Universitat Politècnica de Catalunya
ETSECCPB

Treball Fi de Màster

Explicit variational multiscale methods for incompressible turbulent flow

by

Liang Yang

Advisor: Santiago Badia

Barcelona, June 2010

‘‘I am an old man now, and when I die and go to
heaven there are two matters on which I hope for
enlightenment. One is quantum electrodynamics,
and the other is the turbulent motion of fluids.
And about the former I am rather optimistic”
–Horace Lamb

Abstract

Explicit variational multiscale methods for incompressible turbulent flow

Liang Yang

An explicit finite element scheme for the simulation of incompressible turbulent flows is presented in this thesis. It consists variational multiscale (VMS) method in which the sub-grid model is dynamic and the resulting sub-grid component is enforced to be orthogonal to the finite element solution. These two particular features of our formulation are basic for the explicit treatment in time. In particular, we have considered the artificial compressibility method for the explicit enforcement of incompressibility. The multiscale treatment of the problem using VMS and the sub-grid model suggested have three positive numerical consequences. The method can be understood as a residual-based stabilized finite element method, and so, the singularly perturbed nature of the Navier-Stokes equations is stabilized as the Reynolds number increases. On the other hand, it allows to avoid the compatibility conditions (the so-called inf-sup) between velocity and pressure finite element spaces, and so, we are able to use equal interpolation. Finally, the modelled sub-grid component acts as a turbulence model in the problem for the finite element component. The turbulence modelling inherent to our multi-scale approach is fully motivated by numerical arguments and can be understood as a large eddy simulation (LES) model. Since we are considering dynamic subscales, the dynamic LES model allows backscatter. In order to show the effect of the turbulent model, the lid driven cavity flow, the plane mixing layer and isotropic turbulence both in two dimensions and three dimensions are studied, obtaining the expected results. All these problems have been solved with equal order velocity-pressure approximation and linear finite elements.

Keywords: Explicit time integration, Artificial compressibility method, Variational multiscale method, Orthogonal subgird scale, Dynamic subgrid scale, Turbulence

Acknowledgments

First and foremost, I would like to thank my thesis advisor, Prof. Santiago Badia, for sharing his knowledge, comments, suggestions and unconditional support during this investigation.

Additionally, I want to express my gratitude to Erasmus Mundus master program for providing me an excellent environment and appropriate funding to complete this thesis. Also, I would like to take this opportunity to thank the International Center for Numerical Methods in Engineering (CIMNE), Laboratori de Càlcul Numèric (LaCàN) of Universitat Politècnica de Catalunya (UPC), and Civil and Computational Engineering Centre of Swansea University for letting me participate and learn.

Finally, I would like to thank my parents and friends for their support and motivation.

Contents

List of Figures

List of Algorithms

Chapter 1

Introduction

1.1 Introduction to turbulence

"*Turbulence is the most important unsolved problem of classical physics* "

–*Richard Feynman*

Most of the flows in the world are turbulent. Examples range from smoke rising from a cigarette to flood water, an aircraft passing through the air to the recent NASA's sun image (See Fig 1.1). The problem of turbulence has been studied by many great scientists and engineers in the past. Beginning with its recognition by da Vinci and jumping to the works of Boussinesq and Reynolds in the 19^{th}century, the interest has continued through the 20^{th} century, with the works of Prandtl, Taylor, Kolmogorov and many others. Till date it is one of the key research field in Computational Fluid Dynamics (CFD).

Before plunging into the computational aspects of the turbulence, it is worthwhile to first discuss physical aspects of the phenomenon.

1.1.1 General properties of turbulence

Von Karman (1937) defined the turbulence:

"*Turbulence is an irregular motion which in general makes its appearance in fluids,*

(a) cigarrete

(b) airplane

(c) water

(d) sun

Figure 1.1: Turbulence in nature

gaseous or liquid, when they flow past solid surfaces or even when neighboring streams of the same fluids flow past or over one another."

(Wilcox, 2006) concludes the following features of turbulence:

• **Instability and Nonlinearity**. Analysis of solutions to the Navier-Stokes(NS) equation shows that turbulence develops as an instability of laminar flow. Mathematically speaking, the instability result mainly from interaction between the NS equation's nonlinear inertial terms and viscous terms. The interaction is very complex because it it rotational, fully dimensional and time dependent.

- **Statistical Aspects** Turbulence is characterized by random fluctuations thus mandating the use of statistical methods to analyze it.

- **Turbulence is a continuum phenomenon** The time-dependent, three-dimensional continuity and NS equations contain all of the physics of a given turbulent flow. As noted by Tennekes and Lumley (1983)

''*even the smallest scale occurring in a turbulent flow are ordinarily far larger than any molecular length scale.*''

- **Vortex stretching** The vorticity in a turbulent flow is itself three dimensional so that vortex lines in the flow are nonparallel. Vortex stretching is absent in two-dimensional flows so that turbulence must be three dimensional.

- **Turbulence scales and the cascade** Turbulence consists of a continuous spectrum of scales ranging from largest to smallest.

- **Large eddies and turbulent mixing**

- **Enhanced diffusivity** The turbulence enhances diffusivity. Turbulent diffusion greatly enhances the transfer of mass, momentum and energy. The stresses in turbulent flows are often several orders of magnitude larger than in corresponding laminar flows.

1.1.2 Spectral representation and the Kolmogorov $-5/3$ law

Since turbulence contains a continuous spectrum of scales, it is often convenient to do analysis in terms of the spectrum distribution of energy. In general, a spectral representation is a Fourier decomposition into wavenumbers, k.

Kolomogorov's (1941) idea is that a range of wave numbers k exists in which the energy transferred by inertial effects dominates.

A typical energy spectrum for turbulent flow is shown in Fig 1.2

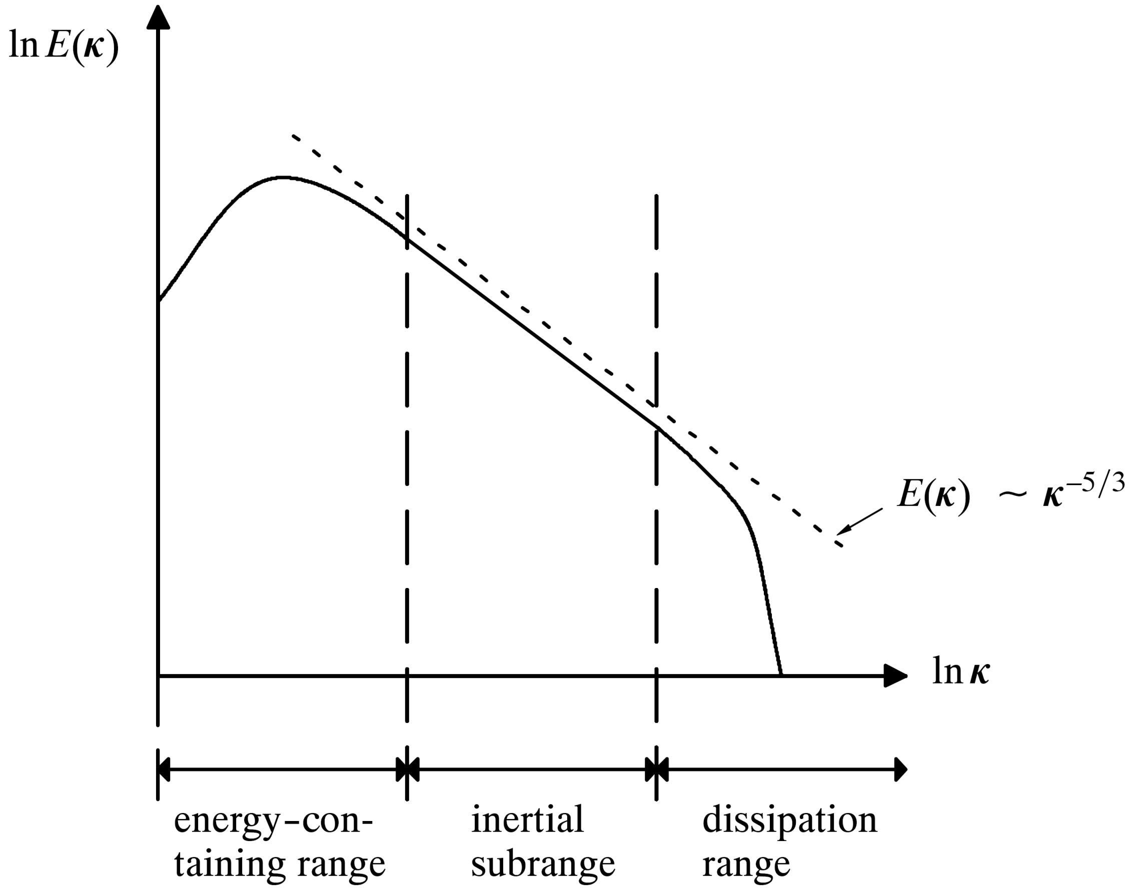


Figure 1.2: Energy spectrum for the turbulent flow

1.2 Popular approaches

Computers have opened a door to the understanding of turbulence. The typical approaches to the simulation of turbulence are:

•**Direct Numerical Simulation (DNS)**

Direct Numerical Simulation (DNS) of Navier-Stokes equations is a brute force approach that solves turbulence. When dealing with turbulent flow, this method tries to resolve all turbulent phenomena at all length and time scales simply by numerically solving Navier-Stokes (NS) equation. The number of degrees of freedom scales increases with Re^3. Due to this reason, the computational costs of high Reynolds number flow simulations exceeds the limits of computing power today.

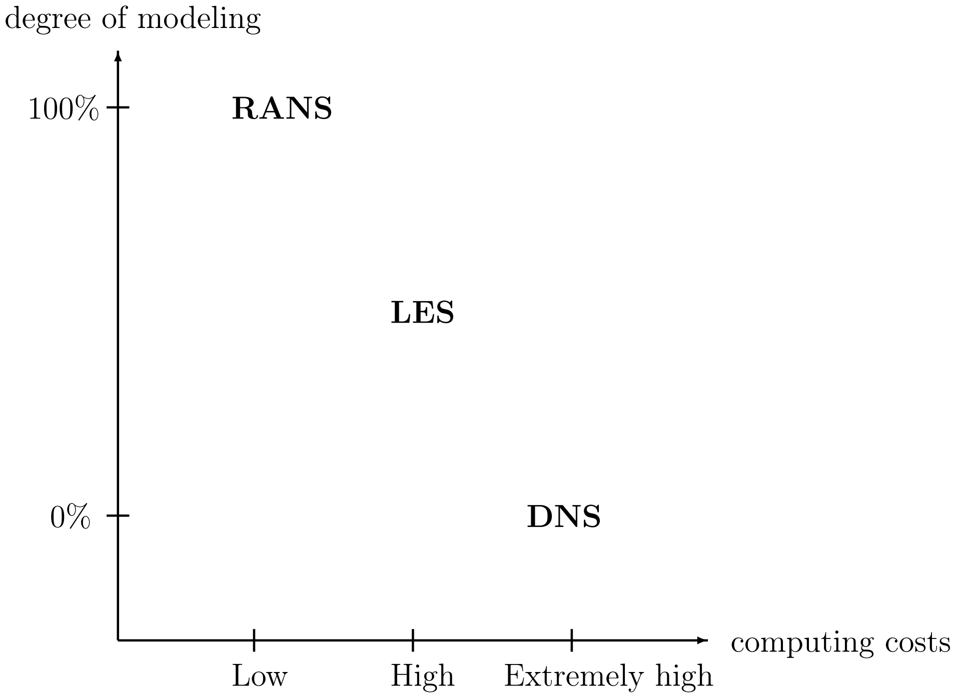
•Large Eddy Simulation (LES)

Usually DNS of a turbulent flow is not feasible. Large eddy simulation, a promising approach, is resolving the large vortexes (eddies) and modeling the effect of the smaller scales. Hence LES is much more economical in term of computational power in comparison to DNS. The velocity and pressure is split in two parts. These quantities are the sum of a mean component and a fluctuation component as follows. It solves the mean component with a numerical method and solves the fluctuations analytically (depending on the turbulent models). Physical modeling and physical parameters are vital parts for the LES simulation.

•Reynolds Average Navier-Stokes (RANS)

In this approach, suggested by Reynolds (1895), quantities like velocity, pressure etc, are decomposed into an ensemble mean flow and a fluctuating perturbation field. That means, this approach only provides mean turbulent quantities, which is the cheapest among these three.

Fig.1.2 illustrates the relations between computational costs and degree of modeling among DNS, LES and RANS.



1.3 FEM in flow problem and its difficulties

The finite element method (FEM) can be traced back to the 1950s in the aircraft industry, and it is popular for structural analysis. In early 1970s, an attempt was made to use it in fluid dynamics. In recent years, the FEM in fluids is becoming a popular research field.

The variational multiscale method for turbulent flow is introduced by (Hughes et al., 2000), the concept of orthogonal subgrid scales by (Codina, 2002) and dynamic subgrid scales by (Codina et al., 2007).

1.3.1 Large reynolds number

The main difficulty is due to the convection term. A very fine mesh is needed, which undermines the practical utility of the method. From a computational point of view, convection operators are non-symmetric, which causes oscillations and can only be removed by refining the mesh. From a physical point of view, an extremely fine mesh is needed to capture the smallest dissipative scales.

1.3.2 Inf-sup condition

Considering the steady state problem, the incompressibility leads to a saddle problem. The continuous problem is well-posed due to continuous inf-sup conditions. In order for the discrete problem to be well-posed, it must also satisfy a discrete inf-sup condition. This condition restrict the choice of the velocity-pressure FEM spaces. For instance, the equal order velocity-pressure approximation does not satisfy the inf-sup condition. In order to avoid the need to satisfy the inf-sup condition, we can rely on stabilized methods for the incompressibility.

Finding a stabilization technique could be interpreted to find a good sub-scale reconstruction. The basic idea of this framework is to introduce a decomposition of the velocity space into a finite element space and a subscales space. The subgrid component is modeled and can be motivated by Fourier analysis arguments, and then substituted into the large scale equation.

1.3.3 Implicit pressure

Computational costs are always a great issue in CFD. The implicit treatment of the pressure implies solving the pressure Laplacian equation, which usually costs 70% of the overall computational time. A fast fully explicit method is an interest approach.

1.4 Aim and Ambition

Introducing an efficient explicit approach using the variational multiscale finite element methods for the simulation of turbulent flow is a new, challenging and highly valuable topic. First, the explicit method is fast, better for parallel computing and able to exhibit rapid transient behavior. Second, in the variational multiscale method framework, no turbulence model is needed and turbulence is fully motivated by the numerical method.

1.5 Overview of the thesis

At the end of this introduction, the following short overview should procure an impression of what will be dealt with in the respective chapters of this work. The thesis is organized as follows.

Chapter 2 is dedicated to present the stabilization of the convection-diffusion problem and the introduction of the variational multiscale method. Both orthogonal sub-scales (OSS) and algebraic sub-grid scale methods (ASGS) are considered and compared.

Chapter 3 provides an introduction to artificial compressibility method for the Stokes problem: explicit treatment of the pressure is studied in this chapter. In the lid-driven cavity, the comparison based on convergence behavior is given.

Chapter 4 presents the Stokes problem and a discussion about the LBB compatibility condition. The classical stabilization method and variational multiscale method which avoid the LBB compatibility condition are studied.

Chapter 5 introduces the explicit variational multiscale method for incompressible flow, the proposed algorithms and their implementation.

Chapter 6 contains various numerical examples for laminar and turbulent flow, including the lid driven cavity flow both in 2D and 3D, the plane mixing layer in 2D and decaying of isotropic turbulence in 2D and 3D.

Chapter 2

Convection-Diffusion Problem

This chapter first illustrates the deficiencies of the standard Galerkin formulation in the convection-dominated problems and presents the SUPG method designed to produce stable and accurate results in the presence of highly convective effects. Then, it introduces the variational multiscale method, which consists of taking as space for the sub-scales the orthogonal to the finite element space, and finally, compares these two sub-scale models (OSS and ASGS).

2.1 Standard Galerkin method and its stabilization

As an introduction to the convection-diffusion problems by means of the generalized Galerkin methods, we start illustrating the basic steps in the formulation of the standard Galerkin finite element method. This will point out the deficiency of the classical Galerkin approach in solving the convection-dominated problem.

2.1.1 Problem statement

First we introduce some notations which are used in this chapter.
Let us consider the transport by convection and diffusion of a scalar quantity $\boldsymbol{u}$ domain $\boldsymbol{u} \subset \mathbb{R}^d$ where $d = 2$ or 3,with boundary $\Gamma = \partial\Omega$.
Strong form:

$$\begin{aligned} \partial_t \boldsymbol{u} + \boldsymbol{a} \cdot \nabla \boldsymbol{u} - \nu \nabla \cdot (\nabla \boldsymbol{u}) = \boldsymbol{f} &\quad \text{in}\, \Omega, t \in]0, T[\\ \boldsymbol{u} = \boldsymbol{u}_D &\quad \text{on}\, \partial\Omega_D, t \in]0, T[\end{aligned} \tag{2.1}$$

In order to obtain the weak form of problem(2.1) we need to introduce some notations.

We denote by $L^p(\Omega)$, $1 \leq p < \infty$. The space of real functions defined on Ω with the p power absolutely integrable with respect to the Lebesgue measure. The space $L^\infty(\Omega)$ consists of essentially bounded functions in Ω. The case $p = 2$ is of special interests. $L^2(\Omega)$ is a Hilbert space endowed with the scalar product (u, v) and its induced norm $\|u\|_0$. The Hilbert space $H^m(\Omega)$ is the space of functions in $L^2(\Omega)$ whose weak derivatives of order less than or equal to $L^2(\Omega)$, m being an integer and $1 \leq p < \infty$. This space is endowed with scalar product and its associated norm $\|\cdot\|_m$. Furthermore, we denote by $H_0^1(\Omega)$ the space of functions of $H^1(\Omega)$ vanishing on Γ and by $H^{-1}(\Omega)$ its dual space. In general, duality pairing will be denoted with the symbol $\langle \cdot, \cdot \rangle$. Given a function $u \in H_0^1(\Omega)$ the Poincaré inequality:

$$\|\boldsymbol{u}\|_1^2 \leq C_\Omega \|\nabla u\|_0^2 \tag{2.2}$$

holds for Ω with Lipschitz continuous boundary. Therefore, the seminorm $|u|_1 = \|\nabla u\|_0^2$ in $H^1(\Omega)$ is a norm in $H_0^1(\Omega)$.

Let $\mathcal{V} \equiv H^1(\Omega)$,$\mathcal{V}_0 \equiv H_0^1(\Omega)$ denote the real Hilbert spaces for velocity, with associated norms $\|\boldsymbol{v}\|_v$, and let $\boldsymbol{f} \in \mathcal{V}^{-1}$.

We shall often consider d-dimensional vector functions with components in one of these spaces. We shall indicate them by boldface letters, for instance $\boldsymbol{H}^m(\Omega) = (H^m(\Omega))^d$. In the following, we will not distinguish between scalar products or norms for scalar or vector-valued functions.

The weak formulation of the convection-diffusion problem then takes the following form:

$$(\partial_t \boldsymbol{u}, \boldsymbol{v}) + \nu \left(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}\right) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \rangle = \langle \boldsymbol{f}, \boldsymbol{v} \rangle \tag{2.3}$$

2.1.2 Standard Galerkin formulation

We now have to discretize the convection-diffusion problem by means of the Galerkin finite element method:

$$(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle = \langle \boldsymbol{f}, \boldsymbol{v}_h \rangle \tag{2.4}$$

Algorithm 2.1 *Explicit* Standard Galerkin algorithm for the convection-diffusion problem

```
read $\boldsymbol{u}_h^0$
while time integration do
    for element do
        velocity $\boldsymbol{u}^n$ at cell nodes
        velocity gradient $\nabla \boldsymbol{u}^n$ at cell nodes
        velocity residual $R = \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \nu \left(\nabla \boldsymbol{u}^n, \nabla \boldsymbol{v}_h\right) - \langle \boldsymbol{f}, \boldsymbol{v}_h \rangle$
    end for
    update the velocity $\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t R$
    prescribe boundary conditions
end while
```

2.1.3 SUPG formulation

In order to ensure that the solution of the differential equation is also a solution of the weak form, the stabilization term should be a function of the residual of the differential equation. The general form of the stabilization technique is

$$(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \langle R(\boldsymbol{u}_h), \tau P(\boldsymbol{v}_h) \rangle = \langle \boldsymbol{f}, \boldsymbol{v}_h \rangle \tag{2.5}$$

For the SUPG method, the stabilization technique is defined by taking

$$P(\boldsymbol{v}_h) = \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h \tag{2.6}$$

Algorithm 2.2 *Explicit* SUPG algorithm for the convection-diffusion problem

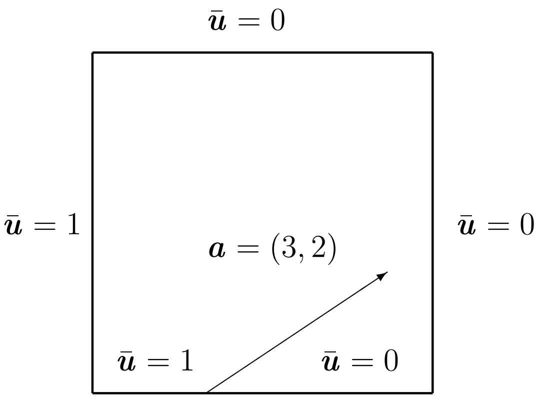
```
read $\boldsymbol{u}_h^0$
while time integration do
    calculate $\tau$
    for element do
        velocity $\boldsymbol{u}^n$ at cell nodes
        velocity gradient $\nabla \boldsymbol{u}^n$ at cell nodes
        velocity residual $R = \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \nu \left(\nabla \boldsymbol{u}^n, \nabla \boldsymbol{v}_h\right) + \tau \left\langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, a \cdot \nabla \boldsymbol{v}_h \right\rangle$
    end for
    update the velocity $\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \Delta t R$
    prescribe boundary conditions
end while
```

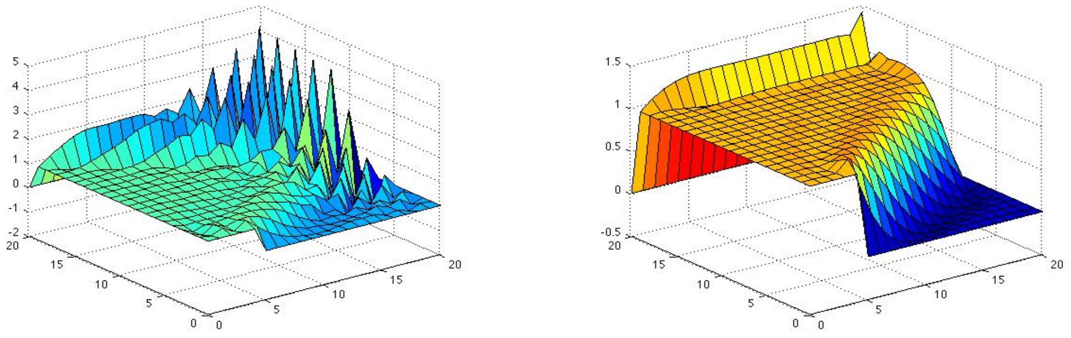
2.1.4 Numerical example

In this section, a test is conducted to show the numerical performance of the Standard Galerkin and SUPG methods. The diffusion coefficient is taken as $k = 10^{-4}$. The finite element mesh employed consists of 20 × 20 Q1 elements. If $\bar{\boldsymbol{u}}$ donates the boundary condition for $\boldsymbol{u}$. The cases are defined and shown in Figure 4.3.

The problem definition:

$\boldsymbol{a} = (3, 2)\,, \bar{\boldsymbol{u}} = 1 \;\text{ on }\; \Gamma_1 = [0, 0.25] \times \{0\}, \bar{\boldsymbol{u}} = 0\,\text{on}\;\; \partial\Omega \backslash \Gamma_1$


$\bar{\boldsymbol{u}} = 0$
$\bar{\boldsymbol{u}} = 1$
$\boldsymbol{a} = (3, 2)$
$\bar{\boldsymbol{u}} = 0$
$\bar{\boldsymbol{u}} = 1$
$\bar{\boldsymbol{u}} = 0$



(a) Galerkin FEM

(b) SUPG

Figure 2.1: Difference between Galerkin FEM and SUPG

We show that the Galerkin finite element method is not suited to solve this problem in 20 × 20 grids. Stabilization is needed in this case.

2.2 Variational multiscale method

The variational multiscale method introduced by Hughes provides the necessary mathematical framework for the construction of so-called sub-grid scale models. It is based on the decomposition of the solution $\boldsymbol{u}$, on a coarse-scale component $\boldsymbol{u}_h$, which can be resolved by the finite element mesh, and a subscale component $\tilde{\boldsymbol{u}}$, which attempts to determine analytically.

2.2.1 Formulation

Consider a scale decomposition of the space $\mathcal{V}$. The multiscale is applied to the weak form of the convection-diffusion equation.
$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \boldsymbol{u}_h \in \mathcal{V}_h, \tilde{\boldsymbol{u}} \in \tilde{\mathcal{V}}$
The problem becomes

$$\left(\partial_t\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \boldsymbol{v}\right)+\nu\left(\nabla\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}\right)+\left\langle\boldsymbol{a} \cdot \nabla\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \boldsymbol{v}\right\rangle=\langle\boldsymbol{f}, \boldsymbol{v}\rangle \tag{2.7}$$

2.2.2 Large-scale problem statement

First, we consider test function $\boldsymbol{v}_h \in \mathcal{V}_h$, the resolved large-scale problem becomes

$$\left(\partial_t\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h\right)+\nu\left(\nabla\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}_h\right)+\left\langle\boldsymbol{a} \cdot \nabla\left(\boldsymbol{u}_h+\tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h\right\rangle=\left\langle\boldsymbol{f}, \boldsymbol{v}_h\right\rangle \tag{2.8}$$

$$\left(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h\right)+\nu\left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right)+\left\langle\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h\right\rangle-\left\langle\tilde{\boldsymbol{u}}, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h\right\rangle=\left\langle\boldsymbol{f}, \boldsymbol{v}_h\right\rangle \tag{2.9}$$

2.2.3 Sub-scale problem statement

Then, we consider $\tilde{\boldsymbol{v}} \in \tilde{\mathcal{V}}$, the unresolved sub-scale problem becomes

$$\partial_t \tilde{\boldsymbol{u}} + \boldsymbol{a} \cdot \nabla \tilde{\boldsymbol{u}} - \nu \Delta \tilde{\boldsymbol{u}} = r_{u,h} \tag{2.10}$$

$$r_{u,h} = -\left(\partial_t \boldsymbol{u}_h + \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h - \nu \Delta \boldsymbol{u}_h - \boldsymbol{f}\right) \tag{2.11}$$

where $r_{u,h}$ is appropriate residual of the finite element components $\boldsymbol{u}_h$ adequately projected onto the space of subscales $\tilde{\mathcal{V}}$. The Projection P can be either the identity or the projection orthogonal to the finite element space. We will refer the choice $P = I$ (identity) as the Algebraic Subgrid-Scale formulation (ASGS), whereas orthogonal projection on the appropriate finite element space, which lead to the so-called Orthogonal Subscales Stabilization (OSS).

2.2.4 Orthogonal projection

If we apply Orthogonal Subscales Stabilization (OSS) to the subscale, the residual simply becomes

$$r_{u,h} = -P\left(\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h - \boldsymbol{f}\right) \tag{2.12}$$

2.2.5 Sub-scale approximation

Using Fourier analysis, we could approximate the subscale, The details will be discussed in section 5.4.2. The following approximation could be motivated

1. Static Subscale

$\tilde{\boldsymbol{u}} = -\tau_0 r_{u,h}$

2. Dynamic Subscale

$\tilde{\boldsymbol{u}}_{n+1} = \frac{\tau_t}{\Delta t}\tilde{\boldsymbol{u}}_n - \tau_t r_{u.h}$

The sub-scale is solved analytically and substituted into the large-scale equation.

Algorithm 2.3 *Explicit* VMS algorithm for the convection-diffusion problem

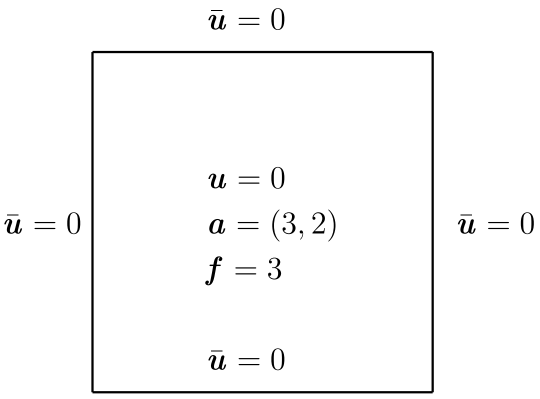
```
read $\boldsymbol{u}_h^0$ and set $\tilde{\boldsymbol{u}} = 0$
while Time integration do
    Calculate the parameters $\tau$
    for Element do
        velocity $\boldsymbol{u}^n$ at cell nodes
        velocity gradient $\nabla \boldsymbol{u}^n$ at cell nodes
        velocity residual $(\boldsymbol{a} \cdot \nabla \boldsymbol{u}^n, \boldsymbol{v}_h) + \nu (\nabla \boldsymbol{u}^n, \nabla \boldsymbol{v}_h) - (\tilde{\boldsymbol{u}}^n, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h) - (f, \boldsymbol{v}_h)$
        contribution to the velocity subscales $\tau_t/\Delta t \boldsymbol{u}^n - \tau_t \boldsymbol{a} \cdot \nabla \boldsymbol{u}^n$
        projection velocity subscales $\boldsymbol{a} \cdot \nabla \boldsymbol{u}^n$(only for OSS)
    end for
    update the velocity
    prescribe boundary conditions
    projection contribution $\tilde{\boldsymbol{u}}^{n+1}$ (only for OSS)
end while
```

2.3 Numerical example

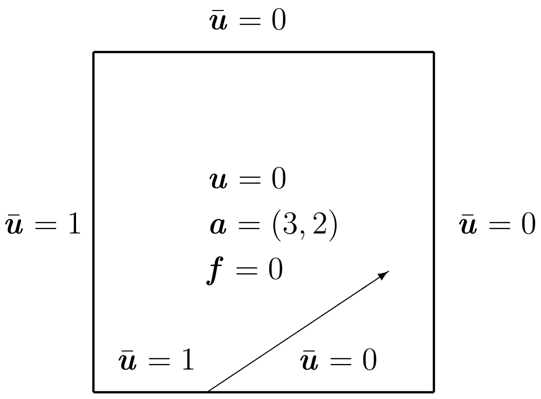
In this section, we test the performance of orthogonal subscale (OSS) and algebraic subscale method (ASGS) using the similar case in (Codina, 2000). The convection-diffusion problem is numerically solved by these method. In the following two cases, the diffusion is taken as $k = 10^{-4}$. The finite element mesh employed consists of 20×20 Q1 elements. $\bar{\boldsymbol{u}}$ donates the boundary condition for $\boldsymbol{u}$. The cases are defined here:

1.$\boldsymbol{f} = 3; \boldsymbol{a} = (3, 2), \bar{\boldsymbol{u}} = 0$

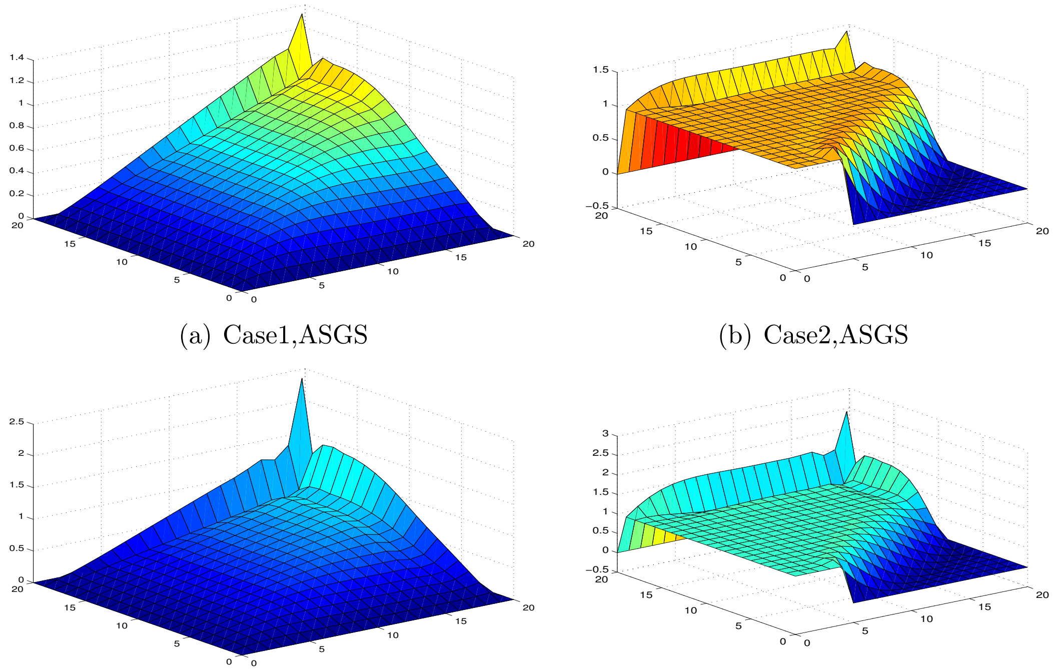
2.$\boldsymbol{f} = 0; \boldsymbol{a} = (3, 2), \bar{\boldsymbol{u}} = 1$ *on* $\Gamma_1 = [0, 0.25] \times \{0\}, \bar{\boldsymbol{u}} = 0$ *on* $\partial\Omega \backslash \Gamma_1$


$\bar{\boldsymbol{u}} = 0$
$\boldsymbol{u} = 0$
$\boldsymbol{a} = (3,2)$
$\boldsymbol{f} = 3$
$\bar{\boldsymbol{u}} = 0$
$\bar{\boldsymbol{u}} = 0$
$\bar{\boldsymbol{u}} = 0$

Case 1


$\bar{\boldsymbol{u}} = 0$
$\boldsymbol{u} = 0$
$\boldsymbol{a} = (3,2)$
$\boldsymbol{f} = 0$
$\bar{\boldsymbol{u}} = 1$
$\bar{\boldsymbol{u}} = 0$
$\bar{\boldsymbol{u}} = 1$
$\bar{\boldsymbol{u}} = 0$

Case 2



(a) Case1,ASGS

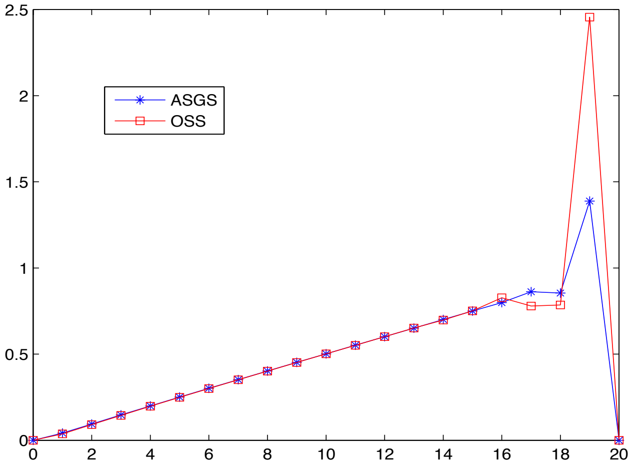
(b) Case2,ASGS

(c) Case1,OSS

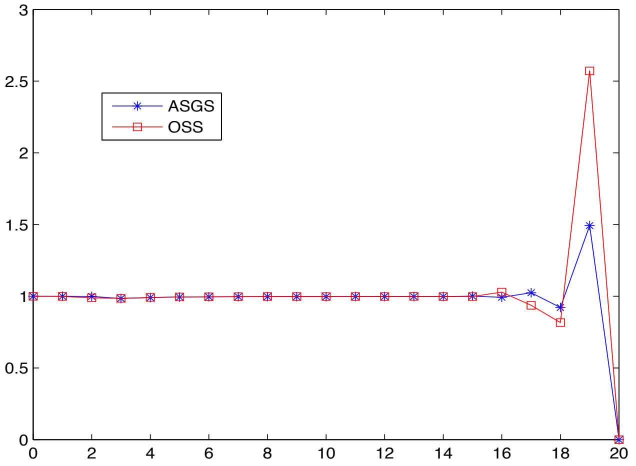
(d) Case2,OSS

Figure 2.2: Difference between OSS and ASGS

The numerical solution for the two cases are shown in Fig.2.2. Both the ASGS and



(a) Case1



(b) Case2

Figure 2.3: Mid-sections plots for the OSS and ASGS methods

the OSS yield very similar solutions in the interior of the computational domain, but the former yields smaller overshoots near the boundary.

To better observe the difference between the solutions obtained using the ASGS and the OSS methods, they are plotted along a mid-section in Fig.2.3. These figures clearly indicate that the ASGS and the OSS formulations only yield different results near the layers.

2.4 Comparison between the static OSS and static ASGS

A static subscale comparison of OSS and ASGS could easily help us to understand it. For convenience, we define the convection term C, stabilization term S and mass term M as

$$C = \langle \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h, \boldsymbol{v}_h \rangle \tag{2.13}$$

$$S = (\boldsymbol{a} \cdot \nabla \boldsymbol{v}_h, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h) \tag{2.14}$$

$$M = (\boldsymbol{v}_h, \boldsymbol{v}_h) \tag{2.15}$$

First, the projection operation P would have the follow property

$$(P, \boldsymbol{v}_h) = (\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h) \tag{2.16}$$

$$\Rightarrow \quad MP = C\boldsymbol{u}_n \tag{2.17}$$

$$\Rightarrow \quad P = M^{-1}C\boldsymbol{u}_h \tag{2.18}$$

Second, project the subscale

$$\begin{aligned} \tilde{\boldsymbol{u}}_h^{OSS} &= \tau r_{r,h}^{OSS} \\ &= \tau(r_{r,h}^{ASGS} - I\left(\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h\right)) \\ &= r_{r,h}^{ASGS} - \tau M^{-1}C\boldsymbol{u}_h \qquad (2.19) \\ &= S\boldsymbol{u}_h - \tau M^{-1}C\boldsymbol{u}_h \qquad (2.20) \end{aligned}$$

In the last, compare the OSS and ASGS

$$\langle \boldsymbol{a} \cdot \nabla \tilde{\boldsymbol{u}}_h, \boldsymbol{v}_h \rangle_{OSS} = \tau \left(S - C^T M^{-1} C \right) \boldsymbol{u}_n \tag{2.21}$$

$$\langle \boldsymbol{a} \cdot \nabla \tilde{\boldsymbol{u}}_h, \boldsymbol{v}_h \rangle_{ASGS} = \tau S \boldsymbol{u}_n \tag{2.22}$$

It is very clear that the OSS takes some diffusion out of the equation. The OSS is less robust than ASGS when dealing with the sharp boundary layers not parallel to the convection velocity.

Chapter 3

Artificial Compressibility Method

The difficulty in computing numerical approximations to the Stokes problem lies in satisfying the divergence-free velocity condition. In this chapter we provide a brief introduction to the theory of Artificial Compressibility (AC) Method for uncoupling the pressure and velocity field, based on a perturbed version of the equation of continuity:

$$\nabla \cdot \boldsymbol{u} = 0 \Rightarrow \partial_t p + \varepsilon \nabla \cdot \boldsymbol{u} = 0 \tag{3.1}$$

where ε is the penalty parameter, which is the square of artificial sound speed.

3.1 What is artificial compressibility method

The artificial compressibility was originally introduced by (Chorin, 1968) with the object of solving steady state incompressible Navier-Stokes equations. With the successful use of the artificial compressibility method for steady problems, it was natural to investigate its applicability to unsteady problems. (Peyret and Taylor, 1983) were some of the first to extend the Artificial Compressibility method to the unsteady incompressible Navier-Stokes equations. Simple explicit time stepping methods would then be applicable.

The ε is a critical value between computational efficiency and mathematical accuracy. On one hand, a reduction of ε will make the system less stiff and permit larger time step in the explicit numerical integration. On the other hand, a certain stiffness of the system (the minimum value of ε) is necessary for a good approximations to the original incompressible system.

The AC schemes reported in the literature can be classified into two categories. The first one is the standard scheme based on the original work by (Chorin, 1968) and the second classification is preconditioned AC schemes. Here we only discuss about the classical AC schemes.

- **Strong form(S)**:

After applied the AC schemes, the transient Stokes equations become

$$\begin{cases} \partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = f \\ \partial_t p + \varepsilon \nabla \cdot \boldsymbol{u} = 0 \end{cases} \tag{3.2}$$

where ε is the square of artificial sound speed.

- **Weak form(W)**:

With the notations introduced in Chapter 2, let $\mathcal{V} \equiv H^1(\Omega)$,$\mathcal{V}_0 \equiv H_0^1(\Omega)$ and $\mathcal{L} \equiv L^2(\Omega)/\mathbb{R}$ denote the real Hilbert spaces for velocity and pressure, with associated norms $\|\boldsymbol{v}\|_v$ and $\|q\|_L$, and let $\boldsymbol{f} \in \mathcal{V}^{-1}$.

$$\begin{cases} (\partial_t \boldsymbol{u}, \boldsymbol{v}) + \nu \left(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}\right) + (\nabla p, \boldsymbol{v}) = \langle \boldsymbol{f}, \boldsymbol{v} \rangle \\ (\partial_t p, q) + (\varepsilon \nabla \cdot \boldsymbol{u}, q) = 0 \end{cases} \tag{3.3}$$

- **Galerkin formulation**:

Consider finite element spaces $\mathcal{V}_h$ and $\mathcal{L}_h$. Such that $\boldsymbol{u}_h \in \mathcal{V}_h, p_h \in \mathcal{L}_h$

$$\begin{cases} (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu\left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) + (\nabla p_h, \boldsymbol{v}_h) = \langle \boldsymbol{f}, \boldsymbol{v}_h \rangle \\ (\partial_t p_h, q_h) + (\varepsilon \nabla \cdot \boldsymbol{u}_h, q_h) = 0 \end{cases} \tag{3.4}$$

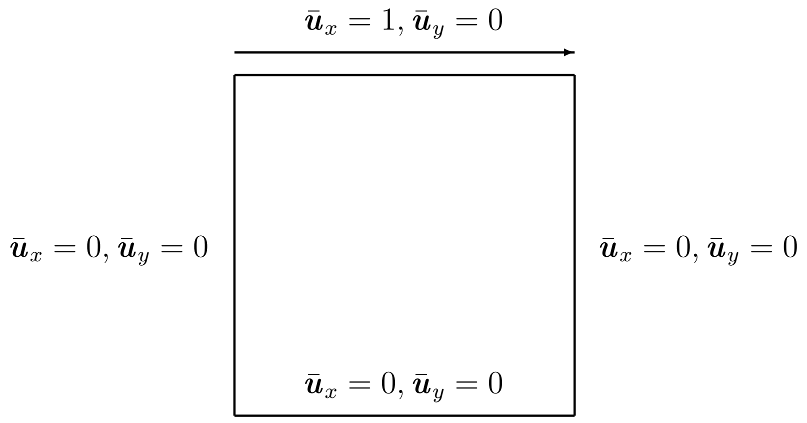
If we define K,M,G_s, respectively, the diffusion, mass, and gradient operator, we can write the Galerkin formulation in matrix form.

$$\begin{bmatrix} M & 0 \\ 0 & \frac{M_P}{\varepsilon} \end{bmatrix} \begin{bmatrix} \dot{U} \\ \dot{P} \end{bmatrix} + \begin{bmatrix} K & G_s \\ G_s^T & 0 \end{bmatrix} \begin{bmatrix} U \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

Discretizing the time we get

$$\begin{bmatrix} \frac{M}{\Delta t} & 0 \\ 0 & \frac{M_p}{\varepsilon \Delta t} \end{bmatrix} \begin{bmatrix} U_{n+1} \\ p_{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} - (\begin{bmatrix} K & G_s \\ G_s^T & 0 \end{bmatrix} - \begin{bmatrix} \frac{M}{\Delta t} & 0 \\ 0 & \frac{M_p}{\varepsilon \Delta t} \end{bmatrix}) \begin{bmatrix} U_n \\ p_n \end{bmatrix}$$

3.2 Numerical example


$\bar{\boldsymbol{u}}_x = 1, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$

Consider the following example, lid-driven square cavity. We use the Q2 Q1 element here, which satisfies the stable test, our concern is only about the explicit treatment of the pressure.

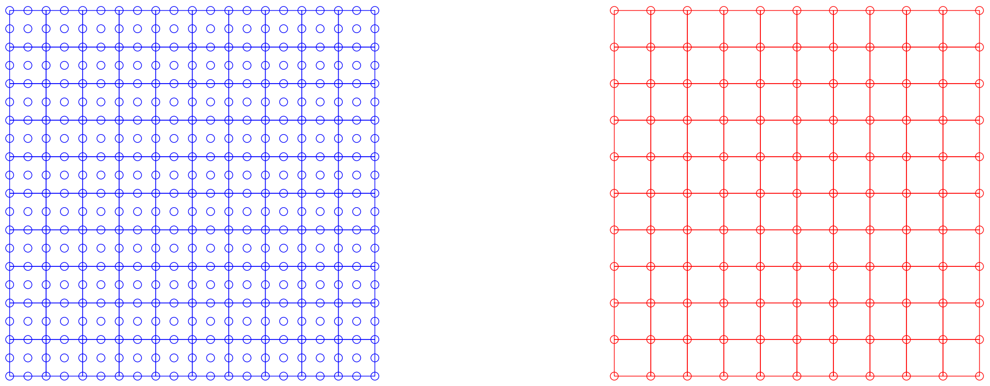


Figure 3.1: (a) mesh for the velocity (b) mesh for the pressure

Five different artificial speeds are employed to solve the cavity flow problem and structured meshes are shown in Figure 3.1, both velocity and pressure.

Figure 3.2 show the convergence history for Stokes flow for different artificial sound speed. The error is calculated from the following L_2 norm

$$error = \frac{1}{NN}\sum_{i=1}^{NN} |\nabla P| \tag{3.5}$$

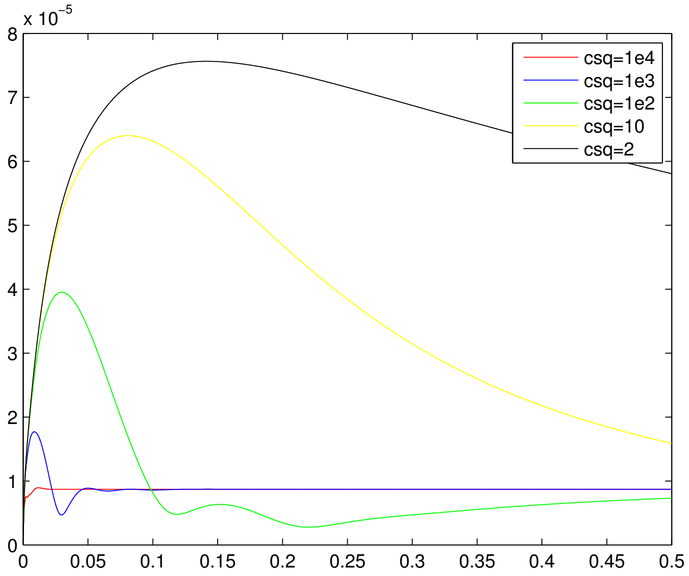


Figure 3.2: ACM convergence history in different artificial sound speed

As seen in Figure 3.2, (csq is the square of the artificial sound speed) when a large artificial sound speed is used, convergence reaches in very short time. We should also notice that large speed require small time step, which limit the use of artificial compressibility method. Often the real compressibility parameter is very high (approaches infinity) for many nearly incompressible flow problems and therefore employing real compressibility leads to severe time step restrictions on the fully explicit form of solution procedure. It is evident from recent work that a locally varying artificial compressibility parameter with a strong relationship to local time step makes the scheme efficient(Nithiarasu, 2003).

Chapter 4

Inf-Sup Condition and Pressure Stabilization

In this chapter we discuss the Stokes Problem. The Stokes problem is a well-posed problem by virtue of the so-called inf-sup condition, which ensures pressure stability. But in the fully discrete level, not all the velocity-pressure pair of finite-dimensional spaces satisfy the discrete inf-sup condition. In order to use the equal interpolation, stabilization of the formulation should be used. From this motivation, we review the typical stabilization and introduce the framework of variational multiscale method. In the end, a simple lid-driven cavity flow problem is studied using the equal interpolation.

4.1 The Stokes problem

4.1.1 Strong form

$$\begin{aligned} \boldsymbol{u}_t - \nu \nabla^2 \boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad & \text{in}\, \Omega \times\,]0, T[\\ \nabla \cdot \boldsymbol{u} = 0 \qquad & \text{in}\, \Omega \\ \boldsymbol{u} = 0 \qquad & \text{on}\, \partial\Omega \end{aligned} \tag{4.1}$$

where:

$\boldsymbol{f}$: Vector of body forces

ν: Kinematic viscosity

Ω: Computational domain

4.1.2 Weak form and LBB condition

(Ladyzhenskaya, 1968), (Babuska, 7071), (Brezzi, 1974) have determined the compatibility condition, known as the LBB(or Inf-sup) condition that continuous and discrete spaces must satisfy to guarantee the stability of a mixed method. The LBB condition states that velocity and pressure spaces cannot be chosen arbitrarily, a link between them is necessary. (Donea, 2003) lists a number of velocity-pressure spaces which pass the compatibility test shown in Fig. 4.1.

We introduce some notation, viscous term:

$$\boldsymbol{a}(\boldsymbol{u}, \boldsymbol{v}) := \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) \quad \forall \boldsymbol{u}, \boldsymbol{v} \in V \tag{4.2}$$

This is a bilinear continuous form on $H_0^1(\Omega)$ which is coercive with respect to$\|.\|_1$. The next form is used for the pressure gradient and the incompressibility constraint.

$$\boldsymbol{b}(\boldsymbol{v}, q) := -(q, \nabla \cdot \boldsymbol{v}) \quad \forall \boldsymbol{v} \in \mathcal{V}_0, \forall q \in \mathcal{L} \tag{4.3}$$

The variational form for the problem can be written in terms of bilinear forms which is also continuous with respect to the norms $\|q\|_0$ and $\|\boldsymbol{v}\|_1$.

Then, the weak form consist of finding $\boldsymbol{u} \in \mathcal{V}_0$ and $p \in \mathcal{L}$ such that:

$$\boldsymbol{a}(\boldsymbol{u},\boldsymbol{v}) + \boldsymbol{b}(\boldsymbol{v},q) = (f,\boldsymbol{v}) \qquad \forall \boldsymbol{u},\boldsymbol{v} \in \boldsymbol{v} \tag{4.4}$$

$$\boldsymbol{b}(\boldsymbol{u},q) = 0 \qquad \forall \boldsymbol{v} \in \mathcal{V}\ \forall q \in \mathcal{L} \tag{4.5}$$

The well-posedness of this problem relies on the coercivity of bilinear form and inf-sup condition:

$$\inf_{q\in\mathcal{L}} \sup_{\boldsymbol{v}\in V_0} \frac{\boldsymbol{b}(\boldsymbol{v},q)}{\|\boldsymbol{v}\|_{\mathcal{V}}\,\|q\|_{\mathcal{L}}} \geq \bar{\beta} > 0 \tag{4.6}$$

The pair of finite element spaces for the velocity and pressure should also satisfy this equation:

$$\inf_{q_h\in\mathcal{L}_h} \sup_{\boldsymbol{v}_h\in V_h} \frac{\boldsymbol{b}(\boldsymbol{v}_h,q_h)}{\|\boldsymbol{v}_h\|_{\mathcal{V}_h}\,\|q_h\|_{\mathcal{L}_h}} \geq \bar{\beta} > 0 \tag{4.7}$$

4.1.3 Stabilization technique

In recent years, the researchers are trying to use the velocity-pressure pairs which do not pass the compatibility test.

The basic idea behind stabilization procedure is to enforce the positive definiteness of matrix. This could be accomplished through a modification of the weak form of the incompressibility condition in order to render non-zero the diagonal term resulting from the incompressibility condition.

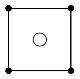
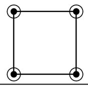
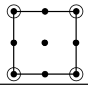
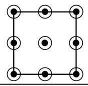
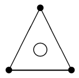
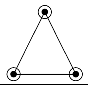
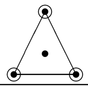
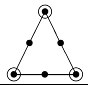
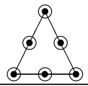
Mixed elements			
Figure	Name	Velocity interpolation	Pressure interpolation
	Q1Q0	Linear	Constant
	Q1Q1	Linear	Linear
	Q2Q1	Quadratic	Linear
	Q2Q2	Quadratic	Quadratic
	P1P0	Linear	Constant
	P1P1	Linear	Linear
	Mini	Linear	Linear
	P2P1	Quadratic	Linear
	P2P2	Quadratic	Quadratic
• Velocity node ○ Pressure node			

Figure 4.1: Types of mixed elements in two dimensions

$$\begin{aligned} \nu\left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{u}_h\right) + \left(\nabla p_h, \boldsymbol{u}_h\right) &= \langle f, \boldsymbol{v}_h \rangle \\ \tau\left(\nabla p_h, \nabla q_h\right) + \left(\nabla \cdot \boldsymbol{u}_h, q_h\right) &= 0 \\ \tau &= \frac{h^2}{\nu} \end{aligned} \tag{4.8}$$

$$\begin{bmatrix} K & G \\ G^T & -\tau K_\tau \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} \tag{4.9}$$

Note that the presence in the second equation of the term $(\nabla p_h, \nabla q_h)$ introduces a non-zero diagonal term. Then, we can use the backward euler and write the implicit formulation with stabilization technique.

$$\begin{aligned} (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu\,(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h) + (\nabla p_h, \boldsymbol{v}_h) &= \langle f, \boldsymbol{v}_h \rangle \\ \tau\,(\nabla p_h, \nabla q_h) + (\nabla \cdot \boldsymbol{u}_h, q_h) &= 0 \\ \tau &= \frac{h^2}{\nu} \end{aligned} \tag{4.10}$$

$$\begin{bmatrix} \frac{M}{\Delta t} + K & G \\ G^T & \tau K \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{n+1} \\ p_{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} - \begin{bmatrix} -\frac{M}{\Delta t} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_n \\ 0 \end{bmatrix} \tag{4.11}$$

In order to use the explicit method, we apply the AC scheme which is discussed in Chapter 3.

$$\begin{aligned} (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu\,(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h) + (\nabla p_h, \boldsymbol{u}_h) &= \langle f, \boldsymbol{v}_h \rangle \\ \varepsilon\,(\partial_t p_h, q_h) + \tau\,(\nabla p_h, \nabla q_h) + (\nabla \cdot \boldsymbol{u}_h, \nabla q_h) &= 0 \\ \tau &= \frac{h^2}{\nu} \end{aligned} \tag{4.12}$$

$$\begin{bmatrix} \frac{M}{\Delta t} & 0 \\ 0 & \varepsilon \frac{M}{\Delta t} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{n+1} \\ p_{n+1} \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} - \begin{bmatrix} K - \frac{M}{\Delta t} & G \\ G^T & \tau K \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_n \\ p_n \end{bmatrix} \tag{4.13}$$

4.2 Variational multiscale method

Analog to the Chapter 3, the variational multiscale method framework is applied to stabilize the pressure filed. But the aim of VMS here is to use the equal interpolation for velocity-pressure field, rather than stabilize the convection term.

Algorithm 4.1 *Explicit* stabilization algorithm for the Stokes problem

```
read $\boldsymbol{u}_h^0$, $p_h^0$ and set $\tilde{\boldsymbol{u}} = 0$
while Time integration do
    Calculate the parameters $\tau$
    for Element do
        velocity $\boldsymbol{u}^n$ and pressure $p^n$ at cell nodes
        velocity gradient $\nabla \boldsymbol{u}^n$ at cell nodes
        velocity residual $\nu\left(\nabla \boldsymbol{u}^n, \nabla \boldsymbol{v}_h\right) + \left(\nabla p^n, \boldsymbol{v}_h\right) - \left(f, \boldsymbol{v}_h\right)$
        pressure residual $\left(\nabla \cdot \boldsymbol{u}, q_h\right) + \tau\left(\nabla p^n, \nabla q_h\right)$
    end for
    update the velocity and pressure
    prescribe boundary conditions
end while
```

4.2.1 Formulation

Using the same notations we introduced in Chapter 2 and Chapter 3. We consider scale decomposition of spaces $\mathcal{V}$ and $\mathcal{L}$. The muti-scale is applied to the weak form of the Navier-Stokes equations.

$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \mathcal{L} = \mathcal{L}_h \oplus \tilde{\mathcal{L}}, \boldsymbol{u}_h \in \mathcal{V}_h, \tilde{\boldsymbol{u}} \in \tilde{\mathcal{V}}, p_h \in \mathcal{L}_h, \tilde{p} \in \tilde{\mathcal{L}}$

Split the velocity $\boldsymbol{u}$ and pressure p as

$$\begin{cases} \boldsymbol{u} = \boldsymbol{u}_h + \tilde{\boldsymbol{u}} \\ p = p_h + \tilde{p} \end{cases} \tag{4.14}$$

where $\boldsymbol{u}_h$ and p_h belong to the finite element spaces and $\tilde{\boldsymbol{u}}$ and $\tilde{p}$ are fine sub-scale components.

The problem becomes

$$\begin{cases} \left(\partial_t\left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}\right) + \nu\left(\nabla\left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}\right) - \left(p_h + \tilde{p}, \nabla \cdot \boldsymbol{v}\right) = \langle f, \boldsymbol{v}\rangle \\ \left(\partial_t\left(p_h + \tilde{p}\right), q\right) + \left(\varepsilon \nabla \cdot \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), q\right) = 0 \end{cases} \tag{4.15}$$

First we consider $\boldsymbol{v}_h \in \mathcal{V}_h, q_h \in \mathcal{L}_h$. The resolved large-scale problem becomes

$$\begin{cases} \left(\partial_t \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h\right) + \nu \left(\nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}_h\right) - \left(p_h + \tilde{p}, \nabla \cdot \boldsymbol{v}_h\right) = \langle f, \boldsymbol{v}_h \rangle \\ \left(\partial_t \left(p_h + \tilde{p}\right), q_h\right) + \left(\varepsilon \nabla \cdot \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), q_h\right) = 0 \end{cases} \tag{4.16}$$

As a result, we have the final large scales equation

$$\begin{cases} \left(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h\right) + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) - \left(p_h, \nabla \cdot \boldsymbol{v}_h\right) - \left(\tilde{p}, \nabla \cdot \boldsymbol{v}_h\right) = 0 \\ \left(\partial_t p_h, q_h\right) + \left(\varepsilon \nabla \cdot \boldsymbol{u}_h, q_h\right) = 0 \end{cases} \tag{4.17}$$

4.2.2 Sub-scale problem statement

Then, we consider $\tilde{\boldsymbol{v}} \in \tilde{\mathcal{V}}, \tilde{q} \in \tilde{\mathcal{L}}$. The unresolved sub-scale problem becomes

$$\begin{cases} \partial_t \tilde{\boldsymbol{u}} - \nu \Delta \tilde{\boldsymbol{u}} + \nabla \tilde{p} = r_{u,h} \\ \partial_t \tilde{p} + \varepsilon \nabla \cdot \tilde{\boldsymbol{u}} = r_{p,h} \end{cases} \tag{4.18}$$

$$\begin{cases} r_{\boldsymbol{u}_h} = -P \left(\partial_t \boldsymbol{u}_h - \nu \Delta \boldsymbol{u}_h + \nabla p_h - f\right) \\ r_{p,h} = -P \left(\partial_t p_h + \varepsilon \nabla \cdot \boldsymbol{u}_h\right) \end{cases} \tag{4.19}$$

where $r_{u,h}, r_{p,h}$ is appropriate residual of the finite element components $\boldsymbol{u}_h$ and p_h adequately projected onto the space of subscales $\tilde{\mathcal{V}}, \tilde{\mathcal{L}}$. As the concept we introduced in chapter 2, we have Algebraic Subgrid-Scale formulation (ASGS) and Orthogonal Subscales Stabilization (OSS).

4.2.3 Sub-scale approximation

We could solve the subscale analytically using the Fourier transform. In this chapter we directly give the approximation solution as follows. The details for subscale approximation will be discussed in Chapter 5.

$$\begin{cases} \tilde{\boldsymbol{u}}_{n+1} = \frac{\tau_t}{\Delta t}\tilde{\boldsymbol{u}}_n - \tau_t\left[r_{u,h} + \tau_2\tilde{p}_n\right] \\ \tilde{p}_{n+1} = \tilde{p}_n - \Delta t\left[r_{p,h} - \varepsilon\tau_2\tilde{\boldsymbol{u}}_n\right] \end{cases} \tag{4.20}$$

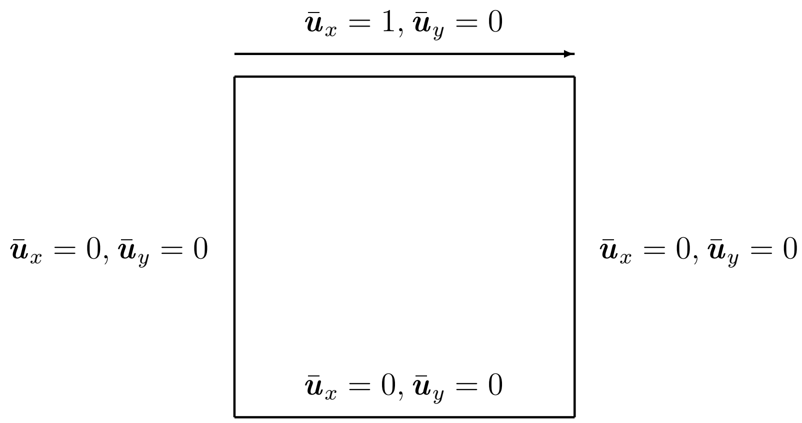
$$where \begin{cases} \tau_1 = c_1\frac{\nu}{h} \\ \tau_2 = c_3\frac{1}{h} \\ \tau_t = \frac{1}{\frac{1}{\Delta t}+\tau_1} \end{cases}$$

Algorithm 4.2 *Explicit* VMS algorithm for the Stokes problem

```
read $\boldsymbol{u}_h^0$, $p_h^0$ and set $\tilde{u} = 0$
while Time integration do
    Calculate the parameters $\tau$
    for Element do
        velocity $\boldsymbol{u}^n$ and pressure $p^n$ at cell nodes
        velocity gradient $\nabla\boldsymbol{u}^n$ at cell nodes
        velocity field with subgrid scale $\boldsymbol{a} = \boldsymbol{u}^n + \tilde{\boldsymbol{u}}^n$
        velocity residual $\nu(\nabla\boldsymbol{u}^n, \nabla\boldsymbol{v}_h) + (\nabla p^n, \boldsymbol{v}_h) - (f, \boldsymbol{v}_h)$
        pressure residual $(\nabla\cdot\boldsymbol{a}, q_h)$
        first contribution to the velocity subscales $\frac{\tau_t}{\Delta t}\boldsymbol{u}^n - \tau_t\nabla p^n$
        projection of first velocity subscales $\nabla p^n$
    end for
    update the velocity and pressure
    prescribe boundary conditions
    projection contribution $\tilde{\boldsymbol{u}}^{n+1}$
end while
```

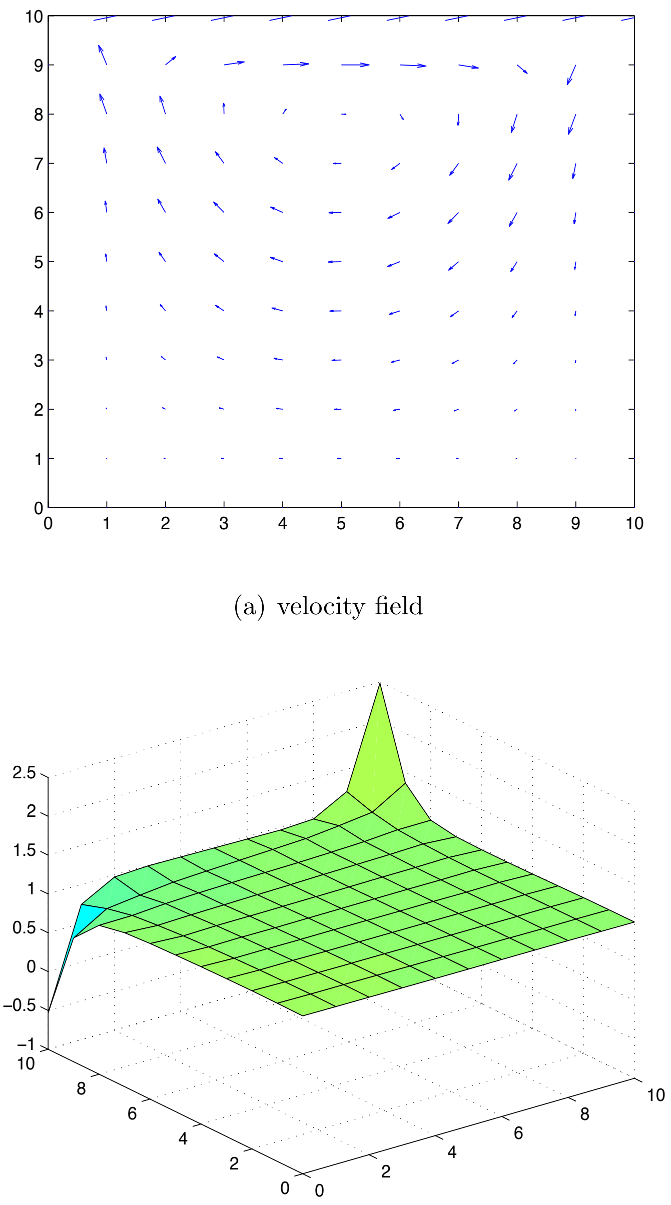
4.3 Numerical example

This section presents the solution of a simple lid-driven cavity problem in two explicit stabilized methods, a straightforward stabilized method and the OSS method with dynamic subscales. The stationary Stokes problem has been solved using a mesh with 10×10 Q1 elements, with equal linear velocity and pressure interpolation.


$\bar{\boldsymbol{u}}_x = 1, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$
$\bar{\boldsymbol{u}}_x = 0, \bar{\boldsymbol{u}}_y = 0$

Here we show the Stokes flow in a cavity, stabilized method Algorithm 4.1 and OSS Algorithm 4.2 are used. the results for both methods are the same and all the stable velocity and pressure fields are obtained. Numerical results are shown in Fig.4.2.

The stabilized method, which could be understood as a static subscales approach, has the same performance as the transient (dynamic) subscales in this case. However, for the transient Navier-Stokes (NS) problem, it is worth to track the subscales in time in a variational multiscale approach (Badia & Codina, 2008), which would be shown in the chapter 5.



(a) velocity field

(b) pressure field

Figure 4.2: Cavity flow for the Stokes problem

Chapter 5

Explicit Variational Multiscale Method for Incompressible Flow

In this chapter presents the key part of this thesis, an explicit variational multiscale method for incompressible flow, suitable for LES-type turbulence modeling. The turbulent modeling is entirely based on the numerical scheme rather than large eddy modeling, which is more fundamental and logically consistent.

This chapter is organized as follows. We first propose the explicit variational multiscales method for incompressible flow, then we explore the property of the discrete formulation and compare with the incompressible Navier-Stokes equations. And we show that the dynamic sub-grid model provides a direct numerical approach to modeling the turbulence and relations with typical LES approaches. In the last, we talk about the time-stepping concerning with stability issue.

5.1 Problem statement

The Newtonian incompressible turbulent flows is governed by Navier-Stokes equations.

Strong form(S): Assume domain $\boldsymbol{u} \subset \mathbb{R}^d$ where $d = 2$ or 3,with boundary $\Gamma = \partial\Omega$

The initial/boundary-value problem is Find the velocity $\boldsymbol{u}$ and the pressure p

$$\begin{aligned} \partial_t \boldsymbol{u} - \nu \nabla^2 \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p &= \boldsymbol{f} \quad \text{in}\, \Omega, t \in]0, T[\\ \nabla \cdot \boldsymbol{u} &= 0 \quad \text{in}\, \Omega, t \in]0, T[\\ \boldsymbol{u} &= 0 \\ \boldsymbol{u}(t = 0) &= 0 \end{aligned} \tag{5.1}$$

As the notations we introduced in the previous chapter, Let $\mathcal{V} \equiv H^1(\Omega)$,$\mathcal{V}_0 \equiv H_0^1(\Omega)$ and $\mathcal{L} \equiv L^2(\Omega)/\mathbb{R}$ denote the real Hilbert spaces for velocity and pressure, with associated norms $\|\boldsymbol{v}\|_v$ and $\|q\|_L$, and let $\boldsymbol{f} \in \mathcal{V}^{-1}$.

The starting point is to apply the artificial compressibility (AC) method, we get the perturbed version of the NS equations.

$$\begin{aligned} \partial_t \boldsymbol{u} - \nu \nabla^2 \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p &= \boldsymbol{f} \quad \text{in}\, \Omega, t \in]0, T[\\ \partial_t p + \varepsilon \nabla \cdot \boldsymbol{u} &= 0 \quad \text{in}\, \Omega, t \in]0, T[\\ \boldsymbol{u} &= 0 \\ \boldsymbol{u}(t = 0) &= 0 \end{aligned} \tag{5.2}$$

5.2 Variational multiscale formulation

Here we recall the concept of variational multiscale(VMS) method. It splits the solution $\boldsymbol{u}, p$, on a coase-scale component $\boldsymbol{u}_h, p_h$, which can be resolved by the finite element mesh, and a subscale component $\tilde{\boldsymbol{u}}$ and $\tilde{p}$, which attempt to determine analytically.

Consider scale decomposition of spaces $\mathcal{V}$ and $\mathcal{L}$. $\mathcal{V}_h$ and $\mathcal{L}_h$ are the finite element spaces to approximate the velocity and pressure. The multiscale is applied to the weak form of the Navier-Stokes equations.

$\mathcal{V} = \mathcal{V}_h \oplus \tilde{\mathcal{V}}, \mathcal{L} = \mathcal{L}_h \oplus \tilde{\mathcal{L}}, \boldsymbol{u}_h \in \mathcal{V}_h, \tilde{\boldsymbol{u}} \in \tilde{\mathcal{V}}, p_h \in \mathcal{L}_h, \tilde{p} \in \tilde{\mathcal{L}}$

Split the velocity $\boldsymbol{u}$ and pressure p as

$$\begin{cases} \boldsymbol{u} = \boldsymbol{u}_h + \tilde{\boldsymbol{u}} \\ p = p_h + \tilde{p} \end{cases} \tag{5.3}$$

where $\boldsymbol{u}_h$ and p_h belong to the finite element spaces and $\tilde{\boldsymbol{u}}$ and $\tilde{p}$ are sub-grid scales or the subscales. We can identify the finite element components of the solution as the resolved scales, whereas the subscales are the unresolved scales.

The problem becomes

$$\begin{cases} \left(\partial_t \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}\right) + \nu \left(\nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}\right) + \left\langle \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right) \cdot \nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), \boldsymbol{v}\right\rangle \\ \qquad - \left(p_h + \tilde{p}, \nabla \cdot \boldsymbol{v}\right) = \langle f, \boldsymbol{v}\rangle \\ \left(\partial_t \left(p_h + \tilde{p}\right), q\right) + \left(\varepsilon \nabla \cdot \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), q\right) = 0 \end{cases} \tag{5.4}$$

First we consider $\boldsymbol{v}_h \in \mathcal{V}_h, q_h \in \mathcal{L}_h$. The resolved large-scale problem becomes

$$\begin{cases} \left(\partial_t \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h\right) + \nu \left(\nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}_h\right) + \left\langle \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right) \cdot \nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), \boldsymbol{v}_h\right\rangle \\ \qquad - \left(p_h + \tilde{p}, \nabla \cdot \boldsymbol{v}_h\right) = \langle f, \boldsymbol{v}_h\rangle \\ \left(\partial_t \left(p_h + \tilde{p}\right), q_h\right) + \left(\varepsilon \nabla \cdot \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h\right), q_h\right) = 0 \end{cases} \tag{5.5}$$

- **Remark 1**:

$$\left(\partial_t \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h\right) = \left(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h\right) \tag{5.6}$$

The term $(\partial_t \tilde{\boldsymbol{u}}, \boldsymbol{v}_h)$vanish because the sub-scales are assumed to be orthogonal to the finite element space.

- **Remark 2**:

$$\begin{aligned}
&\nu \left(\nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \nabla \boldsymbol{v}_h\right) \\
&= \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) + \nu \left[\sum_K \left(-\tilde{\boldsymbol{u}}, \Delta \boldsymbol{v}_h\right)_K + \langle \tilde{\boldsymbol{u}}_h, n \cdot \nabla \boldsymbol{v}_h \rangle_{\partial K}\right] \\
&= \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right)
\end{aligned} \tag{5.7}$$

The $\left(-\tilde{\boldsymbol{u}}, \Delta \boldsymbol{v}_h\right)_K$ vanish because linear element is used. $\langle \tilde{\boldsymbol{u}}, n \cdot \nabla \boldsymbol{v}_h \rangle_{\partial K}$ vanish because velocity subscales vanishing on the boundary like bubble functions.

- **Remark 3**:

$$\begin{aligned}
&\langle \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right) \cdot \nabla \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right), \boldsymbol{v}_h \rangle \\
&= \langle \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right) \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \int \left(\boldsymbol{u} + \tilde{\boldsymbol{u}}\right)_i \partial_i \tilde{\boldsymbol{u}}_j \boldsymbol{v}_j \\
&= \int \partial_i \tilde{\boldsymbol{u}}_j \left(\left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \boldsymbol{v}_j\right) \\
&= \int \partial_i \left(\tilde{\boldsymbol{u}}_j \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \boldsymbol{v}_j\right) - \int \tilde{\boldsymbol{u}}_j \partial_i \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \boldsymbol{v}_j \\
&= \int_{\partial\Omega} n_i \left(\tilde{\boldsymbol{u}}_j \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \boldsymbol{u}_j\right) - \int \tilde{\boldsymbol{u}}_j \partial_i \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \boldsymbol{v}_j \\
&= -\int \tilde{\boldsymbol{u}}_j \left[\partial_i \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i\right] \boldsymbol{v}_j - \int \tilde{\boldsymbol{u}}_j \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)_i \partial_i \boldsymbol{v}_j \\
&= \langle \left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right) \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle - \langle \tilde{\boldsymbol{u}}, \left(\left(\boldsymbol{u}_h + \tilde{\boldsymbol{u}}\right)\right) \cdot \nabla \boldsymbol{v}_h \rangle
\end{aligned} \tag{5.8}$$

- **Remark 4**:

$$\left(\partial_t \left(p_h + \tilde{p}\right), q_h\right) = \left(\partial_t p_h, q_h\right) \tag{5.9}$$

In the last, we have the equation

$$\left\{\begin{array}{c} (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) - (p_h, \nabla \cdot \boldsymbol{v}_h) - (\tilde{p}, \nabla \cdot \boldsymbol{v}_h) \\ - \langle \tilde{\boldsymbol{u}}, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h \rangle = 0 \\ \boldsymbol{a} = \boldsymbol{u}_h + \tilde{\boldsymbol{u}} \\ (\partial_t p_h, q_h) + (\varepsilon \nabla \cdot \boldsymbol{a}, q_h) = 0 \end{array}\right. \tag{5.10}$$

5.3 Main feature of the formulation

Let us analyze the implications of the formulation and write the first equation in another form:

$$\begin{aligned} &(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) - (p_h, \nabla \cdot \boldsymbol{v}_h) - (\tilde{p}, \nabla \cdot \boldsymbol{v}_h) - \langle \tilde{\boldsymbol{u}}, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h \rangle \\ &\quad = (\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \nu \left(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h\right) + \langle \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v} \rangle \\ &\qquad - (p_h, \nabla \cdot \boldsymbol{v}_h) - \langle \boldsymbol{v}_h, f \rangle \quad \text{Galerkin terms} \\ &\qquad + \langle \tilde{\boldsymbol{u}}, \boldsymbol{u}_h \cdot \nabla \boldsymbol{v}_h \rangle \quad \text{Stabilization terms} \\ &\qquad + \langle \tilde{u} \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \rangle - \langle \tilde{\boldsymbol{u}}, \tilde{\boldsymbol{u}} \cdot \nabla \boldsymbol{v}_h \rangle \quad \text{Effect of } \tilde{\boldsymbol{u}} \text{ in material derivative} \end{aligned} \tag{5.11}$$

The stabilization terms appear also in the stationary and linear problem, which are use to stabilized the convection-dominated flows and make equal velocity and pressure interpolation possible, as we seen in Chapter 2 and Chapter 4.

5.4 Local sub-scale problem and its approximation

In this section, we discuss the sub-scale formulation and then give the analytical approximation solution by Fourier transform. In the last, the orthogonal subscales in the Explicit VMS method is given.

5.4.1 Sub-scale problem statement

$$\begin{cases} \partial_t \tilde{\boldsymbol{u}} + (\boldsymbol{u}_h + \tilde{\boldsymbol{u}}) \cdot \nabla \tilde{\boldsymbol{u}} - \nu \Delta \tilde{\boldsymbol{u}} + \nabla \tilde{p} = r_{u,h} \\ \partial_t \tilde{p} + \varepsilon \nabla \cdot \tilde{\boldsymbol{u}} = r_{p,h} \end{cases} \tag{5.12}$$

$$\begin{cases} r_{\boldsymbol{u}_h} = -P\left(\partial_t \boldsymbol{u}_h + (\boldsymbol{u}_h + \tilde{\boldsymbol{u}}) \cdot \nabla \boldsymbol{u}_h - \nu \Delta \boldsymbol{u}_h + \nabla p_h - f\right) \\ r_{p,h} = -P\left(\partial_t p_h + \varepsilon \nabla \cdot \boldsymbol{u}_h\right) \end{cases}$$

where $r_{u,h}$ and $r_{p,h}$ are appropriate residual of the finite element components. As the concept we introduced in chapter 2 and chapter 4, we have Algebraic Subgrid-Scale formulation (ASGS) and Orthogonal Subscales Stabilization (OSS).

5.4.2 Sub-scale approximation

(Codina, 2002) analysis of the problem for the sub-scales. Consider the Fourier transform of a generic function $\boldsymbol{g}$ defined on $\boldsymbol{K}$:

$$\hat{\boldsymbol{g}}(\boldsymbol{k}) := \int_{\boldsymbol{K}} \boldsymbol{e}^{-\frac{\boldsymbol{k} \cdot \boldsymbol{x}}{h}} \boldsymbol{g}(\boldsymbol{x}) d\Omega \tag{5.13}$$

where $i = \sqrt{-1}$,h is now the diameter of element $\boldsymbol{K}$ and $\boldsymbol{k} = (k_1, ..., k_d)$ is the dimensionless wave number.

The subscales $\tilde{\boldsymbol{u}}$ are the continuous solution which do not belong to the finite element spaces. This means the Fourier representation will be dominated by the components with high wave numbers. If n_j is the jth component of normal exterior to $\boldsymbol{K}$, it can be known that

$$\frac{\partial \hat{\boldsymbol{g}}}{\partial x_j}(\boldsymbol{k}) = \int_{\partial K} n_j \boldsymbol{e}^{-i\frac{\boldsymbol{k}\cdot\boldsymbol{x}}{h}} \boldsymbol{g}\left(\boldsymbol{x}\right) d\Gamma_x + i\frac{\boldsymbol{k}_j}{h}\hat{}(\boldsymbol{k}) \tag{5.14}$$

From this expression it is seen that if we are interested in high wave numbers, the second term in the right -hand side of this expression dominates the first one, Thus, for functions with high wave numbers we may approximate

$$\frac{\partial \hat{\boldsymbol{g}}}{\partial \boldsymbol{x}_j}(\boldsymbol{k}) \approx \int_{\partial \boldsymbol{k}} i\frac{\boldsymbol{k}_j}{h}\hat{\boldsymbol{g}}(\boldsymbol{k}) \tag{5.15}$$

Applying them to the subscale problem

$$\left\{ \begin{array}{c} \partial_t \hat{\boldsymbol{u}}(\boldsymbol{k}) + (\nu\frac{|\boldsymbol{k}|^2}{h^2} + i\frac{\boldsymbol{a}\cdot\boldsymbol{k}}{h})\hat{\boldsymbol{u}}(\boldsymbol{k}) + i\frac{\boldsymbol{k}}{h}\hat{\boldsymbol{p}}(\boldsymbol{k}) = r_{u,h} \\ \partial_t \hat{\boldsymbol{p}}(\boldsymbol{k}) + i\varepsilon\frac{\boldsymbol{k}}{h}\hat{\boldsymbol{u}}(\boldsymbol{k}) = r_{p,h} \end{array} \right. \tag{5.16}$$

Here we get an ordinary partial differential equation, then discretize

$$\left\{ \begin{array}{c} \tilde{\boldsymbol{u}}_{n+1} = \frac{\tau_t}{\Delta t}\tilde{\boldsymbol{u}}_n - \tau_t\left[r_{u,h} + \tau_2 \tilde{p}_n\right] \\ \tilde{p}_{n+1} = \tilde{p}_n - \Delta t\left[r_{p,h} - \varepsilon\tau_2\tilde{\boldsymbol{u}}_n\right] \end{array} \right. \tag{5.17}$$

$$where \left\{ \begin{array}{c} \tau_1 = c_1\frac{\nu}{h} + c_2\frac{|u_h + \tilde{u}|}{h} \\ \tau_2 = c_3\frac{1}{h} \\ \tau_t = \frac{1}{\frac{1}{\Delta t} + \tau_1} \end{array} \right.$$

5.4.3 Orthogonal subscales

(Codina, 2002) first introduced the concept 'orthogonal subscales'. Here we apply the concept to the Explicit VMS subscales.

$$\begin{array}{l} \prod_h^{\perp}\left(\partial_t \boldsymbol{u}_h + (\boldsymbol{u}_h + \tilde{\boldsymbol{u}})\cdot\nabla\boldsymbol{u}_h - \nu\Delta\boldsymbol{u}_h + \nabla p_h - f\right) \\ = \left(\partial_t \boldsymbol{u}_h + (\boldsymbol{u}_h + \tilde{\boldsymbol{u}})\cdot\nabla\boldsymbol{u}_h - \nu\Delta\boldsymbol{u}_h + \nabla p_h - f\right) \\ - \prod_h\left(\partial_t \boldsymbol{u}_h + (\boldsymbol{u}_h + \tilde{\boldsymbol{u}})\cdot\nabla\boldsymbol{u}_h - \nu\Delta\boldsymbol{u}_h + \nabla p_h - f\right) \\ = (\boldsymbol{u}_h + \tilde{\boldsymbol{u}})\cdot\nabla\boldsymbol{u}_h + \nabla p_h - \prod_h\left((\boldsymbol{u}_h + \tilde{\boldsymbol{u}})\cdot\nabla\boldsymbol{u}_h + \nabla p_h\right) \end{array} \tag{5.18}$$

$$\begin{aligned} & \prod\nolimits_h^{\perp} \left(\partial_t p_h + \varepsilon \nabla \cdot \boldsymbol{u}_h\right) \\ & = \left(\partial_t p_h + \varepsilon \nabla \cdot \boldsymbol{u}_h\right) - \prod\nolimits_h \left(\partial_t p_h + \varepsilon \nabla \cdot \boldsymbol{u}_h\right) \\ & = \varepsilon \nabla \cdot \boldsymbol{u}_h - \prod\nolimits_h \varepsilon \nabla \cdot \boldsymbol{u}_h \end{aligned} \tag{5.19}$$

The subscales could also be approximated by Algebraic Subgrid-Scale formulation (ASGS). Here the Orthogonal Subscales Stabilization (OSS) is used and substitute the subscales to large scales, we have the Algorithm 5.1.

Algorithm 5.1 *Explicit* VMS algorithm for incompressible turbulent flow

```
read $\boldsymbol{u}_h^0$, $p_h^0$ and set $\tilde{\boldsymbol{u}} = 0$
while Time integration do
    Calculate the parameters $\tau$
    for Element do
        velocity $\boldsymbol{u}^n$ and pressure $p^n$ at cell nodes
        velocity gradient $\nabla \boldsymbol{u}^n$ at cell nodes
        velocity with subgrid scale $\boldsymbol{a} = \boldsymbol{u}^n + \tilde{\boldsymbol{u}}^n$
        velocity residual $(\boldsymbol{a} \cdot \nabla \boldsymbol{u}^n, \boldsymbol{v}_h) + \nu (\nabla \boldsymbol{u}^n, \nabla \boldsymbol{v}_h) + (\nabla p^n, \boldsymbol{v}_h) - (\tilde{\boldsymbol{u}}^n, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h) -$
        $(f, \boldsymbol{v}_h)$
        pressure residual $(\nabla \cdot \boldsymbol{a}, q_h)$
        first contribution to the velocity subscales $\frac{\tau_t}{\Delta t} \boldsymbol{u}^n - \tau_t (\boldsymbol{a} \cdot \nabla \boldsymbol{u}^n + \nabla p^n)$
        projection of first velocity subscales $(\boldsymbol{a} \cdot \nabla \boldsymbol{u}^n + \nabla p^n)$
    end for
    update the velocity and pressure
    prescribe boundary conditions
    projection contribution $\tilde{\boldsymbol{u}}^{n+1}$
end while
```

5.5 A door to turbulence

(Codina et al., 2007) and (Badia & Codina, 2009) state the relation between VMS method and LES model. The LES model often adds a divergence of the "*residual stress tensor or subgrid-scale tensor* "

$R := \overline{\boldsymbol{u} \otimes \boldsymbol{u}} - \bar{\boldsymbol{u}} \otimes \bar{\boldsymbol{u}}$

The residual stress tensor, R is often decomposed into Reynolds stress, Cross stress and Leonard stress.

$-\left\langle \tilde{\boldsymbol{u}}, \tilde{\boldsymbol{u}} \cdot \nabla \boldsymbol{v}_h \right\rangle = -\left\langle \tilde{\boldsymbol{u}} \otimes \tilde{\boldsymbol{u}}, \nabla \boldsymbol{v}_h \right\rangle$ Reynolds stress.

$\left\langle \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h \right\rangle - \left\langle \tilde{\boldsymbol{u}}, \boldsymbol{u}_h \cdot \nabla \boldsymbol{v}_h \right\rangle + \left\langle \tilde{\boldsymbol{u}} \cdot \nabla \boldsymbol{v}_h \right\rangle$
$= -\left\langle \boldsymbol{u}_h \otimes \boldsymbol{u}_h, \nabla \boldsymbol{v}_h \right\rangle$ Convection part
$-\left\langle \boldsymbol{u}_h \otimes \tilde{\boldsymbol{u}} + \tilde{\boldsymbol{u}} \otimes \boldsymbol{u}_h, \nabla \boldsymbol{v}_h \right\rangle$ Cross stress

The convective term of the residual in the subscale equation and take $P = I$, we have
$\left\langle (\boldsymbol{u}_h + \tilde{\boldsymbol{u}}) \cdot \nabla \boldsymbol{u}_h, \tilde{\boldsymbol{v}} \right\rangle = -\left\langle \boldsymbol{u}_h \otimes \boldsymbol{u}_h, \nabla \tilde{\boldsymbol{v}} \right\rangle$ Leonard stress
$-\left\langle \boldsymbol{u}_h \otimes \tilde{\boldsymbol{u}}, \nabla \tilde{\boldsymbol{v}} \right\rangle$

The divergence of R in the LES equation is automatically included in the variational multiscale method. The formulation depends on the validity of the approximation made to derive the evolution for the subscales, In order to check the performance, some benchmark problems for turbulent flows are tested in the next chapter. The model should be able to reproduce the Kolmogorov energy cascade in the wavenumber Fourier space that displays an inertial range, where $E(k,t) = C_k \varepsilon^{2/3} k^{-5/3}$($\varepsilon$ being the energy dissipation rate, k the wave-space and E the kinetic energy).

5.6 Time stepping

As mentioned in Chapter 3, the compressible wave speed for many problem of fluid dynamics is very large and the solution scheme becomes stiff and imposes severe time step restrictions. However, it can be replaced by an appropriate artificial sound speed β.

$$\beta = \frac{u_{max}}{Mach} \tag{5.20}$$

where $Mach$ is artificial Mach number always less than 1.

From the explicit method stability consideration, the Courant number calculated from artificial sound speed β should less than 1.

$$Co = \frac{h}{\beta \Delta t} < 1 \tag{5.21}$$

where h is the element size

Then we can chose the step Δt by

$$\Delta t = \alpha \frac{h}{\beta} \tag{5.22}$$

where α is the safe factor.

Chapter 6

Numerical Examples

The following numerical examples are intended to show the performance of explicit variational multiscale method introduced in this thesis. Two dimensional cases (the lid-driven cavity flow, plane mixed layer and decaying of the 2D isotropic turbulence) and three dimensional cases (lid-driven cavity flow and decaying of the 3D isotropic turbulence) are studied.

6.1 Lid-driven cavity in 2D (low Reynolds flows)

The first problem considered here is the well-known benchmark case of lid-driven square cavity. The flow is driven by the uniform motion of one side of a box. Four cases in a wide range of Reynolds numbers, $Re = 100$, $Re = 1000$, $Re = 5000$ and $Re = 10000$ are employed to test the lid-driven cavity flow problem. Here the 'low Reynolds flows' means the flow would reach to steady state in these Reynolds numbers.

6.1.1 Convergence test

Figure 6.1 shows the convergence histories in different Reynolds numbers.

The error is calculated from the following L^2 norm

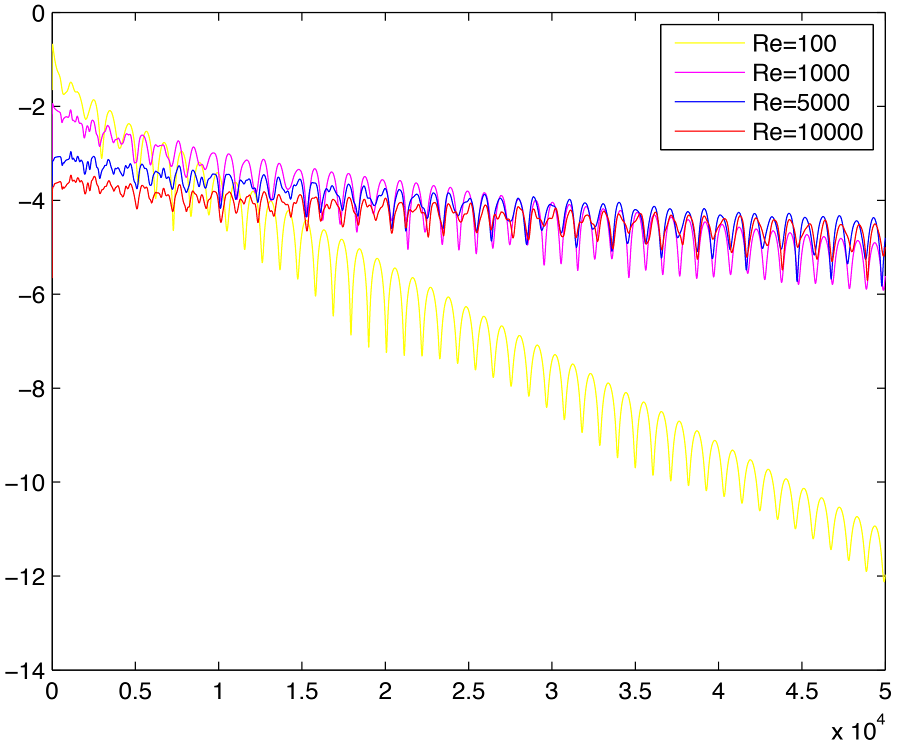


Figure 6.1: Convergence history for a 100 × 100 uniform mesh of quadrilaterals.

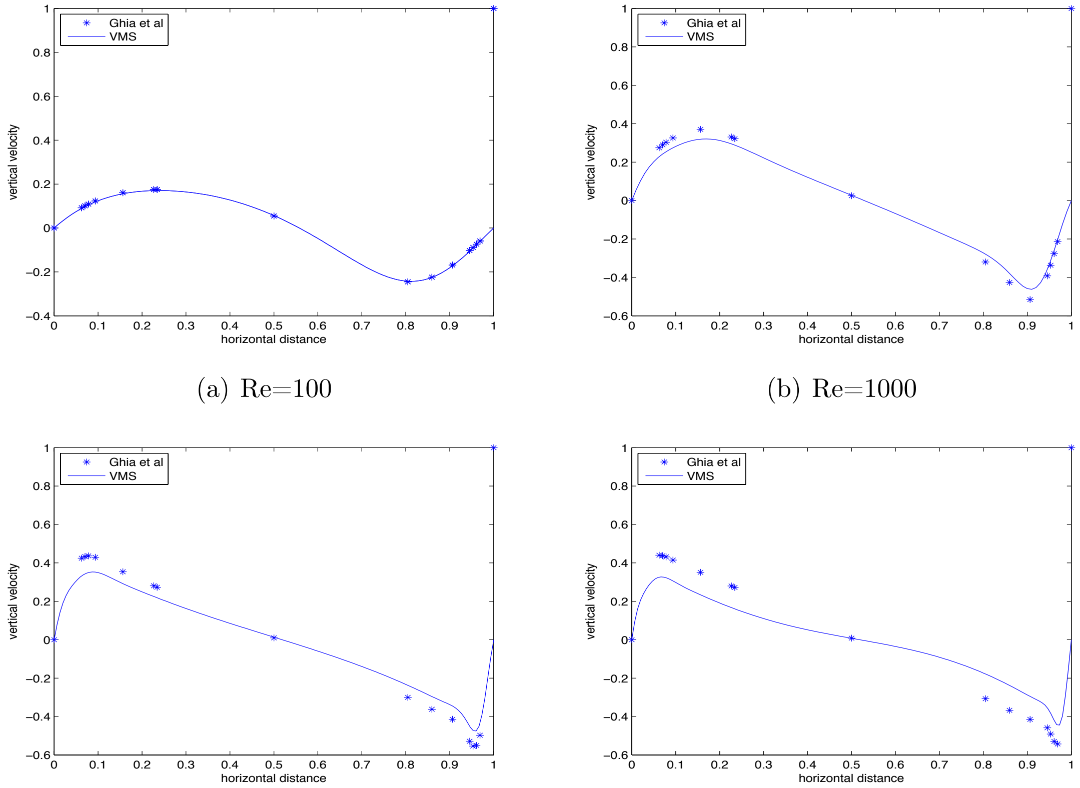
$$Error = \frac{1}{NN} \sum_{i=1}^{NN} \left[\frac{1}{\beta^2} \left(\frac{p^{n+1} - p^n}{\Delta t} \right) \right]^2 \tag{6.1}$$

where NN is the total number of nodes.

The convergence rate decreases when the Reynolds number goes up. The solution convergence histories are excellent even for Reynolds number 10000 in a 100 × 100 mesh. However, for Re=100000, the error first oscillates then blows up in the 100×100 grid.

6.1.2 Numerical result

100 × 100 bilinear quadrilateral element mesh and OSS method is applied to solve this problem. Fig.6.4, Fig.6.5, Fig.6.6, Fig.6.7 compare the velocity, pressure, velocity x-component and velocity y-component in different Reynolds number $Re = 100$, $Re = 100$, $Re = 100$ and $Re = 10000$. The data obtained by Ghia et al. (1982), which become a standard reference, are compared with the result from VMS method.



(a) Re=100

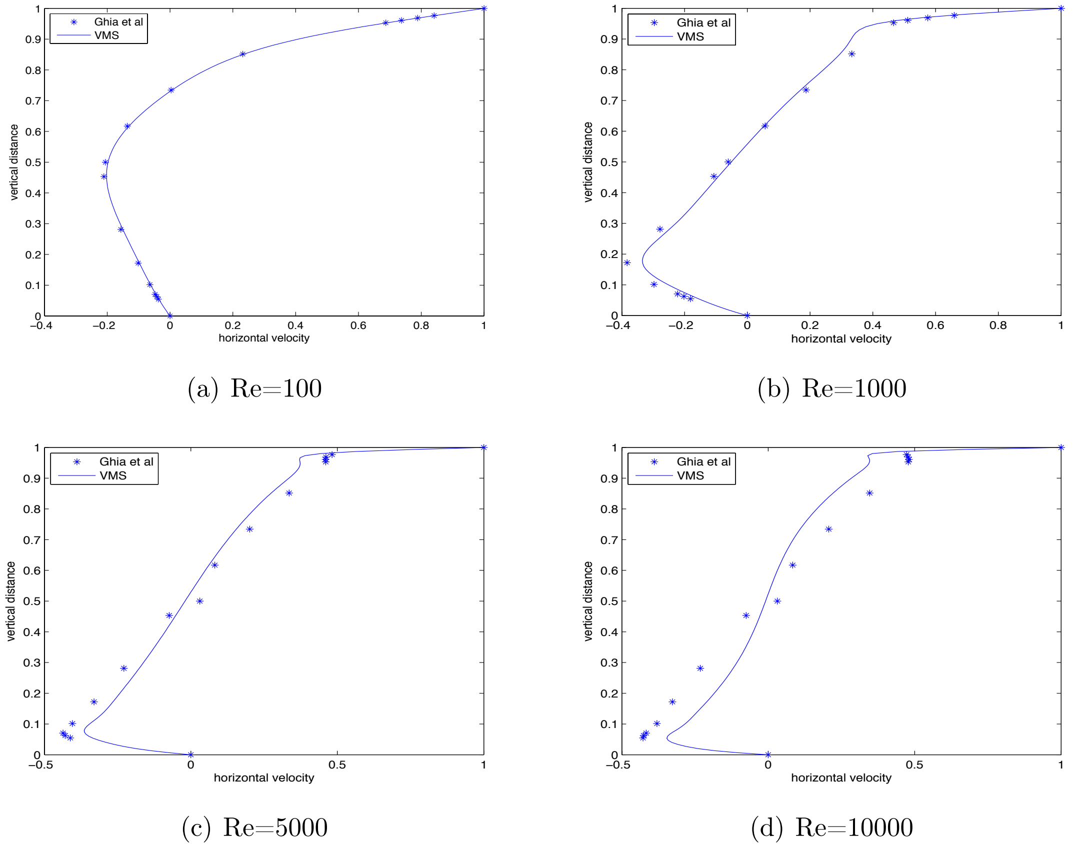
(b) Re=1000

(c) Re=5000

(d) Re=10000

Figure 6.2: Flow in a lid-driven cavity. Comparison with Ghia et al. of u_y velocity distribution along the vertical line.

In the Fig.6.2 and Fig.6.3 the velocity distributions at various Reynolds numbers are compared with the benchmark solution by Ghia et al.(1982). The horizontal velocity u_x components along the mid-vertical line are compared in Fig.6.2 and the vertical



(a) Re=100

(b) Re=1000

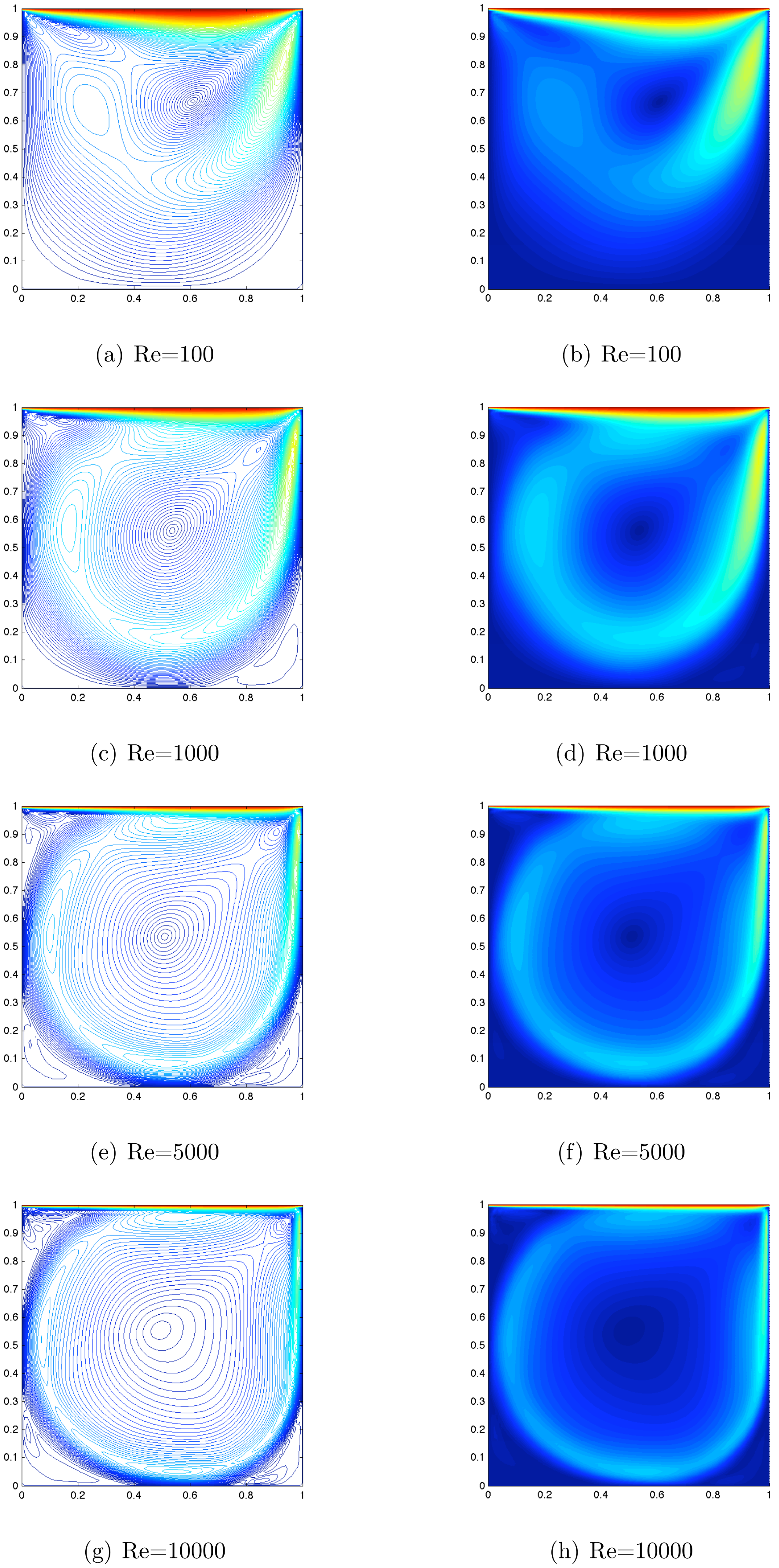
(c) Re=5000

(d) Re=10000

Figure 6.3: Flow in a lid-driven cavity. Comparison with Ghia et al. of u_x velocity distribution along the vertical line.

velocity u_y components along the mid-height line are compared in Fig 6.2.

At lower Reynolds numbers the comparison is excellent. However, at $Re = 5000$ and $Re = 10000$ small deviations close to peaks are noticed. This is mainly due to the coarseness of the mesh. Because the structured mesh is used, we need more elements to get a better solution especially finer mesh near the boundary.



(a) Re=100

(b) Re=100

(c) Re=1000

(d) Re=1000

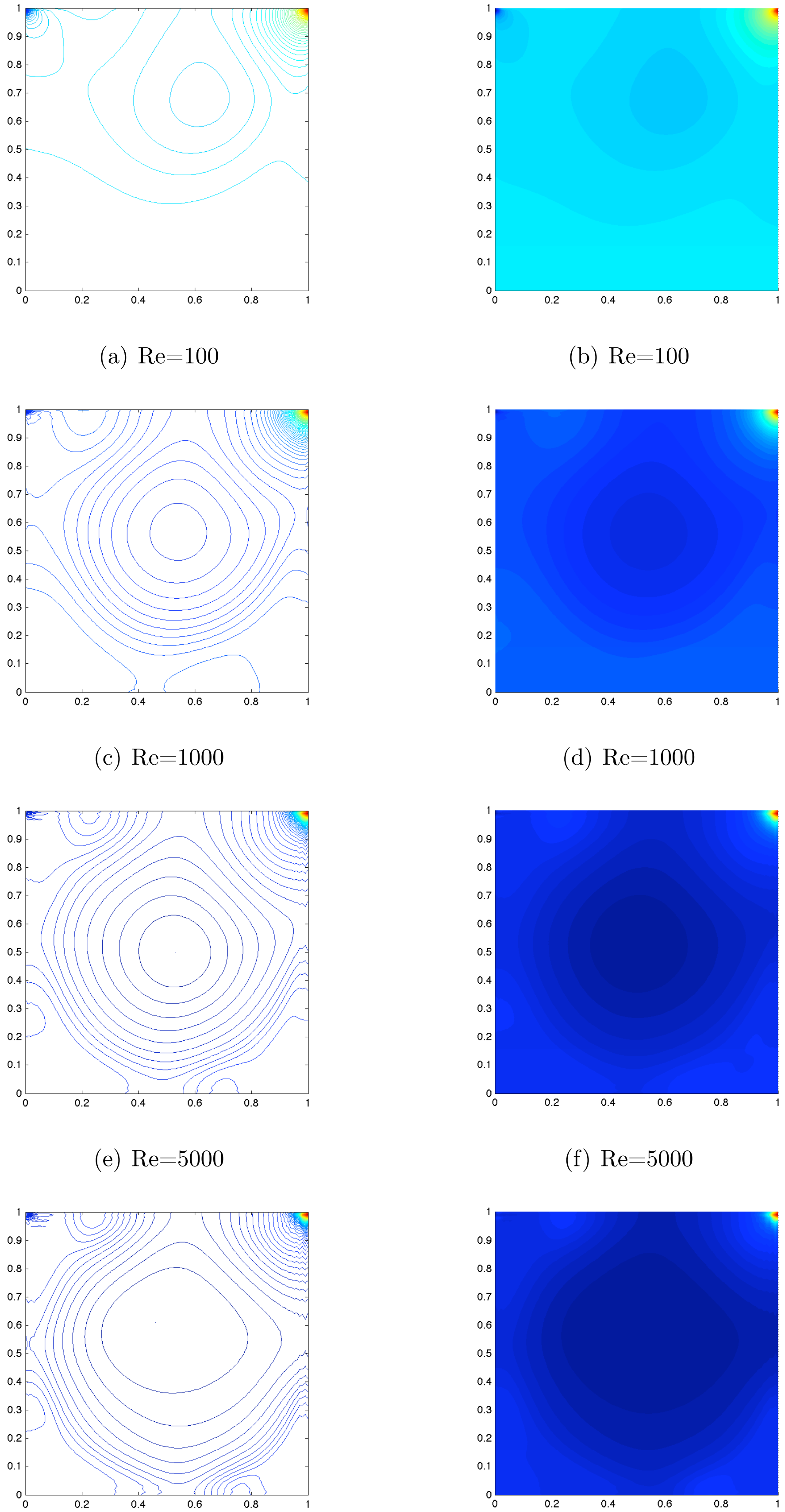
(e) Re=5000

(f) Re=5000

(g) Re=10000

(h) Re=10000

Figure 6.4: Laminar cavity velocity at different Reynolds number



(a) Re=100

(b) Re=100

(c) Re=1000

(d) Re=1000

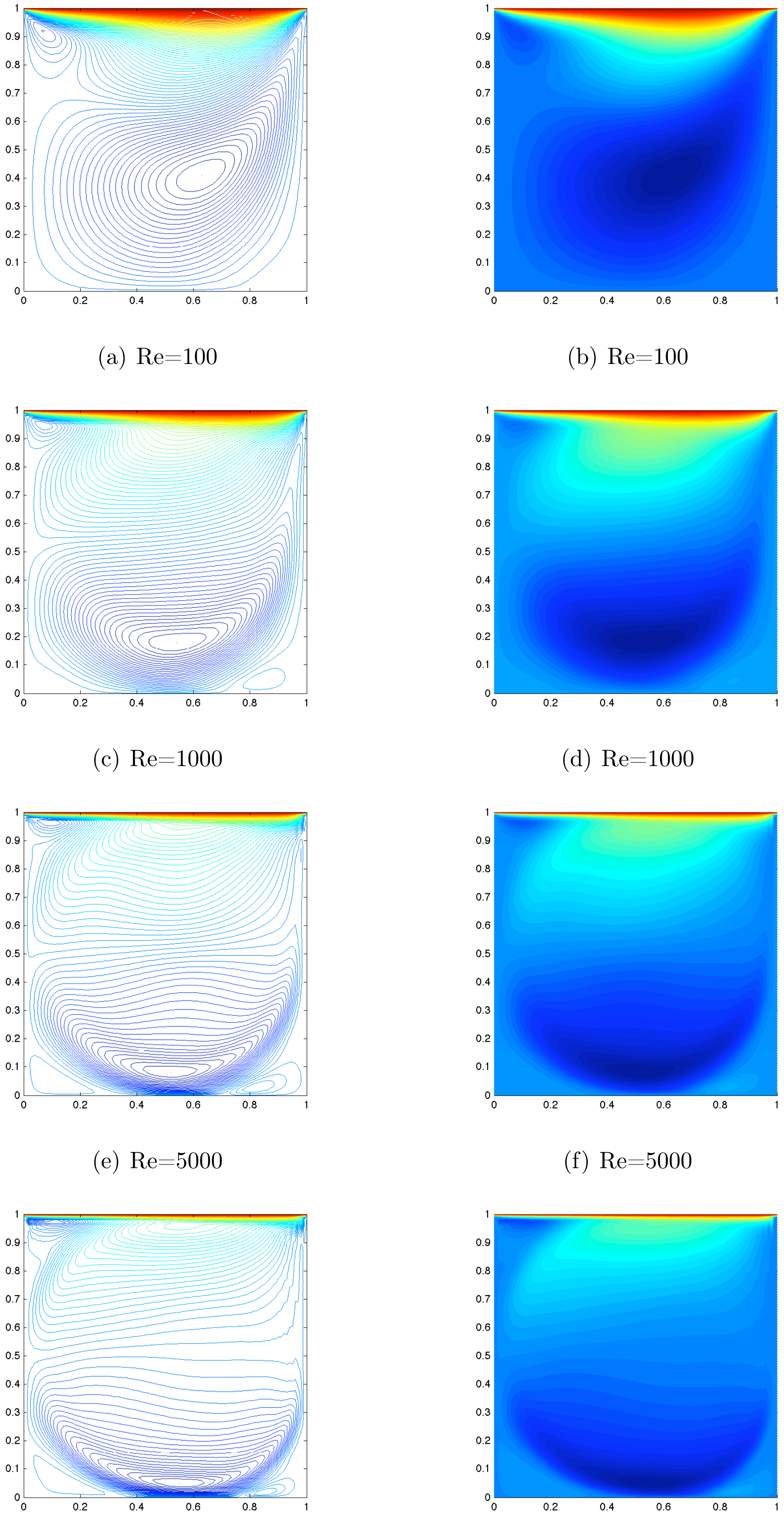
(e) Re=5000

(f) Re=5000

(g) Re=10000

(h) Re=10000

Figure 6.5: Laminar cavity pressure at different Reynolds number



(a) Re=100

(b) Re=100

(c) Re=1000

(d) Re=1000

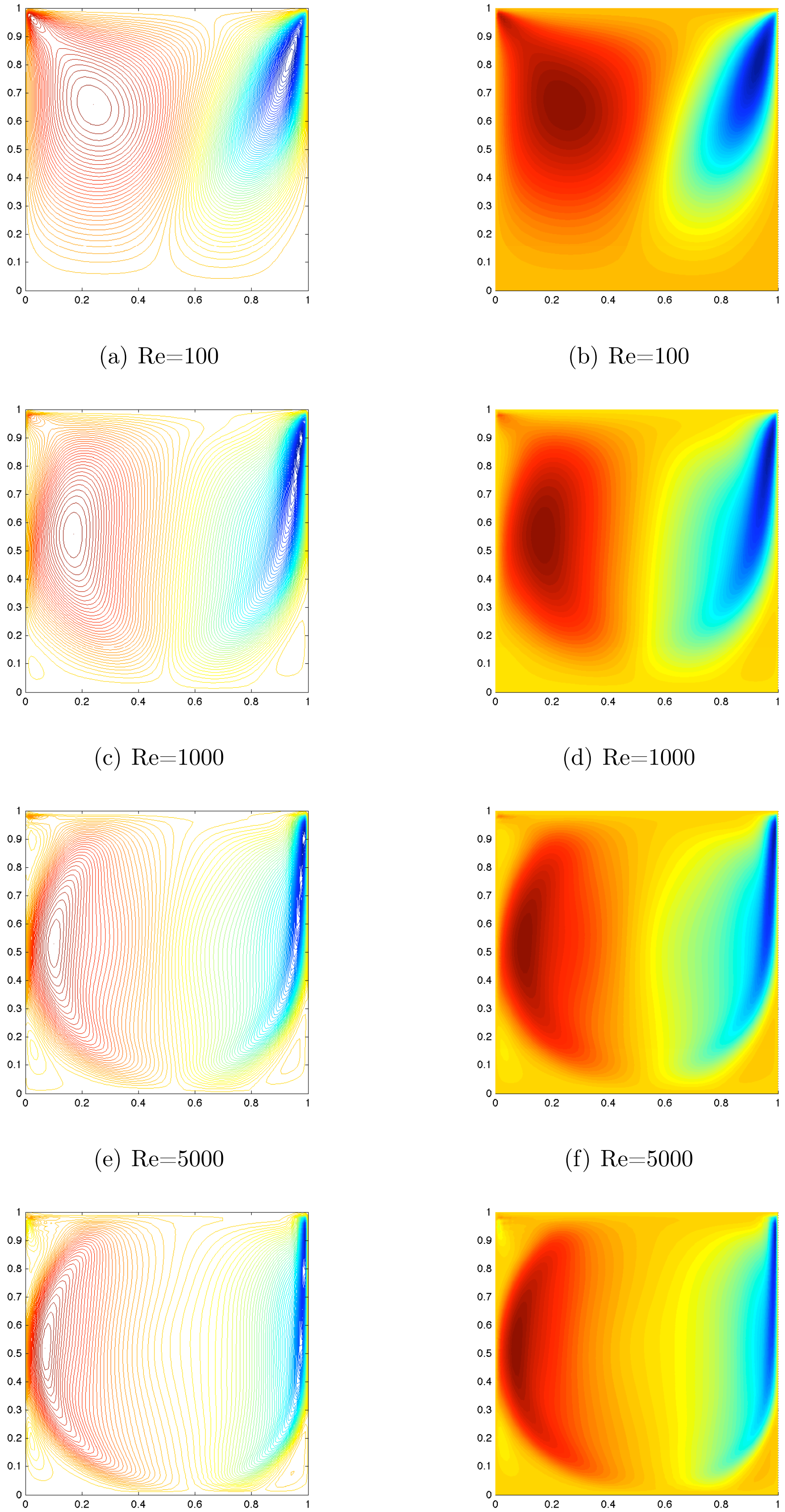
(e) Re=5000

(f) Re=5000

(g) Re=10000

(h) Re=10000

Figure 6.6: Laminar cavity velocity x-componet at different Reynolds number



(a) Re=100

(b) Re=100

(c) Re=1000

(d) Re=1000

(e) Re=5000

(f) Re=5000

(g) Re=10000

(h) Re=10000

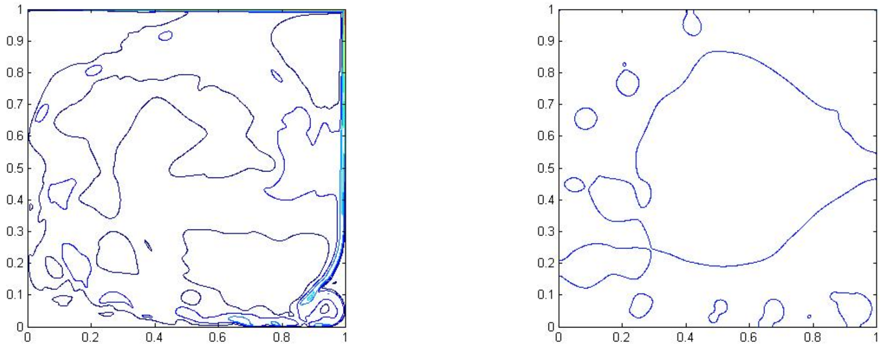
Figure 6.7: Laminar cavity velocity y-componet at different Reynolds number

6.2 Lid-driven cavity in 2D (high Reynolds flows)

This numerical experiment gives transient solutions of the 2D lid-driven cavity flow at Reynolds number $Re = 100000$. Similar to the first example, bilinear quadrilateral element and OSS are used, but a larger Reynolds number is chosen. In order to avoid the instability, 512×512 bilinear quadrilateral element is applied.

The aim of this numerical experiment is not to give benchmark result but to study the qualitative behavior of the solution. Unlike the previous case, the 2D lid-driven cavity flow in $Re = 100000$ would not leads to steady state.

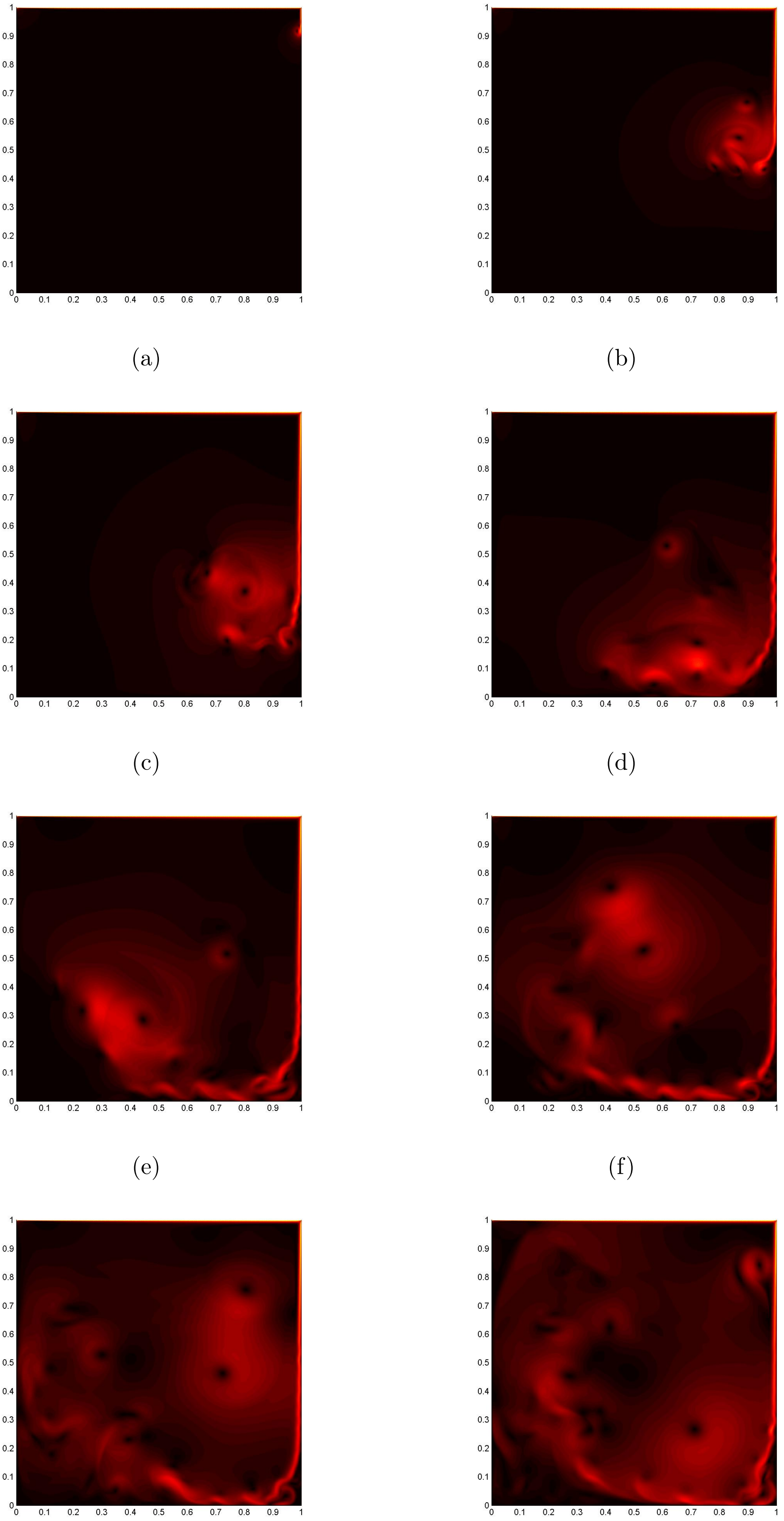
Fig. 6.8 shows the velocity and pressure contour view at $t = 50$s. A varying number of vortices could be noticed inside the cavity.



(a) velocity contour view (b) pressure contour view

Figure 6.8: Velocity and pressure contour view at Re=100000 in $t = 1$s

Figure 6.9 shows the evolution of velocity at Re=100000 in each time step.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

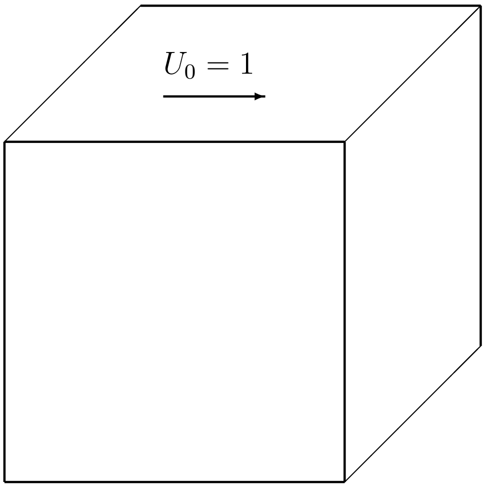
Figure 6.9: Evolution of cavity flow at Re=100000

6.3 Lid-driven cavity in 3D

Under the explicit VMS method, the extension of the scheme to three dimensions is straightforward. This section presents results of three-dimensional driven cavity flow simulations for Reynolds number of $Re = 100$ and $Re = 1000$. This is an interesting flow, with a complex behavior in a simple geometry. The structured mesh is employed to calculate this problem.

6.3.1 Problem statement

Fig 6.3.1 shows the cubical cavity. The flow motion is induced by the top lid that moves in the x-direction with a constant unit velocity $U_0 = 1$.



6.3.2 Numerical result

The simulation results of the lid-driven cavity flow in 3D for Reynolds number $Re = 100$ and $Re = 1000$ are shown. Fig. 6.10 and Fig. 6.11 show velocity isosurfaces at $V = 0.1$ and the velocity streamlines. The post-processing is performed in Gid for convenience.

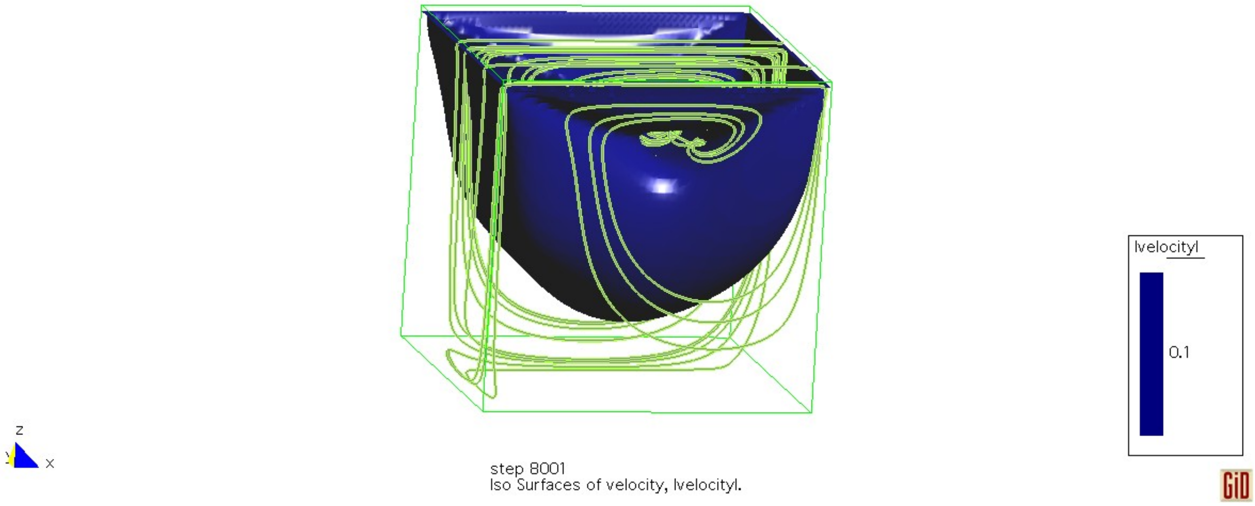


Figure 6.10: Vorticity isosurfaces at $V = 0.1$ and velocity streamlines in 100^3 mesh at $Re = 100$ in 3D cavity flow

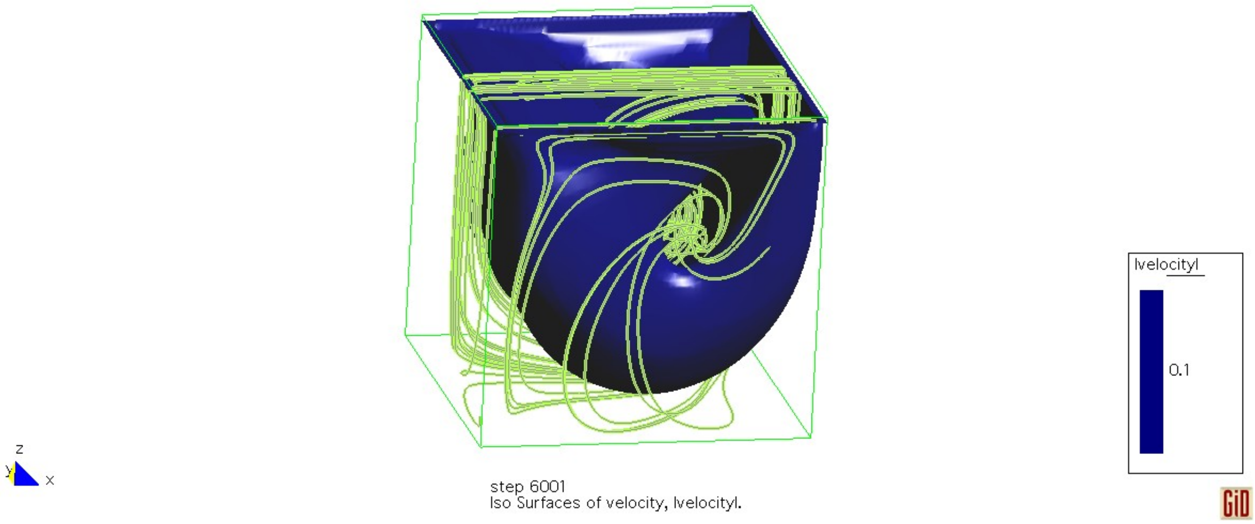


Figure 6.11: Vorticity isosurfaces at $V = 0.1$ and velocity streamlines in 100^3 mesh at $Re = 1000$ in 3D cavity flow

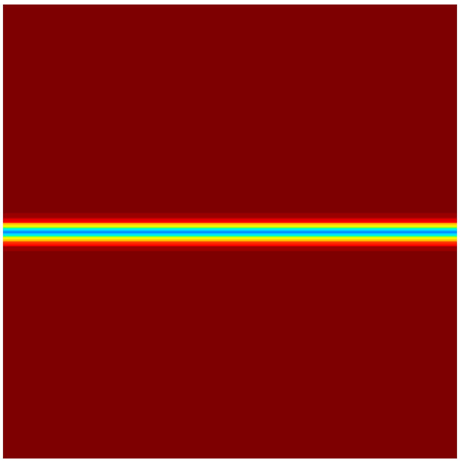
6.4 Plane mixing layer in 2D

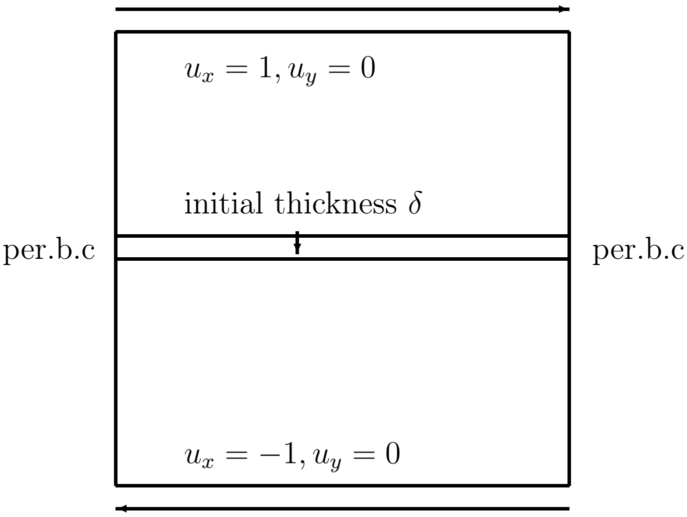
Mixing layers are encountered in aerodynamics, in the atmosphere or the ocean. It is a flow developing far away from boundaries, and the influence of boundaries is eliminated in this example. This 2D case would not lead to what is usually perceived as ’turbulence’, but it shows some features of the flow considering it to be ’turbulent in a certain sense’ (Gravemeier, 2003).

For example, the flow is extremely sensitive to the initial condition. It would be completely unpredictable in an infinite domain. Besides, the flow shows the typical turbulent eddy evolution.

6.4.1 Problem description

The domain is defined to be $\Omega = [0,1] \times [0,1]$, see Fig 6.4.1, with periodic boundary condition at the boundaries $x_1 = 0$ and as well as $x_1 = 1$





A initial velocity field is given by a hyperbolic tangent profile

$$u_x = u_{\max} \tanh(\frac{2y-1}{\delta_0}) \tag{6.2}$$

where δ_0 denotes the initial vortex thickness.

Boersma el al(1997) chose δ_0 to be 1/28. The small perturbation in the initial condition is expected to be amplified by Kelvin-Helmholtz instabilities during the evolution of the flow. The velocity component u_y is chosen by

$$u_y = u_{noise} \sin(8\pi x) \tag{6.3}$$

This problem has been investigated experimentally, for instance, in Brown and Roshko (1974) as well as Winant and Browand (1974). Numerically, it has been analyzed in a quasi-DNS in finite difference method by Lesieur (1988). The corresponding Implicit VMS method for this case is studied by (Volk., 2006).

6.4.2 Numerical result

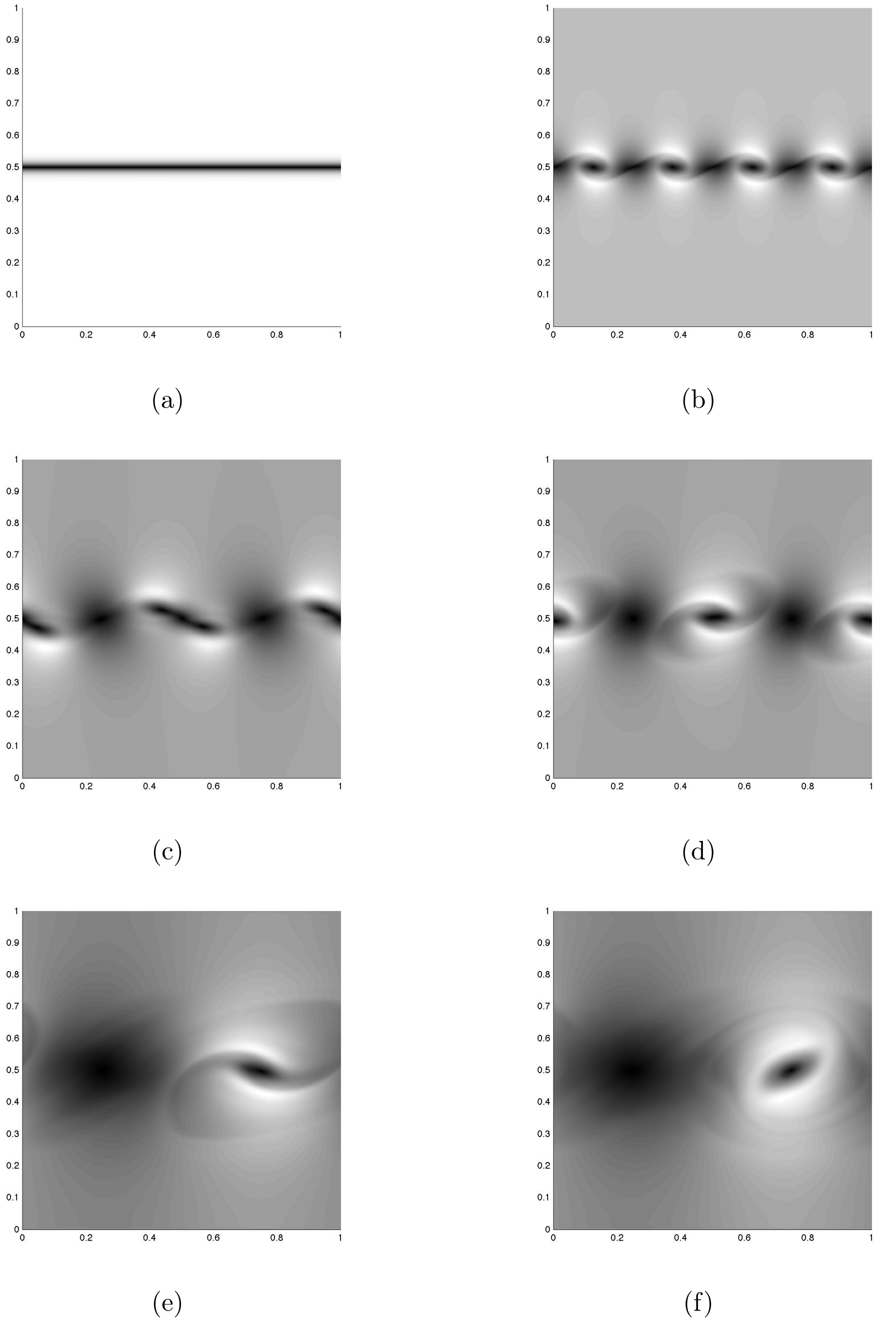
In the described case, Reynolds number is taken to be $Re = 10000$, basic discretization with 240×240 mesh, bilinear quadrilateral elements of uniform length have been chosen and OSS method is applied. The physical evolution of the flow wil be explained with the help of Fig. 6.12. This figure illustrates the evolution of the eddy paring and four different stages of this flow could be easily distinguished:

- **Development of four primary eddies**

- **First pairing**

- **Second pairing**

- **Rotation of the final vortex**



(a)

(b)

(c)

(d)

(e)

(f)

Figure 6.12: Eddy pairing

6.5 Decaying of isotropic turbulence in 2D

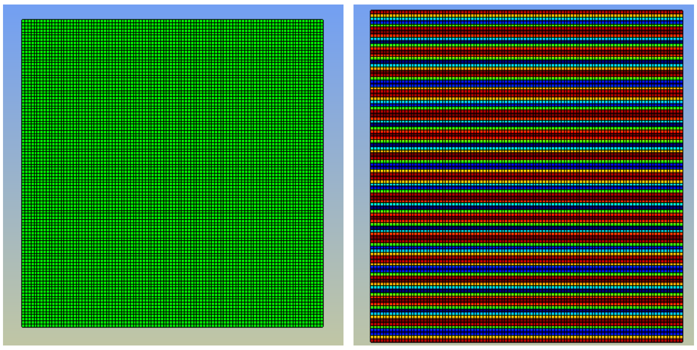
Two dimensional turbulence is very important in some area of physics and useful to test ideas and theories regarding some aspects of three-dimensional turbulence. However, there is an essential difference between two-dimensional and three dimensional turbulence: the lack of vortex stretching in the former. A broadband energy spectrum $E(k)$ of slope between K^{-4} and K^{-3} is developed.

6.5.1 Problem description

The domain of the problem is defined to be $\Omega = [0,1] \times [0,1]$, with periodic boundary condition on $\partial\Omega$ The initial velocity is given

$$u = [\cos(16\pi x), 0] \tag{6.4}$$

The 2D mesh and initial field are shown in Fig. 6.13

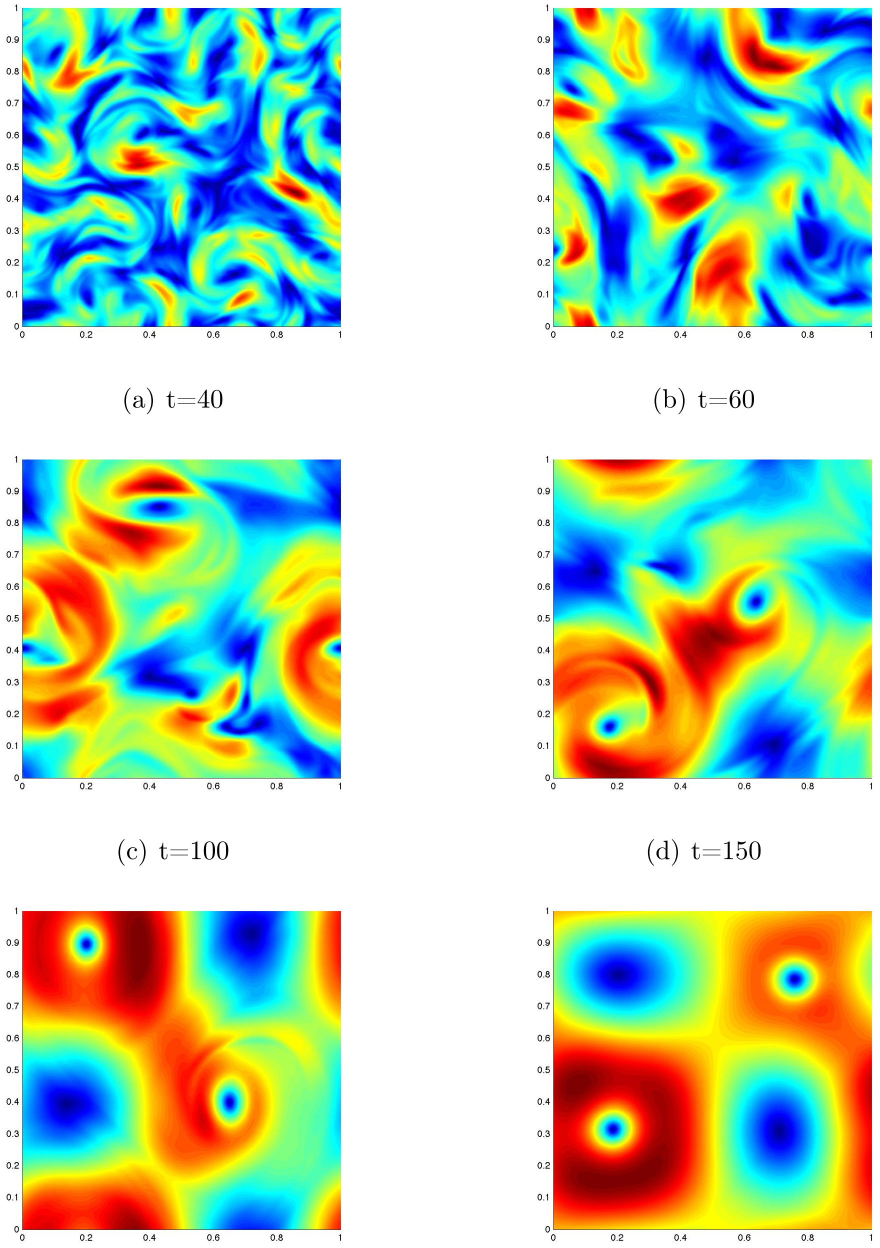


(a) Mesh 240 × 240 (b) Initial velocity

Figure 6.13: The 240×240 mesh and Initial velocity field for the decaying of isotropic turbulence in 2D

6.5.2 Numerical result and spectra energy analysis

Figure 6.14 shows the evolution of isotropic 2D turbulence. At time $t = 40$s, the turbulence is fully developed.



(a) t=40

(b) t=60

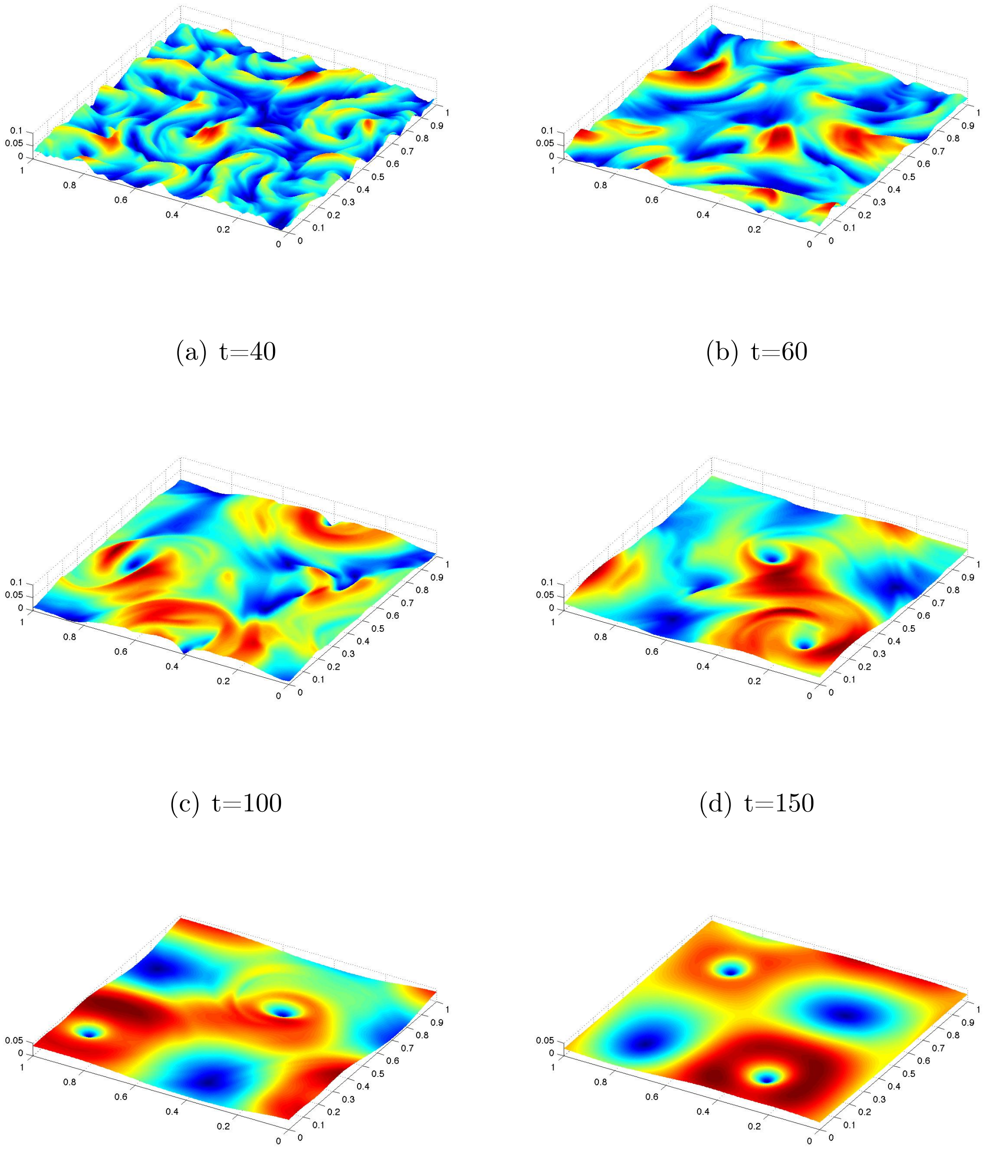
(c) t=100

(d) t=150

(e) t=250

(f) t=500

Figure 6.14: Vorticity modulus field at different time steps. 240 × 240 mesh and the OSS finite element method



(a) t=40

(b) t=60

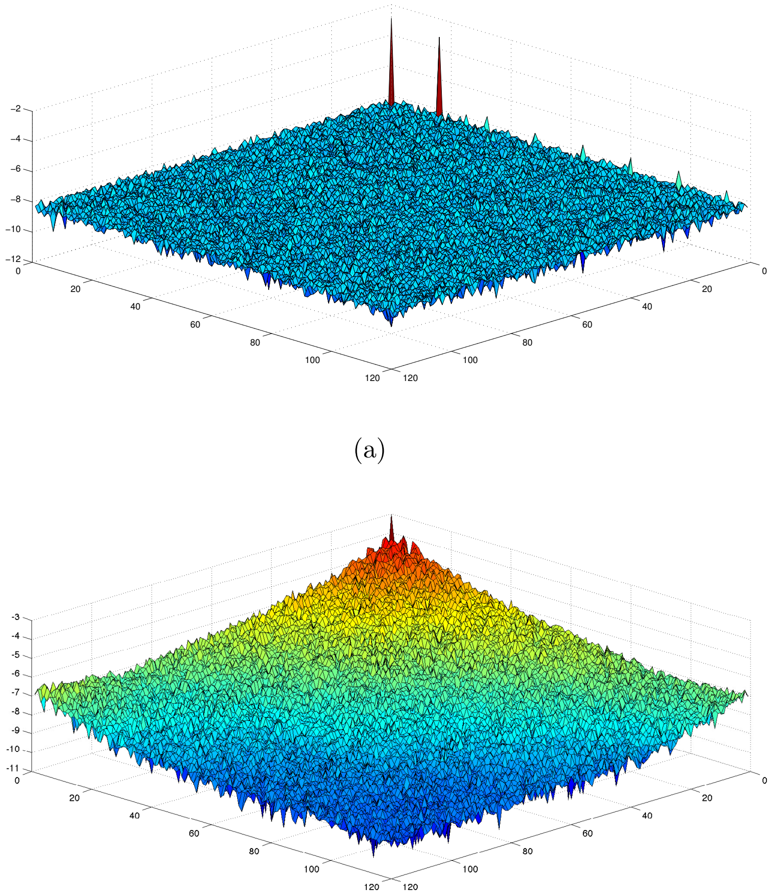
(c) t=100

(d) t=150

(e) t=250

(f) t=500

Figure 6.15: Vorticity surfaces at different time steps. These results are those of Fig 6.14 but showing elevation surfaces.



(a)

(b)

Figure 6.16: Energy spectra $E(k_x, k_y)$ at different time steps. a) Initial condition $t = 0$ s, b) $t = 40$ s.

In Fig. 6.16(a) we plot the energy spectrum $E(K_x, K_y)$ for the initial condition($t = 0$ s). Obviously only the wavenumber (K_x, K_y) has a non-null value. In Fig. 6.16(b) we have plotted $E(K_x, K_y)$ for ($t = 40$s), which corresponds to the velocity field whose vorticity has been plotted in fig 6.14. As seen, almost all wavenumber have been excited by this time.

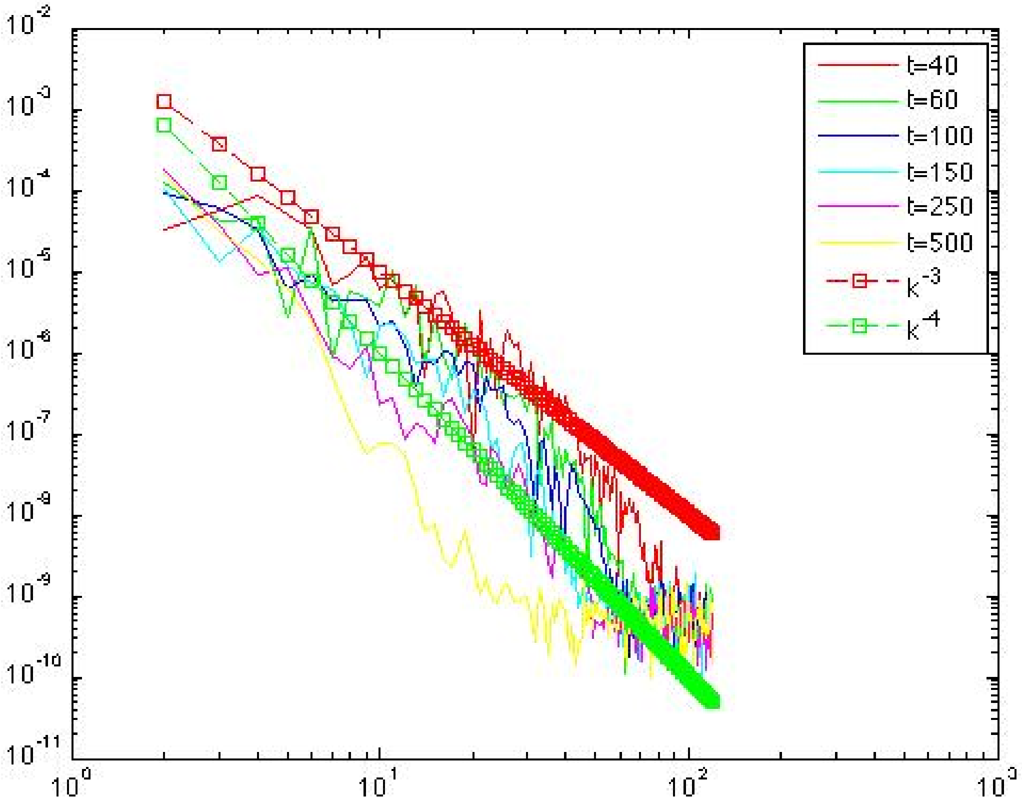


Figure 6.17: Energy spectra E (k) for various time steps. 240 × 240 mesh using the OSS finite element method.

The spectrum of turbulent velocity fluctuations shows how the turbulent energy is distributed over various scales, with their size being the inverse of the wave number k. In Fig 6.17 we plot the energy spectrum $E(k)$ for different time steps together with the k^{-3} and k^{-4} slopes. The result of this figure have been obtained using the OSS finite element method with a mesh of 240 × 240 elements.

6.6 Decaying of isotropic turbulence in 3D

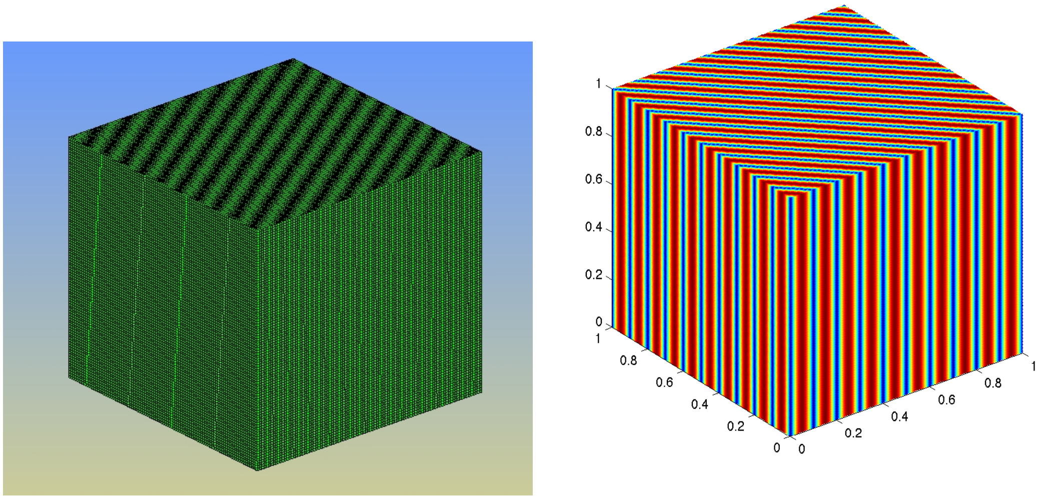
Turbulence in an incompressible fluid obeying the NS equation under periodic boundary conditions may be regarded as one of the simplest systems that retains the essence of the three-dimensional turbulence dynamics. In this section, the decaying of isotropic turbulence in 3D is studied. Energy spectrum in the middle line is given to approximate the well-known $-5/3$ slope.

6.6.1 Problem description

The domain of the problem is defined to be $\Omega = [0,1] \times [0,1] \times [0,1]$, with periodic boundary condition on $\partial\Omega$. The initial velocity is given by

$$u = [\cos(16\pi x), 0, 0] \tag{6.5}$$

The 3D mesh and initial field are shown in Fig. 6.18



(a) Mesh 100^3

(b) Initial velocity

Figure 6.18: The 3D mesh and Initial velocity field for the decaying of isotropic turbulence

6.6.2 Numerical result and spectra energy analysis

The energy spectra distribution at the center line of the cubic are shown in Fig. 6.19. It is observed that the $-5/3$ slope is well approximated without using any LES model. The result have been obtained using the OSS finite element method with a mesh of 100^3 elements.

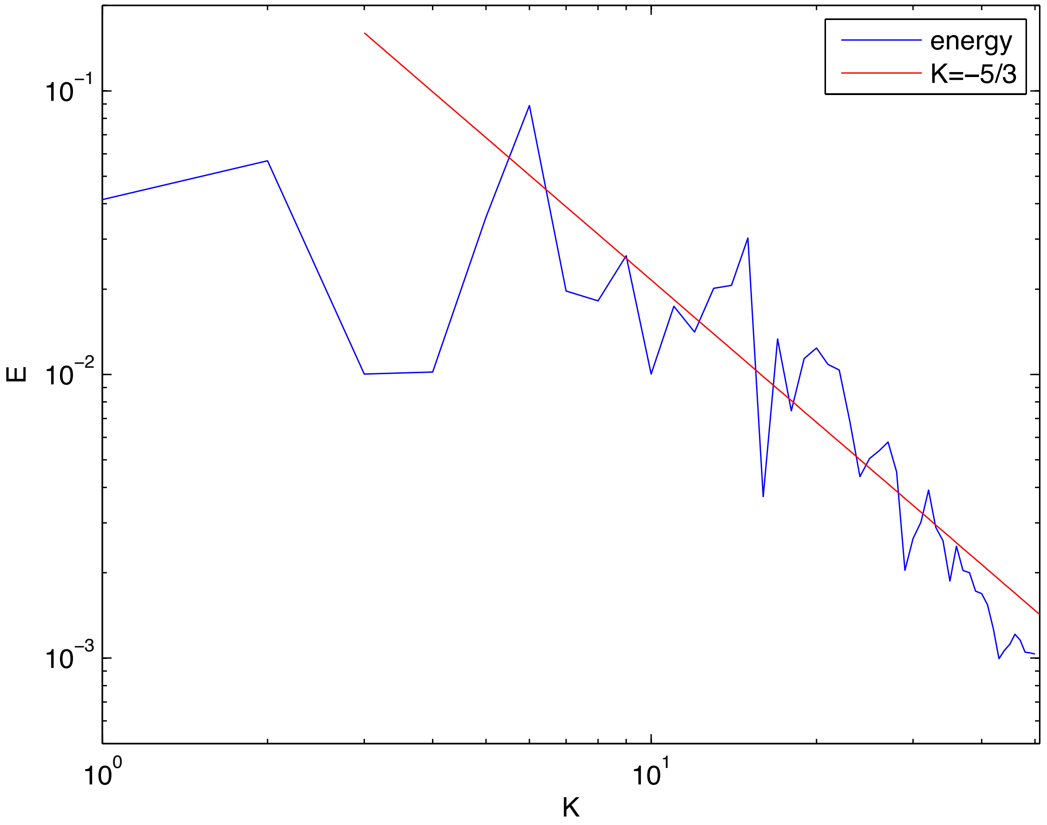


Figure 6.19: The energy spectrum for the decaying of isotropic turbulence in 3D

Fig. 6.20 shows snapshot of vorticity isosurfaces and velocity streamlines computed on 100^3 mesh. The visualizations are performed using MATLAB.

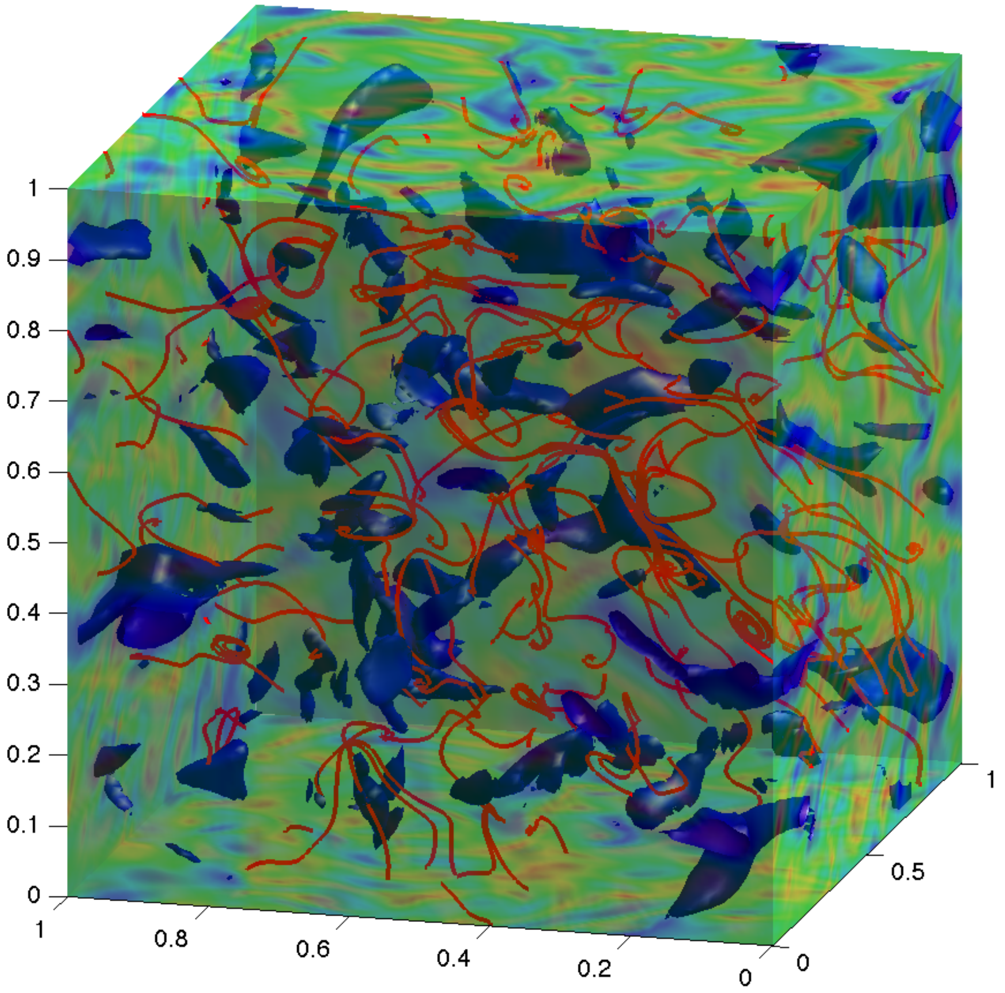


Figure 6.20: Vorticity isosurfaces, velocity streamlines, and vorticity contours plotted on the entire computational domain.

Appendices

Appendix A

Code implementation

Here I list some important equations which are used in code implementations in 2D.

∇N_{ig}^{a} is defined on the gauss point ig and node a.

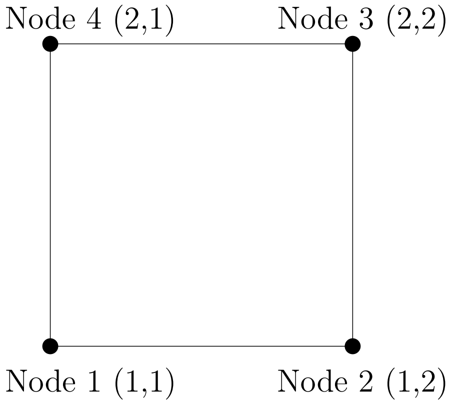
$$\nabla N_1^1 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \nabla N_2^1 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \nabla N_3^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \nabla N_4^1 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$\nabla N_1^2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \nabla N_2^2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \nabla N_3^2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \nabla N_4^2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\nabla N_1^3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \nabla N_2^3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \nabla N_3^3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \nabla N_4^3 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\nabla N_1^4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \nabla N_2^4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \nabla N_3^4 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \nabla N_4^4 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

For the linear square element, we have the relation:



$$\sum_{ig=1}^{4} \partial_j N_{ig}^{a} = \partial_j N_a^a + \partial_j N_{tw(a,j)}^{a}$$

$$\sum_{a=1}^{4} \partial_j N_{ig}^{a} = \partial_j N_{ig}^{ig} + \partial_j N_{ig}^{tw(ig,j)}$$

$$\partial_j N_{tw(a,j)}^{a} = \partial_j N_a^a$$

$$\partial_j N_{ig}^{tw(ig,j)} = -\partial_j N_{ig}^{tw(ig,ig)}$$

$$\begin{aligned}
&\nabla u_h \\
&= \sum_{a=1}^{4} \partial_j N_{ig}^{a} u^{a,i} \\
&= \partial_j N_{ig}^{ig} u^{ig,i} + \partial_j N_{ig}^{tw(ig,j)} u^{w(ig,j),i} \\
&= \partial_j N_{ig}^{ig} (u^{ig,i} - u^{w(ig,j),i})
\end{aligned}$$

$$\begin{aligned}
&(\nabla u_h, \nabla v_h) \\
&= \sum_{ig=1}^{4} \sum_{j=1}^{2} (\partial_j u_h)_{ig}^{i} \partial_j N_{ig}^{a} \left|J\right| w_{ig} \\
&= \sum_{j=1}^{2} \left[(\partial_j u_h)_a \nabla N_a^a + (\partial_j u_h)_{tw(a,j)}^{i} \partial_j N_{tw(a,j)}^{a} \right] \left|J\right| w_a \\
&= \sum_{j=1}^{2} \partial_j N_a^a \left[(\partial_j u_h)_{ig}^{i} + (\partial_j u_h)_{tw(a,j)}^{i} \right] \left|J\right| w_a
\end{aligned}$$

$$\left(\nabla^T u_h, \nabla v_h\right)$$
$$= \sum_{ig=1}^{4} \sum_{j=1}^{2} (\partial_i u_h)^j_{ig} \partial_j N^a_{ig} \left|J\right| w_{ig}$$
$$= \sum_{j=1}^{2} \left[(\partial_i u_h)^j_a \nabla N^a_a + (\partial_i u_h)^j_{tw(a,j)} \partial_j N^a_{tw(a,j)} \right] \left|J\right| w_a$$
$$= \sum_{j=1}^{2} \partial_j N^a_a \left[(\partial_i u_h)^j_{ig} + (\partial_i u_h)^j_{tw(a,j)} \right] \left|J\right| w_a$$

$$a \cdot \nabla u_h$$
$$= \sum_{j=1}^{2} \sum_{a=1}^{4} a_{ig,j} \partial_j N^a_{ig} u^{a,i}$$
$$= \sum_{j=1}^{2} a_{ig,j} \partial_j N^{ig}_{ig} u^{ig,i} + a_{ig,j} \partial_j N^{tw(ig,j)}_{ig} u^{tw(ig,j),i}$$
$$= \sum_{j=1}^{2} a_{ig,j} \partial_j N^a_a \left(u^{ig,i} + u^{tw(ig,j),i} \right)$$

$$(a \cdot \nabla u_h, a \cdot \nabla v_h)$$
$$= \sum_{j=1}^{2} \sum_{ig=1}^{4} \left(a \cdot \nabla u_h\right)_{ig,i} a_{ig,j} \partial_j N^a_{ig} \left|J\right| w_{ig}$$
$$= \sum_{j=1}^{2} \left[\left(a \cdot \nabla u_h\right)_{a,i} a_{a,j} \partial_j N^a_a + \left(a \cdot \nabla u_h\right)_{tw(a,j),i} a_{tw(a,j),j} \partial_j N^a_{tw(a,j)} \right] \left|J\right| w_a$$
$$= \sum_{j=1}^{2} \left[\left(a \cdot \nabla u_h\right)_{a,i} a_{a,j} + \left(a \cdot \nabla u_h\right)_{tw(a,j),i} a_{tw(a,j),j} \right] \partial_j N^a_a \left|J\right| w_a$$

$$\nabla p_h$$
$$= \sum_{a=1}^{4} \partial_j N^a_{ig} p^a$$
$$= \partial_j N^{ig}_{ig} p^{ig} + \partial_j N^{tw(ig,j)}_{ig} p^{tw(ig,j)}$$
$$= \partial_j N^{ig}_{ig} (p^{ig} - p^{tw(ig,j)})$$

$$(\nabla p_h, v_h)$$
$$= \partial_i N^a_a (p^a - p^{tw(a,i)}) \left|J\right| w_a$$

$$(u_h, \nabla q_h)$$
$$= \sum_{j=1}^{2} \sum_{ig=1}^{4} u^i_{ig} \partial_j N^a_{ig} \left|J\right| w_{ig}$$
$$= \sum_{j=1}^{2} \partial_j N^a_a \left(u^j_a + u^j_{w(a,j)} \right) \left|J\right| w_a$$

$$
\begin{aligned}
&(\nabla p_h, \nabla q_h) \\
&= \sum_{j=1}^{2} \sum_{ig=1}^{4} (\nabla p_h)_{ig,j} \, \partial_j N_{ig}^a \, |J| \, w_{ig} \\
&= \sum_{j=1}^{2} \sum_{ig=1}^{4} \partial_j N_a^a \left((\nabla p_h)_{ig,j} + (\nabla p_h)_{tw(ig,j),j} \right) |J| \, w_a
\end{aligned}
$$

Appendix B

A brief survey of computational tools used

B.1 Matlab code

In chapter 3, the implicit Matlab (Matrix form) code is originated from Dr. Esther Sala Lardies and is modified to test the AC scheme.

In chapter 2,4, the vector form explicit code is written in MATLAB.

B.2 Fortran code

Considering the computing efficiency, the algorithm introduced in Chapter 5 is based on the programming language FORTRAN and generated the result for chapter 6.

B.3 Postprocessing

The most important aspect of the postprocessing stage consists in the visual presentation of the achieved data. For this purpose, the fortran generated result file are readable by MATLAB and Gid.

Bibliography

Babuska, I. (1970/71). Error-bounds for finite element method. *Numer. Math. 16*(322-333).

Brezzi, F. (1974). On the existence, uniqueness and approximation of saddle point problems arising from lagrange multipliers. *RAIRO Analyse Numerique 8*(129–151).

Chorin, A. J. (1968). Numerical solution of the navier-stokes equations. *Math. Comp.* (22), 745–762.

Codina, R. (2000). Stabilization of incompressibility and convection through orthogonal sub-scales in ®nite element methods. *Comput. Methods Appl. Mech.*.

Codina, R. (2002). Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *COMPUTER METHODS IN APPLIED MECHANICS AND EN 191*(39-40), 4295–4321.

Codina, R., J. Principe, O. Guasch, and S. Badia (2007). Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *COMPUTER METHODS IN APPLIED MECHANICS AND EN 196*(21-24), 2413–2430.

Donea, A. H. (2003). *Finite Element Methods for Flow Problems.* John Wiley Sons, Ltd.

Gravemeier, V. (2003). *The variational multiscale method for laminar and turbulent incompressible flow.* Ph. D. thesis, Institut für Baustatik der Universität Stuttgart.

Hughes, T., V. Calo, and G. Scovazzi (2000). Variational and multiscale methods in turbulence. *In Proceedings of the XXI International Congress of Theoretical and Applied Mechanics (IUTAM). Kluwer*.

Ladyzhenskaya, O. (1968). *The Mathematical Theory of Viscous Incompressible Flow.* Gordon and Breach.

Nithiarasu, P. (2003). An efficient artificial compressibility (ac) scheme based on the characteristic based split (cbs) method for incompressible flows. *International Journal for Numerical Methods in Engineering 56*(13), 1815–1845.

Peyret, R. and T. D. Taylor (1983). *Computational Methods for Fluid Flow. Springer Series in Computational Physics*. Spriver.

S. Badia, R. C. (2008). On a multiscale approach to the transient stokes problem: Dynamic subscales and anisotropic space–time discretization. *Applied Mathematics and Computation*.

Volk., G. (2006). The variational multiscale method for laminar and turbulent flow. *Archives of Computational Mechanics—State of the Art Reviews*.

Wilcox, D. C. (2006). *Turbulence Modeling for CFD*. Dcw Industries.