ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

DIRECT NUMERICAL SIMULATION OF TURBULENT FLOWS USING AN IMPLICIT FINITE VOLUME ALGORITHM

Ph.D. THESIS

İlyas YILMAZ

Department of Aeronautics and Astronautics Engineering

Aeronautics and Astronautics Engineering Graduate Programme

NOVEMBER 2012

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

DIRECT NUMERICAL SIMULATION OF TURBULENT FLOWS USING AN IMPLICIT FINITE VOLUME ALGORITHM

Ph.D. THESIS

İlyas YILMAZ
(511032012)

Department of Aeronautics and Astronautics Engineering

Aeronautics and Astronautics Engineering Graduate Programme

Thesis Advisor: Prof. Dr. Fırat Oğuz EDİS
Thesis Co-advisor : Prof. Dr. Hasan SAYGIN

NOVEMBER 2012

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

TÜRBÜLANSLI AKIŞLARIN KAPALI SONLU HACİM ALGORİTMASI İLE DOĞRUDAN SAYISAL BENZETİMİ

DOKTORA TEZİ

İlyas YILMAZ
(511032012)

Uçak ve Uzay Mühendisliği Anabilim Dalı

Uçak ve Uzay Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Fırat Oğuz EDİS
Tez Eş-danışmanı: Prof. Dr. Hasan SAYGIN

KASIM 2012

İlyas YILMAZ, a **Ph.D.** student of ITU **Graduate School of Science Engineering and Technology** student ID 511032012 successfully defended the **thesis** entitled **"DIRECT NUMERICAL SIMULATION OF TURBULENT FLOWS USING AN IMPLICIT FINITE VOLUME ALGORITHM"**, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Fırat Oğuz EDİS** Istanbul Technical University	....
Co-advisor :	**Prof. Dr. Hasan SAYGIN** Istanbul Aydın University	....
Jury Members :	**Prof. Dr. Mehmet Şerif KAVSAOĞLU** Anadolu University	....
	Prof. Dr. Seyhan Uygur ONBAŞIOĞLU Istanbul Technical University	....
	Prof. Dr. Aydın MISIRLIOĞLU Istanbul Technical University	....
	Doç. Dr. Metin MURADOĞLU Koç University	....
	Prof. Dr. A. Rüstem ASLAN Istanbul Technical University	....

Date of Submission : **08 April 2012**
Date of Defense : **26 November 2012**

To my wonderful parents Seher, Mehmet, Mustafa and my beloved wife Gülçin,

FOREWORD

First of all, I would like to express my deep gratitude to my advisor Prof. Fırat Oğuz EDİS for providing me the opportunity to continue with my PhD at this department. This study wouldn't be possible without his knowledge, endurance and leading.

I particularly want to thank to my co-advisor Prof. Hasan SAYGIN for his continuous academic and mental support. He has believed in me all the time since we first met. He also helped me to keep my research assistant position at the beginning of this study which was crucial.

One of the people who I have met and the most influenced during my PhD was Prof. Lars DAVIDSON from Chalmers. I spent a year in his group in the beautiful, cold city of Göteborg in Sweden. He has left a permanent effect on my life and provided me an extraordinary academic and social vision that will cover the rest of my life. His knowledge and the way of perception of life are really marvelous. I am deeply grateful to him.

What else I really understood in this study is that world would easily turn into an intolerable place without women. Especially the support from two of them are crucial and indispensable; my mother and my wife. Thank you my true-hearted mom for your continuous support and praying. And, of course, my beautiful wife Gülçin. She is a priceless gift given to me. Her beautiful and strong mind is always with me.

I should thank to an innocent sweet kitty, "peanut", has joined us a year ago. We loved her too much. She has a lot of fun and acts as a relaxation factor to us.

I also want to thank to the thesis committee for their valuable comments and to Istanbul Technical University for academic, technical and financial support.

November 2012

İlyas YILMAZ
(Astronomer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

AGARD : Advisory Group for Aerospace Research and Development
AMG : Algebraic Multi-grid Method
CFD : Computational Fluid Dynamics
DA : Distributed Array
DD : Domain Decomposition
DNS : Direct Numerical Simulation
EOS : Equation of State
FVM : Finite Volume Method
GMRES : Generalized Minimum Residual
HYPRE : Library for Solving Large, Sparse Linear Systems on Massively Parallel Computers
iDNS : Implicit Direct Numerical Simulation
ILU : Incomplete Lower-Upper
ILES : Implicit Large Eddy Simulation
KSP : Krylov Subspace
LES-URANS: Large Eddy Simulation-Unsteady Reynolds Averaging of Navier-Stokes Equations
MPI : Message Passing Interface
NSE : Navier-Stokes Equations
PC : Preconditioner
PDE : Partial Differential Equations
PETSc : Portable and Extensible Toolkit for Scientific Computations
RTI : Rayleigh-Taylor Instability
SPMD : Single Program Multiple Data
TGV : Taylor-Green Vortex
TSL : Turbulent Shear Layer
UMFPACK : Unsymmetric Multi-frontal Sparse LU Factorization Package
WENO : Weighted Essentially Non-oscillatory

LIST OF TABLES

LIST OF FIGURES

Page

LIST OF SYMBOLS

ρ	:	Density
t	:	Time
x_i	:	Cartesian coordinate vector
u_i	:	Velocity vector
σ_{ij}	:	Stress tensor
g_i	:	Gravity
e_T	:	Total energy per unit mass
e	:	Internal energy
q_j	:	Heat flux
δ_{ij}	:	Kronecker delta
τ_{ij}	:	Viscous stress tensor
μ	:	Dynamic viscosity coefficient
λ	:	Second viscosity coefficient
T	:	Temperature
k	:	Thermal conductivity coefficient
$(\cdot)^*$	:	Dimensional quantity
$(\cdot)_0$	:	Reference quantity
c_p, c_v	:	Specific heat constants
Pr	:	Prandtl number
R	:	Specific gas constant
h	:	Enthalpy
γ	:	Ratio of specific heats
p	:	Pressure
Re	:	Reynolds number
Fr	:	Froud number
Ec	:	Eckert number
M	:	Mach number
l	:	Length scale
V_n	:	Face-normal velocity
∇	:	Gradient
ϕ	:	Scalar
Σ	:	Summation
A	:	Area of cell face
V	:	Volume of cell
δ_p	:	Pressure difference
Δt	:	Time step
N	:	Face-normal vector
$\Vert\ \Vert$	:	$l_2 - norm$
τ	:	Characteristic time scale

κ	:	Wave number
Δ	:	Grid spacing
L	:	Length
ν	:	Kinematic viscosity coefficient
K	:	Total kinetic energy
$epsilon$	:	Dissipation
Re_λ	:	Reynolds number based on Taylor micro-scale
ζ	:	Ratio of grid spacing to Kolmogorov's scale
κ_{max}	:	Grid cut-off wave number
Q	:	Q-criterion for visualization
S_{ij}	:	Symmetric part of velocity gradient tensor
Ω_{ij}	:	Anti-symmetric part of velocity gradient tensor
tr	:	Trace
$< \cdot >$	:	Spatial average over all directions
$-dK/dt$	:	Mean dissipation rate
ω	:	Vorticity
$E(\kappa)$	:	Turbulent kinetic energy spectrum in wave number space
δ_θ	:	Momentum thickness
δ_ω	:	Vorticity thickness
M_c	:	Convective Mach number
c	:	Speed of sound
$\hat{u}^n, \psi^n, \sigma_i^n$	:	Amplitude, phase and direction of the n^{th} Fourier mode
$\varphi^n, \alpha^n, \theta^n$	:	Random angles
ψ^n	:	Random phase
$\underline{\kappa_0}$	:	Peak wave number
$\bar{\phi}$	:	Reynolds averaged variable
$\phi^{'}$	:	Reynolds fluctuation
$\tilde{\phi}$	:	Favre averaged variable
$\phi^{''}$	:	Favre fluctuation
$R_{ij}, P_{ij}, \varepsilon_{ij}, T_{ijk}, \Pi_{ij}$	:	Turbulent stresses, production, dissipation, transport and pressure-strain terms
b_{ij}	:	Anisotropy of Reynolds stresses
A	:	Atwood number
h_b, h_s	:	Penetration lengths for bubble and spike
α_b, α_s	:	Growth rate coefficients for bubble and spike
A_0	:	Amplitude
χ	:	Mole or mixture fraction
θ	:	Mixedness

DIRECT NUMERICAL SIMULATION OF TURBULENT FLOWS USING AN IMPLICIT FINITE VOLUME ALGORITHM

SUMMARY

This Ph.D. thesis study includes an application of all-speed Direct Numerical Simulation (DNS) algorithm to challenging problems of fluid dynamics such as transition to turbulence and hydrodynamic instabilities.

The algorithm is a fully implicit finite volume solution algorithm which solves compressible Navier-Stokes Equations (NSE). It has second-order accuracy in both space and time. It uses symmetric time and space discretization. The flow variables are co-located in space. In addition to this, the face-normal velocity is also defined at the center of the cell-faces in order to prevent the odd-even decoupling in space. To handle the problems including incompressible or low Mach number regimes, an incompressible scaling which yields suitable form of the equations depending on the Mach number is applied to the equations in non-dimensionalization step. At incompressible limit, the algorithm conserves kinetic energy discretely. It is also non-dissipative and robust at high Reynolds numbers. These features make it suitable for the problems including turbulent flow regime. To increase the convergence rate at low Mach number regime, an additional time fixing procedure was added to pressure.

In order to perform the study, an in-house flow solver has been developed based on the algorithm mentioned above. The solver, so-called *i*DNS, is an implicit DNS solver. No turbulence model is incorporated. It was written in Fortran. Since DNS requires high amount of computational resources, *i*DNS is completely parallelized using PETSc to be able to utilize the power of parallel programming in simulations. PETSc is a parallel library which can be used as a toolkit for building parallel programs. The large sparse linear systems arising from the discretization are solved by using either iterative Krylov subspace methods such as GMRES or Algebraic Multi-grid methods (AMG). Before solving the coefficient matrices, they are preconditioned by incomplete-LU (ILU(0)) method to obtain better eigenvalue distribution which makes the systems relatively easy to solve. All these methods are efficiently programmed and optimized in PETSc and available in parallel using the corresponding calls.

The primary objective of this thesis study is to perform further assessments of the algorithm by applying it to the problems including spatial and temporal multi-scale physics simultaneously such as hydrodynamic instabilities where laminar, transitional, and turbulent regimes are found together with varying Mach number regimes. For this purpose, the transition to turbulence problem in Taylor-Green Vortex (TGV), the velocity-induced mixing problem in Turbulent Shear Layer (TSL), and the gravity-induced mixing problem in Rayleigh-Taylor Instability (RTI) were chosen as test cases.

We also want to make contributions to the area of modeling incompressible and compressible turbulent flows using DNS by developing an in-house, parallel, implicit DNS solver and by presenting it as an open research tool. As it can be seen in the related chapters, the developed parallel solver is efficient and ready to use in various fluid flow problems.

TGV is the simplest fundamental case that has been used as a prototype for vortex stretching and the consequent production of small-scale eddies, resulting into homogeneous isotropic turbulence and its decay. It was shown that there are two stages in the evolution of the TGV flow. At early times, i.e. up to a non-dimensional time between 3 and 4, well-organized structures are formed and the flow remains essentially inviscid, mostly two-dimensional and is characterized as laminar. At later times, the effect of viscosity can no longer be neglected and highly distorted, dissipative structures develop. If the Reynolds number is large enough, this results into process of vortex-stretching and eventually into a breakdown of the large scales into smaller ones which is interpreted as turbulence. Therefore, TGV flow is assumed to describe the fundamental mechanism of transition into homogeneous isotropic turbulence. Obviously, this transitional behavior is determined entirely by the choice of the Reynolds number.

The results of our simulation show that, the described physics of TGV flow is well captured by the algorithm and the solver as well. The transition time is predicted for the viscous case and the kinetic energy is preserved during the course of long time simulation for the inviscid case, very well.

Turbulent Shear Layers are of interest in many physical and engineering flows such as jets, wakes and mixing layers. They are induced by strong gradient of shear stress. It is very important and is often necessary to understand the physical mechanisms behind such kind of flows, for example, in order to control the mixing processes in supersonic combustion or to control the flows emerging in downstream of an airfoil. Although their relatively simple configuration, more complex physical phenomena are included by these flows. They are mostly dominated by large scale quasi-2D organized structures. However, they can rapidly undergo mixing and transition to turbulence in 3D.

It can be studied both spatially and temporally. In spatially-evolving mixing layers, mixing layer thickness of two emerging stream of fluids develop in spatial coordinate in stream-wise direction. Such development requires large extend of domains. Since this is computationally much more expensive to track than the temporally-evolving mixing layers where the thickness of the layer increases as a function of time rather than as a function of the stream-wise coordinate, a temporally-evolving case is considered in our simulation to show the capability of the algorithm's ability to evolve a linear perturbation into nonlinear hydrodynamic turbulence. The simulation results are successfully compared to both experimental and numerical studies found in the literature. The effect of compressibility on the growth rate of the mixing layers is also analyzed and found that it is stabilizing.

Rayleigh-Taylor Instability occurs when a heavy fluid of density ρ_h on top is supported against the gravity g by a light fluid of density ρ_l on bottom. The fluid is initially in hydrostatic equilibrium. Pressure gradient is balanced by the gravity which acts only in vertical direction. In this case pressure and density gradients are in opposite direction and infinitesimal perturbations grow at the interface exponentially. Then, vorticity causes the fluids to interpenetrate as bubbles and spikes into each other and mix. It is observed in many physical and engineering flows such as type Ia supernovae in astrophysics, inertial confinement fusion, cavitation bubbles and oceanic and atmospheric flows. Rayleigh-Taylor instability also serves as a very challenging test case for algorithms, since it includes many aspects of flow modeling like body force treatment, turbulent mixing, diffusion and interface capturing.

Our results are in agreement with the prior studies, as well as the findings of linear theory. It was shown that the algorithm and the solver as well are able to handle the physics of the problems including multi-scales, transition and mixing simultaneously. The compressibility effects on the Rayleigh-Taylor mixing are also analyzed by modifying the pressure field and found that it increases the growth rate of the instability, while reducing the mixing efficiency.

Some further research topics are also suggested in the last chapter, regarding our experiences gained during this study. In order to make the algorithm, and the solver as well, applicable to high Reynolds number flows, a hybrid Large Eddy Simulation-Unsteady Reynolds Averaging of NSE (LES-URANS) procedure can be developed. Similarly, complex boundaries, other than walls, can be implemented. To handle the problems on boundary-fitted grids, the metrics can also be incorporated into the solver. It is also possible to extend the algorithm to magneto-hydrodynamic flows to analyze the turbulent flows with magnetic fields. Another research topic could be the separated-reattached turbulent flows which mostly encountered in engineering flows as a very challenging problem.

TÜRBÜLANSLI AKIŞLARIN KAPALI SONLU HACİM ALGORİTMASI İLE DOĞRUDAN SAYISAL BENZETİMİ

ÖZET

Bu doktora tezi bir Doğrudan Sayısal Benzetim (DNS) uygulama çalışmasını içermektedir. Bu kapsamda, son dönemde önerilen bir çözüm algoritması seçilmiş ve türbülansa geçiş ve hidrodinamik kararsızlık içeren akış problemlerine uygulanmıştır.

Bu doktora tezi çalışması , tüm hızlarda kullanılabilen bir Doğrudan Sayısal Benzetim (DNS) algoritmasının, türbülansa geçiş ve hidrodinamik kararsızlıklar gibi akışkanlar dinamiğinin zor problemlerine uygulanmasınıiçermektedir.

Bu, sıkıştırılabilir Navier-Stokes denklemlerini (NSE) çözen, tamamiyle kapalı bir sonlu hacim çözüm algoritmasıdır. Uzayda ve zamanda ikinci mertebe doğruluktadır. Simetrik uzay ve zaman ayrıklaştırması kullanır. Akış değişkenleri uzay örgüsü içerisinde eş-yerleşimlidir. Buna ilave olarak, uzayda basınç ayrışması probleminin önlenmesi amacıyla, hücre yüzeylerinin merkezinde bir yüzey-normal hız değişkeni tanımlanmıştır. Sıkıştırılamaz veya düşük Mach sayılı akışrejimleri içeren problemleri ele alabilmek için, ilgili denklemleri sözkonusu koşullarda uygun formlarına indirgeyen bir sıkıştırılamazlık ölçeklemesi (boyutsuzlaştırması) kullanır. Sıkıştırılamazlık limitinde, algoritmanın ayrıklaştırılmış hali kinetik enerjiyi korur. Algoritma ayrıca dissipasyon içermemekte ve yüksek Reynolds sayılı akışlarda gürbüzlüğünü korumaktadır. Bu özellikler algoritmayı türbülanslı akış rejimi içeren problemler için uygun hale getirmektedir.

Bu çalışmayı gerçekleştirmek için, yukarıda bahsedilen algoritmayı esas alan bir akış çözücüsü geliştirilmiştir. Bu DNS çözücüsü, *i*DNS, olarak adlandırlmıştır. Herhangi bir türbülans modeli kullanılmamıştır. Fortran programlama dili kullanılarak yazılmıştır. DNS metodu yüksek miktarda hesaplamalı kaynak gerektirdiğinden, paralel programlamanın gücünden yararlanmak gerekli hale gelmiştir. Bu amaçla çözücü PETSc paralel kütüphanesi kullanarak, başından itibaren ve tümüyle parallel olarak kodlanmıştır. PETSc, paralel programlar geliştirilmesi amacıyla hazırlanmış farklı alt kütüphanelerden oluşan son derece verimli bir paralel programlama kütüphanesidir. Ayrıklaştırmadan kaynaklanan geniş, seyrek katsayılar matrisleri, ya GMRES gibi yinelemeli Krylov alt-uzay metodları ile ya da cebirsel multigrid (AMG) metodu ile çözülmüştür. Bu çözümleri yapmadan önce, ilgili matrislerin özdeğerlerini iyileştirmek ve çözümlerini kolaylaştırmak amacıyla, tamamlanmamış (incomplete)-LU (ILU(0)) yöntemi kullanılarak matrisler bir ön-hazırlık (preconditioning) safhasından geçirilmiştir. Bahsedilen tüm bu yöntemler PETSc paralel programlama kütüphanesi içerisinde verimli ve iyileştirilmiş olarak, ve kullanıma hazır biçimde yeralmaktadır.

Bu çalışmanın ana amacı, sözkonusu algoritmayı aynı anda uzaysal ve zamansal olarak çok-ölçekli (multi-scale) fiziksel mekanizmalar içeren problemlere (örneğin laminer, geçişli ve türbülanslı rejimleri ve değişken Mach sayılarını bir arada içeren hidrodinamik kararsızlıklar) uygulamak yoluyla ileri testlerini yapmaktır. Bu amaçla, Taylor-Green Vortex akışındaki (TGV) türbülansa geçiş problemi, türbülanslı kayma tabakasındaki (TSL) hıza bağlı karışım problemi ve Rayleigh-Taylor kararsızlığındaki (RTI) yerçekimine bağlı karışım problemi özenle seçilmiştir.

Ayrıca, sıkıştırılamaz ve sıkıştırılabilir türbülanslı akışların DNS metodu ile yapılan modellemesinde araç olarak kullanılabilecek bir çözücü geliştirmek ve bunu araştırmacıların kullanımına sunmak suretiyle bu alana katkıda bulunmayı da amaçladık.

TGV, girdap gerilmesi ve sonucunda küçük ölçekli dönen yapıların (eddy) olustuğu ve bunların eşit dağılımlı, yönden bağımsız türbülanslı rejime evrildiği en temel akış problemi olarak kabul edilir. İki aşamalı bir evrim süreci geçirir. Erken zamanlarda, yani boyutsuz zamanın 3 ila 4 olduğu döneme kadar, başlangıçtaki yapılar sürtünme kuvvetlerinden etkilenmezler ve şekillerini korurlar. Akış esas olarak iki boyutlu devam eder ve laminer karakter gösterir. İlerleyen zamanda sürtünme kuvvetleri etkisini göstermeye başlar ve yapılar bozulmaya başlar. Eğer Reynolds sayısı yeterince büyükse, girdap gerilmesi sonucu küçük yapılar oluşur ve sonunda akış türbülans olarak yorumlanabilecek çalkantılı bir rejime doğru evrilir. Bu sepebten, TGV akışının türbülansa geçişe ilişkin temel mekanizmaları içerdiği kabul edilir. Bu geçiş mekanizmasının tamamıyle Reynolds sayısı tarafından belirlendiği açıktır.

Bizim benzetim sonuçlarımız gösteriyor ki, TGV akışının yukarıda betimlenen fiziksel mekanizması, hem algoritma hem de geliştirilen çözücü tarafından oldukça doğru bir biçimde elde edilmiştir. Sürtünmeli durum için geçiş zamanı doğru tahmin edilmiş ve sürtünmenin ihmal edildiği durum için ise kinetik enerji, uzun benzetim süresi boyunca korunmuştur.

TSL problemi pekçok fiziksel ve mühendislik probleminde karşılaşılan temel akışlardan biridir. Kuvvetli kayma gerilmesi gradyeni ile karakterize edilir. Bu çeşit akışların arkasında yatan fiziksel mekanizmaların anlaşılması önemli ve çoğu zaman gereklidir. Örnek olarak, sesüstü (supersonic) yanma reaksiyonlarının tasarlanması ve kontrolü veya kanat profilinin alt-akımındaki akışların birleştiği iz bölgesinin (wake) kontrolü verilebilir. Bu tip akışlar görece basit konfigürasyonlarına rağmen, karmaşık fiziksel olaylar içermektedir. Başlangıçtaki sanki-iki boyutlu (quasi-2D) akışkan yapıları , hız gradyenlerinin etkisialtında kolayca bozunmakta, üç boyutlu ve bir karı csım ve türbülanslı rejime doğru evrilebilmektedir.

TSL akış problemi zaman veya uzay ölçeğinde çalışılabilir. Uzayda evrilen TSL probleminde, karışan iki ana akıma ait karışım kalınlığı uzaysal koordinatlarda, akım yönünde artmakta ve gelişmektedir. Bu durumun takibi, zamanda evrilen TSL problemine göre sayısal olarak çok daha pahalıdır. Bu sebepten, karışım kalınlığının akım doğrultusu yerine zamanın bir fonksiyonu olarak geliştiği, zamanda evrilen TSL problemi tercih edilmiştir. Bu problem seçilen algoritmanın, bir doğrusal çalkantının zamanla bir hidrodinamik türbülansa evrildiği durumu betimleme ve yakalama kapasitesini göstermesi açısından önemlidir. Benzetim sonuçlarımız, daha önce ilgili kaynakçada yeralan teorik, deneysel ve sayısal sonuçlar ile başarılı bir şekilde kıyaslanmıştır. Ayrıca sıkıştırılabilirlik etkilerinin karışımın büyüme hızına etkisi de incelenmiş ve onu yavaşlattı ğıgözlemlenmiştir.

Rayleigh-Taylor kararsızlık problemi, yoğunluğu görece yüksek üstte yeralan bir akışkanın yerçekimi kuvvetlerine karşıyoğunluğu görece düşük bir başka akışkan tarafından desteklendiği durumda ortaya çıkar. Akışkan başlangıçta hidrostatik denge konumundadır. Basınç gradyeni dikey yönde etki eden yerçekimi tarafından dengelenmiş durumdadır. Basınç ve yoğunluk gradyenleri ters yönde etki etmekdedirler ve bu durumda sonsuz küçük çalkantılar iki akışkanın kesiştiği bölgede exponansiyel olarak büyür. Daha sonra vortisite (vorticity) akışkanların birbiri içerisine baloncuklar ve ince uzantılar şeklinde dahil olmasına ve sonunda karışıma yolaçar. Bu tür akışlar örneğin astrofizikteki tip 1a süpernovalarında, füzyon reaksiyonlarında, baloncuklu kavitasyon akışında ve pekçok atmosfer ve okyanus akışlarında gözlemlenebilir. Rayleigh-Taylor kararsızlığı, ilave kütle kuvvetlerinin denklemlerdeki varlığıve ele alınması , akışkan arayüzeylerinin ve akışkanların birbiri içerisine nüfuzunun doğru biçimde yakalanmasının zorunluluğu vb. fiziksel mekanizmalar açısından, hesaplamalı akışkanlar dinamiğinin (CFD) sayısal çözüm algoritmaları için oldukça zorlu bir problem anlamına gelir.

Aynı şekilde, benzetim sonuçlarımız hem bu konudaki teorik öngörüler hem de önceki çalışmalar ile uyum içerisindedir. Olayı yönlendiren fiziksel mekanizmalar doğru biçimde yakalanmıştır. Yine sıkıştırılabilirlik etkileri incelenmiş ve önceki çalışmalar ile örtüşen biçimde, kararsızlığın gelişimini hızlandırırken, karışım verimliliğini azalttığı gösterilmiştir.

Bu çalışma sırasında ve sonucunda edinilen deneyimler doğrultusunda bazı ilave araştırma konuları ortaya çıkmıştır. Örneğin algoritmanın ve tabii ki çözücünün yüksek Reynolds sayılı akışlara uygulanabilmesi için bir melez büyük-eddy benzetimi/zamana bağlı Reynolds ortalaması alma (LES/URANS) prosedürü geliştirilebilir. Benzer şekilde, algoritma ve ilgili çözücü duvar dışındaki karmaşık sınırlar içeren problemleri kapsayacak şekilde geliştirilebilir. Manyetik alanların türbülanslı akışlar üzerindeki etkilerini incelemek üzere ilgili denklemlerin çözümlerini eklemek de yine yapılabilecek çalışmalar arasındadır. Bir başka ve önemli çalışma ise genellikle mühendislik problemlerinde karşılaşılan, ayrılan-yeniden birleşen (seperated-reattached) türbülanslı akışlara ilişkin uygulamalar olabilir.

1. INTRODUCTION

1.1 Motivation and Objectives

Direct Numerical Simulation (DNS) is a numerical tool to solve the Navier-Stokes Equations (NSE) in Computational Fluid Dynamics (CFD) without using any turbulence model. DNS attempts to resolve the whole range of spatial and temporal scales of the turbulence, from the smallest dissipative scales (Kolmogorov scales), up to the scale where most of the kinetic energy is contained (integral scale), on the computational mesh. [1, 2].

Since the range of scales in turbulent flows increases rapidly with the Reynolds number, the amount of computational resources required for the simulation increase proportionally. DNS of NSE has been made possible by the developments in the computer technology in the beginning of 1970's. One of the first application was done by Orszag [3]. He solved the decaying isotropic turbulence problem on 32^3 resolution at Reynolds number (based on the Taylor micro-scale) 35 using a spectral method with triply periodic boundary conditions. Then, Rogallo [4] used 128^3 grid resolution for the same problem in early 1980's. Relatively complex domains including walls were begun to study using DNS in late 1980's by Kim et. al. [5] and Spalart [6]. Higher Reynolds number simulations with more complex domains and physics have also been started in late 1990's. Many examples can be found in the corresponding literature. Finally, DNS became a valuable and an indispensable research tool in CFD.

The numerical methods in DNS have also evolved in time, in parallel to the development of the computer technology. The early methods were mostly spectral, i.e., Fourier spectral method. Then, the explicit finite difference schemes were used to handle the problems with relatively complex flows. The main problem with these schemes are their time-step limitations and stability problems. Recently, the finite volume schemes gained an attention and importance in DNS applications, due to their more physically consistent derivation and the accuracy. Since the implicit time advanced methods require large matrix inversions, the wide range usage of implicit finite volume methods were postponed until the development of efficient serial/parallel linear solution algorithms as well as computer resources.

Today, many DNS examples which use efficient implicit finite volume algorithms can be found in the literature, for example, see Sahin and Owens [7], Martin and Candler [8], and Seidl et. al. [9] and references therein. Their usage in problems is getting increase, due to their stability properties and relatively low time-step restrictions, and of course, with the help of the advances in technology.

Recently, Hou and Mahesh [10] proposed an algorithm that has many desirable features inside. The algorithm is a pressure-correction type and second-order in both space and time. It utilizes the low-Mach number scaling to allow studying low-Mach number flows. It is called as *all-speed* solution algorithm. The algorithm uses face-normal velocity, Vn , which is treated as a separate flow variable like the velocity and is stored at cell-faces directly, to mimic the staggered grids for preventing the odd-even decoupling in pressure in space. They also showed that their algorithm is non-dissipative, robust at high Reynolds numbers and conserves kinetic energy at discrete level in low Mach number limit.

By doing this study, we aim to perform further assessments of the algorithm and make some improvements by applying it to the problems including spatial and temporal multi-scale physics simultaneously such as hydrodynamic instabilities where laminar, transitional, and turbulent regimes are found together with varying Mach number regimes.

We additionally want to make contributions into the area of modeling incompressible and compressible turbulent flows using DNS. For this purpose, an in-house, parallel, implicit DNS solver was developed and presented as an open source research tool.

1.2 Achievements

This study involves the application of an implicit, non-dissipative, kinetic energy conserving, all-speed algorithm to Direct Numerical Simulation of complex incompressible and compressible turbulent flows with instabilities.

The major achievements are:

- Analyzing the behavior of the all-speed, implicit, non-dissipative, kinetic energy conserving, DNS algorithm and testing its performance and applicability to flows including complex physical mechanisms

- Developing an in-house parallel DNS solver as an open source research tool based on the employed algorithm,

- Utilizing and incorporating an efficient parallel library, PETSc, into the solver to obtain an advanced research tool,

- Improving stability and convergence characteristics of an existing algorithm by adding a time-correction procedure,

- Testing various iterative linear solution algorithms such as GMRES and Algebraic Multigrid (AMG) to solve resulting discretized form of equations,

- Analyzing the compressibility effects on the development of hydrodynamic instabilities,

- Developing parallel post-processing routines for real-time analyzing,

- Contributing to the DNS databases for hydrodynamic instabilities.

1.3 Outline of Thesis

Thesis starts with an introduction that places the study in the big picture of CFD. Chapter 2 mentions the governing equations of fluid flow and their non-dimensionalization. Chapter 3 details the algorithm employed and the iterative solution procedure. An additional correction procedure to increase the stability and convergence characteristics of the algorithm in time is given. The physical boundary conditions and their numerical treatments are also explained in the same chapter. The development of an in-house, parallel, implicit solver is presented in Chapter 4. The incorporating PETSc into the solver is detailed. The properties of resulting parallel solver such as handling boundaries in parallel, domain decomposition and the parallel performance of the solver are also included in the chapter. Transition to turbulence in Taylor-Green flow using DNS is performed in Chapter 5. This chapter also serves as a verification of the developed solver. Chapter 6 includes DNS of initially turbulent mixing layer which is velocity-induced instability problem. It includes a very wide range of scales and is a challenging test case for many algorithms in order to check the ability of algorithms. Similarly, DNS of Rayleigh-Taylor instability which is a gravity-induced instability problem is included in Chapter 7. The further difficulties are presented by this instability such as density ratios which may force the algorithms to fail. The last chapter makes conclusions of this thesis study and recommends some possible future research topics.

2. GOVERNING EQUATIONS

In this chapter, the equations that govern fluid flow are briefly presented. Their derivation is based on the fundamental physical concepts of the conservation of mass, momentum and energy. Details of the derivation can be found in any elementary textbook on fluid mechanics or CFD [1, 2, 11] and will not be repeated here.

2.1 The Compressible Navier-Stokes Equations

The governing equations of the three-dimensional, time-dependent motion of compressible, viscous fluid are known as the Navier-Stokes equations. Denoting the dimensional values by asterisk, $(*)$, these equations can be expressed in conservative form using Cartesian tensor notation as,

$$\frac{\partial \rho^*}{\partial t^*} + \frac{\partial \left(\rho^* u_j^*\right)}{\partial x_j^*} = 0 \quad \textbf{(2.1}\text{a)}$$

$$\frac{\partial \left(\rho^* u_i^*\right)}{\partial t^*} + \frac{\partial \left(\rho^* u_i^* u_j^*\right)}{\partial x_j^*} = \frac{\partial \sigma_{ij}^*}{\partial x_j^*} + \rho^* g_i^* \quad \textbf{(2.1}\text{b)}$$

$$\frac{\partial \left(\rho^* e_T^*\right)}{\partial t^*} + \frac{\partial \left(\rho^* e_T^* u_j^*\right)}{\partial x_j^*} = -\frac{\partial q_j^*}{\partial x_j^*} + \frac{\partial \sigma_{ij}^* u_i^*}{\partial x_j^*} + \rho^* g_i^* u_i^* \quad \textbf{(2.1}\text{c)}$$

where x_j^* denotes the j^{th} co-ordinate direction, t^* denotes time, ρ^* denotes the mass density, u_i^* denotes the fluid velocity in the x_i^* direction.

The total energy per unit mass, e_T^* is the sum of the internal energy per unit mass, e^*, and the kinetic energy per unit mass,

$$e_T^* = e^* + \frac{1}{2} u_j^* u_j^* \quad \textbf{(2.2)}$$

σ^*_{ij} is the stress tensor consisting of two components,

$$\sigma^*_{ij} = -p^*\delta_{ij} + \tau^*_{ij} \tag{2.3}$$

where δ is the Kronecker delta, p^* is the pressure and τ^*_{ij} is the viscous stress tensor. Assuming the fluid is Newtonian, τ^*_{ij} is given as,

$$\tau^*_{ij} = \mu^*\left(\frac{\partial u^*_i}{\partial x^*_j} + \frac{\partial u^*_j}{\partial x^*_i}\right) + \lambda^*\delta_{ij}\left(\frac{\partial u^*_k}{\partial x^*_k}\right) \tag{2.4}$$

where μ^* is the dynamic viscosity coefficient, λ^* is the second viscosity coefficient. Using the hypothesis introduced by Stokes [12], $\lambda^* = -\frac{2}{3}\mu^*$, viscous stress tensor can be given by,

$$\tau^*_{ij} = \mu^*\left(\frac{\partial u^*_i}{\partial x^*_j} + \frac{\partial u^*_j}{\partial x^*_i} - \frac{2}{3}\delta_{ij}\frac{\partial u^*_k}{\partial x^*_k}\right) \tag{2.5}$$

The dependence of the dynamic viscosity to temperature can be expressed with Sutherland's power law [13],

$$\frac{\mu^*}{\mu^*_0} = \left(\frac{T^*}{T^*_0}\right)^n \tag{2.6}$$

where μ^*_0 and T^*_0 are the reference dynamic viscosity and the reference temperature, respectively. The value of n is set to 0.7 in the simulations.

Using the Fourier's law of heat conduction, the heat flux, q^*_j, in the energy equation is written as,

$$q^*_j = -k^*\frac{\partial T^*}{\partial x^*_j} \tag{2.7}$$

k^* is the thermal conductivity coefficient given as,

$$k^* = c^*_p\frac{\mu^*}{Pr} \tag{2.8}$$

where Pr is the Prandtl number.

In order to close the system of equations, the equation of state for the ideal gas is introduced,

$$p^* = \rho^* R^* T^* \tag{2.9}$$

where R^* denotes the universal gas constant. For an ideal gas, internal energy and enthalpy, h^*, can be related to temperature as,

$$e^* = c_v^* T^* \tag{2.10a}$$

$$h^* = c_p^* T^* \tag{2.10b}$$

respectively. Here c_v^* and c_p^* are the constant specific heats. Their ratio,

$$\gamma = \frac{c_p^*}{c_v^*} \tag{2.11}$$

is also constant and set to 1.4 in the simulations.

Finally, g_i^* is the source term contribution to the equations due to the gravity.

2.2 Non-dimensional Form of the Equations

Non-dimensionalization or normalization improves the performance of numerical algorithms and the accuracy of solutions as well. It is also a necessary step to normalize the pressure in an appropriate way as shown below, so-called *incompressible* or *low-Mach number scaling*, to yield the incompressible equations from the compressible ones, when the Mach number approaches to zero [13, 14].

The non-dimensional quantities can be defined using the appropriate reference values as following,

$$x_i = x_i^*/l_0^* \tag{2.12a}$$

$$t = t^*/\left(l_0^*/u_0^*\right) \tag{2.12b}$$

$$\rho = \rho^*/\rho_0^* \tag{2.12c}$$

$$u_i = u_i^*/u_0^* \tag{2.12d}$$

$$p = \left(p^* - p_0^*\right)/\left(\rho_0^* u_0^{*2}\right) \tag{2.12e}$$

$$g_i = g_i^*/g_0^* \tag{2.12f}$$

$$\mu = \mu^*/\mu_0^* \tag{2.12g}$$

where $(\cdot)_0$ denotes the reference quantities. l^* is the reference length-scale. p_0^* is also written as,

$$p_0^* = \rho_0^* R^* T_0^* \tag{2.13}$$

Using the definitions above, the non-dimensional continuity and momentum equations are

$$\frac{\partial \rho}{\partial t} + \frac{\partial\left(\rho u_j\right)}{\partial x_j} = 0 \tag{2.14a}$$

$$\frac{\partial\left(\rho u_i\right)}{\partial t} + \frac{\partial\left(\rho u_i u_j\right)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re}\frac{\partial \tau_{ij}}{\partial x_j} + \frac{1}{Fr^2}\rho g_i \tag{2.14b}$$

The energy equation is the most influenced one by the low-Mach number scaling. After rearrangement, it takes the following form,

$$\begin{split} M^2\left[\frac{\partial}{\partial t}\left(p + \frac{\gamma-1}{2}\rho u_i u_i\right) + \frac{\partial}{\partial x_j}\left(\gamma p + \frac{\gamma-1}{2}\rho u_i u_i\right)u_j\right] + \frac{\partial u_j}{\partial x_j} \\ = \frac{(\gamma-1)M^2}{Re}\frac{\partial\left(\tau_{ij}u_i\right)}{\partial x_j} + \frac{1}{RePr}\frac{\partial}{\partial x_j}\left(\mu\frac{\partial T}{\partial x_j}\right) + Ec(\rho g_i u_i) \end{split} \tag{2.15}$$

The non-dimensional form of the equation of state is also obtained as,

$$\rho T = \gamma M^2 p + 1 \tag{2.16}$$

In the above set of non-dimensional, time-dependent, compressible Navier-Stokes equations;
the Reynolds number, Re, is given by,

$$Re = \frac{\rho_0^* u_0^* l_0^*}{\mu_0^*} \tag{2.17}$$

the *Mach* number, M , is given by,

$$M = \frac{u_0^*}{\sqrt{\gamma \frac{P_0^*}{\rho_0^*}}} \tag{2.18}$$

the Froude number, Fr, is given by,

$$Fr = \frac{u_0^*}{\sqrt{g_0^* l_0^*}} \tag{2.19}$$

and the Eckert number, Ec, is given by,

$$Ec = \frac{l_0^* g_0^*}{c_p^* T_0^*} \tag{2.20}$$

which is the ratio of kinetic energy to enthalpy, as it will be seen in Chapter 7. For fluids with constant specific heats, it can also be approximated as [15],

$$Ec \approx (\gamma - 1) M^2 \tag{2.21}$$

Further details of the normalization procedure can be found in [16].

3. NUMERICAL METHOD

In this chapter, the numerical method used in this thesis study is briefly discussed. First, the features of the method are summarized. Then, derivation of the discrete set of equations and the procedure to solve them are described.

3.1 Introduction

A fully implicit, non-dissipative, discrete energy conserving algorithm proposed by Hou and Mahesh [10] is used in this thesis study to develope a DNS solver, and to solve the discrete set of time-dependent, compressible Navier-Stokes equations. It combines some desirable and unique features of the previously developed algorithms to solve the turbulent flow problems.

The algorithm is second-order in both space and time on orthogonal grids. It uses *predictor-corrector* approach to update the values of flow variables in time. The pressure difference is used to perform the correction step, so-called *pressure-correction*. It utilizes the incompressible or low-Mach number scaling proposed by White [13], and by Wesseling [14] for the pressure, as described in the previous chapter. The resulting non-dimensional form of the equations satisfy the incompressibility condition, and allow to study low-Mach number flows with the same algorithm without encountering the stifness problem, when the Mach number goes to zero. It can be called as *all-speed* or *all-Mach* solution algorithm.

The algorithm was designed for co-located grids. The variables are stored at cell-centers, as shown in Fig.3.1. The discretizations are centered in both space and time, as it was previously suggested by Wall et. al. [17] to ensure the non-dissipativeness. Thermodynamic quantities such as density, pressure, and temperature are also half a time step staggered in time from velocities.

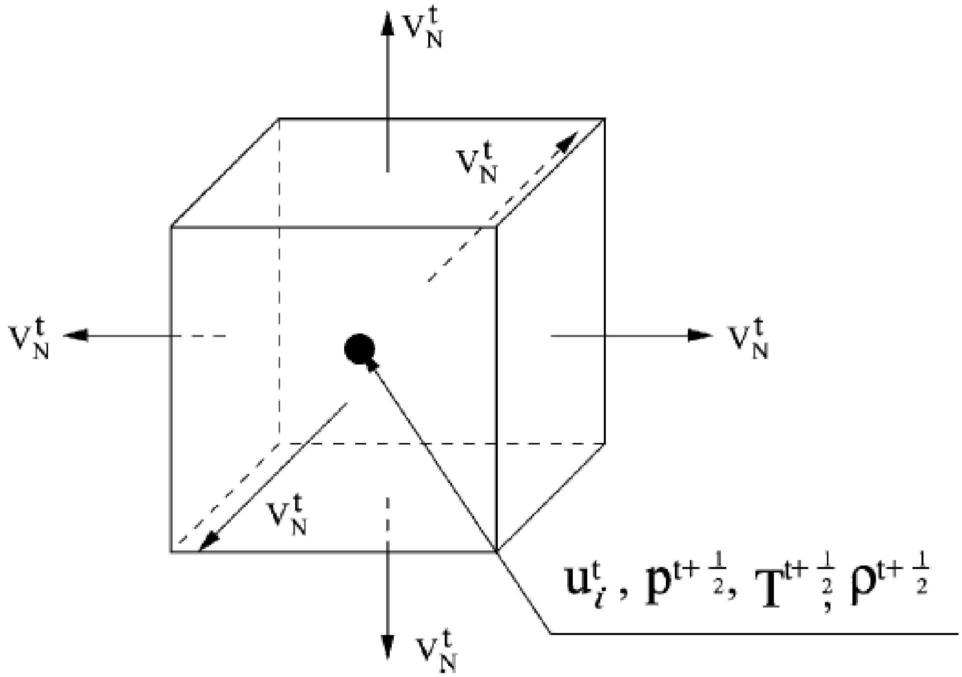


Figure 3.1: Storage of flow variables, taken from [10].

Unlike the many previous co-located methods that interpolate the velocities from the adjacent cell-centers to obtain the velocities at cell faces, the algorithm uses *face-normal velocity*, V_n, as it was previously used by Kim and Choi [18], and Zang et. al. [19]. In this way, V_n is treated as a seperate flow variable like velocity and is stored at cell-faces directly, instead of cell-centers. In fact, this strategy is utilized to mimic the staggered grids to prevent the odd-even decoupling in pressure in space. However, an additional correction to pressure in time is needed and adopted into the algorithm to enhance the convergence properties by preventing the pressure oscillations in time. The details of this correction approach are given in Sec. 3.4.

In order to fully implicit treatment of pressure and to remove the acoustic stiffness at low-Mach numbers, the algorithm uses a new pressure variable, $\bar{p}$. $\bar{p}$, which is obtained by applying the trapezoidal rule, is a weighted sum of previous, current, and predicted values of the pressure. This approach was also first introduced by Wall et. al. [17]

The values of the variables at cell-faces are obtained by *symmetric interpolation*, which is the simple averaging of two adjacent cell values. It was shown by Benhamadouche et. al. [20] that this is necessary to conserve kinetic energy discretely at low-Mach limit. The only disadvantage is that second-order accuracy is hardly achieved, when the grid non-orthogonality is severe. However, more serious problems may occur, if the kinetic energy is not conserved discretely for certain type of problems, due to the usage of the non-symmetric interpolations or non-orthogonal correction terms to obtain cell-face values. In addition, symmetric interpolation keeps the computational molecule compact. It saves the memory and also reduces the computational time. In case of fully implicit algorithm, it is a great advantage. The choice seems problem-dependent.

3.2 Discretization

Separate discretization in space and in time, so-called *method of lines*, is employed in this thesis together with the Finite Volume Method (FVM) to discretize the equations in space. An implicit discretization is used in time.

The FVM satisfies the global conservation, through the direct discretization of the integral form of the conservation laws. So, it is the natural choice for the governing equations in conservation form introduced in Chapter 2.

The FVM also performs the spatial discretization directly in the physical space. Thus, there is no need to transform the physical space to the computational space or vice versa.

As a first step, physical space or solution domain is divided into number of control volumes (cells). No free spaces among the control volumes are allowed. The control volumes do not overlap on each other. Then, differential equations are integrated over each of the control volume and *Gauss-divergence* theorem is applied which states that [21] if V is the volume bounded by a closed surface S and F is vector function of position with continuous derivatives, then,

$$\int_V (\nabla \cdot F)\, dV = \int_S (F \cdot N)\, dS \qquad \textbf{(3.1)}$$

where N is the face-normal vector. The resulting surface integral is approximated by a sum of the fluxes crossing the faces of the cells. Flux is assumed constant and calculated at the midpoint of the cell-face. The cell-face values are also approximated using the adjacent cell-center values. In the present method, this is done by arithmetic averaging.

Similarly, the required values for the cell gradients are obtained by *Green-Gauss* theorem which states that [21] the surface integral of a scalar function, ϕ, is equal to the volume integral (over the volume bound by the surface) of the gradient of the scalar function,

$$\int_V \nabla\phi \, dV = \int_S \phi \, N \, dS \tag{3.2}$$

Assuming that $\nabla\phi$ is constant over the control volume, the Green-Gauss equation can be rewritten as ,

$$\int_V \nabla\phi \, dV = \nabla\phi \, V = \int_S \phi \, N \, dS \tag{3.3}$$

The integral over the surface is further approximated as a summation of the average scalar value in each face times the face's surface vector *S*,

$$\nabla\phi = \frac{1}{V} \sum_{faces} \phi_{face} \vec{S}_{face} \tag{3.4}$$

Following the notation in Hou and Mahesh [10], the resulting discrete continuity equation is,

$$\frac{\rho_{cv}^{t+3/2} - \rho_{cv}^{t+1/2}}{\Delta t} + \frac{1}{V} \sum_{faces} \rho_{face}^{t+1} V_n^{t+1} A_{face} = 0 \tag{3.5}$$

$(\cdot)_{cv}$ and $(\cdot)_{face}$ represent the value of the variable at the cell-center and at the cell-face, respectively. $\sum_{faces}$ denotes summation over all the faces of the cell. V_n is the corresponding face–normal velocity at the each cell-face. V and A are the volume of the cell and the area of the each cell-face, respectively. ρ_{face}^{t+1} is obtained by symmetric interpolation in time.

The resulting discrete momentum equations have the following form,

$$\frac{(\rho u_i)^{t+1}_{cv} - (\rho u_i)^{t}_{cv}}{\Delta t} + \frac{1}{V}\sum_{faces}(\rho u_i)^{t+\frac{1}{2}}_{face} V_n^{t+\frac{1}{2}} A_{face} =$$
$$-\frac{\partial}{\partial x_i}(\bar{p}^t) + \frac{1}{Re}\frac{1}{V}\sum_{faces}\left(\tau_{ij}^{t+\frac{1}{2}}\right)_{face} N_j A_{face} + \frac{1}{Fr^2}\rho^{t+1}_{cv} g_{i,cv} \quad \textbf{(3.6)}$$

where ρu_i is momentum in the i^{th} direction. $\tau_{ij}^{t+\frac{1}{2}}$ is the stress tensor at face. g_i is the gravitational acceleration in the i^{th} direction. N_j is the corresponding face-normal vector pointing outside of the face. $\bar{p}^t$ is the modified pressure written as,

$$\bar{p}^t = \left(\frac{1}{4}\right) p^{t-\frac{1}{2}} + \left(\frac{1}{2}\right) p^{t+\frac{1}{2}} + \left(\frac{1}{4}\right) p^{t+\frac{3}{2}} \quad \textbf{(3.7)}$$

The resulting discrete energy equation reads,

$$M^2\left[\frac{\partial}{\partial t}\left(p_{cv} + \frac{\gamma-1}{2}\rho u_i u_i\right)^{t+\frac{1}{2}} + \frac{1}{V}\sum_{faces}\left(\gamma p_{cv} + \frac{\gamma-1}{2}\rho u_i u_i\right)^{t+\frac{1}{2}}_{face}\cdot V_N^{t+\frac{1}{2}} A_{face}\right]$$
$$+\frac{1}{V}\sum_{faces} V_n^{t+\frac{1}{2}} A_{face} = \frac{(\gamma-1)M^2}{Re}\frac{1}{V}\sum_{faces}\left(\tau_{ij}u_i\right)^{t+\frac{1}{2}}_{face} N_j A_{face}$$
$$+\frac{1}{RePr}\frac{1}{V}\sum_{faces}\left(\mu\frac{\partial T^{t+\frac{1}{2}}}{\partial N}\right) A_{face} + Ec\left(\rho^{t+1}_{cv} g_{i,cv} V_n^{t+\frac{1}{2}}\right) \quad \textbf{(3.8)}$$

In order to obtain the set of discrete equations to be used for implementing into solver, further derivations are needed. The discrete momentum equation is rewritten in predictor form,

$$\frac{(\rho u_i)^{\star}_{cv} - (\rho u_i)^{t}_{cv}}{\Delta t} + \frac{1}{V}\sum_{faces}(\rho u_i)^{t+\frac{1}{2},\star}_{face} V_n^{t+\frac{1}{2},k} A_{face} =$$
$$-\frac{\partial}{\partial x_i}(\bar{p}^{t,k}) + \frac{1}{Re}\frac{1}{V}\sum_{faces}\left(\tau_{ij}^{t+\frac{1}{2},k}\right) N_j A_{face} + \frac{1}{Fr^2}\rho^{t+1,k+1}_{cv} g_{i,cv} \quad \textbf{(3.9)}$$

The predicted momentum values at the k^{th} outer loop are denoted by star, $(\star)$.

The pressure at the end of the k^{th} outer loop is also decomposed as,

$$p^{t+\frac{3}{2},k+1} = p^{t+\frac{3}{2},k} + \delta p \quad \textbf{(3.10)}$$

Putting it into exact momentum equation and taking the difference between exact and predictor momentum equations give the necessary expressions to be used in the corrector step,

$$(\rho u_i)_{cv}^{t+1,k+1} = (\rho u_i)_{cv}^{\star} - \frac{\Delta t}{4}\left(\frac{\partial \delta p}{\partial x_i}\right)_{cv} \tag{3.11}$$

$$u_{i,cv}^{t+1,k+1} = u_{i,cv}^{\star} - \frac{\Delta t}{4\rho_{cv}^{t+1}}\left(\frac{\partial \delta p}{\partial x_i}\right)_{cv} \tag{3.12}$$

$$(V_n)^{t+1,k+1} = (V_n)^{\star} - \frac{\Delta t}{4\rho_{face}^{t+1}}\frac{\partial \delta p}{\partial N} \tag{3.13}$$

where

$$V_n^{*} = (u_i^{\star})_{face} \cdot N_i \tag{3.14}$$

The specific kinetic energy is also defined by,

$$(u_i)^{t+1,k+1}(u_i)^{t+1,k+1} = u_i^{\star}u_i^{\star} - \frac{\Delta t}{2\rho_{cv}^{t+1}}u_i^{\star}\frac{\partial \delta p}{\partial x_i} + \textit{higher order terms} \tag{3.15}$$

After substituting the Eq.**3.10** to Eq.**3.14** into the discrete energy equation and linearizing i.e., keeping δp linear and omitting its higher-order terms, *the pressure-correction* form of the energy equation is finally obtained,

$$
\begin{aligned}
&\left\{\frac{M^2}{2\Delta t}+\frac{M^2}{V}\sum_{faces}\left[-\frac{\gamma-1}{8}V_n^{t+\frac{1}{2},\star}+\frac{\gamma}{8}V_n^{t+\frac{1}{2},\star}-\frac{\gamma-1}{8}V_n^{t+\frac{1}{2},\star}\frac{\rho_{face}^{t+\frac{1}{2}}}{\rho_{face}^{t+1}}\frac{u_{j,face}^{t+\frac{1}{2},\star}\Delta t}{\Delta_{nb}}N_j\right.\right.\\
&\qquad\left.\left.-\frac{\Delta t\,p_{face}^0 N_j}{8\rho_{face}^{t+1}\Delta_{nb}}\right]A_{face}-\frac{1}{V}\sum_{faces}\frac{\Delta t N_j}{8\rho_{face}^{t+1}\Delta_{nb}}A_{face}\right\}\delta p+\\
&\left\{\qquad\frac{M^2}{V}\sum_{faces}\left[-\frac{\gamma-1}{8}V_n^{t+\frac{1}{2},\star}+\frac{\gamma}{8}V_n^{t+\frac{1}{2},\star}+\frac{\gamma-1}{8}V_n^{t+\frac{1}{2},\star}\frac{\rho_{face}^{t+\frac{1}{2}}}{\rho_{face}^{t+1}}\frac{u_{j,face}^{t+\frac{1}{2},\star}\Delta t}{\Delta_{nb}}N_j\right.\right.\\
&\qquad\left.\left.+\frac{\Delta t\,p_{face}^0 N_j}{8\rho_{face}^{t+1}\Delta_{nb}}\right]A_{face}+\frac{1}{V}\sum_{faces}\frac{\Delta t N_j}{8\rho_{face}^{t+1}\Delta_{nb}}A_{face}\right\}\delta p_{nb}\\
&=\frac{(\gamma-1)M^2}{Re}\frac{1}{V}\sum_{faces}(\tau_{ij}u_i)_{faces}^{t+\frac{1}{2},k}N_jA_{face}+\frac{1}{Re}\frac{1}{Pr}\frac{1}{V}\sum_{faces}\mu_{face}\frac{\partial T^{t+\frac{1}{2},k}}{\partial N}A_{face}\\
&-\left\{\frac{M^2}{\Delta t}\left[\frac{p^{t+\frac{3}{2},k}-p^{t-\frac{1}{2}}}{2}+\frac{\gamma-1}{2}(\rho^{t+1}u_i^\star u_i^\star-\rho^t u_i^t u_i^t)+\frac{M^2}{V}\sum_{faces}p_{face}^0 V_n^{t+\frac{1}{2},\star}A_{face}\right.\right.\\
&\qquad\left.\left.+\frac{1}{V}\sum_{faces}V_n^{t+\frac{1}{2},\star}A_{face}\right]\right\}+Ec\rho_{cv}^{t+1,k+1}u_{i,cv}^{t+\frac{1}{2},\star}g_{i,cv}
\end{aligned}
\tag{3.16}
$$

where $(\cdot)_{nb}$ denotes the neighboring cells. $\Delta_{nb}=(\vec{x_{nb}}-\vec{x_{cv}})\cdot\vec{N}$ is the normal distance between neighboring cell centers. $u_i^{t+\frac{1}{2},\star}=\frac{1}{2}(u_i^\star+u_i^t)$ and $V_n^{t+\frac{1}{2},\star}=\frac{1}{2}(V_n^\star+V_n^t)$. p_{face}^0 is also given by,

$$
p_{face}^0=\gamma\overline{p}_{face}^t+\frac{\gamma-1}{2}\rho_{face}^{t+\frac{1}{2}}(u_i^{t+\frac{1}{2},\star}u_i^{t+\frac{1}{2},\star})_{face}
\tag{3.17}
$$

This discretization corresponds to a $7-point\,stencil$ and yields a non-symmetric, hepta-diagonal structured matrix in three-dimensional space.

For further details of the derivation and discretization, please refer to Hou and Mahesh [10].

3.3 The Iterative Solution Procedure

Here, a segregated iterative pressure-correction solution procedure given in Hou and Mahesh [10] is summarized. Further details can be found in the corresponding reference.

* The outer loop, that is the loop taken within the time-step until convergence is reached, is initialized by setting the time values to loop values at initial.

* The discrete continuity equation given in Eq.**3.5** is solved for new density value.

* The predictor form of momentum equation given in Eq.**3.9** is solved for $(\rho u_i)^\star$.

* The intermediate velocities, $u_i^\star$'s, are obtained via dividing $(\rho u_i)^\star$ by the average of current and new densities and the intermediate face-normal velocities, $V_n^\star$'s, are calculated.

* The pressure-correction equation given in Eq.**3.16** is solved for δp.

* Pressure, momentum, velocity components and face-normal velocity are updated using Eq.**3.10** to Eq.**3.13**, respectively. The new temperature and the dynamic viscosity are also found using Eq.**2.16** and Eq.**2.6**, respectively.

* Outer loop convergence is checked for density, momentum, and pressure-correction.

The outer loop convergence test is simply based on the $l2-norm$ of the residual for density, each momentum, and pressure-correction,

$$\|\rho\| = \sqrt{\frac{1}{ncells}\sum_{ncells}(\rho^{t+\frac{3}{2},k+1} - \rho^{t+\frac{3}{2},k})^2} < \varepsilon_{outer} \quad \textbf{(3.18)}$$

$$\|(\rho u_i)\| = \sqrt{\frac{1}{ncells}\sum_{ncells}((\rho u_i)^{t+1,k+1} - (\rho u_i)^{t+1,k})^2} < \varepsilon_{outer} \quad \textbf{(3.19)}$$

$$\|\delta p\| = \sqrt{\frac{1}{ncells}\sum_{ncells}\delta p^2} < \varepsilon_{outer} \quad \textbf{(3.20)}$$

where *ncells* denotes number of cells in domain. The differences are obtained from two consecutive outer loop iterations. Convergence is decided by the decrease of all the residual norms below to a given tolerance, ε_{outer}, which is set to 10^{-3}. For pressure-correction, since we want δp to converge to zero when the outer loop has converged, error calculation is slightly different from others [22].

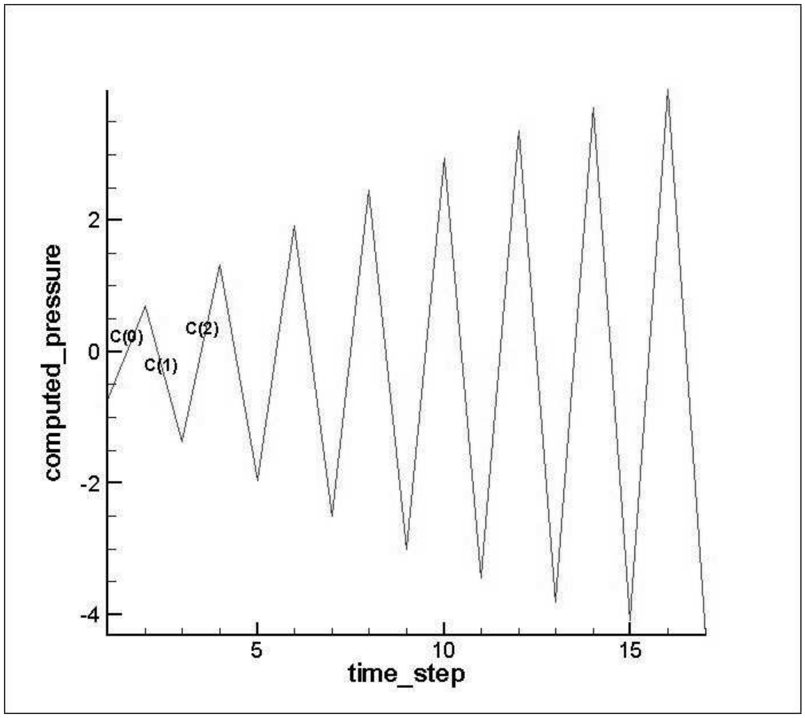


Figure 3.2: The illustration of the oscillatory behavior in pressure in time. $C(i)$ denotes the pressure difference between current and old time steps.

3.4 Additional Correction to Pressure in Time

During the simulations, especially performed at low-Mach numbers, a very slow convergence behavior is encountered, due to the oscillating successive corrections for the pressure in time. This situation is illustrated in Fig.3.2 at a single cell. As it is seen, pressure converges to two different values. The each successive pressure correction is a variable ratio of the previous correction.

In order to avoid this issue, a time-relaxation to pressure, similar to the procedure suggested by Walton [23], is applied at the end of the each time-step in the following form,

$$p_{new,relax} = p_{new} - \frac{(p_{new} - p)}{1 - \frac{(p_{new} - p)}{(p - p_{old})}} \tag{3.21}$$

where p_{old}, p and p_{new} are the *old, current* and *new* time-step values of the pressure, respectively. It performs like the Steffensen iteration [24]. This procedure provides a time- and space-dependent relaxation factor to pressure. The time-relaxed pressure, $p_{new,relax}$, is then used in the next time step.

Since the values of p_{old}, p and p_{new} are strongly correlated to each other in compressible flow, the relaxation term (i.e., the 2^{nd} term at the right side of Eq. **3.21**) and its overall effect are expected to be small and negligible at high Mach numbers.

3.5 Boundary Conditions

Boundary conditions are implemented using ghost cell approach. A layer of ghost cells surrounding the physical domain are defined and are used to store the necessary values.

For explicit solvers, it is enough to represent boundary conditions within the solver in a global manner. However, a numerical difficulty arises with the implicit solvers that is the boundary conditions must also be represented within the linear system arising from the discretization of each equation. This requires modification of the corresponding entries in the linear systems before sending them to the linear solver. Since the application of the boundary conditions for each variable is different, modifications of the corresponding linear system must also be done separately.

In case of periodicity, the ghost cell layer corresponds to the boundary cells at the opposite periodic boundary for all variables (density, velocity, pressure, momentum, temperature, pressure-correction and dynamic viscosity).

For the inviscid slip-wall, ghost cell values are simply determined based on the discrete version of homogeneous Neumann boundary condition for all variables, $\frac{\partial \phi}{\partial N} = 0$, where ϕ is the corresponding flow variable.

High-order numerical treatment of boundary conditions are, of course, possible. However, this comes with an extra storage in matrix, as well as with a complexity in programming PETSc. Results show that the above treatment works well in our cases.

There is another point that must be taken into account for the current algorithm. As it was mentioned in the previous chapters, it uses face-normal velocities. They are stored at the center of the cell-faces and updated separately like the other conservative variables. If the cell-face corresponds to an inviscid wall boundary (slip-wall), they must be further modified and set to zero in the corrector step (see Eq. **3.13**). The only contribution must come from the pressure terms.

4. AN IN-HOUSE PARALLEL DNS SOLVER: *i*DNS

During this thesis study, an *in-house* fully parallel solver, *i*DNS (fully **i**mplicit **DNS**), was developed based on the chosen numerical algorithm. *i*DNS is used to simulate the turbulent flows introduced in the following chapters. The details of the solver and the solution of the linear systems arising from the discretization are given in this chapter. The parallelization of the solver and its parallel performance are also presented.

4.1 Code Structure and Parallelization

*i*DNS is a single-block structured finite-volume solver working on hexahedral cells. It consists of three parts; *pre-processing*, *solver* and *post-processing*. In pre-processing, PETSC framework is initialized. Grid quantities and initial conditions for the simulation are generated. Solver part preforms all the steps defined in Sec.3.3. Post-processing step calculates all the necessary quantities for analyzing and comparison, and also stores them into files. Since *i*DNS was written in a modular fashion, it is a collection of subroutines. Each part has its own subroutines. Flowchart of *i*DNS is illustrated in Fig.4.1

*i*DNS is written in completely parallel using PETSc [25, 26] with Fortran syntaxing. PETSc is an object-oriented library developed by Argonne National Laboratory. As it is described in the user manual [27], *"The Portable, Extensible Toolkit for Scientific Computation (PETSc) is a suite of data structures and routines that provide the building blocks for the implementation of large-scale application codes on parallel (and serial) computers. PETSc uses the Message Passing Interface (MPI) [28] standard for all message-passing communication. PETSc includes an expanding suite of parallel linear, nonlinear equation solvers and time integrators that may be used in application codes written in Fortran, C, and C++. PETSc provides many of the mechanisms needed within parallel application codes, such as parallel matrix and vector assembly routines".* PETSc has interfaces to many external

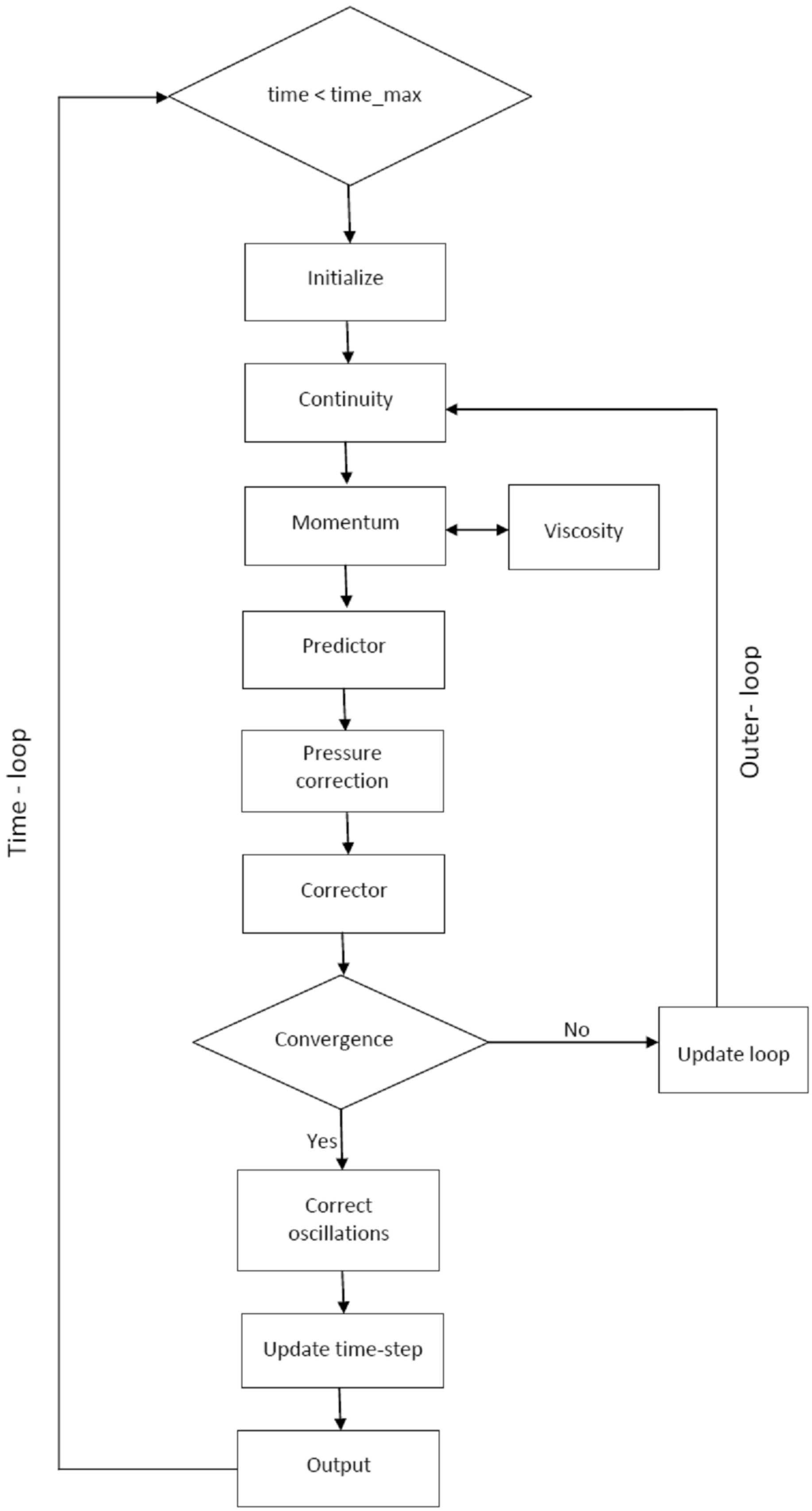


Figure 4.1: Flowchart of *i*DNS.

solvers such as Hypre [29] and ParMETIS [30]. As shown in 4.2, PETSc consists of modules or objects including Index Sets (IS), Vectors (Vec), Matrices(Mat), Distributed Arrays(DA), Krylov Subspace (KSP) methods, Preconditioners (PC), multi-grid and sparse direct solvers, nonlinear solvers, and time-steppers (TS) for solving time-dependent (non-linear) Partial Differential Equations (PDE).

*i*DNS is based on the Single Program Multiple Data (SPMD) approach. All processes use the same program, but each has its own data. This approach is suitable for distributed memory architectures.

Many of the details of message passing such as vector scatters/gathers and distributed arrays to manage the data in parallel are provided within PETSc. These are embedded into parallel objects, such as vectors, matrices, and solvers. However, it should be mentioned that PETSc has a steep learning curve.

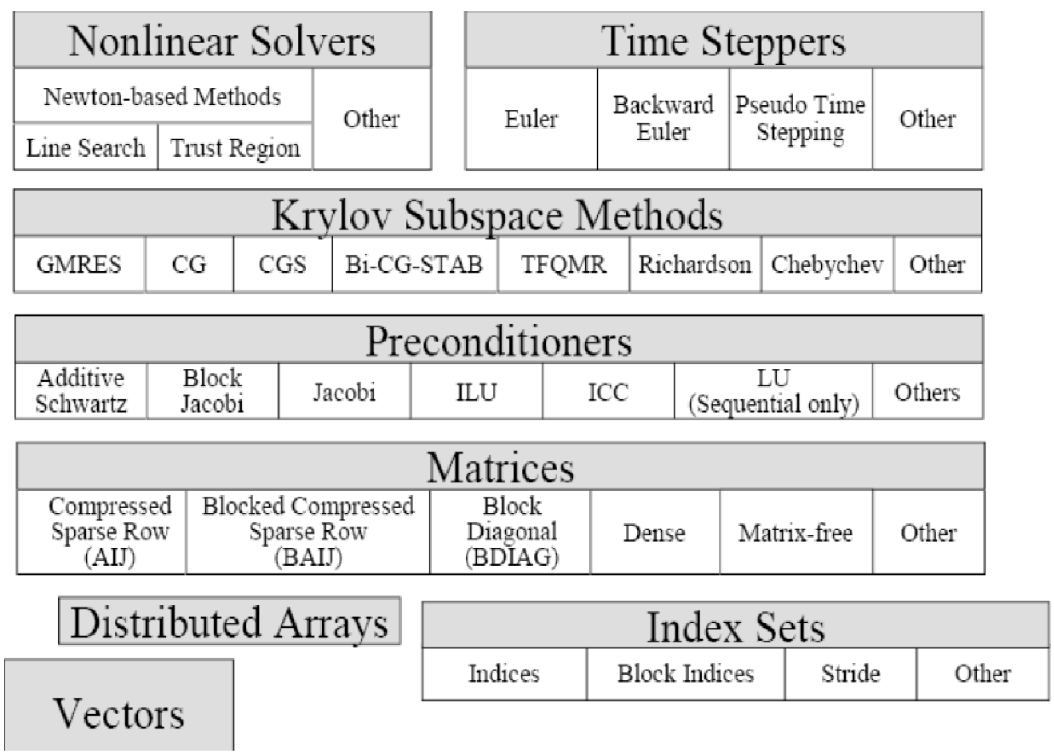


Figure 4.2: Hierarchical structure of numerical libraries in PETSc (petsc-3.1-p8) [27].

The Vector (Vec) is one of the simplest and basic PETSc objects. Vectors are used to store variables, solutions, right-hand sides for linear systems, etc. Many parallel vector operations are already defined in PETSc.

Various dense and sparse matrix formats are supported. Matrix operations are also already available in parallel.

Distributed Arrays (DA) are designed for handling the communication of non-local data on logically regular rectangular grids. Evaluation of a local function, $f(x)$, at the interfaces of the sub-domains owned by each process requires inter-process communication. This is automatically managed by DA object.

DA, more specifically, is a topology interface that handles parallel data layout on structured grids. It handles local and global indices via routines *DAGetGlobalIndices()* and *DAGetAO()*. It provides local and global vectors via routines *DAGetGlobalVector()* and *DAGetLocalVector()*. It also performs updating ghost values via routines *DAGetGlobalToLocal()* and *DAGetLocalToGlobal()*.

DA object contains only topology information. All field data are stored in PETSc Vecs. Global vectors are parallel. Each process stores a unique local portion of global vector. Local vectors are sequential and include ghost values based on a given stencil information in creating DA.

KSP is the another fundamental object which is designed for solving nonsingular linear systems of the form, $Ax = b$, where A denotes the matrix, b is the right-hand-side vector, and x is the solution vector. KSP is the combination of Krylov subspace methods and variety of matrix preconditioners (PC).

Further details about PETSc such as downloading, installation, usage, other objects and profiling can be found in references [25–27].

4.2 Domain Decomposition

4.2.1 Definition

Domain decomposition (DD) is a general name of collection of techniques to solve PDE by dividing the physical solution domain into sub-domains. It is based on an old idea of *divide and conquer*. It was first introduced by Schwarz [31] in 1890. DD is a natural method for effective parallel algorithms for distributed memory computers.

Domain decomposition can mathematically be expressed as,

$$\Omega = \bigcup_{1}^{s} \Omega_i \tag{4.1}$$

where s denotes number of sub-domains, Ω_i.

Domain decomposition may have different meanings depending on the technique that you apply in solving PDE [32, 33]. Two of them will be mentioned in the following subsections.

4.2.2 Domain decomposition as grid partitioning technique

In pre-processing step in parallel CFD, it may refer to process of dividing domain into sub-domains and assigning each of them to separate processors. Data is also decomposed among the processors, associated with sub-domain. This can be done either in a simple way of generating equally-sized sub-blocks or using an advanced graph representations for partitioning [30].

4.2.3 Domain decomposition as preconditioning technique

When solving linear systems, domain decomposition also serves as preconditioner. In this case, large linear systems is divided into smaller ones and their individual solutions can be used to produce a preconditioner.

4.2.4 Other details of domain decomposition

Domain decomposition can be performed in a way that sub-domains are either overlapped or non-overlapped. Overlapping domain decomposition methods include *the Schwarz alternating method* and *the additive Schwarz method*. In non-overlapping methods, the sub-domains intersect only on their interface. Non-overlapping domain decomposition methods are also called *iterative sub-structuring methods*.

Additionally, two important issues must be addressed in parallel DD; load balancing and inter-process(or) communications.

4.2.5 Domain decomposition in PETSc

In PETSC, all approaches are available. Domain decomposition in pre-processing step can be done using simple decomposition strategy. For example, DA automatically divide the domain along the stream-wise(x), then normal-wise(y), then span-wise(z) and assign each part to a separate processor. This can also be controlled manually in creating DAs. Using the PETSc interfaces to external libraries such as ParMETIS [30], advanced methods for unstructured grids can also be utilized.

In solving linear systems, overlapping and non-overlapping additive Schwarz methods are already implemented in PETSC as preconditioner.

4.2.6 Domain decomposition approaches used in *i*DNS

Since PETSc is an optimized library, default options mostly work well for most of the cases. In our calculations, default options in PETSc DA are kept in both pre-processing step and solving the linear systems.

Simple domain decomposition inherently implemented in PETSc DA for pre-processing and pre-defined linear solver with preconditioning in solving linear system were used during the computations. Load balancing is determined at the beginning. Inter-process(or) communications, i.e., communications among sub-domains, are handled using local ghost spaces. They are filled by explicit point-to-point communications from processes adjacent to each other.

4.3 Matrix Preconditioning and Linear Solvers

The implicit discretization of the equations results into a large sparse coefficient matrix that must be solved with an iterative method such as *Krylov subspace methods*. However, efficiency, robustness and convergence of such methods are not always guaranteed. It strongly depends on the eigenvalue spectrum of the sparse matrix to be solved. The most obvious cure here is to transform the original matrix into a one which has the same solution with better eigenvalue spectrum. This procedure is called *preconditioning* [34]. The resulting preconditioned matrix has a small condition number, and is likely to be easier to solve with an iterative solver.

Preconditioning or a preconditioned system can mathematically be expressed as,

$$M^{-1}Ax = M^{-1}b \qquad \textbf{(4.2)}$$

where M^{-1} is *the preconditioner matrix* which can be calculated explicitly from the original matrix A. An easy and effective way to obtain M is to decompose the original system into lower (L), upper (U) and diagonal (D) parts. This is called *incomplete factorization*. This decomposition also results in a residual matrix, $R = LU - A$. Then, M is available, for example, in the form of incomplete LU factorization $M = LU$ or in the form of incomplete Cholesky factorization $M = LL^T$, depending on the solution technique.

Since default preconditioning in PETSc is very-well optimizated for computational efficiency, incomplete LU factorization with no fill-in, *ILU(0),* is mostly preferred in our calculations. Some cases were also run with algebraic multigrid (AMG) preconditioning implemented in an external library Hypre [29] by calling it within PETSc via an interface. This AMG implementation in Hypre is called BoomerAMG. It gives slightly better results than ILU(0) preconditioning . However, initial setup of BoomerAMG takes longer and computationally more expensive. It should be noted that both can be further optimizated using inner parameters such as fill-in for ILU or multigrid levels for AMG. Our observation is only based on default parameters. One, of course, should seek for an optimum cost/accuracy ratio for the specific problem.

The combination of a preconditioner with a Krylov subspace method is the most modern approach to obtain the iterative solution of linear systems. This is done via KSP object within PETSC. The default KSP uses left preconditioned GMRES with a restart of 30, using modified Gram-Schmidt orthogonalization. KSP object must be created and set suitably, according to solver's needs in pre-processing step. It has many tuning parameters.

The default convergence test is based on the $l_2 - norm$ of the residual. Convergence (or divergence) is decided by three quantities: the decrease of the residual norm relative to the norm of the right hand side, $rtol$, the absolute size of the residual norm, $atol$, and the relative increase in the residual, $dtol$. As it is given in the PETSC manual [27], convergence is detected at iteration k if,

$$\|r_k\|_2 < max(rtol * \|b\|_2, atol) \tag{4.3}$$

where $\|r_k\|_2 = b - Ax_k$. Divergence is detected if,

$$\|r_k\|_2 > dtol * \|b\|_2 \tag{4.4}$$

The defaults are $rtol = 10^{-5}$, $atol = 10^{-50}$, $dtol = 10^5$, and maximum iterations are limited to 10^5. These parameters can also be specified by the user, either in the routine or in run-time.

4.4 A Further Note on Domain Decomposition and Solving Linear Systems in PETSc

Since grid partitioning and matrix partitioning are both implemented in PETSc and work together, the connection (or the distinction) between them should be clarified.

All variables are stored in *Vec* object in PETSc. After grid is partitioned based on either simple or advanced domain decomposition technique in pre-processing step, each process(or) works on its portion of Vec in order to evaluate required functions.

Each process(or) also generates its own portion (row) of matrix (in *Mat* object). Each processor needs to insert only elements that it owns locally. Any non-local elements will be sent to the appropriate processor during matrix assembly. Then matrix is assembled using the *assembly* routines defined in PETSc. Matrix assembly is a three step process; each process sets or adds values, begin communication to send values to the correct process, and complete the communication. In assembling matrix, contiguous rows of vectors or matrix in linear system to be solved are assigned to processes, starting from the process with rank zero by PETSc. Matrix decomposition in contiguous chunks is simple and makes coupling of *Mat* object in PETSc with external solvers easy.

4.5 Parallel Treatment of Boundary Conditions with PETSc

Petsc also handles boundaries using ghost cells outside the physical domain. One or two layers of ghost cells surrounding the physical domain can be defined in DA at the pre-processing step. Similarly, the type of the boundaries must be specified, when creating DA. The Fortran support of DA for three-dimensional boundary conditions in the version of PETSc (petsc-3.1-p8) used in this thesis is limited to periodic boundaries.

A special treatment is needed to handle the wall boundaries. In order to perform this with PETSc, DA are defined to have periodic boundary conditions and the periodic ghost points are used to store the needed values of variables. There are no equations associated with them. Ghost cells for periodicity are used to store *exterior* to boundary values that there is no periodicity. For an example of handling non-periodic boundaries, see petsc-3.1-p8/src/snes/examples/tutorials/ex31.c in PETSc tarball.

It should be noted that new versions of PETSc come an advanced boundary conditions support for C, as well as Fortran. It should be preferred for implementation.

4.6 Parallel Performance of the Solver

There are number of metrics to measure the parallel performance of a solver, such as total execution time, speedup, efficiency, and total overhead [35]. Each of them gives an idea about certain properties of the parallel code. Here, speedup and efficiency are introduced to demonstrate the parallel performance.

Speedup (S) is defined as the ratio between sequential execution time (t_s) and parallel execution time (t_p) for a given problem size n,

$$S(n,p) = \frac{execution\ time\ using\ one\ core}{execution\ time\ using\ p\ cores} = \frac{t_s(n)}{t_p(n)} \tag{4.5}$$

Usually, speedup is an increasing function of problem size. For an ideal system, it is equal to number of cores, p, . Efficiency (E) is defined as the ratio of speedup to the number of cores,

$$E(n,p) = \frac{\textit{execution time using one core}}{\textit{execution time using p cores} \times p} = \frac{t_s(n)}{pt_p(n)} = \frac{S(n,p)}{p} \qquad \textbf{(4.6)}$$

Efficiency is equal to one in ideal case. Additionally, a solver is scalable, if it maintains the efficiency constant by simultaneously increasing number of processors and problem size [36].

DNS of TGV is chosen for the parallel performance analysis. Omitting the pre-processing and post-processing parts, only the solver(core) part of *i*DNS is evaluated using the profiling routines in PETSc. The code was run for a fixed number of time-steps. In order to obtain speedup and efficiency for a given problem size (128^3), run performed with an increasing number of cores. Then, it is repeated for a higher problem size (256^3).

Results are presented in Fig. 4.3 and Fig. 4.4, respectively. The obtained values are quite acceptable. It is seen that better speedup and efficiency values are achieved with higher problem size. PETSc is very-well scalable up to where number of processors are equal to problem size. This also shows that PETSc can handle the very large problems and it is a very-well load-balanced and optimized library.

There are also many other parameters that affect the parallel performance such as processor/memory architecture and communication type among nodes. These are currently out of scope of this thesis and will not be considered here.

PETSc also provides detailed metrics via run-time option -*log_summary*. In this output, many other details such as communication and computation times, load balancing analysis, and number of floating point operations (*nflops*) are also summarized. This is also very useful for analyzing, when developing a parallel solver using PETSc.

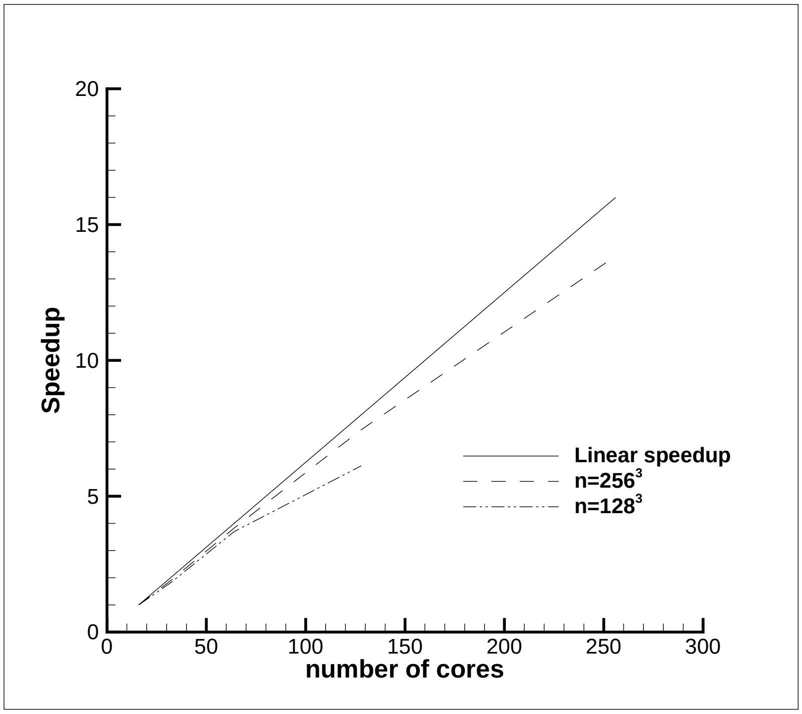


Figure 4.3: Speedup values obtained by two different problem sizes, relative to 16 cores. Higher problem domain yields better speedup value.

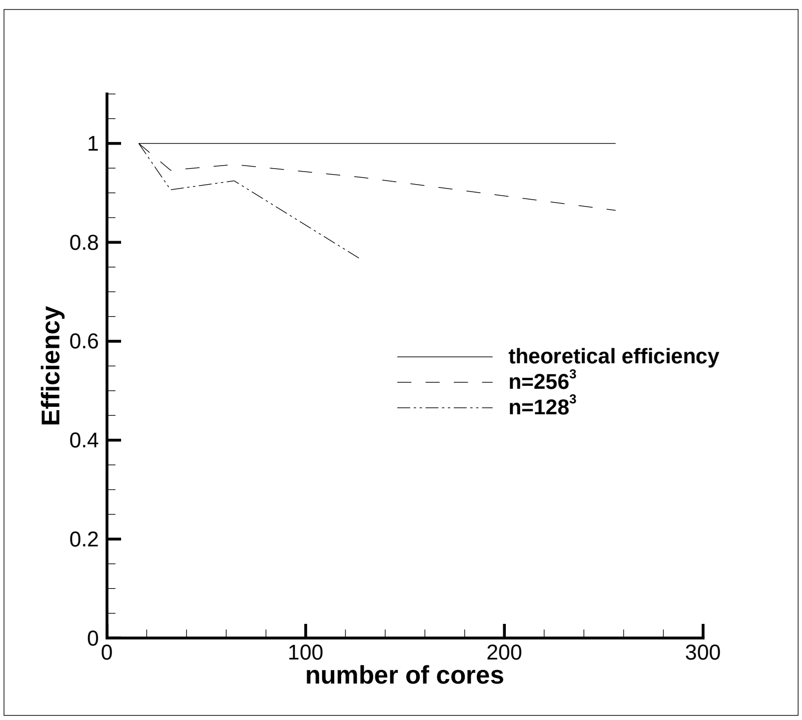


Figure 4.4: Efficiency values obtained by two different problem sizes, relative to 16 cores.

5. DNS OF TAYLOR-GREEN FLOW

In this chapter, Direct Numerical Simulation of Taylor-Green Vortex flow is performed using *i*DNS. After giving an introduction, initial and boundary conditions are defined in the following sections. The simulation results are presented and successfully compared to previous numerical studies found in the literature.

5.1 Introduction

In fluid dynamics, *the vortex stretching* is the lengthening of vortices in three-dimensional fluid flow, associated with a corresponding increase of the component of vorticity in the stretching direction due to the conservation of angular momentum [37], as illustrated in Fig.5.1. In two-dimensional flows, the stretching does not occur, because the vortex lines are perpendicular to the velocity.

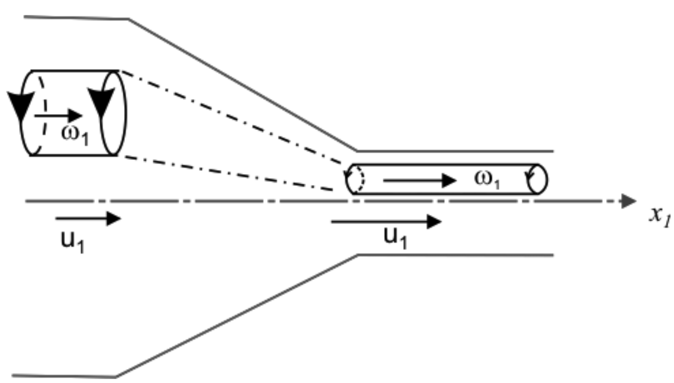


Figure 5.1: An illustration of vortex stretching taken from [37]. As vortex tube stretches, its length and vorticity increase due to the conservation of angular momentum.

The vortex stretching is also the core mechanism on which the turbulence energy cascade relies. In general, the stretching mechanism implies thinning of the vortices in the direction perpendicular to the stretching direction due to volume conservation of fluid elements. As a result, the radial length scale of the vortices decreases and the larger flow structures break down into smaller structures. The process continues until the small scale structures are small enough to the extent where their kinetic energy is overwhelmed by the fluid's molecular viscosity and dissipated into heat [37].

The simplest fundamental case that has been used as a prototype for vortex stretching and the consequent production of small-scale eddies, resulting into homogeneous isotropic turbulence and its decay, is the vortex system introduced by Taylor and Green [38].

The dynamics of the inviscid and the viscous Taylor-Green Vortex (TGV) have been discussed in detail using pseudo-spectral DNS by Brachet et. al. [39] and Brachet [40], DNS/LES by Fauconnier et. al. [41], implicit LES (ILES) by Drikakis et al. [42, 43], a spectral method with WENO by Shu et. al. [44], and more recently by van Rees et. al. [45] using Discontinuous Galerkin method at Reynolds numbers up to 5000. The results of the pseudo-spectral DNS methods are accepted as benchmark and are widely used for the comparison.

It was shown that there are two stages in the evolution of the TGV flow. At early times, i.e. up to a non-dimensional time between 3 and 4, well-organized structures are formed and the flow remains essentially inviscid and is characterized as *laminar*. Here, the dynamics in the flow is quasi-two dimensional. At later times, the non-linear terms in the Navier-Stokes equations start to generate successively smaller structures which interact with the large scales. The effect of viscosity can no longer be neglected and highly distorted, dissipative structures develop. If the Reynolds number is large enough, this results into process of vortex-stretching and eventually into a breakdown of the large scales into smaller ones which is interpreted as *turbulence*. Therefore, the Taylor-Green flow is assumed to describe the fundamental mechanism of transition into isotropic turbulence. Obviously, this transitional behavior is determined entirely by the choice of the Reynolds number. Brachet et. al. [39] observed that for Re $\geq$ 500, the small-scale structures in the Taylor-Green flow undergo profound changes. Indeed, for high Reynolds numbers, the turbulent flow becomes nearly isotropic at non-dimensional time ≈ 7 with no memory to the initial conditions, whereas the dissipation rate reaches a maximum at non-dimensional time ≈ 9 and the kinetic energy follows a power law spectrum closer to Kolmogorov's -5/3. The dissipation peaks are consistent for the higher Reynolds numbers. This may correspond to viscosity independent limit [46]. However, for low Reynolds numbers, the maximum dissipation occurs earlier, whereas the flow retains some of its initial anisotropy at all times. Clear inertial range behavior was observed once Re ≥ 1000. The flow structures as well as the dissipation rate eventually decay to zero, then dies out due to viscosity and the lack of an external energy source. For the inviscid case, it is not clear whether finite-time inviscid singularity exists [44]. Numerically, kinetic energy should be preserved during the inviscid simulations, due to the lack of viscous effects. In general, all the methods predict the global dynamics of the Taylor-Green flow reasonably well, but differences are observed mostly in the form of the dissipation peak structure.

The aim of this chapter is to assess the behavior of the present non-dissipative, fully implicit, and discrete energy conserving DNS algorithm for simulating the transition to turbulence in TGV flow. The next section describes the initial and boundary conditions. The results are presented in the following section and compared to the previous studies in literature.

5.2 Numerical Setup of the Flow

The initial solenoidal velocity field for the three-dimensional incompressible vortex is defined by,

$$u = \quad u_0 \sin(\kappa x)\cos(\kappa y)\cos(\kappa z), \tag{5.1a}$$

$$v = -u_0 \cos(\kappa x)\sin(\kappa y)\cos(\kappa z), \tag{5.1b}$$

$$w = \quad 0 \tag{5.1c}$$

κ is the wave-number, $\kappa = 2\pi/L$. L is the domain length and u_0 is the reference velocity taken as 1. The initial pressure field, p, is provided by the solution of the corresponding Poisson equation for the given velocity field in the following form,

$$p = p_0 + \frac{\rho_0 u_0^2}{16}[\cos(2kx) + \cos(2ky)][2 + cos(2kz)] \tag{5.2}$$

with the reference pressure, p_0 chosen as 100 to limit the Mach number,

$$M = \frac{u_0}{\sqrt{\gamma \frac{p_0}{\rho_0}}} \quad \textbf{(5.3)}$$

to approximately 0.08. ρ_0 is the reference density. The initial density, ρ, is uniform and is set to unity everywhere. The initial density is also chosen as the reference density.

Boundary conditions are triply-periodic on cubical domain which has 2π length in each direction. Since the non-dimensional velocity is bounded by the unity and the reference length $l_0 = L/2\pi = 1$, the Reynolds number (Re) is defined only as the inverse of the kinematic viscosity, $1/\nu$. For the viscous case, it is set to 1600. These choices lead to *the eddy-turnover time*, τ, of order unity. Simulations are followed up to $t/\tau = 60$ with a non-dimensional time step, 10^{-3}. The symmetries of the Taylor–Green flow will not be considered here.

Since the physics of the turbulent Taylor-Green flow are fundamentally similar to the isotropic turbulence, the minimum grid resolution requirement in case of homogeneous isotropic turbulence in a periodic box with 2π length can be used to determine the grid resolution requirement for the current DNS. Following the calculations given in Fauconnier [47], the Reynolds number can be related to the Taylor micro-scale Reynolds number as,

$$Re_\lambda = \sqrt{2Re} \approx 56 \quad \textbf{(5.4)}$$

For the given Re_λ, the minimum number of uniformly distributed cells, N per each direction is,

$$N \approx \frac{8}{\pi}\left(\frac{3}{20}\right)^{\frac{3}{4}} \frac{\pi}{\zeta}\left(\frac{12}{Re_\lambda} + 0.43\right) Re_\lambda^{\frac{3}{2}} \quad \textbf{(5.5)}$$

where ζ denotes the ratio of the grid spacing to the Kolmogorov length scale. Substituting $\zeta = 2.1$ [48] and $Re_\lambda \approx 56$ into the Eq. **5.5**, it gives $N \approx 252$. This can be represented as $N = 2^8 = 256$. Then, the grid cut-off wave-number, $\kappa_{max} = \frac{\pi}{\Delta}$ is 128, where Δ is grid spacing. Hence, the DNS of the TGV flow at $Re = 1600$ $(Re_\lambda \approx 56)$

is performed on the uniform grid with 256^3 cells. That is only the first 128^3 Fourier modes are resolved.

5.3 Results and Discussions

As turbulence develops in time, initially well-organized structures start to break into smaller ones and lose their initial symmetries due to the energy cascade and the vortex stretching for the viscous case. Then, becomes heavily distorted and finally breaks down around . For $t \geq 9$, the flow is turbulent and nearly-isotropic. This physics is well captured by the algorithm, as shown in Fig.5.2. Visualizations are based on Q-criterion [49] which is calculated as,

$$Q = -\frac{1}{2}u_{i,j}u_{j,i} = \frac{1}{2}\left(\|\Omega\|^2 - \|S\|^2\right) \tag{5.6}$$

where $\|S\| = [tr(S_{ij}S^t_{ij})]^{1/2}$ and $\|\Omega\| = [tr(\Omega_{ij}\Omega^t_{ij})]^{1/2}$. S_{ij} and Ω_{ij} are the symmetric and anti-symmetric parts of the velocity gradient tensor, respectively.

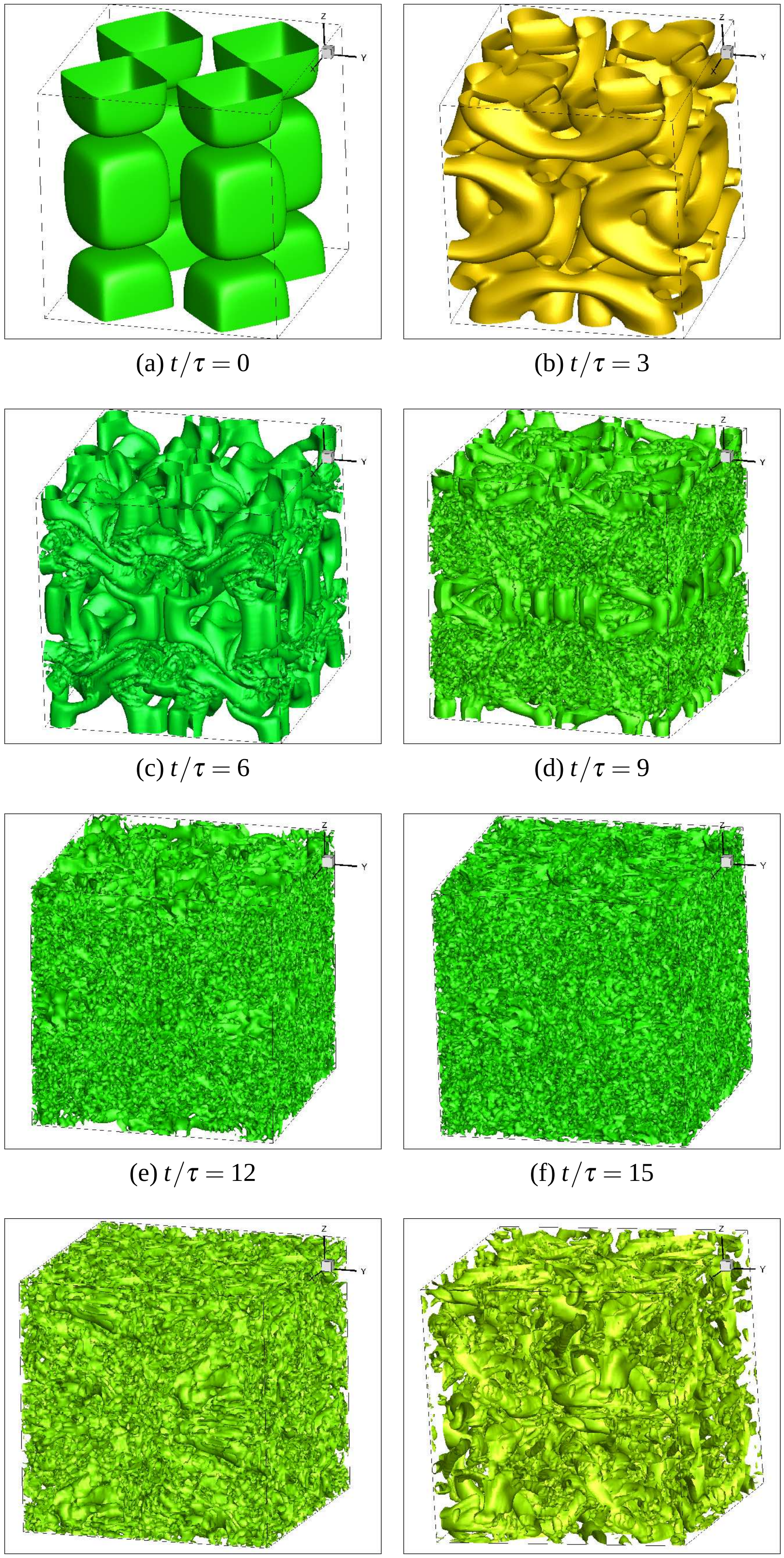
Following Hahn [50], some other quantities are also introduced to analyze the flow field such as the temporal evolution of the mean kinetic energy, the mean kinetic energy decay rate, the mean enstrophy, the energy spectrum at $t/\tau = 9$, Taylor micro-scales and velocity structure functions.

The mean kinetic energy, K, is calculated as,

$$K = \frac{1}{2}\langle\rho|\mathbf{u}|^2\rangle \tag{5.7}$$

where $< \cdot >$ represents the spatial average over all directions. As in time, it decreases under the effect of the viscous forces, as seen in Fig.5.3. It should be preserved during the inviscid simulations due to the lack of viscous effects, if the algorithm is conservative.

The mean kinetic energy decay rate is calculated as $-dK/dt$, which is the slope of the kinetic energy decay. Increasing value of the mean kinetic energy decay rate is due to increasing vortical interactions. Its time evolution is given in Fig.5.4.



(a) $t/\tau = 0$ (b) $t/\tau = 3$

(c) $t/\tau = 6$ (d) $t/\tau = 9$

(e) $t/\tau = 12$ (f) $t/\tau = 15$

(g) $t/\tau = 30$ (h) $t/\tau = 60$

Figure 5.2: Time evolution of the Q-iso surfaces in TGV flow. Q is set to 0.02 for all snapshots.

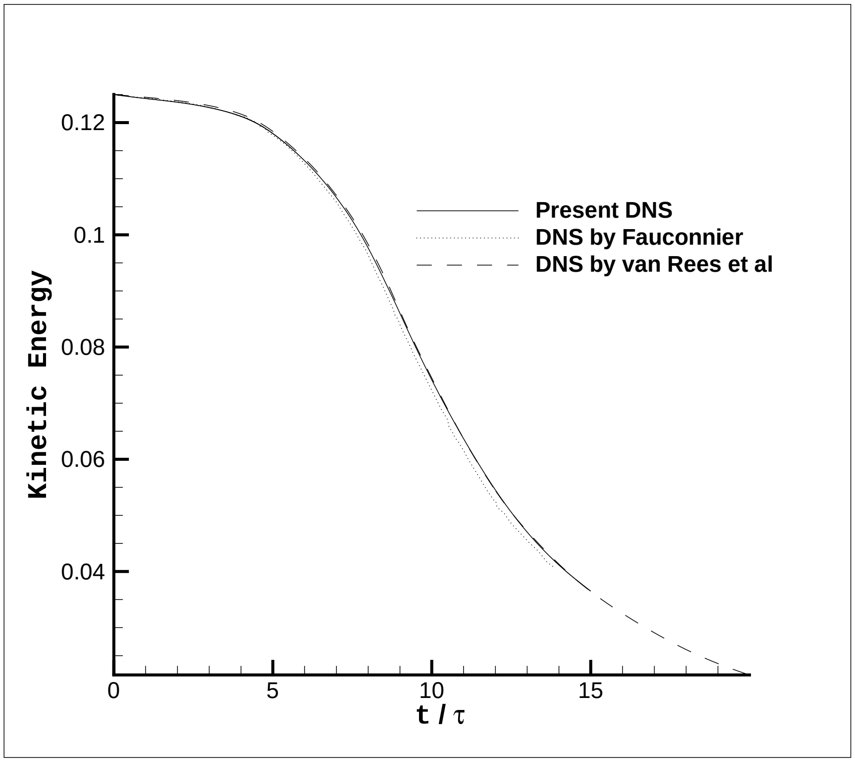


Figure 5.3: Temporal evolution of the mean kinetic energy.

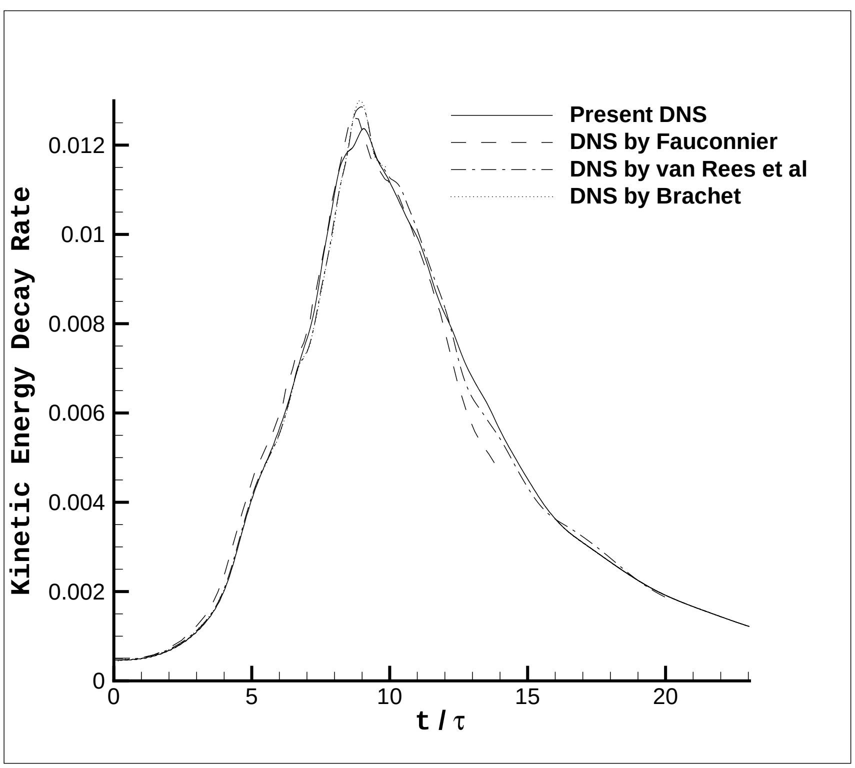


Figure 5.4: Temporal evolution of the mean kinetic energy decay rate.

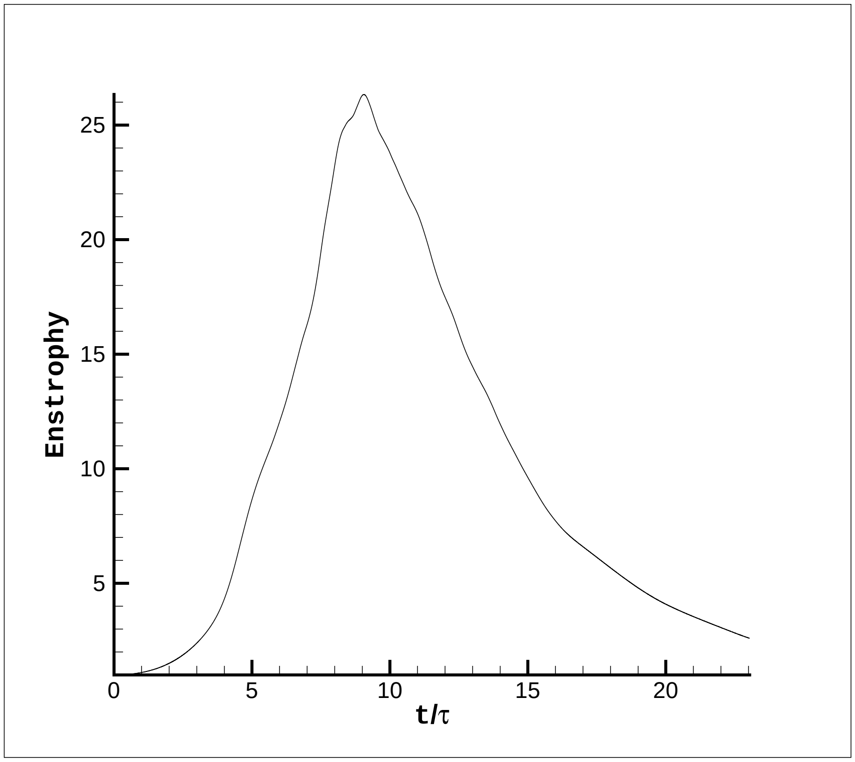


Figure 5.5: Temporal evolution of the enstrophy.

The mean enstrophy, ω, is given as the square of vorticity,

$$\langle \omega^2 \rangle = \langle |\nabla \times \mathbf{u}|^2 \rangle \tag{5.8}$$

Enstrophy is the resolving power of a numerical scheme that is a measure of its ability to represent the flow physics accurately on a finite number of grid cells [44]. As seen in Fig. 5.5, it follows a path similar to the mean kinetic energy decay rate. For the inviscid case, an unbounded increase should be observed.

As Kolmogorov stated [51], for homogeneous isotropic turbulence $E(k) \approx k^{-5/3}$ within the inertial sub-range. In order to show the transition to turbulence occurs in TGV flow at $t/\tau \approx 9$, E(k) is also presented in Fig. 5.6. The scaling is not fully achieved. Obviously, a much clear inertial range behavior can be observed at higher Reynolds number [39]. An increase is observed in E(k) at high wave-numbers. The kinetic energy of the flow is not efficiently dissipated at high wave-numbers. This can be associated to the compressibility effects included by the non-divergence free part of the velocity field.

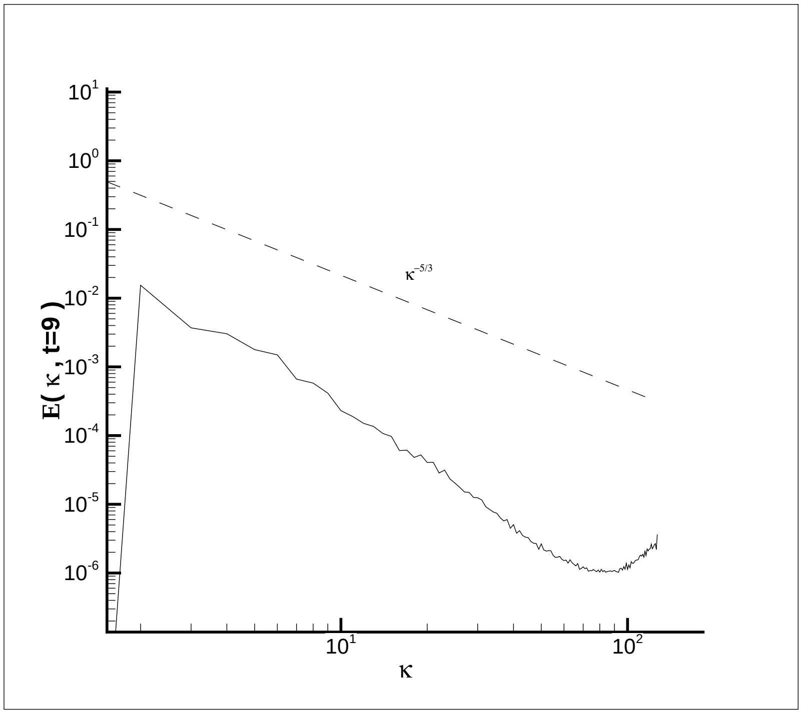


Figure 5.6: The energy spectrum at $t/\tau = 9$.

Our results are well-compared to Brachet et al. [39], Fauconnier et al. [41], and Hahn [50] on 256^3 grid resolution for $Re = 1600$. The transition time, when the dissipation rate reaches its maximum, is correctly predicted at $t/\tau \approx 9$. However, the peak values of the mean kinetic energy decay rates and the forms of the peak structures in Fig. 5.7 slightly differ from each other. This can be due to the evaluating of gradients in methods.

Taylor micro-scale averaged over the three homogeneous spatial directions is given as,

$$\lambda = \frac{(\lambda_x + \lambda_y + \lambda_z)}{3} \tag{5.9}$$

where

$$\lambda_x = \sqrt{\frac{\langle (u)^2_{rms} \rangle}{\langle (\frac{\partial u}{\partial x})^2 \rangle}}; \qquad \lambda_y = \sqrt{\frac{\langle (v)^2_{rms} \rangle}{\langle (\frac{\partial v}{\partial y})^2 \rangle}}; \qquad \lambda_z = \sqrt{\frac{\langle (w)^2_{rms} \rangle}{\langle (\frac{\partial w}{\partial z})^2 \rangle}} \tag{5.10}$$

The Taylor micro-scale is a length scale which was first introduced by Taylor [52]. It does not have an exact physical meaning [48]. However, it is often used to define a Reynolds number that characterizes grid turbulence [37]. It is an intermediate length

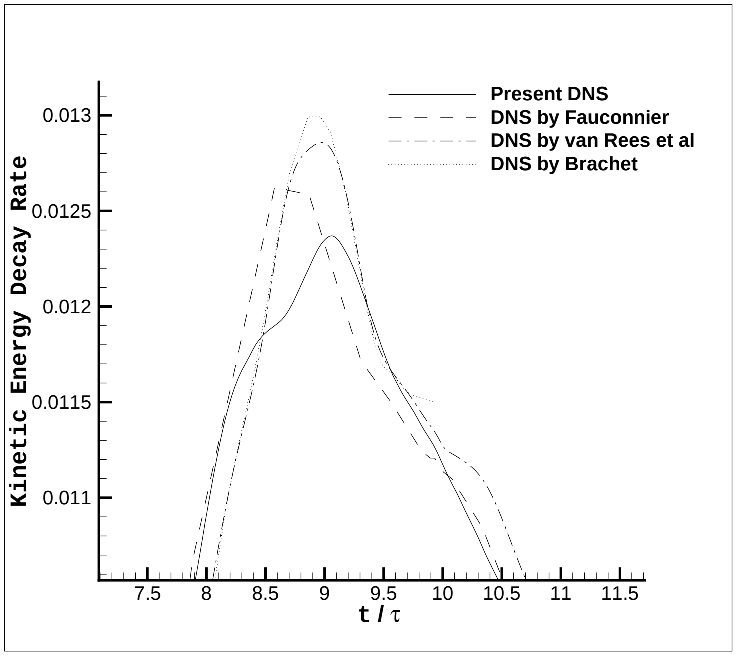


Figure 5.7: Close-up view for $-dK/dt$.

scale between integral and Kolmogorov's length scales. It is roughly assumed that, below the Taylor micro-scale, fluid viscosity significantly affects the dynamics of turbulent eddies in the flow, the turbulent motions are subject to strong viscous forces, and kinetic energy is dissipated into heat [53]. Time evolution of the Taylor micro-scale is plotted in Fig. 5.8.

Skewness $(n=3)$ and flatness $(n=4)$ are the high-order velocity-derivative moments (structure functions) given as,

$$S(n) = \frac{(S_x(n)+S_y(n)+S_z(n))}{3} \qquad \textbf{(5.11)}$$

where

$$S_x(n) = (-1)^n \frac{\langle (\frac{\partial u}{\partial x})^n \rangle}{\langle (\frac{\partial u}{\partial x})^2 \rangle^{\frac{n}{2}}}; \quad S_y(n) = (-1)^n \frac{\langle (\frac{\partial v}{\partial y})^n \rangle}{\langle (\frac{\partial v}{\partial y})^2 \rangle^{\frac{n}{2}}}; \quad S_z(n) = (-1)^n \frac{\langle (\frac{\partial w}{\partial z})^n \rangle}{\langle (\frac{\partial w}{\partial z})^2 \rangle^{\frac{n}{2}}} \qquad \textbf{(5.12)}$$

The skewness characterizes the rate at which enstrophy increases by vortex stretching. Whereas, the flatness is a measure of the intermittency of the vorticity field driven by vortex stretching and folding.

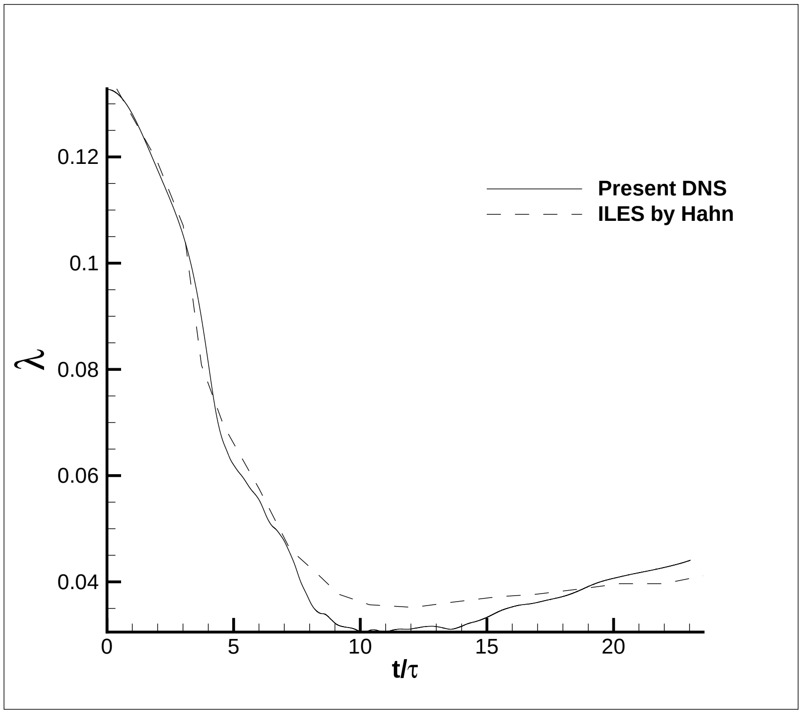


Figure 5.8: Time evolution of the normalized Taylor micro-scale.

Their average values given in the literature change from -0.2 to -0.7 for skewness, and from 3 to 40 for flatness [54]. Our results given in Fig. 5.9 and Fig. 5.10 are in agreement with these values.

The evolution of the kinetic energy decay and the enstrophy for the inviscid case are given in Fig. 5.11 and in Fig. 5.12, respectively. In the absence of viscosity, kinetic energy should be preserved during the course of simulation. However, both the inherent dissipation of the algorithm and the accumulated numerical errors may decrease the kinetic energy in time. Inviscid simulations can be used to measure the dissipation included by the numerical scheme. The kinetic energy is very well-preserved by the present algorithm during the course of long-term inviscid simulation and the enstrophy unboundedly grows in time. This shows the capability of the algorithm to capture the flow dynamics on finite number of cells.

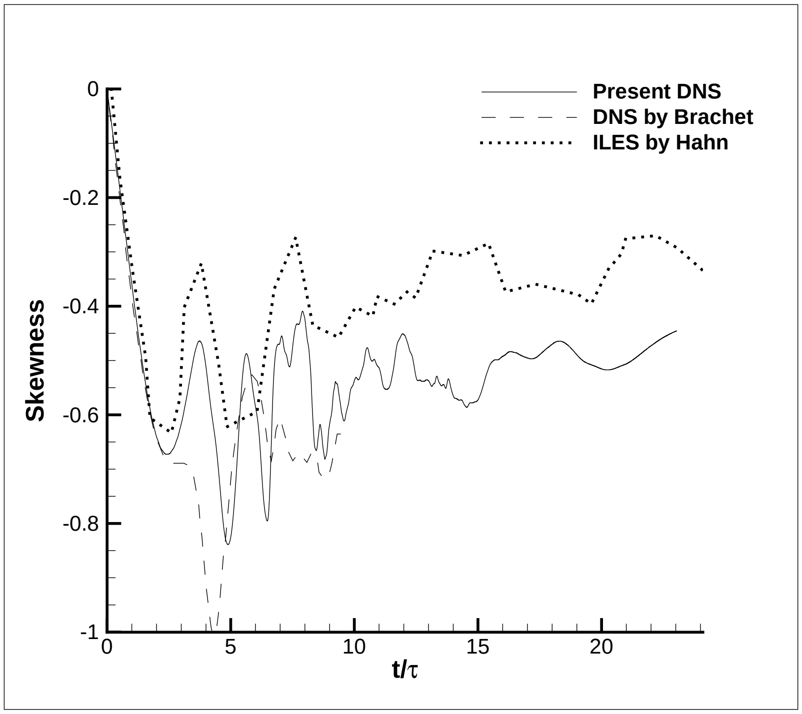


Figure 5.9: Time evolution of the skewness.

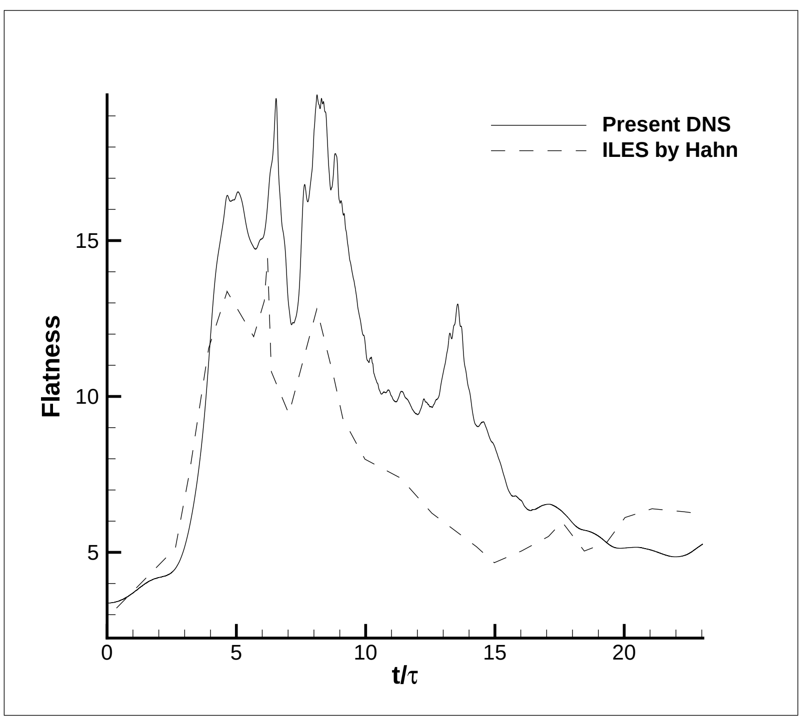


Figure 5.10: Time evolution of the flatness.

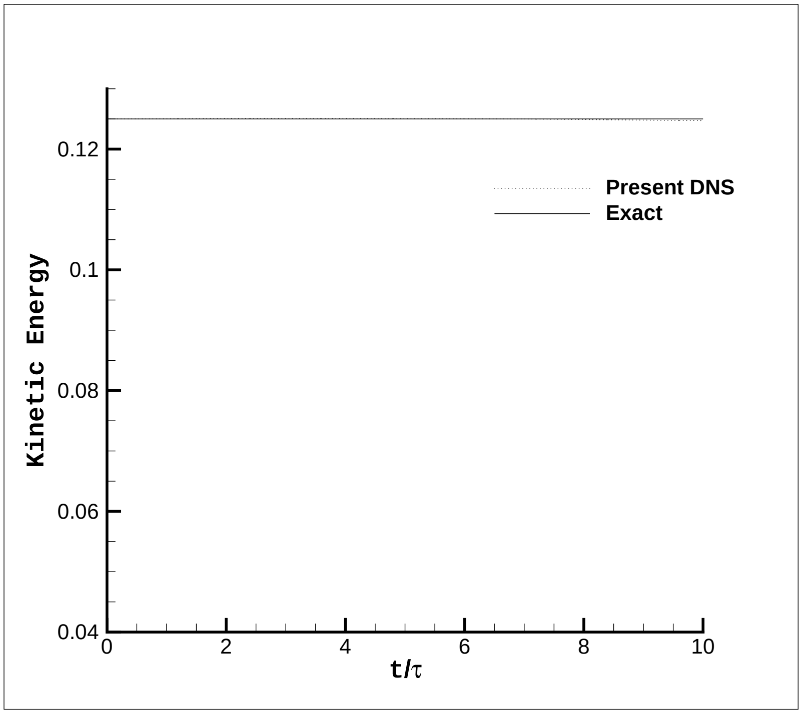


Figure 5.11: Time evolution of the mean kinetic energy for the inviscid case.

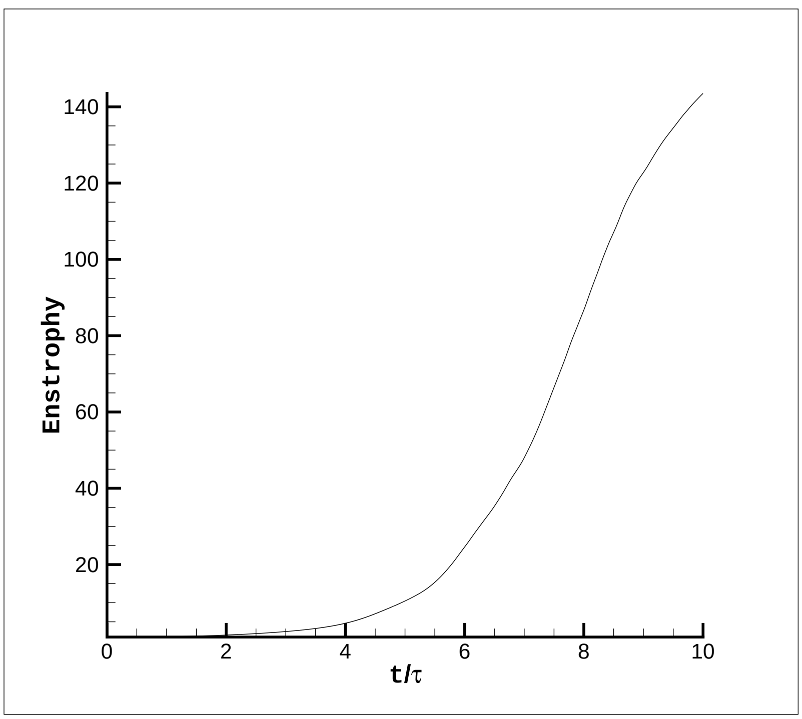


Figure 5.12: Time evolution of the enstrophy for the inviscid case.

6. DNS OF TURBULENT SHEAR LAYER

In this chapter, Direct Numerical Simulation of initially turbulent temporal shear layer is performed using *i*DNS. Quasi-incompressible and compressible cases are considered. After giving an introduction, initial and boundary conditions are defined in the following sections. The simulation results are presented and successfully compared to both experimental and numerical studies found in the literature. Compressibility effects on the growth rate of the turbulent mixing are also studied.

6.1 Introduction

Turbulent shear layers are interest of many physical and engineering flows such as jets, wakes and mixing layers. They are induced by strong gradient of shear stress.

It is very important and is often necessary to understand the physical mechanisms behind the such kind of flows, for example, in order to control the mixing processes in supersonic combustion or to control the flows emerging in downstream of an airfoil.

Although their relatively simple configuration, more complex physical phenomena are included by these flows. They are mostly dominated by large scale quasi-2D organized structures. However, they can rapidly undergo transition to turbulence in 3D.

In a mixing layer, two stream of fluids which have different speeds emerge and evolve in a way that qualitatively described above. It can be studied both spatially and temporally. In spatially-evolving mixing layers, mixing layer thickness of two emerging stream of fluids develop in spatial coordinate in stream-wise direction. Such development requires large extend of domains. This is computationally much more expensive to track than the temporally-evolving mixing layers where the thickness of the layer increases as a function of time rather than as a function of the stream-wise coordinate. As explained in [55], the temporal mixing layer simulations have two consequences. Since the flow evolves in time, it is not possible to do time averaging to obtain converged statistical quantities. Averaging is done in the homogeneous spatial directions (stream-wise and span-wise). Additionally, the flow evolution is strongly dependent on the initial conditions.

The mixing layers were experimentally investigated by Spencer and Jones [56] and Bell and Mehta [57]. They provided experimental data for the Reynolds stress distributions using a mixing-layer wind tunnel where two separate streams come and merge at the end of the splitter inside a tunnel.

Early DNS of two-dimensional incompressible temporal mixing layer starts with Riley and Metcalfe [58]. First high-resolution and comprehensive three-dimensional simulations using DNS were performed by Rogers and Moser [59], Moser and Rogers [60] and Rogers and Moser [61]. They used spectral methods and studied the evolution of the large-scale vortical structures, the mechanisms responsible for the growth, the onset of transition to turbulence and the self-similarity of the mixing layers.

Because of the recent developments in supersonic flight technology, high-speed mixing layers have received an attention. These propulsion systems are based on supersonic combustion where mixing plays an important role for efficiency. So, the physical understanding of the compressibility effects on mixing and transition to turbulence are crucial for developing efficient systems.

Bogdanoff [62] and Papamoschou and Roshko [63] experimentally studied the evolution of the compressible mixing layers. They introduced the *convective Mach number* in order to define the compressiblity effects on the evolution of mixing layers. Elliot and Samimy [64], Goebel and Dutton [65], and Clemens and Mungal [66] were further studied the compressible mixing layers experimentally.

All studies pointed out a strong reduction effect on growth rate of mixing layers, when the convective Mach number is higher than 0.6. In addition to this, structural changes were also found [64, 66]. Instead of simple roll-ups, large-scale coherent three-dimensional vortical structures were observed at high convective Mach numbers.

In order to clarify the physical mechanisms behind the reduction effect on the growth rate, Vreman [67], Vreman et al. [68], Sarkar [69], Freund [70], and Pantano and Sarkar [71] performed DNS of compressible mixing layers up to convective Mach number 1.8. They found that the effect is related to decrease in pressure-strain correlation via reduced pressure fluctuations.

In addition to these pressure-strain models, some other attempts have also been made such as stability analysis, models based on the dilatation terms, and the sonic-eddy models [72]. However, none of them give a complete picture of the mechanism responsible for this effect. It is still under investigation due to the current technological importance.

The aim of this chapter has two folds; to show the capability of the algorithm's ability to evolve a linear perturbation into nonlinear hydrodynamic turbulence, and to analyze the effect of compressibility on the growth rate of the mixing layers using an implicit non-dissipative algorithm.

6.2 Numerical Setup of the Flow

6.2.1 Initial and boundary conditions

The initial conditions follow the DNS paper of Pantano and Sarkar [71]. As common to many previous setups, flow is initialized using a hyperbolic tangent profile for the mean stream-wise velocity ($\bar{u}$),

$$\bar{u}(y) = \frac{\Delta u}{2} \tanh\left(-\frac{y}{2\delta_\theta(0)}\right) \quad \textbf{(6.1)}$$

where $\delta_\theta(0)$ is the *initial momentum thickness*. This defines a two parallel streams moving in opposite directions, as it is shown in Fig. 6.1. Δu is the velocity difference between lower and upper streams. Each stream is moving to opposite directions with same velocity, $\frac{\Delta u}{2}$. The normal-wise ($\bar{v}$) and the span-wise ($\bar{w}$) mean velocities are set to zero. The initial densities are equal for both streams and are set to unity. The initial pressure, p_0, is uniform as well. It is set to give the speed of sound as unity.

The Reynolds number, Re_θ, based on the initial momentum thickness, the velocity difference, and the average dynamic viscosity is 160 and the Reynolds number, Re_ω, based on the initial vorticity thickness, $\delta_\omega(0)$, is 680. The definitions of δ_θ and δ_ω are given in the following sections.

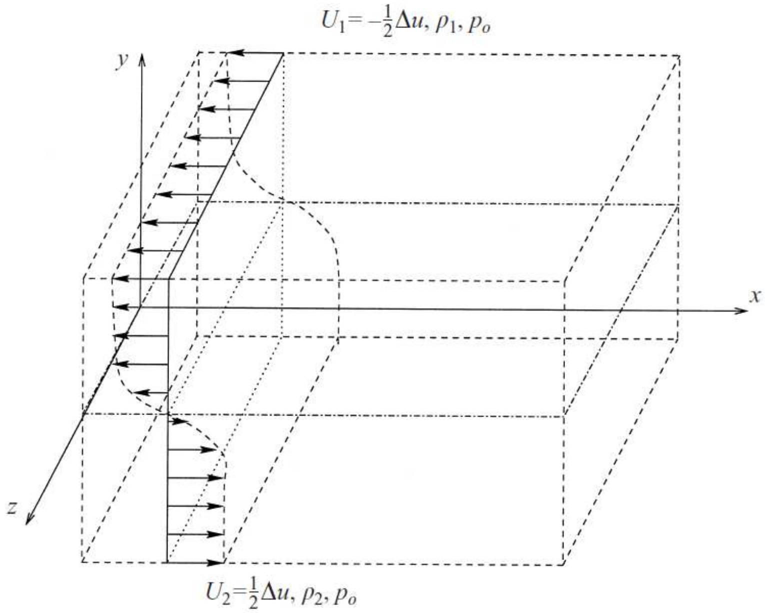


Figure 6.1: A representation of the initial conditions for TSL problem [71].

The ratio of specific heats, γ, is taken as 1.4. The Prandtl number, Pr, is 0.72.

Only the case where the density ratio between streams is equal to 1 from Pantano and Sarkar [71] was chosen here to perform the simulations with different convective Mach numbers. The idea of the convective Mach number was first introduced by Bogdanoff [62] and further used by Papamoschou and Roshko [63] in order to define a unique Mach number and to identify the compressibility effects on the shear layer growth rate. For equal densities and specific heats, it can be written as,

$$M_c = \frac{\Delta u}{c_1 + c_2} \qquad \textbf{(6.2)}$$

where c_1 and c_2 denote the speeds of sound for each stream. Since the speeds of sound are fixed to unity and the streams have equal velocities, $\Delta u/2$, in opposite directions, the free stream Mach numbers are also the same and used as the reference Mach number in our calculations.

The initial density, velocity difference, pressure, and momentum thickness are selected as the reference density, velocity, pressure, and length scale. The non-dimensional temperature field is obtained from the non-dimensional EOS given in Eq. **2.16**. Similarly, the non-dimensional dynamic viscosity can be calculated and updated in the code using the two-coefficient Power Law of viscosity given in Eq. **2.6**. Also, the reference time scale is,

$$\tau = \frac{\delta_\theta(0)}{\Delta u} \tag{6.3}$$

Boundary conditions are periodic in stream-wise(x) and span-wise(z) directions. Inviscid slip-wall boundary conditions as defined in Blazek [1] are used at bottom and top boundaries in normal-wise(y) direction. Face-normal velocities (V_n's) are also required special treatment and must be set to zero on inviscid slip-walls.

The non-dimensional time-steps are set to 6.5×10^{-3} for Mach number 0.3, and 1×10^{-3} for Mach number 1.1.

6.2.2 Computational requirements

256 cores were used for runs. Each core has approximately 2 GB of RAM. The total wall-clock time for each computation is between 350 and 500 hours, depending on the M_c.

6.2.3 Grid details

The same problem domain with Pantano and Sarkar [71] is used for our computations. The domain is very large, especially in streamwise(x) direction, in order to allow the flow to evolve into self-similar state. This resolution is also enough to represent the large scale structures in the flow, as shown by the authors using the calculation of the integral length scales (which are sufficiently small compared to the domain size in homogeneous directions) and two-point correlations (which are decorrelated over half the domain size in homogeneous directions) during the self-similar state. The domain size is given in terms of initial momentum thickness as, $L_x \times L_y \times \times L_z = 345\delta_\theta(0) \times 172\delta_\theta(0) \times 86\delta_\theta(0)$ with a corresponding resolution $N_x \times N_y \times \times N_z = 512 \times 256 \times 128$ for all cases considered. The initial momentum thickness is set to 0.093. The uniform orthogonal grid is used. Neither grid stretching nor sponge layers are employed.

6.2.4 Generation of the initial turbulent fluctuating velocity field

In addition to these mean values, turbulent three-dimensional velocity fluctuations are superimposed on the initial mean velocity components. The fluctuations are generated based on a technique described by Davidson [73]. This technique essentially relies on the approach first proposed by Kraichnan [74]. Using the notation in the paper of Davidson [73], the fluctuating velocity field, u_i', can be represented as a sum of N random Fourier modes,

$$u_i'(x_j) = 2\sum_{n=1}^{N} \hat{u}^n \cos(\kappa_j^n x_j + \psi^n)\sigma_i^n \quad \textbf{(6.4)}$$

where $\hat{u}^n$, ψ^n, and σ_i^n are the amplitude, phase and direction of the n^{th} Fourier mode, respectively. x_j denotes the coordinates in physical space. The magnitude of the direction vector of the Fourier modes, $|\sigma_i^n|$, is equal to unity. The wave-number vector κ_i^n is chosen randomly in wave-space in order to ensure isotropy of the generated velocity field. The wave-number vector κ_i^n and the direction vector σ_i^n are also orthogonal in physical space for each wave number n, i.e., $\sigma_i^n k_i^n = 0$. Fig. 6.2 shows the wave-number space geometry for n^{th} Fourier mode.

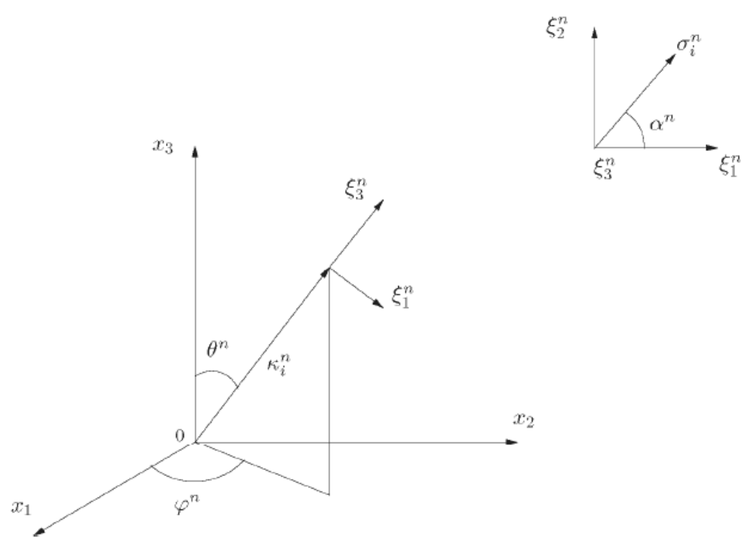


Figure 6.2: The wave-number vector, κ_i^n and the velocity unit vector, σ_i^n are orthogonal (in physical space) for each wave number n [73].

For each Fourier mode n, random angles φ^n, α^n, and θ^n and random phase ψ^n are created. While φ^n and θ^n determine the direction of the wave number vector κ_i, α^n

denotes the direction of the fluctuating velocity vector u'_i. They are produced randomly with probability functions which ensure that the wave number vector is uniformly distributed on the surface of the sphere shown in Fig. 6.3.

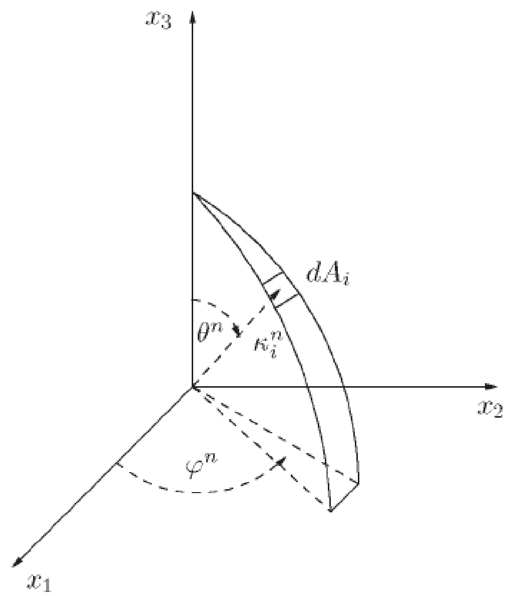


Figure 6.3: The probability of a randomly selected direction of a wave in wave-space is the same for all dA_i on the shell of a sphere [73].

The wave number space is equally divided into N number modes between κ_{max} and κ_1. κ_{max} is defined based on mesh resolution $\kappa_{max} = \frac{2\pi}{2\Delta}$, where Δ is the grid size. κ_1 is set to 1. Then, the wave number vector and the unit direction vector are calculated according to Fig. 6.3 as,

$$\kappa_1^n = \sin(\theta^n)\cos(\varphi^n) \tag{6.5a}$$

$$\kappa_2^n = \sin(\theta^n)\sin(\varphi^n) \tag{6.5b}$$

$$\kappa_3^n = \cos(\theta^n) \tag{6.5c}$$

and

$$\sigma_1^n = \cos(\varphi^n)\cos(\theta^n)\cos(\alpha^n) - \sin(\varphi^n)\sin(\alpha^n) \tag{6.6a}$$

$$\sigma_2^n = \sin(\varphi^n)\cos(\theta^n)\cos(\alpha^n) + \cos(\varphi^n)\sin(\alpha^n) \tag{6.6b}$$

$$\sigma_3^n = -\sin(\theta^n)\cos(\alpha^n) \tag{6.6c}$$

respectively. Since σ_i^n and κ_i^n are orthogonal, σ_i^n lies in a plane normal to the wave number vector κ_i^n.

The amplitude $\hat{u}^n$ of each Fourier mode in Eq. **6.4** is then calculated as,

$$\hat{u}^n = \left(E(\kappa)|\kappa_j^n|\Delta\kappa\right)^{\frac{1}{2}} \tag{6.7}$$

where $\Delta\kappa = \frac{\kappa_{max}-\kappa_1}{N}$. The isotropic turbulence energy spectrum $E(\kappa)$ used in the simulations is,

$$E(\kappa) = u_{rms}'^2 (\frac{\kappa}{\kappa_0})^4 \exp(-2(\frac{\kappa}{\kappa_0})^4) \tag{6.8}$$

where κ_0 is the *peak wave number*. κ_0 is adjusted so as to have 48 peak wavelengths in the stream-wise direction for the simulations. u'_{rms} denotes square of the root mean square of the velocity fluctuations and is given as $u'_{rms} = (ti \cdot \Delta u)$ where ti is the turbulent intensity set to 0.1.

Using the procedure outlined above, the velocity fluctuations are calculated and superimposed on the mean velocity values in order to get the turbulent velocity field at initial.

The obtained initial turbulent velocity fluctuating field is limited to the shear layer by multiplying with a shape function in the form,

$$\exp(-(\frac{y}{2\delta_\theta(0)})^2) \tag{6.9}$$

6.3 Results and Discussions

6.3.1 The quasi-incompressible case

The quasi-incompressible case, where the convective Mach number is equal to 0.3, is chosen as the base case. This case is also used for the code verification, and the comparison as well.

In order to analyze the results, both the Reynolds and the Favre averaging procedures are first introduced, as usual for compressible flows. In Reynolds averaging, a variable ϕ is decomposed as,

$$\phi = \overline{\phi} + \phi^{'} \tag{6.10}$$

where $\overline{\phi}$ denotes the Reynolds averaging and $\phi^{'}$ denotes Reynolds fluctuations. As mentioned above, the Reynolds averaging is obtained over the homogeneous directions (stream-wise and span-wise) in this problem. Similarly, a Favre averaged variable ϕ is written as,

$$\phi = \tilde{\phi} + \phi^{''} \tag{6.11}$$

where $\tilde{\phi}$ denotes the Favre averaging and $\phi^{''}$ denotes Favre fluctuations. $\tilde{\phi}$ is given by,

$$\tilde{\phi} = \frac{\overline{\rho\phi}}{\overline{\rho}} \tag{6.12}$$

The time evolution of the shear layer thickness, i.e., momentum thickness, δ_θ,

$$\delta_\theta = \frac{1}{\rho_0 \Delta u^2} \int_{-\infty}^{+\infty} \overline{\rho}(\frac{1}{2}\Delta u - \tilde{u}_1)(\frac{1}{2}\Delta u + \tilde{u}_1)dy \tag{6.13}$$

is also followed as suggested by Rogers and Moser [61].

As an integral quantity, the momentum thickness is less sensitive to statistical noise than the vorticity thickness,

$$\delta\omega = \frac{\Delta u}{(\frac{\partial \bar{u}}{\partial y})_{max}} \tag{6.14}$$

and evolves in time smoothly. The ratio of vorticity thickness to momentum thickness in the self-similar region shown by the previous DNS studies is approximately 4.5. The growth rate then can be found as a slope of linear curve fit,

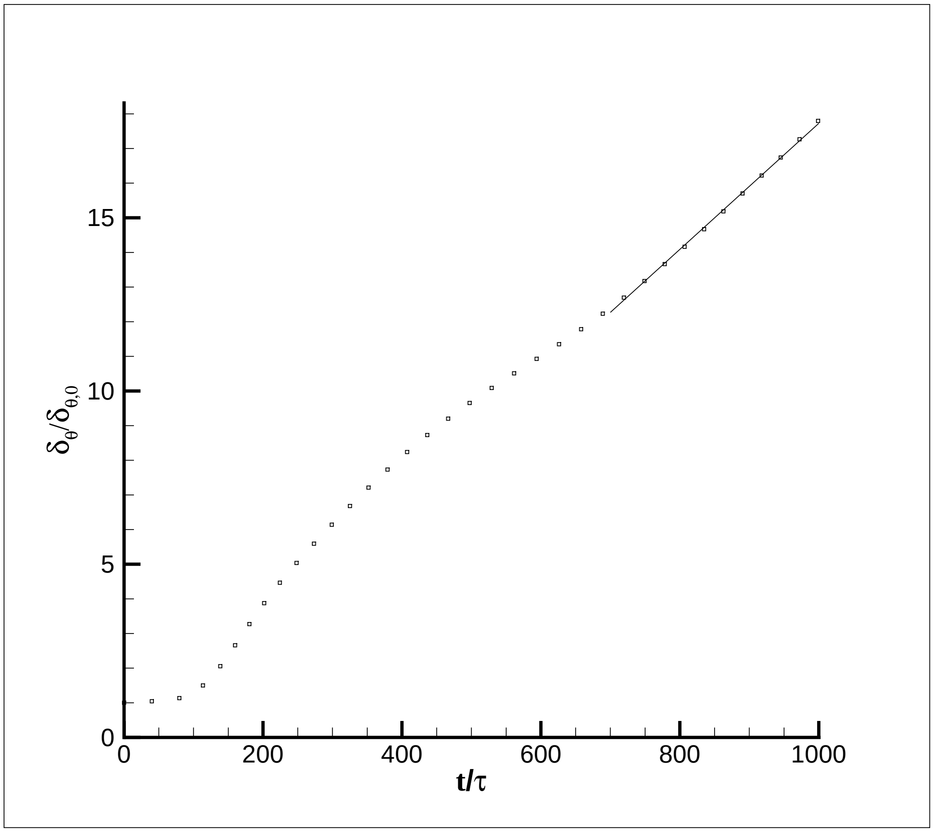


Figure 6.4: Time evolution of momentum thickness for $M_c = 0.3$.

$$\dot{\delta_\theta} = \frac{1}{\Delta u}\frac{d\delta_\theta}{dt} \qquad \textbf{(6.15)}$$

Time evolution of the momentum thickness is given in Fig. 6.4. After an initial settling period, which mainly depends on the initial fluctuations, an approximate linear growth is observed, in consistent with the previous results. The growth rate is found as 0.0182 by taking the slope of the curve in the region where the flow is self-similar. Pantano and Sarkar [71] calculated as 0.0184 for quasi-incompressible case from a DNS database. They also found 0.016 in their DNS study for $M_c = 0.3$. Our value is in good agreement with Pantano and Sarkar [71].

Also, the ratio of the vorticity thickness to momentum thickness, D_ω, is approaching to its nominal value ($\approx 4.5-5.0$) through the end of our simulation.

The Reynolds stress transport equation is introduced as in [71] to calculate the Reynolds stress tensor and turbulent kinetic energy budget,

$$\frac{\partial(\bar{\rho}R_{ij})}{\partial t}+\frac{\partial(\bar{\rho}\tilde{u}_k R_{ij})}{\partial x_k}=\bar{\rho}(P_{ij}-\varepsilon_{ij})-\frac{\partial T_{ijk}}{\partial x_k}+\Pi_{ij} \tag{6.16}$$

where R_{ij} ,P_{ij} ,ε_{ij} ,T_{ijk} ,Π_{ij} are turbulent stresses, production, dissipation, transport and pressure-strain terms, respectively. Their explicit forms are given as follows,

$$R_{ij}=\frac{\overline{\rho u_i'' u_j''}}{\bar{\rho}} \tag{6.17a}$$

$$P_{ij}=-\left(R_{ik}\frac{\partial \tilde{u}_j}{\partial x_k}+R_{jk}\frac{\partial \tilde{u}_i}{\partial x_k}\right) \tag{6.17b}$$

$$\varepsilon_{ij}=\frac{1}{\bar{\rho}}\overline{\left(\tau_{jk}'\frac{\partial u_i''}{\partial x_k}+\tau_{ik}'\frac{\partial u_j''}{\partial x_k}\right)} \tag{6.17c}$$

$$T_{ijk}=\overline{\rho u_i'' u_j'' u_k''}+\overline{p' u_i''}\delta_{jk}++\overline{p' u_j''}\delta_{ik}-(\overline{\tau_{jk}' u_i''}+\overline{\tau_{ik}' u_j''}) \tag{6.17d}$$

$$\Pi_{ij}=\overline{p'\left(\frac{\partial u_i''}{\partial x_j}+\frac{\partial u_j''}{\partial x_i}\right)} \tag{6.17e}$$

Then, the contraction is performed to obtain the values of turbulent kinetic energy budget. The contraction of a tensor is done by setting two of unlike (free) indices equal and summing over the indices left according to the summation convention. Contraction reduces the tensor rank by 2. It can be done repeatedly, depending on the rank of tensor. The contraction once of a tensor with rank 2 simply equals to trace of it and summing up gives a scalar.

Fig.6.5 shows the time evolution of the *rms* of the velocity fluctuations in stream-wise direction. In consistent to the previous studies, it starts from an initial value, reaches a peak and then decreases to its specific value.

The comparison of the Reynolds stress terms are given in figures from Fig.6.6 to Fig.6.9, in comparison with the results from experiments and other DNS studies. An averaging procedure over the time period between 200 and 600 was applied to obtain the curves. The values are in agreement. The stream-wise component of Reynolds stress term is higher than the other components.

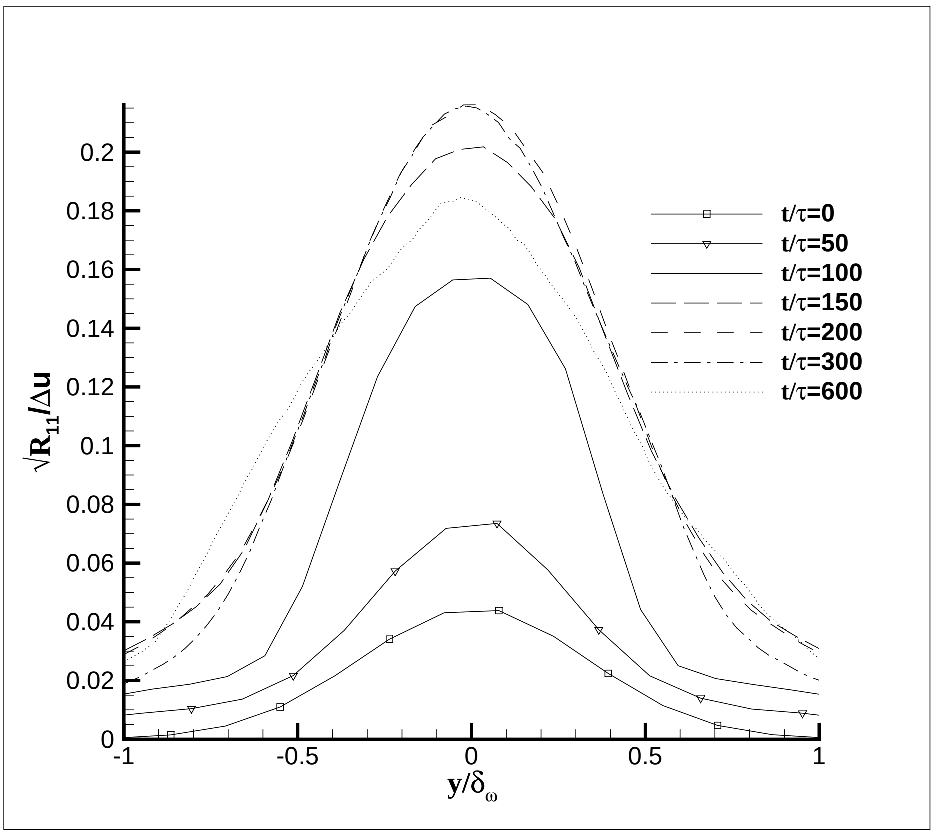


Figure 6.5: Time evolution of the *rms* of the velocity fluctuations in stream-wise direction.

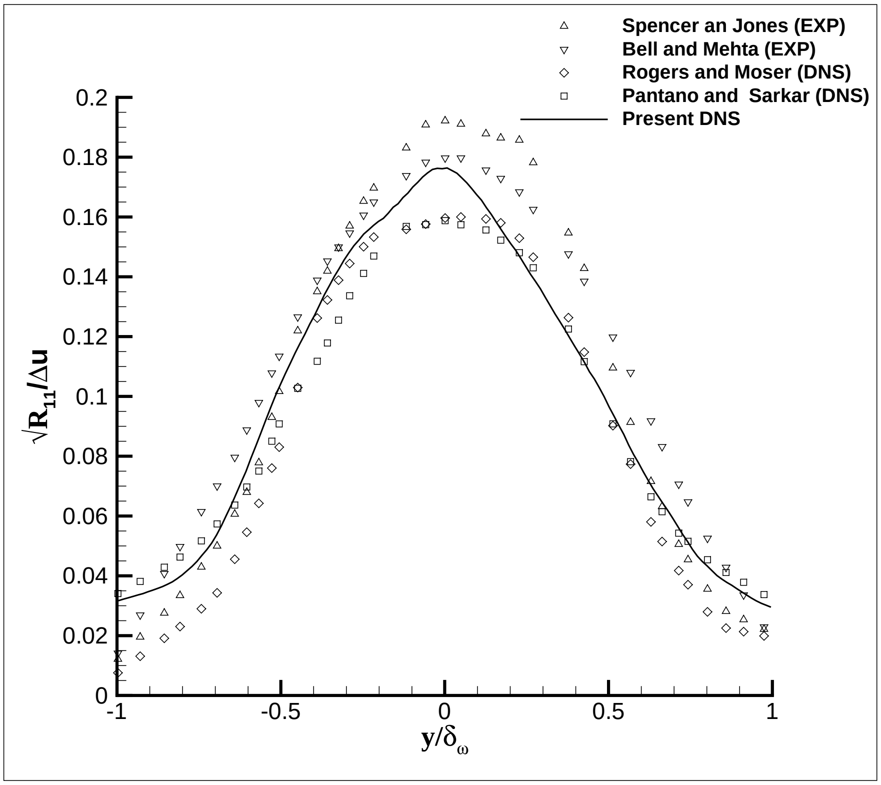


Figure 6.6: Comparison of R_{11} for $M_c = 0.3$.

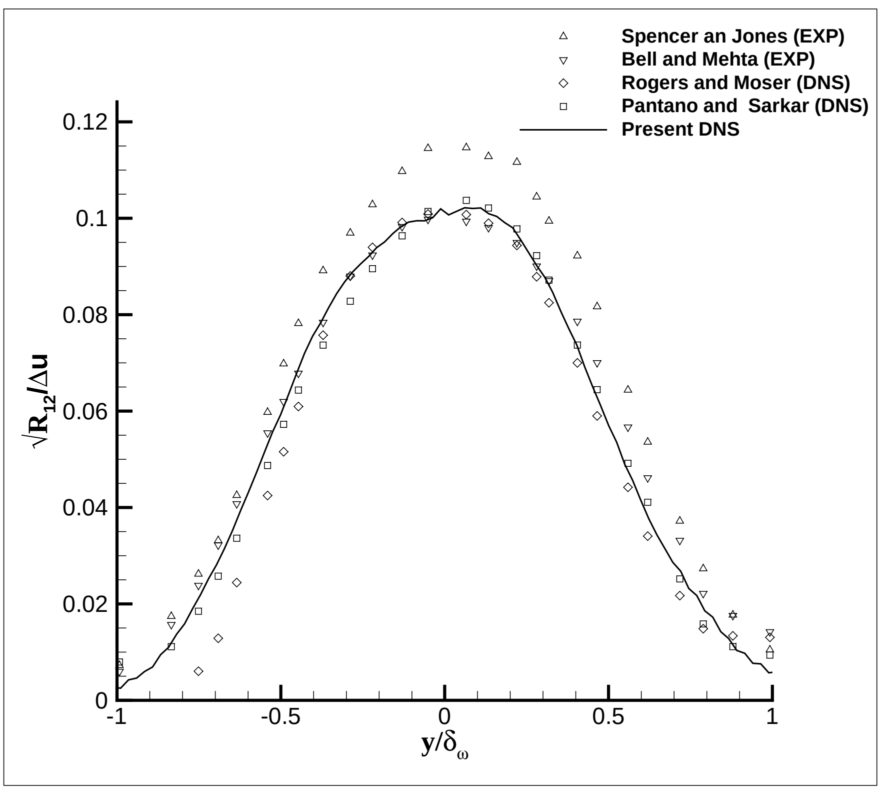


Figure 6.7: Comparison of R_{12} for $M_c = 0.3$.

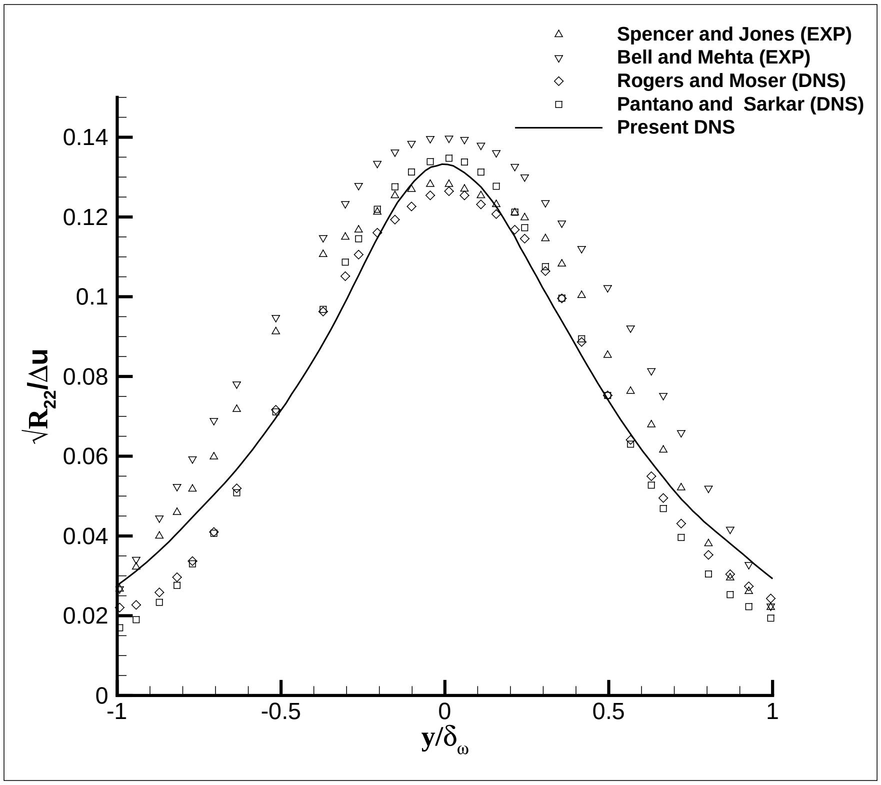


Figure 6.8: Comparison of R_{22} for $M_c = 0.3$.

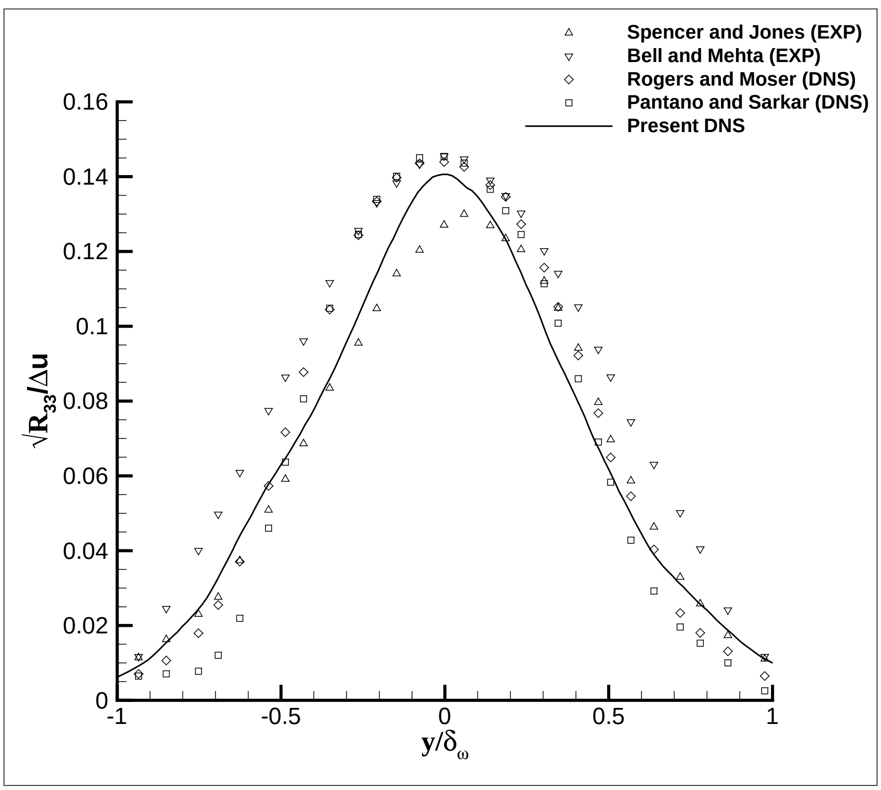


Figure 6.9: Comparison of R_{33} for $M_c = 0.3$.

Table 6.1: Comparison of the peak turbulent intensities of Reynolds stress terms for $M_c = 0.3$.

	$\sqrt{R_{11}}/\Delta u$	$\sqrt{R_{22}}/\Delta u$	$\sqrt{R_{33}}/\Delta u$	$\sqrt{R_{12}}/\Delta u$
Bell and Mehta [57] (EXP, $M_c = 0$)	0.180	0.140	0.146	0.100
Pantano and Sarkar [71] (DNS)	0.155	0.134	0.143	0.103
Hadjadj *et. al.* [75] (LES)	0.170	0.134	0.143	0.106
Foysi and Sarkar [76] (LES)	0.174	0.129	0.143	0.106
Present DNS	0.176	0.133	0.141	0.102

Table 6.1 also compares the peak turbulent intensities of Reynolds stress terms to some prior numerical studies. Our DNS gives similar values to the others.

Turbulent production and dissipation are presented in Fig. 6.10 and in Fig. 6.11. Our values are in well agreement with the other DNS studies.

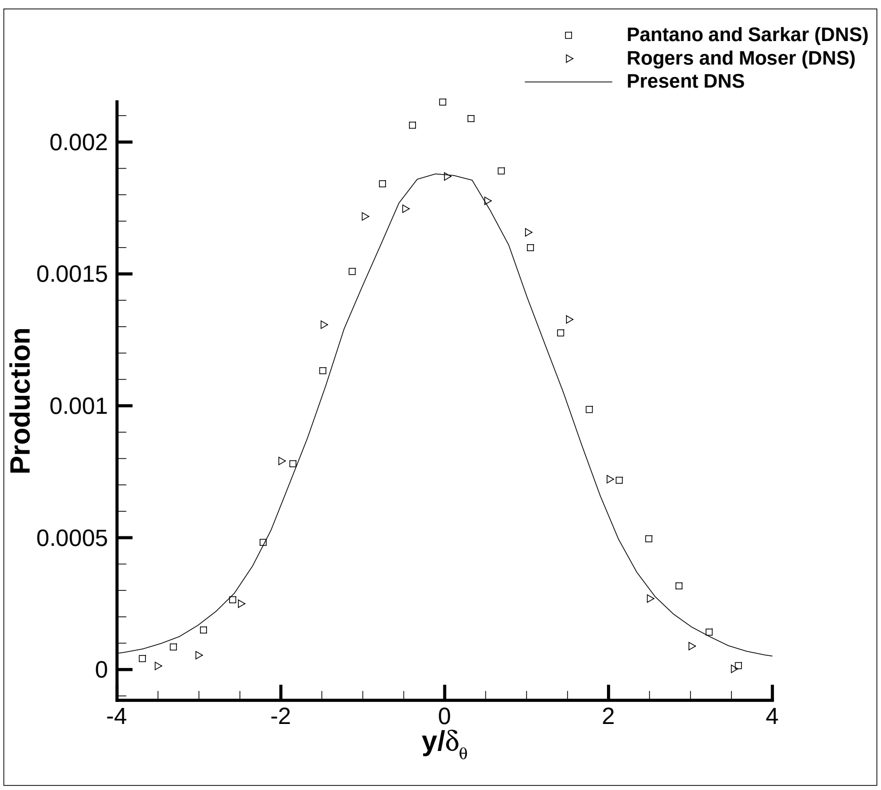


Figure 6.10: Comparison of turbulent production for $M_c = 0.3$.

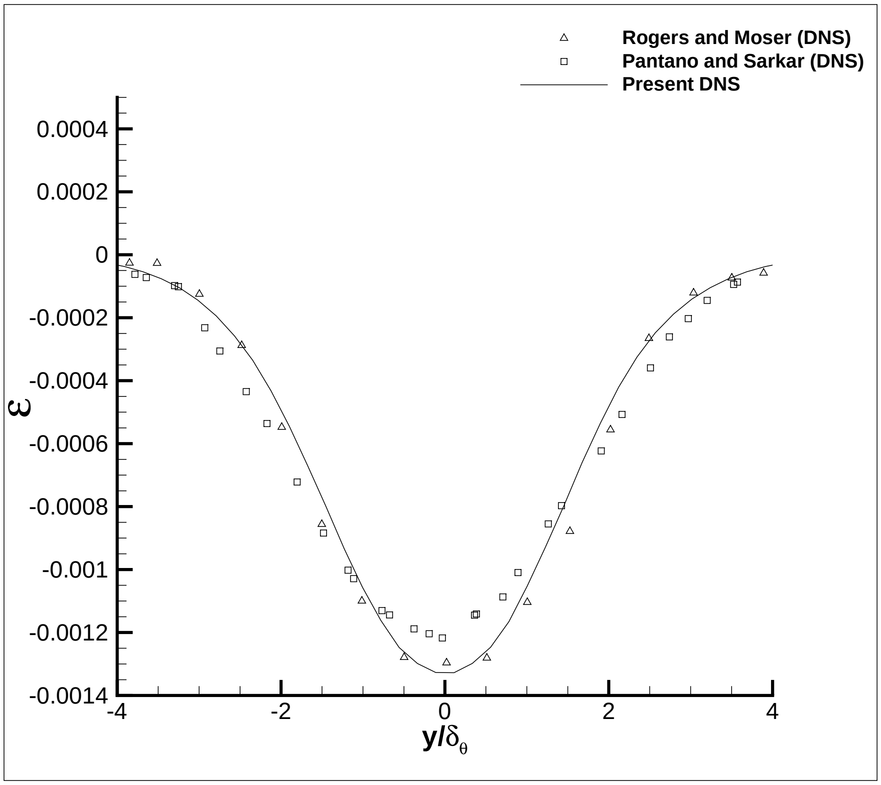


Figure 6.11: Comparison of turbulent dissipation for $M_c = 0.3$.

Fig.6.12 also shows change of the mean velocity in normal-wise direction and compares it to previous experimental and numerical studies. u_{mean} is again obtained via an ensemble averaging in between $t/\tau = 200$ and $t/\tau = 600$. The agreement is well among the results.

The anisotropy of Reynolds stresses show the character and the dependency of the velocity fluctuations to the direction in turbulent flows. It is defined as,

$$b_{ij} = \frac{R_{ij}}{2K} - \frac{1}{3}\delta_{ij} \tag{6.18}$$

and is calculated by integrating the Eq. **6.18** along the shear layer. Time evolution of the anisotropy of the Reynolds stress terms and the comparison of their peak values are also given in Fig. 6.13 and Table 6.2, respectively. In the self-similar region (i.e., after a sufficiently long time), the values of the diagonal components of b_{ij} reach a constant value. The peak values we have found are also very close to Pantano and Sarkar [71]. Similarly, the anisotropy in stream-wise is stronger than the other directions.

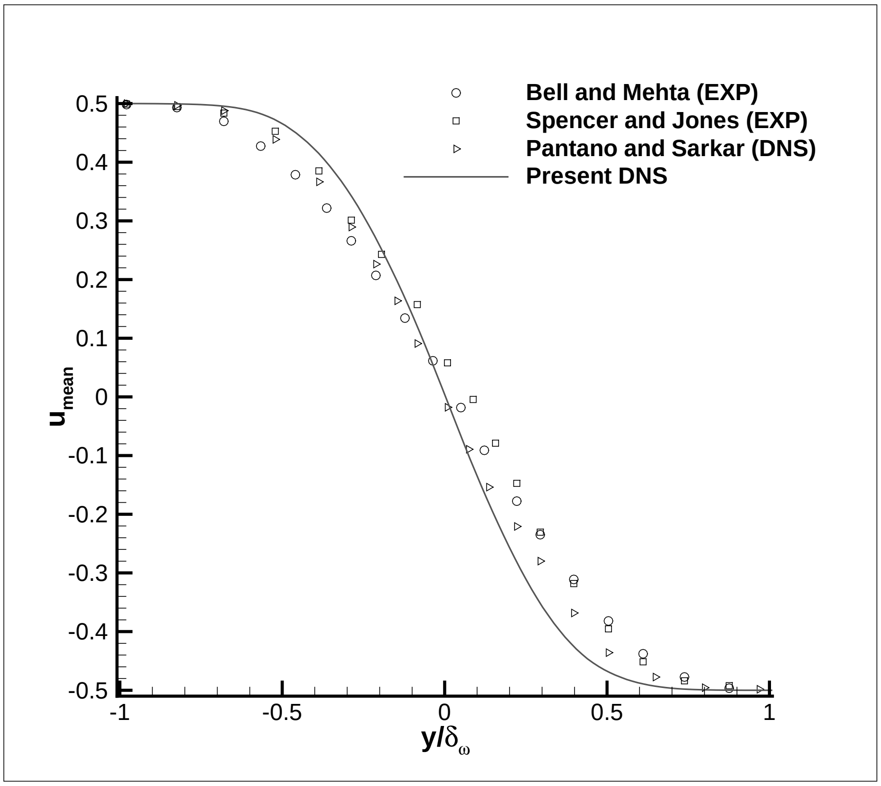


Figure 6.12: Comparison of u_{mean} for $M_c = 0.3$.

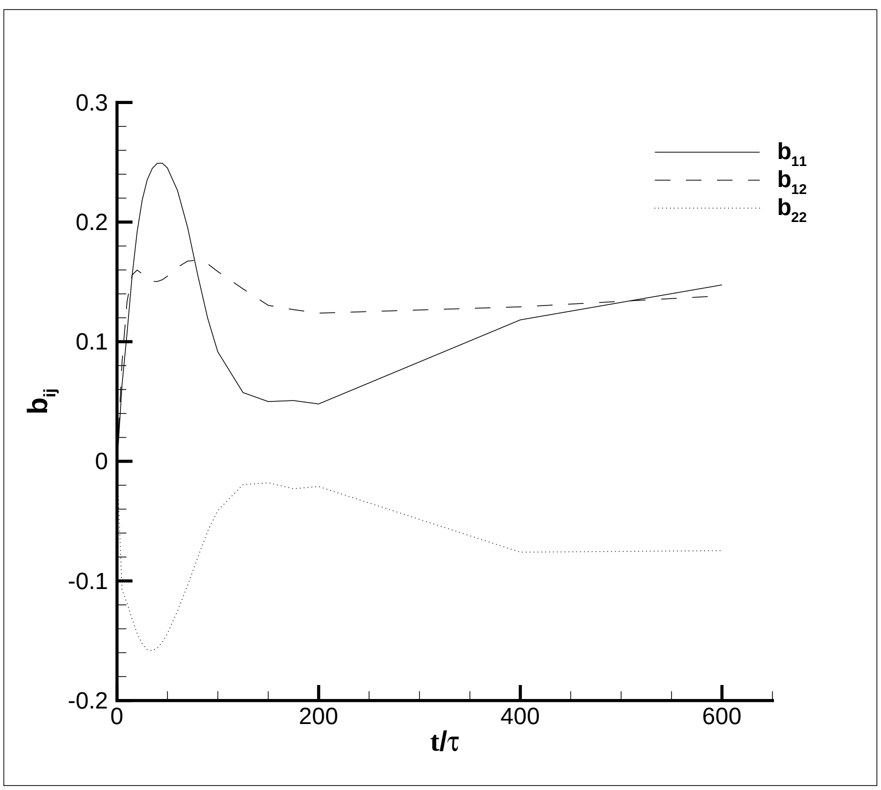


Figure 6.13: Time evolution of the anisotropy of Reynolds stress terms for $M_c = 0.3$.

Table 6.2: Comparison of the peak Reynolds stress anisotropy values for $M_c = 0.3$.

	b_{11}	b_{22}	b_{12}
Pantano and Sarkar [71] (DNS)	0.26	-0.16	0.19
Present DNS	0.25	-0.16	0.17

Fig.6.14 shows the Q- iso-surfaces at $t/\tau = 1000$ where the flow is self-similar for $M_c = 0.3$. Q is set to $0.02Q_{max}$. Positive Q values show the low pressure high speed vortex dominated regions in the flow. Lots of small and localized vortical structures, which can regarded as turbulent spots, are observed and the flow evolve into a fully turbulent regime in time.

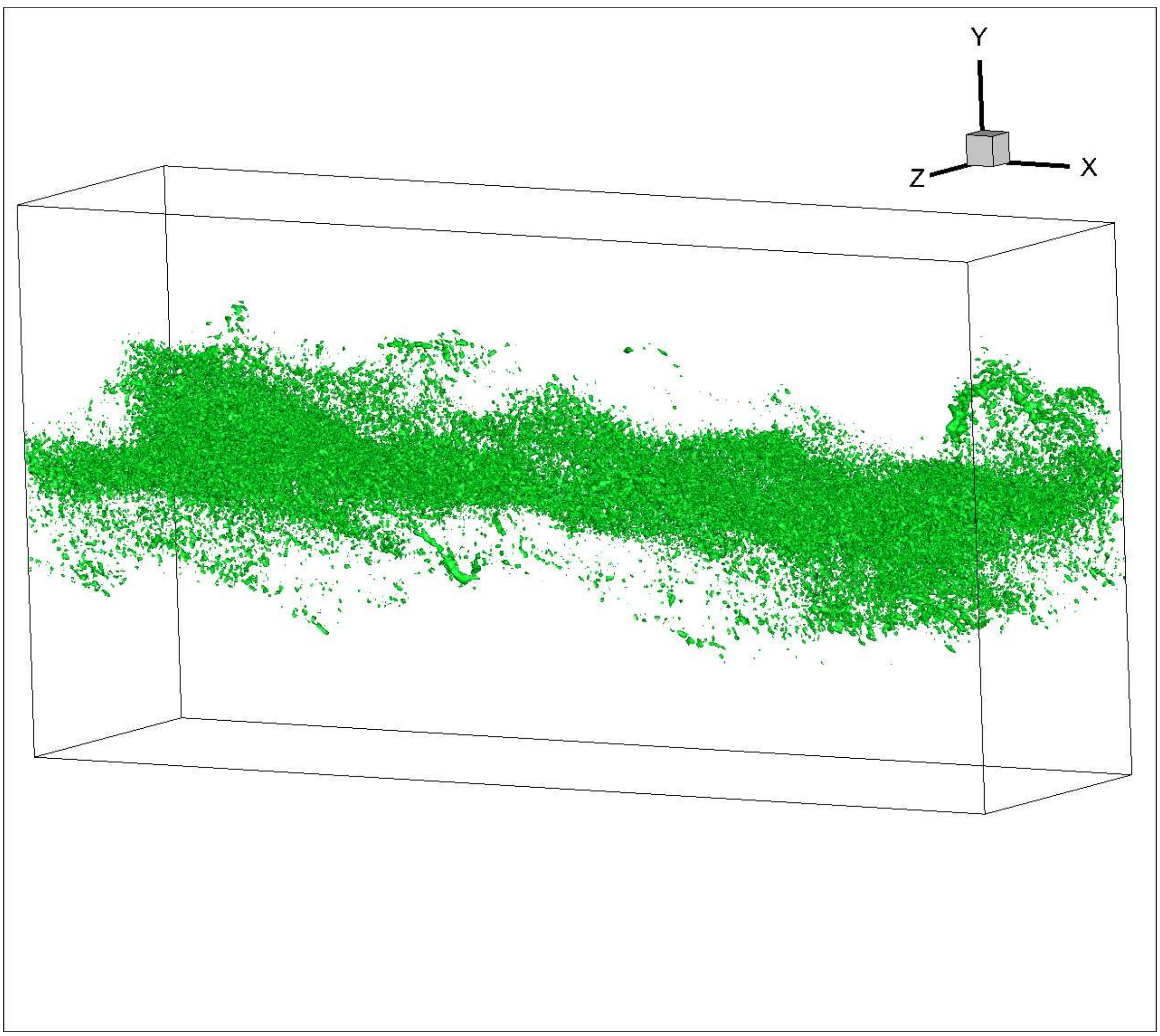


Figure 6.14: Q- iso-surfaces at $t/\tau = 1000$ for $M_c = 0.3$.

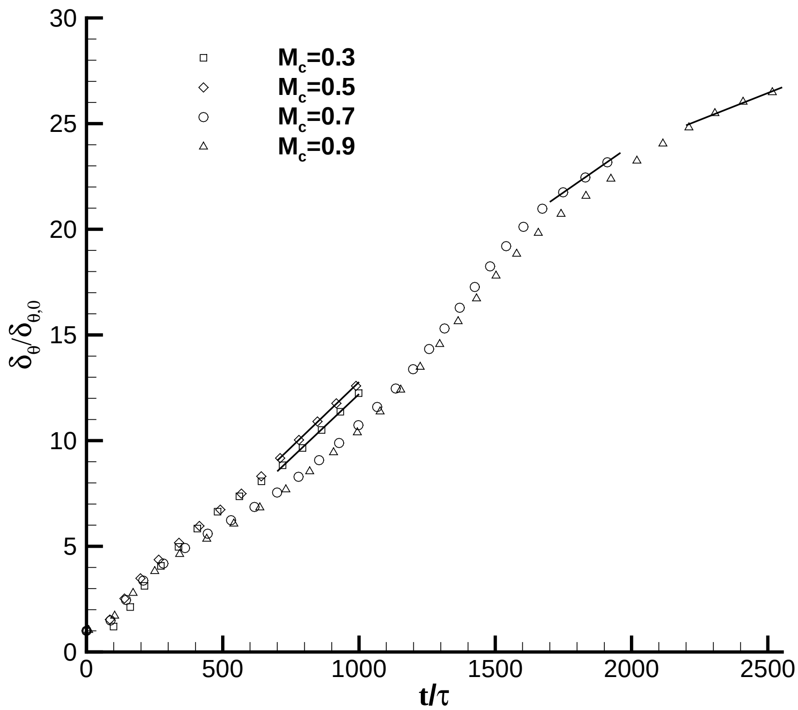


Figure 6.15: Comparison of the momentum thicknesses and the growth rates for different M_c's.

6.3.2 The compressible case

Having demonstrated via the base quasi-incompressible results that the algorithm and the solver developed based on it produce the correct physical mechanisms behind TSL instability. In the following cases, behavior of the algorithm in the presence of compressibility effects is investigated. For this purpose, M_c's changing from 0.3 to 0.9 are studied. The results are presented and are compared with the previous experimental and numerical studies. Compressibility effects are also analyzed.

Fig.6.15 compares the time evolution of momentum thicknesses for different convective Mach numbers. A linear fit for each curve is plotted in the self-similar region for calculation and comparison of the growth rates. It is observed that time required to obtain to a self-similar state increases with increasing M_c.

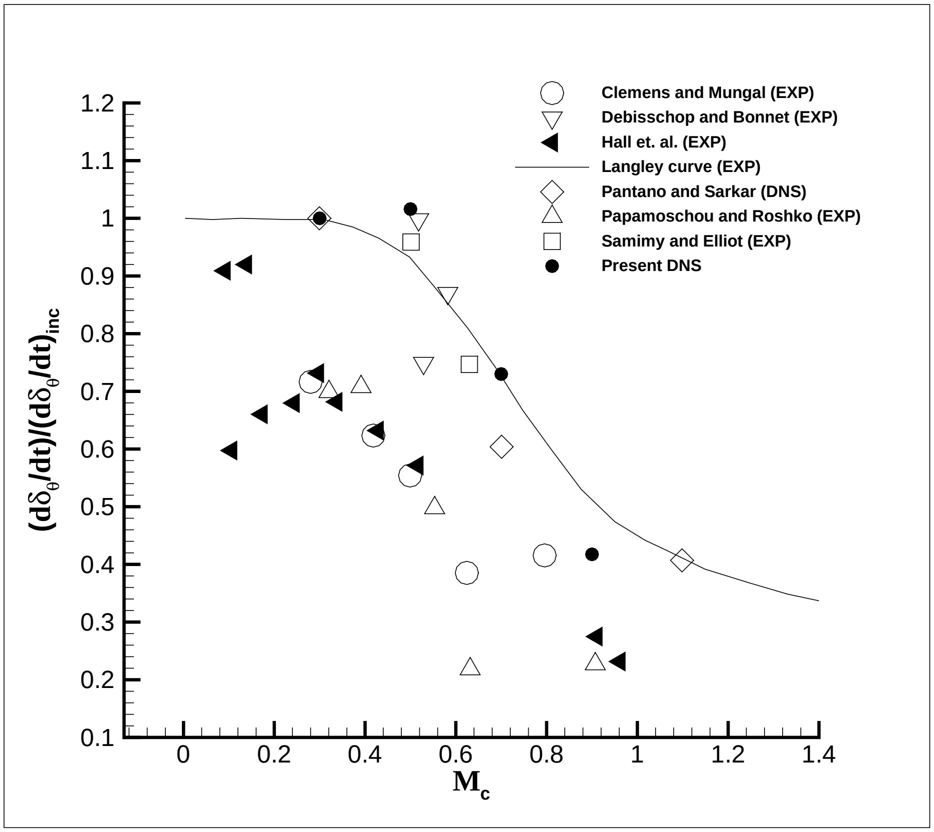


Figure 6.16: Change of the growth rates with different M_c's.

The growth rate values then are normalized by the base incompressible growth rate value and are plotted against the M_c in Fig.6.16. The filled circles represent our results. As it is clearly seen from this figure that compressibility reduces to growth rate of the instability, agreeing well with the many previous results and the widely accepted Langley experimental curve represented by solid line.

Compressibility effects on the Reynolds stress terms are presented in the figures from Fig.6.17 to Fig.6.20. Figures from Fig.6.21 to Fig.6.22 also compare the peak values of the stream-wise and the normal-wise Reynolds stress terms respectively to the previous studies. Consistently to the previous studies ([60, 61, 67, 71, 76]), the results show a reduction in the turbulent velocity fluctuations accompanying the decrease in the growth rate.

Fig. 6.23 gives the change in the turbulent production and compares our results to Pantano and Sarkar [71]. Present DNS results are represented by different pattern of

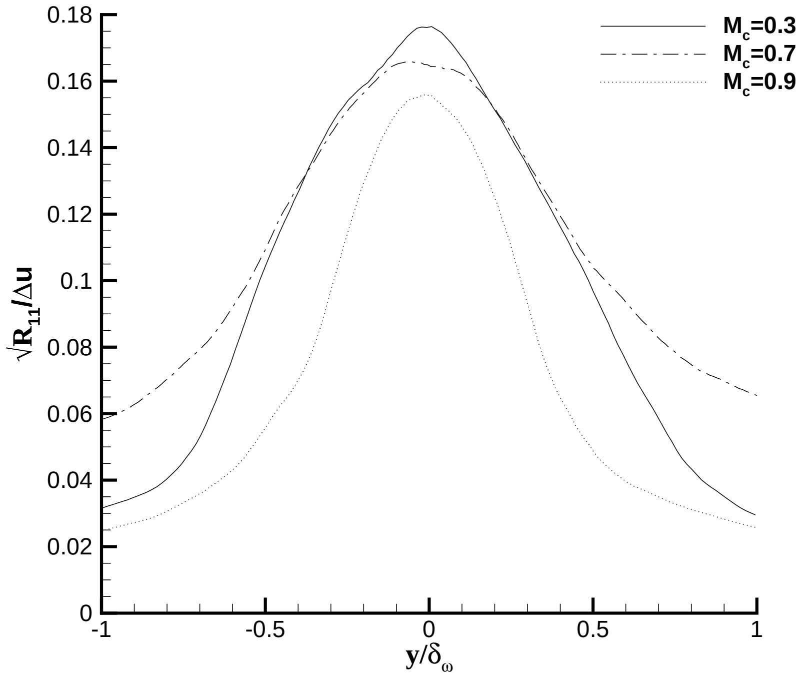


Figure 6.17: Change of R_{11} with different M_c's.

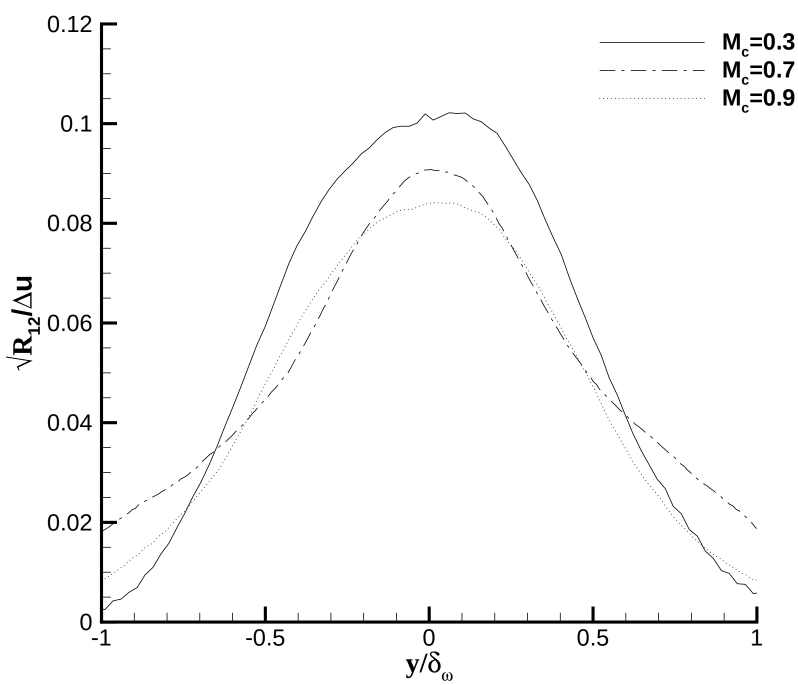


Figure 6.18: Change of R_{12} with different M_c's.

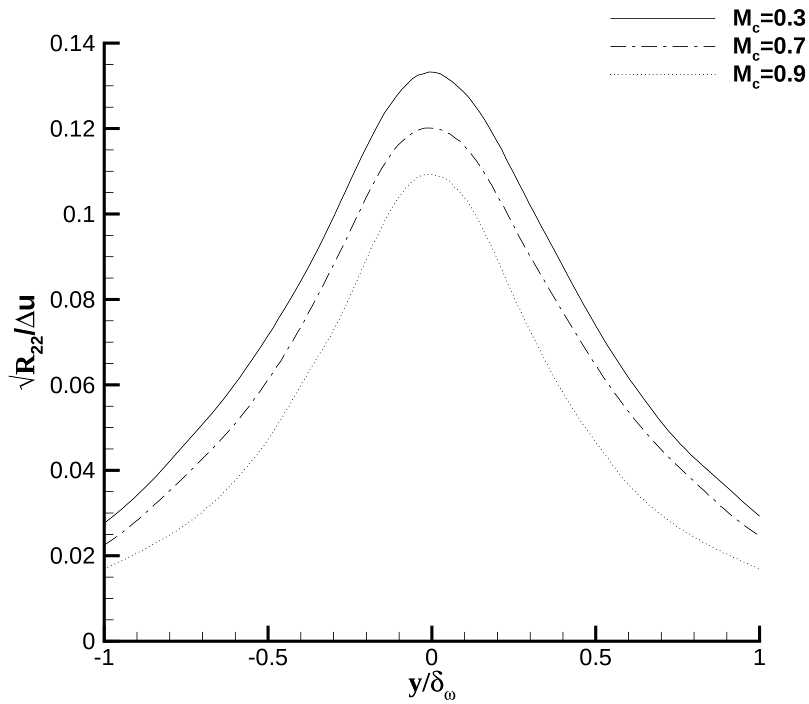


Figure 6.19: Change of R_{22} with different M_c's.

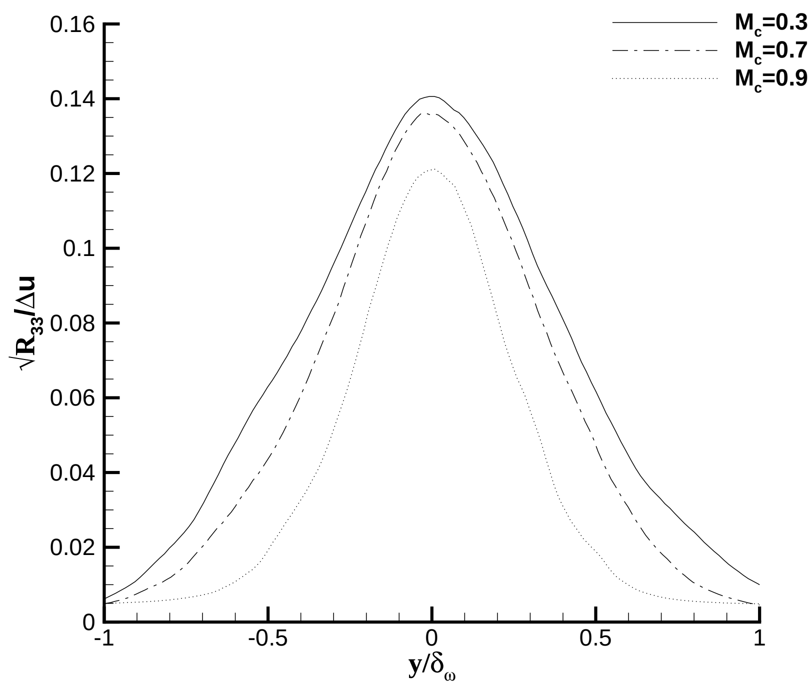


Figure 6.20: Change of R_{33} with different M_c's.

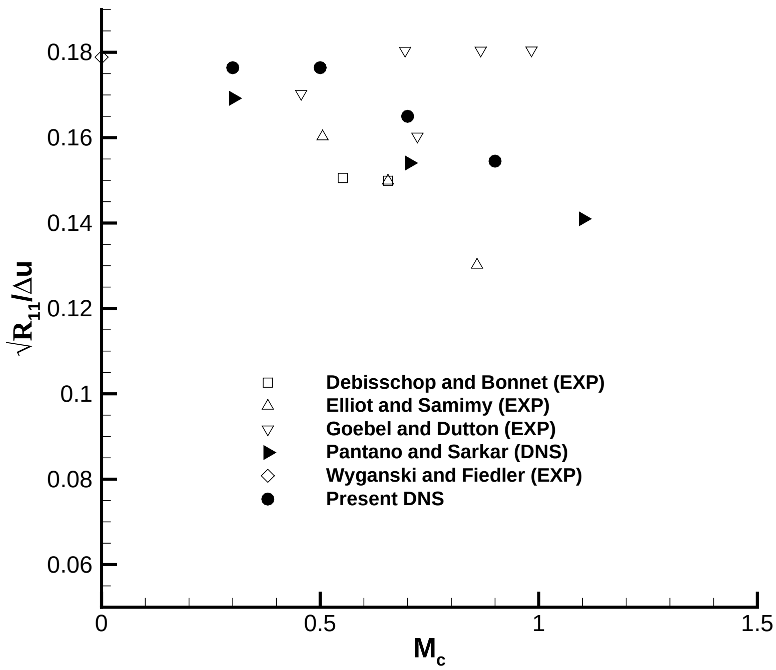


Figure 6.21: Comparison of the peak values for R_{11}.

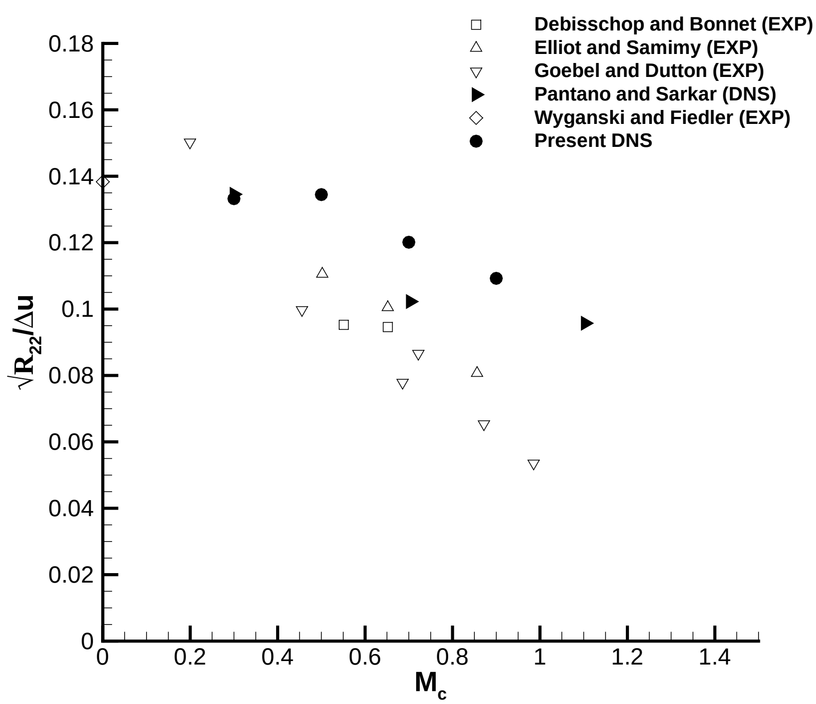


Figure 6.22: Comparison of the peak values for R_{22}.

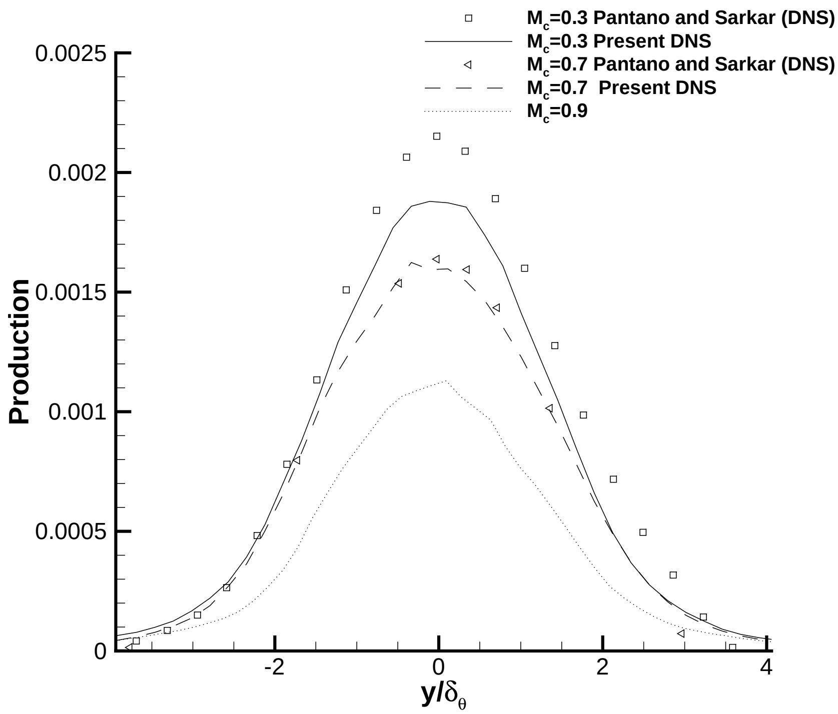


Figure 6.23: Comparison of the turbulent production term.

lines. Production reduces with the increasing M_c. It is the source for the decrease in the growth rate, confirming the other DNS studies [61, 71, 76].

Similar behavior is seen for the turbulent dissipation given in Fig. 6.24. However, it is less affected by the compressibility than the other terms, as previously noted in [71,76].

The Reynolds stress anisotropies are again calculated and their change with increasing M_c are plotted in Fig. 6.25. A slight increase is observed in the peak values at early times. Diagonal components are more effected than off-diagonal ones. Overall effect of increasing M_c on b_{ij}'s are not so severe, consistently to previous numerical studies [71, 76].

Table 6.3 compares the peak values of the Reynolds stress anisotropies with Pantano and Sarkar [71] for $M_c = 0.7$.

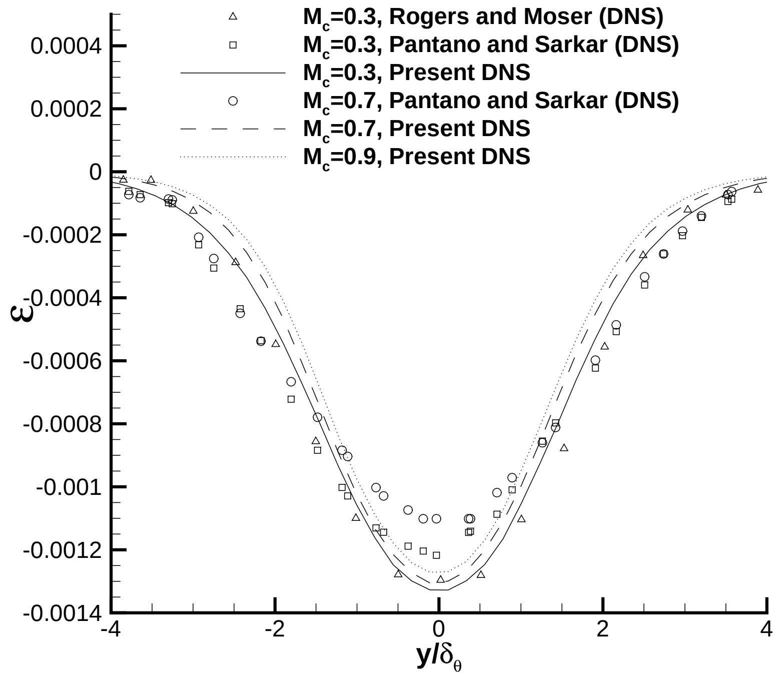


Figure 6.24: Comparison of the turbulent dissipation term.

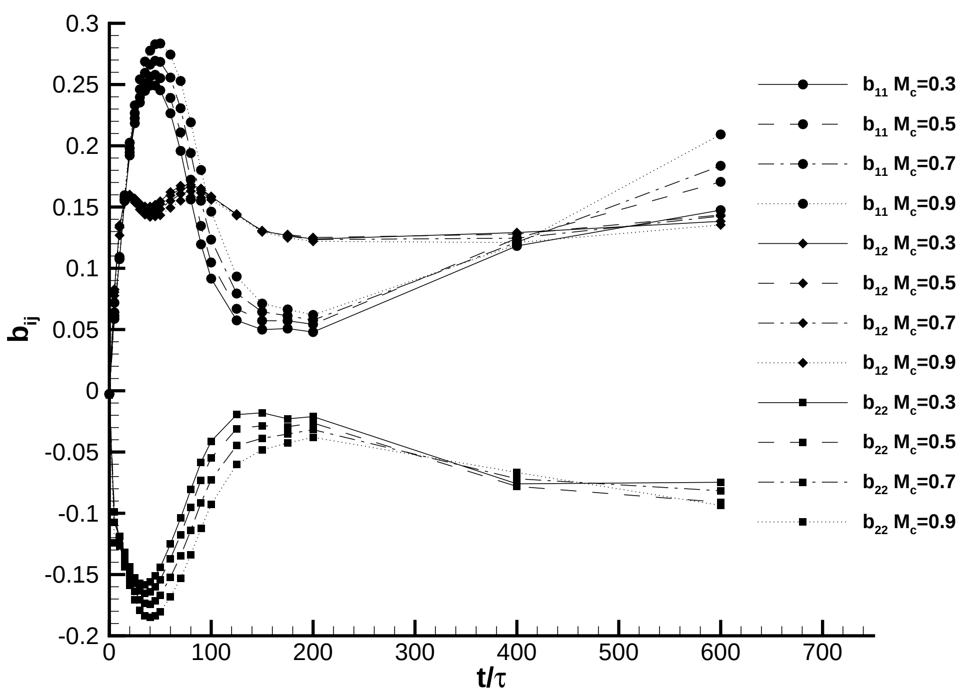


Figure 6.25: Comparison of the Reynolds stress anisotropy terms.

Table 6.3: Comparison of the peak Reynolds stress anisotropy values for $M_c = 0.7$.

	b_{11}	b_{22}	b_{12}
Pantano and Sarkar [71] (DNS)	0.29	-0.19	0.19
Present DNS	0.27	-0.175	0.17

We have applied our in-house solver, *i*DNS, to turbulent shear layer instability problem. The results show that the algorithm, as well as the solver we developed, are capable of producing the correct physics of the evolution of system with a perturbation into a nonlinear hydrodynamic regime.

7. DNS OF RAYLEIGH-TAYLOR INSTABILITY

In this chapter, Direct Numerical Simulation of Rayleigh-Taylor Instability is performed using *i*DNS. Incompressible and compressible cases are considered. After giving an introduction, initial and boundary conditions are explained in the following sections. The simulation results are presented and are compared to both experimental and numerical studies found in the literature. Compressibility effects on the development of the instability and mixing are also analyzed.

7.1 Introduction

Rayleigh-Taylor Instability occurs when a heavy fluid of density, ρ_h, on top is supported against the gravity, g, by a light fluid of density, ρ_l, on bottom [77–79]. Fluid is initially in hydrostatic equilibrium. Pressure gradient is balanced by the gravity acting only in the vertical direction. In this case pressure and density gradients are in opposite and infinitesimal perturbations grow at the interface exponentially. Then, the baroclinic vorticity causes the fluids to interpenetrate as bubbles and spikes into each other and mix.

It is observed in many physical and engineering flows such as type Ia supernovae, inertial confinement fusion, cavitation bubbles and oceanic and atmospheric flows. Rayleigh-Taylor Instability also serves as a very challenging test case for algorithms, since it includes many aspects of flow modeling like body force treatment, turbulent mixing, diffusion and interface capturing.

The linear theory [79, 80], as well as the experiments and the previous numerical studies, show that after an initial settling period, instability grows exponentially in time t,

$$\exp(\gamma t) \qquad \textbf{(7.1)}$$

where

$$\gamma = \sqrt{Ag\kappa} \tag{7.2}$$

is the growth rate and A is the Atwood number given as

$$A = \frac{\rho_h - \rho_l}{\rho_h + \rho_l} \tag{7.3}$$

and κ is the wave-number. Then, flow evolves into a nonlinear mixing regime which can be regarded as the onset of turbulence. The penetration of the light fluid into the heavy one as bubbles or the penetration of the heavy fluid into the light one as spikes can be modelled using [81, 82],

$$h_b = \alpha_b A g t^2 \tag{7.4}$$

and

$$h_s = -\alpha_s A g t^2 \tag{7.5}$$

respectively. The mixing zone is defined as,

$$h = h_b(t) - h_s(t) = \alpha A g t^2 \tag{7.6}$$

and the corresponding growth rate is found,

$$\dot{h} = 2\alpha A g t \tag{7.7}$$

where $\alpha = \alpha_b + \alpha_s$ is the total growth rate factor given as $0.01 \leq \alpha \leq 0.07$ in the literature [83].

The compressibility effects on the development of Rayleigh-Taylor Instability have also been studied by many authors [84–87]. There was a contradiction among the results. Some of them found a stabilizing effect, while some others found that compressibility increases the growth of mixing. Recently, Livescu [88] resolved this contradiction and showed that compressibility can be characterized by two parameters, the ratio of specific heats and the speed of sound, which they have opposite effects on the growth.

Our aim here is, once, to demonstrate the algorithm's ability to evolve perturbations into turbulent mixing for the base incompressible flow with the Atwood number $1/3$, then to analyze the effects of compressibility on the growth rate of the instability by increasing the Mach number.

7.2 Numerical Setup of the Flow

7.2.1 Initial and boundary conditions

The flow is initialized with a density ratio 2 which gives Atwood number $1/3$. The lower fluid density is set to 1. The constant gravitational acceleration is in the normal-wise direction. Fluids are initially at rest.

Domain height, L_y, is taken as the reference length scale, l_0. The reference velocity, u_0, is the free-fall velocity given as,

$$u_0 = \sqrt{g l_0} \tag{7.8}$$

The characteristic time scale is then,

$$\tau = \sqrt{\frac{l_0}{Ag}} \tag{7.9}$$

The higher fluid density is taken as reference density ρ_0. The Mach number, $M = u_0/\sqrt{\gamma p_0/\rho_0}$, is set to 0.1 for the incompressible case and 1.1 for the compressible case. Since the mean velocities are initially zero and the ratios of the specific heats are the same and constant, the Mach number here is defined as the ratio of the characteristic velocity (i.e., free-fall velocity) to the speed of sound. The compressibility effects are obtained by changing the thermodynamic state of the system. The reference pressure, p_0, is calculated using M, ρ_0 and u_0. The pressure is initially in hydrostatic equilibrium,

$$p = p_0 - \rho g y \tag{7.10}$$

The kinematic viscosity is calculated using the grid-dependent formula given in the paper by Alpha group collaboration [89], that is,

$$\nu = \bar{\omega}\sqrt{Ag\Delta^3} \tag{7.11}$$

where $\bar{\omega}$ is the coefficient taken as 0.21 and Δ is the grid spacing. Then, dynamic viscosities for each fluid are obtained via multiplying ν by their densities. The dynamic viscosity of higher density fluid, μ_h, is chosen as the reference to use it in calculating the scaling Reynolds number.

Using the reference values, the value of the working (scaling) Reynolds number,

$$Re = \frac{\rho_0 u_0 l_0}{\mu_0} \tag{7.12}$$

is $\sim 10^5$. It is almost three-order of magnitude higher than the one used in the DNS of Cook and Dimotakis [90]. The difference is due to the choice of the scaling parameters (i.e., the reference values). These choices were made in order to perform compressible simulations with the current algorithm. Otherwise, the incompressible (low-Mach number) scaling can cause unphysical non-dimensional temperature and dynamic viscosity fields at initial or in updating, during the compressible simulations.

The Eckert number, Ec, which is the ratio of kinetic energy to enthalpy arises from the non-dimensionalization of the energy equation. It is approximated using Eq. **2.21**. The Froude number, Fr, which is the ratio of the reference velocity to the free-fall velocity is also equal to unity. The constant gravitational acceleration is also scaled by itself.

Boundary conditions are periodic in stream-wise and span-wise direction. Inviscid slip-wall boundary conditions are applied at top and bottom boundaries, similar to the Turbulent Shear Layer Instability in the previous chapter.

Flow is followed up to when the bubbles and the spikes reach to the end of the domain in each simulation with a time step, $dt = 1.5 \times 10^{-5}$.

7.2.2 Grid details

The length of the grid used here is $2\pi x 4\pi x 2\pi$ with $256x512x256$ cells in stream-wise, normal-wise and span-wise directions, respectively. The cell size is uniform. The interface between fluids is located at $y = 0$.

7.2.3 Initial perturbations

Usually, the interface is perturbed to trigger the instability. However, this requires an additional smoothing, if amplitude of the perturbation is smaller than the grid resolution. In addition, a small initial perturbations may be necessary to satisfy the divergence-free velocity field depending on the equations to be solved and the method.

Jun *et.al.* [91] and Stone and Gardiner [92] showed that perturbing the velocity field directly is superior to perturbing interface position, since no smoothing is required and resolution-independent perturbations are obtained which yield identical results.

We followed the latter way to initialize the instability in our simulations using the perturbations given in the following form,

$$v^{'} = A_0 Rand(1 + \cos(2\pi y/L_y)) \tag{7.13}$$

with the amplitude A_0 is 0.005 and *Rand* is the random number between -1 and $+1$. This perturbation is only added initially to the vertical velocity component.

7.2.4 Computational requirements

256 cores were used for runs. Each core has approximately 2 GB of RAM. The total wall-clock time for each computation is about 500 hours.

7.3 Results and Discussions

7.3.1 The incompressible case

The first case considered here is the incompressible case where the Mach number is 0.1. In order to measure the mixing, the local mole fraction or mixture fraction based on the density of the heavy fluid is defined as,

$$\chi(x,y,z,t) = \frac{\rho(x,y,z,t) - \rho_l}{\rho_h - \rho_l} \tag{7.14}$$

Fig 7.1 shows the time evolution of the local mole fraction. As it is seen from the figure, the bubbles and the spikes are formed in time. While the bubbles are penetrating into the heavy fluid, the spikes go deep into the light fluid. Initially smaller structures merge in time and evolve into larger ones. The main responsible mechanism at early stages of this evolution observed here is the diffusion.

The local mole fraction field is averaged over the homogeneous directions to give the mixing zone growth,

$$\langle \chi(x,y,z,t) \rangle_{xz} = \frac{1}{N_x N_z} \sum_N \chi(x,y,z,t) \tag{7.15}$$

where $< \cdot >_{xz}$ denotes the averaging. Fig.7.2 shows the evolution of averaged local mole fraction field in time. Its step-like shape deforms in time due to the growth of the mixing zone and the merging of the small-scale mushroom-like structures, similar to Cook and Dimotakis [90].

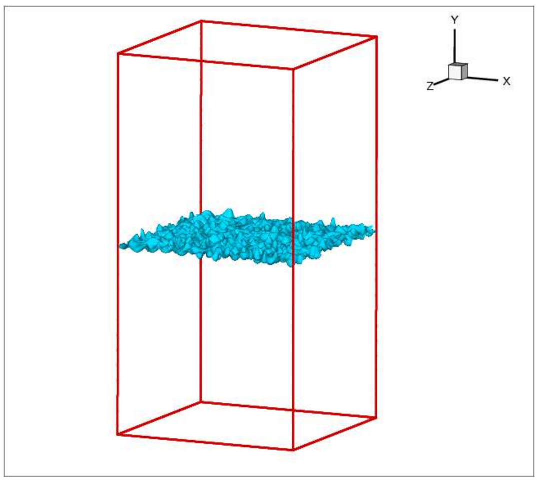
The penetration lengths for bubbles $h_b(t)$ and spikes $h_s(t)$ are defined as the distances where,

$$\langle \chi(x,y,z,t) \rangle_{xz} \leq 1 - \varepsilon \tag{7.16}$$

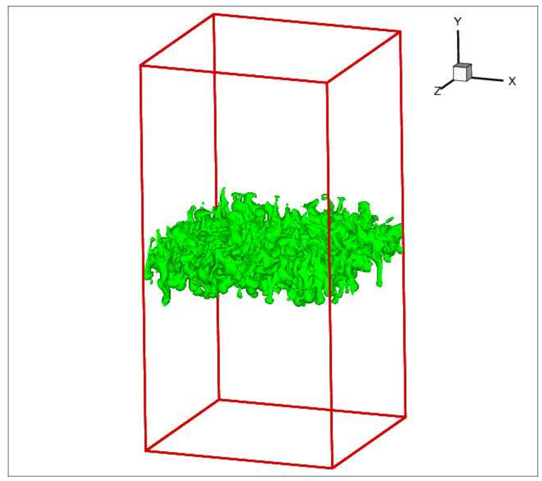
and

$$\langle \chi(x,y,z,t) \rangle_{xz} \geq \varepsilon \tag{7.17}$$

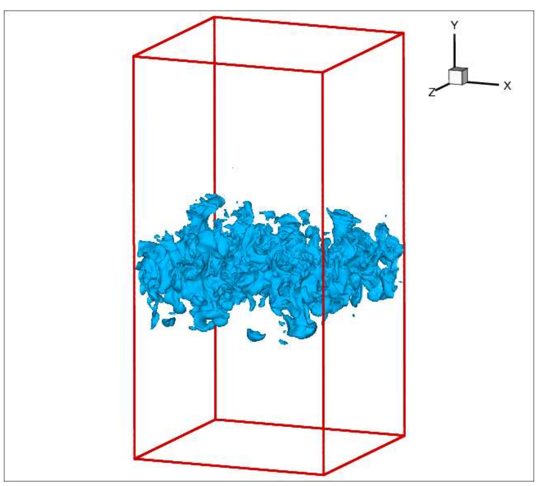
respectively. ε is traditionally set to 0.01.



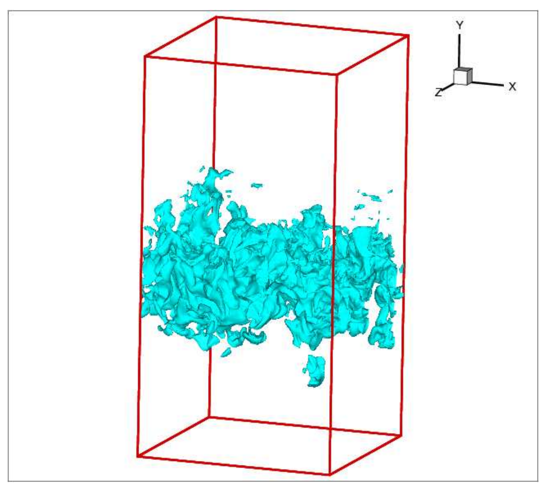
(a) $t/\tau = 2$



(b) $t/\tau = 3$



(c) $t/\tau = 4$



(d) $t/\tau = 5$

Figure 7.1: The time evolution of the local mole fraction field in terms of iso-surfaces (χ=0.5). Since it is stored into separate files, colors are different.

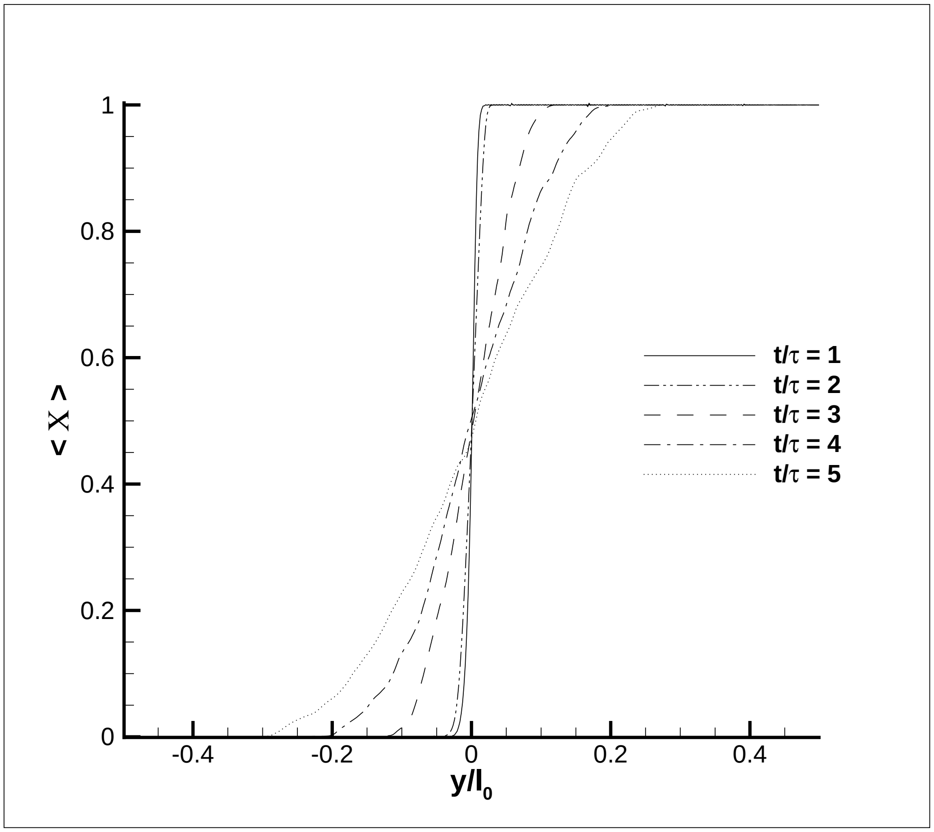


Figure 7.2: Evolution of the averaged local mole fraction field in time.

Once $h_b(t)$ and $h_s(t)$ are found, the growth rate coefficients for the bubbles, α_b, and the spikes, α_s, the total extend of mixing zone, $h(t)$, and its derivative, $\dot{h}(t)$, can be calculated using the equations from Eq.**7.4** to Eq.**7.7**.

Fig.7.3 shows the development of bubble and spike penetration lengths in time. The evolutions are not symmetric. The penetration of spikes are faster than bubbles. The ratio of h_s/h_b is around ≈ 1.2 at the end of the simulation with an aim to increase. The value is consistent with the experimental value 1.3 reported by Read [93] for the given Atwood number, ≈ 1.25 by LES of Burton [94], ≈ 1.3 by DNS of Cook and Dimotakis [90] using higher aspect ratio domains and ≈ 1.2 DNS by Alpha group [89].

As seen from the total penetration length in Fig.7.4, there are two main stages. The growth in early stage development is mainly diffusive. The late time non-linear development is faster than the diffusive growth. This is in good agreement with Cook and Dimotakis [90].

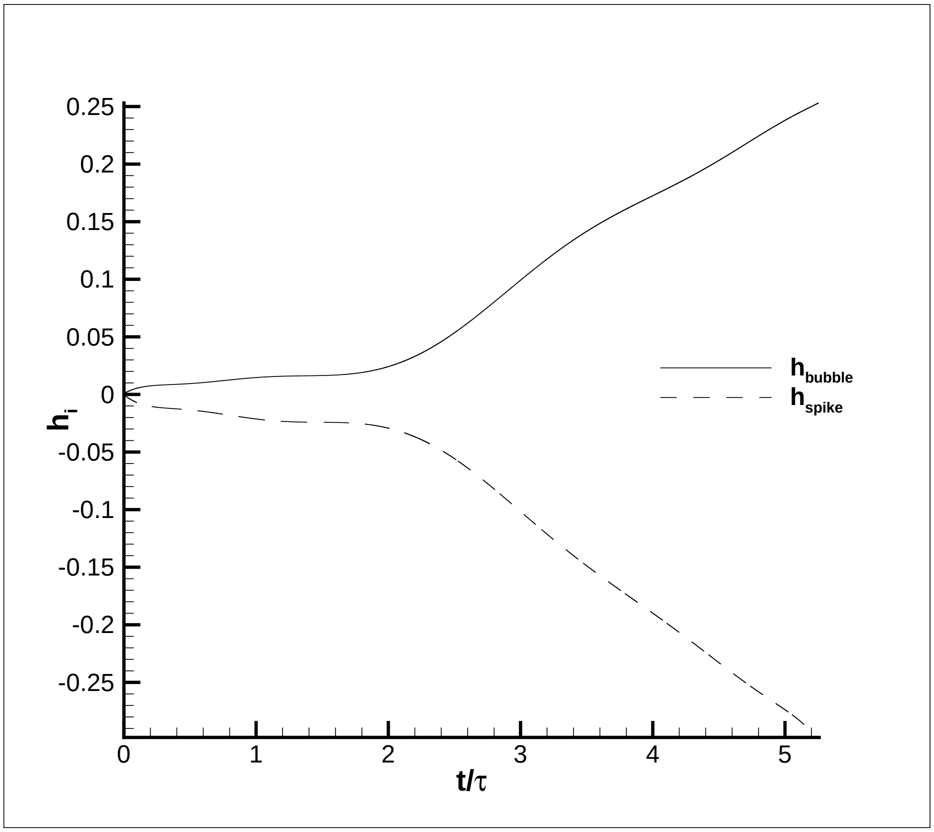


Figure 7.3: Time evolution of the penetration lengths for bubble and spike.

The bubble and spike growth rate coefficients are plotted in Fig.7.5. Following the exponential growth, the both coefficients reach a constant value. The late time value of α_{bubble} is ~ 0.03. Cook and Dimotakis [90] reported ~ 0.03 in their DNS study. Alpha group also found 0.025 ∓ 0.003 in their numerical studies.

As seen in Fig.7.6, α is around 0.07. α values reported by Cook and Dimotakis [90] and Burton [94] are $0.07-0.08$ and $0.06-0.07$, respectively. Dimonte and Schneider [83] give a range, $0.01 \le \alpha \le 0.07$, for α.

In order to quantify the amount of mixed fluid in the mixing layer, a mixing fluid mole fraction (i.e., product thickness) is also defined as,

$$X_p(\chi) = \begin{cases} \frac{\chi}{X_s} & \chi \le X_s \\ \\ \frac{1-\chi}{1-X_s} & \chi > X_s \end{cases} \tag{7.18}$$

where X_s is the heavy fluid mole fraction for the stoichiometric mixture which is set to 0.5. The word "stoichiometric" defines the optimum amount or ratio that the reaction

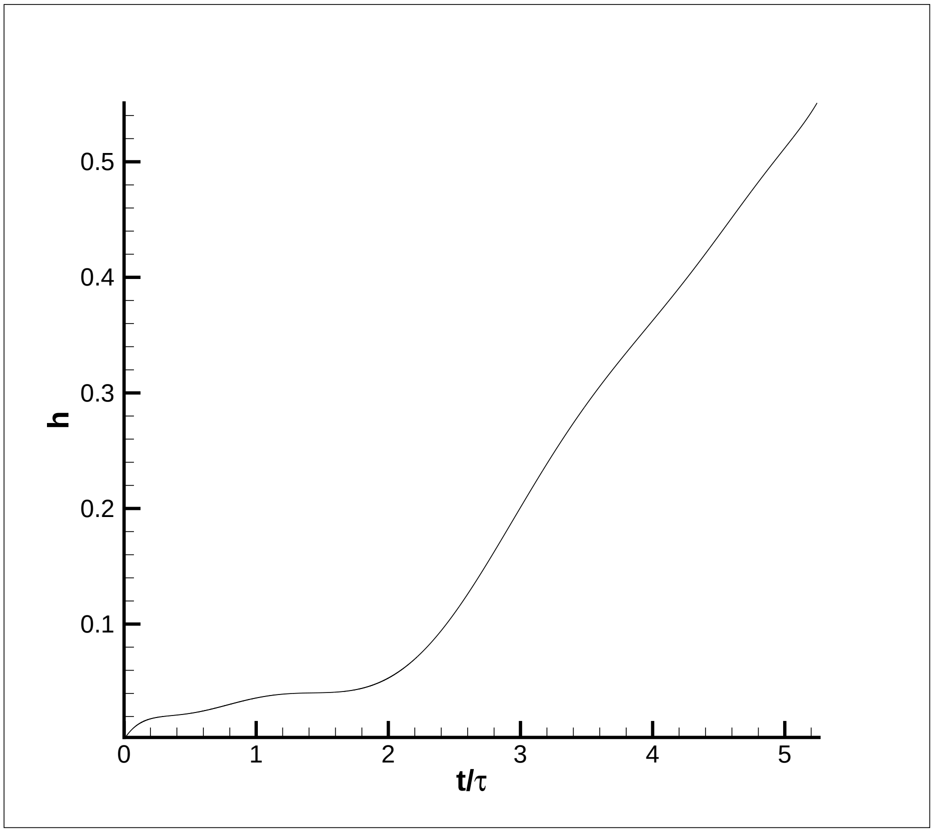


Figure 7.4: Time evolution of the total penetration length.

is completed with consumption of all substances [95]. The stoichiometric mixture fraction is introduced to describe the mixing of fluids considering a hypothetical reaction to measure the amount of fully mixed fluid [96].

Then, the total thickness is,

$$P_t = \int_H \langle X_p(\chi) \rangle_{xz} dy \quad \textbf{(7.19)}$$

Again, following the Cook and Dimotakis [90], the maximum mole fraction (i.e., maximum product thickness) is defined as,

$$P_m = \int_H X_p(\langle \chi \rangle_{xz}) dy \quad \textbf{(7.20)}$$

The ratio of these two thicknesses (i.e., the ratio of mixing length to an entrainment length), θ, is used to measure the mixing efficiency in the mixing layer,

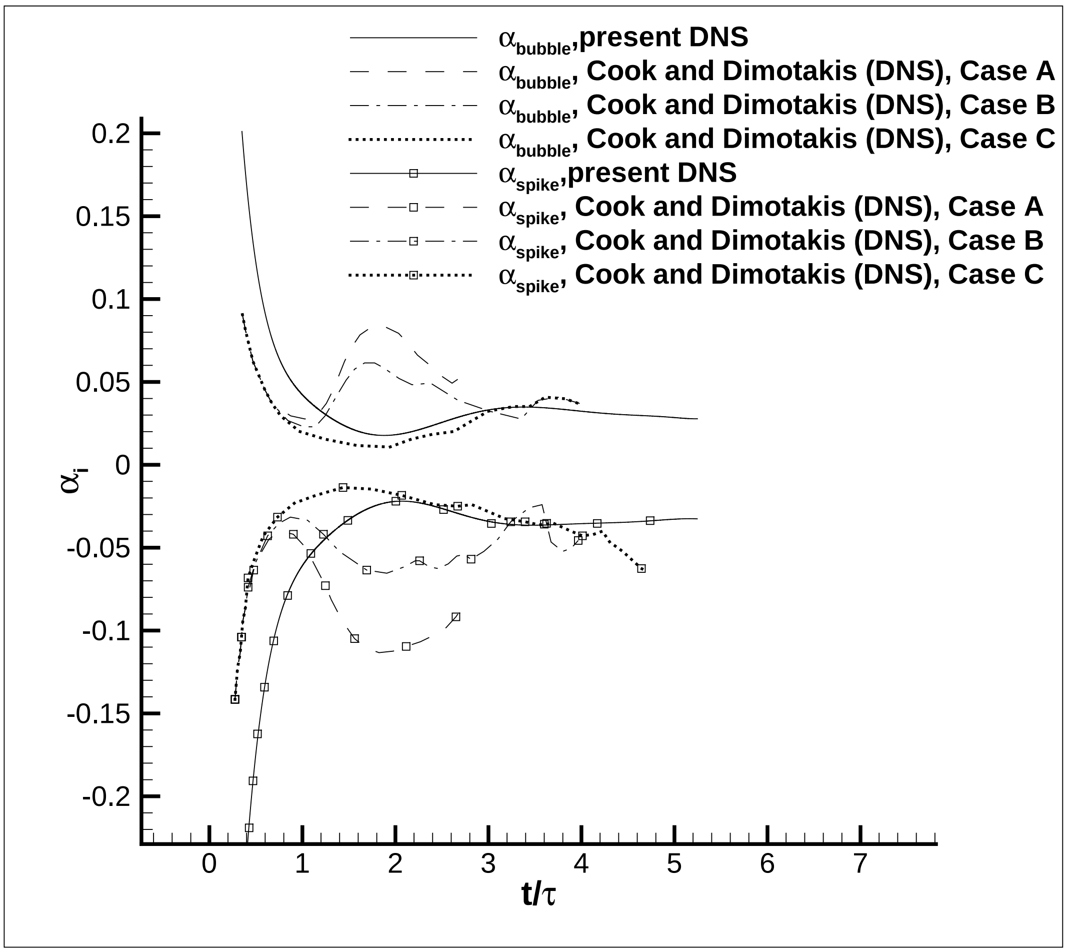


Figure 7.5: The evolution of the growth rate coefficients for bubble and spike.

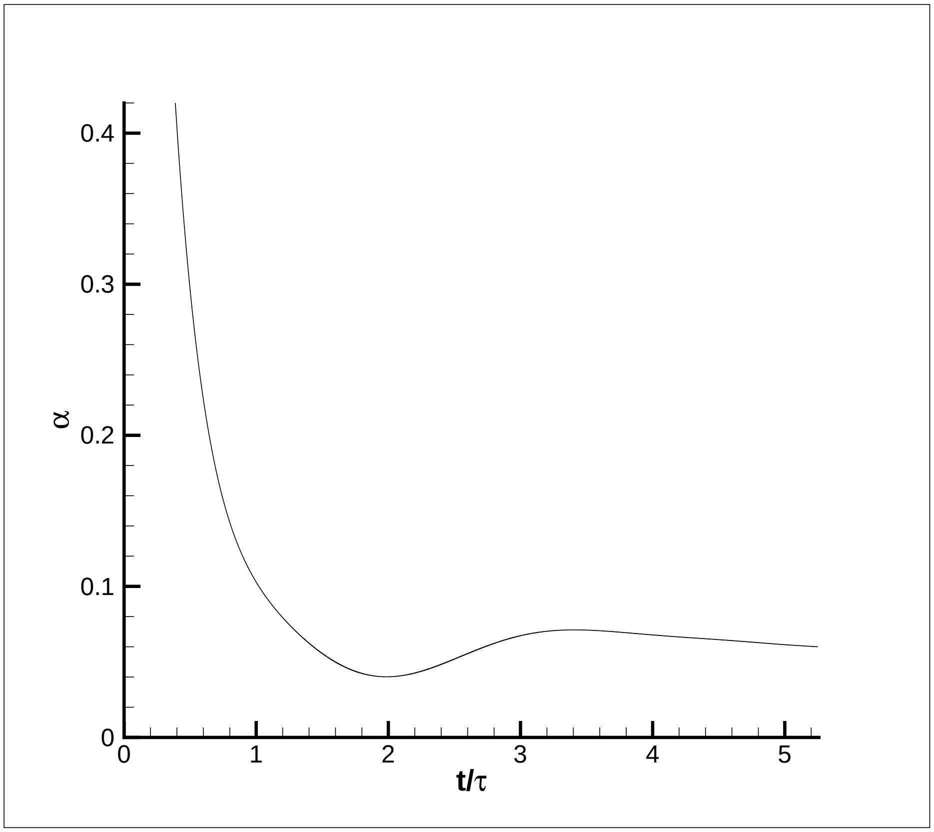


Figure 7.6: The evolution of the total growth rate coefficient.

$$\theta = \frac{P_t}{P_m} \tag{7.21}$$

It gives a similar result with the metric suggested by Youngs [97],

$$\theta = \frac{\int_{-\infty}^{+\infty} < X(1-X) > dy}{\int_{-\infty}^{+\infty} < X >< (1-X) > dy} \tag{7.22}$$

The both θ values are plotted against time in Fig. 7.7 and give similar results. $\theta = 1$ means all fluid in the mixing layer is molecularly mixed, while $\theta = 0$ corresponds to the immiscible case where the two fluids do not mix. Its value changes in time, it is not constant. The final values in our simulation is ~ 0.80 for θ_{Youngs} and ~ 0.83 for θ_{Cook}. These are in good agreement with Cook and Dimotakis [90], Cook *et. al.* [98], and Burton [94]. They report ≈ 0.8.

Taylor micro-scales are associated with the local strain rate and can be written for this flow via averaging over the homogeneous directions as,

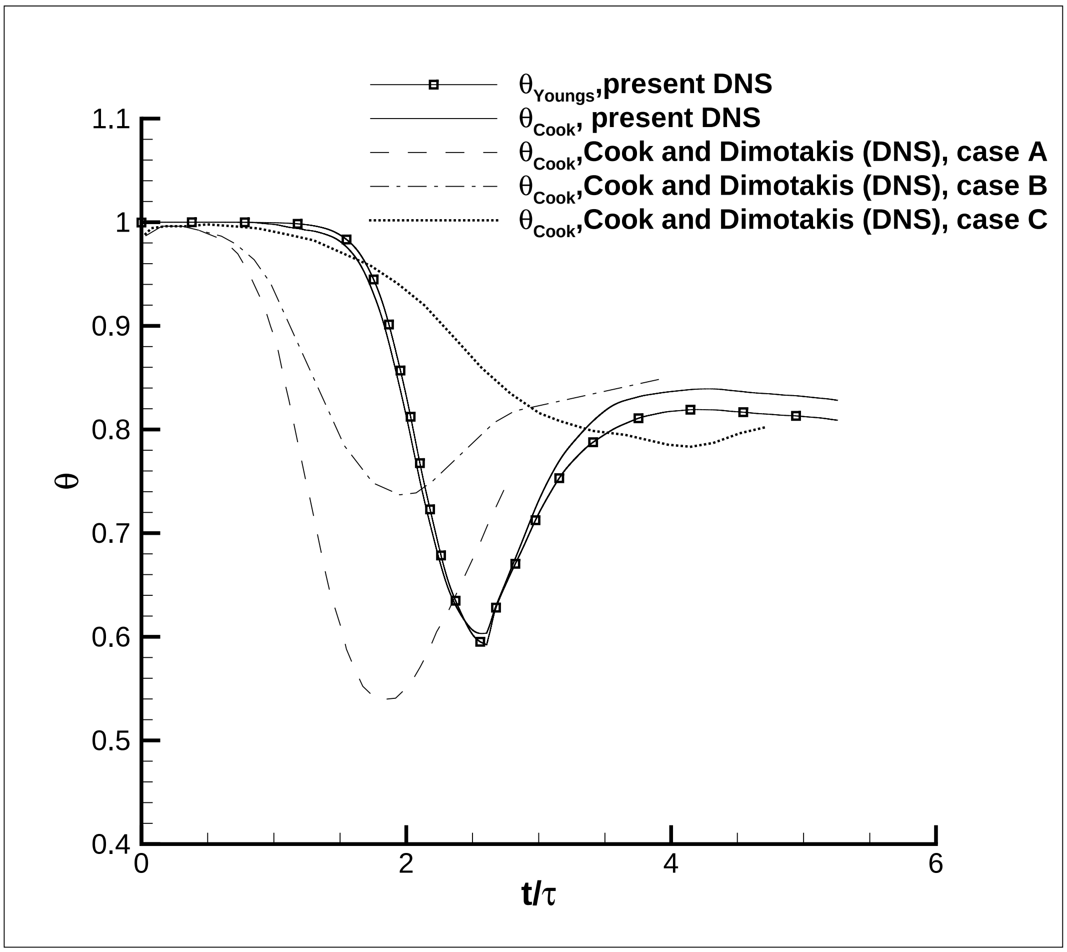


Figure 7.7: Time evolution of the mixing parameter θ.

$$\lambda_x = \sqrt{\frac{\langle u^2 \rangle_{xz}}{\langle (\frac{\partial u}{\partial x})^2 \rangle_{xz}}}; \qquad \lambda_y = \sqrt{\frac{\langle v^2 \rangle_{xz}}{\langle (\frac{\partial v}{\partial y})^2 \rangle_{xz}}}; \qquad \lambda_z = \sqrt{\frac{\langle w^2 \rangle_{xz}}{\langle (\frac{\partial w}{\partial z})^2 \rangle_{xz}}} \tag{7.23}$$

on the $y = 0$ plane. Assuming the statistical isotropy, on that plane, λ_x and λ_z can be averaged into a single micro-scale,

$$\lambda_{xz} = \frac{\lambda_x + \lambda_z}{2} \tag{7.24}$$

Then, Reynolds numbers can also be defined for each micro-scale as,

$$Re_{\lambda,x} = \frac{\langle \rho \rangle_{xz} \lambda_x \sqrt{\langle u^2 \rangle_{xz}}}{\mu_0}; \qquad Re_{\lambda,y} = \frac{\langle \rho \rangle_{xz} \lambda_y \sqrt{\langle v^2 \rangle_{xz}}}{\mu_0}; \qquad Re_{\lambda,x} = \frac{\langle \rho \rangle_{xz} \lambda_z \sqrt{\langle w^2 \rangle_{xz}}}{\mu_0} \tag{7.25}$$

and for the averaged Taylor micro-scale,

$$Re_{\lambda,xz} = \frac{Re_{\lambda,x} + Re_{\lambda,z}}{2} \tag{7.26}$$

The temporal evolution of the Taylor micro-scales and corresponding Reynolds numbers in the $y = 0$ plane are presented in Fig.7.8 and in Fig.7.9, respectively. Taylor micro-scales and corresponding Reynolds numbers increase in time as small-scale structures are merging into larger ones. Their differences also grow in time which is a measure of anisotropy [90, 99]. The ratio of vertical to horizontal micro-scales is found ~ 4.8 at maximum, and approaches to ~ 1.8 at late times. Cook and Zhou [99] found 3.7 and 1.4 in their high resolution DNS. Cook and Dimotakis give ≈ 4.5 and ≈ 2 in their DNS studies for the Case C. Taylor Reynolds numbers reach to ~ 90 for vertical and ~ 30 for horizontal at the end of the simulation. Zhou and Cook [99] and Cook and Dimotakis [90] obtain almost the same values in their DNS studies. A Taylor Reynolds number value of ~ 100 is needed for mixing transition [100].

In order to analyze turbulent nature of the development of the RTI, both the Reynolds stress terms and their anisotropies were also introduced. The Reynolds stresses are obtained as,

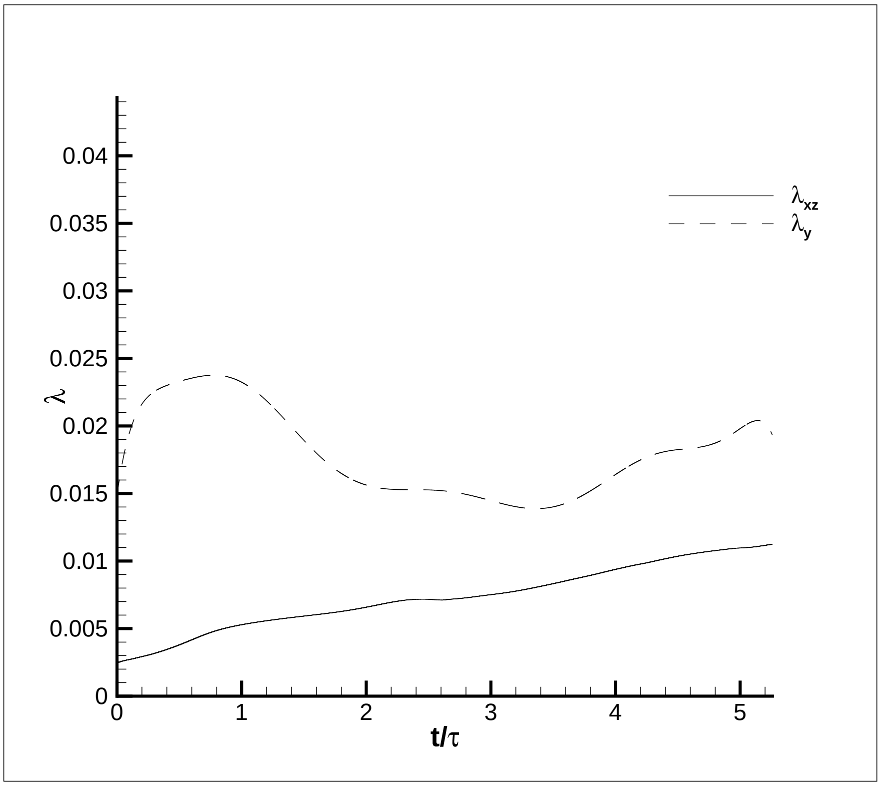


Figure 7.8: The evolution of the Taylor micro-scales in time.

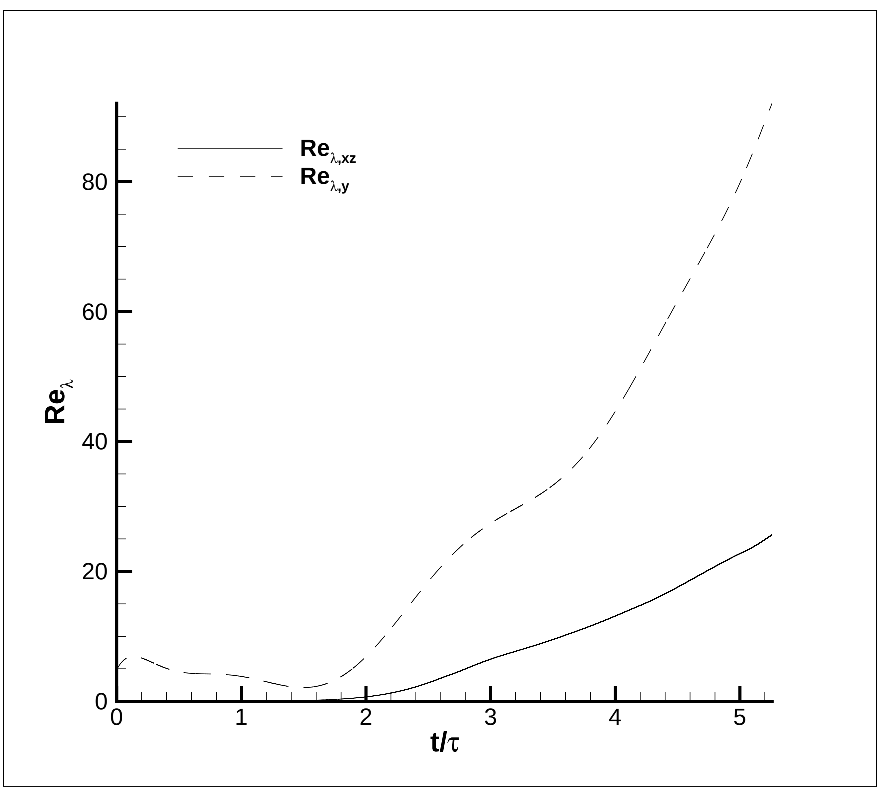


Figure 7.9: The time evolution of the Reynolds numbers based on Taylor micro-scales.

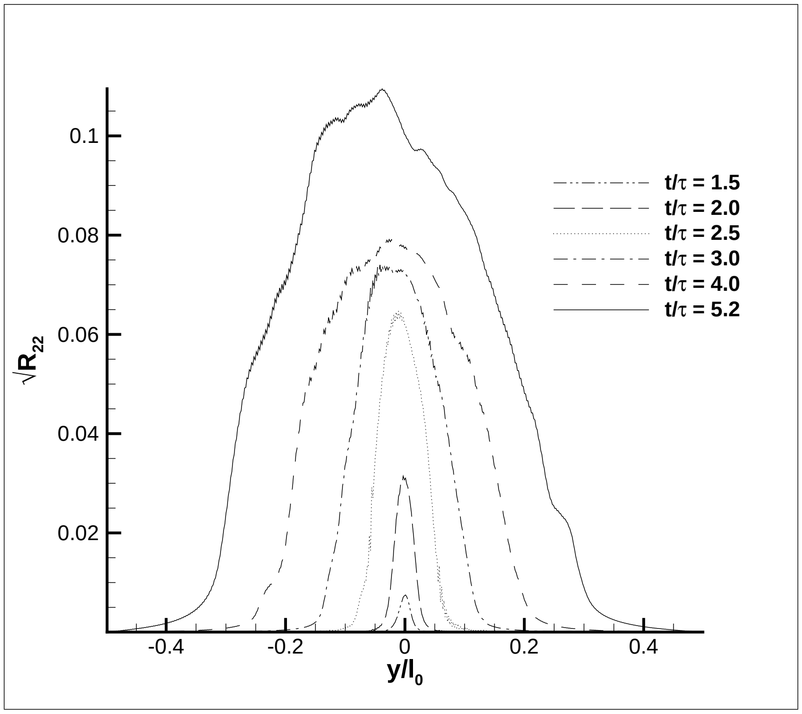


Figure 7.10: The evolution of the *rms* of the vertical component of the Reynolds stress term in time.

$$R_{ij} = \frac{\overline{\rho u_i'' u_j''}}{\overline{\rho}} \tag{7.27}$$

where $\{\}''$ denotes Favre fluctuations. Fig. 7.10 shows the development of the *rms* of the normal-wise component of the Reynolds stress term in time. It starts from an initial value and reaches to their specific peak values at the end of the simulation.

Similarly, the peak values of the Reynolds stress terms for most of directions given in Fig. 7.11. The peak value of the normal-wise component is almost two times higher than the other components. This is compatible with the direction of the gravitational acceleration. We have also found that cross-components of terms are very small due to approximate symmetry of the large-scale structures. These two observations are perfectly matched to the results obtained by Ramapraphu [101].

Once R_{ij}'s are found, anisotropy of these terms, b_{ij} can also be calculated easily via,

$$b_{ij} = \frac{R_{ij}}{2K} - \frac{1}{3}\delta_{ij}, \tag{7.28}$$

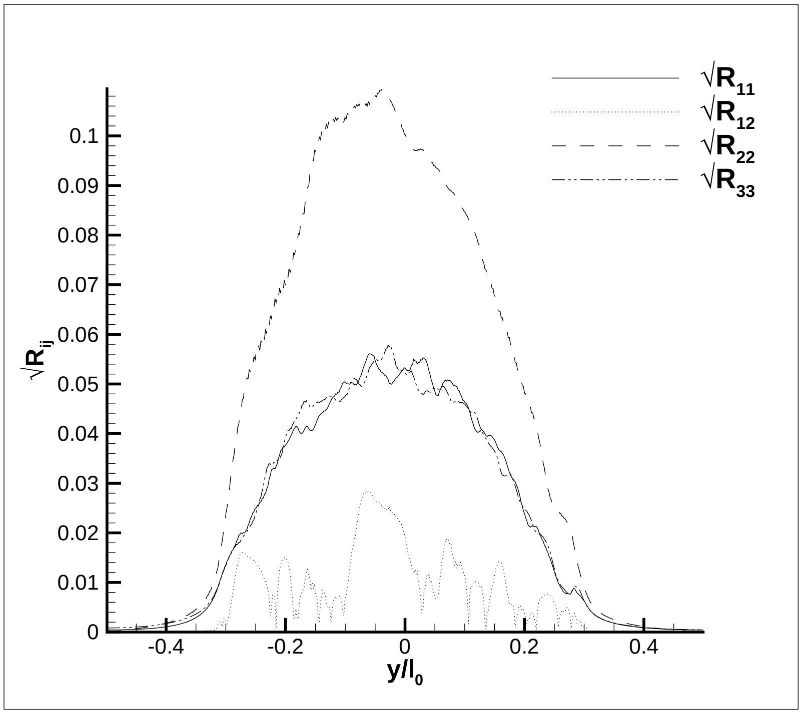


Figure 7.11: The magnitude of the turbulent stress terms at the end of the simulation.

where K is the turbulent kinetic energy and δ_{ij} is the Kronocker delta. Then, b_{ij} is integrated along the vertical direction. The anisotropy of Reynolds stresses show the character and the dependency of the velocity fluctuations to the direction in turbulent flows. Time evolution of some of the b_{ij} terms are given in Fig. 7.12.

The change in the cross terms are negligible and b_{33} is also similar to b_{11}. In the non-linear region (i.e., after a sufficiently long time), the values of the components of b_{ij} reach a constant value. The difference characterizes the large-scale anisotropy, as pointed out in [101]. The peak anisotropy value in the normal-wise direction is also stronger than the other directions.

The most important Reynolds number for this flow is the outer-scale Reynolds number, since only the total extend of the mixing zone, $h(t)$, and its growth rate, $\dot{h}(t)$, change of its value in time, i.e.,

$$Re_h = \frac{\rho_{av} h \dot{h}}{\mu_{av}} \tag{7.29}$$

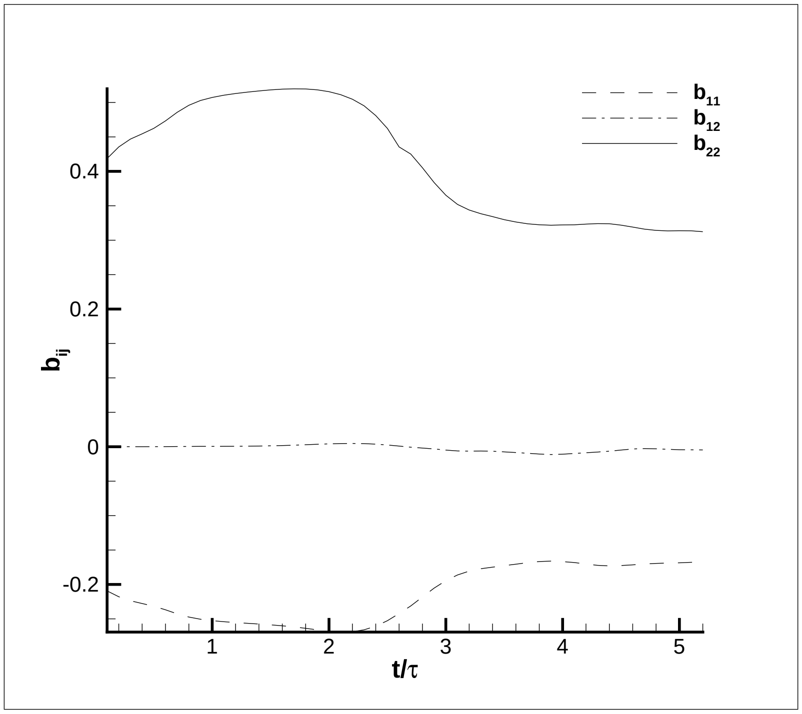


Figure 7.12: The growth of the components of the Reynolds stress anisotropy terms in time.

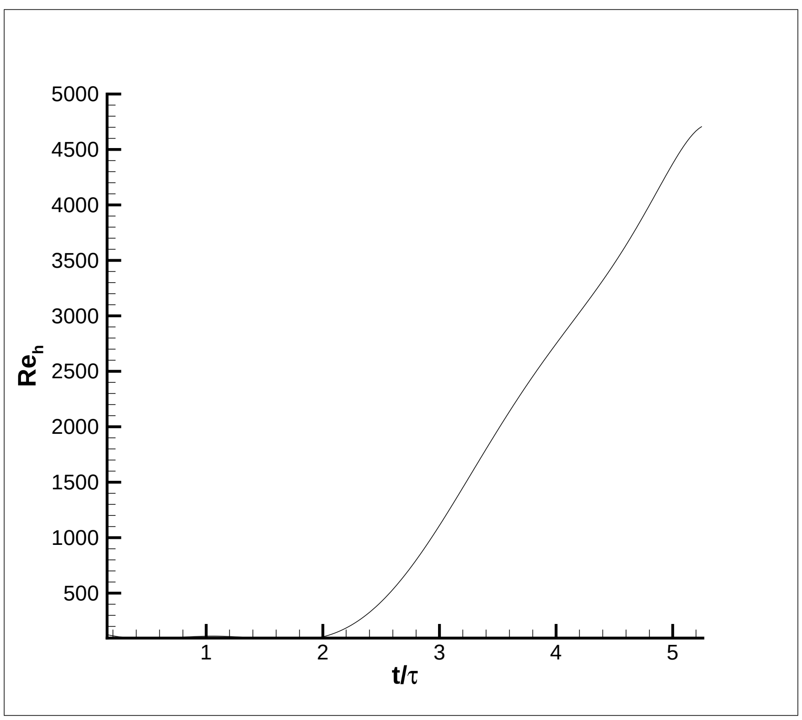


Figure 7.13: The growth of the outer-scale Reynolds number in time.

where $\{\}_{av}$ denotes the initial *average* of the variable. It reaches to a value about 5000 at the end of the simulation, in consistency with the previous DNS results [90, 99].

7.3.2 The compressible case

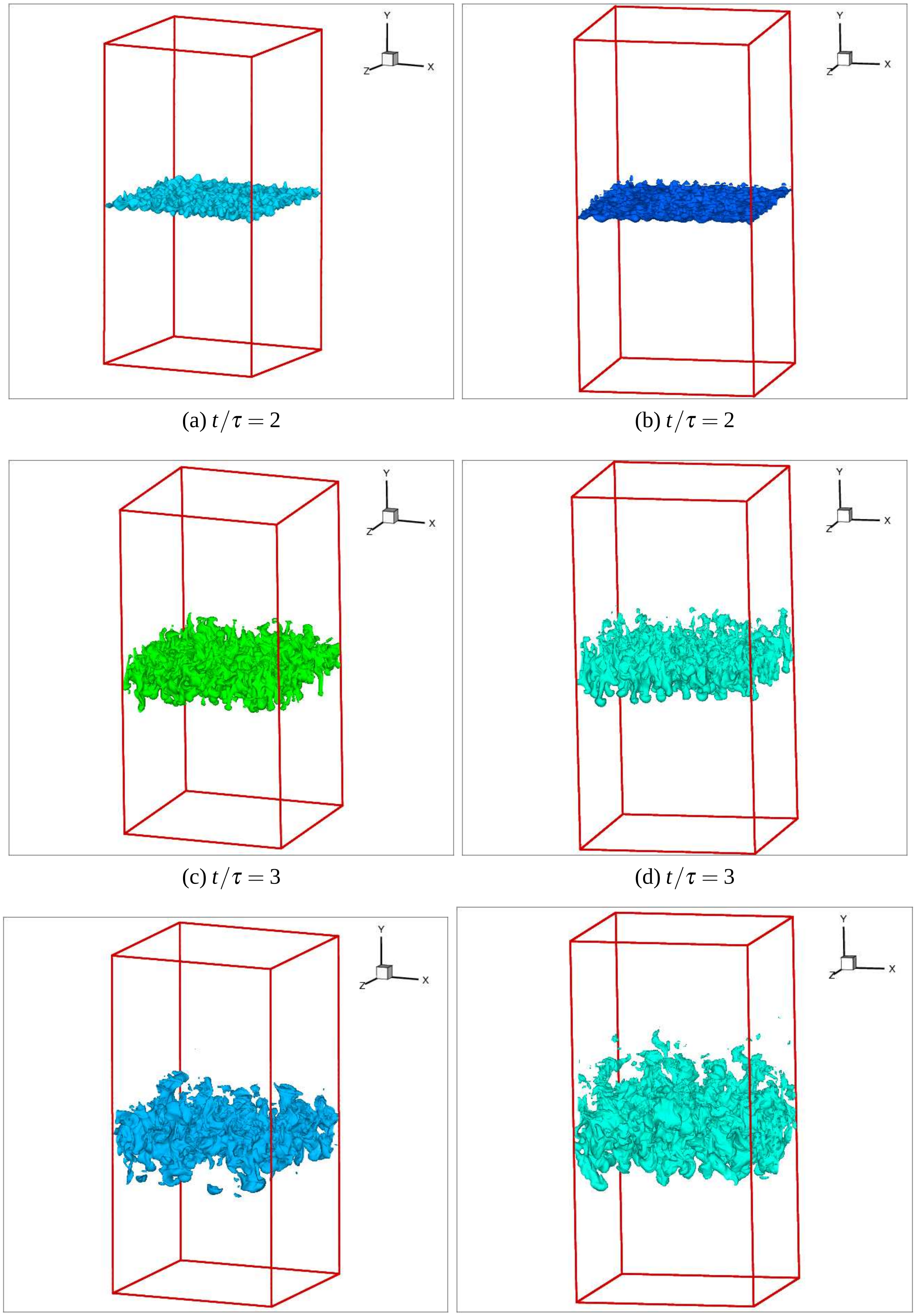
After showing that the present implicit, non-dissipative DNS algorithm and the solver developed based on it, *i*DNS, are capable of capturing the correct physics of RTI, the Mach number effects on the development of the instability were also studied. In this case, the Mach number was increased to 1.1 by decreasing the equilibrium pressure. Other simulation details were kept same with the base incompressible case.

Fig.7.14 shows the evolution of the local mole fraction for the compressible case, and compares it to the base incompressible case. During the early stage diffuse growth, there is no significant difference. However, together with the non-linear phase (i.e., $t/\tau \sim 4.0$), the structures develop and merge faster. Then, the compressible simulation reaches the edge of the domain rapidly, and finishes earlier.

Fig. 7.15 and Fig. 7.16 compare bubble and spike penetration lengths and the growth rate coefficients, respectively. The effect described above is more obvious here. An increase in the h_{bubble} and α_{bubble} at non-linear stage are seen from the figures. Spikes are affected through the late non-linear stage. The slow change in the spike morphology can possibly be explained by the combination of pressure drag and reflecting shocks from the bottom [102, 103], since they fall into the heavier pressure region and encounter reflecting shocks.

As mentioned above, during the simulations, the bow shocks with different strengths were also produced [103] at both sides. They inevitably reflect from the boundaries to inside and cause the statistics to be more wrinkle-type. However, their overall effects are not so severe at this Mach number. Larger domains are required to allow the instability to develop freely and to understand the compressibility effects better.

Fig.7.17 shows the Mach number effects on turbulent mixing in the mixing region. Unlike the incompressible case, two definitions of θ produced different results.



(a) $t/\tau = 2$ (b) $t/\tau = 2$

(c) $t/\tau = 3$ (d) $t/\tau = 3$

(e) $t/\tau = 4$ (f) $t/\tau = 4$

Figure 7.14: Comparison of the development of the local mole fraction field in terms of iso-surfaces (X=0.5) for incompressible (left column) and compressible cases (right column).

As explained in [104], a faster growth rate is accompanied by a reduction in mixing. The θ value predicted by θ_{Youngs} is more consistent with the results of Stone and Gardiner [104].

We have applied our in-house DNS solver, *i*DNS, to Rayleigh-Taylor Instability problem and successfully compared our results to the prior studies, as well as the findings of linear theory. It was shown that the algorithm and *i*DNS are able to handle the physics of the problems including multi-scales, transition and mixing simultaneously.

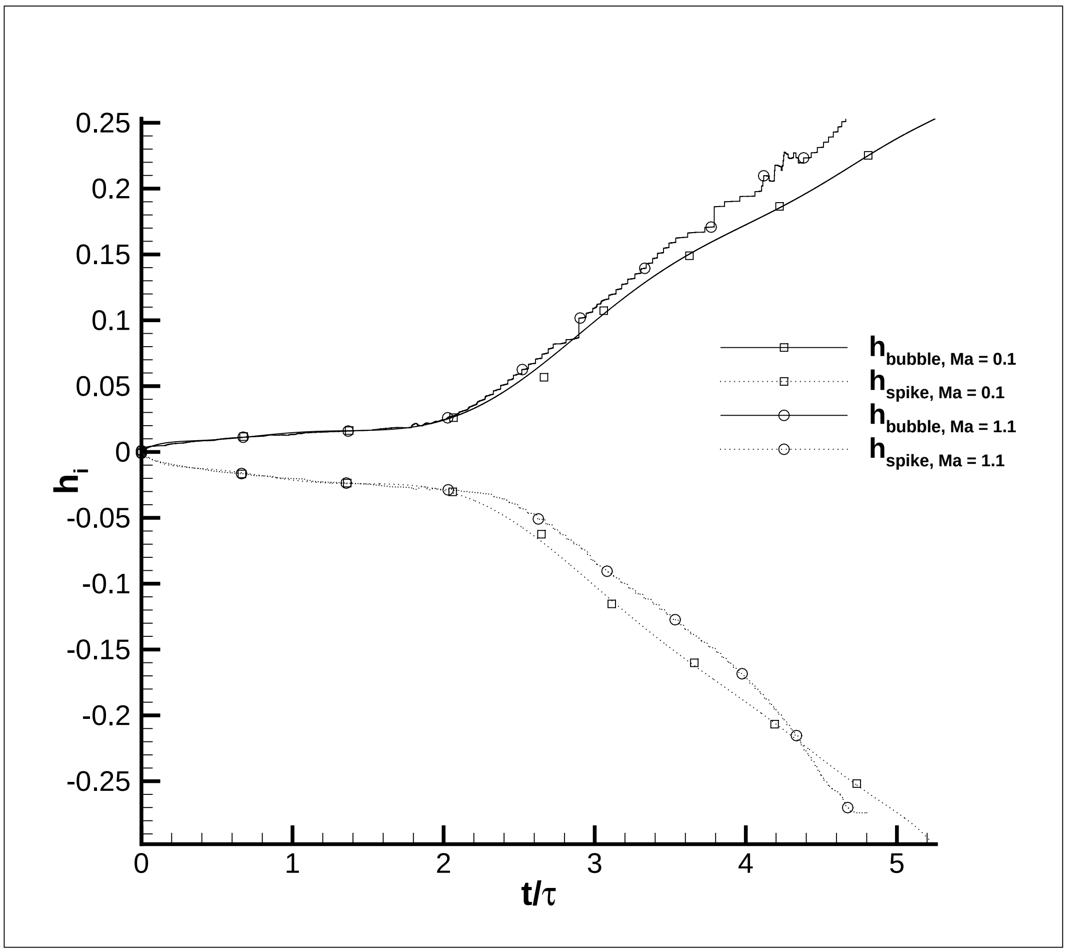


Figure 7.15: The comparison of the bubble and spike penetration lengths for incompressible and compressible cases.

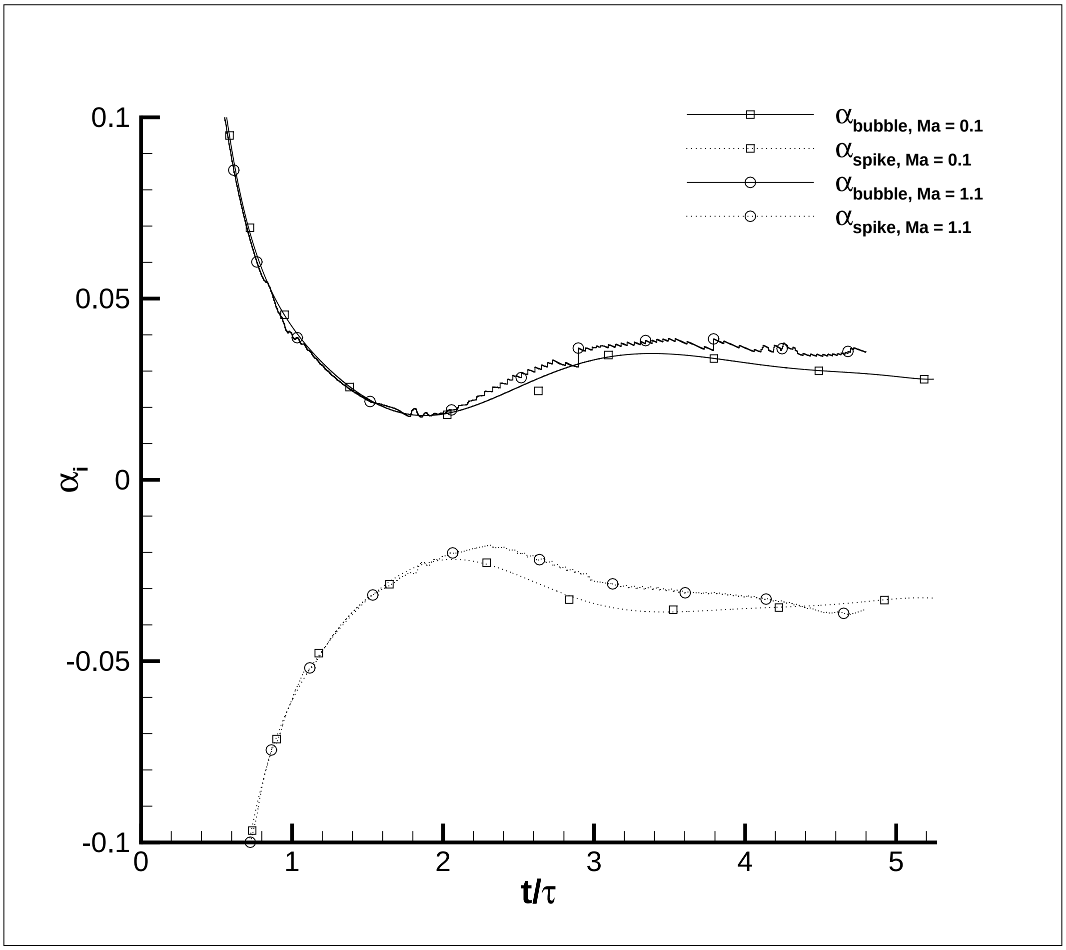


Figure 7.16: The comparison of the bubble and spike growth rate coefficients for incompressible and compressible cases.

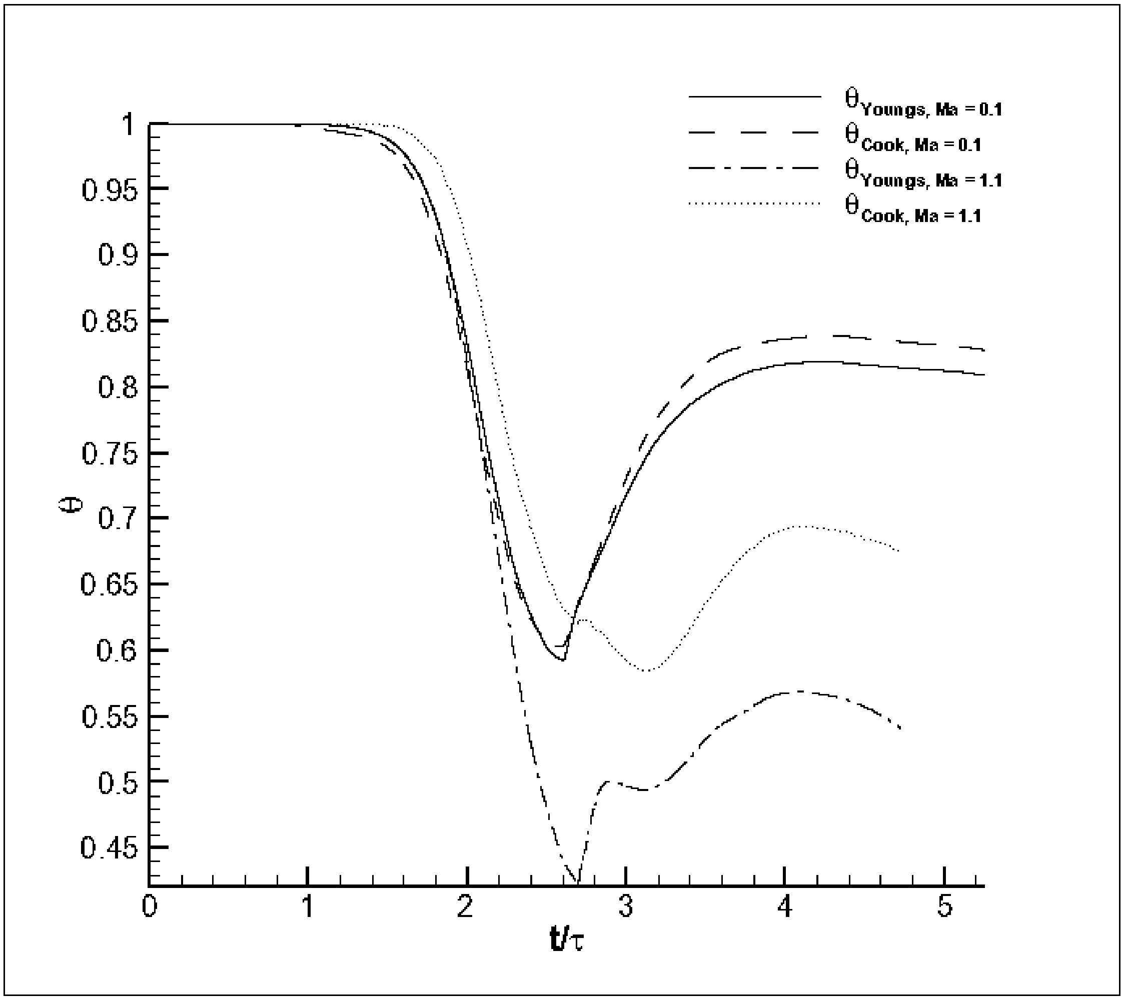


Figure 7.17: The comparison of two different θ's for incompressible and compressible cases.

8. CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions

In this study, a fully implicit, non-dissipative, discrete kinetic energy conserving, finite volume solution algorithm which solves compressible Navier-Stokes equations was chosen and the fully parallel DNS solver so-called *i*DNS implemented based on it. Then, *i*DNS has been applied to the incompressible and the compressible turbulent flow problems. These problems were selected from the literature to assess the algorithm's ability to capture multi-scale physics which laminar, transitional and turbulent regimes are simultaneously found together with varying Mach numbers. This type of flow problems serve as challenging test cases for the various features of CFD algorithms and are encountered in many physical and engineering flows. In order to improve the convergence properties of the algorithm, a relaxation procedure to pressure in time was also added into the solver. During these simulations, density was updated and allowed to change, even in nearly incompressible simulations.

Transition to turbulence in Taylor-Green Vortex (TGV) flow, mixing in Turbulent Shear Layer (TSL) and Rayleigh-Taylor Instability (RTI) were chosen for these purposes. Various flow diagnostics specific to the selected problem were calculated to analyze the results.

TGV flow was designed to introduce the fundamental mechanisms of vortex stretching and energy cascade. It has been studied for many years by authors to check their algorithms and codes. In our TGV flow simulations, the time for transition to turbulence and the peak value of kinetic energy decay rate were well-predicted by the algorithm. This problem also served as a verification case for the developed solver.

TSL is still an active research area. It can be encountered in many engineering and physical flows. The flow can rapidly evolve into turbulent state in three-dimension. Additionally, compressibility effects on TSL have re-gained an importance with the development of supersonic flight technology. We studied density ratio 1 in all cases. Our quasi-incompressible results where $M_c = 0.3$ are in agreement with prior experimental and numerical studies. Different M_c's up to 0.9 were simulated for analyzing the compressibility effects on the growth rate of instability. The effects of compressibility were found stabilizing characterized especially by a decrease in turbulent production and pressure strain, consistently to the other studies.

RTI is another challenging problem of fluid dynamics. It is driven by baroclinic vorticity. Incompressible and compressible cases were studied in this thesis. Atwood number is set to 0.3 for all simulations. Unlike the many other methods that perturb the interface, vertical velocity initially was perturbed to trigger the instability. The base incompressible simulations were successfully compared with the experimental and numerical data available in the literature. In order to perform compressible simulations, the equilibrium pressure was decreased to a specific value and the reference quantities were chosen properly to prevent unphysical fields both at initial and in updating step. In our compressible simulations showed that, at early stage diffuse development, there is no significant difference between incompressible and compressible growth rates. However, in late time non-linear development, faster growth rate was observed. Spikes and bubbles reach to boundaries of the domain earlier than the incompressible case. Mixing efficiency reduces to a lower value, as previously found in the literature. Additionally, shock waves were generated in the flow. The reflected shock waves from the boundaries affect the development of the instability and the statistics at very late times. Since simulation was ended when spikes/bubbles reached to end of the domain, this was not really a concern. Larger domain sizes are required for high Mach number simulations.

8.2 Future Works

Some further research topics are also suggested in this section, regarding our experiences gained during the study. In order to make the algorithm and the solver as well, applicable to high Reynolds number flows, a hybrid Large Eddy Simulation-Unsteady Reynolds Averaging (LES-URANS) procedure can be developed. Similarly, complex boundaries, other than walls, can also be implemented into the solver. To handle the problems on boundary-fitted grids, the algorithm can be modified. It is also possible to extend the algorithm to magneto-hydrodynamic flows to analyze the turbulent flows with magnetic fields. Another research topic could be the separated-reattached turbulent flows which mostly encountered in engineering flows as a very challenging problem. An external dissipative or shock capturing procedure would also be useful when simulating very high Mach number flows.

REFERENCES

[1] **Blazek, J.**, (2001). *Computational Fluid Dynamics: Principles and Applications*, Elsevier.

[2] **Ferziger, J. and Perić, M.**, (2002). *Computational Methods for Fluid Dynamics*, Springer-Verlag.

[3] **Orszag, S.A.**, (1970). Analytical theories of turbulence, *Journal of Fluid Mechanics*, **41(02)**, 363–386.

[4] **Rogallo, R.S.**, (1981). Numerical experiments in homogeneous turbulence, *NASA STI/Recon Technical Report*, **81**, 31508.

[5] **Kim, J.**, **Moin, P. and Moser, R.**, (1987). Turbulence statistics in fully developed channel flow at low Reynolds number, *Journal of Fluid Mechanics*, **177**, 133–166.

[6] **Spalart, P.R.**, (1988). Direct simulation of a turbulent boundary layer up to $R_\theta =$ 1410, *Journal of Fluid Mechanics*, **187**, 61–98.

[7] **Sahin, M. and Owens, R.G.**, (2003). A novel fully implicit finite volume method applied to the lid-driven cavity problem—Part I: High Reynolds number flow calculations, *International Journal for Numerical Methods in Fluids*, **42(1)**, 57–77.

[8] **Martin, M.P. and Candler, G.V.**, (2006). A parallel implicit method for the direct numerical simulation of wall-bounded compressible turbulence, *Journal of Computational Physics*, **215(1)**, 153 – 171.

[9] **Seidl, V.**, **Muzaferija, S.**, **Peric, M. and Thiele, F.**, (1996). Direct Numerical Simulation of Flow around Sphere with an Unstructured Fully Implicit Parallel Navier-Stokes Solver, **N.S.J.P. P. Schiano A. Ecer**, editor, *Parallel Computational Fluid Dynamics: Algorithms and Results Using Advanced Computers : Proceedings of the Parallel CFD '96 Conference*, Elsevier, pp.100–107.

[10] **Hou, Y. and Mahesh, K.**, (2005). A robust, colocated, implicit algorithm for direct numerical simulation of compressible, turbulent flows, *Journal of Computational Physics*, **205(1)**, 205 – 221.

[11] **Hirsch, C.**, (2007). *Numerical Computation of Internal and External Flows*, Butterworth-Heinemann.

[12] **Stokes, G.G.**, 1845. On the Theories of Internal Friction of Fluids in Motion, *Trans. Cambridge Phil. Soc.*, **8**, 287–305.

[13] **White, F.**, (1991). *Viscous Fluid Flow*, McGraw-Hill.

[14] **Wesseling, P.**, (2001). *Principles of Computational Fluid Dynamics*, Springer-Verlag.

[15] **Kay, J.M. and Nedderman, R.M.**, (1985). *Fluid Mechanics and Transfer Processes*, Cambridge University Press.

[16] **Doom, J.J.**, (2009). Direct Numerical Simulation of Turbulent, Chemically Reacting Flows, Ph.D. thesis, University of Minnesota, Aerospace Engineering and Mechanics, http://www.aem.umn.edu/ mahesh/publpdf/thesis/Jeff_thesis.pdf.

[17] **Wall, C.**, **Pierce, C.D. and Moin, P.**, (2002). A Semi-implicit Method for Resolution of Acoustic Waves in Low Mach Number Flows, *Journal of Computational Physics*, **181(2)**, 545 – 563.

[18] **Kim, D. and Choi, H.**, (2000). A Second-Order Time-Accurate Finite Volume Method for Unsteady Incompressible Flow on Hybrid Unstructured Grids, *Journal of Computational Physics*, **162(2)**, 411 – 428.

[19] **Zang, Y.**, **Street, R.L. and Koseff, J.R.**, (1994). A Non-staggered Grid, Fractional Step Method for Time-Dependent Incompressible Navier-Stokes Equations in Curvilinear Coordinates, *Journal of Computational Physics*, **114(1)**, 18 – 33.

[20] **S., B.**, **K., M. and Constantinescu, G.**, (2002). Colocated finite-volume schemes for large-eddy simulation on unstructured meshes, *Proceedings of the 2002 Summer Program*, Center for Turbulence Research, Stanford University and NASA Ames.

[21] **Lipschutz, S.**, (2009). *Vector Analysis and An Introduction to Tensor Analysis*, McGraw-Hill, New York.

[22] **Deng, Q.H. and Tang, G.F.**, (2002). Special Treatment of Pressure Correction Based on Continuity Conservation in a Pressure-based Algorithm, *Numerical Heat Transfer Part B: Fundamentals*, **42(1)**, 73–92.

[23] **Walton, G.N.**, (1989). AIRNET - A Computer Program for Bulding Airflow Network Modeling, Technical Report*NISTIR 89-4072*, National Institute of Standards and Technology, `www.bfrl.nist.gov/IAQanalysis/docs/AIRNET.pdf`.

[24] **Conte, S.D. and de Boor, C.**, (1972). *Elementary Numerical Analysis: An Algorithmic Approach*, McGraw-Hill.

[25] **Balay, S.**, **Gropp, W.D.**, **McInnes, L.C. and Smith, B.F.**, (2008). PETSc home page, http://www.mcs.anl.gov/petsc.

[26] **Balay, S.**, **Gropp, W.D.**, **McInnes, L.C. and Smith, B.F.**, (1997). Efficienct Management of Parallelism in Object Oriented Numerical Software Libraries, **E. Arge**, **A.M. Bruaset and H.P. Langtangen**, editors, *Modern Software Tools in Scientific Computing*, Birkhauser Press, p.163–202.

[27] **Balay, S.**, **Gropp, W.D.**, **McInnes, L.C. and Smith, B.F.**, (2009). *PETSc Users Manual*, Technical ReportANL-95/11 - Revision 3.1, Argonne National Laboratory.

[28] **Gropp, W.**, **Lusk, E. and Skjellum, A.**, (1994). *Using MPI: Portable Parallel Programming with the Message-Passing Interface*, MIT Press, Cambridge, MA.

[29] **Falgout, R.**, **Baker, A.**, **Chow, E.**, **Henson, V.E.**, **Jones, J.**, **Lee, B.**, **Painter, J.**, **Tong, C.**, **Treadway, T.**, **Vassilevski, P. and Yang, U.M.**, (2006), *Hypre*, `http://www.llnl.gov/CASC/hypre/`.

[30] **Karypis, G. and Kumar, V.**, (2009), *MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0*, `http://www.cs.umn.edu/~metis`.

[31] **Schwarz, H.A.**, (1890). *Gesammelte Mathemathische Abhandlungen*, Springer.

[32] **Smith, B.F.**, **Bjørstad, P.E. and Gropp, W.D.**, (1996). *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, New York.

[33] **Wohlmuth, B.** *Discretization Techniques and Iterative Solvers Based on Domain Decomposition*, volume 17, Springer, Heidelberg.

[34] **Saad, Y.**, (2003). *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition.

[35] **Grama, A.**, **Gupta, A.**, **Karypis, G. and Kumar, V.**, (2003). *Introduction to Parallel Computing, second edition*, Addison-Wesley, Essex, England.

[36] **Hill, M.D.**, (1990). What is scalability?, *SIGARCH Comput. Archit. News*, **18**, 18–21.

[37] **Tennekes, H. and Lumley, J.**, (1972). *A First Course in Turbulence*, The MIT Press.

[38] **Taylor, G.I. and Green, A.E.**, (1937). Mechanism of the Production of Small Eddies from Large Ones, *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences*, **158(895)**, 499–521.

[39] **Brachet, M.E.**, **Meiron, D.I.**, **Orszag, S.A.**, **Nickel, B.G.**, **Morf, R.H. and Frisch, U.**, (1983). Small-scale structure of the Taylor–Green vortex, *Journal of Fluid Mechanics*, **130**, 411–452.

[40] **Brachet, M.**, (1991). Direct simulation of three-dimensional turbulence in the Taylor-Green vortex, *Fluid Dynamics Research*, **8(1–4)**, 1 – 8.

[41] **Fauconnier, D.**, **Langhe, C.D. and Dick, E.**, (2009). A family of dynamic finite difference schemes for large-eddy simulation, *Journal of Computational Physics*, **228(6)**, 1830 – 1861.

[42] **Drikakis, D.**, **Fureby, C.**, **Grinstein, F.**, **Hahn, M. and Youngs, D.**, (2006). LES of Transition to Turbulence in the Taylor Green Vortex, **E. Lamballais**, **R. Friedrich**, **B. Geurts and O. Métais**, editors, *Direct and Large-Eddy Simulation VI*, Springer Netherlands, pp.159–166.

[43] **Drikakis, D.**, **Fureby, C.**, **Grinstein, F.F. and Youngs, D.**, (2007). Simulation of transition and turbulence decay in the Taylor–Green vortex, *Journal of Turbulence*, N20.

[44] **Shu, C.W.**, **Don, W.S.**, **Gottlieb, D.**, **Schilling, O. and Jameson, L.**, (2005). Numerical Convergence Study of Nearly Incompressible, Inviscid Taylor–Green Vortex Flow, *Journal of Scientific Computing*, **24**, 1–27.

[45] **van Rees, W.M.**, **Leonard, A.**, **Pullin, D. and Koumoutsakos, P.**, (2011). A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high Reynolds numbers, *Journal of Computational Physics*, **230(8)**, 2794 – 2805.

[46] **Frisch, U.**, (1995). *Turbulence: The Legacy of AN Kolmogorov*, Cambridge University Press.

[47] **Fauconnier, D.**, (2008). Development of a Dynamic Finite Difference Method for Large-Eddy Simulation, Ph.D. thesis, Ghent University, Faculty of Engineering, Department of Flow, Heat and Combustion Mechanics, http://users.ugent.be/ dfauconn/research.htm.

[48] **Pope, S.**, (2000). *Turbulent Flows*, Cambridge University Press.

[49] **Jeong, J. and Hussain, F.**, (1995). On the identification of a vortex, *Journal of Fluid Mechanics*, **285**, 69–94.

[50] **Hahn, M.**, (2008). Implicit Large-Eddy Simulation of Low-Speed Separated Flows Using High-Resolution Methods, Ph.D. thesis, Cranfield University, School of Engineering, Department of Aerospace Sciences, https://dspace.lib.cranfield.ac.uk/handle/1826/2633.

[51] **Kolmogorov, A.**, (1941). Dissipation of Energy in the Locally Isotropic Turbulence, *Proceedings of the USSR Academy of Sciences*, **32(16)**.

[52] **Taylor, G.I.**, (1935). Statistical theory of turbulence, *Proceedings of the Royal Society of London Series A, Mathematical and Physical Sciences*, **151(873)**, 421–444.

[53] **Landahl, T. and Mollo-Christensen, E.**, (1992). *Turbulence and Random Processes in Fluid Mechanics*, Cambridge University Press.

[54] **Sreenivasan, K.R. and Antonia, R.A.**, (1997). The Phenomenology of small-scale turbulence, *Annual Review of Fluid Mechanics*, **29**, 435–472.

[55] ***AGARD-AR-345***, (1998). A Selection of Test Cases for the Validation of Large-Eddy Simulations of Turbulent Flows, Technical Report 345.

[56] **Spencer, B.W. and Jones, B.G.**, (1971). Statistical investigation of pressure and velocity fields in the turbulent two-stream mixing layer, *AIAA Paper 71-613*.

[57] **Bell, J.H. and Mehta, R.D.** Development of a two-stream mixing layer from tripped and untripped boundary layers, *AIAA Journal*.

[58] **Riley, J. and Metcalfe, R.**, (1980). Direct Numerical Simulation of a perturbed turbulent mixing layer, *AIAA Paper 80-0274*.

[59] **Rogers, M.M. and Moser, R.D.**, (1992). The three-dimensional evolution of a plane mixing layer: the Kelvin–Helmholtz roll-up, *Journal of Fluid Mechanics*, **243**, 183–226.

[60] **Moser, R.D. and Rogers, M.M.**, 1(993). The three-dimensional evolution of a plane mixing layer: pairing and transition to turbulence, *Journal of Fluid Mechanics*, **247**, 275–320.

[61] **Rogers, M.M. and Moser, R.D.**, (1994). Direct simulation of a self-similar turbulent mixing layer, *Physics of Fluids*, **6(2)**, 903–923.

[62] **Bogdanoff, D.W.** Compressibility effects in turbulent shear layers, *AIAA Journal*.

[63] **Papamoschou, D. and Roshko, A.**, (1988). The compressible turbulent shear layer: an experimental study, *Journal of Fluid Mechanics*, **197**, 453–477.

[64] **Elliott, G.S. and Samimy, M.**, (1990). Compressibility effects in free shear layers, *Physics of Fluids A: Fluid Dynamics*, **2(7)**, 1231–1240.

[65] **Goebel, S.G. and Dutton, J.C.** Experimental study of compressible turbulent mixing layers, *AIAA Journal*.

[66] **Clemens, N.T. and Mungal, M.G.**, (1995). Large-scale structure and entrainment in the supersonic mixing layer, *Journal of Fluid Mechanics*, **284**, 171–216.

[67] **Vreman, A.W.**, (1995). Direct and large eddy simulation of the compressible turbulent mixing layer, Ph.D. thesis, University of Twente, http://search.proquest.com/docview/304209114?accountid=11638.

[68] **Vreman, A.W.**, **Sandham, N.D. and Luo, K.H.**, (1996). Compressible mixing layer growth rate and turbulence characteristics, *Journal of Fluid Mechanics*, **320**, 235–258.

[69] **Sarkar, S.**, (1995). The stabilizing effect of compressibility in turbulent shear flow, *Journal of Fluid Mechanics*, **282**, 163–186.

[70] **Freund, J.B.**, **Lele, S.K. and Moin, P.**, (2000). Compressibility effects in a turbulent annular mixing layer. Part 1. Turbulence and growth rate, *Journal of Fluid Mechanics*, **421**, 229–267.

[71] **Pantano, C. and Sarkar, S.**, (2002). A study of compressibility effects in the high-speed turbulent shear layer using direct simulation, *Journal of Fluid Mechanics*, **451**, 329–371.

[72] **Breidenthal, R.**, (1990). The sonic eddy-A model for compressible turbulence, *AIAA Paper 90-0495*.

[73] **Davidson, L.**, (2007). Using Isotropic Synthetic Fluctuations as Inlet Boundary Conditions for Unsteady Simulations, *Advances and Applications in Fluid Mechanics*, **1(1)**, 1–35.

[74] **Kraichnan, R.H.**, (1970). Diffusion by a Random Velocity Field, *Physics of Fluids*, **13(1)**, 22–31.

[75] **Hadjadj, A.**, **Yee, H. and Sjögreen, B.**, (2012). LES of temporally evolving mixing layers by an eighth-order filter scheme, *International Journal for Numerical Methods in Fluids*, **70(11)**, 1405–1427.

[76] **Foysi, H. and Sarkar, S.**, (2010). The compressible mixing layer: an LES study, *Theoretical and Computational Fluid Dynamics*, **24**, 565–588.

[77] **Rayleigh, J.W.S.**, (1883). Investigation of the Character of the Equilibrium of an Incompressible Heavy Fluid of Variable Density, *Proc. London Math. Soc.*, **14**, 170–177.

[78] **Taylor, G.I.**, (1950). The Instability of Liquid Surfaces when Accelerated in a Direction Perpendicular to their Planes, *Proc. Roy. Soc. A*, **201**, 192–196.

[79] **Chandrasekhar, S.**, (1961). *Hydrodynamic and Hydromagnetic Stability*, International series of monographs on physics, Clarendon Press.

[80] **Emmons, H.W.**, **Chang, C.T. and Watson, B.C.**, (1960). Taylor instability of finite surface waves, *Journal of Fluid Mechanics*, **7(02)**, 177–193.

[81] **Sharp, D.H. and Wheeler, J.A.**, (1961). Late stage of Rayleigh-Taylor instability, *NASA STI/Recon Technical Report*, **75**, 32399.

[82] **Youngs, D.L.**, (1984). Numerical simulation of turbulent mixing by Rayleigh-Taylor instability, *Physica D: Nonlinear Phenomena*, **12(1–3)**, 32 – 44.

[83] **Dimonte, G. and Schneider, M.**, (2000). Density ratio dependence of Rayleigh–Taylor mixing for sustained and impulsive acceleration histories, *Physics of Fluids*, **(12)**, 304–321.

[84] **Baker, L.**, (1983). Compressible Rayleigh–Taylor instability, *Physics of Fluids*, **26(4)**, 950–952.

[85] **Bernstein, I.B. and Book, D.L.**, (1983). Effect of compressibility on the Rayleigh–Taylor instability, *Physics of Fluids*, **26(2)**, 453–458.

[86] **Li, X.L.**, (1993). Study of three-dimensional Rayleigh–Taylor instability in compressible fluids through level set method and parallel computation, *Physics of Fluids A: Fluid Dynamics*, **5(8)**, 1904–1913.

[87] **Turner, L.**, (2002). Rayleigh–Taylor instabilities and gravity waves in compressible fluids, Technical ReportLA-UR-02-6439, Los Alamos National Laboratory.

[88] **Livescu, D.**, (2004). Compressibility effects on the Rayleigh–Taylor instability growth between immiscible fluids, *Physics of Fluids*, **16(1)**, 118–127.

[89] **Dimonte, G.**, **D.L., Y. and et. al.**, (2004). A comparative study of the turbulent Rayleigh–Taylor instability using high–resolution three–dimensional numerical simulations: The Alpha–Group collaboration, *Physics of Fluids*, **16**, 1668–1693.

[90] **Cook, A.W. and Dimotakis, P.E.**, (2001). Transition stages of Rayleigh–Taylor instability between miscible fluids, *Journal of Fluid Mechanics*, **443**, 69–99.

[91] **Jun, B.I.**, **Norman, M.L. and Stone, J.M.**, (1995). A Numerical Study of Rayleigh-Taylor Instability in Magnetic Fluids, *Astrophysical Journal*, **453**, 332.

[92] **Stone, J. and Gardiner, T.**, (2007). Nonlinear evolution of the magnetohydrodynamic Rayleigh-Taylor instability, *Physics of Fluids*, **19(9)**, 094104.

[93] **Read, K.**, (1984). Experimental investigation of turbulent mixing by Rayleigh-Taylor instability, *Physica D: Nonlinear Phenomena*, **12(1–3)**, 45 – 58.

[94] **Burton, G.C.**, (2011). Study of ultrahigh Atwood-number Rayleigh–Taylor mixing dynamics using the nonlinear large-eddy simulation method, *Physics of Fluids*, **23(4)**, 045106.

[95] **McAllister, S.**, **Chen, J.Y.**, **Fernandez-Pello and Carlos, A.**, (2011). *Fundamentals of Combustion Processes*, Springer.

[96] **Livescu, D. and Ristorcelli, J.R.**, (2008). Variable-density mixing in buoyancy-driven turbulence, *Journal of Fluid Mechanics*, **605**, 145–180.

[97] **Youngs, D.**, (1991). Three-dimensional numerical simulation of turbulent mixing by Rayleigh–Taylor instability, *Physics of Fluids*, **A3**, 1312–1320.

[98] **Cook, A.W.**, **Cabot, W. and Miller, P.**, (2004). The mixing transition in Rayleigh–Taylor instability, *Journal of Fluid Mechanics*, **511**, 333–362.

[99] **Cook, A.W. and Zhou, Y.**, (2002). Energy transfer in Rayleigh-Taylor instability, *Phys. Rev. E*, **66**, 026312.

[100] **Dimotakis, P.**, (2000). The mixing transition in turbulent flows, *Journal of Fluid Mechanics*, **409**, 69–98.

[101] **Ramaprabhu, P.**, 2003. On the dynamics of Rayleigh–Taylor mixing, Ph.D. thesis, Texas A&M University.

[102] **Drake, R.**, (2012). Spike penetration in blast–wave–driven instabilities, *Astrophysical Journal*, **744**, 184.

[103] **Gardner, C.**, (1985). Compressible Rayleigh–Taylor instability of supersonic accelerated surfaces, Technical ReportDOE/ER/03077-264, U.S. Department of Energy.

[104] **Stone, J. and Gardiner, T.**, (2007). The magnetic Rayleigh–Taylor instability in three dimensions, *Astrophysical Journal*, **671**, 1726.

[105] **Saad, Y.**, (1994), *SPARSKIT: a basic tool kit for sparse matrix computations - Version 2*.

[106] **Klein, R.**, (1995). Semi-implicit extension of a godunov-type scheme based on low mach number asymptotics I: One-dimensional flow, *Journal of Computational Physics*, **121(2)**, 213 – 237.

[107] **Munz, C.**, **Roller, S.**, **Klein, R. and Geratz, K.**, (2003). The extension of incompressible flow solvers to the weakly compressible regime, *Computers and Fluids*, **32(2)**, 173 – 196.

[108] **Thornber, B.**, **Mosedale, A.**, **Drikakis, D.**, **Youngs, D. and Williams, R.**, (2008). An improved reconstruction method for compressible flows with low Mach number features, *Journal of Computational Physics*, **227(10)**, 4873 – 4894.

[109] **Yee, H.C.**, **Sandham, N.D. and Djomehri, M.J.**, (1999). Low-dissipative high-order shock-capturing methods using characteristic-based filters, *J. Comput. Phys.*, **150(1)**, 199–238.

[110] **Ghia, U.**, **Ghia, K.N. and Shin, C.T.**, (1982). High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *Journal of Computational Physics*, **48**, 387–411.

[111] **Erturk, E.**, (2009). Discussions on driven cavity flow, *International Journal for Numerical Methods in Fluids*, **60**, 275–294.

[112] **Xiao, F.**, **Akoh, R. and Li, S.**, (2006). Unified formulation for compressible and incompressible flows by using multi-integrated moments II: Multi-dimensional version for compressible and incompressible flows, *Journal of Computational Physics*, **213(1)**, 31–56.

[113] **Rizzi, A. and Viviand, H.**, (1981). *Numerical methods for the computation of inviscid transonic flows with shock waves : a GAMM workshop*, Vieweg, Braunschweig :.

[114] **Bijl, H. and Wesseling, P.**, (1998). A Unified Method for Computing Incompressible and Compressible Flows in Boundary-Fitted Coordinates, *Journal of Computational Physics*, **141(2)**, 153 – 173.

[115] **Eidelman, S.**, **Colella, P. and Shreeve, R.P.**, (1984). Application of the Godunov method and its second-order extension to cascade flow modeling, *AIAA Journal*, **22**, 1609–1615.

[116] **Jameson, A.**, **Schmidt, W. and Turkel, E.**, editors, (1981). Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes.

[117] ***AGARD-AR-211***, (1985). Test Cases for Inviscid Flow Fields, Technical Report.

[118] **Mesaros, L.**, (1995). Multi-dimensional Fluctuation Splitting Schemes for the Euler Equations on Unstructured Grids, Ph.D. thesis, University of Michigan.

[119] **Paillere, H.**, (1995). Multi-dimensional Upwind Residual Distribution Schemes for the Euler and Navier-Stokes Equations on Unstructured Grids, Ph.D. thesis, Universite Libre de Bruxells.

[120] **Wenneker, I.**, (2002). Computation of Flows Using Unstructured Staggered Grids, Ph.D. thesis, Technical University of Delft.

[121] **Nejat, A.**, (2007). A Higher-Order Accurate Unstructured Finite Volume Newton-Krylov Algorithm for Inviscid Compressible Flows, Ph.D. thesis, University of British Columbia.

[122] **Eraslan, E.**, (2006). Implementation of Different Flux Evaluation Schemes into a Two-dimensional Euler Solver., Ph.D. thesis, Middle East Technical University.

APPENDICES

APPENDIX A : Some Additional Test Cases for Validation and Verification

APPENDIX A : Some Additional Test Cases for Validation and Verification

This appendix includes some other test cases for verification of the serial implementation of the algorithm that uses Saad's SPARSKIT [105] library for the solving linear systems arising from the discretization. Here, it is also aimed to assess the behavior of the algorithm by applying it to the problems including various physical mechanisms.

These problems were selected regarding their easy implementation and availability of their exact or well-known results.

The first test case is the *inviscid acoustic wave propagation* or *density layering* in 1D [106–108]. This used to validate that the scheme can advect large density variations at $M = 0.02$ at very low dissipation, whilst resolving a low wave-length acoustic wave which passes through the density layer. The initial conditions are defined by,

$$\rho(x,0) = 1.0 + 0.5\phi(x)\sin\left(\frac{40\pi x}{L}\right) + M\left(1.0 + \cos\left(\frac{\pi x}{L}\right)\right) \tag{\textbf{A.1}a}$$

$$u(x,0) = 1.0 + \sqrt{\gamma}\left(1.0 + \cos\left(\frac{\pi x}{L}\right)\right) \tag{\textbf{A.1}b}$$

$$p(x,0) = 1.0 + M\gamma\left(1.0 + \cos\left(\frac{\pi x}{L}\right)\right) \tag{\textbf{A.1}c}$$

$$for - L \leq x \leq L, \quad with \quad \phi(x) = \begin{cases} 0 & for \quad -\frac{1}{L} \leq x < 0 \\ \frac{1}{2}\left(1.0 - \cos\left(\frac{5\pi x}{L}\right)\right) & for \quad 0 \leq x \leq \frac{2L}{5} \\ 0 & for \quad x > \frac{2L}{5} \end{cases}$$

and the boundary conditions are defined to be periodic. These data generate an acoustic wave of wavelength L propagating to the right. It is overlaid by short wavelength density oscillations which are restricted to a region with length $2L/5$ by the function ϕ. For the Mach number we prescribed the value $M = 1/51$ and the length of the computational domain has been chosen in such a way that one complete acoustic wave is accommodated: $L = 1/M$. In our calculations a fine grid of 1020 grid points is used to resolve the short wavelength density fluctuations which are set in motion by the large wavelength acoustic wave. The problem becomes much more difficult if short wave-length density oscillations are over-imposed on the acoustic wave. The density fluctuations are set into motion by the long wavelength acoustic pulse.

Fig.A.1 shows the density, the velocity and the pressure results respectively, in comparison with the results taken from references [106–108] The initial values are plotted by the dashed lines and the numerical results by the straight lines. The calculation time is chosen in such a way that the acoustic wave has passed the computational domain about two and half times and is steepened due to non-linear effects.

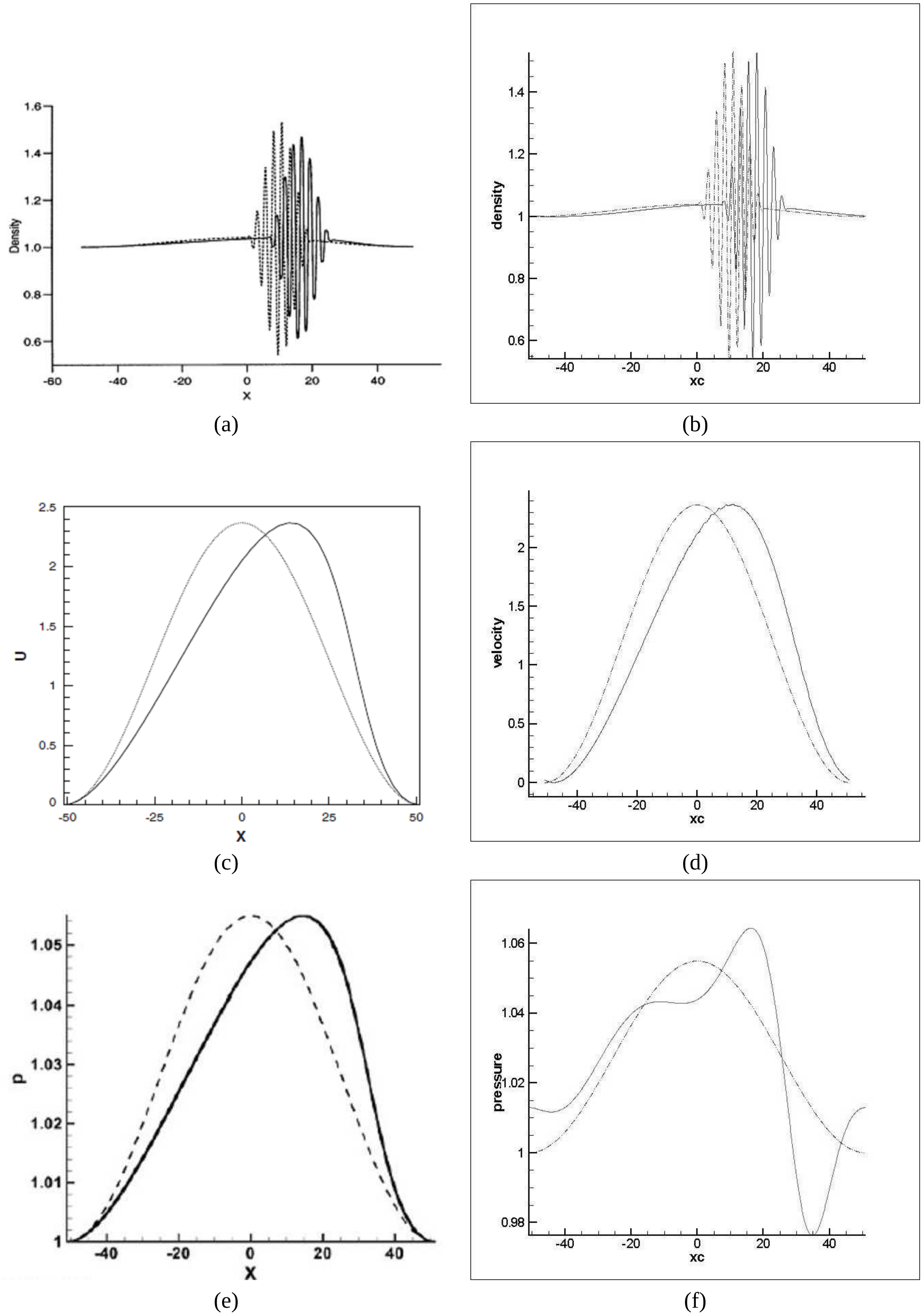


Figure A.1: The comparison of the density (top line), the velocity (middle line) and the pressure (bottom line) results. While the left column shows the reference results, our results are presented at the right column.

The key aspect of this example is, however, the advection of the density profile due to the recurring interaction with the acoustic wave. The theoretical prediction is that the density fluctuation amplitude should be preserved in the process and that only small amplitude long wave variations should occur due to the adiabatic compression as acoustic wave passes by. Besides this, density layering should just move to the right, driven by cumulative influence of the acoustic pulses. This is well captured in the numerical calculations. The amplitude of the density fluctuations is kept very well. No numerical dissipation seen in the convection step.

Another aspect of this example is that, at the location of the density fluctuations, pressure variations occur when acoustic wave passes by. These are accompanied by first-order fluctuations of the velocity. These small amplitude pressure variations are more clearly visible.

Notice that, density fluctuation amplitude is preserved better than the reference. However, pressure is somewhat more affected from the short wavelength density oscillations than expected. A very small time-step or a much more finer grid may be required to capture the pressure oscillations.

The second test case is the *isentropic vortex evolution* in 2D, both *steady* and *horizontally moving*. These two sub-cases were used to evaluate to the implementation's accuracy and the algorithm's ability to preserve the vortex in a free-stream [109].

Following [109], the initial free-stream density, pressure and temperature is set to unity for both cases. The initial free-stream velocities, u_∞ and v_∞ are $(0,0)$ for the steady case and $(1,0)$ for the horizontally convecting case.

The following perturbations are added to the corresponding flow variables,

$$(\delta u, \delta v) = \frac{\beta}{2\pi} e^{(1-r^2)/2} (-\bar{y}, \bar{x}) \qquad (\textbf{A.2}\text{a})$$

$$\delta T = -\frac{(\gamma-1)\beta^2}{8\gamma\pi^2} e^{(1-r^2)} \qquad (\textbf{A.2}\text{b})$$

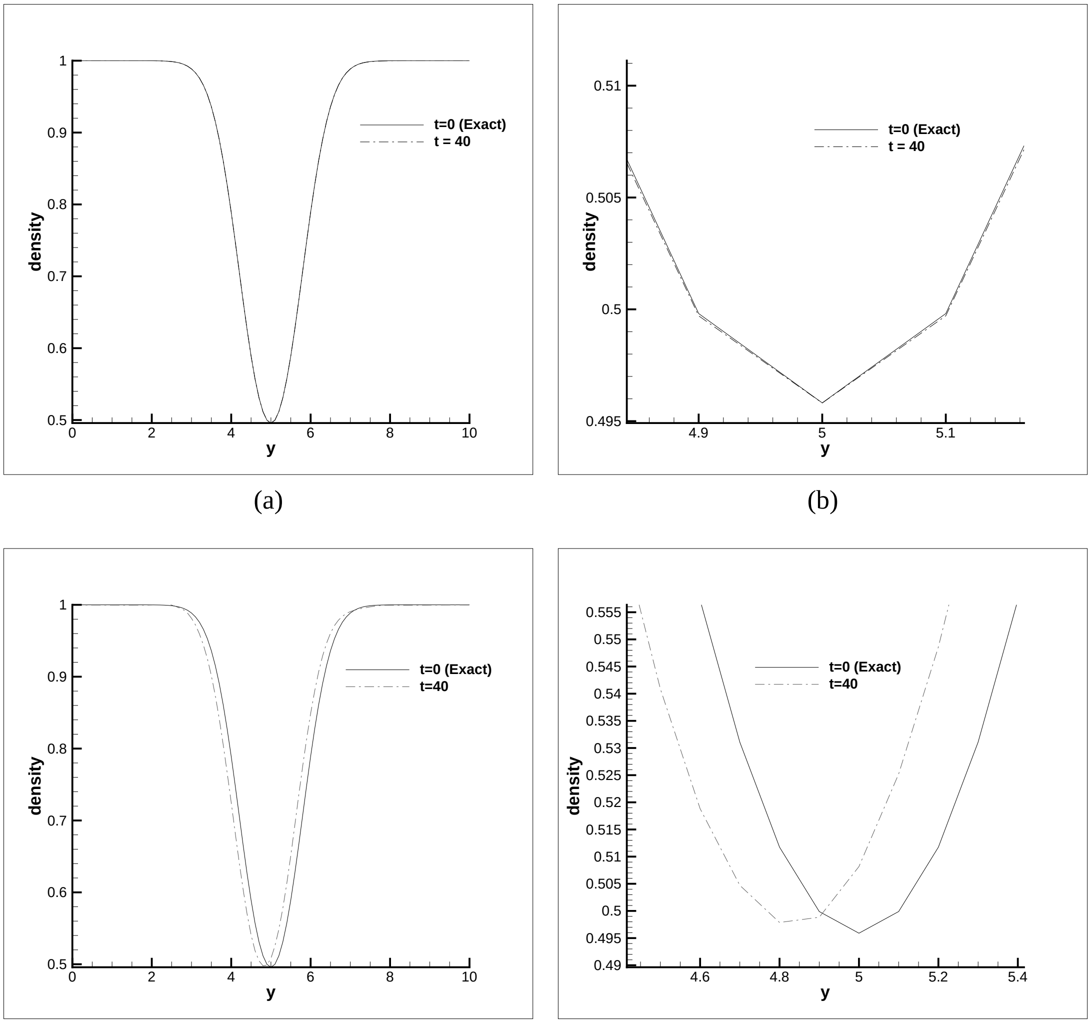
where β is the vortex strength. T and $\bar{x}, \bar{y}$ are calculated as $\frac{p}{\rho}$ and $(x - x_{v0}, y - y_{v0}$ respectively. $(x_{v0}, y_{v0}$ are the coordinates of the center of vortex at initial and $r^2 = \bar{x}^2 + \bar{y}^2$. For isentropic flow field, $\frac{p}{\rho^\gamma} = 1$. After adding perturbations, the initial flow field (also exact solutions) takes the following form,

$$\rho = T^{1/(\gamma-1)} = (T_\infty + \delta T)^{1/(\gamma-1)} = \left[1 - \frac{(\gamma-1)\beta^2}{8\gamma\pi}\right]^{1/(\gamma-1)} \qquad (\textbf{A.3}\text{a})$$

$$\rho u = \rho(u_\infty + \delta u) = \rho \left[1 - \frac{\beta}{2\pi} e^{(1-r^2)/2}\right] \qquad (\textbf{A.3}\text{b})$$

$$\rho v = \rho(v_\infty + \delta v) = \rho \left[1 + \frac{\beta}{2\pi} e^{(1-r^2)/2}\right] \qquad (\textbf{A.3}\text{c})$$

$$p = \rho^\gamma \qquad (\textbf{A.3}\text{d})$$



(a) (b) (c) (d)

Figure A.2: The comparison of the results with the exact solution for density. Top line is the steady case and the bottom line is the horizontally convecting case.

Since there is no shocks or turbulence in the domain, this problem serves as a tool for measuring the accuracy of a scheme. All test cases employ a uniform cartesian grid. Density profiles at the x-centerline, $x = 5$, cutting through the center of the vortex are used for comparison. The domain length is 10 for each direction. Gird resolution is 100×100 with a grid spacing 0.1. Time step is 0.01. $(x_{v0} = 5, y_{v0} = 5)$ are the coordinates of the center of vortex at initial. In the case of the horizontally moving vortex, the vortex is convected to the right by the mean flow velocity.

All numerical solutions are very accurate and fall almost on the top of the exact solution. A slight displacement (i.e., phase-lagging) is seen on Fig.A.2. We also performed our simulations on 32*x*32, 64*x*64 and 100*x*100 grids. They show that, this looks like a grid-dependent behavior. Since, the displacement is decreasing with an increasing grid resolution. Another aspect of this slight displacement is global order of accuracy. Our algorithm is second-order in space. This is not an issue, for example, for a fourth or sixth-order algorithm [109] with a filter. There is an obvious superiority of very high-order algorithms in long-time simulations over lower-order ones.

The *unsteady lid-driven cavity flow* [110] is the third test case considered here. It is one of the most studied problem of CFD to test new computational schemes under low-Mach number or incompressible viscous flow conditions. Unsteady lid-driven cavity flow includes a main vortex at the center of the domain and multiple counter-rotating recirculating regions on the corners of the cavity depending on the Reynolds number. Since detailed numerical and experimental results are available, it serves as an ideal framework for algorithms and implementations. Also, a wide range of Reynolds numbers can also be studied with a fixed domain, as an advantage. Fig. A.3 shows the schematic representation of the problem and the boundary conditions.

Three different sub-cases were included to study the effects of grid resolution, Reynolds numbers and Mach number. The unit cavity, $aspect\,ratio(depth/width) = 1$, is used for all cases. The equation of state of ideal gas in non-dimensional form. The initial non-dimensional density is unity for all cases, and the initial non-dimensional velocities are zero in everywhere (except lid). The initial non-dimensional pressure field is also zero inside domain. The Reynolds number is based on the cavity height and the kinematic viscosity.

Note that, at a wall, no-slip boundary condition applies, i.e. the velocity of the fluid is equal to wall velocity, a Dirichlet boundary condition. However, there is another condition that can be directly imposed in a finite-volume method; the normal viscous stress is zero at wall. This follows from the continuity equation, e.g. for a wall at $y = 0$, and it should be implemented directly, rather than using only the condition that $v = 0$ at the wall. For details see [2].

For the grid resolution effect, $20\times20, 40\times40, 60\times60, 80\times80$ and 100×100 uniform grids were used. Computations were performed for the Reynolds number 1000 and for the Mach number 0.001 for all grids. The non-dimensional time-step was 0.01 for the grids $20\times20, 40\times40, 60\times60$ and 0.005 for the grids $80\times80, 100\times100$. Flow reaches steady-state around $t \approx 30$. Note that this time-steps correspond to acoustic CFL numbers, c_0dt/dx, from 200 to 600 for different grids and convective CFL numbers, udt/dx, from 0.2 to 0.6 in terms of the driving velocity of the upper lid $(u = 1)$ for different grids. As it is shown from the Fig.A.4, The results are getting closer to the reference results, and the finer grid computation (i.e., 100×100) is in good agreement with Ghia [110].

To investigate the Reynolds number effect, 1000, 5000 and 10000 were used. All computations were performed on 40×40 uniform grid and with the Mach number 0.001. The non-dimensional time-step is 0.01. The Reynolds number was controlled by modifying the kinematic viscosity. Note that, this time-steps corresponds to an acoustic CFL number, c_0dt/dx, 400 and a convective CFL number, udt/dx, 0.4 in terms of the driving velocity of the upper lid $(u = 1)$. The results were represented in Fig.A.5. It is observed that time for reaching steady-state is increasing with the increasing Reynolds number. The primary vortex grows both in size and in strength (i.e.,getting closer to the walls), as found in the literature. It can also be seen from the slightly wavy-shape at high-Re numbers that finer grid is needed to capture the flow physics correctly at high-Re numbers, because of the starting of the transition to turbulence due to Hopf bifurcations [111]. See also page 12 of the same reference for discussions on the effect of cell Reynolds number in obtaining steady-state solutions.

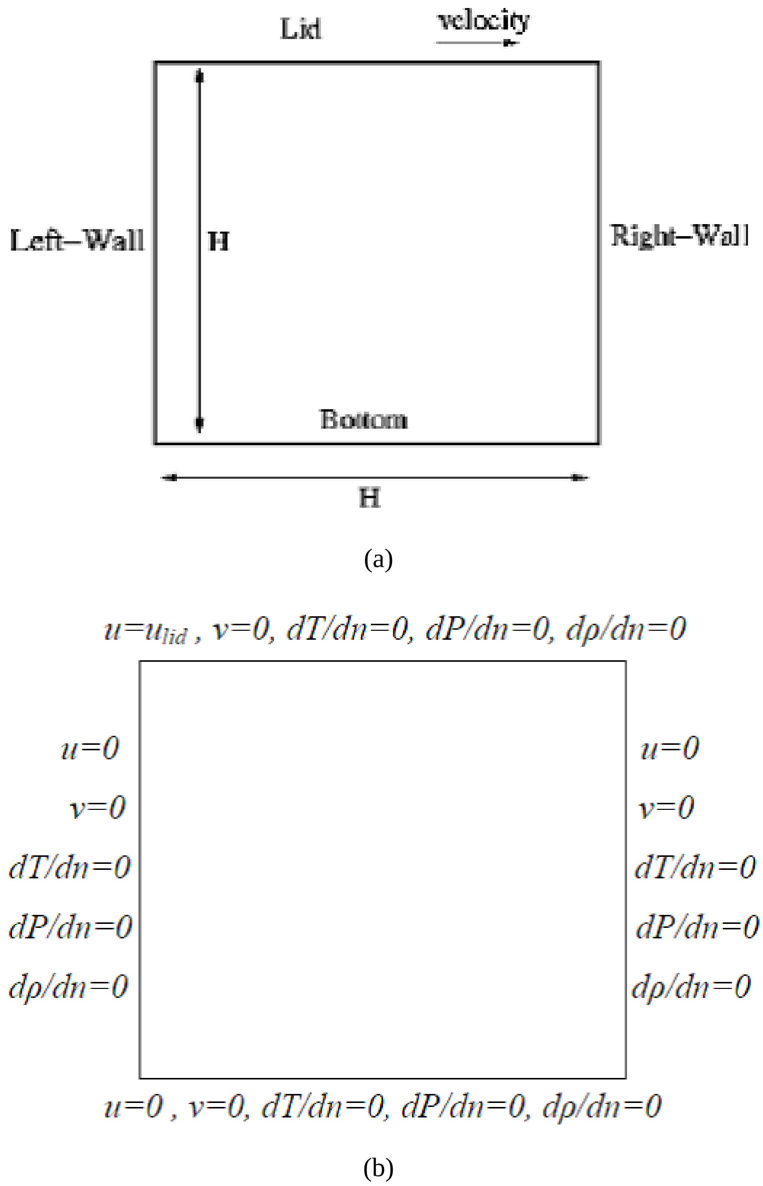
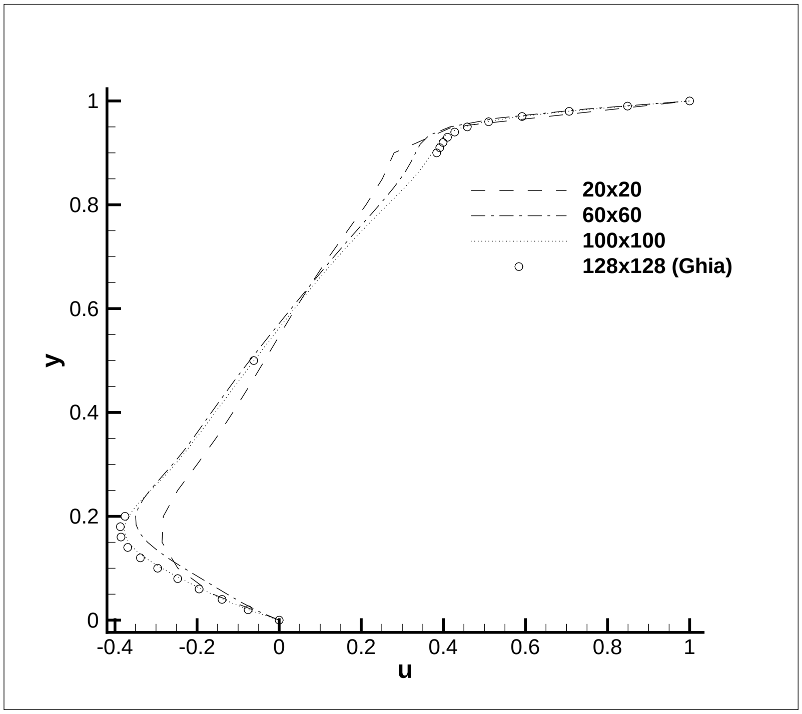
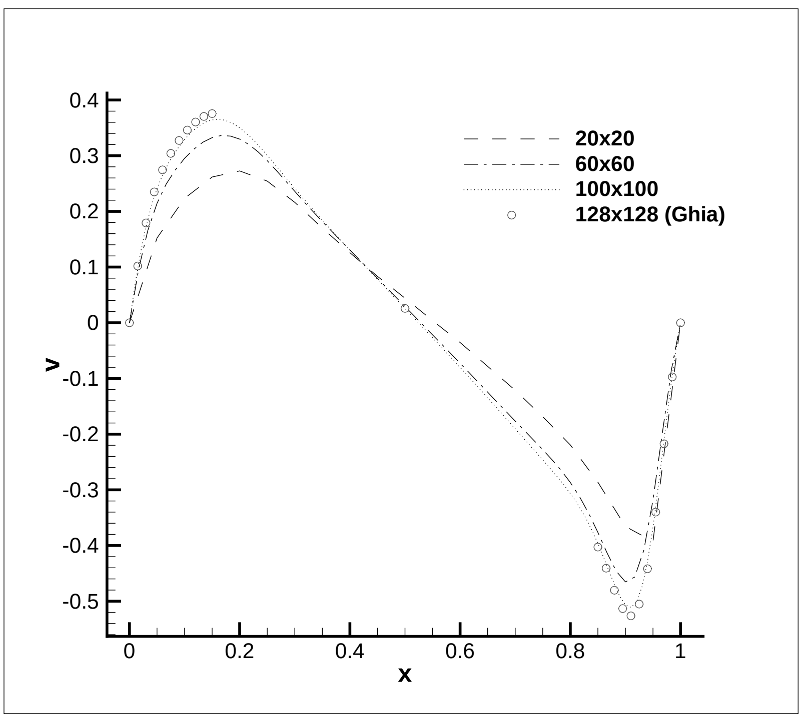


Figure A.3: Schematic representation of cavity flow problem and boundary conditions.

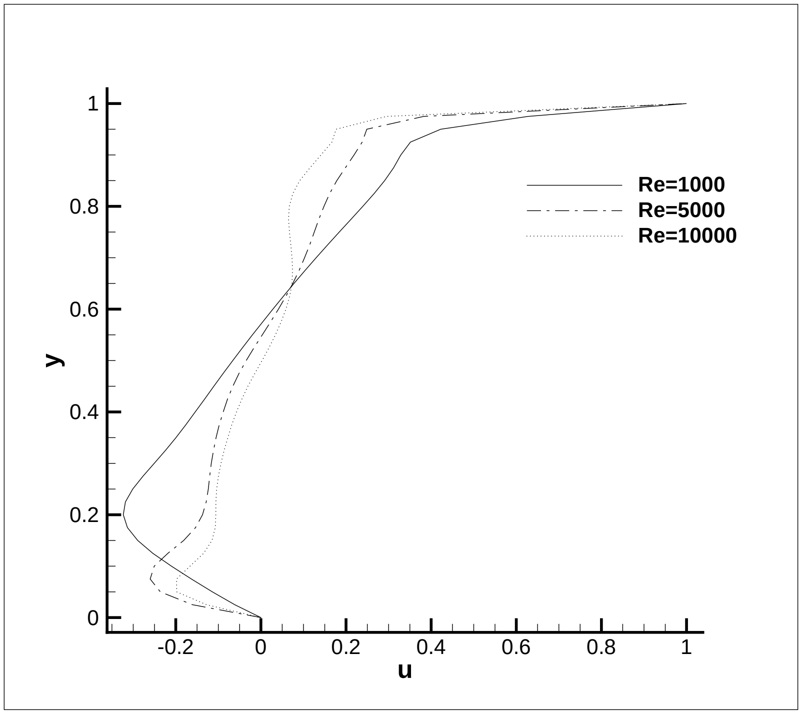


(a)

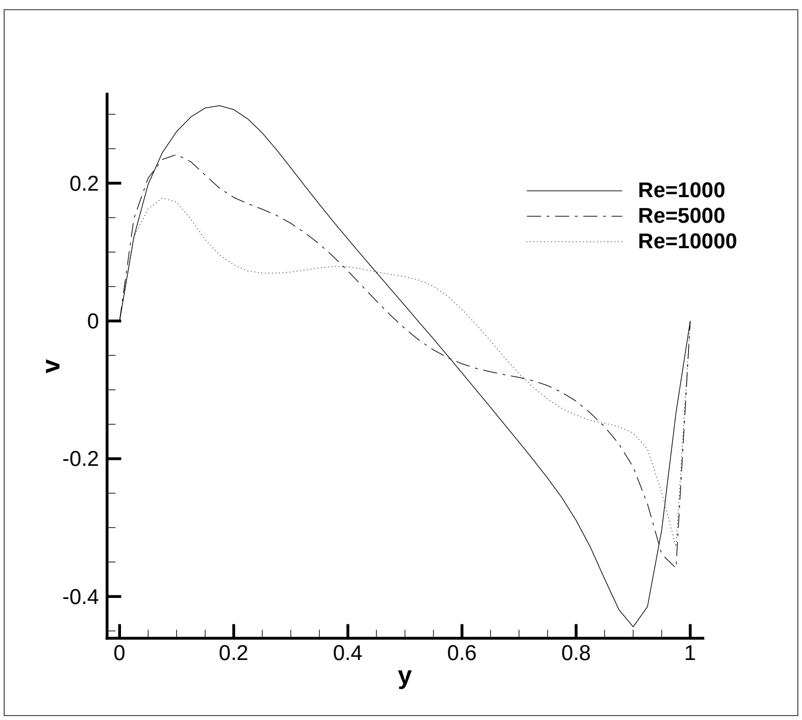


(b)

Figure A.4: The effect of the grid resolution on results.



(a)



(b)

Figure A.5: The effect of Reynolds number on results.

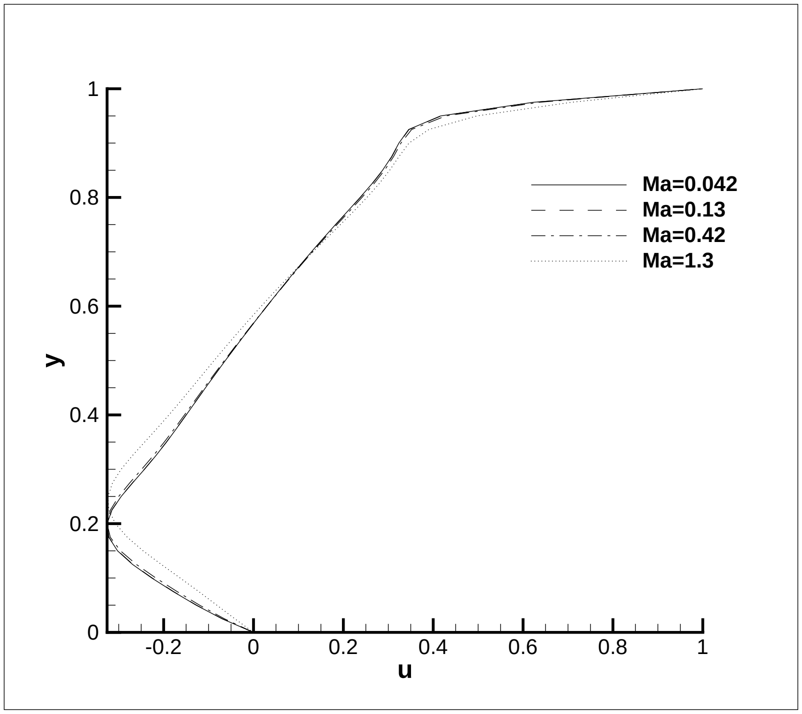
It was also interesting to see how the compressibility affects the development of the cavity flow. We carried out a series numerical experiments for this purpose. Four different Mach number were used in this case; $0.042, 0.13, 0.42$ and 1.3. The compressibility of the flow was controlled by choosing different pressures. All numerical details are same with Xiao *et. al.* [112] for comparison. All computations were done on 40×40 uniform grid with the Reynolds number 1000. The non-dimensional time-step was taken 0.00875. Note that this time-steps correspond to acoustic CFL numbers, $c_0 dt/dx$, $0.7, 2.4, 7.7$ and 23.8 for different Mach numbers and a convective CFL number,udt/dx, 0.35 in terms of the driving velocity of the upper lid $(u = 1)$. The results at $t = 25$ were represented in Fig.A.6.

Compressibility causes an oscillation in the strength and the location of the primary vortex. $M = 0.42$ has less oscillations than $M = 1.3$ in the development of the vortex, while the vortex center moves. It is observed that amplitude of the oscillations increases and frequency of oscillations decreases, while Mach number increases. In the low Mach or less compressible cases, the primary vortex is monotonically strengthened and moves to its final position in a much stable way. $M = 1.3$ case takes very long time $(t \approx 70-80)$ to reach steady-state due to the primary vortex oscillations. Even after this period, convergence still was not fully achieved. Similar observations have been done by Xiao *et. al.* [112], together with a stronger effect of an increasing Mach number. The subsonic and the incompressible cases do not make noticeable difference.

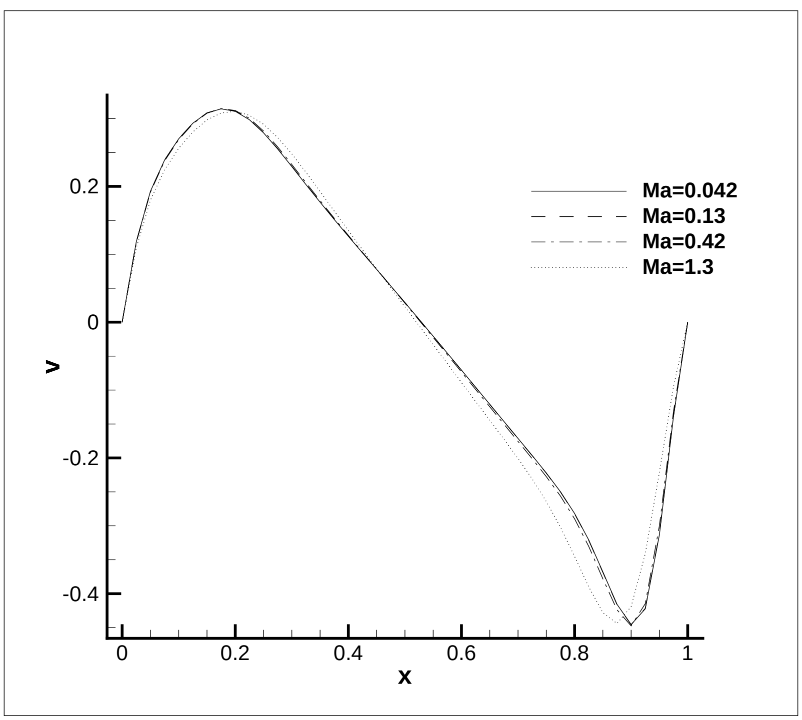
The <u>fourth test case</u> is the *inviscid, laminar flow inside channel with bump*. This is the standart test case proposed at GAMM conference [113] and used to check the accuracy and the stability of their numerical method by many authors. The size of the channel is $[-1, 2] \times [0, 1]$, in which a boundary-fitted non-uniform grids of 96×32 cells for subsonic and transonic cases and 120×40 cells for supersonic case are generated (See Fig. A.7).

The algorithm was also extended/modified to handle the boundary-fitted grids and implemented into the code. Thickness-to-chord ratio is 10 percent for subsonic/transonic calculations and 4 percent for supersonic calculations. The initial conditions are distributed to domain as appropriate to the conditions at inlet. All quantities are non-dimensionalized by the inlet conditions, except for the pressure. For the pressure, static pressure at outlet is chosen as reference pressure. Characteristic boundary conditions are used for all boundaries. Top surface is chosen as symmetry where both gradient normal to the boundary of a scalar quantity and tangential velocity must be vanish, as well as normal velocity along the boundary. Bottom surface is taken as inviscid (slip) wall where tangential velocities are allowed and zero mass flux are prescribed. Subsonic inflow and subsonic outflow are used for the inlet and outlet of the channel. At inlet, upstream far-field variables, except for the pressure, are specified. At outlet, all variables are extrapolated, except for the pressure.

The subsonic flow was computed first. The Mach number is 0.5 at inlet. The Mach number distribution along the walls and in the whole domain are presented in Fig.A.8. Since there are no shocks, a symmetry in the solution is observed as it must be for subsonic flow through the specified channel. This result is comparable to others that can be found in the literature [1, 2, 114, 115]. No oscillations are observed around

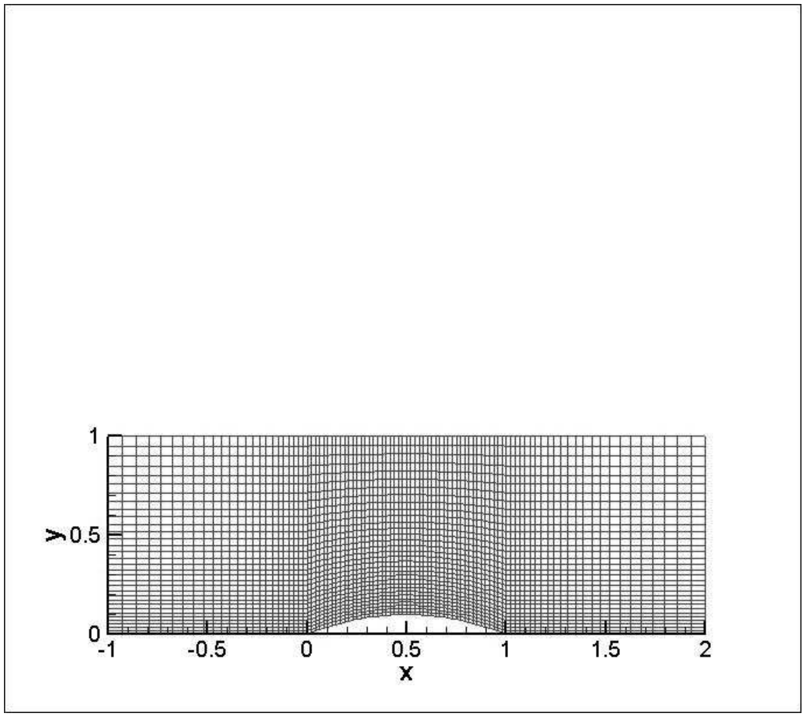


(a)

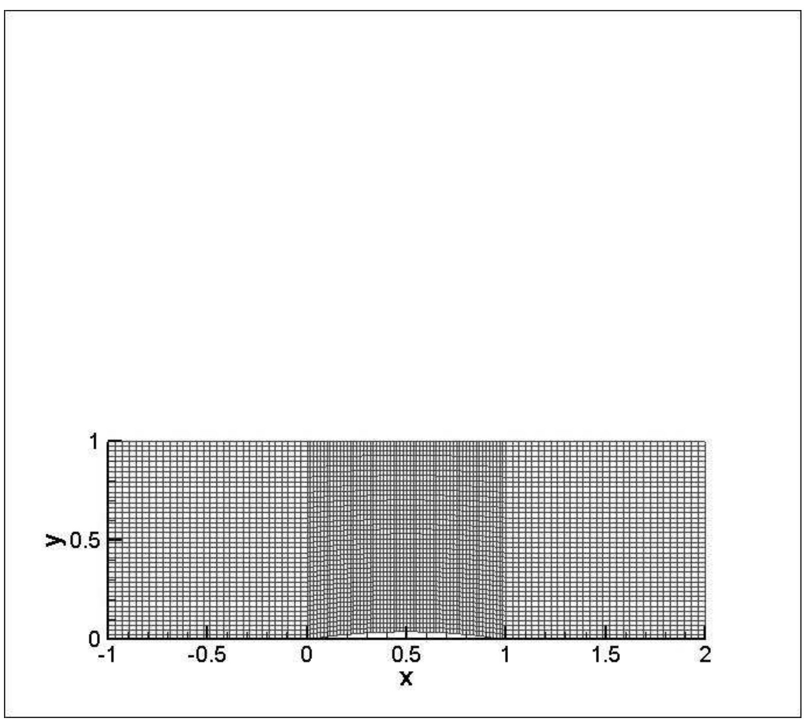


(b)

Figure A.6: The effect of the Mach number on results.



(a) subsonic/transonic grid



(b) supersonic grid

Figure A.7: The grids used in channel flow computations.

the bump where mesh has some ratio of skewness, which shows the accuracy of the boundary-fitted extention.

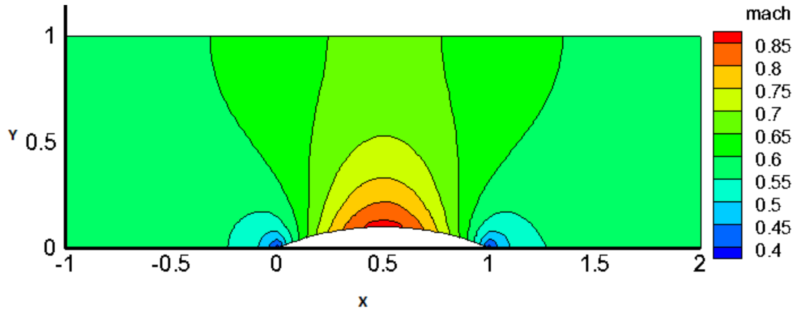
The transonic flow was simulated. The inlet Mach number is 0.675 for the case. Since the method is non-dissipative, a shock-capturing procedure must be added to the algorithm, in order to be able to capture shocks. For this purpose, Jameson-type scalar artificial dissipation mechanism [116] was implemented into the algorithm while calculating explicit fluxes. A shock is formed at 72 percent of the bump and resolved very well by the mechanism (within 2-3 cells), as it is shown in Fig.A.9. The maximum Mach number reported in the literature is around $1.32 - 1.43$. In our simulation, the maximum Mach number is found as 1.33. Again, the results are quite comparable to those available in the literature [2, 114, 115]. No oscillations are observed around the shock.

The supersonic flow was also simulated with the Mach number 1.65 given at inlet. Jameson-type scalar artificial dissipation mechanism is also activated for the case. Shocks are stronger than the previous case. Leading edge shock reflects from top surface. Although, the shocks are relatively well resolved by the algorithm, an amount of smearing (expansion) in the shocks is observed, as it is shown in Fig.A.10. Small wiggles before and after leading edge shock are due to coarse grid resolution. The reflected shock is little bit smeared which is similar to the references. It is obvious that very high-order inherently-dissipative methods give sharper results, as previously presented in the literature. However, the overall results are acceptable and also quite similar to many of the results presented in the literature [2, 114, 115].

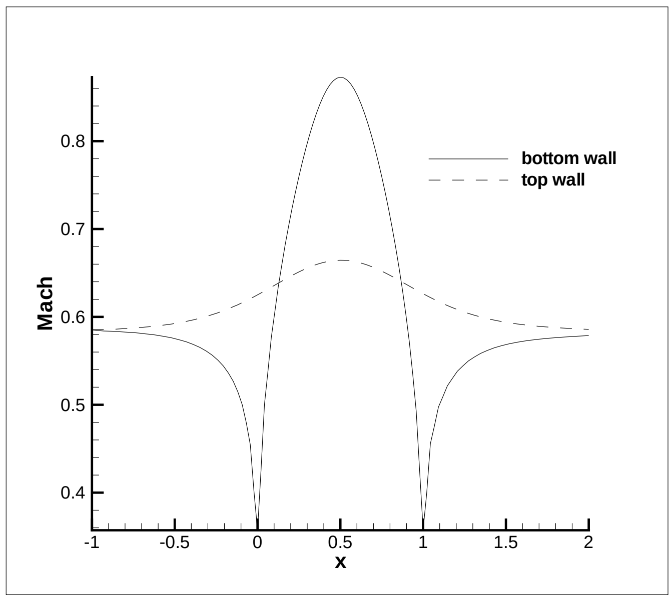
The <u>fifth and the final test case</u> is the *inviscid, laminar flow around NACA0012 airfoil* [117]. Two sub-cases were also introduced; low Mach number (nearly incompressible) [118–120] and subsonic (subcritical) [118–122]. Here, it is aimed to test the performance of the compressible method for the low Mach number conditions, to check the accuracy of the boundary-fitted implementation and to assess the behavior of the Jameson shock-capturing mechanism on the more serious problem. The free-stream values were chosen to the reference values for this problem. A C-type grid with 192×32 was used for all cases, shown in Fig.A.11. The grid has 128 cells around the airfoil. The distance to far-field has 20 cells, and the number of cells in the wake is 32. The cord length was scaled to unity. The problems were simulated until the steady-state was reached.The far-field (i.e., the characteristic) boundary conditions were used with vortex correction. The airfoil surface boundary condition was inviscid slip-wall. The cut boundary condition was also used for the wake region .

For the low Mach number case, the Mach number was 0.1, and the angle of attack was zero degree. The Reynolds number was set to 100. The time step was taken as 1×10^{-4}. Fig.A.12 shows the final Mach number distribution, C_p distribution, and the time evolution of the density residual for the low Mach number case.

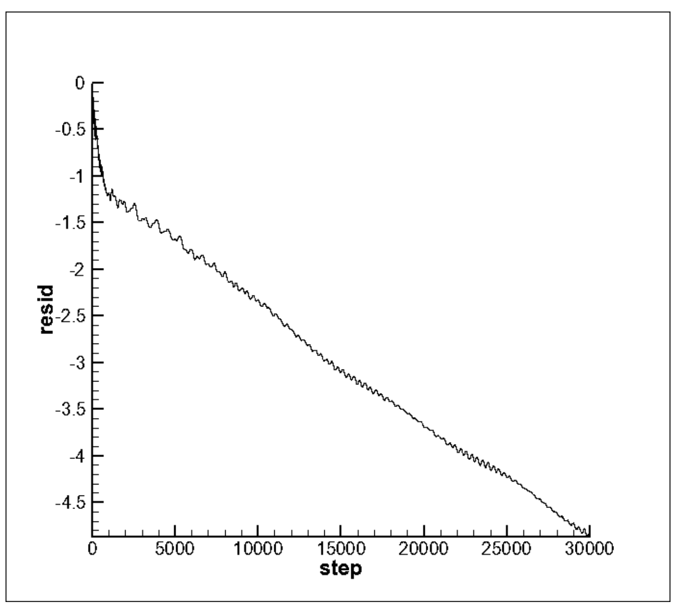
For the subsonic (sub-critical) case, the Mach number was 0.63, and the angle of attack was 2°. The Reynolds number was set to 400. The time step was taken as 1×10^{-3}. Fig.A.13 shows the final Mach number distribution, C_p distribution, and the time evolution of the density residual for the sub-critical case. The density residual does not converge to a specified value. It shows an oscillatory behavior. C_l,C_d and maximum Mach number are found 0.3, 0.01 and 0.88, respectively. The reference



(a)

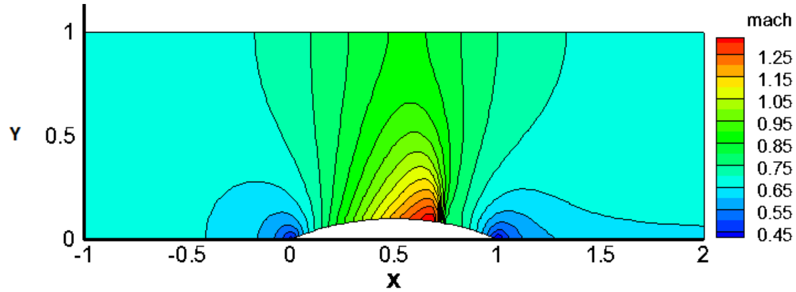


(b)

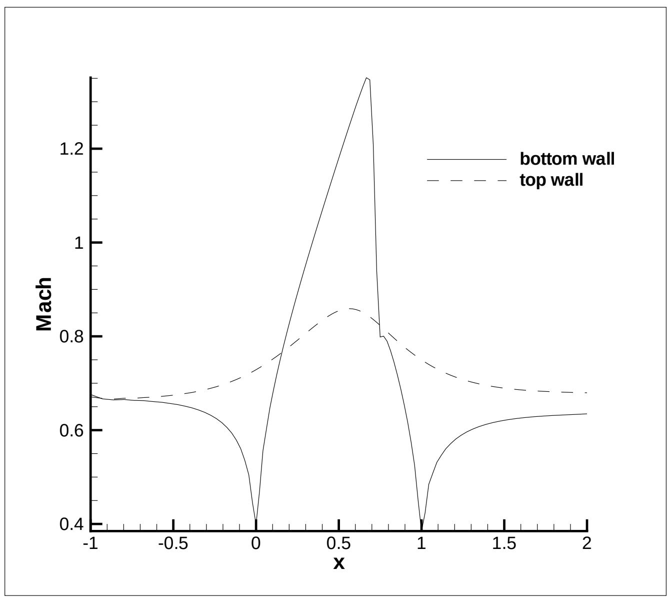


(c)

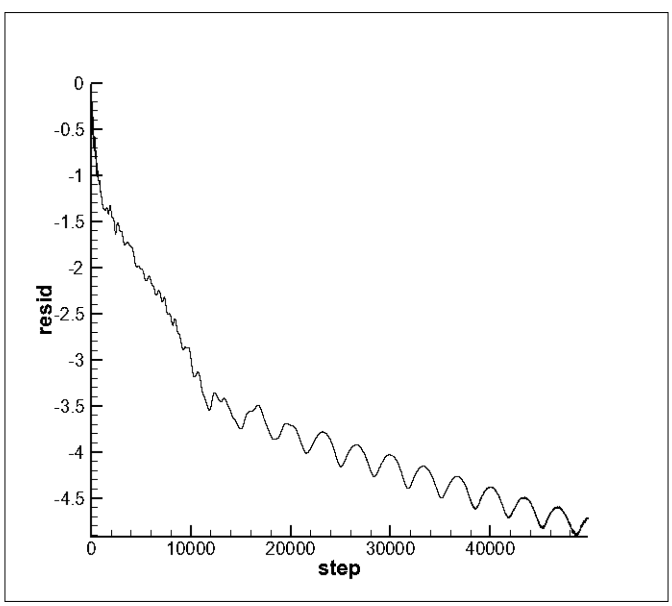
Figure A.8: The subsonic channel flow results. The final Mach distribution (a), the Mach distribution at walls (b), and residual history (c).



(a)

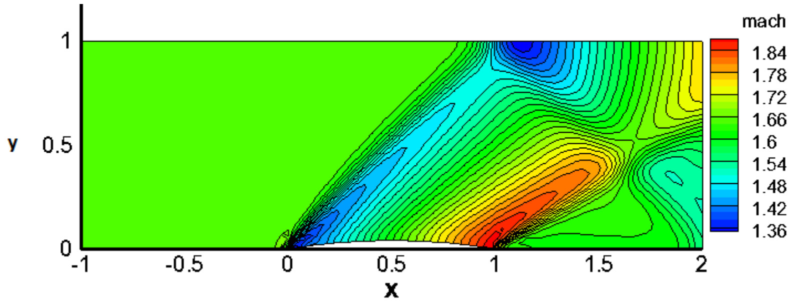


(b)

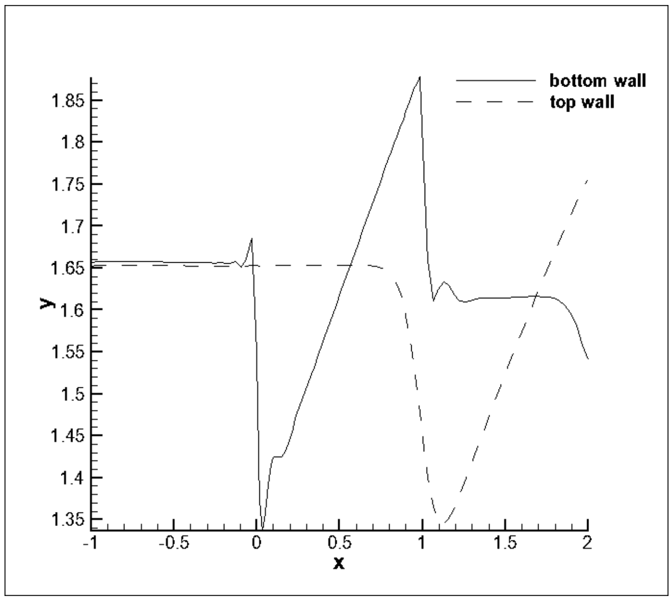


(c)

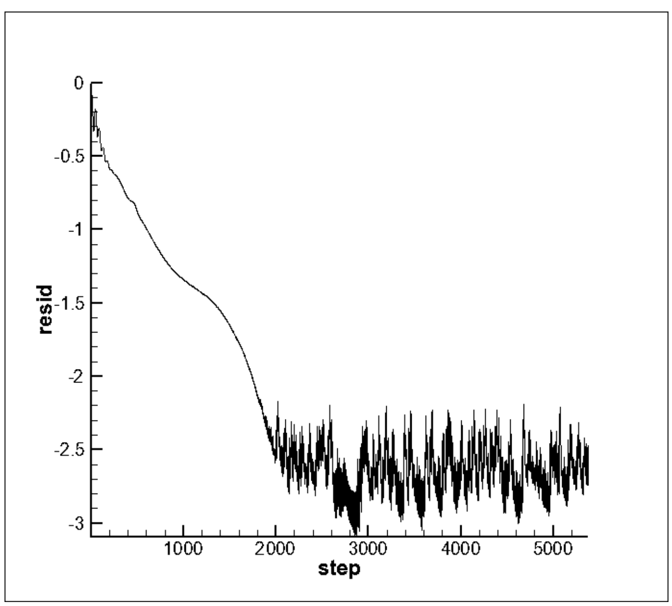
Figure A.9: The transonic channel flow results. The final Mach distribution (a), the Mach distribution at walls (b), and residual history (c).



(a)



(b)



(c)

Figure A.10: The supersonic channel flow results. The final Mach distribution (a), the Mach distribution at walls (b), and residual history (c).

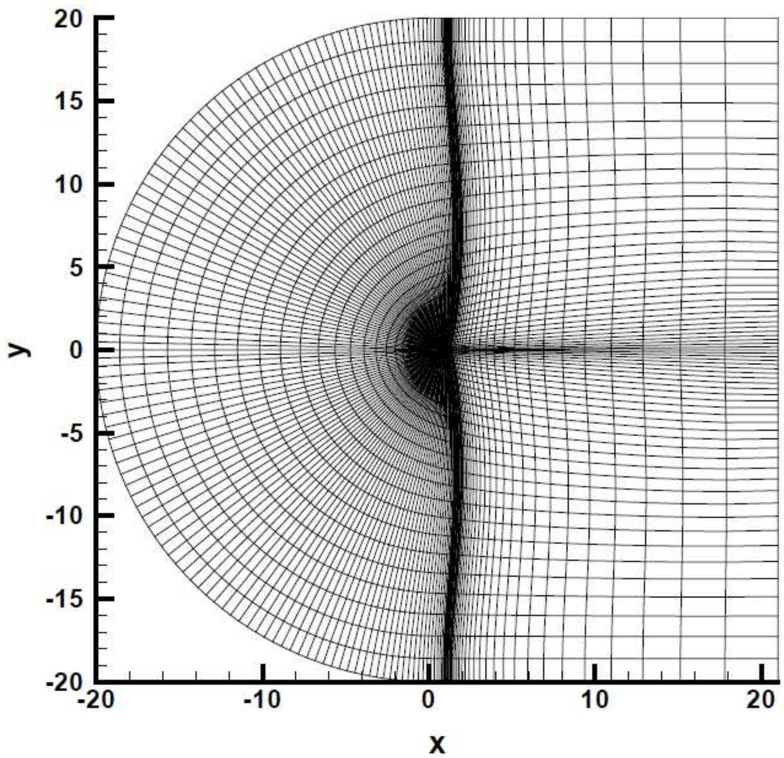
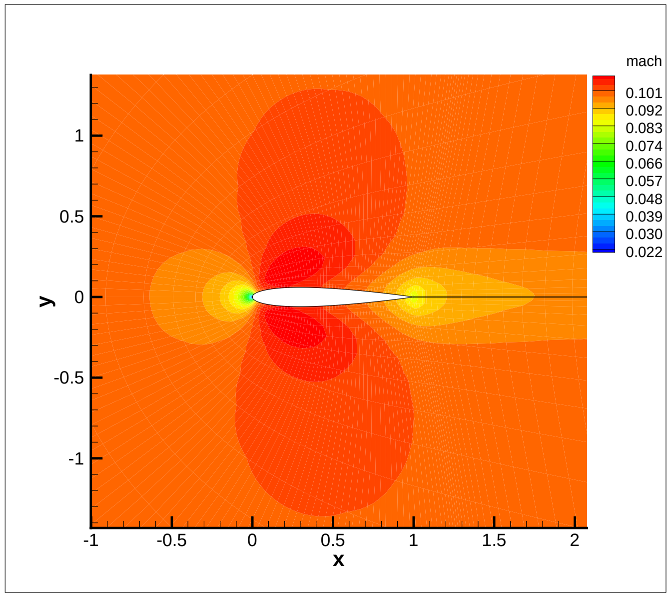


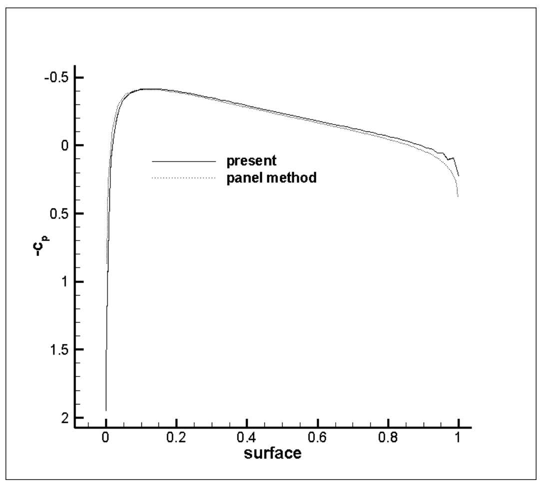
Figure A.11: The C-type grid used in the airfoil test case.

values [117] for this case are 0.33 for C_l, 0 for C_d and 0.99 for maximum Mach number. An increasing loss of accuracy is observed with an increasing Mach number.

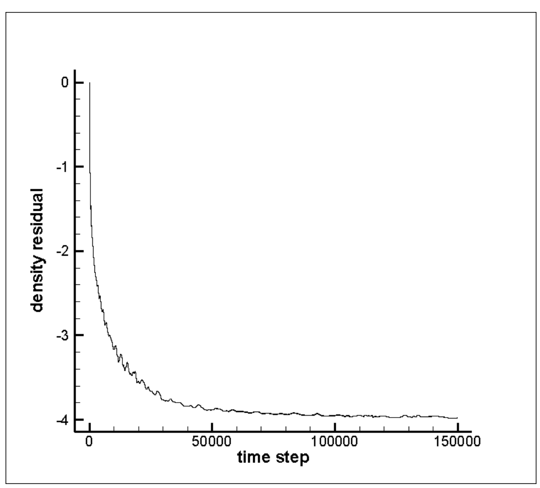
These differences observed in the final test case above could possibly be related to the coarse grid resolution and the loss of global accuracy on boundary-fitted highly skewed grids. An external dissipative mechanism is also required to capture shocks in the domain where Mach number is locally around unity.



(a)

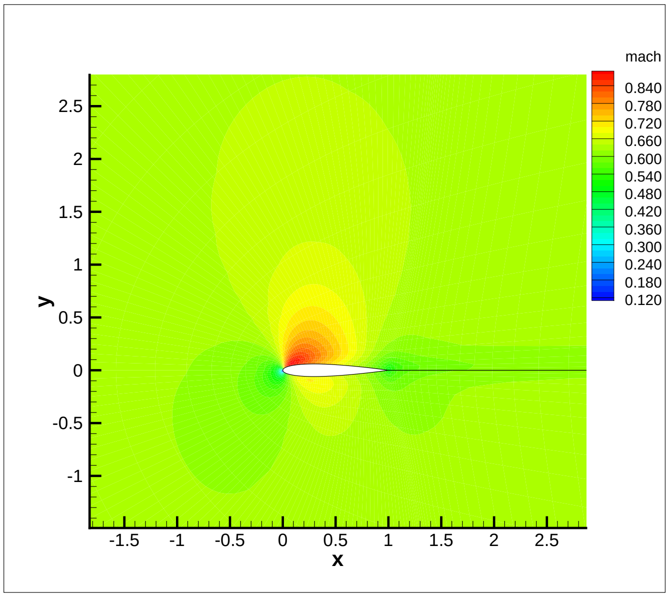


(b)

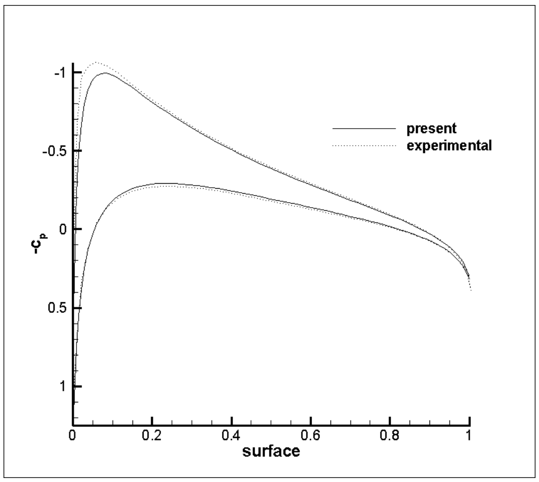


(c)

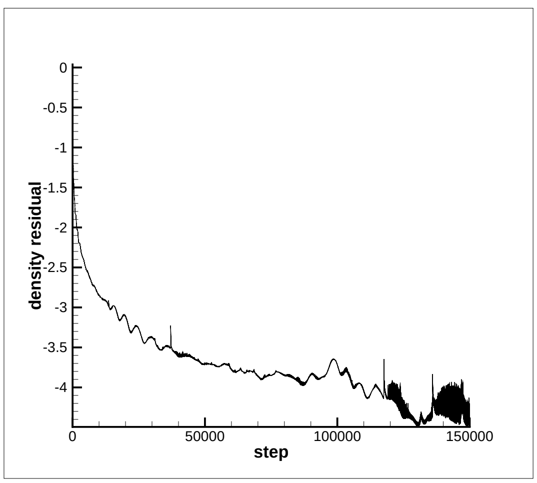
Figure A.12: The low Mach number airfoil case results. The final Mach distribution (a), the C_p distribution on the airfoil surface (b), and the density residual history (c).



(a)



(b)



(c)

Figure A.13: The sub-critical airfoil case results. The final Mach distribution (a), the C_p distribution on the airfoil surface (b), and the density residual history (c).

CURRICULUM VITAE

Name Surname: İlyas YILMAZ

Place and Date of Birth: Safranbolu, 20.07.1974

Adress: ITU Energy Institute Ayazaga Campus Maslak ISTANBUL

E-Mail: yilmazily@itu.edu.tr

B.Sc.: Astronomy and Space Sciences, Istanbul University, 1996

M.Sc.: Nuclear Energy, Istanbul Technical University, 2001

Professional Experience and Rewards:

Research Assistant, ITU Energy Institute , 2000-2001

Research Assistant, ITU Informatics Institute, 2001-2003

Research Assistant, ITU Energy Institute , 2003-2012

ITU Abroad Research Scholarship for PhD Students, September 2009-December 2009, Chalmers University of Technology, Goteborg, Sweden

TUBITAK Abroad Research Scholarship for PhD Students (2214), January 2010 - June 2010, Chalmers University of Technology, Goteborg, Sweden

ECCOMAS Scholarship September 2012

List of Publications and Patents:

- A.B.Tuğrul, **İ. Yılmaz**, B. Ataksor, S. Sahin, "Evaluation of Radioactivity Levels For Water Resources in Bursa Region", *Int. Symp. on Water Resources And Environmental Impact*, Proceedings, pp. 97-106, 11-13 July 2001, Istanbul.
- A.B.Tuğrul, E. Meric, N. Avsar, F. Baytas, N. Altınsoy, B. Ayaz, N. Doğan, S. Sahin, **İ. Yılmaz**, "Haliç Geç Kuvaterner (Holosen) İstifinin Radyoaktivite Seviyesi Tayini ile Değerlendirilmesi ve Diğer Değerlendirme Teknikleri ile Karşılaştırılması",*Haliç 2001 Sempozyumu*, Bildiri Kitabı, s: 253-267, 3-4 Mayıs 2001, İstanbul.

- A.B.Tuğrul, Z. Er, **İ. Yılmaz**, A.O. Yılmaz, "Tuzla'da Deniz Radyoaktivitesinin-Tayini ve Değerlendirmesi", *Marmara-2000 Sempozyumu*, Bildiri Kitabı, s: 566-574, 10-12 Ekim 2000, İstanbul.
- A.B.Tuğrul, **İ. Yılmaz**, S. Sahin, "İstanbul Su Rezervuarlarının Radyoaktivite Seviyelerinin Değerlendirilmesi", *Türkiye Su Kongresi*, Bildiri Kitabı, Cilt II, s: 611-619, 17-19 Ocak 2000, İstanbul.

PUBLICATIONS/PRESENTATIONS ON THE THESIS

- **I. Yilmaz**, F. O. Edis, H. Saygin and L. Davidson, "Parallel Implicit DNS of Temporally-Evolving Turbulent Shear Layer Instability", *(submitted to Journal of Computational and Applied Mathematics)*
- **I. Yilmaz**, F. O. Edis and H. Saygin, "Application of an all-speed implicit non-dissipative DNS algorithm to hydrodynamic instabilities", *(submitted to Computers and Fluids)*
- **I. Yilmaz**, L. Davidson, F. O. Edis and H. Saygin, "Application and Comparison of Two Different DNS Algorithms for Simulating Transition to Turbulence in Taylor-Green Vortex Flow", *Springer Proceedings in Physics, Progress in Turbulence and Wind Engineering IV* , vol.141, pp. 91-94, 2012.
- **I. Yilmaz**, F. O. Edis and H. Saygin, "Parallel Implicit DNS of Rayleigh-Taylor Instability", *Proceedings CD of ECCOMAS 2012*, 6th *European Conference on Computational Methods in Applied Sciences and Engineering*, 10-14 September, 2012, Vienna, Austria.
- **I. Yilmaz**, L. Davidson, F. O. Edis and H. Saygin, "Numerical simulation of Kelvin-Helmholtz instability using an implicit, non-dissipative DNS algorithm", *Journal of Physics: Conf. Ser.,*, Vol. 318, 032024, pp. 1-7, 2011.
- **I. Yilmaz**, L. Davidson, H. Saygin and F.O. Edis, "Application of An Implicit, Non-Dissipative Algorithm to Temporal Kelvin-Helmholtz Instability", *EPFDC 2010, European Postgraduate Fluid Dynamics Conf.*, 3-7 July, 2010, Paris, France.