Journal of Computational Physics 228 (2009) 5961–5977

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A finite element variational multiscale method for incompressible flows based on two local gauss integrations ☆

Haibiao Zheng *, Yanren Hou, Feng Shi, Lina Song

College of Science, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 26 November 2008
Received in revised form 30 March 2009
Accepted 4 May 2009
Available online 15 May 2009

MSC:
65M55
65M70

Keywords:
Incompressible flows
Variational multiscale (VMS) method
Projection
Two local Gauss integrations

ABSTRACT

In this article, we present a finite element variational multiscale (VMS) method for incompressible flows based on two local Gauss integrations, and compare it with common VMS method which is defined by a low order finite element space L_h on the same grid as X_h for the velocity deformation tensor and a stabilization parameter α. The best algorithmic feature of our method is using two local Gauss integrations to replace projection operator. We theoretically discuss the relationship between our method and common VMS method for the Taylor–Hood elements, and show that the nonlinear system derived from our method by finite element discretization is much smaller than that of common VMS method computationally.

Additionally we present numerical simulations to demonstrate the effectiveness, storage, computational complexity of our method. Finally, we give some numerical simulations of the nonlinear flow problems to show good stability and accuracy properties of the method.

1. Introduction

The incompressible Navier–Stokes equations model Newtonian fluids, such as air flow at low speed and water flow, and finite element methods for their simulations have been one class of the most successful methods [1–3]. However, there still remain some important but challenging problems. For example, the discretization of the Navier–Stokes equations by finite element methods may generally bring out two shortcomings: the violation of the discrete inf-sup condition and spurious oscillations due to the domination of convection term. There are huge literature on finite element methods for Navier–Stokes equations. Among them, we list some methods as follows: recently developed stabilized methods, such as, Galerkin least square(GLS) method introduced in [4–6] by Franca, Hughes, and their collaborators; classical large eddy simulation (LES) approach in [7,8] which treats the large scales as an average in space given by convolution with an appropriate filter function; two-level stabilization scheme in [9]; variational multiscale (VMS) method which defines the large scales in a different way, namely by a projection into appropriate subspaces, see Guermond [10], Hughes et al. [11–13] and Layton [14], and other literatures on VMS methods [15–24]; residual-free bubbles(RFB) method [25–27]; three-level method [28] and local projection stabilization [29]; etc.

☆ Supported by NSF of China (Grant No. 10871156) and NCET.
* Corresponding author. Tel.: +86 29 82675559; fax: +86 29 83237910.
E-mail addresses: hbzheng13@gmail.com (H. Zheng), yrhou@mail.xjtu.edu.cn (Y. Hou), fengshi81@yahoo.com.cn (F. Shi), lnsong365@gmail.com (L. Song).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.05.006

In common VMS methods, the large scales are defined by projections into appropriate function spaces. There is a class of VMS methods based on a three-scale decomposition of the flow field into large, resolved small and unresolved scales [30]. The two local Gauss integrations method was first developed to offset the discrete pressure space by the residual of the simple and symmetry term at element level in order to circumvent the inf-sup condition (see e.g., [31,32]). This paper focuses on a finite element VMS method for incompressible flows based on two local Gauss integrations. It also can be cast in the framework of VMS method which is defined by a low order finite element space L_h on the same grid as X_h for the velocity deformation tensor. This method avoids constructing the projection operator, and keeps the same efficiency, does not add extra storage compared with common VMS method which introduces additional dependent variables.

The organization of this paper is as follows. In Section 2 we introduce the governing equations, the notations and some well-known results used throughout this paper. A finite element VMS method based on two local Gauss integrations is formulated in Section 3. In next section we discuss the relation between common VMS method and our method with respect to their implementation. Then in Section 5, numerical simulations of the nonlinear steady flow problems are shown to verify the good stability and accuracy properties of our method. Finally, we end with a short conclusion in Section 6.

2. Governing equations

Incompressible flows are modeled by the following Navier–Stokes equations:

$$\begin{aligned} &-\nu\Delta u+(u\cdot\nabla)u+\nabla p=f \quad \text{in } \Omega,\\ &\nabla\cdot u=0 \quad \text{in } \Omega,\\ &u=0 \quad \text{on } \partial\Omega, \end{aligned} \tag{2.1}$$

where Ω represents a polyhedral domain in $R^d, d=2,3$, with boundary $\partial\Omega, u$ the fluid velocity, p the pressure, f the prescribed body force, and $\nu>0$ the kinematic viscosity. Given a characteristic length scale L and velocity scale U, the Reynolds number is defined by $Re=UL/\nu$.

The classical weak formulation of (2.1) reads: find $(u,p)\in(X,Q)$ satisfying

$$\begin{aligned} &\nu a(u,v)+b(u,u,v)-d(p,v)=(f,v) \quad \forall\ v\in X,\\ &d(q,u)=0 \quad \forall\ q\in Q. \end{aligned} \tag{2.2}$$

Here, we used notations

$$\begin{aligned} &X=H_0^1(\Omega)^d,\ Y=L^2(\Omega)^d,\ Q=L_0^2(\Omega)=\left\{q\in L^2(\Omega);\int_\Omega p\mathbf{d}x=0\right\},\\ &a(u,v)=(\nabla u,\nabla v), d(p,v)=(\nabla\cdot v,p), b(u,v,w)=((u\cdot\nabla)v,w), \end{aligned}$$

and $(\cdot,\cdot)$ is the inner product in $L^2(\Omega)$ or $L^2(\Omega)^{d\times d}$. The norm in $L^2(\Omega)^d$ and norm in the standard Sobolev space $H^k(\Omega)^d$ are denoted by $\|\cdot\|_0$ and $\|\cdot\|_k$, respectively. The space X is equipped with the norm $\{(\nabla\cdot,\nabla\cdot)\}^{1/2}$ or $\|\cdot\|_1$, and Q is equipped with the usual L^2 norm.

For the finite element discretization, let τ_h be the regular triangulations of the domain Ω, and define the mesh parameter $h=\max_{\Omega_e\in\tau_h}\{\mathrm{diam}(\Omega_e)\}$. We choose the conforming velocity-pressure finite element space $(X_h,Q_h)\subset(X,Q)$ satisfying the discrete inf-sup condition

$$\inf_{q_h\in Q_h}\sup_{v_h\in X_h}\frac{(q_h,\nabla\cdot v_h)}{\|q_h\|_0\|\nabla v_h\|_0}\geqslant\beta>0, \tag{2.3}$$

where β is independent of h. Here we consider the Taylor–Hood elements [2]:

$$\begin{aligned} &X_h=\{u_h\in C(\Omega)^d|u_h|_{\Omega_e}\in P_2(\Omega_e)^d \quad \forall\Omega_e\in\tau_h\},\\ &Q_h=\{q_h\in C(\Omega)|q_h|_{\Omega_e}\in P_1(\Omega_e) \quad \forall\Omega_e\in\tau_h\}, \end{aligned}$$

where $P_k(\Omega_e), k=1,2$ is the space of k^{th}order polynomials on Ω_e. We will also need the piecewise constant space

$$R_0=\{v_h\in L^2(\Omega)|v_h|_{\Omega_e}\in P_0(\Omega_e) \quad \forall\Omega_e\in\tau_h\},$$

where $P_0(\Omega_e)$ is the space of all constant polynomial on Ω_e.

Table 4.1
Information on the grids and the numbers of degrees of freedom in two dimensions.

$1/h$	cells	X_h	M_h	L_h	$G(u_h,v_h)$
4	32	384	96	128	0
8	128	1536	384	512	0
16	512	6144	1536	2048	0
32	2048	24576	6144	8192	0

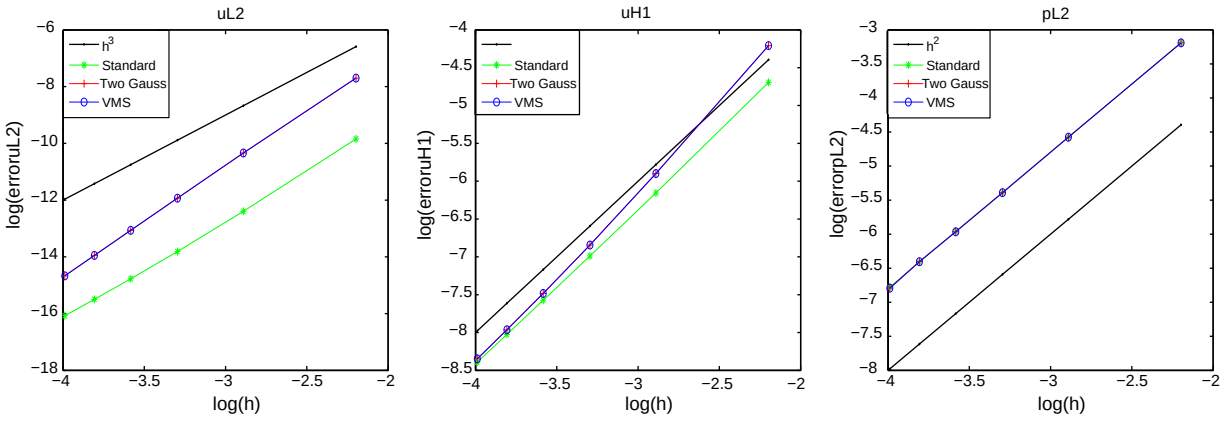


Fig. 5.1. Convergence analysis for the velocity and the pressure using different methods. left: L^2 error for the velocity; middle: H^1 error for the velocity; right: L^2 error for the pressure.

Table 5.1
Rates of convergence using common VMS method with $Re = 10000$ and $\alpha = 0.1\,h^2$.

h	$\frac{\|u-u_h\|_{L^2}}{\|u\|_{L^2}}$	Order	$\frac{\|u-u_h\|_{H^1}}{\|u\|_{H^1}}$	Order	$\frac{\|p-p_h\|_{L^2}}{\|p\|_{L^2}}$	Order	Iterations	CPU(s)
1/8	0.1221	/	0.3063	/	0.01562	/	66	10.957
1/16	0.01920	2.6684	0.1127	1.4422	0.003906	2.0000	37	23.943
1/24	0.005530	3.0701	0.05401	1.8150	0.001736	2.0001	24	35.954
1/32	0.002193	3.2161	0.02955	2.0968	0.0009766	1.9999	21	57.456

Table 5.2
Rates of convergence using our stabilization method with $Re = 10000$ and $\alpha = 0.1\,h^2$.

h	$\frac{\|u-u_h\|_{L^2}}{\|u\|_{L^2}}$	order	$\frac{\|u-u_h\|_{H^1}}{\|u\|_{H^1}}$	order	$\frac{\|p-p_h\|_{L^2}}{\|p\|_{L^2}}$	order	Iterations	CPU(s)
1/8	0.1221	/	0.3063	/	0.01562	/	66	5.939
1/16	0.01920	2.6684	0.1127	1.4422	0.003906	2.0000	37	13.419
1/24	0.005530	3.0701	0.05401	1.8150	0.001736	2.0001	24	20.593
1/32	0.002193	3.2161	0.02955	2.0968	0.0009766	1.9999	21	32.063

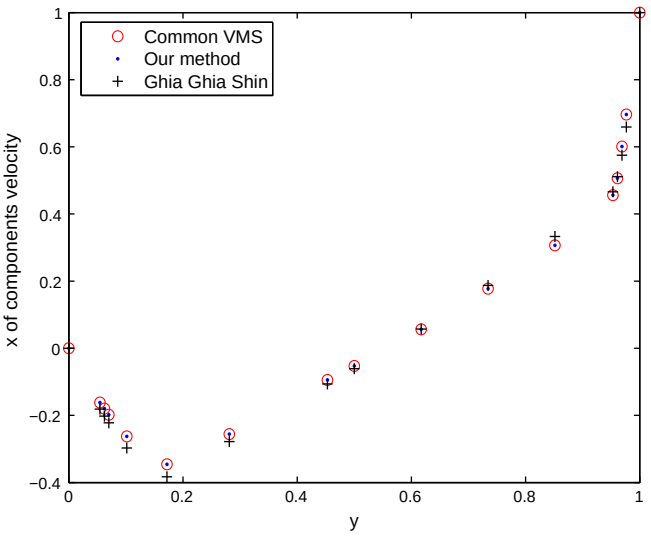


Fig. 5.2. vertical midlines for $Re = 1000$, $h = 1/24$.

3. A finite element variational multiscale method based on two local Gauss integrations

As mentioned in the introduction, a Galerkin finite element discretization of (2.2) is unstable in the case of higher Reynolds number (or smaller viscosity). Therefore, stabilization becomes necessary. We firstly consider a common version of VMS methods which was proposed in [14] for the steady case. We define two spaces $L = L^2(\Omega)^{d\times d}$ and $L_h = R_0(\Omega)^{d\times d}$, the latter is defined on the same grid as X_h for the velocity deformation tensor. The common VMS method we consider hereafer is: find $(u_h, p_h) \in (X_h, Q_h)$ and $g_h \in L_h$ satisfying

$$\begin{aligned} &(v+\alpha)a(u_h, v_h) - \alpha(g_h, \nabla v_h) + b(u_h, u_h, v_h) - d(p_h, v_h) = (f, v_h) \quad \forall v_h \in X_h, \\ &d(q_h, u_h) = 0 \quad \forall q_h \in Q_h, \\ &(g_h - \nabla u_h, l_h) = 0 \quad \forall l_h \in L_h. \end{aligned} \tag{3.1}$$

This system is determined by the choices of L_h and α. The stabilization parameter α in this scheme acts only on the small scales. There is also another version of VMS methods which is based on two-grid stabilization such as schemes in [15,19], and will not be discussed in this paper.

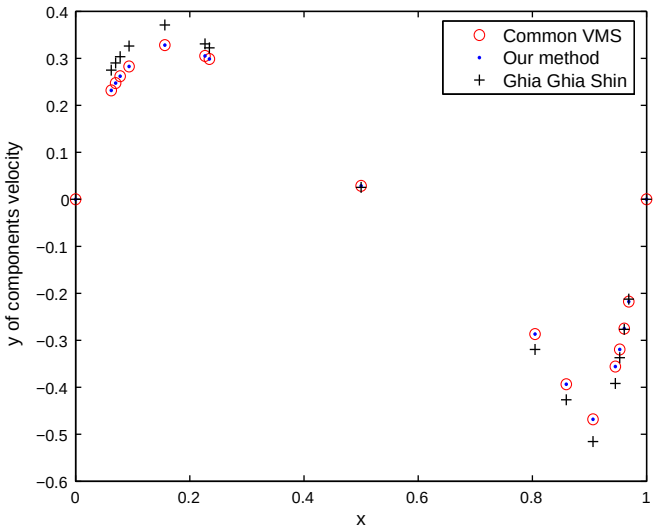


Fig. 5.3. Horizontal midlines for $Re = 1000$, $h = 1/24$.

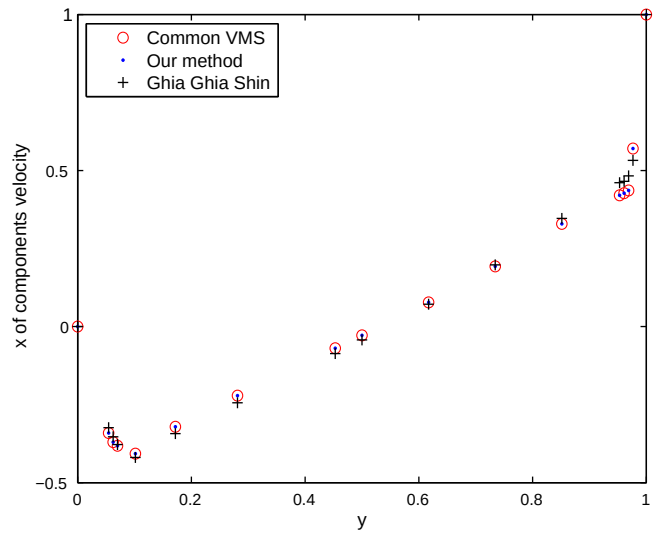


Fig. 5.4. vertical midlines for $Re = 3200$, $h = 1/32$.

Although this method (3.1) is shown to preserve stability and efficiency, the extra storage might be significant since it introduces four additional dependent variables in g_h. This situation will be more serious for three dimensional case. In [2,14] the authors exploit an equivalent form of the viscous term for div-free functions to reduce this extra storage. In the following, we will give another more efficient method to reduce this extra storage without introducing any additional variables. It is easy to verify that the last equation in (3.1) implies that g_h is the L^2 projection of ∇u_h onto L_h. Now we define the orthogonal projection operator $\Pi : L \to L_h$ with the following properties:

$$((I-\Pi)l, g_h) = 0 \quad \forall l \in L, g_h \in L_h, \quad (H1)$$
$$\|\Pi l\|_0 \leqslant C\|l\|_0 \quad \forall l \in L, \quad (H2)$$
$$\|(I-\Pi)l\|_0 \leqslant Ch\|l\|_1 \quad \forall l \in L \cap H^1(\Omega)^{d\times d}. \quad (H3)$$

Then (3.1) can be rewritten as

$$\begin{aligned} &va(u_h, v_h) + \alpha((I-\Pi)\nabla u_h, (I-\Pi)\nabla v_h) + b(u_h, u_h, v_h) - d(p_h, v_h) = (f, v_h) \quad \forall v_h \in X_h, \\ &d(q_h, u_h) = 0 \quad \forall q_h \in Q_h. \end{aligned} \tag{3.2}$$

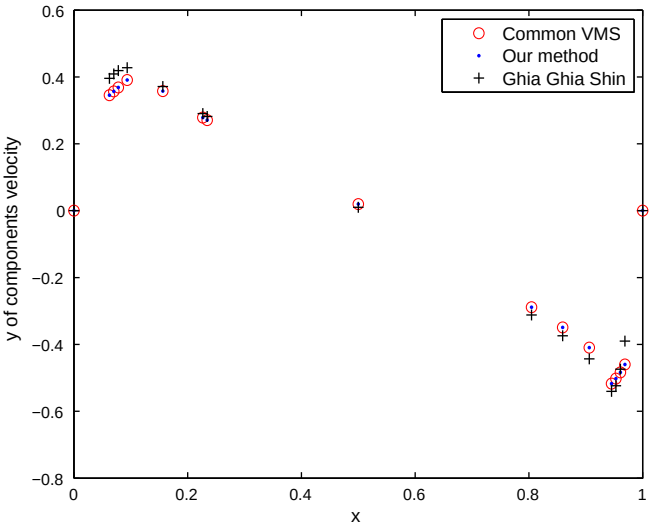


Fig. 5.5. Horizontal midlines for $Re = 3200$, $h = 1/32$.

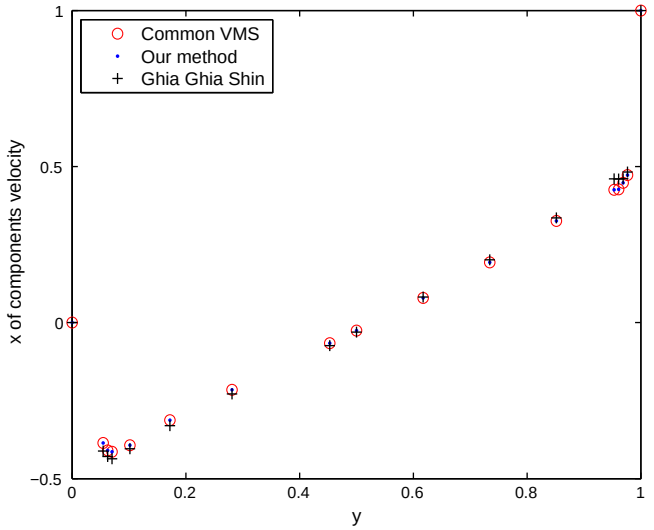


Fig. 5.6. vertical midlines for $Re = 5000$, $h = 1/48$.

To deal with the stabilized term $\alpha((I-\Pi)\nabla u_h,(I-\Pi)\nabla v_h)$ in (3.2), we supply the local stabilization form of the difference between a consistent and an under-integrated mass matrices based on two local Gauss integrations at element level as follows:

$$G(u_h,v_h)=\alpha(a_k(u_h,v_h)-a_1(u_h,v_h)). \tag{3.3}$$

Here

$$\begin{aligned}
&a_k(u_h,v_h)=u_G^T M_k v_G, \quad a_1(u_h,v_h)=u_G^T M_1 v_G,\\
&u_G^T=[u_1,u_2,\cdots,u_N]^T, \quad v_G=[v_1,v_2,\cdots,v_N],\\
&M_{ij}=(\nabla\phi_i,\nabla\phi_j),u_h=\Sigma_{i=1}^N u_i\phi_i,u_i=u_h(x_i) \quad \forall u_h\in X_h, \quad i=1,2,\cdots,N,\\
&M_k=(M_{ij}^k)_{N\times N}, \quad M_1=(M_{ij}^1)_{N\times N}
\end{aligned}$$

and ϕ_i is the basis function of the velocity on the domain Ω such that its value is one at node x_i and zero at other nodes; and N is the dimension of X_h; the symmetric and positive matrices $M_{ij}^k, k\geqslant 2$ and M_{ij}^1 are the stiffness matrices computed by using

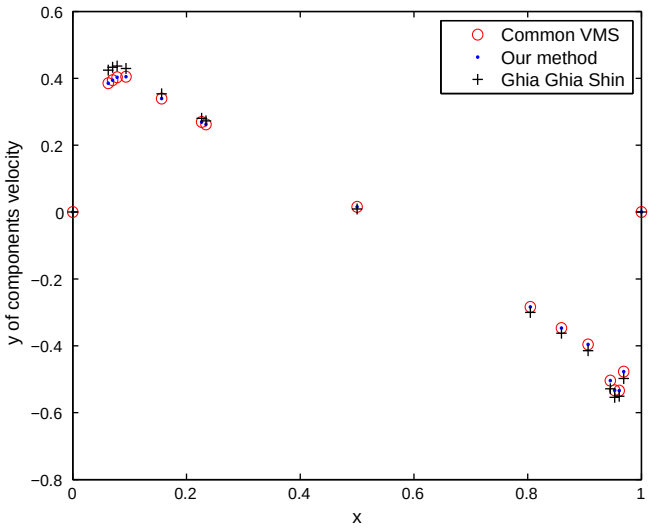


Fig. 5.7. Horizontal midlines for $Re=5000$, $h=1/48$.

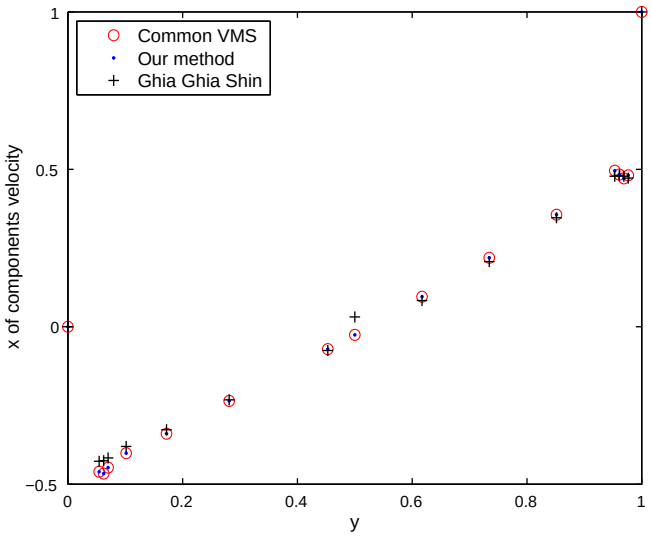


Fig. 5.8. Vertical midlines for $Re=10,000$, $h=1/64$.

k-order and 1-order Gauss integrations at element level, respectively; u_i and $v_i, i = 0, 1, \cdots, N$ are the values of u_h and v_h at the node x_i.

In detail, the stabilized term can be rewritten as

$$G(u_h, v_h) = \alpha \sum_{\Omega_e \in \tau_h} \left\{ \int_{\Omega_e, k} \nabla u_h \nabla v_h \mathbf{d}x - \int_{\Omega_e, 1} \nabla u_h \nabla v_h \mathbf{d}x \right\} \quad \forall u_h, v_h \in X_h,$$

where $\int_{\Omega_e, i} g(x)\mathbf{d}x$ denotes an appropriate Gauss integral over Ω_e which is exact for polynomials of degree $i, i = 1, k$. For all test functions $v_h \in X_h, \nabla u_h$ must be piecewise constant when $i = 1$.

Therefore, we give a stabilization method based on two local Gauss integrations as follows: find $(u_h, p_h) \in (X_h, Q_h)$ such that

$$\begin{aligned} &va(u_h, v_h) + b(u_h, u_h, v_h) - d(p_h, v_h) + G(u_h, v_h) = (f, v_h) \quad \forall v_h \in X_h, \\ &d(q_h, u_h) = 0 \quad \forall q_h \in Q_h. \end{aligned} \tag{3.4}$$

Remark 1. Our method based on two local Gauss integrations is only suitable for the Galerkin method using the Taylor–Hood spaces. Throughout the paper, our analysis and numerical tests are all carried out for this case.

Remark 2. Although we don't discuss the choice of α here, to our knowledge, for higher Reynolds number, the constant α should be chosen as the scale of $O(h)$ in order to stabilize the convective term appropriately. But, in order to keep the rates of convergence with the Galerkin method using the Taylor–Hood elements, we set $\alpha = O(h^2)$, and α may depend on Re. While computing the benchmark problem, we choose $\alpha = O(h)$ for simplicity, because we just need to show the stability and efficiency.

4. Relationship between common VMS method and our method

In this section, we will compare our method (3.4) with the common VMS method (3.1).

Firstly, we will discuss the equivalence between these two methods.

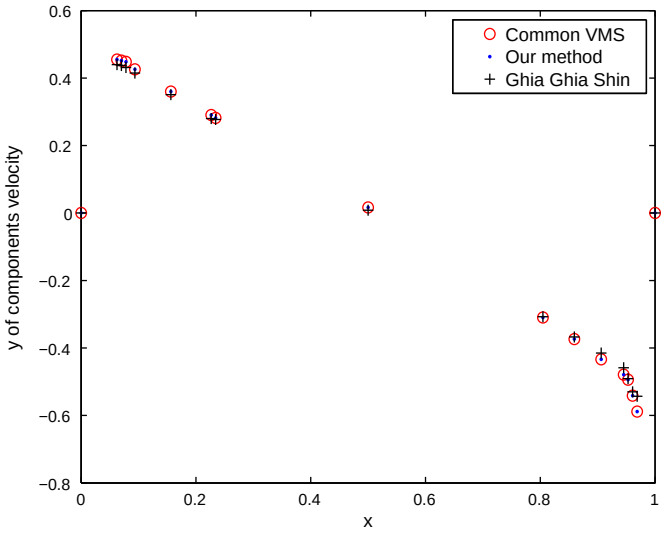


Fig. 5.9. Horizontal midlines for $Re = 10,000$, $h = 1/64$.

Table 5.3
CPU time(s) for the two methods.

Re	1000	3200	5000	10,000
h	1/24	1/32	1/48	1/64
Common VMS	64.87	271.29	1078.05	4940.34
Our method	28.41	114.32	368.37	799.31
Save	56.2%	57.9%	65.8%	83.8%

To prove the equivalence between our method (3.4) and the common VMS method (3.1), we only need to show $G(u_h, v_h) = \alpha((I-\Pi)\nabla u_h, (I-\Pi)\nabla v_h)$. Due to the orthogonality of projection operator Π, it suffices to prove that at each element

$$\int_{\Omega_e,1} \nabla u_h \nabla v_h \mathbf{d}x = (\Pi\nabla u_h, \Pi\nabla v_h)_{\Omega_e}.$$

Assume that $q_i, i = 1,2,3$ are the three vertices of Ω_e. For $u_h, v_h \in X_h$, we have $\nabla u_h, \nabla v_h \in P_1(\Omega)^{d\times d}$. Then we obtain

$$\begin{aligned}\int_{\Omega_e,1} \nabla u_h \nabla v_h \mathbf{d}x &= \int_{\Omega_e,k} \nabla u_h\Big(\frac{q_1+q_2+q_3}{3}\Big)\nabla v_h\Big(\frac{q_1+q_2+q_3}{3}\Big)\mathbf{d}x \\ &= \int_{\Omega_e,k} \frac{\nabla u_h(q_1)+\nabla u_h(q_2)+\nabla u_h(q_3)}{3}\,\frac{\nabla v_h(q_1)+\nabla v_h(q_2)+\nabla v_h(q_3)}{3}\mathbf{d}x \\ &= \int_{\Omega_e,k} \Pi\nabla u_h \Pi\nabla v_h \mathbf{d}x = (\Pi\nabla u_h, \Pi\nabla v_h)_{\Omega_e}. \end{aligned} \tag{4.1}$$

The equivalence is true.

Now, we will discuss the difference between the two methods in the aspect of implementation.

Since we use the Taylor–Hood spaces (X_h, Q_h), we define L_h on the same grid as (X_h, Q_h) using piecewise constant polynomials. We will describe the implementation in two dimension. The modifications in three dimensions are obvious.

Let's begin with common VMS method (3.1). Let the velocity vector u_h and the symmetric tensor G_h be given by

$$u_h = \begin{pmatrix} u_h^1 \\ u_h^2 \end{pmatrix}, \quad G_h = \begin{pmatrix} g_h^{11} & g_h^{12} \\ g_h^{21} & g_h^{22} \end{pmatrix}$$

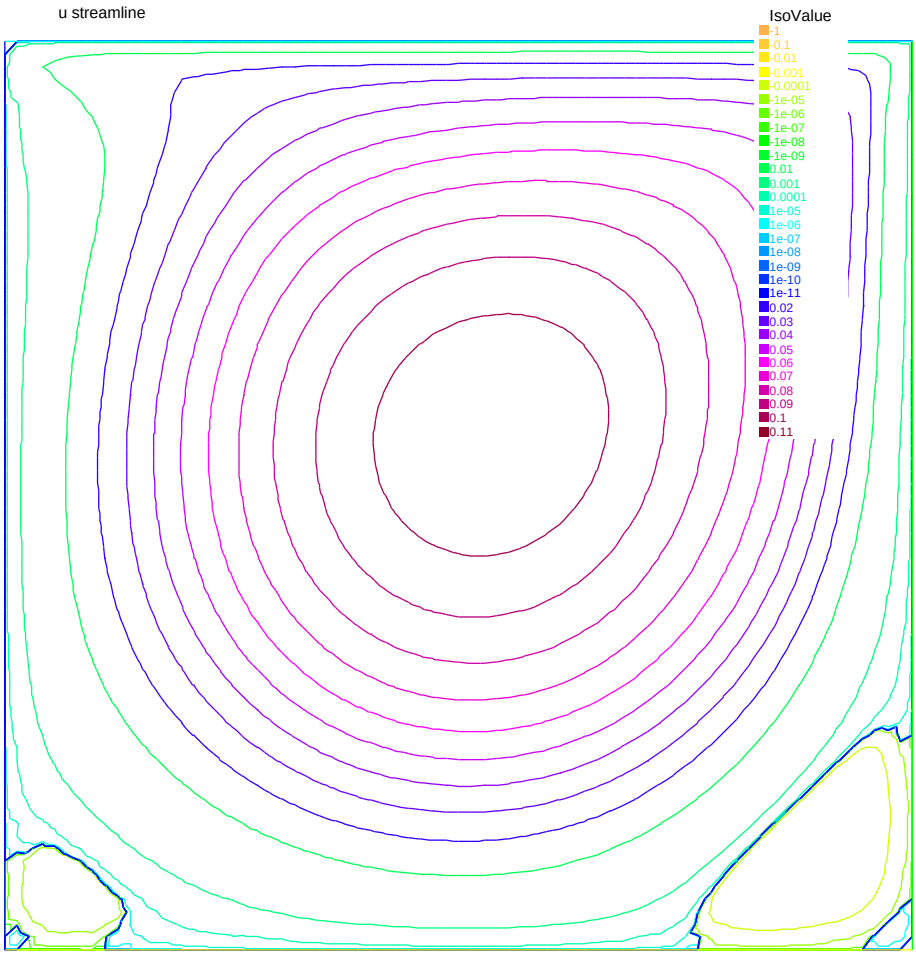


Fig. 5.10. Velocity streamlines for $Re = 1000$, $h = 1/24$.

and the spaces X_h and L_h be equipped with the bases

$$X_h = span\left\{\begin{pmatrix}\phi_h^i\\0\end{pmatrix},\begin{pmatrix}0\\\phi_h^i\end{pmatrix} : i = 1,\cdots,N_0\right\},$$

$$L_h = span\left\{\begin{pmatrix}\psi_h^i & 0\\0 & 0\end{pmatrix},\begin{pmatrix}0 & \psi_h^i\\0 & 0\end{pmatrix},\begin{pmatrix}0 & 0\\\psi_h^i & 0\end{pmatrix},\begin{pmatrix}0 & 0\\0 & \psi_h^i\end{pmatrix} : i = 1,\cdots,N_1\right\},$$

where N_0 and N_1 indicate the numbers of basis of X_h and L_h, respectively.

After an appropriate linearization of the convective term by Newton iteration, one obtains a linear saddle point problem of the following form:

$$\begin{pmatrix}
A_{11} & A_{12} & B_1^T & \tilde{G}_{11}^T & \tilde{G}_{12}^T & \tilde{G}_{13}^T & \tilde{G}_{14}^T\\
A_{21} & A_{22} & B_2^T & \tilde{G}_{21}^T & \tilde{G}_{22}^T & \tilde{G}_{23}^T & \tilde{G}_{24}^T\\
B_1 & B_2 & 0 & 0 & 0 & 0 & 0\\
G_{11} & G_{21} & 0 & M & 0 & 0 & 0\\
G_{12} & G_{22} & 0 & 0 & M & 0 & 0\\
G_{13} & G_{23} & 0 & 0 & 0 & M & 0\\
G_{14} & G_{24} & 0 & 0 & 0 & 0 & M
\end{pmatrix} * \begin{pmatrix}u_h^1\\u_h^2\\p_h\\g_h^{11}\\g_h^{12}\\g_h^{21}\\g_h^{22}\end{pmatrix} = \begin{pmatrix}f_h^1\\f_h^2\\0\\0\\0\\0\\0\end{pmatrix}. \tag{4.2}$$

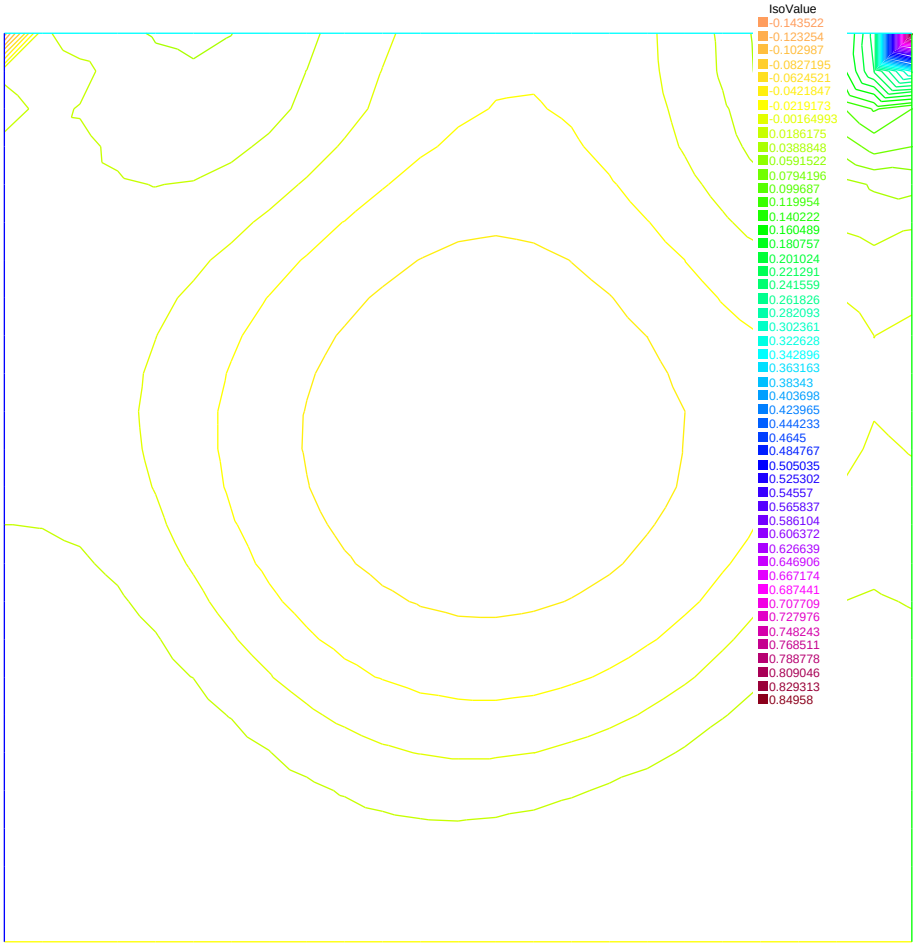


Fig. 5.11. Pressure level lines for $Re = 1000$, $h = 1/24$.

The matrices $A_{11},\cdots,A_{22}$ and B_1,B_2 have to be assembled if (3.1) is discretized without the terms involving G_h, i.e., if all scales are stabilized. The matrix M is the mass matrix of L_h: $(M)_{ij}=(\psi_h^i,\psi_h^j)$. The general entries of the matrices $G_{11},\cdots,G_{24}$ can be computed easily using the basis of X_h and L_h. Straightforward calculations give

$$(G_{11})_{ij}=\left(\begin{pmatrix}(\phi_h^i)_x & (\phi_h^i)_y\\ (\phi_h^i)_y & 0\end{pmatrix},\begin{pmatrix}\psi_h^j & 0\\ 0 & 0\end{pmatrix}\right)=((\phi_h^i)_x,\psi_h^j),$$
$$\left(\tilde{G}_{11}^T\right)_{ij}=\alpha(G_{11}^T)_{ij}$$

and all other blocks $G_{mn},\tilde{G}_{mn}^T$ follow from the definition easily.

However, for our method (3.4) we just obtain a linear saddle point problem of the following form:

$$\begin{pmatrix}\hat{A}_{11} & A_{12} & B_1^T\\ A_{21} & \hat{A}_{22} & B_2^T\\ B_1 & B_2 & 0\end{pmatrix}*\begin{pmatrix}u_h^1\\ u_h^2\\ p_h\end{pmatrix}=\begin{pmatrix}f_h^1\\ f_h^2\\ 0\end{pmatrix},\tag{4.3}$$

where

$$\begin{pmatrix}\hat{A}_{11} & 0\\ 0 & \hat{A}_{22}\end{pmatrix}=\begin{pmatrix}A_{11} & 0\\ 0 & A_{22}\end{pmatrix}+\begin{pmatrix}S_{11} & 0\\ 0 & S_{22}\end{pmatrix}$$

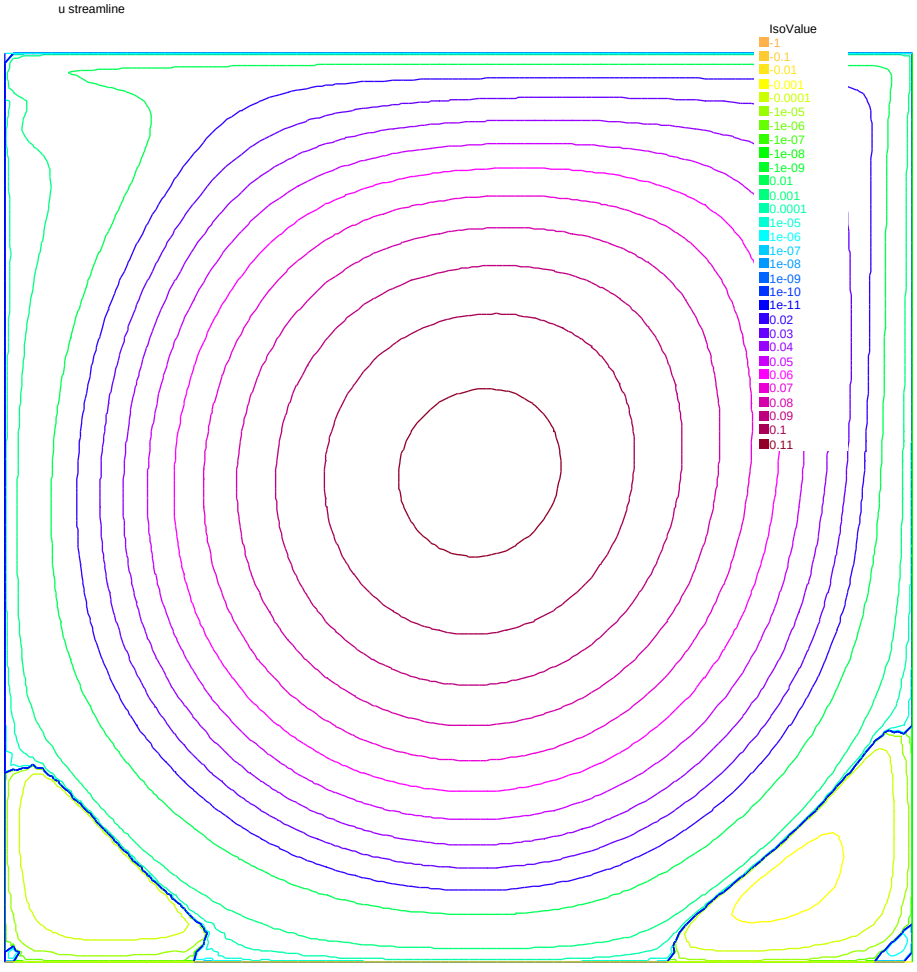


Fig. 5.12. Velocity streamlines for $Re=3200$, $h=1/32$.

and

$$S_{ii} = \alpha \sum_{\Omega_e \in \tau_h} \left\{ \int_{\Omega_e,k} \nabla \phi_h^i \nabla \phi_h^i \mathbf{d}x - \int_{\Omega_e,1} \nabla \phi_h^i \nabla \phi_h^i \mathbf{d}x \right\}.$$

Comparing (4.3) with (4.2), we can see that: for common VMS method (3.1), we need to solve a more larger linear system than that of our method, because our method doesn't introduce any additional dependent variables while the common VMS method introducing four additional dependent variables.

Besides, we also give the information on the grids and the numbers of degrees of freedom for the two methods in Table 4.1. From this table, we find that common VMS method will add more extra degrees of freedom in L_h than the standard Galerkin method while h decreases, and our method doesn't add any extra degrees of freedom.

Combining all discussions above, we can derive a conclusion that our method is more efficient than common VMS method without introducing any extra variables and degrees of freedom, but get the same stabilization effect. The numerical tests later will also verify this conclusion.

Remark 3. While taking the numerical tests, the stabilized term in our method (3.4) will be treated explicitly, such as:

$$G(u_h^{j+1}, v_h) = \alpha \sum_{\Omega_e \in \tau_h} \left\{ \int_{\Omega_e,k} \nabla u_h^{j+1} \nabla v_h \mathbf{d}x - \int_{\Omega_e,1} \nabla u_h^{j} \nabla v_h \mathbf{d}x \right\} \quad \forall u_h^j, v_h \in X_h,$$

where j denotes the number of Newton iteration. Meanwhile the stabilization term in (3.1) will be treated similarly.

5. Numerical results

In all experiments, the nonlinear systems are solved by Newton iteration, and algorithms are implemented using public domain finite element software [35].

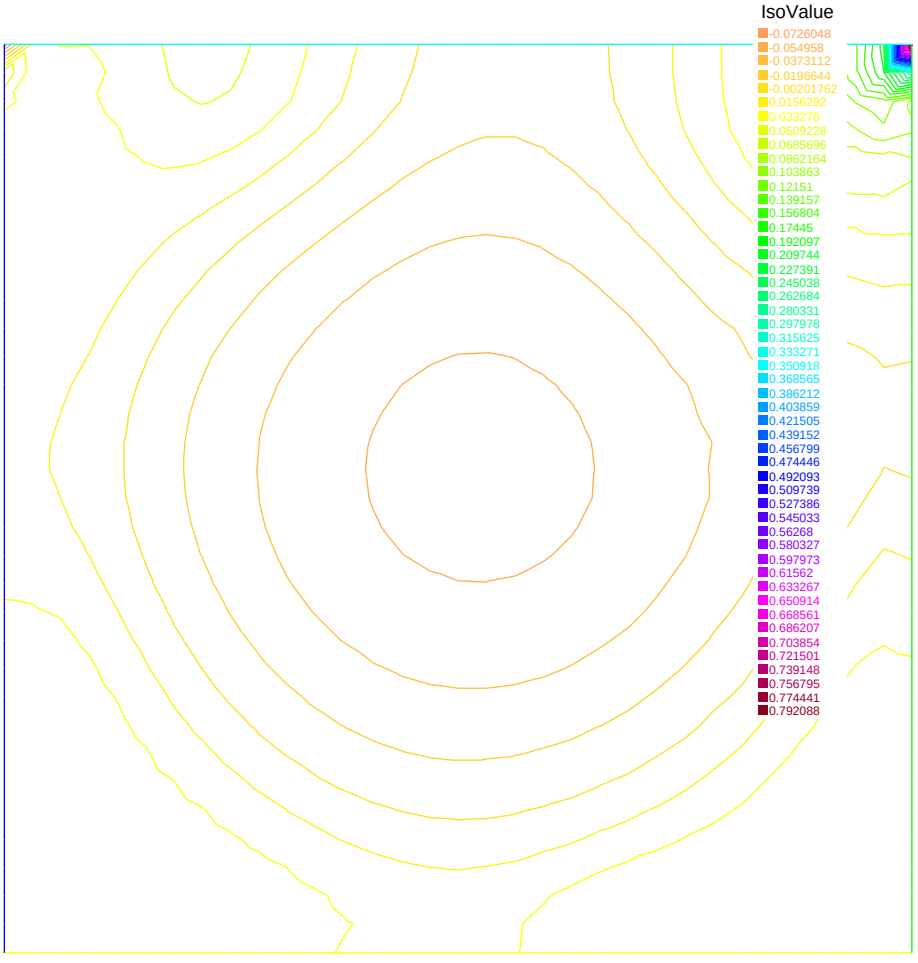


Fig. 5.13. Pressure level lines for $Re = 3200$, $h = 1/32$.

The numerical examples are broadly divided into two parts. The first part presents the rates of convergence. The second part deals with the problem of nonlinear steady flow. We will compare CPU time for the common VMS method and our method to support the discussions in the former sections, and also show the stability and efficiency of our method.

5.1. Rates of convergence study

We consider Ω as the unit square in R^2. The uniform mesh is obtained by dividing Ω into squares and then drawing a diagonal in each square in the same direction.

We choose the exact solution $(u = (u_1, u_2), p)$ as follows:

$$u_1 = 10x^2(x-1)^2y(y-1)(2y-1),$$
$$u_2 = -10x(x-1)(2x-1)y^2(y-1)^2,$$
$$p = 10(2x-1)(2y-1),$$

where f is determined by (2.1) with $\nu = 1.0/Re$.

For the Taylor-Hood elements, using classical Galerkin method, the theory predicts a convergence rate of $O(h^2)$ in the energy norm, $O(h^3)$ in the L^2 norm for the velocity, and $O(h^2)$ for the pressure. To keep the convergence rates, we set $\alpha = O(h^2)$. Here, we choose $\alpha = 0.1\ h^2$. Firstly, we consider the case of low Reynolds number $Re = 100$, for which the standard Galerkin method gets good stability. We show in Fig. 5.1 that for this case common VMS method and our method work well and keep the convergence rates just like the theoretical analysis.

When the Reynolds number is up to 10,000, the standard Galerkin method is unstable. However, from Tables 5.1 and 5.2, we can see that both common VMS method and our method convergence as h decreases, and both methods keep the convergence rates. Also, we can see that our method doesn't need additional iteration compared with the common VMS method.

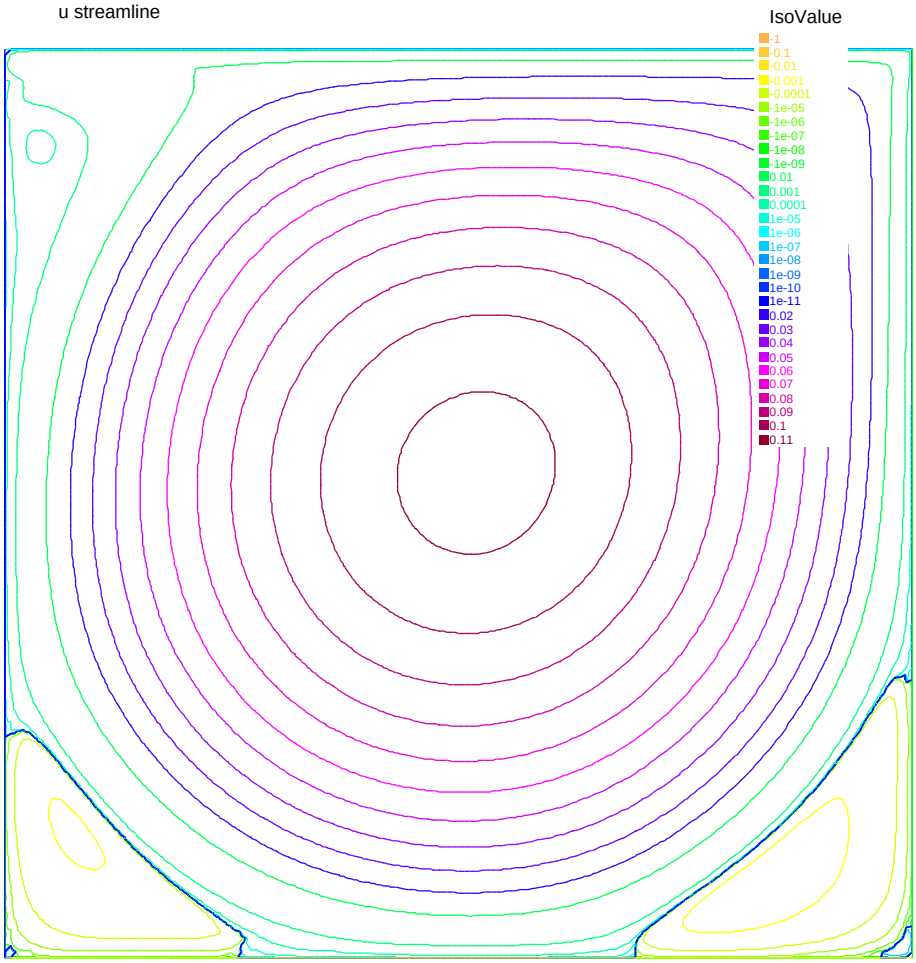


Fig. 5.14. Velocity streamlines for $Re = 5000$, $h = 1/48$.

Besides, we also confirm the equivalence between our method and common VMS method. Concerning with the CPU time for both methods, we find that our method is less time-consuming than common VMS method, actually, it nearly save half the CPU time.

5.2. The driven cavity flow

A popular benchmark problem for testing numerical schemes is the'lid driven cavity'. This problem is chosen because some benchmark data is available for comparison. In this problem, computations are carried out in the domain $\Omega = [0,1] \times [0,1]$. Flow is driven by the tangential velocity field applied to the top boundary in the absence of other body forces. On the top side $\{(x,1) : 0 < x < 1\}$, the velocity is equal to $u = (1,0)$, on the rest of the boundary, zero Dirichlet conditions are imposed. The computational results based on our method and common VMS method for a set of different higher Reynolds numbers $(Re = 1000, 3200, 5000, 10,000)$ are shown in Figs. 5.2–5.9 compared with the results obtained by Ghia et al. [34]. Here we choose $\alpha = 0.1$ h. In particular, we draw the x component of velocity along the vertical centerline and y component of velocity along the horizontal centerlines. Ghia et al.s algorithm is based on the time dependent stream function using the coupled implicit and multigrid methods. The present numerical simulations are computed on a mesh $(h = 1/24, 1/32, 1/48, 1/64)$ and they are compared to the fairly finer mesh $(h = 1/129)$ of [34]. Good agreement with the benchmark data of Ghia et al. [34] verify the two methods. Moreover, these figures show the equivalence of the two methods computationally.

To show the effectiveness of our stabilization method based on two local Gauss integrations compared with the common VMS method, we give the CPU time in Table 5.3 based on above tests. From this table, we can see that our method will save more CPU time as h decreases and for more higher Reynolds number.

Besides, to show the stability of our method, we present the streamlines and the pressure contours of the cavity flows at different Reynolds numbers in Figs. 5.10–5.17. As we know, the position of the main vortex moves towards the center of the

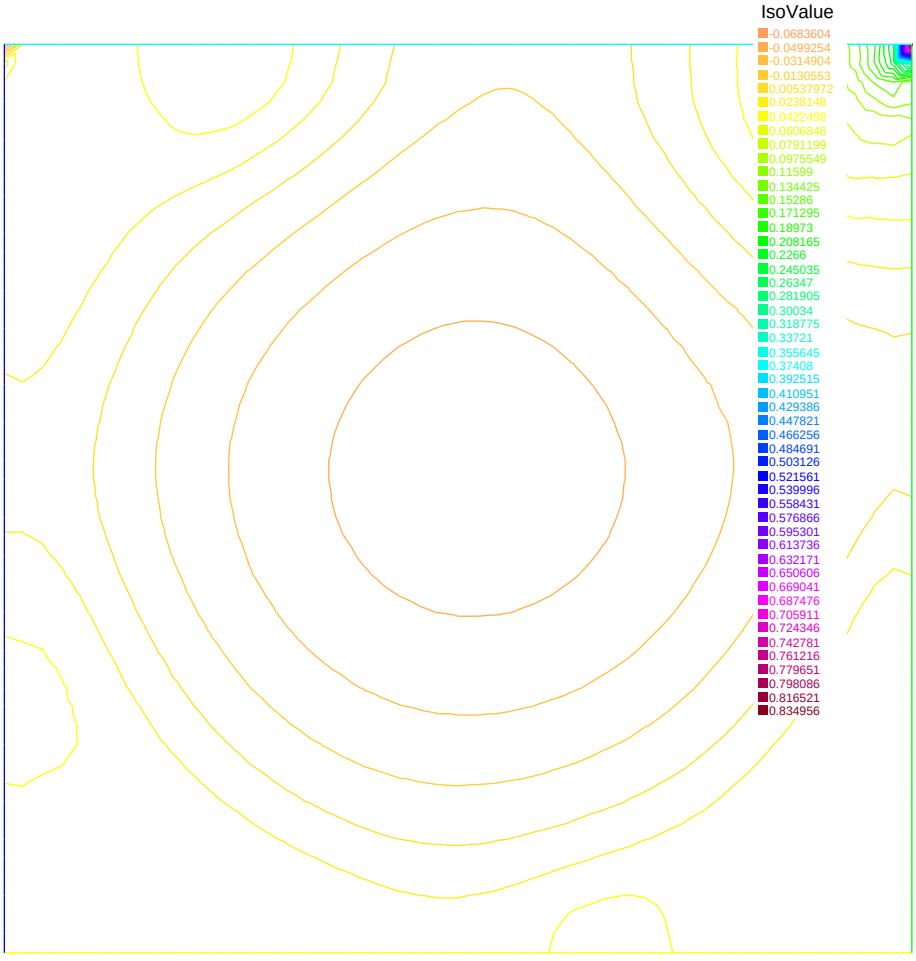
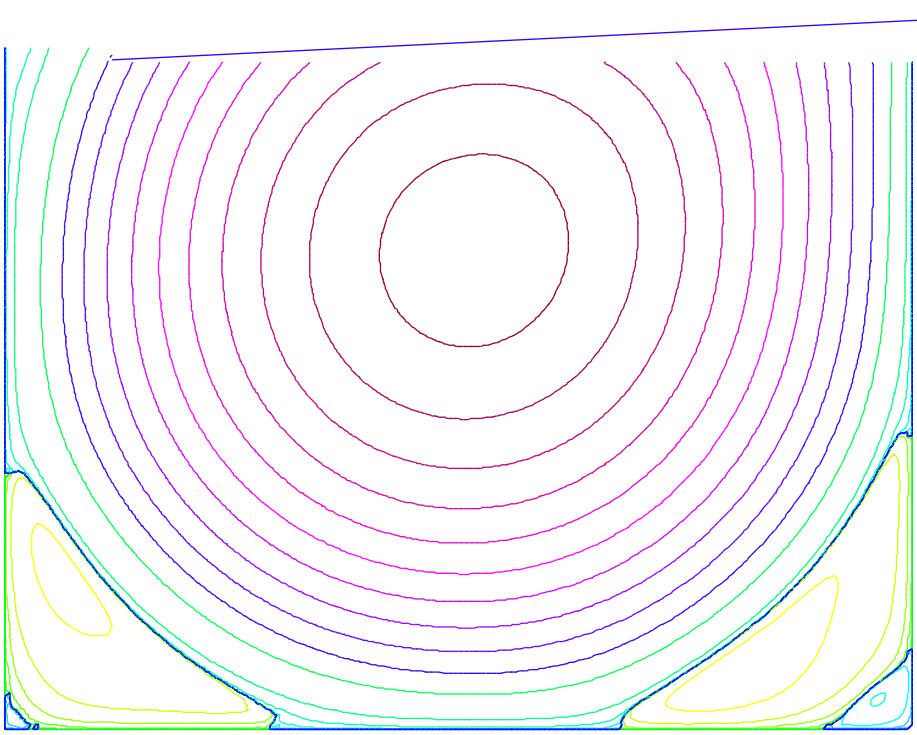


Fig. 5.15. Pressure level lines for $Re = 5000$, $h = 1/48$.



cavity when the Reynolds number increases, and additional second vortex may appear in the right bottom corner of the cavity and a third vortex appear at the lower left corner. When Reynolds number is up to 10,000, five resolved vortices are captured, this result fits to the result of Ghia et al. [34]. The streamlines and pressure contours agree with the results in [28].

5.3. The backward facing step flow

To show stability and efficiency of our method, we test another benchmark problem, the backward facing step problem, which is known to possess a corner singularity.

The geometry and the boundary conditions are shown in Fig. 5.18 as in [24]. For this problem, at upper and lower computational boundaries and at the inflow section, a uniform free-stream velocity boundary conditions is imposed. The Reynolds number is 150, which is based on the maximum inlet velocity $v_{x\max} = 1$ and the height of the inlet. Three different meshes which compose of 250, 960 and 2140 triangle elements respectively are employed. Figs. 5.19, 5.20 show the vector field and the contour plots of the pressure, respectively. Although, for the coarsest mesh, our method shows more oscillatory around the corner. But, as h decreased, we note that the pressure contours are presented with less oscillatory. All these results agree with those of [24] and [27].

6. Conclusion

In this article, we presented and discussed a finite element algorithm for VMS method in solving the steady Navier–Stokes equations based on two local Gauss integrations. The main feature of our method is using two local Gauss integrations to replace the projection operator without adding any variables. And this method is equivalent to common VMS method for

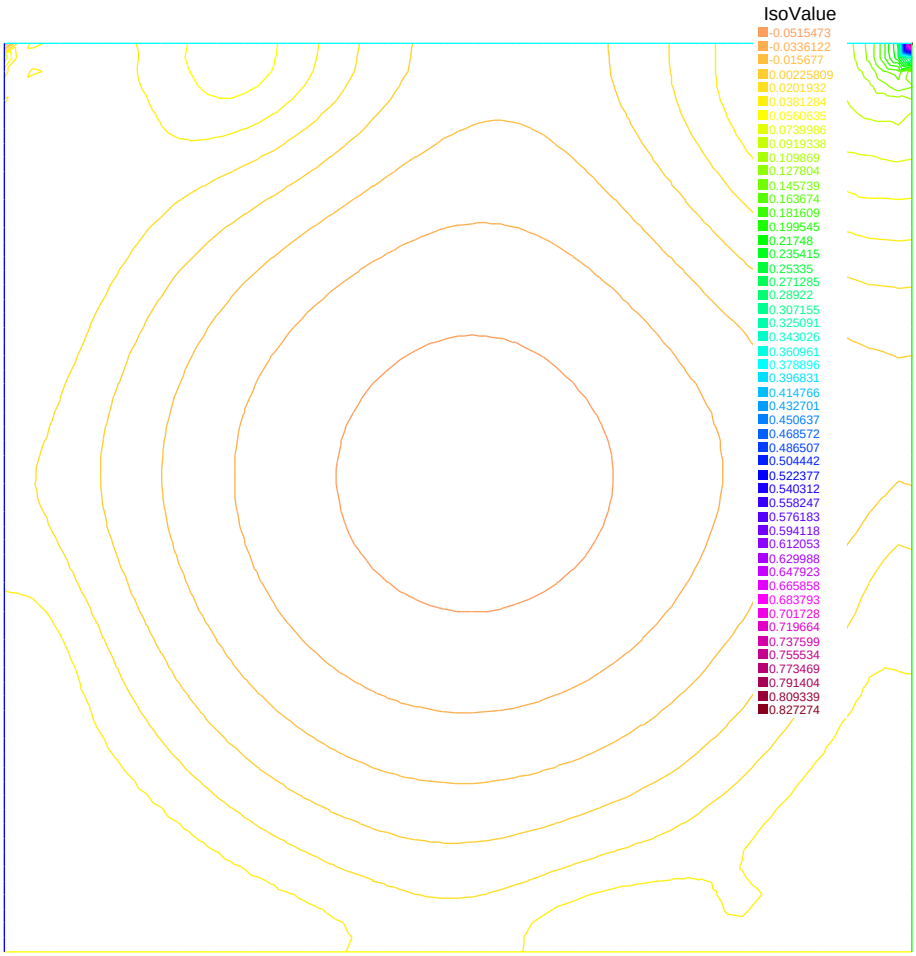


Fig. 5.17. Pressure level lines for $Re = 10,000$, $h = 1/64$.

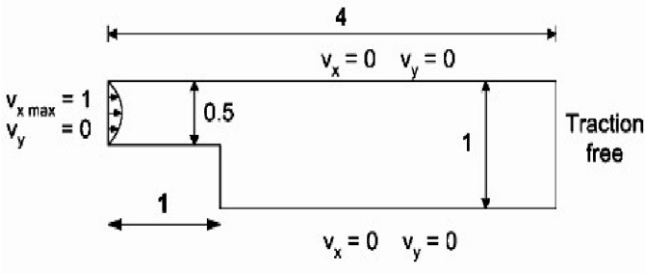


Fig. 5.18. Problem description of the backward facing step.

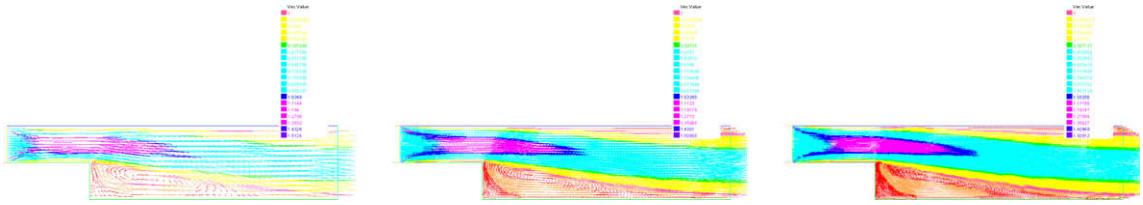


Fig. 5.19. The velocity vector fields with three different meshes. From left to right: vector filed on 250, 960 and 2140 triangle elements respectively.

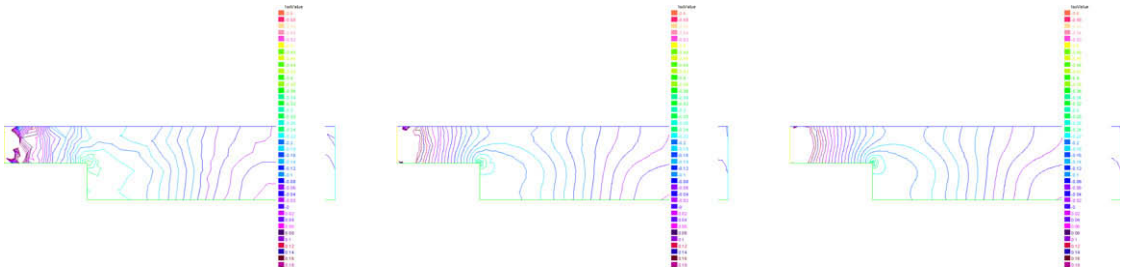


Fig. 5.20. The pressure fields with three different meshes. From left to right: pressure contours on 250, 960 and 2140 triangle elements respectively.

the Taylor–Hood elements but saves a lot of CPU time. There are many open questions including the possible extension of the method to time dependent problems, numerical analysis and more extensive testing.

Acknowledgment

The authors would like to thank the referees for their comments that lead to great improvements in content.

References

[1] V. Girault, P.-A. Raviart, Finite Element Methods for the Navier–Stokes Equations: Theory and Algorithms, Springer Ser. Comput. Math., vol. 5, Springer, Berlin, 1986.
[2] M. Gunzburger, Finite Element Methods for Viscous Incompressible Flows: A Guide to Theory, Practice and Algorithms, Academic Press, Boston, 1989.
[3] M. Gunzburger, R. Nicolaides, Incompressible Computational Fluid Dynamics: Trends and Advances, Cambridge University Press, Cambridge, New York, 1993.
[4] L.P. Franca, S.L. Frey, A.L. Madureira, Two- and three-dimensional simulations of the incompressible Navier–Stokes equations based on stabilized methods, Computational Fluid Dynamics, vol. 94, John Wiley and Sons Ltd., 1994, pp. 121–128.
[5] L.P. Franca, T.J.R. Hughes, Convergence analyses of Galerkin least-squares methods for symmetric advective–diffusive forms of the Stokes and incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 105 (1993) 285–298.
[6] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics. VIII. The Galerkin/least-squares method for advective–diffusive equations, Comput. Methods Appl. Mech. Eng. 73 (1989) 173–189.
[7] V. John, Large Eddy Simulation of Turbulent Incompressible Flows, Analytical and Numerical Results for a Class of LES Models, Lecture Notes in Computational Science and Engineering, vol. 34, Springer, Berlin, Heidelberg, New York , 2004.
[8] P. Sagaut, Large Eddy Simulation for Incompressible Flows, second ed., Springer, Berlin, Heidelberg, New York, 2003.
[9] R. Becker, M. Braack, A two-level stabilization scheme for the Navier–Stokes equations, in: M. Feistauer et al. (Eds.), Numerical Mathematics and Advanced Applications, ENUMATH 2003, Springer-Verlag, Berlin, 2004, pp. 123–130.
[10] J.-L. Guermond, Stabilization of Galerkin approximations of transport equations by subgrid modeling, M2AN Math. Model. Numer. Anal. 33 (1999) 1293–1316.
[11] T.J.R. Hughes, L. Mazzei, A.A. Oberai, The multiscale formulation of large eddy simulation: decay of homogeneous isotropic turbulence, Phys. Fluids 13 (2001) 505–511.
[12] T. Hughes, L. Mazzei, K. Jansen, Large eddy simulation and the variational multiscale method, Comput. Vis. Sci. 3 (2000) 47–59.
[13] T.J.R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid-scale models, bubbles and the origins of stabilized methods, Comput. Methods Appl. Mech. Eng. 127 (1995) 387–401.
[14] W. Layton, A connection between subgrid scale eddy viscosity and mixed methods, Appl. Math. Comput. 133 (2002) 147–157.
[15] S. Kaya, B. Riviere, A twogrid stabilization method for solving the steadystate Navier–Stokes equations, Numer. Methods PDEs Equat. (3) (2006) 728–743.
[16] S. Kaya, W. layton, B. Riviere, Subgrid stabilized defect correction methods for the Navier–Stokes equations, SIAM Numer. Anal. 44 (2006) 1639–1654.
[17] V. John, S. Kaya, Finite Element Error Analysis of a Variational Multiscale Method for the Navier–Stokes Equations, preprint, Otto-von-Guericke Universitat Magdeburg, Fakultat fur Mathematik, 2004.
[18] V. John, S. Kaya, A finite element variational multiscale method for the Navier–Stokes equations, SIAM J. Sci. Comput. 26 (2005) 1485–1503.
[19] V. John, S. Kaya, W. Layton, A two-level variational multiscale method for convection-dominated convection–diffusion equations, Comput. Methods Appl. Mech. Eng. 195 (2006) 4594–4603.
[20] U. Piomelli, Large-eddy simulation: achievements and challenges, Prog. Aerospace Sci. 35 (1999) 335–362.
[21] C. Meneveau, J. Katz, Scale-invariance and turbulence models for large-eddy simulation, Annu. Rev. Fluid Mech. 32 (2000) 1–32.
[22] E. Hylin, J. McDonough, Chaotic small-scale velocity fields as prospective models for unresolved turbulence in an additive decomposition of the Navier–Stokes equations, Int. J. Fluid Mech. Res. 26 (1999) 539–567.
[23] V. John, Reference values for drag and lift of a two-dimensional time dependent flow around a cylinder, Int. J. Numer. Methods Fluids 44 (2004) 777–788.
[24] A. Masud, R.A. Khurram, A multiscale finite element method for the incompressible Navier–Stokes equations, Comput. Meth. Appl. Mech. Eng. 195 (2006) 1750–1777.
[25] F. Brezzi, A. Russo, Choosing bubbles for advection–diffusion problems, Math. Models Methods Appl. Sci. 4 (1994) 571–587.
[26] L.P. Franca, A. Russo, Deriving upwinding, mass lumping and selective reduced integration by residual-free bubbles, Appl. Math. Lett. 9 (1996) 83–88.
[27] L.P. Franca, A. Nesliturk, On a two-level finite element method for the incompressible Navier–Stokes equations, Int. J. Numer. Methods Eng. 52 (2001) 433–453.
[28] V. Gravemeier, W.A. Wall, E. Ramm, A three-level finite element method for the instationary incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 193 (2004) 1323–1366.
[29] M. Braack, E. Burman, Local projection stabilization for the Oseen problem and its interpretation as a variational multiscale method, SIAM J. Numer. Anal. 43 (2006) 2544–2566.
[30] S.S. Collis, Monitoring unresolved scales in multiscale turbulence modeling, Phys. Fluids 13 (2001) 1800–1806.
[31] J. Li, Y.N. He, A stabilized finite element method based on two local Gauss integrations for the Stokes equations, J. Comp. Appl. Math. 214 (2008) 58–65.

[32] Y.N. He, J. Li, A stabilized finite element method based on local polynomial pressure projection for the stationary Navier–Stokes equations, Appl. Numer. Math. 58 (2008) 1503–1514.
[34] U. Ghia, K.N. Ghia, C.T. Shin, High-resolutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[35] FREEFEM++, version 2.19.1, <http://www.freefem.org/>.