sect. 7.

2 Preliminary

We use standard definitions for the Sobolev spaces $H^s(D)$ and their associated inner products $(\cdot,\cdot)_{s,D}$, norms $\|\cdot\|_{s,D}$, and seminorms $|\cdot|_{s,D}$ for $s \geq 0$. When $D = \Omega$, we drop the subscript D in the norm and inner product notation. We shall also assume the source term $\mathbf{f} \in \mathbf{L}^2(\Omega)$. In the following, we shall assume C is a generic constant which may take different values in different locations.

2.1 Finite Element Spaces

Let $\mathcal{T}_h$ be a partition of the domain Ω consisting of shape-regular triangles in two dimensional space or tetrahedrons in three dimensional space. Denote by $\mathcal{E}_h$ the set of all flat faces in $\mathcal{T}_h$, and let $\mathcal{E}_h^0 = \mathcal{E}_h \backslash \partial\Omega$ be the set of all interior faces. For $k \geq 1$ and given $\mathcal{T}_h$, define two finite element spaces, for approximating velocity

$$\mathbf{V}_h = \left\{\mathbf{v} \in \boldsymbol{L}^2(\Omega) : \mathbf{v}|_T \in [P_k(T)]^d, \quad \forall T \in \mathcal{T}_h\right\} \tag{4}$$

and for approximating pressure

$$W_h = \left\{q \in L_0^2(\Omega) : \ q|_T \in P_{k-1}(T), \quad \forall T \in \mathcal{T}_h\right\}. \tag{5}$$

Let T_1 and T_2 be two elements in $\mathcal{T}_h$ sharing $e \in \mathcal{E}_h$. For $e \in \mathcal{E}_h$ and $\mathbf{v} \in \mathbf{V}_h + \boldsymbol{H}_0{}^1(\Omega)$, the jump $[\![\mathbf{v}]\!]$ is defined as

$$[\![\mathbf{v}]\!] = \mathbf{v} \quad \text{if } e \subset \partial\Omega, \quad [\![\mathbf{v}]\!] = \mathbf{v}|_{T_1} - \mathbf{v}|_{T_2} \quad \text{if } e \in \mathcal{E}_h^0. \tag{6}$$

The order of T_1 and T_2 is not essential. For $e \in \mathcal{E}_h$ and $\mathbf{v} \in \mathbf{V}_h + \boldsymbol{H}_0^1(\Omega)$, the average $\{\mathbf{v}\}$ is defined as

$$\{\mathbf{v}\} = 0 \quad \text{if } e \subset \partial\Omega, \quad \{\mathbf{v}\} = \frac{1}{2}(\mathbf{v}|_{T_1} + \mathbf{v}|_{T_2}) \quad \text{if } e \in \mathcal{E}_h^0, \tag{7}$$

For a function $\mathbf{v} \in \mathbf{V}_h + \boldsymbol{H}_0{}^1(\Omega)$, its weak gradient $\nabla_w \mathbf{v}|_T \in [\mathbf{RT}_k(T)]^d$ is a piecewise polynomial tensor satisfies the following equation for each $T \in \mathcal{T}_h$,

$$(\nabla_w \mathbf{v}, \ \tau)_T = -(\mathbf{v}, \ \nabla \cdot \tau)_T + \langle\{\mathbf{v}\}, \ \tau \cdot \mathbf{n}\rangle_{\partial T} \quad \forall \tau \in [\mathbf{RT}_k(T)]^d, \tag{8}$$

and its weak divergence $\nabla_w \cdot \mathbf{v}|_T \in P_{k-1}(T)$ is a piecewise polynomial satisfies the following equation for each $T \in \mathcal{T}_h$,

$$(\nabla_w \cdot \mathbf{v},\ q)_T = -(\mathbf{v},\ \nabla q)_T + \langle \{\mathbf{v}\} \cdot \mathbf{n},\ q\rangle_{\partial T} \quad \forall q \in P_{k-1}(T). \tag{9}$$

First we introduce two semi-norms $|||\mathbf{v}|||$ and $\|\mathbf{v}\|_{1,h}$ for any $\mathbf{v} \in \mathbf{V}_h + \boldsymbol{H}_0{}^1(\Omega)$ as follows:

$$|||\mathbf{v}|||^2 = \sum_{T\in\mathcal{T}_h} (\nabla_w \mathbf{v}, \nabla_w \mathbf{v})_T, \tag{10}$$

$$\|\mathbf{v}\|_{1,h}^2 = \sum_{T\in\mathcal{T}_h} \|\nabla \mathbf{v}\|_T^2 + \sum_{e\in\mathcal{E}_h} h_e^{-1} \|[\![\mathbf{v}]\!]\|_e^2. \tag{11}$$

The following lemma has been established in [1,33].

Lemma 1 *Both $\|\mathbf{v}\|_{1,h}$ and $|||\mathbf{v}|||$ define a norm for $\mathbf{v} \in \mathbf{V}_h$ and the following norm equivalence holds true*

$$C_1\|\mathbf{v}\|_{1,h} \le |||\mathbf{v}||| \le C_2\|\mathbf{v}\|_{1,h} \quad \forall \mathbf{v} \in \mathbf{V}_h. \tag{12}$$

2.2 Velocity Reconstruction Operators and Their Properties

Let $\mathbb{Q}_h$, $\mathbf{Q}_h$ and Q_h be the element-wise defined L^2 projections onto the local spaces $[\mathbf{RT}_k(T)]^d$, $[P_k(T)]^d$ and $P_{k-1}(T)$ for $T \in \mathcal{T}_h$, respectively. Below, we shall introduce another projection operator Π_h.

Definition 1 For $\mathbf{v} \in \mathbf{V}_h$, we define the projection $\Pi_h\mathbf{v} \in \mathbf{RT}_{k-1}(T)$ such that $\Pi_h\mathbf{v} \in H(\mathrm{div}, \Omega)$ and $\Pi_h\mathbf{v}\cdot\mathbf{n} = 0$, $\nabla \cdot \Pi_h\mathbf{v} \in P_{k-1}(T)$ on each $T \in \mathcal{T}_h$ and satisfies:

$$(\Pi_h\mathbf{v},\ \mathbf{w})_T = (\mathbf{v},\ \mathbf{w})_T \quad \forall \mathbf{w} \in [P_{k-2}(T)]^d, \quad k \ge 2 \tag{13}$$

$$\langle \Pi_h\mathbf{v}\cdot\mathbf{n},\ q\rangle_{\partial T} = \langle\{\mathbf{v}\}\cdot\mathbf{n},\ q\rangle_{\partial T} \quad \forall q \in P_{k-1}(e), e \subset \partial T. \tag{14}$$

Remark 1 The similar velocity reconstruction can also be employed for $\mathbf{v} \in H_0^1(\Omega)$ and such reconstruction just follows the definition of Raviart–Thomas interpolation operator [3].

Lemma 2 *For $\mathbf{v} \in \mathbf{V}_h + \boldsymbol{H}_0^1(\Omega)$, there exists a projection $\Pi_h\mathbf{v} \in \boldsymbol{RT}_{k-1}(T)$ such that $\Pi_h\mathbf{v} \in H_0(div, \Omega) := \{\mathbf{v} \in H(div, \Omega), \mathbf{v}\cdot\mathbf{n}|_{\partial\Omega} = 0\}$, $\nabla \cdot \Pi_h\mathbf{v} \in P_{k-1}(T)$ on each $T \in \mathcal{T}_h$ and satisfies:*

$$\|\Pi_h\mathbf{v} - \mathbf{v}\| \le Ch\|\mathbf{v}\|_{1,h}, \tag{15}$$

Lemma 3 *Let $\boldsymbol{\phi} \in \boldsymbol{H}_0^1(\Omega)$, then the following equations hold true on $T \in \mathcal{T}_h$*

$$\nabla_w \phi = \mathbb{Q}_h \nabla \boldsymbol{\phi}, \tag{16}$$

$$\nabla_w \cdot \phi = Q_h \nabla \cdot \boldsymbol{\phi}, \tag{17}$$

$$\nabla \cdot \Pi_h \phi = \nabla_w \cdot \boldsymbol{\phi}, \tag{18}$$

$$\nabla_w \cdot \Pi_h \phi = \nabla_w \cdot \boldsymbol{\phi}. \tag{19}$$

Proof Using (8) and integration by parts, we have that for any $\tau \in [\mathbf{RT}_k(T)]^d$

$$\begin{aligned}(\nabla_w \boldsymbol{\phi}, \tau)_T &= -(\boldsymbol{\phi}, \nabla\cdot\tau)_T + \langle\{\boldsymbol{\phi}\}, \tau\cdot\mathbf{n}\rangle_{\partial T}\\ &= -(\boldsymbol{\phi}, \nabla\cdot\tau)_T + \langle\boldsymbol{\phi}, \tau\cdot\mathbf{n}\rangle_{\partial T}\\ &= (\mathbb{Q}_h\nabla\phi, \tau)_T,\end{aligned}$$

which implies the desired identity (16). Similarly, we can prove (17). It follows from (13) and (14) that for any $q \in P_{k-1}(T)$,

$$\begin{aligned}(\nabla \cdot \Pi_h \boldsymbol{\phi}, q)_T &= -(\Pi_h \boldsymbol{\phi}, \nabla q)_T + \langle \Pi_h \boldsymbol{\phi} \cdot \mathbf{n}, q \rangle_{\partial T} \\ &= -(\boldsymbol{\phi}, \nabla q)_T + \langle \{\boldsymbol{\phi}\} \cdot \mathbf{n}, q \rangle_{\partial T} \\ &= (\nabla_w \cdot \boldsymbol{\phi}, q)_T,\end{aligned}$$

which proves (18). Similarly, for any $q \in P_{k-1}(T)$

$$\begin{aligned}(\nabla_w \cdot \Pi_h \boldsymbol{\phi}, q)_T &= -(\Pi_h \boldsymbol{\phi}, \nabla q)_T + \langle \{\Pi_h \boldsymbol{\phi}\} \cdot \mathbf{n}, q \rangle_{\partial T} \\ &= -(\boldsymbol{\phi}, \nabla q)_T + \langle \Pi_h \boldsymbol{\phi} \cdot \mathbf{n}, q \rangle_{\partial T} \\ &= -(\boldsymbol{\phi}, \nabla q)_T + \langle \{\boldsymbol{\phi}\} \cdot \mathbf{n}, q \rangle_{\partial T} \\ &= (\nabla_w \cdot \boldsymbol{\phi}, q)_T.\end{aligned}$$

We have proved the lemma. □

For any function $\varphi \in H^1(T)$, the following trace inequality holds true (see [31] for details):

$$\|\varphi\|_e^2 \le C\left(h_T^{-1}\|\varphi\|_T^2 + h_T\|\nabla\varphi\|_T^2\right). \tag{20}$$

3 Numerical Schemes

For the sake of notation simplification, the following notations will also be employed,

$$(u, v)_{\mathcal{T}_h} := \sum_{T \in \mathcal{T}_h} (u, v)_T, \ \langle u, v \rangle_{\partial \mathcal{T}_h} := \sum_{T \in \mathcal{T}_h} \langle u, v \rangle_{\partial T}, \ \langle u, v \rangle_{\mathcal{E}_h} := \sum_{e \in \mathcal{E}_h} \langle u, v \rangle_e.$$

For functions $\mathbf{w}, \mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$, define

$$A_h(\mathbf{w}, \mathbf{v}) := (\nabla_w \mathbf{w}, \nabla_w \mathbf{v})_{\mathcal{T}_h}, \ B_h(\mathbf{v}, q) = -(\nabla_w \cdot \mathbf{v}, q).$$

Besides, we shall also employ the notations in the interior penalty discontinuous Galerkin finite element methods. With each edge/face $e \in \mathcal{E}_h^0$, we associate a unit normal vector $\mathbf{n}_e$. If e is on the boundary $\partial\Omega$, then $\mathbf{n}_e$ is taken to be the unit ourward vector normal to $\partial\Omega$. Define the following bilinear terms for $\mathbf{w}, \mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$,

$$\begin{aligned}\mathcal{A}_h(\mathbf{w}, \mathbf{v}) = (\nabla \mathbf{w}, \nabla \mathbf{v})_{\mathcal{T}_h} &- \langle \{\nabla \mathbf{w}\}\mathbf{n}_e, [\![\mathbf{v}]\!] \rangle_{\mathcal{E}_h} \\ &- \langle \{\nabla \mathbf{v}\}\mathbf{n}_e, [\![\mathbf{w}]\!] \rangle_{\mathcal{E}_h} + \frac{\eta}{h_e} \langle [\![\mathbf{w}]\!], [\![\mathbf{v}]\!] \rangle_{\mathcal{E}_h},\end{aligned}$$

where η is a penalty parameter which is strictly positive and big enough to guarantee stability. Also define

$$\mathcal{B}_h(\mathbf{v}, q) = -(\nabla \cdot \mathbf{v}, q)_{\mathcal{T}_h} + \langle \{q\}, [\![\mathbf{v} \cdot \mathbf{n}_e]\!] \rangle_{\mathcal{E}_h}.$$

In the following, we shall first cite the classical discontinuous Galerkin schemes as Algorithms 1 and 3, and then propose the enhanced pressure robust discontinuous Galerkin schemes in Algorithms 2 and 4.

Algorithm 1 (Conforming Discontinuous Galerkin (CDG) Scheme [34]) A finite element method for (1)–(3) is seeking $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in W_h$ such that for all $\mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$,

$$\begin{aligned}\nu A_h(\mathbf{u}_h, \mathbf{v}) + B_h(\mathbf{v}, p_h) &= (\mathbf{f}, \mathbf{v}),\\ B_h(\mathbf{u}_h, q) &= 0.\end{aligned}$$

Algorithm 2 (Enhanced CDG Scheme) A new finite element method for (1)–(3) is seeking $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in W_h$ such that for all $\mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$,

$$\nu A_h(\mathbf{u}_h, \mathbf{v}) + B_h(\mathbf{v}, p_h) = (\mathbf{f}, \Pi_h \mathbf{v}) \tag{21}$$

$$B_h(\mathbf{u}_h, q) = 0. \tag{22}$$

Algorithm 3 (Interior Penalty Discontinuous Galerkin (IPDG) Scheme [28]) A finite element method for (1)–(3) is seeking $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in W_h$ such that for all $\mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$

$$\begin{aligned}\nu \mathcal{A}_h(\mathbf{u}_h, \mathbf{v}) + \mathcal{B}_h(\mathbf{v}, p_h) &= (\mathbf{f}, \mathbf{v}),\\ \mathcal{B}_h(\mathbf{u}_h, q) &= 0.\end{aligned}$$

Algorithm 4 (Enhanced IPDG Scheme) A new finite element method for (1)–(3) is seeking $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in W_h$ such that for all $\mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$

$$\nu \mathcal{A}_h(\mathbf{u}_h, \mathbf{v}) + \mathcal{B}_h(\mathbf{v}, p_h) = (\mathbf{f}, \Pi_h \mathbf{v}), \tag{23}$$

$$\mathcal{B}_h(\mathbf{u}_h, q) = 0. \tag{24}$$

Here, one can prove the equivalence of $B_h(\mathbf{v}, q)$ and $\mathcal{B}_h(\mathbf{v}, q)$ for $\mathbf{v} \in \mathbf{V}_h$ and $q \in W_h$ as follows,

$$\begin{aligned}B_h(\mathbf{v}, q) &= -\sum_{T\in\mathcal{T}_h} \int_T \nabla_w \cdot \mathbf{v} q dT = \sum_{T\in\mathcal{T}_h} \left(-\int_{\partial T} \{\mathbf{v}\} \cdot \mathbf{n} q ds + \int_T \mathbf{v} \cdot \nabla q dT \right)\\ &= \sum_{T\in\mathcal{T}_h} \left(\int_{\partial T} (\mathbf{v} - \{\mathbf{v}\}) \cdot \mathbf{n} q ds - \int_T \nabla \cdot \mathbf{v} q dT \right)\\ &= \mathcal{B}_h(\mathbf{v}, q).\end{aligned} \tag{25}$$

Then together with the modifications in the source terms, we have the following remarks.

Remark 2 Algorithm 1 only differs Algorithm 3 in the terms $A_h(\mathbf{w}, \mathbf{v})$ and $\mathcal{A}_h(\mathbf{w}, \mathbf{v})$. However, Algorithms 1 is parameter/stabilizer free but Algorithm 3 requires the penalty parameter η to be large enough for the sake of stability.

Remark 3 The stiffness matrix produced by Algorithm 2 is the same as that in Algorithm 1; the stiffness matrix of Algorithm 4 is the same as that in Algorithm 3. The only modification in Algorithm 2/Algorithm 4 is the external source assembling. Through such minor modification, we can enhance the classical schemes with the desire pressure-robustness.

Lemma 4 *There exists a positive constant β independent of h such that for all $\rho \in W_h$*

$$\sup_{\mathbf{v}\in\mathbf{V}_h} \frac{B_h(\mathbf{v}, \rho)}{|||\mathbf{v}|||} \geq \beta \|\rho\|. \tag{26}$$

Proof For any given $\rho \in W_h \subset L_0^2(\Omega)$, it is known [10] that there exists a function $\tilde{\mathbf{v}} \in \boldsymbol{H}_0^1(\Omega)$ such that

$$\frac{(\nabla \cdot \tilde{\mathbf{v}}, \rho)}{\|\tilde{\mathbf{v}}\|_1} \geq C\|\rho\|, \tag{27}$$

where $C > 0$ is a constant independent of h. Let $\mathbf{v} = \Pi_h\tilde{\mathbf{v}} \in \mathbf{V}_h$. Here Π_h is the Raviart Thomas (RT) interpolation operator to the space $\mathbf{RT}_{k-1}(T)$, which belongs to the finite element space $\mathbf{V}_h$. It follows from the equivalence of norms (12), trace inequality (20) and $\tilde{\mathbf{v}} \in \boldsymbol{H}_0^1(\Omega)$,

$$\begin{aligned}
|||\mathbf{v}|||^2 &\le C\|\mathbf{v}\|_{1,h}^2 = C(\sum_{T\in\mathcal{T}_h}\|\nabla\mathbf{v}\|_T^2 + \sum_{e\in\mathcal{E}_h^0} h_e^{-1}\|[\![\mathbf{v}]\!]\|_e^2)\\
&\le C\sum_{T\in\mathcal{T}_h}\|\nabla\Pi_h\tilde{\mathbf{v}}\|_T^2 + \sum_{e\in\mathcal{E}_h^0} h_e^{-1}\|[\![\Pi_h\tilde{\mathbf{v}} - \tilde{\mathbf{v}}]\!]\|_e^2\\
&\le C\sum_{T\in\mathcal{T}_h}\|\nabla\Pi_h\tilde{\mathbf{v}}\|_T^2 + h^{-2}\|\Pi_h\tilde{\mathbf{v}} - \tilde{\mathbf{v}}\|_T^2 + \|\nabla\Pi_h\tilde{\mathbf{v}} - \nabla\tilde{\mathbf{v}}\|_T^2\\
&\le C\|\tilde{\mathbf{v}}\|_1^2.
\end{aligned} \tag{28}$$

The last step used (15) and the property of RT interpolation. It follows from (18) and (19) that

$$(\nabla_w\cdot\mathbf{v},\ \rho) = (\nabla_w\cdot\Pi_h\tilde{\mathbf{v}},\ \rho) = (\nabla\cdot\tilde{\mathbf{v}},\ \rho).$$

Using the above Eqs. (28) and (27), we have

$$\frac{|B_h(\mathbf{v},\rho)|}{|||\mathbf{v}|||} = \frac{|(\nabla_w\cdot\mathbf{v},\rho)|}{|||\mathbf{v}|||} \ge \frac{|(\nabla\cdot\tilde{\mathbf{v}},\rho)|}{C\|\tilde{\mathbf{v}}\|_1} \ge \beta\|\rho\|,$$

for a positive constant β. This completes the proof. □

Remark 4 The similar proof of inf-sup condition for IPDG can be found in Proposition 10 in [14] and Proposition 4.4 in [6].

Remark 5 The well-posedness of the scheme can be found in [34] for Algorithm 1. The proof of existence, uniqueness, and stability of Algorithm 3 can be found in [28]. Since Algorithms 2 and 4 share the same stiffness matrices as Algorithms 1 and 3, the new proposed Algorithms 2 and 4 directly inherit the well-posedness results. Here, we shall cite the following inequalities needed by later sections,

$$\forall\mathbf{v},\mathbf{w}\in\mathbf{V}_h + \boldsymbol{H}_0^1,\ \mathcal{A}_h(\mathbf{w},\mathbf{v}) \le C_1\|\mathbf{w}\|_{1,h}\|\mathbf{v}\|_{1,h}, \tag{29}$$

$$\forall\mathbf{v}\in\mathbf{V}_h, \mathcal{A}_h(\mathbf{v},\mathbf{v}) \ge C_2\|\mathbf{v}\|_{1,h}^2 \tag{30}$$

$$\forall\rho\in W_h,\ \sup_{\mathbf{v}\in\mathbf{V}_h}\frac{\mathcal{B}_h(\mathbf{v},\rho)}{\|\mathbf{v}\|_{1,h}} \ge \beta\|\rho\|, \tag{31}$$

where β, C_1, and C_2 are a positive constants independent of h.

4 Error Equations

In this section, we will derive the equations that the errors satisfy for the enhanced CDG scheme (Algorithm 2) and enhanced IPDG scheme (Algorithm 4). Let $\mathbf{e}_h = \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h$, $\boldsymbol{\epsilon}_h = \mathbf{u} - \mathbf{u}_h$ and $\varepsilon_h = Q_h p - p_h$.

4.1 Error Equations for CDG Scheme in Algorithm 2

Lemma 5 *The following error equations hold true for any $(\mathbf{v}, q) \in \mathbf{V}_h \times W_h$,*

$$\nu A_h(\mathbf{e}_h, \mathbf{v}) + B_h(\mathbf{v}, \varepsilon_h) = \nu\ell_1(\mathbf{u}, \mathbf{v}) - \nu\ell_2(\mathbf{u}, \mathbf{v}) + \nu\ell_3(\mathbf{u}, \mathbf{v}), \tag{32}$$

$$B_h(\mathbf{e}_h,\ q) = \ell_4(\mathbf{u}, q), \tag{33}$$

where

$$\ell_1(\mathbf{u},\ \mathbf{v}) = \langle(\nabla\mathbf{u} - \mathbb{Q}_h(\nabla\mathbf{u}))\cdot\mathbf{n},\ \mathbf{v} - \{\mathbf{v}\}\rangle_{\partial\mathcal{T}_h}, \tag{34}$$

$$\ell_2(\mathbf{u}, \mathbf{v}) = (\nabla_w(\mathbf{u} - \mathbf{Q}_h\mathbf{u}), \nabla_w\mathbf{v})_{\mathcal{T}_h}, \tag{35}$$

$$\ell_3(\mathbf{u}, \mathbf{v}) = (\Delta\mathbf{u}, \Pi_h\mathbf{v} - \mathbf{v})_{\mathcal{T}_h}, \tag{36}$$

$$\ell_4(\mathbf{u}, q) = \langle\{\mathbf{u} - \mathbf{Q}_h\mathbf{u}\}\cdot\mathbf{n}, q\rangle_{\partial\mathcal{T}_h}. \tag{37}$$

Proof For a $\mathbf{v} \in \mathbf{V}_h$, we test (1) by $\Pi_h\mathbf{v}$ to obtain

$$-\nu(\Delta\mathbf{u},\ \Pi_h\mathbf{v}) + (\nabla p,\ \Pi_h\mathbf{v}) = (\mathbf{f},\ \Pi_h\mathbf{v}), \tag{38}$$

which implies

$$-\nu(\Delta\mathbf{u},\ \mathbf{v}) + (\nabla p,\ \Pi_h\mathbf{v}) = (\mathbf{f},\ \Pi_h\mathbf{v}) + \nu\ell_3(\mathbf{u}, \mathbf{v}), \tag{39}$$

Integration by parts and the fact $\langle\nabla\mathbf{u}\cdot\mathbf{n},\ \{\mathbf{v}\}\rangle_{\partial\mathcal{T}_h} = 0$ give

$$-(\Delta\mathbf{u},\ \mathbf{v}) = (\nabla\mathbf{u}, \nabla\mathbf{v})_{\mathcal{T}_h} - \langle\nabla\mathbf{u}\cdot\mathbf{n}, \mathbf{v} - \{\mathbf{v}\}\rangle_{\partial\mathcal{T}_h}. \tag{40}$$

It follows from integration by parts, (8) and (16),

$$\begin{aligned}(\nabla\mathbf{u}, \nabla\mathbf{v})_{\mathcal{T}_h} &= (\mathbb{Q}_h\nabla\mathbf{u}, \nabla\mathbf{v})_{\mathcal{T}_h}\\ &= -(\mathbf{v}, \nabla\cdot(\mathbb{Q}_h\nabla\mathbf{u}))_{\mathcal{T}_h} + \langle\mathbf{v}, \mathbb{Q}_h\nabla\mathbf{u}\cdot\mathbf{n}\rangle_{\partial\mathcal{T}_h}\\ &= (\mathbb{Q}_h\nabla\mathbf{u}, \nabla_w\mathbf{v})_{\mathcal{T}_h} + \langle\mathbf{v} - \{\mathbf{v}\}, \mathbb{Q}_h\nabla\mathbf{u}\cdot\mathbf{n}\rangle_{\partial\mathcal{T}_h}\\ &= (\nabla_w\mathbf{u}, \nabla_w\mathbf{v})_{\mathcal{T}_h} + \langle\mathbf{v} - \{\mathbf{v}\}, \mathbb{Q}_h\nabla\mathbf{u}\cdot\mathbf{n}\rangle_{\partial\mathcal{T}_h}.\end{aligned} \tag{41}$$

Combining (40) and (41) gives

$$-(\Delta\mathbf{u},\ \mathbf{v}) = (\nabla_w\mathbf{u}, \nabla_w\mathbf{v})_{\mathcal{T}_h} - \ell_1(\mathbf{u}, \mathbf{v}). \tag{42}$$

Using integration by parts, $\Pi_h\mathbf{v} \in H_0(\mathrm{div}, \Omega)$ and (18), we have

$$\begin{aligned}(\nabla p,\ \Pi_h\mathbf{v}) &= -(p, \nabla\cdot\Pi_h\mathbf{v})_{\mathcal{T}_h} + \langle p, \Pi_h\mathbf{v}\cdot\mathbf{n}\rangle_{\partial\mathcal{T}_h}\\ &= -(Q_hp, \nabla\cdot\Pi_h\mathbf{v})_{\mathcal{T}_h}\\ &= -(Q_hp, \nabla_w\cdot\mathbf{v})_{\mathcal{T}_h}.\end{aligned} \tag{43}$$

Substituting (42) and (43) into (39) gives

$$\nu(\nabla_w\mathbf{u}, \nabla_w\mathbf{v})_{\mathcal{T}_h} - (Q_hp, \nabla_w\cdot\mathbf{v})_{\mathcal{T}_h} = (\mathbf{f}, \Pi_h\mathbf{v}) + \nu\ell_1(\mathbf{u}, \mathbf{v}) + \nu\ell_3(\mathbf{u}, \mathbf{v}). \tag{44}$$

The difference of (44) and (21) implies

$$\nu(\nabla_w\boldsymbol{\epsilon}_h, \nabla_w\mathbf{v})_{\mathcal{T}_h} - (\varepsilon_h, \nabla_w\cdot\mathbf{v})_{\mathcal{T}_h} = \nu\ell_1(\mathbf{u}, \mathbf{v}) + \nu\ell_3(\mathbf{u}, \mathbf{v}) \quad \forall\mathbf{v} \in \mathbf{V}_h. \tag{45}$$

Adding and subtracting $(\nabla_w\mathbf{Q}_h\mathbf{u} - \nabla_w\mathbf{u}, \nabla_w\mathbf{v})_{\mathcal{T}_h}$ in (45), we have

$$\nu(\nabla_w\mathbf{e}_h, \nabla_w\mathbf{v})_{\mathcal{T}_h} - (\varepsilon_h, \nabla_w\cdot\mathbf{v})_{\mathcal{T}_h} = \nu\ell_1(\mathbf{u}, \mathbf{v}) - \nu\ell_2(\mathbf{u}, \mathbf{v}) - \nu\ell_3(p, \mathbf{v}), \tag{46}$$

which implies (32).

Testing Eq. (2) by $q \in W_h$ and using (9) give

$$\begin{aligned}(\nabla \cdot \mathbf{u},\ q) &= -(\mathbf{u},\ \nabla q)_{\mathcal{T}_h} + \langle \mathbf{u}\cdot\mathbf{n}, q\rangle_{\partial\mathcal{T}_h}\\ &= -(\mathbf{Q}_h\mathbf{u},\ \nabla q)_{\mathcal{T}_h} + \langle \mathbf{u}\cdot\mathbf{n}, q\rangle_{\partial\mathcal{T}_h}\\ &= (\nabla_w \cdot \mathbf{Q}_h\mathbf{u},\ q)_{\mathcal{T}_h} + \langle\{\mathbf{u} - \mathbf{Q}_h\mathbf{u}\}\cdot\mathbf{n},\ q\rangle_{\partial\mathcal{T}_h}\\ &= (\nabla_w \cdot \mathbf{Q}_h\mathbf{u},\ q)_{\mathcal{T}_h} + \ell_4(\mathbf{u}, q).\end{aligned}$$

which implies

$$(\nabla_w \cdot \mathbf{Q}_h\mathbf{u},\ q)_{\mathcal{T}_h} = -\ell_4(\mathbf{u}, q). \tag{47}$$

The difference of (47) and (22) implies (33). We have proved the lemma. □

The following lemma will be used in the L^2 error estimate.

Lemma 6 *For any* $(\mathbf{v}, q) \in \mathbf{V}_h \times W_h$*, then we have,*

$$\nu A_h(\boldsymbol{\epsilon}_h, \Pi_h\mathbf{v}) + B_h(\mathbf{v}, \varepsilon_h) = \nu\ell_1(\mathbf{u}, \Pi_h\mathbf{v}) \quad \forall \mathbf{v} \in V_h. \tag{48}$$

Proof Letting $\mathbf{v} = \Pi_h\mathbf{v}$ in (42), we have

$$-(\Delta\mathbf{u},\ \Pi_h\mathbf{v}) = (\nabla_w\mathbf{u}, \nabla_w\Pi_h\mathbf{v})_{\mathcal{T}_h} - \ell_1(\mathbf{u}, \Pi_h\mathbf{v}). \tag{49}$$

Combining (49) and (43) with (38) gives

$$\nu(\nabla_w\mathbf{u}, \nabla_w\Pi_h\mathbf{v})_{\mathcal{T}_h} - (Q_h p, \nabla_w \cdot \mathbf{v})_{\mathcal{T}_h} = (\mathbf{f}, \Pi_h\mathbf{v}) + \nu\ell_1(\mathbf{u}, \Pi_h\mathbf{v}). \tag{50}$$

Letting $\mathbf{v} = \Pi_h\mathbf{v}$ in (21) gives

$$\nu(\nabla_w\mathbf{u}_h,\ \nabla_w\Pi_h\mathbf{v}) - (\nabla_w \cdot \Pi_h\mathbf{v},\ p_h) = (f,\ \Pi_h\mathbf{v}). \tag{51}$$

Using (19), the Eq. (51) becomes

$$\nu(\nabla_w\mathbf{u}_h,\ \nabla_w\Pi_h\mathbf{v}) - (\nabla_w \cdot \mathbf{v},\ p_h) = (f,\ \Pi_h\mathbf{v}), \tag{52}$$

The difference of (50) and (52) implies

$$\nu(\nabla_w\boldsymbol{\epsilon}_h, \nabla_w\Pi_h\mathbf{v})_{\mathcal{T}_h} - (\varepsilon_h, \nabla_w \cdot \mathbf{v})_{\mathcal{T}_h} = \nu\ell_1(\mathbf{u}, \Pi_h\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{V}_h. \tag{53}$$

By the definitions of A_h, B_h, we have proved the lemma. □

4.2 Error Equations for IPDG Scheme in Algorithm 4

Lemma 7 *The following error equations hold true for IPDG scheme: any* $(\mathbf{v}, q) \in \mathbf{V}_h \times W_h$*,*

$$\nu\mathcal{A}_h(\boldsymbol{\epsilon}_h, \mathbf{v}) + \mathcal{B}_h(\mathbf{v}, \varepsilon_h) = \nu\ell_3(\mathbf{u}, \mathbf{v}), \tag{54}$$

$$\mathcal{B}_h(\mathbf{e}_h,\ q) = \ell_4(\mathbf{u}, q). \tag{55}$$

Proof Since the equivalence of $B_h(\mathbf{v}, q)$ and $\mathcal{B}_h(\mathbf{v}, q)$ as (25) for $\mathbf{v} \in \mathbf{V}_h, q \in W_h$, (55) follows from (33). For a $\mathbf{v} \in \mathbf{V}_h$, we test (1) by $\Pi_h\mathbf{v}$ to obtain (39). The definition of $\mathcal{A}_h(\cdot, \cdot)$ gives

$$-(\Delta\mathbf{u}, \mathbf{v}) = \mathcal{A}_h(\mathbf{u}, \mathbf{v}) = \mathcal{A}_h(\mathbf{u}, \mathbf{v}), \tag{56}$$

Substituting the above to (39), (43), and using the equivalence of $B_h(\cdot, \cdot)$ and $\mathcal{B}_h(\cdot, \cdot)$ as (25), we have

$$\nu\mathcal{A}_h(\mathbf{u}, \mathbf{v}) + \mathcal{B}_h(\mathbf{v}, Q_h p) = (\mathbf{f},\ \Pi_h\mathbf{v}) + \nu\ell_3(\mathbf{u}, \mathbf{v}).$$

The difference of the above equation and (23) gives (54). □

Lemma 8 *For any* $(\mathbf{v}, q) \in \mathbf{V}_h \times W_h$*, then we have,*

$$\nu \mathcal{A}_h(\boldsymbol{\epsilon}_h, \Pi_h \mathbf{v})_{\mathcal{T}_h} + \mathcal{B}_h(\mathbf{v}, \varepsilon_h)_{\mathcal{T}_h} = 0 \quad \forall \mathbf{v} \in V_h. \tag{57}$$

Proof Letting $\mathbf{v} = \Pi_h \mathbf{v}$ in (56), we have

$$-(\Delta \mathbf{u}, \Pi_h \mathbf{v}) = \mathcal{A}_h(\mathbf{u}, \Pi_h \mathbf{v}).$$

Combining $(\nabla p, \Pi_h \mathbf{v}) = \mathcal{B}_h(\mathbf{v}, \mathcal{Q}_h p)$ with (38), we obtain

$$\nu \mathcal{A}_h(\mathbf{u}, \Pi_h \mathbf{v}) + \mathcal{B}_h(\mathbf{v}, \mathcal{Q}_h p) = (\mathbf{f},\ \Pi_h \mathbf{v}),$$

and thus the conclusion follows by taking the difference of the above equations and (23). □

5 Main Results

5.1 Error Estimates in Energy Norm

Optimal order error estimates will be derived in this section for the velocity $\mathbf{u}_h$ and pressure p_h in $|||\cdot|||$ norm and in the L^2 norm, respectively. The following equations hold true for $\{\mathbf{v}\}$ defined in (7),

$$\|\mathbf{v} - \{\mathbf{v}\}\|_e = \|[\![\mathbf{v}]\!]\|_e \ \text{ if } e \subset \partial\Omega, \quad \|\mathbf{v} - \{\mathbf{v}\}\|_e = \frac{1}{2}\|[\![\mathbf{v}]\!]\|_e \ \text{ if } e \in \mathcal{E}_h^0. \tag{58}$$

Lemma 9 *Let* $(\mathbf{w}, \rho) \in \boldsymbol{H}^{k+1}(\Omega) \times H^k(\Omega)$ *and* $(\mathbf{v}, q) \in \mathbf{V}_h \times W_h$*. Then, the following estimates hold true*

$$|\ell_1(\mathbf{w},\ \mathbf{v})| \le Ch^k |\mathbf{w}|_{k+1} |||\mathbf{v}|||, \tag{59}$$

$$|\ell_2(\mathbf{w},\ \mathbf{v})| \le Ch^k |\mathbf{w}|_{k+1} |||\mathbf{v}|||, \tag{60}$$

$$|\ell_3(\mathbf{w},\ \mathbf{v})| \le Ch^k |\mathbf{w}|_{k+1} |||\mathbf{v}|||. \tag{61}$$

$$|\ell_4(\mathbf{w}, q) \le Ch^k |\mathbf{w}|_{k+1} \|q\|. \tag{62}$$

Proof Using the Cauchy–Schwarz inequality, the trace inequality (20), (58) and (12), we have

$$\begin{aligned}
|\ell_1(\mathbf{w},\ \mathbf{v})| &= |\sum_{T\in\mathcal{T}_h} \langle \mathbf{v} - \{\mathbf{v}\},\ \nabla \mathbf{w}\cdot\mathbf{n} - \mathbb{Q}_h(\nabla \mathbf{w})\cdot\mathbf{n}\rangle_{\partial T}| \\
&\le C \sum_{T\in\mathcal{T}_h} \|\nabla \mathbf{w} - \mathbb{Q}_h \nabla \mathbf{w}\|_{\partial T} \|\mathbf{v} - \{\mathbf{v}\}\|_{\partial T} \\
&\le C \left(\sum_{T\in\mathcal{T}_h} h_T \|(\nabla \mathbf{w} - \mathbb{Q}_h \nabla \mathbf{w})\|_{\partial T}^2\right)^{\frac{1}{2}} \left(\sum_{e\in\mathcal{E}_h} h_e^{-1} \|[\![\mathbf{v}]\!]\|_e^2\right)^{\frac{1}{2}} \\
&\le Ch^k |\mathbf{w}|_{k+1} |||\mathbf{v}|||.
\end{aligned}$$

It follows from (8), integration by parts, (20) and (58) that for any $\mathbf{q} \in [\mathbf{RT}_{k-1}(T)]^d$,

$$\begin{aligned}
|(\nabla_w(\mathbf{w} - \mathbf{Q}_h \mathbf{w}), \mathbf{q})_T| &= |-(\mathbf{w} - \mathbf{Q}_h \mathbf{w}, \nabla\cdot\mathbf{q})_T + \langle \mathbf{w} - \{\mathbf{Q}_h \mathbf{w}\}, \mathbf{q}\cdot\mathbf{n}\rangle_{\partial T}| \\
&= |(\nabla(\mathbf{w} - \mathbf{Q}_h \mathbf{w}), \mathbf{q})_T + \langle \mathbf{Q}_h \mathbf{w} - \{\mathbf{Q}_h \mathbf{w}\}, \mathbf{q}\cdot\mathbf{n}\rangle_{\partial T}|
\end{aligned}$$

$$
\begin{aligned}
&\le \|\nabla(\mathbf{w}-\mathbf{Q}_h\mathbf{w})\|_T\|\mathbf{q}\|_T + Ch^{-1/2}\|[\![\mathbf{Q}_h\mathbf{w}]\!]\|_{\partial T}\|\mathbf{q}\|_T\\
&\le \|\nabla(\mathbf{w}-\mathbf{Q}_h\mathbf{w})\|_T\|\mathbf{q}\|_T + Ch^{-1/2}\|[\![\mathbf{w}-\mathbf{Q}_h\mathbf{w}]\!]\|_{\partial T}\|\mathbf{q}\|_T\\
&\le Ch^k|\mathbf{w}|_{k+1,T}\|\mathbf{q}\|_T. \qquad (63)
\end{aligned}
$$

Letting $\mathbf{q} = \nabla_w\mathbf{v}$ in (63) and taking summation over T, we have

$$
\begin{aligned}
|\ell_2(\mathbf{w},\ \mathbf{v})| &= |(\nabla_w(\mathbf{w}-\mathbf{Q}_h\mathbf{w}),\nabla_w\mathbf{v})_{\mathcal{T}_h}|\\
&\le Ch^k|\mathbf{w}|_{k+1}|\!|\!|\mathbf{v}|\!|\!|.
\end{aligned}
$$

If $k = 1$, it follows from Cauchy–Schwarz inequality and (15) that

$$
\begin{aligned}
|\ell_3(\mathbf{w},\ \mathbf{v})| &= |\sum_{T\in\mathcal{T}_h}(\Delta\mathbf{w},\Pi_h\mathbf{v}-\mathbf{v})_T| \le \sum_{T\in\mathcal{T}_h}\|\Delta\mathbf{w}\|_T\|\Pi_h\mathbf{v}-\mathbf{v}\|_T\\
&\le Ch|\mathbf{w}|_2\|\mathbf{v}\|_{1,h}.
\end{aligned}
$$

If $k \ge 2$, denote by $Q_{k-2,T}$ the L^2 projection onto $P_{k-2}(T)$. It follows from the definition of $Q_{k-2,T}$, (13) and (15) that

$$
\begin{aligned}
|\ell_3(\mathbf{w},\ \mathbf{v})| &= |\sum_{T\in\mathcal{T}_h}(\Delta\mathbf{w},\Pi_h\mathbf{v}-\mathbf{v})_T|\\
&= |\sum_{T\in\mathcal{T}_h}(\Delta\mathbf{w}-Q_{k-2,T}(\Delta\mathbf{w}),\Pi_h\mathbf{v}-\mathbf{v})_T|\\
&\le Ch^k|\mathbf{w}|_{k+1}\|\mathbf{v}\|_{1,h}.
\end{aligned}
$$

Similarly we have

$$
\begin{aligned}
|\ell_4(\mathbf{w},q)| &= |\sum_{T\in\mathcal{T}_h}\langle\{\mathbf{w}-\mathbf{Q}_h\mathbf{w}\}\cdot\mathbf{n},q\rangle_{\partial T}|\\
&\le C\sum_{T\in\mathcal{T}_h}\|\mathbf{w}-\mathbf{Q}_h\mathbf{w}\|_{\partial T}\|q\|_{\partial T}\\
&\le C\left(\sum_{T\in\mathcal{T}_h}h_T^{-1}\|\mathbf{w}-\mathbf{Q}_h\mathbf{w}\|_{\partial T}^2\right)^{\frac{1}{2}}\left(\sum_{T\in\mathcal{T}_h}h_T\|q\|_e^2\right)^{\frac{1}{2}}\\
&\le Ch^k|\mathbf{w}|_{k+1}\|q\|.
\end{aligned}
$$

We have proved the lemma. □

Theorem 1 *Let $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times W_h$ be the solutions from Algorithms 2 or 4. Assume η is chosen big enough when Algorithm 4 is employed. Then, the following error estimates hold true*

$$
\|\mathbf{u}-\mathbf{u}_h\|_{1,h} \le Ch^k|\mathbf{u}|_{k+1}, \qquad (64)
$$

$$
\|Q_hp-p_h\| \le C\nu h^k|\mathbf{u}|_{k+1}. \qquad (65)
$$

Proof 1). We first prove the results for Algorithm 2. It follows from (32) that for any $\mathbf{v}\in\mathbf{V}_h$

$$
\begin{aligned}
|(\varepsilon_h,\ \nabla_w\cdot\mathbf{v})_{\mathcal{T}_h}| &= |\nu(\nabla_w\mathbf{e}_h,\ \nabla_w\mathbf{v})_{\mathcal{T}_h} - \ell_1(\mathbf{u},\mathbf{v}) + \nu\ell_2(\mathbf{u},\mathbf{v}) - \nu\ell_3(\mathbf{u},\mathbf{v})|\\
&\le C\nu(|\!|\!|\mathbf{e}_h|\!|\!| + h^k|\mathbf{u}|_{k+1})|\!|\!|\mathbf{v}|\!|\!|. \qquad (66)
\end{aligned}
$$

Then the estimate (66) and the inf-sup condition (26) yield

$$
\|\varepsilon_h\| \le C\nu(|\!|\!|\mathbf{e}_h|\!|\!| + h^k|\mathbf{u}|_{k+1}). \qquad (67)
$$

By letting $\mathbf{v} = \mathbf{e}_h$ in (32) and $q = \varepsilon_h$ in (33) and subtracting the two resulting equations, we have

$$\nu \|\!|\mathbf{e}_h|\!\|^2 = |\nu\ell_1(\mathbf{u}, \mathbf{e}_h) - \nu\ell_2(\mathbf{u}, \mathbf{e}_h) + \nu\ell_3(\mathbf{u}, \mathbf{e}_h) - \ell_4(\mathbf{u}, \varepsilon_h)|. \tag{68}$$

It then follows from (59)–(62) and (67) that

$$\begin{aligned}\nu \|\!|\mathbf{e}_h|\!\|^2 &\le C\nu h^k |\mathbf{u}|_{k+1} \|\!|\mathbf{e}_h|\!\| + Ch^k|\mathbf{u}|_{k+1}\|\varepsilon_h\| \\ &\le C\nu h^k |\mathbf{u}|_{k+1} \|\!|\mathbf{e}_h|\!\| + C\nu h^k|\mathbf{u}|_{k+1}(\|\!|\mathbf{e}_h|\!\| + h^k|\mathbf{u}|_{k+1}) \\ &\le C\nu h^{2k}|\mathbf{u}|_{k+1}^2 + \frac{\nu}{2}\|\!|\mathbf{e}_h|\!\|^2 \end{aligned} \tag{69}$$

which implies

$$\|\!|\mathbf{e}_h|\!\| \le Ch^k|\mathbf{u}|_{k+1}. \tag{70}$$

It follows from (12)

$$\|\mathbf{e}_h\|_{1,h} \le C\|\!|\mathbf{e}_h|\!\| \le Ch^k|\mathbf{u}|_{k+1}. \tag{71}$$

The estimate (71) and the triangle inequality imply (64). The pressure error estimate (65) follows immediately from (67) and (70).

2). Next, we prove the convergence results for Algorithm 4. By (54) and the continuity of $\mathcal{A}_h(\cdot,\cdot)$ in (29), we have

$$\begin{aligned}|\mathcal{B}_h(\mathbf{v}, \varepsilon_h)| &\le |\nu\ell_3(\mathbf{u}, \mathbf{v})| + |\nu\mathcal{A}_h(\boldsymbol{\epsilon}_h, \mathbf{v})| \\ &\le C\nu\left(h^k|\mathbf{u}|_{k+1} + \|\boldsymbol{\epsilon}_h\|_{1,h}\right)\|\mathbf{v}\|_{1,h},\end{aligned}$$

and thus by the inf-sup condition (31), it gives

$$\|\varepsilon_h\| \le C\nu\left(h^k|\mathbf{u}|_{k+1} + \|\boldsymbol{\epsilon}_h\|_{1,h}\right). \tag{72}$$

By (12), coercivity of bilinear form $\mathcal{A}_h(\cdot,\cdot)$, (54), (55), (61), and (62), we have

$$\begin{aligned}\nu\|\!|\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h|\!\| &\le C\nu\|\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h\|_{1,h}^2 \le C\nu\mathcal{A}_h(\mathbf{Q}_h\mathbf{u} - \mathbf{u}_h, \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h) \\ &= \nu\mathcal{A}_h(\mathbf{Q}_h\mathbf{u} - \mathbf{u} + \mathbf{u} - \mathbf{u}_h, \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h) \\ &= \nu\mathcal{A}_h(\mathbf{Q}_h\mathbf{u} - \mathbf{u}, \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h) + \nu\ell_3(\mathbf{u}, \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h) - \ell_4(\mathbf{u}, \varepsilon_h) \\ &\le C\nu h^k|\mathbf{u}|_{k+1}\|\!|\mathbf{e}_h|\!\| + Ch^k|\mathbf{u}|_{k+1}\|\varepsilon_h\| \\ &\le C\nu h^k|\mathbf{u}|_{k+1}\|\!|\mathbf{e}_h|\!\| + C\nu h^k|\mathbf{u}|_{k+1}(\|\!|\mathbf{e}_h|\!\| + h^k|\mathbf{u}|_{k+1}) \\ &\le C\nu h^{2k}|\mathbf{u}|_{k+1}^2 + \frac{\nu}{2}\|\!|\mathbf{e}_h|\!\|^2\end{aligned}$$

and thus

$$\|\!|\mathbf{e}_h|\!\| \le C\nu h^k|\mathbf{u}|_{k+1}.$$

Then the above estimate combines with (72) completes the proof of this lemma.

□

Remark 6 In comparison, we also cite the following convergence results for energy error estimate of Algorithms 1 and 3, which can be found in [28,34]. Let $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times W_h$ be the solutions from Algorithms 1 or 3. Assume η is chosen big enough when Algorithm 3 is employed. We have

$$\|\mathbf{u} - \mathbf{u}_h\|_{1,h} \le Ch^k\left(|\mathbf{u}|_{k+1} + \frac{1}{\nu}\|p\|_k\right), \tag{73}$$

$$\|Q_h p - p_h\| \le Ch^k \left(\nu|\mathbf{u}|_{k+1} + \|p\|_k\right). \tag{74}$$

It is noted that the pressure dependency has been removed in Theorem 1.

5.2 Error Estimates in L^2-Norm

In this section, the L^2-error estimate for the velocity approximation is investigated. Recall that $\mathbf{e}_h = \mathbf{Q}_h\mathbf{u} - \mathbf{u}_h$, $\boldsymbol{\epsilon}_h = \mathbf{u} - \mathbf{u}_h$ and $\varepsilon_h = Q_h p - p_h$. Consider the dual problem of seeking $(\boldsymbol{\psi}, \xi)$ such that

$$-\nu\Delta\boldsymbol{\psi} + \nabla\xi = \Pi_h\boldsymbol{\epsilon}_h \quad \text{in } \Omega, \tag{75}$$

$$\nabla\cdot\boldsymbol{\psi} = 0 \quad \text{in } \Omega, \tag{76}$$

$$\boldsymbol{\psi} = 0 \quad \text{on } \partial\Omega. \tag{77}$$

Assume that the dual problem has the $\boldsymbol{H}^2(\Omega)\times H^1(\Omega)$-regularity in the sense that the solution $(\boldsymbol{\psi}, \xi) \in \boldsymbol{H}^2(\Omega)\times H^1(\Omega)$ satisfies

$$\nu\|\boldsymbol{\psi}\|_2 + \|\xi\|_1 \le C\|\Pi_h\boldsymbol{\epsilon}_h\|. \tag{78}$$

Theorem 2 *Let $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times W_h$ be the solutions of Algorithms 2 or 4. Assume that (78) holds true and η is chosen big enough if Algorithm 4 is employed. Then we have*

$$\|\mathbf{u} - \mathbf{u}_h\| \le Ch^{k+1}|\mathbf{u}|_{k+1}. \tag{79}$$

Proof 1). First we shall prove the L^2-error estimate for CDG scheme in Algorithm 2. Testing (75) by $\Pi_h\boldsymbol{\epsilon}_h$ gives

$$(\Pi_h\boldsymbol{\epsilon}_h, \Pi_h\boldsymbol{\epsilon}_h) = -\nu(\Delta\boldsymbol{\psi},\ \Pi_h\boldsymbol{\epsilon}_h) + (\nabla\xi,\ \Pi_h\boldsymbol{\epsilon}_h). \tag{80}$$

It is obvious

$$-(\Delta\boldsymbol{\psi}, \Pi_h\boldsymbol{\epsilon}_h) = -(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) + (\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h - \Pi_h\boldsymbol{\epsilon}_h). \tag{81}$$

To estimate the first term on the right hand side of (81), we use integration by parts and the fact $\langle\nabla\boldsymbol{\psi}\cdot\mathbf{n}, \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h} = 0$ to obtain

$$\begin{aligned}
-(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) &= (\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \langle\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h - \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\\
&= (\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} + (\nabla\boldsymbol{\psi} - \mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}\\
&\quad -\langle\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h - \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\\
&= -(\nabla\cdot\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} + \langle\mathbb{Q}_h\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h\rangle_{\partial\mathcal{T}_h}\\
&\quad +(\nabla\boldsymbol{\psi} - \mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \langle\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h - \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\\
&= (\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} + \langle\mathbb{Q}_h\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h - \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\\
&\quad +(\nabla\boldsymbol{\psi} - \mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \langle\nabla\boldsymbol{\psi}\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h - \{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\\
&= (\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} + (\nabla\boldsymbol{\psi} - \mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \ell_1(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h),
\end{aligned}$$

which implies

$$\begin{aligned}
-(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) &= (\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}\\
&\quad + (\nabla\boldsymbol{\psi} - \mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \ell_1(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h).
\end{aligned} \tag{82}$$

It follows from (16) that

$$(\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} = (\nabla_w\boldsymbol{\psi},\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}$$

$$
\begin{aligned}
&= (\nabla_w \Pi_h(\mathbf{Q}_h\boldsymbol{\psi}),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&+ (\nabla_w(\boldsymbol{\psi} - \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h}. \qquad (83)
\end{aligned}
$$

The Eq. (47) implies

$$
(\varepsilon_h, \nabla_w \cdot \mathbf{Q}_h\boldsymbol{\psi})_{\mathcal{T}_h} = -\ell_4(\boldsymbol{\psi}, \varepsilon_h). \qquad (84)
$$

Using the Eqs. (48) and (84), we have

$$
\nu(\nabla_w \Pi_h(\mathbf{Q}_h\boldsymbol{\psi}),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} = \nu\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})) - \ell_4(\boldsymbol{\psi}, \varepsilon_h). \qquad (85)
$$

It follows from (83) and (85)

$$
\begin{aligned}
\nu(\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} &= \nu\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})) - \ell_4(\boldsymbol{\psi}, \varepsilon_h) \\
&+ \nu(\nabla_w(\boldsymbol{\psi} - \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h}. \qquad (86)
\end{aligned}
$$

Using (82) and (86), we have

$$
\begin{aligned}
-\nu(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) &= \nu\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})) - \ell_4(\boldsymbol{\psi}, \varepsilon_h) + \nu(\nabla_w(\boldsymbol{\psi} - \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&+ \nu(\nabla\boldsymbol{\psi} - \mathbf{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \nu\ell_1(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h). \qquad (87)
\end{aligned}
$$

Using (87), (81) becomes

$$
\begin{aligned}
-\nu(\Delta\boldsymbol{\psi}, \Pi_h\boldsymbol{\epsilon}_h) &= \nu\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})) - \ell_4(\boldsymbol{\psi}, \varepsilon_h) + \nu(\nabla_w(\boldsymbol{\psi} - \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&+ \nu(\nabla\boldsymbol{\psi} - \mathbf{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} - \nu\ell_1(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) + \nu(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h - \Pi_h\boldsymbol{\epsilon}_h). \qquad (88)
\end{aligned}
$$

It follows from integration by parts and (2), (17) and (22) that

$$
\begin{aligned}
(\nabla\xi,\ \Pi_h\boldsymbol{\epsilon}_h) &= -(\xi,\ \nabla\cdot\Pi_h\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} + \langle\xi,\ \Pi_h\boldsymbol{\epsilon}_h\cdot\mathbf{n}\rangle_{\partial\mathcal{T}_h} \\
&= -(\xi,\ \nabla\cdot\Pi_h\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&= -(Q_h\xi,\ \nabla\cdot\Pi_h\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&= -(Q_h\xi,\ \nabla_w\cdot\boldsymbol{\epsilon}_h)_{\mathcal{T}_h} \\
&= -(Q_h\xi,\ \nabla_w\cdot\mathbf{u})_{\mathcal{T}_h} + (Q_h\xi,\ \nabla_w\cdot\mathbf{u}_h)_{\mathcal{T}_h} \\
&= 0. \qquad (89)
\end{aligned}
$$

Combining (89) and (88) with(80), we have

$$
\begin{aligned}
\nu\|\Pi_h\boldsymbol{\epsilon}_h\|^2 &\le |\nu\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi}))| + |\ell_4(\boldsymbol{\psi}, \varepsilon_h)| + |\nu(\nabla_w(\boldsymbol{\psi} - \Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w \boldsymbol{\epsilon}_h)_{\mathcal{T}_h}| \\
&+ |\nu(\nabla\boldsymbol{\psi} - \mathbf{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}| + |\nu\ell_1(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h)| + |\nu(\Delta\boldsymbol{\psi}, \boldsymbol{\epsilon}_h - \Pi_h\boldsymbol{\epsilon}_h)| \\
&= \nu T_1 + T_2 + \nu T_3 + \nu T_4 + \nu T_5 + \nu T_6. \qquad (90)
\end{aligned}
$$

Next we estimate all the terms on the right hand side of (90). Using the Cauchy–Schwarz inequality, (20), adding and subtracting $\mathbf{Q}_h\boldsymbol{\psi}$, (15) and the definitions of $\mathbf{Q}_h$ and $\mathbb{Q}_h$ we obtain

$$
\begin{aligned}
T_1 &= |\ell_1(\mathbf{u}, \Pi_h(\mathbf{Q}_h\boldsymbol{\psi}))| \le \left|\langle(\nabla\mathbf{u} - \mathbb{Q}_h\nabla\mathbf{u})\cdot\mathbf{n},\ \Pi_h(\mathbf{Q}_h\boldsymbol{\psi}) - \{\Pi_h(\mathbf{Q}_h\boldsymbol{\psi})\}\rangle_{\partial\mathcal{T}_h}\right| \\
&\le C\left(\sum_{T\in\mathcal{T}_h} h_T\|(\nabla\mathbf{u} - \mathbb{Q}_h\nabla\mathbf{u})\|_{\partial T}^2\right)^{1/2}\left(\sum_{T\in\mathcal{T}_h} h_T^{-1}\|[\![\Pi_h(\mathbf{Q}_h\boldsymbol{\psi}) - \boldsymbol{\psi}]\!]\|_{\partial T}^2\right)^{1/2} \\
&\le C\left(\sum_{T\in\mathcal{T}_h} \|\nabla\mathbf{u} - \mathbb{Q}_h\nabla\mathbf{u}\|_T^2 + h_T^2\|\nabla\mathbf{u} - \mathbb{Q}_h\nabla\mathbf{u}\|_{1,T}^2\right)^{1/2}
\end{aligned}
$$

$$\left(\sum_{T\in\mathcal{T}_h} h_T^{-2}\|\Pi_h(\mathbf{Q}_h\boldsymbol{\psi})-\boldsymbol{\psi}\|_T^2+\|\Pi_h(\mathbf{Q}_h\boldsymbol{\psi})-\boldsymbol{\psi}\|_{1,T}^2\right)^{1/2}$$
$$\le Ch^{k+1}|\mathbf{u}|_{k+1}\|\boldsymbol{\psi}\|_2.$$

It follows from (62) and (65) that

$$T_2=|\ell_4(\boldsymbol{\psi},\varepsilon_h)|\le Ch\|\boldsymbol{\psi}\|_2\|\varepsilon_h\|\le C\nu h^{k+1}|\mathbf{u}|_{k+1}|\boldsymbol{\psi}|_2.$$

It follows from (63) and (64) that

$$\begin{aligned}
T_3&=|(\nabla_w(\boldsymbol{\psi}-\Pi_h(\mathbf{Q}_h\boldsymbol{\psi})),\ \nabla_w\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}|\\
&\le C{|\!|\!|}\boldsymbol{\epsilon}_h{|\!|\!|}\,{|\!|\!|}\boldsymbol{\psi}-\Pi_h(\mathbf{Q}_h\boldsymbol{\psi}){|\!|\!|}\\
&\le Ch^{k+1}|\mathbf{u}|_{k+1}|\boldsymbol{\psi}|_2.
\end{aligned}$$

The estimates (12) and (64) imply

$$\begin{aligned}
T_4&=|(\nabla\boldsymbol{\psi}-\mathbb{Q}_h\nabla\boldsymbol{\psi},\ \nabla\boldsymbol{\epsilon}_h)_{\mathcal{T}_h}|\\
&\le C\left(\sum_{T\in\mathcal{T}_h}\|\nabla\boldsymbol{\epsilon}_h\|_T^2\right)^{1/2}\left(\sum_{T\in\mathcal{T}_h}\|\nabla\boldsymbol{\psi}-\mathbb{Q}_h\nabla\boldsymbol{\psi}\|_T^2\right)^{1/2}\\
&\le C\left(\sum_{T\in\mathcal{T}_h}(\|\nabla(\mathbf{u}-\mathbf{Q}_h\mathbf{u})\|_T^2+\|\nabla(\mathbf{Q}_h\mathbf{u}-\mathbf{u}_h)\|_T^2)\right)^{1/2}\\
&\quad\times\left(\sum_{T\in\mathcal{T}_h}\|\nabla\boldsymbol{\psi}-\mathbb{Q}_h\nabla\boldsymbol{\psi}\|_T^2\right)^{1/2}\\
&\le Ch|\boldsymbol{\psi}|_2(h^k|\mathbf{u}|_{k+1}+{|\!|\!|}\mathbf{e}_h{|\!|\!|})\\
&\le Ch^{k+1}|\mathbf{u}|_{k+1}|\boldsymbol{\psi}|_2.
\end{aligned}$$

Using (12) and (64), we obtain

$$\begin{aligned}
T_5&=|\ell_1(\boldsymbol{\psi},\boldsymbol{\epsilon}_h)|=\left|\langle(\nabla\boldsymbol{\psi}-\mathbb{Q}_h\nabla\boldsymbol{\psi})\cdot\mathbf{n},\ \boldsymbol{\epsilon}_h-\{\boldsymbol{\epsilon}_h\}\rangle_{\partial\mathcal{T}_h}\right|\\
&\le\sum_{T\in\mathcal{T}_h}h_T^{1/2}\|\nabla\boldsymbol{\psi}-\mathbb{Q}_h\nabla\boldsymbol{\psi}\|_{\partial T}h_T^{-1/2}\|[\![\boldsymbol{\epsilon}_h]\!]\|_{\partial T}\\
&\le Ch\|\boldsymbol{\psi}\|_2(\sum_{T\in\mathcal{T}_h}h_T^{-1}(\|[\![\mathbf{e}_h]\!]\|_{\partial T}^2+\|[\![\mathbf{u}-\mathbf{Q}_h\mathbf{u}]\!]\|_{\partial T}^2))^{1/2}\\
&\le Ch\|\boldsymbol{\psi}\|_2({|\!|\!|}\mathbf{e}_h{|\!|\!|}+(\sum_{T\in\mathcal{T}_h}h_T^{-1}\|[\![\mathbf{u}-\mathbf{Q}_h\mathbf{u}]\!]\|_{\partial T}^2)^{1/2}\\
&\le Ch^{k+1}|\mathbf{u}|_{k+1}|\boldsymbol{\psi}|_2.
\end{aligned}$$

Using (15), (64) and (78), we have

$$\begin{aligned}
T_6&=|(\Delta\boldsymbol{\psi},\boldsymbol{\epsilon}_h-\Pi_h\boldsymbol{\epsilon}_h)|\le Ch\|\boldsymbol{\psi}\|_2\|\boldsymbol{\epsilon}_h\|_{1,h}\\
&\le Ch^{k+1}\|\mathbf{u}\|_{k+1}\|\boldsymbol{\psi}\|_2.
\end{aligned}$$

Combining all the estimates above with (90) yields

$$\nu\|\Pi_h\boldsymbol{\epsilon}_h\|^2\le C\nu h^{k+1}|\mathbf{u}|_{k+1}\|\boldsymbol{\psi}\|_2.$$

Using (78), we have

$$\|\Pi_h \boldsymbol{\epsilon}_h\| \le Ch^{k+1}|\mathbf{u}|_{k+1}.$$

It follows from the triangle inequality, the above estimate and (64)

$$\begin{aligned}\|\boldsymbol{\epsilon}_h\| &\le \|\boldsymbol{\epsilon}_h - \Pi_h \boldsymbol{\epsilon}_h\| + \|\Pi_h \boldsymbol{\epsilon}_h\| \\ &\le Ch\|\boldsymbol{\epsilon}_h\|_{1,h} + Ch^{k+1}|\mathbf{u}|_{k+1} \\ &\le Ch^{k+1}|\mathbf{u}|_{k+1}. \end{aligned} \tag{91}$$

We have completed the proof for Algorithm 2.

2). Next we shall prove the results for Algorithm 4. Testing (75) by $\Pi_h \boldsymbol{\epsilon}_h$ gives

$$\nu(\Pi_h \boldsymbol{\epsilon}_h, \Pi_h \boldsymbol{\epsilon}_h) = -\nu(\Delta \boldsymbol{\psi}, \Pi_h \boldsymbol{\epsilon}_h) + (\nabla \xi, \Pi_h \boldsymbol{\epsilon}_h). \tag{92}$$

Similarly as (89), by using the equivalence property in (25), we have

$$(\nabla \xi,\ \Pi_h \boldsymbol{\epsilon}_h) = 0. \tag{93}$$

By subtracting and adding $\nu(\Delta \boldsymbol{\psi}, \boldsymbol{\epsilon}_h)$ to (92) and using (93), we obtain

$$\begin{aligned}\nu(\Pi_h \boldsymbol{\epsilon}_h, \Pi_h \boldsymbol{\epsilon}_h) &= -\nu(\Delta \boldsymbol{\psi}, \boldsymbol{\epsilon}_h) + \nu(\Delta \boldsymbol{\psi}, \boldsymbol{\epsilon}_h - \Pi_h \boldsymbol{\epsilon}_h) \\ &= \nu \mathcal{A}_h(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) + \nu(\Delta \boldsymbol{\psi}, \boldsymbol{\epsilon}_h - \Pi_h \boldsymbol{\epsilon}_h). \end{aligned} \tag{94}$$

Here we have used the definition of bilinear form $\mathcal{A}_h(\cdot,\cdot)$. Next by subtracting and adding $\Pi_h \mathbf{Q}_h \boldsymbol{\psi}$, using (57), (25) and (47), it implies

$$\begin{aligned}\nu \mathcal{A}_h(\boldsymbol{\psi}, \boldsymbol{\epsilon}_h) &= \nu \mathcal{A}_h(\boldsymbol{\psi} - \Pi_h \mathbf{Q}_h \boldsymbol{\psi}, \boldsymbol{\epsilon}_h) + \nu \mathcal{A}_h(\Pi_h \mathbf{Q}_h \boldsymbol{\psi}, \boldsymbol{\epsilon}_h) \\ &= \nu \mathcal{A}_h(\boldsymbol{\psi} - \Pi_h \mathbf{Q}_h \boldsymbol{\psi}, \boldsymbol{\epsilon}_h) - \mathcal{B}_h(\mathbf{Q}_h \boldsymbol{\psi}, \varepsilon_h) \\ &= \nu \mathcal{A}_h(\boldsymbol{\psi} - \Pi_h \boldsymbol{\psi}, \boldsymbol{\epsilon}_h) - \ell_4(\boldsymbol{\psi}, \varepsilon_h). \end{aligned}$$

Combining (94) and above equation gives

$$\begin{aligned}\nu\|\Pi_h \boldsymbol{\epsilon}_h\|^2 &\le \nu\|\boldsymbol{\psi} - \Pi_h \boldsymbol{\psi}\|_{1,h}\|\boldsymbol{\epsilon}_h\|_{1,h} + \nu\|\boldsymbol{\psi}\|_2\|\boldsymbol{\epsilon}_h - \Pi_h \boldsymbol{\epsilon}_h\| + |\ell_4(\boldsymbol{\psi}, \varepsilon_h)| \\ &\le C\left(\nu h\|\boldsymbol{\psi}\|_2\|\boldsymbol{\epsilon}_h\|_{1,h} + h|\boldsymbol{\psi}|_2\|\varepsilon_h\|\right) \\ &\le C\nu h^{k+1}\|\Pi_h \boldsymbol{\epsilon}_h\|\ |\mathbf{u}|_{k+1}. \end{aligned}$$

Canceling the common factors in the above estimate and combining (15) (64), we finally prove estimate

$$\begin{aligned}\|\boldsymbol{\epsilon}_h\| &\le \|\boldsymbol{\epsilon}_h - \Pi_h \boldsymbol{\epsilon}_h\| + \|\Pi_h \boldsymbol{\epsilon}_h\| \\ &\le Ch\|\boldsymbol{\epsilon}_h\|_{1,h} + h^{k+1}|\mathbf{u}|_{k+1} \\ &\le Ch^{k+1}|\mathbf{u}|_{k+1}. \end{aligned}$$

□

Remark 7 We also cite the L^2 error estimate for Algorithms 1 and 3 for comparison. Let $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times W_h$ be the solutions from Algorithms 1 or 3. Assume η is chosen big enough when Algorithm 3 is employed. We have

$$\|\mathbf{u} - \mathbf{u}_h\| \le Ch^{k+1}\left(\|\mathbf{u}\|_{k+2} + \frac{1}{\nu}\|p\|_{k+1}\right).$$

The pressure dependent term $\frac{1}{\nu}\|p\|_{k+1}$ has been eliminated in Theorem 2. Thus, we claim that Algorithms 2 and 4 are pressure-robust schemes.

6 Numerical Experiments

In this section, several numerical experiments will be carried out to validate our theoretical conclusions in Theorems 1 and 2 and compare the results for the non-pressure-robust schemes. By using discontinuous elements with degree $k \geq 1$, we are expecting the optimal rate in convergence as follows,

$$\begin{aligned} H^1\text{-seminorm for velocity: } & \|\nabla \boldsymbol{\epsilon}_h\| := \|\nabla(\mathbf{u} - \mathbf{u}_h)\| \approx \mathcal{O}(h^k), \\ L^2\text{-norm for velocity: } & \|\boldsymbol{\epsilon}_h\| := \|\mathbf{u} - \mathbf{u}_h\| \approx \mathcal{O}(h^{k+1}) \\ L^2\text{-norm for pressure: } & \|\varepsilon_h\| := \|p - Q_h p\| \approx \mathcal{O}(h^k). \end{aligned}$$

We shall test the following two-dimensional problem on the uniform triangular grid with mesh size h. Let $\Omega = (0, 1) \times (0, 1)$, and the exact solutions with homogeneous Dirichlet boundary condition for velocity, are chosen as follows for testing:

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

6.1 Test 1—Convergence Tests for CDG Schemes

The first test has been carried out for testing the convergence results for the proposed enhanced Algorithm 2 and non-pressure-robust Algorithm 1. The error profiles and convergence results are reported in Table 1, 2 and 3. We observe:

- The convergence orders for varying polynomial degrees are all optimal as expected.
- In the case of $k = 1, 2$, as we reduce the values in ν, the velocity errors produced by Algorithm 2 will be increased by a factor of $\frac{1}{\nu}$, though optimal rates in convergence can still be achieved. Such increasing errors are due to the non-pressure-robustness in the scheme. In contrast to the velocity error, the pressure error will remain the same as we decrease the values in ν. This confirms the results in Remarks 6 and 7.
- In the case of $k = 1, 2$, all the errors are smaller than the non-pressure-robust scheme Algorithm 1 and thus Algorithm 2 outperforms Algorithm 1. Furthermore, the velocity errors remain the same for varying values in ν; the pressure errors are reduced by a factor of ν. This validates the robustness in our proposed schemes.
- When the simulation approximates velocity by $[P_3(T)]^2$ element and approximates pressure by $P_2(T)$ element, the pressure has been fully resolved. In this case, one can observe that the velocity errors produced by Algorithms 1 and 2 are comparable. However, due to the introduced inconsistency errors, the pressure error from Algorithm 2 is slightly larger than that from Algorithm 1, though both two types errors are at the same magnitude.
- All the above observations agree with our theoretical conclusions.

6.2 Test 2—Convergence Test for IPDG Schemes

In this section, we shall compare the numerical performance for our proposed enhance scheme with classical IPDG finite element methods. In all the computation, we take $\eta = 4k^2$. The numerical results and error profiles are reported in Table 4, 5 and 6. The similar numerical conclusions as 6.1 can be obtained. Again, we validate the theoretical conclusions.

Table 1 Example 6.1. Error profiles and convergence results for CDG schemes with $k = 1$

$1/h$	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	ν = 1, Non-pressure-robust algorithm 1						ν = 1, Pressure-robust algorithm 2					
4	3.39E–1		3.51E–2		9.74E–1		1.53E–1		7.96E–3		2.65E–2	
8	1.80E–1	0.9	1.01E–2	1.8	4.79E–1	1.0	7.96E–2	0.9	2.69E–3	1.6	1.26E–2	1.1
16	9.05E–2	1.0	2.63E–3	1.9	2.29E–1	1.1	3.93E–2	1.0	7.89E–4	1.8	9.87E–3	0.4
32	4.51E–2	1.0	6.66E–4	2.0	1.10E–1	1.1	1.94E–2	1.0	2.11E–4	1.9	5.98E–3	0.7
64	2.25E–2	1.0	1.67E–4	2.0	5.39E–2	1.0	9.64E–3	1.0	5.40E–5	2.0	3.20E–3	0.9
	ν = 1E–2, Non-pressure-robust algorithm 1						ν = 1E–2, Pressure-robust algorithm 2					
4	3.07E+1		3.47		9.74E–1		1.53E–1		7.96E–3		2.65E–4	
8	1.64E+1	0.9	9.91E–1	1.8	4.79E–1	1.0	7.96E–2	0.9	2.69E–3	1.6	1.26E–4	1.1
16	8.21	1.0	2.56E–1	2.0	2.29E–1	1.1	3.93E–2	1.0	7.89E–4	1.8	9.87E–5	0.4
32	4.09	1.0	6.43E–2	2.0	1.10E–1	1.1	1.94E–2	1.0	2.11E–4	1.9	5.98E–5	0.7
64	2.04	1.0	1.61E–2	2.0	5.37E–2	1.0	9.64E–3	1.0	5.40E–5	2.0	3.20E–5	0.9
	ν = 1E–4, Non-pressure-robust algorithm 1						ν = 1E–4, Pressure-robust algorithm 2					
4	3.07E+3		3.47E+2		9.74E–1		1.53E–1		7.96E–3		2.65E–6	
8	1.64E+3	0.9	9.91E+1	1.8	4.79E–1	1.0	7.96E–2	0.9	2.69E–3	1.6	1.26E–6	1.1
16	8.21E+2	1.0	2.56E+1	2.0	2.29E–1	1.1	3.93E–2	1.0	7.89E–4	1.8	9.87E–7	0.4
32	4.09E+2	1.0	6.43	2.0	1.10E–1	1.1	1.94E–2	1.0	2.11E–4	1.9	5.98E–7	0.7
64	2.04E+2	1.0	1.61	2.0	5.37E–2	1.0	9.64E–3	1.0	5.40E–5	2.0	3.20E–7	0.9
	ν = 1E–6, non-pressure-robust algorithm 1						ν = 1E–6, pressure-robust algorithm 2					
4	3.07E+5		3.47E+4		9.74E-1		1.53E–1		7.96E–3		2.65E–8	
8	1.64E+5	0.9	9.91E+3	1.8	4.79E–1	1.0	7.96E–2	0.9	2.69E–3	1.6	1.26E–8	1.1
16	8.21E+4	1.0	2.56E+3	2.0	2.29E–1	1.1	3.93E–2	1.0	7.89E–4	1.8	9.87E–9	0.4
32	4.09E+4	1.0	6.43E+2	2.0	1.10E–1	1.1	1.94E–2	1.0	2.11E–4	1.9	5.98E–9	0.7
64	2.04E+4	1.0	1.61E+2	2.0	5.37E–2	1.0	9.64E–3	1.0	5.40E–5	2.0	3.20E–9	0.9

Table 2 Example 6.1. Error profiles and convergence results for CDG schemes with $k = 2$

$1/h$	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	$\nu = 1$, Non-pressure-robust algorithm 1						$\nu = 1$, Pressure-robust algorithm 2					
4	6.76E–2		2.89E–3		4.48E–2		3.74E–2		1.15E–3		1.29E–2	
8	1.75E–2	2.0	3.70E–4	3.0	9.99E–3	2.2	9.84E–3	1.9	1.34E–4	3.1	4.31E–3	1.6
16	4.39E–3	2.0	4.66E–5	3.0	2.44E–3	2.0	2.49E–3	2.0	1.59E–5	3.1	2.20E–3	1.0
32	1.10E–3	2.0	5.86E–6	3.0	6.15E–4	2.0	6.22E–4	2.0	1.96E–6	3.0	7.67E–4	1.5
64	2.75E–4	2.0	7.35E–7	3.0	1.55E–4	2.0	1.55E–4	2.0	2.44E–7	3.0	2.24E–4	1.8
	ν = 1E–2, Non-pressure-robust algorithm 1						ν = 1E–2, Pressure-robust algorithm 2					
4	5.55		2.75E–1		4.52E–2		3.74E–2		1.15E–3		1.29E–4	
8	1.41	2.0	3.52E–2	3.0	9.93E–3	2.2	9.84E–3	1.9	1.34E–4	3.1	4.31E–5	1.6
16	3.55E–1	2.0	4.45E–3	3.0	2.35E–3	2.1	2.49E–3	2.0	1.59E–5	3.1	2.20E–5	1.0
32	8.89E–2	2.0	5.61E–4	3.0	5.75E–4	2.0	6.22E–4	2.0	1.96E–6	3.0	7.67E–6	1.5
64	2.22E–2	2.0	7.03E–5	3.0	1.42E–4	2.0	1.55E–4	2.0	2.44E–7	3.0	2.24E–6	1.8
	ν = 1E–4, Non-pressure-robust algorithm 1						ν = 1E–4, Pressure-robust algorithm 2					
4	5.55E+2		2.75E+1		4.52E–2		3.74E–2		1.15E–3		1.29E–6	
8	1.41E+2	2.0	3.52	3.0	9.93E–3	2.2	9.84E–3	1.9	1.34E–4	3.1	4.31E–7	1.6
16	3.55E+1	2.0	4.45E–1	3.0	2.35E–3	2.1	2.49E–3	2.0	1.59E–5	3.1	2.20E–7	1.0
32	8.89	2.0	5.61E–2	3.0	5.75E–4	2.0	6.22E–4	2.0	1.96E–6	3.0	7.67E–8	1.5
64	2.22	2.0	7.03E–3	3.0	1.42E–4	2.0	1.55E–4	2.0	2.44E–7	3.0	2.24E–8	1.8
	ν = 1E–6, non-pressure-robust algorithm 1						ν = 1E–6, pressure-robust algorithm 2					
4	5.55E+4		2.75E+3		4.52E–2		3.74E–2		1.15E–3		1.29E–8	
8	1.41E+4	2.0	3.52E+2	3.0	9.93E–3	2.2	9.84E–3	1.9	1.34E–4	3.1	4.31E–9	1.6
16	3.55E+3	2.0	4.45E+1	3.0	2.35E–3	2.1	2.49E–3	2.0	1.59E–5	3.1	2.20E–9	1.0
32	8.89E+2	2.0	5.61	3.0	5.75E–4	2.0	6.22E–4	2.0	1.96E–6	3.0	7.67E–10	1.5
64	2.22E+2	2.0	7.03E–1	3.0	1.42E–4	2.0	1.55E–4	2.0	2.44E–7	3.0	2.24E–10	1.8

Table 3 Example 6.1. Error profiles and convergence results for CDG schemes with $k = 3$

$1/h$	$\|\nabla\epsilon_h\|$	Rate	$\|\epsilon_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla\epsilon_h\|$	Rate	$\|\epsilon_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	ν = 1, Non-pressure-robust algorithm 1						ν = 1, Pressure-robust algorithm 2					
4	7.59E–3		1.25E–4		1.60E–3		7.37E–3		1.43E–4		2.40E–3	
8	1.05E–3	2.9	8.85E–6	3.8	3.56E–4	2.2	1.02E–3	2.9	9.81E–6	3.9	5.07E–4	2.2
16	1.32E–4	3.0	5.80E–7	3.9	6.60E–5	2.4	1.30E–4	3.0	6.39E–7	3.9	1.18E–4	2.1
32	1.65E–5	3.0	3.71E–8	4.0	9.86E–6	2.7	1.63E–5	3.0	4.07E–8	4.0	1.89E–5	2.6
64	2.06E–6	3.0	2.35E–9	4.0	1.34E–6	2.9	2.04E–6	3.0	2.57E–9	4.0	2.64E–6	2.8
	ν = 1E–2, non-pressure-robust algorithm 1						ν = 1E–2, pressure-robust algorithm 2					
4	7.59E–3		1.25E–4		1.60E–5		7.37E–3		1.43E–4		2.40E–5	
8	1.05E–3	2.9	8.85E–6	3.8	3.56E–6	2.2	1.02E–3	2.9	9.81E–6	3.9	5.07E–6	2.2
16	1.32E–4	3.0	5.80E–7	3.9	6.60E–7	2.4	1.30E–4	3.0	6.39E–7	3.9	1.18E–6	2.1
32	1.65E–5	3.0	3.71E–8	4.0	9.86E–8	2.7	1.63E–5	3.0	4.07E–8	4.0	1.89E–7	2.6
64	2.06E–6	3.0	2.35E–9	4.0	1.34E–8	2.9	2.04E–6	3.0	2.57E–9	4.0	2.64E–8	2.8
	ν = 1E–4, non-pressure-robust algorithm 1						ν = 1E–4, pressure-robust algorithm 2					
4	7.59E–3		1.25E–4		1.60E–7		7.37E–3		1.43E–4		2.40E–7	
8	1.05E–3	2.9	8.85E–6	3.8	3.56E–8	2.2	1.02E–3	2.9	9.81E–6	3.9	5.07E–8	2.2
16	1.32E–4	3.0	5.80E–7	3.9	6.60E–9	2.4	1.30E–4	3.0	6.39E–7	3.9	1.18E–8	2.1
32	1.65E–5	3.0	3.71E–8	4.0	9.86E–10	2.7	1.63E–5	3.0	4.07E–8	4.0	1.89E–9	2.6
64	2.06E–6	3.0	2.35E–9	4.0	1.34E–10	2.9	2.04E–6	3.0	2.57E–9	4.0	2.64E–10	2.8
	ν = 1E–6, non-pressure-robust algorithm 1						ν = 1E–6, pressure-robust algorithm 2					
4	7.59E–3		1.25E–4		1.60E–9		7.37E–3		1.43E–4		2.40E–9	
8	1.05E–3	2.9	8.85E–6	3.8	3.56E–10	2.2	1.02E–3	2.9	9.81E–6	3.9	5.07E–10	2.2
16	1.32E–4	3.0	5.80E–7	3.9	6.60E–11	2.4	1.30E–4	3.0	6.39E–7	3.9	1.18E–10	2.1
32	1.65E–5	3.0	3.71E–8	4.0	9.86E–12	2.7	1.63E–5	3.0	4.07E–8	4.0	1.89E–11	2.6
64	2.06E–6	3.0	2.35E–9	4.0	1.34E–12	2.9	2.04E–6	3.0	2.57E–9	4.0	2.64E–12	2.8

Table 4 Example 6.2. Error profiles and convergence results for IPDG schemes with $k = 1$

$1/h$	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	ν = 1, Non-pressure-robust algorithm 3						ν = 1, Pressure-robust algorithm					
4	5.69E–1		5.20E–2		5.76E–1		1.96E–1		1.02E–2		3.28E–2	
8	2.63E–1	1.1	1.25E–2	2.1	2.11E–1	1.4	1.03E–1	0.9	3.19E–3	1.7	1.41E–2	1.2
16	1.16E–1	1.2	2.86E–3	2.1	7.25E–2	1.5	4.67E–2	1.1	8.71E–4	1.9	1.02E–2	0.5
32	5.12E–2	1.2	6.60E–4	2.1	2.64E–2	1.5	2.13E–2	1.1	2.26E–4	1.9	6.39E–3	0.7
64	2.32E–2	1.1	1.56E–4	2.1	1.08E–2	1.3	1.01E–2	1.1	5.75E–5	2.0	3.52E–3	0.9
	ν = 1e–2, Non-pressure-robust algorithm 3						ν = 1e–2, Pressure-robust algorithm					
4	5.34E+1		5.13		5.75E–1		1.96E–1		1.02E–2		3.28E–4	
8	2.44E+1	1.1	1.22	2.1	2.10E–1	1.5	1.03E–1	0.9	3.19E–3	1.7	1.41E–4	1.2
16	1.07E+1	1.2	2.77E–1	2.1	7.12E–2	1.6	4.67E–2	1.1	8.71E–4	1.9	1.02E–4	0.5
32	4.65	1.2	6.33E–2	2.1	2.52E–2	1.5	2.13E–2	1.1	2.26E–4	1.9	6.39E–5	0.7
64	2.09	1.2	1.48E–2	2.1	9.86E–3	1.4	1.01E–2	1.1	5.75E–5	2.0	3.52E–5	0.9
	ν = 1e–4, Non-pressure-robust algorithm 3						ν = 1e–4, Pressure-robust algorithm					
4	5.34E+3		5.13E+2		5.75E–1		1.96E–1		1.02E–2		3.28E–6	
8	2.44E+3	1.1	1.22E+2	2.1	2.10E–1	1.5	1.03E–1	0.9	3.19E–3	1.7	1.41E–6	1.2
16	1.07E+3	1.2	2.77E+1	2.1	7.12E–2	1.6	4.67E–2	1.1	8.71E–4	1.9	1.02E–6	0.5
32	4.65E+2	1.2	6.33	2.1	2.52E–2	1.5	2.13E–2	1.1	2.26E–4	1.9	6.39E–7	0.7
64	2.09E+2	1.2	1.48	2.1	9.86E–3	1.4	1.01E–2	1.1	5.75E–5	2.0	3.52E–7	0.9
	ν = 1e–6, Non-pressure-robust algorithm 3						ν = 1e–6, Pressure-robust algorithm					
4	5.34E+5		5.13E+4		5.75E–1		1.96E–1		1.02E–2		3.28E–8	
8	2.44E+5	1.1	1.22E+4	2.1	2.10E–1	1.5	1.03E–1	0.9	3.19E–3	1.7	1.41E–8	1.2
16	1.07E+5	1.2	2.77E+3	2.1	7.12E–2	1.6	4.67E–2	1.1	8.71E–4	1.9	1.02E–8	0.5
32	4.65E+4	1.2	6.33E+2	2.1	2.52E–2	1.5	2.13E–2	1.1	2.26E–4	1.9	6.39E–9	0.7
64	2.09E+4	1.2	1.48E+2	2.1	9.86E–3	1.4	1.01E–e	1.1	5.75E–5	2.0	3.52E–9	0.9

Table 5 Example 6.2. Error profiles and convergence results for IPDG schemes with $k = 2$

$1/h$	$\|\nabla\boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla\boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	ν = 1, Non-pressure-robust algorithm 3						ν = 1, Pressure-robust algorithm					
4	5.24E–2		2.06E–3		4.98E–2		4.04E–2		1.51E–3		9.12E–2	
8	1.35E–2	2.0	2.52E–4	3.0	1.60E–2	1.6	1.04E–2	2.0	1.59E–4	3.2	2.58E–2	1.8
16	3.36E–3	2.0	3.11E–5	3.0	4.61E–3	1.8	2.59E–3	2.0	1.72E–5	3.2	6.03E–3	2.1
32	8.37E–4	2.0	3.88E–6	3.0	1.23E–3	1.9	6.43E–4	2.0	2.03E–6	3.1	1.38E–3	2.1
64	2.09E–4	2.0	4.86E–7	3.0	3.16E–4	2.0	1.61E–4	2.0	2.54E–7	3.0	3.45E–4	2.0
	ν = 1e–2, Non-pressure-robust algorithm 3						ν = 1e–2, pressure-robust algorithm					
4	3.30		1.65E–1		6.43E–2		4.04E–2		1.51E–3		9.12E–4	
8	8.16E–1	2.0	2.08E–2	3.0	1.51E–2	2.1	1.04E–2	2.0	1.59E–4	3.2	2.58E–4	1.8
16	2.02E–1	2.0	2.62E–3	3.0	3.65E–3	2.0	2.59E–3	2.0	1.72E–5	3.2	6.03E–5	2.1
32	5.04E–2	2.0	3.30E–4	3.0	8.97E–4	2.0	6.43E–4	2.0	2.03E–6	3.1	1.38E–5	2.1
64	1.26E–2	2.0	4.13E–5	3.0	2.24E–4	2.0	1.61E–4	2.0	2.54E–7	3.0	3.45E–6	2.0
	ν = 1e–4, Non-pressure-robust algorithm 3						ν = 1e–4, pressure-robust algorithm					
4	3.30E+2		1.65E+1		6.50E–2		4.04E–2		1.51E–3		9.12E–6	
8	8.16E+1	2.0	2.08	3.0	1.53E–2	2.1	1.04E–2	2.0	1.59E–4	3.2	2.58E–6	1.8
16	2.02E+1	2.0	2.62E–1	3.0	3.72E–3	2.0	2.59E–3	2.0	1.72E–5	3.2	6.03E–7	2.1
32	5.04	2.0	3.30E–2	3.0	9.16E–4	2.0	6.43E–4	2.0	2.03E–6	3.1	1.38E–7	2.1
64	1.26	2.0	4.13E–3	3.0	2.29E–4	2.0	1.61E–4	2.0	2.54E–7	3.0	3.45E–8	2.0
	ν = 1e–6, Non-pressure-robust algorithm 3						ν = 1e–6, pressure-robust algorithm					
4	3.30E+4		1.65E+3		6.50E–2		4.04E–2		1.51E–3		9.12E–8	
8	8.16E+3	2.0	2.08E+2	3.0	1.53E–2	2.1	1.04E–2	2.0	1.59E–4	3.2	2.58E–8	1.8
16	2.02E+3	2.0	2.62E+1	3.0	3.72E–3	2.0	2.59E–3	2.0	1.72E–5	3.2	6.03E–9	2.1
32	5.04E+2	2.0	3.30	3.0	9.16E–4	2.0	6.43E–4	2.0	2.03E–6	3.1	1.38E–9	2.1
64	1.26E+2	2.0	4.13E–1	3.0	2.29E–4	2.0	1.61E–4	2.0	2.54E–7	3.0	3.45E–10	2.0

Table 6 Example 6.2. Error profiles and convergence results for IPDG schemes with $k = 3$

$1/h$	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate	$\|\nabla \boldsymbol{\epsilon}_h\|$	Rate	$\|\boldsymbol{\epsilon}_h\|$	Rate	$\|\varepsilon_h\|$	Rate
	$\nu = 1$, Non-pressure-robust algorithm 3						$\nu = 1$, Pressure-robust algorithm					
4	7.62E–3		1.79E–4		2.34E–2		7.77E–3		1.96E–4		2.14E–2	
8	1.05E–3	2.9	1.14E–5	4.0	4.01E–3	2.5	1.05E–3	2.9	1.21E–5	4.0	4.19E–3	2.4
16	1.34E–4	3.0	7.18E–7	4.0	5.78E–4	2.8	1.34E–4	3.0	7.46E–7	4.0	6.41E–4	2.7
32	1.68E–5	3.0	4.52E–8	4.0	7.74E–5	2.9	1.68E–5	3.0	4.66E–8	4.0	8.82E–5	2.9
64	2.10E–6	3.0	2.83E–9	4.0	9.68E–6	3.0	2.10E–6	3.0	2.91E–9	4.0	1.10E–5	3.0
	ν = 1e–2, Non-pressure-robust algorithm 3						ν = 1e–2, Pressure-robust algorithm					
4	7.62E–3		1.79E–4		2.34E–4		7.77E–3		1.96E–4		2.14E–4	
8	1.05E–3	2.9	1.14E–5	4.0	4.01E–5	2.5	1.05E–3	2.9	1.21E–5	4.0	4.19E–5	2.4
16	1.34E–4	3.0	7.18E–7	4.0	5.78E–6	2.8	1.34E–4	3.0	7.46E–7	4.0	6.41E–6	2.7
32	1.68E–5	3.0	4.52E–8	4.0	7.74E–7	2.9	1.68E–5	3.0	4.66E–8	4.0	8.82E–7	2.9
64	2.10E–6	3.0	2.83E–9	4.0	9.68E–8	3.0	2.10E–6	3.0	2.91E–9	4.0	1.10E–7	3.0
	ν = 1e–4, Non-pressure-robust algorithm 3						ν = 1e–4, Pressure-robust algorithm					
4	7.62E–3		1.79E–4		2.34E–6		7.77E–3		1.96E–4		2.14E–6	
8	1.05E–3	2.9	1.14E–5	4.0	4.01E–7	2.5	1.05E–3	2.9	1.21E–5	4.0	4.19E–9	2.4
16	1.34E–4	3.0	7.18E–7	4.0	5.78E–8	2.8	1.34E–4	3.0	7.46E–7	4.0	6.41E–10	2.7
32	1.68E–5	3.0	4.52E–8	4.0	7.74E–9	2.9	1.68E–5	3.0	4.66E–8	4.0	8.82E–11	2.9
64	2.10E–6	3.0	2.83E–9	4.0	9.68E–10	3.0	2.10E–6	3.0	2.91E–9	4.0	1.10E–11	3.0
	ν = 1e–6, Non-pressure-robust algorithm 3						ν = 1e–6, Pressure-robust algorithm					
4	7.62E–3		1.79E–4		2.34E–8		7.77E–3		1.96E–4		2.14E–8	
8	1.05E–3	2.9	1.14E–5	4.0	4.01E–9	2.5	1.05E–3	2.9	1.21E–5	4.0	4.19E–9	2.4
16	1.34E–4	3.0	7.18E–7	4.0	5.78E–10	2.8	1.34E–4	3.0	7.46E–7	4.0	6.41E–10	2.7
32	1.68E–5	3.0	4.52E–8	4.0	7.74E–11	2.9	1.68E–5	3.0	4.66E–8	4.0	8.82E–11	2.9
64	2.10E–6	3.0	2.83E–9	4.0	9.68E–12	3.0	2.10E–6	3.0	2.91E–9	4.0	1.10E–12	3.0

7 Conclusion

In this paper, we investigate the velocity reconstruction techniques for enhancing the pressure-robustness in the existing DG scheme. Both theoretical analysis and numerical experiments have been provided to show the optimal rate in convergence and pressure-robustness. By only modifying the source term, our new schemes can achieve the desired pressure-robust features with minimal efforts. Both theoretical analysis and numerical experiments are provided to demonstrate the advantages in the new schemes. In the future, we shall investigate more applications in the incompressible fluid simulation and time dependent problems.

References

1. Al-Taweel, A., Wang, X.: A note on the optimal degree of the weak gradient of the stabilizer free weak Galerkin finite element method. Appl. Numer. Math. **150**, 444–451 (2020)
2. Brennecke, C., Linke, A., Merdon, C., Schöberl, J.: Optimal and pressure-independent L^2 velocity error estimates for a modified Crouzeix–Raviart Stokes element with BDM reconstructions. J. Comput. Math. **33**, 191–208 (2015)
3. Brezzi, F., Boffi, D., Demkowicz, L., Durán, R.G., Falk, R.S., Fortin, M.: Mixed finite elements, compatibility conditions, and applications. Lectures given at the C.I.M.E. Summer School held in Cetraro, Italy, June 26–July 1, 2008. Springer, Berlin, Heidelberg (2008)
4. Brezzi, F., Falk, R.S.: Stability of higher-order Hood–Taylor methods. SIAM J. Numer. Anal. **28**, 581–590 (1991)
5. Chen, L., Wang, M., Zhong, L.: Convergence analysis of triangular MAC schemes for two dimensional Stokes equations. J. Sci. Comput. **63**, 716–744 (2015)
6. Cockburn, B., Kanschat, G., Schötzau, D.: A locally conservative LDG method for the incompressible Navier–Stokes equations. Math. Comput. **74**, 1067–1095 (2005)
7. Cockburn, B., Kanschat, G., Schötzau, D.: A note on discontinuous Galerkin divergence-free solutions of the Navier–Stokes equations. J. Sci. Comput. **31**, 61–73 (2007)
8. Crouzeix, M., Raviart, P.A.: Conforming and nonconforming finite element methods for solving the stationary Stokes equations I. ESAIM: Mathematical Modelling and Numerical Analysis–Modélisation Mathématique et Analyse Numérique **7**, 33–75 (1973)
9. Falk, R., Neilan, M.: Stokes complexes and the construction of stable finite elements with pointwise mass conservation. SIAM J. Numer. Anal. **51**, 1308–1326 (2013)
10. Girault, V., Raviart, P.: Finite Element Methods for the Navier-Stokes Equations: Theory and Algorithms. Springer, Berlin (1986)
11. Gunzburger, M.D.: Finite Element Methods for Viscous Incompressible Flows. Practice and Algorithms. A Guide to Theory, Academic Press, San Diego (1989)
12. Guzmán, J., Neilan, M.: Conforming and divergence free Stokes elements on general triangular meshes. Math. Comput. **83**, 15–36 (2014)
13. Guzmán, J., Neilan, M.: Conforming and divergence-free Stokes elements in three dimensions. IMA J. Numer. Anal. **34**, 1489–1508 (2014)
14. Hansbo, P., Larson, M.G.: Discontinuous Galerkin methods for incompressible and nearly incompressible elasticity by Nitsches method. Comput. Methods Appl. Mech. Eng. **191**, 1895–1908 (2002)
15. Hood, P., Taylor, C.: Numerical solution of the Navier–Stokes equations using the finite element technique. Comput. Fluids **1**, 1–28 (1973)
16. Jenkins, E., John, V., Linke, A., Rebholz, L.: On the parameter choice in grad-div stabilization for the Stokes equations. Adv. Comput. Math. **40**, 491–516 (2014)
17. John, V., Linke, A., Merdon, C., Neilan, M., Rebholz, L.: On the divergence constraint in mixed finite element mehtods for incompressible flows. SIAM Review
18. Lederer, P.: 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 (2016)
19. Lederer, P., Schöberl, J.: Polynomial robust stability analysis for h(div)-conforming finite elements for the stokes equations. IMA J. Numer. Anal. **38**, 1832–1860 (2018)
20. Linke, A.: A divergence-free velocity reconstruction for incompressible flows. C. R. Math. Acad. Sci. Paris **350**, 837–840 (2012)

21. Linke, A.: On the role of the Helmholtz decomposition in mixed methods for incompressible flows and a new variational crime. Comput. Methods Appl. Mech. Eng. **268**, 782–800 (2014)
22. Linke, A., Matthies, G., Tobiska, L.: Robust arbitrary order mixed finite element methods for the incompressible Stokes equations with pressure independent velocity errors. ESAIM?: M2AN **50**, 289–309 (2016)
23. Mu, L.: Pressure robust weak Galerkin finite element methods for Stokes problems. SIAM J. Sci. Comput. **42**, B608–B629 (2020)
24. Mu, L., Wang, X., Ye, X.: A modified weak Galerkin finite element method for the Stokes equations. J. Comput. Appl. Math. **275**, 79–90 (2015)
25. Mu, L., Ye, X., Zhang, S.: A Stabilizer free, pressure robust, and superconvergence weak Galerkin finite element method for the stokes equations on polytopal mesh. SIAM J. Sci. Comput. (Preprint)
26. Olshanskii, M., Olshanskii, A.: Grad-div stabilization for Stokes equations. Math. Comput. **73**, 1699–1718 (2004)
27. Olshanskii, M., Lube, G., Heister, T., Löwe, J.: Grad-div stabilization and subgrid pressure models for the incompressible Navier–Stokes equations. Comput. Methods Appl. Mech. Eng. **198**, 3975–3988 (2009)
28. Rivière, B.: Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation. Society for Industrial and Applied Mathematics (2008)
29. Scott, L., Vogelius, M.: Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials. ESAIM:M2AN **19**, 111–143 (1985)
30. Wang, J., Ye, X.: New finite element methods in computational fluid dynamics by H(div) elements. SIAM J. Numer. Anal. **45**, 1269–1286 (2007)
31. Wang, J., Ye, X.: A weak Galerkin mixed finite element method for second-order elliptic problems. Math. Comput. **83**, 2101–2126 (2014)
32. Ye, X., Zhang, S.: A conforming discontinuous Galerkin finite element method. Int. J. Numer. Anal. Model. **17**, 110–117 (2020)
33. Ye, X., Zhang, S.: A conforming discontinuous Galerkin finite element method: part II. Int. J. Numer. Anal. Model. **17**, 281–296 (2020)
34. Ye, X., Zhang, S.: A conforming discontinuous Galerkin finite element method for the Stokes problem on polytopal meshes. Int. J. Numer. Meth. Fluids **93**, 1913–1928 (2021)
35. Zhang, S.: Divergence-free finite elements on tetrahedral grids for $k \geq 6$. Math. Comput. **80**, 669–695 (2011)

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