Stabilizing Filters for High-Order Implicit Large Eddy Simulation

Mohsen Hamedi

A Thesis
in
The Department
of
Mechanical, Industrial, and Aerospace Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science at
Concordia University
Montreal, QC, Canada

July 2019

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Mohsen Hamedi**

Entitled: **Stabilizing Filters for High-Order Implicit Large Eddy Simulation**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final Examining Committee:

Chair
Dr. Alex De Visscher

Examiner
Dr. Charles Basenga Kiyanda

Examiner
Dr. S. Samuel Li

Supervisor
Dr. Brian Vermeire

Approved by
Chair of Department or Graduate Program Director

2019

Dean of Faculty

Abstract

Stabilizing Filters for High-Order Implicit Large Eddy Simulation

Mohsen Hamedi

High-order Flux Reconstruction (FR) schemes can simulate unsteady turbulent flows using Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS) in the vicinity of complex geometries. The application of FR schemes can be limited by non-linear instabilities, related to oscillatory behaviour of the element-wise numerical solution, causing nonphysical solutions. In this study, filtering is studied for hexagonal element types and solution polynomial of degrees 3, 4, and 5 at different Mach numbers ranging from 0.1 to 0.5. A new exponential filtering function is applied globally to all elements and artificially damps high-frequency oscillations to improve numerical stability. Numerous numerical tests have been performed to investigate different parameters in the exponential filtering function. The optimum set of these parameters is obtained such that the highest solution polynomial modes are damped while the lower ones remain untouched to preserve accuracy. The solution polynomial is filtered after each time step; however, the filtering operator is defined, through the concept of characteristic time, to be independent of the time-step size. The exponential filter has been implemented and rigorously studied to evaluate its accuracy. To verify the order of accuracy, advection of an isentropic vortex has been analyzed. To study accuracy for LES, the Taylor-Green vortex test case is studied as a free turbulent flow. Finally, a previously unstable wall-bounded turbulent channel flow test case along with a NACA0020 airfoil at high angle of attack are considered to study the effects of the proposed filtering on stability and accuracy of practical applications.

Acknowledgements

First and foremost, I would like to thank my beloved wife for her patience, encouragement, and support, without whom my life has no meaning.

I am so thankful to my parents as well who have been supporting me and my studies since the very first day that I stepped into a school and have made great sacrifices for my siblings and me.

Special thanks to my supervisor, Dr. Brian Vermeire, who increased my passion for engineering. I want to thank him for his continuous guidance and support throughout this research.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

CFD	Computational Fluid Dynamics
CPU	Central Processing Unit
DG	Discontinuous Galerkin
DNS	Direct Numerical Solution
DoF	Degrees of Freedom
FDM	Finite Difference Method
FEM	Finite Element Method
$FLOP/s$	Floating Point Operations per second
FR	Flux Reconstruction
FVM	Finite Volume Method
HPC	High-Performance Computing
$IATA$	International Air Transport Association
$ICAO$	International Civil Aviation Organization
$ILES$	Implicit Large Eddy Simulation
LES	Large Eddy Simulation
$NASA$	National Aeronautics and Space Administration
NS	Navier-Stokes
OoA	Order of Accuracy
PDE	Partial Differential Equation
$RANS$	Reynolds-Averaged Navier-Stokes
$RK44$	Fourth-order four-stage Runge-Kutta scheme
SD	Spectral Difference

SG	Staggered Grid
SGS	Sub-Grid Scale
SV	Spectral Volume

Greek Letters

α	Angle of attack
α	Damping parameter of the filtering function (in the context of filtering)
γ	Specific heat ratio
δ	Channel half width
ϵ	Dissipation rate of the energy
ε	Enstrophy
η	Kolmogorov length scale (in the context of turbulence)
η	Exponents summation of orthonormal basis highest degree term (in the context of filtering)
η_c	Cut-off degree
η_{max}	Maximum of exponents summation of orthonormal basis highest degree term
μ	Kinetic viscosity
ν	Kinematic viscosity
ξ, η	Spatial coordinates in reference element
ρ	Density
σ	Stress tensor
σ	Filtering function (in the context of filtering)
τ	Total shear stress
$\bar{\tau}$	Viscous shear stress
τ^R	Reynolds stresses
ϕ	Nodal basis function
ψ	Orthonormal basis function
$\boldsymbol{\omega}$	Vector of vorticity
Γ	Linear mapping function
Λ	Filtering matrix
Ω	Control Volume

Symbols

A	Characteristic area
C_D	Drag coefficient
C_L	Lift coefficient
C_P	Pressure coefficient
c	Chord length of airfoil
E	Total specific energy
E_k	Kinetic energy
$\boldsymbol{F}$	Vector of body forces
$\boldsymbol{f}$	Vector of flux function
f	Flux function in the x-direction
g	Correction function
h	Flux function in the y-direction
L	Characteristic length
L_d	Linear dimension of computational domain
L_x	Length of the 2D domain
L_y	Width of the 2D domain
l	Scale of the largest eddy
Ma	Mach number
N	Minimum number of sampling points required for DNS
N_e	Total number of elements
N_p	Total number of solution points
$\hat{\boldsymbol{n}}$	Normal vector to the surface
P	Pressure
P_0	Reference pressure
$\boldsymbol{q}$	Vector of heat flux
R	Gas constant
Re	Reynolds number
Re_τ	Reynolds number based on friction velocity
Re_b	Reynolds number based on bulk velocity

Re_c	Reynolds number based on centerline velocity
S	Control surface
S_v	Vortex strength
s	Strength of the filtering function
St	Strouhal number
T_0	Reference temperature
T_{char}	Characteristic time
t	Time
U_0	Characteristic velocity
U_{up}	Upstream velocity
$\boldsymbol{u}$	Vector of nodal coefficients
$\hat{\boldsymbol{u}}$	Vector of modal coefficients
u	Conserved variable (in the context of flux reconstruction)
u, v, w	Velocity components
u_τ	Friction velocity
u_b	Bulk velocity
u_c	Centerline velocity
$\boldsymbol{V}$	Velocity vector
V	Vandermond matrix
$\mathcal{D}$	Drag force
$\mathcal{L}$	Lift force
$\mathcal{P}$	Degree of solution polynomial
$\mathcal{R}$	Residual (semi-discrete space operator)

Superscripts

C	Correction
D	Discontinuous
L	Left boundary of the element
R	Right boundary of the element
δ	Numerical approximation

$+$	Normalization based on friction velocity (in the context of turbulent channel)
$+$	Right side of the interface (in the context of flux reconstruction)
$-$	Left side of the interface (in the context of flux reconstruction)
$*$	Non-dimensional

Subscripts

i	Solution point indicator in the x-direction
j	Solution point indicator in the y-direction
k	Element indicator in the x-direction
L	Left boundary of the element
l	Element indicator in the y-direction
R	Right boundary of the element

Chapter 1

Introduction

This chapter provides motivation for the current study and background on Computational Fluid Dynamics (CFD) and high-order methods. It begins with the impacts of air travel on the environment and denotes the importance of green aviation. This is followed by an introduction to turbulent flows, and then the governing equations of this study are presented. Then the National Aeronautics and Space Administration (NASA)'s CFD Vision 2030 report is summarized, and the importance of developing next-generation CFD tools is discussed. Finally, the objectives of this research and the thesis outline are given as closure to this chapter.

1.1 Environmental Impacts of Aviation

In the last two decades, global air passenger traffic demand has continued to increase. In 2017, the annual number of air passengers was 4.1 billion [4]. The International Air Transport Association (IATA) predicts that total air transport could double in the next two decades following present trends [5]. Hence, there is considerable debate about the environmental impacts of air travel, particularly the emissions of air pollutants and their impacts on public health and climate change.

The International Civil Aviation Organization (ICAO) has implemented emission standards for aircraft engines since the late 1970s. One of the important actions of this organization is a protocol recommended by them to measure nitrous oxides, carbon monoxide, unburned hydrocarbons and the smoke number for new engines [6]. This guideline has led to considerable improvements in air pollution, however, the predicted growth in global air passenger traffic demand might offset these gains, as predicted in the European Aviation Environment Report. Based on this report, the average fuel consumption of commercial aircraft was 3.4 litres fuel per 100 passenger kilometres, with a decrease of 8 percent since 2014 [7]. However, the full-flight CO_2 emissions, globally, is reported at 859 million tons in 2017 [8].

It follows that the environmental impacts of aircraft must be taken into consideration, due to the drastic growth in global air passenger traffic. If the current level of aircraft technology growth remains the same throughout the next decade, there will be a catastrophic rise in hazardous greenhouse gas emissions and noise pollution associated with aviation. To address these issues, engineers must take steps toward the advancement of green aviation technologies by designing next-generation aircraft to be as quiet, fuel efficient, and environmentally responsible as possible. In order to do this, current CFD tools have to be improved, as stated by the CFD 2030 vision report of NASA [9]. This is due to the complexity of unsteady turbulent flows, and their impact on the aerodynamic performance of an aircraft design.

1.2 Turbulence

Understanding turbulence is of critical importance in aircraft design, since it is the natural state of most fluid flows. This is not an easy task, and it requires both a knowledge of applied mathematics and physical insights into fluid dynamics. It has been a long time that mathematicians, engineers, and physicists have tried to model turbulent flows accurately, but they have alas ended up with modelling defficiencies [1]. Turbulence emerges in a wide range of applications, like the flow over aircraft and cars, ocean currents and rivers, the flow of blood through the cardiovascular system, and so on, which make it an interesting and important subject for the focus of engineers and scientists. The main characteristic feature of turbulent flows is chaotic velocity fluctuations in time, which is shown in Figure 1.1. Also, this velocity field is sensetive to initial conditions, with minor changes in the initial condition growing exponentially with time. For engineering applications, the velocity function is often decomposed into mean and fluctuating components, as shown in Figure 1.1 and described by Equation 1.1, where the mean component is obtained under the assumption of infinite time averaging.

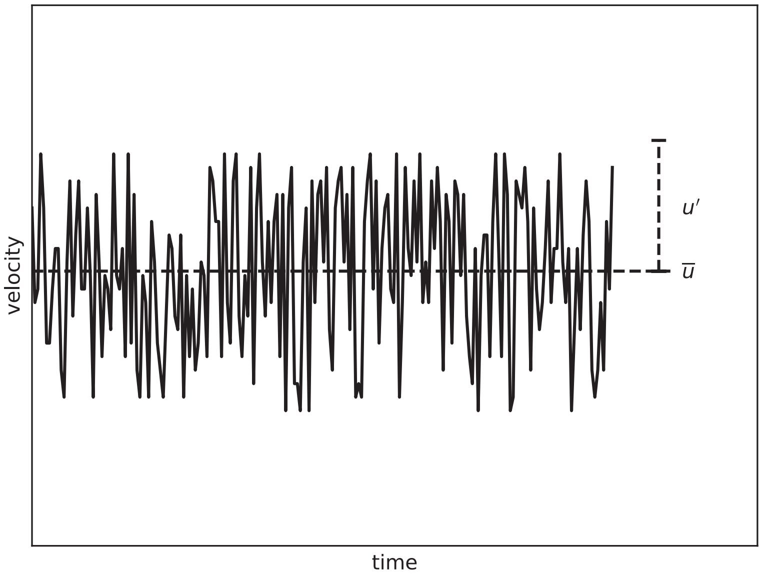


Figure 1.1. The mean and fluctuating velocity components in turbulent flow.

$$u(x,t) = \bar{u}(x) + u'(x,t), \tag{1.1}$$

where $\bar{u}(x)$ is the mean velocity and $u'(x,t)$ is the fluctuating component of the motion.

In determining the state of flow, the Reynolds number plays a key role. Reynolds number, defined in Equation 1.2, is the ratio of inertial to viscous forces. At high Reynolds numbers, the viscous forces and hence the viscous dissipation is low, and the flow is more likely to become unstable and turbulent [1]. The Reynolds number is defined by

$$Re = \frac{U_{up}L}{\nu}, \tag{1.2}$$

where U_{up} is the upstream fluid velocity, L is the characteristic length, and ν is the kinematic viscosity of the fluid.

Turbulent flows consist of eddies of different scales. The largest eddies are comparable to the characteristic length scale of the flow, and the smallest ones depend on the Reynolds number and are proportional to the dissipation length scale, known as the Kolmogorov length scale η [10]. In high Reynolds number flows, the largest eddies are created due to bulk flow instabilities. These large eddies are inertially unstable, which causes them to break-up into smaller eddies. The lifespan of these vortices is on the order of the turn-over time defined as L/u. The energy cascade from the large scale eddies to the smaller ones is driven by inertial forces and is due to inviscid instabilities. The energy cascade continues until the Reynolds number reaches on the order of unity, where viscous effects become dominant. At this point, the dissipation increases and starts to destroy the small scale turbulent structures. This energy cascade is shown in Figure 1.2, where at the end of the process, viscosity acts like a bin and dissipates all the turbulent kinetic energy at this scale.

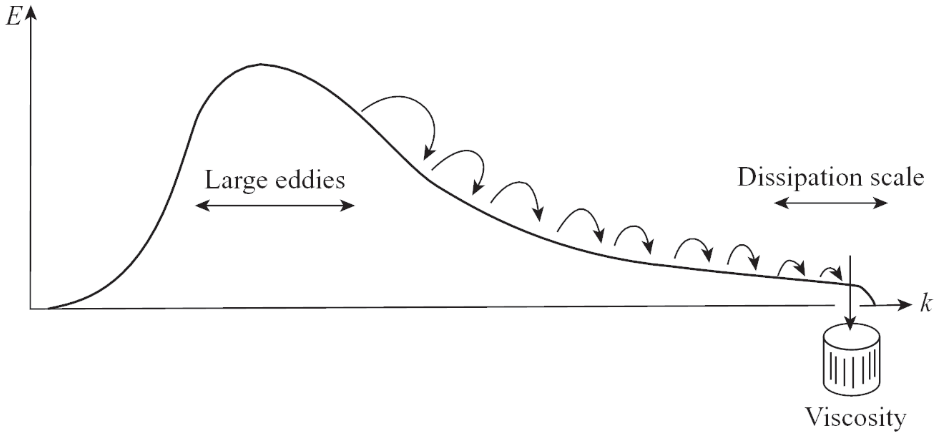


Figure 1.2. Diagram of the turbulent kinetic energy cascade in terms of energy versus wavenumber [1].

The relation between the smallest scale eddies, Kolmogorove scale, and the Reynolds number based on the largest scale eddies is [1]

$$\eta \sim l\,Re^{-\frac{3}{4}} = \left(\frac{\nu^3}{\epsilon}\right)^{\frac{1}{4}}, \tag{1.3}$$

where l is the scale of the largest eddy, known as integral scale, and $Re = ul/\nu$, and ϵ is the dissipation rate of the energy. The higher the Reynolds number, the larger the separation between the largest and smallest scales, which will have implications for CFD modelling.

In aerospace engineering, turbulence plays a crucial role in designing aircraft. To design the next-generation of more fuel efficient and environmentally friendly aircraft, one must understand the mechanism and structures of turbulence. Prior to the study of turbulence, the governing equations of fluid flow must be described.

1.3 Governing Equations

The governing equations for compressible Newtonian fluids under the continuum assumption are known as the Navier-Stokes (NS) equations, a set of Partial Differential Equations (PDEs), as given below.

- Conservation of Mass

$$\frac{d}{dt}\int_{\Omega}\rho\, d\Omega + \int_{S}\rho\,(\boldsymbol{V}.\hat{\boldsymbol{n}})\, dS = 0, \tag{1.4}$$

- Conservation of Momentum

$$\frac{d}{dt}\int_{\Omega}\rho\boldsymbol{V}\, d\Omega + \int_{S}\Big(\rho\boldsymbol{V}\,(\boldsymbol{V}.\hat{\boldsymbol{n}}) - \hat{\boldsymbol{n}}\sigma\Big)\, dS = \int_{\Omega}\boldsymbol{F}\, d\Omega, \tag{1.5}$$

- Conservation of Energy

$$\frac{d}{dt}\int_{\Omega}E\, d\Omega + \int_{S}(E\boldsymbol{V} - \sigma\boldsymbol{V} + \boldsymbol{q})\,.\hat{\boldsymbol{n}}\, dS = \int_{\Omega}\boldsymbol{F}.\boldsymbol{V}\, d\Omega, \tag{1.6}$$

where Ω is the control volume, S is the control surface, ρ is the density, $\boldsymbol{V}$ is the velocity vector, $\hat{\boldsymbol{n}}$ is normal vector to the surface, σ is the stress tensor, $\boldsymbol{F}$ is the vector of body forces, E is the total specific energy, and $\boldsymbol{q}$ is the heat flux vector.

The Navier-Stokes equations can be written in divergence form, using Gauss's theorem. The approach that has been investigated in this study is based on the differential form of governing equations.

- Conservation of Mass

$$\frac{\partial\rho}{\partial t} + \boldsymbol{\nabla}.(\rho\boldsymbol{V}) = 0. \tag{1.7}$$

- Conservation of Momentum

$$\frac{\partial(\rho\boldsymbol{V})}{\partial t} + \boldsymbol{\nabla}.(\rho\boldsymbol{V}\boldsymbol{V} - \sigma) = \boldsymbol{F}. \tag{1.8}$$

- Conservation of Energy

$$\frac{\partial E}{\partial t} + \boldsymbol{\nabla}.(E\boldsymbol{V} - \sigma\boldsymbol{V} + \boldsymbol{q}) = \boldsymbol{F}.\boldsymbol{V}. \tag{1.9}$$

1.4 Computational Fluid Dynamics

The modern history of fluid dynamics dates back to the seventeenth century, where pure experimental approaches were established. Eighteenth and nineteenth century, improvements in theoretical fluid dynamics lead researchers to combine both experimental and theoretical approaches in the twentieth century. The advent of powerful computers along with numerical algorithms to solve these physical problems revolutionalized this area of science. Hence, a third approach appeared, Computational Fluid Dynamics [11].

Computational fluid dynamics is a branch of fluid dynamics that utilizes digital computers along with applied mathematics to produce quantitative cost-effective predictions of real fluid flow phenomena based on approximate numerical solutions of the conservation laws governing fluid motion. CFD has developed quickly during the past several decades, which has led to improved simulations of aerodynamic flows. Implementing advanced CFD techniques to design aircraft reduces ground-based and in-flight tests, wind tunnel time for an aircraft development program, and the cost and risk of optimizing designs. CFD as a physics-based simulation technology improves our understanding and insight into critical physical phenomena. Designing and analyzing engineering systems often requires experimental testings that is impractical due to model complexity or wind tunnel limitations, giving rise to the inevitable need for CFD. Thus, the development of CFD techniques is necessary to make such studies feasible, and also reduces the cost and risk of designing such systems.

The utility of CFD also arises due to the lack of a general analytical solution to the governing equations. The very first step in solving PDEs using CFD is discretization, which is a process of transferring the continuous PDE system into a discrete approximation of that PDE to solve it numerically. The approximate numerical solution of any PDE gives answers only at a set of discrete points in the domain, known as grid points, while the analytical solution gives a continuous solution that varies continuously throughout the domain.

For simplicity, we consider a 1D conservation law with a general divergence form of

$$\frac{\partial u(x,t)}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \tag{1.10}$$

where $u(x,t)$ is the conserved variable, and $f(u)$ is the flux function.

Different methods for spatial and temporal discretization of PDEs of this form are explained in the following sections.

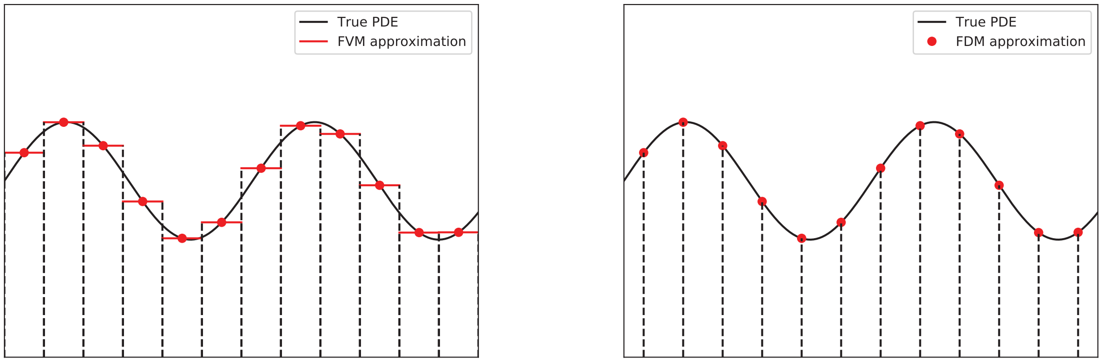
1.4.1 Spatial Discretization

Well-known methods for spatial discretization include the Finite Volume Method (FVM), Finite Difference Method (FDM), and Finite Element Method (FEM) [12].

In the FV method, the solution is represented using a discrete number of volumes with a constant value inside of each. Since the assumption is that the solution is piecewise constant, it introduces some inaccuracies. This method solves conservation laws in integral form. In the FD method, a discrete number of points is used to represent the approximate solution of the true PDE. The conservation laws are solved in divergence form. However, this method does explicitly control the values between discrete points.

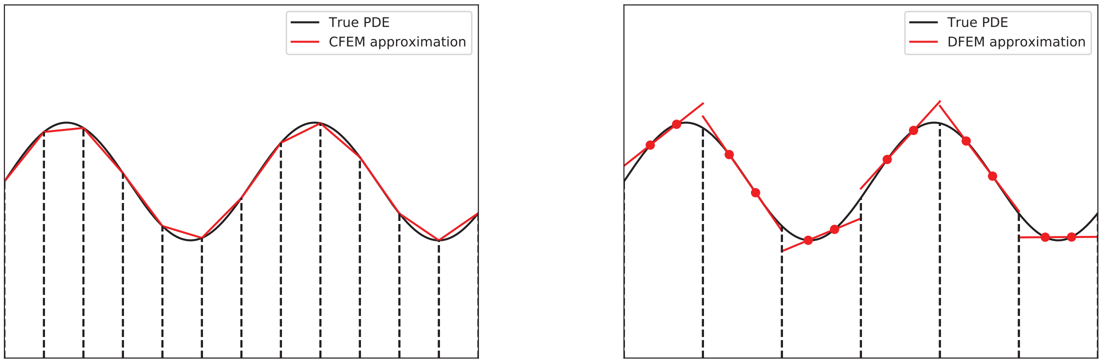
Finally, the FE method is subcategorized into continuous and discontinuous FE methods. In the continuous case, the solution is represented using a discrete number of points and basis functions are defined at each point to describe values between. In the discontinuous FE method, a discrete number of volumes is used, where the solution is represented by points in each volume and basis functions at each point. The basis functions have the property of returning the value of one at each point and zero at all neighbours within each volume. In both methods, the interpolation of the approximated solution between the grid points could be of any degree. The discontinuous FE method can be more accurate than the FV method. It allows the solution to be discontinuous like the FVM, but with a smooth internal solution. Inside each volume, the solution can be approximated using a polynomial of any desired degree passing through the internal grid points.

Examples of these spatial discretization techniques are shown in Figure 1.3, for second-order accuracy. In the discontinuous finite-element method, such as the Discontinuous Galerkin Method (DGM), the order of accuracy can be easily increased by adding more solution points within each element. The spatial discretization method of this study is discussed thouroughly in the next chapter.



(a) Finite-Volume Method

(b) Finite-Difference Method



(c) Continuous Finite-Element Method

(d) Discontinuous Finite-Element Method

Figure 1.3. Different second-order spatial descritization techniques.

1.4.2 Temporal Discretization

After discretizing spatial derivatives, the temporal discretization must be considered. Now, the 1D conservation law can be written as

$$\frac{du}{dt} = \mathcal{R}(u), \tag{1.11}$$

where $\mathcal{R}(u)$ is the semi-discrete space operator. So, the left-hand-side temporal term of this equation must be discretized.

Perhaps the most popular approach for handling this temporal derivative is the class of Runge-Kutta time integrators. These Runge-Kutta methods are typically described using a Butcher tableau

[13]. This tableau has a general form,

Table 1.1. The general form of the Butcher tableau.

$$
\begin{array}{c|cccc}
c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\
c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\
\hline
 & b_1 & b_2 & \cdots & b_s
\end{array}
$$

From this, the solution at different stages can be obtained,

$$\left[u_{stage1} = u + \Delta t(a_{11}\mathcal{R}_1 + a_{12}\mathcal{R}_2 + \cdots + a_{1s}\mathcal{R}_s)\right]^t, \tag{1.12}$$

$$\left[u_{stage2} = u + \Delta t(a_{21}\mathcal{R}_1 + a_{22}\mathcal{R}_2 + \cdots + a_{2s}\mathcal{R}_s)\right]^t, \tag{1.13}$$

$$\vdots$$

$$\left[u_{stageS} = u + \Delta t(a_{s1}\mathcal{R}_1 + a_{s2}\mathcal{R}_2 + \cdots + a_{ss}\mathcal{R}_s)\right]^t, \tag{1.14}$$

and the solution at the next time-step will be

$$u^{t+1} = u^t + \Delta t\Big(b_1\mathcal{R}_1 + b_2\mathcal{R}_2 + \cdots + b_s\mathcal{R}_s\Big), \tag{1.15}$$

where in these equations, $\mathcal{R}_i = \mathcal{R}(u_i)$ is obtained from the spatial discretization technique. If the matrix A is strictly lower-triangle, it is an explicit scheme, otherwise it is implicit. Examples of temporal descritization techniques and their corresponding Butcher tableaus are given below.

1.4.2.1 Explicit Euler Scheme

The Butcher tableau is

Table 1.2. The Butcher tableau for explicit Euler scheme.

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

This scheme is explicit and contains only one stage. The solution at that stage and also at the next time-step are

$$u_{stage1} = u^t + \Delta t(0\mathcal{R}_1), \tag{1.16}$$

$$u^{t+1} = u^t + \Delta t\mathcal{R}_1. \tag{1.17}$$

1.4.2.2 Fourth-Order Four-Stage Runge-Kutta Scheme (RK44)

The Butcher tableau is

Table 1.3. The Butcher tableau for RK44 scheme.

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

From the Butcher tableau, it is clear that this scheme has 4 stages and is explicit. The solution at different stages and also at the next time-step are

$$u_{stage1} = u^t, \tag{1.18}$$

$$u_{stage2} = u^t + \frac{1}{2}\Delta t\mathcal{R}_1, \tag{1.19}$$

$$u_{stage3} = u^t + \frac{1}{2}\Delta t\mathcal{R}_2, \tag{1.20}$$

$$u_{stage4} = u^t + \Delta t \mathcal{R}_3, \tag{1.21}$$

$$u^{t+1} = u^t + \Delta t \left(\frac{1}{6}\mathcal{R}_1 + \frac{1}{3}\mathcal{R}_2 + \frac{1}{3}\mathcal{R}_3 + \frac{1}{6}\mathcal{R}_4 \right). \tag{1.22}$$

1.5 Modern Hardware and CFD

When using a discontinuous FE method and explicit time stepping, such as RK44, there are 3 fundamentally different types of operations.

1. Element-wise Operations. These are performed on all solution points in all elements at once, using matrix multiplies.
2. Point-wise Direct Operations. These are performed point by point, but they use only information from that point alone. So, there is a one-to-one operation at all points, and can be performed in a structured manner.
3. Point-wise Indirect Operations. These require information from two or more different points, often not located beside each other, often even in different matrices, and cannot be performed in a structured way.

Current CFD tools for scale-resolving simulations of unsteady compressible flows are mostly based on the FV method with second-order accuracy in space [14]. The FV method works well with unstructured meshes and also in the vicinity of complex geometries, however, it is dominated by type 3 operations that access the memory indirectly, making this method memory bandwidth limited. Historically, Central Processing Units (CPUs) were quite slow, while memory bandwidth was relatively fast. This is shown in Figure 1.4.

Hence, it has been observed that the FV method can only achieve about 3 percent of peak Floating Point Operations per second (FLOP/s), which makes the need for CFD tools improvement inevitable [15]. Around 2003, the FLOP/s per byte ratio increased, surpassing bandwidth, and conventional algorithms are now limited by memory bandwidth, rather than computing performance. While the FV method fails at taking advantage of High-Performance Computing (HPC), the Flux

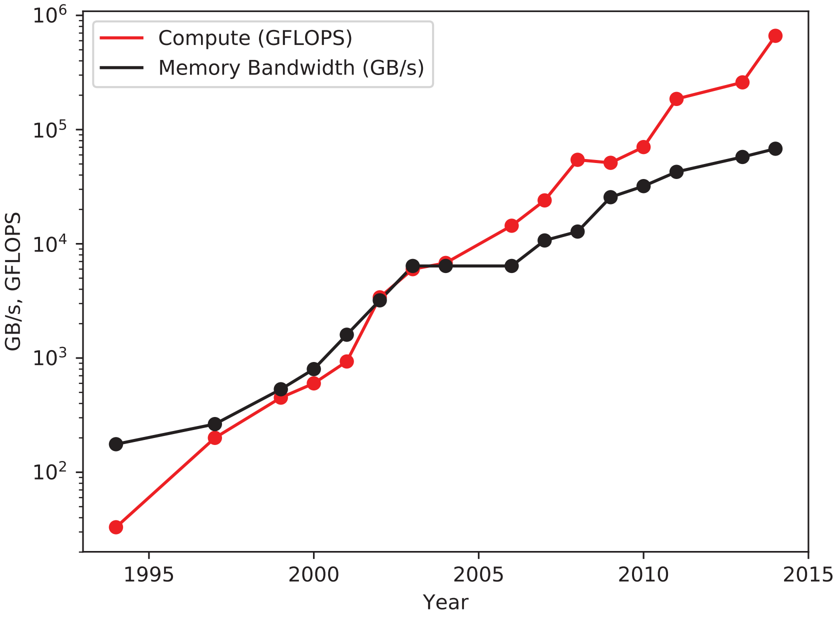


Figure 1.4. Trends in peak compute performance and memory bandwidth [2].

Reconstruction (FR) method can achieve over 55 percent of peak FLOP/s [14]. Hence, the FR approach is the focus of the current work, and will be discussed in the forthcoming sections.

1.6 High-Order Methods

Ekaterinaris [16] has provided a review on unstructured high-order numerical methods. The classical FEM has three main features: it is an unstructured numerical method, has relatively low computational cost, and high-order accuracy can be achieved. In the FEM, the solution is globally continuous and is typically represented using high polynomial degrees on an element-wise basis, and the interfaces of neighboring elements share the same value. Since the solution is defined globally, a global mass matrix must be inverted, which has a high computational cost. However, in the FVM, the stiffness matrix is single valued and must be inverted only inside each element because the solution is defined locally.

The combination of element-wise high-order accuracy of the FEM and localized solution representation of the FVM leads to defining new approaches: Discontinuous Galerkin (DG) [17, 18], Spectral Volume (SV) [19], Spectral Difference (SD) [20], and Staggered Grid (SG) [21]. These methods represent the high-degree solution locally (element-wise).

Flux Reconstruction, which is discussed in Chapter 2, is a single framework having the ability to recover a number of these different unstructured high-order element-wise FEM numerical methods.

1.7 Thesis Objectives

While useful, the high-order flux reconstruction approach can be sensitive to non-linear instabilities arising due to oscillatory behaviour of the element-wise numerical solution. Typically, a total number of degrees of freedom (DoF) is required to reach a desired level of accuracy for a given simulation, but this might be insufficient for stability. So, a simulation might need a higher number of total DoF than required for the desired level of accuracy, to be stable. In this thesis, a filtering operator is defined and studied, by which the stability can be ensured using fewer DoF, while accuracy is preserved. This is shown in Figure 1.5.

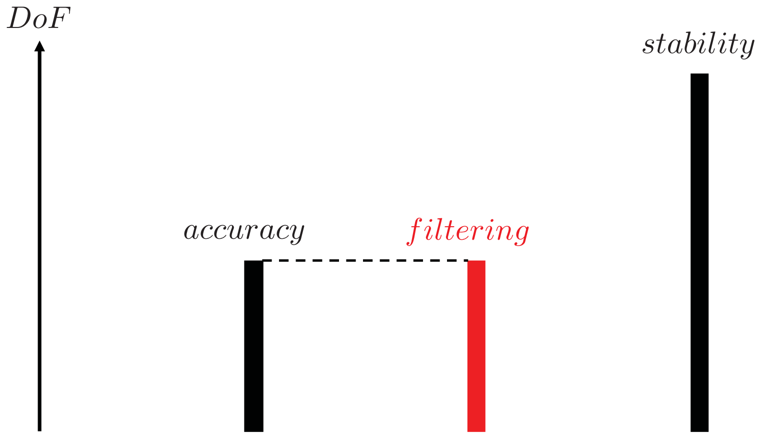


Figure 1.5. The required total number of DoF.

1.8 Thesis Outline

In Chapter 2, the theoretical framework is discussed. Different turbulence modelling approaches are compared and the necessity of using higher-order methods is provided. Then, the flux reconstruction approach is explained in one and multi-dimensions. Then the main focus of this thesis, a filtering stabilization technique, is thoroughly discussed in Chapter 3. In the next chapter, this technique is validated by studying the observed order of accuracy and the accuracy of large eddy simulations. Using these filters, in Chapter 5, two practical cases have been simulated, a previously unstable turbulent channel using a course mesh ($\approx \frac{3}{8} DoF_{DNS}$) along with a NACA0020 airfoil at a high angle of attack. Finally, conclusions and recommendations for future work are provided in Chapter 6.

Chapter 2

Theoretical Framework

2.1 Turbulence Modelling

The application of turbulent flows ranges from the flow of air around an aircraft to the flow of blood in our cardiovascular system. Since these flows are commonplace in most real-life scenarios, accurate prediction is of high concern. There is no analytical solution to the evolution of such flows hitherto, despite many studies. However, the advent of digital computers has made it feasible to predict the evolution of turbulent flows via mathematical models.

There are different approaches to predict the evolution of turbulent flows, such as Direct Numerical Simulation (DNS), Large Eddy Simulation (LES), Implicit LES, and Reynolds-Averaged Navier-Stokes Simulations (RANS). In these approaches, the initial and boundary conditions can be precisely controlled, whereas this is difficult to achieve in the laboratory. Also, the entire history of the evolution of the flow is stored and can be used for further investigations. Each of the aforementioned approaches is explained briefly in the following sections.

2.1.1 Direct Numerical Simulation

The growing field of DNS is considered an attractive simulation method for fully developed turbulent flows, since the whole span of eddy sizes is simulated from the smallest to the largest. In 1972, Orszag and Patterson [22], performed the first-ever computer simulation of a fully developed turbulent flow using DNS.

The Kolmogorov microscale, as the smallest eddy size in turbulence, was given in Equation 1.3. In order to solve all of the scales from the largest, integral length scale l, to the smallest, Kolmogorov length scale η, the grid spacing size, Δx, must be proportional to η. So, the minimum grid-spacing size required for DNS scales like [1]

$$\Delta x \sim \eta \sim l\, Re^{-\frac{3}{4}}. \tag{2.1}$$

For any three-dimensional domain, L_d being the linear dimension of the computational domain, the minimum number of sampling points required for DNS will scale like [1]

$$N \sim \left(\frac{L_d}{\Delta x}\right)^3 \sim \left(\frac{L_d}{l}\right)^3 Re^{\frac{9}{4}}. \tag{2.2}$$

This equation shows that a vast number of grid points is required for DNS to simulate high Reynolds number regimes, makes it inapplicable to most flows of interest for engineers, like the flow over an aircraft where the Reynolds number is particularly high.

2.1.2 Large Eddy Simulation

In turbulent flows, the small scale eddies are often relatively isotropic and low energy, so those scales can be left to a model. The Large Eddy Simulation method resolves the important, energy-containing, large eddies of turbulence accurately, and then it uses models to predict the effect of non-resolved small scale eddies on the flow, known as SubGrid-Scale (SGS) modelling. In Figure 2.1 the resolved scales of both DNS and LES are shown, along with the SGS model range in LES.

2.1.3 Implicit Large Eddy Simulation

The discretization schemes of the Navier-Stokes equations typically produces truncation errors which can, in particular situations, act as a simple SGS model. This approach is known as Implicit LES (ILES) [23]. In this study, ILES has been used for turbulence modelling. ILES relies on numerical dissipation, concentrated at high frequencies, to damp the under-resolved flow features.

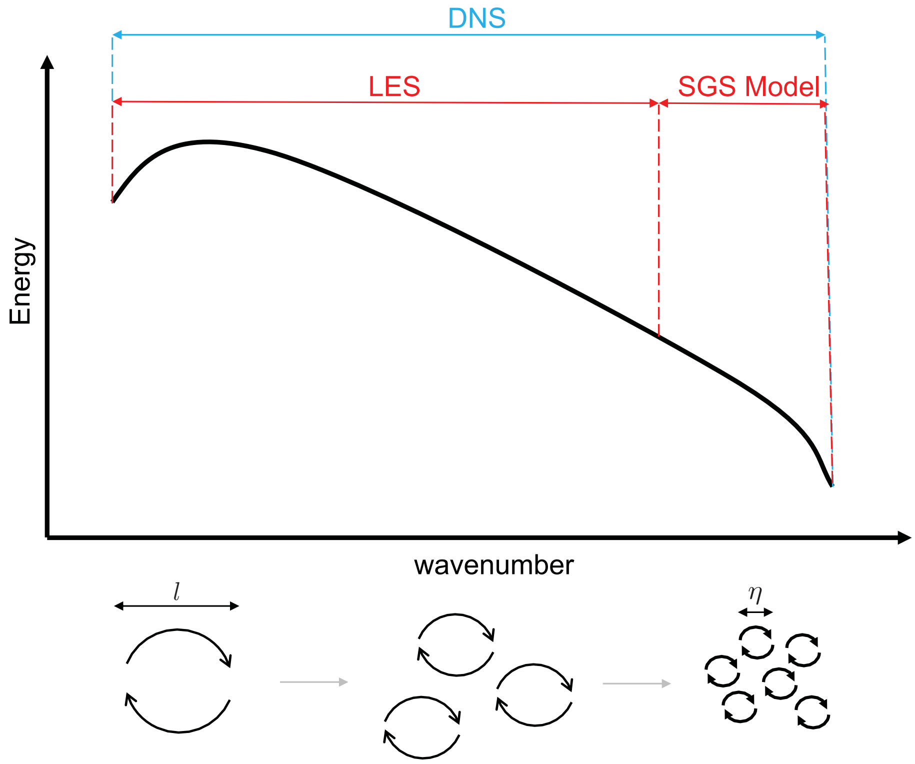


Figure 2.1. Diagram of resolved scales in DNS and LES, adapted from [3].

2.1.4 Reynolds-Averaged Navier-Stokes

Fluid flows can also be investigated using the Reynolds-averaged Navier–Stokes (RANS) approach. The RANS equations are the time-averaged Navier-Stokes equations, in which each variable is decomposed into a time-averaged and fluctuating parts, similar to Equation 1.1. In unsteady separated flow cases, RANS usually gives unsatisfactory results with respect to experiments [24]. Derivation of RANS equations [25] are given below.

The time-averaging method defines the time-averaged fluctuating values to be zero,

$$\overline{u'} = 0. \tag{2.3}$$

Conservation of mass for incompressible flows can be written as

$$\boldsymbol{\nabla} . \boldsymbol{V} = 0. \tag{2.4}$$

This can be expanded as

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0. \tag{2.5}$$

The continuity equation is decomposed to the mean and fluctuating parts as

$$\frac{\partial(\overline{u} + u')}{\partial x} + \frac{\partial(\overline{v} + v')}{\partial y} + \frac{\partial(\overline{w} + w')}{\partial z} = 0. \tag{2.6}$$

In the RANS approach, all of the equations are time-averaged, so the continuity equation will become

$$\overline{\frac{\partial(\overline{u} + u')}{\partial x} + \frac{\partial(\overline{v} + v')}{\partial y} + \frac{\partial(\overline{w} + w')}{\partial z}} = 0. \tag{2.7}$$

According to Equation 2.3, the continuity equation via the RANS approach is

$$\frac{\partial \overline{u}}{\partial x} + \frac{\partial \overline{v}}{\partial y} + \frac{\partial \overline{w}}{\partial z} = 0. \tag{2.8}$$

Similarly, the conservation of momentum for incompressible flow and in the x-direction is expanded as

$$\rho\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\right) = F_x - \frac{\partial P}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right). \tag{2.9}$$

The last three terms of the left hand side of Equation 2.9, can be written as

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} - u\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right). \tag{2.10}$$

The last paranthesis in Equation 2.10 is the continuity equation which is equal to zero. By decomposing all of the variables and rearranging them, the time-averaged momentum equations will be

$$\rho\left(\frac{\partial \overline{u}}{\partial t} + \overline{u}\frac{\partial \overline{u}}{\partial x} + \overline{v}\frac{\partial \overline{u}}{\partial y} + \overline{w}\frac{\partial \overline{u}}{\partial z}\right) = F_x - \frac{\partial \overline{P}}{\partial x} + \mu\nabla^2\overline{u} - \rho\left(\frac{\partial \overline{u'u'}}{\partial x} + \frac{\partial \overline{u'v'}}{\partial y} + \frac{\partial \overline{u'w'}}{\partial z}\right). \tag{2.11}$$

The total shear stress in the x-direction is

$$\tau = \mu\left(\frac{\partial\overline{u}}{\partial x} + \frac{\partial\overline{u}}{\partial y} + \frac{\partial\overline{u}}{\partial z}\right) - \rho\left(\overline{u'u'} + \overline{u'v'} + \overline{u'w'}\right), \tag{2.12}$$

where $\overline{u'u'} + \overline{u'v'} + \overline{u'w'}$ is known as Reynolds stresses, which must be modelled.

Advances in RANS methods have decreased computational costs relative to LES and DNS, which improve the affordability of numerous runs. However, all of the turbulent structures must be modelled, which is often inaccurate for transitional and separated flows.

2.1.5 Comparison

DNS, LES, and ILES are known as scale-resolving techniques. In spite of providing fairly accurate results, the computational cost of these techniques is more expensive than the RANS approach. In terms of accuracy, DNS is the most accurate method, however, it needs a large number of degrees of freedom to cover all of the eddy sizes. In DNS, the most expensive part of the computation is the intermediate to small scale eddies. Due to the fact that energy cascades from the larger eddies down to the smaller ones, the large scale eddies might not be heavily influenced by the small scale eddies [1]. So, one can think of LES instead of DNS, where those small scale eddies can remained unresolved and modelled, whereas the largest scales to the intermediate ones are solved accurately. In LES, the effect of unresolved small scale eddies on the turbulent flow are modelled using SGS models. However, ILES relies on truncation error to act as the SGS model.

As stated by NASA, it is expected that scale-resolving techniques will enable unprecedented insight in the aerodynamic design process, enabling superior designs at reduced cost and risk [9]. The ability of current RANS-based CFD methods is limited to problems that require an extensive experience base, and are not reliable for turbulent-separated flows. To overcome this obstacle, RANS modelling alone is not likely to be sufficient [9]. Turbulent-separated flows are poorly modelled by RANS approaches, so DNS, LES, and ILES are of industrial interest due to their capability in these regimes.

2.2 Flux Reconstruction

The Flux Reconstruction approach, a high-order accurate numerical method first introduced by Huynh [26], is gaining considerable attention due to its accuracy, generality, robustness, and suitability for modern hardware. In comparison to common low-order numerical methods, FR provides more accurate solutions using fewer total DoF and has less computational cost [27]. FR is a unifying approach that can recover existing schemes such as the spectral difference and discontinuous Galerkin methods. As with the SD and DG approaches, FR makes use of a high-order polynomial basis to represent the solution on each element. Such schemes are able to simulate unsteady turbulent flows using large eddy simulation and direct numerical simulation in the vicinity of complex geometries.

The FR framework is explained here, following Huynh's formulation [26].

2.2.1 One-Dimensional Formulation

For simplicity consider a 1D general conservation law of the form

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0, \tag{2.13}$$

where x is the spatial coordinate, t is time, $u = u(x, t)$ is the conserved scalar quantity, and $f = f(u)$ is the flux of u in the x direction.

The FR approach can be used to spatially discretise the spatial operator of such PDEs. In the one spatial dimension case, an arbitrary domain Ω is partitioned into a mesh of N_e non-overlapping elements such that

$$\Omega = \bigcup_{k=1}^{N_e} \Omega_k, \quad \bigcap_{k=1}^{N_e} \Omega_k = \emptyset, \tag{2.14}$$

where $\Omega_k = \{x \mid x_k < x < x_{k+1}\}$.

Each element consists of N_p discrete solution points. The exact solution, u, is approximated numerically, and within each element, via a solution polynomial of degree $\mathcal{P} = N_p - 1$, that is interpolated using these N_p discrete values of the approximated solution. Similarly, the exact flux, f, within each element, Ω_k, is approximated by a flux polynomial of degree $\mathcal{P} + 1$, which usually is discontinuous across cell interfaces. Hence, the total numerical solution, u^δ, and flux, f^δ, can be represented as the direct sum of their element-wise approximations u_k^δ and f_k^δ,

$$u^\delta = \bigoplus_{k=1}^{N_e} u_k^\delta \approx u, \qquad f^\delta = \bigoplus_{k=1}^{N_e} f_k^\delta \approx f. \tag{2.15}$$

In order to make FR simpler and more efficient, all operations are performed in a reference space, and each Ω_k is transferred to this reference space of $\xi \in [-1, 1]$. A linear mapping function is used as

$$\xi = \Gamma_k(x) = 2\left(\frac{x - x_k}{x_{k+1} - x_k}\right) - 1, \tag{2.16}$$

where Γ_k is the linear mapping function, and x_k and x_{k+1} are the left and right boundaries of Ω_k, respectively, and ξ is the location in the reference space. The linear mapping function also has a linear inverse of

$$x = \Gamma_k^{-1}(\xi) = \left(\frac{1-\xi}{2}\right)x_n + \left(\frac{1+\xi}{2}\right)x_{n+1}. \tag{2.17}$$

The solution polynomials collectively form a global solution approximation, which is discontinuous across each cell interface. The solution polynomial within each element can be interpolated using the solution values at each solution point, in the reference space, using the nodal basis function at each solution point via

$$u_k^\delta(\xi, t) = \sum_{i=1}^{N_p} u_{k,i}^\delta(t)\phi_i(\xi), \tag{2.18}$$

where $u_k^\delta(\xi, t)$ is the interpolated solution polynomial within a reference element, $u_{k,i}^\delta(t)$ is the approximated value of the solution at the i^{th} solution point, ξ_i, and $\phi_i(\xi)$ is the corresponding nodal basis function of the i^{th} solution point, ξ_i, in the reference space. The nodal basis functions in the one-dimensional spatial coordinates are the well-known Lagrange polynomials, shown in Figure 2.2, for a $\mathcal{P} = 5$ element.

$$\phi_i(\xi) = \prod_{j=1, j\neq i}^{N_p} \frac{\xi - \xi_j}{\xi_i - \xi_j}. \tag{2.19}$$

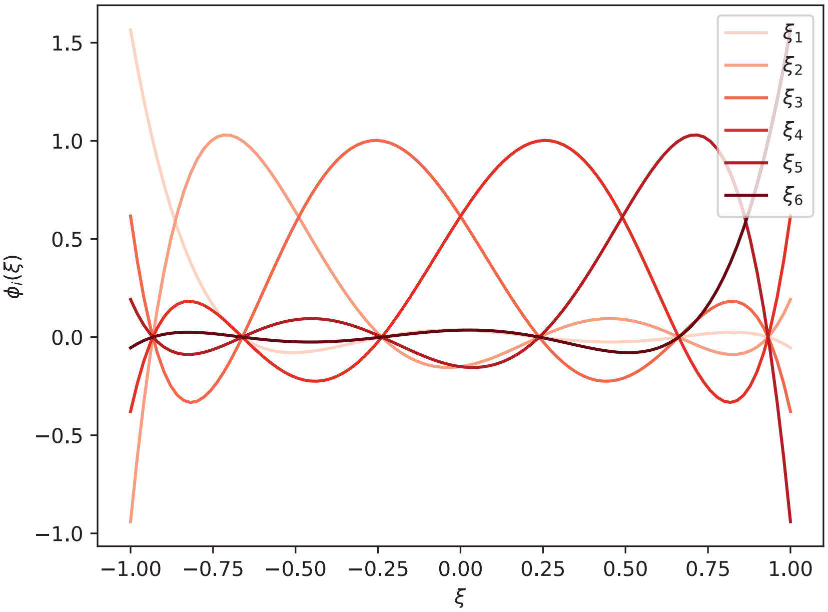


Figure 2.2. Nodal basis functions in 1D for $\mathcal{P} = 5$ using Gauss points.

The general conservation law in the reference space can then be written approximately as

$$\frac{\partial u_k^\delta}{\partial t} + \frac{\partial f_k^\delta}{\partial \xi} = 0, \tag{2.20}$$

where f_k^δ is an interpolated continuous flux function in the reference space, which its construction with details given below.

The flux values at each solution point are computed using the solution values, then the flux function is interpolated to the cell boundaries and is usually discontinuous across cell interfaces. The discontinuous flux function is constructed similar to the solution polynomial,

$$f_k^{\delta D}(\xi, t) = \sum_{i=1}^{N_p} f_{k,i}^\delta(t)\phi_i(\xi), \tag{2.21}$$

where $f_k^{\delta D}(\xi, t)$ is the discontinuous flux function within a reference element, $f_{k,i}^\delta(t)$ is the approximated value of the flux at the i^{th} solution point, ξ_i, and $\phi_i(\xi)$ is the corresponding nodal basis function of the i^{th} solution point, ξ_i, in the reference space. The superscript D denotes the discontinuity of the current flux approximation.

Since the spatial derivative of the flux function must be computed using a general conservation law, this function must be continuous to take the neighbouring elements data into account, and approximate the derivative of true flux function properly. In other words, to maintain global conservation, the flux between two elements must be continuous [28]. To address this discontinuity issue of the flux function, Huynh [26] proposed that a flux correction polynomial of degree $\mathcal{P}+1$ must be added to the discontinuous flux function,

$$f_k^\delta = f_k^{\delta D} + f_k^{\delta C}. \tag{2.22}$$

The computation of these corrections are given by Huynh [26] as follows, which approximates the zero function within the interior of the reference space.

$$f_k^{\delta C} = \left(f_k^{CL} - f_{k,L}^{\delta D}\right) g_L + \left(f_k^{CR} - f_{k,R}^{\delta D}\right) g_R, \tag{2.23}$$

where $f_{k,L}^{\delta D} = f_k^{\delta D}(-1,t)$, and $f_{k,R}^{\delta D} = f_k^{\delta D}(1,t)$. Also, $f_k^{CL} = f_k^{CL}(u_{k,L}^-, u_{k,L}^+)$ and $f_k^{CR} = f_k^{CR}(u_{k,R}^-, u_{k,R}^+)$ are common interface fluxes, a function of extrapolated values of the solution at each edge of neighbouring elements. f_k^{CL} and f_k^{CR} are computed by a Riemann solver at the flux points between elements. The correction functions, $g_L = g_L(\xi)$ and $g_R = g_R(\xi)$, of degree $\mathcal{P}+1$ have the following constraints

$$g_L(-1) = 1, \quad g_L(1) = 0, \tag{2.24}$$

$$g_R(-1) = 0, \quad g_R(1) = 1. \tag{2.25}$$

The spatial derivative of the approximated continuous flux function of the general conservation law in reference space is now

$$\frac{\partial f_k^\delta}{\partial \xi} = \frac{\partial f_k^{\delta D}}{\partial \xi} + \frac{\partial f_k^{\delta C}}{\partial \xi} = \sum_{i=0}^{N_p} f_{k,i}^\delta \frac{\partial \phi_i}{\partial \xi} + \left(f_k^{CL} - f_{k,L}^{\delta D}\right) \frac{\partial g_L}{\partial \xi} + \left(f_k^{CR} - f_{k,R}^{\delta D}\right) \frac{\partial g_R}{\partial \xi}, \tag{2.26}$$

which is in the same polynomial space as $\partial u_k^\delta / \partial t$, and so both terms of the general conservation law are of a degree $\mathcal{P}$.

As stated earlier, FR is a single framework capable of recoverying various schemes. There are three factors that FR relies on for such a property. The location of the solution points, ξ_i, the

Riemann solver in obtaining the common interface fluxes, f_k^{CR} and f_k^{CL}, and the choice of the correction functions, g_R and g_L.

The choice of the solution points location plays a key role in the stability and accuracy properties of the scheme when the flux is non-linear. However, in the case of linear fluxes it only affects initial projection error due to collocation projection of the initial condition onto the polynomial space of the solution [29].

Huynh [26] also showed that by choosing g_L and g_R as the right and left Radau polynomials, the collocation based nodal DG scheme will be recovered.

2.2.2 Two-Dimensional Formulation

In this section, the extension of the FR approach in 2D and for quadrilateral elements is given. The general conservation law in two dimensions is

$$\frac{\partial u}{\partial t} + \boldsymbol{\nabla}.\boldsymbol{f} = 0, \tag{2.27}$$

where $\boldsymbol{f} = f\hat{\boldsymbol{\imath}} + h\hat{\boldsymbol{\jmath}}$ is the flux function. The expanded form of the general conservation law in 2D is

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial h}{\partial y} = 0. \tag{2.28}$$

The extension of the FR approach in 2D can be cast as a tensor product of 1D operations. Firstly, assume that the computational domain is N_e elements of uniform rectangular shape and each cell, $\Omega_{k,l}$, is of length L_x, and width L_y. The centre of each cell is located at $(x_{k,l}, y_{k,l})$, where k is the element indicator in the x-direction, and l is that of the y-direction.

Similar to the 1D formulation, all of the calculations are done in a reference space. In 2D and for quadrilateral element types the reference element is a biunit square of $I \times I = I^2$ where $I \in [-1, 1]$. The solution points are the same in both x and y directions for simplicity. The solution points in 2D are (ξ_i, η_j) where $i, j = 1, ..., N_p$.

In a global perspective, on each rectangular element, $\Omega_{k,l}$, the solution, $u_{k,l}$, is approximated at $N_p \times N_p$ solution points, $(x_{k,l,i,j}, y_{k,l,i,j})$, giving the approximate numerical solutions, $u^{\delta}_{k,l}$.

In reference space, the solution polynomial of degree $\mathcal{P} = N_p - 1$ is interpolated using the solution values by the definition of nodal basis functions, shown in Figure 2.3.

$$u^{\delta}_{k,l}(\xi,\eta,t) = \sum_{i,j=1}^{N_p} u^{\delta}_{k,l,i,j}(t)\phi_{i,j}(\xi,\eta), \tag{2.29}$$

where,

$$\phi_{i,j}(\xi,\eta) = \phi_i(\xi)\phi_j(\eta), \tag{2.30}$$

where $\phi_{i,j}$ has the value of 1 at the solution point (ξ_i,η_j) and 0 at all other $N_p^2 - 1$ solution points.

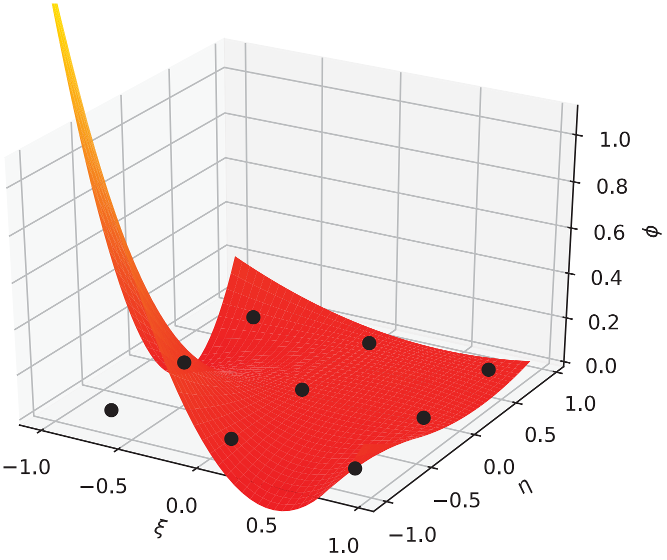


Figure 2.3. Nodal basis function in two-dimensions for $\mathcal{P} = 2$ and one of the solution points.

The solution polynomial can be, and usually is, discontinuous across cell interfaces, which causes the flux function to be discontinuous as well. However, the flux function across neighbouring cells must be continuous to preserve conservation, as in the 1D case. The discontinuous flux functions are evaluated on the reference element as

$$f_{k,l,i,j} = f(u_{k,l,i,j}), \tag{2.31}$$

$$h_{k,l,i,j} = h(u_{k,l,i,j}). \tag{2.32}$$

Similar to the numerical solution polynomial, the flux polynomial can be interpolated by the nodal basis function.

$$f^{\delta D}_{k,l}(\xi,\eta,t) = \sum_{i,j=1}^{N_p} f^{\delta}_{k,l,i,j}(t)\phi_{i,j}(\xi,\eta), \tag{2.33}$$

$$h^{\delta D}_{k,l}(\xi,\eta,t) = \sum_{i,j=1}^{N_p} h^{\delta}_{k,l,i,j}(t)\phi_{i,j}(\xi,\eta), \tag{2.34}$$

where the superscript D denotes the discontinuity of fluxes at the cell interfaces. From now on, we abbreviate the solution $u(\xi,\eta,t)$ to $u(\xi,\eta)$, and similarly for the fluxes.

In order to build the continuous flux function, the solution must be extrapolated to the boundaries. The extrapolated numerical solution at the left boundary is $u^{\delta}_{k,l}(-1,\eta)$, at the right boundary is $u^{\delta}_{k,l}(1,\eta)$, at the top boundary is $u^{\delta}_{k,l}(\xi,1)$, and finally the bottom boundary is $u^{\delta}_{k,l}(\xi,-1)$. These extrapolations can be done by the 1D procedure, and the continuous flux functions, $f^{\delta}_{k,l}$ and $h^{\delta}_{k,l}$, are reconstructed similar to Equation 2.22. Example solution and flux points are shown in Figure 2.4.

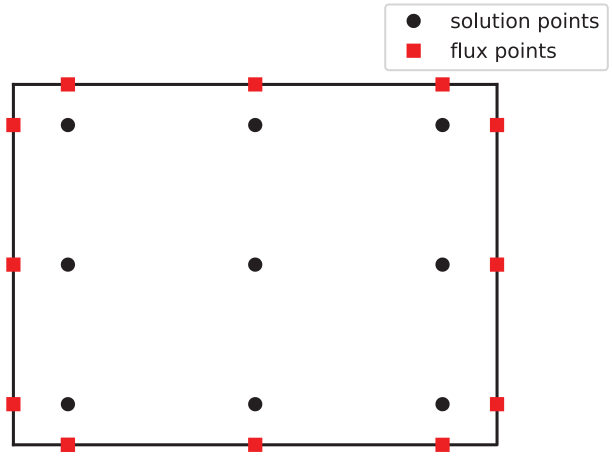


Figure 2.4. The solution and flux points for a quadrilateral element and $\mathcal{P} = 2$.

Similarly, the flux reconstruction approach can be extended to three-dimensions for hexagonal elements, and other types of elements as well. The filtering operator in this study is applied to hexagonal elements in 3D and is discussed thoroughly in the next chapter.

Chapter 3

Filtering

High-order Flux Reconstruction schemes are more sensitive to numerical instabilities due to their relatively low numerical dissipation. These numerical instabilities, mainly arising from non-linear behaviour of Navier-Stokes equations, can be dealt with using different techniques, which will be explained briefly.

The simplest way to decrease the non-linearity effects of NS equations is to choose a good set of points. Based on Runge's phenomena, equidistant points are a poor choice having spurious oscillations at the edges of an interval. To alleviate this, a set of points should be chosen that are biased towards the ends of the domain, i.e. Gauss points. In non-linear equations, aliasing error is usually of high concern. Aliasing arises when a higher degree polynomial is projected to a lower one. If a polynomial is of degree $\mathcal{P}$, then $\mathcal{P} + 1$ points are required to represent it exactly. In the NS equations, the flux is a non-linear function of the conserved quantity, therefore aliasing arises in the projection of the flux polynomial onto the space of the solution polynomial which is of degree $\mathcal{P}$. Based on mathematical procedure behind anti-aliasing, which is out of the scope of this work, it is evident that this technique is particularly expensive to implement. It is shown that anti-aliasing does remove aliasing error at significant computational cost, but there is also no guarantee in removing all instabilities [30].

The limiting technique is used for problems with shocks and strong discontinuities. The limiter essentially reconstructs the solution to reduce its oscillatory behaviour, using information from neighbouring elements. The positivity preserving limiter can also be used to ensure that the density and pressure are always positive. This limiting technique uses a discrete number of points and checks the value of density and pressure at these points. If the value of density and/or pressure is below a prespecified constant (usually close to zero) at any point, the limiter will scale the solution towards the mean cell solution to prevent negative values of pressure and/or density. However, this destroys the local accuracy in the limited cell. Another technique widely used for dealing with shocks is the idea of adding artificial dissipation. This technique needs a shock detector to add the artificial dissipation locally to preserve accuracy. However, this approach smears small-scale structures, such as vortices.

Different stabilizing techniques are shown in Table 3.1. In this study, filtering is investigated and implemented for implicit LES. In order to filter the solution, the solution polynomial must be represented in a different form, which is explained in the next section.

Table 3.1. Comparing different stabilizing techniques.

Stabilizing Technique	Computational Cost	Accuracy
Anti-Aliasing	×	✓
Limiting	✓	×
Positivity Preserving Limiter	✓	×
Artificial Dissipation	✓	×
Filtering	✓	?

3.1 Polynomial Representation

Polynomials can be represented in different equivalent forms, among which we work with nodal and modal forms, nodal in a sense that the polynomial is built at nodes, and modal in a sense that coefficients are polynomial modes. The general form of these nodal and modal representations of the solution, omitting the element index for brevity, is

$$u^\delta(\boldsymbol{\xi}, t) = \sum_{i=1}^{N_p} u_i^\delta(t)\phi_i(\boldsymbol{\xi}), \tag{3.1}$$

$$u^\delta(\boldsymbol{\xi}, t) = \sum_{i=1}^{N_p} \hat{u}_i^\delta \psi_i(\boldsymbol{\xi}), \tag{3.2}$$

where u_i^δ is the nodal coefficient or the solution values at each solution point in the reference element, $\hat{u}_i^\delta$ is the i^{th} modal coefficient, $\phi_i(\boldsymbol{\xi})$ is the nodal basis function of degree i, and $\psi_i(\boldsymbol{\xi})$ is the orthonormal basis function whose highest-order term is obtained from Pascal's triangle, that is explained later.

In the 1D case, the nodal and orthonormal basis functions are the well-known Lagrange and Legendre polynomials, respectively. These two polynomials are shown in Figure 3.1 and 3.2, for $\mathcal{P} = 5$.

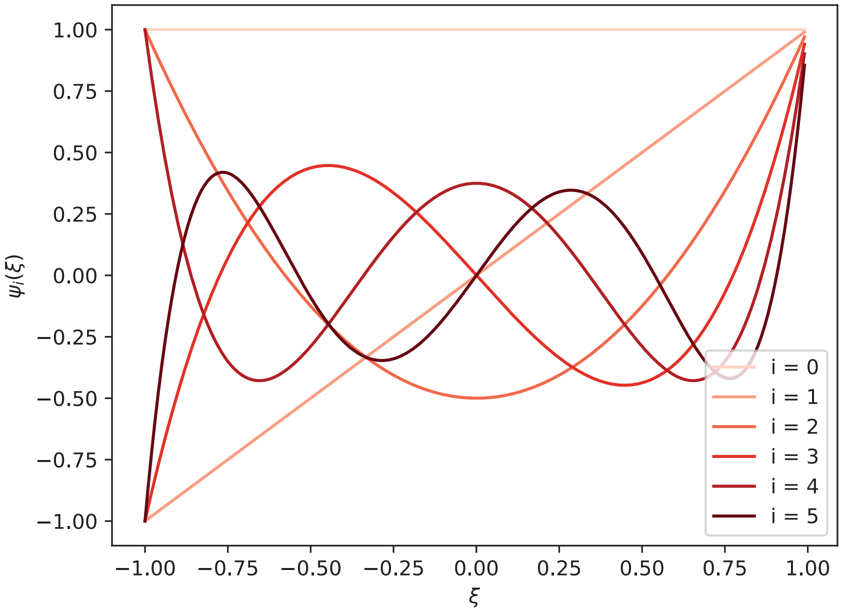


Figure 3.1. Legendre polynomials of different degrees.

To switch back and forth between these two polynomial representations, the Vandermond matrix is used,

$$\boldsymbol{u}^{\delta} = V\hat{\boldsymbol{u}}^{\delta}, \tag{3.3}$$

$$\hat{\boldsymbol{u}}^{\delta} = V^{-1}\boldsymbol{u}^{\delta}, \tag{3.4}$$

where $\boldsymbol{u}^{\delta}$ is the vector of nodal coefficients, or solution values at solution points within an element, and $\hat{\boldsymbol{u}}^{\delta}$ is the vector of modal coefficients. V is the Vandermond matrix which is defined as

$$V_{ij} = \psi_{j-1}(\boldsymbol{\xi}_i), \tag{3.5}$$

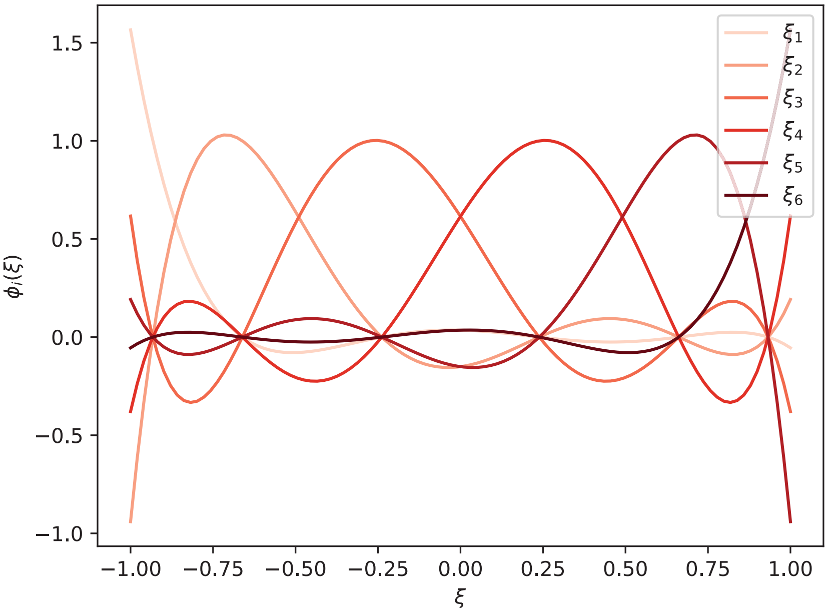


Figure 3.2. Lagrange polynomials at Gauss points for $\mathcal{P} = 5$.

where ψ_{j-1} is the orthonormal basis function of degree $j-1$, and $\boldsymbol{\xi}_i$ is the location of i^{th} solution point in the reference element. So, $\psi_{j-1}(\boldsymbol{\xi}_i)$ is the orthonormal basis function of degree $j-1$ evaluated at the i^{th} solution point.

3.2 Filtering Operator

The modal form of representing a polynomial is similar to a Fourier series, where the higher modes are responsible for the oscillatory behaviour of that polynomial, as seen in Figure 3.1. So, in the FR approach, in order to reduce the oscillation of the solution, the solution polynomial can be represented in the modal form. Then a filtering operator will be applied to reduce the energy of the higher modes. Then the modal representation of the solution must be switched back to the nodal form since in the nodal FR approach, this is the desired form.

The filtering operator can be defined as [31]

$$F = V \times \Lambda \times V^{-1}, \tag{3.6}$$

Table 3.2. The dimensional versus non-dimensional filtering operator for $T_{char} = 10$.

Dimensional Filtering Operator			Non-Dimensional Filtering Operator		
Δt	$niter$	$\boldsymbol{u}^{\delta f}$	n	F^*	$\boldsymbol{u}^{\delta f}$
0.1	100	$F^{100}\boldsymbol{u}^{\delta}$	100	$F^{\frac{1}{100}}$	$(F^*)^{100}\boldsymbol{u}^{\delta} = F\boldsymbol{u}^{\delta}$
0.01	1000	$F^{1000}\boldsymbol{u}^{\delta}$	1000	$F^{\frac{1}{1000}}$	$(F^*)^{1000}\boldsymbol{u}^{\delta} = F\boldsymbol{u}^{\delta}$

where Λ is a modal filtering matrix, and V^{-1} is the inverse of the Vandermond matrix. Using this definition, switching back and forth between different forms of polynomial representations is done in a single operation. The filtering operator must be applied to the vector of solution or nodal coefficients, $\boldsymbol{u}^{\delta}$. By multiplying Equation 3.6 to $\boldsymbol{u}^{\delta}$, the right hand side will be $V \times \Lambda \times \hat{\boldsymbol{u}}^{\delta}$. So, the modal filtering matrix will be multiplied to the vector of modal coefficients, which gives us the filtered modal coefficient, $\hat{\boldsymbol{u}}^{\delta f}$. Finally $V \times \hat{\boldsymbol{u}}^{\delta f}$ would be equal to the filtered solution, $\boldsymbol{u}^{\delta f}$.

The filtering operator is applied after each time-step which makes it dependant on the time-step size. So, if the size of the time-step changes, the strength of the filtering will change as well. To eliminate this issue, the filtering operator must be non-dimensionalized with respect to the size of the time-step. To do so, the filtering operator can be normalized with respect to the time-step size. The normalized form of the Λ matrix is

$$\Lambda^* = \Lambda^{\frac{1}{n}}, \tag{3.7}$$

where n is

$$n = \frac{T_{char}}{\Delta t}, \tag{3.8}$$

where T_{char} is a characteristic time relevant to the flow of interest, and Δt is the time-step size. Using this definition, the filtering operator will be

$$F^* = V \times \Lambda^* \times V^{-1}. \tag{3.9}$$

So, the filtering matrix is conserved over each characteristic time, no matter the time-step size. F^* will be applied after each time step, and after each characteristic time will be applied n times. This is summarized in Table 3.2, for a simulation with $T_{char} = 10$, where $niter$ is the number of iterations required to reach one characteristic time.

3.3 Filtering Matrix

The proposed modal filtering matrix is a square diagonal matrix with the filtering function values on its main diagonal, defined as

$$\Lambda_{ii} = \sigma(\eta), \tag{3.10}$$

where σ is the filtering function and η is the summation of the exponents of the highest degree term of the corresponding orthonormal basis function. This term is obtained using Pascal's triangle shown in Figure 3.3. To find the aforementioned terms in 3D, Pascal's triangle is used in different layers. These layers for hexagonal elements and $\mathcal{P} = 3$ are shown in Figure 3.4, where the highest degree term of each orthonormal basis function is highlighted in red. Also, all of the orthonormal basis functions for this example are shown in Table A.1.

$$\begin{array}{ccccccccc}
 & & & & 1 & & & & \\
 & & & x & & y & & & \\
 & & x^2 & & xy & & y^2 & & \\
 & x^3 & & x^2y & & xy^2 & & y^3 & \\
x^4 & & x^3y & & x^2y^2 & & xy^3 & & y^4
\end{array}$$

Figure 3.3. Pascal's triangle.

The dimension of the filtering matrix is dependent on the element type and the solution polynomial degree, $\mathcal{P}$, and is equal to the total number of DoF, i.e. the number of solution points. As an example, for a quadrilateral element type and for $\mathcal{P} = 2$, the highest degree term of each orthonormal basis function is shown in Figure 3.5 and the Λ matrix would be

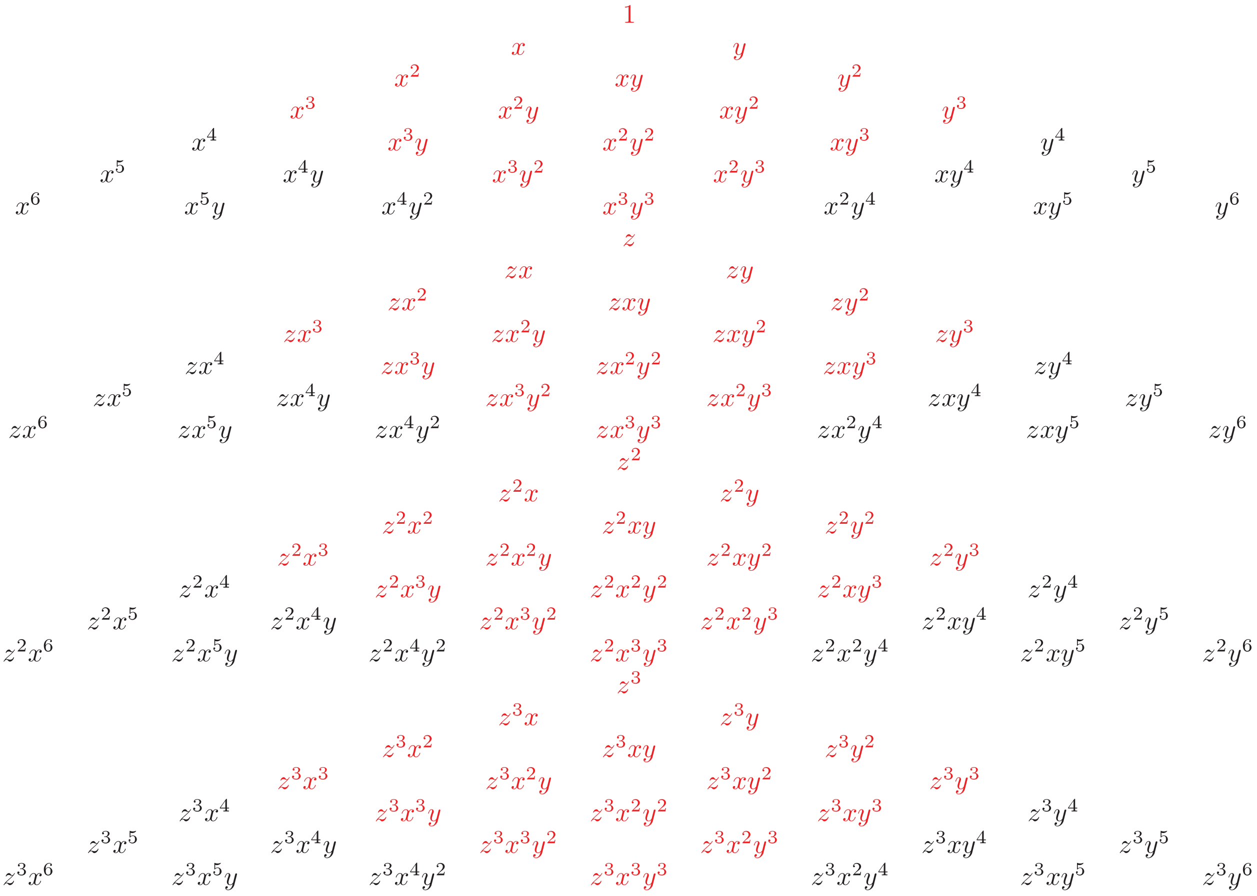


Figure 3.4. Highest degree terms of the orthonormal basis function.

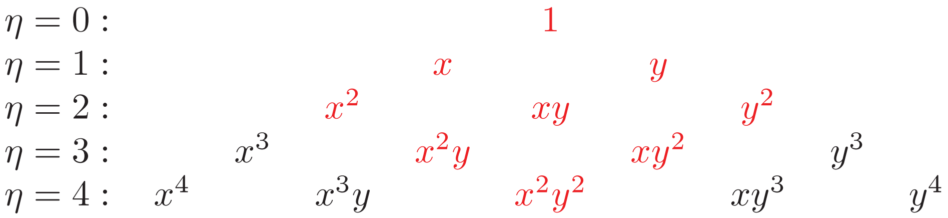


Figure 3.5. Highest degree terms of the orthonormal basis functions for a quadrilateral element type and $\mathcal{P} = 2$.

$$\Lambda = \begin{bmatrix} \sigma(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma(1) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma(1) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma(2) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma(2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma(2) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma(3) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma(3) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma(4) \end{bmatrix}_{9\times 9} . \tag{3.11}$$

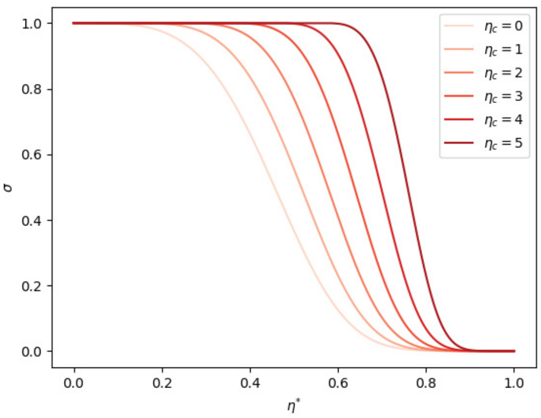
3.4 Filtering Function

The commonly used exponential filtering function is investigated in this study which is defined as [31]

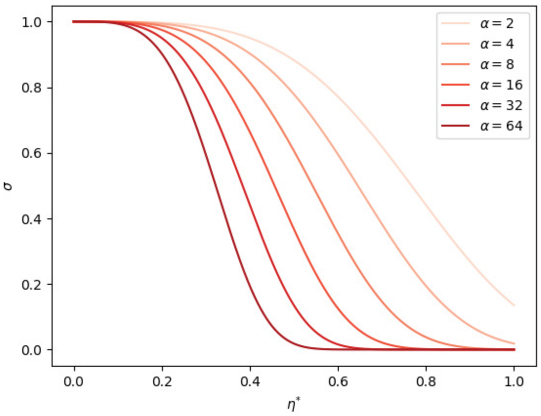
$$\sigma(\eta) = \begin{cases} 1 & 0 \leq \eta \leq \eta_c \\ exp(-\alpha(\frac{\eta-\eta_c}{\eta_{max}-\eta_c})^s) & \eta_c \leq \eta \leq \eta_{max} \\ 0 & \eta > \eta_{max} \end{cases} , \tag{3.12}$$

where α and s are the damping and strength parameters of the filter function, respectively, η_{max} is the maximum summation of the exponents of the orthonormal basis, and η_c is the cut-off degree, i.e. if $\eta_c = 2$, it means that the orthonormal basis terms of degrees less than and equal to 2 will remain untouched.

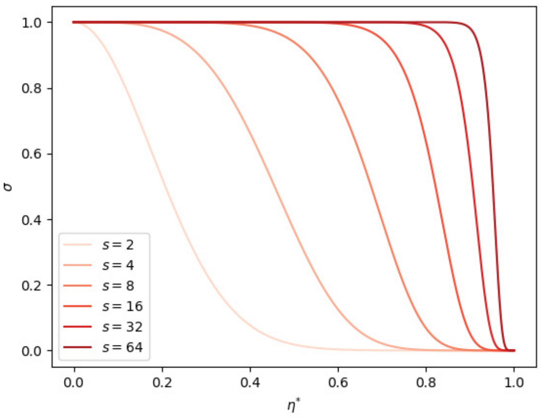
The relation between these filtering parameters are shown in Figure 3.6. It can be seen that the filtering function will move toward higher degrees by increasing the value of s and η_c and decreasing α. Determining optimal sets of filtering parameters is the focus of the next section.



(a) $\alpha = 16$, $s = 4$.



(b) $s = 4$, $\eta_c = 0$.



(c) $\alpha = 16$, $\eta_c = 0$.

Figure 3.6. The effect of different parameters on the filter function.

3.5 Parameter Specification

To find the optimum set of the filtering parameters, the Taylor-Green Vortex (TGV) test case has been studied. Our objective is to find the filter of minimal strength that can stabilize this test case in the limit of infinite Reynolds number. Hence, the Euler equations are used.

3.5.1 Taylor-Green Vortex

3.5.1.1 Introduction

In the TGV problem, the Navier-Stokes equations are solved in 3D to simulate a freely decaying turbulent flow in a periodic domain, generating a detailed turbulent spectrum. The TGV is a challenging benchmark test for high-order CFD methods. Several authors have analyzed the TGV using discontinuous high-order CFD methods. The TGV has been studied using modal Discontinuous Galerkin method [32], recovery-based Discontinuous Galerkin method [33], Discontinuous Galerkin Spectral Element Method [34, 35], and we study it using the Flux Reconstruction approach, and the results are given in Chapter 4. In this chapter, the TGV is investigated on a very coarse mesh to explore stability in the limit of infinite Reynolds number.

The initial flow field for the TGV for compressible flows is specified as [28]

$$u = +U_0 \sin(x/L)\cos(y/L)\cos(z/L), \tag{3.13}$$

$$v = -U_0 \cos(x/L)\sin(y/L)\cos(z/L), \tag{3.14}$$

$$w = 0, \tag{3.15}$$

$$P = P_0 + \frac{\rho_0 U_0^2}{16}(\cos(2x/L) + \cos(2y/L))(\cos(2z/L) + 2), \tag{3.16}$$

$$\rho = \frac{P}{RT_0}, \tag{3.17}$$

where u, v, and w are the velocity components, P is the pressure, and ρ is the density. The constant value of characteristic velocity, U_0, is obtained from the Mach number and the characteristic length, L, is equal to 1. Furthermore

$$\rho_0 = 1.0, \tag{3.18}$$

$$RT_0 = 1, \tag{3.19}$$

$$U_0 = Ma\sqrt{\gamma RT_0}. \tag{3.20}$$

3.5.1.2 Computational Details

The Euler equations in 3D have been solved with the TGV initial flow field, in the limit of $Re \to \infty$, with solution polynomials of degree $\mathcal{P} = 3,\ 4$, and 5. The domain for this study is a periodic cube with dimensions of $0 \leq x, y, z \leq 2\pi$, meshed with hexagonal elements, and the total number of DoF is 16^3. The mesh for $\mathcal{P} = 3$ is shown in Figure 3.7. A very coarse mesh is used to have an initially unstable simulation, and the filtering operator is applied using a wide range of filtering parameters. The strength parameter of the filter is set to fixed values of $s = 2, 4, 8, 16$, and the two other parameters are optimized to have the weakest filtering operator, to preserve accuracy, while stabilizing the TGV case. In order to have a wide range of turbulence intensities, the TGV is studied at different Mach numbers ranging from $Ma = 0.1$ to $Ma = 0.5$ and a total of more than 14000 simulations.

3.5.2 Results

The threshold of stability, determined via bisection optimization, for different solution polynomial degrees and different Mach numbers are given in Figures 3.8-3.12.

Any values of parameters on the left side of each line of Figures 3.8-3.12 is considered a weak filter that fails at stabilization, while parameter values on the right side of each line, form a strong filtering function that stabilizes the simulation but it might, and usually will, reduce accuracy via excessive filtering.

The challenging question now would be which set of the filtering parameters along these sets of lines is the best for each polynomial degree. The oscillatory behaviour of the solution is due to the highest modes in the polynomials, those higher modes must be damped. On the other hand, the filtering operator should not damp the lower modes, to preserve accuracy. Hence, the best set

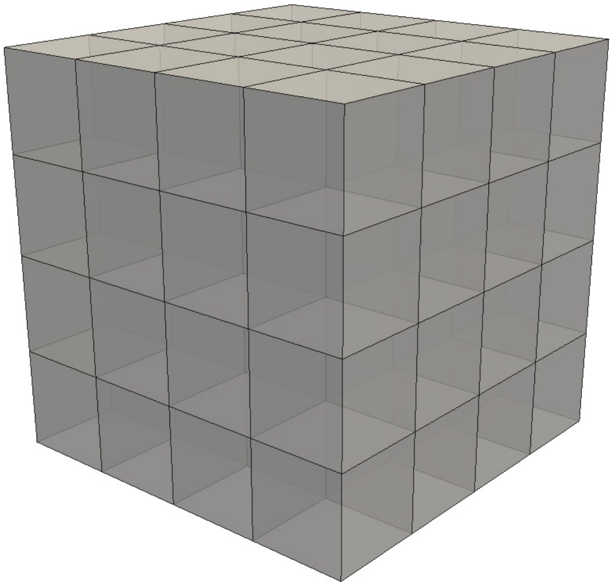
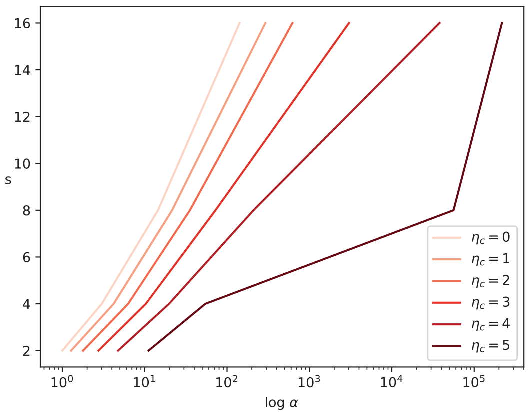


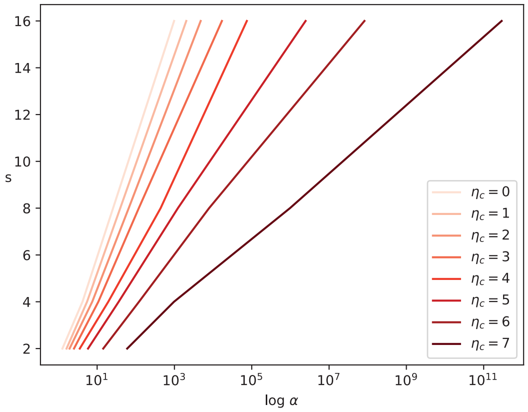
Figure 3.7. The coarse mesh used for $\mathcal{P} = 3$.

of filtering parameters is the one that damps the higher modes as much as possible, while leaving the lower modes untouched. All of the possible filtering functions are shown in Figures 3.13-3.17, where the best filtering function under these criteria is highlighted.

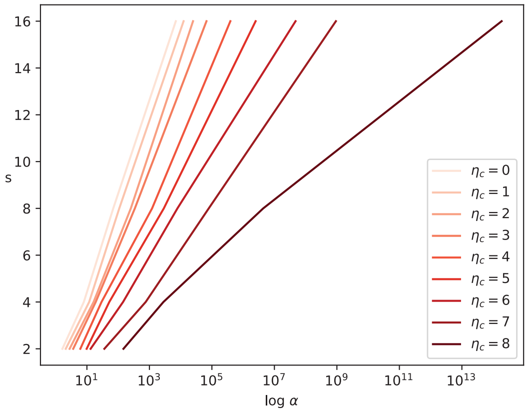
After specifying the best filtering function for each polynomial degree, the accuracy of these operators is investigated in the next chapter.



(a) $\mathcal{P} = 3$

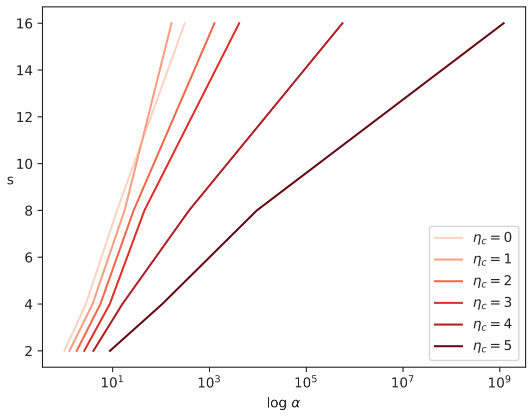


(b) $\mathcal{P} = 4$

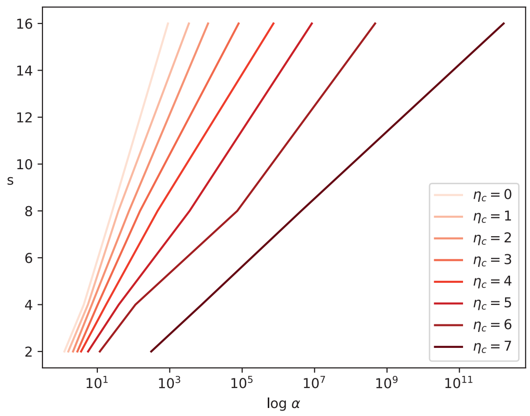


(c) $\mathcal{P} = 5$

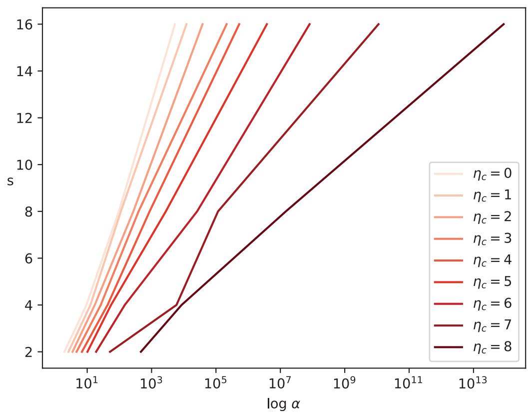
Figure 3.8. Plots of stability for different $\mathcal{P}$'s at $Ma = 0.1$.



(a) $\mathcal{P} = 3$

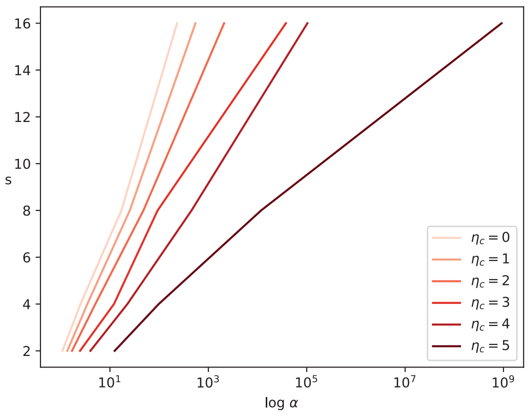


(b) $\mathcal{P} = 4$

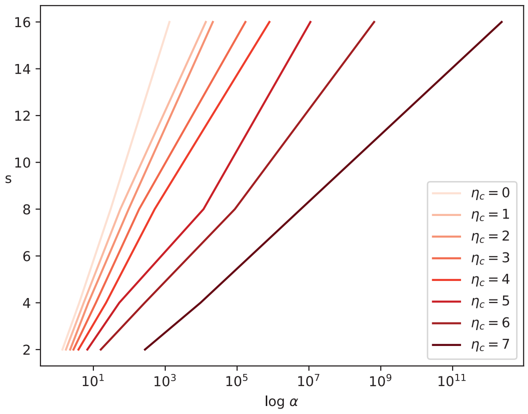


(c) $\mathcal{P} = 5$

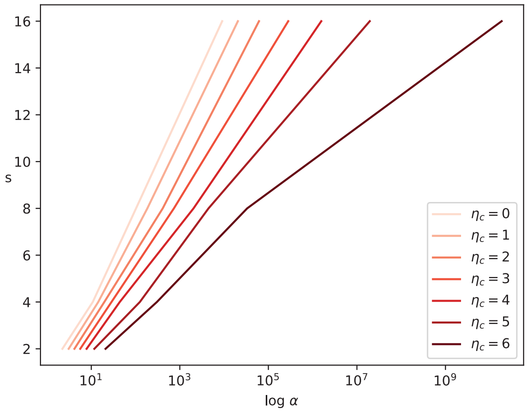
Figure 3.9. Plots of stability for different $\mathcal{P}$'s at $Ma = 0.2$.



(a) $\mathcal{P} = 3$

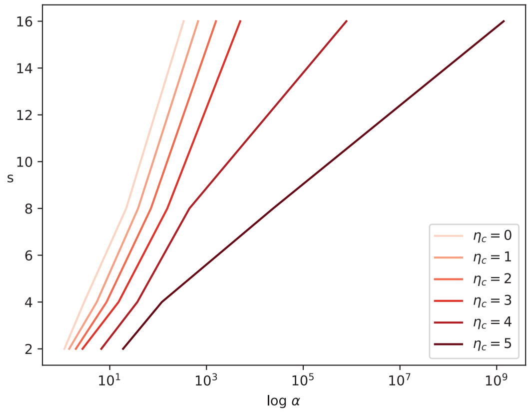


(b) $\mathcal{P} = 4$

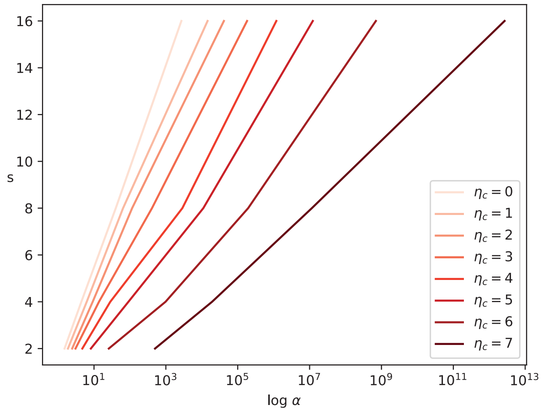


(c) $\mathcal{P} = 5$

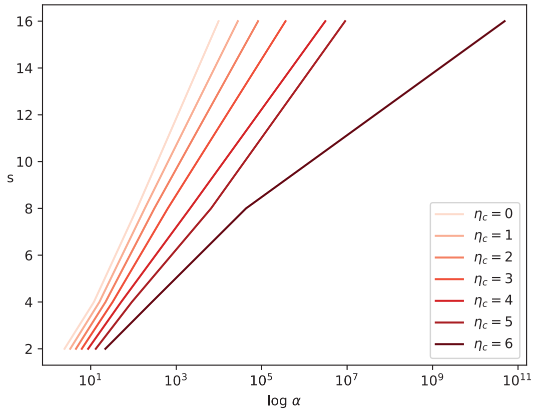
Figure 3.10. Plots of stability for different $\mathcal{P}$'s at $Ma = 0.3$.



(a) $\mathcal{P}$ = 3

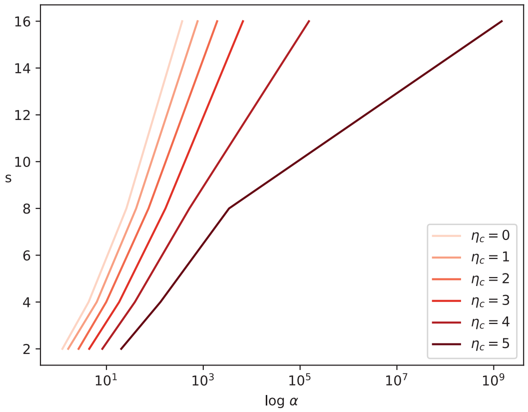


(b) $\mathcal{P}$ = 4

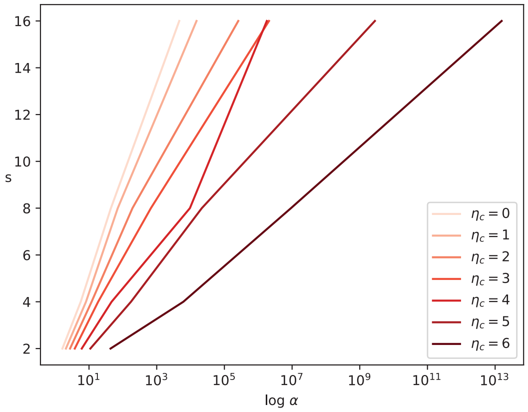


(c) $\mathcal{P}$ = 5

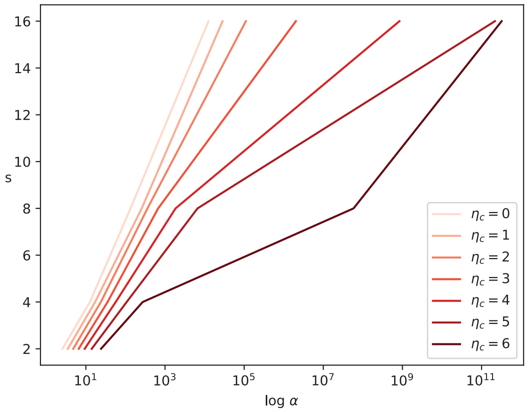
Figure 3.11. Plots of stability for different $\mathcal{P}$'s at Ma = 0.4.



(a) $\mathcal{P} = 3$

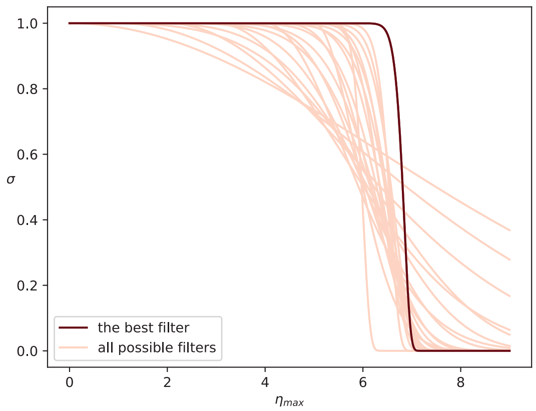


(b) $\mathcal{P} = 4$

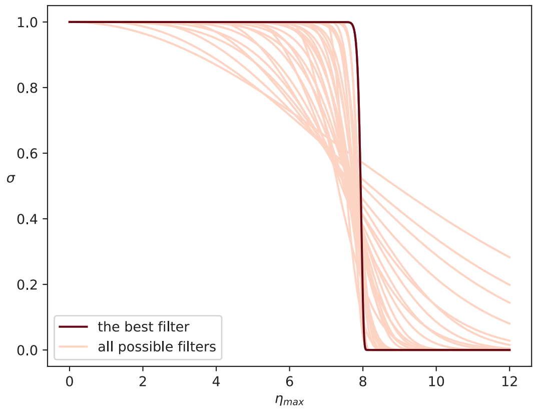


(c) $\mathcal{P} = 5$

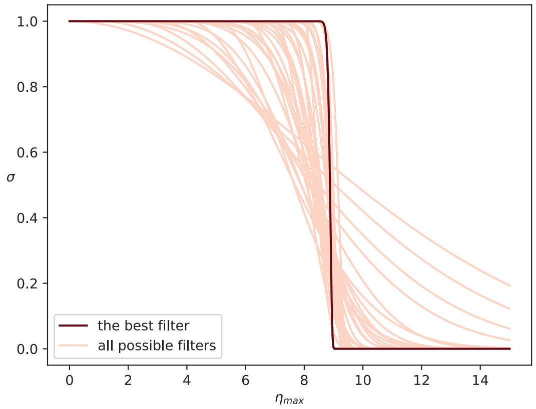
Figure 3.12. Plots of stability for different $\mathcal{P}$'s at $Ma = 0.5$.



(a) $\mathcal{P} = 3$

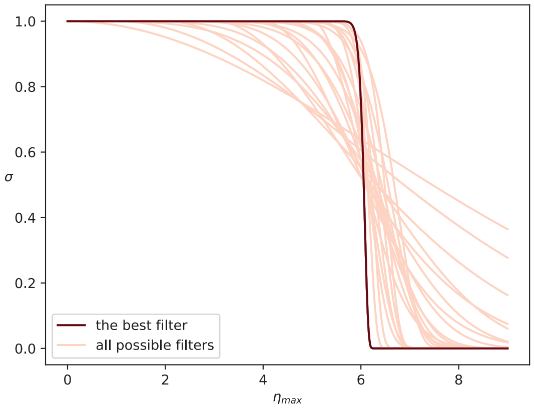


(b) $\mathcal{P} = 4$

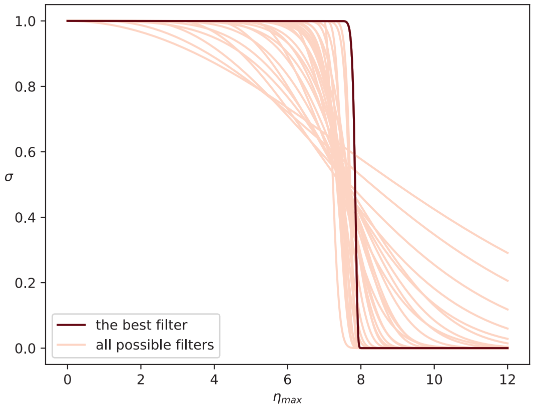


(c) $\mathcal{P} = 5$

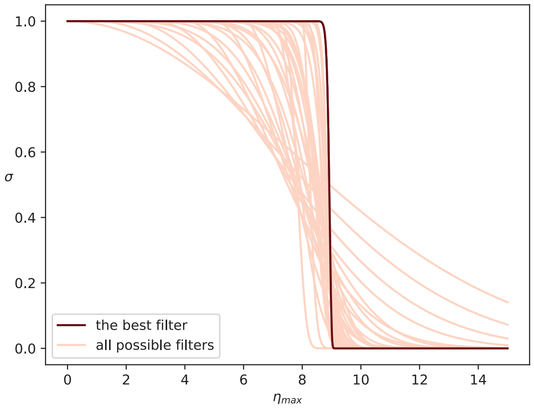
Figure 3.13. Plots of all possible filter functions for different $\mathcal{P}$'s and the best one at $Mach = 0.1$.



(a) $\mathcal{P} = 3$

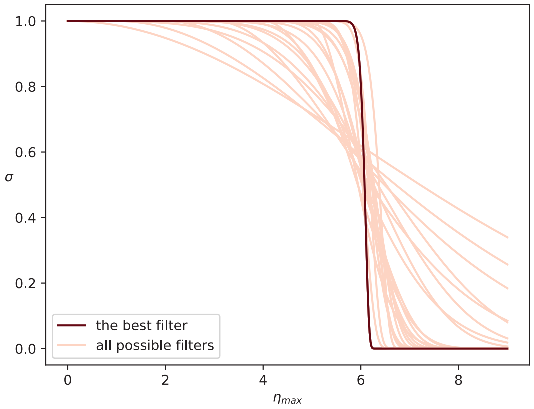


(b) $\mathcal{P} = 4$

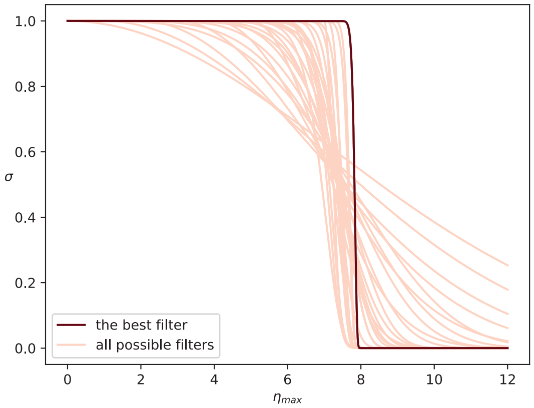


(c) $\mathcal{P} = 5$

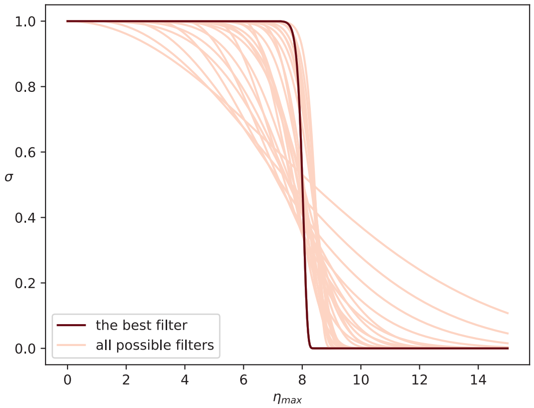
Figure 3.14. Plots of all possible filter functions for different $\mathcal{P}$'s and the best one at $Mach = 0.2$.



(a) $\mathcal{P} = 3$

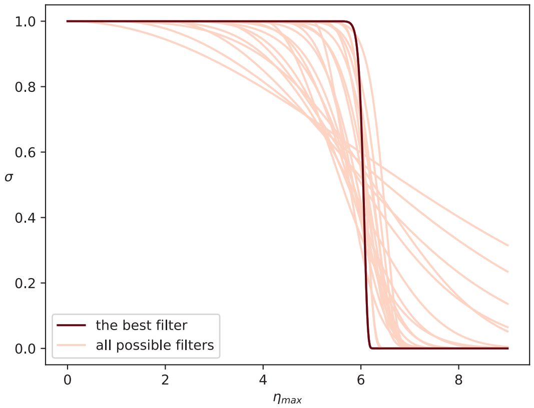


(b) $\mathcal{P} = 4$

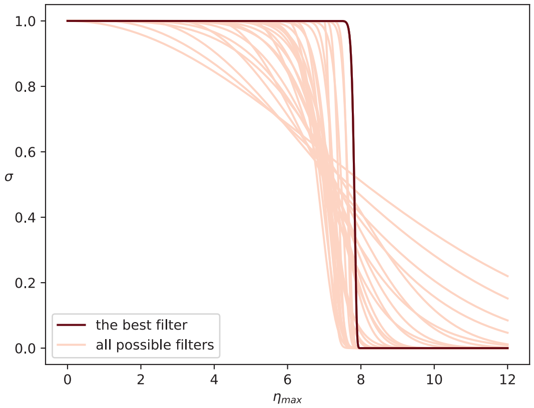


(c) $\mathcal{P} = 5$

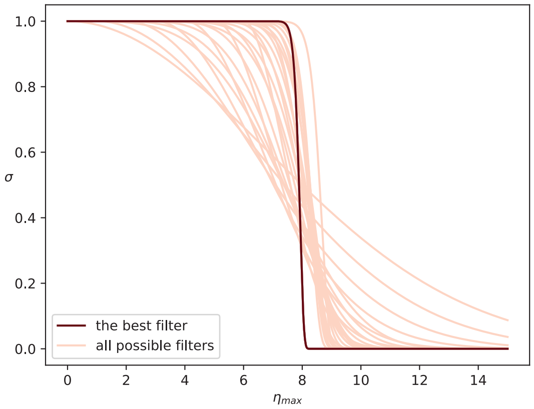
Figure 3.15. Plots of all possible filter functions for different $\mathcal{P}$'s and the best one at $Mach = 0.3$.



(a) $\mathcal{P} = 3$

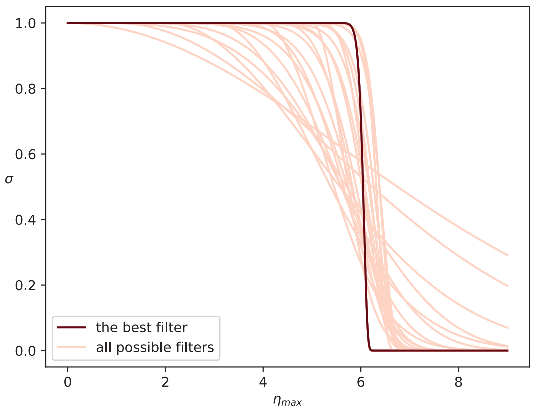


(b) $\mathcal{P} = 4$

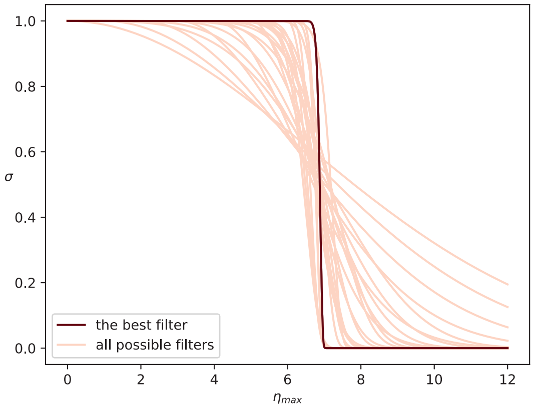


(c) $\mathcal{P} = 5$

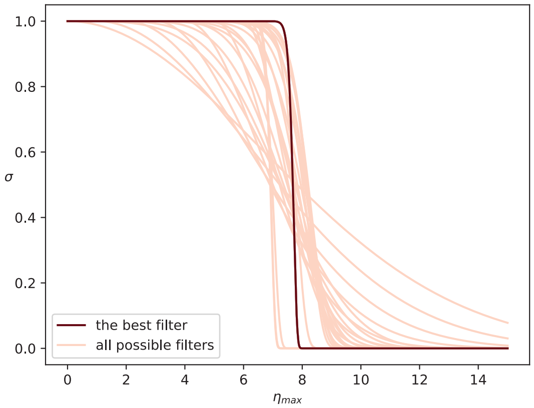
Figure 3.16. Plots of all possible filter functions for different $\mathcal{P}$'s and the best one at $Mach = 0.4$.



(a) $\mathcal{P} = 3$



(b) $\mathcal{P} = 4$



(c) $\mathcal{P} = 5$

Figure 3.17. Plots of all possible filter functions for different $\mathcal{P}$'s and the best one at *Mach* = 0.5.

Chapter 4

Verification and Validation

The objective of the current chapter is to verify that the proposed optimal stabilizing filters maintain accuracy and improve stability for practical simulations of turbulent flows.

4.1 Isentropic Vortex Advection

4.1.1 Introduction

The isentropic vortex advection is commonly used to test the accuracy of a flow solver. This case is used due to its simple implementation and known exact analytical solution at all times. The advection of the vortex with the mean flow is simulated using the Euler equations, where the exact entropy remains constant everywhere in the domain. The isentropic vortex advection has an initial flow field as

$$\rho = \left[1 - \frac{S_v^2 Ma^2(\gamma - 1)e^{2\varphi}}{8\pi^2}\right]^{\frac{1}{\gamma-1}}, \tag{4.1}$$

$$u = \frac{S_v y e^{\varphi}}{2\pi R}, \tag{4.2}$$

$$v = 1 - \frac{S_v x e^{\varphi}}{2\pi R}, \tag{4.3}$$

$$P = \frac{\rho^{\gamma}}{\gamma Ma^2}, \tag{4.4}$$

where ρ is the density, u and v are the velocity components, P is the pressure, $S_v = 13.5$ is the vortex strength, $Ma = 0.4$ is the free-stream Mach number, $\gamma = 1.4$ is the heat capacity ratio, $R = 1.5$ is the radius of the vortex, and φ is

$$\varphi = \frac{1 - x^2 - y^2}{2R^2}. \tag{4.5}$$

4.1.2 Computational Details

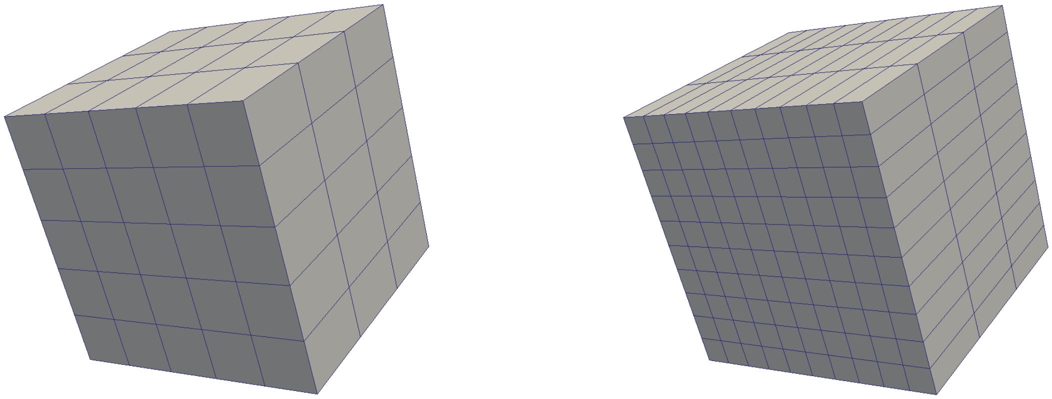
The Euler equations are solved at Mach number of $Ma = 0.4$. The computational domain of this study is a cube of length $[L_x, L_y, L_z] = [20, 20, 20]$, with the center of vortex initialized at the coordinate origin. The boundary conditions are specified as a periodic in all directions. The number of elements in the z-direction is always kept at 3, to make the domain consistent, and different numbers of elements in the x and y directions are used, which are shown in Figure 4.1. The classical four stage fourth order Runge-Kutta scheme is used for time discretization. The simulation is run for $4T_{char}$, and the L_2 norm of the error is computed. This case is investigated with and without using the filtering operator of different strengths to explore the influence of filtering on accuracy. The solution and flux points are located at tensor products of Gauss points, and a Rusanov Riemann solver is used.

4.1.3 Results

An example density contour of the isentropic vortex case is given in Figure 4.2. The filtered and non-filtered results of different filtering functions are given in Table 4.1. In this table N_e is the number of elements, Err is the L_2 norm of the error and is computed as the difference between the numerical and exact solutions, and OoA stands for the Order of Accuracy.

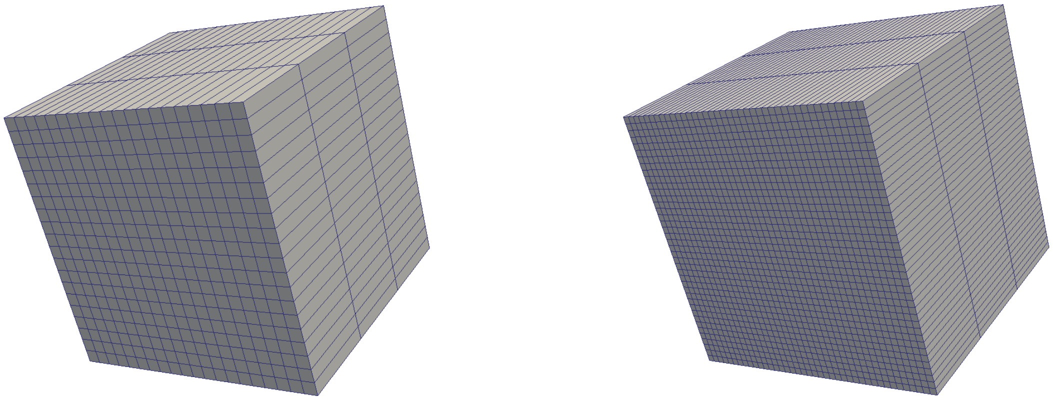
4.1.4 Discussion

From Table 4.1 we observe that the filtering operator does not significantly degrade the solution accuracy. In fact, in some cases accuracy is improved when the filter is on. Furthermore, all schemes maintain the expected order of accuracy of $\mathcal{P} + 1$ with the filter. Hence, we can conclude that these optimal stabilizing filters can maintain the accuracy of the FR approach.



(a) 5 elements.

(b) 10 elements.



(c) 20 elements.

(d) 40 elements.

Figure 4.1. The mesh used for studying the isentropic vortex.

Table 4.1. The order of accuracy of the isentropic vortex simulation.

Non-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	4.35e-2	—	5	3.00e-2	—	5	2.08e-2	—
10	1.25e-2	1.7961	10	1.48e-3	4.3395	10	5.33e-4	5.2849
20	4.28e-4	4.8710	20	2.70e-5	5.7777	20	6.34e-6	6.3938
40	1.42e-5	**4.9148**	40	8.01e-7	**5.0761**	40	7.75e-8	**6.3539**
$Ma = 0.1$-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	4.35e-2	—	5	3.00e-2	—	5	2.14e-2	—
10	1.25e-2	1.7961	10	1.48e-3	4.3372	10	5.36e-4	5.3189
20	4.28e-4	4.8710	20	2.70e-5	5.7781	20	6.28e-6	6.4156
40	1.42e-5	**4.9148**	40	8.01e-7	**5.0761**	40	7.71e-8	**6.3478**
$Ma = 0.2$-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	6.23e-2	—	5	1.71e-2	—	5	5.00e-2	—
10	1.05e-2	2.5637	10	2.23e-3	2.9437	10	2.76e-4	7.4983
20	3.42e-4	4.9428	20	2.31e-5	6.5915	20	5.66e-6	5.6111
40	1.63e-5	**4.3943**	40	8.59e-7	**4.7483**	40	8.77e-8	**6.0117**
$Ma = 0.3$-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	6.23e-2	—	5	1.70e-2	—	5	5.33e-2	—
10	1.05e-2	2.5640	10	2.22e-3	2.9314	10	2.72e-4	7.6154
20	3.42e-4	4.9429	20	2.31e-5	6.5907	20	5.68e-6	5.5790
40	1.63e-5	**4.3943**	40	8.59e-7	**4.7480**	40	8.71e-8	**6.0283**
$Ma = 0.4$-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	6.23e-2	—	5	1.69e-2	—	5	5.35e-2	—
10	1.05e-2	2.5635	10	2.22e-3	2.9257	10	2.71e-4	7.6244
20	3.42e-4	4.9427	20	2.31e-5	6.5903	20	5.71e-6	5.5704
40	1.63e-5	**4.3943**	40	8.59e-7	**4.7479**	40	8.68e-8	**6.0397**
$Ma = 0.5$-Filtered Simulation								
$\mathcal{P} = 3$			$\mathcal{P} = 4$			$\mathcal{P} = 5$		
N_e	*Err*	*OoA*	N_e	*Err*	*OoA*	N_e	*Err*	*OoA*
5	6.23e-2	—	5	1.57e-2	—	5	5.37e-2	—
10	1.05e-2	2.5634	10	2.16e-3	2.8663	10	2.63e-4	7.6754
20	3.42e-4	4.9427	20	2.28e-5	6.5624	20	5.65e-6	5.5397
40	1.63e-5	**4.3943**	40	8.57e-7	**4.7347**	40	8.63e-8	**6.0321**

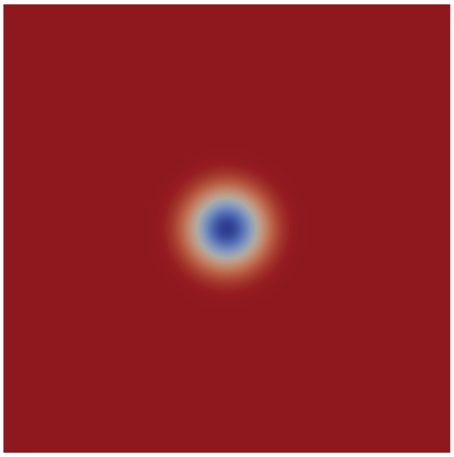


Figure 4.2. Density contours of the isentropic vortex.

4.2 The Taylor Green Vortex

4.2.1 Introduction

In order to study the suitability of the proposed filters for large eddy simulation, the Navier-Stokes equations are solved using the Taylor-Green Vortex case. The rate of kinetic energy dissipation along with enstrophy are computed to study LES accuracy. The temporal evolution of total kinetic energy integrated over the domain is

$$E_k = \frac{1}{\rho_0 \Omega} \int_{\Omega} \rho \frac{\boldsymbol{V} \cdot \boldsymbol{V}}{2} d\Omega, \tag{4.6}$$

where E_k is the total kinetic energy, Ω is the volume of the domain, and $\boldsymbol{V}$ is the velocity vector. The energy-based dissipation rate is defined as

$$\epsilon(E_k) = -\frac{dE_k}{dt}. \tag{4.7}$$

The temporal evolution of enstrophy is

$$\varepsilon = \frac{1}{\rho_0 \Omega} \int_{\Omega} \rho \frac{\boldsymbol{\omega} \cdot \boldsymbol{\omega}}{2} d\Omega, \tag{4.8}$$

where $\boldsymbol{\omega}$ is the vorticity. And the vorticity-based dissipation rate is [36]

$$\epsilon(\varepsilon) = \frac{2\mu}{\rho} \varepsilon. \tag{4.9}$$

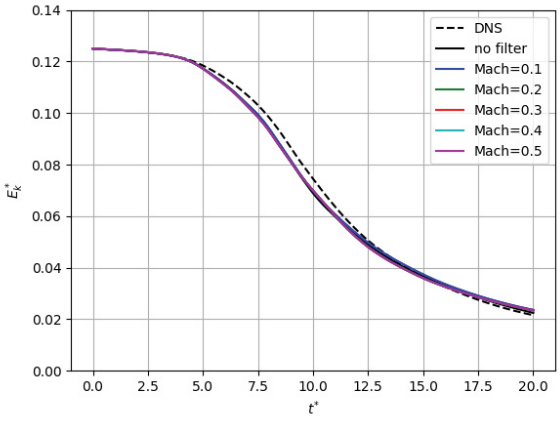
In the incompressible limit, the difference between the physical dissipation, $\epsilon(\varepsilon)$, and the observed dissipation, $\epsilon(E_k)$, is due to numerical error.

4.2.2 Computational Details

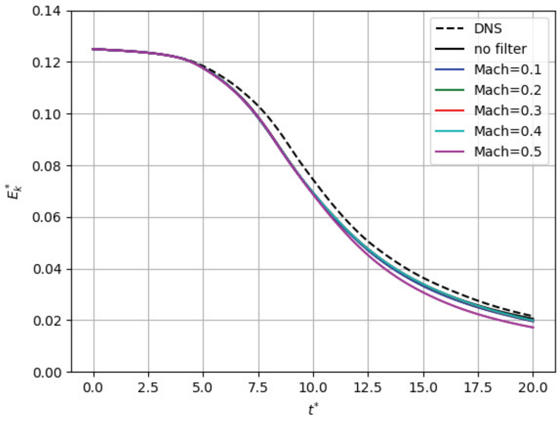
The vortices are initialized at $Re = 1600$ based on the length scale L and velocity scale U_0, and the Mach number of $Ma = 0.1$. The domain is a periodic cube of dimensions $0 \leq x, y, z \leq 2\pi$, with nominally 64^3 total number of DoF. Each simulation is run with $\mathcal{P} = 3, 4$ and 5 using different strengths of the filtering function optimized for $Ma = 0.1, 0.2, 0.3, 0.4$, and 0.5 along with a non-filtered simulation. The results are compared to the DNS of [36]. The solution and flux points are located at tensor products of Gauss points, and a Rusanov and the second method of Bassi and Rebay (BR2) is used for the common viscous flux.

4.2.3 Results

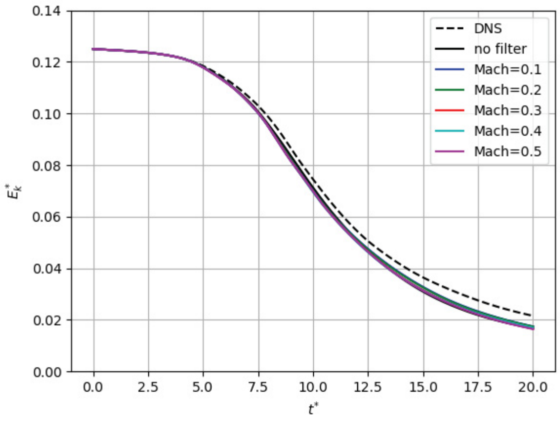
Plots of the kinetic energy evolution in time, the rate of E_k dissipation based on both E_k and enstrophy for different solution polynomial degrees of $\mathcal{P} = 3, 4$, and 5, using different filtering functions are given in Figures 4.3-4.5. The kinetic energy spectrum of high wavenumbers with different strengths of the filtering operator along with the non-filtered simulation are computed utilizing a spectral code by F. Navah [37], and is shown in Figure 4.6. Also isosurfaces of q-criterion for the TGV at $20T_{char}$ for different solution polynomial degrees are shown, with and without the filtering operator, in Figure 4.7.



(a) $\mathcal{P} = 3$

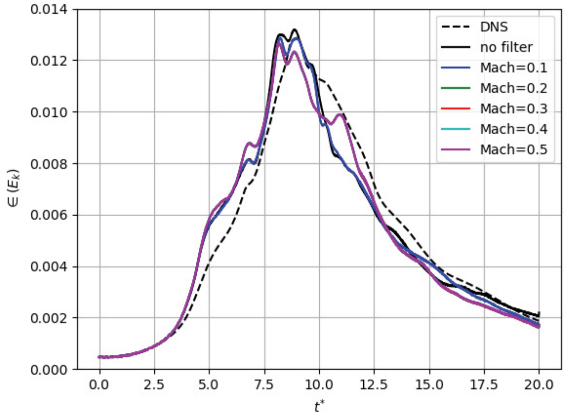


(b) $\mathcal{P} = 4$

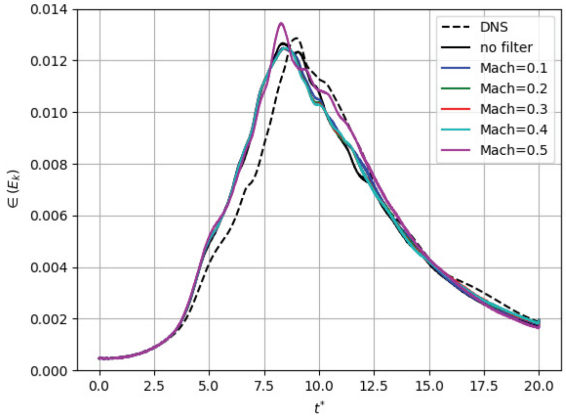


(c) $\mathcal{P} = 5$

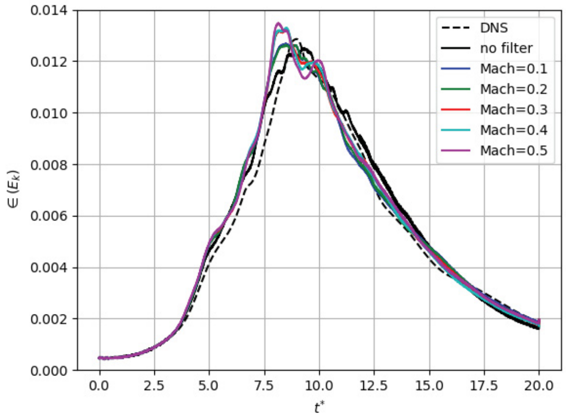
Figure 4.3. The evolution of kinetic energy in time for different solution polynomials.



(a) $\mathcal{P} = 3$

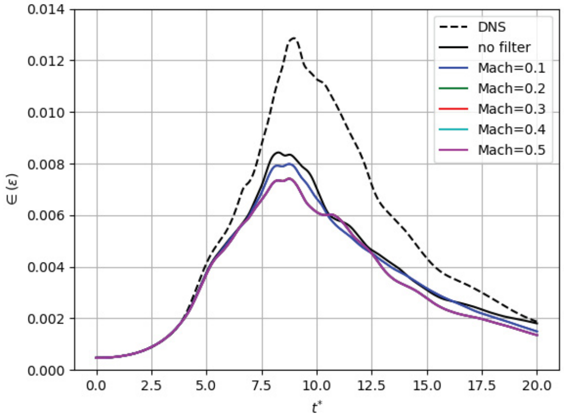


(b) $\mathcal{P} = 4$

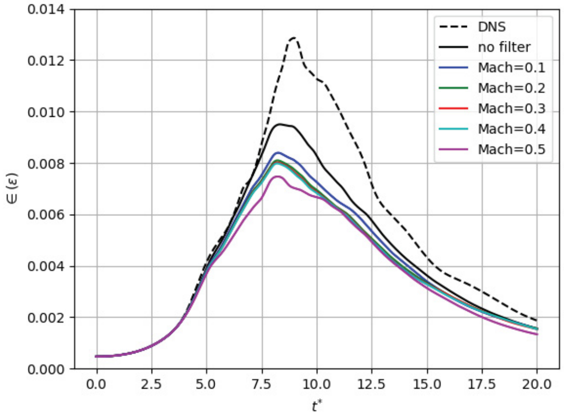


(c) $\mathcal{P} = 5$

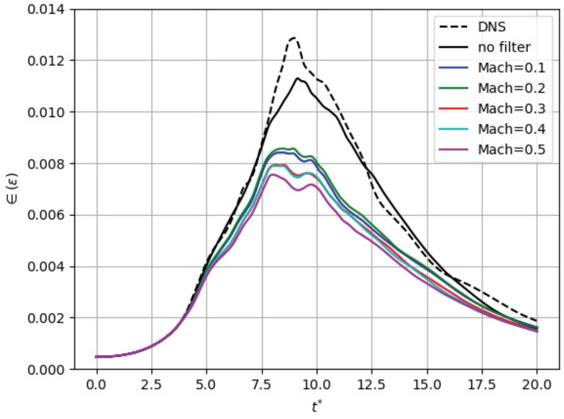
Figure 4.4. The energy-based rate of kinetic energy dissipation for different solution polynomials.



(a) $\mathcal{P} = 3$

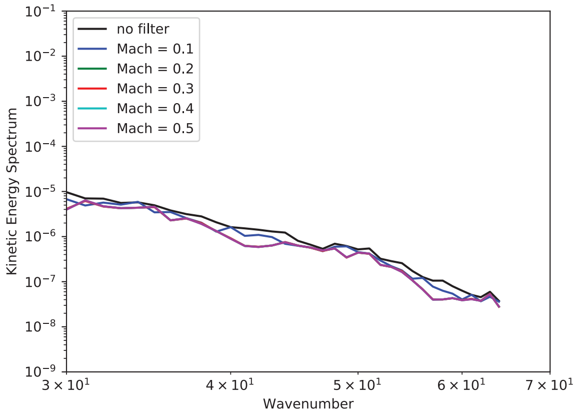


(b) $\mathcal{P} = 4$

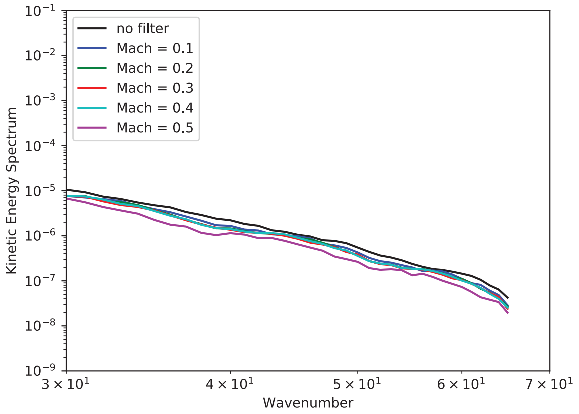


(c) $\mathcal{P} = 5$

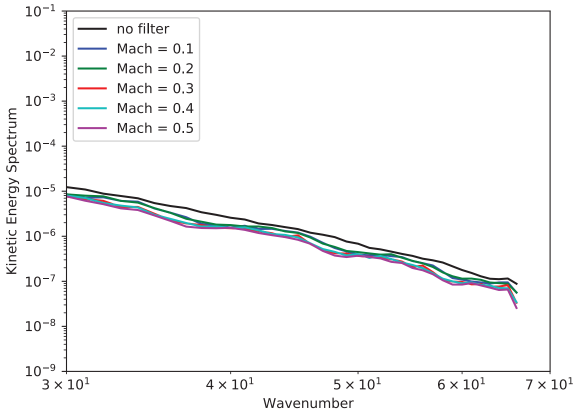
Figure 4.5. The enstrophy-based rate of kinetic energy dissipation for different solution polynomials.



(a) $\mathcal{P} = 3$

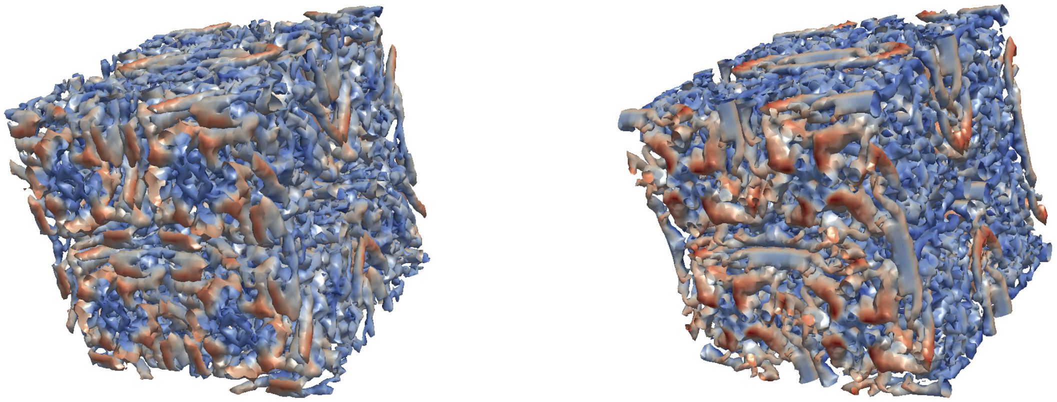


(b) $\mathcal{P} = 4$



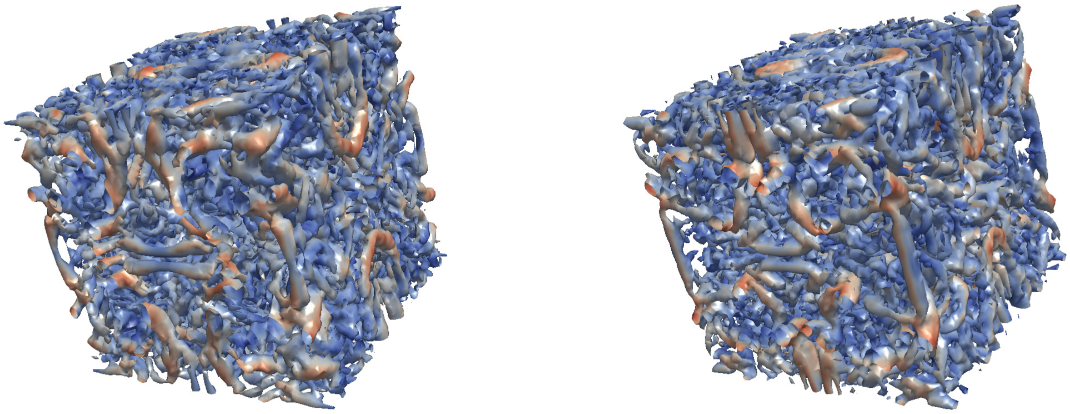
(c) $\mathcal{P} = 5$

Figure 4.6. The kinetic energy spectrum of high wavenumbers at $20T_{char}$ for different solution polynomial degrees.



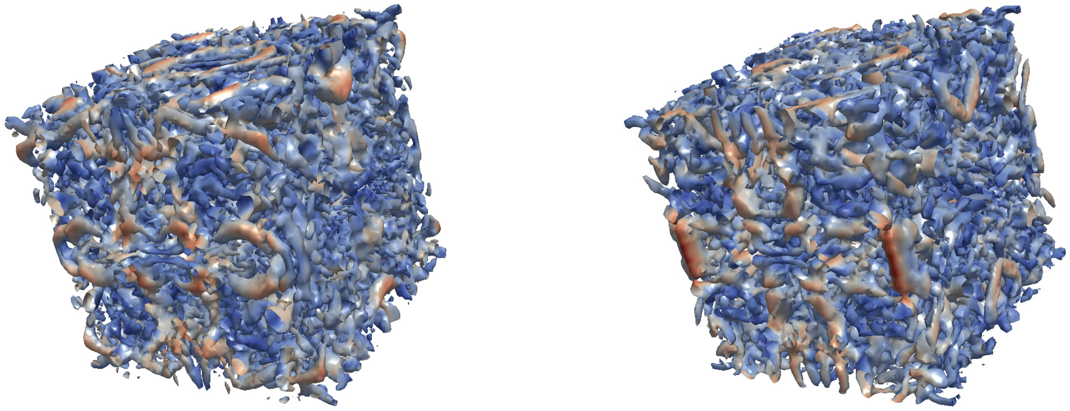
(a) $\mathcal{P} = 3$, without filtering.

(b) $\mathcal{P} = 3$, with the strongest filtering.



(c) $\mathcal{P} = 4$, without filtering.

(d) $\mathcal{P} = 4$, with the strongest filtering.



(e) $\mathcal{P} = 5$, without filtering.

(f) $\mathcal{P} = 5$, with the strongest filtering.

Figure 4.7. Isosurfaces of q-criterion for the TGV at $20T_{char}$ for different polynomial degrees.

4.2.4 Discussion

In Figure 4.3, it can be seen that the dissipation of E_k is slightly higher for the filtered simulation, and the E_k is always less than the DNS results except for $\mathcal{P} = 3$. The higher dissipation of a filtered solution is expected, since the filtering operator is removing high wavenumber components of the flow field. The reason that for $\mathcal{P} = 3$ there is more energy than the DNS result may be that the truncation error of the spatial discretization for this mesh resolution was not enough to model the unresolved subgrid-scale structures.

The energy-based rate of E_k dissipation is shown in Figure 4.4. This rate for filtered simulations is expected to be higher than that of non-filtered ones. However, a lower rate is observed in some regions. The reason for the lower energy dissipation rate in some regions is that, since the energy has been over-dissipated so far, there is less energy present in the filtered simulation at later times. It is observed in this figure that the dissipation of E_k is higher for the filtered simulations.

Figure 4.5 shows that the enstrophy-based E_k dissipation is under-predicted in the filtered simulations. Based on Equation 4.8, the small scale structures are enstrophy dominated structures in the turbulent flow, which are filtered when the filtering operator is active. Hence, the lower enstrophy observed in the filtered simulations.

Since the filtering operator dissipates the energy of the higher modes, there will be less energy in high wavenumbers for filtered simulations. This is depicted in Figure 4.6, where the kinetic energy spectrum of high wavenumbers is given at $20T_{char}$. It can be seen that for stronger filtering operators, there is reduced energy in higher modes.

Finally, from Figure 4.7, the isosurfaces of q-criterion for the TGV, it is evident that when the filtering operator is applied, the general scale of the turbulent structures is larger than the non-filtered simulations.

In summary, we observe that the filtering operators tend to increase the amount of numerical dissipation for all polynomial degrees. This is primarily due to damping of high-frequency modes in the solution. Nevertheless, the overall accuracy is not impacted significantly, considering the overall stability benefits of using the optimized filters.

Chapter 5

Numerical Examples

In this chapter we studied two simulations, a previously unstable turbulent channel test case, along with an airfoil at a high angle of attack. These two simulations are discussed further in the next sections. The objective is to explore the stabilization properties of the proposed filters, and their accuracy.

5.1 Turbulent Channel

5.1.1 Introduction

Fully-developed turbulent channel flow is studied as validation for wall-bounded turbulent flows, due to its simple geometry and boundary conditions. It consists of a flow driven by a pressure gradient, dP/dx. The geometry of the turbulent channel is shown in Figure 5.1. The fully-developed assumption means that all statistical properties of the flow are independent of time. In turbulent channel studies, it is conventional to define the friction velocity as

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}, \tag{5.1}$$

where τ_w is the wall shear stress.

There are different Reynolds number definitions used in the context of the turbulent channel, which include

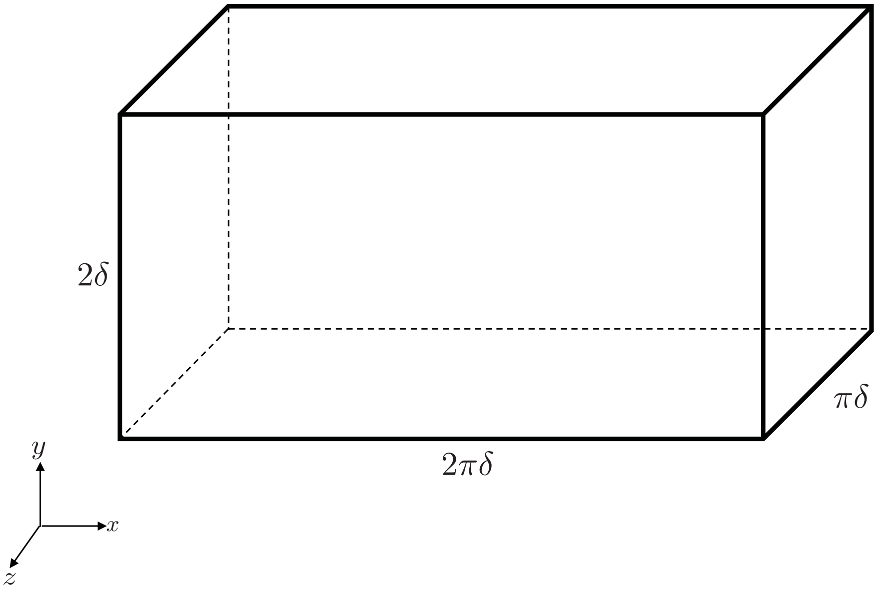


Figure 5.1. Geometry of the turbulent channel.

- Reynolds number based on the friction velocity

$$Re_\tau = \frac{u_\tau \delta}{\nu}, \tag{5.2}$$

- Reynolds number based on the centerline velocity

$$Re_c = \frac{u_c \delta}{\nu}, \tag{5.3}$$

- Reynolds number based on the bulk velocity

$$Re_b = \frac{u_b 2\delta}{\nu}. \tag{5.4}$$

The wall-normal direction, y, is normalized in two different ways. One is a normalization based on the channel half-width, δ, and the other is a normalization based on the friction velocity, u_τ, both of which are given

$$y^* = \frac{y}{\delta}, \tag{5.5}$$

$$y^+ = \frac{u_\tau y}{\nu}. \tag{5.6}$$

As shown in Equation 2.12, the total shear stress, τ, is a summation of viscous shear stress, $\overline{\tau}$, and Reynolds shear stress, τ^R. The sum of these two stresses is always constant, and as we move closer to the wall, the viscous stress becomes larger. So, for $y^+ \gg 1$ the viscous stress is negligible and $\tau \approx \tau^R$. The Reynolds stresses can be written as a tensor,

$$\tau_{ij}^R = \rho\overline{u_i' u_j'} = \rho \begin{bmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{v'u'} & \overline{v'v'} & \overline{v'w'} \\ \overline{w'u'} & \overline{w'v'} & \overline{w'w'} \end{bmatrix} \quad . \tag{5.7}$$

The turbulent channel is divided into 4 different regions [1]:

1. *Viscous sublayer*, $y^+ < 5$, where the viscous effects are dominant.
2. *Inner region*, $y^* \ll 1$, or close the wall.
3. *Outer region*, $y^+ \gg 1$, where the Reynolds stresses are dominant.
4. *Overlap region*, $y^* \ll 1$ and $y^+ \gg 1$, where the Reynolds stresses are dominant, however, the flow is close to the wall.

The different layers of turbulent channel flow are depicted in Figure 5.2, and are [1]

- *Viscous Sublayer*, $y^+ < 5$. Since the flow is viscous, there is a no-slip boundary condition, and the velocity at the wall is zero. So, as the flow gets closer to the wall, the Reynolds number will decrease, and at some point, the flow becomes laminar. The flow in the viscous sublayer is laminar, since viscosity effects are dominant.
- *Buffer Layer*, $5 < y^+ < 40$. The flow is turbulent, but both of the viscous and Reynolds stresses are important. Small scale turbulent structures are generated in this layer, so it is often called the turbulence generation layer.

- *Velocity Defect Law*, $y^+ \gg 1$ ($y^+ > 40$). This law says that in the region where viscous effects are negligible, the fluctuating part of the velocity is a function of wall distance only,

  $$\frac{u'}{u_\tau} = f(y^*). \tag{5.8}$$

- *Law of the Wall*, $y^* \ll 1$. This law says that in the near wall region, the mean velocity of the turbulent flow is a function of y^+ only,

  $$\frac{\overline{u}}{u_\tau} = g(y^+). \tag{5.9}$$

- *Log-Law of the Wall*, $y^* \ll 1,\ y^+ \gg 1$. This law comes into effect when $Re \gg 1$ and consequently there will be an overlap region in the flow. In this region, since $y^* \ll 1$, the total shear stress, τ, is constant, and since $y^+ \gg 1$ ($y^+ > 40$), the viscous shear stress, $\overline{\tau}$, is negligible [1]. This law can be expressed as

  $$\frac{\overline{u}}{u_\tau} = \frac{1}{\kappa} lny^+ + A, \tag{5.10}$$
  $$\frac{u'}{u_\tau} = -\frac{1}{\kappa} lny^* + B, \tag{5.11}$$

  where κ is the Karman's constant which is usually in the range of 0.38 to 0.43, and A and B are constants [1].

In 1987, Kim, Moin and Moser [38] performed DNS of turbulent channel flow. In the present research, ILES is performed and compared to these DNS results.

5.1.2 Computational Details

The domain is a cube of $[L_x, L_y, L_z] = [2\pi\delta, 2\delta, \pi\delta]$ with periodic boundary conditions in the streamwise (x) and spanwise (z) directions and a no-slip boundary condition at the walls, where $\delta = 1$ is the channel half width. The initial conditions are Mach number of $Ma = 0.3$ and a friction Reynolds $Re_\tau = 395$. Given the one-dimensional behavior of the statistically steady flow, it is expected that a shear stress will develop at the wall due to the no-slip boundary condition and will

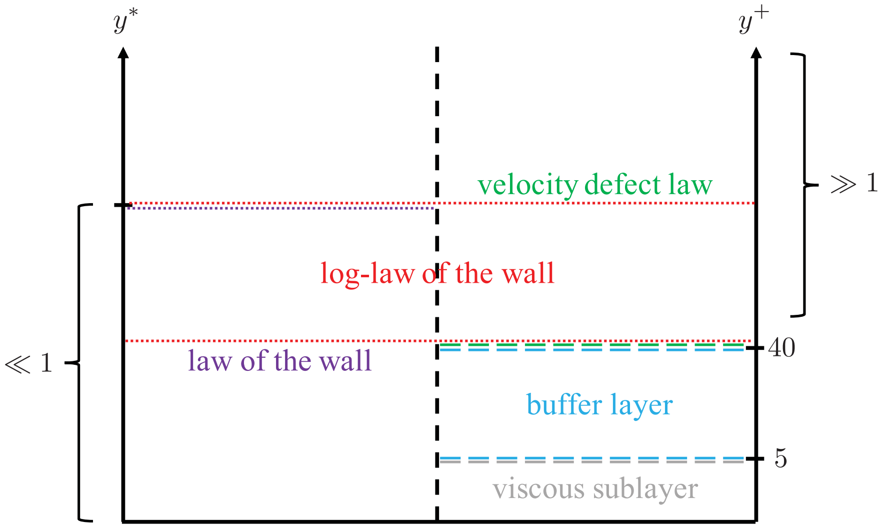


Figure 5.2. Different layers of turbulent channel flow.

Table 5.1. Parameters of the simulations where N_{ei} is the number of elements in i^{th}-direction and Δi^+ is the non-dimesional distance of the first grid point off the boundary in i^{th}-direction.

$\mathcal{P}$	*Filter*	Re_τ	L_x/δ	L_y/δ	L_z/δ	T_f	Δx^+	Δy^+	Δz^+	N_{ex}	N_{ey}	N_{ez}	*DoF*
3	off	395	2π	2	π	200	25.85	0.91	17.23	24	18	18	497664
3	on	395	2π	2	π	200	25.85	0.91	17.23	24	18	18	497664
4	on	395	2π	2	π	200	20.70	0.73	13.78	24	18	18	972000
5	on	395	2π	2	π	200	25.85	0.91	17.23	16	12	12	497664

equal the pressure gradient in magnitude. The grid points in the y-direction are computed using the following hyperbolic function [39]

$$y_j = \frac{1}{2\alpha}\tanh\left[\left(-1 + \frac{2j}{DoF_y}\right)\tanh^{-1}\alpha\right] + 0.5, \tag{5.12}$$

where $\alpha = 0.96$ is the stretching factor and DoF_y is the number of solution points in the y direction. The present simulations are summarized in Table 5.1.

In this study, we used 3/8 of the total number of DoF required for DNS in each direction to simulate the turbulent channel with polynomial degrees of $\mathcal{P} = 3, 4$, and 5. This simulation is stable for $\mathcal{P} = 3$, so we ran it with and without the filtering operator to compare the filtered simulation with a non-filtered one, and investigate the performance of the filtering operator. Since the simulations with $\mathcal{P} = 4$ and 5 are unstable without filtering, the filtering operator is applied for these simulations to stabilize them. In the case of $\mathcal{P} = 4$, the filtering operator fails to stabilize the simulation, so the number of DoF in all directions was increased by 20%. However, the filtering operator makes the

$\mathcal{P} = 5$ case stable for the original number of DoF ($= \frac{3}{8} DoF_{DNS}$).

5.1.3 Results

The power spectral density for each simulation is given in Figures 5.3-5.5 for 5 different distances from the wall. The mean velocity, Reynolds stresses, and root-mean-squared velocity fluctuations are then given in Figures 5.6, 5.7, and 5.8, respectively, for different polynomial degrees.

5.1.4 Discussion

From the power spectral density plots for $\mathcal{P} = 3$, shown in Figure 5.3, it can be seen that for the filtered simulation, only the higher modes have reduced power relative to the non-filtered simulations, which shows that filtering only affects the higher modes, as by design. Figures 5.4 and 5.5, show the power spectral density for $\mathcal{P} = 4$ and 5 at different locations, in which less energy is observed at high wavenumbers.

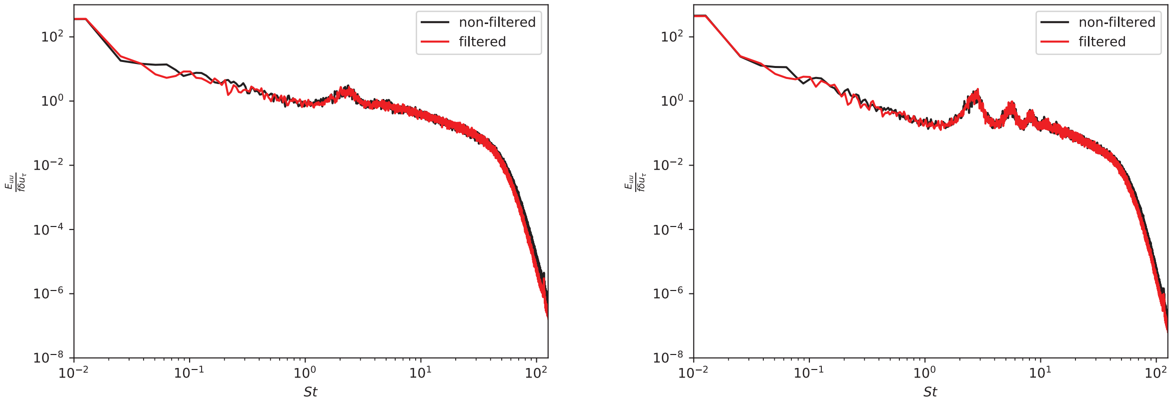
In Figure 5.6a, we can see excellent agreement between the mean velocity profile of the filtered and non-filtered simulations for $\mathcal{P} = 3$. Similar behavior is observed in Figures 5.6b and 5.6c between the filtered simulation and the DNS results for $\mathcal{P} = 4$ and $\mathcal{P} = 5$.

The Reynolds stresses shown in Figures 5.7 and 5.8 also show good agreement between the ILES results and the DNS. More accurate results are also obtained with filtering in the near wall region compared to the non-filtered simulations, as shown in Figures 5.7a and 5.8a. The maximum streamwise velocity fluctuation occurs at $y^+ \approx 12$ for both filtered and non-filtered simulations, consistent with the DNS, as observed in Figure 5.8a.

In summary we found that the filtering operators were suitable for ILES of turbulent channel flow. They stabilized two otherwise unstable simulations, and produced results consistent with the reference DNS data for all polynomial degrees.

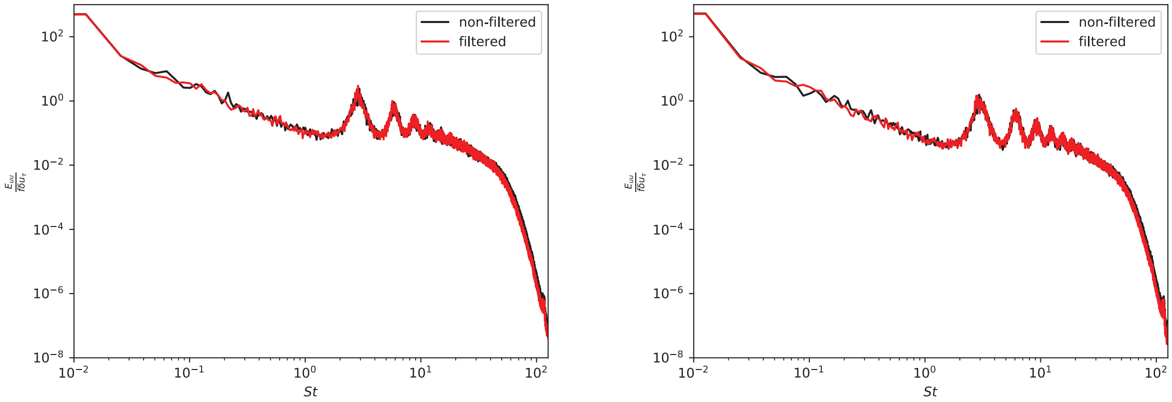
5.2 Airfoil

In this section, a NACA0020 airfoil at high angle of attack is simulated both with and without filtering. The effects of applying the filtering operator are investigated, and it is shown that the



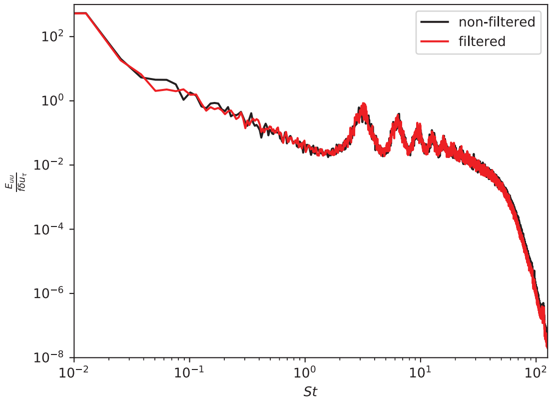
(a) $y^* = 0.1$

(b) $y^* = 0.3$



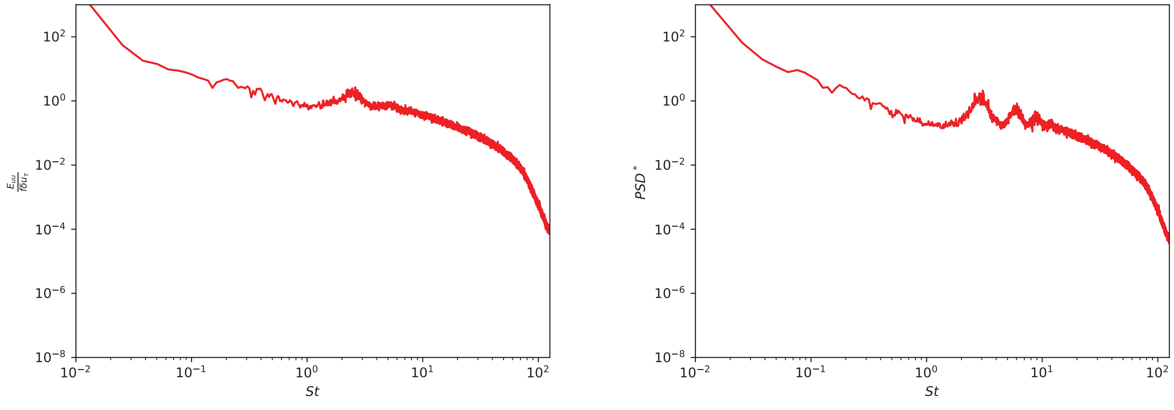
(c) $y^* = 0.5$

(d) $y^* = 0.7$



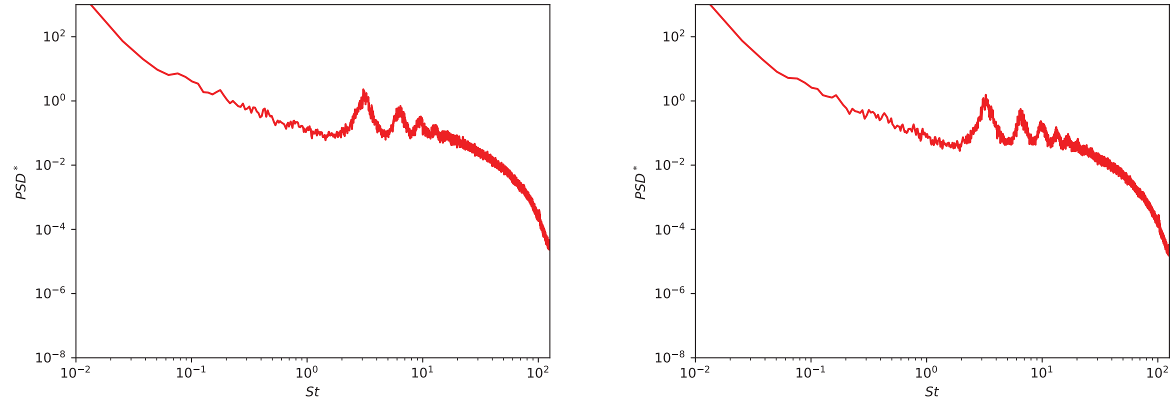
(e) $y^* = 0.9$

Figure 5.3. Power spectral density at different locations for $\mathcal{P} = 3$.



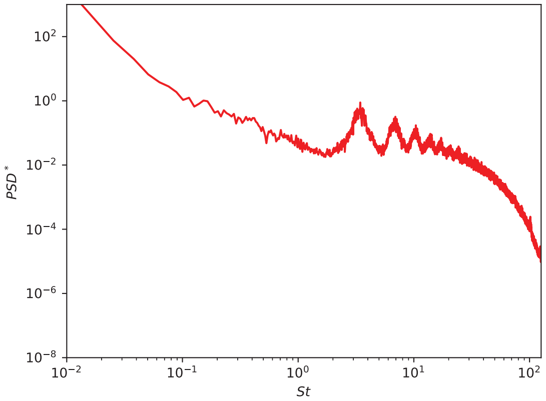
(a) $y^* = 0.1$

(b) $y^* = 0.3$



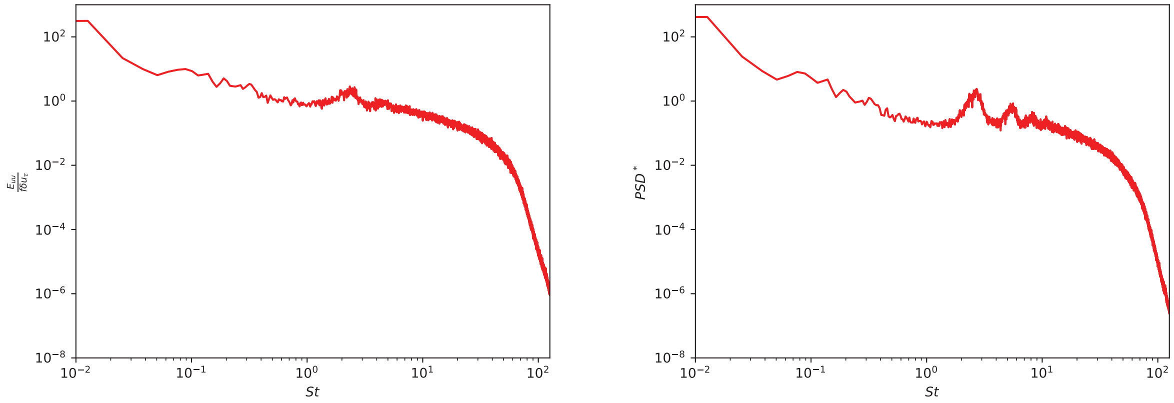
(c) $y^* = 0.5$

(d) $y^* = 0.7$



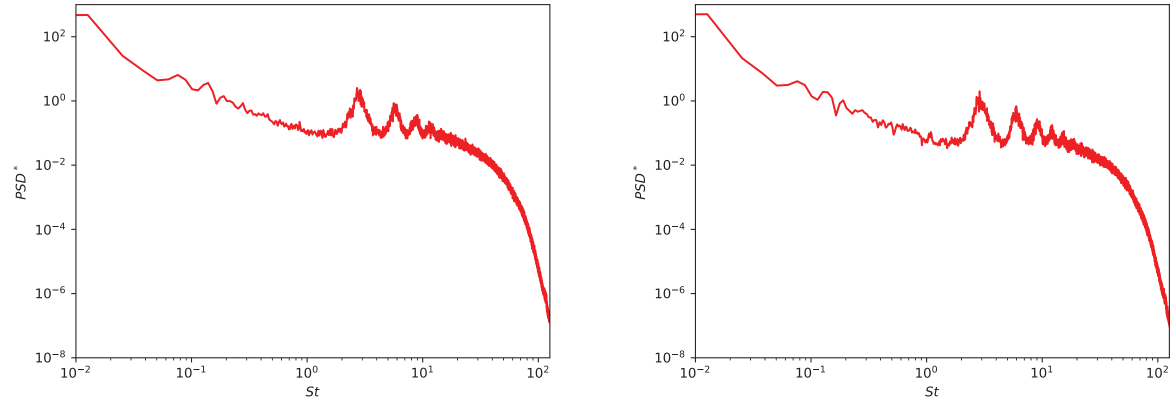
(e) $y^* = 0.9$

Figure 5.4. Power spectral density at different locations for $\mathcal{P} = 4$.



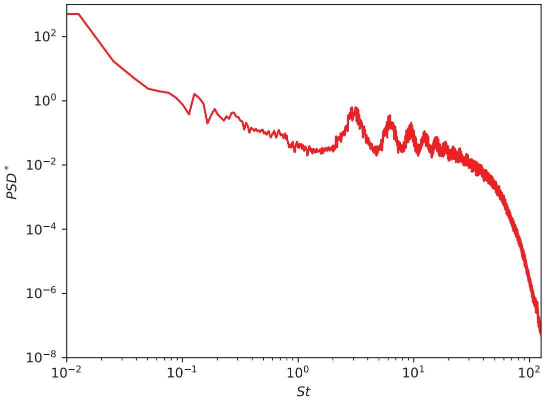
(a) $y^* = 0.1$

(b) $y^* = 0.3$



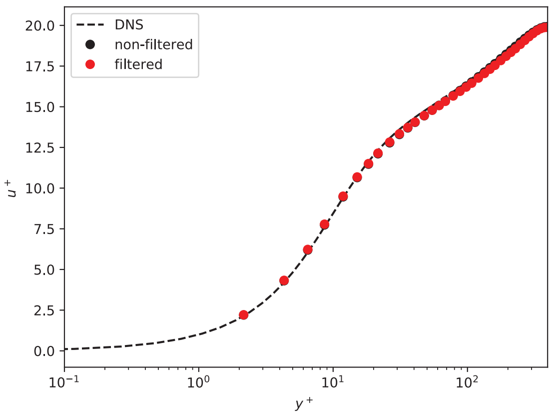
(c) $y^* = 0.5$

(d) $y^* = 0.7$

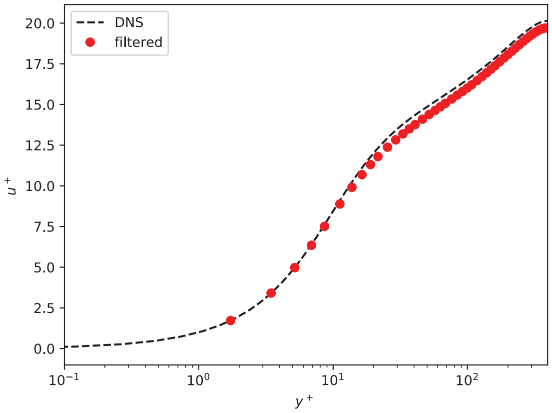


(e) $y^* = 0.9$

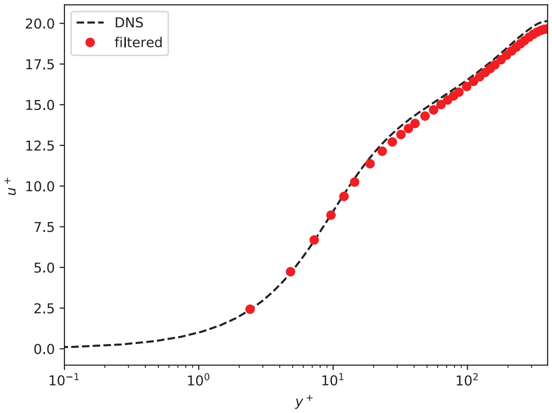
Figure 5.5. Power spectral density at different locations for $\mathcal{P} = 5$.



(a) $\mathcal{P} = 3$

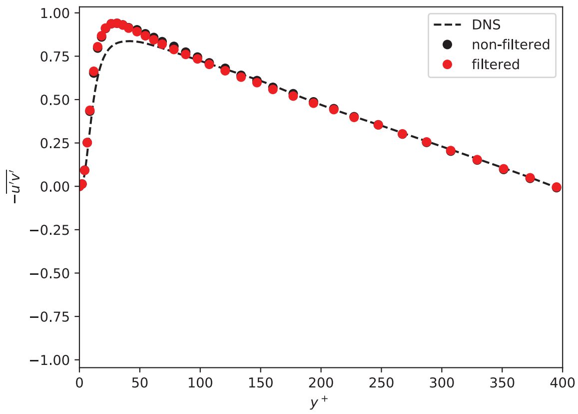


(b) $\mathcal{P} = 4$

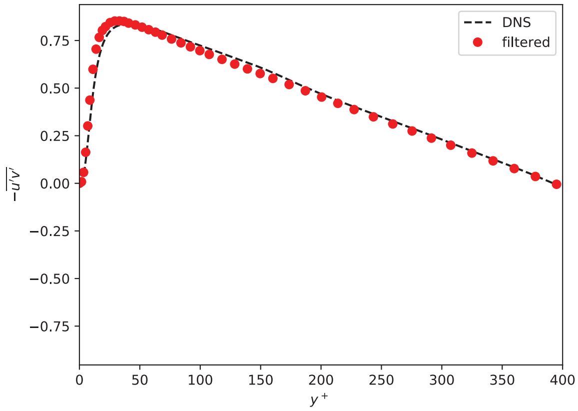


(c) $\mathcal{P} = 5$

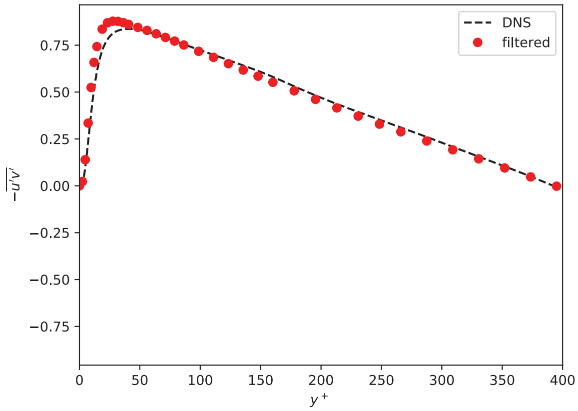
Figure 5.6. Mean velocity profile for different solution polynomial degrees.



(a) $\mathcal{P} = 3$

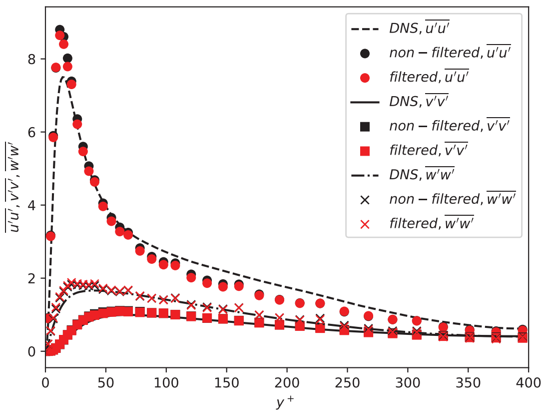


(b) $\mathcal{P} = 4$

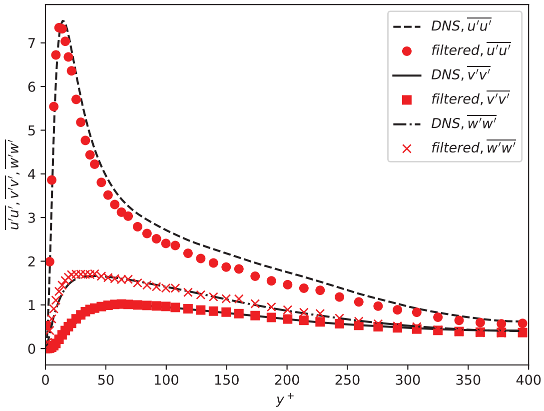


(c) $\mathcal{P} = 5$

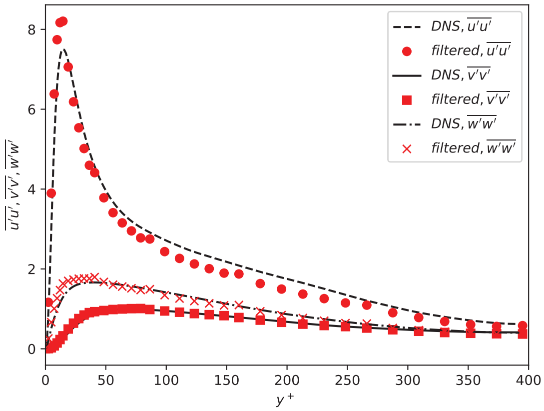
Figure 5.7. Reynolds shear stresses for different solution polynomial degrees.



(a) $\mathcal{P} = 3$



(b) $\mathcal{P} = 4$



(c) $\mathcal{P} = 5$

Figure 5.8. Root-mean-squared velocity fluctuations for different solution polynomial degrees.

filtered simulation is in excellent agreement with the non-filtered one. These results are then compared to DNS results [40], also showing good agreement.

5.2.1 Introduction

Flow over an airfoil at moderate Reynolds numbers, will form a boundary layer close to the surface of the airfoil. In the boundary layer, viscous effects are dominant, and the flow can be either laminar or turbulent depending on the transition location. The velocity of flow at the surface of the airfoil is zero due to the no-slip boundary condition, and it increases in the boundary layer until it reaches the upstream velocity at the edge of the boundary layer. Flow separation could also occur in the boundary layer due to an adverse pressure gradient. This separation forms a separation bubble, in which the fluid flows in the opposite direction of upstream flow. The separated region leads to a phenomenon known as a stall. As angle of attack increases, the lift coefficient will increase as well up to a degree, at which point stall occurs. The maximum lift coefficient is at stall, and after that point, the drag coefficient increases whereas the lift coefficient starts to decrease. In the current study, the NACA0020 is simulated in a fully stalled configuration.

An interaction between an airfoil and a fluid will generate forces. The resultant force in the direction of and normal to upstream velocity are termed the drag, $\mathcal{D}$, and lift, $\mathcal{L}$, force, respectively [41]. Nondimensional lift and drag coefficients are defined as

$$C_L = \frac{\mathcal{L}}{\frac{1}{2}\rho U_{up}^2 A}, \tag{5.13}$$

$$C_D = \frac{\mathcal{D}}{\frac{1}{2}\rho U_{up}^2 A}, \tag{5.14}$$

where $\mathcal{L}$ and $\mathcal{D}$ are lift and drag forces, respectively, U_{up} is the upstream velocity, and A is the characteristic or planform area. Similarly a pressure coefficient can be defined,

$$C_P = \frac{P - P_0}{\frac{1}{2}\rho U_{up}^2}, \tag{5.15}$$

where P_0 is the reference pressure.

5.2.2 Computational Details

Flow over an NACA0020 airfoil at an angle of attack of $\alpha = 20$, Mach number of $Ma = 0.2$, and $Re = 20000$ is simulated. The second order P-ERK scheme is used for marching in time. ILES is performed, and the non-filtered and filtered simulations are compared to DNS results. The mesh used for this simulation is shown in Figure 5.9, which consists of 67330 hexahedral elements. The domain has a periodic span of $0.45c$, which is sufficient for span-wise decorrelation [40]. Furthermore, the first solution point off the wall is located at $y^+ = 0.7$, within the viscous sublayer. Each simulation is started with $\mathcal{P} = 1$, and then restarted at $\mathcal{P} = 3$ and run for 100 convective times. Statistical averages are computed over the final 80 convective times.

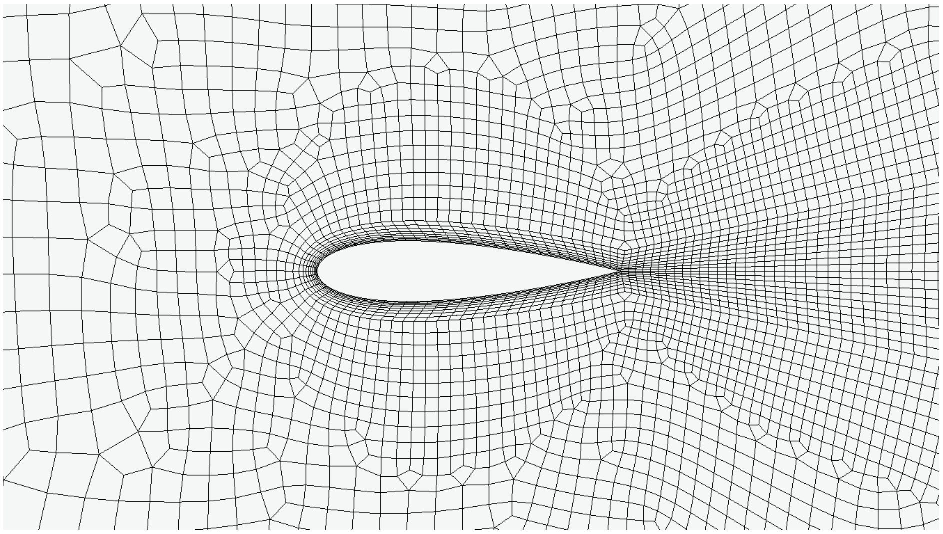


Figure 5.9. Mesh used for the simulation.

5.2.3 Results

The lift and drag coefficients evolution in time are given in Figure 5.10, along with the time-averaged pressure coefficient in Figure 5.11. The time-averaged C_L and C_D in the current study and those of DNS are given in Table 5.2. And the isosurfaces of q-criterion for both filtered and non-filtered simulations are shown in Figure 5.12.

---

[1] The data of this simulation was available in the lab and is not simulated by the author

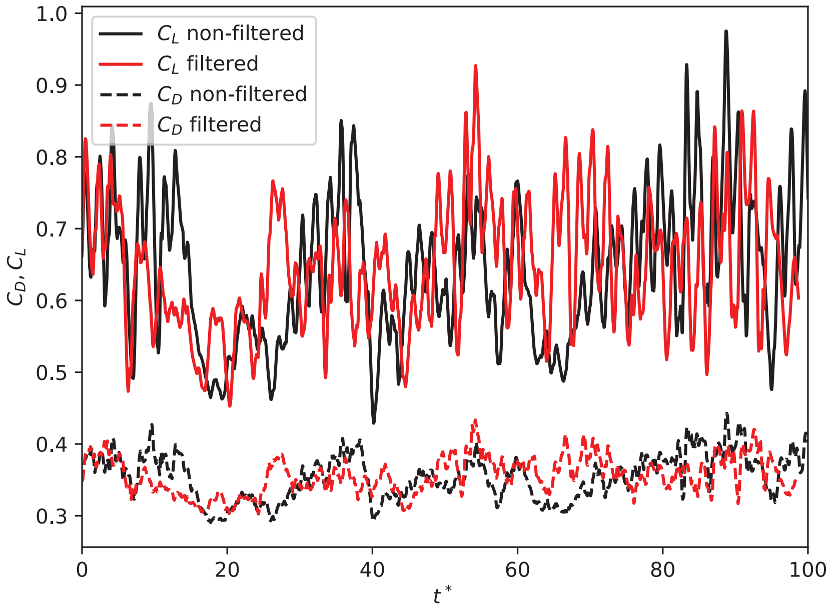


Figure 5.10. The lift and drag coefficients evolution in time.

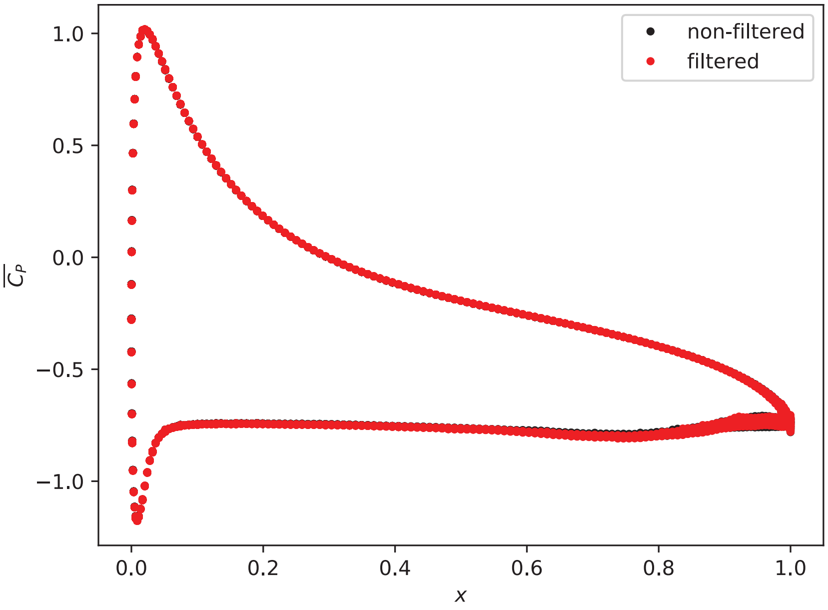
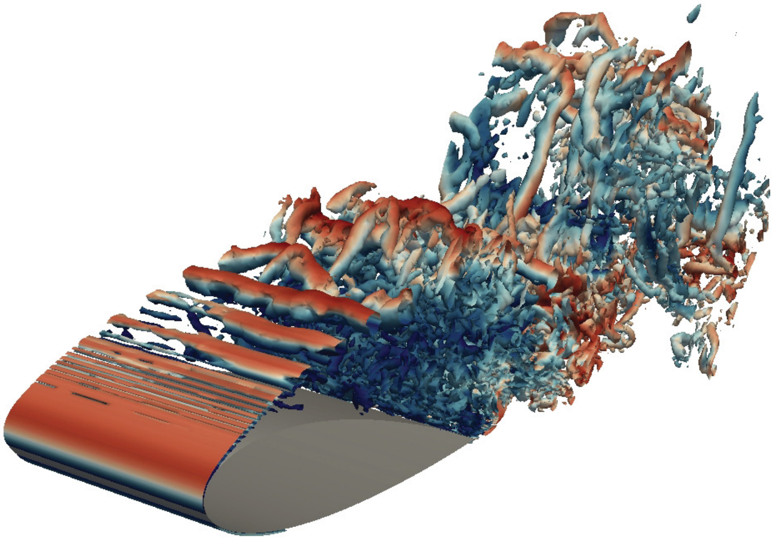
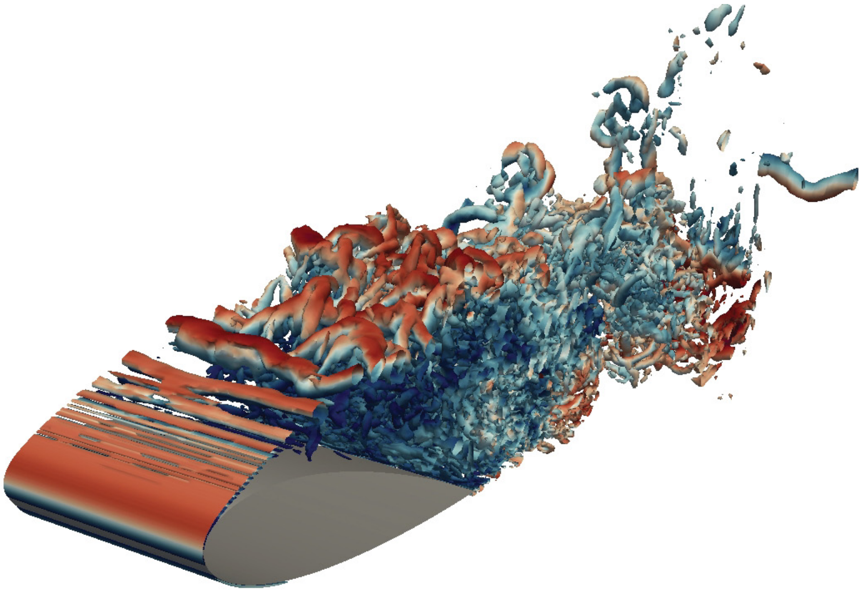


Figure 5.11. The time-averaged pressure coefficient on the surface of the airfoil.



(a) The non-filtered simulation.



(b) The filtered simulation.

Figure 5.12. The isosurfaces of q-criterion after $200T_{char}$.

Table 5.2. Comparison between the non-filtered, filtered, and DNS results.

Simulation	$\overline{C_L}$	$\overline{C_D}$
DNS [40]	0.64	0.35
non-filtered ILES[1]	0.639420	0.353248
filtered ILES	0.650771	0.357541

5.2.4 Discussion

There is good agreement between the lift and drag coefficients of the filtered and non-filtered simulations, as shown in Figure 5.10. The time-averaged values, $\overline{C_L}$ and $\overline{C_D}$, are compared to DNS results in Table 5.2, which shows excellent agreement between the DNS and both filtered and non-filtered ILES results. The time-averaged pressure coefficient for both the filtered and non-filtered simulations are shown in Figure 5.11, where the results are almost the same. As a conclusion, applying the filtering operator does not degrade the solution accuracy, and the results are in excellent agreement with the non-filtered simulation and reference DNS.

Finally, isosurfaces of q-criterion coloured by velocity magnitude are shown in Figure 5.12. Qualitatively, the large scale structures show similar behaviour, while spurious small scale structures appear to be damped in the filtered simulation.

Chapter 6

Conclusions and Future Work

In this research a new filtering operator, as a stabilizing technique, was proposed. Filtering operators of different strengths were obtained and validated. It was observed that all of the filtering operators preserve the order of accuracy for solution polynomial of degrees $\mathcal{P} = 3, 4$, and 5 while maintaining stability in the limit of an infinite Reynolds number TGV. In some cases, applying the filtering operator also improved the accuracy of the solution. More dissipation is observed in filtered simulations, which is due to adding dissipation to the highest resolved modes. Less energy is observed in these high-frequency modes since, the filtering operator is active there. Applying the filtering operator stabilized the turbulent channel simulation using a very coarse mesh, which was initially unstable with the same numbers of DoF without filtering. So, the filtering operator can be applied to stabilize the simulation, while the overall accuracy is not affected significantly.

For future work, the following are suggested.

1. Optimizing the filtering operator for higher polynomial degrees.
2. Since most engineering applications are at high Reynolds number, the need for optimal filtering operators for higher Reynolds number wall-bounded ILES is inevitable.
3. The filtering operator can be investigated with a combination of other stabilizing techniques, such as de-aliasing and limiters.
4. Shock capturing using the filtering operator can be studied.

5. In this study, the filtering operator is applied globally. One can consider implementing a filtering operator locally to filter only troubled-cells.

Appendices

Appendix A

Orthonormal Basis Functions for $\mathcal{P} = 3$ Hexagonal Elements

Table A.1. Orthonormal basis functions for hexagonal elements and $\mathcal{P} = 3$

i	orthonormal basis function corresponding to i^{th} solution point
1	0.3535
2	$0.6124x$
3	$0.6124y$
4	$0.6124z$
5	$1.1858x^2 - 0.3953$
6	$1.061xy$
7	$1.1858y^2 - 0.3953$
8	$1.0607xz$
9	$1.0607yz$
10	$1.1858z^2 - 0.3953$
11	$2.3385x^3 - 1.4031x$
12	$2.0540yx^2 - 0.6846y$
13	$2.0540xy^2 - 0.6846x$
14	$2.3385y^3 - 1.4031y$

15 $2.0540zx^2 - 0.6846z$

16 $1.8371xyz$

17 $2.0540zy^2 - 0.6846z$

18 $2.05340xz^2 - 0.6846x$

19 $2.05340yz^2 - 0.6846y$

20 $2.338535866733714z^3 - 1.403121520040229z$

21 $4.0505yx^3 - 2.4303xy$

22 $3.9775x^2y^2 - 1.3258x^2 - 1.3258y^2 + 0.4419$

23 $4.0505xy^3 - 2.4303xy$

24 $4.0505zx^3 - 2.4303xz$

25 $3.5576yzx^2 - 1.1858yz$

26 $3.5576xzy^2 - 1.1858xz$

27 $4.0505zy^3 - 2.4303yz$

28 $3.9775x^2z^2 - 1.3258x^2 - 1.3258z^2 + 0.4419$

29 $3.5576xyz^2 - 1.1858xy$

30 $3.9775y^2z^2 - 1.3258y^2 - 1.3258z^2 + 0.4419$

31 $4.0505xz^3 - 2.4303xz$

32 $4.0505yz^3 - 2.4303yz$

33 $7.8437x^3y^2 - 2.6146x^3 - 4.7062xy^2 + 1.5687x$

34 $7.8437x^2y^3 - 4.7062yx^2 - 2.6146y^3 + 1.5687y$

35 $7.0156yzx^3 - 4.2094xyz$

36 $6.8892zx^2y^2 - 2.2964zx^2 - 2.2964zy^2 + 0.7655z$

37 $7.0156xzy^3 - 4.2094xyz$

38 $7.8437x^3z^2 - 2.6146x^3 - 4.7062xz^2 + 1.5687x$

39 $6.8892yx^2z^2 - 2.2964yx^2 - 2.2964yz^2 + 0.7655y$

40 $6.8892xy^2z^2 - 2.2964xy^2 - 2.2964xz^2 + 0.7655x$

41 $7.8437y^3z^2 - 2.6146y^3 - 4.7062yz^2 + 1.5687y$

42 $7.8437x^2z^3 - 4.7062zx^2 - 2.6146z^3 + 1.5687z$

43 $7.0156xyz^3 - 4.2094xyz$

44 $7.8437y^2z^3 - 4.7062zy^2 - 2.6146z^3 + 1.5687z$

45 $15.4680x^3y^3 - 9.2808yx^3 - 9.2808xy^3 + 5.5685xy$

46 $13.5857zx^3y^2 - 4.5285zx^3 - 8.1514xzy^2 + 2.7171xz$

47 $13.5857zx^2y^3 - 8.1514yzx^2 - 4.5285zy^3 + 2.7171yz$

48 $13.5857yx^3z^2 - 4.5285yx^3 - 8.1514xyz^2 + 2.7171xy$

49 $13.3409x^2y^2z^2 - 4.4469x^2y^2 - 4.4469x^2z^2 + 1.4823x^2 - 4.4469y^2z^2 + 1.4823y^2 + 1.4823z^2 - 0.4941$

50 $13.5857xy^3z^2 - 4.5285xy^3 - 8.1514xyz^2 + 2.7171xy$

51 $15.4680x^3z^3 - 9.2808zx^3 - 9.2808xz^3 + 5.5685xz$

52 $13.5857yx^2z^3 - 8.1514yzx^2 - 4.5285yz^3 + 2.7171yz$

53 $13.5857xy^2z^3 - 8.1514xzy^2 - 4.5285xz^3 + 2.7171xz$

54 $15.4680y^3z^3 - 9.2808zy^3 - 9.2808yz^3 + 5.5685yz$

55 $26.7913zx^3y^3 - 16.0748yzx^3 - 16.0748xzy^3 + 9.6449xyz$

56 $26.3x^3y^2z^2 - 8.8x^3y^2 - 8.8x^3z^2 + 2.9x^3 - 15.8xy^2z^2 + 5.3xy^2 + 5.3xz^2 - 1.7x$

57 $26.3x^2y^3z^2 - 8.8x^2y^3 - 15.8yx^2z^2 + 5.3yx^2 - 8.8y^3z^2 + 2.9y^3 + 5.3yz^2 - 1.7y$

58 $26.7913yx^3z^3 - 16.0748yzx^3 - 16.0748xyz^3 + 9.6449xyz$

59 $26.3x^2y^2z^3 - 15.8zx^2y^2 - 8.8x^2z^3 + 5.2zx^2 - 8.8y^2z^3 + 5.3zy^2 + 2.9z^3 - 1.7z$

60 $26.7913xy^3z^3 - 16.0748xzy^3 - 16.0748xyz^3 + 9.6449xyz$

61 $51.9x^3y^3z^2 - 17.3x^3y^3 - 31.1yx^3z^2 + 10.4yx^3 - 31.1xy^3z^2 + 10.4xy^3 + 18.7xyz^2 - 6.2xy$

62 $51.9x^3y^2z^3 - 31.1zx^3y^2 - 17.3x^3z^3 + 10.3zx^3 - 31.1xy^2z^3 + 18.7xzy^2 + 10.4xz^3 - 6.2xz$

63 $51.9x^2y^3z^3 - 31.1zx^2y^3 - 31.1yx^2z^3 + 18.7yzx^2 - 17.3y^3z^3 + 10.4zy^3 + 10.4yz^3 - 6.2yz$

64 $102.3x^3y^3z^3 - 61.4zx^3y^3 - 61.4yx^3z^3 + 36.8yzx^3 - 61.4xy^3z^3 + 36.8xzy^3 + 36.8xyz^3 - 22.1xyz$

Bibliography

[1] Peter Davidson. *Turbulence: an introduction for scientists and engineers*. Oxford University Press, 2015.

[2] Freddie D Witherden, Antony M Farrington, and Peter E Vincent. PyFR: An open source framework for solving advection–diffusion type problems on streaming architectures using the flux reconstruction approach. *Computer Physics Communications*, 185(11):3028–3040, 2014.

[3] Brian Vermeire. *Adaptive implicit-explicit time integration and high-order unstructured methods for implicit large eddy simulation*. PhD thesis, McGill University Libraries, 2014.

[4] Air passenger market analysis. Technical report, International Air Transport Association (IATA), September 2018.

[5] Air passenger market analysis. Technical report, International Air Transport Association (IATA), October 2018.

[6] Mauro Masiol and Roy M Harrison. Aircraft engine exhaust emissions and other airport-related contributions to ambient air pollution: A review. *Atmospheric Environment*, 95:409–455, 2014.

[7] European aviation environmental report 2019. Technical report, EASA, EEA and Eurocontrol, 2019.

[8] Air passenger market analysis. Technical report, International Air Transport Association (IATA), May 2018.

[9] Jeffrey Slotnick, Abdollah Khodadoust, Juan Alonso, David Darmofal, William Gropp, Elizabeth Lurie, and Dimitri Mavriplis. CFD vision 2030 study: A path to revolutionary computational aerosciences. 2014.

[10] Andrey Nikolaevich Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. *Cr Acad. Sci. URSS*, 30:301–305, 1941.

[11] John David Anderson and J Wendt. *Computational Fluid Dynamics*, volume 206. Springer, 1995.

[12] TJ Chung. *Computational Fluid Dynamics*. Cambridge University Press, 2010.

[13] John Charles Butcher and Nicolette Goodwin. *Numerical methods for ordinary differential equations*, volume 2. Wiley Online Library, 2008.

[14] Peter Vincent, Freddie Witherden, Brian Vermeire, Jin Seok Park, and Arvind Iyer. Towards green aviation with Python at petascale. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, page 1. IEEE Press, 2016.

[15] Johannes Langguth, Nan Wu, Jun Chai, and Xing Cai. On the GPU performance of cell-centered finite volume method over unstructured tetrahedral meshes. In *Proceedings of the 3rd Workshop on Irregular Applications: Architectures and Algorithms*, page 7. ACM, 2013.

[16] John A Ekaterinaris. High-order accurate, low numerical diffusion methods for aerodynamics. *Progress in Aerospace Sciences*, 41(3-4):192–300, 2005.

[17] Bernardo Cockburn, Suchung Hou, and Chi-Wang Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case. *Mathematics of Computation*, 54(190):545–581, 1990.

[18] Bernardo Cockburn and Chi-Wang Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework. *Mathematics of Computation*, 52(186):411–435, 1989.

[19] Zhi Jian Wang. Spectral (finite) volume method for conservation laws on unstructured grids. Basic formulation: Basic formulation. *Journal of Computational Physics*, 178(1):210–251, 2002.

[20] Yen Liu, Marcel Vinokur, and ZJ Wang. Discontinuous spectral difference method for conservation laws on unstructured grids. In *Computational Fluid Dynamics 2004*, pages 449–454. Springer, 2006.

[21] David A Kopriva. A conservative staggered-grid Chebyshev multidomain method for compressible flows. II. A semi-structured method. *Journal of Computational Physics*, 128(2):475–488, 1996.

[22] Steven A Orszag and GS Patterson Jr. Numerical simulation of three-dimensional homogeneous isotropic turbulence. *Physical Review Letters*, 28(2):76, 1972.

[23] NA Adams and S Hickel. Implicit large-eddy simulation: Theory and application. In *Advances in Turbulence XII*, pages 743–750. Springer, 2009.

[24] Giancarlo Alfonsi. Reynolds-averaged Navier-Stokes equations for turbulence modeling. *Applied Mechanics Reviews*, 62(4):040802, 2009.

[25] Stephen B Pope. Turbulent flows, 2001.

[26] Hung T Huynh. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In *18th AIAA Computational Fluid Dynamics Conference*, page 4079, 2007.

[27] Brian C Vermeire, Freddie D Witherden, and Peter E Vincent. On the utility of GPU accelerated high-order methods for unsteady flow simulations: A comparison with industry-standard tools. *Journal of Computational Physics*, 334:497–521, 2017.

[28] Brian C Vermeire and Peter E Vincent. On the properties of energy stable flux reconstruction schemes for implicit large eddy simulation. *Journal of Computational Physics*, 327:368–388, 2016.

[29] Kartikey Asthana, Manuel R López-Morales, and Antony Jameson. Non-linear stabilization of high-order flux reconstruction schemes via Fourier-spectral filtering. *Journal of Computational Physics*, 303:269–294, 2015.

[30] Huynh Huynh and Leonard S Feldt. Estimation of the box correction for degrees of freedom from sample data in randomized block and split-plot designs. *Journal of Educational Statistics*, 1(1):69–82, 1976.

[31] Jan S Hesthaven and Tim Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer Science & Business Media, 2007.

[32] Jean-Baptiste Chapelier, Marta De La Llave Plata, and Florent Renac. Inviscid and viscous simulations of the Taylor-Green Vortex flow using a modal discontinuous Galerkin approach. In *42nd AIAA Fluid Dynamics Conference and Exhibit*, page 3073, 2012.

[33] Eric Johnsen, Sreenivas Varadan, and Bram Van Leer. A three-dimensional recovery-based discontinuous Galerkin method for turbulence simulations. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 515, 2013.

[34] Gregor J Gassner and Andrea D Beck. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theoretical and Computational Fluid Dynamics*, 27(3-4):221–237, 2013.

[35] Laslo Tibor Diosady and Scott M Murman. Design of a variational multiscale method for turbulent compressible flows. Technical report, NASA Ames Research Center, Moffett Field, CA, USA, 2013.

[36] Hillewaert. Direct numerical simulation of the Taylor-Green Vortex at $Re = 1600$. Technical report, 3rd International Workshop on High-Order CFD Methods, at the 53rd AIAA Aerospace Sciences Meeting in Kissimmee, Florida (Orlando), January 3-4, 2015.

[37] Farshad Navah, Marta de la Llave Plata, and Vincent Couaillier. A high-order variational multiscale approach to turbulence for compact nodal schemes. *arXiv preprint arXiv:1809.03966*, 2018.

[38] John Kim, Parviz Moin, and Robert Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 1987.

[39] Hiroyuki Abe, Hiroshi Kawamura, and Yuichi Matsuo. Direct numerical simulation of a fully developed turbulent channel flow with respect to the Reynolds number dependence. *Journal of Fluids Engineering*, 123(2):382–393, 2001.

[40] Marco E Rosti, Mohammad Omidyeganeh, and Alfredo Pinelli. Direct numerical simulation of the flow around an aerofoil in ramp-up motion. *Physics of Fluids*, 28(2):025106, 2016.

[41] Bruce Roy Munson, Theodore Hisao Okiishi, Wade W Huebsch, and Alric P Rothmayer. *Fluid Mechanics*. Wiley, Singapore, 2013.